

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

МАТЕМАТИЧНИЙ ФАКУЛЬТЕТ

Кафедра комп'ютерних наук

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему: «РОЗРОБКА СИСТЕМИ ЗБОРУ ДАНИХ
ІНТЕРНЕТ МАГАЗИНІВ З ВИКОРИСТАННЯМ
ФРЕЙМВОРКУ LARAVEL»

Виконав: студент 2 курсу, групи 8.1222
спеціальності 122 Комп'ютерні науки
(шифр і назва спеціальності)
освітньої програми Комп'ютерні науки
(назва освітньої програми)

М.Л. Кольцов
(ініціали та прізвище)
доцент кафедри комп'ютерних наук,
доцент, к.т.н. Решевська К.С.
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Рецензент завідувач кафедри програмної інженерії, доцент, к.ф.-м.н.
Лісняк А.О.
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Запоріжжя

2023

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет математичний
Кафедра комп'ютерних наук
Рівень вищої освіти магістр
Спеціальність 122 Комп'ютерні науки
(шифр і назва)
Освітня програма Комп'ютерні науки

ЗАТВЕРДЖУЮ
Завідувач кафедри комп'ютерних наук,
д.т.н., професор

_____ Шило Г.М.
(підпис)

“ 05 ” травня _____ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТОВІ

Кольцову Марку Лоллійовичу
(прізвище, ім'я та по-батькові)

1. Тема роботи Розробка системи збору даних інтернет магазинів з використанням фреймворку Laravel

керівник роботи Решевська Катерина Сергіївна, к.т.н., доцент
(прізвище, ім'я та по-батькові, науковий ступінь, вчене звання)

затверджені наказом ЗНУ від « 01 » травня 2023 року № 642-с

2. Строк подання студентом роботи 29.11.2023

3. Вихідні дані до роботи 1. Постановка задачі.
2. Перелік літератури.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
1. Постановка задачі.
2. Основні теоретичні відомості.
3. Розробка системи збору даних.

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

6. Консультанти розділів роботи

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|--------|---|----------------|------------------|
| | | завдання видав | завдання прийняв |
| | | | |
| | | | |
| | | | |

7. Дата видачі завдання 05.05.2023

КАЛЕНДАРНИЙ ПЛАН

| № | Назва етапів кваліфікаційної роботи | Строк виконання етапів роботи | Примітка |
|----|--|-------------------------------|----------|
| 1. | Розробка плану роботи. | 04.06.2023 | |
| 2. | Збір вихідних даних. | 20.06.2023 | |
| 3. | Обробка методичних та теоретичних джерел. | 20.07.2023 | |
| 4. | Розробка першого та другого розділу. | 14.08.2023 | |
| 5. | Розробка третього розділу. | 01.09.2023 | |
| 6. | Оформлення та нормоконтроль кваліфікаційної роботи магістра. | | |
| 7. | Захист кваліфікаційної роботи. | | |

Студент _____
(підпис)

М.Л. Кольцов _____
(ініціали та прізвище)

Керівник роботи _____
(підпис)

К.С. Решевська _____
(ініціали та прізвище)

Нормоконтроль пройдено

Нормоконтролер _____
(підпис)

О.Г. Спиця _____
(ініціали та прізвище)

РЕФЕРАТ

Кваліфікаційна робота магістра «Розробка системи збору даних інтернет магазинів з використанням фреймворку Laravel»: 54 с., 16 рис., 1 табл., 12 джерел, 2 додатки.

AWS, LARAVEL, MYSQL, NODE.JS, PHP, PYTHON, SCRAPY, АНАЛІЗ ДАНИХ, ДОБУВАННЯ ДАНИХ, ІНТЕРНЕТ МАГАЗИН, ПАРСЕР, СКРАПІНГ, ШТУЧНИЙ ІНТЕЛЕКТ.

Об'єкт дослідження – системи збору даних з інтернет магазинів.

Мета роботи: спроектувати та програмно реалізувати систему збору даних інтернет магазинів з використанням фреймворку Laravel.

Метод дослідження – емпіричний.

В результаті розробки системи збору даних інтернет магазинів з використанням фреймворку Laravel були отримані дані о продуктах з таких інтернет магазинів як Amazon та Walmart. Також з використанням вже підготовленого і натренованого AI було проаналізовано відгуки про товар.

Система може використовуватись для забезпечення точною та актуальною інформацією про стан ринку та продуктів у інтернет магазинах, а також може використовуватись як інструмент підвищення продажу конкретних брендів, груп товарів і окремих продуктів.

В майбутньому система може розвиватись шляхом збільшення кількості підтримуваних інтернет магазинів, а також інтеграції з новими системами на основі AI, наприклад збільшення взаємодії з AI для більш ретельного аналізу зібраної інформації про товари.

Ця тема є дуже актуальною і перспективною, оскільки зібрані дані можуть бути основою для тренування AI, що в свою чергу може бути використано для створення прогнозів продажів товарів або для створення рекомендацій ціноутворення на основі зібраних даних конкурентів.

SUMMARY

Master's Qualifying Theses «Development of the Online Stores Data Collection System Using the Laravel Framework»: 54 pages, 16 figures, 1 tables, 12 references, 2 supplements.

ARTIFICIAL INTELLIGENCE, AWS, DATA ANALYSIS, DATA MINING, LARAVEL, MYSQL, NODE.JS, ONLINE STORE, PARSER, PHP, PYTHON, SCRAPING, SCRAPY, WEB STORE.

Object of the study – data collection system from web stores.

Aim of the study: to design and programmatically implement a data collection system for online stores using the Laravel framework

Method of research – empirical

As a result of the development of a data collection system from web stores using the Laravel framework, data about products from such online stores as Amazon and Walmart was obtained. Also, product reviews were analyzed using already prepared and trained AI.

The system can be used to provide accurate and up-to-date information on the state of the market and products in online stores, and can also be used as a tool to increase sales of specific brands, product groups, and individual products.

In the future, the system can be improved by increasing the number of supported online stores, as well as integration with new AI-based systems, for example, increasing interaction with AI for more detailed analysis of collected data about products.

This topic is very relevant and promising, because the collected data can be the basis for AI training, which in turn can be used to create product sales forecasts or to create pricing recommendations based on the collected data of competitors.

ЗМІСТ

| | |
|---|----|
| Завдання на кваліфікаційну роботу..... | 2 |
| Реферат | 4 |
| Summary | 5 |
| Вступ..... | 8 |
| 1 Теоретичні аспекти збору даних в інтернет магазинах..... | 9 |
| 1.1 Поняття інтернет магазину | 9 |
| 1.2 Методи збору даних та їх порівняння..... | 11 |
| 1.3 Етичні аспекти збору даних..... | 14 |
| 1.4 Висновки за розділом 1 | 16 |
| 2 Проектування системи збору даних для інтернет магазину..... | 18 |
| 2.1 Функціональні та технічні вимоги до системи | 18 |
| 2.2 Архітектура системи та інтеграція інструментів для збору даних.. | 21 |
| 2.3 Розгортання системи..... | 24 |
| 2.4 Висновки за розділом 2 | 28 |
| 3 Аналіз та вибір інструментів..... | 29 |
| 3.1 Аналіз та вибір інструментів для збору даних..... | 29 |
| 3.2 Аналіз та вибір інструментів конфігурації системи та візуалізації зібраних даних | 34 |
| 3.3 Висновки за розділом 3 | 37 |
| 4 Розробка програмного забезпечення для збору даних..... | 38 |
| 4.1 Розробка інтерфейсу користувача..... | 38 |
| 4.2 Реалізація функціональності для збору даних | 41 |
| 4.3 Висновки за розділом 4 | 50 |
| Висновки | 51 |
| Перелік посилань..... | 52 |
| Додаток А Структура бази даних | 53 |
| Додаток Б Структура класів | 54 |

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧКИ

| | |
|-----|-----------------------------------|
| AI | Artificial intelligence |
| API | Application programming interface |
| DB | Data base |
| DNS | Domain name system |
| DOM | Document object model |
| EC2 | Elastic compute cloud |
| IP | Internet protocol |
| ORM | Object-relational mapping |
| RSS | Really simple syndication |
| S3 | Simple storage service |
| SQS | Simple queue service |
| UI | User interface |
| WAF | Web application firewall |

ВСТУП

Сучасний ринок електронної комерції переживає стрімкий розвиток, забезпечуючи споживачам зручність та доступність для здійснення покупок з будь-якої точки світу. Зростання конкуренції в електронному бізнесі вимагає від підприємств створення дієвих інструментів для збору та аналізу даних з метою оптимізації їх стратегій та забезпечення успіху.

Ця дипломна робота присвячена розробці системи збору даних інтернет магазинів. Мета дослідження полягає в створенні інструменту, який допоможе ефективно збирати, аналізувати та використовувати дані для підвищення конкурентоспроможності та оптимізації бізнес-процесів електронної комерції.

У цій дипломній роботі будуть розглянуті ключові аспекти розробки системи збору даних для інтернет магазинів, включаючи архітектурні рішення, інтеграцію з іншими системами, методи збору та обробки даних, а також інструменти аналізу з використанням штучного інтелекту. Дослідження базується на аналізі сучасних тенденцій у галузі електронної комерції, інноваційних підходах до збору та аналізу даних, а також на власних дослідженнях і практичного досвіду.

Результати цієї дипломної роботи сприятимуть покращенню роботи інтернет магазинів, допоможуть їм зрозуміти потреби своїх клієнтів, підвищити ефективність маркетингових стратегій, приймати обґрунтовані рішення для подальшого розвитку та відстежувати актуальність розміщеної інформації. Важливість цієї теми полягає в тому, що вона сприяє покращенню конкурентоспроможності виробників товарів та інтернет магазинів на глобальному ринку.

У наступних розділах дипломної роботи будуть розглянуті деталі методології, використані технології, результати дослідження та рекомендації щодо впровадження системи збору даних для інтернет магазинів

1 ТЕОРЕТИЧНІ АСПЕКТИ ЗБОРУ ДАНИХ В ІНТЕРНЕТ МАГАЗИНАХ

1.1 Поняття інтернет магазину

Інтернет магазин (онлайн магазин, електронний магазин) – це електронна торговельна платформа, яка дозволяє покупцям здійснювати покупки товарів і послуг через Інтернет.

Інтернет магазин є сучасною формою електронної комерції, яка надає споживачам можливість придбання товарів і послуг через Інтернет. Ця концепція стала ключовою складовою сучасного бізнесу, забезпечуючи зручність, доступність та широкий вибір для клієнтів.

Інтернет магазини можуть варіюватися за своєю структурою та функціоналом, але всі вони спрямовані на надання віртуальної торговельної площадки, на якій клієнти можуть здійснювати покупки. Вони дозволяють не лише придбати товари, а й отримати інформацію про продукцію, порівняти ціни, скористатися зручними фільтрами та отримати замовлені товари безпосередньо до дверей.

Основними рисами інтернет магазину є:

- віртуальність – інтернет магазин існує в онлайн-середовищі;
- каталог товарів – інтернет магазин має великий каталог товарів або послуг, які можна переглядати, а також шукати і вибирати необхідні позиції;
- електронна корзина – покупці можуть додавати товари до електронної корзини, переглядати її зміст і здійснювати оплату товарів чи послуг онлайн за допомогою платіжних систем;
- зручність і швидкість – покупці можуть швидко і легко придбати товари чи замовити послуги, не залишаючи свого дому;
- доставка – інтернет магазини зазвичай надають послуги доставки, щоб клієнти могли отримати свої замовлення у зручне для них місце;

- онлайн-комунікація – інтернет магазини можуть мати системи онлайн-чату або інші засоби зв'язку для вирішення питань і консультацій з покупцями;

- захист інформації – інтернет магазини зазвичай забезпечують захист особистої інформації клієнтів і безпеку операцій, щоб гарантувати конфіденційність даних.

Інтернет магазини можуть пропонувати різноманітні товари, від одягу і електроніки до послуг, і вони відіграють важливу роль в сучасній електронній комерції, надаючи споживачам можливість здійснювати покупки онлайн.

Однією з ключових переваг інтернет магазинів є безпосередній доступ до глобального ринку, що відкриває безмежні можливості для підприємців та споживачів. Клієнти можуть здійснювати покупки з будь-якого місця світу, а підприємці мають можливість розширити свої бізнес-операції за межі місцевого ринку.

Інтернет магазини використовують різноманітні технології для забезпечення безпеки та конфіденційності транзакцій. Шифрування даних, безпечні платіжні системи та захист від шахрайства роблять електронні торговельні платформи надійними для використання.

Узагальнюючи, інтернет магазин – це інноваційний і ефективний механізм для реалізації товарів та послуг в електронному середовищі, що відповідає вимогам сучасного споживача та сприяє розвитку бізнесу в онлайн-середовищі.

Інтернет магазини містять великий обсяг інформації про товари, яка можуть бути використані для подальшого аналізу, наприклад такі поля як:

- назва товару;
- опис:
 - а) опис характеристик, можливостей та особливостей товару;
 - б) відгуки покупців стосовно товару.
- категорія товару;
- ціна;
- бренд;

- доступність;
 - рейтинг;
 - зображення;
 - додаткові параметри, як колір, тип джерела;
 - варіанти доставки;
 - спеціальні пропозиції та акції;
 - сумісні аксесуари;
 - відомості про виробника;
 - додаткова інформація;
- а) посилання на інші товари від того ж виробника або подібні товари від інших виробників;
- б) рекомендації для покупців.

Ці дані допомагають покупцям знайти необхідну інформацію про товар та прийняти інформоване рішення про покупку.

1.2 Методи збору даних та їх порівняння

Веб-скрапінг – це процес автоматичного збору даних з веб-сайтів. Це важлива та складна задача, яка включає різноманітні методи. Один із способів – це реєстрація користувачів та створення для них особистого кабінету. Це дозволяє отримати особисті дані та забезпечити зручний процес управління замовленнями.

Під час замовлення товарів збирається детальна інформація про вибрані продукти, їх кількість та адресу доставки. Платіжні дані також потрапляють в облік, включаючи інформацію про кредитні картки або інші методи оплати.

Аналітика та слідкування за діяльністю користувачів важливі для оптимізації роботи магазину. Використання cookies дозволяє відстежувати взаємодію користувачів із сайтом, отримувати дані про перегляди сторінок та обрані товари.

Також інформація збирається через контактні форми, анкети та відгуки користувачів. Співпраця з партнерами може включати обмін даними для покращення обслуговування та рекламних кампаній.

Збір даних у сучасному інтернет магазині – це комплексний процес, що враховує різні аспекти взаємодії із користувачами та оптимізації бізнес-процесів.

Загалом, методи збору даних в інтернет магазинах поєднують у собі технічні, аналітичні та соціальні компоненти. Це допомагає забезпечувати оптимальний сервіс для клієнтів, адаптуватися до ринкових змін та підвищувати конкурентоспроможність у сфері електронної комерції.

Співпраця з соціальними мережами дозволяє отримувати дані про користувачів, які взаємодіють з магазином через ці платформи. Взаємодія з партнерами та інтеграція із сторонніми сервісами розширює можливості обміну інформацією для поліпшення якості обслуговування.

Форми збору даних також включають технічні деталі, такі як тип пристрою, роздільна здатність екрану та використовуваний браузер. Це надає додаткову інформацію про технічні характеристики користувачів.

Автоматизовані сповіщення та листи також збирають інформацію про те, як користувачі реагують на електронні повідомлення. Збір даних відбувається також під час взаємодії з контентом, що може включати відгуки, рейтинги та коментарі.

Усі ці методи створюють повну картину взаємодії користувачів із інтернет магазином, що допомагає оптимізувати роботу платформи, покращити обслуговування та надати користувачам зручний та персоналізований досвід покупок.

Окрім технічних аспектів, процес збору даних в інтернет магазинах враховує також взаємодію з клієнтами. Отримання відгуків та коментарів дозволяє отримати споживчу інформацію та удосконалити асортимент товарів та якість обслуговування.

Розуміння та врахування психологічних аспектів покупки також є частиною збору даних. Спостереження за тим, як користувачі взаємодіють із веб-

сайтом та реагують на різні стимули, дозволяє створювати більш ефективні маркетингові стратегії.

Збір даних з веб-сайтів може бути виконаним різними методами, в залежності від конкретних потреб та обмежень проекту. Нижче приведено порівняння методів збору даних з веб-сайтів:

- використання API:
 - а) переваги – цей метод найбільш надійний та легальний, оскільки дозволяє отримувати структуровані дані без порушення правил сайту;
 - б) недоліки – деякі сайти можуть не надавати API, або мати обмеження доступу. Іноді це може бути платним сервісом;
- веб-скрапінг (Web Scraping):
 - а) переваги – дозволяє отримувати дані з майже будь-якого веб-сайту. Може бути корисним, якщо API недоступний;
 - б) недоліки – може супроводжуватися юридичними проблемами, у разі приватної інформації, та частими змінами у структурі сайту. Також це може навантажувати сервери сайту;
- збір даних через RSS-стрічки або sitemaps:
 - а) переваги – деякі сайти надають RSS-стрічки або sitemaps, які спрощують збір даних без необхідності скрапінгу кожної сторінки окремо;
 - б) недоліки – цей метод може не надавати повних даних, які потрібні;
- збір даних через інтеграцію зі службами різних сторінок:
 - а) переваги – деякі сторінки мають можливість інтеграції з іншими службами, що дозволяє отримувати дані, в більшості випадків такі інтеграції використовують API;
 - б) недоліки – цей метод може бути обмеженим конкретними інтеграціями, і не всі дані можуть бути доступні;
- комбінація різних методів:

- а) переваги – іноді комбінація декількох методів може дати кращі результати, дозволяючи отримати більше даних та забезпечити їхню якість та актуальність;
- б) недоліки – може вимагати більше зусиль для розробки та підтримки.

Обираючи метод збору даних, важливо враховувати правові та етичні аспекти, а також технічні обмеження. Кращий метод буде залежати від конкретних цілей та обставин вашого проекту.

Окремим аспектом збору даних є використання технологій аналізу та машинного навчання. Ці інструменти дозволяють не лише збирати, а й аналізувати величезний обсяг даних для отримання важливих інсайтів щодо купівельної поведінки, попиту на товари та ефективності маркетингових кампаній.

Зокрема, здійснення аналізу даних допомагає виявити та передбачити тенденції споживчого ринку, що дозволяє магазину адаптуватися до змін у попиті та покращити стратегії просування товарів.

Важливо враховувати етичні питання при зборі даних і забезпечувати високий рівень конфіденційності та захисту особистої інформації користувачів. Враховуючи сучасні тенденції в електронній комерції та використання інноваційних методів збору та аналізу даних, інтернет магазини можуть успішно пристосовуватись до вимог ринку та надавати споживачам більш персоналізований та зручний сервіс.

1.3 Етичні аспекти збору даних

Збір та обробка даних в інтернет магазинах – процес, що вносить значний вклад у функціонування бізнесу, але вимагає уважності та дотримання етичних стандартів для забезпечення конфіденційності та захисту прав клієнтів.

Збір або парсинг даних з різних джерел, включаючи інтернет магазини, пов'язаний з численними етичними питаннями і вимагає дотримання кількох важливих аспектів. Ось деякі етичні аспекти збору даних з інтернет магазинів:

- згода та засади використання даних. Перш за все, важливо отримати згоду від магазину або користувачів для збору та використання їхніх даних. Повинні бути чіткі правила щодо того, як будуть використовуватися зібрані дані та в яких цілях;

- обмеження доступу до публічних даних. Деякі дані можуть бути доступні публічно, але не всі. Дотримання принципу "парсити тільки те, що публічно доступно" є важливим. Забороняється обходити обмеження доступу, наприклад, ігноруючи файл robots.txt;

- обмеження на навантаження серверів. Під час збору даних потрібно бути обережними, щоб не завантажувати сервери магазину занадто сильно. Це може спричинити проблеми з доступністю для інших користувачів;

- захист особистих даних. Якщо зібрані дані містять особисту інформацію, то вони повинні бути анонімізовані або використовуватися з урахуванням правил щодо конфіденційності та захисту особистих даних користувачів;

- рівність та справедливість. Під час аналізу зібраних даних слід уникати поділу користувачів на основі раси, статі, релігії або інших особистих характеристик, а також уникати створення алгоритмів, які можуть призвести до дискримінації;

- відповідальність та прозорість. Інформація про те, які дані збираються і як вони використовуються, повинна бути доступною для всіх зацікавлених сторін. Також важливо бути готовим до відповіді на запити від користувачів щодо їхніх даних;

- захист інтелектуальної власності. Під час збору даних, особливо коли вони містять інформацію про товари, важливо дотримуватися законів про авторські права та інтелектуальну власність;

– моральність та добродесність. Дотримання високих стандартів моралі та добродесності при зборі та використанні даних є важливим аспектом. Треба уникати недобродесних практик, які можуть завдати шкоди іншим користувачам або магазинам;

– відповідність законодавству. Алгоритм збору даних повинен дотримуватися всіх відповідних законів та нормативів щодо захисту даних та інших відповідних аспектів.

1.4 Висновки за розділом 1

Ознайомившись з методами та етичними аспектами збору даних з'являється можливість обрати підходящий метод для збору даних з інтернет магазинів Walmart та Amazon за допомогою порівняльної таблиці 1.1.

Таблиця 1.1 – Порівняльна таблиця методів збору даних для веб-магазинів

| Метод збору | Walmart | Amazon |
|------------------|--|--|
| Використання API | API вимагає послідовних запитів для отримання даних. Недоліком є часті зміни в структурі API, що змушує постійно оновлювати код, а також необхідність написання запитів на мові GraphQL, що підвищує вимоги до розробника. | Потребує реєстрації і верифікації на платформі Amazon, а також накладає ряд обмежень на отримувані дані. З огляду цього не є оптимальним методом. |
| Веб-скрапінг | Дозволяє отримати дані в вигляді, як їх бачить людина на веб-сторінці, без необхідності додаткових запитів. | Дозволяє отримати дані в вигляді, як їх бачить людина на веб-сторінці, без необхідності додаткових запитів. Недоліком є необхідність емуляції браузера для отримання всієї інформації. |

Продовження таблиці 1.1

| Метод збору | Walmart | Amazon |
|---|--|--|
| Збір даних через RSS-стрічки або sitemapс | Не зможе надати повних даних про товар, які потрібні виходячи з завдання. | Не зможе надати повних даних про товар, які потрібні виходячи з завдання. |
| Збір даних через інтеграцію зі службами різних сторінок | Потребує використання готових сторонніх платних сервісів, що не підходить до умов завдання | Потребує використання готових сторонніх платних сервісів, що не підходить до умов завдання |
| Комбінація різних методів | В даному випадку не є необхідною, оскільки всі дані можна отримати за допомогою веб-скрапінга. В разі необхідності, в залежності від додаткових вимог, має підійти комбінація веб-скрапінга + API. | В даному випадку не є необхідною, оскільки всі дані можна отримати за допомогою веб-скрапінга. В разі необхідності, в залежності від додаткових вимог, має підійти комбінація веб-скрапінга + API. |

На основі порівняння методів збору даних для інтернет магазинів Amazon та Walmart можна зробити висновок, що найбільш підходящим та універсальним методом буде веб-скрапінг, який дозволить зібрати всі необхідні дані з мінімальною кількістю запитів, а також позитивно вплине на навантаження як для системи, що розробляється, так і для сайту інтернет магазину. При цьому дуже важливим залишається дотримання етичних аспектів збору даних не тільки на етапі вибору методів, але і на етапі розробки системи.

Слід відмітити, що протягом часу актуальність методів збору даних може змінитись, це може залежати від багатьох факторів, таких як зміни вимог до системи, даних, що збираються, змін на сайтах, з яких збираються дані, зміни в законодавстві і навіть в етичних аспектах збору даних.

2 ПРОЕКТУВАННЯ СИСТЕМИ ЗБОРУ ДАНИХ ДЛЯ ІНТЕРНЕТ МАГАЗИНУ

2.1 Функціональні та технічні вимоги до системи

При проектуванні системи збору даних для інтернет магазину слід враховувати функціональні та технічні вимоги. Функціональні вимоги включають в себе необхідність збору та обробки різних видів даних, таких як особисті дані користувачів, інформація про замовлення, інвентар, статистика продажів тощо.

До функціональних вимог також входить можливість генерації звітів та аналізу даних для прийняття ефективних управлінських рішень. Крім того, система повинна бути легко інтегрованою з іншими сервісами, такими як платіжні системи чи системи управління відносинами з клієнтами (CRM).

До функціональних вимог відносяться:

- можливість реєстрації нових користувачів та Авторизація користувачів для доступу до особистого кабінету;
- можливість конфігурації даних про товари:
 - а) додавання товарів до каталогу, включаючи назву, опис, фотографії;
 - б) встановлення категорій товарів;
 - в) додавання брендів товарів;
 - г) визначення продавців товарів;
 - д) можливість редагування атрибутів товарів (наприклад, розмір).
- збір даних товарів згідно каталогу, включаючи назву, опис, ціну, наявність, фотографії і т.д;
- аналітика та звіти:
 - а) відображення глобальної статистики на UI;

б) генерація звітів для прийняття рішень щодо асортименту та маркетингу.

- можливість збирати відгуки та оцінки товарів
- забезпечення конфіденційності та безпеки персональних даних

користувачів

- пошук та фільтрація:

а) можливість швидкого пошуку товарів;

б) можливість фільтрації за різними параметрами при генерації звітів.

Технічні вимоги визначають архітектуру системи, використовувані технології та засоби зберігання даних. Важливо забезпечити безпеку та конфіденційність даних, а також врахувати можливість масштабування системи в майбутньому.

Технічні вимоги включають в себе вибір відповідних баз даних, засобів для розробки та тестування, а також розгляд аспектів безпеки, таких як шифрування даних та захист від несанкціонованого доступу.

Технічні вимоги включають:

- збір даних:

а) система повинна бути здатна витягнути дані з різних інтернет магазинів. Це може бути веб-скрапінг, використання API або інших методів;

б) система має мати можливість обходити алгоритми блокувань інтернет магазинів.

- збереження даних:

а) зібрані дані повинні зберігатися в базі даних для подальшого аналізу та використання;

б) система має мати окреме сховище для збереження зображень;

в) зібрані дані мають мати резервне копіювання.

- масштабованість:

а) система повинна мати можливість масштабуватись, щоб в разі потреби збирати дані з більшої кількості інтернет магазинів без великих зусиль з боку розробників;

б) система має бути поділена на логічні частини, які можуть масштабуватись відокремлено одна від одної.

– автоматизація:

а) налаштована система повинна бути здатною автоматично оновлювати дані на регулярній основі без необхідності втручання людини;

б) система має автоматично масштабуватись в залежності від навантаження.

– обробка даних:

а) система повинна забезпечувати базову обробку та структурування зібраних даних;

б) фільтрація та трансформація даних для підготовки до аналізу.

– система повинна мати засоби моніторингу та логування, щоб слідкувати за її роботою та виявляти можливі проблеми;

– можливість інтеграції з іншими системами для обміну даними, наприклад, з системами аналітики або CRM;

– UI та сховища даних мають бути захищеним від несанкціонованого доступу;

– система повинна бути швидкою та надійною в роботі;

– наявність докладної технічної документації для системи та її інтеграції;

– система повинна мати план для резервного відновлення системи у випадку виникнення непередбачених ситуацій.

Загалом, система збору даних повинна бути гнучкою, легко налаштовуваною та готовою відповідати змінним вимогам бізнесу та ринку електронної торгівлі.

Узагальнюючи, функціональні та технічні вимоги повинні забезпечити ефективну та безпечну роботу системи збору даних для інтернет магазину, сприяючи оптимізації управління та покращенню обслуговування клієнтів.

2.2 Архітектура системи та інтеграція інструментів для збору даних

Для архітектури системи інтернет магазину рекомендується використовувати мікросервісну модель, де кожна функціональна частина реалізується окремим мікросервісом. Це дозволяє підтримувати гнучкість та масштабованість системи.

У системі повинна бути використана відповідна база даних для зберігання різноманітної інформації, такої як дані про товари, замовлення, користувачів. Для ефективного управління базою даних можна розглядати різні системи, такі як MySQL, PostgreSQL або NoSQL рішення в залежності від потреб.

Щодо інтеграції, важливо забезпечити взаємодію із зовнішніми сервісами, такими як платіжні шлюзи, системи доставки та інші інструменти, які допоможуть оптимізувати роботу магазину та полегшити процеси для клієнтів.

Враховуючи ці принципи, система буде здатна ефективно обробляти та зберігати дані, забезпечуючи стабільну та масштабовану архітектуру.

Головною концепцією системи є модульність, що дозволить отримати наступні переваги:

- легший розвиток. Розподілення на фронтенд та бекенд дозволяє розділити функціональність системи на менші, незалежні модулі. Це спрощує розробку та підтримку, оскільки кожен модуль може розроблятися та підтримуватися окремо;
- покращена масштабованість. Модульність дозволяє додавати нові ресурси та функціональність лише там, де це потрібно. Це полегшує масштабування системи, оскільки стає можливо масштабувати окремі найбільше навантажені модулі;

- забезпечення стабільності та надійності. Якщо один з модулів виходить з ладу, інші модулі можуть продовжувати працювати. Це допомагає запобігти відмовам у системі та забезпечує вищу стабільність та надійність;
- зменшення залежностей. Модульна архітектура зменшує залежності між різними частинами системи, що полегшує розробку та підтримку, а також сприяє швидшому відновленню після відмов;
- легше тестування та налагодження. Модульність дозволяє проводити тестування кожного модуля окремо, що спрощує виявлення та виправлення помилок;
- забезпечення безпеки. Ви можете керувати доступом до різних частин системи окремо, що дозволяє підвищити безпеку даних та системи в цілому.

Загалом, модульність та розподілення на фронтенд та бекенд сприяють створенню більш ефективних, стабільних та масштабованих систем для збору даних.

Для системи збору даних можна виділити наступні модулі:

- Scheduler Server – відповідальний за запуск збору даних за розкладом;
- Scrapers API Servers – група серверів, відповідальна за доступ до даних, а також роботу з даними в рамках процесу збору;
- Scrapers Servers – група серверів, відповідальна за збір даних з інтернет магазинів;
- Proxy Servers – група серверів, відповідальна за зміну фізичної локації запита на збір даних;
- Saver Servers – група серверів, відповідальна за попередню обробку та збереження інформації за допомогою Scrapers API Servers;
- Workers Servers – група серверів, відповідальна за виконання довгострокових задач, наприклад операції порівняння зображень за допомогою AI;

– DB – група серверів, відповідальна за довгострокове збереження даних.

Взаємодія між зазначеними модулями наглядно представлена на рисунку 2.1.

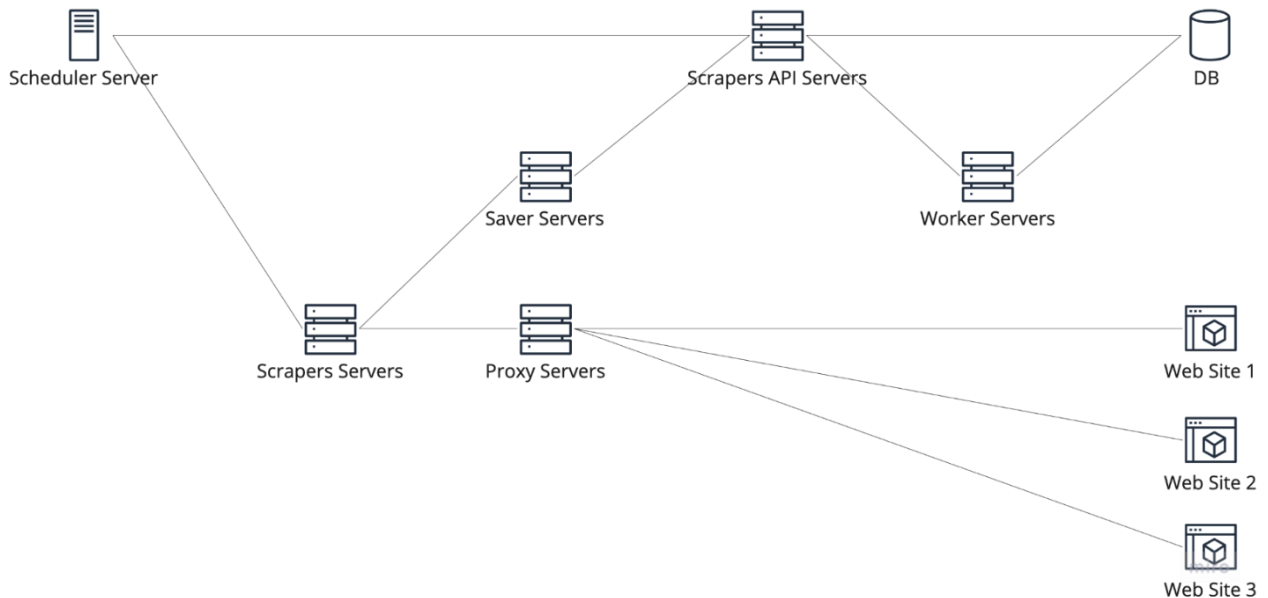


Рисунок 2.1 – Архітектурна схема розподілення на модулі системи збору даних

Для UI системи збору даних можна виділити наступні модулі:

- WAF – firewall для підвищення рівня захисту додатку від зовнішніх атак;
- UI Servers – група серверів, відповідальна за програмний інтерфейс, з яким буде працювати користувач системи;
- Worker Servers – група серверів, відповідальна за виконання довгострокових задач, наприклад підготовки імпорту або експорту даних;
- DB – група серверів, відповідальна за довгострокове збереження даних;
- Storage – група серверів, відповідальна за довгострокове збереження об'єктів, наприклад зображень або звітів.

Зв'язок між зазначеними модулями наглядно представлений на рисунку 2.2.

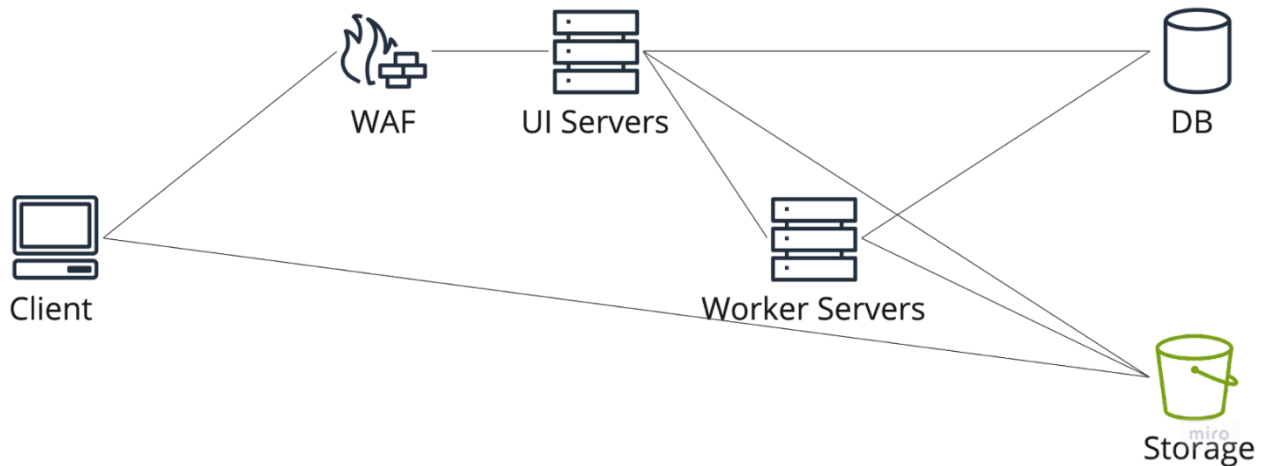


Рисунок 2.2 – Архітектурна схема розподілення на модулі UI

Інтеграція інструментів для збору даних з інтернет магазинів може вимагати використання сервісів проксі та рішень для обходу систем капчі. Сервіси проксі дозволяють отримувати дані з інтернет магазинів, обмежуючи блокування IP-адрес користувача. Використання сервісів для розв'язування капчі, у свою чергу, надають можливість автоматизовано розв'язувати капчі, що можуть виникнути під час збору даних, забезпечуючи безперервну роботу інтегрованої системи збору даних.

В залежності від обсягів даних, збір може вимагати значних системних ресурсів для підтримки процесу збору та збереження інформації.

2.3 Розгортання системи

Для розгортання системи потрібно вибрати хостинг, який надає хмарні послуги. Збір даних може потребувати великих обсягів ресурсів, а також можливості детальної конфігурації, це потрібно враховувати при виборі.

Хостинг від AWS, Microsoft Azure та Google Cloud Platform – це хмарні послуги, які надають інфраструктуру для розгортання та управління веб-додатками і сервісами. AWS пропонує широкий спектр рішень та гнучкі опції, Azure відзначається інтеграцією з продуктами Microsoft та розширеними можливостями для корпоративних систем, а Google Cloud Platform відомий

своєю швидкістю, інструментами для штучного інтелекту та аналітики даних. Кожна з цих платформ пропонує великий перелік сервісів та функціональні можливості для задоволення потреб розробників та підприємств у хмарному середовищі.

Деплоймент на AWS може бути найбільш вигідним з кількох причин:

- широкий спектр послуг. AWS пропонує величезний вибір послуг, які можуть використовуватися для різноманітних потреб, включаючи обчислення, зберігання, використання бази даних, машинне навчання, аналітику та інше. Це дозволяє легко масштабувати та налаштовувати інфраструктуру під конкретні вимоги;
- доступність. AWS має дата-центри по всьому світу, що дозволяє розгортати додатки територіально ближче до аудиторії. Це забезпечує менший час відгуку та поліпшує продуктивність для користувачів у різних регіонах;
- масштабованість та гнучкість. AWS дозволяє легко масштабувати кількість серверів, як в більшу, так і в меншу сторону, в залежності від потреб додатку. Також він надає можливість використання служб автоматичного масштабування, щоб забезпечити ефективне використання ресурсів;
- безпека. AWS забезпечує високий рівень безпеки та відповідність стандартам. Також є можливість використовувати різні інструменти для керування доступом, шифрування та моніторингу, щоб захистити дані;
- інновації та нові технології. AWS постійно впроваджує нові сервіси та технології, які дуже легко використовувати у власному додатку, що може полегшити подальшу підтримку додатків;
- спільнота та підтримка. AWS має велику та активну спільноту розробників. Це означає, що можна швидко знайти відповіді на свої питання, а також скористатися різними ресурсами та матеріалами для розвитку проекту.

Вибір хостингу надає можливість оновити і деталізувати архітектуру системи з використанням визначених сервісів і ресурсів.

Результатом оновлення архітектури на рисунку 2.1, з використанням рекомендованих позначень для сервісів від AWS – є рисунок 2.3, на якому було введено наступні сервіси:

- Amazon CloudFront – це служба CDN (Content Delivery Network), яка дозволяє розповсюджувати контент, такий як зображення, відео та інші статичні файли, по серверах, що розташовані по всьому світу, для поліпшення швидкості завантаження та забезпечення низької затримки для кінцевих користувачів;
- Amazon Route 53 – це служба управління доменними іменами та системи назв (DNS), яка надає можливість реєстрації доменів, управління записами DNS та забезпечення високої доступності та надійності для додатків;
- Amazon Load Balancer – це сервіс балансування навантаження, який автоматично розподіляє вхідний трафік між різними екземплярами Amazon EC2 або іншими ресурсами, забезпечуючи високу доступність та оптимізацію використання ресурсів;
- Amazon WAF – це служба захисту веб-додатків, яка дозволяє захищати веб-додатки від вразливостей та атак, використовуючи налаштування правил та фільтрів, щоб контролювати та мінімізувати потенційні загрози;
- Amazon EC2 Auto Scaling – це сервіс автоматичного масштабування, який дозволяє автоматично змінювати кількість екземплярів Amazon EC2, щоб забезпечити оптимальну продуктивність та ефективне використання ресурсів в залежності від обсягу роботи;
- Amazon Aurora – це служба керування базами даних, яка надає високозабезпечений та масштабований сервіс для MySQL та PostgreSQL. Вона забезпечує високу доступність, надійність та автоматичне резервне копіювання;
- Amazon S3 – це служба зберігання об'єктів, яка надає масштабоване та безпечне зберігання для об'єктів, таких як файли, зображення та дані. Вона дозволяє зберігати та отримувати дані з будь-якого місця у мережі Інтернет.

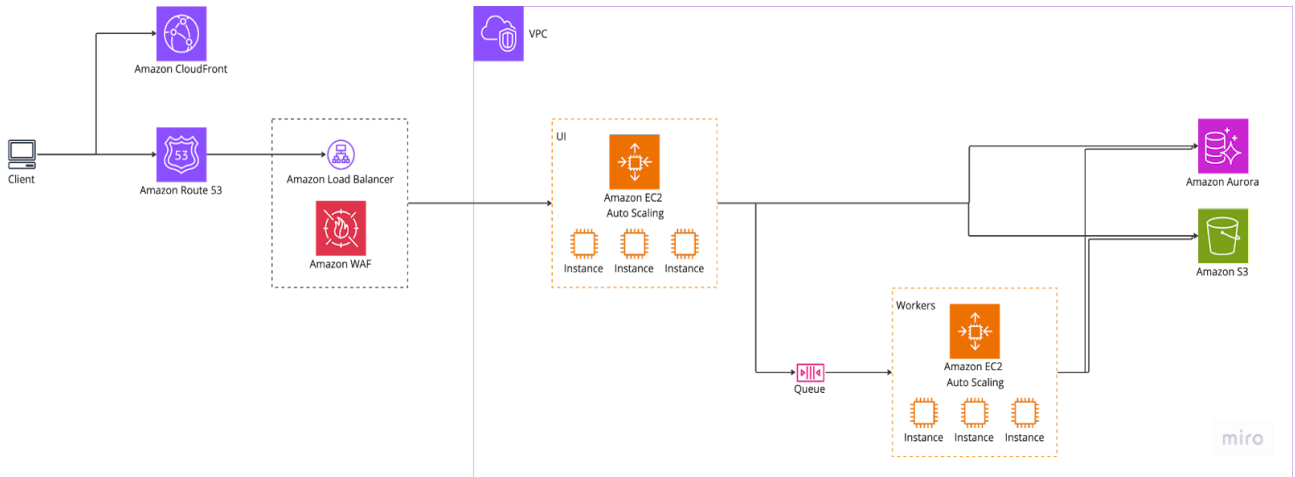


Рисунок 2.3 – Архітектурна схема розподілення на модулі UI на AWS

Результатом оновлення архітектури на рисунку 2.2, з використанням рекомендованих позначень для сервісів від AWS – є рисунок 2.4, на якому можна побачити додаткове використання черг для комунікації між модулями. Черги на AWS реалізовані за допомогою Amazon SQS. Amazon SQS – це служба керування чергами повідомлень, яка дозволяє легко розподіляти роботу між різними компонентами додатка або системи. Вона надає простий та масштабований спосіб обміну повідомлень між різними частинами архітектури, а також забезпечує високу доступність та надійність.

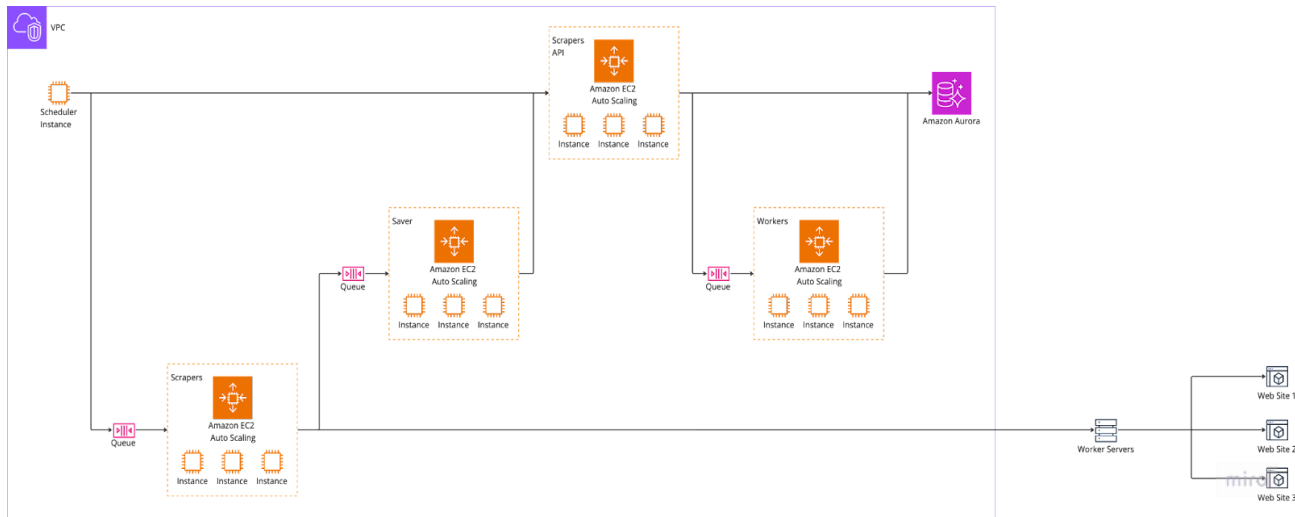


Рисунок 2.4 – Архітектурна схема розподілення на модулі системи збору даних на AWS

2.4 Висновки за розділом 2

Ефективне та гарно продумане проектування системи збору даних є ключовим етапом для успішної розробки системи збору даних інтернет магазинів.

У підрозділі "Функціональні та технічні вимоги до системи" були визначені основні функціональні можливості, які повинна підтримувати система, а також головні технічні вимоги, яким система має відповідати. Це надає чітку основу для розробки, розгортання та тестування програмного забезпечення.

Опис функціональних та технічних вимог дозволив досягти розбиття системи на модулі, які будуть відповідати поставленим вимогам і зможуть забезпечити ефективне функціонування системи, а також взаємодію з інструментами для збору даних.

Все це допомогло визначитись з критеріями для розгортання системи а також вибором хостингу. Використання Amazon дозволяє отримати гнучку, здатну масштабуватись систему з великим рівнем надійності та захищеності, а також надає можливість використовувати переваги великої кількості доступних сервісів, які пропонує Amazon і які часто сумісні між собою.

Загалом, проектування системи збору даних для інтернет-магазину враховує як функціональні, так і технічні аспекти, що сприяє створенню ефективної та надійної інфраструктури для збору та обробки даних.

3 АНАЛІЗ ТА ВИБІР ІНСТРУМЕНТІВ

3.1 Аналіз та вибір інструментів для збору даних

Активний розвиток електронної комерції, веб-сайтів та штучного інтелекту дали потужний поштовх розвитку інструментів та бібліотек для збору даних.

Інтернет магазини використовують різноманітні інструменти для збору даних та аналізу їхнього функціонування. Одним із ключових інструментів є аналітичні платформи, такі як Google Analytics, Adobe Analytics та Yandex Metrica. Ці інструменти надають глибокий аналіз трафіку, поведінки користувачів та результативності маркетингових кампаній.

Трекінг конверсії – ще одна важлива частина збору даних. Інструменти, такі як Hotjar та Crazy Egg, дозволяють відстежувати взаємодію користувачів із сторінками, аналізувати кліки та прокручування сторінок. Оптимізація конверсії може бути проведена за допомогою інструментів типу Optimizely та VWO.

Для збору даних про взаємодію користувачів із контентом інтернет магазини використовують системи управління контентом, а також інтеграції з соціальними мережами. Такі дані можуть включати відгуки, рейтинги товарів та коментарі користувачів.

Додатково, інтернет магазини можуть застосовувати інструменти машинного навчання для аналізу купівельної поведінки та прогнозування тенденцій ринку. Це допомагає бізнесу приймати обґрунтовані рішення щодо асортименту товарів та стратегій маркетингу.

Використання всіх цих інструментів дозволяє інтернет магазинам отримувати повний обсяг даних, необхідних для ефективного управління та вдосконалення свого функціонування.

До інструментів для збору даних також можна віднести системи управління взаємодією з клієнтами (CRM), які відстежують та аналізують

інформацію про клієнтів, їхні покупки та інші важливі дані. Це дозволяє персоналізувати обслуговування та підходити до клієнтів індивідуально.

У сфері маркетингу використовуються інструменти автоматизації, такі як Mailchimp, SendinBlue чи HubSpot, для збору даних про ефективність електронних розсилок, взаємодії з рекламними кампаніями та конверсії.

Інтернет магазини також можуть використовувати інструменти аналізу соціальних мереж, які надають важливу інформацію про реакції користувачів на продукти та бренд через платформи, такі як Facebook Insights чи Twitter Analytics.

Загалом, використання різноманітних інструментів для збору та аналізу даних допомагає інтернет магазинам приймати обґрунтовані рішення, вдосконалювати свою діяльність та надавати користувачам персоналізований та ефективний досвід покупок.

Для збору даних інтернет магазинів найбільш універсальним методом є веб-скрапінг, при цьому вже є готові системи збору:

- Arify – це хмарна платформа для веб-скрапінгу та автоматизації веб-діяльності. Вона надає інструменти для створення веб-скраперів, розгортання їх в хмарі та керування видобутком даних;
- Octoparse – це візуальний веб-скрапер, який дозволяє користувачам створювати скрапери без програмування. Він має інтуїтивний інтерфейс та підтримує автоматичний збір даних з інтернет магазинів;
- Import.io – це інтернет-сервіс для збору даних з веб-сторінок. Він надає можливість створювати API для веб-сайтів, щоб легко добувати дані для подальшого аналізу;
- WebHarvy – це програма для парсингу вмісту веб-сторінок. Вона дозволяє користувачам витягувати дані з інтернет магазинів та зберігати їх у різних форматах, таких як CSV або Excel.

Перелічені вище системи вже є готовими програмними продуктами для надання послуг, при цьому вони або надають тільки можливості збору даних, без можливості аналітики, або мають обмежений функціонал.

Під час веб-скрапінгу важливо розуміти HTML-структуру веб-сторінок та взаємодію з DOM-деревом. Важливо звертати увагу на стійкість коду до можливих змін у HTML-структурі веб-сайту, оскільки вони можуть виникнути з часом. Регулярна перевірка та оновлення коду в разі потреби допоможуть уникнути проблем зі скрапінгом.

Перед тим як розпочати веб-скрапінг, рекомендується перевірити файл "robots.txt" веб-сайту для визначення обмежень щодо скрапінгу. Використання API від власників веб-сайтів може бути ефективнішим та менш конфліктогенним шляхом отримання даних.

Важливо враховувати, що при веб-скрапінгу потрібно уникати надмірного навантаження на сервери веб-сайтів та дотримуватися політик їх використання. Власники веб-сайтів можуть встановлювати обмеження та блокувати доступ для веб-скрапінгу, тому важливо дотримуватися правил і обмежень, вказаних на веб-сайті. Автоматизація процесу дозволяє здійснювати регулярне оновлення даних.

В цілому, веб-скрапінг є корисним інструментом для отримання даних, але використовувати його слід відповідально та з урахуванням етичних та юридичних аспектів.

Для реалізації нової системи веб-скрапінгу слід розглянути самі популярні інструменти, які в комбінації дозволять досягти мети проекту, це: Scrapy та Puppeteer.

Scrapy – це потужний фреймворк для веб-скрапінгу та видобування даних з веб-сайтів. Він написаний на мові програмування Python і створений з урахуванням потреб веб-розробників і дослідників даних. Scrapy надає ряд переваг, що роблять його зручним і ефективним інструментом для збору і аналізу інформації з Інтернету. Ось деякі з них:

- асинхронність. Scrapy підтримує асинхронний підхід, що дозволяє швидко і ефективно збирати дані з різних джерел одночасно. Це допомагає зменшити час, потрібний для скрапінгу великих обсягів інформації;

- вбудовані засоби вилучення. Scrapy має потужні інструменти для вилучення даних із сторінок веб-сайтів. Ви можете використовувати CSS-

селектори, XPath або регулярні вирази для точного визначення та вилучення необхідних даних;

- обробка вхідних даних. Scrapy дозволяє обробляти дані в реальному часі, перетворюючи їх у зручний для подальшого аналізу формат, наприклад, у JSON, CSV або базу даних;

- плагіни та розширення. Scrapy підтримує різні плагіни та розширення, що спрощують роботу з різними типами даних і джерелами, а також дозволяє легко налаштовувати і розширювати функціональність фреймворку;

- розподілені запити. Scrapy може розподіляти запити до веб-серверів для зменшення навантаження на ці сервери і збільшення швидкості скрапінгу.

Scrapy дозволяє автоматизувати процес збору даних і отримувати оновлення з веб-сайтів, що регулярно публікують нову інформацію. Фреймворк є надійним і широко використовується у галузі аналізу даних, розвідки і моніторингу веб-сайтів.

Puppeteer – це високорівнева Node.js бібліотека, яка надає інтерфейс для керування та автоматизації веб-переглядачів, таких як Google Chrome або Chromium. Ця бібліотека використовується для веб-скрапінгу, тестування веб-сайтів, автоматизації веб-додатків і багатьох інших завдань. Ключові особливості Puppeteer:

- контроль браузера. Puppeteer дозволяє створювати новий екземпляр веб-переглядача і керувати ним з коду, наприклад можна надсилати команди для відкриття сторінок, навігації, взаємодії зі сторінкою та багатьма іншими операціями;

- захоплення вмісту. Puppeteer може використовуватись для отримання HTML-коду сторінки, зображень, стилів, JavaScript-файлів та іншого вмісту, що завантажується в браузері;

- робота з формами і кліками. Puppeteer надає зручний інтерфейс для взаємодії з формами, відправки запитів, кліків на посиланнях і багатьох інших подій на веб-сторінці;

- скріншоти і PDF-звіти. Puppeteer дозволяє створювати скріншоти сторінок та генерувати PDF-звіти з вмісту сторінок за допомогою;
- простота встановлення та використання. Puppeteer легко встановити через npm, і він має дружній API, який дозволяє швидко створювати скрапери та автоматизовані завдання;
- підтримка різних платформ. Puppeteer підтримує як Chrome, так і Chromium, що надає можливість вибрати, який браузер використовувати в залежності від потреби;
- асинхронність і чекери. Puppeteer ідеально підходить для асинхронної роботи та чекерів, що дозволяє ефективно очікувати завантаження сторінок та реагувати на події в браузері.

Загалом, Puppeteer є потужним і зручним інструментом для веб-скрапінгу і автоматизації завдань, пов'язаних з веб-переглядачами, завдяки чому він корисний для розробників і дослідників даних.

Важливо пам'ятати, що веб-скрапінг має бути використаний лише в межах закону та відповідно до політик конкретного веб-сайту. Слід зосереджуватися на забезпеченні конфіденційності та безпеки отриманих даних. Якщо отримані дані містять особисту інформацію, важливо дотримуватися відповідних стандартів і правил, щоб не порушити приватність користувачів та власників інформації.

Для покращення ефективності веб-скрапінгу, важливо використовувати асинхронні запити та оптимізуйте код. Це допоможе зменшити час виконання та збільшити швидкість отримання даних. Також треба наголосити на необхідності регулярного моніторингу та аналізу результатів веб-скрапінгу, щоб переконатися, що вони відповідають очікуванням та не потребують коригувань.

Враховуючи всі ці аспекти, веб-скрапінг може бути ефективним інструментом для отримання даних з Інтернету з урахуванням етичних, юридичних та технічних аспектів.

3.2 Аналіз та вибір інструментів конфігурації системи та візуалізації зібраних даних

Окрім безпосереднього збору даних, дуже важливою частиною системи є UI, який дозволить налагодити взаємодію користувача з системою.

Серед великої кількості інструментів, доступних для створення UI, в першу чергу треба визначити мову програмування, а вже потім фреймворк для розробки.

RНР є однією з найпоширеніших мов програмування для розробки веб-інтерфейсів, RНР дозволяє швидко розробляти веб-додатки завдяки своєму простому синтаксису та величезній кількості готових функцій. Ця простота робить RНР доступним для широкого кола розробників.

RНР добре інтегрується з багатьма системами управління БД, що робить його ефективним інструментом для роботи з даними. Відкритий характер мови дозволяє розробникам легко взаємодіяти з іншими технологіями і створювати гнучкі та масштабовані веб-додатки.

RНР також користується широкою підтримкою та активною спільнотою, що означає наявність великої кількості ресурсів, документації та зовнішніх бібліотек. Все це полегшує вирішення завдань розробки та сприяє швидкому вдосконаленню навичок програмістів. RНР є дуже практичним і гнучким інструментом для створення веб-інтерфейсів, забезпечуючи надійність та ефективність веб-розробки.

Після визначення мови програмування наступним кроком має бути вибір фреймворку. Найбільш підходящим фреймворком для обраної мови програмування є Laravel. Основними критерії для вибору фреймворку є:

- сучасність. Laravel є одним з найсучасніших RНР-фреймворків, який використовує сучасні підходи та добре реагує на вимоги сучасної веб-розробки.
- ефективність та продуктивність. Laravel надає ефективні та продуктивні інструменти, що дозволяють швидко створювати високоякісні веб-додатки.

- зручний синтаксис. Laravel має зручний та лаконічний синтаксис, що робить код чистим і легким для зрозуміння, а інтеграція з Eloquent ORM дозволяє зручно взаємодіяти з базою даних та спростити взаємодію з даними в програмі.

- зручна система маршрутизації та контроль доступу. Laravel надає зручні інструменти для визначення маршрутів та контролю доступу, що спрощує розробку функціональних і безпечних веб-додатків.

- активна спільнота. Laravel має велику та активну спільноту розробників, що означає, що фреймворк і в подальшому матиме активну підтримку та розвиток.

- широкий вибір розширень. доступ до великої кількості безкоштовних open source бібліотек, які підтримуються та постійно розвиваються.

Одним з дуже корисних розширень для Laravel є AdminLTE. AdminLTE – це безкоштовний адміністративний шаблон (admin dashboard template) з відкритим кодом для веб-додатків. Цей шаблон побудований на базі популярного фреймворку для розробки веб-інтерфейсів – Bootstrap. AdminLTE надає готовий і стильний інтерфейс для адміністративних панелей, які використовуються для управління та моніторингу веб-додатків або інших систем.

Головними перевагами AdminLTE є:

- сучасний та стильний дизайн. AdminLTE має сучасний та естетичний дизайн, що робить адміністративний інтерфейс приємним для користувача та відповідає сучасним дизайн-трендам.

- адаптивний шаблон. AdminLTE адаптується до різних розмірів екрану, що робить його корисним для використання на різних пристроях, таких як комп'ютери, планшети та смартфони.

- багатофункціональність. AdminLTE містить багато готових компонентів та елементів інтерфейсу, таких як таблиці, графіки, форми, які можна легко інтегрувати та використовувати для побудови потужного адміністративного інтерфейсу.

- гнучкість та легкість використання. AdminLTE побудований на базі Bootstrap, що робить його легко налаштовуваним та розширюваним. AdminLTE можна швидко адаптувати до своїх потреб та додавати власну функціональність.

- активна спільнота та підтримка. AdminLTE має велику та активну спільноту розробників, що означає підтримку та постійне оновлення. Окрім готового шаблону є можливість легко знайти різноманітні розширення та поради в спільноті.

- безкоштовний та відкритий код. AdminLTE розповсюджується під ліцензією MIT, що дозволяє використовувати його безкоштовно та впроваджувати в проекти.

Вибір AdminLTE дозволяє значно пришвидшити процес розробки UI, мінімізувати кількість проблем з адаптивністю а також отримати готовий та стильний інструмент для швидкого розгортання адміністративного інтерфейс майбутнього веб-додатка.

Отримання даних для AdminLTE буде виконуватись за допомоги Laravel Eloquent. Laravel Eloquent – це ORM, вбудований у фреймворк, який дозволяє взаємодіяти з базою даних через об'єктно-орієнтований синтаксис. Це дозволяє легко виконувати операції з базою даних без необхідності написання складних SQL-запитів.

Eloquent використовує моделі для представлення таблиць у базі даних. Модель – це клас, який співставляється з таблицею у базі даних та дозволяє взаємодіяти з записами цієї таблиці як з об'єктами. В результаті взаємодії буде сформовано і виконано SQL-запит.

Eloquent спрощує інтерфейс взаємодії з базою даних у Laravel-проектах, забезпечуючи ефективні та зручні інструменти для розробників.

3.3 Висновки за розділом 3

В третьому пункті було вибрано мови програмування і фреймворки для розробки системи збору даних інтернет магазинів. Окремо були розглянуті інструменти для збору даних і для побудови UI, з яким безпосередньо буде взаємодіяти користувач.

Використання таких інструментів як Scrapy на Python та Puppeteer на Node.js дозволять збирати дані з більшості існуючих веб-сайтів. При цьому Scrapy підійде як для веб-скрапінгу, так і для збору даних через API, а Puppeteer дозволить повторювати послідовність дій користувача в режимі емуляції браузера. Обидва інструменти взаємодоповнюють один одного і дозволяють реалізувати всі методи збору даних, навіть з урахування можливих змін в майбутньому.

Завдяки сучасному PHP фреймворку Laravel і open source бібліотеці AdminLTE є можливість швидко розробити сучасний і адаптивний UI для налаштування системи та експорту даних. Дуже важливим є великий потенціал використаних інструментів, що дозволить в майбутньому розширяти функціонал системи без необхідності переходу на інші мови програмування або технології.

Слід зауважити, що активний розвиток Laravel і AdminLTE може бути і певного роду недоліком, оскільки для використання сучасних технологій і новітніх можливостей система буде потребувати постійної підтримки і оновлення, а остання версія фремворку може перестати підтримуватись вже через декілька років.

4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ЗБОРУ ДАНИХ

4.1 Розробка інтерфейсу користувача

Розробка інтерфейсу користувача для системи збору даних є важливим етапом в створенні функціонального та ефективного інструменту. Простота та логічність взаємодії з системою дозволяють користувачам швидко вивчити та використовувати всі її функції.

При розробці інтерфейсу важливо враховувати різноманітні завдання, які можуть виконувати користувачі. Інтерфейс повинен бути адаптивним до різних завдань, починаючи від введення даних і закінчуючи їхнім аналізом. Інтуїтивне розміщення елементів та проста навігація сприяють зручності використання системи. Структура класів необхідних для забезпечення функціонування системи представлена у додатку Б.

Використання фреймворку Laravel, шаблонізатора Blade, а також бібліотеки з відкритим доступом AdminLTE дозволяють значно підвищити швидкість розробки за рахунок використання готових компонентів інтерфейсу і навіть цілих сторінок. Прикладом готової сторінки – є сторінка логіну, для реалізації функціонування якої треба написати тільки бекенд частину.

Сторінка логіну представлена на рисунку 4.1, та представляє собою форму з полями для заповнення. Вона забезпечує можливість користувачам увійти в систему. В поточному додатку можливість реєстрації відсутня, тому для першого входу потрібно згенерувати користувача з правами admin, який вже зможе створити інших користувачів.

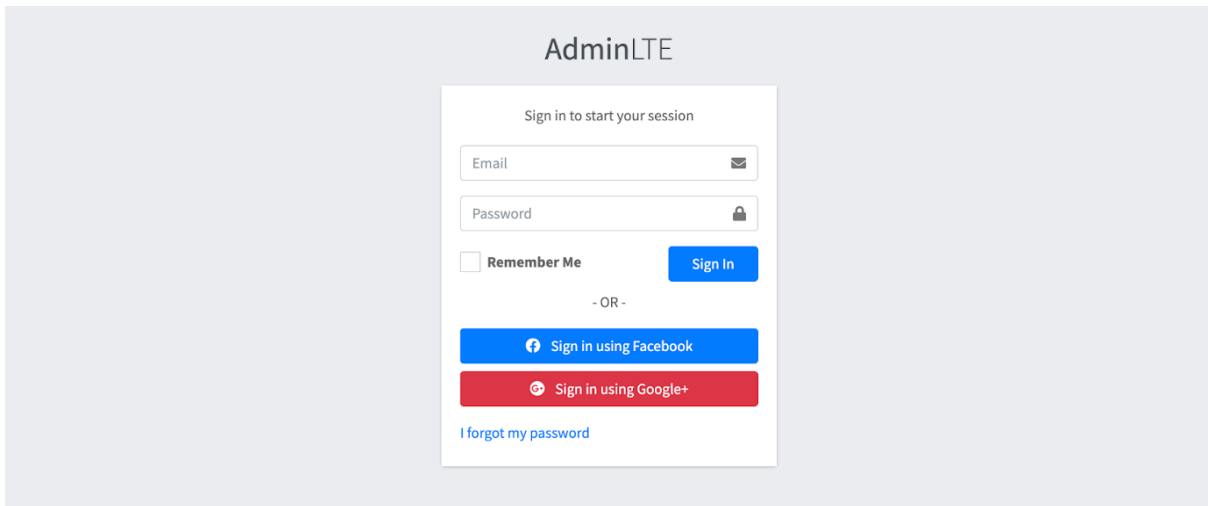


Рисунок 4.1 – Сторінка логіну

Після успішної авторизації користувач бачить сторінку Dashboard (рисунок 4.2), на якій має змогу ознайомитись з деякою статистикою стосовно роботі системи.

Завдяки AdminLTE і вбудованим інструментам візуалізації, таким як Chart.js, є можливість розширити статистику і додати нові типи графіків.

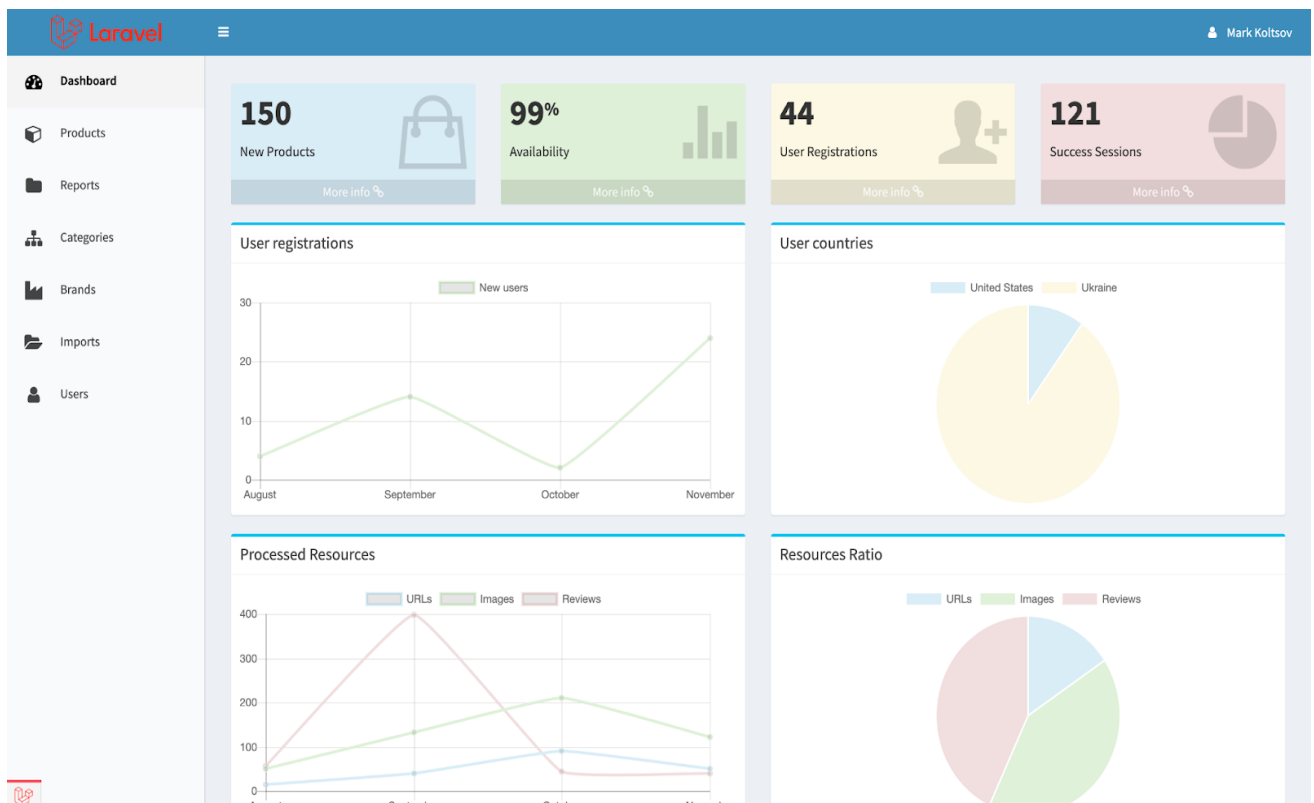
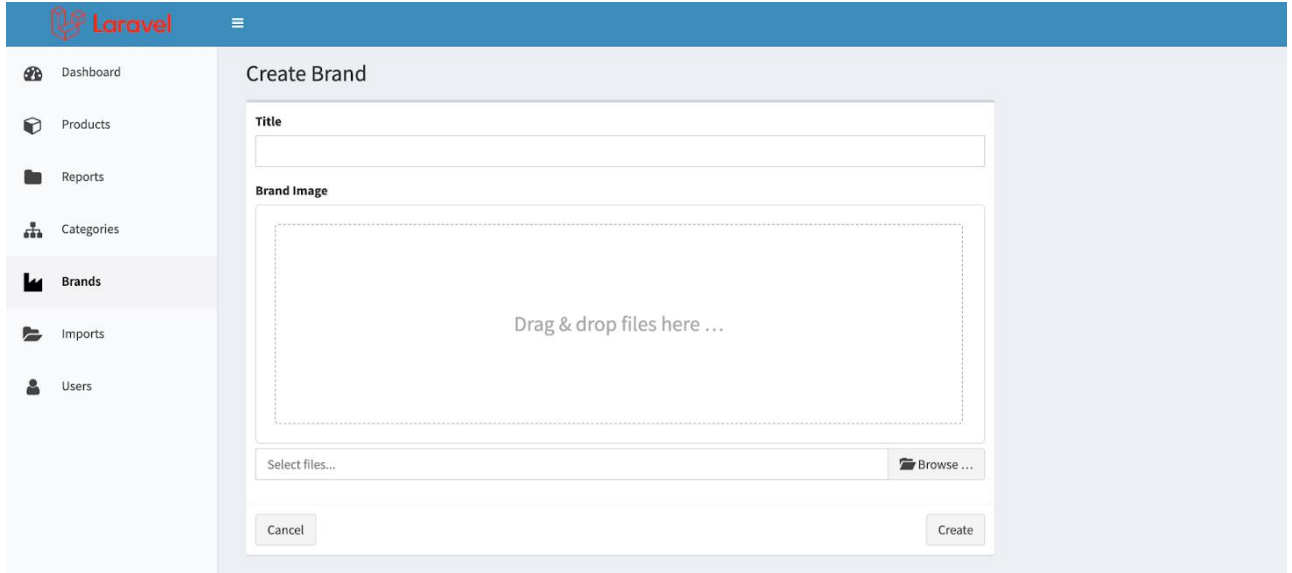


Рисунок 4.2 – Сторінка Dashboard

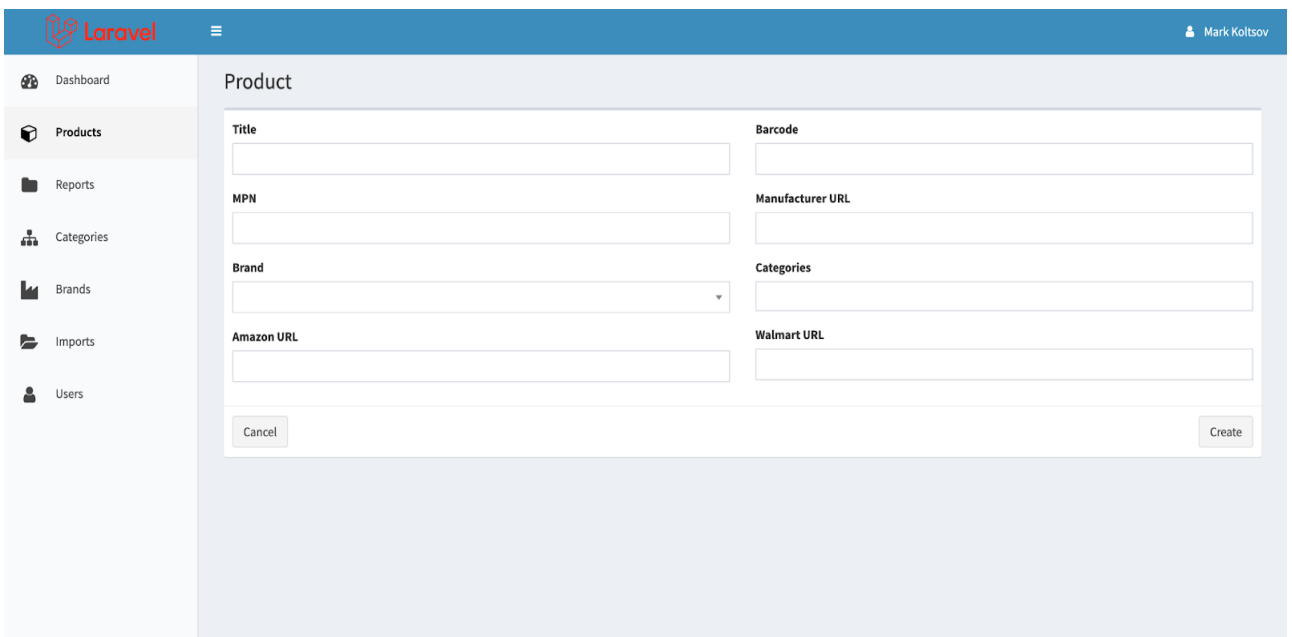
Сторінка створення бренду (рисунок 4.3) дозволяє створювати і налаштовувати бренди продуктів, а зручний редактор з функцією попереднього перегляду дозволить завантажити відповідне зображення бренду.



The screenshot shows the 'Create Brand' form in a Laravel application. The form is located in the main content area, with a sidebar on the left containing navigation links for Dashboard, Products, Reports, Categories, Brands, Imports, and Users. The form itself has a 'Title' input field at the top. Below it is the 'Brand Image' section, which features a large dashed rectangular area for image upload with the text 'Drag & drop files here ...'. Underneath the image area is a 'Select files...' input field and a 'Browse ...' button. At the bottom of the form are 'Cancel' and 'Create' buttons.

Рисунок 4.3 – Сторінка створення бренду

Сторінка створення продукту (рисунок 4.4) дозволяє налаштувати посилання, за якими буде проходити збір інформації а також заповнити базову інформацію про продукт.



The screenshot shows the 'Product' form in a Laravel application. The form is located in the main content area, with a sidebar on the left containing navigation links for Dashboard, Products, Reports, Categories, Brands, Imports, and Users. The form has several input fields: 'Title', 'Barcode', 'MPN', 'Manufacturer URL', 'Brand' (a dropdown menu), 'Categories', 'Amazon URL', and 'Walmart URL'. At the bottom of the form are 'Cancel' and 'Create' buttons. The top right of the page shows the user's name 'Mark Koltsov'.

Рисунок 4.4 – Сторінка створення продукту

Для забезпечення функціонування UI була розроблена структура БД. Її частина представлена на рисунку 4.5, а повна структура надана в додатку А.

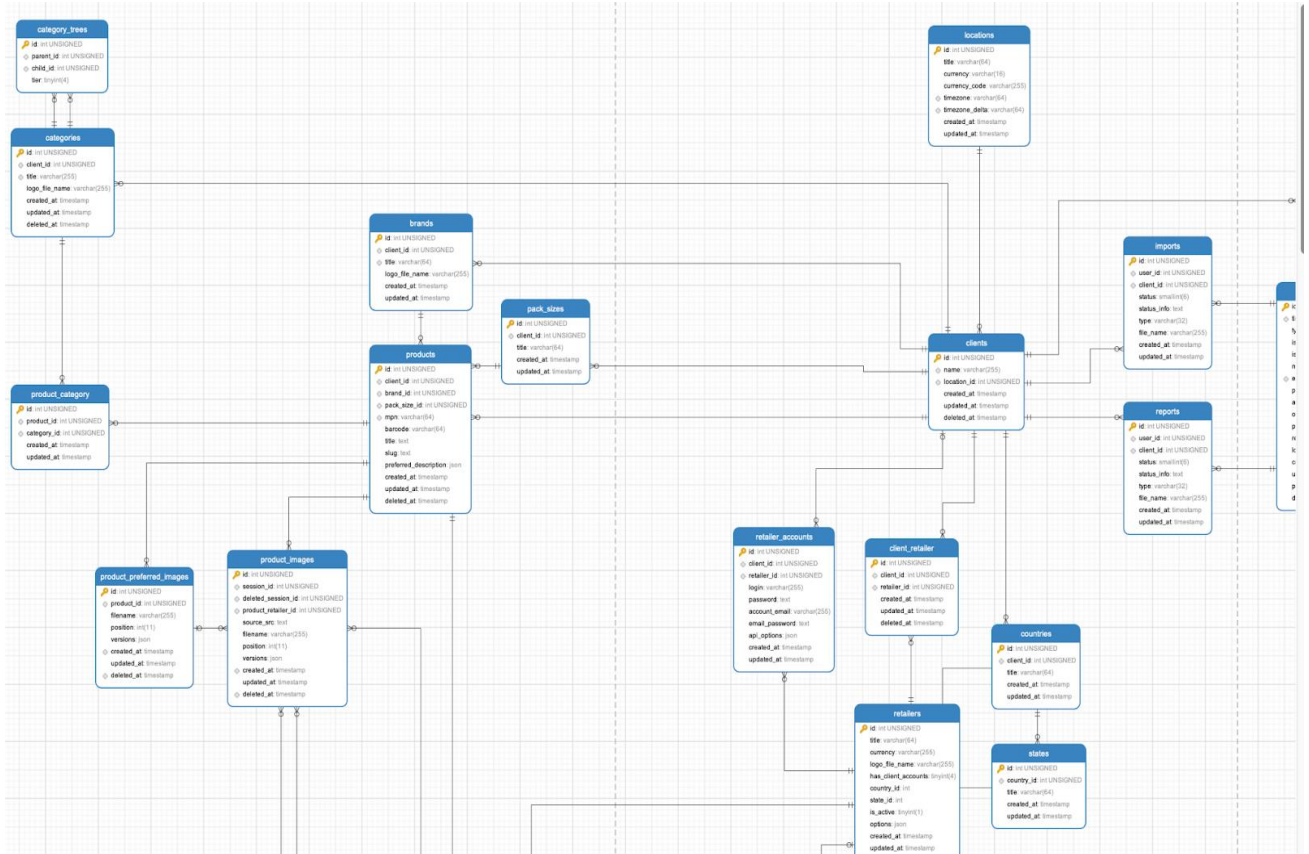


Рисунок 4.5 – Частина структури БД

4.2 Реалізація функціональності для збору даних

Технологія для збору даних з інтернет магазину в першу чергу залежить від технологій використаних для створення інтернет магазину, а також від методів захисту контенту. Складність збору даних може дуже сильно відрізнятись і бути як простим HTTPS запитом без використання додаткових сервісів, так і складним процесом з емуляцією веб браузера і дій користувача для отримання необхідної інформації.

Для реалізації збору даних з популярного інтернет магазину Walmart підійде фреймворк Scrapy, реалізований на мові Python. Процес збору даних з інтернет магазину Walmart можна поділити на частини:

- отримання веб сторінки шляхом виконання HTTP запити;
- пошук, вибірка і підготовка необхідних даних шляхом обробки відповіді запита;
- збереження отриманих даних в БД або передача на відповідний API.

Крок отримання веб сторінки шляхом виконання HTTP запити не має додаткових складностей, крім необхідності використання проксі серверів, які допоможуть уникнути блокування IP-адреси сервера, з якого виконуються запити. Блокування IP-адреси призведе до повної неможливості подальшої роботи з веб сторінкою. Рішенням може стати тільки заміна IP-адреси або використання проксі серверів.

Пошук даних на веб сторінці виконується за допомогою вбудованих інструментів в фреймворк Scrapy, які дозволяють взаємодіяти з DOM структурою веб сайту та шукати необхідні елементи за допомогою селекторів. На рисунку 4.6 можна побачити приклад веб сторінки інтернет магазину Walmart з частиною DOM структури, відповідальної за вміст назви товару. Після визначення ключових позицій, які потрібно зібрати, з'являється можливість визначення необхідних DOM селекторів а також їх ієрархії.

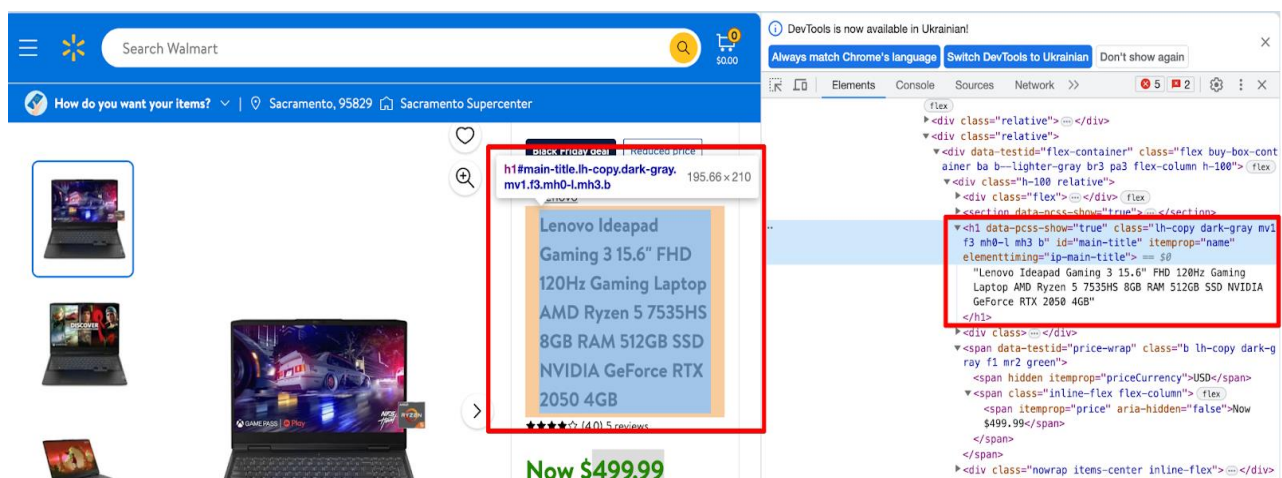


Рисунок 4.6 – Приклад сторінки і DOM структури продукту інтернет магазину

Сучасні веб-сайти часто використовують JavaScript для візуалізації даних, що дозволяє надати веб-сторінкам інтерактивності, але це може стати проблемою для процесу збору даних. При використанні Scrapy – виконується тільки HTTP запит, що означає що будь які JavaScript-и не будуть виконані, це може потребувати додаткових HTTP запитів або додаткової обробки результатів першого запиту.

В випадку з Walmart – деяка інформація може бути отримана з JavaScript, який завантажується разом зі сторінкою продукту. Часто це може бути скрипт з `id="__NEXT_DATA__"`, який містить в собі JSON з інформацією про продукт. Приклад такого скрипту з додатковою інформацією про продукт візуально представлений на рисунку 4.7.

```
> s('${__NEXT_DATA__}')
<script id="__NEXT_DATA__" type="application/json" nonce>
  {"props":{"pageProps":{"initialData":{"data":{"contentLayout":{"modules":{"configs":{"__typename":"TempoM_GLASSMWWFulfillmentConfigs","showWPlusComponent":"False","wPlusSubtitle":"You get with","wPlusTitleColor":"#0071DC","isStickyBuybox":"False"},"moduleId":"89b60386-66c5-4ecd-83a1-fb61899fc008","matchedTrigger":{"pageType":"ItemPageGlobal","pageId":"global_evergreen","zone":"pm"},"type":"Fulfillment","version":"11","status":"published","publishedDate":"1699318859587"},"configs":{"__typename":"TempoM_GLASSMWWBuyBoxAdConfigs","rawConfigs":{"moduleLocation":"top"},"m":{"status":"SUCCESS","moduleType":"BuyBoxAd","platform":"DESKTOP"},"pageId":"5021430193","pageType":"ITEM","storeId":"3081","stateCode":"CA","zipCode":"95829","pageContext":{"itemContext":{"itemId":"5021430193"},"productId":"58EKKK020QVL","abstractProductId":"","categoryPath":"","categoryPathName":"0:3944:1089430:7052607:1849032:3237391","categoryPathName":"Home Page/Electronics/Computers, Laptops Laptop AMD Ryzen 5 7535HS 8GB RAM 512GB SSD NVIDIA GeForce RTX 2050 4GB","brand":"Lenovo","partTypeId":"","manufactureNumber":"825800SLUS","aaaBrand":"","tireSize":"","tireWidth":"","tireRatio":"","tireDiameter":"","wheelDiameter":"","speedRating":"","loadI Laptop-AMD-Ryzen-5-7535HS-8GB-RAM-512GB-SSD-NVIDIA-GeForce-RTX-2050-4GB/5021430193"},"moduleConfigs":{"moduleLocation":"top"},"adsContext":{"locationContext":{"zipCode":"95829","stateCode":"CA","storeId":"3081","pickupStore":"3081","deliveryStore":"3081"},"intent":"SHIPPING","incatchment":true,"itemId":"5021430193","categoryId":"3944_1089430_7052 Gaming Laptops","brand":"Lenovo","productName":"Lenovo Ideapad Gaming 3 15.6\" FHD 120Hz Gaming Laptop AMD Ryzen 5 7535HS 8GB RAM 512GB SSD NVIDIA GeForce RTX 2050 4GB","productId":"710\" {\"type\":\"SPONSORED_PRODUCTS\",\"data\":{\"__typename\":\"SponsoredProducts\",\"adId\":\"1fac97f1-39d9-4c0d-9625-8fc002593943\",\"adExpInfo\":{\"adExpId\":\"2909\",\\ \"flags\":null,\"labels\":null,\"tags\":{\"__typename\":\"BaseBadge\",\"id\":\"L1001\",\"text\":\"Pickup\",\"key\":\"PICKUP\"},\\ \"__typename\":\"BaseBadge\",\"id\":\"L1002\",\"text\":\"Delivery\",\"key\":\"DELIVERY\"},\\ \"type\":{\"rollback\":null,\"reducedPrice\":null,\"eligibleForAssociateDiscount\":true,\"clearance\":null,\"strikethrough\":null,\"submapType\":null,\"priceDisplayAmount\":null,\"unitOfMeasure\":null,\"pricePerIn {\"price\":229,\"priceString\":\"$229.00\",\"priceDisplay\":\"$229.00\"},\"wasPrice\":null,\"listPrice\":null,\"priceRange\":null,\"unitPrice\":null,\"savingsAmount\":null,\"comparisonPrice\":null,\"subscriptionPr {19}UOKU4xdYQ5CTV53\",\"clickBeacon\":{\"https://wrd.walmart.com/track?adId=1fac97f1-39d9-4c0d-9625-8fc002593943&u0026pgid=5021430193&u0026spqs=kd5CKW7sJgdW0B6bn1LKmR6vKwG8VFX_AlB_UyIqzKn0gP19UOKU4xdYQ5CTV53&u0026storeId=3081&u0026pt=Item&u0026mLoc=Item&u0026mLnt=plmt&u0026mLnt=plmt&u0026eventSt=EventST&u0026pos=_pos_&u0026bt=_bt_ Celeron-N4500-4GB-RAM-64GB-eMMC-Arctic-Grey-82N4002HUS/951435754\",\"numberOfReviews\":116,\"averageRating\":4.1,\"availabilityStatus\":\"IN_STOCK\",\"imageInfo\":{\"thumbnailUrl\":\"https://i5.walmartna 96fe-45121408370.965a3f54d8347947e6f23ebd3ec1dfd.jpeg\",\"allImages\":null},\"name\":\"Lenovo Ideapad 3i Chromebook, 15.6\" FHD, Intel Celeron N4500, 4GB RAM, 64GB eMMC, Arctic Grey, 82N4002HUS {\"predictedQuantity\":null,\"flags\":null,\"labels\":null,\"brand\":\"Lenovo\"}}},{\"isStickyBuybox\":null,\"moduleId\":\"ddc1943c-88b6-4184-b5ba-7b932e1c1cd3\",\"matchedTrigger\":{\"pageType\":\"ItemPageGlo dweb\",\"type\":\"BuyBoxAd\",\"version\":\"2\",\"status\":\"published\",\"publishedDate\":\"1657668297781\",\"configs\":{\"__typename\":\"TempoM_GLASSMWMarqueeDisplayAdConfigs\",\"rawConfigs\":{\"moduleLocation\":\"ma {\"pageType\":\"ItemPageGlobal\",\"pageId\":\"global_evergreen\",\"zone\":\"contentZone68\",\"inheritable\":false,\"name\":\"Marquee Display Ad 1 RWeb\",\"type\":\"MarqueeDisplayAd\",\"version\":\"27\",\"status\":\"publi {\"moduleLocation\":\"brandbox\",\"lazy\":\"9800\"},\"isStickyBuybox\":null,\"moduleId\":\"9e74cfd6-6c3d-40ee-a762-4311ff6d02ef\",\"matchedTrigger\":{\"pageType\":\"ItemPageGlobal\",\"pageId\":\"global_evergreen Module\",\"type\":\"BrandBoxDisplayAd\",\"version\":\"2\",\"status\":\"published\",\"publishedDate\":\"1664844924483\",\"configs\":{\"__typename\":\"TempoM_GLASSMWWAddToCartConfigs\",\"rawConfigs\":{\"isStickyBuybox 3da27f1feded\",\"matchedTrigger\":{\"pageType\":\"ItemPageGlobal\",\"pageId\":\"global_evergreen\",\"zone\":\"buyBoxZone13\",\"inheritable\":false,\"name\":\"DESKTOP New Add To Cart Module, Monday, March 13, 2 {\"__typename\":\"TempoM_GLASSMWMCapitalOneBannerConfigsV1\",\"bannerBackgroundColor\":\"#FFFFFFF\",\"primaryImage\":{\"alt\":\"CapitalOne\",\"src\":\"https://i5.walmartimages.com/dfw/4ff9c6c9-4cbf/k2-19b3c Hard_May12_2021_Teflon_asset.png\"},\"bannerCta\":{\"ctaLink\":{\"LinkText\":\"Learn more\",\"title\":\"Learn more\"},\"clickThrough\":{\"value\":\"/cp/6324027atpdc=c359f401-b3f0-4e3b-8496- 40c1D701199&u0026thgpid=AthenaItempage&u0026thgnid=AthenaModuleZone&u0026thmtid=AthenaCapitalOneBanner&u0026thvid=6u0026ththnata=true&u0026thcaponeid=1\"},\"uid\":\"jilt6-2X\"},\"textColor\": Walmart.com,\"isBold\":true,\"isUnderlined\":true,\"underlinedColor\":\"#0071DC\",\"textColor\":\"#2e2f32\"},\"text\":\"See if you're pre-approved with no credit risk.\"},\"isBold\":false,\"isUnderline {\"pageType\":\"ItemPageGlobal\",\"pageId\":\"global_evergreen\",\"zone\":\"contentZone30\",\"inheritable\":false,\"name\":\"dWeb Capital One Banner Web Module - Desktop - 4.29.21\",\"type\":\"CapitalOneBanner\" {\"__typename\":\"TempoM_GLASSMWWItemDigitalRewardsConfigs\",\"rawConfigs\":{}},\"moduleId\":\"61361e30-e96b-44e8-942f-64bdaf713a8c\",\"matchedTrigger\":{\"pageType\":\"ItemPageGlobal\",\"pageId\":\"global_e 2023, 9:14:11:88 pm\",\"type\":\"ItemDigitalRewards\",\"version\":\"2\",\"status\":\"published\",\"publishedDate\":\"170075951669\",\"configs\":{\"__typename\":\"TempoM_GLASSMWWSubscriptionOptionsModuleConfigs\", \"2c8aaf1372_v1.png\",\"height\":\"192\", \"assetId\":\"eds80f20-c373-1ed-abc0-3132441c68e\",\"src\":\"https://i5.walmartimages.com/dfw/4ff9c6c9-9f35/k2-8e7dd129-8704-4b72-bf43-904eb658bcca_v1.png\"},\"w icon\",\"uid\":\"ydxs7Mde\",\"primaryTitle\":\"New!\", \"secondaryBoldedTitle\":\"More savings, less stress.\"},\"subTitle\":\"Subscribe to get what you need, when you need it. Cancel anytime.\"},\"scriptio {\"type\":\"url\",\"value\":\"https://www.walmart.com\",\"rawValue\":\"https://www.walmart.com\"},\"moduleId\":\"0f596623-b845-4bc8-b848-6f8cfc615c2f\",\"matchedTrigger\":{\"pageType\":\"Item Options Module Module, Wednesday, March 15, 2023, 1:58:05:80 pm\",\"type\":\"SubscriptionOptionsModule\",\"version\":\"2\",\"status\":\"published\",\"publishedDate\":\"170076005921\",\"configs\":... </script>
```

Рисунок 4.7 Додаткова інформація про продукт на сторінці Walmart

За допомогою обробки DOM структури сторінки, а також JSON інформації зі скрипта `id="__NEXT_DATA__"`, з'являється можливість отримати всю необхідну інформацію про товар. Обробка представляю собою парсинг скрипту та його розбір на окремі частини. Приклад коду такої обробки представлено на рисунку 4.8.


```

def parse_item(self, response):
    product = ProductItem()
    product['id'] = response.meta.get('product_id')
    product['url'] = response.meta['product_url']
    product['name'] = response.css('h1[itemprop="name"]::text').get().strip()
    product['price'] = response.css('span[itemprop="price"]::text').get().strip()
    product['currency'] = response.css('span[itemprop="priceCurrency"]::text').get().strip()

    match = re.compile(r'/(\\d+)?').search(product['url'])
    if match:
        product['code'] = match.group(1)

    xpath = '''
        //div[@data-testid="vertical-carousel-container"]
        //div[@data-testid="media-thumbnail" and not(./*[@data-testid="ui-video-play-button-testid"])]//img/@src
    '''
    images = response.xpath(xpath).getall()
    product['images'] = [i.split('?')[0] for i in images]

    additional_data = json.loads(response.css('#__NEXT_DATA__::text').get())

    product['short_description'] = additional_data['props']['pageProps']['initialData']['data']['idml']['shortDescription']
    product['long_description'] = additional_data['props']['pageProps']['initialData']['data']['idml']['longDescription']

    availability = additional_data['props']['pageProps']['initialData']['data']['product']['availabilityStatus']
    product['stock'] = int(availability == "IN_STOCK")

    yield product

```

Рисунок 4.8 – Приклад коду для збору інформації про продукт з інтернет магазину Walmart за допомогою фреймворка Scrapy

Інтернет магазин Amazon має інші способи захисту сторінки, ніж Walmart, також багато контенту формується за допомогою JavaScript, що робить неможливим отримати всю інформацію про продукт завдяки HTTP запиту. Підвищена складність збору інформації вимагає використання додаткових інструментів, таких як відкрита бібліотека Puppeteer, реалізована на Node.js.

Процес збору даних з інтернет магазину Amazon можна поділити на частини:

- отримання веб сторінки шляхом емуляції браузера;
- пошук, вибірка і підготовка необхідних даних шляхом обробки відповіді запити;
- збереження отриманих даних в БД або передача на відповідний API.

Головною відмінністю від скрапінгу Walmart є перший крок, в якому потрібно виконати емуляцію браузера і вже потім збирати інформацію завдяки вбудованим інструментам в Puppeteer для роботи з DOM. Приклади структури і коду наведені на рисунках 4.9 і 4.10.

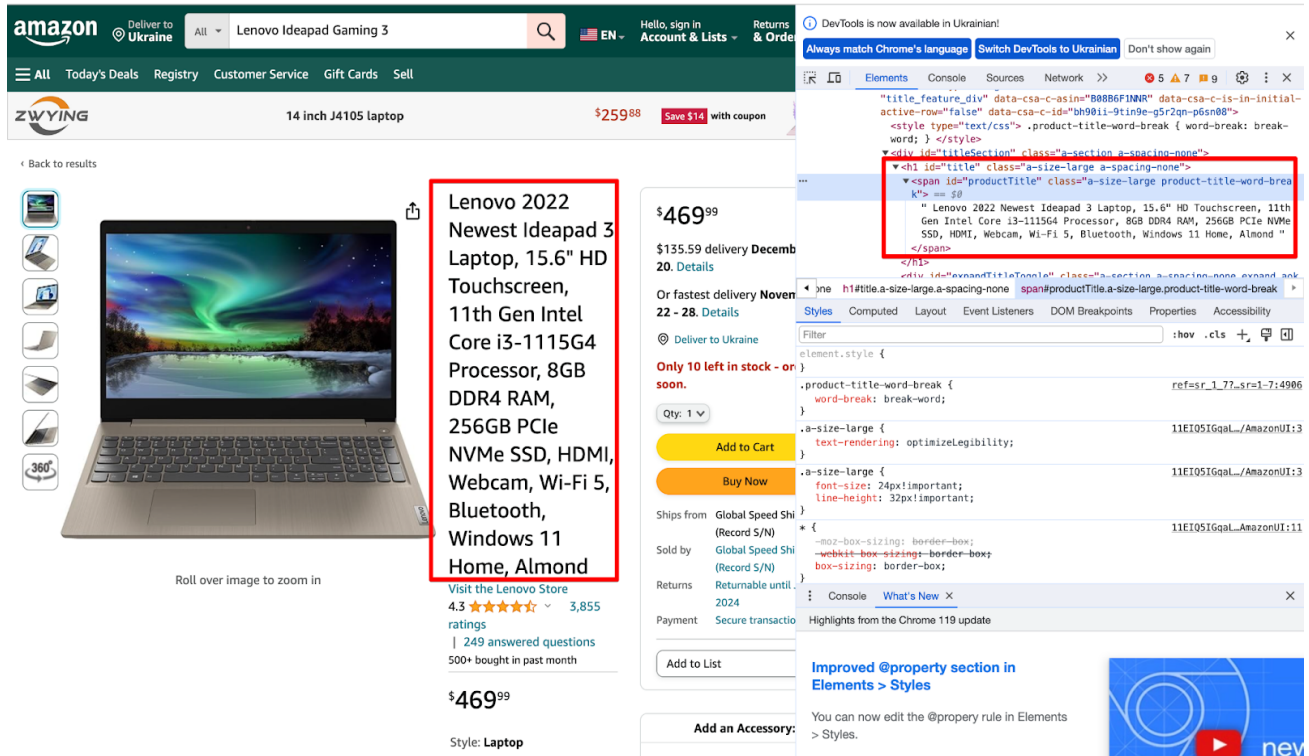


Рисунок 4.9 – Приклад сторінки і DOM структури продукту інтернет магазину Amazon

```

async parseTitle(_response) : Promise<any> {
  return this.page.evaluate(
    () :... => {
      let title : Element = document.querySelector(selectors: 'span#productTitle');

      if (title !== null && typeof title.innerText === 'string') {
        return title.innerText.trim();
      }

      return '';
    }
  );
}

```

Рисунок 4.10 – Приклад коду для збору інформації про заголовок продукту з інтернет магазину Amazon за допомогою Puppeteer

Окрім головної інформації про продукт є можливість збирати відгуки покупців про товар (рисунок 4.11). Така інформація може допомогти виробникам продуктів покращити товар, а продавцям коригувати ціну і плани продажів.

The screenshot shows the Walmart website interface. At the top, there's a search bar and navigation links like 'Departments', 'Services', 'Reorder My Items', and 'Sign In Account'. Below the navigation, there are category tabs: 'Deals', 'Grocery & Essentials', 'Thanksgiving', 'The Holiday Shop', 'Gift Finder', 'Electronics', 'Toy Shop', and 'Home'. The main content area is divided into two columns. The left column displays several customer reviews for a product, each with a star rating, a 'Verified Purchase' badge, and the review text. The right column shows the product details, including the current price of \$598.00 (reduced from \$699.00), a payment option of \$31/mo with affirm, and a blue 'Add to cart' button. Below the price, there are specifications for 'Hard Drive Capacity' and 'RAM Memory'. At the bottom of the right column, there are options for 'Shipping', 'Pickup', and 'Delivery', along with seller information for 'VIPOUTLET' and 'Extended holiday returns'.

Рисунок 4.11 – Відгуки товару інтернет магазину Walmart

Зібрані відгуки можуть принести більше користі, якщо вони будуть проаналізовані, відсортовані і розбиті на ключові слова завдяки Amazon Comprehend. Amazon Comprehend – це служба обробки природної мови (NLP), яка використовує машинне навчання для аналізу текстів (рисунок 4.12).

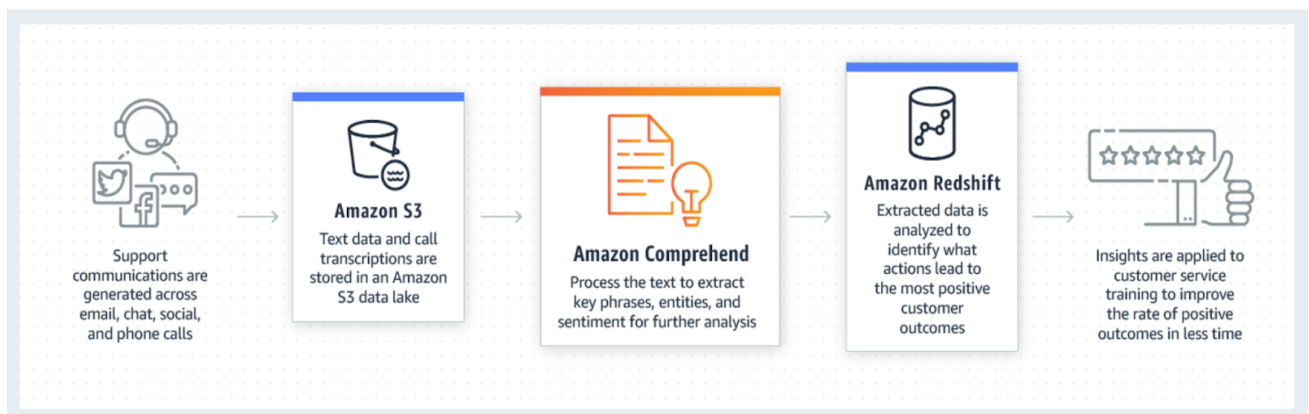


Рисунок 4.12 – Приклад взаємодії з Amazon Comprehend

Робота в Amazon Comprehend може проходити як з вже натренованими моделями, які пропонує AWS, так і з новими моделями, навчання яких буде проходити на ресурсах AWS, на основі даних завантажених користувачем.

Прикладом роботи з Amazon Comprehend може стати аналіз одного за відгуків користувача про товар на моделі, яку тренував AWS. Параметри запиту наступні:

```
{
  "Text": "Are you perhaps looking for a portable, entry-level solution to pc gaming while forgoing the temptation of buying a Steam deck OLED? Look no further! For $518.00 (the amount I paid when it went on sale), I can safely say it was worth every penny: -Pros- -It's light weight... -It's the best iteration (or variant) I've seen, coming in with the ridiculously, industrial standard of 16gb of system memory %26 a RTX 3050, which is much better than 8gb of ram %26 better than (on the graphic card side) the RTX 2050 %26 GTX 1650 variants... -It's fast! Like, really fast! To me, anyway. I only upgrade every five to seven years, so this experience for me is a brave new world type thing... -It's display is 1080p @ 144hz (Upon mentioning this, I'm stressing upon the hertz to be the benefit here rather than the resolution...) -Cons- -The overall construction feels cheap... Although I don't mind it since I paid a little over $500.00 for it. -I recommend not playing unplug, for the battery life isn't that good %26 you lose fps... -The GPU (3050) inside is only worth 4gb of vram, which isn't great. -The touchpad is... meh... But I don't mind it... Overall, I'm very much pleased with it. Back in the day, if you wanted to play any AAA game at a reasonable setting, you'd have to shell out at least a grand. This review (and the positivity thereof) is really centered about getting this sucker at the price I did or even lower. Any higher %26 I'd say it isn't worth it. When I first saw this deal, it was better %26 exclusive for Walmart plus members %26 it sold out quick %26 for a good reason. I lucked out with the second batch which saw a slight price increase. On the whole, I'm happy.",
  "LanguageCode": "en"
}
```

Аналіз настрою (Sentiment Analysis) – це сервіс, який використовує машинне навчання для визначення тону тексту, а саме чи відбувається у тексті позитивний, негативний чи нейтральний відгук. AWS Comprehend може виявляти емоції та ступінь впевненості у висловлюваннях. Для використання аналізу настрою в AWS Comprehend, ви можете відправити текстові дані на обробку за допомогою API-виклику або використовувати консоль AWS Comprehend.

Процес аналізу настрою включає в себе наступні етапи:

- введення тексту – передача тексту для аналізу до сервісу Comprehend;
- аналіз настрою. Comprehend використовує алгоритми машинного навчання для визначення того, наскільки текст виражає позитивний, негативний або нейтральний настрій. Також, він може надати інформацію про емоції, такі як радість, сум, гнів тощо;
- вивід – результат аналізу повертається у вигляді структурованої відповіді, де вказано виявлений настрій та, можливо, емоції.

Нижче наведено приклад аналізу відгуку:

```
{
  "Sentiment": {
    "Sentiment": "POSITIVE",
    "SentimentScore": {
      "Positive": 0.8915513157844543,
      "Negative": 0.0081268809735775,
      "Neutral": 0.006918027065694332,
      "Mixed": 0.09340381622314453
    }
  }
}
```

Розбиття на ключові фрази (Keyphrase Extraction) – Comprehend може робити визначення та виділення найбільш важливих та семантично значущих

фраз у тексті у межах процесу розбиття на ключові фрази. Ця функція допомагає розуміти головні теми або поняття, які взято за ключові, у тексті. Для відгуків про товар таке розбиття може виділити ключові слова або фрази, які найкраще описують основні враження чи коментарі користувачів. Це може допомогти у визначенні загальної тенденції або головних ідей, які виражені у тексті.

Процес розбиття на ключові фрази схожий з процесом аналізу настрою, окрім другого етапу, де замість аналізу настрою йде розбиття на ключові слова.

Результат розбиття на ключові фрази може містити сотні елементів. У прикладі нижче наведена частина результату обробки відгуку:

```
{
  "KeyPhrases": [
    {
      "Score": 0.6280012726783752,
      "Text": "pc gaming",
      "BeginOffset": 64,
      "EndOffset": 73
    },
    {
      "Score": 0.9999241828918457,
      "Text": "the temptation",
      "BeginOffset": 89,
      "EndOffset": 103
    },
    {
      "Score": 0.9939930438995361,
      "Text": "a Steam deck OLED",
      "BeginOffset": 114,
      "EndOffset": 131
    },
    {

```

```
"Score": 0.9998616576194763,  
"Text": "the amount",  
"BeginOffset": 163,  
"EndOffset": 173  
}  
]  
}
```

4.3 Висновки за розділом 4

В четвертому пункті було реалізовано інтерфейс користувача, а також безпосередньо систему збору даних інтернет магазинів, що відповідає всім заявленим функціональним вимогам до системи.

Розроблена структура БД дозволяє зберігати зібрані дані, а також містить конфігурації системи, необхідні для її роботи. Завдяки розробленому UI є можливість взаємодії з системою, при цьому ця взаємодія може бути як з мобільних пристроїв, планшетів, так і з персональних комп'ютерів, а завдяки скраперам є можливість отримувати свіжі дані з таких інтернет магазинів як Walmart та Amazon.

Використання таких інструментів як Amazon Comprehend надає потужні можливості для подальшої обробки і аналізу зібраних даних, наприклад аналіз настрою в відгуках можна використати для моніторингу задоволеності клієнтів від товару або підвищення рівня продажів завдяки маркетинговим кампаніям.

Загалом зібрані дані можливо використовувати з різними цілями, наприклад для аналізу і регулювання цін, аналізу популярності товарів в різних інтернет магазинах, а також сезонних трендів.

ВИСНОВКИ

У процесі кваліфікаційної роботи було розроблено систему збору даних інтернет магазинів. Для досягнення поставленої мети було вирішено наступні завдання:

- проведено огляд існуючих систем збору даних;
- розглянуто основні методи і етичні аспекти збору даних;
- проведено аналіз існуючих інструментів для веб розробки та збору даних;
- сформульовано функціональні та технічні вимоги до системи;
- розроблено архітектуру системи та описано взаємодію її елементів;
- реалізовано систему збору даних інтернет магазинів.

Розроблена система відповідає всім заявленим функціональним та технічним вимогам а також має такі переваги як:

- зручний, сучасний та адаптивний UI;
- можливість швидкого доступу до важливої статистики;
- високий рівень захисту;
- високий рівень доступності;
- можливості для розширення функціональності, як шляхом додавання нового функціоналу так і нових інтернет магазинів.

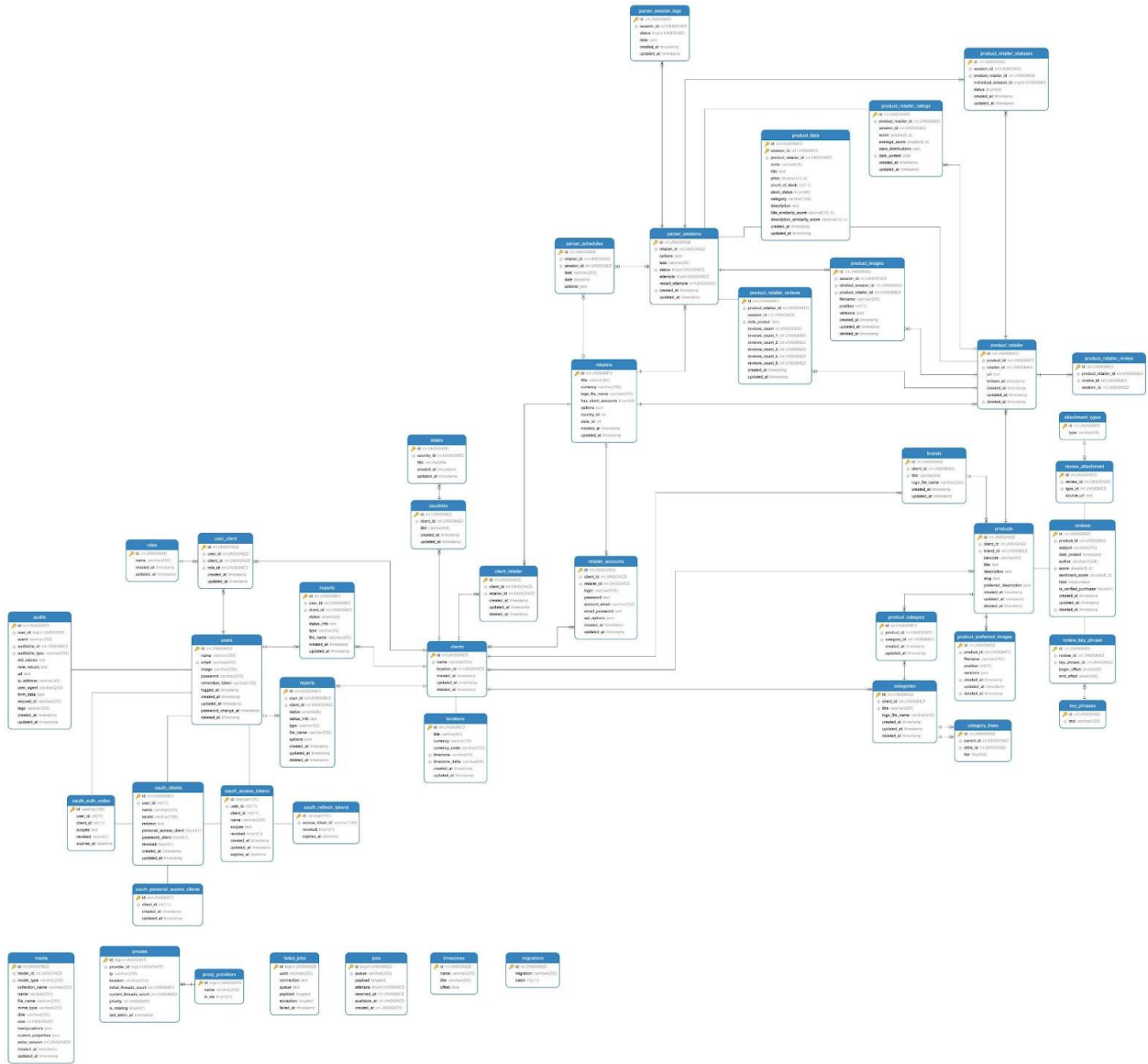
Розроблена система може використовуватись в інтернет комерції для забезпечення точною та актуальною інформацією про стан ринку та продуктів у інтернет-магазинах, а також як інструмент підвищення продажу конкретних брендів, груп товарів і окремих продуктів. Можливість використання мови запитів SQL дозволяє побудувати безліч запитів для відділу аналітики, з дуже потужними можливостями групування і фільтрації зібраних даних, які будуть обмежені тільки потужностями сервера, що використовується.

ПЕРЕЛІК ПОСИЛАНЬ

1. Офіційна документація PHP. URL: <https://www.php.net/> (дата звернення: 20.06.2023).
2. Офіційна документація Laravel. URL: <https://laravel.com/> (дата звернення: 05.07.2023).
3. Офіційна документація AdminLTE. URL: <https://adminlte.io/> (дата звернення: 06.07.2023).
4. M.Stauffer. Laravel: Up and Running. URL: <https://laravelupandrunning.com/> (дата звернення: 06.07.2023).
5. Adel F. Architecture of complex web applications: With examples in Laravel (PHP). Independently published, 2019.
6. Офіційна документація Python. URL: <https://www.python.org/> (дата звернення: 19.07.2023).
7. Офіційна документація Scrapy. URL: <https://scrapy.org/> (дата звернення: 21.07.2023).
8. Офіційна документація Node.js. URL: <https://nodejs.org/en/docs/> (дата звернення: 05.08.2023).
9. Офіційна документація Puppeteer. URL: <https://pptr.dev/> (дата звернення: 10.08.2023).
10. Redmond Paul. Lumen Programming Guide: Writing PHP Microservices, REST and Web Service APIs. Apress, 2016.
11. Pettit T., Cosentino S. The MySQL Workshop: A practical guide to working with data and managing databases with MySQL. Birmingham: Packt Publishing, 2022.
12. Офіційна документація AWS. URL: <https://aws.amazon.com/> (дата звернення: 7.09.2023).

ДОДАТОК А

Структура бази даних



ДОДАТОК Б

Структура класів

