МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет Магістерської та аспірантської підготовки
Кафедра інформаційних технологій

КОМПЛЕКСНА МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Проектування та розробка автоматизованої системи збору даних про нерухомість міста Одеса»

Склад:

Частина 1. «Проектування та розробка серверної частини системи»

Виконавець: Кушнір Л. О.
(П.І.Б.)

Керівник: Шпінарева І.М.
(П.І.Б.)

Частина 2. «Розробка клієнтської частини системи»

Виконавець: Долготер Д.О.
(П.І.Б.)

Керівник: Шпінарева І.М.
(П.І.Б.)

Староста роботи: Кушнір Л. О.
(П.І.Б.)

Провідний керівник проекту: к.ф.-м.н.,доц., Шпінарева І.М.
(П.І.Б.)

Рецензент: : к.ф.-м.н.,доц., Вітавецька Л.А.
(П.І.Б.)

ОДЕСА 2018

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет Магістерської та
аспірантської підготовки
Кафедра інформаційних технологій

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Проектування та розробка серверної частини системи»

Виконав студент 1 курсу групи
МК- 61 спеціальності 122
Комп'ютерні науки
та інформаційні технології
Кушнір Леонід Олександрович

Керівник к.ф.-м.н., доц.
Шпінарева Ірина Михайлівна

Консультант

Рецензент к.ф.-м.н., доц.
Вітавецька Лариса Анатоліївна

Одеса 2018

АНОТАЦІЯ

Тема магістерської роботи «Проектування та розробка автоматизованої системи збору даних про нерухомість міста Одеса».

Актуальність магістерської роботи полягає в необхідності розробки web-додатку, який здійснює парсинг інформації про нерухомість в єдину комплексну систему централізованого знаходження необхідної інформації для клієнтів.

Об'єкт дослідження – процеси розробки автоматизованої системи збору даних та інтеграції даних з інших сайтів в базу даних цієї системи.

Мета роботи – проектування та створення автоматизованої WEB-системи збору даних про нерухомість м.Одеси з інших web-сайтів та відображання її.

Мета роботи частини 1 – проектування та розробка WEB-системи, бази даних та необхідних функцій для роботи з базою даних серед яких розробка функції парсингу для застосування її в web-системі.

Для реалізації поставленої мети були вирішені наступні питання: проведено дослідження та аналіз існуючих систем по збору інформації про нерухомість; спроектовано архітектуру розробляємої web-системи; спроектовано та реалізовано базу даних, яка задовольняє всім вимогам до неї; розроблено парсинг даних з дошки оголошень OLX, з сайтів нерухомості ugo-zapad та president; розроблено автоматичне оновлення бази даних, для максимально актуальної інформації; розроблена функція реєстрації; розроблена функція видалення не актуальної інформації з бази даних; обрано хостинг.

Практична цінність магістерської роботи полягає в тому, що розроблена web-система по збору даних про нерухомість може бути використана для повноцінного пошуку користувачем необхідної інформації.

Ключові слова: web-система, пошукова система, пошук нерухомості.

Магістерська робота містить 64 сторінок, 1 таблиця, 26 рисунків, 16 посилань та 5 листів додатку.

ANNOTATION

The topic of master work "Designing and developing an automated system for collecting real estate data in the city of Odessa".

The urgency of the master's work is the need to develop a web-application that realizes parsing information about real estate into a single integrated system of centralized finding of the necessary information for clients.

The object of the study - the processes of developing an automated system for data collection and integration of data from other sites into the database of this system.

The purpose of the project is the design and creation of an automated WEB-system for collecting real estate data in Odesa from other web-sites and displaying it.

The purpose of Part 1 is the design and development of the WEB-system, databases and necessary functions for working with the database, including the development of a parsing function for the application of it in the web-system being developed.

To realize the set goal, the following issues were solved:
research and analysis of existing systems for the collection of information about real estate was conducted; the architecture of the web-system being developed is designed; the database has been designed and implemented that meets all the requirements for it; developed parsing of data from the OLX bulletin board, from ugo-zapad and president's real estate sites; developed an automatic updating of the database, for the most up-to-date information; the registration function has been developed; hosting is chosen; the function of removing non-relevant information from the database is developed.

The practical value of the master's work is that a developed web-system for collecting real estate data can be used to fully search the user for information of interest.

Keywords: web-system, search engine, real estate search.

ПОЯСНЮВАЛЬНА ЗАПИСКА

Останнім часом завдяки доступності інтернету усі кто хоче придбати нерухомість шукають інформацію в мережі інтернет, за допомогою різноманітних сайтів про нерухомості.

Так як сайтів дуже багато клієнти витрачають дуже багато часу для того, щоб знайти необхідну їм інформацію, переходячи з одного сайту на інший витрачаючи час на те, щоб розібратися в новому інтерфейсі.

Актуальність теми полягає в вирішені проблеми переходу по багатьом сайтам для пошуку необхідної інформації. На цей час є кілька сайтів які надають такі послуги, але вони не є повними, вони не отримують інформацію з усіх доступних ресурсів, деякі з них платні, деякі не мають достатньо добре побудованого пошуку, не всі з цих сайтів мають інтуітивно зрозумілий інтерфейс.

Метою даної комплексної магістерської роботи є розробка повнофункціональної web-системи, яка буде виключати недоліки все існуючих сайтів, на мові програмування PHP, мові розмітки HTML, системи управління базами даних MySQL, а також допоміжними засобами, такими як: мова JavaScript, бібліотека phpQuery, мова описання зовнішнього виду CSS та бібліотека curlQuery. Тестування даної системи буде виконано за допомогою спеціалізованого ресурсу loaddy.com.

Робота розділена на дві частини, по яких сформовані окремі мета та задачі:

Частина 1. Проектування та розробка серверної частини системи (виконавець – Кушнір Л.О.).

Мета роботи – проектування та розробка бази даних та необхідних функцій для роботи з базою даних серед яких розробка функції парсингу для застосування її в web-системі, що розробляється.

Для досягнення поставленої мети, треба вирішити наступні задачі:

1) спроектувати архітектуру WEB-системи;
2) спроектувати базу даних;
3) розробити парсинг;
4) розробити автоматичне оновлення бази даних, для максимально актуальної інформації;
5) розробити функцію реєстрації;

6) обрати хостинг;

7) розробити функцію видалення не актуальної інформації.

Частина 2. Розробка клієнтської частини системи (виконавець – Долготер Д.О.).

Мета роботи – розробка адаптивного та інтуітивно зрозумілого інтерфейсу системи.

Для досягнення поставленої мети, треба вирішити наступні задачі:

1) розробити зручний інтерфейс;
2) розробка функції пошуку необхідного об’єкту нерухомості за любими заданими характеристиками: вид нерухомості, район городу, кількість кімнат, обмеження по площі, ціні та т.п.;
3) розробити адаптивне відображення веб системи;
4) оптимізувати швидке завантаження веб системи;
5) оптимізувати коректне відображення на всіх сучасних браузерах;
6) розробити сучасний UI\UX;
7) дизайн сайту повинен відповідати потребам цільової аудиторії, бути чітким, зручним і зрозумілим.

Староста комплексної роботи Кушнір Л.О.

Провідний керівник комплексної роботи Шпінарева І.М.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ ТА ТЕРМІНІВ 9
ВСТУП 10
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ОГЛЯД ПОДІБНИХ СИСТЕМ 12
1.1 Web-сайт, загальні поняття 12
1.2 Класифікація web-сайтів 13
1.3 Огляд і аналіз відомих web-систем 17
1.4 Постановка задачі загальна 22
1.4.1 Постановка задачі 23
2 ПРОЕКТУВАННЯ СИСТЕМИ 24
2.1 Моделювання предметної області 24
2.2 Клієнт серверна архітектура 26
2.3 Триланкова архітектура системи 27
2.4 Засоби реалізації 30
2.4.1 Клієнтські технології (HTML5, CSS, Bootstrap, JavaScript) 31
2.4.2 Серверні технології PHP 35
2.4.3 Система управління базами даних MySQL 36
2.4.4 Парсер 37
2.5 Опис хостингів 39
3 ПРАКТИЧНА РЕАЛІЗАЦІЯ 41
3.1 Фізична архітектура 41
3.2 Проектування і розробка бази даних системи 43
3.2.1 Виділення інформаційних об'єктів 43
3.2.2 Визначення атрибутів об'єктів 44
3.2.3 Побудова концептуальної схеми 45
3.2.4 Особливості реалізації 47
3.3 Реалізація бази даних 48
3.4 Взаємодія з базою даних 50
3.5 Інформація по експлуатації web-системи 51
4 ОЦІНЮВАННЯ СИСТЕМИ 53
4.1 Оцінка системи на швидкість роботи 53
ВИСНОВКИ 55
ПЕРЕЛІК ПОСИЛАНЬ 56
Д О Д А Т К И 58

ДОДАТОК А Код бібліотеки CURLQUERY 59
ДОДАТОК Б Код функції реєстрації 60
ДОДАТОК В Код функції парсингу з сайту UGO-ZAPAD.UA 62

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ ТА ТЕРМІНІВ

Cascading Style Sheets , CSS – каскадні таблиці стилів

CRM, Customer relationship management – управління відносинами з клієнтами

HyperText Markup Language, HTML – мова гіпертекстової розмітки

Personal Home Page Tools, PHP – інструменти для створення персональних веб-сторінок

КІС – комп'ютерна інформаційна система

Управління відносинами з клієнтами – поняття, що охоплює концепції, котрі використовуються компаніями для управління взаємовідносинами зі споживачами, включаючи збір, зберігання і аналіз інформації про споживачів, постачальників, партнерів та інформації про взаємовідносини з ними.

Мініатюра (англ. Thumbnail), тж. «Прев'ю» (англ. Preview) – зменшена до невеликих розмірів копія зображення для компактного представлення користувачеві. Такі зменшені копії є звичайною практикою в будь-яких програмах і веб-додатках для роботи з графічними файлами (растровими, векторними, фрактальні).

ВСТУП

Пошук нерухомості – це завжди актуальне питання. Якщо потрібно знайти квартиру, будинок, землю для аренди або куплі/продажу. Потрібно звернутися до агентства нерухомості і заплатити купу грошей посереднику.

Web-технології впевненно проникають у всі сфери діяльності сучасної людини, особливо сильний розвиток вони отримують у сферах, зв’язанних з просуванням або реалізацією будь яких товарів.

В данній роботі мова іде об одній з таких сфер, а саме о сфері продажі нерухомості. Як правило інформація, що надається агентствами нерухомості різних фірм про пропозиції продажи житла не є повною. Багато з них не надають весь об’єм даних. Інші прагнуть підтримати весь об’єм в збиток актуальності даних – дані, отриманні від таких агрегаторів часто розходяться з фактичними пропозиціями забудовника. Це говорить о том, що об’єм даних на ринку нерухомості вище, ніж можливий для обробки співробітниками агентств вручну. Існують і так звані «професійні сервіси», послуги котрих недоступні для фізичних осіб.

Сайт будь-якого агентство нерухомості надає наступну інформацію про послуги:

- покупка / продаж квартир і кімнат;
- здача / підбір житла в наєм;
- реалізація споруджуваних об'єктів нерухомості за цінами забудовників;
- організація продажу квартир на стадії будівництва і за договорами поступки права вимоги;
- аренда комерційної нерухомості (здача / підбір);
- покупка / продаж будинків, котеджів, земельних ділянок.

Дуже часто власники квартир самостійно поміщають інформацію про продаж на сайтах оголошень OLX і ін. Для оптимізації роботи ріелторів, покупців і орендарів доцільно створення системи для автоматизованого збору даних про пропозиції продажу і оренди нерухомості. Першорядною задачею даної системи є мінімізація витрат часу на збереження і пошук даних в рамках єдиної інформаційної бази.

Метою дипломної роботи є створення web-системи для автоматизованого збору даних о пропозиціях продажи первинного та вторинного житла з інших сайтів.

Використяння електроної системи нерухомості дозволить економити час та гроші людини, що прагне купити, продати, здати або взяти в аренду нерухомість в Одесі.

Система дозволить легко ввести пошуковий запит, що засновано на фільтрах, та отримати найсучаснішу інформацію завдяки функції парсингу, що буде шукати інформацію яка опублікована останнім часом на спеціалізованних сайтах, а не лише з власної БД. Це забезпечить максимальну актуальність знайдених оголошень.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ОГЛЯД ПОДІБНИХ СИСТЕМ

1.1 Web-сайт, загальні поняття

Веб-сайт – у комп'ютерній мережі об'єднана під однією адресою (доменним ім'ям або IP-адресою) сукупність документів приватної особи або організації. За замовчуванням мається на увазі що сайт розташовується в мережі Інтернет. Всі веб-сайти Інтернету в сукупності складають Всесвітню павутину. Для прямого доступу клієнтів до веб-сайтів на серверах був спеціально розроблений протокол HTTP. Веб-сайти інакше називають Інтернет-представництвом людини чи організації. Коли говорять «своя сторінка в Інтернет», то мається на увазі цілий веб-сайт або особиста сторінка в складі чужого сайту. Крім веб-сайтів в мережі Інтернет так само доступні WAP-сайти для мобільних телефонів.

Спочатку веб-сайти являли собою сукупності статичних документів. В даний час більшості з них властива динамічність і інтерактивність. Для таких випадків фахівці використовують термін веб-додаток – готовий програмний комплекс для вирішення завдань веб-сайту. Веб-додаток входить до складу веб-сайту, але веб-додаток без даних сайтом є тільки технічно.

У більшості випадків в Інтернеті одному веб-сайту відповідає одне доменне ім'я. Саме по доменних іменах сайти ідентифікуються в глобальній мережі. Можливі інші варіанти: один сайт на декількох доменах або кілька сайтів під одним доменом. Зазвичай кілька доменів використовують великі сайти (веб-портали) щоб логічно відокремити різні види послуг, що надаються (mail.google.com, news.google.com, maps.google.com). Непоодинокі й випадки виділення окремих доменів для різних країн або мов. Наприклад, google.ru та google.fr логічно є сайтом Google на різних мовах, але технічно це різні сайти. Об'єднання декількох сайтів під одним доменом характерно для безкоштовних хостингів. Іноді для ідентифікації сайтів в адресі після вказівки хоста стоїть тильда і ім'я сайту: example.com/~my-site-name/, іноді використовується доменне ім'я третього рівня: my-site-name.example.com.

Апаратні сервера для зберігання веб-сайтів називаються веб-серверами. Сама послуга зберігання називається веб-хостингом. Раніше кожен сайт зберігався на своєму власному сервері, але з ростом Інтернету технологічним поліпшенням серверів на одному комп'ютері стало можливе розміщення без-

лічі сайтів (віртуальний хостинг). Зараз сервера для зберігання тільки одного сайту називаються виділеними (Dedicated).

Один і той же сайт може бути доступний за різними адресами і зберігається на різних серверах. Копія оригінального сайту в такому випадку називається дзеркалом. Існує так само поняття оффлайновая версія сайту – це копія сайту, яка може бути переглянута на будь-якому комп'ютері без підключення до комп'ютерної мережі і використання серверного програмного забезпечення [1].

1.2 Класифікація web-сайтів

Основні види сайтів: соціальні мережі, бізнес сайти, веб-сервіс, інформаційні сайти, комбіновані сайти.

Соціальна мережа – спрямована на побудову спільнот в Інтернеті з людей зі схожими інтересами та/або діяльністю. Зв'язок здійснюється за допомогою сервісу внутрішньої пошти або миттєвого обміну повідомленнями.

Сайт-візитка – містить загальні дані про власника сайту (організація або індивідуальний підприємець). Вид діяльності, історія, прайс-лист, контактні дані, реквізити, схема проїзду. Фахівці розміщують своє резюме. Тобто детальна візитна картка.

Представницький сайт – так іноді називають сайт-візитку з розширеною функціональністю: докладний опис послуг, портфоліо, відгуки, форма зворотного зв'язку і т.д.

Корпоративний сайт – містить повну інформацію про компанії-власника, послуги / продукцію, події в житті компанії. Відрізняється від сайту-візитки і представницького сайту повнотою наданої інформації, часто містить різні функціональні інструменти для роботи з контентом (пошук і фільтри, календарі подій, фотогалереї, корпоративні блоги, форуми). Може бути інтегрований з внутрішніми інформаційними системами компанії-власника (КІС, CRM, бухгалтерськими системами). Може містити закриті розділи для тих чи інших груп користувачів – співробітників, дилерів, контрагентів тощо.

Каталог продукції – в каталозі є докладний опис товарів/послуг, сертифікати, технічні та споживчі дані, відгуки експертів і т.д. На таких сайтах розміщується інформація про товари/послуги, яку неможливо помістити в прайс-лист.

Інтернет-магазин – веб-сайт з каталогом продукції, за допомогою якого клієнт може замовити потрібні йому товари. Використовуються різні системи розрахунків: від пересилання товарів післяплатою або автоматичною пересилання рахунку по факсу до розрахунків за допомогою пластикових карт.

Промо- сайт – сайт про конкретну торгову марку або продукт, на таких сайтах розміщується вичерпна інформація про бренд, різних рекламних акціях (конкурси, вікторини, ігри і т.п.).

Сайт-квест – Інтернет-ресурс, на якому організовано змагання з розгадування послідовності взаємопов'язаних логічних загадок.

Інформаційні сайти.

Тематичний сайт – веб-сайт, що надає специфічну вузькотематичні інформацію про будь-якої теми.

Комерційні сайти – до комерційних слід віднести ті сайти, які безпосередньо пов'язані з веденням бізнесу. З цієї точки зору серед них можна виділити: стимулюючі продажу в «оффлайн»-бізнес (а саме бізнесі, який сам по собі існує незалежно від існування Інтернету), орієнтовані на онлайн – комерцію (а саме ті види бізнесу, які без Інтернету неможливі, наприклад інтернет-магазини).

Основною аудиторією комерційного сайту є дійсні і потенційні клієнти. За своїм обсягом комерційний сайт може бути різних типів:

Сайт-візитка – складається буквально з п'яти-десяти сторінок і містить загальні відомості про власника сайту: його діяльність, реквізити, контактна інформація. Тобто такий сайт дійсно являє собою розширену візитну картку підприємця або організації.

Сайт-каталог – від сайту-візитки відрізняється в першу чергу наявністю докладного прайс-листа, каталогу продукції, що продається, в якому можуть бути перераховані докладні технічні характеристики товару.

Інтернет-магазин – містить каталог продукції, з якого користувач може прямо на сайті вибрати і замовити потрібні йому товари. Тобто обов'язковою ознакою інтернет-магазину є можливість зробити онлайн замовлення. Найчастіше ця можливість доповнюється функцією «кошика», яка дозволяє за один раз замовити кілька товарів.

Промо-сайт – створюється для презентації будь-якого конкретного товару або торгової марки і є частиною загальної маркетингової програми по

його просуванню. Містить докладний опис технічних характеристик і переваг товару, інформацію про різні акції, знижки і т.п.

Бізнес інтернет-портал – зазвичай належать великим компаніям з широким діапазоном видів діяльності. Містить дуже велика кількість сторінок, охоплює велику кількість тем. Відмінною рисою є надання користувачу широкого діапазону різних сервісів: пошта, форум, чат і т.п.

Некомерційні сайти – зазвичай здійснюють надання користувачам різної інформації (інформаційні сайти) або надання будь-яких онлайн послуг (веб-сервіси).

Особисті сторінки – так звані homepages (домашні сторінки), на яких людина розміщує інформацію про себе, фотографії та будь-яку ще інформацію по своєму бажанню. В даний час практично повністю витіснені блогами і соціальними мережами.

Тематичні інформаційні сайти – присвячені будь-якій певній, досить вузькій тематиці. Як приклади можна навести численні сайти шанувальників певних письменників, фільмів, музичних груп.

Пошукові системи. В даний час велика частина пошукових систем перетворилися на повноцінні інтернет-портали (Google, Яндекс, Рамблер). Проте як приклад «чистої» пошукової системи можна назвати Nigma.ru.

Поштові сервіси – всі великі поштові сервіси або стали великими Інтернет – порталами (наприклад, Ukr.net), або є частиною інших порталів (наприклад, Gmail.com як сервіс, що входить в Google, або Яндекс.Пошта як частина Яндекса).

Соціальні мережі – набули останнім часом просто фантастичну популярність. Концептуальний принцип таких веб-сайтів базуються на соціологічних теоріях соціальних мереж. Фактично їх функції полягають у можливості зареєстрованих користувачів знаходити в даній мережі певних людей і підтримувати з ними зв'язок, організовуватися в спільноти за інтересами, а також у наданні користувачам можливості розміщувати власний текстовий контент. Найбільшими соціальними мережами в світі є Facebook і MySpace.

Блогові сервіси – надають майданчики для ведення блогів, дозволяють користувачам вести колективні блоги (спільноти) і організовувати зручне взаємне читання. Слово «блог» (blog) – це скорочення від weblog – «веб-журнал». Фактично вдає із себе публічний онлайн щоденник. В даний час блоги мають дуже широкий спектр використання: від побутових заміток для

кількох друзів, до політичної трибуни для тисяч людей. Як приклади блогових сервісів можна привести livejournal.com, liveinternet.ru, сервіс мікроблогів twitter.com та інші.

Сервіси для зберігання фотографій – дані сервіси дають користувачам безкоштовний хостинг фотозображень і можливості для їх структурування та класифікації. Найбільш популярними фотохостинг є flickr.com, photobucket.com.

Сервіси для зберігання і відтворення відео – на такі сайти користувачі можуть завантажувати свої відеозаписи, які можуть переглядати всі бажаючі. Більш того, сервіси надають можливість розмістити це відеозображення на будь-якому сайті. Найбільшим місцем для розміщення відеофайлів є YouTube.com.

Дошки оголошень – фактично аналогічні звичайним побутовим дошкам оголошень або рекламним газетам. Кожен бажаючий може вивісити своє оголошення, а всі відвідувачі сайту – прочитати його. Дошки оголошень, як правило, поділені на кілька тематичних розділів.

Каталоги сайтів – містять структуровані списки посилань на веб-сайти, забезпечені їх короткими описами. Масштаб різних каталогів може сильно різнитися: від декількох сотень посилань до глобальних всесвітніх рейтингів. До останніх слід віднести DMOZ і каталог Yahoo.

Тематичні інформаційні Інтернет – портали представляють собою розширену версію тематичних сайтів з великою кількістю новин, статей та інших матеріалів. Зазвичай мають більш широкою тематикою, ніж тематичні сайти, і надають користувачам різні сервіси та інші можливості. Як приклад можна привести портали avto.ru, sports.ru.

Онлайн енциклопедії – в переважній більшості випадків базується на вікі-принципі, який полягає в тому, що будь-який користувач може самостійно додавати нові статті та вносити зміни (редагування) в уже існуючі. Найвідомішою вікі-енциклопедією є Wikipedia.

Очевидно, що більша частина перерахованих вище сайтів тим чи іншим способом займаються зароблянням грошей (розміщенням реклами або будь-якими платними послугами для користувачів). Однак такі сайти в першу чергу створені не для отримання прибутку, а для задоволення потреб користувачів, тому їх можна назвати некомерційними. Втім, подібні розмежування до-

сить примарні, так що багато хто з цих сайтів можна віднести до категорії «умовно некомерційних» [2,3,4,5].

1.3 Огляд і аналіз відомих web-систем

Web-система HOTPRO.ua

На сьогоднішній день, ринок подібних систем є дуже обмеженим. Немає єдиної цілісної системи, яка б надавала повну зручність та актуальність.

В основі назви HOTPRO.ua лежать англійські слова hot (гаряче, актуально) та property (майно, нерухомість). HOTPRO.ua пропонує продаж квартир, домів, комерційну нерухомість та земельні частки. (рис. 1.1).

Рисунок 1.1 – Головна сторінка Hotpro

Портал HOTPRO.ua дає можливість зручно і швидко знайти тільки цікаві саме вам пропозиції по нерухомості, процедура модерації публікуються пропозицій відсіває всі пропозиції, які не задовольняють правилам порталу, а в додаткових розділах ви знайдете допоміжну інформацію, яка може знадобитися (рис.1.2).

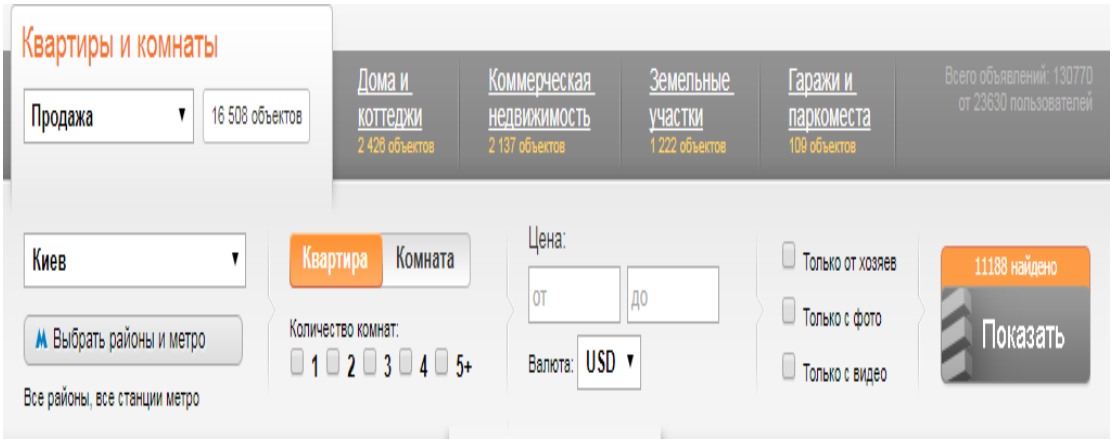


Рисунок 1.2 – Пошук на сайті

Портал HOTPRO.ua включає в себе все найкраще з досвіду вітчизняних і зарубіжних колег, які працюють в сфері нерухомості. Крім цього, реалізовані ідеї, унікальні для вітчизняного, а в якійсь мірі і для світового онлайн-ринку нерухомості. Всі рішення, представлені на порталі, оформлені в сучасному дизайні і забезпечені простим, інтуїтивно зрозумілим інтерфейсом, що робить роботу ефективною і приємною (рис.1.3).

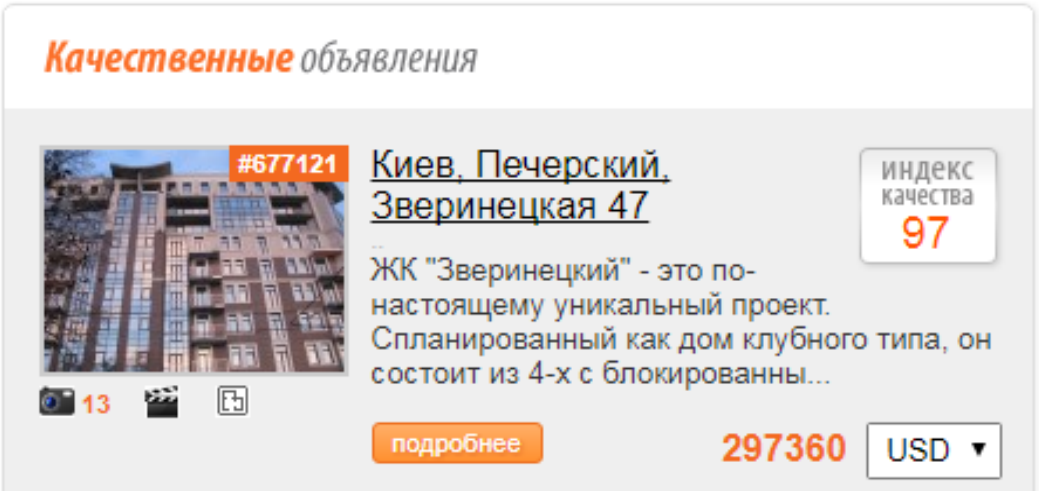


Рисунок 1.3 – Прев'ю поста

З недоліків можна виділити такі як:

– не достатньо точний пошук;

– довге завантаження при виборі конкретного типу нерухомості.

Web-система LUN.ua

LUN.ua – це пошукова система, як Google або Яндекс, яка шукає оголошення по нерухомості на всіх сайтах і порталах з нерухомості в Україні Для того, щоб оголошення з'явилося на LUN.ua, його потрібно додати на один або кілька порталів по нерухомості.

Головна сторінка на сайті LUN.ua (рис. 1.4) зустрічає нас панорамою, видом з квадрокоптеру.

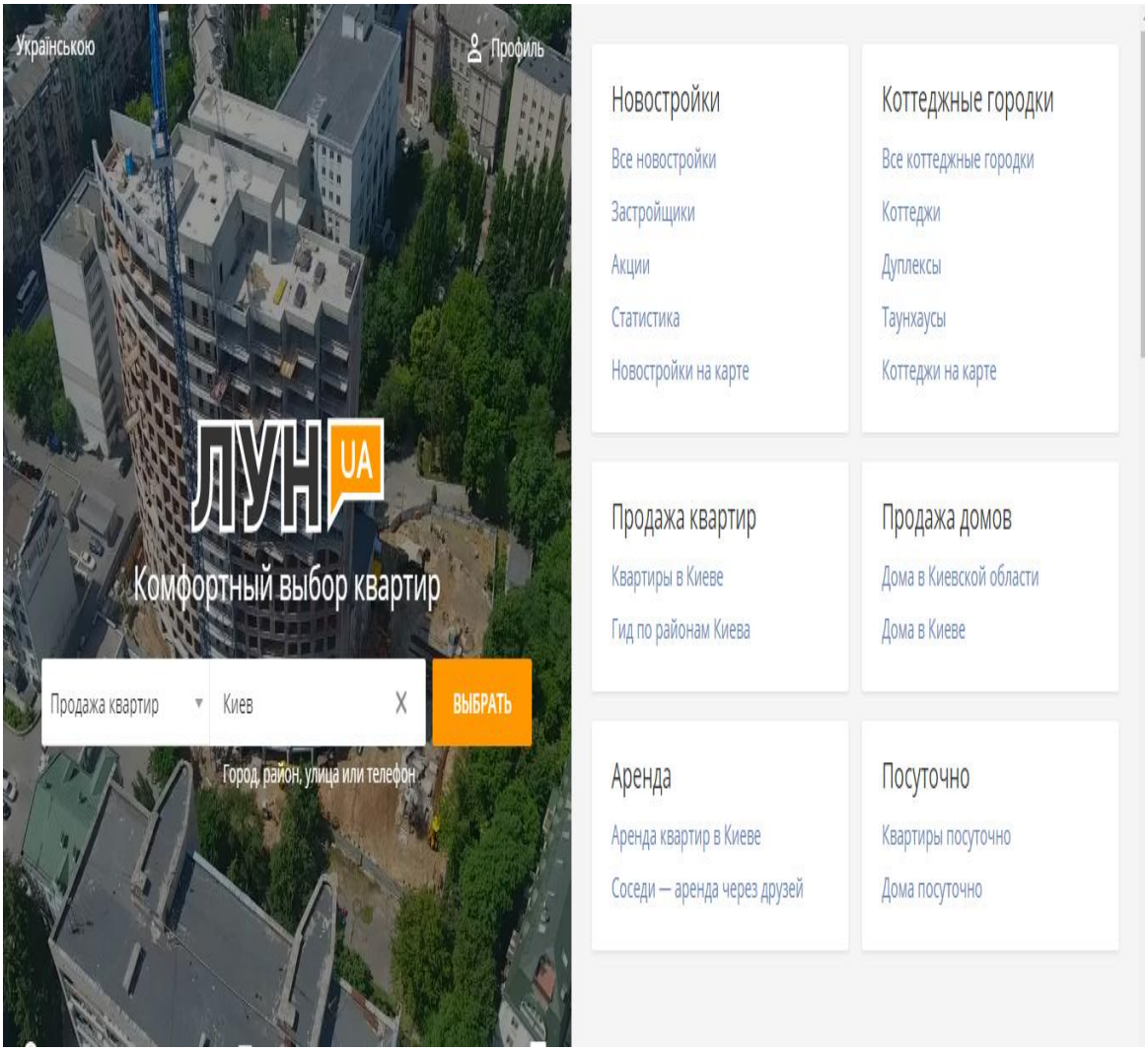


Рисунок 1.4 – Головна сторінка на сайті ЛУН

Пошук на сайті (рис.1.5) для зручності поділено на декілька полів.

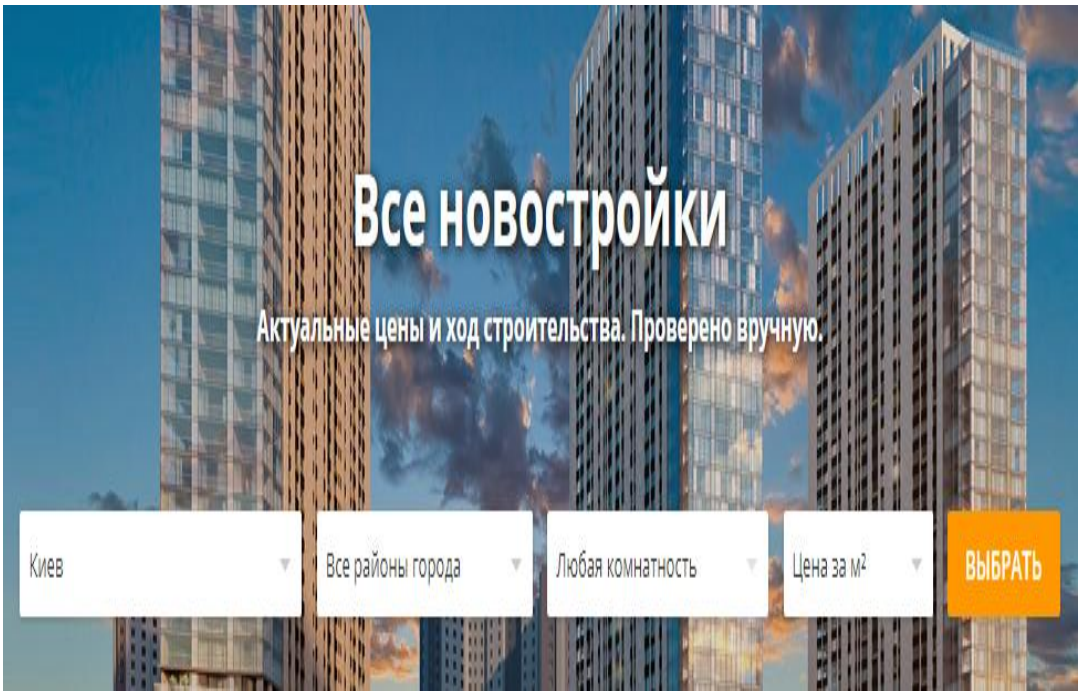


Рисунок 1.5 – Пошук на сайті

Прев'ю оголошення (рис. 1.6) зображення і підписи з інформацією про квартиру.

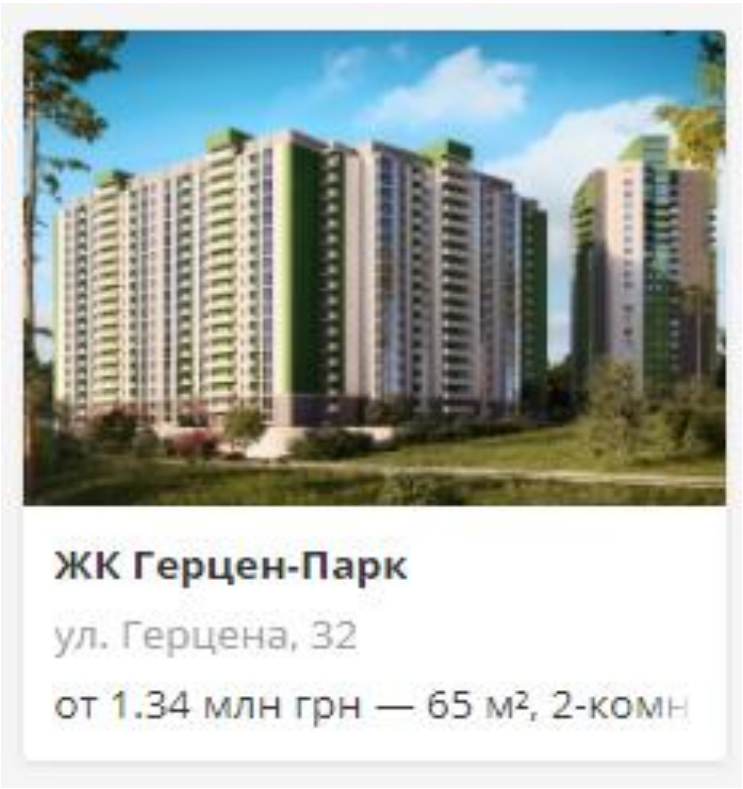


Рисунок 1.6 – Прев'ю поста

Повний опис поста для квартири містить: опис квартири, адресу, ціну, метраж (рис.1.7).

Рисунок 1.7 – Повний пост

З недоліків можна виділити такі як не зручний пошук.

Висновок

В результаті дослідження інформаційних сайтів пропозицій продажу нерухомості популярних в Одесі був сформований перелік всіх недоліків, котрі необхідно виправити:

- незручний інтерфейс;
- складно знайти потрібну функцію;
- довге завантаження;
- не зручний інтерфейс на мобільних телефонах.

Переваги, котрі необхідно підкреслити:

- світлий дизайн;
- є вся необхідна користувачеві інформація.

В результаті дослідження сайтів нерухомості пропозицій продажу та аренди нерухомості і інформаційних сайтів популярних в Одесі забудовників був сформований перелік всіх зустрінутих параметрів квартир.

Район міста – назва району міста, до якого територіально відноситься квартира

Житловий комплекс – назва житлового комплексу. це один або (частіше) кілька багатоквартирних житлових будинків, об'єднаних єдиною, спеціально спланованою територією, комплексом підприємств обслуговування (підприємства торгівлі, послуг, соціального призначення та т. д.), побудованих в єдиному архітектурному стилі і утворюють єдину територіально-просторову цілісність.

Термін здачі первинної нерухомості – дата здачі нерухомості.

Поверховість – кількість поверхів в будинку.

Поверх – поверх, на якому розташована квартира, яка продається.

Вулиця – частина адреси квартири.

Номер будинку – частина адреси квартири.

Кількість кімнат.

Площа загальна – сумарна площа всіх приміщень квартири.

Площа житлова – сумарна площа житлових кімнат.

Площа кухні.

Ціна – вартість в грн.

Продавець – назва юридичної особи, яка виступає продавцем квартири.

1.4 Постановка задачі загальна

Метою роботи є проектування та створення автоматизованої WEB-системи збору даних про нерухомість м.Одеси з інших web-сайтів та відображання її.

Ціль автоматизації – систематизація та підвищення ефективності роботи клієнтів та ріелторів з даними об об’єктах нерухомості.

Для виконання данної мети требо виконати наступні задачі:

–проектування архитектури WEB-системи;

–проектування та ведення бази даних об’єктів нерухомості, продаваних та здаваємих в аренду: квартир, житлових домів, комерційної нерухомості, земельних часток;

–зберігання для любого об’єкта нерухомості довільного набору характеристик (площа, поверх, обробка, стадія будування та т.п.);

–розробка парсера для збору даних о нерухомості з Web-сайтів;

–розробка автоматизованого оновлення бази даних;

–розробка функції пошуку необхідного об’єкту нерухомості за любими заданими характеристиками: вид нерухомості, район городу, кількість кімнат, обмеження по площі, ціні та т.п.;

–розробка функції видалення не актуальної інформації;

–розробка функції реєстрації, розділення прав доступу і авторизації користувачів;

–розробка функції печаті вихідних документів: список об’єктів нерухомості, відібраних по заданим крітеріям.;

–розробка адаптивного інтерфейсу та кросбраузерність;

– вібир хостингу.

1.4.1 Постановка задачі

Мета роботи – проектування та розробка WEB-системи, бази даних та необхідних функцій для роботи з базою даних серед яких розробка функції парсингу для застосування її в web-системі що розробляється.

Для досягнення поставленої мети розробки, треба вирішити наступні задачі:

1) спроектувати архітектуру WEB-системи;
2) спроектувати базу даних;
3) розробити парсинг;
4) розробити автоматичне оновлення бази даних, для максимально актуальної інформації;
5) розробити функцію реєстрації;
6) обрати хостинг;
7) розробити функцію видалення не актуальної інформації.

2 ПРОЕКТУВАННЯ СИСТЕМИ

2.1 Моделювання предметної області

Побудова контекстної діаграми

Для побудови контекстної діаграми було використано програму ERwin Process Modeller (рис. 2.1).

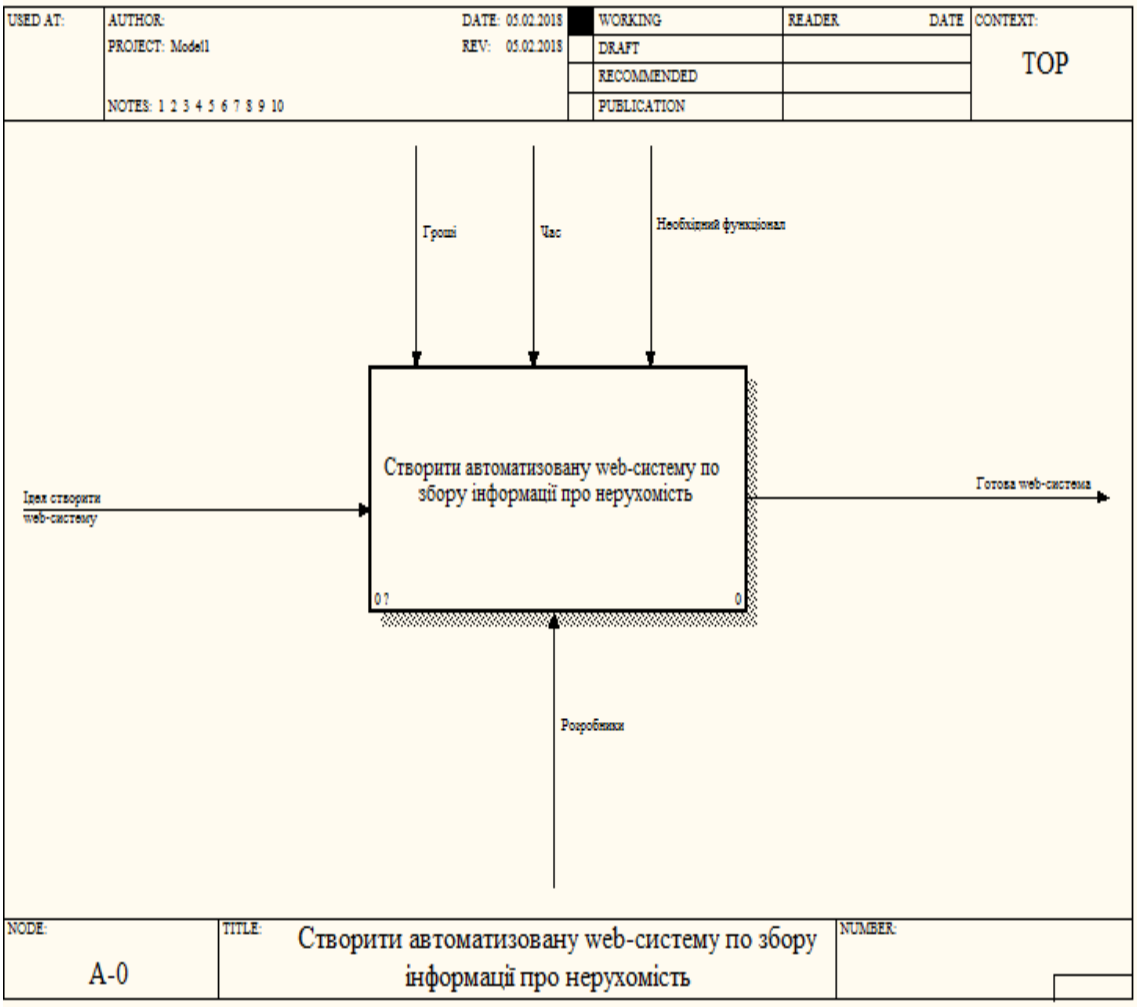


Рисунок 2.1 – Контекстна діаграма

За допомогою цієї програми можна легко зрозуміти послідовність необхідну для написання web-системи.

Головною метою та остаточною цілю є «Працююча web-система». Роблячи web-системи нам необхідно враховувати інтереси і побажання замовника. Також програміст повинен розрахувати кількість часу необхідного для напи-

сання web-системи і гроші витрачені на хостинг і можливо дизайн сайту.

Побудова діаграми декомпозиції

При створенні діаграмі декомпозиції необхідно в першу чергу розбити написання web-системи на деякі етапи (рис. 2.2). Розбиваючи на етапи виділяються головні пункти в самій роботі.

Головними пунктами будуть:

1) розробити функціонал;
2) розробити базу даних;
3) розробити інтерфейс;
4) отримати хостинг;
5) розмістити web-систему в інтернеті.

Виконуючи кожний пункт, буде отримано нові вхідні параметри для наступного пункту.

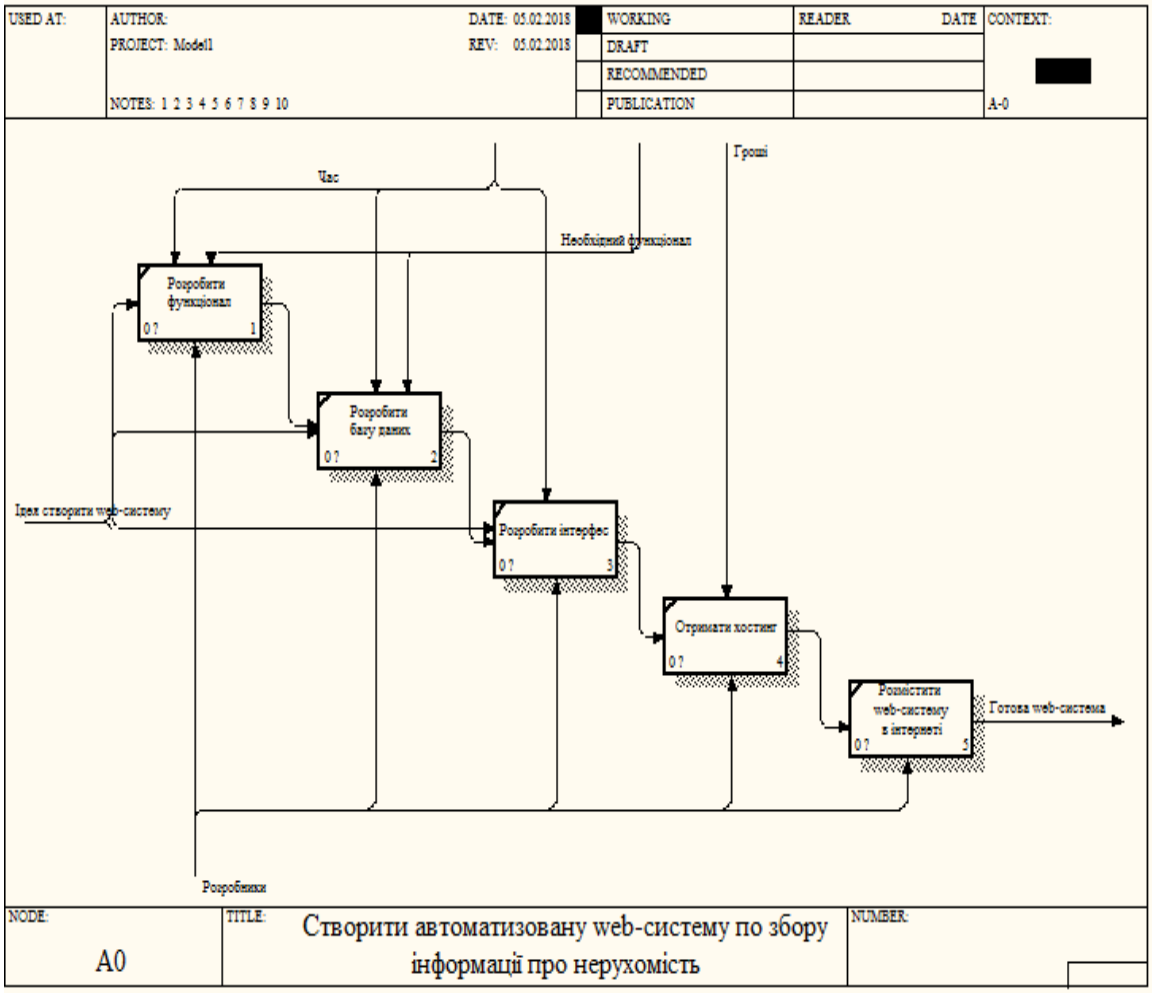


Рисунок 2.2 – Діаграма декомпозиції

Розробляєма система призначена для сбору інформації зі всіх сайтів о нерухомості міста. Таким чином, користувачі та ріелтори можуть на сайті отримати інформацію о продажу або аренді інтересуючої нерухомості (рис.2.3).

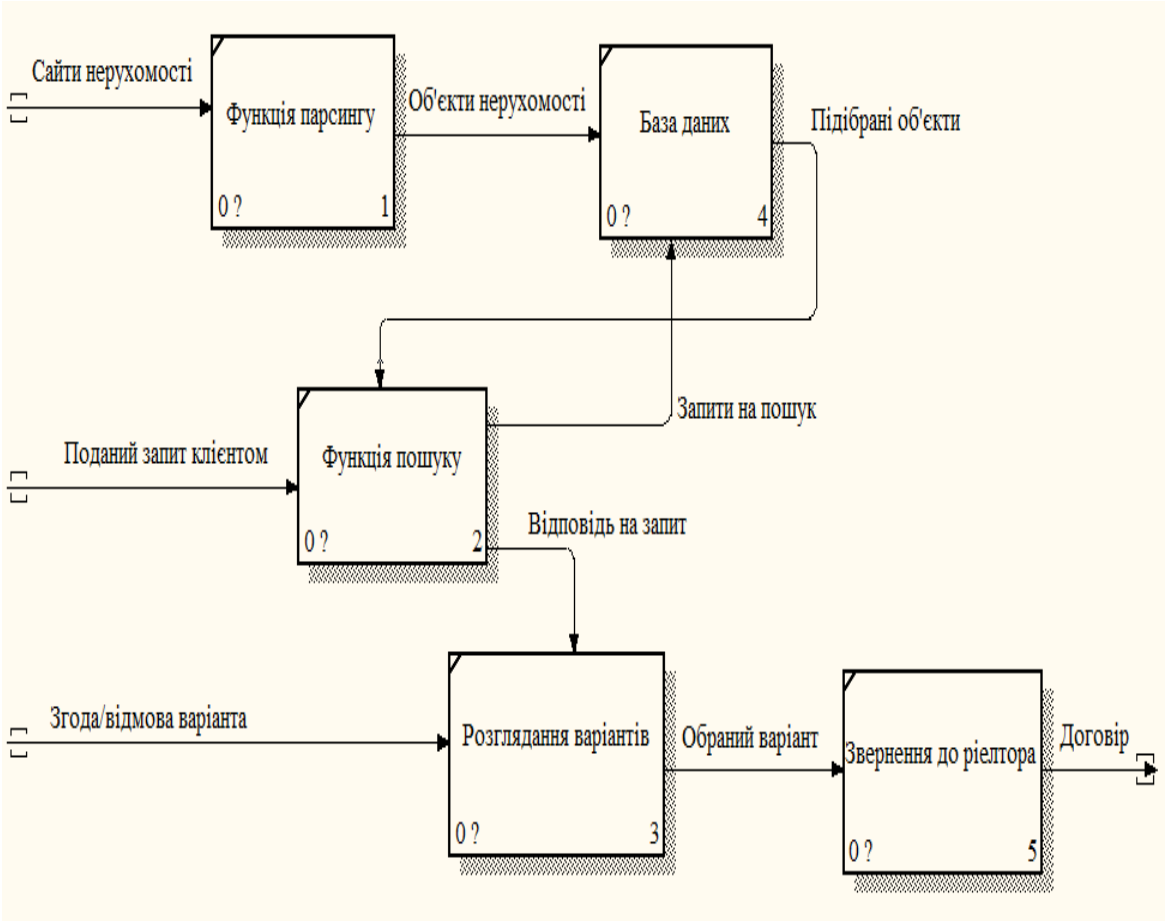


Рисунок 2.3 – Контекстная диаграмма роботи системи

2.2 Клієнт серверна архітектура

Двохланкова «клієнт – серверна» архітектура (рис. 2.4) дозволяє обмінюватися клієнту та серверу мінімально необхідними об’ємами інформації. При цьому основна обчислювальна нагрузка лягає на сервер. Клієнт може виконувати функції попередньої обробки перед передачею інформації на сервер, але в основному його функції полягають в організації доступу користувача до сервера.

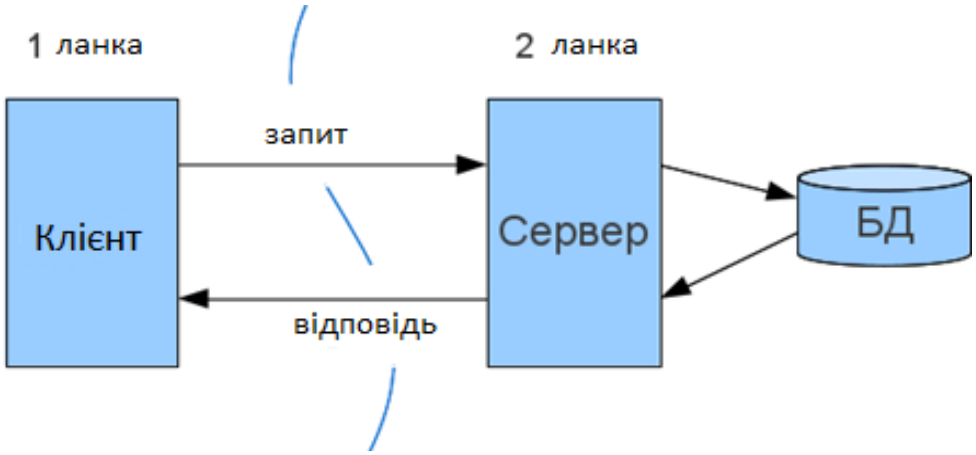


Рисунок 2.4 – Двухланкова «клієнт-серверна» архітектура

Двухланкова архітектура використовується в клієнт – серверних системах, де сервер відповідає на клієнтські запити напряму та в полному об’ємі, при цьому використовуючи тільки власні ресурси. Тобто сервер не викликає сторонні мережеві додатки та не звертається до сторонніх ресурсів для виконання будь-якій частини запросу.

2.3 Триланкова архітектура системи

Для реалізації взаємодії клієнта з сервером була використана триланкова архітектура (рис.2.5). Перевагою триланокової архітектури перед дволанковою є можливість поділу серверної частини на кілька окремих серверів, що дозволяє збільшити продуктивність програми.

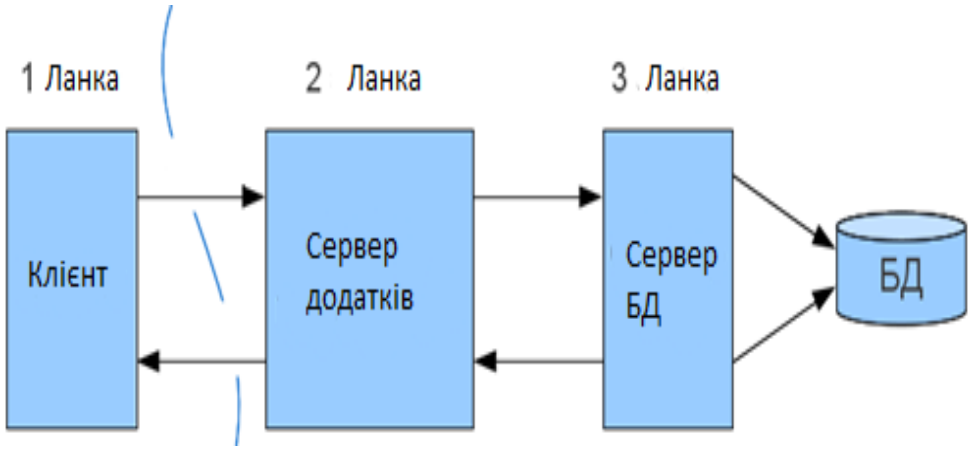


Рисунок 2.5 – Триланкова архітектура

Для реалізації проекту було прийнято рішення використовувати архітектуру MVC. Модель MVC була використана в зв'язку з необхідністю відділення шару бізнес-логіки від її подання, можливістю юніт-тестування (рис.2.6) .

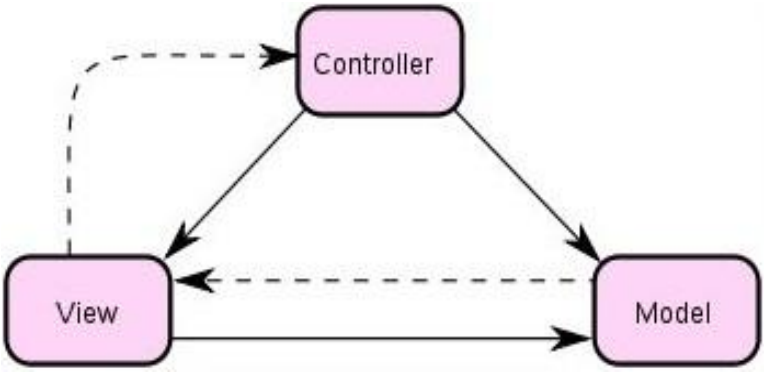


Рисунок 2.6 – Архітектура MVC

Загальна структура Web-додаток складається з клієнтської та серверної частини, тим самим реалізуючи технологію "клієнт-сервер" (рис. 2.7).

Рисунок 2.7 – Загальна структура web-додатка

Клієнтська частина реалізує інтерфейс що призначений для користувача, формує запити до сервера та обробляє відповіді від нього.

Серверна частина отримує запит від клієнта, виконує обчислення, після цього формує web-сторінку та відправляє її клієнту по мережі вікористовуючи протокол HTTP.

Сам web-додаток може виступати в якості клієнта інших служб, наприклад бази данних або іншого web-додатку, розташованного на іншому сервері.

Для створення web-додатків на стороні сервера використовуются різнообразні технології та будь які мови программування, які здатні здійснювати вивід в стандартну консоль.

Для web-додатку що парсить данні була розроблена функціональна структура, яка представлена на рис. 2.8.

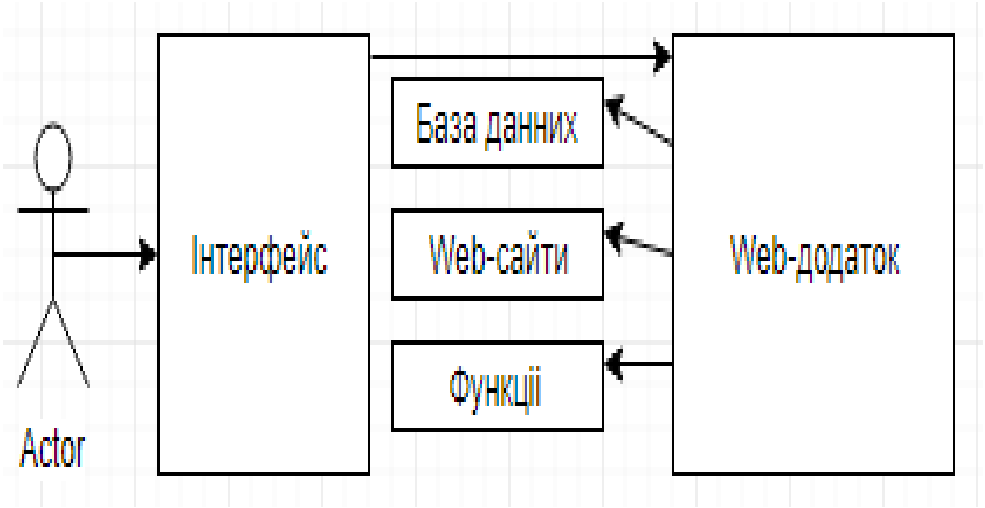


Рисунок 2.8 – Структура web-додатку

Етап розробки архітектури web-додатку є дуже важливим, з огляду на те, що на цьому етапі визначаються принципи роботи системи, користувацькі інтерфейси, а також компоненти системи та зв’язки між ними. Запропонована система складається з 5 модулів: інтерфейсу, бази данних, переліку сайтів, функцій та обробника (web-додатку).

Модуль обробник служить для синхронізації роботи всіх модулів web-додатку і організації взаємодії з клієнтом. З його допомогою здійснюється вибір необхідних для цього методів і моделей.

Модуль візуалізації служить для графічного відображення оброблених даних.

Модуль бази даних дозволяє зберігати дані. Являє собою БД певної структури, яка забезпечує швидке відображення даних. Дані з БД використовуються безпосередньо web-додатком для відображення данних клієнту.

Модуль функцій містить у собі опис для обробки даних, виконавчий код та методи роботи з базою даних.

Модуль переліку сайтів служить для зберігання переліку сайтів, опис змінних які потрібно знайти на сайті, метод підключення до сайтів і т.п.

Для проектування web-додатку скористаємося об’єктно-орієнтованим підходом на базі універсальної мови моделювання UML, а саме діаграмою варіантів використання (use case diagram).

Проаналізувавши всі вимоги до системи, виділено наступні варіанти використання. Вони зображені в вигляді діаграми варіантів використання (рис. 2.9). Діаграма варіантів використання дозволяє швидко побачити основні функції, які буде виконувати система.

Рисунок 2.9 – Діаграма варіантів використання

2.4 Засоби реалізації

Щоб система працювала так, як треба і змогла удовольнити усіх потреб, вірішено було використовувати наступні інструменти.

2.4.1 Клієнтські технології (HTML5, CSS, Bootstrap, JavaScript)

HTML5 – наступна версія мови HTML. До складу робочої групи з HTML5 увійшли AOL, Apple, Google, IBM, Microsoft, Mozilla, Nokia, Opera та кілька сотень інших виробників.

Існує деяка плутанина щодо версійності, оскільки існують дві незалежні групи розробників – WHATWG та W3C.

WHATWG відмовились від принципу «версійності», в користь «вічної розробки» при прийнятті HTML специфікації. Таке рішення було спричинено намаганням пришвидшити втілення стандарту в життя, тобто розробникам веб браузерів не потрібно чекати допоки вийде офіційна затверджена версія специфікації (специфікація перейде в стан recomendation), вони можуть втілювати певні частини специфікації вже зараз. Тому за версією WHATWG існує тільки одна специфікація, яка постійно розвивається – HTML.

Ці дві групи працювали в тандемі, WHATWG писав специфікації в режимі «живого стандарту», а W3C брав ці специфікації як «знімки», і впроваджував їх у чіткі версії своєї специфікації. W3C працював значно повільніше, бо повинен забезпечувати вимоги більшого спектра користувачів, а не тільки веб-браузерів.

28 жовтня 2014 консорціум W3C оголосив про надання набору специфікацій HTML5 статусу рекомендованого стандарту. На час стандартизації HTML5 вже давно став стандартом де-факто і активно використовується у веб-застосунках. Фактичне затвердження стандарту лише формально поставило крапку в просуванні HTML5 і підтвердило повсюдність і коректність його реалізації.

Специфікації HTML5 не обмежуються тільки розміткою і включають в себе низку веб-технологій, котрі у сукупності формують відкриту Веб-платформу– програмне оточення для роботи крос-платформових застосунків, здатних взаємодіяти з обладнанням, і які підтримують засоби для роботи з відео, графікою і анімацією, що надає розширені мережеві можливості.

Каскадні таблиці стилів – спеціальна мова, що використовується для опису зовнішнього вигляду сторінок, написаних мовами розмітки даних.

Найчастіше CSS використовують для візуальної презентації сторінок, написаних HTML та XHTML, але формат CSS може застосовуватися до інших видів XML-документів.

Специфікації CSS були створені та розвиваються Консорціумом Всесвітньої мережі.CSS3 – це новий стандарт оформлення HTML документів значно розширює можливості попереднього стандарту CSS2.1.

Багато можливості, які були важкодоступні в CSS2.1, тобто вимагали використання додаткових зовнішніх програм (таких як Adobe Photoshop), скриптів (таких як JavaScript) або спеціальних "хитрощів" можуть легко досягатися в CSS3 за рахунок використання нових властивостей оформлення.

В CSS3 можно:

– створювати елементи зі згладженими кутами;

– створювати лінійні і сферичні градієнти;

– гнучкіше оформляти фонову картинку елементів;

– додавати до елементів і до тексту елементів тіні;

– використовувати небезпечні шрифти (не боячись при цьому, що вони будуть не підтримуватися браузером користувача);

– створювати анімацію і різні ефекти переходів;

– задавати кольору декількома новими способами і багато іншого.

Bootstrap – це безкоштовний набір інструментів з відкритим кодом, призначений для створення веб-сайтів та веб-додатків, який містить шаблони CSS та HTML для типографіки, форм, кнопок, навігації та інших компонентів інтерфейсу, а також додаткові розширення JavaScript. Він спрощує розробку динамічних веб-сайтів і веб-додатків.

Bootstrap – це клієнтський фреймворк, тобто інтерфейс для користувача, на відміну від коду серверної сторони, який знаходиться на сервері. Репозиторій з даним фреймворком є одним з найбільш популярних на GitHub. Серед інших, його використовують NASA і MSNBC.

Четверта версія фреймворку має:

- використання синтаксису Sass замість Less;
- підтримка flexbox, компоненту з HTML5;
- прев'ю та панелі замінено компонентом «карти» – це віднині невеликі за форматом елементи з прев'ю зображень, текстових блоків з бордерами;
- всі HTML-резети зібрано в єдиному модулі під назвою «Reboot» (в попередніх версіях цей код зберігався в Normalize.css);

– нові можливості з кастомізації шаблонів. Для оновлення стилів за замовчуванням досить відредагувати змінну в Sass-файлі і отримати оновлених файл css;
– оновлення всіх плагінів JavaScript;
– оновлено роботу спливних вікон і підказок;
– покращено документацію і пошук сайтом фреймворку.

JavaScript – це особлива мова програмування, який базується на об'єктному представленні браузера. Він потрібен для того, щоб надати сайту більше інтерактивності в порівнянні зі звичайним статичним HTML-документом. Відмінність JavaScript полягає в тому, що текст програми вбудовується в документ HTML і аналізується самим браузером. JavaScript – це мова програмування сценаріїв на веб-сторінках.

Відмінність JavaScript від Java.

Знаючи, що таке JavaScript, багато користувачів все одно плутають цей термін з іншим поняттям – Java. Хоч мови і схожі за назвою, але мають вони різні значення. Основні відмінності полягають в складності і в кількості можливостей.

Реалізація JavaScript більш вільна у порівнянні з Java. Перетворення типів даних, наприклад, відбувається набагато простіше. Також програмісту не потрібно буде компілювати вихідний код програми на мові JavaScript, тобто JavaScript є інтерпретується мовою.

У JavaScript програма обробляється рядок за рядком, та інформація про помилки видається після кожної прочитаної рядки, якщо вони є. В в Java компілятор видає ці відомості після прочитання всього тексту програми. Зверніть увагу, що JavaScript не розглядається як заміна мови програмування Java. Краще за все перший використовувати як доповнення до другого.

Переваги JavaScript.

– За допомогою JavaScript створюються динамічні документи HTML.

– JavaScript пов'язує воєдино всі будівельні блоки програми, це як би засіб побудови фундаменту.

– JavaScript здійснює перевірку полів форм HTML до того, як вони передалися на сервер.

– Управління програмою на даній мові програмування йде через локальний введення інформації.

jQuery – популярна JavaScript-бібліотека з відкритим сирцевим кодом. Вона була представлена у січні 2006 року у BarCamp NYC Джоном Ресіґом (John Resig). Згідно з дослідженнями організації W3Techs, JQuery використовується понад половиною від мільйона найвідвідуваніших сайтів. jQuery є найпопулярнішою бібліотекою JavaScript, яка посилено використовується на сьогоднішній день.

jQuery є вільним програмним забезпеченням під ліцензією MIT (до вересня 2012 було подвійне ліцензування під MIT та GNU General Public License другої версії).

Синтаксис jQuery розроблений, щоб зробити орієнтування у навігації зручнішим завдяки вибору елементів DOM, створенню анімації, обробки подій, і розробки AJAX-застосунків. jQuery також надає можливості для розробників, для створення плагініву верхній частині бібліотеки JavaScript. Використовуючи ці об'єкти, розробники можуть створювати абстракції для низькорівневої взаємодії та створювати анімацію для ефектів високого рівня. Це сприяє створенню потужних і динамічних веб-сторінок.

jQuery UI побудований на базі фреймворка і спрямований на оформлення основних елементів інтерфейсної частини сайту (меню, текстові блоки, галереї і т.д.). Використання фреймворків в розробці є характерною ознакою сучасного сайту і, в даний час, є необхідним елементом.

На базі jQuery розроблений новий фреймворк керуючий динамічним оформленням клієнтської частини сайту Bootstrap – популярний фреймворк, який дозволяє швидко і якісно створювати статичні веб-сайти і веб-додатки. По суті, це безкоштовний набір інструментів, що дозволяє використовувати Html, CSS і JavaScript «великими мазками».

Brackets – текстовий редактор від компанії Adobe, призначений для редагування JavaScript, HTML і CSS. Сирцевий кодBrackets написаний з використанням веб-технологій (JavaScript, HTML і CSS) і поширюється під ліцензією MIT. Редактор оформлений у вигляді відокремленого настільного застосунка, для установки якого підготовлені deb-, dmg- і msi- пакети для Linux, OS X і Windows.

Brackets підтримує режим Live-розробки, при якому редагований контент (JavaScript, HTML і CSS) у міру зміни відразу відображається в синхронізованому з редактором вікні браузера – розробник може змінювати вміст і відразу спостерігати до яких наслідків приводять дані зміни. Налагодження

також може виконуватися синхронно із браузером, розробник може встановити точку зупину або відкотитися на крок назад при перегляді результатів. Є вбудована підтримка препроцесорів LESS і SCSS. В інтерфейсі застосовується система контекстно-залежних інструментів, що з'являються в міру необхідності в основному вікні розробки. Для розширення можливостей редактора розвивається система доповнень.

2.4.2 Серверні технології PHP

PHP – скриптова мова програмування, була створена для генерації HTML-сторінок на сторонівеб-сервера. PHP є однією з найпоширеніших мов, що використовуються у сфері веб-розробок (разом із Java.NET, Perl, Python, Ruby). PHP підтримується переважною більшістю хостинг-провайдерів. PHP – проектвідкритого програмного забезпечення.

PHP інтерпретується веб-сервером у HTML-код, який передається на сторону клієнта. На відміну від скриптової мови JavaScript, користувач не бачить PHP-коду, бо браузер отримує готовий html-код. Це є перевага з точки зору безпеки, але погіршує інтерактивність сторінок. Але ніщо не забороняє використовувати PHP для генерування і JavaScript-кодів які виконуються вже на стороні клієнта.

PHP є мовою програмування з динамічною типізацією, що не вимагає вказівки типу при оголошенні змінних, так само як і самого оголошення змінних. Перетворення між скалярними типами часто здійснюються неявно без додаткових зусиль (втім, PHP надає широкі можливості і для явного перетворення типів).

Велика різноманітність функцій PHP дає можливість уникати написання багаторядкових функцій, призначених для користувача, як це відбувається в C або Pascal.

1) Наявність інтерфейсів до багатьох баз даних. В PHP вбудовані бібліотеки для роботи з MySQL, PostgreSQL, mSQL, Oracle, dbm, Hyperware, Informix, InterBase, Sybase. Через стандарт відкритого інтерфейсу зв'язку з базами даних (Open Database Connectivity Standard) можна підключатися до всіх баз даних, до яких існує драйвер.

2) Традиційність. Мова PHP здаватиметься знайомою програмістам, що працюють в різних областях. Багато конструкцій мови запозичені з C, Perl.

Код PHP дуже схожий на той, який зустрічається в типових програмах на C або Pascal. Це помітно знижує початкові зусилля при вивченні PHP. PHP – мова, що поєднує переваги Perl і C і спеціально спрямована на роботу в Інтернеті, мова з універсальним і зрозумілим синтаксисом. І хоча PHP є досить молодою мовою, вона здобула таку популярність серед web-програмістів, що в наш час є мало не найпопулярнішою мовою для створення веб-застосунків (скриптів).

3) Наявність сирцевого коду та безкоштовність. Стратегія Open Source, і розповсюдження початкових текстів програм в масах, безсумнівно справили благотворний вплив на багато проектів, в першу чергу – Linux хоч і успіх проекту Apache сильно підкріпив позиції прихильників Open Source. Сказане відноситься і до історії створення PHP, оскільки підтримка користувачів зі всього світу виявилася дуже важливим чинником в розвитку проекту PHP.

Ухвалення стратегії Open Source і безплатне розповсюдження початкових текстів PHP надало неоціненну послугу користувачам. Окрім цього, користувачі PHP в усьому світі є свого роду колективною службою підтримки, і в популярних електронних конференціях можна знайти відповіді навіть на найскладніші питання.

4) Ефективність.Ефективність є дуже важливим чинником у програмуванні для середовищ розрахованих на багато користувачів, до яких належить і web. Важливою перевагою PHP є те, що ця мова належить до інтерпретованих. Це дозволяє обробляти сценарії з достатньо високою швидкістю. За деякими оцінками, більшість PHP-сценаріїв (особливо не дуже великих розмірів) обробляються швидше за аналогічні їм програми, написані на Perl. Проте хоч би що робили розробники PHP, виконавчі файли, отримані за допомогою компіляції, працюватимуть значно швидше – в десятки, а іноді і в сотні разів. Але продуктивність PHP достатня для створення цілком серйозних веб-застосунків [6,7,8,9].

2.4.3 Система управління базами даних MySQL

MySQL – вільна реляційна система управління базами даних. Розробка та підтримка сайта MySQL здійснює корпорація Oracle, яка отримала права на торговельну марку разом з поглиненої Sun Microsystems, яка раніше придбала шведську компанію MySQL AB. Продукт поширюється як під GNU

General Public License, так і під власною комерційною ліцензією. Крім цього, розробники створюють функціональність за замовленням ліцензійних користувачів. Саме завдяки такому замовленню майже в найраніших версіях з'явився механізм реплікації.

MySQL є рішенням для малих і середніх додатків. Входить до складу серверів WAMP, AppServ, LAMP і в портативні збірки серверів Денвер, XAMPP, VertrigoServ. Зазвичай MySQL використовується як сервер, до якого звертаються локальні або видалені клієнти, проте в дистрибутив входить бібліотека внутрішнього сервера, що дозволяє включати MySQL в автономні програми.

Гнучкість СУБД MySQL забезпечується підтримкою великої кількості типів таблиць: користувачі можуть вибрати як таблиці типу MyISAM, що підтримують повнотекстовий пошук, так і таблиці InnoDB, що підтримують транзакції на рівні окремих записів. Більш того, СУБД MySQL поставляється із спеціальним типом таблиць EXAMPLE, що демонструє принципи створення нових типів таблиць. Завдяки відкритій архітектурі і GPL-ліцензуванню, в СУБД MySQL постійно з'являються нові типи таблиць.

Спільнотою розробників MySQL створені різні відгалуження коду, такі як Drizzle, OurDelta, Percona Server і MariaDB. Всі ці відгалуження вже існували на момент поглинання компанії Sun корпорацією Oracle.

MySQL виникла як спроба застосувати mSQL до власних розробок компанії: таблицям, для яких використовувалися ISAM- підпрограми низького рівня. В результаті був вироблений новий SQL-інтерфейс, але API-інтерфейс залишився в спадок від mSQL.

2.4.4 Парсер

Парсер – це програма для автоматизації процесу парсинга, тобто обробки інформації за певним алгоритмом.

Парсинг – це процес синтаксичного і лексичного аналізу, розбору і перетворення будь-якого документа або вибору з цього документа, що цікавлять нас даних.

На ринку програмного забезпечення представлено кілька продуктів для парсинга контенту в інтернеті.

Парсер Datacol представляє собою універсальний інструмент для збору інформації в інтернеті. На даний момент цей парсер є безсумнівним лідерів в Рунеті за своїми можливостями і функціоналу.

Цей інструмент дозволяє збирати в інтернеті дані наступного типу:

– результати пошукової видачі;

– сбор контенту з заданих сайтів;

– сбор внутрішніх і зовнішніх посилань для інтернет сайту;

– сбор графічної інформації, аудіо контенту, відео матеріалів;

– парсінг СЕО показників сайтів з різних сервісів;

– і багато всілякої інформації з різних інтернет ресурсів.

Вся отримана інформація зберігається в зручному для користувача вигляді для подальшого аналізу і використання. І найголовніше програма вміє експортувати дані в базу даних MySQL.

Парсер Content Downloader являє собою універсальну програму. Володіє дуже широким функціоналом і являє собою цілий набір парсинг-функцій, ось перелік основних з них:

– парсер товарів;

– парсер інтернет-магазинів;

– парсер картинок;

– парсер відео;

– RSS парсер;

– парсер посилань;

– парсер новин.

І це ще неповний перелік всіх функцій. Програма платна, продається кілька варіантів, в залежності від функціоналу коливається і вартість.

Парсер контенту X-Parser, основні функції програми також складаються їх не

– скількох програмних блоків:

– парсер видачі будь-яких пошукових систем по ключових запитах;

– парсер контенту з будь-якого сайту;

– парсер контенту по ключових запитах з видачі пошукової системи;

– парсер контенту за списком URLов;

– парсер внутрішніх/ зовнішніх посилань.

Однак, недолік всіх програм парсер це вартість. Тому розглянемо бібліотеки.

Simple HTML DOM-PHP – бібліотека, що дозволяє парсить HTML-код за допомогою ручних jQuery-подібних селекторів.Бібліотека вміє працювати навіть з невалідним HTML-кодом, що значно спрощує роботу. Simple HTML DOM вміє шукати і фільтрувати вкладені елементи, звертатися до їх атрибутам і навіть вибирати окремі логічні елементи коду, наприклад, коментарі.

PHPQuery – бібліотека для парсинга, аналог jQuery на PHP Заснована на DOM, бібліотека є однією з найшвидших. PHPQuery працює з селекторами, атрибутами, Ajax подіями.

В результаті дослідженні, було вирішено, що найкраще користуватися бібліотекою phpQuery[10,11,12,13].

2.5 Опис хостингів

Хостинг провайдери забезпечують своїм користувачам ресурси, необхідні для розміщення інформації на сервері. В основну функцію хостингу входить збереження сайту на серверах і забезпечення відвідувачам постійної доступності до нього. Тому до вибору інтернет компанії, що надає дану послугу, слід підійти відповідально. Без якісного і надійного хостингу веб-сайт не буде загальнодоступний для користувачів. Робота сайту буде повільної, пошукові системи не зможуть його індексувати.

Кожен провайдер намагається виділитися серед своїх конкурентів і зробити гучну рекламу про те, що саме їх послуга є найбільш якісною, а хостинг найшвидшим. Але на практиці більшість організацій, що працюють в комунікаційної та інформаційної сфері, приписують собі неіснуючі гідності. І тут головне − не припуститися помилки. Клієнт завжди повинен бути впевнений в тому, що він вкладає кошти, а за це йому надаються відмінна і безперебійна робота хостингу. Саме тому розробки сайту для успішного ведення бізнесу не можуть бути засновані на безкоштовні послуги.

У нинішній час існує безліч провайдерів. Вони можуть бути вітчизняними або закордонними. Але приналежність таких організацій не впливає на їх рейтинг. Все залежить від того, наскільки серйозно інтернет компанія підходить до обслуговування клієнтів, яке обладнання використовує і скільки років працює в сфері інформаційного ринку.

Кожен хостинг-провайдер пропонує свої тарифні плани, за якими можна підібрати пакет послуг. Незмінно основна умова − надання дискового про-

стору. Правильний вибір провайдера дозволить сайту працювати стабільно, а завдяки гарному обладнанню можна досягти відмінних результатів. При цьому повинна забезпечуватися якісна і оперативна технічна підтримка. Компаніями також можуть надаватися додаткові услугі.Основние клієнти хостинг-провайдерів України − українські фірми. Наш рейтингу хостингу може у виборі. Сервери деяких компаній досить потужні, щоб забезпечувати високу якість роботи. Завдяки великій конкуренції можна знайти хорошу якість за доступною ціною. Хостинг-провайдерами України також реєструються доменні імена.Сейчас часто зустрічається рейтинг хостинг-провайдерів, який вибудовується за популярністю хостингів, ціною або якістю послуг.

Опис хостингу HOstiq. Як стандартна платформи для наших Shared, VIP і Reseller серверів ми використовуємо 1U сервери Supermicro з 6-ядерними процес-сорамі Intel Xeon E5 останнього покоління, 16 ГБ оперативної пам'яті і чотирма жорсткими дисками в RAID-10 масиві на основі RAID контролера LSI MegaRAID SAS 9266 з 1 ГБ кеш-пам'яті на борту. Загальний річний uptime не нижче 99.5%.

В якості платформи для VPS-носіїв використовуються 2U сервери Supermicro з двома 6-ядерними процесорами Intel Xeon E5 останнього покоління, 64-128 ГБ оперативної пам'яті і вісьмома жорсткими дисками в RAID-10 масиві на основі RAID контролера LSI MegaRAID SAS 9265 з 1 ГБ кеш -пам'яті на борту.

Незалежно від того, наскільки надійна система зберігання даних, завжди є ризик втрати даних в результаті аварії або людської помилки. Крім надлишкової надійності, забезпечується RAID-масивами 10-го рівня, в обов'язковому порядку роблять резервні копії всіх даних, розташованих на Shared, VIP і Reseller серверах на окреме сховище, розташоване в іншому приміщенні, тим самим значно знижуючи ризик втрати даних в результаті виходу з ладу дискової підсистеми.

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ

3.1 Фізична архітектура

Для реалізації взаємодії клієнта з сервером була використана триланкова архітектура (рис.3.1).

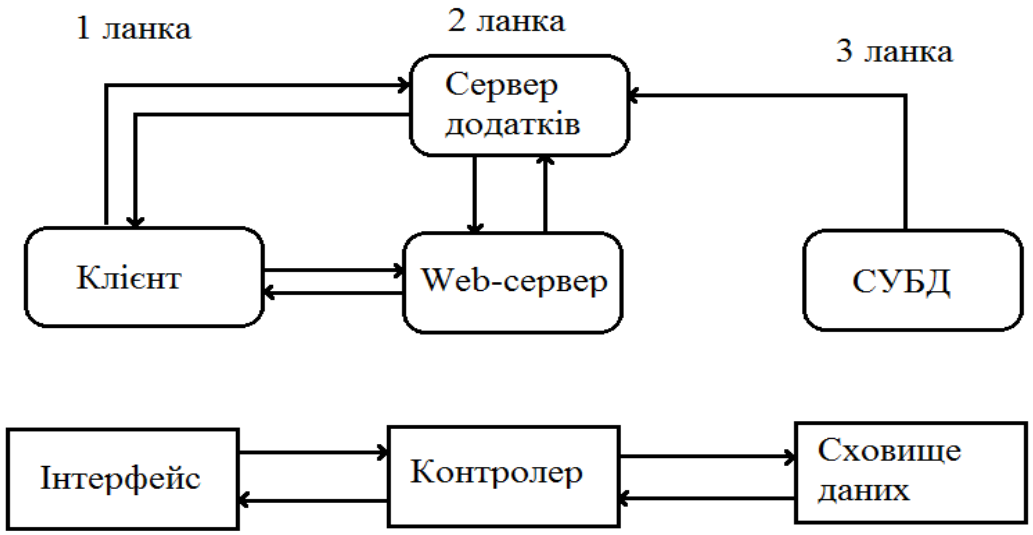


Рисунок 3.1 – Архітектура системи

Програмні засоби реалізації вміщують.

– PHP в якості мови сценаріїв на стороні сервера.

– MySQL в якості сервера управління базою даних.

– HTML, CSS, JavaScript в якості клієнтських технологій розмітки та динамічного управління гіпертекстом.

– Технологія Ajax як інтеграція JavaScript, HTML, CSS і PHP використовується для реалізації асинхронних користувацьких інтерфейсів, що застосовувалося серед іншого для реалізації понятійно-тезисного редактора, який передбачає інтенсивну взаємодію користувача з системою у процесі редагування даних.

– Інфраструктурне програмне забезпечення на стороні сервера – Web-сервер, як правило, у ролі якого виступає Apache під управлінням Unix-подібної операційної системи, що застосовується на серверах хостингу.

На стороні клієнта достатньо сучасного Web-браузера не залежно від ОС клієнтського комп’ютера.

Фізично вся бізнес-логіка та функції роботи з базою даних знаходяться на сервері додатків хостинг-провайдера, в той час як база даних знаходиться на виділеному сервері хостинг-провайдера. Користувачі отримують доступ до системи через мережу Інтернет, використовуючи веб-браузери[14,15,16].

Фізична архітектура системи зображена на рис. 3.2.

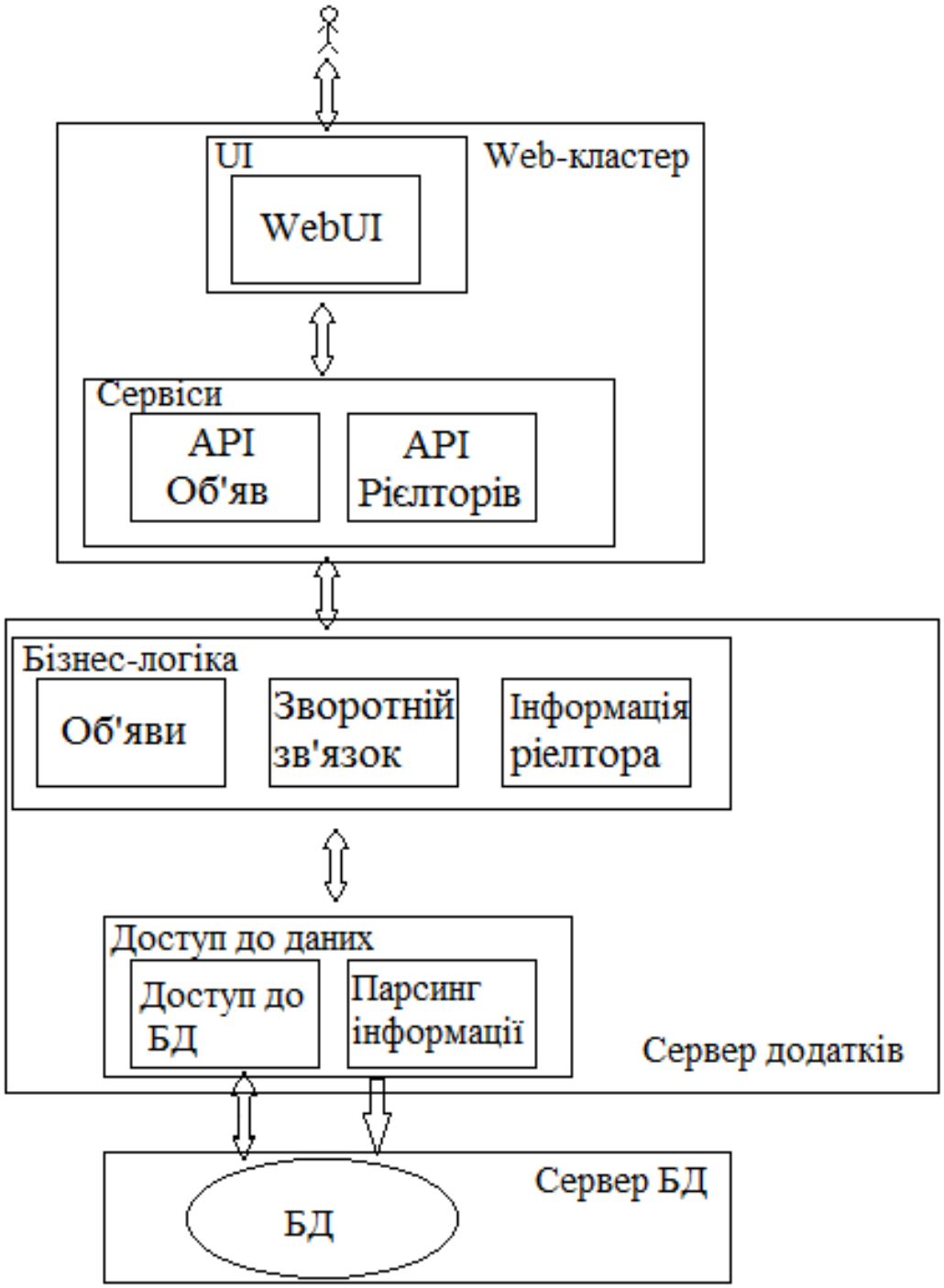


Рисунок 3.2 – Фізична архітектура системи

3.2 Проектування і розробка бази даних системи

Компонент керування даними виконує функцію зберігання інформації, яку необхідно використовувати для web-системи. Ця інформація надходить з інших джерел (рис. 3.3): з дошки оголошень OLX, з сайтів нерухомості ugo-zapad та president.

Рисунок 3.3 – Парсинг даних з інших сайтів

3.2.1 Виділення інформаційних об'єктів

Інформаційний об'єкт – це опис некої сутності у вигляді сукупності логічно зв'язаних інформаційних елементів. Інформаційний об'єкт має безліч реалізацій – екземплярів, кожний з котрих представлених сукупністю конкретних значень інформаційних елементів та ідентифікуються значенням ключа.

Почнемо розгляд інформаційних об'єктів з «information property».

Тут вказана вся інформація о нерухомості. Для того, щоб знайти необхідну нам нерухомість необхідні такі об’єкти як:

– «deal type» – в ньому міститься інформація по типу угоди;

– «rent type» – в ньому міститься інформація по типу аренди;

– «property type» – в ньому міститься тип нерухомості;

– «commerce type» – в ньому міститься тип коммерційної нерухомості;

– «region» – в ньому міститься назва району, де знаходиться нерухомість.

Для того, щоб домовитися переглянути нерухоміть нам необхідно знати інформацію об об’єкті «Realtor», який містить інформацію об ріелторі котрий займається цією нерухомістю.

3.2.2 Визначення атрибутів об'єктів

Атрибут об’єкта – це значення характеризуюче об’єкт в його класі. Серез атрибутів поділяють постоянні та перемінні. Перемінні атрибути характеризують об’єкт в його класі. Перемінні атрибути характеризують поточний стан об’єкта.Змінюючи значення цих атрибутій ми змінюємо стан об’єкта. Об’єкт – абстракція або будьяка річ з чітко окресленими гранями.

Визначимо атрибути для кожного представленого вище об’єкта (табл.3.1).

Таблиця 3.1 – Атрибути об'єктів

Об'єкт	Атрибути об'єктів	Ключевий атрибут
deal type	Id, type	id
rent type	Id, id_deal, type	id
property type	Id, id_rent, type	id
commerce type	Id, type, id_property	id
region	Id, name, id_property	id
information property	Id, id_region, microdistrict, address, price, space, rooms, floor, image, id_property, id_realtor	id
realtor	Id, name, surname, secondname, telephone, image	id

Визначення відносин між об’єктами.

Розглянемо відносини між об’єктами системи.

Deal type – rent type. Deal type – головний об’єкт, rent type – підлеглий об’єкт. Тип зв’язку між об’єктами – «один-до-багатьох», так як у аренди є кілька типів. Зв’язок між об’єктами здвйснює атрибут «id_deal», додаваємий в «rent type» в якості зовнішнього ключа.

Property type – rent type. Rent type – головний об’єкт, property type – підлеглий об’єкт. Тип зв’язку між об’єктами «один-до-багатьох», так як у нерухомості є багато типів, зв’язок між цими об’єктами здійснює атрибут «id_rent», додаваємий в property type у якості зовнішнього ключа.

Commerce type – property type. Property type – головний об’єкт, commerce type – підлеглий об’єкт. Тип зв’язку між об’єктами – «один-до-багатьох», так як у комерційної нерухомості є кілька типів. Зв’язок між цими об’єктами здійснює атрибут «id_property», додаваємий в commerce type в якості зовнішнього ключа.

Region – property type. Property type – головний об’єкт, region – підлеглий об’єкт. Тип зв’язку між об’єктами «один-до-багатьох» так як багато нерухомості може знаходитися в одному районі. Зв’язок між цими об’єктами здійснює атрибут «id_property», додаваємий в region в якості зовнішнього ключа.

Information property – region. Information property – головний об’єкт, region – підлеглий об’єкт. Тип зв’язку між об’єктами – «один-до-одного», так як нерухомість знаходиться в одному районі. Зв’язок між цими об’єктами здійснює атрибут «id_region», додаваємий в information property в якості зовнішнього ключа.

Information property – realtor. Information property – головний об’єкт, realtor – підлеглий об’єкт. Тип зв’яку між об’єктами – «один-до-багатьох», так як один ріелтор займається багатьма об’єктами нерухомості. Зв’язок між цими об’єктами здійснює атрибут «id_realtor», додаваємий в information property в якості зовнішнього ключа.

3.2.3 Побудова концептуальної схеми

На основі отриманих об’єктів, атрибутів об’єктів та відносин між ними будується концептуальна модель у вігляді ER-діаграми (рис.3.4).

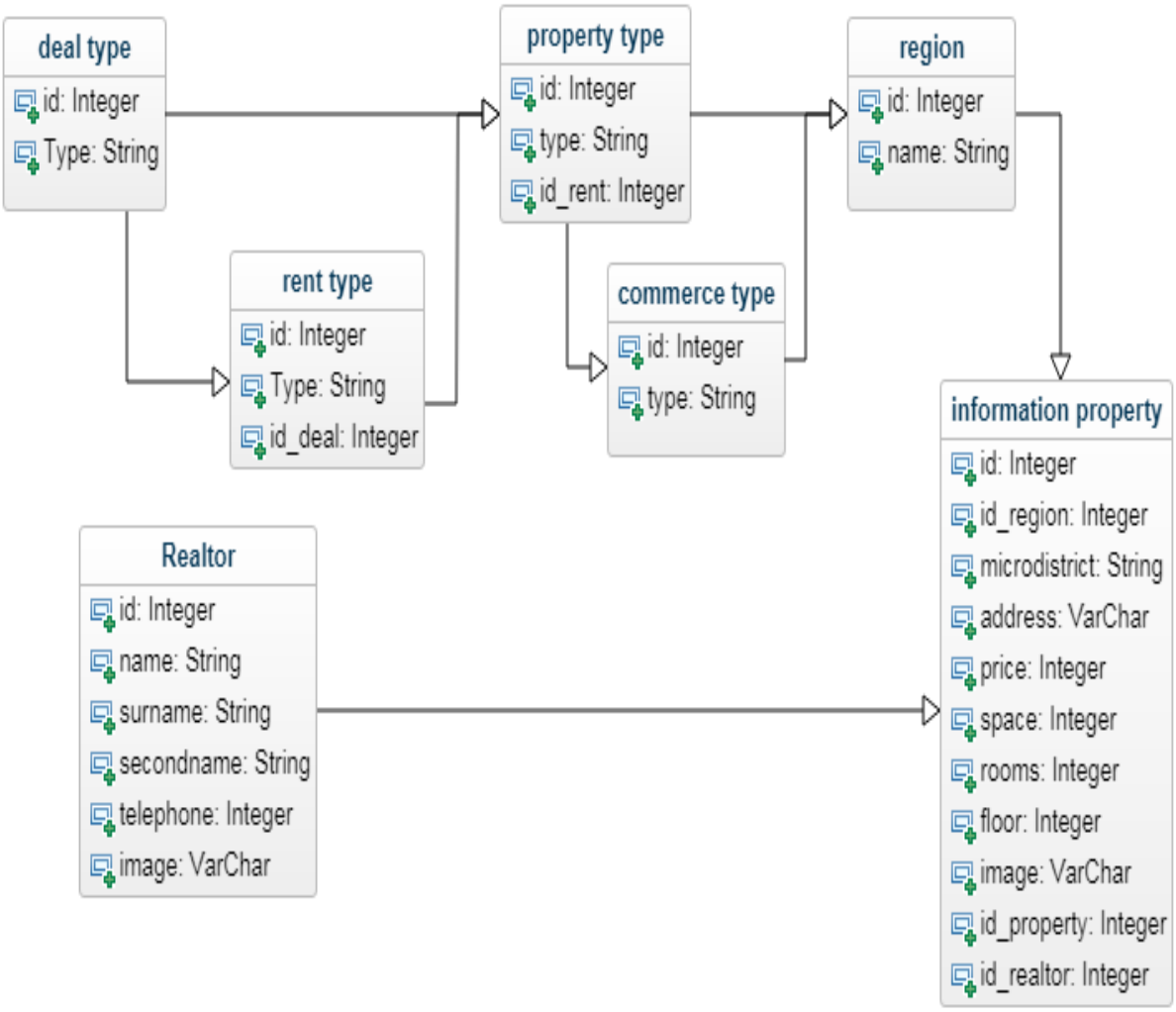


Рисунок 3.4 – Концептуальна схема бази даних системи

Діаграма взаємозв’язків сутностей використовується на концептуальному рівні проектування бази даних. На основі усіх сутностей та зв’язків між ними була створена база даних.

База даних міститиме такі сутності.

Сутність deal type міститиме унікальний ідентифікатор та тип угоди.

Сутність rent type міститиме унікальний ідентифікатор, тип аренди та посилання на тип угоди.

Сутність property type міститиме унікальний ідентифікатор, тип нерухомості та посилання на тип угоди.

Сутність commerce type міститиме унікальний ідентифікатор та тип комерційної нерухомості.

Сутність region міститиме унікальний ідентифікатор та назву району.

Сутність Realtor міститиме унікальний ідентифікатор, ім’я, прізвище, ім’я по батькові, телефон та фотографію якщо є.

Сутність information property міститиме унікальний ідентифікатор, посилання на район, посилання на тип нерухомості, посілання на ріелтора, назву мікрорайону, адрессу, ціну, площу, кількість кімнат, етажність та фотографії.

3.2.4 Особливості реалізації

До особливостей реалізації можна віднести:

– бібліотеку curlQuery;

– бібліотеку phpQuery.

Бібліотека curlQuery – це бібліотека для створення запитів схожих на реального клієнта, вона дозволяє коду робити багато запитів без ризику бути заблокованим сайтом.

Бібліотека phpQuery – бібліотека для парсинга, аналог jQuery на PHP Автор– польський програміст Тобіас Судник. Спочатку розробив невеликий парсер для своїх потреб – це був збирач рецензій на фільми з торрент-треккеров. Нескладний пошуковий робот з часом перетворився в одне з найкращих засобів синтаксичного аналізу.

Заснована на DOM, бібліотека є однією з найшвидших. phpQuery працює з селекторами, атрибутами, Ajax-му, подіями тощо. Також є і JavaScript-версія – jQueryServer.

Хоча автор є чинним програмістом, останнє оновлення бібліотеки датується травнем 2009 року.

Парсинг сайту за допомогою phpQuery на PHP виконується наступним чином.Насамперед потрібно отримати html зміст потрібної сторінки сайту для парсинга. Для цього необхідно використовувати функцію file_get_content, вказавши url сторінки в якості параметра(рис.3.5).

```
<?php
$page = file_get_contents('http://www.example.com/?param=1');
echo $page;
?>
```

Рисунок 3.5 – Функція file_get_content

Наступний етап це отримання вмісту web сайту і вибір необхідної інформації (рис.3.6)

```
$document = phpQuery::newDocument($content);
 $list_elements = $document->find('.list1, .list2');
foreach ($list_elements as $element)
{
   $pq = pq($element);
   $title_news = $pq->find('h3')->text();
   echo '<br><b>Заголовок новости:</b> '.$title_news.'<br>';
   $pq->find('.gray, .sub, h3')->remove();
   echo $pq->text();
}
```

Рисунок 3.6 – Вибір інформації

3.3 Реалізація бази даних

Для того, щоб створити базу даних необхідно виконати наступний sql запит (рис.3.7,3.8):

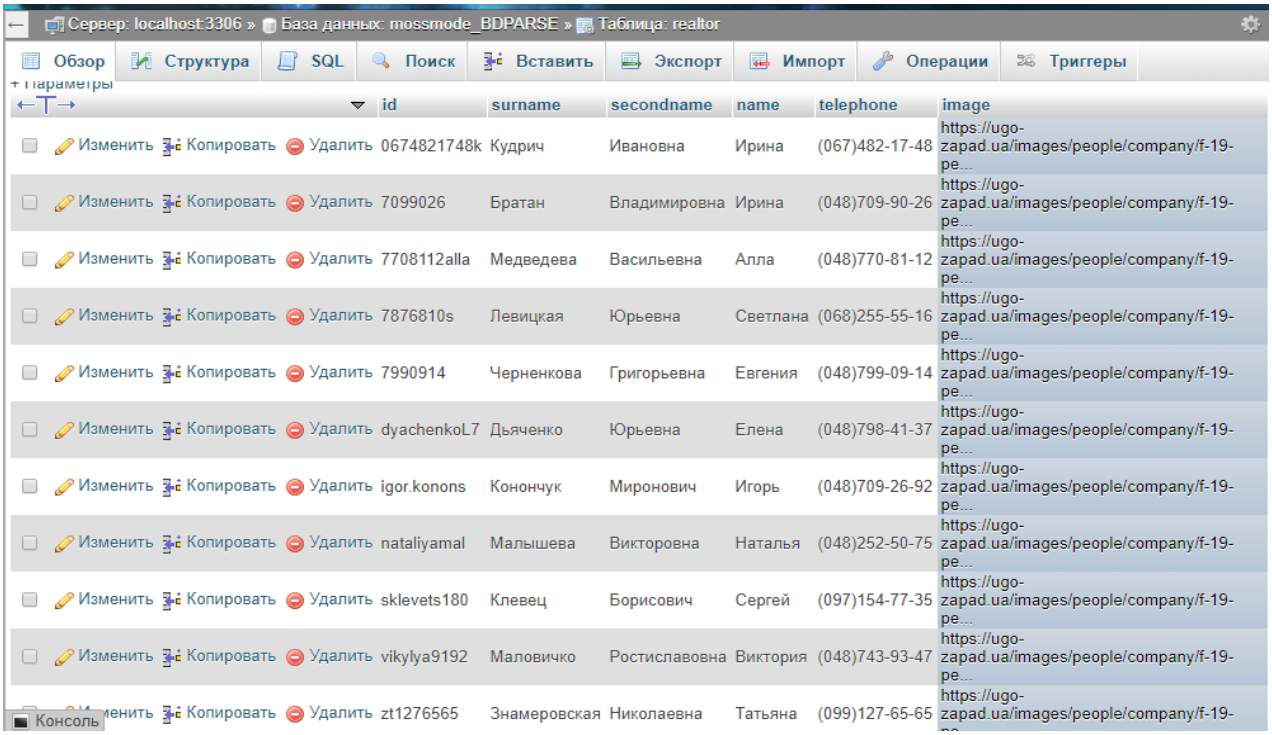
```
CREATE DATABASE "mossmode_BDPARSE"
WITH OWNER = adminservice
ENCODING = 'UTF8'
```

В результаті виконання запиту створиться база даних Service Center. Для того, щоб створити таблицю, первинний ключ і додати зв'язок необхідно виконати наступний запит:

```
CREATE TABLE login.UserPassword
(
Login TEXT NOT NULL PRIMARY KEY,
Password TEXT NOT NULL,
Role INT NOT NULL REFERENCES login.UserRole (ID),
UNIQUE (Login, Role)
);
```

Для заповнення бази даних необхідно використовувати запити наступного виду:

INSERT INTO login.UserPassword (Login, Password, Role)
VALUES (LoginP, PasswordP, 3);

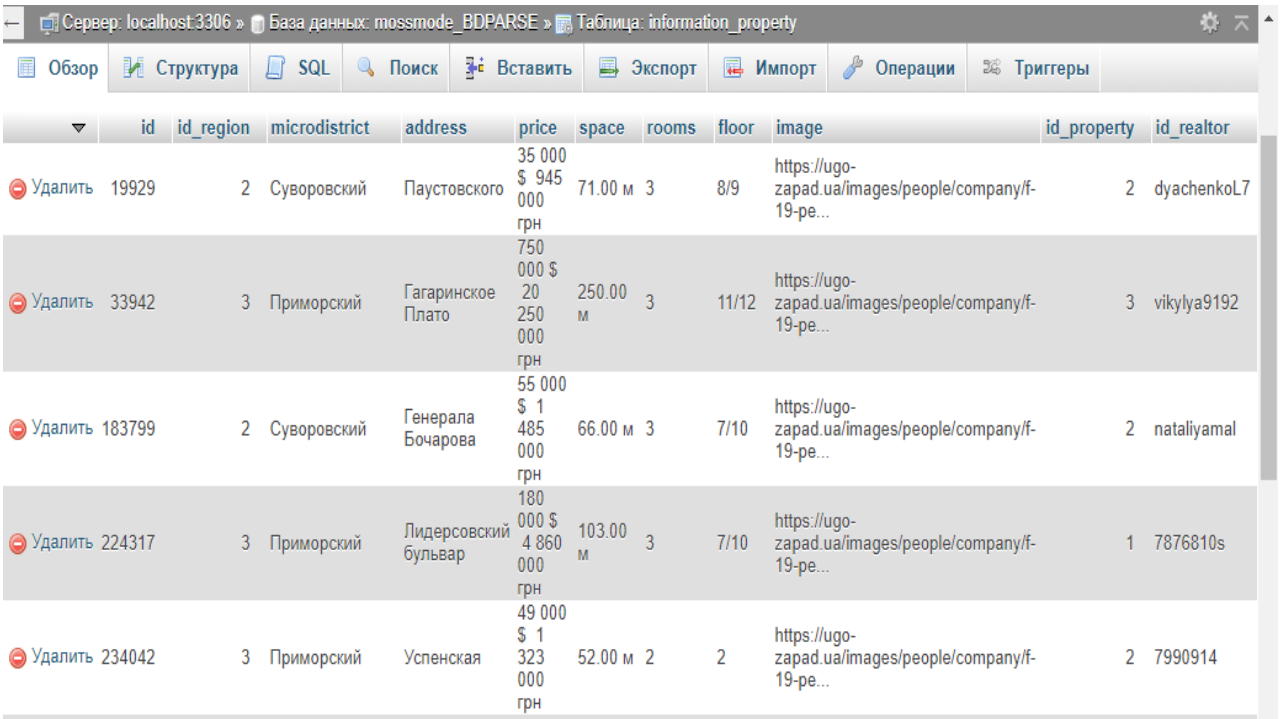

Сервер: localhost:3306 » База данных: mossmode_BDPARSE » Таблица: realtor

Обзор | Структура | SQL | Поиск | Вставить | Экспорт | Импорт | Операции | Триггеры

	id	surname	secondname	name	telephone	image
Изменить Копировать Удалить	0674821748k	Кудрич	Ивановна	Ирина	(067)482-17-48	https://ugo-zapad.ua/images/people/company/f-19-pe...
Изменить Копировать Удалить	7099026	Братан	Владимировна	Ирина	(048)709-90-26	https://ugo-zapad.ua/images/people/company/f-19-pe...
Изменить Копировать Удалить	7708112alla	Медведева	Васильевна	Алла	(048)770-81-12	https://ugo-zapad.ua/images/people/company/f-19-pe...
Изменить Копировать Удалить	7876810s	Левицкая	Юрьевна	Светлана	(068)255-55-16	https://ugo-zapad.ua/images/people/company/f-19-pe...
Изменить Копировать Удалить	7990914	Черненкова	Григорьевна	Евгения	(048)799-09-14	https://ugo-zapad.ua/images/people/company/f-19-pe...
Изменить Копировать Удалить	dyachenkoL7	Дьяченко	Юрьевна	Елена	(048)798-41-37	https://ugo-zapad.ua/images/people/company/f-19-pe...
Изменить Копировать Удалить	igor.konons	Кончук	Миронович	Игорь	(048)709-26-92	https://ugo-zapad.ua/images/people/company/f-19-pe...
Изменить Копировать Удалить	nataliyamal	Малышева	Викторовна	Наталья	(048)252-50-75	https://ugo-zapad.ua/images/people/company/f-19-pe...
Изменить Копировать Удалить	sklevets180	Клевец	Борисович	Сергей	(097)154-77-35	https://ugo-zapad.ua/images/people/company/f-19-pe...
Изменить Копировать Удалить	vikylya9192	Маловичко	Ростиславовна	Виктория	(048)743-93-47	https://ugo-zapad.ua/images/people/company/f-19-pe...
Изменить Копировать Удалить	zt1276565	Знаменовская	Николаевна	Татьяна	(099)127-65-65	https://ugo-zapad.ua/images/people/company/f-19-pe...

Консоль

Рисунок 3.7 – Таблиця ріелторів


Сервер: localhost:3306 » База данных: mossmode_BDPARSE » Таблица: information_property

Обзор | Структура | SQL | Поиск | Вставить | Экспорт | Импорт | Операции | Триггеры

	id	id_region	microdistrict	address	price	space	rooms	floor	image	id_property	id_realtor
Удалить	19929	2	Суворовский	Паустовского	35 000 $ 945 000 грн	71.00 м	3	8/9	https://ugo-zapad.ua/images/people/company/f-19-pe...	2	dyachenkoL7
Удалить	33942	3	Приморский	Гагаринское Плато	750 000 $ 20 250 000 грн	250.00 м	3	11/12	https://ugo-zapad.ua/images/people/company/f-19-pe...	3	vikylya9192
Удалить	183799	2	Суворовский	Генерала Бочарова	55 000 $ 1 485 000 грн	66.00 м	3	7/10	https://ugo-zapad.ua/images/people/company/f-19-pe...	2	nataliyamal
Удалить	224317	3	Приморский	Лидерсовский бульвар	180 000 $ 4 860 000 грн	103.00 м	3	7/10	https://ugo-zapad.ua/images/people/company/f-19-pe...	1	7876810s
Удалить	234042	3	Приморский	Успенская	49 000 $ 1 323 000 грн	52.00 м	2	2	https://ugo-zapad.ua/images/people/company/f-19-pe...	2	7990914

Рисунок 3.8 – Таблиця нерухомості

3.4 Взаємодія з базою даних

Звернення до бази даних відбувається через запит по адресу, виду нерухомості, району городу, кількісті кімнат, обмеження по площі (рис.3.9)

```
$query=mysql_query("INSERT INTO information_property
VALUES('$id','$id_region','$microdistrict','$address','
$price','$space','$rooms','$floor','$image','$id_property','$id_realtor')")or die
(mysql_error());
```

Рисунок 3.9 – Запит до бази даних про нерухомість

Також звернення до бази даних відбувається через запит за даними ріелтора (рис.3.10).

```
$query=mysql_query("INSERT INTO realtor
VALUES('$id_realtor','$surname','$secondname','$name','$telephone','$imager')")
or die (mysql_error());
```

Рисунок 3.10 – Запит до бази даних за даними ріелтора

На сайті ріелтори можуть реєструватися. Всі дані заносяться в базу даних. Паролі зберігаються в вигляді хеш-образів (рис.3.11).

```
   $query=mysql_query("SELECT * FROM registered");
   while($row = mysql_fetch_array($query)){
   if($row['login']==$login){
   die ('Такой пользователь уже существует!');    } }
   if($password==$r_password){   $password=md5($password);
      $query=mysql_query("INSERT INTO registered
VALUES('','$name','$second_name','$email', '$telephone','$password','')")or die
(mysql_error());    }
   else{       die('Пароли не совпадают, проверьте правильность ввода!');    }
```

Рисунок 3.11 – Запит до бази даних для аутентифікації ріелтора

3.5 Інформація по експлуатації web-системи

Мета роботи є надання користувачам повну та актуальну інформацію про нерухомість в місті Одеса на web-системі. Тому на сайті реалізовані наступні функції: реєстрація; пошук необхідної інформації;зворотній зв’язок.

Реєстрація – функція для ріелторів, робота функції реєстрації починається з форми у яку необхідно ввести дані користувача:

– ім’я;
– прізвище;
– по батькові;
– номер телефону;
– email;
– фото.

Алгоритм роботи цієї функції представленій на рис. 3.11.

Після того як інформація введена треба натиснути кнопку "реєстрація", за наявності натискання на кнопку, функція перевіряє всі дані, введені чи ні, якщо ні – функція покаже що необхідно заповнити усі дані, якщо всі данні введені – вона починає перевірку, існує такий користувач чи ні.

Функція підключається до бази даних та починає пошук, якщо такий користувач існує в базі даних – вона виводить інформацію що такий користувач вже зареєстрований, якщо користувач не зареєстрований – функція заносить його до бази даних, після чого робота завершується

Доступ до сайту можливий усім, кожен з клієнтів може зайти на сайт та знайти необхіжну йому інформацію за допомогою функції пошуку.

Доступ до сайту можливий усім, кожен з клієнтів може зайти на сайт та знайти необхіжну йому інформацію за допомогою функції пошуку.

Функція підключається до бази даних та починає пошук, якщо такий користувач існує в базі даних – вона виводить інформацію що такий користувач вже зареєстрований, якщо користувач не зареєстрований – функція заносить його до бази даних, після чого робота завершується.

Пошук інформації здійснюється завдяки зручному та інтуітивно зрозумілому пошуку який складається з таких полів.

Зворотній зв’язок це функція для зв’язку з адміністратором для того щоб повідомити про якусь помилку чи запропонувати якось покращити систему.

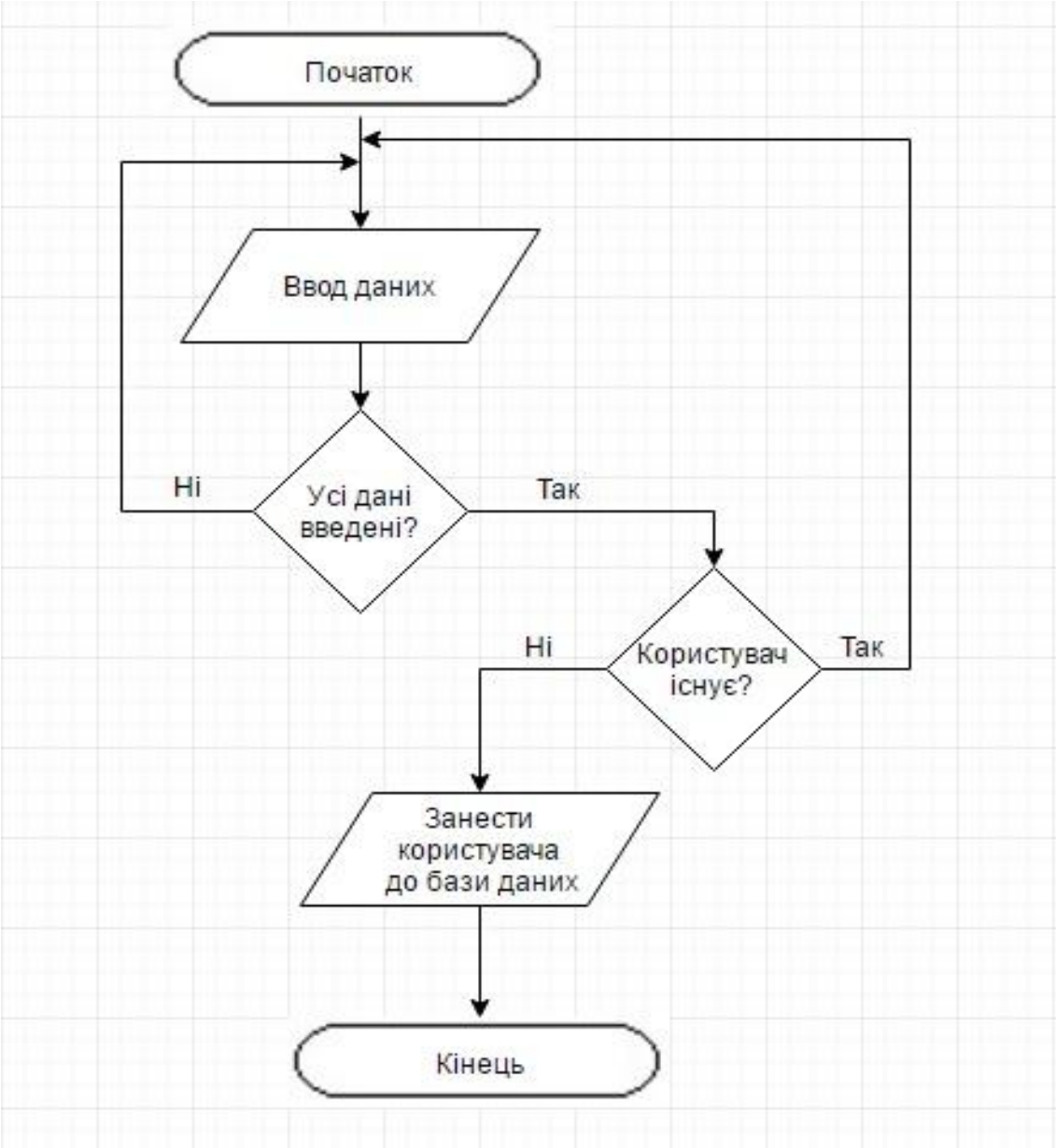


Рисунок 3.11 – Алгоритм роботи реєстрації

4 ОЦІНЮВАННЯ СИСТЕМИ

В магістерській роботі була проведена оцінка швидкості роботи системи, оцінки продуктивності веб додатків, перевірка навантаження сайту.

4.1 Оцінка системи на швидкість роботи

Gtmetrix.com – це онлайн сервіс, який надає можливість оцінити швидкість роботи веб-сайту, знайти його переваги і недоліки.

Результати тестування розробленої системи представлені на рис.4.1.

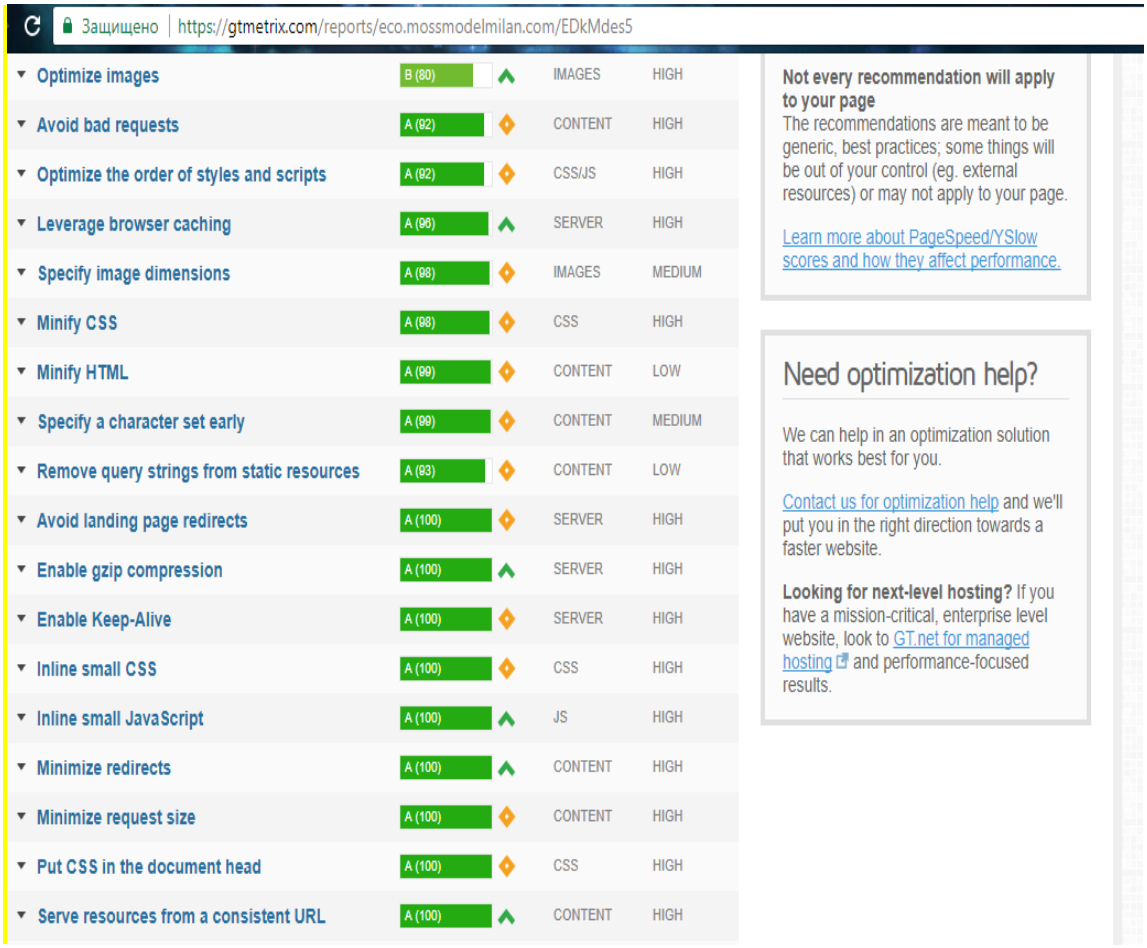


Рисунок 4.1 – Тестування системою Gtmetrix.com

Lighthouse – це розширення для браузера Google Chrome, створене компанією Google для оцінки продуктивності веб додатків. Результати тестування за допомогою розширення Lighthouse представлені на рис.4.2.

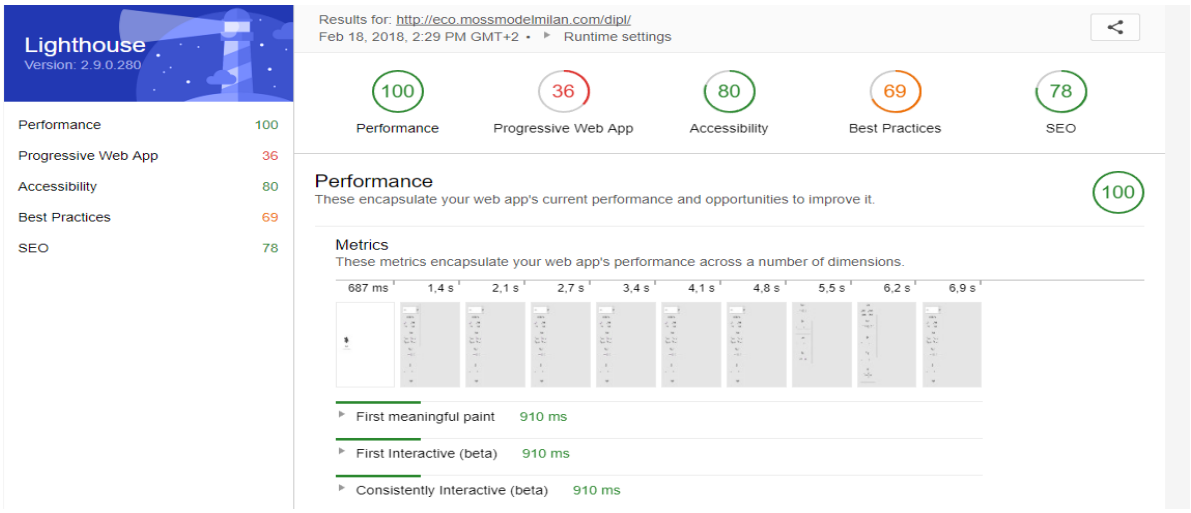


Рисунок 4.2 – Тестування додатком Lighthouse

У розділі Performance була порада позбавитися від блокуючих рендер таблиць стилів.

У розділі Accessibility була порада збільшити контрастність тексту для більшої читабельності, в розділі Best Practices недоліком є відсутність короткого імені у додатку.

Loaddy.com – це інструмент навантажувального тестування, який дозволяє перевірити навантаження, яке витримує Ваш хостинг або сервер, а також показується, як зміниться швидкість завантаження всього сайту або окремих сторінок. Результати тестування зображені на рис 4.3.

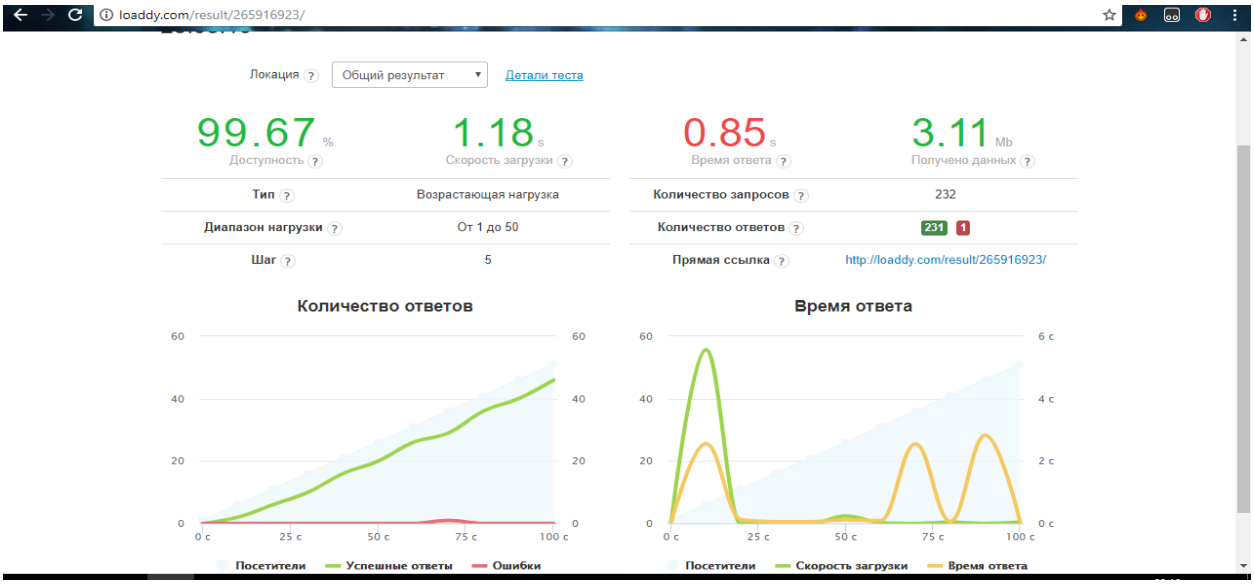


Рисунок 4.3 – Тестування за допомогою loaddy.com

ВИСНОВКИ

У ході виконання комплексної дипломної роботи було спроектовано та створенно автоматизована WEB-система збору даних про нерухомість м.Одеси з інших web-сайтів та відображання їх.

Були досягнені наступні задачі:

– спроектована архитектури WEB-системи;

– спроектована та реалізована база даних об’єктів нерухомості, продаваних та здаваємих в аренду: квартир, житлових домів, комерційної нерухомості, земельних часток;

– реалізована функція зберігання для любого об’єкта нерухомості довільного набору характеристик (площа, поверх, обробка, стадія будування та т.п.);

– розроблений парсер для збору даних о нерухомості з Web-сайтів;

– розроблена функція автоматизованого оновлення бази даних;

– розроблена функція пошуку необхідного об’єкту нерухомості за любими заданими характеристиками: вид нерухомості, район городу, кількість кімнат, обмеження по площі, ціні та т.п.;

– розроблена функція видалення не актуальної інформації;

– розроблена функція реєстрації, розділення прав доступу і авторизація користувачів;

– розроблена функція печаті вихідних документів: список об’єктів нерухомості, відібраних по заданим крітеріям.;

– розроблений адаптивний інтерфейс та кросбраузерність;

– обран хостинг.

Отримана система має досить високі показники швидкості і отримала високі оцінки в системах, що спеціалізуються на оцінці якості веб-додатків.

ПЕРЕЛІК ПОСИЛАНЬ

1. The History of Groupon [Електронний ресурс]. – Режим доступу: https://www.groupon.com/merchant/article/the-history-of-groupon.
2. RetailMeNot, Inc – About [Електронний ресурс]. – Режим доступу: https://www.retailmenot.com/corp/about/.
3. Digital Coupons | Quotient Technology Inc. [Електронний ресурс]. – Режим доступу: https://www.quotient.com/digital-coupons/.
4. CupoNation Group [Електронний ресурс]. – Режим доступу: https://www.cupo nation.com/cuponation-group-becomes-global-savings-group/.
5. ShopAtHome [Електронний ресурс]. – Режим доступу: http://www.bioportfolio.com/corporate/company/12639/ShopAtHome-com.html.
6. Angular vs Knockout vs Ember [Електронний ресурс]. – Режим доступу: https://jsperf.com/angular-vs-knockout-vs-ember/292. – 02.06.2017.
7. Хаген Граф. Создание веб-сайтов с помощью Joomla! 1.5. – СПб.: Издательский дом "Вильямс", 2009. – 304 с.
8. Ромашев В. CMS Drupal: Система управления содержимым сайта. – СПб.: Питер, 2010. – 256 с.
9. Прохоренок Н. HTML, JavaScript, PHP и MySQL. Джентельменский набор Web-мастера. – СПб.: БХВ-Петербург, 2009. – 880 с.
10. Кузнецов М. PHP. Практика создания Web-сайтов./ Кузнецов М., Симдянов И. – 2-е изд. перераб. и доп. – СПб.: БХВ-Петербург, 2008. – 1264 с.
11. Анатольев А.Г. Web-сервисы, как способ интеграции приложений в WWW. [Електронний ресурс] – Режим доступу: http://www.4stud.info /networking/web-service.html
12. Томасон Л. Разработка Web-приложений на PHP и MySQL: Пер. с анг./Л. Томсон. –Спб.: ООО ДиаСофтЮП, 2003. – 672 с.
13. Фримен Э. Изучаем HTML, XHTML и CSS – СПб.: «Питер», 2010. – 656с.
14. Титтел Э.HTML, XHTML и CSS для чайников, 7-е издание / Титтел Э., Ноубл Д. – М.: «Диалектика», 2011. – 400 с.
15. Парсинг html-сайтов с помощью PHP [Електронний ресурс] – Режим доступу: http://parsing.valemak.com/ru/php/library/phpquery/

16. Данилин А.Архитектура и стратегия. «Инь» и «Янь» информационных технологий предприятия./Данилин А., Слюсаренко А. – М.: Интернет-университет информационных технологий, 2005. – 504 с.
17. Фаулер М. Архитектура корпоративных программных приложений.: Пер. с англ. – М.: Издательский дом «Вильямс», 2006. – 544 с.

Д О Д А Т К И

ДОДАТОК А
КОД БІБЛІОТЕКИ CURLQUERY

```
<?php
function curl_get ($url, $referer = 'https://www.google.com.ua') {
   $ch = curl_init();
   curl_setopt($ch, CURLOPT_URL, $url);
   curl_setopt($ch, CURLOPT_HEADER, 0);
   curl_setopt($ch, CURLOPT_USERAGENT, "Mozilla/5.0 (Windows NT 6.1;
WOW64;ry:38.0) Gecko/20100101 Firefox/38.0");
   curl_setopt($ch, CURLOPT_REFERER, $referer);
   curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
   $data = curl_exec($ch);
   curl_close($ch);
   return $data;
}
?>
```

ДОДАТОК Б
КОД ФУНКЦІЇ РЕЄСТРАЦІЇ

```
<?php
$servername = "localhost";
$username = "mossmode_estzi";
$password = "#AzVVB[kkfvD";
$dbname = "mossmode_BDPARSE";
$connect=mysql_connect($servername,$username,$password)or
die(mysql_error());
mysql_select_db($dbname);
mysql_set_charset( 'utf8' );
if(isset($_POST['submit'])) {
    $name=$_POST['name'];
    $second_name=$_POST['second_name'];
    $email=$_POST['email'];
    $telephone=$_POST['telephone'];
    $password=$_POST['password'];
    $r_password=$_POST['r_password'];
    $query=mysql_query("SELECT * FROM registered");
   while($row = mysql_fetch_array($query)){
   if($row['email']==$email){
   die ('Такой пользователь уже существует!');
   } }
   if($password==$r_password){
      $password=md5($password);
      $query=mysql_query("INSERT INTO registered
VALUES('','$name','$second_name','$email', '$telephone','$password')")or die
(mysql_error());
   }
   else{
      die('Пароли не совпадают, проверьте правильность ввода!');
   } }
?>
<form method="post">
```

```
    <input type="text" name="name" placeholder=" | Имя" required/> <br>
    <input type="text" name="second_name" placeholder=" | Фамилия" required/>
<br>
    <input type="text" name="email" placeholder=" | Email" required/> <br>
    <input type="text" name="telephone" placeholder=" | Телефон" required/>
<br>
    <input type="text" name="password" placeholder=" | Пароль" required/> <br>
    <input type="text" name="r_password" placeholder=" | Повторите пароль"
required/> <br>
    <input type="submit" name="submit" value="ADD"/>
</form>
```

ДОДАТОК В
КОД ФУНКЦІЇ ПАРСИНГУ З САЙТУ UGO-ZAPAD.UA

```
<?php
require 'phpQuery.php';
require 'curlQuery.php';
$url='https://ugo-zapad.ua/kvartira';
$servername = "localhost";
$username = "mossmode_estzi";
$password = "#AzVVB[kkfvD";
$dbname = "mossmode_BDPARSE";
$conn = new mysqli($servername, $username, $password, $dbname);
if ($conn->connect_error) {
   die("Connection failed: " . $conn->connect_error);
}
function parse_main ($url) {
   $file = curl_get($url);
   $doc = phpQuery::newDocument($file);
   $doc1 = $doc->find('.category-view .price');
   foreach ($doc1->find('a') as $article){
      $article = pq($article)->attr('href');
      $file1 = curl_get($article);
      $docum = phpQuery::newDocument($file1);
      $image = $docum->find('.project-left .flexslider');
      $address = $docum->find('.project-address');
      $floor = $docum->find('.project-left .tabl2');
      $space = $docum->find('.project-left .tabl3');
      $price = $docum->find('.project-left .project-price-inner-wrapper');
      $iameger = $docum->find('.realtor .image');
      $id = $docum->find('.project-left .vendor');
      $id_region = $docum->find('.project-left .region');
      $microdistrict = $docum->find('.project-left .destrict');
      $rooms = $docum->find('.project-left .rooms');
      $id_property = $docum->find('.project-left .realty');
      $id_realtor = $docum->find('.realtor .email');
```

```
        $surname = $docum->find('.realtor .surname');
        $secondname = $docum->find('.realtor .patronymic');
        $name = $docum->find('.realtor .name');
        $telephone = $docum->find('.realtor .contact');
        $query=mysql_query("INSERT INTO information_property
VALUES('$id','$id_region','$microdistrict','$address','
$price','$space','$rooms','$floor','$image','$id_property','$id_realtor')")or die
(mysql_error());
        $query=mysql_query("INSERT INTO realtor
VALUES('$id_realtor','$surname','$secondname','$name','$telephone','$imager')")o
r die (mysql_error());
    }}
    parse_main($url);
?>
```