

БАКАЛАВРСЬКА РОБОТА

БР. ІІ - 17.00.00.000 ПЗ

Група ІІ-22-4

Химич Іліано

2026

Івано-Франківський національний технічний університет нафти і газу

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Химич Іліано

(прізвище, ім'я, по батькові)

УДК 004.942

(індекс)

БАКАЛАВРСЬКА РОБОТА

Програмні рішення для автоматизації процесу інвентаризації на складах

(назва роботи)

Інженерія програмного забезпечення

(назва освітньої програми)

121 - Інженерія програмного забезпечення

(шифр і назва спеціальності)

Робота містить результати власних досліджень, використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело:

Здобувач освітнього ступеня Химич І.
(підпис, ініціали та прізвище здобувача)

Науковий керівник Корнута Володимир Андрійович, доцент, к.т.н.
(підпис, прізвище, ім'я, по батькові, науковий ступінь, вчене звання керівника)

Допущено до захисту
Завідувач кафедри

доц. Бандура В.В.
(посада) (підпис) (дата) (ініціали та прізвище)

Івано-Франківський національний технічний університет нафти і газу
Інститут, факультет інформаційних технологій
Кафедра інженерії програмного забезпечення
Ступінь вищої освіти бакалавр
Спеціальність 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:

Зав. кафедрою _____ **ІПЗ**
доцент. _____ **В.В. Бандура**

“ _____ ” _____ 2026 р.

ЗАВДАННЯ
НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТОВІ
Химич Іліано

_____ (прізвище, ім'я, по-батькові)

1. Тема проекту (роботи) "Програмні рішення для автоматизації процесу інвентаризації на складах"

керівник проекту (роботи) Корнута Володимир Андрійович, доцент, к.т.н.

затверджені наказом вищого навчального закладу від “ 21 ” квітня 2026 р. № 179/7

2. Строк подання студентом проекту (роботи) 10 червня 2026 р.

3. Вихідні дані до проекту (роботи) Результати і матеріали отримані під час проходження пер-
ддипломної практики

4. Зміст розрахунково - пояснювальної записки(перелік питань, які потрібно розробити)

1. Аналіз вимог до програмного забезпечення

2. Аналіз вимог і постановка задачі

3. Проектування системи

4. Реалізація проекту

5. Тестування та впровадження

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

1 Зразок параметрів товару (рис. 2.1, ст.33)

2 Зразок розділу (рис. 2.2, ст. 34)

4. Основний кодт (рис. 2.4 ст. 36)

4 Функція виявлення qr-кодів41 (рис. 3.1, ст. 41)

5 Визначення найкорисніших інструкцій під час навчання (а) (рис. 3.5, ст. 43)

6. Консультанти розділів проекту (роботи)

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

2. Дата видачі завдання 2026 р.

Керівник

_____ (підпис)

Завдання прийняв до виконання

_____ (підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту	Примітка
1	Визначення та обґрунтування теми роботи	17.01.2026	виконано
2	Огляд існуючих концепцій, рішень та сервісів в даній області	21.02.2026	виконано
3	Побудова моделі або алгоритму власного рішення	01.03.2026	виконано
4	Документування реалізації власного оригінального рішення вибраними засобами	21.04.2026	виконано
5	Оформлення пояснювальної записки бакалаврської роботи	02.05.2026	виконано

Студент

_____ Химич І.
(підпис)

Керівник роботи

_____ Корнута В.А.
(підпис)

АНОТАЦІЯ

Бакалаврська робота містить 66 сторінки, 30 рисунків, список використаних джерел із 40 найменувань.

Метою роботи є дослідження методів автоматизації складських процесів та розробка програмного рішення для автоматизації інвентаризації на складах.

Об'єкт дослідження: процеси автоматизації інвентаризації на складських підприємствах.

Предмет дослідження: методи, технології та програмні засоби автоматизації складської інвентаризації, включаючи системи ідентифікації товарів, QR-коди та інструменти обробки даних.

Результати дослідження: проведено аналіз сучасних підходів до автоматизації складських процесів, досліджено методи використання QR-кодів для інвентаризації та реалізовано програмне рішення для автоматизації обліку.

У першому розділі розглянуто теоретичні основи автоматизації складських процесів, проблеми інвентаризації та вплив робототехніки й цифрових технологій на ефективність складських систем.

У другому розділі досліджено методи та інструменти автоматизації інвентаризації, інтеграцію QR-кодів, підходи до формування наборів даних та архітектуру програмної системи.

У третьому розділі розглянуто реалізацію функціональних можливостей системи, оцінку впливу автоматизації на ефективність складських процесів та аналіз продуктивності виявлення QR-кодів.

Висновок: використання програмних рішень для автоматизації інвентаризації на складах дозволяє підвищити точність обліку товарів, зменшити вплив людського фактора та підвищити ефективність.

КЛЮЧОВІ СЛОВА: АВТОМАТИЗАЦІЯ СКЛАДУ, ІНВЕНТАРИЗАЦІЯ, QR-КОДИ, СКЛАДСЬКІ СИСТЕМИ, ОБРОБКА ДАНИХ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ВЕБ-ТЕХНОЛОГІЇ, ОБЛІК ТОВАРІВ,

АНОТАЦІЯ

The thesis consists of 66 pages, 30 figures, , and a reference list containing 40 sources.

The aim of the work is to study methods of warehouse process automation and to develop a software solution for warehouse inventory automation.

Research object: warehouse inventory automation processes.

Research subject: methods, technologies, and software tools for warehouse inventory automation, including product identification systems, QR codes, and data processing tools.

Research results: modern approaches to warehouse automation were analyzed, methods of using QR codes for inventory management were studied, the architecture of the software system was examined, and a software solution for automating warehouse inventory accounting was developed.

The first section describes the theoretical foundations of warehouse process automation, inventory management problems, and the impact of robotics and digital technologies on warehouse system efficiency.

The second section analyzes methods and tools for inventory automation, QR code integration, approaches to dataset formation, and the architecture of the software system.

The third section covers the implementation of the system's functional capabilities, evaluation of the impact of automation on warehouse process efficiency, and analysis of QR code detection performance.

Conclusion: the use of software solutions for warehouse inventory automation improves inventory accuracy, minimizes the influence of the human factor, and increases the efficiency of warehouse management processes.

KEY WORDS: WAREHOUSE AUTOMATION, INVENTORY MANAGEMENT, QR CODES, WAREHOUSE SYSTEMS, DATA PROCESSING, SOFTWARE, WEB TECHNOLOGIES, PRODUCT ACCOUNTING, АВТОМАТИЗАЦІЯ ПРОЦЕСІВ.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ АВТОМАТИЗАЦІЇ ПРОЦЕСІВ ІНВЕ- НТАРИЗАЦІЇ НА СКЛАДАХ	10
1.1 Основні принципи автоматизації складських процесів	10
1.2 Аналіз проблем інвентаризації та сучасних підходів до їх вирішення	16
1.3 Вплив робототехніки та цифрових технологій на ефективність складських си- стем	18
1.4 Висновок по розділу	27
РОЗДІЛ 2. МЕТОДИ ТА ІНСТРУМЕНТИ РОЗРОБКИ ПРОГРАМНОГО РІ- ШЕННЯ	29
2.1 Методи автоматизації інвентаризації та інтеграція QR-кодів	29
2.2 Формування та обробка набору даних для системи інвентаризації	32
2.3 Аналіз архітектури та програмної реалізації системи	39
2.4 Висновок по розділу	45
РОЗДІЛ 3. РЕАЛІЗАЦІЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ СИСТЕМИ ІНВЕ- НТАРИЗАЦІЇ	46
3.1 Реалізація функціональних можливостей програмної системи	46
3.2 Оцінка впливу системи на ефективність автоматизації складських процесів	53
3.3 Аналіз продуктивності та ефективності виявлення QR-кодів	64
3.4 Висновок по розділу	69
ВИСНОВКИ	71
СПИСОК ПОСИЛАНЬ НА ДЖЕРЕЛА	73

					БР.ІІ –17.00.00.000 ІІЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	Програмні рішення для автоматизації процесу інвентаризації на складах 3 Пояснювальна записка	Літ.	Арк.	Акрушіє
Розроб.		Химич І.						
Перевір.		Корнута В.А.					7	
Реценз.		Лютак І.З				ІФНТУНГ ІІ-22-3		
Н. Контр.		Піх М.М.						
Затверд.		Бандура В. В.						

ВСТУП

У сучасних умовах розвитку логістики та електронної комерції автоматизація складських процесів стає важливою складовою підвищення ефективності роботи підприємств. Постійне зростання обсягів товарообігу, необхідність швидкої обробки даних та мінімізації помилок під час обліку продукції вимагають впровадження сучасних інформаційних технологій у сферу складського управління. Одним із ключових процесів, що потребує оптимізації, є інвентаризація товарів, оскільки традиційні методи ручного обліку займають значний час, потребують великої кількості ресурсів і часто супроводжуються помилками через людський фактор. Використання програмних рішень, систем автоматичної ідентифікації та цифрових технологій дозволяє значно підвищити точність і швидкість виконання складських операцій.

Актуальність теми зумовлена необхідністю створення ефективних програмних рішень для автоматизації процесу інвентаризації на складах, які забезпечують швидкий облік товарів, зменшення витрат часу на перевірку залишків та підвищення точності даних. Існуючі системи інвентаризації часто мають обмежену функціональність або потребують дорогого спеціалізованого обладнання, що ускладнює їх впровадження для малих і середніх підприємств. Сучасні технології, зокрема QR-коди, засоби автоматизованої обробки даних та інтеграція програмних систем із складськими платформами, дозволяють створювати доступні та ефективні рішення для автоматизації складських процесів.

Метою роботи є розробка програмного рішення для автоматизації процесу інвентаризації на складах, яке забезпечує швидке виявлення та облік товарів, обробку інформації про складські залишки та підвищення ефективності управління складськими процесами.

Завданнями дослідження є аналіз предметної області та сучасних підходів до автоматизації складської інвентаризації, визначення функціональних і нефункціональних вимог до програмної системи, дослідження методів автоматичної ідентифікації товарів за допомогою QR-кодів, аналіз технологій обробки та зберігання даних, проєктування архітектури програмного рішення,

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

реалізація основних функціональних модулів системи, тестування роботи програмного забезпечення та оцінка ефективності його використання.

Об’єктом дослідження є процеси автоматизації інвентаризації на складах, що включають облік товарів, обробку даних, ідентифікацію продукції та контроль складських залишків.

Предметом дослідження є методи, технології та програмні засоби автоматизації інвентаризації, включаючи використання QR-кодів, алгоритми обробки даних, архітектурні підходи до побудови складських систем та інструменти розробки програмного забезпечення.

Методи дослідження включають аналіз наукових джерел і сучасних технологій автоматизації складських процесів, порівняльний аналіз існуючих систем інвентаризації, моделювання бізнес-логіки програмної системи, експериментальний метод для реалізації та тестування програмного забезпечення, а також аналіз продуктивності системи при роботі з QR-кодами та складськими даними.

Розроблене програмне рішення передбачатиме можливість автоматизованого обліку товарів, швидкого сканування QR-кодів, контролю складських залишків та формування звітності щодо інвентаризації. Особлива увага приділятиметься швидкодії системи, точності виявлення QR-кодів, зручності інтерфейсу та можливості інтеграції з іншими складськими платформами. Очікується, що впровадження програмного рішення дозволить зменшити витрати часу на проведення інвентаризації, підвищити точність обліку товарів та покращити ефективність управління складськими процесами.

Бакалаврська робота містить 66 сторінки, 30 рисунків, 3 розділи, список використаних джерел із 40 найменувань.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ АВТОМАТИЗАЦІЇ ПРОЦЕСІВ ІНВЕНТАРИЗАЦІЇ НА СКЛАДАХ

1.1 Основні принципи автоматизації складських процесів

Управління запасами на складі – це процес управління зберіганням, переміщенням та обробкою товарів на складі або в розподільчому центрі. Він відіграє вирішальну роль в управлінні ланцюгом поставок, забезпечуючи точність обліку запасів.

Ранні методи управління запасами були повністю ручними та спиралися на прості інструменти, такі як лічильні палички та глиняні жетони для відстеження товарів [1]. Лічильні палички, по суті, шматки дерева, використовувалися торговцями для підрахунку товарів на складі, тоді як глиняні жетони представляли різні види товарів і часто використовувалися бухгалтерами для контролю руху запасів в організаціях. Зі зростанням бізнесу та його ускладненням стали очевидними обмеження цих рудиментарних методів, що призвело до розробки більш досконалих систем.

Початок 20-го століття ознаменувався значним поворотним моментом в історії управління запасами з впровадженням механічних систем. Ці ранні системи використовували перфокарти та інші механічні пристрої для відстеження рівня запасів, що ознаменувало початок автоматизації в управлінні запасами. Поява комп'ютерів у 1950-х роках ще більше революціонізувала цю галузь, дозволивши підприємствам впроваджувати електронні системи управління запасами, які забезпечували більшу ефективність та можливість відстежувати запаси в режимі реального часу. До 1980-х років впровадження реляційних баз даних дозволило інтегрувати управління запасами з іншими ключовими бізнес-функціями, такими як бухгалтерський облік та управління взаємовідносинами з клієнтами (CRM), створюючи більш гнучкі та масштабовані системи.

У 1990-х та 2000-х роках з'явилися веб- та хмарні системи управління запасами, що ще більше трансформувало цю галузь. Інтернет дозволив

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

підприємствам отримувати доступ до даних про запаси з будь-якого місця, що полегшило співпрацю з постачальниками та клієнтами в глобальному масштабі. Хмарні обчислення зробили цей етап еволюції ще одним кроком, пропонуючи хмарні системи управління запасами, розміщені в Інтернеті, що зробило їх доступнішими, зручнішими та простішими у використанні, ніж традиційні локальні системи. Сьогодні підприємства будь-якого розміру можуть вибирати з широкого спектру систем управління запасами, адаптованих до їхніх конкретних потреб та бюджету, продовжуючи спадщину інновацій у цій галузі.

Управління запасами в роздрібних магазинах, особливо у великих мережевих магазинах, створює значні труднощі [2]. Традиційні методи відстеження запасів часто передбачають ручне сканування штрих-кодів товарів, що може займати багато часу та бути схильним до людських помилок. Цей процес особливо громіздкий на великих складах, таких як Walmart, де обсяг продукції та частота зміни запасів вимагають постійного моніторингу. Неефективність управління запасами може призвести до таких проблем, як дефіцит товарів, перевитрати та неточні записи про запаси, що зрештою впливає на загальну операційну ефективність та задоволеність клієнтів. Цей проект спрямований на вирішення цих проблем шляхом розробки автоматизованої системи, яка використовує QR-коди та методи комп'ютерного зору для відстеження запасів.

Кожен QR-код містить універсальний код продукту. Коли QR-код виявляється та обробляється, вбудований унікальний код точно витягується та ідентифікується, що сприяє точному відстеженню та управлінню продуктом.

QR-коди мають кілька ключових переваг порівняно з традиційними штрих-кодами, що робить їх особливо придатними для автоматизованої системи управління запасами на великих складах, таких як Walmart.

По-перше, QR-коди мають значно більшу ємність для зберігання даних порівняно зі штрих-кодами. Хоча типовий штрих-код може містити лише універсальний код продукту (UPC), QR-код може зберігати великий обсяг інформації, включаючи назву продукту, дату виробництва, термін придатності тощо. Ця розширена ємність дозволяє здійснювати більш детальне та всебічне

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

відстеження в системі управління запасами. Крім того, QR-коди пропонують перевагу багатонаправленого сканування. На відміну від штрих-кодів, які необхідно точно вирівняти зі сканером, QR-коди можна сканувати під будь-яким кутом. Ця гнучкість підвищує швидкість та ефективність процесів сканування, особливо в динамічному середовищі великого складу. Крім того, QR-коди бездоганно інтегруються з технологіями комп'ютерного зору та іншими автоматизованими системами, що дозволяє відстежувати в режимі реального часу та автоматично оновлювати записи про запаси. Ця інтеграція є життєво важливою для досягнення покращень автоматизації та точності, висвітлених у цій роботі. Нарешті, QR-коди універсальні за розміром, що дозволяє їх компактно друкувати, зберігаючи при цьому більшу ємність даних. Ця універсальність робить їх ідеальними для маркування широкого спектру товарів та розділів на складі.

Використовуючи ці переваги, QR-коди підвищують ефективність, точність та масштабованість систем управління запасами, ефективно усуваючи обмеження традиційних методів сканування штрих-кодів.

Замінивши традиційні штрих-коди QR-кодами та встановивши камери в кожній секції складу, можна захоплювати та аналізувати зображення для виявлення та декодування QR-кодів за допомогою алгоритмів комп'ютерного зору, що надаються бібліотекою OpenCV [3]. Декодована інформація потім використовується для оновлення рівня запасів та створення комплексних звітів, що значно зменшує потребу в ручному скануванні та підвищує точність і ефективність процесів управління запасами.

Запропонована система передбачає створення спеціальних QR-кодів для секцій складу та окремих товарів, інтеграцію цих QR-кодів у складське середовище та використання методів комп'ютерного зору для обробки зображень, отриманих камерою. Цей підхід спрямований на зниження витрат на оплату праці, покращення прийняття рішень та розподілу ресурсів, що зрештою призводить до ефективнішого управління запасами у великих роздрібних магазинах.

Метою цієї роботи є детальний опис розробки програмного забезпечення для розрахунку окружності колеса приводу навантаження, яке б автоматизувало

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

існуючий процес зазначеного розрахунку. Метою цієї дипломної роботи є створення та впровадження програмного забезпечення для розрахунку окружності колеса приводу навантаження, яке використовує інтерфейс прикладного програмування (API) відкритості повністю інтегрованої автоматизації (TIA), використовуючи як бізнес-знання, так і знання з програмування, отримані автором під час досліджень. Другою метою цієї дипломної роботи є окреслення важливості автоматизації в сучасний час, а також у недавньому минулому, а також викладення останніх розробок у галузі автоматизації управління складами [4].

Найважливішими попередніми знаннями для розуміння цієї роботи є розуміння TIA Portal та TIA Openness API. TIA Portal - це інженерний фреймворк та платформа для програмування та налаштування систем автоматизації. Він широко використовується в промисловій автоматизації для програмування програмованих логічних контролерів (ПЛК), пристроїв інтерфейсу людина-машина (HMI) та інших компонентів автоматизації. TIA Openness API - це API, що надається Siemens в екосистемі TIA Portal. Він дозволяє програмістам отримувати доступ до даних проектів TIA Portal та маніпулювати ними програмно.

Це практична функціональна робота. Збір даних для цієї роботи має переважно якісний характер, з використанням таких методів збору даних, як контент-аналіз, де інформація збирається з онлайн-джерел разом з літературою, а додаткова інформація збирається з тематичного дослідження. У практичній частині роботи експериментальний метод полягає в експериментуванні з можливостями TIA Openness API. Основна ідея цього експериментального тематичного дослідження полягає в тому, щоб навчитися на досвіді процесу розробки розрахункового застосунку, документуючи всі етапи його розробки.

Компанія, яка надала цій роботі можливість стажування та замовила програмне забезпечення, що є основою цієї роботи, промислова компанія, що спеціалізується на встановленні, ремонті та обслуговуванні промислового обладнання для заводів, складів та будь-яких інших підприємств, які потребують промислового обладнання. Як спеціаліст з міжнародного бізнесу, автор хотів би детально розповісти про свій досвід розробки та впровадження зазначеного програмного

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

забезпечення, спілкування та роботи з галузевими фахівцями, а також про розширення своїх знань про управління складами та загальну автоматизацію та її вплив на бізнес-операції.

Ця робота пропонує повний аналіз процесу розрахунку діаметра приводного вантажного колеса за допомогою розробленого програмного забезпечення, а також слугує документацією всього процесу розробки розрахункового застосунку [5]. Основна увага в цій роботі приділяється не стільки програмному аспекту створення застосунку, скільки опису всього процесу розробки, від співпраці з кінцевим користувачем до вивчення основ TIA Portal.

У розділі теоретичних основ розглядаються концепції, необхідні для правильного розуміння роботи. Наступні концепції є актуальними для цього дослідження, оскільки вони забезпечують як передумови, так і сучасний контекст автоматизації складів.

TIA Portal використовується в цій роботі на етапі навчання розробки застосунку для розуміння екосистеми TIA Portal. Він також є актуальним, оскільки використовується компанією-підрядником для програмування ПЛК, тому застосунок, розроблений як основа цієї дипломної роботи, повинен взаємодіяти з TIA Portal та проектами всередині нього. Тому для визначення та маніпулювання параметрами в проектах TIA Portal необхідні ґрунтовні знання TIA Portal.

TIA Portal, який використовується для повністю інтегрованої автоматизації, – це універсальна інженерна платформа Siemens, яка революціонізує ландшафт автоматизації. Вона спрощує складні процеси та оптимізує машинобудування, інтегруючи різні компоненти, такі як ПЛК, екрани НМІ, приводи зі змінною швидкістю та сервоприводи, в єдиний програмний пакет.

У минулому керування цими компонентами означало боротьбу з кількома програмними платформами, але TIA Portal змінив правила гри, пропонуючи єдиний інтерфейс [6]. Ця консолідація не лише підвищує ефективність, але й вводить нові функції, такі як підтримка кількох користувачів та управління живленням, що робить проекти промислової автоматизації більш спрощеними та доступними,

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

ніж будь-коли раніше.

API відкритості TIA широко використовується в цій роботі, оскільки це єдиний спосіб отримати властивості проекту TIA Portal та програмно маніпулювати ними. З огляду на те, що головною метою програми, як її замовила контактна особа в TP-Kunnossapito Oy, є взаємодія з проектами та властивостями TIA Portal всередині них, API відкритості TIA є невід'ємною частиною як проекту, так і цієї роботи.

Siemens TIA Openness являє собою API, вбудований у TIA Portal, який дозволяє взаємодіяти з платформою через користувацькі програми. Незважаючи на це включення, Openness ще не отримала широкого поширення, головним чином тому, що програмісти неохоче використовують іншу мову програмування, а саме C#, разом зі своїми навичками програмування ПЛК.

Відкритість TIA має кілька переваг, які можуть значно підвищити ефективність розробки ПЛК. Помітною функцією є розробка модульного коду, призначеного для пристроїв, який автоматично забезпечує модифікації, специфічні для конкретного застосування. Визначаючи базові шаблони, користувачі отримують можливість виконувати повноцінні завдання без навичок програмування ПЛК та НМІ, що спрощує впровадження та забезпечує кращу доступність.

Об'єктно-орієнтоване програмування є неминучим предметом у випадку цієї роботи. Об'єктно-орієнтоване програмування – це обчислювальна модель програмування, яка структурує розробку програмного забезпечення на основі сутностей даних, відомих як об'єкти, а не зосереджується виключно на функціях та процедурній логіці. Об'єкт у цьому контексті втілює структуру даних, що містить відмінні характеристики та функціональні можливості.

Під час фази навчання розробки застосунку було докладено значних зусиль для переходу до використання класів та об'єктно-орієнтованого програмування загалом. Оскільки калькулятор є складним, його програмування без використання об'єктно-орієнтованого програмування було б дуже нудним та трудомістким, що підштовхнуло автора до використання класів, зробивши об'єктно-орієнтоване програмування ключовою частиною цього проекту, а також цієї роботи [7].

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

1.2 Аналіз проблем інвентаризації та сучасних підходів до їх вирішення

Сама проблема, а також важливість проєкту були передані автору у формі інтерв'ю перед реєстрацією на перше стажування. Наступний розділ виконано у формі тематичного дослідження, що документує проєкт, а також обґрунтовує його розробку

На великих складських приміщеннях Walmart традиційний метод ручного сканування штрих-кодів передбачає фізичне сканування кожного товару. Цей процес вимагає значних людських зусиль і часу, стаючи дедалі громіздкішим і неефективнішим зі зростанням обсягу запасів. Такі трудомісткі методи призводять до затримок в оновленні запасів і підвищення витрат на робочу силу.

Ручне управління запасами схильне до людських помилок, включаючи не-правильне введення даних, пропущені сканування та неправильну ідентифікацію товарів. Ці помилки можуть спричиняти неточності в обліку запасів, що призводить до дефіциту товарів, перевитрати та розбіжностей між фізичними та зареєстрованими рівнями запасів.

Неточні записи інвентаризації

Залежність від ручного сканування та введення даних у Walmart часто призводить до неточних записів про запаси. Ця неточність впливає на операційну ефективність, спричиняючи дисбаланс запасів та ускладнюючи управління рівнем запасів.

Відсутність оновлень у режимі реального часу

Традиційні методи управління запасами, що використовуються Walmart, не забезпечують оновлення рівня запасів у режимі реального часу. Як наслідок, записи про запаси можуть не відображати поточний стан запасів, що призводить до затримки у прийнятті рішень та неоптимального розподілу ресурсів..

Проблема ефективності автоматизації складу

Основною проблемою, як зазначила компанія-наймач, була неефективність процесу розрахунку діаметра приводного колеса роботів. Крім того, існувало бажання спростити створення проєктів, що визначають елементи керування

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

роботом, можливо, шляхом створення легкозастосовного шаблону, проте цю частину загального проекту пізніше було вилучено під час розробки.

Проблема для роботів

Проблема для роботів полягала в тому, що щоразу, коли відбувалися зміни, виникала гостра потреба розрахувати діаметр ведучого колеса навантаження в проекті керування роботом [8]. Згадана зміна могла стосуватися будь-чого: від заміни старих ланцюгів або ременів, що змінювало б діаметр, оскільки новий ланцюг або ремінь у хорошому стані сиділи б на машині по-іншому, до встановлення нових деталей або простого зносу існуючих деталей.

Гостра потреба в перерахунку довжини кола приводного вантажного колеса впливає з потреби в точності, оскільки помилка в процесах управління складом може мати катастрофічні наслідки, що призведе до втрати прибутку або навіть ризиків для здоров'я людей.

Проблема для експертів з автоматизації

Постійна потреба в розрахунку окружності приводного колеса створює проблему для ефективності відділу автоматизації, оскільки постійно потрібно їхати на завод і перераховувати діаметр приводного колеса. Цей процес займає багато часу з кількох причин, а саме: необхідність щоразу одягатися в захисне спорядження, а потім знову перевдягатися в зручний одяг, додання відстані до заводу від офісу і назад, саме тестування та сам розрахунок, ручне отримання параметрів для розрахунку нового діаметра та ручне перезаписування діаметра та, можливо, інших змінених параметрів.

Переваги автоматизації таких трудомістких та повторюваних завдань можуть проявлятися як у самій ефективності діяльності компанії, так і в моральному настрої та світогляді співробітників. Компанія отримує велику користь від автоматизації нудних, потенційно небезпечних та трудомістких завдань, оскільки це звільняє час для співробітників для роботи над більш важливими та особистісно задовольняючими завданнями [9]. Крім того, важливо зазначити, що повторювані завдання потенційно можна виконувати швидше та ефективніше за умови автоматизації, що безпосередньо вигідно для компанії, підвищуючи ефективність

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

операцій. Однак співробітники отримують користь від автоматизації рутинних завдань, яка відрізняється від такої для організації в цілому, оскільки ці нудні завдання не залучають співробітника, який, змушений їх виконувати, може втратити моральний дух, що може призвести до зниження ефективності. Автоматизація цих завдань дозволяє співробітнику зосередитися на більш захопливих завданнях, що призводить до підвищення морального духу та підвищення продуктивності та загального довгострокового благополуччя співробітника, що, знову ж таки, вигідно для роботодавця.

1.3 Вплив робототехніки та цифрових технологій на ефективність складських систем

У цьому підрозділі заглиблюється в трансформаційний вплив робототехніки та автоматизації на складські операції, висвітлюючи переваги та проблеми, пов'язані з цими технологіями. У ньому обговорюється зростання робототехніки, роль штучного інтелекту (ШІ) в оптимізації складів та майбутні тенденції, з акцентом на ключові особливості та переваги, вплив на ефективність та поточну технологічну революцію в логістиці [10].

Склади, ключові компоненти глобальних ланцюгів поставок, зазнають значних трансформацій завдяки інтеграції робототехніки та технологій автоматизації. Це впровадження забезпечує підвищення ефективності, покращення точності, економію коштів та підвищення безпеки складських операцій.

Переваги робототехніки та автоматизації на складах: у складських операціях робототехніка та автоматизація дозволяють точно та швидко виконувати повторювані завдання, що призводить до підвищення ефективності комплектування. Запрограмовані на високу точність, роботи зменшують кількість помилок, підвищуючи точність виконання замовлень та задоволеність клієнтів.

Незважаючи на початкові інвестиційні витрати, автоматизація призводить до довгострокової економії, а компанії повідомляють про зниження операційних витрат. Автоматизація також сприяє підвищенню безпеки робочої сили,

					БР.ІП - 17.00.00.000 ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

зменшуючи небезпеки, автоматизуючи завдання, визначаючи реальне місцезнаходження працівників та мінімізуючи травми.

Системи на базі штучного інтелекту ще більше революціонізують складські операції, забезпечуючи інтелектуальне прийняття рішень та адаптивність. Алгоритми машинного навчання аналізують дані, оптимізують процеси та прогнозують попит, що призводить до ще більшого підвищення ефективності.

Проблеми впровадження робототехніки та автоматизації включають початкові витрати, проблеми адаптації робочої сили, складну інтеграцію з існуючими системами, а також необхідність проактивного обслуговування та надійної технічної підтримки. Ретельний аналіз витрат і вигод, гнучкі моделі ціноутворення, прозора комунікація, навчання, залучення досвідчених консультантів та проактивні стратегії обслуговування мають вирішальне значення для вирішення цих проблем.

Майбутні тенденції включають розробку колаборативних роботів (коботів), призначених для роботи разом з людьми, впровадження автономних мобільних роботів (AMR) для гнучких та динамічних операцій, використання аналітики великих даних для оптимізованого управління запасами та інтеграцію Інтернету речей (IoT) для передачі даних у режимі реального часу та прогнозного обслуговування.

Інтеграція робототехніки та технологій автоматизації відкриває трансформаційні можливості для складів, пропонуючи ефективність, точність та економію коштів. З розвитком логістичної галузі, відстеження нових тенденцій та подолання викликів матиме вирішальне значення для підприємств, які прагнуть процвітати в сучасному ландшафті ланцюгів поставок.

Автоматизовані складські роботи: типи та переваги

Нещодавні дані свідчать про те, що приблизно 20% складів впровадили певну форму робототехніки, причому лише Amazon може похвалитися понад 500 000 активних роботів на своїх об'єктах [11]. Ці дані підкреслюють сучасну реальність того, що складські роботи є невід'ємною частиною сучасного ланцюга поставок, а не просто футуристичною концепцією.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

Зростаюча популярність автоматизованих складських роботів на сучасних складах може бути пов'язана з різними факторами. Використання роботів у складських процесах допомагає зменшити залежність від людської праці, що призводить до більш економічно ефективного, продуктивного та безпечного об'єкта. Ці переваги є лише частиною загальних переваг, які пропонують автоматизовані складські роботи.

Щоб повністю зрозуміти, що собою являють автоматизовані складські роботи та як вони можуть оптимізувати складські операції, важливо глибше зануритися в їхні функціональні можливості. За словами Ерхана Мусаоглу (2023): ось ключові висновки, які слід зрозуміти про автоматизовані складські роботи:

Мінімальне втручання людини: Автоматизовані складські роботи розроблені як роботизовані системи для виконання складських завдань з мінімальним втручанням людини або без нього. Цей аспект значно зменшує потребу в ручній праці та мінімізує людські помилки, сприяючи більш ефективній та надійній роботі [12].

Різноманітність рішень: Існують різні рішення для складських роботів, і вибір правильного для вашого об'єкта залежить від конкретних складських процесів та цілей. Вибір роботів повинен відповідати характеру завдань, плануванню складу та бізнес-цілям для досягнення оптимальної ефективності.

Типи складських роботів: До поширених типів складських роботів належать автоматизовані мобільні роботи (AMR), роботи для доставки товарів до людини (G2P), дрони та інші. Кожен тип робота краще підходить для різних типів завдань, використання кількох типів роботів на одному об'єкті може призвести до підвищення ефективності.

Безперервність роботи: Автоматизовані роботи здатні підтримувати робочий процес складу цілодобово, підвищуючи продуктивність та прибутковість. Безперервна робота гарантує, що такі завдання, як сортування, пакування та транспортування, можуть виконуватися без перерв, тим самим пришвидшуючи весь процес ланцюга поставок.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

Інтеграція із системами управління складом (WMS): Інтеграція комплексної системи управління складом (WMS) має вирішальне значення для максимального використання потенціалу складських робіт. WMS сприяє координації та контролю всіх видів роботизації, забезпечуючи безперебійну роботу, точне управління запасами та ефективне виконання замовлень.

Автоматизовані складські роботи розроблені для виконання низки складських операцій, включаючи комплектування, пакування, сортування, зберігання та маркування, мінімізуючи участь людини [13]. Ці роботи використовують датчики та камери для картографування та переміщення складських приміщень, оснащені маніпуляторами для обробки товарів.

У співпраці із системами управління складом (WMS), такими як Logiwa, сучасні автоматизовані складські роботи використовують дані з підключених систем. WMS призначає завдання та програмує робітників для автономного виконання процесів. Моніторинг та відстеження продуктивності складських робіт також можна ефективно здійснювати за допомогою WMS.

Існує чотири основні переваги складських робіт. Ці переваги перелічені нижче:

Зниження витрат: на відміну від працівників-людей, яким потрібна заробітна плата та пільги, роботи працюють без заробітної плати, що пропонує економічно ефективне рішення для зменшення операційних витрат на склад.

Підвищена ефективність: складські роботи, будучи невтомними, зменшують ймовірність помилок, пов'язаних з втомою людини. Автоматизація повторюваних або складних завдань підвищує ефективність та продуктивність.

Підвищена безпека: автономні мобільні роботи можуть обробляти важкі вантажі без втручання людини, що зменшує ризик нещасних випадків, пов'язаних з небезпечним обладнанням або важкими предметами.

Дані та аналітика в режимі реального часу: автоматизовані роботи надають практичну інформацію для оптимізації складських операцій. Інтеграція з WMS, такою як Logiwa, забезпечує легкий доступ до найновіших даних та аналітики через зручну панель керування.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

Автономно керовані транспортні засоби (AGV)

AGV – це мобільні роботи, запрограмовані для перевезення товарів у межах складів. Працюючи автономно за заздалегідь визначеними маршрутами за допомогою лазерів, датчиків або магнітів, AGV пропонують безпечнішу альтернативу вилковим навантажувачам, мінімізуючи втручання людини.

Автоматизовані мобільні роботи (AMR)

На відміну від AGV, AMR вільно переміщуються по складах, забираючи та доставляючи товарні запаси. Їхня автономність та гнучкість дозволяють їм легко адаптуватися до змін у плануванні складів, підвищуючи операційну ефективність.

Роботи для комплектування та пакування

Роботизовані маніпулятори, запрограмовані для отримання та підготовки замовлень, оснащені передовими датчиками та алгоритмами. Роботи для комплектування та пакування підвищують продуктивність складу, обробляючи замовлення безперервно, вдень і вночі, перевершуючи людську швидкість та ефективність.

Коботи (колаборативні роботи)

Розроблені для роботи разом із людьми, коботи допомагають у таких завданнях, як комплектування, підйом, контроль якості, пакування та відвантаження. Оснащені датчиками для запобігання зіткненням, коботи підвищують точність виконання замовлень.

Роботи для палетування

Автоматизуйте штабелювання та упорядкування товарів на піддонах, оптимізуючи простір для зберігання, підвищуючи безпеку та мінімізуючи ризик пошкодження товарів. Палетування саме по собі - це практика переміщення продуктів на піддон для подальшого транспортування або зберігання.

Роботи для автоматичної системи зберігання та пошуку (ASRS)

Роботи ASRS автоматизують зберігання, пошук та обробку складських товарів, оптимізуючи простір для зберігання та підвищуючи ефективність роботи запасів за допомогою таких методів, як FIFO або LIFO. Метод «першим вийшов»

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						23
Змн.	Арк.	№ докум.	Підпис	Дата		

(FIFO) передбачає, що першою продається найдавніша одиниця запасу або перша одиниця, що надходить на склад. Метод «Останнім прийшов, першим вийшов» (LIFO) передбачає, що першою продається остання або найновіша одиниця, додана до складу.

Роботи для доставки товарів до людини (G2P)

Витягувати товари зі складу та доставляти їх пакувальникам для виконання замовлень, мінімізуючи час ходьби людини та максимізуючи ефективність. Такі роботи зазвичай використовуються в центрах виконання замовлень електронної комерції.

Дрони

Безпілотні літальні апарати використовуються для картографування, спостереження, відстеження товарів, огляду важкодоступних місць, оновлення обліку запасів, перевірки рівня запасів, а також моніторингу працівників і великих просторів на складах. Їх не слід плутати з комерційними дронами, які потребують безпосереднього керування людиною, тоді як ці дрони здебільшого автономні.

Ці автоматизовані роботи разом сприяють покращенню складських операцій, забезпечуючи ефективність, безпеку та адаптивність. Все вищезазначене сприяє постійно зростаючому інтересу до автоматизації та, зокрема, до роботів.

Як покращити складські операції за допомогою роботів та автоматизації. Критика часто спрямована на процес складування через пов'язані з ним витрати, що спонукає деяких людей розглядати тенденцію до аутсорсингу як спосіб обійти складнощі, пов'язані з аналізом внутрішніх логістичних витрат. Окрім очевидних витрат, таких як запаси, технології, робоча сила та операції, зберігання запасів створює труднощі та непрямі витрати. Для досягнення ефективних систем розподілу товарів з мінімізацією запасів необхідні структуровані та прозорі процеси, що сприяють системам управління складом (WMS). Вирішальним аспектом ефективної WMS є формування довіри до системи управління та контролю, тим самим зменшуючи надмірні страхові запаси, спричинені управлінською невизначеністю. Покращення безпеки даних та прозорості в WMS сприяє постійній оптимізації системи, підвищуючи керованість та управліність процесами, а також

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						24
Змн.	Арк.	№ докум.	Підпис	Дата		

збільшуючи час реагування та гнучкість. Інтерфейси з підпорядкованими системами забезпечують адаптацію до змінних структур.

Складські операції можна значно покращити завдяки інтеграції роботів та автоматизації. Ось кілька способів, як ці технології можуть підвищити ефективність, точність та загальну продуктивність на складі:

Більше про роботів-палетизаторів та їхній вплив

Роботизоване палетування передбачає використання промислових роботів для укладання та штабелювання продуктів на піддони для відвантаження. Різноманітні роботи можуть виконувати завдання палетування, найпоширенішим методом є використання спеціалізованих роботів-палетизаторів, призначених для швидкої обробки великих, важких предметів.

Роботи-палетизатори виконують подвійну функцію: позбавляють людей від виснажливих та фізично вимогливих завдань, які потребують складних допоміжних інструментів, що призводить до тривалого часу обробки. На відміну від повністю автоматизованих спеціалізованих установок, ці роботи пропонують більшу гнучкість та економічну ефективність, особливо при виконанні менших робочих навантажень. До поширених типів роботів, що використовуються для таких завдань, належать порталні та шарнірні роботи. Роботи-палетизатори оснащені роботизованою рукою та захоплювачем продукту, що дозволяє їм точно піднімати та позиціонувати предмети, плавно підлаштовуючись під зміни розміру та ваги. Ці роботи використовують датчики та розширене програмування для оцінки розміщення кожного предмета, забезпечуючи безпечне розташування. Завдяки автоматизації цих завдань ці роботи не тільки економлять час операторів, але й мінімізують ризик пошкодження предметів та оптимізують місткість сховища.

Підвищення ефективності комплектування замовлень

Роботи відіграють певну роль у завданнях комплектування замовлень, або встановлені на стелажних подавачах для вилучення товарів з полиць, або працюючи в окремих зонах, де вони збирають більші стандартизовані одиниці зі штабелів піддонів. Ефективність та продуктивність цих роботів значною мірою залежать від якості наданих пристроїв захвату. Чим точніше надання цих пристроїв,

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

тим плавніше інтегрується робот у загальну систему. Однак ці системи висувають значні вимоги до точного позиціонування, зняття та обробки товарів, що призводить до трудомістких та дорогих процесів.

Ефективність логістики

Ефективна логістика та управління складом є ключовими аспектами ланцюга поставок компанії. Оптимізація складських операцій не лише скорочує витрати, але й підвищує задоволеність клієнтів завдяки швидкій та точній доставці. Мобільні роботи стали цінними інструментами, що дозволяють автоматизувати логістику. Це включає програмне забезпечення для управління складами, автоматизовані конвеєрні системи та універсальні мобільні роботи, здатні виконувати такі завдання, як транспортування товарів та організація запасів [14]. Ці автономні мобільні роботи невинно працюють у спільних середовищах, зменшуючи кількість помилок, скорочуючи витрати та підвищуючи продуктивність складу.

Наприклад, роботи, що підвищують ефективність логістики в WMS, – це автоматизовані керовані транспортні засоби (AGV) та автоматизовані керовані візки (AGC).

AGV (автономні мобільні роботи) – це роботизовані системи, що використовуються на складах і дотримуються заздалегідь визначених шляхів за допомогою датчиків, маркерів і лазерів для забезпечення точної навігації. На відміну від автономних мобільних роботів (AMR), AGV покладаються на фіксовані маршрути та вказівки операторів складу для своєї роботи, що обмежує їхню адаптивність до неочікуваних змін у навколишньому середовищі.

AGC – це роботи, призначені для транспортування запасів, використовуючи заздалегідь визначені шляхи. Оснащені магнітною стрічкою, мітками радіочастотної ідентифікації (RFID) та датчиками, AGC переміщуються по складу та виявляють перешкоди. На відміну від AGV, AGC обмежені своїм нижчим профілем та меншим шасі, що робить їх придатними лише для перевезення легших вантажів.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		

Зі зростанням автоматизації в галузі знову зростає потреба в більш сучасних складських приміщеннях [15]. Хоча переваги автоматизації широко визнані, виправдання змін у порівнянні з існуючими методами може бути складним завданням. Спочатку висока вартість впровадження повністю автоматизованого рішення часто відлякує компанії від участі. Щоб виправдати інвестиції в автоматизацію, компанії повинні розуміти її загальний вплив на їхній бізнес. Згідно з JR Automation, ось п'ять способів, якими автоматизація може принести бізнес-вигоди, позитивно впливаючи на прибуток:

Вирішення кадрових проблем: Незалежно від того, чи йдеться про проблеми з наймом чи звільненням, чи про нестачу кваліфікованих працівників, кадрові обмеження можуть перешкоджати продуктивності. Автоматизація може усунути прогалини, дозволяючи бізнес-підрозділам виконувати критично важливі завдання, коли робочої сили не вистачає.

Зменшення експлуатаційних витрат : Автоматизація зменшує ймовірність неправильного використання або пошкодження матеріалів, зменшує невідповідності в інвентаризації та зменшує помилки комплектування та упаковки, що заощаджує час і гроші, а також пришвидшує виконання завдань, які в іншому випадку зайняли б багато часу.

Оптимізація управління запасами: Вартість переміщення та зберігання запасів може бути значною, особливо зі збільшенням обсягу замовлень. Ручне управління запасами вимагає багато годин роботи, тоді як роботизовані системи, такі як AGV або AMR, забезпечують послідовне та безперервне переміщення матеріалів, зменшуючи трудомісткі завдання. Підвищення ефективності та продуктивності: Завдяки аутсорсингу повторюваних завдань автоматизації, співробітники можуть зосередитися на складних та цікавих завданнях, підвищуючи продуктивність [16]. Автоматизація забезпечує надійний робочий процес із цілодобовою пропускнуою здатністю та дозволяє швидше обробляти та виконувати замовлення.

Зменшення ризиків для безпеки: Автоматизація може замінити людей у діяльності, яка схильна до травм або ергономічного стресу. Робота, пов'язана з

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

повторюваним переміщенням або підняттям важких предметів , створює ризик для персоналу, який можна зменшити за допомогою автоматизації, що зменшить ймовірність травматизму на виробництві та забезпечить безперебійну роботу складу.

Зменшення помилок: швидкість і точність вибору замовлень значно впливають на задоволення клієнтів. Автоматизація процесів покращує показники комплектації та зменшує кількість помилок у виборі, уникаючи помилок, спричинених втомою людини. На відміну від людей, роботи розроблені для точності та прецизійності.

На завершення, автоматизація має потенціал трансформувати складські операції, якщо її впровадити правильно. Стратегічно використовуючи дані та автоматизацію, можна досягти значного підвищення продуктивності. Наша команда готова супроводжувати вас на цьому шляху, допомагаючи вибрати та інтегрувати правильні рішення для автоматизації, які відповідають потребам вашого бізнесу.

1.4 Висновок по розділу

У першому розділі розглянуто теоретичні основи автоматизації процесів інвентаризації на складах. Проаналізовано принципи побудови автоматизованих складських систем, типові проблеми ручної інвентаризації та сучасні підходи до їх усунення. Окремо розглянуто вплив цифрових технологій і робототехніки на підвищення точності та швидкості обліку товарів. Отримані результати формують теоретичну базу для розробки програмних рішень у сфері складської автоматизації.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 2. МЕТОДИ ТА ІНСТРУМЕНТИ РОЗРОБКИ ПРОГРАМНОГО РІШЕННЯ

2.1 Методи автоматизації інвентаризації та інтеграція QR-кодів

Традиційні штрих-коди, що використовуються в управлінні запасами Walmart, замінено QR-кодами, які пропонують розширену ємність для зберігання інформації та краще працюють за різних умов сканування. Кожному продукту та секції складу Walmart присвоюється унікальний QR-код, що дозволяє точно та детально ідентифікувати товари та їх місцезнаходження.

Методи комп'ютерного зору

Система використовує передові алгоритми комп'ютерного зору для виявлення та декодування QR-кодів із зображень, отриманих у складському середовищі Walmart [17]. OpenCV, провідна бібліотека комп'ютерного зору, використовується для забезпечення точного та надійного виявлення та декодування QR-кодів із зображень складу.

Автоматизована обробка зображень

Стратегічне розміщення камер по всьому складу дозволяє безперервно знімати зображення складських секцій. Ці зображення обробляються в режимі реального часу для ідентифікації та декодування QR-кодів, тим самим автоматизуючи процес відстеження запасів та усуваючи необхідність ручного сканування персоналом складу.

Обробка даних та звітність

Після виявлення та декодування QR-кодів система обробляє дані для оновлення записів про запаси Walmart. Вона створює детальні та вичерпні звіти та візуалізації, включаючи звіт про оброблені продукти, звіт про непрочитані продукти, звіт про непрочитані розділи та діаграми кількості запасів. Ця автоматизована система звітності підвищує прозорість управління запасами та надає цінну інформацію про розподіл та стан запасів.

Вплив та переваги

					БР.ІП - 17.00.00.000 ПЗ	Арк.
						29
Змн.	Арк.	№ докум.	Підпис	Дата		

Очікується, що впровадження автоматизованої системи управління запасами принесе такі переваги:

Зниження витрат на оплату праці: Автоматизація зменшує потребу в ручній праці, що призводить до значної економії коштів.

Підвищена точність: Мінімізуючи людські помилки, пов'язані з ручним скануванням та введенням даних, система забезпечує точніший облік запасів. Підвищена ефективність: Автоматизований процес прискорює управління запасами, скорочуючи час, необхідний для відстеження та оновлення запасів. Покращене прийняття рішень: Оновлення в режимі реального часу та комплексні звіти надають цінну інформацію для кращого розподілу ресурсів та управління запасами [18]. Інтеграція з іншими системами: Інтеграція автоматизованої системи управління запасами з різними іншими системами може значно підвищити її функціональність та ефективність. Підключаючись до систем планування ресурсів підприємства (ERP), система управління запасами може досягти синхронізації даних у режимі реального часу та автоматизувати процеси повторного замовлення, оптимізуючи робочі процеси та підтримуючи оптимальний рівень запасів. Інтеграція із системами управління складом (WMS) покращить роботу складу та точність відстеження. Підключення до систем управління взаємовідносинами з клієнтами (CRM) забезпечує оновлення запасів у режимі реального часу для відділів продажів та обслуговування клієнтів, покращуючи обслуговування клієнтів та пропонує цінну інформацію про продажі. Синхронізація з системами точок продажу (POS) забезпечує негайне оновлення рівня запасів та сприяє детальній аналітиці продажів. Інтеграція з платформами електронної комерції допомагає запобігти перепродажам та оптимізувати виконання замовлень. Крім того, підключення до систем управління ланцюгами поставок (SCM) та управління транспортом (TMS) покращує видимість ланцюга поставок та ефективність логістики. Інтеграція систем фінансового управління підтримує точну фінансову звітність та аналіз витрат, тоді як інтеграція систем управління персоналом (HRM) допомагає в ефективному управлінні робочою силою та дотриманні вимог. Нарешті, інтеграція інструментів аналітики та бізнес-аналітики (BI) дозволяє використовувати розширений аналіз

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

даних та налаштовувати інформаційні панелі, що сприяє кращому прийняттю рішень та стратегічному плануванню.

Налаштування середовища

Мета: Переконатися, що необхідне програмне забезпечення та бібліотеки встановлені, а каталог проекту структурований правильно.

Заходи: Встановіть необхідні бібліотеки Python (pandas, matplotlib, opencv-python). Упорядкуйте каталог проекту за логічними папками (наприклад, initialDataset, results, reports).

Обробка зображень

Мета: Виявлення та декодування QR-кодів зі зображень складу.

Заходи: Колекція зображень: Зберіть та помістіть усі відповідні зображення в каталог initialDataset/images/.

Виявлення QR-кодів: Завантажте кожне зображення за допомогою OpenCV. Виявляйте та декодуйте QR-коди за допомогою QRCodeDetector від OpenCV. Розрізняйте QR-коди розділів та QR-коди товарів. Зберігайте виявлену інформацію у структурованому форматі.

Обробка даних

Мета: Завантаження та керування даними, необхідними для створення звіту.

Заходи: Завантаження даних JSON: Завантажте інформацію про продукт з productsList.json. Завантажте деталі секції складу з warehouseSections.json.

Зберегти дані JSON: Збережіть оброблені дані QR-коду у файлі results/imageProcessingResult.json.

Генерація звітів

Мета: Аналіз оброблених даних та створення звітів про стан запасів.

Заходи: Створити звіт про оброблені продукти: Зіставте виявлені QR-коди товарів зі списком товарів. Збережіть інформацію про зіставлені товари у файлі Excel (reports/processed_products.xlsx).

Згенерувати звіт про непрочитані товари:

Визначте товари у списку товарів, яких не було у складських секціях.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

Збережіть інформацію про непрочитані товари у файлі Excel (reports/unread_products.xlsx).

Згенерувати звіт про непрочитані розділи: Визначте секції складу, які не були виявлені на жодному зображенні. Збережіть деталі непрочитаних розділів у файлі Excel (reports/unread_sections.xlsx).

Візуалізація даних

Мета: Створення візуальних представлень даних інвентаризації для полегшення аналізу та прийняття рішень.

Заходи: Створити діаграму інвентаризації: Завантажте дані про оброблені продукти з файлу Excel. Використайте Matplotlib для створення стовпчастої діаграми, яка показує кількість коробок товарів за секціями. Збережіть діаграму як файл зображення (reports/inventory_count_by_section.jpg).

2.2 Формування та обробка набору даних для системи інвентаризації

Файл «initialDataset/ProductsList.json» містить детальну інформацію про 514 товарів у Walmart. Кожен запис про товар містить такі важливі атрибути, як назва товару, вартість одиниці, UPC. Назва товару позначає назву товару, тоді як вартість одиниці вказує на ціну за окрему одиницю. UPC, або універсальний код товару, служить унікальним ідентифікатором для кожного товару. Ці дані необхідні для створення QR-кодів та полегшення точної ідентифікації в системі управління запасами. Ця інформація є ключовою для забезпечення ефективного відстеження та управління кожним товаром протягом усього процесу інвентаризації.

Складські секції

Файл «initialDataset/warehouseSections.json» перераховує різні розділи на складі Walmart, кожен з яких ідентифікується унікальним кодом розділу. Ці розділи структуровані як масив об'єктів JSON, де кожен об'єкт містить ключ «section», що вказує код розділу. Розділи організовані послідовно від «S101-001-001» до «S101-001-062» див. рис. 2.1.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						32
Змн.	Арк.	№ докум.	Підпис	Дата		

```
[
  {
    "Vendor Stk Nbr": "116113",
    "Item Status": "A",
    "Acct Dept Nbr": 92,
    "Item Nbr": 30364361,
    "UPC": "0088428222962",
    "Product title": "ALOOATTA DIWALI NUTS GIFT PACK",
    "Size Desc": "390",
    "WHPK Qty": 20,
    "Unit Cost": 7.25,
    "Unit Retail": 7.97,
    "MU %": 0.090339,
    "WHPK Cost": 145
  },
  ...
]
```

Рисунок 2.1 - Зразок параметрів товару

Використання цих унікальних кодів дозволяє систематично організувати склад, полегшуючи точне визначення місцезнаходження та ідентифікацію певних зон. Сегментуючи склад на окремі секції, ці коди забезпечують структуроване та впорядковане середовище. Такий рівень організації є критично важливим для ефективного управління запасами, оскільки дозволяє швидко та точно ідентифікувати різні зони складу. Отже, цей набір даних є невід'ємною частиною підтримки безперебійної та ефективної роботи складу див. рис 2.2.

```
[
  {
    "section": "S101-001-001"
  },
  ...
]
```

Рисунок 2.2 - Зразок розділу

Папка зображень

Папка «initialDataset/images» містить 58 файлів зображень, що документують різні секції складу Walmart. Кожне зображення відображає певну секцію складу та містить QR-коди, що використовуються для автоматичної ідентифікації та відстеження. Ці зображення мають вирішальне значення для здатності системи виявляти та декодувати QR-коди, забезпечуючи точне та ефективне оновлення складських записів у режимі реального часу без ручного сканування див.рис. 2.3.

					БР.ІП - 17.00.00.000 ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		



Рисунок 2.3 - Зразок зображення

Сама проблема, а також важливість проєкту були передані автору у формі інтерв'ю перед реєстрацією на перше стажування. Наступний розділ виконано у формі тематичного дослідження, що документує проєкт, а також обґрунтовує його розробку [19].

Проблема ефективності автоматизації складу

Основною проблемою, як зазначила компанія-наймач, була неефективність процесу розрахунку діаметра приводного колеса роботів. Крім того, існувало бажання спростити створення проєктів, що визначають елементи керування роботом, можливо, шляхом створення легкозастосовного шаблону, проте цю частину загального проєкту пізніше було вилучено під час розробки.

Проблема для роботів

Проблема для роботів полягала в тому, що щоразу, коли відбувалися зміни, виникала гостра потреба розрахувати діаметр ведучого колеса навантаження в проєкті керування роботом. Згадана зміна могла стосуватися будь-чого: від заміни старих ланцюгів або ременів, що змінювало б діаметр, оскільки новий ланцюг або ремінь у хорошому стані сиділи б на машині по-іншому, до встановлення нових деталей або простого зносу існуючих деталей.

Гостра потреба в перерахунку довжини кола приводного вантажного колеса впливає з потреби в точності, оскільки помилка в процесах управління складом може мати катастрофічні наслідки, що призведе до втрати прибутку або навіть ризиків для здоров'я людей.

Проблема для експертів з автоматизації

Постійна потреба в розрахунку окружності приводного колеса створює

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						34
Змн.	Арк.	№ докум.	Підпис	Дата		

проблему для ефективності відділу автоматизації, оскільки постійно потрібно їхати на завод і перераховувати діаметр приводного колеса. Цей процес займає багато часу з кількох причин, а саме: необхідність щоразу одягатися в захисне спорядження, а потім знову перевдягатися в зручний одяг, додання відстані до заводу від офісу і назад, саме тестування та сам розрахунок, ручне отримання параметрів для розрахунку нового діаметра та ручне перезаписування діаметра та, можливо, інших змінених параметрів [20].

Переваги автоматизації таких трудомістких та повторюваних завдань можуть проявлятися як у самій ефективності діяльності компанії, так і в моральному настрої та світогляді співробітників. Компанія отримує велику користь від автоматизації нудних, потенційно небезпечних та трудомістких завдань, оскільки це звільняє час для співробітників для роботи над більш важливими та особистісно задовольняючими завданнями. Крім того, важливо зазначити, що повторювані завдання потенційно можна виконувати швидше та ефективніше за умови автоматизації, що безпосередньо вигідно для компанії, підвищуючи ефективність операцій. Однак співробітники отримують користь від автоматизації рутинних завдань, яка відрізняється від такої для організації в цілому, оскільки ці нудні завдання не залучають співробітника, який, змушений їх виконувати, може втратити моральний дух, що може призвести до зниження ефективності. Автоматизація цих завдань дозволяє співробітнику зосередитися на більш захопливих завданнях, що призводить до підвищення морального духу та підвищення продуктивності та загального довгострокового благополуччя співробітника, що, знову ж таки, вигідно для роботодавця [21].

2.3 Аналіз архітектури та програмної реалізації системи

Згідно рис. 2.4 основного коду бачимо

Крок 1: Обробка зображень для виявлення QR-кодів. Код: `imageProcessing.py`

Функція: `'process_images(папка_з_зображеннями, output_json)'`

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

Мета: Виявлення QR-кодів на зображеннях, що зберігаються у вказаній папці, та збереження даних у файл JSON.

```
# main.py
from imageProcessing import process_images
from jsonHandling import load_json_data
from reportGeneration import (
    generate_read_products_report,
    generate_unread_products_report,
    generate_unread_sections_report,
    generate_inventory_count_chart
)

# Step 1: Process images to detect QR codes and save data to JSON
process_images('initialDataset/images', 'results/imageProcessingResult.json')

# Step 2: Load the saved QR code data and JSON data for products and warehouse sections
image_processing_result = load_json_data('results/imageProcessingResult.json')
products_list = load_json_data('initialDataset/productsList.json')
warehouse_sections = load_json_data('initialDataset/warehouseSections.json')

# Step 3: Generate reports
generate_read_products_report(image_processing_result, products_list)
generate_unread_products_report(image_processing_result, products_list)
generate_unread_sections_report(image_processing_result, warehouse_sections)

# Step 4: Generate inventory count chart
generate_inventory_count_chart()
```

Рисунок 2.4 - Основний код

Процес:

Колекція файлів зображень: Бібліотека `glob` використовується для отримання списку всіх файлів зображень із розширенням `.jpg` з вказаної папки. Це дозволяє пакетну обробку зображень, що зберігаються у вказаному каталозі.

Виявлення QR-кодів: Для кожного файлу зображення викликається функція `detect_qr_codes`. Ця функція обробляє зображення, щоб ідентифікувати та декодувати QR-коди, що містяться в ньому.

Класифікація даних: Виявлені QR-коди поділяються на дві категорії:

Код секції: Позначає секцію складу або складської зони, де було зроблено зображення.

QR-коди продуктів: Представляють унікальні QR-коди, пов'язані з продуктами.

Зберігання даних: Результати для кожного зображення об'єднуються у список словників. Кожен словник містить:

`image_file`: Ім'я файлу обробленого зображення.

`розділ`: Код розділу, витягнутий із зображення.

`qrcode`: Список декодованих QR-кодів, знайдених на зображенні.

					БР.ІП - 17.00.00.000 ПЗ	Арк.
						36
Змн.	Арк.	№ докум.	Підпис	Дата		

Збереження результатів: Зібрані дані зберігаються у файл JSON за допомогою функції `save_to_json`. Цей файл JSON містить усі виявлені QR-коди, класифіковані за відповідними файлами зображень та розділами.

Цей результат демонструє успішну обробку зображень для виявлення та категоризації QR-кодів, які потім систематично зберігаються в

«`results/imageProcessingResult`» – це формат JSON для подальшого аналізу та інтеграції. Він включає файл зображення, код секції та UPC. Файл зображення показує адресу каталогу зображень. Код секції визначає конкретну секцію складу, де розташовані продукти, а UPC показують унікальні коди продукту для ідентифікації кожної конкретної коробки продукту. Цей структурований формат полегшує подальший аналіз та інтеграцію в систему. Процес гарантує, що дані впорядковані та готові до наступних кроків. Ці впорядковані дані можна використовувати для покращення загальної системи управління запасами [22].

Крок 2: Завантаження даних JSON

Код: `jsonHandling.py`

Функції: `'load_json_data(json_path)'`

Мета: Завантаження JSON-даних з файлів для подальшої обробки.

Процес: Функція зчитує: `'results/imageProcessingResult.json'`, `'initialDataset/productsList.json'` та `'initialDataset/warehouseSections.json'`, і повертає їх як словник Python рис. 2.5.

Крок 3: Генерація звітів. Код: `reportGeneration.py`

Функція 1: `'generate_read_products_report(результат_обробки_зображень, список_продуктів)'`

Мета: Сформувати звіт про прочитані продукти та зберегти його у форматі Excel.

Процес: Для кожного запису QR-коду зіставте QR-коди продукту зі списком продуктів. Створіть список товарів, що відповідають запиту, включаючи коди їхніх розділів та інші відповідні деталі. Збережіть його як файл Excel.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						37
Змн.	Арк.	№ докум.	Підпис	Дата		

Функція 2:

'generate_unread_products_report(результат_обробки_зображення,
список_продуктів)'

Мета: Створити звіт про непрочитані продукти та зберегти його у форматі
файлу Excel.

```
{
  "image_file": "initialDataset/images/48.jpg",
  "section": "S101-001-048",
  "qrcode": [
    "0890115512211",
    "0890115514422",
    "0890115510921",
    "0890115540631",
    "0890115511021",
    "0890115541431",
    "0890115530144",
    "0890115511521",
    "0890115511621",
    "0890115511521",
    "0890115512211",
    "0890115542731",
    "0890115530144",
    "0890115544131",
    "0890115541431",
    "0890115512211",
    "0890115541431",
    "0890115511621",
    "0890115510811",
    "0890115511521",
    "0890115530144",
    "0890115511621",
    "0890115511521",
    "0890115511621"
  ]
},
{
  "image_file": "initialDataset/images/49.jpg",
  "section": "S101-001-049",
  "qrcode": [
    "0890115510522",
    "0890115548441",
    "0890115548341",
    "0890115510221",
    "0890115511121",
    "0890115511121"
  ]
}
```

Рисунок 2.5 - Приклад результату

Процес: Визначте продукти, яким не було порівняно жодного QR-коду.
Створіть список цих непрочитаних товарів. Збережіть його як файл Excel.

Функція 3:

'generate_unread_sections_report(результат_обробки_зображень,сек-
ції_складів)'

Мета: Сформувати звіт про непрочитані розділи та зберегти його у файлі
Excel.

Процес: Визначте ділянки, які не були виявлені на жодному зображенні.
Створіть список цих непрочитаних розділів. Збережіть його як файл Excel.

Крок 4: Створення діаграми інвентаризації

					БР.ІП - 17.00.00.000 ПЗ	Арк.
						38
Змн.	Арк.	№ докум.	Підпис	Дата		

Код: reportGeneration.py

Функція: 'generate_inventory_count_chart()'

Мета: Створити стовпчасту діаграму, яка візуалізує кількість запасів за розділами.

Процес: Завантажте зчитані дані продукту з файлу Excel. Використовуйте «панди», щоб підрахувати кількість товарів у кожному розділі. Побудуйте графік даних за допомогою 'matplotlib'. Збережіть діаграму як файл зображення.

Наступне обговорення представлено у формі тематичного дослідження, що документує досвід автора з вивчення та подальшої розробки програмного забезпечення для розрахунку окружності приводного колеса. Розробка була розділена на фазу навчання та фазу виконання, тоді як фази навчання та виконання були розділені на кілька етапів.

Фаза навчання

Автор розпочав цей проєкт лише через 6 місяців після початку вивчення та практики програмування, тому перед початком роботи над проєктом потрібно було ще багато чого вивчити. Перший місяць стажування в TP-Kunnossapito Oy був присвячений вивченню складних обчислень у C#, переходу від «циклічного» консольного кодування до кодування на основі «дії/реакції», і, найголовніше, об'єктно-орієнтованого програмування та використання класів замість повторюваного коду [23].

Навчання та перехід до об'єктно-орієнтованого програмування здійснювалися за допомогою навчальної платформи Microsoft Learn, а також виконання практичних вправ, як згадувалося раніше. Особливо корисною була стаття Microsoft Learn про об'єктно-орієнтоване програмування на C#. Ця стаття містить безліч прикладів та достатньо інформації, щоб мати змогу почати практикувати об'єктно-орієнтоване програмування за допомогою практичних вправ.

Доведення навичок програмування до належного рівня

Цей крок передбачав розробку автором різних програм для вивчення ключових концепцій цього проєкту. Серед них були розробка наукового калькулятора

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

для вивчення складних обчислень на C# та програми-щоденника для вивчення різних способів зберігання даних у пам'яті програми. Фінальним навчальним проектом був калькулятор довжини кола приводу, щоб поєднати нещодавно набуті здібності автора та перехід до об'єктно-орієнтованого програмування та початок використання класів. Оскільки калькулятор довжини кола приводу є складним, його програмування без використання об'єктно-орієнтованого програмування було б дуже нудним та трудомістким, що підштовхнуло автора до використання класів, завершуючи процес навчання.

Вивчення основ програмування та інженерії автоматизації

Наступним кроком після підвищення рівня навичок програмування автора було розуміння процесу, який проходять фахівці з автоматизації під час розрахунку діаметра приводного колеса навантаження, та загального процесу програмування ПЛК у TIA Portal. Так розпочався період вивчення автором принципів роботи TIA Portal, від створення проекту до моделювання ПЛК.

Процес навчання здебільшого полягав у виконанні онлайн-уроків, яких небагато, оскільки це програмне забезпечення використовується лише в спеціалізованій галузі інженерії, що ускладнює навчання. Однак автору пощастило отримати допомогу від фахівців-інженерів з компанії, яка проводила стажування/дипломну роботу, що забезпечило загалом безпроблемний процес навчання [24].

2.4 Висновок по розділу

У другому розділі розглянуто методи та інструменти розробки програмного рішення для автоматизації інвентаризації. Досліджено підходи до використання QR-кодів, методи формування та обробки даних, а також архітектурні особливості побудови системи. Визначено ключові принципи створення програмного забезпечення, що забезпечує точний та швидкий облік складських ресурсів. У результаті сформовано структурну основу для реалізації автоматизованої системи інвентаризації.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ СИСТЕМИ ІНВЕНТАРИЗАЦІЇ

3.1 Реалізація функціональних можливостей програмної системи

Функції проекту виявлення_qr_кодів (шлях_до_зображення) див. рис. 3.1

Опис: Ця функція використовує методи комп'ютерного зору для виявлення QR-кодів на зображенні. Вона зчитує зображення, вказане параметром image_path, застосовує алгоритми виявлення QR-кодів та повертає декодовану інформацію з будь-яких знайдених QR-кодів. Використані бібліотеки:

- cv2 (OpenCV): Надає функції для завантаження зображень (cv2.imread()), виявлення QR-кодів (cv2.QRCodeDetector()) та геометричних перетворень (cv2.polylines()). process_images(папка_зображень, output_json)

```
import cv2 # Computer Vision library for image processing

def detect_qr_codes(image_path):
    """
    Detect QR codes in an image using OpenCV.

    Parameters:
    image_path (str): Path to the image file.

    Returns:
    list: Decoded information from QR codes found in the image.
    """
    # Load the image
    image = cv2.imread(image_path)

    # Initialize the QRCodeDetector
    qr_detector = cv2.QRCodeDetector()

    # Detect and decode QR codes in the image
    retval, decoded_info, points, _ = qr_detector.detectAndDecodeMulti(image)

    if len(decoded_info) > 0:
        for i in range(len(decoded_info)):
            qr_points = points[i]

            # Draw a rectangle around the detected QR code
            cv2.polyline(image, [qr_points.astype(int)], True, (255, 0, 0), 2)
    else:
```

Рисунок 3.1 - Функція виявлення qr-кодів

Опис: Ця функція обробляє папку зображень (image_folder) див. рис.3.2 для виявлення QR-кодів у кожному зображенні. Вона компілює результати у структурований формат JSON та зберігає їх у файлі output_json. Вона обробляє кожен файл зображення окремо, виявляючи QR-коди та зберігаючи відповідні дані, такі як шлях до файлу зображення, код розділу та виявлені QR-коди.

					БР.ІП - 17.00.00.000 ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

Використані бібліотеки:

- cv2 (OpenCV): Для обробки зображень та розпізнавання QR-кодів.
- glob: Для обробки шляхів до файлів та ітерації по файлах зображень (glob.glob()).

```
import os          # Operating system functions
import cv2         # Computer Vision library for image processing
import glob        # File path management
from jsonHandling import save_to_json

def process_images(image_folder, output_json):
    """
    Process images to detect QR codes and save data to a JSON file.

    Parameters:
    image_folder (str): Path to the folder containing images.
    output_json (str): Path to the output JSON file.
    """
    image_files = glob.glob(os.path.join(image_folder, '*.jpg'))
    qr_data = []

    for file_path in image_files:
        try:
            # Detect QR codes in the image
            qr_codes = detect_qr_codes(file_path)

            if qr_codes:
                section_code = ""
```

Рисунок 3.2 - Функція обробляє папку зображень

save_to_json(дані, шлях_до_файлу)

Опис: Зберігає дані, надані в параметрі data, у файл JSON див. рис. 3.3, визначений параметром file_path. Форматує дані як JSON та записує їх у файл для подальшого отримання або аналізу.

```
product_qr_codes = []

# Identify section and product QR codes
for code in qr_codes:
    if code.upper().startswith('S') and not section_code:
        section_code = code
    else:
        product_qr_codes.append(code)

qr_data.append({
    "image_file": file_path,
    "section": section_code,
    "qrcode": product_qr_codes
})
else:
    print(f"No QR Codes detected in {file_path}")

except Exception as e:
    print(f'Error processing file {file_path}: {e}')

save_to_json(qr_data, output_json)
```

Рисунок 3.3 - Зберігає дані у файл JSON

					БР.ІП - 17.00.00.000 ПЗ	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дата		

Використані бібліотеки:

- json: Стандартна бібліотека Python для кодування та декодування даних JSON (json.dump()).

load_json_n_data(json_path)

Опис: Завантажує дані JSON з файл див. рис. 3.4 у, визначеного параметром json_path, та повертає їх як словник Python. Ця функція використовується для отримання раніше збережених даних JSON для подальшої обробки або аналізу.

Використані бібліотеки:

- json: Стандартна бібліотека Python для читання даних JSON (json.load()).

```
import json # JSON file handling

def save_to_json(data, file_path):
    """
    Save data to a JSON file.

    Parameters:
    data (list): Data to be saved.
    file_path (str): Path to the JSON file.
    """
    with open(file_path, 'w') as file:
        json.dump(data, file, indent=4)
```

Рисунок 3.4 - Завантажує дані JSON з файлу

generate_read_products_report(qr_data, products_list)

Опис: Генерує звіт про продукти, для яких на зображеннях було виявлено QR-коди див. рис. 3.5. Він зіставляє QR-коди, знайдені в qr_data, з продуктами зі списку products_list та створює детальний звіт, включаючи шляхи до файлів зображень, коди розділів, UPC, назви продуктів та вартість. Потім цей звіт зберігається як файл Excel (processed_products.xlsx).

Використані бібліотеки:

- pandas: Для створення та маніпулювання DataFrames (pd.DataFrame()).
- matplotlib.pyplot: Для візуалізації даних, зокрема для створення стовпчастих

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

```
# Convert the matched products list to a DataFrame
df_processed_products = pd.DataFrame(read_products)

# Save the DataFrame of matched products to an Excel file
df_processed_products.to_excel('reports/processed_products.xlsx', index=False)
```

Рисунок 3.5 - Зіставляє QR-коди, знайдені в qr_data

```
import pandas as pd # Data manipulation and analysis

def generate_read_products_report(qr_data, products_list):
    """
    Generate and save the read products report.

    Parameters:
    qr_data (list): QR code data.
    products_list (list): List of products.
    """
    read_products = []
    for entry in qr_data:
        section_code = entry["section"]
        product_qr_codes = entry["qrcode"]

        if section_code:
            matched_items = [item for item in products_list if item.get("UPC") in
                             product_qr_codes]

            if matched_items:
                for item in matched_items:
                    product_obj = {
                        "image_file": entry["image_file"],
                        "section_code": section_code,
                        "UPC": item["UPC"],
                        "Product title": item["Product title"],
                        "Cost": item["Unit Cost"]
                    }
                    read_products.append(product_obj)
```

Рисунок 3.6 - Генерує звіт про продукти

generate_unread_products_report(qr_data, products_list) згідно рис. 3.6.

Опис: Генерує звіт зі списком товарів, яких не було на зображеннях розділу складу. Він порівнює QR-коди, знайдені в qr_data, з товарами зі списку products_list, щоб виявити товари, що не збігаються. Цей звіт містить UPC, назви товарів та вартість товарів, яких немає на складі, і зберігається у вигляді файлу Excel (unread_products.xlsx) [25].

Використані бібліотеки:

					БР.ІП - 17.00.00.000 ПЗ	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дата		

- Pandas: Для створення та маніпулювання DataFrames (pd.DataFrame()) зображено на рис.3.7.

```
import pandas as pd # Data manipulation and analysis

def generate_unread_products_report(qr_data, products_list):
    """
    Generate and save the unread products report.

    Parameters:
    qr_data (list): QR code data.
    products_list (list): List of products.
    """
    read_products = []
    for entry in qr_data:
        section_code = entry["section"]
        product_qr_codes = entry["qrcode"]

        if section_code:
            matched_items = [item for item in products_list if item.get("UPC") in product_qr_codes]

            if matched_items:
                for item in matched_items:
                    read_products.append(item["UPC"])

    unread_products = []

    processed_upcs = {upc for upc in read_products}
    for product in products_list:
        if product["UPC"] not in processed_upcs:
            unread_products.append({
                "UPC": product["UPC"],
                "Product title": product["Product title"],
                "Cost": product["Unit Cost"]
            })

    # Convert unread products list to a DataFrame
    df_unread_products = pd.DataFrame(unread_products)

    # Save unread products data to an Excel file
    df_unread_products.to_excel('reports/unread_products.xlsx', index=False)
```

Рисунок 3.7 - Генерує звіт зі списком товарів, яких не було на зображеннях розділу складу

generate_unread_sections_report(qr_data, складські_розділи) див.рис. 3.8

Опис: Генерує звіт зі списком розділів на складі, для яких на зображеннях не було виявлено QR-кодів. Він порівнює розділи з warehouse_sections з кодами розділів, знайденими в qr_data, щоб ідентифікувати непрочитані розділи. Цей звіт зберігається як файл Excel (unread_sections.xlsx).

Використані бібліотеки:

					БР.ІП - 17.00.00.000 ПЗ	Арк.
						45
Змн.	Арк.	№ докум.	Підпис	Дата		

- Pandas: Для створення та маніпулювання DataFrames (pd.DataFrame()).

```
import pandas as pd      # Data manipulation and analysis

def generate_unread_sections_report(qr_data, warehouse_sections):
    """
    Generate and save the unread sections report.

    Parameters:
    qr_data (list): QR code data.
    warehouse_sections (list): List of warehouse sections.
    """
    read_sections = {entry["section"] for entry in qr_data}
    unread_sections = [{'section': section['section']} for section in warehouse_sections
                        if section['section'] not in read_sections]

    # Convert unread sections to a DataFrame
    df_unread_sections = pd.DataFrame(unread_sections)

    # Save unread sections data to an Excel file
    df_unread_sections.to_excel('reports/unread_sections.xlsx', index=False)
```

Рисунок 3.8 - Генерує звіт зі списком розділів на складі

generate_inventory_count_chart() див.рис. 3.9

Опис: Створює стовпчасту діаграму для візуалізації кількості товарів за секціями на основі оброблених даних. Зчитує дані з файлу processed_products.xlsx (згенерованого за допомогою generate_read_products_report()), обчислює кількість коробок товарів у кожній секції та візуалізує цю інформацію за допомогою стовпчастої діаграми. Діаграма зберігається як файл зображення (inventory_count_by_section.jpg).

Використані бібліотеки:

- Pandas: Для читання даних з Excel (pd.read_excel()).
- matplotlib.pyplot: Для візуалізації даних, зокрема для створення стовпчастих діаграм (plt.bar()).

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

```
import matplotlib.pyplot as plt # Data visualization

def generate_inventory_count_chart():
    """
    Generate a bar chart to visualize inventory counts by section.
    """
    df_processed_products = pd.read_excel('reports/processed_products.xlsx')
    inventory_counts = df_processed_products['section_code'].value_counts()
    inventory_counts.plot(kind='bar')
    plt.xlabel('Section Code')
    plt.ylabel('Number of Boxes')
    plt.title('Inventory Count by Section')
    plt.savefig('reports/inventory_count_by_section.jpg')
    plt.show()
```

Рисунок 3.9 - Створює стовпчасту діаграму для візуалізації кількості товарів за секціями на основі оброблених даних

3.2 Оцінка впливу системи на ефективність автоматизації складських процесів

Проблема, яку вирішує цей проект, полягає в необхідності ручного отримання параметрів для розрахунку, навантаження, його обчислення та ручного запису нових параметрів. Спочатку ідея полягала в тому, щоб створити програму, яка могла б відкрити проект, в якому знаходяться параметри керування роботом або параметри ПЛК, знайти та отримати необхідні параметри, розрахувати поточний діаметр приводного колеса навантаження, а потім, використовуючи дані від користувача, розрахувати новий діаметр, після чого вона б записала скориговані параметри назад у проект. Крім того, спочатку була ідея включити функціональність "креслення проекту", що дозволило б пришвидшити процес створення нового проекту, застосувавши креслення найпоширеніших параметрів та налаштувань, проте ця ідея була пізніше відхилена під час розробки.

Цей проект значно пришвидшив би процес розрахунку окружності приводного колеса навантаження завдяки великій кількості параметрів, які потрібно було знаходити та взаємодіяти з ними щоразу [26]. Пошук серед тисяч параметрів на порталі просто займає багато часу, що призводить до потреби в рішенні, яке з'явилося у вигляді програми, розробленої як тема цієї роботи. Також цінується додаткова економія часу завдяки автоматичному виконанню всіх розрахунків програмою.

					БР.ІП - 17.00.00.000 ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

Результат і важливість цього проекту важко переоцінити, оскільки зекономлений час можна буде використати для розробки подальших проектів автоматизації систем управління складом та роботів. Загальне підвищення ефективності команди автоматизації, безумовно, призведе до покращення продуктивності.

Зрештою, підвищення операційної ефективності супроводжується зменшенням кількості помилок, у цьому випадку шляхом усунення людського фактору під час копіювання та переписування, а зі зменшенням кількості помилок зменшується час, необхідний для виконання завдань. Окрім зменшення кількості помилок, операційної ефективності можна досягти шляхом зменшення потреби в ручній праці, у випадку цієї роботи саме зменшення ручної праці є основою проекту, що йде пліч-о-пліч зі збільшенням операційної ефективності [27]. Ефективно та точно функціонуюче обладнання означає додаткову операційну ефективність, а також зменшення потреби в додаткових налаштуваннях, у цьому проекті це означає, що точний розрахунок усуває необхідність повторного розрахунку невдовзі після цього.

Цільове використання проекту

Спочатку передбачалося, що програма матиме дві основні функції: 1. розрахунок діаметра приводного колеса навантаження, а 2. створення нових проектів за допомогою, можливо, готових креслень або бібліотек. Невизначеність щодо другої запланованої функціональності виникає через те, що вона так і не була реалізована, оскільки під час процесу розробки програми було вирішено, що немає потреби інвестувати значні кошти в розробку цієї функції, принаймні за нинішнього стану справ.

Програма була призначена для використання будь-яким персоналом, який намагається розрахувати вантажопідйомне колесо робота, можливо, після заміни деталі, встановлення нового робота або в багатьох інших сценаріях, що потребують розрахунку вантажопідйомного колеса [28]. Потім персонал повинен підключитися до робота або деталі та скопіювати її проект на свою машину, після чого людина має запустити програму, а решта процесу буде виглядати наступним чином:

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						48
Змн.	Арк.	№ докум.	Підпис	Дата		

Під час запуску користувач може зіткнутися з таким сповіщенням:

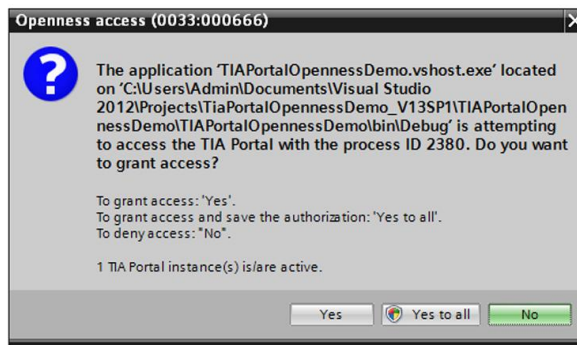


Рисунок 3.11 - Знімок екрана сповіщення, яке з'являється під час спроби доступу до проекту

Як видно зі знімка екрана, API відкритості ТІА потрібно надати доступ до порталу ТІА програмно. Натискання кнопки «Ні» не дозволить це з'єднання, що призведе до закриття програми. Однак натискання кнопки «Так» або «Так для всіх» призведе до встановлення з'єднання з порталом ТІА та належного запуску програми.

Після запуску програми користувач побачить наступний інтерфейс (Рисунок 3.12)

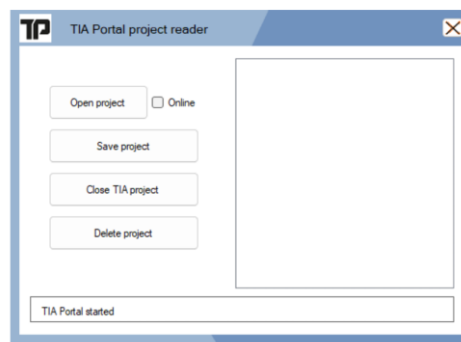


Рисунок 3.12 - Знімок екрана головного вікна інтерфейсу.

Натисніть кнопку «Відкрити проєкт» (перед відкриттям проєкту потрібно вибрати, чи буде редагування здійснюватися онлайн чи офлайн, вибір можна зробити, просто поставивши позначку в полі «Онлайн»)) (рисунок 3.13).

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

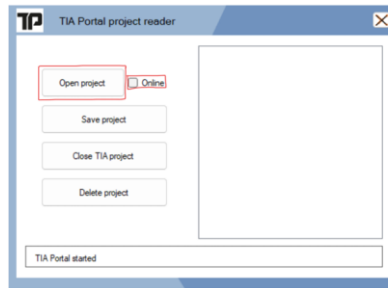


Рисунок 3.13 - Знімок екрана головного вікна інтерфейсу, що показує, як відкрити вікно вибору проекту.

Виберіть щойно скопійований проект (якщо програма раніше не використовувалася на цьому комп'ютері, тоді потрібно вказати папку проекту, натиснувши кнопку «Папка проекту» та перейшовши до потрібної папки), клацнувши на проекті, а потім натиснувши кнопку «ОК» (у випадку цієї візуальної презентації було обрано програму «Демо» за замовчуванням, щоб уникнути розголошення конфіденційної інформації про роботодавця) (рис. 3.14).

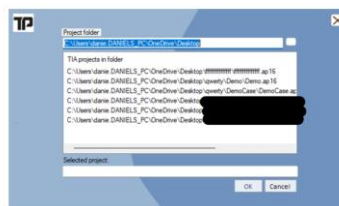


Рисунок 3.14 - Знімок екрана вікна «Вибір проекту».

Перейдіть до потрібного об'єкта диска в дереві (у цьому випадку це був «blue») і клацніть на ньому (рисунок 3.15).

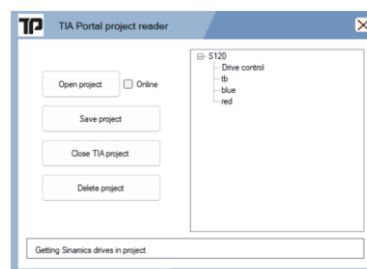
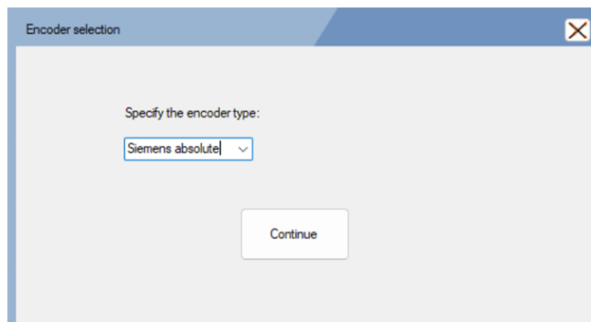


Рисунок 3.15 - Знімок екрана головного вікна інтерфейсу після відкриття проекту.

					БР.ІП - 17.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		50

Виберіть тип енкодера для об'єкта приводу (у разі наявності 2 енкодерів програма запропонує вибрати 2 типи енкодерів), а потім натисніть «Продовжити» (рисуюнок 3.16).



Рисуюнок 3.16 - Знімок екрана вікна «Вибір енкодера».

Відкриється наступне вікно, у якому всі значення, доступні для програмного управління, будуть автоматично скопійовані безпосередньо з проекту [29]. Тут користувачеві потрібно буде ввести значення, недоступні для програмного управління.



Рисуюнок 3.17 - Знімок екрана сторінки «Налаштування енкодера» інтерфейсу розрахунку

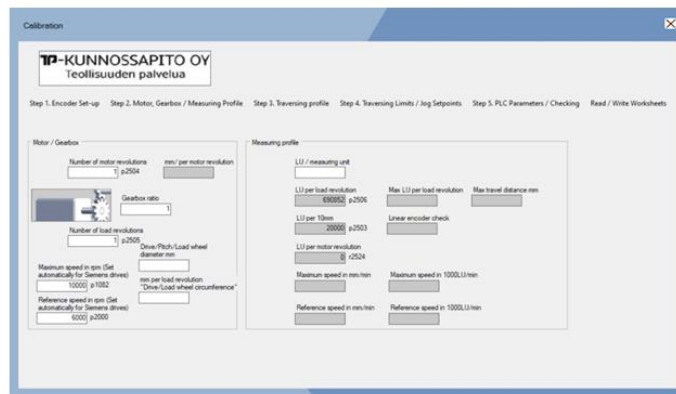


Рисунок 3.18 - Знімок екрана сторінки «Двигун, редуктор / Профіль вимірювання» інтерфейсу розрахунку.

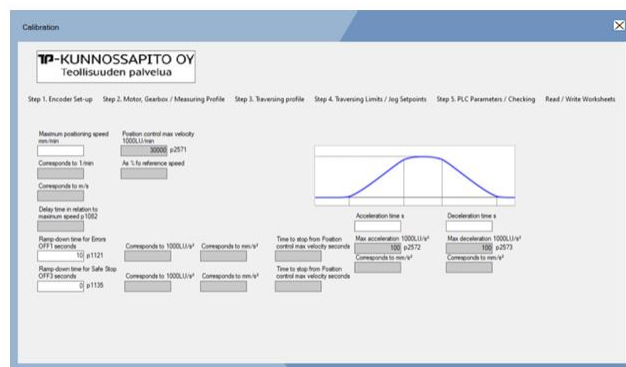


Рисунок 3.19 - Знімок екрана сторінки «Профіль траверсування» інтерфейсу розрахунку.

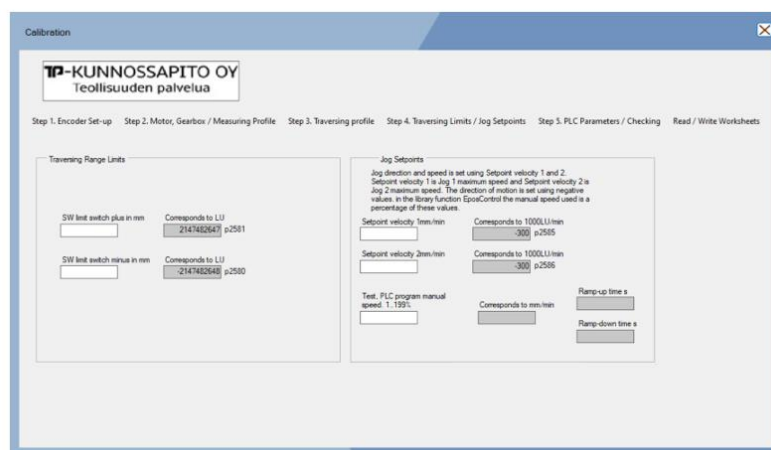


Рисунок 3.20 - Знімок екрана сторінки «Межі переміщення / Задані значення поштовхового переміщення» інтерфейсу розрахунку.



Рисунок 3.21 - Знімок екрана сторінки «Параметри ПЛК / Перевірка» інтерфейсу розрахунку.

На сторінці інтерфейсу розрахунку користувач може натиснути кнопку «Зберегти неімпортовані змінні», яка збереже значення, введені користувачем вручну, що полегшить розрахунок довжини кола приводного колеса того ж проекту в майбутньому, ще більше автоматизуючи процес [30]. Натискання кнопки «Записати»

Натисканням кнопки «Новий робочий аркуш» користувач накаже програмі записати нові змінені вручну значення, включаючи нову довжину кола приводного вантажного колеса, у вихідний проект TIA Portal (рис. 3.22).

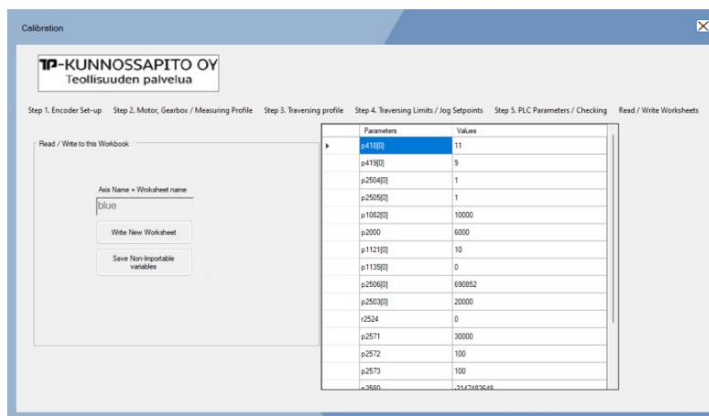


Рисунок 3.22 - Знімок екрана сторінки «Читання / запис робочих аркушів» інтерфейсу розрахунку.

Наступним кроком буде виконання самого розрахунку. Після ручного введення всіх програмно недоступних даних користувач спробує керувати роботом, наказавши йому переміститися на певну відстань (у цьому прикладі 100 одиниць

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		53

вимірювання), вимірявши результат руху, а потім ввівши вхідні та фактичні вимірювання на сторінці 5 програми. Наведений нижче приклад демонструє, як це виглядає, якщо окружність приводного вантажного колеса була правильною з самого початку (рисунок 3.23).



Рисунок 3.23 - Знімок екрана сторінки «Параметри ПЛК / Перевірка» інтерфейсу розрахунку із завершеним розрахунком.

Однак, якщо реальні вимірювання не відповідають вхідним, програма розрахує скориговану довжину кола, а після натискання кнопки DLWC (довжина кола приводного колеса) програма запише нове значення замість старого (Примітка: для найточнішого розрахунку рекомендується повторити цей процес кілька разів). Наведений нижче приклад показує, як це виглядатиме, якщо реальні вимірювання не збігатимуться з вхідними (рисунок 3.23).

Очікуваний ефект проєктів

Як згадувалося раніше, цільовим використанням та ефектом проєкту з точки зору співробітників, відповідальних за автоматизацію, була автоматизація процесу розрахунку діаметра приводного вантажного колеса в проєктах TIA Portal PLC [31]. Бажаним ефектом такої автоматизації було скорочення часу, необхідного для виконання зазначеного розрахунку, шляхом усунення виснажливого ручного копіювання необхідних параметрів, обчислень та запису нових параметрів назад у проєкт.

Очікувалося, що вплив на компанію в цілому полягатиме в незначному загальному підвищенні ефективності відділу автоматизації, оскільки, скоротивши

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		54

виснажливу та повторювану частину розрахунку окружності приводного колеса, можна було б витратити більше часу на роботу над іншими проектами [32].

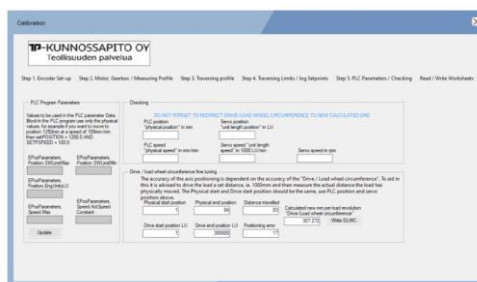


Рисунок 3.24 - Знімок екрана сторінки «Параметри ПЛК / Перевірка» інтерфейсу розрахунку на момент виконання розрахунку

Наступний розділ представлено у формі тематичного дослідження, що документує досвід автора з вивчення та подальшої розробки програмного забезпечення для розрахунку окружності приводного колеса. Розробка була розділена на фазу навчання та фазу виконання, тоді як фази навчання та виконання були розділені на кілька етапів.

Фаза навчання

Автор розпочав цей проєкт лише через 6 місяців після початку вивчення та практики програмування, тому перед початком роботи над проєктом потрібно було ще багато чого вивчити. Перший місяць стажування в TP-Kunnossapito Oy був присвячений вивченню складних обчислень у C#, переходу від «циклічного» консольного кодування до кодування на основі «дії/реакції», і, найголовніше, об'єктно-орієнтованого програмування та використання класів замість повторюваного коду.

Навчання та перехід до об'єктно-орієнтованого програмування здійснювалися за допомогою навчальної платформи Microsoft Learn, а також виконання практичних вправ, як згадувалося раніше. Особливо корисною була стаття Microsoft Learn про об'єктно-орієнтоване програмування на C#. Ця стаття містить безліч прикладів та достатньо інформації, щоб мати змогу почати практикувати об'єктно-орієнтоване програмування за допомогою практичних вправ.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		55

Доведення навичок програмування до належного рівня

Цей крок передбачав розробку автором різних програм для вивчення ключових концепцій цього проекту [33]. Серед них були розробка наукового калькулятора для вивчення складних обчислень на C# та програми-щоденника для вивчення різних способів зберігання даних у пам'яті програми. Фінальним навчальним проектом був калькулятор довжини кола приводу, щоб поєднати нещодавно набуті здібності автора та перехід до об'єктно-орієнтованого програмування та початок використання класів. Оскільки калькулятор довжини кола приводу є складним, його програмування без використання об'єктно-орієнтованого програмування було б дуже нудним та трудомістким, що підштовхнуло автора до використання класів, завершуючи процес навчання.

Вивчення основ програмування та інженерії автоматизації

Наступним кроком після підвищення рівня навичок програмування автора було розуміння процесу, який проходять фахівці з автоматизації під час розрахунку діаметра приводного колеса навантаження, та загального процесу програмування. Так розпочався період вивчення автором принципів, від створення проекту до моделювання ПЛК.

Процес навчання здебільшого полягав у виконанні онлайн-уроків, яких небагато, оскільки це програмне забезпечення використовується лише в спеціалізованій галузі інженерії, що ускладнює навчання. Однак автору пощастило отримати допомогу від фахівців-інженерів з компанії, яка проводила стажування/дипломну роботу, що забезпечило загалом безпроблемний процес навчання.

Розуміння API відкритості

API відкритості— це API, призначений для доступу, редагування, створення та видалення проектів, змінних та схем. Для цього проекту було використано API відкритості, оскільки саме цю версію використовує компанія-підрядник.

Щоб зрозуміти справді широкий функціонал цього API, автору потрібно було ознайомитися з посібником TIA Portal Openness API V16, обсяг якого

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						56
Змн.	Арк.	№ докум.	Підпис	Дата		

становив 1292 сторінки, що зробило ознайомлення значно тривалим процесом. Підхід автора до цього посібника був наступним:

Спочатку пройдіться по всьому посібнику, позначивши розділи, які можуть мати відношення до цього проекту (розділи, що стосуються читання, запису та редагування параметрів, створення, збереження, редагування та видалення проекту, а також використання шаблонів проектів та об'єктів, були найбільш цікавими спочатку).

Потім уважно прочитайте всі раніше позначені розділи.

Після цього поекспериментуйте з прикладами коду та іншою інформацією, наведеною в цих розділах, що було спрощено за допомогою доступних демонстраційних програм, доступних на офіційному веб-сайті Siemens SIE Portal. Нарешті, поекспериментуйте з використанням інформації, отриманої з посібника, для створення «демо»-версії проекту.

Фаза планування

Фаза планування розпочалася одночасно з фазою ознайомлення, оскільки планування мало сенс лише на основі новоотриманої інформації. Таким чином, планування було здебільшого завершено до моменту завершення «демо» або «підтвердження концепції» версії проекту. Таким чином, план виглядав наступним чином:

1. Створити версію для підтвердження концепції, що демонструє можливість отримання та редагування даних з проектів TIA Portal за допомогою TIA Openness API.
2. Мав бути узгоджений план приблизного остаточного вигляду заявки.
3. Повинно бути досягнуто об'єднання та інтеграцію двох окремих частин, а саме інтерфейсу розрахунку та інтерфейсу, що використовує TIA Portal Openness API.
- 4-й. Завершення роботи над заявкою.
- 5-й. Презентація підряднику.
- 6-е. Отримання відгуків та редагування проекту відповідно до цих відгуків.

Розуміння завдання

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						57
Змн.	Арк.	№ докум.	Підпис	Дата		

Під час планування проєкту було важливо зрозуміти, що він передбачає та які особливості завдання. Більшість інформації, необхідної для розуміння завдання, було отримано від контактної особи компанії-наймача, а решту деталей – з посібника TIA Openness API та розуміння обмежень API.

Що стосується інформації, наданої контактною особою, її було більш ніж достатньо, щоб сформулювати уявлення про те, як має виглядати проєкт до кінця, та створити ескіз необхідних кроків для досягнення цієї мети, як зазначено вище. Щоразу, коли виникали проблеми з розумінням, автор просто зв'язувався з контактною особою для уточнення, що сприяло безперебійному проходженню процесу планування та роботи. Однак, що стосується інструкцій та практичних навчальних частин, виникало багато труднощів через розбіжності в початковому плані та обмеження API [34]. Більше про це питання можна дізнатися нижче в розділах «Етап виконання» та «Частина проєкту, що стосується API відкритості TIA» цього сегмента.

Розділення завдання на сегменти

Як вже згадувалося раніше в плані проєкту, вже існували ідеї щодо сегментації проєкту з міркувань зручності редагування, концентрації та налагодження. Якщо проєкт розділений на сегменти, то немає сумнівів щодо того, звідки походять можливі помилки. Проєкт було вирішено розділити на чотири частини.

Перша - це розрахункова частина, яка відповідає лише за розрахунок довжини кола нового приводного колеса, коли задані всі необхідні для цього параметри. Друга – частина TIA Openness API, яка відповідає лише за зв'язок між авторським проєктом та TIA Portal. Третє – дизайн заявки, який мав бути затверджений контактною особою компанії-наймача. І остання, значно менша, але важлива частина – це канал зв'язку для інтерфейсів розрахунку та TIA Openness API.

Етап виконання

Після визначення плану проєкту та його розділення на окремі керовані частини нарешті настав час розпочати сам процес розробки. Як зазначалося вище, частина розрахунків була частково завершена в рамках початкового процесу

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

навчання. Тому етап виконання розпочався з розробки частини інтерфейсу API TIA Portal.

Під час етапу виконання автор коливався між розробкою сегмента інтерфейсу API TIA Portal та вдосконаленням існуючого сегмента обчислень. Причина цього полягає в тому, що в галузі програмування існує явище, подібне до «тунельного зору», коли людина зосереджується на одній проблемі протягом тривалого часу [35]. Тому розділення проекту на сегменти призвело до неочікуваних переваг протягом усього процесу розробки.

Розрахункова частина проекту

Як неодноразово зазначалося раніше, сегмент розрахунків був частково завершений як навчальна вправа в тій частині стажування, яка була зосереджена на підвищенні навичок програмування автора до необхідного рівня. Після цього фокус перемістився на частину проекту, пов'язану з API, і залишався поза межами інтерфейсу розрахунків до обміну автором у Берліні.

Під час обміну автору вдалося успішно об'єднати дві неповні частини проекту, що висвітлило потребу в коригуванні в сегменті розрахунків. Серед частин, які потребували коригування, був вибір типу кодера, який спочатку виконувався в інтерфейсі розрахунків, але якщо під час відкриття інтерфейсу розрахунків мав бути тип кодера за замовчуванням, значення могли бути втрачені під час зміни типу кодера. Тому вибір типу кодера було перенесено безпосередньо перед відкриттям інтерфейсу розрахунків, оскільки у випадку двох кодерів з'являється спливаюче вікно із запитом на тип або типи кодерів. Додаткові функції, які потребували налаштування, включали спосіб зберігання та отримання пам'яті застосунком, оптимізацію розрахунків та загальне доопрацювання візуальних аспектів інтерфейсу.макет.

Збір відгуків та впровадження коригувань

Під час збору відгуків та внесення коригувань на основі цих відгуків до проекту було внесено багато змін. Ці зміни варіювалися від візуальних налаштувань до зміни або повного видалення цілих функцій. Наприклад, рішення про

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		59

скасування можливості створення та видалення проєктів було прийнято під час однієї з таких сесій зворотного зв'язку.

Серед інших помітних змін у проєкті були:

Рішення перенести список властивостей до інтерфейсу розрахунку, оскільки спочатку він відображався одразу після вибору проєкту на інтерфейсі зв'язку. Рішення замінити перенесений список властивостей на оглядач проєкту. Рішення реалізувати інтерфейс вибору кодера до етапу розрахунку використання програми.

Те, наскільки автору вдалося отримати відгуки та співпрацю з підрядником і кінцевим користувачем, дало унікальну можливість максимально адаптувати кінцевий досвід. Фактичне використання кінцевого продукту ще не розпочалося, оскільки його планувалося використовувати приблизно влітку, і після польових випробувань очікується, що знадобляться додаткові оновлення та виправлення помилок.

Технічне обслуговування та оновлення

Автор очікує виявлення помилок у польових умовах, оскільки програму можна було протестувати лише за допомогою сервера, щедро наданого Siemens, і хоча вона, здається, добре працювала на сервері, вона може не повністю відображати кінцевий користувацький досвід. Помилки є неминучим аспектом програмування, і якщо програміст не використовує попередньо написаний і протестований код, існує майже неминуча ймовірність зіткнутися з помилками, що призводить до необхідності виправлень.

3.3 Аналіз продуктивності та ефективності виявлення QR-кодів

Використання OpenCV для виявлення QR-кодів на складі Walmart виявилось ефективним, з високим рівнем успішного виявлення та точного декодування QR-кодів. Система добре працювала за різних якостей зображення та умов освітлення, хоча деякі проблеми спостерігалися в умовах надзвичайно низької освітленості або коли зображення мали значні перешкоди [36].

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

Обробка даних та звітність

Процес обробки даних, від виявлення QR-кодів до створення звітів, був ефективним та точним. Функції обробки даних JSON та створення звітів Excel у Walmart працювали надійно, забезпечуючи правильне зіставлення даних про запаси та звітність. Однак обробка дуже великих обсягів зображень може вимагати додаткової оптимізації для підвищення продуктивності.

Візуалізація та аналітика

Діаграми підрахунку запасів, згенеровані системою, надавали чітке та практичне уявлення про розподіл запасів. Ці візуалізації допомагали визначати секції з високим або низьким рівнем запасів, сприяючи кращому розподілу ресурсів та прийняттю рішень щодо управління запасами.

Порівняння

Точність: Традиційне ручне управління запасами схильне до людських помилок, таких як неправильне введення даних та пропущені сканування. Автоматизована система значно зменшує кількість цих помилок завдяки точному виявленню та обробці QR-кодів.

Ефективність: Ручні методи були трудомісткими та трудомісткими.

Автоматизована система підвищує ефективність, швидко обробляючи зображення та оновлюючи записи інвентаризації, що призводить до суттєвої економії часу та більш оптимізованого робочого процесу.

Вартість: Ручне управління запасами передбачає постійні витрати на оплату праці. Автоматизована система зменшує ці витрати, мінімізуючи ручну працю, що робить її більш економічно ефективним рішенням у довгостроковій перспективі.

Автоматизовані системи проти інших автоматизованих технологій

Системи штрих-кодів: Хоча традиційні системи штрих-кодів пропонують певну автоматизацію, вони все ще вимагають сканування прямої видимості та менш універсальні, ніж QR-коди, які можуть зберігати більше інформації та зчитуватися з різних кутів.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						61
Змн.	Арк.	№ докум.	Підпис	Дата		

Системи RFID: Системи RFID пропонують аналогічні переваги в автоматизації та точності даних, але зазвичай дорожчі у впровадженні, ніж QR-коди. Вибір між QR-кодами та RFID залежить від конкретних потреб та бюджетних міркувань, причому QR-коди пропонують більш економічно ефективне рішення, водночас забезпечуючи надійну функціональність.

Заключні звіти та результати

Звіт про оброблені продукти для Walmart ідентифікує та реєструє продукти, QR-коди яких було успішно виявлено та розшифровано на зображеннях складу. У ньому перераховано продукти разом з відповідними кодами секцій, UPC, назвами продуктів та вартістю [37]. Список продуктів та результати обробки зображень слугують основними джерелами даних для цього звіту. Список продуктів містить вичерпну інформацію про товари, тоді як результати обробки зображень ідентифікують конкретні продукти, виявлені в різних секціях складу. Інтегруючи ці джерела даних, звіт сприяє ретельному аналізу та порівнянню інформації, забезпечуючи точність та актуальність даних про запаси в магазині Walmart.

Цей звіт надає важливу інформацію про правильне зіставлення продуктів з відповідними розділами складу. Таке зіставлення є важливим для ведення точного обліку запасів [38]. Крім того, звіт допомагає у прийнятті стратегічних рішень, виявляючи продукти, яких може бути недостатньо на складі. Він пропонує практичну інформацію, яка підтримує рішення щодо повторного замовлення продуктів.

Використовуючи результати цього звіту, можна ефективніше керувати рівнями запасів, тим самим мінімізуючи ризик дефіциту. Таким чином, звіт відіграє ключову роль у вдосконаленні процесів планування та оптимізації загального управління запасами.

Звіт про непрочитані товари містить список товарів, яких не було на зображеннях складу. Цей звіт було створено з використанням списку товарів та результатів обробки зображень. Його метою є визначення товарів, які є у списку товарів, але не відображаються в результатах обробки зображень.

Звіт містить ключові деталі, такі як універсальні коди товарів (UPC), назви

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						62
Змн.	Арк.	№ докум.	Підпис	Дата		

товарів та їх вартість. Ця інформація є важливою для виявлення товарів, яких наразі немає на складі. Оперативне повторне замовлення цих товарів, яких немає на складі, є важливим для підтримки оптимального рівня запасів та забезпечення належного постачання на склад [39].

Звіт про непрочитані товари життєво важливий для ефективного управління запасами, оскільки він допомагає запобігти дефіциту товарів, забезпечуючи наявність усіх необхідних товарів на складі. Сприяючи своєчасному повторному замовленню, звіт відіграє вирішальну роль у підтримці належного запасу та підтримці загальної ефективності складських операцій.

Остаточний звіт – це Звіт про непрочитані секції, який визначає секції складу, які не були виявлені QR-кодами на зображеннях або де захоплення зображень було проблематичним. Цей звіт було створено шляхом аналізу даних секцій складу разом з результатами обробки зображень. Він виділяє області, де виявлення QR-кодів могло не вдатися, допомагаючи визначити секції, які можуть потребувати нових зображень. Крім того, звіт точно визначає області, де камера зіткнулася з труднощами, надаючи цінну інформацію для оцінки та виправлення потенційних проблем з обладнанням камери.

Цей звіт має вирішальне значення для забезпечення комплексного моніторингу всіх секцій складу. Визначаючи секції, які можуть бути не повністю охоплені, звіт підтримує зусилля щодо повного та точного відстеження запасів.

Остаточна діаграма

У цьому розділі розглядається остаточна діаграма, діаграма підрахунку запасів, яка візуалізує розподіл запасів по різних секціях складу Walmart. Діаграма представлена у вигляді стовпчиків, де кожен стовпчик відображає кількість коробок з запасами в кожній секції. Дані для цієї діаграми отримані зі списку товарів та результатів обробки зображень.

Таблиця інвентаризації рисунок 3.5 забезпечує чітке відображення кількості продукції, що зберігається в кожній секції складу. Вона полегшує ідентифікацію секцій з надлишком або нестачею запасів, тим самим сприяючи більш обґрунтованому розподілу ресурсів та покращуючи прийняття рішень в управл

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						63
Змн.	Арк.	№ докум.	Підпис	Дата		

інні запасами.

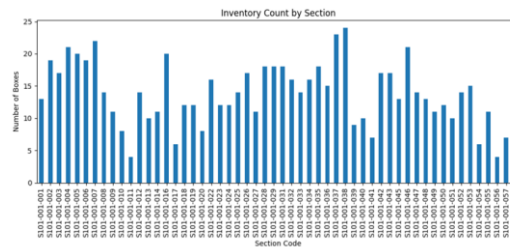


Рисунок 3.25 - Діаграма інвентаризації

Впровадження автоматизованої системи управління запасами за допомогою QR-кодів та комп'ютерного зору значно підвищує точність та ефективність складських операцій [40]. Традиційні ручні методи відстеження запасів схильні до помилок та неефективності, що призводить до таких проблем, як розбіжності в запасах та збільшення витрат на оплату праці. Завдяки інтеграції QR-кодів та використанню технології комп'ютерного зору через бібліотеку OpenCV, цей проект ефективно вирішує ці проблеми, як продемонстровано на прикладі магазину Walmart, де він був впроваджений.

Здатність системи виявляти та декодувати QR-коди зі зображень складів Walmart дозволяє оновлювати записи про запаси в режимі реального часу. Це не лише зменшує залежність від ручного сканування, але й гарантує, що дані про запаси є більш точними та актуальними. Автоматизований підхід оптимізує процес управління запасами, мінімізує людські помилки та надає детальні звіти та візуалізації, що допомагають у прийнятті кращих рішень.

Проект демонструє, що використання сучасних технологій, таких як QR-коди та комп'ютерний зір, може призвести до суттєвого покращення практики управління запасами не лише у Walmart, але й на складах інших великих роздрібних магазинів та мережових магазинів.

Розширені алгоритми обробки зображень та машинне навчання: Майбутні вдосконалення автоматизованої системи управління запасами можуть бути зосереджені на інтеграції більш складних методів обробки зображень, таких як згорткові нейронні мережі (CNN) або моделі глибокого навчання, система може

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						64
Змн.	Арк.	№ докум.	Підпис	Дата		

досягти вищої точності у виявленні QR-кодів навіть у складних умовах, таких як низьке освітлення або перекриття. Алгоритми машинного навчання також можуть бути використані для постійного підвищення точності та ефективності розпізнавання QR-кодів з часом шляхом навчання на минулих помилках та успіхах.

Інтеграція пристроїв Інтернету речей: Впровадження пристроїв Інтернету речей (IoT), таких як розумні камери, оснащені передовими датчиками зображення та датчиками навколишнього середовища (наприклад, температури, вологості), може значно покращити моніторинг та збір даних у режимі реального часу. Розумні камери забезпечуватимуть зображення високої роздільної здатності та відеопотоки, що дозволить точніше відстежувати запаси, тоді як датчики навколишнього середовища зможуть контролювати умови складських приміщень, забезпечуючи оптимальне середовище для різних типів продукції.

Аналітика в режимі реального часу та інтерактивні інформаційні панелі: Розширення можливостей системи для включення аналітики в режимі реального часу дозволить негайно аналізувати дані про запаси в міру їх збору, надаючи інформацію про рівні запасів, рух запасів та потенційні проблеми в міру їх виникнення. Можна розробити інтерактивні інформаційні панелі, щоб користувачі могли налаштовувати та візуалізувати дані відповідно до своїх конкретних потреб, що полегшить виявлення тенденцій, виявлення аномалій та швидке прийняття обґрунтованих рішень.

3.4 Висновок по розділу

У третьому розділі реалізовано програмну систему автоматизації інвентаризації та проведено оцінку її ефективності. Виконано розробку функціональних можливостей системи, тестування механізмів зчитування QR-кодів та аналіз впливу на оптимізацію складських процесів. Отримані результати підтвердили підвищення швидкості та точності інвентаризації при використанні програмного рішення. Це свідчить про практичну ефективність запропонованого підходу та доцільність його впровадження у складські системи.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						65
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У ході виконання дипломної роботи було розроблено програмне рішення для автоматизації процесу інвентаризації на складах, спрямоване на підвищення ефективності обліку товарів, оптимізацію складських процесів та зменшення впливу людського фактора. У роботі проаналізовано сучасні підходи до автоматизації складської діяльності, досліджено методи використання QR-кодів для ідентифікації товарів, а також реалізовано функціональні можливості системи, які забезпечують швидке та точне проведення інвентаризації. Процес розробки охопив аналіз вимог, моделювання бізнес-логіки, проєктування архітектури системи, реалізацію функціональних модулів, тестування та оцінку ефективності програмного забезпечення.

Роботу було розпочато з дослідження предметної області та аналізу проблем, пов'язаних із проведенням інвентаризації на складах. Було визначено основні недоліки традиційних методів обліку товарів, серед яких значні витрати часу, висока ймовірність помилок та складність обробки великих обсягів даних. На основі проведеного аналізу сформовано функціональні та нефункціональні вимоги до системи, які враховують потреби сучасних складських підприємств щодо швидкодії, точності, масштабованості та зручності використання.

У процесі дослідження було розглянуто сучасні методи автоматизації складських процесів, зокрема використання QR-кодів та засобів цифрової ідентифікації товарів. Проведено аналіз підходів до формування наборів даних і способів їх обробки, що дозволило визначити найбільш ефективні методи реалізації системи інвентаризації. Значну увагу приділено вивченню впливу робототехніки та цифрових технологій на підвищення ефективності складських операцій і оптимізацію логістичних процесів.

Реалізація програмного рішення охопила створення функціональних модулів для сканування QR-кодів, обробки даних про товари, контролю складських залишків та формування звітності. Було реалізовано інтерфейс користувача, який забезпечує зручну взаємодію із системою та швидкий доступ до інформації про

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						66
Змн.	Арк.	№ докум.	Підпис	Дата		

складські операції. Особлива увага приділялася швидкості обробки даних, стабільності роботи системи та точності виявлення QR-кодів у різних умовах використання.

У ході тестування програмного забезпечення було проведено перевірку функціональності, продуктивності та надійності системи. Аналіз результатів тестування дозволив виявити окремі проблеми, пов'язані з обробкою великих обсягів даних та швидкістю розпізнавання QR-кодів, що стало підставою для подальшої оптимізації алгоритмів обробки інформації. Проведене тестування підтвердило відповідність програмного рішення поставленим вимогам і його придатність для використання в реальних складських умовах.

Загалом, розроблене програмне рішення відповідає поставленим цілям та забезпечує ефективну автоматизацію процесу інвентаризації на складах. Основними перевагами системи є підвищення точності обліку товарів, скорочення часу проведення інвентаризації, зменшення впливу людського фактора та можливість інтеграції з іншими складськими системами. Використання QR-кодів та сучасних програмних технологій дозволяє забезпечити високу швидкодію та гнучкість системи.

Водночас система має певні обмеження, пов'язані з необхідністю подальшої оптимізації продуктивності при роботі з великими обсягами складських даних та залежністю від якості сканування QR-кодів. У майбутньому програмне рішення може бути вдосконалене шляхом інтеграції технологій машинного навчання для автоматичного аналізу складських процесів, підтримки мобільних пристроїв, використання RFID-технологій та впровадження засобів аналітики в реальному часі, що дозволить ще більше підвищити ефективність управління складськими операціями.

					БР.ІІІ - 17.00.00.000 ПЗ	Арк.
						67
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ПОСИЛАНЬ НА ДЖЕРЕЛА

1. Angular Documentation. (2025). angular.io. <https://angular.io/docs>

2. ASP.NET Core Documentation. (2025). learn.microsoft.com. <https://learn.microsoft.com/en-us/aspnet/core/>

3. Azure Application Insights Overview. (2025). learn.microsoft.com. <https://learn.microsoft.com/en-us/azure/azure-monitor/app/app-insights-overview>

4. Azure DevOps CI/CD Pipelines. (2025). learn.microsoft.com. <https://learn.microsoft.com/en-us/azure/devops/pipelines/>

5. Bhattacharya, S., & Roy, P. (2024). Smart warehouse inventory systems using QR-code technologies. Journal of Logistics Automation, 14(2), 88–102. <https://doi.org/10.1016/j.jla.2024.02.005>

6. Bootstrap Documentation. (2025). getbootstrap.com. <https://getbootstrap.com/docs/>

7. Brown, T., & Lewis, J. (2023). Automation technologies in warehouse management systems. International Journal of Supply Chain Management, 18(4), 201–215. <https://doi.org/10.1108/IJSCM-2023-0045>

8. Clean Architecture in .NET. (2024). learn.microsoft.com. <https://learn.microsoft.com/en-us/dotnet/architecture/modern-web-apps-azure/>

9. Cypress End-to-End Testing Guide. (2025). docs.cypress.io. <https://docs.cypress.io/guides/overview/why-cypress>

10. Deloitte. (2025). Digital transformation in warehouse logistics. deloitte.com. <https://www2.deloitte.com/global/en/pages/operations/articles/digital-warehouse-solutions.html>

11. Fowler, M. (2002). Patterns of Enterprise Application Architecture. martinowler.com. <https://martinfowler.com/eaCatalog/>

12. Freeman, E., & Robson, E. (2021). Head First Design Patterns. oreilly.com. <https://www.oreilly.com/library/view/head-first-design/9781492077992/>

13. Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). Design Patterns: Elements of Reusable Object-Oriented Software. pearson.com. <https://www.pearson.com/store/p/design-patterns-elements-of-reusable-object-oriented/P100000252058>
14. Google ZXing Project. (2025). QR code processing library. github.com. <https://github.com/zxing/zxing>
15. Hassan, M., & Khan, A. (2023). Scalable software architectures for warehouse automation. IEEE Transactions on Industrial Informatics, 19(3), 142–156. <https://doi.org/10.1109/TII.2023.3248901>
16. ImpactQA. (2024). Importance of software testing in logistics systems. impactqa.com. <https://www.impactqa.com/software-testing-in-logistics-industry/>
17. Inventory Management with QR Codes. (2024). Zebra Technologies. zebra.com. <https://www.zebra.com/us/en/solutions/qr-code-inventory-anagement.html>
18. JavaScript Charts for Data Visualization. (2025). chartjs.org. <https://www.chartjs.org/docs/latest/>
19. Kent, J., & Williams, R. (2024). Warehouse robotics and inventory optimization. Robotics and Automation Review, 9(1), 34–49. <https://doi.org/10.1016/j.robot.2024.01.009>
20. Kumar, S., & Sharma, R. (2023). Microservices architecture for warehouse management applications. Journal of Software Engineering Research and Development, 11(2), 67–82. <https://doi.org/10.1186/s40411-023-00189-9>
21. Logistics Management Systems Overview. (2025). IBM. <https://www.ibm.com/topics/warehouse-management-system>
22. Martin, R. C. (2017). Clean Architecture: A Craftsman’s Guide to Software Structure and Design. pearson.com. <https://www.pearson.com/store/p/clean-architecture-a-craftmans-guide-to-software-structure-and-design/P100001465201>
23. Microsoft SQL Server Documentation. (2025). learn.microsoft.com. <https://learn.microsoft.com/en-us/sql/sql-server/>
24. MongoDB Documentation. (2025). mongodb.com. <https://www.mongodb.com/docs/>

					БР.ІІІ - 17.00.00.000 ІІЗ	Арк.
						69
Змн.	Арк.	№ докум.	Підпис	Дата		

25. .NET Best Practices for Backend Development. (2024). learn.microsoft.com. <https://learn.microsoft.com/en-us/dotnet/core/extensions/best-practices>
26. OpenCV Documentation. (2025). opencv.org. <https://docs.opencv.org/>
27. Osherove, R. (2019). The Art of Unit Testing. manning.com. <https://www.manning.com/books/the-art-of-unit-testing-third-edition>
28. OWASP Top Ten Security Risks. (2024). owasp.org. <https://owasp.org/www-project-top-ten/>
29. Postman API Platform Documentation. (2025). learning.postman.com. <https://learning.postman.com/docs/>
30. React Documentation. (2025). react.dev. <https://react.dev/>
31. RFID and QR Code Integration in Warehouse Systems. (2024). Oracle. oracle.com. <https://www.oracle.com/scm/warehouse-management/>
32. SASS Documentation. (2025). sass-lang.com. <https://sass-lang.com/guide>
33. Selenium Documentation. (2025). selenium.dev. <https://www.selenium.dev/documentation/>
34. Siemens. (2024). Smart warehouse automation solutions. siemens.com. <https://www.siemens.com/global/en/products/automation/topic-areas/warehouse-automation.html>
35. Smith, A., & Johnson, K. (2025). Real-time warehouse inventory analytics using cloud technologies. ACM Transactions on Information Systems, 21(1), 1–19. <https://doi.org/10.1145/3654321>
36. Taylor, M., & Brown, A. (2025). Security frameworks for logistics web applications. IEEE Security & Privacy, 23(1), 45–53. <https://doi.org/10.1109/MSEC.2024.347890>
37. Vue.js Documentation. (2025). vuejs.org. <https://vuejs.org/guide/introduction.html>
38. Warehouse Automation Trends Report. (2025). Gartner. gartner.com. <https://www.gartner.com/en/supply-chain/trends/warehouse-automation>

					БР.ІІІ - 17.00.00.000 ІІЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		70

39. Warehouse Management with Cloud Solutions. (2024). dataversity.net.
<https://www.dataversity.net/cloud-based-warehouse-management-systems/>

40. Zhang, Y., & Chen, L. (2024). Performance analysis of QR-code detection systems in industrial warehouses. International Journal of Computer Vision Applications, 17(3), 118–133. <https://doi.org/10.1016/j.ijcva.2024.03.011>

					БР.ІІІ - 17.00.00.000 ІІЗ	Арк.
						71
Змн.	Арк.	№ докум.	Підпис	Дата		

БІБЛІОГРАФІЧНА ДОВІДКА

Тема бакалаврської роботи: " Програмні рішення для автоматизації процесу інвентаризації на складах"

Обсяг пояснювальної записки: 66 аркушів

Дата закінчення бакалаврської роботи 10 червня 2026р.

Підпис студента _____