

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

МАТЕМАТИЧНИЙ ФАКУЛЬТЕТ

Кафедра програмної інженерії

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

**на тему: «РОЗРОБКА MARKETING СИСТЕМИ
ДЛЯ СФЕРИ НЕРУХОМОСТІ»**

Виконав: студент 4 курсу, групи 6.1212-пі-3
спеціальності 121 інженерія програмного забезпечення
(шифр і назва спеціальності)
освітньої програми програмна інженерія
(назва освітньої програми)

І.Г. Гузенко

(ініціали та прізвище)

Керівник доцент кафедри програмної інженерії,
к.ф.-м.н. Мильцев О.М.
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Рецензент професор кафедри фундаментальної та прикладної
математики, професор, д.т.н. Гребенюк С.М.
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет математичний

Кафедра програмної інженерії

Рівень вищої освіти бакалавр

Спеціальність 121 інженерія програмного забезпечення

(шифр і назва)

Освітня програма програмна інженерія

ЗАТВЕРДЖУЮ

Завідувач кафедри програмної
інженерії, к.ф.-м.н., доцент

_____ Лісняк А.О.
(підпис)

“ _____ ” _____ 2025 р.

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТОВІ

Гузенку Іллі Геннадійовичу

(прізвище, ім'я та по-батькові)

1. Тема роботи Розробка Marketing системи для сфери нерухомості

керівник роботи Мильцев Олександр Михайлович, к.ф.-м.н.

(прізвище, ім'я та по-батькові, науковий ступінь, вчене звання)

затверджені наказом ЗНУ від « 02 » грудня 2025 року № 1741-с

2. Строк подання студентом роботи 01.06.2026 р.

3. Вихідні дані до роботи 1. Постановка задачі.

2. Перелік літератури.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Аналіз предметної області.

2. Проектування маркетингової системи.

3. Реалізація маркетингової системи.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) _____

презентація за темою доповіді

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання 09.12.2025 р.**КАЛЕНДАРНИЙ ПЛАН**

№	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Робота з літературою.	30.01.2026	
2.	Вивчення загальної теорії розробки багато-платформових застосунків.	13.02.2026	
3.	Розробка структури бази даних.	27.02.2026	
4.	Написання програмного застосунку та його налагодження.	20.04.2026	
5.	Оформлення та нормоконтроль кваліфікаційної роботи.	25.05.2026	
6.	Захист кваліфікаційної роботи.	18.06.2026	

Студент

(підпис)

І.Г. Гузенко

(ініціали та прізвище)

Керівник роботи

(підпис)

О.М. Мильцев

(ініціали та прізвище)**Нормоконтроль пройдено**

Нормоконтролер

(підпис)

А.В. Столярова

(ініціали та прізвище)

РЕФЕРАТ

Кваліфікаційна робота бакалавра «Розробка Marketing системи для сфери нерухомості»: 45 с., 15 рис., 2 табл. 27 джерел.

БАЗА ДАНИХ, ВЕБЗАСТОСУНОК, ЛІД, МАРКЕТИНГОВА СИСТЕМА, НЕРУХОМІСТЬ, CRM, GO, REACT.

Об'єкт дослідження – маркетингові інформаційні системи у сфері нерухомості.

Мета роботи: створення маркетингової інформаційної системи, яка забезпечує управління об'єктами нерухомості, обробку лідів, підтримку маркетингових кампаній та аналіз ефективності рекламної діяльності.

Методи дослідження – методи системного аналізу предметної області, методи збору та аналізу вимог, методи об'єктно-орієнтованого проєктування та моделювання, методи конструювання й тестування програмного забезпечення.

Кваліфікаційна робота присвячена розробці маркетингової системи для сфери нерухомості. Було розглянуто особливості сучасного ринку нерухомості, проаналізовано основні підходи до цифрового маркетингу та досліджено існуючі програмні рішення, що використовуються для автоматизації взаємодії з клієнтами. У роботі використано сучасні технології веброзробки, зокрема REST API, реляційні бази даних та клієнт-серверну архітектуру. Запропонована система дозволяє автоматизувати процеси роботи агентства нерухомості, зменшити час обробки заявок та підвищити ефективність маркетингових кампаній. Практична цінність роботи полягає у можливості використання розробленої системи в діяльності агентств нерухомості для оптимізації процесів продажу та взаємодії з клієнтами.

SUMMARY

Bachelor's qualifying paper "Development of a Real Estate Marketing System": 45 pages, 15 figures, 2 tables, 27 references.

DATABASE, WEB APPLICATION, LEAD, MARKETING SYSTEM, REAL ESTATE, CRM, GO, REACT.

The object of the study is marketing information systems in the real estate sector.

The aim of the study is to create a marketing information system that enables the management of real estate properties, lead processing, support for marketing campaigns, and analysis of the effectiveness of advertising activities.

Research methods include methods of system analysis of the subject area, methods of requirements gathering and analysis, methods of object-oriented design and modeling, as well as methods of software construction and testing.

This thesis is devoted to the development of a marketing system for the real estate sector. The paper examines the characteristics of the modern real estate market, analyzes the main approaches to digital marketing, and investigates existing software solutions used to automate customer interactions. The thesis utilizes modern web development technologies, including REST API, relational databases, and client-server architecture. The proposed system allows for the automation of real estate agency workflows, reduces the time required to process requests, and increases the effectiveness of marketing campaigns. The practical value of this work lies in the ability to use the developed system in real estate agencies to optimize sales processes and customer interactions.

ЗМІСТ

Завдання на кваліфікаційну роботу.....	2
Реферат	4
Summary	5
Вступ.....	7
1 Аналіз предметної області.....	9
1.1 Особливості ринку нерухомості.....	9
1.2 Маркетинг у сфері нерухомості.....	9
1.3 Аналіз існуючих рішень	10
1.4 Постановка задачі	13
2 Проєктування маркетингової системи	15
2.1 Загальна архітектура системи.....	15
2.2 Функціональні вимоги до системи.....	15
2.3 Нефункціональні вимоги.....	16
2.4 UML-моделювання системи	17
2.5 Проєктування бази даних	19
2.6 Алгоритми роботи системи.....	25
3 Реалізація маркетингової системи.....	30
3.1 Вибір технологій розробки	30
3.2 Архітектура програмного забезпечення	32
3.3 Реалізація серверної частини	34
3.4 Реалізація клієнтської частини	39
3.5 Тестування системи	41
Висновки	42
Перелік посилань.....	44

ВСТУП

Сучасний ринок нерухомості характеризується високим рівнем конкуренції, динамічністю та активним впровадженням цифрових технологій. У таких умовах ефективність діяльності агентств нерухомості значною мірою залежить від використання сучасних маркетингових інструментів, які дозволяють залучати клієнтів, аналізувати їх поведінку та оптимізувати процес продажу об'єктів.

Традиційні методи маркетингу поступово втрачають ефективність, поступаючись місцем цифровим каналам просування, таким як соціальні мережі, пошукова реклама, e-mail-маркетинг та спеціалізовані платформи нерухомості. Водночас виникає потреба у створенні інтегрованих маркетингових систем, які об'єднують функції управління об'єктами нерухомості, обробки клієнтських запитів та аналізу ефективності рекламних кампаній.

Актуальність теми полягає у необхідності розробки сучасної маркетингової інформаційної системи, яка дозволить автоматизувати процеси просування нерухомості та підвищити ефективність взаємодії з клієнтами.

Метою роботи є розробка маркетингової системи для сфери нерухомості, яка забезпечує управління об'єктами, обробку потенційних клієнтів та аналіз ефективності маркетингових кампаній.

Для досягнення поставленої мети необхідно вирішити такі **завдання**:

- проаналізувати ринок нерухомості та його особливості;
- дослідити сучасні маркетингові підходи у сфері нерухомості;
- провести аналіз існуючих інформаційних систем;
- сформулювати вимоги до системи;
- розробити архітектуру та модель даних;
- реалізувати програмний продукт;
- оцінити ефективність запропонованого рішення.

Об'єктом дослідження є процеси маркетингу у сфері нерухомості.

Предмет дослідження – методи та засоби розробки маркетингових інформаційних систем.

Практична цінність роботи полягає у створенні програмного продукту, який може бути використаний агентствами нерухомості для автоматизації маркетингової діяльності.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Особливості ринку нерухомості

Ринок нерухомості є складною економічною системою, яка включає операції купівлі, продажу та оренди об'єктів житлової та комерційної нерухомості. Його характерними особливостями є висока вартість об'єктів, тривалий цикл прийняття рішень клієнтами та значний вплив зовнішніх факторів, таких, наприклад, як економічна ситуація або законодавство.

Однією з ключових характеристик ринку є поділ на первинний та вторинний сегменти. Первинний ринок включає новобудови від забудовників, тоді як вторинний – вже існуючі об'єкти. Кожен із сегментів має свої особливості маркетингового просування.

Також важливою особливістю є індивідуальність кожного об'єкта нерухомості, що ускладнює стандартизацію процесів продажу та потребує персоналізованого підходу до клієнтів.

1.2 Маркетинг у сфері нерухомості

Маркетингом (від англ. Marketing – збут) називається діяльність, спрямована на вивчення ринку, задоволення потреб споживачів та отримання прибутку для бізнесу [1]. У більш широкому сенсі маркетинг займається визначенням та задоволенням людських і суспільних потреб.

Маркетинг у сфері нерухомості спрямований на залучення лідів (він англ. Lead – потенційний клієнт, який проявив інтерес до продуктів або послуг компанії) або орендарів та ефективне представлення об'єктів.

Сучасний маркетинг у цій галузі базується на використанні цифрових каналів:

- пошукова реклама;
- соціальні мережі;
- спеціалізовані платформи;
- e-mail-маркетинг.

Особливу роль відіграє **лідогенерація**, тобто процес отримання контактних даних потенційних клієнтів. Для цього використовуються:

- landing pages;
- форми зворотного зв'язку;
- онлайн-чати.

Ефективність маркетингу значною мірою залежить від швидкості обробки лідів та якості взаємодії з клієнтами.

1.3 Аналіз існуючих рішень

На сучасному ринку представлено велику кількість інформаційних систем, що підтримують маркетингову діяльність у сфері нерухомості.

До основних типів систем належать:

- CRM-системи;
- ERP-системи з маркетинговими модулями;
- платформи розміщення оголошень;
- аналітичні системи;
- CDP-системи.

CRM-системи (від англ. Customer Relationship Management – управління взаємовідносинами з клієнтами) дозволяють зберігати інформацію про клієнтів, відслідковувати взаємодії та автоматизувати комунікацію. Однак вони часто не мають спеціалізованого функціоналу для роботи з нерухомістю. Розглянемо детальніше найбільш відомі CRM-системи.

Salesforce – одна з найбільш популярних CRM-систем, яка має розширену функціональність саме для управління продажами, маркетингом та

роботи клієнтів. Ця програма має інтуїтивно-зрозумілий простий інтерфейс, потужні засоби для автоматизації підготовки звітності та здійснення аналітики, крім того вона підтримує можливість інтеграції з іншими інформаційними системами [2].

HubSpot – є CRM-системою з відкритим початковим кодом, яка надає реалізує управління продажами, маркетинг та роботу з клієнтами. Вона також має інтуїтивно зрозумілий і простий інтерфейс та засоби для підготовки звітності, аналітики, управління контактами тощо. Крім того HubSpot підтримує можливість інтеграції з іншим відповідним інструментарієм [3].

Zoho CRM – це хмарна CRM-система, що автоматизує продажі, маркетинг та роботу з клієнтами. Вона пропонує великий набір функцій, таких як управління потенційними клієнтами, підтримку роботи з електронною поштою, календарями, автоматизацію побудови звітів та аналітики тощо [4].

ERP-системи (від англ. Enterprise Resource Planning – планування ресурсів підприємства) – це комплексні програмні рішення для управління всіма ресурсами підприємства, що дозволяють автоматизувати бізнес-процеси та підвищити ефективність роботи компанії [5].

Серед великої кількості таких систем можна виділити наступні: SAP ERP, Microsoft Dynamics 365 та Odoo, де на високому рівні було реалізовано інтеграцію маркетингу з фінансами, виробництвом і логістикою.

Особливо цікавою є саме продукт від компанії Microsoft, оскільки він надає рішення для роботи з клієнтами, продажами та маркетингом, які інтегровані з іншими продуктами Microsoft, а саме Outlook і Office 365, що дає можливість зручно працювати з даними і використовувати поширені інструменти.

Платформи розміщення оголошень забезпечують широкий охопит аудиторії, але не дозволяють ефективно управляти маркетинговими кампаніями. Серед найбільш відомих зарубіжних платформ можна відзначити:

- Google Ads – глобальна рекламна мережа, охоплює пошук, YouTube, партнерські сайти [6];

- Facebook Ads / Meta Ads Manager – реклама у Facebook, Instagram, Messenger [7];
- Craigslist – популярна платформа оголошень у США та світі [8];
- eBay Classifieds – оголошення про продаж товарів і послуг [9].

Серед вітчизняних платформ можна виділити:

- OLX Україна – найбільший маркетплейс оголошень (товари, послуги, нерухомість, авто) [10];
- Prom.ua – маркетплейс для бізнесу, орієнтований на продаж товарів онлайн [11];
- Ria.com (AUTO.RIA, DOM.RIA, RIA Travel) – спеціалізовані платформи для авто, нерухомості та туризму [12];
- Kabanchik.ua – сервіс замовлення послуг (ремонт, доставка, навчання) [13].

Крім вищезазначених рішень у маркетингу також широко використовуються аналітичні системи – цифрові платформи та інструменти, які збирають, обробляють і візуалізують дані про клієнтів, ринок та ефективність кампаній.

Найпопулярнішими серед них є: Google Analytics [14], Adobe Marketing Cloud [15], Power BI [16], а також спеціалізовані CDP-платформи (від англ. Customer Data Platform – платформа даних клієнтів), такі як Segment [17], BlueConic [18] та інші [19]. Вони допомагають бізнесу приймати стратегічні рішення на основі фактів, а не інтуїції.

У таблиці 1.1 наведено порівняльні характеристики популярних маркетингових систем, які широко використовуються у наш час.

Однак, розробка нових маркетингових інформаційних та аналітичних систем залишається актуальною в наш час, навіть попри існування великої кількості вже готових рішень. Це пояснюється наступними факторами:

- швидкими змінами ринку, які постійно спостерігаються у останні роки;
- появою і активним впровадженням нових інформаційних технологій;

- необхідністю локалізації наявних платформ;
- прагненням бізнесу до індивідуальних рішень.

Таблиця 1.1 – Порівняння популярних маркетингових систем

Категорія	Приклади	Основні функції	Переваги
CRM	Salesforce, HubSpot	Управління клієнтами, автоматизація продажів	Глибока персоналізація, інтеграція з іншими сервісами
BI	SAP BI, Power BI	Аналітика, прогнозування, візуалізація	Потужні інструменти для аналізу великих даних
ERP	Dynamics 365, Odoo	Інтеграція маркетингу з бізнес-процесами	Єдина система для всієї компанії
Маркетингова аналітика	Google Analytics, Adobe Marketing Cloud	Вебаналітика, управління кампаніями	Простота використання, інтеграція з рекламою
CDP	Segment, BlueConic	Централізовані дані про клієнтів	Персоналізовані стратегії маркетингу

Таким чином, розробка нових маркетингових систем є не дублюванням вже наявних, а відповіддю на нові виклики та можливості.

1.4 Постановка задачі

На основі проведеного аналізу сформульовано задачу розробки маркетингової системи для сфери нерухомості.

Система повинна забезпечувати:

- зберігання та управління інформацією про об'єкти нерухомості;
- обробку запитів клієнтів;

- ведення бази лідів;
- ведення бази рекламних кампаній і т.п.

Основною метою системи є підвищення ефективності маркетингової діяльності та оптимізація процесу взаємодії з клієнтами.

2 ПРОЄКТУВАННЯ МАРКЕТИНГОВОЇ СИСТЕМИ

2.1 Загальна архітектура системи

Розроблювана маркетингова система для сфери нерухомості реалізується як багаторівнева веборієнтована інформаційна система, що забезпечує взаємодію між користувачами та серверною частиною через мережу Інтернет.

Архітектура системи базується на клієнт-серверній моделі та включає такі основні компоненти:

- клієнтський рівень (frontend);
- серверний рівень (backend);
- рівень зберігання даних (database).

Клієнтський рівень відповідає за відображення інформації та взаємодію з користувачем. Серверний рівень реалізує бізнес-логіку системи, обробку запитів та взаємодію з базою даних.

Взаємодія між клієнтом і сервером може здійснюватися різними способами, наприклад, з використанням REST API [20], що забезпечує гнучкість та масштабованість системи.

2.2 Функціональні вимоги до системи

Функціональні вимоги визначають основні можливості системи. Розглянемо, що саме повинна забезпечувати система.

Управління об'єктами нерухомості:

- додавання, редагування та видалення об'єктів (користувачів, лідів, об'єктів нерухомості тощо);
- зберігання характеристик об'єктів нерухомості (площа, адреса, ціна, опис і т. п.);

- додавання фотографій або вебпосилань на них.

Управління клієнтами (лідами):

- реєстрація клієнтів;
- зберігання контактної інформації;
- обробка запитів;
- відслідковування статусу лідів.

Маркетингові кампанії:

- створення рекламних кампаній;
- прив'язка кампаній до об'єктів;
- аналіз ефективності.

Аналітику та звітність:

- перегляд статистики;
- аналіз конверсій;
- формування звітів.

2.3 Нефункціональні вимоги

До нефункціональних вимог належать:

- продуктивність: реальна бізнес-система повинна вміти одночасно обробляти велику кількість запитів;
- масштабованість: можливість розширення інформаційної системи при виникненні такої потреби;
- надійність: інформаційна система повинна забезпечувати надійне збереження даних (у т.ч. й зберігати комерційну таємницю й приватність клієнтів від несанкціонованого доступу);
- безпека: захист персональних даних клієнтів;
- зручність використання: інтуїтивно зрозумілий та дружній до користувача інтерфейс.

2.4 UML-моделювання системи

Для формалізації структури інформаційних систем зазвичай використовуються UML-діаграми [21].

Use Case діаграма описує взаємодію користувачів із системою:

- адміністратор;
- менеджер;
- клієнт.

Основні сценарії:

- додавання об'єкта;
- обробка ліда;
- перегляд статистики.

Діаграма класів. Основні класи системи:

- Property (об'єкт нерухомості);
- User (користувач);
- Lead (лід);
- Campaign (кампанія).

Діаграма послідовності описує послідовність кроків, які реалізують процес обробки заявки клієнта:

- а) клієнт надсилає запит;
- б) система зберігає лід;
- в) менеджер обробляє заявку.

Інформаційну маркетингову систему, що розробляється, можна описати UML-діаграмою, яка складається з наступних елементів:

- *Use Case діаграма* з акторами (адміністратор, менеджер, клієнт) та сценаріями (додавання об'єкта, обробка ліда, перегляд статистики тощо);
- *діаграма класів* із основними класами системи (Property, User, Lead, Campaign) та їхніми атрибутами й зв'язками;
- *діаграма послідовності* для процесу обробки заявки клієнта (запит → збереження ліда → обробка менеджером) (див. рис. 2.1).

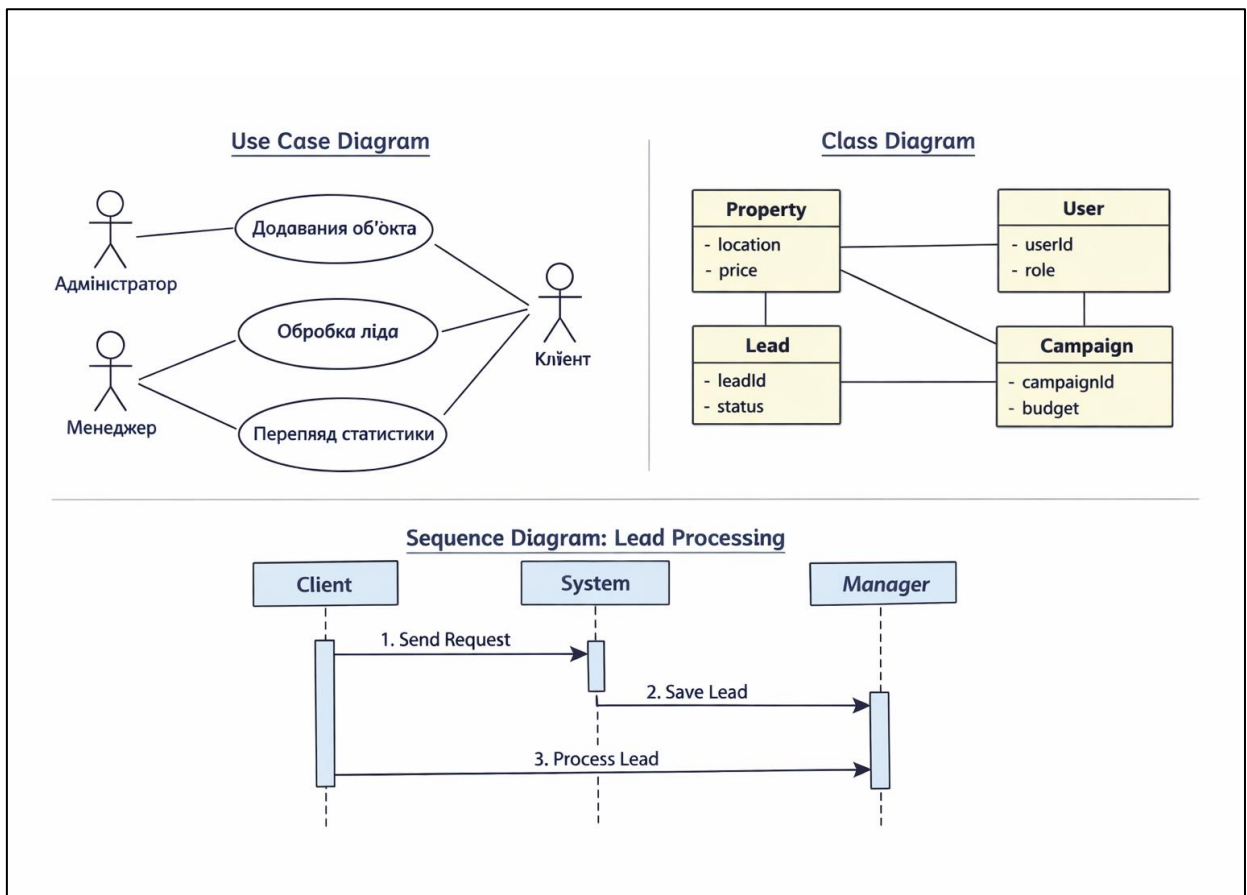


Рисунок 2.1 – UML-діаграма маркетингової системи для сфери нерухомості

Тут Use Case діаграма показує взаємодію користувачів із системою. Акторами є: адміністратор, менеджер, клієнт.

Адміністратор – має повний доступ до системи, додає нові об'єкти нерухомості, переглядає статистику, керує користувачами.

Менеджер – обробляє ліди, може додавати об'єкти, переглядати статистику кампаній і т.п.

Клієнт – надсилає запити (заявки), взаємодіє із системою через відповідну форму або сайт.

Система повинна підтримувати наступні сценарії (Use Cases):

- *додавання об'єкта* – створення нового запису про нерухомість;
- *обробка ліда* – прийом і опрацювання заявки клієнта;
- *перегляд статистики* – аналіз ефективності кампаній, кількості лідів, активності користувачів.

Діаграма класів описує основні класи системи (див. табл. 2.1).

Таблиця 2.1 – Основні класи маркетингової системи нерухомості

Клас	Головні атрибути	Зв'язки
Property	location, price, type	Пов'язаний із Lead (об'єкт, на який подано заявку)
User	userId, role, name	Може бути менеджером або адміністратором; обробляє Lead
Lead	leadId, status, date	Пов'язаний із Property та User
Campaign	campaignId, budget, startDate	Може містити кілька Lead, пов'язана з менеджером

Типи зв'язку між відношеннями:

- *User – Lead*: один менеджер може обробляти багато лідів;
- *Lead – Property*: кожен лід стосується одного об'єкта нерухомості;
- *Campaign – Lead*: кампанія може генерувати кілька лідів.

Діаграма послідовності (Sequence Diagram)

Процес обробки заявки клієнта складається з наступних кроків:

- а) клієнт надсилає запит через форму, після чого повідомлення потрапляє до системи;
- б) система створює новий об'єкт типу Lead і зберігає його в базі даних;
- в) менеджер отримує сповіщення про новий лід, переглядає його дані та змінює статус (наприклад, «в роботі» або «закрито»).

У результаті заявка клієнта проходить повний цикл – від створення до обробки менеджером.

2.5 Проєктування бази даних

База даних є одним із ключових компонентів маркетингової інформаційної системи для сфери нерухомості, оскільки саме вона забезпечує централізоване зберігання, обробку та взаємозв'язок усіх даних, необхідних

для функціонування системи. При проєктуванні бази даних було використано реляційну модель, що забезпечує цілісність даних, можливість встановлення зв'язків між сутностями та ефективне виконання запитів.

Основною метою створення бази даних є забезпечення автоматизації процесів управління об'єктами нерухомості, обробки заявок клієнтів, ведення маркетингових кампаній та контролю взаємодії з потенційними покупцями.

Структура бази даних складається з таких основних таблиць:

- users;
- properties;
- leads;
- campaigns;
- campaign_leads;
- interactions;
- deals.

Кожна таблиця виконує окрему функцію в системі та пов'язана з іншими таблицями за допомогою зовнішніх ключів.

Таблиця *users* призначена для зберігання інформації про користувачів системи. До користувачів належать адміністратори, менеджери з продажу та інші співробітники агентства нерухомості, які працюють із системою.

Основними полями таблиці є:

- ідентифікатор користувача;
- ім'я;
- електронна пошта;
- номер телефону;
- роль користувача;
- дата створення запису.

Ідентифікатор користувача є первинним ключем та використовується для зв'язку з іншими таблицями. Поле «ролі» визначає права доступу користувача до функціональних можливостей системи. Наприклад, адміністратор має повний доступ до всіх модулів, тоді як менеджер працює лише з об'єктами нерухомості та лідами.

Таблиця *users* використовується в інших таблицях для визначення відповідальних осіб за створення об'єктів, обробку заявок та управління маркетинговими кампаніями.

Таблиця *properties* містить інформацію про об'єкти нерухомості, які додаються до системи для подальшого продажу або оренди. Основними атрибутами цієї таблиці є:

- назва об'єкта;
- опис;
- ціна;
- площа;
- адреса;
- тип нерухомості;
- статус;
- дата створення.

Кожний об'єкт нерухомості має унікальний ідентифікатор, який використовується як первинний ключ.

Поле типу нерухомості дозволяє класифікувати об'єкти, наприклад:

- квартира;
- будинок;
- офісне приміщення;
- комерційна нерухомість.

Поле «статус» використовується для визначення поточного стану об'єкта, наприклад:

- доступний;
- проданий;
- заброньований.

Таблиця *properties* має зв'язок із таблицею *users*, що дозволяє визначити користувача, який створив або редагував запис про об'єкт.

Таблиця *leads* є однією з найважливіших у маркетинговій системі, оскільки вона містить інформацію про потенційних клієнтів, які проявили інтерес до об'єктів нерухомості.

У таблиці зберігаються:

- ім'я клієнта;
- контактні дані;
- джерело надходження ліда;
- статус заявки;
- посилання на об'єкт нерухомості;
- відповідальний менеджер.

Лід є потенційним клієнтом, який ще не здійснив покупку, але вже взаємодіяв із системою, наприклад залишив заявку на сайті або зв'язався з агентством через рекламне оголошення.

Поле «джерело» ліда використовується для аналізу ефективності маркетингових каналів. Джерелами можуть бути:

- пошукова реклама;
- соціальні мережі;
- e-mail-розсилки;
- зовнішні рекламні платформи.

Статус ліда дозволяє контролювати процес роботи з клієнтом:

- новий;
- у процесі обробки;
- завершений;
- успішний продаж.

Таблиця leads пов'язана з таблицею properties, оскільки кожний лід формується відносно конкретного об'єкта нерухомості.

Таблиця campaigns використовується для зберігання інформації про маркетингові кампанії, які проводяться агентством нерухомості. Основними полями тут є:

- назва кампанії;
- бюджет;
- дата початку;
- дата завершення;
- відповідальний користувач.

Маркетингові кампанії дозволяють здійснювати просування об'єктів нерухомості через різні канали реклами та аналізувати ефективність рекламних витрат. Наприклад, кампанія може бути спрямована на просування новобудови через соціальні мережі або контекстну рекламу.

Таблиця *campaign_leads* реалізує зв'язок типу «багато-до-багатьох» між маркетинговими кампаніями та лідами. Необхідність створення окремої таблиці пояснюється тим, що одна кампанія може генерувати багато лідів, а також тим, що один лід може бути пов'язаний із кількома маркетинговими активностями.

Таблиця містить наступні поля:

- ідентифікатор кампанії;
- ідентифікатор ліда.

Використання цієї таблиці дозволяє проводити аналіз результативності рекламних кампаній та визначати найбільш ефективні маркетингові канали.

Таблиця *interactions* призначена для зберігання історії взаємодії менеджерів із потенційними клієнтами. У таблиці фіксуються:

- телефонні дзвінки;
- e-mail-повідомлення;
- зустрічі;
- коментарі менеджерів.

Основними полями є:

- тип взаємодії;
- дата;
- примітка;
- відповідальний користувач;
- посилання на лід.

Наявність історії взаємодій дозволяє контролювати процес комунікації з клієнтом та підвищувати якість обслуговування.

Таблиця *deals* містить інформацію про завершені або активні угоди купівлі-продажу нерухомості.

У таблиці зберігаються:

- об'єкт нерухомості;
- клієнт;
- сума угоди;
- статус;
- дата завершення.

Після успішного завершення роботи з лідом створюється запис у таблиці deals, що дозволяє відокремити потенційних клієнтів від реальних угод.

Статус угоди може мати такі значення:

- активна;
- завершена;
- скасована.

Таблиця deals використовується для фінансової аналітики та оцінки ефективності діяльності агентства нерухомості.

Загальна структура взаємозв'язків – усі таблиці бази даних пов'язані між собою через систему первинних та зовнішніх ключів. Основні зв'язки реалізовано за принципами:

- «один-до-багатьох»;
- «багато-до-багатьох».

Така структура забезпечує:

- цілісність даних;
- уникнення дублювання інформації;
- можливість виконання складних аналітичних запитів.

Таким чином, запропонована база даних побудована за реляційною моделлю та включає сутності користувачів, об'єктів нерухомості, лідів, маркетингових кампаній та взаємодій.

Основними зв'язками є зв'язки типу «один-до-багатьох» між користувачами та об'єктами, а також «багато-до-багатьох» між кампаніями та лідами. Її ER-діаграма може бути зображена так, як представлено на рисунку 2.2.

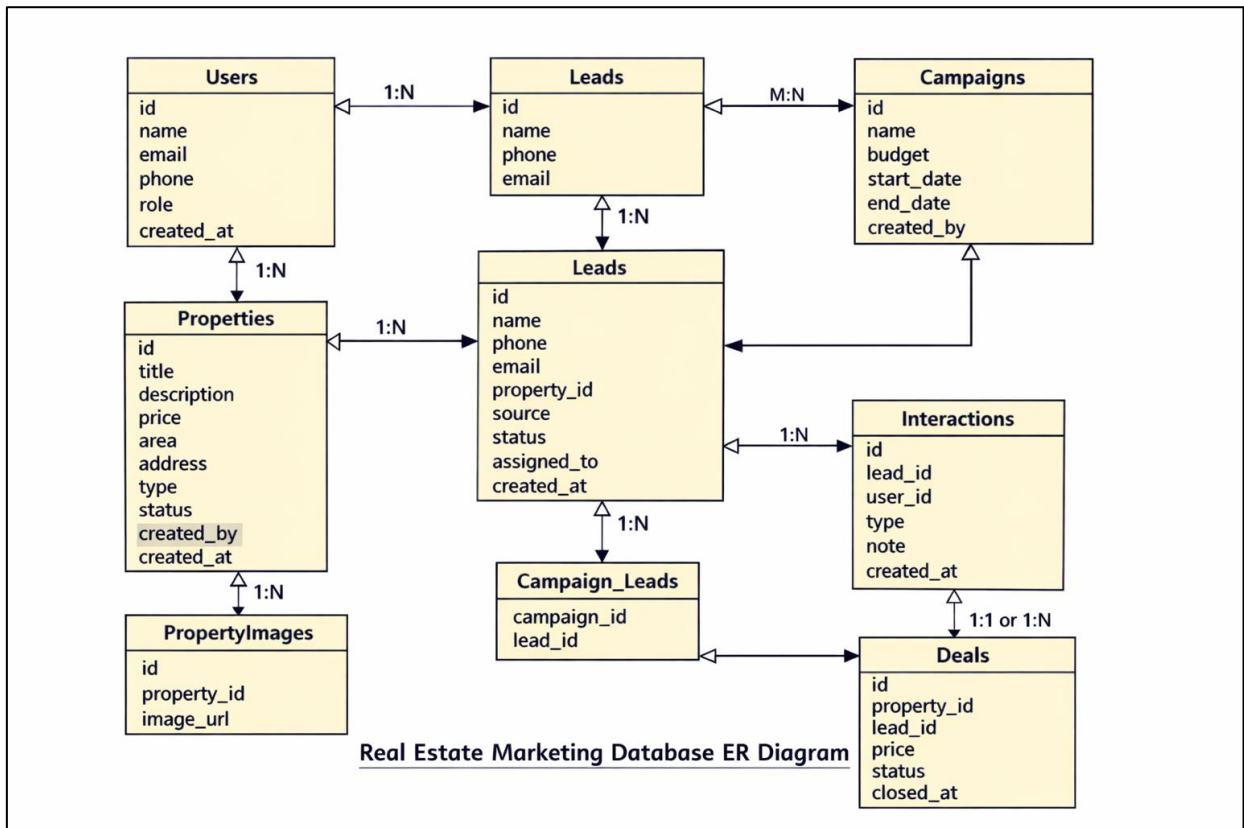


Рисунок 2.2 – ER-діаграма бази даних агентства нерухомості

Запропонована структура бази даних інформаційної системи продажу об'єктів нерухомості є масштабованою та може бути розширена додатковими таблицями й функціональними можливостями залежно від потреб системи. Також важливо зазначити, що запропонована структура не залежить від особливостей конкретної СУБД, а може бути реалізована у різних системах, наприклад, MySQL [22] або PostgreSQL [23].

2.6 Алгоритми роботи системи

Одним із ключових компонентів маркетингової системи для сфери нерухомості є алгоритм обробки лідів. Ефективність роботи маркетингового агентства значною мірою залежить від швидкості реагування на заявки клієнтів, правильності розподілу звернень між менеджерами та контролю процесу взаємодії з потенційними покупцями.

Як вже зазначалося, лідом у системі вважається потенційний клієнт, який проявив інтерес до певного об'єкта нерухомості та залишив свої контактні дані через вебсайт, рекламне оголошення або інший канал комунікації.

Алгоритм обробки ліда складається з кількох **основних етапів**.

Отримання заявки від потенційного клієнта. Заявка може надходити з різних джерел:

- вебформи на сайті агентства;
- соціальних мереж;
- контекстної реклами;
- e-mail-розсилки;
- телефонного дзвінка.

На цьому етапі система отримує основну інформацію про клієнта:

- ім'я;
- номер телефону;
- електронну пошту;
- об'єкт нерухомості, який зацікавив клієнта;
- джерело надходження заявки.

Після заповнення форми дані передаються на сервер, наприклад, через REST API та проходять первинну перевірку коректності. Зокрема перевіряється:

- наявність обов'язкових полів;
- коректність e-mail-адреси;
- правильність формату номера телефону.

У разі успішного проходження перевірки система переходить до наступного етапу.

Збереження у базі даних. Після отримання та перевірки інформації система створює новий запис у таблиці leads. Для кожного ліда автоматично формуються:

- унікальний ідентифікатор;
- дата створення;

- початковий статус;
- прив'язка до конкретного об'єкта нерухомості.

На цьому етапі також фіксується джерело надходження ліда, що дозволяє у подальшому аналізувати ефективність маркетингових каналів.

Після збереження інформації система може автоматично надсилати сповіщення менеджеру або адміністратору про появу нового клієнта.

Призначення менеджера відповідального за роботу з клієнтом. Призначення може здійснюватися як вручну адміністратором, так і автоматично системою. У разі автоматичного розподілу можуть використовуватись різні критерії:

- поточне навантаження менеджерів;
- спеціалізація;
- регіон роботи;
- тип нерухомості.

Після призначення менеджера у таблиці leads оновлюється поле відповідального користувача. Менеджер отримує доступ до контактної інформації клієнта та може розпочати комунікацію:

- телефонний дзвінок;
- e-mail;
- організація зустрічі;
- демонстрація об'єкта.

Усі подальші взаємодії з клієнтом можуть фіксуватись у таблиці interactions, що дозволяє вести історію комунікацій.

Зміна статусу ліда. Під час роботи з клієнтом менеджер змінює статус ліда залежно від поточного етапу обробки. Основними статусами можуть бути:

- новий;
- у роботі;
- очікування відповіді;
- успішно завершений;
- скасований.

Зміна статусу дозволяє контролювати стан кожної заявки та аналізувати ефективність роботи менеджерів.

У випадку успішного продажу на основі ліда може бути створений запис у таблиці deals, що означає перехід потенційного клієнта до категорії реальних угод.

Таким чином, алгоритм обробки лідів забезпечує централізоване управління взаємодією з клієнтами та дозволяє оптимізувати процес продажу нерухомості.

Для оцінки результативності маркетингової системи використовується **алгоритм аналізу ефективності** рекламних кампаній та роботи з лідами. Основною метою алгоритму є визначення:

- кількості отриманих лідів;
- ефективності рекламних каналів;
- рівня конверсії;
- результативності роботи менеджерів.

Алгоритм складається з кількох **основних етапів**.

Підрахунок кількості лідів. На першому етапі система виконує підрахунок кількості лідів, отриманих за певний період часу.

Аналіз може проводитись як за окремими днями, тижнями або місяцями, так і за окремими маркетинговими кампаніями. Для цього використовуються дані з таблиці leads.

Система може формувати статистику:

- загальної кількості заявок;
- кількості нових лідів;
- кількості успішно оброблених лідів;
- кількості відхилених заявок.

Крім того, аналізується джерело надходження кожного ліда, що дозволяє визначити найбільш ефективні рекламні канали. Наприклад:

- Google Ads;
- Facebook;

- Instagram;
- органічний пошук;
- e-mail-маркетинг.

Отримані результати можуть використовуватись для оптимізації рекламного бюджету.

Розрахунок конверсії. Другим важливим етапом є розрахунок конверсії, яка показує співвідношення між кількістю отриманих лідів та кількістю успішно завершених угод.

Формула розрахунку конверсії має вигляд:

$$Conversion = \frac{Deals}{Leads} \times 100\%,$$

де *Deals* – кількість успішних угод;

Leads – загальна кількість лідів.

Наприклад, якщо за місяць було отримано 100 лідів і укладено 12 угод, то конверсія становитиме 12%.

Розрахунок конверсії дозволяє:

- оцінювати ефективність маркетингових кампаній;
- аналізувати якість роботи менеджерів;
- визначати проблемні етапи взаємодії з клієнтами.

Також система може обчислювати додаткові показники:

- середню вартість ліда;
- вартість залучення клієнта;
- рентабельність рекламної кампанії.

Результати аналізу можуть відображатись у вигляді таблиць, графіків та аналітичних звітів, що значно спрощує прийняття управлінських рішень.

Таким чином, алгоритм оцінки ефективності забезпечує можливість контролю маркетингової діяльності агентства нерухомості та дозволяє підвищувати ефективність продажів шляхом оптимізації рекламних процесів.

3 РЕАЛІЗАЦІЯ МАРКЕТИНГОВОЇ СИСТЕМИ

3.1 Вибір технологій розробки

Для реалізації маркетингової системи було обрано сучасні технології, що забезпечують високу продуктивність, масштабованість та зручність підтримки.

Серверна частина системи реалізована з використанням мови програмування Go [24]. Вибір цієї мови обумовлений наступними чинниками.

Висока продуктивність. Мова Go компілюється у машинний код, тому вебсервери на ньому працюють швидше, ніж інтерпретовані рішення, створені з використанням скриптових мов (наприклад, Python або Node.js). Його ефективна модель пам'яті та мінімальні накладні витрати забезпечують стабільну роботу навіть під великим навантаженням.

Вбудована підтримка багатопоточності. Завдяки вбудованим паралельним функціям (горутинам) і безпечним каналам обміну даними між потоками мова Go дозволяє легко реалізовувати паралельну обробку запитів. Це особливо важливо для вебсерверів, які повинні одночасно обслуговувати тисячі клієнтів.

Простота синтаксису та швидкість розробки. Код у Go лаконічний і зрозумілий. Розробники можуть швидко створювати REST API або мікросервіси без зайвої складності – і при цьому зберігати високу надійність.

Вбудовані інструменти. Go має потужну стандартну бібліотеку для роботи з HTTP, JSON, логуванням, тестуванням і профілюванням. Це зменшує залежність від сторонніх пакетів і спрощує розгортання готового продукту.

Легка масштабованість і розгортання. Один бінарний файл без зовнішніх залежностей – це дуже важлива перевага при використанні сучасної технології DevOps [25]. Go-сервери легко контейнеризуються із використанням Docker або Kubernetes і швидко запускаються на будь-якій платформі.

Таким чином, мова Go поєднує продуктивність, простоту та потужну підтримку паралельності, що робить її ідеальним вибором для створення сучасних вебсерверів, мікросервісів та хмарних API.

Клієнтська частина маркетингової системи реалізована з використанням JavaScript-фреймворку **React** [26], який дозволяє швидко створювати надійний і високоякісний інтерактивний інтерфейс користувача. Вибір React зумовлений його наступними основними **перевагами** для веброзробки.

Компонентна архітектура. React дозволяє розбивати інтерфейс на незалежні компоненти, які легко повторно використовувати. Це спрощує підтримку великих проєктів і забезпечує чисту структуру коду.

Висока продуктивність завдяки Virtual DOM. Замість прямої роботи з DOM браузера React використовує віртуальний DOM, що мінімізує кількість оновлень і значно прискорює рендеринг сторінок.

Односторонній потік даних. Дані передаються зверху вниз (parent → child), що робить логіку передбачуваною і спрощує налагодження навіть складних інтерфейсів.

Велика екосистема та спільнота. React має тисячі бібліотек і готових рішень – від маршрутизації (React Router) до управління станом (Redux, Zustand, Recoil). Це дозволяє швидко створювати масштабовані вебдодатки.

Підтримка TypeScript і SSR. React чудово інтегрується з TypeScript для типізації та з фреймворками на кшталт Next.js для серверного рендерингу, що покращує SEO та швидкість завантаження.

Таким чином, фреймворк React поєднує гнучкість, продуктивність і потужну екосистему, що робить його ідеальним вибором для створення інтерактивних, масштабованих і зручних вебінтерфейсів.

Для зберігання даних було використано реляційну систему керування базами даних **PostgreSQL** [23], яка забезпечує надійність та підтримку складних запитів. Її основними **переваги** є наступні чинники.

Відкритий код і безкоштовність. PostgreSQL є open-source, що дозволяє використовувати його без ліцензійних витрат і адаптувати під власні потреби.

Розширюваність. Можна створювати власні типи даних, функції, індекси та навіть мови програмування для збережених процедур. Це робить PostgreSQL надзвичайно гнучким.

Підтримка складних запитів. PostgreSQL має потужний SQL-движок із підтримкою CTE, window-функцій, JSON/JSONB, XML, масивів та геопросторових даних (PostGIS).

Надійність і транзакційність. Повна підтримка ACID, багаторівневі механізми блокувань, реплікація та резервне копіювання забезпечують стабільність роботи навіть у критично важливих системах.

Масштабованість і продуктивність. Завдяки індексам (B-Tree, GiST, GIN, BRIN), партиціюванню та оптимізованому планувальнику запитів PostgreSQL добре працює як із невеликими, так і з дуже великими базами.

Активна спільнота та екосистема. PostgreSQL має багату екосистему розширень (наприклад, TimescaleDB для time-series, PostGIS для геоданих), а також велику спільноту, яка постійно вдосконалює систему.

Отже, можна зробити наступний висновок: PostgreSQL – це універсальна, надійна й розширювана СУБД, яка підходить як для невеликих вебпроектів, так і для складних корпоративних систем із великими обсягами даних.

3.2 Архітектура програмного забезпечення

Програмна система реалізована за принципами багаторівневої архітектури, що включає:

- рівень представлення (frontend);
- рівень бізнес-логіки (backend);
- рівень доступу до даних (database layer).

Серверна частина побудована за модульним принципом і складається з таких компонентів:

- модуль управління користувачами;
- модуль роботи з об'єктами нерухомості;
- модуль обробки лідів;
- маркетинговий модуль.

Взаємодія між клієнтом і сервером реалізована через REST API з використанням формату JSON.

Загальна архітектура маркетингової системи для сфери нерухомості може бути представлена так, як показано на рисунку 3.1.

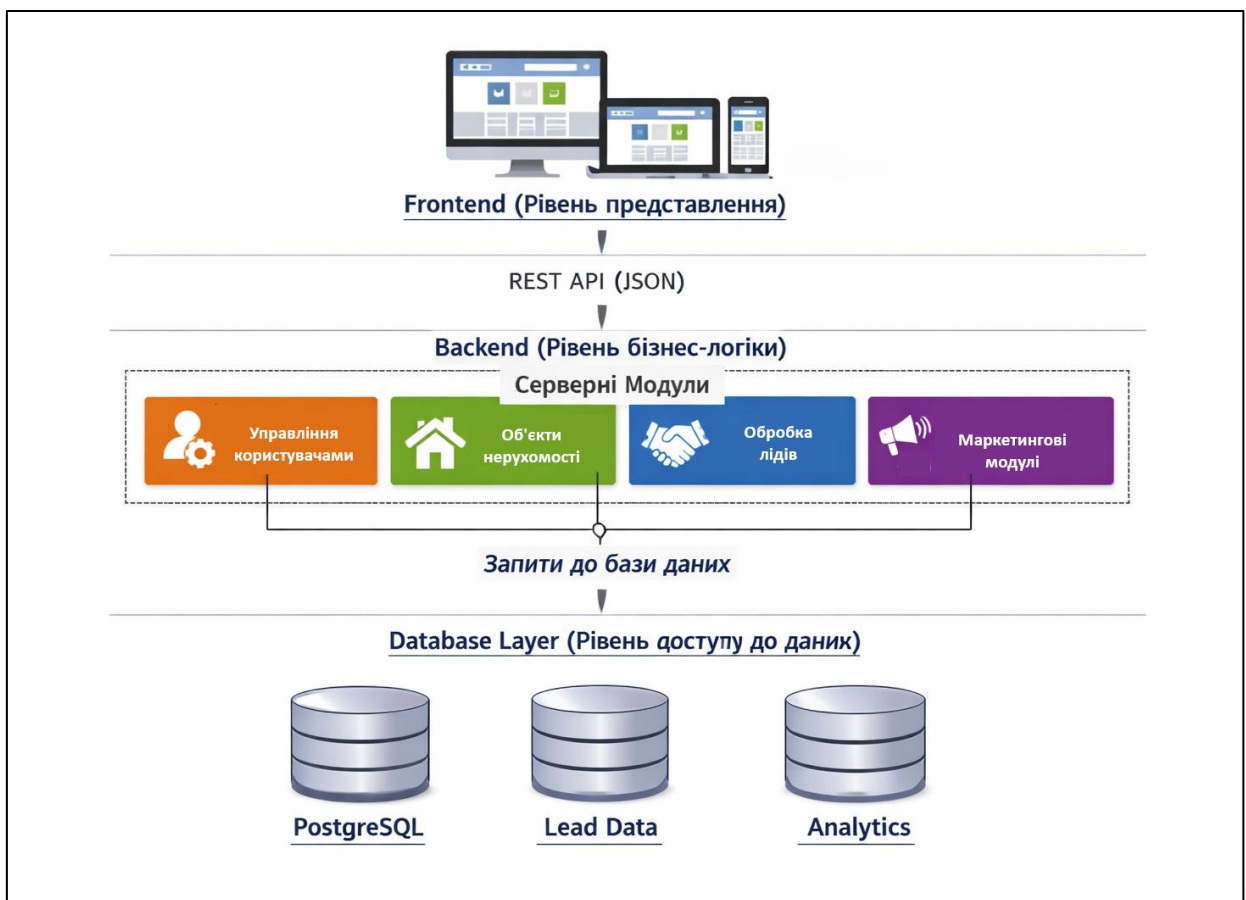


Рисунок 3.1 – Діаграма багаторівневої архітектури маркетингової системи сфери нерухомості

Тут рівень представлення (frontend) реалізує інтерфейс користувача, забезпечує введення та відображення даних. Для його створення використовується фреймворк React, що підтримує компонентну модель і динамічне оновлення контенту).

Рівень бізнес-логіки (backend) відповідає за обробку запитів, реалізацію бізнес-правил та взаємодію з базою даних. Серверна частина реалізована з використанням мови Go, що забезпечує високу продуктивність і підтримку багатопоточності. До складу бекенду входять наступні основні модулі:

- управління користувачами;
- робота з об'єктами нерухомості;
- обробка лідів;
- маркетинговий модуль.

Рівень доступу до даних (Database Layer) реалізований на основі PostgreSQL. Він забезпечує збереження, цілісність і узгодженість даних. Також слід зазначити, що модель даних побудована з урахуванням зв'язків між сутностями (Users, Properties, Leads, Campaigns, Deals).

3.3 Реалізація серверної частини

Серверна частина системи реалізує обробку HTTP-запитів, взаємодію з базою даних та виконання бізнес-логіки.

Для роботи з запитами до бази даних маркетингової системи нерухомості було реалізовано відповідні структури даних.

Структура сутності «Користувач» представлена на рисунку 3.2.

```
type User struct {
    Id      int    `json:"id"`
    Name    string `json:"name"`
    Email   string `json:"email"`
    Phone   string `json:"phone"`
    Role    string `json:"role"`
}
```

Рисунок 3.2

Структура сутності «Об’єкт нерухомості» представлена на рисунку 3.3.

```
type Property struct {  
    Id          int    `json:"id"`  
    Title       string `json:"title"`  
    Description string `json:"description"`  
    Price       float64 `json:"price"`  
    Area        float64 `json:"area"`  
    Address     string `json:"address"`  
    ImageUrl    string `json:"image_url"`  
    Type        string `json:"type"`  
    Status      string `json:"status"`  
    CreatedBy   int    `json:"created_by"`  
}
```

Рисунок 3.3

Структура даних для роботи з лідами представлена на рисунку 3.4.

```
type Lead struct {  
    Id      int    `json:"id"`  
    Name    string `json:"name"`  
    Phone   string `json:"phone"`  
    Email   string `json:"email"`  
    Source  string `json:"source"`  
    Status  string `json:"status"`  
    CreatedAt string `json:"created_at"`  
    AssignedTo int  `json:"assigned_to"`  
    PropertyId int  `json:"property_id"`  
}
```

Рисунок 3.4

Структура сутності «Кампанія» представлена на рисунку 3.5.

```
type Campaign struct {  
    Id      int    `json:"id"`  
    Name    string `json:"name"`  
    Budget  float64 `json:"budget"`  
    StartDate string `json:"start_date"`  
    EndDate  string `json:"end_date"`  
    CreatedBy int    `json:"created_by"`  
}
```

Рисунок 3.5

Структура сутності «Взаємодія» представлена на рисунку 3.6.

```
type Interaction struct {  
    ID      int    `json:"id"`  
    LeadId  int    `json:"lead_id"`  
    UserId  int    `json:"user_id"`  
    Type    string `json:"type"`  
    Note    string `json:"note"`  
    CreatedAt string `json:"created_at"`  
}
```

Рисунок 3.6

Головним елементом будь-якого вебсервера є функція, що оброблює запити (здійснює їх маршрутизацію).

Початковий код відповідної функції вебсервера маркетингової системи має вигляд, як представлено на рисунку 3.7.

```
package main

import (
    "net/http"
    "server/handlers"
)

func InitRoutes() {
    http.HandleFunc("/add_user", handlers.AddUser)
    http.HandleFunc("/get_users", handlers.GetUsers)
    http.HandleFunc("/update_user/", handlers.UpdateUser)
    http.HandleFunc("/delete_user/", handlers.DeleteUser)

    http.HandleFunc("/add_property", handlers.AddProperty)
    http.HandleFunc("/get_properties", handlers.GetProperties)
    http.HandleFunc("/update_property/", handlers.UpdateProperty)
    http.HandleFunc("/delete_property/", handlers.DeleteProperty)

    http.HandleFunc("/add_campaign", handlers.AddCampaign)
    http.HandleFunc("/get_campaigns", handlers.GetCampaigns)
    http.HandleFunc("/update_campaign/", handlers.UpdateCampaign)
    http.HandleFunc("/delete_campaign/", handlers.DeleteCampaign)

    http.HandleFunc("/add_lead", handlers.AddLead)
    http.HandleFunc("/get_leads", handlers.GetLeads)
    http.HandleFunc("/update_lead/", handlers.UpdateLead)
    http.HandleFunc("/delete_lead/", handlers.DeleteLead)

    http.HandleFunc("/add_interaction", handlers.AddInteraction)
    http.HandleFunc("/get_interactions", handlers.GetInteractions)
    http.HandleFunc("/update_interaction/", handlers.UpdateInteraction)
    http.HandleFunc("/delete_interaction/", handlers.DeleteInteraction)
}
```

Рисунок 3.7

Цей програмний код реєструє оброблювачі відповідних запитів до бази даних маркетингової системи. Так, наприклад, обробка запиту до бази даних для оновлення відомостей про користувача (user) може бути реалізована так, як представлено на рисунку 3.8.

```

func UpdateUser(w http.ResponseWriter, r *http.Request) {

    w.Header().Set("Access-Control-Allow-Origin", "*")
    w.Header().Set("Access-Control-Allow-Headers", "Content-Type")
    w.Header().Set("Access-Control-Allow-Methods", "PUT, OPTIONS")

    if r.Method == http.MethodOptions {
        return
    }

    if r.Method != http.MethodPut {
        http.Error(w, "Method not allowed",
            http.StatusMethodNotAllowed)
        return
    }

    // Отримуємо id з URL (/api/users/5)
    idStr := r.URL.Path[len("/update_user/"): ]
    id, err := strconv.Atoi(idStr)
    if err != nil || id <= 0 {
        http.Error(w, "Invalid user ID", http.StatusBadRequest)
        return
    }

    // Читаємо JSON з тіла запиту
    var updateData models.User

    if err := json.NewDecoder(r.Body).Decode(&updateData); err != nil {
        http.Error(w, "Invalid JSON", http.StatusBadRequest)
        return
    }

    dbName := "properties_db"
    connStr := fmt.Sprintf("user=postgres password=user
                                dbname=%s sslmode=disable", dbName)
    db, err := sql.Open("postgres", connStr)
    if err != nil {
        log.Fatal("Помилка підключення:", err)
    }
    defer db.Close()

    query := `
UPDATE users
SET name = $1, email = $2, phone = $3, role = $4, created_at = NOW()
WHERE id = $5
`

    _, err = db.Exec(query, updateData.Name, updateData.Email,
        updateData.Phone, updateData.Role, id)
    if err != nil {
        http.Error(w, "Failed to update user",
            http.StatusInternalServerError)
        return
    }

    w.Header().Set("Content-Type", "application/json")
    w.Header().Set("Access-Control-Allow-Origin", "*")
    w.WriteHeader(http.StatusOK)
    json.NewEncoder(w).Encode(map[string]string{"message":
        "User updated successfully"})
}

```

Рисунок 3.8

За аналогічної схемою реалізуються й інші запити до бази даних: SELECT, DELETE та INSERT.

На рисунку 3.9 наведено зовнішній вигляд запущеного у середовищі IDE GoLand [27] вебсерверу, який працює як застосунок без графічного інтерфейсу користувача.



Рисунок 3.9 – Вікно вебсервера

3.4 Реалізація клієнтської частини

Клієнтський інтерфейс забезпечує взаємодію користувача з системою та реалізує такі функції:

- перегляд об'єктів нерухомості;
- фільтрація за параметрами;
- відправка заявки;
- перегляд статистики.

Інтерфейс реалізовано з використанням компонентного підходу, що дозволяє повторно використовувати елементи інтерфейсу. Так, наприклад, компонент, що забезпечує можливість редагування окремої комірки таблиці, значення якої вибирається з іншої таблиці, може бути реалізований так, як представлено на рисунку 3.10.

Функція, що реалізує запит до вебсервера (хук `useState` бібліотеки React), може бути, наприклад, написана так, як представлено на рисунку 3.11.

```

export function ReferredTableCell(props) {
  return {
    accessorKey: props.key,
    header: props.header,
    cell: ({ row }) => {
      const isEditing = props.editingCell?.rowId === row.original.id &&
        props.editingCell?.columnId === props.key;
      const id = row.original[props.key];
      const name = props.select.find(u => u[props.refId] === id) ?
        props.select.find(u => u[props.refId] === id)[props.refName] : '—';
      return isEditing ? (
        <select
          ref={props.inputRef}
          value={id || ''}
          onChange={(e) => props.saveEdit(row.original.id, 'created_by',
            Number(e.target.value))}

          onBlur={props.cancelEdit}
          style={{ width: '100%' }}
        >
          /*<option value="">— Select user —</option>*/
          {props.select.map(val => (
            <option key={val[props.refId]} value={val[props.refId]}>
              {val[props.refName]}
            </option>
          ))}
        </select>
      ) : (
        <span
          onClick={() => props.startEditing(row.original.id,
            'created_by')}
          style={{ cursor: 'pointer', display: 'block', minHeight: '24px',
            padding: '4px 6px' }}
        >
          {name}
        </span>
      );
    }
  };
}

```

Рисунок 3.10

```

// Завантаження даних
useEffect(() => {
  fetch(window.serverURL + "/get_properties")
    .then(res => res.json())
    .then(data => setProperties(data || []))
    .catch(err => console.error("Upload error:", err));
}, []);

```

Рисунок 3.11

Загальний вигляд прототипу клієнтської частини маркетингової інформаційної системи наведено на рисунку 3.12.



Рисунок 3.12 – Загальний інтерфейс клієнтської частини

Вебсторінка для роботи з об'єктами нерухомості наведена на рисунку 3.13. Тут добре видно, що користувач має змогу редагувати дані у табличному вигляді, хоча й передбачено можливість введення інформації у модальне вікно.

Title	Description	Price	Area	Address	Image URL	Type	Status	Created By	Actions
White House	Very big house	100502	20000	Washington DC	www.yahoo.com	House	Available	Serhio	Delete
Burj Khalifa	Megatall skyscraper in Dubai	100500	30000	Dobai	https://en.wikipedia.org/wiki/Burj_Khalifa#Development	Apartment	Available	Serhio	Delete
Big House	Very big hause	123876	365	New Vasyuki DC	—	House	Available	Serhio	Delete

Рисунок 3.13 – Вебінтерфейс редагування наявних в базі даних об'єктів нерухомості

3.5 Тестування системи

У процесі розробки було проведено тестування:

- модульне тестування;
- інтеграційне тестування;
- тестування API.

Тестування дозволило виявити та усунути помилки, а також забезпечити стабільну роботу системи.

ВИСНОВКИ

У дипломній роботі було розглянуто та вирішено задачу розробки маркетингової системи для сфери нерухомості. У ході виконання роботи досягнуто поставлену мету, яка полягала у створенні програмного рішення, що забезпечує автоматизацію процесів управління об'єктами нерухомості, обробки клієнтських запитів та аналізу ефективності маркетингової діяльності.

У першому розділі було проведено аналіз предметної області, зокрема досліджено особливості ринку нерухомості та сучасні підходи до маркетингу в цій сфері. Встановлено, що ефективність діяльності агентств значною мірою залежить від використання цифрових технологій та автоматизації процесів взаємодії з клієнтами. Також було проаналізовано існуючі програмні рішення та визначено їхні переваги і недоліки, що дозволило сформулювати вимоги до розроблюваної системи.

У другому розділі виконано проектування маркетингової системи. Було визначено архітектуру програмного забезпечення, сформульовано функціональні та нефункціональні вимоги, а також розроблено модель даних і UML-діаграми. Запропонована архітектура забезпечує гнучкість, масштабованість та можливість подальшого розвитку системи.

У третьому розділі описано процес реалізації системи з використанням сучасних технологій. Було обґрунтовано вибір інструментів розробки, реалізовано серверну та клієнтську частини, а також інтеграцію із зовнішніми сервісами. Реалізовані функціональні можливості дозволяють ефективно управляти об'єктами нерухомості, обробляти ліди та здійснювати аналіз маркетингових кампаній.

У процесі виконання роботи було досягнуто таких результатів:

- проведено аналіз ринку нерухомості та маркетингових підходів;
- розроблено архітектуру маркетингової інформаційної системи;

- створено модель бази даних та основні алгоритми роботи системи;
- реалізовано програмний прототип маркетингової системи нерухомості, який складається із клієнтської та серверних частин.

Практична цінність отриманих результатів полягає у можливості використання розробленої системи в діяльності агентств нерухомості для підвищення ефективності маркетингових процесів та оптимізації взаємодії з клієнтами.

Перспективи подальших досліджень можуть бути пов'язані з розширенням функціональних можливостей системи, зокрема впровадженням інтелектуальних алгоритмів аналізу даних, рекомендаційних систем, а також інтеграцією з додатковими маркетинговими платформами та сервісами (Google Analytics).

Таким чином, розроблена маркетингова система відповідає поставленим вимогам та може бути використана як основа для створення повноцінного програмного продукту у сфері нерухомості.

ПЕРЕЛІК ПОСИЛАНЬ

1. Сенишин О. С., Кривешко О. В. Маркетинг : навчальний посібник. Львів : ЛНУ ім. І. Франка, 2020. 347 с.
2. Salesforce : The #1 AI CRM. URL: <https://www.salesforce.com/> (дата звернення: 18.03.2026).
3. HubSpot | Software & Tools for your Business – Homepage. URL: <https://www.hubspot.com/home-b> (дата звернення: 18.03.2026).
4. Zoho CRM – Software for Businesses. URL: <https://zoho-crm.digital/> (дата звернення: 18.03.2026).
5. Павленко П. М., Філоненко С. Ф., Бабіч К. С. Інформаційні системи і технології : навч. посіб. Київ : НАУ, 2013. 324 с.
6. Google Ads – Залучайте більше клієнтів і подавайте більше за допомогою онлайн-реклами. URL: <https://business.google.com/ua/google-ads/?pli=1> (дата звернення: 18.03.2026).
7. Facebooks Ads Manager : Ads Manager for Facebook, Instagram. URL: <https://www.facebook.com/business/tools/ads-manager/> (дата звернення: 18.03.2026).
8. Craigslist. *California*. URL: <https://www.craigslist.org/> (дата звернення: 18.03.2026).
9. Electronics, Cars, Fashion, Collectibles & More. *eBay*. URL: <https://www.ebay.com/> (дата звернення: 18.03.2026).
10. Оголошення OLX Україна. URL: <https://www.olx.ua/uk/> (дата звернення: 18.03.2026).
11. Prom – найбільший маркетплейс України. URL: <https://prom.ua/ua/> (дата звернення: 18.03.2026).
12. Оголошення RIA.com : безкоштовна дошка оголошень України. URL: <https://www.ria.com/uk/> (дата звернення: 18.03.2026).
13. Kabanchik.ua – онлайн-сервіс замовлення послуг. URL: <https://kabanchik.ua/> (дата звернення: 18.03.2026).

14. Google Analytics. URL: <https://marketingplatform.google.com/info?authuser=0> (дата звернення: 18.03.2026).
15. Marketing Cloud | Adobe Experience Cloud. URL: <https://business.adobe.com/products/marketing-cloud/main.html> (дата звернення: 18.03.2026).
16. Power BI. URL: <https://app.powerbi.com/home> (дата звернення: 18.03.2026).
17. Twilio Segment Customer Data Platform. *Twilio*. URL: <https://www.twilio.com/en-us/segment> (дата звернення: 18.03.2026).
18. BlueConic | The Platform Powering Customer Growth. URL: <https://www.blueconic.com/> (дата звернення: 18.03.2026).
19. Top 10 CDP. URL: <https://esputnik.com/uk/blog/top-ten-cdp-platform> (дата звернення: 18.03.2026).
20. REST API Introduction. URL: <https://www.geeksforgeeks.org/node-js/rest-api-introduction/> (дата звернення: 18.03.2026).
21. UML overview – where and why each UML diagram is used. URL: <https://www.drawio.com/blog/uml-overview> (дата звернення: 18.03.2026).
22. MySQL. URL: <https://www.mysql.com/> (дата звернення: 19.03.2026).
23. PostgreSQL: The world's most advanced open source database. URL: <https://www.postgresql.org/> (дата звернення: 19.03.2026).
24. The Go Programming Language. URL: <https://go.dev/> (дата звернення: 19.03.2026).
25. What is DevOps? Meaning, methodology and guide. URL: <https://www.techtarget.com/searchitoperations/definition/DevOps> (дата звернення: 19.03.2026).
26. React. The library for web and native user interfaces. URL: <https://react.dev/> (дата звернення: 19.03.2026).
27. GoLand by JetBrains: More than just a Go IDE. URL: <https://www.jetbrains.com/go/> (дата звернення: 19.03.2026).