МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

МАТЕМАТИЧНИЙ ФАКУЛЬТЕТ
Кафедра програмної інженерії

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

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

Виконав: студент 4 курсу, групи 6.1212-пі-1
спеціальності 121 інженерія програмного забезпечення
(шифр і назва спеціальності)
освітньої програми програмна інженерія
(назва освітньої програми)
В.В. Кириченко
(ініціали та прізвище)

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

Рецензент доцент кафедри комп'ютерних наук, к.т.н. Добровольський Г.А.
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Запоріжжя – 2026

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

Кириченку Владиславу Віталійовичу
(прізвище, ім’я та по-батькові)

1. Тема роботи Розробка CRM 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 та постановка задачі
2. Проєктування CRM системи для сфери нерухомості.
3. Реалізація та тестування CRM 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.

К А Л Е Н Д А Р Н И Й П Л А Н

№	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Розробка плану роботи.	19.12.2025	
2.	Збір вихідних даних.	12.01.2026	
3.	Обробка методичних та теоретичних	02.02.2026	
	джерел.		
4.	Розробка першого та другого розділу.	23.03.2026	
5.	Розробка третього розділу.	12.05.2026	
6.	Оформлення та нормоконтроль	25.05.2026	
	кваліфікаційної роботи бакалавра.		
7.	Захист кваліфікаційної роботи.	15.06.2026	

Студент _____ В.В. Кириченко
(підпис) (ініціали та прізвище)

Керівник роботи _____ О.М. Мильцев
(підпис) (ініціали та прізвище)

Нормоконтроль пройдено

Нормоконтролер _____ А.В. Столярова
(підпис) (ініціали та прізвище)

РЕФЕРАТ

Кваліфікаційна робота бакалавра «Розробка CRM системи для сфери нерухомості»: 82 с., 32 рис., 21 табл., 28 джерел, 1 додаток.

АВТОМАТИЗАЦІЯ, АГЕНТСТВО НЕРУХОМОСТІ, БАЗА ДАНИХ, ВЕБЗАСТОСУНОК, ЗАЯВКА КЛІЄНТА, КЛІЄНТСЬКА БАЗА, НЕРУХОМІСТЬ.

Об’єкт дослідження – процес автоматизації діяльності агентства нерухомості під час роботи з клієнтами, об’єктами, заявками, переглядами й угодами.

Мета роботи: розробити CRM систему для сфери нерухомості, призначену для централізованого обліку клієнтів, об’єктів, заявок, переглядів, угод і підтримки управлінської звітності агентства.

Методи дослідження – аналіз предметної області, порівняльний аналіз програмних аналогів, формалізація функціональних вимог, UML-моделювання, проєктування бази даних, методи веброзробки, функціональне тестування програмного забезпечення.

В кваліфікаційній роботі розглянуто процеси взаємодії агентства нерухомості з клієнтами, охарактеризовано проблеми розрізненого збереження даних та обґрунтовано потребу в централізованому програмному засобі для підтримки рієлторської діяльності. Проведено огляд CRM-рішень, які застосовуються для автоматизації продажів і роботи з нерухомістю, визначено їх переваги та обмеження для локального агентства.

Практичну реалізацію виконано у вигляді вебзастосунку з використанням Node.js, Express.js, SQLite, HTML, CSS, JavaScript і EJS. Розроблена система підтримує авторизацію користувачів, рольове розмежування доступу, ведення клієнтів, облік об’єктів нерухомості, роботу з заявками, планування переглядів, супровід угод, перегляд звітності та адміністрування користувачів.

SUMMARY

Bachelor’s qualifying paper “Development of a Real Estate CRM System”: 82 pages, 32 figures, 21 tables, 28 references, 1 supplement.

AUTOMATION, CLIENT BASE, CLIENT REQUEST, CRM SYSTEM, DATABASE, REAL ESTATE, REAL ESTATE AGENCY, ROLE-BASED ACCESS, TRANSACTION, WEB APPLICATION.

The object of the study is the process of automating the activities of a real estate agency during work with clients, property objects, requests, viewings, and transactions.

The aim of the study is to develop a CRM system for the real estate sector intended for centralized accounting of clients, property objects, requests, viewings, transactions, and support of management reporting in an agency.

The methods of research include subject area analysis, comparative analysis of software analogues, formalization of functional requirements, UML modeling, database design, web development methods, and functional testing of software.

The qualifying paper examines the processes of interaction between a real estate agency and its clients, characterizes the problems of fragmented data storage, and substantiates the need for a centralized software tool to support real estate activities. The paper reviews CRM solutions used for sales automation and real estate management, identifies their advantages, and determines their limitations for a local agency.

The paper formulates functional requirements for the CRM system, defines the roles of administrator, manager, and real estate agent, and develops the software architecture, database structure, and UML diagrams. The practical implementation is performed as a web application using Node.js, Express.js, SQLite, HTML, CSS, JavaScript, and EJS.

ЗМІСТ

Завдання на кваліфікаційну роботу 2
Реферат 4
Summary 5
Вступ 7
1 Аналіз предметної області та постановка задачі 10
1.1 Характеристика процесів взаємодії агентства нерухомості з клієнтами 10
1.2 Огляд програмних рішень для автоматизації рієлторської діяльності 13
1.3 Формування функціональних вимог до CRM системи 19
1.4 Постановка задачі розробки програмного продукту 24
1.5 Висновки до розділу 1 28
2 Проєктування crm системи для сфери нерухомості 29
2.1 Визначення ролей користувачів та сценаріїв роботи системи 29
2.2 Побудова архітектури програмного забезпечення 33
2.3 Моделювання структури даних інформаційної системи 38
2.4 Розробка UML-діаграм для опису логіки взаємодії 42
2.5 Висновки до розділу 2 47
3 Реалізація та тестування CRM-системи 49
3.1 Обґрунтування вибору технологічного стеку розробки 49
3.2 Реалізація серверної частини програмного продукту 52
3.3 Створення клієнтського інтерфейсу CRM системи 57
3.4 Перевірка працездатності основних функцій застосунку 65
3.5 Підготовка керівництва користувача для роботи з системою 72
3.6 Висновки до розділу 3 74
Висновки 76
Перелік посилань 78
Додаток А Лістинг головного файла запуску CRM-системи 81

ВСТУП

Ринок нерухомості характеризується високою інтенсивністю комунікації між агентствами, власниками об’єктів, покупцями, орендарями та інвесторами. В процесі роботи, рієлторська організація опрацьовує значний обсяг відомостей про клієнтів, об’єкти, заявки, перегляди, домовленості та фінальні угоди. Частина інформації має комерційну цінність, а частина стосується персональних даних клієнтів, тому її збереження, оновлення та контроль доступу потребують впорядкованого програмного забезпечення.

Актуальність теми зумовлена потребою агентств нерухомості в централізованому інструменті для керування клієнтською базою, об’єктами, заявками, переглядами та угодами. Використання електронних таблиць, особистих нотаток, месенджерів та розрізнених файлів не забезпечує належного рівня прозорості робочих процесів.

CRM-система в сфері нерухомості має поєднувати операційні й управлінські функції. Агенту потрібні засоби для швидкого додавання клієнтів, створення заявок, планування переглядів і ведення угод. Керівнику необхідні інструменти контролю результативності працівників, аналізу активних звернень і перегляду звітності. Адміністратор повинен мати можливість керувати обліковими записами, ролями та журналом дій.

Метою кваліфікаційної роботи є розробка CRM-системи для сфери нерухомості, призначеної для централізованого обліку клієнтів, об’єктів, заявок, переглядів, угод та підтримки управлінської звітності агентства.

Для досягнення поставленої мети необхідно виконати наступні завдання:

- проаналізувати предметну область діяльності агентства нерухомості та визначити проблеми взаємодії з клієнтами;
- розглянути програмні рішення, які застосовуються для автоматизації рієлторської діяльності;

- сформувати функціональні вимоги до CRM-системи та конкретизувати постановку задачі розробки;
- визначити ролі користувачів, основні сценарії роботи та архітектурну організацію програмного продукту;
- побудувати структуру даних та UML-діаграми для опису логіки роботи системи;
- обґрунтувати технологічний стек та реалізувати серверну частину CRM-системи;
- створити клієнтський інтерфейс для роботи адміністратора, керівника, агента з нерухомості;
- перевірити працездатність основних функцій застосунку та підготувати керівництво користувача.

Об’єктом дослідження є процес автоматизації діяльності агентства нерухомості під час роботи з клієнтами, об’єктами, заявками, переглядами та угодами.

Предметом дослідження є програмні засоби створення CRM-системи для централізованого керування інформацією агентства нерухомості.

Методи дослідження охоплюють аналіз предметної області, порівняльний аналіз програмних аналогів, формалізацію функціональних вимог, UML-моделювання, проєктування бази даних, методи веброзробки, реалізацію серверної логіки, створення клієнтського інтерфейсу та функціональне тестування.

Практичне значення роботи полягає в створенні працездатного вебзастосунку, який може використовуватися для демонстрації автоматизації основних процесів агентства нерухомості. Розроблена CRM-система забезпечує авторизацію користувачів, рольове розмежування доступу, ведення клієнтів, облік нерухомості, роботу з заявками, планування переглядів, супровід угод, перегляд звітності та адміністрування користувачів.

Структура кваліфікаційної роботи складається зі вступу, трьох основних розділів, висновків, переліку посилань та додатку. В першому

розділі розглянуто предметну область, аналоги, вимоги та постановку задачі. В другому розділі виконано проєктування ролей, сценаріїв, архітектури, структури даних і UML-моделей. В третьому розділі описано технологічний стек, реалізацію серверної частини, клієнтський інтерфейс, перевірку працездатності й керівництво користувача.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Характеристика процесів взаємодії агентства нерухомості з клієнтами

Сфера нерухомості належить до напрямів підприємницької діяльності, де результат роботи значною мірою залежить від якості комунікації з клієнтом, швидкості оброблення звернень та точності збереження інформації про об’єкти. Агентство нерухомості одночасно працює з власниками житла, покупцями, орендарями, інвесторами, забудовниками, нотаріальними партнерами й фінансовими установами. Кожна взаємодія формує інформаційний слід, який впливає на подальший продаж, оренду, супровід угоди або повторне звернення клієнта. За відсутності впорядкованої CRM-системи, значна частина відомостей зберігається в особистих нотатках агентів, месенджерах, електронній пошті або окремих електронних таблицях, що ускладнює контроль виконання заявок та знижує прозорість роботи агентства [1].

Особливість рієлторської діяльності полягає в поєднанні консультаційної, інформаційної, організаційної та договірної складових. Клієнт рідко приймає рішення одразу після першого звернення, оскільки купівля або оренда нерухомості пов’язана з високою вартістю, юридичними ризиками, аналізом місця розташування, стану об’єкта, інфраструктури та майбутніх витрат. Агентство має підтримувати довготривалу комунікацію, фіксувати історію переговорів, зберігати вимоги клієнта та своєчасно повертатися до нього з релевантними пропозиціями.

Початковий етап взаємодії розпочинається зі звернення через сайт, телефонний дзвінок, соціальні мережі, рекламний майданчик або особисту рекомендацію. На практиці, один клієнт може залишити кілька звернень через різні канали, тому агентству необхідно ідентифікувати дублікати,

визначати джерело ліда та фіксувати первинний інтерес. Для CRM-системи важливим є не лише факт звернення, а й контекст, тобто тип операції, очікувана ціна, бажаний район, кількість кімнат, термін прийняття рішення, джерело інформації про агентство [2]. Основні процеси взаємодії агентства нерухомості з клієнтами представлено в таблиці 1.1.

Таблиця 1.1 – Основні процеси взаємодії агентства нерухомості з клієнтами

Етап взаємодії	**Зміст роботи агентства**	**Дані, що потребують фіксації**	**Проблеми без CRM-системи**
Первинне звернення	Прийняття заявки, уточнення потреби, визначення типу операції	Контактні відомості, канал звернення, бюджет, район, формат нерухомості	Втрата лідів, дублювання записів, відсутність джерела звернення
Кваліфікація клієнта	Оцінювання реальності наміру, перевірка платоспроможності, визначення пріоритетності	Статус клієнта, готовність до угоди, термін пошуку, особливі вимоги	Нерівномірне навантаження агентів, робота з неактуальними заявками
Підбір об’єктів	Порівняння вимог клієнта з базою нерухомості та підготовка пропозицій	Параметри об’єктів, відповідність запиту, історія надісланих варіантів	Повторне надсилання однакових пропозицій, неповна історія комунікації
Організація переглядів	Узгодження часу, маршруту, присутності власника й агента	Дата, час, відповідальний працівник, результат перегляду	Зрив зустрічей, конфлікти в розкладі, відсутність контролю виконання
Супровід домовленостей	Фіксація умов угоди, погодження ціни, підготовка документів	Стадія угоди, домовленості сторін, перелік документів, ризики	Втрата важливих деталей, затримка оформлення, слабкий управлінський контроль
Післяугодна робота	Підтримання контакту, збір відгуку, пропозиція додаткових послуг	Рівень задоволеності, дата повторного контакту, рекомендації	Втрата повторних продажів, відсутність клієнтської бази для розвитку

Взаємодія агентства нерухомості з клієнтом охоплює повний цикл обробки інформації, де кожен етап залежить від коректності попередніх дій. Якщо агент не зафіксував причину відмови після перегляду, наступний підбір може залишитися неточним. Якщо керівник не бачить, на якій стадії перебуває угода, виникають труднощі з прогнозуванням доходу, плануванням рекламного бюджету й оцінюванням роботи працівників [3].

Для агентства об’єкт є інформаційною одиницею, пов’язаною з власником, історією переглядів, рекламними публікаціями, фото- та відеоматеріалами, юридичним статусом, коментарями агентів і поточним станом угоди.

Комунікація між агентом та клієнтом в сфері нерухомості має персоналізований характер. Покупець очікує, що його вимоги будуть враховані без повторного пояснення, а орендодавець розраховує на регулярне інформування про зацікавленість об’єктом. CRM-система повинна підтримувати збереження історії дзвінків, листування, нотаток після зустрічей та результатів переглядів. За рахунок єдиної інформаційної бази, агент може швидко відновити контекст спілкування, а керівник отримує можливість контролювати якість сервісу не лише за фінальним результатом, але й за проміжними діями [4].

Для рієлторського бізнесу важливою є сегментація клієнтів за намірами, бюджетом, типом операції та рівнем готовності до укладення угоди. Частина звернень має короткий цикл обробки, оскільки клієнт вже визначився з районом та фінансовими умовами. Інша частина потребує тривалого супроводу, консультацій та періодичного повернення до діалогу після появи нових об’єктів. Без автоматизованої системи агентство не може стабільно підтримувати різні сценарії обслуговування, оскільки контроль нагадувань і статусів залишається залежним від дисципліни окремого працівника.

Інформаційні потреби учасників рієлторського процесу узагальнено в таблиці 1.2.

Таблиця 1.2 – Інформаційні потреби учасників рієлторського процесу

Учасник процесу	**Основна інформаційна потреба**	**Очікуваний результат автоматизації**
Клієнт-покупець	Отримання пропозицій, що відповідають бюджету, району та параметрам житла	Швидкий підбір релевантних об’єктів та зменшення кількості зайвих переглядів
Клієнт-продавець	Контроль зацікавленості об’єктом, переглядів і переговорів щодо ціни	Прозора історія роботи з об’єктом та регулярне інформування власника
Агент з нерухомості	Доступ до заявок, об’єктів, контактів, нагадувань та результатів зустрічей	Зменшення ручної роботи та підвищення точності комунікації
Керівник агентства	Оцінювання завантаження працівників, стадій угод та фінансового прогнозу	Управлінський контроль, аналітика продажів і планування ресурсів
Адміністратор системи	Керування обліковими записами, правами доступу та довідниками	Захищене адміністрування даних та розмежування відповідальності

Управління клієнтськими даними в агентстві нерухомості потребує розмежування доступу, оскільки інформація про контакти, фінансові можливості, документи та домовленості має комерційну цінність [5]. Агент повинен працювати з заявками та об’єктами, за які він відповідає, керівник має бачити узагальнену картину діяльності, а адміністратор забезпечує стабільність роботи системи та коректність налаштувань.

1.2 Огляд програмних рішень для автоматизації рієлторської діяльності

Автоматизація рієлторської діяльності розвивається в напрямі поєднання CRM-функціональності, керування об’єктами нерухомості,

підтримки комунікацій з клієнтами та аналітики продажів. Для агентства нерухомості недостатньо використовувати звичайну таблицю з контактами, оскільки робота охоплює заявки, власників, покупців, орендарів, перегляди, домовленості, документи, рекламні джерела та етапи укладання угод. Програмне рішення має забезпечувати централізоване збереження відомостей, контроль відповідальних працівників, відстеження стадій угоди та швидкий доступ до історії взаємодії з клієнтом.

На ринку представлено дві групи програмних продуктів, які можуть застосовуватися в рієлторській діяльності. До першої групи належать універсальні CRM-платформи, орієнтовані на продажі в різних галузях. До другої групи належать спеціалізовані рішення для агентств нерухомості, брокерів або інвесторів. Універсальні системи зазвичай мають потужні інструменти керування лідами, угодами та аналітикою, однак потребують додаткового налаштування під роботу з об’єктами. Галузеві системи краще враховують специфіку рієлторського процесу, проте можуть бути менш гнучкими для невеликого агентства або мати надмірну вартість впровадження.

Salesforce Sales Cloud є однією з найвідоміших CRM-платформ для керування продажами. Сервіс пропонує функції контактного менеджменту, автоматизації, співпраці та штучного інтелекту, що робить платформу придатною для великих команд з розгалуженими процесами продажів. Перевагою Salesforce є масштабованість та гнучке налаштування бізнес-процесів, а недоліком виступає складність впровадження, потреба в адмініструванні та висока залежність від правильної конфігурації модулів [6]. Головна сторінка сервісу Salesforce Sales Cloud наведена на рисунку 1.1.

HubSpot Sales Hub поєднує CRM, керування угодами, комунікації з клієнтами, автоматизацію персоналізованих звернень, планування зустрічей та роботу з єдиною базою даних. Sales Hub допомагає перетворювати потенційних клієнтів на угоди, зберігати дані в одному середовищі та використовувати AI-функції для підвищення продуктивності продажів.

Система не має глибокої початкової орієнтації на картки об’єктів нерухомості, тому для повноцінного використання в агентстві потрібні кастомні поля, налаштування етапів та додаткові інтеграції [7]. Головна сторінка сервісу HubSpot Sales Hub наведена на рисунку 1.2.

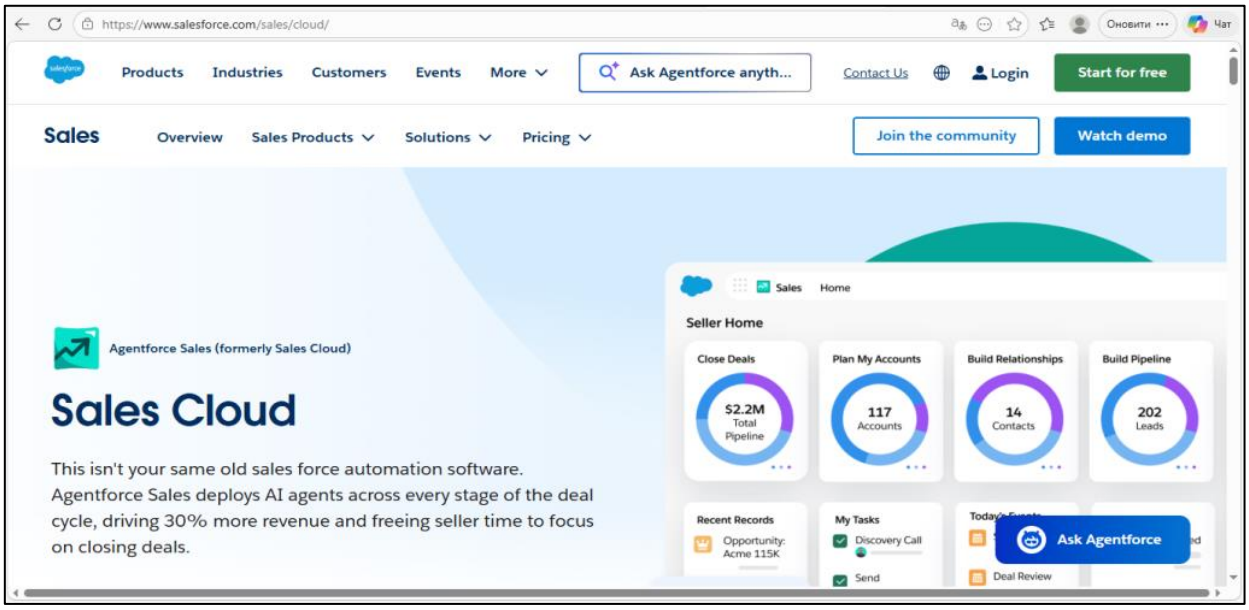


Рисунок 1.1 – Головна сторінка сервісу Salesforce Sales Cloud

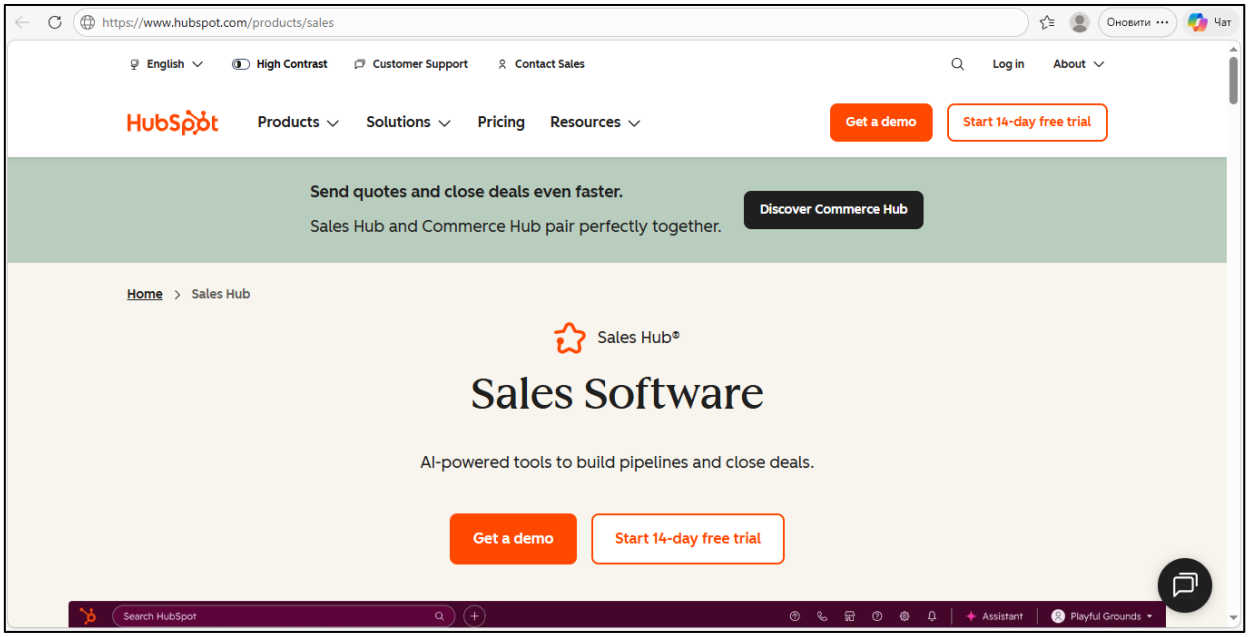


Рисунок 1.2 – Головна сторінка сервісу HubSpot Sales Hub

Pipedrive позиціонується як CRM для керування воронкою продажів, лідами, угодами та автоматизацією процесу продажу. Pipedrive зазначає

можливість збереження відомостей про покупців, продавців, інвесторів, об’єкти, історію контактів і транзакції в одному середовищі. Значною перевагою Pipedrive є візуальна логіка воронки, завдяки якій менеджер бачить стадію кожної заявки або угоди. Недоліком є потреба в адаптації структури даних під рієлторські завдання, оскільки типові CRM-сутності не завжди повністю відображають параметри об’єкта, юридичний стан, історію переглядів та зв’язок між власником та потенційним покупцем [8]. Головна сторінка сервісу Pipedrive наведена на рисунку 1.3.

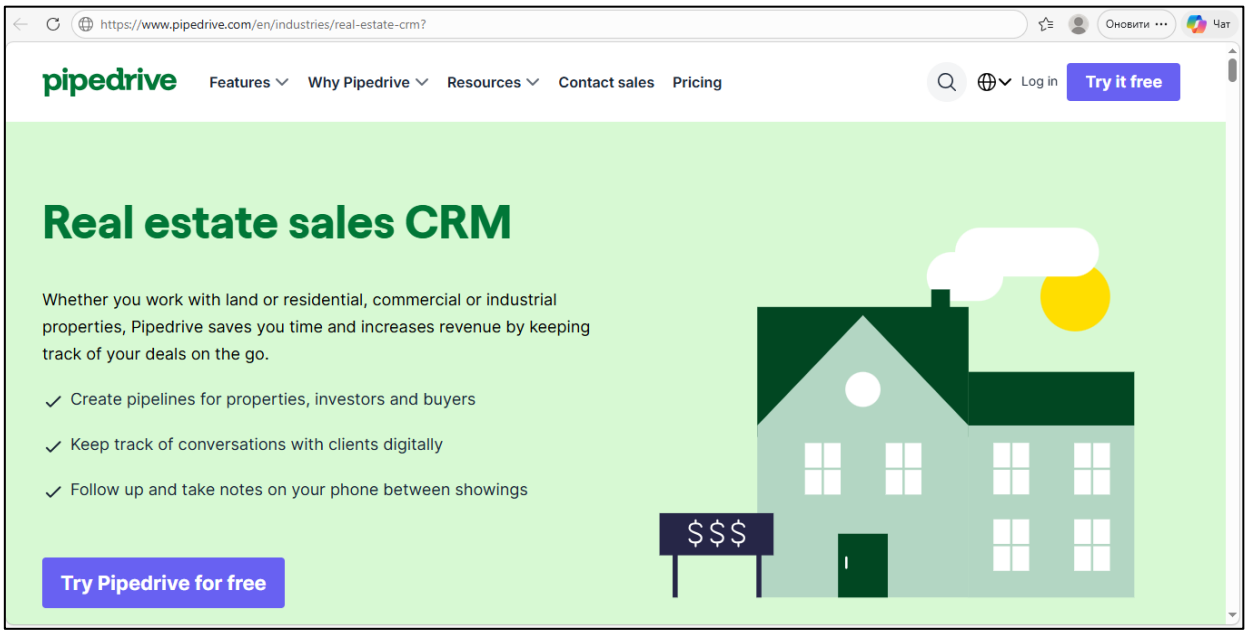


Рисунок 1.3 – Головна сторінка сервісу Pipedrive

Zoho CRM є універсальною платформою для продажів, яка підтримує автоматизацію, керування лідами, аналітику, налаштування бізнес-процесів, інтеграції та AI-функції. Перевагою Zoho CRM є гнучкість налаштувань та можливість побудови власних етапів роботи з клієнтами. Недоліком є складність первинної конфігурації для невеликої команди, оскільки універсальна платформа потребує попереднього проєктування структури об’єктів, полів, ролей і правил автоматизації [9]. Головна сторінка сервісу Zoho CRM зображена на рисунку 1.4.

Спеціалізованим прикладом галузевої CRM є Follow Up Boss, орієнтована на команди агентів нерухомості. Система акцентує увагу на

автоматичному листуванні та текстових повідомленнях для нових лідів, підказках щодо пріоритетних дзвінків, мобільній роботі агентів, відстеженні розмов, нагадуваннях і сповіщеннях. Для рієлторської сфери цінність Follow Up Boss полягає в підтримці швидкого реагування на нові звернення, що особливо важливо для продажів і оренди, де затримка відповіді часто призводить до втрати клієнта. Обмеженням для локального агентства може бути орієнтація на ринок з власною екосистемою рекламних джерел і комунікаційних сервісів [10].

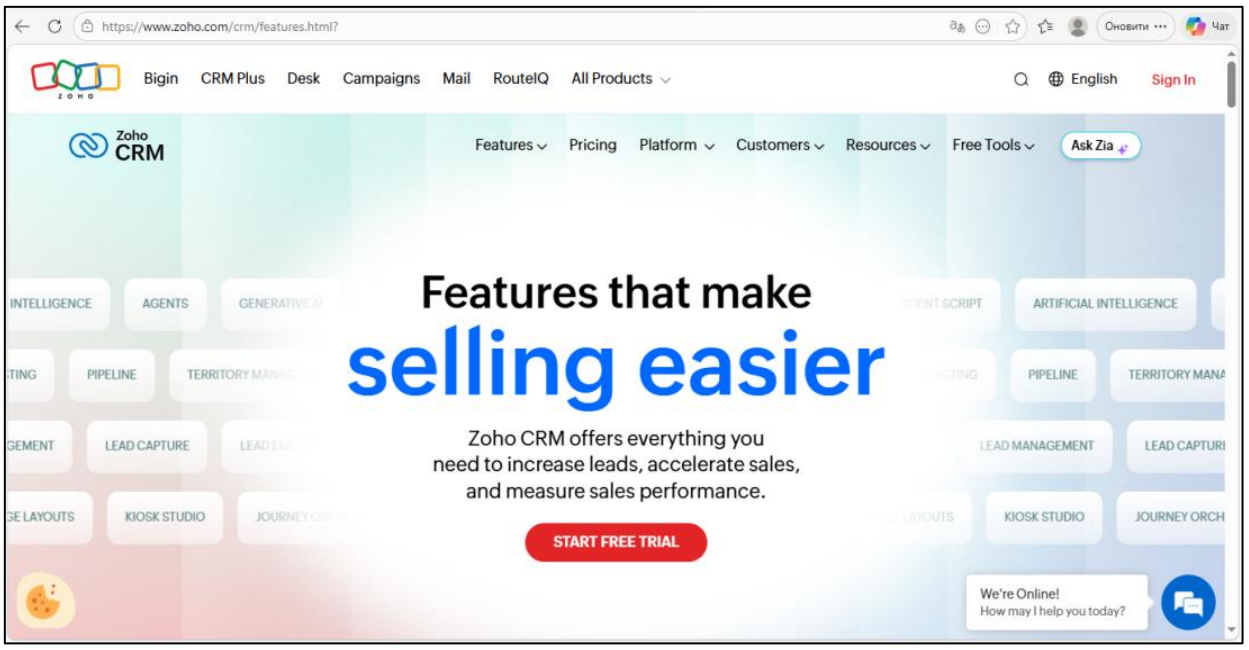


Рисунок 1.4 – Головна сторінка сервісу Zoho CRM

Propertybase by Lone Wolf належить до рішень, створених спеціально для нерухомості. Платформа характеризується як Salesforce-powered real estate CRM, що централізує ліди та об’єкти, автоматизує повсякденні процеси та поєднує маркетинг, генерацію лідів, відстеження угод та звітність. Перевагою Propertybase є глибша відповідність рієлторським процесам порівняно з універсальними CRM [11].

Real Geeks поєднує CRM та вебінструменти для залучення клієнтів у сфері нерухомості. Перевагою Real Geeks є поєднання CRM з маркетинговими інструментами, що дозволяє працювати не лише з наявними

клієнтами, але й з джерелами залучення [12]. Порівняльна характеристика CRM-рішень для рієлторської діяльності представлена в таблиці 1.3.

Таблиця 1.3 – Порівняльна характеристика CRM-рішень для рієлторської діяльності

Програмне рішення	**Основне призначення**	**Сильні сторони**	**Обмеження для агентства нерухомості**
Salesforce Sales Cloud	Універсальна CRM для керування продажами	Масштабованість, автоматизація, мобільний доступ, аналітика, AI-функції	Складне впровадження, потреба в адмініструванні, відсутність моделі об’єктів нерухомості
HubSpot Sales Hub	CRM і платформа продажів для команд різного масштабу	Зручна робота з лідами, угодами, зустрічами, комунікаціями й автоматизацією	Галузева специфіка нерухомості потребує додаткового налаштування
Pipedrive	CRM для візуального керування воронкою продажів	Просте відстеження стадій угод, автоматизація, зручна робота з лідами	Параметри нерухомості та переглядів потребують кастомізації
Zoho CRM	Гнучка CRM-платформа для продажів і аналітики	Автоматизація, налаштування процесів, аналітика, інтеграції	Первинне налаштування може бути складним для невеликої команди
Follow Up Boss	CRM для агентів і команд нерухомості	Швидка робота з лідами, нагадування, мобільний доступ, історія комунікацій	Орієнтація на усталену екосистему англомовного ринку
Propertybase	Галузева CRM для агентств нерухомості	Централізація лідів, об’єктів, маркетингу, угод і звітності	Висока складність впровадження для невеликого агентства
Real Geeks	CRM з вебінструментами для залучення лідів	Поєднання сайту, CRM, відстеження активності й маркетингу	Частина можливостей виходить за базові потреби локальної CRM

Універсальні системи мають розвинену інфраструктуру, але часто потребують складного налаштування [13]. Галузеві системи краще відображають специфіку рієлторської діяльності, однак можуть бути перевантажені функціями, орієнтованими на великі команди, іноземний ринок або інвесторські процеси.

1.3 Формування функціональних вимог до CRM системи

Формування функціональних вимог до CRM-системи для сфери нерухомості має спиратися на особливості рієлторської діяльності, результати аналізу аналогів та потреби користувачів, які беруть участь в роботі агентства. Програмний продукт повинен забезпечити не лише збереження контактів клієнтів, аде й повний супровід процесу роботи з заявками, об’єктами нерухомості, переглядами, угодами та управлінською звітністю.

CRM-система для агентства нерухомості має працювати як єдине інформаційне середовище, де кожен клієнт, об’єкт, заявка та угода пов’язані між собою. Якщо клієнт звертається з запитом на купівлю квартири, система повинна дозволяти зафіксувати його контактні відомості, бюджет, район пошуку, бажану площу, кількість кімнат та додаткові побажання. Після реєстрації звернення, агент отримує можливість підібрати об’єкти, запланувати перегляд, внести результат зустрічі, змінити статус заявки та передати угоду на наступний етап [14].

Основними користувачами системи є адміністратор, керівник агентства та агент з нерухомості. Адміністратор відповідає за облікові записи, ролі, довідники й загальні налаштування. Керівник агентства контролює роботу працівників, переглядає аналітичні показники, оцінює стан заявок і угод. Агент виконує щоденні операції з клієнтами, об’єктами, переглядами та нотатками. Ролі користувачів CRM-системи та їх функціональне призначення наведено в таблиці 1.4.

Таблиця 1.4 – Ролі користувачів CRM-системи та їх функціональне призначення

Роль користувача	**Функціональне призначення**	**Очікуваний результат роботи**
Адміністратор	Керування користувачами, правами доступу та загальними параметрами системи	Підтримка коректної структури даних та безпечної роботи застосунку
Керівник агентства	Перегляд заявок, угод, звітів, активності агентів і стану об’єктів	Контроль діяльності персоналу
Агент з нерухомості	Обробка клієнтів, додавання об’єктів, планування переглядів та ведення угод	Оперативне виконання рієлторських завдань та збереження історії комунікації
Клієнтська база	Непряма роль, що відображається через картки клієнтів і заявки	Накопичення структурованих відомостей для повторної роботи й аналітики

Система повинна надавати можливість створювати, редагувати, переглядати й архівувати картки клієнтів. В картці варто зберігати ім’я, номер телефону, електронну пошту, тип клієнта, джерело звернення, бюджет, коментарі агента та поточний статус взаємодії. Для керівника важливо бачити, хто відповідає за клієнта, коли відбувся останній контакт і який результат було отримано.

Кожен об’єкт повинен мати окрему картку з технічними, фінансовими та описовими характеристиками. До основних атрибутів належать тип нерухомості, адреса або район, площа, кількість кімнат, поверх, ціна, статус доступності, дані власника, опис переваг, фотографії та службові коментарі [15]. Основні функціональні вимоги до CRM-системи наведено в таблиці 1.5.

Функціональний блок заявок має поєднувати клієнта з його потребою. Один клієнт може мати кілька запитів, наприклад пошук квартири для купівлі й окремий інтерес до оренди комерційного приміщення. Заявка повинна зберігати тип операції, бажані параметри нерухомості, фінансовий

діапазон, пріоритетність, статус і відповідального агента. Статуси заявок мають відображати реальний перебіг роботи, тобто, нове звернення, уточнення потреби, підбір варіантів, перегляд, переговори, завершення або відмова. Система повинна дозволяти змінювати статуси без втрати попередньої історії.

Таблиця 1.5 – Основні функціональні вимоги до CRM-системи

Функціональний блок	**Зміст вимоги**	**Значення для агентства нерухомості**
Авторизація користувачів	Вхід до системи за логіном і паролем з урахуванням ролі	Захист доступу до клієнтських та комерційних даних
Керування клієнтами	Створення, редагування, пошук і перегляд карток клієнтів	Формування єдиної бази контактів та історії звернень
Облік об’єктів	Додавання, оновлення, фільтрація та перегляд нерухомості	Підтримка актуального каталогу пропозицій
Оброблення заявок	Реєстрація потреб клієнта, призначення відповідального агента, зміна статусу	Контроль проходження клієнта від звернення до угоди
Планування переглядів	Фіксація дати, часу, об’єкта, клієнта, агента та результату зустрічі	Зменшення ризику пропущених зустрічей і накладок в розкладі
Супровід угод	Ведення стадій переговорів, умов домовленості й фінального результату	Підвищення прозорості продажів та прогнозування доходу
Пошук і фільтрація	Відбір клієнтів, заявок та об’єктів за ключовими параметрами	Прискорення щоденної роботи агентів
Звітність	Формування показників за заявками, угодами, агентами та джерелами звернень	Підтримка управлінських рішень
Журнал дій	Збереження відомостей про створення, зміну або видалення важливих записів	Контроль відповідальності користувачів і підвищення надійності даних

Планування переглядів є окремим важливим процесом, оскільки рієлторська діяльність пов’язана з особистими зустрічами, узгодженням часу між клієнтом, агентом і власником об’єкта. Система має підтримувати створення запису про перегляд, вибір клієнта, прив’язку до об’єкта, зазначення дати та часу, відповідального працівника та результату зустрічі. Після завершення перегляду, агент повинен внести коментар щодо реакції клієнта, причин зацікавленості або відмови [15].

Модуль угод має відображати фінальний етап роботи агентства. До нього варто передавати заявки, за якими клієнт виявив готовність до купівлі, оренди або продажу. Угода повинна містити інформацію про клієнта, об’єкт, агента, ціну, комісію, дату очікуваного завершення, поточну стадію та результат. Для керівника важливо бачити загальну кількість активних угод, прогнозований дохід і ймовірність успішного завершення. Для агента головним є контроль домовленостей та своєчасне виконання наступних дій.

Без зручного пошуку CRM-система швидко перетворюється на пасивне сховище записів, робота з яким потребує зайвого часу. Користувач повинен мати можливість знаходити клієнтів за ПІБ, телефоном, електронною поштою, статусом або відповідальним агентом. Для об’єктів, необхідні фільтри за типом, районом, ціною, площею, кількістю кімнат та статусом доступності.

Вимоги до даних, які обробляються в CRM-системі наведені в таблиці 1.6.

Звітність має забезпечувати керівника агентства узагальненою інформацією про роботу команди. Система повинна формувати показники за кількістю нових заявок, активних клієнтів, проведених переглядів, відкритих угод, завершених операцій і відмов. Додатково, важливо передбачити аналітику за агентами, щоб керівник міг оцінити інтенсивність роботи, результативність комунікації та навантаження. Звітні дані можуть виводитися у вигляді таблиць, діаграм або інформаційних карток на головній панелі [16].

Таблиця 1.6 – Вимоги до даних, які обробляються в CRM-системі

Інформаційна одиниця	**Основні дані**	**Зв’язок з іншими одиницями**
Клієнт	ПІБ, контакти, тип клієнта, джерело звернення, коментарі	Пов’язується з заявками, переглядами та угодами
Об’єкт нерухомості	Тип, адреса, район, площа, ціна, статус, опис, власник	Пов’язується з переглядами, заявками та угодами
Заявка	Тип операції, вимоги, бюджет, статус, відповідальний агент	Поєднує клієнта з потребою та добором об’єктів
Перегляд	Дата, час, клієнт, об’єкт, агент, результат	Відображає практичну взаємодію сторін
Угода	Стадія, ціна, комісія, прогноз завершення, фінальний результат	Завершує процес роботи з заявкою
Користувач	Логін, роль, стан облікового запису, контактні дані	Визначає доступ до операцій і відповідальність за записи

Розмежування доступу є обов’язковою функціональною вимогою для CRM-системи, оскільки база клієнтів та об’єктів має комерційну цінність. Агент повинен мати доступ до робочих записів, потрібних для виконання посадових завдань. Керівник має отримувати ширший огляд даних, зокрема статистику, активність працівників та стан угод. Адміністратор повинен керувати обліковими записами та налаштуваннями, але його дії також мають фіксуватися в журналі.

Для підвищення зручності роботи система повинна мати інтуїтивно зрозумілий інтерфейс з логічним поділом на розділи. Головна панель має відображати ключові показники, до яких входять кількість нових заявок, заплановані перегляди, активні угоди та нагадування. Картки клієнтів, об’єктів і угод мають бути побудовані за єдиною логікою, щоб користувач не витрачав час на пошук потрібних полів.

Функціональні вимоги до розроблюваної CRM-системи орієнтовані на створення цілісного програмного продукту для підтримки основних процесів агентства нерухомості. Майбутня система повинна забезпечити ведення

клієнтів, об’єктів, заявок, переглядів, угод, користувачів і звітності в єдиному інформаційному середовищі.

1.4 Постановка задачі розробки програмного продукту

Постановка задачі розробки CRM-системи для сфери нерухомості ґрунтується на результатах аналізу предметної області, характеристиці роботи агентства з клієнтами та визначених функціональних вимогах. Рієлторська діяльність пов’язана з постійною обробкою значної кількості контактів, заявок, об’єктів, переглядів і угод, тому ефективність агентства залежить від здатності швидко знаходити потрібну інформацію, контролювати поточні домовленості й підтримувати якісну комунікацію з клієнтами.

Головна проблема полягає в розрізненому збереженні інформації, яка використовується працівниками агентства під час щоденної роботи. Частина відомостей може перебувати в електронних таблицях, інша частина, в месенджерах, телефонних нотатках або особистих файлах агентів. Через розподіленість джерел, керівник не завжди має повну картину щодо стану клієнтської бази, ефективності роботи працівників, кількості активних заявок і прогнозованих угод. Працівники витрачають додатковий час на пошук інформації, повторне уточнення потреб клієнтів і ручне оновлення даних про об’єкти нерухомості.

Програмний продукт має забезпечувати реєстрацію користувачів з розмежуванням ролей, ведення клієнтської бази, облік об’єктів нерухомості, опрацювання заявок, планування переглядів, супровід угод і формування управлінської звітності. Система повинна мати зрозумілий інтерфейс, логічну структуру модулів та механізми захисту даних від несанкціонованого доступу [18].

Метою розробки є створення програмного продукту, який забезпечить централізоване керування клієнтами, об’єктами нерухомості, заявками,

переглядами та угодами в діяльності агентства. Досягнення поставленої мети передбачає побудову інформаційної системи з ролями користувачів, базою даних, інтерфейсом для щоденної роботи агентів і засобами контролю для керівника. Результатом має стати працездатна CRM-система, придатна для демонстрації основних сценаріїв роботи рієлторської організації.

Об’єктом автоматизації виступають процеси взаємодії агентства нерухомості з клієнтами під час пошуку, продажу або оренди об’єктів. Предметом розробки є програмні засоби керування клієнтською базою, заявками, об’єктами нерухомості, переглядами й угодами. Зміст програмного продукту визначається потребою об’єднати операційні дії агентів та контрольні функції керівника в одному інформаційному середовищі.

Система повинна підтримувати кілька категорій користувачів. Агент має працювати з клієнтами, заявками, переглядами й об’єктами, які належать до його робочої зони. Керівник повинен отримувати доступ до узагальнених показників, контролювати завантаження працівників і переглядати стан угод. Адміністратор відповідає за користувачів, ролі та службові налаштування. Розмежування прав дає змогу підвищити безпеку даних і зробити роботу персоналу прозорішою. Межі розроблюваного програмного продукту представлені в таблиці 1.7.

Таблиця 1.7 – Межі розроблюваного програмного продукту

Компонент системи	**Передбачене призначення**	**Очікуваний результат**
Модуль авторизації	Перевірка облікових даних і визначення ролі користувача	Захищений вхід до системи та розмежування доступу
Модуль клієнтів	Збереження контактів, потреб, джерел звернення та коментарів	Формування структурованої клієнтської бази
Модуль об’єктів	Облік нерухомості з параметрами, описом, ціною та статусом	Підтримання актуального каталогу пропозицій
Модуль заявок	Реєстрація запитів клієнтів і контроль стадій опрацювання	Упорядковане ведення лідів від першого контакту до результату

Продовження таблиці 1.7

Компонент системи	**Передбачене призначення**	**Очікуваний результат**
Модуль переглядів	Планування зустрічей і фіксація результатів показу об’єктів	Контроль розкладу агентів і збереження реакції клієнтів
Модуль угод	Супровід домовленостей, вартості, комісії та фінального стану	Прозоре керування продажами або орендними операціями
Аналітичний блок	Відображення показників роботи агентства	Підтримка управлінських рішень керівника
Адміністративний блок	Керування користувачами, ролями та довідковими значеннями	Підтримання стабільної роботи системи

Під час постановки задачі важливо визначити вхідні та вихідні дані системи. Вхідними даними виступають відомості про користувачів, клієнтів, об’єкти нерухомості, заявки, перегляди та угоди. Частина інформації вводиться агентами вручну, зокрема контактні дані клієнтів, параметри об’єктів, коментарі після переглядів і стадії домовленостей. Вихідними даними є сформовані картки, результати пошуку, таблиці активних заявок, календар переглядів, перелік угод і звітні показники для керівника. Вхідні та вихідні дані CRM-системи узагальнено в таблиці 1.8.

Таблиця 1.8 – Вхідні та вихідні дані CRM-системи

Категорія даних	**Приклади вхідної інформації**	**Очікувана вихідна інформація**
Користувачі	Логін, пароль, роль, контактні дані, стан облікового запису	Перелік активних працівників і права доступу
Клієнти	ПІБ, телефон, електронна пошта, тип клієнта, бюджет, коментар	Картка клієнта та історія взаємодії
Об’єкти	Тип нерухомості, адреса, район, площа, ціна, статус	Каталог доступних пропозицій з можливістю фільтрації
Заявки	Тип операції, вимоги до об’єкта, відповідальний агент, статус	Робочий список звернень за стадіями опрацювання

Продовження таблиці 1.8

Категорія даних	**Приклади вхідної інформації**	**Очікувана вихідна інформація**
Перегляди	Дата, час, клієнт, об’єкт, агент, результат зустрічі	Розклад показів і журнал результатів переглядів
Угоди	Клієнт, об’єкт, ціна, комісія, стадія, дата завершення	Перелік активних і завершених угод
Звіти	Дані про заявки, перегляди, угоди, агентів і джерела звернень	Аналітичні показники діяльності агентства

Розроблювана система має відповідати вимогам зручності, надійності та логічної узгодженості. Інтерфейс повинен давати змогу швидко переходити між розділами, створювати нові записи, редагувати наявні дані та виконувати пошук без зайвих дій. База даних повинна підтримувати зв’язки між клієнтами, заявками, об’єктами, переглядами та угодами, оскільки ізольоване зберігання записів не забезпечує повноцінної автоматизації рієлторського процесу.

Критеріями успішності виконання задачі є можливість виконання основних сценаріїв роботи агентства без використання сторонніх таблиць або неструктурованих нотаток. Користувач повинен мати змогу увійти до системи, створити картку клієнта, додати об’єкт нерухомості, сформувати заявку, запланувати перегляд, внести його результат та перевести роботу в угоду. Керівник повинен отримати інструменти для перегляду загального стану заявок, результатів роботи агентів і поточних угод. Адміністратор повинен мати засоби для підтримання коректного складу користувачів і ролей.

Розробка повинна охоплювати інформаційну модель, інтерфейс користувача, логіку авторизації, операції з клієнтами та об’єктами, механізми керування заявками, переглядами та угодами, а також засоби отримання звітної інформації.

1.5 Висновки до розділу 1

В першому розділі виконано аналіз предметної області, пов’язаної з автоматизацією діяльності агентства нерухомості. Розглянуто специфіку взаємодії агентів з клієнтами, власниками об’єктів, покупцями, орендарями й керівництвом організації. Рієлторська діяльність потребує постійного збереження великого обсягу взаємопов’язаних даних, серед яких важливе місце займають контакти клієнтів, параметри нерухомості, заявки, перегляди, домовленості, стадії угод і результати комунікації.

Проведений аналіз показав, що використання розрізнених електронних таблиць, телефонних нотаток, месенджерів та окремих файлів не забезпечує належного рівня керованості бізнес-процесів. За відсутності єдиного інформаційного середовища виникають ризики втрати заявок, дублювання клієнтів, несвоєчасного оновлення відомостей про об’єкти, пропуску запланованих переглядів і недостатнього контролю за роботою агентів.

Огляд програмних рішень для автоматизації рієлторської діяльності дав змогу визначити переваги та обмеження наявних CRM-платформ. Універсальні системи забезпечують розвинені інструменти керування лідами, угодами, комунікаціями й аналітикою, однак потребують адаптації до роботи з об’єктами нерухомості. Галузеві рішення краще враховують особливості рієлторського процесу, проте часто орієнтовані на великі команди, іноземний ринок або інвесторські сценарії.

На основі вивчення предметної області сформовано функціональні вимоги до майбутнього програмного продукту. Визначено необхідність реалізації модулів авторизації, керування клієнтами, обліку об’єктів нерухомості, оброблення заявок, планування переглядів, супроводу угод, формування звітності й адміністрування користувачів.

Розроблюваний програмний продукт повинен забезпечити централізовану роботу з клієнтами, заявками, об’єктами, переглядами та угодами, зменшити залежність агентства від ручного ведення даних і підвищити прозорість управління.

2 ПРОЄКТУВАННЯ CRM СИСТЕМИ ДЛЯ СФЕРИ НЕРУХОМОСТІ

2.1 Визначення ролей користувачів та сценаріїв роботи системи

Проєктування CRM-системи для сфери нерухомості потрібно починати з визначення складу користувачів, оскільки саме від ролей залежать доступні функції, структура інтерфейсу, правила обробки даних та логіка подальшого моделювання. Агентство нерухомості працює з різними інформаційними потоками, серед яких особливе значення мають клієнтські контакти, заявки, картки об’єктів, перегляди, угоди, службові коментарі й аналітичні показники.

Розроблювана CRM-система орієнтована на внутрішнє використання працівниками агентства нерухомості. Клієнт не виступає окремим авторизованим користувачем, оскільки взаємодія з ним фіксується через картки клієнтів, заявки, перегляди та угоди. Основними ролями виступають адміністратор, керівник агентства та агент з нерухомості [19]. Кожна роль має власне призначення, рівень відповідальності та набір операцій. Ролі користувачів CRM-системи окреслено в таблиці 2.1.

Адміністратор забезпечує організаційну та технічну основу роботи CRM-системи. Його діяльність пов’язана з обліковими записами працівників, призначенням ролей, підтриманням довідників, контролем активності користувачів і журналюванням службових подій.

Керівник агентства використовує CRM-систему для оцінювання поточного стану діяльності, контролю завантаження працівників і прийняття управлінських рішень. Для нього важливими є не лише окремі картки клієнтів або об’єктів, а й узагальнені показники до яких входить кількість нових заявок, частка активних угод, результативність переглядів, динаміка завершених операцій і продуктивність агентів.

Агент з нерухомості є основним операційним користувачем системи. Він безпосередньо працює з клієнтами, приймає звернення, уточнює потреби,

додає об’єкти, планує перегляди, вносить коментарі після зустрічей і супроводжує угоди. Для агента, CRM-система повинна бути швидким робочим інструментом, а не складним довідковим середовищем. Інтерфейс має забезпечувати просте створення записів, зручний пошук, фільтрацію за параметрами нерухомості та швидкий перехід між пов’язаними даними.

Таблиця 2.1 – Ролі користувачів CRM-системи

Роль користувача	**Призначення в системі**	**Основні операції**	**Обмеження доступу**
Адміністратор	Підтримання працездатності системи та керування службовими параметрами	Створення облікових записів, зміна ролей, блокування користувачів, налаштування довідників, перегляд журналу дій	Не виконує супровід клієнтських угод як робочу операцію агентства
Керівник агентства	Організаційний контроль діяльності працівників і аналіз результатів роботи	Перегляд клієнтів, заявок, об’єктів, переглядів, угод, звітів, активності агентів	Не змінює технічні параметри системи без адміністративних прав
Агент з нерухомості	Виконання щоденних операцій з клієнтами й об’єктами	Додавання клієнтів, створення заявок, ведення карток об’єктів, планування переглядів, оновлення стадій угод	Не має доступу до повного адміністрування користувачів і системних налаштувань

Функціональна взаємодія користувачів з системою описується через сценарії роботи. Сценарій відображає послідовність дій користувача, результат виконання операції та дані, які змінюються після завершення процесу. Для CRM-системи сфери нерухомості важливо передбачити сценарії, які охоплюють повний шлях від входу користувача до завершення

угоди. Основні сценарії роботи користувачів CRM-системи наведено в таблиці 2.2.

Таблиця 2.2 – Основні сценарії роботи користувачів CRM-системи

Сценарій роботи	**Учасник**	**Початкова умова**	**Результат виконання**
Вхід до системи	Адміністратор, керівник, агент	Користувач має активний обліковий запис	Відкрито робочу панель відповідно до ролі
Додавання нового клієнта	Агент	Отримано звернення від потенційного покупця, орендаря або власника	Створено картку клієнта з контактами, потребами та джерелом звернення
Реєстрація об’єкту нерухомості	Агент	Отримано відомості про квартиру, будинок або комерційне приміщення	Додано картку об’єкта з параметрами, ціною, статусом і власником
Створення заявки	Агент	Клієнт повідомив потреби щодо купівлі, продажу або оренди	Заявку прив’язано до клієнта та відповідального агента
Підбір пропозицій	Агент	В системі наявні активні об’єкти та зареєстрована заявка	Сформовано перелік об’єктів, що відповідають вимогам клієнта
Планування перегляду	Агент	Клієнт зацікавився конкретним об’єктом	Записано дату, час, об’єкт, клієнта та відповідального працівника
Фіксація результату перегляду	Агент	Перегляд відбувся або був скасований	Оновлено статус зустрічі та додано коментар щодо подальших дій
Супровід угоди	Агент, керівник	Клієнт готовий перейти до домовленостей	Створено або оновлено угоду зі стадією, ціною та прогнозованим результатом

Продовження таблиці 2.2

Сценарій роботи	**Учасник**	**Початкова умова**	**Результат виконання**
Контроль роботи агентів	Керівник	У системі накопичено дані щодо заявок і угод	Отримано звітні показники за працівниками та стадіями роботи
Адміністрування доступу	Адміністратор	Необхідно додати, змінити або заблокувати користувача	Оновлено обліковий запис і права доступу

Сценарій входу до системи є початковою дією для всіх авторизованих користувачів. Після введення логіну та пароля система перевіряє облікові дані, визначає роль і відкриває відповідну робочу панель. Для агента пріоритетними елементами панелі мають бути клієнти, заявки, об’єкти, перегляди й угоди. Для керівника важливим є доступ до зведених показників і контролю активності працівників. Для адміністратора головними виступають користувачі, ролі, довідники та журнал дій [19].

Сценарій додавання клієнта починається після отримання звернення телефоном, через сайт, соціальні мережі або особисту рекомендацію. Агент вносить контактні дані, зазначає тип клієнта, джерело звернення, бюджет, бажаний район, коментар і поточний статус взаємодії. Після збереження, картка клієнта може використовуватися для створення заявок, планування переглядів та подальшого супроводу угоди.

Сценарій реєстрації об’єкта нерухомості пов’язаний з додаванням квартири, будинку, земельної ділянки або комерційного приміщення до каталогу агентства. Агент заповнює параметри об’єкту, вказує район, площу, кількість кімнат, ціну, стан доступності, власника й опис. Для підвищення інформативності картка може містити фотографії та службові примітки. Після створення запису об’єкт стає доступним для пошуку, фільтрації та прив’язки до заявок або переглядів.

Опрацювання заявки є центральним сценарієм CRM-системи, оскільки саме заявка відображає потребу клієнта. Агент створює заявку на основі картки клієнта, зазначає тип операції, параметри бажаної нерухомості, бюджет, пріоритетність і статус. Далі заявка проходить кілька робочих станів: первинне уточнення, підбір варіантів, організація перегляду, переговори або закриття. Зміна стану заявки дозволяє керівнику бачити реальний перебіг роботи та своєчасно реагувати на затримки.

Сценарій планування перегляду призначений для координації зустрічі між клієнтом, агентом і власником об’єкта. В системі фіксується дата, час, об’єкт, клієнт, відповідальний працівник і початковий статус зустрічі. Після завершення перегляду агент вносить результат, зазначає реакцію клієнта та визначає подальші дії. Наявність даних про перегляди важлива для оцінювання привабливості об’єктів, роботи агентів та причин відмови клієнтів.

Сценарій супроводу угоди відображає перехід від звичайної заявки до фінансово значущого процесу. Якщо клієнт готовий до купівлі, продажу або оренди, агент створює запис угоди, прив’язує клієнта та об’єкт, зазначає попередню ціну, очікувану комісію, стадію та прогнозовану дату завершення. Керівник може відстежувати активні угоди, оцінювати потенційний дохід і контролювати своєчасність виконання домовленостей.

Розмежування сценаріїв за ролями впливає на майбутню архітектуру системи. Для агента потрібно передбачити швидкий доступ до операцій створення та оновлення записів. Для керівника необхідна панель перегляду статистики та контролю поточного стану процесів. Для адміністратора важливими є інструменти керування користувачами та налаштуваннями.

2.2 Побудова архітектури програмного забезпечення

Архітектура програмного забезпечення CRM-системи для сфери нерухомості визначає загальну організацію програмного продукту, розподіл

функцій між його компонентами та правила обміну інформацією між користувацьким інтерфейсом, серверною логікою та базою даних. Для розроблюваної системи потрібно застосувати багаторівневу архітектуру, оскільки вона дозволяє відокремити роботу з інтерфейсом від обробки бізнес-операцій та збереження інформації.

Основу CRM-системи формує клієнт-серверна модель. Користувач працює з інтерфейсом через браузер або локальний клієнтський модуль, надсилаючи запити до серверної частини. Сервер обробляє запити, перевіряє права доступу, виконує бізнес-логіку та звертається до бази даних. База даних зберігає облікові записи користувачів, клієнтів, об’єкти нерухомості, заявки, перегляди, угоди й службові записи. Вказана модель забезпечує централізоване збереження інформації та єдині правила доступу до неї [20]. Рівні архітектури CRM-системи представлено в таблиці 2.3.

Таблиця 2.3 – Рівні архітектури CRM-системи

Рівень архітектури	**Призначення**	**Основні функції**	**Очікуваний результат**
Презентаційний рівень	Забезпечення взаємодії користувача із системою	Відображення форм, таблиць, панелей, повідомлень і результатів пошуку	Зручна робота агента, керівника й адміністратора з даними
Рівень бізнес-логіки	Оброблення операцій і контроль правил роботи	Авторизація, перевірка ролей, створення заявок, оновлення угод, формування звітів	Узгоджене виконання функцій CRM-системи
Рівень доступу до даних	Виконання операцій збереження та отримання інформації	Запити до бази даних, оновлення записів, вибірка пов’язаних даних	Коректна робота з інформаційними об’єктами
Рівень збереження даних	Централізоване зберігання відомостей	Таблиці користувачів, клієнтів, нерухомості, заявок, переглядів і угод	Цілісна база даних для операційної роботи агентства

Презентаційний рівень має забезпечувати доступ до основних робочих екранів системи.

Для агента першочергове значення мають сторінки клієнтів, об’єктів, заявок, переглядів та угод. Для керівника необхідна інформаційна панель із показниками діяльності агентства. Для адміністратора потрібні екрани керування користувачами, ролями та службовими довідниками.

Структура інтерфейсу повинна враховувати роль користувача після входу, щоб кожна категорія працівників отримувала лише потрібні інструменти.

На серверному рівні перевіряється коректність введених даних, визначається можливість виконання операції, встановлюються зв’язки між клієнтом, заявкою, об’єктом та угодою. Наприклад, система не повинна дозволяти створення перегляду без обраного клієнта, об’єкта та відповідального агента.

Рівень доступу до даних виконує роль проміжної ланки між бізнес-логікою і базою даних. Його використання допомагає уникнути прямого звернення інтерфейсу до сховища інформації. Всі операції створення, редагування, пошуку та видалення записів проходять через серверну частину, де здійснюється перевірка прав користувача та контроль коректності операції. В результаті, зменшується ризик несанкціонованої зміни відомостей і забезпечується єдина логіка роботи з даними [21]. Архітектура CRM-системи для сфери нерухомості наведена на рисунку 2.1.

Діаграма архітектури демонструє поділ системи на користувацьку, серверну та інформаційну частини. Кожна роль взаємодіє з окремою робочою панеллю, після чого запити передаються до серверних модулів. Серверна частина містить функціональні блоки, які відповідають за авторизацію, роботу з клієнтами, нерухомістю, заявками, переглядами, угодами, звітністю та адмініструванням. База даних є централізованим сховищем, до якого звертаються лише серверні модулі. Основні модулі серверної частини CRM-системи узагальнено в таблиці 2.4.

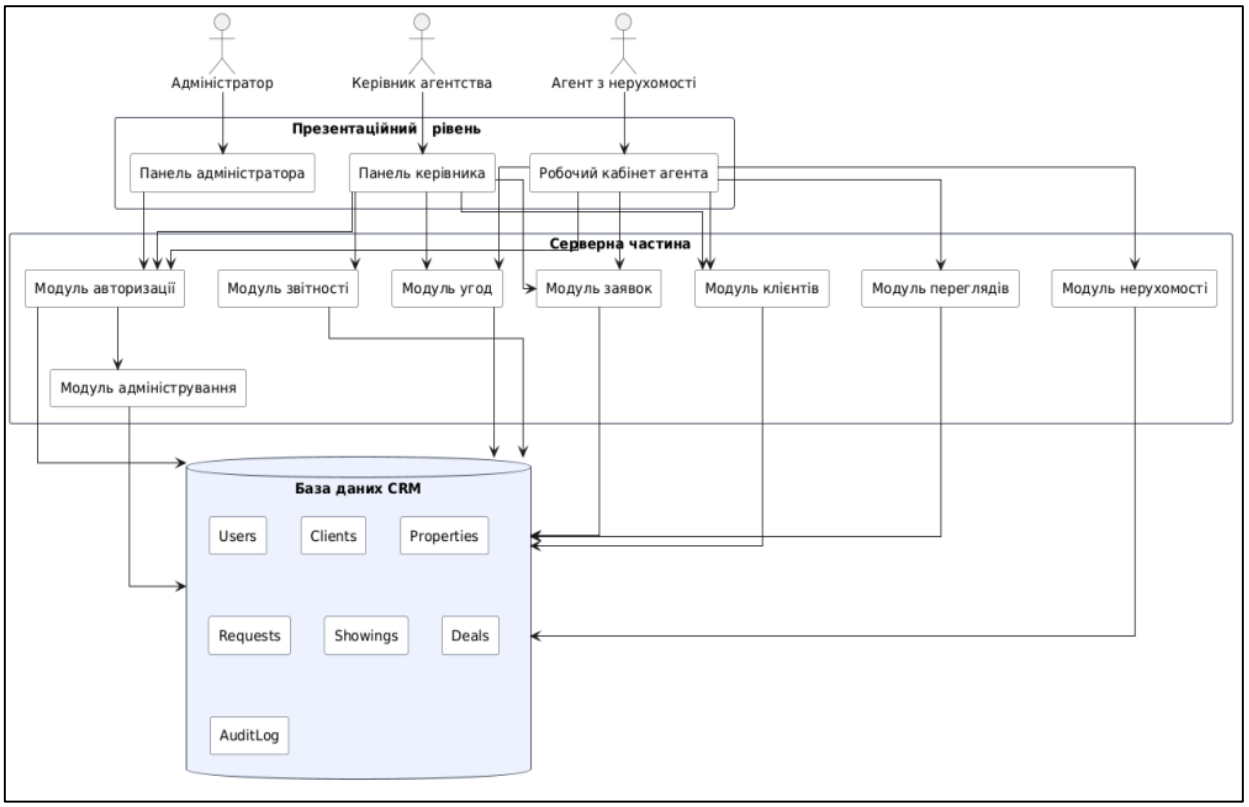


Рисунок 2.1 – Архітектура CRM-системи для сфери нерухомості

Таблиця 2.4 – Основні модулі серверної частини CRM-системи

Модуль	**Призначення**	**Дані, з якими працює модуль**
Модуль авторизації	Перевірка облікових даних і визначення ролі користувача	Користувачі, ролі, статус облікового запису
Модуль клієнтів	Створення та оновлення карток клієнтів	Контакти, потреби, джерела звернення, коментарі
Модуль нерухомості	Ведення каталогу об’єктів агентства	Тип, адреса, район, площа, ціна, статус, власник
Модуль заявок	Реєстрація та контроль потреб клієнтів	Тип операції, бюджет, вимоги, статус, агент
Модуль переглядів	Планування зустрічей і збереження результатів	Дата, час, клієнт, об’єкт, агент, підсумок зустрічі
Модуль угод	Супровід фінальних домовленостей	Стадія, ціна, комісія, дата завершення, результат
Модуль звітності	Підготовка управлінських показників	Заявки, перегляди, угоди, активність агентів
Модуль адміністрування	Керування службовими параметрами системи	Облікові записи, ролі, довідники, журнал дій

Після успішного входу, система визначає роль користувача та надає доступ до відповідних операцій. Агент може створювати та оновлювати робочі записи, керівник переглядає аналітичні показники та контролює процеси, адміністратор керує обліковими записами. Розмежування доступу реалізується на серверному рівні, тому обмеження не залежать лише від відображення кнопок або сторінок в інтерфейсі.

Інформаційна взаємодія між модулями повинна бути побудована на принципі зв’язаних сутностей. Картка клієнта може бути пов’язана з кількома заявками, заявка може мати кілька переглядів, перегляд стосується конкретного об’єкту, а угода поєднує клієнта, об’єкт та відповідального агента [21]. Обмін даними між ключовими компонентами системи представлено в таблиці 2.5.

Таблиця 2.5 – Обмін даними між ключовими компонентами системи

Джерело запиту	**Компонент оброки**	**Операція**	**Результат**
Робочий кабінет агента	Модуль клієнтів	Створення або оновлення картки клієнта	Збережено контактні дані та потреби клієнта
Робочий кабінет агента	Модуль нерухомості	Додавання або зміна об’єкта	Оновлено каталог нерухомості
Робочий кабінет агента	Модуль заявок	Реєстрація нового звернення	Створено заявку з відповідальним агентом
Робочий кабінет агента	Модуль переглядів	Планування зустрічі	Сформовано запис перегляду
Робочий кабінет агента	Модуль угод	Оновлення стадії домовленостей	Змінено стан угоди
Панель керівника	Модуль звітності	Отримання статистики	Виведено показники роботи агентства
Панель адміністратора	Модуль адміністрування	Зміна ролі або блокування користувача	Оновлено параметри доступу

Операції з важливими записами мають супроводжуватися перевіркою обов’язкових полів, контролем коректності введення та фіксацією змін в журналі дій. Журналювання допомагає встановити, хто створив або змінив запис, коли була виконана операція та який модуль використовувався [21].

Поділ на модулі дозволяє розробляти програмний продукт поступово, перевіряти окремі функції незалежно одна від одної та надалі розширювати систему новими можливостями без повної перебудови її структури.

2.3 Моделювання структури даних інформаційної системи

Моделювання структури даних CRM-системи для сфери нерухомості є необхідним етапом проєктування, оскільки від логічної організації бази даних залежить коректність роботи всіх функціональних модулів. Інформаційна система повинна зберігати відомості про користувачів, клієнтів, об’єкти нерухомості, заявки, перегляди, угоди та службові події. Кожна інформаційна одиниця має власний набір атрибутів і водночас, пов’язана з іншими записами, що відображає реальний процес роботи агентства.

Основою інформаційної моделі виступають сутності, які відповідають ключовим об’єктам предметної області. До них належать користувачі системи, клієнти агентства, об’єкти нерухомості, заявки, перегляди, угоди, ролі та журнал дій. Для кожної сутності визначаються поля, первинний ключ, зовнішні ключі та правила збереження [22]. Первинний ключ забезпечує унікальну ідентифікацію запису, а зовнішній ключ підтримує зв’язки між таблицями. Основні сутності бази даних CRM-системи наведено в таблиці 2.6.

Сутність Users призначена для збереження даних авторизованих працівників. Вона містить відомості, потрібні для входу до системи, визначення ролі й контролю активності облікового запису. Паролі не повинні

зберігатися у відкритому вигляді, тому в таблиці передбачається поле для хешованого значення.

Таблиця 2.6 – Основні сутності бази даних CRM-системи

Сутність	**Призначення**	**Основні атрибути**
Users	Збереження облікових записів працівників агентства	Ідентифікатор, ПІБ, електронна пошта, хеш пароля, роль, стан запису, дата створення
Clients	Ведення клієнтської бази агентства	Ідентифікатор, ПІБ, телефон, електронна пошта, тип клієнта, джерело звернення, коментар
Properties	Облік об’єктів нерухомості	Ідентифікатор, тип об’єкта, район, адреса, площа, кількість кімнат, вартість, статус, опис
Requests	Реєстрація потреб клієнтів	Ідентифікатор, клієнт, агент, тип операції, бюджет, вимоги, пріоритет, статус
Showings	Планування та фіксація переглядів	Ідентифікатор, клієнт, об’єкт, агент, дата, час, результат, коментар
Deals	Супровід домовленостей і фінальних операцій	Ідентифікатор, клієнт, об’єкт, агент, стадія, вартість угоди, комісія, дата завершення
AuditLog	Збереження службових подій системи	Ідентифікатор, користувач, дія, таблиця, запис, дата й час виконання
Dictionaries	Підтримання довідкових значень	Ідентифікатор, група значень, назва, стан активності

Сутність Clients формує клієнтську базу агентства. В ній зберігаються контактні дані, тип клієнта, джерело звернення та службові коментарі. Один клієнт може мати кілька заявок, оскільки одна особа здатна одночасно цікавитися купівлею житла та орендою комерційного приміщення.

Сутність Properties містить інформацію про нерухомість, з якою працює агентство. Для кожного об’єкта зберігаються технічні, описові та фінансові характеристики. Статус об’єкта відображає його доступність для продажу або оренди.

Сутність Requests описує потребу клієнта. Вона поєднує клієнта з відповідальним агентом та містить параметри пошуку або продажу нерухомості. Поля бюджету, типу операції, вимог і статусу дозволяють керувати заявкою протягом всього життєвого циклу.

Сутність Showings відображає процес організації переглядів. Вона зв'язує клієнта, об'єкт нерухомості та агента, який відповідає за зустріч. Для кожного перегляду зберігаються дата, час, результат і коментар після завершення.

Сутність Deals призначена для супроводу фінальних домовленостей. Вона містить дані про клієнта, об'єкт, відповідального агента, стадію, вартість угоди, комісію й дату завершення [22]. Зв'язки між сутностями інформаційної моделі представлено в таблиці 2.7.

Таблиця 2.7 – Зв'язки між сутностями інформаційної моделі

Перша сутність	**Друга сутність**	**Тип зв'язку**	**Пояснення**
Users	Requests	Один до багатьох	Один агент може опрацьовувати багато заявок
Users	Showings	Один до багатьох	Один агент може проводити багато переглядів
Users	Deals	Один до багатьох	Один агент може супроводжувати багато угод
Clients	Requests	Один до багатьох	Один клієнт може мати кілька заявок
Clients	Showings	Один до багатьох	Один клієнт може брати участь у кількох переглядах
Clients	Deals	Один до багатьох	Один клієнт може мати кілька угод
Properties	Showings	Один до багатьох	Один об'єкт може переглядатися різними клієнтами
Properties	Deals	Один до багатьох	Один об'єкт може мати історію угод за різні періоди
Users	AuditLog	Один до багатьох	Дії користувача фіксуються в журналі подій

Дані про клієнта, об'єкт, заявку та угоду не повинні неконтрольовано повторюватися в різних таблицях. Наприклад, телефон клієнта зберігається в таблиці Clients, а таблиці Requests, Showings і Deals використовують лише

посилання на клієнта через зовнішній ключ. Аналогічний принцип застосовується до об’єктів нерухомості й облікових записів агентів.

Для забезпечення достовірності даних необхідно передбачити обмеження цілісності. Обов’язковими мають бути поля, без яких запис не може виконувати функціональне призначення. Обмеження цілісності даних в CRM-системі представлено в таблиці 2.8.

Таблиця 2.8 – Обмеження цілісності даних в CRM-системі

Група обмежень	**Зміст обмеження**	**Практичне значення**
Первинні ключі	Кожна таблиця має унікальний ідентифікатор запису	Забезпечення однозначного звернення до об’єкта бази
Зовнішні ключі	Заявки, перегляди, угоди посилаються на клієнтів, агентів	Підтримання логічних зв’язків між робочими процесами
Обов’язкові поля	Критичні атрибути не можуть залишатися порожніми	Запобігання створенню неповних записів
Унікальність	Електронна пошта користувача не повинна повторюватися	Коректна авторизація та уникнення дублювання облікових записів
Контроль статусів	Заявки, об’єкти та угоди мають зберігати допустимі стани	Упорядкування життєвого циклу робочих процесів
Журналювання	Важливі зміни записуються до AuditLog	Підвищення прозорості дій користувачів

ER-діаграма потрібна для перевірки повноти інформаційної моделі перед переходом до фізичного створення бази даних та реалізації серверних маршрутів. ER-діаграма бази даних CRM-системи представлена на рисунку 2.2.

Запропонована ER-діаграма відображає базову інформаційну структуру CRM-системи. Центральними елементами моделі є Clients, Properties і Users, оскільки саме навколо них формуються заявки, перегляди та угоди. Таблиця Requests забезпечує фіксацію потреби клієнта, таблиця Showings підтримує організацію переглядів, а таблиця Deals відповідає за фінальний етап роботи

агентства. AuditLog доповнює модель службовою інформацією, потрібною для контролю змін.

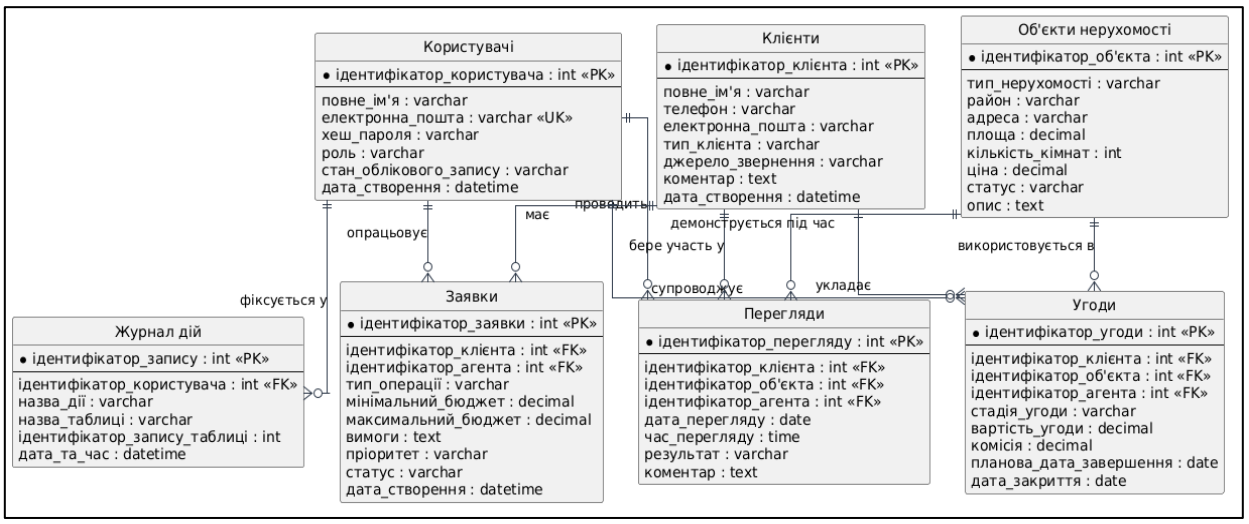


Рисунок 2.2 – ER-діаграма бази даних CRM-системи

Під час фізичної реалізації бази даних потрібно врахувати типи полів. Ідентифікатори потрібно зберігати як цілі числа з автоматичним збільшенням. Грошові значення, площа та комісія мають використовувати числовий тип з фіксованою точністю. Для дат і часу переглядів потрібні окремі типи, які дозволяють виконувати пошук за періодом.

Модель даних повинна підтримувати подальше розширення програмного продукту. При розширенні CRM-системи до неї можна додати таблиці для фотографій об’єктів, документів, повідомлень, нагадувань або інтеграцій із рекламними майданчиками.

2.4 Розробка UML-діаграм для опису логіки взаємодії

Для системи автоматизації рієлторської діяльності необхідно відобразити ролі користувачів, послідовність виконання операцій, внутрішню структуру класів та логіку проходження основних бізнес-процесів. Графічне представлення знижує ризик неоднозначного трактування вимог та

допомагає узгодити майбутню реалізацію з функціональними потребами агентства нерухомості.

Діаграма прецедентів показує доступні функції для адміністратора, керівника та агента. Діаграма послідовності деталізує порядок взаємодії під час створення заявки. Діаграма активності відображає логіку обробки звернення клієнта. Діаграма класів демонструє програмну структуру основних об’єктів системи [23].

Для CRM-системи сфери нерухомості головними акторами виступають адміністратор, керівник агентства та агент з нерухомості. Агент виконує операційні дії з клієнтами, заявками, об’єктами, переглядами та угодами. Керівник переглядає звітність, контролює стан угод і оцінює активність працівників. Адміністратор відповідає за користувачів, ролі, довідники та журнал дій. Діаграма прецедентів CRM-системи наведена на рисунку 2.3.

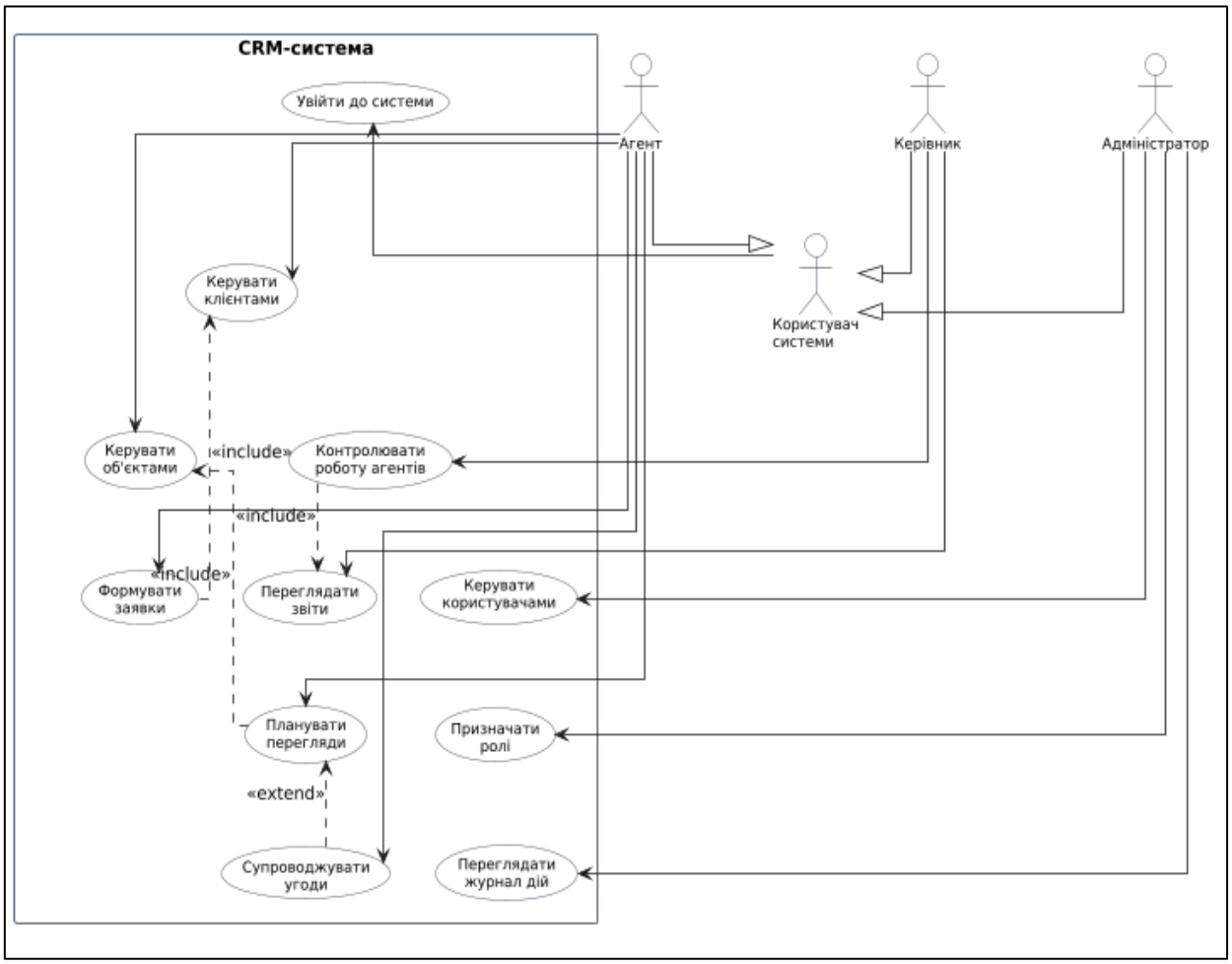


Рисунок 2.3 – Діаграма прецедентів CRM-системи

Діаграма прецедентів показує, що всі користувачі починають роботу з авторизації, але після входу отримують різні можливості. Агент має доступ до операцій, пов’язаних з щоденною рієлторською роботою. Керівник отримує переважно контрольні та аналітичні функції. Адміністратор працює зі службовими параметрами, які забезпечують стабільність системи. Включення прецеденту керування клієнтами до створення заявки пояснюється тим, що заявка не може існувати без прив’язки до клієнта. Зв’язок між переглядами та угодами відображає можливість переходу від проведеної зустрічі до етапу фінальної домовленості.

Для деталізації операції створення заявки використовується діаграма послідовності. Вона показує, як агент взаємодіє з інтерфейсом, серверною частиною та базою даних. Основний сценарій передбачає вибір клієнта, введення параметрів потреби, перевірку даних, збереження заявки й оновлення робочої панелі. Важливим елементом є перевірка прав доступу, оскільки створення заявок повинно виконуватися лише авторизованим користувачем з відповідною роллю. Діаграма послідовності створення заявки клієнта наведена на рисунку 2.4.

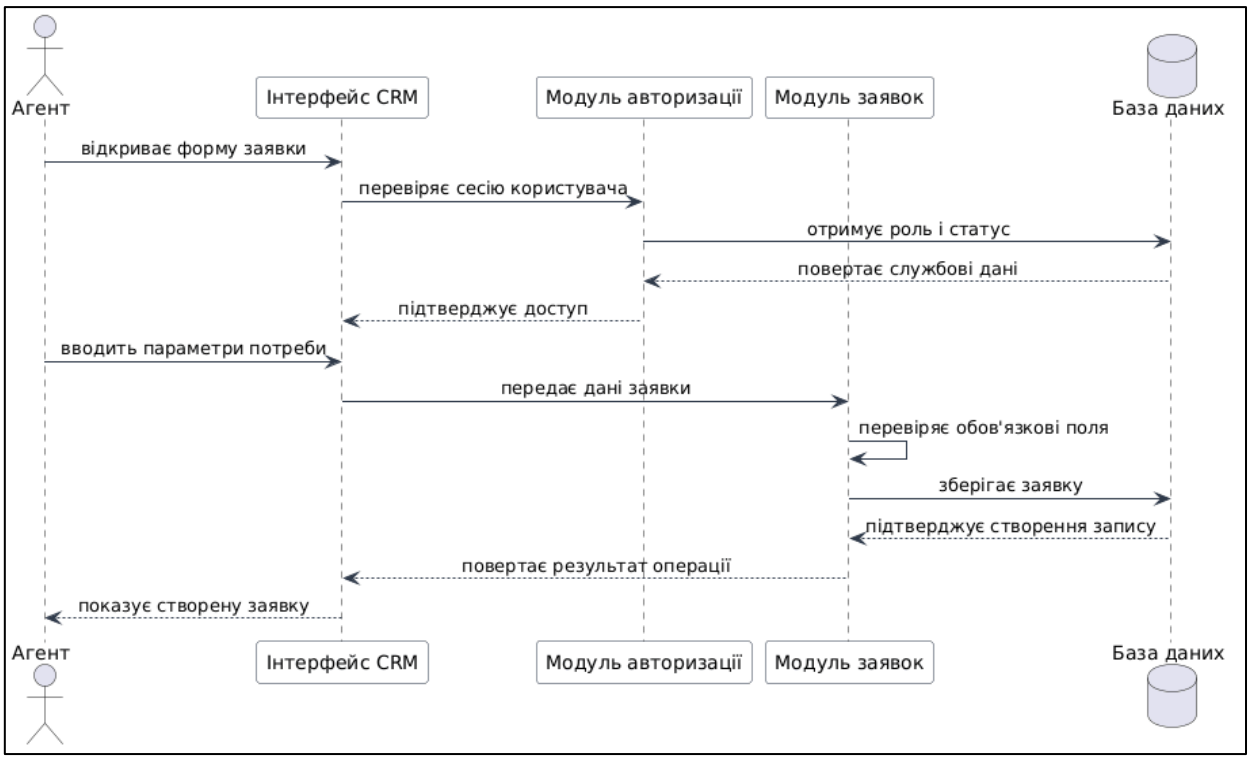


Рисунок 2.4 – Діаграма послідовності створення заявки клієнта

Інтерфейс лише приймає дані від агента та передає їх до серверної частини. Модуль авторизації перевіряє можливість виконання операції. Модуль заявок контролює повноту введених відомостей і виконує збереження через базу даних.

Для опису життєвого циклу заявки потрібно використати діаграму активності. Вона відображає шлях звернення від першого контакту клієнта до завершення роботи. Заявка може завершитися успішною угодою або відмовою, тому процес повинен передбачати кілька варіантів розвитку. Агент уточнює потреби, підбирає об’єкти, організовує перегляд і фіксує результат. Якщо клієнт зацікавлений, робота переходить до угоди. Якщо пропозиції не відповідають очікуванням, заявка повертається на етап підбору або закривається. Діаграма активності обробки заявки клієнта наведена на рисунку 2.5.

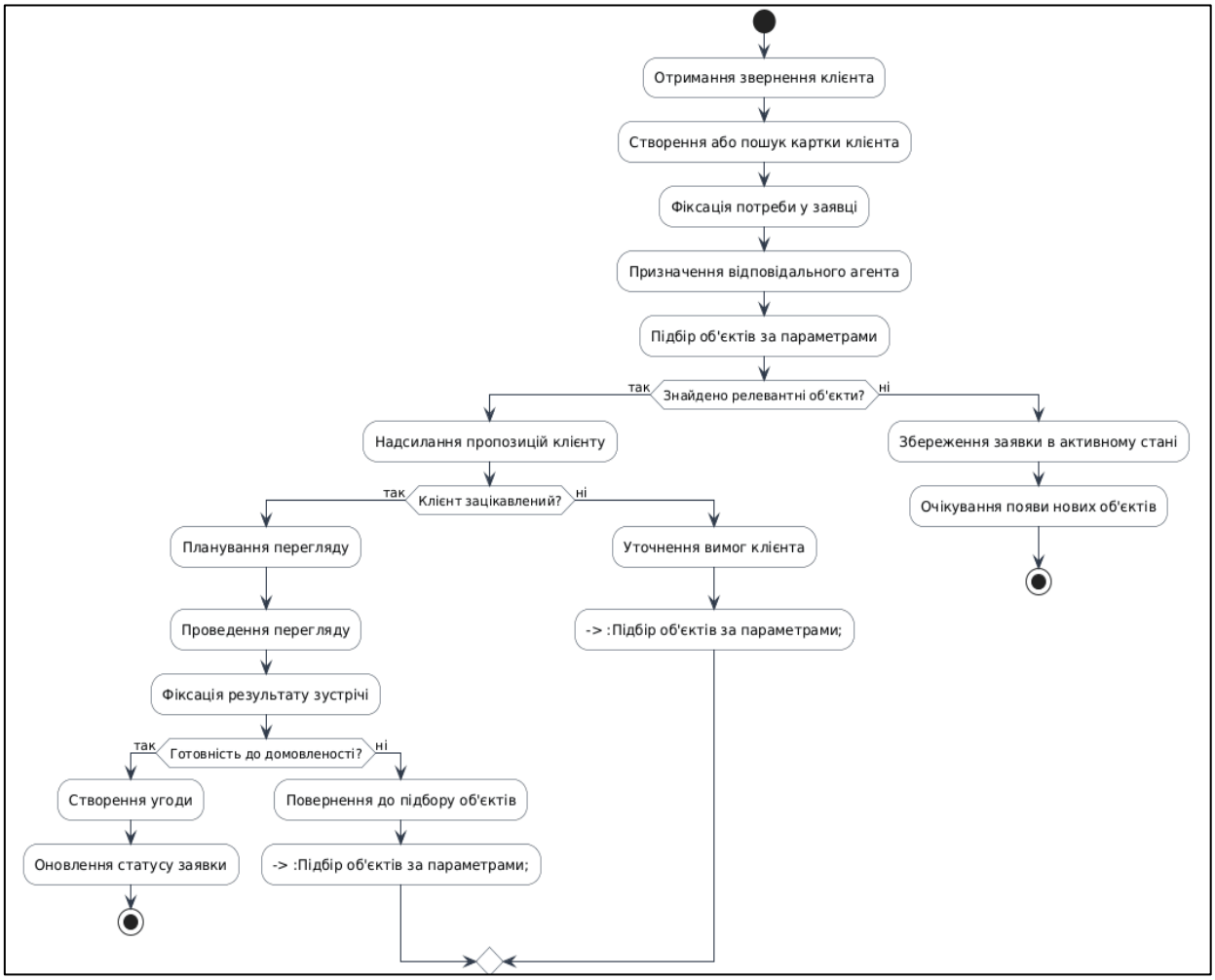


Рисунок 2.5 – Діаграма активності обробки заявки клієнта

Заявка проходить кілька стадій, кожна з яких має практичне значення для агента й керівника. Система повинна підтримувати повернення до попередніх етапів, оскільки клієнт може змінити бюджет, район або вимоги після перегляду. Одночасно потрібна можливість закриття заявки без угоди, якщо клієнт відмовився від співпраці або тимчасово припинив пошук.

Діаграма класів потрібна для переходу від логічної моделі даних до програмної структури майбутньої реалізації. Вона відображає основні класи, їхні атрибути, методи та зв'язки. Для CRM-системи, базовими класами виступають User, Client, Property, Request, Showing, Deal і AuditLog. Класи відповідають основним сутностям бази даних, але додатково містять методи, які виконують операції над даними. Наприклад, клас Request має метод зміни статусу, клас Showing, метод фіксації результату, а клас Deal, метод завершення угоди. Діаграма класів CRM-системи наведена на рисунку 2.6.

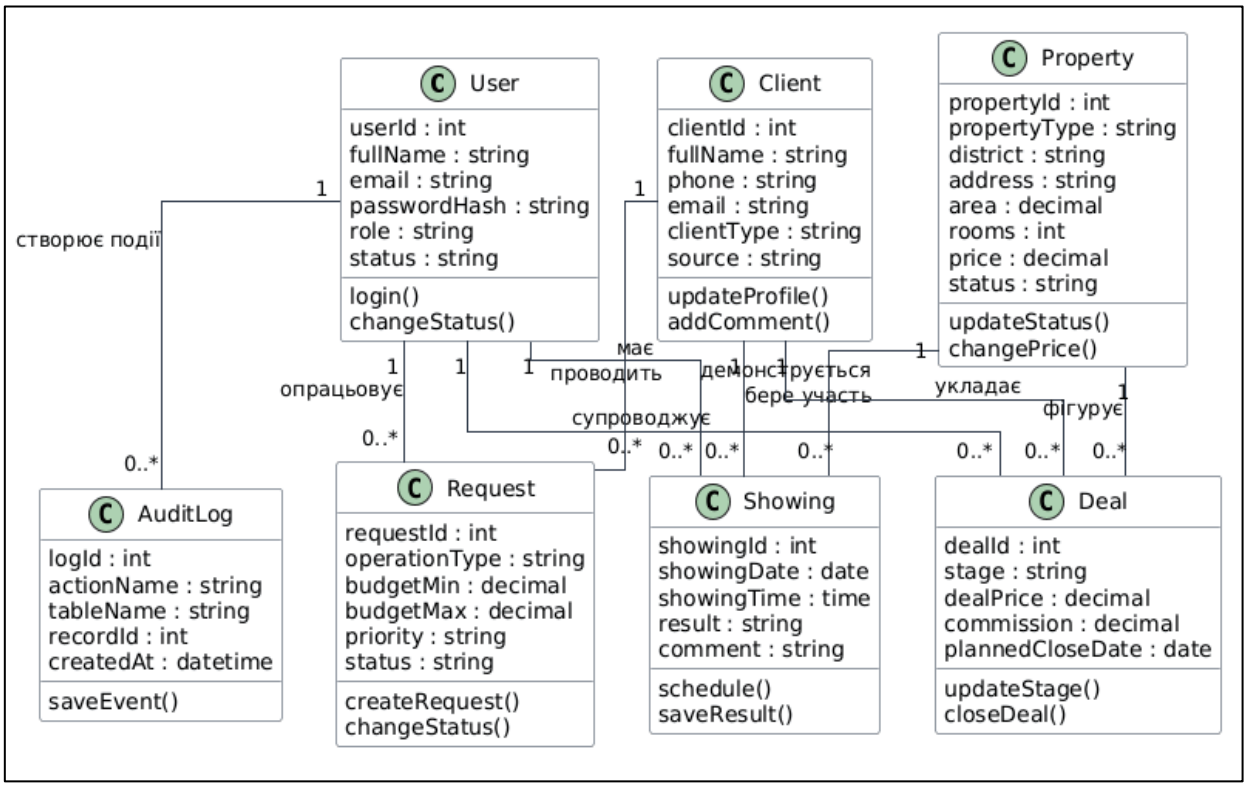


Рисунок 2.6 – Діаграма класів CRM-системи

Діаграма класів узгоджується з ER-моделлю, але має інше проєктне призначення. ER-діаграма описує структуру бази даних, а діаграма класів

показує програмні об’єкти, які будуть використані під час реалізації серверної логіки. Зв’язок між класами Client і Request відображає можливість створення кількох заявок для одного клієнта. Зв’язок між Property і Showing показує, що один об’єкт може демонструватися різним клієнтам. Зв’язок між User і AuditLog потрібний для фіксації дій працівників у системі.

Розроблені UML-діаграми формують проєктну основу майбутньої CRM-системи. Діаграма прецедентів визначає функціональні можливості користувачів і підтверджує необхідність розмежування ролей. Діаграма послідовності уточнює порядок виконання операції створення заявки. Діаграма активності описує реальний бізнес-процес оброблення звернення клієнта в агентстві нерухомості. Діаграма класів переводить логічну структуру даних у формат, придатний для програмної реалізації.

2.5 Висновки до розділу 2

В другому розділі виконано проєктування CRM-системи для сфери нерухомості з урахуванням функціональних вимог, визначених під час аналізу предметної області. Основну увагу приділено ролям користувачів, архітектурній організації програмного забезпечення, структурі даних і UML-моделюванню логіки взаємодії. Проєктні рішення спрямовано на створення програмного продукту, здатного підтримувати операційну роботу агентства, забезпечувати контроль заявок, об’єктів, переглядів і угод, а також, надавати керівництву інструменти для оцінки результатів діяльності.

В процесі визначення ролей користувачів обґрунтовано розподіл системних можливостей між адміністратором, керівником агентства та агентом з нерухомості. Адміністратор відповідає за службові налаштування, облікові записи та права доступу. Керівник використовує систему для контролю роботи працівників, перегляду стану угод і аналізу показників діяльності. Агент виконує основні операції з клієнтами, об’єктами, заявками,

переглядами та угодами. Розмежування повноважень дає змогу підвищити безпеку даних, упорядкувати робочі процеси та уникнути неконтрольованого доступу до комерційно важливої інформації.

Архітектуру програмного забезпечення побудовано за клієнт-серверною моделлю з поділом на презентаційний рівень, рівень бізнес-логіки, рівень доступу до даних і рівень збереження інформації. В структурі системи передбачено модулі авторизації, клієнтів, нерухомості, заявок, переглядів, угод, звітності й адміністрування.

Результати проєктування підтвердили, що CRM-система для агентства нерухомості повинна мати цілісну інформаційну модель, чітку архітектуру, контроль доступу та узгоджені сценарії роботи користувачів. Розроблені проєктні матеріали створюють основу для переходу до програмної реалізації, вибору технологічного стеку, побудови бази даних, створення клієнтського інтерфейсу та перевірки працездатності основних функцій застосунку.

3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ CRM-СИСТЕМИ

3.1 Обґрунтування вибору технологічного стеку розробки

Вибір технологічного стеку для CRM-системи сфери нерухомості має спиратися на функціональні вимоги, архітектурну модель, складність майбутньої реалізації та можливість демонстрації готового програмного продукту під час захисту.

Для реалізації CRM-системи доречно використати веборієнтований стек, побудований на Node.js, Express.js, SQLite, HTML, CSS і JavaScript.

Node.js є відкритим кросплатформним середовищем виконання JavaScript, яке призначене для створення серверів, вебзастосунків, командних інструментів та скриптів [24].

Express.js визначається як мінімалістичний та гнучкий фреймворк для Node.js, що надає набір засобів для вебзастосунків і API [25].

SQLite є самодостатнім SQL-рушієм без окремого серверного процесу, що робить його придатним для програмного продукту з локальним збереженням даних [26].

HTML, CSS і JavaScript утворюють основу клієнтської частини вебзастосунку, оскільки HTML задає структуру сторінки, CSS відповідає за візуальне оформлення, а JavaScript забезпечує динамічну поведінку інтерфейсу [27].

Технологічний стек програмного продукту описано в таблиці 3.1.

Вибір Node.js для серверної частини пояснюється потребою створити вебзастосунок, який швидко обробляє запити користувачів та підтримує логіку взаємодії з базою даних. Для CRM-системи серверна частина повинна виконувати авторизацію, перевірку ролей, створення клієнтів, оновлення об’єктів, реєстрацію заявок, планування переглядів і супровід угод. Використання JavaScript на сервері та в браузері спрощує розробку, оскільки програмна логіка створюється в єдиній мовній екосистемі.

Таблиця 3.1 – Технологічний стек програмного продукту

Компонент	**Обрана технологія**	**Призначення в CRM-системі**	**Обґрунтування вибору**
Серверне середовище	Node.js	Виконання серверної логіки та оброблення HTTP-запитів	Єдина мова JavaScript для клієнтської й серверної частин зменшує складність розробки
Серверний фреймворк	Express.js	Побудова маршрутів, оброблення форм, підключення middleware, реалізація API	Мінімалістична структура добре підходить для дипломного застосунку з чіткою модульною логікою
База даних	SQLite	Збереження клієнтів, об’єктів, заявок, переглядів, угод і користувачів	Локальний файл бази спрощує перенесення проєкту й перевірку працездатності
Клієнтська розмітка	HTML	Формування сторінок, таблиць, форм і робочих панелей	Стандартна вебструктура забезпечує сумісність з браузерами
Стилізація інтерфейсу	CSS	Оформлення панелей, кнопок, таблиць, карток і форм	Дає змогу створити зручний інтерфейс без підключення складних UI-бібліотек
Клієнтська логіка	JavaScript	Перевірка форм, динамічне оновлення окремих елементів, робота з подіями	Підвищує зручність взаємодії користувача із застосунком
Система шаблонів	EJS або аналогічний механізм рендерингу	Формування HTML-сторінок на основі серверних даних	Дозволяє виводити таблиці клієнтів, об’єктів, заявок і угод без дублювання розмітки

Express.js обрано як фреймворк для організації маршрутів та серверних модулів. В CRM-системі кожен функціональний блок може мати власний набір маршрутів, клієнти, об’єкти, заявки, перегляди, угоди, звіти та адміністрування.

SQLite варто використати як базу даних, оскільки вона не потребує окремого серверного встановлення та може зберігатися в одному файлі разом з проєктом. Структура бази даних має містити таблиці користувачів, клієнтів, об’єктів нерухомості, заявок, переглядів, угод і журналу дій.

Клієнтська частина на основі HTML, CSS і JavaScript дозволяє створити інтерфейс, зрозумілий для агента, керівника та адміністратора. HTML використовується для побудови сторінок, таблиць і форм введення. CSS забезпечує візуальну єдність елементів, коректні відступи, читабельність таблиць і зручне розташування робочих блоків. JavaScript застосовується для клієнтської перевірки форм, реакції на дії користувача та динамічного оновлення окремих частин інтерфейсу без зайвого ускладнення архітектури. Відповідність технологій функціональним вимогам CRM-системи узагальнена в таблиці 3.2.

Таблиця 3.2 – Відповідність технологій функціональним вимогам CRM-системи

Функціональна вимога	**Технологічне забезпечення**	**Очікуваний результат**
Авторизація користувачів	Node.js, Express.js, сесії, хешування паролів	Захищений вхід і розмежування доступу за ролями
Ведення клієнтської бази	Express.js, SQLite, HTML-форми	Створення, редагування, пошук і перегляд карток клієнтів
Облік об’єктів нерухомості	SQLite, серверні маршрути, таблиці інтерфейсу	Збереження параметрів нерухомості та швидка фільтрація записів
Реєстрація заявок	Express.js, SQLite, серверна перевірка даних	Контроль потреб клієнтів і призначення відповідальних агентів
Планування переглядів	HTML-форми, SQLite, JavaScript-перевірка дат	Фіксація зустрічей і результатів демонстрації об’єктів
Супровід угод	Серверні маршрути, база даних, статуси	Відстеження стадій домовленостей і фінального результату
Формування звітності	SQL-запити, серверна агрегація даних, таблиці інтерфейсу	Отримання показників за агентами, заявками, переглядами й угодами

Продовження таблиці 3.2

Функціональна вимога	**Технологічне забезпечення**	**Очікуваний результат**
Адміністрування	Express.js, рольова перевірка, журнал дій	Керування користувачами та контроль службових операцій

Важливим аргументом на користь обраного стеку є можливість чіткого поділу проєкту на логічні частини. Серверний код може розміщуватися в окремих файлах маршрутів, моделі даних описуються через SQL-структуру, статичні ресурси інтерфейсу зберігаються в окремій папці, а шаблони сторінок формують візуальний шар застосунку.

Окремої уваги потребує безпека програмного продукту. CRM-система опрацьовує клієнтські контакти, коментарі агентів, фінансові параметри угод і службову інформацію, тому доступ до неї має бути контрольованим. На рівні серверної частини потрібно передбачити перевірку авторизації перед відкриттям захищених сторінок, обмеження дій відповідно до ролі, хешування паролів, валідацію введених даних і журналювання важливих операцій. Реалізація вказаних механізмів можлива засобами Node.js, Express.js і допоміжних бібліотек, які підключаються через менеджер пакетів.

Для реалізації CRM-системи обраний стек є достатнім за функціональними можливостями та виправданим за складністю. Він дозволяє створити повноцінний вебзастосунок з ролями користувачів, базою даних, серверною логікою, робочими формами, таблицями, звітністю й базовими засобами захисту.

3.2 Реалізація серверної частини програмного продукту

Серверний рівень відповідає за запуск вебзастосунку, обробку HTTP-запитів, перевірку авторизації, розмежування доступу за ролями, взаємодію з

базою даних SQLite, журналювання дій користувачів і передавання підготовлених даних до шаблонів інтерфейсу.

В програмному продукті використано файлову структуру, яка відокремлює основний файл запуску, модулі маршрутів, middleware авторизації, шаблони сторінок, статичні ресурси та модуль бази даних. Вказана побудова спрощує супровід коду, оскільки кожен функціональний блок системи має окрему зону відповідальності. Файл src/app.js виконує роль точки входу до застосунку, файл src/db.js відповідає за створення та обслуговування бази даних, папка src/routes містить маршрути предметних модулів, а папка src/middleware використовується для службових перевірок доступу. Структура серверної частини CRM-системи наведена в таблиці 3.3.

Таблиця 3.3 – Структура серверної частини CRM-системи

Елемент проєкту	**Призначення**	**Роль в роботі серверної частини**
package.json	Опис залежностей і команд запуску	Дає змогу встановити бібліотеки, запустити сервер і виконати скидання бази
src/app.js	Головний файл вебзастосунку	Налаштовує Express, сесії, шаблонізатор, статичні файли та маршрути
src/db.js	Модуль бази даних	Створює SQLite-базу, таблиці, стартові записи та службові функції запитів
src/middleware/auth.js	Middleware авторизації	Перевіряє вхід користувача, роль і передає дані сесії до шаблонів
src/routes/auth.js	Маршрути входу й виходу	Обробляє авторизацію, перевіряє пароль і створює сесію користувача
src/routes/clients.js	Модуль клієнтів	Забезпечує створення, редагування, перегляд, пошук і архівацію клієнтів
src/routes/properties.js	Модуль нерухомості	Підтримує роботу з картками об’єктів і статусами доступності
src/routes/requests.js	Модуль заявок	Реалізує створення, оновлення та закриття звернень клієнтів
src/routes/showings.js	Модуль переглядів	Зберігає заплановані зустрічі та результати демонстрації об’єктів
src/routes/deals.js	Модуль угод	Дає змогу створювати угоди, змінювати стадії та переглядати деталі домовленостей

Продовження таблиці 3.3

Елемент проєкту	**Призначення**	**Роль в роботі серверної частини**
src/routes/reports.js	Модуль звітності	Формує агреговані показники для керівника й адміністратора
src/routes/admin.js	Адміністративний модуль	Керує обліковими записами й журналом дій

Запуск серверної частини виконується через команду npm start, яка вказана у файлі package.json і викликає node src/app.js. Перед стартом застосунок ініціалізує базу даних, створює таблиці за їх відсутності та додає стартові демонстраційні записи, якщо таблиця користувачів порожня. Після успішної ініціалізації сервер відкриває порт 3000 або інший порт, переданий через змінну середовища PORT. В консоль виводиться повідомлення з адресою локального запуску, що підтверджує готовність серверної частини до приймання запитів.

Головний файл src/app.js містить конфігурацію Express-застосунку. В ньому підключаються модулі path, express, express-session, функція ініціалізації бази даних і middleware для передачі відомостей про авторизованого користувача в шаблони. У файлі налаштовано шаблонізатор EJS, папку подань, оброблення даних із форм, статичні файли, сесії та маршрути всіх функціональних модулів. Окремо передбачено обробку запитів до неіснуючих сторінок і серверних помилок, що підвищує стабільність роботи застосунку під час некоректних дій користувача. Серверні маршрути CRM-системи наведено в таблиці 3.4.

Таблиця 3.4 – Серверні маршрути CRM-системи

Базовий маршрут	**Файл реалізації**	**Функціональне призначення**
/login	auth.js	Вхід користувача до системи
/logout	auth.js	Завершення сесії користувача
/dashboard	dashboard.js	Відображення головної панелі з підсумковими показниками

Продовження таблиці 3.4

Базовий маршрут	**Файл реалізації**	**Функціональне призначення**
/clients	clients.js	Робота з клієнтами агентства
/properties	properties.js	Робота з об’єктами нерухомості
/requests	requests.js	Реєстрація та супровід заявок
/showings	showings.js	Планування переглядів і внесення результатів
/deals	deals.js	Супровід угод і фінансових параметрів
/reports	reports.js	Формування звітних показників
/admin	admin.js	Керування користувачами й перегляд журналу дій

Робота з даними реалізована у файлі src/db.js. На початку модуля створюється папка src/database, після чого відкривається файл crm.sqlite. Для виконання SQL-запитів описано допоміжні функції run, get і all, які використовуються всіма маршрутами. Функція initializeDatabase активує зовнішні ключі, створює таблиці користувачів, клієнтів, об’єктів, заявок, переглядів, угод і журналу дій. За відсутності користувачів викликається заповнення стартовими даними, що дає змогу одразу перевірити роботу системи після першого запуску.

Авторизація реалізована через файл src/routes/auth.js і middleware src/middleware/auth.js. Під час входу, користувач вводить електронну пошту та пароль, після чого сервер знаходить обліковий запис в таблиці users, перевіряє активний стан і порівнює введений пароль з хешем, збереженим в базі. Для порівняння, використовується бібліотека bcryptjs, тому пароль не зберігається у відкритому вигляді. Після успішної перевірки в сесію записуються ідентифікатор, ім’я, електронна пошта та роль користувача. Форма входу не містить підказок з даними адміністратора, керівника або агента, завдяки чому службові облікові дані не демонструються безпосередньо на сторінці авторизації.

Middleware requireLogin перевіряє наявність активної сесії. Якщо користувач не увійшов до системи, сервер повертає його на сторінку входу.

Middleware requireRole додатково перевіряє роль користувача та блокує доступ до сторінок, які не відповідають його повноваженням. Наприклад, маршрути звітності доступні для адміністратора та керівника, а адміністративний модуль доступний лише для адміністратора. Завдяки перевіркам на серверному рівні обмеження доступу не залежать тільки від видимості кнопок у меню.

Функціональні маршрути серверної частини реалізують операції з основними сутностями CRM-системи. В модулі клієнтів передбачено перегляд загального списку, пошук, створення, редагування, перегляд детальної картки та архівацію запису. Модуль нерухомості працює з об’єктами, їх параметрами, ціною, адресою, статусом та відповідальним агентом. Модуль заявок зберігає потреби клієнтів, тип операції, бюджет, район, кількість кімнат, пріоритет і стан опрацювання. Модуль переглядів фіксує дату, час, клієнта, об’єкт, агента та результат зустрічі. Модуль угод забезпечує створення запису про домовленість, зміну стадії, збереження ціни, комісії та планової дати завершення.

В кожному модулі після успішного створення або оновлення важливого запису викликається функція logAction. Вона додає запис до таблиці audit_logs, де зберігаються користувач, назва дії, таблиця, ідентифікатор запису, службовий опис і дата виконання. Журнал дій особливо важливий для CRM-системи, оскільки клієнтські контакти та домовленості мають комерційну цінність. Наявність журналу дозволяє адміністратору відстежувати входи до системи, створення користувачів, оновлення клієнтів, зміну об’єктів та інші важливі операції.

Звітність реалізована в окремому модулі, доступ до якого обмежено ролями адміністратора та керівника. Сервер формує підсумкові дані за клієнтами, об’єктами, заявками, переглядами, угодами й агентами за допомогою SQL-запитів. Агрегування на серверному рівні зменшує навантаження на інтерфейс, оскільки агрегування виконується на серверному рівні. Користувач отримує вже підготовлені показники, які можуть бути виведені у вигляді інформаційних карток або таблиць.

3.3 Створення клієнтського інтерфейсу CRM системи

Клієнтський інтерфейс CRM-системи розроблено з орієнтацією на щоденну роботу агентства нерухомості, де користувачеві потрібно швидко переходити між клієнтами, об’єктами, заявками, переглядами, угодами та звітами. Інтерфейсна частина побудована на основі HTML, CSS, JavaScript і шаблонів EJS, які формуються сервером після отримання даних з бази SQLite [28].

Під час створення інтерфейсу головну увагу приділено зрозумілій навігації, візуальному поділу функціональних блоків та зменшенню кількості зайвих дій під час роботи з даними. Важливо, щоб агент міг швидко створити клієнта, додати об’єкт, сформувати заявку або запланувати перегляд без переходу через складну багаторівневу структуру меню. Керівник повинен бачити підсумкові показники діяльності, а адміністратор мати доступ до службових розділів керування користувачами та журналом дій. Основні сторінки клієнтського інтерфейсу CRM-системи узагальнено в таблиці 3.5.

Таблиця 3.5 – Основні сторінки клієнтського інтерфейсу CRM-системи

Сторінка інтерфейсу	**Призначення**	**Основні елементи**
Сторінка входу	Авторизація користувача в системі	Поля електронної пошти й пароля, кнопка входу, повідомлення про помилки
Головна панель	Узагальнене відображення стану роботи	Інформаційні картки, швидкі переходи, показники заявок і угод
Клієнти	Робота з клієнтською базою	Таблиця клієнтів, пошук, кнопки додавання, редагування й перегляду
Об’єкти	Ведення каталогу нерухомості	Перелік об’єктів, статуси, ціна, район, параметри, службові дії
Заявки	Оброблення звернень клієнтів	Статуси, пріоритети, відповідальний агент, зв’язок із клієнтом

Продовження таблиці 3.5

Сторінка інтерфейсу	**Призначення**	**Основні елементи**
Перегляди	Планування зустрічей і фіксація результатів	Дата, час, клієнт, об’єкт, агент, підсумок перегляду
Угоди	Супровід фінальних домовленостей	Стадія, вартість, комісія, планова дата завершення
Звіти	Аналітичне відображення роботи агентства	Показники за агентами, заявками, угодами й об’єктами
Адміністрування	Керування користувачами та журналом дій	Облікові записи, ролі, статуси, історія операцій

Сторінка входу є початковим елементом взаємодії користувача з CRM-системою. Вона містить лише необхідні поля для введення електронної пошти та пароля. Дані демонстраційних облікових записів не розміщуються у вікні авторизації. Якщо користувач вводить некоректні дані або обліковий запис заблоковано, система повертає повідомлення про помилку без розкриття зайвих службових подробиць. Сторінка авторизації користувача CRM-системи наведена на рисунку 3.1.

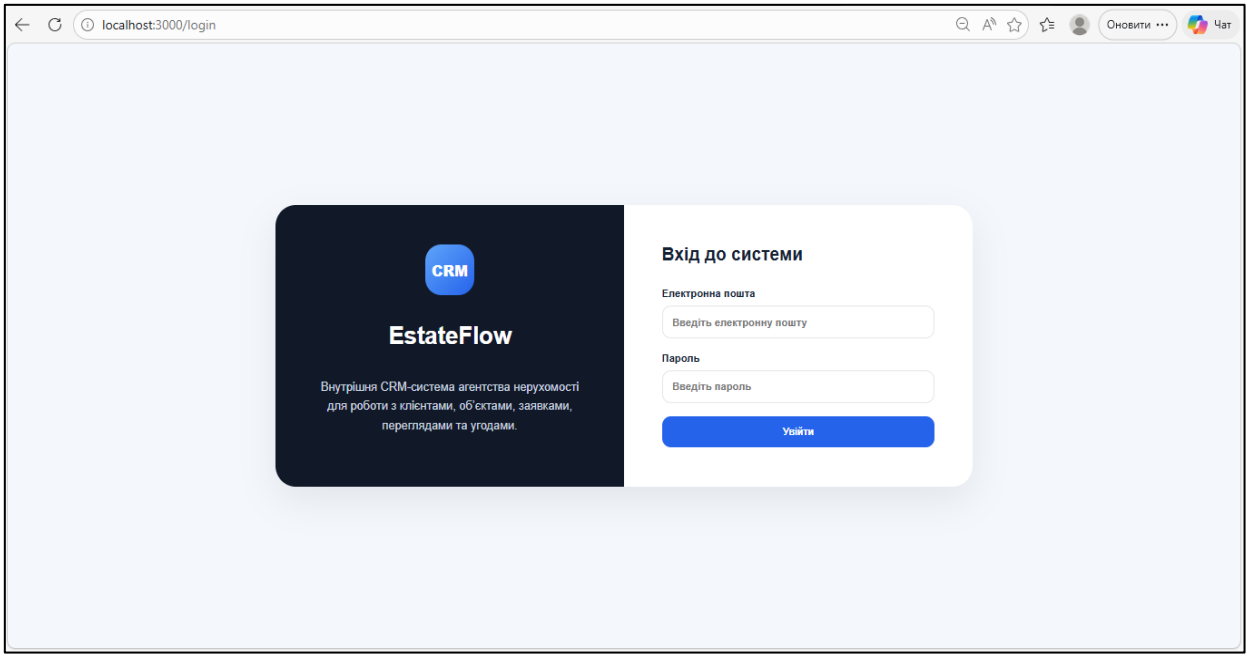


Рисунок 3.1 – Сторінка авторизації користувача CRM-системи

Після успішного входу користувач переходить на головну панель. Її зміст залежить від ролі працівника. Для агента найбільш важливими є швидкі переходи до клієнтів, об’єктів, заявок, переглядів і угод. Для керівника на перший план виходять підсумкові показники, які дають змогу оцінити стан роботи агентства. Для адміністратора доступними є службові елементи, пов’язані з користувачами та журналом дій. Візуально головна панель побудована через картки з числовими показниками та навігаційні блоки, що скорочує час переходу до потрібної операції.

Головна панель CRM-системи після входу користувача після входу користувача представлена на рисунку 3.2.

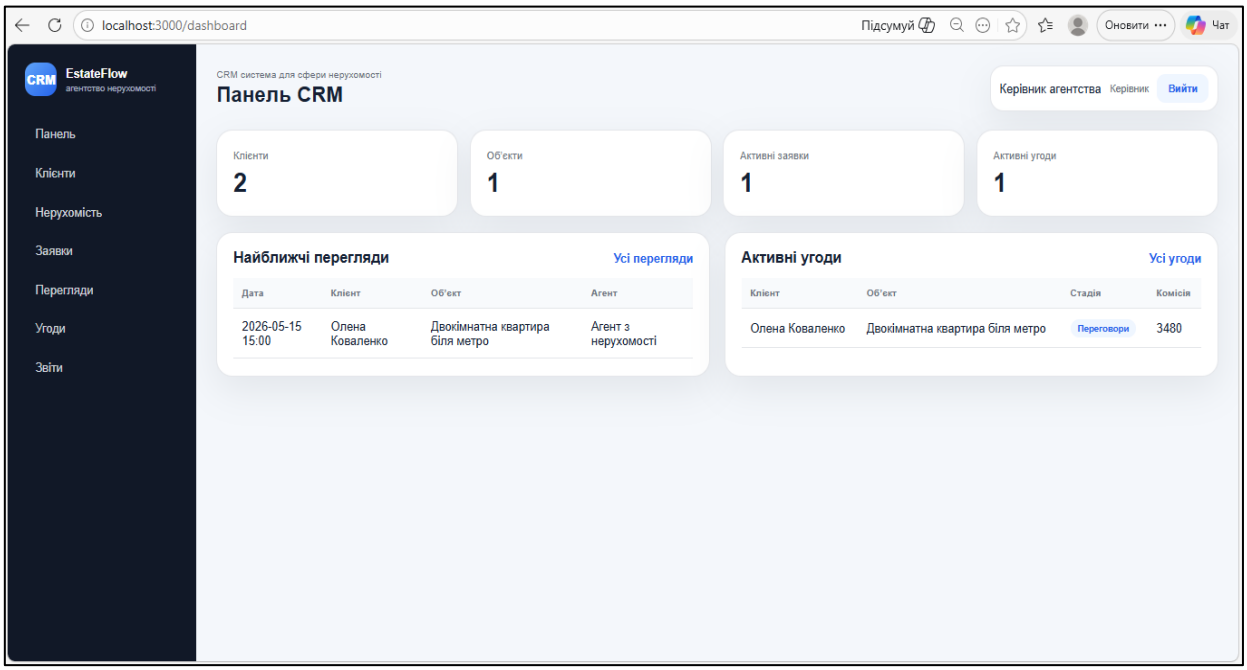


Рисунок 3.2 – Головна панель CRM-системи після входу користувача

Модуль клієнтів реалізовано у вигляді сторінки з таблицею, пошуковим полем і кнопкою додавання нового запису. Для кожного запису передбачено дії перегляду або редагування. Форма створення клієнта містить поля для ПІБ, телефону, електронної пошти, типу клієнта, джерела звернення та коментаря. Сторінка керування клієнтами CRM-системи представлена на рисунку 3.3.

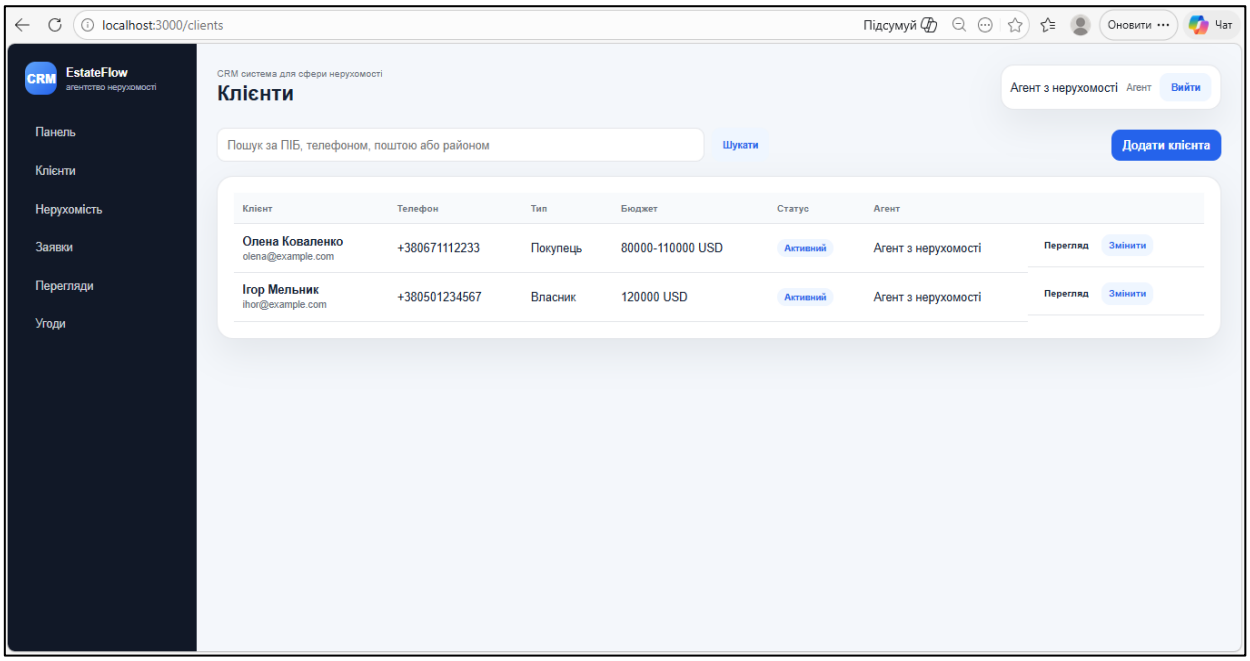


Рисунок 3.3 – Сторінка керування клієнтами CRM-системи

Модуль об’єктів нерухомості призначений для ведення каталогу пропозицій агентства. Інтерфейс сторінки містить таблицю об’єктів з зазначенням типу нерухомості, району, адреси, площі, кількості кімнат, ціни та статусу доступності. Форма додавання об’єкту побудована за принципом структурованого введення параметрів, що спрощує подальший пошук і фільтрацію. Сторінка каталогу об’єктів нерухомості наведена на рисунку 3.4.

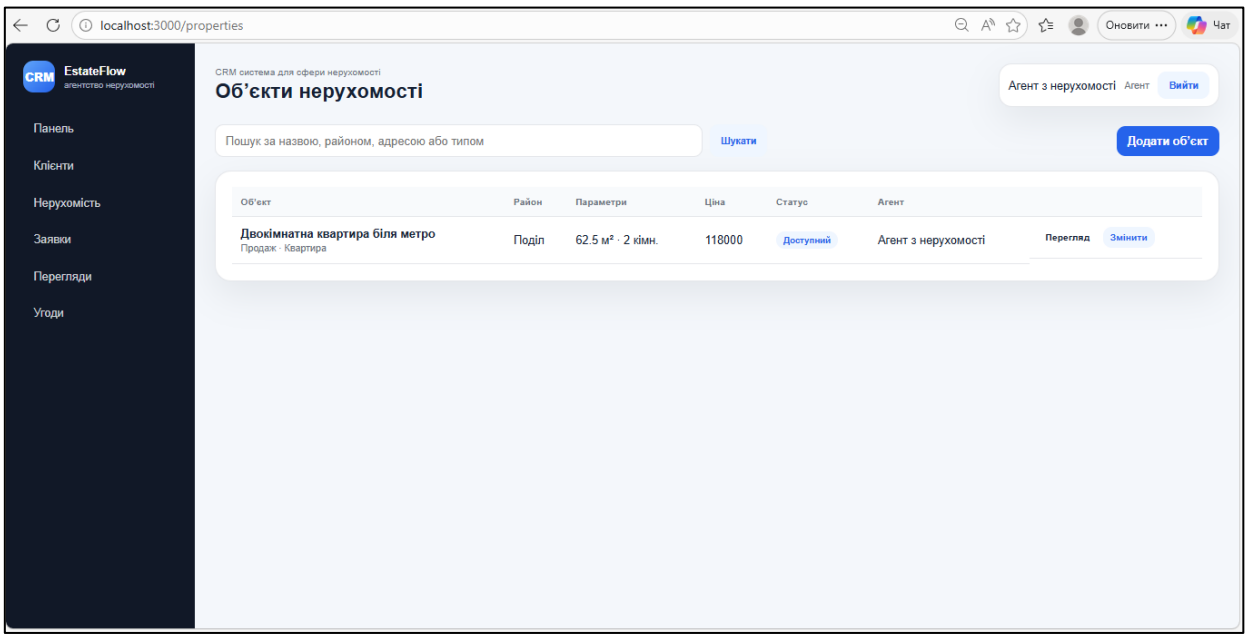


Рисунок 3.4 – Сторінка каталогу об’єктів нерухомості

Далі, система надає можливість створити картку об’єкту нерухомості. Відповідна форма представлена на рисунку 3.5.

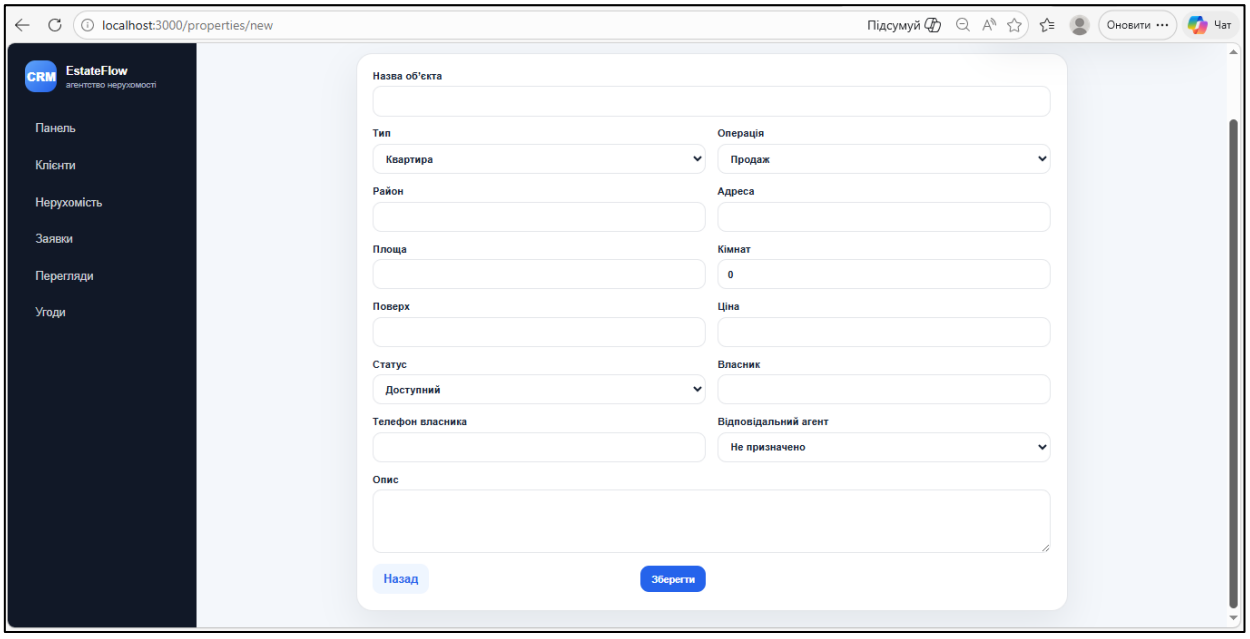


Рисунок 3.5 – Форма створення картки об’єкта нерухомості

В таблиці заявок виводяться клієнт, тип операції, бюджет, район, кількість кімнат, пріоритет, статус і відповідальний агент. Сторінка заявок клієнтів у CRM-системі представлена на рисунку 3.6.

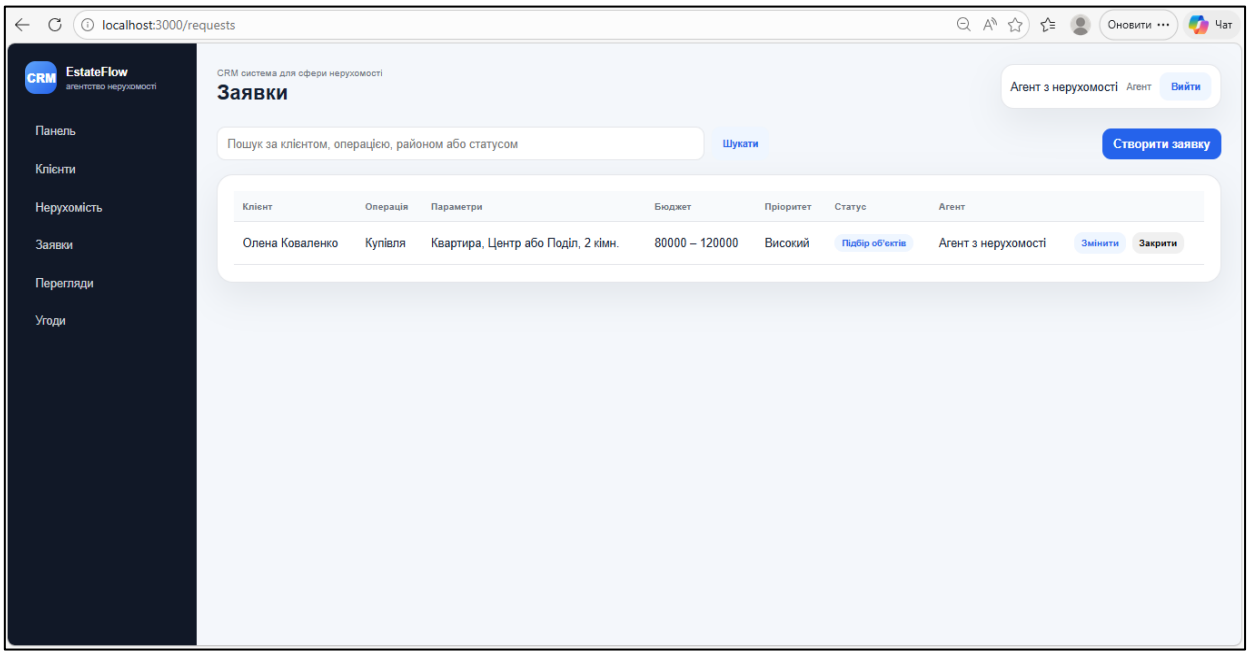


Рисунок 3.6 – Сторінка заявок клієнтів у CRM-системі

Перегляди оформлено окремим інтерфейсним модулем, оскільки демонстрація об’єкта клієнту є важливим етапом рієлторського процесу. Після проведення зустрічі агент вносить результат, що дозволяє зберегти реакцію клієнта та визначити подальші дії. Сторінка планування переглядів об’єктів нерухомості представлена на рисунку 3.7.

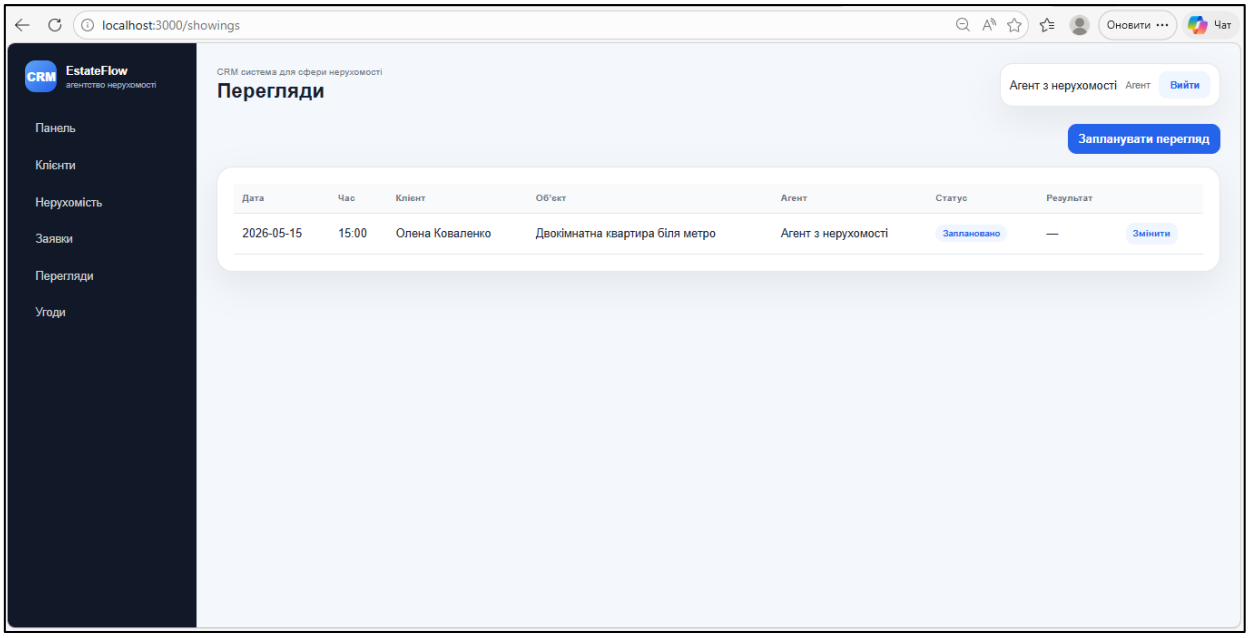


Рисунок 3.7 – Сторінка планування переглядів об’єктів нерухомості

Модуль угод відображає фінальний етап роботи з клієнтом. Інтерфейс сторінки містить перелік угод зі стадіями, сумами, комісіями, плановими датами завершення й відповідальними агентами. Для керівника, сторінка угод має управлінське значення, оскільки дозволяє оцінити потенційний дохід агентства та контролювати перебіг домовленостей. Для агента, модуль угод є інструментом фіксації реального результату роботи з заявкою та об’єктом. Сторінка супроводу угод у CRM-системі наведена на рисунку 3.8.

Звітний інтерфейс доступний користувачам з управлінськими правами. Він містить підсумкові показники за клієнтами, об’єктами, заявками, переглядами, угодами та роботою агентів. Для візуального сприйняття використовуються картки та таблиці, які дають змогу швидко оцінити стан діяльності агентства без підрахунку. Звітний модуль пов’язаний з

серверними SQL-запитами, тому дані в інтерфейсі відображають фактичний стан бази. Звітна панель CRM-системи для керівника агентства представлена на рисунку 3.9.

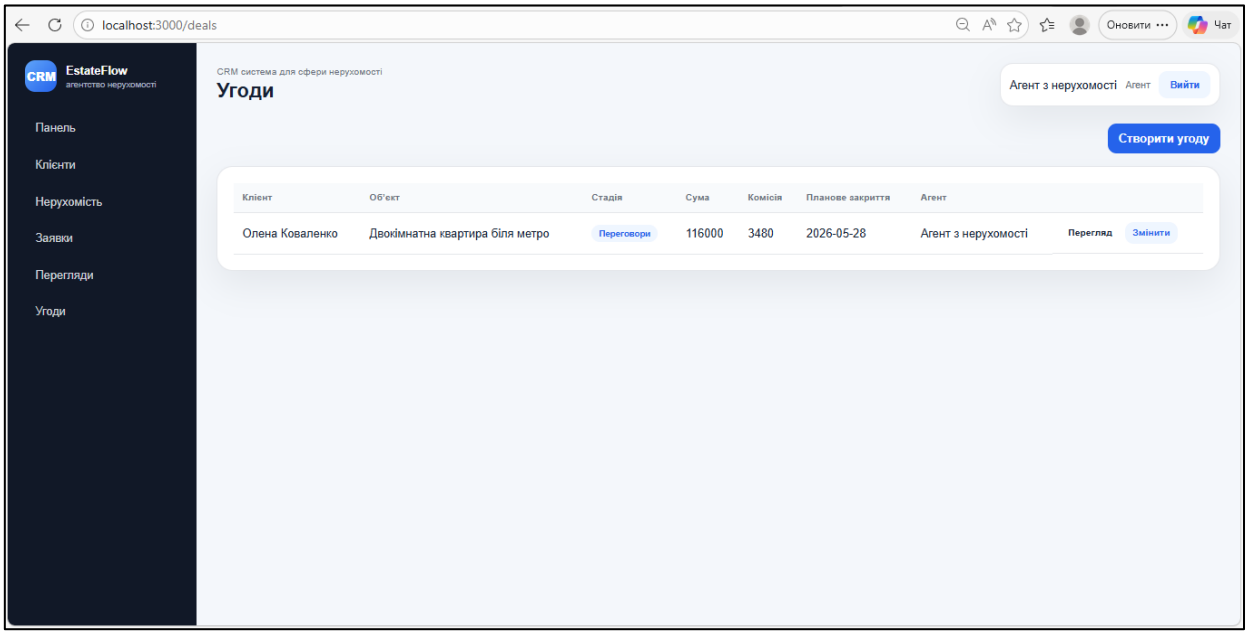


Рисунок 3.8 – Сторінка супроводу угод в CRM-системі

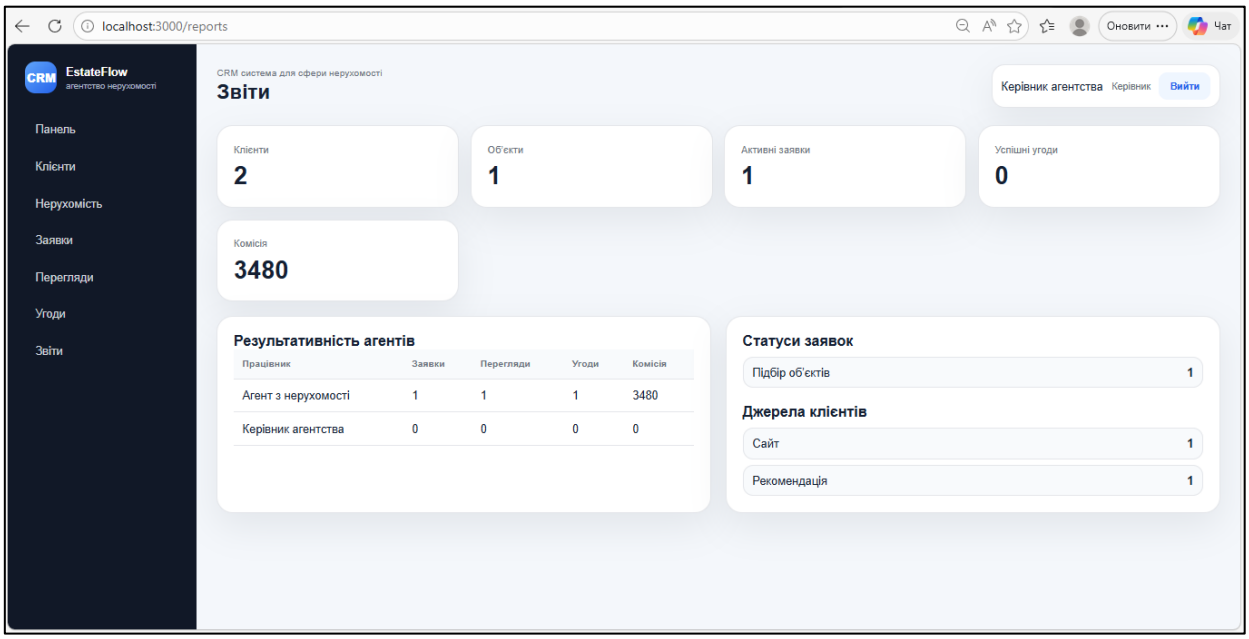


Рисунок 3.9 – Звітна панель CRM-системи для керівника агентства

Адміністративний інтерфейс призначений для керування користувачами та перегляду журналу дій. На сторінці користувачів

адміністратор може бачити облікові записи, ролі, електронні адреси та стан активності. Окремий блок журналу дій відображає службові події, пов’язані з входами, виходами, створенням або зміною записів. Наявність адміністративного інтерфейсу забезпечує контроль за роботою системи й підтримує прозорість дій працівників. Адміністративна панель керування користувачами наведена на рисунку 3.10.

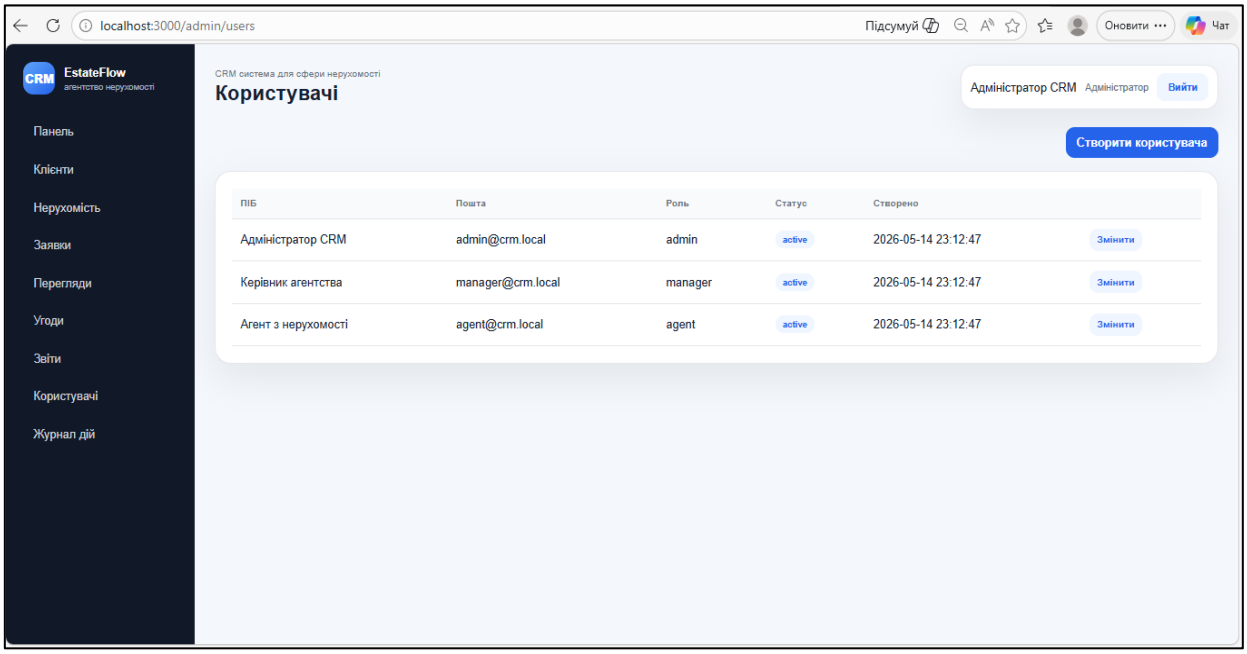


Рисунок 3.10 – Адміністративна панель керування користувачами

Візуальне оформлення CRM-системи зосереджене на читабельності та робочій зручності. Таблиці мають чіткі заголовки, кнопки дій візуально відокремлені від основного тексту, а форми містять логічно згруповані поля. Для головних операцій використовуються помітні кнопки, а службові або другорядні дії розміщуються ближче до відповідних записів. Простір між елементами підібрано так, щоб сторінки не виглядали перевантаженими, але водночас містили достатній обсяг робочої інформації.

Клієнтський інтерфейс створено як практичну оболонку для серверної логіки CRM-системи. Він забезпечує доступ до функцій, необхідних агенту, керівнику й адміністратору, а також, дає змогу виконувати основні операції без безпосередньої роботи з базою даних. Реалізовані сторінки входу,

головної панелі, клієнтів, об’єктів, заявок, переглядів, угод, звітів і адміністрування підтверджують завершеність користувацької частини програмного продукту.

3.4 Перевірка працездатності основних функцій застосунку

Перевірка працездатності CRM-системи виконувалася після завершення реалізації серверної частини, клієнтського інтерфейсу та структури бази даних. Метою тестування було підтвердження коректної роботи основних сценаріїв, які відображають діяльність агентства нерухомості, вхід користувача до системи, роботу з клієнтами, додавання об’єктів, створення заявок, планування переглядів, супровід угод, формування звітів і адміністрування облікових записів. Перевірка проводилася на локальному сервері після запуску застосунку через команду npm start.

Працездатність програмного продукту оцінювалася за практичними діями користувача в інтерфейсі. Для кожного сценарію перевірялася наявність очікуваного результату в системі, коректність запису до бази даних і відсутність аварійного завершення роботи сервера. Особливу увагу приділено операціям, пов’язаним зі створенням нових записів, оскільки саме вони підтверджують взаємодію між формами інтерфейсу, серверними маршрутами та SQLite-базою.

Першим етапом перевірки стала авторизація. Користувач відкривав сторінку входу, вводив електронну пошту та пароль, після чого система перевіряла облікові дані через серверний маршрут. При правильному введені відкривалася головна панель, а в сесії зберігалися роль, ім’я та ідентифікатор користувача. При введенні неправильного паролю доступ до внутрішніх сторінок не надавався, що підтвердило роботу механізму захисту.

У формі створення клієнта вводилися тестові контактні дані, тип клієнта, джерело звернення та коментар. Після натискання кнопки

збереження серверний маршрут обробляв запит, виконував SQL-команду додавання запису та повертав користувача до списку клієнтів. Новий запис відображався в таблиці, що підтвердило коректність взаємодії між інтерфейсом, маршрутом clients.js і таблицею clients. Заповнення форми додавання нового клієнта представлено на рисунку 3.11. Відображення доданого клієнта в загальній таблиці наведено на рисунку 3.12.

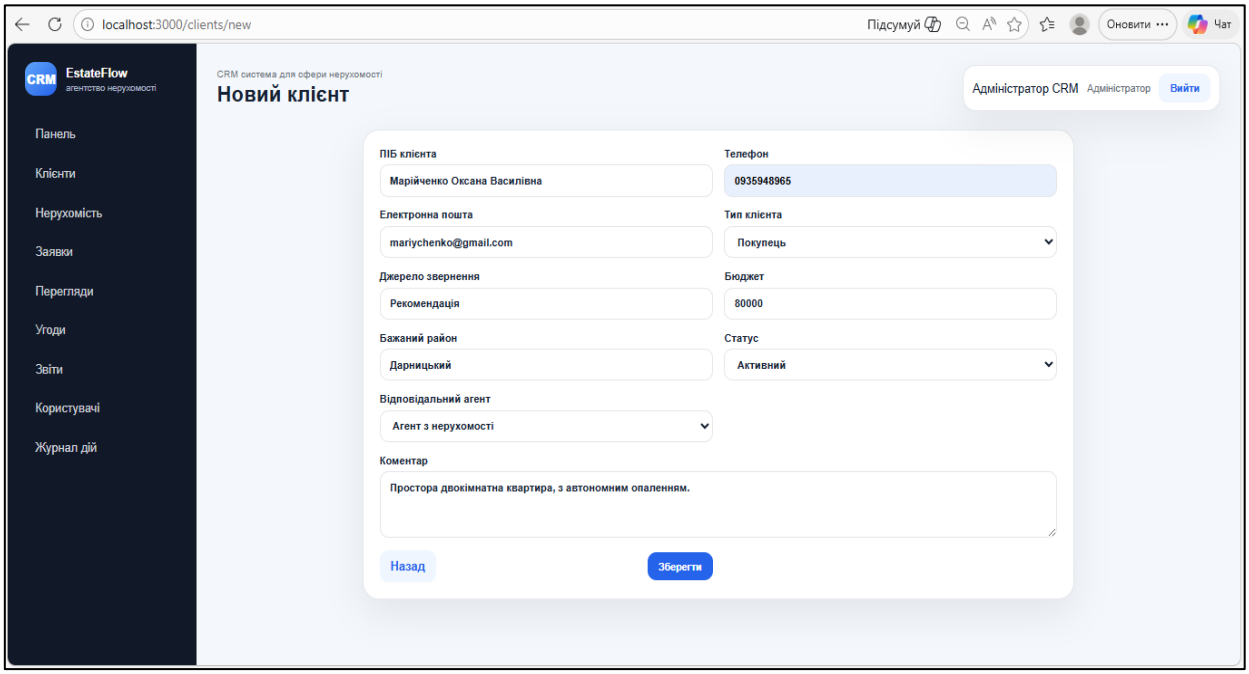


Рисунок 3.11 – Заповнення форми додавання нового клієнта

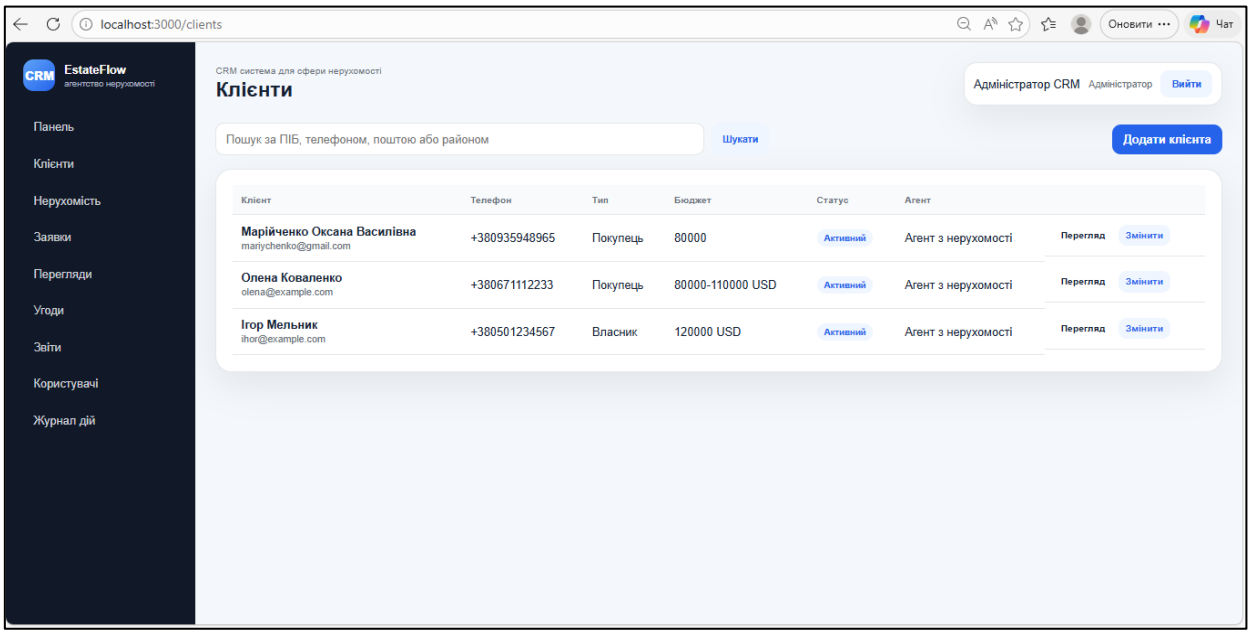


Рисунок 3.12 – Відображення доданого клієнта в загальній таблиці

Перевірка модуля об’єктів нерухомості виконувалася шляхом створення нової картки об’єкта. У форму вводилися тип нерухомості, район, адреса, площа, кількість кімнат, ціна, статус і текстовий опис. Після збереження, система додавала об’єкт до каталогу, а сторінка об’єктів відображала новий запис разом з заданими параметрами. Заповнення форми створення об’єкта нерухомості представлено на рисунку 3.13. Новий об’єкт в каталозі нерухомості зображено на рисунку 3.14.

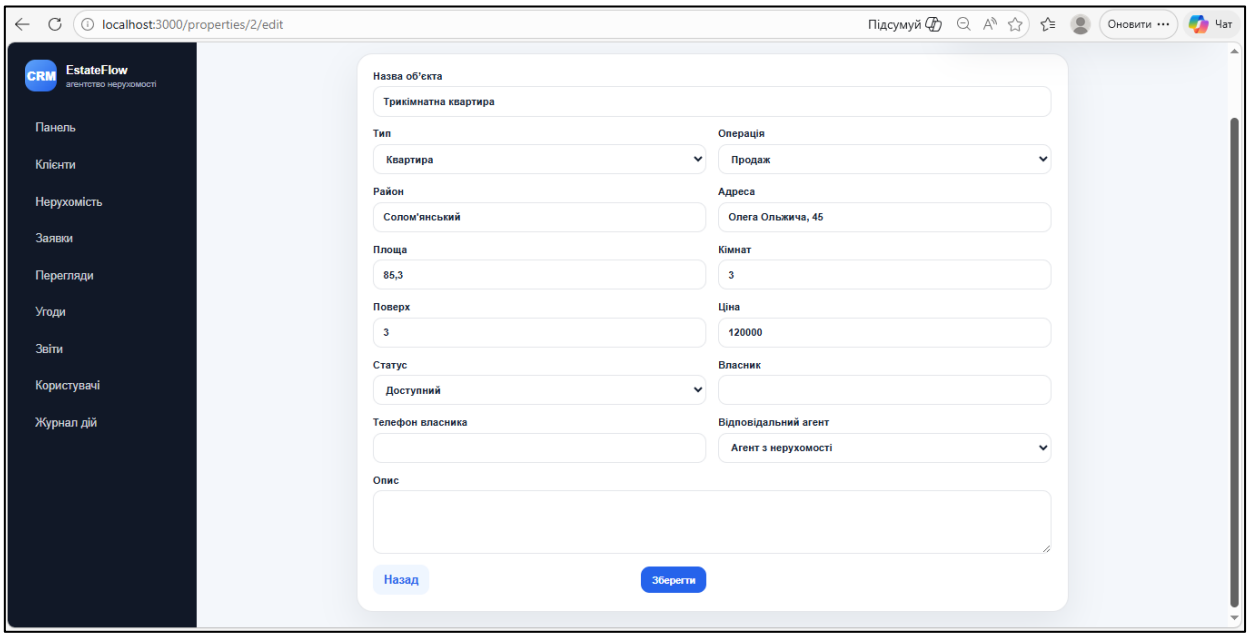


Рисунок 3.13 – Заповнення форми створення об’єкта нерухомості

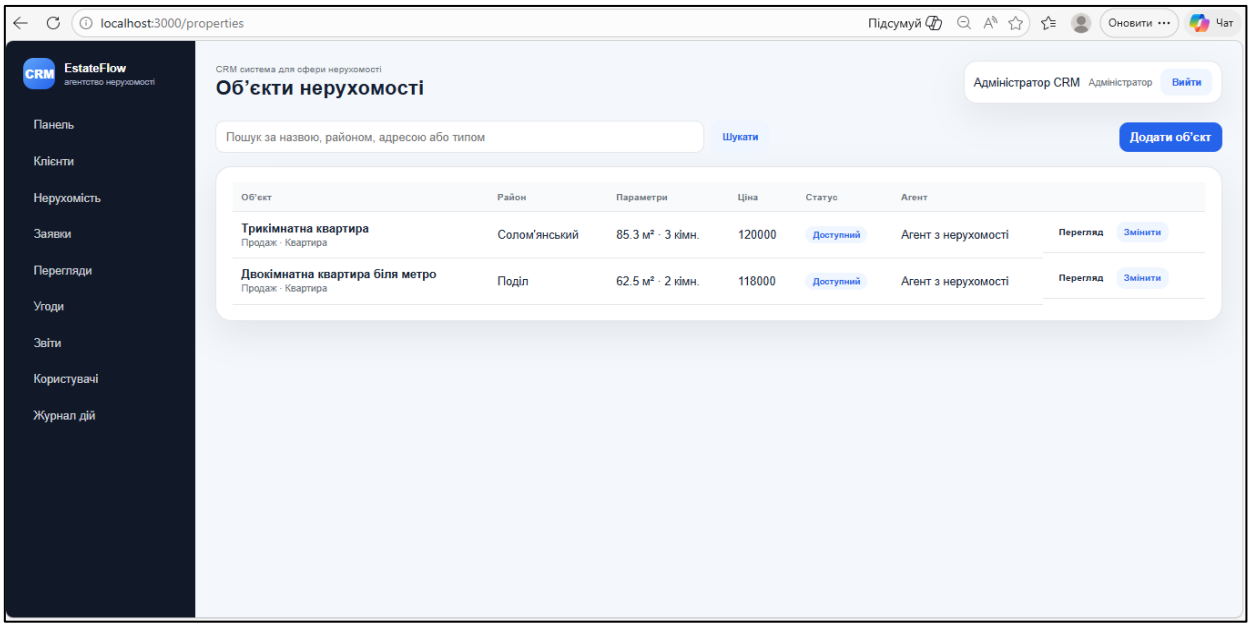


Рисунок 3.14 – Новий об’єкт в каталозі нерухомості

Після перевірки клієнтів і об’єктів було протестовано створення заявки. Заявка формувалася на основі вже внесеного клієнта, що відповідає логіці предметної області. У формі зазначалися тип операції, бажаний район, бюджет, кількість кімнат, пріоритет і статус. В загальному списку заявок з’являвся новий запис, що підтвердило правильну роботу модуля звернень. Створення заявки клієнта на підбір нерухомості представлено на рисунку 3.15. Додана заявка в робочому списку CRM-системи представлена на рисунку 3.16.

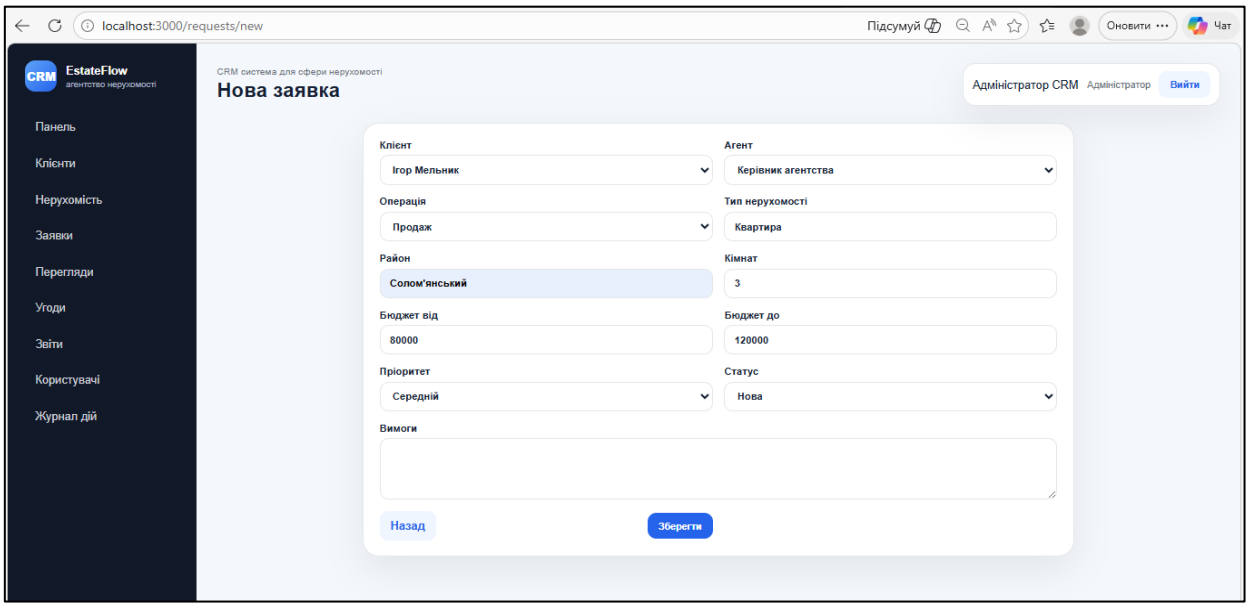


Рисунок 3.15 – Створення заявки клієнта на підбір нерухомості

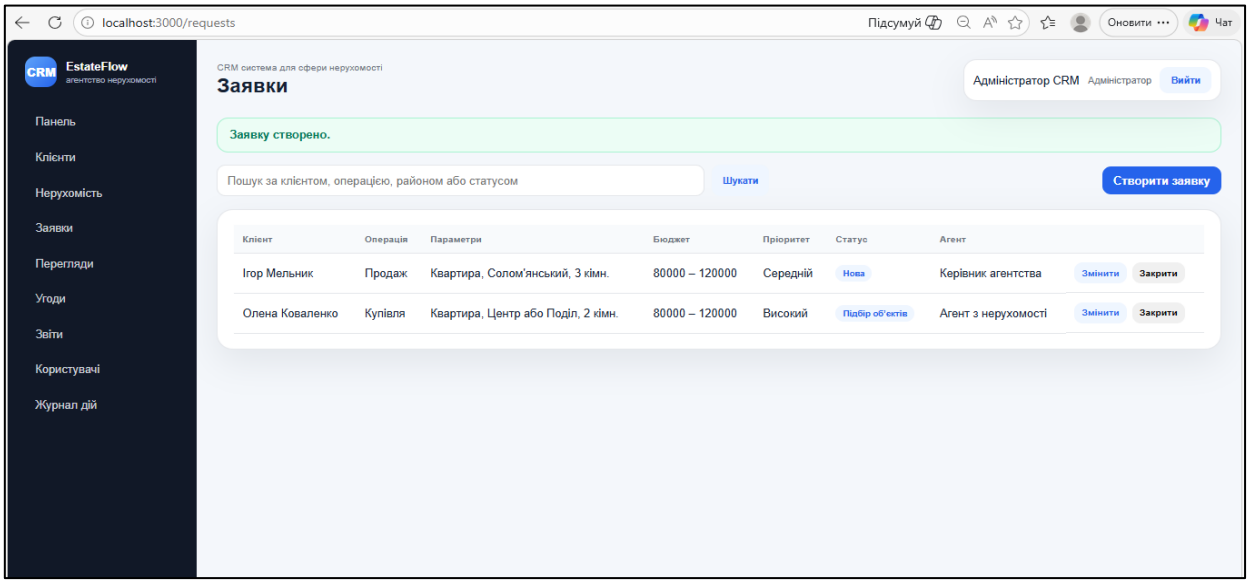


Рисунок 3.16 – Додана заявка в робочому списку CRM-системи

Модуль переглядів перевірявся після створення клієнта та об’єкта нерухомості. У формі планування вказувалися клієнт, об’єкт, відповідальний агент, дата, час і початковий результат. Після збереження, перегляд відображався в загальному списку зустрічей. Планування перегляду об’єкта нерухомості представлено на рисунку 3.17. Запланований перегляд в таблиці CRM-системи представлено на рисунку 3.18.

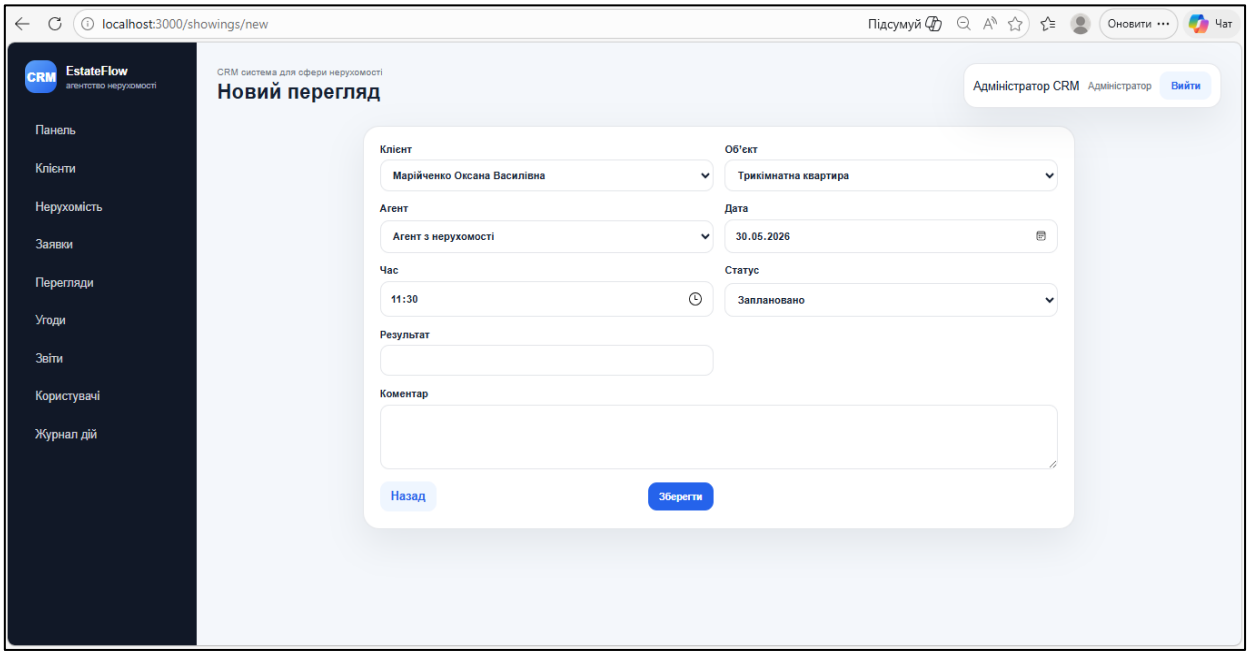


Рисунок 3.17 – Планування перегляду об’єкта нерухомості

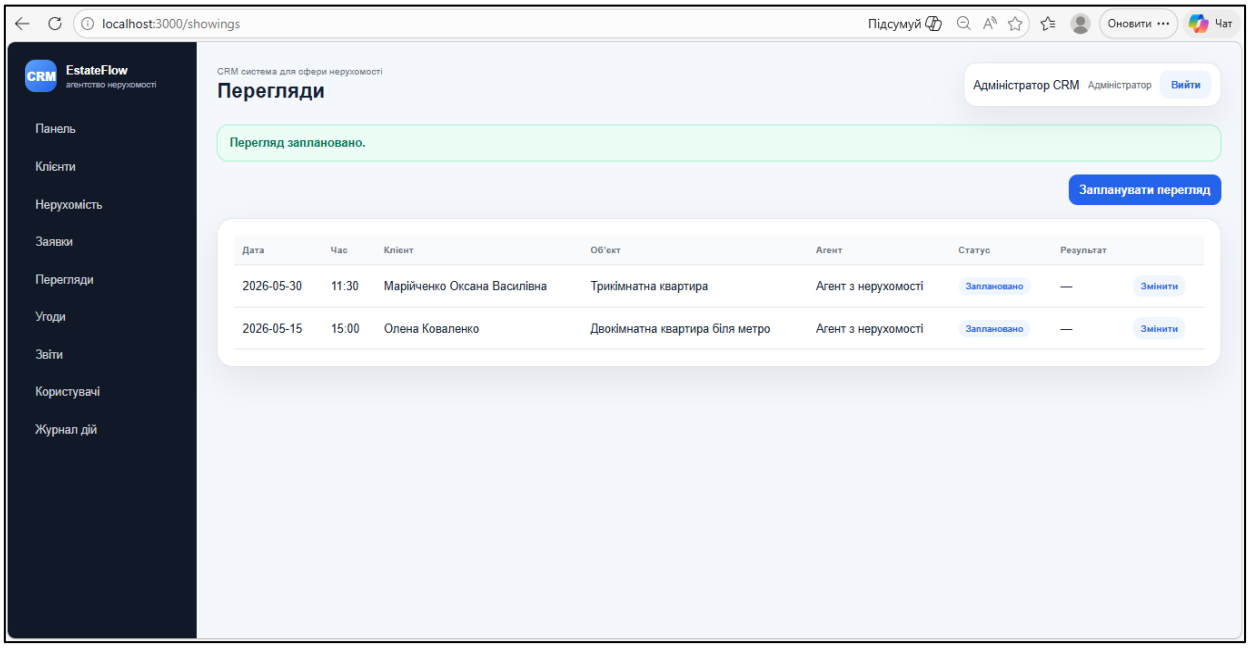


Рисунок 3.18 – Запланований перегляд у таблиці CRM-системи

Перевірка модуля угод виконувалася через створення запису з прив’язкою до клієнта, об’єкта та відповідального агента. У формі зазначалися стадія угоди, вартість, комісія та планова дата завершення. Після збереження запис відображався в таблиці угод. Заповнення форми створення угоди представлено на рисунку 3.19. Відображення створеної угоди в CRM-системі представлено на рисунку 3.20.

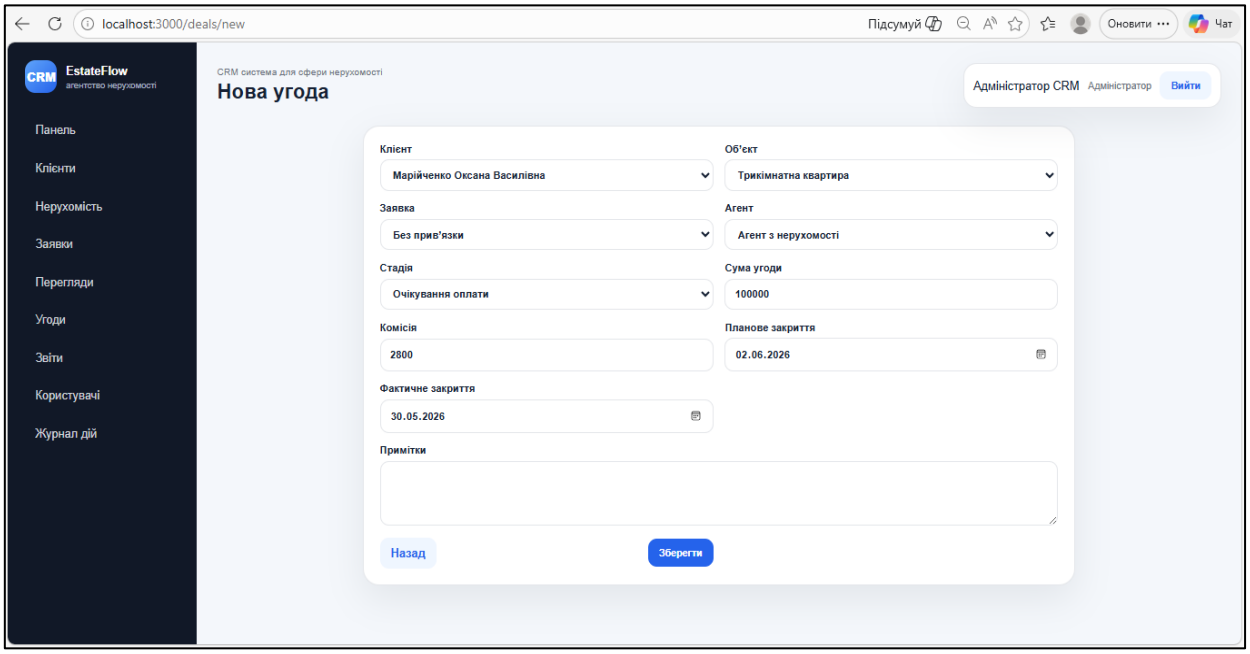


Рисунок 3.19 – Заповнення форми створення угоди

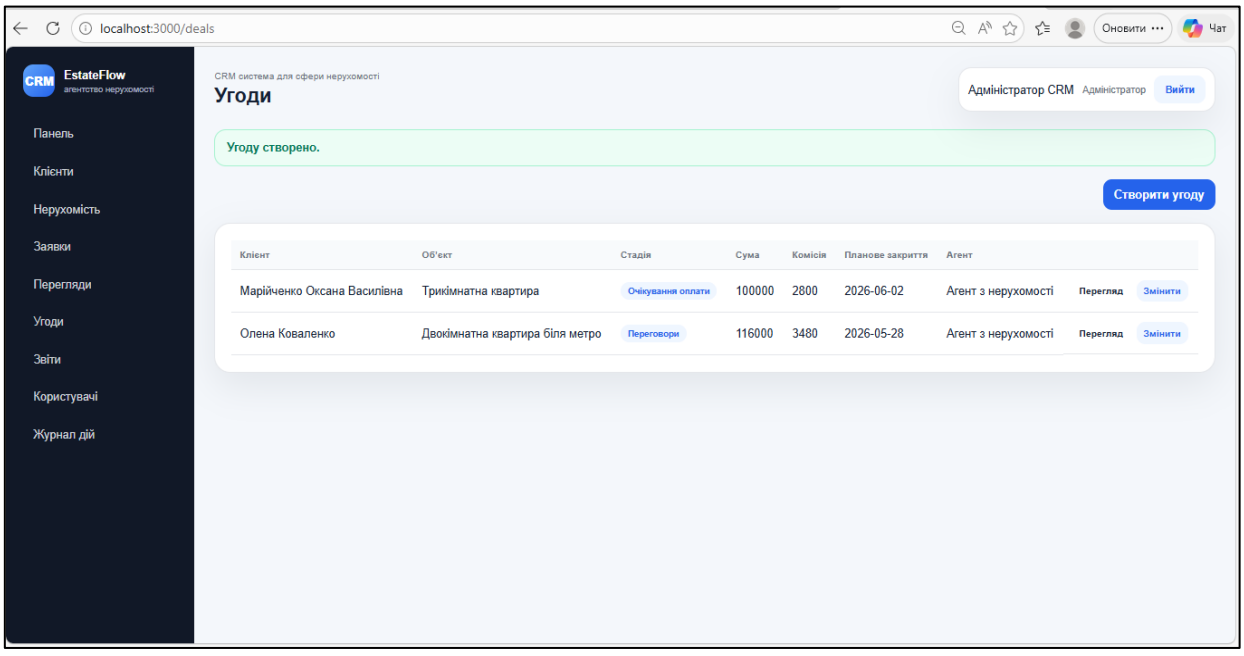


Рисунок 3.20 – Відображення створеної угоди в CRM-системі

Окремо перевірялася звітність. В систему було внесено кілька тестових записів клієнтів, об’єктів, заявок, переглядів і угод, після чого виконано перехід до звітного розділу під роллю керівника. Система відобразила підсумкові числові показники та таблиці з результатами роботи агентів. Перевірка звітного модуля CRM-системи представлена на рисунку 3.21.

Адміністративний модуль перевірявся під обліковим записом адміністратора. Після створення або оновлення записів у функціональних модулях журнал відображав нові службові події. Журнал дій користувачів в адміністративному модулі представлено на рисунку 3.22.

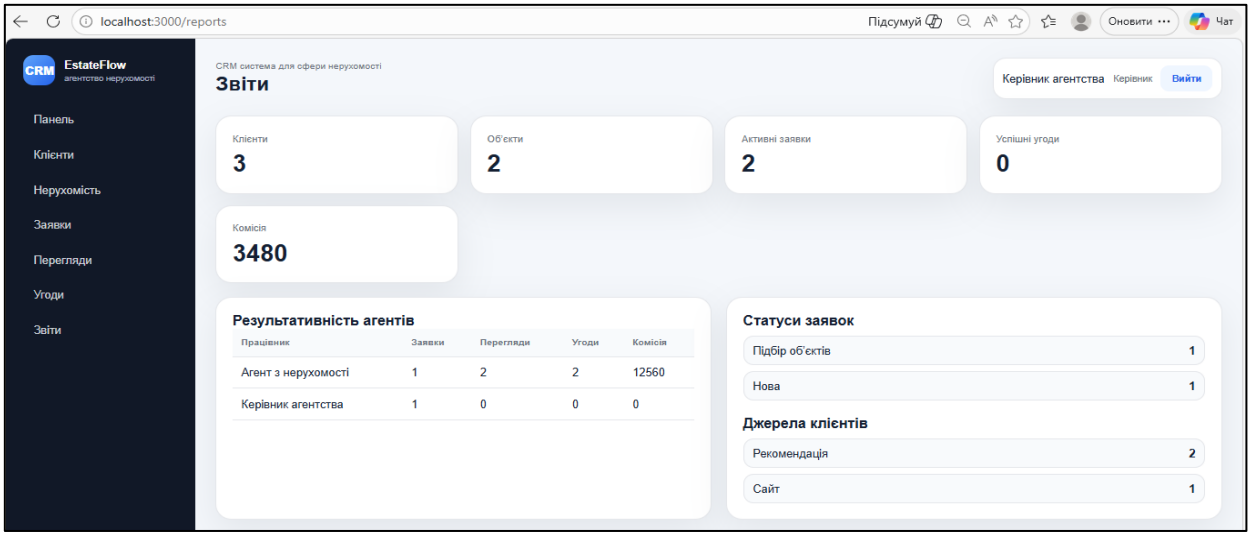


Рисунок 3.21 – Перевірка звітного модуля CRM-системи

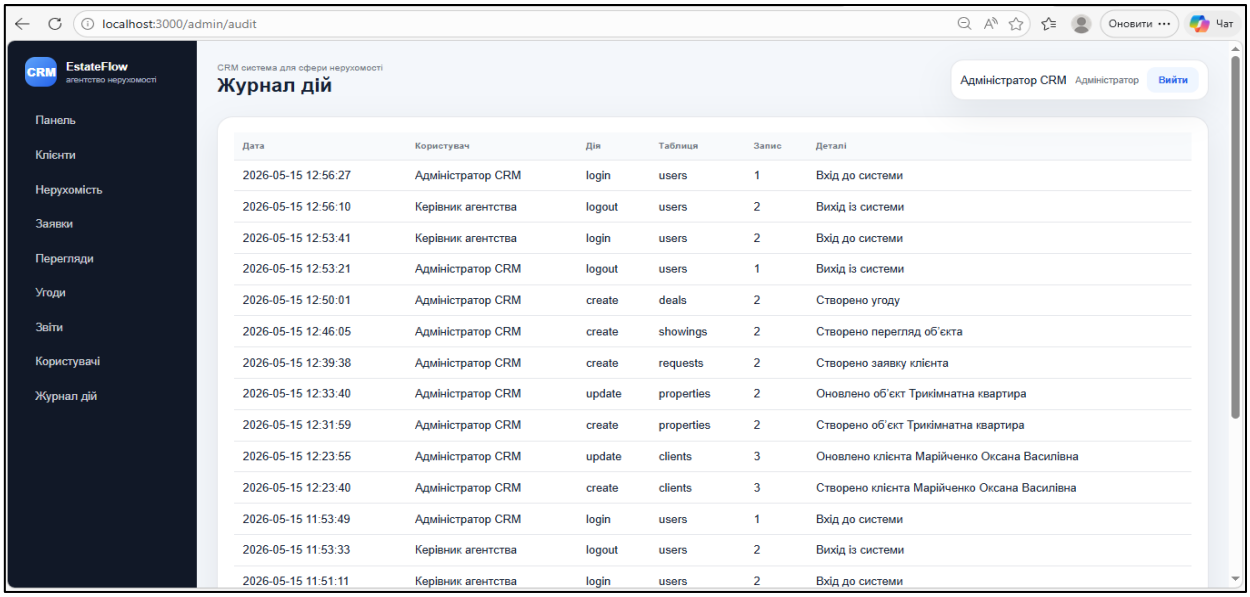


Рисунок 3.22 – Журнал дій користувачів в адміністративному модулі

Проведене тестування підтвердило, що CRM-система виконує основні функції, передбачені під час постановки задачі та проєктування. Авторизація працює з урахуванням ролей користувачів, операційні модулі зберігають записи до бази даних, звітність формує узагальнені показники, а адміністративний модуль фіксує службову активність.

3.5 Підготовка керівництва користувача для роботи з системою

Керівництво користувача завершує опис реалізації CRM-системи, оскільки воно пояснює порядок запуску програмного продукту та правила виконання основних робочих операцій. Розроблена система призначена для працівників агентства нерухомості, які виконують адміністративні, управлінські й операційні дії. Практична робота з застосунком починається з запуску локального сервера, переходу до вебінтерфейсу та авторизації під відповідною роллю.

Перед початком роботи користувач повинен розпакувати проєкт, відкрити командний рядок в папці застосунку, встановити залежності командою npm install і запустити сервер командою npm start. Після успішного запуску CRM-система стає доступною в браузері за адресою http://localhost:3000. Якщо сервер працює коректно, браузер відкриває сторінку авторизації, де користувач вводить електронну пошту та пароль. У вікні входу службові облікові дані не відображаються, що зменшує ризик їх випадкового розголошення під час демонстрації.

Після авторизації користувач потрапляє на головну панель. Її призначення полягає в швидкому переході до робочих розділів і перегляді загального стану даних. Агент використовує панель як стартову сторінку для роботи з клієнтами, нерухомістю, заявками, переглядами та угодами. Керівник отримує зручний доступ до показників діяльності агентства. Адміністратор переходить до керування користувачами та журналом дій.

Зміст меню залежить від ролі, тому працівник бачить лише доступні для нього розділи.

Робота з клієнтами починається з переходу до відповідного розділу через головне меню. На сторінці відображається таблиця з контактами, типом клієнта, джерелом звернення та службовими діями. Для створення нового запису користувач відкриває форму додавання, вводить ПІБ, телефон, електронну пошту, тип клієнта, джерело звернення та коментар. Після збереження система повертає користувача до таблиці, де новий клієнт стає доступним для подальшої роботи із заявками, переглядами й угодами.

Розділ об’єктів нерухомості використовується для ведення каталогу пропозицій. Користувач відкриває сторінку об’єктів, переглядає наявні записи або створює нову картку. Під час додавання нерухомості вводяться тип об’єкта, район, адреса, площа, кількість кімнат, ціна, статус і опис. Після збереження запис потрапляє до загального каталогу, де може використовуватися під час планування перегляду або створення угоди.

Заявки відображають потреби клієнтів і дають змогу контролювати процес опрацювання звернення. Для створення заявки користувач обирає клієнта, зазначає тип операції, бажаний район, бюджет, кількість кімнат, пріоритет і статус. Після збереження запис з’являється в таблиці заявок, де агент може змінювати стан опрацювання. Керівник використовує сторінку заявок для контролю навантаження працівників і своєчасності реагування на звернення.

Планування переглядів виконується після того, як клієнт зацікавився конкретним об’єктом. Користувач переходить до розділу переглядів, обирає клієнта, нерухомість, відповідального агента, дату та час зустрічі. Після проведення перегляду в систему вноситься результат і короткий коментар.

Розділ угод використовується на фінальному етапі роботи з клієнтом. Користувач створює угоду, обираючи клієнта, об’єкт нерухомості, відповідального агента, стадію, вартість, комісію та планову дату завершення. Після збереження угода відображається в таблиці, де можна контролювати її стан.

Звітний розділ призначений для керівника та адміністратора. В ньому відображаються підсумкові показники за клієнтами, об’єктами, заявками, переглядами, угодами й роботою агентів. Користувач не вводить дані вручну, оскільки звіти формуються серверною частиною на основі записів в базі SQLite.

Адміністративний розділ призначений для підтримання коректної роботи системи. Адміністратор може переглядати користувачів, їх ролі, статуси облікових записів і журнал дій. За потреби він створює нового працівника або змінює стан наявного облікового запису.

Журнал дій відображає важливі події, зокрема входи до системи, створення клієнтів, додавання об’єктів, зміну заявок та інші службові операції.

Після завершення роботи користувач повинен натиснути кнопку виходу. Сервер завершує сесію, а браузер повертає користувача на сторінку авторизації. Завершення сесії є важливою дією, оскільки CRM-система містить контактні, фінансові й службові дані агентства. Особливо важливо виконувати вихід після роботи на спільному або демонстраційному комп’ютері.

3.6 Висновки до розділу 3

В третьому розділі виконано практичну реалізацію CRM-системи для сфери нерухомості, обґрунтовано технологічний стек, описано серверну частину, клієнтський інтерфейс, перевірку працездатності та порядок роботи користувача із застосунком. Обрана технологічна основа у вигляді Node.js, Express.js, SQLite, HTML, CSS, JavaScript і EJS дала змогу створити вебзастосунок з локальним запуском, зручною структурою файлів, базою даних, авторизацією, ролями користувачів і робочими модулями для агентства нерухомості.

Серверна частина реалізує основну логіку програмного продукту. Вона відповідає за запуск вебсервера, оброблення маршрутів, взаємодію з SQLite-базою, перевірку авторизації, розмежування доступу й журналювання дій.

Клієнтський інтерфейс побудовано як набір сторінок, орієнтованих на щоденну роботу адміністратора, керівника й агента з нерухомості. Реалізовано сторінку входу, головну панель, таблиці клієнтів, каталог об’єктів, сторінки заявок, переглядів, угод, звітів і адміністративний розділ. Форма авторизації не містить демонстраційних логінів і паролів, що відповідає вимогам безпечнішого представлення системи під час перевірки або захисту. Меню та доступні функції залежать від ролі користувача, тому працівник отримує лише потрібні для нього інструменти.

Перевірка працездатності підтвердила коректне виконання основних сценаріїв. Успішно протестовано вхід до системи, відхилення неправильного пароля, додавання клієнта, створення об’єкта нерухомості, реєстрацію заявки, планування перегляду, створення угоди, перегляд звітності та роботу адміністративного журналу. Отримані результати свідчать, що інтерфейс, серверна частина і база даних працюють узгоджено, а програмний продукт виконує функції, визначені під час постановки задачі.

ВИСНОВКИ

В кваліфікаційній роботі виконано розробку CRM-системи для сфери нерухомості, орієнтованої на автоматизацію основних процесів агентства. Під час дослідження встановлено, що рієлторська діяльність потребує впорядкованої роботи з клієнтами, об’єктами, заявками, переглядами, угодами й управлінською звітністю. Розрізнене зберігання відомостей в таблицях, месенджерах і особистих нотатках не забезпечує належного рівня контролю, тому для агентства важливим є перехід до єдиного інформаційного середовища.

В процесі аналізу предметної області охарактеризовано взаємодію агентства нерухомості з клієнтами та визначено основні інформаційні потоки. Клієнтська заявка не є ізольованим записом, оскільки вона пов’язується з потребами клієнта, параметрами об’єктів, переглядами, результатами комунікації та фінальними угодами.

Огляд програмних аналогів дав змогу порівняти універсальні CRM-платформи та спеціалізовані рішення для нерухомості. Універсальні системи мають широкі можливості для продажів та аналітики, але потребують адаптації під рієлторські процеси. Галузеві рішення краще враховують специфіку нерухомості, проте часто орієнтовані на великі команди, іноземний ринок або складні інвесторські сценарії.

Під час формування вимог визначено основні ролі користувачів, функціональні блоки та межі системи. В CRM-системі передбачено ролі адміністратора, керівника й агента з нерухомості. Адміністратор відповідає за користувачів і службові параметри, керівник отримує інструменти контролю та звітності, агент виконує операційну роботу з клієнтами, об’єктами, заявками, переглядами та угодами. Рольова модель дала змогу забезпечити логічне розмежування доступу та підвищити керованість системи.

В проєктній частині розроблено архітектуру програмного забезпечення, структуру даних і UML-діаграми. Архітектуру побудовано за клієнт-серверною моделлю з поділом на презентаційний рівень, бізнес-логіку, доступ до даних і рівень збереження інформації. Структура бази даних містить таблиці користувачів, клієнтів, об’єктів, заявок, переглядів, угод і журналу дій. UML-діаграми прецедентів, послідовності, активності та класів формалізували логіку роботи та створили основу для програмної реалізації.

Практичну реалізацію виконано у вигляді вебзастосунку на основі Node.js, Express.js і SQLite. Серверна частина забезпечує маршрутизацію, авторизацію, перевірку ролей, роботу з базою даних і журналювання. Клієнтський інтерфейс створено засобами HTML, CSS, JavaScript і EJS. В програмному продукті реалізовано сторінку входу, головну панель, модулі клієнтів, об’єктів, заявок, переглядів, угод, звітності та адміністрування.

Перевірка працездатності підтвердила виконання основних сценаріїв. Було протестовано авторизацію, відмову при некоректному паролі, додавання клієнта, створення об’єкта нерухомості, реєстрацію заявки, планування перегляду, створення угоди, перегляд звітів і адміністративний журнал. Всі перевірені функції виконалися коректно, а записи зберігалися в SQLite-базі. Підготовлене керівництво користувача пояснює порядок запуску системи та роботу з основними модулями.

Результатом роботи став готовий програмний продукт, який демонструє автоматизацію ключових процесів агентства нерухомості. CRM-система забезпечує централізоване збереження даних, рольове розмежування доступу, зручну роботу з клієнтами та об’єктами, контроль заявок, переглядів і угод, а також, формування управлінської звітності. Подальший розвиток може передбачати інтеграцію з рекламними майданчиками, додавання календарних сповіщень, завантаження фотографій об’єктів, експорт звітів і розширення аналітичного блоку.

ПЕРЕЛІК ПОСИЛАНЬ

1. Про інформацію : Закон України від 02.10.1992 № 2657-XII. База даних «Законодавство України» / Верховна Рада України. URL: https://zakon.rada.gov.ua/go/2657-12 (дата звернення: 15.02.2026).

2. Про захист персональних даних : Закон України від 01.06.2010 № 2297-VI. База даних «Законодавство України» / Верховна Рада України. URL: https://zakon.rada.gov.ua/go/2297-17 (дата звернення: 15.02.2026).

3. Цивільний кодекс України : Кодекс України від 16.01.2003 № 435-IV. База даних «Законодавство України» / Верховна Рада України. URL: https://zakon.rada.gov.ua/go/435-15 (дата звернення: 15.02.2026).

4. Buttle F., Maklan S. Customer Relationship Management: Concepts and Technologies : 4th ed. New York : Routledge, 2019. 444 p. URL: https://doi.org/10.4324/9781351016551 (дата звернення: 15.02.2026).

5. Payne A., Frow P. A Strategic Framework for Customer Relationship Management. *Journal of Marketing*. 2005. Vol. 69, No. 4. P. 167–176. URL: https://journals.sagepub.com/doi/10.1509/jmkg.2005.69.4.167 (дата звернення: 17.02.2026).

6. Sales Force Automation Software by Sales Cloud. *Salesforce*. URL: https://www.salesforce.com/sales/cloud/ (дата звернення: 17.02.2026).

7. Sales Hub. *HubSpot*. URL: https://www.hubspot.com/products/sales (дата звернення: 17.02.2026).

8. Commercial Real Estate CRM Software. *Pipedrive*. URL: https://www.pipedrive.com/en/industries/real-estate-crm (дата звернення: 19.02.2026).

9. Zoho CRM’s Full Feature Suite. *Zoho*. URL: https://www.zoho.com/crm/features.html (дата звернення: 19.02.2026).

10. The Real Estate Team OS. *Follow Up Boss*. URL: https://www.followupboss.com/ (дата звернення: 20.02.2026).

11. Salesforce-powered Real Estate CRM. *Propertybase by Lone Wolf*. URL: https://get.lwolf.com/propertybase/ (дата звернення: 20.02.2026).

12. Real Estate CRM and Website Solution. *Real Geeks*. URL: https://www.realgeeks.com/ (дата звернення: 24.02.2026).

13. Real Estate CRM Feature. *REsimpli*. URL: https://resimpli.com/features/crm/ (дата звернення: 24.02.2026).

14. ДСТУ ISO/IEC/IEEE 29148:2025. Інженерія систем і програмних засобів. Процеси життєвого циклу. Інженерія вимог (ISO/IEC/IEEE 29148:2018, IDT). URL: https://online.budstandart.com/ua/catalog/docpage.html?id_doc=116517 (дата звернення: 03.03.2026).

15. Kumar V., Reinartz W. Customer Relationship Management: Concept, Strategy, and Tools : 3rd ed. Berlin : Springer, 2018. 314 p. URL: https://books.google.com/books/about/Customer_Relationship_Management.html?id=wBLYtNotoE0C (дата звернення: 03.03.2026).

16. Nguyen T. H., Sherif J. S., Newby M. Strategies for Successful CRM Implementation. *Information Management & Computer Security*. 2007. Vol. 15, No. 2. P. 102-115. URL: https://doi.org/10.1108/09685220710748001 (дата звернення: 03.03.2026).

17. Про захист інформації в інформаційно-комунікаційних системах : Закон України від 05.07.1994 № 80/94-ВР. База даних «Законодавство України» / Верховна Рада України. URL: https://zakon.rada.gov.ua/go/80/94-%D0%B2%D1%80 (дата звернення: 05.03.2026).

18. Про електронні документи та електронний документообіг : Закон України від 22.05.2003 № 851-IV. База даних «Законодавство України» / Верховна Рада України. URL: https://zakon.rada.gov.ua/go/851-15 (дата звернення: 05.03.2026).

19. Sommerville I. Software Engineering : 10th ed. Boston : Pearson, 2015. 816 p.

20. SWEBOK Guide V4.0: Guide to the Software Engineering Body of Knowledge / ed. H. Washizaki. IEEE Computer Society, 2024. URL:

https://www.computer.org/education/bodies-of-knowledge/software-engineering (дата звернення: 14.03.2026).

21. ДСТУ ISO/IEC 25010 : 2016. Інженерія систем і програмних засобів. Вимоги до якості систем і програмних засобів та її оцінювання (SQuaRE). URL: https://online.budstandart.com/ua/catalog/doc-page.html?id_doc=69134 (дата звернення: 14.03.2026).

22. Connolly T., Begg C. Database Systems: A Practical Approach to Design, Implementation, and Management : 6th ed. Harlow : Pearson, 2015. 1440 p.

23. Fowler M. UML Distilled: A Brief Guide to the Standard Object Modeling Language : 3rd ed. Boston : Addison-Wesley, 2004. 208 p. URL: https://books.google.com/books/about/UML_Distilled.html?id=nHZslSr1gJAC (дата звернення: 22.03.2026).

24. About Node.js. *Node.js Official documentation*. URL: https://nodejs.org/en/about (дата звернення: 27.03.2026).

25. Node.js Web Application Framework. *Express.js*. URL: https://expressjs.com/ (дата звернення: 27.03.2026).

26. About SQLite. *SQLite Official documentation*. URL: https://sqlite.org/about.html (дата звернення: 02.04.2026).

27. HTML: HyperText Markup Language. *MDN Web Docs*. URL: https://developer.mozilla.org/en-US/docs/Web/HTML (дата звернення: 07.04.2026).

28. Embedded JavaScript Templates. Official documentation. *EJS*. URL: https://ejs.co/ (дата звернення: 07.04.2026).

ДОДАТОК А

Лістинг головного файла запуску CRM-системи

```
const path = require('path');
const express = require('express');
const session = require('express-session');
const { initializeDatabase } = require('./db');
const { exposeUser } = require('./middleware/auth');

const authRoutes = require('./routes/auth');
const dashboardRoutes = require('./routes/dashboard');
const clientsRoutes = require('./routes/clients');
const propertiesRoutes = require('./routes/properties');
const requestsRoutes = require('./routes/requests');
const showingsRoutes = require('./routes/showings');
const dealsRoutes = require('./routes/deals');
const reportsRoutes = require('./routes/reports');
const adminRoutes = require('./routes/admin');

const app = express();
const PORT = process.env.PORT || 3000;

app.set('view engine', 'ejs');
app.set('views', path.join(__dirname, 'views'));

app.use(express.urlencoded({ extended: true }));
app.use(express.json());
app.use(express.static(path.join(__dirname, 'public')));
app.use(
  session({
    secret: process.env.SESSION_SECRET || 'crm-real-estate-session-secret',
    resave: false,
    saveUninitialized: false,
    cookie: { maxAge: 1000 * 60 * 60 * 8 }
  })
);
app.use(exposeUser);

app.use('/', authRoutes);
app.use('/', dashboardRoutes);
app.use('/clients', clientsRoutes);
app.use('/properties', propertiesRoutes);
app.use('/requests', requestsRoutes);
app.use('/showings', showingsRoutes);
app.use('/deals', dealsRoutes);
app.use('/reports', reportsRoutes);
app.use('/admin', adminRoutes);

app.use((req, res) => {
  res.status(404).render('error', {
    title: 'Сторінку не знайдено',
    message: 'Запитана сторінка відсутня або була переміщена.'
  });
});

app.use((err, req, res, next) => {
  console.error(err);
  res.status(500).render('error', {
```

```
    title: 'Помилка сервера',
    message: 'Під час оброблення запиту сталася помилка. Перевірте введені
дані або повторіть дію.'
  });
});

initializeDatabase()
  .then(() => {
    app.listen(PORT, () => {
      console.log(`CRM система запущена: http://localhost:${PORT}`);
    });
  })
  .catch((error) => {
    console.error('Не вдалося ініціалізувати базу даних:', error);
    process.exit(1);
                                        });
```