

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра обчислювальної техніки

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА
на тему:
**СИСТЕМА ЕЛЕКТРОННОГО ДОКУМЕНТООБІГУ З ІНТЕГРОВАНИМ
МОДУЛЕМ БЕЗПЕЧНОГО QR-ПІДПИСУ**

Виконав: студент 2 курсу, групи ІКІ-24м
спеціальності 123 — «Комп'ютерна інженерія»

ny Клейманов І. О.

Керівник: к.т.н., доц. каф. ОТ

[Signature] Тарновський М. Г.

« 12 » 12 2025 р.

Опонент: к.т.н., доц. каф. ПЗ

[Signature] Черноволик Г. О.

« 12 » 12 2025 р.

Допущено до захисту

Завідувач кафедри ОТ

д.т.н., проф. Азаров О. Д.

[Signature]
« 10 » 12 2025 р.

ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій та комп'ютерної інженерії

Кафедра обчислювальної техніки

Освітньо-кваліфікаційний рівень — магістр

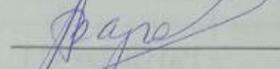
Галузь знань — 12 Інформаційні технології

Спеціальність — 123 Комп'ютерна інженерія

Освітньо-професійна програма — Комп'ютерна інженерія

ЗАТВЕРДЖУЮ

Завідувач кафедри ОТ, д.т.н., проф.



Азаров О. Д.

“ 25 ” вересня 2025 року

ЗАВДАННЯ

НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Клейманову Івану Олександровичу

1 Тема роботи: «Система електронного документообігу з інтегрованим модулем безпечного QR-підпису», керівник роботи к.т.н., доц. каф. ОТ Тарновський М.Г, затверджено наказом ВНТУ №313 від 24.09.2025 р.

2 Термін подання студентом роботи 04.12.2025 р

3 Вихідні дані до роботи: предметна область — електронний документообіг; вимоги до функціональності — взаємодія з базою даних документів, модулем авторизації користувачів, підсистемою генерації QR-кодів, підтримка REST API для обміну даними між компонентами; інструментальні засоби розробки — фреймворк Django (серверна частина), PostgreSQL (база даних), Docker (контейнеризація), мова програмування Python 3.10.

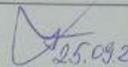
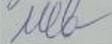
4 Зміст пояснювальної записки: вступ, аналіз предметної області, розробка обґрунтування методу побудови системи, програмна реалізація системи електронного документообігу та модуля підписання документів за допомогою QR-коду, тестування та перевірка функціональності системи, економічна частина.

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень): архітектурна схема системи електронного документообігу, блок-схема

процесу перевірки документа за допомогою модуля QR-підпису, UML-діаграм класів.

6 Консультанти розділів роботи представлені у таблиці 1

Таблиця 1 — Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-4	Тарновський М.Г., к. т. н., доц. каф. ОТ	 25.09.25	 12.12.25
5	Ратушняк О.Г., к.т.н., доц. каф. ЕПВМ		
Нормоконтроль	Швець С.І., асист. каф. ОТ		

7. Дата видачі завдання 25.09.2025 р

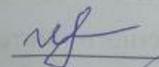
8. Календарний план виконання МКР наведений в таблиці 2

Таблиця 2 — Календарний план

№ з/п	Назва етапів МКР	Строк виконання	Примітка
1	Постановка задачі роботи	08.09.2025	виконано
2	Обґрунтування актуальності теми. Аналіз сучасних технологій електронного документообігу.	20.09.2025	виконано
3	Обґрунтування методу побудови та моделювання системи електронного документообігу	27.09.2025	виконано
4	Програмна реалізація системи електронного документообігу	30.09.2025	виконано
5	Тестування та перевірка функціональності системи	15.10.2025	виконано
6	Оцінка комерційного потенціалу розробки	17.10.2025	виконано
7	Оформлення пояснювальної записки та ілюстративного матеріалу	04.11.2025	виконано
8	Перевірка якості виконання магістерської кваліфікаційної роботи та усунення недоліків	10.11.2025	виконано

Студент

Керівник роботи




Клейманов І.О

Тарновський М.Г.

АНОТАЦІЯ

УДК 004.9

Клейманов І. О. Система електронного документообігу з інтегрованим модулем безпечного QR-підпису. Магістерська кваліфікаційна робота зі спеціальності 123 — Комп'ютерна інженерія, освітня програма — Комп'ютерна інженерія. Вінниця: ВНТУ, 2025 — 138 с. На укр. мові; Бібліогр.: 20 назв; рис.: 25; табл. 19.

У роботі проаналізовано сучасні підходи до автоматизації документообігу та методи захисту електронних даних. Розроблено архітектуру клієнт-серверної системи, яка забезпечує приймання, обробку та зберігання електронних документів. Реалізовано механізм QR-підпису, що поєднує візуальну зручність і криптографічний захист даних. Програмний засіб створено з використанням технологій Python (Django), React та Docker. Проведено тестування, яке підтвердило ефективність і надійність роботи системи. Розроблено систему, яка може бути використана підприємствами для підвищення безпеки та ефективності внутрішнього документообігу.

Ключові слова: електронний документообіг, програмний засіб, QR-підпис, безпека даних, обробка документів, веб-додаток, Docker.

ABSTRACT

UDC 004.9

Klieimanov I. O. Electronic document management system with an integrated secure QR signature module. Master's thesis in the field of 123 — Computer Engineering, educational programme — Computer Engineering. Vinnytsia: VNTU, 2025. 138 p. In Ukrainian language. Bibliographer: 20 titles; fig.: 25; tabl 19.

The work analyses modern approaches to document flow automation and methods of electronic data protection. The architecture of a client-server system has been developed to ensure the receipt, processing and storage of electronic documents. A QR signature mechanism has been implemented, combining visual convenience and cryptographic data protection. The software tool was created using Python (Django), React and Docker technologies. Testing was carried out, which confirmed the efficiency and reliability of the system. A system has been developed that can be used by enterprises to improve the security and efficiency of internal document flow.

Keywords: electronic document management, software tool, QR signature, data security, document processing, web application, Docker.

ЗМІСТ

ВСТУП	9
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	11
1.1 Аналіз сучасних тенденцій в області електронного документообігу.....	11
1.2 Огляд існуючих застосунків електронного документообігу.....	15
1.3 Постановка задачі	24
2 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ЕЛЕКТРОННОГО ДОКУМЕНТООБІГУ З МОДУЛЕМ QR-ПІДПISУ	27
2.1 Обґрунтування методу побудови системи електронного документообігу з безпечним модулем QR-підпису	27
2.1.1 Теоретичне обґрунтування методу хешування та перевірки	27
2.1.2 Математична модель системи журналу подій	28
2.1.3 Використання QR-підпису як методу подвійної автентифікації	28
2.2 Аналіз структури та компонентів системи електронного документообігу з безпечним модулем QR-підпису	29
2.3 Моделювання процесів функціонування системи електронного документообігу з модулем QR-підпису.....	30
2.3.1 Діаграма варіантів використання системи	31
2.3.2 Діаграма послідовності процесу підписання та перевірки документа....	32
2.4 Математична модель та алгоритм функціонування модуля QR-підпису ..	35
3 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ ЕЛЕКТРОННОГО ДОКУМЕНТООБІГУ ТА МОДУЛЯ ПІДПISАННЯ ДОКУМЕНТІВ ЗА ДОПОМОГОЮ QR-КОДУ	38
3.1 Архітектура системи електронного документообігу з модулем QR-підпису.	38
3.2 Опис основних класів та модулів програми.....	43
3.2.1 Клас DocumentUpload.....	43
3.2.2 Клас StorageManager.....	44
3.2.3 Клас UserAuth.....	45
3.2.4 Клас AdminSignService.....	45

3.2.5 Клас QRSignModule	46
3.3 Алгоритм роботи модуля QR-підпису	47
3.3.1 Реалізація перевірки справжності документа	49
3.3.2 Програмна реалізація модуля QR-підпису	50
3.4 Реалізація взаємодії між компонентами системи	50
3.4.1 Обмін даними через REST API.....	51
3.4.2 Взаємодія між Backend і базою даних	52
3.5 Розробка інтерфейсу користувача.....	53
4 ТЕСТУВАННЯ ТА ПЕРЕВІРКА ФУНКЦІОНАЛЬНОСТІ СИСТЕМИ ЕЛЕКТРОННОГО ДОКУМЕНТООБІГУ З МОДУЛЕМ БЕЗПЕЧНОГО QR-КОДУ	55
4.1 Методика проведення тестування.....	55
4.2 Функціональне тестування користувацьких операцій.....	56
4.3 Сценарій роботи програми зі сторони користувача.....	58
4.4 Сценарій роботи програми зі сторони адміністратора	59
4.5 Тестування взаємодії з сервером і базою даних	61
4.6 Аналіз результатів тестування.....	62
5 ЕКОНОМІЧНА ЧАСТИНА	63
5.1 Оцінювання комерційного потенціалу розробки	63
5.2 Прогнозування витрат на виконання науково-дослідної роботи.....	71
5.3 Розрахунок економічної ефективності науково-технічної розробки	79
5.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності..	80
ВИСНОВКИ	83
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	85
ДОДАТОК А Технічне завдання.....	87
ДОДАТОК Б ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ.....	92
ДОДАТОК В Архітектурна схема системи електронного документообігу з QR-підписом.....	93

ДОДАТОК Г Блок-схема процесу перевірки документа за допомогою модуля QR-підпису.....	94
ДОДАТОК Д UML-діаграма класів	95
ДОДАТОК Е Лістинг програми модуля безпечного QR-коду.....	96
ДОДАТОК Ж Лістинг програми модуля підписання документа	98
ДОДАТОК И Лістинг програми модуля журналу подій	99

ВСТУП

В умовах глобалізації та цифровізації економіки ефективне управління інформацією стає ключовим чинником успіху будь-якої організації. В Україні, де процеси цифровізації набирають обертів, особливо важливим є впровадження систем електронного документообігу (СЕД) для оптимізації управлінських процесів та забезпечення прозорості в роботі державних та приватних установ.

Актуальність теми дослідження полягає в тому, що системи електронного документообігу дозволяють автоматизувати процеси створення, зберігання, обробки та обміну документами в електронному форматі, що значно підвищує ефективність роботи організацій. Вони пропонують користувачам різноманітні функції, зокрема зберігання і пошук документів, підтримку функціонування відділу канцелярії, рух і контроль за виконанням документів, аналітичну звітність, інформаційну безпеку та ряд додаткових функцій [1]. Крім того, СЕД забезпечують користувачам різноранговий доступ до електронних документів відповідно до ролей і прав співробітників, при цьому контроль прав доступу може реалізовуватися на різних рівнях: на рівні сервера, на рівні папок як елементів платформи, а також на рівні конкретних виконуваних функцій. Нарешті, електронний документообіг полегшує архівування, збереження та відновлення документів за потреби [2].

Однак, незважаючи на значний прогрес, існують проблеми, які потребують вирішення. Серед них: недостатній рівень технічного забезпечення, відсутність єдиних стандартів обміну даними, а також необхідність забезпечення високого рівня безпеки та захисту персональних даних, зокрема достовірності, захищеності та автентичності документів, що циркулюють у мережевому просторі.

Метою роботи є вдосконалення програмного засобу для ефективного оброблення та зберігання даних електронних документів із застосуванням модуля безпечного QR-підпису, що підтверджує автентичність документів та захищає їх від підробки.

Для досягнення поставленої мети необхідно вирішити такі **задачі**:

- аналіз існуючих рішень у сфері електронного документообігу;
- проектування структури програмного засобу з інтегрованим модулем QR-підпису;
- розробка алгоритмів обробки, зберігання та перевірки автентичності документів;
- програмна реалізація системи електронного документообігу;
- тестування та перевірка функціональності розробленої системи;
- оцінювання економічної ефективності запропонованого рішення.

Методи дослідження — системного аналізу, об'єктно-орієнтованого проектування, тестування.

Об'єкт дослідження — процес обробки, зберігання та верифікації електронних документів у рамках систем електронного документообігу..

Предмет дослідження — методи та технічні засоби автоматизованої обробки, зберігання та верифікації електронних документів у системах електронного документообігу.

Новизна роботи полягає в удосконаленні підходу до реалізації електронного документообігу шляхом використання QR-коду для криптографічного захисту електронних документів, що забезпечує спрощення перевірки їх цілісності й автентичності.

Практичне значення роботи полягає в тому, що розроблений програмний засіб може бути впроваджений у державних та приватних організаціях для оптимізації процесів обробки, зберігання та перевірки автентичності електронних документів, що сприятиме підвищенню ефективності, прозорості та безпеки документообігу.

Апробація результатів магістерської кваліфікаційної роботи здійснена в доповіді на Міжнародній науково-практичній Інтернет-конференції студентів, аспірантів та молодих науковців «Молодь в науці: дослідження, проблеми, перспективи (МН-2026)».

Режим доступу: <https://conferences.vntu.edu.ua/index.php/mn/mn2026/paper/view/26642> [3].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз сучасних тенденцій в області електронного документообігу

Електронний документообіг (ЕД) є невід'ємною складовою сучасного управління інформацією в державних та приватних організаціях. Він включає процеси створення, обробки, зберігання та передачі документів у цифровому форматі. Основна мета ЕД — забезпечити швидкий, безпечний та контрольований рух інформації, що підвищує ефективність організаційних процесів та знижує ризики помилок, властивих паперовому документообігу [4].

Основними перевагами електронного документообігу в інформаційних відділах є ефективність обробки документів, зручний доступ до інформації, безпека та контроль доступу, а також архівування та збереження документів. Електронний документообіг дозволяє автоматизувати багато рутинних процесів, таких як реєстрація, маркування, підписання та розсилка документів, що значно зменшує час і витрати на ручну обробку.

Принципами системи електронного документообігу є одноразова реєстрація документа, яка дозволяє його однозначно ідентифікувати; паралельне виконання операцій, що скорочує переміщення документів і підвищує оперативність їх обробки; безперервний рух документа, що забезпечує постійний контроль за відповідальною особою протягом усього життєвого циклу документа; наявність єдиної бази документаційної інформації, що запобігає дублюванню документів; ефективна система пошуку, яка дозволяє швидко знаходити документи навіть за мінімальної інформації; а також розвинена система звітності за різними атрибутами документів, що сприяє контролю їх руху [5].

Системи управління документами (СУД) дозволяють структурувати, охороняти та впорядковувати документи, забезпечуючи зручний доступ до них для відповідних користувачів. Базы даних та індексація документів забезпечують швидкий пошук та автоматичну класифікацію документів. Такий підхід дозволяє користувачам оперативно знаходити потрібні файли, а системі — ефективно організувати структуру даних для подальшої обробки та аналізу [6].

Контроль за діями користувачів і адміністраторів у системі забезпечується механізмами аудиту та логування. Ведення детальних журналів дозволяє відслідковувати, хто і коли отримував доступ до документів, а також які операції були виконані, що підвищує прозорість та безпеку процесів електронного документообігу [7].

Останніми тенденціями ринку СЕД стали: мобільність, простота, зручність і раціональність систем, впровадження нових засобів обробки інформації, можливість працювати з неструктурованими даними, хмарні технології. Сучасними вимогами до функціоналу СЕД є:

- повноцінне управління документами з підтримкою всього життєвого циклу інформації (від створення документа до його надсилання на архівне зберігання);
- конфіденційність інформації з розмежуванням прав доступу до різних категорій користувачів СЕД;
- можливість застосування електронного підпису, створення юридично значущого документообігу;
- можливість кастомізації СЕД під потреби підприємства;
- здатність інтеграції з іншими інформаційними системами, що є в організації, наприклад, з ERP-системами (насамперед з модулями HR, фінансовим модулем), CRM-системами, банківськими та бухгалтерськими системами;
- забезпечення доступу до СЕД та роботи з інформацією та корпоративними програмами з різних мобільних пристроїв.

Інтеграція СЕД з корпоративними системами є однією із сучасних тенденцій в області електронного документообігу, що дозволяє автоматизувати обробку рахунків, договорів, звітів та інших типів документів. Завдяки такій інтеграції СЕД може взаємодіяти з обліковими, фінансовими та ERP-системами, що значно підвищує ефективність бізнес-процесів. Перевага корпоративної системи СЕД у порівнянні з системами діловодства і документообігу полягає в тому, що це рішення забезпечує універсальне, повсюдно доступне середовище для роботи та

зберігання усіх типів документів у масштабі всієї організації, дозволяє реалізовувати довільні схеми роботи з документами [8].

Для зберігання важливої інформації, зокрема договорів, довідкових матеріалів, статистичних даних тощо використовують різноманітні інформаційні бази та реєстри. Хмарні платформи дозволяють організаціям зберігати документи на віддалених серверах, забезпечуючи доступ із будь-якої точки світу. Це значно спрощує роботу з документами для віддалених працівників та дозволяє організаціям зменшити витрати на локальну інфраструктуру [9].

Тенденція використання хмарних сервісів спостерігається серед невеликих компаній. Хмари дозволяють малому бізнесу використовувати сервіси ЕДО без значних витрат на ІТ-інфраструктуру — закупівлю серверів або оренду обчислювальних потужностей. Хмарні сервіси ЕДО мають достатню для автоматизації малого та середнього бізнесу функціональність, але розміщені на віддалених серверах, які обслуговує провайдер.

Основним механізмом забезпечення автентичності електронних документів є цифровий підпис. Він дозволяє встановити, що документ був підписаний конкретним суб'єктом та не піддавався змінам після підписання. Модулі безпечного підпису реалізують застосування електронного цифрового підпису (ЕЦП) або криптографічно захищених QR-кодів для підтвердження автентичності документа. Це гарантує, що документи не були змінені після підписання та що підпис належить конкретному користувачу, що підвищує довіру до електронного документообігу [10].

Новим трендом є використання мобільних підпис — технології, яка дозволяє зберігати ключ підпису на пристрої користувача, а не на сервері або у хмарі. Вона доступна у вигляді програми, і її можна використовувати без доступу до Інтернету.

Ще однією із сучасних тенденцій є зростання попиту на інтелектуальні послуги в системах ЕДО. Якщо раніше системи документообігу лише зберігали і передавали файли, тепер вони вчаться передбачати дії користувачів, знаходити помилки і навіть давати рекомендації. Основними функціями, які забезпечують

технології штучного інтелекту, є автоматична класифікація, вилучення та аналіз даних, перевірка на помилки

Автоматична класифікація використовується для розпізнавання типів документів і їх розподіл за папками або маршрутами. Алгоритми машинного навчання навчаються на історичних даних, щоб прогнозувати сценарії. Наприклад, система розпізнає, що вхідний PDF — це рахунок-фактура, а не комерційна пропозиція, і відправляє його до потрібного відділу. Штучний інтелект може виявити аномалію: припустимо, виявити невідповідність у накладних.

Вилучення та аналіз даних базується засобах обробки природньої мови (Natural Language Processing, NLP) для виділення ключових реквізитів таких, як дати, суми, ІПН, що прискорює заповнення баз даних. У результаті реалізується не просто пошук ключових слів, а визначення сенсу під час аналізу контексту. Наприклад, аналіз договору оренди на предмет прихованих умов з подальшим виділенням їх для юриста.

Перевірка на помилки — штучний інтелект допомагає порівняти документи з типовими шаблонами та сигналізує про розбіжності, ризики, може прочитати скановані документи, рукописні нотатки та навіть інженерні креслення. Крім того, надається можливість розпізнавання печаток та рукописних підписів з метою перевірки їх справжності.

Впровадження штучного інтелекту в електронний документообіг дозволяє досягти, перш за все, скорочення помилок та економії часу. Однак необхідно враховувати особливості технології. Наприклад, якщо штучний інтелект навчають на неструктурованих архівах, він може перейняти помилки. Тому потрібно проводити попереднє очищення даних

Тенденції розвитку штучний інтелект у сфері документообігу дозволяють прогнозувати, що за рахунок технологій штучний інтелект будуть складатися проекти договорів, спираючись на прецеденти та побажання сторін. З'являться і цифрові паспорти товарів, які самі оновлюють дані про сертифікацію чи термін придатності через інтеграцію з IoT-датчиками. А технології блокчейну дозволять

зберігати документи децентралізовано, автоматично перевіряючи їхню справжність [11].

1.2 Огляд існуючих застосунків електронного документообігу

Сучасні організації активно впроваджують системи електронного документообігу (СЕД) для автоматизації обробки, зберігання та передачі документів у цифровому форматі. Такі програмні продукти дозволяють підвищити ефективність управлінських процесів, зменшити витрати часу на обробку документів та забезпечити контроль за їх рухом. Сучасні СЕД надають функціонал для створення, редагування, архівування документів, а також організації доступу користувачів.

Основні завдання таких систем включають централізоване зберігання документів, автоматизацію робочих процесів, забезпечення зручного пошуку та класифікації інформації. Важливим компонентом є модулі безпечного підпису, які дозволяють підтверджувати автентичність документа та забезпечують захист від підробки або несанкціонованого доступу. Інтеграція з корпоративними системами дозволяє автоматизувати обробку рахунків, договорів, звітів та інших типів документів, що значно підвищує продуктивність та прозорість внутрішніх процесів.

Сьогодні ринок пропонує широкий вибір систем електронного документообігу (СЕД). Розглянемо та порівняємо деякі з них, які активно використовуються у сучасних організаціях: "DocuWare" від компанії "DocuWare GmbH"[12], "М.Е.Дос" від компанії "SoftElegance"[13], "eDocs" від компанії "GlobalDocs Solutions"[14], "Vchasno" від компанії "Вчасно ЕДО"[15].

Система "DocuWare" є одним із найвідоміших міжнародних рішень в області управління документами, що поєднує централізоване зберігання, автоматизацію робочих процесів та забезпечення високого рівня безпеки.

Цей продукт розроблений німецькою компанією "DocuWare GmbH" і орієнтований на середні та великі підприємства. "DocuWare" дозволяє збирати

документи з різних джерел — сканерів, електронної пошти чи файлових систем — і автоматично їх індексувати, забезпечуючи швидкий пошук за метаданими або повним текстом. Система має розвинену підтримку OCR-розпізнавання, завдяки чому користувач може легко знаходити потрібні документи навіть серед великої кількості відсканованих копій.

Крім того, "DocuWare" надає гнучкі можливості керування доступом: можна призначати права окремим користувачам або групам, налаштовувати журнали дій, забезпечувати контроль версій документів. Програма підтримує розгортання як у хмарі, так і локально, а також інтегрується з іншими бізнес-системами.

Серед переваг "DocuWare" виділяють потужну автоматизацію workflow, високу швидкість пошуку та безпеку зберігання даних. Недоліками можна вважати високу ціну ліцензії, складність початкового налаштування й потребу у навчанні персоналу. Незважаючи на це, система користується великою популярністю в корпоративному секторі.

Однак варто враховувати, що висока технологічна складність та розширена функціональність системи "DocuWare" можуть зробити її досить вартісним рішенням, особливо для невеликих або малорозвинених компаній із обмеженим бюджетом. Система передбачає глибоку інтеграцію з корпоративними мережами, ERP- та CRM-рішеннями, підтримує електронний цифровий підпис, хмарне зберігання даних і високий рівень безпеки доступу до документів. Саме ці фактори забезпечують потужні можливості для автоматизації документообігу, але водночас підвищують витрати на впровадження, налаштування та щомісячне обслуговування ліцензій.

Також, "DocuWare" потребує наявності кваліфікованого персоналу для адміністрування системи або співпраці з офіційними партнерами компанії. Тому перед вибором цього програмного продукту варто ретельно оцінити не лише його функціональні переваги та рівень безпеки, а й відповідність фінансовим можливостям і реальним потребам організації. На рисунку 1.1 продемонстровано інтерфейс системи "DocuWare"

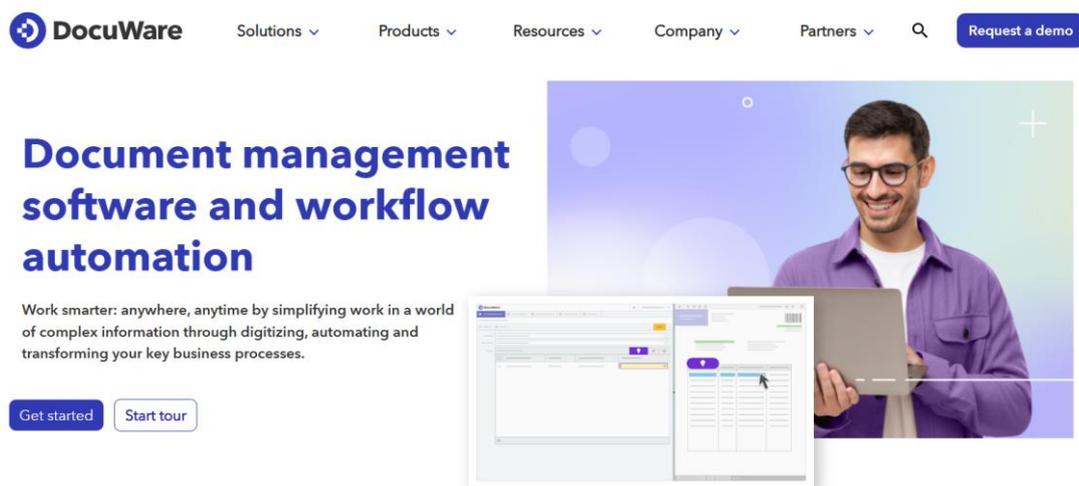


Рисунок 1.1 — Інтерфейс системи "DocuWare"

Ще однією популярною системою є система "М.Е.Дос" — програмне забезпечення, яке вже понад десятиліття використовується українськими підприємствами для обміну звітністю з державними органами та ведення бухгалтерського документообігу. М.Е.Дос дає змогу створювати, надсилати та зберігати податкову звітність, накладні, акти, рахунки та інші фінансові документи у форматі, що відповідає законодавчим вимогам. Система також підтримує обмін електронними документами між підприємствами, використовуючи сертифікати електронного підпису.

Програмне забезпечення має широкий набір шаблонів, регулярно оновлюється відповідно до змін у законодавстві, а також забезпечує інтеграцію з бухгалтерськими програмами, такими як 1С чи BAS.

Основною перевагою "М.Е.Дос" є глибока локалізація під українські стандарти, зручність у роботі для бухгалтерів і автоматичне формування звітів. Водночас варто зазначити, що система має обмежену універсальність — вона здебільшого орієнтована на бухгалтерський облік, а не на загальний документообіг.

Крім того, історично М.Е.Дос асоціюється з кібератакою 2017 року (вірус NotPetya), коли через оновлення програми поширювалося шкідливе програмне забезпечення. Попри це, система продовжує активно використовуватись,

розробники покращили безпеку та впровадили сучасні механізми захисту даних. Зараз "М.Е.Дос" є одним із наймасовіших продуктів для податкової звітності в Україні.

Однак варто враховувати, що система "М.Е.Дос", попри свою широке розповсюдження серед українських підприємств, має низку недоліків, які можуть ускладнювати її використання. По-перше, для повноцінної роботи користувач обов'язково повинен мати кваліфікований електронний підпис (КЕП), без якого неможливо виконувати обмін документами, подавати звітність чи підтверджувати юридичну силу електронних файлів. Це створює додаткові труднощі, адже отримання й оновлення підписів потребує часу, фінансових витрат і технічної підтримки. По-друге, система досить вимоглива до технічних ресурсів: потрібна ліцензійна версія програми, регулярне оновлення, стабільне інтернет-з'єднання та наявність серверного середовища або «хмарної» підтримки, що не завжди доступно для невеликих організацій.

Також варто звернути увагу на складність інтерфейсу, який потребує попереднього навчання або консультацій фахівців, оскільки нові користувачі часто стикаються з труднощами під час налаштування та початку роботи. Іншою проблемою є можливі збої під час оновлень програми або пікових навантажень на сервери, що може призводити до тимчасової недоступності сервісів. У минулому система також піддавалася критиці через ризики кібербезпеки, що ще раз підкреслює необхідність постійного моніторингу й технічного супроводу.

У сукупності ці чинники роблять "М.Е.Дос" менш гнучким рішенням, яке потребує додаткових ресурсів для впровадження та підтримки. Тому для невеликих підприємств або організацій із обмеженим бюджетом така система може виявитися занадто складною та затратною, порівняно з більш сучасними, простими у використанні платформами електронного документообігу. На рисунку 1.2 продемонстровано інтерфейс системи "М.Е.Дос".



Обирайте рішення для своєчасної звітності

Рисунок 1.2 — Інтерфейс системи "М.Е.Doc"

Ще однією сучасною системою електронного документообігу є "eDocs" — комплексне рішення, розроблене компанією "GlobalDocs Solutions", що орієнтоване на автоматизацію бізнес-процесів, пов'язаних зі створенням, обробкою, погодженням і зберіганням документів у цифровому форматі. Система активно використовується у корпоративному секторі, а також у державних і приватних установах, які прагнуть мінімізувати використання паперових документів і забезпечити прозорий документообіг. На рисунку 1.3 продемонстровано інтерфейс системи "eDocs".

Система "eDocs" забезпечує централізоване сховище документів із гнучкою системою контролю доступу. Користувачі можуть працювати з документами в реальному часі, відстежувати зміни, коментувати, затверджувати чи відхиляти файли. Програмне забезпечення інтегрується з популярними офісними пакетами — Microsoft 365, Google Workspace, а також із CRM- і ERP-системами, що дозволяє організаціям об'єднати всі процеси управління інформацією в єдиному середовищі.

Платформа трансформації процесів на основі AI

Індивідуальні рішення для управління закупівлями, договірною діяльністю, бухгалтерськими документами та HR-процесами.



Рисунок 1.3 — Інтерфейс системи "eDocs"

Серед основних переваг системи можна виділити гнучкість налаштування, високий рівень безпеки, а також зрозумілий інтерфейс, який не потребує спеціальних технічних знань. eDocs підтримує хмарну інфраструктуру, що дозволяє користувачам отримувати доступ до документів з будь-якого пристрою та місця, де є підключення до інтернету.

Разом із тим, попри широкий функціонал, "eDocs" має певні недоліки. Найсуттєвішим є відсутність підтримки криптографічно захищеного QR-підпису, який міг би забезпечити додатковий рівень автентифікації документів у процесі обміну між різними сторонами. Такий механізм дозволив би уникнути ризиків, пов'язаних із піддробкою або несанкціонованим доступом до файлів, а також зробити процес перевірки документів швидшим і зручнішим. Крім того, система потребує кваліфікованих IT-фахівців для налаштування інтеграцій, а повноцінна версія програмного забезпечення є комерційною, що може стати перешкодою для малих підприємств.

Загалом "eDocs" є ефективним інструментом для організацій, які прагнуть забезпечити прозорий та безпечний документообіг, але потребує подальшого вдосконалення у сфері захисту документів та впровадження інноваційних підходів, таких як QR-підпис, що реалізується у розроблюваній системі. Це цікава та досить нова пропозиція на ринку. Однак ця система є занадто вартісною для

використання як веб-додатку, особливо з урахуванням технічної підтримки. Вибір цієї системи можливий для простих сценаріїв, де кількість документів невелика, а процеси не дуже складні.

Іншим прикладом сучасного рішення є українська система "Vchasno", яка створена для спрощення електронного документообігу між підприємствами та державними установами. Це вебплатформа, що дозволяє створювати, погоджувати, підписувати та обмінюватися електронними документами онлайн без необхідності встановлення додаткового програмного забезпечення. Усі документи зберігаються у хмарному архіві, що гарантує швидкий доступ до них з будь-якого пристрою. Система адаптована до вимог українського законодавства, що робить її зручною для бізнесу, який веде діяльність в Україні. "Vchasno" підтримує кваліфікований електронний підпис (КЕП) і дає можливість підписувати документи навіть через мобільний додаток. Головними перевагами платформи є простота інтерфейсу, швидкість обміну документами, можливість колективної роботи та коментування файлів у режимі реального часу. Крім того, система має налаштовані рівні доступу, що дозволяє захистити інформацію від сторонніх осіб. На рисунку 1.4 продемонстровано інтерфейс системи "Vchasno ЕДО".

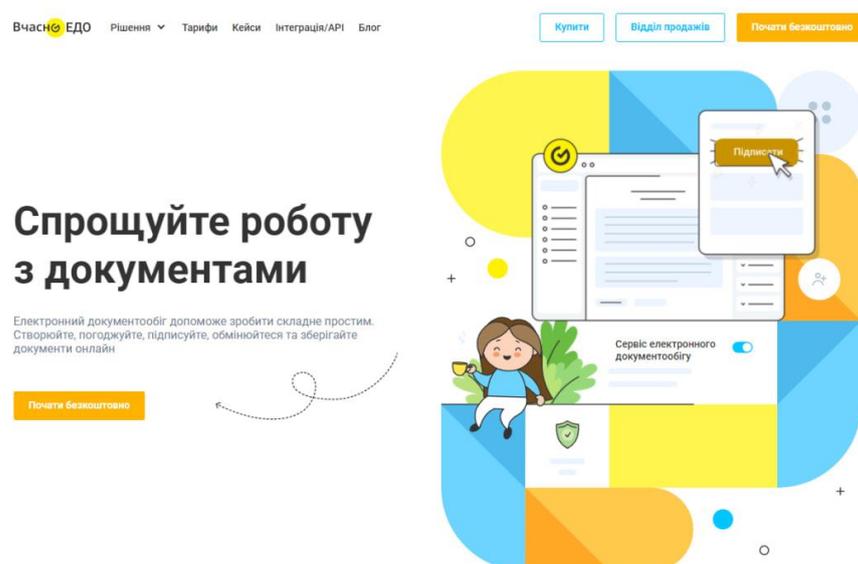


Рисунок 1.4 — Інтерфейс системи "Vchasno ЕДО"

Недоліком можна вважати те, що "Vchasno" поки що поступається міжнародним аналогам у складності автоматизації бізнес-процесів і гнучкості налаштувань, проте для більшості компаній в Україні цього функціоналу більш ніж достатньо. Система активно розвивається, з'являються нові модулі та можливості для інтеграції з ERP-рішеннями.

Аналіз сучасних систем електронного документообігу, таких як "DocuWare", "M.E.Doc.", "eDocs", "Vchasno", показує, що вони ефективно вирішують базові задачі зберігання та обміну документами, проте мають певні обмеження та недоліки, що знижують ефективність роботи організацій і ставлять під загрозу безпеку інформації. До основних недоліків існуючих рішень можна віднести обмежені можливості безпечної верифікації документів: більшість систем використовують традиційні електронні підписи, що не завжди зручні для масового користувача або інтеграції з мобільними та веб-додатками, а відсутність інтегрованого модуля безпечного QR-підпису ускладнює швидку перевірку автентичності документа.

На підставі проведеного аналізу встановлено, що існуючі системи електронного документообігу мають низку недоліків. Зокрема, це висока вартість ліцензій та послуг підтримки, потреба у кваліфікованих ІТ-фахівцях для налаштування та обслуговування системи, а також додаткові витрати на навчання персоналу та адаптацію програмного забезпечення під специфічні потреби організації.

Також, проблема полягає у відсутності універсального та безпечного інструменту електронного документообігу, який би забезпечував перевірку автентичності документів без необхідності використання складних криптографічних підписів, зберігаючи при цьому простоту інтеграції в робоче середовище підприємства.

Тому більшість існуючих рішень не забезпечують інтеграцію сучасних механізмів безпечного підпису, таких як QR-коди для верифікації документів, що обмежує швидкість перевірки автентичності та контроль за електронними документами. Ці недоліки підкреслюють необхідність розробки нової системи

електронного документообігу з інтегрованим модулем безпечного QR-підпису, що забезпечує підвищену безпеку та зручність використання.

З технічного погляду, предметна область, яку автоматизують інструменти документообігу, є досить простою. Однак будь-яка система документообігу фактично є лише сховищем неструктурованих документів. Тому усі розглянуті вище системи реалізують однакові функції і можуть бути описані за такими основними параметрами:

- можливість перегляду документа;
- контроль версій, історія змін;
- повнотекстовий пошук, фільтри, категорії;
- шифрування, контроль доступу, аудит дій користувачів;
- підтримка ЕЦП або QR-підпису;
- сповіщення про нові підпис документів або зміни;
- зручний інтерфейс;
- можливість перегляду дій;
- резервне копіювання;
- безкоштовне використання.

До вищезазначених критеріїв доцільно додатково включити кілька важливих аспектів неповної функціональності, які також впливають на вибір: підтримка електронного підпису (ЕЦП) або сучасних механізмів верифікації, зокрема QR-підпису; масштабованість та готовність до розгортання в хмарі; вартість впровадження і подальшої технічної підтримки; а також доступність документації та сервісної підтримки/спільноти. Для оцінки рекомендовано використовувати шкалу (наприклад, 1–5) з можливістю призначення ваг окремим критеріям відповідно до пріоритетів організації — це дозволить отримати зважений рейтинг і обґрунтувати вибір найкращого продукту для подальшого впровадження.

Для простоти порівняння кожної функції продуктів поставимо оцінки за п'ятибальною шкалою та проаналізуємо їх за вище перерахованими функціями в табл. 1.1.

Таблиця 1.1 — Порівняння функцій продуктів

№ критерію функціональних вимог	DocuWare	M.E.Doc.	eDocs	Vchasno
1.	5	4	4	4
2.	5	3	3	3
3.	5	3	3	4
4.	5	4	3	4
5.	5	5	3	4
6.	5	4	3	3
7.	5	3	4	4
8.	5	3	3	3
9.	5	3	3	3
10.	-	4	3	4
Підсумковий середній бал	4.5	3.5	3.2	3.6

Крім функціональних можливостей також розглянемо ціни розглянутих СЕД. Більшість з них занадто дорогі, а процес їх впровадження виявляється складним та тривалим, тому для реалізації цих продуктів потрібно багато часу та коштів, що не під силу малим компаніям. На сьогоднішній день на ринку існує велика кількість систем електронного документообігу, кожна з яких має свої особливості, сильні та слабкі сторони.

1.3 Постановка задачі

Передумовами для впровадження електронного документообігу в інформаційних відділах є наявність прагнення керівника установи до впровадження нової системи та здатності здійснювати ефективні управлінсько-організаційні заходи для її роботи. Важливим є аналіз існуючих процесів документообігу з метою їх перегляду, спрощення та оптимізації, оскільки

електронний документообіг суттєво відрізняється від паперового в частині доставки документів виконавцям, спільної обробки документів та їх проходження всередині організації.

Ефективне впровадження залежить не лише від волі керівника та готовності персоналу, але й від розвиненої телекомунікаційної інфраструктури, серверного обладнання або застосування «хмарних технологій», а також від дієвої ІТ-служби підтримки, яка забезпечує щоденну роботу системи, консультування користувачів та методичну допомогу. Необхідним є наявність програми впровадження з поетапними кроками, забезпечення всіх працівників комп'ютерною технікою та ліцензованим програмним забезпеченням, а також постійне навчання персоналу[16].

На основі аналізу виявлених недоліків визначено постановку задачі розробки: створити систему електронного документообігу з інтегрованим модулем безпечного QR-підпису, яка забезпечує ефективно зберігання та обробку документів у цифровому форматі, просту і зручну перевірку автентичності документів через QR-підпис, детальне логування всіх дій користувачів і адміністраторів, інтеграцію з корпоративними системами для автоматизації обліку та обробки рахунків, договорів та звітів, а також можливість масштабування та доступу до документів через хмарні сервіси. Розв'язання поставленої задачі дозволить підвищити безпеку, прозорість та ефективність електронного документообігу в організаціях, що особливо актуально для України в умовах розвитку електронного урядування та впровадження сучасних інформаційних технологій[17].

Розроблювана система є програмним компонентом корпоративної інформаційної системи підприємства. Вона взаємодіє з базою даних документів, модулем авторизації користувачів та підсистемою генерації QR-кодів. Вхідними даними є цифрові копії документів (формату .pdf, .jpg, .docx), а вихідними — перевірені документи з унікальним QR-підписом та журнал подій[18,19].

Тому для розробленого програмного засобу результати впровадження передбачають суттєве підвищення ефективності роботи з електронними

документами та оптимізацію внутрішнього документообігу. Продукт орієнтований на створення безпечного, інтуїтивно зрозумілого та функціонального середовища для зберігання, обробки й контролю документів.

Впровадження підсистеми обробки даних передбачає такі результати:

- завантаження та зберігання електронних документів у форматі .pdf;
- можливість сортування документів;
- створення персонального розділу документів користувача;
- журнал дій користувача (логування операцій);
- автоматичне генерування QR-коду для кожного документа.

2 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ЕЛЕКТРОННОГО ДОКУМЕНТООБИГУ З МОДУЛЕМ QR-ПІДПISУ

2.1 Обґрунтування методу побудови системи електронного документообігу з безпечним модулем QR-підпису

У сучасному цифровому середовищі, де обмін документами між організаціями та користувачами здійснюється в електронній формі, головними вимогами стають достовірність, захищеність та простежуваність інформації. Система електронного документообігу з безпечним модулем QR-підпису покликана вирішити ці завдання, забезпечуючи контроль автентичності, запобігання модифікації документів і фіксацію дій користувачів через журнал подій. На відміну від традиційних систем, у яких підпис базується лише на криптографічних сертифікатах, запропонована модель поєднує цифровий підпис із QR-кодом, що містить шифровані метадані документа. Це дозволяє виконати верифікацію документу навіть за допомогою зовнішнього пристрою (наприклад, смартфона), без прямого доступу до бази даних системи.

Основна мета розробки методу полягає у створенні архітектури, що поєднує зручність користування, гнучкість інтеграції та надійний рівень криптографічного захисту. Для цього застосовано принцип модульності, що дозволяє розділити систему на незалежні компоненти: серверну частину, клієнтський інтерфейс, модуль керування документами, модуль QR-підпису та журнал подій.

2.1.1 Теоретичне обґрунтування методу хешування та перевірки

Для забезпечення цілісності електронних документів використовується метод криптографічного хешування. Нехай документ описується множиною байтів:

$$D = \{d_1, d_2, \dots, d_n\}, \quad (2.1)$$

Для нього обчислюється хеш-значення:

$$H(D) = \text{SHA256}(D), \quad (2.2)$$

Це хеш-значення зберігається у базі даних разом із метаданими документа. При накладанні QR-підпису хеш шифрується і кодується у QR-код, який містить ідентифікатор адміністратора, дату підписання, контрольну суму документа.

Під час перевірки автентичності виконується повторне обчислення

$$H(D') = \text{SHA256}(D') \quad (2.3)$$

та перевіряється умова:

$$H(D') = H(D), \quad (2.4)$$

Якщо рівність зберігається — документ вважається справжнім.

2.1.2 Математична модель системи журналу подій

Для забезпечення контролю за всіма діями користувачів у системі використовується журнал подій, а також він доступний для користувачів, для персонального використання:

$$L = \{e_1, e_2, \dots, e_m\}, e_i = (t_i, u_i, a_i), \quad (2.5)$$

де t_i — час виконання дії,

u_i — користувач,

a_i — тип дії.

Математична умова повноти журналу визначається як:

$$\forall a_i \in A, \exists e_i \in L: a_i = e_i(a), \quad (2.6)$$

2.1.3 Використання QR-підпису як методу подвійної автентифікації

QR-підпис реалізує принцип подвійної автентифікації, поєднуючи візуальну перевірку з цифровим підписом. QR-код створюється за допомогою алгоритму Ріда-Соломона, що забезпечує корекцію помилок при пошкодженні зображення. Ступінь відновлення визначається за формулою:

$$R = \frac{Q}{255} \times 100\%, \quad (2.7)$$

де Q — рівень корекції (L, M, Q, H).

Для рівня Q коефіцієнт відновлення складає приблизно 25%, що дозволяє впевнено використовувати QR-підпис навіть на фізично пошкоджених або сканованих копіях документів.

Отже, метод побудови системи електронного документообігу з безпечним модулем QR-підпису базується на використанні криптографічного хешування, структурованого журналу подій, принципів подвійної автентифікації та об'єктно-орієнтованої архітектури. Його застосування дозволяє створити захищену, надійну та масштабовану систему, здатну ефективно функціонувати як у локальних, так і у хмарних середовищах.

2.2 Аналіз структури та компонентів системи електронного документообігу з безпечним модулем QR-підпису

Архітектура системи електронного документообігу з безпечним модулем QR-підпису побудована з урахуванням принципів модульності, масштабованості та безпеки даних. Основна мета архітектурного рішення полягає в забезпеченні ефективної обробки, зберігання та автентифікації документів у цифровому середовищі з можливістю перевірки достовірності підпису через QR-код. Відповідно до цього, архітектура системи складається з кількох взаємопов'язаних рівнів: користувацького інтерфейсу, прикладної логіки, рівня даних та модуля безпеки.

Процес взаємодії між користувачем і адміністратором побудований за принципом двоступеневої автентифікації: користувач завантажує документ, після чого адміністратор накладає цифровий QR-підпис, що засвідчує достовірність документа. Загальну структуру системи наведено в Лістингу В.

Користувацький інтерфейс реалізує доступ до функцій системи через веб-додаток. Він забезпечує можливість авторизації користувачів, перегляду документів, додавання нових записів, підписання файлів та відстеження історії

дій. Для зручності використання передбачено фільтрацію, пошук документів і перегляд станів обробки (чернетка, підписано, перевірено).

Прикладний рівень є центральною частиною системи, що реалізує логіку обробки запитів, перевірки доступу, формування QR-підписів і взаємодію з базою даних. У цьому шарі працюють сервіси, які керують документами (DocumentService), користувачами (UserService), генерацією QR-кодів (QRService) та журналом подій (EventLogService). Така структурна сегментація дозволяє розширювати систему без впливу на інші модулі, що відповідає принципам SOLID-архітектури та концепції розподілених об'єктних систем.

База даних використовується для централізованого зберігання документів, даних користувачів, QR-підписів та журналу подій. Вона реалізована на основі реляційної моделі, що забезпечує цілісність і узгодженість даних. Основні таблиці відповідають сутностям Document, User, QRSignature та EventLog. Кожна з них має унікальний ідентифікатор та набір атрибутів, які відображають їхню роль у системі.

Модуль безпеки відповідає за створення QR-підписів і їх подальшу верифікацію. У процесі генерації підпису система формує унікальний QR-код, у який вбудовується зашифрований хеш документа або URL-посилання для перевірки його автентичності. Таким чином, навіть без доступу до системи користувач може відсканувати QR-код і перейти на сторінку перевірки достовірності документа. Це забезпечує додатковий рівень надійності та зручності використання.

2.3 Моделювання процесів функціонування системи електронного документообігу з модулем QR-підпису

Моделювання процесів функціонування системи електронного документообігу з безпечним модулем QR-підпису є важливим етапом її проектування, оскільки дозволяє візуально представити логіку взаємодії між користувачами, компонентами та підсистемами. За допомогою моделей UML (Unified Modeling Language) можна наочно продемонструвати, як саме система

реагує на дії користувача, які процеси відбуваються на кожному етапі та які дані передаються між компонентами.

Основна мета цього етапу — описати функціональні сценарії роботи системи, від моменту авторизації користувача до перевірки достовірності документа. Для цього застосовуються такі діаграми:

- діаграма варіантів використання (Use Case Diagram) — описує дії користувачів і взаємодію з системою;
- діаграма послідовності (Sequence Diagram) — демонструє порядок виконання операцій у часі;
- додатково може бути застосована діаграма діяльності (Activity Diagram) для зображення потоків робіт усередині системи.

2.3.1 Огляд варіантів використання системи

Діаграма варіантів використання показує ролі користувачів (акторів) і набір дій, які вони можуть виконувати. У системі передбачено кілька типів користувачів:

- адміністратор;
- зареєстрований користувач;
- перевіряючий (зовнішній користувач).

Адміністратор керує користувачами, встановлює права доступу та контролює журнал подій. Зареєстрований користувач може створювати, переглядати та ініціювати перевірку підписів. Перевіряючий, як зовнішній користувач, може відсканувати QR-код і перевірити достовірність документа без входу до системи.

У процесі обробки адміністратор отримує повідомлення про нові документи користувачів, після чого накладає QR-підпис. Підписаний документ стає доступним для перегляду користувачем разом із QR-кодом, який веде на сторінку перевірки.

Основні сценарії взаємодії включають авторизацію користувача, коли користувач вводить облікові дані, система перевіряє їх через базу даних і видає

токен доступу. Після авторизації користувач може завантажити файл, вказати його тип, опис та теги. Підписання документа адміністратором відбувається через створення QR-підпису, який додається до документа у вигляді зображення коду.

Перевірка документа дозволяє будь-якій особі відсканувати QR-код і перейти на вебсторінку для перевірки автентичності. Керування користувачами включає можливість адміністратора додавати або блокувати користувачів та переглядати історію змін.

Ця структура допомагає визначити функції на кожному рівні доступу та слугує основою для подальшого створення діаграм класів, послідовностей і активностей.

2.3.2 Діаграма послідовності процесу підписання та перевірки документа

Діаграма послідовності демонструє часову послідовність взаємодії між основними об'єктами системи під час виконання типового сценарію. У нашій системі це процес створення, підписання та перевірки документа через QR-код.

Основні об'єкти системи:

- user;
- administrator;
- frontend;
- application server;
- database;
- QR verification page.

Основні об'єкти системи включають користувача (User), який ініціює дії та взаємодіє з системою через інтерфейс. Інтерфейс взаємодії користувача реалізований через Frontend, який забезпечує зручний доступ до функцій системи та відображення інформації.

Application Server відповідає за основну бізнес-логіку, обробляючи запити користувачів, керуючи підписанням документів, авторизацією та іншими процесами. Усі дані зберігаються в Database, яка слугує надійним сховищем інформації про користувачів, документи, підписи та історію дій.

Для перевірки достовірності підписів використовується QR Verification Page — сторінка, на яку можна перейти після сканування QR-коду, щоб перевірити автентичність документа без входу до системи.

Послідовність дій у системі починається з авторизації користувача через вебінтерфейс. Користувач вводить свої облікові дані, система перевіряє їх у базі даних і видає токен доступу, що дозволяє надалі виконувати дії у захищеному середовищі.

Після успішного входу користувач може завантажити документ через інтерфейс. Application Server приймає файл, перевіряє його на відповідність формату, зберігає метадані та додає сам документ до бази даних, забезпечуючи цілісність і доступність даних для подальшої обробки.

Коли адміністратор натискає кнопку "Підписати", система через QRService формує унікальний QR-підпис для документа. Згенерований QR-код зберігається в таблиці QRSignature у базі даних і відображається на сторінці користувача, що дозволяє одразу перевірити правильність підпису та завантажити документ із кодом.

Для перевірки документа інша особа може сканувати QR-код. Після сканування відбувається запит до сторінки перевірки QR-підпису (QR Verification Page), де система порівнює дані з базою і повертає статус документа — справжній або змінений. Цей механізм дозволяє швидко і безпечно визначати автентичність документів навіть сторонніми користувачами без входу в систему.

Усі дії системи фіксуються в журналі подій через EventLogService. Це забезпечує прозорість операцій, дозволяє відслідковувати історію змін, виявляти помилки або несанкціоновані дії, а також є важливим елементом для аудиту та безпеки.

Така послідовність дій дозволяє візуалізувати весь процес роботи системи, показуючи черговість повідомлень, запитів і відповідей між компонентами. Вона допомагає виявити потенційні точки відмови, перевірити логіку взаємодії та оптимізувати продуктивність, забезпечуючи надійну і зручну роботу для користувачів та адміністраторів.

Схема процесу створення документа з QR-підписом наведена на рисунку 2.1



Рисунок 2.1 — Схема процесу створення документа з QR-підписом

2.4 Математична модель та алгоритм функціонування модуля QR-підпису

Основним об'єктом моделювання є електронний документ, який користувач завантажує у систему для підписання. Документ розглядається як структурований набір цифрових даних, який можна подати у вигляді послідовності байтів або символів. Для забезпечення унікальності кожного документа обчислюється його

хеш-значення за допомогою криптографічної функції SHA-256. Це значення виступає цифровим відбитком, який змінюється при будь-якому редагуванні документа.

Таким чином, QR-підпис можна описати як функцію від двох аргументів — документа та ключа адміністратора:

$$Q = F(D, K), \quad (2.1)$$

де Q — результат генерації QR-підпису;

D — дані документа;

K — криптографічний ключ або токен адміністратора.

У реальній реалізації функція F включає обчислення хешу, шифрування та графічне кодування даних у двовимірний QR-код. Такий підхід забезпечує одночасно компактність і надійність збереження інформації, а також дає можливість швидкої верифікації через зовнішні пристрої.

Процес перевірки підпису реалізується у зворотному порядку. При скануванні QR-коду користувачем система отримує закодовану інформацію, розшифровує її за відповідним ключем і порівнює отримане хеш-значення з тим, що обчислюється безпосередньо з поточного документа. У випадку рівності цих значень документ визнається автентичним, і користувач отримує підтвердження про достовірність підпису. Якщо ж дані не збігаються, система реєструє подію у журналі аудиту та формує повідомлення про спробу підробки або змінення вмісту.

Математично перевірку автентичності можна подати як перевірку тотожності двох функцій:

$$V = \begin{cases} 1, & \text{якщо } H(D) = H(D'), \\ 0, & \text{інакше} \end{cases}, \quad (2.2)$$

де $H(D)$ — еталонний хеш, збережений при створенні підпису,

$H(D')$ — хеш перевірюваного документа.

Результат $V = 1$ означає автентичність документа, тоді як $V = 0$ вказує на розбіжність або порушення цілісності.

Алгоритмічно модуль QR-підпису складається з трьох основних етапів: ініціалізації, генерації підпису та перевірки. На етапі ініціалізації система встановлює ключі шифрування, створює сесію користувача та перевіряє його повноваження. Далі виконується обчислення хеш-значення документа, формування структури даних для QR-коду і безпосереднє створення графічного підпису. Під час перевірки, у свою чергу, здійснюється дешифрування інформації з коду, обчислення нового хешу та співставлення результатів. Така архітектура забезпечує цілісність процесу і дозволяє незалежно оцінювати достовірність кожного етапу.

З точки зору математичного моделювання, модуль QR-підпису можна представити як детерміновану систему переходів станів. Нехай множина можливих станів системи позначається як:

$$S = \{s_0, s_1, s_2, s_3\}, \quad (2.3)$$

де s_0 — стан підготовки документа;

s_1 — стан обчислення хешу;

s_2 — генерація підпису;

s_3 — перевірка достовірності.

Перехід між станами здійснюється за детермінованим правилом:

$$s_{i+1} = f(s_i, x_i), \quad (2.4)$$

де x_i — зовнішня подія або вхідні дані користувача.

У результаті кожен крок функціонування системи описується предикатом, що визначає стан готовності або завершеності етапу.

Для оцінки ефективності функціонування модуля вводиться критерій продуктивності, який враховує час генерації QR-підпису, час верифікації та рівень коректності перевірок. Нехай T_g — середній час генерації, T_v — час перевірки, а

P_c — відсоток успішно підтверджених документів. Тоді узагальнений коефіцієнт ефективності можна подати у вигляді:

$$E = \frac{P_c}{T_g + T_v}, \quad (2.5)$$

Цей показник дозволяє кількісно порівнювати різні варіанти реалізації алгоритму QR-підпису та визначати оптимальні параметри його функціонування у системі електронного документообігу.

Отже, розроблена математична модель модуля QR-підпису відображає логіку функціонування системи від моменту підготовки документа до підтвердження його достовірності. Побудована модель дає змогу формалізувати ключові етапи обробки, оцінити надійність криптографічного підпису та створити основу для подальшого вдосконалення алгоритмів перевірки, зокрема у напрямку підвищення швидкодії та адаптивного реагування на потенційні спроби підробки цифрових даних.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ ЕЛЕКТРОННОГО ДОКУМЕНТООБІГУ ТА МОДУЛЯ ПІДПИСАННЯ ДОКУМЕНТІВ ЗА ДОПОМОГОЮ QR-КОДУ

3.1 Архітектура системи електронного документообігу з модулем QR-підпису

Система електронного документообігу з модулем підписання документів за допомогою QR-коду створена з метою забезпечення безпечного, надійного та зручного процесу обробки електронних документів між користувачем та адміністратором. Архітектура системи базується на багаторівневому принципі, що включає клієнтський рівень (Frontend), серверну частину (Backend), модуль безпечного QR-підпису та рівень зберігання даних. Така структура забезпечує чітке розмежування функцій, масштабованість системи та можливість подальшого розвитку без порушення її основної логіки.

На верхньому рівні архітектури розташований інтерфейс користувача, реалізований у вигляді веб-додатку. Через нього здійснюється основна взаємодія користувача із системою: завантаження документів, перегляд статусу їхнього опрацювання, отримання доступу до підписаних файлів та перевірка їх автентичності шляхом сканування QR-коду. Веб-інтерфейс має адаптивний дизайн і забезпечує коректну роботу на будь-якому пристрої — від персонального комп'ютера до мобільного телефону. Усі дії користувача передаються до серверної частини через захищені API-запити, що гарантує безпечну комунікацію та цілісність даних.

Важливу роль у системі відіграє адміністратор, який виступає як спеціальний користувач із розширеними повноваженнями. Саме він здійснює перевірку достовірності документів, приймає рішення щодо їхнього підписання або відхилення та формує QR-підпис. Після завантаження документа користувачем адміністратор отримує повідомлення у власній панелі керування, переглядає поданий файл, перевіряє його зміст та, у разі позитивного рішення, ініціює процес підписання. Внаслідок цього у документ вбудовується QR-код,

який містить зашифровані дані про автора підпису, час створення та унікальне хеш-значення документа. Таким чином забезпечується надійна ідентифікація адміністратора та захист документів від підробки.

Користувацький рівень архітектури системи електронного документообігу зображено на рисунку 3.1



Рисунок 3.1 — Користувацький рівень архітектури

Центральним елементом архітектури є серверна частина системи, що реалізує основну логіку обробки даних. Сервер отримує всі запити від клієнтської частини, здійснює їхню валідацію, автентифікацію користувачів, керує процесом збереження документів, а також координує роботу модуля QR-підпису. Для реалізації серверної логіки використано фреймворк Django, який побудований на мові Python та характеризується високою стабільністю, зручністю в розробці та підтримкою REST API. Взаємодія між клієнтською та серверною частинами здійснюється у форматі JSON, що забезпечує простоту інтеграції та швидкість обміну даними.

Серверна частина архітектури системи електронного документообігу зображено на рисунку 3.2

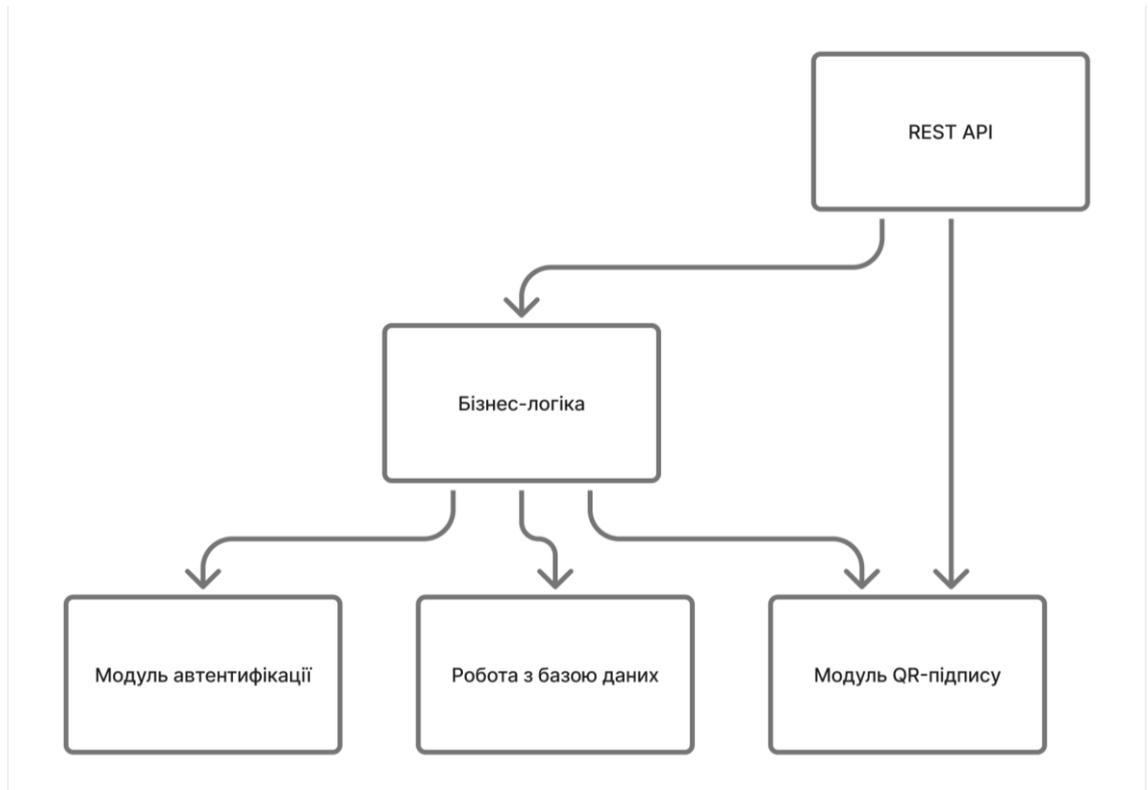


Рисунок 3.2 — Серверна частина архітектури

Користувацький інтерфейс реалізовано за допомогою фреймворку React, який дозволяє створювати динамічні веб-застосунки з високою швидкістю відгуку. Завдяки компонентній структурі React забезпечує гнучкість і розширюваність коду, що є важливим для системи, яка може розвиватися у майбутньому. У межах інтерфейсу реалізовано декілька основних функціональних сторінок: завантаження документів, перегляд їх статусів, панель адміністратора з інструментами підписання, а також сторінка перегляду готового підписаного документа з інтегрованим QR-кодом. Важливо, що всі дії користувача виконуються асинхронно, без потреби перезавантаження сторінки, що значно підвищує зручність користування системою.

Одним із ключових компонентів програмного комплексу є модуль безпечного QR-підпису, який відповідає за формування та перевірку цифрового підпису у вигляді QR-коду. Цей модуль функціонує на стороні сервера та

взаємодіє з іншими частинами системи через внутрішні API. Його робота ґрунтується на криптографічних методах хешування та шифрування. Для кожного документа обчислюється хеш-значення за алгоритмом SHA-256, після чого воно шифрується за допомогою приватного ключа адміністратора. Отримані дані кодуються у QR-зображення, яке додається до фінального документа. У базі даних зберігаються відомості про адміністратора, час створення підпису та оригінальний хеш, що дозволяє здійснити перевірку автентичності у будь-який момент. Таким чином, QR-підпис виконує роль цифрового сертифіката, який можна перевірити з будь-якого пристрою.

Архітектуру модуля безпечного QR-підпису зображено на рисунку 3.3



Рисунок 3.3 — Модуль безпечного QR-підпису

Уся логіка роботи системи базується на послідовній взаємодії між компонентами. Спочатку користувач завантажує документ через вебінтерфейс, після чого сервер зберігає його у базі даних та повідомляє адміністратора про новий запис. Адміністратор переглядає файл, перевіряє його зміст і у разі схвалення запускає процес QR-підпису. Сформований QR-код додається до документа, а статус оновлюється у системі. Після цього користувач отримує можливість завантажити підписаний документ і перевірити його справжність шляхом сканування QR-коду з будь-якого пристрою, що має доступ до Інтернету. Такий підхід забезпечує не лише зручність у роботі, а й високий рівень довіри до підписаних документів, оскільки їхня достовірність може бути підтверджена незалежно від системи, у якій вони були створені.

Для реалізації всіх компонентів системи використано сучасний набір технологій: на клієнтському рівні — React, на серверному — Django REST Framework, для збереження даних — PostgreSQL. Генерація QR-кодів і шифрування здійснюється за допомогою бібліотек `python-qrcode`, `hashlib` та `cryptography`. Така технологічна основа забезпечує надійність, масштабованість і відповідність сучасним вимогам до систем електронного документообігу.

Рівень даних архітектури системи електронного документообігу зображено на рисунку 3.4

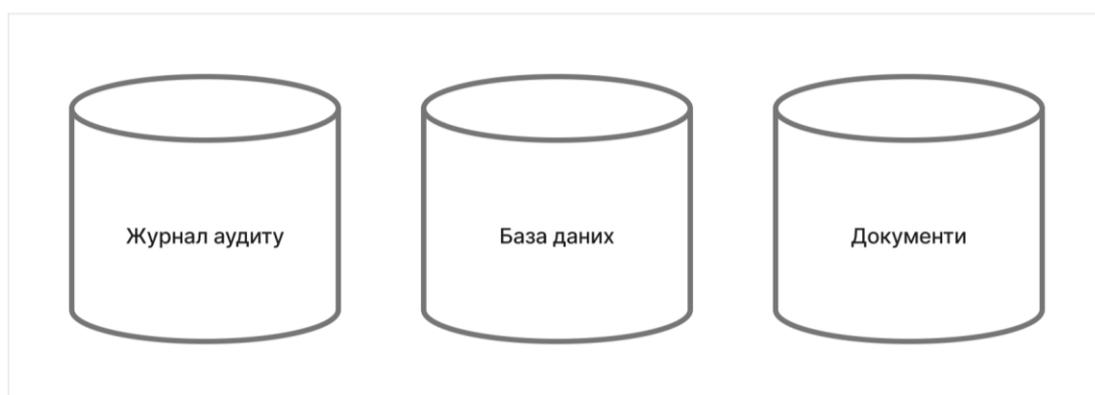


Рисунок 3.4 — Рівень даних

Отже, архітектура розробленої системи забезпечує чіткий розподіл ролей між користувачем та адміністратором, високий рівень захищеності даних, а також

гнучкість у подальшому розвитку проєкту. У наступному підрозділі буде наведено детальний опис основних класів та модулів програми, які реалізують зазначену архітектуру на практиці.

3.2 Опис основних класів та модулів програми

Програмна система електронного документообігу з модулем QR-підпису реалізована з урахуванням принципів модульності та розділення відповідальності. Кожен компонент має чітко визначену роль — від завантаження файлів користувачем до генерації QR-підпису адміністратором. Архітектурно система поділена на клієнтську частину (Frontend), серверну логіку (Backend) та модуль безпечного підписання.

Основні класи системи можна поділити на п'ять ключових груп: керування документами, зберігання файлів, автентифікація користувачів, підписання адміністратором, генерація QR-підпису.

3.2.1 Клас DocumentUpload

Клас DocumentUpload відповідає за логіку взаємодії користувача з системою під час завантаження документа.

Він реалізований на рівні клієнтського інтерфейсу (React-компонент), що викликає API-метод POST /api/upload

Програмна реалізація класу DocumentUpload наведено в Лістингу 3.1

Лістинг 3.1 — Клас DocumentUpload

```
const handleUpload = async (file) => {
  const formData = new FormData();
  formData.append('document', file);
  const response = await fetch('/api/upload', {
    method: 'POST',
    body: formData,
    headers: {
      Authorization: `Bearer ${userToken}`,
    },
  });
  if (response.ok) {
    alert('Документ успішно завантажено!');
  }
}
```

Основні методи класу `DocumentUpload`:

- `uploadFile(file)` — виконує завантаження файлу та валідацію;
- `createMetadata()` — формує описові дані для збереження в базі;
- `sendToServer()` — передає дані про документ до бекенду через API.

Після завантаження дані передаються на сервер, де викликається функція обробки збереження. Клас також відповідає за первинну валідацію (перевірку розміру, типу, імені файлу) та відображення статусу завантаження у веб-інтерфейсі.

3.2.2 Клас `StorageManager`

У серверній частині реалізовано клас `StorageManager`, який відповідає за фізичне збереження документів.

Він обробляє файли після завантаження користувачем, генерує унікальні шляхи, додає записи в базу даних і забезпечує контроль цілісності.

Програмна реалізація класу `StorageManager` наведено в Лістингу 3.2

Лістинг 3.2 — Клас `StorageManager`

```
from django.conf import settings
from hashlib import sha256
class StorageManager:
    def save_document(self, file_obj, user):
        file_hash = sha256(file_obj.read()).hexdigest()
        file_path = os.path.join(settings.MEDIA_ROOT, f"{file_hash}.pdf")
        with open(file_path, 'wb') as f:
            f.write(file_obj.read())
        Document.objects.create(
            owner=user,
            file_path=file_path,
            hash=file_hash,
            status="uploaded"
        )
        return file_hash
```

Основні методи класу `StorageManager`:

- `saveDocument(document)` — зберігає файл і створює запис у базі;
- `getDocumentById(id)` — надає доступ до збереженого документа;

— `deleteDocument(id)` — виконує безпечне видалення документа після перевірки прав доступу.

Така реалізація забезпечує надійне зберігання з можливістю перевірки цілісності документа за хеш-значенням.

3.2.3 Клас `UserAuth`

Клас `UserAuth` реалізує логіку входу користувача до системи. Він перевіряє облікові дані та формує JWT-токен для подальшої взаємодії із сервером. У моделі передбачено розмежування ролей: звичайний користувач може лише завантажувати та переглядати документи, тоді як адміністратор має право підписувати їх.

Програмна реалізація класу `StorageManager` наведено в Лістингу 3.3

Лістинг 3.3 — Клас `UserAuth`

```
from rest_framework_simplejwt.tokens import RefreshToken
class UserAuth:
    def login_user(self, username, password):
        user = authenticate(username=username, password=password)
        if not user:
            raise ValueError("Невірний логін або пароль")
        refresh = RefreshToken.for_user(user)
        return {
            "access": str(refresh.access_token),
            "role": user.role
        }
```

Основні методи класу `UserAuth`:

- `login(email, password)` — здійснює автентифікацію користувача;
- `logout()` — завершує сесію;
- `verifyRole()` — перевіряє рівень доступу користувача до певних дій.

3.2.4 Клас `AdminSignService`

Після завантаження документа адміністратор має можливість переглянути його та ініціювати процес підписання.

Цей функціонал реалізовано в класі `AdminSignService`, який взаємодіє з модулем QR-підпису. Він перевіряє, чи документ не змінювався після завантаження, створює підпис і додає QR-код.

Програмна реалізація класу `AdminSignService` наведено в Лістингу 3.4

Лістинг 3.4 — Клас `AdminSignService`

```
from .qr_sign import QRSignModule
from datetime import datetime
class AdminSignService:
    def sign_document(self, document_id, admin):
        document = Document.objects.get(id=document_id)
        qr = QRSignModule()
        signed_qr = qr.generate_qr_signature(document.hash, admin.private_key)
        document.qr_signature = signed_qr
        document.signed_by = admin.username
        document.signed_at = datetime.now()
        document.status = "signed"
        document.save()
        return document
```

Основні методи класу `AdminSignService`:

- `reviewDocument(document)` — відкриває документ для перевірки;
- `signDocument(document)` — створює запит на генерацію підпису;
- `rejectDocument(document)` — відхиляє документ у разі помилки чи невідповідності.

Таким чином, `AdminSignService` виконує роль центрального сервісу для верифікації та затвердження документів.

3.2.5 Клас `QRSignModule`

Клас `QRSignModule` реалізує криптографічну частину системи. Він відповідає за обчислення хешу документа, шифрування його приватним ключем адміністратора та створення QR-коду з підписом. При перевірці підпису модуль розшифровує дані та звіряє їх із базою даних.

Лістинг 3.5 — Клас `QRSignModule`

```
import qrcode
from cryptography.hazmat.primitives import hashes, serialization
```

```

from cryptography.hazmat.primitives.asymmetric import padding
class QRSignModule:
    def generate_qr_signature(self, document_hash, private_key):
        signature = private_key.sign(
            document_hash.encode(),
            padding.PKCS1v15(),
            hashes.SHA256()
        )
        qr = qrcode.make(signature.hex())
        qr_path = f"media/qr/{document_hash}.png"
        qr.save(qr_path)
        return qr_path
    def verify_qr_signature(self, document_hash, signature, public_key):
        try:
            public_key.verify(
                bytes.fromhex(signature),
                document_hash.encode(),
                padding.PKCS1v15(),
                hashes.SHA256()
            )
            return True
        except Exception:
            return False

```

Основні методи класу QRSignModule:

- generateHash(document) — обчислює хеш документа (SHA-256);
- encryptHash(hash, key) — шифрує хеш за допомогою приватного ключа адміністратора;

- generateQRCode(data) — створює зображення QR-підпису;

Модуль побудовано з використанням бібліотек cryptography та qrcode, що забезпечує високий рівень надійності та сумісність із сучасними алгоритмами безпеки. У лістингу В відображено UML-діаграму класів DocumentUpload, StorageManager, UserAuth, AdminSignService, QRSignModule та їхні зв'язки типу «використання» та «виклик методів»

3.3 Алгоритм роботи модуля QR-підпису

Модуль QR-підпису є центральною складовою системи електронного документообігу, оскільки саме він забезпечує створення, збереження та перевірку цифрового підпису у вигляді QR-коду. Його функціонування базується на

криптографічних принципах, зокрема на алгоритмах хешування та шифрування. Головна мета роботи цього модуля — забезпечити автентичність, цілісність і невідмовність документа, що підписується адміністратором.

Після того як звичайний користувач завантажує документ до системи, він зберігається у базі даних і отримує унікальний ідентифікатор. Далі документ передається на розгляд адміністратору, який має повноваження його перевірити та підписати. У момент, коли адміністратор ініціює підписання, активується модуль QR-підпису. На цьому етапі відбувається формування цифрового відбитку документа, шифрування даних та створення графічного QR-коду, який містить зашифрований підпис.

Процес створення QR-підпису можна поділити на кілька логічних етапів. Першим кроком є отримання документа, який надходить на сервер у вигляді бінарного файлу. Сервер формує унікальний ідентифікатор та здійснює базову перевірку структури документа. Після цього система переходить до обчислення хеш-значення.

На другому етапі відбувається генерація хешу документа за допомогою алгоритму SHA-256. Отримане хеш-значення є цифровим відбитком файлу — тобто коротким, але унікальним представленням його вмісту. Важливо, що навіть незначна зміна у тексті документа призводить до суттєвої зміни хешу, що забезпечує високий рівень контролю цілісності даних. Це гарантує, що документ, який підписується, ідентичний тому, який був завантажений користувачем, і не зазнав жодних змін.

Далі відбувається шифрування хешу за допомогою приватного ключа адміністратора. Цей етап реалізує криптографічний механізм підпису: шифрування гарантує, що лише власник відповідного приватного ключа міг створити підпис, а розшифрувати його можна буде тільки за допомогою публічного ключа, який є загальнодоступним у системі. Таким чином забезпечується невідмовність — адміністратор не може заперечити факт підписання конкретного документа.

На наступному етапі формується QR-зображення, у якому зберігаються зашифровані дані: хеш документа, ідентифікатор адміністратора, дата й час підписання, а також метадані, необхідні для подальшої перевірки. QR-код генерується у форматі PNG або SVG, після чого інтегрується безпосередньо у PDF-документ як візуальний елемент. Розміщення коду у фінальному файлі забезпечує наочність та зручність перевірки — будь-який користувач може легко відсканувати його зі сторінки документа.

Після завершення формування QR-коду система зберігає оновлений документ у базі даних. До бази також записуються службові відомості: ідентифікатор підпису, дата, час і ключ адміністратора. Таким чином, створюється повна історія дій, яка може бути використана для аудиту або перевірки безпеки.

3.3.1 Реалізація перевірки справжності документа

Особливістю запропонованої системи є те, що процес перевірки підпису може бути виконаний з будь-якого пристрою, що має камеру та доступ до мережі. Коли користувач або зовнішній перевіряючий відскановує QR-код, система переходить до режиму перевірки. У цей момент клієнтський застосунок або вебінтерфейс передає зашифровані дані з QR-коду на сервер, де здійснюється розшифрування за допомогою публічного ключа адміністратора. Результатом цієї операції є відновлене хеш-значення.

Після розшифрування сервер обчислює новий хеш з актуальної версії документа, що зберігається в базі даних. Якщо обидва хеші збігаються, система формує позитивний результат перевірки і повідомляє, що документ є автентичним та не зазнав змін після підписання. У результаті користувач бачить повідомлення «Підпис дійсний», а також дані про адміністратора, який здійснив підпис, дату та час операції. Якщо ж хеші не збігаються, система виводить попередження про те, що підпис є недійсним або документ було змінено.

Таким чином, перевірка здійснюється без потреби у додатковому програмному забезпеченні: достатньо смартфона або будь-якого пристрою зі

стандартним QR-сканером і доступом до мережі. Після сканування користувача, який відсканував документ перенаправить на сторінку підтвердження. Це робить систему максимально зручною, доступною та незалежною від операційного середовища.

3.3.2 Програмна реалізація модуля QR-підпису

Нижче наведено фрагмент коду, який відображає основну логіку роботи модуля QR-підпису.

Лістинг 3.6 — Реалізація QRSignModule

```
class QRSignModule:
    def sign_document(self, document, admin_private_key):
        hash_value = sha256(document.content)
        encrypted_hash = encrypt(hash_value, admin_private_key)
        qr_data = {
            "doc_id": document.id,
            "signature": encrypted_hash,
            "admin_id": document.admin_id,
            "timestamp": get_current_time()
        }
        qr_image = generate_qr(qr_data)
        document.attach_qr(qr_image)
        document.status = "Підписано"
        save(document)
    def verify_qr(self, qr_data, public_key):
        decrypted_hash = decrypt(qr_data["signature"], public_key)
        document = get_document_by_id(qr_data["doc_id"])
        original_hash = sha256(document.content)
        return decrypted_hash == original_hash
```

У представленому алгоритмі чітко простежується двоетапна логіка: створення підпису (*sign*) і перевірка підпису (*verify*). Під час створення підпису модуль генерує унікальний хеш, шифрує його та додає до документа у вигляді QR-коду. Цей підхід гарантує неможливість підробки підпису без володіння приватним ключем адміністратора.

3.4 Реалізація взаємодії між компонентами системи

Взаємодія між клієнтською та серверною частинами системи реалізована за архітектурою клієнт–сервер із використанням принципів REST API. Це забезпечує

чітке розмежування логіки: клієнтська частина (Frontend) відповідає за інтерфейс користувача та візуальне представлення даних, тоді як серверна частина (Backend) — за обробку запитів, бізнес-логіку та взаємодію з базою даних.

Основний формат передачі даних між компонентами — JSON (JavaScript Object Notation). Його використання дозволяє легко серіалізувати та десеріалізувати об'єкти між React-інтерфейсом і сервером Django, а також забезпечує сумісність із іншими потенційними клієнтами, наприклад мобільними застосунками.

3.4.1 Обмін даними через REST API

Після автентифікації користувач отримує токен доступу (наприклад, JWT), за допомогою якого надсилає подальші запити до серверу. Кожна дія в системі — завантаження документа, перегляд статусу, підписання або перевірка QR — реалізована як окремий REST-маршрут.

Найважливіші точки взаємодії включають:

— `POST /api/documents/upload` — завантаження нового документа на сервер;

— `GET /api/documents/:id/status` — отримання статусу документа (очікує перевірки, підписаний тощо);

— `POST /api/documents/:id/sign` — підписання адміністратором документа;

— `GET /api/documents/:id` — отримання підписаного документа користувачем;

— `POST /api/verify-qr` — перевірка дійсності QR-підпису.

Після завантаження сервер викликає функцію `save_document()`, яка зберігає метадані у базі даних через ORM Django (Object-Relational Mapping). Цей підхід дозволяє працювати з таблицями бази даних у вигляді Python-класів, забезпечуючи зручність, безпечність та відсутність необхідності писати сирі SQL-запити.

3.4.2 Взаємодія між Backend і базою даних

Серверна частина, реалізована на фреймворку Django, комунікує з базою даних PostgreSQL через ORM-рівень. Для кожного документа створюється запис у таблиці Document, який містить такі поля, як ідентифікатор користувача, шлях до збереженого файлу. При виконанні операції підписання або перевірки система оновлює відповідні поля, зокрема дату підпису та ідентифікатор адміністратора.

Коли адміністратор підтверджує документ, сервер викликає внутрішній модуль QRSignModule, який генерує унікальний QR-код із криптографічним підписом. Згенерований код зберігається у форматі PNG та вбудовується у фінальний PDF-документ. Після цього користувач може отримати підписаний файл через запит "GET /api/documents/105"

У відповідь сервер надсилає файл або посилання на нього як показано у Лістингу 3.7

Лістинг 3.7 — запит на завантаження документа:

```
{
  "document_url": "https://server.com/files/signed/contract_v2_qr.pdf",
  "signed_by": "admin_01",
  "signed_at": "2025-11-09T13:42:00Z"
}
```

3.4.3 Взаємодія між Frontend та користувачами

Frontend-застосунок, реалізований на React, періодично оновлює стан інтерфейсу за допомогою запитів до API. Завдяки асинхронній обробці запитів користувач одразу бачить статус операцій — від завантаження файлу до підтвердження підпису. Це створює зручну логіку користувацької взаємодії та робить систему динамічною і зрозумілою.

Ця основна взаємодія між компонентами системи побудована на принципах модульності, прозорості та безпечної передачі даних. Кожен запит і відповідь мають чітко визначену структуру, що дозволяє легко масштабувати систему або інтегрувати її з іншими сервісами в майбутньому.

3.5 Розробка інтерфейсу користувача

Інтерфейс користувача є одним із ключових елементів розробленої системи, оскільки саме через нього відбувається взаємодія між користувачем, адміністратором і серверною частиною. Веб-інтерфейс створено з використанням технологій React, HTML5, CSS3 та бібліотеки Bootstrap для забезпечення адаптивного і зручного дизайну. Основна мета інтерфейсу — зробити процес завантаження, підписання та перевірки документів максимально простим та інтуїтивно зрозумілим навіть для користувачів без технічної підготовки.

Початкова сторінка системи містить мінімалістичне меню для входу. Користувач повинен ввести поле «Username» та «Password» аби отримати доступ до користування системою. На початковій сторінці також присутнє додаткове поле «Remember Me», що відповідає за функцію запом'ятовування користувача. На рисунку 3.5 зображено початкову сторінку системи.

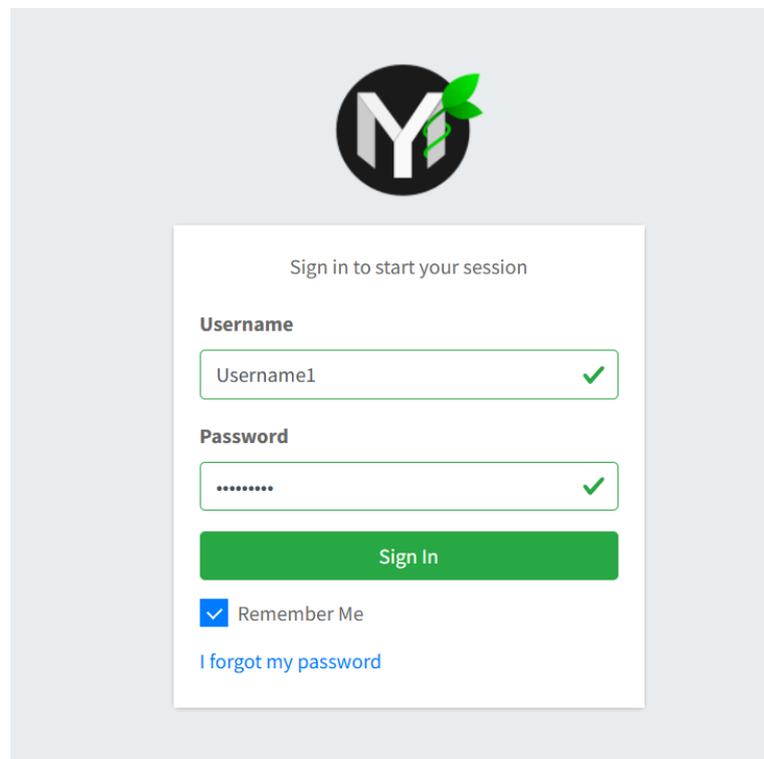


Рисунок 3.5 — Початкова сторінка системи

Головна сторінка системи містить мінімалістичне меню навігації з такими розділами: "Documents", "Inbox", "Tags" та "Logs". Наявні розділи

продемонстровані на рисунку 3.6

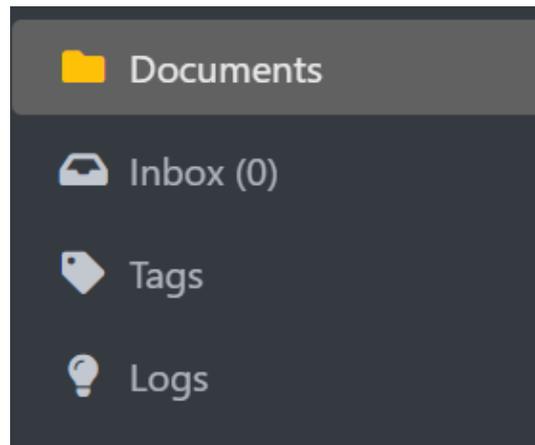


Рисунок 3.6 — Розділи головної сторінки системи

Головна сторінка сайту також містить кнопку «Upload» для завантаження нових документів та поле «OCR Language» щоб змінити мову для вдалого розпізнавання тексту.

Головна сторінка системи з головними функціями зображена на рисунку 3.7

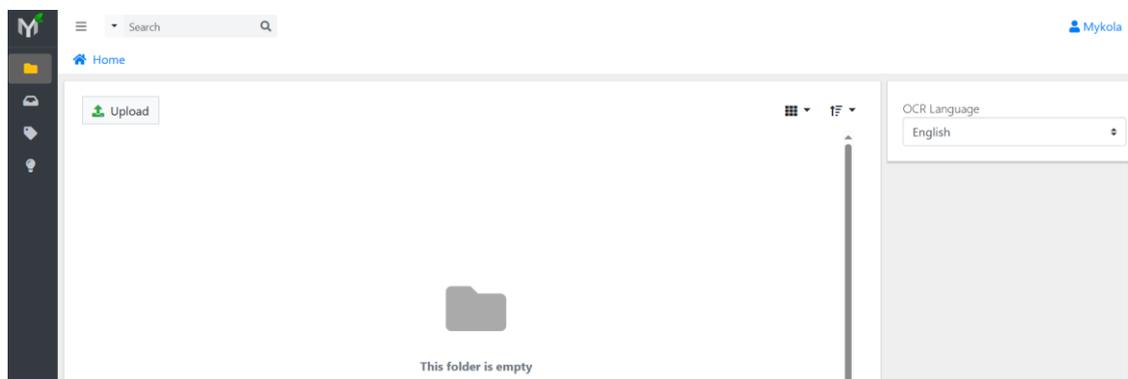


Рисунок 3.7 — Головна сторінка системи

Для кожного запису відображається назва файлу, дата додавання, статус підписання та кнопки для перегляду або завантаження документа. Завдяки простій структурі сторінки користувач може швидко орієнтуватися в наявних файлах і розуміти, на якому етапі обробки перебуває кожен документ. Ця сторінка виконує роль центральної панелі взаємодії користувача з системою.

4 ТЕСТУВАННЯ ТА ПЕРЕВІРКА ФУНКЦІОНАЛЬНОСТІ СИСТЕМИ ЕЛЕКТРОННОГО ДОКУМЕНТООБІГУ З МОДУЛЕМ БЕЗПЕЧНОГО QR-КОДУ

4.1 Методика проведення тестування

Метою тестування є перевірка працездатності та відповідності розробленої системи вимогам, визначеним на етапі проектування. Тестування дозволяє оцінити коректність виконання базових функцій, стабільність роботи системи при виконанні основних операцій користувачів, а також перевірити надійність обміну даними між клієнтською частиною, сервером та базою даних.

Тестування проводилося на розгорнутому локальному сервері з операційною системою Windows 10 Pro за допомогою Docker Compose. Клієнтська частина системи доступна через вебінтерфейс браузера Google Chrome. Для зберігання даних використовувалася база даних PostgreSQL, а серверна логіка реалізована на основі Django (Python).

Основні цілі тестування полягали у перевірці таких аспектів:

- автентифікації користувачів;
- завантаження, збереження та перегляду документів;
- підписання документів адміністратором;
- генерації та верифікації QR-підпису;
- оцінки зручності користувацького інтерфейсу.

На зображенні 4.1 показано інтерфейс користувача з панеллю навігації, полем пошуку та списком завантажених документів. Цей екран є початковою точкою роботи з документами у системі електронного документообігу. Також на ній розташовані дві кнопки для навігації по документах, перша кнопка відповідає за сортування документів, можна вибрати як список так і сіткове відображення файлів. В той час як інша кнопка відповідає за сортування за назвою, датою завантаження, та типом документа.



Рисунок 4.1 — Інтерфейс користувача з додатковими функціями

4.2 Функціональне тестування користувацьких операцій

Для перевірки основних можливостей системи було виконано функціональне тестування, спрямоване на оцінку коректності роботи користувацьких сценаріїв — від автентифікації до перевірки QR-підпису.

Тестування проводилося від двох ролей — користувача та адміністратора. Користувач мав можливість завантажити документ і переглянути його, тоді як адміністратор — перевірити, підписати та забезпечити можливість перевірки QR-коду.

Таблиця 4.1 — Результати функціонального тестування системи

№ критерію тестування	Сценарій тесту	Очікуваний результат	Результат тестування	Статус
1.	Авторизація користувача	Успішний вхід у систему	Авторизація виконана	Пройдено
2.	Завантаження PDF-документа	Документ відображається у списку	Завантажено успішно	Пройдено
3.	Перегляд документа	Додається QR-підпис	Візуалізація коректна	Пройдено
4.	Підписання адміністратором	Відображаються дані про документ та ким підписано	Підпис створено	Пройдено
5.	Перевірка QR-коду	Відображаються дані про документ та ким підписано	QR перевірено	Пройдено

Щоб завантажити файл, користувачу необхідно натиснути кнопку «Upload», після чого можна буде обрати кілька файлів на завантаження. На рисунку 4.2 зображено вікно завантаження документа користувачем.

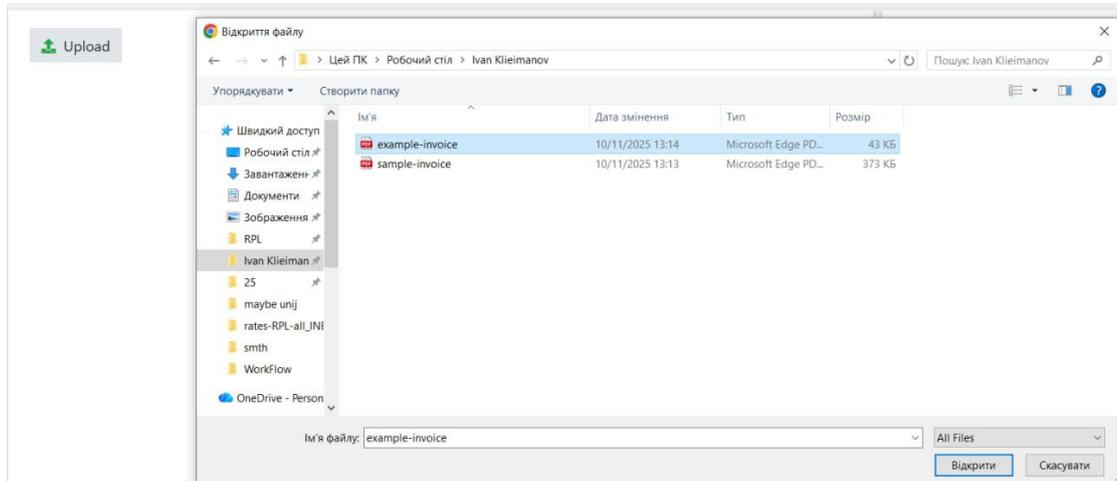


Рисунок 4.2 — Вікно завантаження документа

Це вікно дозволяє користувачу вибрати файл зі свого пристрою (наприклад, у форматі PDF або DOCX) і завантажити його до серверу.

У нижній частині форми присутня кнопка «Відкрити», після натискання якої документ передається на обробку через REST API-запит до серверної частини системи. Після успішного завантаження користувач бачить повідомлення про підтвердження вдалої операції завантаження.

Даний елемент інтерфейсу є ключовим для ініціації процесу електронного документообігу та демонструє інтеграцію клієнтської частини з API-сервісом.

На рисунку 4.3 зображено панель адміністратора.

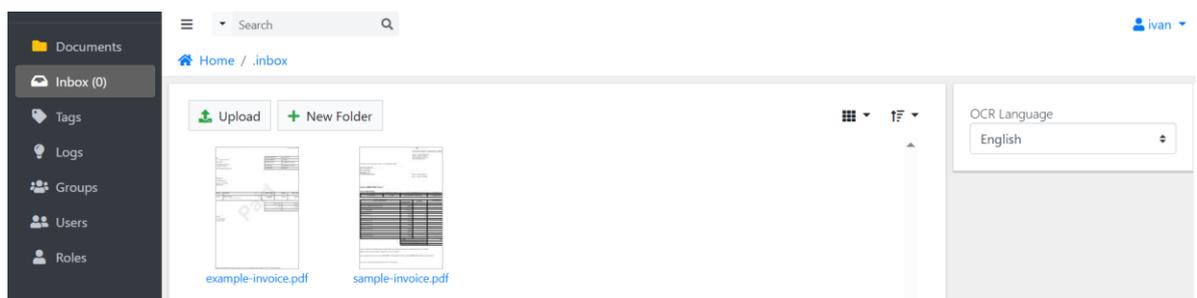


Рисунок 4.3 — Панель адміністратора

Адміністратор має розширені права доступу порівняно з користувачем. У його панелі відображаються всі документи, що очікують на перевірку або підписання. При натисканні на документ адміністратор може ознайомитися з його вмістом, перевірити достовірність інформації, після чого натиснути кнопку “Підписати документ”. Під час цієї дії система викликає модуль QRSignModule, який генерує криптографічний QR-підпис і додає його у фінальну версію документа.

Отже, панель адміністратора виконує функцію централізованого керування процесом підписання документів.

4.3 Сценарій роботи програми зі сторони користувача

Після авторизації користувач потрапляє на сторінку "Documents", де відображається список усіх завантажених файлів із зазначенням дати та доступними діями. Натиснувши кнопку «Upload», користувач обирає файл на своєму пристрої. Система відправляє його на сервер через REST API, а після успішного завантаження відображає повідомлення про успішне завантаження. Завантажений документ для підписання зображено на рисунку 4.4

Home / example-invoice.pdf

Fit Width ▾

Info

OCR Language: eng
 Owner: Username1
 Created: 2025-11-10 11:14:22
 Modified: 2025-11-10 11:14:25
 Version(s):
 Original v0 ▾

Download ▾

SlicedInvoices

Invoice

From:
 DEMO - Sliced Invoices
 Suite 5A-1204
 123 Somewhere Street
 Your City AZ 12345
 admin@slicedinvoices.com

To:
 Test Business
 123 Somewhere St
 Melbourne, VIC 3000
 test@test.com

Invoice Number	INV-3337
Order Number	12345
Invoice Date	January 25, 2016
Due Date	January 31, 2016
Total Due	\$93.50

Hrs/Qty	Service	Rate/Price	Adjust	Sub Total
1.00	Web Design This is a sample description...	\$85.00	0.00%	\$85.00
	Sub Total			\$85.00
	Tax			\$8.50

Рисунок 4.4 — Завантажений файл на підпис

Після того, як адміністратор підписує документ, користувач бачить оновлений статус і може натиснути на документ та переглянути його. У відкритому вікні з'являється PDF-файл, у який інтегровано QR-код із криптографічним підписом. У нижній частині сторінки також відображається коротка інформація: хто підписав документ, дата та час підпису.

Після сканування якщо документ справжній, користувач отримує повідомлення: "Документ підтверджено. Підписано адміністратором ivan 10.11.2025 о 14:16."

На рисунку 4.5 продемонстрована сторінка після успішного сканування QR-коду з підписаного документа.

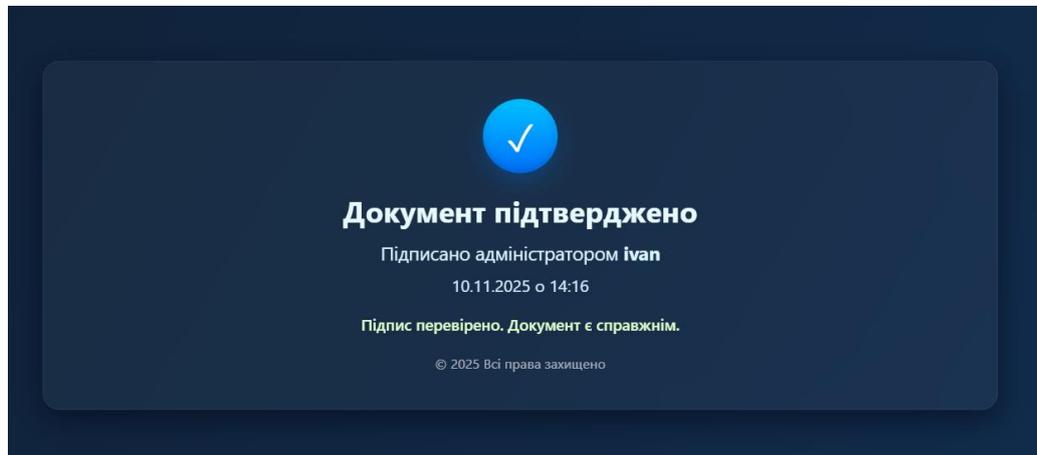


Рисунок 4.5 — Сторінка після сканування QR-коду

4.4 Сценарій роботи програми зі сторони адміністратора

Інтерфейс адміністратора містить розширену панель керування, яка дозволяє переглядати всі документи користувачів, здійснювати фільтрацію за статусами та датами, переглядати журнал подій, а також виконувати підписання через модуль QR.

У журналі подій, адміністратор може бачити які здійснювались дії користувачами в системі, хто та коли завантажував чи завантажував документ. Журнал подій відображено на рисунку 4.6

TIME	MESSAGE	USER	LEVEL
<input type="checkbox"/> 2025-11-10 11:56:55	COMPLETE OCR for document 1invoice.pdf, page=3, language=eng, d...	Username1	Info
<input type="checkbox"/> 2025-11-10 11:56:53	STARTED OCR for document 1invoice.pdf, page=3, language=eng, do...	Username1	Info
<input type="checkbox"/> 2025-11-10 11:56:53	COMPLETE OCR for document 1invoice.pdf, page=2, language=eng, d...	Username1	Info
<input type="checkbox"/> 2025-11-10 11:56:51	STARTED OCR for document 1invoice.pdf, page=2, language=eng, do...	Username1	Info
<input type="checkbox"/> 2025-11-10 11:56:51	COMPLETE OCR for document 1invoice.pdf, page=1, language=eng, d...	Username1	Info

Рисунок 4.6 – Журнал подій

На рисунку 4.7 показано як адміністратор може підписати документ, за допомогою функції QR-підпису.

The screenshot shows a document viewer interface. The main content area displays a table with invoice details:

Hrs/Qty	Service	Rate/Price	Adjust	Sub Total
1.00	Web Design This is a sample description...	\$85.00	0.00%	\$85.00
Sub Total				\$85.00
Tax				\$8.50
Total				\$93.50

Below the table, there is bank information: ANZ Bank, ACC # 1234 1234, BSB # 4321 432. A large 'Paid' watermark is overlaid on the document.

The right sidebar contains the following information:

- Info
- OCR Language: eng
- Owner: Username1
- Created: 2025-11-10 11:14:22
- Modified: 2025-11-10 11:14:25
- Version(s): Original v0
- Download button
- Add QR signature section with an Add button

Рисунок 4.7 — Вид документа від адміністратора перед підписом

Після вибору документа, адміністратор бачить, хто завантажив цей документ та коли, яка мова документа, а також завантажити початкову версію. Адміністратор також може натиснути кнопку "Add", після чого система автоматично генерує хеш документа, формує QR-підпис і додає його у фінальний PDF. На рисунку 4.8 продемонстрований підписаний адміністратором документ.

Hrs/Qty	Service	Rate/Price	Adjust	Sub Total
1.00	Web Design <small>This is a sample description...</small>	\$85.00	0.00%	\$85.00
Sub Total				\$85.00
Tax				\$8.50
Total				\$93.50

ANZ Bank
ACC # 1234 1234
BSB # 4321 432



Payment is due within 30 days from date of invoice. Late payment is subject to fees of 5% per month.
Thanks for choosing DEMO - Sliced Invoices | admin@slicedinvoices.com
Page 1/1

Рисунок 4.8 — Підписаний адміністратором документ

4.5 Тестування взаємодії з сервером і базою даних

Комунікація між клієнтською та серверною частинами здійснюється через REST API, що використовує формат обміну даними JSON. Для перевірки коректності роботи API було проведено серію запитів за допомогою Postman та Django REST Framework, які підтвердили стабільну роботу серверної частини та правильність обробки запитів. Приклад запиту для завантаження документа наведено нижче у Лістингу 4.1 та 4.2.

Лістинг 4.1 — Запит на завантаження документа

```
POST /api/upload_document/
Content-Type: multipart/form-data
{
  "file": "contract_2025.pdf",
  "user_id": 12
}
```

Лістинг 4.2 — Відповідь сервера на запит

```
{
  "status": "success",
  "message": "Document uploaded successfully",
  "document_id": 31
}
```

На рисунку 4.9 зображено вдалий запит завантаження документа на сервер.

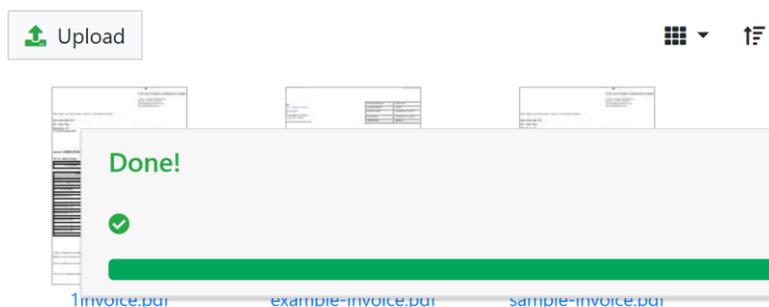


Рисунок 4.9 — Вдалий запит на завантаження документа

Якщо ж користувач вибере файл невідповідного формату, який не підтримує сервер, то цей файл не буде завантажено і запит буде відхилено. На рисунку 4.10 зображено невдалий запит завантаження документа на сервер.

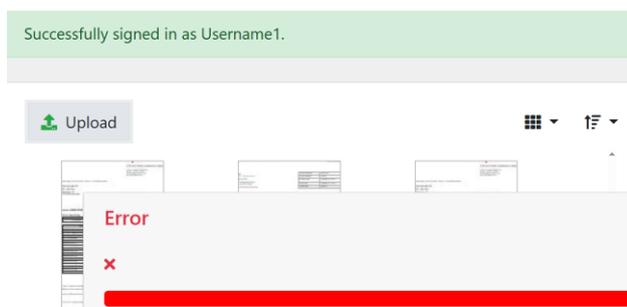


Рисунок 4.10 — Невдалий запит на завантаження документа

4.6 Аналіз результатів тестування

Результати тестування підтвердили, що розроблена система електронного документообігу з підтримкою QR-підпису повністю відповідає вимогам функціональності та стабільності. Усі базові сценарії роботи користувача та адміністратора виконані успішно. Особливу увагу під час перевірки було приділено коректності передачі даних між компонентами. Інтерфейс користувача відзначається інтуїтивністю та простотою використання. Тестування також підтвердило працездатність усіх основних модулів системи, включно з функціями авторизації, керування документами та підписання QR-кодом. Розроблена система демонструє стабільну роботу, відповідає вимогам надійності та забезпечує захищену взаємодію між усіма її компонентами.

5 ЕКОНОМІЧНА ЧАСТИНА

5.1 Оцінювання комерційного потенціалу розробки

Основна мета проведення комерційного та технологічного аудиту є вдосконалення програмного засобу для ефективного оброблення та зберігання даних електронних документів із застосуванням модуля безпечного QR-підпису, що підтверджує автентичність документів та захищає їх від підробки.

Для проведення технологічного аудиту було залучено 3-х незалежних експертів: Тарновський М. Г., к.т.н, доцент кафедри обчислювальної техніки Вінницького національного технічного університету, Кадук О. В., к.т.н, доцент кафедри обчислювальної техніки Вінницького національного технічного університету та Дудник О. В. к.т.н, доцент кафедри програмного забезпечення Вінницького національного технічного університету.

Для проведення технологічного аудиту було використано таблицю 5.1 в якій за п'ятибальною шкалою використовуючи 12 критеріїв здійснено оцінку комерційного потенціалу.

Таблиця 5.1 — Рекомендовані критерії оцінювання комерційного потенціалу розробки та їх можлива бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-терій	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів

Продовження табл. 5.1

5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкуренція немає
Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки пові-домлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Таблиця 5.2 — Рівні комерційного потенціалу розробки

Середньоарифметична сума балів СБ, розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0-10	Низький
11-20	Нижче середнього
21-30	Середній
31-40	Вище середнього
41-48	Високий

В таблиці 5.3 наведено результати оцінювання експертами комерційного потенціалу розробки.

Таблиця 5.3 — Результати оцінювання комерційного потенціалу розробки

Критерії	Прізвище, ініціали, посада експерта		
	к.т.н, доц Тарновський М.Г.	к.т.н, доц Кадук О.В	к.т.н, доц Дудник О.В
	Бали, виставлені експертами:		
1	4	4	3
2	3	3	3
3	4	4	4
4	3	3	3
5	4	4	3
6	3	3	4
7	3	3	2
8	4	4	4
9	4	3	4
10	5	4	4
11	4	3	2
12	4	4	4
Сума балів	СБ ₁ =49	СБ ₂ =42	СБ ₃ =41
Середньоарифметична сума балів $\overline{СБ}$	$\overline{СБ} = \frac{\sum_1^3 СБ_i}{3} = \frac{49 + 42 + 41}{3} = 44$		

Середньоарифметична оцінка, отримана на основі експертних висновків, становить 44 бали, і згідно з таблицею 5.2, це вказує на високий рівень комерційного потенціалу результатів проведених досліджень.

Розроблена система електронного документообігу з модулем QR-підпису може бути реалізована у вигляді веб-додатку, розгорнутого на корпоративному сервері або у хмарному середовищі. Такий підхід забезпечує гнучкість у розгортанні, простоту адміністрування та можливість доступу користувачів з будь-якого пристрою, що має підключення до мережі Інтернет.

Практична реалізація системи може бути здійснена у державних установах, навчальних закладах, приватних компаніях та підприємствах, де щоденно відбувається обмін великою кількістю внутрішніх документів. Наприклад, система може бути інтегрована у структуру внутрішнього документообігу бухгалтерії, відділу кадрів, юридичного відділу або в системи погодження договорів.

У перспективі система може бути масштабована для використання на рівні міжвідомчого документообігу, що дозволить органам державної влади або великим підприємствам здійснювати електронне підписання документів із збереженням повної юридичної сили та можливістю миттєвої перевірки автентичності будь-яким користувачем.

Проведемо оцінку якості і конкурентоспроможності нової розробки порівняно з аналогом.

У якості аналога для розробки було обрано систему електронного документообігу “Vshasno”, яка є одним із найпоширеніших рішень на українському ринку для обміну та підписання документів між підприємствами.

Основними недоліками аналога є залежність від зовнішніх сертифікаційних центрів та необхідність використання КЕП (кваліфікованого електронного підпису), що створює певні обмеження для невеликих компаній і приватних користувачів.

Крім того, процес перевірки підписаних документів у “Вчасно” можливий лише в межах самої платформи, що ускладнює перевірку достовірності сторонніми користувачами без реєстрації у системі.

Також до недоліків можна віднести закритість архітектури, через що інтеграція з іншими внутрішніми або галузевими системами підприємства є складною або потребує додаткових фінансових витрат. Крім того, система не

дозволяє зберігати локальні копії документів із можливістю незалежної перевірки підпису поза межами платформи, що знижує її гнучкість у використанні.

У розробці дана проблема вирішується шляхом інтеграції механізму QR-підпису, який не потребує зовнішніх криптографічних токенів чи сертифікаційних серверів. QR-код містить зашифровану інформацію про підписанта, дату і час підписання, а також хеш документа, що дозволяє перевірити автентичність будь-якому користувачеві, просто відсканувавши код смартфоном. Таким чином, процес перевірки стає повністю незалежним від самої системи документообігу.

Також система випереджає аналог за такими параметрами, як:

— доступність перевірки — автентичність документа можна підтвердити через будь-який пристрій без авторизації;

— швидкість підписання — QR-підпис генерується миттєво без участі сторонніх криптосерверів;

— гнучкість розгортання — система може працювати як локально на сервері підприємства, так і у хмарі;

— прозорість аудиту — усі операції підписання фіксуються у журналі подій із можливістю простеження дій користувачів;

— відкритість архітектури — модуль QR-підпису легко інтегрується у вже існуючі платформи документообігу.

Таким чином, запропонована система забезпечує спрощений, незалежний та безпечний процес підписання електронних документів, усуваючи обмеження, характерні для “Vchasno”, та роблячи перевірку документів максимально зручною для кінцевого користувача.

В таблиці 5.4 наведені основні техніко-економічні показники аналога і нової розробки.

Проведемо оцінку якості продукції, яка є найефективнішим засобом забезпечення вимог споживачів та порівняємо її з аналогом.

Таблиця 5.4 — Основні параметри нової розробки та товару-конкурента

Показник	Варіанти		Відносний показник якості	Коефіцієнт вагомості параметра
	Базовий (товар-конкурент)	Новий (інноваційне рішення)		
1	2	3	4	5
Зручність перевірки автентичності	30	100	3,33	30%
Незалежність від зовнішніх сервісів	50	100	2	25%
Гнучкість розгортання	50	100	2	20%
Продуктивність системи	70	90	1,3	15%
Вимоги до ресурсів (RAM), МБ	100	50	3	10%

Визначимо відносні одиничні показники якості по кожному параметру за формулами (5.1) та (5.2) і занесемо їх у відповідну колонку табл. 5.5.

$$q_i = \frac{P_{Hi}}{P_{Bi}} \quad (5.1)$$

або

$$q_i = \frac{P_{Bi}}{P_{Hi}} \quad (5.2)$$

де P_{Hi} , P_{Bi} — числові значення i -го параметру відповідно нового і базового виробів.

$$q_1 = \frac{100}{30} = 3,33;$$

$$q_2 = \frac{100}{50} = 2;$$

$$q_3 = \frac{100}{50} = 2;$$

$$q_4 = \frac{90}{70} = 1,3;$$

$$q_5 = \frac{100}{50} = 2.$$

Відносний рівень якості нової розробки визначаємо за формулою:

$$K_{\text{я.в.}} = \sum_{i=1}^n q_i \cdot \alpha_i, \quad (5.3)$$

$$K_{\text{я.в.}} = 3,33 \cdot 0,3 + 2 \cdot 0,25 + 2 \cdot 0,2 + 1,3 \cdot 0,15 + 2 \cdot 0,1 = 2,3$$

Відносний коефіцієнт показника якості нової розробки більший одиниці, отже нова розробка якісніший базового товару-конкурента.

Наступним кроком є визначення конкурентоспроможності товару. Конкурентоспроможність товару є головною умовою конкурентоспроможності підприємства на ринку і важливою основою прибутковості його діяльності.

Однією із умов вибору товару споживачем є збіг основних ринкових характеристик виробу з умовними характеристиками конкретної потреби покупця. Такими характеристиками найчастіше вважають нормативні та технічні параметри, а також ціну придбання та вартість споживання товару.

В табл. 5.5 наведено технічні та економічні показники для розрахунку конкурентоспроможності нової розробки відносно товару-аналога, технічні дані взяті з попередніх розрахунків.

Таблиця 5.5 — Нормативні, технічні та економічні параметри нової розробки і товару-виробника

Показники	Варіанти	
	Базовий (товар- конкурент)	Новий (інноваційне рішення)
1	2	3
<i>1. Нормативно-технічні показники</i>		
Зручність перевірки автентичності	30	100
Незалежність від зовнішніх сервісів	50	100
Гнучкість розгортання	50	100
Продуктивність системи	70	90
Вимоги до ресурсів (RAM), МБ	100	50
<i>2. Економічні показники</i>		
Ціна придбання, грн	4000	2000

Загальний показник конкурентоспроможності інноваційного рішення (К) з урахуванням вищезазначених груп показників можна визначити за формулою:

$$K = \frac{I_{m.n.}}{I_{e.n.}}, \quad (5.4)$$

де $I_{m.n.}$ — індекс технічних параметрів;

$I_{e.n.}$ — індекс економічних параметрів.

Індекс технічних параметрів є відносним рівнем якості інноваційного рішення. Індекс економічних параметрів визначається за формулою (5.5)

$$I_{e.n.} = \frac{\sum_{i=1}^n P_{Hei}}{\sum_{i=1}^n P_{Bei}}, \quad (5.5)$$

де P_{Hei} , P_{Bei} — економічні параметри відповідно нового та базового товарів.

$$I_{e.n.} = \frac{4000}{2000} = 2;$$

$$K = \frac{2,3}{2} = 1,15.$$

Зважаючи на розрахунки, можна зробити висновок, що нова розробка буде конкурентоспроможніше, ніж конкурентний товар.

Розробка системи електронного документообігу з QR-підписом сприяє підвищенню ефективності роботи організацій, зменшенню бюрократичного навантаження та покращенню доступності цифрових сервісів для користувачів. Завдяки переходу від паперових процесів до електронних, скорочується час обробки документів, підвищується продуктивність праці. Використання QR-підпису дозволяє забезпечити прозорість і достовірність документів без необхідності фізичного підписання, що підвищує довіру до цифрових процесів та зменшує ризики помилок чи підробок.

5.2 Прогнозування витрат на виконання науково-дослідної роботи

Витрати, пов'язані з проведенням науково-дослідної роботи групуються за такими статтями: витрати на оплату праці, витрати на соціальні заходи, матеріали, паливо та енергія для науково-виробничих цілей, витрати на службові відрядження, програмне забезпечення для наукових робіт, інші витрати, накладні витрати.

Основна заробітна плата кожного із дослідників Z_o , якщо вони працюють в наукових установах бюджетної сфери визначається за формулою:

$$Z_o = \frac{M}{T_p} * t \text{ (грн)} \quad (5.6)$$

де M — місячний посадовий оклад конкретного розробника (інженера, дослідника, науковця тощо), грн.;

T_p — число робочих днів в місяці; приблизно $T_p \approx 21...23$ дні;

t — число робочих днів роботи дослідника.

Зведемо сумарні розрахунки до таблиці 5.6.

Таблиця 5.6 – Заробітна плата дослідників в науковій установі бюджетної сфери

Найменування посади	Місячний посадовий оклад, грн.	Оплата за робочий день, грн.	Число днів роботи	Витрати на заробітну плату грн.
Керівник	17000	809,5	5	4048
Програміст	20000	952,4	21	20000
Всього				24048

Витрати на основну заробітну плату робітників (Z_p) за відповідними найменуваннями робіт розраховують за формулою:

$$Z_p = \sum_{i=1}^n C_i \cdot t_i, \quad (5.7)$$

де C_i — погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

t_i — час роботи робітника на виконання певної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду C_i можна визначити за формулою:

$$C_i = \frac{M_M \cdot K_i \cdot K_c}{T_p \cdot t_{зм}}, \quad (5.8)$$

де M_M — розмір прожиткового мінімуму працездатної особи або мінімальної місячної заробітної плати (залежно від діючого законодавства), грн;

K_i — коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду;

K_c — мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати.

T_p — середня кількість робочих днів в місяці, приблизно $T_p = 21 \dots 23$ дні;

$t_{зм}$ — тривалість зміни, год.

Таблиця 5.7 – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Погодинна тарифна ставка, грн	Величина оплати на робітника, грн
1. Підготовчі	2	1	47,6	95,2
2. Монтажні	2	3	64,3	128,6
3. Інтеграційні	2	5	81,0	161,9
4. Налагоджувальні	6	2	52,4	314,3
5. Випробувальні	4	4	71,4	285,7
Всього				985,7

Додаткова заробітна плата Z_d всіх розробників та робітників, які приймали участь в розробці нового технічного рішення розраховується як 10 - 12 % від основної заробітної плати робітників.

На даному підприємстві додаткова заробітна плата начисляється в розмірі 11% від основної заробітної плати.

$$Z_d = (Z_o + Z_p) * \frac{H_{\text{дод}}}{100\%} \quad (5.9)$$

$$Z_d = 0,11 * (24048 + 985,7) = 2753,67 \text{ (грн)}$$

Нарахування на заробітну плату $H_{\text{ЗП}}$ дослідників та робітників, які брали участь у виконанні даного етапу роботи, розраховуються за формулою (5.10):

$$H_{\text{ЗП}} = (Z_o + Z_p + Z_d) * \frac{\beta}{100} \text{ (грн)} \quad (5.10)$$

де Z_o — основна заробітна плата розробників, грн.;

Z_d — додаткова заробітна плата всіх розробників та робітників, грн.;

Z_p — основну заробітну плату робітників, грн.;

β — ставка єдиного внеску на загальнообов'язкове державне соціальне страхування, % .

Дана діяльність відноситься до бюджетної сфери, тому ставка єдиного внеску на загальнообов'язкове державне соціальне страхування буде складати 22%, тоді:

$$H_{\text{ЗП}} = (24048 + 985,7 + 2753,67) * \frac{22}{100} = 6113,14 \text{ (грн)}$$

До статті «Сировина та матеріали» належать витрати на сировину, основні та допоміжні матеріали, інструменти, пристрої та інші засоби й предмети праці, які придбані у сторонніх підприємств, установ і організацій та витрачені на проведення досліджень за прямим призначенням згідно з нормами їх витрачання, а також витрачені придбані напівфабрикати, що підлягають монтажу або виготовленню й додатковій обробці в цій організації, чи дослідні зразки, що виготовляються виробниками за документацією наукової організації.

Витрати на матеріали (М) у вартісному вираженні розраховуються окремо для кожного виду матеріалів за формулою:

$$M = \sum_{i=1}^n H_j \cdot C_j \cdot K_j - \sum_{i=1}^n B_j \cdot C_{vj}, \quad (5.11)$$

де H_j — норма витрат матеріалу j -го найменування, кг;

n — кількість видів матеріалів;

C_j — вартість матеріалу j -го найменування, грн/кг;

K_j — коефіцієнт транспортних витрат, ($K_j = 1,1 \dots 1,15$);

B_j — маса відходів j -го найменування, кг;

C_{vj} — вартість відходів j -го найменування, грн/кг.

Проведені розрахунки зведені в таблицю 5.8.

Таблиця 5.8 — Витрати на матеріали

Найменування матеріалу, марка, тип, сорт	Ціна за 1 кг, грн	Норма витрат, шт	Вартість витраченого матеріалу, грн
Папір	180	1	180
Ручка	20	1	20
Блокнот	50	1	50
Флешка	300	1	300
З врахуванням коефіцієнта транспортування			605

Витрати на комплектуючі вироби (K_e), які використовують при дослідженні нового технічного рішення, розраховуються, згідно з їхньою номенклатурою, за формулою:

$$K_e = \sum_{j=1}^n H_j \cdot C_j \cdot K_j \quad (5.12)$$

де H_j — кількість комплектуючих j -го виду, шт.;

C_j — покупна ціна комплектуючих j -го виду, грн;

K_j — коефіцієнт транспортних витрат, ($K_j = 1,1 \dots 1,15$).

Проведені розрахунки бажано звести до таблиці 5.9.

Таблиця 5.9 — Витрати на комплектуючі

Найменування комплектуючих	Кількість, шт.	Ціна за штуку, грн	Сума, грн
Маршрутизатор (бюджетний для тестів)	1	700	700
Кабелі мережеві (комплект)	1	150	150
Додаткові дрібні компоненти (адаптери)	1	100	100
Всього з врахуванням транспортних витрат			1045,00

Балансову вартість програмного забезпечення розраховують за формулою:

$$B_{npz} = \sum_{i=1}^k C_{inpg} \cdot C_{npz.i} \cdot K_i, \quad (5.13)$$

де C_{inpg} — ціна придбання одиниці програмного засобу даного виду, грн;

$C_{npz.i}$ — кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

K_i — коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($K_i = 1, 10 \dots 1, 12$);

k — кількість найменувань програмних засобів.

Отримані результати необхідно звести до таблиці:

Таблиця 5.10 — Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Домен (1 рік)	1	300	300
Всього з врахуванням налагодження			330

Балансову вартість спекулятивання розраховують за формулою:

$$B_{спец} = \sum_{i=1}^k C_i \cdot C_{np.i} \cdot K_i, \quad (5.14)$$

де C_i — ціна придбання одиниці спецустаткування даного виду, марки, грн;

$C_{np.i}$ — кількість одиниць устаткування відповідного найменування, які придбані для проведення досліджень, шт.;

K_i — коефіцієнт, що враховує доставку, монтаж, налагодження устаткування тощо, ($K_i = 1,10...1,12$);

k — кількість найменувань устаткування.

Отримані результати необхідно звести до таблиці:

Таблиця 5.11 — Витрати на придбання спецустаткування по кожному виду

Найменування устаткування	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Зовнішній жорсткий диск (для бекапів) 1 ТБ	1	1200	1200
Кабелі/адаптери (комплект)	1	200	200
Всього			1400
			1540

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо, можуть бути розраховані з використанням прямолінійного методу амортизації за формулою:

$$A_{обл} = \frac{C_{б.}}{T_{г}} \cdot \frac{t_{вик}}{12}, \quad (5.15)$$

де $C_{б.}$ — балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{вик}$ — термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_{г}$ — строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

Проведені розрахунки необхідно звести до таблиці 5.12.

Таблиця 5.12 — Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
Комп'ютер	25000	2	3	1041,67
Всього				1041,67

До статті «Паливо та енергія для науково-виробничих цілей» відносяться витрати на всі види палива й енергії, що безпосередньо використовуються з технологічною метою на проведення досліджень.

$$B_e = \sum_{i=1}^n \frac{W_{yt} \cdot t_i \cdot C_e \cdot K_{впі}}{\eta_i} \quad (5.16)$$

де W_{yt} — встановлена потужність обладнання на певному етапі розробки, кВт;

t_i — тривалість роботи обладнання на етапі дослідження, год;

C_e — вартість 1 кВт-години електроенергії, грн;

$K_{впі}$ — коефіцієнт, що враховує використання потужності, $K_{впі} < 1$;

η_i — коефіцієнт корисної дії обладнання, $\eta_i < 1$.

Для написання магістерської роботи використовується персональний комп'ютер для якого розрахуємо витрати на електроенергію.

$$B_e = \frac{0,5 \cdot 180 \cdot 12,69 \cdot 0,5}{0,8} = 713,81$$

Витрати за статтею «Службові відрядження» розраховуються як 20...25% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{св} = (Z_o + Z_p) * \frac{N_{св}}{100\%}, \quad (4.17)$$

де $N_{св}$ — норма нарахування за статтею «Службові відрядження».

$$B_{CB} = 0,2 * (24048 + 985,7) = 5006,67$$

Накладні (загальновиробничі) витрати $B_{нзв}$ охоплюють: витрати на управління організацією, оплата службових відряджень, витрати на утримання, ремонт та експлуатацію основних засобів, витрати на опалення, освітлення, водопостачання, охорону праці тощо. Накладні (загальновиробничі) витрати $B_{нзв}$ можна прийняти як (100...150)% від суми основної заробітної плати розробників та робітників, які виконували дану МКНР, тобто:

$$B_{нзв} = (Z_o + Z_p) \cdot \frac{H_{нзв}}{100\%}, \quad (5.18)$$

де $H_{нзв}$ — норма нарахування за статтею «Інші витрати».

$$B_{нзв} = (24048 + 985,7) \cdot \frac{100}{100\%} = 25033,33 \text{ грн}$$

Сума всіх попередніх статей витрат дає витрати, які безпосередньо стосуються даного розділу МКНР

$$B = 24048 + 985,7 + 2753,67 + 6113,14 + 605 + 1045 + 330 + 1540 + 1041,67 + 713,81 + 5006,67 + 25033,33 = 69215,62 \text{ грн}$$

Прогнозування загальних втрат ZB на виконання та впровадження результатів виконаної МКНР здійснюється за формулою:

$$ZB = \frac{B}{\eta}, \quad (5.19)$$

де η — коефіцієнт, який характеризує стадію виконання даної НДР.

Оскільки, робота знаходиться на стадії науково-дослідних робіт, то коефіцієнт $\beta = 0,7$.

Звідси:

$$ЗВ = \frac{69215,62}{0,7} = 98879,46 \text{ грн.}$$

5.3 Розрахунок економічної ефективності науково-технічної розробки

У даному підрозділі кількісно спрогнозуємо, яку вигоду, зиск можна отримати у майбутньому від впровадження результатів виконаної наукової роботи. Розрахуємо збільшення чистого прибутку підприємства $\Delta\Pi_i$, для кожного із років, протягом яких очікується отримання позитивних результатів від впровадження розробки, за формулою

$$\Delta\Pi_i = \sum_1^n (\Delta\Pi_o \cdot N + \Pi_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\nu}{100}\right) \quad (5.20)$$

де $\Delta\Pi_o$ — покращення основного оціночного показника від впровадження результатів розробки у даному році;

N — основний кількісний показник, який визначає діяльність підприємства у даному році до впровадження результатів наукової розробки;

ΔN — покращення основного кількісного показника діяльності підприємства від впровадження результатів розробки;

Π_o — основний оціночний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки;

n — кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки;

λ — коефіцієнт, який враховує сплату податку на додану вартість. Ставка податку на додану вартість дорівнює 20%, а коефіцієнт $\lambda = 0,8333$;

ρ — коефіцієнт, який враховує рентабельність продукту. $\rho = 0,25$;

ν — ставка податку на прибуток. У 2025 році – 18%.

Припустимо, що ціна зростає на 500 грн. Кількість одиниць реалізованої продукції також збільшиться: протягом першого року на 200 шт., протягом другого року – на 250 шт., протягом третього року на 350 шт. Реалізація продукції до

впровадження розробки складала 1 шт., а її ціна до 2000 грн. Розрахуємо прибуток, яке отримає підприємство протягом трьох років.

$$\Delta\Pi_1 = [500 \cdot 1 + (2000 + 500) \cdot 200] \cdot 0,833 \cdot 0,25 \cdot \left(1 + \frac{18}{100}\right) = 85498,66 \text{ грн.}$$

$$\begin{aligned} \Delta\Pi_2 &= [500 \cdot 1 + (2000 + 500) \cdot (200 + 250)] \cdot 0,833 \cdot 0,25 \cdot \left(1 + \frac{18}{100}\right) \\ &= 192679,81 \text{ грн.} \end{aligned}$$

$$\begin{aligned} \Delta\Pi_3 &= [500 \cdot 1 + (2000 + 500) \cdot (200 + 250 + 350)] \cdot 0,833 \cdot 0,25 \cdot \left(1 + \frac{18}{100}\right) \\ &= 342153 \text{ грн.} \end{aligned}$$

5.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності

Розрахуємо основні показники, які визначають доцільність фінансування наукової розробки певним інвестором, є абсолютна і відносна ефективність вкладених інвестицій та термін їх окупності.

Розрахуємо величину початкових інвестицій PV , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки.

$$PV = k_{\text{інв}} \cdot ЗВ, \quad (5.21)$$

$k_{\text{інв}}$ — коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію. Це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо ($k_{\text{інв}} = 2 \dots 5$).

$$PV = 2 \cdot 98879,46 = 197758,91$$

Розрахуємо абсолютну ефективність вкладених інвестицій $E_{\text{абс}}$ згідно наступної формули:

$$E_{\text{абс}} = (ПП - PV) \quad (5.22)$$

де $ПП$ — приведена вартість всіх чистих прибутків, що їх отримає підприємство від реалізації результатів наукової розробки, грн.;

$$ПП = \sum_1^T \frac{\Delta\Pi_i}{(1 + \tau)^t}, \quad (5.23)$$

де $\Delta\Pi_i$ — збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої НДЦКР, грн.;

T — період часу, протягом якою виявляються результати впровадженої НДЦКР, роки;

τ — ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні; для України цей показник знаходиться на рівні 0,2;

t — період часу (в роках).

$$ПП = \frac{85498,66}{(1 + 0,2)^1} + \frac{192679,81}{(1 + 0,2)^2} + \frac{342153}{(1 + 0,2)^3} = 403980,47 \text{ грн.}$$

$$E_{абс} = (403980,47 - 210143,02) = 206221,56 \text{ грн.}$$

Оскільки $E_{абс} > 0$ то вкладання коштів на виконання та впровадження результатів НДЦКР може бути доцільним.

Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій $E_{в}$. Для цього користуються формулою:

$$E_{в} = \sqrt[T_{жс}]{\left(1 + \frac{E_{абс}}{PV}\right)} - 1, \quad (5.24)$$

$T_{жс}$ — життєвий цикл наукової розробки, роки.

$$E_{в} = \sqrt[3]{1 + \frac{206221,56}{197758,91}} - 1 = 0,46 = 46\%$$

Визначимо мінімальну ставку дисконтування, яка у загальному вигляді визначається за формулою:

$$\tau = d + f, \quad (5.25)$$

де d — середньозважена ставка за депозитними операціями в комерційних банках; в 2025 році в Україні $d = (0,14 \dots 0,2)$;

f — показник, що характеризує ризикованість вкладень; зазвичай, величина $f = (0,05 \dots 0,1)$.

$$\tau_{\min} = 0,18 + 0,05 = 0,23$$

Так як $E_g > \tau_{\min}$ то інвестор може бути зацікавлений у фінансуванні даної наукової розробки.

Розрахуємо термін окупності вкладених у реалізацію наукового проекту інвестицій за формулою:

$$T_{ок} = \frac{1}{E_g} \quad (5.26)$$

$$T_{ок} = \frac{1}{0,46} = 2,2 \text{ роки}$$

Так як $T_{ок} \leq 3 \dots 5$ -ти років, то фінансування даної наукової розробки в принципі є доцільним.

ВИСНОВКИ

У магістерській роботі виконано комплексне дослідження технологій електронного документообігу та запропоновано підхід до підвищення рівня захищеності та зручності роботи з цифровими документами за допомогою модуля QR-підпису. Аналіз існуючих систем показав, що велика частина сучасних рішень потребує використання сертифікаційних центрів, апаратних ключів або закритої інфраструктури, що ускладнює процес перевірки документів сторонніми особами та збільшує витрати на впровадження. Це визначило актуальність створення системи, що забезпечує незалежну, швидку та зрозумілу верифікацію без додаткових технічних засобів.

У роботі сформовано концептуальну архітектуру електронного документообігу з модулем QR-підпису, яка поєднує механізми хешування, генерацію унікального цифрового підпису, створення QR-коду та перевірку автентичності документа. Побудовані математичні моделі, алгоритми функціонування та структурні схеми дозволили визначити логіку роботи системи та забезпечити узгодженість між компонентами серверної частини, базою даних і користувацьким інтерфейсом. Реалізація системи за допомогою Django, PostgreSQL та Docker забезпечила модульність, масштабованість і можливість розгортання в реальних умовах.

Інтегрований у систему модуль QR-підпису забезпечує формування цифрового відбитку документа, автоматичне накладення адміністративного підпису та генерацію QR-коду, який дозволяє здійснювати перевірку автентичності через веб-сторінку без використання спеціалізованих інструментів. Проведене тестування показало стабільність роботи системи, правильність обробки документів, коректність перевірки QR-підпису та виявлення підроблених або змінених файлів. Результати дослідження підтвердили, що впроваджений механізм QR-підпису забезпечує достатній рівень безпеки та простоту використання, а запропонована архітектура може застосовуватися як у корпоративному, так і в державному секторі.

Розроблена система електронного документообігу здатна забезпечувати контроль цілісності, автентичності та незалежної перевірки документів. Система може бути розширена додатковими функціями, такими як удосконалений журнал подій, багаторівневе підписання, підключення зовнішніх сервісів і поглиблені засоби криптографічного захисту.

Оцінка комерційного потенціалу розробленої системи електронного документообігу показала економічну доцільність її впровадження, термін окупності витрат складатиме 2,2 роки

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Карпенко М. Ю. Системи електронного документообігу : конспект лекцій. Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. Харків : ХНУМГ ім. О. М. Бекетова, 2021. 68 с.
2. Ситник І. П., Мельниченко А. І. Системи електронного документообігу в електронному бізнесі. Науковий вісник Ужгородського національного університету. Вип. 4. 2015. С. 174-178.
3. Клейманов І. О., Тарновський М. Г. СИСТЕМА ЕЛЕКТРОННОГО ДОКУМЕНТООБІГУ З ІНТЕГРОВАНИМ МОДУЛЕМ БЕЗПЕЧНОГО QR-ПІДПИСУ // Матеріали всеукраїнської науково-практичної Інтернет-конференції студентів, аспірантів та молодих науковців «Молодь в науці: дослідження, проблеми, перспективи (МН-2026)». Вінниця. Режим доступу: <https://conferences.vntu.edu.ua/index.php/mn/mn2026/paper/view/26642>
4. Клименко І. В., Линьов К.О. Системи електронного документообігу в державному управлінні: Навч. метод. пос. Київ: Вид-во НАДУ, 2006. 32 с.
5. Lee, K., & Park, S. Database-driven Document Indexing for Efficient Retrieval. Information Systems, 2022, 12–28 с.
6. Johnson, P., & Evans, R. Audit Trails and Logging in Electronic Document Management. Journal of Digital Security, 2020, 101–117 с.
7. Müller, H., & Wagner, T. Enterprise Systems Integration for Automated Document Processing. International Journal of Business Information Systems, 2022, 210–230 с.
8. Smith, J., & Brown, A. Cloud-based Document Management Systems. Journal of Information Technology, 2021, 45–58 с.
9. Chen, L., & Zhao, Y. Secure Digital Signatures and QR-codes in Document Management. Computers & Security, 2021, 110 с.
10. Yli-Huumo, J., Ko, D., & Choi, S. Blockchain-Based Document Verification and Secure Storage Systems. Future Generation Computer Systems, 2021, pp. 693–703 с.

11. Електронний документообіг DocuWare, система електронного документообігу (СЕД) веб-сайт. URL: <https://start.docuware.com/> (дата звернення: 23.09.2025)

12. Електронний документообіг М.Е.Дос, система електронного документообігу (СЕД) веб-сайт. URL: <https://medoc.ua/> (дата звернення: 23.10.2025)

13. Електронний документообіг eDocs, система електронного документообігу (СЕД) веб-сайт. URL: <https://e-docs.ua/> (дата звернення: 23.10.2025)

14. Електронний документообіг Vchasno, система електронного документообігу (СЕД) веб-сайт. URL: <https://vchasno.ua/> (дата звернення: 23.10.2025)

15. Азаров О. Д., Захарченко С. М., Кадук О. В. Комп'ютерні мережі: підручник. — Вінниця: ВНТУ, 2020. — 378 с.

16. ДСТУ 4145:2023. Інформаційні технології. Криптографічний захист інформації. Електронний цифровий підпис. Формування та перевіряння. — Київ: ДП «УкрНДНЦ», 2023.

17. Кузьмін В. П. Системи електронного документообігу: архітектура, технології, безпека. — Львів: ЛНУ ім. І. Франка, 2021. — 256 с.

18. IEEE Xplore. QR Code-based Digital Signature Systems for Secure Document Authentication. — IEEE Transactions on Information Forensics and Security, 2023. 110–130 с.

19. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. — Вінниця : ВНТУ, 2021. — 42 с.

20. Методичні вказівки до виконання магістерських кваліфікаційних робіт студентами спеціальності 123 «Комп'ютерна інженерія». / Укладачі О.Д. Азаров, О.В. Дудник, С.І. Швець — Вінниця: ВНТУ, 2023. — 57 с.

ДОДАТОК А

Технічне завдання

Міністерство освіти і науки України

Вінницький національний технічний університет

Факультет інформаційних технологій та комп'ютерної інженерії

Кафедра обчислювальної техніки

ЗАТВЕРДЖУЮ

Завідувач кафедри ОТ

проф., д.т.н.. Азаров О.Д.

" 3 " жовтня 2025 р.**ТЕХНІЧНЕ ЗАВДАННЯ**

на виконання магістерської дипломної роботи

«Система електронного документообігу з інтегрованим модулем безпечного
QR-підпису».

Науковий керівник к.т.н. доц. каф. ОТ

Тарновський М. Г

Студент групи 1КІ-24м

Клейманов І.О.

1 Підстава для виконання магістерської кваліфікаційної роботи (МКР)

1.1 Актуальність досліджень полягає в тому, що системи електронного документообігу дозволяють автоматизувати процеси створення, зберігання, обробки та обміну документами в електронному форматі, що значно підвищує ефективність роботи організацій. Незважаючи на значний прогрес в області створення таких систем потребують вирішення проблеми ефективної перевірки достовірності та автентичності документів, що циркулюють у мережевому просторі.

1.2 Наказ про затвердження теми МКР №313 від 24.09.2025 р.

2 Мета і призначення МКР

2.1 Метою роботи є вдосконалення програмного засобу для ефективного оброблення та зберігання даних електронних документів із застосуванням модуля безпечного QR-підпису, що підтверджує автентичність документів та захищає їх від підробки.

2.2 Призначення розробки — програмна реалізація системи електронного документообігу та модуля підписання документів за допомогою QR-коду.

3 Вихідні дані для виконання МКР

3.1 Предметна область — електронний документообіг.

3.2 Інтеграція з базою даних документів, модулем авторизації користувачів, підсистемою генерації QR-кодів, підтримка REST API для обміну даними між компонентами.

3.3 Функціональність — завантаження й зберігання документів, сортування та структуризація документів, створення персональних розділів документів, журнал дій користувача (логування), автоматичне формування QR-підпису для кожного документа.

3.4 Інструментальні засоби розробки — фреймворк Django (серверна частина), PostgreSQL (база даних), Docker (контейнеризація), мова програмування Python 3.10.

4 Технічні вимоги до виконання МКР

Технічні вимоги:

- проведення аналізу технологій електронного документообігу та стандартів електронного цифрового підпису;
- розробка алгоритмів обробки, зберігання та перевірки автентичності документів;
- програмна реалізація системи електронного документообігу;
- тестування та перевірка функціональності розробленої системи;
- оцінювання економічної ефективності запропонованого рішення.

5 Етапи МКР та очікувані результати

Етапи роботи та очікувані результати приведено в Таблиці А.1

Таблиця А.1 — Етапи МКР

№ з/п	Назва етапу	Термін виконання		Очікувані результати
		початок	закінчення	
1	Формування мети, завдань та обґрунтування актуальності теми дослідження.	08.09.25	08.09.25	Задачі дослідження
2	Аналіз наукових джерел, нормативних документів та існуючих систем електронного документообігу.	11.09.25	17.09.25	Розділ 1
3	Проектування архітектури системи та визначення структури програмних модулів.	18.09.25	25.09.25	Розділ 2
4	Розробка механізму перевірки QR-підпису та створення сторінки підтвердження підпису документів.	27.09.25	12.10.25	Розділ 3
5	Проведення тестування працездатності системи.	13.10.25	17.10.25	Розділ 4

Продовження таблиці А.1

№ з/п	Назва етапу	Термін виконання		Очікувані результати
		початок	закінчення	
6	Виконання економічної частини	20.10.25	25.10.25	Розділ 5
7	Оформлення матеріалів до захисту МКР	початок	закінчення	ПЗ, графічний матеріал і презентація
8	Перевірка якості виконання магістерської роботи та усунення недоліків	25.11.25	28.10.25	Оформлені документи
9	Підписи супроводжувальних документів у нормоконтролера, керівника, опонента	01.12.25	03.12.25	Оформлені документи
10	Перевірка на антиплагіат та ШІ	04.12.25	05.11.25	Оформлені документи

6 Матеріали, що подаються до захисту МКР

До захисту подаються: пояснювальна записка МКР, графічні і ілюстративні матеріали, протокол попереднього захисту МКР на кафедрі, відгук наукового керівника, відгук опонента, протоколи складання державних екзаменів, анотації до МКР українською та іноземною мовами.

7 Порядок контролю виконання та захисту МКР

Виконання етапів графічної та розрахункової документації МКР контролюється науковим керівником згідно зі встановленими термінами. Захист МКР відбувається на засіданні Екзаменаційної комісії, затвердженої наказом ректора.

8 Вимоги до оформлювання та порядок виконання МКР

8.1 При оформлюванні МКР використовуються:

— ДСТУ 3008: 2015 «Звіти в сфері науки і техніки. Структура та правила оформлювання»;

— ДСТУ 8302: 2015 «Бібліографічні посилання. Загальні положення та правила складання»;

— ГОСТ 2.104-2006 «Єдина система конструкторської документації. Основні написи»;

— методичні вказівки до виконання магістерських кваліфікаційних робіт зі спеціальності 123 — «Комп'ютерна інженерія»;

— документи на які посилаються у вище вказаних.

8.2 Порядок виконання МКР викладено в «Положення про кваліфікаційні роботи на другому (магістерському) рівні вищої освіти СУЯ ВНТУ-03.02.02-П.001.01:21».

ДОДАТОК Б

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Назва роботи: Система електронного документообігу з інтегрованим модулем безпечного QR-підпису

Тип роботи: магістерська кваліфікаційна робота

(бакалаврська кваліфікаційна робота / магістерська кваліфікаційна робота)

Підрозділ кафедра обчислювальної техніки, ФІТКІ, 1КІ-24м

(кафедра, факультет, навчальна група)

Коефіцієнт подібності текстових запозичень, виявлених у роботі системою StrikePlagiarism (КП1) 3 %

Висновок щодо перевірки кваліфікаційної роботи (відмітити потрібне)

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак академічного плагіату, фабрикації, фальсифікації. Роботу прийняти до захисту.
- У роботі не виявлено ознак плагіату, фабрикації, фальсифікації, але надмірна кількість текстових запозичень та/або наявність типових розрахунків не дозволяють прийняти рішення про оригінальність та самостійність її виконання. Роботу направити на доопрацювання.
- У роботі виявлено ознаки академічного плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень. Робота до захисту не приймається.

Експертна комісія:

Азаров О. Д., д.т.н., зав. каф. ОТ

(прізвище, ініціали, посада)

_____ (підпис)

Мартинюк Т. Б., д.т.н., проф. каф. ОТ

(прізвище, ініціали, посада)

_____ (підпис)

Особа, відповідальна за перевірку _____

(підпис)

Захарченко С. М.

(прізвище, ініціали)

З висновком експертної комісії ознайомлений(-на)

Керівник _____

(підпис)

Тарновський М. Г., доц. каф. ОТ

(прізвище, ініціали, посада)

Здобувач _____

(підпис)

Клейманов І. О

(прізвище, ініціали)

ДОДАТОК В

Архітектура схема системи електронного документообігу з модулем QR-підпису

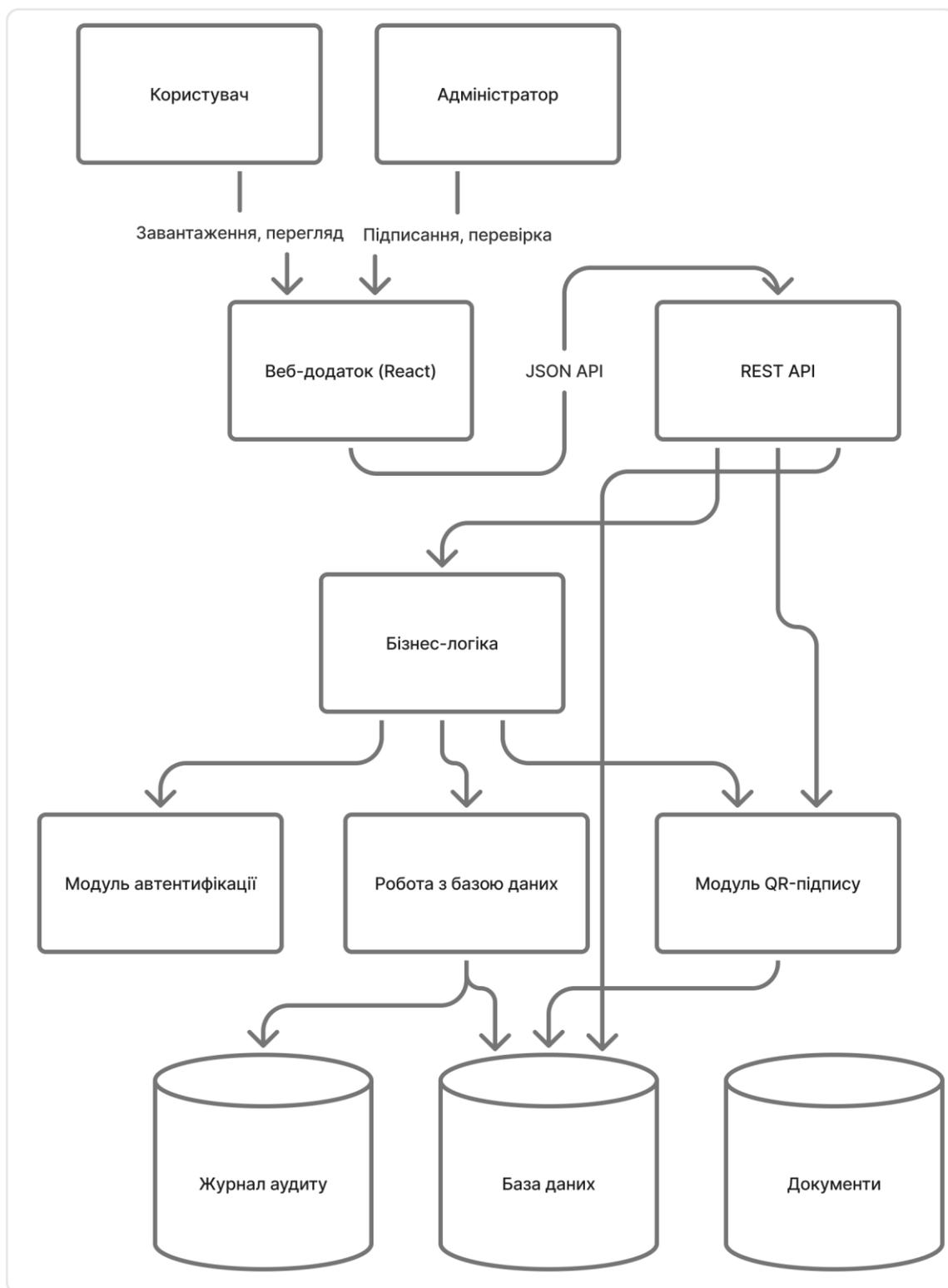


Рисунок В.1 — Архітектурна схема системи електронного документообігу з QR-підписом

ДОДАТОК Г

Блок-схема процесу перевірки документа за допомогою модуля QR-підпису



Рисунок Г.1 — Блок-схема процесу перевірки документа за допомогою модуля QR-підпису

ДОДАТОК Д

UML-діаграма класів

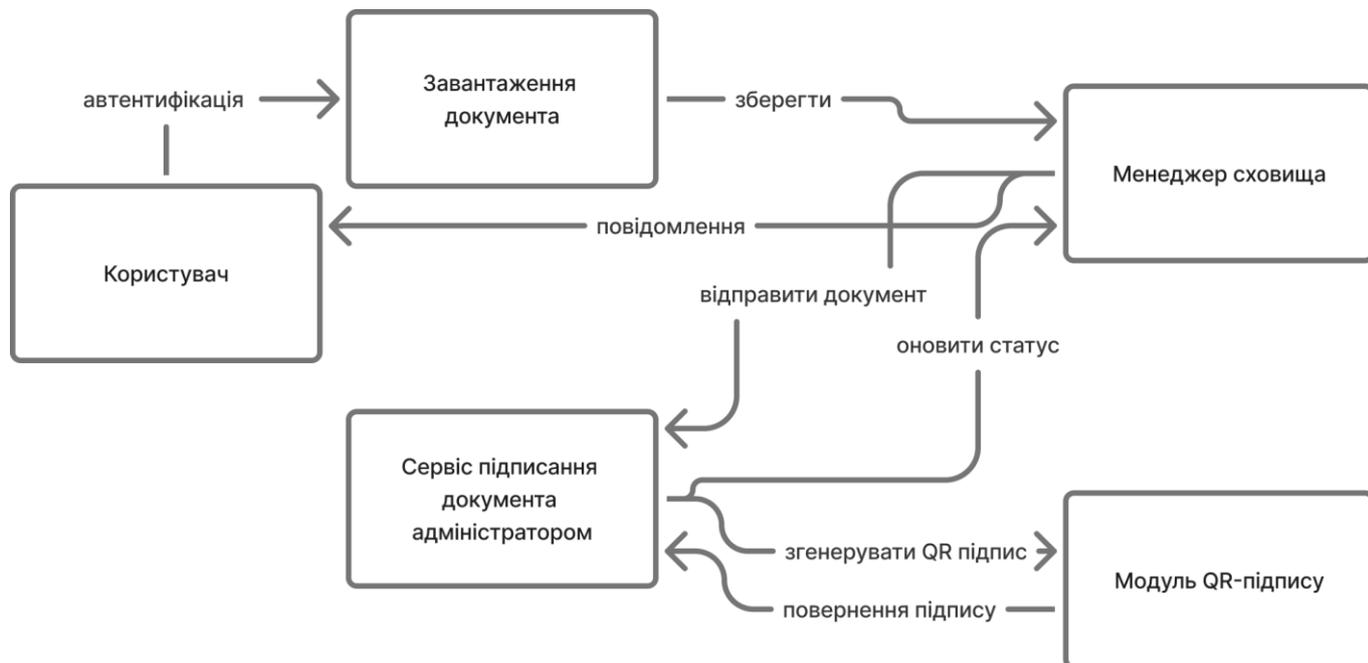


Рисунок Д.1 — UML-діаграма класів

ДОДАТОК Е

Лістинг програми модуля безпечного QR-коду

```
import json
from datetime import datetime

from pathlib import Path

from typing import Tuple

from uuid import uuid4

from crypto_manager import CryptoManager
from document_hash import DocumentHash

from qr_utils import generate_qr_image, stylize_qr_with_label, paste_qr_to_pdf_page_as_image

from PIL import Image

import tempfile

import shutil

import subprocess.

class QRSignModule:

    def __init__(self, storage_dir: str, crypto: CryptoManager | None = None):

        self.storage_dir = Path(storage_dir)

        self.storage_dir.mkdir(parents=True, exist_ok=True)

        self.crypto = crypto or CryptoManager()

    def _make_payload(self, document_id: int, doc_hash: str, admin_user: str, signed_at: str) -> dict:

        return {

            "doc": document_id,

            "hash": doc_hash,

            "admin": admin_user,

            "signed_at": signed_at,

            "nonce": uuid4().hex

        }

    def sign_document_and_embed(self, document_path: str, document_id: int, admin_user: str) -> dict:

        doc_hash = DocumentHash.sha256_of_file(document_path)

        signed_at = datetime.utcnow().isoformat()

        payload = self._make_payload(document_id, doc_hash, admin_user, signed_at)

        payload_json = json.dumps(payload, ensure_ascii=False).encode("utf-8")

        token = self.crypto.encrypt_and_protect(payload_json)
```

```

verification_url = f"https://verificationqrcode.netlify.app/?token={token}"
qr_img = generate_qr_image(verification_url, box_size=6, border=2)
label = f"Signed by {admin_user} {signed_at}"
styled_qr = stylize_qr_with_label(qr_img, label=label)
out_qr_path = self.storage_dir / f"qr_{document_id}_{uuid4().hex}.png"
styled_qr.convert("RGB").save(out_qr_path, format="PNG")
out_pdf_path = self.storage_dir / f"signed_{Path(document_path).name}"
success_embed = False

try:
    tmp_dir = Path(tempfile.mkdtemp())
    png_base = tmp_dir / "page"
    subprocess.run(["pdftoppm", "-singlefile", "-png", document_path, str(png_base)], check=True)
    page_png = tmp_dir / "page.png"
    if page_png.exists():
        page_img = Image.open(page_png).convert("RGBA")
        pos = (page_img.width - styled_qr.width - 20, page_img.height - styled_qr.height - 20)
        merged = paste_qr_to_pdf_page_as_image(page_img, styled_qr, pos)
        merged_rgb = merged.convert("RGB")
        merged_pdf_path = tmp_dir / "merged_page.pdf"
        merged_rgb.save(merged_pdf_path, "PDF", resolution=100.0)
        shutil.copy(merged_pdf_path, out_pdf_path)
        success_embed = True
    shutil.rmtree(tmp_dir)
except Exception:
    success_embed = False

return {
    "token": token,
    "verification_url": verification_url,
    "qr_path": str(out_qr_path),
    "signed_pdf_path": str(out_pdf_path) if success_embed else None,
    "payload": payload
}

```

ДОДАТОК Ж

Лістинг програми модуля підписання документа

```
from pathlib import Path
from django.contrib.auth import get_user_model
from app.services.qr_sign_module import QRSignModule
from app.services.event_log_service import EventLogService
from app.models import Document, QRSignature
User = get_user_model()
def sign_doc_flow(document_id: int, admin_user_id: int):
    doc = Document.objects.get(id=document_id)
    admin = User.objects.get(id=admin_user_id)
    qr_module = QRSignModule(storage_dir="/var/www/myapp/storage/qr")
    res = qr_module.sign_document_and_embed(str(doc.file.path), document_id, admin.username)
    qr_record = QRSignature.objects.create(
        document=doc,
        token=res["token"],
        verification_url=res["verification_url"],
        qr_image=res["qr_path"] if res["qr_path"] else None,
        signed_by=admin,
        signed_at=res["payload"]["signed_at"]
    )
    doc.current_hash = res["payload"]["hash"]
    doc.status = "signed"
    doc.save()
    return qr_record, res
```

ДОДАТОК И

Лістинг програми модуля журналу подій

```

from django.db import models
from django.contrib.auth import get_user_model

User = get_user_model()

class EventLog(models.Model):
    SEVERITY_CHOICES = [
        ('INFO', 'Info'),
        ('WARN', 'Warning'),
        ('ERROR', 'Error'),
        ('SEC', 'Security'),
        ('AUD', 'Audit'),
    ]

    timestamp = models.DateTimeField(auto_now_add=True, db_index=True)
    user = models.ForeignKey(User, null=True, blank=True, on_delete=models.SET_NULL)
    action = models.CharField(max_length=128)
    details = models.JSONField(null=True, blank=True) # структуровані дані
    ip_address = models.GenericIPAddressField(null=True, blank=True)
    severity = models.CharField(max_length=8, choices=SEVERITY_CHOICES, default='INFO')
    related_document_id = models.IntegerField(null=True, blank=True, db_index=True)
    note = models.TextField(null=True, blank=True)

    class Meta:
        ordering = ['-timestamp']
        indexes = [
            models.Index(fields=['related_document_id']),
            models.Index(fields=['severity']),
        ]

    def __str__(self):
        return f" {self.timestamp} {self.user} {self.action}"

# services/event_log_service.py

import csv

from io import StringIO

```

```

from django.utils import timezone

from .models import EventLog

class EventLogService:

    @staticmethod

    def log_event(user=None, action:str="unknown", details:dict=None,

                 ip_address:str=None, severity:str="INFO",

                 related_document_id:int=None, note:str=None) -> EventLog:

        if details is None:

            details = { }

        ev = EventLog.objects.create(

            user=user,

            action=action,

            details=details,

            ip_address=ip_address,

            severity=severity,

            related_document_id=related_document_id,

            note=note

        )

        return ev

    @staticmethod

    def query_events(start_ts=None, end_ts=None, severity=None, user=None, doc_id=None,

                    action_contains=None):

        qs = EventLog.objects.all()

        if start_ts:

            qs = qs.filter(timestamp__gte=start_ts)

        if end_ts:

            qs = qs.filter(timestamp__lte=end_ts)

        if severity:

            qs = qs.filter(severity=severity)

        if user:

            qs = qs.filter(user=user)

        if doc_id:

            qs = qs.filter(related_document_id=doc_id)

```

```
if action_contains:
    qs = qs.filter(action__icontains=action_contains)
return qs.order_by('-timestamp')

@staticmethod
def export_to_csv(queryset) -> str:
    output = StringIO()
    writer = csv.writer(output)
    writer.writerow(['timestamp','user','action','severity','related_document_id','ip_address','details','note'])
    for ev in queryset:
        writer.writerow([
            ev.timestamp.isoformat(),
            ev.user.username if ev.user else "",
            ev.action,
            ev.severity,
            ev.related_document_id or "",
            ev.ip_address or "",
            json.dumps(ev.details, ensure_ascii=False),
            ev.note or ""
        ])
    return output.getvalue()
```