

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА
Навчально-науковий інститут
«Інститут інформаційних технологій в економіці»
Кафедра інформаційних систем в економіці**

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»
галузь знань 12 «Інформаційні технології»
спеціальність 122 «Комп'ютерні науки»

Форма навчання: денна

КВАЛІФІКАЦІЙНА БАКАЛАВРСЬКА РОБОТА

на тему: «Проектування автоматизованої системи обліку та моніторингу
матеріальних запасів складу»

здобувача Горніча Антона Миколайовича _____

Науковий керівник:

д. ф.-м. н., проф.

_____Миронцов М.Л.

**Робота допущена до захисту перед
екзаменаційною комісією з**

атестації здобувачів вищої освіти

завідувач кафедри:

к.е.н., доцент

_____Тішков Б.О.

Київ 2025

Міністерство освіти і науки України
Київський національний економічний університет імені Вадима Гетьмана
Навчально-науковий інститут «Інститут інформаційних технологій в економіці»

Кафедра інформаційних систем в економіці

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»

галузь знань 12 «Інформаційні технології»

спеціальність 122 «Комп'ютерні науки»

ПОГОДЖЕНО:

Керівник проектної групи(гарант)
освітньо-професійної програми

Помазун О.М.

“ ” 2025 р.

ЗАТВЕРДЖУЮ:

Завідувач кафедри

Тішков Б.О.

“ ” 2025 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

здобувачу вищої освіти ***Горніча Антона Микоалйовича***

очної (денної) форми навчання

на підготовку кваліфікаційної бакалаврської роботи
на тему: «Проектування автоматизованої системи обліку та моніторингу матеріальних запасів»

Тему затверджено наказом ректора Університету від « 7 » березня 2025 р.
 № 466- ст.

Кваліфікаційна бакалаврська робота виконується на матеріалах отриманих під час навчання, навчальних посібників, матеріалів з мережі Інтернет, а також зібраного фактичного матеріалу з обраної теми дослідження під час виконання курсових робіт.

План кваліфікаційної бакалаврської роботи

Розділ I Характеристика та аналіз предметної області

Розділ II Розробка вимог і моделювання інформаційної системи

Розділ III Проектування та реалізація компонентів системи

Об'єкт дослідження Проектування автоматизованої систему обліку та моніторингу матеріальних запасів

Предмет дослідження Методи та засоби розробки автоматизованої інформаційної системи обліку та моніторингу матеріальних запасів складу.

Мета кваліфікаційної бакалаврської роботи Розробка функціональної та зручної інформаційної системи для автоматизованого обліку та моніторингу матеріальних запасів, яка дозволяє ефективно управляти товарними залишками, контролювати терміни придатності, забезпечує генерацію звітів, підтримку сповіщень, налаштування прав доступу, а також можливість розширення аналітичного функціоналу і адаптації під потреби малого та середнього бізнесу.

Конкретні завдання, які здобувач повинен виконати для досягнення поставленої мети:

У розділі I надати загальну характеристику галузі складської логістики та управління матеріальними запасами, із акцентом на процес автоматизованого обліку товарів за допомогою сучасних веб-застосунків. Необхідно визначити об'єкт дослідження – інформаційну систему обліку та моніторингу складських запасів, а також структуру управління цим процесом з урахуванням потреб малого та середнього бізнесу. Особливу увагу слід приділити ключовим проблемам, які виникають під час організації та впровадження таких систем. Зокрема, йдеться про обмежений бюджет підприємств, недостатній рівень автоматизації, складність адаптації готових рішень до специфіки окремого складу, а також потребу в інтуїтивно зрозумілому інтерфейсі для непрофесійних користувачів.

У розділі II провести аналіз потреб цільової аудиторії, на основі чого сформулювати бізнес функціональні та нефункціональні вимоги до автоматизованої системи. Необхідно здійснити постановку задачі з урахуванням техніко-економічної доцільності автоматизації, розробити алгоритм основних бізнес-процесів, а також відповідну блок-схему. Слід побудувати математичну модель розв'язання задачі згідно з вимогами методичних рекомендацій кафедри. Окремо необхідно представити інформаційну модель системи: діаграми прецедентів, класів, зв'язків, матриці трасування. Слід описати структуру даних, вхідну й вихідну інформацію, її джерела.

У розділі III здійснити проектування компонентів ІС: інформаційного, технічного та програмного забезпечення. Представити реалізовані функції: облік товарів, контроль залишків, моніторинг термінів, генерацію звітів, сповіщення про критичні рівні.

**Завдання підготував
науковий керівник**

Миронцов Микита Леонідович
« 10 » березня 2025 р.

**Завдання одержав
здобувач**

Горніч Антон Миколайович
« 10 » березня 2025 р..

Відгук
 про кваліфікаційну бакалаврську роботу
 здобувача навчально-наукового інституту
 «Інститут інформаційних технологій в економіці»
 освітньо-професійної програми
 «Комп'ютерні науки»

Горніча Антона Миколайовича

на тему
***«Проектування автоматизованої системи обліку та моніторингу
 матеріальних запасів»***

1. ***Актуальність теми.*** Тематика є надзвичайно актуальною в умовах цифрової трансформації підприємств. Зростаюча потреба в ефективному управлінні складськими процесами обумовлює необхідність впровадження автоматизованих рішень. Тема відповідає сучасним тенденціям розвитку ІТ у сфері управління ресурсами.
2. ***Позитивні риси кваліфікаційної бакалаврської роботи.*** Робота відзначається системним підходом та якісною технічною реалізацією. Студент показав глибоке розуміння складської діяльності, проаналізував існуючі рішення та запропонував ефективну архітектуру хмароорієнтованої системи на основі React і Firebase.
3. ***Наявність самостійних розробок автора.*** Горніч А.М самостійно спроектував архітектуру, базу даних, реалізував основні модулі системи: облік запасів, контроль термінів, генерацію звітів, автоматичні сповіщення. Робота містить авторські діаграми, моделі та повну реалізацію веб-інтерфейсу.
4. ***Цінність теоретичних висновків та практичних рекомендацій.*** Висновки ґрунтуються на аналізі сучасних підходів до автоматизації. Рекомендації орієнтовані на потреби малого та середнього бізнесу, що підтверджує практичну значущість і потенціал комерційного використання.
5. ***Наявність недоліків.*** Система може бути в подальшому вдосконалена за рахунок покращення та розширення функціональних можливостей.
6. ***Загальна оцінка кваліфікаційної бакалаврської роботи та її допущення до захисту перед ЕК.*** Кваліфікаційна бакалаврська робота студента Горніча А.М. відповідає встановленим вимогам, рекомендується до захисту та заслуговує позитивної оцінки.

Науковий керівник

_____ проф., д.ф.-м.н. Миронцов М.Л.
 (підпис)

“ ____ ” _____ 2025 р.

Рецензія

на кваліфікаційну бакалаврську роботу
здобувача вищої освіти

Горніч Антон Миколайович

(прізвище, ім'я, по батькові)

Тема «Проектування автоматизованої системи обліку та моніторингу матеріальних запасів складу»

Актуальність теми кваліфікаційної бакалаврської роботи і доцільність її розроблення Тема є надзвичайно актуальною в умовах цифровізації бізнес-процесів. Розроблення хмароорієнтованої системи обліку складських запасів відповідає сучасним вимогам оптимізації логістики та підвищення ефективності складських операцій.

Якість проведеного дослідження Робота демонструє високий рівень технічної підготовки здобувача. Виконано детальний аналіз предметної області, обґрунтовано вибір архітектурних рішень та технологічного стеку (React, Firebase). Автор вдало поєднав теоретичні засади з практичною реалізацією.

Позитивні риси кваліфікаційної бакалаврської роботи Серед ключових переваг роботи варто відзначити застосування сучасних технологій для забезпечення масштабованості, реалізацію адаптивного та інтуїтивного інтерфейсу, підтримку хмарного зберігання та доступу в реальному часі, функціональність контролю термінів придатності та автоматичних сповіщень, а також орієнтацію на потреби малого та середнього бізнесу.

Зауваження Відсутність модулів прогнозування залишків та обмежене охоплення інтеграцій з ERP-рішеннями. Проте ці недоліки не впливають на загальну якість роботи.

Практична значимість висновків і рекомендацій Результати мають чітке практичне спрямування. Система має високий потенціал для впровадження та може стати основою для комерційного продукту з можливістю адаптації під різні типи підприємств.

Місце роботи та посада рецензента
Науковий ступінь, учене звання (за наявності)
ТОВ «ПРОВАЙД-ГРУП», директор

Підпис засвідчує:



(посада, підпис)

Директор ТОВ «ПРОВАЙД-ГРУП»
Станіслав СТРИЖАК

Місце печатки організації, де працює рецензент

АНОТАЦІЯ

кваліфікаційної бакалаврської роботи
здобувача першого (бакалаврського) рівня вищої освіти, 4 курсу,
виконаної на тему: «Проектування автоматизованої системи обліку та
моніторингу матеріальних запасів складу»

Київ: кафедра інформаційних систем в економіці, 2025 р.

Кваліфікаційна бакалаврська робота присвячена розробці автоматизованої інформаційної системи обліку та моніторингу матеріальних запасів складу, актуальність якої зумовлена потребою у цифровізації логістичних процесів і підвищенні ефективності управління складськими ресурсами, особливо в умовах малого та середнього бізнесу.

У роботі спроектовано та реалізовано веборієнтовану систему, що забезпечує облік надходжень, витрат і залишків товарів, контроль термінів зберігання, генерацію звітів і сповіщення про критичний рівень запасів. Система підтримує роботу з реальними даними в режимі реального часу завдяки використанню хмарної платформи Firebase і фреймворку React.

Робота складається з трьох логічно пов'язаних розділів.

У першому розділі проведено аналіз предметної області, розглянуто існуючі інформаційні системи складського обліку, визначено їхні переваги та недоліки, особливо в контексті доступності для невеликих підприємств.

Другий розділ присвячений проектуванню системи: сформульовано функціональні та нефункціональні вимоги, визначено основний функціонал, спроектовано архітектуру системи, описано вибір технологій і побудовано моделі компонентів.

У третьому розділі наведено реалізацію системи: створено вебінтерфейс із розділами для обліку товарів, звітності, редагування складу, налаштування прав доступу та авторизації користувачів. Також описано результати візуального тестування на прикладі умовного складу, що підтверджують працездатність і зручність розробленого програмного продукту.

У висновках підсумовано основні досягнення проєкту, виокремлено переваги, а також окреслено перспективи розвитку системи.

РЕФЕРАТ

Кваліфікаційна бакалаврська робота містить 73 сторінок, 16 таблиць, 44 рисунків, список використаних джерел з 25 найменувань, 3 додатки.

«ПРОЄКТУВАННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ З ОБЛІКУ ТА МОНІТОРИНГУ МАТЕРІАЛЬНИХ ЗАПАСІВ СКЛАДУ»

Об'єкт дослідження: спроектувати автоматизовану систему обліку та моніторингу матеріальних запасів

Предмет дослідження: методи та засоби розробки автоматизованої інформаційної системи обліку та моніторингу матеріальних запасів складу.

Метою роботи є проєктування та розробка автоматизованої інформаційної системи для обліку та моніторингу матеріальних запасів складу, що дозволить підвищити точність облікових процесів, зменшити ризики втрат і оптимізувати управління логістикою.

Відповідно до поставленої мети визначені такі завдання: проаналізувати предметну галузь та існуючі програмні рішення для обліку матеріальних запасів; сформулювати та специфікувати функціональні й нефункціональні вимоги до системи; розробити архітектуру інформаційної системи, включаючи моделі її поведінки та структури; спроектувати інформаційне, технічне та програмне забезпечення; реалізувати ключові компоненти системи: вебінтерфейс, модулі обліку, звітності, сповіщення та контролю залишків; провести тестування розробленої системи на імітаційному прикладі.

Теоретична значущість полягає в узагальненні підходів до створення автоматизованих інформаційних систем для управління складськими ресурсами в умовах цифровізації бізнес-процесів.

Методична цінність роботи: полягає в розробці комплексної методики створення подібних систем, що охоплює аналіз вимог, проєктування архітектури, вибір технологічного стеку та реалізацію програмних модулів з можливістю масштабування та інтеграції.

Практична значущість: полягає у створенні функціонального прототипу інформаційної системи, який є доступним, гнучким і придатним до використання в реальних умовах підприємства. Система дозволяє автоматизувати облік, контролювати терміни зберігання товарів, формувати звітність.

Ключові слова: автоматизована система, складський облік, автоматизація, логістика, React, Firebase, моніторинг запасів, хмарні технології.

Рік виконання кваліфікаційної бакалаврської роботи – 2025.

Рік захисту роботи – 2025.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧКИ	10
ВСТУП	11
РОЗДІЛ 1	14
ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ.....	14
1.1 Характеристика предметної галузі та об'єкта дослідження	14
1.2 Аналіз літературних джерел та практичного досвіду використання ІС і технологій в предметній галузі	16
РОЗДІЛ 2	20
РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ	20
2.1 Аналіз і специфікація вимог до інформаційної системи	20
2.1.1 Бізнес-вимоги	20
2.1.2 Функціональні вимоги	22
2.3 Нефункціональні вимоги до системи	23
2.2 Постановка задачі	25
2.2.1 Характеристика задачі	25
2.2.2 Вихідна інформація	27
2.2.3 Вхідна інформація	28
2.2.4 Алгоритм розв'язання задачі	29
2.3 Моделювання інформаційної системи	32
2.3.1 Моделювання поведінки системи	32
2.3.2 Модель структури системи	34
2.3.3 Розподіл вимог за компонентами системи	35
РОЗДІЛ 3	39
ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ СИСТЕМИ	39
3.1 Інформаційне забезпечення.	39
3.1.1 Загальна характеристика інформаційного забезпечення	39
3.1.2 Організація збору і передання первинної інформації	40
3.1.3 Структура інформаційних масивів.....	41
3.1.4 Вибір СКБД	50
3.1.5 Інформаційна та деталогічна моделі бази даних	51

3.2 Технічне забезпечення	53
3.3 Програмне забезпечення	54
3.4 Результати реалізації автоматизованої системи.....	56
ВИСНОВКИ	63
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	65
ДОДАТКИ	68
ДОДАТОК А	68
ДОДАТОК Б.....	72
ДОДАТОК В.....	75

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧКИ

АС	Автоматизована система
БД	База даних
UI	Інтерфейс користувача (User Interface)
API	Application programming interface
СКБД	Система керування базою даних
FIFO	First In, First Out
LIFO	Last In, First Out
AVCO	Average Cost
ПЗ	Програмне забезпечення
ІС	Інформаційна система

ВСТУП

У сучасному діловому середовищі, яке характеризується швидким розвитком цифрових технологій, нестабільною економічною ситуацією та високим рівнем конкуренції, ефективне управління матеріальними запасами набуває стратегічного значення для будь-якого підприємства. Саме складські ресурси відіграють ключову роль у забезпеченні безперервності бізнес-процесів, формуванні фінансової стійкості компанії, а також у підвищенні рівня клієнтського сервісу. У таких умовах автоматизація процесів обліку та моніторингу запасів стає не просто корисною інновацією, а необхідною складовою успішної діяльності організації.

Насамперед автоматизовані інформаційні системи дозволяють значно зменшити вплив людського фактора на точність даних, знизити витрати, пов'язані з ручним обліком, та підвищити прозорість усіх логістичних операцій. Особливо це актуально для українських підприємств, які перебувають у стані адаптації до нових ринкових умов, воєнних ризиків та необхідності інтеграції до цифрової економіки. Для багатьох компаній — особливо малого та середнього бізнесу — впровадження простих у використанні, недорогих і ефективних систем є чи не єдиною можливістю оптимізувати внутрішні процеси без залучення великих інвестицій.

Аналізуючи наукові джерела та практичні розробки, можна помітити, що фокус сучасних досліджень переважно зміщується на великі корпорації. Автори, як-от О.І. Тарасенко[1], досліджують методи оптимізації логістичних процесів за допомогою складних ІТ-рішень, а М.Ю. Верес[2] акцентують увагу на впровадженні потужних ERP-систем у логістичні модулі підприємств. У міжнародному контексті активно розглядаються RFID-технології (наприклад, у роботах Дж. Саймона)[4], які дозволяють здійснювати автоматичний контроль руху товарів. Проте такі рішення здебільшого орієнтовані на великі компанії з відповідними фінансовими та технічними ресурсами.

У контексті малого та середнього бізнесу ситуація значно складніша. Більшість доступних рішень або надто дорогі, або надмірно складні для інтеграції та щоденного використання. З іншого боку, на ринку присутні більш

доступні альтернативи. Система Odoo Inventory пропонує модульну структуру та можливість поступового розширення функціоналу, проте може викликати складнощі з налаштуванням для специфічних потреб українського бізнесу. 1С:Підприємство залишається популярною серед вітчизняних компаній завдяки локалізації та підтримці українського законодавства, однак часто потребує додаткових модулів та налаштувань для повноцінної логістичної функціональності. Zoho Inventory виділяється своїм зручним інтерфейсом та хмарною архітектурою, але орієнтований переважно на західний ринок і має обмеження щодо інтеграції з локальними системами. Ці рішення, незважаючи на свої переваги, все ще не повністю відповідають специфіці українського бізнес-середовища та потребам малого і середнього підприємництва.

Це створює чітку нішу для розробки власних програмних продуктів — гнучких, адаптивних, масштабованих, локалізованих і таких, що відповідають конкретним потребам користувача. Саме таку задачу було поставлено — створити інформаційну систему для обліку матеріальних запасів складу, що ґрунтується на сучасних веб-технологіях, має інтуїтивно зрозумілий інтерфейс і не потребує значних ресурсів для запуску.

Мета моєї дослідницької роботи полягає у проєктуванні та реалізації автоматизованої інформаційної системи обліку та моніторингу складських запасів з використанням React як інструменту для побудови клієнтської частини та Firebase як хмарної бази даних і бекенд-платформи. Такий стек технологій дозволяє досягти високої швидкості розробки, гнучкості, а також зручної масштабованості рішення.

Для реалізації цієї мети були поставлені наступні завдання: провести аналіз предметної області та існуючих рішень, визначити функціональні й нефункціональні вимоги до системи, розробити архітектуру ПЗ, спроектувати та реалізувати інтерфейс користувача, виконати тестування і дати оцінку результатам. У дослідженні застосовано методи системного аналізу, об'єктно-орієнтованого проєктування (з використанням UML-діаграм), моделювання бізнес-процесів та практичного експериментального підходу до перевірки гіпотез.

Результатом роботи стало створення прототипу інформаційної системи, адаптованої під специфіку невеликих підприємств, із можливістю подальшого

розширення. Основними функціональними компонентами системи стали: облік залишків на складі, контроль термінів придатності, формування сповіщень про низький залишок та звітність у зручному вигляді. Інтерфейс був реалізований у вигляді адаптивного веб-застосунку, що дозволяє працювати з системою як з комп'ютера, так і з мобільного пристрою.

Теоретична значущість дослідження полягає у систематизації сучасних підходів до розробки інформаційних систем обліку складських ресурсів з акцентом на хмарні технології. Методична цінність полягає у використанні сучасного програмного стеку, що дозволяє швидко створювати ефективні веб-додатки. Практична цінність — у розробці реального програмного продукту, який може бути впроваджений у діяльність малого чи середнього підприємства без значних затрат.

Апробація результатів дослідження відбулася під час проходження переддипломної практики в компанії ТОВ «Провайд-Груп», де розроблена система була протестована в умовах реального бізнес-середовища. Це дозволило оцінити її зручність, ефективність і адаптивність до реальних потреб користувачів.

Таким чином, автоматизація обліку складських запасів є не лише актуальним напрямом розвитку ІТ-технологій у логістиці, а й реальним інструментом підвищення ефективності малого і середнього бізнесу в Україні. Моя робота демонструє, що завдяки правильному вибору технологій і глибокому аналізу потреб ринку можна створити рішення, яке не поступається за якістю комерційним аналогам, але перевершує їх за доступністю, адаптивністю та відповідністю локальним умовам.

РОЗДІЛ 1

ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Характеристика предметної галузі та об'єкта дослідження

Сучасна економіка працює в умовах швидких змін та високої конкуренції. Це змушує компанії ефективно управляти своїми матеріальними ресурсами. Особливо важливими є логістика та складське господарство, які забезпечують постачання, зберігання та облік товарів. Саме в цій сфері потрібна автоматизація обліку та контролю, що є головною темою нашого дослідження. Ми розробляємо веб-систему, яка дозволяє вести облік, моніторинг та контроль термінів придатності товарів у реальному часі.

Наша предметна галузь охоплює складську логістику — це процеси прийому, зберігання, видачі та контролю матеріальних ресурсів. В інформаційних технологіях ця сфера характеризується великою кількістю товарів для обліку, включаючи продукцію з обмеженим терміном зберігання. Для таких товарів особливо важливо автоматично відстежувати терміни придатності, щоб уникнути втрат та знизити ризики. З цією метою в системі реалізовано принцип послідовного вибуття товарів — у порядку їх надходження на склад, що дозволяє ефективно управляти запасами та уникати накопичення простроченої продукції.

До об'єктів нашої галузі належать склади різного призначення — від централізованих логістичних баз до складських приміщень малих підприємств, де відбувається зберігання та реалізація товарів. Усі ці об'єкти потребують ефективної системи обліку, контролю залишків і строків придатності для забезпечення безперебійної діяльності та мінімізації втрат.

У цій сфері приймаються важливі управлінські рішення: визначення потреб у матеріалах, розподіл ресурсів між відділами, виявлення нестач або надлишків, формування замовлень та контроль термінів придатності продукції, облік переміщення ресурсів. Головні особи, що приймають рішення

— це керівники складів, комірники, бухгалтери, матеріально відповідальні особи та керівництво підприємства. При колективному прийнятті рішень використовуються узгоджені підходи з чітким розподілом обов'язків та відповідальності.

На прийняття рішень впливають внутрішні фактори (організаційна структура, технічне обладнання складу, кваліфікація працівників) та зовнішні фактори (ринкова ситуація, постачальники, транспортна логістика, законодавчі вимоги). Ступінь цього впливу може бути різним — від незначного до критичного, особливо в кризових або нестандартних ситуаціях.

Рішення в нашій галузі тісно пов'язані між собою: облік наявних залишків впливає на рішення про нові замовлення; звітність стає основою для фінансового планування; зміни в графіку постачання змінюють плани зберігання та відвантаження. Таким чином, кожне рішення впливає як на поточну ситуацію, так і на майбутні дії.

Ситуації прийняття рішень у цій сфері можуть бути як передбачуваними (за відомих умов), так і непередбачуваними — в умовах ризику, невизначеності або навіть у кризових обставинах, наприклад, при збоях постачання. Різні ситуації вимагають різних підходів: від чіткого планування до гнучкого управління та моделювання ситуацій.

Отже, предметна галузь, пов'язана з обліком та моніторингом матеріальних запасів складу, має складну структуру та динамічне середовище прийняття рішень. Це вимагає впровадження автоматизованих інформаційних систем, які дозволяють швидко обробляти дані, приймати ефективні рішення та забезпечувати контроль на всіх етапах руху матеріальних ресурсів. Наша система призначена не лише для обліку, а й для активного контролю стану складських запасів, що дозволяє уникати прострочених товарів та забезпечувати безперервність логістичних операцій.

1.2 Аналіз літературних джерел та практичного досвіду використання ІС і технологій в предметній галузі

У сфері складської логістики автоматизовані інформаційні системи (АС) стали ключовим інструментом управління матеріальними потоками. Літературні джерела [1–4] свідчать про стрімке впровадження цифрових технологій в облік, моніторинг та планування запасів. Сучасні АС дозволяють підприємствам ефективно відслідковувати переміщення товарів, автоматизувати документообіг, контролювати залишки та запобігати псуванню продукції.

У літературі виділяють кілька основних методів обліку товарно-матеріальних запасів:

FIFO (First In, First Out) — Ця схема передбачає, що товари, закуплені першими, будуть першими продані клієнтам. Акцент робиться на облік часу, коли вони були придбані, а не коли вони були продані або використані з іншою метою.[22]

LIFO (Last In, First Out) — Ця схема працює аналогічно FIFO, але натомість передбачає, що непродані запаси, що були закуплені останніми, будуть продані першими. Тому під час обліку важливо звертати увагу на те, коли вони були придбані востаннє.[22]

AVCO (Average Cost) — З використанням цього методу контролю запасів вартість кожного товару визначається з урахуванням його середньої ціни протягом певного проміжку часу. Це дає змогу керівникам магазинів точно відслідковувати витрати на закупівлю, навіть якщо ціни коливаються.[22]

У контексті обліку АС — метод FIFO є найбільш доцільним. Він дозволяє автоматично контролювати черговість видачі товарів, зменшувати ризики утилізації прострочених запасів та забезпечує логічну відповідність реальному руху товарів на складі. Також цей метод добре підтримується більшістю сучасних складських програм.

У практиці впровадження автоматизованих систем можна виділити такі популярні рішення:

1. 1С:Підприємство: Один із найвідоміших продуктів на пострадянському просторі. Забезпечує комплексний облік товарів, партій, серій, рухів по складу, а також інтеграцію з бухгалтерією. Однак система орієнтована на російський ринок і не підтримує українську мову «з коробки», що є критичним недоліком для українських підприємств. Також виникають проблеми з ліцензуванням через санкції.[23]

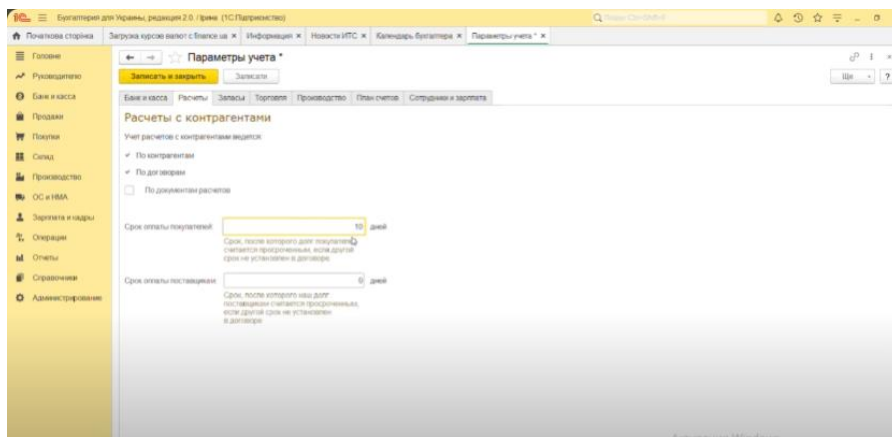


Рисунок 1.2 – 1С:Підприємство

Джерело: скрін з відео <https://www.youtube.com/watch?v=NcsmP-IkFw>

1 . Odoo Inventory: Модульна ERP-система з відкритим кодом. Дає змогу налаштувати облік за методами FIFO, FEFO, LIFO, відстежувати партії, терміни придатності, працювати з мобільними сканерами. Має українську локалізацію, активну спільноту та гнучкість адаптації під конкретні бізнес-процеси. Підходить для малого і середнього бізнесу. [24]

Product	Location	On Hand	Forecast	Route	Min	Max	To Order
[E-COM06] Corner Desk ...	WH/Stock	4.00	-1.00	Buy	0.00	0.00	1.00
[E-COM09] Large Desk	WH/Stock	1.00	-4.00	Buy	0.00	0.00	4.00
[FURN_9001] Flipover	WH/Stock	5.00	-6.00	Buy	0.00	0.00	6.00
[FURN_9666] Table	WH/Stock	2.00	-1.00	Buy	0.00	0.00	1.00
[FURN_7777] Office Chair	WH/Stock/Asse...	4.00	4.00	Buy	5.00	10.00	6.00
[FURN_8888] Office Lamp	WH/Stock/Asse...	8.00	8.00	Buy	10.00	10.00	2.00
[FURN_8900] Drawer Black	WH/Stock/Asse...	12.00	12.00	Manufacture	0.00	0.00	0.00
[FURN_9001] Flipover	WH/Stock/Asse...	5.00	5.00	Manufacture	0.00	0.00	0.00
[E-COM06] Corner Desk ...	WH/Stock/Flat P...	4.00	4.00	Manufacture	0.00	0.00	0.00
[E-COM09] Large Desk	WH/Stock/Flat P...	1.00	1.00	Manufacture	2.00	2.00	1.00
[FURN_9666] Table	WH/Stock/Flat P...	2.00	2.00	Manufacture	5.00	10.00	8.00

Рисунок 1.2 – Odoo Inventory

Джерело: <https://www.odoo.com/>

3. Zoho Inventory: Хмарна система управління запасами з простим інтерфейсом та широкими можливостями для малого та середнього бізнесу. Дозволяє контролювати запаси в кількох місцях, автоматизувати облік, створювати звіти. Має англійський інтерфейс, але відсутня офіційна підтримка української мови. Основна перевага — швидкість впровадження і мінімальні технічні вимоги.[25]

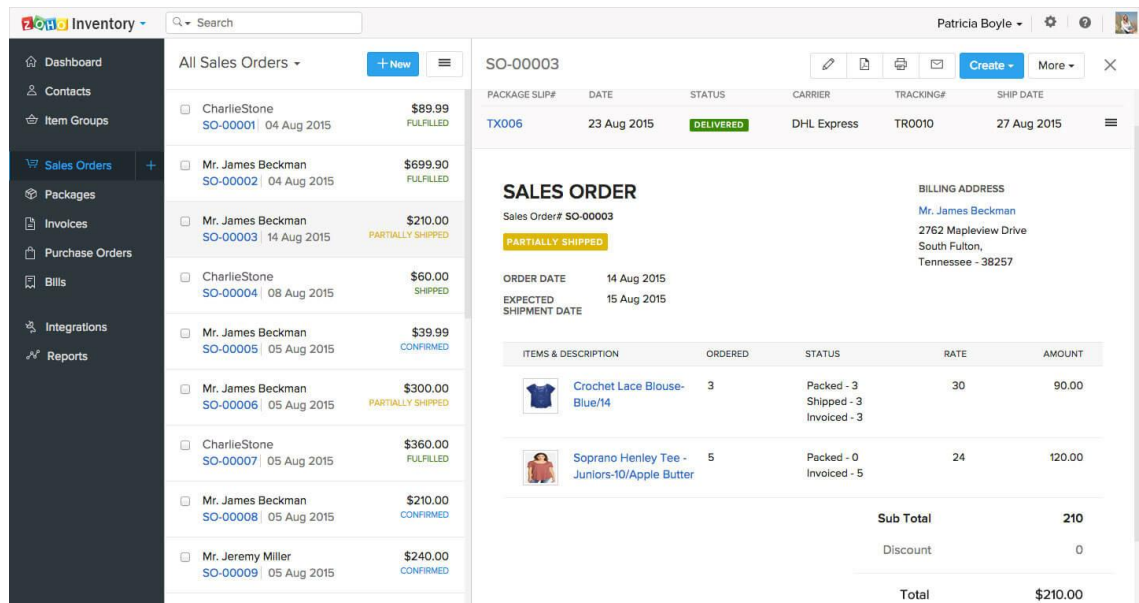


Рисунок 1.2 – Zoho Inventory

Джерело: <https://www.zoho.com/blog/inventory/introducing-zoho-inventory.html>

Таблиця 1.1 – Формуляр ІС

№	Критерій	1С:Підприємство	Odoo Inventory	Zoho Inventory
1	Облік товарів, партій, серій	так	так	так
2	Підтримка методів FIFO, FEFO, LIFO	ні	так	так
3	Інтеграція з бухгалтерією	так	так	ні
4	Наявність української локалізації	ні	неповністю	ні
5	Підтримка спільноти або офіційна підтримка	ні	так	так

6	Можливість адаптації під бізнес-процеси	так	так	ні
7	Підходить для малого та середнього бізнесу	ні	так	так
8	Хмарне рішення	ні	так	так
9	Швидкість впровадження	ні	ні	так
10	Низькі технічні вимоги	ні	ні	так
11	Проблеми з ліцензуванням через санкції	так	ні	ні

Джерело сформовано автором на основі виконаного дослідження

На основі порівняльного аналізу трьох систем управління складськими запасами всі рішення мають критичні недоліки для українського ринку. 1С:Підприємство демонструє найгірші результати (3 з 11 критеріїв) через відсутність підтримки сучасних складських методів FIFO/FEFO, серйозні проблеми з ліцензуванням через санкції, відсутність української локалізації та офіційної підтримки, неможливість хмарного розгортання та складність впровадження. Odoo Inventory, маючи позитивні оцінки за 7 критеріїв, страждає від неповної української локалізації, складності швидкого впровадження та високих технічних вимог для підтримки. Zoho Inventory, показуючи найкращі результати для швидкого старту (5 критеріїв), має повну відсутність української локалізації, відсутність інтеграції з бухгалтерією та неможливість адаптації під специфічні бізнес-процеси. Таким чином, жодна з розглянутих систем не забезпечує повноцінного рішення для українського ринку без суттєвих компромісів або додаткових інвестицій у доопрацювання.

РОЗДІЛ 2

РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1 Аналіз і специфікація вимог до інформаційної системи

2.1.1 Бізнес-вимоги

Інформаційна система замовлення залізничних квитків є складним комплексом, успішне функціонування якого залежить від задоволення бізнес-цілей усіх зацікавлених сторін.

Внутрішні стейкхолдери, зокрема розробники, зацікавлені у створенні якісної, надійної та масштабованої автоматизованої системи обліку й моніторингу матеріальних запасів складу. Вони прагнуть забезпечити стабільну роботу, можливість інтеграції з іншими системами, технічну підтримку та регулярні оновлення.

Керівництво компанії розглядає впровадження такої системи як засіб підвищення прибутковості, оптимізації внутрішніх процесів, зниження витрат і зміцнення конкурентних позицій підприємства на ринку.

Зовнішні стейкхолдери – це користувачі системи (менеджери, комірники, бухгалтери тощо), які очікують зручного, зрозумілого та ефективного інтерфейсу для обліку та контролю залишків. Вони хочуть автоматизації рутинних операцій, швидкого доступу до даних і мінімізації помилок.[5]

Для забезпечення гармонійної взаємодії та ефективної співпраці з усіма стейкхолдерами, до автоматизованої системи обліку та моніторингу матеріальних запасів складу висувається низка ключових бізнес-вимог.

Бізнес-вимоги – це специфікації, які окреслюють те, що потрібно бізнесу від інформаційної системи чи рішення. Вони часто містять опис функцій і функцій, а також будь-які технічні обмеження [6].

Далі наведено діаграму бізнес-вимог і бізнес-цілей кожного учасника, що взаємодіє з системою (див. рис. 2.1). Основними учасниками є власник/керівник підприємства, розробник (адміністратори), користувачі системи

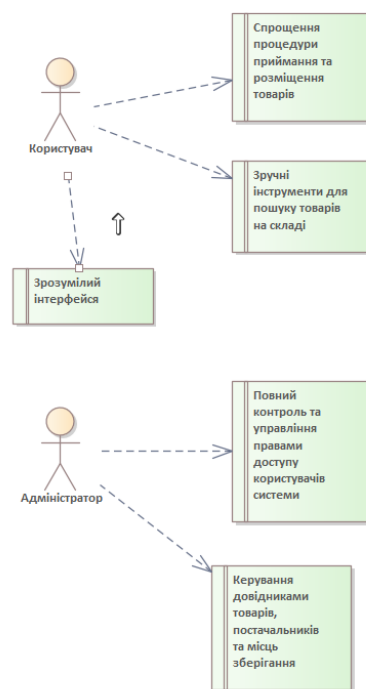


Рисунок 2.1 – Діаграма бізнес-вимог

Джерело сформовано автором на основі виконаного дослідження

2.1.2 Функціональні вимоги

Функціональні вимоги – це вимоги до програмного забезпечення, які описують внутрішню роботу системи, її поведінку: обчислення даних, маніпулювання даними, обробка даних та інші специфічні функції, які має виконувати система [7].

Функціональні вимоги до автоматизованої системи обліку та моніторингу матеріальних запасів складу визначають дії, які система має виконувати для задоволення потреб користувачів та адміністраторів.

Ключовими підсистемами є:

- Облік товарів – передбачає реєстрацію надходжень і відвантажень продукції, ведення обліку терміну придатності, інвентаризації та повного контролю за залишками на складі.
- Звітність – забезпечує формування звітів у реальному часі для аналізу запасів, руху товарів та загального стану складських ресурсів.
- Керування користувачами – включає автентифікацію, розподіл ролей та прав доступу між адміністраторами і користувачами системи.
- Сповіщення – автоматично інформує відповідальних осіб про критичні залишки або наближення термінів придатності продукції.
- Адміністрування – дозволяє адміністраторам керувати системними налаштуваннями, оновлювати дані про товарні позиції, формувати шаблони звітів, контролювати доступ і забезпечувати безперебійну роботу системи.

Ці підсистеми забезпечують зручну, безпечну й ефективну роботу всієї системи, враховуючи вимоги реальних складських процесів та потреби різних категорій користувачів.

Детальніше переглянути функціональні вимоги можна на діаграмі, що зображена на рис. 2.2, а також у специфікації вимог, сформованих за допомогою менеджера специфікацій (Specification Manager) Enterprise Architect[14], що наведено у «Додатку Б».

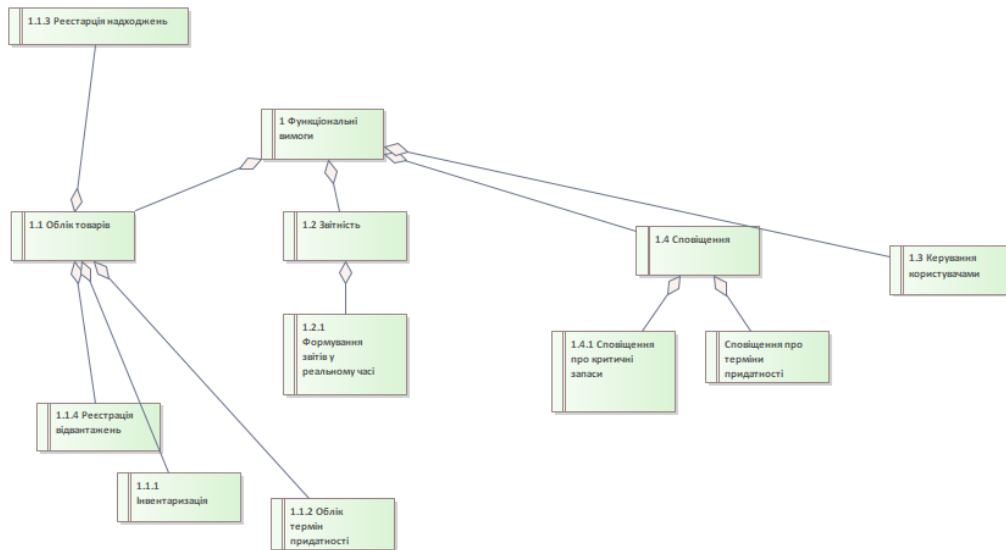


Рисунок 2.2 – Діаграма функціональних вимог

Джерело сформовано автором на основі виконаного дослідження

2.3 Нефункціональні вимоги до системи

Нефункціональні вимоги – це вимоги до програмного забезпечення, які задають критерії для оцінки якості його роботи. На відміну від функціональних вимог, які визначають що система повинна робити, нефункціональні вимоги визначають якою система повинна бути. [8]. У рамках автоматизованої системи обліку та моніторингу матеріальних запасів складу, нефункціональні вимоги є не менш важливими за функціональні, оскільки визначають якість роботи системи, комфорт користування та рівень довіри з боку користувачів і адміністраторів.

Зручність використання полягає в наданні інтуїтивно зрозумілого інтерфейсу, що дозволяє працівникам складу легко орієнтуватися у функціоналі системи без тривалого навчання. Інтерфейс має бути адаптивним до різних роздільних здатностей екранів а також передбачати зручні елементи навігації та пошуку.

Безпека системи передбачає захист даних про товарні залишки, історію операцій і доступ користувачів. Необхідним є шифрування паролів у базі

даних, розмежування прав доступу між користувачами та адміністраторами, а також захист від несанкціонованого втручання у внутрішні процеси системи.

Надійність реалізується через забезпечення безперебійної роботи системи, збереження даних у разі збоїв, резервне копіювання та автоматичне відновлення. Це критично важливо для уникнення втрати інформації про залишки або переміщення товарів.

Продуктивність визначає швидкість обробки запитів, швидкодію при реєстрації надходжень чи відвантажень, а також оперативність формування звітів і проведення інвентаризації.

Масштабованість передбачає здатність системи підтримувати зростання обсягів товарів, користувачів та запитів без погіршення продуктивності. Гнучка архітектура має забезпечити можливість інтеграції з іншими програмними рішеннями (наприклад, ERP-системами чи обліковими сервісами).

Загальні нефункціональні вимоги системи наведено на рис. 2.3, а також у вигляді специфікації в «Додатку Б», що були сформовані за допомогою інструмента Specification Manager у середовищі Enterprise Architect.

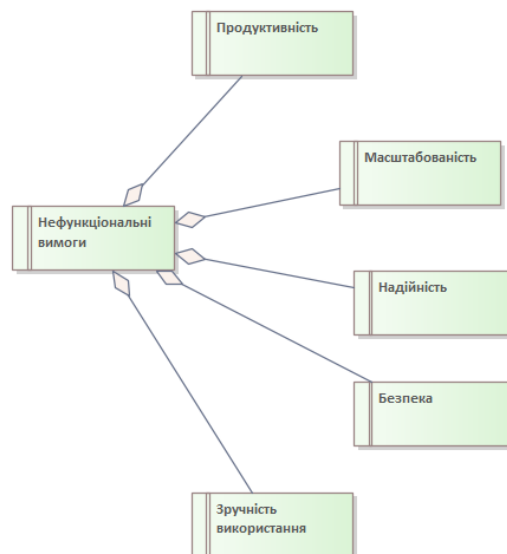


Рисунок 2.3 – Діаграма нефункціональних вимог

Джерело сформовано автором на основі виконаного дослідження

2.2 Постановка задачі

2.2.1 Характеристика задачі

Призначенням задачі є створення зручної автоматизованої інформаційної системи, за допомогою якої користувачі зможуть здійснювати авторизацію для доступу до облікових функцій, а також отримувати актуальну інформацію про наявність та рух матеріальних запасів на складі. Адміністратори, мають розширені права доступу — вони можуть керувати всіма об'єктами системи: додавати та редагувати користувачів, додавати редагувати нові склади, налаштовувати параметри обліку, редагувати інформацію по товарам. Таким чином, адміністратор відповідає за загальне управління системою та контроль її функціонування.

Вихідна інформація використовується для відображення результатів запитів користувачів (наприклад, перегляду залишків, руху матеріалів), формування аналітичних звітів та забезпечення ефективного управління запасами. Дані надаються автоматично та оперативно за запитом у режимі реального часу, що дозволяє швидко приймати управлінські рішення та забезпечувати безперервність складських операцій.

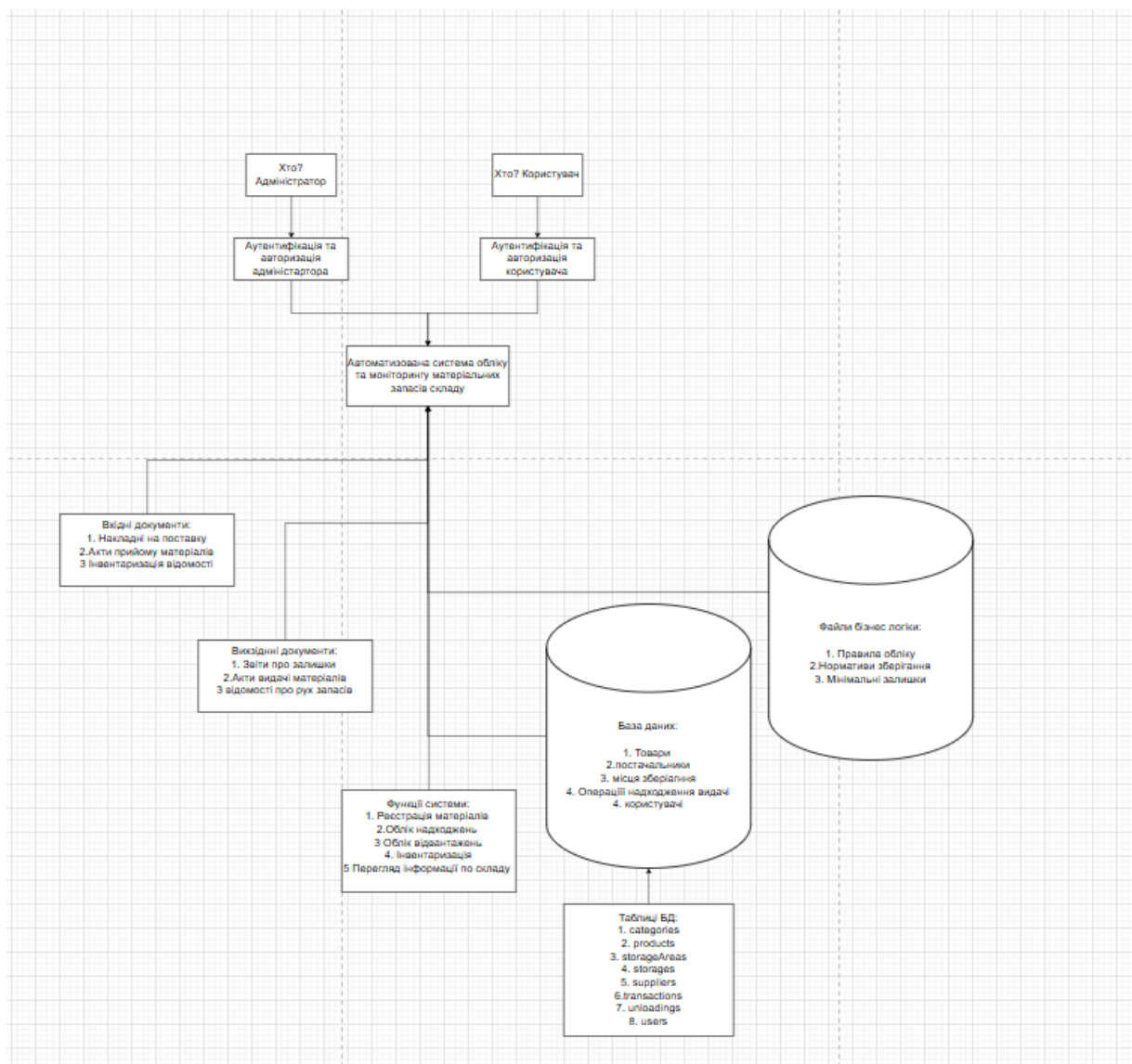


Рисунок 2.3 – Інформаційна модель АС

Джерело сформовано автором на основі виконаного дослідження

У рамках підрозділу також було створено інформаційну модель інформаційної системи (див. рис. 2.3). Вона відображає взаємодію користувачів і адміністраторів із системою обліку та моніторингу матеріальних запасів складу через форми введення даних, які пов'язані з відповідними таблицями бази даних: товарів, постачальників, місць зберігання, операцій із надходження та видачі, а також облікових записів користувачів. Вся інформація обробляється централізовано в рамках інформаційної системи.

2.2.2 Вихідна інформація

Вихідна інформація формується системою в результаті виконання облікових операцій та взаємодії з базою даних. Вона використовується для контролю наявності матеріалів, аналізу руху запасів, моніторингу залишків і прийняття управлінських рішень. Формати вихідних документів передбачають як екранне відображення

Інформація надається в реальному часі або за запитом користувача. Основними користувачами вихідної інформації є адміністратори системи, складські працівники, відділ закупівель, керівництво та бухгалтерія.

Перелік та опис вихідних повідомлень наведено у таблиці 2.1

Таблиця 2.1 – Опис Вихідних Повідомлень

№ з/п	Назва вихідного повідомлення	Ідентифікатор	Форма подання і вимоги до неї	Періодичність видання	Термін видання і допустимий час затримки	Користувачі інформації
1	Звіт про залишки на складі	OUT-001	Екранна форма	Щодня / За запитом	Миттєво, затримка – до 5 секунд	Адміністратор, користувач
2	Акт прийому матеріалів	OUT -002	Екранна форма	Після кожного прийому	Миттєво, затримка – до 5 секунд	Адміністратор, користувач
3	Звіт про рух матеріалів	OUT -003	Екранна форма	За запитом	Миттєво, без затримки	Адміністратор, користувач
4	Інвентаризаційна відомість	OUT -004	Екранна форма	Щомісяця / За розпорядженням	Миттєво, без затримки	Адміністратор, користувач
5	Звіт про історію змін і операцій у системі	OUT -005	Екранна форма	За запитом	Миттєво, без затримки	Адміністратор, користувач

Джерело сформовано автором на основі виконаного дослідження

2.2.3 Вхідна інформація

Вхідна інформація є основою функціонування системи та забезпечує актуальність даних, необхідних для обліку та контролю матеріальних запасів. До неї належать відомості про нові надходження товарів, списання, зміни в номенклатурі, дані від постачальників, інформація про переміщення матеріалів між складами, а також дії користувачів у системі.

Інформація вводиться через інтерфейси користувача вручну або імпортується з зовнішніх джерел, зокрема з облікових документів, скан-копій, таблиць або інтегрованих API. Основними джерелами вхідної інформації виступають складські працівники, постачальники, системні адміністратори та зовнішні електронні джерела.

Таблиця 2.2 – Перелік і опис вхідних повідомлень

№ з/п	Назва вхідного повідомлення	Ідентифікатор	Форма подання	Термін і частота надходження	Джерело
1	Дані про надходження матеріалів	IN-001	Електронна форма введення / імпорт з Excel	Після кожного постачання	Адміністратор, користувач
2	Інформація про видачу матеріалів	IN-002	Форма введення в системі	Після кожної видачі	Адміністратор, користувач
3	Зміни у номенклатурі товарів	IN-003	Адмін-форма	За потреби	Адміністратор
4	Дані про користувача	IN-004	Форма реєстрації / редагування	При створенні або зміні запису	Адміністратор
5	Дані про переміщення між складами	IN-005	Електронна форма введення	Після кожного переміщення	Адміністратор, користувач
6	Результати інвентаризації	IN-006			

Джерело сформовано автором на основі виконаного дослідження

2.2.4 Алгоритм розв'язання задачі

Алгоритм розв'язання задачі обліку та моніторингу матеріальних запасів складу ґрунтується на обробці поточних і архівних даних, що надходять у систему від користувачів, а також автоматично з форм введення у веб-інтерфейсі. Основна мета використаної інформації — забезпечення актуального стану обліку товарів, контролю їх залишків, термінів придатності та виявлення критичних ситуацій (низький рівень запасів, закінчення терміну зберігання тощо).

Для функціонування інформаційної системи розроблено структуру даних, представлену у вигляді масивів, що містять ключову інформацію про складські операції.

2.3 — Перелік масивів використовуваної інформації

№	Масив	Ідентифікатор
1	Поточні надходження	M_INCOMING
2	Витрати товарів	M_OUTGOING
3	Залишки на складі	M_REMAINS
4	Історія операцій	M_HISTORY
5	Сповіщення та попередження	M_ALERTS

Джерело сформовано автором на основі виконаного дослідження

Алгоритм обробки даних виконує аналіз надходжень та витрат, обчислює актуальні залишки, порівнює їх із критичними значеннями, а також перевіряє строки придатності товарів. У разі виявлення порушення встановлених меж система формує відповідні повідомлення для користувача і рекомендації до дій (наприклад, дозамовлення продукції, списання товару).

2.4 — Перелік масивів використовуваної інформації

№	Масив	Ідентифікатор
1	Звіт про складські залишки	R_REMAINS
2	Рекомендації щодо дій	R_RECOMMEND
3	Графіки обігу товарів	R_GRAPHS
4	Повідомлення про критичні стани	R_ALERTS

Джерело сформовано автором на основі виконаного дослідження

Математична модель оцінки критичності залишків ґрунтується на порівнянні поточного рівня товару з визначеним пороговим значенням, яке задається вручну або визначається автоматично на основі середнього споживання:

$$f = \frac{Q_{\text{залишок}}}{Q_{\text{середньодобове}} \times T}$$

Де:

$Q_{\text{залишок}}$ – поточна кількість товару;

$Q_{\text{середньодобове}}$ – середньодобове споживання товару (розраховується за історією витрат);

T — критичний поріг у днях (за замовчуванням 7 днів).

Якщо значення $f < 1$, система генерує сповіщення про необхідність дозамовлення товару.

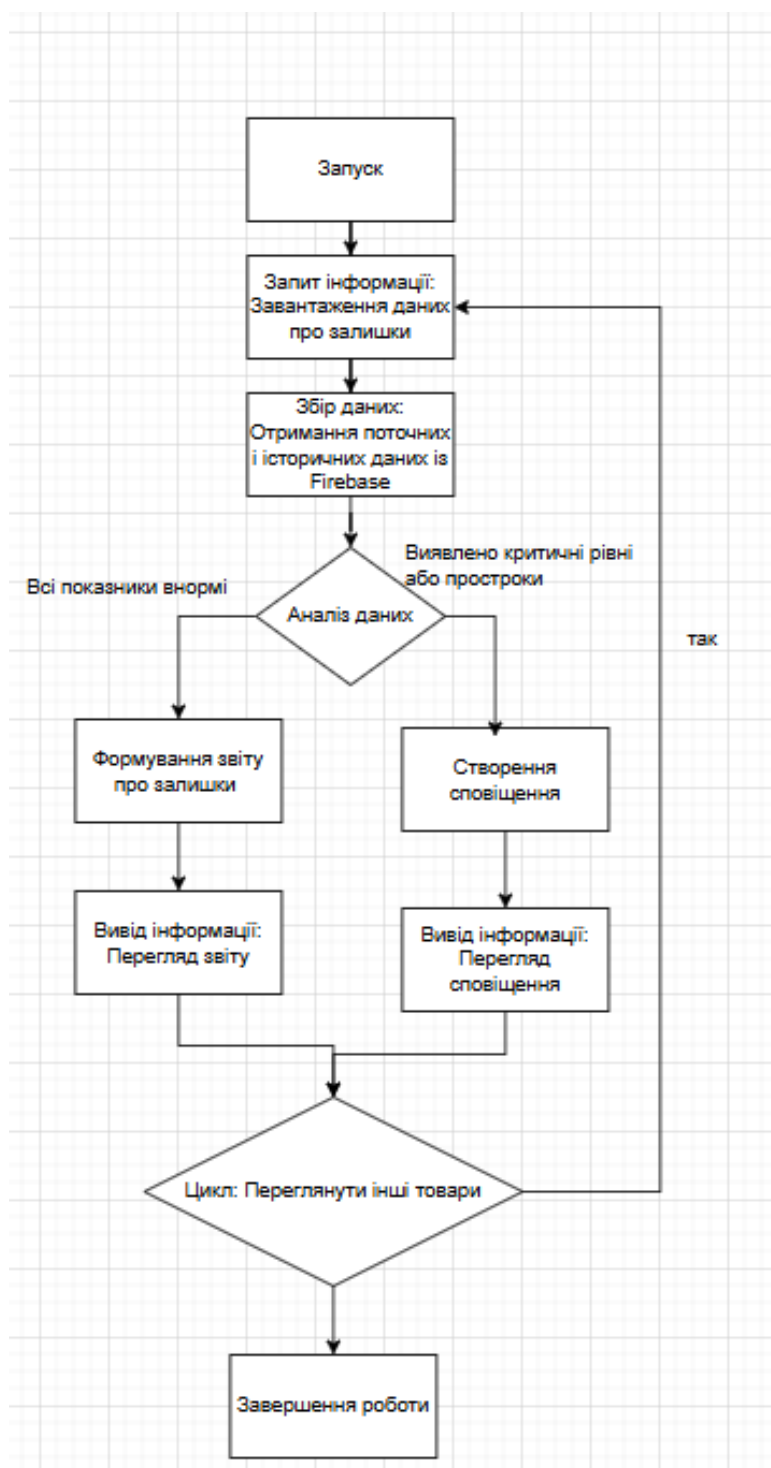


Рисунок 2.4 – Діаграма станів взаємодії користувача з автоматизованою системою

Джерело сформовано автором на основі виконаного дослідження

Діаграма станів моделює типовий сценарій взаємодії користувача з автоматизованою системою (див. рис. 2.4):

2.3 Моделювання інформаційної системи

2.3.1 Моделювання поведінки системи

Діаграма Прецедентів візуально зображає різноманітні сценарії взаємодії між акторами (користувачами) і прецедентами (випадками використання) та описує функціональні аспекти системи (бізнес логіку). [9].

Діаграма прецедентів демонструє архітектуру системи управління товарами з двома основними акторами: користувачем та адміністратором системи. Діаграма відображає ієрархічну структуру функціональних можливостей, де базовий прецедент "Вхід в систему" є центральним елементом, від якого розгалужуються інші функції системи.

Користувач має доступ до широкого спектру операцій, включаючи реєстрацію надходження товарів, пошук товарів у складі, формування звітів та проведення інвентаризації. Кожен з цих прецедентів пов'язаний відношеннями типу "include" або "extend", що вказує на залежності між функціями. Зокрема, процес реєстрації товарів включає підпроцеси введення в систему та контролю термінів придатності, що забезпечує комплексність управління товарними запасами.

Адміністратор системи має розширені права доступу до всіх функцій, включаючи можливість корегування адміністративних даних. Структура діаграми показує, що система побудована за принципом модульності, де кожна функція може використовуватися незалежно або в комбінації з іншими процесами, забезпечуючи гнучкість та ефективність управління товарними операціями підприємства.

Взаємодія дійових осіб та функціональних можливостей системи відображена на діаграмі прецедентів (див. рис. 2.5).[21]

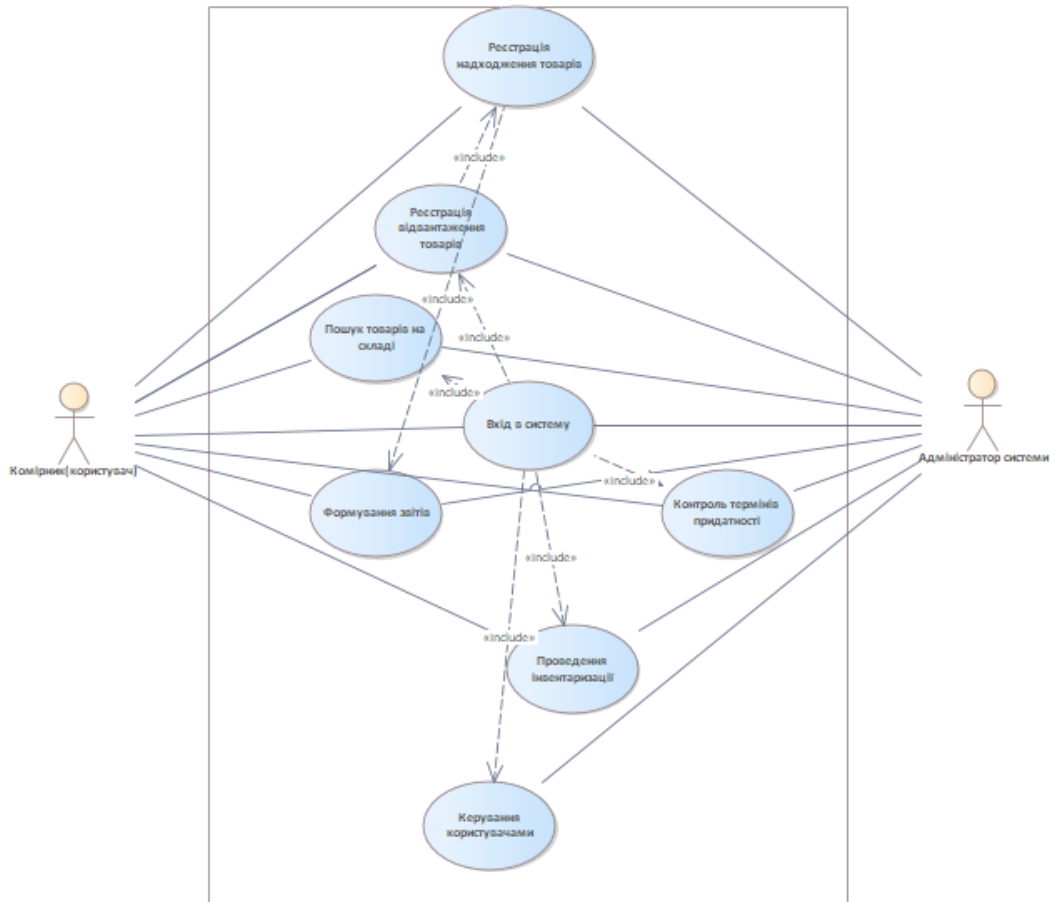


Рисунок 2.5 – Діаграма прецедентів

Джерело сформовано автором на основі виконаного дослідження

Діаграми послідовностей додаток А (A1-A8) детально ілюструють основні бізнес-процеси системи обліку та моніторингу матеріальних запасів складу. Кожна діаграма демонструє взаємодію між різними учасниками системи - від користувачів та адміністраторів до контролерів та зовнішніх сервісів, таких як Firebase Auth API. Архітектура системи побудована за принципом багатошарової моделі, де екранні форми виступають інтерфейсним шаром, контролери забезпечують бізнес-логіку, а репозиторії та API надають доступ до даних.

Діаграми охоплюють повний життєвий цикл роботи зі складськими операціями: від управління користувачами та авторизації до основних процесів реєстрації надходження та відвантаження товарів. Особлива увага приділена контролю якості через процеси моніторингу термінів придатності

та проведення інвентаризації, що забезпечує високий рівень точності обліку. Система пошуку товарів реалізована з можливістю застосування фільтрів для уточнення результатів, що підвищує ефективність роботи користувачів.

Процес формування звітів автоматизований та інтегрований у головну сторінку системи, забезпечуючи менеджерів та адміністраторів актуальною аналітичною інформацією для прийняття управлінських рішень. Всі діаграми демонструють синхронний характер взаємодій з чіткою послідовністю дій, що гарантує надійність та передбачуваність роботи системи в різних сценаріях використання.

2.3.2 Модель структури системи

Модель структури системи реалізована на основі об'єктно-орієнтованої архітектури з використанням NoSQL бази даних Firebase Firestore, що забезпечує високу гнучкість та горизонтальну масштабованість. Діаграма класів демонструє взаємозв'язки між основними сутностями системи через систему асоціацій типу "один до багатьох", де центральну роль відіграє клас Product з повним набором атрибутів для опису товарної позиції, включаючи номер партії (batchNumber) для забезпечення трасованості продукції.

Структурна організація класів відображає реальну бізнес-логіку складського господарства: Category забезпечує класифікацію товарів, Storage та StorageArea моделюють фізичну інфраструктуру складів та зон зберігання, Supplier містить інформацію про постачальників з повними контактними даними. Операційні процеси представлені класами Transaction та Unloading, які фіксують всі рухи товарів та зміни складських залишків з детальною інформацією про відповідальних осіб, причини операцій та часові мітки.

Система управління доступом реалізована через клас User з розмежуванням ролей та адміністративних прав, що забезпечує контрольований доступ до функцій системи. Використання унікальних ідентифікаторів для всіх сутностей та їх зв'язків дозволяє підтримувати

цілісність даних навіть при великих обсягах інформації. Архітектура підтримує складні аналітичні запити та формування звітності завдяки нормалізованій структурі зв'язків між класами, що критично важливо для ефективного функціонування системи складського обліку.

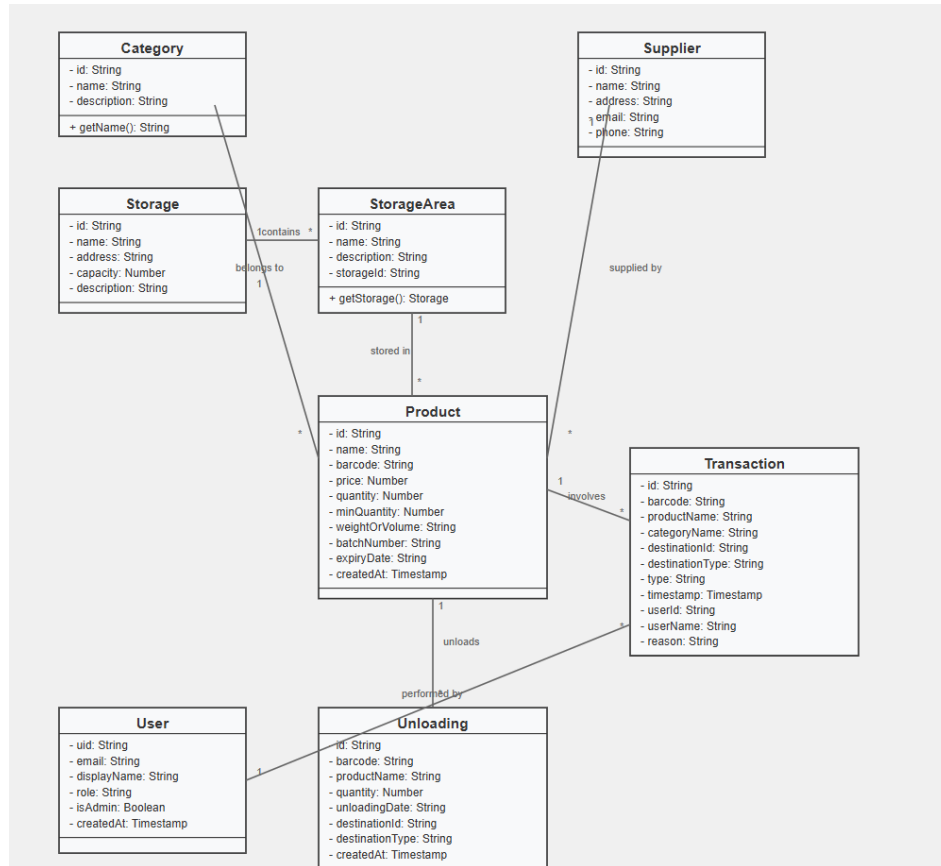


Рисунок 2.6 – Діаграма класів

Джерело сформовано автором на основі виконаного дослідження

2.3.3 Розподіл вимог за компонентами системи

Діаграму трасування було змодельовано у середовищі Enterprise Architect. На цій діаграмі представлені такі ключові компоненти системи: функціональні вимоги, прецеденти (Use Cases), тестові сценарії та класи, які реалізують відповідну логіку в системі (рис. 2.7).

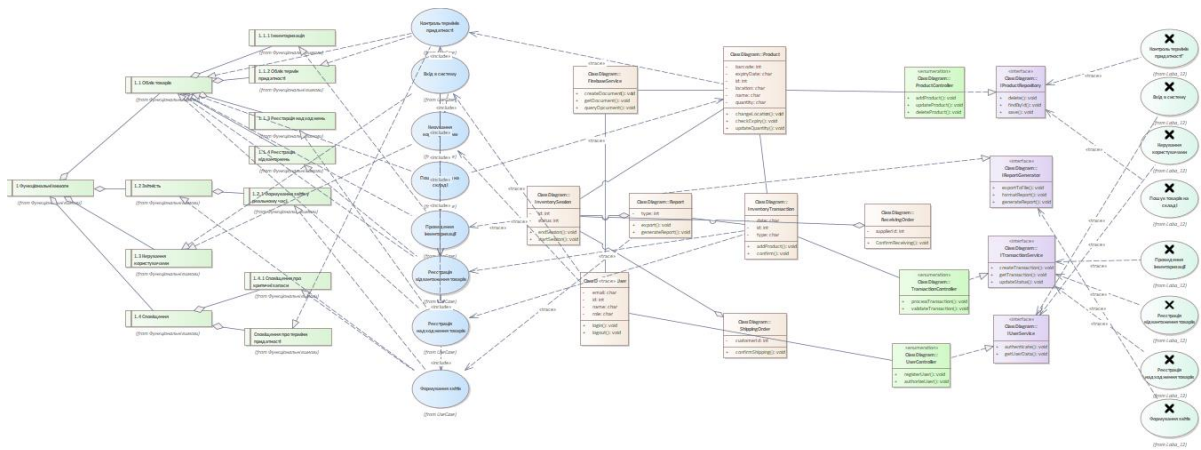


Рисунок 2.7 – Діаграма трасування

Джерело сформовано автором на основі виконаного дослідження

Зліва на діаграмі (рис. 2.7) розташовані функціональні вимоги, зокрема: керування користувачами, облік термінів придатності, пошук та облік товарів на складі, проведення інвентаризації, реєстрація надходжень і відвантажень, формування звітів тощо. У центральній частині діаграми — сценарії використання (use cases), що моделюють дії користувача в системі. Ці сценарії пов'язані з відповідними класами із діаграм класів, які реалізують бізнес-логіку.

Класи, що реалізують функціональність, представлені у вигляді прямокутників і охоплюють елементи прикладної логіки, API та інтерфейсні компоненти. Вони мають зв'язки з елементами use case, що відображає пряме відношення між вимогами користувача та реалізацією системи.

З правої сторони діаграми розміщені тестові випадки (test cases), які демонструють, яким чином кожен сценарій використання перевіряється в процесі модульного та інтеграційного тестування. Таким чином, забезпечується повнота трасування вимог до реалізації та валідації програмного продукту.

Target +	Laba_12::Вхід в систему	Laba_12::Керування користув.	Laba_12::Контроль термінів пн	Laba_12::Пошук товарів на ск.	Laba_12::Проведення інвентар	Laba_12::Реєстрація відванта.	Laba_12::Реєстрація надходж	Laba_12::Формування звітів
+ Source								
Class Diagram::IProductRe...			↑	↑				
Class Diagram::IReportGen...								↑
Class Diagram::ITransactio...					↑	↑	↑	
Class Diagram::IUserService	↑	↑						

Рисунок 2.8 – Зв'язок між Інтерфейсами та Тест Кейсами

Джерело сформовано автором на основі виконаного дослідження

Матриці трасування дозволяють деталізовано показати взаємозв'язки між різними компонентами інформаційної системи. На рисунку 2.8 зображена матриця зв'язків між класами та прецедентами, яка відображає, які класи з діаграми класів реалізують або підтримують відповідні сценарії використання.

Source: Class Diagram

Set

Type: Class

Link Type: Trace

Profile:

Target: UseCase

Set

Type: UseCase

Direction: Both

Overlays: <None>

Target +	UseCase:Вхід в систему	UseCase:Керування користув	UseCase:Контроль термінів п	UseCase:Пошук товарів на ск	UseCase:Проведення інвента	UseCase:Реєстрація відванта	UseCase:Реєстрація надходж	UseCase:Формування звітів
+ Source								
Class Diagram::FirebaseSer...								
Class Diagram::InventoryS...					↑			
Class Diagram::InventoryTr...						↑	↑	
Class Diagram::Product			↑	←				
Class Diagram::ReceivingO...								
Class Diagram::Report								↑
Class Diagram::ShippingOr...								
Class Diagram::User	↑	↑						

Рисунок 2.9 – Зв'язок між Класами та Use Cases

Джерело сформовано автором на основі виконаного дослідження

На рисунку 2.9 представлено матрицю зв'язків між функціональними вимогами та прецедентами. Така діаграма дозволяє перевірити відповідність кожної вимоги конкретному сценарію використання системи, забезпечуючи функціональне покриття і відповідність розробки поставленим завданням.

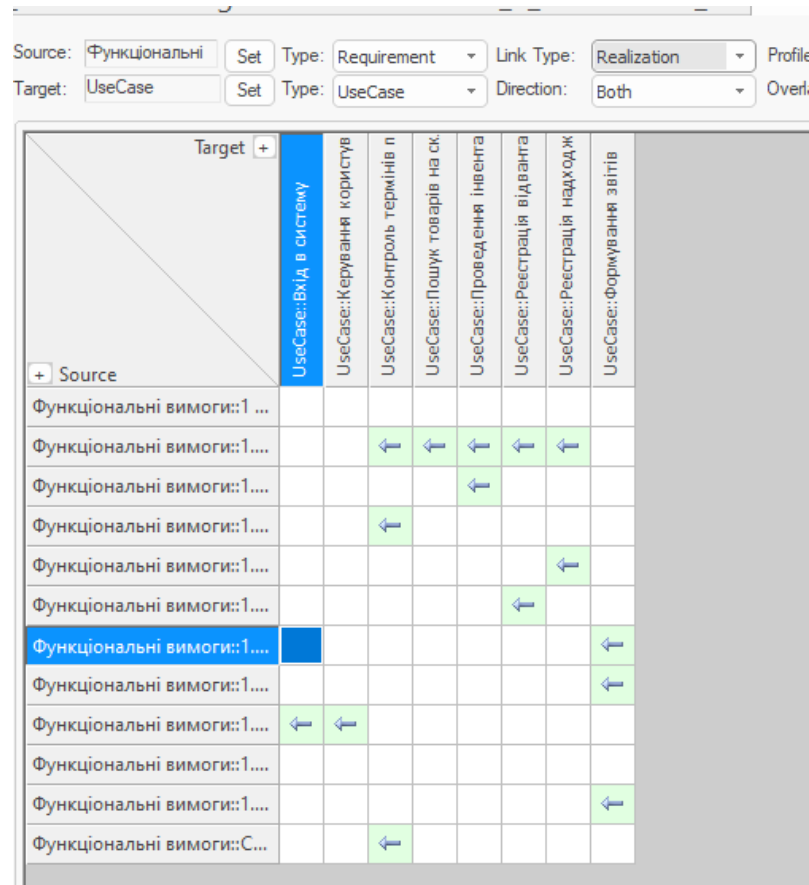


Рисунок 2.10 – Зв'язок між Класами та Use Cases

Джерело сформовано автором на основі виконаного дослідження

На рисунку 2.10 представлено матрицю трасування зв'язків між функціональними вимогами та прецедентами використання системи. Матриця відображає кореляцію між Source (функціональні вимоги) та Target (UseCase - прецеденти використання), де зелені стрілки вказують на реалізацію конкретної функціональної вимоги через відповідний прецедент. Така діаграма трасування дозволяє забезпечити повне покриття всіх функціональних вимог відповідними сценаріями використання, перевірити цілісність системної архітектури та гарантувати, що кожна бізнес-вимога має свою реалізацію у вигляді конкретного прецеденту, що є критично важливим для валідації повноти проектного рішення.

РОЗДІЛ 3

ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ СИСТЕМИ

3.1 Інформаційне забезпечення.

3.1.1 Загальна характеристика інформаційного забезпечення

Інформаційне забезпечення розробленої системи управління складськими операціями являє собою сучасний комплекс рішень, побудований на основі хмарних технологій Google Firebase [10] та клієнтської частини на React[11]. Така архітектура забезпечує високу надійність, масштабованість та доступність системи в режимі 24/7.

Основу інформаційного забезпечення складають наступні компоненти: база даних Firestore для зберігання структурованих даних, Firebase Authentication для управління користувачами, Firebase Storage для файлів та документів, а також React-додаток як інтерфейс користувача. Дана архітектура забезпечує синхронізацію даних в реальному часі між усіма користувачами системи.

Типом носія даних виступають хмарні сховища Google, що гарантує високий рівень відмовостійкості та резервного копіювання. Вибір Firebase як основної платформи обумовлений її інтеграцією з екосистемою Google, простотою масштабування, вбудованими засобами безпеки та можливістю роботи в режимі реального часу.

Контроль інформації здійснюється на кількох рівнях: валідація даних на клієнтській стороні за допомогою React-форм, серверна валідація через Firebase Security Rules, а також автоматичне резервне копіювання та версіонування даних. Надійність забезпечується через розподілену архітектуру Firebase, а достовірність - через систему прав доступу та аудиту операцій.

3.1.2 Організація збору і передавання первинної інформації

Система отримує первинну інформацію з декількох джерел, що забезпечує комплексний облік складських операцій. Основними джерелами інформації виступають:

Внутрішні джерела: працівники складу, які через React-інтерфейс вносять дані про надходження товарів, відвантаження, інвентаризацію та переміщення; менеджери, що формують звіти та налаштовують параметри системи; адміністратори, які управляють користувачами та налаштуваннями безпеки.

Зовнішні джерела: постачальники, дані про яких надходять через API інтеграції або ручне введення; клієнти, замовлення яких обробляються через систему; зовнішні системи обліку, які можуть інтегруватися через Firebase Functions.

Періодичність надходження інформації варіюється залежно від типу операцій: дані про надходження та відвантаження товарів надходять щоденно в режимі реального часу; інвентаризаційні дані - щомісячно або за потреби; звітні дані формуються щоденно, щотижнево та щомісячно; налаштування системи оновлюються за необхідності.

Спосіб передавання інформації здійснюється через веб-інтерфейс React-додатку з автоматичною синхронізацією через Firestore, що забезпечує миттєве оновлення даних для всіх користувачів системи.

3.1.3 Структура інформаційних масивів

Структура даних системи складського обліку організована у вигляді взаємопов'язаних колекцій NoSQL бази даних Firestore, кожна з яких має чітко визначену структуру та призначення.

Масив loadings (завантаження товарів) - містить дані про надходження товарів на склад, з інформацією про партії, постачальників, категорії, обсяги та дати постачання.

Ідентифікатор масиву: loadings

Носій: Firebase Firestore

Максимальний обсяг: 1 000 000 записів

Довжина запису: ~512 байт

Метод організації: Документно-орієнтований

Ключі упорядкування: createdAt, barcode

Індексний масив: IDX_load_time

Таблиця 3.1 – інформаційний масив loadings

Найменування	Ідентифікатор	Позначення	Формат	Бізнес-правила	Зв'язки	Ключ	Умова	Обов'язк.	Індекс
Штрихкод	barcode	bc	STRING	13 цифр	products(barcode)	FK	≠ null	так	ІНД
Назва товару	productName	pn	STRING	Не порожнє	-	-	≠ null	так	
Дата створення партії	createdAt	dt_c	TIMESTAMP	Дата і час занесення	-	PK	> 2020	так	ІНД
Кількість	quantity	q	INTEGER	> 0	-	-	> 0	так	
Ціна	price	p	NUMBER	> 0.0	-	-	> 0	так	

Категорія	categoryId	cid	STRING	Зовн. ключ	categories(id)	FK	≠ null	так	
Склад	storageId	sid	STRING	Зовн. ключ	storages(id)	FK	≠ null	так	
Секція складу	storageAreaId	said	STRING	Зовн. ключ	storageAreas(id)	FK	≠ null	так	
Постачальник	supplierId	spid	STRING	Зовн. ключ	suppliers(id)	FK	≠ null	так	
Термін придатності	expiryDate	expd	DATE (string)	Формат YYYY-MM-DD	-	-	≠ null	так	
Номер партії	batchNumber	bn	STRING	Унікальність у межах однієї категорії	-	УН	≠ null	так	ІНД
Відповідальний	responsible Person	rp	STRING	Email	users(email)	FK	≠ null	так	

Масив transactions (транзакції) містить інформацію про переміщення товарів між об'єктами (склад, магазин), із вказанням обсягів, типу операції, часу, відповідальних осіб та пов'язаної партії.

Ідентифікатор масиву: transactions

Носій: Firebase Firestore

Максимальний обсяг: 500 000 записів

Довжина запису: ~640 байт

Метод організації: Документно-орієнтований

Ключі упорядкування: timestamp, type

Індексний масив: IDX_trans_time_type

Таблиця 3.2 – інформаційний масив transactions

Найменування	Ідентифікатор	Позначення	Формат	Бізнес-правила	Зв'язки	Ключ	Умова	Обов'язк.	Індекс
Штрихкод	barcode	bc	STRING	13 цифр	products(barcode)	FK	≠ null	так	ІНД
Тип операції	type	t	STRING	Напр. product_unloading_fifo	-	-	≠ null	так	ІНД
Дата/час	timestamp	ts	TIMESTAMP	Час виконання	-	PK	> 2020	так	ІНД
Кількість	unloaded Quantity	uq	INTEGER	> 0	-	-	> 0	так	
Причина	reason	r	STRING	Не порожня	-	-	≠ null	так	
Партія	batchId	bid	STRING	Зовн. ключ	loadings(batchNumber)	FK	≠ null	так	
Склад відправник	sourceStorageId	ssid	STRING	Не обов'язкове	storages(id)	-	-	ні	
Об'єкт призначення	destinationId	did	STRING	Зовн. ключ	shops(id) або storages(id)	FK	≠ null	так	
Відповідальний	responsiblePerson	rp	STRING	Email	users(email)	FK	≠ null	так	

Масив products (продукти) – містить основну інформацію про товари, що зберігаються на складі: назву, штрихкод, категорію, ціну, мінімальну кількість, термін придатності та місце зберігання.

Ідентифікатор масиву: products

Носій: Firebase Firestore

Максимальний обсяг: 100 000 записів

Довжина запису: ~384 байт

Метод організації: Документно-орієнтований

Ключі упорядкування: createdAt, barcode

Індексний масив: IDX_prod_code

Таблиця 3.3 – інформаційний масив products

Найменування	Ідентифікатор	Позначення	Формат	Бізнес-правила	Зв'язки	Ключ	Умова	Обов'язк.	Індекс
Назва товару	name	n	STRING	Не порожнє	-	-	≠ null	так	
Штрихкод	barcode	bc	STRING	Унікальний	loadings(barcode)	УН	≠ null	так	ІНД
Категорія	idCategory	cid	STRING	Зовн. ключ	categories(id)	FK	≠ null	так	
Кількість	quantity	q	INTEGER	≥ 0	-	-	≥ 0	так	
Мін. залишок	minQuantity	minq	INTEGER	≥ 0	-	-	≥ 0	так	
Ціна	price	p	NUMBER	> 0	-	-	> 0	так	
Вага/Об'єм	weightOrVolume	wv	STRING	Напр. "1 кг", "0.5 л"	-	-	≠ null	так	
Склад	idStorage	sid	STRING	Зовн. ключ	storages(id)	FK	≠ null	так	
Секція	idStorageArea	said	STRING	Зовн. ключ	storageAreas(id)	FK	≠ null	так	
Постачальник	idSupplier	spid	STRING	Зовн. ключ	suppliers(id)	FK	≠ null	так	
Дата придатності	expiryDate	exp	DATE	Формат YYYY-MM-DD	-	-			

Масив unloadings (відвантаження товарів) – зберігає інформацію про відвантаження товарів зі складу, включаючи дату, відповідальних осіб, кількість, пункт призначення та тип операції.

Масив unloadings (відвантаження товару)

Ідентифікатор масиву: unloadings

Носій: Firebase Firestore

Максимальний обсяг: 500 000 записів

Довжина запису: ~480 байт

Метод організації: Документно-орієнтований

Ключі упорядкування: createdAt, barcode

Індексний масив: IDX_unload_time

Таблиця 3.4 – інформаційний масив unloadings

Найменування	Ідентифікатор	Позначення	Формат	Бізнес-правила	Зв'язки	Ключ	Умова	Обов'язк.	Індекс
Штрихкод	barcode	bc	STRING	-	products(barcode)	FK	≠ null	так	ІНД
Назва партії	batchName	bn	STRING	-	loadings(batchNumber)	FK	-	так	
Кількість	quantity	q	INTEGER	> 0	-	-	> 0	так	
Дата	unloadingDate	ud	DATE	Формат YYYY-MM-DD	-	-	≠ null	так	ІНД
Призначення	destinationId	did	STRING	Зовн. ключ	shops(id)	FK	≠ null	так	
Тип призначення	destinationType	dtype	STRING	shop або інше	-	-	≠ null	так	
Відповідальний	responsiblePerson	rp	STRING	Email	users(email)	FK	≠ null	так	

Масив categories (категорії товарів) – використовується для класифікації продукції за типами, спрощуючи навігацію, пошук та аналітику.

Масив categories (категорії товарів)

Ідентифікатор масиву: categories

Носій: Firebase Firestore

Максимальний обсяг: 1000 записів

Довжина запису: ~128 байт

Метод організації: Документно-орієнтований

Ключі упорядкування: name

Індексний масив: IDX_cat_name

Таблиця 3.5 – інформаційний масив categories

Найменування	Ідентифікатор	Позначення	Формат	Бізнес-правила	Зв'язки	Ключі	Умова	Обов'язк.	Індекс
Назва категорії	name	n	STRING	Не порожня	-	УН	≠ null	так	ІНД
Опис	description	d	STRING	-	-	-	-	ні	

Масив users (користувачі системи) – містить облікові записи користувачів, їхні ролі, контактні дані, рівень доступу та інші атрибути.

Масив users (користувачі)

Ідентифікатор масиву: users

Носій: Firebase Firestore

Максимальний обсяг: 10 000 записів

Довжина запису: ~256 байт

Метод організації: Документно-орієнтований

Ключі упорядкування: createdAt

Індексний масив: IDX_user_time

Таблиця 3.6 – інформаційний масив users

Найменування	Ідентифікатор	Позначення	Формат	Бізнес-правила	Зв'язки	Ключ	Умова	Обов'язк.	Індекс
UID	uid	uid	STRING	Унікальний	-	УН	≠ null	так	ІНД
Email	email	em	STRING	Унікальний, валідний email	-	УН	≠ null	так	
ПІБ/Ім'я	displayName	dn	STRING	-	-	-	≠ null	так	
Роль	role	r	STRING	Напр. admin, user	-	-	≠ null	так	
Адмін	isAdmin	ia	BOOLEAN	true/false	-	-	-	так	

Масив storages (склади) – містить інформацію про основні склади підприємства: назву, адресу, ємність та відповідальну особу.

Ідентифікатор масиву: storages

Носій: Firebase Firestore

Максимальний обсяг: 1 000 записів

Довжина запису: ~256 байт

Метод організації: Документно-орієнтований

Ключі упорядкування: name

Індексний масив: IDX_storage_name

Таблиця 3.7 – інформаційний масив storages

Найменування	Ідентифікатор	Позначення	Формат	Бізнес-правила	Зв'язки	Ключ	Умова	Обов'язк.	Індекс
Назва складу	name	n	STRING	Унікальна	-	УН	≠ null	так	ІНД
Адреса	address	a	STRING	-	-	-	≠ null	так	

Менеджер	managerName	m	STRING	-	-	-	≠ null	так	
Контакт	contactEmail	ce	STRING	Валідний email	users(email)	FK	≠ null	так	
Телефон	phone	ph	STRING	-	-	-	-	ні	

Масив storageAreas (зони зберігання) – деталізує внутрішню структуру складів, дозволяючи розміщувати товари по секціях або типах зберігання.

Масив storageAreas (зони зберігання)

Ідентифікатор масиву: storageAreas

Носій: Firebase Firestore

Максимальний обсяг: 5 000 записів

Довжина запису: ~192 байт

Метод організації: Документно-орієнтований

Ключі упорядкування: name

Індексний масив: IDX_area_name

Таблиця 3.8 – інформаційний масив storageAreas

Найменування	Ідентифікатор	Позначення	Формат	Бізнес-правила	Зв'язки	Ключ	Умова	Обов'язк.	Індекс
Назва секції	name	n	STRING	-	-	-	≠ null	так	
Тип зони	type	t	STRING	Температурна / інша	-	-	-	так	
Опис	description	d	STRING	-	-	-	-	ні	
Належить до складу	storageId	sid	STRING	Зовн. ключ	storages (id)	FK	≠ null	так	ІНД

Масив shops (торгові точки) – включає дані про магазини, куди постачається продукція: адресу, менеджера, контактну інформацію.

Масив shops (торгові точки)

Ідентифікатор масиву: shops

Носій: Firebase Firestore

Максимальний обсяг: 10 000 записів

Довжина запису: ~256 байт

Метод організації: Документно-орієнтований

Ключі упорядкування: name

Індексний масив: IDX_shop_name

Таблиця 3.9 – інформаційний масив shops

Найменування	Ідентифікатор	Позначення	Формат	Бізнес-правила	Зв'язки	Ключ	Умова	Обов'язк.	Індекс
Назва магазину	name	n	STRING	Унікальна	-	УН	≠ null	так	ІНД
Адреса	address	a	STRING	-	-	-	≠ null	так	
Менеджер	managerName	m	STRING	-	-	-	≠ null	так	
Контакт	contactEmail	ce	STRING	Email	users(email)	FK	≠ null	так	
Телефон	phone	ph	STRING	-	-	-	-	ні	

Масив suppliers (постачальники) – зберігає контактну інформацію про постачальників, назви компаній, адреси та інші дані, необхідні для обліку поставок.

Масив suppliers (постачальники)

Ідентифікатор масиву: suppliers

Носій: Firebase Firestore

Максимальний обсяг: 5 000 записів

Довжина запису: ~256 байт

Метод організації: Документно-орієнтований

Ключі упорядкування: name

Індексний масив: IDX_supplier_name

Таблиця 3.10 – інформаційний масив suppliers

Найменування	Ідентифікатор	Позначення	Формат	Бізнес-правила	Зв'язки	Ключ	Умова	Обов'язк.	Індекс
Назва постачальника	name	n	STRING	Унікальна	-	УН	≠ null	так	ІНД
Адреса	address	a	STRING	-	-	-	≠ null	так	
Контактна особа	contactPerson	cp	STRING	-	-	-	≠ null	так	
Email	email	e	STRING	Email	-	-	≠ null	так	
Телефон	phone	p	STRING	-	-	-	-	ні	

3.1.4 Вибір СКБД

Для реалізації системи складського обліку обрано Google Firestore — хмарну NoSQL документо-орієнтовану базу даних, що входить до складу Firebase. Вибір обумовлений потребою у роботі в реальному часі та простотою масштабування.[12]

Технічні переваги Firestore для складського обліку: автоматична синхронізація даних між усіма користувачами системи в реальному часі, що критично для актуального відображення залишків; підтримка запитів із фільтрацією та сортуванням за кількома полями; вбудована система безпеки (Firebase Security Rules) для контролю доступу.

Архітектурні рішення: гнучка структура документів дозволяє зберігати різнотипну інформацію без фіксованої схеми; автоматичне масштабування забезпечує стабільність при зростанні навантаження; підтримка офлайн-режиму з подальшою синхронізацією — гарантія безперервної роботи.

Економічна ефективність: модель оплати за фактичне використання знижує стартові витрати; не потрібна власна серверна інфраструктура; резервне копіювання та відновлення — автоматичні.

Обмеження та їх врахування в проектуванні: обмеження на розмір документа (1 МБ) — великі масиви винесено окремо; ліміт записів (1 операція/сек на документ) — розподілено по різних документах; створено необхідні індекси з урахуванням специфіки NoSQL.

3.1.5 Інформаційна та деталогічна моделі бази даних

Інфологічна модель системи складського обліку представляє концептуальну схему взаємозв'язків між основними бізнес-сутностями, адаптовану для специфіки NoSQL архітектури Firestore та вимог управління складськими операціями.[13]

Центральні сутності моделі включають Product (товар), Category (категорія товарів), Supplier (постачальник), Storage (склад), StorageArea (зона зберігання), Transaction (складська операція), Unloading (операція вивантаження), User (користувач системи). Кожна сутність має власний набір атрибутів та встановлені логічні зв'язки, які в NoSQL середовищі реалізуються через посилання на ідентифікатори документів.

- Сутність Product є ключовою в системі та містить всю необхідну інформацію для ідентифікації товару: штрих-код для унікальної ідентифікації, назву та опис, цінову інформацію, дані про кількість та мінімальні запаси, характеристики упаковки (вага/об'єм), інформацію про партію та термін придатності. Продукт пов'язаний з категорією (через idCategory), постачальником (idSupplier), складом (idStorage) та конкретною зоною зберігання (idStorageArea).

- Сутність Category організовує товари за функціональними групами з назвою та детальним описом призначення. Сутність Supplier містить контактну інформацію постачальників для ведення документообігу. Сутності Storage та StorageArea формують ієрархічну структуру місць зберігання, де склад містить кілька зон зберігання з різним призначенням.

- Сутність Transaction забезпечує повний аудит складських операцій, зберігаючи не тільки факт операції, але й повну контекстну інформацію на

момент її виконання: деталі товару, інформацію про склад та зону зберігання, дані про користувача, часову мітку. Сутність Unloading реалізує складну логіку вивантаження товарів за принципом FIFO з деталізацією по партіях.

- Сутність User управляє доступом до системи з підтримкою рольової моделі безпеки. Модель забезпечує цілісність даних через систему посилань та дублювання критичної інформації для оптимізації запитів, що є характерним для NoSQL підходу до проектування.

Фізична реалізація моделі в середовищі Google Firestore організована у вигляді колекцій документів з оптимізованою структурою для складських операцій та забезпечення високої продуктивності системи.

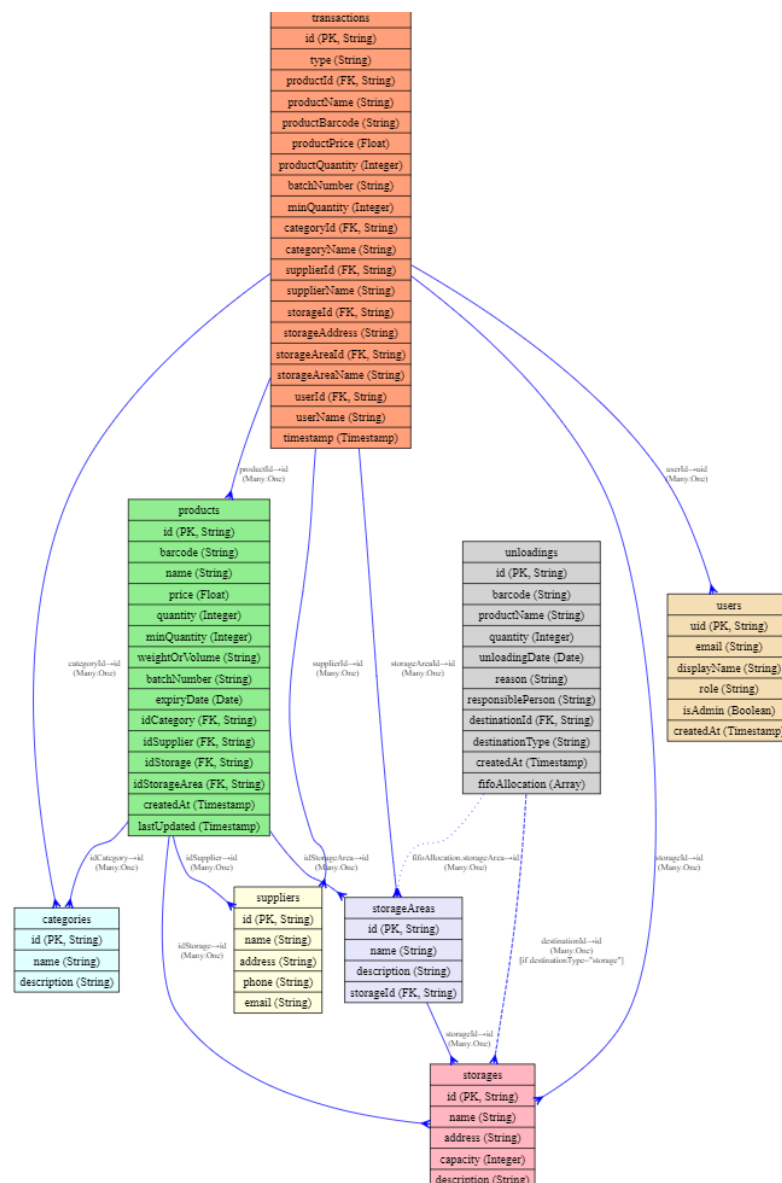


Рисунок 3.1– Інфологічна модель

Джерело: сформовано автором на основі виконаного дослідження

Налаштування індексації: compound indices створені для оптимізації складних запитів - products індексується за комбінацією (idCategory, name), transactions - за (type, timestamp), unloadings - за (unloadingDate, barcode). Single-field indices налаштовані для всіх полів, що використовуються в WHERE та ORDER BY клаузулах.

3.2 Технічне забезпечення

Для ефективного функціонування автоматизованої системи обліку та моніторингу матеріальних запасів необхідне відповідне технічне забезпечення, яке включає клієнтські пристрої, серверні ресурси, мережеву інфраструктуру та засоби хостингу інтерфейсу користувача.

Клієнтська частина представлена персональними комп'ютерами або мобільними пристроями (смартфонами, планшетами), з яких користувачі отримують доступ до системи через веб-інтерфейс або мобільний застосунок. Мінімальні вимоги до клієнтських пристроїв:

- Наявність сучасного браузера (Chrome, Firefox, Edge);
- Підключення до Інтернету;

Серверна частина реалізована на базі хмарної платформи Firebase, яка не потребує розгортання власного фізичного або віртуального сервера. Зберігання даних здійснюється у Google Firestore, що гарантує високу доступність, автоматичне масштабування та резервне копіювання.

Хостинг інтерфейсу користувача (UI) реалізовано через Firebase Hosting, який дозволяє розмістити React-додаток як набір статичних файлів (HTML, CSS, JavaScript). Для роботи UI не потрібен окремий сервер, оскільки вся логіка взаємодії з базою даних реалізується безпосередньо на клієнті за допомогою Firebase SDK. Це значно спрощує розгортання та знижує витрати на підтримку інфраструктури.

Мережева інфраструктура повинна забезпечувати стабільне підключення клієнтів до хмарної бази даних. Мінімальні вимоги до швидкості з'єднання — від 1 Мбіт/с для забезпечення коректної роботи в реальному часі. Для внутрішніх складів доцільно використовувати Wi-Fi мережу з підтримкою стандарту не нижче 802.11n.

Загальні технічні вимоги:

- Операційна система: Windows 10/11, Linux або Android/iOS для мобільного використання;
- Процесор: не нижче Intel Core i3 / ARM A55;
- Оперативна пам'ять: від 4 ГБ;
- Стабільне підключення до Інтернету.

Запропонована технічна інфраструктура дозволяє забезпечити стабільну, безпечну та масштабовану роботу системи без необхідності значних початкових витрат.

3.3 Програмне забезпечення

Для реалізації автоматизованої системи обліку та моніторингу матеріальних запасів було використано сучасний стек вебтехнологій, орієнтований на швидку розробку, масштабованість та зручність супроводу.

Клієнтська частина реалізована з використанням бібліотеки React — сучасного JavaScript-фреймворку для створення вебзастосунків. Завдяки компонентному підходу забезпечено модульність коду, легкість у тестуванні та повторному використанні елементів інтерфейсу.[15][16]

Структура React-додатку розділена на окремі логічні модулі:

- auth/ — компоненти авторизації (LoginForm.js, RegistrationForm.js);
- dashboard/, home/, inventory/, loading/, unloading/ — окремі частини інтерфейсу, що відповідають за перегляд залишків, завантаження/розвантаження товарів;

- layout/ — компоненти загального макету (шапка, футер, захист маршрутів);
- contexts/ — реалізовано контекст для глобального управління станом користувача або даних (React Context API).

Серверна частина побудована на хмарній платформі Firebase, яка забезпечує:

Базу даних Firestore — документо-орієнтоване сховище даних з підтримкою запитів у реальному часі

- Аутентифікацію Firebase Authentication — для реєстрації та входу користувачів;
- Хостинг Firebase Hosting — для розміщення React-додатку як статичного вебсайту;
- Правила безпеки (Security Rules) — для обмеження доступу до даних відповідно до ролей користувачів.

Підключення до Firebase реалізовано в окремому конфігураційному файлі `FirebaseConfig.js`, що містить параметри ініціалізації та імпортовані сервіси.

Зовнішні бібліотеки, що використовувались у проєкті:

- `react-router-dom` — для реалізації маршрутизації між сторінками;
- `firebase` — офіційний SDK для роботи з Firebase;
- `dotenv` — для зберігання секретних ключів у `.env` файлі.

Процес розробки вівся у середовищі Visual Studio Code з використанням Git для контролю версій. Проєкт ініціалізовано за допомогою `create-react-app`, що надало стандартну структуру та конфігурацію для швидкого старту розробки.

Завдяки обраному стеку технологій забезпечено:

- гнучкість при реалізації логіки обліку;
- швидкий рендеринг інтерфейсу;
- масштабованість за рахунок хмарної інфраструктури;

- зручність розгортання та оновлення без додаткового серверного обслуговування.

3.4 Результати реалізації автоматизованої системи

Під час впровадження автоматизованої системи для складу було реалізовано низку практичних технічних і програмних рішень, що забезпечили основу для автоматизації обліку та контролю складських операцій.

Розроблена система орієнтована на потреби складських підприємств і забезпечує комплексний облік та контроль руху товарно-матеріальних цінностей. Вона дозволяє автоматизовано відстежувати надходження, відвантаження, проводити інвентаризацію та формувати звітність.

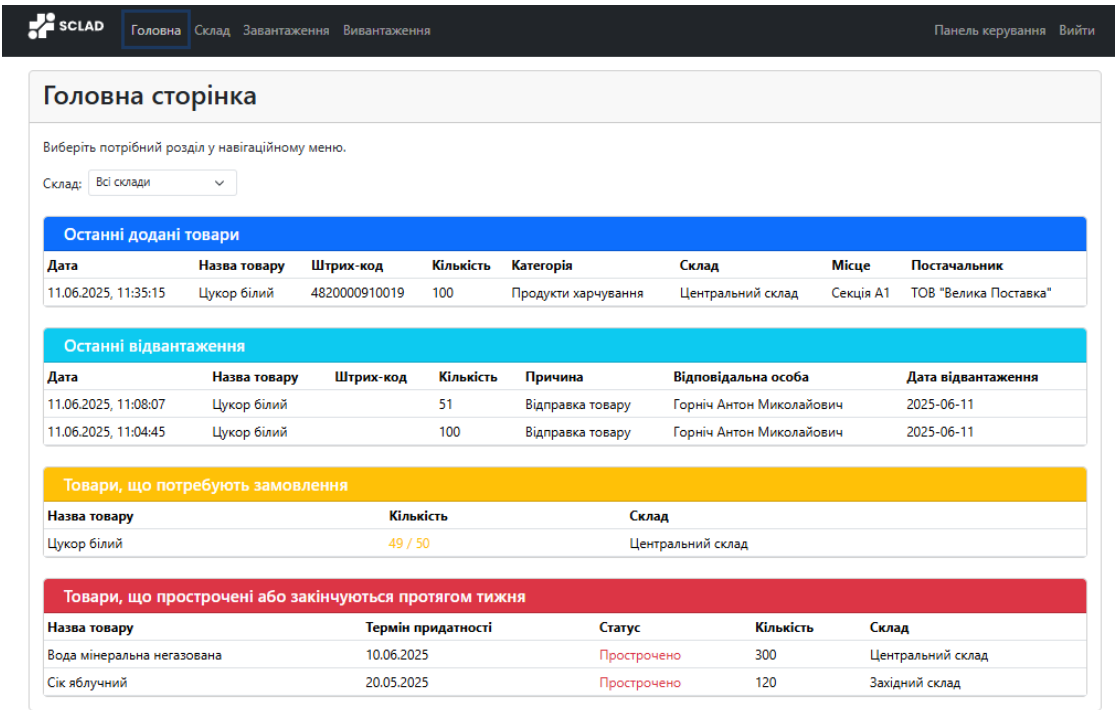


Рисунок 3.2– Головна сторінка

Джерело: сформовано автором на основі виконаного дослідження

Головний екран системи — це централізована інформаційна панель, що надає огляд складських операцій і забезпечує швидкий доступ до основних функцій: "Головна", "Склад", "Завантаження", "Вивантаження". Інтерфейс

побудований з інформаційних блоків, кожен із яких відповідає за певну ділянку роботи.

Центральна частина складається з чотирьох основних блоків:

- Останні додані товари (синій) — містить дані про нещодавні надходження: дата, назва, штрих-код, кількість, постачальник тощо.
- Останні відвантаження (блакитний) — показує останні операції відпуску товарів із зазначенням дати, кількості, відповідальної особи.
- Товари, що потребують замовлення (жовтий) — відображає товари з критично низьким рівнем запасів для своєчасного поповнення.
- Прострочені товари (червоний) — інформує про продукцію з вичерпаним або близьким до завершення терміном придатності.

Колірне кодування та табличний формат забезпечують зручне візуальне сприйняття й ефективну роботу користувачів.

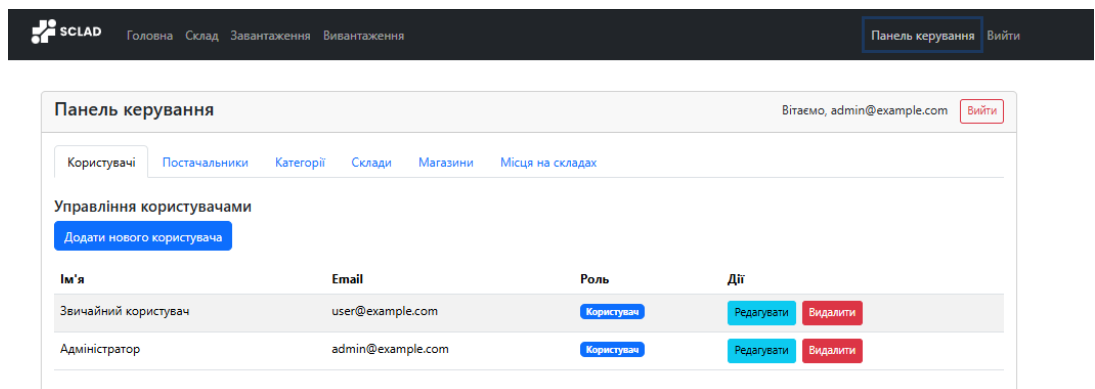


Рисунок 3.3– Панель керування

Джерело: сформовано автором на основі виконаного дослідження

Панель керування системи забезпечує адміністрування ключових елементів складської діяльності. Інтерфейс побудований у вигляді вкладок, що дозволяє зручно перемикатися між основними розділами: Користувачі, Постачальники, Категорії, Склади, Магазини, Місця на складах. Ця сторінка доступна тільки адміністратору

У верхній частині розташована навігаційна панель з доступом до основних модулів системи: У правому куті відображається привітання з email користувача та кнопка Вийти для завершення сеансу

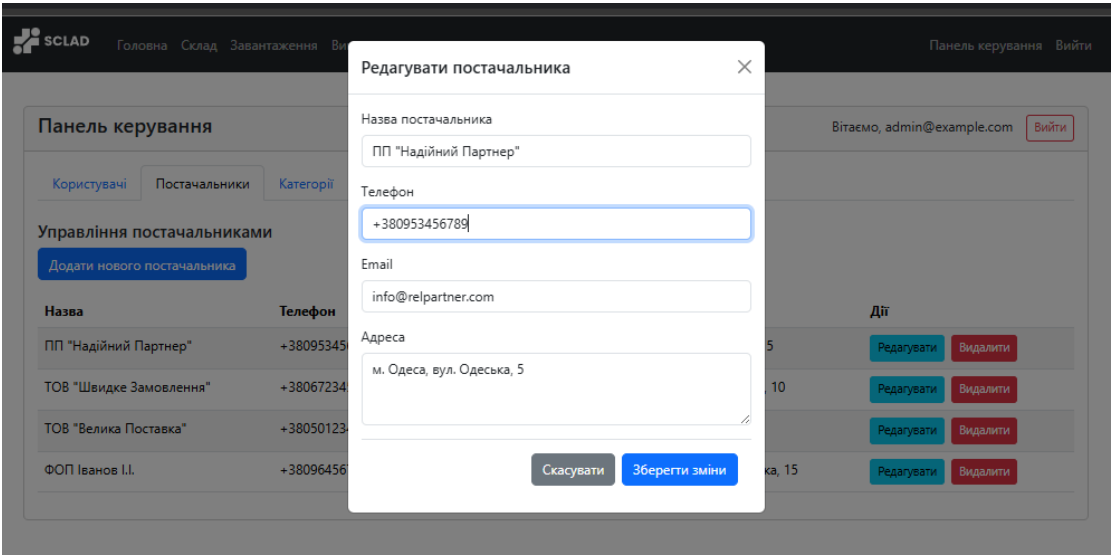


Рисунок 3.4– Вікно редагування пастачальника

Джерело: сформовано автором на основі виконаного дослідження

В панелі керування для кожного елемента також доступна можливість редагування, додавання, та видалення товарів

The image displays the 'Склад' (Warehouse) section of the SCLAD system. It includes a search and filter section at the top, followed by a table of goods stored in the warehouse.

Список товарів на складах

Фільтри та пошук

Пошук за назвою або штрих-кодом:

Категорія: Постачальник: Склад: Місце на складі:

Знайдено товарів: 13 з 13

№	Назва товару	Штрих-код	№ Партії	Ціна	Вага/Об'єм	Термін придатності	Категорія	Постачальник	Склад	Місце	Кількість	Мін. кількість	Дії
1	Цукор білий	4820000910019	-	42.50 грн	1 кг	31.12.2025	Продукти харчування	ТОВ "Велика Поставка"	Центральний склад	Секція A1	49	50	Ред Ви
2	Цукор білий	4820000910019	BATCH-20250611110345	42.50 грн	1	31.12.2025	Продукти харчування	ТОВ "Велика Поставка"	Центральний склад	Секція A1	200	50	Ред Ви
3	Цемент будівельний	4820000960014	-	165.00 грн	25 кг	01.12.2025	Будівельні матеріали	ФОП Іванов І.І.	Східний склад	Секція C1	40	10	Ред Ви
4	Футболка чоловіча	4820000940016	-	350.00 грн	150 г	-	Одяг	ПП "Надійний Партнер"	Центральний склад	Секція B2	100	20	Ред Ви
5	Макарони спагетті	4820000910033	-	45.20 грн	500 г	25.11.2025	Продукти харчування	ПП "Надійний Партнер"	Південний склад	Секція A1	180	40	Ред Ви
6	Вода мінеральна негазована	4820000920018	-	18.90 грн	1.5 л	10.06.2025	Напої	ТОВ "Швидке Замовлення"	Центральний склад	Секція A2	300	100	Ред Ви
7	Наушники бездротові	4820000930024	-	1250.00 грн	200 г	-	Електроніка	ФОП Іванов І.І.	Центральний склад	Секція B1	30	5	Ред Ви
8	Зарядний пристрій USB Type-C	4820000930017	-	450.00 грн	50 г	-	Електроніка	ФОП Іванов І.І.	Центральний склад	Секція B1	50	10	Ред Ви
9	Сік яблучний	4820000920025	-	54.30 грн	1 л	20.05.2025	Напої	ТОВ "Швидке Замовлення"	Західний склад	Секція A2	120	25	Ред Ви

Рисунок 3.5– Вікно сторінки “Склад”

Джерело: сформовано автором на основі виконаного дослідження

Вкладка "Склад" — сторінка для перегляду наявних товарів на складах. Вона відображає повний список усіх одиниць продукції з можливістю швидкого пошуку, фільтрації та, за потреби, адміністрування.

The screenshot shows a modal window titled "Редагувати товар" (Edit item) with a close button (X) in the top right corner. The form is organized into two columns and includes the following fields:

- Назва товару** (Item name): Цукор білий (White sugar)
- Штрих-код** (Barcode): 4820000910019
- Номер партії** (Batch number): Введіть номер партії (Enter batch number)
- Ціна** (Price): 42,5
- Вага/Об'єм** (Weight/Volume): 1 кг
- Термін придатності** (Expiry date): 31.12.2025
- Кількість** (Quantity): 49
- Мінімальна кількість** (Minimum quantity): 50
- Категорія** (Category): Продукти харчування (Food products)
- Постачальник** (Supplier): ТОВ "Велика Поставка" (LLC "Big Delivery")
- Склад** (Warehouse): Центральний склад (Central warehouse)
- Місце на складі** (Location in warehouse): Секція А1 (Section A1)

At the bottom right of the form are two buttons: "Скасувати" (Cancel) and "Зберегти зміни" (Save changes). Below the modal window, a table header is partially visible with columns: 20250611110345, грн, харчування, Поставка, склад, А1.

Рисунок 3.6— Вікно редагування товару

Джерело: сформовано автором на основі виконаного дослідження

Редагування товару — це функція, доступна лише адміністраторам, яка дозволяє оновлювати ключову інформацію про товар безпосередньо через інтерфейс.

Після натискання кнопки "Редагувати" у таблиці товарів відкривається спеціальне модальне вікно з формою, де можна змінити дані

Джерело: сформовано автором на основі виконаного дослідження

Сторінка "Заповнення товару на склад" інформаційної системи SCLAD, яка призначена для обліку партій товарів згідно з принципом FIFO. Сторінка дає змогу імпортувати товари з Excel або додати їх вручну, заповнюючи такі поля, як назва товару, штрих-код, ціна, вага, термін придатності, постачальник, склад, місце зберігання, кількість та мінімальний залишок. Цей функціонал необхідний для точного ведення складського обліку, контролю залишків і оптимізації логістичних процесів.

Вивантаження товарів

Фільтр по складу (необов'язково):

Шукати за: Пошук:

Доступні товари (всі склади):

Штрих-код	Назва	Доступна кількість	Склади	Дія
4820000910019	Цукор білий	449	Центральний склад: 449	Вибрати
4820000960014	Цемент будівельний	40	Сібрний склад: 40	Вибрати
4820000940016	Футболка чоловіча	100	Центральний склад: 100	Вибрати
4820000910033	Макарони спагетті	180	Глибинський склад: 180	Вибрати
4820000920018	Вода мінеральна негазована	300	Центральний склад: 300	Вибрати

Штрих-код товару *

Кількість для вивантаження *

Тип призначення

Пункт призначення

Дата вивантаження *

Причина вивантаження

Відповідальна особа *

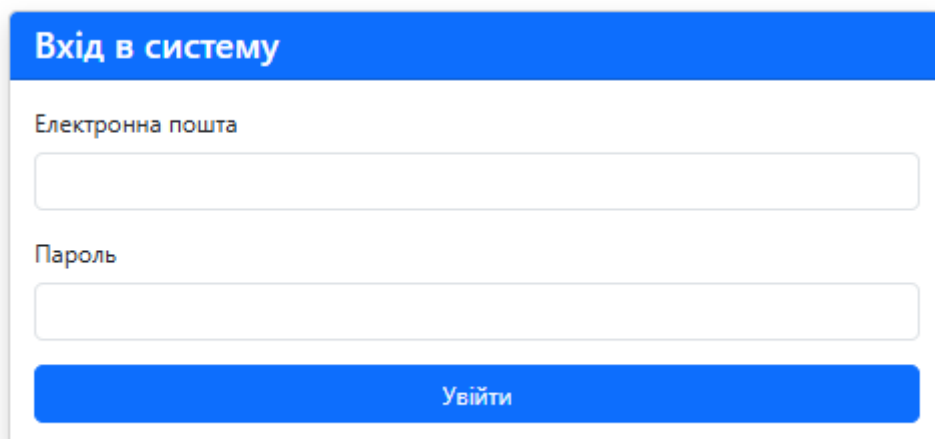
[Вивантажити товар](#)

Масове вивантаження з Excel
[Завантажити шаблон Excel](#) [Вибрати Excel файл](#)

Рисунок 3.7– Сторінка вивантаження товару на склад

Джерело: сформовано автором на основі виконаного дослідження

На рисунку 3.7 представлено сторінку "Вивантаження товарів" інформаційної системи, яка призначена для управління вихідними операціями зі складу. Інтерфейс включає фільтри для відбору товарів за складом і функцію пошуку, таблицю з переліком доступних товарів (штрих-код, назва, доступна кількість, склад), а також форму для оформлення вивантаження з полями для введення штрих-коду товару, типу призначення, пункту призначення, дати вивантаження, причини вивантаження та відповідальної особи. Додатково реалізовано функціонал масового вивантаження через імпорт Excel-файлів, що забезпечує ефективне управління великими обсягами вихідних операцій.



Вхід в систему

Електронна пошта

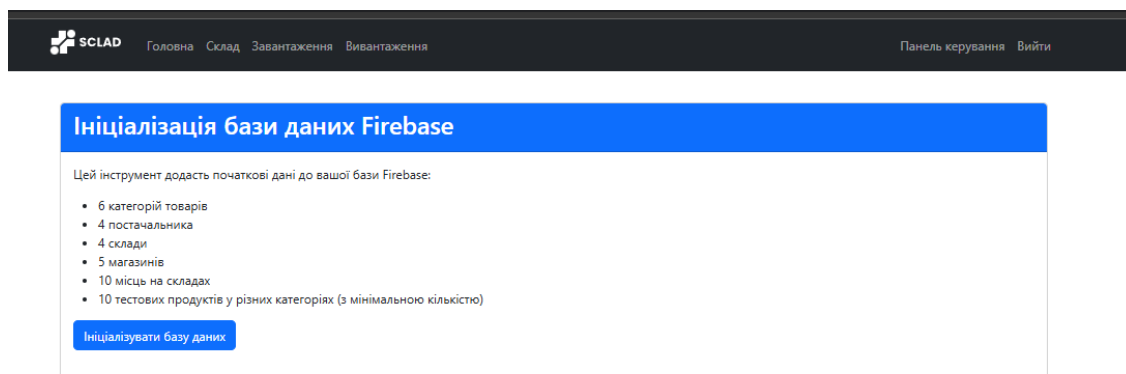
Пароль

Увійти

Рисунок 3.7– Вікно входу в акаунт

Джерело: сформовано автором на основі виконаного дослідження

Сторінка призначена для введення облікових даних — електронної пошти та пароля — з метою забезпечення безпечного доступу до функціоналу системи. Після успішного входу користувач отримує можливість працювати з внутрішніми модулями обліку, управління товарами та моніторингу складських операцій.



SCLAD Головна Склад Завантаження Вивантаження Панель керування Вийти

Ініціалізація бази даних Firebase

Цей інструмент додасть початкові дані до вашої бази Firebase:

- 6 категорій товарів
- 4 постачальника
- 4 склади
- 5 магазинів
- 10 місць на складах
- 10 тестових продуктів у різних категоріях (з мінімальною кількістю)

Ініціалізувати базу даних

Рисунок 3.7– Сторінка заповненням БД тестовими даними

Джерело: сформовано автором на основі виконаного дослідження

Сторінка призначена для заповнення бази даних даними для тестування системи

ВИСНОВКИ

У межах кваліфікаційної бакалаврської роботи було спроектовано та реалізовано сучасну веборієнтовану автоматизовану систему обліку та моніторингу матеріальних запасів складу. Система дозволяє контролювати надходження, витрати, залишки продукції, строки придатності, формувати звіти та швидко отримувати інформацію через зручний інтерфейс. Реалізовані функції сприяють підвищенню точності обліку, зменшенню втрат і ефективнішому управлінню.

Аналіз показав, що більшість доступних рішень орієнтовані на великі підприємства, є дорогими й складними у впровадженні. Це підтвердило доцільність створення доступної, масштабованої та гнучкої системи для малого й середнього бізнесу, адаптованої до українських реалій.

На основі зібраної інформації було сформульовано функціональні та нефункціональні вимоги. Основний функціонал включає облік товарів, контроль залишків, сповіщення про критичний рівень запасів, контроль строків зберігання, генерацію звітів і доступ з різних пристроїв.

Для реалізації обрано сучасні технології: React — для створення адаптивного веб-інтерфейсу, Firebase — як хмарну платформу для зберігання даних і реалізації бекенду. Це забезпечило швидкість розробки, масштабованість та зручне адміністрування.

Було реалізовано інтерфейс із розділами для перегляду товарів, редагування складу, формування звітів і обробки запитів на додавання, оновлення чи видалення записів у реальному часі. Система також підтримує авторизацію та налаштування прав доступу.

Проведено тестування системи у середовищі умовного невеликого складу. Перевірено функціональність: додавання, редагування та перегляд товарів, роботу звітів і сповіщень. Система показала коректність відображення даних, зручність інтерфейсу та відповідність вимогам.

Серед переваг — доступність, хмарна архітектура, відсутність потреби в дорогому серверному обладнанні, гнучкість у налаштуванні та можливість

масштабування. Інтерфейс інтуїтивно зрозумілий, а система адаптована до українських умов.

У перспективі планується впровадження сканування штрихкодів для пришвидшення введення товарів та зменшення помилок. Також можливе додавання елементів штучного інтелекту — для автоматичного розпізнавання товарів чи прогнозування закупівель. Передбачається розширення аналітичних функцій через інтерактивні дашборди й візуалізацію даних, а також створення мобільної версії для зручної роботи зі смартфонів.

Таким чином, створена система здатна підвищити ефективність управління складом, зменшити витрати та прискорити ухвалення рішень у сфері малого й середнього бізнесу. Вона має реальний практичний потенціал і може бути впроваджена в бізнес-процеси.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Тарасенко О.І. Методи оптимізації логістичних процесів за допомогою ІТ-рішень (дата звернення: 02.05.2025)
2. Верес М.Ю. Впровадження ERP-систем у логістичні модулі підприємств (дата звернення: 01.05.2025)
3. Дудка І.О. Логістичні модулі в корпоративних системах управління (дата звернення: 02.05.2025)
4. Simon J. RFID Technologies for Automatic Goods Movement Control (дата звернення: 03.05.2025)
5. Стейкхолдери [Електронний ресурс] – Режим доступу до ресурсу: <https://online.novaposhta.education/blog/hto-taki-stejkholderi-ta-yak-z-nimi-pracyuvati> (дата звернення: 11.05.2025)
6. Бізнес-вимоги до програмного забезпечення [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Вимоги_до_програмного_забезпечення (дата звернення: 07.05.2025)
7. Функціональні вимоги [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Функціональні_вимоги (дата звернення: 03.05.2025)
8. Нефункціональні вимоги [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Нефункціональні_вимоги (дата звернення: 12.05.2025)
9. Як будувати UML-діаграми. [Електронний ресурс] – Режим доступу до ресурсу: <https://dou.ua/forums/topic/40575/> (дата звернення: 07.05.2025)
10. Документація з роботою Firebase. [Електронний ресурс] – Режим доступу до ресурсу: <https://firebase.google.com/docs> (дата звернення: 12.05.2025)
11. Документація з React. [Електронний ресурс] – Режим доступу до ресурсу: <https://react.dev/> (дата звернення: 17.05.2025)

- 12.Порівняння SQL та NOSQL баз даних [Електронний ресурс] – Режим доступу до ресурсу:
https://tech.vernadskyjournals.in.ua/journals/2018/6_2018/part_2/7.pdf
(дата звернення: 01.05.2025)
- 13.Моделі даних [Електронний ресурс] – Режим доступу до ресурсу:
https://elib.lntu.edu.ua/sites/default/files/elib_upload/BD_2016_3/page5.html
1 (дата звернення: 02.05.2025)
- 14.Документація для роботи з Enterprise Architect [Електронний ресурс] – Режим доступу до ресурсу: <https://sparxsystems.com/> (дата звернення: 07.05.2025)
- 15.Firebase React Course [Електронний ресурс] – Режим доступу до ресурсу: <https://youtu.be/2hR-uWjBAGw?si=ejIcQnkvNWwlZvb4> (дата звернення: 07.05.2025)
- 16.Firebase React Course [Електронний ресурс] – Режим доступу до ресурсу: <https://www.udemy.com/course/reactjs-firebase-project/?couponCode=LETSLEARNNOW> (дата звернення: 03.05.2025)
- 17.Firebase React Course [Електронний ресурс] – Режим доступу до ресурсу: <https://www.udemy.com/course/reactjs-firebase-project/?couponCode=LETSLEARNNOW> (дата звернення: 03.05.2025)
- 18.Публікація: Розроблення автоматизованої системи управління виробничим підприємством з вбудованою системою складської логістики [Електронний ресурс] – Режим доступу до ресурсу:
<https://openarchive.nure.ua/entities/publication/acff067a-e227-4402-b787-9d3934d4d1f6> (дата звернення: 02.05.2025)
- 19.Система управління запасами [Електронний ресурс] – Режим доступу до ресурсу: <https://windtime.com.ua/obzory/skladskij/uk/vedenie-skladskij-oblik/#m1> (дата звернення: 10.05.2025)
- 20.Управління товарними запасами [Електронний ресурс] – Режим доступу до ресурсу: <http://yak.vlynko.com/?p=394> (дата звернення: 09.05.2025)

21. Діаграми Прецедентів (Use Case UML Diagram) [Електронний ресурс] – Режим доступу до ресурсу: <https://lvivqaclub.blogspot.com/2008/10/use-case-uml-diagram.html> (дата звернення: 07.05.2025)
22. Облік товару [Електронний ресурс] – Режим доступу до ресурсу: <https://remonline.ua/blog/how-to-inventory-in-store/> (дата звернення: 12.05.2025)
23. 1С:Підприємство [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/1С:_Підприємство (дата звернення: 08.05.2025)
24. Odoo Inventory [Електронний ресурс] – Режим доступу до ресурсу: https://www.odoo.com/uk_UA/app/inventory (дата звернення: 10.05.2025)
25. Zoho Inventory [Електронний ресурс] – Режим доступу до ресурсу: <https://www.zoho.com/inventory/> (дата звернення: 09.05.2025)

ДОДАТКИ

ДОДАТОК А

Діаграми послідовності

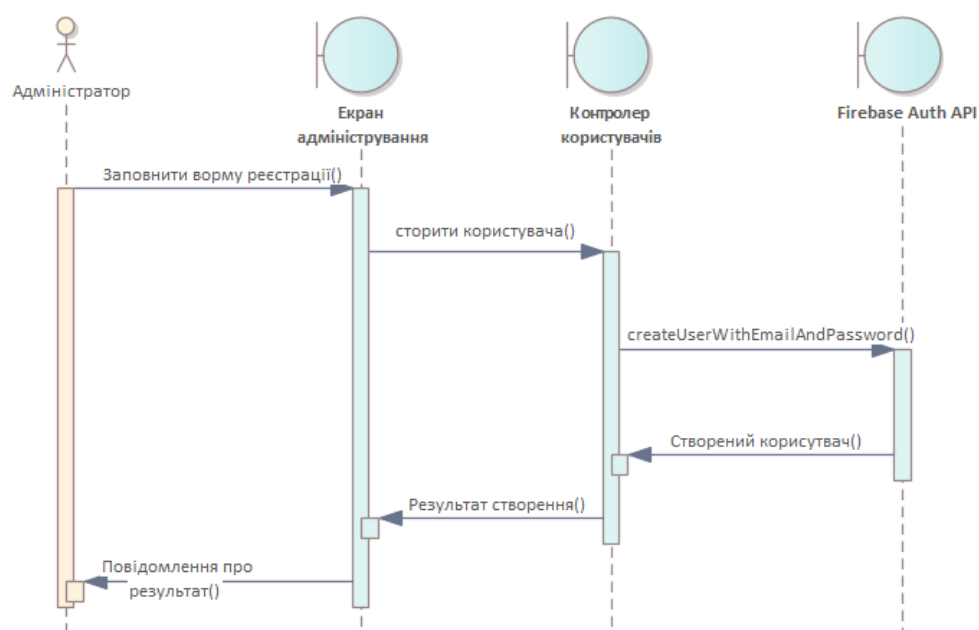


Рисунок А.1 – Діаграма послідовності процесу «Керування користувачами»

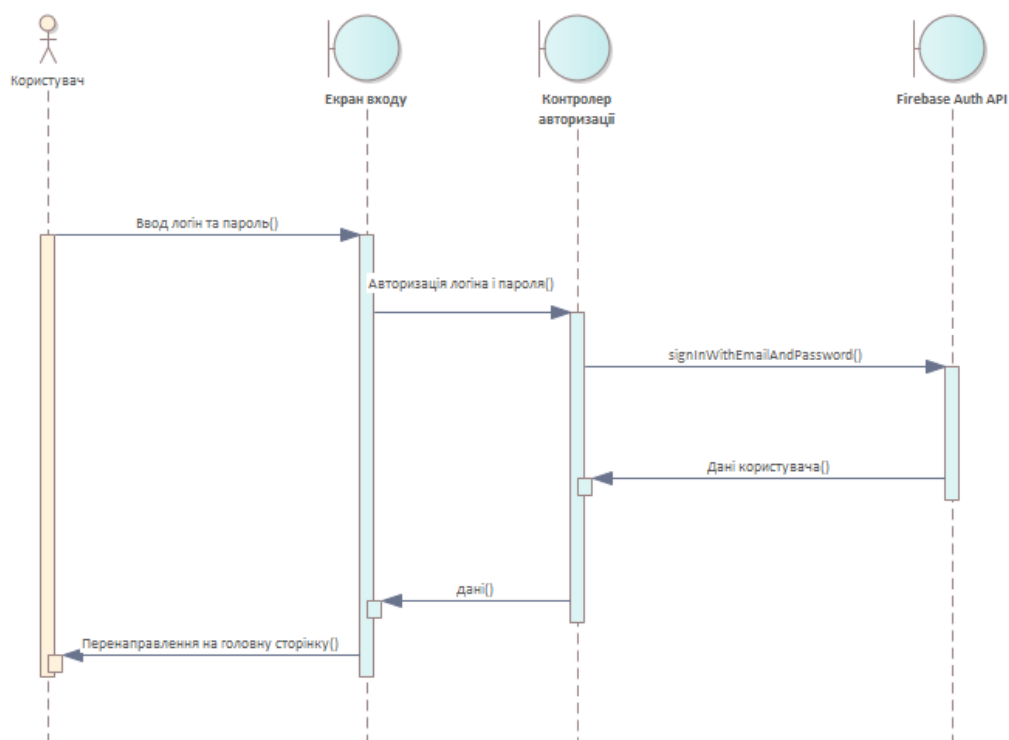


Рисунок А.2 – Діаграма послідовності процесу «Вхід в систему»

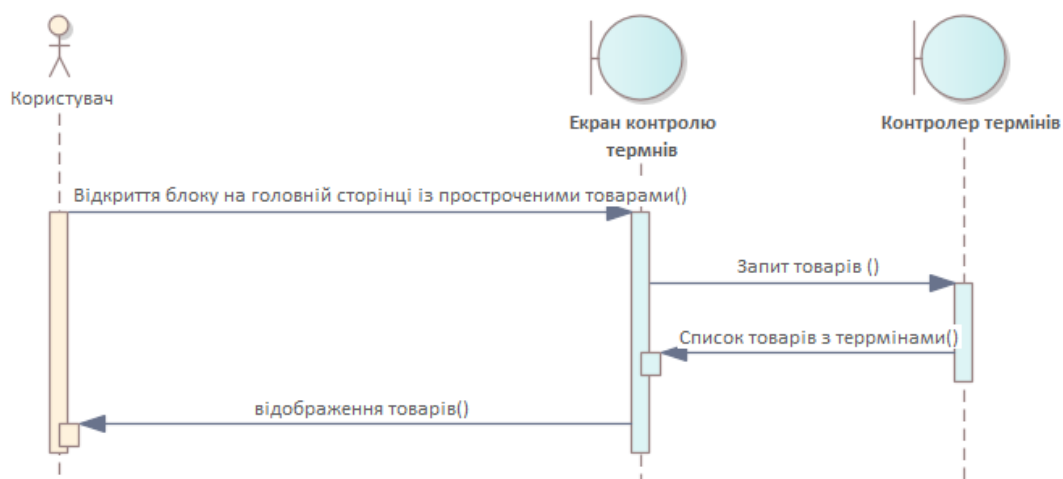


Рисунок А.3 – Діаграма послідовності процесу «Контроль терміну придатності»

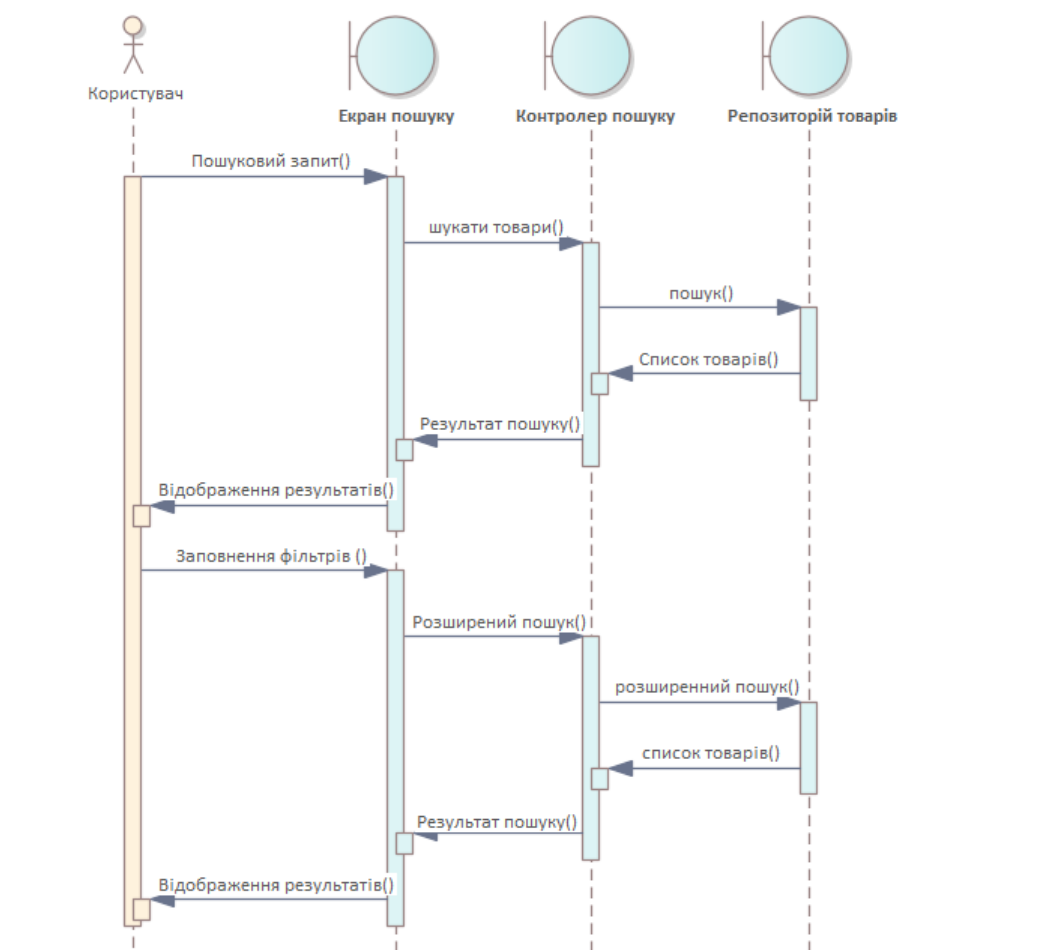


Рисунок А.4 – Діаграма послідовності процесу «Пошук товарів на складі»

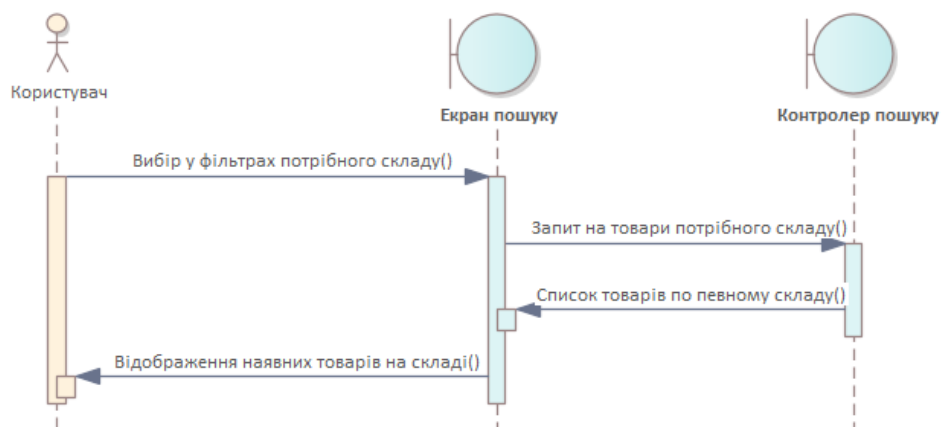


Рисунок А.5 – Діаграма послідовності процесу «Проведення інвентаризації»

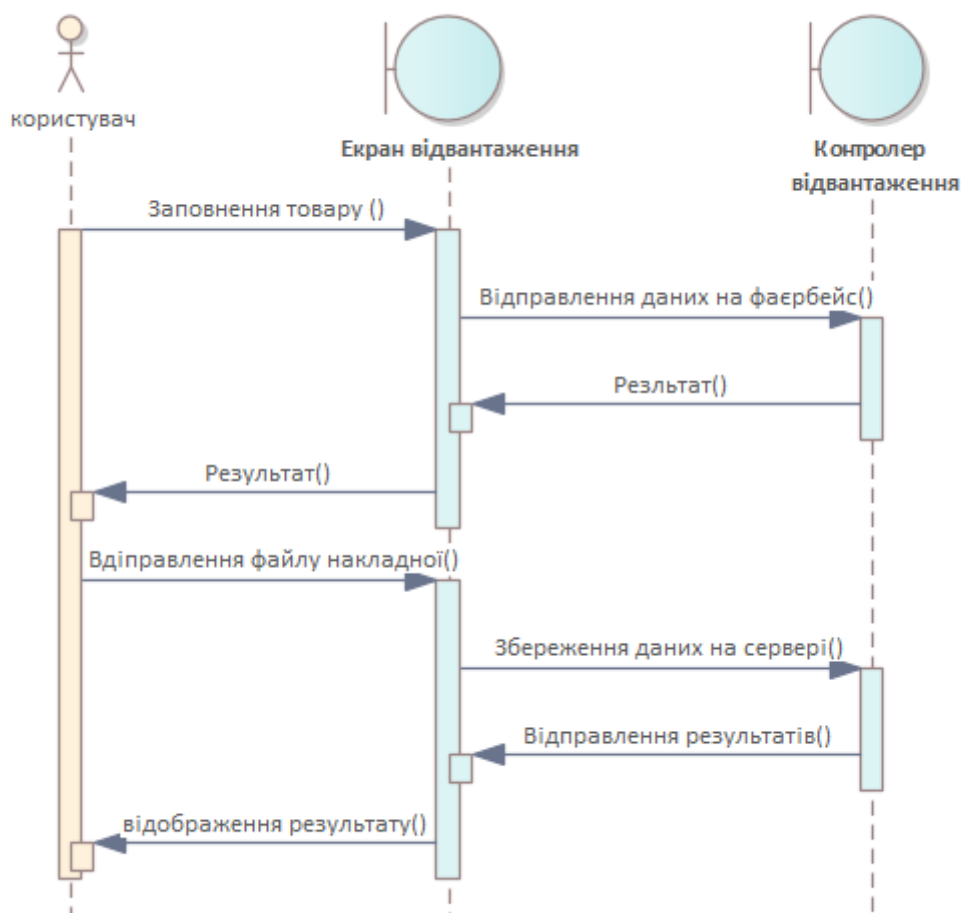


Рисунок А.6 – Діаграма послідовності процесу «Проведення інвентаризації»

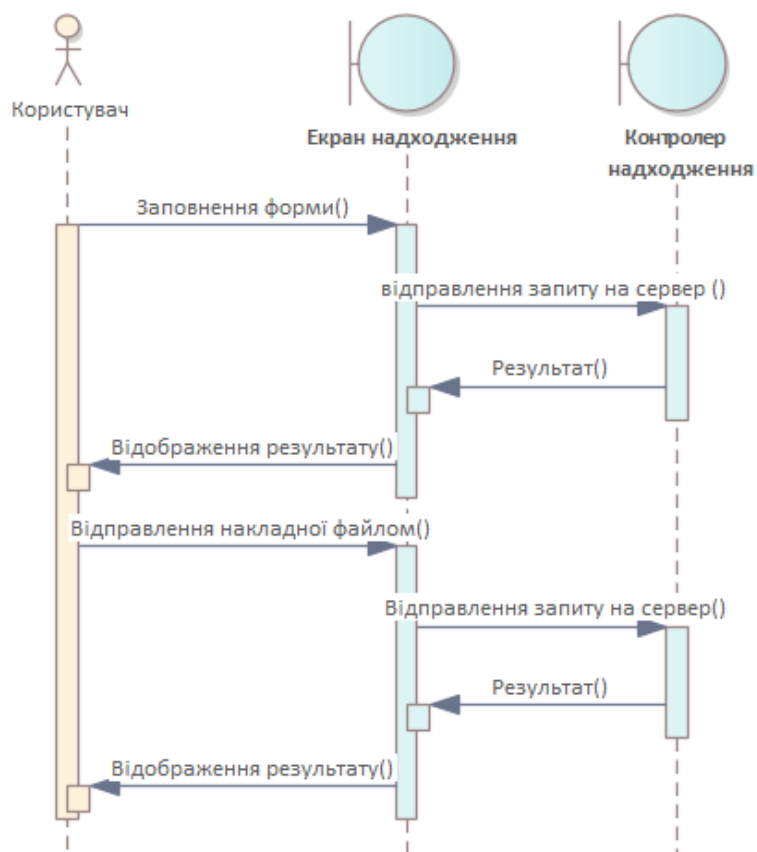


Рисунок А.7 – Діаграма послідовності процесу «Реєстрація надходження товарів»

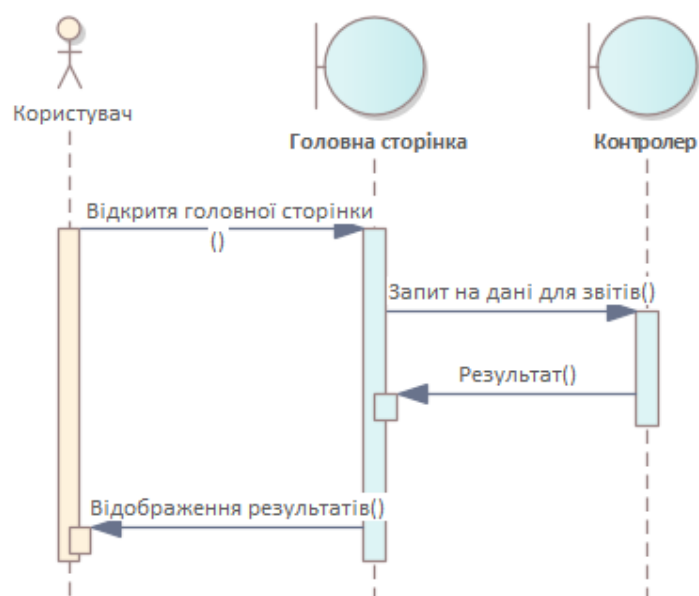


Рисунок А.8 – Діаграма послідовності процесу «Формування звітів»

ДОДАТОК Б

Деталізовані специфікації вимог

Item	Stereotype	Status	Difficulty	Priority
1 Інформаційна система замовлення залізничних квитків	Functional	Proposed	High	Medium
1.1 Система пошуку та фільтрації рейсів	Functional	Approved	Medium	Medium
1.1.1 перегляд доступних рейсів	Functional	Implement...	Low	Low
1.1.2 фільтрація результатів пошуку	Functional	Approved	Low	Low

Рисунок Б.1 – Специфікація функціональних вимог «Система пошуку та фільтрації рейсів»

Item	Stereotype	Status	Difficulty	Priority
1.2 Система бронювання та замовлення квитків	Functional	Proposed	Medium	Medium
1.2.1 можливість вибору рейсу та введення даних пасажирів	Functional	Implement...	Low	Low
1.2.2 можливість додавання до кошику	Functional	Proposed	Medium	Medium
1.2.3 можливість видалення або зміни квитків у кошику	Functional	Proposed	Medium	Medium
1.2.4 підтвердження та статус замовлень	Functional	Proposed	Medium	Medium
1.2.5 можливість перегляду актуального прогнозу погоди	Functional	Implement...	Low	Low

Рисунок Б.2 – Специфікація функціональних вимог «Система бронювання та замовлення квитків»

Item	Stereotype	Status	Difficulty	Priority
1.3 Система оплати квитків	Functional	Proposed	Medium	Medium
1.3.1 відображення інформації про квиток перед оплатою	Functional	Proposed	Medium	Medium
1.3.2 можливість оплати квитка	Functional	Proposed	Medium	Medium
1.3.3 генерація посадкового документу після оплати квитка	Functional	Proposed	Medium	Medium

Рисунок Б.3 – Специфікація функціональних вимог «Система оплати квитків»

Item	Stereotype	Status	Difficulty	Priority
1.4 Система автентифікації користувачів	Functional	Proposed	Medium	Medium
1.4.1 можливість авторизації за допомогою існуючого облікового запису	Functional	Proposed	Medium	Medium
1.4.2 можливість реєстрації	Functional	Proposed	Medium	Medium
1.4.3 доступ до кабінету користувача	Functional	Proposed	Medium	Medium
1.4.4 можливість автентифікації адміністратором, за наявності такої ролі	Functional	Proposed	Medium	Medium

Рисунок Б.4 – Специфікація функціональних вимог «Система автентифікації користувачів»

Item	Stereotype	Status	Difficulty	Priority
1.5 Система адміністрування	Functional	Proposed	Medium	Medium
1.5.1 доступ до інформації про користувачів та рейси	Functional	Implement...	Low	Low
1.5.2 можливість додавання рейсів та користувачів	Functional	Proposed	Medium	Medium
1.5.3 можливість редагування рейсів та управління обліковими записами користувачів	Functional	Proposed	Medium	Medium

Рисунок Б.5 – Специфікація функціональних вимог «Система адміністрування»

Item	Stereotype	Status	Difficulty	Priority
Інформаційна система замовлення залізничних квитків	interface	Proposed	High	High
інтерфейс користувача	Non-functional	Approved	Medium	Medium
інтуїтивно зрозуміла та легка для навігації система	Non-functional	Approved	High	Medium
сумісність з різними пристроями	Non-functional	Approved	High	Medium
адаптивний дизайн для коректного відображення на будь-якому екрані	Non-functional	Approved	High	Medium
можливість налаштувати темну тему інтерфейсу	Non-functional	Approved	Medium	Medium

Рисунок Б.6 – Специфікація нефункціональних вимог «Інтерфейс користувача»

Item	Stereotype	Status	Difficulty	Priority
Безпека	Non-functional	Implement...	High	Medium
шифрування паролів користувачів у БД	Non-functional	Implement...	High	Medium
розмежування прав доступу між користувачами та адміністраторами	Non-functional	Implement...	High	Medium

Рисунок Б.7 – Специфікація нефункціональних вимог «Безпека»

Item	Stereotype	Status	Difficulty	Priority
Швидкодія	Non-functional	Proposed	Medium	Medium
автоматична синхронізація даних між різними частинами системи	Non-functional	Proposed	Medium	Medium
швидкість обробки даних та масштабованість	Non-functional	Proposed	High	High

Рисунок Б.8 – Специфікація нефункціональних вимог «Швидкодія»

Item	Stereotype	Status	Difficulty	Priority
Надійність	Non-functional	Proposed	Medium	Medium
безперебійна доступність сервісу	Non-functional	Proposed	High	High
механізми запобігання втраті даних при помилках користувача	Non-functional	Validated	Medium	Low
підтвердження видалення запису з БД				

Рисунок Б.9 – Специфікація нефункціональних вимог «Надійність»

Item	Stereotype	Status	Difficulty	Priority
Масштабованість	Non-functional	Proposed	Medium	Medium
підтримка збільшення кількості користувачів без втрати продуктивності	Non-functional	Proposed	Medium	High
гнучка архітектура для інтеграції з зовнішніми сервісами	Non-functional	Implement...	Medium	Medium
API прогнозу погоди				

Рисунок Б.10 – Специфікація нефункціональних вимог «Масштабованість»

ДОДАТОК В

Звіт подібності

StrikePlagiarism.com

Українська версія

Дата звіту6/21/2025

Дата редагування---

Звіт не був оцінений

Звіт подібності

метадані

Назва організації
Kyiv National Economic University named after Vadym Hetman KNEU

Заголовок
Проектування автоматизованої системи обліку та моніторингу матеріальних запасів складу

АвторНауковий керівник / Експерт
Горніч Антон МиколайовичМиронцов М.Л.

підрозділ
кафедра інформаційних систем в економіці

Обсяг знайдених подібностей

Коефіцієнт подібності визначає, який відсоток тексту по відношенню до загального обсягу тексту було знайдено в різних джерелах. Зверніть увагу, що високі значення коефіцієнта не автоматично означають плагіат. Звіт має аналізувати компетентна / уповноважена особа.

5.43%

5.43%

КП 1

25

Довжина фрази для коефіцієнта подібності 2

3.15%

3.15%

КП 2

7959

Кількість слів

0.68%

0.68%

КЦ

64947

Кількість символів

Тривога

У цьому розділі ви знайдете інформацію щодо текстових спотворень. Ці спотворення в тексті можуть говорити про МОЖЛИВІ маніпуляції в тексті. Спотворення в тексті можуть мати навісний характер, але частіше характер технічних помилок при конвертації документа та його збереженні, тому ми рекомендуємо вам підходити до аналізу цього модуля відповідально. У разі виникнення запитань, просимо звертатися до нашої служби підтримки.

Заміна букв		7
Інтервали		0
Мікропробіли		0
Білі знаки		0
Парафрази (SmartMarks)		29

Подібності за списком джерел

Нижче наведений список джерел. В цьому списку є джерела із різних баз даних. Колір тексту означає в якому джерелі він був знайдений. Ці джерела і значення коефіцієнту подібності не відображають прямого плагіату. Необхідно відкрити кожне джерело і проаналізувати зміст і правильність оформлення джерела.

10 найдовших фраз

Колір тексту

ПОРЯДКОВИЙ НОМЕР	НАЗВА ТА АДРЕСА ДЖЕРЕЛА URL (НАЗВА БАЗИ)	КІЛЬКІСТЬ (ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ))
1	https://remonline.app/uk/blog/how-to-inventory-in-store/	47 0.59 %
2	https://ir.kneu.edu.ua/bitstreams/a5c7419f-c089-4b0c-8e70-3923cf3566a2/download	43 0.54 %
3	https://ir.kneu.edu.ua/bitstreams/958c3965-af04-49ca-8d39-b0213a87d87a/download	39 0.49 %
4	http://dspace.opu.ua/jspui/bitstream/123456789/12967/1/%D0%A6%D1%83%D1%80%D0%BA%D0%B0%D0%BD.pdf	33 0.41 %