поширеності порушень доступності цей напрям набуває чітко вираженої практичної значущості.

Список використаних джерел

1. Web Content Accessibility Guidelines (WCAG) 2.2. *W3C Recommendation.* URL: https://www.w3.org/TR/WCAG22/ (дата звернення: 29.11.2025).

2. The 2025 Report on the Accessibility of the Top 1,000,000 Home Pages. *The WebAIM Million.* URL: https://webaim.org/projects/million/ (дата звернення: 30.11.2025).

3. Machine Learning Driven Adaptive Accessibility Interfaces for Enhanced User Experience / S. Patel, R. Mendoza, I. Cruz, N. Kim, P. Singh, A. Martinez. URL: https://doi.org/10.13140/RG.2.2.12545.36965 (дата звернення: 27.04.2026).

4. Deep Learning-Enhanced Adaptive Interface for Improved Accessibility in E-Government Platforms / X. Li; Y. Wang et al. *Proceedings of the 6th International Conference on Frontier Technologies of Information and Computer (ICFTIC 2024).* China, 2024. P. 1692–1695. URL: https://doi.org/10.1109/ICFTIC64248.2024.10912984 (дата звернення: 27.04.2026).

УДК 004.428.2

Дикий Д. С., здобувач 2 курсу
ОС «Магістр» спеціальності
F3 Комп'ютерні науки,
Потапова Н. А., канд. екон. наук, доцент,
доцент кафедри інформаційних технологій

ВЕБПОРТАЛ УПРАВЛІННЯ ЛОГІСТИКОЮ МАГАЗИНУ КОМПЛЕКТУЮЧИХ СМАРТСИСТЕМ

Донецький національний університет імені Василя Стуса, м. Вінниця

Сучасні інформаційні системи дедалі частіше будуються на основі принципів гнучкості, масштабованості та інтегрованості, оскільки традиційні програмні рішення вже не здатні ефективно задовольняти потреби бізнесу в умовах постійного зростання обсягів даних та високої варіативності користувацьких сценаріїв. Особливо це актуально для систем, що забезпечують підтримку логістичних процесів, де швидкість обробки інформації, стабільність роботи та можливість адаптації до змін є критично важливими.

Створення вебпорталу для автоматизації логістики магазину комплектуючих для смартсистем потребувало формування архітектури, яка б поєднувала високу продуктивність, простоту супроводу та можливість гнучкого масштабування. Важливо було забезпечити чітке структурування компонентів, стабільність обміну даними, а також здатність до інтеграції зі сторонніми сервісами та інструментами.

Архітектура розробленого вебпорталу ґрунтується на принципах клієнт-серверної взаємодії та патерні Model-View-Controller (MVC), що дає змогу чітко

розмежувати рівні даних, бізнес-логіки та інтерфейсу користувача. Такий підхід суттєво підвищує керованість системи, спрощує тестування і забезпечує можливість швидкої модернізації окремих її компонентів. На рис. 1 зображено загальний підхід до MVC-моделі.

Серверна частина порталу реалізована на Java з використанням фреймворка Spring, який надає розвинені засоби інверсії керування, гнучку конфігурацію та ефективну роботу з HTTP-запитами. Керування залежностями здійснюється за допомогою Gradle, що пришвидшує процес складання та полегшує інтеграцію сторонніх модулів. Для роботи з даними застосовано ORM-технологію Hibernate, яка забезпечує абстракцію над реляційною моделлю та мінімізує кількість ручних SQL-операцій.

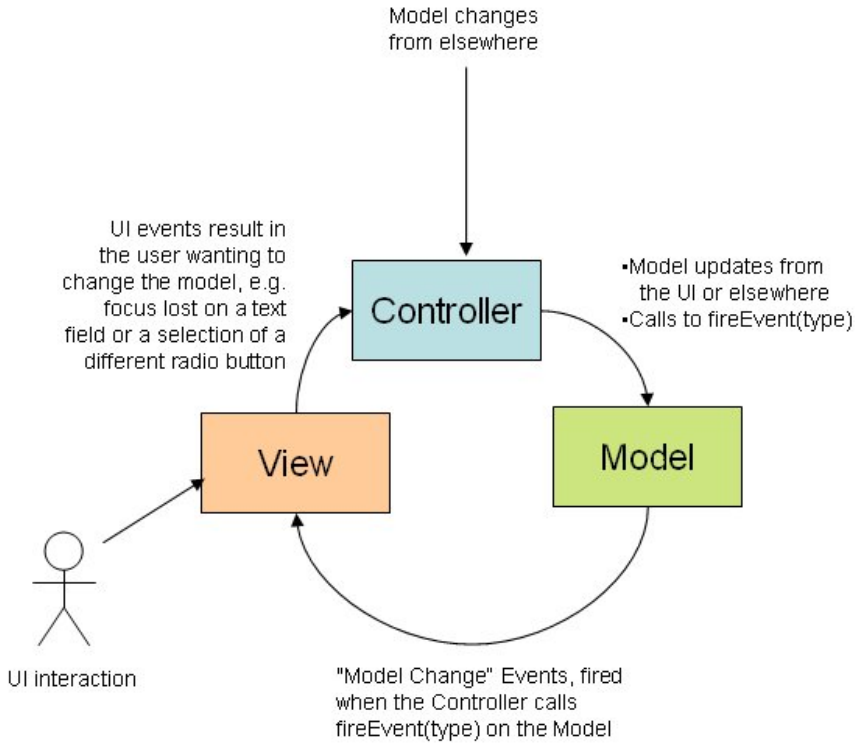

Рисунок 1 – Загальний підхід до побудови MVC-моделі

Контроль версій реалізовано через Liquibase, що гарантує узгодженість змін між середовищами розробки, тестування й продакшену. Основою для зберігання даних обрано MySQL як продуктивну та масштабовану СУБД. На рис. 2 зображено загальну архітектуру системи з поєднанням усіх її компонентів у єдине ціле.

Клієнтська частина побудована з використанням HTML, CSS та JavaScript, що дає змогу створити адаптивний та інтуїтивно зрозумілий інтерфейс. Така фронтенд-архітектура забезпечує зручну взаємодію користувача із системою, швидку обробку подій та мінімальні затримки під час навігації вебпорталом. Для підвищення продуктивності реалізовано кешування даних за допомогою Caffeine Cache, що значно скорочує час відповіді сервера та зменшує навантаження на базу даних.

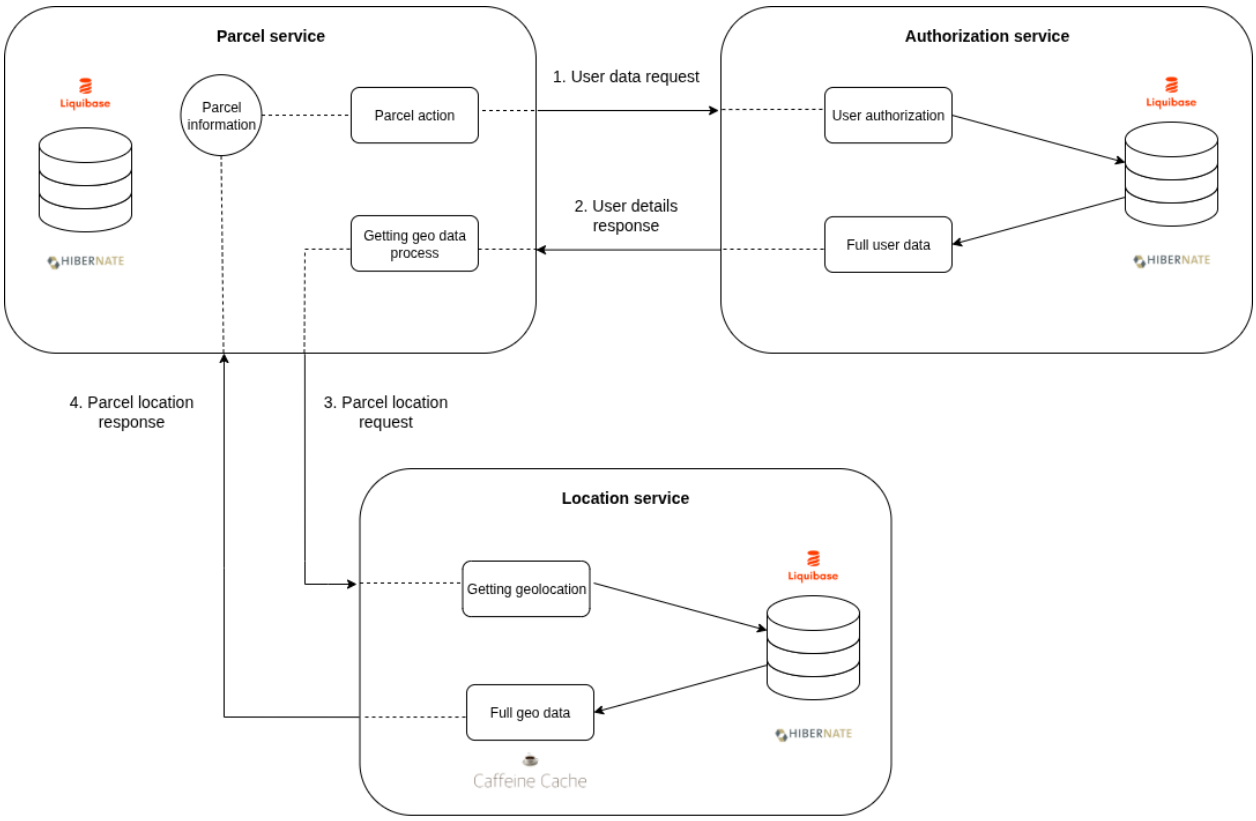

Рисунок 2 – Загальна архітектура системи

Опрацювання операцій із посилками в межах сервісу охоплює повний цикл взаємодії з логістичними даними – від формування нового відправлення до його моніторингу, редагування параметрів та можливого скасування. Система забезпечує узгоджену обробку всіх етапів руху посилок, що є критично важливим для підтримання безперебійної логістики компонентів смартсистем. Для наочного подання структури та взаємодій між основними елементами сервісу на рис. 3 представлено UML-діаграму, яка демонструє логіку виконання дій з посилками та описи відповідних операцій.

Реалізація серверної частини на базі Java та Spring, використання Hibernate і Liquibase для роботи з даними, а також інтеграція MySQL як основної СУБД, дали змогу сформувати стабільну платформу для обробки та зберігання інформації. Застосування кешування Caffeine забезпечило швидший доступ до даних і зниження навантаження на систему.

Клієнтська частина, розроблена з використанням HTML, CSS та JavaScript, забезпечила інтуїтивно зрозумілий та зручний для користувача інтерфейс, що суттєво підвищує ефективність роботи з системою. Модель обробки дій із посилками охоплює повний цикл логістичних операцій, забезпечуючи коректність та узгодженість усіх етапів переміщення товарів.

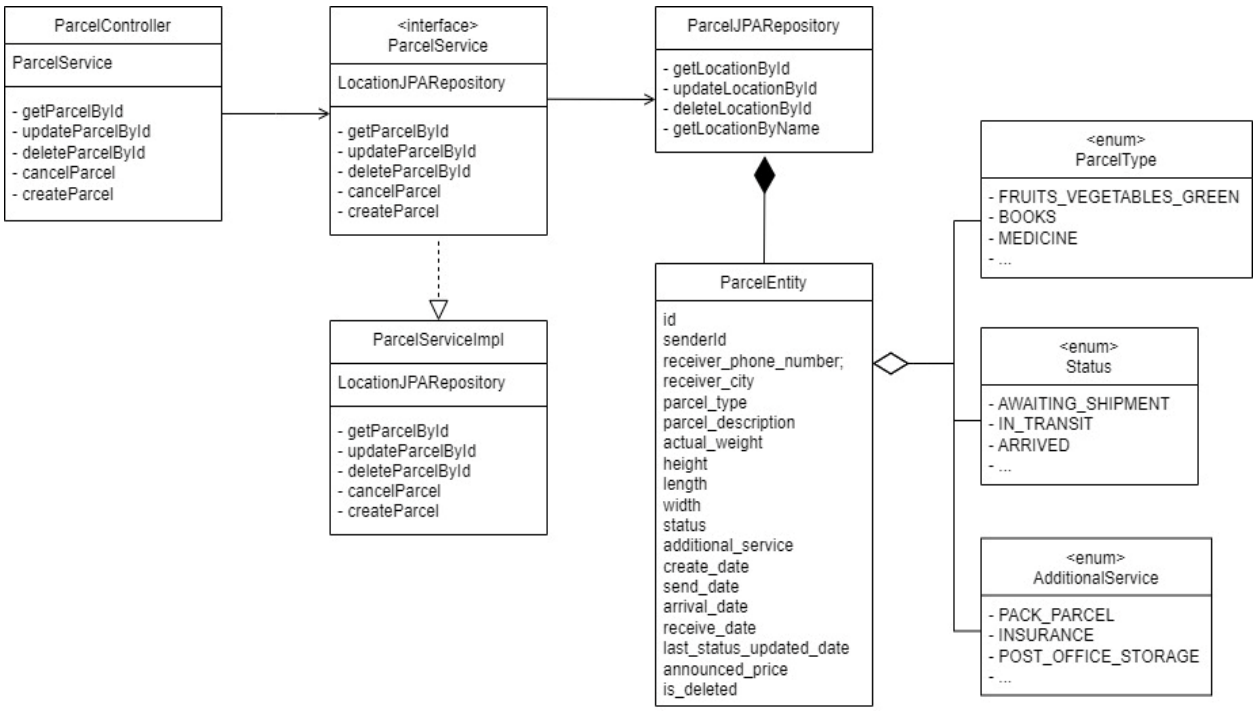

Рисунок 3 – Логіка виконання дій з посилками

Отже, за результатами проведеного аналізу та розроблення вебпорталу для автоматизації логістики компонентів смартсистем було створено функціональну, масштабовану та технологічно сучасну інформаційну систему. Запропонована архітектура, заснована на принципах клієнт-серверної взаємодії та патерні MVC, довела свою ефективність у контексті забезпечення чіткого розподілу відповідальностей, підвищення продуктивності та спрощення супроводу програмного забезпечення.

Список використаних джерел

1. Freeman E., Robson E. Head First Design Patterns. 2nd edition. O'Reilly, 2020. P. 312–333.

2. MySQL Indexes. URL: https://www.geeksforgeeks.org/mysql/mysql-indexes/ (дата звернення: 01.12.2025).

3. MVC. URL: https://developer.mozilla.org/en-US/docs/Glossary/MVC (дата звернення: 01.12.2025).

4. Guide to Caching in Java. URL: https://www.baeldung.com/caching-series (дата звернення: 01.12.2025).

5. Як будувати UML-діаграми. URL: https://dou.ua/forums/topic/40575/ (дата звернення: 01.12.2025).