

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»**

**Факультет інформатики та обчислювальної техніки**

**Кафедра інформаційних систем та технологій**

«На правах рукопису»  
УДК 004.8

До захисту допущено:  
Завідувач кафедри  
\_\_\_\_\_ Олександр РОЛІК  
«  » \_\_\_\_\_ 2024 р.

**Магістерська дисертація**

**на здобуття ступеня магістра**

**за освітньо-професійною програмою  
«Інтегровані інформаційні системи»**

**зі спеціальності 126 «Інформаційні системи та технології»**

**на тему: «Еквайрингова платформа з використанням штучного  
інтелекту»**

Виконав:

студент 2 курсу, групи ІА-31мп  
Гулящий Ігор Сергійович \_\_\_\_\_

Керівник:

доцент каф. ІСТ, к.т.н., доцент  
Писаренко Андрій Володимирович \_\_\_\_\_

Рецензент:

доцент каф. ІПІ, к.т.н., доцент  
Лісовиченко Олег Іванович \_\_\_\_\_

Засвідчую, що у цій магістерській  
дисертації немає запозичень з праць  
інших авторів без відповідних  
посилань.

Студент \_\_\_\_\_

Київ – 2024 року

**Національний технічний університет України**  
**«Київський політехнічний інститут імені Ігоря Сікорського»**  
**Факультет інформатики та обчислювальної техніки**  
**Кафедра інформаційних систем та технологій**

Рівень вищої освіти – другий (магістерський)

Спеціальність – 126 «Інформаційні системи та технології»

Освітньо-професійна програма «Інтегровані інформаційні системи»

ЗАТВЕРДЖУЮ

Завідувач кафедри

\_\_\_\_\_ Олександр РОЛІК

«\_\_» \_\_\_\_\_ 2024 р.

**ЗАВДАННЯ**  
**на магістерську дисертацію студенту**  
**Гулящому Ігорю Сергійовичу**

1. Тема дисертації «Еквайрингова платформа з використанням штучного інтелекту», науковий керівник дисертації Писаренко Андрій Володимирович, доц., затверджені наказом по університету від «08» 11 2024 р. № 5016-с
2. Термін подання студентом дисертації «09» 12 2024 р.
3. Об'єкт дослідження: процеси фінансового еквайрингу в електронних платіжних системах.
4. Вихідні дані: навантажена система здатна витримувати до 30 тисяч запитів на секунду, система без вразливостей, відповідність стандартам платіжних установ.
5. Перелік завдань, які потрібно розробити: провести аналіз існуючих рішень, спроектувати архітектуру системи, реалізувати бек-енд та фронт-енд, провести тестування, підготувати текстову та графічну частину пояснювальної записки
6. Орієнтовний перелік графічного (ілюстративного) матеріалу: структурна схема, діаграма варіантів використання, діаграма послідовності налаштування клієнтів, ER-діаграма, діаграма потоку даних, діаграма статусів платежу, блок-схема вибору типу інвойсу.
7. Дата видачі завдання 02.09.2024 р.

### Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1.	Аналіз предметної області	03.09.2024	
2.	Огляд існуючих рішень	07.09.2024	
3.	Вибір технологій та особливості реалізації	14.09.2024	
4.	Розроблення серверної частини	20.09.2024	
5.	Розроблення веб-інтерфейсів	20.10.2024	
6.	Тестування та виправлення помилок	30.10.2024	
7.	Оформлення пояснювальної записки	05.11.2024	
8.	Подання роботи на основний захист	16.12.2024	

Студент

Ігор ГУЛЯЦЬКИЙ

Науковий керівник

Андрій ПИСАРЕНКО

## РЕФЕРАТ

Еквайрингова платформа з використанням штучного інтелекту: 119 с., 100 табл., 22 рис., 9 дод., 24 джерела.

ЕКВАЙРИНГОВА ПЛАТФОРМА, ШТУЧНИЙ ІНТЕЛЕКТ, PCI DSS сертифікація, E-COMMERCE, ІНВОЙСИ, ПЛАТЕЖІ.

Актуальність теми полягає у забезпеченні безпеки зберігання інформації, за рахунок використання технологій штучного інтелекту. В умовах зростання кількості онлайн-платежів і використання цифрових фінансових сервісів загроза шахрайських дій значно збільшується. Забезпечення надійного виявлення фроду сприяє підвищенню довіри клієнтів до платіжних систем та створює конкурентні переваги для компаній.

Мета дослідження: підвищення ефективності процесів фінансового еквайрингу шляхом розроблення еквайрингової платформи, що використовує алгоритми штучного інтелекту для автоматизації, оптимізації транзакцій та забезпечення безпеки.

Для досягнення поставленої мети були сформульовані та вирішені наступні задачі:

- аналіз сучасного стану та основних проблем еквайрингу в електронних платіжних системах;
- дослідження можливості застосування алгоритмів штучного інтелекту для розв'язання задач еквайрингу;
- розроблення архітектури еквайрингової платформи з використанням методів штучного інтелекту;
- реалізація еквайрингової платформи з використанням штучного інтелекту;
- тестування еквайрингової платформи.

Об'єкт дослідження: процеси фінансового еквайрингу в електронних платіжних системах.

Предмет дослідження: методи штучного інтелекту для автоматизації, оптимізації транзакцій та забезпечення безпеки в еквайрингових платформах.

Система поєднує в собі сучасні моделі штучного інтелекту та можливість адаптації до конкретних еквайрингових сценаріїв.

Практична новизна полягає в поєднанні машинного навчання з поведінковим аналізом, що дозволяє мінімізувати хибні спрацьовування та підвищити точність виявлення шахрайства.

## ABSTRACT

Acquiring platform using artificial intelligence: 119 p., 100 tables, 22 figures, 9 appendices, 24 sources.

ACQUIRING PLATFORM, ARTIFICIAL INTELLIGENCE, PCI DSS certification, E-COMMERCE, INVOICES, PAYMENTS.

The relevance of the topic lies in ensuring the security of information storage through the use of artificial intelligence technologies. In the context of the growth of the number of online payments and the use of digital financial services, the threat of fraudulent actions increases significantly. Ensuring reliable fraud detection helps increase customer trust in payment systems and creates competitive advantages for companies.

Research objective: increasing the efficiency of financial acquiring processes by developing an acquiring platform that uses artificial intelligence algorithms to automate, optimize transactions and ensure security.

To achieve the goal, the following tasks were formulated and solved:

- analysis of the current state and main problems of acquiring in electronic payment systems;
- study of the possibility of using artificial intelligence algorithms to solve acquiring problems;
- development of the architecture of the acquiring platform using artificial intelligence methods;
- implementation of the acquiring platform using artificial intelligence;
- testing the acquiring platform.

Object of research: financial acquiring processes in electronic payment systems.

Subject of research: artificial intelligence methods for automation, optimization of transactions and ensuring security in acquiring platforms.

The system combines modern artificial intelligence models and the ability to adapt to specific acquiring scenarios.

The practical novelty lies in the combination of machine learning with behavioral analysis, which allows minimizing false positives and increasing the accuracy of fraud detection.

## ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ .....	11
ВСТУП.....	12
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	14
1.1 Аналіз предметної області .....	14
1.2 Аналіз існуючих рішень.....	14
1.2.1 Платформа LiqPay.....	14
1.2.2 Платформа WayForPay .....	16
1.2.3 Платформа UAPAY .....	17
1.3 Огляд різновидів платежів .....	18
1.3.1 Різновиди e-commerce платежів .....	18
1.4 Аналіз способів підтвердження платежів.....	19
1.5 Аналіз PCI DSS сертифікації .....	20
Висновки до розділу 1 .....	23
2 ВИБІР ТЕХНОЛОГІЙ ТА ОСОБЛИВОСТІ РЕАЛІЗАЦІЇ.....	24
2.1 Архітектура серверної частини .....	24
2.1.1 Шаблони проєктування .....	25
2.1.2 Вибір шини повідомлень.....	26
2.3 Бази даних.....	27
2.3.1 Вибір бази даних .....	27
2.3.2 Проєктування структури бази даних .....	28
2.4 Вимоги згідно PCI DSS сертифікації .....	35
2.4.1 Вимоги до способу зберігання чутливої інформації .....	36
2.4.2 Вимоги до адміністрування .....	36
2.4.3 Вимоги до інтеграції з еквайринговою платформою .....	37
2.5 Вибір технологій штучного інтелекту .....	38
2.6 Вибір мови програмування для серверної частини .....	38
2.7 Бібліотеки для платформи Node.js .....	39
2.7.1 Фреймворк веб-серверу.....	39



	9
2.7.2 Бібліотека ORM для роботи з базою даних.....	41
2.7.3 Авторизація та аутентифікація.....	43
2.7.4 Другорядні бібліотеки.....	45
2.8 Вибір технологій для інтерфейсів.....	46
Висновки до розділу 2.....	46
3 ОПИС СТРУКТУРИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	47
3.1 Функціональні вимоги до серверної частини.....	48
3.2 Функціональні вимоги до інтерфейсів.....	51
3.2.1 Функціональні вимоги до платіжної сторінки.....	51
3.2.2 Функціональні вимоги до адміністративної панелі.....	52
3.2.3 Функціональні вимоги до особистого кабінету клієнта.....	52
3.3 Структура мікросервісів серверної частини.....	53
3.3.1 Опис мікросервісу API-Gateway.....	53
3.3.2 Опис мікросервісу InvoiceManager.....	55
3.3.3 Опис мікросервісу AIModel.....	58
3.3.4 Опис мікросервісу ClientManager.....	60
3.3.5 Опис мікросервісу Reporter.....	61
3.3.6 Опис мікросервісу AdminManager.....	63
Висновки до розділу 3.....	64
4 ТЕСТУВАННЯ.....	65
4.1 Тестування веб-серверу.....	65
4.1.1 Тестування роботи з клієнтом.....	65
4.1.2 Тестування роботи з інвойсами.....	68
4.1.3 Тестування роботи адміністраторів.....	72
4.1.4 Тестування звітності.....	74
4.2 Тестування сторінки оплати.....	75
4.3 Тестування особистого кабінету користувача.....	77
Висновки до розділу 4.....	79
5 РОЗРОБЛЕННЯ СТАРТАП ПРОЄКТУ.....	80
5.1 Опис ідеї проєкту.....	80

	10
5.3 Технологічний аудит ідеї проєкту.....	83
5.4 Розроблення ринкової стратегії проєкту .....	95
Висновки до розділу 5 .....	106
ВИСНОВКИ .....	107
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	109
ДОДАТОК А .....	111
ДОДАТОК Б .....	112
ДОДАТОК В .....	113
ДОДАТОК Г .....	114
ДОДАТОК Д .....	115
ДОДАТОК Е .....	116
ДОДАТОК Ж .....	117
ДОДАТОК И .....	118
ДОДАТОК К .....	119

## ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ

БД – База даних

3DS – 3-D Secure – трьохстороння перевірка

API – Application Programming Interface – інтерфейс взаємодії з системою

CVV – Card Verification Value – Ключ верифікації кредитної карти

DI – Dependency Injection – Композиція залежностей

HTTP – HyperText Transfer Protocol – протокол передачі даних

JWT – JSON Web Token – JSON ключ

ORM – Object-relation Mapping – Об’єктно-реляційне відображення

P2P – Person to Person – платіж між двома фізичними особами

PCI DSS – Payment Card Industry Data Security Standard – Стандарти безпеки системи кредитних карт

RPC – Remote Procedure Call – Віддалений виклик

## ВСТУП

У сучасному бізнес-середовищі електронна комерція стала невід'ємною складовою діяльності компаній різного масштабу. З розвитком інформаційних систем і поширенням інтернету все більше підприємств переходять до онлайн-продажів, задовольняючи попит споживачів на зручні й швидкі платежі. У цьому контексті особливо важливими стають технології, що дозволяють забезпечити безпечний і ефективний процес обробки фінансових транзакцій. Онлайн-платформам потрібна сертифікація та проходження аудиту для проведення платежів, а також інтеграція з банками-еквайрами для виконання фінансових операцій.

Еквайрингові платформи забезпечують обробку транзакцій і інтеграцію з банками. Вони значно спрощують процес роботи з платежами, допомагаючи бізнесу різного масштабу отримувати оплату від клієнтів у зручний та безпечний спосіб. Окрім технічної інтеграції, еквайрингові платформи сприяють оптимізації бухгалтерської звітності, оскільки беруть на себе ведення фінансової документації, що полегшує процес оподаткування та контроль доходів.

Малий бізнес також отримує значні переваги від використання таких платформ. Підприємства, де працює лише одна або кілька осіб, можуть легко інтегрувати еквайрингові рішення в свою діяльність, відкривши особистий рахунок у банку та підписавши договір з еквайринговою платформою. Це забезпечує легалізацію доходів і спрощує фінансову звітність. Крім того, еквайрингові платформи дозволяють вибирати найвигідніші умови серед банків-еквайрів, залежно від обсягу транзакцій, кількості платежів або підтримки платіжних систем, таких як Google Pay і Apple Pay.

Одним із основних завдань сучасних еквайрингових платформ є забезпечення безпеки фінансових операцій. Технології штучного інтелекту відіграють усе більшу роль у цьому процесі, дозволяючи відслідковувати підозрілу активність, запобігати шахрайству та контролювати коректність транзакцій.

Штучний інтелект здатен працювати в режимі реального часу, забезпечуючи вищий рівень надійності й ефективності еквайрингових рішень.

Еквайрингова платформа допомагає підприємствам будь-якого масштабу безпечно й ефективно працювати в електронній комерції, що підвищує їхньою конкурентоспроможність та зручність для клієнтів.

Для досягнення поставленої мети були сформульовані та вирішені наступні задачі:

- аналіз сучасного стану та основних проблем еквайрингу в електронних платіжних системах;
- дослідження можливості застосування алгоритмів штучного інтелекту для розв'язання задач еквайрингу;
- розроблення архітектури еквайрингової платформи з використанням методів штучного інтелекту;
- реалізація еквайрингової платформи з використанням штучного інтелекту;
- тестування еквайрингової платформи.

Об'єкт дослідження: процеси фінансового еквайрингу в електронних платіжних системах.

Предмет дослідження: методи штучного інтелекту для автоматизації, оптимізації транзакцій та забезпечення безпеки в еквайрингових платформах.

Система поєднує в собі сучасні моделі штучного інтелекту та можливість адаптації до конкретних еквайрингових сценаріїв.

Практична новизна полягає в поєднанні машинного навчання з поведінковим аналізом, що дозволяє мінімізувати хибні спрацьовування та підвищити точність виявлення шахрайства.

## 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

### 1.1 Аналіз предметної області

Еквайрингова платформа є важливим компонентом сучасної фінансової інфраструктури, забезпечуючи обробку платежів для торговців і підприємств шляхом інтеграції з платіжними системами та банками. Еквайринг стосується процесу прийняття та обробки безготівкових платежів, що здійснюються за допомогою платіжних карт. Еквайрингова платформа виступає посередником між торговцями, які приймають оплату картами, та банками-еквайрами, які здійснюють обробку фінансових операцій.

Враховуючи, що кількість бізнесів зростає, а Україна все більше і більше цифровізується, з'являються нові схеми обходу системи оподаткування, незаконного збагачення, відмивання грошей, цифрових крадіжок. Еквайрингова платформа є невід'ємною частиною кожного бізнесу, який пов'язаний з продажами, тож ці схеми частково стосуються і цієї ніші. Розглянемо найпопулярніші еквайрингові платформи, задля аналізу їх системи безпеки. Також виконаний огляд функціоналу і можливостей цих систем, щоб зрозуміти переваги та недоліки в конкурентному середовищі.

### 1.2 Аналіз існуючих рішень

Оскільки темою дослідження є еквайрингові платформи, то доцільно буде оцінювати якість систем, які безпосередньо є конкурентами. Розглянуті три подібних сервіси, оскільки переважна більшість мають однаковий функціонал.

#### 1.2.1 Платформа LiqPay

LiqPay – це українська платіжна система, заснована ПриватБанком, яка дозволяє підприємствам і приватним особам приймати онлайн-платежі. Платформа підтримує широкий спектр платіжних інструментів, таких як банківські картки,

мобільні платежі, перекази через соціальні мережі та інші методи, що робить її зручною для різних типів користувачів. LiqPay є популярною серед малого та середнього бізнесу, завдяки своїй простоті інтеграції та доступності. На рисунку 1.1 зображено платіжну сторінку.

QR-код для оплати

Використовуйте Privat24

До оплати: 290,00 грн  
Hosting.UA - Рахунок #2577

Сплатити через Приват24

24 Pay

Google Pay | VISA \*\*\*\* 4442

Карта Інший спосіб

Номер картки  
\*\*\*\* \* \* \* \*

Термін дії CVV2  
ММ/РР \* \* ?

Натискаючи на кнопку «Оплатити», ви підтверджуєте, що ознайомлені з переліком інформації про послугу та приймаєте умови публічного договору

Сплатити

Скасувати оплату

Рисунок 1.1 – Платіжна сторінка LiqPay [1]

Можливості платформи:

- перекази з карти на рахунок;
- перекази з рахунку на карту;
- фрейм для токенизації карти. Це необхідно для кастомізації платіжної сторінки і повної інтеграції з LiqPay;
- оплата за допомогою мобільного додатку Privat24. Така можливість є лише у клієнтів банку.

Недоліки:

- неможливість вибрати кращі тарифи для користування, оскільки інтеграція відбувається виключно з банком-екваєром – «ПриватБанк»;

- особливість оплати за допомогою додатку Privat24, є перевагою та недоліком одночасно, оскільки не всі платники є клієнтами цього банку;
- немає можливості P2P переказів.

Також не було знайдено жодної згадки використання штучного інтелекту, або інших механізмів, для покращення конверсії чи то збільшення безпеки клієнтів.

### 1.2.2 Платформа WayForPay

WayForPay – українська еквайрингова платформа, яка не має жодного банку, як основного екваєру (рисунок 2). Послуги, які надаються:

- P2P перекази;
- створення інвойсу, рахунку на оплату;
- онлайн кредитування;
- антифрод API;
- переказ коштів з рахунку на рахунок.

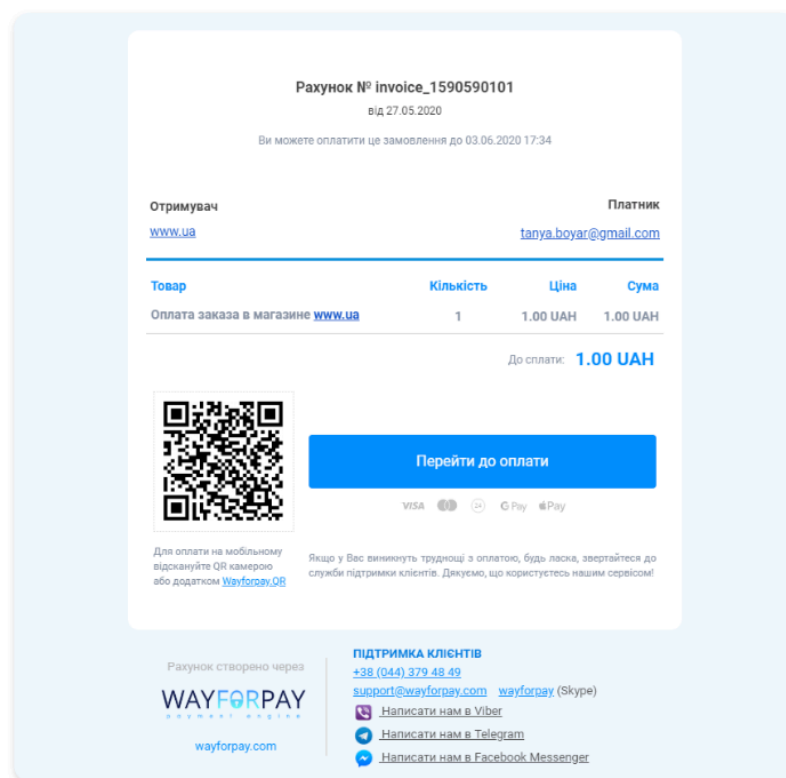


Рисунок 1.2 – Платіжна сторінка WayForPay [2]



В цій платформі є антифрод API, який надається клієнтам, але це означає, що він опціональний. Тобто кожен, хто не хоче витратити час та ресурси на повну інтеграцію – залишиться в потенційній небезпеці.

Недоліки:

- немає власної кастомізації платіжної сторінки. Тобто клієнти зобов'язані користуватись тією сторінкою, яку надає платформа;
- немає можливості оплати GooglePay чи ApplePay (рисунок 1.2).

У відкритій документації не було згадки використання штучного інтелекту.

Загалом WayForPay надає більше різних способів використання ніж LiqPay, але також має і недоліки.

### 1.2.3 Платформа UAPAY

UAPAY – українська еквайрингова платформа, яка є одним із партнерів OLX, але також має відкритий API для проведення транзакцій.

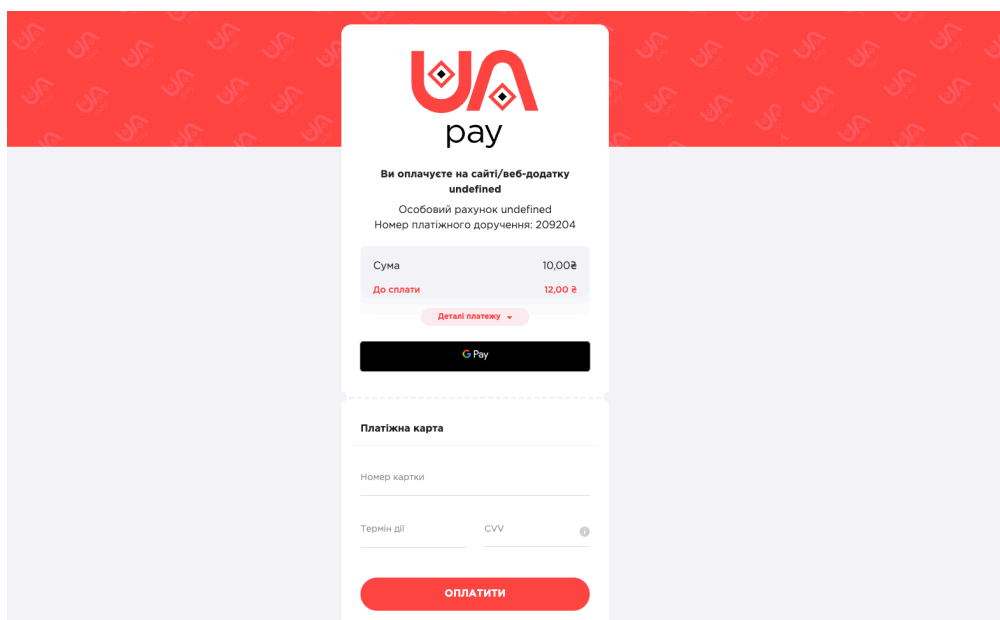


Рисунок 1.3 – Платіжна сторінка UAPAY [3]

Функціонал, який надає дана платформа:

- оплата за інвойсом, на платіжній сторінці (рисунок 1.3);

- картковий фрейм, для повної інтеграції;
- панель адміністрування мерчантом, де можна створити посилання на оплату на платіжну сторінку.

Недоліки:

- для повної інтеграції недоступна оплата GooglePay та ApplePay;
- відсутність будь-яких інструментів для забезпечення безпеки, чи протидії фроду.

### 1.3 Огляд різновидів платежів

Платежі, які проводяться в еквайринговій платформі зазвичай поділяються на два види:

- P2P – переказ з карти на карту. Тобто переказ грошей від фізичної особи до фізичної особи;
- E-commerce – перекази з картки на рахунок. Тобто оплата з карти на юридичний рахунок, ФОП рахунок або будь-який рахунок IBAN.

Враховуючи, що система має слідкувати за шахрайством, а заробіток в обхід оподаткуванню – є шахрайством, платформа надає можливість проводити лише e-commerce платежі.

#### 1.3.1 Різновиди e-commerce платежів

E-commerce платежі також поділяються на декілька типів[4]:

- авторизаційний платіж. Платіж який знімає одразу вказану суму з карти платника на користь отримувача;
- преавторизаційний платіж. Це платіж, який відбувається в два етапи. Перший, коли кошти бронюються на рахунку платника, очікуючи другий етап. Та другий етап – коли підтверджується списання коштів, але із вказанням суми. В свою чергу, сума може бути менша або більша згідно договору;

– рекурентний платіж. Платіж-підписка, який відбувається з підтвердженням зі сторони платника лише одноразово, і в залежності від умов повторно списує гроші коли підходить термін списання.

Для гнучкості в наданні послуг розроблена еквайрингова платформа підтримує усі види e-commerce платежів.

#### 1.4 Аналіз способів підтвердження платежів

Важливу роль відіграє безпека зі сторони платника, щоб унеможливити списання коштів з фізичної карти, або у випадку зламу, платіжних даних.

Для цього існують різні типи підтвердження платежу платником. Розглянемо основні типи підтвердження платежу: 3DS та LOOKUP.

LOOKUP більш простий в реалізації. Його суть полягає в тимчасовому блокуванні певної суми коштів, які повертаються після підтвердження за допомогою коду з СМС повідомлення. Повна послідовність зображена на рисунку 1.4.

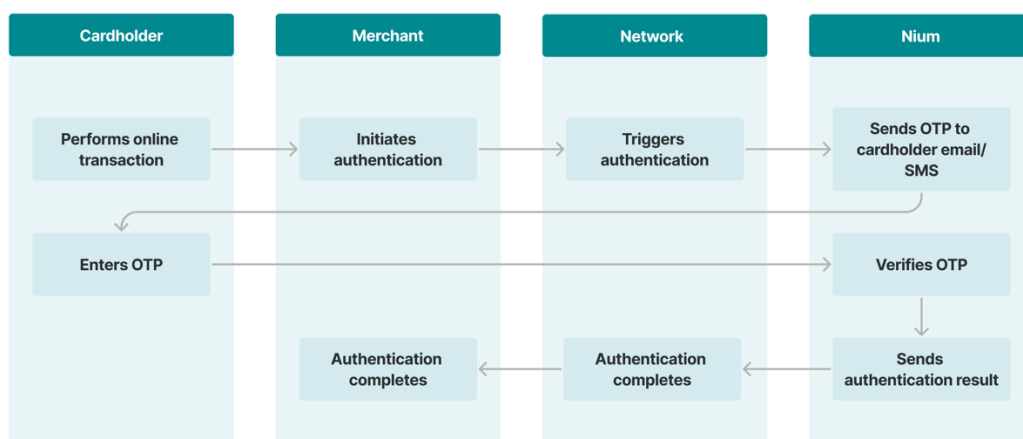


Рисунок 1.4 – Послідовність підтвердження платежу за допомогою LOOKUP [5]

Перехопити СМС повідомлення в час розвитку технологій не так складно, а отже запроваджений новий тип підтвердження платежу – 3DS. Послідовність зображено на рисунку 1.5.

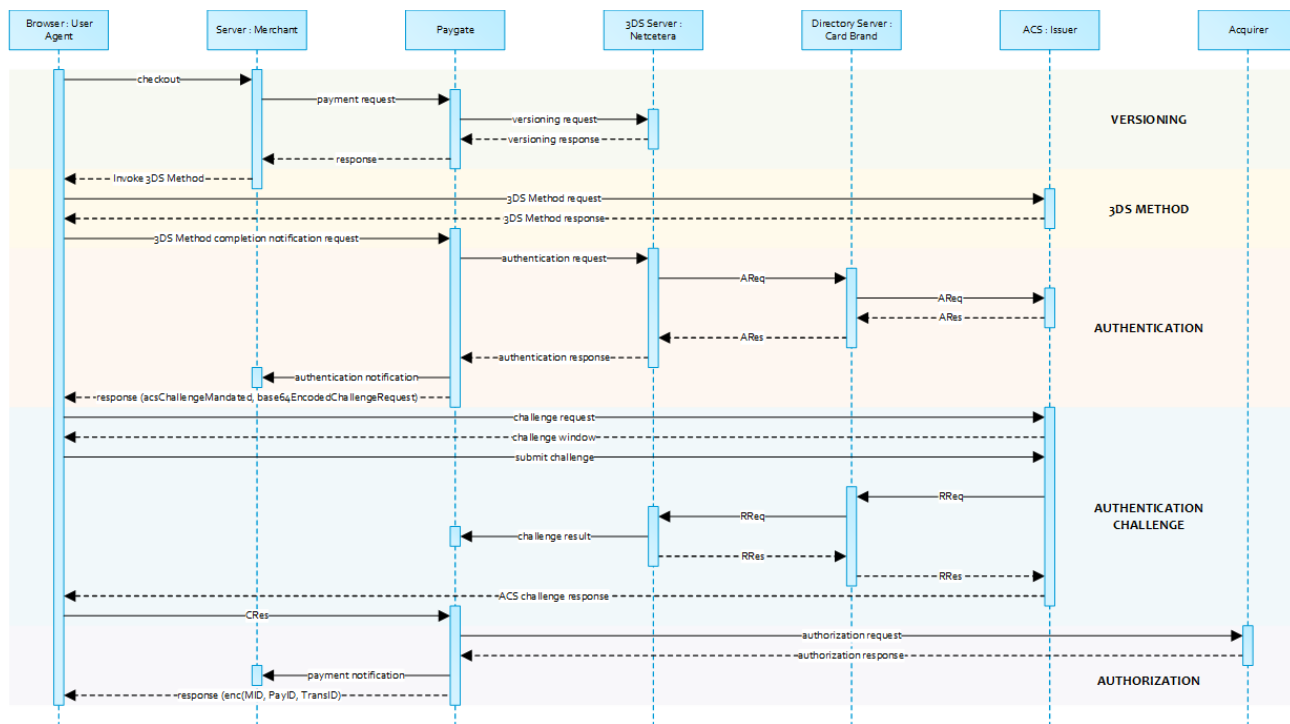


Рисунок 1.5 – Послідовність проходження 3DS перевірки [6]

Основна особливість такого підходу – залученість усіх трьох сторін: банк-екваєр, банк-емітент та ініціатор платежу. Старі версії цього протоколу також відправляють СМС, але починаючи з другої версії підтвердження відбувається в додатку банку-емітенту за його наявності в платника [6].

### 1.5 Аналіз PCI DSS сертифікації

PCI DSS – це набір вимог і рекомендацій, розроблених для забезпечення безпеки даних платіжних карт. Стандарт впроваджений Радою зі стандартів безпеки індустрії платіжних карт до якої входять провідні платіжні системи, такі як Visa, MasterCard, American Express, Discover та JCB [7]. Основною метою PCI DSS є захист конфіденційної інформації платіжних карток від несанкціонованого доступу, шахрайства та витоків даних. Ліцензія зображена на рисунку 1.6.



Рисунок 1.6 – PCI DSS ліцензія [7]

Важливою особливістю стандарту є його гнучкість у застосуванні, оскільки він дозволяє адаптувати вимоги до конкретних умов бізнесу. Однак недотримання стандарту може призвести до значних штрафних санкцій, втрати довіри з боку клієнтів і платіжних систем, а також підвищених ризиків витоку даних.

У контексті еквайрингових платформ, відповідність PCI DSS є основною вимогою для забезпечення безпеки фінансових транзакцій та захисту даних користувачів. Використання штучного інтелекту в таких системах може сприяти підвищенню рівня безпеки через автоматичне виявлення аномалій та потенційних загроз у режимі реального часу, що полегшує дотримання стандартів безпеки.

Стандарт містить 12 основних вимог, які охоплюють різні аспекти безпеки інформаційних систем і процесів обробки платіжних карт. Ці вимоги поділяються на шість основних категорій і охоплюють різні аспекти захисту інформації, яка обробляється, зберігається або передається під час використання платіжних карт.

Перша категорія – забезпечення безпечної мережі. Вона включає в себе:

- підтримку та встановлення безпечної мережевої інфраструктури, використовуючи брандмауери;

- використання належної конфігурації безпеки для всіх систем і виключення можливості використовувати стандартні паролі та інші налаштування, надані постачальником.

Друга категорія – захист даних власників платіжних карт. Вона включає:

- захист збережених даних власників карт;
- шифрування даних карт при передачі по відкритих мережах.

Третя категорія – підтримка системи управління вразливостями:

- зобов'язання використовувати й регулярно оновлювати антивірусне програмне забезпечення або інші механізми захисту, встановлення та виправлення знайдених вразливостей;

- всі програми та системи мають оновлюватись постійно до найновіших версій.

Четверта категорія – контроль доступу:

- обмеження доступу до даних карт;
- призначення і використання унікальних ідентифікаційних даних кожному користувачеві, що має доступ до системи;

- обмеження фізичного доступу до даних карт, захист фізичних серверів.

- П'ята категорія – моніторинг і тестування мережі:

- відстеження та контроль всіх доступів до мережевих ресурсів і даних карт;
- регулярне тестування системи безпеки та процесів.

Шоста категорія – підтримка політики безпеки інформації. Зобов'язує підтримувати політику безпеки для всього персоналу.

Ці вимоги включають контроль доступу до даних, шифрування інформації, моніторинг мережевої активності, регулярне тестування системи безпеки та забезпечення належного захисту від шкідливого програмного забезпечення. PCI DSS призначений для всіх організацій, які зберігають, передають або обробляють дані платіжних карт, незалежно від їх розміру чи обсягу операцій.

## Висновки до розділу 1

В цьому розділі розглянуті три еквайрингові платформи. Кожна має переваги над іншою, та покриває потреби кінцевого користувача, але жодна з систем не впровадила використання штучного інтелекту для вирішення задач безпеки чи покращення конверсії платформи. Тому поставлена задача є актуальною. Також розглянуті вимоги для підтримки стандартів PCI DSS, оскільки еквайрингова платформа може буде запущена тільки в випадку її ліцензювання.

## 2 ВИБІР ТЕХНОЛОГІЙ ТА ОСОБЛИВОСТІ РЕАЛІЗАЦІЇ

### 2.1 Архітектура серверної частини

Еквайрингові платформи мають велике навантаження. Основною вимогою до таких систем є стабільність та відмовостійкість, оскільки це опрацювання мільйонів платежів, тисячі генерацій бухгалтерських звітів та десятки інтеграцій. Тож прийняте рішення використовувати мікросервісну архітектуру.

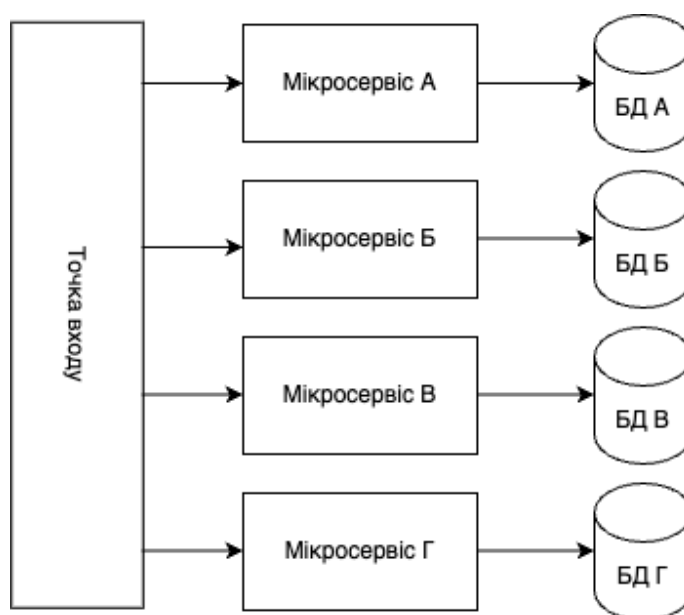


Рисунок 2.1 – Мікросервісна архітектура

Перевагами такого підходу є:

- легкість масштабування. В свою чергу масштабування є горизонтальне та вертикальне. Горизонтальне – це збільшення кількості екземплярів певного мікросервісу. Вертикальне – це збільшення ресурсів віртуальної чи фізичної машини;
- відмовостійкість. Кожен мікросервіс має свою базу даних, тож при непередбачуваних обставинах лише частина функціоналу усієї системи не буде працювати, допоки не відбудеться відновлення;
- розподілене розроблення. В такій великій системі повинна бути присутня підтримка усієї екосистеми, а це і декілька веб-проектів, і підтримка різних



інтеграцій, і підтримка бекенду, який в свою чергу має обслуговувати усі ці кінцеві продукти, і підтримка мобільної інтеграції. Тож, розділивши усе це на міні-проекти та бекенд на мікросервіси, буде не складно розділити відповідальність між різними командами, які мають компетенцію в конкретній галузі.

### 2.1.1 Шаблони проєктування

Оскільки обрана мікросервісна архітектура, для неї потрібні шаблони проєктування системи, щоб переваги, були підсилені реалізацією.

Шаблон проєктування CQRS вирішує проблему розділення запитів на запис і на читання, що дозволяє керувати навантаженням на систему. Ідея в тому, щоб розділити механізми транспорту запитів між мікросервісами на два види:

- асинхронні запити для запису;
- синхронні запити для читання.

Асинхронні запити дозволяють розвантажити систему, оскільки початковий користувач не очікуватиме довго відповідь від навантаженого запиту. Наприклад, платник оплачує інвойс. В цьому кейсі відбуваються операції запису на диск, щоб зберегти інформацію в БД, також відбувається спілкування з банком-екваєром. І весь цей час користувач очікує відповіді з лоадером на веб-сторінці. Асинхронні запити дозволяють реалізувати «оптимістичне створення». Платіж по інвойсу буде створений в будь-якому випадку, але не завжди буде успішним. Тож нічого не забороняє оптимістично створити платіж, і не затримувати кінцевого користувача. Такий підхід реалізує патерн «Команда» та шина повідомлень

Реалізація синхронних запитів в мікросервісній архітектурі може бути з використанням:

- HTTP-запитів між мікросервісами;
- RPC-запитів за допомогою шини повідомлень.

Переваги HTTP-рішення – швидка реалізація. Потрібно лише створювати веб-сервер для кожного мікросервіса.

Недоліки HTTP-рішення – відсутність «кешу» запитів.

Головна перевага RPC за допомогою шини повідомлень – це сама шина. Вона дозволяє «заморожувати» запити між мікросервісами, якщо кінцевий мікросервіс не відповідає. Це підсилює цілісність системи, оскільки непрацюючий сервіс обробить запити, які йому надсилались, після усунення проблем в роботі.

Доменна організація структури (далі DDD) обрана як основний підхід до написання мікросервісу. DDD реалізує конкретну модель обраної предметної області, тобто кожна сутність, відповідає тільки за зміну свого стану і процеси, які пов'язані з нею.

### 2.1.2 Вибір шини повідомлень

При виборі шини повідомлень розглянуті наступні варіанти:

- RabbitMQ;
- Kafka;
- AWS SQS.

Переваги RabbitMQ:

- підтримка протоколів TOMP, MQTT, AMQP;
- більше можливостей в маршрутизації повідомлень в черги, та їх перенаправлення;
- можливість реалізації RPC протоколу.

Переваги Kafka:

- доступний API роботи з потоками. Але синхронізація баз даних вирішується на інфраструктурному рівні;
- паралельна обробка черг.

Переваги AWS SQS:

- проста реалізація оформлення черг;
- проста документація;
- підтримка екосистеми AWS.

Враховуючи вимоги до використання шини повідомлень, RabbitMQ покриває усі потреби в використанні шини повідомлень, а саме: реалізація протоколу RPC, та реалізація патерну «Команда».

## 2.3 Базы даних

### 2.3.1 Вибір бази даних

Під час вибору БД розглянуті два варіанти: PostgreSQL та MySQL. Для порівняння розглянуті наступні фактори:

- продуктивність читання. PostgreSQL зазвичай швидше обробляє складні запити завдяки розширеним можливостям індексування та підтримці Index-Only Scans. MySQL ефективніше працює із простими завданнями читання, але поступається PostgreSQL у випадку складних запитів;

- продуктивність запису. PostgreSQL значно перевершує MySQL у навантаженнях, орієнтованих на записи. Його механізм Write-Ahead Logging забезпечує кращу багатозадачність і швидше відновлення після збоїв. У тестах PostgreSQL був приблизно в 3,5 рази швидше за MySQL у завданнях запису [8];

- загальна затримка. PostgreSQL демонструє у 2,3 раза кращу продуктивність у середньому по набору поширених операцій, особливо перевершуючи MySQL у завданнях групування та сканування таблиць;

- конкурентність та масштабованість. PostgreSQL використовує багатоверсійну контрольну систему конкурентності, що дозволяє виконувати транзакції без блокування читання, забезпечуючи високу пропускну здатність для великої кількості одночасних запитів. MySQL також підтримує MVCC, але гірше справляється з високим навантаженням. PostgreSQL має паралельне виконання запитів, розділення таблиць на партиції та кращі можливості горизонтального масштабування, що робить його ідеальним для великих і складних систем [9];

- типи даних. PostgreSQL підтримує більшу кількість типів даних, зокрема JSONB для роботи з напівструктурованими даними, а також геометричні та мережеві типи. Це робить його більш універсальним для сучасних додатків;

– індексація. PostgreSQL підтримує розширені функції, як-от часткові та функціональні індекси, які суттєво покращують продуктивність у специфічних сценаріях. MySQL таких функцій не має;

– надійність та відповідність стандартам. PostgreSQL суворо дотримується принципів ACID і стандартів SQL, забезпечуючи високу цілісність даних. Хоча MySQL також підтримує ACID, його обробка транзакцій менш надійна у складних сценаріях [10].

PostgreSQL поєднує масштабованість, продуктивність і розширені функції, що робить його для проєктів з високими вимогами до обробки даних.

### 2.3.2 Проєктування структури бази даних

Проєктування структури БД є настільки ж важливим, як і вибір технологій чи вибір архітектури. Гарно спроектована БД завжди дається в знаки – легко підтримувати, легко розширювати. Проєктування БД для еквайрингової платформи складається з:

- проєктування основної частини;
- проєктування частини адміністрування;
- проєктування механізмів зберігання чутливої інформації про карти.

Структура БД наведено в додатку Г.

Детальний опис таблиць наведено в таблицях 2.1–2.15.

Таблиця 2.1 – Таблиця карт платників

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор

Обмеження	Назва поля	Тип даних	Опис поля
Вторинний ключ	panId	char(36)	Посилання на ідентифікатор пану картки
Вторинний ключ	keyId	char(36)	Посилання на ідентифікатор ключа яким було зашифровано чутливу інформацію
	encrypted	varchar	Зашифрована чутлива інформація
	hashExpiresAt	varchar	Хешована дата життя карти
	createdAt	timestamp	Дата створення карти в системі

Таблиця 2.2 – Таблиця номерів карт платників

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
	hashPan	varchar	Захешований пан карти платника

Таблиця 2.3 – Таблиця ключів шифрування

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
	key	varchar	Зашифрований ключ шифрування

Таблиця 2.4 – Таблиця клієнтів

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
	edrpou	varchar	ЄДПОУ ТОВа, або ПІН ФОПа
	secretKey	varchar	Приватний ключ клієнта
	companyName	varchar	Назва організації
	createdAt	timestamp	Дата створення
	updatedAt	timestamp	Дата оновлення

Таблиця 2.5 – Таблиця інвойсів

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
	receiptId	char (36)	Ідентифікатор квитанції
	type	enum	Тип інвойсу
	status	enum	Статус інвойсу
	createdAt	timestamp	Дата створення
	updatedAt	timestamp	Дата оновлення

Таблиця 2.6 – Таблиця платежів

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
Вторинний ключ	clientId	char(36)	Посилання на клієнта
Вторинний ключ	terminalId	char(36)	Посилання на термінал

Обмеження	Назва поля	Тип даних	Опис поля
Вторинний ключ	invoiceId	char(36)	Посилання на інвойс
Вторинний ключ	cardId	char(36)	Посилання на карту платника
	amount	integer	Тіло платежу
	commission	integer	Комісія внутрішня
	inFeeCommission	integer	Комісія зовнішня
	status	enum	Статус платежу
	createdAt	timestamp	Дата створення
	updatedAt	timestamp	Дата оновлення

Таблиця 2.7 – Таблиця банків-екваєрів

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
	name	Varchar	Назва банку
	status	enum	Статус
	createdAt	timestamp	Дата створення

Таблиця 2.8 – Таблиця реквізитів клієнта

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
Вторинний ключ	clientId	char(36)	Посилання на клієнта
	commissionRules	json	Правила розрахунку комісій

Обмеження	Назва поля	Тип даних	Опис поля
	inFeeCommission	json	Правила розрахунку комісій клієнта

Таблиця 2.9 – Таблиця терміналів

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
Вторинний ключ	acquiringBankId	char(36)	Посилання на банк-екваєр
	bankCommissionPercent	float	Відсоток комісії банку
	acceptance	boolean	Підтримка безакцептності
	googlePayAllowed	boolean	Підтримка оплати Google Pay
	applePayAllowed	boolean	Підтримка оплати Apple Pay

Таблиця 2.10 – Таблиця маршрутизації транзакцій клієнта

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
Вторинний ключ	clientId	char(36)	Посилання на клієнта
Вторинний ключ	terminalId	char(36)	Посилання на термінал



Обмеження	Назва поля	Тип даних	Опис поля
	rule	Json	Правило маршрутизації

Таблиця 2.11 – Таблиця зв'язку платежів з реквізитами

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
Вторинний клбч	paymentServiceId	char(36)	Посилання на реквізити
Вторинний ключ	paymentId	char(36)	Посилання на платіж
	amount	integer	Винагорода
	commission	integer	Комісія внутрішня
	inFeeCommission	integer	Комісія зовнішня
	createdAt	timestamp	Дата створення
	updatedAt	timestamp	Дата оновлення

Таблиця 2.12 – Таблиця адміністраторів

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
Вторинний ключ	positionId	char(36)	Посилання на позицію адміністратора
	name	varchar	Ім'я адміністратора
	lname	varchar	Прізвище адміністратора
	email	varchar	Робоча пошта

Обмеження	Назва поля	Тип даних	Опис поля
	password	varchar	Захешований пароль
	lastLoggedAt	timestamp	Дата останньої авторизації
	createdAt	timestamp	Дата створення

Таблиця 2.13 – Таблиця позицій

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
	name	varchar	Назва позиції

Таблиця 2.14 – Таблиця кодів підтверджень

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
Вторинний ключ	adminId	char(36)	Посилання на адміністратора
Обмеження	Назва поля	Тип даних	Опис поля
	code	varchar	Код підтвердження
	createdAt	timestamp	Дата створення

Таблиця 2.15 – Таблиця логів адміністраторів

Обмеження	Назва поля	Тип даних	Опис поля
Первинний ключ	id	char(36)	Ідентифікатор
	adminId	char(36)	Посилання на адміністраторів
	actionName	varchar	Дія

Обмеження	Назва поля	Тип даних	Опис поля
	createdAt	timestamp	Дата дії

Зв'язки між таблицями БД наведено в таблиці 2.16

Таблиця 2.16 – Зв'язки між таблицями БД

Назва таблиці	Зв'язок	Назва таблиці
Таблиця панів карт	1:M	Таблиця карт платника
Таблиця ключ шифрування	1:M	Таблиця карт платника
Таблиця карт платника	1:M	Таблиця платежів
Таблиця інвойсів	1:M	Таблиця платежів
Таблиця клієнтів	1:M	Таблиця платежів
Таблиця платежів	1:M	Таблиця зв'язку платежа та реквізитів
Таблиця платежів	M:N	Таблиця реквізитів
Таблиця платежів	1:1	Таблиця терміналів
Таблиця зв'язку платежа та реквізитів	1:1	Таблиця терміналів
Таблиця терміналів	1:1	Таблиця маршрутизації
Таблиця терміналів	1:M	Таблиця банків-еквасрів
Таблиця реквізитів	M:1	Таблиця клієнтів
Таблиця адміністраторів	1:1	Таблиця позицій
Таблиця адміністраторів	1:1	Таблиця логів

## 2.4 Вимоги згідно PCI DSS сертифікації

Вище розглянуті вимоги до PCI DSS сертифікації, згідно якої аудит відбувається для кожної еквайрингової платформи без виключення. Ці вимоги дуже загальні, тож потрібно висвітлити підходи, які використані в ході розроблення програмного забезпечення, щоб відповідати даним вимогам.

### 2.4.1 Вимоги до способу зберігання чутливої інформації

Щоб відповідати вимозі зберігання чутливої інформації, поставлені наступні задачі:

- в БД ніколи не має зберігатись CVV карти платника. Код підтвердження карти потрібен лише на момент оплати для передачі банкам-екваєрам, а отже, не зберігаючи CVV, знімається відповідальність з системи, у разі інцидентів з витоком даних;

- в БД ніколи не має зберігатись інформація про номер карти та дата придатності в відкритому вигляді, а отже тільки в зашифрованому. Такий підхід дозволить вберегти чутливу інформацію, навіть у випадку витоку і злому БД;

- пошук карти платника може відбуватись за допомогою контрольної суми, тобто хешу. З таким підходом, не потрібно кожен раз розшифровувати чутливу інформацію, щоб дізнатися чи існує така карта;

- побудувати процес зміни ключа шифрування. Проектування БД, яке розглянуто раніше, дозволяє використовувати різні ключі шифрування. Що в свою чергу дозволить безболісно оновлювати і перешифровувати дані, не вимикаючи сервіс, оскільки чутлива інформація, яка ще не була перешифрована, може використати старий ключ шифрування для передачі інформації;

- передача інформації про карту платника, при спілкуванні з банками-екваєрами має відбуватись тільки зашифрованому вигляді.

Дотримуючись усіх цих підходів, система буде відповідати вимозі зберігання і шифрування чутливої інформації.

### 2.4.2 Вимоги до адміністрування

Четверта, п'ята та шоста вимоги – про безпеку, контроль доступу, моніторинг системи.

Щоб дотримуватися цих вимог, поставлені наступні задачі:

- чітка сітка ролей і їх доступів до можливостей адміністрування. Спроектowana БД для адміністрування включає можливість розділення адміністраторів на ролі, і відповідно на можливості в адміністративній панелі;
- авторизація має відбуватися кожні 15хв;
- авторизація має відбуватися за допомогою двофакторного підтвердження;
- логування і контроль дій адміністраторів. Спроектowana БД записує усі дії адміністраторів;
- адміністратори не мають прав на перегляд чутливої інформації. Оскільки уся інформація зашифрована, адміністрація може користуватись автоматичними механізмами і ідентифікаторами, для завершення дій пов'язаних з платежами: зарахування, повернення тощо.

#### 2.4.3 Вимоги до інтеграції з еквайринговою платформою

Еквайрингова платформа надає гнучкі можливості для клієнтів в способі використання:

- готова платіжна сторінка;
- клієнтська панель адміністрування, де можна створити посилання на готову платіжну сторінку;
- повна інтеграція, задля кастомізації своєї сторінки оплати.

Останній пункт дуже чутливий, оскільки клієнт, який не матиме PCI DSS ліцензії, зможе реалізувати власні рішення. За таких обставин дуже важливо дотримуватися вимоги зберігання чутливої інформації.

Отже, щоб відповідати цим вимогам, треба:

- ізолювати інтеграцію клієнта від створення карткових даних в нашій системі. Цей пункт легко вирує реалізація фрейму для створення карти платника в системі;
- не мати в відкритому API точок входу, які віддають чутливу інформацію;

– для взаємодії з системою використовувати асиметричне шифрування для підписів усіх запитів і відповідей.

## 2.5 Вибір технологій штучного інтелекту

Інтеграція штучного інтелекту в систему керування платежами є основним інструментом для забезпечення високої точності та ефективності у виявленні підозрілих транзакцій і прийнятті рішень щодо їх подальшої обробки.

Python є одним з найкращих варіантів для задач виявлення фроду завдяки таким факторам:

- розвинена екосистема бібліотек. Бібліотеки для машинного та глибинного навчання PyTorch, TensorFlow [9] спрощують процес створення, навчання та тестування моделей;
- інтеграція з БД та API. Python забезпечує підключення до БД, що полегшує збір і обробку транзакційних даних.

Зважаючи на вимоги до точності та швидкості, глибинні нейронні мережі на базі TensorFlow є оптимальним рішенням. Цей підхід дозволяє:

- виявляти складні, приховані шаблони в транзакціях;
- забезпечувати масштабованість у випадку збільшення обсягів даних;
- інтегрувати рішення в еквайрингову систему.

Такий набір технологій дозволяє вирішувати завдання визначення безпеки транзакцій і прийняття рішень без додаткової перевірки, забезпечуючи високу точність і швидкість обробки.

## 2.6 Вибір мови програмування для серверної частини

JavaScript та TypeScript є популярними мовами програмування, які використовуються для створення серверної частини веб-додатків на платформі Node.js. Обидві мови мають низку переваг, які роблять їх хорошим вибором для еквайрингової платформи. Основні переваги:

- швидкість розробки, яка дозволяє швидко створювати прототипи та тестувати ідеї;
- асинхронність, найшвидша серед усіх мов програмування. JavaScript має лише один потік, на відміну від Java чи C#, але проблеми навантаженості вирішуються горизонтальним масштабуванням на рівні інфраструктури, що нівелює перевагу раніше згаданих конкурентів.
- гнучкість, оскільки має дуже великий спектр бібліотек, що дає свободу у виборі підходу до розроблення.

TypeScript є надбудовою над JavaScript, яка додає статичну типізацію та покращує зручність і надійність розроблення. Явна типізація дозволяє різним командам, швидко орієнтуватись в кодовій базі, задля вирішення питань.

Тож, враховуючи усі ці фактори, обрана мова TypeScript як основна мова програмування серверної частини.

## 2.7 Бібліотеки для платформи Node.js

Node.js [10] надає великі можливості як у виборі бібліотек, так і виборі пакетних менеджерів. Надійним та протестованим часом є – node package manager, або npm [11]. Але, оскільки кожен може викласти свою бібліотеку в відкритий доступ, слід вибирати дуже ретельно, щоб це ніяк не повпливало на роботу системи, чи її злому, через застарілість чи відсутність підтримки бібліотеки.

### 2.7.1 Фреймворк веб-серверу

Обрані три найпопулярніші фреймворки для розроблення веб-серверів: Koa.js, Fastify та Express.js.

Бенчмарк-порівняння пропускної спроможності кожного з них зображено на рисунку 2.2.

Framework	Version	Router?	Requests/sec
Express	4.17.3	✓	14,200
hapi	20.2.1	✓	42,284
Restify	8.6.1	✓	50,363
Koa	2.13.0	✗	54,272
<b>Fastify</b>	<b>4.0.0</b>	<b>✓</b>	<b>77,193</b>
-			
http.Server	16.14.2	✗	74,513

Рисунок 2.2 – Бенчмарк пропускної спроможності фреймворків [15]

Отже Fastify виглядає найкращим з кандидатів. Розглянуті і порівняні інші критерії, які наведені в таблиці 2.17.

Таблиця 2.17 – Порівняльна характеристика фреймворків

Назва критерію	Fastify	Koa.js	Express.js
Продуктивність	90,000–100,000 запитів в секунду	70,000 запитів в секунду	50,000–60,000 запитів в секунду
Підтримку модульності та розширюваності	Має вбудовану підтримку плагінів	Доведеться самостійно інтегрувати модулі	Складна інтеграція модулів та плагінів



Назва критерію	Fastify	Koa.js	Express.js
Типізація TypeScript	Найкраща підтримка TypeScript	Підтримує TypeScript	Підтримується, але через старішу архітектуру не завжди зрозуміла
Вбудована валідація	Наявна	Не має	Не має
Спільнота та документація	Спільнота молода, проєкт також, але активно зростає. Документація чітка	Має невелику, але активну спільноту. Документація більш технічна, потребує часу на вивчення	Найбільша спільнота та найбільша кількість ресурсів, але документація може бути застарілою для нових вимог

Fastify є найбільш технічно досконалим рішенням, що поєднує високу продуктивність, зручність у розробці та відповідність сучасним стандартам. Це робить його кращим вибором для серверної частини еквайрингової платформи.

### 2.7.2 Бібліотека ORM для роботи з базою даних

Об'єктно-реляційні відображення – це додаткова абстракція, яка дозволяє працювати з таблицями БД, як з об'єктами ООП мови програмування. Розглянуті наступні бібліотеки: Drizzle, TypeORM, Prisma. Порівняння наведено в таблиці 2.18.

Таблиця 2.18 – Порівняння бібліотек ORM

Критерій	Drizzle	TypeORM	Prisma
Продуктивність	Забезпечує високу продуктивність завдяки генерації SQL-запитів під час компіляції TypeScript. Це дозволяє уникнути зайвого оверхеду під час виконання, який часто є в інших ORM	Відомий як важчий ORM. Генерація SQL-запитів може бути повільнішою, особливо для складних запитів	Продуктивність добра, але через генерацію запитів і специфічну інфраструктуру (Prisma Client і Prisma Engine) може бути трохи повільнішим порівняно з Drizzle у сценаріях високого навантаження.
Типізація та інтеграція з TypeScript	Забезпечує статичну типізацію на основі схеми бази даних. Наявне автодоповнення і статична перевірка типів, які допомагають уникнути помилок під час розробки	Має підтримку TypeScript, але типізація менш надійна і менш автоматизована. Вимагає ручної роботи для синхронізації моделі і схеми бази	Має дуже сильну підтримку типів і автоматично генерує типи на основі схеми Prisma

Критерій	Drizzle	TypeORM	Prisma
Гнучкість	Не накладає строгих обмежень на структуру проекту, дозволяючи створювати запити SQL на основі бізнес-логіки. Це добре підходить для кастомних та високопродуктивних додатків	Більш "монолітний". Підтримує декоратори, які спрощують визначення моделей, але ускладнюють кастомізацію	Підходить для стандартних CRUD-операцій, але може бути менш гнучким для складних кастомних запитів через свій декларативний підхід
Підтримка міграцій	Міграції генеруються на основі змін у схемі. Система проста і дозволяє легко керувати версіями бази даних	Має вбудовану систему міграцій, але її використання може бути складним і часом ненадійним	Має добре продуману систему міграцій, але її складність зростає при роботі зі складними базами даних

Порівнявши готові рішення, обрана бібліотека Drizzle, як основна бібліотека ORM, через її високу продуктивність, надійність типізації, гнучкості для складних запитів та реалізації простих міграцій.

### 2.7.3 Авторизація та аутентифікація

Авторизація та аутентифікація і підхід до їх реалізації також дуже важливий з точки зору безпеки еквайрингової платформи. Авторизація присутня в двох місцях системи, а аутентифікація в одному.

Розглянуті наступні варіанти авторизації: JWT та сесії.

Принцип роботи сесії зображений на рисунку 2.3.

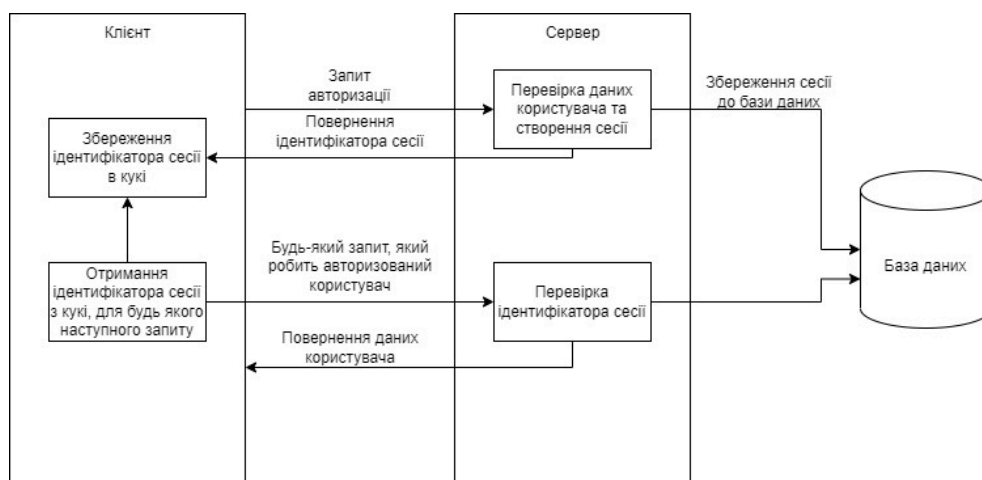


Рисунок 2.3 – Принцип роботи сесії

Основна ідея полягає в тому, що будь-який авторизований запит проходить через додатковий запит в БД, що значно навантажує систему.

Принцип роботи JWT зображений на рисунку 2.4.

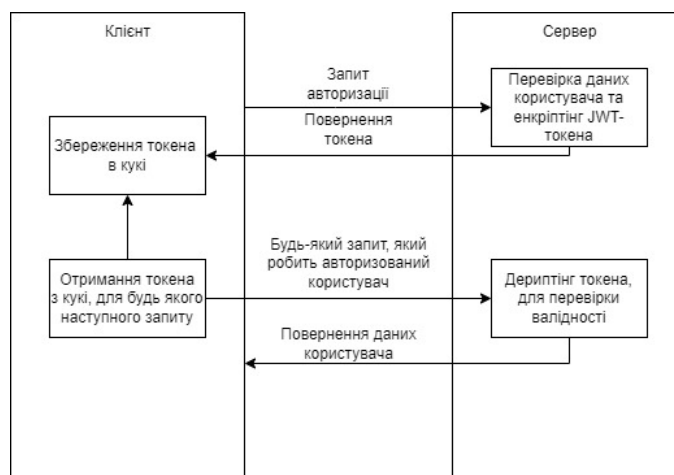


Рисунок 2.4 – Принцип роботи JWT

Уникнення додаткового запиту в БД, який пропонує принцип роботи з JWT, більш прийнятне.

Аутентифікація потрібна, якщо клієнт платформи бажає інтегрувати «онлайн оплату» безпосередньо в систему. В цьому випадку необхідно бути впевненим, що

запити які надсилаються ззовні мають бути завірені. Таку проблему вирішує підпис. Існують різні алгоритми підпису, але підхід асиметричного шифрування вирішує цю проблему. У разі витоку даних клієнта система не буде нести відповідальність за те, що потенційно саме через неї відбувся витік даних. Симетричне шифрування має лише один ключ, тож в цьому випадку така ймовірність існує.

#### 2.7.4 Другорядні бібліотеки

Вище виконаний аналіз бібліотек широкого спектру використання або їх концептуально важливого змісту. Обрана низка другорядних бібліотек, які не мають аналогів, та вирішують вузькі проблеми. Список таких бібліотек наведено в таблиці 2.19.

Таблиця 2.19 – Список другорядних бібліотек

Назва бібліотеки	Опис
Fastify	Фреймворк для веб-серверу [15]
Drizzle	ORM [16]
zod	Бібліотека валідації [17]
amqplib	Бібліотека спілкування з RabbitMQ [18]
InversifyJS	Бібліотека імплементації DI [19]
jsonwebtoken	Бібліотека генерації JWT [20]
dotenv	Бібліотека змінних оточення [21]
pino	Бібліотека логування [22]
iconv	Бібліотека генерування звітів [23]
ioredis	Бібліотека підключення до Redis [24]
typescript	Бібліотека TypeScript [25]

## 2.8 Вибір технологій для інтерфейсів

Для розробки було обрано Vue3 фреймворк для мови JavaScript. Vue3 – це прогресивний фреймворк для розроблення веб-додатків, створений для того, щоб бути простим у використанні, але досить потужним для побудови складних додатків. Vue3 є останньою версією цього фреймворку, яка представила кілька значних покращень у продуктивності, функціональності та гнучкості порівняно з попередніми версіями.

Особливості фреймворку:

- новий спосіб організації логіки компонентів, який дає більше гнучкості та покращує повторне використання коду. Це доповнює традиційний Options API;
- підтримка реактивності на основі Proxy;
- покращена продуктивність. Vue3 оптимізований для швидкості завантаження, рендерингу та оновлення. Це досягається завдяки новій компіляційній стратегії та оптимізації компонентів;
- підтримка Fragment, Teleport і Suspense.

### Висновки до розділу 2

Отже, обрана мікросервісна архітектура задля вирішення проблем масштабування та навантаженості. Обрані технології для роботи зі штучним інтелектом. Обрані технології для розроблення веб-інтерфейсів. Спроектowana БД, структура якої описує усі можливі варіанти використання. Також обрані протоколи взаємодії за допомогою шини повідомлень і саму шину повідомлень. Також розроблена структурна схема системи, зображена в додатку Б.

### 3 ОПИС СТРУКТУРИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

В ході виконання роботи розроблена діаграма прецедентів, яка описує усі можливі варіанти використання системи. Діаграма зображена в додатку В. Опис наведено в таблиці 3.1-3.3.

Таблиця 3.1 – Варіанти використання системи клієнтом

№	Опис варіанту використання	Включає	Розширює (номер варіанту використання)
1.	Завести особистий кабінет	-	-
2.	Авторизуватись в особистому кабінеті	-	1
3.	Створити посилання на інвойс	-	2
4.	Переглянути список платежів	-	2
5.	Отримати ключ доступу для інтеграції	-	2
6.	Повернути кошти після оплати	Перерахувати кошти	2
7.	Отримати звітність	-	2
8.	Додати реквізити	-	2

Таблиця 3.2 – Варіанти використання системи адміністратором

№	Опис варіанту використання	Включає	Розширює (номер варіанту використання)
1.	Авторизуватись на адміністративному порталі	-	-
2.	Налаштувати клієнта системи	-	1

№	Опис варіанту використання	Включає	Розширює (номер варіанту використання)
3.	Повернути кошти	Перерахувати кошти	1
4.	Заблокувати клієнта	-	1
5.	Вивантажити звітність клієнту	-	1
6.	Переглянути список платежів	-	1

Таблиця 3.3 – Варіанти використання системи платником

№	Опис варіанту використання	Включає	Розширює (номер варіанту використання)
1.	Сплатити послуги	-	-
2.	Переглянути квитанцію	-	1

Виходячи з усіх можливих варіантів використання системи, описані усі функціональні вимоги для кожної частини системи окремо, для більш чіткого розуміння бізнес вимог.

### 3.1 Функціональні вимоги до серверної частини

Серверна частина для еквайрингової платформи – фундамент усієї екосистеми продуктів. Функціональні вимоги до неї мають задовільняти усі потреби інтерфейсів, мати інтеграції з потрібними державними реєстрами та банками-екваєрами. Тож вимоги до серверної частини:

- можливість відкритої інтеграції з серверною частиною в обхід інтерфейсів;
- відповідність стандартам PCI DSS;
- інтеграція з банками-екваєрами;



- інтеграція з державними реєстрами для перевірки клієнтів за їх ІПН, якщо це ФОП, та ЄДРПОУ для це ТОВ;
- можливість проводити платежі через банки-екваєри;
- можливість взаємодіяти з платежами;
- можливість повернення коштів після оплати;
- можливість динамічного налаштування роутизації та створення клієнтів.

Зображено в додатку И;

- строга валідація вхідних платіжних даних;
- штучний інтелект для виявлення фроду;
- швидкість проведення платежів;
- архітектура, яка забезпечує роботу адміністраторів.

Дотримуючись усіх вищезгаданих вимог при розробленні серверної частини, остання задовольняє потребами необхідними для гнучкої взаємодії з системою.

Оскільки основною сутністю платформи є – інвойс, в таблицях 3.4-3.5 наведені усі вимоги до видів та типів інвойсів. Детальний вибір типу та виду інвойсу зображений в додатку Ж.

Таблиця 3.4 – Види однореквізитних інвойсів та платежів

Тип платежу	Вид інвойсу		
	Однореквізитний		
	Підтримка типу платежу	Підтримка вільної суми оплати	Підтримка багаторазовості
Авторизаційний	так	так	так
Преавторизаційний	так	так	так
Платіж-підписка	так	ні	так

В таблиці 3.5 зображено види муьтиреквізитних інвойсів.

Таблиця 3.5 – Види мультиреквізитних інвойсів та платежів

Тип платежу	Вид інвойсу		
	Мультиреквізитний		
	Підтримка типу платежу	Підтримка вільної суми оплати	Підтримка багаторазовості
Авторизаційний	так	Так	так
Преавторизаційний	так	так	так
Платіж-підписка	ні	ні	ні

Оскільки є різні типи платежів, важливо розділяти їх підсудності різними статусами, щоб легко орієнтуватись при виявленні проблеми. Статуси платежів наведені в таблиці 3.6. Також діаграма стану статусів платежу зображено в додатку Д.

Таблиця 3.6 – Статуси платежів

Для яких типів платежів	Статус	Опис статусу
Авторизаційний, Преавторизаційний, Платіж-підписка	CREATED	Платіж створено в системі, очікується створення транзакції в банку-екваєру
Авторизаційний, Преавторизаційний, Платіж-підписка	PENDING	Платіж в обробці банком-екваєром
Авторизаційний, Преавторизаційний, Платіж-підписка	NEEDS_CONFIRMATION	Платіж очікує підтвердження платником
Преавторизаційний	HELD	Суму на картці платника утримано, до завершення списання третьою стороною

Для яких типів платежів	Статус	Опис статусу
Авторизаційний, Преавторизаційний, Платіж-підписка	FINISHED	Платіж завершено третьою стороною в випадку преавторизаційного платежу. Та гроші списано в випадку інших платежів
Авторизаційний, Преавторизаційний	REVERSED	Суму платежу повернуто платнику
Авторизаційний, Преавторизаційний, Платіж-підписка	REJECTED	Платіж відхилено банком-екваєром

### 3.2 Функціональні вимоги до інтерфейсів

Легкий, інформативний та функціональний інтерфейс завжди має перевагу серед конкурентів. Оскільки еквайрингова платформа має декілька інтерфейсів, було розглянуті вимоги, які мають бути дотримані.

#### 3.2.1 Функціональні вимоги до платіжної сторінки

Платіжна сторінка є основним інструментом, яким користуватимуться більшість бізнесів, тому що:

- відсутність потреби мати будь-який додаток, сайт. Все що потрібно – створити посилання на оплату в особистому кабінеті;
- спрощений спосіб інтеграції. Клієнти, які інтегруються можуть перекласти відповідальність на еквайрингову платформу для надання сторінки оплати. В такому випадку клієнту потрібно інтегрувати лише точку входу створення інвойсу.

Тож функціональні вимоги:

- зручна форма для заповнення карткових даних;
- наявність індикатору завантаження, які повідомляють клієнту, що платіж в обробці;
- наявність зручного відображення відповіді обробки платежу;
- наявність можливості повернути платника на сторінку клієнта, звідки була ініційована оплата;
- можливість скачати квитанцію.

### 3.2.2 Функціональні вимоги до адміністративної панелі

Адміністративна панель є важливою частиною життєвого циклу еквайрингової платформи, оскільки інциденти стаються завжди, тому гарно продумане адміністрування – є вирішенням багатьох проблем.

Функціональні вимоги до адміністративної панелі:

- форма для авторизації адміністратора;
- форма для зміни паролю адміністратором;
- сторінка перегляду платежів;
- можливість зробити повернення;
- можливість працювати зі звітністю;
- можливість керування клієнтами;
- сторінка налаштування клієнтів.

### 3.2.3 Функціональні вимоги до особистого кабінету клієнта

Особистий кабінет виступає частиною екосистеми еквайрингової платформи, як додаткова можливість використання менеджменту платежів, звітності клієнтами. Також для малих бізнесів, які не мають свого ПЗ – це можливість створювати посилання на оплату інвойсів.

Функціональні вимоги до особистого кабінету клієнта:

- можливість створювати інвойси;
- можливість переглядати платежів;
- можливість повертати гроші;
- можливість додавання нових реквізитів;
- можливість видаляти реквізити;
- можливість вивантажувати звітність.

### 3.3 Структура мікросервісів серверної частини

Класичні мікросервіси виглядають як «одна сутність – один мікросервіс». Тим самим збільшується відмовостійкість, але швидкість роботи – зменшується. Натомість, обраний більш сконцентрований на оптимізацію підхід до вибору домену, де домен – є абстракцією з багатьма сутностями, які його описують. Нижче розглянуті сутності та їх властивості в ході розроблення ПЗ.

#### 3.3.1 Опис мікросервісу API-Gateway

API-Gateway мікросервіс – це точка доступу для реалізації клієнт-серверної архітектури та інтеграції клієнтів платформи. Реалізований за допомогою патерну API Gateway, задача якого полягає в маршрутизації трафіку запитів. Уся логіка рознесена по іншим сервісам.

Цей мікросервіс не має сутностей, але має багато контролерів та інших функцій, що відповідають за безпеку. Контролери були розглянути в таблицях 3.8-3.10.

Таблиця 3.8 – Контролер клієнтів

Назва методу	Додатковий захист	Опис методу
signUp	-	Реєстрація клієнта

Назва методу	Додатковий захист	Опис методу
signIn	-	Авторизація клієнта, результатом є отримання JWT
get	JWT	Отримання загальної інформації про клієнта із системи
getPublicKey	JWT	Отримання публічного ключа для інтеграції з точкою доступу

Таблиця 3.9 – Контролер карт платників

Назва методу	Додатковий захист	Опис методу
createCard	Для вже інтегрованих клієнтів – підпис запиту алгоритмом RSA-2048. Для платіжної сторінки відсутній, оскільки він не потрібен	Створення картки платника
getCard	Для вже інтегрованих клієнтів – підпис запиту алгоритмом RSA-2048. Для платіжної сторінки відсутній, оскільки він не потрібен	Отримання інформації про карту платника

Таблиця 3.10 – Контролер інвойсів

Назва методу	Додатковий захист	Опис методу
createInvoice	JWT або підпис запиту алгоритмом RSA-2048	Створення інвойсу
getInvoice	JWT або підпис запиту алгоритмом RSA-2048	Отримання інформації про інвойс
invoicePay	JWT або підпис запиту алгоритмом RSA-2048	Оплата інвойсу
checkStatus	JWT або підпис запиту алгоритмом RSA-2048	Отримання статусу проходження аутентифікації
confirmPayment	JWT або підпис запиту алгоритмом RSA-2048	Підтвердження LOOKUP
completePayment	JWT або підпис запиту алгоритмом RSA-2048	Завершення преавторизаційного платежу

Описані контролери є відкритими як для готових рішень платіжної сторінки, так і для повної інтеграції клієнта зі своєю системою. Наявні також закриті контролери, які відповідають за додаткові налаштування клієнтів, системи, та можливості надані в адміністративній панелі.

### 3.3.2 Опис мікросервісу InvoiceManager

Даний мікросервіс включає в себе роботу з інвойсами, платежами, зв'язки платежів з їх реквізитами. Також має інтеграції з банками-еквасрами, які проводять транзакції.

Нижче, в таблиці 3.11, описано структуру сутності інвойсів.

Таблиця 3.11 – Структура сервісу інвойсів

Назва властивості чи методу	Опис властивості чи методу
paymentService	Композиція класу сервісу платежів, опис якого знаходиться в таблиці 3.5
clientServicesService	Композиція класу реквізитів клієнта, опис якого знаходиться в таблиці 3.6
get	Отримання інформації про інвойс
checkStatus	Отримання статусу аутентифікації платежу
invoicePay	Оплата інвойсу
confirmPayment	Підтвердження LOOKUP
completePayment	Завершення преавторизаційного платежу

Таблиця 3.12 – Структура сервісу платежу

Назва властивості чи методу	Опис властивості чи методу
repository	Властивість, композиція класу підключення до БД
psm	Композиція класу зв'язку платежу з реквізитами
routeService	Композиція класу роутів
terminalService	Композиція класі терміналів
create	Створення платежу
calculateCommission	Розрахунок комісії в залежності від тіла платежу
checkStatus	Отримання статусу проходження 3DS перевірки
confirmPayment	Підтвердження аутентифікації LOOKUP



Назва властивості чи методу	Опис властивості чи методу
completePayment	Завершення преавторизаційного платежу

Таблиця 3.13 – Структура сервісу зв'язку платежу з реквізитами

Назва властивості чи методу	Опис властивості чи методу
psmRepository	Композиція класу підключення до БД
create	Створення зв'язку
get	Отримання зв'язку

Сутність зв'язку платежу з реквізитами потрібна, щоб покрити кейс, коли оплата має відбуватись одночасно за різні послуги, тобто оплата мультиреквізитного інвойсу. Приклад – оплата комунальних послуг. В цьому кейсі усі отримувачі мають різний IBAN. Тож сутність описує розділення абстрактної сутності платежів на менші частинки, для чіткого розподілення винагороди між усіма отримувачами.

Сервіс також відповідає за обрахунок комісій. Останні поділяються на три види, і кожна з них підбиратиметься точково під клієнта. Види комісій наведено в таблиці 3.14.

Таблиця 3.14 – Види комісій

Вид комісії	Опис комісії
Зовнішня комісія	Комісія, яку платник сплачуватиме поверх суми тіла платежу
Внутрішня комісія	Комісія, яку отримуватиме еквайрингова платформа, як винагорода за послуги
Банківська комісія	Комісія, яку знімає банк, змінюється в залежності від терміналу, який підключається клієнту

Сума до сплати визначається за формулою (3.1):

$$PayAmount = Fee + BodyAmount, \quad (3.1)$$

де *Fee* – зовнішня комісія;

*BodyAmount* – тіло платежу.

Сума для зарахування клієнту визначається за формулою (3.2):

$$ReceiveAmount = BodyAmount - (Commission + BankCommission), \quad (3.2)$$

де *BodyAmount* – тіло платежу;

*Commission* – внутрішня комісія;

*BankCommission* – банківська комісія.

Зовнішня та внутрішня комісія визначається за формулою (3.3):

$$Fee = FixedFee + Percent * BodyAmount, \quad (3.3)$$

де *FixedFee* – фіксована сума комісії;

*Percent* – відсоток комісії від тіла платежу;

*BodyAmount* – тіло платежу.

Банківська комісія завжди є фіксованою, та визначається банком-екваєром.

### 3.3.3 Опис мікросервісу AIModel

Основа задача цього мікросервісу – взаємодія з моделлю штучного інтелекту для виявлення фроду.

Для цього обраний підхід, який комбінує машинне навчання з поведінковим аналізом. Архітектура рішення, яка дозволяє забезпечити гнучкість та адаптивність моделі зображена на рисунку 3.1.



Рисунок 3.1 – Архітектура рішення

Виділені основні функції ознак для реалізації поведінкового аналізу:

- загальний оборот клієнта;
- частота оплат;
- середня сума транзакції;
- кількість запитів на оплату.

Для нормалізації даних виконане масштабування значень до діапазону від нуля до одиниці.

Навчання моделі реалізоване за допомогою інструментів обраних раніше. Приклад вибірки даних (датасету), яка навчала модель, наведений в таблиці 3.15.

Таблиця 3.15 – Приклад датасету для навчання моделі

Структура вибірки						Чи пропускати транзакцію
ID транзакції	Оборот клієнта, грн/місяць	Частота оплат, шт/місяць	Середній чек, грн	Кількість запитів, шт/год	Сума на оплату	
1	100 136	122	820	6	900	1
2	540 400	80	6755	10	5000	1
3	1 000 000	2034	491	250	6000	0
4	80 000	230	347	10 000	215	0

Таким чином для комбінування дані проходять через модель поведінкового аналізу, для виявлення аномалій, результатом цього аналізу додається додаткова ознака до ML-моделі. Основна модель використовує ці ознаки для класифікації транзакції.

Методи взаємодії з мікросервісом наведено в таблиці 3.16.

Таблиця 3.16 – Структура контролеру мікросервісу взаємодії з моделлю штучного інтелекту

Назва властивості чи методу	Опис властивості чи методу
startWatchingClient	Метод, який додає початкову інформацію для моделі штучного інтелекту, щоб остання орієнтувалась на них, під час прийняття рішень
isPaymentEnabled	Метод взаємодії із штучним інтелектом, який вирішує чи пропускати платіж

### 3.3.4 Опис мікросервісу ClientManager

Даний мікросервіс відповідає за збереження інформації про клієнтів, тобто тих, хто використовує платформу в своїх цілях. Також ця частина системи

відповідає за роботу з реквізитами клієнтів. Опис структури класів наведено в таблиці 3.17.

Таблиця 3.17 – Структура сервісу ClientManager

Назва властивості чи методу	Опис властивості чи методу
create	Створення клієнта
login	Авторизація клієнта, шляхом видачі JWT токена для роботи в особистому кабінеті
getPublicKey	Отримання ключа шифрування RSA 2048
getClient	Отримання інформації про клієнта

Також мікросервіс взаємодіє з реквізитами клієнта. Опис наведений в таблиці 3.18.

Таблиця 3.18 – Опис методів роботи з реквізитами

Назва властивості чи методу	Опис властивості чи методу
createService	Створення реквізиту
updateService	Оновлення реквізиту
deleteService	Видалення реквізиту
getServices	Отримання списку реквізитів клієнта

### 3.3.5 Опис мікросервісу Reporter

Даний сервіс відповідає за роботу з флет-даними та генерацію і маршрутизацію звітів.

Має декілька циклічних задач, наведених в таблиці 3.19.

Таблиці 3.19 – Відкладені циклічні задачі

Назва задачі	Опис задачі
invoicesFlat	Вивантаження кожної ночі усієї інформації, що стосується інвойсів, на основі якої будуть генеруватися звіти. Вивантаження відбувається з мікросервісу InvoiceManager
paymentsFlat	Вивантаження кожної ночі усієї інформації, що стосується платежів, на основі якої будуть генеруватися звіти. Вивантаження відбувається з мікросервісу InvoiceManager
clientsFlat	Вивантаження кожної ночі усієї інформації, що стосується клієнтів, на основі якої будуть генеруватися звіти. Вивантаження відбувається з мікросервісу ClientManager

Флет-дані – сирі вибірки даних, які потрібні для формування наступних звітів:

- IFOBS. Система електронного банкінгу для юридичних осіб та клієнтів, яка дозволяє здійснювати управління фінансами. Його задача – це деталізація транзакцій за певний період, який містить інформацію про залишки, надходження, а також структуризації виплат за IBAN, який робитиме бухгалтерський відділ. Генерується з paymentsFlat;

- звіт реєстрових платежів, який мають можливість отримати будь-який клієнт платформи, щоб ввести свій бухгалтерський облік. Генерується з paymentsFlat;

- звіт про клієнтів, який отримуватиме відділ аналітики. Генерується з clientsFlat;

- звіт про інвойси, який отримуватиме відділ аналітики. Генерується з invoicesFlat.

### 3.3.6 Опис мікросервісу AdminManager

Даний мікросервіс відповідає за роботу адміністраторів. Його задача контролювати адміністраторів системи, слідкувати за їх діями, а також керування правами доступу, для відповідності стандартам PCI DSS.

Ролі адміністраторів наведено в таблиці 3.20.

Таблиця 3.20 – Ролі адміністраторів

Роль	Опис
SUPPORT	Технічна підтримка, яка має можливість повертати кошти, працювати з платежами, та звітністю клієнтів
FINANCE_SPECIALIST	Фінансовий спеціаліст, який має доступ до бухгалтерської звітності
ANALITIC_SPECIALIST	Спеціаліст аналітики, який має доступ до аналітичних звітів, для подальшої обробки

В таблиці 3.21 наведено методи класів AdminManager мікросервіса.

Таблиця 3.21 – Методи класів AdminManagar

Назва властивості чи методу	Опис властивості чи методу
createAdmin	Створення адміна
getAdmin	Отримання інвормації про адміна
loginAdmin	Авторизація адміна
reversePayment	Повернення платежу
downloadReport	Вивантаження звіту
createLog	Створення дії адміна

### Висновки до розділу 3

В ході розроблення поставлені функціональні вимоги до усієї екосистеми платформи, а саме:

- платіжної сторінки;
- адміністративної панелі;
- особистого кабінету клієнта;
- серверної частини.

Також розроблена архітектура навчання і використання моделі штучного інтелекту для виявлення фроду. Описані основні мікросервіси, їх значення, та задачі, які вони вирішують. Висвітлені ролі адміністраторів, стани статусів платежів, різниці між комісіями, та підбір типу інвойсу.



## 4 ТЕСТУВАННЯ

Мета тестування полягає у перевірці коректної роботи системи, забезпеченні правильності реалізації всіх сценаріїв використання, визначених у вимогах та діаграмах прецедентів. Також тестування спрямоване на виявлення помилок і недоліків у функціонуванні системи для їх подальшого аналізу та усунення, що дозволяє забезпечити стабільність та відповідність системи заданим критеріям якості.

Для тестування серверної частини використовувався Postman, який дозволяє надсилати HTTP запити.

Для тестування інтерфейсів системи, обраний ручний варіант, описані тест-кейси, для більш широкого покриття тестами.

### 4.1 Тестування веб-серверу

Тестування веб-серверу розбите на блоки частин функціоналу:

- робота з клієнтом;
- робота з інвойсами;
- робота з адміністраторами;
- робота зі звітністю

Нижче розглянуті тести, які проводились.

#### 4.1.1 Тестування роботи з клієнтом

Тестування роботи з клієнтом включає:

- реєстрацію клієнта;
- авторизацію клієнта;
- роботу з реквізитами;
- отримання ключа для інтеграції з точкою входу.

В таблицях 4.1-4.4 описані тест-кейси, очікувана поведінка та результати.

Таблиця 4.1 – Тест-кейси реєстрації клієнта

Опис кейсу	Результат тесту	Очікуваний результат
Успішна реєстрація	Статус відповіді 201.	Статус відповіді 201
Неуспішна реєстрація через низьку складність паролю	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки
Неуспішна реєстрація через несправжню поштову адресу	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки

Таблиця 4.2 – Тест-кейси авторизації клієнта

Опис кейсу	Результат тесту	Очікуваний результат
Успішна авторизація	Статус відповіді 200. Тіло відповіді містить JWT токен для подальшої взаємодії з системою	Статус відповіді 201. Тіло відповіді містить JWT токен для подальшої взаємодії з системою
Неуспішна авторизація через невідповідність паролів	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки
Неуспішна авторизація через не відповідність поштової адреси	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки

Таблиця 4.3 – Тест-кейси роботи з реквізитами

Опис кейсу	Результат тесту	Очікуваний результат
Успішне створення реквізитів	Статус відповіді 201. Тіло відповіді містить ідентифікатор реквізиту	Статус відповіді 201. Тіло відповіді містить ідентифікатор реквізиту

Опис кейсу	Результат тесту	Очікуваний результат
Неуспішне створення реквізитів через невалідність IBAN	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки
Успішне оновлення реквізиту	Статус відповіді 200. Тіло відповіді повертає той же ідентифікатор	Статус відповіді 200. Тіло відповіді повертає той же ідентифікатор
Неуспішне оновлення реквізиту через невалідність IBAN	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки
Успішне видалення реквізиту	Статус відповіді 206	Статус відповіді 206

Таблиця 4.4 – Тест кейси отримання ключа для інтеграції

Опис кейсу	Результат тесту	Очікуваний результат
Успішне отримання ключа доступу	Статус відповіді 200. Тіло відповіді містить публічний ключ алгоритму RSA 2048	Статус відповіді 200. Тіло відповіді містить публічний ключ алгоритму RSA 2048
Неуспішне отримання ключа доступу через відсутність JWT-токена	Статус відповіді 401. Тіло відповіді містить текст помилки	Статус відповіді 401. Тіло відповіді містить текст помилки

Тестування роботи з клієнтом пройдено успішно.

#### 4.1.2 Тестування роботи з інвойсами

Тестування роботи з інвойсами включає:

- створення інвойсу;
- отримання інформації інвойсу за його ідентифікатором;
- створення картки в системі, для сплати;
- оплати інвойсу;
- отримання списку платежів;
- повернення коштів платнику.

В таблицях 4.5-4.10 описані тест-кейси, очікувана поведінка та результати.

Таблиця 4.5 – Тест-кейси створення інвойсу

Опис кейсу	Результат тесту	Очікуваний результат
Успішне створення одnoreквізитного інвойсу	Статус відповіді 201. Тіло відповіді містить посилання на оплату	Статус відповіді 201. Тіло відповіді містить посилання на оплату
Успішне створення мультиреквізитного інвойсу	Статус відповіді 201. Тіло відповіді містить посилання на оплату	Статус відповіді 201. Тіло відповіді містить посилання на оплату
Неуспішне створення інвойсу через неправильні ідентифікатори реквізитів	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки
Успішне створення інвойсу з різними типами платежу	Статус відповіді 201. Тіло відповіді містить посилання на оплату	Статус відповіді 201. Тіло відповіді містить посилання на оплату
Опис кейсу	Результат тесту	Очікуваний результат

Успішне створення інвойсу без чіткої суми оплати	Статус відповіді 201. Тіло відповіді містить посилання на оплату	Статус відповіді 201. Тіло відповіді містить посилання на оплату
--	--	--

Таблиця 4.6 – Тест-кейси отримання інформації про інвойс

Опис кейсу	Результат тесту	Очікуваний результат
Успішне отримання інформації	Статус відповіді 200. Тіло відповіді містить усю інформацію про інвойс та платіж	Статус відповіді 200. Тіло відповіді містить усю інформацію про інвойс та платіж
Неуспішне отримання через невідповідність ідентифікатора	Статус відповіді 404. Тіло відповіді містить текст помилки	Статус відповіді 404. Тіло відповіді містить текст помилки

Таблиця 4.7 – Тест-кейси роботи з картою платника в системі

Опис кейсу	Результат тесту	Очікуваний результат
Успішне створення карти	Статус відповіді 201. Тіло відповіді містить ідентифікатор карти в системі	Статус відповіді 201. Тіло відповіді містить ідентифікатор карти в системі
Неуспішне створення картки через невалідність номеру карти	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки
Неуспішне створення карти через часу життя, який не є майбутнім часом	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки
Опис кейсу	Результат тесту	Очікуваний результат

Успішне отримання інформації картки платника	Статус відповіді 200. Тіло відповіді містить замасковану інформацію про номер карти та ідентифікатор в системі	Статус відповіді 200. Тіло відповіді містить замасковану інформацію про номер карти та ідентифікатор в системі
Неуспішне отримання інформації карти платника	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки

Таблиця 4.8 – Тест-кейси оплати інвойсу

Опис кейсу	Результат тесту	Очікуваний результат
Успішна оплата інвойсу	Статус відповіді 201. Тіло відповіді містить ідентифікатор інвойсу	Статус відповіді 201. Тіло відповіді містить ідентифікатор інвойсу
Неуспішна оплата інвойсу через недостатню кількість коштів	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки
Неуспішна оплата інвойсу через некоректність переданого ідентифікатора карти платника	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки
Неуспішна оплата інвойсу через невалідність переданого IBAN	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки
Опис кейсу	Результат тесту	Очікуваний результат

Успішне підтвердження преавторизаційного платежу	Статус відповіді 200. Тіло відповіді містить ідентифікатор інвойсу та платежу	Статус відповіді 200. Тіло відповіді містить ідентифікатор інвойсу та платежу
--	---	---

Таблиця 4.9 – Тест-кейси отримання списку платежів

Опис кейсу	Результат тесту	Очікуваний результат
Успішний пошук з пагінацією	Статус відповіді 200. Тіло відповіді містить інформацію про інвойс та оплату	Статус відповіді 200. Тіло відповіді містить інформацію про інвойс та оплату
Успішний пошук з фільтрацією по періоду оплати	Статус відповіді 200. Тіло відповіді містить інформацію про інвойс та оплату, відфільтрований по періоду оплати	Статус відповіді 200. Тіло відповіді містить інформацію про інвойс та оплату, відфільтрований по періоду оплати
Успішний пошук з фільтрацією по сумі оплати	Статус відповіді 200. Тіло відповіді містить інформацію про інвойс та оплату відфільтрований по сумі оплаті	Статус відповіді 200. Тіло відповіді містить інформацію про інвойс та оплату відфільтрований по сумі оплаті
Пустий список, якщо задати дату майбутнього	Статус відповіді 206. Тіло відповіді містить пустий масив даних	Статус відповіді 206. Тіло відповіді містить пустий масив даних

Опис кейсу	Результат тесту	Очікуваний результат
Успішний пошук з фільтрацією по статусу оплати	Статус відповіді 200. Тіло відповіді містить інформацію про інвойс та оплату відфільтрований по статусу	Статус відповіді 200. Тіло відповіді містить інформацію про інвойс та оплату відфільтрований по статусу

Таблиця 4.10 – Тест-кейси повернення коштів платнику

Опис кейсу	Результат тесту	Очікуваний результат
Успішне повернення коштів	Статус відповіді 200. Тіло відповіді містить статус платежу, суму повернення та ідентифікатор інвойсу	Статус відповіді 200. Тіло відповіді містить статус платежу, суму повернення та ідентифікатор інвойсу
Неуспішне повернення коштів, через те що кошти вже повернуто	Статус відповіді 400. Тіло відповіді вказує, що кошти вже повернуто	Статус відповіді 400. Тіло відповіді вказує, що кошти вже повернуто
Неуспішне повернення коштів, через те що інвойс ще не був оплачений	Статус відповіді 400. Тіло відповіді вказує, що інвойс ще не оплачений	Статус відповіді 400. Тіло відповіді вказує, що інвойс ще не оплачений

Тестування роботи інвойсів успішно пройдено.

#### 4.1.3 Тестування роботи адміністраторів

Тестування роботи адміністраторів включає:

- авторизація адміністратора;
- права доступу згідно ролі;
- логування усіх дій адміністратора в системі.



В таблицях 4.11-4.13 описані тест-кейси, очікувана поведінка та результати.

Таблиця 4.11 – Тест-кейси авторизації адміністраторів

Опис кейсу	Результат тесту	Очікуваний результат
Успішна авторизація	Статус відповіді 200. Тіло відповіді містить JWT токен для подальшої взаємодії з системою	Статус відповіді 201. Тіло відповіді містить JWT токен для подальшої взаємодії з системою
Неуспішна авторизація через невірність паролю	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки
Неуспішна авторизація через невірність поштової адреси	Статус відповіді 400. Тіло відповіді містить текст помилки	Статус відповіді 400. Тіло відповіді містить текст помилки

Таблиця 4.12 – Тест-кейси прав доступу

Опис кейсу	Результат тесту	Очікуваний результат
FINANCE_SPECIALIST повертає гроші	Запит відхилено, через права доступу	Запит відхилено, через права доступу
SUPPORT повертає гроші	Запит виконано, права доступу присутні	Запит виконано, права доступу присутні
SUPPORT вивантажує звітність	Запит виконано, права доступу присутні	Запит виконано, права доступу присутні

Таблиця 4.13 – Тест-кейси логування дій адміністраторів

Опис кейсу	Результат тесту	Очікуваний результат
Адміністратор авторизувався	В БД записалась дія «авторизувався»	В БД записалась дія «авторизувався»
Опис кейсу	Результат тесту	Очікуваний результат

Адміністратор повернув кошти	В БД записалась дія «повернув кошти»	В БД записалась дія «повернув кошти»
Адміністратор вивантажив звітність	В БД записалась дія «вивантажив звітність»	В БД записалась дія «вивантажив звітність»

Тестування роботи системи для адміністраторів успішно пройдене.

#### 4.1.4 Тестування звітності

Звіти є невід’ємною частиною функціонування фінансових установ. В ході тестування переглянута структура сформованих звітів перелічених раніше.

Структура звіту IFOBS зображена на рисунку 4.1.

```
<?xml version="1.0" encoding="WINDOWS-1251"?>
<ROWDATA>
  <ROW>
    DOCUMENTDATE="20231107"
    DOCUMENTNO="1708034401"
    AMOUNT="314850"
    IBAN="UA243770900000026502001301523"
    CORRIBAN="UA243770900000035002001401624"
    CORRNAME="ТОВ 'Постачальник Послуг'"
    CORRIDENTIFYCODE="12345678"
    DETAILSOFPAYMENT="Оплата за інтернет, рахунок №12345"
  />
  <ROW>
    DOCUMENTDATE="20231108"
    DOCUMENTNO="1708034402"
    AMOUNT="102500"
    IBAN="UA243770900000026502001301523"
    CORRIBAN="UA243770900000045602001501725"
    CORRNAME="ТОВ 'Комунальні Платежі'"
    CORRIDENTIFYCODE="87654321"
    DETAILSOFPAYMENT="Оплата за електроенергію"
  />
</ROWDATA>
```

Рисунок 4.1 – IFOBS звіт

Звіт має чітку структуру, згідно вимог до використання такого типу звітів, наповнення атрибутів відбувається з зазначеним кодуванням, та має формат XML.

Наступним є звіт про реєстрові платежі. Його зміст надає клієнтам необхідну інформацію для ведення бухгалтерського обліку в CSV форматі. Вивантажений і перевірений звіт зображений на рисунку 4.2.

```
PaymentAmount,InternalFee,ExternalFee,BankFee,PaymentStatus,PaymentPurpose,PaymentID
314850,500,200,300,FINISHED,"Оплата за услуги интернету",1708034401
102500,300,150,250,REVERSED,"Оплата за коммунальные услуги",1708034402
500000,1000,400,600,FINISHED,"Оплата за товар, рахунок №12345",1708034403
```

Рисунок 4.2 – Звіт реєстрових платежів

Перевірені звіти з інформацією про клієнтів і інвойсів зображено на рисунках 4.3-4.4.

```
ClientID,ClientType,CompanyNameOrFullName,MCC,TransactionCount,Turnover,ContractDate
12345,FOP,"Іваненко Іван Іванович",5812,120,3500000,2023-01-15
67890,TOV,"ТОВ 'Ресторація Смаку'",5813,85,1450000,2021-09-10
11223,FOP,"Петренко Олег Миколайович",4829,65,720000,2022-03-05
44556,TOV,"ТОВ 'Техно Плюс'",5732,200,5300000,2020-11-20
```

Рисунок 4.3 – Звіт з інформацією про клієнтів

```
InvoiceID, InvoiceType, PaymentAmount, Status, CreationDate
101, Single, 2500, Active, 2023-11-01
102, Recurring, 5000, Inactive, 2022-08-15
103, Single, 3200, Active, 2023-10-05
104, Recurring, 1500, Inactive, 2021-12-20
```

Рисунок 4.4 – Звіт з інформацією про інвойси

## 4.2 Тестування сторінки оплати

Сторінка оплати (рисунок 4.5) протестована на відображення результату в залежності від стану оплати.

**Payment**

Card number

0000 0000 0000 0000

Expiry date      CVV

MM/YY      \*\*\*

Cardholder name

John Doe

By clicking on the 'Pay' button, you confirm that you are familiar with the list of information about the service and accept the terms of the public agreement.

**Pay 100 UAH**

Рисунок 4.5 – Сторінка оплати

На рисунку 4.6 зображена сторінка після успішної оплати інвойсу.

**Payment Successful**

**Receipt**

Payment Method	Visa **** 1234
Payment Amount	100.00UAH
Date	Nov 1, 2024
Service	Professional Web Design (2 Weeks)

**Return to Site**

Рисунок 4.6 – Успішна оплата

На рисунку 4.7 зображена сторінка після неуспішної оплати інвойсу.

**Payment failed**

We couldn't process your payment. Please check with your bank to make sure you have enough funds or if there are any other issues.

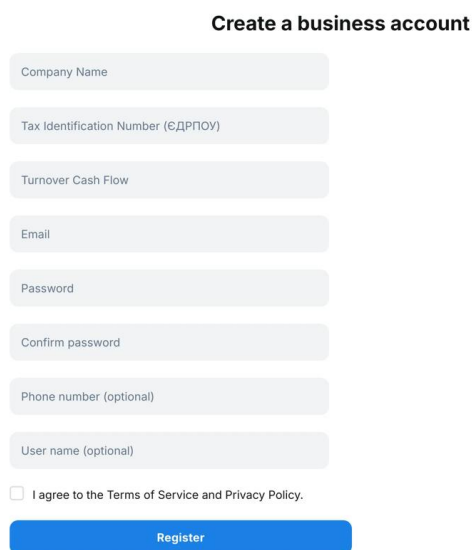
**Try again**

Рисунок 4.7 – Неуспішна оплата

Тестування платіжної сторінки пройдено успішно.

#### 4.3 Тестування особистого кабінету користувача

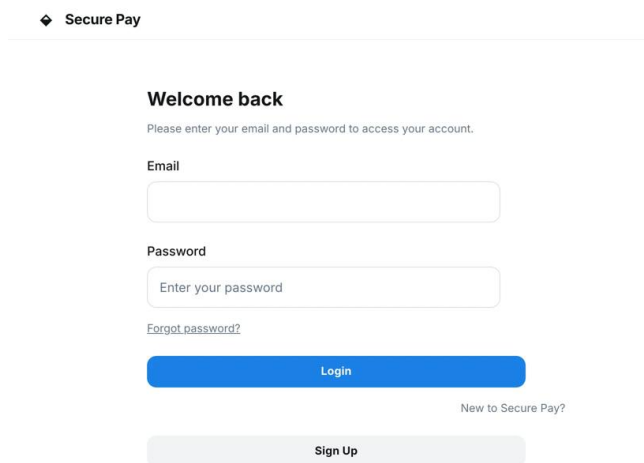
Особистий кабінет простий та інтуїтивно зрозумілий. На рисунку 4.8 зображена форма створення бізнес акаунту для клієнтів платформи.



The form is titled "Create a business account". It contains several input fields: "Company Name", "Tax Identification Number (ЄДРПОУ)", "Turnover Cash Flow", "Email", "Password", "Confirm password", "Phone number (optional)", and "User name (optional)". Below the fields is a checkbox labeled "I agree to the Terms of Service and Privacy Policy." and a blue "Register" button.

Рисунок 4.8 – Форма створення бізнес акаунту

На рисунку 4.9 зображена форма автотризації в особистий кабінет



The form is titled "Welcome back" and includes the instruction "Please enter your email and password to access your account." It features input fields for "Email" and "Password" (with the placeholder "Enter your password"). A link "Forgot password?" is located below the password field. There is a blue "Login" button and a grey "Sign Up" button. The text "New to Secure Pay?" is positioned above the "Sign Up" button. The form is part of the "Secure Pay" interface, as indicated by the logo at the top.

Рисунок 4.9 – Форма авторизації

Після успішного створення особистого кабінету та авторизації, клієнт має змогу створити посилання на інвойс, завантажити публічний ключ для інтеграції та переглянути список доданих реквізитів (рисунок 4.10).

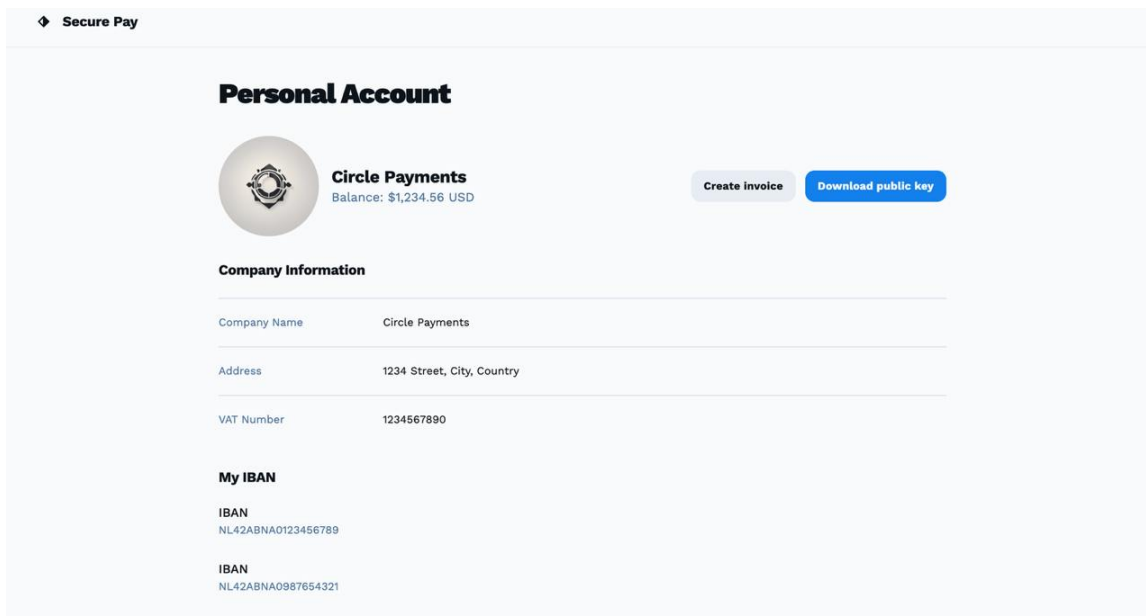


Рисунок 4.10 – Особистий кабінет клієнта

На рисунку 4.11 зображений розділ перегляду платежів

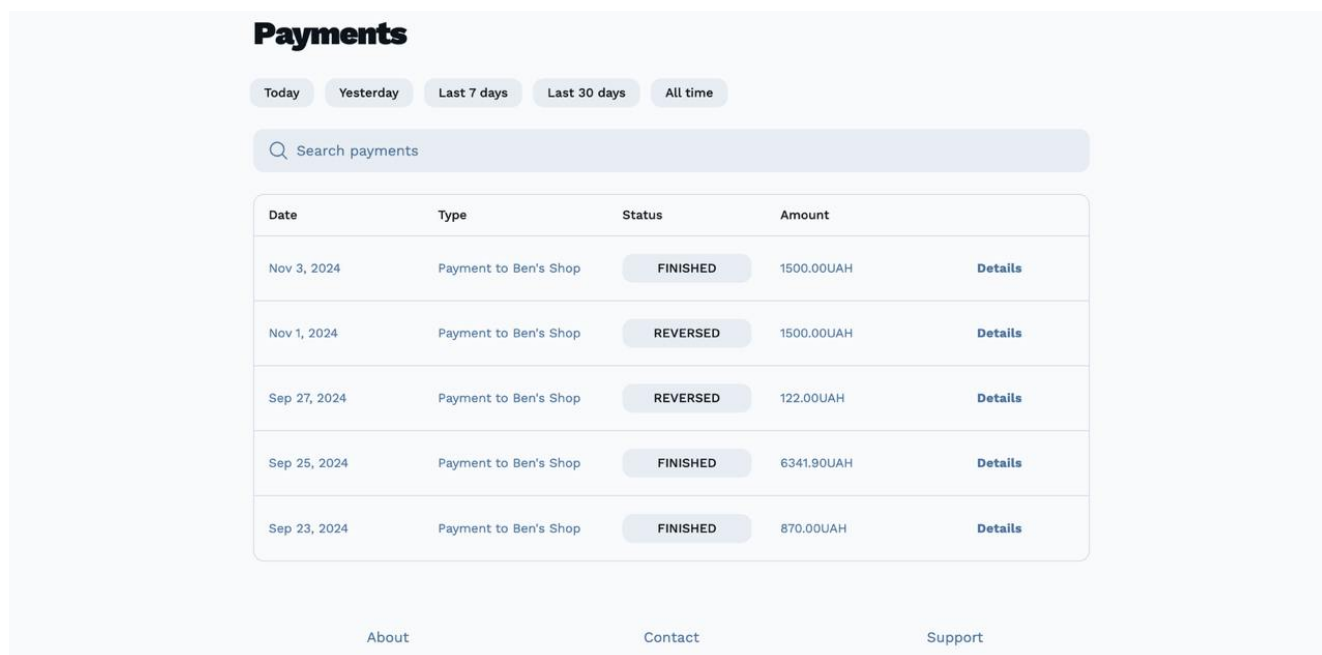


Рисунок 4.11 – Розділ пошуку і перегляду списку платежів

## Висновки до розділу 4

У розділі тестування проведена всебічна перевірку функціональних компонентів еквайрингової платформи, що включає модель штучного інтелекту для виявлення фроду, веб-сервер, а також: платіжну сторінку та особистий кабінет клієнта. Метою тестування є забезпечення коректності роботи кожної складової платформи, відповідності описаним сценаріям використання, а також виявлення та усунення потенційних помилок.

Результати тестування підтвердили функціональну готовність платформи до впровадження, її здатність ефективно виявляти та попереджати шахрайські операції, а також забезпечувати надійний і зручний сервіс для користувачів. Виконані заходи з виявлення та усунення помилок дозволили значно підвищити стабільність і продуктивність системи, забезпечивши її відповідність сучасним вимогам якості та безпеки.

## 5 РОЗРОБЛЕННЯ СТАРТАП ПРОЄКТУ

### 5.1 Опис ідеї проєкту

Еквайрингова платформа, яка використовує штучний інтелект для виявлення шахрайства, пропонує сучасне рішення для оптимізації процесів обробки платежів.

Для оцінки потенціалу впровадження проєкту та визначення ключових ринкових можливостей, розроблений опис базової ідеї стартапу. У ньому деталізуються зміст пропозиції, напрями застосування платформи в різних галузях та вигоди, які отримують користувачі.

В таблиці 5.1 наведені основні характеристики стартапу, включаючи унікальність рішення, можливості адаптації до різних сегментів ринку та користь для цільової аудиторії.

Таблиця 5.1 – Опис ідеї стартап-проєкту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Еквайрингова платформа з особистим кабінетом та можливістю створення посилань на різні типи інвойсів	Послуги фінансових технологій (FinTech)	Автоматизація створення та обробки інвойсів, що знижує витрати часу на ручну роботу
Платформа забезпечує створення та проведення оплат по мультиреквізитним, одноразовим і багаторазовим інвойсам	Інтернет-торгівля (e-commerce)	Забезпечення зручності для клієнтів через підтримку мультиреквізитності та різних способів оплати
Унікальність рішення в застосуванні штучного інтелекту для детекції фроду	Бухгалтерські та платіжні системи підприємств	Підвищення рівня безпеки транзакцій за допомогою алгоритмів ШІ для виявлення шахрайських дій



Зміст ідеї	Напрямки застосування	Вигоди для користувача
Підтримка різних типів бізнесу: малий, середній, великий	Криптовалютні платформи	Адаптація до потреб компаній різного масштабу, від малого бізнесу до корпорацій
Простий та інтуїтивно зрозумілий інтерфейс особистого кабінету	Онлайн-сервіси з підписками (SaaS-платформи)	Спрощення взаємодії для користувачів через доступний і зрозумілий інтерфейс
Підтримка регулярних платежів та налаштування автоматичних інвойсів	Роздрібна торгівля з інтеграцією платіжних рішень	Зручність повторних платежів для постійних клієнтів через автоматизацію процесів
Гнучка інтеграція платформи з існуючими системами підприємств	Послуги фінансових технологій (FinTech)	Легка інтеграція з іншими сервісами або бухгалтерськими програмами, що знижує витрати на адаптацію

Для оцінки конкурентоспроможності еквайрингової платформи проведений порівняльний аналіз її техніко-економічних характеристик із основними конкурентами: LiqPay, UAPAY та WayForPay. У таблиці 5.2 визначено сильні, слабкі та нейтральні сторони проєкту, що дозволяє сформулювати стратегічні пріоритети для вдосконалення платформи.

Таблиця 5.2 – Технологічна здійсненність ідеї проекту

№	Техніко-економічні характеристики ідеї	Товари/концепції конкурентів				Слаб-ка сторона	Нейт-раль-на сто-рона	Силь-на сто-рона
		Роз-робле-на систе-ма	Конку-рент LiqPay	Конку-рент UAPA Y	Конку-рент WayForPay			
1	Швидкість обробки платежів	Ви-сока	Висо-ка	Висо-ка	Висока		+	
2	Зручність інтерфейсу	Ви-сока	Сере-дня	Сере-дня	Висока			+
3	Інтеграція ШІ для виявлення шахрайства	+	-	-	-			+
4	Гнучність комісій	Ви-сока	Сере-дня	Сере-дня	Середня			+
5	Точність виявлення шахрайства	Ви-сока	Низь-ка	Немає	Немає			+
6	Вартість інтергації	Низь-ка	Сере-дня	Сере-дня	Висока			+
7	Доступність API	+	+	+	+		+	
8	Підтримка e-commerce платежів	Усі	Усі	Усі	Усі		+	

№	Техніко-економічні характеристик і ідеї	Розроблена система	Конкурент LiqPay	Конкурент UAPA Y	Конкурент WayForPay	Слабка сторона	Нейтральна сторона	Сильна сторона
10	SDK для мобільної інтеграції	Немає	Є	Є	Є	+		

### 5.3 Технологічний аудит ідеї проекту

Таблиця 5.3 – Технологічна здійсненність ідеї проекту

№	Ідея проекту	Технології та реалізація	Наявність технологій	Доступність технологій
1.	Система управління платежами	Інтеграція з банками-еквівалентами	Існують. Потребують інтеграції	Потребують договорів
2.	Відкритий API	Розробка API для роботи з платіжними системами	Існують	Доступні
3.	Алгоритми штучного інтелекту для виявлення шахрайства	TensorFlow, моделі Deep Learning	Існують	Доступні
4.	Особистий кабінет	Фронтенд і бекенд розробка	Потрібно розробити	Доступні

№	Ідея проєкту	Технології реалізація	Наявність технологій	Доступність технологій
5.	Реєстрація клієнта	Фронтенд і бекенд розробка, а також інтеграція з відкритими реєстрами	Потрібно розробити	Потребують договорів
Основні компоненти MVP включають інтеграцію з банками-еквівалентами, базовий функціонал API, а також ключові алгоритми фроду на базі відкритих бібліотек III.				

За результатами аналізу технологічна реалізація проєкту можлива, але з певними обмеження.

### 5.3 Аналіз ринкових можливостей запуску стартап-проєкту

Для оцінки потенціалу ринкових можливостей стартап-проєкту важливо провести попередній аналіз стану ринку. Це дозволяє зрозуміти поточний стан конкурентного середовища, обсяг ринку, динаміку його розвитку, а також можливі бар'єри для входу та вимоги до сертифікації. Нижче наведена таблиця 3.3 з ключовими показниками, що допомагають оцінити ринок та сформулювати стратегічний план впровадження продукту.

Таблиця 5.4 – Попередня характеристика потенційного ринку стартап-проєкту

№	Показники стану ринку (найменування)	Характеристика
1.	Кількість головних гравців, од	Близько 10 ключових гравців, включаючи LiqPay, UAPAY, WayForPay
2.	Загальний обсяг продаж, грн/ум.од	Понад 200 млрд грн на рік (обсяг електронних платежів в Україні)
3.	Показники стану ринку (найменування)	Характеристика
4.	Динаміка ринку (якісна оцінка)	Зростає через активне впровадження фінтех-рішень та електронної комерції
5.	Наявність обмежень для входу	Юридичні: реєстрація як ТОВ або ФОП. Специфічні ліцензії не потрібні
6.	Специфічні вимоги до стандартизації та сертифікації	Відповідність PCI DSS (сертифікація забезпечує безпеку даних карток)
7.	Середня норма рентабельності в галузі, %	10–30% залежно від обсягу транзакцій та структури витрат

В таблиці 5.5 наведені потенційні групи клієнтів, характеристики, які покриває потреба цих груп системою та орієнтовний перелік вимог до продукту.

Таблиця 5.5 – Характеристика потенційних клієнтів стартап-проєкту

№	Потреба, що формує ринок	Цільова аудиторія	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1.	Безпека транзакцій, зниження рівня шахрайства, швидкість обробки платежів	Малий та середній бізнес (e-commerce, фізичні магазини), Фінтех-компанії, Клієнти великих корпорацій	Малий бізнес шукає простоту у впровадженні. Банки потребують інтеграції з іншими системами та високого рівня безпеки	До продукції: інтуїтивно зрозумілий інтерфейс, точність у виявленні фроду, швидкість обробки До постачальника: технічна підтримка, наявність ліцензій, довіра до бренду

Після визначення груп клієнтів, проведений аналіз ринкового середовища, а саме факторів загроз та можливостей, які наведені в таблицях 5.6-5.7.

Таблиця 5.6 – Фактори загроз

№	Фактор	Зміст загрози	Можлива реакція компанії
1.	Висока конкуренція	Наявність сильних гравців на ринку, що вже мають лояльну клієнтську базу	Використання сучасних технологій для підвищення точності та швидкості обробки транзакцій
2.	Складність отримання ліцензій PCI DSS	Регуляторні вимоги можуть затягнути процес виходу на ринок	Найм фахівців з юридичного супроводу та партнерство з регуляторами
3.	Технічні проблеми	Можливі збої у роботі платформи, особливо на етапі запуску	Тестування платформи перед запуском, найм висококваліфікованих технічних спеціалістів
4.	Регуляторні бар'єри	Високі витрати на відповідність нормам знижують рентабельність	Залучення інвесторів для покриття початкових витрат
5.	Брак клієнтів	Відсутність користувачів може призвести до збитків через договори і оплату послуг партнерів	Активна маркетингова кампанія для залучення клієнтів і партнерів

Таблиця 5.7 – Фактори можливостей

№	Фактор	Зміст можливості	Можлива реакція компанії
1.	Зростання попиту на безпечні транзакції	Бізнеси все більше шукають рішення для зниження шахрайства та підвищення безпеки фінансових операцій	Розроблення та інтеграція інноваційних алгоритмів на основі штучного інтелекту для виявлення фроду
2.	Тенденція діджиталізації бізнесу	Зростання кількості e-commerce і потреба у платіжних рішеннях	Просування продукту через партнерства з інтернет-магазинами та платформами електронної комерції
3.	Інтерес інвесторів до фінтех-проектів	Фінансові компанії та венчурні фонди активно інвестують у новітні технології	Активне залучення інвесторів через презентації, демонстрацію потенційного прибутку і високої рентабельності
4.	Розвиток партнерських мереж	Можливість інтеграції з іншими платформами (банківськими сервісами, платіжними системами)	Налагодження партнерств із банками, фінтех-компаніями та глобальними платіжними системами
5.	Попит на прості та зручні рішення	Малий та середній бізнес шукає платформу з інтуїтивним інтерфейсом	Забезпечення простого дизайну і максимальної зручності для користувачів

В таблиці 5.8 наведений аналіз конкуренції на ринку.



Таблиця 5.8 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства
1. Тип конкуренції: олігополія	Ринок онлайн-еквайрингу в Україні домінують кілька великих гравців (LiqPay, UAPAY, WayForPay)	Використання унікальних інновацій (ШІ для детекту фроду), пропозиція конкурентних цін та функцій
2. Рівень конкурентної боротьби: національний	Ринок обмежений межами України, основні гравці – українські фінтех-компанії	Зосередження на потребах українських користувачів, локалізація функцій та високий рівень підтримки клієнтів
3. За галузевою ознакою: внутрішньогалузева	Конкуренція між фінтех-компаніями, які надають послуги онлайн-еквайрингу	Виділення через технологічні інновації, особливо ШІ для фроду, і покращення функціоналу
4. Конкуренція за видами товарів: товарно-видова	Основний акцент на послугах онлайн-платежів із функціональними відмінностями	Баланс між функціональністю, зручністю та конкурентною ціною
5. За характером конкурентних переваг: цінова та нецінова	Основні переваги – доступна ціна, інновації (ШІ для виявлення фроду) та зручність використання	Пропонувати конкурентні ціни та унікальний функціонал для залучення клієнтів

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства
6. За інтенсивністю: немарочна	Платформа не має брендової впізнаваності, в той час як конкуренти мають сформовані бренди	Інвестувати у маркетинг, брендинг і побудову довіри серед клієнтів

Таблиця 5.9 – Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
	LiqPay, UAPAY, WayForPay	Компанії, які можуть увійти на ринок з подібними послугами	Постачальники серверів, хмарних послуг, API для ІІІ	Українські підприємці, онлайн-магазини, фінансові компанії	Альтернативні методи оплати (банківські перекази, криптовалюти, готівковий розрахунок)
Висновки	Інтенсивність конкуренції висока через велику кількість	Бар'єри входження високі через необхідність інвестицій та	Постачальники можуть диктувати умови через обмежену кількість якісних рішень	Клієнти диктують вимоги до безпеки, зручності та швидкості роботи	Замінники можуть становити загрозу через дешевизну та доступність

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
	реалізованих систем	технічної складності			

Таблиця 5.10 – Обґрунтування факторів конкурентоспроможності

№	Фактор конкурентоспроможності	Обґрунтування
1.	Використання штучного інтелекту	Забезпечує високу точність і швидкість виявлення фроду, що відсутнє у конкурентів
2.	Інтеграція з іншими системами	Легке впровадження в існуючу інфраструктуру бізнесу завдяки готовим SDK
3.	Висока швидкість обробки даних	Забезпечує мінімізацію часу транзакцій і кращий клієнтський досвід
4.	Відповідність регуляторним вимогам	Сертифікація НБУ та PCI DSS підвищує довіру споживачів і відповідає законодавству
5.	Зручність використання	Інтуїтивно зрозумілий інтерфейс і адаптивність під потреби різних користувачів
6.	Конкурентна ціна	Оптимальний баланс між ціною і функціональністю порівняно з іншими платформами

За результатами обґрунтування факторів конкурентоспроможності оцінений і порівняний проєкт із конкурентами в таблиці 5.11.

Таблиця 5.11 – Порівняльний аналіз сильних та слабких сторін «Еквайрингової платформи»

№	Фактор конкуренто-спроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з UAPAY, LiqPay, WayForPay						
			-3	-2	-1	0	+1	+2	+3
	Використання штучного інтелекту	20	UAPAY, LiqPay, WayForPay						
	Інтеграція з іншими системами	10			UAPAY			LiqPay, WayForPay	
	Висока швидкість обробки даних	19			UAPAY, LiqPay, WayForPay				
	Відповідність регуляторним вимогам	20						UAPAY, LiqPay, WayForPay	
	Зручність використання	15						UAPAY, LiqPay, WayForPay	
	Конкурентна ціна	14	LiqPay		UAPAY, WayForPay				

Згідно результатів порівняльного аналізу зроблений SWOT-аналіз, який наведений в таблиці 5.12.

Таблиця 5.12 – SWOT-аналіз стартап-проєкту

Сильні сторони: швидкість обробки платежів, безпека за рахунок використання ШІ для виявлення фроду	Слабкі сторони: відсутність інтеграцій з державними реєстрами, які надають інформацію про клієнта, за його діяльністю. Відсутність SDK
Можливості: співпраця з реєстрами та НБУ, для надання платіжної інформації, можливість розширення функціоналу системи, та залучення мобільного розроблення для цього	Загрози: великі конкуренти, які мають клієнтську базу, відсутність фінансування, нестабільність системи, залежність від партнерів

У результаті SWOT-аналізу розроблена альтернатива ринкової поведінки, яка наведена в таблиці 5.13.

Таблиця 5.13 – Альтернативи ринкового впровадження стартап-проєкту

№	Альтернатива ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1.	Розроблення SDK для інтеграції системи з платіжними сервісами клієнтів	Висока (можливе залучення грантів або інвесторів)	6-9 місяців
2.	Інтеграція з державними реєстрами для збору даних про клієнтів (через співпрацю з НБУ)	Середня (залежить від доступу до державних API)	9-12 місяців
3.	Запуск мобільного застосунку для бізнес-клієнтів з аналітичним модулем	Середня (потрібне фінансування для мобільної розробки)	6-8 місяців

№	Альтернатива ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
4.	Партнерство з банками та фінансовими організаціями для пілотного запуску продукту	Висока (існує зацікавленість банків у антифрод-рішеннях)	3-6 місяців
5.	Запуск маркетингової кампанії з фокусом на безпеку платежів і швидкість обробки	Середня (залежить від наявного бюджету на рекламу)	2-4 місяці
6.	Побудова відносин із потенційними інвесторами та венчурними фондами для отримання фінансування	Середня (інтерес до FinTech рішень є, але потрібні переконливі результати)	3-6 місяців
7.	Вивчення конкурентів та оновлення функціоналу відповідно до їхніх ключових переваг	Висока (можливо внутрішні ресурси достатні)	Постійний процес (періодично)
8.	Проведення технічного аудиту системи для підвищення стабільності та зменшення залежності від партнерів	Висока (можливе залучення власних інженерів)	2-3 місяці

## 5.4 Розроблення ринкової стратегії проєкту

Таблиця 5.14 – Вибір цільових груп споживачів

№	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
	Малі та середні підприємства, які використовують онлайн-платежі	Висока (зацікавленість у зниженні ризиків фроду)	Середній, але стабільний	Середня (конкуренція з локальними системами)	Висока (мінімальні бар'єри входу)
	Великі ритейлери з онлайн-продажами	Середня (залежить від ціни та функціональності)	Високий	Висока (домінують міжнародні платформи)	Середня (потрібні партнерства з банками)
	Фінансові компанії (небанківські установи, кредитні спілки)	Середня (залежить від технічної інтеграції)	Високий	Висока (велика кількість спеціалізованих рішень)	Середня (потрібен технічний ресурс)
	Стартапи у FinTech сфері	Висока (відкритість до нових технологій)	Можливий швидкий ріст	Низька (сегмент вузький)	Висока (спрощена співпраця в індустрії)

№	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
	Державні установи та компанії, що працюють за державними контрактами	Низька (потребують сертифікацій та перевірок)	Середній	Середня (конкуренція з великими гравцями)	Низька (бюрократичні бар'єри)

За результатами дослідження обрані:

- малі та середні підприємства, які використовують онлайн-платежі;
- великі ритейлери з онлайн-продажами.

Визначення базової стратегії розвитку наведене в таблиці 5.15.

Таблиця 5.15 – Визначення базової стратегії розвитку

№	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
1.	Орієнтація на малі та середні підприємства (SME), які використовують онлайн-платежі	Концентрований маркетинг	Доступна вартість послуг. Простота інтеграції. АІ-виявлення фроду, що мінімізує ризики	Стратегія спеціалізації: зосередження на потребах SME та забезпечення якісного обслуговування



№	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
2.	Орієнтація на великі ритейлери з онлайн-продажами	Диференційований маркетинг	Висока продуктивність системи. Гнучкість для адаптації до потреб ритейлерів. Надійна інфраструктура.	Стратегія диференціації: створення унікальних функцій для великих клієнтів, які підвищують їхню лояльність

Стратегія спеціалізації зосереджена на вузькому сегменті клієнтів із пропозицією максимально адаптованого продукту. Акцент зроблений на простоті інтеграції, швидкості впровадження, доступності послуг та зниженні витрат для клієнта.

Стратегія диференціації з основним акцентом на надання ритейлерам унікальних можливостей, таких як швидкість обробки великої кількості транзакцій та налаштування продукту під специфічні потреби великих клієнтів.

Далі вибрані стратегії конкурентної поведінки, які наведені в таблиці 5.16.

Таблиця 5.16 – Визначення базової стратегії конкурентної поведінки

№	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
1.	Проект є «першопрохідцем» на ринку для певного сегмента (наприклад, ШІ-аналіз платежів без 3DS, та виявлення фроду)	Переважно шукати нових споживачів, створюючи попит серед компаній, які ще не використовували такі сервіси	Ні, компанія створює власні унікальні рішення (ШІ-виявлення, простота інтеграції, адаптивність до ринкових умов)	Стратегія лідера: розширення первинного попиту через впровадження нових функцій та формування попиту на інноваційний продукт
2.	Проект не є «першопрохідцем», але має значний потенціал в окремих нішах ринку (наприклад, малий бізнес)	Фокус на залученні клієнтів конкурентів, пропонуючи вигідніші умови та унікальні функції	Так, частково копіюватиме характеристики конкурентів, але вдосконалюючи їх	Стратегія виклику лідера: флангова атака на зниження цін та поліпшення сервісу

У разі першопрохідця проекту доцільно акцентувати увагу на формуванні попиту, пропонуючи революційні рішення, які ще не присутні на ринку. Це дозволить швидко завоювати лояльність клієнтів і закріпити позиції.

Для сегментів, де конкуренція вже існує, ефективною буде флангова атака. Це дозволить використати слабкі сторони лідерів ринку, забезпечуючи конкурентоспроможність.

В таблиці 5.17 наведений результат розроблення стратегії позиціонування.

Таблиця 5.17 – Визначення стратегії позиціонування

№	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1.	Надійність і безпека (ефективне виявлення шахрайства), простота інтеграції, підтримка швидких транзакцій	Диверсифікація, створення унікальних продуктів для різних сегментів клієнтів	Унікальна AI-технологія для виявлення шахрайства без 3DS. Простота інтеграції API. Висока швидкість обробки транзакцій.	Безпека, інноваційність, зручність

№	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проєкту	Вибір асоціацій, які мають сформувати комплексну позицію власного проєкту (три ключових)
2.	Мінімальні витрати на інтеграцію, підтримка малих бізнесів, адаптація до специфіки локальних ринків	Концентрація на нішевих сегментах	Орієнтація на малий та середній бізнес. Ефективна підтримка локальних ринків. Доступна вартість інтеграції.	Доступність, локалізація, простота

### 5.5 Розроблення маркетингової програми стартап-проєкту

Далі сформована маркетингова концепція товару, який отримає клієнт (наведена в таблиці 5.18).

Таблиця 5.18 – Визначення ключових переваг концепції потенційного товару

№	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами
1.	Захист транзакцій від шахрайства	Підвищення рівня безпеки платежів за допомогою інноваційної ШІ-технології	Існуючі: передова технологія виявлення шахрайства, відсутність потреби у 3DS. Необхідні: розширення алгоритмів для аналізу специфіки локальних ринків

№	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами
2.	Проста інтеграція платіжних рішень у бізнес-процеси клієнтів	Мінімальні витрати часу та ресурсів на інтеграцію API	Існуючі: гнучкість і простота в налаштуванні. Необхідні: додаткові інструкції та інструменти для самостійної інтеграції
3.	Швидкість обробки транзакцій	Зменшення часу на виконання платежів і зниження ризику втрати клієнтів	Існуючі: високошвидкісна обробка даних. Необхідні: забезпечення масштабованості для великих обсягів даних
4.	Доступна вартість інтеграції та обслуговування	Економія коштів для малого та середнього бізнесу	Існуючі: конкурентні ціни. Необхідні: акції для нових клієнтів і програма лояльності
5.	Підтримка локальних особливостей бізнесу та регуляторних вимог	Можливість адаптації під конкретний ринок	Існуючі: модульний підхід до налаштувань. Необхідні: більше можливостей кастомізації та відповідність локальним регуляціям

Таблиця 5.19 – Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
I. Товар за задумом	Задоволення потреба у проведенні платежів, та відсутності необхідності проходження будь-яких сертифікацій, щоб мати змогу працювати з платежами		
II. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх/Тл/Е/Ор
	Економічні: низька вартість інтеграції Призначення: можливість проводити безпечні платежі для e-commerce продажів Ергономічні: зручний інтерфейс	-/+	+ / + / + / + / +
	Якість: постійне тестування, та зворотній зв’язок		
	Марка: FreePay		
III. Товар із підкріпленням	До продажу: маркетингова кампанія		
	Після продажу: підтримка та розвиток системи		
За рахунок ліцензії і реєстрації бренду			

Наступним кроком є визначення меж встановлення ціни на товар, який наведений в таблиці 5.20, оскільки це є основним прибутком компанії.

Таблиця 5.20 – Визначення меж встановлення ціни

№	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1.	Платежі через стандартні платіжні системи (без AI). Ціна за транзакцію – \$0.30 - \$1.00	Інші AI-рішення для запобігання шахрайства, які мають схожу функціональність. Ціна послуг - \$50 - \$200/місяць	Малий та середній бізнес з річним доходом до \$5 млн. середня заробітна плата працівників \$1,500 - \$4,000/місяць.	Нижня межа ціни: \$40/місяць (для малого бізнесу) Верхня межа ціни: \$150/місяць (для середнього бізнесу)

Далі в таблиці 5.21 є визначення оптимальної системи збуту, в межах якого приймається рішення.

Таблиця 5.21 – Формування системи збуту

№	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1.	Цільова аудиторія: Малі та середні підприємства, які активно використовують онлайн-платформи для бізнесу,	Функції постачальника: розробка та доставка програмного забезпечення, технічна	Глибина каналу: Канал прямий, без великих посередників, оскільки продукт є	Власна система збуту: Прямий збут через онлайн-платформи, прямі продажі через вебсайт,

№	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
	шукають прості та ефективні рішення для інтеграції у свої процеси. Зазвичай це довгострокові рішення на основі підписки.	підтримка, надання консультацій та навчання, оновлення продукту	спеціалізованим і потребує прямої взаємодії з кінцевими споживачами	підтримка через телефон чи чат. Альтернатива – залучення сторонніх консультантів для надання консультацій.

Останньою складовою маркетингової програми є розроблення концепції маркетингу, яка наведена в таблиці 5.22.

Таблиця 5.22 – Концепція маркетингових комунікацій

№	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1.	Цільова аудиторія: Малі підприємства, які шукають ефективні та економічні	Канали комунікацій: 1. Соціальні мережі (Facebook, Instagram)	Ключові позиції: 1. Простота та ефективність товару 2. Надійність	Завдання рекламного повідомлення: 1. Підкреслити основні вигоди	Концепція рекламного звернення: 1. Емоційний акцент: "Простота, яка



№	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
	рішення для автоматизації бізнес-процесів, зокрема через онлайн-інструменти. Вони схильні до використання сучасних технологій і шукають прості та доступні рішення	2. Електронна пошта 3. Онлайн-реклама (Google Ads, контекстна реклама) 4. Вебінари та онлайн-курси 5. Рекомендації партнерів і клієнтів (сарафанне радіо)	3. Своєчасність та доступність	використання продукту (збільшення продуктивності, зменшення витрат). 2. Надати чітку інформацію про переваги перед конкурентами (найкраща ціна, простота інтеграції).	допоможе зекономити час і гроші". 2. Інформативний акцент: "Лідери ринку обирають наше рішення для автоматизації бізнесу" – підкреслюючи надійність і технологічність продукту.

## Висновки до розділу 5

У даному розділі розроблена маркетингова програма для стартап-проекту, яка охоплює всі основні елементи стратегії просування продукту на ринок. На основі аналізу конкурентоспроможності, потреб споживачів та характеристик ринку сформована стратегія позиціонування та цінова політика, що забезпечують оптимальний баланс між конкурентоспроможністю і прибутковістю. Визначені ефективні канали збуту та комунікації, що сприяють залученню та утриманню цільової аудиторії. Розроблена програма дає змогу стартапу максимально адаптуватися до ринкових умов та забезпечити його успішне впровадження.

## ВИСНОВКИ

У ході виконання магістерської дисертації проведений комплексний аналіз існуючих рішень для еквайрингових платформ, зокрема LiqPay, UAPAY та WayForPay. Це дозволило визначити основні підходи й функціональні можливості, які забезпечують ефективну обробку платежів та мінімізацію ризиків для учасників фінансових операцій. Розгляд основних типів платежів, таких як p2p та e-commerce, дозволив вивчити їхні особливості, а також визначити можливості для інтеграції у різні бізнес-моделі.

Дослідження різновидів транзакцій для e-commerce, включаючи авторизаційні, преавторизаційні та рекурсивні платежі, підкреслило потребу в гнучкому та адаптивному підході до обробки транзакцій. Це, своєю чергою, обумовило вибір архітектури серверної частини, яка здатна ефективно масштабуватися й забезпечувати стабільну роботу платформи за різних умов навантаження.

У процесі дослідження також вивчені вимоги до платіжних систем, зокрема, щодо PCI DSS сертифікації, що є важливим для забезпечення безпеки даних користувачів. Це дозволило побудувати платформу, що відповідає міжнародним стандартам безпеки, а також адаптувати систему до вимог щодо захисту персональних даних.

Розроблена платформа побудована на основі мікросервісної архітектури, що забезпечує гнучкість і масштабованість, а також спрощує її подальший розвиток. Для забезпечення надійної взаємодії між компонентами системи використана шина повідомлень, що дозволяє ефективно обробляти великий обсяг транзакцій і даних. Платформа використовує сучасні технології, зокрема TypeScript та Python, що дозволяє збудувати потужну систему для обробки та аналізу платежів, а також для реалізації алгоритмів штучного інтелекту для виявлення фроду.

Завдяки використанню мікросервісної архітектури, система може бути легко адаптована для роботи з новими платіжними провайдерами та інтеграцією нових функціональних можливостей. Обрана реляційна база даних спроектована таким

чином, щоб забезпечити надійність і цілісність даних, а також масштабованість для роботи з великими обсягами транзакцій.

Розроблена еквайрингова платформа зі штучним інтелектом має значний потенціал для подальшого розвитку та масштабування. Вона здатна задовольнити потреби як малого, так і великого бізнесу, що працює в сфері електронної комерції, надаючи ефективні інструменти для боротьби з шахрайством і забезпечення безпеки транзакцій. Рішення, яке включає в себе новітні технології для виявлення фроду та високу надійність системи, дозволяє прогнозувати успіх цієї платформи на ринку України.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Про LiqPay. URL: <https://www.liqpay.ua/doc>
2. Про WayForPay. URL: <https://wiki.wayforpay.com/>
3. Про UAPAY. URL: <https://uapayua.atlassian.net/wiki/spaces/AC/pages/753795086>
4. Про E-commerce. URL: <https://interkassa.com/blog/shho-take-elektronna-komerciya-e-commerce-dlya-pochatkivciv>
5. Про LOOKUP. URL: <https://docs.nium.com/apis/docs/otp-based-3ds-authentication-flow>
6. Про 3DS. URL: [https://uk.wikipedia.org/wiki/3-D\\_Secure](https://uk.wikipedia.org/wiki/3-D_Secure)
7. Про PCI DSS сертифікацію. URL: <https://getpci.com/>
8. Порівняння PostgreSQL. URL: <https://www.dolthub.com/blog/2024-07-16-mysql-postgres-sysbench-latency/>
9. Порівняння MySQL. URL: <https://blog.pipeops.io/postgresql-and-mysql-which-is-better-a-2024-comparison/>
10. Бенчмарки баз даних. URL: <https://www.pingcap.com/article/mysql-vs-postgresql-a-complete-comparison-in-2024/>
11. Про використання mermaid. URL: <https://mermaid.js.org/>
12. Про TensorFlow. URL: <https://www.tensorflow.org/>
13. Про NodeJS. URL: <https://nodejs.org/en>
14. Про пакетні менеджери. URL: <https://romanglushach.medium.com/comparing-npm-yarn-and-pnpm-package-managers-which-one-is-right-for-your-distributed-project-to-4d7de2f0db8e>
15. Про Fastify фреймворк. URL: <https://www.npmjs.com/package/fastify#benchmarks>
16. Про ORM. URL: <https://www.npmjs.com/package/drizzle-orm>
17. Про роботу з RabbitMQ. URL: <https://www.npmjs.com/package/amqplib>

18. Про бібліотеку валідації. URL:  
<https://www.npmjs.com/package/zod>
19. Про JWT. URL: <https://www.npmjs.com/package/jsonwebtoken>
20. Про TypeScript. URL: <https://www.npmjs.com/package/typescript>
21. Про роботу з Redis. URL: <https://www.npmjs.com/package/ioredis>
22. Про бібліотеку для генерації звітів. URL:  
<https://www.npmjs.com/package/iconv>
23. Про бібліотеку логування. URL:  
<https://www.npmjs.com/package/pino>
24. Про змінні оточення. URL: <https://www.npmjs.com/package/dotenv>