

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ
СІКОРСЬКОГО»

Навчально-науковий інститут атомної та теплової енергетики

Кафедра цифрових технологій в енергетиці

"На правах рукопису"
УДК _____

«До захисту допущено»
Завідувач кафедри
Наталія АУШЕВА
“ ” _____ 2024 р.

Магістерська дисертація

на здобуття ступеня другого (магістерського) рівня вищої освіти
за освітньою програмою “Цифрові технології в енергетиці”
зі спеціальності 122 “Комп’ютерні науки”

на тему «Автоматизація конвертації банківських виписок на прикладі інформаційної системи ODOO»

Виконав: студент 2 курсу, групи ТР-31мп

Заступайло Максим Ігорович
(прізвище, ім’я, по батькові)

(підпис)

Науковий керівник професор кафедри ЦТЕ, д.т.н., доцент
Олексій ШУШУРА
(посада, науковий ступінь, вчене звання, ім’я ПРІЗВИЩЕ)

(підпис)

Рецензент професор кафедри ІПЗЕ, д.т.н, професор Євген ГАВРИЛКО
(посада, науковий ступінь, вчене звання, ім’я ПРІЗВИЩЕ)

(підпис)

Н.контроль старший викладач кафедри ЦТЕ Ольга БЕСПАЛА
(посада, ім’я ПРІЗВИЩЕ)

(підпис)

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць
інших авторів без відповідних
посилань.
Студент _____
(підпис)

4) обґрунтувати вибір засобів розробки системи автоматизації конвертації

5) розробити програмне забезпечення, провести тестування, здійснити опис програмної реалізації

6) описати послідовність роботи користувача із системою

5. Орієнтовний перелік ілюстративного матеріалу: діаграма взаємодії складових, модель PaymentConversion, діаграма варіантів діяльності користувача, діаграма варіантів діяльності адміністратора, файлова структура серверного модуля, клас взаємодії з базою даних, клас Transaction, таблиця tokentransactions, сценарій роботи та взаємодії клієнта із системою, алгоритм дій користувача з установкою та використанням системи

6. Дата видачі завдання «15» вересня 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів роботи	Примітка
1	Вибір предметної області магістерської дисертації та напрямку роботи	22.10.2023	виконано
2	Вивчення та аналіз завдання, розгляд та аналіз документації, літературних джерел, сучасних систем конвертації	28.02.2024	виконано
3	Визначення структури магістерської дисертації, вибір засобів розробки	03.09.2024	виконано
4	Постановка вимог до системи автоматизації конвертації та розробка її архітектури	20.09.2024	виконано
5	Програмна реалізація системи	18.10.2024	виконано
6.	Захист розробленого програмного забезпечення	23.10.2024	виконано
7	Оформлення магістерської дисертації	06.12.2024	виконано

Студент

(підпис)

Максим ЗАСТУПАЙЛО

(ім'я, ПРІЗВИЩЕ)

Керівник роботи

(підпис)

Олексій ШУШУРА

(ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

Актуальність теми. Для успішного розвитку сучасних підприємств та для забезпечення їхньої конкурентоздатності необхідністю є автоматизація обробки фінансових даних у сучасних компаніях. Забезпечення швидкого та точного імпорту платіжних виписок з банківських систем в ODOO є важливим етапом для забезпечення актуальної та достовірної фінансової інформації. Тому необхідною є розробка, яка надавала б змогу клієнту завантажувати банківські виписки з різних банків та в різних форматах, а сервіс в свою чергу повертав би конвертований файл в універсальному форматі. А особливо важливим і потрібним є такий функціонал, що дозволить спростити підтримку змін у форматах виписок банків для інформаційної системи.

Мета роботи – розробка автоматизованої системи конвертації банківських виписок на основі інформаційної системи ODOO.

Завдання дослідження:

- проаналізувати існуючі методи обміну даними з банками, що застосовуються в сучасних інформаційних системах;
- розробити модуль конвертації банківських виписок на прикладі інформаційної системи ODOO;
- проаналізувати та розробити механізм взаємодії з ODOO;
- забезпечити можливість аналізувати використання модуля конвертації та розробити механізм обмеження доступу;
- розробити користувацький та адміністративний модулі на прикладі системи ODOO для аналізу інтеграції та здійснення автоматичної конвертації;
- протестувати розроблену систему автоматизації конвертації.

Об'єкт дослідження – процес автоматизації конвертації банківських виписок.

Предмет дослідження – це алгоритми реалізації та функціонування системи автоматизації конвертації, її структура та програмне втілення.

Методи та засоби: прийоми порівняльного аналізу, проектування, прототипування, ERP-система ODOO, СУБД PostgreSQL, мова програмування Python, фреймворк Flask, середовище розробки PyCharm.

Результат – архітектура й програмне забезпечення системи автоматизації конвертації банківських виписок на прикладі інформаційної системи ODOO.

Практична цінність отриманих в роботі результатів полягає в наданні можливості забезпечити конвертацію платіжних виписок в форматі, зрозумілому для системи, та спрощенні врахування змін у форматах виписок.

Апробація результатів дисертації. Окремі аспекти дослідження було висвітлено в рамках XXI Міжнародної науково-практичної конференції молодих вчених і студентів «Сучасні проблеми наукового забезпечення енергетики» та оприлюднено в матеріалах до неї в тезах доповіді «Автоматизація конвертації банківських виписок для інформаційних систем підприємств».

Дисертація складається зі вступу, шести розділів та висновків. Загальний обсяг роботи – 107 сторінок, у тому числі 85 сторінок основного тексту, 24 таблиці, 19 супроводжуючих рисунків, 4 сторінки списку використаних джерел у кількості 40 найменувань.

Ключові слова: *система автоматизації конвертації, банківські виписки, інформаційна система ODOO, модуль Payment Conversion, модуль Conversion Control Panel.*

ABSTRACT

Relevance. For the successful development of modern enterprises and to ensure their competitiveness, it is necessary to automate the processing of financial data in modern companies. Ensuring fast and accurate import of payment statements from bank systems into ODOO is an important step in providing up-to-date and reliable financial information. Therefore, it is necessary to develop a solution that would allow the client to upload bank statements from different banks and in different formats, and the service would return the converted file in a universal format. And especially important and necessary is such functionality that will simplify the support of changes in the formats of bank statements for the information system.

The aim of the work is to develop an automated system for converting bank statements based on the ODOO information system.

Objectives of the study:

- to analyze the existing methods of data exchange with banks used in modern information systems;
- to develop a module for converting bank statements on the example of the ODOO information system;
- to analyze and develop a mechanism for interaction with ODOO;
- to provide the ability to analyze the use of the conversion module and develop a mechanism for restricting access;
- to develop user and administrative modules on the example of the ODOO system to analyze integration and perform automatic conversion;
- to test the developed conversion automation system.

The object of research is the process of automating the conversion of bank statements.

The subject of research is the algorithms for the implementation and functioning of the conversion automation system, its structure and software implementation.

Methods and tools: methods of comparative analysis, design, prototyping, ODOO

ERP system, PostgreSQL database, Python programming language, Flask framework, PyCharm development environment.

The result is the architecture and software of an automation system for converting bank statements using the ODOO information system as an example.

The practical value of the results obtained in this study is to provide an opportunity to convert payment statements in a format understandable to the system and to simplify the accounting of changes in statement formats.

Publication: Some aspects of the study were highlighted within the framework of the XXI International Scientific and Practical Conference of Young Scientists and Students “Modern Problems of Scientific Support of Energy” and published in the materials for it in the abstracts of the report “Automation of Conversion of Bank Statements for Enterprise Information Systems”.

The dissertation consists of an introduction, six chapters and conclusions. The total volume of the work is 107 pages, including 85 pages of the main text, 24 tables, 19 accompanying figures, 4 pages of the list of references consisting of 40 titles.

Keywords: *conversion automation system, bank statements, ODOO information system, Payment Conversion module, Conversion Control Panel module.*

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ ТА ТЕРМІНІВ	10
ВСТУП.....	13
1 ПОСТАНОВКА ЗАДАЧІ РОЗРОБКИ СИСТЕМИ АВТОМАТИЗАЦІЇ КОНВЕРТАЦІЇ БАНКІВСЬКИХ ВИПИСОК НА ПРИКЛАДІ ІНФОРМАЦІЙНОЇ СИСТЕМИ ODOO.....	18
1.1 Вимоги до системи автоматизації конвертації банківських виписок та її задачі	18
1.2 Методи та прийоми реалізації	20
Висновки до розділу 1.....	23
2 ДОСЛІДЖЕННЯ СУЧАСНИХ СИСТЕМ КОНВЕРТАЦІЇ БАНКІВСЬКИХ ВИПИСОК.....	25
2.1 Огляд основних підходів до обміну даними між банками та інформаційними системами підприємств	26
2.2 Аналіз ресурсів, розробок та систем, що існують на сучасному етапі	29
Висновки до розділу 2.....	30
3 ВИБІР ЗАСОБІВ РОЗРОБКИ СИСТЕМИ АВТОМАТИЗАЦІЇ КОНВЕРТАЦІЇ БАНКІВСЬКИХ ВИПИСОК НА ПРИКЛАДІ ІНФОРМАЦІЙНОЇ СИСТЕМИ ODOO.....	32
3.1 Інформаційна система ODOO	32
3.2 База даних PostgreSQL	34
3.3 Мова програмування Python	36
3.4 Фреймворк Flask.....	37
3.5 Середовище розробки PyCharm.....	38

Висновки до розділу 3.....	40
4 ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ СИСТЕМИ АВТОМАТИЗАЦІЇ КОНВЕРТАЦІЇ БАНКІВСЬКИХ ВИПИСОК НА ПРИКЛАДІ ІНФОРМАЦІЙНОЇ СИСТЕМИ ODOO.....	42
4.1 Проєктування архітектури системи автоматизації	42
4.2 Внутрішня структура та засоби реалізації модуля конвертації	44
4.3 Структура та особливості реалізації клієнтського модуля	47
4.4 Структура та особливості реалізації адміністративного модуля	51
Висновки до розділу 4.....	53
5 РОБОТА КОРИСТУВАЧА З СИСТЕМОЮ АВТОМАТИЗАЦІЇ КОНВЕРТАЦІЇ.....	55
5.1 Рекомендовані системні вимоги	55
5.2 Послідовність встановлення системи користувачем.....	56
5.3 Використання системи автоматизації конвертації банківських виписок клієнтом	59
5.4 Використання системи автоматизації конвертації банківських виписок адміністратором.....	63
Висновки до розділу 5.....	64
6 РОЗРОБЛЕННЯ СТАРТАП-ПРОЄКТУ	66
6.1 Опис ідеї стартап-проєкту	66
6.2 Технологічний аудит ідеї стартап-проєкту	68
6.3 Здійснення аналізу ринкових можливостей запуску стартап-проєкту.....	69
6.4 Розроблення ринкової стратегії стартап-проєкту	76
6.5 Розроблення маркетингової програми стартап-проєкту.....	79
Висновки до розділу 6.....	82

ВИСНОВКИ	84
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	86
ДОДАТОК А	90
ДОДАТОК Б	102

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ ТА ТЕРМІНІВ

ODOO – раніше OpenERP – відкрита багатомодульна платформа управління бізнес-процесами та підтримки підприємств.

JSON – Java Script Object Notation – формат обміну даними.

API – Application Programming Interface – набір правил і протоколів, які дозволяють різним програмним системам взаємодіяти між собою.

ERP – Enterprise Resource Planning – система планування ресурсів підприємства.

ISO 20022 – міжнародний стандарт для електронного обміну даних між фінансовими установами.

FIX – Financial Information Exchange – стандартах обміну даними.

OCA – Odoo Community Association – некомерційна організація, яка сприяє розвитку та підтримці відкритої спільноти ODOO.

Xero – хмарна облікова система.

Bank2XML – програмне забезпечення, яке дозволяє конвертувати банківські виписки в формат XML.

BizConnect – хмарна облікова система.

OFX – Open Financial Exchange – відкритий стандарт обміну фінансовою інформацією між фінансовими установами, програмними додатками і системами.

PostgreSQL – потужна об'єктно-реляційна система керування базами даних (ОРСБД), яка пропонує розширені можливості для зберігання та обробки даних.

СУБД – СКБД – система управління базами даних або система керування базами даних. Це програмне забезпечення, яке дозволяє організовувати, зберігати, керувати та взаємодіяти з базами даних.

pgAdmin – графічний інтерфейс користувача (GUI) для керування та адміністрування базами даних PostgreSQL.

Фреймворк – англ. Framework– це набір засобів, бібліотек, правил та шаблонів, які використовуються для розробки програмного забезпечення.

Flask – веб-фреймворк для мови програмування Python.

IDE – Integrated Development Environment – інтегроване середовище розробки, програмний інструмент, який надає розробникам усі необхідні засоби для розробки, тестування та налагодження програмного забезпечення в одній оболонці.

PyCharm – інтегроване середовище розробки (IDE) для мови програмування Python.

Кастомний модуль – налаштований модуль – це модуль, створений або модифікований користувачем для виконання конкретних завдань або функціональних вимог.

HTTP – HypertextTransfer Protocol – протокол передачі гіпертексту.

Токен – строковий або числовий ідентифікатор, який використовується для представлення і перевірки автентичності або доступу до ресурсів.

URL – UniformResourceLocator – уніфікований локатор ресурсу, що є стандартизованим форматом адреси, який використовується для ідентифікації та доступу до ресурсів в Інтернеті.

ВСТУП

Обмін даними з банківськими інформаційними системами є важливою складовою діяльності багатьох підприємств, компаній та організацій. Він сприяє автоматизації бухгалтерських і фінансових процесів, скорочує час, який витрачається на ручне введення даних, а також знижує можливість помилок під час обробки фінансових даних.

Інформаційна система ODOO є одним із широко використовуваних інструментів для автоматизації бізнес-процесів і управління компанією. Проте, щоб ефективно працювати з банківськими інформаційними системами, необхідною є розробка спеціалізованого модуля для автоматизації обміну даними, який забезпечить інтеграцію між ODOO і банківськими системами [1].

Сучасні банківські інформаційні системи швидко розвиваються і стають дедалі складнішими, що створює нові виклики для їх інтеграції з іншими програмними продуктами. Для забезпечення ефективної роботи часто потрібні спеціалізовані інструменти, які здатні адаптуватися до особливостей кожної системи. Багато банківських систем використовують власні протоколи для обміну даними, що ускладнює їхнє поєднання з іншими платформами. Це означає, що для повноцінної інтеграції потрібні гнучкі рішення, які можуть адаптуватися до різних стандартів та забезпечувати безперебійний зв'язок між різними інформаційними системами.

Важливим аспектом фінансового управління є обробка банківських виписок, які містять дані про здійснені підприємством платежі та фінансові транзакції.

Таким чином, створення системи для конвертації банківських платіжних виписок для інформаційної системи ODOO не тільки дозволить полегшити роботу підприємств, але й сприяє підвищенню їхньої конкурентоспроможності на ринку. Це рішення дозволяє оптимізувати фінансові процеси і забезпечити більш ефективне управління ресурсами.

Актуальність роботи обумовлена необхідністю автоматизації обробки

фінансових даних у сучасних компаніях. Сьогодні багато підприємств застосовують інформаційну систему ODOO для управління різними бізнес-процесами, зокрема фінансовими операціями. Швидкий і точний імпорт платіжних виписок з банківських систем у ODOO є ключовим кроком для забезпечення актуальності та достовірності фінансової інформації, що важливо для прийняття обґрунтованих управлінських рішень.

Дана робота вирішує це завдяки розробці системи автоматизації конвертації банківських виписок на прикладі інформаційної системи ODOO. Такий програмний продукт здатний імпортувати платіжні виписки з різних банків та їх форматів у систему ODOO з легкістю. Це допоможе зменшити ручну роботу та підвищити ефективність обробки фінансової інформації. Модуль надає змогу клієнту завантажувати банківські виписки з різних банків та в різних форматах, а сервіс в свою чергу повертає конвертований файл в універсальному форматі, з яким далі уже можна працювати в самій інформаційній системі.

Застосування модуля дозволить спростити процес автоматизації обробки платіжних виписок, що мінімізуватиме ймовірність помилок та збільшить швидкість доступу до фінансових даних. Важливим аспектом актуальності роботи є те, що ODOO є популярною відкритою інформаційною системою для управління різними бізнес-процесами. Інтеграція системи автоматизації конвертації платіжних виписок в ODOO розширить її можливості та зробить її більш привабливою для компаній, які використовують систему ODOO для фінансового управління.

Показником необхідності розробки подібної системи були регулярні затягування процесів фінансових операцій в економічній сфері, а насамперед в банківській, через те, що потрібними були окремі налаштування для кожної платформи, кожного випадку тощо. Це все займало багато часу і вимагало значних фінансових вкладів. Тож розроблений сервіс є прекрасним рішенням, адже надає універсальність, що значно економитиме час та гроші.

Спрощеним є також механізм врахування змін. Різними системами використовуються різні банки. За умови необхідності у зміні банком чогось у

форматі чи за умови підключення нового банку, це все потребує вирішення в системі ODOO. Це не створює зручності, оскільки при оновленні системи кожен раз є ймовірність виникнення різних нюансів з налаштуваннями користувачів. Однак система автоматизації конвертації здатна вирішити ці проблеми. Якщо появляються зміни в платіжних банківських виписках, розробник просто додає або змінює обробку банківського файлу на сервісі конвертації – і в клієнта вона вже працює: не потрібними стають окремі налаштування і роз'яснення для кожного користувача.

Наприклад, якщо в клієнта змінився список банків, з якими він співпрацює, чи у банку змінився формат файлу, то у випадку попередніх рішень розробникам необхідно було звертатися до кожного клієнта індивідуально і надсилати або замінювати файли. А з точки зору отриманого програмного продукту все інакше: якщо щось змінилося, і хтось з клієнтів звернувся з цією проблемою, то виправлення її в сервісі автоматично виправляє цю проблему у всіх клієнтів. До того ж, якщо співпраця з банками налагоджена, то й така ймовірність зникає, тому що інформацію щодо змін можна отримати саме від банків, що сприятиме підготовці оновлень до змін, які впроваджені банком. Таким чином, клієнти не помічатимуть змін, адже для них все функціонуватиме у звичному форматі.

Таким чином, розробка системи має на меті спростити та автоматизувати процес обробки платіжних виписок у системі ODOO шляхом розробки системи автоматизації конвертації. Це не тільки допоможе підприємствам з ефективним управлінням фінансовою інформацією, але й підвищить точність, зручність та швидкість обробки цих даних.

Щодо розробок у цій сфері загалом, а саме в галузі конвертування банківських виписок, то варто зауважити, що вони швидко розвиваються, особливо в контексті реалізації відкритого банківського середовища та цифрового трансформування фінансового сектору. Багато розробників працюють над розширенням функціональності, покращенням безпеки та стандартизацією обміну даними.

Більша частина розробок, що функціонують на сучасному етапі, хоч і мають значну кількість переваг, проте мають і багато недоліків, які необхідно враховувати [2]. Серед найрозповсюдженіших мінусів тих систем, що існують на сучасному етапі, варто виокремити складнощі з інтегруванням, значні витрати, різноманітність стандартів тощо. Усе це стає причиною виникнення перешкод при взаємодії з банками чи системами, створює складнощі та ризики.

Мета роботи полягає у розробці автоматизованої системи конвертації банківських виписок на основі інформаційної системи ODOO. Відповідно, основним **завданням** є підвищити ефективність обміну даними з банками та знизити ймовірну кількість помилок завдяки автоматизації конвертації виписок.

Досягнення мети передбачає вирішення таких **завдань**:

- проаналізувати існуючі методи обміну даними з банками, що застосовуються в сучасних інформаційних системах;
- розробити модуль конвертації банківських виписок на прикладі інформаційної системи ODOO;
- проаналізувати та розробити механізм взаємодії з ODOO;
- забезпечити можливість аналізувати використання модуля конвертації та розробити механізм обмеження доступу;
- розробити користувацький та адміністративний модулі на прикладі системи ODOO для аналізу інтеграції та здійснення автоматичної конвертації;
- протестувати розроблену систему автоматизації конвертації.

Об’єктом дослідження є процес автоматизації конвертації банківських виписок. **Предмет** дослідження – це алгоритми реалізації та функціонування системи автоматизації конвертації, її структура та програмне втілення.

Апробація результатів. Окремі аспекти дослідження було висвітлено в рамках XXI Міжнародної науково-практичної конференції молодих вчених і студентів «Сучасні проблеми наукового забезпечення енергетики» та оприлюднено в матеріалах до неї в тезах доповіді «Автоматизація конвертації банківських виписок для інформаційних систем підприємств» [4].

Щодо структури та обсягу роботи, то до складу магістерської роботи входять технічне завдання, перелік скорочень, умовних позначень і термінів, які зустрічаються в роботі, анотації українською й англійською мовами, зміст роботи, вступ, шість розділів, висновки, список використаної літератури (40 позицій) та два додатки. Загальний обсяг роботи – 107 сторінок, а основного тексту (до списку джерел, використаних під час роботи) – 85 сторінок. Щодо ілюстративного матеріалу, то в тексті магістерської є 19 супроводжуючих рисунків та 24 таблиці. У додатках відображено програмний код серверного модуля (11 сторінок) та стаття-апробація магістерської роботи (5 сторінок).

1 ПОСТАНОВКА ЗАДАЧІ РОЗРОБКИ СИСТЕМИ АВТОМАТИЗАЦІЇ КОНВЕРТАЦІЇ БАНКІВСЬКИХ ВИПИСОК НА ПРИКЛАДІ ІНФОРМАЦІЙНОЇ СИСТЕМИ ODOO

Кінцевою метою роботи є створення системи автоматизації конвертації банківських виписок в форматі, зручної для подальшої інтеграції в інформаційну систему ODOO, та забезпечення безперебійного обміну даними між фінансовими системами.

Створений програмний продукт вирішує поставлені перед ним цілі завдяки вирішенню комплексу задач. До системи автоматизації конвертації загалом, а зокрема саме до модуля конвертації поставлено конкретні функціональні та не функціональні вимоги, які мають бути реалізовані. Завдяки комплексному підходу та використанню необхідних методів та прийомів досягнення мети та розв'язання поставлених завдань є можливим.

1.1 Вимоги до системи автоматизації конвертації банківських виписок та її задачі

Для ефективності роботи із сучасними банківськими інформаційними системами, які стають все складнішими, необхідними є нові інструменти. Деякими банківськими системами використовуються спеціалізовані протоколи обміну даними, які забезпечують постійну сумісність з іншими системами.

Завданням магістерської роботи було розробити систему автоматизації конвертації банківських виписок на прикладі ODOO, яка складатиметься з трьох функціональних блоків – модуля конвертації, клієнтського та адміністративного

модулів – та дозволятиме клієнтам завантажувати файли виписок з підбором необхідних параметрів (обрання необхідного банку, формату файлу тощо), а адміністраторам дозволятиме контролювати процес максимально лаконічно та оперативно. Програмний продукт має можливість перевіряти усі доступні формати для того банку, що обирається клієнтом, та здійснювати обробку файлу, обираючи правила обробки відповідно до вказаних параметрів.

Провідні функціональні та нефункціональні вимоги до системи автоматизації конвертації банківських виписок та її задачі відтворено в таблиці 1.1.

Таблиця 1.1 – Основні вимоги до системи автоматизації конвертації та її задачі

№ п/п	Вимоги та задачі	
	Функціональні	Нефункціональні
01	Здійснення забезпечення можливості завантажувати файли з банківськими виписками	Забезпечення можливостей легкого розширення сервісу для додавання нових правил обробки файлів
02	Підтримка системою вибору банків та форматів файлів для обробки	Створення інтуїтивного та зручного інтерфейсу для користувачів
03	Забезпечення перевірки доступності конвертації та їхньої кількості для користувача	
04	Обробка файлів відповідно до вимог обраного банку та формату згідно з правилами обробки	
05	Збереження конвертованих файлів в універсальному форматі json	
06	Забезпечення можливості керування адміністратором кількістю доступних конвертацій по клієнту (додавання, віднімання, очищення) та перегляду історії по кожному користувачу	

Завдяки тому, що процес інтеграції банківських виписок буде спрощено, можливими стануть легка та швидка інтеграція банківських виписок клієнтом в

його інформаційну систему ODOO. Зникає необхідність розроблювати власне рішення для конвертування та обробки виписок, системою забезпечується автоматизованість і стандартизованість процесу опрацювання виписок. Автоматизація конвертації файлів відповідно до обраного формату дозволяє системі надавати змогу користувачам обирати необхідний банк та формат файлу, після чого відбувається автоматична конвертація виписок в обраний формат, що підходить подальшій обробці в інформаційній системі.

Для контролю використання обмежених ресурсів та для запобігання неправомірного використання системи необхідною є перевірка доступності конвертацій та обмежень, яка сприятиме тому, щоб перевіряти доступність клієнта та кількість доступних конвертацій перед початком обробки виписки.

Система повертає конвертовану виписку в універсальному форматі виведення – JSON, оскільки саме такий формат відзначається зручністю та універсальністю для наступної обробки даних в інформаційній системі ODOO; відповідно, такий підхід робить можливим забезпечення стандартизації та сумісності даних. Це полегшує спрощення майбутнього аналізу та звітності.

Відповідно до вимог, забезпечення розширюваності системи та її гнучкості робить програмний продукт таким, що може бути легко інтегрованим не лише в ODOO, а й в інші системи. Це стає можливим завдяки розробці його мовою програмування Python та використанню бібліотек. Згідно з вимогами клієнтів можна додавати правила обробки файлів, що забезпечуватиме розширення та гнучкість системи.

1.2 Методи та прийоми реалізації

Задля реалізації мети та завдань роботи було використано різноманітні методи та прийоми.

Під час розробки системи автоматизації конвертації банківських виписок для

інформаційної системи ODOO використано різноманітні методи аналізу, дослідження та практичної реалізації.

Першочергово було досліджено літературу, яка відповідає темі та яка допомогла комплексно поглянути на можливості реалізації мети, розглянути попередні розробки, надбання напряму тощо. Було обрано джерела з інформацією, що дала змогу поглибити розуміння проблематики теми, вивчити особливості обміну даними з банками, розглянути стандарти, підходи і протоколи, які використовують спеціалісти в цій галузі. Це сприяло визначенню найкращих практик та розробці фундаментального усвідомлення теми.

Після цього було проведено дослідження розроблених схем зі схожим або дотичним функціоналом [3]. Увагу приділяли розробкам, які на сучасному етапі використовуються клієнтами. Важливим стало вивчення їхньої документації, специфіки функціонування, можливостей інтеграції до систем компаній та співпраці з ними [4]. Опісля було здійснено систематизацію отриманих даних, завдяки чому виокремлено плюси та мінуси в роботі існуючих програмних продуктів [5].

Важливим етапом стало дослідження потреб ймовірних користувачів. Завдяки цьому було підсумовано ті очікування, які клієнти можуть мати щодо програмного продукту, відповідно, виокремлено як функціональні вимоги до системи, так і нефункціональні. Відповідно до цього було розроблено орієнтир основних ймовірних функцій системи автоматизації, її можливостей.

Наступним кроком стало проєктування вже власне самої системи автоматизації, її розробки, перевірки функціональних можливостей, продуктивності, а також надійності. Для цього було застосовано комплекс підходів та прийомів.

Для оцінки переваг та недоліків отриманого програмного продукту, визначення його унікальності, конкурентної спроможності тощо було проведено порівняльний аналіз системи у співвідносності з аналогами і розробками, що вже існують на сучасному етапі. Це дало змогу оцінити також рентабельність продукту

на фінансовому ринку, його потенціал в реалізації на поприщі фінансів та бізнесу [6]. Після первинного аналізу було внесено необхідні зміни в специфіку роботи, що дало змогу покращити реалізацію функцій і можливостей системи автоматизації конвертації. Це було здійснено для покращення конкурентоспроможності програмного продукту та для того, щоб отримана система змогла вирішувати не тільки ті питання, які є доступними для схожих аналогів, але й ті, які є недоступними для них.

При власне самій розробці програмного продукту базовою можна виокремити методику ітеративної моделі. При всій її подібності до інкрементних або каскадних моделей, саме вона сприяє можливості поступового покращення продукту з можливостями не тільки покращувати наявні компоненти, але й додавати нові.

Тестування системи конвертації здійснювалося паралельно з її реалізацією, відповідно було використано фрагменти V-подібної моделі, що дозволило оцінювати зв'язок компонентів та їхню функціональність.

Для практичного втілення системи автоматизації конвертації важливу роль зіграли експертні оцінки. Під час тестування результатів створеної системи та її перевірки було запрошено для оцінки експертів у сфері обміну даними чи розробок програмних продуктів. Особливу увагу було приділено саме тим експертам, які безпосередньо працюють з інформаційною системою ODOO, знають її особливості, орієнтування та основний ринок користувачів. Отримано їхні фахові оцінки щодо якості, ефективності та потенціалу системи. Для кращої оцінки було використано підхід прототипування. Це сприяло отриманню реального зворотного зв'язку від можливих клієнтів, що допомогло внесенню коректив до продукту перед кінцевою реалізацією.

Після реалізації було зібрано дані щодо роботи системи автоматизації конвертації банківських виписок та проведено їхній аналіз. Оцінено ефективність, продуктивність та відповідність тим вимогам, які клієнти ставлять перед продуктом. Ці кроки було здійснено завдяки використанню методів оцінки

результату та кількісно-статистичного аналізу.

Комплексний підхід до обрання методів як дослідження теми, так і самих методів розробки системи автоматизації конвертації був орієнтований на найпродуктивніший шлях реалізації програмного продукту як готової системи, так і подальшої його реалізації на фінансовому поприщі.

Висновки до розділу 1

Практичним завданням роботи є створення системи автоматизації конвертації банківських виписок на прикладі інформаційної системи ODOO та забезпечення стабільності в обміні даними між системами. Таким чином, система автоматизації конвертації банківських виписок на прикладі інформаційної системи ODOO вирішує завдання полегшення інтеграції виписок, автоматизації конвертації форматів, контролю доступу та обмежень, генерації універсального формату виведення та можливості розширення для користувачів, гнучкості тощо.

Реалізація мети розвідки стала можливою завдяки тому, що було комплексно використано різні прийоми і методи, які направлено на втілення максимально можливого найкращого та найоптимальнішого результату, що є можливим за умов роботи та при її специфіці.

Протягом роботи над системою автоматизація конвертації було здійснено декілька етапів втілення та реалізації. Для того, щоб краще усвідомлювати проблематику обміну даними, їх конвертацію першочергово було проаналізовано теоретичну літературу за темою, а також наукові джерела, які висвітлюють саме цю галузь. З тією ж метою було досліджено стандарти, які реалізуються в цій сфері, а також підходи і протоколи. Для більшого розуміння потреб користувачів далі було проаналізовано сучасні аналогічні або схожі системи. Це дало змогу виявити як їхні переваги, на які можна було орієнтуватися, та їхні недоліки, вирішити які стало необхідним. Уже під час реалізації програмного продукту було задіяно

методології, які максимально були прорахованими під виробництво (ітеративна модель, V-подібна модель тощо). Надано велике значення думкам експертів, які працюють в сфері обміну даними, оскільки вони мають набагато більший досвід у розумінні цієї галузі, мають практичні уявлення про специфіку роботи та потреби користувачів. Протягом переддипломної практики, під час якої було поглиблено розуміння теми, завершено проєктування та розробку системи автоматизації і проаналізовано її продуктивність, надійність та функціональні можливості було створено прототип системи, що дало змогу внести необхідні корективи до програмного продукту, перш ніж вважати його реалізацію як остаточну. Завдяки цьому отримано уявлення про перспективи розвитку напрямку загалом, а зокрема і про можливості подальшого розвитку сервісу, його ймовірне майбутнє.

2 ДОСЛІДЖЕННЯ СУЧАСНИХ СИСТЕМ КОНВЕРТАЦІЇ БАНКІВСЬКИХ ВИПИСОК

У сучасному світі важливість систем для конвертації банківських виписок важко переоцінити, оскільки вони мають критично важливу роль у діяльності багатьох компаній, а особливо тих, які здійснюють фінансові операції та співпрацюють з банками.

Однією з ключових причин створення і використання таких систем є необхідність автоматизувати фінансові процеси. Такі сервіси дають змогу компаніям налагоджувати автоматичний обмін інформацією з банками, що значно знижує необхідність ручного введення даних і зменшує обсяг рутинної роботи. При цьому підвищується продуктивність і зменшується ризик помилок. До того ж, автоматизація позитивно впливає на загальну стабільність компанії, оскільки дозволяє швидше реагувати на зміни у фінансовому стані підприємства.

Позитивна сторона таких систем ще й в тому, що вони гарантують точність та цілісність даних. У контексті фінансової стабільності будь-якої компанії важливим є забезпечення надійності та безперебійності обміну даними з банками. Завдяки інтеграції подібних систем з інформаційними системами підприємств, зокрема і з ODOO, можна досягнути високого рівня безпеки та швидкості обміну даними. Це дозволяє уникати втрат інформації, зменшувати ймовірність помилок та забезпечувати актуальність фінансових даних, що є надзвичайно важливим для прийняття обґрунтованих управлінських рішень [1].

Популярність систем автоматизації конвертації невпинно зростає через стрімкий розвиток технологій та цифровізацію банківської сфери. Сучасні банки все частіше пропонують своїм користувачам доступ до онлайн-платформ та API, які полегшують обмін даними. Для компаній інтеграція з цими сервісами тає значеннєвим елементом підтримки конкурентоспроможності, адже вона сприяє забезпеченню автоматизованого доступу до фінансової інформації, допомагає

оперативному здійсненню платежів, відслідковує транзакції та проводить глибокий аналіз фінансової діяльності. Це допомагає швидко отримувати точну інформацію про фінансовий стан компанії, а це є ключовим для її гнучкості та ефективності у веденні бізнесу.

Окрім того, системи конвертації, які інтегруються з платформою ODOO, значно спрощують процес управління фінансами. Завдяки ним можна автоматично завантажувати фінансові дані безпосередньо в саму систему. Це сприяє ефективності аналізу фінансових показників, допомагає формувати звітність та планувати бюджет. Завдяки цьому керівництво підприємств отримує можливість швидко приймати зважені рішення, опираючись на перевірену, точну та актуальну фінансову інформацію.

2.1 Огляд основних підходів до обміну даними між банками та інформаційними системами підприємств

Наукові праці та практичні дослідження і розвідки, присвячені тематиці автоматизації обробки банківських виписок та інтеграції фінансових систем із платформами, що є подібними до ODOO, є досить популярними. Проте найчастіше вони є фрагментальними або вузькоорієнтованими тільки на окремі віхи напрямку. Конкретних досліджень, які б уже повноцінно розкривали специфіку теми на сучасному етапі, досить небагато, окрім того, дослідження, які безпосередньо стосуються цього напрямку і розкривають тему, є зазвичай обмеженими.

Переважно дослідження концентрують увагу на загальних підходах обміну інформацією та даними між системами підприємств (зокрема, на електронних системах управління компаніями – Enterprise Resource Planning – ERP) [7]) та банками. І хоч увагу приділяють темі оптимізації бізнес-процесів, та дуже часто в таких дослідженнях відсутня стандартизація, що ускладнює масштабування та впровадження на практиці. Проте варто відмітити, що деякі з таких праць можуть

бути адаптованими до контексту обміну даними з інформаційною системою ODOO. І це можливо й у напрямку конвертації файлів та автоматизації фінансових процесів. Деякі дослідження пропонують варіанти математичних моделей та алгоритмів для автоматизації аналізу даних, які надходять з банківських виписок (аналіз транзакцій, автоматичне розпізнавання типів доходів або витрат, прогнозування фінансових потоків).

Статті в наукових журналах та на спеціалізованих сайтах, а також дослідження в галузі фінансової інформатики, інтеграції з банками мають деякі інформаційні дані про деталі обміну фінансовими даними з банківськими установами. Зокрема, окремі з них зосереджені на стандартизації фінансових форматів для автоматичного обміну інформацією (наприклад, ISO 20022 [8], Financial Information Exchange (FIX) [9]), що можна використовувати при взаємодії корпоративних систем з банками.

Крім того, практичний досвід і приклади інтеграції систем з інформаційними платформами на кшталт ODOO переважно висвітлюються в тематичних блогах та вебінарах, на спеціальних форумах та в документації ODOO [2]. Такі джерела демонструють детальний опис реальних проектів, окреслюють виклики, переваги, а також подекуди вміщують елементи впровадження модулів для конвертації банківської інформації. Також вони дозволяють дізнатися більше про досвід послуговування такими розробками з погляду клієнта, почути бажання користувачів.

Проте варто зауважити, що стрімкий розвиток технологій і зміни у фінансовій та сферах ІТ часто роблять багато досліджень застарілими. Інновації в банківській сфері та вдосконалення підходів до розробки ERP-систем мають вплив на актуальність існуючих модулів і програм. Відповідно, з метою забезпечення максимально релевантної інформації для дослідження було сфокусовано увагу на аналізі сучасних наукових статей, матеріалах конференцій, публікацій у профільних журналах, а особливо – на актуальній документації ODOO.

З метою розуміння особливостей практичного застосування існуючих

програмних продуктів у цьому напрямку було детально розглянуто документацію платформи ODOO, простежено тенденцію клієнтських відгуків на тематичних форумах.

На офіційному сайті ODOO [1] можна знайти інструкції щодо налаштування та використання доступних модулів, навчальні матеріали, а також активний форум, де розглядаються актуальні питання стосовно роботи системи. Там же можна знайти і навчальний посібник, при ознайомленні з яким можна відзначити найновіші розробки у цьому напрямку.

Асоціація Odoo Community Association (OCA) [10] також стала цінним джерелом для дослідження. На її ресурсі публікується нова документація та оприлюднюються різноманітні модулі, які є доступними для вивчення та аналізу.

Окрему увагу приділено дослідженням, присвяченим мові Python, адже ця мова програмування широко застосовується в проектах ODOO. Завдяки її простоті, високорівневості, багатофункціональності та простоті і читабельності її коду Python є ідеальною мовою для створення рішень у фінансовій сфері.

З метою поглиблення знань про Python було вивчено офіційну документацію [11], яка надає детальний опис синтаксису мови, стандартних бібліотек та прикладів.

Також розглянуто публікації та книги, зокрема «Основи Python: Навчіться думати як програміст» Алена Б. Дауна [12] та «Automate the Boring Stuff with Python» Ела Свейгарта [13], які допомагають краще зрозуміти можливості мови. Python має велику спільноту розробників, яка активно обговорює ідеї та рішення на форумах, у соціальних мережах та на інших платформах, що стають зручним середовищем для обміну досвідом.

Такі дослідження демонструють важливість автоматизації фінансових процесів та інтеграції з ERP-системами для підвищення ефективності та точності роботи компаній.

2.2 Аналіз ресурсів, розробок та систем, що існують на сучасному етапі

Програмні продукти, які здатні здійснювати конвертацію банківських виписок у середовищі ODOO є ключовим інструментом для автоматизації управління фінансами й оптимізації обробки банківських транзакцій у бізнес-процесах. Є різні варіанти подібних систем, які мають свої переваги та недоліки, проте у межах дослідження основну увагу приділено трьом популярним програмним рішенням, оскільки вони займають велику нішу та широко використовуються на ринку.

Першою варто згадати систему Xero [14], яка є хмарною платформою для обліку, що забезпечує імпорт і експорт виписок у форматах, сумісних з різними інформаційними системами, у тому числі й ODOO. Серед основних переваг цього програмного продукту можна виокремити багатофункціональність (надання широкого спектру інструментів для управління фінансами, включно з бухгалтерським обліком, створенням звітів та управлінням платежами), гнучкість інтеграції (завдяки надаванню API для гарантованої інтеграції з різними системами, у тому числі й ODOO), візуалізацію даних тощо.

При усіх позитивних сторонах не можна оминати стороною і висвітлення недоліків Xero. Першим є те, що задля повноцінного функціонування програми клієнту необхідними є значні фінансові затрати, оскільки Xero має платну основу. Це, як і певні обмеження функціональності у специфічних аспектах фінансового керування, може стати вагомим обмеженням для компаній з невеликим бюджетом.

Наступним програмним забезпеченням, що дозволяє конвертувати банківські файли в формат XML, є система Bank2XML [15], яку також можна імпортувати в систему ODOO. Перевагами цієї системи є сумісність з ODOO, оскільки спеціалізація на форматі XML дозволяє інтегрувати виписки в ERP-систему, а також гнучкість налаштувань та використання мови програмування Python. Гнучкість забезпечує підтримку широкого спектру банківських форматів виписок,

а використання Python спрощує налаштування та кастомізацію. Проте функціональність програми обмежується переважною конвертацією у формат XML, а також вимагає високого рівня технічних знань від користувачів.

Третьою системою є BizConnect [16]. Вона вирізняється тим, що підтримує багато форматів виписок, включно з CSV, OFX і XLS, а також можливістю налаштування процесів обробки даних відповідно до потреб компанії. Відповідно, плюсами можна вважати широкий вибір форматів, гнучкість налаштувань і, безумовно, можливість інтеграції з ODOO. Мінусами при цьому є висока вартість системи для користувачів та обмежена підтримка деяких банків.

Таким чином, Xero, Bank2XML і BizConnect демонструють важливість таких програмних продуктів у сучасному бізнесі, але при цьому кожен з них має свої особливості, переваги та обмеження. Вибір оптимального рішення залежить від потреб компанії, бюджету і технічних можливостей. Відповідно, актуальним досі є пошук альтернативи, яка максимально вирішувала б ті недоліки, які мають існуючі системи. Цьому і було присвячено роботу.

Висновки до розділу 2

Проведений аналіз демонструє необхідність розвитку на сучасному етапі систем автоматизації конвертації банківських виписок для того, щоб корпорації могли займати свою конкурентоздатну нішу на фінансовому поприщі. Основними функціями подібних систем є імпорт банківських даних, автоматичне розпізнавання форматів, класифікація транзакцій та інтеграція з іншими модулями ODOO, такими як бухгалтерія. Такі системи також виконують перевірку цілісності даних, перевіряючи правильність рахунків, сум транзакцій та інші аспекти.

Рівень наукового дослідження цієї сфери не є абсолютним. З технічним розвитком попередні наукові праці, розробки та технології втрачають свою актуальність.

Загальними рисами існуючих систем є можливість імпорту банківських виписок, завдяки чому можна швидко та доступно отримувати фінансові дані без прив'язки до ручного введення інформації. Сюди ж можна віднести можливість автоматичного визначення форматів, класифікацію транзакцій тощо. Системи можна пов'язувати з іншими модулями ODOO, на кшталт бухгалтерії тощо. Подекуди вони також здатні перевіряти цілісність та достовірність банківської інформації.

Серед недоліків можна відзначити високу вартість, технічну складність налаштування, потребу у регулярних оновленнях і обмежену сумісність із деякими форматами. Порівняння Xero, Bank2XML та BizConnect підтвердило ці аспекти, дозволяючи виділити ключові напрями для вдосконалення існуючих рішень і пошуку нових підходів.

Подальші дослідження у цьому напрямку спрямовані на вибір оптимальних інструментів для розробки власного модуля конвертації, який би максимально відповідав сучасним потребам бізнесу та специфіці роботи з ODOO.

З ВИБІР ЗАСОБІВ РОЗРОБКИ СИСТЕМИ АВТОМАТИЗАЦІЇ КОНВЕРТАЦІЇ БАНКІВСЬКИХ ВИПИСОК НА ПРИКЛАДІ ІНФОРМАЦІЙНОЇ СИСТЕМИ ODOO

Засоби розробки програмного продукту є критичними для створення сучасних програм. Вони допомагають автоматизувати багато процесів, підвищують ефективність і забезпечують високу якість коду.

З огляду на особливості реалізації задачі та специфіки роботи, для створення сервісу, який автоматизує процес конвертації банківських виписок, було обрано низку ефективних інструментів. Основою стала ODOO 16 Community Edition, яка працює у зв'язці з базою даних PostgreSQL [17]. Для реалізації програмної логіки використовувалася мова програмування Python разом з набором бібліотек, які надають широкі можливості для розробки. Як фреймворк для створення веб-сервісу обрано Flask, що забезпечує простоту інтеграції та високу продуктивність. Як середовище розробки використовувалася інтегрована платформа розробки PyCharm [18], яка завдяки своїм інструментам і функціям значно полегшує процес розробки та тестування програмного забезпечення.

3.1 Інформаційна система ODOO

Система ODOO є потужним, гнучким і сучасним рішенням для автоматизації бізнес-процесів. Вона відноситься до класу ERP-систем з відкритим вихідним кодом та здійснює забезпечення інтегрованого управління широким спектром операцій підприємства. Важливим є те, що цю систему реалізовано на модульній структурі, що робить її налаштування під вимоги користувачів досить легким, комфортним та гнучким. Вона стала універсальним інструментом для підвищення

ефективності роботи підприємств. Завдяки значній кількості функціональних модулів ODOO охоплює пріоритетні сфери діяльності компанії, включно з управлінням продажами, закупівлями, фінансами, виробництвом, складом, людськими ресурсами тощо.

Головною перевагою ODOO є її відкритий вихідний код, що дозволяє розробникам вільно адаптувати її щодо специфічних потреб підприємства. Це забезпечує глибоку кастомізацію, можливість додавання нових функцій і можливостей та розширення системи за рахунок інтеграції із зовнішніми додатками або іншими платформами.

Серед переваг варто також наголосити на широкому спектрі функціональності системи, адже велика кількість доступних клієнту модулів дозволяє охоплювати різні сторони управління бізнесом, забезпечуючи при цьому комплексну автоматизацію. Також необхідно підкреслити гнучкість налаштування, завдяки якій систему легко адаптувати до специфічних бізнес-процесів, та зручність використання.

Важливим пріоритетом системи є те, що у неї активна спільнота розробників та великий користувацький потенціал, завдяки чому продукується підтримка розвитку системи, створюються нові програмні продукти, відбувається забезпечення постійного обміну досвідом та рішеннями питань, що виникають [19].

Завдяки таким характеристикам ODOO стала популярним вибором серед підприємств у всьому світі, надаючи користувачам не тільки гнучкість в налаштуванні, але й надійність у роботі.

На рисунку 3.1 відтворено інтерфейс інформаційної системи ODOO з відображенням можливості вибору доступних клієнтам модулів для встановлення. Модульність структури, конфігурація без додаткового програмування, можливість інтеграції з додатковими модулями та інтуїтивний інтерфейс стали тими зручними особливостями, які змушують користувачів надавати перевагу саме цій системі.

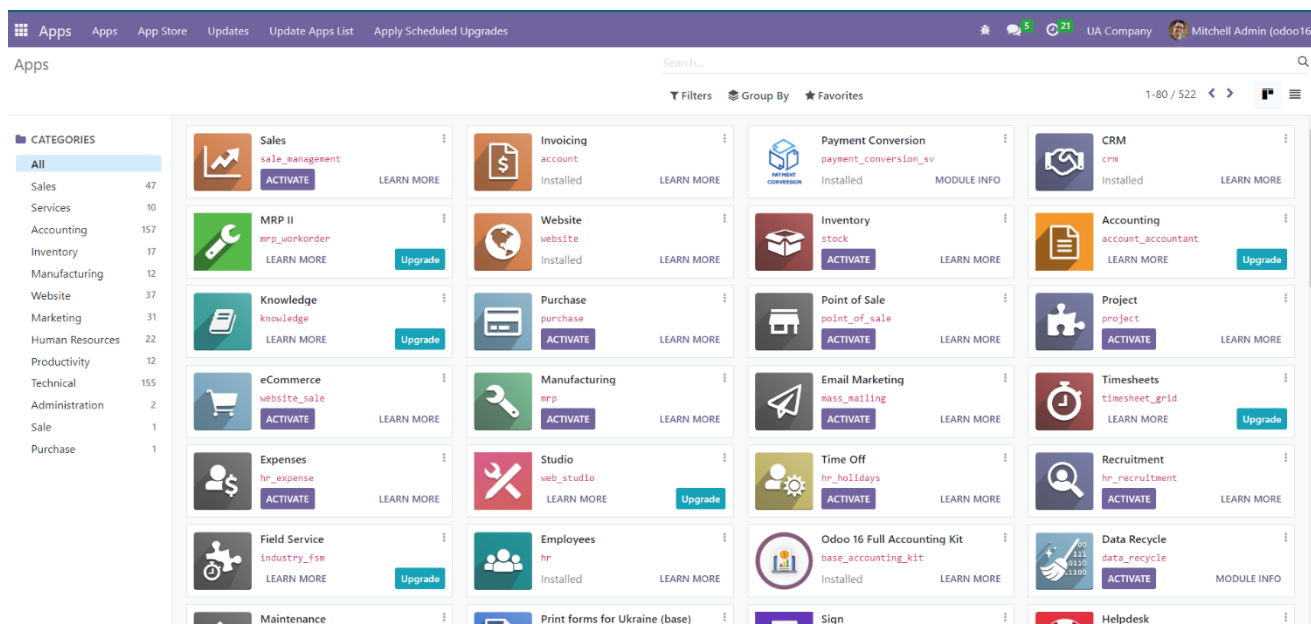


Рисунок 3.1 – Інтерфейс ODOO

Шляхом використання фреймворка ODOO було здійснено розробку клієнтської частини системи автоматизації конвертації банківських виписок.

3.2 База даних PostgreSQL

Система управління базами даних (СУБД) PostgreSQL зазвичай є основною для роботи з даними в ODOO, а загалом є одним із найпопулярніших виборів для розробки сучасних інформаційних систем. PostgreSQL вирізняється своєю потужністю, надійністю та гнучкістю, надаючи багато можливостей для зберігання, обробки й доступу до даних [20].

Серед ключових переваг цієї бази даних можна виокремити підтримку масштабувань, індексів, транзакцій, констрейнтів, відношень між таблицями тощо. PostgreSQL – це популярний вибір для потужних програм, оскільки характеризується чудовими розширюваністю та продуктивністю. Ця СУБД дає змогу зберігати великі об’єми даних, при цьому виконуючи складні запити та

забезпечуючи цілісність і надійність даних. Під час роботи з ODOO PostgreSQL є основним інструментом для зберігання й опрацювання даних, забезпечуючи необхідну інфраструктуру для функціонування ERP-системи. Під час розробки системи конвертації банківських виписок PostgreSQL використовується для валідації клієнтів, організації доступу до даних і взаємодії з доступними користувачу кількістю конвертацій.

Для управління базами даних PostgreSQL широко використовується зручний інструмент pgAdmin [21]. Цей графічний інтерфейс надає засоби для адміністрування, моніторингу й розробки баз даних, при цьому спрощуючи процес роботи з даними та зменшуючи технічну складність. Він дозволяє створювати, налаштовувати й оптимізувати структури бази. Це сприяє ефективній роботі програмного забезпечення та забезпечує стабільність системи. Цей інструмент відображено на рисунку 3.2.

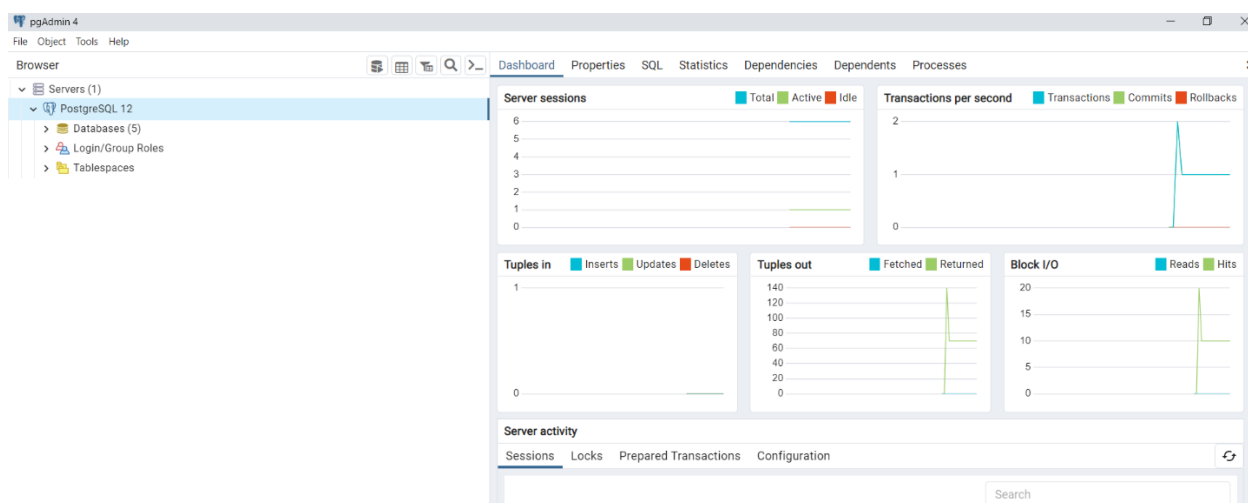


Рисунок 3.2 – Інтерфейс pgAdmin

Інструмент PgAdmin – це відкрите програмне забезпечення, яке є доступним для використання без фінансових витрат. Завдяки наданню ним розширених можливостей для роботи з СУБД PostgreSQL, це допомагає забезпечувати комфортне та продуктивне керування базою даних.

PostgreSQL залишається надійним вибором для реалізації складних інформаційних систем, таких як ODOO, завдяки своїй гнучкості й ефективності. Використання цього засобу розробки забезпечує високу якість обробки даних і можливість налаштування під різні потреби бізнесу.

3.3 Мова програмування Python

Наступним засобом розробки системи автоматизації конвертації було обрано мову програмування Python. Це одна з найпопулярніших і найпоширеніших мов програмування у світі, що здобула заслужене визнання завдяки своїй простоті, гнучкості та широким можливостям. Її обирають як початківці, які тільки розпочинають свій шлях у програмуванні, так і досвідчені розробники, які працюють у таких галузях, як веб-розробка, штучний інтелект, аналіз даних, автоматизація процесів та багато інших. Перевагами цієї мови програмування є її простота та зрозумілість, оскільки Python має простий та лаконічний синтаксис, що дозволяє легко читати та розуміти код. Ця особливість робить мову доступною для новачків і сприяє швидкому засвоєнню основ програмування.

Ще однією вагомою перевагою Python є її велика кількість бібліотек та модулів, як стандартних, так і сторонніх. Завдяки цьому розробники можуть швидко реалізовувати складні завдання без необхідності створювати все з нуля.

Python також підтримує різні парадигми програмування, включно з процедурним, об'єктно-орієнтованим та функціональним, що робить цю мову універсальним інструментом. Це дає змогу розробникам адаптувати свої підходи до потреб конкретного проєкту, забезпечуючи більшу ефективність та оптимальність у вирішенні задач.

Серед важливих характеристик Python, які мали важливе значення для роботи, варто виділити її переносимість. Програми, які написані цією мовою, можуть працювати на різних операційних системах без значних змін у коді. Це

забезпечує зручність використання та економію часу при розробці.

Мову програмування Python було обрано не випадково: вона є провідною для інформаційної системи ODOO. Її використовують для розробки модулів ERP-системи завдяки легкій інтеграції з іншими компонентами ODOO. Сама система автоматизації конвертації банківських виписок, розроблена в межах цього дослідження, тому й була написана на Python, оскільки ця мова програмування повністю відповідає вимогам платформи. Її простота, зручність і можливість реалізовувати навіть складні функції за допомогою готових інструментів роблять Python ідеальним вибором для подібного роду завдань.

3.4 Фреймворк Flask

Фреймворк Flask було обрано тому, що це легкий, проте потужний мікрофреймворк, призначений для створення веб-додатків, написаних мовою програмування Python [22]. Цей інструмент розроблений з акцентом на простоту використання, надаючи базовий набір функцій, необхідних для роботи з веб-запитами. Flask забезпечує ключові можливості, такі як маршрутизація, обробка HTTP-запитів, шаблонізація, а також можливість легко додавати додаткові функції за потреби.

Однією з переваг Flask є його гнучкість та мінімальне навантаження архітектурними вимогами. Він не нав'язує розробникам суворих правил або жорсткої архітектури, що дозволяє створювати додатки різної складності. Розробники можуть самостійно обирати бібліотеки та інструменти для реалізації необхідного функціоналу та створювати оптимальне середовище для кожного конкретного проєкту. Flask підходить і для невеликих проєктів, і для більш складних систем, де потрібно інтегрувати різні сервіси чи технології.

Як і в Python, додатковою перевагою Flask є його активна та численна спільнота розробників, яка забезпечує швидкий розвиток фреймворка, регулярне

оновлення та широкий спектр сторонніх бібліотек. Це дозволяє легко знайти відповіді на питання, обрати готові рішення для поширених завдань або отримати підтримку, якщо виникають труднощі чи питання. Завдяки своїй простоті та популярності, Flask є одним із найкращих інструментів для тих, хто хоче швидко створити веб-додаток на Python.

Під час розробки системи автоматизації для системи ODOO, Flask відіграв важливу роль. За допомогою нього був реалізований сервер взаємодії, який забезпечує злагоджену комунікацію між клієнтською частиною та модулем конвертації банківських виписок. Flask продемонстрував свою продуктивність у створенні швидких і надійних рішень для обробки запитів та обміну даними в системі.

3.5 Середовище розробки PyCharm

Досягнення гарного результату практичної реалізації системи автоматизації конвертації стало можливим завдяки використанню одного з ключових інструментів для розробників – середовища розробки PyCharm. Це інтегроване середовище для програмування забезпечує комфортну, функціональну та продуктивну платформу для написання, тестування та управління програмами мовою Python [23].

Середовище PyCharm володіє великим набором інструментів, які значно полегшують і пришвидшують роботу розробників. Однією з основних переваг є функція автодоповнення коду. Вона дозволяє економити час, зменшуючи кількість друкарських помилок. Це сприяє швидшому написанню програм. Крім цього, PyCharm має вбудовані інспектори помилок, які допомагають одразу ідентифікувати й виправляти потенційні проблеми в коді. Відлагоджувач у PyCharm дозволяє тестувати програму, стежачи за її виконанням, що забезпечує надійне налагодження.

Важливим фактором є доступність PyCharm для різних операційних систем, зокрема Windows, macOS і Linux. Така підтримка платформ робить це середовище універсальним інструментом, що підходить для роботи в різних середовищах, оскільки є незалежним від обрання операційних систем.

Ще одним важливим пріоритетом є можливість інтеграції PyCharm з іншими популярними інструментами, які активно використовуються під час розробки. Наприклад, PyCharm підтримує системи контролю версій, такі як Git [23], що полегшує можливість відстежувати зміни у коді та співпрацю між розробниками. Середовище також легко інтегрується з веб-фреймворками, базами даних та віртуальними середовищами, що дозволяє втілювати проєкти будь-якої складності.

На рисунку 3.3 можна побачити загальний вигляд середовища розробки PyCharm.

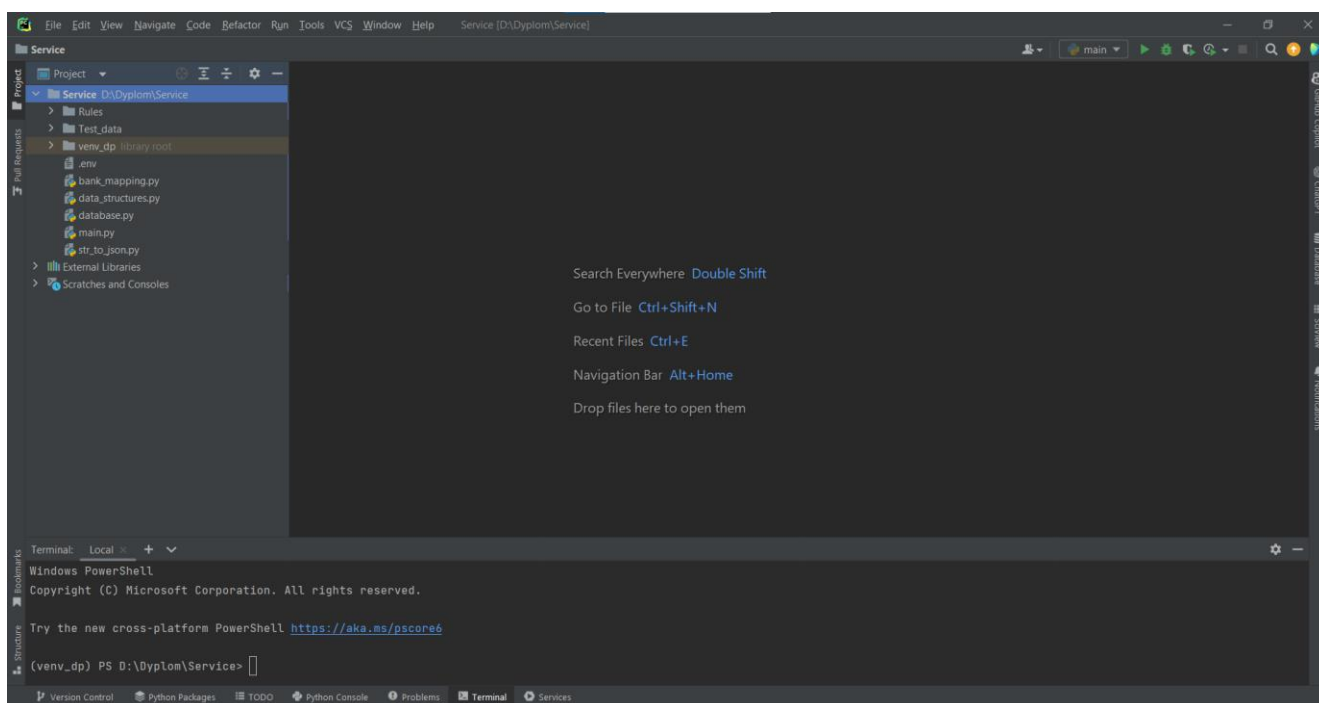


Рисунок 3.3 – Інтерфейс середовища розробки PyCharm

Таким чином, PyCharm – це необхідний та потужний інструмент для тих, хто здійснює розробки мовою програмування Python. Завдяки йому розробники

отримують зручне, комфортне і результативне середовище для того, щоб написати, протестувати та налагодити свій код. Відповідно, PyCharm відіграє важливу роль під час розробки програмного забезпечення, надаючи розробникам інструменти для створення якісного коду. У цьому проєкті середовище було незамінним для реалізації системи автоматизації конвертації, що стало вагомим внеском у її успіх.

Висновки до розділу 3

Засоби розробки – це інструменти та технології, які використовуються для створення програмного продукту. Вони забезпечують розробникам можливість писати код, тестувати його, налагоджувати помилки, організовувати роботу з даними та інтегрувати різні системи. Підбір правильних засобів розробки програмного продукту є критичними для створення сучасних програм. Вони допомагають автоматизувати багато процесів, підвищують ефективність і забезпечують високу якість коду.

Під час розробки системи автоматизації конвертації банківських виписок на прикладі інформаційної системи ODOO було використано набір сучасних, потужних та перевірених інструментів. Основними засобами розробки стали мова програмування Python, фреймворки Flask та ODOO, система управління базами даних PostgreSQL, а також інтегроване середовище розробки PyCharm.

Кожен із цих інструментів відіграє ключову роль у створенні якісного програмного забезпечення та є популярним серед розробників завдяки своїм унікальним можливостям. ODOO забезпечує багатофункціональну платформу для управління бізнес-процесами, дозволяючи легко інтегрувати нові модулі у вже існуючу систему. Мова програмування Python, завдяки своїй простоті, гнучкості та багатству можливостей, стала основною для створення програмного коду. У свою чергу Flask забезпечив швидке й ефективне створення веб-додатків для взаємодії частинами системи. Потужна і надійна система управління базами даних

PostgreSQL дала змогу забезпечити ефективне зберігання та обробку великих обсягів інформації. І, нарешті, PyCharm надав інструменти для зручного написання, тестування та налагодження програмного забезпечення.

Обрання таких засобів розробки було не випадковим. Воно ґрунтувався на аналізі мети та завдань, поставлених перед проєктом, а також на необхідності забезпечити відповідність кінцевого продукту функціональним та нефункціональним вимогам, які було поставлено перед ним. Кожен інструмент, що був використаний під час розробки системи автоматизації конвертації, був ретельно обраний для забезпечення максимальної продуктивності та зручності.

4 ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ СИСТЕМИ АВТОМАТИЗАЦІЇ КОНВЕРТАЦІЇ БАНКІВСЬКИХ ВИПИСОК НА ПРИКЛАДІ ІНФОРМАЦІЙНОЇ СИСТЕМИ ODOO

Для успішної розробки системи автоматизації конвертації було необхідно виконати декілька підготовчих етапів, серед яких були і завантаження та встановлення необхідного програмного забезпечення. Зокрема, було встановлено платформу Odoo 16 [24] та базу даних PostgreSQL, яка стала основою для зберігання та обробки даних. Крім цього, було встановлено Python [25] – мову програмування, яка використовується для створення модуля, середовище розробки PyCharm, що забезпечує зручність і продуктивність роботи з кодом, а також веб-фреймворк Flask для реалізації серверної частини веб-додатка. Далі потрібно спроектувати та реалізувати архітектуру системи автоматизації, до складу якої входять модуль конвертації та клієнтський і адміністративний модулі.

4.1 Проєктування архітектури системи автоматизації

Одним із перших і найважливіших кроків стала розробка архітектури самого модуля конвертації. Було створено структурну схему програмного продукту, який буде виконувати автоматизоване перетворення банківських виписок із одного формату в інший.

Архітектура визначає, як різні компоненти системи взаємодіють між собою, які технології будуть використовуватися, як зберігатимуться та оброблятимуться дані, і як відбуватиметься інтеграція з іншими системами.

Таке планування дало змогу чітко визначити, як буде виглядати та функціонувати продукт, зокрема, модуль конвертації банківських виписок, який

автоматично перетворюватиме виписки з одного формату в інший. Після цього було реалізовано користувацький та адміністративний модулі в системі ODOO. Ці модулі відіграють важливу роль у забезпеченні інтеграції власне самого модуля конвертації до системи ODOO підприємства.

Сполучення та об'єднання модуля конвертації та службової складової в ERP-системі компанії завбачає взаємодію модулів для забезпечення автоматичного обміну банківськими виписками та спрощення врахування коректив у форматах платіжок в системі підприємства.

Особливість взаємодії усіх складових відтворено на рисунку 4.1.

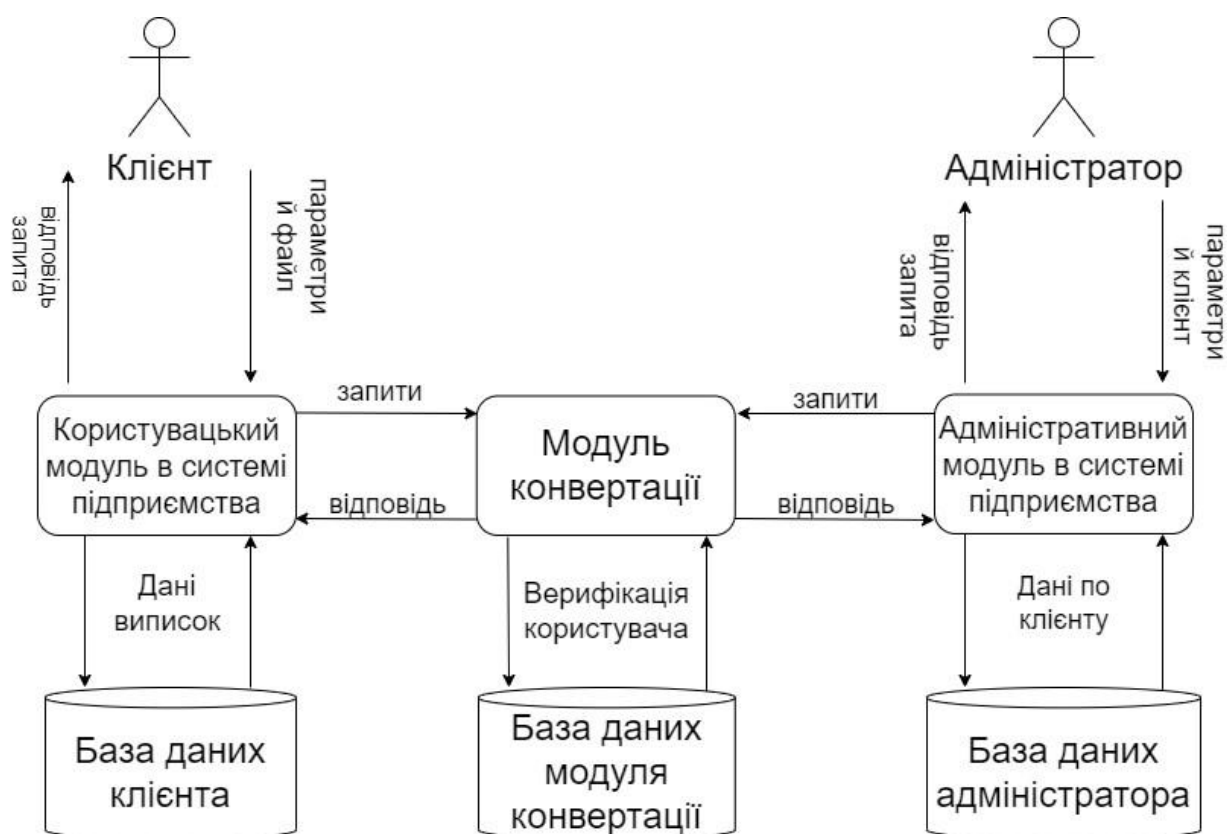


Рисунок 4.1 – Взаємодія складових

Об'єднання складових системи створює всі умови для ефективної взаємодії між компонентними модулями. Це дозволяє автоматизувати обмін банківськими виписками. Також це значно полегшує процес адаптації до змін у форматах

виписок у межах інформаційної системи ODOO, забезпечуючи її стабільне та зручне функціонування.

4.2 Внутрішня структура та засоби реалізації модуля конвертації

Основна функція модуля конвертації – трансформація платіжних файлів у стандартний формат JSON [26], який є універсальним і зручним для подальшого імпорту в інформаційну систему ODOO. При цьому модуль оперує набором файлів, кожен із яких виконує конкретні функції та має своє призначення для забезпечення коректної роботи процесу автоматизації. Файлову структуру продемонстровано на рисунку 4.2.

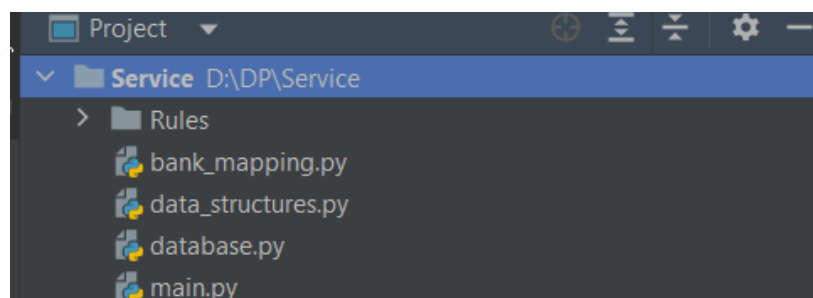


Рисунок 4.2 – Файлова структура модуля конвертації

Структура модуля конвертації складається з кількох ключових файлів, кожен із яких виконує свою роль:

- main.py – основний файл продукту, який забезпечує ключову логіку роботи і маршрутизацію запитів, які надходять від клієнтської частини в ODOO;
- database.py – модуль, який відповідає за взаємодію з базою даних PostgreSQL. Він містить клас для створення з'єднання, виконання запитів і роботи з таблицею token_transactions, яка зберігає дані про токени та кількість транзакцій

для будь-якого з них (приклад класу показано на рисунку 4.3).

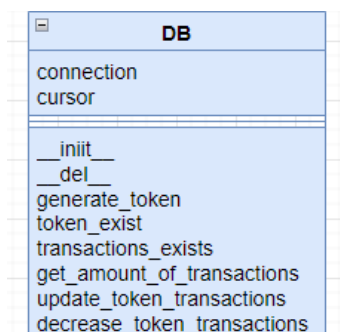


Рисунок 4.3 – Клас взаємодії з базою даних

- `bank_mapping.py` – модуль, що вміщує відображення кодів банкових організацій на об'єкти-парсерів для опрацювання платіжних файлів;
- `data_structures.py` – модуль, де визначено структури даних; серед них, зокрема, і клас `Transaction`, який використовується задля представлень платіжних транзакцій; приклад класу відображено на рисунку 4.4.

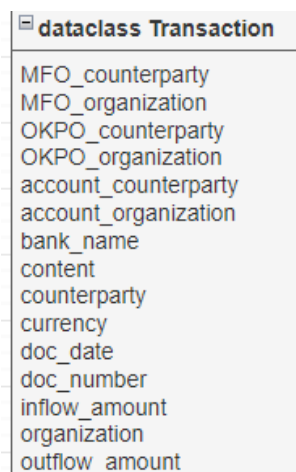


Рисунок 4.4 – Клас Transaction

Ці файли забезпечують чітку і злагоджену роботу модуля, здійснюючи оптимізацію процесу обробки банківських виписок.

У папці Rules вміщено файли, у яких зазначено правила обробки файлів платіжок відповідно до кожного банку, що підтримується системою. Як приклад таких файлів можна розглянути ukrsib.py – модуль, у якому вміщено правила для аналізу та обробки платіжних файлів банку «Ukrsibbank» [26]. Цей клас продемонстровано на рисунку 4.5.

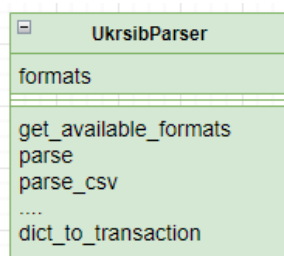


Рисунок 4.5 – Клас UkrsibParser

Модуль є сервером, створеним на основі Flask, що забезпечує функціонування HTTP API для конвертації файлів, отриманих від різних банків. Його функціонал дозволяє реєструвати клієнтів і видавати їм унікальні токени, які використовуються для обміну банківськими платіжними файлами.

Для роботи сервером використовується база даних PostgreSQL, де зберігається інформація про видані токени та кількість транзакцій, що обробляються для кожного клієнта. Таблиця наведена на рисунку 4.6.

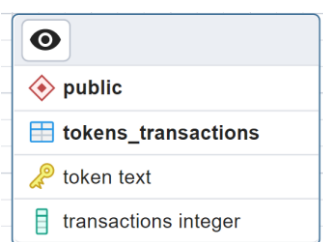


Рисунок 4.6 – Таблиця token_transactions

Основну логіку роботи сервера реалізовано в провідному файлі `main.py`, який вміщує кілька ендпойнтів [25] для виконання головних функцій:

`/generate-token` – відповідає за реєстрацію користувача та наділення його унікальним токеном;

`/get_available_formats` – запитує підтримувані формати платіжних файлів від процесора визначеного банку;

`/payment-conversion` – здійснює конвертацію платіжних файлів. Клієнтом здійснюється надсилання серверу платіжного файлу у вигляді рядка, а також вказування формату файлу та банк. Сервер перевіряє валідність токена клієнта та доступну кількість транзакцій. За умови, що все в нормі, відбувається виклик процесору банку для конвертації файлу.

`/decrease-conversions` – зменшення кількості можливих транзакцій для клієнта після того, як файл було конвертовано;

`/get-amount-of-transactions` – повернення кількості можливих транзакцій клієнту.

Цими API-маршрутами та їхніми функціями забезпечуються реєстрація користувачів, обробка файлів, отримання форматів, управління лічильником транзакцій та загальну взаємодію між клієнтом та сервером, що робить систему гнучкою та ефективною для обробки даних із різних банків та в різних форматах.

4.3 Структура та особливості реалізації клієнтського модуля

Клієнтська частина – це кастомний модуль, створений на базі на фреймворку ODOO. Її структура включає модель для роботи з базою даних, яка зберігає інформацію, введену користувачем, а також представлення, яке забезпечує зручний інтерфейс для виконання основних дій [30]. Через цей інтерфейс клієнт може завантажувати різноманітні платіжні виписки, обирати банк та відповідний формат файлу, переглядати конвертовані дані тощо.

Усі складові модуля згруповано у відповідних директоріях та файлах, що забезпечують його працездатність, функціональність та інтеграцію з інформаційною системою ODOO.

Кожна структурна складова модуля направлена на реалізацію працездатності, забезпечення функціоналу та зручності взаємодії з користувачем. Вигляд структури клієнтської частини наведено на рисунку 4.7.

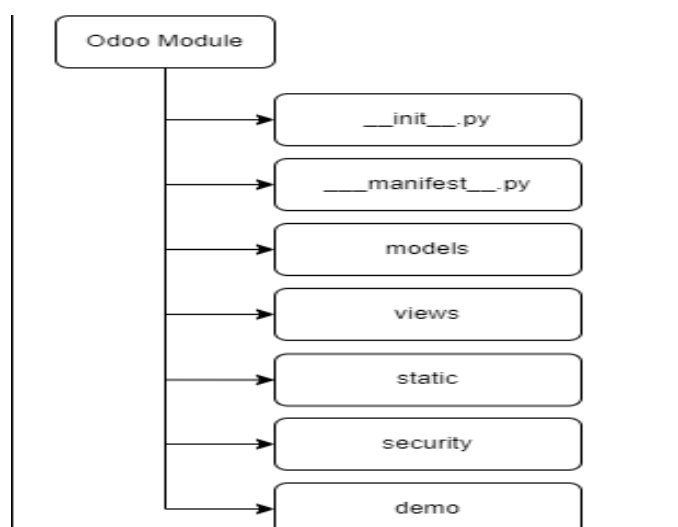


Рисунок 4.7 – Структура клієнтського модуля в ODOO

Файл «`__init__.py`» у Python позначає директорію як пакет або модуль, забезпечуючи цим правильну роботу імпортів і структури проєкту. Файл «`__manifest__.py`» є обов’язковим елементом модуля ODOO, що містить основну інформацію, як-от назва, версія, залежності, опис, автори й налаштування. Його також використовують для того, щоб включити інші файли конфігурації клієнтського модуля.

У складовій «`models`» розташовано файли Python з описами моделей даних. Вони застосовуються з метою визначити структуру таблиць бази даних, поведінку та логіку, пов’язану з моделлю, атрибути та методи моделей тощо.

Файли XML з відтвореннями представлень вміщуються у «`views`». Ці

компоненти застосовуються для демонстрації даних моделей у веб-інтерфейсі ODOO.

Складова «security» містить XML-файли для визначення ролей і прав доступу користувачів. Вони необхідні для того, щоб обмежувати доступ клієнтів до різноманітних функцій модуля та його даних.

Ресурси для зовнішнього вигляду (його налаштування), такі як зображення чи стилі, знаходяться у директорії «static». Для того, щоб розміщувати файли XML з інформацією демонстраційних записів, використовується додаткова директорія «demo», що присутня в конструкції модуля ODOO.

Для повноцінної роботи клієнтської частини модуля було створено модель таблиці для бази даних PostgreSQL, яка тимчасово зберігає введені клієнтом дані й забезпечує виконання операцій, необхідних для подальшої роботи з модулем. Поля цієї моделі зображені на рисунку 4.8.

PaymentConversion
_name _description
file file_name file_format_id
bank
amount_of_available_transaction unlimited
converted_file converted_file_name converted_payment_ids

Рисунок 4.8 – Модель PaymentConversion

До її структури належать поля, які є відповідальними за збереження даних про файли, банки, до яких належать платежі, токени клієнтів, а також доступні для них кількості транзакцій.

Модель бази даних клієнтського модуля також включає поле для збереження конвертованого файлу, а також реалізує низку методів для забезпечення функціональності клієнтської частини. Серед них – методи для запитів до модуля конвертації з метою отримання доступних форматів файлів залежно від обраного банку, перевірки кількості доступних конвертацій для клієнта, а також генерації власне самого токена.

Основним методом є «convert_payment», який формує запит із необхідними даними, надсилає їх до модуля конвертації й отримує у відповідь готовий конвертований файл.

Відповідно, клієнтський модуль в ODOO забезпечує користувачеві доступ до ключових дій, які проілюстровано на діаграмі варіантів використання (use case) на рисунку 4.9.



Рисунок 4.9 – Діаграма варіантів використання клієнтом

Після того, як користувач зареєструється в сервісі, він отримає доступну йому кількість конвертацій, після чого зможе завантажувати файли з платіжками до системи, з таблиці обирати банк, визначати доступний та необхідний формат файлу, після чого відправляти файл на конвертацію. По завершенню цього процесу клієнту стає доступним вивантаження отриманого файлу.

Варто відзначити, що користувач також отримує можливість перегляду історії попередніх конвертацій та можливість формувати виписки відповідно до рахунку в інформаційній системі ODOO.

4.4 Структура та особливості реалізації адміністративного модуля

Подібно до клієнтського модуля адміністративний службовий модуль також надає інтерфейс, але уже для користувача-адміністратора. Якщо клієнт отримує змогу здійснювати конвертації необхідних банківських виписок, обираючи необхідний банк із списку доступних, формат файлу, яким буде послуговуватися під час конвертації тощо, то адміністратор у цьому сервісі отримує змогу здійснювати відстеження використання клієнтами модуля конвертації та його функціональних можливостей, аналіз їхньої діяльності, а також керування клієнтським доступом до сервісу.

Файлова структура адміністративної частини є аналогічною до структури клієнтського модуля. Відмінним насамперед є функціонал цього структурного модуля системи автоматизації конвертації банківських виписок, який визначає можливості адміністратора при роботі з системою.

Поля моделі панелі адміністратора відображено на рисунку 4.10.

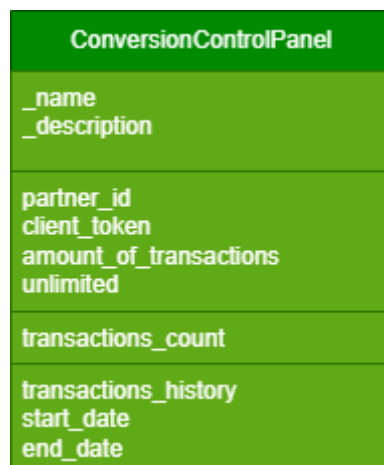


Рисунок 4.10 – Модель ConversionControlPanel

При використанні цього модуля, який є розробленим в ODOO, адміністратору стають доступними дії, що відображено на usecase діаграмі рисунка 4.11.



Рисунок 4.11 – Діаграма варіантів використання адміністратором

Адміністративний модуль в ODOO, який знаходиться в базі дистрибутера системи конвертації, надає адміністратору можливості вибору клієнта та генерації

токена для нього, керування кількістю доступних конвертацій, надавання і відняття безліміту, а також перегляду історії в структурованому списку з визначенням того, хто і які дії виконував.

Висновки до розділу 4

Під час опису програмного втілення системи автоматизації конвертації банківських виписок на прикладі інформаційної системи ODOO необхідно сконцентрувати увагу на демонстрації внутрішньої структури модуля конвертації та засобів його реалізації і на відображенні клієнтської частини та адміністративного службового модуля.

Модуль конвертації реалізований у вигляді Flask-сервера, що реалізує HTTP API для того, щоб конвертувати платіжні файли різних банківських установ. Його функція полягає у перетворенні платіжних файлів у формат JSON, що є універсальним, для подальшого імпорту в ODOO. Сервер дає змогу реєстрації клієнту, забезпечує інтеграцію даних та видає токени, які використовуються для обміну платіжками.

Клієнтський модуль побудований на базі фреймворку ODOO і забезпечує зручний та доступний розумінню інтерфейс для завантаження файлів, обрання банків та необхідних форматів, перегляду результатів здійснених конвертацій тощо. Введені клієнтом дані зберігаються в моделі бази даних, що є ґрунтом функціонування модуля. Користувач отримує змогу не тільки переглядати форматовані файли, але й вивантажувати їх на свій носій для подальшого використання.

Окрім клієнтського модуля, було розроблено адміністративний модуль, який надає інструменти для адміністраторів. Цей модуль дозволяє відслідковувати та керувати використанням системи конвертації клієнтами, зокрема контролювати кількість доступних конвертацій, призначати чи знімати безлімітні права на

конвертацію, а також переглядати історію дій клієнтів. Це допомагає забезпечити безперебійне функціонування системи та своєчасне реагування на потреби користувачів.

Завдяки взаємодії та інтеграції цих компонентів системи автоматизації користувачі отримують автоматизоване рішення для обробки банківських транзакцій без ручного введення даних.

5 РОБОТА КОРИСТУВАЧА З СИСТЕМОЮ АВТОМАТИЗАЦІЇ КОНВЕРТАЦІЇ

Для того, щоб взаємодіяти із системою автоматизації конвертації банківських виписок в інформаційній системі ODOO користувач використовує спеціальний інтерфейс, який інтегровано безпосередньо в систему. Цей інтерфейс забезпечує легкий і зручний доступ до функціоналу системи, дозволяючи виконувати всі необхідні операції.

Для початку роботи з програмним продуктом користувач повинен встановити клієнтську частину системи в своє середовище ODOO. Під час інсталяції варто враховувати вимоги до системи, оскільки вони забезпечують коректну роботу модуля. Після того, як усі попередні дії було виконано, а клієнтську частину успішного налаштовано, користувач може безперешкодно переходити уже до самої конвертації тих файлів, які потребують обробки.

5.1 Рекомендовані системні вимоги

Рекомендовані системні вимоги до програмного продукту – це набір параметрів апаратного та програмного забезпечення, який забезпечує оптимальну продуктивність та стабільну роботу програмного продукту. Вони пропонуються як орієнтир для клієнтів, щоб уникнути потенційних проблем із продуктивністю чи сумісністю.

Для того, щоб система автоматизації конвертації банківських виписок у системі ODOO працювала коректно, користувачеві необхідно врахувати певні системні вимоги, які забезпечать стабільну й безперебійну роботу програми. Ці вимоги зазначено у таблиці 5.1.

Таблиця 5.1 – Системні вимоги

№ п/п	Системні вимоги	Рекомендації	Коментар
01	Операційна система	Windows, macOS і Linux	Користувач повинен забезпечити наявність комп'ютера або сервера, що працює на одній із цих платформ, оскільки це є ключовою умовою для інсталяції та запуску системи
02	Апаратні вимоги	Об'єм оперативної пам'яті, процесора та обсягу сховища	Для ефективної роботи необхідно використовувати комп'ютер або сервер із достатньою потужністю. Заявлені вимоги мають відповідати розмірам даних, що обробляються, кількості користувачів та інтенсивності навантаження на систему. Обладнання має бути розраховане на стабільну роботу навіть за умов підвищеного навантаження
03	Мережеве з'єднання	Стабільний і швидкий доступ до інтернету	ODOO функціонує на основі серверної архітектури, тому для завантаження та обробки даних потрібен стабільний і швидкий доступ до Інтернету. Надійне мережеве з'єднання гарантує своєчасну передачу та синхронізацію інформації між клієнтом і сервером
04	Веб-браузер	Google Chrome, Safari або Microsoft Edge тощо	Для доступу до інтерфейсу системи користувач має використовувати сучасний веб-браузер. Рекомендується завантажувати останні версії браузерів, щоб забезпечити максимальну сумісність і коректне відображення інтерфейсу

Дотримання вказаних вимог є запорукою успішного налаштування та використання системи автоматизації конвертації користувачем, що дозволить забезпечити стабільність роботи і точність виконання завдань.

5.2 Послідовність встановлення системи користувачем

Щоб успішно інстальювати клієнтський блок модуля в систему ODOO, користувачеві треба виконати кілька простих кроків. Спочатку необхідно увійти до

системи ODOO, використовуючи свій обліковий запис. Далі необхідно знайти та обрати розділ «APPS» [1] у головному меню. Після цього на екрані відкриється сторінка, де будуть відображені всі доступні модулі, які можуть бути додані до системи, як це відображено на рисунку 5.1.

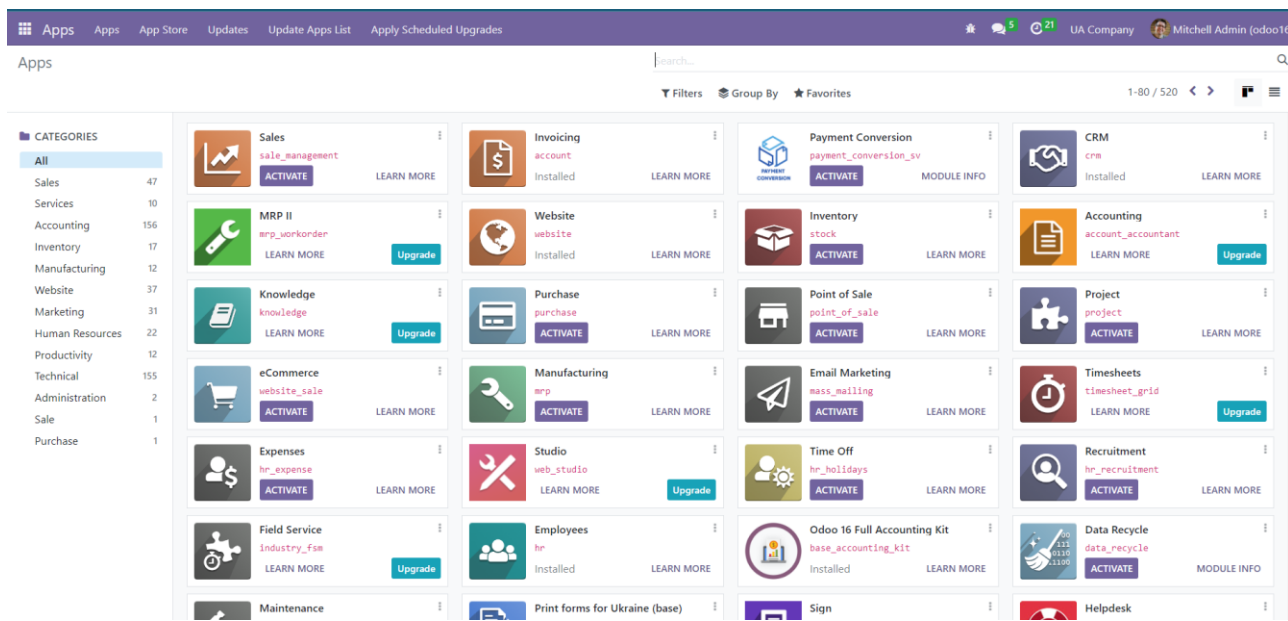


Рисунок 5.1 – Вкладка APPS

Відображений клієнту список модулів включає різні інструменти та розширення, доступні для встановлення та використання, які можна інсталиювати для розширення та поглиблення функціональних можливостей інформаційної системи ODOO користувача. Завдяки зручному інтерфейсу клієнт може переглядати, вибирати та встановлювати модулі відповідно до його потреб та потреб, що забезпечує високу гнучкість у налаштуванні системи під конкретні бізнес-завдання.

Після цього потрібно в пошуковому вікні інформаційної системи ввести назву розробленого програмного продукту – PaymentConversion. Після того, як сервіс буде знайдено, необхідно встановити його до інформаційної системи ODOO клієнта натисканням кнопки Activate, як це продемонстровано на рисунку 5.2.

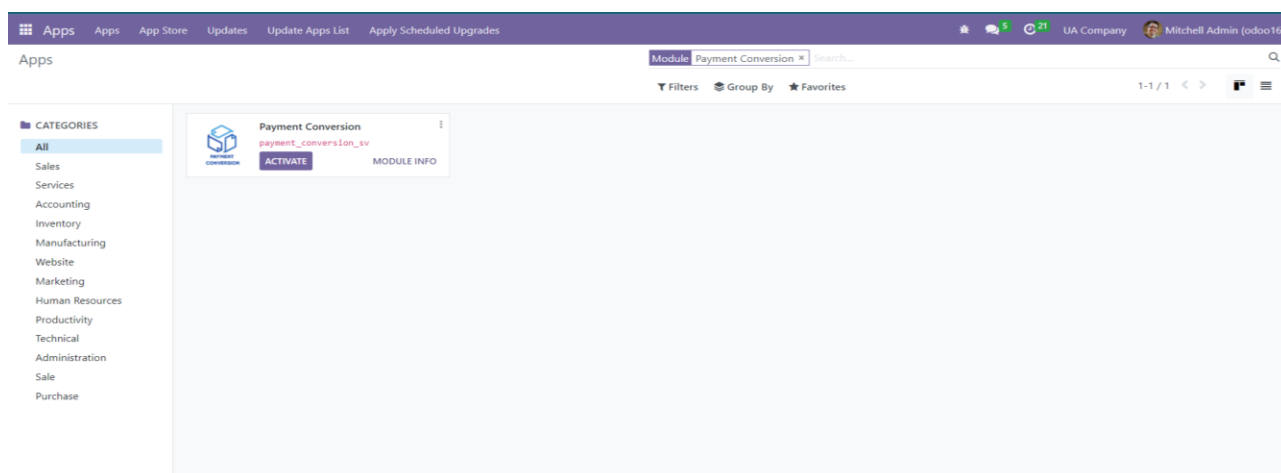


Рисунок 5.2 – Встановлення модуля

Після того, як процес встановлення програмного продукту буде завершено, користувач зможе побачити його серед доступних опцій у вкладці «Меню» системи ODOO. Цей пункт меню дозволяє швидко переходити до функціоналу встановленого модуля. Як приклад, зображення на рисунку 5.3 ілюструє, як саме виглядає відображення модуля після його успішного додавання до системи.

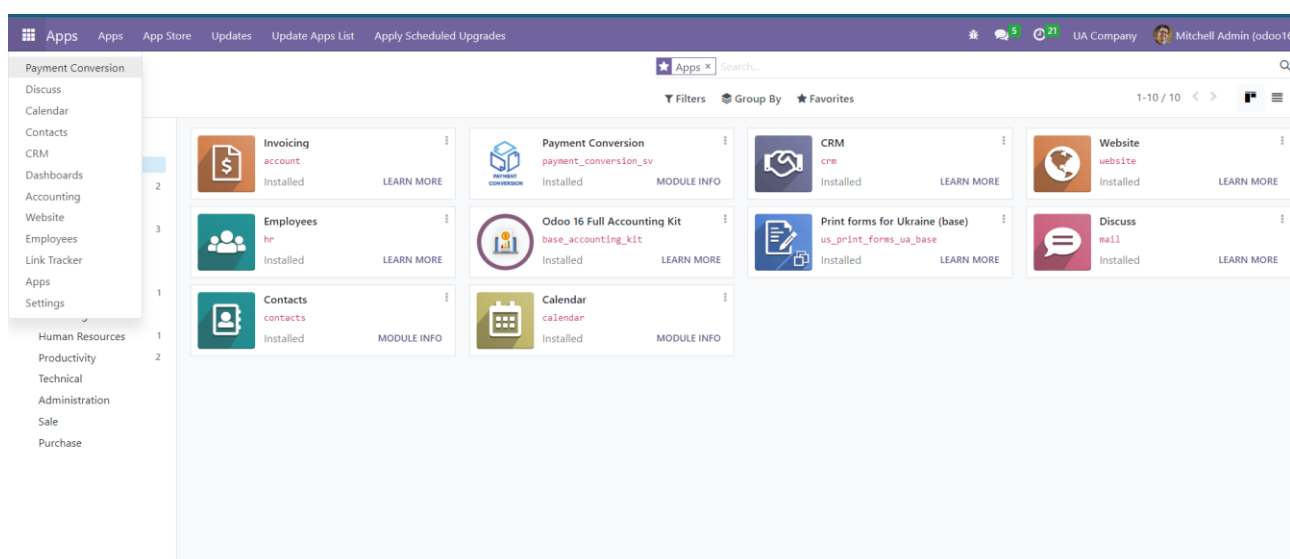


Рисунок 5.3 – Вкладка «Меню» із встановленим Payment Conversion

Тепер користувач має змогу легко знайти встановлений модуль у загальному інтерфейсі платформи. Це покращує доступ до функцій, які надає сервіс, і дозволяє розпочати клієнтом роботу з його функціоналом.

Таким чином, розроблений та реалізований програмний продукт є відкритим для того, щоб клієнт міг розпочати його використання.

5.3 Використання системи автоматизації конвертації банківських виписок клієнтом

Користувачами системи автоматизації конвертації банківських виписок є клієнт та адміністратор.

Після успішної інсталяції програмного продукту клієнт може розпочати його використання. Для цього необхідно перейти до пункту меню «Payment Conversion», який з'явиться у системі. Після обрання цього пункту користувач потрапляє до спеціального інтерфейсу клієнтського модуля. На екрані відобразиться вікно, у якому надалі буде доступна історія всіх виконаних конвертацій. Цей вигляд інтерфейсу відображено на рисунку 5.4.

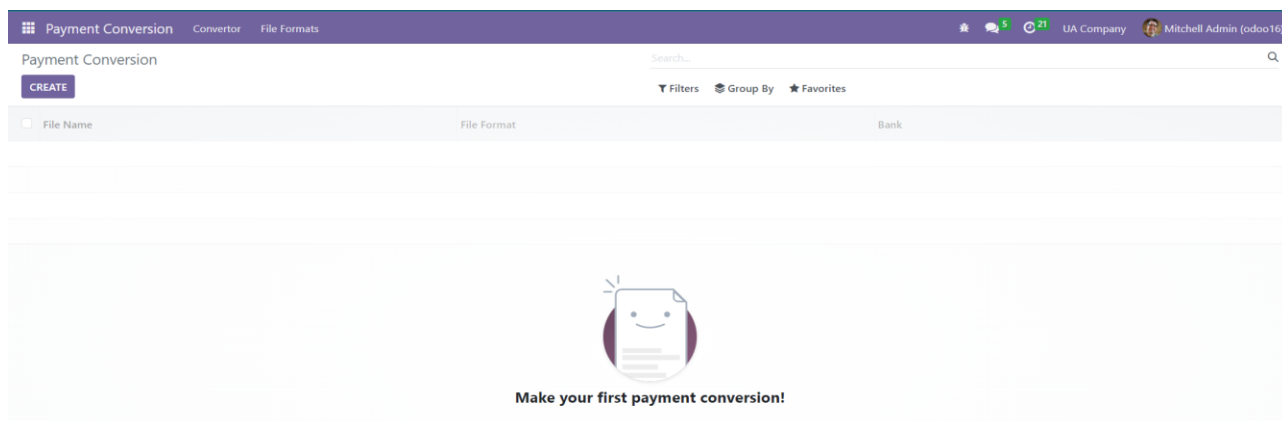
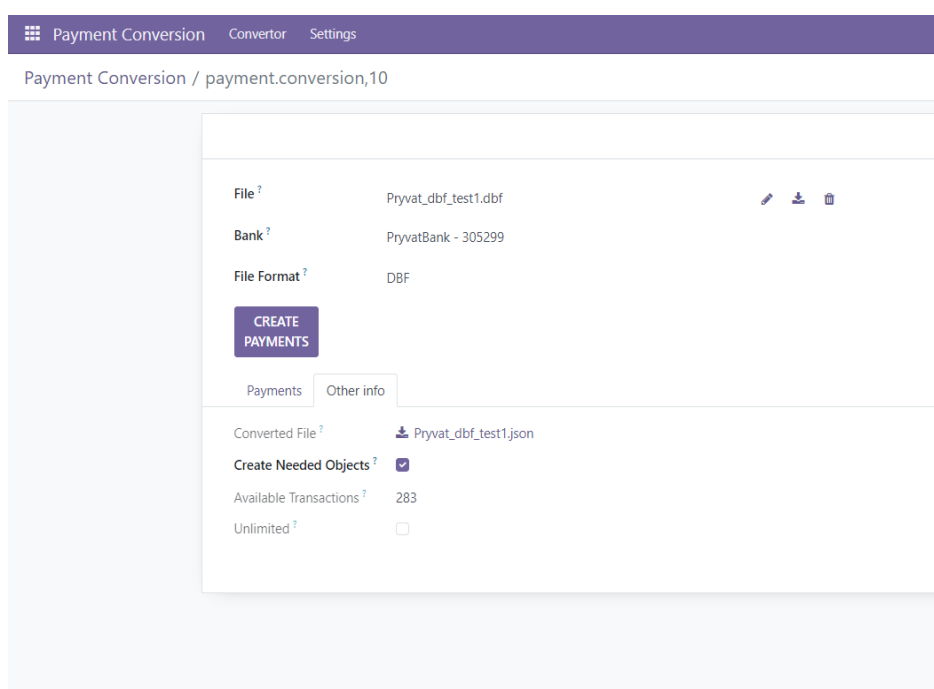


Рисунок 5.4 – Вікно історії конвертацій

Таким чином, модуль надає клієнту інтуїтивний, зручний та зрозумілий спосіб управління конвертаціями із забезпеченням доступу до інформації про попередні операції та про підготовленість до подальшого використання ним сервісу Payment Conversion.

Після того, як клієнт натисне кнопку Create, далі у цій вкладці буде відкрито форму, яку користувач може заповнювати для здійснення конвертації файлу. Детальніше це відображено на рисунку 5.5.



The screenshot displays a web application interface for 'Payment Conversion'. At the top, there is a navigation bar with 'Payment Conversion', 'Converter', and 'Settings'. Below this, a breadcrumb trail shows 'Payment Conversion / payment.conversion,10'. The main form area contains several input fields: 'File' with the value 'Pryvat_dbf_test1.dbf', 'Bank' with 'PryvatBank - 305299', and 'File Format' with 'DBF'. A prominent purple button labeled 'CREATE PAYMENTS' is positioned below these fields. Below the button are two tabs: 'Payments' (selected) and 'Other info'. Under the 'Payments' tab, there is a 'Converted File' field showing 'Pryvat_dbf_test1.json', a 'Create Needed Objects' checkbox which is checked, and two other fields: 'Available Transactions' with the value '283' and 'Unlimited' which is unchecked.

Рисунок 5.5 – Форма заповнення для конвертації файлу

Коли клієнт натисне кнопку «Upload your file», система відкриє вікно, де він зможе обрати необхідний для подальшої конвертації файл. У цьому вікні користувач отримає можливість знайти необхідний документ на своєму пристрої, як це ілюструється на рисунку 5.6.

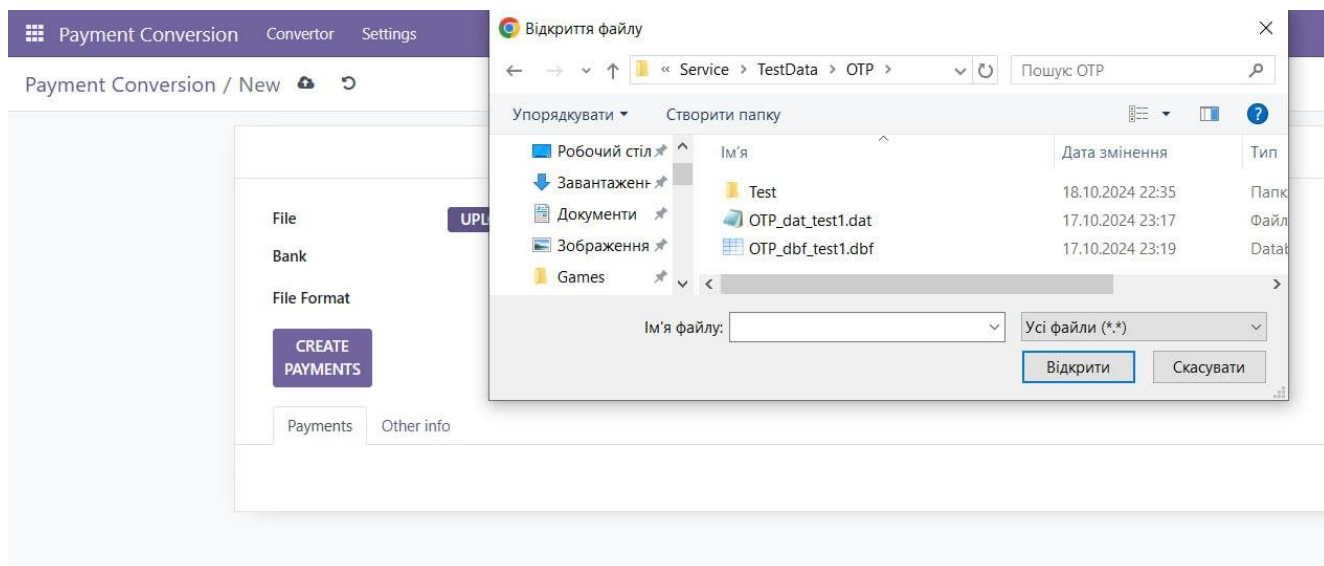


Рисунок 5.6 – Завантаження файлу

Наступним кроком має бути обрання потрібного клієнту банку в полі «Bank», як це відображено на рисунку 5.7.

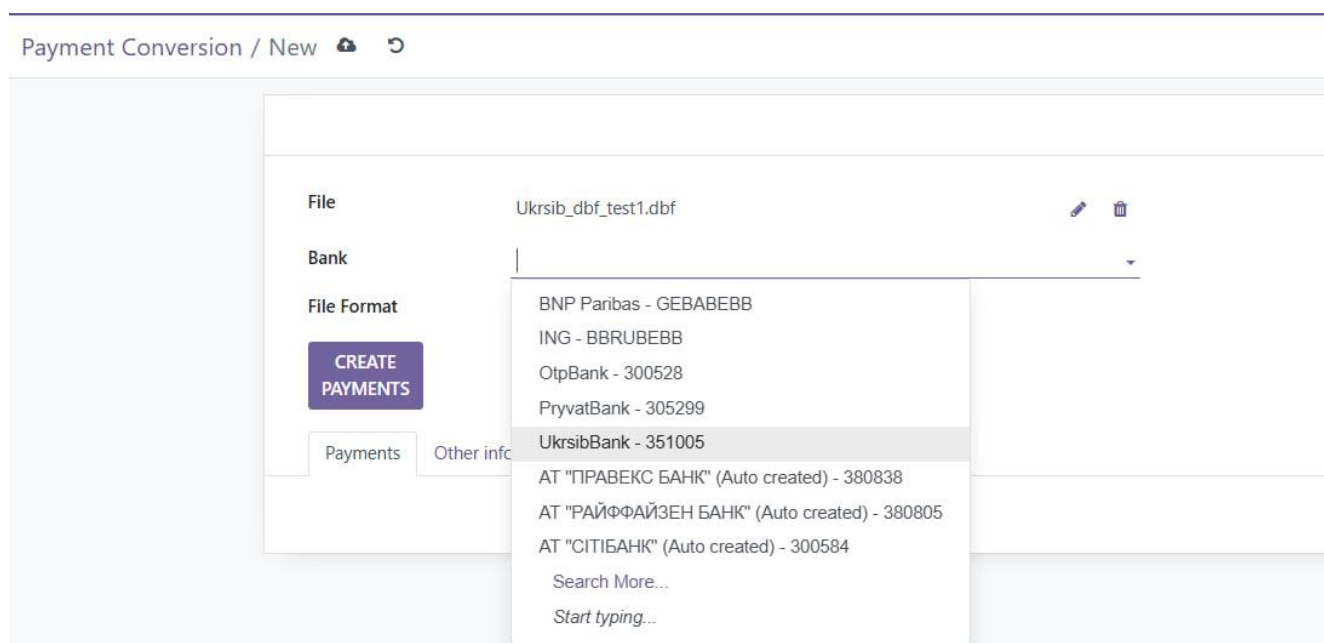


Рисунок 5.7 – Вибір банку

Після того, як банк буде обрано із доступних в запропонованому переліку, у

полі «File Format» буде відображено список форматів, що є доступними для конвертації.

Після внесення клієнтом всіх необхідних даних до всіх полів форми, він може натиснути кнопку «Create Payments», зачекати декілька секунд та в полі «Converted File» отримати конвертований файл, а також у вкладці «Payments» отримати структуровану таблицю банківських платіжних виписок, поля яких заповнені або за необхідності створені в базі клієнта. Надалі клієнт може використовувати файли в банкінгу в ODOO. Це відображено на рисунку 5.8.

The screenshot shows the 'Payment Conversion' interface in ODOO. At the top, there's a header bar with 'Payment Conversion', 'Converter', and 'Settings' tabs. Below it, a breadcrumb trail reads 'Payment Conversion / payment.conversion,10'. The main area is divided into two sections: 'Payments' (selected) and 'Other info'. The 'Payments' section displays a table of converted bank statements. The table has columns for various fields including bank details, account numbers, dates, and transaction descriptions. Each row is preceded by a green 'Create' button. The first row shows a transaction from 'PryvatBank' with a date of '03/29/2024' and a description in Ukrainian. Subsequent rows show transactions from 'КАЗНАЧЕЙС...' (Treasury) with similar details.

№	Bank St...	Counter...	O...	Account...	Docume...	Bank Na...	Inflow A...	Outflow...	Docume...	Organiz...	Account...	Content
1	BNK3/2024...	ПЕТРЕНКО ...	29654...	UA8030529...	456	PryvatBank	0.00	3,161.19	03/29/2024	Твоя Компа...	UA6530529...	Заробітна плата за другу пс сплачено в повному обсязі 29.03.2024
2	ОДВ, ГУ ДКС...	37993...	UA19899998...	455	КАЗНАЧЕЙС...	0.00	169.05	03/29/2024	Твоя Компа...	UA6530529...	101 податок на доходи з фіз	
3	ЗА ДЕБЕТУВ...	14360...	UA38305299...	37082NX1UY	PryvatBank	0.00	3.00	03/29/2024	Твоя Компа...	UA6530529...	101 Комісія за виконання платеж відкритою офертою банку І ГДБ.	
4	ОДВ, ГУ ДКС...	37993...	UA71899998...	454	КАЗНАЧЕЙС...	0.00	50.71	03/29/2024	Твоя Компа...	UA6530529...	101 військовий збір за бере:	
5	ЗА ДЕБЕТУВ...	14360...	UA38305299...	37082NX1UY	PryvatBank	0.00	3.00	03/29/2024	Твоя Компа...	UA6530529...	101 Комісія за виконання платеж відкритою офертою банку І ГДБ.	
6	ГОЛОВНЕ У...	44116...	UA88899998...	453	КАЗНАЧЕЙС...	0.00	743.81	03/29/2024	Твоя Компа...	UA6530529...	101 єдиний внесок, нарахов	

Рисунок 5.8 – Отримання конвертованих виписок

Отже, завдяки використанню модуля конвертації банківських виписок під назвою «Payment Conversion», клієнт зможе без зайвих труднощів отримати готовий результат, який повністю відповідатиме його потребам. Інтерфейс та функціональність модуля забезпечують простий, доступний та зручний процес роботи, при цьому дозволяючи досягти бажаного результату швидко та ефективно.

5.4 Використання системи автоматизації конвертації банківських виписок адміністратором

Після того, як буде встановлено програмний продукт, адміністратор, подібно до звичайного користувача, отримає доступ до нового пункту меню в інформаційній системі ODOO. У списку доступних для нього розділів з'явиться вкладка з модулем під назвою «Conversion Control Panel», як це відображено на рисунку 5.9.

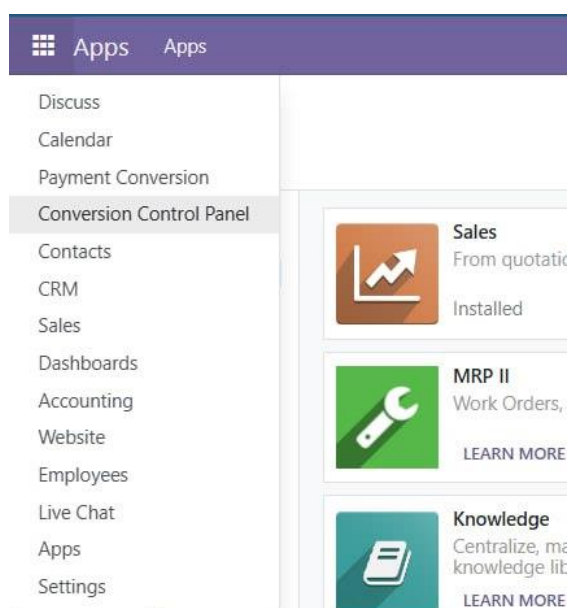


Рисунок 5.9 – Вкладка «Меню» із встановленим Conversion Control Panel

Ця панель служить для управління процесами конвертації, налаштування параметрів модуля та забезпечення контролю за його роботою.

За допомогою цього програмного продукту він може здійснювати адміністрування над клієнтами-користувачами. Інтерфейс адміністративної частини відображено на рисунку 5.10.

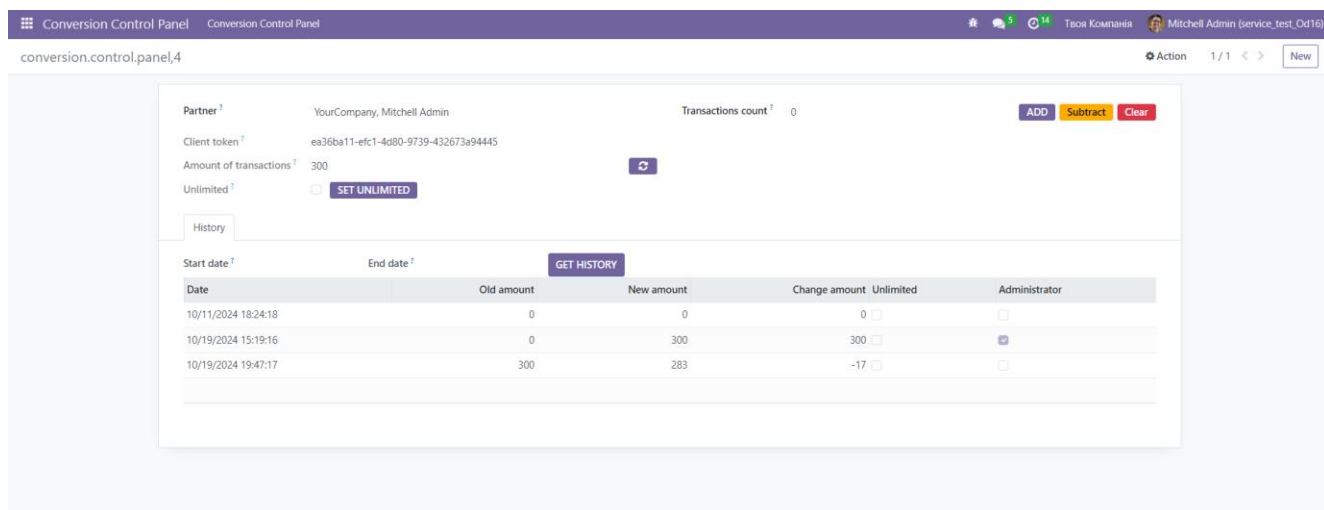


Рисунок 5.5 – Інтерфейс адміністратора

Цей модуль надає йому змогу обирати клієнта, переглядати доступну кількість конвертацій, керувати їхньою кількістю або взагалі очищувати, також надавати певним клієнтам безліміт, а також переглядати історію змін, структуровані таблиці, де вказано, коли і ким були внесені зміни.

Висновки до розділу 5

Користувачами системи автоматизації конвертації є клієнт та адміністратор. Система презентує кожному з них функціональний набір, який зробить взаємодію максимально лаконічною та зручною.

Дотримання рекомендованих системних вимог, необхідних для повноцінного функціонування програмного продукту, забезпечить його оптимальну продуктивність та стабільну роботу. Для коректного функціонування системи автоматизації конвертації в ODOO необхідно використовувати одну з підтримуваних операційних систем, забезпечити необхідне апаратне забезпечення, стабільне мережеве з'єднання та сучасний веб-браузер. Дотримання цих вимог

гарантує ефективну роботу програми та коректне виконання завдань.

Користувач має змогу безперешкодно встановити собі необхідний блок системи до своєї інформаційної системи ODOO. Після всіх виконаних кроків для встановлення, клієнтам та адміністраторам стає доступним необхідний функціонал.

Взаємодія користувача з модулем автоматизації конвертації банківських виписок здійснюється через спеціально розроблений інтерфейс. Тут клієнт має можливість переглянути історію всіх своїх попередніх конвертацій, а також заповнити форму для завантаження нових файлів, що потребують конвертації. Після завершення конвертації новий файл стане доступним для завантаження. Користувач зможе зберегти цей файл на комп'ютері та використовувати його відповідно до потреб. Користувач-адміністратор же, у свою чергу, може здійснювати адміністрування над користувачами-клієнтами, відповідно до необхідних йому можливостей.

Інтерфейс модуля – максимально простий, зручний і зрозумілий для всіх категорій користувачів. Як для клієнтів-користувачів, так і для адміністраторів взаємодія з програмним продуктом буде комфортною, що значно полегшуватиме роботу із системою ODOO.

6 РОЗРОБЛЕННЯ СТАРТАП-ПРОЄКТУ

На сьогодні розробки програмного забезпечення мають величезне значення для сучасного світу, охоплюючи економічну, соціальну, наукову і навіть культурну сфери. Окрім того, що вони кардинально полегшують життя людства, майже всі розробки є бізнес-проєктами, що є невід’ємною стороною життя суспільства в епоху фінансової конкуренції. Створення програмного забезпечення зараз не є перспективним без врахування його актуальності, оцінки конкурентоздатності, розрахунку фінансового потенціалу, визначення ймовірних користувачів тощо [31].

У цьому розділі визначено ідею стартап-проєкту за темою магістерської дисертації, здійснено аналіз ринкових можливостей запуску проєкту, охарактеризовано потенційних клієнтів, визначено конкурентоздатність, розроблено ринкову стратегію та маркетингову програму та визначено інші важливі аспекти стартапу [32].

6.1 Опис ідеї стартап–проєкту

Опис ідеї стартап-проєкту – це стислий, але при цьому чіткий та точний виклад концепції стартапу, його основної мети та завдань, цільової аудиторії та потенційної цінності тощо. Це ключовий документ, який допомагає розуміти суть проєкту, його унікальність, перспективність та важливість для ринку [32].

Опис ідеї стартап-проєкту за темою магістерської дисертації у першу чергу структурується на відображенні змісту самої ідеї, напрямках застосування та демонстрації вигоди для користувачів, які будуть послуговуватися програмним продуктом. Детальніше це можна побачити в таблиці 6.1.

Таблиця 6.1 – Опис ідеї стартап-проєкту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Конвертація платіжних банківських виписок для інформаційної системи ODOO	1. Забезпечення автоматичного обміну платіжними виписками	1. Ефективність процесу (автоматичний обмін платіжними виписками дозволяє користувачам ефективно та швидко отримувати актуальну фінансову інформацію) та мінімізація помилок (автоматизований процес зменшує ризик помилок, які можуть виникнути при ручному введенні або обробці виписок).
	2. Спрощення врахування змін у форматах виписок	2. Гнучкість (зміни у форматах виписок від різних банків часто можуть бути викликані оновленнями або регуляторними вимогами. Спрощення врахування цих змін забезпечує гнучкість і зручність використання системи для користувачів) та зекономлений час (користувачі можуть витратити менше часу на вирішення проблем, пов'язаних із змінами у форматах виписок, що дозволяє їм більше зосередитися на своїх основних справах).

Також при описі ідеї проєкту варто визначати характеристики, порівняно з аналогічними продуктами, роблячи акцент на сильних сторонах, зазначаючи нейтральні та демонструючи можливі слабкі, як це відображено в таблиці 6.2.

Таблиця 6.2 – Визначення сильних, слабких та нейтральних характеристик

№ п/п		Потенційні товари/концепції конкурентів	
		Проєкт магістерської	Xero
1	W слабка сторона	Призначений насамперед для конвертації платіжних банківських виписок	Широкофункціональний інструмент
2	N нейтральна сторона	Надання функціоналу для імпорту та експорту платіжних виписок	

Закінчення таблиці 6.2.

3	Сильна сторона	Простий в користуванні	Потребує високого рівня технічних знань користувача
4		Доступна ціна	Потребує значних фінансових можливостей у компанії
5		Конвертація виписок з різних банків і різних форматів в універсальний формат	Має обмеження щодо певних форматів та банків
6		Легке та відносно швидке додавання та оновлення обробок для форматів виписок	Додавання нових форматів потребує значної кількості часу та об'ємної роботи [33]

Для розуміння перспектив та переваг далі необхідним є здійснення технологічного аудиту ідеї.

6.2 Технологічний аудит ідеї стартап–проекту

Технологічний аудит перевіряє, чи відповідає розроблений продукт сучасним вимогам [34]. Відображення технологічної здійсненності ідеї продемонстровано в таблиці 6.3.

Таблиця 6.3 – Технологічна здійсненність ідеї проекту

Но п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Інтуїтивний і зручний інтерфейс для користувачів	Фреймворк ODOO	Наявна	Доступна безкоштовно
2	Обробка та конвертація виписок	Python	Наявна	Доступна безкоштовно

Закінчення таблиці 6.3.

3	База даних	PostgreSQL	Наявна	Доступна безкоштовно
4	Серверна частина модуля конвертації	Flask	Наявна	Доступна безкоштовно

Обрані технології для розробки системи автоматизації конвертації банківських виписок є доступними та досяжними для втілення, відповідно, проєкт реалізувати можливо.

Далі для запуску стартапу необхідно проаналізувати ринкові можливості.

6.3 Здійснення аналізу ринкових можливостей запуску стартап–проєкту

Вивчення та дослідження ринкового попиту варто розпочати з проведення попередньої оцінки потенційного ринку реалізації продукту. Зведені показники та характеристики продемонстровано в таблиці 6.4.

Таблиця 6.4. Попередня характеристика потенційного ринку стартап–проєкту

Но п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	5-10 основних компаній (з дотичним функціоналом, залежно від регіону)
2	Загальний обсяг продаж, грн/ум.од	>100000 грн
3	Динаміка ринку (якісна оцінка)	стабільне зростання
4	Наявність обмежень для входу	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Немає

Закінчення таблиці 6.4.

6	Середня норма рентабельності в галузі (або по ринку), %	25-30
---	---	-------

Отже, можна зазначити, що явних перешкод для входження продукту на ринок немає. ODOO є популярною відкритою інформаційною системою для управління різними бізнес-процесами, відповідно, створення програмного продукту для збільшення функціоналу системи є перспективним [37].

Далі треба визначити цільову аудиторію проєкту, охарактеризувати потенційних користувачів, як це здійснено в таблиці 6.5.

Таблиця 6.5. Характеристика потенційних клієнтів стартап-проєкту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Конвертація платіжних банківських виписок в універсальний формат	Компанії, що мають справи з клієнт-банкінгом; бухгалтерія	Особливості купівлі: великі компанії заключають довготривалі договори, компанії меншого розміру надають перевагу короткотривалим договорам або договорам з меншою кількістю доступних конвертацій. Для компаній з більшим фінансовим обігом, які постійно взаємодіють з різними банками, необхідним є стабільний доступ до можливостей конвертації, тоді як компаніям з меншими фінансовим обігом достатньою є кількість конвертацій у межах фінансового ліміту.	стабільність роботи, універсальність формату виписок, невисока ціна, наявність пробного періоду, наявність документації

Після аналізу потенційної клієнтської бази варто врахувати фактори загроз, визначаючи і сам зміст загрози, і можливу реакцію підприємства на неї. Це підсумовано в таблиці 6.6.

Таблиця 6.6. Фактори загроз

Но п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Зміна формату файлів виписок банком	При зміні без попереднього поінформування формату виписок банку виникає потреба в оперативному корегуванню та зміні обробок.	Зв'язок з адміністратором щодо виникнення проблем з форматом
2	Обмеженість доступних форматів і банків	При відсутності певного банку чи формату у списку доступних/наявних виникає потреба в розширенні або додаванні обробок конвертації.	Запит на додавання необхідного банку чи формату

Також варто врахувати фактори можливостей, здійснити їхній аналіз. Це дозволить краще розуміти потреби ринку та оцінювати можливий попит в окремих користувачів або й цілих компаній на програмний продукт.

Технічний розвиток обумовлює зростання інтересу до автоматизації бізнес-процесів. Варіації факторів можливостей, їхнього змісту та можливих реакцій відображено в таблиці 6.7.

Таблиця 6.7. Фактори можливостей

Но п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Можливість розширення бази доступних банків та форматів	За необхідності можна розширювати перелік доступних банків та форматів за запитами користувачів	Зміна чи доповнення форматів та банків можлива та виконується за запитом компанії.

Закінчення таблиці 6.7.

2	Використання універсального формату	Забезпечення стандартизації та сумісності даних, що спрощує подальший аналіз та звітність	Полегшене імпортування транзакцій в систему ODOO.
---	-------------------------------------	---	---

Далі необхідно здійснити ступеневий аналіз конкуренції за ринком, що є методичним підходом до оцінки рівня конкуренції в певній галузі чи ринковому сегменті, який розглядає конкурентів на різних рівнях впливу [35]. Метою такого аналізу стало визначення того, як кожен із гравців ринку впливає на споживачів, продукцію чи послуги, ціни, інновації та інші ключові аспекти тощо. Здійснення такого аналізу у межах стартапу відображено в таблиці 6.8.

Таблиця 6.8. Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	У чому проявляється така характеристика	Вплив на діяльність компанії (можливі дії фірми, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції – монополія/олігополія/ монополістична/чиста	чиста	Прямі договори з стартапами, презентація продукту на виставках
2. За рівнем конкурентної боротьби – локальний/національний/...	національний	Публікація статей на міжнародних сайтах
3. За галузевою ознакою – міжгалузева/ внутрішньогалузева	внутрішньогалузева	Розвивати напрямки, нерозвинуті конкурентами
4. Конкуренція за видами товарів: – товарно–родова – товарно–видова – між бажаннями	товарно–видова	Розповідати про свої переваги перед конкурентом у цій галузі

Закінчення таблиці 6.8.

5. За характером конкурентних переваг – цінова / нецінова	нецінова	Надання функцій, які не надають конкуренти
6. За інтенсивністю – марочна/не марочна	марочна	Надання функцій, які не надають конкуренти

Завдяки проведеному ступеневому аналізу, було поглиблено розуміння ринкових позицій з ціллю визначення власної позиції щодо конкурентів, визначено ті переваги, які надає програмний продукт, та здійснено оптимізацію стратегії.

Для додаткової оцінки конкурентного середовища галузі та виявлення тих факторів, які впливають на прибутковість бізнесу в межах певного сегменту ринку, було проаналізовано перспективи за моделлю п'яти сил Портера [38], що відображено в таблиці 6.9.

Таблиця 6.9. Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари–замінники
	Відсутні	Хмарна облікова система Хего	Відсутність залежності від постачальників	Контроль якості	Відсутність аналогічна продуктів
Висновки:	Конкурентна боротьба відсутня	Є можливості виходу на ринок, оскільки існуючі рішення не надають потрібних переваг	Автономія від постачальників, зумовлена самостійністю продукту	Клієнти диктують вимоги згідно з умовами експлуатації	Товари-замінники не створюють перешкод роботі на ринку

Аналіз за Портером допомагає в розумінні сильних та слабких сторін

конкретної бізнес-ідеї, а також сприяє можливості знайти способи успішної конкуренції на ринку. Цей аналіз використовується для формування стратегії компанії або стартапу, щоб визначити, як залишатися конкурентоспроможним і отримувати стабільний дохід.

Далі необхідним є обґрунтувати фактори конкурентоспроможності, які здійснюються завдяки аналізу провідних характеристик та переваг продукту і тієї послуги, яку він надає, тобто, чим програмний продукт є вигіднішим, цікавішим та привабливішим для клієнтів, ніж пропозиції можливих чи потенційних клієнтів [39]. Це обґрунтування відображено в таблиці 6.10.

Таблиця 6.10. Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проєктів значущим)
1	Відносно низька собівартість	Вартість використання Хего може бути високою для деяких малих підприємств.
2	Простота роботи з програмою	Існуючий програмний комплекс Хего вимагає суттєвих специфічних знань
3	Обмежені можливості регіональної адаптації	Хего може бути менш ефективним у певних регіонах та для певних податкових систем.

Правильне обґрунтування конкурентної спроможності дає змогу краще позиціонувати продукт та залучати інвесторів.

При цьому вартим уваги є порівняльний аналіз сильних сторін програмного продукту та його слабких боків. Розгляд цього представлено в таблиці 6.11.

Таблиця 6.11. Порівняльний аналіз сильних та слабких сторін

No п/п	Фактор конкурентоспроможності	Бал и 1– 20	Рейтинг товарів–конкурентів у порівнянні з модулем конвертації платіжних банківських виписок (даним продуктом)						
			–3	–2	–1	0	1	2	3
1	Використання універсального формату	20	+						
2	Інтерфейс користувача	15		+					
3	Вартість користування	17		+					
4	Застосування сучасних засобів програмування (Python, PostgreSQL)	15				+			
5	Одночасна робота з різними форматами та банками.	19		+					

Порівняння мінусів та плюсів дає перспективи в баченні реальних перспектив проекту та дає змоги вдосконалювати продукт за необхідності.

Оцінка сильних та слабких сторін, можливостей та загроз наведена в таблиці 6.12.

Таблиця 6.12. SWOT–аналіз стартап–проекту

Сильні сторони: Впровадження універсального формату. Робота з різними форматами та банками. Застосування сучасних засобів програмування (Python, PostgreSQL). Зручний інтерфейс.	Слабкі сторони: Перелік форматів та банків може потребувати доповнення.
Можливості: Відсутність повноцінних альтернатив. Можливість подальшого супроводження та розвитку проекту.	Загрози: Відсутність поінформованості можливих клієнтів щодо новоствореного продукту.

Під час реалізації стартап-проєкту необхідно враховувати альтернативні можливості впровадження на ринку, відстежувати підходи та способи, що допоможуть визначити найбільш ефективний шлях досягнення цільової аудиторії, закріплення на ринку та, відповідно, отримання продукту. Альтернативи реалізації на ринку здійсненої розробки відображено в таблиці 6.13.

Таблиця 6.13. Альтернативи ринкового впровадження стартап-проєкту

No п/ п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Орієнтація програмного забезпечення щодо застосування в компаніях/бухгалтеріях, що користуються клієнт-банкінгом	90 %	1 рік

Після проведення всіх аналізів та врахування всіх можливостей далі необхідно розробити ринкову стратегію.

6.4 Розроблення ринкової стратегії стартап-проєкту

Без розробки чіткої ринкової стратегії навіть технічно досконалий стартап-проєкт може не досягти комерційного успіху, тому її створення є необхідним етапом для впровадження. Вона може охоплювати аналіз ринку, визначення цільової аудиторії проєкту, позиціонування продукту, його цінову політику, маркетингові інструменти, канали дистрибуції тощо.

Метою ринкової стратегії є гарантування ефективного виходу на ринок, забезпечення потреби споживачів і досягнення сталого розвитку бізнесу.

Перш за все треба обрати цільову групу потенційних користувачів. Для розробленого стартап-проєкту це продемонстровано в таблиці 6.14.

Таблиця 6.14. Вибір цільових груп потенційних споживачів

№ п/п	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
1	Компанії/бухгалтерії, що користуються клієнт-банкінгом	Готові	Високий	Відсутня	Просто

Розроблення ринкової стратегії вимагає від реалізаторів стартапу визначення базової стратегії розвитку, як це зазначено в таблиці 6.15.

Таблиця 6.15. Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентос– проможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	Орієнтація модуля конвертації на компанії/бухгалт ерії, що користуються клієнт-банкінгом	Стратегія концентрованого маркетингу	Компанії/бухгалтерії, що користуються клієнт-банкінгом потребують конвертації платіжних банківських виписок в універсальний формат	Стратегія спеціалізації (спирається на диференціаці ю)

Без визначення базової стратегії конкурентної поведінки стартап ризикує втратити фокус та не зайняти свою нішу на ринку. Це ключовий інструмент для тих, хто прагне досягнення довгострокового успіху. Конкретне визначення цього продемонстровано в таблиці 6.16.

Таблиця 6.16. Визначення базової стратегії конкурентної поведінки

N о п/ п	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	Так	Шукати нових споживачів	Ні	Стратегія заняття конкурентної ніші

Далі необхідним є визначити стратегію позиціонування, яка полягає у визначенні та комунікації ключового місця, яке продукт та послуга, що здійснюється ним, мають займати у свідомості цільових користувачів порівняно з конкурентами. Це допомагає у визначенні того, чому клієнти обирають саме цю пропозицію, і створює відповідні асоціації, які виділяють продукт серед альтернатив. Демонстрація цього відображена в таблиці 6.17.

Таблиця 6.17. Визначення стратегії позиціонування

N о п/ п	Вимоги цільової аудиторії до товару	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проєкту	Вибір асоціацій, які мають сформувати комплексну позицію власного проєкту
1	Універсальний формат виписки, стабільність роботи, невисока ціна, наявність пробного періоду, наявність документації	Стратегія спеціалізації (спирається на диференціацію)	Стабільність роботи та гнучкість в налаштуваннях системи забезпечується її модульністю.	Універсальний формат, стабільність роботи, наявність пробного періоду

Без чітко визначеної стратегії позиціонування стартапу складно привернути увагу клієнтів до продукту і виділитися серед конкурентів, що зменшує його шанси на успіх.

6.5 Розроблення маркетингової програми стартап–проекту

Важливим кроком для реалізації стартапу є розроблення маркетингової програми. Вона дає змогу визначити, як проєкт позиціонуватиметься на ринку, як досягати конкурентних переваг та як залучати клієнтів. Без продуманої маркетингової програми стартап може мати труднощі з просуванням та виживанням на ринку конкуренції.

Під час розробки програми варто визначити ті ключові переваги концепції продукту, які він має перед можливими конкурентами. Це визначення здійснено в таблиці 6.18.

Таблиця 6.18. Визначення ключових переваг концепції потенційного товару

Но п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Універсальність	Використання універсального формату виписок	Хмарна облікова система Хего не надає конвертації виписок в універсальному форматі.
2	Простота використання	НЕ потребує спеціальних знань для роботи	Програмний комплекс Хего потребує специфічних знань для роботи.

Далі необхідно здійснити характеристику трьох рівнів моделі продукту (таблиця 6.19). Це необхідно для глибшого розуміння потреб споживачів, ефективного позиціонування продукту, формування конкурентної переваги,

оптимізації бізнес-процесів, адаптації до змін ринку тощо.

Таблиця 6.19. Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
I. Товар за задумом	Модуль конвертації платіжних банківських виписок для інформаційної системи ODOO.		
II. Товар у реальному виконанні	Властивості/ характеристики	М/Нм*	Вр/Тх /Тл/Е/Ор*
	можливість оптимізації витрат часу	М	Тл
	відповідність актуальним технологіям	Нм	Тх
	Відповідає міжнародним вимогам ISO 20022.		
	Пакування: готовий до використання файл з інсталяцією		
	Марка: PaymentConversion		
III. Товар із підкріпленням	Потенційний користувач може ознайомитись з поточним товаром з наукових публікацій у наукометричних та фахових виданнях, а також наукових конференцій, на яких була представлена інформація про даний продукт		
* М/Нм – монотонні/немонотонні; Вр/Тх/Тл/Е/Ор – вартісні/ технічні/ технологічні/ ергономічні/ органолептичні [36]			

Товар буде убезпечено від неправомірних дій з копіюванням захистом інтелектуальної власності.

Наступним кроком при розробці маркетингової програми є визначення можливих меж встановлення ціни на продукт та послугу. Необхідно враховувати

фактори, що впливають на нижню та верхню межу їхньої ціни (таблиця 6.20).

Таблиця 6.20. Визначення меж встановлення ціни

№ п/п	Рівень цін на товари– замінники	Рівень цін на товари– аналоги	Рівень доходів цільових споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	0 грн	0 грн	Від 80400 грн	1200-8000 грн (залежно від пакету може варіюватися)

З метою забезпечення ефективної реалізації продукту, стабільного прибутку, зменшення витрат тощо необхідно формувати систему збуту. Для стартапу це втілено в таблиці 6.21.

Таблиця 6.21. Формування системи збуту

№ п/п	Особливості закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальн а система збуту
1	Клієнт повинен надаватися в режимах “тріал”, “лімітований” та “повний”. Сплата має проходити через банк.	Простота у встановленні програмного продукту, наявність документації по роботі з програмним забезпеченням	4: Розробник даного продукту – ERP ODOO – Користувач.	Проводити своїми силами.

Завершенням розробки маркетингової програми є формування концепції маркетингових комунікацій (таблиця 6.22), що полягає в процесі розробки стратегії, яка використовується компанією, та підходів для взаємодії з цільовою аудиторією. Завдяки цьому можна збільшити продажі, підвищити попит на пропонуванний продукт.

Таблиця 6.22. Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікації, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Придбання програмного комплексу через офіційний сайт або зв'язавшись з представником	Веб-сайти, менеджери	Універсальність, модульність, гнучкість налаштувань, доступність	Продемонструвати універсальність програмного продукту, зручність, відносно низьку вартість за весь обсяг послуг.	Конвертація платіжних банківських виписок в інформаційну систему ODOO

Ефективні маркетингові комунікації є важливими для стартап-проєкту, адже сприяють створенню унікальної ідентичності, допомагають побудувати довготривалі стосунки з користувачами, а також збільшують охоплення аудиторії та значно збільшують прибуток.

Висновки до розділу 6

Розробка стартап-проєктів важлива не тільки для розвитку бізнесу, але й для загального прогресу суспільства. Вони є джерелом інновацій, рушієм економіки та засобом покращення життя мільйонів людей.

Створення та аналіз стартап-проєкту, пов'язаного з автоматизацією конвертації банківських виписок на основі інформаційної системи ODOO, дали можливість визначити, що проєкт має значний потенціал та перспективність у

занятті своєї ніші на фінансовому ринку, оскільки має багато переваг, які є недоступними можливим конкурентам. Варто відзначити, що прямих конкурентів, які б надавали абсолютно аналогічні послуги, у програмного продукту немає. Це також є позитивним аспектом реалізації. Продуктивність стартапу опирається на ринковий попит продукту, його конкурентні переваги, адже має унікальну функціональність, а також можливості масштабування та продуманість усіх ризиків та викликів, які можуть стояти на шляху виходу на ринок. Можливості для зростання є значними, особливо у сфері автоматизації фінансових процесів, що зараз на піку популярності. Отже, стартап можна вважати успішним та перспективним.

ВИСНОВКИ

У межах магістерської дисертації виконано розробку системи автоматизації конвертації банківських виписок на прикладі інформаційної системи ODOO, що дало змогу здійснити забезпечення конвертації платіжок у форматі, який є зрозумілим для системи, та спростувати можливу необхідність внесення змін до форматів виписок.

Було проаналізовано та досліджено спеціалізовану літературу та відповідну документацію, що присвячені різним віхам визначеного напрямку роботи, та проаналізовано існуючі програмні продукти та системи зі схожим або дотичним функціоналом. Це дало змогу підсумувати потреби та очікування ймовірних користувачів, проаналізувати позитивні та негативні сторони існуючих сервісів і внаслідок цього сформувати бачення щодо вимог до системи автоматизації конвертації.

Для реалізації системи використано загальні наукові методи досліджень та спеціалізовані підходи і методології при розробці програмного продукту. Важливим було прототипування сервісу та врахування досвіду експертів у галузі.

Для розробки було обрано такі засоби, як мова програмування Python, фреймворки ODOO та Flask, СУБД PostgreSQL, а також середовище розробки PyCharm. Кожен із перерахованих інструментів був обраний з урахуванням його доцільності та ефективності для досягнення найкращих результатів.

Здійснено проектування структури системи автоматизації конвертації, основними складовими якої є модуль конвертації, клієнтська та адміністративна частини. Поєднання структурних елементів в ODOO здійснює забезпечення автоматизації обміну банківськими виписками, покращує роботу з ними, сприяє легкому адмініструванню функціонування системи та роботи клієнтів з нею, та, що особливо важливо, спрощує врахування змін та необхідності внесення коректив у форматах виписок у інформаційній системі ODOO.

Відповідно до цієї композиції за допомогою обраних методів, прийомів та

засобів реалізації було розроблено програмний продукт, що пройшов тестування та продемонстрував, що він готовий до реалізації на ринку та експлуатації. За темою дисертації було створено стартап-проект, який підтвердив перспективність розробленої системи автоматизації. Результати роботи можна використовувати в компаніях, що послуговуються ODOO, та на підприємствах, які користуються клієнт-банкінгом.

Як можливий напрям подальшої роботи над темою та можливості розвитку результатів можна обрати курс адаптації та створення системи автоматизації конвертації для інших систем управління підприємством. Можна використовувати базу реалізованої серверної частини з метою спрощення врахування змін у форматах платіжок і для цих ERP-систем.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Odoo Documentation. User Docs, Install and Maintain, Developer, Contributing. URL: <https://www.odoo.com/documentation/16.0/> (дата звернення: 27.09.2024).
2. Форум Odoo. Спільнота. Допомога. Теми. URL: https://www.odoo.com/uk_UA/forum/dopomoga-1 (дата звернення: 26.10.2024).
3. Заступайло М. І. Модуль обміну даними з банками інформаційної системи ODOO. *Сучасні проблеми наукового забезпечення енергетики*: матеріали XX Міжнар. наук.-практ. конф. молод. вчених і студ. Київ: КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2023. Т. 2. С. 132–133.
4. Заступайло М. І. Автоматизація конвертації банківських виписок для інформаційних систем підприємств. *Сучасні проблеми наукового забезпечення енергетики*: матеріали XXI Міжнар. наук.-практ. конф. молод. вчених і студ. Київ: КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2024. Т. 2. С. 230–232.
5. Документація та підтримка Bank2CSV. URL: <http://www.propersoft.net/support> (дата звернення: 13.09.2024).
6. OFX Converter. URL: <http://www.ofxconverter.com/> (дата звернення: 13.09.2024).
7. Financial Reporting Software. *Capterra*. URL: <https://www.capterra.com/financial-reporting-software/#top-20> (дата звернення: 16.09.2024).
8. What is ERP. URL: <https://www.sap.com/products/erp/what-is-erp.html> (дата звернення: 16.10.2024).
9. Упровадження стандарту ISO 20022. URL: <https://bank.gov.ua/ua/payments/project-iso20022> (дата звернення: 13.09.2024).
10. Financial Information Exchange (FIX). URL: <https://www.investopedia.com/terms/f/financial-information-exchange.asp> (дата звернення: 14.09.2024).
11. Odoo Community Association, OCA. URL: <https://odoo-community.org> (дата звернення: 17.11.2024).

12. The official home of the Python Programming Language. URL: <https://www.python.org> (дата звернення: 03.10.2024).
13. Allen B. Downey. Think Python. How to think like a computer scientist. Published O'Reilly Media, Inc. 2015. 290 с.
14. Sweigart A. Automate the Boring Stuff with Python. Practical Programming for Total Beginners: 2nd edition. NoStarchPress, 2019. 592 p.
15. Офіційний веб-сайт Xero. URL: <https://www.xero.com/> (дата звернення: 13.09.2024).
16. Xpath Expression. *ScienceDirect. Journals & Books*. URL: <https://www.sciencedirect.com/topics/computer-science/xpath-expression> (дата звернення: 13.09.2024).
17. BizConnect. URL: <https://bizconnectus.com> (дата звернення: 14.09.2024).
18. Бази даних PostgreSQL. Посібник користувача. URL: http://ukrhosting.ua/bazi_danih_postgresql-p-263951.html (дата звернення: 15.11.2024).
19. Pycharm: середовище розробки для Python. URL: <https://blog.desdelinux.net/uk/pycharm-un-entorno-de-desarrollo-para-python/> (дата звернення: 17.10.2024).
20. Odoo – LinkedIn. URL: <https://be.linkedin.com/company/odoo> (дата звернення: 16.11.2024).
21. СУБД: які бувають, як вибрати. URL: <https://highload.today/uk/subd-yaki-buvayut-yak-vibrati/> (дата звернення: 18.09.2024).
22. PgAdmin.Documentation. URL:<https://www.pgadmin.org/docs/> (дата звернення: 18.09.2024)
23. Flask vs Django. Обираємо фреймворк для веб-розробки. URL: <https://alternativescience.net/programming/140-flask-vs-django-obyayemo-frejmvork-dlya-veb-rozrobky/> (дата звернення: 16.09.2024).
24. Корисні інструменти для Python-розробників. Українські програмісти – про те, що спрощує їм роботу. URL: <https://dou.ua/lenta/articles/tools-for-python->

- developers/ (дата звернення: 14.10.2024).
25. Installing Odoo. ODOO docs. URL: <https://www.odoo.com/documentation/16.0/administration/install.html> (дата звернення: 21.09.2024).
 26. Odoo Studio Book – Python Code. URL: <https://www.cybrosys.com/odoo/odoo-books/odoo-15-studio/automation/python-code/> (дата звернення: 14.10.2024).
 27. Що таке JSON. Усе про цей формат передачі даних в інтернеті. URL: <https://apix-drive.com/ua/blog/useful/scho-take-json> (дата звернення: 17.09.2024).
 28. Офіційний сайт «Укрсиббанк». URL: <https://ukrsibbank.com> (дата звернення: 13.09.2024).
 29. API Endpoints – What Are They? Why Do They Matter? URL: <https://smartbear.com/learn/performance-monitoring/api-endpoints/> (дата звернення: 16.09.2024).
 30. Jorge Martinez. 10 Best Bank Statement Converter Software in 2024. *DocuClipper*. URL: <https://www.docuclipper.com/blog/best-bank-statement-converter-software> (дата звернення: 02.10.2024).
 31. King B. Bank 4.0: Banking Everywhere, Never at a Bank: 1st edition. Wiley, 2017. 384 p.
 32. Russell S., Norvig P. Artificial Intelligence: A Modern Approach, 4th US ed. US, Global. URL: <https://aima.cs.berkeley.edu> (дата звернення: 20.09.2024).
 33. Kleppmann M. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. 2017. 614 p.
 34. Ries E. The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses. 1st Edition. Crown Currency, 2011. 336 p.
 35. Thiel P. Zero to One: Notes on Startups, or How to Build the Future. 1st edition. Crown Currency, 2014. 224 p.
 36. Стартап. Опис трьох рівнів товару. URL: <https://studfile.net/preview/5992853/page:7/> (дата звернення: 13.11.2024).

37. Разумовський Ю. Чим цікава ERP система Odoo українському бізнесу і як допомагає країні. URL: <https://cases.media/en/article/chim-cikava-erp-sistema-odoo-ukrayinskomu-biznesu-i-yak-dopomagaye-krayini> (дата звернення: 16.11.2024).
38. Cooper B., Vlaskovits P. The Lean Entrepreneur: How Visionaries Create Products, Innovate with New Ventures, and Disrupt Markets. John Wiley & Sons Inc; Original edition, 2013. 256 p.
39. Aulet B. Disciplined Entrepreneurship: 24 Steps to a Successful Startup. 1st edition. Wiley, 2013. 288 p.
40. Horowitz B. The Hard Thing About Hard Things: Building a Business When There Are No Easy Answers. Harper Business, 2014. 304 p.

ДОДАТОК А

Серверна частина системи автоматизації конвертації
банківських виписок на прикладі інформаційної системи ODOO

Текст програмного модуля

УКР.НТУУ''КПІ ім.. Ігоря Сікорського''_ІАТЕ_ЦТЕ_ТР-31мп

Аркушів 11

Київ – 2024

#Файл main.py містить основну логіку роботи модуля конвертації

#1. Імпорт і встановлення залежностей

```
from flask import Flask, request, jsonify
```

```
import base64
```

```
from database import DB
```

```
from bank_mapping import get_bank_processor
```

```
from data_structures import Transaction
```

```
from data_classes import asdict
```

```
import json
```

```
from typing import List
```

```
import chardet
```

```
db = DB()
```

```
app = Flask(__name__)
```

#2. Основні API маршрути

```
@app.route('/generate-token', methods=['GET'])
```

```
def generate_token():
```

```
    try:
```

```
        client_token = db.generate_token()
```

```
    except Exception as ex:
```

```
        return jsonify({'error': str(ex)}), 400
```

```
    print(client_token)
```

```
    return jsonify({'client_token': client_token})
```

```
@app.route('/get_available_formats', methods=['POST'])
```

```

def get_available_formats():
    data = request.get_json()
    bank = data.get('bank')
    formats = None
    try:
        if bank:
            bank_processor = get_bank_processor(bank)
            formats = bank_processor.get_available_formats()
    except Exception as ex:
        return jsonify({'error': str(ex)}), 400

```

```

def detect_encoding(file: bytes) -> str:
    return chardet.detect(file)['encoding']

```

```

@app.route('/payment-conversion', methods=['POST'])
def convert_payment():
    data = request.get_json()
    file_str = data.get('file')
    file_name = data.get('file_name')
    file_format = data.get('file_format')
    bank = data.get('bank')
    client_token = data.get('client_token')

    file_data = base64.b64decode(file_str.encode('utf-8'))
    encoding = detect_encoding(file_data)
    file_str = file_data.decode(encoding)

```

```

if not db.token_exists(client_token):
    return jsonify({'error': 'Invalid token'}), 401

if not db.transactions_exists(client_token):
    return jsonify({'error': 'No transactions found'}), 404

if file_data is None:
    return jsonify({'error': 'No file provided'}), 400

try:
    bank_processor = get_bank_processor(bank)
    amount_of_transactions = db.get_amount_of_transactions(client_token)
    transactions: List[Transaction] = bank_processor.parse(file_str, file_format,
amount_of_transactions)
    except Exception as ex:
        return jsonify({'error': str(ex)}), 400
    used_transactions = len(transactions)

    json_transaction = json.dumps([asdict(transaction) for transaction in transactions],
indent=4, ensure_ascii=False)
    converted_file = base64.b64encode(json_transaction.encode('utf-8')).decode('utf-
8')
    converted_file_name = file_name.split(".")[0] + ".json"

    return jsonify({
        'converted_file': converted_file,

```

```

        'converted_file_name': converted_file_name,
        'used_transactions': used_transactions
    })

```

```

@app.route('/decrease-conversions', methods=['POST'])
defdecrease_conversions():
    data = request.get_json()
    client_token = data.get('client_token')
    used_transactions = data.get('used_transactions')
    db.decrease_token_transactions(client_token, int(used_transactions))
    return jsonify({'success': 'done'}), 200

```

```

@app.route('/get-amount-of-transactions', methods=['POST'])
defget_amount_of_transactions_route():
    data = request.get_json()
    client_token = data.get('client_token')

    ifclient_tokenisFalseornotdb.token_exists(client_token)
    ornotdb.transactions_exists(client_token):
        return jsonify({'amount_of_transactions': 0})

    amount_of_transactions = db.get_amount_of_transactions(client_token)
    return jsonify({'amount_of_transactions': amount_of_transactions}), 200

```

```

#Файл database.py містить клас для взаємодії модуля з базою даних
import psycopg2
import os
from dotenv import load_dotenv
import uuid
load_dotenv()

class DB:
    def __init__(self):
        self.connection = psycopg2.connect(
            host=os.environ.get('HOST'),
            user=os.environ.get('USER'),
            password=os.environ.get('PASSWORD'),
            database=os.environ.get('DB_NAME')
        )
        self.connection.autocommit = True
        self.cursor = self.connection.cursor()

    def token_exists(self, token):
        self.cursor.execute("SELECT COUNT(*) FROM tokens_transactions WHERE
token = %s", (token,))
        return self.cursor.fetchone()[0] > 0

    def transactions_exists(self, token):
        self.cursor.execute("SELECT COUNT(*) FROM tokens_transactions WHERE
token = %s;", (token,))
        amount_of_transactions = self.cursor.fetchone()[0]

```

```
return amount_of_transactions > 0
```

```
def get_amount_of_transactions(self, token):
    self.cursor.execute("SELECT transactions FROM tokens_transactions WHERE
token = %s;", (token,))
    amount_of_transactions = self.cursor.fetchone()[0]
    return amount_of_transactions
```

```
def decrease_token_transactions(self, token, decrease_value):
    self.cursor.execute("SELECT transactions FROM tokens_transactions WHERE
token = %s;", (token,))
    current_transactions = self.cursor.fetchone()[0]
    new_transactions = max(current_transactions - decrease_value, 0)
    self.cursor.execute("UPDATE tokens_transactions SET transactions = %s
WHERE token = %s;",
                        (new_transactions, token))
    a = self.cursor.fetchone()
```

```
def update_token_transactions(self, token, amount_of_transactions):
    self.cursor.execute("UPDATE tokens_transactions SET transactions = %s
WHERE token = %s;",
                        (amount_of_transactions, token))
```

```
def generate_token(self):
    token = str(uuid.uuid4())
    while self.token_exists(token):
        token = str(uuid.uuid4())
    self.cursor.execute("INSERT INTO tokens_transactions (token, transactions)
VALUES (%s, %s);",
```

```

        (token, 0))

    return token

    def __del__(self):
        if self.connection:
            self.cursor.close()
            self.connection.close()
            print("[INFO] Connection closed.")

```

#Файл bank_mapping.py містить в собі функціонал, який шукає необхідні правила обробки файла відносно банку

```

def get_bank_processor(bank_code):
    bank_processor = banks.get(bank_code, None)
    if bank_processor:
        return bank_processor
    raise ValueError("Bank not supported")

```

#Файл ukrsib.py містить в собі приклад правил обробки виписок з банку "Укрсиббанк"

```

from data_structures import Transaction
import csv
import io

class UkrsibParser:
    def __init__(self):
        self.formats = {'csv': 'CSV'}

```

```
def get_available_formats(self):
    return self.formats
```

```
    @staticmethod
    def get_csv_delimiter(csv_string):
        sniffer = csv.Sniffer()
        dialect = sniffer.sniff(csv_string)
        return dialect.delimiter
```

```
def parse(self, file, file_format, amount_of_transactions):
    if file_format == 'csv':
        return self._parse_csv(file, amount_of_transactions)
    else:
        raise Exception('Unknown file format')
```

```
def _parse_csv(self, file_str, amount_of_transactions):
    try:
        delimiter = self.get_csv_delimiter(file_str)
        rows = csv.reader(io.StringIO(file_str), delimiter=delimiter)
        header = next(rows)
    except Exception as ex:
        raise Exception("Invalid csv file")
```

```
    data = []
    for row in rows:
        row_dict = {}
        for j, value in enumerate(row):
            row_dict[header[j]] = value
        data.append(row_dict)
```

```

        if len(data) > amount_of_transactions:
            raise Exception(f"Number of transactions ({len(data)}) exceeds the limit of {amount_of_transactions}.")

```

```

    transactions = []

```

```

    for fc in data:

```

```

        try:

```

```

            temp = self.dict_to_transaction(fc)

```

```

            if temp is not None:

```

```

                transactions.append(temp)

```

```

            except Exception as ex:

```

```

                raise Exception(f"Invalid data in file: {ex}")

```

```

    return transactions

```

```

    def dict_to_transaction(self, row_csv):

```

```

        if row_csv['МФО'] == "":

```

```

            row_csv['МФО'] = row_csv['Рахунок'][4:10]

```

```

        elif row_csv['МФО'] != row_csv['Рахунок'][4:10]:

```

```

            return None

```

```

        if row_csv['МФО банка'] == "":

```

```

            row_csv['МФО банка'] = row_csv['Рахунок корреспондента'][4:10]

```

```

        if row_csv['Дебет'] == "":

```

```

            row_csv['Дебет'] = '0'

```

```

if row_csv['Кредит'] == "":
    row_csv['Кредит'] = '0'

transaction = Transaction(counterparty=row_csv['Кореспондент'],
    MFO_counterparty=row_csv['МФО банку'],
    ОКРО_counterparty=row_csv['ЄДРПОУ кореспондента'],
    account_counterparty=row_csv['Рахунок кореспондента'],
    doc_number=row_csv['Номер документа'],

    inflow_amount=float(row_csv['Дебет']),
    outflow_amount=float(row_csv['Кредит']),
    content=row_csv['Призначення платежу'],
    doc_date=row_csv['Дата документа'],

    organization="",
    MFO_organization=row_csv['МФО'],
    ОКРО_organization=row_csv['ЄДРПОУ'],
    account_organization=row_csv['Рахунок'],
    currency=row_csv['Валюта'],
    bank_name=row_csv['Назва банку'])

return transaction

```

#Файл data_structures.py містить в собі клас, який зберігає сконвертовані дані

ВИПИСОК

```

from dataclasses import dataclass

```

```
@dataclass
class Transaction:
    counterparty: str
    MFO_counterparty: str
    OKPO_counterparty: str
    account_counterparty: str

    doc_number: str

    inflow_amount: float
    outflow_amount: float
    content: str

    doc_date: str

    organization: str
    MFO_organization: str
    OKPO_organization: str
    account_organization: str

    currency: str
    bank_name: str
```

ДОДАТОК Б

Автоматизація конвертації банківських виписок для
інформаційних систем підприємств

АПРОБАЦІЯ

Міжнародна науково-практична конференція «Сучасні проблеми
наукового забезпечення енергетики»: Матеріали XXI Міжнародної
науково-практичної конференції молодих вчених і студентів
м. Київ, 23–26 квітня 2024 року

УКР.НТУУ''КПІ ім.. Ігоря Сікорського''_ІАТЕ_ЦТЕ_ТР-31мп

Аркушів 5

Київ – 2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

СУЧАСНІ ПРОБЛЕМИ НАУКОВОГО ЗАБЕЗПЕЧЕННЯ ЕНЕРГЕТИКИ

Матеріали XXI Міжнародної
науково-практичної конференції
молодих вчених і студентів
м. Київ, 23–26 квітня 2024 року

ТОМ 2



Київ- 2024

**СЕКЦІЯ №9 КОМП'ЮТЕРНИЙ ЕКОЛОГО-ЕКОНОМІЧНИЙ
МОНІТОРИНГ ТА ГЕОМЕТРИЧНЕ МОДЕЛЮВАННЯ ПРОЦЕСІВ І
СИСТЕМ**

Оптимізація кроссплатформних ігор на рушії Unity за допомогою Occlusion Culling.	210
<i>КОВАЛЕНКО І.В., магістрант</i>	211
<i>Керівник - проф., д.т.н. Аушева Н.М.</i>	
Оцінка залежності споживання температури та електроенергії у місті.	214
<i>КРІВДА Д.О., магістрант</i>	
<i>Керівник - доц., к.е.н. Сегеда І.В.</i>	
Використання інструментів ArcGIS в процесах верифікації та валідації геоданих та картографічних виробів..	216
<i>КРІВОВ'ЯЗЮК М.О., магістрант</i>	
<i>Керівник - доц., к.т.н. Сидоренко Ю.В.</i>	
Виявлення аномалій в даних на основі методів машинного навчання.	218
<i>МОРОЗ Є.О., магістрант</i>	
<i>Керівник - проф., д.т.н. Шушур О.М.</i>	
Персональний інформатор пацієнта про критичність стану забрудненості повітря.	221
<i>ЧУЙКО Д.С., бакалавр</i>	
<i>Керівник - доц., к.ф.-м.н. Тарнавський Ю.А.</i>	

**СЕКЦІЯ №10 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА КОМП'ЮТЕРНЕ
МОДЕЛЮВАННЯ**

System of experimental research of cryptographic protection methods.	224
<i>VARENKOV A.V., MS's programme</i>	
<i>Academic leader - assoc.prof., cand.eng.sc. Shapovalova S.I.</i>	
Удосконалення алгоритму вирівнювання стоматологічних моделей для сканування в 3D принтерах.	226
<i>ВЕРЕМІЙЧУК І.А., аспірант</i>	
<i>Керівник - проф., д.т.н. Аушева Н.М.</i>	
Лазерний доплерівський віброметр.	228
<i>ЖИЛА Д.Ю., магістрант</i>	
<i>Керівник - доц., к.т.н. Дивич М.П.</i>	
Автоматизація конвертації банківських виписок для інформаційних систем підприємств.	230
<i>ЗАСТУПАЙЛО М.І., магістрант</i>	
<i>Керівник - проф., д.т.н. Шушур О.М.</i>	
Використання графових баз даних та нечіткої логіки для аналізу та моделювання соціальних мереж.	233
<i>КОКІДЬКО Б.С., аспірант</i>	
<i>Керівник - проф., д.т.н. Шушур О.М.</i>	
Нейромережеві методи генерації текстури заданого матеріалу для тривимірних моделей.	236
<i>НЕРОСЛІК М.Є., магістрант</i>	
<i>Керівник - доц., к.т.н. Шаповалова С.І.</i>	
Проблеми управління процесами підтримки льотної придатності повітряних суден з урахуванням неповної інформації під час використання інтернету речей.	238
<i>ПЛУГОВОЙ А.О., аспірант; САЛІМОВ Р.М., мол. вчений</i>	

УДК 004.62

¹ Магістрант 1 курсу Заступайло М.І.¹ Проф., д.т.н. Шушура О.М.<https://scholar.google.com.ua/citations?user=beUGko0AAAAJ&hl=en>¹ КПП ім. Ігоря Сікорського

АВТОМАТИЗАЦІЯ КОНВЕРТАЦІЇ БАНКІВСЬКИХ ВИПИСОК ДЛЯ ІНФОРМАЦІЙНИХ СИСТЕМ ПІДПРИЄМСТВ

Постановка проблеми та її актуальність. У сучасному світі, де велика кількість підприємств залежить від швидкості та ефективності обробки фінансової інформації, проблема ручного аналізу та конвертації банківських виписок стає ключовим фактором, що обмежує розвиток та оптимізацію інформаційних систем підприємств. Необхідним є пошук нових підходів та рішень для автоматизації цього процесу та забезпечення його ефективності.

Вирішення цього питання дозволить автоматизувати процеси бухгалтерського та фінансового обліку, зменшити час, що витрачається на ручне введення даних, а також зменшити ризик помилок при обробці фінансових даних.

Незважаючи на переваги розробок, що існують на сучасному етапі, більшість з них мають і значну кількість недоліків, які варто врахувати. Серед потенційних та реальних мінусів багатьох існуючих систем варто зосередити увагу на складності їхньої інтеграції, високих витратах, різноманітні стандартів, які призводять до складнощів при спільній роботі з різними банками або системами, подекуди і ризики щодо безпеки та конфіденційності тощо [1].

Рішенням цих проблем, яке дозволить максимально оптимально автоматизувати конвертацію банківських виписок для інформаційних систем підприємств, є розробка модуля конвертації, який можна інтегрувати в різні системи підприємств, налаштовуючи лише клієнтську і адміністративну частину в самій системі окремо, у той час коли модуль конвертації залишається сталим. Це дозволить імпортувати платіжні виписки з різних банків та їх форматів у систему підприємства з легкістю. Така розробка дозволить завантаження банківських виписок з різних банків і в різних форматах та повертатиме конвертований файл в універсальному форматі для подальшого використання в системі.

Аналіз останніх досліджень. Варто зазначити, що розробки у галузі обміну даними з банками загалом, а зокрема саме конвертації платіжних банківських виписок швидко розвиваються, особливо в контексті впровадження відкритого банківського середовища та цифрової трансформації фінансового сектору.

Дослідження, які хоча б частково висвітлюють тему конвертації платіжних банківських виписок для інформаційних систем підприємств, представляють значний обсяг в літературі і наукових роботах. Проте конкретні наукові та практично орієнтовані праці, які б повноцінно висвітлювали тему на сучасному етапі та безпосередньо стосувалися б цього напрямку, дуже часто є обмеженими. Більшість досліджень переважно концентрують увагу на загальних принципах обміну даними між банками та корпоративними системами, при цьому не акцентуючи на пропозиціях та реалізаціях конкретного практичного інтегрування в роботу систем підприємств [2]. А ті дослідження, які мають хоча б якісь практичні рекомендації, зміщують увагу зазвичай на автоматизований аналіз виписок, ніж на демонстрацію конкретних ресурсів для конвертації та впровадження їх в інформаційні системи.

Формулювання мети. Метою цього дослідження є вдосконалення та впровадження процесу автоматизованої конвертації банківських виписок та забезпечення швидкого та точного обліку фінансів компанії. Основним завданням є підвищення ефективності обміну

даними з банками та зниження кількості помилок завдяки автоматизації конвертації виписок.

Відповідно, серед *основних задач* варто виокремити:

- розроблення модуля конвертації банківських виписок для інформаційних систем підприємств;
- аналіз та розробка механізму взаємодії з інформаційними системами компаній;
- забезпечення можливості аналізу використання модуля конвертації та розробка механізму обмеження доступу;
- розробка користувацького та адміністративного модулів для певної системи підприємства для аналізу інтеграції та роботи автоматичної конвертації.

Основна частина. На сучасному етапі інформаційні системи підприємств забезпечують ефективний інформаційно-технічний супровід всіх аспектів функціонування організації з метою досягнення стратегічних та оперативних цілей. Мета використання таких систем полягає в автоматизації та оптимізації бізнес-процесів, створенні конкурентної переваги, забезпеченню якості та інновацій, ефективної комунікації тощо.

Для аналізу поточного стану можливостей автоматизації конвертації банківських виписок для інформаційних систем підприємств було проведено дослідження різних джерел, серед яких документація окремих інформаційних систем, форуми спільнот, розділи документації, присвячені конвертації платіжної інформації тощо. Було виявлено, що в інформаційних систем є часткова можливість конвертації банківських виписок, проте вона зазвичай суттєво обмежена та не дозволяє автоматичного обміну з багатьма банками.

Для реалізації модуля конвертації банківських виписок для інформаційних систем підприємств найоптимальнішими стали такі засоби розробки, як мова програмування Python з бібліотеками [3], база даних PostgreSQL, фреймворк Flask, а також середовище розробки PyCharm. Розробка модуля була проведена з урахуванням потреб користувачів та з метою максимальної автоматизації процесу обміну даними з банками. Задля об'єктивного бачення можливостей модуля та оцінки його практичної доцільності було використано конкретну інформаційну систему підприємств – систему ODOO [4], у яку модуль і було інтегровано.

Першим кроком до реалізації автоматизованої конвертації банківських виписок стало проектування архітектури самого модуля конвертації. Далі на прикладі інформаційної системи ODOO було реалізовано користувацький та адміністративний службові модулі, які є необхідними для інтеграції модуля конвертації до інформаційної системи підприємства. Поеднання модуля конвертації платіжних банківських виписок та службової частини в інформаційній системі підприємства передбачає взаємодію модулів для того, щоб забезпечити автоматичний обмін платіжними виписками та спростити врахування змін у форматах виписок в системі компанії.

Користувацький службовий модуль забезпечує інтерфейс користувача, завдяки якому клієнти можуть завантажити файл платіжної банківської виписки, вказати параметри (банк, формат файлу тощо) та надіслати запит до модуля конвертації. Також він забезпечує відображення результатів конвертації та надає користувачеві конвертований файл в універсальному форматі JSON, який є зручним для використання в системах компаній. Адміністративний службовий модуль також надає інтерфейс, проте уже для адміністраторів, у якому можна відстежувати та аналізувати використання модуля конвертації клієнтами та керувати їх доступом до нього.

Модуль конвертації приймає запити від користувацької та адміністративної частини. У випадку отримання запиту від клієнтської частини модуль перевіряє наявність доступу користувача до нього, щоб далі приступити до аналізу та конвертації файлу з виписками, що прийшов за запитом: модуль перевіряє, чи є клієнт за вказаним токеном в базі даних і чи є у нього доступні конвертації. Після перевірки модуль обробляє файл виписки відповідно до вказаних параметрів, використовуючи правила обробки для відповідного банку та формату файлу. Отримані результати конвертації надсилаються назад до

клієнтської частини для відображення без збереження даних із виписок у модулі (Рис. 1).

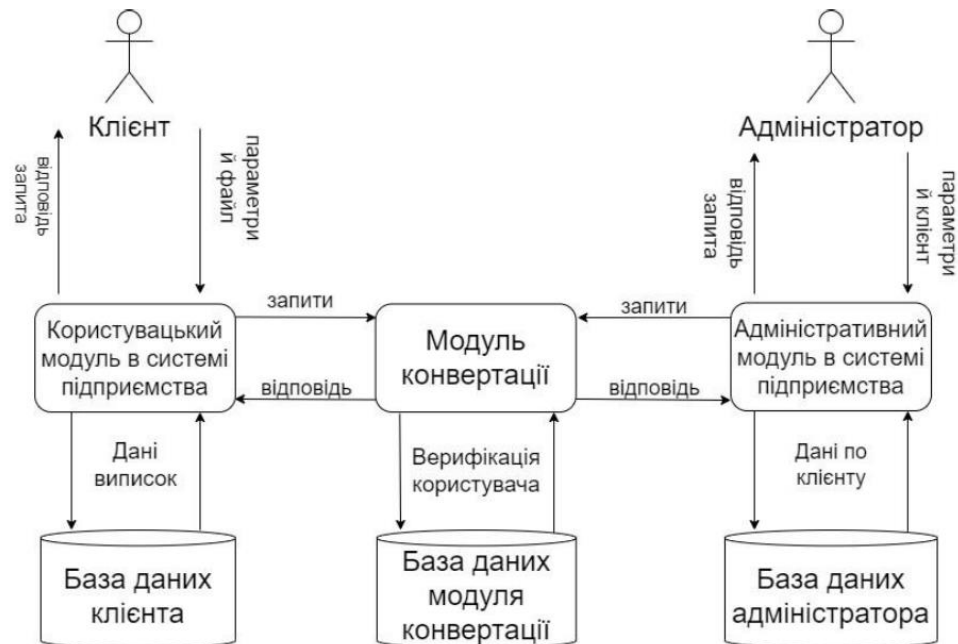


Рисунок 1 – Взаємодія складових

У випадку отримання запиту від адміністративної частини модуль валідує адміністратора та перевіряє токен клієнта, над яким адміністратор хоче виконати дію. Відповідно до змісту запиту модуль може надати дані по використанню свого функціоналу клієнтом, додати чи обмежити кількість доступних конвертацій або взагалі заблокувати доступ користувачу.

Висновки. Багато компаній використовують інформаційні системи для обліку та управління своєю діяльністю. Різні банки та платіжні системи можуть використовувати різні формати платіжних виписок. Розробка модуля конвертації дозволяє стандартизувати ці формати та забезпечити їх зрозумілість для інформаційних систем, що спростить обробку та аналіз фінансових даних. Інтеграція модуля конвертації платіжних виписок до інформаційної системи підприємства дозволить покращити обробку фінансових даних та забезпечить уніфікацію інформації в системі.

Отриманий продукт можна успішно використовувати в будь-якому бізнес-секторі за потреби взаємодії з банками та фінансовими установами. Модуль дозволить повноцінно автоматизувати конвертацію банківських виписок, що зробить фінансовий облік більш точним та прозорим.

Перелік посилань:

1. Jorge Martinez. 10 Best Bank Statement Converter Software in 2024. *DocuClipper*. URL: <https://www.docuclipper.com/blog/best-bank-statement-converter-software> (дата звернення: 02.03.2024).
2. Financial Reporting Software. *Capterra*. URL: <https://www.capterra.com/financial-reporting-software/#top-20> (дата звернення: 26.02.2024).
3. Sweigart A. Automate the Boring Stuff with Python. Practical Programming for Total Beginners : 2nd edition. NoStarchPress, 2019. 592 p.
 Odoo Documentation. User Docs, Install and Maintain, Developer, Contributing. URL: <https://www.odoo.com/documentation/16.0/> (дата звернення: 27.02.2024).