МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Г. Ю. Чемерис

UX/UI дизайн

Навчальний посібник
для здобувачів ступеня вищої освіти бакалавра
спеціальності «Дизайн»
освітньо-професійної програми «Графічний дизайн»

Затверджено
вченою радою ЗНУ
Протокол № 8 від 30.03.2021

Запоріжжя
2021

УДК 004.451.84 (075.8)
Ч-42

Чемерис Г. Ю. UX/UI дизайн : навчальний посібник для здобувачів
Ч-42 ступеня вищої освіти бакалавра спеціальності «Дизайн» освітньо-
професійної програми «Графічний дизайн». Запоріжжя : ЗНУ, 2021. 290 с.

Навчальний посібник присвячено актуальним проблемам набуття практичних навичок з дизайну інтерфейсів та допоможе студентам якісно засвоїти програмний матеріал з навчальної дисципліни «UX/UI дизайн».

Основна мета запропонованого посібника полягає у формуванні уявлення про професію UX/UI дизайнера (дизайнера користувацьких інтерфейсів) і наданні основ для набуття навичок у цій галузі діяльності. Посібник об'єднує фундаментальні основи класичної теорії дизайну, що були подані крізь призму практичного досвіду у індустрії UX/UI дизайну та викладено у зрозумілій для студентів формі. Висвітлено основні поняття композиції, кольорознавства та типографіки для розробки інтерфейсів. Розглянуто технологічні особливості підготовки технічних документації та керівництв зі стилю. Розглянуто питання дослідження користувацького досвіду та формування «шляху користувача». Посібник містить термінологічний довідник.

Навчальний посібник укладено відповідно до робочої програми навчальної дисципліни «UX/UI дизайн» для здобувачів ступеня вищої освіти бакалавра спеціальності «Дизайн» освітньо-професійної програми «Графічний дизайн».

Рецензент

Л. Ю. Москальова доктор педагогічних наук, професор, професор кафедри дошкільної освіти і соціальної роботи Навчально-наукового інституту соціально-педагогічної та мистецької освіти, проректор з наукової роботи Мелітопольського державного педагогічного університету імені Богдана Хмельницького.

Відповідальний за випуск

Л. М. Коваль, кандидат мистецтвознавства, доцент, в. о. завідувача кафедри дизайну Запорізького національного університету.

© Чемерис Г. Ю., 2021
© Запорізький національний університет, 2021

ЗМІСТ

ВСТУП.....7
1. ТЕОРІЯ ДИЗАЙНУ. ВСТУП ДО UI ДИЗАЙНУ. ОСНОВИ UX.....10
1.1. Історія комп’ютерної графіки.....11
1.2. Дозвіл зображення і його розмір.....15
1.3. Види комп'ютерної графіки.....16
1.4. Формати графічних зображень.....20
1.5. Історія користувацького інтерфейсу.....28
1.6. Етапи роботи над створенням UI дизайну.....40
1.7. Визначення цілей та проведення досліджень.....43
1.8. Введення в проектування UX.....49
1.9. Юзабіліті в дизайні.....56
1.10. Процес проектування UX та дослідження.....58
Контрольні питання.....63
Теми для самостійного опрацювання.....63
Тестові завдання.....63
2. КОМПОЗИЦІЯ – ЯК ЗАСІБ ПОЄДНАННЯ ЕЛЕМЕНТІВ UI ДИЗАЙНУ.....65
2.1. Основи композиції.....66
2.2. Паттерни читання.....73
2.3. Точка фокусування та візуальні направляючі.....78
2.4. Принципи Гештальту.....79
2.5. Центральний візуальний образ.....80
2.7. Колір як засіб композиційної побудови.....81
2.8. Класичні принципи досягнення балансу.....82
2.9. «Повітря».....87
2.10. Використання відступів.....87
2.11. Створюємо баланс в інтерфейсах.....90
2.12. Характеристики балансу.....90
2.13. Закони взаємодії.....93
Контрольні питання.....94
Теми для самостійного опрацювання.....94
Тестові завдання.....94
Практичні завдання.....95
3. ПРОТОТИПУВАННЯ ЯК СПОСІБ ПЕРЕДПРОЕКТНОГО ДОСЛІДЖЕННЯ.....96
3.1. Мозковий штурм і пошук ідей.....96
3.2. Пошук натхнення.....99

3.3. Процес створення дизайну.....99
3.4. Планування проекту.....101
3.5. Прототипи.....103
3.6. Логічна карта сайту.....115
3.7. Карта навігації.....116
Контрольні питання.....118
Теми для самостійного опрацювання.....118
Тестові завдання.....118
Практичні завдання.....119
4. ФІЛОСОФІЯ КОЛЬОРОПОДІЛУ. ТИПОГРАФІКА ТА ОСОБЛИВОСТІ ЇЇ ВИКОРИСТАННЯ У UI ДИЗАЙНІ.....120
4.1. Моделі математичного опису кольору.....121
4.2. Кольорове коло.....128
4.3. Принципи вибору та поєднання кольорів.....129
4.4. Кольорова температура.....136
4.5. Тон, насиченість і світлосила.....142
4.6. Хроматичність та ахроматика.....145
4.7. Створення колірних схем.....145
4.8. Типографіка для Інтернет.....152
4.9. Візуалізація шрифтів.....152
4.10. Адаптація.....153
4.11. Класифікація шрифтів.....153
4.12. Основні атрибути стилю тексту.....155
4.13. Вибір і поєднання шрифтів.....159
4.14. Рекомендації по роботі з верстkою.....168
Контрольні питання.....169
Теми для самостійного опрацювання.....169
Тестові завдання.....170
Практичні завдання.....171
5. КЕРУВАННЯ КОНТЕНТОМ СТОРІНОК. ОПТИЧНИЙ БАЛАНС. ТИПОГРАФІЧНА ІЄРАРХІЯ.....172
5.1. Розмір файлів проекту і дозвіл.....172
5.2. Макет заснований на сітці.....174
5.3. Ієрархія контенту.....181
5.4. Візуальний баланс.....181
5.5. Вирівнювання.....182
5.6. Техніки ієрархізації тексту.....183
5.7. Оптичний баланс у типографіці для UI.....186

5.8. Негативний вплив перевантаження контентом.....197
Контрольні питання.....197
Теми для самостійного опрацювання.....198
Тестові завдання.....198
Практичні завдання.....199
6. ACCESSEBILITY ДИЗАЙН.....200
6.1. Вступ до доступного дизайну.....200
6.2. Законодавче регулювання.....200
6.3. Стандарти accessibility (консорціум W3C).....202
6.4. Стандарти та практика веб доступності.....202
6.5. Області порушень в Accessibility дизайні.....203
6.6. Принципи Web Content Accessibility Guidlines (WCAG).....204
6.7. Рекомендації щодо Accessebility дизайну.....205
Контрольні питання.....208
Теми для самостійного опрацювання.....208
Тестові завдання.....208
Практичні завдання.....210
7. ДИЗАЙН-ДОСЛІДЖЕННЯ. UX.....211
7.1. Типи дизайн-досліджень.....211
7.2. Основні методи дослідження користувацької аудиторії і їх сутність.....214
7.3. Більше гіпотез.....222
7.4. Відмінність ринкових досліджень від дизайн-досліджень.....226
Контрольні питання.....227
Теми для самостійного опрацювання.....227
Тестові завдання.....227
Практичні завдання.....229
8. ДИЗАЙН ПІД МОБІЛЬНІ ПРИСТРОЇ.....230
8.1. Дизайн для мобільних пристроїв.....231
8.2. Чому дизайн для мобільний пристроїв важливий.....233
8.3. Responsive Design.....233
8.4. Adaptive Design.....234
8.5. Порівняння Responsive Design та Adaptive Design.....239
8.6. Створення адаптивного веб-дизайну.....240
Контрольні питання.....248
Теми для самостійного опрацювання.....248
Тестові завдання.....248
Практичні завдання.....249

9. АНІМАЦІЯ ВЗАЄМОДІЇ UI. ЕТАП ФІНАЛІЗАЦІЇ ПРОЕКТНОЇ РОЗРОБКИ....250
9.1. Анімація і послідовності подій....251
9.2. Візуальний зворотний зв'язок....252
9.3. Переходи і зміна станів....253
9.4. Типи анімацій....254
9.5. Правила та принципи для анімації в UX/UI-дизайні....255
9.7. Хореографія анімації для UI....265
9.8. Суть анімації мікровзаємодії для макрорезультату....269
9.10. Основні правила анімації....276
Контрольні питання....277
Теми для самостійного опрацювання....277
Тестові завдання....277
Практичні завдання....278
ВІДПОВІДІ НА ТЕСТОВІ ЗАПИТАННЯ....279
ТЕЗАУРУС....280
ВИКОРИСТАНА ЛІТЕРАТУРА....288
РЕКОМЕНДОВАНА ЛІТЕРАТУРА....288
ІНФОРМАЦІЙНІ ДЖЕРЕЛА....288

ВСТУП

Навчальний посібник присвячено актуальним проблемам набуття практичних навичок з дизайну інтерфейсів та допоможе студентам якісно засвоїти програмний матеріал з навчальної дисципліни «UX/UI дизайн». Дисципліна «UX/UI дизайн» відноситься до циклу професійної підготовки здобувачів ступеня вищої освіти бакалавра спеціальності «Дизайн» освітньо-професійної програми «Графічний дизайн». Мета вивчення навчальної дисципліни «UX/UI дизайн» полягає у набутті базисних навичок з проектування користувацьких інтерфейсів та засвоєнні основ з UX/UI дизайну. Даний курс допоможе виробити навички розробки дизайну на всіх основних етапах, починаючи від вимог закінчуючи готовим користувацьким інтерфейсом. Курс побудований таким чином, щоб максимально підготувати слухача до реальних завдань, тому в ньому розглядаються сучасні підходи до розробки, а також враховуються актуальні тренди дизайну.

В процесі навчання студенти отримають теоретичну базу знань і зможуть закріпити її на практиці використовуючи сучасні інструменти і підходи які знадобляться їм для роботи на реальних проектах.

Основними **завданнями** вивчення дисципліни «UX/UI дизайн» є:

- набуття знань і практичних навичок з проектування високоякісних інтерфейсів, орієнтованих на користувача;
- формування компетентності щодо аналізу предметної області, вміння формувати, аналізувати та моделювати вимоги до програмного забезпечення;
- вироблення здатності ідентифікувати, класифікувати та описувати проектні завдання, знаходити раціональні методи й підходи до їх розв’язання;
- формування компетентності, необхідної для проектування інтерфейсів програмних систем, орієнтованих на користувача;
- формування компетентності щодо здійснення дизайн досліджень з метою покращення ергономічності дизайну інтерфейсів;
- підготовка до провадження дослідницької та/або інноваційної діяльності в галузі проектування інтерфейсів користувача.

Проблематика курсу «UX/UI дизайн» ґрунтується на знаннях, вміннях та навичках, що були опановані під час вивчення таких базових дисциплін: «Айдентика»; «Веб-дизайн»; «Шрифти і типографіка»; «Комп’ютерні технології»; «Мультимедійні технології»; «Кольорознавство»; «Основи композиції»; «Комп’ютерні технології у дизайні».

У результаті вивчення навчальної дисципліни «UX/UI дизайн» студент повинен набути таких результатів навчання (знання, уміння тощо) та компетентностей:

- Знання та розуміння предметної області, розуміння професійної діяльності.

– Здатність працювати в команді.
– Здатність оцінювати та забезпечувати якість виконуваних робіт.
– Навички використання сучасних інформаційних і комунікаційних технологій, медіаграмотність.
– Здатність генерувати нові ідеї (креативність) і приймати обґрунтовані рішення.
– Навички проєктної діяльності.
– Здатність застосовувати сучасні методики проєктування одиничних, комплексних, багатофункціональних об’єктів дизайну.
– Здатність здійснювати формоутворення, макетування і моделювання об’єктів дизайну.
– Здатність здійснювати композиційну побудову об’єктів дизайну.
– Здатність застосовувати знання історії українського і зарубіжного мистецтва та дизайну в художньо-проєктній діяльності.
– Здатність використовувати сучасне програмне забезпечення для створення об’єктів дизайну.
– Здатність здійснювати колористичне вирішення майбутнього дизайн-об’єкта.
– Знання та розуміння психології дизайну, теорій, принципів, методів і понять у сфері професійної діяльності та/або навчання.
– Здатність застосовувати в дизайн-проєктуванні здобутки шрифтової культури.
– Здатність застосовувати мультимедійні технології у професійній діяльності та навчанні.
– Застосовувати набуті знання і розуміння предметної області та сфери професійної діяльності у практичних ситуаціях.
– Збирати та аналізувати інформацію для обґрунтування дизайнерського проєкту, застосовувати теорію і методику дизайну, фахову термінологію (за професійним спрямуванням), основи наукових досліджень.
– Визначати мету, завдання та етапи проєктування.
– Розуміти і сумлінно виконувати свою частину роботи в команді; визначати пріоритети професійної діяльності.
– Усвідомлювати відповідальність за якість виконуваних робіт, забезпечувати виконання завдання на високому професійному рівні.
– Аналізувати, стилізувати, інтерпретувати та трансформувати об’єкти для розроблення художньо-проєктних вирішень.
– Оцінювати об’єкт проєктування, технологічні процеси в контексті проєктного завдання, формувати художньо-проєктну концепцію.

- Створювати об’єкти дизайну засобами проєктно-графічного моделювання.
- Розробляти композиційне вирішення об’єктів дизайну у відповідних техніках і матеріалах.
- Дотримуватися стандартів проєктування та технологій виготовлення об’єктів дизайну у професійній діяльності.
- Застосовувати сучасне загальне та спеціалізоване програмне забезпечення у професійній діяльності.
- Демонструвати вміння інтегрувати об’єкти графічного дизайну та їх окремі елементи в інформаційне та предметно-просторове середовище.
- Використовувати електронні ресурси, комп’ютерні технології та засоби мультимедіа для створення й обробки двовимірних зображень, тривимірних об’єктів, розробки, демонстрації мультимедійних продуктів та/або у процесі навчання.
- Володіти основами проєктного менеджменту (компетентностями з підготовки та реалізації дизайн-проєктів), розуміти особливості світової, європейської, загальноукраїнської проєктної культури з врахуванням регіонального аспекту.

Навчальний посібник супроводжується анімаційними матеріалами, що доступні за допомогою технології доповненої реальності (завантажити додаток можливо за посиланням is.gd/qpjgqZ, або зчитавши QR код), та позначено відповідною поміткою:





1. ТЕОРІЯ ДИЗАЙНУ. ВСТУП ДО UI ДИЗАЙНУ. ОСНОВИ UX.

План:

1.1. Історія комп’ютерної графіки
1.2. Дозвіл зображення і його розмір
1.3. Види комп'ютерної графіки
  1.3.1. Растрова графіка.
  1.3.2. Переваги растрової графіки:
  1.3.3. Недоліки растрової графіки:
  1.3.4. Векторна графіка.
  1.3.5. Переваги векторної графіки
  1.3.6. Недоліки векторної графіки
  1.3.7. Фрактальна графіка.
1.4. Формати графічних зображень
  1.4.1. Растрові веб-формати
  1.4.2. Векторні веб-формати
  1.4.3. Анімаційні веб-формати
  1.4.4. Інші формати графічних файлів
1.5. Історія користувацького інтерфейсу
1.6. Етапи роботи над створенням UI дизайну
1.7. Визначення цілей та проведення досліджень
  1.7.1. Основне завдання
  1.7.2. Визначення цільової аудиторії
  1.7.3. Складання технічного завдання
  1.7.4. Визначення типу веб-сайту
1.8. Введення в проектування UX
  1.8.1. Для чого необхідне проектування інтерфейсів
  1.8.2. Коротка історія UX
  1.8.3. Еволюція методології UX процесу
  1.8.4. Класичний дизайн користувацького досвіду
  1.8.5 Agile
  1.8.6. Економний UX (Lean UX)
  1.8.7. Дизайн спринт від Google Ventures
  1.8.8. Dual Track дизайн
  1.9. Юзабіліті в дизайні
  1.10. Процес проектування UX та дослідження
  1.11. Як вибрати метод дослідження UX?
  1.12. Як здійснюється тестування
  1.13. Чому необхідно проводити дослідження UX

1.1. Історія комп’ютерної графіки

Комп'ютерна графіка в період виникнення була далеко не настільки ефектною, якою вона стала в наші дні. Комп'ютери перебували на ранній стадії розвитку і були здатні відтворювати тільки найпростіші контури (лінії). Ідея комп'ютерної графіки не відразу була підхоплена, але її можливості швидко росли, і поступово вона стала займати одну з найважливіших позицій в інформаційних технологіях.

Першою офіційно визнаною спробою використання дисплея для виведення зображення з ЕОМ з'явилося створення в *Массачусетському технологічному університеті* машини Whirlwind-I в *1950* р Таким чином, виникнення комп'ютерної графіки можна віднести до *1950-х* років. Сам ж *термін "комп'ютерна графіка"* придумав в *1960* р співробітник компанії *Boeing У. Феттер.* Перше реальне застосування комп'ютерної графіки пов'язують з ім'ям Дж. Уїтні. Він займався кіновиробництвом в 50-60-х роках і вперше використав комп'ютер для створення титрів до кінофільму.

Наступним кроком у своєму розвитку комп'ютерна графіка зобов'язана *А. Сазерленду*, який в *1961* р, ще будучи студентом, створив програму малювання, названу їм *Sketchpad* (альбом для малювання). Програма використовувала *світлове перо* для малювання найпростіших фігур на екрані. Отримані картинки можна було зберігати і відновлювати. У цій програмі було розширене коло основних графічних примітивів, зокрема, крім ліній і точок був введений прямокутник, який задавався своїми розмірами і розташуванням.

Спочатку комп'ютерна графіка була *векторною*, тобто оригінал формувався з тонких ліній. Ця особливість була пов'язана з технічною реалізацією комп'ютерних дисплеїв. Надалі більш широке застосування отримала растрова графіка, заснована на представленні зображення на екрані у вигляді матриці однорідних елементів (пікселів).

У тому ж *1961* р студент *С. Рассел* створив першу комп'ютерну відеогру *Spacewar* ("Зоряна війна"), а науковий співробітник Bell Labs Е. Зеджек створив *анімацію* "Simulation of a two-giro gravity control system".

У зв'язку з успіхами в області комп'ютерної графіки великі корпорації почали виявляти до неї інтерес, що в свою чергу *стимулювало прогрес в області її технічної підтримки*.

Університет штату Юта стає *центром досліджень в області комп'ютерної графіки завдяки Д. Евансу і А. Сазерленду*, які в цей час були найпомітнішими фігурами в цій області. Пізніше їх коло став швидко розширюватися. Учнем Сазерленда став *Е. Кетмул*, майбутній творець алгоритму *видалення невидимих поверхонь* з використанням *Z-буфера (1978)*. Тут же працювали *Дж. Варнок*, автор алгоритму *видалення невидимих граней на основі розбиття області (1969)* і засновник *Adobe System (1982), Дж. Кларк*, майбутній засновник компанії *Silicon Graphics (1982)*. Всі ці дослідники дуже сильно просунули алгоритмічну сторону комп'ютерної графіки.

Того ж *1971* р Гольдштейн і Нагель вперше реалізували метод трасування променів з використанням логічних операцій для формування *тривимірних зображень*.

У *1970-х* роках відбувся *різкий стрибок* у розвитку обчислювальної техніки *завдяки винаходу мікропроцесора*, в результаті чого почалася мініатюризація комп'ютерів і швидке зростання їх продуктивності. І в цей же час починає інтенсивно розвиватися індустрія *комп'ютерних ігор*. Одночасно комп'ютерна графіка починає широко використовуватися на телебаченні і в кіноіндустрії. Дж. Лукас створює відділення комп'ютерної графіки на *Lucasfilm*.

У *1977* р з'являється новий журнал "Computer Graphics World".

У середині *1970-х* років графіка продовжує розвиватися в бік все *більшої реалістичності зображень*. Е. Кетмул в *1974* р створює перші алгоритми текстурування криволінійних поверхонь. У *1975* р з'являється метод зафарбовування Фонга. У *1977* р Дж. Блін пропонує алгоритми реалістичного зображення шорстких поверхонь (*мікрорельєфів*); Ф. Кроу розробляє методи усунення ступеневої ефекту при зображенні контурів (*антіалайзінг*). Дж. Брезенхему створює ефективні алгоритми побудови *растрових образів відрізків, кіл і еліпсів***.** Рівень розвитку обчислювальною техніки до цього часу вже дозволив використовувати *"жадібні" алгоритми*, що вимагають великих обсягів пам'яті, і в 1978 р Кетмул пропонує метод Z-буфера, в якому використовується область пам'яті для зберігання інформації про "глибину" кожного пікселя екранного зображення. В цьому ж році Сайрус і Бек розвивають *алгоритми кліпування (відсікання) ліній*. А в *1979* р Кей і Грінберг вперше реалізують *зображення напівпрозорої поверхні*.

У *1980* р Т. Уиттед розробляє *загальні принципи трасування променів*, що включають відображення, заломлення, затінення і методи *антіалайзінга*. У *1984* р групою дослідників (ГОРЕЛ, Торренс, Гринберг і ін.) Була запропонована модель випромінювання, одночасно розвиваються методи прямокутного кліпування областей.

У *1980-ті* роки з'являється цілий ряд компаній, що займаються прикладними розробками в області комп'ютерної графіки. У *1982* р Дж. Кларк створює Silicon Graphics, тоді ж виникає *Ray Tracing Corporation*, Adobe System, в *1986* році компанія *Pixar* відгалужується від Lukasfilm.

У ці роки комп'ютерна графіка вже міцно впроваджується в кіноіндустрію, розвиваються додатки до інженерних дисциплін. У *1990-ті* роки у зв'язку з виникненням мережі Internet у комп'ютерної графіки з'являється ще одна сфера застосування.

Слід зазначити, що пріоритет у розвитку даного напрямку в інформаційних технологіях досить міцно утримують американські дослідники. Але і у вітчизняній науці теж були свої розробки, серед яких можна назвати ряд технічних реалізацій дисплеїв, виконаних в різні роки:

1968 ВЦ АН СРСР, машина БЕСМ-6 – перший вітчизняний растровий дисплей з відеопам'яттю на магнітному барабані;

1972, Інститут автоматики і електрометрії (ІАіЕ), векторний дисплей "Символ";

1973 ІАіЕ, векторний дисплей "Дельта";

1977, ІАіЕ, векторний дисплей ЕПМ-400;

1982, Київ, НДІ периферійного обладнання, векторний дисплей СМ-7316, 4096 символів, здатність 2048

Інститут прикладної фізики, серія растрових кольорових напівтонових дисплеїв "Гамма".Останні дисплеї даної серії мали таблицю кольоровості, *підтримували вікна, плавне масштабування.*

Таким чином, в процесі розвитку комп'ютерної графіки можна виділити кілька етапів.

У *70ті* роки вона формувалася *як наукова дисципліна*. У цей час розроблялися основні методи і алгоритми: відсікання, растрова розгортка графічних примітивів, зафарбування візерунками, реалістичне зображення просторових сцен (видалення невидимих ліній і граней, трасування променів, що випромінюють поверхні), моделювання освітленості.

У *1980-і* графіка розвивається більш *як прикладна дисципліна.* Розробляються методи її застосування в самих різних областях людської діяльності.

У *1990-ті* роки методи комп'ютерної графіки стають *основним засобом* організації *діалогу "людина-комп'ютер"* і залишаються такими по теперішній час.

Розвиток комп'ютерної графіки, особливо на її початкових етапах, в першу черга пов'язано і з розвитком технічних засобів і особливо дисплеїв:

- Довільне сканування променя;
- Растрове сканування променя;
- Запам'ятовують трубки;
- Плазмова панель;
- Рідкокристалічні індикатори;
- Електролюмінесцентні індикатори;
- Дисплеї з емісією полем.

Довільне сканування променя. Дисплейна графіка з'явилася, як спроба використовувати *електропроменеві трубки* (ЕПТ) з довільним скануванням променя для виведення зображення зі ЕОМ. Як пише *Ньюмен* "мабуть, першою машиною, де ЕПТ використовувалася в якості пристрою виводу була ЕОМ Whirlwind-I (Ураган-I), виготовлена в *1950р*. В Массачусетському технологічному інституті. З цього експерименту почався етап розвитку векторних дисплеїв (дисплеїв з довільним скануванням променя, каліграфічних дисплеїв). На професійному жаргоні вектором називається відрізок прямої. Звідси і походить назва «векторний дисплей».

При переміщенні променя по екрану в точці, яку потрапив промінь, збуджується світіння люмінофора екрану. Це свічення досить швидко припиняється під час переміщення променя в іншу позицію (звичайний час післясвітіння – менше 0.1 с). Тому, для того щоб зображення було постійно видимим, доводиться його перевидавать (регенерувати зображення) 50 або 25 разів в секунду. Необхідність перевидачі зображення вимагає збереження його

опису в спеціально виділеній пам'яті, званої пам'яттю регенерації. Сам опис зображення називається дисплейним файлом. Зрозуміло, що такий дисплей вимагає досить швидкого процесора для обробки дисплейного файлу і управління переміщенням променя по екрану.

Зазвичай серійні *векторні дисплеї* встигали 50 раз в секунду будувати тільки близько 3000-4000 відрізків. При більшій кількості відрізків зображення починає мерехтіти, так як відрізки, побудовані на початку чергового циклу, повністю згасають до того моменту, коли будуть будуватися останні.

Іншим недоліком векторних дисплеїв є мале число градацій по яскравості (зазвичай 2-4). Були розроблені, але не знайшли широкого застосування двох-триколірні ЕПТ, також забезпечували кілька градацій яскравості.

У векторних дисплеях легко стерти будь-який елемент зображення – досить при черговому циклі побудови видалити стирається елемент з дисплейного файлу.

Текстовий діалог підтримується за допомогою алфавітно-цифрової клавіатури. Непрямий графічний діалог, як і у всіх інших дисплеях, здійснюється переміщенням перехрестя (курсора) по екрану за допомогою тих чи інших засобів управління перехрестям – координатних коліс, керуючого важеля (джойстика), трекболу (кульовий рукоятки), планшета тощо. Відмітною рисою векторних дисплеїв є можливість безпосереднього графічного діалогу, що полягає в простому вказівці за допомогою світлового пера об'єктів на екрані (ліній, символів тощо). Для цього досить з допомогою фотодіода визначити момент промальовування і, отже, почала світіння люмінофора) будь-якій частині необхідного елемента.

Перші серійні *векторні дисплеї* за кордоном з'явилися в кінці 60-х років.

Растрове сканування променя. Прогрес в технології мікроелектроніки привів до того, з середини 70-х років переважна поширення набули дисплеї з растровим скануванням променя.

Запам'ятовуючі трубки. В кінці 60-х років з'явилася ЕПТ, яка здатна досить тривалий час (до години) прямо на екрані зберігати побудоване зображення. Отже, не обов'язкова пам'ять регенерації і не потрібен швидкий процесор для виконання регенерації зображення. Стирання на такому дисплеї можливо тільки для всієї картинки в цілому. Складність зображення практично не обмежена. Дозвіл, досягнуте на дисплеях на пам'ятною трубці, таку ж, як і на векторних або вище – до 4096 точок.

Текстовий діалог підтримується за допомогою алфавітно-цифрової клавіатури, непрямий графічний діалог здійснюється переміщенням перехрестя по екрану зазвичай за допомогою координатних коліс.

Поява таких дисплеїв з одного боку сприяло широкому поширенню комп'ютерної графіки, з іншого боку представляло собою певний регрес, так як поширювалася порівняно низькоякісна і низько-швидкісна, не дуже інтерактивна графіка.

Плазмова панель. У 1966р. була винайдена плазмова панель, яку можна представити як матрицю з маленьких різнокольорових неонових лампочок,

кожна з яких включається незалежно і може світитися з регульованою яскравістю. Ясно, що системи відхилення не потрібно, не є обов'язковою також і пам'ять регенерації, так як за напругою на лампочці можна завжди визначити горить вона чи ні, т. Е. Є чи ні зображення в даній точці. У певному сенсі ці дисплеї об'єднують в собі багато корисних властивостей векторних і растрових пристроїв. На недоліків слід віднести велику вартість, недостатньо висока роздільна здатність і велика напруга живлення. В цілому ці дисплеї не знайшли широкого поширення.

*Рідкокристалічні індикатори***.** Дисплеї на рідкокристалічних індикаторах працюють аналогічно індикаторам в електронних годинниках, але, звичайно, зображення складається не з декількох сегментів, а з великого числа окремо керованих точок. Ці дисплеї мають найменші габарити і енергоспоживання, тому широко використовуються в портативних комп'ютерах незважаючи на менший дозвіл, меншу контрастність і помітно більшу ціну, ніж для растрових дисплеїв на ЕПТ.

Електролюмінісцентні індикатори. Найбільш високі яскравості, контрастності, робочий температурний діапазон і міцність мають дисплеї на електролюмінісцентних індикаторах. Завдяки досягненням в технології вони почали доступні для застосування не лише в дорогих висококласних системах, але і в загальнопромислових системах. Робота таких дисплеїв заснована на світінні люмінофора під впливом щодо високої змінної напруги, що прикладається до взаємно наборам електродів, між якими знаходиться люмінофор.

*Дисплеї з емісією поля***.** Дисплеї на електронно-променевих трубках, незважаючи на їх відносну дешевизну і широке поширення, механічно неміцні, вимагають високої напруги харчування, споживають велику потужність, мають великі габарити і обмежений термін служби, пов'язаний з втратою емісії катодом. Одним з методів усунення вказаних недоліків, є створення плоских дисплеїв з емісією полем з холодних катодів в вигляді сильно загострених мікроголок.

Таким чином, стартувавши в *1950р.*, комп'ютерна графіка до теперішнього часу пройшла шлях від екзотичних експериментів до одного з найважливіших, всепроникаючих інструментів сучасної цивілізації, починаючи від наукових досліджень, автоматизації проектування і виготовлення, бізнесу, медицини, екології, засобів масової інформації, дозвілля і закінчуючи побутовим обладнанням.

1.2. Дозвіл зображення і його розмір

Слід чітко розрізняти *дозвіл екрана, дозвіл друкувального пристрою і дозвіл зображення*. Всі ці поняття відносяться до різних об'єктів. Один з одним ці види дозволу ніяк не зв'язані, поки не буде потрібно довідатися, який фізичний розмір буде мати зображення на екрані монітора, відбиток на папері або файл на дисковому просторі.

Роздільна здатність екрану – властивість комп'ютерної відеосистеми (залежить від параметрів монітора і відеокарти) і операційної системи

(залежить від налаштувань Windows). Дозвіл екрану вимірюється в пікселях на дюйм *(ppi – pixel per inch)* і визначає розмір зображення, яке може бути розміщено на екрані цілком.

Дозвіл друкованого відбитку – властивість принтера, що виражає кількість окремих крапок, які можуть бути надруковані на ділянці одиничної довжини. Воно вимірюється в одиницях *dpi (dots per inch – точки на дюйм***)** і визначає розмір зображення при заданій якості або, навпаки, якість зображення при заданому розмірі.

Дозвіл зображення – властивість самого зображення. Воно вимірюється в точках на дюйм (dpi) при створенні зображення на графічному редакторі або за допомогою сканера. Значення дозволу зображення зберігається у файлі зображення і нерозривно пов'язане з іншою властивістю зображення – *його фізичним розміром.*

Фізичний розмір зображення. Може вимірюватися як в пікселах, так і в одиницях довжини (міліметрах, сантиметрах, дюймах). Він задається при створенні зображення і зберігається разом з файлом.

Якщо зображення готують для демонстрації на екрані, то його ширину і висоту задають у пікселях, щоб знати, яку частину екрана воно займає. Якщо зображення готують для друку, то його розмір задають в одиницях довжини, щоб знати, яку частину аркуша паперу воно займе. Неважко перерахувати розмір зображення з пікселів в одиниці довжини і навпаки, якщо відомо дозвіл зображення (Табл. 1.1.):

Таблиця 1.1. Відповідність дозволу та розмірів

Розмір ілюстрації в пікселях	**Розмір відбитка, мм при дозволах**			
75 dpi	**150 dpi**	**300 dpi**	**600 dpi**	**1200 dpi**
640x480	212x163	108x81	55x40	28x20
800x600	271x203	136x102	68x51	34x26
1024x768	344x260	173x130	88x66	44x33
1152x864	390x292	195x146	98x73	49x37
1600x1200	542x406	271x203	136x102	68x51

1.3. Види комп'ютерної графіки

Незважаючи на те, що для роботи з КГ існує безліч класів програмного забезпечення, виділяють всього три види КГ: растрову, векторну і фрактальну графіку. Вони розрізняються принципами формування зображення при відображенні на екрані монітора або при друці на папері.

Растрова графіка застосовується при розробці електронних і поліграфічних видань. Ілюстрації, виконані засобами растрової графіки, рідко створюють вручну за допомогою комп'ютерних програм. Найчастіше для цієї

мети сканують ілюстрації, підготовлені художником на папері, або фотографії. Останнім часом для введення растрових зображень в комп'ютер широко використовують цифрові фото- і відеокамери. Відповідно, більшість графічних редакторів, призначених для роботи з растровими ілюстраціями, *орієнтовані не стільки на створення зображення, скільки на їх обробку. В Інтернеті, в основному, застосовуються растрові ілюстрації.*

Програмні засоби для роботи з векторною графікою навпаки призначені, в першу чергу, *для створення ілюстрацій і в меншій мірі для їх обробки. Такі засоби широко використовують в рекламних агентствах*, дизайнерських бюро, редакціях і видавництвах. Оформлювальні роботи, засновані на застосуванні шрифтів і найпростіших геометричних елементів, вирішуються засобами векторної графіки набагато простіше. Існують приклади високохудожніх творів, створених засобами векторної графіки, але вони скоріше виключення, ніж правило, оскільки художня підготовка ілюстрацій засобами векторної графіки надзвичайно складна.

Програмні засоби для роботи з фрактальною графікою призначені *для автоматичної генерації зображень шляхом математичних розрахунків. Створення фрактальної художньої композиції полягає не в малюванні або оформленні, а в* програмуванні. Фрактальну графіку рідко застосовують для створення друкованих або електронних документів, але її часто використовують у розважальних програмах.

1.3.1. Растрова графіка.

Основним елементом растрового зображення є точка. Якщо зображення екранне, то ця точка називається пікселем. Відмінними рисами пікселя є його однорідність (всі пікселі за розміром однакові) і неподільність (піксель не містить більш дрібних пікселів). Залежно від того, на яке графічне роздільна здатність екрану налаштована операційна система комп'ютера, на екрані можуть розміщуватися зображення, які мають 640x480, 800x600, 1024x768 і більш пікселів.

З розміром зображення безпосередньо пов'язано його дозвіл. Цей параметр вимірюється у точках на дюйм (dots per inch – dpi). У монітора з діагоналлю 15 дюймів розмір зображення на екрані складає приблизно 28x21 см. Знаючи, що в 1 дюймі 25,4 мм, можна розрахувати, що при роботі монітора в режимі 800x600 пікселів дозвіл екранного зображення дорівнює 72 dpi.

При печатці дозвіл має бути набагато вище. Поліграфічний друк повноколірного зображення вимагає дозволу не менше 300 dpi. Стандартний фотознімок розміром 10x15 см повинен містити приблизно 1000x1500 пікселів.

Колір будь-якого пікселя растрового зображення запам'ятовується в комп'ютері за допомогою комбінації бітів. Чим більше бітів, тим більше відтінків кольорів можна отримати. Число бітів, використовуваних комп'ютером для кожного пікселя, називається *бітової глибиною пікселя*. Найбільш просте растрове зображення, що складається з пікселів мають тільки два кольори – чорний і білий, називається *однобітовими зображеннями.* Число доступних кольорів або градацій сірого кольору дорівнює 2k рівній кількості бітів в

пікселі. Кольори, що описуються 24 бітами, забезпечують більше 16 мільйонів доступних кольорів і їх називають *природними кольорами*.

Растрові зображення володіють безліччю характеристик, які повинні бути організовані і фіксовані комп'ютером. Розміри зображення і розташування пікселів в ньому це дві основні характеристики, які файл растрових зображень повинен зберегти, щоб створити картинку. Навіть якщо зіпсована інформація про колір будь-якого пікселя і будь-яких інших характеристиках комп'ютер все одно зможе відтворити версію малюнка, якщо буде знати, як розташовані всі його пікселі. Піксель сам по собі не має розміром, він всього лише область пам'яті комп'ютера, що зберігає інформацію про колір, тому коефіцієнт прямокутності зображення (визначає кількість пікселів матриці малюнка по горизонталі і по вертикалі) не відповідає ніякої реальної розмірності. Знаючи тільки коефіцієнт прямокутності зображення з деякою роздільною здатністю можна визначити справжні розміри малюнка.

Роздільна здатність растра – це просто число елементів (пікселів) заданої області (дюйм). Файли растрової графіки займають велику кількість пам'яті комп'ютера. Найбільший вплив на кількість пам'яті надають три фактори:

1. розмір зображення;
2. бітова глибина кольору;
3. формат файлу, який використовується для зберігання зображення.

1.3.2. Переваги растрової графіки:

1. апаратна реалізація;
2. програмна незалежність (формати файлів, призначені для збереження точкових зображень, є стандартними, тому не мають вирішального значення, в якому графічному редакторі створено те чи інше зображення);
3. фотореалістичність зображень.

1.3.3. Недоліки растрової графіки:

1. значний обсяг файлів (визначається твором площі зображення на дозвіл і на глибину кольору (якщо вони приведені до єдиної розмірності);
2. принципові складності трансформування піксельних зображень;
3. ефект пікселізації – пов'язаний з неможливістю збільшення зображення для розгляду деталей. Оскільки зображення складається з точок, то збільшення призводить до того, що точки стають більшими. Ніяких додаткових деталей при збільшенні растрового зображення розглянути не вдається, а збільшення точок растру візуально спотворює ілюстрацію і робить її грубою;
4. апаратна залежність – причина багатьох помилок;
5. відсутність об'єктів.

1.3.4. Векторна графіка.

Якщо в растровій графіці основним елементом зображення є точка, то у векторній графіці – лінія (при цьому неважливо, пряма це лінія крива).

Зрозуміло, в растровій графіці теж існують лінії, але там вони розглядаються як комбінації точок. Для кожної точки лінії в растровій графіці відводиться одна або кілька осередків пам'яті (чим більше кольорів можуть мати точки, тим більше клітинок їм виділяється). Відповідно, чим довше

растрова лінія, тим більше пам'яті вона займає. У векторній графіці об'єм пам'яті, займаний лінією, не залежить від розмірів лінії, оскільки вона представляється у вигляді формули, а точніше кажучи, у кількох параметрів. Що б ми не робили з цією лінією, змінюються лише її параметри, що зберігаються в осередках пам'яті. Кількість же осередків залишається незмінним для будь-якої лінії.

Лінія – це елементарний об'єкт векторної графіки. Все, що є у векторній ілюстрації, складається з ліній. Найпростіші об'єкти об'єднуються в більш складні (наприклад, об'єкт чотирикутник можна розглядати як чотири пов'язані лінії, а об'єкт куб ще більш складний: його можна розглядати або як 12 пов'язаних ліній, або як 6 пов'язаних чотирикутників). Через такого підходу векторну графіку часто називають об'єктно-орієнтованої графікою.

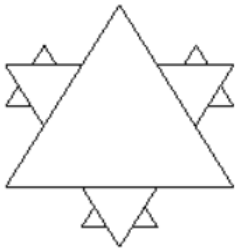


Рисунок 1.1 – Фрактал

Приклад. У загальному випадку рівняння кривої третього порядку можна записати у вигляді

$$x^3 + a_1 y^3 + a_2 x^2 y + a_3 x y^2 + a_4 x^2 + a_5 y^2 + a_6 x y + a_7 x + a_8 y + a_9 = 0.$$

Видно, що для запису досить дев'яти параметрів. Для завдання відрізка кривої третього порядку треба мати на два параметри більша. Якщо додати до них параметри, що виражають такі властивості лінії, як товщина, колір, характер та інше, то для зберігання одного об'єкта достатньо буде 20-30 байтів оперативної пам'яті. Досить складні композиції, що налічують тисячі об'єктів, витрачають лише десятки і сотні Кбайт.

Як і всі об'єкти, лінії мають ***властивості****:* форма лінії, її товщина, колір, характер (суцільна, пунктирна тощо) Замкнені лінії мають властивість заповнення. Внутрішня область замкнутого контуру може бути заповнена кольором, текстурою, картою. Найпростіша лінія, якщо вона не замкнута, має дві вершини, які називаються вузлами. Вузли теж мають властивості, від яких залежить, як виглядає вершина лінії і як дві лінії сполучаються між собою.

Зауважимо, що об'єкти векторної графіки зберігаються в пам'яті як набору параметрів, але на екран всі зображення все одно виводяться у вигляді точок (просто тому, що екран так влаштований). Перед виведенням на екран кожного об'єкта програма робить обчислення координат екранних точок в зображенні об'єкта, тому векторну графіку іноді називають *обчислюється графікою. Аналогічні обчислення проводяться і при виведенні об'єктів на принтер.*

1.3.5. Переваги векторної графіки

1. повна свобода трансформації (зміна масштабу без втрати якості і практично без збільшення розмірів вихідного файлу);
2. величезна точність;
3. невеликий розмір файлу в порівнянні з растровим зображенням;
4. прекрасну якість друку;
5. відсутність проблем з експортом векторного зображення в растрове;

6. об'єктно-орієнтований характер векторної графіки (можливість редагування кожного елемента зображення окремо);

7. апаратна незалежність.

1.3.6. Недоліки векторної графіки

1. відсутність апаратної реалізованості;

2 програмна залежність;

3. практично неможливо експортувати з реєстрового формату у векторний (можна, звичайно, трасувати зображення, хоча отримати хорошу векторну картинку нелегко);

4. неможливе застосування великої бібліотеки ефектів, використовуваних при роботі з растровими зображеннями.

1.3.7. Фрактальна графіка.

Фрактальна графіка, як і векторна – *обчислюється,* але відрізняється від неї тим, що ніякі об'єкти в пам'яті комп'ютера не зберігається. Зображення будується за рівнянням (або по системі рівнянь), тому нічого, крім формули, зберігати не треба. Змінивши коефіцієнти у рівнянні, можна отримати зовсім іншу картину.

Розглянемо приклад побудови фрактального зображення. Найпростішим фрактальним об'єктом є *фрактальний трикутник* (Рисунок 1.1).

1. Побудуйте звичайний рівносторонній трикутник зі стороною *a*.

2. Розділіть кожну з його сторін на три відрізка.

3. На середньому відрізку сторони побудуйте рівносторонній трикутник зі стороною, рівною 1/3 сторони початкового трикутника.

4. З отриманими трикутниками повторіть ті ж операції.

З малюнка видно, що трикутники наступних поколінь *успадковують властивості* своїх батьківських структур. Так народжується фрактальна фігура.

Процес спадкування можна продовжувати до нескінченності. Фрактальні властивості мають багато об'єктів живої і неживої природи. Звичайна сніжинка, багаторазово збільшена, виявляється фрактальним об'єктом. Фрактальні алгоритми лежать в основі росту кристалів і рослин. Погляньте на гілку папоротевого рослини, і ви побачите, що кожна дочірня гілка багато в чому повторює властивості гілки більш високого рівня.

Здатність фрактальної графіки моделювати образи живої природи обчислювальним шляхом часто використовують для автоматичної генерації незвичайних ілюстрацій.

1.4. Формати графічних зображень

У підготовці зображень для веб-сторінок варто бути обізнаним в популярних графічних форматах та їх властивостях.

На щастя, сьогодні немає такого різноманіття форматів, як на початку 90-х, коли кожна компанія-виробник графічних редакторів створювала свої файлові формати, проте кожен, з усталених на сьогодні форматів, пройшов природний відбір, довів свою життєздатність і користь. Вони мають характерні особливості і можливості, що роблять їх незамінними в роботі.

Комп’ютерні графічні файли можна поділити на дві великі гілки: растрову і векторну.

- **Векторні файли** представляють математичний опис об'єктів відносно точки відліку координат. Наприклад, для того, щоб відобразити пряму потрібно вказати координати двох точок, які об'єднаються за коротшим шляхом, для дуги задається радіус тощо. Таким чином, векторне зображення є набором геометричних примітивів. Більшість векторних форматів можуть містити втілені у файл растрові об'єкти. Складність при переведенні чи перенесенні даних з одного векторного формату до іншого полягає у використанні в програмах різних алгоритмів, різних математичних формул для побудови векторних примітивів та опису растрових об'єктів.
- **Растровий файл** влаштовано простіше. Він представляє прямокутну матрицю (*bitmap*), що поділена на пікселі. Растрові формати різняться між собою здатністю містити додаткову інформацію: різні колірні моделі, вектори, Альфа-канали, прошарки різних типів, інтерліньяж (черезрядкове підвантаження), анімацію, можливості стиснення тощо.

1.4.1. Растрові веб-формати

1.4.1.1. GIF (CompuServe Graphics Interchange Format)

https://www.w3.org/Graphics/GIF/spec-gif89a.txt

Апаратно незалежний формат GIF розроблено в 1987 році (GIF87a) фірмою CompuServe для передачі растрових зображень по мережах. У 1989-му формат модифіковано (GIF89a), до нього додано підтримку прозорості та анімації.

GIF використовує LZW-компресію, що дозволяє добре стискати файли, в яких багато ділянок з однорідним заповненням (логотипи, написи, схеми).

Метод стиснення LZW (Lempel-Ziv-Welch) розроблено в 1978 році ізраїльтянами Лемпелом і Зівом, і допрацьовано пізніше в США. Принцип стискання полягає у пошуку однакових послідовностей – фраз у всьому файлі. Виявлені послідовності зберігаються в таблиці, їм привласнюються короткі маркери – ключі. Так, якщо в зображенні є набори з рожевого, оранжевого і зеленого пікселів, що повторюються 50 разів, LZW виявляє цей набір, привласнює йому окреме число (наприклад, 7) і зберігає ці дані 50 разів у вигляді числа 7.

Метод LZW ефективно діє на ділянках однорідних, вільних від шуму кольорів, добре стискає довільні графічні дані, але процес кодування і розпаковування відбувається відносно повільно.

GIF дозволяє записувати зображення «через рядок» (Interlaced), завдяки чому, маючи лише частину файлу, можна побачити зображення цілком, але з меншою роздільністю. Це досягається за рахунок завантаження, спочатку 1, 5, 10 і далі рядків, за другим проходом підвантажуються 2, 6, 11 рядки, і згодом зображення набуває початкового вигляду. Черезрядковий запис дещо збільшує розмір файлу, але надає більшої зручності для користувачів. У GIF можна

застосовувати прозорі ділянки, вони лишаються прозорими в браузерах та інших програмах і через них просвічується фоновий колір. Прозорість забезпечується за рахунок додаткового Alpha-каналу, що зберігається разом з файлом. Файл GIF спроможні містити кілька растрових картинок, які браузери підвантажують одну за іншою із вказаною у файлі частотою. Так досягається ілюзія руху (GIF-анімація).

Особливості

- Кількість кольорів в зображенні може бути від 2 до 256, але це можуть бути будь-які кольори з 24-бітової палітри.
- Файл у форматі GIF може містити лише 100% прозорі ділянки. Якщо використовується відмінний від білого кольору фон, він буде просвічуватися крізь прозорі ділянки в зображенні.
- GIF підтримує покадрову зміну зображень, що робить формат популярним для створення банерів і простої анімації.
- Використовує вільний від втрат метод стиснення.

Область застосування

Текст, логотипи, ілюстрації з чіткими краями, анімовані малюнки, зображення з прозорими ділянками, банери.

1.4.1.2. JPEG (Joint Photographic Experts Group)

https://jpeg.org/

JPEG це є назва формату та алгоритму стиснення, який засновано не на пошуку однакових елементів, як у LZW, а на різниці між пікселями.

Кодування даних відбувається в кілька етапів. Спочатку графічні дані конвертуються в колірний простір типу LAB, відкидається половина або три чверті інформації про колір (в залежності від реалізації алгоритму).

Далі аналізуються блоки 8x8 пікселів. Для кожного блоку формується набір чисел. Перші кілька чисел представляють колір блоку в цілому, в той час, як подальші числа відображають тонкі деталі.

На наступному етапі, залежно від обраного рівня якості, відкидається певна частина чисел, що представляють тонкі деталі. На останньому етапі використовується кодування за методом Хафмана для ефективного стиснення кінцевих даних. Відновлення даних відбувається в зворотному порядку. Метод стиснення Хафмана (Huffman) розроблено в 1952 році і використовується як складова частина в ряді інших схем стиснення, в тому числі і у LZW. В методі Хафмана аналізується набір символів для визначення частоти кожного символу. Для символів, що найчастіше зустрічаються, використовується позначення у вигляді мінімальної можливої кількості бітів. Наприклад, найчастіше в англійських текстах зустрічається буква "e". Використовуючи кодування Хафмана можна представити літеру "e" лише двома бітами (1 і 0), замість вісьмох бітів, необхідних для представлення букви "e" в кодуванні ASCII.

Таким чином, чим вище рівень компресії, тим більше даних відкидається, тим нижчою є якість. Використовуючи JPEG можна отримати файл в 2-500 разів менше, ніж BMP. Формат JPEG є апаратно незалежним, повністю підтримується

на PC і Macintosh. JPEG відтворює спектр кольорів TrueColor (2^{24}). JPEG краще стискає растрові картинки фотографічної якості, ніж логотипи або схеми – в них більше півтонових переходів, в той час серед однотонних заливок з'являються небажані переходи. Краще і з меншими втратами стискаються великі зображення для Веб з високою роздільністю (200-300 dpi і більше), ніж з низькою (72-150 dpi), оскільки в кожному квадраті 8x8 пікселів переходи виходять м'якшими, за рахунок того, що таких квадратів є більше.

Даний формат називають стисненням з втратами, оскільки алгоритм JPEG вибірково відкидає дані. Не бажано зберігати у JPEG-форматі будь-які зображення, де важливими є тонкі нюанси кольорів, оскільки під час стиснення відбувається відкидання колірної інформації. У JPEG слід зберігати лише кінцевий варіант роботи, оскільки кожне повторне збереження призводить до нових втрат (відкидання) даних і початкове зображення може бути вкрай зіпсованим. Формат JPEG не підтримує прозорість і при збереженні зображення з прозорими ділянками, вони зафарбовуються в певний колір.

Особливості

- В зображенні може бути понад 16 мільйонів кольорів, що цілком достатньо для збереження фотографічних зображень.
- Основною характеристикою формату є якість, яка визначає кінцевий розмір файлу. Слід пам’ятати, що формат застосовує стиснення з втратами. Чим вище стиснення, тим менше якість і навпаки.
- Підтримка технології прогресивного JPEG. Спочатку у вікні перегляду з'являється версія зображення з низькою роздільністю, яке при повному завантаженні поступово набуває початкового вигляду.

Область застосування

Використовується переважно для фотографій. Не є доцільним для зображень з прозорими ділянками, великими одноколірними ділянками.

1.4.1.3. PNG (Portable Network Graphics)

http://www.libpng.org/pub/png/

PNG – Інтернет формат, який долає обмеження GIF. Використовує стиснення без втрат Deflate, подібне до LZW. Стиснуті індексовані файли PNG, зазвичай, є меншими за аналогічні GIF.

Глибина кольору може бути любою, до 48 біт. Використовується двовимірний interlacing (не лише рядків, але і стовпців), що, подібно до GIF, дещо збільшує розмір файлу. На відміну від GIF, де застосовується 100% прозорість, PNG підтримує також напівпрозорі пікселі (в діапазоні прозорості від 0 до 99%) за рахунок Альфа-каналу з 256 градаціями сірого.

У файл формату PNG записується інформація про гамму. Гамма є певним числом, що характеризує залежність яскравості світіння екрану монітора від напруги на електродах кінескопа. Це число обчислюється з файлу і дозволяє вводити поправку яскравості при відображенні. Воно потрібне для однакового відображення інформації незалежно від апаратної платформи комп’ютера. PNG підтримується у всіх сучасних браузерах.

1.4.1.3.1. PNG-8

PNG-8 — формат подібний до GIF, має покращений формат стиснення даних.

Особливості

- Використовує 8-бітову палітру (256 кольорів) в зображенні, за що і отримав в своїй назві цифру вісім. При цьому можна вибирати, скільки кольорів буде задіяно у файлі — від 2 до 256.
- На відміну від GIF, не відображає анімацію.

Область застосування

Текст, логотипи, ілюстрації з чіткими краями, зображення з градієнтною прозорістю.

1.4.1.3.2. PNG-24

PNG-24 — формат, аналогічний до PNG-8, але використовує 24-бітову палітру кольору Подібно до формату JPEG, зберігає яскравість і відтінки кольорів у фотографіях. Подібно до форматів GIF і PNG-8, зберігає деталі зображення та прозорість.

Особливості

- Використовує понад 16 млн. кольорів, тому, застосовується для повнокольорових зображень.
- Підтримує багаторівневу прозорість, це дозволяє створювати плавний перехід від прозорої області зображення до колірної, так званий градієнт.
- Алгоритм стиснення зберігає всі кольори і пікселі в зображенні незмінними. Якщо порівнювати з іншими форматами, то в PNG-24 кінцевий об'єм графічного файлу виходить найбільшим.

Область застосування

Фотографії, малюнки, що містять прозорі ділянки, малюнки з великою кількістю кольорів і чіткими краями зображень.

1.4.2. Векторні веб-формати

1.4.1.1. SVG. Масштабована векторна графіка



SVG (Scalable Vector Graphics) – це тип векторних файлів, що описують зображення в форматі XML. Формат з'явився в 2001, однак популярність серед веб-розробників він отримав нещодавно, після впровадження підтримки у сучасні браузери. Формат є відкритим стандартом, на відміну від більшості інших форматів, SVG не є чиєюсь власністю.

Файл із зображенням в цьому форматі є звичайним текстовим файлом, який можна відкрити в блокноті і відредагувати. У цьому форматі можна описати не тільки статичну, а й динамічну картинку (анімація), змішати створені вектори з растровою картинкою. Завдяки тому, що кожна фігура для браузера є елементом DOM, за допомогою JavaScript можна описувати досить складні сценарії, взаємодіяти з користувачем.

Розмір об'єктів SVG є меншим за розмір растрових зображень, а самі зображення не втрачають якості при масштабуванні. На відміну від растрових форматів можна взаємодіяти із зображеннями у форматі SVG – за допомогою

CSS можна змінювати параметри графіки: колір, прозорість або межі, а за допомогою JavaScript – анімувати зображення. У SVG є маса функцій, які роблять цей формат рекомендованим для Вебу, особливо якщо SVG використовується для простих зображень типу логотипів, карт, іконок, маркерів.

Переваги формату SVG

- SVG часто важать менше за растрові зображення.
- Формат масштабується, що забезпечує чіткість за любої роздільності екрану.
- SVG-код можна помістити в HTML і налаштовувати через CSS.
- SVG-зображення можна анімувати, в тому числі окремі частини, як за допомогою CSS, так і JS.

Втім занадто складні SVG-зображення збільшують розмір файлу. SVG не застосовний до фотографій, тут краще підійдуть формати JPG і webP.

1.4.3. Анімаційні веб-формати

1.4.3.1. APNG. Анімований PNG



aPNG (Animated Portable Network Graphics) запропоновано в 2004 році компанією Mozilla. Новий стандарт базується на форматі PNG, додано можливість анімації та покращено алгоритми стиснення. Втім компанія, що підтримує формат PNG не визнала новий стандарт, і aPNG не пішов в реліз. Основна причина відмови звучала так «PNG – це формат для нерухомих зображень».

До 2008 року не було жодних спроб широкого впровадження aPNG. В 2008 році, компанія Mozilla вносить його в свій браузер Firefox, пізніше таку підтримку додано в браузер Opera. З приходом HTML 5, ситуація почала змінюватися. Google, Twitter, Facebook і інші популярні сервіси, стали один за одним відмовлятися від застарілих технологій, і переходити на нові, паралельно, задаючи стандарти розробки.

1.4.3.2. WebP



https://developers.google.com/speed/webp/

Сучасний формат зображень з відкритим вихідним кодом, розроблений компанією Google спеціально для Інтернету, є відгалуженням відеоформату WebM.

Подібно до APNG, він також підтримує 24-бітні зображення, 8-розрядну прозорість. У ньому є стиснення з втратами та без втрат, що у деяких випадках дозволяє досягти невеликого розміру файлів із забезпеченням високої якості, що робить його універсальним. Формат поєднує переваги JPG і PNG без збільшення розміру файлу. Сьогодні YouTube використовує перетворення мініатюр для відео в формат WebP.

Поки формат підтримується лише браузером Chrome, і має певні недоліки щодо відображення дрібних деталей. Для підтримки в інших браузерах існують способи обходу обмежень (додаткові плагіни або конфігурація), але вони не сприяють використанню формату повсюдно.

1.4.3.3. FLIF Free Lossless формат зображення

http://flif.info/



FLIF – це новий та інноваційний формат зображення, що підтримує анімацію з альфа-прозорістю, прогресивне завантаження (що дозволяє відображати зображення в меншій якості, поки він довантажується), а також інші можливості у порівнянні з різними форматами. Формат поки не підтримується всіма браузерами, але розробляються JavaScript-плагіни, що дозволяють використовувати FLIF у будь-якому браузері.

1.4.4. Інші формати графічних файлів

1.4.4.1. PSD (Adobe Photoshop Document)



Внутрішній формат популярного растрового редактора Photoshop останнім часом підтримується великою кількістю програм. Він дозволяє записувати зображення з багатьма прошарками, їх масками, додатковими Альфа-каналами і каналами базових кольорів, контурами та іншою інформацією.

Для стиснення застосовують метод RLE (Run Length Encoding), кодування із змінною довжиною рядка. Дія методу полягає в пошуку однакових пікселів в одному рядку. Якщо в рядку, наприклад, є 3 пікселя білого кольору, 21 – чорного, 14 – білого, то застосування RLE надає можливість не запам'ятовувати кожен з них (38 пікселів), а записати як 3 білих, 21 чорний і 14 білих в першому рядку.

Подібно до методу LZW, алгоритм RLE добре працює з штучними і ілюстрованими картинками і гірше з фотографіями. У випадку, якщо фотографія має багато дрібних деталей, RLE може навіть збільшити розмір файлу.

1.4.4.2. TIFF (Tagged Image File Format)



Апаратно незалежний формат TIFF, на сьогоднішній, день є одним з найпоширеніших і надійніших, його підтримують практично всі програми на PC і Macintosh, що пов’язані з графікою. TIFF є кращим вибором при імпорті растрової графіки у векторні програми та видавничі системи. Йому доступно весь діапазон колірних моделей від монохромної до RGB, CMYK і додаткових кольорів Pantone. TIFF може зберігати контури, Альфа-канали та інші додаткові дані.

TIFF має два різновиди: для Macintosh і PC. Це пов'язано з тим, що процесори Motorola читають і записують числа зліва направо, а процесори Intel – навпаки. Сучасні програми можуть без проблем використовувати обидва варіанти формату. Зазвичай, дані у форматі TIFF не стискаються, але може бути використано LZW-стиснення.

1.4.4.3. BMP (Windows Device Independent Bitmap)



Рідний формат Windows, який підтримується всіма графічними редакторами, що працюють під управлінням цієї операційної системи. Застосовується, в основному, для збереження растрових зображень, що призначені для використання в Windows. Може зберігати як індексовані (до 256 кольорів), так і RGB-кольори (понад 16 млн. відтінків). Можливе застосування стиснення за алгоритмом RLE.

1.4.4.4. PDF (Portable Document Format)



PDF запропоновано фірмою Adobe, як платформонезалежний формат для створення електронної документації, презентацій, передачі верстки та графіки через мережі.

Односторінкові файли PDF відмінної якості може створювати Photoshop. Багатосторінкові PDF можуть створювати програми Adobe Acrobat, PageMaker і програми пакету MS Office.

PDF спочатку проектувався як компактний формат для електронної документації. Тому, всі дані в ньому можуть стискатися, причому, до різного типу інформації застосовуються різні, найбільш прийнятні для них типи стиснення: JPEG, RLE, CCITT, ZIP. Програма Acrobat дозволяє розставляти гіперпосилання, заповнені поля, додавати у файл PDF відео чи звук, інші дії.

Файл PDF може бути оптимізованим. З нього видаляються елементи, що повторюються, встановлюється посторінковий порядок завантаження сторінок через Веб, з пріоритетом спочатку для тексту, потім для графіки. Якщо елементів, що повторюються, немає, файл, після оптимізації, як правило, дещо збільшується.

PDF найчастіше використовується для передачі по мережах графіки і зверстаного тексту в компактному виді. Він може зберігати всю інформацію для пристрою виведення, яка була в початковому файлі.

1.4.4.5. WMF (Windows Metafile)



Векторний формат WMF є рідним форматом Windows і використовує його графічну мову. Призначений для передачі векторних даних через буфер обміну (Clipboard). Розпізнається практично всіма програмами Windows, так чи інакше пов'язаними з векторною графікою. Користуватися форматом WMF варто лише у випадках передачі «чистих» векторів. WMF спотворює колір, не зберігає певні параметри, які привласнюються об'єктам в різних векторних редакторах, не містить растрові об'єкти, не розпізнається багатьма програмами на Macintosh.

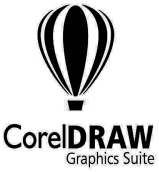
1.4.4.6. AI (Adobe Illustrator Document)



Adobe Illustrator – популярний графічний редактор від Adobe. Векторний формат Illustrator можна безпосередньо відкрити у Photoshop, його підтримують майже всі програми Macintosh і Windows так чи інакше пов'язані з векторною графікою і графікою взагалі.

Формат Illustrator є найкращим посередником при передачі векторів з однієї програми в іншу, з PC на Macintosh і навпаки. Втілені або пов'язані з документом растрові файли при обміні через формат Illustrator втрачаються.

1.4.4.7. CRD (CorelDraw Document)



Векторний формат, що має незаперечне лідерство на платформі PC. Багато програм на PC (FreeHand, Illustrator, PageMaker тощо) можуть імпортувати файли CorelDraw.

У файлах застосовується окрема компресія для векторів і растрів, можуть втілюватися шрифти, файли CorelDraw мають величезне робоче

поле 45х45 метрів (цей параметр є важливим для зовнішньої реклами), підтримується багатосторінковість.

1.5. Історія користувацького інтерфейсу

Біля витоків знайомого всім користувацького інтерфейсу в комп'ютерах і смартфонах лежать напрацювання Apple**,** яка змогла довести до розуму і впровадити в масові продукти цікаві, а іноді навіть геніальні знахідки інших компаній і людей. Давайте згадаємо, як все це було. Щодня мільйони людей активно використовують свої мобільні пристрої. Вони друкують, клацають, гортають, перетягують і змінюють розмір зображень на сенсорних дисплеях. Багато хто з нас пам'ятають, що так було далеко не завжди.

Спочатку була *клавіатура***.** Точніше, спершу з'явилися *перфокарти* і *електронно-променеві трубки***.** Вони допомогли в створенні перших гігантських комп'ютерів, які могли використовувати тільки спеціально навчені люди. Лише пізніше клавіатура стала таким собі посередником між людиною і програмою. Вона допомогла розібратися в управлінні інформацією, чітко ієрархічно організованою на жорстких дисках перших комп'ютерів. Використання *друкованого слова* для маніпуляцій і взаємодії з технікою – цілком розумний і логічний крок, адже слово і є способом ведення діалогу не тільки між людьми, але і між людьми і їх винаходами.

У підсумку за допомогою клавіатури користувачі друкували інструкції, які виконувалися програмами. Причому, деякі до цих пір працюють з ПК за тим же принципом, це розробники і програмісти, які мають справу з більш високими матеріями, ніж простий обиватель, для якого використання ПК зводиться до веб-серфінгу і ігор.

60-70-і поява миші, компанія Xerox та перший графічний інтерфейс

В 1960-і на світ з'явилася комп'ютерна миша (Рисунок 1.2.) завдяки її винахіднику зі Стенфорда, Д. Енгельбарту. Так зародилася нова модель взаємодії з ПК, а дизайнери перших користувацьких інтерфейсів отримали можливість створювати меню, якими було набагато простіше управляти користувачам. За допомогою мишки стало зручно буквально перетягувати по екрану ПК значки. Почалася епоха, в якій людині для управління комп'ютером не потрібно було бути експертом в даній сфері, адже сам ПК міг запропонувати своєму користувачеві різноманітні меню, повні тих чи інших опцій.



Рисунок 1.2 – Комп’ютерна миша 1960-і роки

У 70-і компанія Xerox використовувала цю ідею, створивши перший *графічний користувацький інтерфейс* в стилі *«що бачиш на екрані, то і отримаєш» (What You See Is What You Get).* Само собою, спочатку ця розробка не планувалася як комерційний продукт. Інженери в Xerox таким способом збиралися спростити підготовку документів до їх друку на принтерах, вироблюваних цією ж компанією.

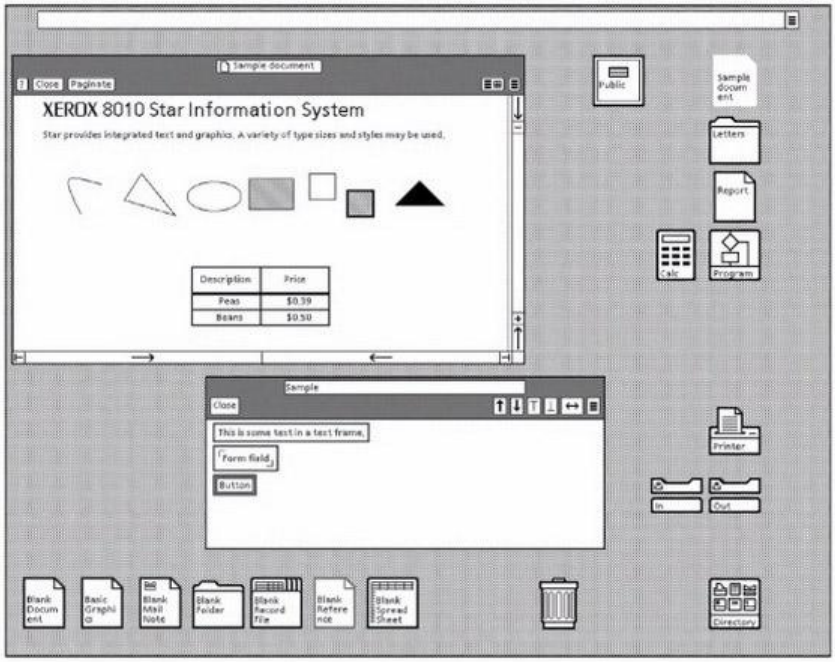


Рисунок 1.3 – Інтерфейс компанії Xerox

80-і. Macintosh та Windows

Незабаром після цього Apple *запозичили нову ідею у Xerox в обмін на певну кількість акцій своєї нової фірми***.** А в 1984 році світ побачив перший Macintosh – комп'ютер для обивателів, за допомогою якого люди могли легко виконувати якісь повсякденні завдання. Концепція персонального комп'ютера стала ще доступнішою широкому загалу.

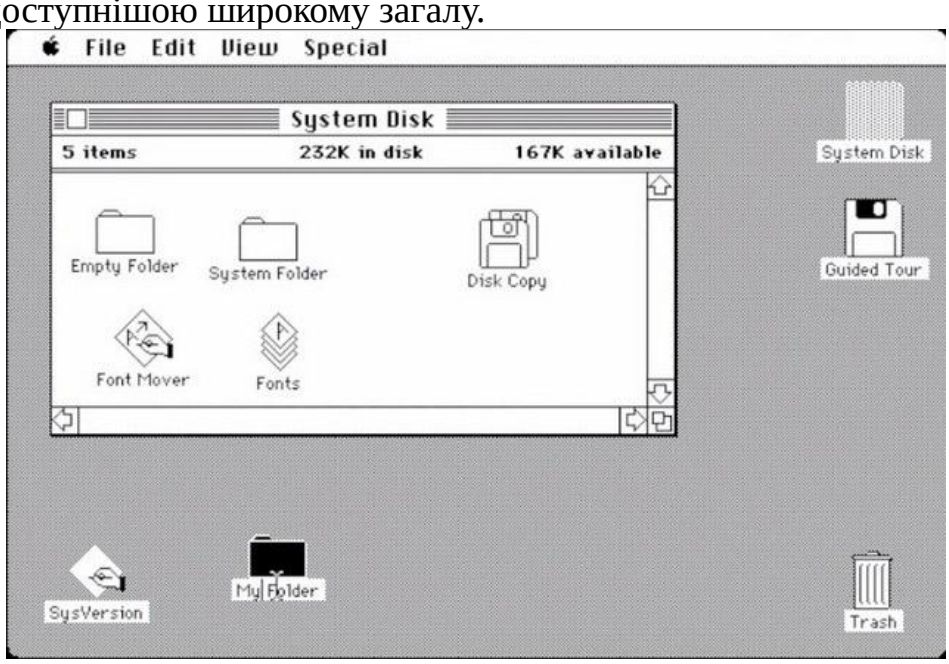


Рисунок 1.4 – Інтерфейс Macintosh

Ідея виявилася настільки успішною, що в дуже незабаром її використовували Microsoft, створивши власну платформу Windows. А поява Windows 95 без перебільшення створила революцію в сфері ПК.



Рисунок 1.5 – Інтерфейс Windows

Перша декада XXI століття – карамельний веб «2.0»

Розвиток комп'ютерних технологій сприяло їх спрощення і, відповідно, зростанню популярності. На початку 90-х персональні комп'ютери почали змінювати світ. Причому вдале поєднання досить непоганий апаратної платформи, зрозумілого ПО і доступу в Інтернет обіцяло практично всім бажаючим можливість познайомитися з таким, здавалося б, технологічно складним процесом управління ПК.

Протягом 90-х інтерфейс комп'ютерів виглядає в цілому однаково, будь то машини на Windows 95 або Windows 98 (на той момент набагато більш популярні, ніж ОС від Apple).



Рисунок 1.6 – Типова веб-сторінка 90-х

Виступаючи на конференції Future of Web Design в Нью-Йорку, відомий американський дизайнер Elliot Jay Stocks описав «те, що люди мають на увазі під візуальним дизайном в стилі веб 2.0», наступними ознаками:

• яскраві кольори;

• бейджі «спеціальна пропозиція», наліплені на сторінку;

• глянцевість, скляні, «карамель» в іконах і ілюстраціях;

• закруглені кути;

• градієнти;

• патерни на фон з діагональними лініями;

• відображення. Ці кліше наповнювали інтернет в основному з 2003 по 2010. Часто технології не дуже встигали за модою. Згадаймо, наприклад, верстку закруглених куточків, яка робилася за допомогою картинок. Однак, як видно з цього ж прикладу з куточками, мода підхльостувала технології.



Рисунок 1.7 – Дизайн у стилі веб 2.0

З плином часу з'явилися і інші пристрої крім комп'ютерів, покликані полегшити життя користувачів і зробити її цікавішою. Наприклад, *музичні плеєри.* Навряд чи кому-то варто нагадувати еволюцію цього пристрою від простого касетного плеєра до CD- та MP3-плеєра. У випадку з касетним і CD-плеєром управління ними реалізовувалося за допомогою звичайних апаратних кнопок. Потім трапився черговий прорив – світло побачили піринговi музичні сервіси типу Napster. Так користувачі отримали доступ буквально до будь-якої бажаної пісні. Тоді-то, на зорі ери цифрової музики і стали популярними MP3-плеєри. Однак саме iPod своєю появою ознаменував доступність і портативність музики в цифровому форматі буквально для всіх.

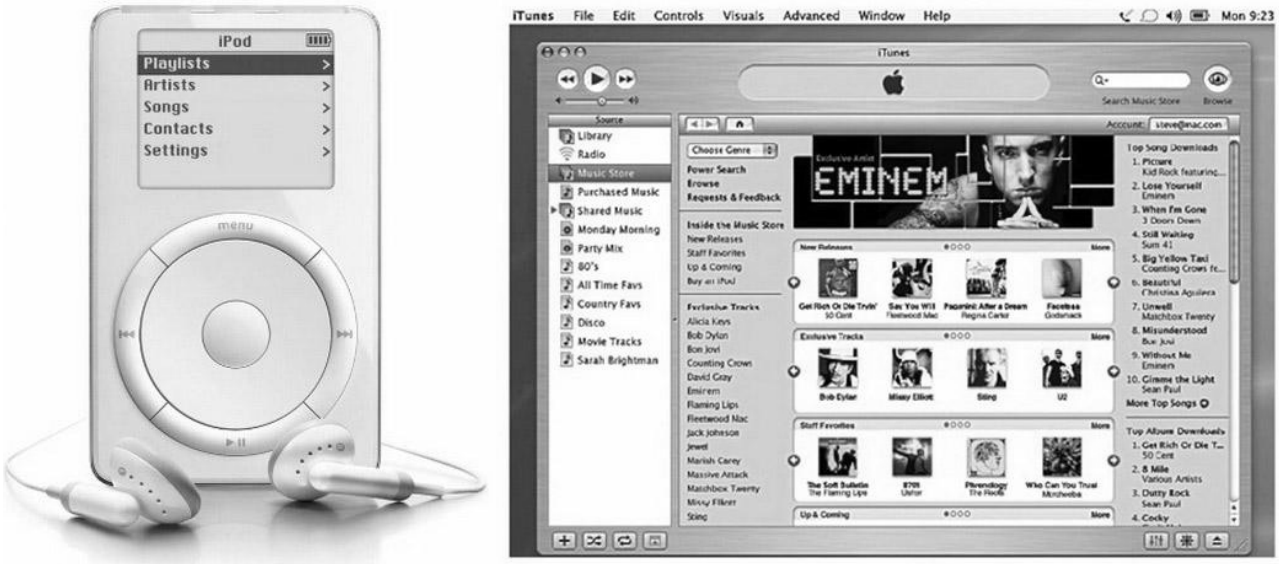


Рисунок 1.8 – Інтерфейс IPod та ITunes

І саме iPod став позиціонуватися як дійсно якісний музичний плеєр. А щоб підкреслити це і «підсадити» користувачів на новий гаджет, Apple забезпечила їх ще і дійсно унікальним ПО для управління та синхронізації їх музичних колекцій – iTunes.

Трохи пізніше виробник уклав угоди із звукозаписними лейблами і запустив магазин з продажу музики, iTunes Music Store. Виявилося, що це лише справа часу – перетворення iTunes в найпопулярніший віртуальний музичний магазин, а iPod – в улюблений багатьма і найпопулярніший в світі музичний плеєр.

Приблизно в цей же час почали з'являтися *стільникові телефони,* які змогли замінити пейджери як засіб віддаленого і мобільного спілкування між людьми. Перші їх моделі були скоріше схожі на прості стаціонарні телефони. З розвитком технологій стільникові пристрої почали оснащувати маленькими поки що ЖК-дисплеями, і незабаром їх власники отримали можливість відправляти один одному короткі текстові повідомлення через стільникову мережу. Щоб написати таке повідомлення, доводилося користуватися фізичними кнопками телефону. Звичайно, життя користувачам трохи спростила функція «Т9», якийсь прототип вже звичного нам предикативного введення.

Перші смартфони насправді зовсім не були телефонами, їх функцію спочатку виконували *КПК***.** Вони, звичайно, були більш функціональні, ніж перші стільникові, та й така-сяка технологія сенсорного введення в них уже була присутня. Хоча для цього доводилося використовувати стилус.

Само собою, Apple вирішила не відставати від конкурентів, і світ побачив Newton. До речі, саме для цього гаджета тоді і був створений термін «КПК».



Рисунок 1.9 – Інтерфейс перших КПК та Newton

До слова, в першу чергу після повернення в Apple Стів Джобс звернув виробництво Newton. Саме тоді було вирішено створювати виключно продукти, без яких користувачам не обійтися, в мінімалістському дизайні і з властивою Apple філософією. І як раз в ті дні до команди приєднався зовсім ще молодий дизайнер Джоні Айв, який допоміг у створенні iMac, MacBook і вже згаданого вище iPod.

Тоді ж Apple почала роботу і над Mac OS X, операційною системою для комп'ютерів Mac. Іконки в її інтерфейсі сам Стів називав «їстівними», адже майже всі вони нагадували льодяники, та й в цілому зовнішній вигляд системи значно відрізнявся від інтерфейсу Windows.

Прийшов час і такі розробки як комп'ютер, стільниковий телефон і MP3-плеєр стали цілком самодостатніми, такими, що відбулися продуктами. В Apple скористалися цим і об'єднали їх в одному пристрої, в першому iPhone. Він вже не був простим телефоном, а й до смартфона, яким ми знаємо його зараз, iPhone 2G було ще далеко.

Проте, смартфони того часу все ще йшли в комплекті зі стилусом і рідко обходилися без фізичних кнопок. На тлі цього iPhone виявився просто таки революційним продуктом в своїй сфері з його сенсорним дисплеєм з підтримкою мультітач. Інші телефони поки ще були оснащені пластиковими резистивним дисплеями, і їх користувачам, щоб реалізувати той чи інший жест, доводилося сильно натискати на екран. Ємнісний же дисплей реагує на електроімпульси в людському тілі, і в сильних натисканнях просто немає необхідності.

Це перший в історії момент, коли *людина змогла управляти комп'ютером, вже кишеньковим, за допомогою прямих маніпуляцій.* Якщо користувачеві потрібно перемістити дані, він може зробити це, буквально перетягнувши пальцями документ або зображення з одного місця екрану в інше. Ні клавіатура, ні миша для цього вже не потрібні.

2011 – скевоморфізм

З початку другого десятиліття XXI століття на зміну «карамельний» тренду прийшов істинно натуралістичний дизайн.

Скевоморфізм – це використання елементів дизайну дуже схожих візуально на їх аналоги в реальному світі. У промисловому дизайні, наприклад,

широко відома історія про те, що двигун в перших автомобілях помістили попереду, так як це здавалося інтуїтивним: адже коні ж спереду воза.

Отже, еволюція плеєра, телефону, комп'ютера відбувалася в першу чергу заради спрощення використання їх вузьким колом фахівців, відмінно розбираються в техніці. Тобто, вона реалізовувалася за рахунок технічного прогресу, але з орієнтацією на спеціалізацію. Коли ж справа дійшла до масового використання, від техніки все ще чекали якихось особливостей, знайомих користувачеві за звичними для них предметів побуту. Так що в основу дизайну продуктів для широких мас лягло таке явище як **скевоморфізм.**



Рисунок 1.10 – Приклад скеморфізму у дизайні додатка для нотаток

Суть його полягає в тому, що якийсь створюваний продукт виглядає або працює (або і те, і інше) так, як інший, звичний для користувачів продукт. До скевоморфізму розробники користувацького інтерфейсу зверталися і раніше, але саме Apple і саме в iPhone зробила цю практику вкрай популярною.

Детально промальована тривимірна іконка «кошик для сміття» або «шкіряний» фон додатки-календаря у iOS того часу – характерні приклади скевоморфізм.

Риси скевоморфізму в інтернеті:

• тривимірні елементи;

• приглушена, натуралістична колірна схема;

• елементи, виглядають як тканина, шкіра або інший фізичний матеріал, з реалістичними швами і текстурою;

• «тиснені» букви, «занавісочки», стрічки, тощо.

Одночасно при цьому тренду в моду входили і більш фундаментальні речі, пов'язані з розвитком браузерних технологій, полегшенням праці верстальника і, скажімо прямо, формуванням смаку у веб-дизайнерів, які, нарешті, остаточно стали окремою професією. Це в моді і до цього дня:

• цікава, різноманітна типографіка;

• дотримання законів друкованого дизайну в оформленні тексту;

• великі фотографії на фон.



Рисунок 1.11 – Скеморфізм у елементах веб 2.0. у дизайні сайту

Треба сказати, що весь цей час законодавцем мод був по суті один великий бренд з інноваційним іміджем, вгадайте хто? Ну звичайно, Apple. Елементи веб 2.0, а потім і скевоморфізм вперше з'явилися на їх сайтах і додатках.

Екрани з підтримкою мультитач стали гігантським кроком у забезпеченні зручності використання смартфонів. Але сама по собі технологія виробництва дисплеїв все ще залишалася на одному рівні з виробництвом екранів для ПК і ноутбуків, а адже смартфон і лептоп – далеко не одне й те саме. Тексти і картинки на таких дисплеях виглядали і сприймалися краще, якщо були прикрашені додатковими відволікаючими деталями.



Рисунок 1.12 – Скеморфізм у додатку iBooks

Інша справа – *Retina-дисплеї* з високою щільністю пікселів, завдяки чому текст і зображення видно набагато чіткіше – ніби на сторінці глянцевого журналу. Коли наше око перестає розрізняти окремі пікселі на екрані пристрою, дизайнерам інтерфейсу немає потреби використовувати прийоми і прикраси, щоб змусити його виглядати привабливо. Тут досить добре пропрацював контенту.

Хоча тут же стає ясно, що і дизайн апаратної складової також не менш важливий для досвіду використання. Візьмемо як приклад додаток «Калькулятор» в iPhone. Спочатку його інтерфейс виглядав так, як виглядає звичайний фізичний калькулятор – адже народ вже знав, як користуватися

останнім. І хоча на даний момент дизайн програми дещо змінився, став більш плоским, в ньому ще можна помітити схожість зі звичним калькулятором.



Рисунок 1.13 – Скеморфізм у дизайні додатку «Калькулятор»

В 2010-2011 роках на сцену стилю вийшли відразу два конкурента.

2010 – Windows Metro (Microsoft design language і Modern UI)

У 2010 році Microsoft представив громадськості свій новий дизайн, радикально відмінний від минулого. Основний упор в стайлгайді був зроблений на «плоскі» елементи і типографіку, на відміну від колишніх, «іконочні» інтерфейсів. Зміни підносили як «по-справжньому цифровий дизайн». Так як і Apple теж в цей час почав вводити елементи мінімалізму, це швидко стало трендом, що захлеснули інтернет.



Рисунок 1. 14 – Windows Metro стиль

Не в останню чергу величезна популярність «плоского» дизайну пов'язана з технічними потребами в епоху стартапів і мобільних пристроїв. Як ніколи важливо стало швидко робити сайти і додатки, які б виглядали адекватно і на робочому столі і на екрані смартфона. Зі скевоморфними елементами важко зробити дійсно responsive дизайн, а з плоскими – будь ласка, і виглядає стильно.

2011 – Twitter Bootstrap

Twitter Bootstrap був представлений як фреймворк для полегшення життя програмістам фронт-енду, а по суті став провідником нового стилю.

Легкість і візуальна акуратність фреймворку зіграли важливу роль в його стрімке поширення. Але основною причиною було, звичайно ж, рішення «під ключ» цілого комплексу характерних завдань фронт-енд розробки. Те, що можна відразу отримати і низькоякісний продукт під всі платформи, і хороший дизайн, привернуло тисячі розробників. А в інтернеті почалися розмови про те, що всі сайти стали на вигляд однакові. Однак існує думка, що ні тоді, ні зараз Bootstrap не був законодавцем мод, а скоріше їх прямим відображенням. У Bootstrap 1.0 застосовувалися градієнти, більше було тіней і іншого ненав'язливого 3d, характерного для епохи занепаду скевоморфізм. Ось, наприклад, як змінилося модальне вікно с кнопками:

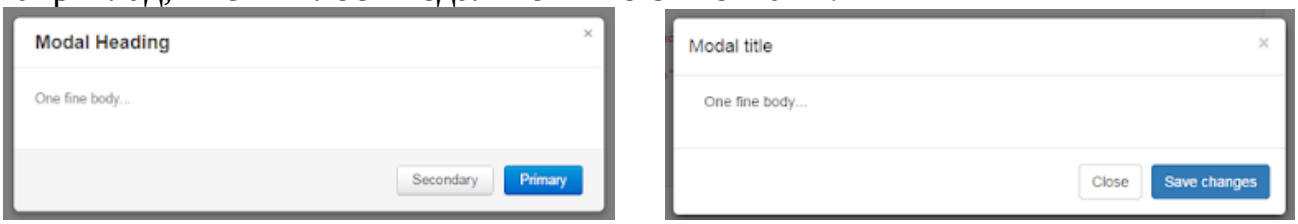


Рисунок 1.15 – Twitter Bootstrap

2012 – Flat Design

Скевоморфізм став перетворюватися в ознаку, а в повну силу увійшов тренд «плоский дизайн» з наступними візуальними ознаками:

- мінімалізм;
- двомірність, без тіней і глибини;
- на першому плані контент;
- повернення яскравих кольорів.

Навіть кнопка, для якої був традиційний affordance – опукла форма, схожа на форму фізичних кнопок, стала плоским прямокутником. Продовжують активно розвиватися тренди, які стали модними ще в епоху скевоморфізму: цікава типографіка і орієнтація на друкований дизайн та великі фотографії і навіть відео на тлі сайтів.

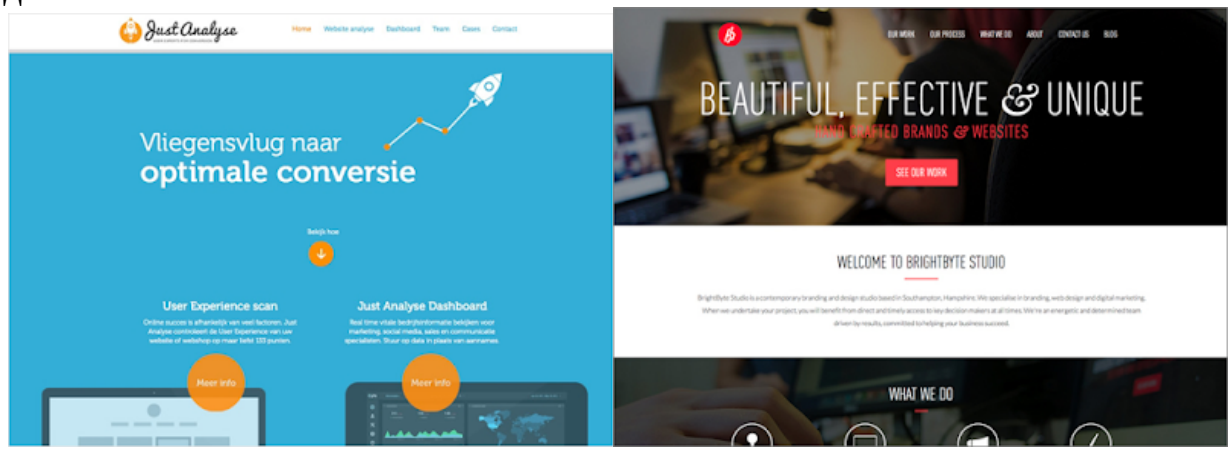


Рисунок 1.16 – Приклад Flat Design

2013 – боротьба трендів

Загострення пристрастей в розпал боротьби плоского і дизайну, що не здається скевоморфним, досяг апогею. Це стало однією з найбільш обговорюваних тем в інтернеті не тільки серед дизайнерів. Аргентинське агентство In tacto навіть випустило на цю тему гру. Тим часом, прогрес не

стоїть на місці і в 2014 році вийшли два «візуальних мови», які, можливо, вирішать результат боротьби.

2014 року – Google Material Design

Докладний стайлгайд від Google вражає не тільки новизною дизайну, скільки рівнем піару і масштабом донесення до аудиторії. Google дійсно вдалося стати законодавцем мод в дизайні додатків, для яких призначався стайлгайд. Однак дуже швидко з'явилося і кілька фреймворків Material Design для вебу, велика частина яких зроблена не в Google (наприклад, ось Material Design в Angular.js).

Як пояснив дизайнер John Weley на конференції Google I / O 2014 року, на противагу підходу Microsoft, які оголошували свій Metro "authentically digital", Google хотіли повернутися до реального, «цифрові» світу, знайти натуралістично, в чомусь навіть скевоморфну метафору для інтерфейсу. Однак метафора, яку вони вибрали, все ж виявилася самим плоским, що було в фізичному світі – це папір.

Відповідно, майже всі в інтерфейсі виглядає і поводиться, як папір. Екрани – це аркуші паперу, а на них – чорнило. У дизайн повернулися тіні, що відкидаються, коли один лист розташовується над іншим. Знову з'явилася піднесена кнопка, але тепер вона не натискається (не "опускається») при кліці, а видає ефект розпливаються чорнила. Папір, проте, має і ряд нереалістичних властивостей, наприклад, у неї немає текстури, а листи можуть безшовно з'єднуватися і розділятися.

Стиль Google Material Design в цілому підтримує тенденції мінімалізму, використання яскравих кольорів і цікавою типографіки.

З нового ж він привів в моду наступне:

• повернення тіней і глибини в дизайн;

• іконки використовуються багато, вони в цілому плоскі, але з тінями, складками і «глибиною»;

• анімації в стилі метафори «паперових листів».

Анімація вперше настільки адекватно підтримує загальну стильову ідею і практично виступає на перший план стайлгайду. Майже в кожному розділі присутні приклади з цікавою і обґрунтованою анімацією.

IBM Design Language

В кінці минулого року компанія IBM виклала у відкритий доступ свій новий design language. Як сказано в блозі IBM, надихає ідеєю для нового дизайну була аналогія з людськими мовами. Мова – це те, що дозволяє людям спілкуватися, розуміти один одного і разом вирішувати завдання. Подібним чином і візуальну мову – design language – покликаний допомогти дизайнерам і розробникам продуктів «заговорити однією мовою», і між собою, і з користувачами.



Рисунок 1.17 – IBM Design Language

На відміну від Google Material Design не дає рекомендацій за розмірами відступів, а задає філософський хороший тон в дизайні. Наприклад, є кілька абзаців про золотий переріз і варіантах сітки, але немає ніяких шаблонів і обмежень «ось так буде в стилі IBM, а ось так зовсім немає», як це зроблено в Material Design. Небажання дарувати ледачим готові шаблони автори стайлгайду пояснюють прагненням дати свободу новим ідеям. На тлі розмов про те, що всі сайти і додатки стали однаковими, звучить заманливо.

Іноді конкретні цифри і гайдлайни все ж можна розшукати. Наприклад, в розділі Іконографіка є наступна рекомендація: "Не скруглять куточки, тоді іконки матимуть більш строгий «науковий» вигляд".

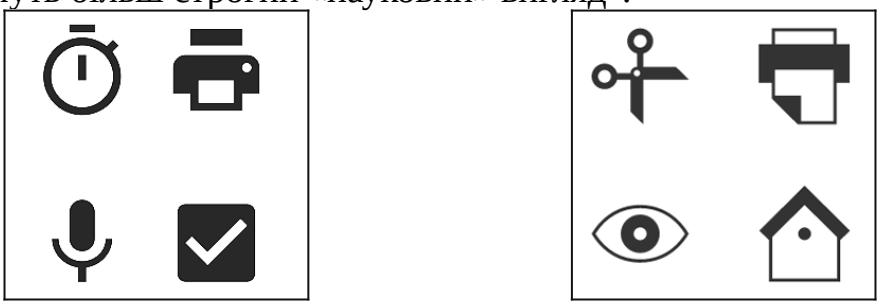


Рисунок 1.18 – Порівняння іконок Google Material Design та IBM Design Language

У IBM Design Language велику увагу приділено акценту на користувача і контент, а також знову-таки анімації. За основу концепції руху взяті безсмертні принципи діснеєвської анімації. Коротко, але досить ємно розказано, в чому секрет привабливості та стильності деяких прийомів. Даються цікаві приклади, наприклад, про те, як блок warning при появі рухається важко, з уповільненням, своєю важкістю привертає увагу і створює враження важливості.

Ось у чому *суть дизайну користувацького інтерфейсу:* використання речей, які вже знайомі і зрозумілі користувачам, для того, щоб навчити їх поводитися з новими речами.

Зараз дизайн переживає справжній ренесанс. Користувачі тепер прекрасно розуміють важливість зручності у використанні і взаємодії зі своїми гаджетами. Роль дизайну більше не зводиться до якогось оформлення вже готової технології. Це – окрема галузь знань, в якій мріють працювати багато найталановитіші люди.

У цій сфері гармонійно співіснують дизайн ПО і дизайн апаратної частини, досягнувши, нарешті, симбіозу. Найкраще ПО зливається з прекрасно пропрацювали апаратними компонентами, і вони разом створюють сам пристрій. Побачивши його, користувач вже не замислюється над тим, як воно працює. В думках його відразу формуються картинки того, як його можна використовувати.

В Apple на даний момент обидва види дизайну контролюються однією людиною – Джоні Айвом. Та й як можна розділити ці дві складові, якщо вони тепер нерозривно пов'язані і є єдиним цілим.

У поп-культурі, до речі, також можна відстежити мрії людини про розвиток користувацького інтерфейсу. Технології роботи дисплеїв удосконалюються практично на очах. Так що інтерфейс на зразок того, що задіяний в шоломі Залізної Людини, не здається більш чимось нереальним або фантастичним. Це, швидше, питання часу. Напевно, майбутнє зараз за непомітними технологіями, які ніяк не проявляють себе, поки в їх використанні немає необхідності. Варто визнати, що для розвитку технологій людську уяву вкрай важливо, але в плані інтерфейсу йому вже точно належить роль першої скрипки. Згадаймо хоча б прообрази iPad і Siri в серіалі *«Зоряний шлях: Наступне покоління»*. Ми мріємо, потім ми втілюємо свої мрії в життя і удосконалюємо кінцевий продукт так, що він виглядає зійшов зі сторінок твору в стилі наукової фантастики минулих років.

Розвиток технологій відбувається з неймовірною швидкістю. Але все революційні відкриття і нові гаджети з'являються не тому, що користувач мріє ще сильніше інтегруватися з технологіями. Наша мета – забезпечити собі більш якісне спілкування, зв'язок один з одним.

1.6. Етапи роботи над створенням UI дизайну

Створення користувацького інтерфейсу або веб-сайту – це трудомісткий і тривалий процес, який відбувається в кілька етапів, в міру проходження яких ідеї замовника перетворюються в реальний функціонуючий сайт.

Розробку користувацького інтерфейсу або веб-сайту можна порівняти з будівництвом будинку, де мають послідовно бути виконані визначені етапи: від проектної документації і закладки фундаменту до внутрішньої і зовнішньої обробки приміщення.

Етапи розробки проекту, як правило, виконуються послідовно, тому, вкрай важливо дотримуватися черговості етапів і розуміти, що будь-які несподівані і неузгоджені заздалегідь зміни чи правки можуть значно вплинути на ефективність роботи.

Роботу над кожним проектом слід проводити в строгій відповідності з приведеними нижче етапами робіт з розробки користувацького інтерфейсу або веб-сайту.

1 етап. Визначення цілей створення користувацького інтерфейсу або веб-сайту та проведення досліджень за темою

Це найважливіший етап в створенні користувацького інтерфейсу або веб-сайту, оскільки не можна досягти мети, якщо її немає або неправильно

визначено. Від цілей буде залежати весь подальший процес створення користувацького інтерфейсу або веб-сайту, кожен його етап. Правильно поставлена мета – це вже половина успіху.

Перш, ніж приступити до розробки, необхідно проаналізувати тему, вивчити сайти потенційних конкурентів. Надалі це допоможе в створенні власної концепції.

2 етап. Складання технічного завдання

Розробка користувацького інтерфейсу або веб-сайту є доволі складним процесом, який вимагає детальної інформації про майбутній проект. Для досягнення повного взаєморозуміння із замовником необхідно правильно складене технічне завдання, в якому будуть відображені всі поставлені завдання. У процесі створення такого проекту проводиться поетапне узгодження виконуваних робіт, що дозволяє зробити сайт, який відповідає всім вимогам клієнта.

На цьому етапі розробник спільно з замовником визначаються з умовами щодо реалізації проекту.

- Складання технічного завдання (ТЗ) з попередніми вимогами до користувацького інтерфейсу або веб-сайту.
- Визначення рамок бюджету, оцінювання фінансових можливостей клієнта.
- Формування команди розробників.
- Узгодження часу на реалізацію проекту.

ТЗ, як мінімум, має містити такі пункти:

- Тип користувацького інтерфейсу або веб-сайту (лендінг, візитка, каталог, корпоративний, магазин тощо).
- Структура користувацького інтерфейсу або веб-сайту (які сторінки повинні бути на сайті).
- Функціонал користувацького інтерфейсу або веб-сайту (пошук, каталог, стрічка новин).
- Стиль дизайну (витриманий, молодіжний, в темних тонах), колірна гама, потреба у створенні нового бренду (логотип, слогани, фірмові кольори та шрифти).

3 етап. Технічні аспекти проектування користувацького інтерфейсу

На цьому етапі визначається архітектура користувацького інтерфейсу або веб-сайту, файлова та логічна структура сторінок.

- *Архітектура веб-сайту* є основною організацією клієнтської та серверної частин користувацького інтерфейсу або веб-сайту, їх взаємозв’язок та зв’язок з кінцевим користувачем. Створення архітектури користувацького інтерфейсу або веб-сайту є першочерговим етапом в його проектуванні.
- *Файлова структура* – це чітко оформлена система організації різних файлів. Продумана і зручна файлова структура допомагає розробнику оптимізувати роботу, а також буде зрозумілою для інших фахівців, що працюють над проектом.

- *Логічна структура користувацького інтерфейсу або веб-сайту* – це внутрішній устрій користувацького інтерфейсу або веб-сайту, його «кістяк», розташування сторінок, розділів, підрозділів, додаткових матеріалів. І першочерговим завданням розробників є створення стрункого порядку з хаотичного скупчення інформації.

4 етап – розробка макета дизайну користувацького інтерфейсу або веб-сайту

Цей етап поділяється на кілька підетапів:

- Генерація ідей дизайну. Набір ідей надається замовнику у вигляді ескізів з текстовими поясненнями.
- Розробка попереднього макета дизайну головної сторінки.
- Виправлення зауважень замовника (якщо є), доробка макета до завершеного вигляду.
- Розробка внутрішніх сторінок за аналогічним алгоритмом.

У макеті повинні бути промальовані всі блоки, які будуть на сайті. Якщо до моменту розробки дизайну текстові та графічні матеріали ще не готові, можна використовувати довільний текст, але не можна залишати в макеті порожні місця.

5 етап – HTML-CSS верстка

Залежно від цілей і завдань користувацького інтерфейсу або веб-сайту, верстка повинна задовольняти деяким вимогам. Зазвичай, ці вимоги такі:

- Кросбраузерність – сторінки повинні однаково відображатися в популярних браузерах (Mozilla Firefox, Google Chrome, Opera, Internet Explorer, Safari і т.д.).
- Адаптивність під різні розміри екрану та мобільні пристрої.
- Гнучкість верстки – можливість легко додавати/видаляти блоки інформації на сторінки.
- Швидкість обробки коду браузером.
- Валідність – відповідність до різних стандартів.
- Семантична коректність – логічне і правильне використання елементів HTML.

6 етап – програмування і система керування контентом (CMS)

Це чисто технічний етап, на якому реалізується весь функціонал користувацького інтерфейсу або веб-сайту. Вимоги до цього етапу визначаються в технічному завданні.

7 етап – заповнення контентом (інформацією)

На цьому етапі відбувається наповнення користувацького інтерфейсу або веб-сайту якісним, професійним контентом. Всі матеріали користувацького інтерфейсу або веб-сайту, будь то тексти чи графіка повинні вписуватися в загальну концепцію користувацького інтерфейсу або веб-сайту, відповідати його цілям і задачам.

8 етап – тестування і виправлення помилок

Тестування працездатності користувацького інтерфейсу або веб-сайту може проводити як розробник, так і замовник. Найкращий варіант – це спільне тестування.

9 етап – публікація в Інтернеті

На цьому етапі для користувацького інтерфейсу або веб-сайту обирається доменна адреса, хостингова площадка, розгортання адміністративної частини або системи управління контентом, фізичне перенесення інформації. Здійснюються тестові перевірки в реальних умовах.

10 етап – просування користувацького інтерфейсу або веб-сайту і реклама в Інтернеті

Після остаточного завершення робіт над сайтом потрібно залучати на нього відвідувачів. Для просування та популяризації користувацького інтерфейсу або веб-сайту можна скористатися контекстною або банерною рекламою, пошуковою оптимізацією, просуванням в соціальних мережах та іншими методами.

1.7. Визначення цілей та проведення досліджень

1.7.1. Основне завдання

Процес створення користувацького інтерфейсу або веб-сайту починається з усвідомлення того, що сайт є інформаційним майданчиком та необхідним інструментом для популяризації справи замовника (бізнес, наукові дослідження, літературні твори, клуб по інтересах тощо).

Спонукальними мотивами для розробки користувацького інтерфейсу або веб-сайту є або його відсутність, або нездатність існуючого користувацького інтерфейсу або веб-сайту вирішувати актуальні для замовника завдання. Причини непридатності користувацького інтерфейсу або веб-сайту можуть бути різними, наприклад, незручна структура користувацького інтерфейсу або веб-сайту, старий («немодний») дизайн, відсутність системи управління сайтом або інші мотиви. Таким чином, усвідомлюючи проблему, клієнт має чітко сформувати цілі і завдання проекту, а також критерії оцінки досягнення цієї мети. Правильна постановка завдання і чітке розуміння кінцевої мети – запорука передбачуваного ефективного результату.

Цілі – це очікування, перспективи, які можуть справдитися внаслідок певних дій. Всі цілі повинні бути: вимірними, досяжними, обмеженими в часі. Чітку відповідь на питання чи досягнуто поставлені цілі, можна отримати по закінченні певного часу, відведеного на їх досягнення. Серед цілей, що висувають клієнти можна виділити наступні:

- Підвищення популярності ресурсу або компанії, підтримання позитивного іміджу.
- Збільшення числа відвідувачів, клієнтів і партнерів замовника.
- Контакт з цільовою аудиторією, надання для неї необхідної та актуальної інформації, оперативна взаємодія з відвідувачами, клієнтами, партнерами.
- Просування ідей (творів, товарів, послуг) за допомогою Інтернету.

Тематика користувацького інтерфейсу або веб-сайту

Для визначення тематики потрібно врахувати кількість людей, що цікавляться даною темою та кількість подібних сайтів-конкурентів. Чим більше сайтів з подібною темою висвічується в пошукових системах, тим менше відвідувачів буде на сайті клієнта, тим складніше буде залучати відвідувачів. Кількість сайтів з певних тем сягає мільйонів і всі прагнуть бачити свій сайт на перших позиціях. Для того, щоб успішно конкурувати, слід розробити свіжі ідеї, які спроможні забезпечити високу відвідуваність.

Аналіз сайтів конкурентів

Завданням будь-якого користувацького інтерфейсу або веб-сайту є втримання відвідувача і змусити його зробити дії, заради яких створювався сайт. Щоб по-справжньому зацікавити відвідувача, сайт повинен бути не просто "одним з багатьох", а повинен стати одним з найкращих у своїй ніші.

Саме з цієї позиції і варто приступати до аналізу сайтів-конкурентів.

Аналіз дозволить оцінити плюси і мінуси проектів подібної тематики:

- Зовнішній вигляд користувацького інтерфейсу або веб-сайту, зручність його дизайну і структури, зрозуміла логіка навігації.
- Зміст користувацького інтерфейсу або веб-сайту, повнота і доцільність поданих матеріалів, наочний матеріал.
- Працездатність наявних сервісів (пошук, зворотній зв’язок).

Аналізуючи сайт конкурента, можна витягнути корисну інформацію про рівень виконання та його рекламну стратегію. Також, дослідження конкурентних ресурсів може принести багато бізнес-ідей для власної практики.

Аналіз рекламної стратегії конкурентів – непросте завдання, яке вимагає певних витрат часу і значного досвіду в області SEO-оптимізації.

1.7.2. Визначення цільової аудиторії

Важливим для створення успішного користувацького інтерфейсу або веб-сайту є аналіз контингенту людей, на чию відвідуваність він найбільше розраховує. Необхідно виявити основні потреби цільової аудиторії, після чого впорядкувати їх за ступенем важливості.

Аудиторію любого користувацького інтерфейсу або веб-сайту в загальному можна поділити:

- *Цільова аудиторія користувацького інтерфейсу або веб-сайту* – це відвідувачі, на яких розраховано проект і яким повинен бути цікавий наданий матеріал.
- *Побічна аудиторія користувацького інтерфейсу або веб-сайту* – це різновид цільової аудиторії, але не настільки явно виражений. Це люди, які заходять на сайт за «побічними» запитами, що певним чином збіглися з тематикою сторінки. Вони заходять на нього, і якщо це нецільовий клієнт, то він піде майже відразу, якщо цільовий, то він залишиться. З цього випливає, що побічна аудиторія користувацького інтерфейсу або веб-сайту – теж цільова і варта уваги.
- *Випадкова аудиторія користувацького інтерфейсу або веб-сайту* – це люди, які заходять за випадковими нетематичними запитами, які чомусь

потрапили в топ пошукової системи. Така аудиторія здебільшого інтересу не представляє, зазвичай, ці люди просто покидають сайт.

В залежності від того, які саме матеріали містить сайт, можна спрогнозувати загальні риси представників цільової аудиторії і скласти «портрет клієнта».

Портрет клієнта – це впорядкована інформація про пересічного представника цільової аудиторії: вік, стать, соціальний рівень, життєві інтереси. Згідно з цим, слід визначати колірну гаму, стиль користувацького інтерфейсу або веб-сайту, складати відповідні тексти, подачу матеріалів і рівень спілкування.

1.7.3. Складання технічного завдання

Технічне завдання (ТЗ) — це письмовий опис сайту, який потребує розробки.

Розробка технічного завдання включає в себе: визначення основних характеристик сайту, його ідеї, напрямку, дизайну, технічних параметрів. ТЗ розробляється і узгоджується із замовником, що дозволяє чітко сформувати всі критерії створення сайту до початку роботи над його візуальною та технічною частиною. Якісно розроблене ТЗ, дозволяє значно зменшити час розробки сайту.

В розділі «Вступ» вказують найменування, коротку характеристику області застосування програми або програмного продукту і об’єкту, в якому будуть використовувати програму.

1. В розділі «Мета (ціль) розробки» вказують документ, на основі якого ведеться розробка (наказ по університету, завдання на лаб.роботу і т.д.); організація, яка затвердила документ та дата його затвердження; найменування теми розробки.

2. В розділі «Призначення розробки» вказують функціональне і експлуатаційне призначення програми

3. Розділ «Вимоги до програми» повинен містити наступні підрозділи:

3.1. «Вимоги до функціональних характеристик» – склад функцій, які буде виконувати програма, організація вхідних і вихідних даних (синтаксис і семантика вхідних даних, формати вихідних повідомлень і відповідні їм ситуації), тимчасові характеристики. В цьому розділі повинно бути описана поведінка системи з точки зору співвідношення входу і виходу без конкретизації внутрішньої структури і реакції програми на передбачені дані на вході.

3.2. «Вимоги до надійності» – контроль вхідних і вихідних даних, наслідки можливих відмов, час відновлення, захист від несанкціонованого доступу.

3.3. «Умови експлуатації» – характеристики операційного середовища, вид обслуговування, кількість і кваліфікація персоналу, число користувачів, а також допустимі параметри навколишнього середовища.

3.4. «Вимоги до складу і параметрів технічних засобів» – конфігурація системи, основні характеристики необхідних пристроїв.

3.5. «Вимоги до інформаційної і програмної сумісності» – вимоги до методів рішення, мовам програмування, програмним засобам, які використовуються системою, протоколам обміну, до СУБД і операційним системам.

4. В розділі «Вимоги до програмної документації» – текст програми, опис програми, керівництво користувачу, керівництво програміста.

5. В розділі «Стадії і етапи розробки» – перелік стадій, розбивки на етапи, вміст, терміни розробки (дні, тижні і т.д.).

6. В розділі «Порядок контролю і приймання» – види випробувань, вимоги до приймання робіт, способи перевірки найважливіших характеристик. Ціль випробувань – встановлення ступеню відповідності готового продукту і характеристикам ТЗ.

Основними пунктами розробки технічного завдання створення сайту є:

- Визначення характеру сайту (рекламний, корпоративний, сайт електронної комерції...)
- Детальне вивчення стилю, змісту
- Вибір назви сайту, його основних рекламних слоганів (за потребою)
- Розробка загального плану головної сторінки
- Розробка схеми меню та карти сайту
- Розробка текстової
- частини сайту (контенту)
- Розробка загальних планів ключових сторінок сайту
- Визначення параметрів технологічної бази
- Вибір доменного імені сайту
- Вибір CMS сайту (системи управління контентом)
- Вибір хостингу

Технічний опис для "Розробка технічного за вдання": Включає в себе уточнення вимог до сайту, їх аналіз, структуризацію та документування, і представлення у вигляді зрозумілого для подальшого програмування опису. Результат може складатися з таких елементів:

- схеми основних процесів, які будуть автоматизовані;
- перелік об'єктів системи, їх структура та взаємозв'язки;
- рекомендована структура баз даних;
- перелік та опис функціональних можливостей системи;
- інтерфейс користувача на рівні схематичного зображення або вимоги
- до інтерфейсу;
- вимоги до безпеки, юзабіліті (зручність).

1.7.4. Визначення типу веб-сайту

На сьогодні типи сайтів в основному збігаються до кількох типів: лендінг, візитка, каталог, корпоративний сайт, Інтернет-магазин.

1.7.4.1. Лендінг

Лендінг (landing page – цільова, посадкова сторінка) – це веб-сторінка, що скерована на спонукання користувача вчинити будь-яку дію: підписатися на розсилку, купити продукт, завантажити електронні матеріали. Лендінг є

потужним інструментом просування продукту на ринку і підвищення ефективності реклами. Перехід до лендінг-сторінки, зазвичай, здійснюється через email-розсилки, соціальні мережі, рекламні посилання в пошуковій видачі. Такі сторінки створюють для вузької аудиторії, їх головним завданням є утримання уваги користувача на конкретному повідомленні та його подальша конвертація в клієнта. У порівнянні зі звичайними сайтами лендінг має певні особливості:

- Зосередженість користувача на одну дію (мету).
- Максимальна стимуляція користувача зробити цільову дію.
- Збільшення числа потенційних споживачів.
- Посилення ефективності рекламної кампанії (зростання конверсії до 10-15%).
- Відносно невисока вартість розробки.

1.7.4.2. Сайт-візитка

Сайт-візитка це віртуальна візитна картка (зазвичай, до 4-5 сторінок) із загальною інформацією, яка надає ознайомчі відомості про компанію або приватну особу. Особливостями є лаконічність, інформативність, індивідуальність.

Основні розділи користувацького інтерфейсу або веб-сайту-візитки:

- Загальні відомості: короткий опис, основні завдання і цілі компанії.
- Перелік послуг і товарів, що надаються компанією з можливим коротким описом.
- Загальні відомості для покупців: прайси, перелік товарів чи послуг.
- Контактні дані: адреси, телефон, електронна пошта, кнопки на соціальні мережі.

Головними цілями користувацького інтерфейсу або веб-сайту-візитки є ознайомлення потенційних клієнтів з компанією, надання інформації про послуги і товари, а також контактні дані для зв’язку з компанією. На сьогодні візитки розробляються разом з бізнес-сторінкою в соціальних мережах і вони мають між собою перехресні посилання.

1.7.4.3. Сайт-каталог

Сайт-каталог це сайт, що містить повну інформацію про компанію та електронний каталог її послуг чи продукції. Відвідувачі користувацького інтерфейсу або веб-сайту можуть ознайомитися з прайсами, описом товарів і послуг, переглянути фотографії та звернутися в компанію для здійснення покупки. Такий сайт може збільшувати кількість продажів за рахунок доступу до повної інформації про продукцію та послуги тих споживачів, які територіально віддалені від компанії-виробника.

1.7.4.4. Корпоративний сайт

Корпоративний сайт це розширена версія користувацького інтерфейсу або веб-сайту-візитки. Як правило, цей сайт може складатися з необмеженої кількості сторінок, і власник може завжди розширювати і доповнювати контент. Містить загальні інформаційні блоки: про компанію, її послуги, продукція,

контактна інформація. Залежно від напряму компанії присутні інші блоки, що потребують програмування:

- Новини, статті, блоги.
- Пошук по користувацького інтерфейсу або веб-сайту, онлайн-консультації, обговорення питань у чаті, інтерактивна карта.
- Реєстрація, форми замовлення послуг або розсилання новин.

Корпоративний сайт є динамічним і потребує програмування або використання системи керування контентом. Для підтримування працездатності користувацького інтерфейсу або веб-сайту потрібна спеціально навчена людина, що має навички у користуванні адміністративною частиною.

1.7.4.5. Інтернет-магазин

більше ніж просто сайт, це складна платформа, яка потребує професійного командного підходу. Вимоги до сучасного Інтернет-магазину:

- *Зручний для користувача.* Він має зрозумілу структуру і категоризацію, працює швидко і безпомилково.
- *Кросплатформовий*. Сайт однаково відображається та працює в різних браузерах, на різних операційних системах, на смартфонах і планшетах.
- *Наповнений якісним контентом***.** Кожен товар повинен супроводжуватися докладним описом із зазначенням ціни, якісними і достовірними зображеннями. Умови оплати, доставки, повернення – вся комерційна інформація повинна бути доступна для користувача без ускладнень.

Важливою частиною Інтернет-магазину є наявність функціональних модулів, що забезпечують взаємодію з покупцями.

- Зручні способи перегляду товару, наявність фільтрації та пошуку за визначеними параметрами.
- Підбір аналогів або супутніх товарів.
- Статистика по популярних товарах, інформація щодо нових надходжень, актуальні прайси.
- Опції «Додати до кошику» або «Швидке замовлення».
- Зрозумілий інтерфейс персонального кабінету, історія замовлень, порівняння характеристик вибраних товарів.
- Можливість додавання відгуків і рекомендацій, онлайн-спілкування з консультантами.
- Зручні способи електронної оплати та доставки.

Технічне обслуговування Інтернет-магазину вимагає цілої команди фахівців різних спеціальностей: контент-менеджерів, програмістів, дизайнерів. Для успішного розвитку бізнесу важливим є застосування Інтернет реклама та маркетингу, відпрацьовані схеми оплати, доставки та гарантії у разі неякісного товару, call-центр з різних питань, юридична підтримка.

Стиль дизайну

Для вибору стилю дизайну, варто враховувати особливості проекту, для якого розробляється сайт:

- Тематика майбутнього проекту.

- Цільова аудиторія – вікова група, основна соціальна та статева приналежність, освіта, коло інтересів.
- Очікувані емоції від дизайну користувацького інтерфейсу або веб-сайту.

Кожен проект є особливим, і до його розробки потрібно підходити індивідуально. Для створення успішного проекту потрібен нестандартний підхід, з помірною часткою креативу. Сучасні основні стилі веб-дизайну надають можливість реалізовувати творчі ідеї, створювати і використовувати нові свіжі стилі. Тренди веб-моди швидкоплинні і тому, перед розробкою користувацького інтерфейсу або веб-сайту варто ознайомитися з останніми новинками, стильними і колірними тенденціями, поданням інформації, сучасні засоби зв’язку тощо.

1.8. Введення в проектування UX

Все більше компаній, як серед стартапів, так і серед великих корпорацій, наймають до штату проектувальника інтерфейсів. Поточне десятиліття в пресі називають «декадою дизайну». Справа в тому, що кількість цифрових продуктів і послуг неухильно зростає, а разом з цим зростає і необхідність в удосконаленні юзабіліті. Офіційне визначення користувацького досвіду це:

«Сприйняття, що виникають у людини при використанні і / або передбачуваному використанні продукту, системи або при отриманні послуги» (ISO 9241-210: 2010, підрозділ 2.15).

Якщо мова йде про проектування UX, то визначення включає в себе прийоми і способи, за допомогою яких дизайнер може поліпшити юзабіліті. Безпосередньо сам «дизайн» спрямований на те, щоб підвищити корисність, зручність використання і ефективність продукту або послуги в призначеному для користувача досвід.

Візуальний дизайн – це те, як продукт виглядає, а юзабіліті – те, як він сприймається. Також юзабіліті включає в себе всі аспекти взаємодії користувача з компанією – від обслуговування клієнтів до якості продукції. Багато UX-дизайнерів використовують терміни «продукт» і «послуга», як взаємозамінні.

Вікіпедія визначає проектування UX як: «Процес підвищення рівня задоволеності користувача від взаємодії з продуктом за рахунок підвищення його зручності, простоти використання і доступності».

Кожен раз, коли ви взаємодієте з продуктом, програмним забезпеченням або будь-яким об'єктом, ви сприймаєте його як користувач. А значить, ключове завдання проектування UX полягає в тому, щоб зробити цей досвід настільки успішним, наскільки це можливо.

Кінцевою метою проектування UX є не тільки задоволення запитів користувача. Процес проектування спрямований на те, щоб допомогти продукту досягти поставлених бізнес-цілей і зробити так, щоб ці цілі збіглися з цілями користувача. З чого випливає наступне питання:

1.8.1. Для чого необхідне проектування інтерфейсів

Проектування інтерфейсів дає дві головні переваги:

1. Воно позитивно впливає на сприйняття продукту користувачем.
2. Успішний досвід користувачів підвищує рівень прийняття продукту.

Основні цілі проектування інтерфейсів:

1. Зрозуміти цілі потенційних клієнтів і контекст використання продукту.
2. Виходячи з цілей клієнтів і контексту використання, розробити продукт, послугу або додаток в рамках обмежень бізнесу і технологій.

Перше, про що думає юзабіліті-спеціаліст – це те, як об'єднати цілі користувача з цілями бізнесу. Наприклад, якщо метою користувача є покупка продукту, то компанія повинна зробити цю покупку корисною, зручною і приємною.

- **Корисною**: Вам потрібно задовольнити потребу клієнта, тобто вирішити його проблему.
- **Зручною**: Високий рівень юзабіліті повинен бути очевидний, щоб клієнти розуміли сферу застосування вашого продукту або використання послуги.
- **Приємною**: Не зайвим буде, якщо покупка вашого продукту доставить клієнту радість.

Якщо метою користувача є отримання інформації, то компанія повинна надати достовірні дані, вселяючи тим самим довіру покупця, а персонал повинен чемно відповідати на поставлені по телефону питання. Задоволений клієнт, який отримав всю необхідну йому інформацію, повернеться, а розчарований – ні.

1. Якщо мета клієнта досягнута (тобто він легко і швидко знайшов те, що шукав), то у нього залишається приємне враження від вашого продукту.
2. Якщо у клієнта залишилося хороше враження, то він з більшою ймовірністю зробить покупку або порекомендує продукт своїм друзям.
3. Коли клієнт повертається на сайт, купує ваш продукт або розповідає про нього друзям, обсяг продажів і рівень конверсії ростуть.

1.8.2. Коротка історія UX

За останні роки термін «Проектування UX» став близький за значенням до технологій і програмного забезпечення, але так було не завжди. Спочатку під «призначеним для користувача досвідом» розумілося просто переживання, що виникають у людини при використанні системи.

Термін «Проектування UX» був введений у вжиток у 1995 році Доном Норманом (Don Norman), який в той час займав пост віце-президента групи розробки технологій в Apple. Він сказав: «Я винайшов цей термін оскільки вважав, що «інтерфейс для людини» (human Interface) і «юзабіліті» були занадто вузькими термінами. Я хотів задіяти всі аспекти користувацького досвіду взаємодії з системою, зокрема з промисловим дизайном продукту, його графіку, інтерфейс і фізичний контакт».

Крім того, Норман написав книгу «Дизайн звичних речей», в якій вперше зручність і функціональність були поставлені вище естетики. Його робота досі вельми популярна в дизайнерських колах. Інтерес до проектування UX виріс, а сам термін став узагальнюючим для ряду різних областей, таких, як Дослідження користувацької аудиторії, Інформаційна архітектура, Розробка юзабіліті, Дизайн сервісу тощо.

1.8.3. Еволюція методології UX процесу

UX процес здається заплутаним, навіть для більшості дизайнерів

На початку був Дональд Норман, і він сформулював ідею дизайну користувацького інтерфейсу: «Я придумав цей термін, тому що вважав, що Human Interface і юзабіліті – занадто вузькі поняття. Я хотів охопити всі аспекти досвіду людини в системі, включаючи графіку промислового дизайну, інтерфейс, фізична взаємодія і керівництво. З тих пір цей термін широко поширився, настільки, що він починає втрачати сенс.»- Дональд Норман

До 2016 року Дон Норман рішуче заявив, що цей термін був абсолютно неправильно зрозумілий. Тепер він стверджує, що він *дизайнер для людей*, але він не згоден з назвою.

Складно знайти кращий привід, щоб почати розбиратися в дивному і заплутаному світі сучасного UX дизайну: *навіть творець нашої галузі не знає, як себе назвати.*

Сучасний UX вимагає розуміння історії дизайну і розробки з 1990-х років. Вони нерозривно пов'язані між собою.

1.8.4. Класичний дизайн користувацького досвіду

У своїй чистій формі UX Design заснований на каскадній моделі розробки.

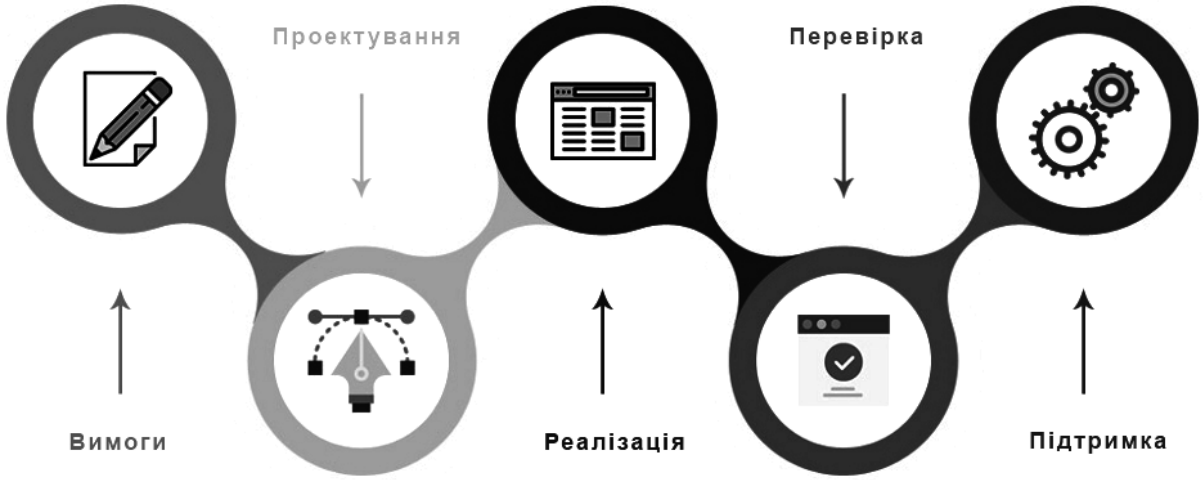


Рисунок 1.19 – Каскадна модель розробки

Продуктова команда, яка використовує в своїй роботі каскадну модель розробки, дізнається все, що можливо, до створення навіть самого найпростішого прототипу. Дослідження можуть тривати місяці або навіть роки; і їх результати диктують роботу дизайн команди. Фіксація вимог до запуску дизайну і фіксація дизайну до початку розробки. Все залишиться незмінним до версії 2.0. Ось як працює каскадна модель.

Класичний процес UX, який зазвичай викладається студентам виглядає так:

1. Проведіть дослідження, щоб виявити проблеми
2. Категоризує проблеми, які ви розкриваєте
3. Створіть персони і карти подорожей

4. Генеруйте ідеї
5. Створіть і протестуйте прототип
6. Передайте фінальну версію прототипу розробникам
7. запустіть продукт
8. Поверніться до першого пункту, ґрунтуючись на відгуках користувачів

Це по суті каскадна модель. Класичні елементи UX (Jesse James Garrett) також дотримуються цієї моделі, забезпечуючи виконання знизу-вгору на підставі добре сформульованих вимог.



Рисунок 1.20 – Модель побудови UI з урахуванням UX

Проблема в тому, що *класичний UX принципово несумісний з методом гнучкої розробки (також відомим як agile).*

1.8.5 Agile

Довгий час інновації в Кремнієвій долині були обумовлені Законом Мура, в якому говориться, що кількість транзисторів в щільній інтегральній схемі подвоюється приблизно раз в два роки. Накладіть сюди каскадну модель розробки, і ви помітите, що вона прекрасно вписується в 24-місячний період. Цикли бізнесу, дизайну і розробки працювали як механізм швейцарського годинника, прекрасно налаштований до випуску нового чіпсета Intel. *Потім в один день Sony, Toshiba і IBM (Альянс STI) вирішили, що закон Мура був занадто повільним.* STI створили перший cell chip, в якому було зібрано 8 мікропроцесорних ядер на одній пластині (на його основі працювала Playstation

2). Багатоядерна архітектура змінила все. Закон Мура не перестав працювати, його «хакнули». Що *дійсно перестало працювати – каскадна модель розробки.*

Швидкість і гнучкість замінили точність і передбачуваність в якості конкурентних переваг. Компанії стали масово переходити до іншої добре розробленою, але недостатньо використовуваної методології розробки — Agile.

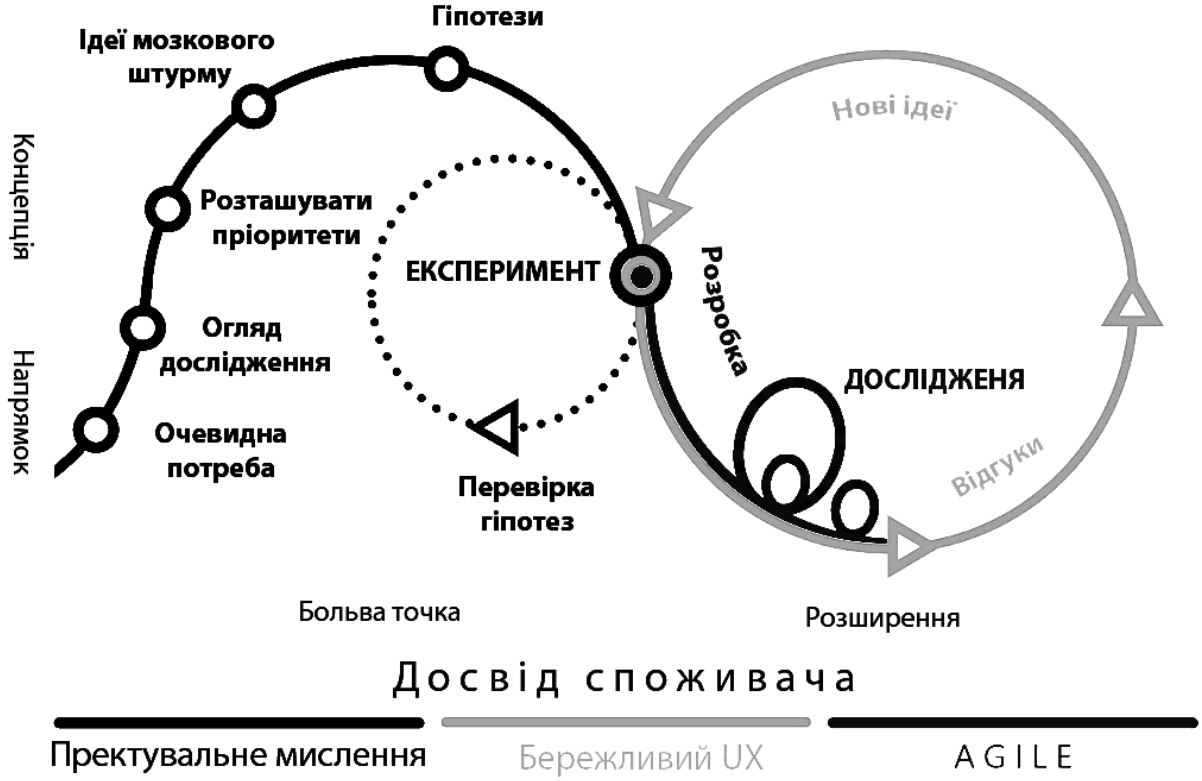


Рисунок 1.21 – Agile

Agile-метод фокусується на проведенні швидкої ітерації. Він випускає новий набір функцій кожні 2-4 тижні, розвиваючи продукт поступово, а не відразу. Він заснований на гіпотезі, експериментах, швидкому релізі і вимірі в реальному часі. У методології гнучкої розробки немає стадії редагування, немає досконалості.

Скрупульозний підхід каскадної моделі «сім разів відміряй, один відріж», вийшов з моди швидше, ніж StarTAC в кімнаті, повної iPhone.

Хоча, була одна маленька проблема: *Класичний UX працював на каскадній моделі.* З випуском функціоналу один або два рази на місяць не було часу чекати, поки UX дизайнери виконають свій процес. У Agile-номенклатурі UX став стопором, і це було погано.

Зіткнувшись з нерухомим блоком, більшість команд просто відмовилися від UX. Вони найняли молодих графічних дизайнерів, які могли випускати ресурси за двотижневі ітерації. Ці дизайнери не були справжніми UX дизайнерами в класичному сенсі, але вони знали досить про дизайн, що орієнтований на користувача, щоб уникнути жахливих помилок.

Це було нормально для команд розробки продуктів. Деталі, як вони вважали, будуть доопрацьовуватися при ітерації.

Що означає Agile для такого дизайнера? UI / UX дизайнери були в центрі бурі, коли Apple довела, що бізнес, ведений дизайном, може стати найціннішим брендом в світі. Вони несли прапор UX, коли UX не міг сам заточити олівець протягом двотижневого спринту, не кажучи вже про розробку функції. Однак це коштувало дорого.

Упередженість UI / UX щодо цифрового та графічного дизайну сильно спотворила сприйняття бізнес-світом того, що робить дизайнер. До сих пір дуже багато підприємств як і раніше вважають UX візуальної дисципліною. Оскільки Дон Норман намагався пояснити, це неправильно. Ситуація не почала поліпшуватися до 2013 року.

1.8.6. Економний UX (Lean UX)

Lean UX – це рішення від Джеффа Готельфа, що дає відповідь на проблему практики UX в гнучкому середовищі. У своєму оригінальному шедеврі Джефф перевернув всю галузь UX з ніг на голову. У його книзі представлений ряд стратегій і дій по вирівнюванню, які дозволяють UX працювати всередині agile та швидко оновлювати проекти на основі відгуків користувачів. У той час, як класичний UX заснований на вимогах, Lean UX заснований на результатах.

Однак, Lean UX не є ідеальним. У той час, як UX тепер міг гармонізувати з ритмом методології гнучкої розробки, економний UX давав збій, якщо продукт був розпливчасто визначено.

Дизайнери виявилися під величезним тиском, щоб заповнити беклог спринту (масив роботи, яка повинна бути виконана), перш ніж вони дійсно зрозуміли, що вони створюють. В результаті багато цикли розробки витрачалися на функції, які ніколи не потрапляли в кінцевий продукт. Серед менеджменту проектів пара Lean UX / Agile була відома великою кількістю витрат і переробок.

У зрілому продукті є багато відгуків користувачів, які чудово керують ітеративним циклом, чим забезпечують Lean UX. Саме тому Lean UX є галузевим стандартом практично для будь-якої групи продуктів.

Проблема витрат і переробки в цих умовах була настільки серйозною, що вона часто заважала прийняттю Lean UX. Це являло собою справжню загадку для стартапів, які хотіли забезпечити сильний UX-компонент, що не повертаючись назад до каскадної моделі розробки.

Джейк Кнапп і Google Ventures, як грім серед ясного неба, вирішили проблему з дизайн спринтом. Темні часи UX нарешті закінчилися.

1.8.7. Дизайн спринт від Google Ventures

Якби ви поглянули на дизайн-спринт і сказали: «Почекайте, це просто дуже швидкий класичний UX», ви б не були далеко від істини. Різниця (і геніальність) полягає в тому, що це *класичний UX з неймовірно низькою точністю.* Це різниця між художнім шедевром і ескізом на серветці, але він працює.

Мета дизайнерського спринту – взяти всі існуючі дослідження проблеми, розкрити їх суть, а потім генерувати ідеї з шаленою швидкістю. Ідеї потім

оцінюються з точки зору дизайну, орієнтованого на людину. Команда голосує за відповідні ідеї. Потім дизайнери створюють прототип з низькою точністю (найчастіше за один день), який *навряд чи* достатньо хороший для тестування потенційних користувачів. Результати тесту, якщо вони позитивні, формують цілі для дизайнерів, щоб вони досягли високої точності; таким чином підживлюючи цикл Lean UX / Agile.

1.8.8. Dual Track дизайн

Dual – track agile часто відноситься до розколу між дослідженнями і проектуванням в середовищі Lean UX. Отже, тепер ми підходимо до сучасної методології UX. В dual track дизайні є два шляхи*:*

1. Дизайн мислення / дослідження дизайн спринту
2. Ітеративне експериментування The Lean UX / Agile

Виробнича команда зазвичай працює на другому шляху, *якщо беклог погано сформульований.* Якщо це станеться, проблема буде оцінена і команда проведе дизайн спринт. Проблему вирішено.

Але звідки береться ця оцінка, якщо виробнича команда зайнята практикою Lean UX? Найефективніша відповідь – виділити на неї спеціальну команду. Дослідницька група вивчає результати більш всебічно у фоновому режимі. Оскільки ця дослідницька група не прив'язана до темпу спринту Agile-методу, у них все ще є можливість виділити 3 місяці, щоб прийти до висновку.

Кожні кілька місяців у виробничих команд з'являється свіжа думка, що поповнює беклог новими ідеями для експериментів. В одній з компаній, з якими я працюю зараз, є ціла команда обробки і аналізу даних, яка передає дані декільком творчим підрозділам. Нові ідеї виникають рідко, так як у дослідників довший цикл доставки, але, коли справа стосується напрацювань на майбутнє – це чудово.

Іноді ідея досить велика, щоб стати приводом дизайн спринту, який значно розвиває бачення продукту. Однак більшу частину часу ідеї просто стимулюють експерименти Lean UX.

Тому dual track дизайн найкраще працює, коли спеціалізована дослідницька група працює поза сферою активної групи продуктів. Ви, напевно, здогадалися, що це дороге підприємство. Ви не помилилися.

1. Спочатку був Дон Норман і він сформулював поняття UX.

2. Інші новатори, такі як Джессі Джеймс Гарретт, розкрили ідеї Дон Нормана і народився класичний процес UX.

3. Класичний UX добре працював зі стандартним ритмом каскадної моделі розробки, і всі були щасливі. UX як і раніше був нишевой практикою.

4. Інновації ігнорували Закон Мура, і каскадна модель розробки більше не працювала.

5. Тому класичний UX був несумісний із сучасною методологією гнучкою розробки.

6. UI / UX дизайнери перейшли на дизайн, орієнтований на користувача. Він має вирішальне значення для зростання UX в області дизайну, але обмежений за охопленням. Його упередження до цифрового і графічного дизайну сильно

спотворило сприйняття бізнес-світом ролі UX. Це залишається проблемою і сьогодні.

7. Lean UX був задуманий Джеффом Готельфом в 2013 році і подарував нам дизайн інтерфейсу в гармонії з розробкою продукту. Раптово дослідники, дизайнери взаємодії, дизайнери інформації і всілякі спеціалізовані UX фахівці почали користуватися попитом.

8. На жаль, Lean UX був неефективний, коли план продукту не було чітко визначено, що призвело до значних витрат і переробок.

9. Google Ventures придумали дизайн спринт, який дозволив командам швидко визначити і протестувати прототипи з низькою точністю. Це початок цикл Lean UX для нових груп продуктів і ефективно усунуло проблему витрат і переробок.

10. Dual Track дизайн поєднав модель дизайн спринту від Google Ventures з методологією Lean UX Джеффа Готельфа. Це вже розвивається дисципліна з початку 2018 року.

Отже, що ж таке UX дизайнер в світі dual – track?

Дизайнер UX – це лідер, генератор креативних ідей, постачальник контексту. UX дизайн насправді не має будь-яких результатів, крім цінності.

1.9. Юзабіліті в дизайні

Юзабіліті – це простота використання і освоєння об'єкта, створеного людиною. Це поняття безпосередньо пов'язане з причинами, за якими займаються проектуванням інтерфейсів – оскільки юзабіліті позитивно впливає на призначений для користувача досвід. Чим простіше освоїти і використовувати продукт, тим кращий досвід отримує користувач. Але продукт повинен бути не тільки зручним у використанні – він повинен вирішувати проблему клієнта. І в більшості випадків робота юзабіліті-спеціаліста полягає в тому, щоб з'ясувати, що потрібно користувачам.

Необхідні для дизайнерів юзабіліті якості

Емпатія – здатність розуміти, чому люди чинять той чи інший спосіб. Для того, щоб займатися проектуванням юзабіліті, потрібно вміти ставити себе на місце інших людей. Ймовірно, це найбільш важлива якість для роботи в цій області, і вам необхідно навчитися розуміти вчинки інших людей. Всупереч розхожій думці, цією здатністю не кожен володіє від природи.

Допитливість – бажання дізнатися, чому люди чинять ту чи іншу дію.

Ясність вираження – здатність пояснити складні поняття в доступній формі для тих, хто не знайомий або мало знайомий з даною галуззю.

Проектувальник прототипів в лабораторії RED компанії Manulife Алі Рушдан Тарік (Ali Rushdan Tariq) пояснює так:

«Хороші юзабіліті-спеціалісти перш за все думають про людей. Вони переживають, якщо дізнаються, що їхні клієнти відчувають незручності при використанні продукту, і прагнуть виправити це. Більш того, хороші юзабіліті-спеціалісти не забувають про те, що на шляху до поліпшення життя людей вони повинні також переслідувати чітко позначені бізнес-цілі».

Як одного разу сказав Стів Джобс: «Дизайн – це не те, як продукт виглядає і сприймається. Дизайн – це те, як він працює».

Коли мова йде про дизайн продукту, цю істину, яку так точно сформулював Стів Джобс, часто забувають. Будь дізнається продукцію компанії Apple по її витонченому і унікальному виду. Дизайн айфонів і макбуків настільки вдалий, що технічні компанії всього світу намагаються його скопіювати.

Однак продукція Apple отримала міжнародне визнання аж ніяк не через своїх естетичних властивостей. І, хоча дизайн цих пристроїв є легко впізнаваним і функціональним, саме для користувача досвід і юзабіліті цих продуктів відмінне Apple від конкурентів і по сьогодні роблять бренд популярним. В даний час великі і невеликі компанії по всьому світу повторюють успіх Apple, зосереджуючи свої зусилля на юзабіліті своєї продукції.

Безумовно, естетичні властивості продукту підсилюють його привабливість. Однак, займаючись проектуванням UX, ви швидко прийдете до висновку, що якщо продукт не працює належним чином, то його зовнішній вигляд мало хвилює користувача. При цьому не варто забувати, що кінцевий успіх продукту залежить не тільки він вашого дизайну, але і від його втілення розробниками, а також від управління проектом. Це ще одна ключова навичка, необхідний для юзабілістів – вміння працювати в команді. Детальніше про це ми поговоримо в останньому розділі.

Проектування позитивного користувацького досвіду включає в себе ретельне планування циклу взаємодії продукту з клієнтом і надання йому допомоги в інтуїтивному пошуку. У будь-якому випадку, клієнти будуть взаємодіяти з продуктом, ґрунтуючись на своєму попередньому досвіді з іншими продуктами. Ваше завдання як проектувальника інтерфейсів – підлаштувати продукт / послугу під уже звична поведінка клієнта.

Дизайн вашого продукту має на увазі функціональність і юзабіліті, а не кольору і картинки (їх пізніше підбирає візуальний дизайнер). Коли за результатами проведеного дослідження користувацької аудиторії ви визначили очікування, цілі і бажання своїх клієнтів, саме про функціональність і юзабіліті продукту або сайту ви повинні думати в першу чергу.

Не забувайте, що будь-яка дія клієнта повинна бути корисним, зручним і приємним. Якщо ви не вирішуєте проблему користувача, він не подивиться ні на кольори, ні на картинки.

Ден Сеффер (Dan Saffer), фахівець з дизайну взаємодії і письменник, поділився іншим радою. Він сказав:

«Ніколи не забувайте чому: чому ви займаєтеся дизайном цього продукту, чому люди будуть ним користуватися, чому ви прийняли те чи інше рішення в процесі проектування. Запишіть ці причини на папері. Розкажіть про них будь-кому, хто буде слухати. Внесіть їх в свої макети та презентації. Це «чому» має задавати напрямок всьому процесу, оскільки саме воно визначають значення, історію і тему (оформлення) продукту».

1.10. Процес проектування UX та дослідження

Проектування UX – це діяльність, спрямована в першу чергу на людей, а значить, ключове вміння проектувальника UX – це здатність розуміти потреби і поведінку користувачів сайту, додатка або продукту. Він повинен прагнути до того, щоб ці потреби і бажання відповідали бізнес-цілям компанії, на яку він працює. Юзабіліті-спеціаліст завжди намагається зробити так, щоб кожне скоєне дію було корисним, зручним і приємним для користувача.

Коли ви придумуєте ідею для майбутнього продукту, ви не сумніваєтеся в її успіху. Ви уявляєте його своїй команді, вони говорять вам те, що ви хочете почути, – і ви думаєте, що вже виграли. Але у нас погані новини – результат може не виправдати Ваших очікувань.

Давайте розберемося, Ви провели великі дослідження своєї цільової аудиторії, і їм сподобалася ваша ідея, концепція має шанс «вистрілити». Але врешті-решт, саме кінцевому користувачеві повинно бути зрозуміло, як працює ваш продукт і як він виглядає, а не вашій команді і приятелям. Коли рішення не лежить на поверхні, почніть питати у користувачів, чого вони хочуть і що їм потрібно, тоді проекти будуть приречені на успіх.

Згідно зі звітом UX і User Research від 2017 року, 22% опитаних компаній заявили, що в 2016 році їх тести на зручність використання стали набагато більш частими в порівнянні з 2015 роком. Крім того, в доповіді показано, що існує чітка тенденція отримувати і аналізувати інформацію про користувачів якомога раніше, часто навіть до створення концепції продукту.

Всі ці статистичні дані говорять про основну мету дослідження користувацького досвіду – відповідей на питання: «наскільки хороший інтерфейс?» І «що дійсно потрібно користувачеві?».

Якщо ви серйозно ставитеся до UX, то знаєте його яку важливу роль він відіграє. Так, дослідження UX дійсно працює. Фактично, це впливає не тільки на окремі продукти, а й на цілі підприємства, засновані на їх відносинах з клієнтами. Формула для успішно поставленої послуги полягає в розумінні користувачів і їхніх потреб, а дослідження UX – це саме той інструмент, який допомагає отримати ці знання.

Але що таке UX-дослідження і як його проводити?

UX-дослідження **–** це безпосереднє спостереження за тим, що роблять користувачі.

Думки – це не докази.

Цей аналіз дозволяє розмістити клієнтів в центрі процесу розробки і враховувати їх потреби під час фактичного проектування. В результаті можна поліпшити свій продукт на ранніх стадіях і уникнути дорогих правок після запуску. Щоб зрозуміти логіку досліджень, необхідно усвідомити, що ця концепція набагато більш глобальна, ніж ви думали. UX виходить далеко за рамки веб-дизайну і розробки.

Дані користувача поведінки використовуються для планування міської навігації, вибору місць для вуличних знаків і створення туалетів. Дональд Норман в своїй книзі «Дизайн повсякденних речей» згадує випадок, коли він

використовував зворотний зв'язок для вибору місць сміттєвих кошиків в дослідницькій лабораторії. Це допомогло людям уникнути втрати важливих заміток і призвело до більш акуратною організації особистих речей.

Такі приклади показують, що співпраця з кінцевими користувачами використовувалося протягом століть, але вивчатися і застосовуватися з опорою на знання стало зовсім недавно. Дослідження UX йде паралельно з багатьма дослідницькими методами. Всі вони можуть бути класифіковані за двома основними характеристиками: якісним і кількісним.

Якісні – спрямовані на те, щоб визначити, чому користувачі поводяться саме так, а не інакше.

Кількісні – визначають, скільки користувачів роблять що-небудь певним чином. Одним словом, останній метод допомагає вимірювати різні ситуації чисельно.



Рисунок 1.22 – Якісні та кількісні дослідження

Як ви могли помітити, UX-дослідження представляє собою досить широку концепцію з великою кількістю застосувань, але з однією основною метою – з'ясувати, що краще для кінцевого користувача. Для досягнення цієї мети дослідники зазвичай покладаються на 3 ключових підходу, які визначають весь процес вивчення користувацького досвіду:

- *Спостереження*. У більшості випадків ваші клієнти не знають, чого вони хочуть або що їм потрібно. Ось чому важливо спостерігати і враховувати поведінку і реакцію користувача, якщо ви хочете отримати корисні дані, а не засипати питаннями.
- *Розуміння*. Збір інформації – це навіть не половина справи. Також важливо отримати максимальну користь від розуміння користувачів.
- *Аналіз і рішення.* Придумати відповідний список сценаріїв, діаграм, статистики, графіків тощо Ці 3 поняття – це свого роду 3 кита UX, це і є дослідження. Більш того, різні методи можуть збільшити результати цих операцій. Щоб дізнатися, який з них використовувати, вам потрібно з'ясувати.

Інформаційна архітектура

«У роботі дуже важливо дотримуватися загальної «мови» і єдиної структури – це необхідно при прийнятті серйозних рішень. Якщо люди у вашій організації використовують різні слова для опису одних і тих же речей, можуть виникнути проблеми. Невирішені семантичні і таксономічні суперечки часто призводять до необґрунтованого збільшення обсягу роботи і плутанини», – вважає Еббі Коверт (Abby Covert), президент організації IAinstitute і автор книги« Як навести порядок в будь-якому бардаку».

Якщо користувач не в змозі впоратися з величезним потоком інформації, або якщо він заплутався в навігації сайту, то його призначений для користувача досвід при взаємодії з продуктом, брендом або послугою буде негативним. Щоб уникнути подібних ситуацій, команда юзабіліті спуціалістів проектує так звану інформаційну архітектуру.

Інформаційна архітектура покликана структурувати і упорядкувати зміст сайту, а також розбити його на категорії, щоб користувачі з легкістю могли знайти те, що їм потрібно для досягнення мети. Через інформаційну архітектуру юзабіліті спеціалістів не тільки з'ясовує, як кожен елемент змісту вписується в сайт, але і як всі ці елементи поєднуються один з одним в цій структурі.

Переваги інформаційної архітектури:

- Підвищення самостійності клієнта і, як наслідок, більше задоволених клієнтів
- ефективна навігація
- Зниження витрат на технічну підтримку клієнтів
- Зниження кількості відмов від вашого продукту або послуги

Карткове сортування

Сформувати ієрархію контенту вашого сайту можна кількома способами. Один з найбільш поширених – карткове сортування. Під час сеансу карткового сортування користувачі розподіляють різні елементи сайту по декількох категоріях. Після цього їм потрібно придумати назву для кожної категорії, яке, на їхню думку, точно опише її зміст. При цьому можна використовувати справжні картки, аркуші паперу або один з кількох онлайн-сервісів по картковій сортуванні. Найкраще використовувати каркасні прототипи, щоб встановити зв'язок між візуальним дизайном сайту і його інформаційної архітектурою.

Прототипування як спосіб передпроектного дослідження

Це метод дослідження дозволяє зосередитися на класифікації і систематизації. З ним ви можете дізнатися, як структурувати свій сайт, назвати розділи меню і продумати контент вашого блогу. Створюючи, таким чином, звичне середовище для ваших користувачів.

Процес «каркасного моделювання» допомагає нам виявити різні способи подання або відображення інформації, а також дозволяє розподілити елементи змісту в порядку їх важливості для користувачів, відповідно до їх очікуваннями і цілями. «Каркаси» дозволяють юзабіліті спеціалістів визначити головний функціонал сторінки і підготувати команду проектувальників до розробки прототипів.

Крім того, wireframes – це швидкий, дешевий і простий метод. В процесі проектування інтерфейсів приступайте до створення «каркасів» якомога раніше і не зупиняйтеся на одному варіанті. «Каркаси» – це відмінний спосіб донести ідею до вашої команди чи до розробників.

Якщо ви не знаєте, з чого почати процес проектування, почніть з утримання. Ви здивуєтеся, скільки ідей у вас з'явиться, як тільки ви почнете писати текст. Коли ви закінчите моделювання «каркаса», перед вами з'явиться

візуальне уявлення того, як може виглядати ваш сайт, з урахуванням результатів проведеного дослідження користувацької аудиторії. Ваші «каркаси» відобразять розподіл змісту, зображень, кнопок і інших інтерактивних елементів на сторінці. Крім того, з їх допомогою ви зможете простежити перехід користувачів з однієї сторінки вашого сайту на іншу на шляху до їх кінцевої мети.

Прототип – це концепція вашого майбутнього сайту або продукту, яка дозволяє вам максимально точно уявити його призначений для користувача інтерфейс перш, ніж приступити до написання коду. Прототипування дозволяє дизайнерам юзабіліті експериментувати і пробувати нові ідеї, а також перевіряти функціональність і юзабіліті продукту, щоб даремно не витрачати гроші на розробку. Завдяки створенню прототипу, значення кожної функції стає очевидним, а команда UX-дизайнерів може побачити дизайн в цілому і усунути будь-які невідповідності і недоліки. Створюючи прототип інтерфейсу перед тим, як приступити до розробки, юзабілістів економлять не лише гроші, а і час.

Додаткові переваги розробки прототипів:

- Ви можете швидко випробувати свої ідеї на користувачах.
- Щоб створити прототип, досить мати під рукою ручку і папір.
- Як тільки прототип готовий, ви можете використовувати його, щоб отримати відгуки від потенційних клієнтів, після чого продовжити розробку і вдосконалити початковий варіант.
- В прототип можна легко і просто внести зміни, витративши при цьому мінімум коштів.
- Прототип можна використовувати для демонстрації вашого продукту начальству, клієнтам та іншим зацікавленим особам, щоб у них склалося чітке уявлення про вашому дизайні.
- Взаємодія завжди вимагає часу і змін. Прототипи дозволяють вам випробувати дизайн на собі і отримати користувацький досвід в режимі реального часу.

1.11. Як вибрати метод дослідження UX?

Томер Шарон, старший науковий співробітник UX в Google, пропонує визначити пріоритетність методу розслідування, для цього необхідно відповісти на питання:

· Яку інформацію ви хочете отримати?

· Що дадуть вам ці дані?

· Які прогалини ви маєте намір заповнити цими знаннями?

· Які рішення?

· Які ваші поточні припущення?

Ці питання використовуються для визначення мети, будь їй вибір продукту, вибір кольору для кнопки і т.д. Щоб відповісти на ці питання, необхідно провести внутрішні дослідження. Після них погляньте, як Томер Шарон категоризує всі ваші запитання:

· Що потрібно людям?

· Чого хочуть люди?

· Як люди можуть це використовувати?

Наприклад, якщо ви хочете запустити новий продукт, ви повинні з'ясувати, що потрібно людям, перш ніж вибрати дизайн.

1.12. Як здійснюється тестування

Тестування здійснюється різними способами:

- прості спостереження;
- опитування;
- анкети;
- інтерв'ю.

По суті, для проведення ефективного юзабіліті-тестування вам знадобиться чіткий план, в якому ви позначили цілі даного тестування, а також кілька реальних користувачів, які будуть взаємодіяти з інтерфейсом, ділитися своєю думкою і відповідати на питання. Перш ніж запрошувати користувачів, вам потрібно задати собі питання: «Яку інформацію мені потрібно отримати з цього тестування?» І, тільки відповівши на нього, ви можете приступати до складання опитувальника. Якщо ви не збираєтеся проводити інтерв'ю або складати анкету, а просто хочете поспостерігати за поведінкою користувачів при взаємодії з вашим сайтом або продуктом, то завдання залишається тією ж.

Можливо, варто запросити всіх членів команди поспостерігати за реакцією користувачів під час тестування. Це допоможе їм зрозуміти проблеми юзабіліті і проявити емпатію (знову це слово!) По відношенню до користувача. Щоб не турбувати користувачів, варто задіяти веб-камери, screen sharing і мікрофони.

Холгер Еггерт (HolgerEggert), головний юзабіліст компанії LevelGreen, дав нам свої рекомендації щодо тестування користувачів:

«Не намагайтеся вирішити всі проблеми за раз. Почніть з найістотніших, після чого знову проведіть тестування. Це дозволить вам заощадити час».

Після закінчення тестування ви зможете проаналізувати результати і на їх основі скласти звіт. При необхідності ви зможете внести зміни в свій дизайн, перш ніж переходити до фази його реалізації.

Проектувальник інтерфейсів в компанії Graph Алекс Ширазі (Alex Shirazi) говорить: «Багато дизайнерів сприймають проектування як лінійний процес. Однак його варто розглядати, як процес динамічний. Наприклад: ідея, каркаси, макет, прототипи. При наявності ресурсів важливо проводити тестування за участю користувачів на кожному етапі цього процесу».

1.13. Чому необхідно проводити дослідження UX

Ви розробляєте продукти для людей, це означає, що всі ваші рішення повинні відштовхуватися від їх бажань і переваг. Дослідження користувача поведінки дозволяє розробляти концепції, орієнтовані саме на їх зручність, і вирішувати більшість потенційних проблем на ранніх етапах, уникаючи внесення змін потім – більш трудомістких і дорогих.

Використовуйте цю можливість і ресурси UX – пропонуйте клієнтам більш якісні послуги і зробіть ваш бізнес більш успішним.

Ось деякі методи досліджень, які повинен знати кожен дизайнер, коли він починає роботу над проектом, і навіть якщо він сам не займається дослідженнями, він може краще спілкуватися з UX дослідниками.

Контрольні питання

1. Сучасні тенденції в дизайні
2. Історія дизайну інтерфейсів та веб-дизайну
3. Етапи роботи над проектуванням інтерфейсу
4. Особливості технічного завдання на розробку дизайну інтерфейсу
5. У чому полягає UX дизайн?
6. Що є основним інструментом у UX дизайні?

Теми для самостійного опрацювання

1. Емпатія в дизайні
2. Ергономіка людино-машинних інтерфейсів
3. Дизайн-концепція інтерфейсів. Принципи візуального дизайну.
4. Дизайн-процеси. Рівні роботи над проектом
5. Методики креативності та генерації ідей
6. Правила компонування
7. Побудова карти сайту для технічної документації.

Тестові завдання

1. Що сприяло появі графічного інтерфейсу людино-машинної взаємодії

a) поява маніпуятора миш
b) поява КПК
c) поява клавіатури
d) винахід процесора

2. Збільшення на результати масштабування растрового зображення впливає

a) позитивно
b) негативно
c) спочатку позитивно, потім негативно
d) спочатку негативно, потім позитивно

3. Виберіть розширення графічного файлу

a) *.doc b) *. flif c) *.exe d) *.bak

4. Мінімальною одиницею, якою оперують в растровому графічному редакторі є

a) палітра кольорів
b) знакомісце (символ)
c) точка екрану (піксель)
d) об'єкт (прямокутник, коло і т.д.)

5. Якість растрового зображення оцінюється:

a) розміром зображення.
b) кількістю пікселів.
c) кількістю пікселів на дюйм зображення.
d) кількістю біт в збереженому зображенні.

6. Піксель на екрані монітора являє собою:

a) електронний промінь.
b) сукупність 16 зерен люмінофора.
c) двійковий код графічної інформації.
d) мінімальний ділянку зображення, якому незалежним чином можна задати колір.

7. Сітка, яку на екрані утворюють пікселі, називається:

a) Дисплейний процесор.
c) Відеоадаптер.
b) Растр.
d) Відеопам'ять.

8. Деформація зображення при зміні розміру малюнка – один з недоліків

a) векторної графіки
c) Тривимірна графіка
b) растрової графіки
d) інженерна графіка

9. Стиль Metro було розроблено компанією

a) IBM
c) Google
b) Microsoft
d) Apple

10. Стиль Material Design було розроблено компанією

a) IBM
c) Google
b) Microsoft
d) Apple

2. КОМПОЗИЦІЯ – ЯК ЗАСІБ ПОЄДНАННЯ ЕЛЕМЕНТІВ UI ДИЗАЙНУ

План:

2.1. Основи композиції
- 2.1.1. Основні види площинних композицій та типи балансу
  - 2.1.1.1 Горизонтальний і вертикальний баланс
  - 2.1.1.2. Радіальний баланс
  - 2.1.1.3. Симетричний і асиметричний баланс
- 2.1.2. Якірні об'єкти та правила їх розміщення
  - 2.1.2.1. Правило розміщення в кутових точках і в центрі.
  - 2.1.2.2. Правило силових сторін.
  - 2.1.2.3. Базові елементи композиції
- 2.1.3. Візуальна ієрархія
- 2.1.4. Групування елементів
- 2.1.5. Рими блоків

2.2. Паттерни читання
- 2.2.1. Діаграма Гутенберга
- 2.2.2. Z-паттерн і його зв'язок з діаграмою Гутенберга
- 2.2.3. F-паттерн і його зв'язок з діаграмою Гутенберга
- 2.2.4. Похідні патерни зигзаг і золотий трикутник

2.3. Точка фокусування та візуальні направляючі

2.4. Принципи Гештальта
- 2.4.1. Закон близькості.
- 2.4.2. Закон схожості.
- 2.4.3. Закон закритості.
- 2.4.4. Закон симетрії.
- 2.4.5. Закон загального поля.
- 2.4.6. Закон пов'язаності.
- 2.4.7. Закон спільної долі.

2.5. Центральний візуальний образ

2.6. Модульна сітка

2.7. Колір як засіб композиційної побудови

2.8. Класичні принципи досягнення балансу
- 2.8.1. Золотий перетин / золота спіраль
- 2.8.2. Правило третин

2.9. «Повітря»

2.10. Використання відступів

2.11. Створюємо баланс в інтерфейсах

2.12. Характеристики балансу
- 2.12.1. Розмір
- 2.12.2. Колір
- 2.12.3. Форма
- 2.12.4. Значимість
- 2.12.5. Розташування

2.13. Закони взаємодії

2.1. Основи композиції

Гармонія в співвідношеннях частин цілого, краса і логіка побудови властива не тільки творінням людини. Ознаки композиції можна виявити в будові всесвіту, тваринних організмів і рослин, природних формах. Тому термін «композиція» можна однаково застосувати до будови веб-сайту, опису квітки і під час характеристики ораторської мови.

Найпростіше говорити про композицію в інтерфейсах, коли вона не дотримана. Кожен з нас інтуїтивно відчуває її присутність / відсутність: чи знаходяться елементи в гармонійній взаємодії, підпорядковані вони головній ідеї сайту, картини, фотографії. Щоб дизайн сайту виглядав природно і гармонійно, щоб не розвалився на окремі складові, дотримуйтесь базових принципів дизайну. Щоб створити дійсно якісний сайт, який буде грати на користь як іміджу компанії, так і надавати прибуток, необхідно приділити увагу всім елементам його візуальної складової.

Композиція – це складання малих частин в одне ціле. Також композицію можна уявити як гру. Є два гравці: ви (творець композиції) і спостерігач. Мета гри – змусити людину виконати певні дії або доставити йому певну інформацію. Якщо у гри немає мети, ви ніколи не виграєте. *Вибудовуючи композицію, ви задаєте правила гри і керуєте поведінкою опонента.*

Чому ми можемо управляти опонентом? Композиція – це ще й середовище, в яке ми поміщаємо спостерігача. А поведінка будь-якої людини безпосередньо залежить від його особистості і середовища, в якому він знаходиться.

$$B = f(P, E)$$

де B – поведінка людини, f – знак функціональної залежності, P – особливості особистості людини, E – середовище

2.1.1. Основні види площинних композицій та типи балансу

До основних видів площинної композиції відносять

- симетричну
- асиметричну

Також виділяють:

- горизонтальну
- вертикальну
- діагональну
- кругову
- глибинну
- ракурсну
- завжди можна створити свій власний варіант

Існує безліч різних типів балансу – тобто способів взаємодії елементів композиції. За допомогою різних методів, описаних в цій статті, ви зможете так налаштувати "гравітацію" елементів, щоб створити у користувача відчуття збалансованості і комфорту.

Види балансу:

- горизонтальний баланс
- вертикальний баланс
- радіальний баланс
- симетричний баланс
- асиметричний баланс

2.1.1.1 Горизонтальний і вертикальний баланс

Відмінний спосіб керувати візуальної важливістю елементів – це збалансувати їх відносно центральної осі: горизонтальної або вертикальної. Горизонтальний баланс досягається балансуванням елементів, розташованих праворуч і ліворуч – *як на вагах.* Вертикальний баланс означає, що збалансовані елементи, які стоять вище і нижче центральної осі.



Рисунок 2.1 – Приклад горизонтального балансу сайту likeness.com та вертикального балансу сайту danielmart.in

2.1.1.2. Радіальний баланс

Замість того, щоб вибудовувати елементи відносно центральної осі, ви можете відштовхуватися від центральної точки – і отримаєте радіальний баланс. Фактично ви розміщуєте елементи навколо центральної осі; можна навіть вибудувати їх так, немов вони вилітають з центру.



Рисунок 2.2 – Демонстрація радіального балансу на прикладі сайту ipolecat.com

2.1.1.3. Симетричний і асиметричний баланс

Ще один тип балансу, який зазвичай комбінують з горизонтальним, вертикальним або радіальним – це симетричний і асиметричний баланс. Симетричний баланс означає, що обидві сторони композиції є дзеркальними відображеннями один одного – немов папір склали навпіл. Симетричний баланс дуже формальний: він приємніше погляду, тому його часто використовують в архітектурі і при створенні веб-лейаутів на основі сітки. З ростом популярності адаптивного дизайну**,** зростає і популярність симетричного балансу, тому що такий баланс простіше відтворити в різних розмірах екрану.



Рисунок 2.3 – Веб-дизайн ресурсу frontend2011.com

Асиметрія, мабуть, найпоширеніша форма балансу. По суті це баланс, зворотний симетричного: елементи композиції збалансовані, але розташовані не строго відносно один одного. Асиметрія набагато цікавіше симетрії, тому що в симетрії зазвичай відсутня візуальна ієрархія, а ось в асиметричних лейаутах ієрархія помітна відразу. Те, що композиція асиметрична, не означає, що в ній немає балансу: просто елементи розташовані не дзеркально.



Рисунок 2.4 – Приклад асиметричного дизайну teamviget.com

2.1.2. Якірні об'єкти та правила їх розміщення

Створення композиції також підпорядковується кільком основним правилам розміщення якірних об'єктів. *Якірні об'єкти* – це ключові точки взаємодії мозку користувача з областю композиції

2.1.2.1. Правило розміщення в кутових точках і в центрі.

Саме ці місця на площині людський мозок обробляє найкраще, вони є найактивнішими в прямокутнику.

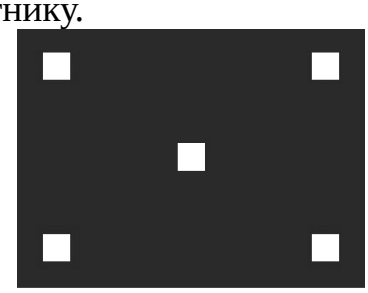


Рисунок 2.5 – Правило розміщення якірних об’єктів в кутових точках і в центрі

2.1.2.2. Правило силових сторін.

Якірні об'єкти також можна розмістити вздовж однієї з силових сторін композиції.



Рисунок 2.6 – Правило силових сторін

Тип композиції і правило розміщення якірних об'єктів необхідно обирати виходячи з цілей контенту. При цьому контент є основою всієї композиції. Вам потрібно зробити таку композицію, щоб контент вдало зіграв з спостерігачем. Для грамотної подачі контенту необхідно розробити свою метричну систему. Для цього необхідно:

1. Визначитися з контентом.
2. Вибрати область композиції, її формат.
3. Врахувати навколишнє середовище
4. Врахувати умови взаємодії
5. Вивчити особливості споживання.
6. Врахувати інші фактори.

Для створення композиції, як правило, використовують сітки. Сітки бувають: стовпчикові, рядкові, в клітку, в косу лінію, радіальні, сітка Вілларда (мікроформатні), ромбічні, фасетного типу тощо. Щоб людина зрозуміла, на що дивиться, мозку потрібно провести певну роботу:

1. Провести селекцію. Людина виділяє об'єкт, на який дивиться.
2. Організувати інформацію. Об'єкт ідентифікується по ряду ознак. Мозок шукає в пам'яті схожі атрибути.

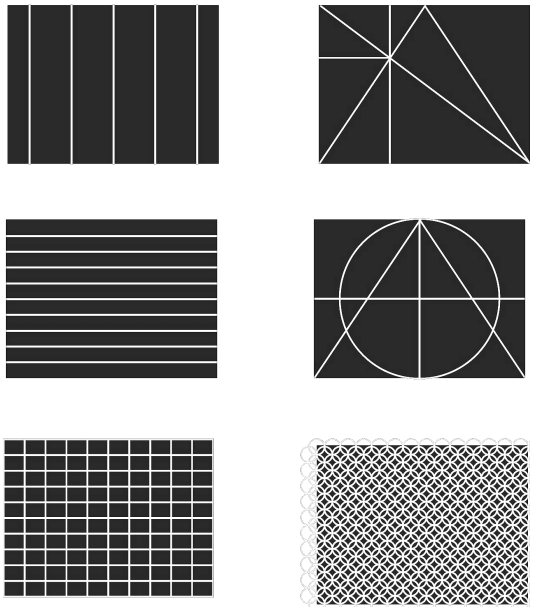


Рисунок 2.7 – Сітки для створення композиції

3. Провести категоризацію і приписати спостережуваного об'єкту властивості інших об'єктів цього класу.

Сприйняттям людини можна управляти. Цього ми і добиваємося в своїй композиції. Ваш опонент буде розглядати сторінку сайту або додатку так, як ви йому скажете і подасте інформацію. Ви можете направити його будь-яким із шляхів, який потрібен для досягнення ваших цілей. Якщо ваша композиція складена правильно, спостерігач буде взаємодіяти з тими якірними об'єктами і в тій послідовності, в якій ви йому покажете.

2.1.2.3. Базові елементи композиції

Компонентами композиції є: крапка, лінія, прямокутник, окружність, багатокутник. Всі макети можуть складатися з цих базових елементів.

2.1.3. Візуальна ієрархія

У кожного базового елементу є свої оптичні особливості.



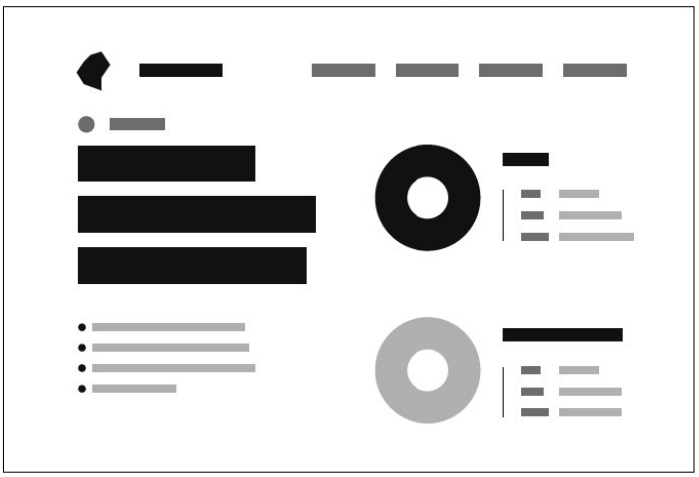


Рисунок 2.8 – Базові елементи та прототип, що побудований з базових елементів

Щоб окружність і прямокутник виглядали одного розміру, окружність повинна бути більше прямокутника, адже мозок нехтує крайніми частинами кола, не сприймаючи його як пляма.

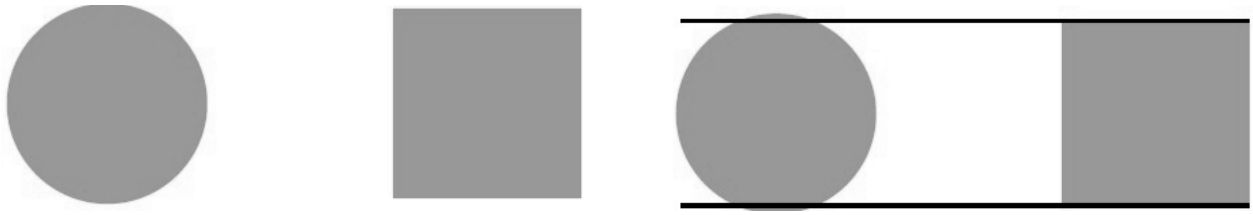


Рисунок 2.9 – Оптичні особливості кола

Те ж правило стосується і гострих кутів.



Рисунок 2.10 – Оптичні особливості трикутника

Багатокутники і прямокутники вирівнюються по центру маси.



Рисунок 2.11 – Оптичні особливості багатокутника

Квадрати і кола будуть здаватися більш ідеальної форми, якщо їх трохи витягнути.

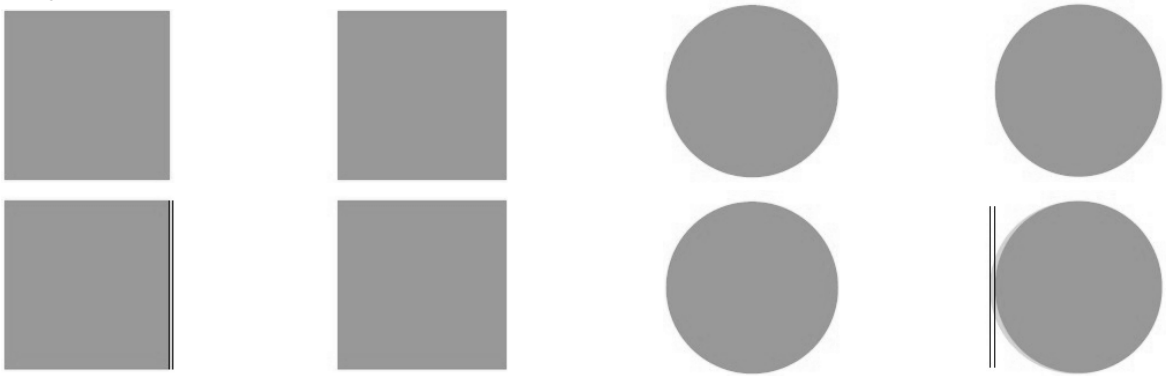


Рисунок 2.12 – Оптичні особливості квадрата та кола

Пам'ятайте про візуальну ієрархію, коли почнете проектувати сторінки, і не забувайте, що у вас під рукою маса самих різних елементів, за допомогою яких можна подати інформацію максимально ефективно, в повній відповідності з правилами візуальної ієрархії. Це кольори і відтінки, тексти і картинки різних розмірів, комбінації шрифтів і, звичайно ж, лінії – вертикальні, горизонтальні і діагональні. Повірте, користувачі з задоволенням пройдуть по запропонованим вами шляху і прочитають все як годиться: спершу найголовніше, а вже потім – все інше. Засоби для вибудови візуальної ієрархії: масштаб, колір, контраст, вирівнювання

Розуміючи, на що людина дивиться в першу чергу, а на що – в другу і в третю, ви можете розподілити інформацію найбільш вигідним для вас способом. Якщо необхідно, щоб покупці не пройшли повз певної кнопки,

розташуйте її туди, куди в першу чергу падає погляд, і виділяйте яскравим кольором. Якщо на меті показати, якість товарів за допомогою фотографій, знайдіть для них найкраще місце і простежте, щоб вони не зливалися з фоном.

Щоб ваш опонент добре вважав якірний об'єкт, потрібно дати йому трохи простору (white space). Це потрібно, щоб полегшити сприйняття елементів мозку вашого опонента. Чим більше «*білого простору*» у об'єкта, тим легше його помітити і вважати. Ви також можете привернути увагу опонента за допомогою збільшення або зменшення цього простору.

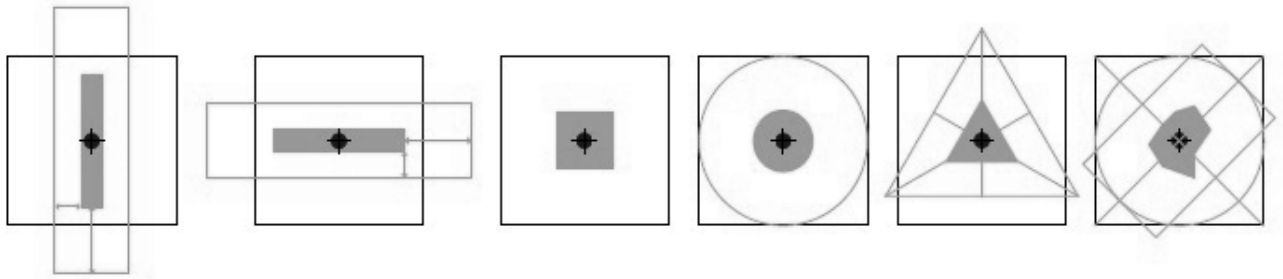


Рисунок 2.13 – Формування «білого простору» в залежності від форми об’єкту

З цього виведено правило *«внутрішнього і зовнішнього»***.**

Внутрішній відступ повинен бути рівний або бути менше, ніж зовнішній.
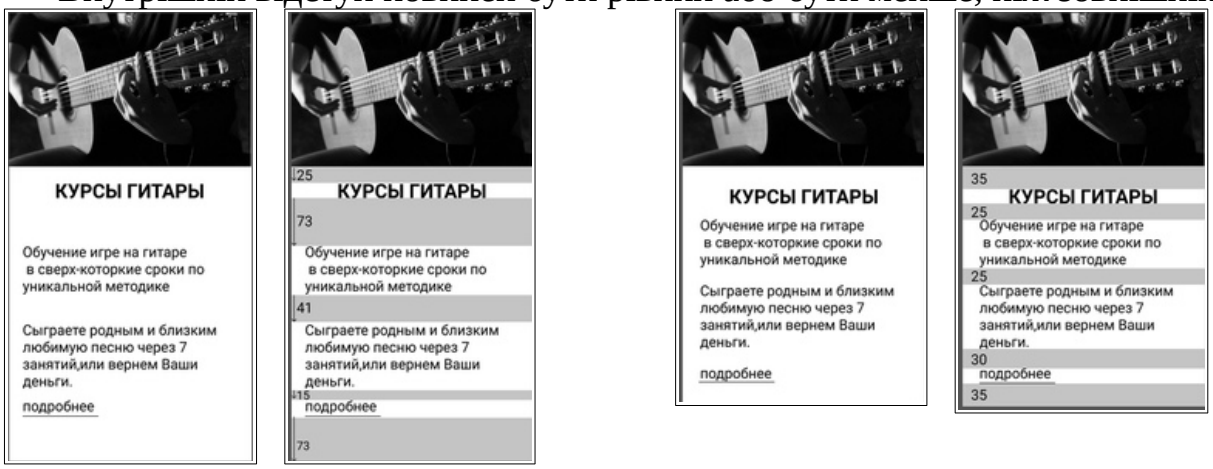


Рисунок 2.14 – Демонстрація формування полів та відступів за правилом «внутрішнього і зовнішнього»

Чим простіше сприймається вашим опонентом композиція, тим краще. Ваше завдання – зробити сприйняття всіх елементів композиції максимально простим.

2.1.4. Групування елементів

Існує 3 типи групування інформації за принципом впливу, які ми можемо подати в площинному дизайні: Інфо-блок, Промо-блок та блок розриву

Інфо-блок – це текст. Він повинен передати повідомлення. Складається з безлічі багатокутників (букв) і складний для сприйняття. Інфо-блок є найповільнішим з передачі інформації опонентові.

Промо-блок – це відео або зображення, взаємодіє з емоціями і повинен заворожити людини, використовуючи основні важелі маніпуляції.

Блок розриву – розрив контенту і сюжету в композиції. Блоком розриву може бути контрформа, смужка або відсутність контенту.

2.1.5. Рими блоків

Мозок найкраще сприймає інформацію, коли вона різноманітна. Мозок не потрібно перенапружувати однотипної інформацією. Права частка мозку більше працює з зображеннями, а ліва збуджується при аналітичному мисленні і обробці текстової інформації. Варто чергувати навантаження на мозок опонента. Блок розриву можна ставити там, де він буде вигідний.

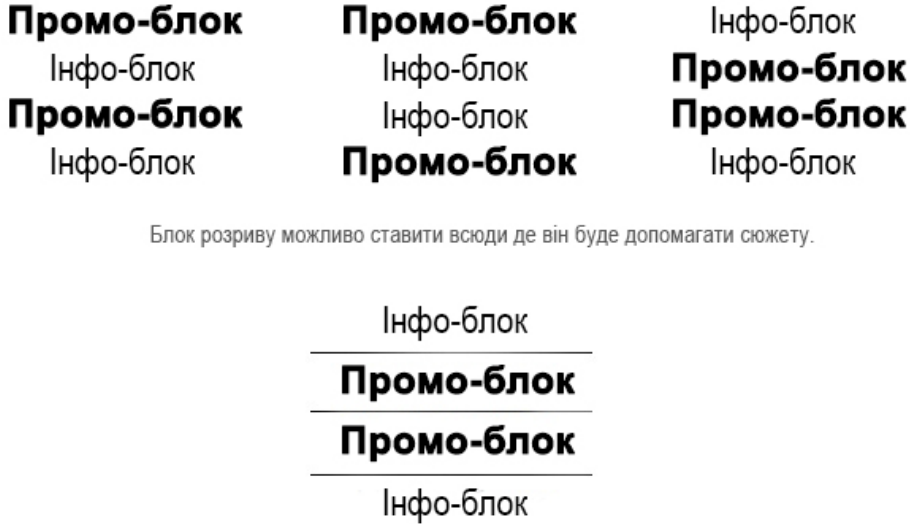


Рисунок 2.15 – Рими блоків

2.2. Паттерни читання

Типограф, автор освітніх проектів по типографіці Меттью Баттерік визначає типографіку, як візуальну складову надрукованого тексту. Гарне, ємне формулювання з великим простором для маневру. Саме через це під терміном часто розуміють досить різні речі. Типографіка – візуальна складова надрукованого тексту. Те, що відноситься до UX, здебільшого стосується макротипографіки – розташування тексту на сторінці. Мікротипографіка – це тонка настройка, робота на рівні символів.

Чи важлива типографіка для проектування користувацького інтерфейсу? Звичайно. Є дослідження, які пояснюють, як параметри шрифту приносять істотні конверсії, як керувати емоціями користувача і дають зрозуміти, що недотримання достатнього контрасту в інтернеті буде дорого коштувати. Подібні речі дійсно впливають на сприйняття. Ключовий момент, з яким потрібно розібратися, – як люди читають. Дуже багато досліджень присвячено тому, як люди переглядають веб-сторінки. Виявляється, вони сканують екран зверху вниз і зліва направо, а траєкторія руху їхніх очей в цей момент нагадує букву F або букву Z. Класичні книги, статті, в основному не зупиняються на фізіології процесу читання Найчастіше мова про діаграму Гутенберга і різних патернах читання.

2.2.1. Діаграма Гутенберга

Діаграма Гутенберга розповідає про те, що при великому обсязі контакт з текстом починається з лівого верхнього кута і слабшає по певним законам. На основі цих даних будуються відомі патерни читання: z-патерн, f-патерн та похідні.

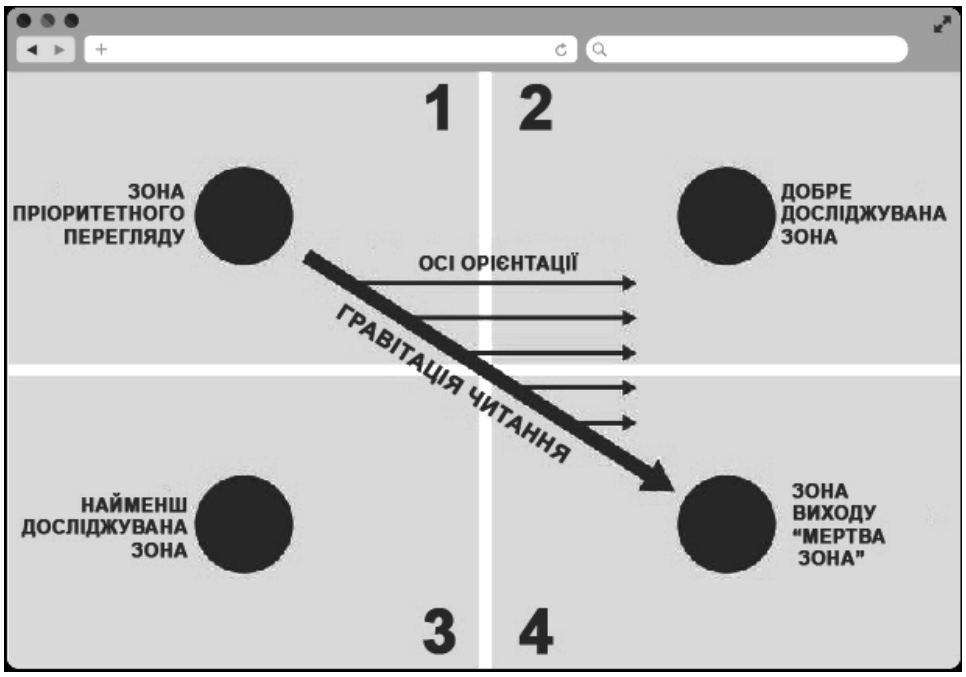


Рисунок 2.16 – Схема траєкторії руху погляду людини за діаграмою Гутенберга

1. Відразу відвідувач дивиться в лівий верхній сектор екрану (зона пріоритетного перегляду).
2. Потім зміщує погляд вправо (добре досліджувана зона).
3. Швидко перескакує на нижній лівий сектор (найменш досліджувана зона).
4. І тут же йде в праву сторону по горизонталі (зона виходу).

Фактично, найбільшу увагу привертають сектори 1 і 4 – пріоритетна і зона виходу. Більшість відвідувачів вашого сайту рухаються по діагоналі від 1-ї до 4-ї зони, лише на мікросекунди затримуючись на секторах 2 та 3. І якщо правому верхньому кутку ще дістається трохи уваги, то лівий нижній практично не проглядається. Це відбувається через так звану гравітацію читання – ми автоматично зміщуємо погляд зверху вниз при вивченні контенту. Пишемо ми зліва направо, тому погляд так само автоматично зміщується в праву сторону. У поєднанні з гравітацією це і спрямовує вектор уваги з лівого верхнього в правий нижній кут. Значущу інформацію – логотип, оффер, слоган – розташуйте в лівому верхньому кутку екрану. Це допоможе користувачеві відразу ідентифікувати вашу компанію, послугу або основні вигоди. Ваше завдання – заволодіти увагою і направити відвідувача на вчинення цільового дії. У правому нижньому секторі – кнопка замовлення, заклик до дії, оффер, мотиватор для скроллу. Саме тут відвідувач приймає рішення – залишитися або піти, швидко просканувати сторінку або уважно прочитати, варто чи не варто мати з вами справи. У правому верхньому секторі можна розташувати другорядну інформацію: телефони, форму для замовлення зворотного дзвінка, час роботи, регіон. Відвідувач пробіжить очима, відзначити, що саме тут є, і піде далі. Якщо знадобиться – підсвідомість дбайливо підкаже, де все це можна знайти.

Лівий нижній сектор – "мертва зона". Якщо розмістити тут важливе повідомлення, відвідувач з великою часткою ймовірності або зовсім його не

помітить, або відфільтрує як неважливе. Щоб було зрозуміліше, як всі ці рекомендації працюють і реалізуються в реальному світі, ми зібрали кілька прикладів.

1. **Стартова сторінка Facebook** – соціальної мережі # 1 в світі – спроектована в суворій відповідності з діаграмою Гутенберга.

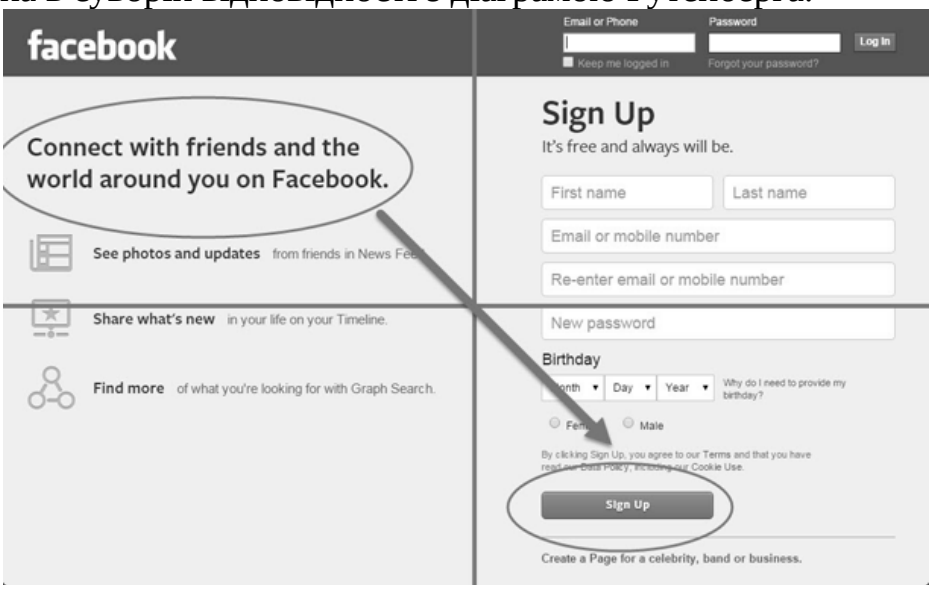


Рисунок 2.17 – Траєкторія руху уваги користувача сторінкою реєстрації Facebook, що ілюструє застосування діаграми Гутенберга

Логотип і заклик до дії – в зоні пріоритетного перегляду, кнопка відправки форми – в зоні виходу. Зверніть увагу, найменш досліджувана зона практично порожня, а в добре досліджувану зону винесена форма входу для існуючих користувачів (вона не відволікає від основного цільового дії – реєстрації, але при цьому її не потрібно шукати).

2.2.2. Z-паттерн і його зв'язок з діаграмою Гутенберга

Науково доведено сотнями досліджень, що людина сканує інформацію на сторінці за певною траєкторією. Рухи зіниць в цей момент нагадують літеру Z, а увага концентрується в кількох ключових точках.

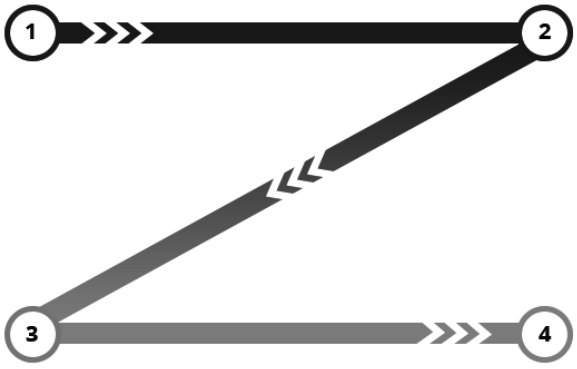


Рисунок 2.18 – Схема траєкторії руху погляду людини за принципом «Z-макет»

Z-макет є зараз дуже поширеним. Відповідно до нього, елементи сторінки, як правило, розташовані в такий спосіб: логотип в лівому верхньому кутку, меню в правому верхньому куті, інформаційні блоки та / або зображення в лівому нижньому кутку, а кнопка із закликом до дії внизу праворуч.

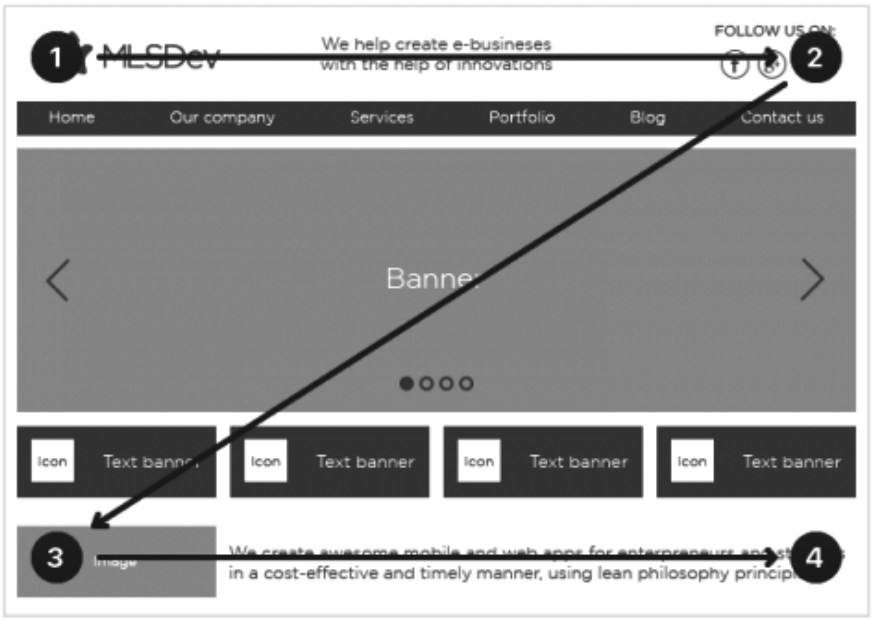


Рисунок 2.19 – Приклад вайрфрейму із застосуванням принципу «Z-макет»

2.2.3. F-паттерн і його зв'язок з діаграмою Гутенберга

Також поширена траєкторія у вигляді літери F. Суть її у наступному – у верхній частині сторінки людина сприймає максимум можливої інформації, до середини – лише половину, і в кінці – зовсім небагато. Найчастіше, якщо на сторінці присутній яскравий і помітний елемент, погляд моментально концентрується саме на ньому, незалежно від того, в якій частині сторінки цей елемент розташований. Тому остерігайтеся того, що він стане своєрідним «образом-вампіром», що буде відволікати відвідувачів від дійсно потрібної інформації.

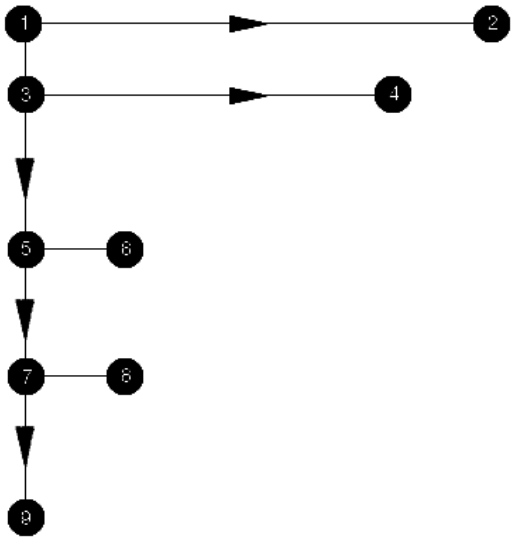


Рисунок 2.20 – Схема траєкторії руху погляду людини за принципом «f-макет»

В середині 2000-х американська консалтингова компанія Nielsen Norman Group провела цікаве дослідження. Аналітики вивчили, як 232 людини, відібраних випадковим чином, переглядають сайти і склали на основі цих даних теплові карти.

З'ясувалося, що користувачі сканують контент по латинської букви F. На зображенні це чітко видно. Відразу – лівий верхній кут, потім перехід по діагоналі майже до кінця активного вікна, вниз і наліво, знову по горизонталі вправо, потім вниз практично вертикально.

Якщо поглянути на знімок сторінки, здається, що F-паттерн суперечить діаграмі Гутенберга. Насправді це не зовсім так.

Діаграма Гутенберга в більшій мірі визначає поведінку користувача при контакті з різними видами контенту на одному екрані: текст, графіка, фото, відео. Вище, на середній ілюстрації, де крім тексту є зображення і винесений окремо контентний блок це чітко видно.

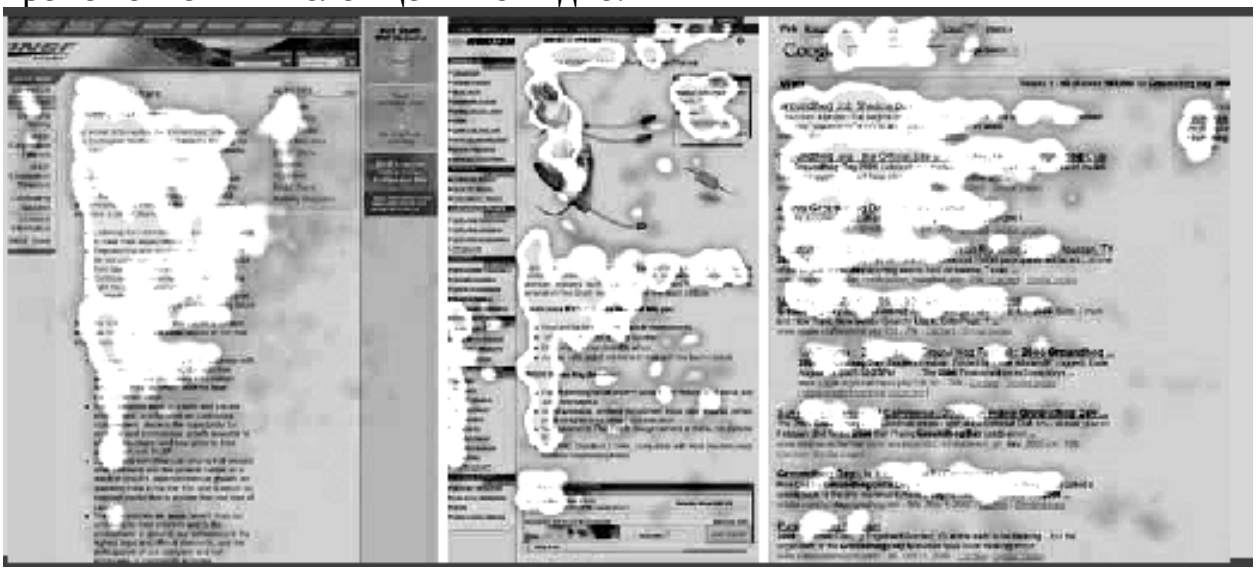


Рисунок 2.21 – Теплова карта (дослідження Nielsen Norman Group)

F-патерн в більшій мірі визначає поведінку користувача при читанні текстового або просто однорідного контенту. Але дослідження проводилося більше 10 років тому. За 10 років дослідження втратило свою актуальність, бо не враховує поведінку мобільних користувачів, яких тоді просто не було, форми подачі контенту з того часу також сильно змінилися.

2.2.4. Похідні патерни зигзаг і золотий трикутник

Ще два патерна, що тісно пов'язані з F і Z моделями – зигзаг і золотий трикутник.

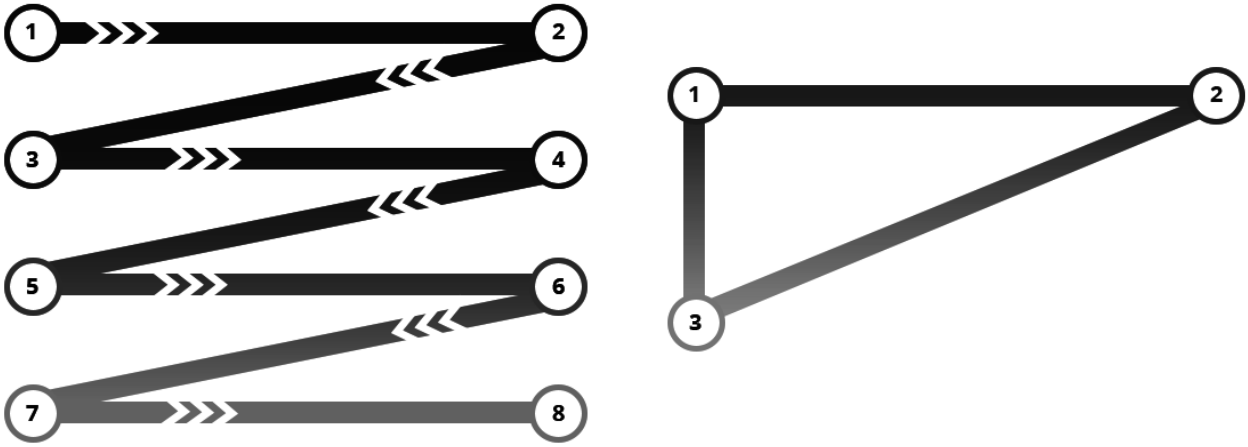


Рисунок 2.22 – Похідні патерни зигзаг і золотий трикутник

Вони добре вписуються в *модель Якоба Нільсена* і описують поведінку користувача при контакті з текстовим контентом. Але інтернет сьогодні – не тільки тексти. Німецький друкар Ян Чіхольд ще в 1928 році прийшов до висновку, що сучасна людина не хоче читати об'ємні тексти. «Сканує» полотно, знаходить щось корисне і тільки тоді зупиняється і починає читати. Навряд чи Чіхольд цікавився дослідженнями по фізіології, але, тим не менш, розумів, як все відбувається. А відбувається наступне: на сітківці ока є маленька западина, яка відповідає за розпізнавання предметів. Вона дуже маленька. Тобто кожен момент часу чітко ми бачимо дуже маленький фрагмент картинки – всього пару градусів огляду. Вся інша картинка розмита. Але мозок картинку домальовує і здається, що чітко бачимо ми набагато більше.



Рисунок 2.23 – Фокусування на частині з великого тексту

2.3. Точка фокусування та візуальні направляючі

Візуальні напрямні – це декоративні компоненти сторінок, які перенаправляють очі користувача в сторону певних елементів дизайну. Елементи дизайну повинні розроблятися відповідно до цілей, які необхідно досягти на сайті. Те, на чому потрібно сфокусувати увагу відвідувача, має привертати увагу і спонукати до дії.



Рисунок 2.24 – Демонстрація дії візуальної направляючої за допомогою теплових карт

Закликаючи користувача зосередити всю свою увагу в певному місці, дуже важливо не перегнути палицю, щоб не здатися занадто настирливим і не відлякати потенційного клієнта.

2.4. Принципи Гештальту

Сутність принципів Гештальту, щодо веб-дизайну, зводиться до того, що всі елементи повинні сприйматися цілісно, щоб виробляти необхідний психологічний вплив на користувача. Тільки так можливо досягти бажаного ефекту.

Грамотне використання принципів Гештальту дозволяє керувати не лише увагою, а й емоціями відвідувачів сайту, спрямовуючи їх у потрібну сторону.

Закони групування через гештальтізм, завдяки яким можна домогтися цілісного сприйняття інтерфейсу:

2.4.1. Закон близькості.

Об'єкти, розташовані поруч, мають тенденцію сприйматися разом.

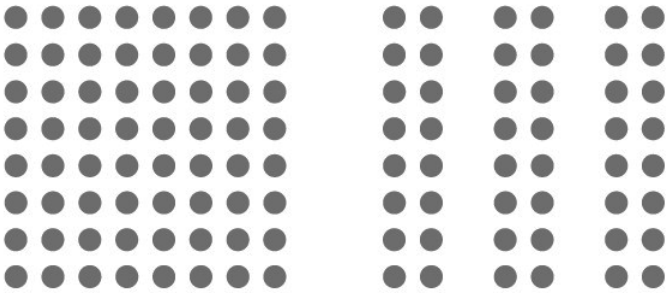


Рисунок 2.25 – Результат сприйняття за законом близькості

2.4.2. Закон схожості.

Об'єкти, які схожі за атрибутами, будуть сприйматися як група.

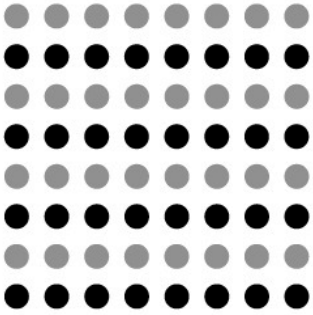


Рисунок 2.26 – Результат сприйняття за законом схожості

2.4.3. Закон закритості.

Неповні силуети добудовуються в сприйнятті до повних.

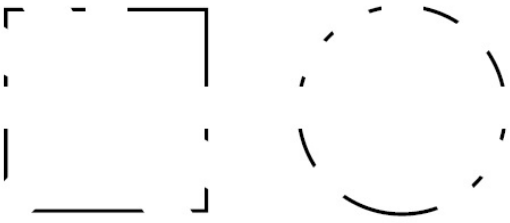


Рисунок 2.27 – Результат сприйняття неповних силуетів за законом закритості

2.4.4. Закон симетрії.

Симетричні об'єкти і вирівняні навколо центральної точки об'єкти сприймаються як частина цілісного об'єкта.

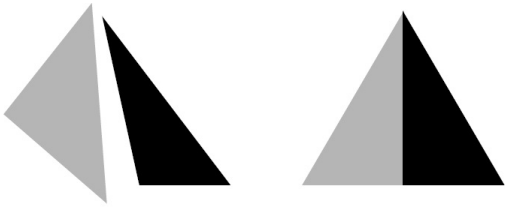


Рисунок 2.28 – Результат сприйняття за законом симетрії

2.4.5. Закон загального поля.

Елементи, які лежать на одній площині або в одному полі, сприймаються як одне з безлічі.

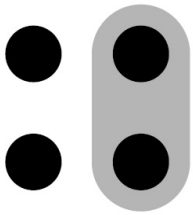


Рисунок 2.29 – Ілюстрування закону загального поля

2.4.6. Закон пов'язаності.

Елементи, об'єднані загальним елементом, сприймаються як одне з безлічі.

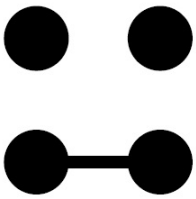


Рисунок 2.30 – Ілюстрування закону пов'язаності

2.4.7. Закон спільної долі.

Елементи з однаковим напрямком руху сприймаються як ціле.





Рисунок 2.31 – Ілюстрування закону спільної долі у рекламі IBM

2.5. Центральний візуальний образ

Суть в тому, щоб розмістити на сайті зображення, що несе в собі сильний емоційний заряд, а інші деталі будувати вже навколо нього. Дуже добре підходить для сайтів музикантів, політичних діячів, окремих кінострічок, а також кулінарної або fashion тематики.

Дуже важливо вибрати правильну картинку, яка буде викликати виключно необхідні для власників сайту емоції.

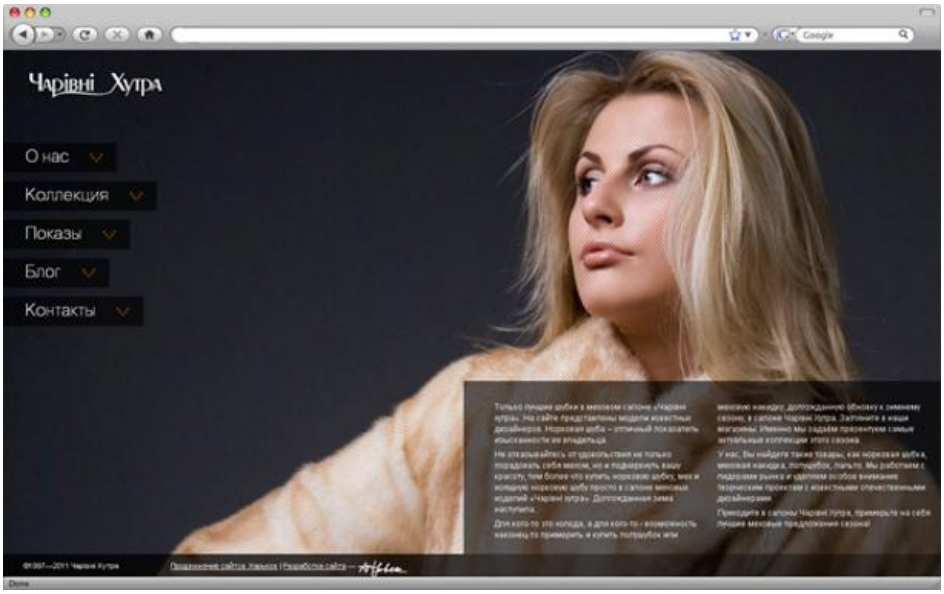


Рисунок 2.32 – Приклад сайту з використанням центрального візуального образу

2.6. Модульна сітка

З її допомогою створюються ідеальні, з точки зору композиції, макети. Використовується вкрай часто, так як дозволяє домогтися досконалості пропорцій і врівноваженості всіх елементів сайту.

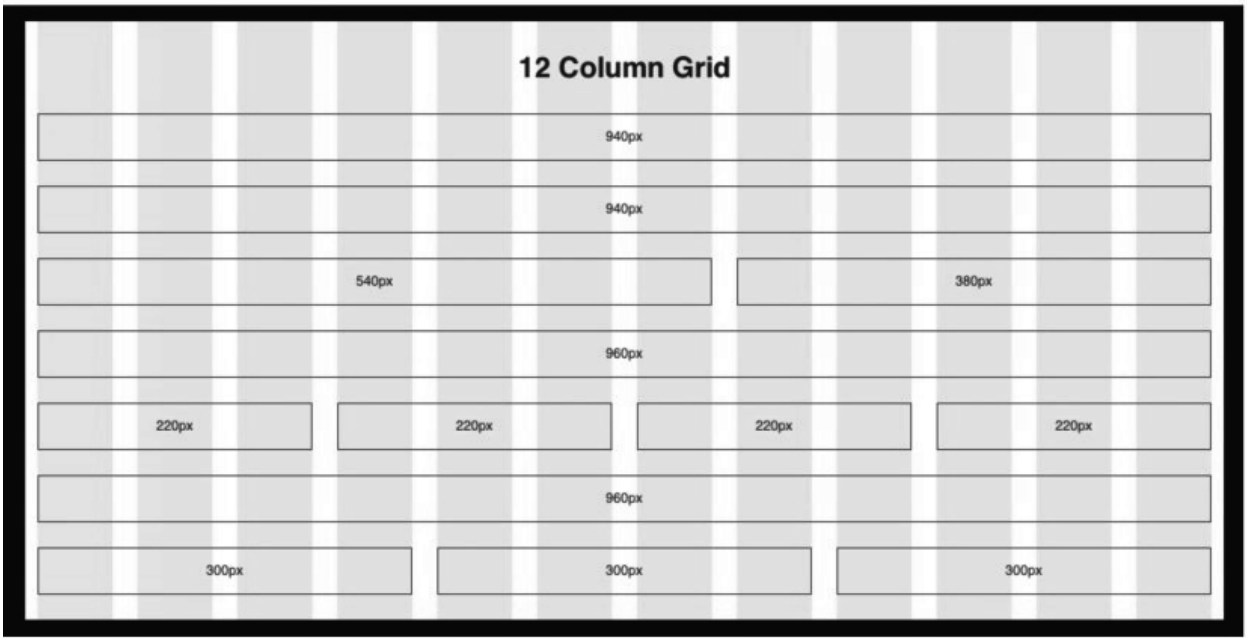


Рисунок 2.33 – Модульна сітка та відповідна система поділу на колонки

2.7. Колір як засіб композиційної побудови

Колір вважається повноцінним елементом композиції. Від нього багато в чому буде залежати те, як люди сприймуть ваш сайт, а разом з ним і продукт. Яскраві, контрастні кольори здаються нам енергійними і бадьорими, приглушені і компліментарні – спокійними, розслабленими. Від вибору кольору залежить те, як люди сприймуть ваш сайт, а заодно, і ваш продукт. Приглушені кольори здаються нам розслабленими і спокійними, а яскраві і контрастні – бадьорими і енергійними.



Рисунок 2.34 – Приклад використання кольору як засобу формування композиції сайту Orangina, що підкреслює імідж бренду

Подумайте, яку емоцію ви бажаєте викликати в відвідувача вашого сайту, а після цього підбирайте відповідну колірну схему.

Для залучення уваги слід використовувати кольорові акценти. У контрастних кольорах можна, наприклад, оформити найважливіші кнопки – «Купити» і «Зв'язатися з нами».

2.8. Класичні принципи досягнення балансу

2.8.1. Золотий перетин / золота спіраль

Майстри через століття зрозуміли сенс поняття "баланс". Фактично, для них він був геометричним і називався «Золотий Перетин», що сформувало «Золоту спіраль». На сьогоднішній день до сих пір залишається загадкою, хто придумав «Золотий перетин», звідки виникло це назва. Однак відомо, що технологія використовується вже на протязі 2400 років. Варто також відзначити, що золотий перетин застосовується в різних галузях науки: в астрономії, математиці, архітектурі, музиці, живопису та багатьох інших. Золотий перетин утворено на прикладі простого математичного рівняння, яке показує відношення. У найбільш простий математичної формі дане відношення виглядає наступним чином:

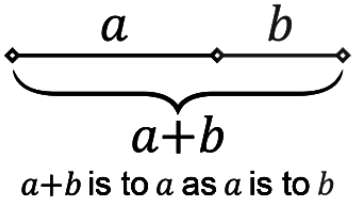


Рисунок 2.35 – Закономірність поділу за принципом золотого перетину

Як ви можете бачити, це є унікальним рівнянням, яке розділяє відносини між двома розмірами і пропорціями ліній. У десятковому обчисленні b поділене на a 1.618033... в разі якщо a>b. У наведеному нижче прикладі припустимо, що b дорівнює 5. Тоді рівняння буде виглядати наступним чином:

$$\frac{a(8.09...) + b(5)}{a(8.09...)} = \frac{a(8.09...)}{b(5)} = \Phi\,(1.618..)$$

Рисунок 2.36 – Обчислення згідно принципу золотого перетину

Можливо, раніше ви вже чули про послідовність Фібоначчі.: наприклад, є ряд чисел, в якому будь-який заданий число створюється за допомогою додавання двох попередніх. Починаючи з 0, то послідовність така 0, 1, 1, 2, 3, 5, 8, 13, 21, 34 …

Письмове вираження представлено у вигляді формули: xn = xn-1 + xn-2.

Послідовність тісно пов'язана із золотою пропорцією, тому що якщо ви берете будь-які дві послідовні цифри і ділите на попередню, дріб виявиться дуже близькою до золотої пропорції. У міру збільшення значення числа, дріб стає ще ближче до золотої пропорції. Наприклад, 8/5 1,6, 34/21 це 1,619 і так далі. Схема називається «*Спіраль Фібоначчі*». Вона насправді досить проста і є найбільш оптимальною для багатьох геометричних фігур. Спіраль створюється за допомогою чверті кіл, які намальовані всередині масиву квадратів на основі послідовності Фібоначчі. На діаграмі нижче представлений зразок:

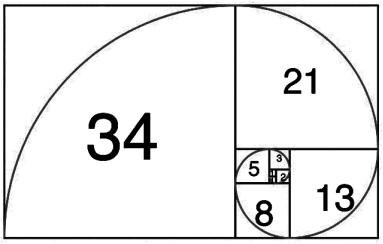


Рисунок 2.37 – Спіраль золотого перетину

Виходить що кожний наступний радіус більше попереднього на число близьке до золотого перетину. Отримана спіраль застосовується в багатьох областях, частіше в малюванні та архітектури, але її можна також спостерігати і в природних явищах. Воно виникло в результаті інтерпретації правила золотого перетину (Леонардо да Вінчі його використовував на своїх картинах «Вітрувіанська людина», Мадонна та ін).

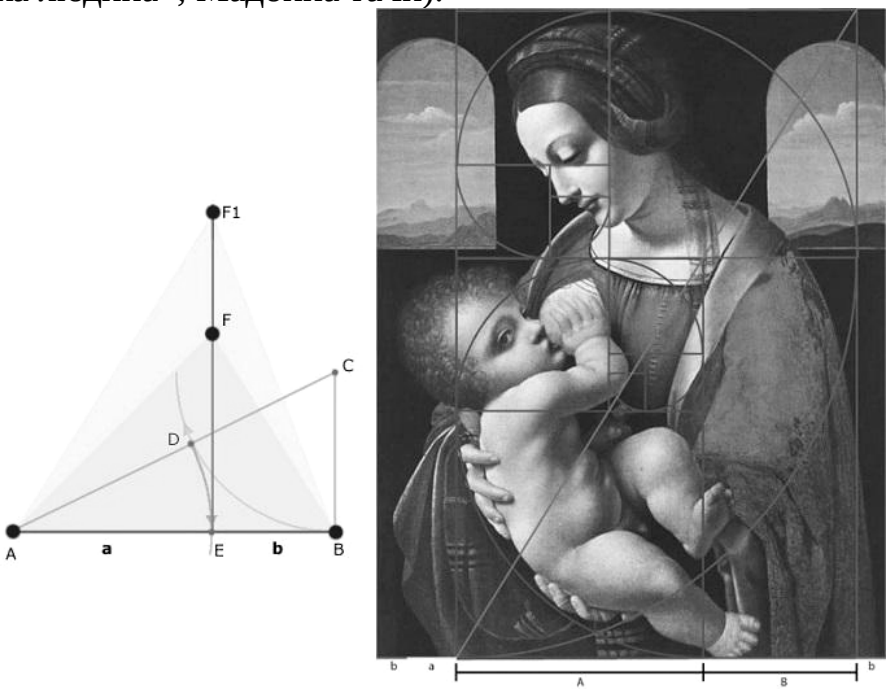


Рисунок 2.38 – Накладання спіралі золотого перетину на Мадонну Леонардо да Вінчі

Багато дизайнерів припускаються помилки, вважаючи що за допомогою звичайного поділу або множення на 1,61... можна отримати гармонійну пропорцію. Це далеко неправда, це всього лише основа процесу. Неможливо просто використовувати якесь число і отримати магічну пропорцію. Однак існують певні способи, які допомагають отримати золотий перетин. Проте деякі дизайнери схильні думати, що теорія про золотий переріз є міфом. Наведемо приклад, як працює золотий перетин в дизайні. Візьмемо прототип сайту і розглянемо застосування золотого перетину на ньому. Оформлення представлено за допомогою золотої пропорції. Якщо використовувати 1 золоту спіраль яка охоплює 960px видно, як розташовувався заголовок, логотип і шапка. Рухаємо спіраль нижче і спираємося на її розміри. Виходить такий собі каскад спіралей в якому основні елементи дизайну вписані в прямокутники із золотою пропорцією. Сітка заснована на золотому перетині має ряд пропорційних відносин всередині нього, які виконані чітко пропорційно всередині прямокутника.



Рисунок 2.39 – Макет, заснований на принципі золотого перетину

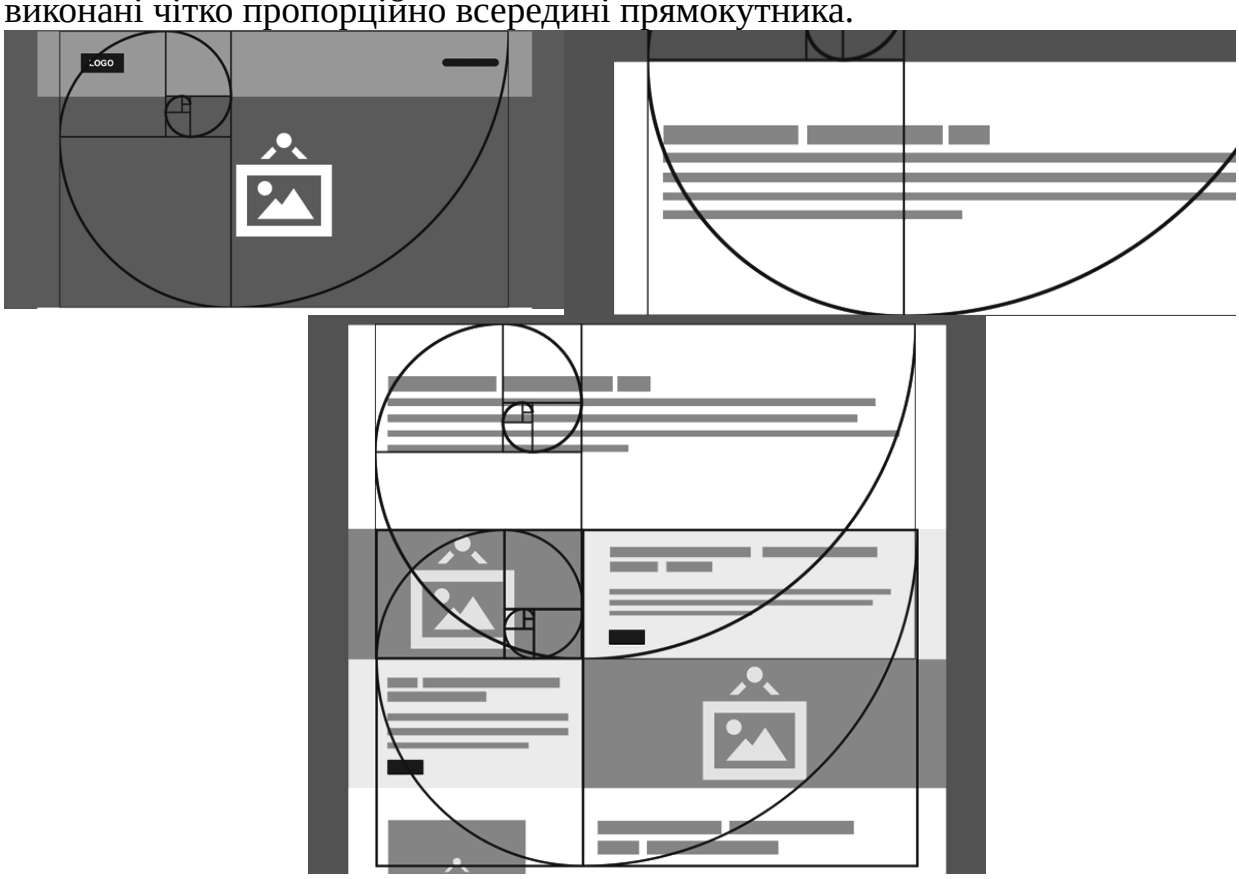


Рисунок 2.40 – Каскад золотих спіралей на фрагменті макету

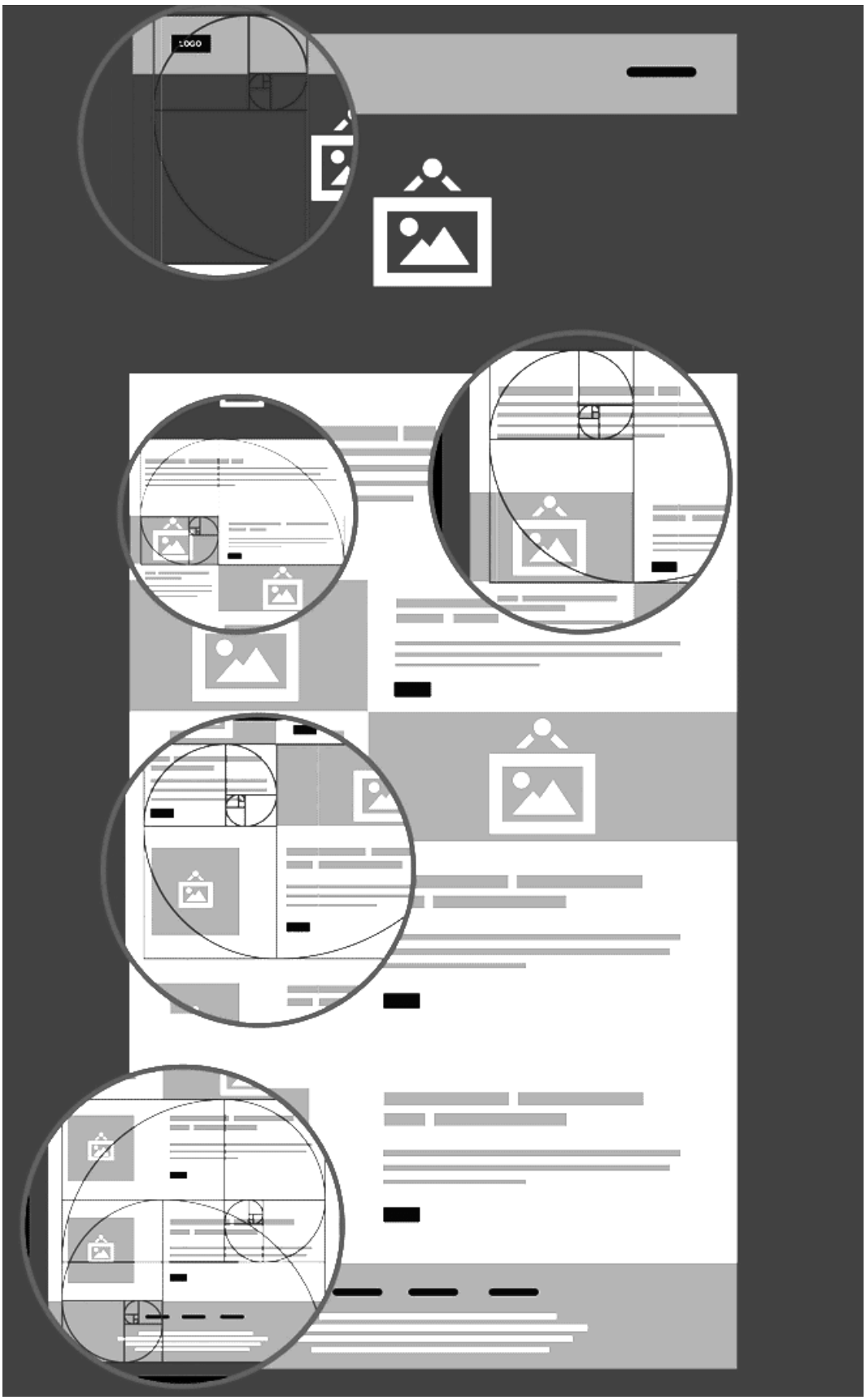


Рисунок 2.41 – Приклади використання золотого перетину у сітці веб-сторінки

2.8.2. Правило третин

Оскільки Золотий Перетин включає в себе деяку складну математику, згодом вона була спрощена Правилом Третин, і саме це правило було включено в більшість веб-проектів. Це надає сітку, яку дизайнер може накласти на розміщення своїх елементів. Сітка, сформована Правилом Третин, має 4 точки перетину. Емпіричне правило полягає в розміщенні найбільш важливих елементів на цих перетинах, тому що саме там око фокусується перш за все. Не всі перетини обов'язково будуть використовуватися. Відвідувачі зазвичай бачать в шаблоні «F», верхній лівий, ймовірно, найважливіший, а потім переміщається вниз до середини і поперек, а потім назад і вниз. Подумки проведіть на сторінці дві вертикальні і дві горизонтальні лінії, поділивши область на дев'ять рівних частин. Вважається, що більшість людей фокусують погляд саме на точках перетину цих ліній. Згадуйте про правило третин кожен раз, коли думаєте, куди розмістити, наприклад, кнопку із закликом до дії.



Рисунок 2.42 – Сітка, побудована за правилом третин



Рисунок 2.43 – Приклад веб-дизайну, в якому використовується правило третин, представлений в статті в журналі Smashing

У макеті використано два перетини для розташування інформації, на якій необхідно аби відвідувачі зосередилися – це мета / перевага (найголовніше). Сторінка збалансована. Це своєрідний естетичний компроміс, який створює

відчуття збалансованості на екрані, не перевантажуючи його, але і не роблячи надмірно статичним.

2.9. «Повітря»

Простір, або, як кажуть веб-дизайнери, «повітря» – дуже важливий композиційний елемент. Якщо його мало – займіться оптимізацією контенту і не піддавайтеся бажанням залишити якомога більше тексту, картинок і іконок. Тому що «повітря» іноді набагато важливіше, ніж контент. Людям подобаються «просторі», не переобтяжені інтерфейси – від них не рябить в очах, вони доносять чітко сформульовану думку і не дають шансу заплутатися. Подумайте, що ви хочете сказати світу, і обмежтеся парою-трійкою картинок і коротким текстом строго по справі.

2.10. Використання відступів

Відступи – це все порожні області, які оточують контент. Дуже важко їх ігнорувати, тому вони є всюди: прогалини навколо зображень, відступи між блоками тексту, прогалини між рядками тексту і навіть прогалини між літерами. Це все відступи, які можуть бути налаштовані.



Рисунок 2.44 – Оригінальний дизайн. Горизонтальні макро-відступи.



Рисунок 2.45 – Вертикальні макро-відступи. Мікро-відступи.

Ви можете використовувати відступи не тільки щоб зробити ваш дизайн більш привабливим, але також і для того, щоб поліпшити конверсію вашого сайту, досвід використання або для спонукання користувачів на якісь дії. Ось що ви повинні знати про використання відступів:

Додайте «повітря» Відступи визначають, наскільки ваш текст зручний до читання. Якщо ви створюєте сторінки, навантажені контентом, вони повинні виглядати дуже ясно, організовано і у людей не повинні викликати проблем з переглядом вашого сайту. Ви повинні використовувати відступи з розумом. Не існує золотого правила про те, як багато порожнього місця треба залишати

навколо певних елементів контенту. Вам потрібно знайти відступи, що зроблять ваш контент легким для читання без нагромаджень. З іншого боку, дизайн з великими пробілами може бути досить заплутаним, користувачу потрібно буде постійно прокручувати сторінку і ваш контент буде просто безцільно парити на сторінці. В цілому ви заподієте менше шкоди якщо ви будете використовувати занадто великі прогалини і дозволите свого контенту дихати, це завжди більш безпечний варіант, можете скористатися ним, якщо ви не до кінця впевнені.

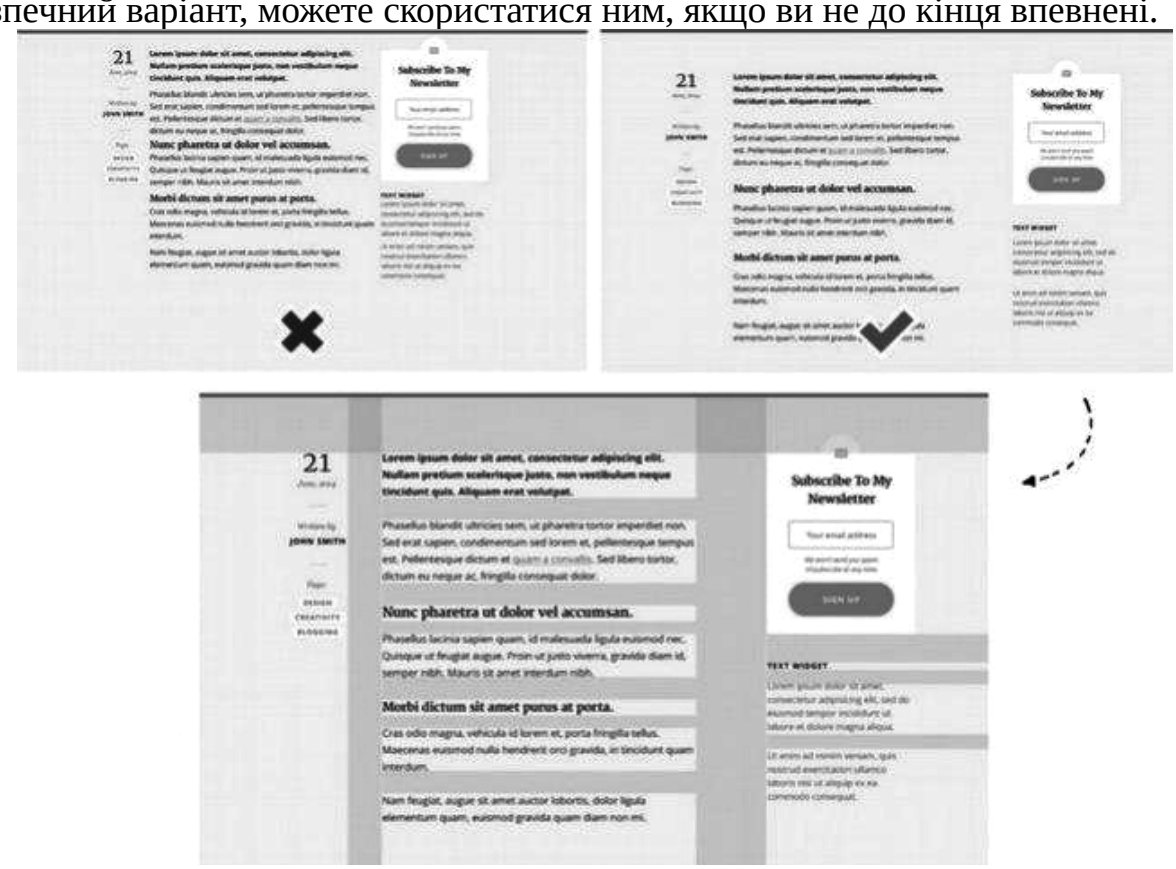


Рисунок 2.46 – Приклад використання відступів для організації контенту

Будьте одноманітними. Одноманітність в дизайні створює шаблони і визначає стиль дизайну.

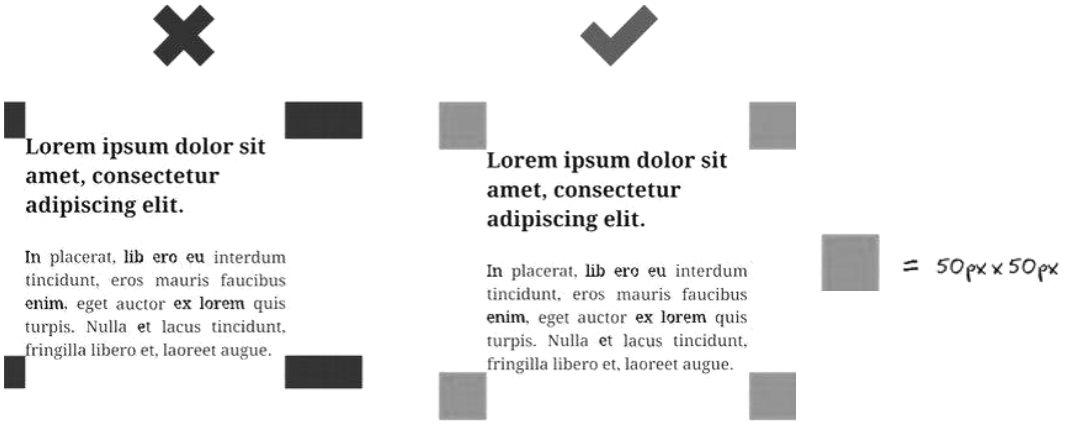


Рисунок 2.47 – Дотримуйтесь одноманітності та єдності у дизайні

Якщо ви одноманітні у використанні відступів протягом усього дизайну, ваш дизайн буде виглядати професійно і органічно. Припустимо, у вас є певний

блок з текстом всередині. Якщо ви вирішите додати відступ в 50 пікселів з правого боку, то тоді ви повинні будете слідувати цим шаблоном і додати ті ж самі 50 пікселів правої сторони: Так, якщо у вас є більше одного блоку з текстом всередині на тій же сторінці або на підсторінку, то вам потрібно підтримувати ті ж самі відступи в цих блоках.

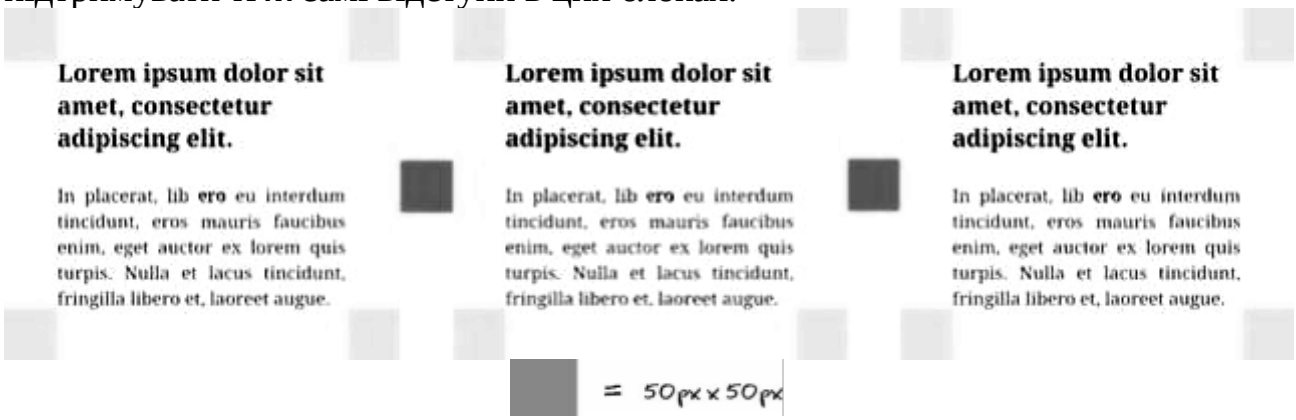


Рисунок 2.48 – Система інтервалів

Добре було б вибрати базовий розмір відступу при роботі над дизайном вашого макета. Потім можете використовувати його в якості стандартного пробілу між основними елементами і використовуйте пробіл в 2 рази більше або в два рази менше для більших або дрібних елементів. Це допоможе вам підтримати однаковість відступів у всьому дизайні.

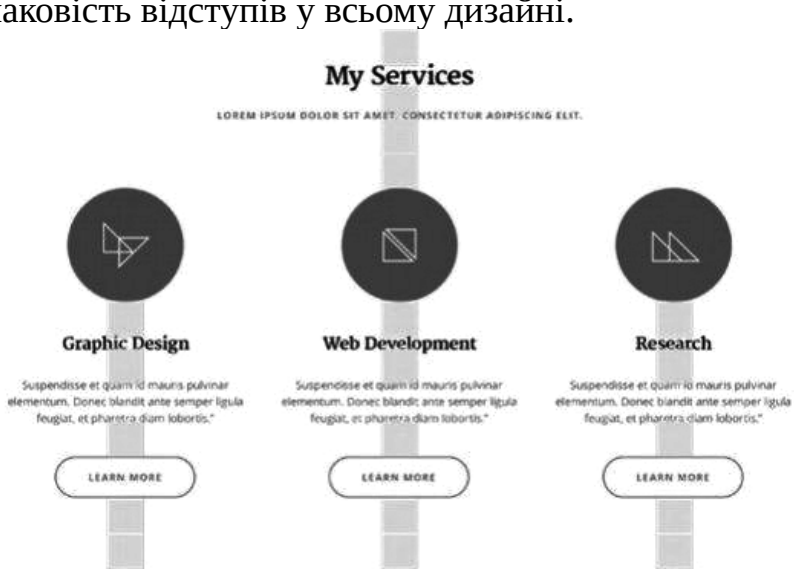


Рисунок 2.49 – Система інтервалів

Так у прикладі вище, квадратні блоки мають однаковий розмір. У окремих випадках їх помножено на 2 або 3, щоб досягти однаковості прогалин в дизайні макета.

Чим менше тим краще. Використовуйте відступи для спрощення вашого дизайну і додавання мінімалістичного характеру. Не намагайтеся заповнити кожен пробіл вашого дизайну контентом. Замість цього використовуйте пробіли для створення прихованого шляху, який проведе ваших користувачів через сторінку. Спробуйте створити непорушний лінійний потік від самого верху сторінки до низу.



Рисунок 2.50 – Приклад використання прогалин як прихованого шляху для користувача

2.11. Створюємо баланс в інтерфейсах

Суть візуального балансу – в балансуванні елементів відносно один одного, немов вони знаходяться на шальках терезів. Як врівноважити шальки терезів? Потрібно підібрати предмети однієї ваги – уява одразу малює відомі терези Феміди. У фізичної ваги і ваги "дизайнерської" багато спільного: фізичний об'єкт може бути дуже маленьким, але важким (незалежно від маси); елемент дизайну також може бути невеликим, але забирати на себе велику частку уваги користувача. Гравітацію фізичних тіл можна порівняти з *візуальної важливістю* елементів дизайну: кожен елемент дизайну притягує і відштовхує інші елементи, так само як гравітаційне поле фізичного тіла здатне притягувати і відштовхувати інші тіла. Коли справа стосується опрацювання *візуальної важливості* елементів дизайну, в гру вступають два чинники: *тип балансу і показники балансу***,** тобто *як збалансувати два або більше елементів між собою* і *як представити візуальні характеристики кожного елемента.*

2.12. Характеристики балансу

Кожен об'єкт в композиції має свої власні характеристики і унікальні особливості взаємодії з іншими об'єктами. Основні характеристики балансу це *розмір, колір, форма, значимість і розташування.* Кожну з цих характеристик можна використовувати по-окремо, якщо ви створюєте симетричний баланс. Якщо ж ви намагаєтеся збалансувати асиметричну композицію, вам потрібно буде враховувати їх все в сукупності: змінивши одну з характеристик, ви вплинете на інші.

2.12.1. Розмір

Порожній об'єкт великого розміру можна візуально врівноважити невеликим, але більш щільним елементом. Візьмемо, наприклад, блог: текстова область блогу зазвичай більше, ніж бічне меню.



Рисунок 2.51 – Приклад atlason.com

Щоб збалансувати ці два елементи, можна додати більше білого простору в текстову частину, а область бокового меню заповнити чимось *візуально більш важким.* А якщо ви домагаєтеся симетричного балансу в дизайні, потрібно робити елементи рівними за розміром, незалежно від візуального ваги.

2.12.2. Колір

Колір може зробити елемент композиції візуально важче або легше. Більш насичені і темні кольори сприймаються як більш важкі, а приглушені і світлі – як більш легкі. Таким чином, червоний колір важче, ніж жовтий, тому що він привертає більше зорової уваги.



Рисунок 2.52 – Приклад використання кольору для збалансування touchtech.co.nz

Якщо ви хочете збалансувати насичений, темний елемент декількома світлими, спробуйте «програти» з іншими характеристиками балансу або поекспериментувати з фоном, на якому розміщені елементи.

2.12.3. Форма

Форма елементів композиції може вплинути на баланс. Хвиляста, нерівна форма виглядає цікавіше – а значить "важить" більше – ніж проста, пряма форма. Тим не менш, не варто намагатися збалансувати одну складну форму іншого – це може збентежити користувачів. Краще збалансувати складний об'єкт простий, але більш рельєфною формою.



Рисунок 2.53 – Збалансування формою infinvision.com

2.12.4. Значимість

Контраст також може бути важливим фактором збалансованості дизайну. Чим *вище контрастність об'єкта, тим він важче* – і навпаки. Існує кілька способів встановлення балансу між об'єктами з різним ступенем контрастності; можна компенсувати низьку контрастність яскравим фоном або цікавою текстурою. А якщо потрібно знизити контрастність елемента, можна додати прості кордону, градієнти і тіні.



Рисунок 2.54 – Приклад galp.in

2.12.5. Розташування

Розташування – це важливий фактор. Саме за рахунок розташування ви зможете врівноважити елементи з різними характеристиками: кольором, формою, розміром і значущістю. Як ми вже говорили раніше, великий об'єкт важче маленького, але вага об'єктів також залежить і від їх віддаленості від центральної осі.



Рисунок 2.55 – Приклад розташування для досягнення балансу

Об'єкти, розташовані ближче до лінії центру, важать більше ніж ті, які знаходяться на відстані. Тобто якщо поруч з центральною віссю знаходиться великий червоний об'єкт, ви можете або посунути його ближче до краю, або перемістити інші об'єкти ближче до лінії центру.

2.13. Закони взаємодії

Закон Фіттса. Час, необхідний для швидкого переходу до цільової області, залежить від співвідношення відстані до цілі і ширини мети. Якщо вам потрібно далеко тягнутися до кнопки, потрібно зробити її більше.

Закон Хіка. Чим менше елементів, тим менше часу займає вибір одного з них.

Закон Якоба. Користувачі воліють, щоб ваш продукт працював так само, як і всі інші продукти, які вони вже знають.

Закон спрощення. Люди будуть сприймати і інтерпретувати двозначні або складні зображення як найпростішу форму, тому що саме інтерпретація вимагає найменшого пізнавального зусилля**.**

Закон Міллера. Короткочасна людська пам'ять, як правило, не може запам'ятати і повторити більш 5-9 елементів.

Бритва Оккама. Не слід залучати нові сутності без крайньої на те необхідності.

Закон Паркінсона. Зростання призводить до ускладненості, а ускладненість – це кінець шляху.

Закон крайньої позиції. Мозок найкраще запам'ятовує перший і останній елемент серії.

Ефект Ресторфф. Об'єкт, що виділяється із загального ряду схожості однорідних об'єктів, запам'ятовується краще за інших.

Ефект Зейгарник. Людина краще запам'ятовує перервані дії, ніж завершені.

Контрольні питання

1. Визначення поняття «композиція». Види композицій.
2. Закономірності зорового сприйняття форми і простору. Властивості сприйняття.
3. Зони активності і напрямки.
4. Художні засоби побудови композиції. Передача симетрії і асиметрії.

Теми для самостійного опрацювання

1. Метричний повтор. Ритм. Контраст. Нюанс.
2. Принцип домінанти. Виділення сюжетно-композиційного центру.
3. Закрита і відкрита композиція.
4. Статика і динаміка в композиції. Правила передачі руху, спокою.

Тестові завдання

1. Спіраль золотого перетину виникла в результаті

a) інтерпретації золотого перетину Стіва Джобса
b) інтерпретації золотого перетину Леонардо да Вінчі
c) інтерпретації золотого перетину Сократа
d) інтерпретації золотого перетину Ісаака Ньютона

2. ***Правило третин —***

a) покращений варіант золотого перетину
b) не має відношення до золотого перетину
c) спрощений варіант золотого перетину
d) жодного вірного варіанту

3. Сітка, сформована Правилом Третин, має

a) 5 точки перетину
b) 3 точки перетину
c) 4 точки перетину
d) 1 точку перетину

4. Правило Третин полягає в

a) розміщенні найбільш важливих елементів на перетинах, тому що саме там око фокусується перш за все
b) розміщенні найбільш важливих елементів в центральному прямокутнику, тому що саме там око фокусується перш за все
c) розміщенні найбільш важливих елементів в верхньому правому прямокутнику, тому що саме там око фокусується перш за все
d) розміщенні найбільш важливих елементів на діагоналях, тому що саме там око фокусується перш за все
e) розміщенні найбільш важливих елементів в першій третині робочого простору, тому що саме там око фокусується перш за все

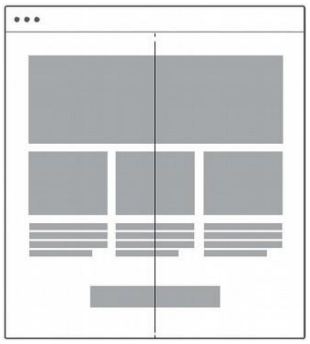
5. ***Засоби для вибудови візуальної ієрархії***

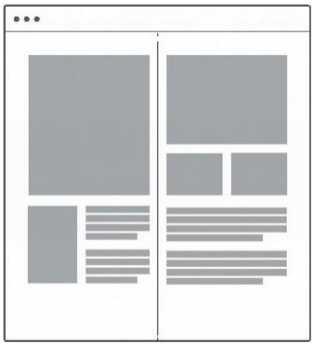
a) масштаб, колір, контраст, вирівнювання, «повітря»
b) масштаб, колір, контраст,вирівнювання
c) масштаб, колір, контраст,
d) масштаб, колір

6. ***Види балансу: (оберіть зайве)***

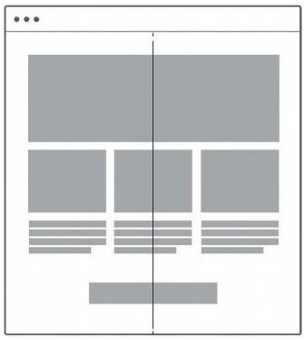
a) горизонтальний баланс та вертикальний баланс
b) дзеркальний баланс
c) радіальний баланс
d) симетричний баланс та асиметричний баланс

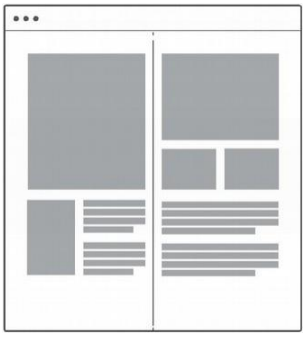
7. ***Симетричний баланс зображено***

a) 

b) 

8. **Асиметричний баланс зображено**

a) 

b) 

9. **Створення непорушного лінійного потоку від самого верху сторінки до низу називається**

a) «Шлях користувача»
b) «Терновий шлях»
c) «Чумацький шлях»
d) «Прихований шлях»

10. Схема траєкторії руху погляду людини на сторінці авторизації Facebook побудовано за принципом

a) «F-паттерн»
b) трикутник
c) зигзаг
d) «Z-макет»

Практичні завдання

Завдання 2.1. Провести передпроектний аналіз аналогів та прототипів;
Завдання 2.2. Створювати документи на розробку UI/UX дизайну.

3. ПРОТОТИПУВАННЯ ЯК СПОСІБ ПЕРЕДПРОЕКТНОГО ДОСЛІДЖЕННЯ.

План:

3.1. Мозковий штурм і пошук ідей
3.2. Пошук натхнення
3.3. Процес створення дизайну
3.4. Планування проекту
  3.4.1. Опис проекту
  3.4.2. Мета проекту
  3.4.3. Цільова аудиторія
  3.4.4. Очікувані результати
3.5. Прототипи як засіб передпроектного дослідження
  3.5.1. Скетч (Scetch)
  3.5.2. Вайрфрейм (Wireframe)
  3.5.3. Прототип або макет (Prototype)
  3.5.4. Мокап (Moqup)
3.6. Логічна карта сайту.

В індивідуальних макетах сторінок можна бути більш креативним і показати створені шаблони і скористатися мовою керівництва по стилю. Керівництво по стилям дозволяє нам бути трохи більш гнучкими в дизайні, так що ми можемо просто скористатися ним, навіть коли у нас немає готового дизайну сторінки.

Першим кроком, що передує безпосередньо прототипуванню є пошук ідей. Найефективнішим у пошуку ідей та генерації креативних рішень є метод мозкового штурму. розглянемо детальніше, яким чином його досягти та застосовувати.

3.1. Мозковий штурм і пошук ідей

Перш ніж ви почнете мозковий штурм, переконайтеся, що ви закінчили вступну частину вашого проекту і визначилися з його цілями. Так ви будете знати, в якому напрямку вам варто рухатися.

Мозковий штурм – це процес креативного мислення, пошуку та обміну ідеями. Він повинен займати більшу частину вашої роботи, незалежно від того чи працюєте ви один або в команді.

Перемикання вашого мозку в режим креативного мислення допоможе вам сфокусуватися на темі і придумати цікаві ідеї. Це не просто. Ми не роботи, і дуже часто ви будете потрапляти в ситуацію, коли ви дуже сильно намагаєтеся придумати щось вартісне. Це нормально, не треба занадто сильно старатися.

Поглянемо на те, як зробити цей процес більш простим і ефективним:

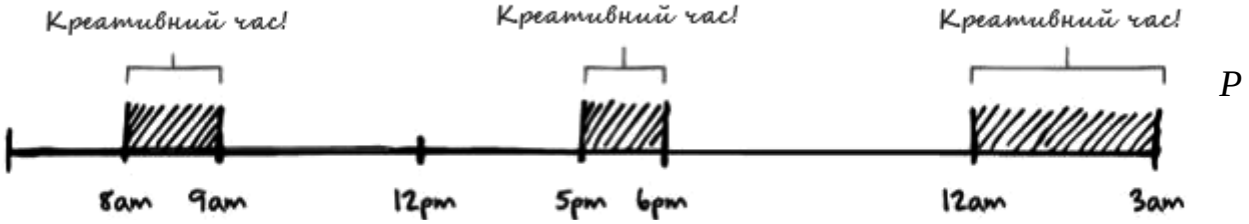
1) **Знайдіть свій найбільш креативний час.**

Людський мозок любить шаблони. Ми всі різні і у нас у всіх різні звички, але у нашому житті є певні шаблони. Наш мозок зазвичай знає, коли нам варто прокинутися, коли настав час йти спати, і коли ми готові до роботи. Так що,

наприклад, не чекайте, що ви будете креативними в момент, коли мозок говорить вам, що пора йти спати.

Постарайтеся знайти час протягом дня, коли ви як можна більш креативні. Спробуйте проводити мозковий штурм протягом перших пари робочих годин. Це дасть вам величезну перевагу. Звичайно ж це не означає, що ви можете бути продуктивними лише в перебігу декількох годин в день.

Креативний час використовується для мозкового штурму і пошуку хороших ідей. Потім ви можете витратити решту дня працюючи над цими ідеями і виконуючи продуктивну роботу.



Р

исунок 3.1 – Приклад пошуку креативного часу у щоденному таймлайні

На жаль протягом дня є не так багато креативних годин.

2) Не засуджуйте ідеї.

Це один з основних принципів мозкового штурму. Немає поганих ідей на даному етапі. Постарайтеся придумати так багато ідей, як тільки можете і не засуджуйте їх. Деякі ідеї можуть виглядати погано або шалено, але вони можуть привести вас до інших відмінних ідей. Якщо в мозковому штурмі бере участь кілька людей, то правило "не засуджуйте" допоможе всім відкрити свідомість і бути більш креативним.

3) Встановіть часові рамки і перерви.

Встановіть часові рамки для сесій мозкового штурму. Дуже просто збитися зі шляху і втратити напрямок роздумів. Вам не варто йти досить далеко від роздуму над оригінальними припущеннями і цілями проекту.

4) Не треба дуже старатися.

На жаль не кожна сесія мозкового штурму успішна. Якщо ви розумієте, що ви занадто сильно намагаєтеся видавити з себе ідею, просто зупиніть сесію і поверніться до цього питання трохи пізніше або на наступний день.

5) Зберігайте усе.

Записуйте, замальовувати і зберігайте все, що ви робите протягом сесії мозкового штурму. Є дуже багато додатків і он-лайн інструментів, які можуть допомогти вам створити вашу карту думок або створити швидкі замітки. Так чи інакше може бути використовувати тільки ручку з папером для запису та замальовки різних ідей.

6) Натхнення.

Звичайно ж, натхнення це фундаментальна частина сесії мозкового штурму. До речі, кожна наша думка або ідея з'явилася в нашій голові завдяки тому, що ми щось бачили або читали про щось. Все це зберігається в нашій пам'яті, і наш мозок створює вірні зв'язки між цими думками під час процесу створення ідей. Не бійтеся шукати натхнення в інший дизайнерській роботі.

7) Діліться своїми ідеями.

Останнє, але не менш важливе: діліться своїми ідеями з іншими. Покажіть людям те, над чим ви працюєте, ваші ідеї або концепти дизайну. Зберіть відгуки і поліпшите свою роботу.

8) Замальовувати свої ідеї

Замальовка своїх дизайнерських ідей – це процес візуалізації ваших думок з використанням простих малюнків і записів. Якщо ви раніше ніколи не замальовували свої ідеї, можливо вам буде важко це робити на початку, але чим більше ви будете практикуватися, тим простіше вам буде робити це.

Існує дуже багато дизайнерів, які ніколи не замальовували свої ідеї. Вони завжди пропускали цей крок. Так, ви можете створювати дизайн веб-сайтів без начерків. У будь-якому завдяки начерками ви можете бути більш креативними, і ви можете швидше почати використовувати свої ідеї. Ви повинні зробити невеликі прототипи інтерфейсів перш ніж перейти до реальної роботи в графічному редакторі. Таким чином, ви зфокусуєтесь на функціональності вашого контенту, перш ніж ви почнете додавати будь-які стилі, покращуючи зовнішній вигляд вашого сайту.

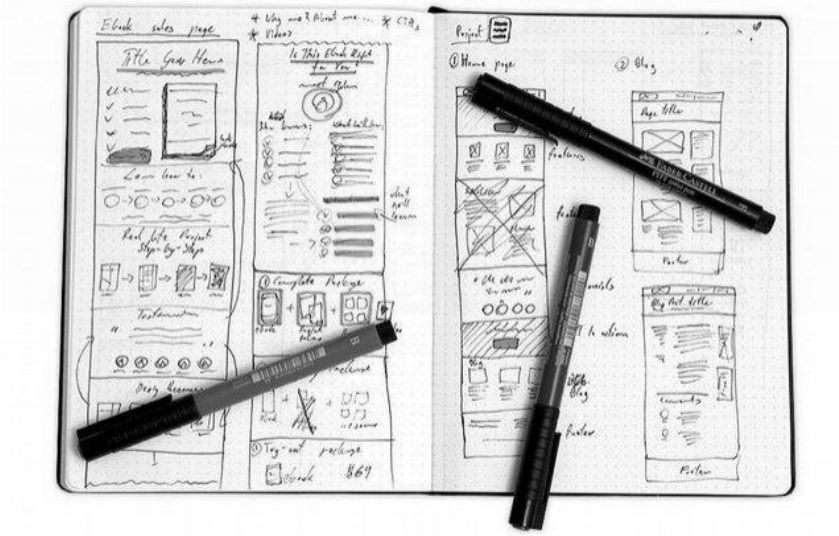


Рисунок 3.2 – Скетчбук

Є декілька правил, які вам варто пам'ятати під час роботи над скетчами (начерками):

• Не робіть їх ідеальними.

Ви не повинні бути художником, щоб робити начерки, ви не повинні їх показувати всім. Малюйте максимально швидко і сфокусуйтеся на концепті, а не на деталях.

• Додавайте нотатки та коментарі.

Деякі начерки можуть бути грубими, так що не забувайте додавати анотації і описувати свої думки. Це допоможе вам зрозуміти ваші начерки дизайну в майбутньому.

• Підтримуйте організованість.

Збирайте начерки для конкретного проекту і зберігайте їх у одному місці.

Найкраще в начерках те, що вам нічого не потрібно для того, щоб замалювати ваші ідеї. У вас може бути ручка і листок паперу. Це все що вам необхідно.

Звичайно, якщо ви хочете зробити начерки частиною процесу створення дизайну, ви швидше за все захочете придбати інструменти більш високої якості, щоб все було дотримано в одному стилі або навіть захочете мати шаблони з макетом мобільного телефону або шаблоном браузера. Варіанти нескінченні і ви знайдете свій власний спосіб, але у мене є кілька рекомендацій з чого ви можете почати:

Почніть користуватися ручками тому що їх неможливо стерти, адже не існує поганих ідей в вашому блокноті з начерками. Жодна з них не заслуговує бути стертою і забутою. Зберігайте всі ідеї, тому що ви ніколи не знаєте, коли одна з них може вам стати в нагоді.

Ви можете малювати начерки, використовуючи різні кольори, такі як, наприклад червоний або зелений для акцентування уваги або виділення важливих областей.

Вам необхідний хоча б один невеликий блокнот для нарисів, який ви можете завжди носити з собою для замальовки різних ідей. Використовуйте його для запису випадкових ідей. Просто пам'ятайте, що необхідно завжди додавати дату і невеликий заголовок для кожної ідеї, так вам буде простіше послатися на них в майбутньому.

3.2. Пошук натхнення

Майже неможливо створити щось з нічого. Не має значення наскільки ви креативні, ваш креатив в будь-якому випадку буде заснований на вашому життєвому досвіді і речах, які ви бачите в навколишньому вас світі.

Звичайно ж, існує тонка грань між натхненням і копіюванням. Так чи інакше, ви повинні бути готовими оцінити свої власні ідеї і вирішити чи можна їх застосовувати в різних ситуаціях.

Кращий спосіб переконатися, що ти не копіюєш чиюсь роботу – це ніколи не фокусуватися тільки на одному конкретному джерелі натхнення. Користуватися різними ресурсами. Проведіть невелике дослідження он-лайн і офф-лайн і об'єднайте всі ідеї. Так само щодня переглядайте роботи інших дизайнерів і зберігати собі те, що сподобалося. Ви можете повернутися назад в свою колекцію улюблених дизайнів перед початком роботи над новим проектом. Пам'ятайте, що ви використовуєте все це заради натхнення і генерації ідей. Коли ви готові розпочати роботу над дизайном, відкладіть в сторону всі ці ресурси і дозвольте вашій уяві почати роботу. Втілюйте ідеї в тому вигляді, в якому ви їх запам'ятали, а не так як ви їх бачили. Це допоможе вам не обмежувати себе, замість цього ви почнете застосовувати свій власний стиль і ідеї, які у вас є. Дозвольте вашому мозку обробити натхнення своїм унікальним способом.

3.3. Процес створення дизайну

У деяких проектах з розробки веб-дизайну працює велика кількість людей, в деяких тільки один. Саме проекти, в яких одна людина є і менеджером,

і програмістом, і дизайнером, є найскладнішими. Процес створення дизайну, у різних проектів з різними цілями і вимогами, абсолютно однаковий.

Тому дуже важливо, аби процес розробки дизайну не був хаотичним, аби ви не відволікалися на випадкові ідеї і натхнення, а відразу на ті, що ставали основою проекту. Гарні навички лише у розробці веб-дизайну та його програмуванні ніщо без вірного робочого процесу, на формування якого інколи витрачається до декількох років. Тому розглянемо приклад хаотичного та не спланованого ведення проекту з розробки веб-дизайну:

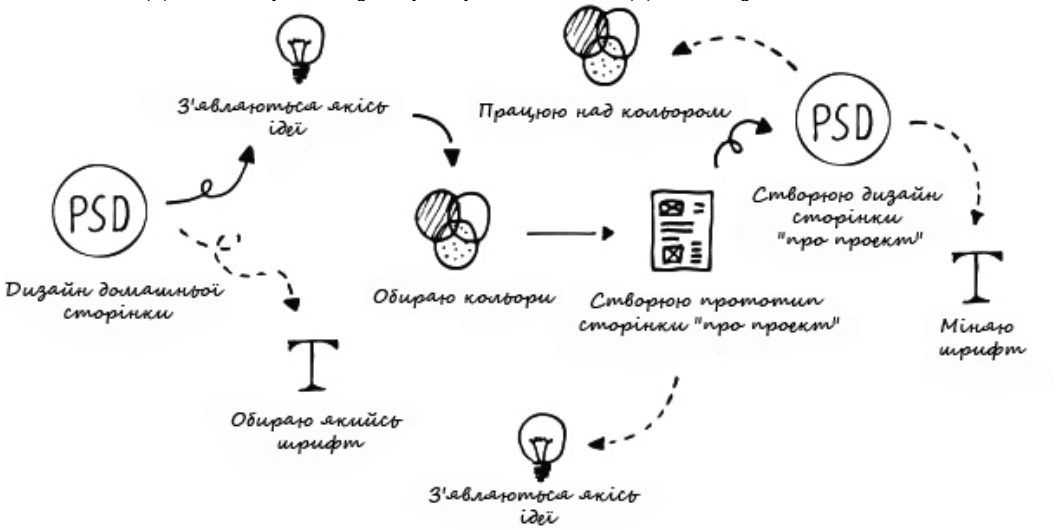


Рисунок 3.3 – Приклад хаотичного та не спланованого ведення проекту

Такий підхід споживає величезну кількість енергії, і кінцевий дизайн найчастіше є досить суперечливим і ніколи не має чіткої структури.

Хороший процес проектування дизайну допоможе вам рухатися в правильному напрямку, стати більш креативним, допоможе вам знайти найкращі ідеї, отримати більше замовлень від ваших клієнтів і навіть дозволить вам робити вашу роботу швидше і більш ефективно. Зараз розглянемо приклад структурованого процесу розробки веб-дизайну, від ідеї і натхнення до створення кінцевого проекту дизайну веб-сайту:

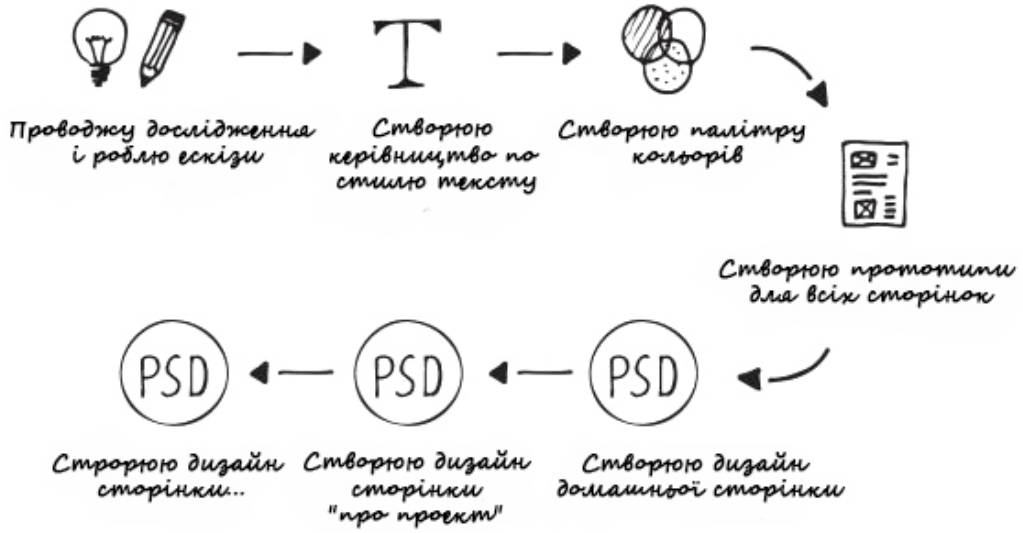


Рисунок 3.4 – Приклад структурованого процесу розробки дизайну

Цей підхід не бездоганний, проте, це найбільш ефективний спосіб створення веб-сайтів і такий підхід дає відмінні результати. Цей приклад процесу є до основою для вашого подальшого пошуку власного алгоритму дій для його покращення власними методами. Процес створення веб-сайту може бути досить складним, над його роботою може працювати декілька чоловік (UX-Дизайнер, візуальний дизайнер, верстальник) може вимагатися величезна кількість досліджень, якщо ви працюєте над великим проектом з величезним бюджетом.

Ми розглянемо процес створення веб-дизайну невеликих проектів, в яких ви є єдиним виконавцем і несете відповідальність за всі аспекти розробки.

Одним з найпоширеніших методів організації проектної роботи є прогресивний процес розробки, в якому кожен крок буде відділений різними віхами. Це дозволить вам зупинитися в будь-який момент, ви зможете дати оцінку вашій роботі або отримати погодження від клієнта. Такий процес вимагає створення керівництва по стилю перш ніж перейти до створення дизайну інтерфейсу веб-сайту.

Так ви будете точно знати, які шрифти та кольори вам варто використовувати у вашому проекті і в створенні файлів вашого дизайну.

Звичайно, іноді вам треба буде повертатися назад на крок і коригувати або оновлювати щось, але керівництво по стилю допоможе вам тримати все в одному стилі. Розглянемо такий процес створення дизайну:

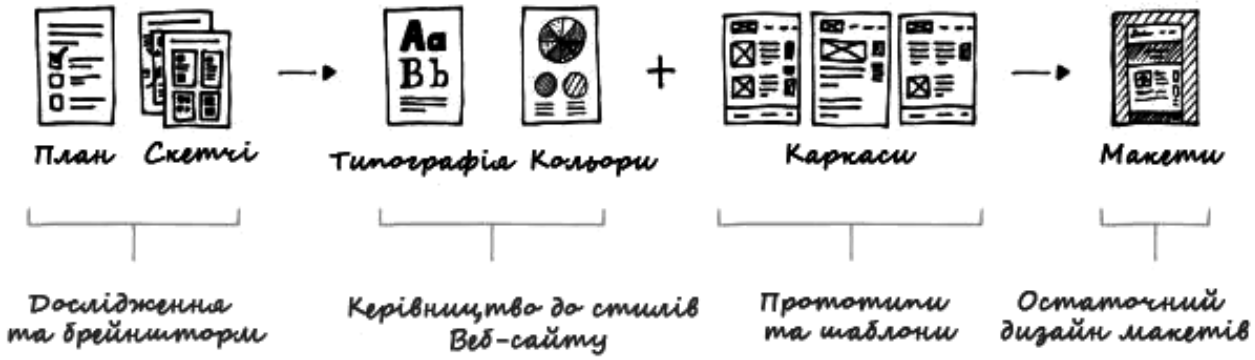


Рисунок 3.5 – Планування прогресивного процесу

3.4. Планування проекту

Кожен проект по створенню дизайну вимагає планування. Це перший і найважливіший крок, тобто в цей момент необхідно прийняти найважливіші рішення, що стосуються всього проекту. Вам потрібно запам'ятати, що дизайн це набагато більше, ніж те, що ви бачите. Тут є над чим подумати і що планувати, перш ніж ви відкриєте графічний редактор.

Підготовка хорошого опису проекту, установка основної мети і планування процесу, за допомогою якого ця мета буде досягнута, допоможе вам зберегти багато часу і допоможе уникнути проблем надалі. Необхідно розуміти, що будь-яка робота починається з розробки докладного технічного завдання на розробку дизайну користувацького інтерфейсу чи то веб-дизайну.

Під час процесу планування ми скористаємося наступними базовими елементами:

- Опис проекту.
- Мета проекту.
- Цільова аудиторія;
- Очікувані результати;
- Управління проектами;
- Карта сайту.

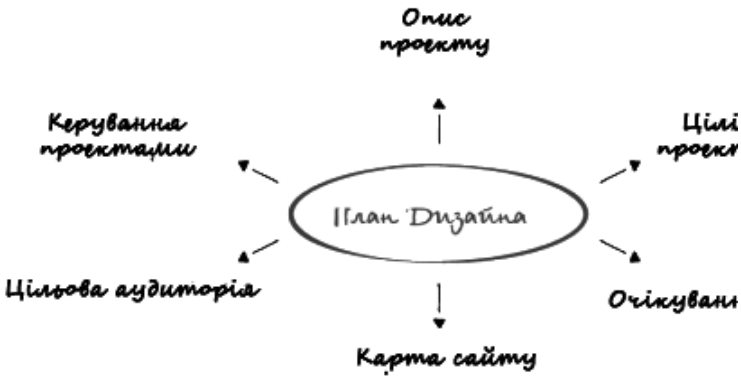


Рисунок 3.6 – План дизайну

3.4.1. Опис проекту

Ви повинні почати написання опису проекту з відповіді на питання: «Що ми створюємо?». Це допоможе вам сфокусуватися на тому, що саме з себе представляє ваш проект. Це інтернет магазин? Кав'ярня? Шаблон блогу? Дайте відповідь на це запитання парою пропозицій. Це допоможе вам очистити ваш розум і визначити тип проекту.

Це дійсно допомагає, коли ви запрошуєте в проект когось ще. Таким способом ви можете дуже просто описати новий проект для інших людей. Коли ви пояснюєте і описуєте, будь- яку ідею іншій людині, то ви дивитеся на неї з іншої перспективи. Це змушує вас думати старанніше і перевіряти ще раз чи має ідея сенс.

Чим менше технічно підкований людина, тим краще. Так, ваші ідеї повинні бути дуже простими, так що у вас не буде проблем в поясненні їх кому б то не було.

3.4.2. Мета проекту

Визначення мети проекту – це фундаментальна частина процесу створення дизайну. Як і в будь-якій поїздці, ми повинні знати пункт призначення. З іншого боку, ми могли б їздити по різних напрямках, але так ми могли б легко загубитися. Встановлюючи зрозумілі цілі для проекту, ми знаємо, куди рухатися і коли проект буде готовий. Набагато простіше буде оцінити роботу і зрозуміти чи зробили ми роботу добре, задавши одне просте запитання: «Чи виконали ми всі поставлені цілі?». Ми повинні бути сфокусовані на головних цілях, будь то підвищення кількості продаж інтернет магазину при його редизайні, чи то створення повністю нового бренду у процесі створення сайту-візитки. І якщо ми досягнемо такого результату, то наші покупці будуть вважати наш продукт дуже цінним. Ґрунтуючись на описі проекту, ми можемо визначити наші основні цілі (на прикладі дизайну сайту-портфоліо для фрилансера):

- Створити тему веб-сайту, яка допоможе фрілансерам і консультантам укладати більше угод.

- Додати функції дизайну, які відобразять список сервісів, портфоліо, покажуть клієнтам відгуки / рекомендації, створити блог (підняти авторитет) і залишатися на зв'язку.
- Створити простий, мінімалістичний і естетичний дизайн.

Ось такі три прості цілі описують основний процес створення дизайну, визначають основну функціональність, яку нам потрібно буде реалізувати, і показують нам те, як ми повинні ставитися до зовнішнього вигляду дизайну, що розроблюємо.

3.4.3. Цільова аудиторія

Цільовою аудиторією є люди, які будуть відвідувати веб-сайт вашого клієнта.

Так, хто ваші клієнти? Чи можете ви описати стандартного клієнта? Здебільшого це чоловіки чи жінки? Який у них віковий склад? Ці та багато інших питань можуть бути використані при створенні профілю (людини) стандартного клієнта / користувача. Замовником сайту-портфоліо можуть стати індивідуальні працівники, такі як копірайтери, розробники, веб-дизайнери, тренери, маркетологи, які купують їх послуги консалтингу. Це підходить і для чоловіків і для жінок, і вікова група може бути 16+. Один загальний елемент, який є у всіх цих людей, це те, що вони вирішили почати свій невеликий особистий бізнес.

Відвідувачів сайту вашого клієнта буде досить складно визначити тому це може бути хто завгодно. Це завдання може бути простіше, якщо ви зменшите групу ваших користувачів до однієї групи фрілансерів, наприклад до веб-розробників. Звідси ви дізнаєтеся, що більшість відвідувачів сайту вашого клієнта скоріше за все є власниками бізнесу або якогось дизайнерського агентства, яке шукає розробника. Створення профілів і проведення пошуку користувачів є досить складним завданням, над якою може працювати ціла команда дизайнерів.

3.5. Прототипи

Найдорожчі помилки – це помилки, допущені на початку проекту, так як саме на початкових етапах приймаються найважливіші рішення. У міру розвитку проекту, вартість помилки знижується. У той же час, початковий рівень контролю якості є мінімальним і підвищується в міру просування процесу розробки. Повноцінне тестування починається тільки на пізніх етапах. Результатом часто є занадто пізнє виявлення найдорожчих помилок і подальша дорога переробка системи або її окремих частин. На перший погляд може здатися, що деякі проекти однакові за своєю суттю та змістом. Скажімо, типовим проектом для вас є інтернет-магазин на базі існуючої у вас системи. Але індивідуальність кожного проекту закладена навіть у самому його визначенні.

На сучасному етапі розвитку сфери розробки програмного забезпечення є проблема змішування термінів як синонімічних, а саме поняття «*скетч*», «*вайрфрейм*», «*прототип*» та «*мокап*», визначаючи їх як виконаний з сіруватих

блоків начерк конкретної ідеї. У зв’язку з цим конкретизуємо визначення термінів далі.

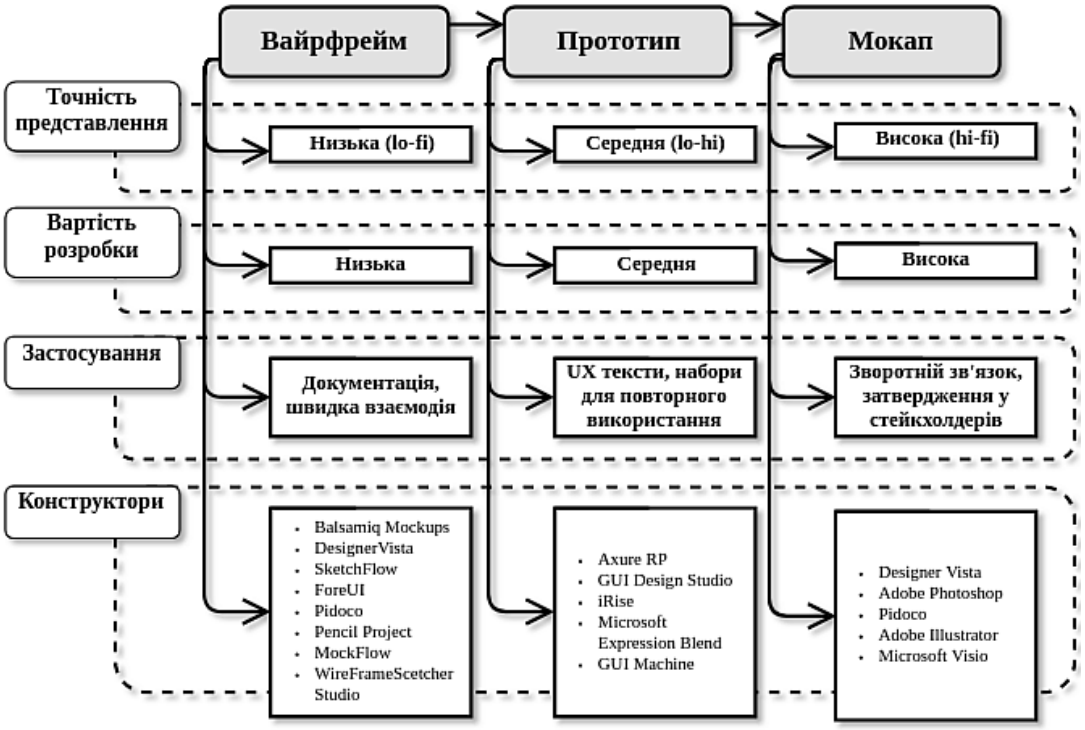


Рисунок 3.7 – Порівняння сутності понять вайрфрейм, прототип та мокап

Прототипування це не панацея, але спосіб уникнути деяких дорогих помилок, знизити ризики. Існує думка про те, що розробка однотипних проектів може проходити без етапу проектування. Але очевидно, що користувачі інтернет-магазину з продажу добрива та сільськогосподарської техніки будуть сильно відрізнятися від користувачів магазину з продажу одягу і гламурних аксесуарів, як і буде відрізнятися їх поведінку на сайті, об'єкти уваги тощо. Чи здатне ваше стандартне рішення врахувати відмінності користувачів? Стандартне рішення – це набір компонентів, відтестованих і випробуваних в різних проектах. Розуміння необхідності їх адаптації (кастомізації) часто виникає в кінці, коли відбувається повноцінне тестування, а вільних ресурсів вже немає. Це і є та сама дорога помилка (перевитрата проектного часу, що призводить до авралу, погіршення якості). Тому першим етапом під час процесу проектування будь-якого програмного забезпечення повинно бути створення прототипу, що передує програмуванню інтерфейсу програмного забезпечення. Прототипування є виключно цінним способом створення першої візуалізації проектів для демонстрації продукту, особливо на ранніх етапах тестування зручності застосування.

Процес прототипування програмного забезпечення передбачає вирішення ряду задач, зокрема: зниження витрат на проектування, скорочення термінів проектування, покращення якості пропонованих рішень, забезпечення нескладного в освоєнні та використанні програмного забезпечення, вивчення та впровадження нових технологій та засобів, досягнення кращих результатів в порівнянні з конкурентами. Задоволеність користувача програмним продуктом

або зручністю його використання в значній мірі визначається інтерфейсом програмного забезпечення.

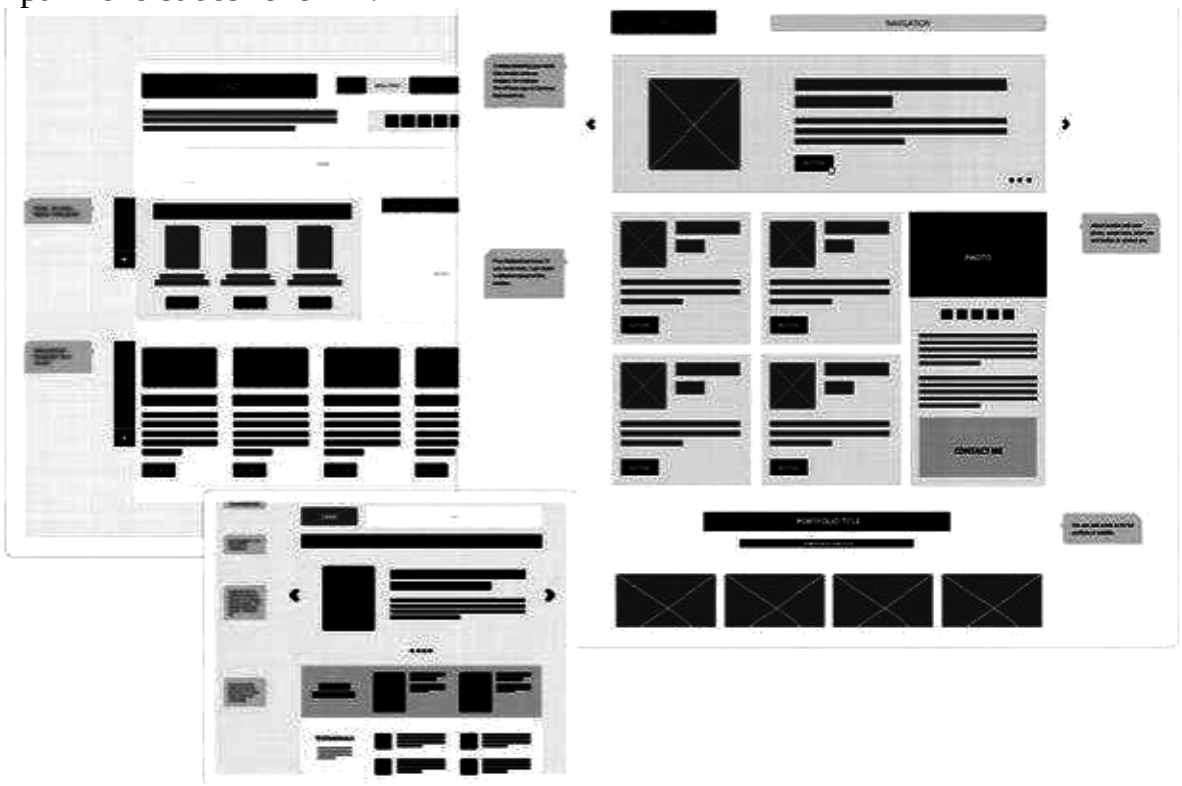


Рисунок 3.8 – Вайрфрейми та коментарі

Мета прототипування полягає у визначенні, наскільки те чи інше рішення перспективно, із подальшою реалізацією одного з найкращих варіантів. Цей підхід дозволяє заощадити час і ресурси, що витрачаються на проектування та розробку. Стандартний прототип відображає найважливіші елементи веб-сайту, такі як навігація, області контенту, положення логотипу, кнопки і якісь відмінні риси. Він використовується для вибору напрямку майбутнього дизайну і допомагає планувати макет всієї сторінки.

Під час процесу розробки програмного забезпечення чільне місце займає етап прототипування, як інструмент для уточнення вимог замовника та швидкої розробки. Розробка прототипів вже давно є невід'ємною частиною у багатьох сферах суспільного життя, зокрема, у галузі архітектурного і виробничого проектування.

В даний час існує безліч засобів прототипування, що вирішують ці проблеми та покликані знижувати марні витрати часу та коштів, але низький відсоток успішності програмних проектів говорить про недостатність обґрунтування ефективності застосування цих засобів в розробці програмного забезпечення.

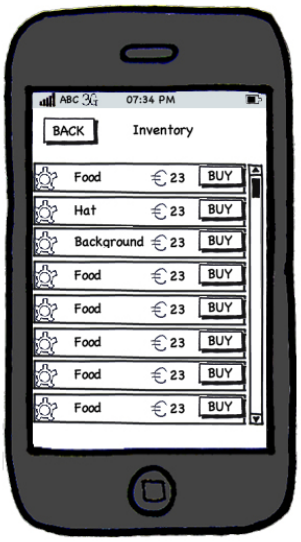
Безсумнівно, прототипування — додатковий обсяг роботи, проте, це є технологією, що дозволяє заощадити як час, так і зменшити витрати. Прототип корисний тим, що надає можливість оцінити логіку роботи програмного забезпечення в цілому на будь-якому етапі реалізації проекту без прив'язки до дизайну.

Мета прототипування полягає у визначенні, наскільки те чи інше рішення перспективно, і його подальша реалізація одна з найкращих. Цей підхід дозволяє заощадити час і ресурси, що витрачається на проектування та розробку.

3.5.1. Скетч (Scetch)

Розпочнемо з розгляду скетчу. Один із способів полягає в тому, щоб замальовувати його у блокноті або створювати з паперу:





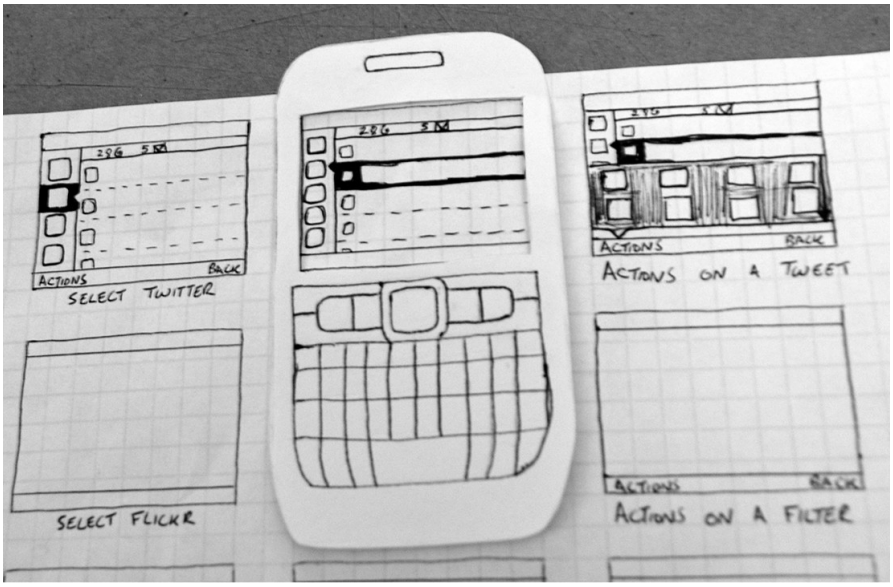


Рисунок 3.9 – Варіанти опрацювання користувацького інтерфейсу

Тож розглянемо принцип опрацювання ідеї у формі скетчу. Ви закінчили ваші пошуки, досліджували аналоги та прототипи і надихнулися їх дизайном. У вас є уявлення про те, як повинен виглядати дизайн, але у вас в голові так само є дуже багато різних ідей. Вам потрібно їх впорядкувати, взяти найкраще з них і рухатися далі. Починати необхідно з створення начерку з планування розташування контенту веб-сайту. Знаючи, які підсторінки будуть, потрібно їх правильно розмістити.

Розпочинати необхідно з замальовки декількох варіантів домашньої сторінки. Це всього лише макет домашньої сторінки. Трохи пізніше

представимо кілька варіантів кожного елемента. Відзначимо червоним основні кнопки, які будуть вести на інші підсторінки.

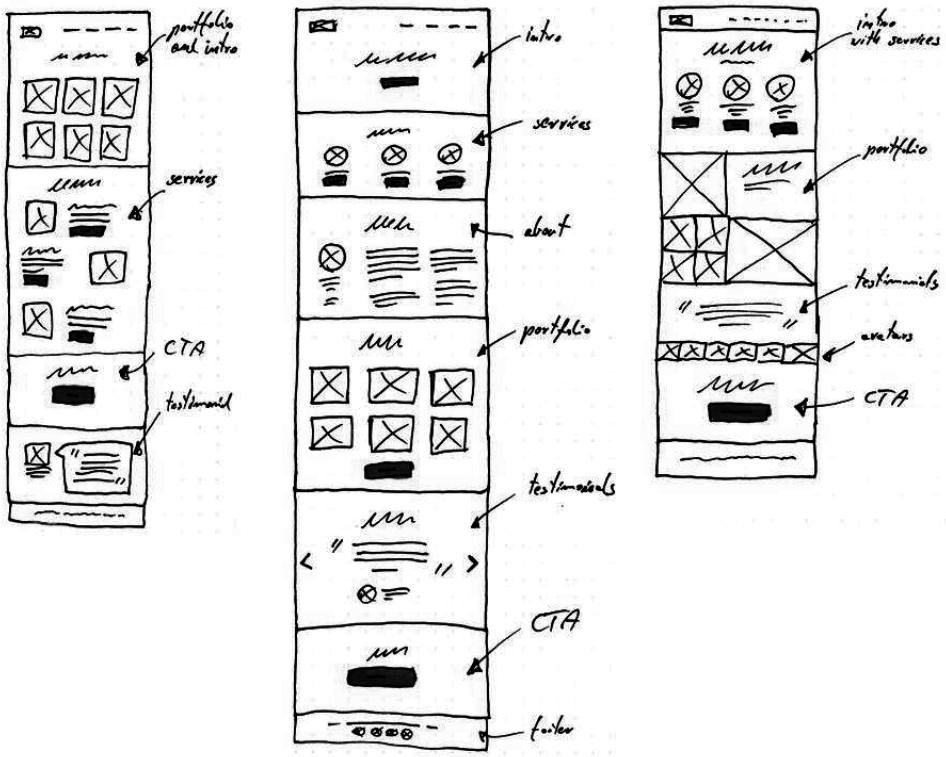


Рисунок 3.10 – Пошукові ескізи

Обираємо найбільш влучний скетч головного екранах майбутнього продукту, та доопрацьовуємо цей концепт. Намалюю невелику блок-схему, яка ілюструє, куди будуть вести різні розділи, ґрунтуючись на карті сайту.

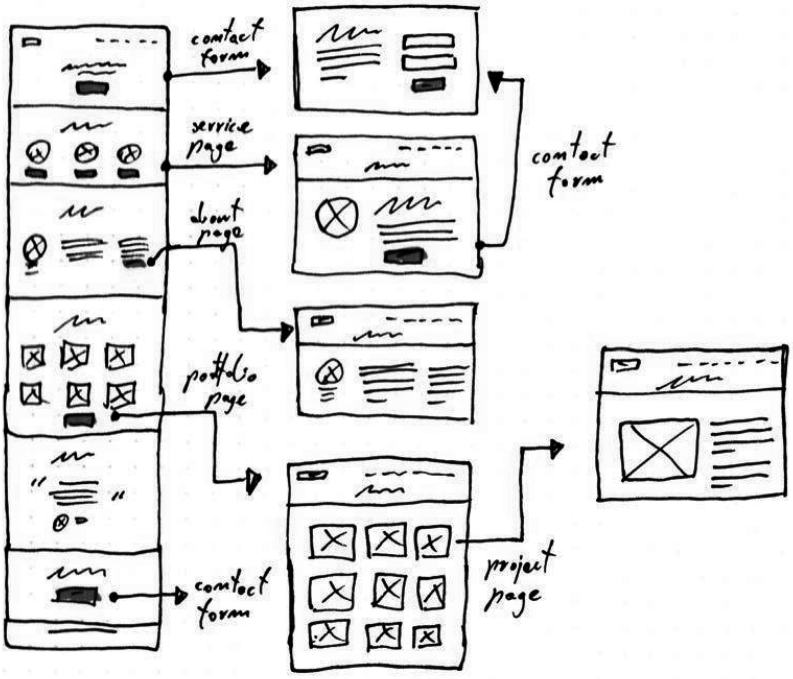


Рисунок 3.11 – Формування концепту на основі пошукових ескізів

У цьому концепті форма зворотнього зв'язку буде спливати на екрані, так що з нею легко буде працювати, і не буде витрачатися час на завантаження

іншої сторінки. Пам'ятайте, на даному етапі немає нічого хорошого і поганого. Замальовувати свої ідеї, не стирайте нічого, продовжуйте. Далі опрацьовуємо начерк для наступного розділу. Так само опрацьовуємо декілька начерків та обираємо найбільш вдалий:

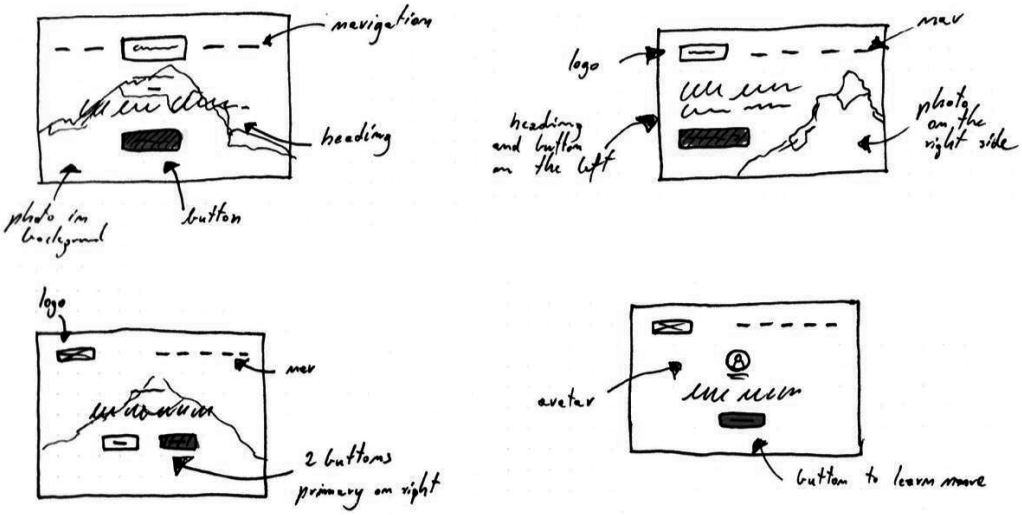


Рисунок 3.12 – Пошукові ескізи спливаючого вікна з коментарями

Тепер перейдемо до промальовування інших розділів домашньої сторінки.

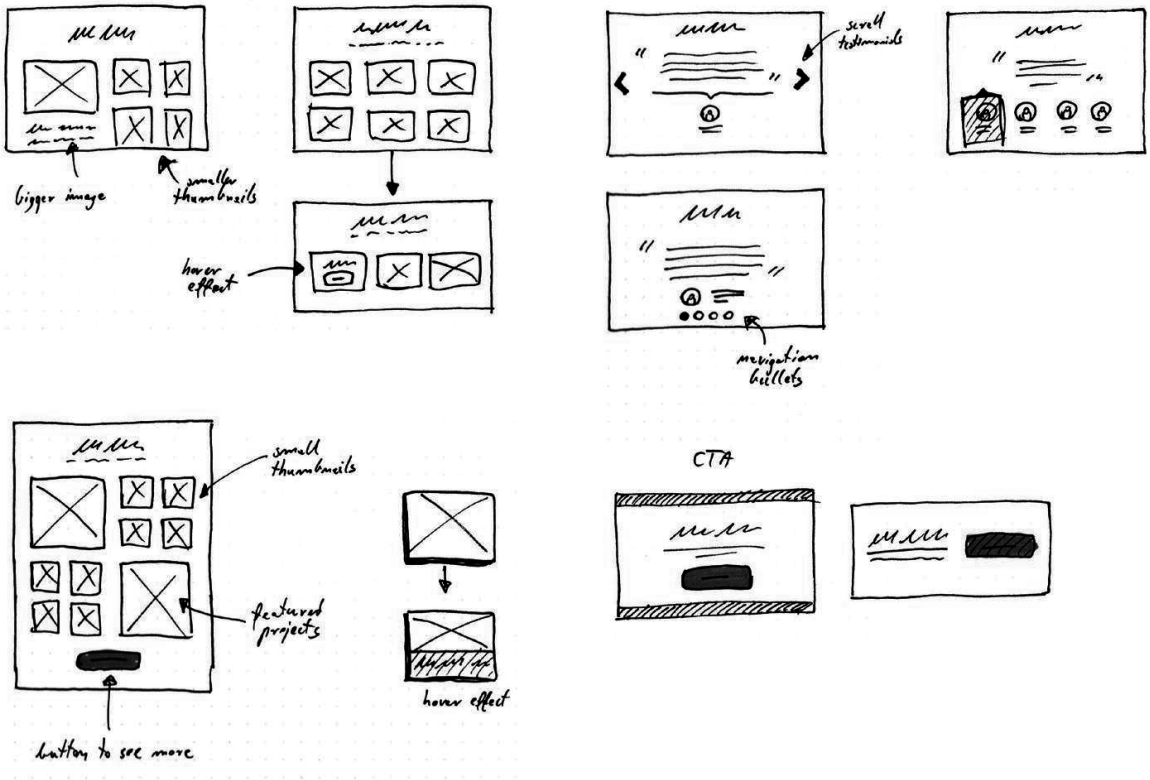


Рисунок 3.13 – Пошукові ескізи для інших розділів ресурсу

Обов’язковим елементом повинна бути типова контентна сторінка, що міститиме «контент + сайдбар», де необхідно продемонструвати приклад виведення текстового блоку великого розміру, з демонстрацією відображення таких типових елементів, як заголовок, підзаголовок, дату записи и автора записи, маркерований та нумерований списки, приклад відображення цитати та

демонстрація розташування у тексті супроводжуючої ілюстрації. Також зобразимо декілька різних екранів того, як буде виглядати запис.

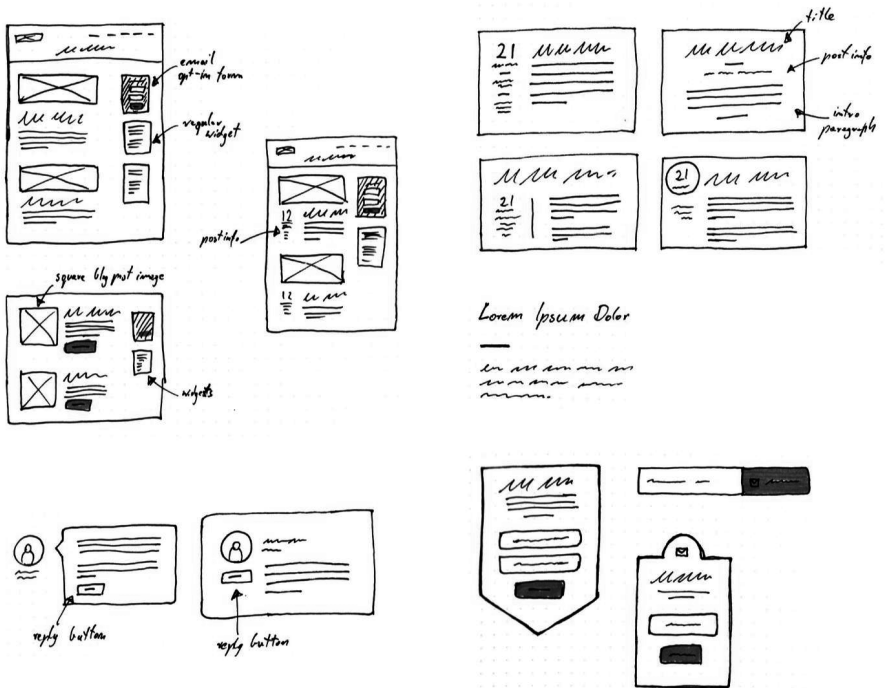


Рисунок 3.14 – Пошукові ескізи для інших розділів ресурсу

Опрацьовуємо також усі необхідні модельні вікна та форми зворотного зв’язку, як, наприклад, форма підписки по e-mail, яка однозначно повинна бути привабливою для користувача Треба переконатися, що використані спливаючі вікна різної форми для залучення уваги. В останньому кроці замальовки ідей необхідно опрацювати різні кнопки. У підсумку приходимо до результату, який будемо використовувати в проекті.

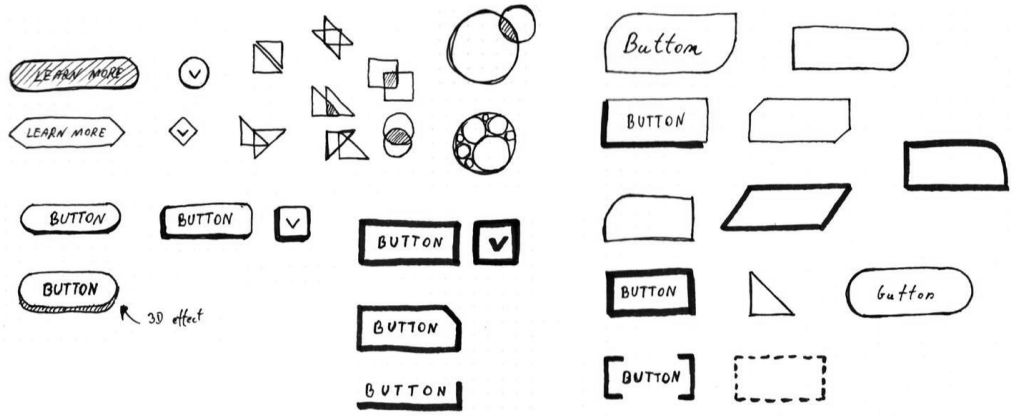


Рисунок 3.15 – Пошукові ескізи кнопок

3.5.2. Вайрфрейм (Wireframe)

Образ дизайну низької точності (lo-fi), що візуалізує дизайн-проект гранично точним чином та допомагає отримати зворотній зв'язок на початковому етапі проектування програмного забезпечення. Вайрфрейм або каркасний прототип – це графічне зображення програми або веб-сайту, що містить найнеобхідніші елементи та вміст. Каркасна конструкція схожа на план будівлі. Коли хтось хоче побудувати масивну будівлю, вони не починають відразу, право. Замість цього вони ескізують, малюють, роблять креслення,

обчислюють тощо. Вайрфрейми не лише набір сірих блоків, а й база майбутньої системи, що зображує кожну деталь фінального проекту. Він повинен чітко визначати основну групу контенту, структуру інформації та описувати базову візуалізацію взаємодії між інтерфейсом і користувачем. Вайрфрейм не містить занадто багато деталей, але створює точний образ фінального продукту, без втрати жодної важливої деталі, та є фундаментом майбутнього проекту та путівником для розробників, графічних дизайнерів, менеджерів проекту. Візуалізація повинна бути побудована за правилами естетики, але сильно спрощена завдяки схематичному відображенню.

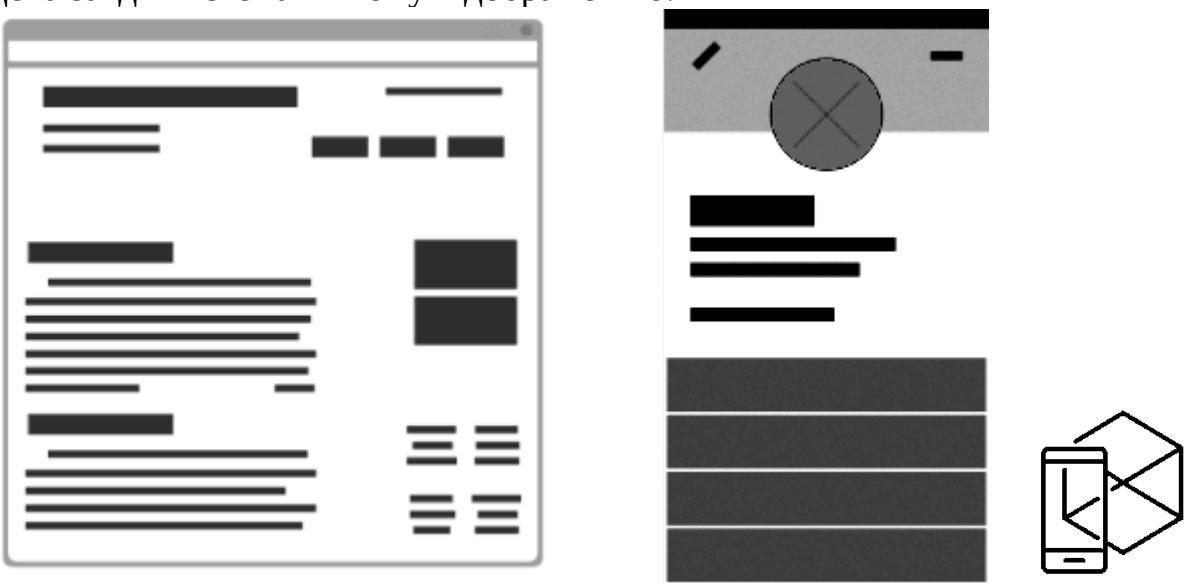


Рисунок 3.16 – Приклади вайрфреймів

Етап прототипування, на якому створюються структурні схеми сторінок (***wireframes***), що показують, яка інформація і елементи керування повинні розташовуватися на сторінках системи, є найважливішим у проектуванні інтерфейсу. Це ще не дизайн, але вже основа для нього, технічне завдання.

Вайрфрейми використовуються у складі документації до проекту. Оскільки вони є статичним і фіксованим способом відображення інтерфейсу, тому вони потребують детального опису, наприклад як частина технічного завдання. Розробка вайрфреймів низької точності допомагає знайти середину між перевитратами та переосмисленням, адже вони вимагають занадто мало інвестицій та у результаті надають занадто багато тестування користувачами. Кілька характерних особливостей каркаса такі:

- Він показує основні фрагменти контенту
- Він малює контур і структуру макета
- Він зображує найпростіший інтерфейс користувача

Однією з величезних переваг створення каркасу є те, що він не дорого і швидко його виконувати. Ви можете показати його потенційним користувачам і попросити зворотний зв'язок, який є великим, тому що люди будуть приділяти більше уваги функціональності та досвіду користувача, ніж естетиці. Пізніше ви все одно зможете налаштувати естетику. Опрацьовані раніше скетчі

оформляємо у вигляді вайрфреймів для заповнення технічної документації. Виглядають вони наступним чином:



Рисунок 3.17 – Реалізація вайрфреймів за пошуковими ескізами

3.5.3. Прототип або макет (Prototype)

Середньо-високе точне відображення (lo-hi) кінцевого продукту, що є динамічною симуляцією остаточної взаємодії між користувачем та інтерфейсом. Воно може не виглядати як кінцевий продукт, але має за мету надати користувачеві уявлення щодо вмісту та взаємодії з інтерфейсом, і надає можливість перевірити основний функціонал програмного забезпечення. Таке

моделювання дій користувача формує основу для перевірки зручності використання інтерфейсу (usability-тестів) до початку розробки.



Рисунок 3.18 – Приклад прототипу

Взаємозалежність інтерфейсу та програмного забезпечення часто відсутня з метою скорочення витрат та пришвидшення циклу розробки. Прототипування – досить дорога за собівартістю та витратами людино/годин форма комунікації, тому доречно створення прототипів, що можуть бути використані в процесі розробки, тобто з використанням гіпертекстової розмітки HTML та каскадних таблиць стилів CSS.

3.5.4. Мокап (Moqup)

Високо деталізований (hi-fi) статичний дизайн-образ інтерфейсу програмного засобу, що використовується для презентації загального відображення дизайну кінцевого продукту.

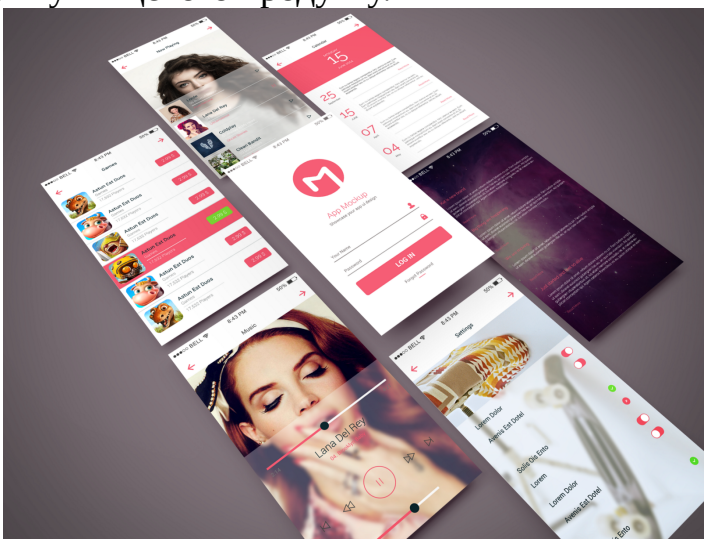


Рисунок 3.19 – Приклад мокапу

Як вайрфрейм, мокап статичний але має багатші візуальні елементи. Він зосереджений на візуалізації продукту шляхом додавання багатьох візуальних елементів, щоб досягти вищої точності, ніж вайрфрейм, включаючи графіку, макет, колір, шрифтове вирішення та іншу більш детальну візуальну презентацію. Часто мокап – це вже актуальний графічний дизайн, що

використовується в дискусіях з візуального представлення продукту для швидкого зворотного зв'язку та вдосконалення візуального дизайну програмного інтерфейсу.

Розглянемо, яким чином намальовані вище скетчі трансформувались після вайрфреймів у мокап:

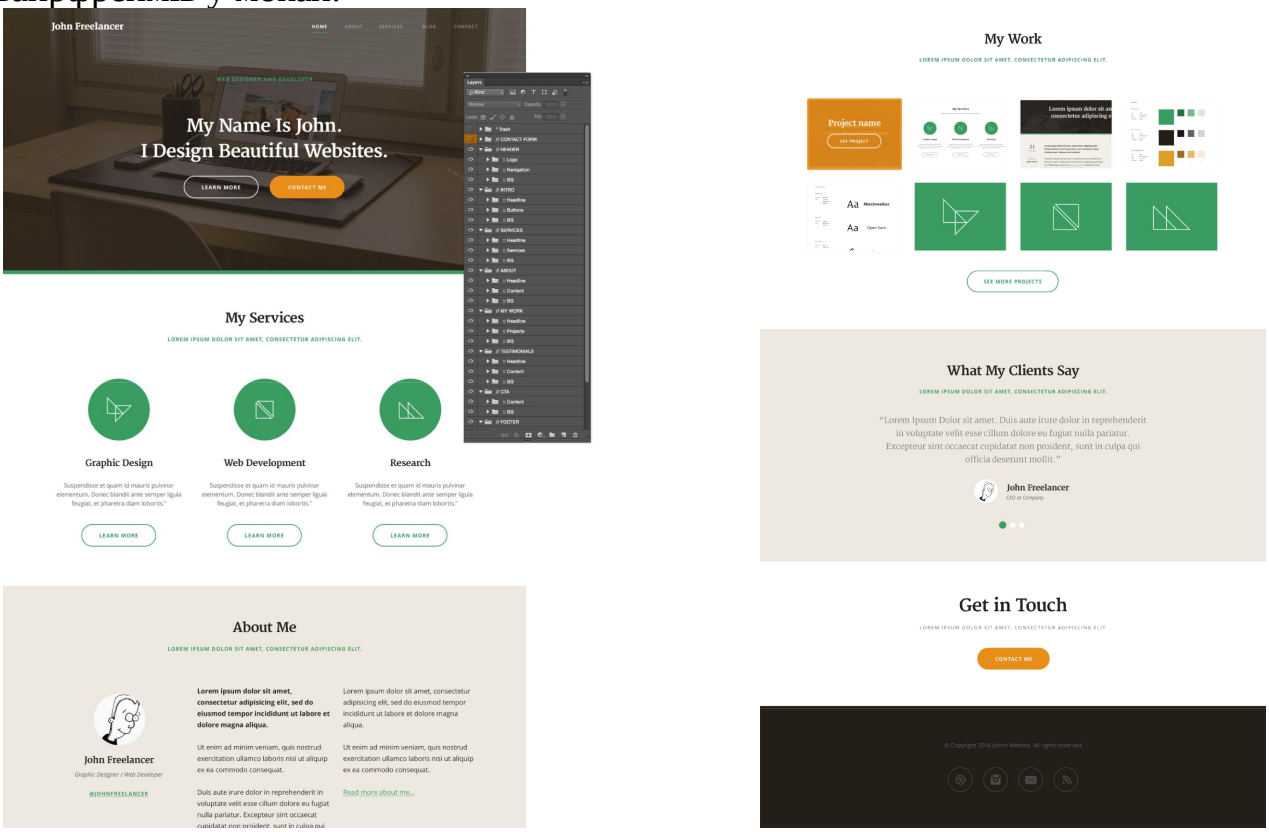


Рисунок 3.20 – Реалізація мокапу за пошуковими ескізами та вайрфреймами

Проте є деякий ризик у роботі з мокапами, адже є велика ймовірність критики вибору шрифтів, колірних комбінацій, розмірів кнопок тощо, нехтуючи формулюванням недоліків щодо концепцій. Замість того, щоб зосередити увагу на підтвердженні основних концепцій проекту, прототипи високої точності перенаправляють увагу на естетику продукту. З іншого боку, вайрфрейм "змушує користувачів думати про зміст, а не про зовнішність". Тож неможливо остаточно визначити який метод прототипування оптимальний, адже кожен з них має як свої недоліки, так і значні переваги.

Так прототипування інтерфейсів на папері в XXI столітті є найшвидшим, найдешевшим, кросплатформним та загальнодоступним методом. Проте прототипування програмними засобами, так і за допомогою олівця та паперу, дозволяє проводити тестування динамічно, та, як було у рамках цього дослідження експериментально підтверджено, паперовий та комп'ютерний носії мають однакову силу для тестування прототипів.

Але незважаючи на окремі переваги у використанні під час прототипування програмного інтерфейсу за допомогою олівця та паперу, застосування під час прототипування спеціалізованих програмних засобів та онлайн-ресурсів є на наш погляд більш доречним з огляду на можливість детально описати взаємодію користувача, продемонструвати інтерактивність, та

працювати над проектуванням програмного засобу командою у режимі спільного доступу.

Чи потрібно вам використовувати прототипи?

Є дуже багато причин, за якими вам варто використовувати прототипи в процесі створення дизайну:

Економія часу і швидкі оновлення.

Саме найбільшу перевагу створення прототипів полягає в тому, що вони можуть зберегти вам цілу купу часу. Вони вимагають від вас прийняття найважливіших рішень про становище контенту на сторінці і структурі сторінки, перш ніж ви поринете в створення графічного дизайну.

Ви так само можете спробувати нові ідеї, ви зможете подивитися на результат без витрачання величезної кількості часу на створення дизайну. Робити різні зміни на даному етапі набагато простіше і швидше, ніж в той момент, коли у вас вже готовий ідеальний, піксель в піксель, дизайн.

Можливість поглянути на дизайн під іншим кутом.

Прототипи немов креслення ваших веб-сайтів. Погляд на ваш сайт з архітектурою перспективи допоможе вам прийняти найкращі рішення про те, який макет і структуру вам варто вибрати. Добре розроблена структура може бути основною для всього дизайну.

Отримання відгуків на ранніх стадіях проекту.

Ви можете використовувати прототипи для візуального спілкування з іншими людьми.

Ви можете показувати їм свої концепти дизайну.

Завжди здорово отримувати відгуки від клієнтів і від інших членів команди на ранніх стадіях створення дизайну. Ти зможеш здійснити будь-які важливі зміни, перш ніж буде занадто пізно і це дозволить тобі працювати набагато менш напружено протягом усього проекту.

Концентрація на зручність використання і функціональності.

Використовуючи тільки шаблон контенту і опускаючи дрібні деталі дизайну ви можете повністю сфокусуватися на зручність використання і функціональності вашого веб-сайту. Прототипи дозволять вам планувати наперед положення важливих елементів веб-сайту і бути більш гнучким пробуючи різні варіанти. Це може бути не так просто під час фази роботи над графічним дизайном. Прототипи є дуже важливою частиною процесу створення дизайну, але це не означає, що вам необхідно створювати прототип кожної сторінки. Використовуйте прототипи тільки, коли це необхідно. Це означає, що деякі сторінки мають очевидну і просту структуру, так що простого начерку буде досить для планування структури.

На чому сфокусуватися при створенні прототипів?

Є кілька важливих речей, які вам варто пам'ятати при створенні прототипів:

Простота.

Дуже хочеться зробити детально-пророблений і красивий прототип. Ви повинні більше фокусуватися на макеті веб-сайту, структуру контенту,

особливості та функціональності, ніж над тим як він виглядає. Пізніше у вас буде величезний простір для створення графічного дизайну і вам не варто витрачати надто багато часу на створення прототипів..

Пишіть анотації.

Прототипи, які дуже просто виглядають може бути досить складно прочитати і зрозуміти іншим людям (і часом тобі самому буде складно їх зрозуміти). Переконайся, що ти записав анотації з поясненням конкретних елементів прототипу. Якщо ви створили вікно з назвою "Call To Action," ви можливо захочете додати коментар з інформацією, на що посилається дане вікно.

Використовуйте реальні розміри.

Робота над вашими прототипами в тому ж самому розмірі в пікселях, що і кінцевий сайт помітно допоможе вам. Це допоможе вам вірно розпланувати весь макет дизайну веб-сайту. Звичайно, не всі повинно бути чудово підігнано і виміряно, але у вас буде можливість краще оцінити кількість місця, яке необхідно для конкретних функцій або елементів вашого веб-сайту.

Пробуйте різні концепти.

прототипами ви можете працювати набагато швидше, ніж з макетами остаточного дизайну. Використовуйте це як перевага і спробуйте різні варіанти дизайну вашого веб-сайту. Експериментуйте з деякими новими макетами і подивіться, чи зможете ви знайти щось, що виглядає краще. У вас не буде цієї гнучкості в майбутньому.

Створюйте макет який вирішує проблеми.

Прототип – це в цілому всього лише макет веб-сайту і хороший макет може вирішити багато дизайнерських проблем. Зосередьтеся на тому, щоб ваш макет було зручно використовувати, і він мав необхідної функціональністю.

Коли ви починаєте створювати прототипи, існують деякі важливі рішення, які вам потрібно зробити. Звичайно ж, можуть бути деякі організаційні зміни, коли ви розробляєте повноцінні графічні макети, але вам потрібно дотримуватися того макету, який ви вибрали на стадії прототипування.

Створення вайрфрейму можна порівнювати з малюванням малюнка на білому аркуші паперу. Коли ви малюєте, ви використовуєте олівець тільки для відображення тіней вашого зображення. Ви не можете малювати світлом, так що вам потрібно думати навпаки і сфокусуватися на зображенні темних областей, замість світлих. Всі інші білі області на папері і створюють ваше зображення. Аналогічно, коли ви створюєте макет веб-сайту вам потрібно сфокусуватися на розробці дизайну даних порожніх білих областей і вам потрібно залишити місце для контенту, яким буде надалі заповнений весь веб-сайт. Замість того щоб створювати макет навколо контенту, вам потрібно спочатку розпланувати макет, а потім вже помістити контент в нього.

3.6. Логічна карта сайту.

Карта сайту – це відмінний інструмент, який дозволить вам створити повну структуру веб-сайту. Вона може допомогти вам отримати більш детальне

уявлення про те, що з себе представляє весь проект, перш ніж ви перейдете до створення дизайну.

Карти сайту зазвичай засновані на невеликому описі проекту і його вимогах. Стандартна карта сайту виглядає як графік, який показує ієрархію і зв'язку між головними і другорядними сторінками. Вона допомагає дизайнерам і розробникам краще розуміти структуру веб-сайту.

Так само, через те що вони відображаються графічно, вам дуже просто буде пояснити іншим членам команди або клієнтам загальну структуру. Вам не потрібні технічно підковані люди, щоб читати карти сайтів.

Карти сайтів повинні відображати навігацію веб-сайту, включаючи, крім головних і другорядних ще й приховані сторінки, лендінги або спливаючі віконця. Карта сайту на прикладі портфоліо для фрилансера досить проста, так що ми сфокусуємось тільки на головних сторінках, які є у кожного веб-сайту. Ось так виглядає карта сайту для проекту портфоліо:



Рисунок 3.21 – Приклад карти сайту

Карти сайту можуть допомогти вам визначити те, для яких сторінок ви будете розробляти дизайн. Очевидно, ми можемо розробити шаблон однієї сторінки, який буде дуже схожий на сторінки "про проект" і всі сторінки сервісів.

3.7. Карта навігації

Розробка дизайну макетів полягає в презентації або візуальному відображенні веб-сторінок. Це ще не повністю функціональний веб-сайт і він не буде працювати в браузері. Це більше схоже на попередній перегляд того, як повинен виглядати сайт. Існує дуже багато дизайнерів, які пропускають процес створення макета і працюють безпосередньо з CSS / HTML кодом. Але варто пройти весь процес створення дизайну: планування, створення начерків, системи навігації, створення прототипів і навіть розробка керівництва з типографського стилю і кольорової палітри.

Без всіх цих елементів розроблених заздалегідь ви ризикуєте привнести в ваш проект величезну кількість невідповідностей. Хороший дизайн проекту критичний для успіху. Бувають такі ситуації, коли створення макетів марні. Якщо ви працюєте над веб-додатком, то варто сфокусуватися тільки на створенні елементів призначених для користувача інтерфейсу. Команда розробників може скористатися ним для створення модульної системи і з'єднає всі сторінки під час програмування. Майже кожен проект вимагає хоча б пари сторінок з макетами.



Рисунок 3.22 – Приклад проектування системи навігації

Керівництво зі стилю може містити не лише інформацію про колір і типографіки, а й кнопки, поля введення, форми, блоки, повідомлення, модальні вікна або навіть приклади фотографій. Це більше схоже на створення системи дизайну та пояснення цієї системи розробникам. У керівництві по стилям можливо відобразити типографіку, кольори і деякі елементи інтерфейсу.

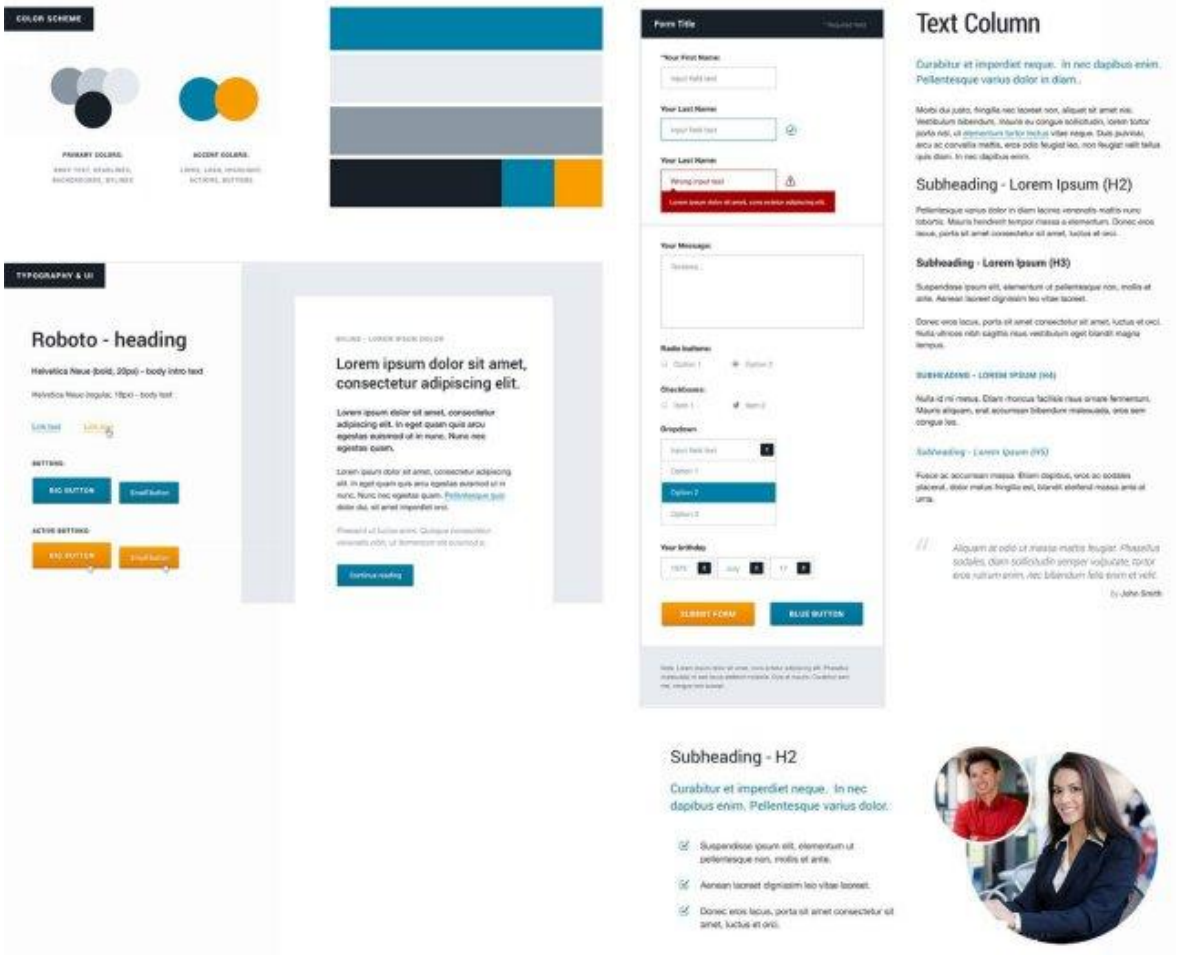


Рисунок 3.23 – Приклад оформлення розширеного керівництва зі стилю

Контрольні питання

1. Назвіть етапи розробки веб-сайту.
2. У чому полягає етап постановки завдання при розробці веб-сайту?
3. Що таке внутрішня структура веб-сайту?
4. Які складові зовнішньої структури сайту?
5. Значення властивостей яких об’єктів вибирають на етапі розробки дизайн-макету сторінок сайту?
6. Чому при розміщенні на сайті мультимедійних файлів потрібно використовувати формати, що передбачають стиснення даних?
7. Назвіть базові елементи дизайну та особливості їх використання.
8. Поясніть, чому при розробці дизайну сайту важливо враховувати основну аудиторію потенційних відвідувачів.
9. Назвіть відомі вам правила оформлення веб-сторінок.

Теми для самостійного опрацювання

1. Візуальна мова інтерфейсу. Створення карти сайту
2. Розробка логічної карти майбутньої моделі програмного засобу у вигляді блок-схем майндмепінгу

Тестові завдання

1. Керівництво по стилю містить:

a) інформацію про типографіку
b) технічні вимоги
c) вимогу до інструментарію
d) логічну карту сайту

2. Група людей схожа за будь-якими параметрами, на яку розрахований певний продукт або послуга:

a) зовнішня специфікація
b) цільова аудиторія
c) Фокус-група
d) розробники

3. Послідовний порядок проектування інтерфейсу:

a) скетч, мокап, вайрфрейм, макет
b) скетч, вайрфрейм, мокап, макет
c) скетч, вайрфрейм, макет, мокап
d) скетч, прототип, мокап, вайрфрейм

4. Дизайн точності lo-fi, що візуалізує дизайн-проект гранично точним чином та допомагає отримати зворотній зв’язок.

a) Вайрфрейм
b) Скетч
c) Прототип
d) Мокап

5. У складі документації до проекту використовується

a) Вайрфрейм
b) Мокап
c) Скетч
d) Прототип

6. Відображення lo-hi дизайну кінцевого продукту, що є динамічною симуляцією остаточної взаємодії користувача з інтерфейсом:

a) Мокап
b) Скетч
c) Вайрфрейм
d) Прототип

7. Можливість перевірити основний функціонал програмного забезпечення дозволяє:

a) Мокап
b) Прототип
c) Вайрфрейм
d) Скетч

8. Перед початком розробки для перевірки зручності використання інтерфейсу застосовують

a) usability-тестування
b) QA-тестування
c) функціональне тестування
d) QC-тестування

9. Високо деталізований hi-fi статичний дизайн-образ інтерфейсу програмного засобу:

a) Каркас
b) Скетч
c) Мокап
d) Вайрфрейм

10. Мокап використовують для:

a) usability-тестування
b) презентації дизайну кінцевого продукту
c) описання взаємодії користувача з інтерфейсом
d) опрацювання ідеї

Практичні завдання

Завдання 3.1. Розробити скетчі та вайрфрейми типових екранів користувацького інтерфейсу для десктопної версії;

Завдання 3.2. Розробити карту системи;

Завдання 3.3. Розробити карту фокусів (навігації);

Завдання 3.3. Доопрацювати дизайн-документ, додавши розроблені скетчі та вайрфрейми, картою сайту та картою фокусів.

4. ФІЛОСОФІЯ КОЛЬОРОПОДІЛУ. ТИПОГРАФІКА ТА ОСОБЛИВОСТІ ЇЇ ВИКОРИСТАННЯ У UI ДИЗАЙНІ

План:

4.1. Моделі математичного опису кольору
- 4.1.1. Модель RGB
- 4.1.2. Модель CMYK
- 4.1.3. Десяткове і шістнадцяткове представлення кольору
- 4.1.4. Представлення кольору
- 4.1.5. Змішування кольорів
- 4.1.6. Взаємозв’язок моделей RGB і CMYK
- 4.1.7. Моделі HSB і HLS
- 4.1.8. Модель Lab
- 4.1.9. Кольорові моделі для веб-дизайну

4.2. Кольорове коло

4.3. Принципи вибору та поєднання кольорів.
- 4.3.1. Монохромна схема
- 4.3.2 Complementary color scheme (Комплементарна схема)
- 4.3.3 Analogous Color Scheme (Аналогова схема)
- 4.3.4. Triadic color scheme (Тріадна схема)
- 4.3.5. Square tetradic colour scheme (Тетрадна, Чотириколірна)
- 4.3.6. Split complementary color scheme (Комплементарна тріадна)
- 4.3.7. Tetradic double complementary scheme (Подвійна розподілена контрастна схеам)
- 4.3.8. Аналогово-комплементарна схема

4.4. Кольорова температура
- 4.4.1. Абсолютна температура кольору.
- 4.4.2. Теплі кольори.
- 4.4.3. Холодні кольори.
- 4.4.4. Відносна температура. Холодні і теплі відтінки кольорів.
- 4.4.5. Холодні і теплі відтінки кольорів.
- 4.4.6. Температура ахроматичних кольорів.

4.5. Тон, насиченість і світлосила

4.6. Хроматичність та ахроматика

4.7. Створення колірних схем

4.8. Типографіка для Інтернет

4.9. Візуалізація шрифтів.

4.10. Адаптація

4.11. Класифікація шрифтів

4.12. Основні атрибути стилю тексту

4.13. Вибір і поєднання шрифтів

4.14. Рекомендації по роботі з версткою

Попри широко розповсюджене уявлення про те, що вибір кольорів для користувацького інтерфейсу в основному залежить від дизайнерського смаку та відчуття краси. Проте процес вибору кольору складніший, ніж здається, і відіграє надважливу роль у дизайні.

Кольори мають великий вплив на наш настрій та поведінку. Саме тому успіх продукту багато в чому залежить від кольорів, обраних для дизайну.

Дослідження, надані компанією Colorcom, показують, що люди потребують лише 90 секунд для того, щоб зробити підсвідому оцінку продукту, від 62% до 90% – лише на основі кольору. Отже, правильно підібрані кольори можуть бути корисними для покращення конверсії вашого продукту, а також для зручності використання продукту.

Щоб створити гарний дизайн та використовувати кольори більш ефективно, потрібно зрозуміти, як формуються кольори та як вони співвідносяться один з одним. Ось чому студенти в художніх школах, коледжах та університетах вивчають науку теорії кольорів, присвячену природі кольору.

Більшість сучасних моніторів можуть відображати понад 16 мільйонів кольорів. Це всі комбінації, які можливо створити, використовуючи різні відтінки, насиченість і тон. Ця кількість є надмірною для сприйняття та тим паче для оперування кольором. З метою ефективного оперування кольором та кольоровими поєднаннями було розроблено теорію кольору. Вона допомагає зрозуміти природу кольору і методи ефективного використання кольорових поєднань, в дизайні інтерфейсів та веб-дизайні зокрема. Все що ми знаємо про колір сьогодні, є результатом інтенсивного вивчення його природи найбільшими умами з усього світу протягом декількох століть. Завдяки цим дослідженням, кольори зібрані в групи, що дозволяє оцінити, як вони співвідносяться один до одного. Розуміння теорії кольору допоможе ліквідувати найчастіші помилки у поєднанні кольорів, які можуть викликати дискомфорт або виглядати важкими для сприйняття. Теорія кольору так само допоможе вибрати вірний відтінок і тон, який буде впливати на поведінку користувачів. Вибір, змішування і використання вірних кольорів може бути критично важливим для успіху вашої роботи над інтерфейсом.

4.1. Моделі математичного опису кольору

Для опису кольору пікселів використовують універсальні моделі, де застосовується точний опис кольору в стандартизованих цифрових і математичних виразах.

Вимоги стосовно колірних моделей

- Колір в моделі повинен визначатися в спосіб, що не залежить від спроможностей певних конкретних пристроїв.
- Модель повинна точно і однозначно визначати гамму (діапазон, колірний обхват) заданих кольорів.
- Модель повинна враховувати особливості сприйняття, поглинання або відбивання гамми заданих кольорів.

Існує багато різних моделей опису кольору, але всі вони належать до одного з трьох типів:

- *Адитивні***.** Засновані на складанні випромінювань окремих зон спектру світла і застосовуються для середовищ, де світло випромінюється (екрані, сканування).
- *Субтрактивні***.** Засновані на поглинанні окремих зон спектру світла при відбиванні або пропусканні світла і застосовуються у середовищах, що поглинають світло (забарвлені поверхні, фарбники).
- *Психологічні***.** Засновані на сприйняття кольорів людиною і пов'язані з особливостями його зорової системи.

Самими поширеними моделями є:

- *RGB* — адитивна колірна модель.
- *CMYK* — субтрактивна колірна модель.
- *HSB* та *Lab* — психологічні колірні моделі.

4.1.1. Модель RGB

Модель RGB є найбільш простою для розуміння і застосовується в пристроях, де світло випромінюється або колір утворюється електронним шляхом: у моніторах, телевізорах, слайд проекторах, сканерах тощо.

Колір в моделі RGB представлено сумою яскравостей трьох базових кольорів — червоного (*Red*), зеленого (*Green*) і синього (*Blue*). Назву моделі утворено з перших літер англійських назв цих кольорів.



Рисунок 4.1 – Колірна модель RGB

Для опису кольору в моделі RGB застосовують 3 байти (24 біти=3х8 біт), по одному байту на кожну колірну складову.

Яскравість (інтенсивність) кожного базового кольору може приймати 256 (2^8) дискретних значень від 0 до 255. Змішування кольорів в різних пропорціях, варіюючи яскравість кожної складової надає 256 х 256 х 256 = 16 777 216 кольорів.

Кожний колір можна представити в закодованому вигляді, що містить значення яскравості трьох колірних складових.

4.1.2. Модель CMYK

Модель RGB добре підходить для моделювання кольорів, що утворюються електронним шляхом, але не є доречною у разі моделювання кольорів для друкованих зображень.

Модель CMYK описує змішування фарб, які використовують для друкованих зображень. В цій моделі теж використовуються три базові кольори:

блакитний (*Cyan*), пурпуровий (*Magenta*) і жовтий (*Yellow*). Крім того, застосовується чорний колір (*blacK*).

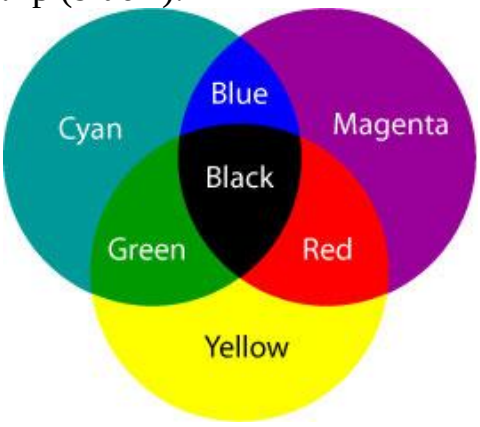


Рисунок 4.2 – Колірна модель CMYK

Кожен з трьох базових кольорів моделі CMYK утворюється в результаті вилучення з білого кольору одного з базових кольорів моделі RGB. Так, наприклад, блакитний (Cyan) утворюється внаслідок вилучення червоного з білого, а жовтий (Yellow) — вилученням синього.

- *Cyan* = RGB – R = GB = (0, 255, 255);
- *Yellow* = RGB – B = RG = (255, 255, 0);
- *Magenta* = RGB – G = RB = (255, 0, 255).

Оскільки базові кольори CMYK утворюються внаслідок вилучення з білого кольору базових кольорів RGB, їх називають додатковими, бо вони доповнюють основні кольори до білого, а модель CMYK субтрактивною (що віднімає).

Базові кольори моделі CMYK є достатньо яскравими кольорами і не цілком підходять для відтворення темних кольорів. Так, при їх змішуванні на практиці утворюється не справжній чорний, а брудно-коричневий колір. Тому, колірна модель CMYK містить ще і чистий чорний колір, який використовується для створення темних відтінків, а також для друкування чорних елементів зображення.

4.1.3. Десяткове і шістнадцяткове представлення кольору

Десяткове представлення — це трійка десяткових чисел, що розділені комами. Перше число відповідає яскравості червоної складової, друге — зеленої, а третє — синьої.

Наприклад, червоний колір: (255,0,0); зелений колір: (0,255,0), синій колір: (0,0,255)

Код кольору в шістнадцятковому представленні має вид 0xXXXXXX. Префікс 0x вказує на те, що в коді використано власне шістнадцяткове число. Після префіксу зазначено шість шістнадцяткових цифр (0, 1, 2... 9, A, B, 3, D, E, F).

Перші дві цифри — шістнадцяткове число, що представляє яскравість червоної складової, друга і третя пари відповідають яскравості зеленої та синьої складових.

Наприклад, червоний колір: 0xFF0000; зелений: 0x00FF00; синій: 0x0000FF.

Якщо всі складові мають максимальну яскравість (255,255,255 — в десятковому представленні; 0xFFFFFF — в шістнадцятковому представленні), виходить білий колір. Мінімальна яскравість (0, 0, 0 або 0x000000) відповідає чорному кольору.

Змішування базових кольорів

- Червоний і зелений – за максимальної яскравості утворюють жовтий колір.

Жовтий колір: 255,255,0 або 0xFFFF00

- Зелений і синій — за максимальної яскравості надають блакитний колір.

Блакитний колір: 0,255,255 або 0x00FFFF

- Червоний і синій — за максимальної яскравості надають фіолетовий колір.

Фіолетовий колір: 255,0,255 або 0xFF00FF

4.1.4. Представлення кольору

Колір в CMYK описується четвіркою десяткових чисел, кожна з яких визначає відсоток фарби даного кольору. Наприклад, для отримання кольору «хакі» слід змішати 30% блакитної фарби, 45% фіолетової фарби, 80% жовтої фарби і 5% чорної. Позначити такий вибір можна в два способи:

(30,45,80,5) або C30M45Y80K5.

В друкарнях колірні зображення друкують в кілька прийомів. Накладаючи на папір почергово блакитний, фіолетовий, жовтий і чорний прошарки, отримують повноколірну ілюстрацію. Тому, готове зображення, що створено на комп'ютері, перед друком розділяють на чотири складові одноколірні зображення. Цей процес називається кольороподілом. Сучасні графічні редактори мають засоби для виконання цієї операції.

4.1.5. Змішування кольорів

Змішування кольорів в моделі CMYK є протилежним до змішування складових в моделі RGB. Проте, кольори субтрактивної моделі не є такими чистими, як кольори адитивної моделі. Цим і пояснюються наступні особливості:

- *Блакитний і фіолетовий* — за максимальної яскравості надають глибокий синій колір з невеликим фіолетовим відтінком.
- *Фіолетовий і жовтий* — за максимальної яскравості утворюється суміш яскраво-червоного кольору.
- *Жовтий і блакитний* — яскраво-зелений колір з невеликим відтінком синього.

Модель CMYK використовують для підготовки друкарських зображень. Тут, світло не випромінюється, а поглинається. Чим більше фарби покладено на папір, тим більше світла вона поглинає і менше відбиває. Поєднання трьох основних кольорів поглинає майже все падаюче світло, і з боку зображення виглядає майже чорним. На відміну від моделі RGB збільшення кількості фарби приводить не до збільшення візуальної яскравості, а навпаки, до її зменшення.

4.1.6. Взаємозв’язок моделей RGB і CMYK

Основні кольори моделей RGB і CMYK знаходяться в залежності, яку представлено на рисунку.

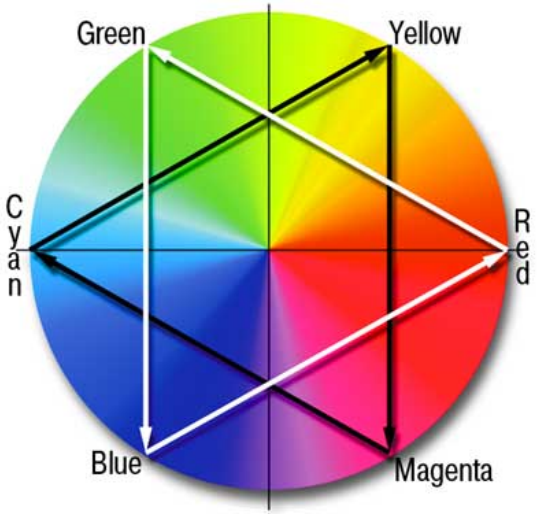


Рисунок 4.3 – Взаємозв’язок моделей RGB і CMYK

Кожен колір розташовано між кольорами, за допомогою яких його утворено і напроти кольору, що його доповнює. Щоб підсилити будь який колір, слід ослабити його доповнюючий колір, що розташований на протилежній стороні кола.

Наприклад, щоб підсилити жовтий (Yellow), слід ослабити синій (Blue). На колі кольорів жовтий розташовано між зеленим (Green) і червоним (Red). Складання цих кольорів надає жовтий (Yellow).

Не всі кольори моделі CMYK можна представити в моделі RGB, і навпаки. У кількісному відношенні колірний діапазон CMYK є меншим за колірний діапазон RGB. Ця обставина є важливою, і має принципове значення, а не обумовлена лише фізичними особливостями монітора, друкуючого пристрою чи фарбників.

4.1.7. Моделі HSB і HLS

Модель HSB засновано на трьох параметрах:

- H — відтінок або тон (Hue).
- S — насиченість (Saturation).
- B — яскравість (Brightness).

Модель HSB краще за моделі RGB і CMYK, відповідає поняттю кольору, яке використовують професійні художники. Дійсно, в них, зазвичай, є кілька основних фарб, а решта утворюються внаслідок додавання до них білої або чорної фарб. Таким чином, потрібні кольори — це певна модифікація основних кольорів: висвітлення або затемнення. Хоча, художники і змішують фарби, але це вже виходить за рамки моделі HSB.

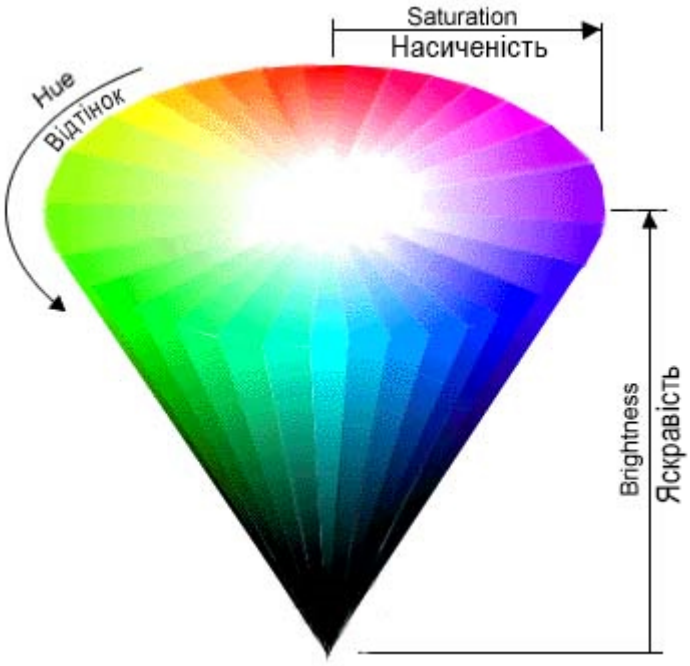


Рисунок 4.4 – Колірна модель HSB

Насиченість характеризує чистоту кольору. Нульова насиченість відповідає сірому кольору, а максимальна насиченість — найбільш яскравому варіанту даного кольору. Можна вважати, що зміна насиченості пов'язана з додаванням білої фарби, тобто зменшення насиченості відповідає додаванню білої фарби.

Яскравість розуміється як ступінь освітленості. При нульовій яскравості колір стає чорним. Максимальна яскравість при максимальній насиченості надають найбільш виразний варіант даного кольору. Можна вважати, що яскравість змінюється шляхом додавання чорної фарби. Чим більше чорної фарби додано, тим менше яскравість.

Графічно модель HSB можна представити у вигляді конусу, вздовж якого розташовано відтінки кольорів. На зовнішньому краю конусу знаходяться чисті спектральні кольори або колірні тони (параметр H вимірюється в кутових градусах, від 0 до 360). Чим ближче до центру кола розташовано колір, тим меншою є його насиченість, тим більш він є бляклим, пастельним (параметр S вимірюється у відсотках). Яскравість (освітленість) відображається на лінійці, що перпендикулярна до площини колірного кола (параметр B вимірюється у відсотках). Всі кольори на зовнішньому колі мають максимальну яскравість.

В певних графічних редакторах, наприклад, в Adobe FreeHand, використовується модель *HLS* (*Hue*, *Lightness*, *Saturation*), яка є схожою до HSB. У моделі HLS, на відміну від HSB, замість яскравості застосовано параметр L — освітленість (*Lightness*). Зменшення освітленості наближає колір до чорного, а збільшення — до білого. Чистий спектральний колір утворюється при освітленості 50%.

Моделі HSB і HLS є апаратно незалежними і не орієнтовані на жоден технічний пристрій.

4.1.8. Модель Lab

Модель Lab засновано на трьох параметрах:

- *L* — яскравість (*Luminosity*)
- *Два колірні параметри — a і b*. Параметр a містить кольори від темно-зеленого через сірий до яскраво-рожевого. Параметр b містить кольори від світло-синього через сірий до яскраво-жовтого.

Поняття яскравості в моделях Lab і HSB не є тотожними. Як і в RGB, змішування кольорів зі шкал a і b дозволяє отримати яскравіші кольори. Зменшити яскравість кольору можна за рахунок параметра яскравості L.

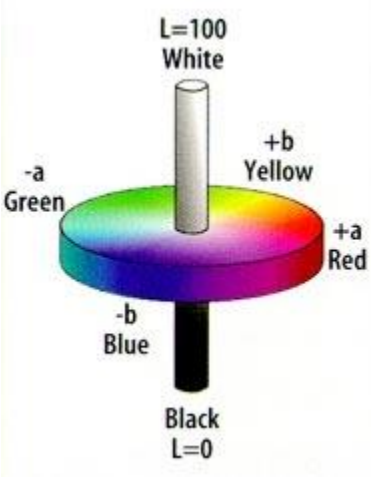


Рисунок 4.5 – Колірна модель Lab

Модель Lab є апаратно незалежною, її колірний діапазон покриває діапазони RGB і CMYK. Графічний редактор Adobe Photoshop при переході від режиму RGB до CMYK використовує Lab як проміжний етап.

4.1.9. Кольорові моделі для веб-дизайну

Налаштування кольору в веб-дизайні досить прості.

Є 2 найпопулярніші колірні схеми:

• RGB (Red, Green, Blue) Ця колірна схема використовується для всієї графіки, яка відображається на екрані.

• CMYK (Cyan, Magenta, Yellow, Black). Ця колірна схема використовується для всієї графіки, яка буде надрукована.

Як ви можете бачити, RGB це єдина колірна схема, з якою ви будете працювати під час створення дизайну веб-сайтів. Проте, існує величезна кількість колірних схем, які використовують колірну палітру RGB; наприклад sRGB, Adobe RGB і т.д. Схеми визначають саме ті кольори, які ви бачите на вашому екрані монітора або на будь-яких інших пристроях з RGB дисплеями.

Деякі дизайнери рекомендують використовувати Adobe RGB (1998) тому в ньому присутні більше кольорів, ніж в sRGB. Це правда, що Adobe RGB має глибшу гамму кольорів, але веб-дизайнерам не варто їй користуватися.

Найкраще використовувати кольорову схему sRGB, яка є стандартною колірною схемою і була створена HP спільно з Microsoft в 1996 році. Використовуючи sRGB на вашому робочому місці, ви будете впевнені, що кольори, які ви бачите вашому графічному редакторі будуть дуже близькі до

тих, які бачать інші люди в їх браузерах. sRGB є найбільш часто використовуваної колірної схемою на екранах комп'ютерів. Звичайно ж те, що ви бачите на екрані не завжди буде на 100% збігатися з тим, що інші люди будуть бачити, тому що монітори у всіх різні. Так чи інакше, вам варто використовувати стандартну колірну схему, це покаже те, що ви хороший дизайнер.

4.2. Кольорове коло

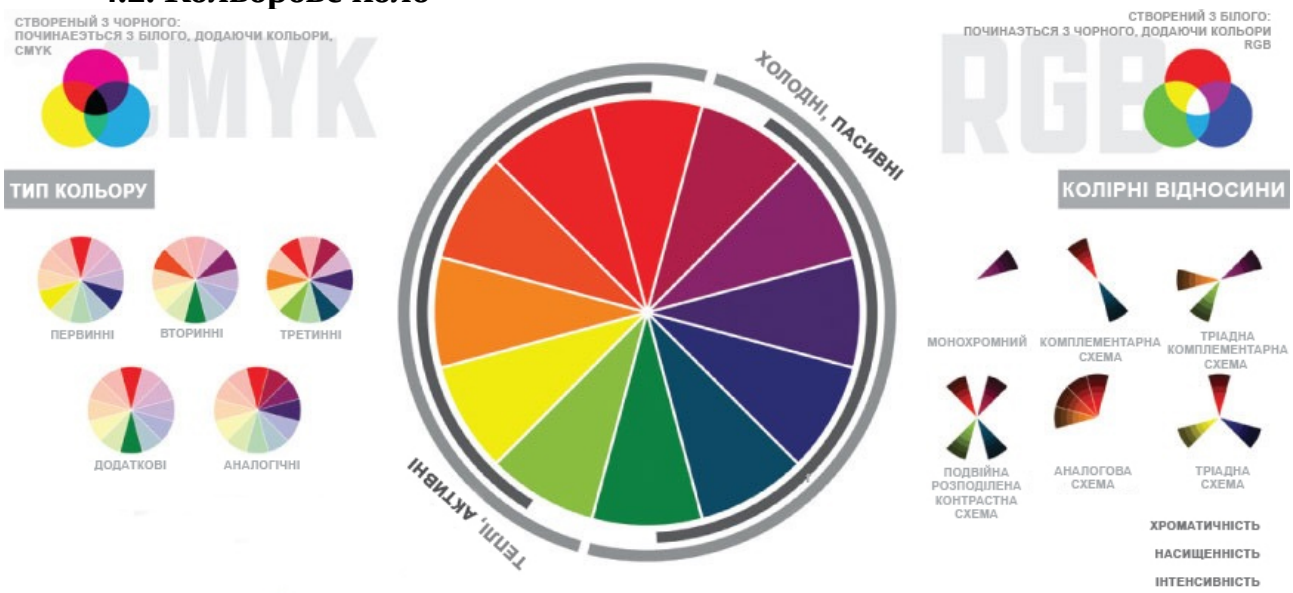


Рисунок 4.6 – Кольорове коло, температура та кольорові моделі CMYK та RGB

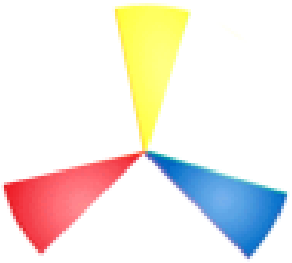


Рисунок 4.7 – Три основні кольори

Кольорове коло – це візуальне відображення відносин між різними групами кольорів: основними, вторинними і третинними кольорами (їх так само називають додатковими кольорами). Перше колірне коло було створено сером Ісааком Ньютоном в 1706 році і з того часу дуже багато художників і вчених досліджують цю ідею, використовуючи трикутники, таблиці, списки і модифіковані версії колірного кільця Ньютона. З тих часів воно пройшло через багато перетворень, але все ще залишається основним інструментом для комбінування кольорів. Стандартне коло, яке використовується, ґрунтується на трьох основних кольорах (Рисунок 4.7) Далі, змішуючи основні кольори ми можемо отримати три вторинних (Рисунок 4.8.) З'єднуючи вторинні кольори з основними кольорами утворюються третинні кольори (Рисунок 4.9).



Рисунок 4.8 – Вторинні кольори



Рисунок 4.9 – Третинні кольори

Кольорове коло допомагає зрозуміти відносини між конкретними кольорами і їх групами за допомогою логічної структури. Кольори в ньому розташовуються по колу, але це приховує безліч технік змішування. Кольорове коло допомагає створити гармонійну колірну схему.

4.3. Принципи вибору та поєднання кольорів.

У дизайні колірний баланс має велике значення, так як враження про сайт або додатку у користувачів складається з першого погляду, і кольори роблять на це сильний вплив. Дизайнери визначили основні і найбільш ефективні колірні палітри, або колірні гармонії.

4.3.1. Монохромна схема

Заснована на одному кольорі і його різних тонах і відтінках. Монохромна палітра завжди є безпрограшним варіантом.

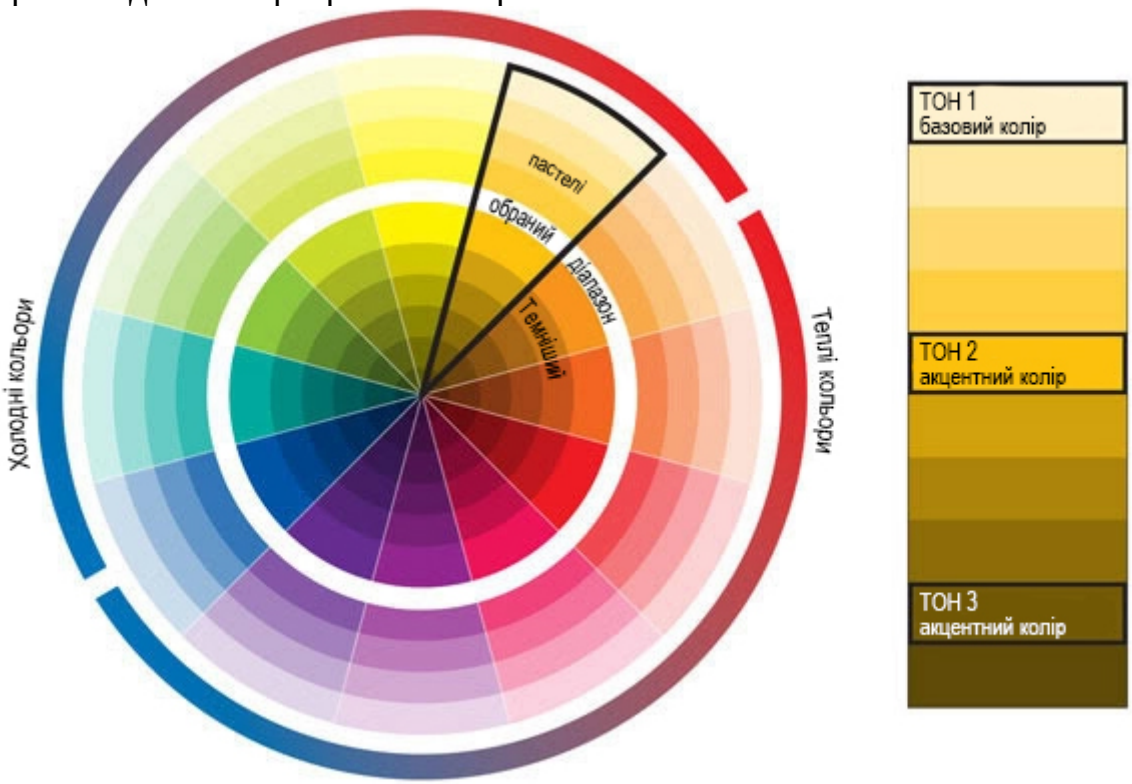


Рисунок 4.10 – Приклад добору монохромної схеми



Рисунок 4.11 – Приклад фрагменту сайту заснованого на монохромному поєднанні

4.3.2 Complementary color scheme (Комплементарна схема, додаткові, контрастні кольори)



Комплементарна колірна схема створюється за допомогою двох кольорів розташованих один проти одного на колірному колі (їх називають complementary – додаткові). При використанні контрастної колірної схеми важливо вибрати домінуючий колір і використовувати додатковий колір для акцентів.

Наприклад, основний колір для фону і додатковий колір, щоб виділяти важливі елементи. Комплементарна колірна схема найбільш контрастна з усіх колірних схем і найбільш сильно привертає увагу, але в цій схемі важче досягти балансу кольорів, ніж в монохроматичної або аналогової схемах, особливо при використанні м'яких, теплих відтінків.

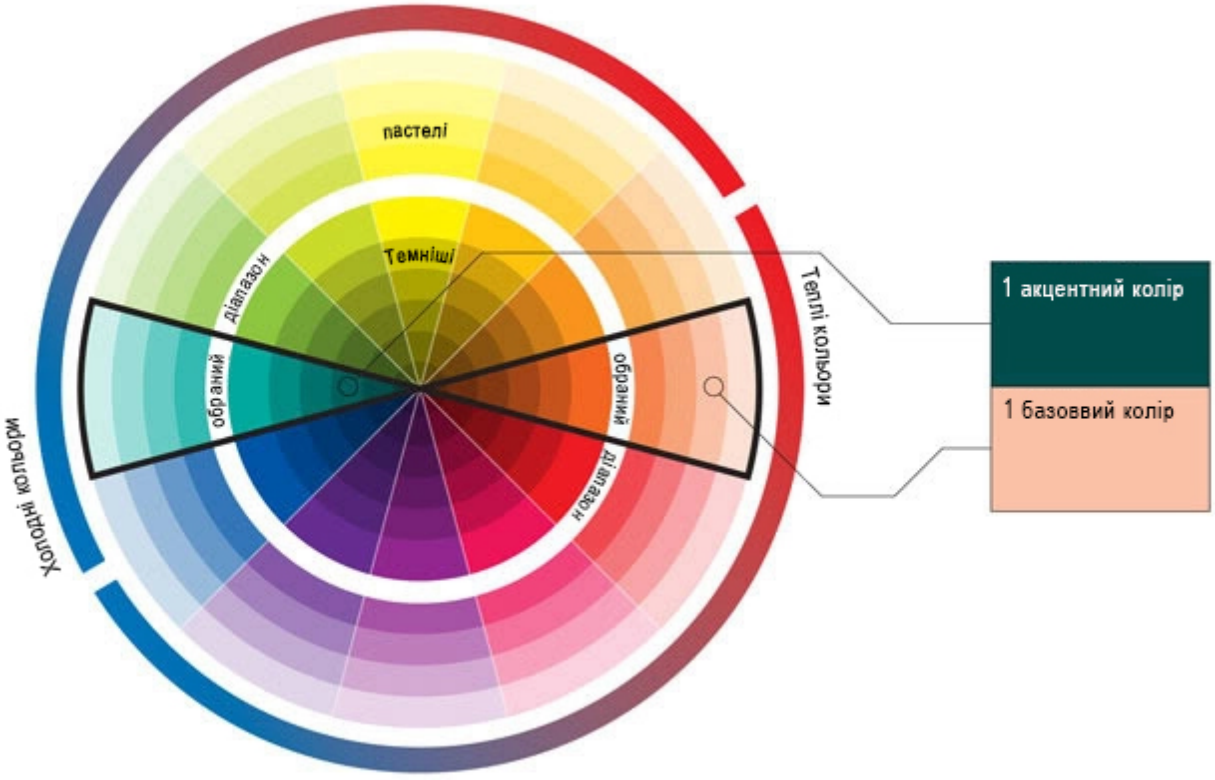


Рисунок 4.12 – Приклад добору комплементарної схеми

Всі протилежності на колірному колі створюють сильний контраст, тому їх використовувати можна, але обережно.

4.3.3 Analogous Color Scheme (*Аналогова схема*)



У аналогової колірній схемі використовуються два кольори, розташовані поруч один з одним на колірному колі. Зазвичай один колір використовується як домінуючий, в той час як інші кольори є допоміжними. Аналогова колірна схема подібна монохроматичної, але пропонує більше нюансів. Цей вид кольорової палітри використовується там, де не потрібен контраст, в тому числі на тлі веб-сторінок або банерів. Не можна користуватися кольорами, розташованими дуже близько один до одного

на колірному колі – дисонанс між такими кольорами буквально б'є по очах (як ріже вухо дисонанс між нотами, які відрізняються тільки на півтону). Найкраще контрастують один з одним кольори, розташовані приблизно на відстані чверті кола один від одного. Крім того, два теплих або два холодні кольори викликають більшу спорідненість один з одним, ніж кольори з протилежних півкуль. Такі кольори на колірному колі не мають сильного контрасту.

Типова аналогова схема не повинна містити більше 3-4 кольорів, інакше колірна гармонія буде порушена. І хоча аналогові колірні схеми легко складати, вони можуть стати монотонними. Потрібно використовувати різні світлини, насиченості і розташування кольорів для розвитку інтересу до такої колірної схемою.

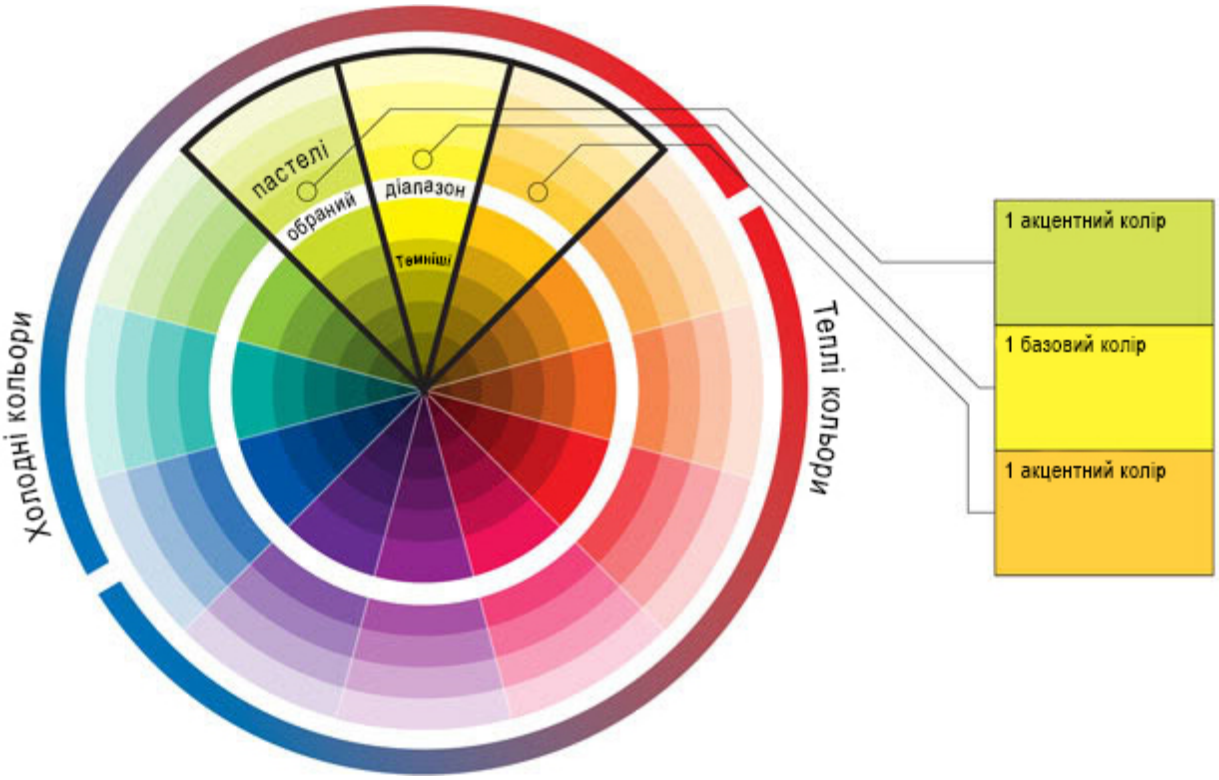


Рисунок 4.12 – Приклад добору аналогової схеми



Рисунок 4.13 – Приклад аналогової схеми у дизайні

Ось деякі способи, за допомогою яких можна надати будь-якого кольору домінуючу роль: виділити для цього кольору більшу площу, зробити його

темніше, зробити його більш насиченим. Уникайте розташування поруч двох однаково яскравих кольорів за допомогою додавання невеликої кількості одного кольору до інших. Інший спосіб підтримки інтересу до аналогової схемою – введення комплементарних акцентів.

4.3.4. Triadic color scheme (*Тріадна схема)*



Практично безпрограшна комбінація – це трикутник.

У тріаді використовуються три кольори, рівновіддалені один від одного на колірному колі. Ця схема популярна, тому що вона пропонує сильний візуальний контраст, зберігаючи при цьому баланс і колірну насиченість. Тріада не така контрастна, як комплементарна схема, але виглядає більш збалансованою і гармонійною.

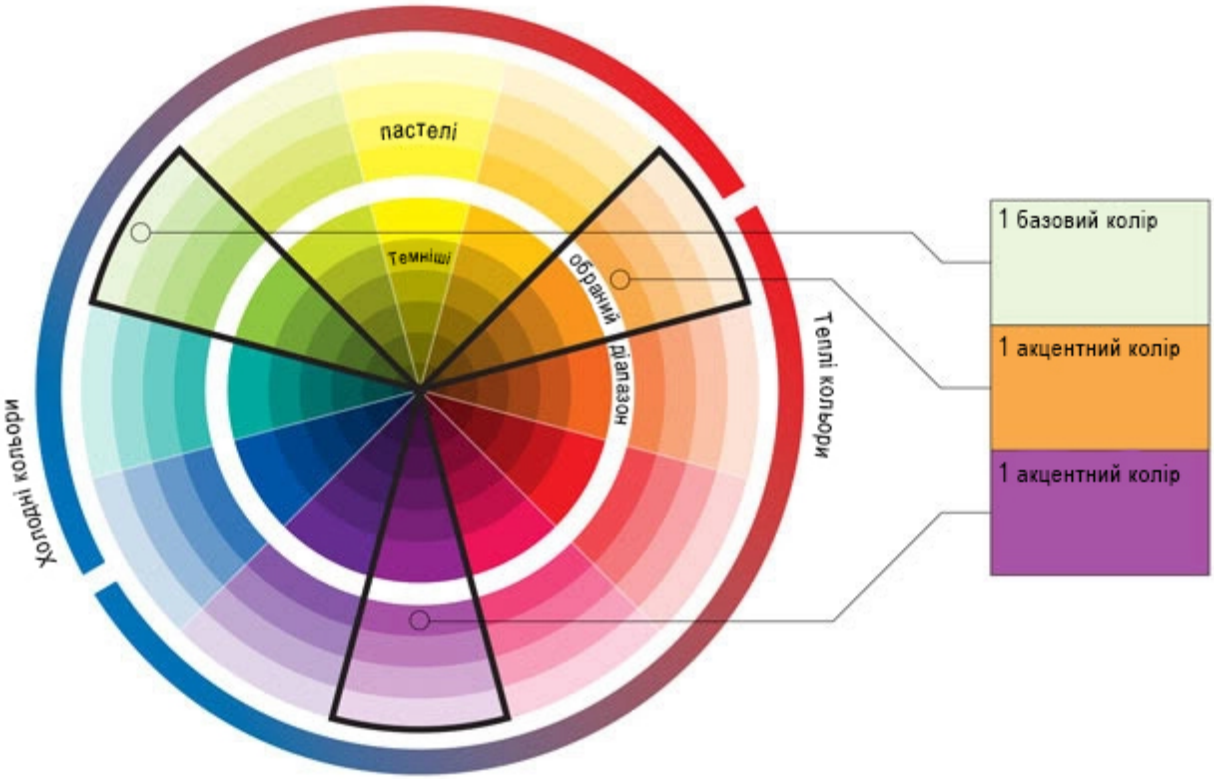


Рисунок 4.14 – Приклад добору тріадної схеми

Тріадна колірна схема будується з основного і пари додаткових, які розташовані строго на 1/3 відстані від основного на колірному колі. Як бачите, схема досить яскрава. Деякі кольори взаємно контрастні. Тому використовувати їх слід також акуратно. Взагалі завжди варто дотримуватися одного простого принципу: два взаємно контрастні кольори можна використовувати як фон і об'єкт одночасно. Інакше буде різати око. Можливості тріад практично безмежні. Наприклад, змініть світлоту і насиченість одного або всіх кольорів тріади теоретичних кольорів, і це все ще буде третинна тріада, але з іншим настроєм. Ваш вибір залежить від того, що ви хочете висловити даної тріадою.

Коли дизайну потрібно більше кольорів, можна вдатися до тріадної схеми. Вона заснована на трьох окремих рівновіддалених один від одного кольорах.

Для збереження балансу в схемі рекомендується використовувати один колір в якості домінуючого, а два інших як акцидентні.

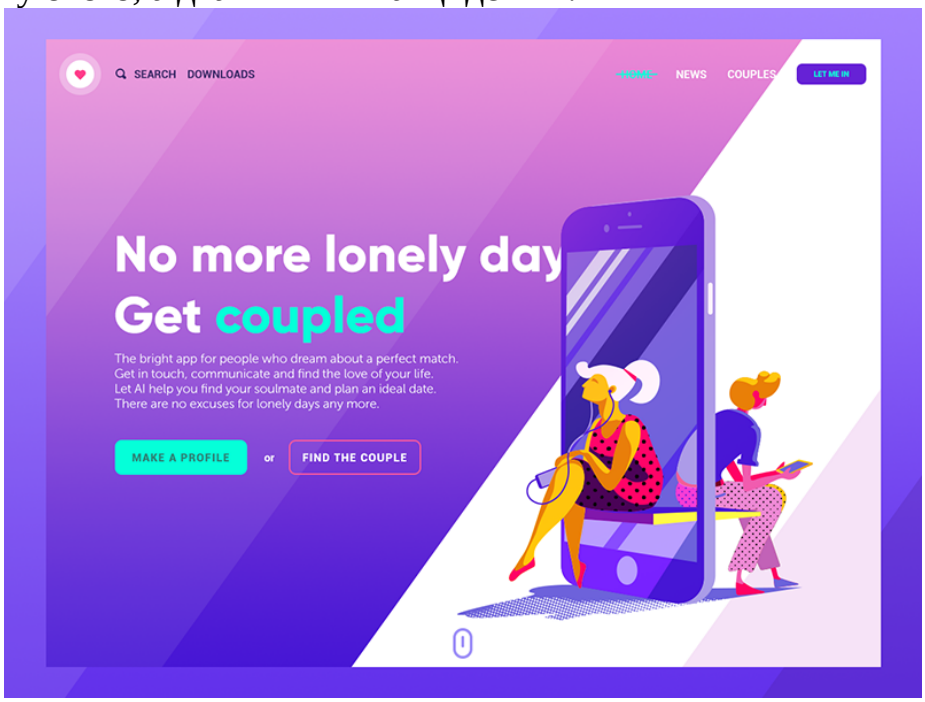


Рисунок 4.15 – Приклад використання тріадної схеми у веб-дизайні

4.3.5. Square tetradic colour scheme (Тетрадна, Чотириколірна)



Дві пари додаткових кольорів слід застосовувати дуже обережно, аби не було хаосу в тепло-холодності. Завжди впевнено заявляє про себе, оскільки вона складається з пари первинного-вторинного кольору і пари третинних.

Квадратна тетрада схожа на прямокутну, але її відмінність в тому, що кольори розташовані на однаковій відстані один від одного (через 2 кольори на 12-ти приватному колірному колі). Оскільки квадратна тетрада складається з двох пар додаткових (комплементарних) кольорів, то її також називають подвійний комплементарної схемою (як і прямокутну). У квадратній схемі потрібно також стежити за балансом теплих і холодних кольорів. Квадратна тетрада краще виглядає на практиці, якщо один з кольорів тетради зробити домінантним. Втім, дане твердження підходить практично всім видам колірних гармоній, і причину цього треба шукати в різному впливі кольору в залежності від займаної площі.

4.3.6. Split complementary color scheme (*Комплементарна тріадна схема)*



Є модифікацією стандартної комплементарної схеми. Комплементарна палітра формується поєднанням кольорів, які знаходяться один навпроти одного на колірному колі. Ця схема протилежна аналоговій і монохромній, так як її метою є створення контрасту. Її відмінність полягає в тому, що можна варіювати тон протилежних кольорів також, як це робиться в аналоговій колірній схемі. Роздільна комплементарна схема пропонує більше нюансів, ніж комплементарна, але більш важка в балансуванні.

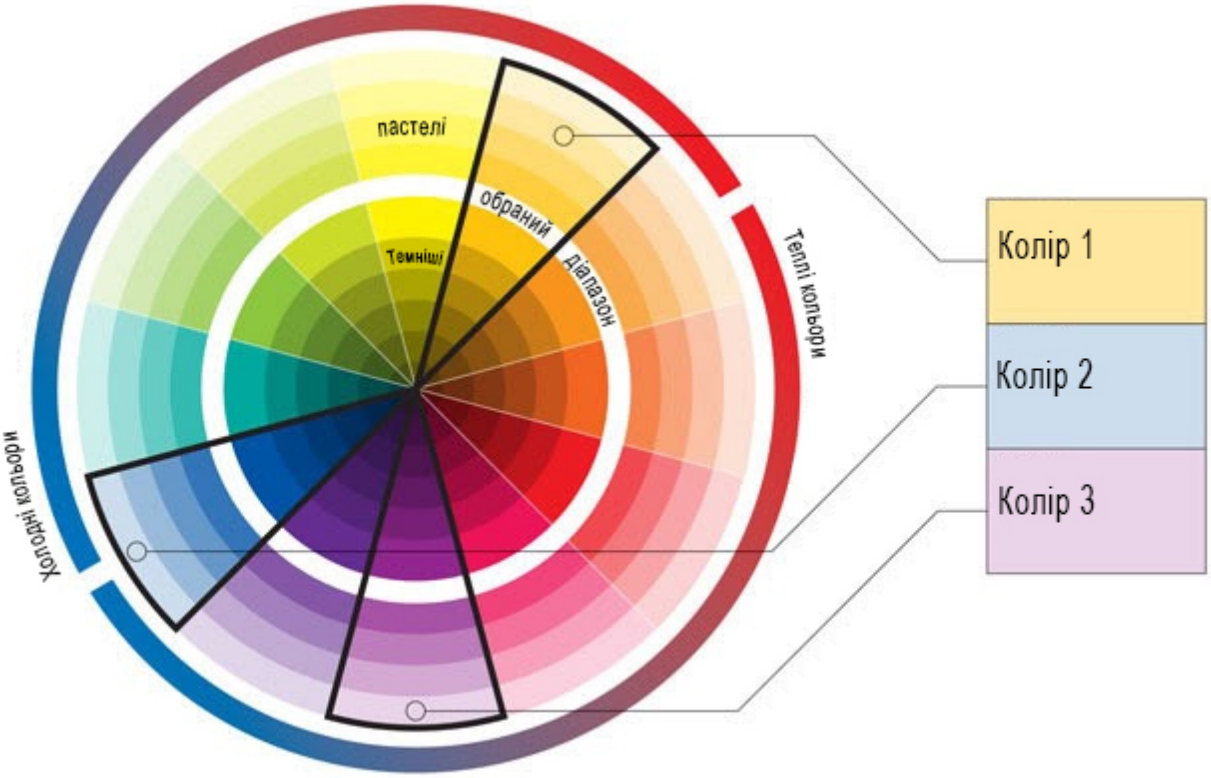


Рисунок 4.16 – Приклад добору комплементарної схеми

У такій схемі рекомендується використовувати один теплий колір проти діапазону холодних кольорів (наприклад, червоний проти синього і синьо-зеленого). Наприклад, в будь-якому інтерфейсі складно буде не побачити помаранчеву кнопку на синьому тлі.

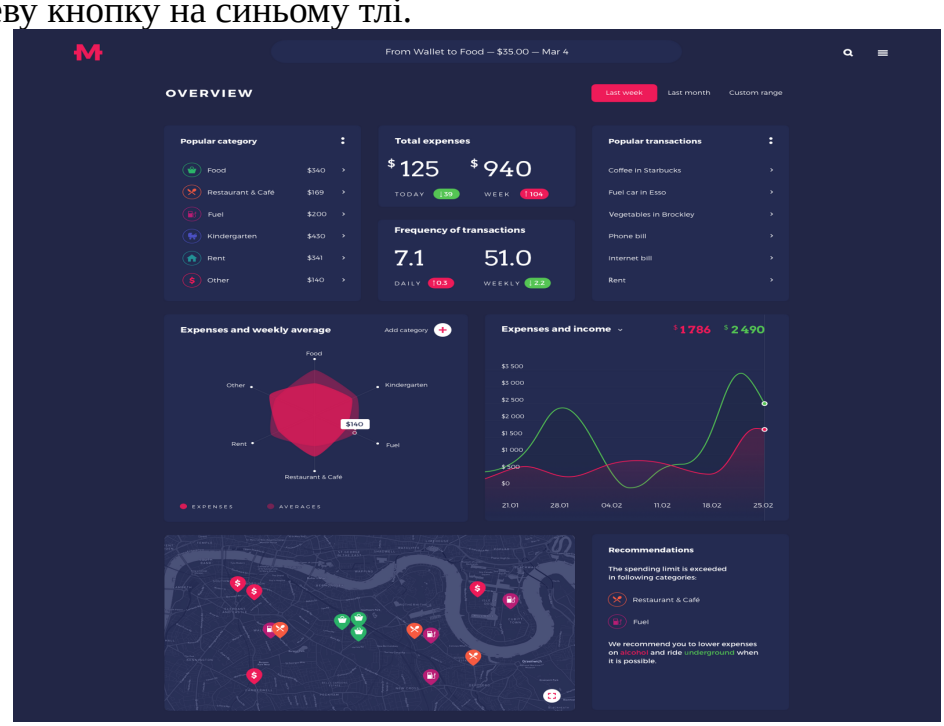


Рисунок 4.17 – Приклад використання комплементарної схеми у інтерфейсі адміністративної частини сайту

4.3.7. Tetradic double complementary color scheme (Подвійна розподілена контрастна схема)



Ця схема найбільш насичена кольорами, ніж всі інші. У ній використовуються чотири кольори – дві пари протилежних (комплементарних) кольорів. У подвійній комплементарної схемою важко досягти гармонії; якщо використовувати в композиції всі чотири кольори в рівних пропорціях, схема буде здаватися розбалансованою, тому треба правильно визначити домінуючий і підлеглі кольору для розстановки акцентів. Близько до подвійного комплементу варто дуальний комплемент (dual complementary), коли 2 пари комплементарних кольорів є сусідніми один одному. У подвійних і дуальних комплементарних схемах одна пара кольорів теплі, а друга холодні, що також представляє труднощі в їх балансуванні стосовно гардеробу. Однак, якщо привести кольору в дуальної і подвійний комплементарних схемах до загального "температурному" знаменника (наприклад, додавши жовтий підтон до холодних кольорів або синій до теплих), то така схема зможе успішно застосовуватися для багатих контрастів, природно, зробивши один колір з 4 х домінантним. Кольорові акценти можна використовувати для залучення уваги. Контрастні кольорові сполучення, наприклад, чорний з білим, в першу чергу звертають на себе увагу, тому ви можете оформити таким чином важливі кнопки – «замовити» або «зв'язатися з нами». Четвертинна колірна схема призначена для використання досвідченими дизайнерами, так як в ній найскладніше досягти балансу. Вона формується з чотирьох кольорів з кола, які становлять комплементарні пари. Якщо з'єднати точки на обраних кольорах, вони утворюють прямокутник. У цій схемі досить складно досягти гармонії, але якщо все робити правильно, результати будуть вражаючими.

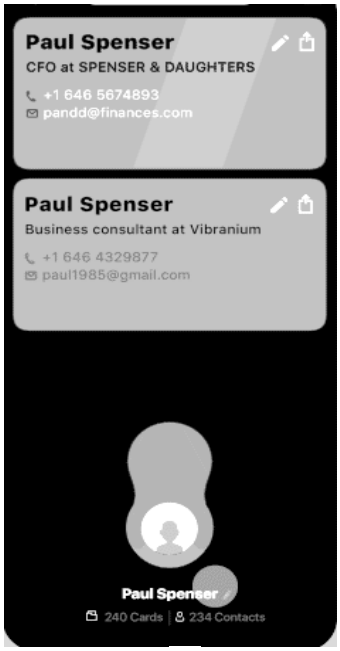


Рисунок 4.18 – Приклад застосування подвійної розподіленої контрастної схеми



4.3.8. Аналогово-комплементарна схема



Використовує родинні кольору, що лежать поруч на колірному колі, з кольором, який прямо комплементарний одному з них. Прямий комплемент стає акцентним кольором. Або вибираємо сусідній прямому комплементу колір (що лежить поруч з одну або іншу сторону з прямим комплементом кольором, що знаходиться всередині аналогової групи кольорів). Таким чином ми створюємо домінуючу колірну

групу з трьох споріднених кольорів, акцентовану прямим комплементом (або сусіднім комплементом) одному з них. Комплементарний який акцентував колір створює цікавий контраст домінуючою групи кольорів. Ця схема виглядає краще, коли ми робимо акцент на теплий колір в холодній аналогової палітрі, або навпаки, холодний акцент до теплої аналогової палітрі. Акцентні кольори вносять додаткову іскру, динаміку і яскравість в дизайн. Оскільки аналогово-комплементарна схема є окремим випадком аналогової і комплементарної схем, вона як би їх з'єднує воєдино, то області її застосування нам зрозумілі з уже вивченого матеріалу з відповідних тем. Ця палітра працює за аналогією з попередньою, але використовує більше кольорів. Наприклад, при виборі синього кольору необхідно ще додати два суміжних відтінки його протилежного кольору, тобто, жовтий і оранжевий. Тут контраст буде не таким різким в порівнянні з комплементарної схемою, але можна використовувати більше кольорів.

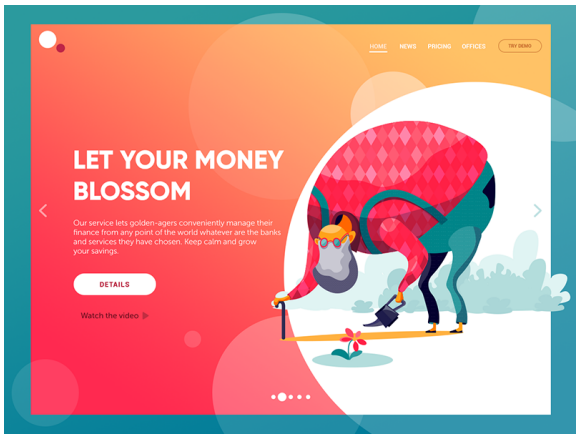


Рисунок 4.19 – Приклад застосування аналогово-комплементарної схеми у веб-дизайні

4.4. Кольорова температура

Температура є одним з атрибутів кольору. Є дві групи: теплі і холодні.

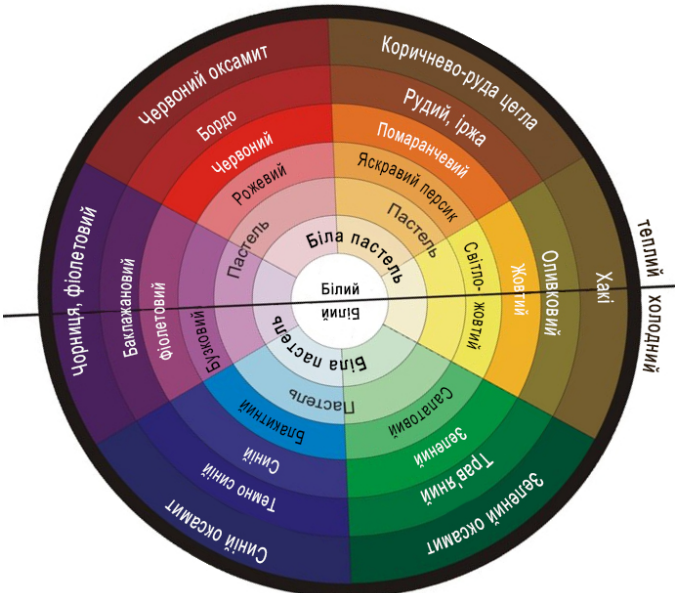


Рисунок 4.20 – Температура кольорового кола

Температуру кольору легко визначити за допомогою колірного кола. Вона буває абсолютною і відносною.

4.4.1. Абсолютна температура кольору.

Розділимо колірний круг на дві половини. На верхньому полюсі розташовується найбільш теплий колір – помаранчевий. Він вважається найтеплішим, тому що не має холодних відтінків, пізніше ми розглянемо це властивість детальніше. На нижньому полюсі знаходиться найхолодніший колір – синій. З боків колірного кола нейтральні по температурі кольору – зелений і пурпурний. Обидва утворені сумішшю холодного і теплого кольору, зелений – жовтого і синього, пурпурний – червоного і синього. Всі кольори верхньої половини вважаються теплими, а нижній — холодними.Ахроматичні кольори: білий, чорний і сірий відносяться до нейтральних.

4.4.2. Теплі кольори.

Теплі кольори асоціюються з денним світлом, заходом і вогнем. Вони дуже енергійні і відображають тепло. Гарячі кольори включають в себе всі відтінки від червоного до жовтого, включаючи рожевий, коричневий, оранжевий і жовтувато-коричневий. Теплі кольори зазвичай домінують над холодними, так що, наприклад, ви можете використовувати їх на кнопках призову до дії. Просто пам'ятайте, що червоний часто асоціюється зі словом "стоп", саме тому помаранчевий і золотий є набагато більш цікавим вибором для кнопок.

4.4.3. Холодні кольори.

Холодні кольори асоціюються з водою або хмарним днем. В якості контрасту з теплими кольорами, холодні кольори можуть заспокоювати людей і надавати розслабляючу дію. Холодні кольори складаються з всіх відтінків від зеленого до синього, включаючи фіолетовий, який ближче до синього і більшість сірих відтінків.

Вони відмінно підходять для використання в якості заднього фону, тому вони можуть додати більше глибини, роблячи ваш дизайн більш ясним і розслабляючим.

4.4.4. Відносна температура. Холодні і теплі відтінки кольорів.

Розуміння відносної температури важливо при роботі з декількома кольорами і палітрою кольорів. Воно допомагає, наприклад, передати за допомогою кольору простір і обсяг на зображенні або поверхні. Крім оранжевого і синього, всі кольори можуть бути одночасно і теплими, і холодними щодо інших. За допомогою колірного кола визначити це також просто, як абсолютну температуру. Теплота знижується в міру наближення до нижнього полюсу і синього кольору, наприклад червоний або жовтий будуть холодніше оранжевого, а лимонний або маджента холодніше червоного і жовтого. Той же принцип діє в підвищенні теплоти: блакитний і фіолетовий будуть тепліше синього, бірюзовий і пурпурний ще тепліше. Температурні градації особливо наочні в аналогових і нюансних палітрах. Колір може бути теплий або холодний не тільки по відношенню інших кольорів, але і до власних відтінкам.

4.4.5. Холодні і теплі відтінки кольорів.

З визначенням температури відтінку найчастіше виникають складності. Такі поняття, як холодний червоний або теплий червоний міцно увійшли в ужиток, проте не всі розуміють під ними одне і те ж. По-перше, відносну температуру відтінку часто плутають з температурою кольору. По-друге, суб'єктивність: немає точного визначення, де починається і закінчується червоний. Тим часом, вміння визначати холодні і теплі тони важливо при роботі із зовнішністю людини, наприклад визначенні кольоротипів і підборі індивідуальних колірних палітр. Це вміння можна розвинути за допомогою досвіду і розуміння нескладного принципу. Будь-який колір, крім оранжевого, може мати теплі, нейтральні і холодні відтінки. Як визначити температуру відтінку за допомогою колірного кола? Беремо будь-який колір і визначаємо його межі. Потім знаходимо приблизний центр. Відтінки кольору, що лежать з боку помаранчевого, будуть теплими. З боку синього – холодні. Проміжні кольору без домішок теплого або холодного називають локальними або нейтральними. Візьмемо для початку зелений. Він утворений теплим жовтим і холодним синім кольорами. Холодний або теплий зелений відтінок виходить через перевагу синього або жовтого. Рухаючись вгору до жовтого кольору, отримаємо теплі відтінки, вниз до синього — холодні.

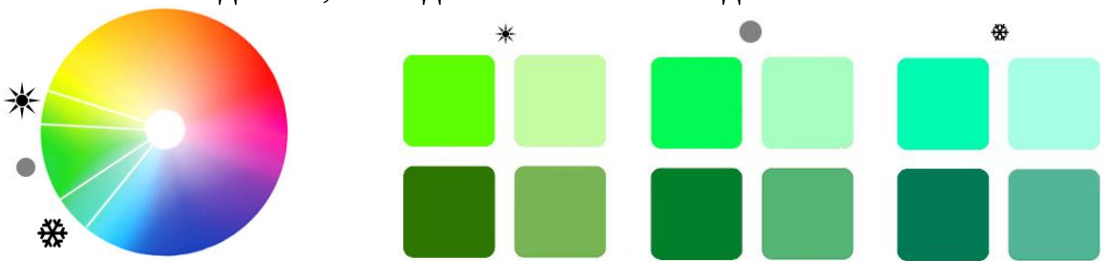


Рисунок 4.21 – Теплі та холодні відтінки зеленого кольору

Той же принцип діє при визначенні інших кольорів, наприклад, жовтого. Наближаючись до помаранчевого, колір теплішає. Спускаючись вниз, жовтий набуває зеленуватий, лимонний, холодний відтінок. Нейтральний жовтий не має явного зеленуватого або оранжевого відливу.

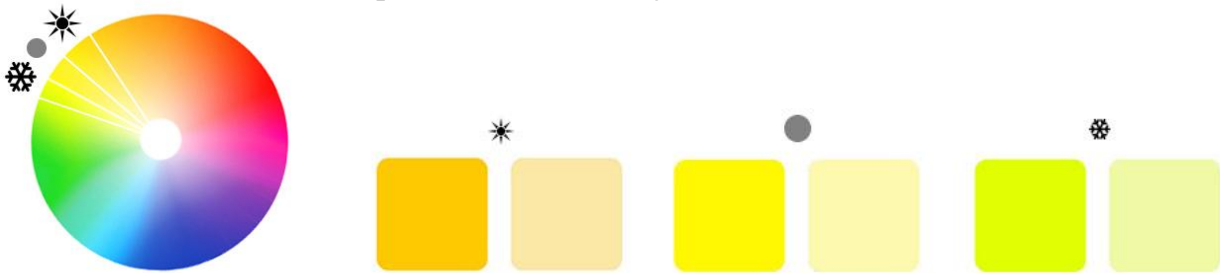


Рисунок 4.22 – Теплі та холодні відтінки жовтого кольору

Помаранчевий колір виділяється особливо. Це найтепліший і єдиний колір, який не має холодних відтінків. Крім того, він поширює теплоту на оточення. Найближчі кольору: жовто-оранжеві і оранжево-червоні теж є виключно теплими.



Рисунок 4.23 – Теплі відтінки помаранчевого кольору

Червоний. Тут діє той же принцип: верхні відтінки, підсвічені жовтим, теплі, нижні з боку пурпурного — холодні.

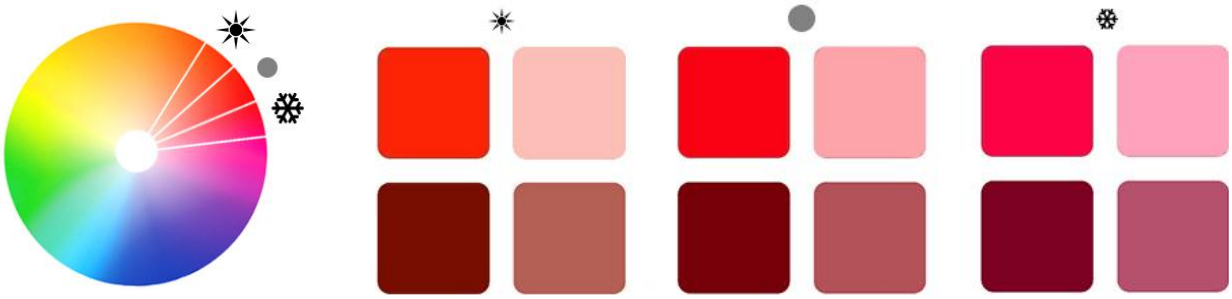


Рисунок 4.24 – Теплі та холодні відтінки червоного кольору

Пурпурний колір сам по собі нейтральний, як і зелений, він утворений сумішшю холодного і теплого кольорів. Велика частка червоного робить його теплим, синього – холодним. З точки зору використання в теплих або холодних гамах це досить складний колір. Відмінності теплого пурпурного від холодного червоного або холодного пурпурного від фіолетового важко вловимі. Складно виділити і локальний пурпурний колір.

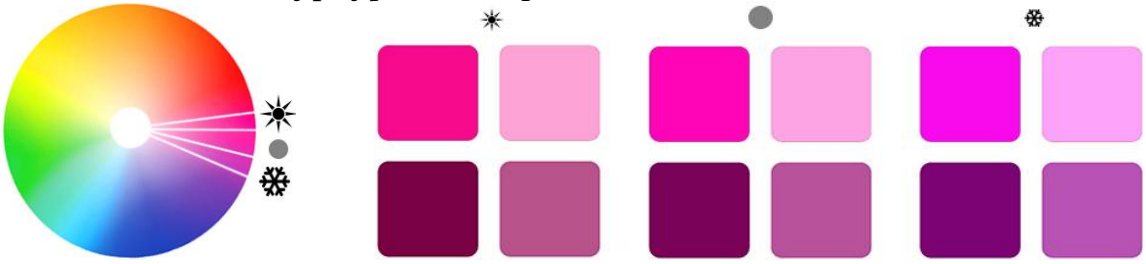


Рисунок 4.25 – Теплі та холодні відтінки пурпурового кольору

Ті ж самі труднощі з визначенням кордонів відносяться до фіолетового. При додаванні червоного кольору він теплішає, синього — холоднішає.

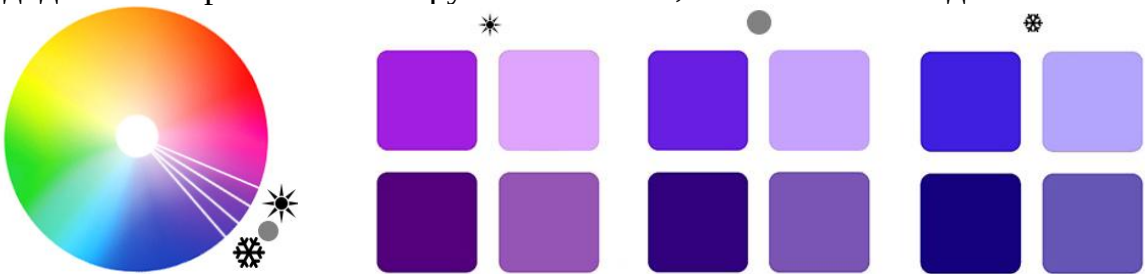


Рисунок 4.26 – Теплі та холодні відтінки синього кольору

Складність визначення температури відтінку в тому, що немає точних і загальноприйнятих розмежувань, де кінчається теплий відтінок одного кольору і починається холодний відтінок іншого. Немає чітких меж і у локальних

відтінків. Зазвичай, коли ми маємо справу з основними кольорами: червоним, синім, жовтим і зеленим цей поділ зрозуміло інтуїтивно, розмежувати інші кольори допомагає досвід. Синій колір найхолодніший з усієї палітри, це антипод оранжевого. Але якщо помаранчевий робить сусідні кольори виключно теплими і не має холодних відтінків, то синій аналогічними властивостями не володіє. Умовно можна виділити теплий синій колір. Деякі вважають, що синій за визначенням не може бути теплим, однак тепла гама кольорів може містити і синій, якщо правильно підібрати його відтінок. Його холодні, вони ж локальні відтінки розташовуються посередині, а теплі по краях: з одного боку синій підсвічений жовтим, з іншого червоним. Ці відтінки будуть тепліше щодо холодного синього.

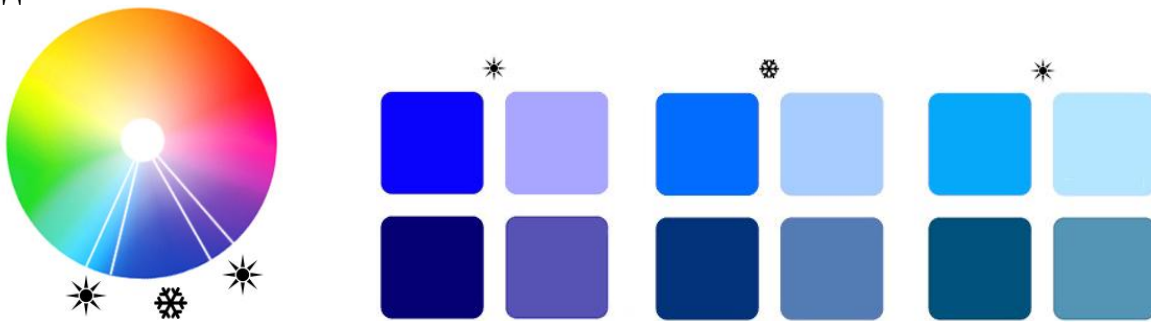


Рисунок 4.27 – Теплі та холодні відтінки блакитного кольору

Окремо виділяються синьо-зелені кольори. Тут теплота-холодність умовна і залежить від того, виділяти їх в окрему групу зі своїм локальним кольором або розглядати в складі зелених і синіх відтінків. Якщо вважати локальним кольором бірюзовий, то розташовані з боку зеленого тону будуть теплими, з боку синього — холодними. Якщо вважати локальним кольором блакитний, ті ж самі тони будуть теплі. Тут у визначенні кольору задіяні світлота і насиченість.

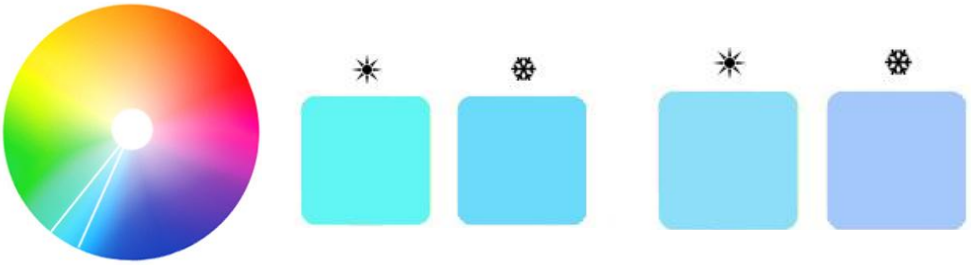


Рисунок 4.28 – Особливості температури синьо-зеленої гами

Отже, ми підійшли до впливу на температуру кольору світлини і насиченості. До цього моменту ми розглядали властивості теплоти-холодності на чистих кольорах і один параметр – тон. Але цього мало, так як найчастіше доводиться мати справу зі складними кольорами, в яких є домішка ахроматичних, тобто враховувати всі три параметри. Світлота змінюється з додаванням білого і чорного кольору, насиченість – з додаванням сірого.

4.4.6. Температура ахроматичних кольорів.

Чисті ахроматичні кольори відносяться до нейтральних. Однак, в природі важко знайти абсолютно нейтральні сірий, білий або чорний, в них завжди є перевага в будь-яку сторону. Так, холодний або теплий білий колір виходить від

домішку інших тонів. Жовто-червоні роблять його теплим, сині – холодним. Те ж саме стосується сірого і чорного.



Рисунок 4.29 – Температура ахроматичної гами кольорів

4.4.7. Температура змішаних кольорів

Для наочності тут буде зручно повернутися до моделі колірного кола і подивитися на його вертикальний зріз. По краях розташовані холодний і теплий полюса колірного кола, по центру – нейтральні кольори. Рухаючись від крайніх температурних характеристик до середини, колір наближається до протилежного полюса і тим самим нейтралізується. Іншими словами, при зменшенні насиченості, збільшенні або зменшенні світлоти, колір буде змішуватися з нейтральними ахроматичними і сам ставати нейтральним.

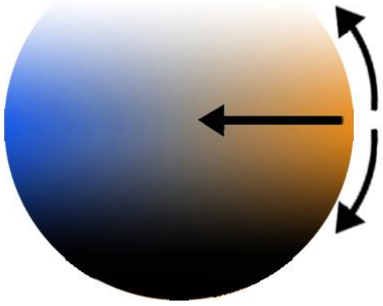


Рисунок 4.30 – Зміна насиченості, світлоти та кольору

Тепла група – червоні, жовті, стають менш теплими, їх розбавлені відтінки здаються холодніше.



Рисунок 4.31 – Температура змішаних кольорів групи червоного кольору

Розведення сірим і чорним кольором швидше за все змінює характер світлих жовтих і лимонних відтінків, вони здаються зеленими і холодними.



Рисунок 4.32 – Температура змішаних кольорів групи жовтого кольору

Помаранчевий колір не набуває холодних відтінків, але стає нейтральним. З розведенням він швидко перестає бути впізнаваним і перетворюється в коричневий.

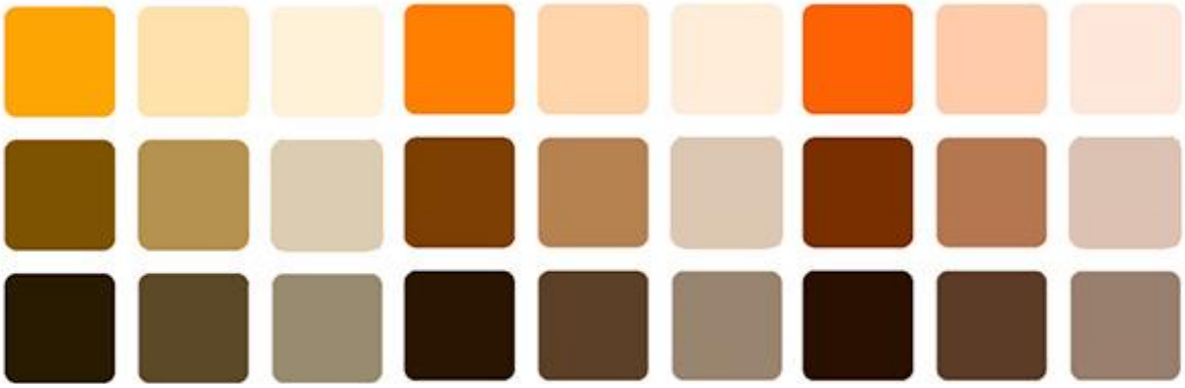


Рисунок 4.33 – Температура змішаних кольорів групи помаранчевого кольору

Сині і фіолетові з додаванням білого і сірого втрачають холодні властивості і здаються більш теплими.



Рисунок 4.34 – Температура змішаних кольорів групи червоного кольору

Як бачимо, за допомогою колірного кола легко відрізнити холодні відтінки від теплих. Труднощі виникають з визначенням синьо-червоних і синьо-зелених відтінків, тут все залежить від того, який колір вважати локальним. Складні і змішані кольору складніше чистих у визначенні теплоти-холодності. Тут необхідно розрізняти нюанси і бачити, як один і той же тон змінюється разом зі світлин та насиченістю.

4.5. Тон, насиченість і світлосила

Терміни як тон, насиченість і світлосила використовуються для того, щоб описати і згрупувати кольори. Підтримка цих трьох значень може бути критично важливими у створенні вірної колірної схеми в дизайні проекту.

Процес створення колірних схем зазвичай починається з вибору кількох чистих колірних тонів. Потім можливо змінювати їх, змінюючи насиченість і світлосилу для того, аби досягти очікуваного ефекту або відчуття. Таким способом буде повний контроль над колірною схемою, і ви зможете додати додаткову колірну схему або замінити існуючу.

4.5.1. Тон (Hue)

Тон – одна з трьох основних характеристик кольору поряд з насиченістю і світлотою. У строгому колориметричному сенсі тон – це напрямок вектора кольоровості (вектора на діаграмі кольоровості з початком в точці білого і кінцем в даній кольоровості). Напрямок може бути задано кутом (це і є колірний тон), в той час як віддаленість від точки білого задається у відсотках і називається насиченістю (ступенем змішування найбільш насичених кольорів – спектральних або крайніх пурпурних з білим). Саме тон визначає назву кольору, наприклад «червоний», «синій», «зелений». Тоном є будь-який чистий колір з усього спектра кольорів. Таким чином, цей аспект кольору зазвичай досить легко розпізнати.



Рисунок 4.35 – Чистота кольорового тону

Термін «тон» описує головну характеристику кольору, яка відрізняє червоний колір від жовтого і синього. Колір в значній мірі залежить від довжини хвилі світла випромінюваного або відбиваного об'єктом. Наприклад, діапазон видимого світла знаходиться між інфрачервоним (довжина хвилі ~700nm) і ультрафіолетовим (довжина хвилі ~ 400nm).



Рисунок 4.36 – Контраст тонів – різні тони.



Рисунок 4.37 – Контраст тону – різні відтінки, однаковий тон (синій)

4.5.2. Насиченість (Colorfulness, chroma, saturation)

Насиченість – інтенсивність певного тону, тобто ступінь візуальної відмінності хроматичного кольору від рівного по світлосилі ахроматичного (сірого) кольору. Насичений колір можна назвати «соковитим», «глибоким», менш насичений – «приглушеним», наближеним до сірому. Повністю ненасичений колір буде відтінком сірого. Чим менше насиченість вашого тону, тим більше в ньому присутність сірості. Сильна насиченість робить колір більш чистим і яскравим.



Рисунок 4.38 – Насиченість тону

У зв'язці з хроматичністю, насиченість говорить про те, як колір виглядає в різних умовах освітленості. Наприклад, кімната пофарбована в один колір, вночі буде виглядати інакше, ніж вдень. Протягом дня, незважаючи на те що колір буде незмінний, його насиченість буде змінюватися. Насиченістю не має

відношення до слів «темний», «світлий». Замість цього використовуйте слова «блідий», «слабкий» і «чистий», «сильний».



Рисунок 4.39 – Насиченість однакова – та ж інтенсивність, різні тони.



Рисунок 4.40 – Контраст насиченості – різні рівні наповнення, тон однаковий.

Насиченість, яку також називають «інтенсивністю кольору» (intensity), описує силу кольору щодо його яскравості (value) або світлини (luminance / lightness). Іншими словами, насиченість кольору позначає його відміну від сірого при певній яскравості освітлення. Наприклад, кольори близькі до сірого ненасичені в порівнянні з більш світлими кольорами. У кольорі властивість «живий» або «повний» є ні чим іншим, як відсутністю домішки сірого чи його відтінків. Важливо відзначити, що насиченість вимірюється уздовж ліній однакової яскравості.

4.5.3. Світлота (Lightness)

Світлота – одна з основних характеристик кольору поряд з насиченістю і тоном. Це суб'єктивна яскравість ділянки зображення, віднесена до суб'єктивної яскравості аналогічно освітленій поверхні, яка сприймається людиною як біла. Світлосила говорить про те, наскільки яскравий або темний у нас тон. Уявіть собі, що ви додаєте білий колір в чистий тон, таким чином, підсилюється його світлосила. При додаванні чорного кольору зменшується світлосила.

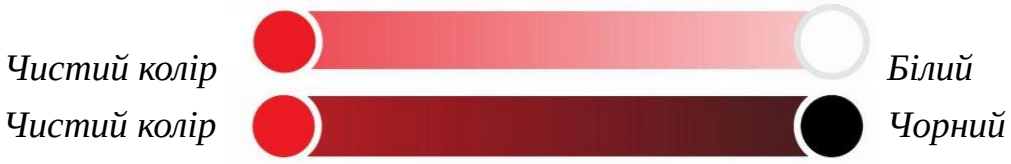


Рисунок 4.41 – Світлота кольору

4.5.4. Яскравість (Value / Brightness)

Коли ми говоримо, що колір «темний» або «світлий», ми маємо на увазі його яскравість. Це властивість повідомляє нам, наскільки світло світлий або темний, в тому сенсі, наскільки він близький до білого. Наприклад, жовтий колір вважається світліше синього «navy blue», який в свою чергу сам світліше чорного. Таким чином, значення (value) канаркового жовтого вище, ніж синього «navy blue» і чорного.

Рисунок 4.42 – Низька яскравість, постійна – однаковий рівень яскравості.



Рисунок 4.43 – Контраст яскравості – сірий = ахроматичний.



Рисунок 4.44 – Контраст яскравості – повне відмінність яскравості.

Яскравість (використовується термін «value» або «brightness») залежить від кількості світла, випромінюваного кольором. Найпростіший спосіб запам'ятати це поняття – це уявити собі шкалу сірого кольору, зі зміною чорного на білий, в якій містяться всі можливі варіанти монохроматичного сірого кольору. Чим більше в кольорі світла, тим він яскравіше. Таким чином

пурпурний – менш яскравий, ніж небесно-блакитний, так як випромінює менше світла.



Рисунок 4.45 – Шкала сірого кольору

Цю шкалу сірого кольору можна прирівняти до кольорової шкали за допомогою того ж рівняння, яке використовується в телебаченні (Яскравість сірого кольору = 0.30 Red + 0.59 Green + 0.11 Blue):



Рисунок 4.46 – Спектр

4.6. Хроматичність та ахроматика

Про хроматична ми говоримо, коли розмірковуємо про «чистоту» кольору. Це властивість кольору говорить нам, наскільки він чистий. Це означає, якщо в кольорі відсутні домішки білого, чорного або сірого, колір має високу чистоту. Ці кольори виглядають живими і чистими. Поняття «хроматична» пов'язано з насиченістю. І його часто плутають з насиченістю. Однак ми будемо продовжувати використовувати ці терміни окремо, тому що на наш погляд вони відносяться до різних ситуацій, про що буде сказано трохи нижче.



Рисунок 4.47 – Висока хроматичність – дуже сяючі, живі кольори.



Рисунок 4.48 – Низька хроматичність – ахроматичні, безбарвні кольори.



Рисунок 4.49 – Хроматичність однакова – середній рівень. Та ж активність кольорів незважаючи на різний тон; чистота менше, ніж у зразків вище.

Високохроматичні кольори містять максимум власне кольору з мінімальними або нульовими домішками білого, чорного або сірого. Іншими словами, рівень безробіття домішок інших кольорів в конкретному кольорі характеризує його хроматична. Хроматична, яку часто називають «соковитістю», є кількістю кольору (hue) в кольорі. Колір без кольору (hue) є ахроматичності або монохроматичності, і бачимо як сірий. Для більшості кольорів, у міру збільшення яскравості збільшується і хроматична, за винятком дуже світлих кольорів.

4.7. Створення колірних схем

Одна з головних правил у виборі кольорів – це простота. Чим більше кольорів ви використовуєте, тим більш заплутаний і той, що викликає різні емоції, дизайн ви отримаєте в кінці.

Кольорові палітри, які включають в себе більше 5 кольорів найчастіше досить складно підтримувати. Подивившись на колірні схеми популярних брендів, ви зрозумієте, що вони найчастіше роблять все якомога простіше і використовують лише кілька кольорів.

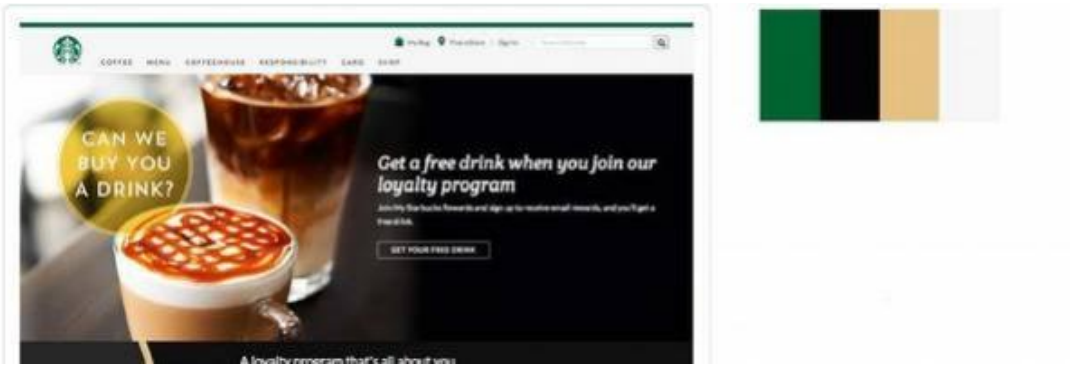


Рисунок 4.50 – Приклад добору колірної схеми

Найкраще використовувати в колірних схемах тільки 3 або 4 кольори. Це основний, вторинний, та акцидентний і колір заднього фону.



Рисунок 4.51 – Приклад добору колірної схеми

4.7.1. Основний колір.

Це ваш основний колір, який визначає ваш бренд і використовується для того, щоб привернути увагу. Дуже часто використовується для важливих заголовків, посилань, кнопок і навігації. Вам слід почати роботу над створенням вашої кольорової палітри з вибору основного кольору. Відтінок не повинен бути ідеальним з самого початку. Виберіть чистий колірний тон, який, на вашу думку, відмінно підійде до вашого проекту. Ви можете попрацювати над насиченістю і світлосилою трохи пізніше. Якщо ви працюєте над проектами для клієнтів, вони зазвичай вже мають основний колір, який був обраний, ґрунтуючись на їх бренді або перевагах. Ви в будь-якому випадку можете додати другорядний колір і створити всю колірну схему базуючись на цих кольорах.



Рисунок 4.52 – Приклад добору основного кольору

Припустимо це основний колір. Створимо колірну схему ґрунтуючись на ньому.

4.7.2. Вторинний колір.

Зазвичай цей колір менш помітний, ніж основний. Ви можете використовувати його в якості кольору заднього фону для елементів, які ви

хочете виділити в вашому дизайні. Краще використовувати темніші відтінки вторинного кольору для тексту тіла сайту або заголовків нижчого рівня. Існує дуже багато різних технік того, як необхідно вибирати другорядний колір. Якщо ви хочете, щоб ваша колірна схема виглядала монохромно, ви можете просто вибрати більш темний відтінок вашого основного кольору. Якщо ви хочете створити динамічний вид, виберіть щось більш контрастне. Просто пам'ятайте, що другорядний колір не повинен бути сильніше і активніше, ніж основний або акцидентний колір.



Рисунок 4.53 – Приклад добору другорядного кольору, відносно до обраного основного

У першому прикладі було обрано темно-синій, щоб охолодити основний і мати більш сильний контраст. У другому прикладі було обрано більш темний відтінок, який виглядає як коричневий. Ця схема виглядає більш монохроматичною. В останньому прикладі я вибрав білий колір для створення контрасту з темним кольором.

4.7.3. Акцидентний колір.

Це дуже важливий колір в вашій кольоровій палітрі. Ми називаємо його акцидентним, тому він використовується для акцентів. Цей колір повинен розбити шаблон вашої колірної схеми і привернути увагу користувача. Ви не повинні зловживати ним в вашому дизайні. Його необхідно використовувати тільки в тому випадку, коли це дійсно необхідно. Так ви досягнете кращих результатів. Він не повинен поєднуватися з вашим основним і другорядним кольором, тому він повинен виділятися. Виберіть той колір, який має досить сильний контраст з основним кольором.

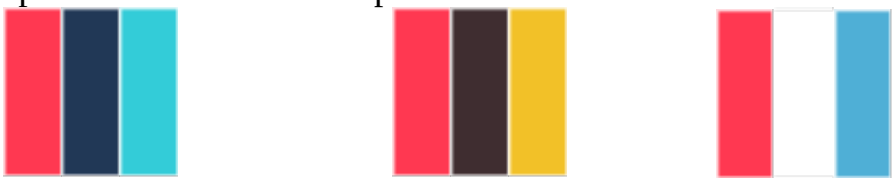


Рисунок 4.54 – Приклад добору акцидентного кольору, відносно до обраних кольорів

У першому і останньому прикладі було обрано синій в якості акцидентного кольору, тому він має відмінний контраст з основним кольором. У центральному прикладі обрано оранжево-жовтий, щоб зробити його трохи більш теплим.

4.7.4. Колір заднього фону.

Це основний колір заднього фону, який найчастіше заповнює величезні області вільного простору. Вам потрібно переконатися, що ваш основний, вторинний і акцидентний кольори бездоганно виглядають на ньому.

Колір заднього фону може бути білим, він є найбільш часто використовуваним кольором в веб-дизайні. Якщо основний колір теплий, варто

вибрати колір заднього фону в діапазоні від білого, до жовтувато-коричневого. З іншого боку, якщо основний колір холодний, зробити колір фону сірувато-синіми. Це додасть більше характерності кольоровій палітрі.

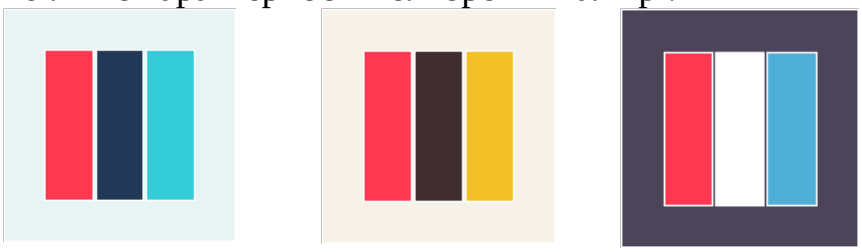


Рисунок 4.55 – Приклад добору акцидентного кольору, відносно до обраних кольорів

У першому прикладі було обрано світло-синій в якості базового кольору для того, щоб відповідати холодним кольорам цієї схеми. У другому прикладі світлий жовтувато-коричневий колір в якості основного. Він підтримує тепло і відповідає вторинному і акцидентному кольору. В останньому прикладі, було обрано темно-фіолетовий в якості мого базового кольору, і ви можете бачити, як інші кольори відмінно з ним контрастують. Останній крок роботи над колірною схемою полягає в її тестуванні спільно з типографікою, якщо постає питання переносу бренду компанії у друк. Для цього потрібно переконатися, що кольори, які були вибрані, відмінно виглядають при використанні разом з текстом. Перевірте як виглядають заголовок і параграфи з основним, другорядним і акцидентним кольорами, як вони будуть виглядати з заднім фоном нейтрального кольору.

Приклади веб-сайтів з вдалими колірними схемами

Насправді немає ніяких обмежень, коли справа стосується створення колірної схеми. Розглянемо приклади сайтів з прекрасним дизайном. Давайте поглянемо на їх колірні схеми.

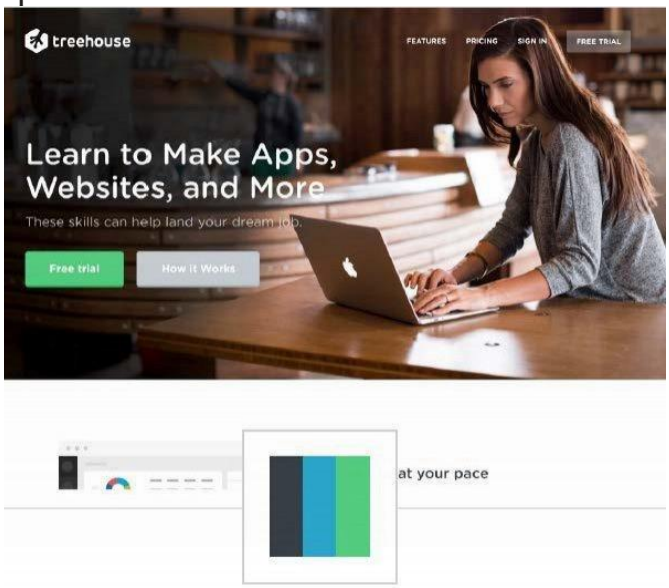


Рисунок 4.56 – Колірна палітра з величезною кількістю білого кольору на тлі

Темно-сірий є основним кольором. Він найчастіше використовується в друкарні і іноді в якості кольору заднього фону для поділу розділів сторінки.

Синій колір використовується для посилань і другорядних кнопок. Зелений колір є акцидентним кольором у даному дизайні і використовується тільки на кнопках призову до дії, які ведуть на сторінку реєстрації або на пробний період.



Рисунок 4.57 – sachagreif.com виглядає монохроматично

Кольори, які тут використовуються, знаходяться з одного боку колірного кола. Різні кольори заднього фону використовуються для поділу секції основного контенту від бічної панелі. Темно-коричневий колір використовується для всієї типографської частини, в той час як червоний колір зарезервований для посилань і кнопок. Насправді в даній колірній схемі немає акцидентного кольору, але бірюзовий міг би додати відмінний контраст в даному дизайні.



Рисунок 4.58 – Контрастна колірну схему greyp-bikes.com

Цей веб-сайт має дуже контрастну колірну схему. Жовтий та чорний створюють відмінний колірний контраст. Область основного контенту має білий колір, який покращує читаність тексту. Так його читати простіше, ніж, якби вона була жовтого кольору.



Рисунок 4.59 – Дизайн сайту InVisionApp.com

Дизайн сайту InVisionApp.com має задній фон основного контенту білого кольору. Це покращує зовнішній вигляд і полегшує процес читання.

Вся типографіка і інші області заднього фону мають різні відтінки сірого. Є активний акцидентний рожевий колір, який розриває шаблон всього сайту і робить кнопку заклику до дії набагато більш привабливою і цікавою. Так само виявив синій колір на деяких підлеглих сторінках, який використовується в якості вторинного кольору для кнопок і посилань.

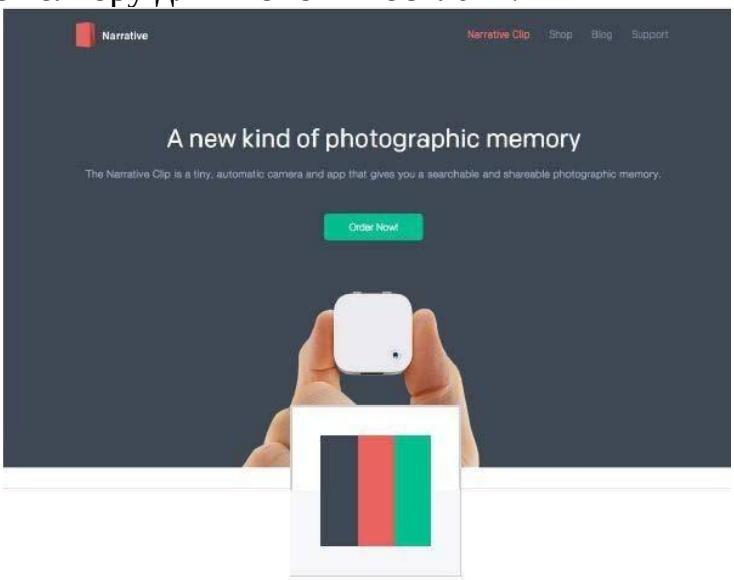


Рисунок 4.60 – getnarrative.com та його контрастна кольорова схема

Синювато-сірий колір використовується тут в якості основного для всього тексту і великих блоків заднього фону.

Він використовується для створення спокійного і холодного дизайну. Червоний колір використовується в якості другорядного, він робить сайт трохи активнішим і живим. І нарешті, зелений колір використовується для основних кнопок і виділяє їх. Це створює відмінний контраст з рештою дизайну.

Типографіка для користувацьких інтерфейсів

Типографіка – це мистецтво і техніка організації шрифтів. Типографіка включає в себе: вибір шрифтів, позиціювання букв в спеціально обраному місці, зміна розміру шрифтів та додавання пробілів між рядками (leading) і між буквами (tracking).



Рисунок 4.61 – Типографіка

Типографіка відіграє особливу роль в дизайні, тому це одночасно і форма мистецтва і спосіб передачі даних. Кожне слово має значення і, розробляючи дизайн, ми можемо донести наше повідомлення більш ефективно та поліпшити його сприйняття. З іншого боку поганий дизайн може нашкодити основній функції типографіки, яка полягає в зручності читання. Він може відволікати читачів від контенту. Суть типографіки не тільки у виборі шрифтів, що добре виглядають. Є дуже багато деталей і дрібних рішень, які вам необхідно буде прийняти в процесі розробки дизайну. Погляньте на зразок нижче і подивіться як важливо підібрати правильні розміри шрифтів, кольори шрифтів, інтервали між рядками (leading), відстань між літерами (tracking) тощо.

Робота над дизайном типографіки є досить широкою темою і випущено дуже багато книг, присвячених тільки цій темі. Ми сфокусуємось на типографіці веб-сайту і вивчимо базові техніки.

За роллю на сторінці шрифти можна поділити на:

- *Текстові (text).* Текстові шрифти використовують для великих фрагментів тексту в книгах та журналах, для набору основного тексту. Особлива увага в дизайні таких шрифтів приділяється зручності при читанні.
- *Акцидентні (display).* Акцидентні шрифти проектуються для великих розмірів и використовуються для набору заголовків.
- *Декоративні (decorative).* Декоративні шрифти широко використовують в рекламі, також можуть використовуватися для заголовків.

4.8. Типографіка для Інтернет

Чи важлива типографіка? Подивіться на будь-який веб-сайт або інтерфейс програмного забезпечення і уявіть, що ви прибрали весь текст з нього. На що він схожий?

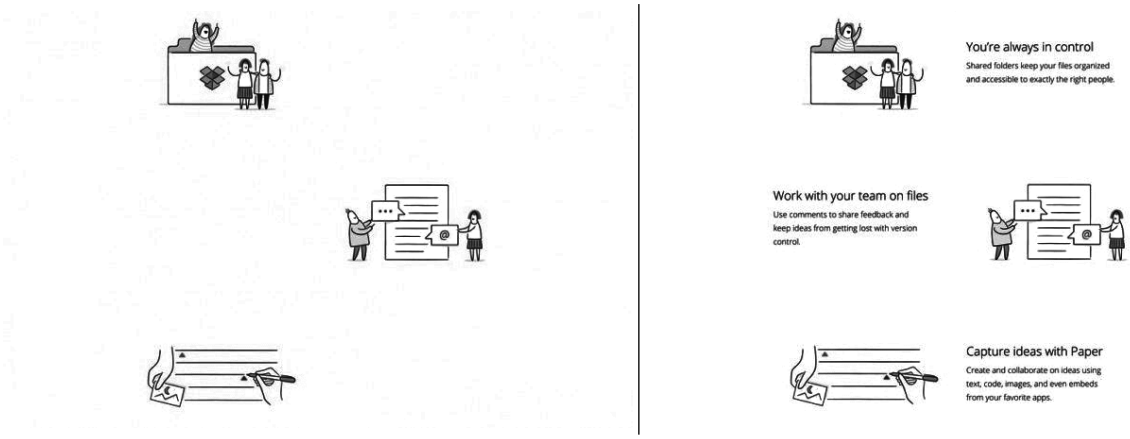


Рисунок 4.62 – Приклад фрагменту веб-сторінки Dropbox без тексту та з текстом

Як ви можете бачити не дуже багато залишилося після того, як ви позбулися всієї типографіки на веб-сайті. Правда в тому, що майже кожен веб-сайт на 90% складається з текстової інформації. Цей приклад з Dropbox показує, що графіка сама по собі не надає такий же рівень комунікації, і дуже двозначно висловлює сенс без текстових пояснень. Так що, важливо навчитись вибирати і поєднувати шрифти, застосовувати цікаві стилістичні техніки і створювати типографічну ієрархію для проектування макетів. Типографіка важлива, тому, починати процес роботи над дизайном доречно з вибору типографіки, перш ніж планувати макет або починати розробляти прототип. Це дозволяє оцінити скільки місця необхідно на веб-сайті і вибрати основний стиль для всього проекту. Розробка дизайну типографіки для інтернету це не те ж саме, що розробка типографіки для поліграфічного друку.

4.9. Візуалізація шрифтів.

Шрифти в макеті будуть виглядати практично однаково на будь-якій операційній системі, але коли ви використовуєте їх в коді HTML / CSS, то він може виглядати по іншому в залежності від браузера і операційної системи. Safari, Google, Firefox і Internet Explorer здійснюють рендерінг шрифтів по різному. Так само ви помітите різницю між Mac OS, Windows Vista, і Windows 7/8. Виглядає так, ніби ви не несете відповідальність за це, тому що ви відповідаєте тільки за розробку дизайну в графічному редакторі, тому варто бути дуже обережними з використанням занадто тонких або занадто жирних шрифтів. У деяких браузерах і операційних системах тонкі шрифти скоріше за все буде дуже складно прочитати при відключеному згладжуванні. З іншого боку, дуже жирний шрифт може виглядати занадто товстим з включеним згладжуванням.

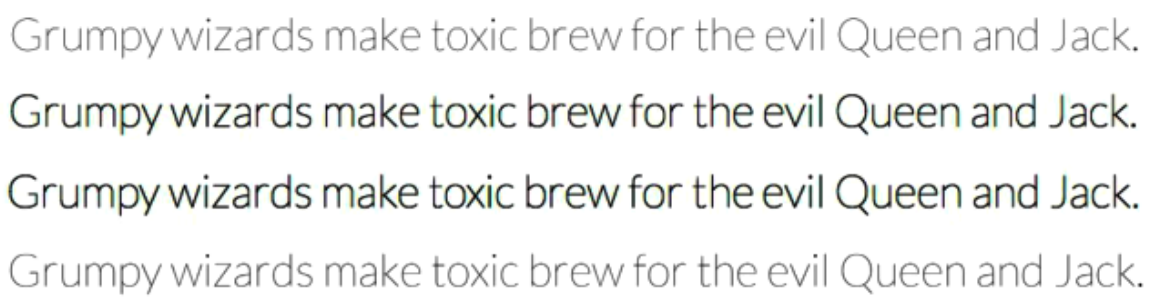


Рисунок 4.63 – Приклад відображення тексту в різних браузерах з включеним і вимкненим згладжуванням.

Завжди необхідно перевіряти різні стилі шрифтів прямо в браузері, перш ніж використовувати їх в дизайні шаблону. Таким чином, ви будете точно знати, що ваша типографіка так само добре виглядає в браузері, як і в проекті дизайну.

4.10. Адаптація

Оскільки текст повинен відображатися на маленьких екранах, ви так само повинні будете вибирати розміри шрифтів і висоти рядків пропорційно до розмірів екрану планшетів і мобільних телефонів.



Рисунок 4.64 – Різниця в типографіці для десктопної та мобільної версій

На прикладі вище, ви можете бачити різницю в типографіці між екраном мобільного пристрою і екраном персонального комп'ютера.

Розмір шрифту не дуже сильно змінюється в тілі сайту. Може бути все на пару пікселів, але дизайн заголовків може істотно змінюватися відповідно до розміру екрана пристрою, на якому запущено веб-сайт.

4.11. Класифікація шрифтів

Існують сотні шрифтів. Завдяки їх специфічним характеристикам ми можемо розділити їх на кілька основних груп:

4.11.1. Serif (З зарубками)

З зарубками – у цього типу присутні невеликі лінії в кінці букв. Такі шрифти набагато простіше читати в пресі, й це одна з причин, за якими вони є найпопулярнішими. Їх вважають за краще використовувати в книгах, журналах і газетах. Вони відмінно підходять як для тексту в тілі сайту, так і для великих заголовків. Шрифти із зарубками можуть додати витонченості та елегантності зовнішньому вигляду дизайну.



Рисунок 4.65 – Гарнітура Serif (З зарубками)

4.11.2. Sans serif (Без зарубок)

Букви без зарубок. Дуже схожі на шрифти із зарубками, шрифти без зарубок відмінно підходять як для тексту в тілі сайту, так і для заголовків. Їх вважають за краще використовувати на ПК, тому що їх набагато простіше читати при дрібних розмірах шрифту. Шрифти без зарубок можуть додати простоти і мінімалістичності вашому дизайну. На даний момент цей шрифт є популярним для тексту статей сайту.



Рисунок 4.66 – Гарнітура Sans serif (Без зарубок)

4.11.3. Slab serif (Прямокутні зарубки)

Прямокутні зарубки надають унікальності і стильності зовнішнім виглядом. Шрифт з прямокутними зарубками важко прочитати, якщо його використовують в тексті тіла сайту, але він відмінно підходить для заголовків. Прямокутні зарубки можуть зробити так, щоб ваші заголовки виглядали цікавіше і візуально відокремлено від решти тексту.

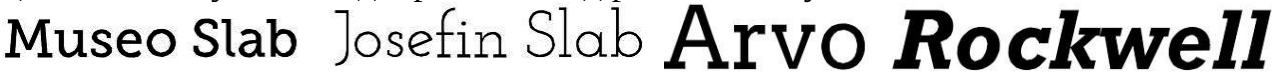


Рисунок 4.67 – Гарнітура Slab serif (Прямокутні зарубки).

4.11.4. Script (Пропис)

Шрифти прописом дуже елегантні і можуть використовуватися в якості додаткового декоративного елемента до дизайну. Так чи інакше, варто бути дуже обережним з їх використанням в якості основних шрифтів заголовків і, особливо, для тексту в тілі сайту, тому що їх досить складно читати.

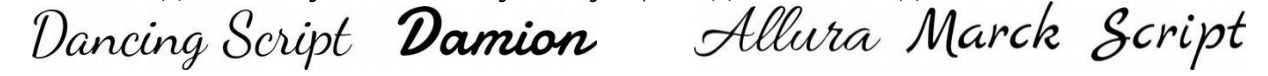


Рисунок 4.68 – Гарнітура Script (Пропис)

4.11.5. Blackletter (Готичний шрифт)

Готичний шрифт або старо-англійський шрифт виглядає дуже декоративно і може додати нотки старовини в дизайн. Непогано використовувати цей шрифт в дизайні середньовічних ресторанів чи то барів.

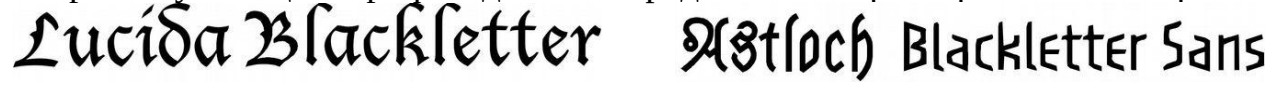


Рисунок 4.69 – Гарнітура Blackletter (Готичний шрифт)

4.11.6. Handwriting (Рукописний шрифт)

Рукописні шрифти імітують ручне написання тексту і вони дуже схожі на пропис. Вони можуть бути відмінним декоративним доповненням до вашого сайту. Рукописні шрифти так само чіпляють погляд через те, що мають свій унікальний стиль. Ви можете використовувати їх, коли вказується на якісь важливі дії, які необхідно зробити користувачам.



Рисунок 4.70 – Гарнітура Handwriting (Рукописний шрифт)

4.11.7. Decorative (Декоративний)

Декоративні шрифти включають в себе всі кумедні і унікальні шрифти. Деякі декоративні шрифти дуже складно прочитати. Деякі з них містять символи, які замінюють літери алфавіту. Вони можуть використовуватися в ситуаціях, в яких ви хочете досягти унікального зовнішнього вигляду.

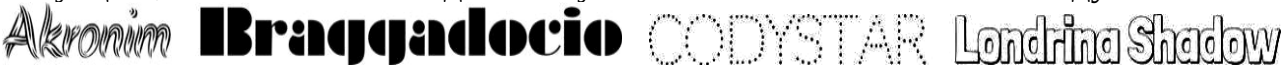


Рисунок 4.71 – Гарнітура Decorative (Декоративний)

4.12. Основні атрибути стилю тексту

Стиль тексту – сукупність всіх параметрів оформлення тексту, властивих даному його відрізку. Атрибути стилю тексту включають такі поняття:

- гарнітура шрифту;
- накреслення;
- кегль;
- інтерліньяж:
- міжлітерний просвіт;
- міжсловний пробіл;
- виключка:
- відступ першого рядка;
- втяжка (відступи справа і зліва);
- міжабзацне відбиття;
- інші прийоми оформлення.



Рисунок 4.72 – Основні атрибути шрифту



Рисунок 4.73 – Базова лінія, медіана, х-висота

4.12.1. Базова лінія

Базова лінія – це лінія, над якою розташована більшість символів. Уявна лінія, що проходить по нижньому краю основного елемента символу, називається базовою лінією. Літера ніби лежить на цій лінії. У звичайному випадку базові лінії символів рядка знаходяться на одному рівні. Проте нинішнє становище базової лінії в деяких програмах можна вільно змінювати. За замовчуванням зсув базової лінії одно 0. Воно вимірюється в пунктах, проте ви можете вводити значення в будь-яких одиницях, вони будуть автоматично

переведені в пункти. Позитивне значення зсуву відповідає підйому символу щодо інших, негативне — опускання.

4.12.2. Медіана

Медіана, з іншого боку, це "стеля" для більшості малих літер.

4.12.3. X-висота

Відстань між базовою лінією і медіаною називається x-висотою. Вона дорівнює висоті малої літери "x", звідси і її назва. Чим більше x-висота в гарнітурі, тим більше читабельна буде вона в дрібних розмірах. Велика частина тексту, який ми читаємо в західному світі, складається з малих літер, і x-висота дуже впливає на читабельність. Більше простору дає шрифту трохи більше можливостей своїм буквах виділятися. Люди читають стрибкоподібно, і поглинають букви швидко і несвідомо. Тому велика x-висота краще для основного тексту, тому що довгі абзаци можуть бути нудними для очей. Розділи, як заголовки, де зазвичай ви будете використовувати великі розміри шрифтів і більш короткий текст, можуть мати гарний вигляд і з меншою x-заввишки. Тут у вас більше варіантів художнього вибору без істотного погіршення читабельності.

4.12.4. Серіф (зарубки)

Серіф – це зарубка на кінці літери. Літери шрифту можуть мати серіф, а можуть і не мати. Вважається, що текст, в якому використані серіфні шрифти, читається краще. У безсерiфних (рубаних) шрифтах букви мають рівні краї і не мають зарубок. Як правило, вони використовуються для оформлення заголовків газет або книг. Також вони застосовуються в текстах, набраних дрібним шрифтом, оскільки має чітку (рубану) форму. Не слід застосовувати рубані шрифти при наборі довгих текстів, оскільки при їх читанні очі швидко втомлюються.

4.12.5. Пуансон

Пуансон – це внутрішній простір літери. Зовнішній вигляд шрифту в великій мірі визначається величиною пуансона. Чим він більший, тим більше світлим і великим здається шрифт. Наприклад, літери, для яких задані одні і ті ж кегль і накреслення, але різний розмір пуансона, здаються різними за величиною. Необхідно пам'ятати, що занадто великі або занадто маленькі пуансони погіршують читаність тексту.

4.12.4. Гарнітура шрифту

Гарнітура шрифту – сукупність шрифтів, об'єднаних спільними стильовими ознаками, відмінними від інших шрифтів. Це як би сім'я шрифтів (сукупність накреслень, об'єднаних загальним характером графічного побудови знаків і рішенням їх елементів). Як у людей є сім'ї, так і у шрифтів є сім'ї – Times і Arial, тобто сімейства варіантів шрифту з загальними стильовими особливостями знаків, що має власну назву. Отже, під гарнітурою розуміються варіації одного шрифтового сімейства, що відрізняються різною насиченістю, пропорціями, нахилом і стилем. Деякі гарнітури мають у своєму розпорядженні великою кількістю накреслень, ніж інші. Завдяки цьому можна побудувати весь документ на одній гарнітурі, використовуючи, де це необхідно різні варіанти

накреслень. Прикладом може служити шрифт Helios який налічує близько 33 накреслень. Термін "Шрифт" означає більш вузьке поняття – набір символів певної гарнітури, розміру і накреслення. Однак часто терміни "гарнітура" і "шрифт" вживаються як синоніми. Наприклад, гарнітура Arial з наступних варіантів накреслення шрифтів: Arial Bold, Arial Italic, ArialNarrow тощо.

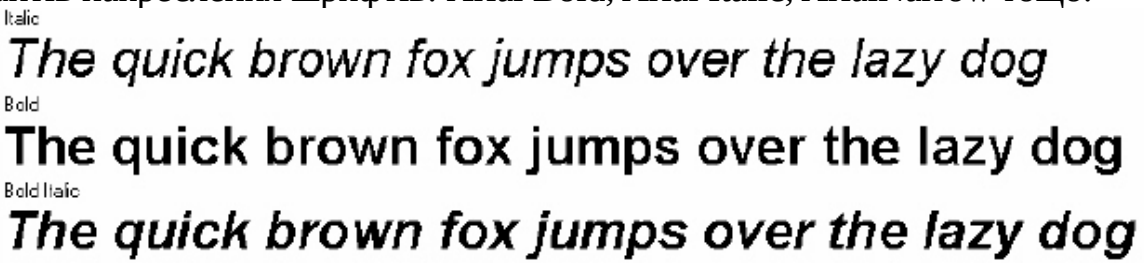


Рисунок 4.74 – Варіанти накреслення одного шрифту

Багато з нині використовуваних гарнітур створені досить давно (наприклад, "Академічна" використовувалася для набору книг ще в минулому столітті), інші придумані недавно. Всі елементи в хорошому шрифті гармонійно поєднуються, утворюючи справжній витвір мистецтва, виконаний художником-графіком.

4.12.5. Кегль

Розмір (висота) шрифту визначається так званим кеглем – відстанню між верхньою і нижньою шрифтовими лініями. У широко розповсюдженому розумінні кегль шрифту – це висота області, відведеної під літеру. Іншими словами це слово позначає висоту символів шрифту, тобто відстань від нижнього до верхнього виносного елемента плюс так звані заплечики – вільний простір над і під літерою. Це вільний простір визначається в самому шрифті і потрібно для того, щоб символи верхньої і нижньої рядків не зливалися. Все це разом становить кегль, що задається в програмі.

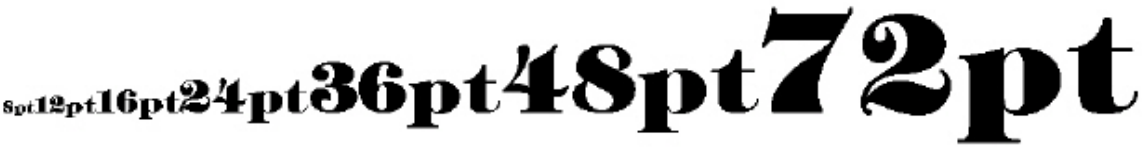


Рисунок 4.75 – Кілька кеглів однієї гарнітури

4.12.6. Leading (інтерліньяж)

Стосовно до настільним видавничим системам інтерліньяж – це відстань між базовими лініями двох сусідніх рядків. Інтерліньяж вимірюється в пунктах і складається з кегля шрифту і відстані між рядками. Наприклад, кегль 12 пунктів при відстані між рядками в 2 пункти називають кеглем 12 пунктів при інтерліньяж 14 пунктів, і такий інтерліньяж пишеться 12/14. Відстань між рядками (інтерліньяж) задається для всього абзацу. Стандартним вважається інтерліньяж, величина якого становить 120% величини кегля.

Інтерліньяж – міжстрочна відстань (відстань між базовими лініями сусідніх рядків). Воно грає практично таку ж роль, як і трекінг, тобто з його допомогою ви можете ущільнювати строки тексту по вертикалі. Іноді це зменшення дозволяє, з одного боку, зменшити розмір площі займаної даними абзацом тексту, а з іншого боку, через це ущільнення абзац виділяється на

загальному тлі (тому що виглядає більш темним). Рекомендується робити інтерліньяж на 1-2 пункту більше кегля шрифту.



Рисунок 4.76 – Інтерлін’яж

Інтерліньяж – відносна величина, він пропорційний кеглю символів. Загальна висота рядка дорівнює кеглю найбільшого шрифту, використаного в ній, плюс невелика відстань. Це і є стандартний інтерліньяж, що задається програмою за замовчуванням. Маніпулюючи величиною інтерліньяжу і приводячи її до дуже маленьким значенням, можна домогтися часткового накладення наступної рядка на попередню. Інтерліньяж вважається негативним, коли він менше, ніж розмір шрифту в рядку, наприклад, якщо інтерліньяж становить 20 пт при розмірі шрифту 24 пунктів. Негативний інтерліньяж при верстці звичайних текстових документів використовується відносно рідко. Але можливість використовувати негативну величину безумовно корисна при оформленні різних логотипів, заголовків, колонтитулів, ілюстрацій і т.д.

4.12.7. Кернінг та трекінг

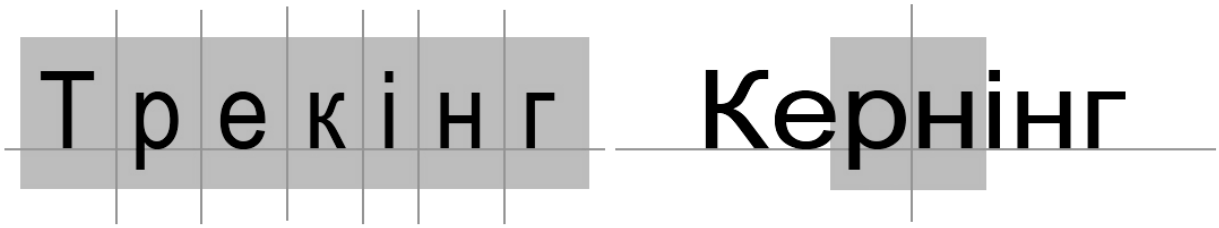


Рисунок 4.77 – Трекінг та кернінг

Трекінг – це рівномірний зміна відстаней між символами у всьому текстовому блоці. Тобто зміна його щільності.

Кернінг – це коректування відстані між окремою парою знаків.

Кернінг – це настройка мікро-просвітів між окремими буквами. Підбір відбувається по-різному для кожного дизайну, спираючись на потреби кожної літери. Рядкова "про" займе менше місця, ніж рядкова "t", наприклад. У веб-дизайні кернінг дуже складно налаштовувати, хоча в друкованої індустрії ця практика дуже поширена.

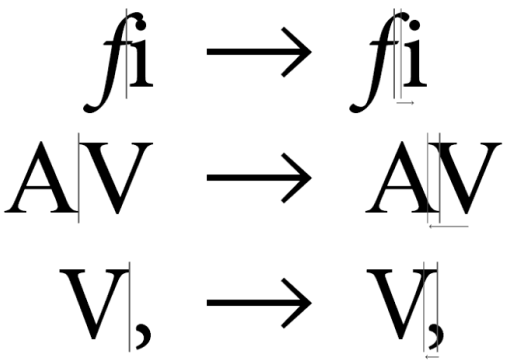


Рисунок 4.78 – Трекінг та кернінг

Можливо застосувати кернінг, спираючись на дані, закладені в самі шрифти. Ця опція підключається за допомогою текстового рендерінга або параметрами шрифтів в CSS. Пам'ятайте, що ці параметри підтримуються далеко не скрізь.

Налаштування цих параметрів може допомогти з проблемою висячих рядків і висячих прийменників. Кернінг регулюють для складання заголовків, дизайну логотипів, для надання тексту певної стилістики. Якщо слово набрано великим кеглем, варто іноді трохи зменшити відстані між буквами, щоб вони не розліталися і не порушувалася цілісність рядки. Якщо букви маленькі, можна трохи збільшити ці відстані, щоб дати більше повітря і полегшити візуально текст. Існує поняття «кернінгові пари». Це пари знаків, відстань між якими доводиться іноді регулювати вручну через особливості їхнього накреслення.

4.13. Вибір і поєднання шрифтів

Досить тривалий проміжок часу ця опція була недоступна при створенні веб-сайтів. Навіть якщо ви і могли вибрати будь-який шрифт для проекту в середовищі графічного редактору, ви повинні були використовувати тільки "веб-сумісні" шрифти, які були найпопулярнішими і були присутні в більшості операційних систем. Так ви могли бути впевнені, що веб-сайт буде коректно відображатися в браузері.

Ці часи вже в минулому, і всі сучасні браузери підтримують правило @ font-face в CSS, яке дозволяє вам використовувати шрифти, які за умовчанням не встановлено в операційній системі відвідувача вашого сайту.

Так чи інакше, це не означає, що ви можете використовувати будь-які шрифти, які забажаєте. Якщо ви хочете використовувати ваш власний шрифт за допомогою правила @ font-face, то вам необхідно завантажити цей шрифт на сервер. Це означає, що ви дасте іншим користувачам доступ до нього, і вони зможуть завантажити його. Більшість шрифтів мають ліцензії, і ви не маєте право поширювати їх вільно. Але існують як платні, так і безкоштовні сервіси, які надають доступ до своєї бази шрифтів і ви можете використовувати їх на ваших веб-сайтах, завантажуючи їх з цих сервісів. Дуже багато з цих шрифтів так само оптимізовані для кращого відображення в браузерах, що так само допомагає зробити дизайн ваших сайтів більш привабливим. Через величезного розмаїття шрифтів існує нескінченна кількість комбінацій, і вибір вірних шрифтів може зайняти багато годин. Це не просто, але ви завжди повинні

засновувати свої рішення на керівництві по стилю вашого проекту і перевагах аудиторії. По-перше, ви повинні почати з вибору основного шрифту для тіла сайту, який буде виступати основним шрифтом для всього проекту. Найчастіше шрифт тексту тіла сайту вибирається або із зарубками, або без зарубок. Це гарантує читаність вашого сайту, так що ви можете зробити невеликий пошук тільки в цих двох групах (тут існує всього декілька тисяч варіантів).

Дуже важливо те, щоб шрифт, який ви вибрали, мав в своєму арсеналі різні стилі, такі як жирний, підкреслений, курсив тощо. Це дасть вам більше гнучкості при їх використанні в вашому проекті. Протестуйте шрифт тіла сайту на великих блоках тексту в різних ситуаціях, такі як списки, параграфи, маленькі і великі розміри шрифту і тощо.

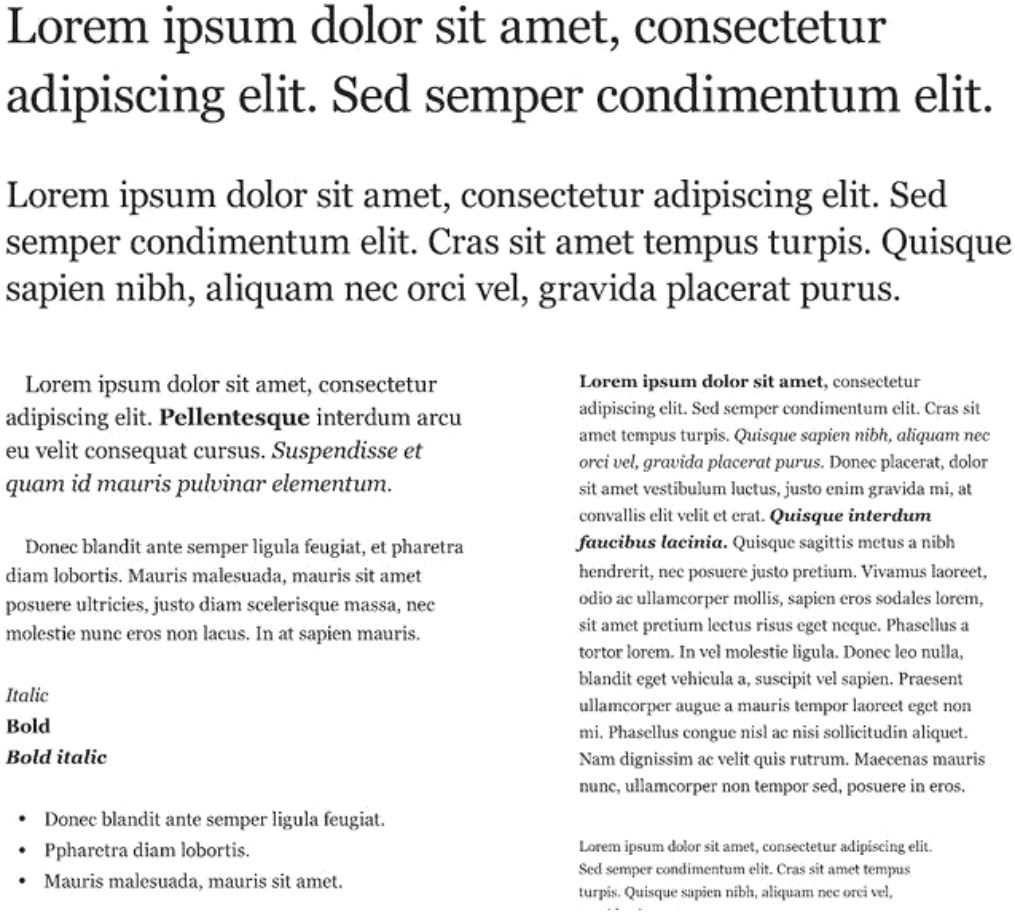


Рисунок 4.79 – Використання системи з різних накреслень шрифту у межах великих блоків тексту

Шрифт тіла сайту повинен бути простим у використанні і універсальним. Протестуйте ваш шрифт тіла сайту в різних ситуаціях і з різними стилями.

По-друге, виберіть ще один шрифт для заголовків. Хоча і не існує якогось правила для цього, але рекомендується вибирати шрифт відмінний від шрифту тіла сайту. Так ваші заголовки будуть більш привабливими. Звичайно ж, ви можете досягти відмінних результатів, якщо будете використовувати шрифт тіла сайту, але злегка змінивши його параметри, припустимо зробивши його трохи більш жирним. Ви можете змішувати групи шрифтів і з'єднувати заголовки із зарубками з текстом тіла сайту, який без зарубок. Ці дві групи шрифтів досить непогано поєднуються і з їх допомогою можна створити досить цікавий і

гарний дизайн типографіки. Протестуйте шрифти заголовків, використовуючи короткі речення та шрифти великого розміру.



Рисунок 4.80 – Заголовки без зарубок разом з текстом із зарубками з лівого боку. Заголовки із зарубками разом з текстом без зарубок з правого боку.

По-третє, ви можете додати ще один шрифт до вашого дизайну, який буде використовуватися для акцентування уваги на якихось конкретних елементах. Робіть їх більш помітними. В даному випадку ви можете використовувати декоративний шрифт або рукописний шрифт тому вони добре привертають увагу.



Рисунок 4.81 – Додавання декоративної гарнітури у якості акцидентного шрифта

Кращі варіанти використання і поєднання шрифтів:

- Зручність читання на першому місці.

Який би шрифт ви не вибрали для тексту в тілі сайту і основних заголовків, пам'ятайте, що вони повинні легко читатися. Звичайно, читаність деяких шрифтів може залежати від того як вони налаштовані: вірний розмір шрифту, висота рядка і колір. Перш ніж використовувати шрифти в вашому дизайні, ви можливо хочете подивитися на те, як вони будуть вести себе в різних ситуаціях і як вони будуть виглядати в різних кольорах. Так ви зможете переконатися, що вони відмінно підходять для потреб вашого проекту.

Деякі шрифти складно читати (подивіться наліво), але ви завжди можете виправити це (подивіться направо). Шрифти з обох сторін однакові, але використовується різна стилістика, яка покращує читаність. У цьому правилі є кілька виключень; наприклад, якщо ви працюєте над дизайном веб-сайту для дітей, ви можливо оберете декоративний шрифт, який сподобається дітям. Суворо дотримуйтеся шрифту з зарубками і шрифту без зарубок адже ці шрифти простіше читати тому вибір декоративного шрифту має бути не найкращим рішенням.

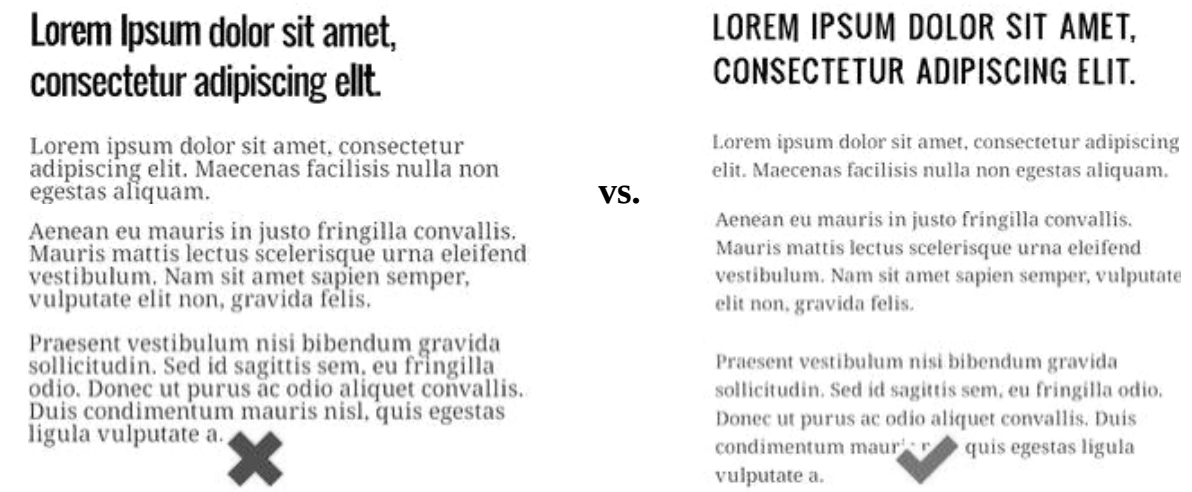


Рисунок 4.82. Приклади невдалого та вдалого використання і поєднання шрифтів

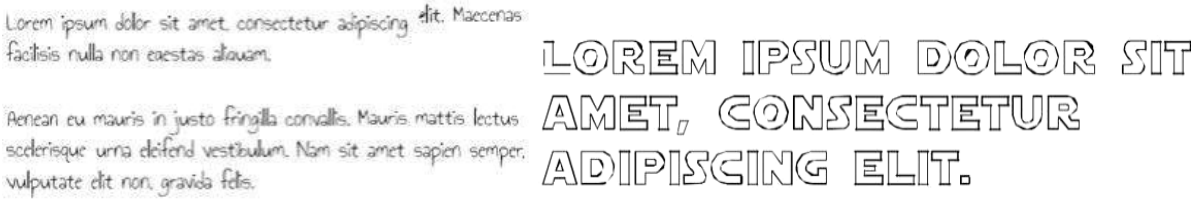


Рисунок 4.83 – Порівняння зручності читання шрифтів

Читання порожніх заголовків і вручну-написаного тексту тіла може бути досить складним завданням. Діти, можливо, не будуть заперечувати цей заголовок в стилі Star Jedi і текст в тілі сайту виглядають так, немов вони самі написали його. Пам'ятайте, що ви розробляєте дизайн для аудиторії, а не для себе або для інших дизайнерів.

- **Не використовуйте занадто багато шрифтів.**

Найкраще використовувати не більше 3-х різних шрифтів (можливо третій шрифт буде в якості опції, він буде використовуватися для відображення особливих елементів) під час роботи над дизайном проекту. Існує як мінімум кілька причин для цього: Якщо ви використовуєте один з сервісів шрифтів, як наприклад Google Fonts або Typekit, кожен новий доданий шрифт збільшує швидкість завантаження сайту. Занадто багато шрифтів починають відволікати, і відвідувачам буде важко сфокусуватись на читанні контенту. Підтримка менше трьох шрифтів вимагає набагато менше часу і при цьому сайт виглядає відмінно.

- **Будьте одноманітними.**

Однаковість дуже важливо, коли справа стосується вибору, змішування і використання шрифтів. Виберіть шрифт та потім використовуйте його протягом усього процесу створення дизайну. Використовуючи однакові шрифти для створення шаблону сайту, ваш дизайн буде не тільки простіше читати, але він так само буде виглядати більш професійно і одноманітно. Завдяки цьому ваш сайт буде більш гармонійний. Ви можете піти далі і використовувати однакові шрифти на всіх міні-сайтах або лендінгах, які належать одній компанії.

- **Уникайте незручних комбінацій.**

Комбінування двох шрифтів, які підходять для заголовків або змішування величезної кількості декоративних шрифтів – це погана ідея. Ці комбінації досить ризиковані і можуть викликати безліч проблем в вашому дизайні. Під час перегляду шрифтів на сайті Typekit.com ви можете фільтрувати їх за рекомендаціями для "параграфів" або "заголовків", що так само може бути корисно.



Рисунок 4.84 – "Futura PT Heavy" виглядає відмінно, як для шрифту заголовків, але занадто жирний для використання в якості шрифту тіла сайту.

4.13.1. Вибір шрифту для екранів мобільних пристроїв

Привабливa і ефективна типографіка – це ключ до ефективного дизайну веб-сайтів. Це особливо важливо, якщо говорити про мобільні пристрої.

Чи можна припустити, що вдало вибрані шрифти для екрану десктопної версії будуть працювати також і на мобільному пристрої? Є багато типографських стилів і методик, які ідеально підходять для великих екранів, але не працюють і створюють проблеми на невеликих, мобільних екранах.

Це означає, що для забезпечення найкращого досвіду відвідувачів з мобільних девайсів, ми повинні розглянути мобільну типографіку і переконатися, що незалежно від того, який вид пристроїв люди використовують, сайти будуть виглядати з привабливим дизайном, яким легко користуватися і приємно читати.

Типографіка повинна розглядатися як важливий елемент для створення сайту, що адаптується. Щоб досягти цього, ми повинні зрозуміти, як люди читають з мобільних пристроїв.

Як люди читають з мобільних пристроїв

Прийдіть в місцеве кафе, ресторан або в будь-яке інше місце, де багато відвідувачів, які використовують мобільні пристрої і ноутбуки. Спостерігайте їх у дії. Ви помітите значну різницю у відстані між користувачем і пристроєм. Відстань буде залежати від типу пристрою, який використовують для читання.

Придивившись, одна з перших речей, яку ви помітите при спостереженні за читачами з мобільними пристроями – це їх близькість до екрану. Люди мають набагато ближчі відносини зі своїми мобільними пристроями, ніж з їх офісними

або домашніми комп'ютерами. З мобільного екрану текст читається на відстані 12,6 дюймів, а з монітора ПК – на відстані 20-23 дюйма від екрану.

Так як екран мобільного пристрою менше, його потрібно підносити ближче, щоб легко і ефективно читати слова з такого екрану. Це означає, що користувачі мобільних фізично набагато ближче до пристрою, як і на зовнішній вигляд вашого сайту. Ключ мобільного типографіки в читаності. Якщо користувач не може прочитати, значить ваш дизайн не працює.

Освітлення і рух

Інший фактор, який вам потрібно дотримуватися при спостереженні за людьми читають з мобільних телефонів – зовнішні умови, в яких вони це роблять. Освітлення, рух і доступ в інтернет – всі ці фактори важливі. Освітлення – дуже важливий фактор, який на пряму пов'язаний з тим, як легко глядач зможе читати з екрану.

Що стосується руху, згадайте про рух по шосе проїжджаючи повз рекламних щитів. У типографіки концепція все та ж: вловити і втримати увагу читача.

Цікавими також є фізичні бар'єри, з якими люди повинні боротися за сумісного використання бездротових пристроїв, а саме тримаючи їх руками і край самого екрану. Коли люди використовують настільний комп'ютер для читання онлайн-контенту, вони використовують миша або клавіатуру. На мобільних сенсорних екранах палець виступає в якості навігації і він обов'язково блокує частину екрану.

Основи адаптивної типографіки

Типографіка не повинна бути «одного розміру, що підходить під всі девайси». У вас повинен бути адаптивний дизайн. Це означає, що ви можете на сайті використовувати один набір типографських стилів для великих екранів, а потім скорегувати ці стилі так, щоб вони підходили під невеликі екрани. Мета полягає не в створенні єдиного стилю для всіх розмірів екрану, а, скоріше, в створенні постійного позитивного досвіду читання, незалежно від розміру екрана.



Рисунок 4.85 – Приклад адаптивності у типографіці

При розробці адаптивної типографіки, є деякі ключові області, на які слід звертати увагу, для забезпечення читабельності:

- Шрифт
- Стилі CSS (вирівнювання, розмір, довжина лінії тощо.)
- Час завантаження

Переважні мобільні шрифти

Шрифти, які ви використовуєте у вашому дизайні будуть відігравати важливу роль в тому, наскільки легко сайт буде читатися на мобільних пристроях.

Занадто складні і тонкі літери не тільки важко читати, особливо тим, у кого порушення зору, але також такі літери мають тенденцію спотворюватися при менших розмірах. Це означає, що при масштабуванні шрифту для мобільних екранів виникають проблеми з читабельністю. Щоб уникнути цієї проблеми, багато дизайнерів вирішили використовувати шрифти без зарубок для мобільних веб-сайтів і додатків. Прості, прямі форми букв, як правило, краще масштабуються.

Різні шрифти для різних екранах?

При виборі шрифтів для вашого сайту, вирішите, чи хочете ви, щоб ваш сайт використовував той же шрифт на всіх розмірах екранів або вам зручніше використовувати різний шрифт для різних екранів.

Якщо ви хочете використовувати той же шрифт на всіх екранах, ви повинні звертати увагу на ті шрифти, які здатні підлаштовуватися до масштабу.

Якщо ви готові налаштувати вибір шрифтів в міру необхідності, ви можете використовувати шрифти із зарубками або більш складні шрифти для великих дисплеїв, а потім замінювати їх більш простими шрифтами для невеликих екранів.

4.13.2. Стилі типографіки

За допомогою кількох рядків CSS, ви можете кардинально змінити зовнішній вигляд вмісту текстового сайту, в тому числі розмір, висоту рядка, довжину рядка і багато іншого. Всі ці стилі грають важливу роль в зручності читання сайту і мають велике значення для мобільних екранів. Два найбільш важливих рішення при виборі відповідного мобільного шрифту – це розмір і простір.

4.13.3. Розмір

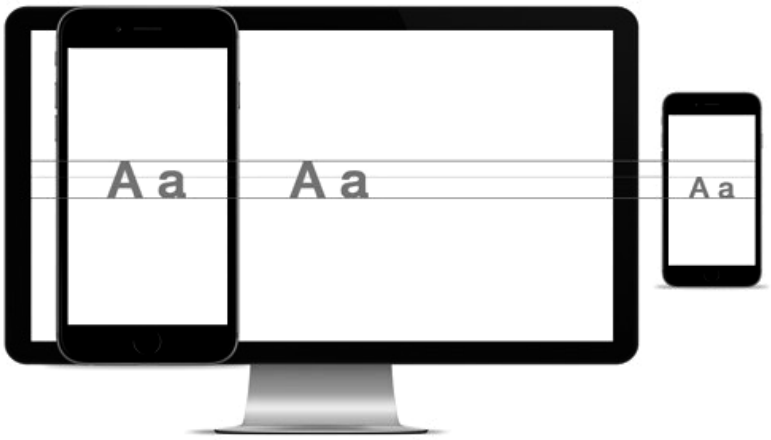


Рисунок 4.86 – Розмір гарнітури повинен залежати від розміру пристрою

З тих пір, як різні шрифти інтерпретуються в різних розмірах, за замовчуванням немає розміру шрифту, який необхідно використовувати для екранів мобільних телефонів. Хоча все, що менше 16 пікселів, стає складним для читання на будь-якому екрані. А надто великий текст буде створювати незручні розриви (те ж саме стосується переносів, які веб-дизайнери рідко використовують.) Для мобільних макетів розміри шрифтів, як правило, встановлюються в em, а не в пікселях, так що розмір шрифту є відносним, готовим реагувати на різні параметри екрану.

4.13.4. Інтервали

Коли справа доходить до невеликих екранів – особливо на вертикальних пристроях – інтервали вкрай важливі. Подумайте про умови з якими користувачі борються: освітлення, закривання екрану пальцями тощо. Внесення додаткового простору між буквами і навколо тексту має вирішальне значення.

Маленькі екрани мобільних пристроїв вже ускладнюють читання рядків тексту, розташованих дуже близько один до одного. Висота рядка, що працює на настільних екранах може бути складною для читання на мобільних телефонах.

Такі речі дуже динамічні, так що різні шрифти можуть знадобитися різні конфігурації. Почніть з висоти лінії 1,5. Це дасть читачам відчуття легкості і легкої читання тексту.

4.13.5. Довжина рядка

Не тільки розмір і простір, але також і розташування тексту може надати величезний вплив на загальний досвід читання на мобільному телефоні. Текст з дуже короткою рядком зменшує нормальний потік читання. Нам дуже важко читати статті, де довгі рядки. Також нам незручно читати пости з дуже короткими рядками. Тому тут потрібно шукати золоту середину. Нижче невелика схема, яка нагадує про оптимальну довжину рядка для набірного тексту.

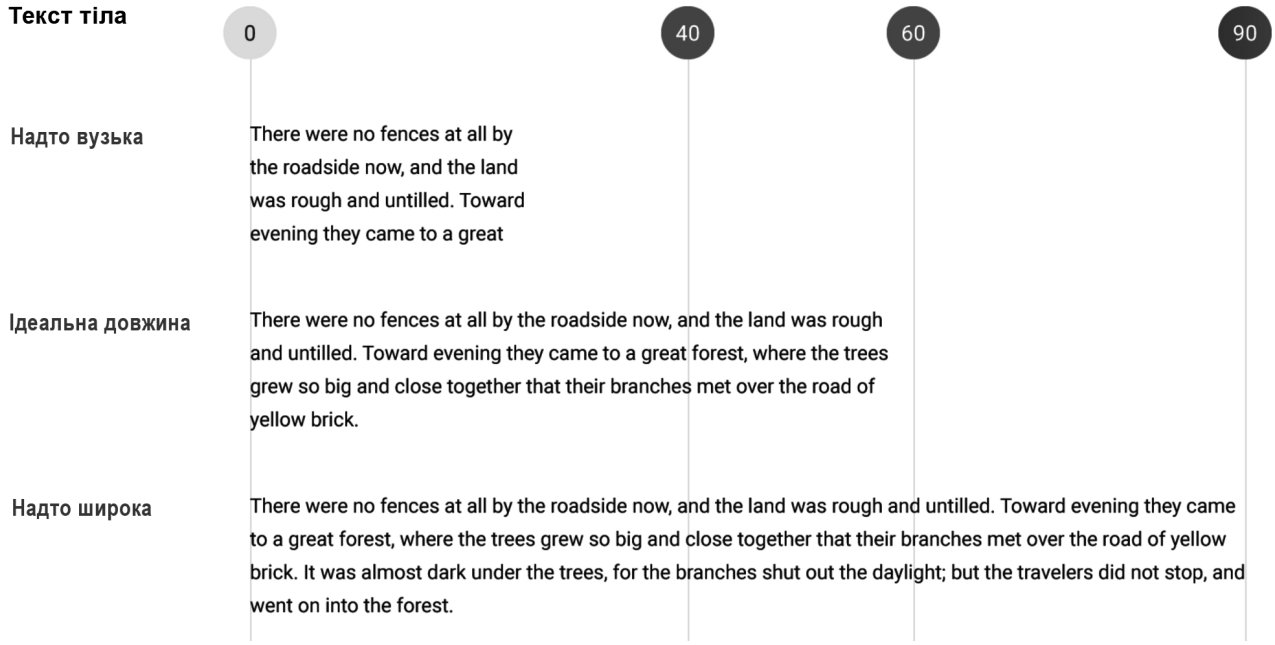


Рисунок 4.87 – Визначення оптимальної довжини рядка

В ідеалі довжина рядка для великих екранів повинна бути близько 65 слів або близько до того. На невеликих пристроях це число повинне бути набагато менше. Газети вже давно прагнули до довжини рядка близько 39 слів в прямих колонках. Це хороший приклад для мобільних пристроїв з вузькими колонками на сайті.

4.13.6. Вирівнювання



Рисунок 4.88 – Використання вирівнювання у відповідності до особливостей використання пристроїв

За замовчуванням веб-сторінки вирівнюють текст по лівому краю. Багато дизайнерів в їх дизайні люблять налаштувати вирівнювання по ширині. У деяких випадках такі стилі можуть працювати дуже добре, але вони часто не працюють на мобільних телефонах.

Для мобільних екранів, які за умовчанням вирівнюють по лівому краю – це часто кращий вибір. За текстом, вирівнюється по лівому краю, легше стежити. Але через малу ширину в більшості випадків колонки тексту на мобільних пристроях можуть створювати незвичайні прогалини в змісті.

Навіть якщо ви використовуєте різні вирівнювання в вашому дизайні на великих дисплеях, ви, можливо, захочете повернутися до вирівнювання по лівому краю на мобільних екранах для поліпшення читабельності.

4.13.7. Продуктивність і функціональність

Останнім чинником в мобільній типографіці вашого сайту є загальна продуктивність і функціональність самого сайту. Те, як швидко сайт завантажується для відвідувачів і як він насправді працює.

Щодо продуктивності, мобільні користувачі з ненадійним з'єднанням мережі будуть страждати від повільних завантажень. Якщо ваш сайт використовує безліч веб-шрифтів – це може негативно впливати на його продуктивність.

4.13.8. Веб безпечні шрифти

Веб-безпечні шрифти можуть бути не такими привабливими, як багато хто з доступних сьогодні варіантів веб-шрифтів, але вибір таких шрифтів покращує продуктивність, так що не скидайте їх з рахунків. Особливо це корисно для мобільних сайтів.

4.13.9. Дизайн до якого можна доторкнутися

Багато з принципів гарної типографіки застосовуються також до мобільного типографіки. Використання шрифтів, які легко читати, з належним інтервалом і чіткими літерами – це необхідно для будь-якого екрану. Ви повинні також розглянути унікальні характеристики цих пристроїв, то, як люди взаємодіють з ними і в різних середовищах. При розгляді типографіки на мобільних пристроях, потрібно зрозуміти, що користувачі будуть стосуватися цього тексту своїми пальцями.

4.14. Рекомендації по роботі з версткою

За чим стежити і над чим працювати в тексті. Основні проблеми верстки та їх рішення Заливаючи текст, дизайнер або верстальник повинен знати, про що текст, об'єктивно співвідносити його кількість і кількість місця для нього. Уже всередині макета починається більш тонка робота (мікротипографіка).

Текст заверстується за допомогою системи сіток, в яких розташовуються текстові блоки і колонки. Залежно від типу сітки і задумки дизайнера текст може заповнювати, наприклад, одну колонку, дві, три або більше. Всередині блоків текст може бути по-різному орієнтований, в залежності від поставленого завдання. Основні типи: *повна виключка (justify), вирівнювання по центру (align center), флагове вирівнювання (align left, align right).*



Рисунок 4.89 – Види виключки

Блок тексту з автоматично виставленої вимкненням за форматом (justify) виглядає «дірявим» і потребує складних налаштуваннях міжлітерних та міжслівних відстаней. Тому набагато частіше в текстах можна зустріти флагову виключку.

4.14.1. «Діряві» і «рідкі» рядки (loose line)

«Рідкі» рядки – явище в текстовому блоці, коли пробілом між словами відстані в якихось місцях стають значно більше інших, утворюючи «дірки» в тексті. «Коридори» або «струмки» – явище, коли пробілом між словами пробіли збігаються в сусідніх рядках по вертикалі або діагоналі.

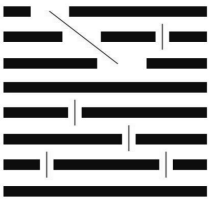


Рисунок 4.90 – «Рідкий» текст: «Дірки» та «струмки»

«Дірки» і «струмки» в тексті найчастіше зустрічаються при повній виключці (вимкненні за форматом). Такий блок найскладніше редагувати.

4.14.2. Рвані краї текстового блоку

Рваний край – занадто великі відстані між словами і знаками в тексті. Часте властивість текстового блоку з флагової вимкненням, так як тільки один край блоку зафіксовано, рівність другого краю залежить від того, скільки знаків поміститься в рядок. Не обов'язково рівняти весь блок як при повній виключці, але занадто велика різниця довжин рядків створює дисбаланс, зайву контрформу і виглядає неакуратно.

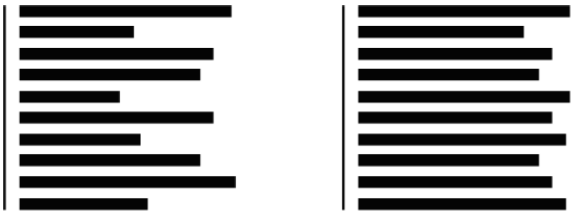


Рисунок 4.91 – Рваний та припустимий краї текстового блоку

Контрольні питання

1. Дайте визначення основних термінів: кегль, трекінг, кернінг, гарнітура, виключка.
2. Що є кернінговою парою?
3. Що є рідкими рядками: «Коридори», «струмки», «дірки»
4. Визначення шрифта, гарнітури.
5. Контрастність, щільність, нахил шрифта, кегль, інтерлін’яж, міжлітерний, міжрядковий пробіл.
6. Перелічіть основні елементи літерної графеми.
7. Поняття контрасту. Види контрастів.
8. Контраст за кольором (колірних зіставлень).
9. Контраст світлого і темного. Контраст холодного і теплого. Контраст додаткових кольорів.
10. Адитивне та субтрактивне поєднання.

Теми для самостійного опрацювання

1. Розпізнавання кольору людиною. Фактори сприйняття кольору.
2. Симультанний контраст. Контраст по насиченості. Контраст по площі колірних плям.
3. Класифікація кольорів. Фізика кольору. Закон І. Ньютона.
4. Колірна система Гете. Колірна система Освальда. Колірна система Манселла. Колірна система Роже де Піль. Колірна система Шеврeля. Колірна модель Ф. О. Рунге. Колірна модель І. Іттена.
5. Поняття колориту. Типи колориту. Види колориту.
6. Монохромний колорит. Двоколірний поліхромний колорит. Насичений або яскравий колорит.
7. Розбілений колорит. Зачерненний колорит. Ламаний колорит. Класичний колорит.
8. Основні принципи візуальної виразності шрифта. Розбірливість шрифта.

9. Класифікація друкарських шрифтів. Цифрові шрифти групи TreTipe та Post Script
10. Вбудовані шрифти, можливості графічних редакторів по модифікації стандартних шрифтів.
11. Основні принципи поєднання шрифта і зображення.

Тестові завдання

1. Трекінг – це

a) рівномірний зміна відстаней між символами у всьому текстовому блоці. Тобто зміна його щільності.

b) коректування відстані між окремою парою знаків.

c) пари знаків, відстань між якими доводиться іноді регулювати вручну через особливості їхнього накреслення.

d) вірний варіант відсутній

2. Кернінг – це

a) рівномірний зміна відстаней між символами у всьому текстовому блоці. Тобто зміна його щільності.

b) коректування відстані між окремою парою знаків.

c) пари знаків, відстань між якими доводиться іноді регулювати вручну через особливості їхнього накреслення.

d) вірний варіант відсутній

3.«Кернінгові пари» – це

a) рівномірний зміна відстаней між символами у всьому текстовому блоці. Тобто зміна його щільності.

b) коректування відстані між окремою парою знаків.

c) пари знаків, відстань між якими доводиться іноді регулювати вручну через особливості їхнього накреслення.

d) вірний варіант відсутній

4. Мистецтво і техніка організації шрифтів це

a) Типографіка
b) Ергономіка
c) Типографія
d) Топографіка

5. Типографіка включає в себе

a) вибір шрифтів, позиціонування букв в спеціально обраному місці, зміна розміру шрифтів та додавання пробілів між рядками і між буквами.

b) позиціонування букв в спеціально обраному місці, зміна розміру шрифтів та додавання пробілів між рядками і між буквами

c) зміна розміру шрифтів та додавання пробілів між рядками і буквами.

d) вірний варіант відсутній

6. За роллю на сторінці шрифти можна поділити на (обрати зайве)

a) Текстові
b) Акцидентні
c) Декоративні
d) Кернінгові

7. До технік стилізації тексту не відносять

a) Зміна розміру та відступів
b) Накреслення літералів
c) Колір та стиль
d) Відстань між буквами та рядками

8. Ахроматичний колір:

a) Чорний
b) Синій
c) Красний
d) Зелений

9. Хроматичний колір:

a) Білий
b) Зелений
c) Сірий
d) Чорний

10. Чи може ахроматичний колір бути теплим або холодним?

a) може
b) іноді
c) не може
d) це питання ще не досліджене

Практичні завдання

Завдання 4.1. Дослідити основні принципи кольорових поєднань за допомогою кольорового кола та скомбінувати кольори для інтерфейсу у кольорову схему: здійснити добір фонового, основного та другорядних кольорів а також акцидентний колір.

Завдання 4.2. Оформити колористичне поєднання у вигляді керівництва по колористичному стилю інтерфейсу;

Завдання 4.3. Дослідити основні принципи шрифтових поєднань та скомбінувати шрифти для інтерфейсу: здійснити добір основного та другорядного, а також акцидентного шрифтів для користувацького інтерфейсу.

Завдання 4.4. Оформити шрифтові поєднання у керівництво з типографського стилю;

Завдання 4.5. Доопрацювати розробленими керівництвами зі стилю дизайн-документ.

5. КЕРУВАННЯ КОНТЕНТОМ СТОРІНОК. ОПТИЧНИЙ БАЛАНС. ТИПОГРАФІЧНА ІЄРАРХІЯ.

План:

5.1. Розмір файлів проекту і дозвіл
5.2. Макет заснований на сітці
  5.2.1. Квадратна модульна сітка
  5.2.2. Колонки
  5.2.3. Вертикальний ритм
  5.2.4. На лінії
  5.2.5. Між лініями
  5.2.6. Без сіток
  5.2.7. Комбінації систем сіток
  5.2.8. Помилки в сітках
  5.2.9. Налагодження та використання системи сіток
  5.2.10. Чуйність і гнучкість.
  .2.11. Вирівнювання і баланс.
  5.2.12. Однаковість.
  5.2.13. Відступи.
5.3. Ієрархія контенту.
5.4. Візуальний баланс.
5.5. Вирівнювання.
5.6. Техніки ієрархізації тексту
  5.6.1. Розмір.
  5.6.2. Колір.
  5.6.3. Стиль.
  5.6.4. Letterspace (Відстань між буквами).
  5.6.5. Leading (Відстань між рядками).
  5.6.6. Space (Відступи)
5.7. Оптичний баланс у типографіці для UI
  5.7.1. Вирівнювання в типографіці
  5.7.2. Нерівномірна відстань
  5.7.3. Типографічна ієрархія
  5.7.4. Розмір шрифту і відстань між рядками.
  5.7.5. Вертикальний ритм.
  5.7.6. Підсилений контраст.
5.8. Негативний вплив перевантаження контентом

5.1. Розмір файлів проекту і дозвіл

Розмір вашого полотна залежить від того над яким проектом ви працюєте. Зазвичай повна ширина веб-сайту становить 1140 пікселів і можливо доречно додати по 150 пікселів по обидва боки, так ви зможете побачити фон навколо сторінки.

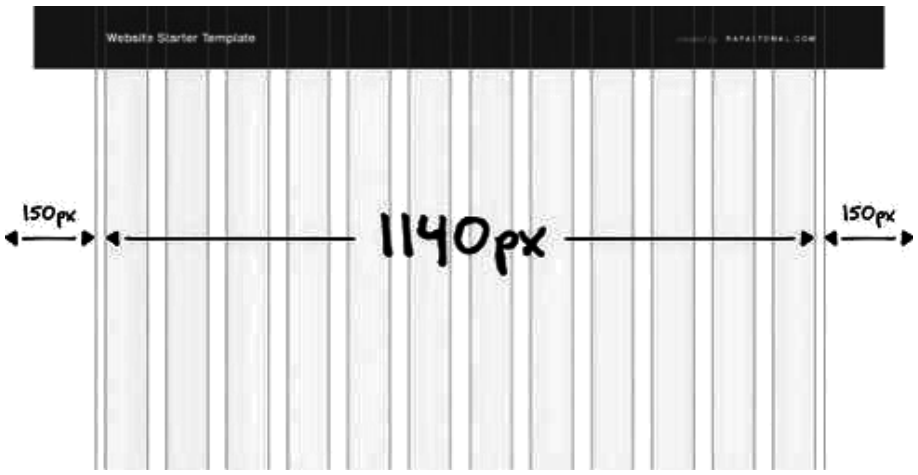


Рисунок 5.1 – Повна ширина сайту для розташування контенту та поля

72 DPI (Dots Per Inch – Точок на дюйм) або PPI (Pixel Per Inch – пікселів на дюйм) це стандартний дозвіл для всієї графіки, яка відображається в інтернеті. До речі, вам не варто турбуватися про розмір зображення під час розробки дизайну для інтернету тому не має значення який DPI / PPI є у вашого зображення. Він завжди виглядає однаково в браузері. DPI / PPI відрізняються один від одного тільки в тому випадку, коли ви створюєте дизайн зображень, які підуть на друк. Вам варто так само запам'ятати, що формати веб зображень, такі як PNG та GIF не мають налаштувань DPI / PPI. Значення, в дизайні для вебу, має тільки розмір в пікселях.

Так чи інакше, все не так просто. Все було б простіше, якби не існувало екранів високої чіткості, такі як, наприклад Retina екрани від Apple. Всі зображення, які були зроблені для дисплеїв ретина, повинні бути експортовані в подвійному розмірі, це дозволить уникнути "замилювання" картинки. Всі ваші зображення повинні мати версію подвійного розміру для екранів Retina.



Рисунок 5.2 – Відмінність retina дисплеїв: 1x size (звичайні дисплеї) і розмір 2x (дисплеї retina).

Воно буде виглядати більш чітким, коли ви будете дивитися на нього в браузері. Тепер вам необхідно запам'ятати, що вам необхідно надати два набору зображень: Розмір: 1x size (звичайні дисплеї) і розмір 2x (дисплеї retina).

Це трохи ускладнює дизайнерську роботу і для того щоб встигати виконувати проекти вам потрібно буде готувати файли дизайну проекту не тільки для звичайних дисплеїв, але і для дисплеїв високої чіткості.

Є, як мінімум два способи як можна створити дизайн і бути впевненим, що графіка може бути експортована для дисплеїв Retina:

*Створіть початковий дизайн в 1x***.** У даному випадку розмір пікселів на полотні буде відповідати фінального розміром в веб-браузері. Ви повинні використовувати тільки векторні фігури і зображення, які легко можуть бути масштабовані без втрати якості до розмірів, які необхідні дисплеям Retina.

Якщо ви працюєте за дисплеєм Retina, вся ваша робота в Photoshop буде розмита. Це може бути досить дратівливо, і ви можливо навіть не зможете так працювати.

Створіть початковий дизайн в 2x. У цьому випадку, розмір пікселів на полотні буде в два рази більше, ніж він повинен бути в браузері. Вам треба пам'ятати, що всі (розмір шрифтів, висота рядків і всі інші розміри) в 2 рази більше в вашому дизайні.

Ви можете використовувати і векторні фігури, і растрові зображення. Ваші растрові зображення не втратять як після зменшення масштабу перед експортом. Якщо ви працюєте за Retina екраном, все в фотошопі буде чітким і різким. Ви так само можете працювати за звичайними екранами, але вам треба буде зменшити розмір дизайну до 50%.

Як ви можете бачити, обидва способи мають і плюси і мінуси. Не існує прекрасного способу. Зазвичай розробляють дизайн при розмірі 2x. Доброю практикою є використання зображень заснованих на векторах і формах, які легко можуть бути перетворювати з використанням CSS і HTML. Це гарантує, що ваш дизайн можна буде легко перетворити в добре оптимізований веб-сайт. Якщо ви повинні використовувати растрові зображення, переконайтеся, що у вас завжди є копія цього зображення з високою роздільною здатністю в папці assets вашого проекту.

5.2. Макет заснований на сітці

Макет, який заснований на сітці, це макет з вертикальними і горизонтальними напрямними, який може допомогти організувати ваш контент і підтримувати однаковість структури дизайну протягом усього проекту. Багато століть сітки використовуються для створення дизайну з оптимальними пропорціями. За допомогою них читати текст стає простіше. Сітка, яку ми використовуємо сьогодні в веб-дизайні, була створена і популяризована дуже талановитими Швейцарськими дизайнерами в 1920-их. З тих пір їх дизайн став міжнародним стандартом того, як потрібно організовувати контент на сторінках. Досить очевидно, що цей дизайн був так само представлений і он-лайн, тому що макет заснований на сітці допомагає вирішувати багато проблем з якими стикаються веб-дизайнери під час своєї роботи. Всі аспекти гарного макета можна втілити системою сіток:

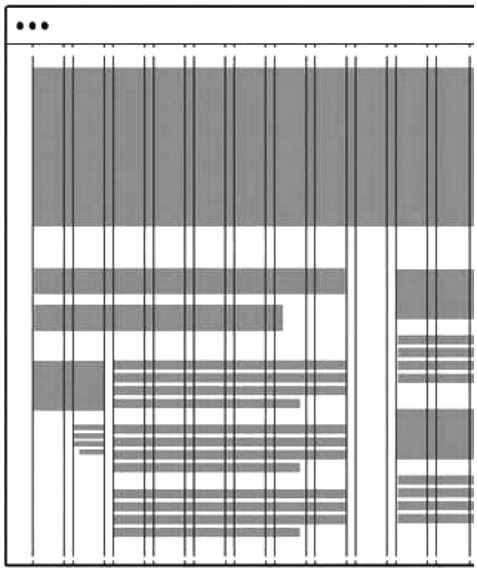


Рисунок 5.3 – Макет заснований на сітці

5.2.1. Квадратна модульна сітка

Тут досить просто скласти і швидко спроектувати рівний дизайн з рівними відступами.

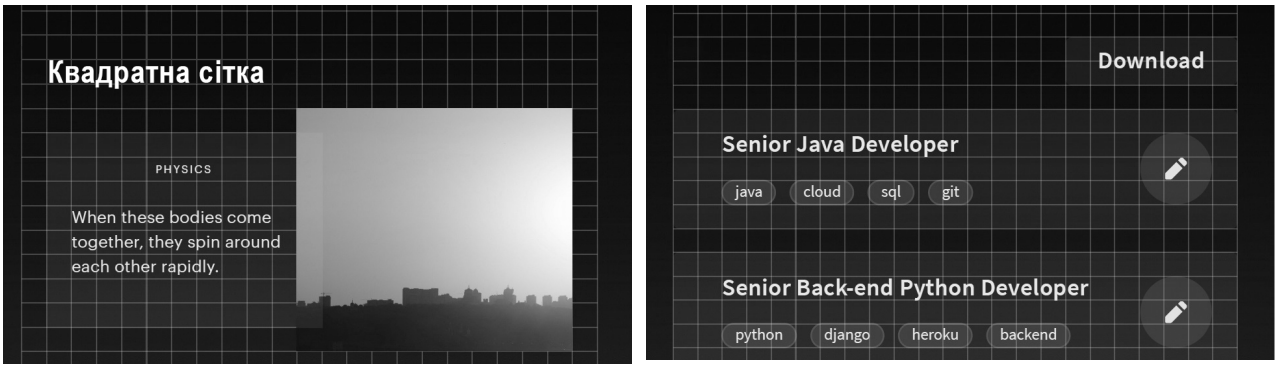


Рисунок 5.4 – Квадратна модульна сітка

Але це не про інтерфейси. Вони зазвичай трохи складніші, але і їх можна збирати по квадратах. Це сильно спрощує завдання.

Але зазвичай інтерфейси не будуються за такими сіток.

5.2.2. Колонки

Зазвичай веб працює саме на цій сітці. Це новий вимір ширини: тепер ви думаєте не про кількість зайнятих пікселів елементом, а про колонки.

Праві блоки займають 8 колонок, а лівий 4. Це означає, що на інших дозволах моніторів вони завжди будуть займати саме таку кількість колонок.

Колонки також можуть підлаштовуватися під конкретний пристрій. Якщо в планшеті не 12 колонок, а, наприклад, 6, тоді все блоки будуть займати саме 6 колонок і розмістяться один під одним.

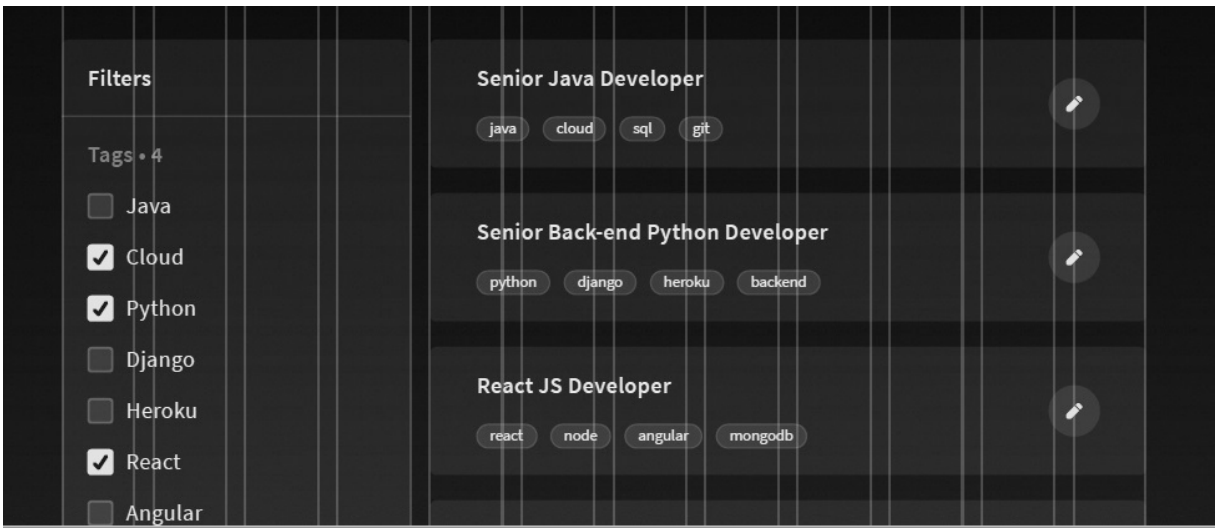


Рисунок 5.5 – Колонки

5.2.3. Вертикальний ритм

Веб і інтерфейси будуються не тільки в ширину. Говорячи про висоту, потрібно пам'ятати про вертикальний ритм. Це як в музиці: якщо ритм ламаний, він може викликати роздратування. В деякі моменти у нас завжди з'являються різні відступи і використання наступних трьох способів допоможе збалансувати ваш дизайн.

5.2.4. На лінії

Це стосується тексту. Наприклад, у вас є великий заголовок, а правіше потрібно розмістити другорядний текст. Дизайнери часто хочуть в такі моменти вирівняти текст по центру контейнера заголовка. Це не правильно.

Люди читають рядками, а не фігурами, які ви собі придумаєте. У таких випадках текст необхідно розміщувати на одній лінії.

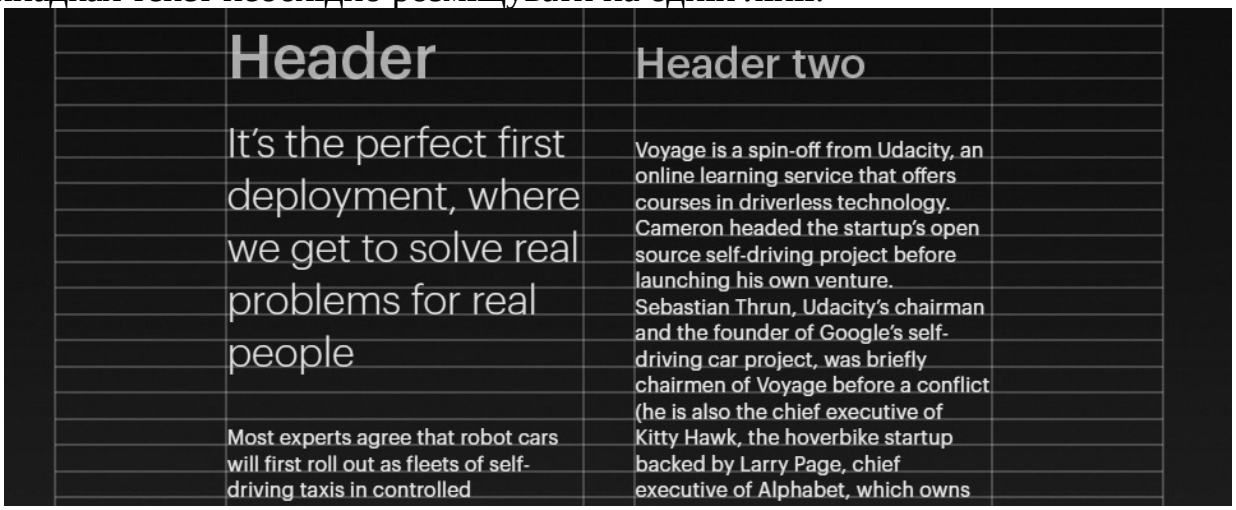


Рисунок 5.6 – На лінії

5.2.5. Між лініями

Текст маємо між лініями сітки. Так ми отримуємо рівні відступи, що спрощує верстку.

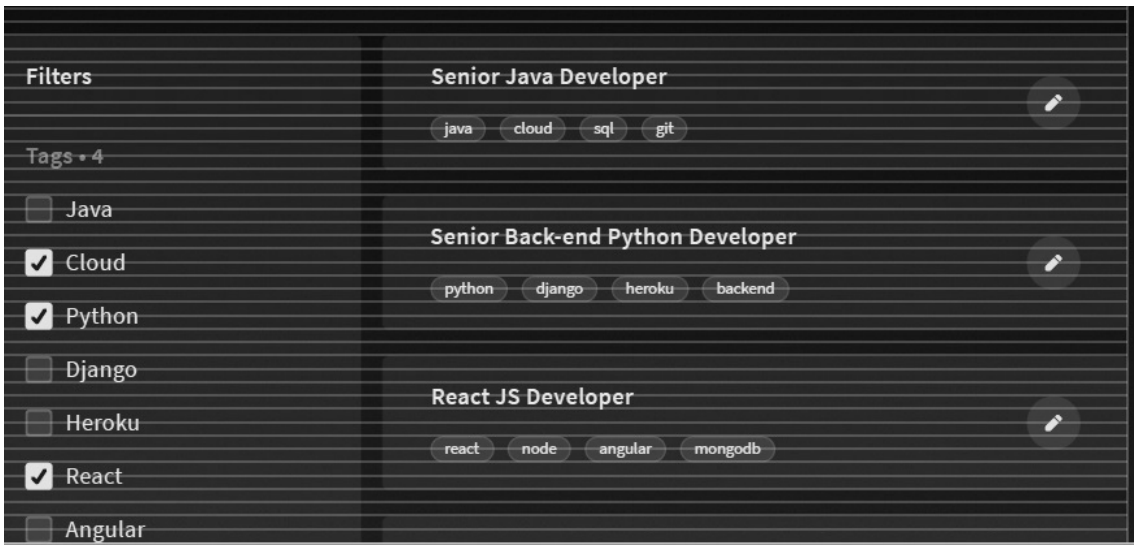


Рисунок 5.7 – Між лініями

5.2.6. Без сіток

Без сіток теж може існувати система, а саме система відступів. Ви маєте в своєму розпорядженні елементи потрібним вам чином і вимірюєте відступи кратними величинами.

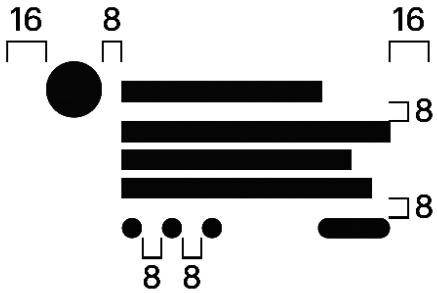


Рисунок 5.8 – Система макету без використання сіток

5.2.7. Комбінації систем сіток

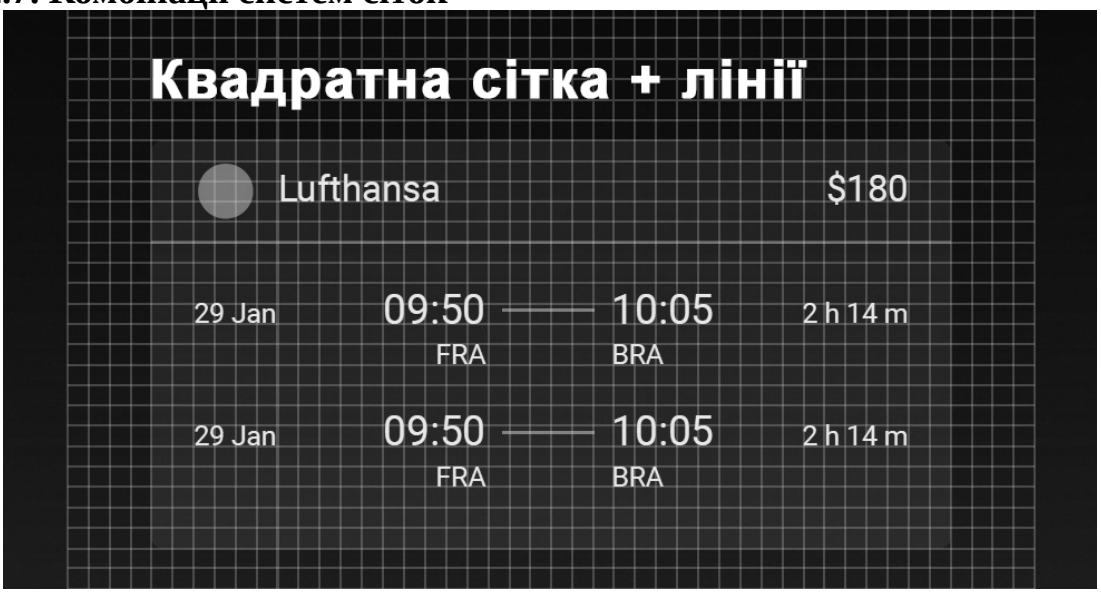


Рисунок 5.9 – Комбінація системи сіток квадратна сітка + лінії

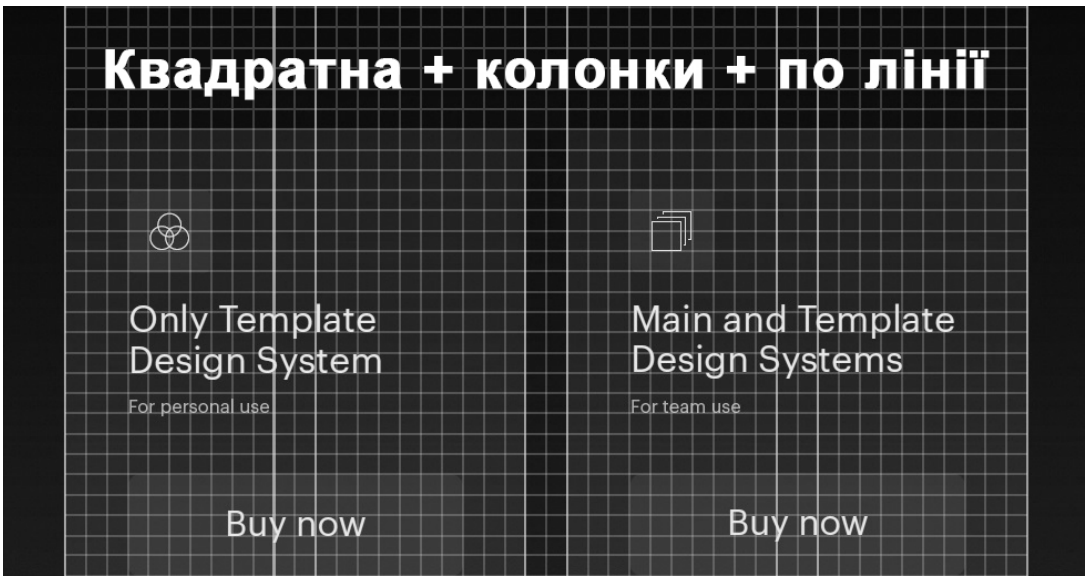


Рисунок 5.10 – Комбінація системи сіток квадратна сітка + колонки+ по лінії



Рисунок 5.11 – Комбінація системи сіток колонки + між лініями

Саме цей спосіб є найбільш практичним, щоб ваші макети потім без проблем верстались. За такою системою виходять рівні відступи, а всі елементи знаходяться в колонках і у верстальників залишається менше питань.

5.2.8. Помилки в сітках

Колонки і хедер. Не варто занадто багато думати про колонках. Хедер може працювати і без колонок.

*Колонки і сайдбар***.** Якщо у вас є елемент, який не повинен розтягуватися, то його краще просто прибити до краю, а все інше вже робити по колонках.

Сітки – це добре. У компанії можна домовитися використовувати конкретну сітку і роботу буде легше дороблятиза іншим учасником команди. Інакше якщо дизайнер захворіє, а сітку він не використав, то залишається тільки здогадуватися, що він мав на увазі під відступом 17. Сітки – це погано, адже вони забирають час для побудови. Краще придумати сітки заздалегідь,

щоб їх можна було використовувати в тому чи іншому випадку. Вони також обмежують творчість і від них пізніше буде складно відмовитися.

5.2.9. Налагодження та використання системи сіток

Сітка, що складається з 12 колонок, є найпопулярнішим макетом в інтернеті. Використовуючи 12 колонок, ви можете просто розділити контент на 4 колонки, три колонки або дві колонки. Макет, що складається з 16 колонок, може дати вам більше можливостей, але він так само ускладнить роботу над вашим дизайном. Я знаходжу макет, що складається з 8-ми колонок дуже простим для роботи, але він не дозволить розбити ваш контент на 3 колонки. 12 – число, яке дає вам дуже багато можливостей в роботі над макетом, і при цьому його не так вже й складно буде підтримувати.

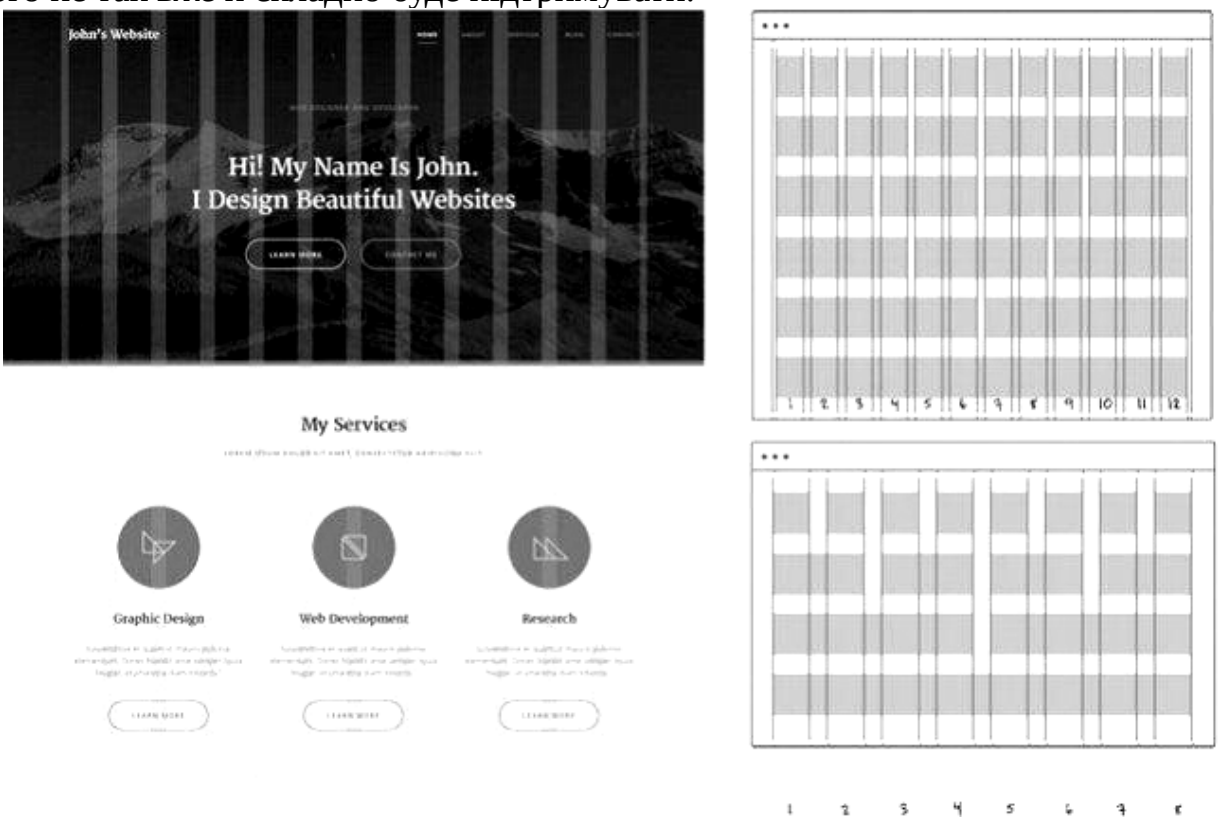


Рисунок 5.12 – Зразок сайту з системою сіток

Всупереч розробки дизайну для друку, у веб-сайтів немає фіксованої висоти, так що ми часто використовуємо тільки колонки, горизонтальні лінії нам практично не потрібні. Звичайно ж, ви можете створити прекрасний статичний дизайн в Photoshop, але веб-розробники не зможуть відтворити його в коді і підтримати гнучкість контенту на сторінці.

Основний принцип полягає в тому, що кожна колонка у вашій сітці повинна мати однакову ширину і у вас повинні бути однакові прогалини між ними. Ви можете скористатися дуже простою математикою, щоб підрахувати необхідну ширину колонок.

Припустимо, ширина сторінки становить 1140 пікселів, і ви хочете використовувати сітку, що складається з 12 колонок, у вас буде 11 прогалин. Для початку вам необхідно визначитися з шириною прогалин, я вибираю 12 пікселів:

11 x 12px = 132px	132 пікселя – це ширина всіх прогалин
1140px – 132px = 1008px	Віднімаємо ширину пробілів з ширини макета
1008px / 12 = 84px	Ділимо отримане значення на кількість колонок

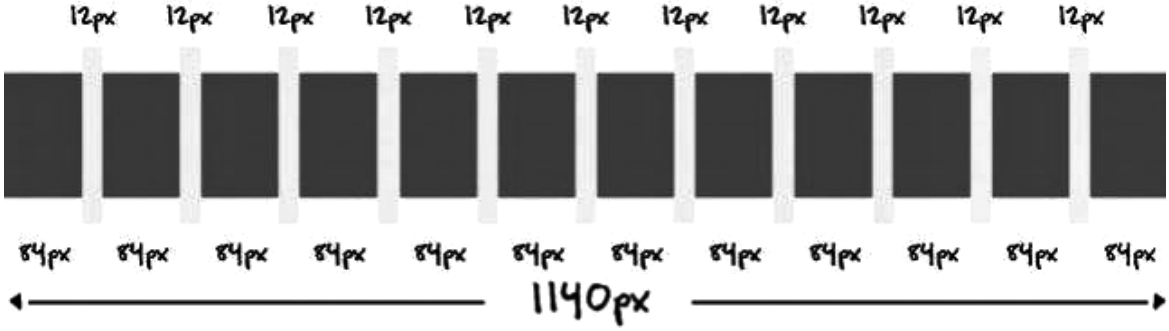


Рисунок 5.13 – Формування колонок

Так у вас вийде макет, в якому у вас є 12 колонок по 84 пікселів і 11 прогалин по 12 пікселів. Вся ширина становить 1140 пікселів. Ви можете використовувати ту ж саму техніку, щоб підрахувати ширину колонок, якщо ви знаєте ширину сторінки і прогалин. Як ви можете помітити, використання сіток може бути дуже корисним в розробці дизайну макета вашого веб-сайту. Це проста і дуже популярна техніка. У вас буде можливість побачити на практиці як створити макет, заснований на решітках в практичній частині даної глави, в якій ми будемо створювати прототипи.

5.2.10. Чуйність і гнучкість.

Через те, що ваш контент розбитий на конкретні колонки, веб-розробникам стає дуже просто передбачити, як вони будуть вести себе при різних розміру екрана. Так само можливо створити "гнучку решітку" в HTML / CSS, яка ґрунтується на відсотках, замість фіксованих значень. Ви можете дуже просто зменшити кількість колонок для маленьких екранів і реорганізувати контент.

5.2.11. Вирівнювання і баланс.

Весь ваш контент вирівнюються уздовж вертикальних колонок протягом всієї сторінки. Це допоможе вам створити чудовий візуальний баланс використовуючи однакові прогалини і горизонтальні контейнери

5.2.12. Однаковість.

Використовуючи однакову систему сіток у всьому вашому веб-сайті, ви можете уявити єдиний однаковий макет і створити шаблон організації вашого контенту. Так ви можете заощадити дуже багато часу використовуючи один і той же макет з направляючими для різних сторінок і це гарантує, що всі елементи матимуть одні і ті ж розміри.

5.2.13. Відступи.

Система, заснована на сітках, дасть вашому контенту можливість "дихати". Це прогалини, які вірно використовуються, а не ті, які були випадково обрані, але це насправді частина вашого візуального шаблону. Прогалини самі по собі створюють візуальну систему, яка представляє собою логічну структуру. Очам дуже просто читати такий сайт.

5.3. Ієрархія контенту.

Ви можете акцентувати увагу на кожному елементі вашого дизайну. Вам потрібна виважена візуальна ієрархія, яка допоможе вам виділити ці важливі області і зробити області, які не дуже важливі менш активними. Вам варто організувати контент вашого веб-сайту та вибудувати його в тому порядку, в якому найзручніше буде працювати людині. Кращий спосіб, це почати з призначення різних рівнів важливості конкретним елементам контенту. Потім, переконайтеся, що ви знайшли найкращі місця на сторінці для цих елементів з високим пріоритетом.



Рисунок 5.14 – Організація контенту та ієрархії

Ви можете управляти ієрархією під час процесу створення макета, позиціонуючи деякі елементи нагорі сторінки або ближче до лівої сторони (якщо ви розробляєте дизайн для мов, в яких люди читають зліва на право), або роблячи ці елементи крупніше, залишаючи великі прогалини між ними (розділяючи їх) або навіть вирівнюючи їх особливим чином. Використання конкретних кольорів і контрастів для розриву шаблону так само допомагає виділити ієрархію і ви можете показати це в вашому макеті, роблячи деякі області більш темними або світлими.

5.4. Візуальний баланс.

Незбалансований дизайн може дуже сильно відволікати від читання. Візуальний баланс – це одна з тих речей, яку ми шукаємо весь час і середньостатистичний користувач може помітити його недосконаліть дуже швидко. Уявіть вертикальні осі в середині вашої сторінки. Тепер спробуйте вирівняти елементи по обидва боки, тепер ми отримали однаковий розподіл візуальної ваги. Ви можете досягти цього двома способами: симетрично і асиметрично. Симетричний баланс означає, що обидві сторони відображаються вздовж вертикальної осі і створюється прекрасне збалансоване зображення:

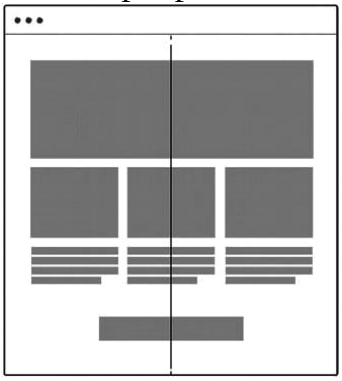


Рисунок 5.15 – Збалансований симетричний баланс

Асиметричний баланс створюється тоді, коли обидві сторони не повністю однакові, але візуальна вага по обидва боки виглядає однаково і створюється візуальний баланс:

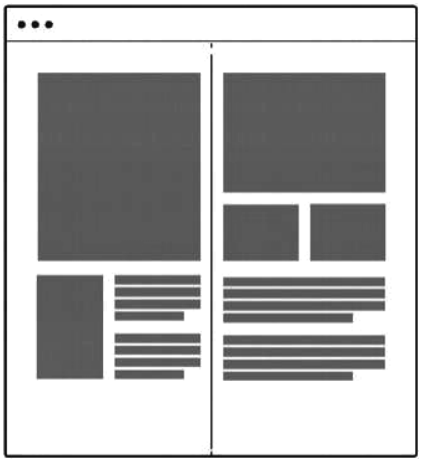


Рисунок 5.16 – Збалансований асиметричний баланс

Коли ви досягаєте добре збалансованого дизайну на сторінці, його стає набагато простіше використовувати за призначенням. Його легко використовувати для залучення уваги відвідувачів в ті моменти, коли вам це необхідно.

5.5. Вирівнювання.

Всі елементи контенту вашого дизайну повинні бути добре вирівняні горизонтально або вертикально.

Ця проста техніка допоможе створити взаємне розташування різних елементів контенту сторінці, вони повинні бути візуально згруповані разом і розташовані в правильному порядку.



Рисунок 5.17 – Вирівнювання елементів по вертикалі відносно напрямних

Веб-сайт є живою формою. Майже кожен веб-сайт в наші дні заснований на системі управління контентом. Це означає, що текст в вашому дизайні редагується людьми, які не є ні дизайнерами, ні веб-розробниками.

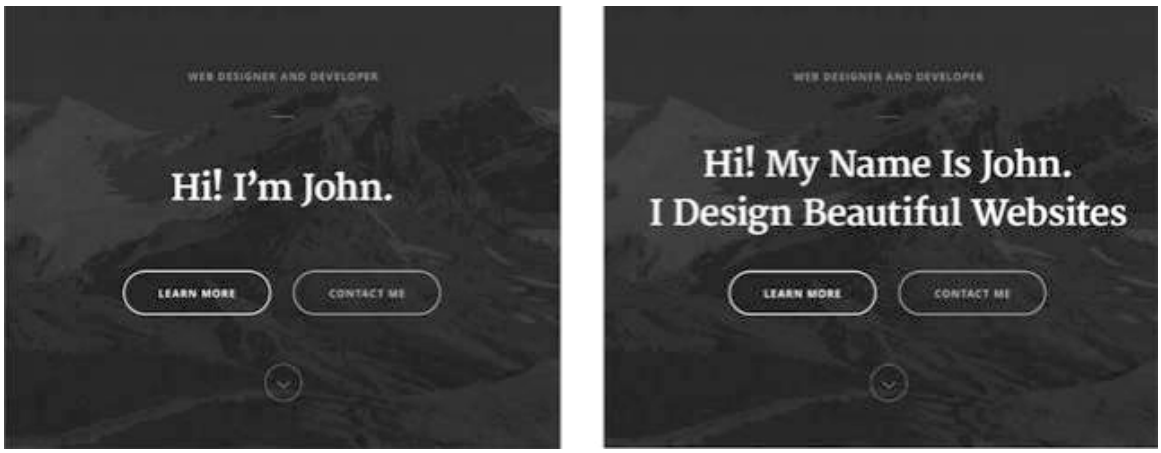


Рисунок 5.18 – Врахування можливості зміни контенту

Ви повинні передбачати як буде вести себе ваш дизайн, якщо хтось змінить контент вашого сайту, що станеться, коли головний заголовок є занадто довгим або якщо запис блозі занадто короткий. Звичайно ж, існує нескінченна кількість можливостей, і ніхто не очікує, що ваш дизайн буде працювати чудово для всіх рішень. Так чи інакше, це те, що ви повинні враховувати і переконайтеся, що ваш дизайн здатний працювати з урахуванням невеликих змін контенту.

5.6. Техніки ієрархізації тексту

Ви можете використовувати різні техніки стилізації типографіки, щоб привернути увагу до тексту, краще організувати масивні блоки контенту, зробити що-небудь більш помітним або просто зробити текст більш естетичним. Ви зазвичай починаєте зі стандартного розміру шрифту, кольору і стилю. Потім ви починаєте працювати над різними стилістичними варіантами. Будьте обережні, не міняйте типографіку занадто сильно. Так ви можете вплинути на читаність або змінити значення оригінального повідомлення.

5.6.1. Розмір.

Зміна розміру – це найпростіший спосіб візуального розділення найважливіших фрагментів тексту. Є просте правило: Чим більше шрифт, тим більше уваги він приверне. Вам потрібно пам'ятати те, що потрібно бути однаковим в тому, який масштаб ви вибираєте (Типографічна ієрархія), так що, наприклад, заголовки пропорційно більше, ніж текст параграфа.



Рисунок 5.19 – Зміна розміру, як спосіб візуального розділення фрагментів тексту

5.6.2. Колір.

Ви можете використовувати дуже контрастні кольори, щоб привернути увагу або використовувати витончений, більш світлий колір, якщо хочете злегка приглушити повідомлення. Використовуйте схожі кольори протягом усього веб-сайту і використовуйте той же самий колір для схожих елементів, щоб створити повторюваний візуальний шаблон.



Рисунок 5.20 – Зміна кольору, як спосіб візуального розділення фрагментів тексту

5.6.3. Стиль.

Ви можете зробити ваш шрифт жирним, якщо ви хочете підкреслити його або використовувати курсив для будь-яких нотаток, цитат або повідомлень. Чи не підкреслюйте текст, якщо він не є посиланням. Тримайте в голові те, що підкреслення в веб-дизайні використовується в основному для посилань і більшість людей вважають, що підкреслений текст є посиланням.



Рисунок 5.21 – Зміна стилю, як спосіб візуального розділення фрагментів тексту

5.6.4. Letterspace (Відстань між буквами).

Змінювати відстань між буквами стоїть при роботі з текстом, написаним великими літерами. Ви можете зробити короткі, великі заголовки, які будуть виглядати дуже цікаво при наявності додаткового простору між буквами.



Рисунок 5.22 – Зміна відстані між буквами, як спосіб візуального розділення фрагментів тексту

5.6.5. Leading (Відстань між рядками).

Відстань між рядками – це відстань між двома базовими лініями тексту. Разом з вірним розміром шрифту може дуже сильно поліпшити зручність читання контенту.



Рисунок 5.23 – Зміна відстані між рядками, як спосіб візуального розділення фрагментів тексту

5.6.6. Space (Відступи)

Відступи можуть використовуватися для кращої візуальної відмінності і організації контенту. Додавання більшого пробілу навколо заголовка або важливого повідомлення може зробити його таким, що набагато більше виділяється.



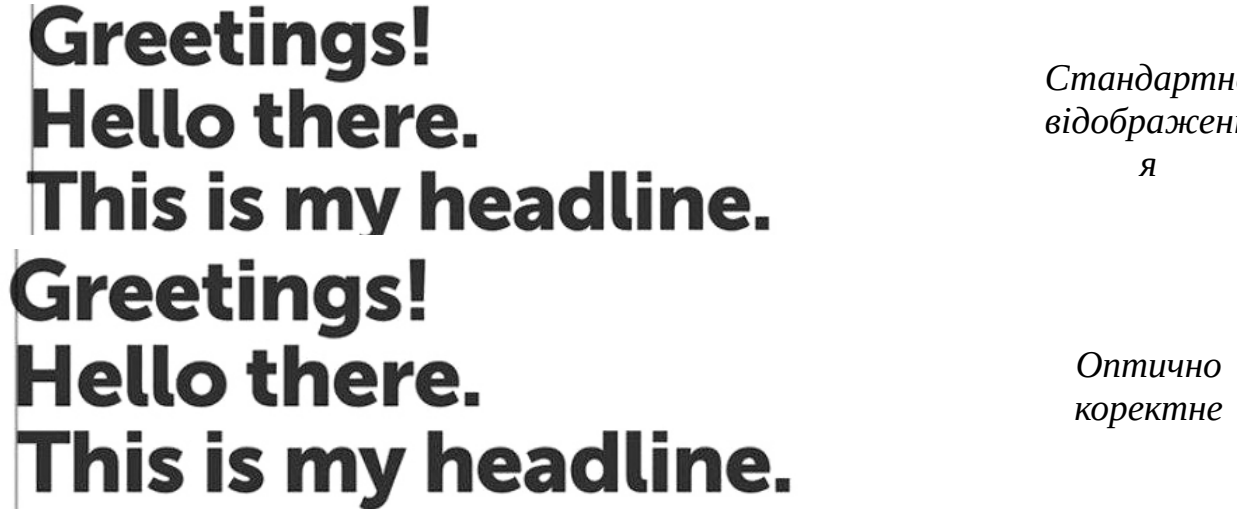
Рисунок 5.24 – Відступи, як спосіб візуального розділення фрагментів тексту

5.7. Оптичний баланс у типографіці для UI

Добре вирівняний та збалансований дизайн виглядає більш організовано та професійно. Проте люди сприймають та обробляють зображення трішки інакшим способом, і те, що може бути ідеально вирівняним та збалансованим для комп'ютера, може здатись не таким же для ваших очей. Ми, як конструктори, повинні враховувати це при розробці користувацьких інтерфейсів. Це може бути однією з причин, чому штучний інтелект не може замінити людину на проектній роботи. Отже, як тільки ми повинні дотримуватись усіх правил проектування, ми також повинні довіряти нашим очам і своїм власним інстинктам, коли мова йде про прийняття рішень візуального дизайну.

5.7.1. Вирівнювання в типографіці

Кернінг та вирівнювання тексту ґрунтуються на показниках, та закріплення його за допомогою CSS потребує великих зусиль або зовнішніх плагінів JS. Не має сталих алгоритмів, що могли б дозволити вирішити всі проблеми. Але є концепція оптимізації дизайну налаштуванням CSS тут и там, щоб це сталося. Одна з найпоширеніших проблем, – це "надмірне" вирівнювання тексту зліва. Це не проблема з невеликим текстом, але це більш очевидно з великими, напівжирними заголовками. Ви можете побачити, що на дисплеї браузера за замовчуванням буква «T» була перенесена з краю на декілька пікселів, щоб візуально виправити її, але відносно інших літер у цьому блоці тексту, вона не мала сенс. Аби це виправити по-перше, вирівнюємо всі літери по лівому краю. Потім зауважте, що «G» не вирівняно з "H" і "T", насправді літера виходить на декілька пікселів з поля. Це все через круглу форму літери, те ж саме з C, O, Q, та S.



Стандартне відображення

Оптично коректне

Рисунок 5.25 – Корекція організації тексту за правилами візуального вирівнювання

Так само необхідно робити і з літералами, що мають гострі краї або лінії, що вказують на ліву сторону, Такі як T, V, W, X, Y та Z. Якщо в однієї літери лінія починається з нижнього регістру, тоді переміщення "T" поза межі краю зовсім зробить текстовий блок оптично краще:



Рисунок 5.26 – Корекція організації тексту за правилами візуального вирівнювання

Ще один приклад, щоб продемонструвати зв'язок між літерами. Для цього перемістимо "W" на кілька пікселів з краю, щоб створити кращий баланс з наступний словом.



Рисунок 5.27 – Корекція організації тексту за правилами візуального вирівнювання

Однак все становиться дещо складнішим для випадку зворотного, коли два слова розташовуються інакше, та слово, що починається з 'W' розташовується у іншому рядку відразу після 'E':



Рисунок 5.28 – Корекція організації тексту за правилами візуального вирівнювання



Рисунок 5.28 – Корекція організації тексту за правилами візуального вирівнювання (продовження)

Тому алгоритм, за яким строки, що починаються з "W", зміщуємо ліворуч ми не можемо застосовувати у будь-якому випадку. Тут немає жодних правил, що можна застосовувати універсально. Все залежить від розміру тексту, стилю шрифту, розміру шрифту, шрифту та використаних букв. Все залежить від співвідношення з іншими літерами. Це той випадок, коли треба довіряти власному зоровому сприйняттю. Вирівнювання до центру:



Стандартне відображення

Оптично коректне

Рисунок 5.29 – Корекція організації тексту за правилами візуального вирівнювання

Друге речення тут закінчується крапкою, що робить цю строку довшою, тому браузер вирівнює текст строго на основі довжини. Проти ваговий баланс не знаходиться саме в середині цього речення. Тому необхідно перемістити верхню лінію на пару пікселів ліворуч, щоб був створений довершений візуальний баланс. Це набуває ще більшої значущості, коли ви додаєте лапки та шрифти за замовчуванням:



Рисунок 5.30 – Корекція організації тексту за правилами візуального вирівнювання

Якщо необхідно вирівняти лапки ліворуч, слід залишити ліва лапку поза межами краю, щоб створити кращий візуальний вигляд:

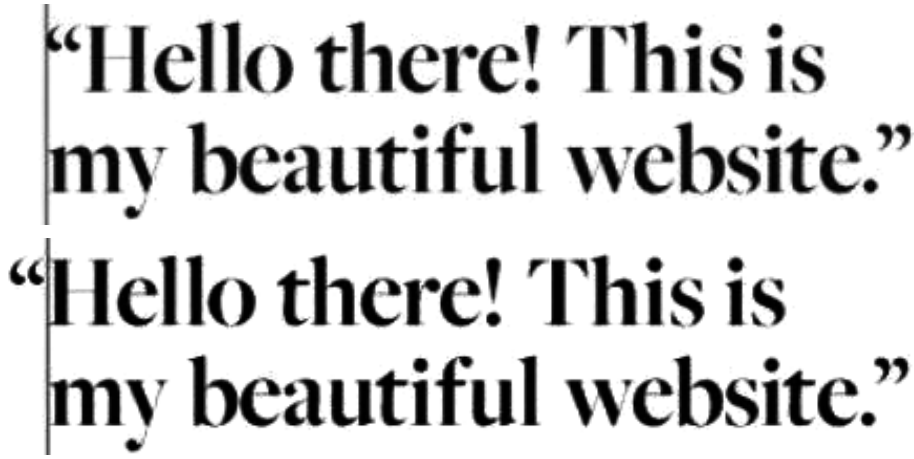


Рисунок 5.31 – Корекція організації тексту за правилами візуального вирівнювання

Висяча пунктуація – це те, що існувало після першої друкованої книги Гутенберга, але потім це забулося, тому що комп'ютери потребували кращого способу відображення тексту на екрані. Це потребує деякої оптимізації як мінімум домашньої сторінки або заголовок цільової сторінки.

5.7.2. Нерівномірна відстань

Говорячи у попередніх лекціях, що поля повинні бути однаковими скрізь, це не стосується текстових блоків. Це проілюстровано на прикладі кнопки:

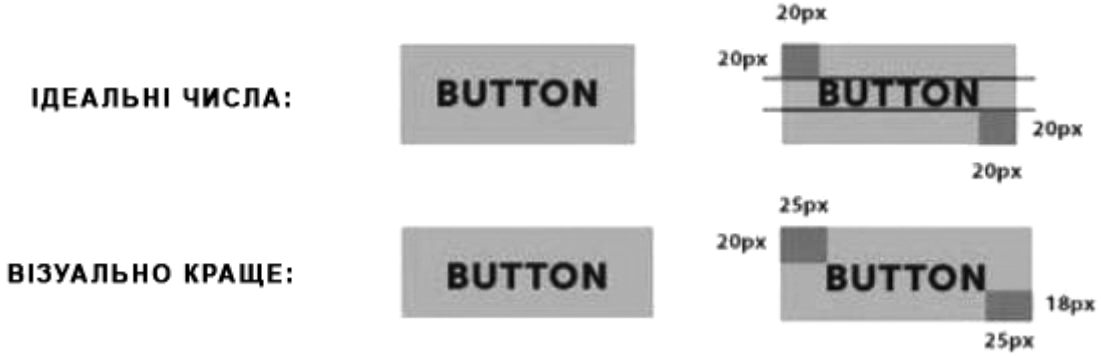


Рисунок 5.32 – Корекція організації тексту на кнопці за правилами візуального вирівнювання

Отже, у нас є дві кнопки з мітками all-caps. Перша з них має рівні відступи від усіх країв навкруги тексту, але насправді це здається візуально гірше. На перший погляд ви можете спостерігати ілюзію, що під текстом більший інтервал, ніж над ним. Щоб виправити це візуально, необхідно зменшити відстань на 2 пікселя від низу та розтягнув горизонтальний інтервал на додаткові 5 пікселів з кожної сторони. Тепер пропорції набагато кращі хоча й недосконалі, дивлячись на цифри. І знову це не є універсальним алгоритмом, тому що все може відрізнятися залежно від вашого шрифту, ваги шрифтів, розміру тощо. Подивіться на одну й ту саму кнопку.

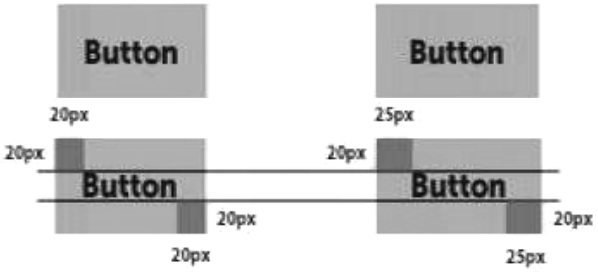


Рисунок 5.33 – Корекція організації тексту на кнопці за правилами візуального вирівнювання

У цьому випадку вертикальний баланс краще, тому що текст у нижньому регістрі більше привертає увагу до базової лінії тексту. Але й тут присутній додатковий ширший інтервал зліва та праворуч, щоб зберегти кращі візуальні пропорції. Ви можете виправити відстань аналогічним чином у вікнах із великим блоком тексту:

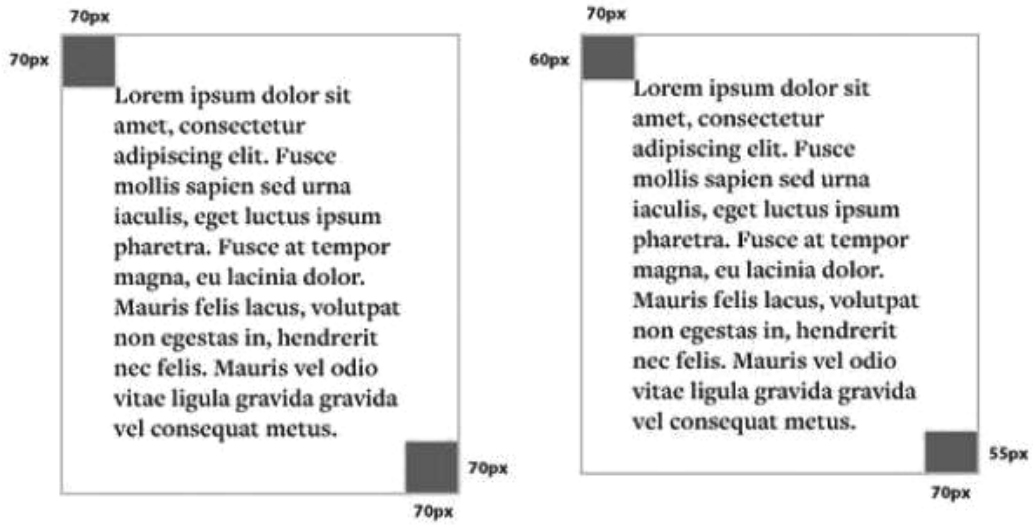


Рисунок 5.34 – Корекція організації тексту у вікні з великим блоком тексту за правилами візуального вирівнювання

Якщо ви вирішите використати рівні всі інтервали, навколо текстового блоку, але це буде здаватися дещо меншим, ніж текстовий блок, де ви зменшите частину вертикального внутрішнього інтервалу. Також пам'ятайте, що нижній інтервал слід виміряти з базової лінії тексту, а не знизу. Ось чому нижнє поле може бути навіть меншим, ніж верхнє. Іншим прикладом може бути розташування піктограм у вікні:

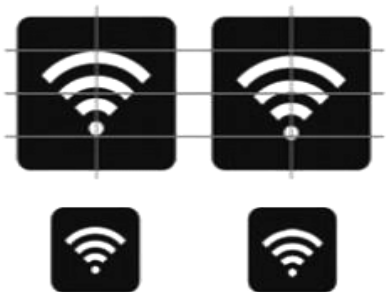


Рисунок 5.35 – Корекція розташування графічного елементу для візуального вирівнювання

Значок ліворуч виглядає дуже добре. Але через збільшену масивну верхню частину піктограми та вузьку нижню створюється візуальний дисбаланс, і здається, ніби піктограма зміщена догори, хоча вона є відцентрованою. Аби позбутись цього ефекту та оптично відцентрувати її, необхідно змістити піктограму нижче центру кнопки. Тому, як ви бачите, для створення візуального балансу необхідно керуватись зоровим сприйняттям ніж програмним центруванням елементів.

5.7.3. Типографічна ієрархія

Типографічна ієрархія – це один з тих елементів, який може допомогти вашому дизайну краще взаємодіяти з користувачами. Ієрархія створює візуальну впорядкованість, так що відвідувачі вашого сайту можуть набагато простіше знайти необхідну їм інформацію в вашому контенті. Хороша ієрархія проведе користувачів через всю сторінку. Ваші користувачі можуть вивчити ваш типографічний шаблон, який допоможе їм швидко пробігтися по контенту і знайти необхідну їм інформацію набагато швидше. Як тільки ви розпланували і застосували ієрархію в вашому проекті, ви повинні слідувати тому ж самому стилю до кінця проекту і навіть додати деякі правила для суміжних проектів. Так ви створите єдиний візуальний бренд. Для пояснення цього концепту продемонструємо пару прикладів того, що з себе представляє дизайн типографіки без застосування ієрархії:



Рисунок 5.36 – Стиль візуальної ієрархії



Рисунок 5.36 – Стиль візуальної ієрархії (продовження)

Дивіться, як сильно візуально відрізняються, один від одного ці приклади через застосування простої типографічної ієрархії. Зауважте, як інформація статті стала набагато більш дрібною і легшою, тепер вона відрізняється від заголовка і параграфа.

Давайте ще раз подивимося на дизайн без ієрархії. Тепер уявіть, що це те місце з якого ви зазвичай починаєте роботу, так що вам потрібно з'ясувати яка інформація повинна виділятися, а яка ні. Після цього є ціла купа невеликих стилістичних рішень, які ви можете використовувати, наприклад розмір шрифту, колір шрифту, інтервали і стилі. Вам завжди варто починати з визначення глобальної типографської ієрархії. У веб-дизайні це буде створення дизайну для всіх основних заголовків від H1 до H6 і для тексту параграфів. Це дасть вам основну ідею того, який розмір у шрифту (простий текст параграфа має найменший розмір, а заголовок H1 має найбільший) і які інтервали ви будете використовувати.

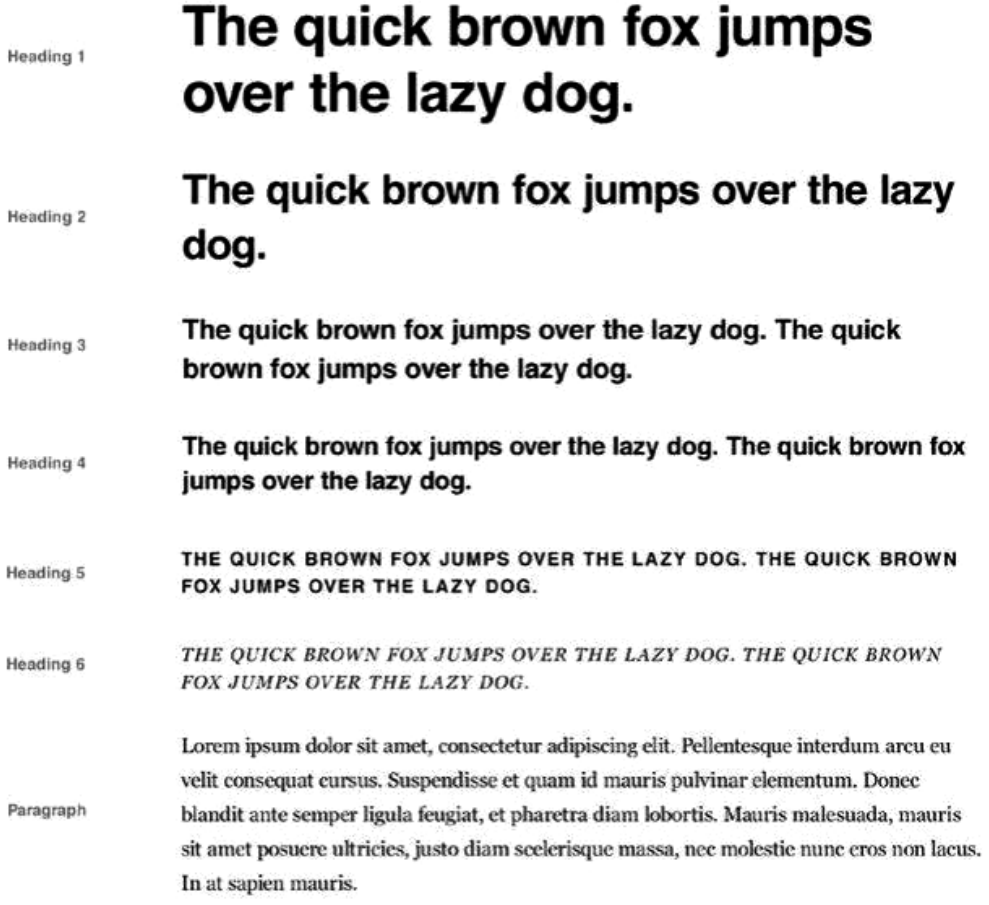


Рисунок 5.37 – Визначення глобальної типографської ієрархії

Вибір вірного масштабу і вертикальних прогалин дуже важливий на цьому кроці. Ваш масштаб повинен пропорційно збільшуватися від тексту параграфа до H1. Ви можете створити свою власну шкалу, яка ґрунтується на традиційній шкалі, яка була популярною в роботі над дизайном протягом багатьох століть:

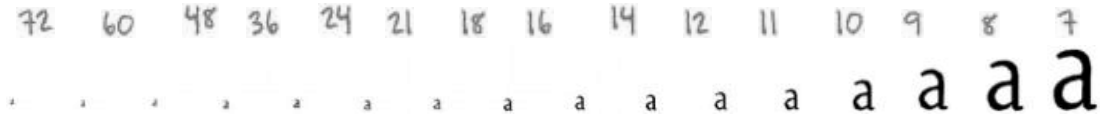


Рисунок 5.38 – Визначення міжлітерних інтервалів

Ваша типографічна шкала може змінюватися в залежності від проекту, ви до сих пір можете базувати її відносно традиційної шкали, наприклад: 16 пікслів для параграфів, 18 пікселів для H4 / H5 / H6, 21 піксель для H3, 24 пікселя для h2 і 48 пікселів для H1

Висота рядків і пробіли перед / після заголовків і параграфів створюють вертикальний ритм вашого контенту. Всі ці числа не повинні бути випадковими, але вони встановлюються в залежності від визначеної шкали. Ви можете використовувати висоту рядків вашого тексту параграфа ґрунтуючись на шкалі і потім використовувати її, коли необхідно застосувати нові прогалини і висоти рядків. Давайте подивимося на приклад:



Рисунок 5.39 – Визначення міжрядкових інтервалів

Найпростіший спосіб визначити висоту рядків тексту це множити базовий розмір шрифту хоча б на 1.5.Наприклад, ваш текст розміром в 18 pt матиме відстань між рядками приблизно 26-28 pt. Це правило може бути

корисно головним чином для тексту тіла сайту, в якому розмір шрифту не дуже великий. У заголовках, де розмір шрифту крупніше 30 pt, ви можливо виберете відстань між рядками в районі 1.25 розміру шрифту. Такий результат буде досить непоганим. Як тільки ви розробите вашу глобальну типографічну ієрархію, ви можете перейти до більш дрібним і більш специфічним частинам вашого дизайну. Це те місце, де ви розробляєте локальну типографічну ієрархію, засновану на глобальній шкалі розмірів шрифту і вертикального розміру. Підтримуючи цей порядок ви зробите так, що ваша типографіка буде виглядати більш організованою і однакової протягом усього проекту.

Покращуємо читаність

Як я згадував раніше на початку глави, коли ви розробляєте вашу типографіку ви можете забути про базовий функціонал, який полягає в читаності.

Звичайно ж, є винятки, коли ви розробляєте твір мистецтва, і ви використовуєте шрифт для розробки абстрактних зображень. Так чи інакше, в будь-якому випадку читаність повинна бути в пріоритеті. Власне кажучи, поліпшити читаність не так складно. Є пара основних причепів, які ви завжди повинні пам'ятати. Якщо ви постійно застосовуєте їх до вашого дизайну, вони швидко перетворяться в звичку, і ви навіть не будете думати про них.

5.7.4. Розмір шрифту і відстань між рядками.

Вибір вірного розміру шрифту і відстані між рядками для вашого тексту сайту залежить від розміру вашої області контенту. Чим ширше ваша область контенту, тим більший шрифт і відстань між рядками ви захочете використовувати.

Оптимальна довжина рядка для простого і комфортного читання в інтернеті становить від 55 до 100 символів. Це досить широкий діапазон і ви можете навіть заходити за межі 100 символів рядку при використанні певних шрифтів і макетів. Так чи інакше, якщо ви хочете, щоб у вас точно не було проблем, то намагайтеся вміститися в ліміт, який складає 100 символів на рядок.

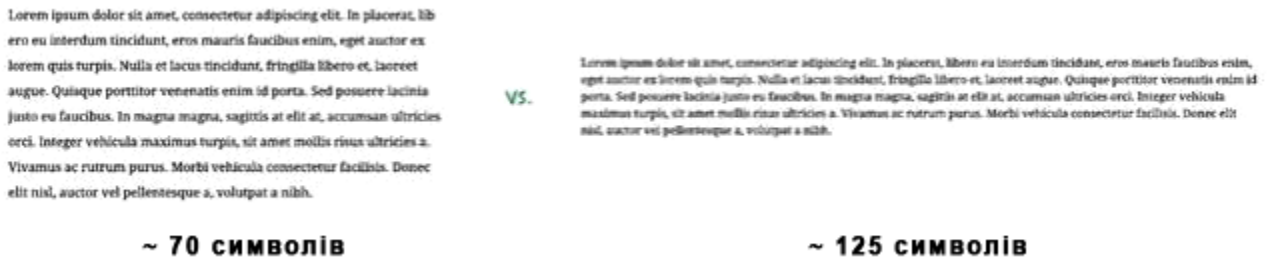


Рисунок 5.40 – Визначення оптимальної довжини рядка

Тримаючи це правило в голові, для контенту шириною в 600 пікселів, вам варто встановити розмір вашого тексту сайту не менше ніж 16 пікселів, так у нас буде приблизно 85 символів на рядку. Як раніше згадувалось, найпростіший спосіб знайти відстань між рядками це помножити розмір шрифту на 1.5. Для шрифту в 16 пікселів цей розмір буде приблизно 24 пікселя, але ви можете зробити це відстань трохи менше або трохи більше в залежності від того

наскільки широка ваша зона контенту. Для контенту розміром в 600 пікселів, ви можете встановити відстань між рядками в 25 або 26 пікселів.

Основним правилом є відсутність невеликих і складно читаються областей тексту в вашому дизайні. Для безпеки ви завжди можете додати трохи більше прогалин, так ваш текст зможе "дихати" і ви так само можете вибрати трохи більший шрифт (16 – 20 пікселів для контенту в тілі сайту). Кріс Персон створив чудовий Golden Ratio Typography Calculator, який може бути дуже корисний, якщо ви хочете швидко підрахувати яку висоту рядків і який розмір шрифту вам варто використовувати для конкретної ширини рядка і CPL (Characters Per Line – Символів на сторінці).

5.7.5. Вертикальний ритм.

згадав вертикальний ритм, коли ми говорили про друкарською ієрархії, але ця тема заслуговує трохи більшої уваги. Чим насправді є вертикальний ритм? Давайте подивимося на цей приклад. У ньому показані дві ситуації. В одній застосовується вірний вертикальний ритм, а в іншій немає:

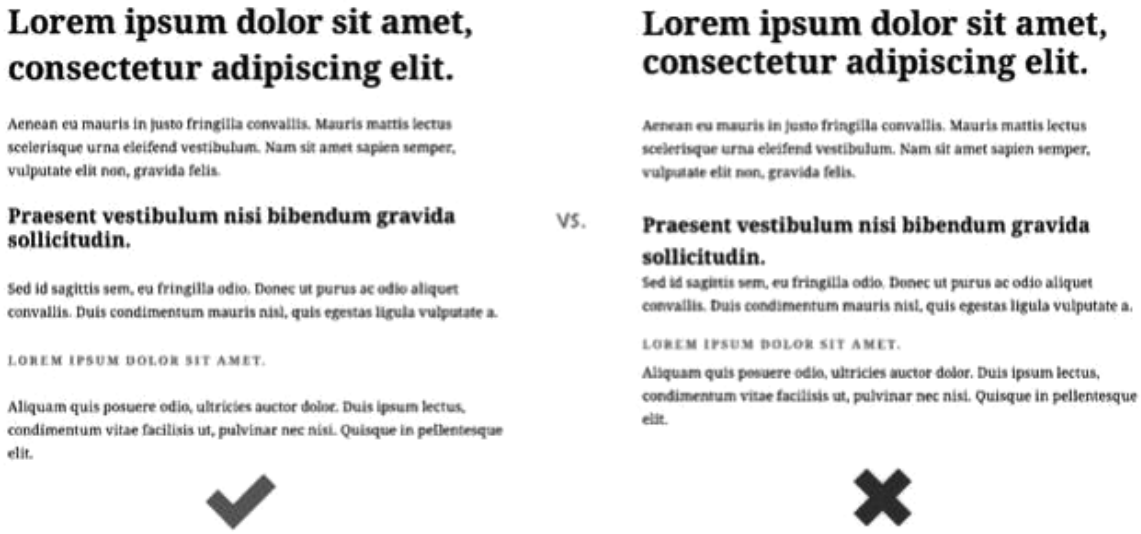


Рисунок 5.41 – Організація вертикального ритму

Вертикальний ритм застосовується тоді, коли ви розробляєте типографіку і інші елементи, ґрунтуючись на базовій лінії сітки, яка розташовується по всій сторінці. Набагато складніше досягти цього в веб-дизайні, тому що тут все дуже рухливого і установка фіксованого вертикального пробілу є неможливим. Не всі елементи можуть бути чудово вирівняні за базовою лінії вашої решітки. Навіть якщо ви зможете один раз домогтися того, щоб вони всі були на одній лінії, вам буде досить важко це підтримати, тому що як ми знаємо, що веб-сайт живе і управляється системою і його відображення повністю залежить від неї. Так чи інакше, ви можете отримати досить хороший вертикальний ритм якщо ви будете діяти простому набору дій:

- Виберіть основну базову лінію, наприклад 24 пікселя.
- Використовуйте розмір вашої базової лінії в якості одиниці вимірювання, що б це не було, відступи або щось ще. Ви можете використовувати пробіл розміром в 48 пікселів під заголовком і використовувати пробіл розміром 24 пікселя між параграфами. Якщо вам потрібен пробіл менше,

ви можете вибрати половину розміру вашої базової лінії, це буде 12 пікселів.

- Використовуйте єдину шкалу розміру шрифтів, яка відповідає вашій базової лінії, наприклад, 16 пікселів при висоті рядка в 24 пікселя і 32 пікселя, при висоті рядка в 48 пікселів (2 x 24 пікселя) і тощо

Використовуйте ті ж самі розміри для схожих елементів протягом усього проекту для підтримки єдиного розміру між рядками. Давайте подивимося на те ж саме зображення з включеною сіткою з базовою лінією в 24 пікселя:



Рисунок 5.42 – Організація вертикального ритму за базовою лінією в 24 пікселя

5.7.6. Підсилений контраст.

Деякий час назад існував досить поганий тренд в Інтернеті. багато дизайнерів використовували сірий шрифт на світло-сірому тлі. Так, це виглядає круто естетично, але роблячи так, ви ігноруєте основну функцію друкарні і погіршує читаність тексту для відвідувачів вашого сайту.



Рисунок 5.43 – Приклад застосування підсиленого контрасту

Слабко контрастний текст дуже важко читається, особливо похилого віку користувачами людьми з поганим зором. Так само люди, які використовують смартфони або люди з дешевими моніторами можуть зазнавати труднощів з читанням контенту веб-сайту. Ви не можете проігнорувати цей сегмент вашої аудиторії. Контент веб-сайту потрібен для того, щоб його читали користувачі і вірите чи ні, люди читають його. Часом деяким особливо цілеспрямованим дизайнерам дуже важко прийняти те, що дизайн повинен страждати на догоду зручності використання. Так чи інакше, це випробування є частиною їх роботи і це одне з найбільших відмінностей, яке відрізняє дизайнерів любителів від професіоналів. Переконайтеся, що у вас завжди використовується висококонтрастна типографіка.

5.8. Негативний вплив перевантаження контентом

Наявність величезної кількості тексту на маленькій ділянці помітно знижує читаність вашого контенту. Відвідувачі вашого сайту будуть постійно відволікатися, і швидко втомляться від такого щільного дизайну. Розробка вертикального ритму може захистити контент від перевантаження, але вертикальних прогалин мало. Ви повинні залишити трохи місця з усіх боків від вашого контенту і зробіть так, щоб існувало візуальне відмінність між різними колонками контенту. Кращий спосіб пояснити те, про що я говорю, це подивитися на приклад:



Рисунок 5.44 – Негативний вплив перевантаження контентом

Контрольні питання

1. Поняття про типографіку.
2. Основні завдання типографіки.
3. Контрасти у типографіці. Гармонія контрасту у типографіці.
4. Поняття тональної колірності.
5. Особливості тональних розрядок.

Теми для самостійного опрацювання

1. Вплив напрямку читання на формотворчі якості шрифту.
2. Принципи геометричного й органічного в типографіці. Оптичні ілюзії при створенні «досконалих» літер та форм.
3. Основні закони типографіки (членування, форми та контрформи, ритму).
4. Функція та форма у типографіці.
5. Колір у типографіці. Історія та сучасні традиції використання кольору.
6. Єдність законів кольору та пропорцій у типографіці.
7. Залежність типографічної форми від мови оригіналу, ширини та густоти набору, міжлітерних пробілів та міжрядь.

Тестові завдання

1. Знайдіть правильне твердження

a) контраст погіршує кольори
b) ваш сайт буде виглядати застарілим з сильним контрастом
c) контраст приглушає кольори
d) контраст не стосується мінімалістичного дизайну

2. Які потрібні умови для виділення тексту

a) яскраві кольори дають можливість виділити певні слова і текст, особливо коли фон тьмяний
b) тьмяні кольори дають можливість виділити певні слова і текст, особливо коли фон тьмяний
c) яскраві кольори дають можливість виділити певні слова і текст, особливо коли фон яскравий
d) тьмяні кольори дають можливість виділити певні слова і текст, особливо коли фон яскравий

3. Тонкі шрифти не

a) виглядають охайно
b) тиснуть на споглядача масивними елементами дизайну
c) роблять сторінку сайту візуально дуже легкою
d) виглядають стильно

4. На одноколірну схему приємно дивитися коли використовується

a) жирний білий шрифт із прозорими кнопками
b) ще декілька яскравих кольорів
c) кнопки іншого кольору
d) тонкий шрифт такого ж кольору

5. Яскраві кольори в UI елементах

a) можуть виділити їх
b) не привертають увагу
c) не спрощують юзабіліті
d) не закликають до дії

6. До технік стилізації тексту не відносять

a) Зміна розміру
b) Накреслення літералів
c) Колір та стиль
d) Відстань між буквами та між рядками

7. Адаптація під Retina використовується для девайсів компанії

a) Microsoft c) Google
b) Apple d) Facebook

8. Не існує наступного виду сітки для макетів:

a) колонки c) квадратна
b) на лінії та між лініями d) фасетна

9. Оптимальна довжина рядка для простого і комфортного читання в інтернеті становить

a) від 155 до 200 символів c) від 55 до 100 символів
b) від 125 символів d) залежить від монітору

10. Різні сітки для організації макетів

a) можна комбінувати у рамках усіх екранів одного інтерфейсу
b) можна комбінувати: окрема сторінка інтерфейсу з окремою сіткою
c) комбінувати заборонено
d) не комбінуються за визначенням

Практичні завдання

Завдання 5.1. Опрацювати дизайн головного екрану згідно до розробеного скетчу або вайрфрейму з використанням модульної сітки. Вибудувати чітку ієрархію у теках проекту та шарах, слідкувати за ім’ям шарів;

Завдання 5.2. Розробити дизайн типових екранів, з опрацьованими модальними вікнами, випадаючим меню, сповіщеннями, реакцією на наведення/натискання мишею тощо. У проекті повинно бути дотримано принципи візуального балансу, побудови типографічної ієрархії та принципи композиційної побудови;

Завдання 5.3. Опрацювати систему відступів. Доопрацювати керівництво зі стилю системою відступів та позначити відступи на макеті головного екрану;

Завдання 5.4. Доопрацювати керівництво зі стилю типовими модальними вікнами, сповіщеннями, кнопками, аватарами тощо.

6. ACCESSEBILITY ДИЗАЙН

План:

6.1. Вступ до доступного дизайну
6.2. Законодавче регулювання
6.3. Стандарти accessibility (консорціум W3C)
6.4. Стандарти та практика веб доступності
6.5. Області порушень в Accessibility дизайні
6.6. Принципи Web Content Accessibility Guidlines (WCAG):
6.7. Рекомендації щодо Accessebility дизайну

6.1. Вступ до доступного дизайну

Приблизно 20% користувачів вебу мають певні ментальні або фізичні порушення. Тобто приблизно кожна п’ята людина відрізняється від того, що заведено називати «нормою». І це безпосередньо впливає на взаємодію з цифровими продуктами й сервісами.

Для початку визначимось із термінами: Доступний, інклюзивний, вільний і універсальний дизайн – в чому різниця?

«Універсальний дизайн» – концепт, придуманий американським архітектором Рональдом Л. Мейсом. Він виділив 7 принципів «універсального дизайну», які описують, як створити хороші умови для людей різного віку і здібностей. Він вірив, що простий доступ до будівель і свобода руху зроблять життя простіше для всіх – не тільки для людей з обмеженими можливостями. С. Голдсміт переформулював цитату Л. Мейса в своїй книзі «Проектування для інвалідів». Він придумав концепт «**вільного дизайну**». Потім ці терміни стали відомі як «інклюзивний» і «доступний дизайн». Так чим вони відрізняються?

«**Доступний дизайн**» – це проста, пряма і швидка взаємодія між користувачем і інтерфейсом, створена для людей з обмеженими можливостями. Такий підхід – частина інклюзивної дизайн-стратегії.

«**Інклюзивний дизайн**» – дизайн, розроблений під потреби людини, інтерфейс, який легко підходить різнорідної групи людей з різними характеристиками і особливостями. У всіх людей відрізняються вік, стать, мову, якою вони говорять, умови середовища, в якій вони живуть. Деякі обмежені в доступі до техніки, в кількості часу і грошей. Інклюзивність робить дизайн доступним і зручним для всіх і кожного.

6.2. Законодавче регулювання

Ось перелік законів, які визначають доступність, описують вимоги й наслідки за їх недотримання.



Британия:

- Disability Discrimination Act 1995
- SENDA (Special Educational Needs and Disability Act) 1995
- Equality Act 2010

Канада:

- Accessible Canada Act (Bill C-81)
- Accessibility for Ontarians with Disabilities Act (AODA)

Аргументи на користь інклюзивного дизайну:

1. Демографічний. Населення планети стрімко старіє, а отже і пенсійний вік, імовірно, збільшуватиметься. Тобто людей, яким потрібен доступний продукт, теж буде більше.
2. Економічний. Люди з інвалідністю на сукупному четвертому місці в рейтингу купівельної спроможності.
3. Законодавчий. Він працює не в усьому світі, але є країни, де виконавці зобов’язані дотримуватись певних норм доступності, інакше вони порушують закон.
4. Юзабіліті для всіх. Якщо робимо дизайн максимально зручним для людей з інвалідністю, то він буде зручним для всіх.
5. Моральний. У людей є моральні зобов’язання, серед яких зменшення кількості перешкод для інших.

Що спільного в цих зображеннях?

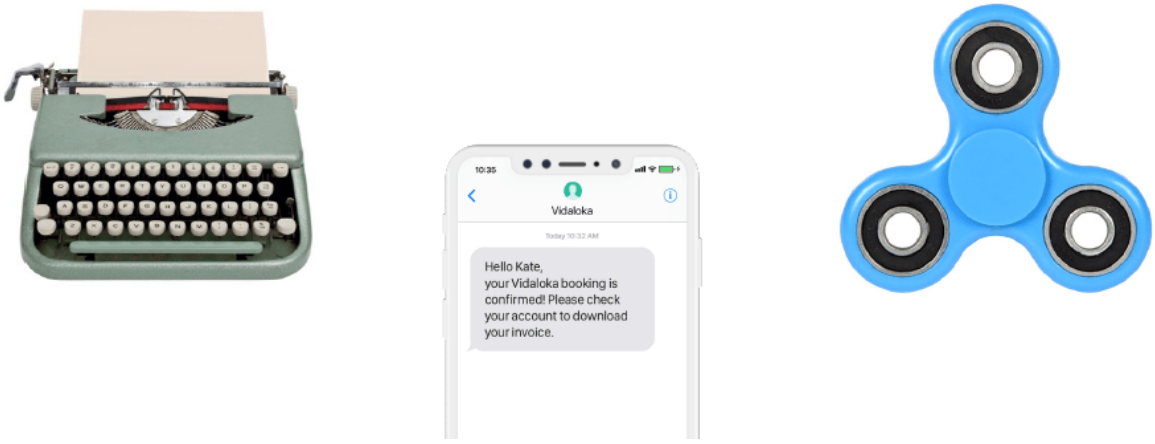


Рисунок 6.1 – Розробки для людей з фізичними порушеннями, які набули широкого розповсюдження

Усі вони були розроблені спочатку для людей з фізичними порушеннями. Імовірно, що друкарську машинку вигадав один італієць для своєї незрячої коханки, аби листуватися з нею без посередників. СМС було запропоновано для людей з порушеннями слуху, а фіджет-спінер розробили для дітей, яким складно концентруватись. Сенс у тому, що речі, створені для конкретної групи людей, зрештою можуть покращувати життя усім.

Медична і соціальна модель інвалідності

Довгий час світ дотримувався медичної моделі. Це той традиційний погляд, знайомий нам з дитинства — мовляв, є фізично обмежені люди, які потребують догляду, лікування і не є дієздатними та повноцінними членами суспільства. При цьому фокус суспільних, державних та соціальних дій — на «виправленні» або компенсуванні обмежень. Приблизно з 1980-х років суспільство поступово переходить на соціальну модель. Вона передбачає інше ставлення до людей, тут фокус переходить на створення прийнятих та комфортних умов. Якщо людина у візку не може піднятись сходами, то проблема не у візку і не в людині, а у відсутності пандуса. Зрозуміло, що через радянське минуле в Україні соціальна модель поки не прижилася. Зараз у нас немає потрібної інфраструктури, тому люди з інвалідністю вимушено ізольовані.

Отже, доступність — це не якісь тенденції, а успішна і дієва бізнес-практика. Окрім того, що є моральні, демографічні й економічні аспекти, в деяких країнах доступність захищається законом.

6.3. Стандарти accessibility (консорціум W3C)

Консорціум Всесвітньої павутини (W3C) займається розробкою міжнародних стандартів, таких як: HTML, CSS, і багато інших. Стандарти консорціуму W3C називаються Рекомендаціями (W3C Recommendations).

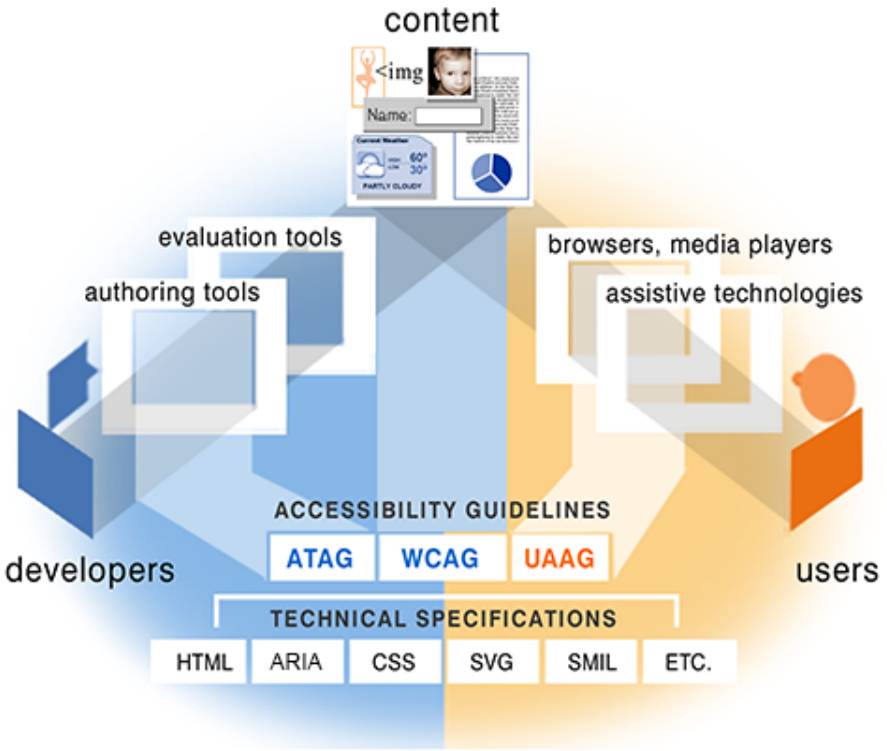


Рисунок 6.2 – Структура стандартів accessibility (консорціум W3C)

6.4. Стандарти та практика веб доступності

Вони є основою веб доступності, роблять веб інтерфейси цілісними, допомагають пріоритезувати певні речі, перевірити відповідність законодавчим нормам тощо.

Є стандарти, присвячені фізичним девайсам і обладнанню, програмному забезпеченню і веб контенту. Ми зосередимось на останньому. Ось три абревіатури, які потрібно запам’ятати:

- Web Accessibility Initiative (WAI) — це організація, що пише та видає гайдлайни про доступність, котрі вже стали стандартами.

- WAI-ARIA — набір стандартів, запущений у 2014 році у відповідь на те, що сайти стали масово використовувати складні інтерактивні елементи, які не є доступними для всіх. Це стосується, наприклад, технології Drag-and-drop або таймерів зворотного відліку.

- WCAG — це основа всієї галузі веб доступності, набір стандартів, які описують і визначають поведінку контенту. Стосується зображень, звуків, мультимедіа, тексту, структури, коду, візуального дизайну.

Перші веб стандарти з’явились ще у 1999 році, тож ця тема не нова і багато аргументів на захист веб доступності вже прозвучали 20 років тому. На сайті Nielsen Norman Group перші статті про це з’явилися теж в 1999, а перше масштабне дослідження та звіт вони зробили в 2001.

6.5. Області порушень в Accessibility дизайні

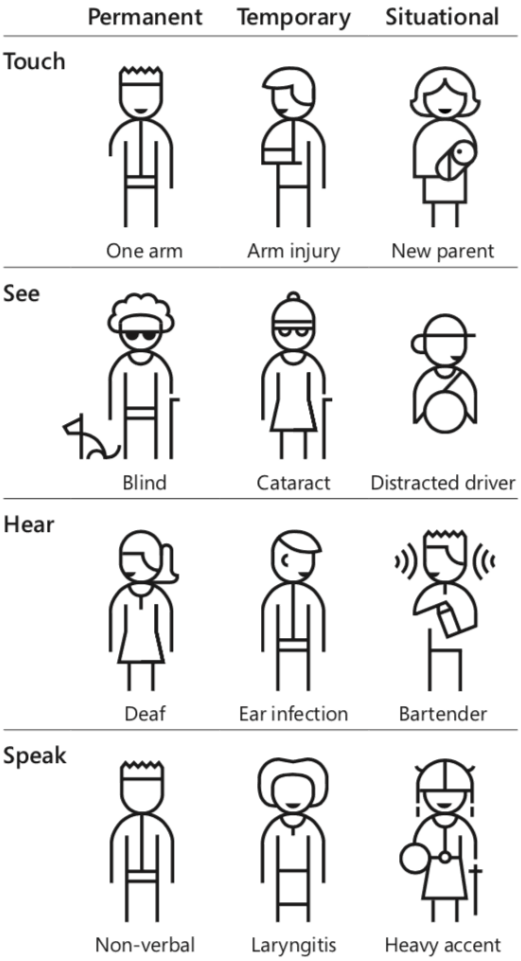
Глобальні розробники наперед думають, як люди з різних категорій будуть сприймати й користуватись запропонованими продуктами або сервісами.

Види порушень:

1. Сенсорні: зір, слух. Серед них — гостра втрата зору (сліпота), порушення зорового поля (глаукома), дальтонізм двох типів. Слух може втратитись чи погіршитись з віком або через травму.
2. Фізичні: моторика. Це може бути повна втрата функціонування (наприклад, відсутність ноги, параліч), втрата рухомості, сили, швидкості, координації м’язів, точності, мобільності.
3. Когнітивні: увага, опрацювання мови. Вони стосуються втрати пам’яті, координації (наприклад, рук і очей), нейрорізноманіття (аутизм, дислексія, диспраксія).
4. Маломобільність: Коли ми говоримо про маломобільність, існує велика кількість можливих ситуацій, в яких практично кожен стає маломобільним. Один з прикладів – замалі кнопки буде важко натискати людям з порушеннями моторики рук, але також і людям, що їдуть в автомобілі, через вібрацію на дорозі.

Для людей з різними порушеннями є спеціальні допоміжні технології. Наприклад, існують спеціальні великі та чутливі клавіатури для людей з порушеннями моторики; є екранні диктори, що озвучують текст, тощо. Існують навіть контролери, котрі керуються за допомогою дихання.

Ще один спосіб класифікації людей з інвалідністю — орієнтуватись на тип порушення: постійне воно, тимчасове чи ситуативне. Ось класифікація Microsoft:



The Persona Spectrum

We use the Persona Spectrum to understand related mismatches and motivations across a spectrum of permanent, temporary, and situational scenarios. It's a quick tool to help foster empathy and to show how a solution scales to a broader audience.

Рисунок 6.3 – Класифікація людей з інвалідністю за Microsoft

Люди, які повністю втратили слух, і ті, в кого це тимчасово, і ті, хто просто стоїть у шумному барі, — всі вони опиняються на якусь мить приблизно в тій самій ситуації. Це, знову ж таки, про те, що доступний дизайн може бути корисним для дуже багатьох різних людей. Якщо це зручно для людей на інвалідних візках і незрячих людей, то це зручно і для велосипедистів, скейтбордистів, людей з дитячими візками, валізами тощо. Те саме і з субтитрами у відео — вони корисні не лише для людей, які не чують, а й, наприклад, для пасажирів на вокзалі чи в аеропорті. Це один із базових концептів універсального дизайну.

6.6. Принципи Web Content Accessibility Guidlines (WCAG):

– **Perceivable** — інтерфейси мають бути розроблені так, щоб люди могли їх сприймати. Вони не можуть бути невидимими, не читатись екранними дикторами, бути винятково звуковими тощо.

– **Operable** — всі взаємодії мають бути доступними. Якщо на сайті певна дія прописана тільки для мишки, то люди, які користуються лише клавіатурою, не зможуть її виконати. Так бути не повинно.

– **Understandable** — інформація й елементи інтерфейсу мають бути зрозумілими та не перевищувати можливостей людини. Це стосується і банального поділу тексту на блоки й заголовки, і абстрактних формулювань, які не всі можуть зрозуміти.

– **Robust** — сайт має підтримувати якомога більше адаптивних технологій (екранних дикторів, різних налаштувань браузера тощо) і при цьому бути зручним (usability).

6.7. Рекомендації щодо Accessebility дизайну

Не використовуйте колір як єдиний спосіб передати відмінність між дизайн-елементами. Використання контрасту, форми, текстури, додаткових лейблів (наприклад, на графіках) значно поліпшить сприйняття інформації людьми з дальтонізм. Trello використовує текстури, щоб допомогти користувачам з дальтонізм розрізняти лейбли. Давайте розглянемо, як це працює для різних типів дальтонізму:

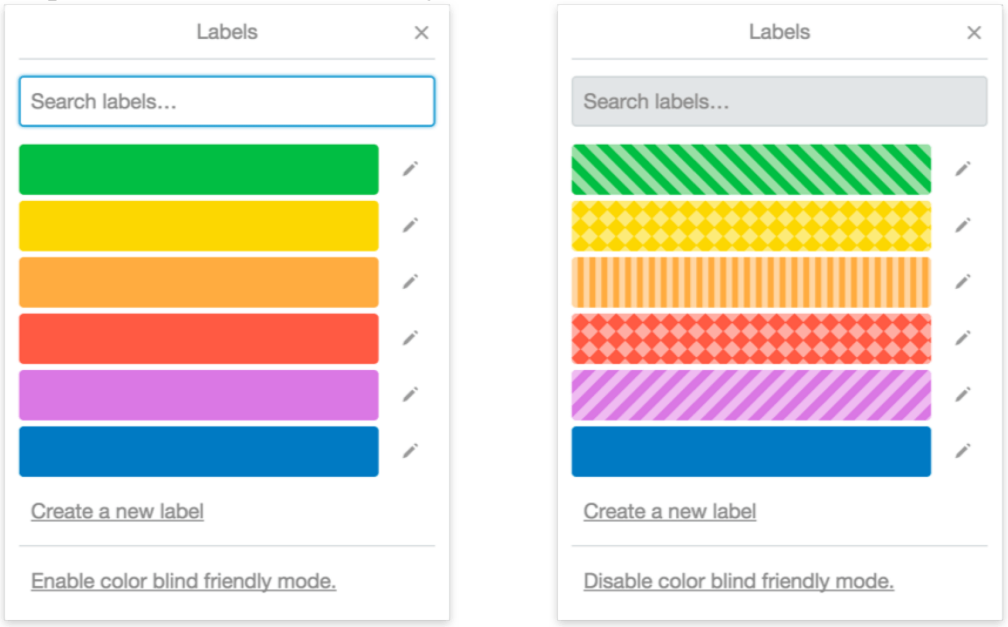


Рисунок 6.4 – Нормальний зір.



Рисунок 6.5 – Дейтеранопія.

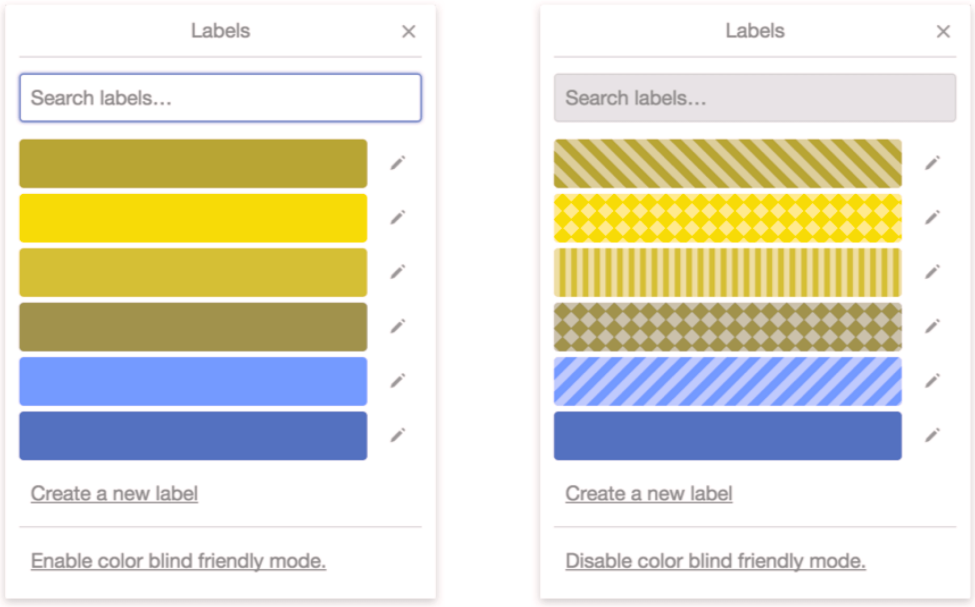


Рисунок 6.6 – Протанопія.

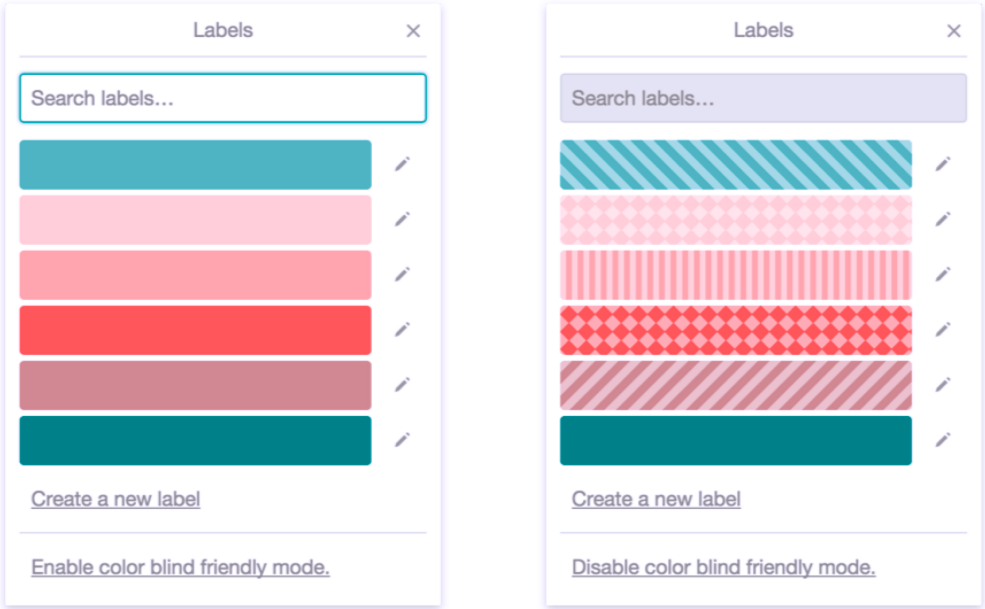


Рисунок 6.7 – Тританопія.

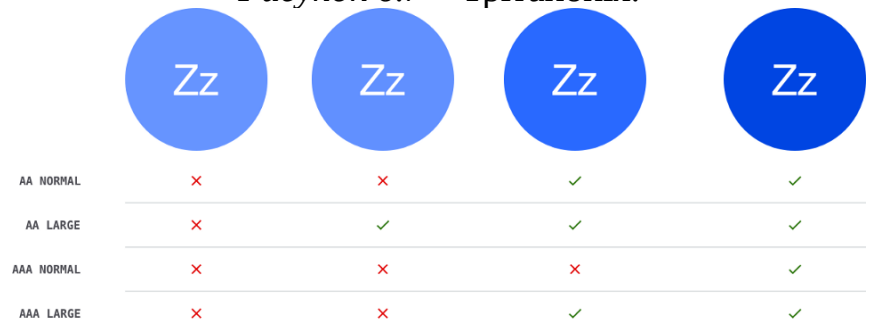


Рисунок 6.8 – Приклади контрастних та неконтрастних відтінків синього

Приділяйте увагу контрасту. Спираючись на гайдлайни WCAG 2.0, можна сказати, що контраст між текстом і фоном повинен складати як мінімум 4.5: 1. Якщо розмір шрифту не менше 24px або 19px bold, мінімальне значення опускається до 3: 1.

Цей принцип заснований на тестуванні серед людей з дальтонізм і іншими проблемами із зором. Слідуючи йому, можна бути впевненим, що текстова інформація буде їм доступна. Є безліч програм і плагінів для перевірки контрастності, наприклад Colour Contrast Analyser. Дотримання таких гайдлайнів вже стало звичайною практикою для світових ЗМІ і сервісів, якими користуються мільйони людей. Це досить легкий спосіб поліпшити продукт, над яким ви працюєте.

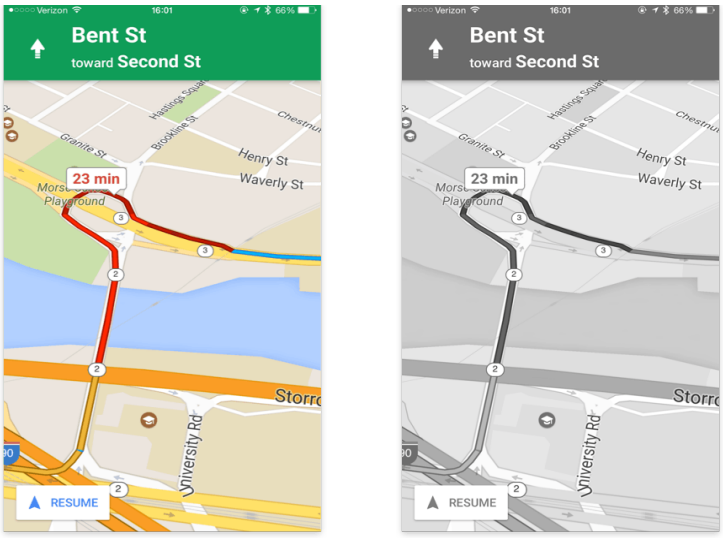


Рисунок 6.9 – Google Maps використовує контраст для індикації щільності трафіку.

Навіть в чорно-білому режимі зони з різною щільністю руху легко розрізняються між собою. Переконайтесь, що всі компоненти достатньо **контрастують** з тлом і самі по собі. Оптимальний мінімальний контраст 4.5:1 для шрифтів в 14px і менше. **Не використовуйте лише колір** для передачі інформації, а додайте іконку.



Рисунок 6.10 – Приклад коригування інтерфейсу за принципами доступного дизайну

Контрольні питання

1. Поясніть, що таке Accessibility. Поясніть відмінність між доступним і інклюзивним дизайном.
2. Якими документами описуються принципи інклюзивного і доступного дизайну?
3. Які особливості розробки Accessibility у дизайні інтерфейсів?
4. Назвіть основні елементи, які повинні підлягати аудиту Accessibility.

Теми для самостійного опрацювання

1. 4 принципи WCAG. Рівні відповідності гайдлайнів.
2. Основні гайдлайни (A/AA) та допоміжні технології.
3. Особливості роботи з кольором у Accessibility дизайні (на прикладі пайчартів та діаграм).
4. Знакова система у Accessibility дизайні.
5. Чекліст WCAG. Google I/O (2019).
6. Accessibility тестування.

Тестові завдання

1. «Доступний дизайн» – це

a) проста, пряма і швидка взаємодія між користувачем і інтерфейсом, створена для людей з обмеженими можливостями. Такий підхід – частина інклюзивної дизайн-стратегії;
b) дизайн, розроблений під потреби людини, інтерфейс, який легко підходить різнорідної групи людей з різними характеристиками і особливостями.
c) багатогранна концепція, яка включає безліч дисциплін: інтерактивний дизайн, інформаційну архітектуру, візуальний дизайн, юзабіліті і взаємодія між людиною і комп'ютером.
d) нічого з вищезазначеного.

2. «Інклюзивний дизайн» – це

a) проста, пряма і швидка взаємодія між користувачем і інтерфейсом, створена для людей з обмеженими можливостями. Такий підхід – частина інклюзивної дизайн-стратегії;
b) дизайн, розроблений під потреби людини, інтерфейс, який легко підходить різнорідної групи людей з різними характеристиками і особливостями.
c) багатогранна концепція, яка включає безліч дисциплін: інтерактивний дизайн, інформаційну архітектуру, візуальний дизайн, юзабіліті і взаємодія між людиною і комп'ютером.
d) нічого з вищезазначеного.

3. «Універсальний дизайн» – концепт, придуманий американським архітектором Рональдом Л. Мейсом пізніше став відомим як

a) «Доступний дизайн»;
b) «Інклюзивний дизайн»;
c) «Вільний дизайн»;
d) «Юзабіліті дизайн».

4. «Вільний дизайну» – концепт С. Голдсміта, який переформулював цитату Л. Мейса в своїй книзі «Проектування для інвалідів»

a) «Доступний дизайн»;
c) «Вільний дизайн»;
b) «Інклюзивний дизайн»;
d) «Юзабіліті дизайн».

5. ***Web Accessibility Initiative (WAI) — це***

a) організація, що пише та видає гайдлайни про доступність, котрі вже стали стандартами.
b) набір стандартів, запущений у 2014 році у відповідь на те, що сайти стали масово використовувати складні інтерактивні елементи, які не є доступними для всіх. Це стосується, наприклад, технології Drag-and-drop або таймерів зворотного відліку.
c) основа всієї галузі веб доступності, набір стандартів, які описують і визначають поведінку контенту. Стосується зображень, звуків, мультимедіа, тексту, структури, коду, візуального дизайну.
d) нічого з вище зазначеного.

6. WAI-ARIA — це

a) організація, що пише та видає гайдлайни про доступність, котрі вже стали стандартами.
b) набір стандартів, запущений у 2014 році у відповідь на те, що сайти стали масово використовувати складні інтерактивні елементи, які не є доступними для всіх. Це стосується, наприклад, технології Drag-and-drop або таймерів зворотного відліку.
c) основа всієї галузі веб доступності, набір стандартів, які описують і визначають поведінку контенту. Стосується зображень, звуків, мультимедіа, тексту, структури, коду, візуального дизайну.
d) Нічого з вище зазначеного

7. WCAG — це

a) організація, що пише та видає гайдлайни про доступність, котрі вже стали стандартами.
b) набір стандартів, запущений у 2014 році у відповідь на те, що сайти стали масово використовувати складні інтерактивні елементи, які не є доступними для всіх. Це стосується, наприклад, технології Drag-and-drop або таймерів зворотного відліку.
c) основа всієї галузі веб доступності, набір стандартів, які описують і визначають поведінку контенту. Стосується зображень, звуків, мультимедіа, тексту, структури, коду, візуального дизайну.
d) нічого з вище зазначеного

8. Доступний дизайн це

a) не нішевий ринок, а невід’ємна складова;
b) необхідність формального дотримання вимог стандартів консорціуму;
c) те ж саме, що й промисловий дизайн;
d) те ж саме, що й юзабіліті дизайн.

9. У концепції універсального дизайну використовують :

a) виключно колір;
b) колір доповнений графічним поясненням або виноскою;
c) колір, доповнений паттерном та графічним поясненням або виноскою;
d) колір заборонено використовувати.

10. Концепція Accessibility дизайну —

a) має відображення у всіх без винятку сферах дизайну;
b) має відображення виключно у напрямах діяльності у сфері графічного дизайну;
c) має своє відображення виключно у дизайні інтерфейсів для застосунків;
d) не має відношення до дизайну.

Практичні завдання

Завдання 6.1. Сформувати та візуалізувати власний Accessibility checklist;

Завдання 6.2. Здійснити Accessebility аудит розробленого дизайну згідно до сформованого чеклисту та за допомогою існуючих інструментів та онлайн ресурсів. Здійснити адаптацію розробленого користувацького інтерфейсу;

Завдання 6.3. Підготувати звіт про зміни у доступності дизайну.

7. ДИЗАЙН-ДОСЛІДЖЕННЯ. UX

План:

7.1. Типи дизайн-досліджень
- 7.1.1. Первинні дослідження
- 7.1.2. Вторинні дослідження
- 7.1.3. Оціночні дослідження
- 7.1.4. Пошукові дослідження
- 7.1.5. Відтворюючі дослідження

7.2. Основні методи дослідження користувацької аудиторії і їх сутність
- 7.2.1. Інтерв'ю
- 7.2.2. Онлайн-опитування
- 7.2.3. Контекстний запит
- 7.2.4. Фокус-групи
- 7.2.5. Створення персон
- 7.2.6. Тестування користувачів
- 7.2.7. Юзабіліті-тестування
- 7.2.8. Віддалене юзабіліті-тестування
- 7.2.9. A / B-тестування (спліт-тестування)

7.3. Більше гіпотез.
- 7.3.1. Мультиваріантне тестування (MVT) і A / B / C /.... тестах.
- 7.3.2. Мікро-конверсії
- 7.3.3. Проведення A / B тестування всього сайту.
- 7.3.4. Збільште тестовий трафік A / B тестування
- 7.3.5. «A / B тест» з використанням тесту в 5 секунд.
- 7.3.6. «B / A тестування»

7.4. Відмінність ринкових досліджень від дизайн-досліджень

7.1. Типи дизайн-досліджень

7.1.1. Первинні дослідження

Первинні дослідження по суті зводяться до нових даних, щоб зрозуміти, для кого ви проектуєте, і що ви плануєте проектувати. Це дозволяє нам перевіряти наші ідеї за допомогою користувачів і розробляти для них більш значущі рішення. Дизайнери зазвичай збирають подібні дані за допомогою інтерв’ю з окремими особами або з невеликими групами, за допомогою опитувань або анкетування.

Важливо зрозуміти, що ви хочете дослідити, перш ніж припинити пошук людей, а також вид або якість даних, які ви хочете зібрати. У статті з Університету Суррея автор звертає увагу на два важливих моменти, які слід враховувати при проведенні первинних досліджень: **обґрунтованість і практичність.**

Обґрунтованість даних відноситься до істини, це те, що вона розповідає про предмет або явище, що вивчається. Можливо, що дані будуть надійними, не будучі при цьому обґрунтованими.

Практичні аспекти дослідження повинні бути ретельно розглянуті при розробці проекту дослідження, наприклад:

- вартість і бюджет;
- час і шкала;
- розмір вибірки.

Браймен в своїй книзі *Методи соціальних досліджень (2001)* визначає чотири типи обґрунтованості, які можуть вплинути на отримані результати:

1. **Обґрунтованість вимірювання або обґрунтованість конструкції:** чи використовує вимірювана міра те, що вона стверджує.

Тобто, чи дійсно статистичні дані про відвідуваність церкви вимірюють силу релігійних переконань?

2. **Внутрішня обґрунтованість:** відноситься до причинності і визначає чи є висновок дослідження або теорії розвиненим істинним відображенням причин.

Тобто, чи дійсно безробіття є причиною злочинності, можливо, є інші пояснення?

3. **Зовнішня обґрунтованість:** розглядає, чи можуть результати конкретного дослідження бути узагальнені для інших груп.

Тобто, якщо в цьому регіоні буде використовуватися один вид підходу до розвитку спільноти, чи буде він мати аналогічний вплив в іншому місці?

4. **Екологічна обґрунтованість:** чи розглядає тезу «… соціальні наукові результати підходять для повсякденного природного середовища людей» (Браймен, 2001)

Тобто, якщо ситуація спостерігається в неправдивій обстановці, як це може вплинути на поведінку людей?

7.1.2. Вторинні дослідження

Вторинні дослідження використовують існуючі дані, такі як Інтернет, книги або статті для підтримки вашого вибору дизайну і контексту. Вторинні дослідження також використовуються як засіб подальшого підтвердження достовірності інформації з первинних досліджень і створення більш міцного випадку для загального дизайну. Як правило, вторинні дослідження вже узагальнили аналітичну картину існуючих досліджень.

Нормально використовувати тільки вторинні дослідження для оцінки дизайну, але, якщо у вас є час, я б точно рекомендував робити первинні дослідження разом з вторинними, щоб дійсно зрозуміти, для кого ви розробляєте і збираєте ідеї, які більш актуальні і привабливі, ніж існуючі дані. Коли ви збираєте дані користувача, специфічні для вашого дизайну, це буде генерувати кращі ідеї і кращий продукт.

7.1.3. Оціночні дослідження

Оціночні дослідження описують конкретну проблему для забезпечення зручності використання і обґрунтування її потребами і бажаннями реальних людей. Одним із способів проведення оціночного дослідження є використання користувачем вашого продукту і надання їм питань або завдань, щоб вони

міркували вголос, коли намагаються їх виконати. Існує два типи оціночних досліджень: *підсумовуюче і формуюче*.

Підсумовуюче оціночне дослідження. Сумарна оцінка спрямована на розуміння результатів або ефектів чого-небудь. Вона більше підкреслює результат, ніж цей процес.

Підсумовуюче дослідження може оцінювати такі речі, як:

- *Фінанси:* вплив з точки зору витрат, заощаджень, прибутку і т. д.
- *Вплив:* широкий ефект, як позитивний, так і негативний, включаючи глибину, поширення і фактор часу.
- *Результати:* чи досягнуто бажані або небажані ефекти.
- *Вторинний аналіз:* аналіз існуючих даних для отримання додаткової інформації.
- *Мета-аналіз:* інтеграція результатів декількох досліджень.

Формуюче оціночне дослідження. Формуюча оцінка використовується, щоб допомогти зміцнити або поліпшити людину, або предмет, який проходить тестування.

Формуюче дослідження може оцінювати такі речі, як:

- *Реалізація:* моніторинг успіху процесу або проекту.
- *Потреби:* погляд на тип і рівень потреби.
- *Потенціал:* здатність використовувати інформацію для формування мети.

7.1.4. Пошукові дослідження

Об’єднання фрагментів даних і їх осмислення є частиною процесу пошукового дослідження.

Пошукові дослідження проводять навколо теми, яка маловідома, або невідома взагалі. Мета пошукового дослідження полягає в тому, щоб отримати глибоке розуміння і знайомство з цією темою, зануривши в неї якомога більше, щоб створити напрямок для потенційного використання цих даних в майбутньому.

З пошуковими дослідженнями у вас є можливість отримати нові ідеї і створити гідні рішення для найбільш значимих проблем.

Пошукові дослідження дозволяють нам підтвердити наші припущення щодо теми, яку часто не беруть до уваги, надаючи можливість генерувати нові ідеї і розвиток для існуючих проблем або можливостей.

Ґрунтуючись на статті з Університету Лінн, пошукові дослідження говорять нам про те, що:

1. Дизайн — зручний спосіб отримання вихідної інформації з певної теми.
2. Пошукові дослідження є гнучкими і можуть вирішувати дослідницькі питання всіх типів (що, чому, як).
3. Пошукові дослідження надають можливість визначати нові терміни і роз’яснювати існуючі концепції.
4. Пошукові дослідження часто використовуються для створення формальних гіпотез і розробки більш точних дослідних проблем.
5. Пошукові дослідження допомагають визначити пріоритети досліджень.

7.1.5. Відтворюючі дослідження

Ці пристрої дозволяють відчути, як бачать, говорять та чують люди з аутизмом.

Відтворюючі дослідження стосуються використання досліджень, які ви провели, і можливості використовувати отриману інформацію для визначення того, яку проблему ви хочете вирішити і створити для неї рішення. Ці рішення, як правило, нові або покращують існуючу проблему.

Оскільки відтворюючі дослідження — це більш-менш етап створення можливостей або рішень, ви повинні заздалегідь розуміти бажання, потреби й цілі своїх користувачів. Відтворюючі дослідження дозволяють нам спостерігати поведінку користувача в природному середовищі, вони дають можливість краще зрозуміти цю поведінку через етнографію, контекстні інтерв’ю, фокус-групи та інтелектуальний аналіз даних.

7.2. Основні методи дослідження користувацької аудиторії і їх сутність

На цьому етапі ваших досліджень ви повинні зосередитися на своїх дослідженнях. Навіщо? Це не робиться спеціально, відповідь криється в нашій основний психології. Існує навіть назва такого явища – ефект Хоторна (він хотів задовольнити інтерв'юера / експериментатора). Користувачі, які оцінюють ваш продукт, швидше за все, дадуть вам відповіді, які ви хочете почути, а не ті, які вам дійсно потрібні.

Вибрати найбільш відповідний метод дослідження для дослідження і використовуйте кілька методів для збору результатів. У нашому керівництві методи класифікуються у відповідності з різними питаннями, на які вони допомагають відповісти, тому давайте заглибимося в цю частину.

В UX дизайні дослідження є фундаментальною частиною рішення відповідних проблем і/або зменшення до «правильних» проблем, з якими стикаються користувачі. Робота дизайнера полягає в тому, щоб їх зрозуміти. Це означає виходити за рамки первісних припущень, щоб поставити себе на місце інших людей, щоб створювати продукти, які відповідають потребам людини.

Хороші дослідження не просто закінчуються хорошими даними, вони закінчуються хорошим дизайном і функціональністю, які користувачі люблять, хочуть і яких вони потребують.

Ці дослідження часто не беруться до уваги, оскільки дизайнери акцентують увагу на візуальній складовій дизайну. Це призводить до поверхневого розуміння людей, для яких він призначений. *Наявність такого мислення суперечить тому, що таке UX. Це орієнтованість на користувача.* UX дизайн зосереджений навколо досліджень, що дозволяють зрозуміти потреби людей і те, яким чином продукти або послуги, які ми створимо, допоможуть їм.

Дослідження користувацької аудиторії – це відправна точка дизайнера в кожному проекті. Воно допомагає нам дізнатися користувачів, мотиви їх поведінки, цілі й потреби. Дослідження також показує, наскільки зручна

навігація в нашій системі, які труднощі виникають у користувачів і, що найважливіше, які відчуття у них з'являються при взаємодії з нашим продуктом.

Дослідження користувацької аудиторії – це основний елемент проектування UX. Незалежно від того, чи працюєте ви у великій корпорації, де спеціальна команда проводить це дослідження для вас, або ви є єдиним юзабіліті-спеціаліст у стартапі, пропустити цю частину процесу проектування ніяк не можна. Будучі UX-дизайнером, ви за визначенням повинні знати все про свій продукт. Однак ваша інтуїція не завжди може безпомилково визначити, що сподобається користувачам, тому проведення дослідження з реальними користувачами так важливо для успішного дизайну.

Нейт Болт (Nate Bolt), засновник лабораторії з дослідження користувацького досвіду Ethnio, ділиться радою: «Не бійтеся застосовувати творчий підхід не тільки до самого процесу проектування, а й до дослідження користувацької аудиторії. Те, як ви проводите інтерв'ю, працюєте з командою і уявляєте отримані результати – все це можливості мислити творчо. Ми звикли вважати проектування частиною творчого процесу, а й дослідження повинно бути не менше творчим».

Ключем тут є емпатія. Ви будете працювати з групами користувачів з різним рівнем підготовки і з різним досвідом за плечима. Ваша робота полягає в тому, щоб зрозуміти, чому вони поступають тим чи іншим чином, і не намагатися змінити їх поведінку або вплинути на нього. Ви повинні зробити так, щоб ваш продукт був корисним для них. Як UX дизайнеру-початківцю, вам дуже важливо проявляти емпатію при кожній можливості.

Дослідження володіє і іншими перевагами. Незаперечність результатів великого дослідження користувацької аудиторії може допомогти отримати підтримку колег або начальства при демонстрації спроектованих рішень.

Далі розглянемо навички, необхідні для проведення дослідження користувацької аудиторії, і обговоримо важливість цього етапу в процесі проектування юзабіліті.

Пітер Мерхольц (Peter Merholz), юзабіліті-спеціаліст і менеджер по продукту компанії PeterMe.com, дав таку пораду, щоб краще розуміти користувачів:

«Не плутайте сам процес з результатом – знати такі елементи проектування UX, як персони (персонажі), «потоки» (flows) і макети UI важливо, але цього не достатньо для формування успішного користувацького досвіду. Важливішим є образ мислення – ви повинні дивитися на світ очима користувачів і робити все можливе, щоб ваша робота мала для них сенс».

«Дослідження не має сенсу, якщо воно не проводиться належним чином. Перехід від вивчення результатів дослідження до проектування – це найбільш важлива частина роботи юзабіліті-спеціалістів», – Гаррі Брігналл (Harry Brignull), фахівець з проектування UX в 90percentofeverything.com.

Коли в своїй роботі ми опираємося тільки на особистий досвід або припущення, ми часто втрачаємо те, яким би міг бути цей досвід для інших людей, зокрема для наших користувачів. Це означає, що ми можемо змарнувати

нагоду поліпшити свою послугу або продукт – зробити їх корисними для користувачів. Крім того, нам може здаватися, що навігація в нашій системі досить проста і зручна, але ж ми працювали в ній довгий час і добре її знаємо.

Простіше кажучи, наші користувачі знають наш продукт не так добре, як його знаємо ми.

Щоб стати юзабіліті спеціалістом, ви повинні завжди дивитися на продукт з позиції користувача; і навчитися цьому можна тільки при роботі з реальними користувачами під час глибинного вивчення аудиторії.

Воно дозволяє нам з'ясувати, що відчувають клієнти при взаємодії з нашим продуктом, і перевірити, чи дійсно наш продукт допомагає їм досягти поставленої мети.

В ході дослідження проектувальник інтерфейсів буде збирати інформацію, використовуючи різні джерела і засоби, для створення позитивного користувацького досвіду. Ми розглянемо деякі з цих методів далі.

Ось, що сказав автор методології Lean UX Джефф Готелф (Jeff Gothelf) про дослідження користувацької аудиторії: *«За останні п'ять років природа програмного забезпечення перейшла в стан постійного і послідовного поліпшення. Завдяки цьому у дизайнерів з'явилася чудова можливість постійно підтримувати зв'язок з аудиторією. Безперервне спілкування – невеликі дослідження, що проводяться часто і регулярно – допомагає чути голос клієнта в процесі прийняття рішень. По суті, воно гарантує, що це дослідження, так само як і розробка, проектування або збір вимог, займе своє місце в процесі ітеративного формування дизайну і розвитку продукту»*.

Ми починаємо процес проектування UX з дослідження користувацької аудиторії, оскільки в іншому випадку наша робота буде ґрунтуватися лише на власному досвіді і припущеннях. А це перешкодить добитися об'єктивності, адже наша думка може відрізнятися від думки клієнтів. Дослідження користувацької аудиторії надає нам дані, необхідні для створення продукту. Без цих даних ми не можемо приступити до роботи.

Починаючи з дослідження, ми економимо собі купу роботи, часу, грошей і ресурсів. В результаті нам залишиться лише внести кілька поправок в наш проект. Але якби ми проводили дослідження вже після того, як закінчили всю роботу, то нам би довелося вносити величезні зміни в інтерфейс, щоб він відповідав запитам опитаних користувачів. Те ж саме стосується і редизайну. У тих, хто працює над редизайном вже існуючого продукту, є перевага – вони бачать, як користувачі сприймають існуючу систему.

На думку Нілу Тернера (Neil Turner), автора блогу UxfortheMasses, хороша основа – ключ до успішного проектування:

«Правильно проведене дослідження користувацької аудиторії – це ключ до створення позитивного користувацького досвіду. Займатися проектуванням, не проводячи дослідження, – це те ж саме, що будувати будинок без міцного фундаменту. Незабаром ваш дизайн почне тріщати по швах і в кінці кінців розвалиться».

Потрібно позначити різницю між тим, щоб слухати користувачів і спостерігати за їх поведінкою. Обидва ці методи дозволяють отримати цінну інформацію під час дослідження. Помилка новачків полягає в тому, що вони занадто концентруються на вислуховуванні користувачів, в той час, як спостереження дозволяє виявити більше інформації за більш короткий термін.

7.2.1. Інтерв'ю

Інтерв'ю з користувачем – це змістовний діалог між інтерв'юером і користувачем з потенційно зацікавленою групи. Його проводять, щоб з'ясувати потреби користувача і його вимоги до продукту. Інтерв'ю може проводитися безпосередньо під час взаємодії клієнта з продуктом; інтерв'юер може ставити питання, щоб дізнатися, про що думає користувач в процесі навігації. Запитайте ваших клієнтів про проблеми, з якими вони стикаються при використанні продукту. Можна навіть запропонувати їм описати, як би виглядав в їхньому уявленні ідеальний продукт. Запитати прямо про цікаві речі. Звичайно, ризик високий (ефект Хоторна нікуди не дінеться), але з ним можна впоратися. Згодом, Ви створите список питань, які допоможуть вам контролювати час, відведений для інтерв'ю, а також його напрямок.

7.2.2. Онлайн-опитування

Онлайн-опитування – це метод збору інформації, що представляє собою набір питань, що розсилаються вашої цільової аудиторії через Інтернет – як правило, у вигляді анкети. Тривалість і формат онлайн-опитування можуть змінюватися в залежності від проекту, але в будь-якому випадку отримані дані заносяться в базу і згодом вивчаються юзабіліті спеціалістом або командою юзабіліті спеціалістів. Перш ніж приступити до складання своєї анкети, проведіть кілька інтерв'ю, щоб повністю зрозуміти предметну область. Це допоможе вам поставити більш точні питання.

7.2.3. Контекстний запит

Цей метод передбачає спостереження за людьми в природних умовах. Йтиметься про повсякденні справи і про те як зробити їх виконання більш комфортним.

В ході такого роду досліджень найкраще скорочувати взаємодія даних. Ви повинні чітко вміти розуміти, що вам потрібно.

7.2.4. Фокус-групи

Процес дослідження фокус-груп передбачає одночасне співбесіду з 3-6 людьми. В такому методі досліджень є багато підводних каменів. Наприклад, колективне мислення може перешкодити і більш сильні особистості можуть пригнічувати голосу слабких. Серед інших питань є також нездатність людей знати наперед, що вони хочуть. І не забувайте про ефект Хоторна.

Якщо ви хочете отримати результати за допомогою фокус-груп, уникайте питань, віддалених від Вашої теми, просто вкажіть кілька провідних тем, які не концентруючи увагу учасників на конкретному елементі.

7.2.5. Створення персон

Персони – це не ваші бажані клієнти. Це ті клієнти, які у вас є або можуть бути. В рамках веб-дизайну персона визначається як опис вигаданого людини в

рамках певної призначеної для користувача аудиторії. Ми розробляємо персони користувачів на основі якісних і кількісних даних дослідження користувацької аудиторії, а також на основі отриманої веб-аналітики.

Використання персон ефективно, якщо вони:

- По-справжньому відображають мотиви, цілі й потреби реальних людей
- Дають ясну картину про очікування користувачів
- Показують, як користувачі взаємодіють з сайтом
- Представляють переважну більшість користувачів сайту

Пітер Морвілло (Peter Morville), відомий як основоположник Інформаційної Архітектури, дав пораду щодо створення персон:

«Портрети і профілі типів користувачів (а також їхні цілі і стереотипи поведінки) нагадують всім нам про те, що ми не є користувачами, і служать компасом при здійсненні проектування та розробки».

7.2.6. Тестування користувачів

Детальніше ми розглянемо цю тему пізніше, але майте на увазі, що при редизайні вже існуючого продукту (на відміну від дизайну нового продукту) таке тестування може послужити цінним джерелом інформації, що дозволяє виявити недоліки продукту з точки зору користувачів.

У своєму висловлюванні юзабілістов і фахівець з питань стратегії з Берліна Хані Ріцька (HanyRizk) підкреслює важливість дослідження користувацької аудиторії в процесі проектування UX:

«Проектування без проведення дослідження користувацької аудиторії повністю суперечить поняттю проектування UX. Дане дослідження допомагає UX-дизайнерам та іншим зацікавленим особам зрозуміти користувачів і їх потреби, а також визначити їх вимоги до продукту. Простіше кажучи, воно замінює припущення в процесі проектування на конкретні дані».

Як і дослідження користувацької аудиторії, тестування є важливою частиною роботи юзабілістов, а також всього процесу проектування юзабіліті. Проектувальники юзабіліті проводять тестування, оскільки воно дозволяє їм удосконалити первісний дизайн продукту або сайту і побачити, як зміни, внесені ними на етапі розробки, витримують критику. Це відмінний спосіб усунути ті недоліки і проблеми, які були виявлені користувачами до початку процесу розробки продукту.

Переконайтеся, що участь в тестуванні беруть реальні користувачі не з кола ваших друзів і родичів!

Тестування – це недооцінене мистецтво. Дуже часто стартапи та підприємці відкладають цей етап, не бажаючи витрачати час і кошти. Деякі з них просто бояться говорити з реальними користувачами. Однак тестування – це не те, що ви можете дозволити собі пропустити, адже навіть саме просто тестування може або затвердити ідею вашого продукту, або зарубати її на корені. Гроші і час, витрачені компанією на тестування на даному етапі, в результаті заощадять набагато більше. Всупереч розхожій думці, тестування зовсім не обов'язково повинно займати багато часу і бути дорогим. Більш того,

дослідження показало, що тестування, в якому беруть участь лише 5 користувачів, допомагає виявити до 85% проблем з юзабіліті.

За допомогою тестування ви можете досягти чудових результатів, докладаючи мінімум зусиль.

Почніть з простого. Тестування вашого продукту не повинно бути складним процесом. Тестування за участю користувачів може бути таким же простим, як створення паперових прототипів або «каркасів» для демонстрації вашого продукту потенційним клієнтам. Ви можете повторно проводити тестування за допомогою цих простих методів до тих пір, поки не знайдете прийнятне рішення тієї чи іншої проблеми. Крім того, ви можете використовувати прототипи, щоб протестувати більше інтерактивних елементів на користувачах.

Пам'ятайте, що чим раніше ви почнете цей процес, тим простіше буде вносити зміни, і, отже, тим більший вплив може надати тестування на якість кінцевого продукту. Тести з залученням користувачів покликані допомогти вам виявити проблеми і знайти їх оптимальні рішення. Але існують також і інші види тестування, як, наприклад, віддалене юзабіліті-тестування і A / B-тестування, докладніше про які ми поговоримо пізніше.

7.2.7. Юзабіліті-тестування

Тести юзабіліті вимагають від користувачів виконання певного набору завдань, підготовлених дослідницькою групою. Це один з найпотужніших інструментів UX. Якщо ви плануєте використовувати тільки один метод для тестування свого продукту, Nielson Norman Group радить вам вибрати тестування юзабіліті.

Юзабіліті-тестування зазвичай здійснюється в «лабораторних» умовах в індивідуальному порядку. Ідея полягає в тому, що учасників (бажано представників вашої цільової аудиторії або персон) просять виконати завдання за допомогою вашого продукту, сайту, додатки, в той час, як проектувальник інтерфейсів або команда проектувальників спостерігає за ними з боку. Мета індивідуального тестування юзабіліті – з'ясувати, з якими проблемами в інтерфейсі зіткнувся користувач, і чому вони виникли. Перевага даного типу тестування в порівнянні з віддаленої перевіркою юзабіліті полягає в тому, що воно дозволяє зафіксувати дії, що вживаються користувачами, а не тільки їх думки на рахунок продукту.

Не забувайте про різницю між вислухуванням користувачів і наглядом за їх поведінкою. Ви можете отримати більше цінної інформації за більш короткий проміжок часу, уважно спостерігаючи за своїми користувачами під час тестування, а також фіксуючи їх зауваження.

Існує 3 способи провести дослідження UX такого роду:

Організоване тестування: зазвичай це виконується в спеціально обладнаних місцях, де незалежні експерти демонструють людям продукт і задають наступні питання. Це ефективно, тому що дозволяє збирати максимальний обсяг інформації, як правило, неупередженої.

Неорганізоване тестування: воно проводиться онлайн, використовуючи спеціальне програмне забезпечення, яке записує активність екрану і аудіо. Інструкції відправляються у вигляді відео або аудіо, користувачі заповнюють тести, коли захочуть.

Таємне тестування: це відносно нова форма дослідження юзабіліті, яка проводиться поза будь-якими лабораторій або будинків користувача. Це робиться в громадських місцях (кафе, автобусні зупинки тщо.), Де користувачам пропонується виконати деякі завдання безкоштовно або за невелику плату.

Але якщо цільова аудиторія вашого продукту обмежена, випадково вибрані люди на вулиці не будуть дуже корисні.

Згідно з дослідженнями Якоба Нільсена, Співзасновника Nielsen і Norman Group, можна виявити 85% основних проблем юзабіліті, використовуючи хоча б 5 чоловік в якості учасників експерименту.

Займаючись проектуванням UX, принципово важливо спостерігати за діями користувачів, не втручаючись і не впливаючи на їх рішення. Іноді учасників просять вголос коментувати свої дії під час проведення юзабіліті-тестування, що дозволяє проектувальникам інтерфейсів краще зрозуміти хід думок користувача під час взаємодії з продуктом.

7.2.8. Віддалене юзабіліті-тестування

Віддалене юзабіліті-тестування може виявитися корисним для команди проектувальників інтерфейсів в разі, якщо немає можливості провести тестування в «лабораторних» умовах в режимі реального часу. Наприклад, коли особиста зустріч з представниками цільової аудиторії є дорогою або важко-здійсненним, або, якщо учасникам тестування складно врегулювати свій розклад.

Учасники, фасилітатори і спостерігачі можуть перебувати далеко один від одного, тому об'єднуються онлайн. Хоча такий формат проведення тестування може бути зручний, цінність його результатів менше (в порівнянні з «лабораторним» тестом), оскільки можливість прямої взаємодії з користувачів забезпечує підвищення якості та точності тесту. Крім того, в разі онлайн-тестування складніше аналізувати отримані дані.

Перевага віддаленого тестування полягає в тому, що користувачі взаємодіють з продуктом в уже знайомій їм обстановці (наприклад, у себе вдома або в офісі). Це допомагає уникнути почуття тривожності і невпевненості, яке долає їх в незнайомому місці і може вплинути на результати тестування.

«Ніколи не забувайте відступити на крок назад і перевірити, чи дійсно ваш дизайн працює. Коли ви займаєтеся дослідженням, пам'ятайте, що він повинен складатися з безлічі ітерацій. Проводити один-єдиний раунд юзабіліті-тестування – це завжди дурна ідея», – цитата Гаррі Брігналла з сайту 90PerCentOfEverything.

7.2.9. A / B-тестування (спліт-тестування)

У цьому типі дослідження ви створюєте дві версії того, що ви хочете змінити / створити. Ви уявляєте їх своєї цільової аудиторії і дозволяєте користувачам вибирати те, що їм більше подобається. Наприклад, створіть два

різних шаблону для інформаційного бюлетеня. Потім розділіть передплатників на групу А і групу В. Після цього відправте кожну версію в відповідну групу і проаналізуйте поведінку кліків / користувачів, щоб вибрати кращий варіант.

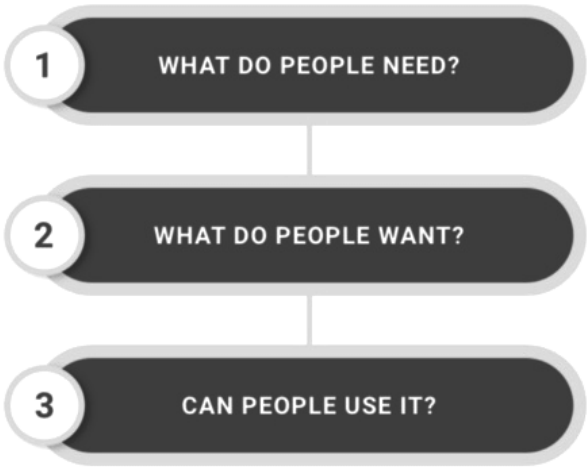


Рисунок 7.5 – Каркас питань для A / B-тестування

Таким чином можна протестувати стиль, зображення, контент, кнопки тощо.

A / B-тестування – це форма порівняльного аналізу кількісних характеристик двох робочих версій сайту, додатки тощо. Його мета – внесення змін, спрямованих на статистично значущі поліпшення тієї чи іншої дії користувача. Даний вид тестування вимагає глибокого розуміння статистики, щоб правильно скласти тест і проаналізувати його результати. Крім того, потрібно мати досвід, щоб вибрати саме ті цілі для тестування, які з більшою ймовірністю призведуть до потрібного результату.

Пам'ятайте: A / B-тестування корисно тільки в тому випадку, якщо у вас вже є готовий продукт (послуга), і вам потрібно його поліпшити. Не варто починати з A / B-тестів на стадії проектування – замість цього проведіть тестування за участю користувачів. Майте на увазі, що, хоча A / B-тестування відмінно підходить для отримання точних даних (які допоможуть вам переконати начальство, наприклад), – не всякий дизайн можна перевірити за його допомогою.

A / B-тестування (відоме також як спліт-тестування) – це метод дослідження, який полягає в порівнянні двох версій веб-сторінки, продукту, електронного листа або системи, і визначенні більш вдалою версії. Створюючи версії A і B, ви можете вносити зміни в інтерфейс, перевіряти гіпотези і покращувати сприйняття користувачів. Мета роздільного тестування полягає в тому, щоб побачити відмінності в поведінці двох груп користувачів і оцінити вплив кожної версії на їх дії.

Незважаючи на те, що за допомогою A / B-тестування ви визначите, який з двох інтерфейсів більш відповідний, ви не дізнаєтеся, чому. Саме тому так важливо проводити якісне тестування.

Наприклад, якщо ви коли-небудь помилково підписувалися на одну і ту ж інформаційну розсилку з двох різних акаунтів, то, ймовірно, помічали, що іноді теми одного і того ж листа в цих акаунтах відрізняються. Так відбувається, тому що автор цієї розсилки проводить A / B-тестування, щоб з'ясувати, який

заголовок більше привертає увагу одержувачів. Один лист може називатися «Зимова розпродаж. Знижка 10% на все трикотажні вироби », а друге -« Шапки, шарфи і рукавички за низькими цінами », при цьому зміст обох листів буде однаковим. В результаті автор розсилки зможе визначити, не тільки яка з версій листи набрала більшу кількість переглядів, а й які з адресатів в результаті вирішили зробити покупку.

Рекомендації з спліт тестування (A/B тестування)

A / B тестування (спліт тестування) – метод маркетингового дослідження, завдання якого – дізнатися які зміни і як можуть вплинути на цільовий показник.

Після запуску нових проектів багато з наших замовників задають питання: Як нам провести A / B тестування?

Що ми всі знаємо про A / B тестування?

"A / B тестування (спліт тестування) – метод маркетингового дослідження, суть якого – порівняння з набором тестових груп, в яких один або кілька показників були змінені, для того, щоб з'ясувати, які з змін покращують цільовий показник."

Для цих маніпуляцій вам буде потрібно досить великий розмір вибірки відвідувачів.

Виникає наступне питання: Наскільки великий повинен бути обсяг вибірки?

Необхідний розмір вибірки завжди залежить від трьох основних параметрів:

- поточний рейтинг конверсій
- поліпшення конверсії
- очікуване
- рівень довіри відвідувача

Ось так виглядають реальні дані багатьох сайтів в Інтернеті:

- Поточна конверсія на сторінці: 2%
- Очікуване поліпшення: 20%
- Рівень довіри відвідувачів: 95%

Розмір вибірки з розрахунку даних повинен складати: 13, 791 користувачів на один варіант при загальних 27 582 користувачів, на одну сторінку. У багатьох стартових проектів такий трафік, досягається за квартал, а то і за рік. Ми пропонуємо кілька по-справжньому робочих методів вирішення проблем, пов'язаних з низьким трафіком на сайті, які допоможуть вам вибудувати правильні A/B тести.

7.3. Більше гіпотез.

Творчо підійдіть до питань побудови альтернативної сторінки для тестування. Вона повинна кардинально відрізнятися.

Подумайте про цінності торговельної пропозиції і того, як вони зараз представлені на вашому сайті: доставка, гарантія, ціна і так далі.

Більше гіпотез і пропозицій – менше вибірка!

Так як в нашому прикладі кілька змін, то і відсоток очікуваного поліпшення теж буде вище, на відміну від попередніх даних.

- Поточна конверсія на сторінці: 2%
- Очікуване поліпшення: 50%
- Рівень довіри відвідувачів: 95%
- Розмір вибірки: 3 819 користувачів на один варіант.

Де попередні 13 791 користувачів і де наші 3 819.

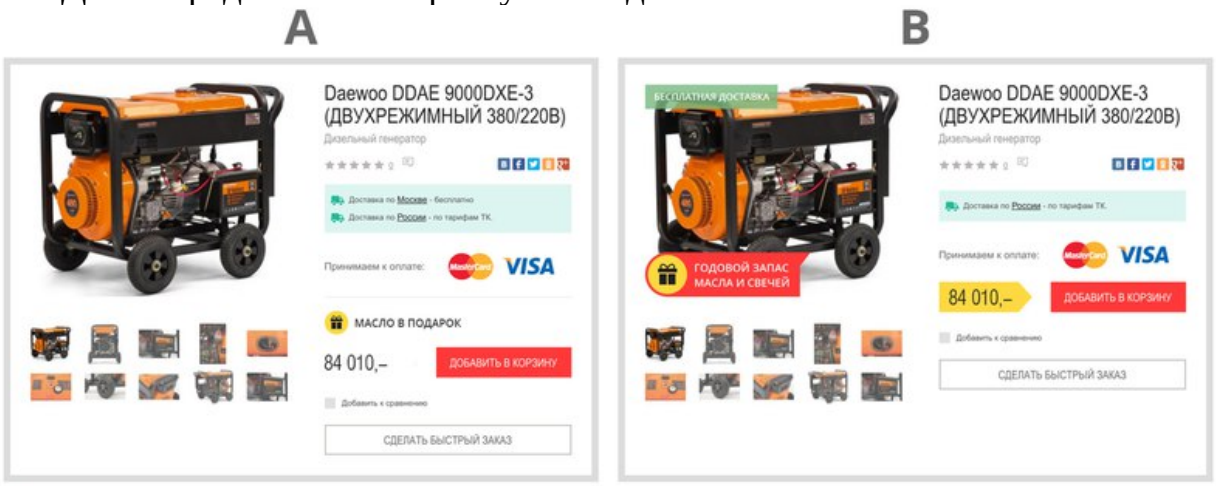


Рисунок 7.6 – Приклад A / B тестування

Хороший приклад безлічі змін на сторінці.

Недоліком даного підходу є те, що вам важко визначити який елемент став ключовим і вплинув на те, що виграв А чи В варіант. Ми вважаємо цей недолік малозначним, так як на старті будь плюс це плюс, конверсія збільшилася – ми перемогли.

7.3.1. Мультиваріантне тестування (MVT) і A / B / C /.... тестах.

MVT – це мультиваріантне тестування, що проводиться за принципом A / B тестування. Краще уникати такого тестування, адже кожна додаткова тестова зміна значно збільшує тривалість тесту.

Наприклад, ваш щотижневий трафік в середньому становить 4 100 відвідувачів. Якщо необхідний розмір вибірки для кожного варіанта становить 3 819 (як в прикладі раніше), то вам знадобиться 2 тижні тільки для того, щоб отримати результати тесту A/B. 3 тижні, щоб отримати результати тесту A / B / C і цілих 4 тижні, щоб отримати результати тестах в варіантах A / B / C / D.

Якщо тести проводяться більше 3-4 тижнів, існує великий ризик забруднення кожного з тестованих варіантів. Свята, акції, технічні проблеми на сайті – зовнішні чинники, від яких страждають результати тестування. Велика кількість відвідувачів може просто видалити свої файли cookie, які відповідають за те, які зміни бачать або не бачать ваші відвідувачі.

За даними компанії Econsultancy (звіт сформований на підставі 1600 онлайн-респондентів), 73% респондентів регулярно перевіряють свої настройки браузера, а також проводять чистку cookie. Коли їх запитали, що вони скажуть, якщо веб-сайт попросив їх дозволити встановлювати cookie при їх відвідуванні, тільки 23% відповіли "так" (60% відповіли "можливо").

Висновок – якщо ваш запланований експеримент A / B / C вимагає 4 тижні і більше, то розділіть його на два коротких A / B тесту. Повірте, якість даних буде набагато вище.

7.3.2. Мікро-конверсії

Оптимізація будь-якого ресурсу – це поліпшення ключових показників ефективності сайту. Показники ефективності прийнято розділяти на дві основні категорії:

Макро-конверсії. Це основні цілі сайту. Наприклад: оформлення замовлення, дзвінок, запит тестового періоду і так далі

Мікро-конверсії. Являють собою проміжні або додаткові кроки до макро-конверсії. Наприклад: перегляд відео, підписка на новини, репост, додавання в порівняння, перегляд на відвідувача і так далі.

З великою ймовірністю, мікро-конверсій на вашому сайті набагато більше ніж макро-конверсій, а показники їх в рази вище. Чим вище коефіцієнт конверсії, тим менше повинна бути довжина A / B тестування:

- Поточна конверсійна ставка: 10%
- Очікуване поліпшення: 20%
- Рівень довіри: 95%
- Розмір вибірки: 3 834 відвідувачів на один варіант

Найголовніше, поліпшення показників мікро-конверсій не завжди призводять до великого зростання макро-конверсій.

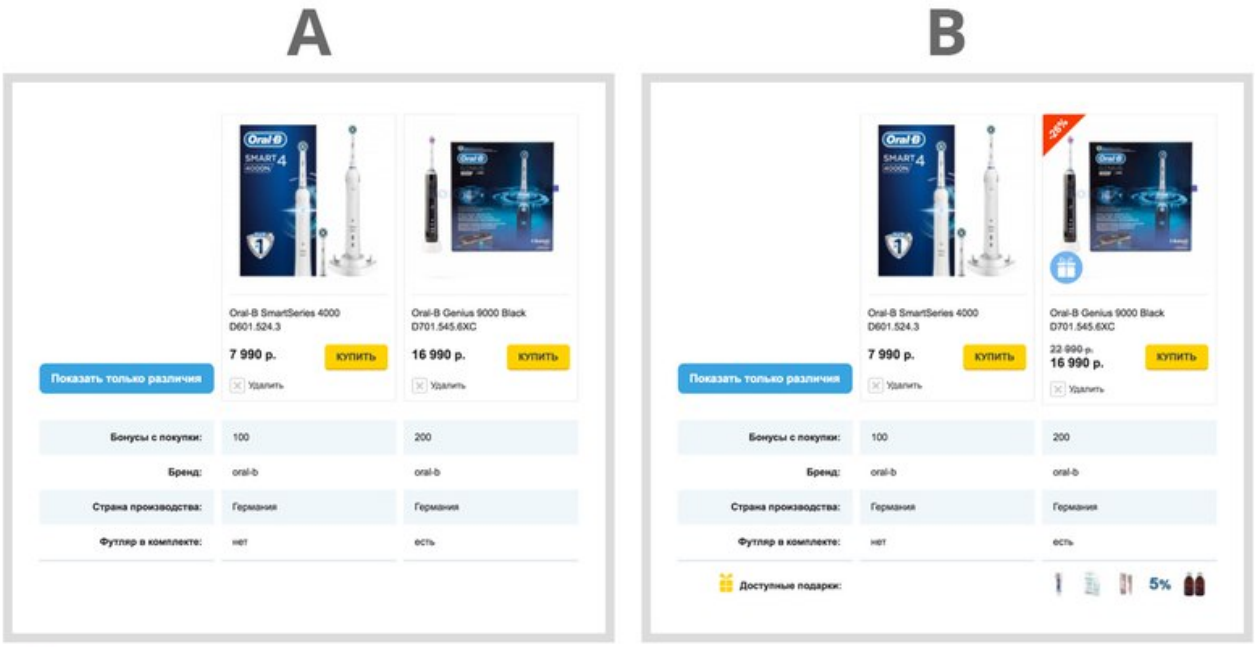


Рисунок 7.7 – Приклад підвищити показники мікро-конверсії за допомогою A / B тестування

За рахунок цього ходу, ми збільшили показники мікро-конверсій в два рази, але показник макро-конверсії зріс всього лише на 5%

7.3.3. Проведення A / B тестування всього сайту.

У ситуації, коли ваш сайт не має великої кількості трафіку, то не обмежуйте свої спліт-тести окремими сторінками або сегментами відвідувачів. Ви повинні використовувати в тестуванні максимальний обсяг трафіку, який отримує ваш сайт.

Класичні приклади використання тестування можуть включати тести заголовків, верхнього меню або взагалі тести нового макета сайту.

Недоліком всіх A / B тестів є необхідний обсяг відвідувачів для тестування. Тому, як згадували раніше, необхідно ширше дивитися на зміни. Однак складно продумати великі A / B тести для сайту, це майже те ж саме, що проводити тестування двох абсолютно різних веб-сайтів.

Проте, наша задача залучити якомога більше трафіку для тесту.

7.3.4. Збільште тестовий трафік A / B тестування

Найпростіше. Коли вам потрібно виконати важливий A / B тест, але зовсім немає часу чекати, ви можете просто тимчасово збільшити обсяг трафіку за допомогою рекламних кампаній або розсилок..

Це абсолютно законний спосіб збільшення обсягу трафіку і, отже, скорочення довжини A / B тестів. Однак цей спосіб теж таїть в собі багато небезпек. Результати подібних експериментів можуть бути забруднені особливістю залученого трафіку.

Це означає, що переможець тесту може бути тим, хто програв в звичайних »не тепличних" умовах. На результат може вплинути:

- якість трафіку
- аудиторія залученого трафіку
- час залучення трафіку

Те ж саме можна сказати про A / B тестування за часів Новорічних свят, Чорної п'ятниці і будь-яких інших свят. Ви повинні уникати тестування протягом цих періодів часу, оскільки ви отримуєте інший тип трафіку, ніж протягом поточного року.

7.3.5. «A / B тест» з використанням тесту в 5 секунд.

5 Seconds Test – простий інструмент юзабіліті, створений Usabilityhub. Ви завантажуєте свій макет сайту або скріншот, додаєте кілька питань, і реальні користувачі побачать його всього на 5 секунд, а потім дадуть відповідь на ваші запитання.

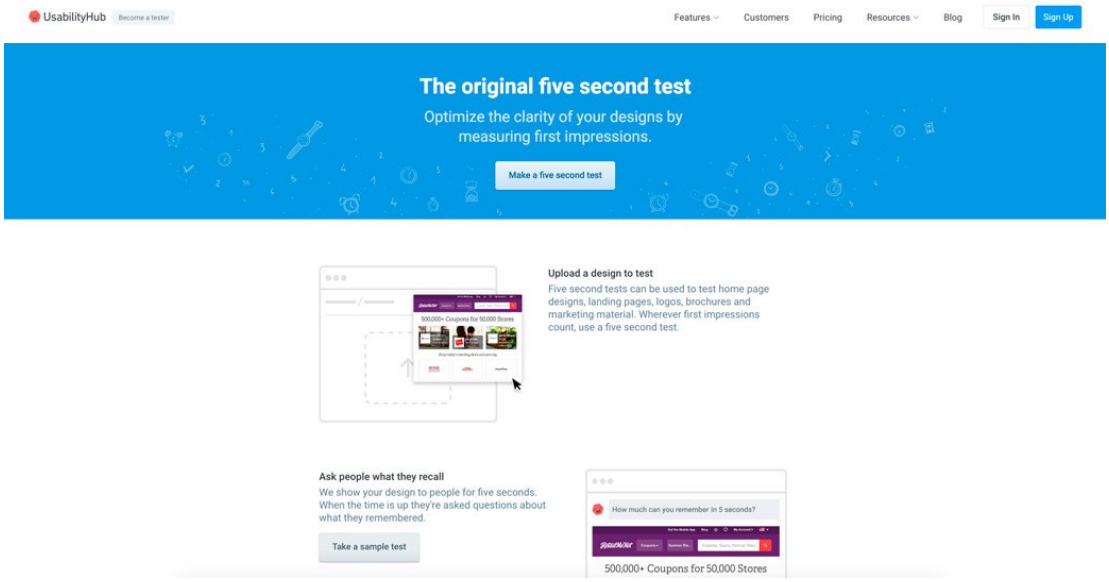


Рисунок 7.8 – 5 Seconds Test юзабіліті, створений Usabilityhub

Найшвидший тест можна провести за допомогою інструментів веб-розробника в браузері. Вносимо зміни в режимі реального часу, потім ви

знімаєте скріншот і завантажуєте як оригінальні, так і альтернативні версії в 5 Seconds Test і миттєво отримуєте результат.

Звичайно, це не справжні A / B тести, але ми вважаємо що це відмінний метод сформувати і перевірити невеликі і прості гіпотези. Наприклад:

- який колір
- який розмір шрифту

7.3.6. «B / A тестування»

Коли для проведення спліт-тесту у вас взагалі немає часу, і ви впевнені в змінах, зробіть, принаймні «B / A тестування» – до / після тестування.

В даному способі є теж свої підводні камені, вони пов'язані в першу чергу з підготовкою. Переконайтеся:

- що ви впроваджуєте зміни в стійкий до трафіку період (знову ж свята, зрізи сезонів)
- що робоча версія дійсно готова до публічної демонстрації (баги, стійка робота на всіх пристроях)
- уважно контролюйте трафік протягом усього часу B / A тесту

Одним з найголовніших переваг A / B тестування є той факт, що обидва варіанти працюють одночасно, тому зовнішні чинники, як правило, зводяться до мінімуму. Це правило не підходить до B / A тестуванням. І, незважаючи на це, ми змушені визнати, що іноді це найкращий і єдиний спосіб внести і контролювати зміни на сайті.

7.4. Відмінність ринкових досліджень від дизайн-досліджень

Ви можете продавати користувачам те, що вони сказали, що хочуть, але дослідження ринку не можуть розповісти вам про рішення проблем, які клієнти не можуть зрозуміти (Ерік Шмідт і Джонатан Розенберг) Основна відмінність між дослідженнями ринку та дизайнерськими дослідженнями полягає в тому, що дизайн-дослідження більш мінливі. Це означає, що дані засновані на тому, як люди відчувають і просто завдяки нашій людській природі з’єднуються з іншими, щоб прийти до розуміння, яке веде до змін. Мотивом дизайнерських досліджень є наближення до з’єднання з іншою людиною, щоб виробити значимість для їх цілей. Дослідження ринку часто засновані на логіці і необхідності компанії оцінити конкуренцію, але поряд з дизайн дослідженнями, обидва можуть використовуватися в поєднанні для створення кращого користувацького досвіду за допомогою встановлення контакту з користувачами і розуміння їх.

	ДОСЛІДЖЕННЯ РИНКУ	ДИЗАЙН-ДОСЛІДЖЕННЯ
Головна мета	Для генерації вартості часто фінансові) для організації	Для генерації значення (корисності) для кінцевого користувача
Процес	Систематичний збір теоретичних даних, які потім аналітично обробляються. обробляються.	Сукупний збір людського досвіду і артефактів, які потім синтетично обробляються.
Інструмент первинної обробки	Логіка	Емпатія
Підходи суспільної комунікації	Звіти та презентації, описання, діаграми та графіки для представлення лінійних аналізів і прогнозів ринку. Панують дані.	Мультимедійні презентації, описаня, фотографії, малюнки. Історії для представлення складних людських архетипів, потреб і зразків. Дані стають розповідними.
Дозволяє практикам	Зробити правильне рішення на основі визначених варіантів.	Розробити надійне рішення, яке відповідає виявленим потребам.

Рисунок 7.9 – Відмінності дизайн-досліджень від досліджень ринку

Дизайн-дослідження дозволяють нам зрозуміти складну поведінку людини, досягаючи кореня проблеми, розуміючи потреби, бажання і цілі користувача. Це також формує досвід користувача, який допомагає нам вирішити їх основні проблемні питання. В цілому, дані, які ми збираємо за допомогою проектних досліджень, дозволяють приймати рішення. Це призводить до застосування цих даних в корисних додатках, які дозволяють нам створювати продукти, що релевантні, доступні і придатні для користувачів і будь-яких людей, з якими ми працюємо: учасників проекту, продуктових менеджерів або інших дизайнерів у команді

Контрольні питання

1. Дайте визначення «Проектування UX», «Юзабіліті».
2. Дайте визначення термінам «Вайрфрейм», «Мокап» та «Прототип».
3. Опишіть відмінності та сферу застосування вайрфреймів, мокапів та прототипу
4. Типи дизайн-досліджень та юзабіліті-тестування. Проведення UX-досліджень

Теми для самостійного опрацювання

1. Психологія поведінки користувача
2. Ідеї Д. Нормана, Дж. Готельфа, та Дж. Гарретта та вплив їх ідей на розвиток UX
3. Побудова інформаційної архітектури проекту. Проектування взаємодії

Тестові завдання

1. Термін «Проектування UX» був введений у вжиток

a) у 1995 році Доном Норманом, який в той час займав пост віце-президента групи розробки технологій в Apple

b) у 1999 році Ів Сен Лораном, який в той час займав пост президента групи технологій тестування в Google

c) у 2005 році Сером Джонсом, який в той час займав пост генерального директора в Sun Microsystems

d) у 2009 році Генрі Фордом, який в той час займав пост виконуючого обов'язки директора в Microsoft

2. Якість характеру, яка є необхідною для проектування юзабіліті

a) Емпатія
b) Допитливість
c) Ясність вираження
d) Ергономічність

3. У якому році з'явилися перші сайти в адаптивному стилі верстки

a) 2014 b) 2013 c) 2011 d) 2012

4. ***Термін «Проектування UX» був введений у вжиток***

a) у 1995 році Доном Норманом
b) у 1982 році Аланом Купером
c) у 1950 році Джоном Тернером
d) у 1925 році Дерсі Уайт

5. Юзабіліті – це

a) здатність розуміти, чому люди чинять той чи інший спосіб
b) бажання дізнатися, чому люди чинять ту чи іншу дію.
c) простота використання і освоєння об'єкта, створеного людиною
d) здатність пояснити складні поняття в доступній формі для тих, хто не знайомий або мало знайомий з даною галуззю.

6. Емпатія – це

a) здатність розуміти, чому люди чинять той чи інший спосіб
b) бажання дізнатися, чому люди чинять ту чи іншу дію.
c) простота використання і освоєння об'єкта, створеного людиною
d) здатність пояснити складні поняття в доступній формі для тих, хто не знайомий або мало знайомий з даною галуззю.

7. Якісні характеристики

a) спрямовані на те, щоб визначити, чому користувачі поводяться саме так, а не інакше.
b) визначають, скільки користувачів роблять що-небудь певним чином. Одним словом, допомагає вимірювати різні ситуації чисельно.
c) поведінка і реакція користувача, якщо ви хочете отримати корисні дані
d) нічого з переліченого

8. Кількісні характеристики

a) спрямовані на те, щоб визначити, чому користувачі поводяться саме так, а не інакше.
b) визначають, скільки користувачів роблять що-небудь певним чином. Одним словом, останній метод допомагає вимірювати різні ситуації чисельно.
c) поведінка і реакція користувача, якщо ви хочете отримати корисні дані
d) нічого з переліченого

9. Дослідження користувацької аудиторії не здійснюється засобами

a) інтерв'ю з замовником
b) контекстний запит
c) фокус-групи
d) створення персон

10. Оберіть не існуючі типи тестування

a) A / B-тестування
b) спліт тестування
c) мультиваріантне тестування
d) інверсний тест

Практичні завдання

Завдання 7.1. Створити персони відповідно до цільової аудиторії ресурсу;
Завдання 7.2. Створити карту емпатії;
Завдання 7.3. Здійснити A / B-тестування;
Завдання 7.4. За необхідності внести корективи у розроблений користувацький інтерфейс.

8. ДИЗАЙН ПІД МОБІЛЬНІ ПРИСТРОЇ

План:

8.1. Дизайн для мобільних пристроїв.
8.2. Чому дизайн для мобільний пристроїв важливий
8.3. Responsive Design
8.4. Adaptive Design
8.4.1. Навіщо потрібен Adaptive Design?
8.4.2. Принципи Adaptive Design
8.4.3. "Mobile First"
8.4.4. Типи макетів Adaptive Design
8.4.4.1. Гумовий
8.4.4.2. Перенесення блоків
8.4.4.3. Перемикання макетів
8.4.4.4. Адаптивність «малою кров'ю»
8.4.4.5. Панелі
8.4.5. Основні помилки при розробці Adaptive Design
8.5. Порівняння Responsive Design та Adaptive Design
8.6. Створення адаптивного веб-дизайну
8.6.1. Починаєте «з нуля»
8.6.2. Витратити не багато коштів
8.6.3. Навіть коли з’являться нові мобільні пристрої
8.6.4. Визначення цілей для розробки мобільної версії
8.6.5. Три основні підходи до створення адаптивного дизайну.
8.6.5.1. Мобільна версія сайту.
8.6.5.2. Мобільний додаток.
8.6.5.3. "Гумовий" дизайн.
8.6.6. Процес
8.6.8. Цілі користувачів на різних пристроях
8.6.9. Технічні інструкції для роботи над функціональністю і контентом 8.6.10. Прототипування
8.6.11. Початок
8.6.12. Створення головного макету
8.6.13. Головна сторінка
8.6.14. Головне меню
8.6.15. «Підвал» сторінки
8.6.16. Інші компоненти
8.6.17. Тестуйте
8.6.18. Зовнішній вигляд
8.6.19. Стиль сторінки
8.6.20. Підбір шрифтів
8.6.21. Верстка сайту

Швидкий розвиток мобільних пристроїв потребує сучасного підходу до розробки веб-сайтів. Інтернет-користувачі переглядають сайти на різних екранах різних розмірів. Розміри екранів завдяки зміні ландшафтної та портретної орієнтації мобільного пристрою змінюються навіть під час однієї сесії по декілька десятків раз, тому важливо, аби сайт адаптувався до будь-якого з них.



Рисунок 8.1 – Портретний режим та зміна на ландшафтний

Підходом до вирішення цієї проблеми є створення окремої мобільної версії сайту, що розміщується на окремому домені виключно для перегляду з мобільних пристроїв. На сьогодні розробники відмовляються від окремих мобільних версій веб-сайтів и створюють єдину версію, яка працює та адаптується безпосередньо під всі пристрої: стаціонарні комп'ютери, ноутбуки, планшети або смартфони.

8.1. Дизайн для мобільних пристроїв.

Нам потрібно думати про веб-сайти як про гнучкі, не статичні форми. Вам безперечно варто планувати заздалегідь, як ваш сайт буде відображатися на різних пристроях.

Вам потрібно думати про гнучкість вашого дизайну весь час. Текст легко може почати наїжджати сам на себе на пристроях з маленькими екранами, і ви можете тільки змінити розмір шрифту. Це не так просто зробити з зображеннями або з великими блоками статичного контенту.

Уявіть, що у вас є великий графічний банер на протязі всієї ширини екрану (темно-сірий прямокутник на зображенні знизу). Коли ви пропорційно змінюєте розмір даного елемента для більш дрібних екранів, він стане занадто маленьким і ви не зможете його розгледіти на мобільних пристроях.

Так, тепер, припустимо, у нас є дизайн з 4-х квадратів (зелені квадрати на малюнку нижче) – всі елементи знаходяться на одному рядку на персональних комп'ютерах. Ці 4 квадрата можуть бути перенесені в дві колонки по два квадрата на планшетних комп'ютерах і 1 квадрат на сходинку на екранах мобільних пристроїв. Працювати з маленькими графічними елементами набагато простіше.



Рисунок 8.2 – Приклад побудови рідкого макету

Ви можете зробити ваш дизайн абсолютно "рідким", це означає, що відображається 100% ширина на всіх екранах і все вимірювання йдуть в процентах. Тут немає фіксованої ширини.

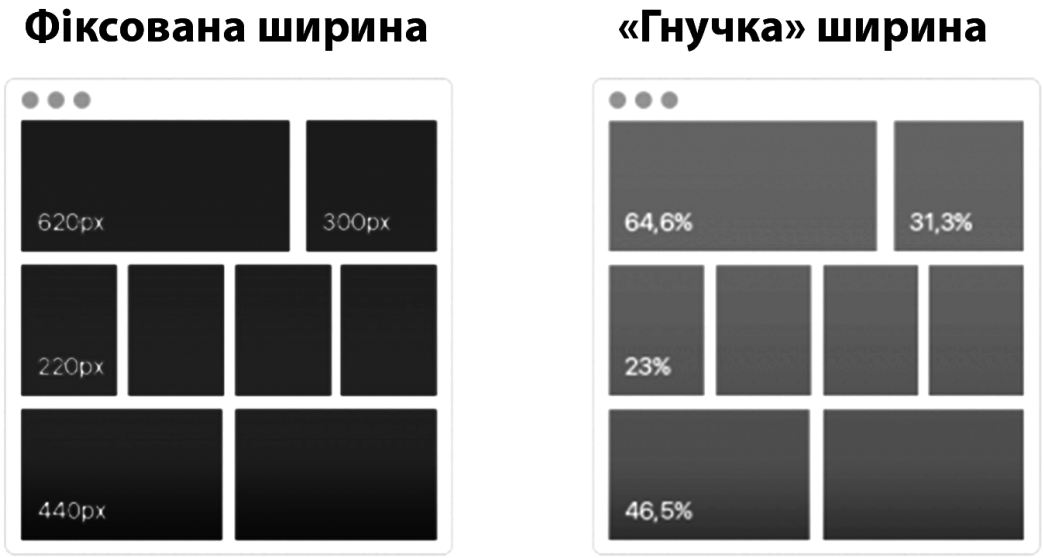


Рисунок 8.3 – Фіксована ширина проти «гнучкої»

Підтримка та розробка дизайну "рідкого" макета є досить складним завданням через нескінченної кількості можливостей того, як може виглядати ваш дизайн в різних ситуаціях.

Так чи інакше, найпоширеніший спосіб, це створення дизайну з фіксованою шириною для моніторів персональних комп'ютерів і створення «рідкого» (100% ширина) макета для планшетів і мобільних телефонів. Таким способом ви можете бути впевненим, що ваш дизайн не буде занадто великим, якщо хтось запустить ваш сайт на великому моніторі, і він так само адаптується до більшості мобільних пристроїв. Щоб переконатися, що веб-сайт повністю чуйний і виглядає чудово на всіх екранах, вам потрібно подбати про це під час фази написання коду вашого проекту.

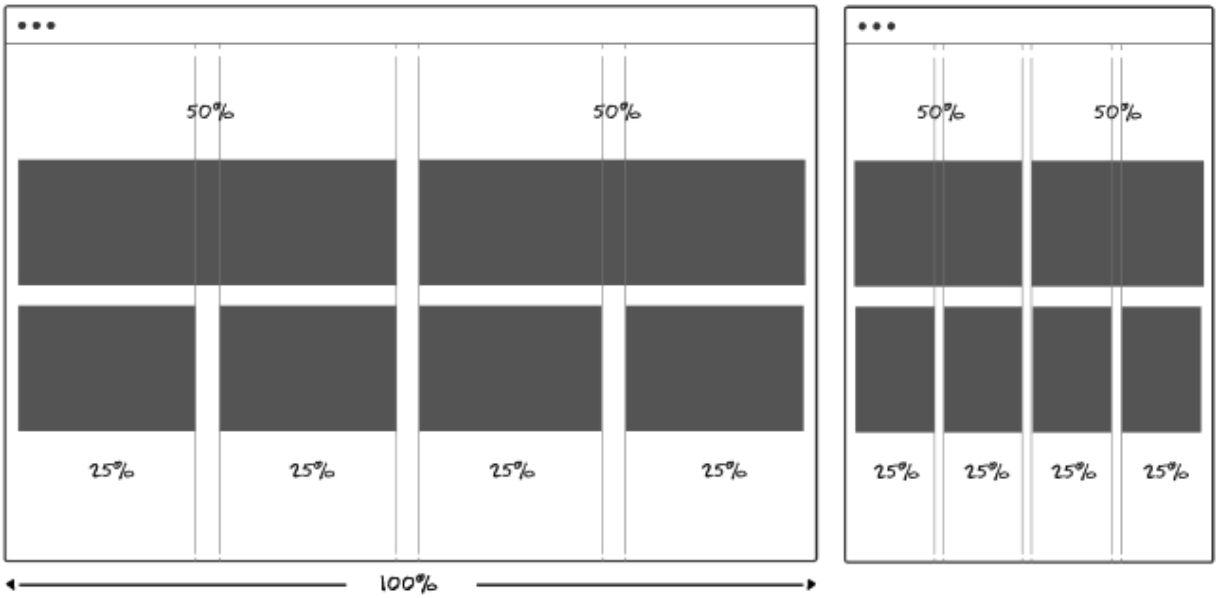


Рисунок 8.4 – Приклад дії «гнучкої» ширини

8.2. Чому дизайн для мобільний пристроїв важливий

Напевно, ви вже помітили, що мобільних пристроїв стає все більше, і, відповідно, людей, які заходять в Інтернет, використовуючи такі пристрої, теж стає більше. Так, 600 мільйонів з 1 мільярда користувачів Facebook заходять на сайт через мобільні пристрої, згідно з нещодавньою статистикою, 15% трафіку фандрайзингових сторінок і сторінок з пожертвуванням в США припадають на мобільні пристрої.

Тому стає все важливішим мати адаптивний дизайн сайту, який буде зручний всім користувачам, незалежно від пристроїв, які вони використовують. Повірте, якщо їм не потрібно буде зменшувати, збільшувати, повертати і здійснювати будь-які ще маніпуляції для того, щоб прочитати вашу останню новину, вони будуть дуже вдячні, і ймовірність того, що вони зайдуть на ваш сайт знову сильно зросте.

Існує два основні підходи для створення сайтів, що легко адаптуються для різних типів пристроїв:

8.3. Responsive Design

(Responsive Web Design, RWD) – чуйний дизайн – проектування сайту з певними значеннями властивостей, наприклад, гнучкою сіткою макета, що дозволяють одному макету працювати на різних прилаштувати;

Responsive Design об'єднує три методики – гнучкий макет на основі сітки (fluid grid), гнучкі зображення (fluid images) та медіазапити. Пропорції та розміри елементів задаються у відсотках. При зменшенні ширини Сторінки весь вміст плавно стискається, структурні елементи зменшуються відносно один одного. Так, наприклад, якщо веб-сайт МАВ 3-х колонкову структуру, то на вузьких екрані ВІН буде мати дві або одну колонки контенту.

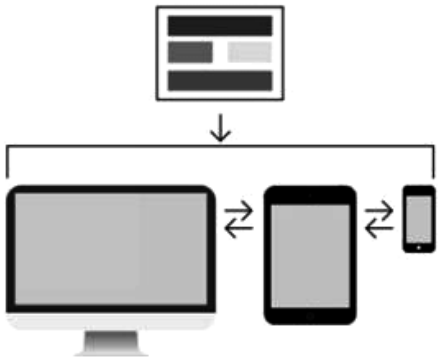


Рисунок 8.5 – Макет для «гнучкого» дизайну (Responsive Design)

Для створення чуйної Версії веб-сайту Використовують медіа запити (media queries) – блоки коду із зазначенням параметрів екранів. Медіа-запити дозволяють застосовувати певні правила (стилі) для виведення контент-блоків у різний порядок та пропорції залежних від ширини екрана та можливість пристрою, на якому відбувається перегляду сайту.

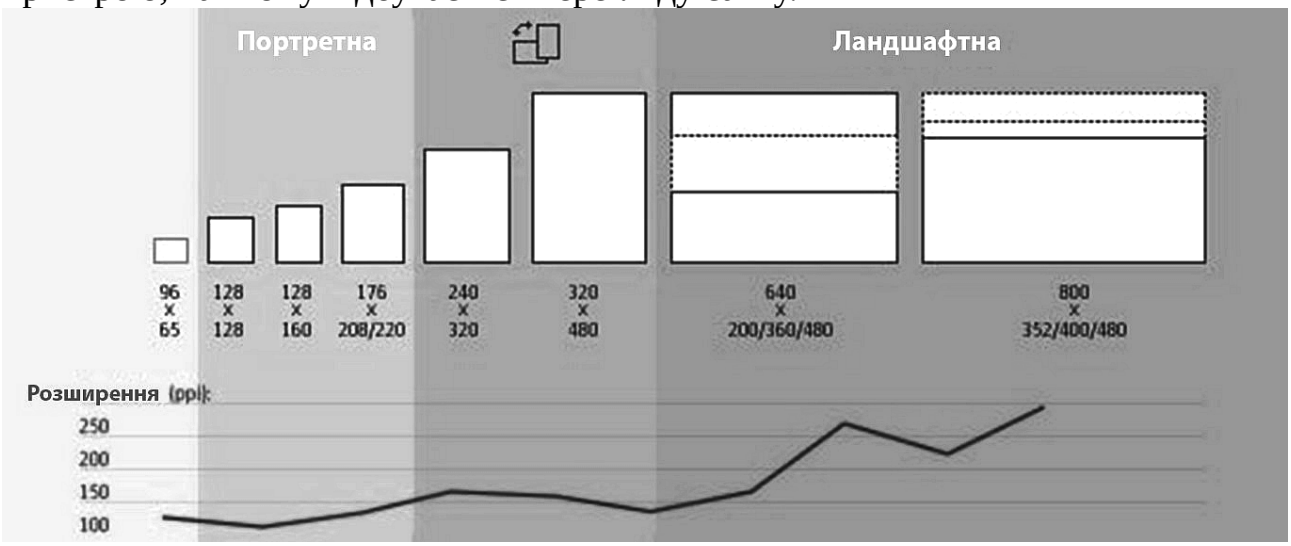


Рисунок 8.6 – Розмірна сітка та шкала співвідношення до розширення

8.4. Adaptive Design

(Adaptive Web Design, AWD) – адаптивний дизайн – проектування сайту, що базується на кількох макетах фіксованої ширини.

Чуйний та адаптивний веб-дизайн тісно пов'язані и вирішують одну задачу, но в різний спосіб. Основна відмінність між цими прийомами – «гнучкий» дизайн – один макет для всіх пристроїв, адаптивний дизайн – різні макети для різних пристроїв. Їх поєднання Забезпечує Ідеальну формулу для функціональних сайтів.

8.4.1. Навіщо потрібен Adaptive Design?

1) **Велика розмаїтість пристроїв**, з яких можна виходити в Інтернет. В даний час існує безліч пристроїв, якими люди користуються, в тому числі, і для того, щоб виходити в Інтернет. Всі ці пристрої розрізняються розміром екрану, дозволом і, відповідно, тим, як може відображатися на них веб-сайт. Тому важливо, щоб ваш сайт добре виглядав і правильно відображався у будь-якого з користувачів, незалежно від того, який пристрій він використовує.

2) **Популярність мобільних пристроїв** з виходом в Інтернет і збільшення мобільного Інтернет-трафіку. З ростом популярності мобільних пристроїв кількості користувачів, які заходять з них на сайти, помітно збільшилась, тому просто ігнорувати їх вже не можна – це не одна-дві людини на півроку, це значна частина вашої аудиторії, і їм повинно бути зручно користуватися вашим сайтом (інакше вони не будуть цього робити).

3) **Термінова інформація.** Якщо ваш ресурс містить новинну / термінову інформацію, і висока ймовірність, що користувачеві може знадобиться прочитати цю інформацію саме з телефону (бо інших пристроїв у нього під рукою немає) в даний момент часу, потрібно подбати про те, щоб у нього була можливість це зробити.

8.4.2. Принципи Adaptive Design

Проектування починається з адаптивною версії веб-сайту для мобільних пристроїв. На цьому етапі дизайнери прагнуть правильно передати зміст і основні ідеї з використанням невеликого екрану і всього однієї колонки. Вміст при необхідності скорочують, видаляючи другорядні інформаційні блоки і залишаючи найважливіше.

- Проектування для мобільних пристроїв на ранніх етапах ("mobile first");
- Застосування гнучкого макета на основі сітки (flexible, grid-based layout);
- Використання гнучких зображень (flexible images);
- Робота з медіазапитами (media queries);
- Застосування поступового поліпшення.

На відміну від Responsive Design, адаптивний дизайн орієнтується на розміри пристроїв. Він використовує кілька статичних макетів для різних типів пристроїв, базуючись на контрольних (переломних) точках. Макети завантажуються при питань комерційної торгівлі розмірах вікна браузера пристрою, а переходи між макетами відбуваються стрибкоподібно, а не плавно.

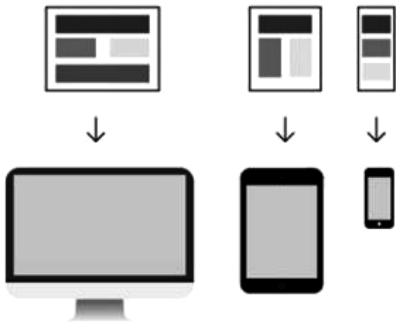


Рисунок 8.7 – Макети для адаптивного дизайну

В адаптивному дизайні Серверні скрипти спочатку визначаються тип пристрою, з допомогою якого користувач намагається отримати доступ до сайту (настільний ПК, телефон або планшет), потім завантажує саме ту версію Сторінки, яка найбільш оптимізована для нього. Для елементів сітки задаються фіксовані розміри.

8.4.3. "Mobile First"

Майже 60% інтернет-користувачів використовують мобільні пристрої для доступу до мережі. Виходячи з цього, став популярним процес проектування, де

необхідно починати з простих макетів для маленьких екранів, а потім додавати складність до цієї основі в міру того, як з'являється вільний простір.



Рисунок 8.8 – Ілюстрування технології процесу проектування "Mobile First"

8.4.4. Типи макетів Adaptive Design

8.4.4.1. Гумовий

Простий в реалізації і очевидний для користувача тип представлення сайту. Основні блоки стискаються до ширини екрану мобільного пристрою, де таке неможливо – перебудовуються в одну довгу стрічку.

8.4.4.2. Перенесення блоків

Очевидний спосіб для багато-колоночного сайту: при зменшенні ширини екрану додаткові блоки (сайдбарі) переносяться в нижню частину макета.

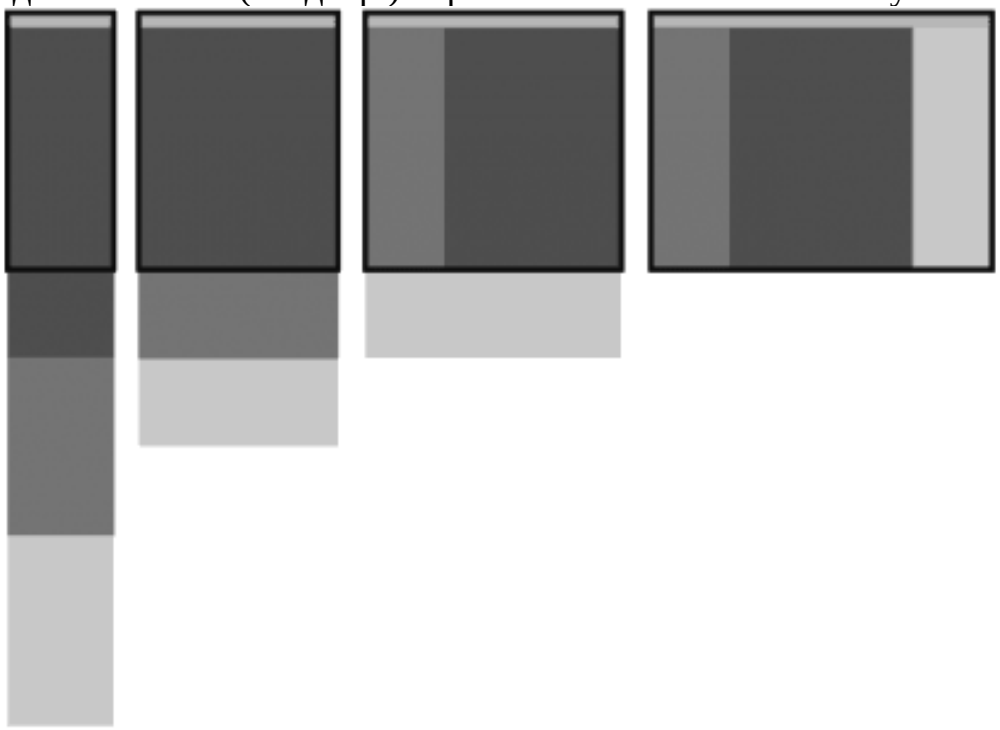


Рисунок 8.9 – Перенесення Блоків.

8.4.4.3. Перемикання макетів

Цей спосіб найбільш зручний при читанні сайту з різних пристроїв: під кожний дозвіл екрана розробляється окремий макет. Спосіб трудомісткий, тому менш популярний, ніж попередні два.

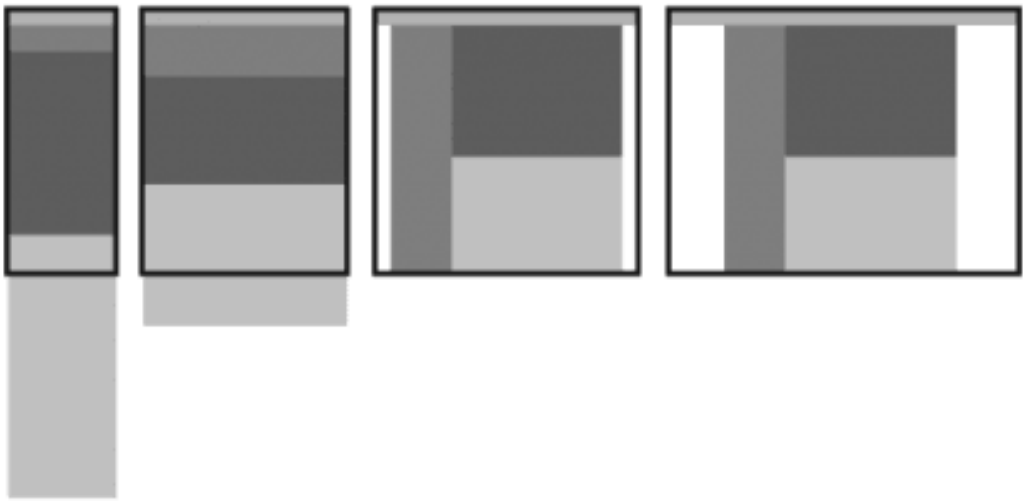


Рисунок 8.10 – Перемикання Макетів.

8.4.4.4. Адаптивність «малою кров'ю»

Дуже простий спосіб, який підходить для нескладних сайтів. Досягається елементарним масштабуванням зображень і типографіки. Не дуже популярний, тому що не є гнучким.



Рисунок 8.11 – Адаптивність "Малою Кров'ю".

8.4.4.5. Панелі

Спосіб, що прийшов з мобільних додатків, де додаткове меню може з'являтися при горизонтальному або вертикальному тапі. Головний недолік – неочевидність дій для користувача: дуже незвично бачити мобільну навігацію на веб-сайті. Але з часом спосіб може стати досить популярним.

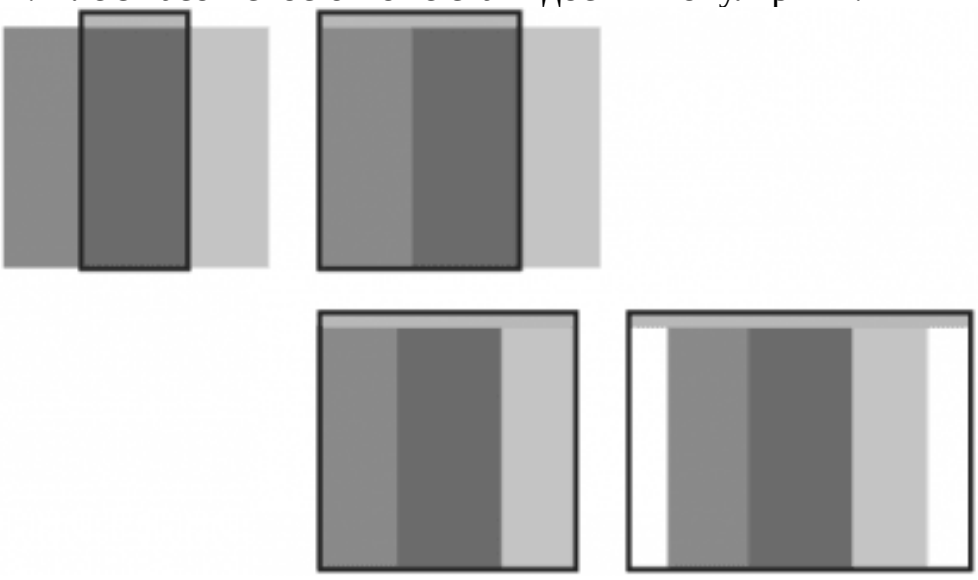


Рисунок 8.12 – Панелі.

Потрібно пам'ятати, що представлені вище макети не є універсальними рішеннями – для кожного проекту необхідно вибирати найбільш оречний для потреб і можливості спосіб.

8.4.5. Основні помилки при розробці Adaptive Design

Адаптивний дизайн стає все популярнішим, сьогодні це вже не просто модна нова технологія. Нижче зібрані найчастіші помилки, яких припускаються при розробці адаптивного сайту, своєрідний чек-лист.

Приховувати контент

Не варто ховати частини сайту тільки тому, що вони не поміщаються на екрані мобільного пристрою. Мобільні користувачеві хочуть бачити той же контент і функціонал, що і користувачі десктопів. Потрібно проектувати інтерфейс так, щоб по можливості містилося все.

Чи не оптимізувати під touch-пристрої

Більшість мобільних пристроїв з touch-управлінням, тому потрібно обов'язково оптимізувати елементи інтерфейсу під нього.

Не оптимізувати типографіку

Розміри екранів, на яких може відображатися сайт, дуже різні, тому необхідно змінювати типографіку так, щоб користувачеві було зручно читати тексти.

Використовувати занадто великі файли

Мобільний інтернет не завжди може дозволити завантажувати занадто важкі сайти, потрібно пам'ятати про це при розробці адаптивного дизайну.

Використовувати занадто великі зображення

По суті це окремий випадок попередньої помилки, який зустрічається надзвичайно часто. Будь-які зображення необхідно оптимізувати, як по вазі, так і за розміром.

Спрощувати дизайн

При розробці адаптивно сайту дуже заманливо робити примітивний дизайн, який просто «складеться» на екрані смартфона. Потрібно робити дизайн, який буде максимально зручний на будь-якому пристрої.

Не думати про контент

Вміст сайту може бути досить різноманітно – великі тексти, фотографії, відео, таблиці і багато іншого. Необхідно продумувати, як все це буде відображатися на сторінках сайту.

Не думати про великих екранах

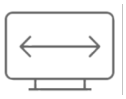
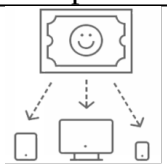
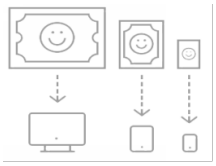
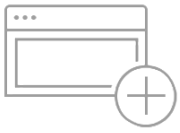
Адаптивний дизайн повинен працювати не тільки на маленьких і звичайних екранах, але й на величезних моніторах і телевізорах. Потрібно тестувати сайт і для них.

Відключати зум

На багатьох адаптивних сайтах розробники відключають можливість ресайз сайту на тач-пристроях, прописуючи це в мета-тезі viewport, і це може сильно дратувати користувачів.

8.5. Порівняння Responsive Design та Adaptive Design

Мобільні версії сайтів і мобільні додатки, спеціально розроблені для різних мобільних пристроїв, також вирішують проблему з зручністю перегляду сайту, але мають деякі недоліки.

	Responsive Design	**Adaptive Design**
Визначення пристрою	 Використовують медіа запити, що формують структуру сторінки відносно розмірів екрану пристрою, для всіх пристроїв використовується один шаблон.	 Використовується серверних або браузерні визначення пристрою, после чого застосовується конкретний шаблон саме для цього типу пристрою.
Оптимізація контенту	 Завантажується весь контент, для всіх типів пристроїв, не залежно від того, використовується він чи ні	 Завантажується підготовленій, та оптимізований контент під конкретний пристрій
Адаптація під пристрій	 Використовується один шаблон для всіх пристроїв. використовують однакові блоки, модулі, медіафайли, як для мобільного, так и для десктопного перегляду.	 Створюються різні унікальні шаблони, що призначені для різних пристроїв. Поведінка сайту змінюється в залежності від типу пристрою.
Швидкість завантаження	 Швидкість завантаження нижча, користувач чекає, поки завантажеться весь контент и всі структурні елементи для всіх розмірів екранах	 Швидкість завантаження вище, завантажується оптимізований контент, створений безпосередньо для певного пристрою и розміру його екранах
Розробка та запуск	 Треба переробка основного сайту. При проектуванні та розробці сайту треба враховувати та оптимізувати взаємодію користувача відразу під всі можливі пристрої.	 Більш витратний, але без змін існуючого сайту Адаптивні макети можуть розроблятися на базі існуючого сайту, без его зміни. Етап проектування і дизайну триває довше, і відповідно витратні

1) Під кожен тип операційної системи потрібно свій додаток / версія сайту. Це вимагає додаткових ресурсів, як тимчасових, так і грошових.

2) Необхідність завантажити програму. Для того, щоб користуватися вашим додатком, користувачам необхідно його завантажити. Це вимагає якихось додаткових зусиль від користувачів, і багато хто не будуть цього робити, якщо точно не впевнені, що додаток їм дуже потрібно і вони планують регулярно його використовувати.

3) Поділ трафіку. З точки зору просування сайту мобільний додаток не зручно тим, що розділяє весь трафік ресурсу на трафік сайту і трафік додатки, що буде виглядати, як менша відвідуваність сайту.

4) Необхідність інтеграції матеріалів сайту. У випадку з мобільним додатком необхідно або синхронізувати сайт з додатком (додаткові ресурси), або робити подвійну роботу по наповненню сайту і додатки матеріалами.

На відміну від мобільних додатків, адаптивний дизайн – це одна адреса сайту, один дизайн, одна система управління і зміст сайту. Безумовно, адаптивний дизайн також має свої мінуси, головним з яких є відносна новизна технології і, як наслідок, брак хороших фахівців і знань про проектування адаптивних сайтів. «Гнучкий» дизайн більше підходять тоді, коли немає різниці між переглядом та використанням веб-сайту на мобільному и десктопному пристрої, а також коли бюджет и срок розробки обмежені. Адаптивні сайти найбільш затребувані тоді, коли важливою є швидкість завантаження сторінок та взаємодія користувача з мобільною версією сильно відрізняється від використання десктопної версії.

Кожний випадок створення мобільної версії сайту необхідно розглядати окремо. Для простоти прийняття рішення нижчих наведено порівняльну таблицю використання чуйного и адаптивного дизайну.

8.6. Створення адаптивного веб-дизайну

Є багато методів вирішення поставленої задачі, коли клієнт звертається з проханням створити мобільне рішення для його сайту. Остаточний вибір залежить від бізнес задач клієнта, а також від його бюджету. Безперечно, створення гнучкого веб-сайту не є повною мобільною стратегією, і він не зможе вирішити всі необхідні завдання. Але його варто розглянути як дуже серйозну опцію, особливо в тому випадку, коли ви створюєте сайт «з нуля».

8.6.1. Починаєте «з нуля»

Розробка нового сайту або мобільного додатку зазвичай являє собою трудомісткий процес. Ви ніколи не знаєте, чи буде сайт успішним, до того моменту, як він з’явиться в Інтернеті. Тому створення окремого мобільного сайту або мобільного додатку в тандемі з сайтом може виявитись звичайною втратою часу та коштів. Через це більш ефективним є досягти якісного виконання вже існуючого сайту перед тим, як братись за створення мобільних рішень для нього.

8.6.2. Витратити не багато коштів

Якісні гнучкі рішення вимагають додаткового часу на дизайн та розробку інтерфейсу, але при цьому не надто вимагають переробки при створенні мобільної версії. Ви витрачаєте в цілому на 20-30% більше часу на створення «чутливого» сайту, при цьому економлячи час на створенні окремого додатку. Написання сайту за цією методикою також означає, що розкручувати і працювати в майбутньому доведеться лише з одним сайтом, що також значно економить кошти.

8.6.3. Навіть коли з’являться нові мобільні пристрої

Звичайний мобільний сайт розпізнає пристрій користувача, тому коли у використанні з’являться нові моделі, сайт повинен бути оновлений. Але гнучке рішення розпізнає лише ширину браузера, тому з появою нових моделей мобільних пристроїв ніяких оновлень робити не потрібно.

Після того, як ви вирішили, що ваш сайт повинен бути більш адаптивним, ви можете слідкувати трьох основних рекомендацій.

8.6.4. Визначення цілей для розробки мобільної версії

Визначте, які ще цілі, крім зручності і читання з усіх пристроїв, є у мобільній версії сайту. Запитайте себе – чи будуть у мобільних користувачів інші потреби, ніж у тих, хто заходить на сайту з комп'ютера?

Наприклад, користувачі сайту музею, які заходять з мобільних пристроїв, ймовірно хочуть мати швидкий доступ до розкладу і можливість купити квиток за допомогою телефону. Вистоюючи структуру сайту таким чином, щоб це було зручно зробити, ви збільшуєте кількість потенційних відвідувачів музею.

Подумайте про те, що можуть захотіти зробити користувачі, які відвідують ваш сайт, і організуйте його мобільну версію відповідним чином.

8.6.5. Три основні підходи до створення адаптивного дизайну.



Рисунок 8.13 – Приклад створення адаптивного дизайну

8.6.5.1. Мобільна версія сайту.

Створити мобільну версію сайту – це значить створити абсолютно окремий сайт, призначений тільки для мобільних пристроїв. Перевагою цього підходу є те, що це версія тільки для мобільних пристроїв, і ви можете спроектувати її таким чином, щоб вона була максимальна зручна для користувачів даних пристроїв. Недоліки – це витрати, як тимчасові, так і грошові.

8.6.5.2. Мобільний додаток.

Мобільний додаток для вашого сайту може бути ще більш зручним з точки зору користувацького досвіду – за рахунок таких додаткових функцій, як доступ до сайту оффлайн, можливість налаштування повідомлень, інтеграція з соціальними мережами тощо Зворотною стороною також є висока вартість і необхідність постійної роботи з додатком. Крім того, користувачам необхідно буде спеціально завантажувати додаток.

8.6.5.3. "Гумовий" дизайн.

Це найпростіший і дешевий підхід до створення адаптивного веб-дизайну. "Гумовий" означає те, що сайт автоматично змінює розмір і розташування змісту, в залежності від розміру екрану пристрою, з якого його переглядають.

8.6.6. Процес

Давайте розглянемо процес створення гнучкого веб-сайту на прикладі готельного веб-сайту, який включав нову структуру, вичерпну інформацію про готель та нову можливість онлайн-бронювання. Нижче наводяться ключові кроки, які були здійснені в роботі над проектом:

5. **Дослідження та пріоритети.** Розуміння того, яким додатковим вимогам повинен відповідати гнучкий веб-сайт;
6. **Прототипування.** Створення модульної структури сайту та його макетів у відповідності до різної ширини екрану;
7. **Зовнішній вигляд.** Вирішення стилістичних питань;
8. **Верстка сайту.** HTML та CSS.

8.6.7. Дослідження та пріоритети

Дослідження завжди важливий етап у розробці, тому варто докласти зусиль і подумати про користувачів, які будуть використовувати різноманітні мобільні пристрої, відвідуючи даний сайт. Розуміння того, що та яким чином вони хочуть бачити на цьому сайті і дасть нам змогу правильно розставити пріоритети даного проекту.

8.6.8. Цілі користувачів на різних пристроях

Відповіді на ці питання на даний момент стають надзвичайно багатогранними. В минулому легко було здогадатись, що відвідувачі сайтів через мобільні телефони намагалися знайти потрібну адресу або швидко забронювати необхідну послугу. Але зараз користувачі стали більш схильні не поспішаючи переглядати сайти в Інтернеті, оскільки в них немає необхідності якомога скоріше завершити необхідну задачу.

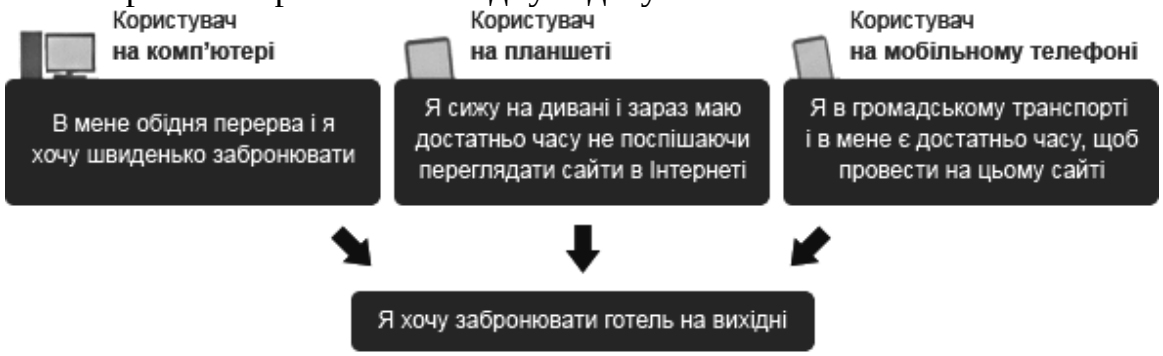


Рисунок 8.14 – Особливості використання девайсу відповідно до потреб

8.6.9. Технічні інструкції для роботи над функціональністю і контентом

Потрібно продумати те, як складна функціональна система може працювати на різноманітних мобільних пристроях. Між тим, гнучкий веб-сайт буде змінювати лише CSS в залежності від ширини екрану пристрою, тому деякі елементи JavaScript можуть бути спотворені і їх краще взагалі приховати.

8.6.10. Прототипування

Логіка зміни стилів може спочатку здатися тяжкою для визначення і може трапитись так, що правильна відповідь прийде вже в момент написання сайту. Але починати з чогось треба, тому пропонуємо розглянути 3 базових ширини

екрану: настільний комп’ютер, iPad та iPhone. У випадку з нашим веб-сайтом це задовільнило вимоги, але для когось цього може бути недостатньо і вам можуть знадобитись інші варіанти ширини.

На цьому етапі роботи ви вже повинні мати ключові шаблони, які ви збираєтесь використати для прототипу. Головною метою тут є визначення логіки того, яким чином CSS змінить вигляд сторінок, а також сфокусуватись на тих сторінках, схеми побудови яких будуть значно відрізнятися. Для сайту, приклад якого ми розглядаємо, ми обрали головну сторінку, всі сторінки, які відкриваються в процесі бронювання, сторінки готелів, а також сторінки з цінами на проживання.

8.6.11. Початок

Для початку необхідно визначити модульну сітку для кожної ключової ширини. Ми створили три сторінки для наступної ширини екранів: 1024px (робочий стіл), 768px (портретний режим для планшету), 320px (портретний режим для смартфону). Далі потрібно визначити структуру сітки для кожної заданої ширини.

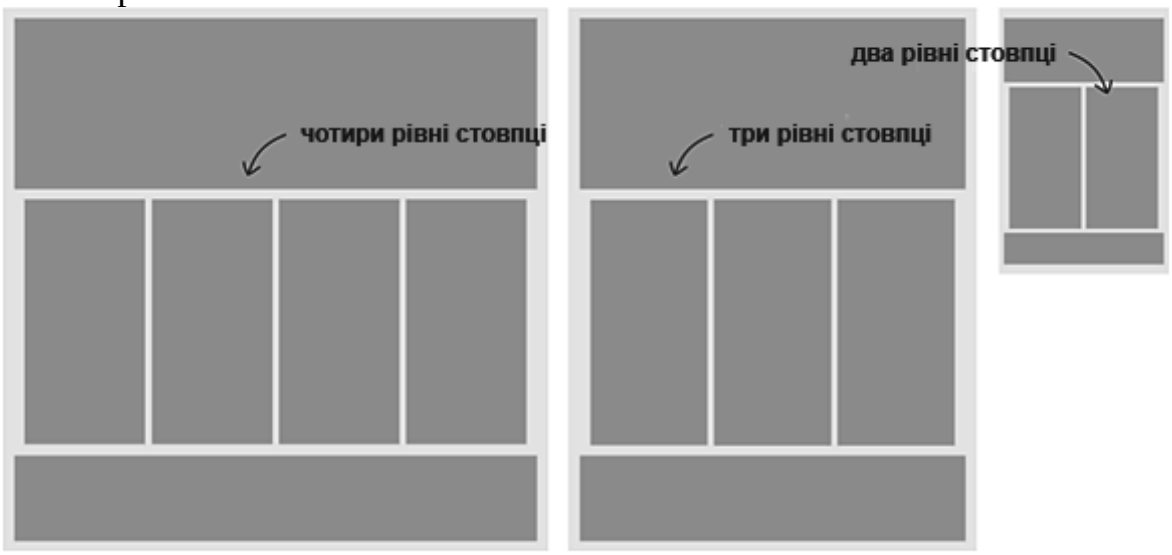


Рисунок 8.15 – Кількість стовпців залежно від пристрою

Дуже проста модульна сітка з однаковими по ширині колонками на кожному із прототипів допомагає нам планувати положення компонентів при зміні ширини екрану.

8.6.12. Створення головного макету

Відразу після створення прототипу необхідно подумати про колонки і про те, яким чином компоненти будуть адаптуватись до зміни ширини екрану. Що буде відбуватись, якщо у вас буде менше місця? Як наповнити чотири колонки? Що відбудеться, якщо на екрані буде лише три із них? При виконанні даного завдання повинен бути постійний зв’язок між дизайнером і розробником сайту, щоб проблеми та питання між зовнішнім виглядом та CSS могли бути оперативно вирішені.

8.6.13. Головна сторінка

На вашому сайті може бути інша сторінка, значення якої буде вищим, але особисто ми починали із головної сторінки. На малюнку можна побачити наш

закінчений прототип. Також зверніть увагу, що головна сторінка, розроблена для мобільного телефону, буде суттєво довшою відповідно до того, яка кількість контенту там розміщена.

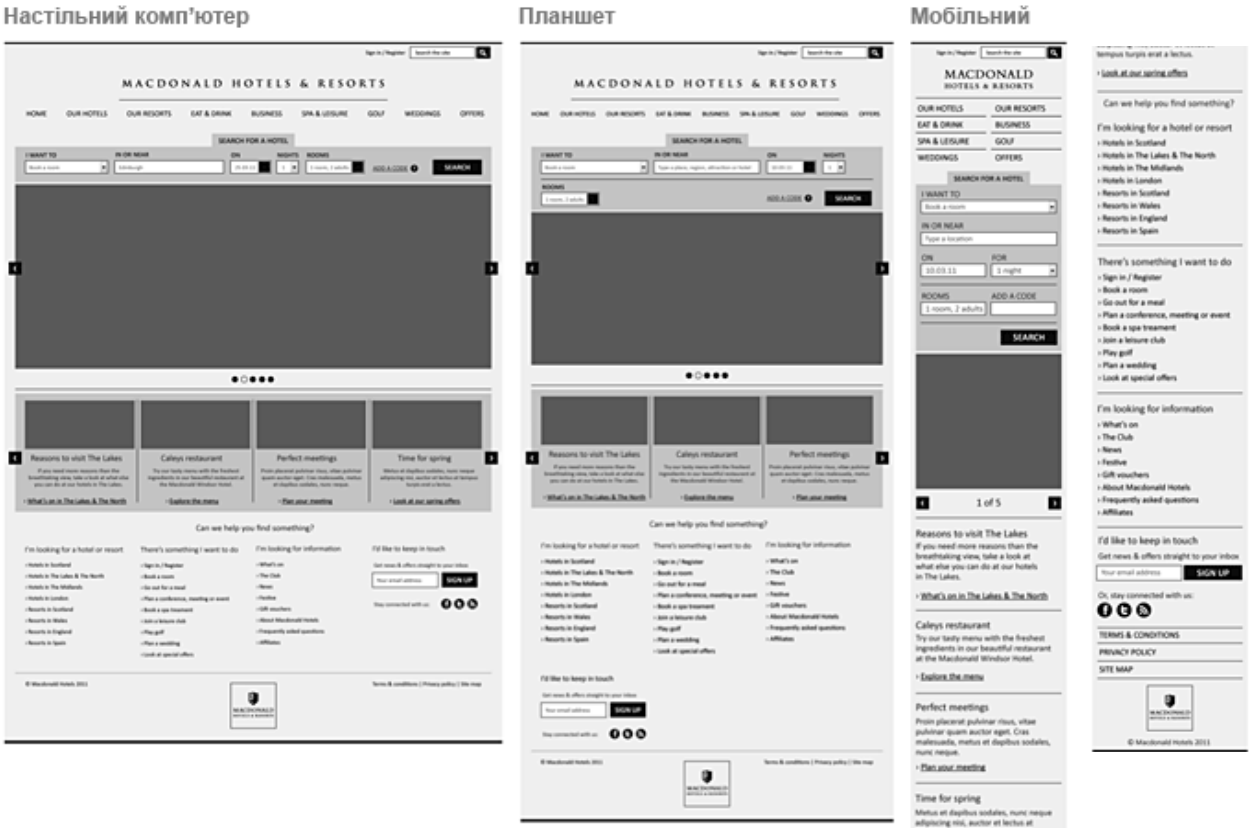


Рисунок 8.16 – Приклад проектування вайрфрейму перебудови макету для різних пристроїв

8.6.14. Головне меню

Ми створили просте горизонтальне меню зверху сторінки, ширина якого може змінюватись відповідно до розміру екрану. При зменшенні ширини вікна пункти меню наближаються одне до одного, а при необхідності навіть переносяться на наступний рядок. Це добре підходить для звичайних комп'ютерних моніторів, ноутбуків, а також для планшетів. Але розробляючи вигляд меню для мобільних телефонів ми зробили його на дві колонки, вмістивши, таким чином, на екран.

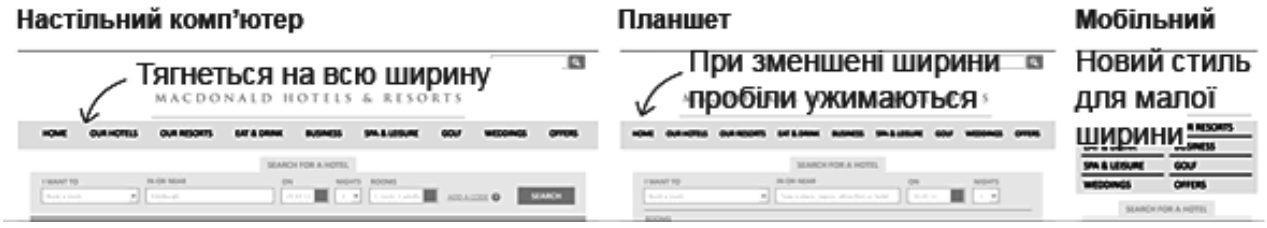


Рисунок 8.17 – Особливості проектування панелі меню для різних пристроїв

Інші основні компоненти розміщені справа і так само можуть переміщуватись разом із зменшенням ширини сторінки.

Коли ви розробляєте стиль навігації, важливо завжди пам'ятати, що розміри екранів у різних користувачів будуть змінюватись. Певні стилі, які

використовують вкладки, буде важко адаптувати під зменшену ширину екрану і вони будуть спотворені.

8.6.15. «Підвал» сторінки (Footer)

Продумати низ сторінки найпростіше — просто вирішіть, який контент вам потрібен і в якому порядку він повинен бути один під одним при зменшеній ширині екрану.

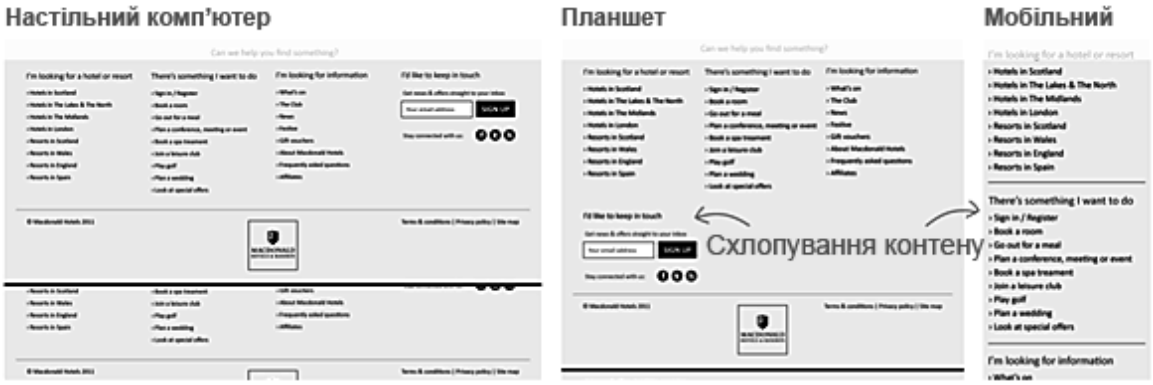


Рисунок 8.18 – Особливості проектування футеру для різних пристроїв

8.6.16. Інші компоненти

Наша проста модульна сітка дала нам можливість легко планувати положення інших блоків, тому на головній сторінці ми розмістили горизонтальний скролер із чотирма компонентами, які змінюють один інший при натисканні на них. Планшетна сітка також дозволяє нам зберегти цей блок, однак скролер складається вже із трьох компонентів. А користувач, який заходить на сайт з мобільного телефону, взагалі не може бачити цього скролеру, натомість він бачить кожний компонент окремо в вертикальному положенні.

Кожна складова дизайну вашого сайту може вимагати різних рішень, тому потрібно добре обдумати те, в якому вигляді ваш користувач бажає бачити сайт на різних пристроях. Наприклад, користувачі, які відвідують сайт з мобільного телефону, більш схильні переміщуватись по ньому зверху-вниз, ніж використовувати невеликі кнопки для навігації по сайту в різні сторони.



Рисунок 8.19.Особливості проектування компонентів для різних пристроїв

8.6.17. Тестуйте

Безумовно, необхідно протестувати ваш мобільний сайт перед запуском, але важливо також тестувати і після запуску. Основною причиною є те, що у вас дуже мало інформації (не рахуючи гіпотез) про те, як люди будуть

використовувати мобільний сайт. Тестування сайту після запуску допоможе вам знайти і усунути різні проблеми, і, відповідно, зробити ваш сайт ще краще і зручніше.



Рисунок 8.20 – Тестування вигляду інтерфейсу на мобільному

Відразу, як ви створили прототип вашого сайту, тестуйте його на відповідному пристрої. Це допоможе вам зрозуміти, наскільки зручно буде майбутньому користувачу переміщуватись сторінкою. Також це дасть змогу зрозуміти з самого початку, чи буде цей прототип вдало працювати. У випадку з нашим сайтом, це також допомогло нам зрозуміти, що працює добре, а що ні. Якщо ви подивитесь на малюнок, ви також зможете побачити нашу першу проблему:

Знаходячись на сайті, єдиним, що бачить користувач, не прокручуючи сторінку, є меню та панель пошуку. А це означає, що при спробі перейти на будь-яку іншу сторінку, користувачеві потрібно буде кожного разу прокручувати її донизу, щоб зрозуміти, чи вона нарешті завантажилась. Проблему було вирішено, коли ми згорнули меню та пошук готелю в панелі, які користувач може розгорнути при необхідності.

Тепер, коли прототип сайту створений і протестований, а проблеми, які виникали б у різних користувачів на різних пристроях, вирішені, саме час зробити так, щоб ваш сайт виглядав добре на будь-якій ширині екрану.

8.6.18. Зовнішній вигляд

Немає необхідності створювати абсолютно різні дизайни для комп’ютера, планшета і мобільного телефону. Головною метою є створити такі HTML-шаблони та стилі CSS, які дадуть можливість забезпечити майже однакову роботу та зовнішній вигляд вашого сайту на всіх пристроях, в яких

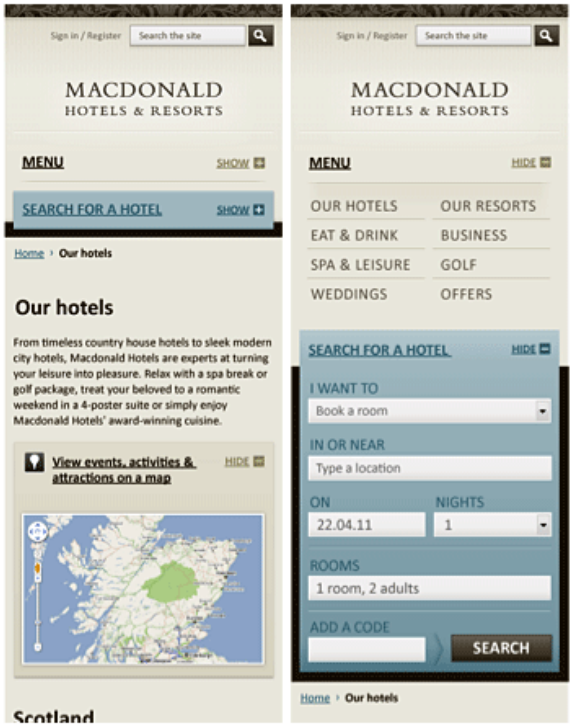


Рисунок 8.21 – Додатковий стиль для навігації

ви зацікавлені. Звичайно, у вас все ж будуть певні не кардинальні відмінності. Наприклад, для мобільної версії вам знадобиться дещо з того, що не потрібно версії для планшету.

8.6.19. Стиль сторінки

Рекомендуємо зберегти стиль версії для мобільного телефону якнайпростішим. Чудово, що CSS3 дає змогу вам не використовувати багато малюнків, щоб досягти цікавих стильних ефектів, але для їх завантаження все ж потрібен певний час.

8.6.20. Підбір шрифтів

Переконайтеся, що шрифт, який ви оберете, буде добре читабельним для користувача будь-якого пристрою.

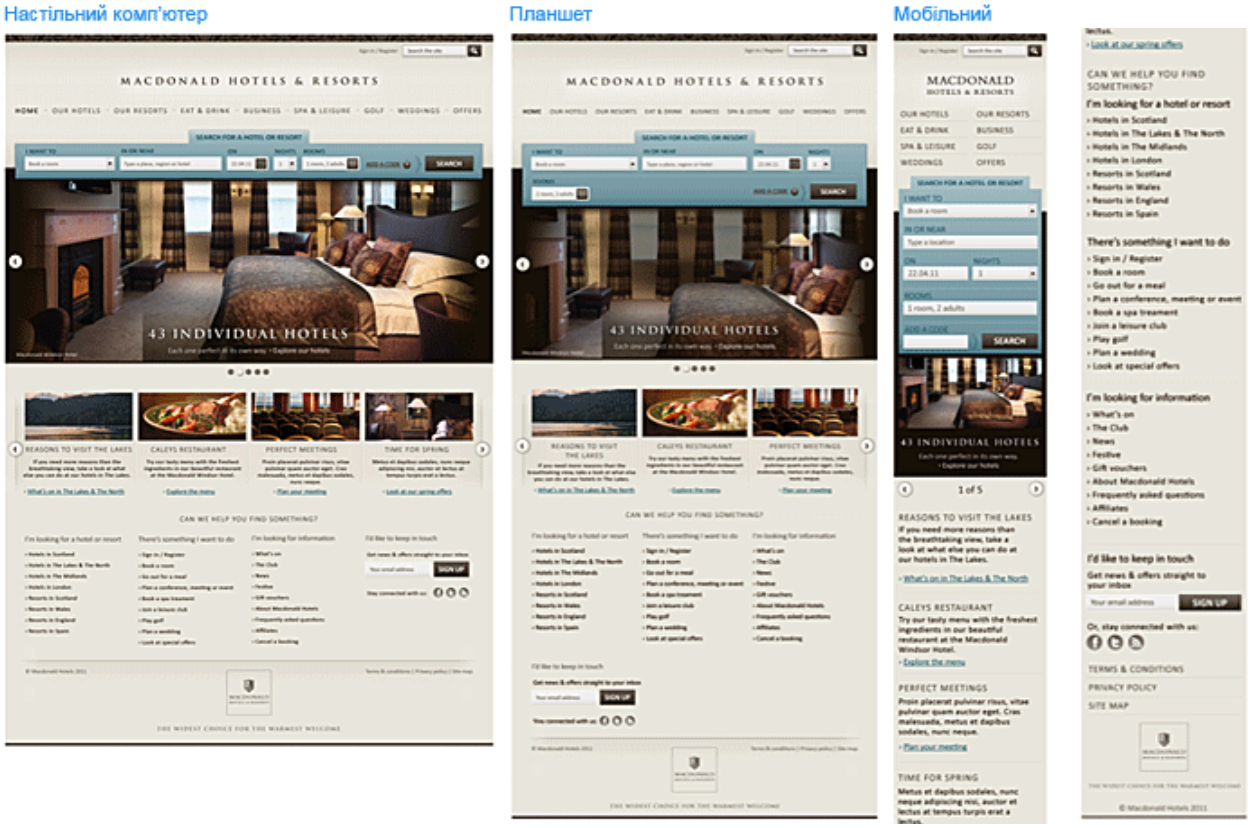


Рисунок 8.22 – Реалізація інтерфейсу під різні пристрої

Пам'ятайте також, що шрифт сайту для мобільного телефону повинен бути більшим за розміром для того, щоб користувач мав можливість вільно його прочитати. І останнє — важливим є збереження балансу між тим, що виглядає добре на екрані монітору та тим, що підходить для екрану мобільного телефону. Подивившись на малюнок, ви можете переконатись, що мобільна версія сайту принципово не відрізняється від основної версії.

8.6.21. Верстка сайту

Робота з HTML та CSS є окремим складним питанням, тому ми не будемо описувати тут всі деталі, а лише надамо декілька нюансів, які варто взяти до уваги:

1. **Зменшення розміру малюнків.** Сайту потрібно завантажувати малюнки в повному розмірі навіть тоді, коли CSS зменшує їх, тому намагайтесь зменшити розміри ваших малюнків настільки, наскільки це максимально

можливо зробити. Ви також можете використовувати різноманітні JavaScript-прийоми, які призначені згладити роботу сайту. В нашому випадку ми завантажили малюнки невеликого розміру, але використали JavaScript для збільшення їх у випадку, коли це необхідно.

2. **Використовуйте сучасний CSS.** Важливо переконати клієнта у використанні сучасних CSS-стилів, надавши їм (стилям) можливість спрощуватись відповідно до можливостей браузера. Це допоможе вам зменшити час завантаження сайту.

Постійна комунікація між усіма членами команди — обов’язкова. Робота над проектом буде проходити значно продуктивніше, якщо між дизайнером та розробником буде триматись постійний зв’язок. Це дозволить вчасно і результативно вирішувати питання, що виникають.

Контрольні питання

1. Перелічіть базові підходи до проектування адаптивного та гнучкого дизайну
2. Визначити цілі користувачів мобільних пристроїв
3. Порівняти Responsive Design та Adaptive Design
4. Перелічити типові помилки при проектуванні інтерфейсу для мобільних пристроїв. Модульна структура адаптивності та гумовості

Теми для самостійного опрацювання

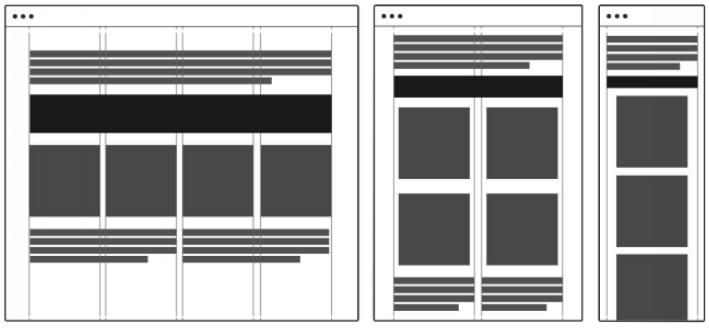
1. Переваги та недоліки використання додатку, адаптивного сайт або мобільної версії
2. Принцип Mobile first. Адаптивність «малою кров’ю»

Тестові завдання

1. Які типи адаптивних макетів НЕ існують

a) Пластичний інтерфейс
b) Перенесення блоків
c) Перемикання макетів
d) Адаптивність «малою кров'ю»

2. Який тип макету зображено?



a) адаптивний макет з гнучкою сіткою
b) адаптивний макет з перенесенням блоків
c) гумовий макет
d) макет з фіксованою шириною сітки

3. Фіксована ширина проти «гнучкої»

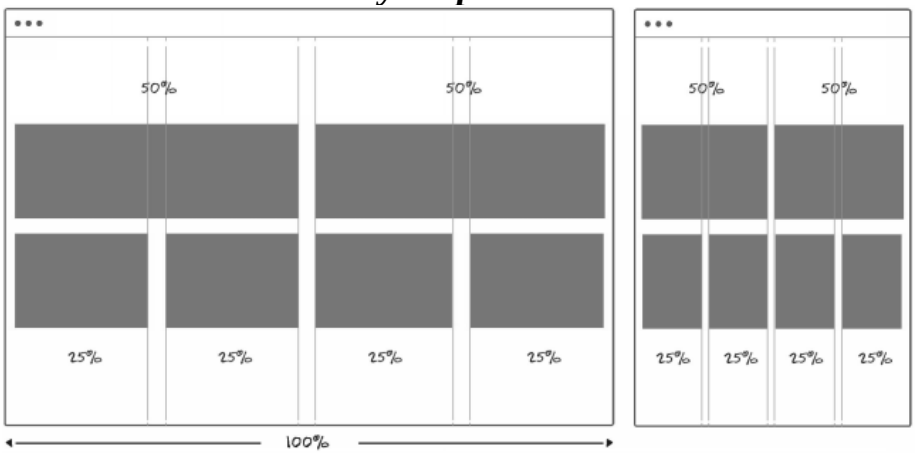
a) блоки мають визначений розмір у пікселях
b) блоки мають фіксований розмір у дюймах
c) блоки задаються співвідношенням у відсотках
d) вірна відповідь відсутня

4. Adaptive Design на відміну від Responsive Design

a) ґрунтується на проектування з гнучкою сіткою макета, що дозволяють одному макету працювати на різних девайсах;

b) ґрунтується на проектування сайту, що базується на кількох макетах фіксованої ширини.

5. Який тип макету зображено?



a) адаптивний макет з гнучкою сіткою
b) адаптивний макет з перенесенням блоків
c) гумовий макет
d) макет з фіксованою шириною сітки

6. «Гнучка» ширина проти фіксованої

a) блоки мають визначений розмір у пікселях
b) блоки мають фіксований розмір у дюймах
c) блоки задаються співвідношенням у відсотках
d) вірна відповідь відсутня

7. Responsive Design на відміну від Adaptive Design

a) ґрунтується на проектування з гнучкою сіткою макета, що дозволяють одному макету працювати на різних девайсах;
b) ґрунтується на проектування сайту, що базується на кількох макетах фіксованої ширини

8. Назвіть, що не входить до основних підходів створення адаптивного дизайну

a) мобільна версія сайту
b) мобільний додаток
c) "гумовий" дизайн
d) «Mobile first

9. Принцип, коли першочергово проектується інтерфейс для мобільного, потім для планшету і на завершення — десктопна версія

a) еволюційний метод
b) Mobile First
c) гнучкий макет
d) device-to-desktop

10. Які існують орієнтації екрану мобільних девайсів?

a) вертикальна та горизонтальна орієнтація
b) широка та довга орієнтація
c) ландшафтна та портретна орієнтація
d) нічого з переліченого

Практичні завдання

Завдання 8.1. Розробити вайрфрейми мобільної версії для планшету та смартфону (портретна+ландшафтна орієнтація);

Завдання 8.2. Розробити карту навігації мобільною версією;

Завдання 8.3. Опрацювати прототип мобільної версії згідно до вайрфреймів прототипи основних екранів, модальних вікон та панелей для мобільної версії.

9. АНІМАЦІЯ ВЗАЄМОДІЇ UI. ЕТАП ФІНАЛІЗАЦІЇ ПРОЕКТНОЇ РОЗРОБКИ.

План:

9.1. Анімація і послідовності подій
9.1.1. Анімація при завантаженні даних
9.1.2. Анімація процесів
9.1.3. Анімація покрокових операцій
9.1.4. Анімація і каркасне відображення веб-сторінок
9.2. Візуальний зворотний зв'язок
9.2.1. Анімована реакція на дії користувача
9.2.2. Анімація елементів керування для десктопних і мобільних сайтів 9.2.3. Привертання уваги за допомогою анімації
9.2.4. Елементи для спонукання здійснити дію (Call to Action)
9.2.5. Принцип AIDA
9.3. Переходи і зміна станів
9.3.1. Плавна зміна станів
9.3.2. Довгий скролл
9.4. Типи анімацій
9.4.1. GIF-анімація
9.4.2. Флеш-анімація
9.4.3. Сінемаграфія – живі фотографії
9.5. Правила та принципи для анімації в UX/UI-дизайні
9.5.1. Крива анімації
9.5.2. Вимоги до анімації
9.5.3. Тривалість та швидкість анімації
9.5.4. Пом’якшення (Easing)
9.6. Лінійний рух
9.6.1. Крива прискорення (Ease-in)
9.6.2. Крива сповільнення (Ease-out)
9.6.3. Стандартна крива (Ease-in-out)
9.7. Хореографія анімації інтерфейсів
9.7.1. Взаємодія рівності
9.7.2. Взаємодія підкорення
9.8. Суть анімації мікровзаємодії для макрорезультату
9.8.2. Навіщо використовувати в дизайні мікро-анімації?
9.8.3. Анімована мікровзаємодія кнопки
9.8.4. Анімація «scroll-to reload»
9.8.5. Анімація «State-of-process»
9.10. Основні правила анімації

Дизайн – це більше ніж візуальне представлення даних, це управління взаємодією користувача з веб-сторінкою. Як результат, мультимедійні об’єкти відіграють надзвичайно важливу роль в справі передачі інформації.

Проектуючи сторінки, необхідно з самого початку пам'ятати про інтерактивну природу веб-простору і сприймати її прояви як природну частину дизайну.

HTML5 є платформою для реалізації не просто сторінок, а веб-додатків, що містять анімацію, графіку, відео і аудіо. Все це робить HTML однією з найпопулярніших мов, яка динамічно розвивається і вдосконалюється.

Взаємодія користувачів з сучасними веб-сайтами серйозно зав'язано на анімації та мікроанімаії.

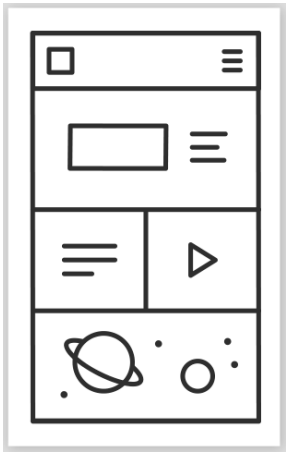




Рисунок 9.1 – Демонстрація анімації вайрфрейму

9.1. Анімація і послідовності подій

Якісна і доречна анімація пожвавлює сторінку і сприяє кращому сприйняттю інформації. Вона здатна повідомляти про певні стани сторінки та привертати увагу. Анімація допомагає користувачеві побачити результат його дій і може впливати на його поведінку.

9.1.1. Анімація при завантаженні даних

Такий прийом створює відчуття того, що дія виконується швидше, ніж насправді. Анімована картинка свідчить, що прогрес завантаження відбувається і відвідувач легше сприймає цей час. Чим простіше анімація завантаження – тим краще. При цьому слід дуже обережно ставитися до будь-яких додаткових ефектів, на зразок звуків. Зазвичай вони просто не потрібні.

9.1.2. Анімація процесів

Анімація може показати виконання певної лінійної послідовності дій, що не залежить від користувача. Типовим прикладом є прогрес завантаження.

9.1.3. Анімація покрокових операцій

Анімацію можна використовувати для процесів, які передбачають виконання користувачем покрокових операцій, щоб відображати послідовність виконаних дій.

9.1.4. Анімація і каркасне відображення веб-сторінок

Варіант каркасного відображення веб-сторінок передбачає наявність простору, яке поступово заповнюється завантаженою інформацією.

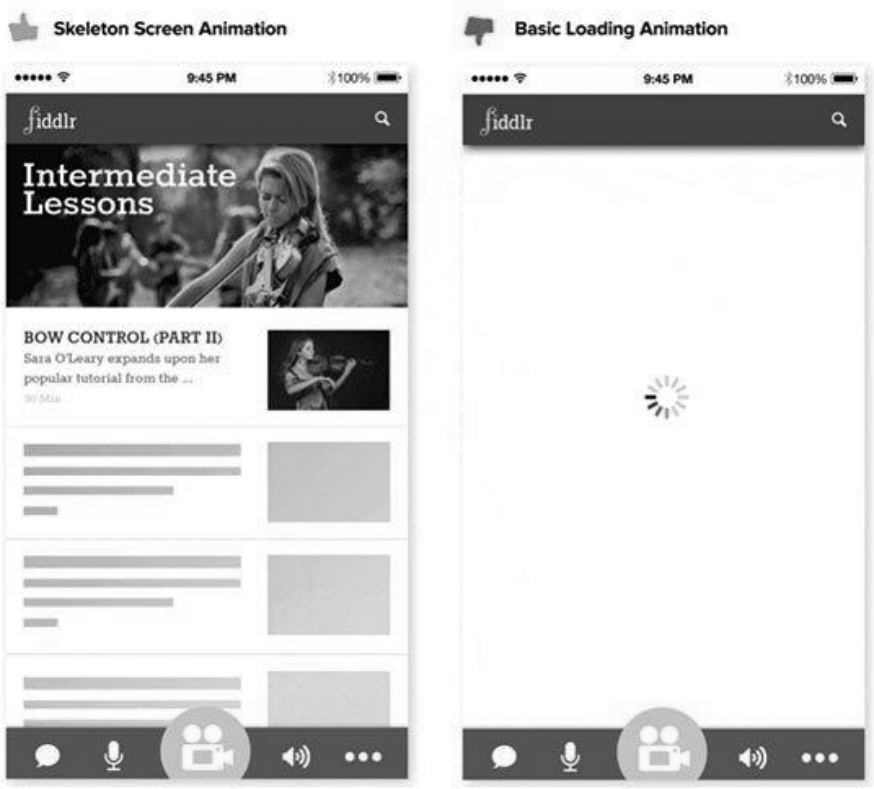


Рисунок 9.2 – Каркасне відображення

Поступове, не розірване періодами «порожнечі», наповнення сторінки, дає відчуття швидкості виконання дій. Цю техніку можна використовувати на практично будь-якому сайті разом з ненав'язливою анімацією процесу завантаження, що забезпечує утримання уваги користувача.

Каркасні екрани завантажують елементи користувацького інтерфейсу ще до того, як буде повністю відображено вміст сторінки.

9.2. Візуальний зворотний зв'язок

9.2.1. Анімована реакція на дії користувача

Хороший дизайн взаємодій з користувачем передбачає зворотний зв'язок, що повідомляє про результати взаємодії, робить його видимим і зрозумілим. Якщо відвідувач сайту не знає, з якими саме елементами сторінок можна працювати, чого від них можна очікувати, він буде відчувати себе розгубленим. Потрібно продумано спроектувати варіанти взаємодії, зробивши їх чіткими і зрозумілими.

9.2.2. Анімація елементів керування для десктопних і мобільних сайтів

Один з найпоширеніших прикладів візуального зворотного зв'язку – анімація при наведенні курсору на інтерактивний елемент.

Якщо користувач не впевнений в призначенні елемента управління, він намагається навести на нього курсор миші. Анімація елемента при наведенні курсору, на інтуїтивному рівні, повідомить користувача про те, що з об'єктом можна взаємодіяти.

На мобільних сайтах, кнопки та інші інтерактивні елементи повинні бути оснащені візуальними підказками. Підказки повинні вказувати на те, як

взаємодіяти до того, як користувач торкнеться елемента. Після торкання потрібно надати візуальний зворотний зв'язок, щоб засвідчити реакцію системи.

9.2.3. Привертання уваги за допомогою анімації

Зміна стану елементів на сторінці скеровує увагу користувача і підкріплює виконувані ним дії. Наприклад, при введенні даних в форму, поле підсвічується і користувач розуміє, що потрібно вводити текст.

9.2.4. Елементи для спонукання здійснити дію (Call to Action)

Спонукання до дії (Call to Action) відноситься до інтерактивних елементів сайту: кнопки, банери та інше. Дані елементи оформлюються таким чином, щоб користувач захотів неодмінно на них натиснути. Наприклад, це може бути кнопка із закликом до дії (Натисніть, Дізнайтеся тощо), яскравий банер з привабливою пропозицією чи яскравою картинкою.

9.2.5. Принцип AIDA

Дане поняття добре вписується в принцип **AIDA** (Attraction Interest Desire Action), який часто застосовується при дизайні головних сторінок, сторінок акцій, де необхідно підштовхнути користувача до тієї або іншої дії: підписка, покупка тощо (рис.6.9).

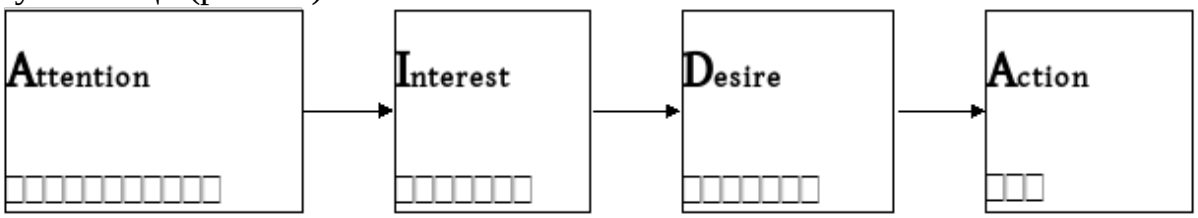


Рисунок 9.3 – Ланцюжок принципу AIDA

Таким чином стає зрозумілим принцип побудови дизайну, що спирається на дане поняття: наприклад, яскрава картинка чи банер повинні привернути увагу користувача, супутній підпис в тексті повинен викликати в нього інтерес і бажання, а завершальним акордом повинна стати, наприклад, кнопка із закликом до дії.

Але цей принцип не працює сам без деяких інших, наприклад, схеми перегляду сторінки чи візуальні напрямків.

9.3. Переходи і зміна станів

9.3.1. Плавна зміна станів

Анімація переходів надає можливість плавно змінити колір фону, прозорість блоку, затемнення. Без анімації подібні зміни будуть з різким переходом, який відчувається як неприродний «стрибок». За допомогою анімації можна забезпечити плавні переходи між екранами. Добре спроектовані переходи дають користувачам чітке розуміння того, на чому слід зосередитися.

9.3.2. Довгий скролл

Раніше дизайнери прагнули заповнити верхню область сторінки цінною інформацією. Тепер же «правило верхньої половини» втратило абсолютну істинність. Дизайнери застосовують красиві плавні ефекти для скролінгу сторінки як вниз, так і вгору. Це робить прокрутку простою і зручною.

Анімації оживляють процес скролінгу і приносять задоволення користувачеві.

9.4. Типи анімацій

Анімація є важливою складовою дизайну веб-сайту і у веб-дизайнера є широкі можливості вибору багатьох технологій, які допоможуть в досягненні конкретних цілей, що ставляться перед сайтом. Тип анімації і програмного забезпечення, необхідного для її створення визначається в залежності від конкретного призначення.

9.4.1. GIF-анімація

Анімація з використанням формату GIF (Graphic Interchange Format) є простою в реалізації і відтворенні на сторінках. Анімовані GIF-зображення працюють як традиційна покадрова анімація, яку застосовують у мультфільмах. Анімоване зображення містить певне число кадрів, де кожен кадр є частиною анімації, також задаються параметри, які дозволяють регулювати швидкість і тривалість анімації. Програма, за допомогою якої створюється GIF-анімація (наприклад, PhotoShop) зберігає кадри та відмічені опції в GIF-файлі, а браузер інтерпретує задані параметри і відтворює анімацію, створюючи зоровий образ руху.

9.4.2. Флеш-анімація

Adobe Flash – це є мультимедійна платформа, яку використовують для створення векторної анімації та інтерактивних застосувань (зокрема, ігор), а також для інтеграції відеороликів у веб-сторінки.

Flash дозволяє створювати повноцінну анімацію з вбудованим звуком і інтерактивністю при відносно невеликих розмірах файлу. Невеликі розміри файлів Flash продиктовані використанням в основному векторних зображень, які описані математичними формулами. Векторні зображення, які збережені у Flash-анімації, позначаються як SVG-зображення. Розміри SVG-зображень можна змінювати без втрати якості зображення і вони завантажуються швидше, ніж традиційні JPEG- або GIF-зображення. На сьогоднішній день технологія Flash є застарілою.

9.4.3. Сінемаграфія – живі фотографії

Сінемаграфія – це нова концепція фотографії, яка об'єднує відео і фото. Сінемаграфічний ефект утворюється з відео, в якому певні частини кадру залишають статичними, а в інших простежується рух. Природно, що такі роботи можуть бути затребувані лише в веб-виданнях і самої вдячної нішею їх використання звичайно ж є реклама. Сінемаграфія відтісняє flash банери, оскільки вихідним форматом такого живого фото є GIF, який не потребує додаткових плагінів і ресурсів. Зараз розроблено досить багато програм під iPhone і Android, що дозволяють робити з відео фото з сінемаграфією. Принцип їх роботи однаковий – з обраного невеликого шматочка відео з бажано зацикленим рухом певного об'єкту виокремлюється та частина, яка буде рухатися, а решта залишаються незмінними як в першому кадрі.

Сінемаграфія – це досить складне мистецтво. По-перше, потрібна концепція і та особливість, яка буде привертати увагу глядача. По-друге, відео

для сінемаграфії має бути знято за певними правилами. І по-третє, сама пост-обробка, яка вимагає кропіткої роботи та уваги. Як правило, для ефекту сінемаграфії, автори використовують такий рух як: погойдування на вітрі, волосся що розвиваються, потік води, рух тіні при нерухомості самого об'єкта і багато іншого.

9.5. Правила та принципи для анімації в UX/UI-дизайні

9.5.1. Крива анімації

Будь-проектувальник або розробник, який створює анімацію в команді, найімовірніше, стикався з подібним сценарієм.

Напевно ви задавалися питанням: а як краще передати іншому фахівцю вимоги для реалізації Моушн-дизайну? Відмінне рішення для спільної роботи над проектуванням і створенням анімації – це використовувати криву анімації.

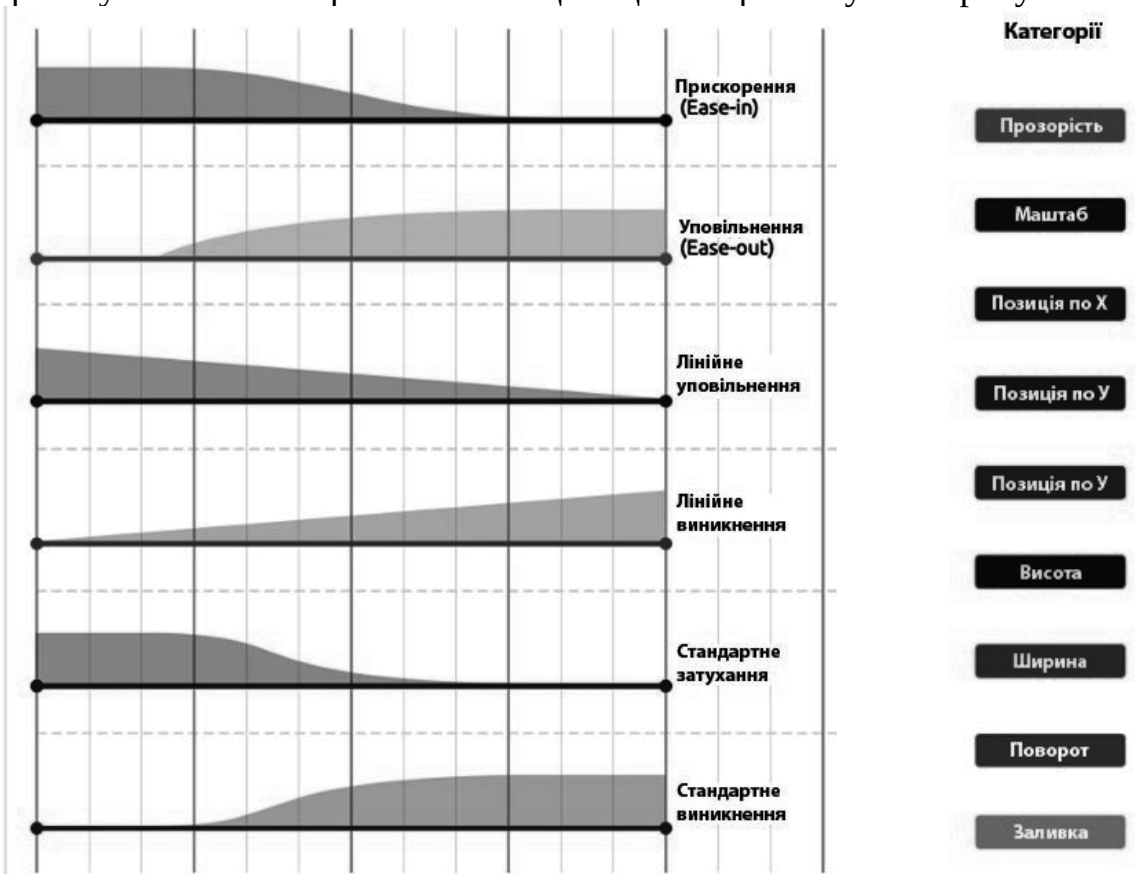


Рисунок 9.4 – Криві анімації

Крива анімації або Моушн-граф – це графічне представлення тимчасової шкали анімації. Крива анімації – як умовні знаки на карті. Це наочна ілюстрація, яка серйозно допомагає зрозуміти вимоги. Крива відповідає на багато питань по анімації:

- Який елемент анімується?
- Яка властивість елемента (прозорість, позиція, кут тощо) змінюється?
- Коли починається і закінчується анімація?
- Який вигляд кривої застосовується?

Навіть швидкий погляд на криву анімації дає чимало інформації. Це дозволяє не витрачати час на вивчення тимчасової шкали тих інструментів, які ви використовували, – адже вона може виявитися ще більш заплутаною.

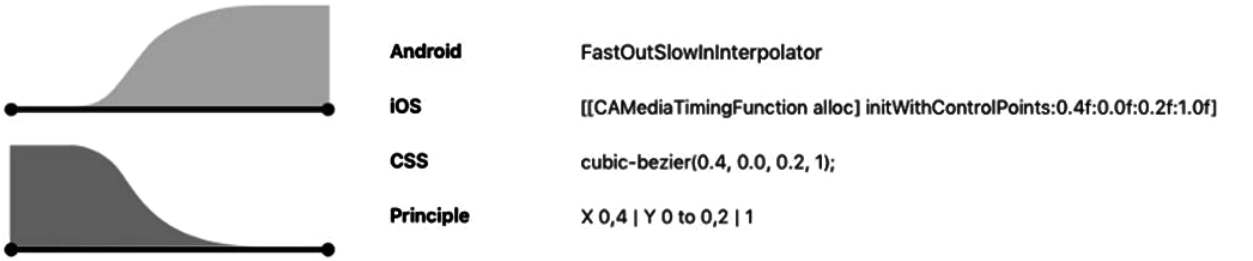


Рисунок 9.5 – Стандартні криві анімації

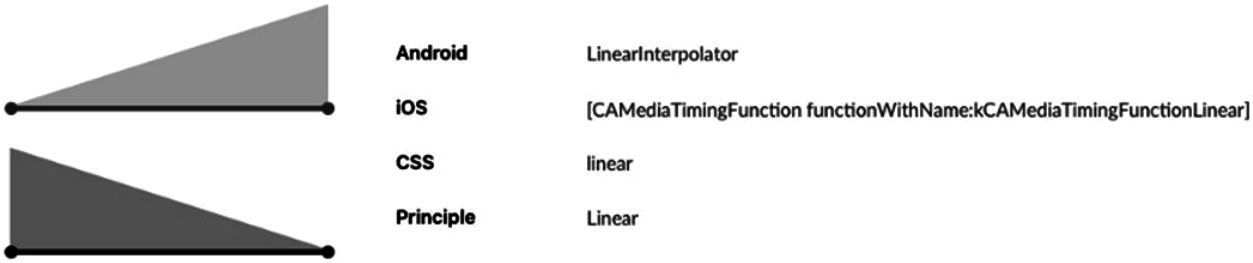


Рисунок 9.6 – Лінійні криві анімації

Основний інструмент опису Моушн-дизайну – це відео або GIF-анімація. При передачі завдання розробникам, так найпростіше продемонструвати бажаний результат.

9.5.2. Вимоги до анімації

Крива – це деталізація анімації, подробиці руху елементів. Така наочність допомагає розробникам зрозуміти особливості анімації і не замислюватися про особливості інструментів проектування. Крива визначає, як і які елементи будуть змінюватися, – і жодна деталь не буде упущена.

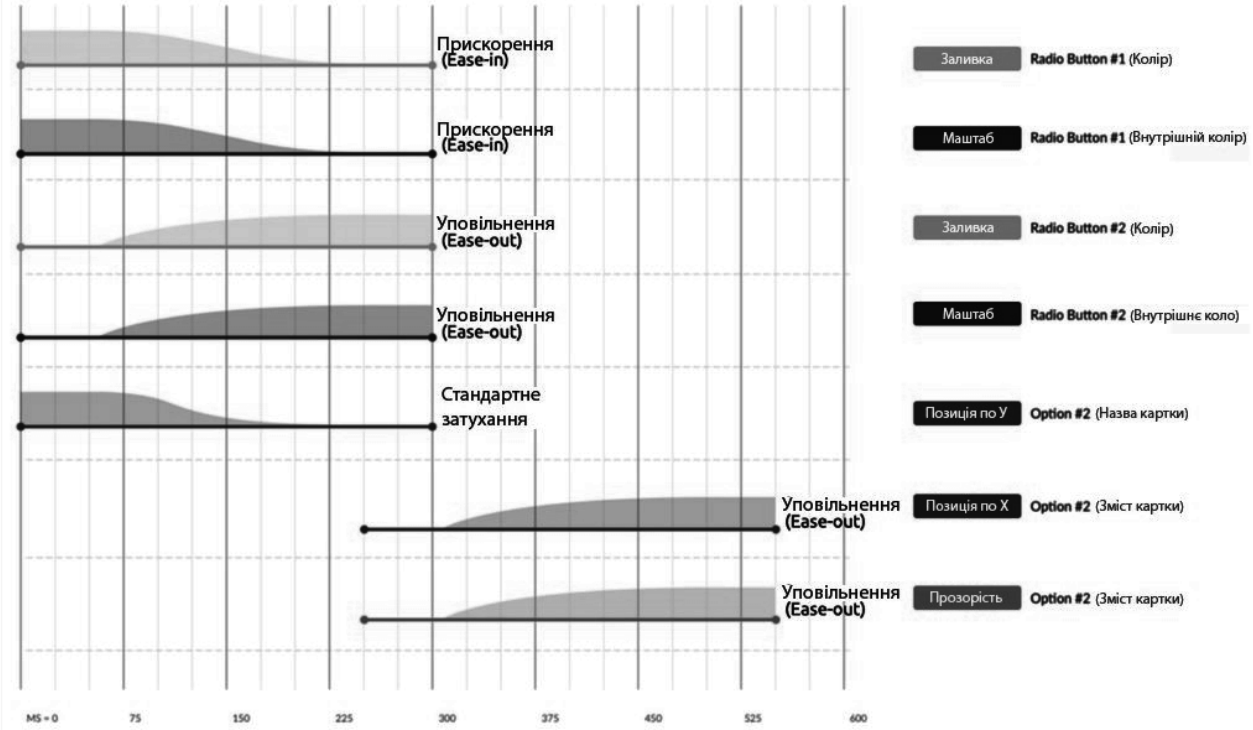


Рисунок 9.7 – Стандартні криві анімації

У цьому конкретному випадку крива дозволяє визначити кілька принципових речей:

Анімація складається з двох окремих частин: перша починається з позначки 0 мс, а друга з 250 мс.

Використовуються три види кривих для анімації об'єктів: «прискорення» (Ease-in), «уповільнення» (Ease-out) та «стандартне загасання».

Анімація зачіпає чотири об'єкти: дві радіокнопки (Radio Button # 1 і # 2), назва і зміст розкривається картки (Option # 2).

Таким же чином порушені чотири властивості об'єктів: заливка, прозорість, позиція по Y і масштаб.

Якщо припустити, що види кривих задані заздалегідь і стандартні для проекту, то організатори повинні тільки встановити початок і тривалість анімації, застосувати вигляд кривої і вибрати мінливий властивість об'єкта. Крива буде непереливки точні значення властивостей об'єктів, наприклад, зміна прозорості від 0% до 100% або зміщення позиції по Y з 450 до 550. Ці значення як і раніше повинні бути документально оформлені дизайнером і вивчені розробником. Але використання кривих анімації куди наочніше, ніж фраза «повторить точь-в-точь».

9.5.3. Тривалість та швидкість анімації

Коли елементи змінюють свою форму або розташування, тривалість анімації повинна бути достатньою, щоб користувач помітив зміни. Але водночас не надто довгою, щоб не змушувати його чекати.

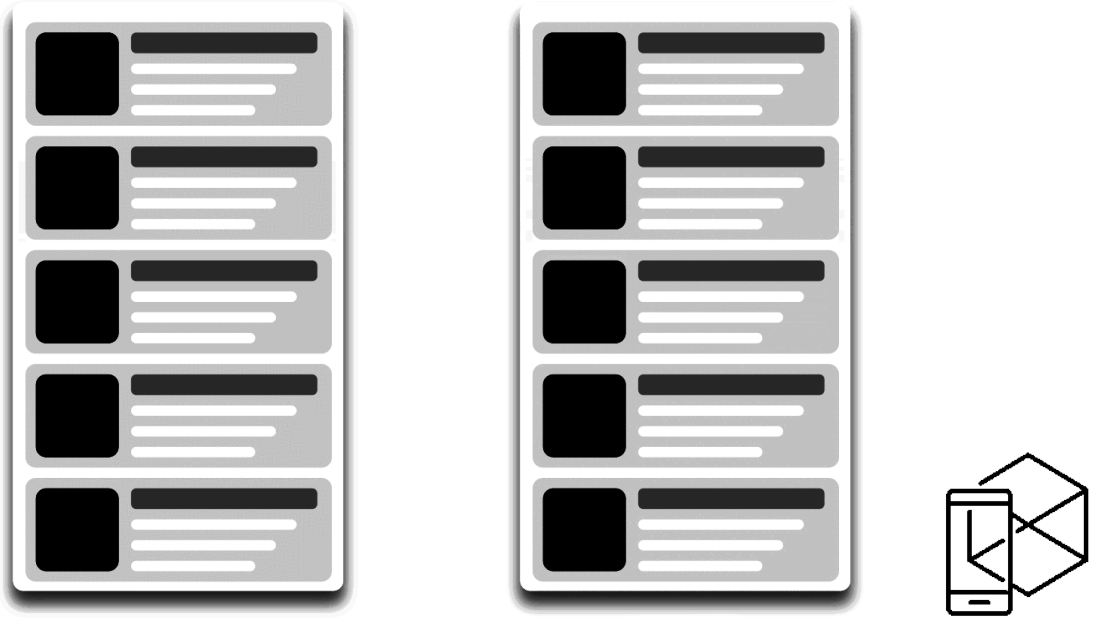


Рисунок 9.8 – Тривалість та швидкість анімації

Згідно з Material Design Guideline, оптимальна тривалість анімації в інтерфейсах може коливатися від 200 мс до 500 мс. Ці цифри базуються на будові нашого мозку. Анімацію коротше за 100 мс важко помітити, а рух довше однієї секунди наш мозок вважатиме затримкою.

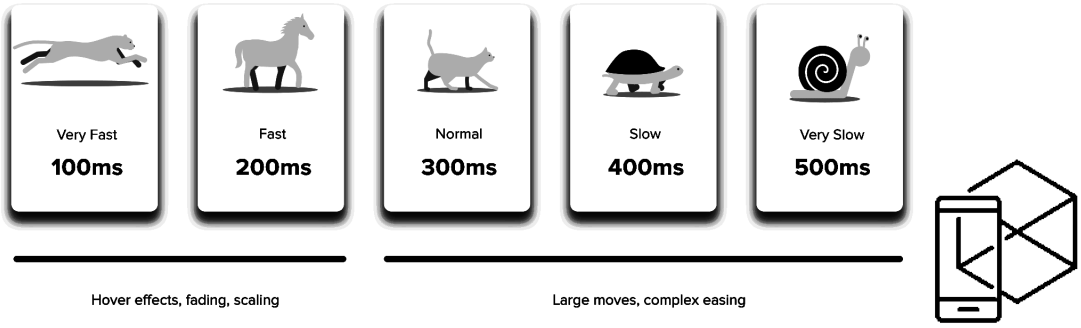


Рисунок 9.9 – Тривалість та швидкість анімації

Гайди з Material Design радять обирати для мобільних пристроїв такі анімації, що мають тривалість 200-300 мс, а на планшетах — 400-450 мс. Пояснюється це легко: на великих екранах для переміщення потрібно більше часу. На розумних годинниках краще обрати тривалість у 150-200 мс.



Рисунок 9.10 – Тривалість та швидкість анімації для різних пристроїв

У веб-анімації працюють інші правила. Ми звикли, що сайти у браузері завантажуються миттєво, тому і очікуємо, що всі зміни також будуть миттєвими. Тому тривалість переміщення елементів у веб-інтерфейсах повинна бути меншою — 150-200 мс. У іншому випадку користувач буде вважати, що є якісь проблеми з комп’ютером чи інтернетом. Але правила існують, щоб їх порушувати. Якщо ви використовуєте анімацію творчо або намагаєтесь привернути увагу —тривалість може бути довшою.



Рисунок 9.11 – Тривалість та швидкість анімації

Незалежно від платформи, тривалість анімації залежить не лише від пройденої відстані, але і від розміру об’єкту. Маленькі елементи повинні рухатися швидше, тому анімація з великими та складними елементами повинна бути повільнішою. Між двома елементами однакового розміру першим зупиняється той, що пройшов менший шлях.

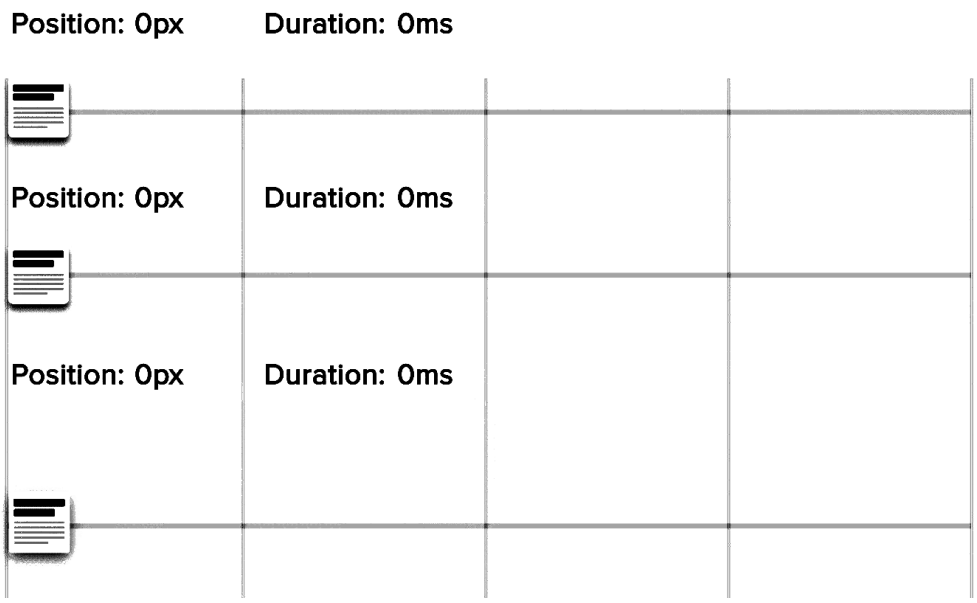


Рисунок 9.12 – Тривалість та швидкість анімації

У разі зіткнення об’єктів, енергія від цього повинна бути розповсюджена між ними згідно з законами фізики. Тому краще знехтувати ефектом пружності. Використовуйте його лише в окремих випадках, коли у цьому є сенс.



Рисунок 9.13 – Тривалість та швидкість анімації

Переміщення об’єктів повинно бути чітким, тому не варто використовувати розмиття протягом руху (motion blur). Цей ефект важко відтворити навіть на сучасних смартфонах, тому в анімації інтерфейсів він не використовується взагалі.

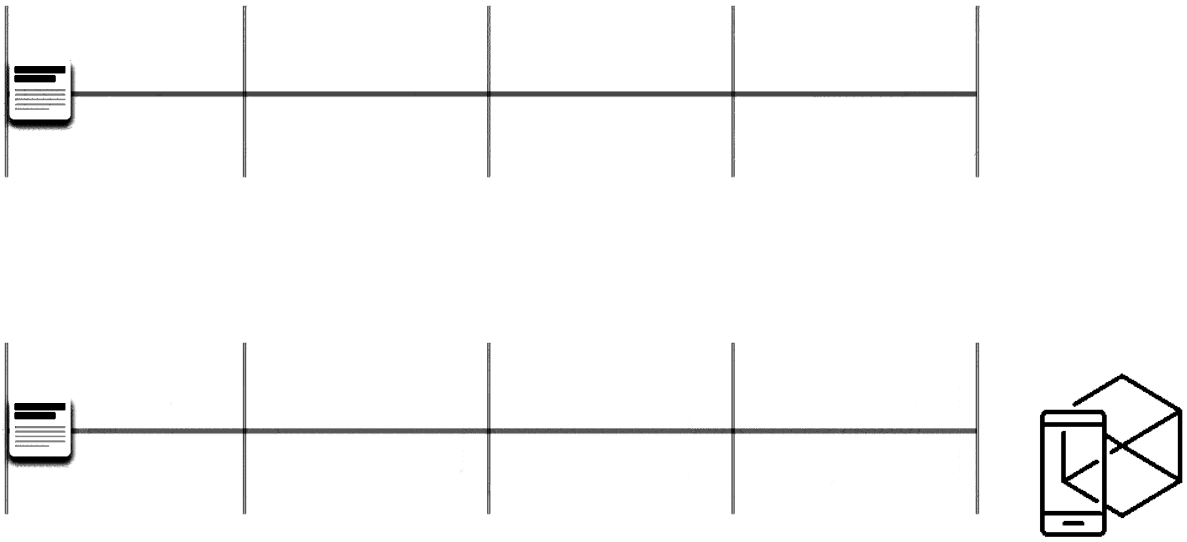


Рисунок 9.14 – Тривалість та швидкість анімації

Елементи списку повинні відтворюватися з невеликою затримкою: наступний йтиме через 20-25 мс після попереднього. Повільніша поява може викликати роздратування.

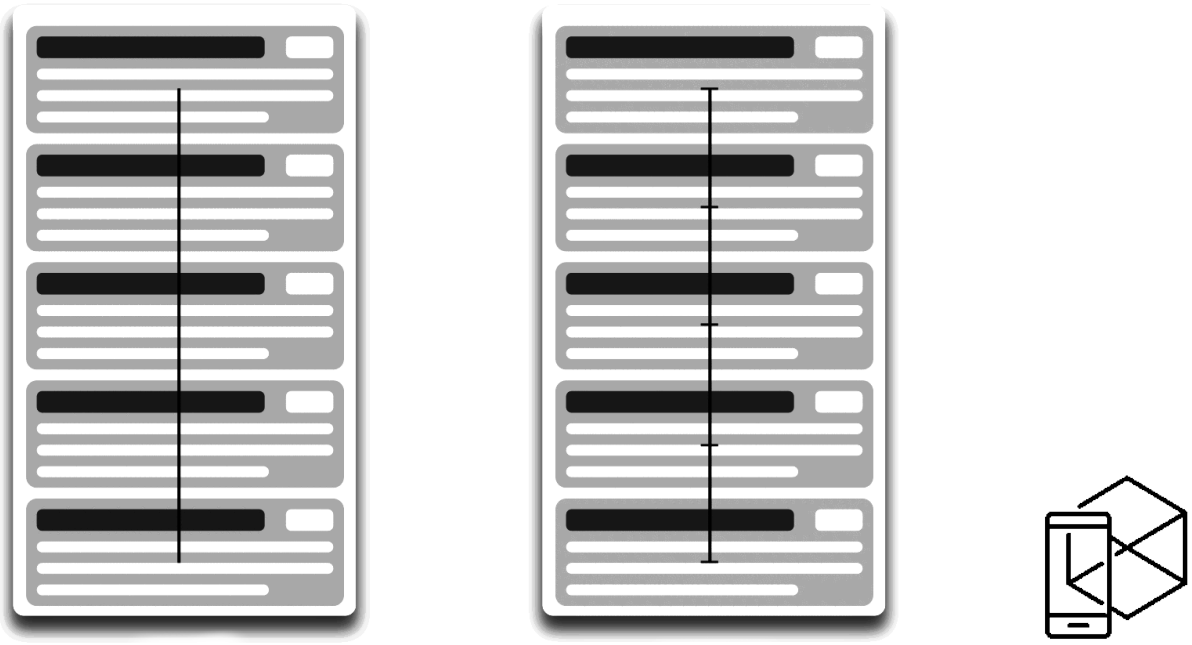


Рисунок 9.15 – Тривалість та швидкість анімації

9.5.4. Пом’якшення (Easing)

Пом’якшення робить рух об’єкту більш природним. Це один із основних принципів анімації. Він ретельно описаний у книзі «The Illusion of Life: Disney Animation» авторства двох діснеївських аніматорів Оллі Джонстона та Франка Томаса. Щоб анімація не виглядала механічною та штучною, об’єкт має рухатись із певним прискоренням або затримкою. Так само, як об’єкти фізичного світу.

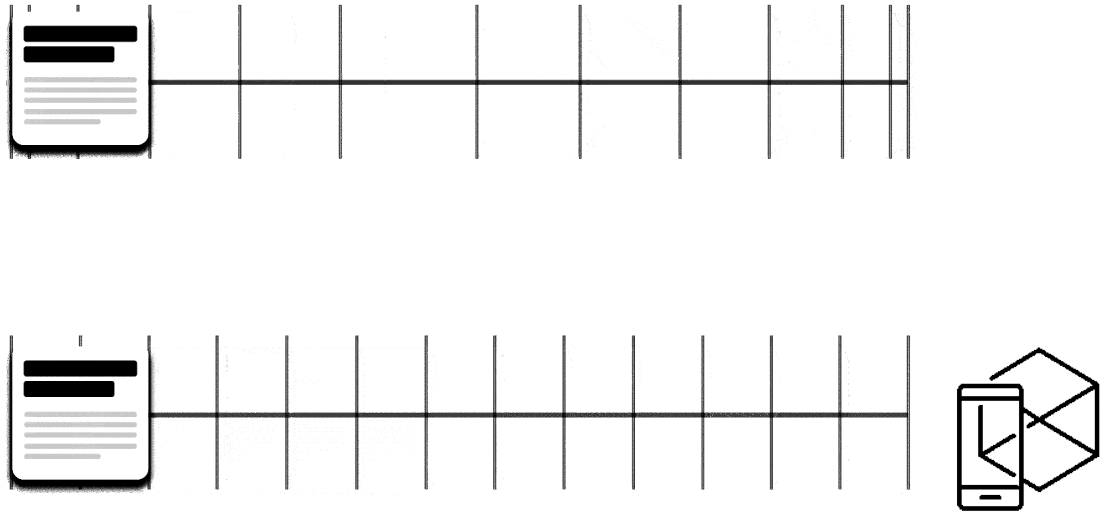


Рисунок 9.16 – Пом’якшення (Easing)

9.6. Лінійний рух

Об’єкти, що не зазнають впливу якоїсь фізичної сили, рухаються лінійно. Інакше кажучи, зі сталою швидкістю. І саме тому вони виглядають дуже ненатурально для людського ока.

Всі програми для анімації використовують так звані «криві анімації». Аби їх читати і розуміти що вони означають, необхідно знати, що крива показує, як позиція об’єкта (вісь Y) змінюється протягом одного інтервалу часу (вісь X). Тут рух лінійний, тож об’єкт проходить відстань, що дорівнює проміжку часу.

Лінійний рух можна використовувати лише якщо змінюється колір чи прозорість об’єкта. Загалом, такий рух використовується тоді, коли об’єкт не змінює позицію.

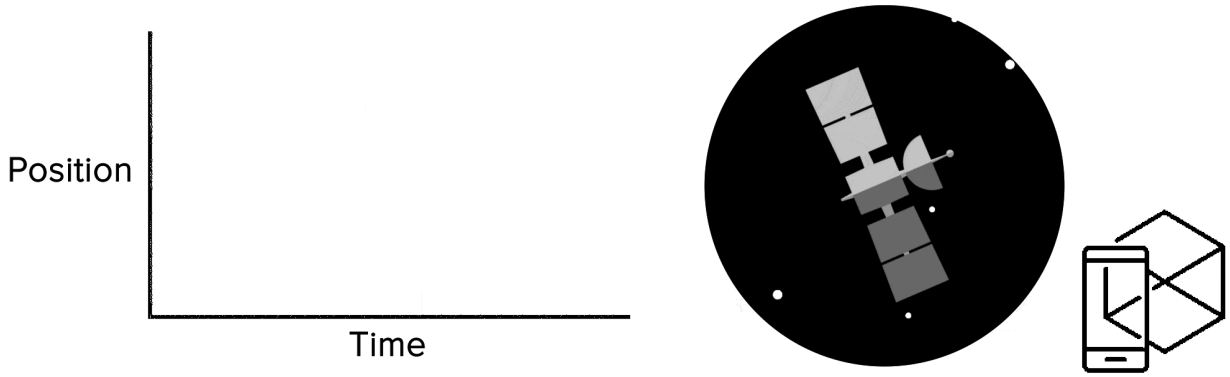


Рисунок 9.17 – Лінійний рух

9.6.1. Крива прискорення (Ease-in)

На цій кривій видно, як повільно спочатку позиція об’єкту змінюється. Згодом вона поступово прискорюється.

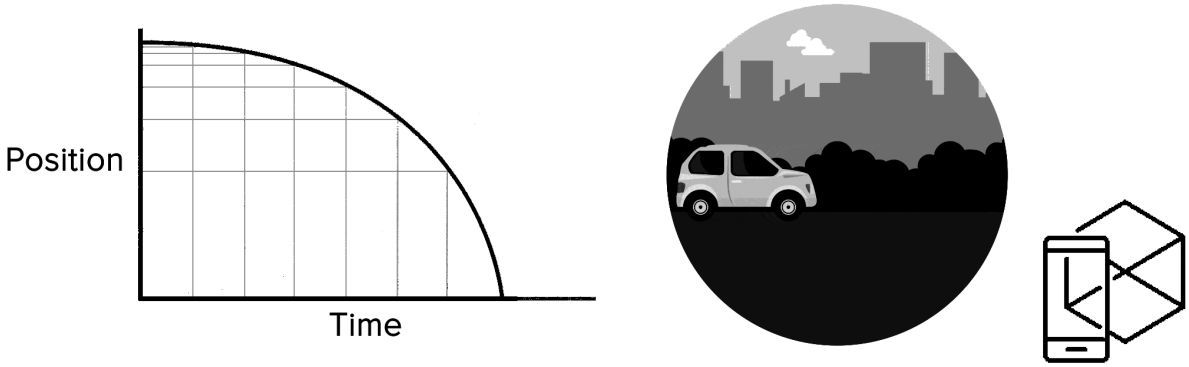


Рисунок 9.18 – Крива прискорення (Ease-in)

Цю криву варто використовувати, коли об’єкт вилітає за межі екрана на повній швидкості. Це можуть бути сповіщення системи чи картки інтерфейсу. Але варто пам’ятати, що це стосується лише об’єктів, які полишають межі екрана назавжди і їх ніяк не можна повернути.

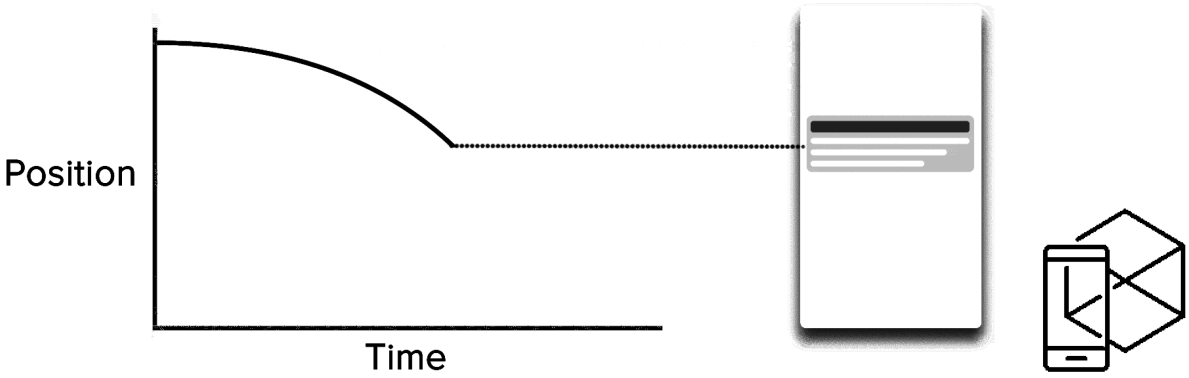


Рисунок 9.19 – Крива прискорення (Ease-in)

Крива анімації допомагає висловити правильний настрій. На прикладі нижче видно, що протяжність руху та дистанція для всіх об’єктів однакові. Але найменші зміни забарвлюють настрій анімації. І звісно, зміна кривої дає можливість рухати об’єкти якнайбільш схожим до реального способом.

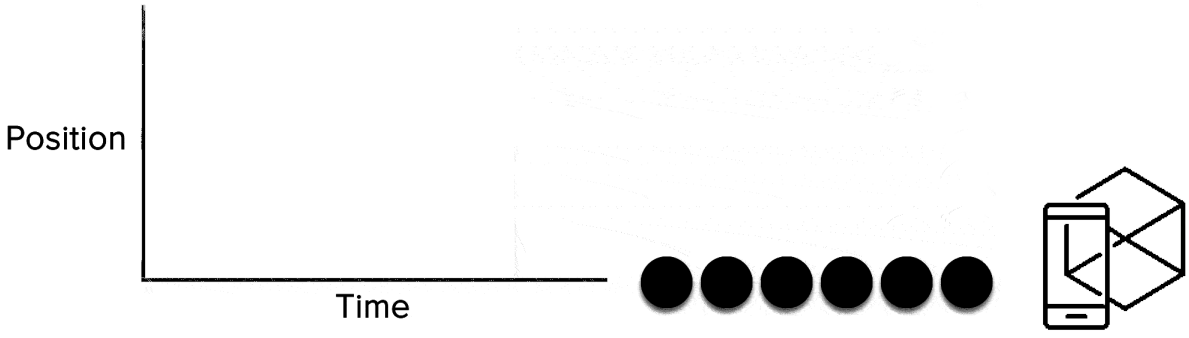


Рисунок 9.20 – Крива прискорення (Ease-in)

9.6.2. Крива сповільнення (Ease-out)

Протилежна до кривої ease-in, крива затримки швидко рухає об’єкт і повільно зменшує швидкість до моменту зупинки. Цей тип кривої краще застосовувати, коли елемент раптово виникає на екрані і заповнює його. Це також стосується різних сповіщень, які вистрибують з-за меж екрану.

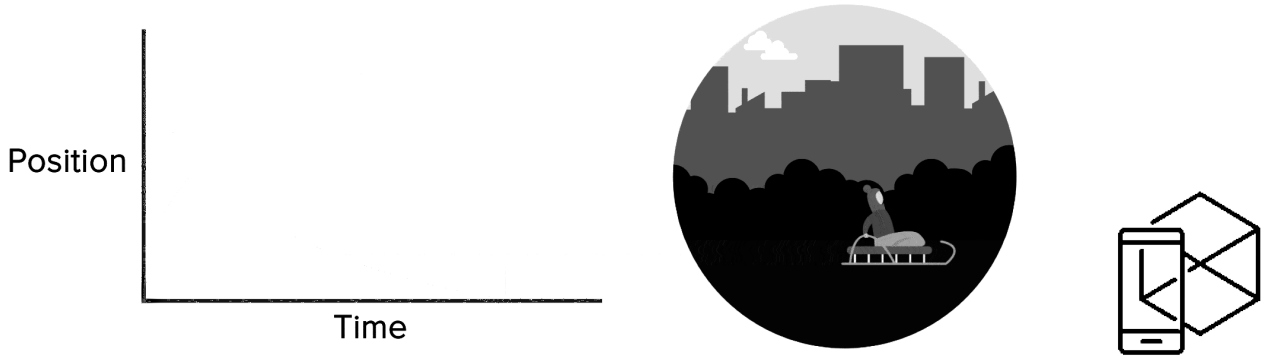


Рисунок 9.21 – Крива сповільнення (Ease-out)

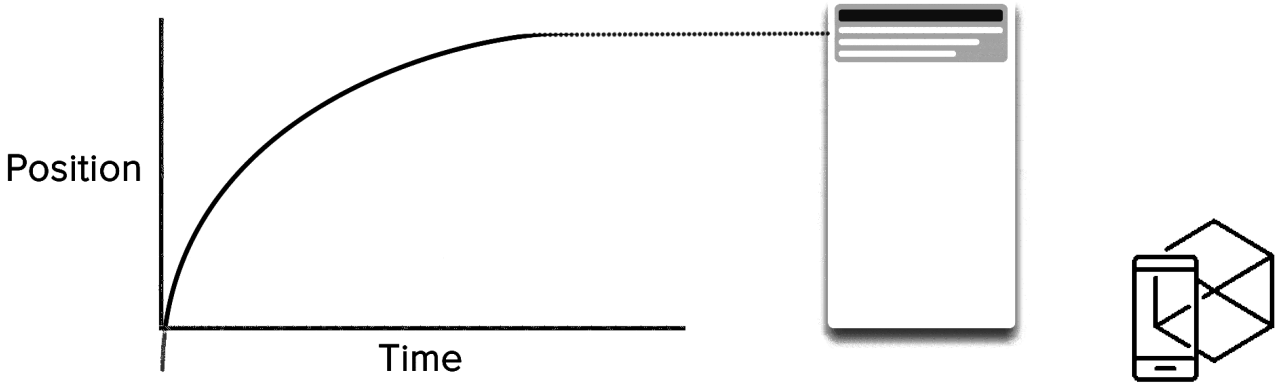


Рисунок 9.22 – Крива сповільнення (Ease-out)

9.6.3. Стандартна крива (Ease-in-out)

Ця крива надає об’єкту певної швидкості і повільно зводить її знову до нуля. Такий тип руху застосовується найчастіше у анімації інтерфейсів. Якщо ви сумніваєтесь, яку криву використати у вашій роботі, обирайте стандартну.

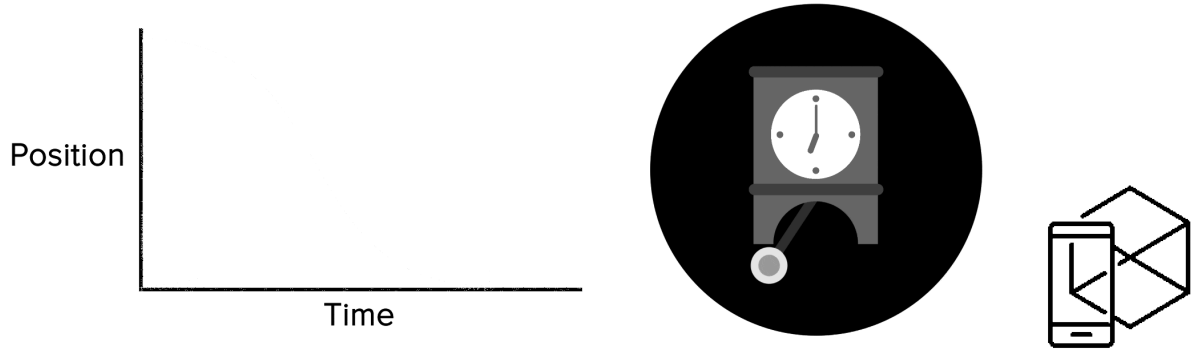


Рисунок 9.23 – Стандартна крива (Ease-in-out)

Відповідно до Material Design Guideline, краще використовувати асиметричну криву, щоб рух здавався більш реалістичним. Закінчення кривої має бути більш підкреслене, ніж її початок. Так, щоб тривалість прискорення була меншою за тривалість сповільнення. У цьому випадку користувач зверне більше уваги на фінальний рух об’єкта і його кінцеве положення.

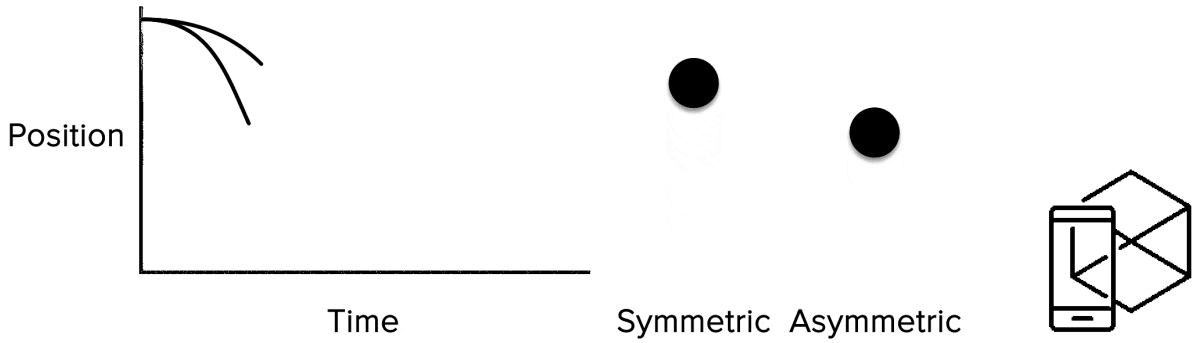


Рисунок 9.24 – Стандартна крива (Ease-in-out)

Ease-in-out стосується руху об’єкта з однієї частини екрана в іншу. Така анімація уникає помітного та драматичного ефекту.

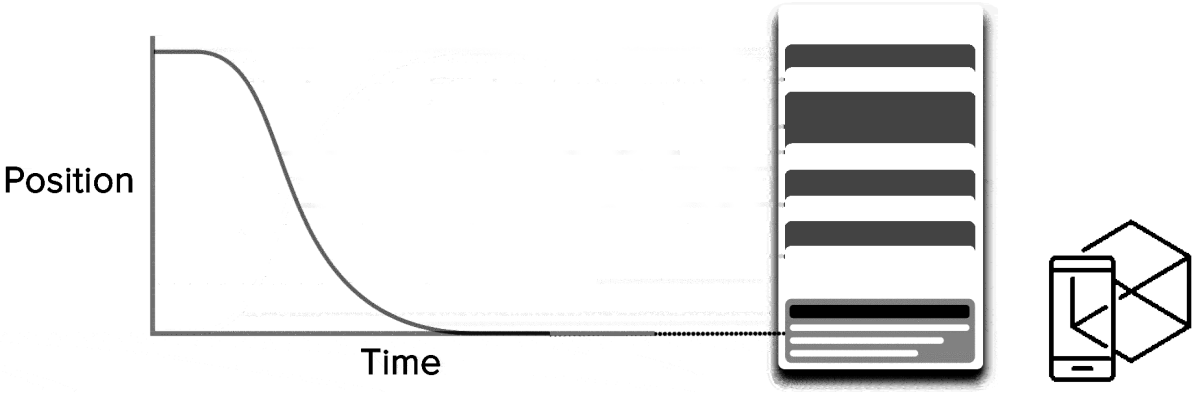


Рисунок 9.25 – Стандартна крива (Ease-in-out)

Той самий тип руху має використовуватися, коли елемент зникає з екрана, але його можна повернути. Це стосується, наприклад, списку навігації.

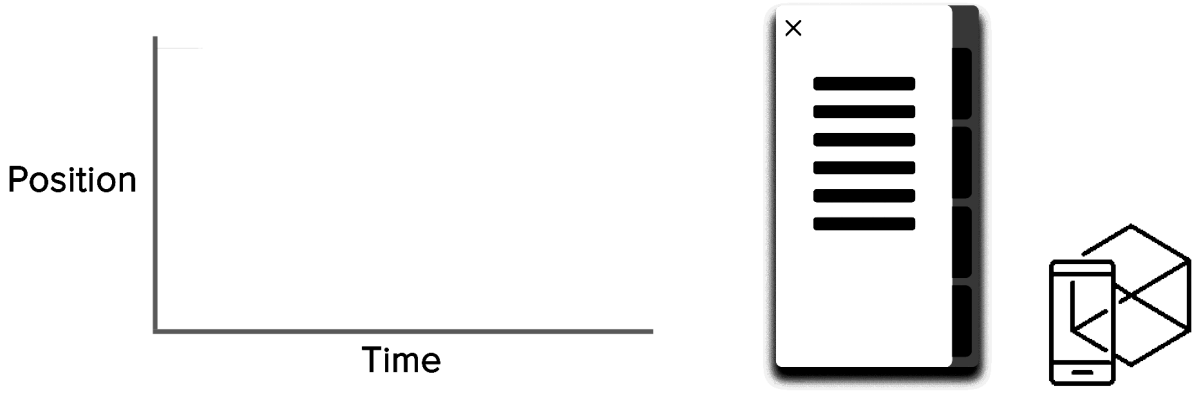


Рисунок 9.26 – Стандартна крива (Ease-in-out)

Із попереднього витікає фундаментальне правило, яким зазвичай нехтують початківці: *Початок анімації не рівний закінченню анімації*

Навігаційний список виникає за кривою сповільнення та зникає за стандартною кривою. Крім того, згідно Google Material Design, час виникнення об’єкту має бути довшим, щоб привернути уваги до нього.

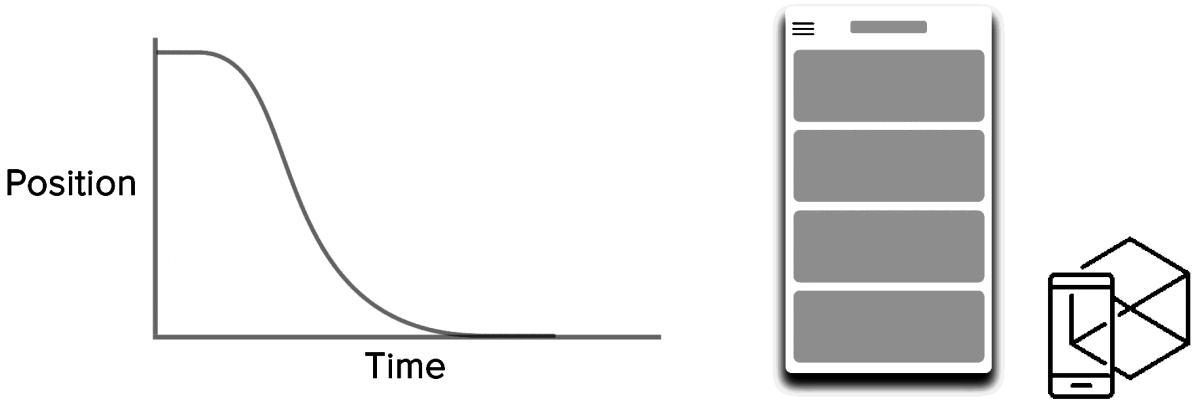


Рисунок 9.27 – Стандартна крива (Ease-in-out)

Для опису кривих використовується кубічна крива Безьє. Вона кубічна, бо побудована на 4 параметрах. Початок графіку — координата 0;0 (зліва внизу). Кінець — координата 1;1 (справа згори). Вони вже позначені на графіку. Перші два параметри — координати X та Y першої точки. Дві інші — координати X та Y другої точки.

Для полегшення роботи із графіками доцільно скористатись ресурсами Easings та Cubic-bezier. На першому є список найбільш вживаних кривих, параметри яких ви можете скопіювати у ваш проект. Другий ресурс дає можливість змінювати різні параметри кривих і одразу ж подивитися на результат.

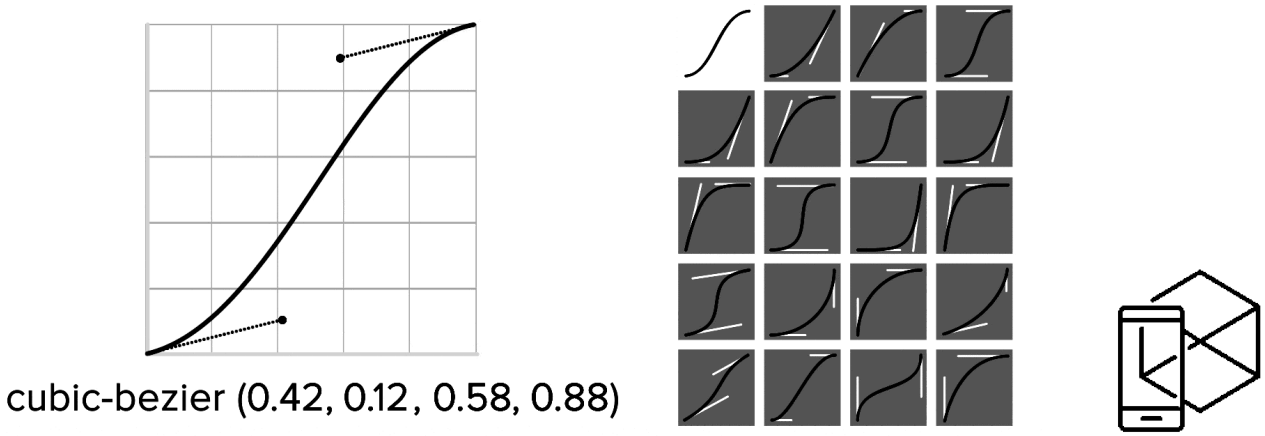


Рисунок 9.28 – Стандартна крива (Ease-in-out)

9.7. Хореографія анімації для UI

Так само як у балеті, ідея анімації полягає у тому, щоб спрямувати увагу користувача у певному напрямку в момент переходу від одного стану в інший. Є два типи хореографії — взаємодія рівності та взаємодія підлеглості.

9.7.1. Взаємодія рівності

Такий тип хореографії означає, що поява усіх об’єктів підкоряється одному спільному правилу. Наприклад, коли поява всіх карток сприймається як єдиний потік, що спрямовує увагу користувача в одному напрямку — згори донизу. Якщо не слідувати порядку, увага буде розсіюватися. А поява усіх елементів водночас також виглядатиме погано.

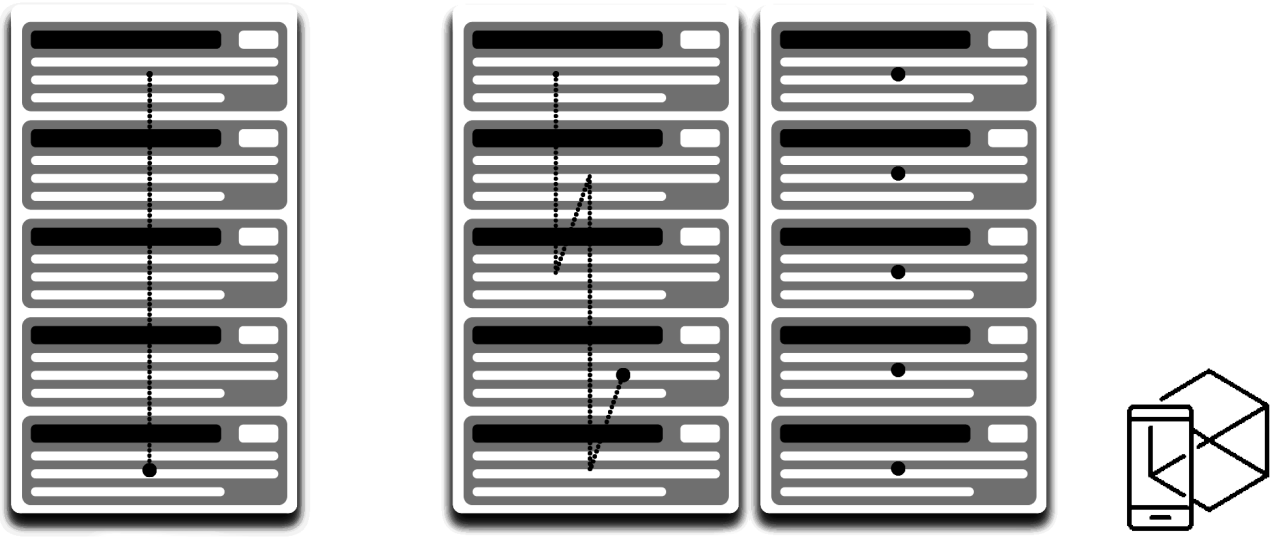


Рисунок 9.29 – Взаємодія рівності

Для вигляду таблиці це стає дещо складнішим. Тут фокус користувача має бути спрямований по діагоналі. Показувати елементи по одному — погана ідея, тому що це зробить анімацію повільною. І увага буде розподілятися зигзагоподібно, а це неправильно.

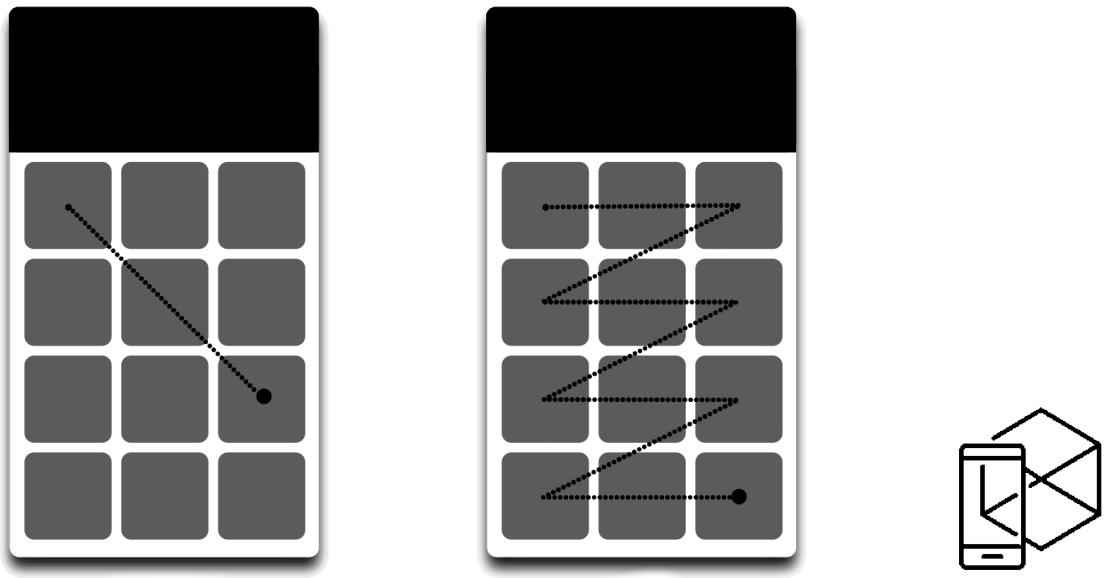


Рисунок 9.30 – Взаємодія рівності

9.7.2. Взаємодія підкорення

Цей тип означає, що є один центральний об’єкт, а всі інші залежать від нього. Така анімація дає відчуття порядку та привертає більше уваги до основного контенту. Часто користувачу складно розібратися, на якому об’єкті сконцентруватися, тому його увага розсіюється. Якщо є кілька елементів, які необхідно оживити, треба чітко визначити послідовність їх руху і анімувати мінімальну кількість об’єктів за раз. Створюйте анімацію тільки для центрального елементу, а решту підкоріть основному. Інакше не буде зрозуміло, за яким об’єктом слідкувати.

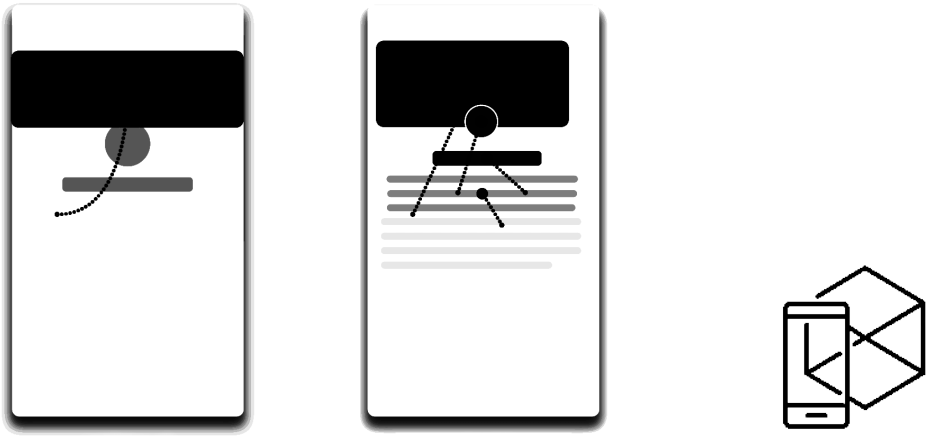


Рисунок 9.31 – Взаємодія підкорення

Згідно Material Design, коли рухливі об’єкти трансформуються непропорційно, вони мають рухатися дугою, а не по прямій. Це допомагає зробити рух більш природним. Пропорційний рух — це зміна висоти та ширини елементу шляхом асиметричного збільшення/зменшення.

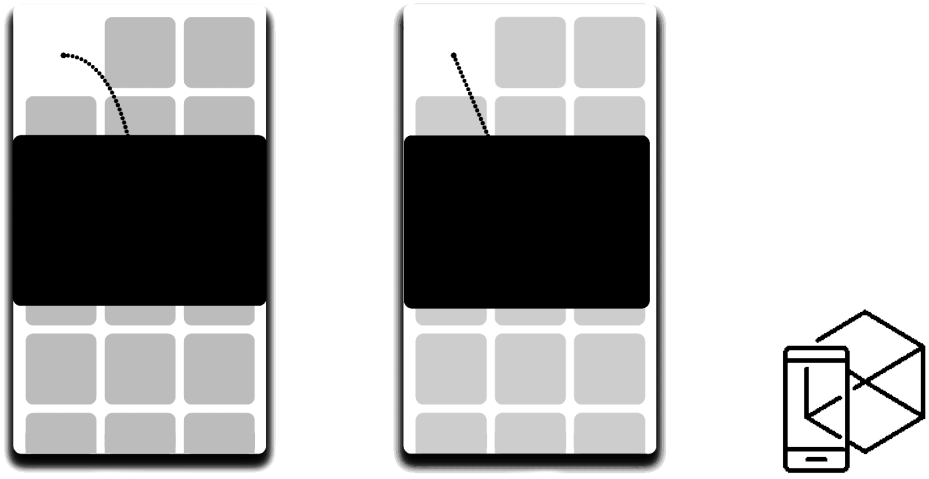


Рисунок 9.32 – Взаємодія підкорення

Рух уздовж лінії використовують коли об’єкт змінює свій розмір пропорційно. Оскільки зробити такий рух набагато легше, правило диспропорційної дуги часто ігнорують.

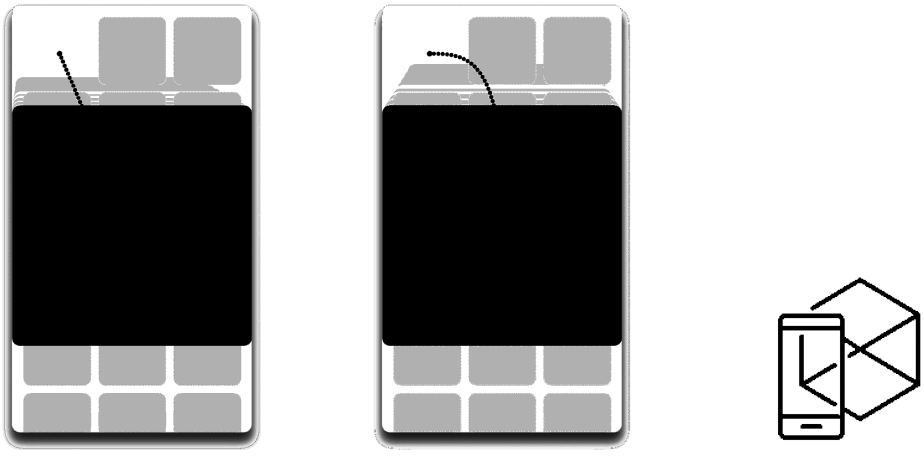


Рисунок 9.33 – Взаємодія підкорення

Руху по кривій можна досягти двома способами: вертикальний і горизонтальний вихід.

- При вертикальному виході крива починає рухатися горизонтально і закінчується вертикальним рухом.
- При горизонтальному — навпаки: об’єкт починає рух вертикально і закінчує горизонтально.

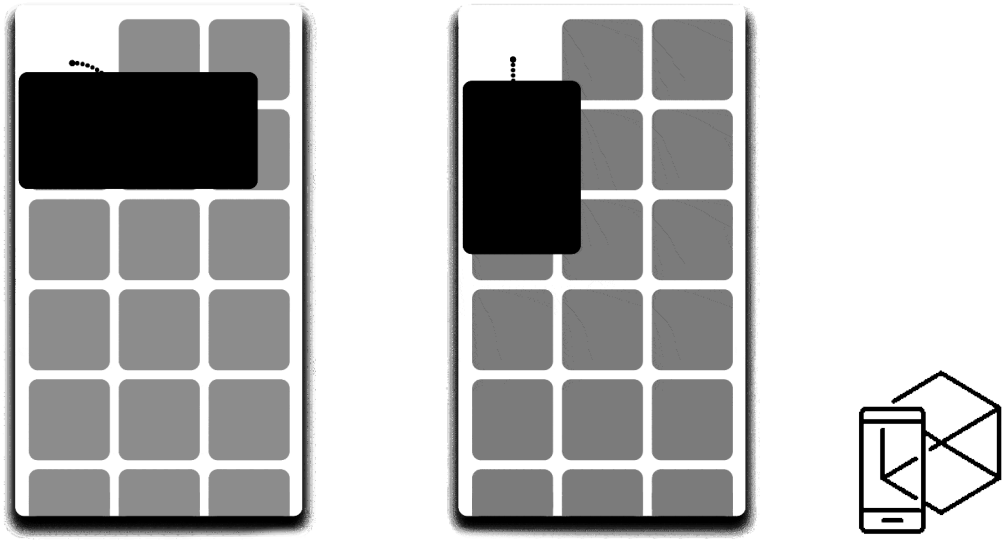


Рисунок 9.34 – Взаємодія підкорення

Якщо шляхи рухливих об’єктів перетинаються, вони не можуть рухатися разом. Об’єкти мають залишати достатньо простору для руху іншої форми, сповільнюючись чи прискорюючись.

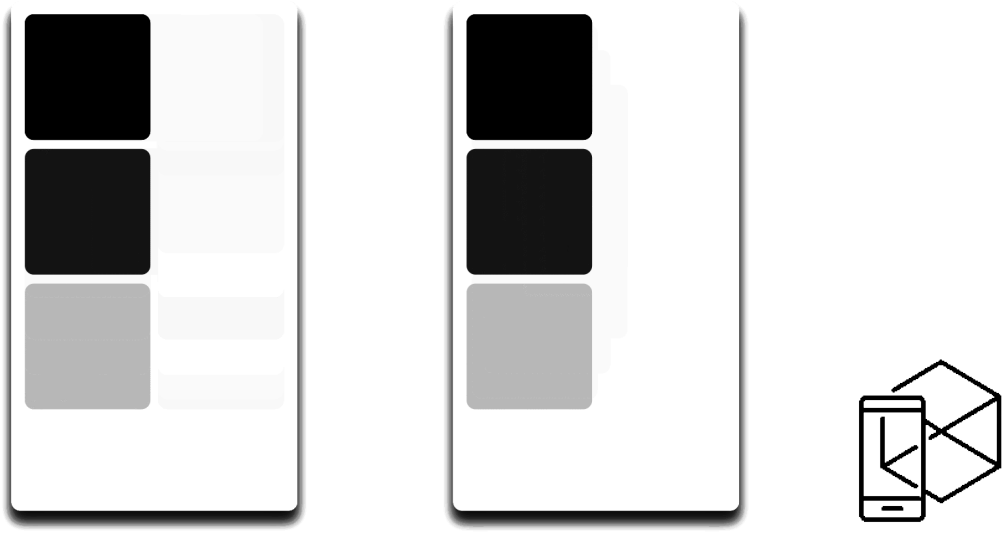


Рисунок 9.35 – Взаємодія підкорення

Також рухлива форма може трохи піднятися над іншими. Але не розчиняючись і не рухаючись через інші об’єкти.

Анімація в інтерфейсі має показувати рухи, які ми знаємо із фізичного світу — тертя, прискорення і так далі. Імітуючи поведінку об’єктів реального світу, ми можемо створити послідовність, яка допомагає користувачам зрозуміти інтерфейс.

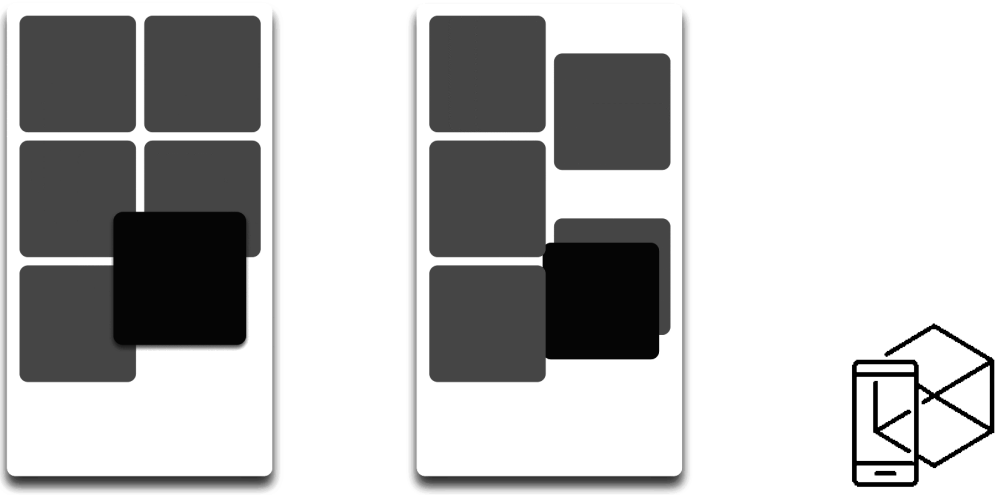


Рисунок 9.36 – Взаємодія підкорення

9.8. Суть анімації мікровзаємодії для макрорезультату

Мікровзаємодія – один з ключових моментів в UI / UX-дизайні. Це, напевно, найкращий доказ того, що увага до дрібних деталей може дати великий і потужний результат. Більшість книг і статей по UX / UI-дизайну несуть одну і ту ж базову ідею: те, що виходить після закінчення процесу дизайну, має бути не тільки красивим, але і бути зручним, приносити користь.

UI / UX-дизайн не просто є твором мистецтва, це продукт, який вирішує проблеми цільової аудиторії, бажано більш швидким і простим шляхом. Потужна основа з досконально продуманих мікровзаємодій може надати дизайну такі ключові характеристики успішного продукту, як юзабіліті, привабливість і функціональність. По суті, **мікровзаємодія** – це один окремий випадок взаємодії користувача з продуктом для виконання конкретного завдання. Наприклад, коли ви натискаєте кнопку like (неважливо, як вона виглядає) і бачите, що ваш лайк відбився в лічильнику (змінилося число, кнопка змінила свій колір або стала неактивна, текст на кнопці відображає, що дія виконана, текст під кнопкою або інший інтерактивний елемент поінформував вас, що ви в списку тих, хто лайкнув, тощо), це теж приклад мікровзаємодії. Коли ви заповнюєте текстове поле пошуковим запитом і відправляєте його в систему, це ще один приклад. Мікровзаємодія трапляються, коли ми додали користувача до своєї мережі контактів в соцмережі або навпаки видалили, коли оцінюємо пост блогу або налаштовуємо таймер – сотні дій, в більшості яких ми не замислюємось, які кроки приведуть до потрібного результату.

В цілому, нескладно зрозуміти, що поведінка користувача в інтерфейсі включає велику кількість мікровзаємодій, і це є тим чинником, що надає показників юзабіліті, ефективності і популярності продукту, якщо про них як слід подбати. Мікровзаємодія в більшості випадків навіть не фіксуються в увазі користувача – і це, насправді, одна з найважливіших завдань з розробки користувацького інтерфейсу: зробити їх максимально природними, чистими і швидкими. Є безліч способів і методів, як задіяти і поліпшити мікровзаємодію. Анімація інтерфейсу – один з таких засобів. Мікровзаємодія, що природним

чином підтримується рухом, складно помітити користувачу, до того моменту, поки вони не стикаються з відсутністю таких мікровзаємодій.

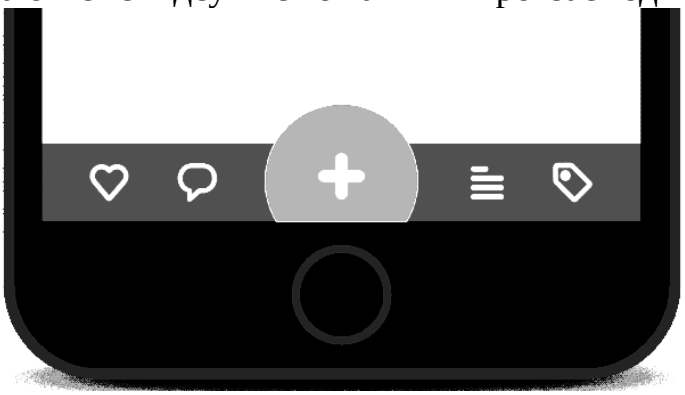


Рисунок 9.37 – GIF-ілюстрація Tap Bar

9.8.2. Навіщо використовувати в дизайні мікро-анімації?

Анімації існують вже давно. Але тепер, коли браузери і мобільні процесори стають все краще, анімації починають грати все більш важливу роль в інтерфейсах. Анімації роблять інтерфейс більш зрозумілим, а процес взаємодії – більш насиченим. Але ось що таке мікро-анімації? І про що потрібно подумати, якщо ми хочемо їх використовувати?

Мікро-анімації – це невеликі, переважно функціональні анімації, які підтримують користувача: забезпечують візуальну зворотний зв'язок і більш зрозуміло відображають зміни, що відбуваються. Мікро-анімації здатні багато чого пояснити без слів.

Інтуїтивні інтерфейси

За допомогою мікро-анімацій користувачеві простіше зрозуміти ієрархічну структуру інтерфейсу, оскільки він бачить елементи в контексті. Це створює фокус. Користувач розуміє кожну операцію, тому що безпосередньо бачить, як вона відбувається.

Прояснюємо зміни і контекст

Відобразити на екрані всю необхідну інформацію – це мистецтво, особливо коли справа стосується невеликих екранів і плоского дизайну. Саме вони найчастіше змушують нас ховати деякі елементи. Проте, можна скористатися мікро-анімаціями, щоб зробити навігацію більш зрозумілою.

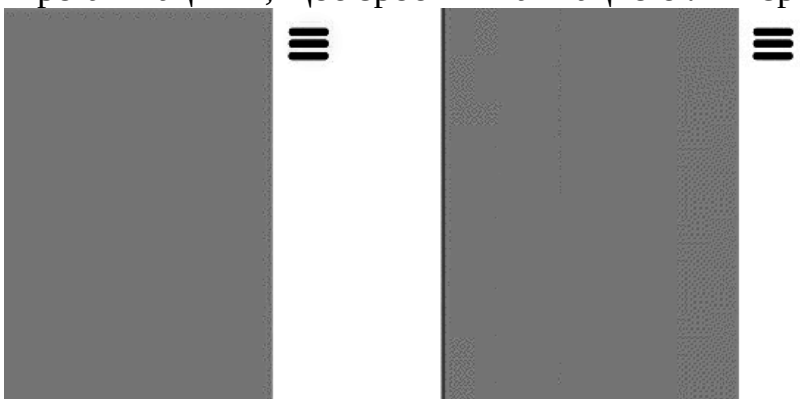


Рисунок 9.38 – Анімація панелі

Візьмемо, приміром, гамбургерне меню. Якщо воно буде відкриватися без анімації, користувачеві буде набагато складніше зрозуміти, звідки воно взялося. Анімація забезпечує контекст, щоб користувач не загубився в інтерфейсі.

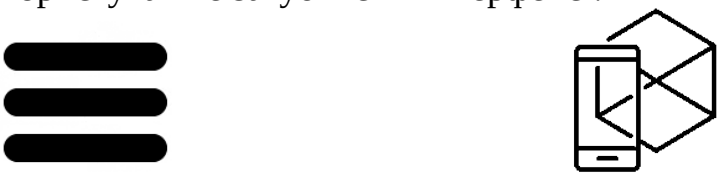


Рисунок 9.39 – Анімація мікровзаємодії бургерного меню

Візьміть іконку гамбургерного меню і додайте анімацію, щоб іконка змінювала форму у міру натискання на неї. Так ви поясните користувачеві, яку дію відбудеться, якщо знову натиснути на цю іконку.



Рисунок 9.40 – Анімація поп-апу реєстрації

Можна піти ще далі. Наприклад, трансформувати кнопку "Залишити коментар" (в даному випадку на голландському: "reageer") в форму. Контекст забезпечується розтягуванням і розширенням елемента.

За допомогою анімацій можна показувати прогрес, а також відповідати на питання, типу: "Я натиснув на цю кнопку?" Або "Завантаження почалася?". Статичний індикатор прогресу теж працює, але його анімована версія дає користувачеві набагато більше – вона акцентує увагу на тому, яка частина шляху вже пройдена. Користувачеві особливо приємно усвідомлювати прогрес при заповненні форм.

Ієрархія і фокус

Створення фокуса – одна з основних причин використання мікро-анімацій. Анімація рухається і привертає увагу, а, значить, з її допомогою ми можемо зосередити увагу користувача на якомусь елементі. Це може бути особливо корисно в момент первинного занурення користувача в суть програми. Зробіть акцент на кнопці, яку користувачеві потрібно натиснути, щоб запустити будь-який процес на сторінці. Але будьте обережні: занадто багато анімацій "розмиють" вашу ієрархію, і повідомлення, яке ви хочете передати, втратить свою вагу. Як відмінно пояснила Софі Пакстон в своїй статті "Your UI is not a Disney movie" (по-російськи: "Ваш інтерфейс – не фільм Діснея"). "Ви ж не зробите кожне речення свого документа заголовком. Те ж саме можна сказати з

приводу анімацій. Вважайте кожен анімований елемент інтерфейсу рівним за значенням заголовку документа. Анімація – як і заголовок – повинна підкреслювати важливість елемента." Ви можете істотно поліпшити свій інтерфейс, якщо будете думати наперед і прогнозувати, як буде вести себе користувач. Пункт меню, який з'являється і працює тільки в певному контексті, може тонко натякнути користувачеві про свою присутність в той момент, коли йому імовірно потрібна саме ця функція.

Коли користувач закриває якийсь розділ, ви можете за допомогою анімації показати, де він зможе знайти цей розділ, коли він йому знову знадобиться. Ось, наприклад опція "Поділитися". Поки ви не зібралися поділитися статтею з друзями, ви не задумуєтесь про конкретні соц. мережах. Але як тільки ви прийняли рішення і натиснули / навели курсор на кнопку "Поділитися", вам відразу надається можлиівсть вибрати, де опублікувати посилання: в Twitter, на Facebook або на LinkedIn.



Рисунок 9.41 – Анімація розшарювання до соціальних мереж

У формах ви можете використовувати анімації, щоб допомагати користувачеві рухатися вперед. Як тільки користувач вибирає будь-які опції, форма може сама прокручуватися до наступного питання. Або, буває, вибір якоїсь опції автоматично збільшує / зменшує список подальших питань. Анімація може прояснювати такі моменти, розширюючи або звужуючи форму, щоб відображалися тільки потрібні питання.

9.8.3. Анімована мікровзаємодія кнопки

Кнопки різних видів і призначення, очевидно, є найпопулярнішими елементами взаємодій. Більш того, вони легко сприймаються користувачами, як елементи, з якими вони можуть активувати функціонал скрін чи веб-сторінки. Вони ідеальні навіть для користувачів без особливих комп’ютерних навичок, так як імітують взаємодію зі звичайними фізичними об'єктами. Кнопки – це елементи, які дійсно впливають на природу взаємодії з користувачем, покращуючи і полегшуючи його, або навпаки, дратуючи і заплутуючи. Кнопка в багатьох випадках самі часто використовувані тригери мікровзаємодії, і життєво важливі елементи навігації. Тому вони потребують належній увазі дизайнера, не тільки в виборі кольору, форми, спецефектів, текстур і положення, а й у визначенні необхідності, доцільності і природи анімації.



Рисунок 9.42 – Анімація гамбургер-меню

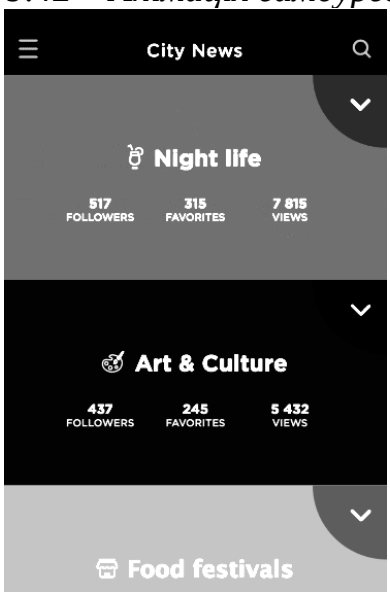


Рисунок 9.43 – Концепт UI-навігації



Рисунок 9.44 – Взаємодія на панелі навігації

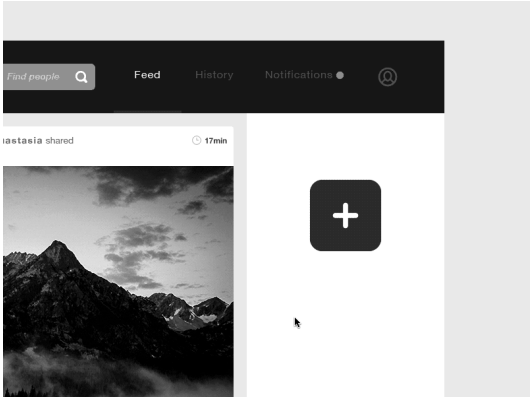


Рисунок 9.45 – Кнопка Add

9.8.4. Анімація «scroll-to reload»

Цей тип анімації для інтерфейсів є найпопулярнішим зараз, і користувачі часто просять її реалізувати на перших порах UI-дизайну для мобільних додатків. З одного боку, одночасно надається два кроки взаємодії з додатком, показуючи, що користувач досяг верхнього ліміту при скроллі контенту, і що вікно оновлене. З іншого боку, вона відкриває неймовірно широке поле для креативу на етапі графічного дизайну інтерфейсу, так як елементи, активовані в цьому процесі, можуть бути не тільки інформативними, а й стильними, розважають користувача. Більш того, цей тип анімації може ефективно використовувати певні елементи брендингу на підтримку впізнаваності бренду, покращуючи запам'ятовуваність лого або емблеми бренду.



Рисунок 9.46 – Анімація «scroll-to reload»

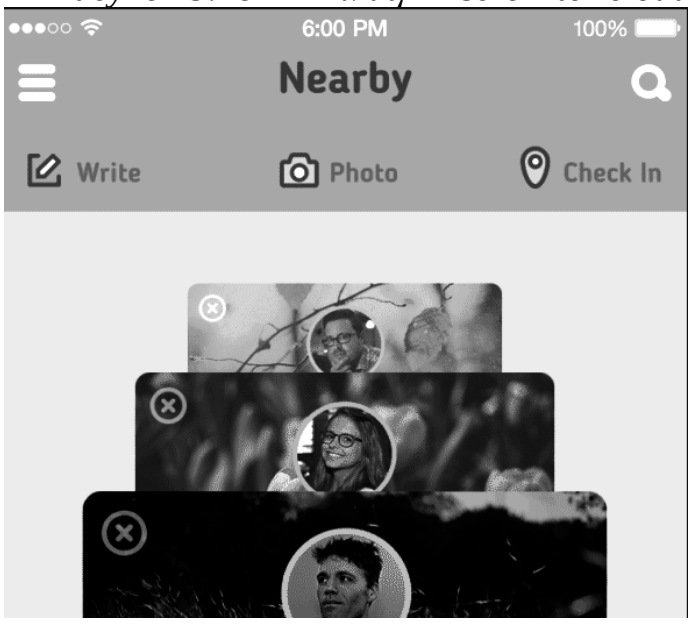


Рисунок 9.47 – Анімація взаємодії «потягни вниз»

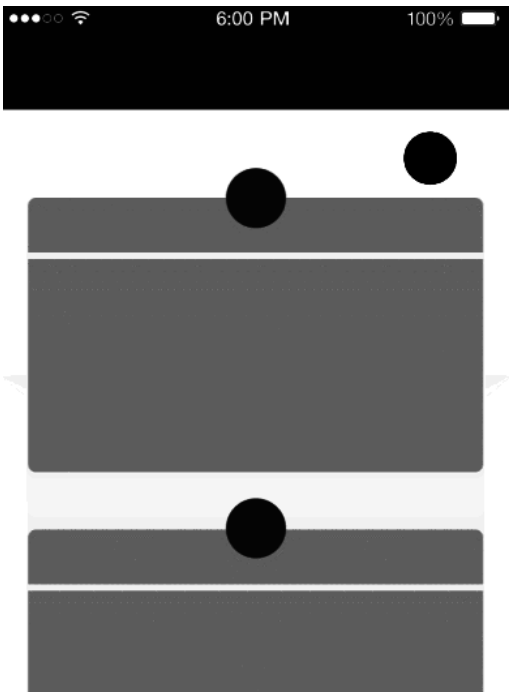


Рисунок 9.48 – GIF-демо «потягніть вниз» – Космічний корабель

9.8.5. Анімація «State-of-process»

Деякі види мікровзаємодії відбуваються миттєво, іншим потрібен час для фіналізації. У будь-якому випадку, для позитивного відгуку у користувачів, їм важливо розуміти, що відбувається в кожен момент з взаємодії з екраном. Тому, процес завантаження і інші подібні процеси, повинні чітко демонструватися користувачу.

Як і в попередньому випадку, можна використовувати деякі стандартні рішення, а також цікаві деталі і ефекти, або елементи брендингу. В цьому випадку необхідно проаналізувати ключову аудиторію, і вирішити, який тип анімації підійде тут краще всього. Чи потрібна смуга або колесо завантаження, або ж якісь додаткові дані на зразок відсотка завантаження та іншого пояснювального контенту.

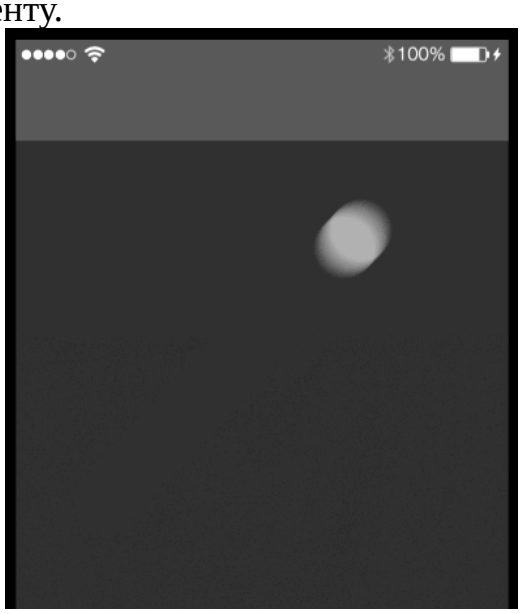


Рисунок 9.49 – Потягніть для оновлення





Рисунок 9.50 – Прелоадер

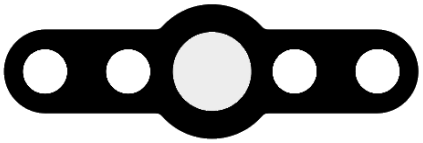




Рисунок 9.51 – Гумовий індикатор

Одна важлива деталь: використання будь-якого типу цифрового інтерактивного продукту повинно забезпечувати комунікацію між користувачем і продуктом, давати відгук від системи. Навіть якщо користувачеві потрібно почекати, користувач повинен бути про це чітко поінформованим. Це основа для всіх типів анімації, описаних вище. Найважливіша мета анімації – пояснювати, спрощувати і прискорювати процес взаємодії, а вже потім створювати так званий «Вау-ефект». Рішення по анімації інтерфейсу повинні реалізовуватися з міркувань юзабіліті, а не чистого прикрашення і розваги. Тому ми закликаємо до здорового глузду у використанні анімацій для мікровзаємодій. Анімація повинна підтримувати користувачів, допомагати їм і застосовуватися там, де вона дійсно доречна, покращуючи юзабіліті, а не відволікаючи користувача і не перезавантажуючи екран.

9.10. Основні правила анімації:

- Вона не повинна перевантажувати сторінку або скрін, викликаючи довге і дратівливе завантаження.
- Потрібно ретельно продумувати анімацію з урахуванням різних пристроїв і умов, з якими користувач стикнеться в роботі з продуктом.
- Анімація не повинна надто відволікати від головних функцій або контенту на екрані або веб-сторінці.
- Вона повинна відповідати загальному стилістичному концепту додатки або веб, щоб підтримувати загальну гармонію в сприйнятті продукту.

Створюючи проекти для ефективних взаємодій, необхідно діяти на перетині дизайну, психології, програмування та інших сфер науки і людської діяльності. Ретельний аналіз цільової аудиторії, тестування ідей і концептів мікровзаємодії зазвичай відкривають двері до більш високого юзабіліті. Мудро використовувана в інтерфейсах анімація може стати одним із способів суттєво прискорити взаємодію, зробити його більш природним, приємним у використанні і більш стильним в дизайні. Якщо анімація побудована правильно, то вона не нав’язується і не відволікає людей від їх цілей. Але не забувайте, що анімація — це радше мистецтво, ніж наука. Тому найкращий варіант — експериментувати і тестувати свої рішення безпосередньо на користувачах.

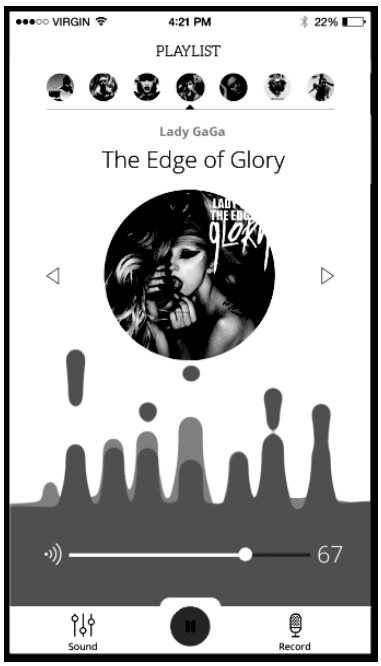


Рисунок 9.52 – Плеєр Juicy

Контрольні питання

1. Що таке анімація? Які бувають види анімації? Принципи анімації
2. Типи анімації взаємодії та мікровзаємодії.
3. Навіщо використовувати в дизайні мікро-анімації?
4. Основні правила та вимоги до анімації взаємодії
5. Перелічити базові трансформації

Теми для самостійного опрацювання

1. Сценарії анімації для інтерфейсу.
2. Шейпові модифікатори. Графіки швидкості. Кейфрейми

Тестові завдання

1. Який з перелічених принципів анімації використовувати не доцільно

a) Пом’якшення (Easing)
b) Крива прискорення (Ease-in)
c) Стандартна крива (Ease-in-out)
d) Розмиття у русі (Motion Blur)

2. Між двома елементами однакового розміру першим під час анімації зупиняється той об’єкт,

a) що пройшов менший шлях.
b) усі відповіді вірні
c) той, що пройшов довший шлях.
d) усі відповіді хибні

3. Обрати НЕ вірне твердження

a) Незалежно від платформи, тривалість анімації залежить не лише від пройденої відстані, але і від розміру об’єкту.
b) Маленькі елементи повинні рухатися швидше, тому анімація з великими та складними елементами повинна бути повільнішою.
c) Між двома елементами однакового розміру першим зупиняється той, що пройшов менший шлях.
d) Залежно від платформи, тривалість анімації залежить від активності користувача у межах додатку та від відстані скролу на об'єкті

4. Одна з основних причин використання мікро-анімацій

a) створення call-to-action
c) створення естетичного вигляду

b) створення фокуса d) нічого з вищезазначеного

5. Гайди з Material Design радять

a) обирати для мобільних пристроїв анімацію, що має середню швидкість анімації, на планшетах — повільнішу, на розумних годинниках — навпаки швидшу, а анімацію веб-сторінок миттєвою

b) обирати для мобільних пристроїв анімацію, що має миттєву швидкість анімації, на планшетах — середньої швидкості, на розумних годинниках — навпаки повільнішу, а анімацію веб-сторінок швидкою

c) обирати для мобільних пристроїв анімацію, що має повільну швидкість анімації, на планшетах — швидшу, на розумних годинниках — миттєвою, а анімацію веб-сторінок середньої швидкості

d) усі відповіді хибні

6. Каркасна анімація відображена

a) 

b) 

c) усі відповіді вірні

d) усі відповіді хибні

7. Завантаження контенту при скролі допомагає

a) не завантажувати сторінки

b) завантажувати повільніше сторінки

c) завантажувати тільки те, що необхідно для візуалізації видимої частини сторінки

d) завантажувати і миттєво відображати вміст сторінки

8. Анімація руху здійснюється за допомогою

a) направляючого шару c) траєкторії або шкали

b) провідної верстви d) вірна відповідь відсутня

9. Що таке таймінг?

a) це час падіння, відскоку, повороту і повернення назад об'єкта;

b) це тривалість фільму;

c) це один з видів анімації;

d) це розрахунок часу, необхідного для створення руху об'єкта.

10. Щоб зробити анімацію,

a) потрібно опрацювати окремо кожен кадр

b) створити тільки ключові кадри, інші створяться автоматично

c) між ключовими кадрами використовувати переходи

d) всі відповіді правильні.

Практичні завдання

Завдання 9.1. Створити анімацію взаємодії та мікроанімацію:

а) Анімація завантаження даних;

б) Анімація елементів керування для інтерфейсу;

в) Анімована мікровзаємодія кнопки тощо.

ВІДПОВІДІ НА ТЕСТОВІ ЗАПИТАННЯ

1

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
a)	b)	b)	c)	c)	d)	b)	b)	b)	c)

2

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
b)	c)	c)	a)	b)	b)	a)	b)	d)	d)

3

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
a)	b)	c)	a)	a)	d)	b)	a)	c)	b)

4

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
a)	d)	c)	b)	a)	b)	c)	a)	d)	b)

5

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
c)	a)	b)	a)	a)	b)	b)	d)	c)	a)

6

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
a)	b)	a)	b)	a)	b)	c)	a)	c)	a)

7

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
a)	a)	c)	a)	c)	a)	a)	b)	a)	d)

8

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
a)	b)	a)	b)	c)	c)	a)	d)	b)	c)

9

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
d)	a)	d)	b)	b)	a)	c)	c)	d)	d)

ТЕЗАУРУС

А/Б-тестування – UX дослідження, в ході якого перевіряється, яка з двох версій онлайн-контенту більше подобається користувачам.

Адаптивний дизайн (Responsive) – це вид дизайну сайту (додатку), який дозволяє відображати сайт (додаток) однаково коректно на пристроях з будь-яким розміром екрану.

Аналіз задач – процес, під час якого дизайнери аналізують дії, які користувачеві потрібно зробити, щоб вирішити своє завдання на сайті або в додатку.

Аналіз клікстриму – дає уявлення, що цікавить вашого користувача, які сайти крім вашого він відвідує і на які сайти йде.

Базова лінія (Baseline) – термін використовується щодо типографіки – це уявна лінія, на якій стоять усі літери в рядку.

Бекенд (Back End) – це зворотня частина сайту, яка невидима відвідувачеві, але яка безпосередньо відповідає за роботу сайту і виконується на стороні сервера. Бекенд є адміністративна частина системи.

Білий простір (Negative Space) – термін, що позначає вільний простір, утворений між елементами на сторінці, всередині або навколо них; область, що не містить текст або зображення.

Вайрфрейми (Wireframes) – образ дизайну низької точності (lo-fi), що візуалізує дизайн-проект гранично точним чином, містить найнеобхідніші елементи та вміст та допомагає отримати зворотній зв'язок на початковому етапі проектування програмного забезпечення.

Векторні зображення (Vector) – спосіб представлення зображення в комп'ютерній графіці, заснований на математичному описі основних геометричних форм (точки, лінії, кола, багатокутники, криві Безьє). Основною відмінністю від растрових зображень є те, що при зміні розміру якість зображення не змінюється.

Верхня смуга (Above the fold) – це термін, який взято з лексикону старих друкованих газет, в яких найважливіші новини розміщувалися в «верхній половині смуги», щоб їх помічали в першу чергу. У дизайні «верхня частина смуги» означає ту частину екрану, яка повністю видна без прокрутки вниз і вміщує основний контент повинен розміщуватися.

Вирівнювання (Alignment) – розташування об'єктів щодо інших об'єктів. Наприклад зліва, справа, по центру тощо.

Віддалене дослідження юзабіліті – процес, в ході якого досліджують те, як користувачі взаємодіють з інтерфейсом. Особливість в тому, що респонденти знаходяться на своїй території, а організатори дослідження використовують програми для спільного використання екрана. Ці програми записують обличчя і голос респондента. Перевага такого дослідження у віддаленості респондентів, і як наслідок, в дослідженні може брати участь велика кількість користувачів, та економиться час.

Вільний дизайн — концепт, розроблений С. Голдсмітом, який переформулював цитату Л. Мейса в своїй книзі «Проектування для інвалідів». Він придумав

концепт «вільного дизайну». Пізніше цей термін став відомі як «доступний дизайн».

Внутрішній відступ (Padding) – внутрішній відступ елемента.

Гештальт-принцип (закон завершеного сприйняття) – свідчить, що люди сприймають об'єкти як частина чогось цілого, а не як набір не пов'язаних один з одним предметів.

Графічний дизайн (Visual Design) – художньо-проектна діяльність, спрямована на створення візуальної графічної програми або системи. Графічний дизайн підтримує розвиток соціально-економічної та культурної сфер життя, сприяючи формуванню значимого візуального ландшафту.

Діаграма Ісікави – допомагає візуально уявити список причин якоїсь проблеми. Діаграма складається з «голови» – проблеми, і «кісток» – причин, які розташовані вздовж хребта. Така схема може бути деревовидної і мати чотири-п'ять рівнів вкладеності.

Діаграма потоку завдань – діаграма, за допомогою якої можна відобразити користувацький сценарій: послідовність завдань, які користувачеві потрібно зробити в додатку або на сайті.

Довга прокрутка (Long scrolling) – структура, де довгі сторінки вимагають постійної прокрутки вниз.

Дозвіл (Resolution) – позначає кількість пікселів на дюйм.

Дослідження бізнес-середовища продукту – збір і аналіз інформації про те, що вплине на користувацький досвід: про майбутніх користувачів, їхні цілі та інструменти, що допомагають ці цілі досягти; про фізичному середовищі, в якій продукт буде використовуватися; про технічні обмеження. Інформація для аналізу збирається за допомогою інтерв'ю, воркшопів, опитувань, перегляду схожих продуктів, роботи фокус-груп, польових досліджень і відповідних пошукових запитів.

Дослідження істинного наміру – допомагає описати досвід користувача від взаємодії з сайтом. В ході цього дослідження дизайнери опитують користувачів, щоб дізнатися відповіді на наступні питання: хто відвідує сайт? чому ці люди приходять на сайт? що їм подобається і не подобається? наскільки успішно сайт допомагає їм справлятися із завданнями? Відповіді на ці питання допомагають дизайнерам краще розуміти користувачів і то, як задовольнити їхні потреби.

Дослідження користувачів – дослідження з метою зрозуміти потреби користувачів, їх звички і мотивації. Для цього використовується спостереження, аналіз завдань і інші методи.

Доступний дизайн (Accessebility design) – це проста, пряма і швидка взаємодія між користувачем і інтерфейсом, створена для людей з обмеженими можливостями. Такий підхід – частина інклюзивної дизайн-стратегії.

Доступність (Accessebility) – показує те, наскільки легко людям працювати з інтерфейсами сайтів і додатків. У тому числі це стосується людей з обмеженими можливостями. Приклад хорошого аксессибіліті веб-інтерфейсу для дальтоніків – адаптація кольорів під їх можливості.

Інклюзивний дизайн – дизайн, розроблений під потреби людини, інтерфейс, який легко підходить різнорідної групи людей з різними характеристиками і особливостями. У всіх людей відрізняються вік, стать, мову, якою вони говорять, умови середовища, в якій вони живуть. Деякі обмежені в доступі до техніки, в кількості часу і грошей. Інклюзивність робить дизайн доступним і зручним для всіх і кожного.

Ергономічність – пристосованість для використання, наявність умов, можливостей для легкого, приємного, необтяжливо користування чим-небудь або задоволення будь-яких потреб, потреб.

Еталонне тестування юзабіліті – фіксування поточного стану юзабіліті в якості еталонного або в якості «точки відліку». З ним будуть порівнюватися всі наступні зміни.

Закон Парето - якщо застосувати цей закон до сайту або додатку, він буде говорити, що 20% функцій продукту принесуть 80% користі. З цього можна робити висновок, що 80% продукту створюються даремно. Мова йде про те, що, швидше за все, тільки 20% функцій забезпечать успіх продукту.

Закон Фіттса- правило, що визначає, як швидко можливо попасти по кнопці чи полю на сайті або в додатку. Чим далі і менше мета, тим більше потрібно часу, щоб потрапити до неї. Це правило потрібно враховувати, коли створюються елементи управління на сайті або в додатку. Кнопки, поля введення, бігунки і інші елементи повинні бути досить великими і розташовані там, де цього очікує користувач.

Закон Хіка – описує швидкість прийняття рішення. При цьому рішення – це результат вибору. Чим більше варіантів, тим більше часу потрібно людині для прийняття рішення.

Залученість – показник того, наскільки добре продукт привертає і утримує призначене для користувача увагу. Для зростання залученості важливо враховувати все, будь то кнопка або меню на сторінці або екрані.

Зовнішній відступ (Margin) – зовнішній відступ від елемента.

Зручність читання (Readability) – властивість текстового матеріалу, що характеризує легкість сприйняття його людиною. Слід розрізняти легкість для читання тексту з точки зору: поліграфічного виконання тексту та лінгвістичних особливостей текстового матеріалу.

Кернінг (Kerning) – відстань між конкретними парами літер, зміна інтервалу між окремими літерами в залежності від їх форми.

Кількісні дослідження – дослідження поведінки користувачів, сфокусоване на чисельних даних і статистиці. Проводиться за допомогою опитувань, анкетування і огляду.

Клікстрим – інформація про те, які сайти відвідував користувач до того, як прийти на ваш сайт, і після того, як він звідти пішов.

Конкурентний аналіз – оцінка сильних і слабких сторін нинішніх і потенційних конкурентів.

Користувацький інтерфейс (User interface) – інтерфейс, що забезпечує передачу інформації між користувачем-людиною і програмно-апаратними компонентами комп'ютерної системи.

Користувацькі маршрути – це послідовність кроків, які виконує користувач, щоб завершити свою задачу. Маршрут, за яким користувачі переміщаються всередині сайту найчастіше – це найпопулярніший маршрут.

Криві Безьє (Bezier Curve) – параметричні криві, які представляють векторний шлях в комп'ютерній графіці.

Людино-машинний інтерфейс – широке поняття, що охоплює інженерні рішення, що забезпечують взаємодію людини з керованими їм машинами.

Ментальна модель – сума уявлень дизайнера про користувача, яка дозволяє орієнтувати функції сайту або програми під конкретні користувацькі цілі. Чим ближче функції продукту до ментальної моделі користувача, тим продукт зручніше.

Метод складання вільного списку – техніка збору інформації, яка допомагає дізнатися, як користувач розуміє певні питання. Для цієї людини просять у форматі списку перерахувати все, що він думає стосовно якогось поняття.

Модульна сітка (Grid) – набір вертикальних і горизонтальних ліній, які формують систему для розташування об'єктів.

Мокап (Mockup) – високо деталізований (hi-fi) статичний дизайн-образ інтерфейсу програмного засобу, що використовується для презентації загального відображення дизайну кінцевого продукту.

Навігація (Navigation) – це система посилань, яка дозволяє користувачеві переміщатися по сайту або додатком.

П'ятисекундний тест – користувачам показують інтерфейс програми або сайту протягом п'яти секунд. Потім їм пропонують перерахувати все, що вони запам'ятали. Так можна дізнатися, чи працюють ключові візуальні образи і заклики до дії.

Пагінація (Pagination) – нумерація сторінок, або ж розбиття цілісного блоку інформації на сторінки. Під поняттям Pagination в інтернет-мережі розуміють показ обмеженої частини інформації на одній сторінці. Це необхідно для зручності сприйняття. Пагінація на сайті дає можливість упорядкувати інформацію, що значно полегшує сприйняття даних.

Персони (User Personas) – образ типового користувача, який дизайнери формують з доступної інформації та інтерв'ю, це вигадані персонажі, які володіють набором якостей, що є характерними для цільової аудиторії. Особисті дані персони можуть бути вигаданими, але інформація, що формує ментальну модель користувача, повинна бути реалістичною. Персони використовуються для ідентифікації цілей користувача, мотивації користувачів тощо.

Плитковий стиль (Style Tiles) – цей структурний стиль став популярний завдяки успіху мережі Pinterest. В його основі лежить структура, що складається з колонок, і паралельна, але не симетрична подача матеріалу у

вигляді «плиток». В результаті виходять різнорозмірні сегменти, які при цьому зберігають організований вигляд.

Поза екраном (Below the Fold) – термін прийшов з газетного справи і позначає частину контенту сайту, яка не вмістилася в межах видимої частини екрану сайту або програми. Тобто, щоб переглянути цей контент, користувачеві доведеться проскролити (прокрутити) сторінку вниз.

Правило трьох кліків – користувачі повинні знайти потрібну сторінку за три кліка, інакше вони підуть з сайту.

Принцип KISS (Keep it Simple Stupid) – це дизайн-принцип, проголошений ВМС США в 1960-м. Він говорить, що більшість систем працює добре, якщо їх строй очевидний. Очевидність – це ключова мета дизайну.

Прототип (Prototype) – середньо-високе точне відображення (lo-hi) кінцевого продукту, що є динамічною симуляцією остаточної взаємодії між користувачем та інтерфейсом.

Растрові зображення (Raster, Bitmap) – зображення, що представляє собою сітку пікселів на екрані пристрою. Редагується за допомогою растрових графічних редакторів. При зміні розміру істотно втрачається якість зображення.

Рівень конверсії – відсоток відвідувачів, які виконали цільові дії на сайті або в додатку. Рівень конверсії збільшується в ході її оптимізації.

Сайт візитка (Promo website) – так званий односторінковий сайт, який має буквально одну сторінку, на якій сконцентрована вся інформація.

Скевоморфізм (Skeuomorph) – це принцип, який має в основі запозичення вигляду елементів дизайну з фізично існуючих об’єктів.

Скетчі (Sketches) – швидкі, позбавлені деталей замальовки майбутнього дизайну.

Сортування карток – метод, що покликаний перевірити, чи зрозуміла кінцевому користувачеві інформаційна архітектура сайту або програми. За допомогою стікерів або онлайн-інструментів непричетні до проекту люди сортують функції або назви сторінок сайту за категоріями. Назви для категорій або придумують учасники (відкрите сортування), або вони вже задані (закрите сортування).

Спільний дизайн – підхід до дизайн-роботи, який передбачає, що в ньому беруть участь всі зацікавлені сторони: замовники, партнери, клієнти та кінцеві користувачі. Мета – переконатися, що результат задовольнить всіх причетних.

Технологія айтрекінгу (Eye tracking) – фіксування активності очей людини, що спостерігає за об'єктом на екрані або в просторі. У цьому допомагають програми, які стежать за рухами очей користувачів на екрані і показують, в якому порядку і на що користувачі дивляться найчастіше.

Трекінг (Tracking) – рівномірна зміна відстані між усіма буквами.

Універсальний дизайн – концепт, придуманий американським архітектором Рональдом Л. Мейсом. Він виділив 7 принципів «універсального дизайну», які описують, як створити хороші умови для людей різного віку і здібностей.

Він вірив, що простий доступ до будівель і свобода руху зроблять життя простіше для всіх – не тільки для людей з обмеженими можливостями. Пізніше цей термін став відомі як «інклюзивний дизайн».

Фокус-група – вид дослідження, який використовують, щоб зібрати думки людей про користувачів, продукт, концепцію, прототип, завдання, стратегію і середовище. Під керівництвом модератора група людей обговорює питання, що пов'язані з продуктом.

Фронтенд (Front End) – частина сайту, що виконується на стороні користувача. В інтернеті в якості фронтенду виступають HTML-верстка, стилі CSS і JavaScript, фронтенд є лицьовою частиною сайту, яку бачить користувач.

Хлібні крихти – один з типів відображення навігації.

Цільова аудиторія – одна або кілька груп користувачів, для яких створюється продукт.

Шкала Лікерту – метод дослідження, в якому людину просять проранжувати свою відповідь за шкалою з позитивними, нейтральними та негативними оцінками.

Щоденникові дослідження – метод збору якісних даних про поведінку користувачів. Через рівні проміжки часу користувачі залишають в спеціальному журналі записи про своє щоденне користування продуктом. Дослідження можуть тривати від кількох днів до місяця. Вони показують, як змінюються звички користувачів, а також те, навіщо вони використовують продукт.

Юзабіліті (usability, зручність використання) – здатність продукту бути зрозумілим, досліджуваним, використовуваним і привабливим для користувача в заданих умовах; властивість системи, продукту або послуги, при наявності якого конкретний користувач може експлуатувати систему в певних умовах для досягнення встановлених цілей з необхідною результативністю, ефективністю та задоволеністю.

Юзабіліті-лабораторія – простір для юзабіліті-тестів, де ведуться спостереження і записи того, як користувач взаємодіє з системою. Спостерігачі стежать за тестом через двосторонні дзеркала або по відеотрансляції, перебуваючи в іншій кімнаті.

Якісні дослідження – дослідження поведінки людей, що відповідає на питання «як?» і «чому?», а не на питання «скільки?». У якісному дослідженні важливо те, що визначає цілі та поведінку користувача, а не чисельні дані, їх описують.

Bootstrap – вільний набір інструментів для створення сайтів і веб-додатків, що зорієнтований на адаптивність під мобільні пристрої.

CMYK – субтрактивна схема формування кольору, яка використовується в поліграфії. У CMYK використовуються чотири кольори – ***Cyan*** (блакитний), ***Magenta*** (пурпурний), ***Yellow*** (жовтий) і ***blacK*** (чорний) для формування кольору.

CX – customer experience (Купівельний досвід) – включає в себе досвід бренду, найчастіше — інтернет-магазинів. CX-дизайн орієнтується на кілька

факторів: зручність використання сайту, відгуки користувачів, індекс лояльності покупців – NPS (Net Promoter Score).

DPI, PPI – кількість точок, які поміщаються на один дюйм екрану (PPI) або друкованого вібтку (DPI).

Flat design – плоский дизайн від Microsoft, філософія дизайну, в основі якої лежить простота і функціональність. У елементів, створених за принципом «плоского дизайну», немає об’єму. У ньому відсутні техніки, спрямовані на надання глибини і об'ємності: немає градієнтів, тіней, текстур і відблисків, які надають об'єкту реалістичний вигляд. Замість надання об'єму, плоский дизайн звертається до основ графіки – яскравих кольорів, простих форм, кнопок і іконок.

HEART-методологія - дозволяє вимірювати якість користувацького досвіду у великому масштабі по параметрам, які показують ставлення користувачів до продукту: щастя (***H*** – ***happiness***), залученість (***E*** – ***engagement***), прийняття (***A*** – ***adoption***), утримання (***R*** – ***retention***) і успішність досягнення мети (***T*** – ***task***).

HSB – триканальна схема формування кольору. Назва походить від перших букв англійських слів: ***Hue*** (колірний тон), ***Saturation*** (насиченість), ***Brightness*** (яскравість).

IA (інформаційна архітектура) – створення структури веб-сайту, додатка або ж іншого проекту, яка дозволяє розуміти, де ми знаходимося, де саме знаходиться бажана інформація щодо нашого становища. Визначає те, як структуровано додаток або сайт, а також як організований контент. Головна мета – допомогти користувачеві знайти інформацію, а також виконати завдання. Інформаційна архітектура призводить до створення карт веб-сайту, ієрархій, навігації, класифікацій, метаданих.

IBM Design Language - приділено увагу золотому перерізу і варіантам сітки, рекомендацій щодо розмірів відступів та філософський тон в дизайні, але немає ніяких шаблонів і обмежень.

IxD (проектування взаємодії) – проектування взаємодії визначає поведінку і структуру інтерактивних систем. Дизайнери взаємодії створюють відносини між людьми, послугами і продуктами, якими ті користуються, від мобільних пристроїв і комп'ютерів до електроприладів. Дизайнери IxD відповідають за різні рухомі елементи.

Landing Page (Лендінг, цільова або посадкова сторінка) – це веб-сторінка, зібрана певним чином, головним завданням якої є збір контактних даних аудиторії або продаж товару. Цільова сторінка зазвичай містить інформацію про товар або послугу.

Material design – принцип дизайну інтерфейсів для операційної системи Android від компанії Google. Ідея дизайну полягає в інтерфейсі, поведінка і вигляд якого наслідують правила поведінки і вигляду паперових карток в реальному житті.

Metro стиль - стиль, заснований на принципах дизайну швейцарського стилю. Основними принципами Metro є акцент на типографіці і великих текстах, що

відразу кидається в очі. Microsoft називає Metro «простим, чистим, сучасним», а також «оновленням» в порівнянні з заснованими на значках інтерфейсів Android і iOS.

MVP (minimum viable product) – продукт, який містить мінімальний набір функцій, необхідних для функціонування. Створення MVP дозволяє оцінити, наскільки продукт цікавий людям, і чи варто його розвивати надалі.

RGB – адитивна схема формування кольору. Потрібний колір утворюється шляхом змішування світлових випромінювань трьох первинних кольорів – червоного, зеленого і синього (***Red***, ***Green***, ***Blue***). Змішання 100% первинних кольорів дає білий колір.

SD (service design, дизайн послуги) – це організація людей, інфраструктури, комунікацій з метою підвищення якості взаємодії між постачальником послуг і клієнтами. Мета SD – в проектуванні зручної, зрозумілої послуги для клієнтів, яка буде успішно конкурувати з іншими пропозиціями на ринку.

UCD (User-Centered Design, орієнтований на користувача дизайн) – це стратегія проектування і процес, в якому потребам, бажанням і обмеженням кінцевих користувачів продукту приділяється велике увагу на кожній стадії процесу проектування.

UID (дизайн користувацького інтерфейсу) – це дизайн для програм і пристроїв, таких як комп'ютери, побутова техніка, мобільні телефони та інші електронні пристрої з акцентом на якість досвіду використання. Метою дизайну UI є створення максимально простої і продуктивної взаємодії з точки зору досягнення цілей користувача.

UX дизайн (користувацький досвід) – поняття включає в себе те, як продукт функціонує, і, які емоції викликає і користувачів.

ВИКОРИСТАНА ЛІТЕРАТУРА

1. Warfel T. Z. Prototyping : a practitioner’s guide. Rosenfeld Media, 2009. 195 p.
2. Пасічник О. Г., Пасічник О. В., Стеценко І. В. Основи веб-дизайну : навч. посіб. Київ : Вид. група BHV, 2009. 336 с.
3. Синєпупова Н. Композиція: Тотальний контроль. Київ : ArtHuss, 2020. 240 с.
4. Чемерис Г. Ю., Осадча К. П. Проектування користувацького інтерфейсу : навч. посіб. для викладачів та студентів закладів вищої освіти, Мелітополь : ФОП Однорог Т., 2019. 300 с.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Браун Д. Web-дизайн и публикация. Энциклопедия пользователя. Київ : ДиаСофт, 1998. 656 с.
2. Голобородько В. М. Ергономіка для дизайнерів : підручник. Харків : ХДАДМ, 2012. 378 с.
3. Емброуз Г., Леонард Н. Основи. Графічний дизайн 02. Дизайнерське дослідження. Пошук успішних креативних рішень. Київ : ArtHuss, 2019. 192 с.
4. Емброуз Г., Леонард Н. Основи. Графічний дизайн 03. Генерування ідей. Київ : ArtHuss, 2019. 192 с.
5. Емброуз Г., Оно-Біллсон Н. Основи. Графічний дизайн 01. Підхід і мова. Київ : ArtHuss, 2019. 192 с.
6. Іваненко Т. Шрифтовий дизайн : основи. Харків : ХДАДМ, 2019. 144 с.
7. Пасічник О. Г., Пасічник О. В., Стеценко І. В. Основи веб-дизайну : навч. посіб. Київ : Видавнича група BHV, 2009. 336 с.
8. Thornsby J. Android UI Design: plan, design, and build engaging user interfaces for your Android applications. Birmingham ; Mumbai : Packt Publishing, 2016. 364 p.
9. UX Design Process Best Practices Documentation for Moving Design Forward. UXPin Inc., 2015. 110 p.
10. Puerta A. R. Supporting User–Centred Design of Adaptive User Interfaces Via Interface Models. First Annual Workshop On Real–Time Intelligent User Interfaces For Decision Support And Information Visualization. San–Francisco, 1998. 10 p.

ІНФОРМАЦІЙНІ ДЖЕРЕЛА:

1. Базовий курс UX | medium URL: https://medium.com/ux-crash-course/ux-16ff8b55235f.
2. Введення в UX дизайн від Технологічного інституту Джорджії | coursera URL: https://www.coursera.org/learn/user-experience-design
3. Дон Норман: Дизайн звичних речей | udacity URL: https://www.udacity.com/course/intro-to-the-design-of-everyday-things--design101

4. Інтерактивний курс школи програмування і веб-дизайну «Створи веб-сайт» | Codecademy URL: https://www.codecademy.com/learn/make-a-website
5. Курс по адаптивному веб-дизайну від Лондонського університету "Адаптивний веб-дизайн" | Coursera URL: https://www.coursera.org/learn/responsive-web-design
6. Курс по веб-дизайну і створення мобільних додатків на Android "UX дизайн для мобільних розробників" | Udacity URL: https://www.udacity.com/course/ux-design-for-mobile-developers--ud849

Навчальне видання
(українською мовою)

Чемерис Ганна Юріївна

UX/UI ДИЗАЙН

Навчальний посібник
для здобувачів ступеня вищої освіти бакалавра
спеціальності «Дизайн»
освітньо-професійної програми «Графічний дизайн»

Рецензент Л. Ю. Москальова
Відповідальний за випуск Л. М. Коваль
Коректор Г. В. Брянцева

Друкується в авторській редакції

290