

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет інформаційної безпеки та електронних комунікацій
(повне найменування факультету)

Кафедра інформаційних технологій електронних засобів
(повне найменування кафедри)

Пояснювальна записка

до дипломного проекту (роботи)

магістр

(ступінь вищої освіти)

на тему ДОСЛІДЖЕННЯ СИСТЕМ ІНТЕРНЕТ РЕЧЕЙ УПРАВЛІННЯ
ЗАПАСАМИ
RESEARCH ON IOT INVENTORY MANAGEMENT SYSTEMS

Виконав: студент(ка) 2 курсу, групи БК-612м
Спеціальності 151 Автоматизація та
комп'ютерно-інтегровані технології
(код і найменування спеціальності)

Освітня програма (спеціалізація)
Автоматизація, мехатроніка та робототехніка

СТОЛБУХОВ І.С.

(ПРІЗВИЩЕ та ініціали)

Керівник МИРОНОВА Н.О.

(ПРІЗВИЩЕ та ініціали)

Рецензент КОРОТУН А.В.

(ПРІЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет інформаційної безпеки та електронних комунікацій
Кафедра інформаційних технологій електронних засобів
Ступінь вищої освіти магістр
Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології
(код і найменування)
Освітня програма (спеціалізація) Автоматизація, мехатроніка та робототехніка
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

В.о.завідувача кафедри ІТЕЗ, к.т.н, доц.
О.Ю. МАЛІЙ
“ ” 2024 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

СТОЛБУХОВ Ілля Сергійович

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження систем інтернет речей управління запасами

Research on IoT inventory management systems

керівник проєкту (роботи) к.т.н., доц., МИРОНОВА Наталя Олексіївна,
(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові,)

затверджені наказом закладу вищої освіти від “23” квітня 2024 року № 162

2. Строк подання студентом проєкту (роботи) 10 червня 2024 року

3. Вихідні дані до проєкту (роботи) програма повинна забезпечувати відображення даних у RFID зчитувачі

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Вступ, 1 Аналіз предметної області та постановка задачі, 2 Дослідження комп'ютерно - інтегрованих технологій для реалізації завдання, 3 Проектування та реалізація системи управління запасами, 4 Експериментальне дослідження розробленої програми, Висновки, Додатки

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

Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
Розділи 1-4	МИРОНОВА Н.О., доцент		
Нормоконтроль	ПОСПЕЄВА І. Є., ст.викладач		

КАЛЕНДАРНИЙ ПЛАН

Студент(ка) _____ Ілля СТОЛБУХОВ
(підпис) (Ім'я ПРИЗВИЩЕ)

Керівник проєкту (роботи) _____ Наталя МИРОНОВА
(підпис) (Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:
94 с., 15 табл., 18 рис., дод., 23 джерел.

СИСТЕМА ІНТЕРНЕТУ РЕЧЕЙ, RFID ТЕГ, RFID ЗЧИТУВАЧ,
СЕНСОР, ДАТЧИК,

Об'єкт дослідження – програмна та апаратна частина систем управління запасами.

Предмет дослідження – система управління запасами на основі технології Інтернет речей .

Мета роботи – розробити прототип системи управління запасами автозапчастин з використанням технології Інтернету речей, що дозволить автоматизувати процеси прийому та обліку товарів з метою зниження рівня помилок та підвищення точності даних.

Для реалізації програмної частини використовувалася мова програмування C++ та середовище розробки Visual Studio Code.

В результаті були розроблені наступні класи: RFIDTag, Product, Warehouse, Inventory, InventoryManager, Customer, Salesperson, PointOfSale.

Програма призначена для автоматизації процесу прийому та обліку товарів з метою зниження рівня помилок, підвищення точності даних та мінімізації кількості помилок через людський фактор.

ABSTRACT

Explanatory note to the diploma qualifying work of the master: 94 pages, 15 tables, 18 figures, appendixes, 23 sources.

INTERNET OF THINGS SYSTEM, RFID TAG, RFID READER, SENSOR, SENSOR,

Object of research is software and hardware of inventory management systems.

The subject of research is an inventory management system based on the Internet of Things technology.

The purpose of the study is to develop a prototype of an inventory management system for auto parts using the Internet of Things technology, which will automate the processes of receiving and accounting for goods in order to reduce errors and increase data accuracy.

The C++ programming language and the Visual Studio Code development environment were used to implement the software part.

As a result, the following classes were developed: RFIDTag, Product, Warehouse, Inventory, InventoryManager, Customer, Salesperson, PointOfSale.

The program is designed to automate the process of receiving and accounting for goods in order to reduce the level of errors, increase data accuracy and minimize the number of errors due to human error.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК	8
ВСТУП	9
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ	11
1.1 Поняття IoT	11
1.2 Огляд проблеми управління запасами товару на фірмі з багатьма філіалами по Україні.....	12
1.3 Способи управління запасами товару без використання IoT	14
1.4 Впровадження IoT	20
1.5 Загальна структура IoT регулювання кількості товару.....	22
2 ДОСЛІДЖЕННЯ КОМП'ЮТЕРНО-ІНТЕГРОВАНИХ ТЕХНОЛОГІЙ ДЛЯ РЕАЛІЗАЦІЇ ЗАВДАННЯ	24
2.1 Вибір апаратних рішень	24
2.2 Вибір технології передачі даних.....	40
2.3 Вибір хмарної платформи	42
2.4 Проектування інтерфейсу користувача	45
2.5 Вибір технологій кібербезпеки	46
2.6 Використання IoT для системи управління запасами	47
2.7 Опис плану відправки товару з цетрального складу на філіал	50
2.8 Опис взаємодії комірника з системою IoT для управління запасами.....	53
2.9 Висновки з розділу	54
3 ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ СИСТЕМИ УПРАВЛІННЯ ЗАПАСАМИ	56
3.1 Архітектура системи	56
3.2 Діаграма прецедентів.....	57
3.3 Діаграма класів	60
3.4 Опис класів процесу прийому товару комірником шляхом сканування... 64	
3.5 Інтерфейс користувача у RFID зчитувачі.....	66
Висновки до розділу 3	68

4 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ РОЗРОБЛЕНОЇ ПРОГРАМИ .	69
4.1 Оцінювання системи інтернет речей управління запасами.....	69
4.2 Оцінка точності даних	70
4.3 Результати експериментального дослідження	71
4.4 Подальші покращення системи	73
4.5 Висновки до розділу 4	74
ВИСНОВКИ.....	75
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	77
Додаток текст програми	92

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

IoT – Internet of Things;

HFT – high frequency tag;

LFT – low frequency tag;

MWT – microwave tags;

RFID – Radio Frequency IDentification;

UHFT – ultra high frequency tag;

ВСТУП

Регулювання кількості товару є критично важливим аспектом управління запасами для будь-якої компанії, що займається роздрібним продажем, зокрема автозапчастин. Ефективне управління запасами дозволяє знизити витрати, покращити обслуговування клієнтів та забезпечити безперебійну роботу компанії.

У сучасних умовах інформаційних технологій та Інтернету речей (IoT) з'являються нові можливості для автоматизації та оптимізації процесів управління запасами.

Актуальність теми. У сучасному світі технології автоматизації процесів та операцій, які раніше виконувалися вручну, розвиваються надзвичайно швидкими темпами. Кожного дня людство вдосконалює та розробляє нові технології та інноваційні системи для полегшення та покращення праці. Автоматизовані системи знаходять застосування у різних галузях, таких як виробництво, складування, логістика, роздрібна торгівля та багато інших. Автоматизація також значно впливає на управління запасами. Тому, актуальним завданням є дослідження технологій та засобів для проектування систем управління запасами з використанням IoT та RFID-технологій.

Мета даної дипломної кваліфікаційної роботи – розробити прототип системи управління запасами автозапчастин з використанням технології Інтернету речей, що дозволить автоматизувати процеси прийому та обліку товарів з метою зниження рівня помилок та підвищення точності даних.

Завдання дослідження включають:

- дослідити існуючі технології реалізації систем управління запасами;
- дослідити апаратні рішення на основі IoT та RFID-технологій;
- виконати проектування та реалізацію прототипу системи управління запасами;
- провести експериментальне дослідження розробленої системи.

Об'єкт дослідження – програмна та апаратна частина систем управління запасами.

Предмет дослідження – система управління запасами на основі технології Інтернет речей.

Методи дослідження. Для реалізації прототипу системи управління запасами використані технологія Інтернет речей, хмарні технології, мова графічного моделювання систем.

Наукова новизна роботи полягає у тому, що виконано дослідження технологій проектування систем управління запасами з використанням технології Інтернету речей, що дозволяють автоматизувати процеси прийому та обліку товарів з метою зниження рівня помилок та підвищення точності даних.

Практична цінність результатів роботи полягає у тому, що розроблено прототип системи управління запасами, що дозволяє автоматизувати процеси реєстрації та відстеження товарів з достатнім рівнем точності.

За результатами роботи були подані тези на науково-практичну конференцію викладачів, науковців, молодих учених, аспірантів та студентів «Тиждень науки-2024».

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Поняття Інтернету речей

Інтернет речей (IoT) є однією з найвпливовіших технологічних інновацій нашого часу. В основі IoT лежить ідея з'єднання фізичних об'єктів із Інтернетом, що дозволяє їм збирати, обмінюватися даними та взаємодіяти один з одним або з користувачами в реальному часі. Ця концепція радикально змінює традиційне сприйняття об'єктів навколишнього середовища, перетворюючи звичайні предмети на "розумні" пристрої з можливістю збору даних, аналізу та автоматизованого управління.

IoT сприяє ефективному використанню ресурсів, зокрема енергії, за рахунок інтелектуального моніторингу та управління споживанням. Це не тільки зменшує витрати, але й сприяє сталому розвитку, зменшуючи вуглецевий слід і сприяючи екологічно чистим ініціативам.

Інтелектуальні IoT-системи можуть адаптуватися до потреб і переваг користувачів, надаючи персоналізовані послуги та вдосконалюючи користувацький досвід. Це досягається за рахунок аналізу поведінки користувачів і їхніх вподобань.

У промисловості та бізнесі IoT сприяє оптимізації процесів, підвищенню продуктивності праці та зменшенню виробничих витрат. Інтелектуальні системи дозволяють автоматизувати рутинні завдання, покращуючи ефективність та скорочуючи можливість людської помилки.

Також IoT відкриває двері для нових бізнес-моделей і створює можливості для інноваційних стартапів і підприємств. Це охоплює не тільки продукти та послуги, пов'язані безпосередньо з IoT, але й широкий спектр додаткових послуг, таких як аналітика даних, безпека, технічна підтримка та багато іншого.

IoT стає ключовим елементом глобальної цифрової трансформації, з'єднуючи разом різноманітні індустрії та сфери життя. Це сприяє створенню глобальної екосистеми, де дані та служби можуть взаємодіяти на

міжнародному рівні, сприяючи глобальному економічному зростанню та інноваціям.

Але попри численні переваги, розгортання IoT також супроводжується викликами, зокрема питаннями безпеки, конфіденційності, стандартизації та сумісності. Це вимагає постійних зусиль для розробки стандартів, захисних механізмів та політик, що регулюють цю сферу.

Загалом IoT є динамічно розвиваючою технологією, що має потенціал радикально змінити наш підхід до використання технологій, впливаючи на різні аспекти нашого повсякденного життя та ділової діяльності. Водночас, для реалізації повного потенціалу IoT необхідно вирішити ряд викликів, зокрема у сферах безпеки, приватності та міжопераційної сумісності.

1.2 Огляд проблеми управління запасами товару на фірмі з багатьма філіалами по Україні

Управління запасами є ключовим аспектом для фірм з багатьма філіалами по Україні, особливо у контексті динамічного ринку та різноманітності логістичних викликів. Оптимізація запасів не тільки забезпечує ефективність роботи, але й впливає на загальну прибутковість підприємства. Ось, на мою думку, проблеми, з якими зіштовхуються такі компанії:

- непослідовність запасів між філіалами: різні філіали можуть мати нерівномірний попит на одні й ті ж товари, що ускладнює прогнозування та оптимізацію загальних запасів, що може призвести до дефіциту товарів у деяких філіалах та надлишку в інших;

- високі витрати на зберігання: надмірні запаси товарів збільшують витрати на їх зберігання; це особливо проблематично для товарів з обмеженим терміном зберігання або великими габаритами, які потребують значного складського простору;

- втрати від застарівання товарів: в динамічному ринковому середовищі товари швидко застарівають, що може призвести до збитків, особливо якщо запаси не розпродані своєчасно;

- проблеми з точністю інвентаризації: неузгодженість даних інвентаризації між різними філіалами може призвести до неправильного управління запасами, що впливає на рішення про поповнення та розподіл товарів;

- логістичні виклики: ефективне переміщення товарів між філіалами в різних регіонах України може бути складним завданням через логістичні обмеження, такі як погані дорожні умови, обмеження транспортної інфраструктури або непередбачуваність постачання;

- виклики у відповідності до нормативних вимог: фірми мають дотримуватися регулятивних вимог щодо зберігання та обігу певних товарів, що може ускладнювати управління запасами, особливо в різних регіонах з їхніми специфікаціями;

- проблеми з прогнозуванням попиту: точне прогнозування попиту є важливим для оптимізації рівнів запасів, але це може бути складно з огляду на регіональні особливості попиту, сезонність та інші фактори.

Але, найголовніша проблема – людський фактор, яка є однією з ключових проблем в управлінні запасами на фірмі, особливо тією, що має багато філіалів по всій країні. Ця проблема проявляється в кількох аспектах, що безпосередньо впливають на ефективність та надійність процесів управління запасами:

- помилки введення даних: одним із найбільш поширених проявів людського фактора є помилки при введенні даних; ручне введення інформації про товари, їх кількості та місцезнаходження може призвести до неточностей, які, у свою чергу, спричиняють надлишки або дефіцит запасів;

- відсутність відповідальності: у багатофіліальних компаніях проблема відсутності чіткої відповідальності за управління запасами може бути особливо гострою – це може призвести до ситуації, коли працівники не

відчувають особистої заангажованості в процес контролю та оптимізації запасів;

- опір змінам: впровадження нових технологій або методик управління запасами часто супроводжується опором з боку співробітників. Люди можуть опиратися змінам через брак розуміння переваг нових систем або через страх втратити роботу, що уповільнює процес впровадження інновацій і може підірвати ефективність системи управління запасами;

- недоліки у навчанні та розвитку персоналу: недостатній рівень навчання та розвитку співробітників, що займаються управлінням запасами, може призвести до неправильного використання систем та інструментів, що, в свою чергу, знижує загальну продуктивність процесу.

Розв'язання цих проблем вимагає комплексного підходу, зокрема впровадження передових технологій управління запасами, таких як системи IoT для реального моніторингу запасів, точності даних інвентаризації між філіалами, а також вдосконалення методів прогнозування попиту та оптимізації логістики.

Також потрібно робити регулярне навчання та розвиток навичок співробітників, що підвищить їхню компетентність та мотивацію правильно використовувати системи управління запасами.

Перехід до більш автоматизованих і інтегрованих систем управління запасами, разом із зосередженням на розвитку та залученні персоналу, може значно зменшити негативний вплив людського фактора на ефективність управління запасами.

1.3 Способи управління запасами товару без використання IoT

Управління запасами товару без використання IoT може включати ряд традиційних методик та підходів, які дозволяють підприємствам ефективно керувати своїми запасами, зберігати оптимальну кількість товарів та відповідати потребам клієнтів. Ось декілька ключових способів.

Система мінімальних/максимальних запасів є одним із методів управління запасами, який дуже зручно застосовувати у різних галузях, особливо коли вона інтегрована з технологіями IoT. Цей метод допомагає компаніям підтримувати оптимальний рівень запасів, забезпечуючи потреби виробництва та продажів, при цьому мінімізуючи витрати на зберігання.

У сфері IoT система мінімальних/максимальних запасів використовує інтелектуальні сенсори та пристрої для моніторингу рівнів запасів у реальному часі. Коли запаси товару досягають встановленого мінімального порогу, система автоматично ініціює процес замовлення для поповнення запасів. Цей мінімальний рівень визначається так, щоб він вистачав для задоволення поточного попиту до моменту прибуття наступної поставки. Це допомагає запобігти ситуаціям, коли товару не вистачає, та забезпечує неперервність бізнес-процесів.

З іншого боку, коли запаси досягають максимально допустимого рівня, система припиняє додаткові замовлення. Цей максимальний рівень розраховується таким чином, щоб не допустити перевантаження складських приміщень та зайвих витрат на зберігання. Він також гарантує, що у компанії завжди буде достатня кількість товару для реагування на ринкові коливання, але без надмірного накопичення, яке може привести до застарівання товарів або зайвих витрат.

Інтеграція IoT у цю систему забезпечує високий рівень точності та оперативності у виявленні та реагуванні на зміни в рівнях запасів. Сенсори та інтелектуальні пристрої можуть неперервно відстежувати стан запасів, надсилати дані в центральну систему управління, де вони аналізуються для виявлення потреби у замовленнях чи припинення їх. Це не тільки автоматизує процеси управління запасами, але й зменшує можливість людської помилки, покращуючи ефективність та знижуючи витрати.

Таким чином, система мінімальних/максимальних запасів, підсилена можливостями IoT, дозволяє компаніям підтримувати баланс між потребою в

матеріалах та витратами на їх зберігання, що є ключовим для забезпечення сталого та прибуткового бізнесу.

Методика "точно вчасно" (Just-In-Time, JIT) є стратегією управління запасами, яка зосереджена на максимальному зниженні витрат шляхом зменшення обсягів запасів на складах. Основна ідея JIT полягає у виробництві або закупівлі товарів тільки тоді, коли це дійсно необхідно для виконання конкретних замовлень або продовження виробничих процесів. Це дозволяє компаніям знизити витрати на зберігання та управління запасами, а також зменшити ризики пов'язані з застаріванням товарів.

Інтеграція технологій IoT у систему JIT може значно підвищити її ефективність. Завдяки IoT, управління ланцюгом поставок стає більш прозорим і оперативним. Сенсори та інтелектуальні пристрої можуть відстежувати стан виробництва, рівні запасів та потреби в матеріалах у реальному часі. Ці дані використовуються для точного планування закупівель та виробництва, що забезпечує постачання необхідних матеріалів "точно вчасно".

Ключовим моментом у реалізації JIT з використанням IoT є можливість швидко реагувати на зміни в попиті та умовах поставок. Системи можуть автоматично коригувати виробничі плани та замовлення на постачання в залежності від поточної ситуації. Наприклад, якщо IoT-сенсори виявляють затримку в доставці важливих компонентів, система може відразу ініціювати альтернативні дії, щоб запобігти зупинці виробничих ліній.

Крім того, IoT допомагає збирати великі обсяги даних про процеси виробництва та споживання, які можуть бути аналізовані для подальшого оптимізування. Це включає в себе аналіз ефективності постачальників, часу доставки, якості матеріалів та продукції. Завдяки цьому можливе не тільки поліпшення надійності ланцюга поставок, але й загальне підвищення якості продукції.

Застосування IoT в контексті JIT також сприяє створенню більш гнучких виробничих систем. Компанії можуть легше адаптуватися до змін у

ринкових умовах, швидко змінюючи виробничі потужності або асортимент продукції залежно від актуальних потреб. Таким чином, використання технологій IoT у рамках ЛІТ не просто спрощує управління запасами, але й значно розширює можливості підприємств у відповідності до змінюваного бізнес-середовища.

Періодичний перегляд запасів — це метод управління запасами, що полягає в регулярному оцінюванні кількості та стану товарів на складі з фіксованою частотою, наприклад, щотижня, щомісяця або щокварталу. Відмінність цього підходу від інших методів управління запасами полягає у тому, що рішення про замовлення нових товарів приймається відповідно до графіка, а не на основі досягнення певного рівня запасів.

Інтеграція технологій IoT в періодичний перегляд запасів може суттєво підвищити ефективність цього процесу. IoT дозволяє автоматизувати збір даних про запаси, що забезпечує точність і актуальність інформації без необхідності ручного втручання. Сенсори і пристрої IoT можуть відстежувати кількість і стан товарів у реальному часі, відправляючи дані в центральну систему управління.

Під час періодичного перегляду ця інформація використовується для оцінки поточних запасів та планування необхідних поповнень. Наприклад, якщо під час місячного перегляду виявляється, що запаси певного товару нижчі за прогнозований попит, система може автоматично ініціювати замовлення, щоб поповнити запаси до наступного перегляду. Це допомагає підтримувати достатній рівень товарів, мінімізувати ризики відсутності на складі та забезпечувати безперервність поставок.

Крім того, використання IoT у періодичному перегляді запасів дозволяє краще розуміти та аналізувати патерни споживання і зміни у попиті. Ці дані можуть бути використані для оптимізації графіка переглядів та кількості замовлень, а також для більш точного прогнозування потреб у майбутньому.

Застосування IoT може також підвищити гнучкість управління запасами. Наприклад, у випадку змін у попиті або поставках, система може

швидко виявити ці зміни та відповідно адаптувати розміри замовлень. Це забезпечує, що компанія залишається конкурентоспроможною та здатною оперативно реагувати на змінні умови ринку.

Таким чином, впровадження IoT у періодичний перегляд запасів не тільки спрощує цей процес, але й робить його більш точним і ефективним, забезпечуючи оптимальне управління ресурсами компанії.

Метод ABC аналізу — це популярний підхід в управлінні запасами, який допомагає оптимізувати ресурси шляхом категоризації товарів на основі їх значущості та впливу на загальні витрати компанії. Згідно з цим методом, товари розділяються на три категорії:

- категорія А: найважливіші товари, які складають невелику частку асортименту, але приносять значну частку доходу або загальних витрат компанії;
- категорія В: товари середньої важливості, які становлять помірну частку асортименту та витрат;
- категорія С: найменш значущі товари, які складають велику частку асортименту, але приносять найменшу частку доходу.

Інтеграція технологій IoT може значно підвищити ефективність ABC аналізу, зокрема через автоматизацію збору даних і точне відстеження поведінки товарів на складі та у процесі продажу. Сенсори та інтелектуальні пристрої IoT дозволяють отримувати актуальні дані про стан запасів, частоту замовлень, швидкість обертання товарів, а також про попит і вподобання клієнтів. Ці дані автоматично аналізуються для визначення категорії кожного товару відповідно до принципів ABC аналізу.

На практиці використання IoT у методі ABC аналізу дозволяє компаніям не лише точно категоризувати товари, але й оперативно реагувати на зміни у попиті та відстежувати вплив цих змін на загальну стратегію управління запасами. Наприклад, якщо товар з категорії С починає демонструвати зростання продажів і стає більш значущим для компанії,

система може автоматично перекласифікувати його до категорії В або А, в залежності від вкладу у доходи.

Крім того, інтеграція IoT з ABC аналізом може сприяти більш точному плануванню закупівель та виробництва. Наприклад, товари категорії А, які вимагають постійного моніторингу і швидкого реагування на зміни у попиті, можуть бути оснащені додатковими сенсорами для неперервного відстеження їх стану та рівнів запасів.

Таким чином, використання IoT у рамках методу ABC аналізу допомагає компаніям не тільки оптимізувати управління запасами, але й підвищити ефективність використання ресурсів, знизити витрати та підвищити задоволеність клієнтів за рахунок забезпечення високої наявності та якості товарів.

Використання програмного забезпечення для управління запасами є ключовим елементом сучасних стратегій логістики та збуту, особливо у контексті інтеграції з технологіями IoT. Це програмне забезпечення дозволяє компаніям ефективно відслідковувати, прогнозувати і управляти запасами в режимі реального часу, значно покращуючи операційну ефективність і знижуючи витрати.

Програмне забезпечення для управління запасами, інтегроване з IoT, забезпечує автоматизацію збору даних про стан товарів на складах, у транспортних засобах і навіть у руках споживачів. Сенсори, вбудовані в упаковку або на складах, можуть вимірювати різні показники, такі як температура, вологість, рівень запасів, швидкість руху товарів і навіть їх геолокацію. Вся ця інформація відправляється до центральної системи управління в реальному часі.

Це дозволяє менеджерам з логістики та запасами отримувати глибокий аналіз стану інвентаря, оптимізувати рівні запасів та зменшити витрати на зберігання. Завдяки постійному моніторингу, програмне забезпечення може автоматично попереджати менеджерів про необхідність дозамовлення товарів або про ризики застарівання запасів.

Крім того, програмне забезпечення для управління запасами з IoT інтеграцією може автоматизувати багато рутинних операцій, зокрема, замовлення товарів і розрахунків оптимальних маршрутів доставки, враховуючи час доставки, вартість та наявність товарів у конкретних локаціях. Це значно зменшує ймовірність людської помилки і підвищує загальну продуктивність ланцюга постачань.

Програмне забезпечення для управління запасами також сприяє підвищенню задоволеності клієнтів, забезпечуючи більш точне виконання замовлень і своєчасну доставку товарів. Використання передових аналітичних інструментів дозволяє фірмам краще розуміти патерни попиту і адаптуватись до змін у споживацьких вподобаннях, що є особливо важливим у швидко змінюваних ринкових умовах.

Таким чином, програмне забезпечення для управління запасами, інтегроване з IoT, стає необхідним інструментом у сучасному світі регулювання кількості товару, забезпечуючи компаніям потужний інструмент для зниження витрат, підвищення ефективності, зменшення ризиків і покращення сервісу.

Кожен з цих методів має свої переваги та обмеження та може бути ефективним у залежності від специфіки бізнесу, обсягу запасів та цілей управління.

1.4 Впровадження IoT

Впровадження IoT систем для відслідковування та регулювання запасів може кардинально змінити підхід до управління запасами в компаніях з багатьма філіалами по Україні. IoT технології дозволяють автоматизувати збір даних про стан запасів, їх рух та споживання в реальному часі, забезпечуючи точність та оперативність інформації, необхідної для ефективного управління.

Основні переваги IoT систем для управління запасами:

- автоматизація відслідковування запасів: IoT девайси, такі як RFID (радіочастотна ідентифікація) теги та сенсори, дозволяють автоматично відслідковувати переміщення товарів всередині філіалів та між ними, зменшуючи потребу в ручному введенні даних та знижуючи ймовірність помилок;
- оперативне реагування на зміни запасів: завдяки реальному часу моніторингу, компанія може швидко реагувати на зміни в запасах, оптимізувати рівні запасів та забезпечити своєчасне поповнення;
- оптимізація логістики та зниження витрат: IoT допомагає аналізувати та оптимізувати логістичні маршрути та складські операції, знижуючи витрати на зберігання та транспортування товарів;
- покращення задоволеності клієнтів: покращення задоволеності клієнтів полягає у глибокому розумінні їхніх потреб та очікувань. Важливо постійно збирати зворотний зв'язок, використовуючи різні методи, такі як опитування та аналіз даних. Якість продукції та послуг повинна бути на найвищому рівні, а також слід впроваджувати інновації та вдосконалення на основі зворотного зв'язку. Швидке та ефективне вирішення проблем клієнтів є критично важливим. Процес подання скарг повинен бути зручним та прозорим, а вирішення проблем – оперативним. Комунікація з клієнтами повинна бути багатоканальною, включаючи телефон, електронну пошту, чат та соціальні мережі. Регулярний контакт через інформаційні розсилки та новини допомагає підтримувати зв'язок;
- прогнозування та аналітика: збір великих обсягів даних дозволяє застосовувати аналітичні та прогнозні інструменти для вдосконалення стратегії управління запасами, планування попиту та оптимізації асортименту;
- інтеграція з іншими системами: IoT системи можуть бути інтегровані з ERP, CRM та іншими інформаційними системами, створюючи єдину

автоматизовану екосистему для управління всіма аспектами діяльності компанії.

Впровадження IoT системи вимагає комплексного підходу, який включає вибір відповідного обладнання (сенсори, RFID теги, читачі тощо), налаштування зв'язку між пристроями, розробку або адаптацію програмного забезпечення для збору, обробки та аналізу даних, а також інтеграцію з існуючими бізнес-процесами та системами.

1.5 Загальна структура IoT регулювання кількості товару

Структура IoT для регулювання кількості товару охоплює використання сенсорів та RFID тегів, які допомагають у ідентифікації та відслідковуванні товарів, а також RFID зчитувачів для збору даних з цих тегів та сенсорів. Дані, зібрані зчитувачами, передаються через шлюзи та мережеве обладнання до хмарної інфраструктури, де вони зберігаються та обробляються. На основі цих даних аналітичні та управлінські інструменти дозволяють аналізувати інформацію та приймати рішення, які відображаються користувачам через додатки та вебпортали, забезпечуючи ефективне моніторинг та управління запасами.

1.6 Висновки до розділу 1

Загалом використання IoT для регулювання кількості товару на фірмі з багатьма філіалами по Україні є досить закономірним, сучасним та ефективним рішенням. За допомогою цієї системи зменшується велика кількість ризиків та водночас підвищується ефективність управління запасами, що веде до оптимізації витрат і збільшення прибутковості. Ця система дозволяє компанії швидше реагувати на зміни в попиті та пропозиції, а також підвищує задоволеність клієнтів завдяки забезпеченню неперервності постачання та зменшенню ймовірності дефіциту товарів. Крім того,

автоматизація процесів та зниження потреби в ручній перевірці запасів зменшують можливість помилок і підвищують точність управління запасами.

Отже, інтеграція IoT в управління запасами стає ключовим фактором для підвищення конкурентоспроможності на ринку, дозволяючи компаніям не лише оптимізувати свої внутрішні процеси, але й надавати високий рівень обслуговування своїм клієнтам.

2 ДОСЛІДЖЕННЯ КОМП'ЮТЕРНО-ІНТЕГРОВАНИХ ТЕХНОЛОГІЙ ДЛЯ РЕАЛІЗАЦІЇ ЗАВДАННЯ

2.1 Вибір апаратних рішень

Для ідентифікації та відслідковування товарів на складі та в торговельному залі використовують RFID Теги.

RFID теги (або мітки) – це критичний компонент систем RFID, які використовуються для зберігання інформації, що може бути зчитана зчитувачами на різній відстані. Існує кілька різних видів RFID тегів, кожен з яких має свої особливості та призначення.

Основні види включають: активні RFID теги, пасивні RFID теги, напівпасивні RFID теги, RFID теги з датчиками, частотні RFID теги.

Розглянемо активні RFID (Radio-Frequency Identification) теги – це електронні пристрої, які містять мікросхему та антенну для передачі та отримання радіосигналів. Вони відрізняються від пасивних RFID тегів тим, що мають вбудований джерело живлення (батарею), що дозволяє їм передавати сигнали на значну відстань і в реальному часі. Активні RFID теги зазвичай використовуються для відстеження та управління запасами, особливо коли необхідно відслідковувати місцезнаходження предметів у реальному часі.

Завдяки своїм можливостям, активні RFID теги забезпечують велику точність та ефективність у відстеженні товарів або обладнання. Вони можуть бути використані у різних сферах, включаючи логістику, транспорт, виробництво та роздрібну торгівлю. Наприклад, виробники можуть використовувати активні RFID теги для відстеження руху матеріалів у виробничому процесі, а роздрібні магазини можуть використовувати їх для відстеження товарів на полицях та управління запасами.

Однією з основних переваг активних RFID тегів є їхня здатність працювати на великих відстанях – до кількох десятків метрів. Це робить їх ідеальними для відстеження великих об'єктів або великої кількості

предметів одночасно. Крім того, активні RFID теги можуть бути використані у важкодоступних місцях або в умовах, де не можуть бути застосовані інші методи відстеження, наприклад, у зоні високої вологості чи температури.

Однак варто враховувати, що активні RFID теги мають деякі обмеження. Вони потребують джерела живлення, тому їх треба періодично заряджати або замінювати батареї. Крім того, їх використання може бути витратним через високу вартість самого тега та необхідність обслуговування батарей.

У загальному, активні RFID теги є потужним інструментом для відстеження та управління запасами у реальному часі. Вони дозволяють компаніям підвищити ефективність своїх операцій та забезпечити високу точність в управлінні запасами.

Активний RFID тег наведено на рисунку 2.1.



Рисунок 2.1 – Активний RFID тег [10]

Пасивні RFID (Radio-Frequency Identification) теги — це електронні пристрої, які містять мікросхему та антенну для передачі та отримання радіосигналів. Вони відрізняються від активних RFID тегів тим, що не мають вбудованого джерела живлення (батареї) і отримують енергію для роботи від сигналу, який надсилається читачем.

Пасивні RFID теги можуть бути використані для відстеження та ідентифікації предметів безпосередньої близькості до читача. Вони зазвичай використовуються для відстеження товарів у логістиці, управлінні запасами, контролі доступу та ідентифікації транспортних засобів. Пасивні RFID теги можуть бути дуже маленькими і дешевими у виробництві, що робить їх ідеальними для масового застосування.

Однією з головних переваг пасивних RFID тегів є їхня невелика вартість виробництва і простота в експлуатації. Вони можуть бути вбудовані у вироби або прикріплені до них, дозволяючи легко відстежувати їх рухи і місцезнаходження. Також вони не вимагають обслуговування через відсутність необхідності заміни батарей.

Однак пасивні RFID теги мають свої обмеження. Оскільки вони отримують енергію для роботи від сигналу читача, їх зчитування може бути обмежене відстанню і швидкістю зчитування. Також, їхні можливості зберігання інформації обмежені порівняно з активними RFID тегами, що може бути проблемою для застосувань, де необхідно зберігати великі обсяги даних.

У загальному, пасивні RFID теги є ефективним та економічним засобом відстеження та ідентифікації предметів у багатьох сферах діяльності. Вони дозволяють підприємствам підвищити ефективність управління запасами, контролювати доступ та забезпечувати безпеку, що робить їх популярними засобами автоматизації процесів.

Пасивний RFID тег наведено на рисунку 2.2.

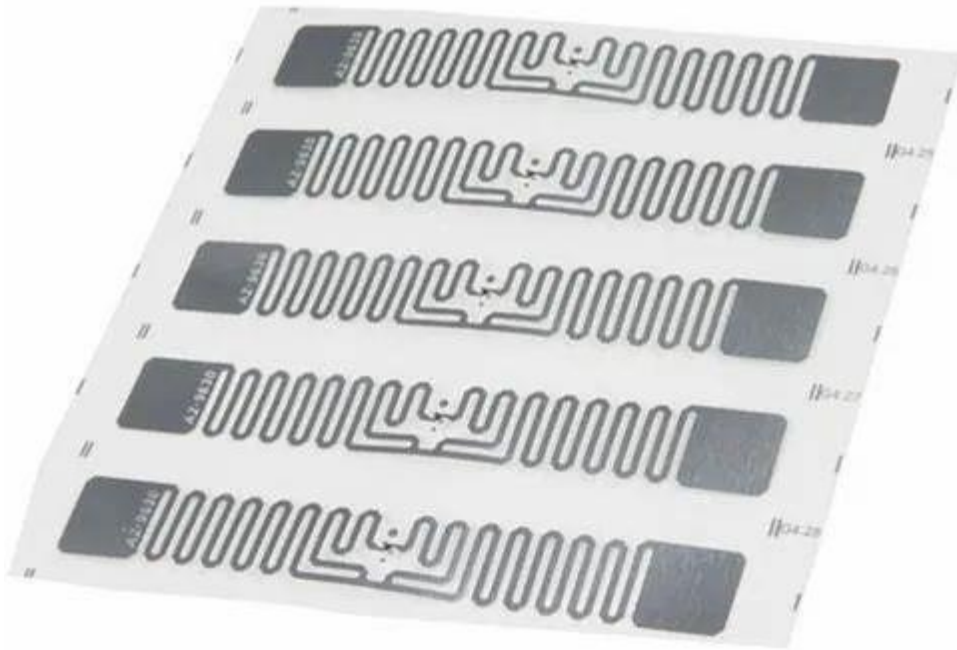


Рисунок 2.2 – Пасивний RFID тег [11]

Напівпасивні RFID теги, також відомі як батарейно-активні, поєднують у собі функціональність пасивних і активних тегів. Вони мають вбудовану батарею, яка живить лише деякі функції тега, такі як передача сигналу або збереження даних, в той час як для інших функцій вони використовують енергію від радіосигналу, який передається від читача.

Основна перевага напівпасивних RFID тегів полягає у поєднанні високої ефективності з невеликими розмірами. Вони можуть працювати на значній відстані від читача і забезпечувати високу швидкість зчитування, що робить їх ідеальними для використання в умовах, коли потрібно відстежувати рухомі об'єкти або об'єкти у важкодоступних місцях.

Однак, напівпасивні RFID теги мають свої обмеження. Вони потребують регулярного заряджання або заміни батареї, що може бути проблематичним у випадку великого обсягу тегів або у випадку, коли теги розміщені у важкодоступних місцях. Також, їхня вартість виробництва може бути вищою порівняно з пасивними RFID тегами.

У цілому, напівпасивні RFID теги є потужним інструментом для відстеження та ідентифікації предметів, особливо в умовах, коли потрібна

висока швидкість та точність зчитування. Вони знаходять широке застосування у логістиці, транспорті, виробництві та інших галузях, де потрібно ефективно відстежувати рухомі об'єкти або об'єкти у складних умовах.

RFID теги з датчиками – це RFID теги, які додатково оснащені різними типами датчиків для збору додаткової інформації про навколишнє середовище або сам тег. Ці датчики можуть вимірювати такі параметри, як температура, вологість, освітленість, тиск, прискорення та інші.

RFID теги з датчиками можуть бути використані для різних цілей в залежності від типу датчиків і умов застосування. Наприклад, вони можуть бути використані для відстеження та моніторингу умов зберігання продуктів, які вимагають певного рівня температури і вологості. Також вони можуть бути використані для моніторингу стану обладнання або активів, щоб вчасно виявляти ознаки пошкодження або несправності.

Однією з основних переваг RFID тегів з датчиками є можливість отримувати реальні дані про стан об'єктів або умови навколишнього середовища безпосередньо на місці. Це дозволяє оперативно реагувати на зміни умов і вчасно вживати заходів для попередження негативних наслідків.

Однак важливо враховувати, що використання RFID тегів з датчиками може мати обмеження, такі як обмежений час роботи в разі використання батарейного живлення та обмежена функціональність датчиків в порівнянні з професійними датчиками. Також, вартість цих тегів може бути вищою порівняно зі звичайними RFID тегами через включення додаткових датчиків.

У цілому, RFID теги з датчиками є цінним інструментом для відстеження та моніторингу умов зберігання продуктів, стану обладнання та інших важливих параметрів. Вони дозволяють підприємствам ефективно контролювати умови їхньої діяльності та вчасно реагувати на зміни, що сприяє підвищенню ефективності та якості управління.

RFID теги з датчиком температури наведено на рисунку 2.3.



Рисунок 2.3 – RFID теги з датчиком температури [12]

Частотні категорії RFID тегів можна класифікувати за їхньою робочою частотою на різні категорії, які визначають їхні можливості та область застосування. Основні частотні категорії RFID тегів включають низькочастотні (low frequency tag, LFT), високочастотні (high frequency tag, HF), ультрависокочастотні (ultra high frequency tag, UHF) та мікрохвильові (microwave tags, MW) теги.

- низькочастотні (LF) RFID теги: ці теги працюють на частоті близько 125 кГц. Вони мають досить невеликий діапазон зчитування (зазвичай до 10 см) та низьку швидкість передачі даних. LF теги використовуються, наприклад, у системах контролю доступу, та відстеження домашніх тварин;

- високочастотні (HF) RFID теги: ці теги працюють на частоті близько 13,56 МГц. Вони мають деяке зростання в діапазоні зчитування (до 1 метра) та можуть передавати дані з високою швидкістю. HF теги

застосовуються у системах безконтактної оплати (наприклад, у картках оплати) та в системах відстеження товарів у роздрібній торгівлі;

- ультрависокочастотні (UHF) RFID теги: ці теги працюють на частоті близько 860-960 МГц. Вони мають великий діапазон зчитування (до 15 метрів або більше) та високу швидкість передачі даних. UHF теги широко використовуються у логістиці та управлінні запасами, оскільки дозволяють швидко відстежувати великі об'єми товарів;

- мікрохвильові (MW) RFID теги: ці теги працюють на високих частотах, близьких до 2,45 ГГц. Вони мають дуже великий діапазон зчитування (до 100 метрів або більше) та високу швидкість передачі даних. MW теги використовуються у складних системах відстеження, наприклад, у промисловості для відстеження великих об'єктів або транспортних засобів.

Кожна з цих частотних категорій має свої переваги та недоліки, і вибір певної категорії залежить від конкретних вимог до системи RFID та її застосування.

У нашій системі ми будемо використовувати пасивні RFID теги.

Радіочастотний RFID тег наведено на рисунку 2.4.



Рисунок 2.4 – Радіочастотний RFID тег [13]

Порівняння RFID тегів наведено у порівняльній таблиці 2.1.

Таблиця 2.1 – Порівнювальна таблиця RFID тегів

Характеристика	Активні RFID теги	Пасивні RFID теги	Напівпасивні RFID теги (батареино-активні)	RFID теги з датчиками	Частотні RFID теги
Живлення	Вбудована батарея	Живлення від сигналу	Джерело живлення для деяких функцій	Вбудована батарея	Різні частоти від LF до MW
Діапазон зчитування	До 100 метрів або більше	До 10 метрів	До 100 метрів або більше	Залежить від типу датчиків	Залежить від частоти
Швидкість передачі даних	Висока	Низька	Висока	Залежить від типу датчиків	Залежить від частоти
Вартість	Висока	Низька	Середня	Висока	Варіює від низької до високої
Використання	Для відстеження об'єктів на великій відстані	Для відстеження об'єктів на невеликій відстані	Для відстеження об'єктів на великій відстані з додатковими можливостями	Для відстеження об'єктів зі збором додаткової інформації	Для відстеження об'єктів на певній частоті

Сенсори: температури, вологості, ваги та інші види сенсорів, що використовуються для моніторингу стану товарів.

Температурні сенсори вимірюють і контролюють температуру навколишнього середовища або самого товару, що є особливо важливим для

зберігання продуктів, які можуть псуватися при невідповідних температурних умовах.

Сенсори вологості контролюють рівень вологи у повітрі, що також важливо для збереження якості багатьох товарів, таких як харчові продукти, паперові вироби або електроніка, яка може бути чутливою до вологи.

Вагові сенсори використовуються для визначення маси товарів, що допомагає точно обліковувати кількість товарів на складі та контролювати процеси завантаження та розвантаження. Використання цих сенсорів у системі управління запасами дозволяє автоматично збирати дані про стан товарів у режимі реального часу. Ці дані можуть бути передані на центральний сервер або у хмару для подальшого аналізу і прийняття рішень. Інформація, зібрана сенсорами, допомагає запобігати псуванню товарів, знижує ризики втрат та забезпечує відповідність умов зберігання до встановлених стандартів.

Інтеграція сенсорів із системами IoT дозволяє створити комплексну систему моніторингу, яка забезпечує високий рівень контролю за станом товарів і допомагає оптимізувати логістичні процеси. Системи на основі сенсорів можуть також генерувати автоматичні сповіщення у разі, якщо виявлено відхилення від заданих параметрів, що дозволяє швидко реагувати на потенційні проблеми і вживати необхідних заходів для їх усунення.

Порівняння сенсорів наведено у порівняльній таблиці 2.2.

Таблиця 2.2 – Порівнювальна таблиця сенсорів

Тип сенсора	Призначення	Переваги	Недоліки	Приклади використання
Температурні сенсори	Вимірювання і контроль температури	Висока точність, швидкий відгук	Потребують регулярної калібровки	Зберігання харчових продуктів, ліків
Сенсори вологості	Вимірювання рівня вологості	Важливі для чутливих до вологості товарів	Можуть бути чутливі до пилу і бруду	Зберігання паперових виробів, електроніки
Вагові сенсори	Вимірювання маси товарів	Точний облік кількості товарів	Потребують стабільної поверхні	Контроль завантаження та розвантаження товарів

RFID-зчитувачі: для швидкої інвентаризації та обліку товарів використовується RFID технологія для зчитування та ідентифікації інформації, яка зберігається на RFID-мітках або транспондерах. Існує кілька різних видів RFID зчитувачів, які можуть відрізнятися за декількома характеристиками, включаючи діапазон дії, частоту, і конструкцію. Ось основні типи RFID зчитувачів:

- фіксовані RFID зчитувачі;
- мобільні RFID зчитувачі;
- зчитувачі, які вбудовані в інші системи;
- RFID зчитувачі з високою продуктивністю.

Фіксовані RFID зчитувачі: їх зазвичай встановлюють на стаціонарних точках, наприклад, на входах або виходах складів, виробничих ліній, або паркових зон. Вони можуть зчитувати мітки на різних відстанях, залежно від їхньої конструкції та потужності.

Фіксований RFID зчитувач наведено на рисунку 2.5.



Рисунок 2.5 – Фіксований RFID зчитувач [14]

Мобільні RFID зчитувачі: ці зчитувачі портативні та часто інтегровані з іншими пристроями, наприклад, з кишеньковими комп'ютерами або смартфонами. Вони зручні для інвентаризації та використовуються в роздрібній торгівлі, логістиці, або контролі за активами.

Мобільний RFID зчитувач наведено на рисунку 2.6.



Рисунок 2.6 – Мобільний RFID зчитувач [15]

Зчитувачі, які вбудовані в інші системи: такі зчитувачі інтегровані безпосередньо в інші технічні системи, як-от турнікети, контрольні точки доступу або платіжні системи.

Зчитувачі, які вбудовані в інші системи наведено на рисунку 2.7.



Рисунок 2.7 – Зчитувачі, які вбудовані в інші системи [16]

RFID зчитувачі з високою продуктивністю: ці пристрої призначені для дуже швидкого зчитування великої кількості міток в складних умовах, наприклад, в логістичних центрах або на виробничих лініях.

RFID зчитувачі також можуть класифікуватися за типом частоти, яку вони використовують:

- низькочастотні (LF) зчитувачі (125 або 134.2 кГц) - мають короткий радіус дії та нижчу швидкість передачі даних;

- високочастотні (HF) зчитувачі (13.56 МГц) - використовуються для більшості платіжних та ідентифікаційних систем;
- ультрависокочастотні (UHF) зчитувачі (860 до 960 МГц) - Забезпечують довгий радіус дії та високу швидкість передачі даних;
- мікрохвильові зчитувачі (2.45 ГГц або вище) - часто використовуються в автоматизованих дорожніх системах і можуть зчитувати мітки на великій відстані.

RFID зчитувач з високою продуктивністю наведено на рисунку 2.8.



Рисунок 2.8 – RFID зчитувач з високою продуктивністю (цей дальної дії) [17]

Кожен тип зчитувача підходить для певних умов використання та специфічних завдань. Вибір зчитувача залежить від багатьох факторів, включаючи відстань зчитування, швидкість обробки даних, стійкість до зовнішніх умов і т.д.

Якщо провести аналіз у сфері продажу автозапчастин, то для такої сфери особливо підходять мобільні RFID зчитувачі для працівників складу, оскільки вони забезпечують гнучкість та зручність при обліку та відстеженні великої кількості деталей. Мобільні RFID зчитувачі дозволяють працівникам швидко і точно зчитувати інформацію про запчастини безпосередньо на місці їх зберігання, що значно підвищує ефективність роботи складу. Вони також спрощують процес інвентаризації, знижуючи ймовірність помилок.

Таким чином, мобільні RFID зчитувачі забезпечують значні переваги у сфері продажу автозапчастин, роблячи процес більш автоматизованим, точним і ефективним. Це не тільки покращує обслуговування клієнтів, але й сприяє кращому управлінню загальними ресурсами компанії.

У нашій системі ми будемо використовувати мобільні RFID зчитувачі.

Порівняння RFID зчитувачів наведено у порівняльній таблиці 2.3.

Таблиця 2.3 – Порівнювальна таблиця RFID зчитувачів

Особливість / Тип зчитувача	Фіксовані зчитувачі	Мобільні зчитувачі	Зчитувачі, вбудовані в інші системи	RFID зчитувачі з високою продуктивністю
1	2	3	4	5
Мобільність	Не мобільні, встановлюються на одному місці	Висока мобільність, переносні	Зазвичай немобільні, вбудовані в обладнання	В залежності від конкретної моделі, може бути різною
Зручність використання	Вимагає мінімального втручання після установки	Вимагає ручного управління	Автоматична робота в рамках системи	Зазвичай потребує мінімального втручання оператора
Діапазон зчитування	Зазвичай більший, оскільки розрахований на ширші зони зчитування	Обмежений до дистанції, яку може подолати оператор	Залежить від конкретної системи, може бути як широким, так і вузьким	Великий, може зчитувати багато тегів одночасно на великій відстані
Вартість	Вища через потужніші компоненти та потребу в постійній установці	Зазвичай нижча порівняно з фіксованими	Вартість інтегрована в загальну ціну системи	Зазвичай вища, оскільки призначені для вимогливих застосувань
Гнучкість у застосуванні	Обмежена конкретними локаціями	Висока гнучкість, може бути використаний у різних локаціях	Обмежена специфікою системи, в яку вбудовані	Висока, може працювати у вимогливих умовах

Продовження таблиці 2.3

1	2	3	4	5
Автоматизація	Висока, ідеально підходить для автоматичног о відстеження	Обмежена можливостями оператора	Висока, повністю інтегрована в систему	Дуже висока, спрощує процес відстеження

2.2 Вибір технології передачі даних

Для передачі даних від IoT пристроїв до мережі використовуються Wi-Fi, Bluetooth, Zigbee, та інші бездротові технології.

У сфері IoT, передача даних від пристроїв до мережі є важливим аспектом для оптимізації процесів, збору даних та управління системами. Бездротові технології, такі як Wi-Fi, Bluetooth, Zigbee та інші, відіграють важливу роль у цьому процесі, оскільки вони дозволяють забезпечити швидкий і надійний обмін інформацією між пристроями і серверами чи хмарними системами.

Wi-Fi є однією з найпоширеніших бездротових технологій, використовуваних у домашніх та комерційних IoT системах завдяки своїй здатності підтримувати високу швидкість передачі даних та велику покриття діапазону. Це особливо корисно для пристроїв, які вимагають пересилання великих обсягів даних, наприклад, відеонагляду або передачі мультимедійних файлів.

Bluetooth, з іншого боку, ідеально підходить для створення малогабаритних мереж, де не потрібна велика дистанція зв'язку або висока швидкість передачі даних. Це особливо корисно для пристроїв, які збирають дані про здоров'я, носимих пристроїв або інших портативних пристроїв, де енергозбереження є критичним фактором.

Zigbee, який використовує низьку потужність для передачі даних на короткі відстані, є ідеальним для створення щільних бездротових мереж сенсорів і актуаторів. Його можна часто знайти у домашніх автоматизаційних системах, де потрібна висока надійність та здатність до самоорганізації мережі. Завдяки своїй надійності та масштабованості Zigbee добре підходить для розумних будинків, де десятки або навіть сотні пристроїв повинні безперервно взаємодіяти між собою.

Крім цих технологій, існують і інші варіанти бездротового зв'язку, такі як LoRaWAN, що забезпечує довгі відстані передачі даних при низькому енергоспоживанні, що робить його ідеальним для застосувань у смарт-сіті або в промислових IoT рішеннях.

Використання цих бездротових технологій у сфері IoT значно полегшує управління запасами та логістику, оскільки компанії можуть в реальному часі відстежувати розташування та стан своїх товарів. Це дозволяє оптимізувати процеси зберігання і доставки, зменшити витрати на зберігання, підвищити ефективність робочих процесів та забезпечити більшу задоволеність клієнтів.

Порівняння технологій передачі даних наведено у таблиці 2.4.

Таблиця 2.4 – Порівняльна таблиця технології передачі даних

Технологія	Діапазон передачі	Швидкість передачі	Енергоспоживання	Основне використання
1	2	3	4	5
Wi-Fi	До 100 метрів	Висока (до 600 Mbps і більше)	Високе	Передача великих обсягів даних, відеонагляд, домашній інтернет

Продовження таблиці 2.4

1	2	3	4	5
Bluetooth	До 100 метрів (Bluetooth 5.0)	Від середньої до високої (до 2 Mbps)	Низьке	Передача даних між портативними пристроями, носимі пристрої
Zigbee	До 100 метрів (Bluetooth 5.0)	Низька (до 250 Kbps)	Дуже низьке	Домашня автоматизація, бездротові сенсорні мережі
LoRaWAN	До 100 метрів (Bluetooth 5.0)	Дуже низька (до 50 Kbps)	Дуже низьке	Домашня автоматизація, бездротові сенсорні мережі

2.3 Вибір хмарної платформи

Хмарні платформи, такі як AWS IoT, Google Cloud IoT та Microsoft Azure IoT, відіграють вирішальну роль у регулюванні кількості товару в сфері Інтернету речей. Вони надають необхідні інструменти та інфраструктуру для збору, обробки, зберігання та аналізу великих обсягів даних з IoT пристроїв. Ці платформи дозволяють компаніям легко масштабувати свої IoT рішення, забезпечуючи глобальне покриття та високий рівень безпеки, що є критично важливим для управління сучасними логістичними ланцюгами.

AWS IoT надає широкий спектр сервісів, які включають в себе керування пристроями, безпеку та аналітику. Це дозволяє компаніям відстежувати стан товарів в реальному часі, передбачати потреби у запасах та

оптимізувати логістичні потоки. Наприклад, інтеграція з Alexa дозволяє використовувати голосові команди для управління ланцюгами постачання.

Google Cloud IoT, з іншого боку, пропонує могутній набір інструментів для аналізу та машинного навчання, які можуть бути використані для прогнозування трендів споживання та автоматизації замовлень запасів. Інтеграція з іншими сервісами Google, такими як BigQuery та Google Analytics, дозволяє з легкістю аналізувати великі обсяги даних і отримувати глибокі інсайти про поведінку споживачів та ефективність ланцюгів постачання.

Microsoft Azure IoT пропонує рішення, що включають розширену аналітику, машинне навчання та штучний інтелект. Ці інструменти допомагають компаніям не тільки стежити за запасами, але й оптимізувати робочі процеси та знизити витрати на обслуговування. Azure також пропонує спеціалізовані рішення для різних галузей, дозволяючи більш точно налаштовувати IoT системи під конкретні потреби компаній.

Користуючись цими хмарними платформами, компанії можуть значно покращити ефективність управління запасами. Вони можуть автоматизувати велику частину роботи з регулювання запасів, зменшуючи людські помилки та забезпечуючи вищу прозорість усіх процесів. Завдяки можливостям швидкої обробки даних і використанню передових технологій, хмарні платформи змінюють спосіб управління товарними запасами в глобальному масштабі, допомагаючи компаніям стати більш гнучкими та реактивними на зміни ринкових умов.

При розробці нашої системи ми використовували хмарну платформу AWS IoT, бо вона найефективніша для підключення пристроїв, збору, обробки та збереження інформації.

Порівняння платформ даних та аналітики наведено у таблиці 2.5.

Таблиця 2.5 Порівняльна таблиця Платформа даних та аналітики

Параметр	AWS IoT	Google Cloud IoT	Microsoft Azure IoT
Особливості	Підтримка множини протоколів MQTT, HTTP, WebSockets	Сильні інтеграції з іншими Google Cloud сервісами	Широкий спектр інтегрованих інструментів та сервісів
Аналітика	IoT Analytics для обробки даних	Сильні інтеграції з іншими Google Cloud сервісами	Azure Stream Analytics, Machine Learning
Безпека	Розширені можливості управління аутентифікації пристроїв	Сильна безпека даних та приватність, включно з End-to-end encryption	Особливий наголос на безпеці та приватності, використання Azure Security Center
Масштабованість	Висока масштабованість для мільйонів пристроїв	Дуже добре підходить для великих масштабів завдяки інфраструктурі Google	Особливий наголос на безпеці та приватності, використання Azure Security Center
Інтеграція	Глибокі інтеграції з іншими AWS сервісами	Легка інтеграція з іншими сервісами Google	Сильна інтеграція з іншими Azure сервісами та іншими Microsoft продуктами
Ціноутворення	Гнучке ціноутворення залежно від використання	Конкурентні ціни, особливо для великих обсягів даних	Гнучкі опції ціноутворення з можливістю кастомізації

2.4 Проектування інтерфейсу користувача

Для доступу до інформації про запаси, моніторингу стану складу, та управління замовленнями в реальному часі використовуються інтерфейси користувача у вигляді мобільних інтерфейсів або вебінтерфейсів.

Мобільні застосунки та вебінтерфейси є важливими компонентами в системах IoT, які дозволяють ефективно керувати запасами та вести моніторинг стану складу в реальному часі. Ці інструменти стають особливо цінними в регулюванні кількості товарів на фірмах, які займаються постачанням автозапчастин, де важливо підтримувати точний облік запасів для забезпечення швидкої відповіді на замовлення клієнтів.

Наприклад, компанія може розробити мобільний застосунок, що дозволяє менеджерам складу переглядати актуальні дані про наявність товарів. Застосунок може отримувати дані від IoT сенсорів, які встановлені на полицях складу та відслідковують кількість кожної запчастини. Якщо рівень запасу певної деталі падає нижче критичного порога, застосунок може надіслати сповіщення менеджеру, що спонукає до дій, наприклад, до замовлення нової партії товару.

Також, вебінтерфейс може бути використаний для більш глобального моніторингу всіх складських операцій. Через вебінтерфейс керівництво може отримувати звіти про продуктивність складу, аналізувати ефективність логістичних процесів та оптимізувати управління запасами на основі зібраних даних. Вебінтерфейс може також інтегруватися з іншими системами, наприклад, з системами CRM або ERP, що дозволяє синхронізувати дані про клієнтів і замовлення з актуальним станом запасів.

Такі мобільні застосунки та вебінтерфейси значно покращують реактивність компанії на зміни в попиті та забезпечують більш ефективне використання ресурсів, мінімізуючи ризики, пов'язані з перевантаженням або недостатністю запасів. Вони стають важливою складовою інтелектуальних систем управління складом, які базуються на технологіях IoT та

забезпечують швидкий доступ до важливої інформації, критичної для успішної комерційної діяльності.

2.5 Вибір технологій кібербезпеки

Для кібербезпеки системи потрібно врахувати захист мережі, шифрування даних, аутентифікацію та авторизацію користувачів, для захисту від несанкціонованого доступу та інших кіберзагроз:

В контексті регулювання кількості товару на фірмі, що займається постачанням автозапчастин, заходи кібербезпеки відіграють критично важливу роль, адже система повинна бути захищена від можливих кібератак та несанкціонованого доступу. Особливо це стосується IoT пристроїв, які часто стають мішенями для хакерів через свою постійну підключеність до мережі і відносну новизну технологій.

Щоб забезпечити безпеку даних, які передаються між IoT пристроями та центральними серверами, використовується шифрування. Це означає, що всі дані, які надсилаються, наприклад, з сенсорів на складі, які відслідковують кількість наявних запасів, зашифровуються перед передачею. Такий підхід гарантує, що навіть у випадку перехоплення даних, вони будуть незрозумілі та не використані зловмисниками.

Крім того, для ідентифікації та авторизації користувачів, які мають доступ до системи управління складом, використовуються суворі процедури аутентифікації. Це може включати багатофакторну аутентифікацію, де для входу в систему потрібно не лише ввести пароль, але й надати додатковий код, що надходить через мобільний додаток або на електронну пошту.

Також важливою є роль захисту мережі, до якої підключені всі IoT пристрої. Використання файрволів, антивірусного програмного забезпечення та інших мережевих захистів допомагає заблокувати спроби несанкціонованого доступу до мережі та захистити дані від витоку.

Ці заходи кібербезпеки допомагають не лише забезпечити безпечне зберігання та обробку даних, але й підтримувати надійність операцій управління запасами на складі, забезпечуючи безперебійну роботу компанії та захист від потенційних збитків, що можуть виникнути внаслідок кібератак.

2.6 Використання IoT для системи управління запасами

На прикладі фірми з багатьма філіалами по Україні, що займається продажем автозапчастин, можна детально розглянути, як IoT система з RFID тегами і зчитувачами ефективно вирішує задачі управління запасами.

RFID теги і зчитувачі використовуються для ідентифікації та відслідковування товарів. На кожен товар прикріплюється RFID мітка (рис.2.9), яка містить всю необхідну інформацію про товар, включно з унікальним серійним номером, датою виробництва, моделлю і ціною. Ці теги не потребують власного джерела енергії і активізуються при зчитуванні RFID зчитувачами, що дозволяє автоматично відстежувати переміщення товарів в межах складу та між філіалами.

RFID мітка наведена на рисунку 2.9.



Рисунок 2.9 – RFID мітка [18]

Сенсори, такі як температурні, вологості або вагові, використовуються для моніторингу стану товарів. Наприклад, температурні сенсори можуть використовуватися для перевірки, що товари, чутливі до температурних змін, такі як харчові продукти або фармацевтичні засоби, зберігаються при оптимальних умовах.

Ось як вони працюють у контексті регулювання кількості товару:

- встановлення RFID тегів на товари: мітки з пасивними RFID тегами прикріплюються до кожного товару, упаковки або палети. Кожен тег містить унікальний серійний номер, який ідентифікує товар; пасивні теги не мають власного джерела живлення і активуються сигналом від RFID зчитувача;

- зчитування інформації за допомогою RFID зчитувачів: RFID зчитувачі розміщуються на ключових точках у місцях зберігання або на лініях пересування товарів (наприклад, на входах/виходах складів, касових зонах); коли товари з RFID тегами проходять поруч із зчитувачем, тег активується і передає дані зчитувачу, який в свою чергу фіксує інформацію про товар;

- автоматичне відслідковування та управління запасами: інформація, отримана від RFID зчитувачів, автоматично оновлює базу даних системи управління запасами. Це дозволяє вести точний облік запасів, виявляти їх місцезнаходження у реальному часі, відслідковувати переміщення товарів в межах складу та між складами.

Система може автоматично генерувати замовлення на поповнення запасів, коли рівень запасів досягає критично низького рівня, та попереджати про надлишок.

Система IoT наведена на рисунку 2.10.



Рисунок 2.10 – Система IoT [19]

Використання RFID технології в такий спосіб не тільки сприяє підвищенню ефективності операцій, але і покращує задоволення клієнтів, оскільки здатне гарантувати наявність необхідних товарів у потрібний час. Це призводить до швидшого обслуговування клієнтів і зменшення часу очікування, що є критично важливим у конкурентному ринку роздрібної торгівлі автозапчастинами.

Крім того, використання IoT і RFID технологій дозволяє фірмі ефективно контролювати і виявляти несанкціоновані переміщення товарів або їх втрату. Наприклад, у випадку спроби викрадення або неправильного розміщення товару, система негайно виявить це через невідповідність даних зчитувачів і місцеположення товару. Такі інциденти можуть бути швидко виявлені і виправлені, забезпечуючи високий рівень безпеки запасів.

Автоматизація процесів з допомогою RFID технологій також значно підвищує точність управління запасами. Замість ручного введення даних або сканування штрих-кодів, що може призвести до помилок, система веде точний облік кількості і місцеположення кожної одиниці товару. Це мінімізує

можливість помилок або збоїв в управлінні запасами, які можуть коштувати компанії значних витрат.

Відповідно до збільшення точності і ефективності управління, використання таких інноваційних технологій також допомагає покращити загальну операційну продуктивність компанії. Завдяки мінімізації часу на обробку замовлень і покращенню точності запасів, компанія може швидше реагувати на зміни в попиті, а також ефективніше планувати свою закупівельну діяльність і логістику.

Загалом, використання IoT з RFID тегами в мережі філіалів не тільки забезпечує оперативне управління запасами, але й приносить значні стратегічні переваги, які сприяють підвищенню загальної конкурентоспроможності фірми на ринку.

2.7 Опис плану відправки товару з центрального складу на філіал

План відправки товару з центрального складу та прийому його на філіалі можна розглянути у таких етапах:

Етап 1. Планування відправки: аналіз запасів і потреби філіалів; формування замовлення на основі потреб філіалу.

Етап 2. Підготовка товару: вибір товарів на центральному складі; пакування та маркування товарів, включно з прив'язкою RFID тегів до кожної одиниці товару. Пакування та маркування товару наведено на рисунку 2.11.



Рисунок 2.11 – Пакування та маркування товару [20]

Етап 3. Оформлення документації: створення накладної та інших супровідних документів; відправка електронних копій документів у філіал для підготовки до прийому товару.

Етап 4. Транспортування: завантаження товарів у транспортний засіб; транспортування товару до філіалу з використанням GPS для моніторингу місцеположення.

Етап 5. Прийом товару на філіалі: перевірка товару і документів при прибутті; зчитування RFID тегів для підтвердження кількості та ідентифікації товарів.

Прийом товару на філіалі наведено на рисунку 2.12.



Рисунок 2.12 – Прийом товару на філіалі [21]

Етап 6. Оновлення системи управління запасами: внесення даних про прийом товару у комп'ютерну систему управління запасами філіалу; синхронізація даних з центральною базою для оновлення загального статусу запасів.

Етап 7. Перевірка та звірка: звірка фактичного стану товарів із даними в системі; вирішення можливих розбіжностей та оформлення акту прийому-передачі товарів.

Етап 8. Звітування: підготовка звітів про успішну доставку і прийом товару; надсилання звітів керівництву та відділам, які це стосується.

Цей приклад забезпечує ефективне і точне переміщення товарів між центральним складом і філіалами, мінімізуючи помилки і підвищуючи продуктивність логістичних операцій.

2.8 Опис взаємодії комірника з системою IoT для управління запасами

Коли комірник взаємодіє з системою IoT для управління запасами автозапчастин, він може скористатися мобільним пристроєм з вбудованим RFID-зчитувачем для вибору товарів на складі. Після вибору товарів, комірник підтверджує їх вибір, скануючи RFID-теги. Далі він пакує товари для відправлення та наклеює RFID-теги на упаковку. Після пакування комірник може відсканувати усі упаковки для підтвердження готовності до відправлення. Коли відправлення досягає філіалу, комірник на філіалі використовує RFID-зчитувач для прийому товарів, підтверджуючи прибуття товарів та оновлюючи інформацію про запаси у філіалі.

Після прийому товарів на філіалі, комірник може використовувати мобільний пристрій для перевірки товарів та їх розміщення на складі. Він може сканувати RFID-теги товарів для підтвердження наявності та місцеположення кожної одиниці товару на складі. Крім того, якщо товари виходять зі складу для використання або продажу, комірник може відсканувати RFID-теги для реєстрації вибуття товарів та оновлення інформації про залишки на складі.

Загалом, взаємодія комірника з системою IoT для управління запасами автозапчастин передбачає використання мобільних пристроїв з RFID-зчитувачами для вибору, пакування, прийому та розміщення товарів на складі, що дозволяє ефективно ведення обліку запасів та покращує процес управління запасами.

Пакування та маркування товару наведено на рисунку 2.13.



Рисунок 2.13 – Комірник зчитує на товарі RFID тег [22]

2.9 Висновки з розділу

У цьому розділі було проаналізовано ключові аспекти використання IoT технологій для регулювання кількості товару на складі. Розглянуто приклад застосування такої системи, який показує, як IoT пристрої сприяють автоматизації процесів відправки товарів з центрального складу на філіали та їх подальшого прийому. Наприклад, скануючи RFID-теги, комірник може підтвердити вибір товарів на центральному складі та їх прийом на філіалі, що значно спрощує контроль і облік товарів.

План відправки товарів з центрального складу на філіал включає кроки від планування та підготовки товарів до їх транспортування та прийому на філіалі. Цей процес стає більш ефективним та точним завдяки використанню RFID-тегів та інших IoT пристроїв, які дозволяють відстежувати та керувати товарами в режимі реального часу.

У висновку можна зазначити, що впровадження IoT технологій для управління запасами дозволяє значно підвищити ефективність та точність процесів управління, зменшити ризики та оптимізувати ланцюг поставок. Завдяки автоматизації багатьох процесів, які раніше вимагали значних

людських ресурсів та часу, підприємства можуть суттєво знизити витрати та підвищити свою конкурентоспроможність. Інтеграція IoT систем в управління запасами відкриває нові можливості для підвищення продуктивності та забезпечення стабільності бізнес-процесів.

3 ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ СИСТЕМИ УПРАВЛІННЯ ЗАПАСАМИ

3.1 Архітектура системи

Система управління запасами складається з декількох основних компонентів, які взаємодіють між собою для забезпечення ефективного моніторингу та регулювання кількості товару.

Основні компоненти системи включають RFID теги, RFID зчитувачі, сенсори, IoT шлюз, хмарну платформу та інтерфейси користувача. RFID теги, наклеєні на кожен товар, містять унікальні ідентифікатори та іншу інформацію про товар, таку як назва, модель та серійний номер. RFID зчитувачі, розташовані на різних етапах логістичного ланцюга (на складі, у точках продажу тощо), зчитують дані з RFID тегів та передають їх до системи управління запасами.

Додаткові сенсори (температури, вологості, ваги тощо) встановлюються для моніторингу стану товарів і забезпечення їх збереження в належних умовах. IoT шлюз збирає дані від RFID зчитувачів та сенсорів і передає їх до хмарної платформи для подальшої обробки. Хмарна платформа (AWS IoT, Google Cloud IoT, Microsoft Azure IoT) обробляє отримані дані, зберігає їх і забезпечує доступ до них для користувачів системи.

Інтерфейси користувача, веб- або мобільні застосунки для менеджерів складу, продавців та інших користувачів, дозволяють переглядати та керувати запасами в реальному часі.

Архітектура системи управління запасами наведена на рисунку 3.1.

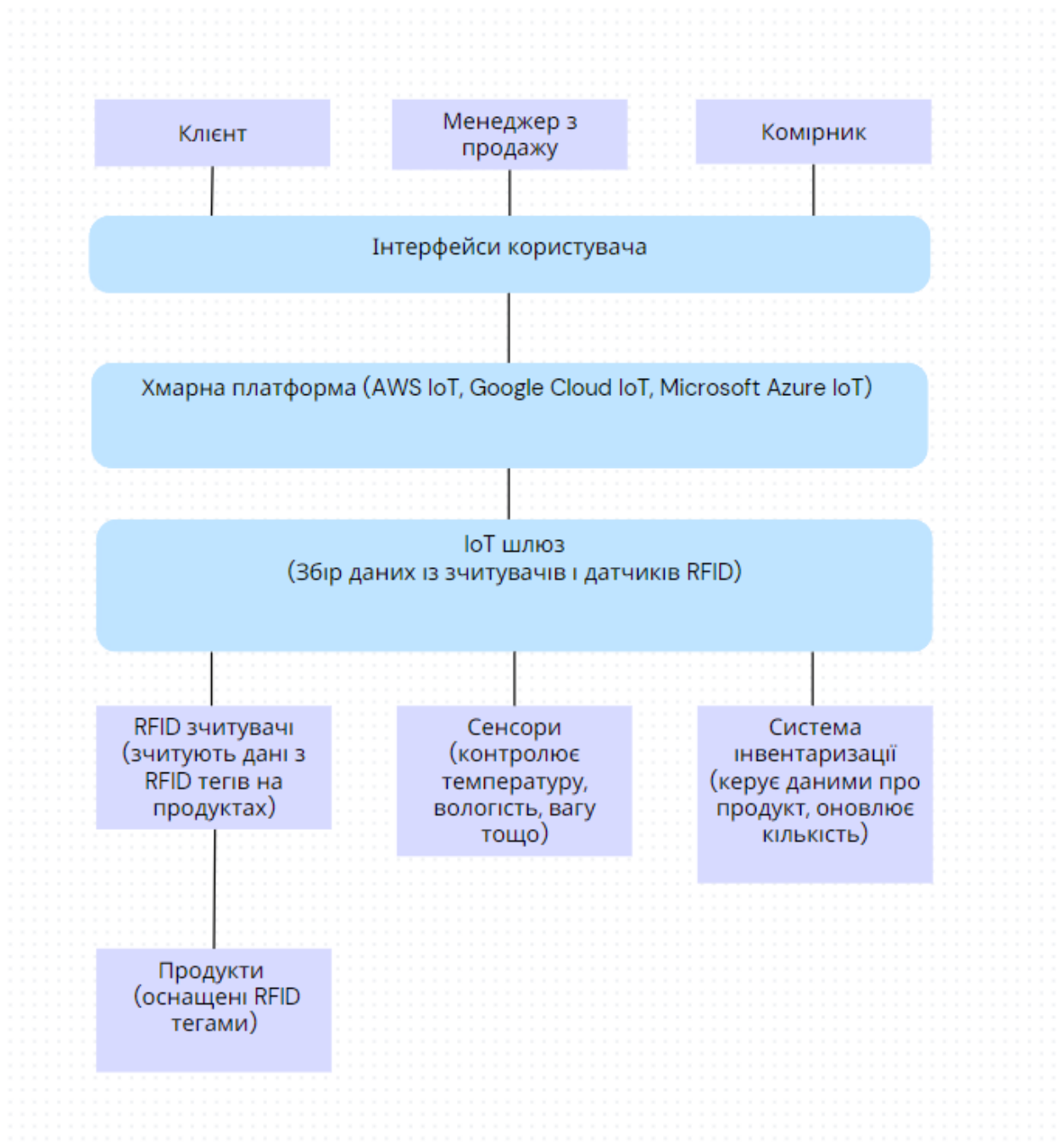


Рисунок 3.1 – Архітектура системи

3.2 Діаграма прецедентів

Використання RFID тегів може значно оптимізувати роботу фірми, що займається роздрібним продажем автозапчастин, починаючи від приймання товару на склад і закінчуючи продажем кінцевому споживачу.

Технологія роботи системи наведена у вигляді наступних етапів:

Етап 1. Приймання товару на склад. Коли нові автозапчастини надходять на склад, кожній одиниці товару присвоюється унікальний RFID тег. Цей тег містить важливу інформацію про товар, таку як назва запчастини, модель, серійний номер, дата виробництва тощо. Товари з тегами скануються спеціалізованим RFID читачем, який автоматично реєструє їх у системі управління запасами. Це дозволяє миттєво оновлювати інформацію про наявність на складі, мінімізуючи ручну роботу та помилки.

Етап 2. Зберігання та інвентаризація: запчастини з RFID тегами легко відстежувати на складі. Система може автоматично ідентифікувати місцезнаходження кожної одиниці, спрощуючи процес пошуку потрібного товару та знижуючи час на підготовку замовлень. Також, завдяки точному відслідковуванню, інвентаризація може проводитися значно швидше і з меншими зусиллями, оскільки персоналу не потрібно вручну перевіряти кожен товар.

Етап 3. Автоматизація продажу та видачі товару: коли клієнт купує автозапчастину, RFID читач на касі може швидко сканувати тег, автоматично ідентифікувати товар та обробити продаж, знову ж таки мінімізуючи ручний ввід даних. Це не тільки прискорює обслуговування клієнтів, але й зменшує ймовірність помилок при розрахунку.

Етап 4. Відстеження переміщення товарів: RFID система дозволяє відстежувати переміщення кожної одиниці товару від складу до точки продажу. Це надзвичайно корисно для аналізу логістики всередині компанії, дозволяючи ідентифікувати "вузькі місця" в процесі обробки товарів та вносити необхідні корективи для оптимізації потоків товарів.

Етап 5. Повернення та гарантійне обслуговування: у разі повернення або гарантійного обслуговування автозапчастин, RFID теги дозволяють легко ідентифікувати товар, перевірити його історію продажів та гарантійні

зобов'язання. Це спрощує процес приймання повернень та швидкого вирішення питань, пов'язаних з гарантією.

Діаграму прецедентів наведено на рисунку 3.2.

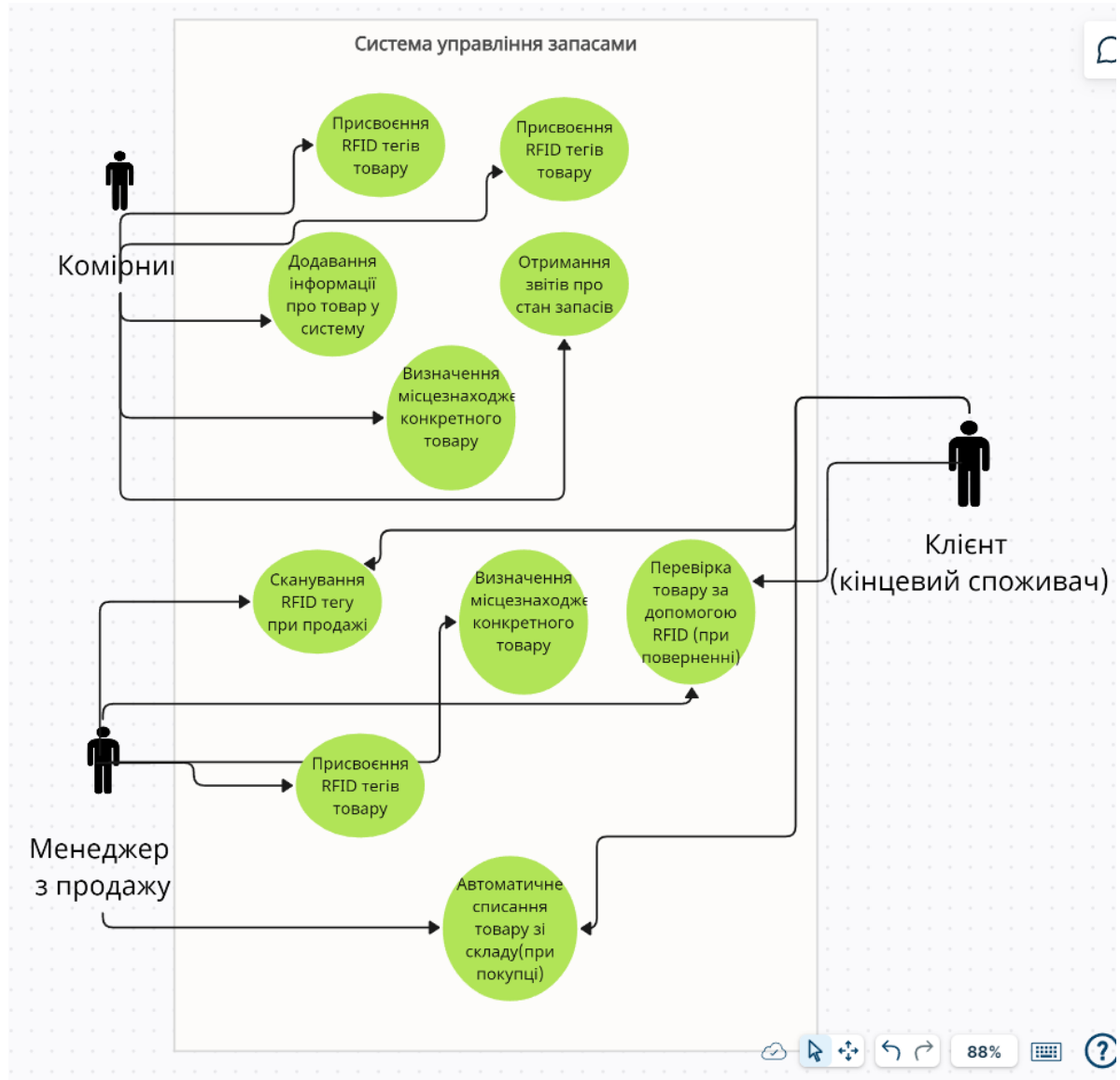


Рисунок 3.2 – Діаграма прецедентів

3.3 Діаграма класів

Діаграма класів в UML (Unified Modeling Language) використовується для візуалізації структури класів системи та їх взаємозв'язків. Діаграма класів системи наведена на рисунку 3.3.

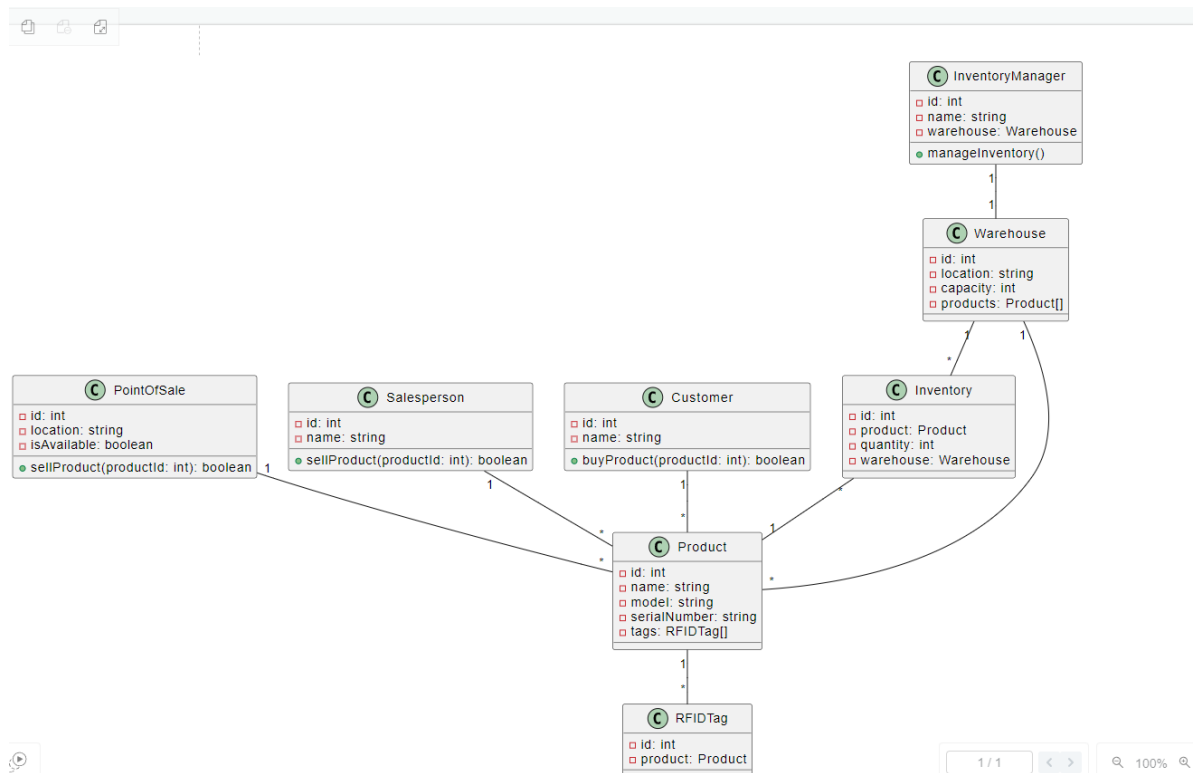


Рисунок 3.3 – Діаграма класів

У контексті системи управління запасами з використанням RFID технології для компанії, що займається роздрібним продажем автозапчастин, розроблені наступні класи.

Клас **RFIDTag** – представляє RFID тег, який містить інформацію про товар (назва, модель, серійний номер тощо). Опис методів та даних класу **RFIDTag** наведено у таблиці 3.1.

Таблиця 3.1 – Опис класу RFIDTag

Атрибути	Тип	Опис
id	int	унікальний ідентифікатор RFID тегу
product	Product	унікальний ідентифікатор RFID тегу

Клас Product – представляє товар, який може мати один або декілька RFID тегів. Опис методів та даних класу Product наведено у таблиці 3.2.

Таблиця 3.2 – Опис класу Product

Атрибути	Тип	Опис
id	int	унікальний ідентифікатор товару
name	string	назва товару
model	string	модель товару
serialNumber	string	серійний номер товару
tags	RFIDTag[]	масив RFID тегів, прикріплених до товару

Клас Warehouse – представляє склад, на якому зберігаються товари. Опис методів та даних класу Warehouse наведено у таблиці 3.3.

Таблиця 3.3 – Опис класу Warehouse

Атрибути	Тип	Опис
id	int	унікальний ідентифікатор складу
location	string	розташування складу
capacity	int	максимальна місткість складу
products	Product[]	список товарів на складі

Клас `InventoryManager` - представляє менеджера або комірника складу, який відповідає за управління запасами на складі. Опис методів та даних класу `InventoryManager` наведено у таблиці 3.4.

Таблиця 3.4 – Опис класу `InventoryManager`

Атрибути	Тип	Опис
id	int	унікальний ідентифікатор менеджера
name	string	ім'я менеджера
warehouse	Warehouse	склад, який керує менеджер
Методи	Повертає	опис
manageInventory()	void	управління запасами на складі

Клас `PointOfSale` – представляє точку продажу, де здійснюється продаж товарів. Опис методів та даних класу `PointOfSale` наведено у таблиці 3.5.

Таблиця 3.5 – Опис класу `PointOfSale`

Атрибути	Тип	Опис
id	int	унікальний ідентифікатор точки продажу
location	string	розташування точки продажу
isAvailable	boolean	доступність точки продажу
Методи	Повертає	опис
sellProduct(productId)	boolean	продаж продукту за його ID

Клас Customer – представляє клієнта, який купує товари. Опис методів та даних класу Customer наведено у таблиці 3.6.

Таблиця 3.6 – Опис класу Customer

Атрибути	Тип	Опис
id	int	унікальний ідентифікатор клієнта
name	string	ім'я клієнта
Методи	Повертає	опис
buyProduct(productId)	boolean	купівля продукту за його ID

Клас Salesperson – представляє продавця, який продає товари клієнтам. Опис методів та даних класу Salesperson наведено у таблиці 3.7.

Таблиця 3.7 – Опис класу Salesperson

Атрибути	Тип	Опис
id	int	унікальний ідентифікатор продавця
name	string	ім'я продавця
Методи	Повертає	опис
sellProduct(productId)	boolean	продаж продукту за його ID

Клас InventorySystem – представляє систему управління запасами, яка взаємодіє з RFID читачами та базою даних. Опис методів та даних класу InventorySystem наведено у таблиці 3.8.

Таблиця 3.8 – Опис класу InventorySystem

Атрибути	Тип	Опис
id	int	унікальний ідентифікатор запису про запаси
product	Product	товар, наявний на складі
quantity	int	кількість даного товару на складі
warehouse	Warehouse	склад, де зберігається товар

Ці класи формують основу системи управління запасами з використанням RFID технології, дозволяючи забезпечити ефективне відстеження товарів і управління запасами в роздрібній торгівлі автозапчастинами.

3.4 Опис класів процесу прийому товару комірником шляхом сканування

Діаграма класів прийому товару комірником шляхом сканування наведена на рисунку 3.4.

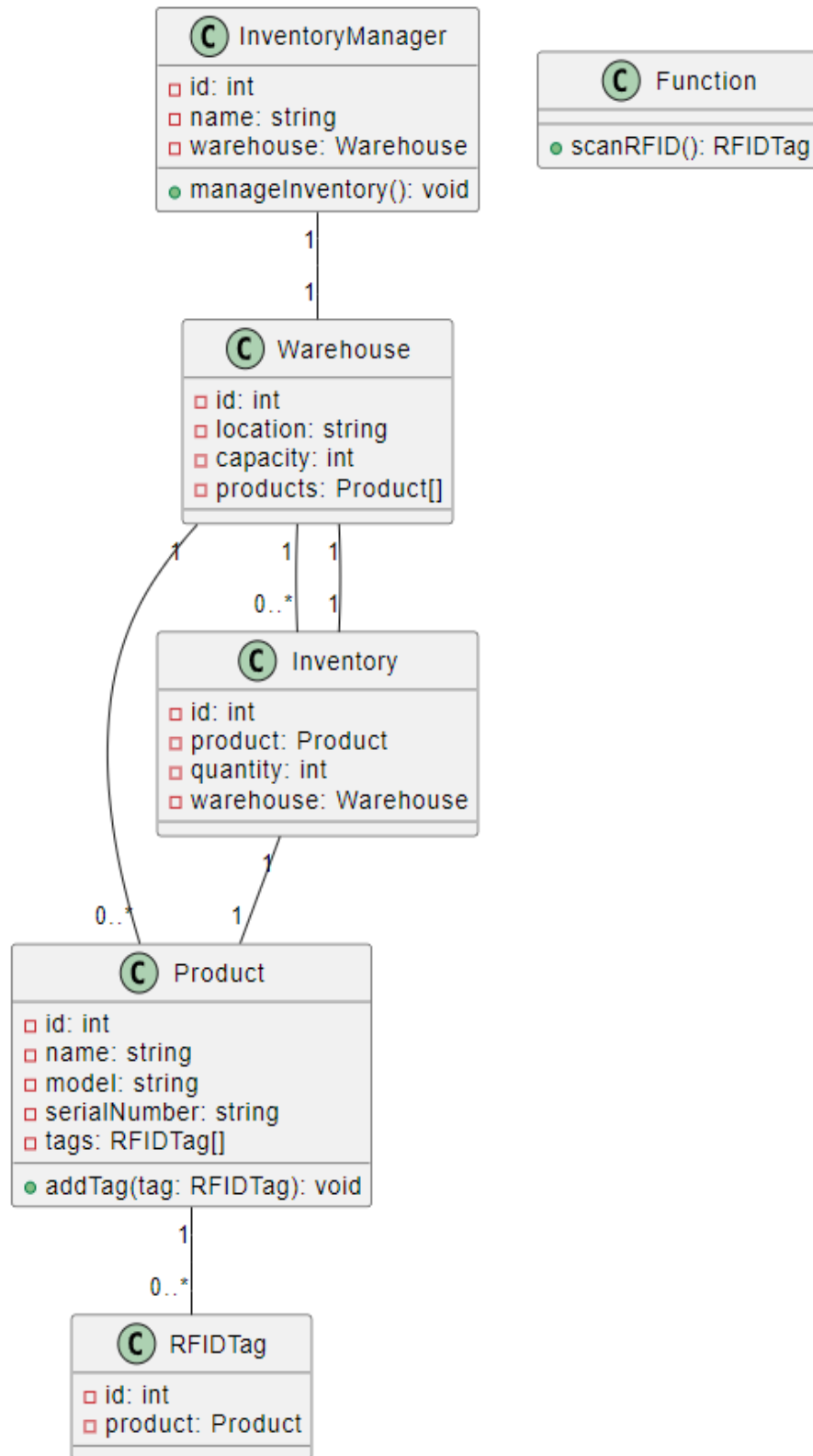


Рисунок 3.4 – Діаграма класів прийому товару комірником

Опис класів діаграми наступний:

- клас **RFIDTag** представляє RFID тег;

- клас Product представляє товар і містить метод для додавання тегу;
- клас Warehouse представляє склад, де зберігаються товари;
- клас Inventory представляє інвентаризацію товарів на складі;
- клас InventoryManager управляє запасами на складі.

Функція scanRFID є макетом для симуляції процесу сканування RFID тегу. У реальній програмі ця функція буде взаємодіяти з реальним RFID зчитувачем.

3.5 Інтерфейс користувача у RFID зчитувачі

Інтерфейс користувача у RFID зчитувачі наведена на рисунку 3.5.

Процес прийняття товару комірником починається зі сканування RFID-тегу товару за допомогою RFID-зчитувача. Коли тег скановано, програма отримує з нього інформацію про товар, таку як унікальний ідентифікатор товару. Далі ця інформація використовується для пошуку відповідного товару в базі даних або системі управління складом. Після ідентифікації товару на екрані інтерфейсу програми відображається назва товару, його модель, серійний номер та фотографія товару для візуальної звірки.

Комірник звіряє товар, що знаходиться перед ним, із зображенням та іншою інформацією, відображеною на екрані, щоб переконатися, що це правильний товар. Після цього комірник вводить кількість товару, яку він отримує, в спеціальне поле в інтерфейсі програми. Програма перевіряє, чи введена кількість не перевищує максимальну місткість складу та чи відповідає іншим логістичним вимогам. Якщо все в порядку, комірник підтверджує прийняття товару, натискаючи відповідну кнопку в інтерфейсі.



Рисунок 3.5 – Інтерфейс користувача у RFID зчитувачі

Після підтвердження програма автоматично оновлює записи про запаси на складі, додаючи новий товар або оновлюючи кількість існуючого товару. Ця інформація зберігається в базі даних для подальшого використання в

управлінні запасами. Оновлені дані можуть включати місцезнаходження товару на складі, його кількість та інші важливі логістичні параметри.

У разі виникнення будь-яких проблем, наприклад, якщо ідентифікований товар не відповідає наявному на складі або виникає помилка при введенні кількості, комірник може побачити відповідне повідомлення на екрані і вжити необхідних заходів для вирішення проблеми. Це може включати перевірку даних повторно або виклик старшого менеджера для додаткової перевірки.

3.6 Висновки до розділу 3

У цьому розділі розробили архітектуру системи, діаграму прецедентів, діаграму класів та наведено опис класів системи.

Також було розроблено інтерфейс користувача у RFID зчитувачі та описано процес прийняття товару користувачем (комірником).

4 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ РОЗРОБЛЕНОЇ ПРОГРАМИ

4.1 Оцінювання системи інтернет речей управління запасами

Оцінювання систем IoT для управління запасами включає кілька важливих показників, які допомагають визначити ефективність, надійність і економічну вигоду від їх впровадження.

Основні показники оцінювання таких систем включають точність відстеження запасів, рівень точності даних, тобто відсоток правильно зареєстрованих і відстежених одиниць запасів, та час оновлення даних, що вказує на час, необхідний для оновлення інформації про запаси в системі після фізичного переміщення або зміни кількості.

Ефективність управління запасами вимірюється зниженням надлишкових запасів, відсотком зменшення надлишкових запасів після впровадження системи, та оптимізацією рівня запасів, що визначається відсотком зменшення випадків нестачі або перевищення запасів.

Економічна ефективність оцінюється за допомогою показника зниження витрат на управління запасами, тобто відсотка зниження операційних витрат, пов'язаних з управлінням запасами, та повернення інвестицій (ROI), що відображає відношення прибутку від використання системи до витрат на її впровадження і обслуговування.

Операційна ефективність вимірюється часом виконання операцій, тобто часом, необхідним для виконання стандартних операцій управління запасами, таких як прийом, зберігання та відвантаження товарів, та швидкістю обробки замовлень, що вказує на час від отримання замовлення до його виконання.

Надійність і стабільність системи визначається часом безвідмовної роботи (uptime), відсотком часу, протягом якого система працює без збоїв, та частотою помилок, тобто кількістю помилок або збоїв на певний обсяг операцій або час роботи.

Інтеграція і масштабованість системи включає сумісність з іншими системами, що вказує на легкість інтеграції системи з існуючими програмами управління підприємством (ERP), CRM тощо, та масштабованість, яка визначається можливістю системи розширювати свої функції та охоплення без значних витрат або зусиль.

Безпека і конфіденційність даних вимірюються рівнем захищеності даних від несанкціонованого доступу та кібератак, а також відповідністю системи вимогам міжнародних стандартів і регуляторних норм у сфері захисту даних. Ці показники дозволяють підприємствам об'єктивно оцінити ефективність і вигоду від впровадження IoT систем для управління запасами, а також визначити можливі області для покращення.

4.2 Оцінка точності даних та часу оновлення даних

Для цього було створено імітаційну процесу реєстрації та відстеження одиниць запасів з використанням RFID-тегів. Будемо моделювати випадкові помилки реєстрації і відстеження та обчислювати відсоток правильно зареєстрованих одиниць за формулою (4.1)

$$\text{accuracy} = \left(\frac{\sum_{i=1}^n |F_i - R_i|}{\sum_{i=1}^n F_i} \right), \quad (4.1)$$

де ассигасу – рівень точності;

F_i – фактична кількість одиниць товару i за результатами фізичної інвентаризації;

R_i – реальна кількість одиниць товару i за даними системи;

n – загальна кількість різних товарів.

Для рахування часу оновлення даних будемо використовувати формулу (4.2)

$$T_{онов} = t_k - t_n \quad (4.2)$$

де $T_{онов}$ – час оновлення даних;

t_k – час завершення оновлення даних;

t_n – час початку оновлення даних.

4.3 Результати експериментального дослідження

Для проведення експериментального дослідження можна змінювати кількість товарів та рівень помилок, щоб дослідити їхній вплив на точність даних. Будемо моделювати випадкові помилки реєстрації і відстеження та обчислювати відсоток правильно зареєстрованих одиниць за формулою (4.1)

Результати експерименту наведені в таблицях 4.1-4.2.

Таблиця 4.1 – Результати експерименту

Загальна кількість товарів	Правильно зареєстровані товари	Рівень точності даних (%)
100	96	96.0
180	178	98.9
195	191	98
210	205	97.6

В таблиці 4.1 загальна кількість товарів – це кількість товарів, доданих до системи під час експерименту; правильно зареєстровані товари – кількість товарів, які були правильно зареєстровані в системі; рівень точності даних (%) це відсоток правильно зареєстрованих товарів від загальної кількості товарів.

Для проведення експериментального дослідження часу оновлення даних: часу, необхідного для оновлення даних про запаси в системі після фізичного переміщення або зміни кількості можна змінювати кількість продуктів та додати випадкову затримку оновлення даних для моделювання реальних умов.

Таблиця 4.2 – Результати експерименту

Номер товару	Час оновлення даних (мс)
1	150
2	189
3	134
4	198
5	157
6	123
7	169
8	140
9	174
10	143
...	...
100	137

В таблиці 4.2 номер товару це ідентифікатор позиції, для якого проводилося оновлення даних; час оновлення даних (мс): час, необхідний для оновлення даних про запас продукту в системі.

Середній час оновлення даних: 150.3 мс (значення можуть варіюватися залежно від випадкової затримки).

Ці результати демонструють час, необхідний для оновлення даних про запаси в системі управління запасами з використанням IoT-технологій та RFID-тегів. Як видно з таблиці, час оновлення даних може варіюватися, але середній час оновлення даних є важливим показником ефективності системи.

4.4 Подальші покращення системи

Для подальшого покращення системи управління запасами з використанням RFID-технологій та IoT можна розглянути декілька напрямів модернізації.

По-перше, доцільно додати нові сенсори, такі як сенсори освітлення, рівня шуму та інших параметрів, що дозволить більш детально контролювати умови зберігання товарів і забезпечити збереження чутливих до зовнішніх факторів товарів.

По-друге, можна використовувати алгоритми машинного навчання та штучного інтелекту для прогнозування попиту на товари та оптимізації запасів, що допоможе зменшити надлишкові запаси та запобігти дефіциту товарів.

Розробка роботизованих систем для автоматичної інвентаризації товарів на складах значно зменшить час та трудовитрати на проведення інвентаризації.

Впровадження новітніх технологій передачі даних, таких як 5G, підвищить швидкість та надійність передачі даних між RFID-зчитувачами, сенсорами та хмарною платформою. Розширення функціональності хмарної платформи через додавання нових аналітичних інструментів та можливостей для звітності дозволить користувачам отримувати більш детальні та інформативні звіти про стан запасів.

Здійснення інтеграції системи управління запасами з іншими системами компанії, такими як система управління виробництвом (MES), система управління замовленнями (OMS) та система управління транспортом (TMS), забезпечить єдиний інформаційний простір та підвищить ефективність управління бізнес-процесами.

Покращення інтерфейсів користувача зробить їх більш зручними та інтуїтивними, що допоможе знизити час на навчання персоналу та підвищити продуктивність роботи з системою.

Впровадження додаткових заходів безпеки для захисту даних від несанкціонованого доступу та кібератак, таких як шифрування даних, багатофакторна аутентифікація та регулярні аудити безпеки, підвищить захист даних. Адаптація системи для роботи в різних регіонах світу з урахуванням локальних умов та вимог включає підтримку різних мов, валют та податкових систем, що дозволить глобально розширити використання системи.

Впровадження екологічно стійких рішень, таких як зменшення енергоспоживання сенсорів та RFID-зчитувачів, використання екологічно безпечних матеріалів для виробництва RFID-тегів, сприятиме підвищенню екологічної стійкості системи.

4.5 Висновки до розділу 4

У цьому розділі було розглянуто процес моделювання та експериментального дослідження системи управління запасами з використанням IoT-технологій.

Використовуючи мову програмування C++, було створено модель, яка імітує реєстрацію, оновлення та відстеження товарів за допомогою RFID-тегів. За допомогою цього моделювання було досліджено точність даних та часу оновлення інформації про запаси. Результати показали, що рівень точності даних та час оновлення значною мірою залежать від налаштувань системи та випадкових факторів. Впровадження IoT-рішень у систему управління запасами дозволяє автоматизувати процеси, підвищуючи ефективність та точність обліку товарів, що у свою чергу сприяє зниженню витрат і покращенню якості обслуговування клієнтів. Використання сучасних технологій, таких як RFID та моделі для управління запасами, є важливим кроком до оптимізації логістичних процесів і забезпечення конкурентних переваг на ринку.

ВИСНОВКИ

Результатом виконання дипломної кваліфікаційної роботи є система IoT регулювання кількості товару за допомогою RFID тегів та RFID зчитувачів, розроблена програма для прийняття та обліку товару.

У ході роботи було вирішено наступні задачі:

- дослідити існуючі технології реалізації систем управління запасами;
- дослідити апаратні рішення на основі IoT та RFID-технологій;
- виконати проектування та реалізацію прототипу системи управління запасами;
- провести експериментальне дослідження розробленої системи.

У першому розділі провели огляд проблеми управління запасами товару та знайшли розв'язання цих проблем; способи управління запасами товару без використання IoT, а саме розібрали кожний метод та прийшли до висновку, що кожен з цих методів має свої переваги та обмеження та може бути ефективним у залежності від специфіки бізнесу, обсягу запасів та цілей управління; розібравши впровадження IoT дійшли до висновку, що воно вимагає комплексного підходу; а також розібрали загальну структуру.

У другому розділі була проаналізована апаратна частина; розроблені порівняльні таблиці RFID тегів, RFID зчитувачів, мережевої інфраструктури, хмарних платформ, була проаналізована кібербезпека.

У третьому розділі були розроблені архітектура системи; діаграма прецедентів та діаграма класів. Також було розроблено інтерфейс користувача у RFID зчитувачі.

У четвертому розділі було проведено оцінювання системи інтернет речей управління запасами, проведені експериментальні дослідження оцінки точності даних та часу оновлення даних; описані подальші покращення системи, завдяки яким система у майбутньому стане ще більш ефективною, точною та надійною, забезпечуючи ще краще управління запасами і знижуючи витрати на їхнє зберігання.

В роботі розроблено прототип системи управління запасами автозапчастин з використанням RFID-технологій у поєднанні з IoT, що дозволяє автоматизувати процеси прийому та обліку товарів з метою зниження рівня помилок та підвищення точності даних.

Розроблена система забезпечує точне і швидке оновлення даних про запаси, підвищуючи ефективність управління логістичними процесами.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Інтернет речей (IoT) [Електроний ресурс]. - Режим доступу: <https://ru.wikipedia.org/wiki/>
2. Nichols, M. R. Using IoT and RFID for Better Inventory Management [Електроний ресурс]. – Режим доступу: <https://www.iotevolutionworld.com/m2m/articles/443664-using-iot-rfid-betterinventory-management.htm>.
3. Shiklo, B. IoT-driven inventory management [Електроний ресурс]. / B. Shiklo // INTERNET OF THINGS - CIO BLOG. – Режим доступу: <https://www.scnsoft.com/blog/iot-for-inventory-management>
4. Diagnosis and fault-tolerant control / [M. Blanke, M. Kinnaert, J. Lunze, M. Staroswiecki]. – Berlin : Springer, 2006. – 672 p..
4. RFID теги, мітки, транспондери (LF 125 кГц, HF 13,56 МГц, UHF 860-960 МГц, активные 2,45 ГГц) [Електроний ресурс]. – Режим доступу: <https://scancode.net.ua/rashodnye-materialy/rfid-metki-tegi-transpondery/rfid-tegi-metki-transpondery-lf-125-kgts-hf-1356-mgts-uhf-860-960-mgts-aktivnye-245-ggts>
5. Технологія RFID, мітки, рідери та її застосування [Електроний ресурс]. – Режим доступу: https://realtrac.com/ru/company/blog/princip_raboty_tehnologii_rfid_i_ee_primenenie/
6. Автоматизація складського обліку на основі RFID [Електроний ресурс]. – Режим доступу: <https://rfidukraine.com.ua/warehouse-automatization/>
7. RFID технології для автоматизації обліку [Електроний ресурс]. – Режим доступу: <https://remonline.app/ru/blog/rfid-technology/>
8. Технологія RFID у роздрібній торгівлі для покращення взаємодії з клієнтами та спрощення інвентаризації [Електроний ресурс]. – Режим доступу: <https://idcard.com.ua/ua/blog/tehnologiya-rfid-v-rozdribnij-torgivli/>
9. RFID-зчитувачі [Електроний ресурс]. – Режим доступу: <https://rfidukraine.com.ua/rfid->

[readers/#:~:text=%D0%9E%D0%B4%D0%BD%D0%B8%D0%BC%20%D0%B8%D0%B7%20%D0%B2%D0%B0%D0%B6%D0%BD%D0%B5%D0%B9%D1%88%D0%B8%D1%85%20%D0%BA%D0%BE%D0%BC%D0%BF%D0%BE%D0%BD%D0%B5%D0%BD%D1%82%D0%BE%D0%B2%20RFID,%D0%B8%D1%81%D1%82%D0%BE%D1%87%D0%BD%D0%B8%D0%BA%D0%BE%D0%BC%20%D1%8D%D0%BD%D0%B5%D1%80%D0%B3%D0%B8%D0%B8%20%D0%B4%D0%BB%D1%8F%20%D0%BF%D0%B0%D1%81%D1%81%D0%B8%D0%B2%D0%BD%D1%8B%D1%85%20%D0%BC%D0%B5%D1%82%D0%BE%D0%BA.](#)

10. Активний RFID тег [Електроний ресурс]. – Режим доступу: https://ru.marktraceiot.com/2-4ghz-active-rfid-tag-for-assets-control_p18.html

11. Пасивний RFID тег [Електроний ресурс]. – Режим доступу: <https://satro-paladin.com/catalog/product/187871/>

12. RFID теги з датчиком температури [Електроний ресурс]. – Режим доступу: <https://idcard.com.ua/cmc3607-active-rfid-temperature-tag/>

13. Радіочастотний RFID тег [Електроний ресурс]. – Режим доступу: https://www.shutterstock.com/ru/search/rfid-protect?image_type=vector

14. Фіксований RFID зчитувач [Електроний ресурс]. – Режим доступу: https://pikabu.ru/story/vopros_pro_pishchalki_v_supermarketakh_3841258

15. Мобільний RFID зчитувач [Електроний ресурс]. – Режим доступу: <https://enmarket.com.ua/zebra-mc3300-mc330m-gl2ha2rw-terminali-zboru-danih>

16. Зчитувачі, які вбудовані в інші системи [Електроний ресурс]. – Режим доступу: <https://delo.ua/ru/banks/v-ibox-soobshhili-kogda-vozobnovyat-rabotu-terminalov-412321/>

17. RFID зчитувач з високою продуктивністю [Електроний ресурс]. – Режим доступу: <https://prom.ua/Antenny-besprovodnyh-setej;c48654>

18. RFID мітка [Електроний ресурс]. – Режим доступу: <https://rozetka.com.ua/ua/176984172/p176984172/>

19. Система IoT [Електроний ресурс]. – Режим доступу: <https://research-journal.org/archive/6-37-2015-july/isledovanie-i-primenenie-texnologii-rfid-radiofrequencyidentification>
20. Пакування та маркування товару [Електроний ресурс]. – Режим доступу: <https://www.rfidfuture.com/ru/rfid-in-supply-chain-management-and-logistics.html>
21. Прийом товару на філіалі [Електроний ресурс]. – Режим доступу: <https://scanport.ru/blog/skladskoj-uchet-s-ispolzovaniem-rfid-oborudovaniya/>
22. Комірник зчитує на товарі RFID тег [Електроний ресурс]. – Режим доступу: <https://scanport.ru/blog/skladskoj-uchet-s-ispolzovaniem-rfid-oborudovaniya/>
23. «Архітектура та технології IoT» [Електроний ресурс]. – Режим доступу: https://learn.ztu.edu.ua/pluginfile.php/201142/mod_resource/content/1/%D0%9B-2.pdf

ДОДАТОК А

Текст програми

```
#include <iostream>
#include <vector>
#include <string>
#include <unordered_map>

// Клас RFIDTag представляє RFID тег
class RFIDTag {
public:
    int id; // Унікальний ідентифікатор тегу
    int productID; // Ідентифікатор товару

    RFIDTag(int id, int productID) : id(id), productID(productID) {}
};

// Клас Product представляє товар
class Product {
public:
    int id; // Унікальний ідентифікатор товару
    std::string name; // Назва товару
    std::string model; // Модель товару
    std::string serialNumber; // Серійний номер товару
    std::vector<RFIDTag> tags; // Список RFID тегів, прикріплених до товару

    Product(int id, const std::string& name, const std::string& model, const
std::string& serialNumber)
        : id(id), name(name), model(model), serialNumber(serialNumber) {}
```

```

// Додає RFID тег до товару
void addTag(const RFIDTag& tag) {
    tags.push_back(tag);
}
};

// Клас Warehouse представляє склад
class Warehouse {
public:
    int id; // Унікальний ідентифікатор складу
    std::string location; // Розташування складу
    int capacity; // Максимальна місткість складу
    std::vector<Product> products; // Список товарів на складі

    Warehouse(int id, const std::string& location, int capacity)
        : id(id), location(location), capacity(capacity) {}

    // Приймає товар на склад
    void receiveProduct(const Product& product) {
        products.push_back(product);
    }
};

// Клас Inventory представляє інвентаризацію товарів на складі
class Inventory {
public:
    int id; // Унікальний ідентифікатор запису про запаси
    Product product; // Товар, наявний на складі
    int quantity; // Кількість даного товару на складі
    Warehouse warehouse; // Склад, де зберігається товар

```

```
Inventory(int id, const Product& product, int quantity, const Warehouse&
warehouse)
```

```
    : id(id), product(product), quantity(quantity), warehouse(warehouse) {}
};
```

```
// Клас InventoryManager представляє менеджера складу
```

```
class InventoryManager {
```

```
public:
```

```
    int id; // Унікальний ідентифікатор менеджера
```

```
    std::string name; // Ім'я менеджера
```

```
    Warehouse warehouse; // Склад, який керує менеджер
```

```
InventoryManager(int id, const std::string& name, const Warehouse&
warehouse)
```

```
    : id(id), name(name), warehouse(warehouse) {}
```

```
// Управління запасами на складі
```

```
void manageInventory() {
```

```
    // Тут буде реалізована логіка управління запасами
```

```
}
```

```
// Приймає товар на склад
```

```
void receiveProduct(const Product& product, int quantity) {
```

```
    for (auto& p : warehouse.products) {
```

```
        if (p.id == product.id) {
```

```
            std::cout << "Товар вже є на складі." << std::endl;
```

```
            return;
```

```
        }
```

```
    }
```

```

        warehouse.receiveProduct(product);

        std::cout << "Прийнято товар: " << product.name << ", Кількість: " <<
quantity << std::endl;
    }
};

// Макет функції для симуляції сканування RFID тегу
RFIDTag scanRFID() {
    // У реальній реалізації ця функція буде взаємодіяти з RFID-зчитувачем
    return RFIDTag(1, 101); // Приклад RFID тегу з id=1 та productID=101
}

int main() {
    // Ініціалізація складу та менеджера складу
    Warehouse warehouse(1, "Головний склад", 1000);
    InventoryManager manager(1, "Іван Іванов", warehouse);

    // Симуляція сканування RFID тегу
    RFIDTag scannedTag = scanRFID();

    // Створення товару на основі сканованого RFID тегу
    Product newProduct(scannedTag.productID, "Гальмівна колодка", "Модель
X", "SN12345");
    newProduct.addTag(scannedTag);

    // Прийом товару на склад
    manager.receiveProduct(newProduct, 10);

    return 0;
}

```

Програмна реалізація рівня точності даних

```

#include <iostream>
#include <vector>
#include <cstdlib>
#include <ctime>

// Клас для представлення товару
class Product {
public:
    int id;
    bool isCorrectlyTracked;

    Product(int id, bool isCorrectlyTracked) : id(id),
isCorrectlyTracked(isCorrectlyTracked) {}
};

// Клас для моделювання процесу реєстрації та відстеження товарів
class InventorySystem {
private:
    std::vector<Product> products;
    double errorRate;

public:
    InventorySystem(double errorRate) : errorRate(errorRate) {}

    void addProduct(int id) {
        // Випадкова генерація помилки на основі встановленого рівня
ПОМИЛОК

```

```

        bool    isCorrectlyTracked    =    (static_cast<double>(rand())    /
RAND_MAX) > errorRate;
        products.push_back(Product(id, isCorrectlyTracked));
    }

double calculateAccuracy() {
    int correctlyTrackedCount = 0;
    for (const auto& product : products) {
        if (product.isCorrectlyTracked) {
            correctlyTrackedCount++;
        }
    }
    return static_cast<double>(correctlyTrackedCount) / products.size() *
100;
}

};

int main() {
    srand(static_cast<unsigned int>(time(0))); // Ініціалізація генератора
випадкових чисел

    // Створення системи управління запасами з рівнем помилок 10%
    InventorySystem inventorySystem(0.1);

    // Додавання 100 товарів
    for (int i = 1; i <= 100; ++i) {
        inventorySystem.addProduct(i);
    }

    // Обчислення та виведення точності даних

```

```
double accuracy = inventorySystem.calculateAccuracy();
std::cout << "Рівень точності даних: " << accuracy << "%" << std::endl;

return 0;

}
```

Програмна реалізація рівня точності даних

```
#include <iostream>
#include <vector>
#include <chrono>
#include <thread>
#include <random>

// Клас для представлення товару
class Product {
public:
    int id;
    std::string name;
    int quantity;

    Product(int id, std::string name, int quantity)
        : id(id), name(name), quantity(quantity) {}
};

// Клас для моделювання процесу оновлення даних про запаси
class InventorySystem {
private:
    std::vector<Product> products;

public:
```

```

void addProduct(int id, std::string name, int quantity) {
    products.push_back(Product(id, name, quantity));
}

void updateProductQuantity(int id, int newQuantity) {
    for (auto& product : products) {
        if (product.id == id) {
            product.quantity = newQuantity;
            break;
        }
    }
}

void simulateUpdateProcess(int id, int newQuantity) {
    auto start = std::chrono::high_resolution_clock::now();

    // Імітація затримки оновлення даних
    std::this_thread::sleep_for(std::chrono::milliseconds(100));

    updateProductQuantity(id, newQuantity);

    auto end = std::chrono::high_resolution_clock::now();
    std::chrono::duration<double, std::milli> duration = end - start;

    std::cout << "Час оновлення даних для продукту " << id << ": " <<
duration.count() << " мс" << std::endl;
}

};

int main() {

```

```
InventorySystem inventorySystem;
```

```
// Додавання продуктів до системи
```

```
inventorySystem.addProduct(1, "Продукт А", 50);
```

```
inventorySystem.addProduct(2, "Продукт В", 30);
```

```
inventorySystem.addProduct(3, "Продукт С", 20);
```

```
// Моделювання оновлення даних для різних продуктів
```

```
inventorySystem.simulateUpdateProcess(1, 60);
```

```
inventorySystem.simulateUpdateProcess(2, 40);
```

```
inventorySystem.simulateUpdateProcess(3, 25);
```

```
return 0;
```

```
}
```

Експериментальне дослідження точності даних з різними рівнями помилок

```
#include <iostream>
```

```
#include <vector>
```

```
#include <chrono>
```

```
#include <thread>
```

```
#include <random>
```

```
// Клас для представлення товару
```

```
class Product {
```

```
public:
```

```
    int id;
```

```
    std::string name;
```

```
    int quantity;
```

```
    Product(int id, std::string name, int quantity)
```

```
        : id(id), name(name), quantity(quantity) {}
```

```

};

// Клас для моделювання процесу оновлення даних про запаси
class InventorySystem {
private:
    std::vector<Product> products;

public:
    void addProduct(int id, std::string name, int quantity) {
        products.push_back(Product(id, name, quantity));
    }

    void updateProductQuantity(int id, int newQuantity) {
        for (auto& product : products) {
            if (product.id == id) {
                product.quantity = newQuantity;
                break;
            }
        }
    }

    double simulateUpdateProcess(int id, int newQuantity) {
        auto start = std::chrono::high_resolution_clock::now();

        // Імітація затримки оновлення даних
        std::this_thread::sleep_for(std::chrono::milliseconds(100 + rand() %
100));

        updateProductQuantity(id, newQuantity);
    }
};

```

```

        auto end = std::chrono::high_resolution_clock::now();
        std::chrono::duration<double, std::milli> duration = end - start;

        return duration.count();
    }
};

int main() {
    srand(static_cast<unsigned int>(time(0))); // Ініціалізація генератора
    випадкових чисел

    InventorySystem inventorySystem;

    // Додавання продуктів до системи
    for (int i = 1; i <= 100; ++i) {
        inventorySystem.addProduct(i, "Продукт " + std::to_string(i), 50 +
rand() % 50);
    }

    // Моделювання оновлення даних для різних продуктів і збирання
    часу оновлення
    std::vector<double> updateTimes;
    for (int i = 1; i <= 100; ++i) {
        updateTimes.push_back(inventorySystem.simulateUpdateProcess(i, 50
+ rand() % 50));
    }

    // Обчислення середнього часу оновлення даних
    double totalUpdateTime = 0;
    for (double time : updateTimes) {

```

```

        totalUpdateTime += time;
    }
    double averageUpdateTime = totalUpdateTime / updateTimes.size();

    std::cout << "Середній час оновлення даних: " << averageUpdateTime
    << " мс" << std::endl;

    return 0;
}

```

У цьому коді ми додаємо 100 продуктів до системи та моделюємо оновлення даних для кожного продукту з випадковою затримкою оновлення.

Експериментальне дослідження часу оновлення даних

```

#include <iostream>
#include <vector>
#include <chrono>
#include <thread>
#include <random>

// Клас для представлення товару
class Product {
public:
    int id;
    std::string name;
    int quantity;

    Product(int id, std::string name, int quantity)
        : id(id), name(name), quantity(quantity) {}
};

// Клас для моделювання процесу оновлення даних про запаси

```

```

class InventorySystem {
private:
    std::vector<Product> products;

public:
    void addProduct(int id, std::string name, int quantity) {
        products.push_back(Product(id, name, quantity));
    }

    void updateProductQuantity(int id, int newQuantity) {
        for (auto& product : products) {
            if (product.id == id) {
                product.quantity = newQuantity;
                break;
            }
        }
    }

    double simulateUpdateProcess(int id, int newQuantity) {
        auto start = std::chrono::high_resolution_clock::now();

        // Імітація затримки оновлення даних
        std::this_thread::sleep_for(std::chrono::milliseconds(100 + rand() %
100));

        updateProductQuantity(id, newQuantity);

        auto end = std::chrono::high_resolution_clock::now();
        std::chrono::duration<double, std::milli> duration = end - start;
    }
}

```

```

        return duration.count();
    }
};

int main() {
    srand(static_cast<unsigned int>(time(0))); // Ініціалізація генератора
випадкових чисел

    InventorySystem inventorySystem;

    // Додавання продуктів до системи
    for (int i = 1; i <= 100; ++i) {
        inventorySystem.addProduct(i, "Продукт " + std::to_string(i), 50 +
rand() % 50);
    }

    // Моделювання оновлення даних для різних продуктів і збирання
часу оновлення
    std::vector<double> updateTimes;
    for (int i = 1; i <= 100; ++i) {
        updateTimes.push_back(inventorySystem.simulateUpdateProcess(i, 50
+ rand() % 50));
    }

    // Обчислення середнього часу оновлення даних
    double totalUpdateTime = 0;
    for (double time : updateTimes) {
        totalUpdateTime += time;
    }
    double averageUpdateTime = totalUpdateTime / updateTimes.size();

```

```
std::cout << "Середній час оновлення даних: " << averageUpdateTime  
<< " мс" << std::endl;
```

```
return 0;
```

```
}
```