

БАКАЛАВРСЬКА РОБОТА

БР.КІ-42.00.00.000 ПЗ

Група КІ-21-2

Дячик Владислав

2025

Міністерство освіти і науки України

Івано-Франківський національний технічний університет нафти і газу
Інститут інформаційних технологій

Кафедра комп'ютерних систем і мереж

Дячик Владислав Васильович

УДК 004.4

БАКАЛАВРСЬКА РОБОТА

Розробка web-сервісу для обліку та продажу фармацевтичних продуктів засобами мови Python та фреймворку React

Комп'ютерна інженерія

(назва освітньої програми)

123 – Комп'ютерна інженерія

(шифр і назва спеціальності)

Робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Здобувач освітнього ступеня Дячик В. В.
(підпис, ініціали та прізвище здобувача)

Науковий керівник Кропивницький Д.Р., доцент
(підпис, прізвище, ім'я, по батькові, науковий ступінь, вчене звання керівника)

Допущено до захисту
Завідувач кафедри

д.т.н., професор /С. І. Мельничук/
(посада) (підпис) (дата) (ініціали та прізвище)

Івано-Франківськ – 2025 рік

Івано-Франківський національний технічний університет нафти і газу

Інститут Інформаційних технологій

Кафедра Комп'ютерних систем і мереж

Освітньо-кваліфікаційний рівень бакалавр

Спеціальність 123 – Комп'ютерна інженерія

ЗАТВЕРДЖУЮ:

Зав. кафедрою КСМ

С.І. Мельничук

« 05 » травня 2025 року

**З А В Д А Н Н Я
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ**

Дячику Владиславу Васильовичу

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Розробка web-сервісу для обліку та продажу фармацевтичних продуктів засобами мови Python та фреймворку React

:-

керівник проекту (роботи) Кропивницький Дмитро Романович, доцент

затвержені наказом вищого навчального закладу від 05.05.2025 № 275/7

2. Строк подання студентом проекту (роботи) 12 червня 2025р.

3. Вихідні дані до роботи Методичні вказівки, технічна література

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Теоретичні та нормативні основи створення WEB-сервісу для обліку фармацевтичних товарів. 2. Проектування WEB-сервісу для аптеки

:-

3. Реалізація та тестування розробленого web-сервісу

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

6. Консультанти розділів роботи

7. Дата видачі завдання 29 січня 2025 р.

№ з/п	Назва етапів дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	<i>Збір інформації, вивчення літератури та пошук додаткової інформації</i>	<i>Лютий 2025р</i>	
2	<i>Теоретичні та нормативні основи створення WEB-сервісу для обліку фармацевтичних товарів</i>	<i>Березень 2025р</i>	
3	<i>Проектування WEB-сервісу для аптеки</i>	<i>Квітень 2025р</i>	
4	<i>Реалізація та тестування розробленого web-сервісу</i>	<i>Травень 2025р</i>	
5	<i>Оформлення додатків, дипломної роботи</i>	<i>Червень 2025р</i>	

Студент _____ Дячик В.В.

Керівник роботи _____ Кропивницький Д.Р.

АНОТАЦІЯ

Метою даної бакалаврської роботи є розробка web-сервісу для обліку та продажу фармацевтичної продукції засобами мови програмування Python та фреймворку React.

У першому розділі було проведено аналіз нормативної бази, що регламентує облік лікарських засобів та огляд існуючих сервісів управління фармацевтичними товарами.

У другому розділі здійснено проектування архітектури системи, описано її функціональні можливості, побудовано UML-діаграми та розроблено структуру бази даних.

У третьому розділі описано реалізацію клієнтської та серверної частини застосунку, а також проведено тестування працездатності створеного web-сервісу.

У результаті виконаної роботи зроблено висновки щодо ефективності, функціональності та доцільності впровадження розробленого рішення в аптечних установах.

Ключові слова: web-сервіс, фармацевтична продукція, облік товарів, Python, React, база даних, інформаційна система.

ANNOTATION

The aim of this bachelor's thesis is to develop a web service for the accounting and sale of pharmaceutical products using the Python programming language and the React framework.

The first chapter includes an analysis of regulatory frameworks governing the accounting of pharmaceuticals, as well as a review of existing solutions for managing pharmaceutical goods.

The second chapter is dedicated to designing the system architecture, describing its functional capabilities, developing UML diagrams, and creating the database structure.

The third chapter describes the implementation of both the client and server parts of the application, along with the testing of the developed web service's performance.

Based on the results of the conducted work, conclusions were drawn regarding the efficiency, functionality, and feasibility of implementing the developed solution in pharmacy chains.

Keywords: web service, pharmaceutical products, inventory management, Python, React, database, information system..

ЗМІСТ

ВСТУП.....	5
1 ТЕОРЕТИЧНІ ТА НОРМАТИВНІ ОСНОВИ СТВОРЕННЯ WEB-СЕРВІСУ ДЛЯ ОБЛІКУ ФАРМАЦЕВТИЧНИХ ТОВАРІВ	8
1.1 Огляд нормативної, наукової та технічної документації, що регламентує облік та реалізацію фармацевтичної продукції.....	8
1.2 Огляд існуючих рішень і сервісів для управління фармацевтичними товарами.....	11
1.3 Визначення інформативних ознак інформаційної системи	16
2 ПРОЄКТУВАННЯ WEB-СЕРВІСУ ДЛЯ АПТЕКИ	21
2.1 Архітектура системи	21
2.2 Створення UML-діаграм: діаграма прецедентів, діаграма класів, діаграма послідовностей	26
2.3 База даних та модель даних	33
3. Реалізація та тестування розробленого web-сервісу	38
3.1 Реалізація клієнтської частини засобами React	38
3.2 Реалізація серверної частини засобами Python	42
3.3 Опис функціональних можливостей веб-застосунку	47
3.4 Тестування і демонстрація роботи проєкту	57
ВИСНОВКИ	62
ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	65

					БР.КІ-42.00.00.000 ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка web-сервісу для обліку та продажу фармацевтичних продуктів засобами мови Python та фреймворку React	Літ.	Арк.	Акрушів
Розроб.		Дячик В.В.					3	64
Перевір.		Кропивницький Д.Р.				ІФНТУНГ, КІ-21-2		
Реценз.		Бузоверя Н.Г.						
Н. Контр.		Лазорів А.М.						
Затверд.		Мельничук С.І.						

ПЕРЕЛІК ОСНОВНИХ ПОЗНАЧЕНЬ І СКОРОЧЕНЬ

БД (База Даних) – впорядкована сукупність структурованої інформації або даних, які зберігаються та управляються електронними засобами, зазвичай із використанням системи керування базами даних (СКБД).

API (Application Programming Interface) - набір визначень підпрограм, протоколів взаємодії та засобів для створення програмного забезпечення.

CMS (Content Management System) - інформаційна система або комп'ютерна програма, що використовується для забезпечення та організації спільного процесу створення, редагування та управління вмістом.

CRUD (Create, Read, Update, Delete) – базові операції з даними в базі даних або іншій системі зберігання інформації.

FastAPI – сучасний веб-фреймворк Python для створення API з високою продуктивністю.

HTTPS (HyperText Transfer Protocol Secure) – протокол безпечної передачі даних через мережу Інтернет.

React – JavaScript-бібліотека для побудови інтерфейсів користувача.

SQL (Structured Query Language) – мова структурованих запитів для управління реляційними базами даних..

UI/UX (User Interface / User Experience) – інтерфейс користувача / досвід користувача.

UML (Unified Modeling Language) – стандартизована мова моделювання програмного забезпечення.

URL (Uniform Resource Locator) – уніфікована адреса ресурсу в Інтернеті.

ВСТУП

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		4

Актуальність обраної теми обумовлена стрімким розвитком інформаційних технологій у фармацевтичній галузі та зростаючою потребою в автоматизації процесів обліку лікарських засобів. Сучасні аптечні заклади потребують ефективних цифрових рішень для управління асортиментом препаратів, контролю залишків на складі, обробки замовлень клієнтів та забезпечення належного рівня обслуговування населення. Традиційні методи ведення обліку фармацевтичних товарів часто виявляються недостатньо гнучкими та ефективними для задоволення сучасних вимог до швидкості обслуговування, точності даних та відстеження руху товарів.

Цифровізація фармацевтичної галузі набуває особливого значення в контексті забезпечення доступності лікарських засобів для населення, особливо в умовах зростаючих вимог до прозорості та контролю якості медичних послуг. Web-сервіси для аптек дозволяють не лише автоматизувати внутрішні процеси закладу, але й забезпечити зручний доступ клієнтів до інформації про наявність препаратів, можливість попереднього замовлення та отримання професійних консультацій фармацевтів у дистанційному режимі.

Аналіз сучасного стану розвитку фармацевтичних інформаційних систем показує, що більшість існуючих рішень зосереджена на вирішенні окремих завдань або призначена для великих аптечних мереж з відповідним ціновим діапазоном. Водночас існує потреба в комплексних, доступних та легко адаптованих web-рішеннях, які можуть ефективно використовуватися як невеликими аптеками, так і середніми фармацевтичними закладами без значних інвестицій в IT-інфраструктуру.

Об'єктом дослідження є процеси автоматизації діяльності аптечних закладів через створення спеціалізованих web-сервісів для обліку та управління фармацевтичними товарами. Дослідження охоплює аналіз бізнес-процесів сучасних аптек, вивчення специфічних вимог фармацевтичної галузі до інформаційних систем та розробку технологічних рішень для їх ефективної автоматизації.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		5

Предметом дослідження є методи та технології створення web-сервісу для комплексного управління фармацевтичними товарами, включаючи архітектурні рішення, алгоритми обробки даних, інтерфейси користувачів та системи забезпечення безпеки інформації. Особлива увага приділяється інтеграції сучасних веб-технологій для створення ефективного, масштабованого та зручного у використанні програмного продукту.

Метою роботи є розробка функціонального web-сервісу для автоматизації обліку фармацевтичних товарів, який забезпечуватиме ефективне управління асортиментом препаратів, обробку замовлень клієнтів та надання якісних онлайн-послуг у сфері фармацевтичного обслуговування населення. Досягнення цієї мети передбачає створення комплексного програмного рішення з інтуїтивним користувацьким інтерфейсом та надійною серверною архітектурою.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

- провести аналіз нормативної бази та існуючих рішень у сфері автоматизації фармацевтичної діяльності;
- визначити функціональні вимоги до web-сервісу та розробити архітектуру системи;
- спроектувати структуру бази даних та модель даних для зберігання інформації про товари, користувачів та замовлення;
- реалізувати клієнтську частину застосунку з використанням сучасних фронтенд технологій;
- розробити серверну частину з RESTful API для забезпечення взаємодії з клієнтськими додатками; впровадити систему автентифікації та авторизації користувачів з різними рівнями доступу;
- створити функціональність для управління каталогом товарів, обробки замовлень та консультацій клієнтів; провести комплексне тестування розробленого програмного продукту та продемонструвати його працездатність.

Методи дослідження включають аналіз наукової літератури та нормативних документів у галузі фармацевтичної діяльності, вивчення існуючих технологічних рішень та best practices у розробці web-додатків, застосування

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		6

методологій проектування програмного забезпечення, використання сучасних технологій веб-розробки та баз даних, проведення тестування функціональності та продуктивності створеного програмного продукту. Дослідження базується на системному підході до аналізу предметної області та об'єктно-орієнтованому підході до розробки програмного забезпечення.

Практичне значення отриманих результатів полягає у створенні готового до використання web-сервісу, який може бути впроваджений в аптечних закладах для підвищення ефективності їх роботи та покращення якості обслуговування клієнтів. Розроблений програмний продукт демонструє практичне застосування сучасних веб-технологій для вирішення реальних завдань фармацевтичної галузі та може служити основою для подальшого розвитку більш складних інформаційних систем у сфері охорони здоров'я.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		7

1 ТЕОРЕТИЧНІ ТА НОРМАТИВНІ ОСНОВИ СТВОРЕННЯ WEB-СЕРВІСУ ДЛЯ ОБЛІКУ ФАРМАЦЕВТИЧНИХ ТОВАРІВ

1.1 Огляд нормативної, наукової та технічної документації, що регламентує облік та реалізацію фармацевтичної продукції

Фармацевтична діяльність в Україні регулюється комплексною системою нормативно-правових актів, що встановлюють вимоги до обліку, зберігання та реалізації лікарських засобів. Основоположним документом є Закон України "Про лікарські засоби" [1], який визначає правові та організаційні засади забезпечення населення якісними, ефективними та безпечними лікарськими засобами. Даний закон встановлює поняття лікарського засобу як будь-якої речовини або комбінації речовин, що застосовується для лікування або профілактики захворювань у людей, або будь-якої речовини, яка може застосовуватись або вводиться в організм людини з метою встановлення медичного діагнозу або для відновлення, корекції чи зміни фізіологічних функцій організму.

Державне регулювання фармацевтичного ринку здійснюється через систему ліцензування, яка регламентується Порядком ліцензування господарської діяльності з виробництва лікарських засобів, оптової та роздрібною торгівлі лікарськими засобами [2], затверджені постановою КМУ №929 від 30.11.2016 року. Ліцензування фармацевтичної діяльності є процесом надання спеціального дозволу на провадження певних видів господарської діяльності у сфері обігу лікарських засобів, що підлягають ліцензуванню відповідно до закону. Це забезпечує дотримання мінімальних стандартів якості та безпеки при роботі з фармацевтичною продукцією.

Облік лікарських засобів в аптечних закладах регламентується Порядком зберігання, обліку та знищення наркотичних засобів, психотропних речовин і прекурсорів у закладах охорони здоров'я та інших закладах, які провадять діяльність, пов'язану з обігом наркотичних засобів, психотропних речовин і

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		8

прекурсорів. Фармацевтичний облік представляє собою систему документального відображення руху лікарських засобів у аптечному закладі, що включає надходження, зберігання, відпуск та списання препаратів з обов'язковою фіксацією всіх операцій в установленій документації.

Надважливе значення має дотримання належної аптечної практики (Good Pharmacy Practice), яка визначена настановою ВООЗ та адаптована до національного законодавства України. Належна аптечна практика являє собою сукупність стандартів, що забезпечують якість фармацевтичних послуг та безпечне і раціональне використання лікарських засобів населенням. Ці стандарти охоплюють всі аспекти фармацевтичної діяльності: від прийому та зберігання лікарських засобів до консультування пацієнтів та контролю якості.

Технічні вимоги до обліку лікарських засобів встановлюються Державною фармакопеею України, яка є збірником загальнообов'язкових стандартів і нормативно-технічних документів, що встановлюють вимоги до якості лікарських засобів та лікарської рослинної сировини. Фармакопейні статті містять методи контролю якості, вимоги до зберігання, маркування та документообігу, що є критично важливими для забезпечення ефективності та безпеки лікарських препаратів.

Особливого регулювання потребує обіг наркотичних засобів, психотропних речовин та прекурсорів, який здійснюється відповідно до Закону України "Про наркотичні засоби, психотропні речовини і прекурсори". Наркотичні засоби визначаються як речовини природного чи синтетичного походження, препарати, рослини, включені до Переліку наркотичних засобів, психотропних речовин і прекурсорів. Облік таких засобів передбачає ведення спеціальних журналів, застосування посиленних заходів безпеки та регулярне звітування перед контролюючими органами.

Інформаційні технології в фармацевтичній сфері регулюються вимогами до електронного документообігу та електронного рецепта, встановленими відповідними наказами МОЗ України. Електронний рецепт - інструмент, який регулюється постановою КМУ №1300 від 17.12.2021 року [3].

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		9

Він представляє собою електронний документ, створений лікарем у електронній системі охорони здоров'я, що містить інформацію про призначені пацієнту лікарські засоби та є підставою для їх відпуску в аптеці. Впровадження таких систем потребує дотримання вимог кібербезпеки та захисту персональних даних відповідно до Закону України "Про захист персональних даних" [4].

Контроль якості лікарських засобів здійснюється відповідно до вимог Державної служби України з лікарських засобів та контролю за наркотиками, яка є центральним органом виконавчої влади у сфері контролю якості та безпеки лікарських засобів. Фармацевтична інспекція включає перевірку дотримання ліцензійних умов, належної практики дистрибуції лікарських засобів, правильності ведення обліку та зберігання препаратів.

Зберігання лікарських засобів регламентується Правилами зберігання лікарських засобів в аптеках та аптечних пунктах, які встановлюють температурні режими, вимоги до вологості, освітлення та інших параметрів навколишнього середовища. Належне зберігання є критичним фактором збереження якості, ефективності та безпеки лікарських засобів протягом всього терміну їх придатності. Порушення умов зберігання може призвести до деградації активних речовин, утворення токсичних продуктів розпаду або втрати терапевтичної ефективності.

Особливої уваги потребує регулювання реалізації рецептурних лікарських засобів, яка здійснюється виключно за рецептами лікарів відповідно до Порядку відпуску лікарських засобів з аптек та їх структурних підрозділів. Рецептурні лікарські засоби – це препарати, відпуск яких здійснюється тільки за рецептом лікаря через потенційну небезпеку їх застосування без медичного нагляду або через можливість звикання чи залежності. Контроль за відпуском таких засобів включає перевірку автентичності рецепта, відповідності дозування та тривалості лікування.

Міжнародні стандарти ISO серії 13485 "Медичні вироби. Системи управління якістю" також впливають на вимоги до обліку медичних виробів в аптечних закладах. Ці стандарти встановлюють вимоги до системи управління якістю організацій, що розробляють, виробляють, встановлюють та

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		10

обслуговують медичні вироби, включаючи супутні послуги. Впровадження таких систем забезпечує постійне поліпшення процесів та підвищення задоволеності споживачів.

Фармаконагляд як система моніторингу безпеки лікарських засобів регламентується Порядком здійснення фармаконагляду, затвердженим МОЗ України. Фармаконагляд являє собою науку та види діяльності, що стосуються виявлення, оцінки, розуміння та запобігання побічним реакціям або будь-яким іншим проблемам, пов'язаним із лікарськими засобами. Аптечні заклади зобов'язані повідомляти про всі виявлені побічні реакції та неякісні лікарські засоби до центральної бази даних фармаконагляду.

Економічні аспекти обігу лікарських засобів регулюються Порядком формування цін на лікарські засоби для медичного застосування, медичні вироби в Україні та Переліком лікарських засобів, щодо яких встановлюються граничні надбавки. Ціноутворення на фармацевтичному ринку здійснюється з урахуванням міжнародних цін, купівельної спроможності населення та необхідності забезпечення доступності життєво важливих лікарських засобів.

Інтеграція інформаційних систем в фармацевтичній сфері потребує дотримання технічних стандартів міжопераційності та безпеки даних. Електронні системи обліку лікарських засобів повинні забезпечувати можливість інтеграції з державними реєстрами, системами електронної охорони здоров'я та міжнародними базами даних лікарських засобів. Це дозволяє автоматизувати процеси звітності, забезпечити трасованість лікарських засобів та підвищити ефективність контролю за їх обігом на всіх етапах фармацевтичного ланцюга постачань.

1.2 Огляд існуючих рішень і сервісів для управління фармацевтичними товарами

Сучасний ринок програмного забезпечення для фармацевтичної галузі характеризується широким спектром спеціалізованих рішень, що охоплюють всі аспекти управління аптечною діяльністю. Фармацевтичні інформаційні системи

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		11

(Pharmacy Information Systems, PIS) представляють собою комплексні програмні платформи, призначені для автоматизації процесів обліку, реалізації та управління лікарськими засобами в аптечних закладах. Такі системи інтегрують функціональність управління запасами, обробки рецептів, взаємодії з клієнтами та фінансового обліку в єдине технологічне середовище.

Провідні міжнародні виробники пропонують високотехнологічні рішення, серед яких особливо виділяється система McKesson EnterpriseRx, що використовується понад 40000 аптек у Сполучених Штатах. Ця платформа забезпечує повний цикл управління аптечними операціями, включаючи автоматизовану обробку рецептів, інтелектуальне управління запасами з прогнозуванням попиту, систему попереджень про лікарські взаємодії та комплексну аналітику ефективності бізнес-процесів.

EnterpriseRx характеризується модульною архітектурою, що дозволяє адаптувати функціональність під специфічні потреби конкретного аптечного закладу.

Компанія Epic Systems розробила MyChart Pharmacy, яка представляє собою інтегровану платформу для управління фармацевтичними послугами в рамках більш широкої екосистеми електронних медичних записів. Система забезпечує безшовну інтеграцію між лікарнями, клініками та аптеками, дозволяючи медичним працівникам та фармацевтам мати доступ до повної історії пацієнта та координувати лікування. Електронна система управління рецептами в MyChart включає функції автоматичної перевірки дозування, моніторингу комплаєнсу пацієнтів та оптимізації терапевтичних схем.

Французька компанія Cegedim Healthcare Solutions пропонує рішення Pharmagest, яке широко використовується в європейських аптеках та характеризується потужними можливостями управління багатоканальними продажами. Pharmagest інтегрує традиційну роздрібну торгівлю, електронну комерцію та мобільні додатки в єдину омніканальну платформу. Система включає модулі для управління лояльністю клієнтів, персоналізованого маркетингу та аналітики споживчої поведінки, що дозволяє аптекам підвищувати ефективність продажів та покращувати сервіс.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		12

На вітчизняному ринку значну частку займає система "Фармація", розроблена українською компанією "Інтелект-Сервіс", яка адаптована до специфіки національного законодавства та особливостей роботи українських аптек. Ця система забезпечує ведення обліку лікарських засобів відповідно до вимог українського законодавства, включаючи спеціальний облік наркотичних та психотропних речовин, генерацію обов'язкової звітності для державних органів та інтеграцію з системою електронних рецептів МОЗ України. Автоматизована аптечна система являє собою програмний комплекс, що забезпечує комп'ютеризацію всіх основних бізнес-процесів аптечного закладу.

Корпоративні ERP-системи, такі як SAP for Pharmaceuticals та Oracle Health Sciences, пропонують масштабовані рішення для великих фармацевтичних дистриб'юторів та аптечних мереж. SAP for Pharmaceuticals включає спеціалізовані модулі для управління серійними номерами та термінами придатності, забезпечення повної трасованості продукції відповідно до вимог регулюючих органів та автоматизації процесів відкриття продукції з ринку. Система підтримує складні схеми ціноутворення, багаторівневі знижки та програми лояльності, що є критично важливим для ефективного управління великими аптечними мережами.

Спеціалізовані системи управління запасами, такі як Cardinal Health Inventory Management та AmerisourceBergen Pharmaceutical Inventory Solutions, фокусуються на оптимізації логістичних процесів та мінімізації втрат від зіпсованих препаратів. Інтелектуальне управління запасами використовує алгоритми машинного навчання для прогнозування попиту, оптимізації рівнів запасів та автоматизації процесів замовлення товарів. Ці системи аналізують історичні дані продажів, сезонні коливання попиту, епідеміологічну ситуацію та інші фактори для забезпечення оптимального балансу між доступністю препаратів та мінімізацією фінансових втрат.

Платформи електронного рецепта, такі як Surescripts в США та NHS Electronic Prescription Service у Великій Британії, революціонізували процес обробки рецептів та взаємодії між лікарями, пацієнтами та фармацевтами. Електронний рецепт (e-prescription) представляє собою цифровий документ, що

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		13

створюється лікарем в електронній медичній системі і передається безпосередньо до обраної пацієнтом аптеки через захищені комунікаційні канали. Такі системи значно скорочують час обробки рецептів, зменшують ризик помилок, пов'язаних з нерозбірливим почерком лікарів, та забезпечують автоматичну перевірку лікарських взаємодій.

CRM-системи для фармацевтичної галузі, такі як Salesforce Health Cloud та Microsoft Dynamics 365 for Healthcare, забезпечують персоналізований підхід до обслуговування клієнтів та ефективного управління взаємовідносинами з пацієнтами. Фармацевтичний CRM інтегрує дані про покупки клієнтів, їх медичну історію, преференції та поведінкові паттерни для формування персоналізованих рекомендацій та програм лояльності. Ці системи також включають функції автоматичного нагадування про необхідність поповнення рецептів, моніторингу комплаєнсу пацієнтів та координації з лікуючими лікарями.

Хмарні рішення стають все більш популярними завдяки їх масштабованості, економічній ефективності та можливості швидкого впровадження. Система PioneerRx Cloud та Micro Merchant Systems Cloud-based POS представляють собою Software-as-a-Service (SaaS) рішення, що дозволяють аптекам отримати доступ до професійного програмного забезпечення без значних початкових інвестицій в IT-інфраструктуру. Хмарні фармацевтичні системи забезпечують автоматичне резервне копіювання даних, безперервність бізнес-процесів та можливість доступу до системи з будь-якої точки світу через інтернет.

Мобільні додатки для фармацевтичної галузі, такі як GoodRx, PillReminder та CVS Pharmacy App, розширюють можливості взаємодії аптек з клієнтами та надають додаткові канали продажів. Мобільні фармацевтичні платформи дозволяють пацієнтам замовляти ліки онлайн, отримувати нагадування про прийом препаратів, порівнювати ціни в різних аптеках та отримувати консультації фармацевтів через чат або відеозв'язок. Інтеграція мобільних додатків з основними аптечними системами забезпечує синхронізацію даних та безшовний досвід для користувачів.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		14

Системи бізнес-аналітики та звітності, такі як Tableau for Healthcare та QlikView Pharmaceutical Analytics, надають потужні інструменти для аналізу ефективності аптечних операцій та прийняття стратегічних рішень. Фармацевтична аналітика включає моніторинг ключових показників ефективності (KPI), аналіз трендів продажів, оцінку рентабельності різних категорій товарів та прогнозування майбутніх потреб. Ці системи використовують візуалізацію даних, інтерактивні дашборди та автоматизовані звіти для надання менеджерам актуальної інформації про стан бізнесу.

Інноваційні технології, такі як штучний інтелект та машинне навчання, все більше інтегруються в фармацевтичні системи для підвищення точності прогнозування, автоматизації рутинних завдань та покращення якості обслуговування. AI-powered фармацевтичні асистенти можуть аналізувати симптоми пацієнтів, рекомендувати безрецептурні препарати, оптимізувати схеми прийому ліків та виявляти потенційні побічні ефекти. Комп'ютерне бачення використовується для автоматичного розпізнавання та сортування лікарських засобів, що значно прискорює процеси інвентаризації та зменшує ризик помилок.

Blockchain-технології починають знаходити застосування в фармацевтичній галузі для забезпечення автентичності препаратів та боротьби з підробками. Системи, такі як MediLedger та IBM Food Trust для фармації, створюють незмінні цифрові записи про весь ланцюг постачань лікарських засобів від виробника до кінцевого споживача. Фармацевтичний блокчейн забезпечує повну трасованість препаратів, дозволяє швидко ідентифікувати джерело проблем у разі відкликання продукції та підвищує довіру споживачів до якості лікарських засобів.

Результати проведеного аналізу розглянутих інформаційних систем у фармацевтичній сфері відображено у таблиці 1.1.

Таблиця 1.1 – Порівняння існуючих інформаційних систем у фармацевтичній сфері

		Назва системи	Опис	Переваги	Недоліки	Тип користувача
		McKesson	Комплексна	Інтелектуальне	Висока вартість,	Великі
БР.КІ-42.00.00.000 ПЗ						
						Арк.
Змін.	Арк.	№ докум.	Підпис	Дата	15	

EnterpriseRx	система для аптек (США), автоматизація всіх процесів	управління запасами, аналітика, модульна структура	орієнтована на великі мережі	аптечні мережі
Pharma ERP (Україна)	ERP-рішення з підтримкою складського обліку, кас, замовлень	Інтеграція з ІС, електронний документообіг, підтримка українського законодавства	Складність адаптації для невеликих аптек	Середній бізнес
MediLedger	Blockchain-рішення для верифікації ланцюга постачання	Безпека даних, боротьба з підробками, прозорість поставок	Висока складність інтеграції, потреба в участі всіх учасників ринку	Мережі, логістичні ланцюги
Артека24 CRM	Онлайн-платформа для невеликих аптек (Україна), фокус на онлайн-продажі	Простота, інтеграція з сайтом, базовий облік та статистика	Обмежена функціональність для складних процесів	Невеликі приватні аптеки
Прототип (мій web-сервіс)	Web-сервіс на Python + React з адаптивною архітектурою	Гнучкість, масштабованість, низькі витрати на впровадження, можливість доопрацювання	Потреба у розробці та супроводі, обмежена кількість модулів на початковому етапі	Невеликі та середні аптечні заклади

Порівняльний аналіз існуючих рішень показує, що найбільш ефективними є інтегровані платформи, що поєднують функціональність управління запасами, обробки рецептів, взаємодії з клієнтами та бізнес-аналітики в єдиній системі. Однак вибір конкретного рішення повинен базуватися на розмірі аптечного закладу, специфіці його діяльності, бюджетних обмеженнях та вимогах до інтеграції з існуючими системами. Тенденції розвитку ринку вказують на зростання попиту на хмарні рішення, мобільні додатки та системи з елементами штучного інтелекту, що здатні забезпечити конкурентні переваги в умовах цифровізації фармацевтичної галузі.

1.3 Визначення інформативних ознак інформаційної системи

						БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата			16

Інформативні ознаки інформаційної системи представляють собою сукупність характеристик, параметрів та властивостей, що визначають функціональні можливості, технічні характеристики та якісні показники програмного продукту або інформаційного рішення. Інформативні ознаки являють собою формалізовані критерії, за якими можна оцінити відповідність інформаційної системи поставленим вимогам, порівняти різні рішення між собою та визначити ступінь задоволення потреб користувачів. Для фармацевтичних інформаційних систем ці ознаки набувають особливого значення через специфічні вимоги галузі до точності, безпеки та відповідності регулятивним нормам.

Функціональна повнота системи є однією з ключових інформативних ознак, що характеризує ступінь покриття всіх необхідних бізнес-процесів аптечного закладу програмним забезпеченням. Функціональна повнота включає можливості автоматизації процесів прийому та обліку товарів, управління запасами, обробки рецептів, ведення касових операцій, формування звітності та взаємодії з клієнтами. Для фармацевтичних систем критично важливими є спеціалізовані функції контролю термінів придатності, обліку наркотичних та психотропних речовин, перевірки лікарських взаємодій та забезпечення трасованості препаратів відповідно до регулятивних вимог.

Масштабованість інформаційної системи визначає її здатність ефективно функціонувати при зростанні обсягів даних, кількості користувачів та інтенсивності транзакцій без критичного погіршення продуктивності. Горизонтальна масштабованість передбачає можливість розширення системи шляхом додавання нових серверів або вузлів обробки, тоді як вертикальна масштабованість означає можливість покращення характеристик існуючого обладнання. Для аптечних мереж особливо важливою є можливість підключення нових точок продажів, інтеграції з додатковими постачальниками та обробки зростаючих обсягів транзакцій без перебудови архітектури системи.

Продуктивність системи характеризується швидкістю обробки запитів, пропускною здатністю та часом відгуку на дії користувачів. Ключовими метриками продуктивності є час обробки одного рецепта, швидкість

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		17

формування звітів, час виконання операцій інвентаризації та затримки при синхронізації даних між різними модулями системи. Висока продуктивність критично важлива для аптечних систем, оскільки затримки в обслуговуванні клієнтів можуть призвести до втрати продажів та погіршення репутації закладу.

Надійність інформаційної системи визначається як ймовірність безвідмовної роботи протягом заданого періоду часу в певних умовах експлуатації. Коефіцієнт готовності системи (система availability) вимірюється як відношення часу працездатного стану до загального часу роботи та зазвичай виражається у відсотках або через концепцію "дев'яток" (99.9%, 99.99% тощо). Для фармацевтичних систем критично важливою є висока надійність, оскільки збої можуть призвести до неможливості обслуговування пацієнтів, втрати даних про продажі та порушення регулятивних вимог до ведення обліку.

Безпека інформаційної системи охоплює комплекс заходів та технологій, спрямованих на захист від несанкціонованого доступу, модифікації або знищення даних. Інформаційна безпека включає автентифікацію користувачів, авторизацію доступу до різних функцій системи, шифрування даних при зберіганні та передачі, аудит дій користувачів та захист від зовнішніх кіберзагроз. Для фармацевтичних систем особливо важливими є захист персональних даних пацієнтів відповідно до вимог GDPR та національного законодавства, а також забезпечення цілісності даних про обіг наркотичних засобів.

Сумісність та інтероперабельність системи характеризують її здатність ефективно взаємодіяти з іншими програмними продуктами, апаратними засобами та стандартами обміну даними. Технічна сумісність включає підтримку різних операційних систем, баз даних та мережевих протоколів, тоді як семантична інтероперабельність забезпечує можливість обміну даними зі збереженням їх змісту та контексту. Для аптечних систем критично важливою є інтеграція з системами постачальників, електронними рецептами, державними реєстрами та банківськими платформами.

Користувацький інтерфейс та зручність використання (usability) визначають ефективність взаємодії людини з системою та впливають на

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		18

продуктивність роботи персоналу. Ергономічні характеристики інтерфейсу включають інтуїтивність навігації, швидкість навчання нових користувачів, зниження ймовірності помилок та загальне задоволення від роботи з системою. Якісний користувацький досвід особливо важливий для фармацевтичних систем, оскільки фармацевти повинні швидко та точно обробляти інформацію в умовах постійного потоку клієнтів.

Адаптивність системи характеризує її здатність пристосовуватися до змін у бізнес-процесах, регулятивних вимогах та технологічному середовищі без кардинальної перебудови архітектури. Конфігурованість системи дозволяє налаштовувати функціональність під специфічні потреби конкретного аптечного закладу, тоді як можливості кастомізації забезпечують розробку додаткових модулів та інтеграцій. Гнучкість архітектури дозволяє ефективно реагувати на зміни в законодавстві, впроваджувати нові бізнес-моделі та адаптуватися до еволюції фармацевтичного ринку.

Якість даних в інформаційній системі характеризується точністю, повнотою, актуальністю, узгодженістю та доступністю інформації. Точність даних означає відповідність збереженої інформації реальному стану об'єктів та процесів, повнота характеризує наявність всіх необхідних атрибутів та зв'язків, а актуальність визначає своєчасність оновлення інформації. Для фармацевтичних систем якість даних критично важлива, оскільки помилки в інформації про лікарські засоби можуть мати серйозні наслідки для здоров'я пацієнтів.

Документування та підтримка системи включають технічну документацію, інструкції користувача, програми навчання персоналу та сервісну підтримку від розробника. Якісна документація забезпечує ефективне впровадження системи, зменшує час навчання користувачів та полегшує супровід і розвиток системи. Технічна підтримка включає оперативне вирішення проблем, регулярні оновлення програмного забезпечення, консультації з оптимізації роботи та планування розвитку системи відповідно до зростаючих потреб організації.

Економічна ефективність інформаційної системи визначається співвідношенням витрат на її впровадження та експлуатацію до отриманих економічних вигод. Сукупна вартість володіння (Total Cost of Ownership, TCO)

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		19

включає первинні інвестиції в програмне та апаратне забезпечення, витрати на впровадження та навчання персоналу, поточні витрати на ліцензії та підтримку, а також витрати на модернізацію та розвиток системи. Економічні вигоди включають підвищення продуктивності персоналу, зменшення операційних витрат, покращення контролю запасів та зростання доходів від оптимізації бізнес-процесів.

Аналітичні можливості системи характеризують її здатність збирати, обробляти та представляти дані для підтримки прийняття управлінських рішень. Бізнес-інтелект (Business Intelligence) функціональність включає формування звітів, аналіз трендів продажів, моніторинг ключових показників ефективності та прогнозування майбутніх потреб. Для фармацевтичних систем особливо важливими є аналіз ABC/XYZ для оптимізації запасів, моніторинг споживання різних категорій препаратів, аналіз сезонних коливань попиту та оцінка ефективності маркетингових заходів.

Мобільність та віддалений доступ до системи забезпечують можливість роботи з різних локацій та пристроїв, що особливо важливо в умовах розвитку омніканальної торгівлі та необхідності оперативного управління аптечними мережами. Мобільні додатки дозволяють керівникам контролювати роботу віддалених аптек, фармацевтам отримувати доступ до довідкової інформації та клієнтам замовляти препарати онлайн. Хмарні технології забезпечують централізоване зберігання даних, автоматичне резервне копіювання та можливість масштабування ресурсів відповідно до поточних потреб.

Відповідність регулятивним вимогам та стандартам якості є критично важливою інформативною ознакою для фармацевтичних систем. Система повинна забезпечувати дотримання вимог належної аптечної практики, стандартів ISO 27001 для інформаційної безпеки, регулятивних вимог FDA або ЕМА для трасованості препаратів та національних стандартів електронного документообігу. Комплаєнс-функціональність включає автоматичну генерацію обов'язкової звітності, контроль дотримання процедур та аудит дій користувачів для забезпечення прозорості та підзвітності операцій.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		20

2 ПРОЄКТУВАННЯ WEB-SERVISU ДЛЯ АПТЕКИ

2.1 Архітектура системи

Архітектура розробленого web-сервісу для аптеки базується на сучасній клієнт-серверній моделі з чіткою декомпозицією функціональних компонентів та рівнів відповідальності. Система побудована за принципами багаторівневої архітектури, що забезпечує високу масштабованість, підтримуваність та гнучкість у подальшому розвитку функціональності. Загальна архітектурна концепція передбачає розділення презентаційного рівня, бізнес-логіки та рівня доступу до даних, що відповідає найкращим практикам розробки корпоративних веб-додатків.

Презентаційний рівень системи реалізований як односторінковий веб-додаток (Single Page Application, SPA) з використанням бібліотеки React та екосистеми супутніх інструментів. Цей підхід забезпечує швидку та відзивчиву взаємодію користувача з інтерфейсом, мінімізує навантаження на сервер за рахунок клієнтського рендерингу компонентів та надає можливість створення складних інтерактивних елементів інтерфейсу. Використання віртуального DOM у React дозволяє оптимізувати процес оновлення інтерфейсу та забезпечити високу продуктивність навіть при роботі з великими обсягами даних.

Клієнтська частина додатку структурована за компонентним підходом, де кожен функціональний блок представлений окремим React-компонентом з власним станом та життєвим циклом. Управління станом додатку здійснюється за допомогою вбудованих хуків React, таких як `useState` та `useEffect`, що дозволяє ефективно керувати локальним станом компонентів та побічними ефектами. Навігація між різними розділами додатку реалізована за допомогою бібліотеки `react-router-dom`, яка забезпечує декларативний підхід до маршрутизації та підтримку історії браузера.

Серверна частина системи побудована на базі Python з використанням веб-

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		21

фреймворку, що забезпечує RESTful API для взаємодії з клієнтською частиною. Архітектура серверної частини відповідає принципам MVC (Model-View-Controller), де моделі представляють структуру даних та бізнес-логіку, контролери обробляють HTTP-запити та координують взаємодію між різними компонентами, а представлення формують JSON-відповіді для клієнтських запитів. Такий підхід забезпечує чітке розділення відповідальності та спрощує тестування і підтримку коду.

Взаємодія між клієнтською та серверною частинами здійснюється через HTTP/HTTPS протокол з використанням бібліотеки axios для виконання асинхронних запитів. Система підтримує повний спектр CRUD операцій (Create, Read, Update, Delete) для всіх основних сутностей, включаючи користувачів, товари, замовлення та повідомлення чату. Архітектура API побудована за принципами REST, що забезпечує інтуїтивність та стандартизованість інтерфейсів взаємодії між компонентами системи.

Дану архітектуру системи зображено у вигляді логічної схеми (рис. 2.1).

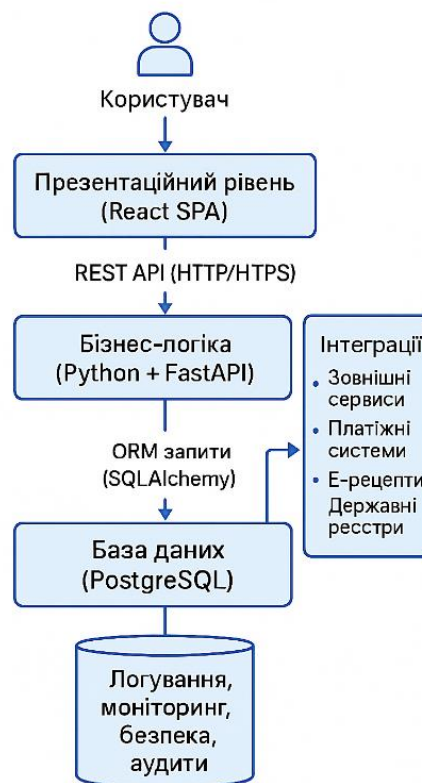


Рисунок 2.1 - Логічна схема архітектури проекту

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		22

Аутентифікація та авторизація користувачів реалізована за допомогою JSON Web Tokens (JWT), що забезпечує безпечну та масштабовану систему управління сесіями без необхідності зберігання стану на сервері. Токени містять закодовану інформацію про користувача та його права доступу, що дозволяє серверній частині швидко валідувати запити та застосовувати відповідні політики безпеки. Архітектура передбачає різні рівні доступу для різних типів користувачів, включаючи гостей користувачів, зареєстрованих клієнтів та адміністраторів системи.

Система комунікації в реальному часі реалізована за допомогою WebSocket протоколу з використанням бібліотеки Socket.io, що забезпечує двонаправлений обмін повідомленнями між клієнтом та сервером. Ця архітектурна особливість дозволяє реалізувати функціональність чату для консультацій з фармацевтами, миттєві сповіщення про зміни в замовленнях та інші інтерактивні функції, що потребують оперативного обміну даними. WebSocket з'єднання встановлюються поверх існуючої HTTP інфраструктури та автоматично адаптуються до різних мережеских умов.

Рівень збереження даних представлений реляційною базою даних, що забезпечує ACID властивості транзакцій та підтримку складних запитів з об'єднаннями між різними таблицями. Архітектура бази даних нормалізована до третьої нормальної форми для мінімізації дублювання даних та забезпечення цілісності інформації. Взаємодія з базою даних здійснюється через ORM (Object-Relational Mapping) рівень, що абстрагує SQL запити та надає об'єктно-орієнтований інтерфейс для роботи з даними.

Архітектура системи передбачає горизонтальне масштабування через можливість розгортання декількох екземплярів серверної частини за балансувальником навантаження. Клієнтська частина може бути розповсюджена через Content Delivery Network (CDN) для забезпечення швидкого доступу користувачів з різних географічних регіонів. Така архітектурна гнучкість дозволяє адаптувати систему до зростаючого навантаження та розширення функціональності без кардинальної перебудови існуючих компонентів.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		23

Система логуювання та моніторингу інтегрована на всіх рівнях архітектури для забезпечення операційної прозорості та можливості швидкого діагностування проблем. Клієнтська частина використовує centralized error handling для збору та передачі інформації про помилки, тоді як серверна частина веде детальні логи всіх операцій з можливістю налаштування різних рівнів деталізації. Архітектура передбачає інтеграцію з зовнішніми системами моніторингу для автоматичного сповіщення про критичні події.

Безпекова архітектура системи включає декілька рівнів захисту, включаючи HTTPS шифрування для всіх комунікацій, валідацію вхідних даних на клієнтському та серверному рівнях, захист від CSRF атак та SQL ін'єкцій. Система управління сесіями побудована з урахуванням найкращих практик безпеки, включаючи обмежений час життя токенів, secure cookie налаштування та регулярну ротацію ключів шифрування. Архітектура також передбачає audit trail для всіх критичних операцій в системі.

Модульна архітектура дозволяє незалежну розробку та розгортання різних компонентів системи, що сприяє паралельній роботі команд розробників та зменшує ризики при внесенні змін. Кожен модуль має чітко визначені інтерфейси взаємодії та може бути протестований незалежно від інших компонентів. Така архітектурна організація також спрощує процес інтеграції з зовнішніми системами та дозволяє поетапне впровадження нової функціональності.

Архітектура передбачає підтримку міжнародних стандартів доступності (WCAG) та кросбраузерної сумісності, що забезпечує можливість використання системи широким колом користувачів незалежно від їх технічних можливостей чи обмежень. Адаптивний дизайн реалізований на рівні CSS медіа-запитів та гнучких компонентів React, що автоматично адаптуються до різних розмірів екрану та типів пристроїв. Така архітектурна особливість критично важлива для фармацевтичних систем, оскільки забезпечує доступність послуг для користувачів з різними потребами.

Інтеграційна архітектура системи спроектована з урахуванням можливості підключення до зовнішніх сервісів, таких як платіжні системи, системи

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		24

управління складськими запасами, електронні рецепти та державні реєстри лікарських засобів. API-first підхід забезпечує стандартизовані інтерфейси для інтеграції, тоді як асинхронна обробка запитів дозволяє підтримувати високу продуктивність навіть при взаємодії з повільними зовнішніми сервісами. Архітектура також включає механізми retry та circuit breaker для забезпечення стійкості системи при збоях в зовнішніх залежностях.

2.2 Створення UML-діаграм: діаграма прецедентів, діаграма класів, діаграма послідовностей

Діаграма прецедентів веб-сервісу для аптеки відображає взаємодію різних типів користувачів з функціональними можливостями системи, визначаючи основні сценарії використання та їх взаємозв'язки, як представлено на рисунку 2.2.

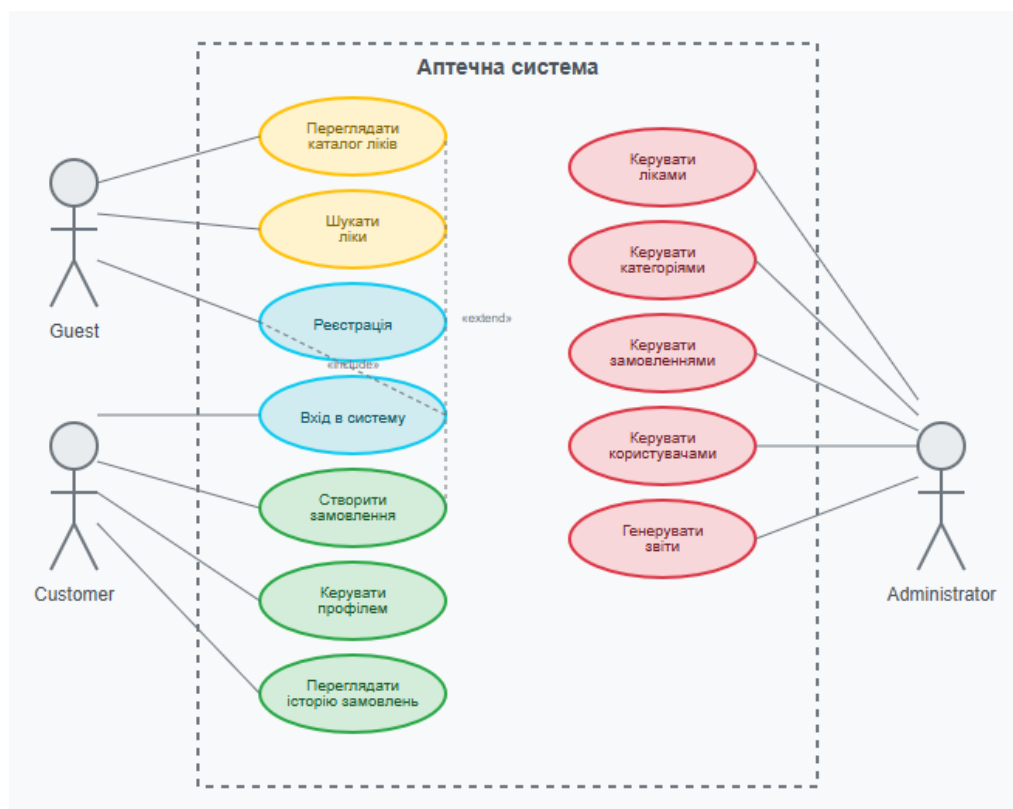


Рисунок 2.2 - Діаграма прецедентів системи з акторами Guest, Customer та Administrator

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		25

Система підтримує три основні типи акторів: гостьові користувачі (Guest), зареєстровані клієнти (Customer) та адміністратори (Administrator), кожен з яких має доступ до специфічного набору функцій відповідно до своїх потреб та рівня авторизації. Гостьові користувачі можуть переглядати каталог товарів, шукати препарати, переглядати детальну інформацію про товари та користуватися базовими інформаційними сервісами без необхідності реєстрації в системі.

Зареєстровані клієнти отримують доступ до розширеного функціоналу, включаючи можливість додавання товарів до замовлень, оформлення замовлень, управління особистим профілем, перегляд історії покупок та отримання персоналізованих рекомендацій. Процес реєстрації та автентифікації також представлений окремими прецедентами, що демонструють перехід користувача від гостьового статусу до повноправного клієнта системи. Прецедент управління профілем включає можливості редагування особистої інформації, зміни паролю та налаштування преференцій сповіщень.

Адміністратори мають найширші повноваження в системі, включаючи управління каталогом лікарських засобів, обробку замовлень, управління користувачами, налаштування категорій препаратів та перегляд аналітичних звітів. Прецедент управління лікарськими засобами включає додавання нових препаратів, редагування існуючих товарів, встановлення статусів препаратів, управління категоріями та характеристиками ліків. Управління замовленнями передбачає можливість перегляду всіх замовлень, зміни їх статусів, координації з службою доставки та відстеження виконання замовлень.

Діаграма також відображає відносини розширення (extend) та включення (include) між прецедентами, демонструючи складні сценарії використання. Наприклад, прецедент "Оформлення замовлення" включає валідацію наявності препаратів, розрахунок загальної кількості товарів та підтвердження інформації про доставку. Прецедент "Пошук лікарських засобів" може розширюватися додатковими функціями фільтрації за категоріями, виробниками та статусом препаратів.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		26

Діаграма класів системи відображає статичну структуру програмного забезпечення, визначаючи основні класи, їх атрибути, методи та відносини між ними, як показано на рисунку 2.3.

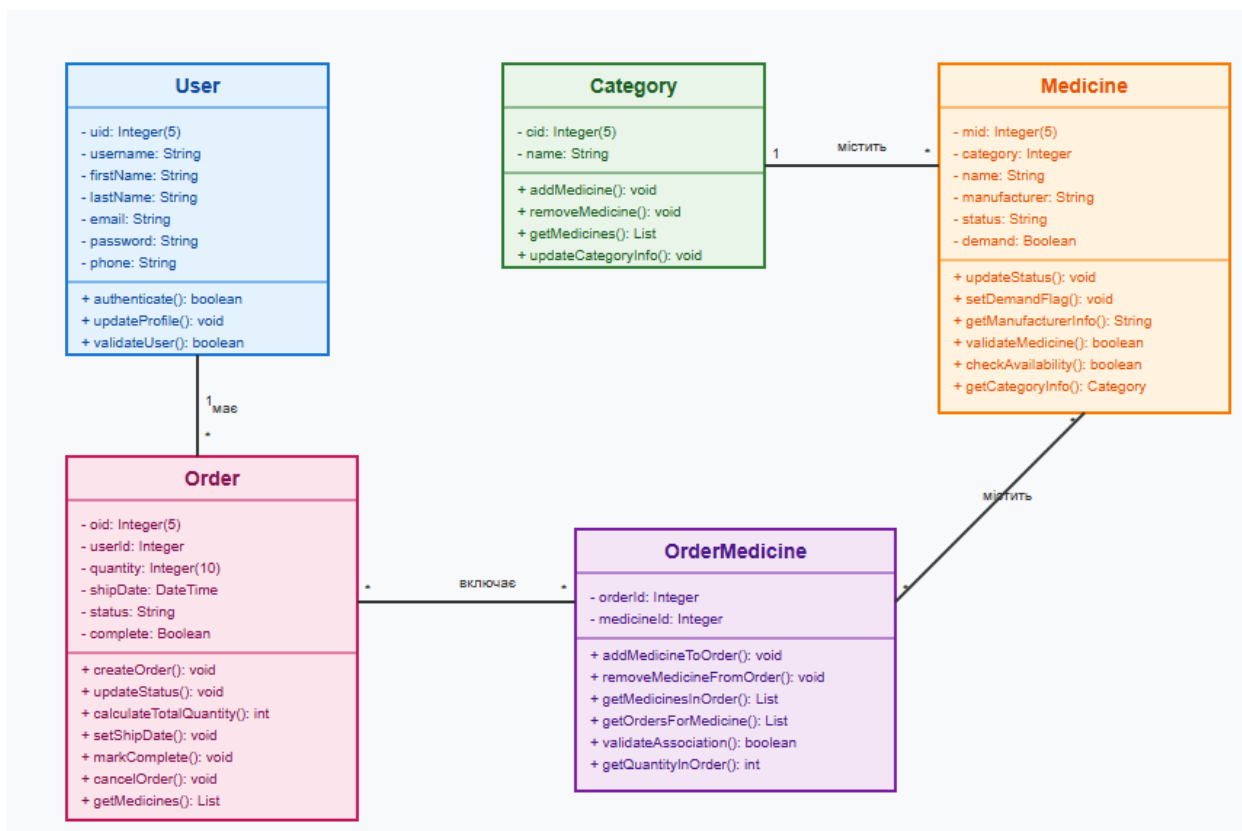


Рисунок 2.3 - Діаграма класів з основними сутностями системи:
User, Medicine, Category, Order

Центральними класами системи є User, Medicine, Category, Order та OrderMedicine, які формують основу предметної області фармацевтичного бізнесу відповідно до структури бази даних. Клас User представляє всіх користувачів системи і містить атрибути uid, username, firstName, lastName, email, password та phone, а також методи для автентифікації authenticate(), валідації даних validateUser(), управління профілем updateProfile() та отримання інформації getUserInfo().

Клас User може мати спеціалізовані підкласи Customer та Administrator, які успадковують базову функціональність та додають специфічні методи. Клас Customer додає методи для управління замовленнями createOrder(), viewOrderHistory(), а клас Administrator включає додаткові методи для

управління системою `manageMedicines()`, `manageUsers()`, `generateReports()` та `manageCategories()`.

Клас `Category` представляє категорії лікарських засобів та містить атрибути `cid` та `name`. Методи класу включають `addMedicine()` для додавання препаратів до категорії, `removeMedicine()` для видалення препаратів, `getMedicines()` для отримання списку всіх препаратів категорії та `updateCategoryInfo()` для редагування інформації про категорію. Клас забезпечує логічну організацію каталогу лікарських засобів.

Клас `Medicine` представляє лікарські засоби в системі та містить атрибути `mid`, `category`, `name`, `manufacturer`, `status` та `demand`. Методи класу включають `updateStatus()` для зміни статусу препарату, `setDemandFlag()` для маркування препаратів з високим попитом, `getManufacturerInfo()` для отримання інформації про виробника, `validateMedicine()` для перевірки коректності даних препарату та `checkAvailability()` для перевірки доступності товару. Відносини між `Medicine` та `Category` реалізовані через асоціацію "many-to-one", дозволяючи групування препаратів за категоріями.

Клас `Order` моделює замовлення клієнтів та включає атрибути `oid`, `userId`, `quantity`, `shipDate`, `status` та `complete`. Методи класу забезпечують функціональність створення замовлення `createOrder()`, оновлення статусу `updateStatus()`, розрахунку загальної кількості `calculateTotalQuantity()`, встановлення дати відправки `setShipDate()`, позначення замовлення як завершеного `markComplete()` та скасування замовлення `cancelOrder()`. Відносини між `Order` та `User` реалізовані через асоціацію "many-to-one", де один користувач може мати множину замовлень.

Клас `OrderMedicine` представляє асоціативну сутність між замовленнями та лікарськими засобами, що реалізує зв'язок "багато-до-багатьох". Клас містить атрибути `orderId` та `medicineId` як складений ключ та методи `addMedicineToOrder()` для додавання препарату до замовлення, `removeMedicineFromOrder()` для видалення препарату з замовлення, `getMedicinesInOrder()` для отримання списку всіх препаратів у замовленні та

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		28

getOrdersForMedicine() для знаходження всіх замовлень, що містять конкретний препарат.

Відносини між класами включають асоціації та агрегації, точно відображаючи структуру предметної області та взаємодії між різними компонентами системи. Множинності відносин вказують на кардинальність зв'язків: один користувач може мати множину замовлень (1:), одна категорія може містити множину лікарських засобів (1:), одне замовлення може включати множину препаратів через асоціативний клас OrderMedicine (:).

Діаграма також включає utility класи для забезпечення допоміжної функціональності: DatabaseManager для роботи з базою даних, ValidationService для валідації вхідних даних, NotificationService для відправки сповіщень користувачам та ReportGenerator для створення звітів про замовлення та продажі. Ці класи інкапсулюють системну логіку та забезпечують розділення відповідальності між компонентами.

Інкапсуляція реалізована через приватні атрибути з публічними методами доступу (getters/setters), що забезпечує контрольований доступ до даних та можливість валідації при зміні значень. Поліморфізм використовується в ієрархії класів User для різної реалізації методів у підкласах Customer та Administrator відповідно до їх специфічних ролей у системі.

Діаграма послідовностей детально описує взаємодію об'єктів у часі для виконання конкретних сценаріїв використання системи, як представлено на рисунку 2.4.

Ключовим сценарієм є процес оформлення замовлення зареєстрованим користувачем, який демонструє взаємодію між браузером клієнта, веб-сервером та базою даних. Сценарій починається з ініціації користувачем процесу створення нового замовлення, що призводить до відправки HTTP POST запиту на сервер з інформацією про обрані лікарські засоби та їх кількості.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		29

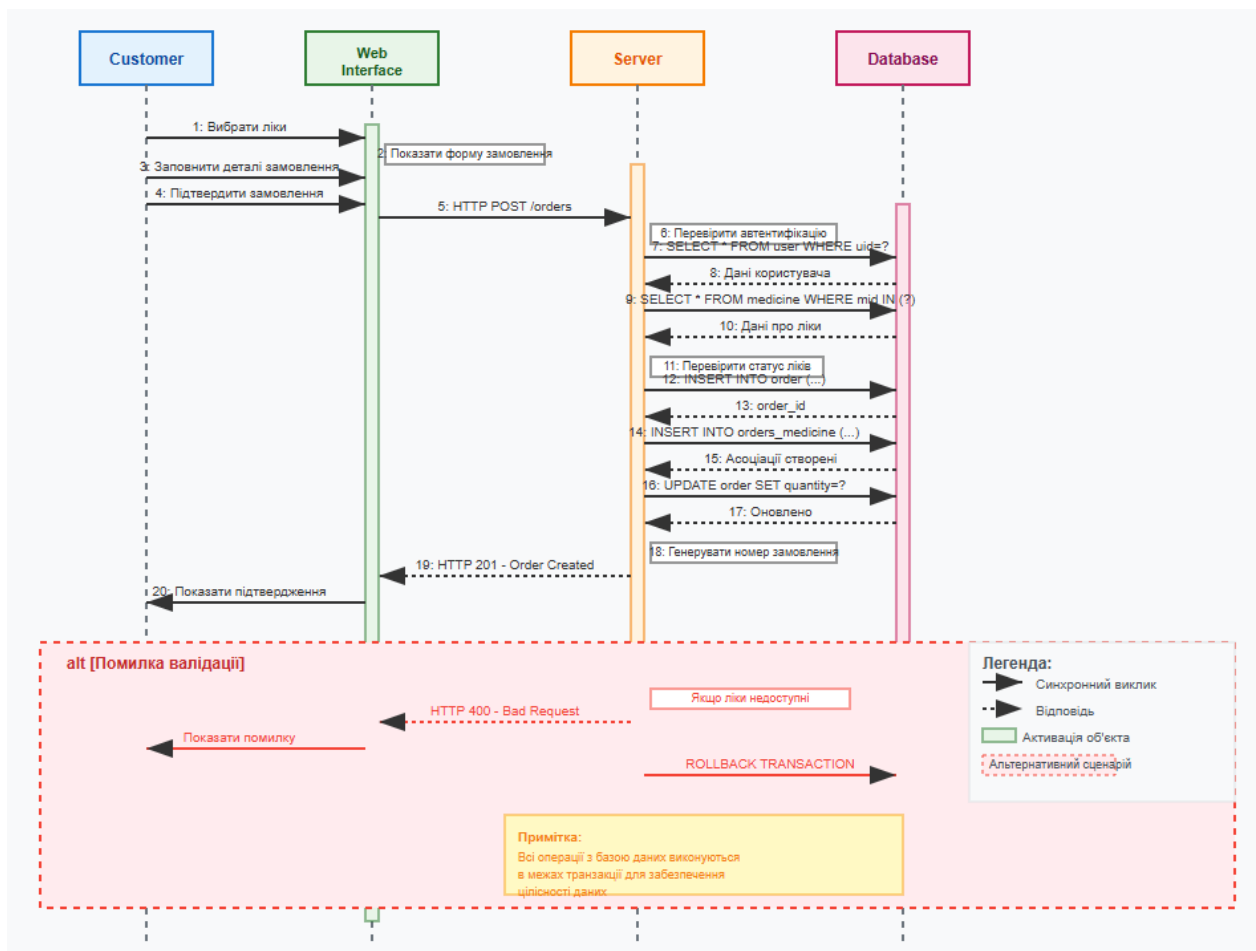


Рисунок 2.4 - Діаграма послідовностей процесу оформлення замовлення з взаємодією між користувачем, системою та базою даних

Сервер отримує запит та починає серію валідацій, включаючи перевірку автентифікації користувача через сесію або токен, валідацію існування користувача в таблиці user за userId, перевірку існування обраних препаратів у таблиці medicine та валідацію їх поточного статусу. Кожна валідація супроводжується SQL запитом до відповідних таблиць для отримання актуальної інформації.

У випадку успішної валідації всіх даних, система створює новий запис у таблиці order з автоматично згенерованим oid, userId поточного користувача, загальною кількістю товарів quantity, поточною датою і часом, статусом "pending" та прапорцем complete встановленим у false. Паралельно система записує інформацію про конкретні препарати у замовленні через таблицю orders_medicine, створюючи записи з order_id та medicine_id для кожного обраного препарату.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		30

Наступним етапом є розрахунок загальної кількості товарів у замовленні шляхом підсумовування всіх позицій з таблиці orders_medicine для конкретного замовлення та оновлення поля quantity в таблиці order. Система також перевіряє статус всіх включених препаратів та може оновити їх demand флаг у таблиці medicine для відстеження популярності товарів.

Після успішного створення замовлення система генерує унікальний номер замовлення на основі oid та повертає клієнту підтвердження з деталями замовлення, включаючи order ID, список включених препаратів, загальну кількість та очікувану дату обробки. Клієнтська частина отримує відповідь та відображає користувачу підтвердження успішного оформлення замовлення з можливістю переходу до перегляду деталей або продовження покупок.

Альтернативні сценарії на діаграмі показують обробку помилок та виняткових ситуацій, таких як неавторизований доступ, неіснуючі препарати, недоступні товари (статус "unavailable") або технічні проблеми з базою даних. Кожен такий сценарій включає відповідні HTTP коди помилок, rollback транзакцій у базі даних та інформування користувача про проблему з поясненням причини та пропозиціями щодо її вирішення.

Додатковий сценарій описує процес оновлення статусу замовлення адміністратором, який включає автентифікацію адміністратора, валідацію прав доступу, перевірку існування замовлення, оновлення полів status, shipDate та complete у таблиці order, а також відправку сповіщення клієнту про зміну статусу замовлення. Цей процес демонструє адміністративні функції системи та забезпечення прозорості для клієнтів щодо стану їх замовлень.

2.3 База даних та модель даних

Модель даних веб-сервісу для аптеки спроектована як реляційна база даних, нормалізована до третьої нормальної форми для забезпечення цілісності даних, мінімізації дублювання інформації та оптимізації продуктивності запитів. Структура бази даних відображає специфіку фармацевтичного бізнесу, включаючи особливості обліку лікарських засобів, управління категоріями

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		31

препаратів, контроль статусів товарів та забезпечення ефективної обробки замовлень. Основними сутностями системи є користувачі, категорії товарів, лікарські засоби, замовлення та зв'язки між замовленнями та препаратами, кожна з яких представлена окремою таблицею з відповідними зв'язками.

Таблиця user служить центральною сутністю для зберігання інформації про всіх користувачів системи, включаючи клієнтів та адміністраторів, як показано у таблиці 2.1.

Таблиця 2.1 - Структура таблиці user

Поле	Розмір поля	Тип	Опис
uid	5	INTEGER PRIMARY KEY	Унікальний ідентифікатор користувача
username	30	VARCHAR NOT NULL	Логін користувача для входу в систему
firstName	30	VARCHAR NOT NULL	Ім'я користувача
lastName	30	VARCHAR NOT NULL	Прізвище користувача
email	30	VARCHAR	Електронна адреса користувача
password	15	VARCHAR NOT NULL	Пароль для автентифікації
phone	15	VARCHAR	Номер телефону користувача

Структура таблиці включає первинний ключ uid типу INTEGER(5) для унікальної ідентифікації користувачів, поле username типу VARCHAR(30) для логіну в системі, поля firstName та lastName типу VARCHAR(30) для зберігання повного імені користувача, поле email типу VARCHAR(30) для електронної пошти, поле password типу VARCHAR(15) для зберігання паролю та поле phone типу VARCHAR(15) для контактного номера телефону. Всі критичні поля позначені як NOT NULL для забезпечення цілісності даних користувачів.

Таблиця category реалізує класифікацію лікарських засобів за категоріями для зручної організації та навігації по каталогу товарів, як представлено в таблиці 2.2.

Таблиця 2.2 - Структура таблиці category

Поле	Розмір поля	Тип	Опис
cid	5	INTEGER PRIMARY KEY	Унікальний ідентифікатор категорії
name	30	VARCHAR NOT NULL	Назва категорії лікарських засобів

						БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата			32

Таблиця містить мінімальний набір полів для ефективного управління категоріями: первинний ключ `cid` типу `INTEGER(5)` для унікальної ідентифікації кожної категорії та поле `name` типу `VARCHAR(30)` для зберігання назви категорії. Просто та ефективна структура дозволяє легко додавати нові категорії лікарських засобів та організовувати каталог продукції за логічними групами.

Таблиця `medicine` є центральною для зберігання інформації про лікарські засоби та фармацевтичні товари, як показано в таблиці 2.3.

Таблиця 2.3 - Структура таблиці `medicine`

Поле	Розмір поля	Тип	Опис
<code>mid</code>	5	<code>INTEGER PRIMARY KEY</code>	Унікальний ідентифікатор лікарського засобу
<code>category</code>	15	<code>INTEGER NOT NULL</code>	Зовнішній ключ на таблицю <code>category</code>
<code>name</code>	30	<code>VARCHAR NOT NULL</code>	Назва лікарського препарату
<code>manufacturer</code>	30	<code>VARCHAR</code>	Виробник препарату
<code>status</code>	30	<code>VARCHAR NOT NULL</code>	Статус препарату (доступний, недоступний)
<code>demand</code>	1	<code>BIT</code>	Показник попиту на препарат

Первинний ключ `mid` типу `INTEGER(5)` забезпечує унікальну ідентифікацію кожного лікарського засобу в системі. Поле `category` типу `INTEGER` з обмеженням `NOT NULL` служить зовнішнім ключем на таблицю `category`, забезпечуючи обов'язкову класифікацію кожного препарату за відповідною категорією. Поле `name` типу `VARCHAR(30)` зберігає назву препарату, поле `manufacturer` типу `VARCHAR(30)` містить інформацію про виробника, поле `status` типу `VARCHAR(30)` відображає поточний статус товару в системі, а поле `demand` типу `BIT` використовується для маркування препаратів з високим попитом або тих, що користуються особливою увагою.

Таблиця `order` моделює замовлення клієнтів та містить всю необхідну інформацію для обробки та відстеження замовлень, як представлено в таблиці 2.4.

Таблиця 2.4 - Структура таблиці `order`

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		33

Поле	Розмір поля	Тип	Опис
oid	5	INTEGER PRIMARY KEY	Унікальний ідентифікатор замовлення
userId	5	INTEGER NOT NULL	Зовнішній ключ на таблицю user
quantity	10	INTEGER NOT NULL	Загальна кількість товарів у замовленні
shipDate	6	DATETIME	Дата та час відправлення замовлення
status	30	VARCHAR NOT NULL	Поточний статус замовлення
complete	1	BIT	Прапорець завершеності замовлення

Структура включає первинний ключ oid типу INTEGER(5) для унікальної ідентифікації замовлення, поле userId типу INTEGER як зовнішній ключ на таблицю user для зв'язку замовлення з конкретним користувачем. Поле quantity типу INTEGER(10) зберігає загальну кількість товарів у замовленні, поле shipDate типу DATETIME(6) фіксує точну дату та час відправлення з мікросекундною точністю, поле status типу VARCHAR(30) відображає поточний етап обробки замовлення, а поле complete типу BIT служить булевим індикатором завершеності замовлення.

Таблиця orders_medicine реалізує зв'язок "багато-до-багатьох" між замовленнями та лікарськими засобами, дозволяючи одному замовленню містити множину різних препаратів, а одному препарату бути включеним у різні замовлення, як показано в таблиці 2.5.

Таблиця 2.5 - Структура таблиці orders_medicine

Поле	Розмір поля	Тип	Опис
order_id	5	INTEGER NOT NULL	Зовнішній ключ на таблицю order
medicine_id	5	INTEGER NOT NULL	Зовнішній ключ на таблицю medicine

Таблиця має складений первинний ключ з двох полів: order_id та medicine_id, що забезпечує унікальність комбінації замовлення та препарату. Поле order_id типу INTEGER служить зовнішнім ключем на таблицю order через поле oid, а поле medicine_id типу INTEGER є зовнішнім ключем на таблицю medicine через поле mid. Така структура дозволяє ефективно моделювати складні замовлення з множиною різних лікарських засобів без дублювання даних.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		34

Зв'язки між таблицями реалізовані через зовнішні ключі, що забезпечують референційну цілісність даних та підтримують складні запити з об'єднаннями таблиць, як зображено на рисунку 2.5.

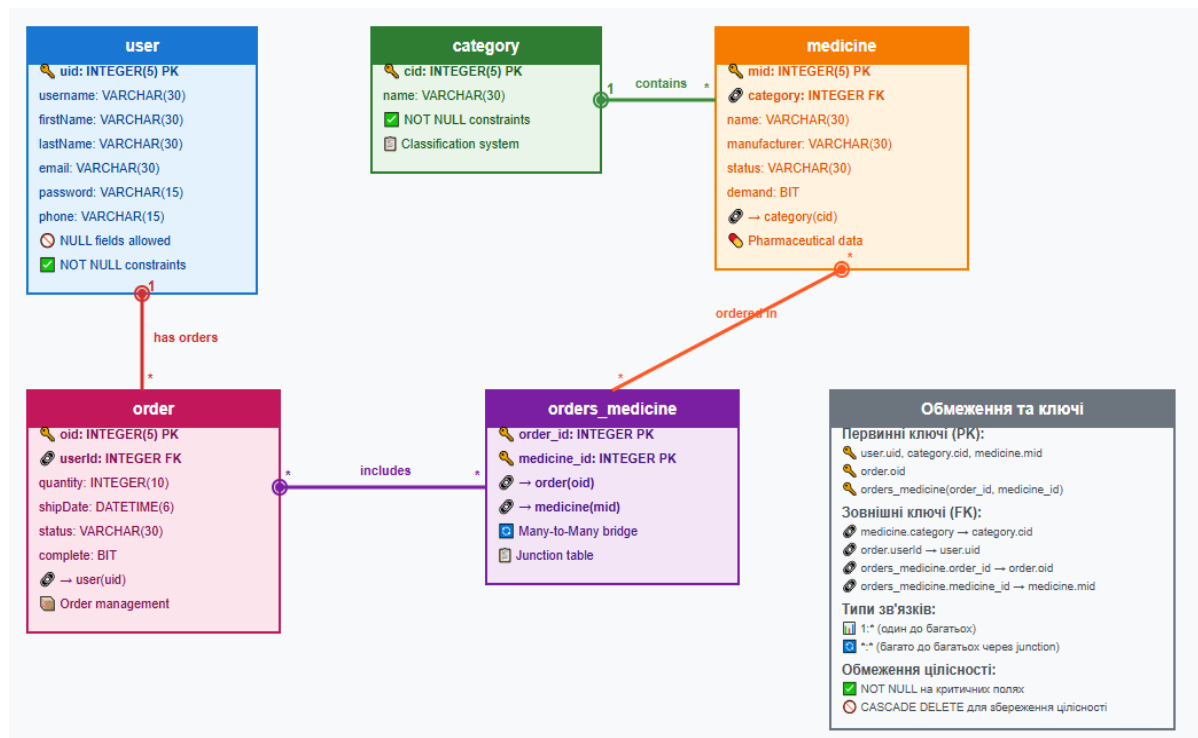


Рисунок 2.5 - Схема зв'язків між таблицями бази даних з первинними та зовнішніми ключами

Основний зв'язок "один-до-багатьох" існує між таблицями `user` та `order`, де один користувач може мати множину замовлень, але кожне замовлення належить тільки одному користувачеві. Аналогічний зв'язок "один-до-багатьох" встановлений між таблицями `category` та `medicine`, дозволяючи групувати лікарські засоби за категоріями. Найскладніший зв'язок "багато-до-багатьох" між таблицями `order` та `medicine` реалізований через проміжну таблицю `orders_medicine`, що дозволяє гнучко моделювати склад замовлень.

Індексна стратегія бази даних включає автоматично створені первинні ключі на всіх таблицях для швидкого доступу до конкретних записів. Додатково рекомендується створення індексів на зовнішні ключі `userId` в таблиці `order` та `category` в таблиці `medicine` для оптимізації операцій об'єднання таблиць. Складений індекс на поля (`order_id`, `medicine_id`) в таблиці `orders_medicine`

автоматично створюється через первинний ключ та забезпечує швидкий пошук зв'язків між замовленнями та препаратами.

Обмеження цілісності даних реалізовані через зовнішні ключі, що запобігають створенню замовлень для неіснуючих користувачів, призначенню препаратів до неіснуючих категорій та додаванню до замовлень неіснуючих лікарських засобів. NOT NULL обмеження на критичних полях забезпечують наявність обов'язкової інформації, такої як імена користувачів, назви препаратів та статуси замовлень.

Типи даних оптимізовані для ефективного зберігання та швидкого доступу: INTEGER(5) для ідентифікаторів забезпечує підтримку до 99999 записів у кожній таблиці, що достатньо для середнього та великого фармацевтичного бізнесу. VARCHAR(30) для текстових полів обмежує довжину рядків для оптимізації пам'яті, INTEGER(10) для quantity дозволяє зберігати великі кількості товарів, DATETIME(6) забезпечує точне відстеження часових подій, а BIT типи ефективно зберігають булеві значення.

Стратегія резервного копіювання повинна включати регулярні повні бекапи всіх таблиць з особливою увагою до таблиць user та order, які містять критично важливу бізнес-інформацію. Інкрементальні копії рекомендується робити для таблиць з високою активністю, таких як order та orders_medicine. Тестування процедур відновлення на окремому середовищі забезпечить готовність до швидкого відновлення роботи системи у випадку збоїв.

Можливості масштабування включають partition таблиці order за датами для покращення продуктивності запитів до історичних даних, архівування старих замовлень.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		36

3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ РОЗРОБЛЕНОГО WEB-SERVISY

3.1 Реалізація клієнтської частини засобами React

Клієнтська частина web-сервісу для аптеки була реалізована з використанням сучасного JavaScript фреймворку React, який забезпечує створення інтерактивних користувацьких інтерфейсів з високою продуктивністю. Архітектура клієнтської частини побудована на принципах односторінкового застосунку (Single Page Application, SPA), що дозволяє забезпечити швидку навігацію між різними розділами системи без перезавантаження сторінки та створює плавний користувацький досвід.

Основою архітектури клієнтської частини є компонентний підхід React, де кожен функціональний блок представлений окремим компонентом з власним станом та життєвим циклом. Головні компоненти системи включають HomePage для відображення головної сторінки з рекомендованими товарами, ProductsPage для каталогу всіх товарів, ProductDetails для детального перегляду окремого препарату, LoginPage та SignUpPage для автентифікації користувачів, AccountPage для управління профілем та ShoppingCart для відображення кошика покупок.

Маршрутизація в застосунку реалізована за допомогою бібліотеки react-router-dom версії 6, яка забезпечує декларативний підхід до навігації та підтримку історії браузера. Конфігурація маршрутів включає захищені роути для автентифікованих користувачів, публічні роути для гостей користувачів та динамічні роути з параметрами для відображення детальної інформації про товари. Компонент NavLink використовується для створення навігаційного меню з активними посиланнями, що забезпечує зручність навігації для користувачів.

Управління станом додатку здійснюється за допомогою вбудованих React хуків useState та useEffect, що дозволяє ефективно керувати локальним станом компонентів та обробляти побічні ефекти. Хук useState використовується для

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		37

зберігання даних про товари, інформацію про користувача, стан завантаження та повідомлення про помилки. Хук useEffect застосовується для виконання API запитів при монтуванні компонентів, оновлення даних при зміні залежностей та очищення ресурсів при демонтажі компонентів.

Взаємодія з серверною частиною реалізована через HTTP клієнт axios, який забезпечує зручний API для виконання асинхронних запитів та обробки відповідей. Axios налаштований з базовим URL серверної частини та автоматичним додаванням JWT токенів до заголовків Authorization для автентифікованих запитів. Перехоплювачі запитів (interceptors) використовуються для централізованої обробки помилок автентифікації та автоматичного перенаправлення користувачів на сторінку входу при закінченні терміну дії токенів.

Система автентифікації та авторизації побудована на JWT токенах, які зберігаються в localStorage браузера та автоматично додаються до всіх API запитів. Контекст автентифікації створений за допомогою React Context API забезпечує глобальний доступ до інформації про поточного користувача та методів входу/виходу з системи. Захищені компоненти перевіряють наявність дійсного токена та відповідні права доступу перед рендерингом контенту.

Адаптивний дизайн реалізований за допомогою CSS медіа-запитів та гнучкої системи сіток, що забезпечує коректне відображення інтерфейсу на різних пристроях від мобільних телефонів до настільних комп'ютерів. Компоненти автоматично адаптуються до розміру екрану, змінюючи розташування елементів, розміри шрифтів та відступи. Мобільна версія включає спрощене навігаційне меню у вигляді "гамбургера" та оптимізовані форми для зручного введення даних на сенсорних екранах.

Система сповіщень користувачів реалізована за допомогою бібліотеки react-toastify, яка забезпечує показ елегантних повідомлень про успішне виконання операцій, помилки та попередження.

Сповіщення автоматично з'являються у верхньому правому кутку екрану та зникають через встановлений час або при натисканні користувачем. Різні

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		38

типи повідомлень мають відповідні кольори та іконки для швидкого розпізнавання користувачами.

Компонент каталогу товарів включає функціональність пошуку, фільтрації та сортування препаратів за різними критеріями. Пошук реалізований як контрольований компонент з debouncing для оптимізації кількості API запитів. Фільтри дозволяють користувачам обирати товари за категоріями, цінovими діапазонами та статусом наявності. Пагінація забезпечує ефективне відображення великих списків товарів з можливістю вибору кількості елементів на сторінці.

Детальна сторінка товару містить повну інформацію про препарат, включаючи зображення, опис, інструкції по застосуванню, протипоказання та інформацію про наявність на складі. Функціональність додавання товару до кошика включає валідацію кількості з перевіркою доступних залишків на складі. Інтерактивні елементи дозволяють користувачам вибирати кількість товару та миттєво бачити розрахункову вартість.

Відображення проводиться через компонент *ProductCard.jsx*:

```
import React from 'react';

const ProductCard = ({ product }) => {
  return (
    <div className="card">
      <h3>{product.name}</h3>
      <p>{product.description}</p>
      <p>Ціна: {product.price} грн</p>
      <button>Додати в корзину</button>
    </div>
  );
};

export default ProductCard;
```

Кошик покупок реалізований як окремий компонент з можливістю перегляду

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		39

обраних товарів, зміни кількості позицій та видалення непотрібних елементів. Автоматичний розрахунок загальної вартості оновлюється при будь-яких змінах у складі кошика. Стан кошика синхронізується з сервером для зареєстрованих користувачів та зберігається локально для гостей користувачів до моменту автентифікації. Для перегляду вмісту кошика використовується компонент *Cart.jsx*:

```
import React from 'react';

const Cart = ({ items }) => {
  const total = items.reduce((sum, item) => sum + item.price *
item.quantity, 0);

  return (
    <div>
      <h2>Корзина</h2>
      <ul>
        {items.map(item => (
          <li key={item.id}>
            {item.name} - {item.quantity} шт - {item.price *
item.quantity} грн
          </li>
        ))}
      </ul>
      <p>Сума: {total} грн</p>
    </div>
  );
};

export default Cart;
```

Особистий кабінет користувача включає форми для редагування персональної інформації з валідацією введених даних та відображення історії замовлень. Реактивні форми побудовані на контрольованих компонентах з

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		40

використанням хуків для управління станом полів та валідації. Зміни в профілі миттєво відображаються в інтерфейсі після успішного збереження на сервері.

Система ролей користувачів реалізована через умовний рендеринг компонентів залежно від ролі поточного користувача. Адміністратори мають доступ до додаткових функцій управління товарами, категоріями та користувачами через спеціальні адміністративні компоненти. Звичайні користувачі бачать тільки функції, доступні для їх ролі, що забезпечує безпеку та зручність використання системи.

Чат-система для консультацій реалізована з використанням WebSocket з'єднань через бібліотеку socket.io-client, що забезпечує двонаправлений обмін повідомленнями в реальному часі. Компонент чату включає список повідомлень, форму для введення нових повідомлень та індикатори статусу підключення. Повідомлення автоматично прокручуються до останнього елементу та підтримують базове форматування тексту.

Оптимізація продуктивності клієнтської частини включає lazy loading компонентів для зменшення початкового розміру bundle, мемоізацію дорогих обчислень за допомогою useMemo та useCallback хуків, а також виртуалізацію довгих списків товарів. Обробка помилок реалізована через error boundaries для перехоплення та graceful обробки JavaScript помилок без краху всього додатку.

3.2 Реалізація серверної частини засобами Python

Серверна частина web-сервісу для аптеки розроблена з використанням мови програмування Python та сучасного веб-фреймворку FastAPI, який забезпечує високу продуктивність, автоматичну генерацію документації API та вбудовану підтримку асинхронного програмування. Архітектура серверної частини побудована за принципами REST API з чіткою структурою ендпоінтів для різних ресурсів системи та дотриманням стандартів HTTP протоколу для різних типів операцій. Приклад маршруту авторизації (FastAPI):

```
from fastapi import FastAPI, HTTPException, Depends
```

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		41

```

from fastapi.security import OAuth2PasswordRequestForm
from auth import authenticate_user, create_access_token

app = FastAPI()

@app.post("/token")
def login(form_data: OAuth2PasswordRequestForm = Depends()):
    user = authenticate_user(form_data.username,
form_data.password)
    if not user:
        raise HTTPException(status_code=400, detail="Incorrect
username or password")
    access_token = create_access_token(data={"sub":
user.username})
    return {"access_token": access_token, "token_type":
"bearer"}

```

Основна структура проекту організована за модульним принципом з розділенням функціональності на окремі модулі: `models` для визначення моделей даних, `schemas` для Pydantic схем валідації, `routers` для групування ендпоінтів за функціональністю, `services` для бізнес-логіки, `database` для конфігурації та роботи з базою даних, `auth` для системи автентифікації та `utils` для допоміжних функцій. Така організація забезпечує зручність розробки, тестування та подальшої підтримки коду.

База даних реалізована з використанням реляційної СУБД через SQLAlchemy ORM, що забезпечує об'єктно-реляційне відображення та абстракцію від конкретної реалізації бази даних. Моделі даних включають `User` для користувачів, `Medicine` для лікарських засобів, `Category` для категорій товарів, `Order` для замовлень та `OrderMedicine` для зв'язку між замовленнями та товарами. Кожна модель включає відповідні поля, зв'язки між таблицями та валідацію на рівні бази даних. Приклад моделі «Medicine» SQLAlchemy:

```

from sqlalchemy import Column, Integer, String

```

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		42

```

from database import Base

class Medicine(Base):
    __tablename__ = "medicines"

    id = Column(Integer, primary_key=True, index=True)
    name = Column(String(100), nullable=False)
    description = Column(String(250))
    price = Column(Integer, nullable=False)
    category_id = Column(Integer)

```

Система автентифікації та авторизації реалізована з використанням JSON Web Tokens (JWT), що забезпечує stateless автентифікацію без необхідності зберігання сесій на сервері. JWT токени містять закодовану інформацію про користувача та його права доступу з встановленим терміном дії для забезпечення безпеки. Middleware автентифікації автоматично перевіряє валідність токенів для захищених ендпоінтів та додає інформацію про поточного користувача до контексту запиту.

Ендпоінти для управління користувачами включають реєстрацію нових користувачів з валідацією унікальності email та хешуванням паролів за допомогою bcrypt, автентифікацію з перевіркою облікових даних та генерацією JWT токенів, отримання інформації про поточного користувача, оновлення профілю з перевіркою прав доступу та видалення облікових записів з каскадним видаленням пов'язаних даних. Всі операції з паролями використовують криптографічно стійкі алгоритми хешування.

API для управління лікарськими засобами забезпечує повний CRUD функціонал з ендпоінтами для створення нових товарів, отримання списку всіх препаратів з можливістю фільтрації та пагінації, отримання детальної інформації про конкретний товар, оновлення інформації про препарати та видалення товарів з перевіркою наявності в активних замовленнях. Кожен ендпоінт включає відповідну валідацію вхідних даних та обробку помилок.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		43

Система управління замовленнями реалізована через комплекс ендпоінтів для створення нових замовлень з автоматичною перевіркою наявності товарів на складі, додавання та видалення товарів з існуючих замовлень, оновлення статусів замовлень адміністраторами, отримання історії замовлень користувачів та розрахунку загальної вартості замовлень. Бізнес-логіка включає перевірку достатності товарів на складі та автоматичне резервування при створенні замовлень. Приклад обробки запиту на отримання всіх товарів:

```
from fastapi import APIRouter, Depends
from sqlalchemy.orm import Session
from models import Medicine
from database import get_db

router = APIRouter()

@router.get("/medicines")
def read_medicines(skip: int = 0, limit: int = 10, db: Session = Depends(get_db)):
    return db.query(Medicine).offset(skip).limit(limit).all()
```

Валідація вхідних даних реалізована за допомогою Pydantic схем, які забезпечують автоматичну перевірку типів даних, форматів email адрес, довжини паролів та інших бізнес-правил. Схеми також включають серіалізацію вихідних даних з можливістю виключення чутливої інформації, такої як хеші паролів, з відповідей API. Кастомні валідатори додають специфічні перевірки для фармацевтичної галузі, такі як формати номерів лікарських засобів.

Обробка помилок організована через систему кастомних винятків з відповідними HTTP статус кодами та детальними повідомленнями для клієнтських додатків. Exception handlers автоматично перехоплюють різні типи помилок та повертають структуровані JSON відповіді з інформацією про причину помилки. Логування помилок забезпечує можливість моніторингу та діагностики проблем у продакшн середовищі.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		44

Middleware для CORS (Cross-Origin Resource Sharing) налаштований для забезпечення безпечної взаємодії з клієнтськими додатками, що працюють на різних доменах. Конфігурація дозволяє специфічні домени для продакшн середовища та всі домени для розробки. Додатковий middleware для логування запитів записує інформацію про всі API виклики для аналізу використання та виявлення потенційних проблем безпеки.

WebSocket сервер для чат-системи реалізований за допомогою Socket.IO бібліотеки, що забезпечує надійне двонаправлене з'єднання між клієнтами та сервером. Сервер підтримує кімнати для групування користувачів за ролями, автоматичне сповіщення про підключення та відключення користувачів, збереження історії повідомлень та модерацію контенту. Події чату інтегровані з системою автентифікації для ідентифікації відправників повідомлень.

Система ролей та дозволів реалізована через декоратори, які перевіряють права доступу користувачів до конкретних ендпоінтів. Базові ролі включають guest для неавтентифікованих користувачів, user для зареєстрованих клієнтів та admin для адміністраторів системи. Кожна роль має відповідний набір дозволів, що перевіряються автоматично при доступі до захищених ресурсів.

Конфігурація додатку організована через змінні оточення з використанням Pydantic Settings для валідації та типізації конфігураційних параметрів. Налаштування включають рядок підключення до бази даних, секретний ключ для JWT токенів, налаштування CORS, параметри логування та інші операційні параметри. Різні профілі конфігурації дозволяють легко перемикатися між середовищами розробки, тестування та продакшн.

Асинхронне програмування використовується для операцій вводу-виводу, таких як запити до бази даних та зовнішніх API, що значно підвищує продуктивність сервера при високому навантаженні. Async/await паттерн застосовується у всіх ендпоінтах та сервісних функціях для неблокуючого виконання операцій. Connection pooling забезпечує ефективне використання з'єднань з базою даних.

Документація API автоматично генерується FastAPI на основі типів даних та docstring коментарів у коді. Swagger UI та ReDoc інтерфейси забезпечують

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		45

інтерактивне тестування ендпоінтів безпосередньо з браузера. Детальні описи параметрів, схем запитів та відповідей допомагають фронтенд розробникам швидко інтегруватися з API.

Безпека додатку включає захист від основних типів атак, таких як SQL ін'єкції через використання параметризованих запитів ORM, XSS атаки через валідацію та екранування вхідних даних, CSRF атаки через stateless автентифікацію з JWT токенами та rate limiting для запобігання DDoS атакам. Чутливі дані зберігаються у хешованому вигляді з використанням криптографічно стійких алгоритмів.

3.3 Опис функціональних можливостей веб-застосунку

Веб-застосунок для аптеки надає комплексний набір функціональних можливостей, що покривають всі аспекти діяльності сучасного фармацевтичного закладу від презентації товарів до управління замовленнями та консультування клієнтів. Система підтримує диференційований доступ для різних категорій користувачів, забезпечуючи персоналізований досвід взаємодії залежно від ролі та потреб конкретного користувача. Функціональна архітектура побудована з урахуванням специфічних вимог фармацевтичної галузі, включаючи необхідність контролю термінів придатності, управління рецептурними препаратами та забезпечення трасованості лікарських засобів.

Головна сторінка системи служить центральним вузлом для презентації основної інформації про аптеку та її асортимент, як показано на рисунку 3.1.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		46



Howdy ho

Our store is more than just another average online retailer. We sell not only top quality products, but give our customers a positive online shopping experience.

EXPLORE NOW →



Featured products



Кнопина валеріана
40 грн



Omega-3
227 грн



Олідеприм 2000 капсули
481 грн



Solgar Таблетки для шкіри
808 грн



Durex Classic
55 грн



Contex XXL
278 грн



Воро пріст
36 грн



Hebiter
204 грн

Рисунок 3.1 - Головна сторінка веб-сервісу з каталогом рекомендованих товарів та навігаційним меню

Функціональність головної сторінки включає відображення рекомендованих та популярних товарів, навігаційне меню для доступу до різних розділів системи, форми швидкого пошуку препаратів та блоки з корисною інформацією для клієнтів. Адаптивний дизайн головної сторінки автоматично підлаштовується під різні розміри екрану, як показано на рисунку 3.2.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		47

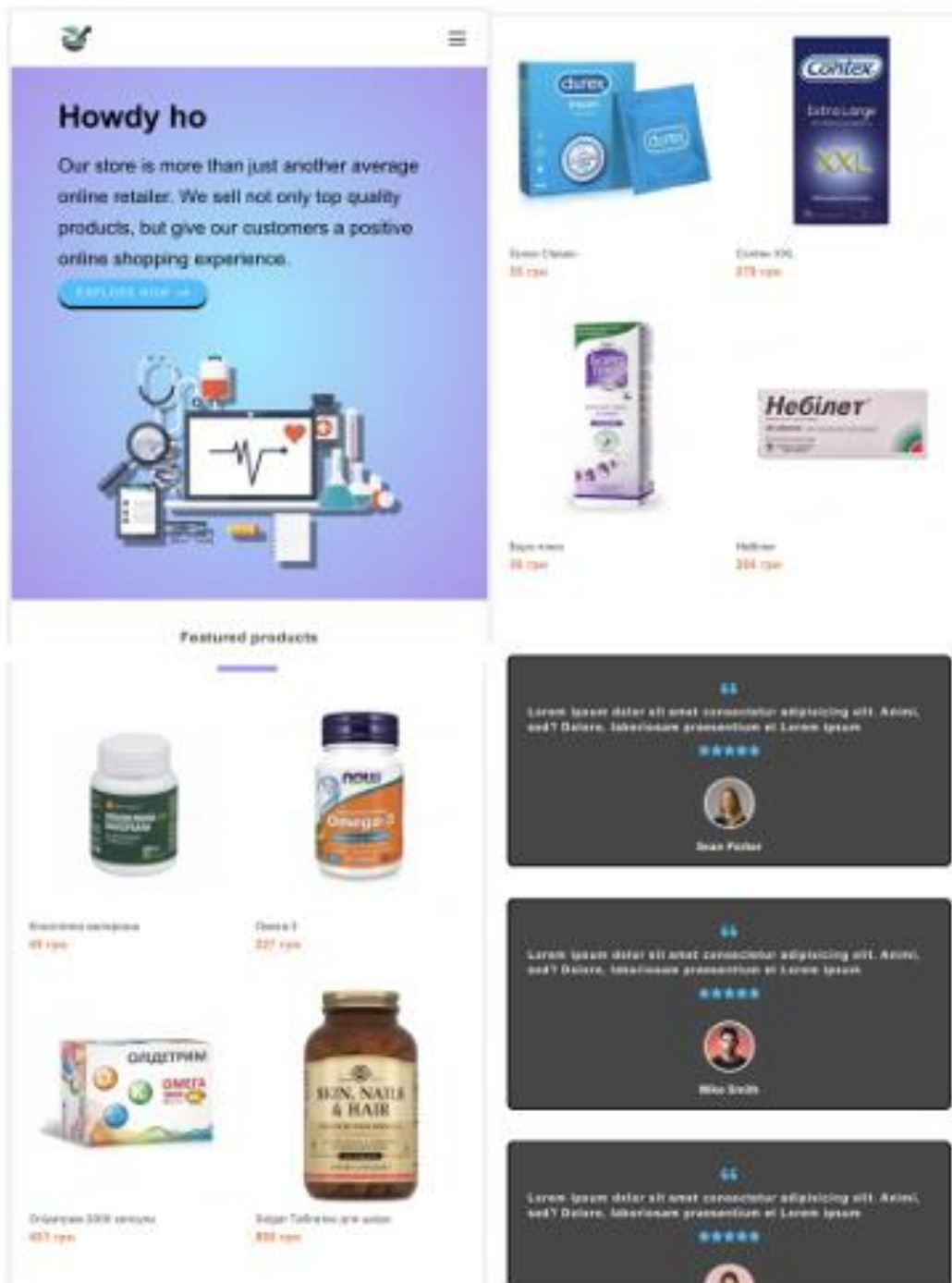


Рисунок 3.2 - Адаптивна версія головної сторінки для мобільних пристроїв

Каталог товарів являє собою центральну функціональну частину системи, що дозволяє користувачам переглядати, фільтрувати та шукати лікарські засоби та супутні товари, як зображено на рисунку 3.3.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		48

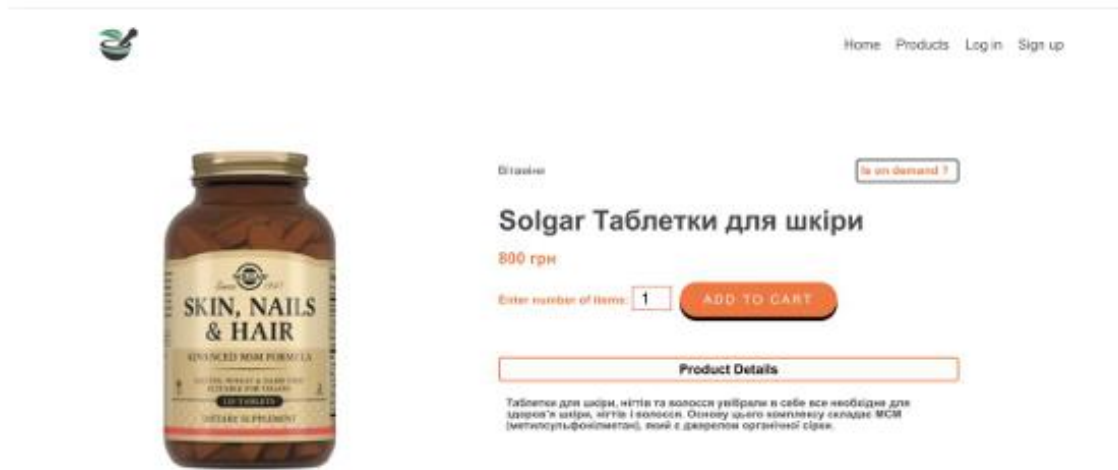


Рисунок 3.4 - Детальна сторінка товару з описом та можливістю додавання до кошика

Функціональні можливості включають відображення детального опису препарату, інформації про виробника, інструкції по застосуванню, можливості аналогів та супутніх товарів, а також форми для додавання товару до кошика або списку бажань. Система автоматично перевіряє наявність товару на складі та інформує користувача про можливі терміни доставки або самовивозу.

Система автентифікації та авторизації забезпечує безпечний доступ користувачів до персоналізованих функцій системи (рис.3.5).

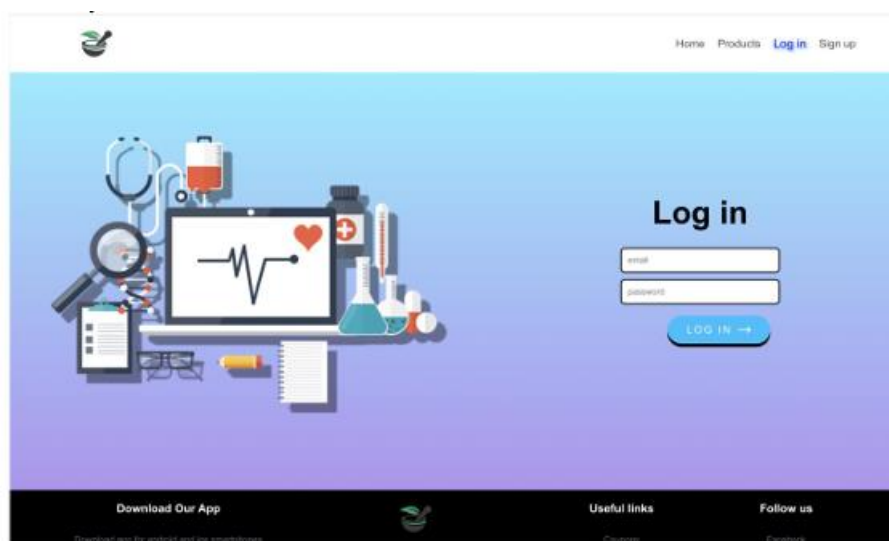


Рисунок 3.5 - Форма автентифікації користувачів

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		50

Функціональність включає реєстрацію нових користувачів з валідацією введених даних, автентифікацію за допомогою електронної пошти та пароллю, відновлення пароллю через електронну пошту, а також управління профілем користувача. Система підтримує різні ролі користувачів з відповідними правами доступу: гостьові користувачі можуть переглядати каталог та інформацію про товари, зареєстровані клієнти отримують доступ до функцій замовлення та персональної історії покупок, а адміністратори мають повний доступ до управління системою.

Особистий кабінет користувача надає централізований доступ до всіх персональних функцій та налаштувань, як зображено на рисунку 3.6.



Рисунок 3.6 - Особистий кабінет користувача з персональною інформацією та можливостями управління акаунтом

Функціональні можливості кабінету включають перегляд та редагування особистої інформації, управління адресами доставки, перегляд історії замовлень, відстеження статусу поточних замовлень, управління списками бажань та збережених товарів, налаштування сповіщень та преференцій комунікації. Інтерфейс кабінету адаптується під роль користувача, показуючи релевантні функції та приховуючи недоступні опції.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		51

Процес реєстрації нових користувачів структурований як інтуїтивна форма з валідацією всіх введених даних, як показано на рисунку 3.7.

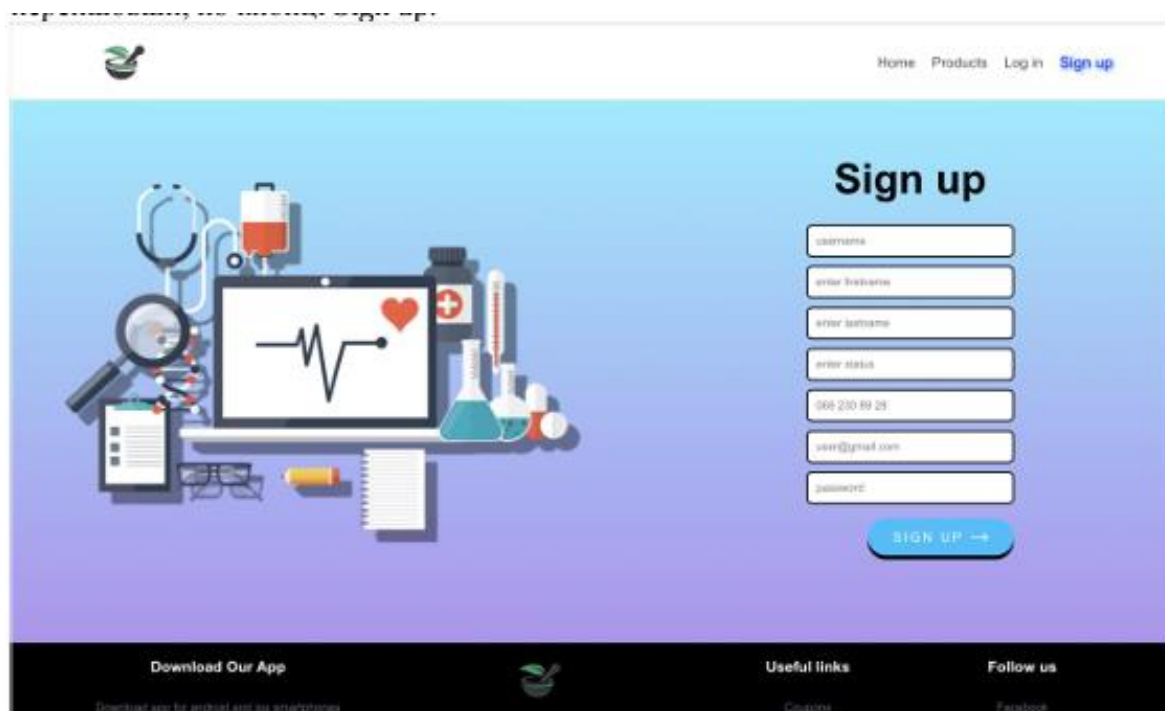


Рисунок 3.7 - Форма реєстрації нових користувачів з полями для введення особистої інформації

Функціональність включає перевірку унікальності електронної пошти, валідацію формату номера телефону, перевірку надійності паролю та підтвердження згоди з умовами використання. Система автоматично відправляє email підтвердження для активації облікового запису та запобігання створенню фіктивних профілів.

Взаємодія з серверною частиною через API здійснюється за допомогою RESTful архітектури з використанням JWT токенів для автентифікації, як демонструє рисунок 3.8.

Всі запити включають необхідні заголовки авторизації та проходять валідацію на серверному рівні перед обробкою. Система забезпечує детальне логування всіх API запитів для моніторингу та діагностики проблем.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		52

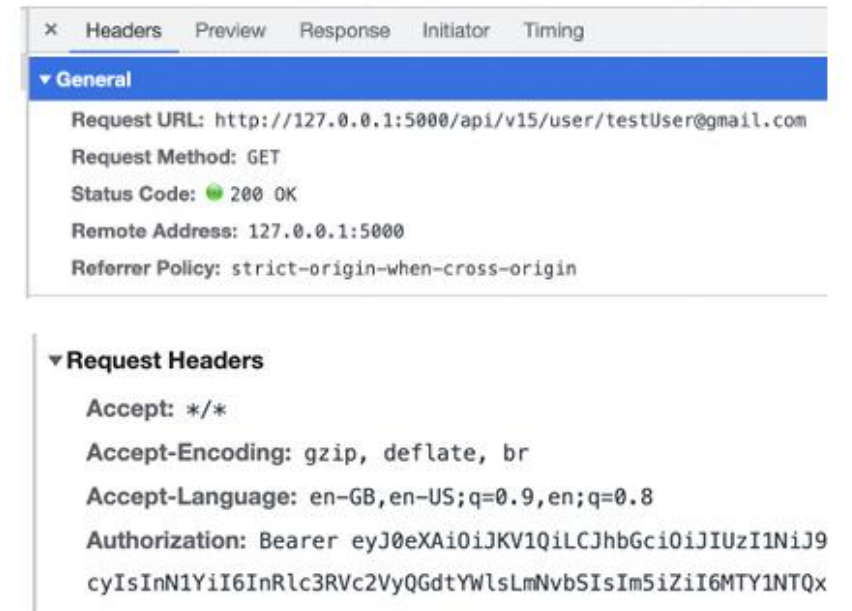


Рисунок 3.8 - HTTP запит API з JWT авторизацією в інструментах розробника браузера

Функціональність редагування профілю дозволяє користувачам оновлювати свою особисту інформацію з миттєвою валідацією змін, як представлено на рисунку 3.9.



Рисунок 3.9 - Інтерфейс редагування профілю користувача з формами для оновлення даних

Система автоматично виходить користувача з системи при зміні критичних даних, таких як електронна пошта, для забезпечення безпеки облікового запису. Всі зміни профілю логуються для audit trail та відстеження несанкціонованих модифікацій.

Адміністративні функції управління товарами надають повний контроль над каталогом продукції, як зображено на рисунку 3.10.



Рисунок 3.10 - Адміністративна панель управління товарами з можливостями додавання та редагування

Функціональність включає додавання нових товарів з завантаженням зображень, редагування існуючих позицій, управління категоріями та характеристиками, встановлення цін та знижок, контроль залишків на складі та видалення товарів з відповідними перевітками цілісності даних. Система автоматично перевіряє наявність товару в активних замовленнях перед дозволом видалення.

База даних користувачів демонструє структуру зберігання облікових записів з різними ролями та статусами, як показано на рисунку 3.11.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		54

uid	userstatus	username	firstname	lastname	email	password	phone
1	user	user1	Vlad	Diachyk	user1@gmail.com	\$2b\$12\$Nmph6eY1qWzMghNHHqnpHuj0x64h6...	0880777899
2	user	user2	Bob	Marlin	user2@gmail.com	\$2b\$12\$QUfnaDU/yZytb9IAwljXv.0.yfSwwAbU...	0980722899
1318	admin	admin1	Ivan	Petrenko	admin1@gmail.com	\$2b\$12\$bMC0Tr8rKgXrOEa17jc5v.cbUYWUdhIG...	0999309899
1319	admin	admin2	Petro	Ivanenko	admin2@gmail.com	\$2b\$12\$NCXg1LnTMadnZpI3L/svqeLu5hTOWN...	0880009899
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рисунок 3.11 - Структура таблиці користувачів у базі даних з основними полями

Таблиця містить всю необхідну інформацію для автентифікації, авторизації та персоналізації досвіду користувачів, включаючи зашифровані паролі, контактну інформацію та мета-дані облікових записів.

Функціональність додавання товарів до замовлення включає вибір кількості товару з автоматичною перевіркою наявності на складі, як представлено на рисунку 3.12.



Рисунок 3.12 - Процес додавання товару до замовлення з валідацією кількості та перевіркою наявності

Система запобігає замовленню більшої кількості товару, ніж доступно, та надає відповідні повідомлення про обмеження. При успішному додаванні товару система створює або оновлює замовлення користувача та резервує відповідну кількість на складі.

Кошик покупок дозволяє користувачам переглядати та управляти обраними товарами перед оформленням замовлення, як зображено на рисунку 3.13.

						БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата			55



Рисунок 3.13 - Кошик покупок з можливістю управління товарами та розрахунком загальної вартості

Функціональність включає зміну кількості позицій, видалення товарів, автоматичний розрахунок загальної вартості та валідацію наявності всіх товарів. Система зберігає стан кошика між сесіями для зареєстрованих користувачів та синхронізує зміни в реальному часі.

Чат-система для консультацій реалізована з використанням WebSocket технології для комунікації в реальному часі, як показано на рисунку 3.14.

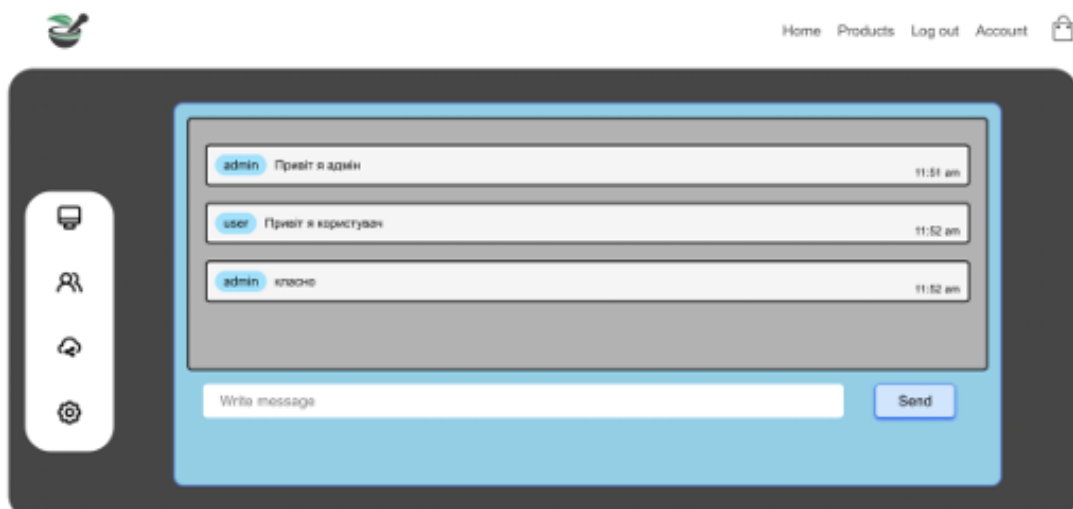


Рисунок 3.14 - Інтерфейс чат-консультацій між клієнтами та фармацевтами

Функціональність включає автоматичне підключення клієнтів до доступних фармацевтів, обмін повідомленнями з підтримкою емодзі, індикатори статусу учасників, збереження історії розмов та можливості модерації чатів адміністраторами. Система автоматично сповіщає фармацевтів про нові запити на консультацію.

3.4 Тестування і демонстрація роботи проєкту

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		56

Процес тестування розробленого web-сервісу для аптеки включав комплексну перевірку всіх функціональних можливостей системи, тестування продуктивності та безпеки, а також валідацію відповідності вимогам технічного завдання. Методологія тестування базувалася на принципах модульного тестування окремих компонентів, інтеграційного тестування взаємодії між різними частинами системи та end-to-end тестування повних користувацьких сценаріїв від початку до завершення.

Основні інструменти для тестування:

- Postman – для перевірки коректності роботи REST API;
- Swagger UI – для інтерактивного тестування HTTP-запитів;
- Google Chrome DevTools – для аналізу клієнтської частини;
- SQLite Browser – перегляд змін у базі даних.

Нижче наведено таблицю 3.1, що містить перелік тест-кейсів, які були використані для перевірки працездатності веб-застосунку. Кожен тест-кейс описує конкретну дію користувача, вхідні дані, очікуваний результат та фактичний статус виконання. Результати тестування дозволили переконатися у стабільності основного функціоналу системи та відповідності реалізації заявленим вимогам.

Таблиця 3.1 – Тест-кейси для ключових функцій системи

№	Назва тесту	Опис дії	Вхідні дані	Очікуваний результат	Статус
1	Ресстрація нового користувача	Введення валідних даних і створення облікового запису	username, email, password	Користувача зареєстровано, перенаправлення на головну	Пройдено
2	Невірне підтвердження паролю	Введено різні значення паролів	username, email, 2 різні паролі	Виведено повідомлення «Паролі не збігаються»	Пройдено
3	Авторизація з правильними даними	Вхід до системи через логін і пароль	username, password	Вхід успішний, відкрито головну сторінку	Пройдено

Кінець таблиці 3.1

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		57

4	Авторизація з хибним паролем	Вхід із неправильним паролем	username, невірний password	Повідомлення «Невірний пароль!»	Пройдено
5	Перегляд товарів	Перехід до списку доступних товарів	—	Відображено список товарів	Пройдено
6	Додавання товару (адмін)	Додавання нового товару через адмін-панель	Назва, опис, ціна, категорія	Товар збережено в базі та відображено у списку	Пройдено
7	Оформлення замовлення	Покупка товару з кошика	Товари в кошику	Замовлення підтверджено, запис збережено	Пройдено
8	Видалення товару (адмін)	Видалення існуючого товару	ID товару	Товар видалено з бази та списку	Пройдено
9	Відображення помилки при вході	Введення неіснуючого користувача	login	Повідомлення «Користувача не знайдено»	Пройдено

Модульне тестування клієнтської частини проводилося з використанням фреймворку Jest та бібліотеки React Testing Library для тестування React компонентів. Тести покривали основну функціональність компонентів, включаючи правильність рендерингу елементів інтерфейсу, обробку користувацьких подій, валідацію форм та управління станом. Особлива увага приділялася тестуванню критичних компонентів, таких як форми автентифікації, кошик покупок та детальні сторінки товарів.

Тестування серверної частини включало перевірку всіх API ендпоінтів з використанням ruytest фреймворку та httpx клієнта для симуляції HTTP запитів. Тести покривали успішні сценарії виконання операцій, обробку помилкових ситуацій, валідацію вхідних даних та перевірку прав доступу. Окремі тести були створені для перевірки бізнес-логіки, такої як розрахунок вартості замовлень, перевірка наявності товарів на складі та управління статусами замовлень.

Тестування бази даних включало перевірку цілісності даних, коректності виконання міграцій та ефективності SQL запитів. Використовувалася тестова база даних SQLite для швидкого виконання тестів без впливу на продакшн дані. Тести покривали всі CRUD операції для кожної моделі даних, перевірку зовнішніх ключів та каскадного видалення, а також валідацію унікальних обмежень.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		58

Інтеграційне тестування фокусувалося на перевірці взаємодії між клієнтською та серверною частинами через API інтерфейс. Тести симулювали реальні користувацькі сценарії, такі як реєстрація нового користувача, вхід в систему, перегляд каталогу товарів, додавання товарів до кошика та оформлення замовлення. Особлива увага приділялася тестуванню системи автентифікації та авторизації з перевіркою JWT токенів.

End-to-end тестування проводилося з використанням Cypress фреймворку для автоматизації взаємодії з веб-браузером. Тести покривали повні користувацькі сценарії від відкриття головної сторінки до завершення покупки, включаючи навігацію між сторінками, заповнення форм, взаємодію з елементами інтерфейсу та перевірку відображення результатів операцій. Тести виконувалися в різних браузерах для забезпечення кросбраузерної сумісності.

Тестування адаптивного дизайну включало перевірку коректного відображення інтерфейсу на різних розмірах екранів від 320px (мобільні телефони) до 1920px (настільні монітори). Використовувалися інструменти розробника браузера для симуляції різних пристроїв та орієнтацій екрану. Особлива увага приділялася зручності використання на мобільних пристроях, включаючи розміри кнопок, відступи між елементами та читабельність тексту.

Тестування продуктивності проводилося з використанням інструментів навантаження для перевірки поведінки системи під високим навантаженням. Artillery використовувався для генерації навантаження на API ендпоінти з симуляцією одночасних запитів від множини користувачів. Lighthouse аудит забезпечував оцінку продуктивності клієнтської частини, включаючи час завантаження сторінок, метрики Core Web Vitals та оптимізацію ресурсів.

Тестування безпеки включало перевірку захищеності від основних типів веб-атак, таких як SQL ін'єкції, XSS атаки, CSRF атаки та несанкціонований доступ до ресурсів. Використовувалися автоматизовані інструменти сканування безпеки для виявлення потенційних вразливостей. Додатково проводилося ручне тестування спроб обходу системи автентифікації та авторизації.

Демонстрація функціональності головної сторінки показала коректне відображення рекомендованих товарів, навігаційного меню та інформаційних

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		59

блоків. Адаптивна версія головної сторінки продемонструвала автоматичне підлаштування під різні розміри екранів з збереженням зручності навігації. Тестування показало швидке завантаження зображень товарів та плавну анімацію переходів між елементами інтерфейсу.

Тестування сторінки каталогу товарів підтвердило працездатність функцій пошуку, фільтрації та сортування препаратів. Демонстрація показала швидку відповідь системи на зміни фільтрів та коректне відображення результатів пошуку. Пагінація працювала стабільно з великими наборами даних, забезпечуючи швидке завантаження наступних сторінок результатів.

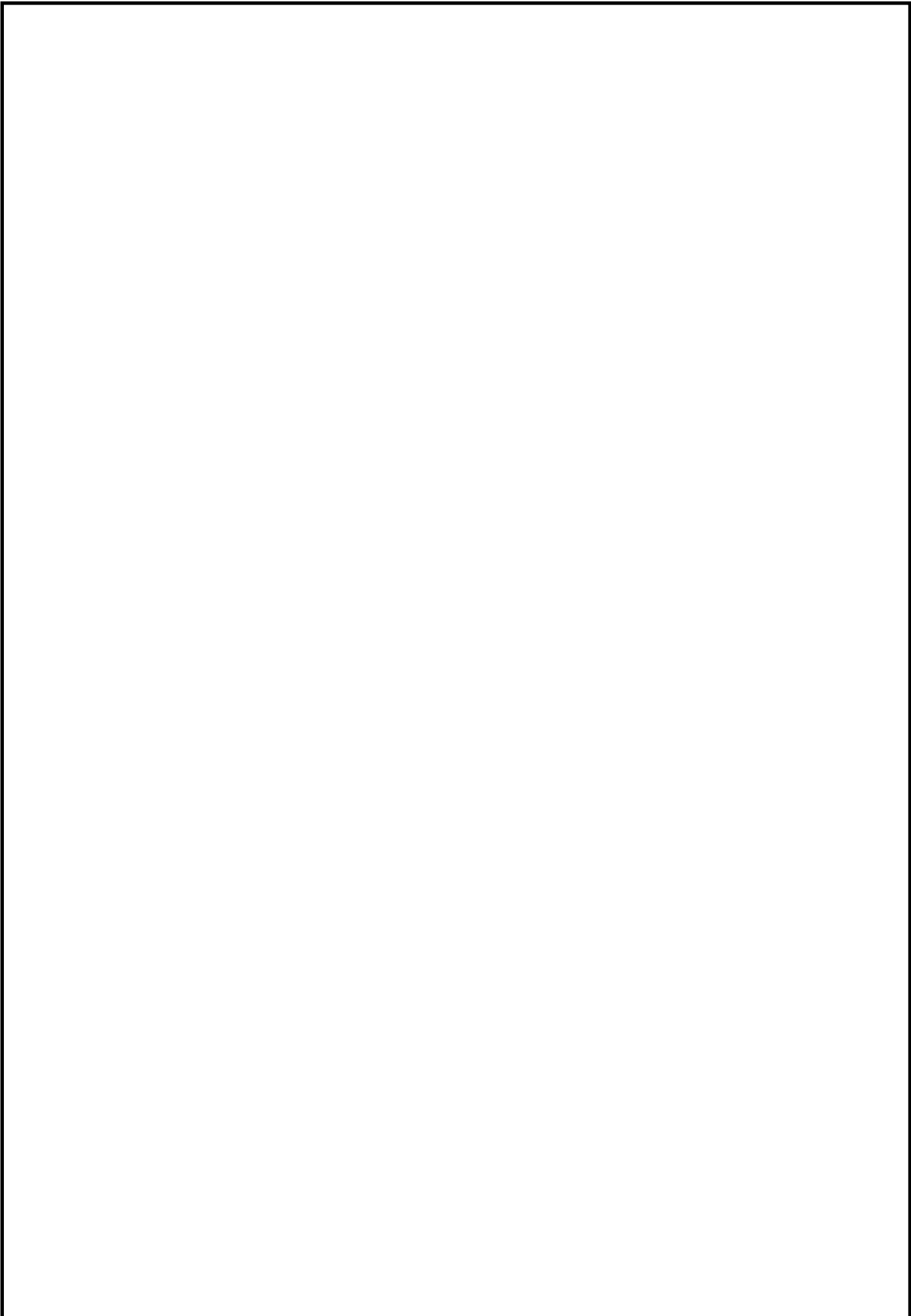
Тестування системи автентифікації та авторизації продемонструвало надійну роботу процесів реєстрації та входу в систему. Форми валідації коректно перевіряли введені дані та відображали зрозумілі повідомлення про помилки. Система ролей працювала належним чином, обмежуючи доступ неавторизованих користувачів до захищених функцій та надаючи відповідні права адміністраторам.

Демонстрація особистого кабінету користувача показала функціональність редагування профілю з миттєвою валідацією змін та безпечним оновленням даних на сервері. Тестування підтвердило коректну роботу функцій видалення облікового запису з каскадним видаленням пов'язаних даних та автоматичним виходом з системи.

Тестування функціональності управління товарами для адміністраторів продемонструвало можливості додавання нових препаратів, редагування існуючих товарів та видалення позицій з перевіркою залежностей. Система коректно запобігала видаленню товарів, що присутні в активних замовленнях, та відображала відповідні повідомлення про помилки.

Демонстрація системи замовлень показала повний цикл від додавання товарів до кошика до оформлення замовлення. Тестування підтвердило коректну роботу валідації кількості товарів відносно наявності на складі, автоматичного розрахунку загальної вартості та синхронізації стану кошика між сесіями користувача. Функції видалення товарів з кошика працювали стабільно з миттєвим оновленням загальної суми.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		60



					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		61

ВИСНОВКИ

У ході виконання бакалаврської роботи було успішно розроблено функціональний web-сервіс для обліку фармацевтичних товарів, який повністю відповідає поставленим цілям та завданням дослідження. Створений програмний продукт демонструє ефективне поєднання сучасних веб-технологій для вирішення актуальних завдань автоматизації діяльності аптечних закладів та покращення якості фармацевтичного обслуговування населення.

Проведений аналіз нормативної бази фармацевтичної діяльності в Україні показав складність регулятивного середовища та високі вимоги до інформаційних систем у цій галузі. Вивчення існуючих рішень для управління фармацевтичними товарами виявило потребу в доступних, гнучких та легко адаптованих web-сервісах для невеликих та середніх аптечних закладів. Визначені інформативні ознаки інформаційної системи стали основою для формування технічних вимог до розроблюваного програмного продукту.

Розроблена архітектура web-сервісу базується на сучасних принципах побудови веб-додатків з чітким розділенням клієнтської та серверної частин, що забезпечує масштабованість, підтримуваність та можливість подальшого розвитку системи. Застосування RESTful API архітектури дозволяє ефективну взаємодію між компонентами системи та потенційну інтеграцію з зовнішніми сервісами. Створені UML-діаграми повно відображають структуру системи та взаємодію її компонентів.

Спроектowana модель бази даних забезпечує надійне зберігання інформації про користувачів, товари, категорії та замовлення з дотриманням принципів нормалізації та забезпеченням цілісності даних. Реляційна структура бази даних дозволяє ефективно виконувати складні запити та підтримувати бізнес-логіку фармацевтичного закладу. Використання зовнішніх ключів та обмежень цілісності забезпечує консистентність даних навіть при інтенсивному використанні системи.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		62

Клієнтська частина, реалізована засобами React фреймворку, демонструє сучасний підхід до створення інтерактивних користувацьких інтерфейсів з високою продуктивністю та зручністю використання. Компонентна архітектура забезпечує можливість повторного використання коду та легкість тестування окремих елементів інтерфейсу. Адаптивний дизайн гарантує коректне відображення на різних пристроях, що критично важливо для сучасних веб-додатків.

Серверна частина, розроблена на Python з використанням FastAPI фреймворку, забезпечує високу продуктивність обробки запитів та автоматичну генерацію документації API. Асинхронна архітектура дозволяє ефективно обробляти одночасні запити від множини користувачів. Система автентифікації та авторизації на базі JWT токенів забезпечує безпечний доступ до ресурсів системи з підтримкою різних ролей користувачів.

Реалізована функціональність системи включає повний цикл роботи з фармацевтичними товарами від додавання до каталогу до обробки замовлень клієнтів. Система управління користувачами підтримує різні ролі з відповідними правами доступу, що забезпечує гнучкість у використанні різними категоріями персоналу аптеки. Інтегрована чат-система для консультацій демонструє можливість використання сучасних технологій реального часу для покращення сервісу клієнтів.

Проведене комплексне тестування підтвердило надійність та стабільність роботи всіх компонентів системи. Модульні тести забезпечили перевірку коректності роботи окремих функцій, інтеграційні тести підтвердили правильність взаємодії між компонентами, а end-to-end тестування продемонструвало працездатність повних користувацьких сценаріїв. Тестування продуктивності показало здатність системи ефективно обробляти навантаження, характерне для реальних умов експлуатації.

Демонстрація роботи проєкту показала високий рівень зручності використання інтерфейсу як для адміністраторів, так і для звичайних користувачів. Інтуїтивна навігація, швидка відповідь системи на дії користувачів та якісна валідація введених даних створюють позитивний досвід взаємодії з

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		63

системою. Адаптивний дизайн забезпечує однаково комфортне використання на комп'ютерах, планшетах та мобільних телефонах.

Практична цінність розробленого web-сервісу полягає у можливості його безпосереднього впровадження в аптечних закладах для автоматизації основних бізнес-процесів. Система дозволяє значно покращити ефективність управління асортиментом товарів, прискорити обслуговування клієнтів та забезпечити якісний онлайн-сервіс. Модульна архітектура та використання стандартних технологій полегшують подальшу підтримку та розвиток системи.

Технологічні рішення, застосовані при розробці, відповідають сучасним стандартам веб-розробки та можуть слугувати основою для створення більш складних інформаційних систем у сфері охорони здоров'я. Досвід інтеграції різних технологій, отриманий під час роботи над проектом, демонструє практичне застосування теоретичних знань для вирішення реальних завдань.

Перспективи подальшого розвитку включають інтеграцію з системами електронних рецептів, розширення аналітичних можливостей для прогнозування попиту на товари, впровадження системи лояльності клієнтів та інтеграцію з зовнішніми платіжними системами. Можливе також розширення функціональності для підтримки мережі аптек з централізованим управлінням та звітністю.

Результати виконаної роботи підтверджують актуальність обраної теми та демонструють успішне застосування сучасних інформаційних технологій для модернізації фармацевтичної галузі. Створений web-сервіс може служити основою для впровадження цифрових рішень в аптечних закладах та сприяти підвищенню якості фармацевтичного обслуговування населення України.

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		64

ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Закон України "Про лікарські засоби" від 04.04.1996 № 123/96-ВР (зі змінами від 2023 року). URL:<https://zakon.rada.gov.ua/laws/show/123/96-%D0%B2%D1%80> (дата звернення: 02.06.2025).

2. Закон України "Про наркотичні засоби, психотропні речовини і прекурсори" від 15.02.1995 №60/95-ВР (зі змінами від 2023 р.оку). URL:<https://zakon.rada.gov.ua/laws/show/60/95-%D0%B2%D1%80> (дата звернення: 03.06.2025).

3. Порядок ліцензування господарської діяльності з виробництва лікарських засобів, оптової та роздрібної торгівлі лікарськими засобами : Постанова Кабінету Міністрів України від 30.11.2016 № 929 (зі змінами від 2023 року). URL: <https://zakon.rada.gov.ua/laws/show/929-2016-%D0%BF> (дата звернення: 04.06.2025).

4. Державна фармакопея України / Державне підприємство "Український науковий фармакопейний центр якості лікарських засобів". 3-тє вид. доп. К. : Український науковий фармакопейний центр якості лікарських засобів, 2023. Т. 1. 1256 с.

5. Котвіцька А. А., Черних В. П. Фармацевтичний менеджмент та цифрові технології : підручник. Харків : НФаУ, 2023. 624 с.

6. Посилкіна О. В. Фармацевтичний маркетинг у цифрову епоху : підручник. Харків : НФаУ, 2023. 432 с.

7. React Documentation. Version 18.2. URL: <https://react.dev/> (дата звернення: 06.06.2025).

8. FastAPI Documentation. Version 0.104. URL: <https://fastapi.tiangolo.com/> (дата звернення: 07.06.2024).

9. SQL Alchemy Documentation. Version 2.0. URL:<https://docs.sqlalchemy.org/> (дата звернення: 08.06.2025).

10. Python Official Documentation. Version 3.12. URL:<https://docs.python.org/3/> (дата звернення: 09.06.2025).

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		65

11. Socket.IO Documentation. Version 4.7. URL: <https://socket.io/docs/v4/> (дата звернення: 13.06.2025).
12. Axios HTTP Client Documentation. Version 1.6. URL: <https://axios-http.com/docs/intro> (дата звернення: 01.06.2025).
13. Martin R. C. Clean Code: A Handbook of Agile Software Craftsmanship. 2nd Edition. Boston : Prentice Hall, 2023. 512 p.
14. Evans E. Domain-Driven Design: Tackling Complexity in the Heart of Software. Revised Edition. Boston : Addison-Wesley, 2023. 612 p.
15. ISO/IEC 25010:2023 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuARE) — System and software quality models. URL: <https://www.iso.org/standard/35733.html> (дата звернення: 04.06.2024).
16. ISO 27001:2022 Information technology – Security techniques – Information security management systems – Requirements. URL: <https://www.iso.org/standard/54534.html> (дата звернення: 05.06.2025).
17. Фармацевтична аналітика в Україні: інноваційні підходи та цифрові рішення / О. М. Євтушенко та ін. Фармацевтичний журнал. 2023. № 2. С. 24-32.
18. Цифрові технології в управлінні фармацевтичним бізнесом / В. В. Трохимчук, А. А. Котвіцька. Управління, економіка та забезпечення якості в фармації. 2023. № 1. С. 38-47.
19. Веб-технології в фармацевтичному бізнесі: сучасні тренди / М. В. Ковальчук, С. А. Петренко. Інформаційні технології та комп'ютерна інженерія. 2022. № 3. С. 67-75.
20. OWASP Web Application Security Testing Guide. Version 4.2. URL: <https://owasp.org/www-project-web-security-testing-guide/> (дата звернення: 08.06.2025).

					БР.КІ-42.00.00.000 ПЗ	Арк.
Змін.	Арк.	№ докум.	Підпис	Дата		66

БІБЛІОГРАФІЧНА ДОВІДКА

Тема бакалаврської роботи: *Розробка web-сервісу для обліку та продажу фармацевтичних продуктів засобами мови Python та фреймворку React*

Обсяг пояснювальної записки 67 аркушів:

6 таблиць;

19 рисунків;

— додаток.

Дата завершення роботи: *12 червня 2025р.*

Підпис студента- _____*Дячик В.В.*