

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра Інформатики
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

РОЗРОБКА ВЕБЗАСТОСУНКУ ДЛЯ РІЄЛТОРСЬКОЇ КОМПАНІЇ
(тема)

Виконав:
студент 4 курсу, групи ІТІНФ-19-1

Шапошніков Д.А.
(прізвище, ініціали)

Спеціальності 122 Комп'ютерні науки
(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва освітньої програми)

Керівник доц. Руденко Д.О.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____
(підпис)

Кобилін О.А.
(прізвище, ініціали)

2023 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)Кафедра Інформатики
(повна назва)Рівень вищої освіти перший (бакалаврський)Спеціальність 122 Комп'ютерні науки
(код і повна назва)Тип програми освітньо-професійнаОсвітня програма Інформатика
(повна назва освітньої програми)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

« _____ » _____ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУстудентові Шапошнікову Давиду Артуровичу
(прізвище, ім'я, по батькові)1. Тема роботи Розробка вебзастосунку для ріелторської компанії

затверджена наказом університету від 15 травня 2023 року № 474 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 27 травня 2023 р.

3. Вихідні дані до роботи науково-методична та науково-технічна література, матеріали конференцій, дані інтернет-мережі.

4. Перелік питань, що потрібно опрацювати в роботі _____

1. Аналіз вебсайтів з продажу нерухомості, виявлення їх переваг та недоліків.

2. Інформаційна модель ріелторської компанії.

3. Реалізація вебзастосунку для ріелторської компанії.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) Актуальність проблеми ріелторської компанії, недоліки вебсайтів ріелторських компаній, постановка задачі, кінцевий результат.

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Консультант з дотримання діючих стандартів та норм	Доцент Творошенко І.С.		

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	10.04.2023	
2	Аналіз завдання, підбір літератури	11.04.23-17.04.23	
3	Аналіз літератури з досліджуваної проблеми	18.04.23-20.04.23	
4	Аналіз технічних засобів	21.04.23-30.04.23	
5	Розробка методу	01.05.23-04.05.23	
6	Програмна реалізація	05.05.23-20.05.23	
7	Оформлення пояснювальної записки	24.05.23-26.05.23	
8	Перевірка на плагіат	27.05.23	
9	Рецензування	28.05.23	
10	Підготовка презентації та доповіді	29.05.23-30.05.23	
11	Занесення роботи в електронний архів	31.05.23	
12	Попередній захист кваліфікаційної роботи	05.06.23	

Дата видачі завдання 10 квітня 2023 р.

Студент _____
(підпис)

Керівник роботи _____ доц. Руденко Д.О.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ/ABSTRACT

Пояснювальна записка до кваліфікаційної роботи: 71 с., 46 рис., 33 джерела.

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, РІЄЛТОРСЬКА КОМПАНІЯ, ВЕБЗАСТОСУНОК, ІНФОРМАЦІЙНА СИСТЕМА, PHP, JS, CSS.

Об'єктом роботи є вебзастосунок для рієлторської компанії.

Метою роботи є розробка вебзастосунка для рієлторської компанії, аби полегшити процес винаймання та купівлі нерухомості.

Було проведено ретельне дослідження усіх можливих прикладів вебзастосунків для ринку рієлторських компаній, на підставі чого було виявлено їх основні переваги та недоліки. Була побудована інформаційна модель рієлторської компанії.

У результаті роботи була здійснена програмна реалізація вебзастосунку для рієлторської компанії.

SOFTWARE, REAL ESTATE COMPANY, WEB APPLICATION, INFORMATION SYSTEM, PHP, JS, CSS.

The object of the work is a web application for a real estate company.

The purpose of the work is to develop a web application for a real estate company to facilitate the process of renting and buying real estate.

A thorough study of all possible examples of web applications for the market of real estate companies was carried out, on the basis of which their main advantages and disadvantages were identified. An information model of a real estate company was built.

As a result of the work, a software implementation of a web application for a real estate company was carried out.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів	6
Вступ.....	7
1 Аналіз предметної галузі і постановка задачі	8
1.1 Загальні відомості з предметної галузі	8
1.2 Переваги та недоліки вебсайту агенства нерухомості.....	9
1.3 Приклади вебзастосунків для продажу нерухомості	11
1.4 Постановка задачі	19
2 Проектування та розробка програмного продукту	20
2.1 Опис використаних технологій та мов програмування	20
2.2 Моделювання інформаційної системи.....	21
2.3 Опис інтерфейсу вебзастосунку	26
3 Комп'ютерна модель вебзастосунку для ріелторської компанії.....	41
3.1 Обґрунтування вибору середовища програмної реалізації	41
3.2 Програмна реалізація.....	44
Висновки	67
Перелік джерел посилання	68

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

БД – база даних

HTML – HyperText Markup Language (мова розмітки гіпертексту)

SQL – Structured query language (мова структурованих запитів)

CSS – Cascading Style Sheets (каскадні таблиці стилів)

JS – JavaScript

PHP – Hypertext Preprocessor (гіпертекстовий препроцесор)

ВСТУП

У наш час дуже швидко розвивається ринок нерухомості. Зростання рівня життя сприяє збільшенню кількості населення. У сучасному світі, коли люди прагнуть до комфортного життя, один із основних аспектів є собі забезпечення житлом. Процес придбання у власність земельної ділянки, приватного будинку або квартири є відповідним. Саме тому багато людей звертаються до ріелторських організацій з продажу, які спеціалізуються на купівлі та нерухомості. Наявність великих баз даних та система відбору найкращих варіантів має вирішальне значення для побудови довіри клієнтам до ріелтора та компанії. З настанням ери Інтернету пошук надійного ріелтора став набагато легший, тому що за короткий проміжок часу клієнт може знайти безліч варіантів ріелторських компаній. Це полегшує обрати дійсно обаяного ріелтора або ріелторську компанію, яка буде відповідати усім вимогам клієнта задля досягнення найкращого результату.

Ця проблема, здається, ніколи ще не була настільки актуальна, як після повномасштабного вторгнення країни агресора. Нажаль через втрату домівок та переїзду в інші міста або країни люди були вимушені винаймати житло і робити це складно через сайти оголошень, бо можна з легкістю натрапити на шахраїв. У той час, як можна знайти у пошуку «Google» вебсайт певної ріелторської компанії та звільнити свій час та думки від пошуку дійсно справжніх оголошень.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ І ПОСТАНОВКА ЗАДАЧІ

1.1 Загальні відомості з предметної галузі

Агентство нерухомості – це компанія, яка спеціалізується на купівлі, продажу, оренді та управлінні нерухомістю. Основні функції агентства нерухомості включають пошук клієнтів, проведення переговорів та складання договорів з продавцем або покупцем, а також допомогу в оформленні всіх документів.

Агентство нерухомості може надавати різноманітні послуги у зв'язку з потребами клієнта. Наприклад, агентство може займатися орендою або будинками, продавати комерційні приміщення, проводити оцінку вартості нерухомості та здійснювати дії по її управлінню, таким як обслуговування та ремонт.

Можна зробити висновок, що дана галузь діяльності стала необхідною складовою життя людей. Багато ріелторських агентств розробили власні вебзастосунки для пошуку нерухомості та додатково володіють командою спеціалістів, які підтримують технічну роботу сайту та надають допомогу клієнтам у продажу нерухомості.

Вебзастосунок – це програма, що виконується на вебсервері, на відміну від класичних десктопних застосунків, що опрацьовуються операційною системою на пристрої користувача. Клієнт може отримати доступ до вебзастосунків за допомогою веббраузера [1–12].

Основна перевага вебзастосунків полягає в тому, що користувачеві не потрібно завантажувати на свій пристрій нічого крім самого клієнтського інтерфейсу застосунку, що економить і час, і пам'ять. Не треба нічого встановлювати, окрім веббраузера, хоча з сучасними операційними системами в комплекті зазвичай браузер постачається.

Інша перевага полягає в тому, що для поширення оновлень вебзастосунку розробнику треба лише оновити застосунок на вебсервері, після чого всім користувачам буде доступна остання версія

Використання вебсервера дозволяє запобігти збереженню даних клієнта на його локальній системі, що дає можливість отримати до них доступ з будь-якого пристрою, на якому встановлений веббраузер. Також таким чином ліквідується можливість втрати даних через несправність пристрою користувача.

Більшу частину обчислень бере на себе вебсервер, тому загальне навантаження на систему користувача значно зменшується в порівнянні з десктопними застосунками [12–23].

Однак, варто зазначити наявність деяких недоліків. Основним серед них є обмеження доступності вебзастосунків у випадку відключення підключення до Інтернету. Крім того, якщо у клієнта низька швидкість Інтернету, це може суттєво уповільнити процес завантаження вебзастосунка.

Через те, що більшу частину обчислень в вебзастосунках бере на себе вебсервер, розробникам не вигідно створювати важкі за обчислюваннями застосунки в вебформі, бо щоб досягти швидкої роботи програми для багатьох користувачів потрібні дуже потужні та дорогі сервери.

Наразі глобальний ринок неможливо собі уявити без сучасних технологій, які полегшують нам життя. На жаль, у нас в Україні онлайн-послуги тільки почали завойовувати свою популярність.

1.2 Переваги та недоліки вебсайту агентства нерухомості

В даній сфері єдиним достойним конкурентом вебзастосунків є ріелторські агентства, проте з поширенням можливостей доступу в Інтернет вони витісняються цією більш зручною та дешевою альтернативою. Зараз

рієлтори самі активно використовують вебзастосунки для пошуку нерухомості або покупця нерухомості для своїх клієнтів.

Вебсайти агентств нерухомості мають деякі переваги перед традиційними агентствами нерухомості:

- доступність і зручність. Вебсайти агентств нерухомості можуть бути доступні для клієнтів з будь-якої точки світу і в будь-який час. Клієнти можуть шукати нерухомість, дивитися фотографії та відео, ознайомлюватися з характеристиками та цінами на нерухомість;

- можливості пошуку. Вебсайти агентств нерухомості надають клієнтам можливість шукати нерухомість за безкоштовними параметрами, такими як відвідування, розмір, кількість кімнат, ціна тощо. Це дозволяє клієнтам знайти варіанти, які відповідають їхнім потребам;

- широкий асортимент нерухомості. На вебсайтах агенств нерухомості можна знайти різні типи нерухомості, такі як квартири, будинки, земельні ділянки, комерційна нерухомість та інші;

- інформаційна база. Вебсайти агентств нерухомості можуть містити корисну інформацію для клієнтів, наприклад, про законодавство, пов'язані з купівлею і продажем нерухомості, про ринок нерухомості, про процес купівлі-продажу та інші;

- зменшення витрат. Вебсайти агентств нерухомості можуть зменшити витрати на рекламу та маркетинг, що дозволяє знизити вартість послуг для клієнтів;

- швидкість операцій. На вебсайтах агентств нерухомості можна швидше виконувати операції, частина документів можна відправляти онлайн, що скорочує час.

Основними пошуками нерухомості в індустрії стали користувачі смартфонів, тому це виправдовує необхідність створення платформи або мобільних застосунків, де користувачі необхідно переглядати та вибирати нерухомість за своїми вподобаннями. Втім, як уже зазначалося раніше, сайти

виступають більш функціональним ланцюжком у сфері пошуку житла, що дозволяє їм мати більше переваг.

1.3 Приклади вебзастосунків для продажу нерухомості.

Одним із провідних українських вебсайтів на ринку нерухомості є «dom.ria.com». За даними компанії-власника «RIA.com Marketplaces OÜ», на цьому сайті доступно понад 480 тисяч активних оголошень про продаж квартир і будинків. Вебзастосунок використовується як приватними особами цими, так і ріелторами. На сайті представлено майже 1500 новобудов від забудовників по всій території України. Крім того, за допомогою цього сайту клієнти можуть домовлятися про короткострокову тастрокову оренду, а також купівлю-продаж земельних ділянок.

Вебсайт та торгова марка «ria» є проєктом естонської компанії «RIA.com Marketplaces OÜ». Загалом «ria group» окрім «dom.ria» включає в себе «auto.ria» – сайт призначений для продажу авто, «hotels24.ua» – сайт для бронювання готелів, «MyRadio.com.ua» – онлайн радіо, «20.ua» – сайт-довідка з відгуками клієнтів про послуги різних компаній, «Moemisto.ua» – новинний портал для організації відпочинку в крупних містах України.

Повертаючись до вебзастосунку dom.ria.com, при потраплянні на головну сторінку (рис. 1.1), користувача просять вказати місто, тип нерухомості та тип угоди яку він або вона хоче в перспективі укласти. Також одразу пропонується вибрати між новобудовами та вторинним ринком нерухомості.

Після цього користувач потрапляє на сторінку пошуку (рис. 1.2), де йому надається можливість скористатися чисельними фільтрами, за умовами яких відображається список з об'єктів нерухомості. В цьому списку показана основна інформація про об'єкт: ціна, адреса розташування, кількість кімнат, площа, короткий опис та, для наочності, дві головні фотографії об'єкту. З

цього екрану клієнт має можливість додати об'єкт в обране, натиснувши на серце справа від оголошення. Також цей список припускає можливість сортування за параметрами новизни та ціни.

Замість списку, за бажанням користувача, об'єкти можуть бути відображені у вигляді галереї та на мапі місцевості.

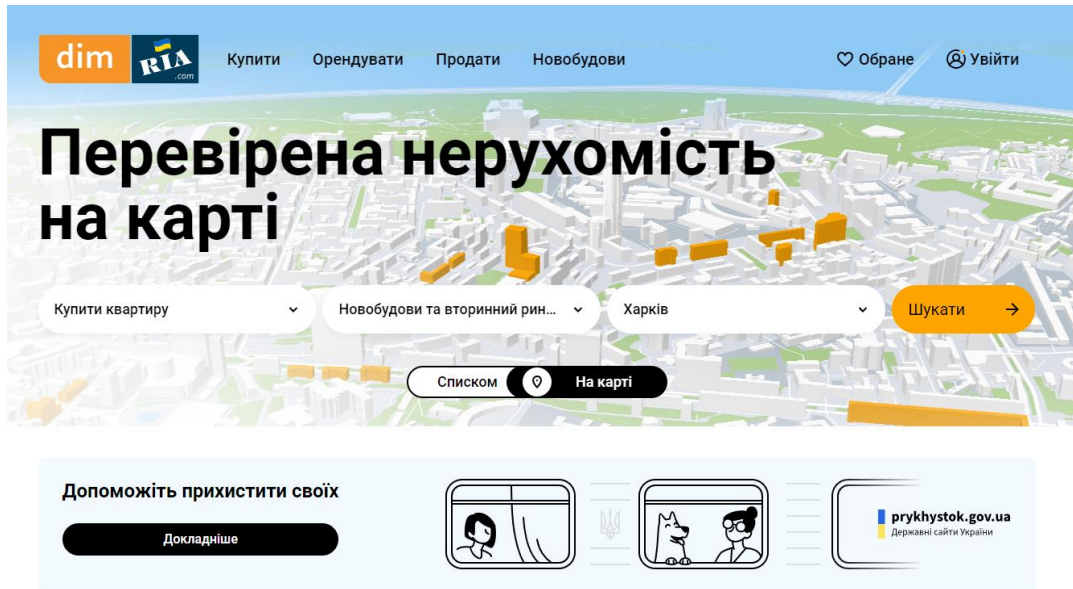


Рисунок 1.1 – Головна сторінка сайту dom.ria.com

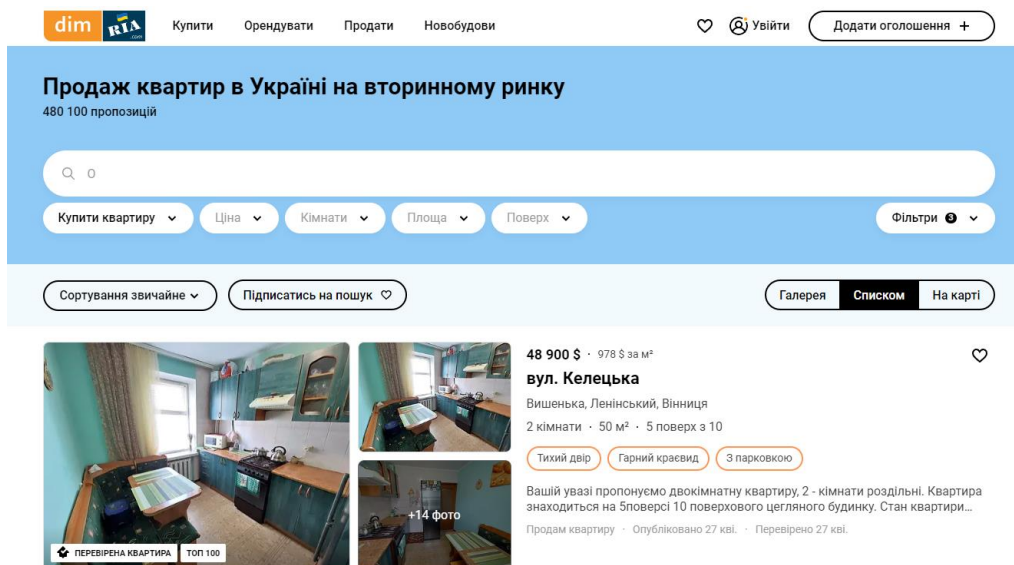


Рисунок 1.2 – Сторінка пошуку dom.ria.com

Після вибору відповідного об'єкта за заданими критеріями користувач може перейти на окрему сторінку цього оголошення (рис. 1.3). На цій сторінці представлений ряд сторінок, зокрема:

- перегляд усіх фотографій, пов'язаних з об'єктом;
- перегляд точного місця нерухомості на електронній карті;
- взаємодія з власником або ріелтором через онлайн-чат або телефонний дзвінок;
- збереження додаткової інформації з повного опису об'єкта, який міститься на сторінці;
- перегляд портфоліо ріелтора на зовнішній сторінці;
- оцінка рейтингу району, де знаходиться нерухомість, на основі відгуків інших користувачів сайту.

Ці функції сприяють більш детальному огляду з обраним об'єктом нерухомості та забезпечують зручний спосіб комунікації з власником або ріелтором для отримання необхідної інформації. Оформлення сайту відповідає високим стандартам сучасної розробки, функціонал вебзастосунку один з найобширніших на ринку.

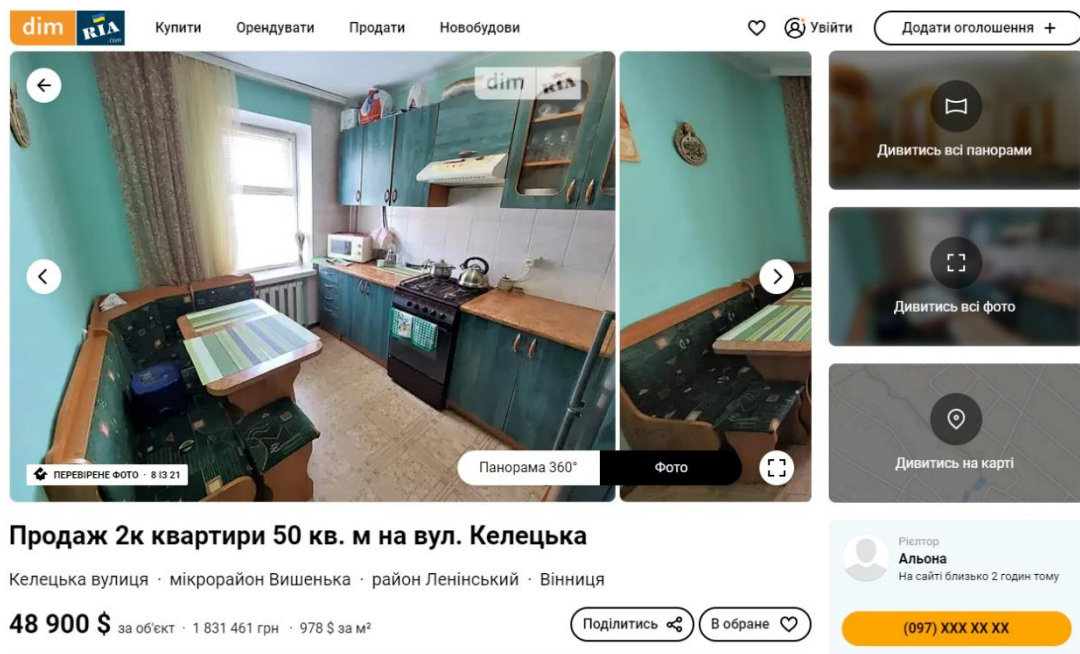


Рисунок 1.3 – Сторінка оголошення dom.gia.com

Переваги вебзастосунку з продажу нерухомості «dom.ria.com»:

- три вигляди відображення оголошень на сторінці пошуку: список, галерея та мапа;
- можливість додати в обране;
- представлення ціни в різних валютах;
- онлайн чат;
- наявність сторінки продавця;
- відповідність високим сучасним стандартам якості вигляду клієнтського інтерфейсу.

Серед недоліків вебзастосунка з продажу нерухомості «dom.ria.com» можна визначити, що сайт має забагато фільтрів, які не уміщуються в одну сторінку.

Іншим вебзастосунком в галузі нерухомості є ріелторське агенство «Благовіст». Засноване в 1993 році, це агенство пропонує широкий спектр житлової, комерційної та замиської нерухомості. протягом періоду з 2005 по 2011 рік агенство було визнано 7 разів переможцем премії «Вибір року» в номінації «Агенство нерухомості року». За власними даними «Благовісту», їх база об'єктів нерухомості є найбільшою в Києві та області. Генеральний директор агентства стверджує, що їх частка становить 55%. Вебзастосунок агентства містить оголошення з об'єктами нерухомості розташованими лише в Києві та області.

На головній сторінці сайту користувачів одразу поділяють на три категорії: покупець, продавець та орендатор (рис. 1.4).

На сторінці продажу нерухомості користувач має вибрати або вид нерухомості, район розташування об'єкту, в випадку квартир – кількість кімнат або в випадку комерційної нерухомості – її тип (рис. 1.5).

Далі користувачу пропонується задіяти потрібні фільтри, вибрати валюту. Надається можливість пошуку за кодом об'єкту та можливість

надіслати заявку ріелтору, щоб за складеним набором фільтрів клієнту підібрали відповідну нерухомість (рис. 1.6).

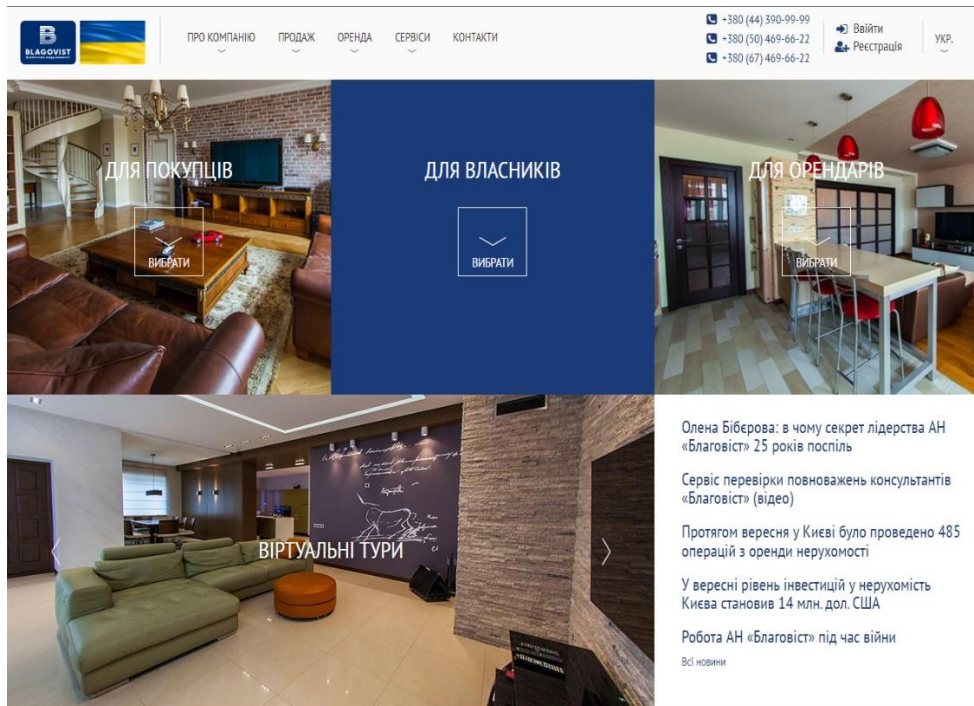


Рисунок 1.4 – Головна сторінка сайту blagovist.ua

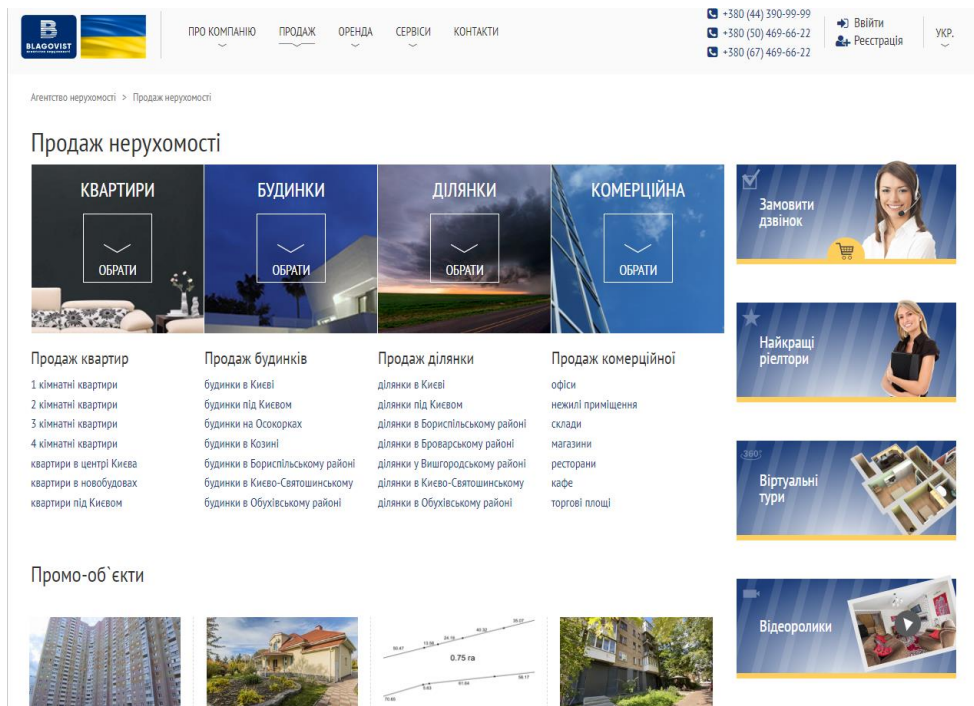


Рисунок 1.5 – Сторінка продажу нерухомості blagovist.ua

Рисунок 1.6 – Сторінка фільтру пошуку blagovist.ua

Після застосування фільтрів користувачу надається список об'єктів нерухомості, відповідних заданим критеріям, а також інтерактивна карта, на якій відображається їх розташування. У списку кожне оголошення має лаконічний опис об'єкту, який включає такі дані: кількість кімнат, точну адресу, ціну, розташування, три заголовні фотографії, код об'єкта та коротку характеристику нерухомості. Клієнту надається можливість сортувати список за ціною та площею об'єктів. Без того, на відміну від інших вебзастосунків, в описі об'єкта обов'язково присутнє планування, яке наглядно демонструє структуру приміщення (рис. 1.7).

Після вибору об'єкту нерухомості користувач може переглянути його повний опис на окремій сторінці оголошення (рис. 1.8). Тут він може переглянути всі фотографії нерухомості в більшому розмірі та збільшену мапу з його місцезнаходженням. У описі на цій сторінці вказані всі головні характеристики об'єкту, які можна було побачити в списку, а також додаткова інформація, така як найближча станція метро, площа квартири (житлова,

загальна, кухонна), поверх розташування, тип будинку та матеріал стін. Крім того, продавцем надаються додаткові дані про інфраструктуру в районі нерухомості та контактні телефонні номери. На цій же сторінці користувач може замовити дзвінок від ріелтора-продавця та перейти на його особисту сторінку на сайті.

Продаж квартир Кібцентр

Купити квартиру Кібцентр Київ

Всього знайдено: 2 Спочатку нові

3 кім. квартира 75 м², вул. Жулянська, 8 корпус 2, Київ

2 484 000* грн. \$ €

Кібцентр, Голосіївський район

Житловий комплекс: Абрикосовий



Трикімнатна квартира в сучасному жк абрикосовий в найзеленішому районі киева – голосіївському, та ек...

3 кім. квартира 75 м², вул. Жулянська, 8 корпус 2, Київ

2 910 000* грн. \$ €

Кібцентр, Голосіївський район

Житловий комплекс: Абрикосовий



Жк "абрикосовий". комплекс збудований, встановлені ліфти, виконуються фасадні роботи та внутрішнє оз...

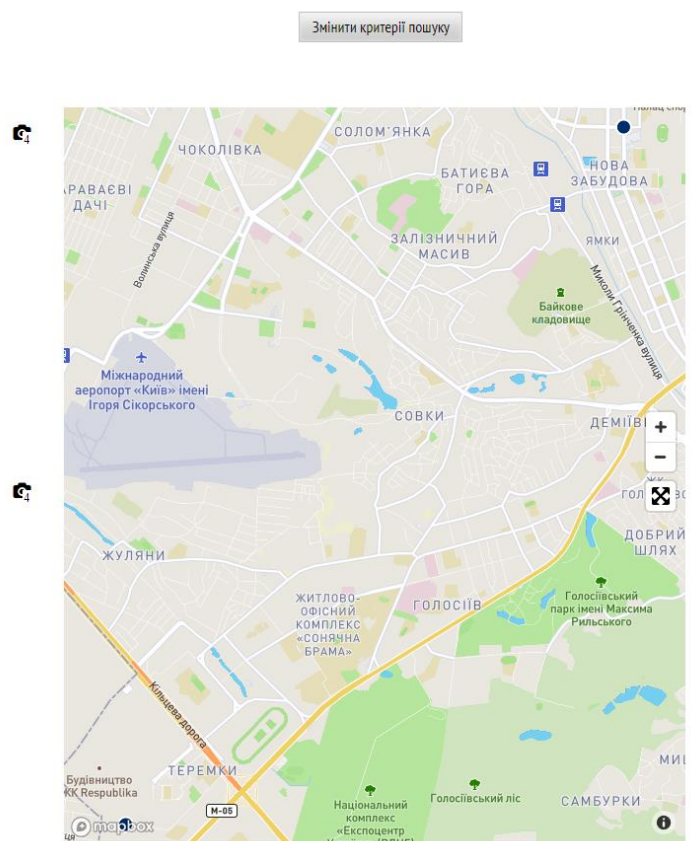


Рисунок 1.7 – Сторінка пошуку blagovist.ua

Переваги вебзастосунка з продажу нерухомості «blagovist.ua»:

- два вигляди відображення оголошень на сторінці пошуку: список та мапа;
- представлення ціни в різних валютах;
- наявність сторінки продавця;
- всі оголошення оформлені за одним шаблоном.

Продаж > Трикімнатна квартира, вул. Жулянська 8 корпус 2 | R-48178



2 484 000* грн. \$ €

Код об'єкту: R-48178

Замовити дзвінок

Рієлтор: Василь Миколайович

Телефон: +380 (44) 498-60-07

+380 (50) 424-40-90

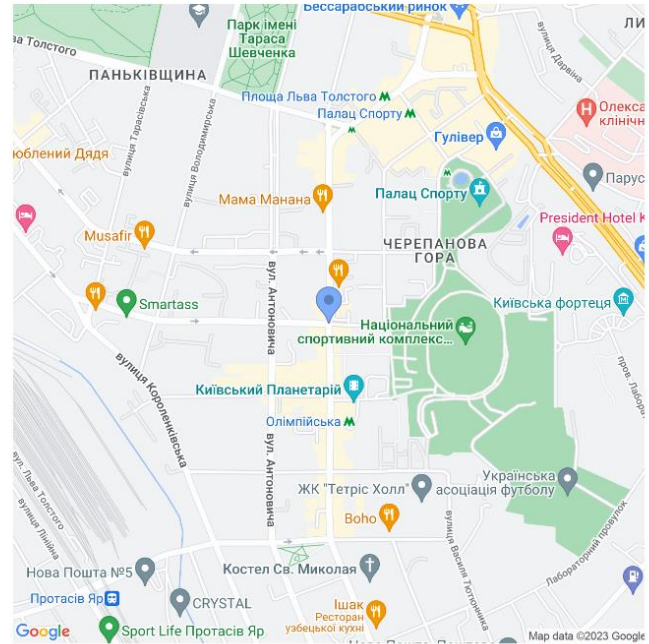
+380 (67) 332-62-46

Вулиця	Жулянська, 8 корпус 2
Житловий комплекс	Абрикосовий
Розташування	Кібіцентр, Голосівський район, Київ
Кількість кімнат	3-кімн.
Площа (загальна/житлова/кухні)	75/40/13
Поверх/Поверхів	21/26
Тип (серія) будинку	Спецпроект 2021-2025 рр
Матеріал стін	Газоблок

Трикімнатна квартира в сучасному ЖК Абрикосовий в найзеленішому районі Києва – Голосівському, та екологічно чистій місцевості. Планування квартири комфортне та зручне, а монолітно-каркасна технологія будівництва дозволяє власникам помешкання зробити перепланування на свій смак та стиль. До станції метро "Теремки" йти 12 хвилин. Поблизу велика кількість шкіл і садочків приватних і державних. Навколо комплексу з одного боку котеджі, а з іншого - низькоповерхові будинки та лісопаркова смуга.

ЖК "Абрикосовий", 2 будинки

План-схема об'єкта нерухомості представлена на основі типових планів приміщень в будівлі. Площі на план-схемі можуть мати незначні відхилення від площ, зазначених в технічному паспорті об'єкту.



Кількість переглядів об'єкта за останні 30 днів : 7



Рисунок 1.8 – Сторінка оголошення blagovist.ua

Недоліки вебзастосунка з продажу нерухомості «blagovist.ua»:

- винесення фільтру «Покупець, власник, орендатор» на окрему сторінку;
- не відповідність сучасним стандартам якості вигляду клієнтського інтерфейсу;
- відсутність можливості додати в обране;
- відсутність онлайн месенджера.

1.4 Постановка задачі

На основі проведеного ретельного аналізу актуальним завданням є розробка вебзастосунка для ріелторської компанії.

Об'єктом роботи є вебзастосунок для ріелторської компанії.

Метою роботи є розробка вебзастосунка для ріелторської компанії, аби полегшити процес винаймання та купівлі нерухомості.

Для досягнення мети необхідно вирішити такі завдання:

- провести аналіз сайтів з продажу нерухомості та виявити їх переваги і недоліки, що було опрацьовано вище;
- побудувати інформаційну модель агенства нерухомості;
- реалізувати вебзастосунок для ріелторської компанії;
- провести тестування.

2 ПРОЄКТУВАННЯ ТА РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ

2.1 Опис використаних технологій та мов програмування

Для створення структури всієї програми використовувалися такі мови програмування як: JS, PHP, SQL, HTML і CSS.

Варто позначити кожен мову окремо [24–26]:

– JS (JavaScript) – це мова програмування, яка використовується в першу чергу для більш складних цілей в документі HTML. JS – це мова, яка дозволяє змінювати інтерфейси вебзастосунків, додавати або змінювати різні функціональні можливості сайту. Сторонні API можуть дозволити додавати різні функції, які вже існують на інших сайтах. Застосування HTML документу сторонні фреймворки і бібліотеки, дозволить прискорити створення сайтів і застосунків ;

– PHP (Hypertext Preprocessor) – це мова програмування, яка потрібна для статистичного відображення сторінки, а також для «довголіття» сайту. Тобто, дана мова використовується для щоденного оновлення сайту, надходження нової свіжої інформації, яка буде актуальна кожен день для користувачів. Дана мова програмування використовується для написання «сценаріїв», які завантажуються в HTML документ, щоб відобразити нестатичність картинки. Справа в тому, що документ HTML здатний відобразити тільки статичну інформацію на сайті, а PHP показує «свіжу»;

– SQL (Structured Query Language) – це мова програмування, яка може використовуватися як ефективний спосіб для таких функцій, як: збереження в БД, пошуку їх частин, поновлення, вилучення з бази даних і видалення звітти. Взаємодія з БД відбувається швидко навіть в ситуаціях, коли обсяги даних при завантаженні досить великі. SQL – це досить зручна мова для великої кількості інформації, так як дозволяє миттєво розробнику перерозподіляти інформацію;

– HTML (HyperText Markup Language) – це мова програмування, яка дозволяє розробнику створювати і структурувати безліч розділів, заголовків, посилань і цілі блоки для вебсторінок і застосунків. Це мова, яка дозволяє створювати структуру вебсторінок;

– CSS (Cascading Style Sheets) – це скоріше не мова програмування, а правила оформлення HTML документа. CSS дозволяє розробнику встановлювати розмітку сторінки, розмір шрифтів, колір, заголовки, а також змінювати шрифт. Можна сказати, що CSS безпосередньо відповідає за візуальну складову сайту, так як жоден HTML документ не може обійтися без CSS [1–3].

2.2 Моделювання інформаційної системи

Основною метою проєктування є забезпечення ефективного функціонування інформаційної системи. Для моделювання системи використовувалась методологія функціонального моделювання IDEF0, яка дозволяє зображати систему у вигляді взаємозалежних блоків. Контекстна діаграма функціонування інформаційної системи у нотації IDEF0 представлена на рисунку 2.1 і показує систему з точки зору співробітника агентства.

Головними складовими бізнес-процесу розроблюваної системи є [27–33]:

- процес: взаємодія з клієнтами агенства продажу нерухомості;
- вхідні дані: запит рієлтора;
- вихідні потоки: угода про продаж або відмова від співпраці з агенством;
- нормативи управління: Цивільний кодекс України;
- ресурси: рієлтор, база даних, вебзастосунок.

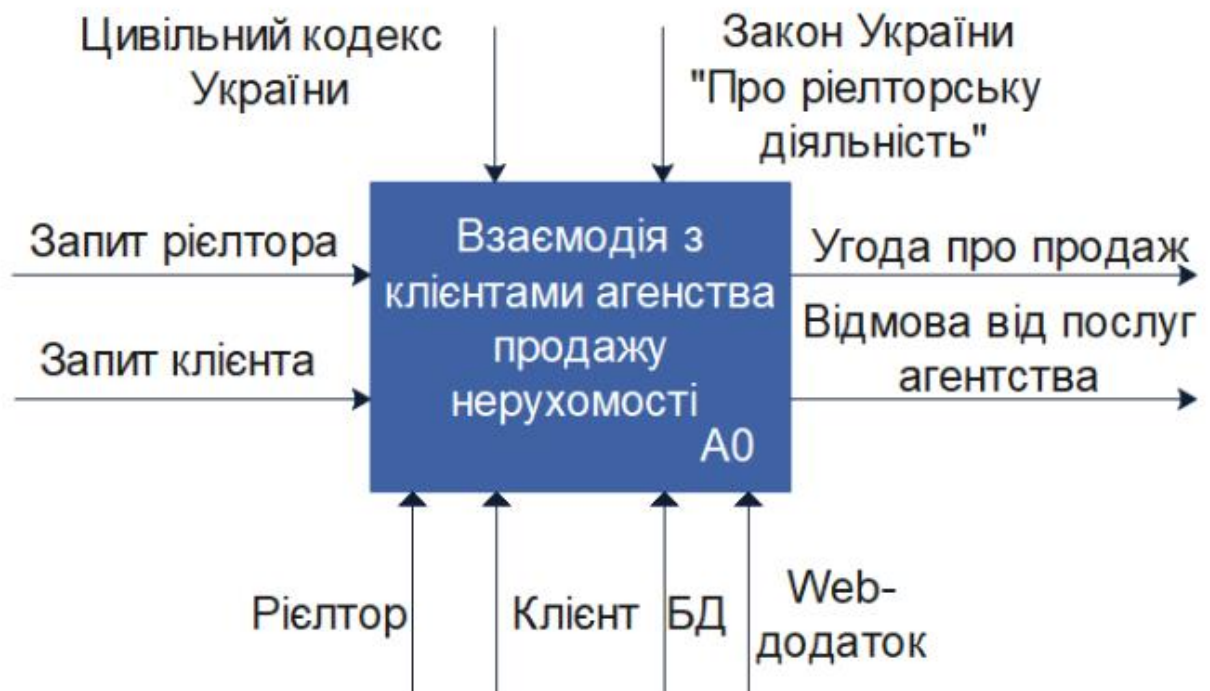


Рисунок 2.1 – Контекстна діаграма з точки зору співробітника

На рисунку 2.2 показана декомпозиція контекстної діаграми A0 в нотації IDEF0. Процес розпочинається з введення необхідних даних про клієнта, продавця та об'єкт нерухомості. Потім відбувається взаємодія між ріелтором і клієнтом шляхом надсилання знайдених пропозицій.

Діаграма декомпозиції контекстної діаграми A0 включає шість блоків, які взаємодіють між собою. Вхідними даними для блоків «Реєстрація продавця», «Додавання об'єкту» та «Реєстрація нового клієнта» є «Запит ріелтора». До блоку «Додавання об'єкту» також надходять дані, отримані після реєстрації продавця. Інформація про додану нерухомість та побажання клієнта є вхідними даними для блоку «Пошук співпадінь». Після надсилання пропозиції клієнту, у разі відмови пошук повторюється. Процеси пошуку та надсилання пропозиції є ітеративними, тому з'єднані стрілкою, що повертається з виходу блоку «Відправлення пропозиції клієнтові» на вхід блоку «Пошук співпадінь». Вихідні дані після виконання блоку «Відправлення пропозиції клієнтові» можуть передаватися на вхід до «Укладення договору»

або завершувати потік даних у випадку відмови клієнта від подальшої співпраці з агентством. Результатом укладання договору є «Угода про продаж».

Для визначення основних функціональностей системи були розроблені діаграми варіантів використання для трьох акторів: рієлтора, менеджера і клієнта. Ці діаграми варіантів використання використовуються для відображення дій, які зовнішні користувачі (актори) можуть виконувати у системі. На рисунку 2.3 наведена діаграма варіантів використання для актора рієлтора.

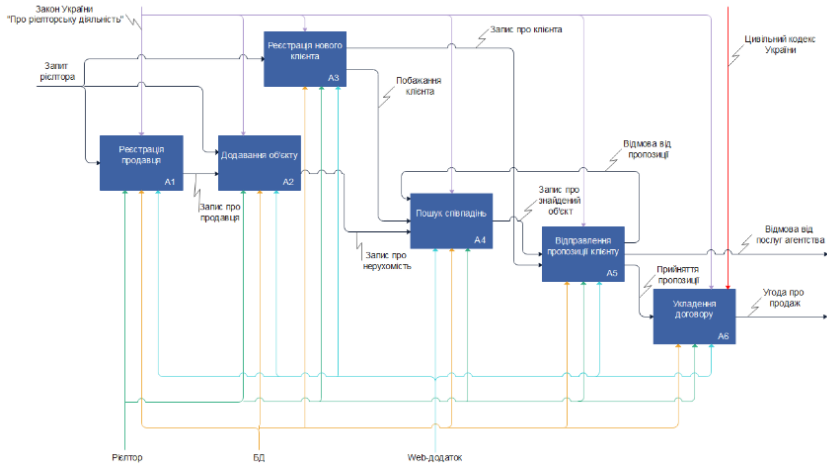


Рисунок 2.2 – Діаграма декомпозиції контекстної діаграми A0

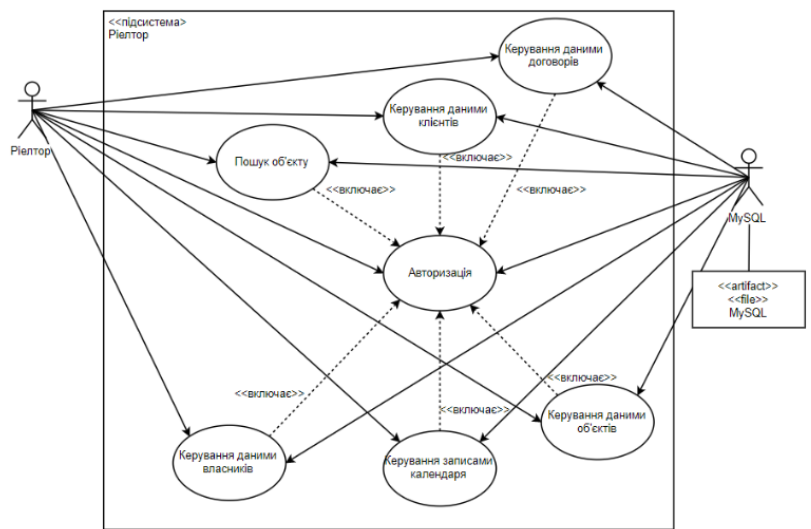


Рисунок 2.3 – Діаграма варіантів використання для актора рієлтора

На діаграмі виділено два актора: ріелтор та база даних MySQL. Варіанти використання для актора ріелтора на діаграмі:

- ВВ авторизація – використовуючи унікальний логін та пароль актор отримує доступ до системи;
- ВВ керування даними власників – дозволяє актору створювати, редагувати, переглядати та видаляти дані (create-read-update-delete (CRUD-функції)) про власників нерухомості;
- ВВ керування даними клієнтів – надає CRUD-функції для роботи з інформацією про клієнтів;
- ВВ керування даними об'єктів – надає CRUD-функції для роботи з інформацією про будинки та квартири;
- ВВ керування даними договорів – надає CRUD-функції для роботи з договорами;
- ВВ керування записами календаря – надає CRUD-функції для управління задачами та зустрічами у календарі;
- ВВ пошук об'єкта – дозволяє актору виконувати пошук пропозицій за побажанням клієнта.

Діаграма варіантів використання для актора ріелтора наведена на рисунку 2.4.

Діаграма варіантів використання включає два актори: менеджера і базу даних MySQL. Роль менеджера розширює можливості ріелтора за допомогою двох блоків варіантів використання:

- ВВ керування даними співробітників – надає CRUD-функції для роботи з інформацією про співробітників;
- ВВ призначення задачі ріелтору – дозволяє менеджеру створювати та призначати задачі ріелтору.

Діаграма варіантів використання для актора клієнта наведена на рисунку 2.5.

Головними акторами на діаграмі виступають: клієнт та база даних MYSQL. Варіанти використання наступні:

- ВВ авторизація – використовуючи унікальний логін та пароль клієнт отримує доступ до системи;
- ВВ перегляд даних – актор має можливість перегляд власних даних, інформацію про рієлтора, надіслані пропозиції та призначені зустрічі;
- ВВ зміна даних – актор може змінювати дані побажання та статус надісланих зустрічей та пропозицій.

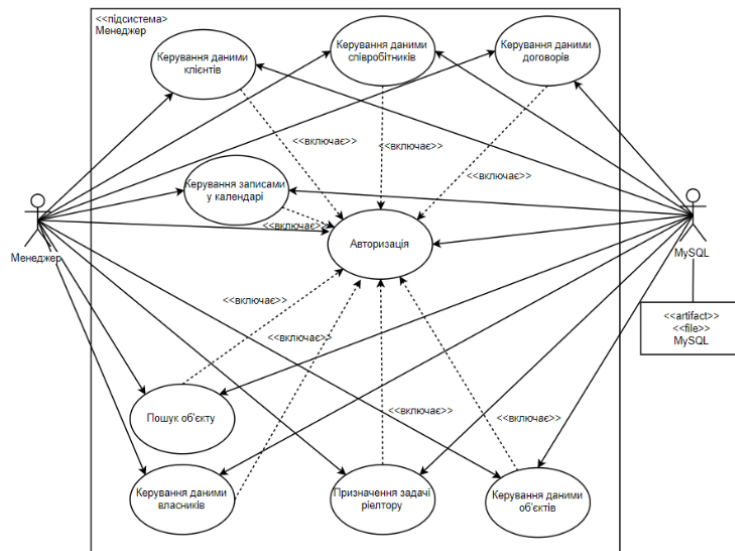


Рисунок 2.4 – Діаграма варіантів використання для актора менеджера

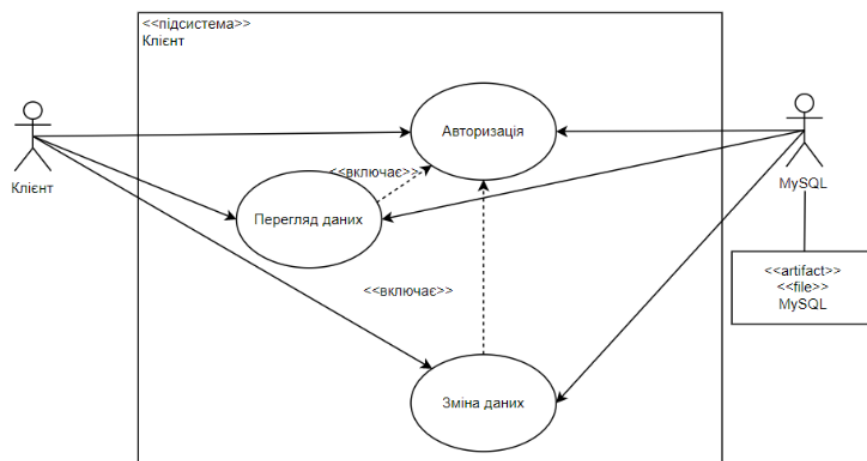


Рисунок 2.5 – Діаграма варіантів використання для актора клієнта

2.3 Опис інтерфесу вебзастосунка

Перше, що бачить користувач, коли заходить на вебсайт – це головна сторінка на якій є запропоновані пропозиції (рис. 2.6).

Зверху користувач бачить «шапку» сайту на якій є назва та розділи сайту.



Рисунок 2.6 – Головна сторінка вебсайту

Спускаючись униз гловної сторінки, клієнт побачить чотири блоки вебсайту: наші послуги, останні нерухомості, кращі агенти, відгуки клієнтів. Розглянемо їх окремо:

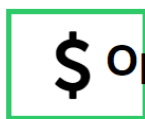
Перший блок – це «Наші послуги» (рис. 2.7). Тут можна дізнатись, які послуги пропонує це агенство нерухомості та його переваги.

Наші послуги



Орендарі

Покупці та потенційні орендарі є розчаровані у пошуку ідеального будинку. Ми можемо фільтрувати та звужувати пропозиції, та презентувати лише ті нерухомості, які задовольняють критерії наших клієнтів.



Оренда

Наша агенція з нерухомості почала посередництво у продажу та купівлі квартир, будинків та офісів, але оренда залишається нашою основною діяльністю та спеціалізацією.



Продаж

Наші клієнти можуть нам довіряти через нашу чесність, досвід та вміння, що зменшує виклики та невизначеності під час продажу або купівлі квартири, будинку чи офісу.

Рисунок 2.7 – Блок «Наші послуги»

Другий блок – це «Останні нерухомості» (рис. 2.8). Тут можна побачити останню нерухомість, яка буда додана на вебсайт та її вартість. При наведенні на нерухомість можна побачити площу, кількість спальних місць, ванних кімнат та гаражних приміщень цього будинку. Натиснувши на кнопку «Клікніть тут для перегляду» клієнт потрапить до сторінки обраної нерухомості (рис. 2.9).

Останні нерухомості

Всі нерухомості >

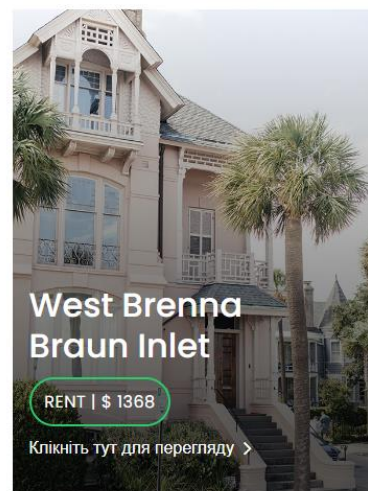
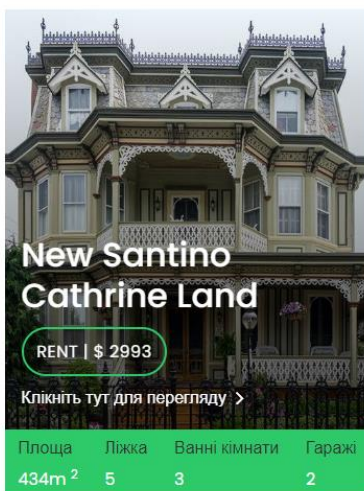


Рисунок 2.8 – Блок «Останні нерухомості»



954 Cathrine Land

New Santino

Головна / Нерухомості
/ 954 Cathrine Land**\$ 252000**

Короткий огляд

Місцезнаходження:	New Santino
Тип нерухомості:	Вілла
Status:	активне
Площа:	434m ²
Ліжка:	5
Ванні кімнати:	3
Гаражі:	2

Опис нерухомості

При виході з приватного ліфта на ліфтовій площадці гостей зустрічає простора вітальня, що виходить на схід, з дров'яним каміном і офіційною обідньою зоною. З кухні-їдальні відкривається чудовий вид на Центральний парк, вона має мармурову підлогу, стільниці та задню стінку, а також шафи з натурального дуба та білого лаку, поліровані хромовані світильники від Dornbracht, великий острів, барну стійку з видом на пієніч, комору для дворецького з мийкою та набір висококласної побутової техніки від Miele. Крім того, в головній вітальні та їдальні є пара вбиралень.

Зручності

- Безкоштовний Wi-Fi
- 24-годинна рецепція для гостей
- Багато стінних розеток
- Безкоштовні зарядні пристрої для електроніки
- Щоденні газети
- Безкоштовний сніданок
- Послуги прання
- Розваги
- Елегантні халати для кліяння

Рисунок 2.9 – Сторінка нерухомості

На сторінці нерухомості можна побачити короткий огляд дому: місцезнаходження, тип нерухомості та статус. В описі нерухомості користувач отримає інформацію про розташування, інтер'єр та вид з вікон. В зручностях описуються переваги цього дому.

Нижче користувач отримає інформацію про агента та його контактні дані (рис. 2.10).

Зв'язатися з агентом



Jack River

Один з найкращих агентів з нерухомості в Нью-Йорку. Його спеціальність зробила багатьох клієнтів щасливими. Він виконує свою роботу! Кар'єру розпочав як агент з нерухомості в вересні 2000 року, після того, як протягом 10 років був кредитором з нерухомості. Він знає бізнес зсередини та ззовні. Робоча етика Джона сильна, і він вирішив використовувати кожну можливу техніку, щоб допомогти своїм клієнтам знайти СВІЙ ідеальний будинок. Він пишається тим, що є одним з найкращих продажних агентів, а також навчає своїх клієнтів про процес нерухомості на кожному кроці.

Телефон: 097 5669 124

Електронна пошта: jack.river@email.com



Рисунок 2.10 – Блок «Зв'язатися з агентом»

Наступний блок головної сторінки – це «Кращі агенти». Тут клієнт має змогу отримати інформацію про кращих агентів вебсайту. При наведенні на фото одного з працівників агенства з'явиться невеликий опис агента та його контактні дані (рис. 2.11).

Четвертий блок головної сторінки вебзастосунку – це «Відгуки клієнтів». Тут показані відгуки клієнтів агенства (рис. 2.12).

Знизу головної сторінки розташовані контактні дані агенства, вхід до адміністрації вебсайта та посилання на соціальні мережі (рис. 2.13).

Кращі агенти

Всі агенти >

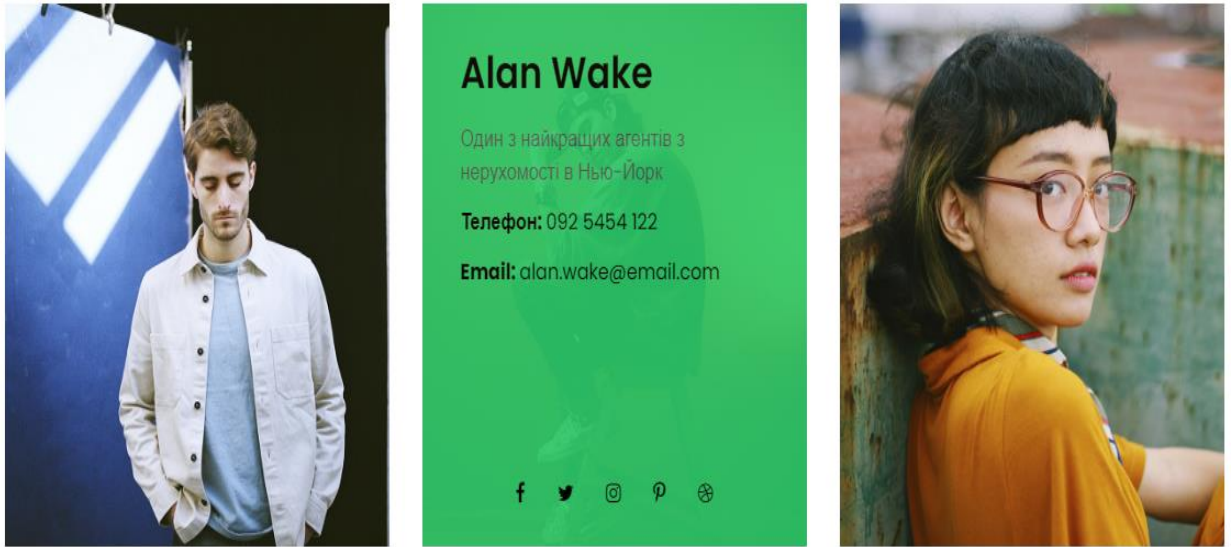


Рисунок 2.11 – Блок «Кращі агенти»

Відгуки клієнтів

<>



Завдяки Unlock Your Dream Home ми знайшли ідеальний будинок. Їхні агенти дуже професійні та чудові у своїй справі



Sasha & Lisa

Рисунок 2.12 – Блок «Відгуки клієнтів»

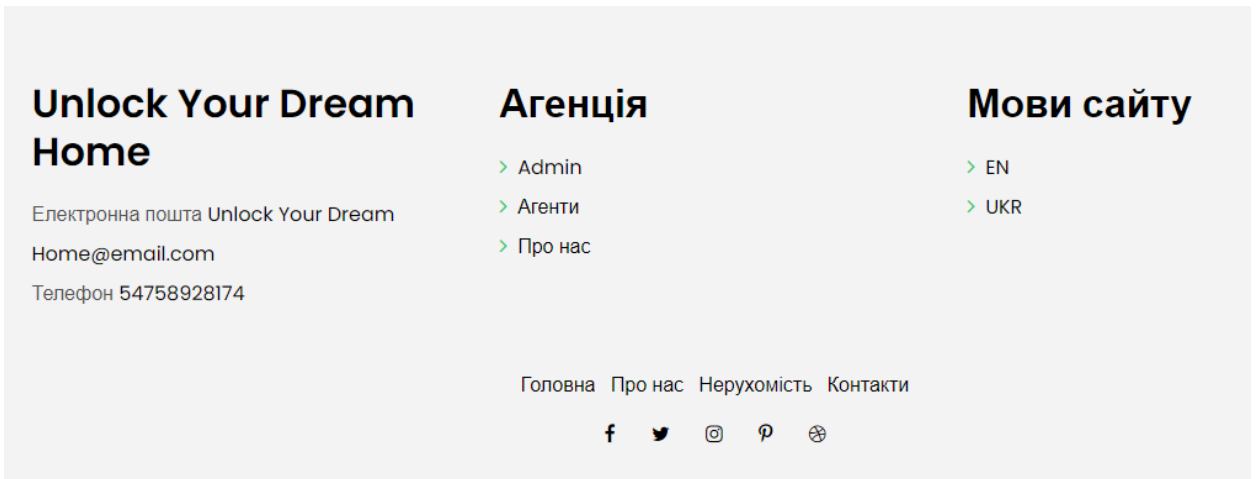


Рисунок 2.13 – Низ головної сторінки

На цьому головна сторінка вебсайту закінчилась. Перейдемо до наступної вкладки сайту – «Про нас» (рис. 2.14). На цій вкладці розповідається про агенство нерухомості (рис. 2.15).

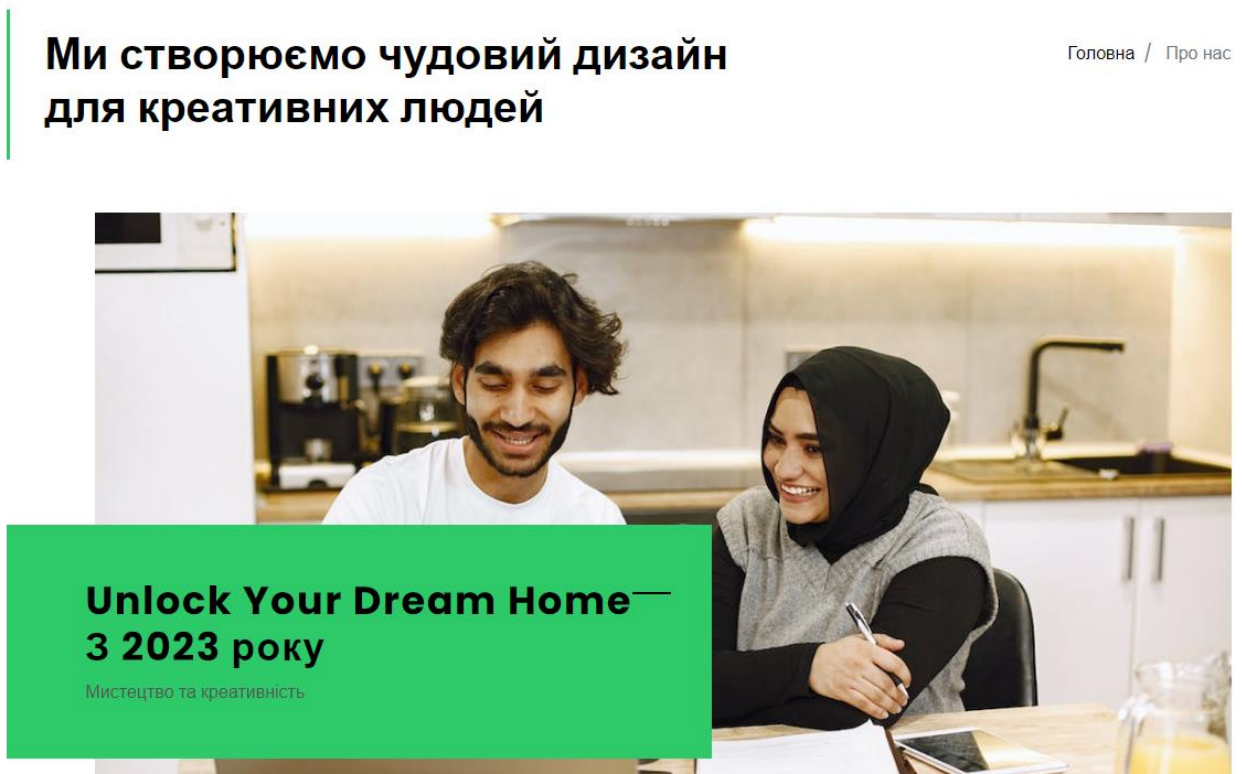


Рисунок 2.14 – Вкладка «Про нас» перша частина



Знайдіть свій мрійний будинок. Перегляньте наші ексклюзивні пропозиції.

Ми пропонуємо найвищий рівень експертизи, обслуговування та чесності.

Unlock Your Dream Home – це провідне агентство нерухомості в Мангеттені, яке допомогло сотням покупців знайти свій мрійний будинок в Нью-Йорку,

що призвело до закриття майже 1 мільярда доларів угод з нерухомістю в останні 10 років.

Ми пропонуємо послуги, пов'язані з розкішною нерухомістю: угоди купівлі-продажу, оренда та управління власністю.

З міжнародним клієнтелем та всесвітньою репутацією.

Знайомтеся з нашою командою

[Всі агенти >](#)



Рисунок 2.15 – Вкладка «Про нас» друга частина

Третя вкладка вебсайту – це «Нерухомість». На цій сторінці зібрані всі будинки агенства. На блоці кожного об'єкта нерухомості вказані його адреса та вартість (рис. 2.16). Над цим блоком є фільтр актуальності нерухомості, де можна подивитись продані або активні будинки (рис. 2.17).

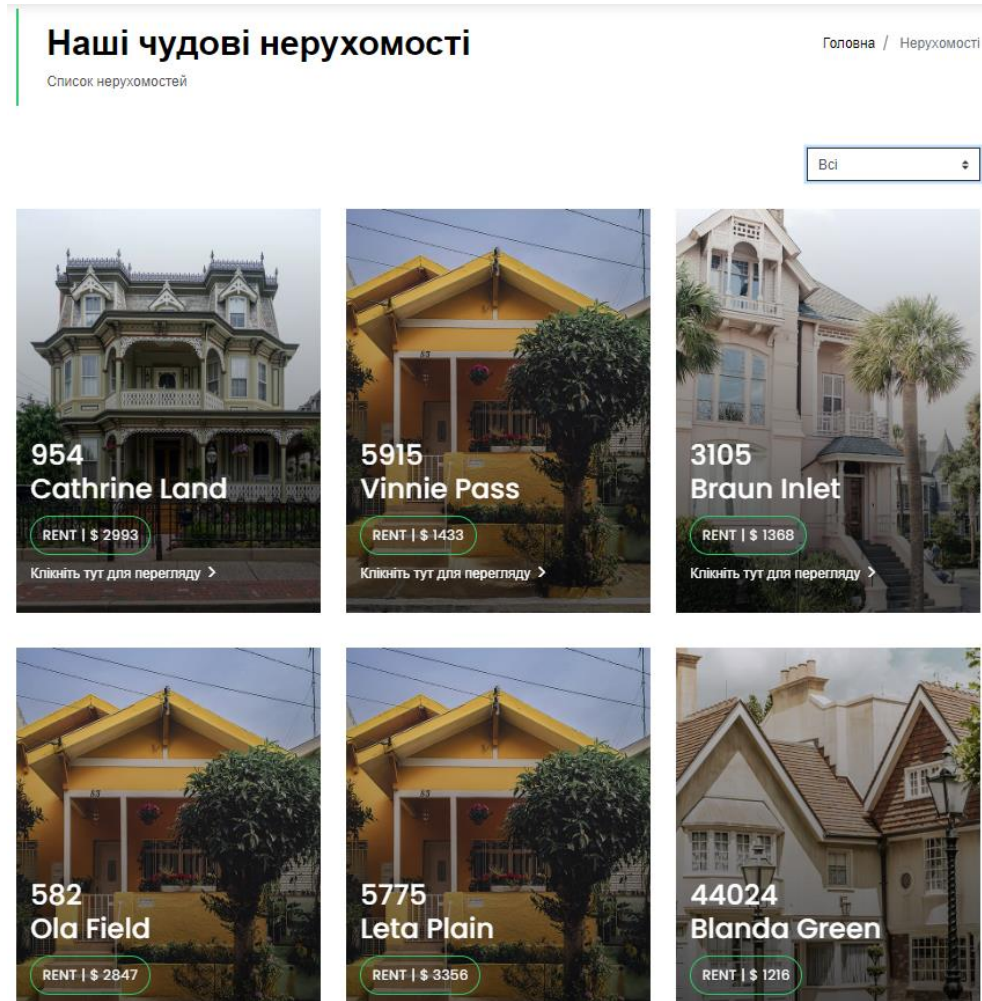


Рисунок 2.16 – Вкладка «Нерухомість»

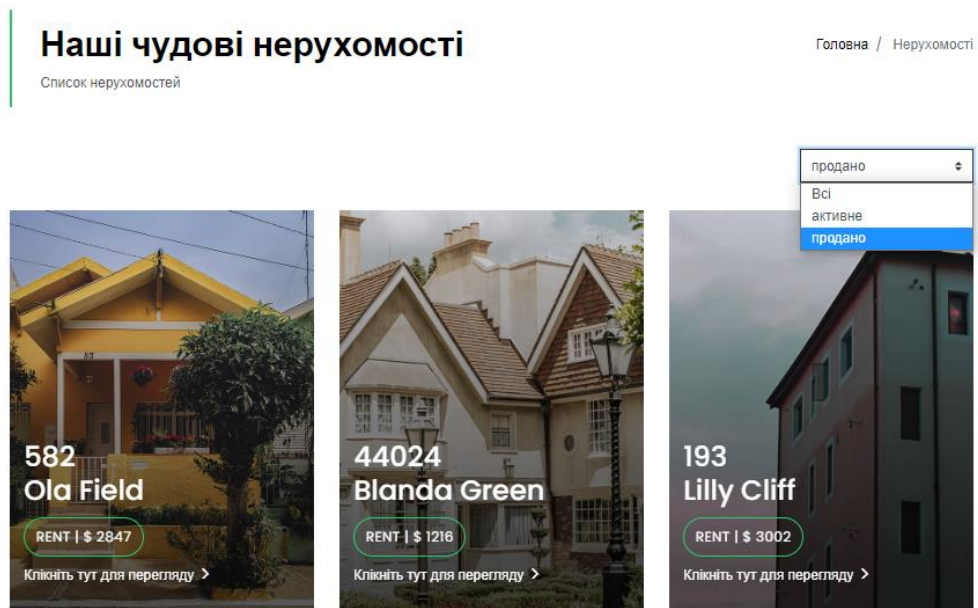


Рисунок 2.17 – Фільтр актуальності нерухомості

На сайті існує пошук нерухомості за фільтрами. Для того щоби він відобразився користувачу необхідно натиснути на значок пошуку у «шапці» вебсайту (рис. 2.18).

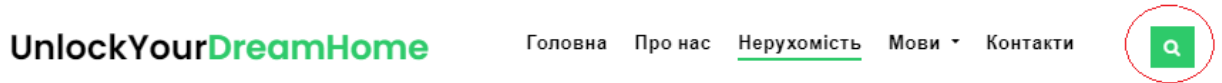


Рисунок 2.18 – Розташування кнопки пошуку нерухомості

Після натискання на цю кнопку клієнту відкривається пошук нерухомості. Тут користувач сайту може знайти нерухомість за шістьма фільтрами. Будь то: тип нерухомості (вілла, будинок, апартаменти); місто; кількість ліжок; кількість гаражних місць; кількість ванних кімнат; мінімальна ціна (рис. 2.19).

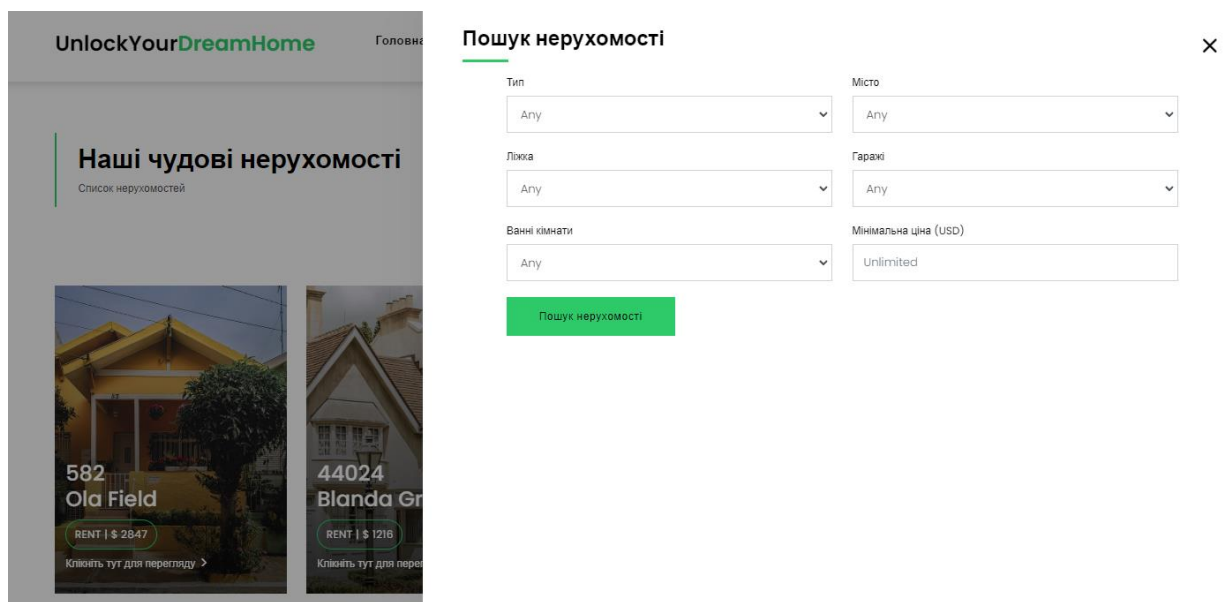



Рисунок 2.19 – Вкладка пошуку нерухомості

Остання вкладка – це «Контакти». Тут клієнт може дізнатись де знаходиться агенство нерухомості. Нижче мапи з розташуванням компанії користувач може звернутися до агенства вписав свої контактні данні та питання яке виникло у клієнта (рис. 2.20).

Зв'яжіться з нами

Контактна інформація

Головна / [Контакти](#)



Subject

Message

Відправити повідомлення

Скажіть привіт

Електронна пошта Unlock Your Dream Home@email.com
Телефон 54758928174

Знайдіть нас тут

New York, 11th Ave 10461

Соціальні мережі

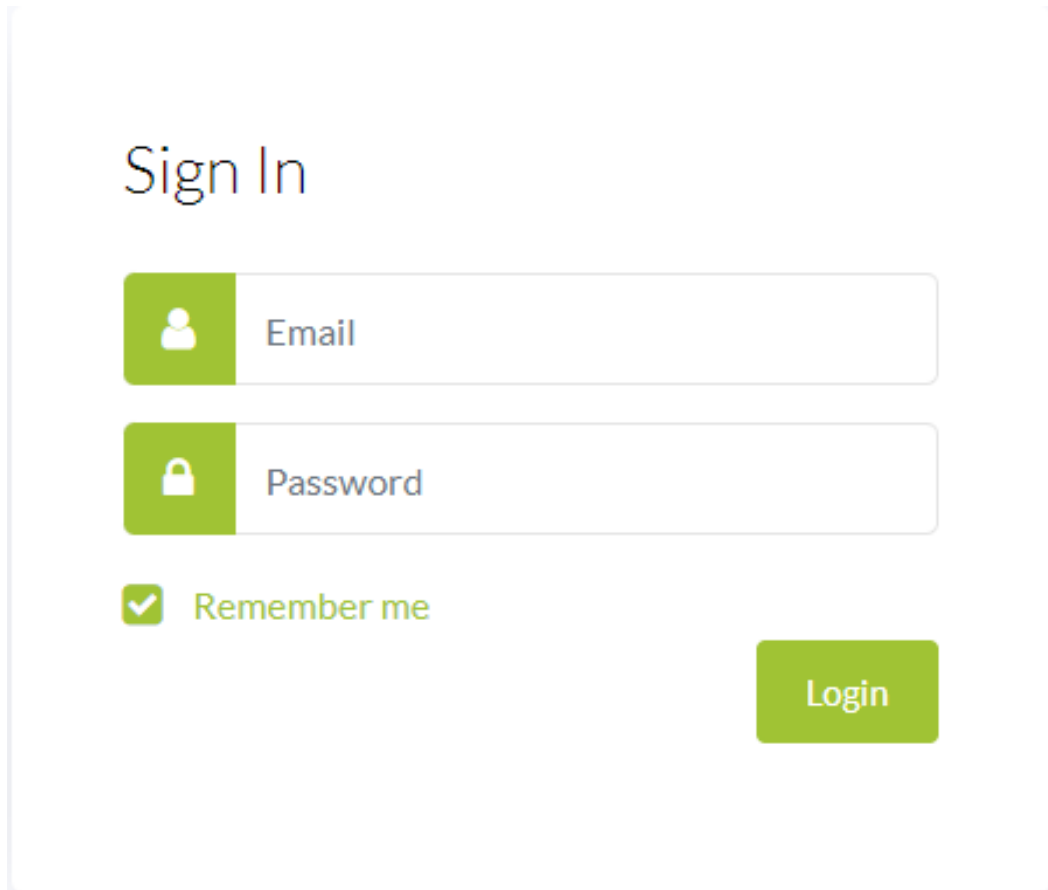
[f](#)
[t](#)
[@](#)
[p](#)
[v](#)

Рисунок 2.20 – Вкладка «Контакти»

Перейдемо до адміністрації сайту. Для цього потрібно увійти до профіля адміністратора. Натиснувши на кнопку «Admin» знизу сайту відкриється сторінка «Sign In» (рис. 2.21).

Уся адміністрація вебсайту виконується на одній сторінці.

Для редагування нерухомості треба перейти у вкладку «Properties» з меню адміністратора (рис. 2.22).



Sign In

Email

Password

Remember me

Login

Рисунок 2.21 – Сторінка входу до адміністрації сайту

ADMIN

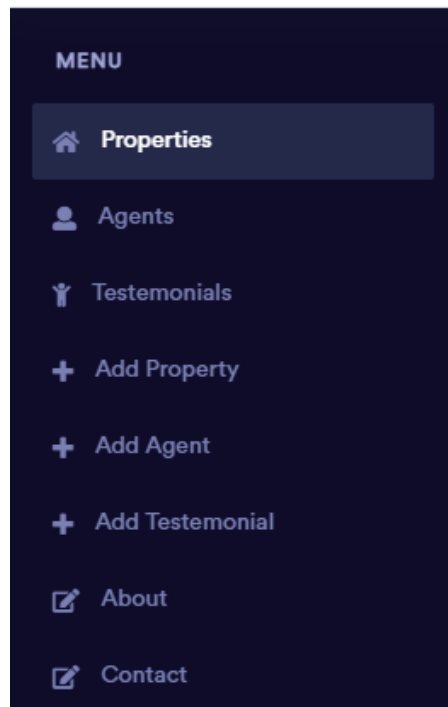


Рисунок 2.22 – Меню адміністратора

Після цього відкривається вікно де можна редагувати нерухомість: місто, статус, тип, ціну купівлі, ціну оренди (рис. 2.23).

ADMIN

MENU

- Properties
- Agents
- Testimonials
- Add Property
- Add Agent
- Add Testimonial
- About
- Contact

Table of properties

City	Status	Type	Price	Rent	Check
New Santino	active	Villa	\$252000	\$2993	Edit ↗
East Murrayborough	active	Apartment	\$164000	\$1433	Edit ↗
West Brenna	active	Villa	\$251000	\$1368	Edit ↗
New Loganside	sold	Villa	\$196000	\$2847	Edit ↗
South Rhettland	active	Villa	\$251000	\$3356	Edit ↗
West Ezra	sold	House	\$197000	\$1216	Edit ↗
Hesselburgh	active	Villa	\$249000	\$1251	Edit ↗
Pourosport	sold	Villa	\$178000	\$3002	Edit ↗
North Keyonshire	active	House	\$227000	\$2614	Edit ↗
South Jarenberg	sold	Villa	\$177000	\$2969	Edit ↗

Рисунок 2.23 – Вкладка редагування нерухомості

Наступна вкладка – це редагування агентів компанії та їх контактних даних (рис. 2.24).

ADMIN

MENU

- Properties
- Agents
- Testimonials
- Add Property
- Add Agent
- Add Testimonial
- About
- Contact

Table of agents

Full name	Phone	Email	
Jack River	097 5669 124	jack.river@email.com	Edit ↗
Alan Wake	092 5454 122	alan.wake@email.com	Edit ↗
Jun Gektor	098 4593 789	jun.gektor@email.com	Edit ↗
Lena Dranik	093 7214 425	lena.dranik@email.com	Edit ↗
Marik Mariv	094 9893 851	marikmariv@email.com	Edit ↗
Tony Jok	096 1289 445	tony.jok@email.com	Edit ↗

Рисунок 2.24 – Вкладка редагування агентів

Третя вкладка – це редагування відгуків клієнтів (рис. 2.25).

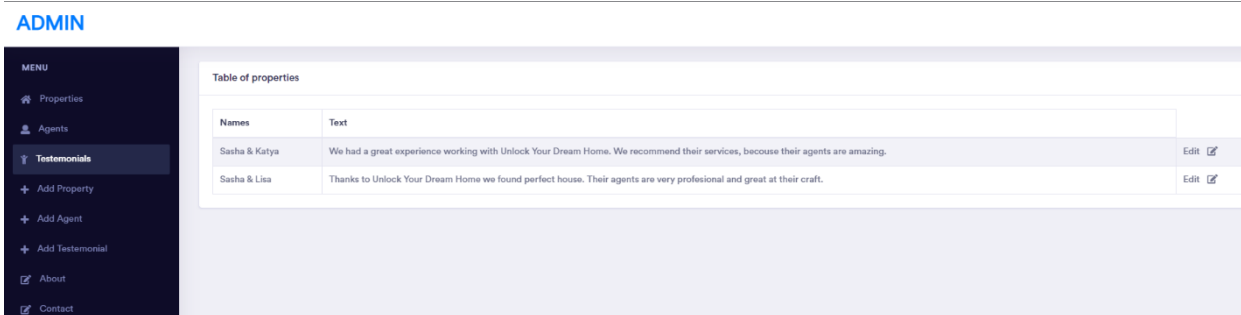


Рисунок 2.25 – Вкладка редагування відгуків

У наступній вкладці адміністратор може додати нерухомість у список та вказати її властивості (рис. 2.26).

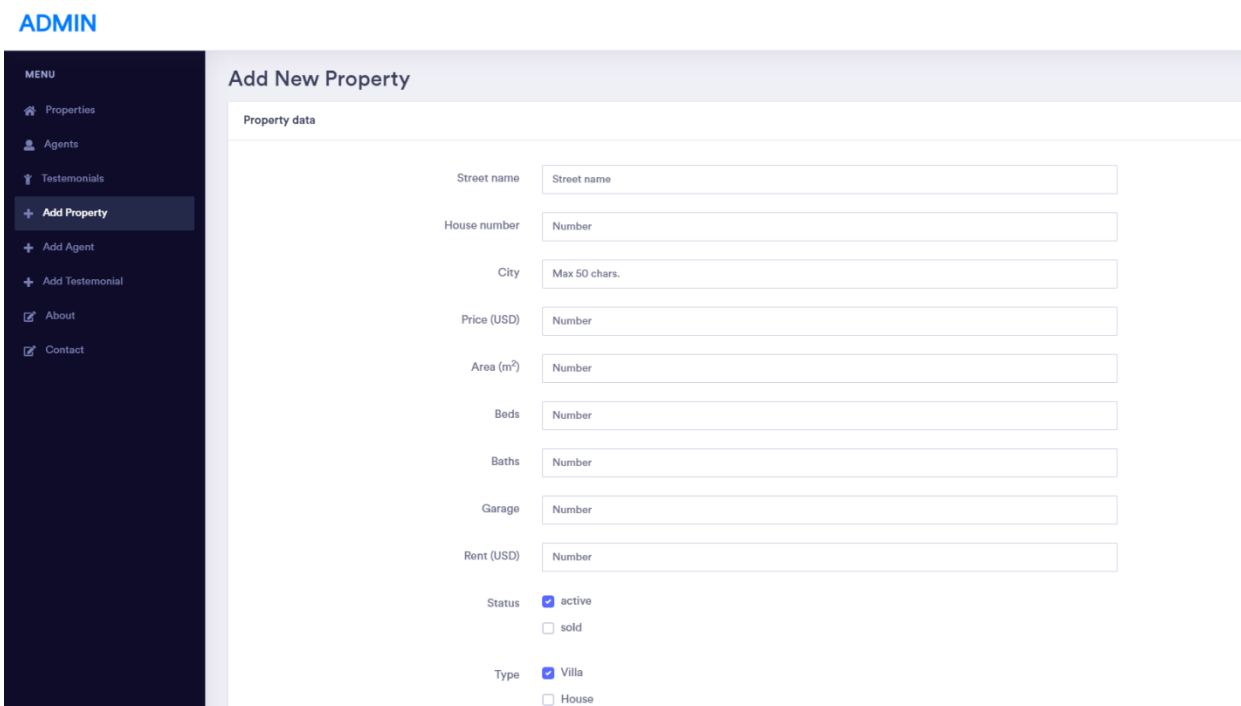


Рисунок 2.26 – Вкладка додавання нерухомості

Також існують вкладки додавання агентів (рис. 2.27) та відгуків клієнтів (рис. 2.28).

ADMIN

ADMIN

MENU

- Properties
- Agents
- Testimonials
- Add Property
- Add Agent**
- Add Testimonial
- About
- Contact

New Agent

Agent data

Full name

Phone number

Email

About (EN)

About (UKR)

Properties

- New Santino, Cathrine Land, 954
- East Murrayborough, Vinnie Pass, 5915
- West Brenna, Braun Inlet, 3105
- New Loganside, Ola Field, 582
- South Rhettland, Leta Plain, 5775
- West Ezra, Blanda Green, 44024
- Hesselburgh, Turcotte Meadows, 57620
- Pourosport, Lilly Cliff, 193
- North Keyonshire, Liyan Tunnel, 2782
- South Jarenberg, Brekke Spur, 330

Рисунок 2.27 – Вкладка додавання агентів

ADMIN

ADMIN

MENU

- Properties
- Agents
- Testimonials
- Add Property
- Add Agent
- Add Testimonial**
- About
- Contact

New Testimonial

Testimonial data

Name(s)

Text (EN)

Text (UKR)

Upload images

Select a image (image 650 x 450):

Select small image (image 80 x 80):

Рисунок 2.28 – Вкладка додавання відгуків клієнтів

Останні дві вкладки – це вкладки редагування сторінок «Про нас» (рис. 2.29) та «Контакти» (рис. 2.30).

ADMIN

ADMIN

MENU

- Properties
- Agents
- Testimonials
- + Add Property
- + Add Agent
- + Add Testimonial
- About**
- Contact

About

About data

Title (EN)	<input type="text" value="We Do Great Design For Creative Folks"/>
Title (UKR)	<input type="text" value="Ми створюємо чудовий дизайн для креативних людей"/>
Subtitle (EN)	<input type="text" value="Find your dream home. View our exclusive listings."/>
Subtitle (UKR)	<input type="text" value="Знайдіть свій мрійний будинок. Перегляньте наші ексклюзивні пропозиції."/>
Text (EN)	<input type="text" value="We offer the highest level of expertise, service, and integrity. Unlock Your Dream Home is the premier Unlock Your Dream Home agency in Manhattan and has helped hundreds of buyers find their dream home in NYC, resulting in almost \$1 Billion of closed residential Unlock Your Dream Home transactions in the last 10 years. We offer services related to luxury Unlock Your Dream Home: sales transactions, rental, and property management. With an international clientele and a worldwide reputation."/>
Text (UKR)	<input type="text" value="Ми пропонуємо найвищий рівень експертизи, обслуговування та чесності. Unlock Your Dream Home - це провідне агентство нерухомості в Мангеттені, яке допомогло сотням покупців знайти свій мрійний будинок в Нью-Йорку, що призвело до закриття майже 1 мільярда доларів угод з нерухомістю в останні 10 років. Ми пропонуємо послуги, пов'язані з розкішною нерухомістю: угоди купівлі-продажу, оренда та управління власністю. З міжнародним клієнтелем та всесвітньою репутацією."/>

Рисунок 2.29 – Вкладка редагування сторінки «Про нас»

ADMIN

ADMIN

MENU

- Properties
- Agents
- Testimonials
- + Add Property
- + Add Agent
- + Add Testimonial
- About
- Contact**

Contact

About data

Email	<input type="text" value="Unlock Your Dream Home@email.com"/>
Phone	<input type="text" value="54758928174"/>
City	<input type="text" value="New York"/>
Street name	<input type="text" value="fth Ave"/>
House number	<input type="text" value="10461"/>

Рисунок 2.30 – Вкладка редагування сторінки «Контакти»

3 КОМП'ЮТЕРНА МОДЕЛЬ ВЕБЗАСТОСУНКУ ДЛЯ РІЄЛТОРСЬКОЇ КОМПАНІЇ

3.1 Обґрунтування вибору середовища програмної реалізації

У рамках кваліфікаційної роботи був розроблений вебзастосунок для рієлторської компанії. Для реалізації було обране середовище VS Code.

Visual Studio Code (VS Code) є безкоштовним та відкритим редактором коду, розробленим компанією Microsoft. Він призначений для розробки програмного забезпечення на різних мовах програмування та підтримується на платформах Windows, macOS та Linux [8].

Одна з основних особливостей Virtual Studio Code – це інтеграція з іншими сервісами. VS Code має плагіні та розширення для інтеграції з різними сервісами, такими як Docker, Azure, Kubernetes та багатьма іншими, що полегшує розробку та розгортання програмного забезпечення.

Visual Studio Code є потужним і гнучким редактором коду, який надає засоби для зручної та продуктивної розробки програмного забезпечення на різних платформах та мовах програмування [9].

Visual Studio Code є популярним редактором коду, який має свої переваги та недоліки. Ось деякі з них:

Переваги Visual Studio Code:

- кросплатформенність: VS Code підтримує роботу на різних операційних системах, включаючи Windows, macOS та Linux, що дозволяє розробникам працювати на своїх улюблених платформах;
- розширюваність: VS Code має потужну систему розширень, яка дозволяє встановлювати додаткові плагіні та розширення для покращення функціональності редактора та відповідності потребам розробника;

- легкість використання: Інтерфейс VS Code є інтуїтивно зрозумілим і простим у використанні, що забезпечує швидку навігацію та зручність у роботі з кодом;
- підсвічування синтаксису та автодоповнення: VS Code надає розширене підсвічування синтаксису для багатьох мов програмування, що полегшує сприйняття коду. Він також пропонує автодоповнення коду, що допомагає швидше та точніше писати програми;
- інтеграція з Git: VS Code має вбудовану підтримку системи контролю версій Git, що дозволяє зручно працювати з репозиторіями, виконувати коміти, розділяти гілки та виконувати інші операції Git безпосередньо з редактора.

Недоліки Visual Studio Code:

- важкість для великих проєктів: У порівнянні з іншими інтегрованими середовищами розробки, VS Code може стикатися з певною тяжкістю при роботі з великими та складними проєктами, оскільки його можливості можуть бути обмежені порівняно з розширеними IDE;
- запуск VS Code може вимагати певних ресурсів системи, особливо при використанні багатьох встановлених розширень або при роботі з великим обсягом коду;
- у деяких випадках, підтримка певних мов програмування або інших технологій може бути менш розвиненою порівняно зі спеціалізованими IDE, що призводить до меншого набору функціональних можливостей;
- недостатня інтеграція з деякими платформами: Хоча VS Code є кроссплатформним, у деяких випадках можуть виникати проблеми з інтеграцією на певних платформах чи середовищах розробки [10–21].

Враховуючи ці переваги та недоліки, Visual Studio Code залишається одним із популярних виборів серед редакторів коду для багатьох розробників, особливо для роботи з невеликими та середніми проєктами.

При розробці вебзастосунка було використано: сервіс Docker, фреймворк Laravel, бібліотека JQuery та система MySQL.

Docker – це відкрите програмне забезпечення, яке дає змогу впаковувати, доставляти та запускати програми у контейнерах. Контейнер є легковажною та автономною середою, яка містить усі необхідні компоненти для роботи програми, включаючи код, залежності, бібліотеки та конфігураційні файли [1].

Основні поняття та компоненти Docker:

- контейнер: Контейнер є стандартизованим пакетом програмного забезпечення, який включає усі необхідні компоненти для роботи програми, включаючи операційну систему, бібліотеки та середовище виконання. Він ізолює програму від інших контейнерів та хост-системи, що забезпечує незалежність та портативність;

- Docker-образ: Образ є шаблоном для створення контейнерів. Він містить вихідний код, залежності та конфігурацію програми. Docker-образи можуть бути створені з використанням Dockerfile, який описує кроки для створення образу;

- Docker-двигун: Docker-двигун є основним компонентом Docker, який виконує контейнеризацію програм та керує їх життєвим циклом. Він дозволяє створювати, запускати та керувати контейнерами на хост-системі;

- Docker Compose: Docker Compose є інструментом для оркестрації багатоконтейнерних застосунків. Він дозволяє описати та керувати багатьма контейнерами, що взаємодіють між собою, за допомогою YAML-файлу конфігурації [21–30].

Загалом, Docker дозволяє розробникам та командам зручно пакувати, доставляти та запускати програми у контейнерах, що спрощує розробку, тестування та розгортання програмного забезпечення.

Laravel – це високорівневий вебфреймворк, розроблений на мові програмування PHP. Він надає розробникам потужні інструменти та зручну

архітектуру для швидкої та ефективної розробки вебзастосунків. Laravel покладається на сучасні практики програмування та пропонує широкий спектр функціональності для спрощення розробки та підтримки застосунків [21].

Laravel є одним із найпопулярніших фреймворків для розробки вебзастосунків на мові програмування PHP. Він позитивно оцінюється за свою простоту використання, широкий функціонал та активну спільноту розробників.

jQuery є популярною бібліотекою JavaScript, яка спрощує маніпулювання HTML-документами, обробку подій, взаємодію із сервером та анімації на вебсторінках. Вона працює як обертка навколо стандартних функцій JavaScript, що робить код більш зрозумілим та спрощує взаємодію з DOM-елементами.

jQuery здобула популярність серед розробників завдяки своїй простоті використання, широкій функціональності та гарній документації. Вона допомагає зекономити годину та упростити процес розробки вебзастосунків, зокрема, коли потрібно швидко взаємодіяти з DOM-елементами, виконувати AJAX-запити або створювати анімації [25].

MySQL є однією з найпопулярніших систем управління базами даних (СУБД). Вона є відкритою та безкоштовною та надає розробникам потужні інструменти для зберігання, управління та доступу до даних. MySQL використовують у широкому спектрі застосунків, від невеликих вебсайтів до великих корпоративних систем. Вона є надійним, швидким та ефективним рішенням для зберігання та управління даними.

3.2 Програмна реалізація

Лістинг 3.1 Створення докера:

```
docker build -t my-mysql-image;
```

```
docker run -p 3306:3306 --name my-mysql-container -d my-mysql-image.
```

Лістинг 3.2 Докер файл з установкою MySQL:

```
FROM mysql:8.0
ENV MYSQL_DATABASE laravel
ENV MYSQL_USER user
ENV MYSQL_PASSWORD root
ENV MYSQL_ROOT_PASSWORD root
ENV DB_CONNECTION mysql
ENV DB_HOST 127.0.0.1
ENV DB_PORT 3306
EXPOSE 3306
```

Це файл створення контейнера Docker на основі образу MySQL версії 8.0.

Далі, у файлі визначено змінні середовища для налаштування бази даних MySQL усередині контейнера:

- `MYSQL_DATABASE`: назва бази даних, яку потрібно створити всередині контейнера, у цьому випадку – «`laravel`»;
- `MYSQL_USER`: ім'я користувача для доступу до бази даних MySQL всередині контейнера;
- `MYSQL_PASSWORD`: пароль для користувача, вказаного в `MYSQL_USER`;
- `MYSQL_ROOT_PASSWORD`: пароль для root-користувача бази даних MySQL усередині контейнера;
- `DB_CONNECTION`: назва з'єднання бази даних для програми Laravel;
- `DB_HOST`: адреса хоста бази даних MySQL, яка буде використовуватися у застосунку Laravel, в даному випадку – «`127.0.0.1`»;
- `DB_PORT`: порт, на якому буде запущено базу даних MySQL усередині контейнера, в даному випадку – `3306`.

Також визначено, що контейнер слухатиме порт `3306`, на якому працює база даних MySQL.

Лістинг 3.3 Установка КОМПОЗЕР КОМПОНЕНТІВ:

```

{
  "name": "laravel/laravel",
  "type": "project",
  "description": "The Laravel Framework.",
  "keywords": ["framework", "laravel"],
  "license": "MIT",
  "require": {
    "php": "^7.3|^8.0",
    "astrotomic/laravel-translatable": "^11.9",
    "fruitcake/laravel-cors": "^2.0",
    "guzzlehttp/guzzle": "^7.0.1",
    "laravel/fortify": "^1.8",
    "laravel/framework": "^8.54",
    "laravel/sanctum": "^2.11",
    "laravel/tinker": "^2.5"
  },
  "require-dev": {
    "facade/ignition": "^2.5",
    "fakerphp/faker": "^1.9.1",
    "laravel/sail": "^1.0.1",
    "mockery/mockery": "^1.4.2",
    "nunomaduro/collision": "^5.0",
    "phpunit/phpunit": "^9.3.3"
  },
  "autoload": {
    "psr-4": {
      "App\\": "app/",
      "Database\\Factories\\": "database/factories/",
      "Database\\Seeders\\": "database/seeders/"
    }
  },
  "autoload-dev": {
    "psr-4": {
      "Tests\\": "tests/"
    }
  },
  "scripts": {
    "post-autoload-dump": [
      "Illuminate\\Foundation\\ComposerScripts::postAutoloadDump",
      "@php artisan package:discover --ansi"
    ],
    "post-update-cmd": [
      "@php artisan vendor:publish --tag=laravel-assets --ansi"
    ],
    "post-root-package-install": [
      "@php -r \"file_exists('.env') || copy('.env.example', '.env');\""
    ]
  }
}

```

```

    ],
    "post-create-project-cmd": [
        "@php artisan key:generate --ansi"
    ]
},
"extra": {
    "laravel": {
        "dont-discover": []
    }
},
"config": {
    "optimize-autoloader": true,
    "preferred-install": "dist",
    "sort-packages": true
},
"minimum-stability": "dev",
"prefer-stable": true
}

```

Цей файл є файлом конфігурації для проєкту Laravel.

Він визначає всі залежності проєкту, у тому числі необхідні пакети PHP та фреймворк Laravel, а також пакети для розробки, такі як PHPUnit, Facade/Ignition та Laravel Sail.

Також визначено автозавантажувачі для класів програми та тестів, а також скрипти, які запускаються після виконання різних команд Composer, наприклад, генерація ключа програми, копіювання файлу `.env.example` тощо.

У конфігурації вказано також різні налаштування для оптимізації роботи програми, такі як оптимізація автозавантажувача, встановлення кращого методу установки пакетів і т.д.

Також визначено додаткові налаштування для фреймворку Laravel, такі як список пакетів, які не потрібно автоматично підключати під час запуску програми.

Застосування міграцій (рис. 3.1).

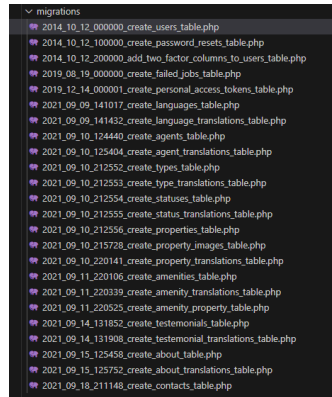


Рисунок 3.1 – Папка з міграціями

Лістинг 3.4 Приклад одного файлу з папки міграцій:

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreatePropertiesTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('properties', function (Blueprint $table) {
            $table->id();
            $table->text("street_name");
            $table->integer("house_number");
            $table->text("city");
            $table->bigInteger("price");
            $table->bigInteger("area");
            $table->integer("beds");
            $table->integer("baths");
            $table->integer("garage");
            $table->bigInteger("rent");
            $table->text("vertical_image");

            $table->unsignedBigInteger('status_id');
            $table->foreign('status_id')->references('id')->on('statuses');
```

```

        $table->unsignedBigInteger('type_id');
        $table->foreign('type_id')->references('id')->on('types');

        $table->unsignedBigInteger('agent_id')->nullable();
        $table->foreign('agent_id')->references('id')->on('agents');

        $table->timestamps();
    });
}

/**
 * Reverse the migrations.
 *
 * @return void
 */
public function down()
{
    Schema::dropIfExists('properties');
}
}

```

Це міграція бази даних для створення таблиці «properties» з колонками, такими як «id», «street_name», «house_number», «city», «price», «area», «beds», «baths», «garage», «rent» та «vertical_image». Також є кілька зовнішніх ключів, які пов'язують таблицю «properties» з іншими таблицями у базі даних. Зокрема, є зовнішні ключі для статусу, типу та агента нерухомості. Ця міграція може бути запущена до Laravel за допомогою команди `php artisan migrate`.

Застосування сидерів (рис. 3.2).

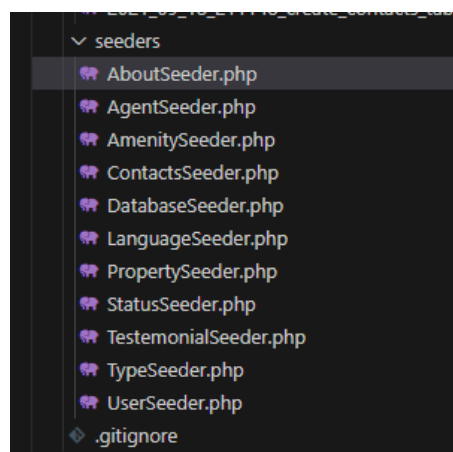


Рисунок 3.2 – Папка сидерів

Лістинг 3.5 Приклад одного файлу з папки сидерів:

```
<?php

namespace Database\Seeders;

use Illuminate\Database\Seeder;
use App\Helpers\GetAboutImage;
use App\Repositories\AboutRepositoryInterface;

class AboutSeeder extends Seeder
{
    private $aboutRepository;

    public function __construct(AboutRepositoryInterface $aboutRepository)
    {
        $this->aboutRepository = $aboutRepository;
    }
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        $about = [
            'horizontal_image' => GetAboutImage::horizontal(),
            'vertical_image' => GetAboutImage::vertical(),
            'en' => [
                'title' => "We Do Great Design For Creative Folks",
                'subtitle' => "Find your dream home. View our exclusive listings.",
                "text" => "We offer the highest level of expertise, service, and
integrity.
                Unlock Your Dream Home is the premier Unlock Your Dream Home agency
in Manhattan and has helped hundreds of buyers find their dream home in NYC,
                resulting in almost $1 Billion of closed residential Unlock Your
Dream Home transactions in the last 10 years.
                We offer services related to luxury Unlock Your Dream Home: sales
transactions,
                rental, and property management. With an international clientele
and a worldwide reputation."
            ],
            'ukr' => [
                'title' => "Ми створюємо чудовий дизайн для креативних людей",
                'subtitle' => "Знайдіть свій мрійний будинок. Перегляньте наші
ексклюзивні пропозиції.",
                "text" => "Ми пропонуємо найвищий рівень експертизи, обслуговування
та чесності.
```

```

        Unlock Your Dream Home - це провідне агентство нерухомості в
Мангеттені, яке допомогло сотням покупців знайти свій мрійний будинок в Нью-
Йорку,
        що призвело до закриття майже 1 мільярда доларів угод з нерухомістю в
останні 10 років.
        Ми пропонуємо послуги, пов'язані з розкішною нерухомістю: угоди
купівлі-продажу, оренда та управління власністю.
        З міжнародним клієнтелем та всесвітньою репутацією."
    ],
    ];

    $this->aboutRepository->create($about);
}
}

```

Цей PHP код описує клас-сівалку (Seeder) для заповнення бази даних початковими даними (seed data). Зокрема, цей сівалка заповнює таблицю «About» (відомості про компанію) початковими даними. У методі «run» задані дані про компанію англійською та українською мовами, включаючи назву, підзаголовки, текстовий опис та зображення для різних орієнтацій екрану. Сівалка використовує репозиторій «AboutRepositoryInterface», який має бути попередньо визначений та реалізований для роботи з моделлю «About» у базі даних. У методі «create» цього репозиторію дані зберігаються до бази даних. У проєкті використано методологію MVC.

Застосування контролерів (рис. 3.3).

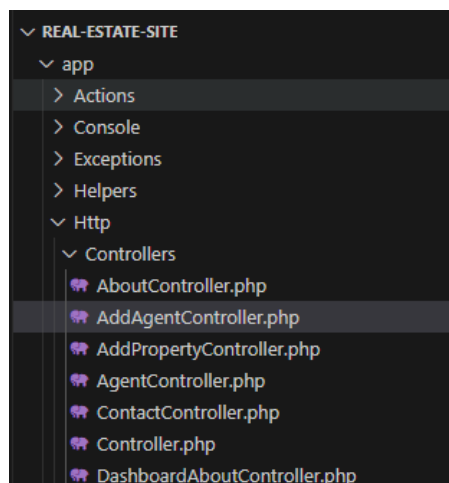


Рисунок 3.3 – Папка контролерів

Лістинг 3.6 Контролер AddAgentController:

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Repositories\LanguageRepositoryInterface;
use App\Repositories\PropertyRepositoryInterface;
use App\Repositories\AgentRepositoryInterface;
use App\Http\Requests\AddNewAgentRequest;
use Illuminate\Support\Facades\Storage;

class AddAgentController extends Controller
{
    private $languageRepository;
    private $propertyRepository;
    private $agentRepository;

    public function __construct(
        LanguageRepositoryInterface $languageRepository,
        PropertyRepositoryInterface $propertyRepository,
        AgentRepositoryInterface $agentRepository
    ) {
        $this->languageRepository = $languageRepository;
        $this->propertyRepository = $propertyRepository;
        $this->agentRepository = $agentRepository;
    }

    public function addAgentView()
    {
        $languages = $this->languageRepository->all();
        $properties = $this->propertyRepository->all();

        return view('adminpanel.add-agent', [
            "languages" => $languages,
            "properties" => $properties
        ]);
    }

    public function addAgent(AddNewAgentRequest $request)
    {
        $fileName = time().'_'.$request->image->getClientOriginalName();

        $agentData = [
            "full_name" => $request->full_name,
            "phone" => $request->phone,
```

```

        "email"    => $request->email,
        "image"   => $fileName
    ];

    // save image
    $path = $request->image->storeAs(
        '/agent_images', $fileName, 'images'
    );

    $languages = $this->languageRepository->all();
    // adding description translations
    foreach ($languages as $language) {
        $agentData[$language->iso] = [
            "about" => $request[$language->iso."-description"]
        ];
    };

    // array of property ids
    $chosenPropertyIds = [];
    $allPropertyIds = $this->propertyRepository-
>allIdsInOneDimensionalArray();

    foreach ($allPropertyIds as $propertyId) {
        if ($request["property".$propertyId] != null) {
            array_push($chosenPropertyIds, $propertyId);
        }
    }

    $this->agentRepository->createAgent($agentData, $chosenPropertyIds);

    return redirect('/home/agents')->with('successMessage', 'Saved
successfully');
}
}

```

Цей код містить контролер `AddAgentController`, який відповідає за додавання нових агентів до бази даних. У конструкторі контролера відбувається ініціалізація трьох репозиторіїв (`LanguageRepositoryInterface`, `PropertyRepositoryInterface` та `AgentRepositoryInterface`), які використовуються в методах контролера.

Метод `addAgentView()` відображає уявлення, в якому можна додати нового агента. У поданні виводиться список мов та нерухомості, які використовуються для опису агента.

Метод `addAgent()` використовується для обробки запиту додавання нового агента. При отриманні запиту метод зберігає зображення агента в сховище і додає дані про агента в базу даних. Метод також отримує список всіх мов з `LanguageRepositoryInterface`, а потім додає переклади опису агента для кожної мови. Метод також отримує список всіх наявних нерухомості з `PropertyRepositoryInterface` і вибирає лише ті, які були обрані на формі додавання нового агента. Потім метод викликає `createAgent()` з `AgentRepositoryInterface`, щоб додати дані про агента і список вибраних нерухомості в базі даних. Після успішного збереження агента в базі даних метод перенаправляє користувача на сторінку зі списком агентів. Якщо виникла помилка, метод поверне відповідне повідомлення про помилку.

Папка з описами моделей (рис. 3.4).

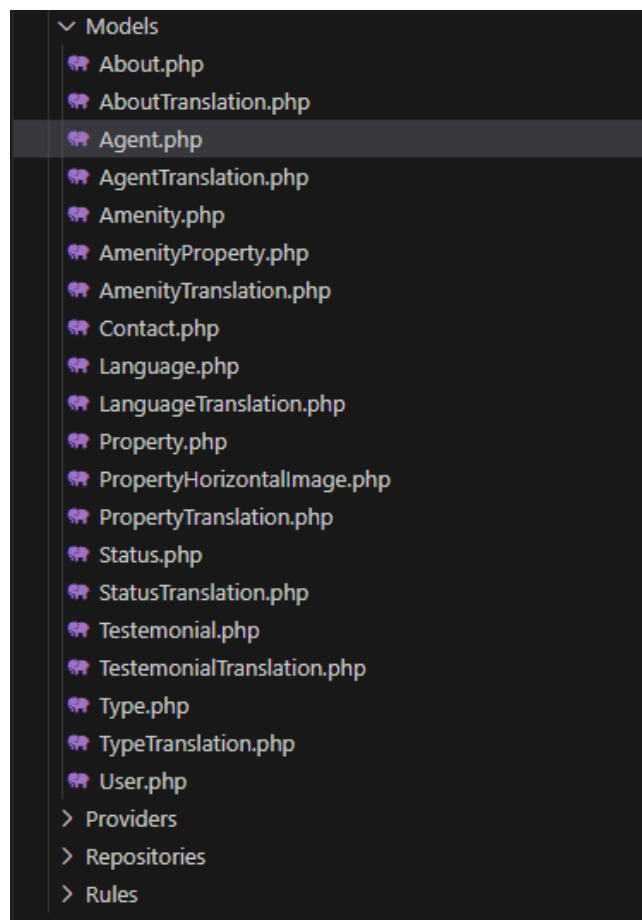


Рисунок 3.4 – Папка моделі

Лістинг 3.7 Опис моделі Agent:

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
use AstroAtomic\Translatable\Contracts\Translatable as TranslatableContract;
use AstroAtomic\Translatable\Translatable;
use App\Models\Property;

class Agent extends Model
{
    use HasFactory;
    use Translatable;

    public $translatedAttributes = ['about'];
    protected $fillable = ['full_name', 'phone', 'email', 'image'];

    public function properties()
    {
        return $this->hasMany(Property::class);
    }

    public function hasThisProperty(Property $property)
    {
        $relatedPropertyIds = array_column($this->properties()->get(['id'])->toArray(), 'id');

        if ( in_array($property->id, $relatedPropertyIds) ) {
            return true;
        }

        return false;
    }

    public function shortAbout()
    {
        if (strlen($this->about) >= 90) {
            return substr($this->about, 0, 90);
        } else {
            return $this->about;
        }
    }
}
```

Цей PHP-код описує модель Agent (Агент) програми, яка використовується для роботи з даними агентів нерухомості.

У цій моделі використовується пакет `Astrotomic\Translatable` для підтримки перекладу атрибута `about` (Опис) кількома мовами. Також визначено зв'язки з моделлю `Property` (Нерухомість): кожен агент може мати кілька пов'язаних об'єктів нерухомості, а функція `hasThisProperty` використовується для перевірки, чи агент має зв'язок з конкретним об'єктом нерухомості.

Функція `shortAbout()` використовується для виведення короткого опису агента, який не перевищує 90 символів.

У моделі визначено заповнювані атрибути `full_name` (Повне ім'я), `phone` (Телефон), `email` (Email), `image` (Зображення).

Папка з шаблонами (рис. 3.5).

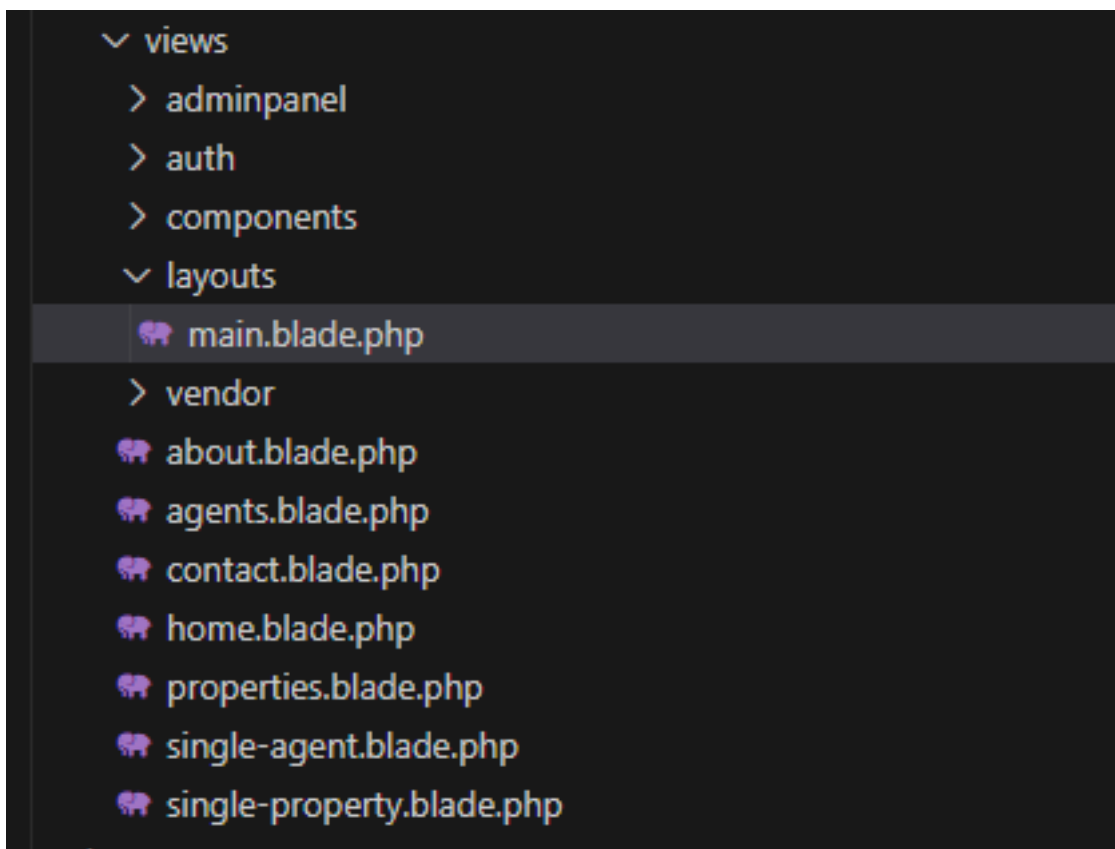


Рисунок 3.5 – Папка «views»

Лістинг 3.8 Шаблон HTML-сторінки:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>Unlock Your Dream Home Agency</title>
  <meta content="width=device-width, initial-scale=1.0" name="viewport">

  <!-- Google Fonts -->
  <link
href="https://fonts.googleapis.com/css?family=Poppins:300,400,500,600,700"
rel="stylesheet">

  <!-- Bootstrap CSS File -->
  <link href="/lib/bootstrap/css/bootstrap.min.css" rel="stylesheet">

  <!-- Libraries CSS Files -->
  <link href="/lib/font-awesome/css/font-awesome.min.css" rel="stylesheet">
  <link href="/lib/animate/animate.min.css" rel="stylesheet">
  <link href="/lib/ionicons/css/ionicons.min.css" rel="stylesheet">
  <link href="/lib/owlcarousel/assets/owl.carousel.min.css" rel="stylesheet">

  <!-- Main Stylesheet File -->
  <link href="/css/style.css" rel="stylesheet">
</head>
<body>

  @include('components.navigation')
  @yield('content')
  @include('components.footer')

</body>
</html>
```

Це шаблон HTML-сторінки, який використовується у вебзастосунку. Він містить посилання на зовнішні бібліотеки стилів (Bootstrap, FontAwesome, Animate, Ionicons, OwlCarousel) та основний файл стилів програми. Також у шаблон включаються компоненти навігації та підвалу, а контентна частина


```

    </p>
    <div class="info-agents color-a">
      <p>
        <strong>{{ __( 'Phone' ) }}: </strong> {{ $agent->phone }}</p>
      <p>
        <strong>{{ __( 'Email' ) }}: </strong> {{ $agent->email }}</p>
      </div>
    </div>
    <div class="card-footer-d">
      <div class="socials-footer d-flex justify-content-center">
        <ul class="list-inline">
          <li class="list-inline-item">
            <a href="#" class="link-one">
              <i class="fa fa-facebook" aria-hidden="true"></i>
            </a>
          </li>
          <li class="list-inline-item">
            <a href="#" class="link-one">
              <i class="fa fa-twitter" aria-hidden="true"></i>
            </a>
          </li>
          <li class="list-inline-item">
            <a href="#" class="link-one">
              <i class="fa fa-instagram" aria-hidden="true"></i>
            </a>
          </li>
          <li class="list-inline-item">
            <a href="#" class="link-one">
              <i class="fa fa-pinterest-p" aria-hidden="true"></i>
            </a>
          </li>
          <li class="list-inline-item">
            <a href="#" class="link-one">
              <i class="fa fa-dribbble" aria-hidden="true"></i>
            </a>
          </li>
        </ul>
      </div>
    </div>
  </div>
</div>
</div>
@endforeach
</div>
<div class="row">
  {{ $agents->links() }}
</div>
</div>
</section>
<!--/ Agents Grid End /-->

```

Тут відображається секція вебсторінки, яка відображає сітку агентів. У циклі `forEach` перебираються агенти, передані в шаблон, і для кожного агента відображається блок із зображенням, ім'ям агента, його коротким описом, контактними даними та посиланнями на соціальні мережі. Наприкінці секції відображається пагінація для перемикання сторінок.

Лістинг 3.10 Приклад зберігання мов:

```
en.json
```

```
{  
  
  "Home": "Home",  
  
  "About": "About",  
  
  "Property": "Property",  
  
  "Languages": "Languages",  
  
  "Blog": "Blog",  
  
  "Contact": "Contact",  
  
  "Our Services": "Our Services",  
  
  "Tenants": "Tenants",  
  
  "Sell": "Sell",  
  
  "Rentals": "Rentals",  
}
```

```
ukr.json
```

```
{  
  
  "Home": "Головна",  
  
  "About": "Про нас",
```

"Property": "Нерухомість",
"Languages": "Мови",
"Blog": "Блог",
"Contact": "Контакти",
"Our Services": "Наші послуги",
"Tenants": "Орендарі",
"Sell": "Продаж",
"Rentals": "Оренда",

Приклад екшенів (рис. 3.6).

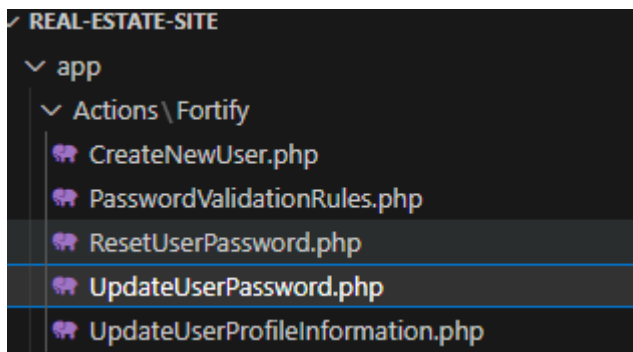


Рисунок 3.6 – Папка екшенів

Лістинг 3.11 Клас CreateNewUser:

```
class CreateNewUser implements CreatesNewUsers
{
    use PasswordValidationRules;

    /**
     * Validate and create a newly registered user.
     *
     * @param array $input
```

```

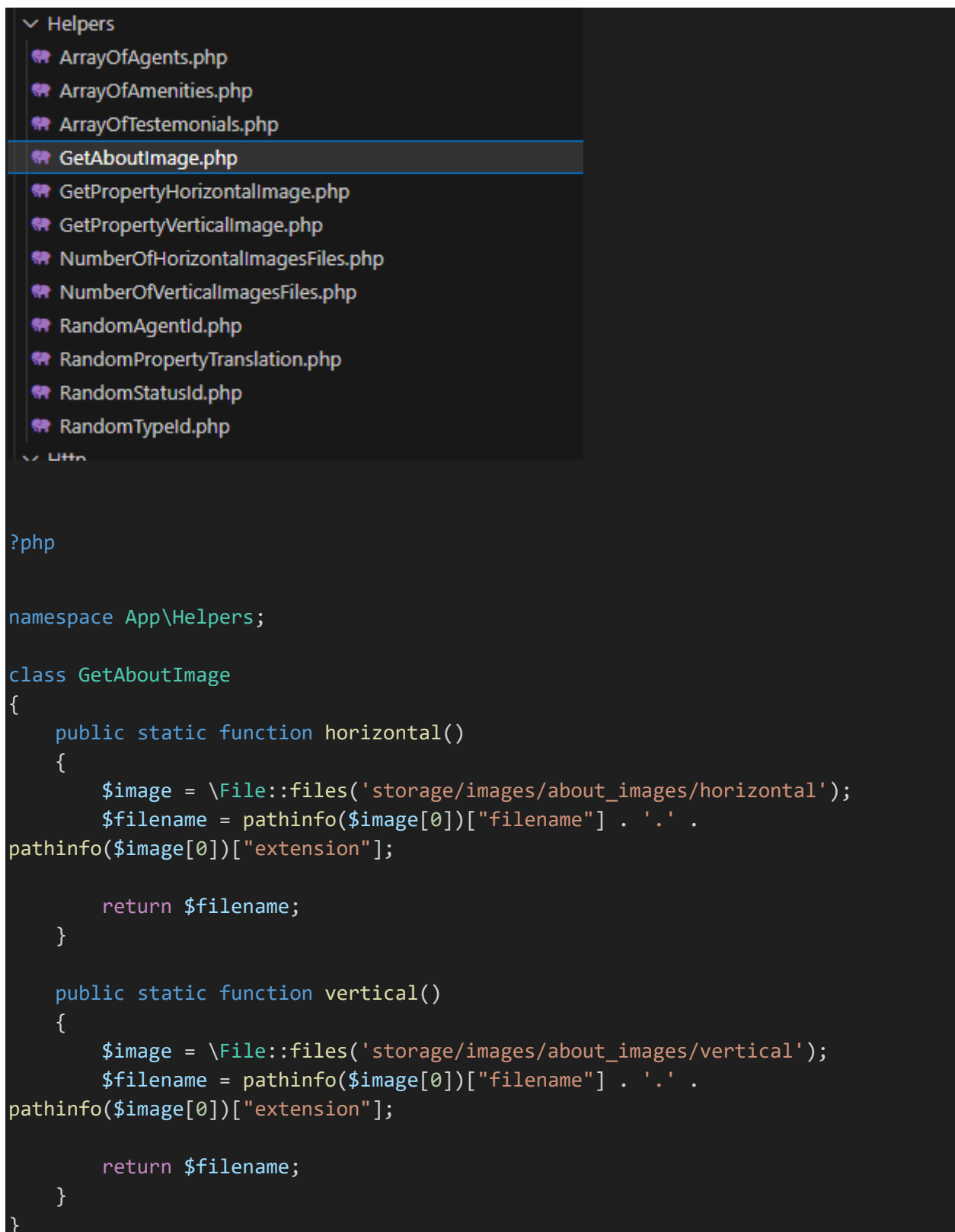
* @return \App\Models\User
*/
public function create(array $input)
{
    Validator::make($input, [
        'name' => ['required', 'string', 'max:255'],
        'email' => [
            'required',
            'string',
            'email',
            'max:255',
            Rule::unique(User::class),
        ],
        'password' => $this->passwordRules(),
    ]->validate());

    return User::create([
        'name' => $input['name'],
        'email' => $input['email'],
        'password' => Hash::make($input['password']),
    ]);
}
}

```

Це код класу CreateNewUser, який реалізує інтерфейс CreatesNewUsers. Він містить метод «create», який приймає масив даних \$input, валідує його, використовуючи правила валідації для поля «password» і створює нового користувача в базі даних, використовуючи модель «User». Якщо дані не проходять валідацію, буде викинуто виняток. Якщо дані проходять валідацію, буде створено новий користувач із зазначеними в масиві \$input даними.

Лістинг 3.12 Екшен отримання зображення:



The image shows a code editor interface. At the top, a file explorer shows a folder named 'Helpers' containing several PHP files. The file 'GetAboutImage.php' is selected and highlighted. Below the file explorer, the PHP code for the 'GetAboutImage' class is displayed. The code defines two static methods: 'horizontal()' and 'vertical()'. Both methods use the '\File::files()' function to retrieve a list of files from a specific directory, then use 'pathinfo()' to extract the filename and extension of the first file in the list. The 'horizontal()' method uses the directory 'storage/images/about_images/horizontal' and the 'vertical()' method uses 'storage/images/about_images/vertical'.

```
?php

namespace App\Helpers;

class GetAboutImage
{
    public static function horizontal()
    {
        $image = \File::files('storage/images/about_images/horizontal');
        $filename = pathinfo($image[0])["filename"] . '.' .
pathinfo($image[0])["extension"];

        return $filename;
    }

    public static function vertical()
    {
        $image = \File::files('storage/images/about_images/vertical');
        $filename = pathinfo($image[0])["filename"] . '.' .
pathinfo($image[0])["extension"];

        return $filename;
    }
}
```

Папка «Middleware» з класом «RedirectIfAuthenticated» (рис. 3.7).

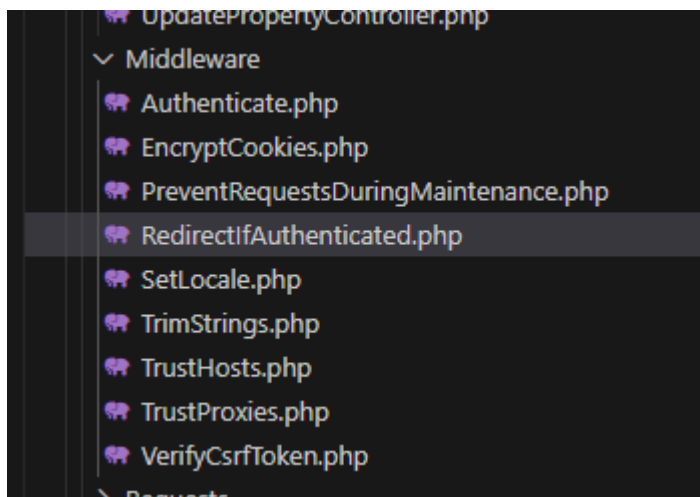


Рисунок 3.7 – Папка «Middleware»

Лістинг 3.13 Клас RedirectIfAuthenticated:

```
class RedirectIfAuthenticated
{
    /**
     * Handle an incoming request.
     *
     * @param \Illuminate\Http\Request $request
     * @param \Closure $next
     * @param string|null ...$guards
     * @return mixed
     */
    public function handle(Request $request, Closure $next, ...$guards)
    {
        $guards = empty($guards) ? [null] : $guards;

        foreach ($guards as $guard) {
            if (Auth::guard($guard)->check()) {
                return redirect(RouteServiceProvider::HOME);
            }
        }

        return $next($request);
    }
}
```

Папка з фронтенд частиною (рис. 3.8).

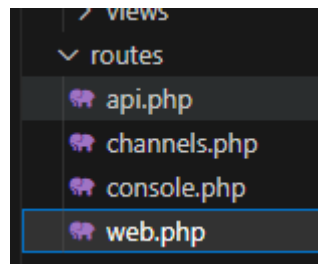


Рисунок 3.8 – Папка «routes»

Лістинг 3.14 Фронтенд частина:

```

|-----|
| Web Routes |
|-----|
| Here is where you can register web routes for your application. These |
| routes are loaded by the RouteServiceProvider within a group which |
| contains the "web" middleware group. Now create something great! |
| |
| */
Route::get('/', function () {
    return redirect(app()->getLocale());
});

Route::group(['prefix' => 'home', 'middleware' => 'verified'], function() {
    Route::get('/', function() {
        return redirect('/home/properties');
    });
    Route::get('/properties', [DashboardPropertyController::class,
'allProperties']);
    Route::get('/property/{id}', [DashboardPropertyController::class,
'singleProperty']);
    Route::put('/property/{id}', [UpdatePropertyController::class,
'updateProperty']);
    Route::delete('/property', [DeletePropertyController::class,
'deleteProperty']);
    Route::put('/property/vertical-image/{id}', [PropertyImageController::class,
'updatePropertyVerticalImage']);

```

```

    Route::delete('/property/horizontal-image', [PropertyImageController::class,
'deletePropertyHorizontalImage']);
    Route::post('/property/horizontal-image', [PropertyImageController::class,
'addPropertyHorizontalImage']);
    Route::get('/add-property', [AddPropertyController::class,
'addPropertyView']);
    Route::post('/add-property', [AddPropertyController::class, 'addProperty']);
    Route::get('/agents', [DashboardAgentController::class, 'allAgents']);
    Route::get('/agent/{id}', [DashboardAgentController::class, 'singleAgent']);
    Route::put('/agent/image', [UpdateAgentController::class, 'updateImage']);
    Route::put('/agent/{id}', [UpdateAgentController::class, 'updateAgent']);
    Route::delete('/agent', [DeleteAgentController::class, 'deleteAgent']);
    Route::get('/add-agent', [AddAgentController::class, 'addAgentView']);
    Route::post('/agent', [AddAgentController::class, 'addAgent']);
    Route::get('/testimonials', [TestemotionalController::class,
'allTestimonials']);
    Route::put('/testimonial/big-image', [TestemotionalController::class,
'updateBigImage']);
    Route::put('/testimonial/mini-image', [TestemotionalController::class,
'updateMiniImage']);
    Route::get('/testimonial/{id}', [TestemotionalController::class,
'singleTestemotional']);
    Route::put('/testimonial/{id}', [TestemotionalController::class,
'updateTestemotional']);
    Route::get('/add-testemotional', [TestemotionalController::class,
'addTestemotionalView']);
    Route::post('/testimonial', [TestemotionalController::class,
'addTestemotional']);
    Route::delete('/testimonial', [TestemotionalController::class,
'deleteTestemotional']);
    Route::get('/about', [DashboardAboutController::class, 'aboutView']);
    Route::post('/about', [DashboardAboutController::class,
'updateOrCreateAbout']);
    Route::get('/contact', [DashboardContactController::class, 'contactView']);
    Route::post('/contact', [DashboardContactController::class,
'updateOrCreateContact']);
});

Route::group(['prefix' => '{locale}', 'middleware' => 'setlocale'], function() {
    Route::get('/', [HomeController::class, 'homePage']);
    Route::get('/about', [AboutController::class, 'aboutPage']);
    Route::get('/properties', [PropertyController::class, 'allProperties']);
    Route::get('/property/{id}', [PropertyController::class, 'singleProperty']);
    Route::get('/agents', [AgentController::class, 'allAgents']);
    Route::get('/agent/{id}', [AgentController::class, 'singleAgent']);
    Route::get('/contact', [ContactController::class, 'contact']);
    Route::get('/search', [SearchController::class, 'search']);
});

```

ВИСНОВКИ

У рамках кваліфікаційної роботи був розроблений і реалізований вебзастосунок для рієлторської компанії за допомогою таких мов програмування: JavaScript, PHP, CSS, HTML.

В результаті чого було отримано вебсайт з потрібним функціоналом для задоволення потреб користувачів. Вебзастосунок відповідає сучасним стандартам якості вигляду клієнтського інтерфейсу. Також було проведено ретельне тестування вебзастосунку.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Руденко Д.О., Бондар В.О. (2020). Огляд можливостей використання стратегій об'єктно-орієнтованого маппінгу для зіставлення сутностей при розробці web додатків. «Priority Directions of Science and Technology Development» Київ, Україна.с.377-380.
2. Docker, I. (2020). Docker. *linea*.[Junio de 2017]. Disponible en: <https://www.docker.com/what-docker>.
3. Bilik, Ö., Kankaya, E. A., & Deveci, Z. (2020). Effects of web-based concept mapping education on students' concept mapping and critical thinking skills: A double blind, randomized, controlled study. *Nurse education today*, 86, 104312.
4. Miell, I., & Sayers, A. (2019). *Docker in practice*. Simon and Schuster.
5. Gyani, J., Ahmed, A., & Haq, M. A. (2022). MCDM and various prioritization methods in AHP for CSS: A comprehensive review. *IEEE Access*.
6. McElroy, E., Kearney, M., Touhey, J., Evans, J., Cooke, Y., & Shevlin, M. (2019). The CSS-12: Development and validation of a short-form version of the cyberchondria severity scale. *Cyberpsychology, Behavior, and Social Networking*, 22(5), 330-335.
7. Frain, B. (2022). *Responsive Web Design with HTML5 and CSS: Build future-proof responsive websites using the latest HTML5 and CSS techniques*. Packt Publishing Ltd.
8. Robbins, J. N. (2012). *Learning web design: A beginner's guide to HTML, CSS, JavaScript, and web graphics*. " O'Reilly Media, Inc."
9. Duckett, J. (2011). *Beginning html, xhtml, css, and javascript*. John Wiley & Sons.
10. Nixon, R. (2014). *Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5*. " O'Reilly Media, Inc."
11. DuBois, P. (2008). *MySQL*. Pearson Education.
12. Christudas, B., & Christudas, B. (2019). *MySQL* (pp. 877-884). Apress.

13. Converse, T., Park, J., & Morgan, C. (2004). *PHP5 and MySQL bible* (Vol. 147). John Wiley & Sons.
14. McFarland, D. S. (2011). *Javascript & jQuery: the missing manual*. "O'Reilly Media, Inc."
15. Lawson, B., & Sharp, R. (2011). *Introducing html5*. New Riders.
16. Cordingly, R., Yu, H., Hoang, V., Perez, D., Foster, D., Sadeghi, Z., ... & Lloyd, W. J. (2020, August). Implications of programming language selection for serverless data processing pipelines. In *2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCCom/CyberSciTech)* (pp. 704-711). IEEE.
17. Руденко Д.О., Колосок Е.В. (2021) Огляд можливостей Google Analytics для роботи з даними.
18. Руденко Д.О. (2017) Метод семантичної інтеграції локально незалежних даних.
19. Руденко Д. О., Кухарчук В. А. (2022) Сучасні напрямки застосування методів супер роздільності зображення. URL: <https://sci-conf.com.ua/viii-mizhnarodna-naukovo-praktichna-konferentsiya-modern-research-in-world-science-29-31-10-2022-lviv-ukrayina-arhiv/> (дата звернення 05.05.2023).
20. V. Lyashenko, D. Rudenko, (2021) Modeling Deformation of Spur Gear. *International Journal of Recent Technology and Applied Science*, vol. 3, no. 2, pp. 81-91, Sep. 2021.
21. Руденко Д.О., Таняньський О.С. (2021) Принципи передобробки даних для машинного навчання. «TOPICAL ISSUES OF MODERN SCIENCE, SOCIETY AND EDUCATION», Харків, 1-3 листопада 2021р. С.381-385.
22. Сучасний підручник з JavaScript. URL: <https://uk.javascript.info/> (дата звернення 05.05.2023).

23. Гороховатський, В. О., & Творошенко, І. С. (2021). Методи інтелектуального аналізу та оброблення даних: навч. посібник.
24. Zeleniy O., Rudenko D., Lyubchenko V., Lyashenko V. (2022) Image Processing as an Analysis Tool in Medical Research. «International Journal of Academic and Applied Research (IJAAR) ». – 2022. – Vol. 6(9). – P. 135-141.
25. V.Baranova, H.Kots, D.Rudenko, V.Lyashenko. (2022) Phase Portrait Models as a Tool for Analyzing Banking Activities. «International Journal of Academic Management Science Research (IJAMSR)» ISSN: 2643-900X . Vol. 6, Issue 12, December - 2022, Pages: 52-58.
26. V. Lyashenko, D. Rudenko, (2021) Modeling Deformation of Spur Gear. « International Journal of Recent Technology and Applied Science», vol. 3, no. 2, pp. 81-91, Sep. 2021.
27. Daradkeh Y.I., Gorokhovatskyi V., Tvoroshenko I., and Zeghid M. (2022) Tools for fast metric data search in structural methods for image classification, *IEEE Access*, 10, pp. 124738-124746.
28. Руденко Д.О., Постніков А.В. (2021) Аналіз загальної класифікації та структури в event -індустрії. «TOPICAL ISSUES OF MODERN SCIENCE, SOCIETY AND EDUCATION», Харків, 1-3 листопада 2021р. С.892-897.
29. Bodyanskiy, Y. V., Shafronenko, A. Y., Rudenko, D. A., & Klymova, I. N. (2020). Online Recurrent Method Of Credibilistic Fuzzy Clusterin..
30. В.О. Гороховатський, Д.О. Руденко, Т.О. Сірик. (2019) Дослідження системи ієрархічних ознак при блочному поданні опису у складі множини ключових точок зображення. URL: <https://openarchive.nure.ua/items/24eeac11-38ed-4f42-9a5b-094302466ee1> (дата звернення 05.05.2023).
31. Гороховатський В., Передрій О., Творошенко І., Марков Т. (2023) Матриця відстаней для множини компонентів структурного опису як інструмент для створення класифікатора зображень, *Сучасні інформаційні системи*, 7(1), С. 5-13.

32. Гороховатський В.О., Творошенко І.С., Чмутов Ю.В. (2022) Застосування систем ортогональних функцій для формування простору ознак у методах класифікації зображень, *Сучасні інформаційні системи*, 6(3), С. 5-12.

33. Gorokhovatskyi V., Tvoroshenko I., Kobylin O., and Vlasenko N. (2023) Search for visual objects by request in the form of a cluster representation for the structural image description, *Advances in Electrical and Electronic Engineering*, 21(1), pp. 19-27.