**INVENTORY MANAGEMENT**

**SYSTEM**

**BY**

**FELIX EHIS OSAWARU**

**2113346**

**INFORMATION TECHNOLOGY FOR ENERGY INDUSTRY**

**ROBERT GORDON UNIVERSITY**

**Supervisor: Dr. Mark ZARB**

**APRIL 2033**

**ACKNOWLEDGEMENT**

This project was developed as part of the Master of Science in Information Technology for the Energy Industry. The joy and accomplishment of completing this project would be incomplete if heartfelt thanks were not conveyed to those whose constant counsel, support, and encouragement made this endeavour possible. I am fortunate to receive aid, support, and encouragement from many people while working on this postgraduate project, and I would want to express my gratitude to them.

First of all, I want to express my gratitude to God for providing me with wisdom, strength, patience, and the opportunity to work on this project. I want to thank the School of Computing Science & Digital Mediaat Robert Gordon University for developing such an outstanding curriculum and urging me to accomplish this work.

Secondly, I want to thank my supervisor, Dr. Mark Zarb. He has been a huge help in giving advice and direction for where this project should go as well as insightful criticism of the work that has been done. It was a pleasure to work with him.

In addition, I want to especially thank my lovely wife, Onome Potokri Osawaru for supporting and encouraging me throughout the project, and for being a Great wife and friend.

Furthermore, I want to say a heartfelt thank you to my siblings Sofia, Michael, Dennis, Tina, and Osakpolor Osawaru for believing in me.

Finally, my greatest gratitude goes to my parents Mr. and Mrs. Osawaru who have directly or indirectly contributed to the successful completion of this project.

**Abstract**

The application is a desktop-based tool meant to monitor inventory information for a company. The major purpose of this project is to build and execute an inventory management application to address a regular organisational demand. Although many businesses still use manual inventory management systems, which may be time-consuming. A computerised inventory system would increase inventory management's efficacy and efficiency. This system may be used to generate daily or weekly sales and inventory reports, preserve inventory detail information, monitor stock levels, and update the inventory based on sales data.

The system was built using VS Code technology, JavaScript, and the Bootstrap framework. PHP (hypertext pre-processor) was utilised for the system's backend development and database management. The user interface of the system was designed and styled using HTML and CSS. The database utilised for this application was MySQL.

Finally, the system will update inventory, keep sales records, provide reports for decision-making, and boost corporate effectiveness.

**Table of Contents**

**ACKNOWLEDGEMENT**.........................................................................................2

**ABSTRACT**.........................................................................................................3

**TABLE OF CONTENT**...........................................................................................4

**LIST OF FIGURES**...............................................................................................7

**LIST OF TABLES**.................................................................................................8

**LIST OF ACRONYMS**...........................................................................................9

**CHAPTER 1: INTRODUCTION**

1.1Background of Project.....................................................................................10

1.2 Project Aims and Objectives............................................................................10

1.3 Problem description.......................................................................................11

1.4 Scope of the project......................................................................................12

1.5 Features of project........................................................................................12

1.6 Relevance of Project......................................................................................13

1.7 Structure of Project.......................................................................................13

**CHAPTER 2: LITRATURE REVIEW**

2.1 Introduction..................................................................................................15

2.2 Description of the Existing system...................................................................15

2.3 Limitation of existing System..........................................................................17

2.4 Review of similar system................................................................................17

2.4.1 Zoho inventory.....................................................................................18

2.4.1 Sortly inventory....................................................................................19

2.5 Summary.....................................................................................................20

**CHAPTER 3: SYSTEM ANALYSIS**

3.1 Background of the system analysis..................................................................21

3.2 Advantage of the proposed system..................................................................21

3.4 Users Requirements......................................................................................21

3.4.1 Functional Requirements.......................................................................22

* + 1. Non Functional Requirement................................................................23

3.5 System Requirements...................................................................................24

3.5.1 Software Requirement..........................................................................25

* + 1. Hardware Requirement.......................................................................25

3.6 Feasibility Study..........................................................................................26

3.6.1 Technical Feasibility.............................................................................26

3.6.2 Economic Feasibility............................................................................26

3.6.3 Organisational Feasibility.....................................................................26

3.7 Functionality of the Proposed System.............................................................27

**CHAPTER 4: SYSTEM DESIGN**

4.1 Physical Design..........................................................................................28

4.2 Logical Design...........................................................................................28

4.3 Architectural Design...................................................................................28

4.3.1 Presentation Tier or User Interface.......................................................29

4.3.2 Business Tier.....................................................................................29

4.3.3 Data Tier..........................................................................................29

4.4 System Design Goal...................................................................................30

4.5 Process or Data Flow Diagram (DFD) ..........................................................30

4.6 Use Case Diagram.....................................................................................30

4.7 Entity Relationship Diagram........................................................................31

4.8 prototype software....................................................................................32

4.9 User Interface (UI)....................................................................................33

4.10 Project Login Page....................................................................................33

4.11 Project Dashboard………………………………………………………………………………………………….34

4.12 Summary………………………………………………………………………………………………………………..40

**CHAPTER 5: TESTING AND DEBUGGING**

5.1 Purpose for Testing and Debugging...............................................................41

5.2 Goals and objectives of testing and debugging...............................................41

5.3 Type of testing..........................................................................................41

5.4 Unit Testing..............................................................................................42

5.5 Test Cases................................................................................................42

5.6 Conclusion……………………………………………………………………………………………………………….46

**CHAPTER 6: CONCLUSION**

6.1 Summary................................................................................................47

6.2 System Benefits.......................................................................................47

6.3 System Limitation....................................................................................47

6.4 Lesson Learned..........................................................................................47

6.5 Future Recommendation...........................................................................48

**REFERENCES**...............................................................................................49

**LIST OF FIGURES**

Figure 2.4.1 Zoho inventory screen…………………………………….……………………………………………19

Figure 2.4.2 Sortly inventory screen………………………………………………………………………….…….20

Figure 4.3: Architectural Tier................................................................................29

Figure 4.5: IMS Process flow diagram....................................................................30

Figure 4.6: IMS Use case diagram.........................................................................31

Figure 4.7: IMS Entity Relationship Diagram...........................................................32

Figure 4.10 Login page of the inventory management system...................................33

Figure 4.11 Project Dashboard………………………………………………………………………………………….34

Figure 4.11 Manage Brand page……………………………………………………………………………………….35

Figure 4.11 Add Brand page………………………………………………………………………………………..……36

Figure 4.11 Mange Categories page…………………………………………………………………………………36

Figure 4.11 Add Categories page………………………………………………………………………………………37

Figure 4.11 Manage Product page…………………………………………………………………………………….37

Figure 4.11 Add product page……………………………………………………………………………….………..38

Figure 4.11 Add order page……………………………………………………………………………………………..38

Figure 4.11 Generate report page...................................................................…...39

Figure 4.11 Add and manage user page………………………………………………………………………….39

Figure 4.11 Change password page………………………………………………………………………………...40

**LIST OF TABLES**

Tables 3.3.1: Functional requirement of the proposed system………………......................22

Tables 3.3.2: Non Functional requirement of the proposed system…………....................23

Tables 5.4: User login test………………………………………………………………………......................43

Tables 5.4: Inventory sales test………………………………’………………………….…......................44

Tables 5.4: Product order test…...........................................................................45

Tables 5.4: User report test………………………………………………………………….…......................45

Table 5.4: Print invoice Test …..............................................................................45

Tables 5.4: Add user test…..................................................................................45

Tables 5.4: Search item test….............................................................................46

Table 5.4**:** Logout the system..............................................................................46

**LIST OF ACRONYMS**

IMS Inventory Management System

DBMS Database Management System

DFD Data Flow Diagram

MSc Master of Science

SQL Structured Query Language

ERD Entity Relationship Diagram

ITEI Information Technology for Energy Industry

UI User Interface

UCD Use Case Diagram

VSC Visual Studio Code

PHP Hypertext Pre-processor

CSS Cascading Style Sheets

HTML Hypertext Markup Language

RAM Random Access Memory

CPU Central Processing Unit

SME Small and Medium Enterprise

DOM Document Object Model

RFID Radio Frequency Identification

**CHAPTER 1.0**

**INTRODUCTION**

**1.1 Background of the project**

The project is a purchase and stock management desktop-based application designed using Microsoft visual studio software. The system will maintain track of an organisation's Products and sales by managing flow, keeping a thorough record of new and sales goods, and ensuring that proper quantity inventory is always accessible. The system will replace manual inventory documentation, enhance quality control, and keep track of stock, retail, and various sorts of sales management for a single branch.

The inventory management system is an application that runs without the Internet. The primary aim of the application is to track and manage stock such that there is neither excess nor understocking. The application also includes product information, order information, and items that are still available. There is also a provision for updating the inventory based on the sale information, which provides details on the balance transaction and the stock balance. The system can produce order receipts and generate reports on orders and stocks. The application includes a user component to manage the inventory and maintain the inventory system with the login page developed to safeguard the organisation's stock management from theft and improper use of the inventory.

**Project Aims and Objectives**

The primary goal of this project is to put in place a system to manage all of the company's present sales and inventory-related functions. By doing this, the company's physical labour will be reduced. The objectives of this project are measured below:

* To develop and implement an inventory management application to solve an organisational day to day requirement
* To test and validate the system, ensuring all relevant requirements are met.
* To provide details information about product, order, and stock balance.
* To improve cash flow, visibility, and decision-making.
* To create receipts in the appropriate formats for client referencing
* Reduced data redundancy
* To provide a function for more effective inventory management in the store. Basic data management operations such as 'add,' 'delete,' and 'edit' will be provided.
* To provide notifications to users when there is a shortage of inventory in the store.

**1.3 Problem description**

Inventory management is one of the critical departments that must be successfully handled to enable the seamless operation of everyday business activities in any organisation. Sadly, according to historical study, the majority of small and medium-sized businesses lack the essential tools; they continue to underestimate the importance of inventory management system, with some still running their inventory management manually. As a result, practically all inventory-related documents or objects are unprotected, daily sales and transactions are relatively low, and manual registration of products and sales takes a long time. As a result of poor sales and inventory management, several small and medium-sized businesses struggle to determine the amount of each item sold each day and the level of products available.

Another issue that has been noted is the inability of the majority of current inventory management systems to compare inventory items and sales on a daily, weekly, and monthly basis. Also, clients who made purchases at the company often do not receive receipts printed from the database as references. Instead, manual receipts are issued. In addition, the existing technology serves to compute the total amount of each customer's purchases manually; it is unable to provide a report of sales at the end of each day. Due to the lack of suitably guidelines, several types of reports are created in the business logbook regularly. The total goods and sales figures based on the logbook are readily falsified, which makes it exceedingly difficult for the business owner to catch theft. Given that a database does not exist to store the daily total sales, all of these issues must be addressed.

Before building this web application, there were some challenges needed to be solved. Below is the problem statement:

* To provide users access to a reliable, useful, and platform-independent application for small and medium scale enterprise.
* To ensure that functional and non-functional requirements like security, quality, scalability, and maintainability are met.
* To put the practical information gained throughout the Master's Degree Program into practise.
* To create a tool that compare inventory items to sales
* To make the system user friendly, accurate, and operational.
* To develop a system that efficiently, and accurate keep track of products and sales, and meet the organisational goals.

**1.4 Scope of the project**

An inventory management system (IMS) is designed for small and medium-scale organisations with a different type of sales management for one branch. The different reports and configurations make it suitable for small and medium enterprises (SMEs). The scope of this application is:

* Only authorised users are responsible for managing the system. That is updating, editing, managing products, orders, and recording inventories
* To find out how many goods are in stock, use the system to generate daily and weekly reports.
* A fast and user-friendly interface with security driven

This programme will be built as a desktop system that will not require an internet connection to function. The first stage is to evaluate another current system and build a viable system. The second phase is to decide on the methodology and tools to be utilised, and the last step is to put the web application into action and test it.

**1.5 Features of project**

The proposed system will be used to manage and monitor sales and purchases of inventory. The main objective of inventory management is to increase the ease and effectiveness of ordering, stocking, storing, and utilising goods for businesses. If you manage your inventory well, you'll always be aware of what is in stock, how much there is of each item, and where it is. Keeping track of your inventory and sales is the fundamental purpose of an inventory system. Below are the main components of the web application.

**User Login:** To access the main dashboard, the user must provide the username and password. The authorised user has the full ability to add, edit, and remove from the organisational stock as needed.

**Dashboard:** The Inventory Dashboard displays metrics that indicate the condition of your inventory, such as the total items that are in stock or items that are low in stocked. In addition, the dashboard also includes inventory categories, brands, product, order, and report, add user and logout page.

**Report Management:** The fundamental goal of management reporting is to gather the necessary information about the products cost, sales, and products order report in other for the organisation's monitor operational outcomes prepare for future control.

**Inventory Category:** This page will accommodate the correct item at the correct time and in the correct amount. This will make it simple for business owners to determine how much inventory is overstocked or understocked.

**Add User:** When a new user is registered, the user information is saved in the database.

**Logout:** The user may exit the programme using this module. The user's exit prevents further procedures from being carried out.

**1.6 Relevance of Project**

The Inventory Management System applies to small and medium-sized enterprises. Many SMEs continue to manually manage inventory rather than adopt a computerised system to assist business owners with product and inventory order. It is why a systematic inventory management system is required to address the shortcomings of the present manual business methods in SMEs.

**1.7 Structure of Project**

This project's introductory chapter provides a summary of the work. The project's aims, objectives, description of the project, scope, features, relevance, and limitation of the project will all be covered in length in this chapter.

The second part of the report's literature review will give a critical assessment of existing similar systems and technologies for implementing this system. This chapter will go through the limitations of the several existing systems that are available.

The third chapter which is system analysis outlines the present process and functional and non-functional requirements, system requirements, feasibility studies, and project functionality. Including hardware and software used.

The fourth chapter will go more into the 'Design' answer to the problem. The design chapter goes into depth on how the proposed system was created. This chapter includes a process diagram, a use-case diagram, an ER diagram, as well as user interface designs. The design implementation will explain the system implementation, and activities performed during system implementation.

The fifth chapter goes through how the system was tested. This will explain the test technique that was used to examine and validate the system, as well as the implications of errors, key source codes, and module structures of systems as well as the data structure, outcome, and platform reliance of the programme.

The last chapter will discuss the successes, setbacks, and lessons learned from this project. Moreover, ideas for potential future advancements ought to be explored.

**CHAPTER 2.0**

**LITRATURE REVIEW**

**2.1 Introduction**

The main goal of inventory management, according to Mogorosi (2023), is to get the appropriate inventory at a suitable place at the proper time, in the right amount, in the right shape, and at the proper cost. Hence, when developing an inventory management system, there is a tendency to assume that its primary purpose is to hold both completed items that are ready for sale and the raw materials or products utilised in the transformation process (Chase et al. 2005). It was predicted that an inventory would be the manufacturing sector's largest investment. This inventory has to be handled effectively, which includes treating it appropriately so that it may be inspected as needed (Handfield, Monczka, Giunipero and Patterson 2011).

Furthermore, Gill, Khullar and Pal Singh (2016) claim that the most important task for organisations is inventory management. It is an essential component and brain centre for every organisation. In addition, Oladele et al. (2021) state that inventory management systems track inventory levels, demands, transactions, and supply. This may also be utilised in manufacturing organisations to generate a job request, resource invoice, and several interconnected production records.

According to Alef (2017, p.21-25), "inventory systems are designed to facilitate identification, classification, and documentation, in order to conserve the assets and integrate them into sustainable research and planning processes". According to A., Fatoba, and Abisoye (2023), inventory management is the process of successfully managing the continual flow of units into and out of an existing stock of goods. As a result, it is vital to have a computer-based inventory management system capable of creating reports, keeping track of stock levels, and giving information on the company's purchases and sales. There are other existing systems. This chapter will look at a few different systems.

**2.2 Description of the Existing system**

According to A., Fatoba and Abisoye (2023), the phrase "inventory management" is commonly used to refer to the practice of efficiently regulating the continuing entry and departure of units from a stock of products. Inventory management strives to keep stocks at the lowest feasible cost while ensuring a continuous supply for current activities (Mpanywa 2005). Therefore, inventory and sales management have always been critical procedures for business owners; and there has always been a need to create a better method to manage inventory and sales.

The ability to count tally is a prerequisite for learning arithmetic and other fields of mathematics, and it is a component of many regular and niche activities. Counting and tallying skills are thus possibly the most fundamental numerical skills used for keeping track of things even before the industrial period (Martin and Lynch 2009). Humanity created the earliest inventory management system some 50,000 years ago, when "tally sticks" were employed for counting for the first time. In addition, archaeologists have discovered evidence of the usage of clay tokens dating back as far as 4,000 years. These had symbols burned into the clay and were used to record things like lambs and other livestock. Throughout antiquity, especially in the civilizations of the ancient Greeks and Egyptians, inventory management evolved into slightly more exact accounting and record-keeping systems.

In 1884, an American inventor created the first modern mechanical calculating machine during the Industrial Revolution. According to Heide (1997), the tabulator and sorter system was created to collect data using punch cards, doing away with the need for pen and paper and saving countless hours of labour. Anybody may use these punch cards to record data, including inventories. A customer fills out a punch card in a store, the system reads it, sends the information to the warehouse, and finally, the item is delivered to the customer. The application may manage the transaction's financial and inventory management components by scanning a database or catalogue for goods.

After that, an American inventor named Norman Woodland 1940 created barcode technology in response to a plea from a concerned grocery store owner who needed help keeping track of inventory. "The Barcode technology is used in various areas of applications in computerising operations to achieve efficiency, and effectiveness and realise optimal benefits from the business by scanning the inflow and outflow of barcoded items using a scanner" (Muyumba and Phiri 2017 p.113). Barcodes and barcode readers, often known as scanners, are used in the technology. In his article on barcode sales and inventory control, Dolinsky (2010), argues that in the early days of inventory keeping, merchants wrote down purchases or counted how many units were left at the end of the day to forecast future requests. Manual inventory tracking has given way to product scanning. In contrast, data entering into computers was still done by hand. For a long time, the only way to keep track of inventory and sales was to take inefficient and incorrect handwritten notes.

In addition, another technology called radio frequency identification (RFID) was created in the 1970s. This technology uses radio waves to identify objects, and a microchip with an antenna includes a set of identifying numbers, and the microchip and antenna—also known as an RFID tag—are read and recorded on a computer using a chip reader (Radio Frequency Identification and Reader 2023). This method can help organisations, notably merchants and logisticians meet future difficulties. RFID also boosts sales and profitability by saving time and increasing stock accuracy.

For a long time, all this software method was the possible historical means of keeping and tracking inventory, including manual recording of sales inventory for small and medium enterprises. Nevertheless, this was no longer an effective, safe, and correct way of managing inventory and sales as technology advances. Therefore, they were a need for businesses to begin investigating and deploying advanced inventory management software to keep records of the complete items, improving cash flow, visibility, and decision-making.

**2.3 Limitations of existing System**

The goal of an inventory management system is to have items in the correct place at the right time. This involves restocking items as soon as possible, having the proper resources in place, and having an effective mechanism to receive and keep records in a computerised system (Mogorosi 2023). As we all know, ineffective or unsafe ways of managing items and sales are time-consuming, inefficient, and inaccurate compared to computerised systems.

The old system's ineffective inventory management is owing to various shortcomings, which can enable businesses to make expensive errors. The following limitations are stated below

**Cost of System:** The system is expensive to use. This cost cannot be justified for small and medium-sized businesses. This has resulted to system of updating company's inventory manually which is time-consuming and can lead to inefficient processes.

**System Security:** There can be malicious setup by cybercriminals. Old systems are not designed to handle the threats of today’s industry and can be vulnerable to hackers. This hacker could gain access to the operating system.

**System Accuracy:** Throughout history, inventory management has been rudimentary and has given little precision. This may increase the risk of the businesses running out of crucial items.

**2.4 Review of similar system**

As mentioned above, “Inventory management is an important aspect of any successful business. It is the process of overseeing and controlling the flow of inventory units a business uses in the production or manufacture of goods for sale or distribution” (Melanie 2017). Competent management of the commodities, raw materials, and final products that comprise inventories is critical to maintaining appropriate stock levels and maximising the organisation’s earning potential. It also allows a corporation to avoid or reduce inventory-related losses. Companies use inventory management software for several objectives, including tracking inventory costs during the production and selling processes, suggesting when to restock, and tracking earnings. Furthermore, it may be used to forecast product demand, inventory levels, and prices.

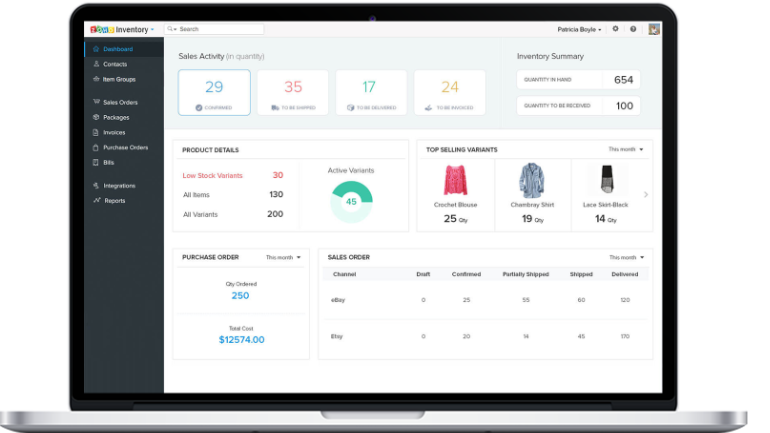
Therefore, these are to categorise and track the company’s entire inventory from the moment of acquisition to the point of sale. It also helps you decide what to get when to order it, and how much to order. The tool also simplifies additional duties like billing, accounting, financial reporting, preparing reports, and many others.

**2.4.1 Zoho inventory**

Sridhar Vembu is the founder and CEO of Zoho, a privately held cloud-based business software company. Zoho has grown 38% in the last year to reach 80 million global users by July 2022. Zoho is a customer relationship management programme that is accessible through the Internet. It is ideally an office suite that includes tools like word processing, automation, presentations, project management, invoicing, and more. Furthermore, this award-winning platform is trusted by over 250,000 enterprises in 180 countries. This makes the online web app more desirable to the company owner.

Zoho Inventory offers a user interface that is straightforward to use, making it ideal for small and medium-sized retail firms. It allows you to manage orders, delivery, purchase orders, sales, invoicing, and shipping. Using Zoho, you can monitor the movement of different items' stock, including every item in every warehouse. Moreover, it offers barcode scanning to guarantee the correct things are sent and packaged. Several e-commerce platforms, such as Amazon, Shopify, Etsy, and eBay, are compatible with Zoho Inventory. The limitations of this software are

* Printing requests is not possible. Hence, when a purchase or transaction is completed, the only option is to capture a screen shot.
* It is only limited if you are working with stock items.
* It is internet-based software. That is, you cannot access it if there is no internet.

****

**Figure 2.4.1 Zoho inventory screen**

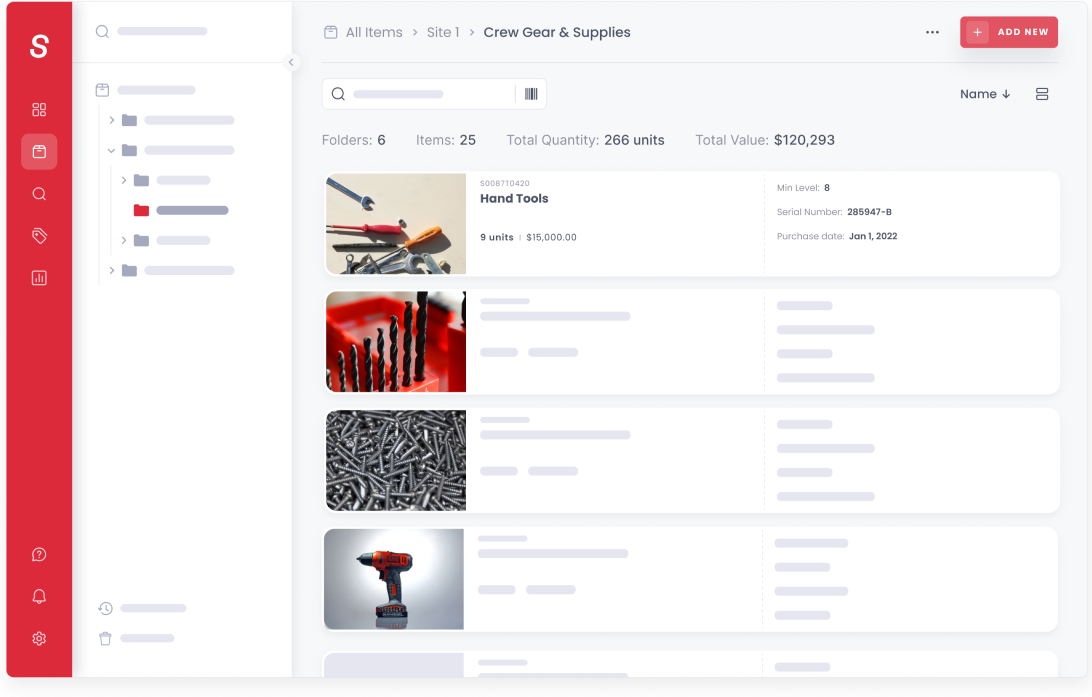
**2.4.2 Sortly inventory**

Sortly is a simple inventory and asset tracking system that allows you to visually manage items and any of their information, such as amount, price, condition, remarks, and so on, enabling a more natural and less difficult manner to keep inventory across several locations. One noticeable feature is that Sortly is users friendly, unlike other inventory monitoring systems. It is designed so that you can set it up and use it independently in matters of minutes. The system will eliminate time-consuming, difficult spreadsheets and inventory management so you can focus more on running your business.

Another noticeable feature is that Sortly allows you to monitor user activity as well as activity at the account, folder, and items levels. You may also filter, categorise, and export the activities based on the activity category or a certain time window. You may filter operations depending on whether an object is moved, amended, removed, produced, restored, had the quantity modified, or is combined. A date range, as well as the most recent or oldest data, can be used to filter the data. The search box may be used to find a certain security identifier (SID), items, user change, or other criteria. This is a significant and cutting-edge feature when compared to most similar software since it helps one to expand their business internationally.

Sortly aids in keeping everything organised so that you can quickly locate items easily. It was founded in 2013 by Dhanush Balachandran and has its headquarters in Redwood City, California. One advantage of this application is that it can be accessed from most mobile devices. There are very few disadvantages to this online application.

* It does not notify the administrator when certain items are deleted from the system.
* It takes a lot of time to become familiar with the program's functions.



**Figure 2.4.2 Sortly inventory screen**

**2.5 Summary**

This chapter gives an overview of the previous system, a study of related systems, and a discussion of its shortcomings. This proposed system application gains knowledge from the already information acquired from existing software. This system will improve facilities and overcome all of the limitations of the existing system. The system will apply all security criteria and allow the user to work in a user-friendly way, allowing the user to perform his or her responsibilities as soon as possible. All of this information, as well as the system's expectations, will be extensively examined in the next chapter.

**CHAPTER 3.0**

**SYSTEM ANALYSIS**

**3.1 Background of the system analysis**

As stated in the previous chapter, the existing system has some limitations. As a result, system processing, updating, and controlling inventory may be challenging. To address the limitations of the existing system, a new system must be developed that can improve the current system's status. The new system should be concerned with meeting the client's needs; it should be dependable, simpler, faster, and more informative. As a result, the prior system must be thoroughly understood and outlined.

Thus, system analysis is the process of researching a company's environment to come up with a systemic solution to an issue or to make changes to such a situation, or the act of examining a method or organisation to identify its goals and purposes and developing systems and processes to achieve them. The system will fulfil the demands of the user. The system will be more secure, dependable, user-friendly, responsive, and informative.

The following qualities are considered when developing the proposed system:

* To develop and implement an inventory and sales management application to solve an organisational day to day demand
* The develop a system with faster response time
* To test and validate the system, ensuring all user requirements are met.
* Assess the feasibility of the system idea.
* To improve cash flow, enhance accountability, and decision making.
* Error reduction
* To create receipts in the appropriate formats for customer referencing
* Reduced data redundancy
* To implement security measures and prevent unauthorised access to user records

**3.2 Advantage of the proposed system**

* A better user experience
* The technology will improve the work of clients who access software, resulting in increased productivity.
* The system will have a secure login and easy for the user to operate, and no unauthorised people are able to access it.
* It will increase organisational efficiency and reduce the time required for various procedures, resulting in increased efficiency and better service.

**3.3 Requirements system analysis**

A system or software project's success or failure depends on the requirements analysis. The requirements have to be well-documented, usable, quantifiable, testable, traceable, tied to recognised business opportunities or needs, and sufficiently specified for system design. As a result, the system analysis requirements are critical while developing the system.

**3.4 User Requirements**

User requirements are precise declarations of what a user wants or expect from a system. Users' needs are frequently developed during the project's requirements analysis stage, and these requirements serve as the foundation for designing and producing a solution that fits the demands of the target audience. User requirements are critical for software development, product design, and service delivery because they guarantee that the result fulfils the user's wishes and expectations. Natural language or other types of documentation are used to establish user needs, which may comprise a variety of aspects such as functional and non-functional requirements.

**3.4.1 Functional Requirements**

Functional requirements outline the features or services that the system must offer. That is how a system should behave, the steps that must be taken by the system to meet user needs or expectations. The table below gives a breakdown of the functional requirement

|  |  |  |
| --- | --- | --- |
| **FR** | **Requirement Statement** | **MoSCoW** |
| User management | The system must allow the authorised user to login and update his or her password and save the change. | Must |
| The system should be capable of adding new users, editing existing users, and removing existing users. | must |
| Users can log onto the system with an authorised username and password. | Must |
| A "Forgotten Password" function should be available to users. | Should |
| Inventory  Management | Users can add brand, categories, and product inventory. | must |
| Users can edit/remove brand, categories, and product inventory. | must |
| Product Management | Users can order item filling the order form in the system. | must |
| Add a product image from the system gallery. | Must |
| Payment Management | This section deals with the creation of invoices and the printing of receipts. | Must |
| User ability to select from different payment options and related information. | Could |
| Report Management | The system shall have a page called “report” where authorised users can review sales and inventory report on a daily, weekly, and monthly basis. | Must |
| Backup Management | The system and database must be backed up to an external host and recovered in the event of a software system failure. | Should |
| Search Icon | The system shall have a search icon button where users can search for items in stock | Could |

Table 3.4.1 Functional requirement of the proposed system

**3.4.2 Non Functional Requirement**

Non-functional requirements explain the system's qualitative characteristics, as well as its operational capabilities and limits, and strive to boost its functionality. These are the specifications that specify how well it will function, such as speed, security, scalability, efficiency, data integrity, and so on. The table below lays out the functional requirements.

|  |  |  |
| --- | --- | --- |
| **NFR** | **Requirement Statement** | **MoSCoW** |
| Security | The system must be safe from hackers and protected from unauthorised users. | Must |
| Those that have access to the system will be able to change their passwords at any time. | Must |
| Functionality | The system must be capable of detecting error handling, reporting errors on primary key duplicates, duplicate usernames, and passwords that are out of range. | Must |
| Compatibility | The system must have a responsive UI design that works in both web browsers and mobile devices. | Must |
| Usability | The system shall be user-friendly | Must |
| The interface transaction navigation prompt must be displayed by the system. | Must |
| Availability | The system must be available 24 hours a day, seven days a week, and accessible from anywhere. | Must |
| Scalability | The application shall be able to function well in other adapt to new changes. | Must |
| Performance Efficiency | System shall have a high response time. It should not take more than 2 seconds to respond to user request | Must |
| System must perform 100% daily order and sales transaction within a given period of time. That is it should not take more than 5 seconds to load information | Must |
| Notifications | Notifications for low stock levels | Must |
| Faulty Recovery | In the event of a failure, the system should be able to recover quickly. | Must |
| Accuracy | The system must be precise since its handles financial transactions. | Must |

Table 3.4.2 Non Functional requirement of the proposed system

**3.5 System Requirements**

* + 1. **Software Specification**
* **Microsoft Visual Studio 2010:** VS code is the best development environment available (IDE). It can create computer programmes like system websites, web applications, and mobile applications. Microsoft's Silverlight, Windows Store Operating System Presentation Foundation, Operating System API, and Operating System Forms, as well as other software development platforms, are all used by Visual Studio. It is capable of managing code. Microsoft Studio's code editor supports both code refactoring and IntelliSense, a code completion capability. The integrated debugger supports both source-level and machine-level debugging. When you choose a symbol, all other instances of that symbol in the Microsoft Studio code editor are highlighted. It also has a Quick Search tool that allows users to rapidly search all symbols in C++, C#, and VB.NET projects. Quick Search includes case and substring searches. With the Call Hierarchy functionality, developers may look at all methods that call the current one as well as methods that call the current one.
* **PHP Coding Language:** is a server-side programming language that is open-source and free that can be used to build a variety of applications, including websites, mobile apps, CRMs, and others. HTML may incorporate this well-known language. Due to its HTML interoperability, the PHP programming language has maintained its popularity among developers. The creation, management, and optimisation of business data and analytical applications are made simpler with SQL Server. It functions as an enterprise data management platform, with a centralised administration panel that allows data administrators from anywhere in your organisation to monitor, administer, and optimise all databases and associated services. It provides extensible management architecture based on SQL management objects that are simple to design, allowing users to customise and expand their management environment and independent software suppliers to add new features and tools to enhance the built-in capabilities.
* **PhpMyAdmin:** is a database administration tool for databases that are MySQL compatible. It is a free and open-source database management tool for MariaDB and MySQL. It is the most used MySQL administration tool, especially among web hosting companies. The majority of administrative tasks, including setting up databases, performing queries, and creating user accounts, may be handled using phpMyAdmin. The deployment, maintenance, and optimisation of corporate data and analytical applications are made easier with SQL Server. It functions as an enterprise data management platform, with a single administration panel that lets data managers from throughout your organisation track, monitor, and optimise all database and related services. It offers an extensible management architecture that is simple to programme using SQL management objects, allowing users to customise and extend their management environment and independent software vendors to provide additional tools and functionality to supplement the pre-built capabilities. Database administrators can inspect server objects, manage objects, monitor system activity, and seek online help in addition to authorising and conducting queries.
* **Microsoft Windows 10:** The operating system is responsible for allowing you to operate a computer. Windows comes preloaded on the vast majority of new personal computers (PCs), contributing to its position as the world's most popular operating system. Windows allows you to do a range of routine tasks on your computer. For example, you can use Windows to browse the Internet, check your email, edit digital photographs, listen to music, play games, and do other things. Since it provides access to productivity tools like calendars, word processors, and spreadsheets, Windows is also widely used for businesses.
* **Bootstrap:** Is an HTML, CSS, and JavaScript framework with built-in features like navigation bars, forms, buttons, and typography, as well as CSS layout, colour, and spacing styles. These components and styles are designed to operate together so that web designers may create professional-looking web pages fast and easily. One of Bootstrap's most important features is its responsive design. Developers may utilise Bootstrap to create websites that dynamically adjust to multiple screen sizes and devices, such as desktops, laptops, tablets, and smartphones. It makes it easier to create web pages that are helpful and accessible on a wide variety of devices.
* **JavaScript:** is an advanced programming language often used to construct server-side programmes, desktop programmes, and web apps? Because JavaScript is dynamically typed, variable types are decided at runtime rather than during compilation, which promotes flexibility but needs more cautious coding practises to avoid unexpected behaviour. One of JavaScript's most important features is its ability to modify the Document Object Model of a web page, allowing programmers to create dynamic and interactive websites. Data validation, form validation, and animation are just a few of the many functions it may perform. The fundamental ideas of JavaScript include variables, functions, objects, arrays, loops, conditionals, and events. It also has a variety of built-in functions and methods that may be used to do typical tasks, such as manipulating strings and arrays, decoding JSON data, and handling dates and timings.
* **JQuery:** It is fast, small, and packed with features. It substantially simplifies activities such as HTML page navigation by offering a user-friendly API that is compatible with a wide range of browsers. By combining flexibility and extensibility, jQuery has altered the way millions of people write JavaScript. The web project's files and open-source framework are freely available.

**3.5.2 Hardware Specification**

* 260MB Random access memory (RAM)
* 8GB Hard Disk
* Central Processing Unit (CPU)
* Flash drive for file sharing
  1. **FEASIBILITY STUDY**

The feasibility study detailed the project's benefits and adaptability. This programme is divided into three parts and has been tested for several feasibility criteria.

* + 1. **Technical Feasibility**

The most important part of the proposed system is its technological aspect. The interface and database functionality will be built using Visual Studio for the front-end design and MySQL for the back end, which will connect the system interface to the data storage. The system will be relatively easy to maintain.

* + 1. **Economic Feasibility**

Economic analysis is the most commonly used approach for determining the efficacy of a proposed system. The proposed solution consists of desktop programmes that can be used even when the internet is not available. The method was created to make money. Increased efficacy and efficiency, rapid automated setup, minimal time-consuming aspects, and superior performance quality are among the benefits. Customers will appreciate the system's versatility and ease of use.

* + 1. **Organisational Feasibility**

This level of practicality focuses on who can use the system. The administrators are the system's users. Administrators will have better access to inventory data and will be able to manage it. The user generates daily and weekly reports to monitor business performance.

**3.7 Functionality of the Proposed System**

* The proposed system does not provide data replication.
* The proposed system has a user-friendly interface.
* The proposed system allows users to access data from the server for the specified sales and supply order numbers of stocks.
* The proposed system provides users with relevant results in response to their requests.

**CHAPTER 4.0**

**SYSTEM DESIGN**

System architecture refers to the pieces that work together to achieve a certain purpose as effectively as possible. As it serves as the basis for all software, the system designer must carefully analyse and prepare for it. It may become clear that effective systems require much preparation and thought in order to prevent design flaws. The system's overall performance might suffer as a result.

Software design is the foundation of software engineering. While developing a good system design, every component of infrastructure must be considered, from hardware and software to data and its storage. There are three design tiers.

**4.1 Physical Design**

The physical architecture is tied to the system's real input and output techniques. This might be represented by the manner in which data is entered, validated or authenticated, processed, and displayed.

**4.2 Logical Design**

An abstract illustration of the system's inputs, outputs, and flows of data is what is meant by a system's logical design. This is often accomplished through the use of an exceedingly abstract and occasionally graphical depiction of the actual system. In the logical design process, diagrams illustrating entity relationships are employed.

**4.3 Architectural Design**

The architectural design of a system puts a focus on developing the architecture of the system, which describes the system's structure, behaviour, and other system viewpoints and analyses. This desktop application is built using a three-tier design.

**4.3.1 Presentation Tier or User Interface**

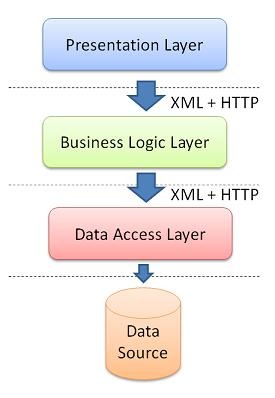
The presentation tier of the system software is the layer for communication and user interface, where end users engage with the application. This tier's primary purpose is to respond to user requests for information that are initiated by user inputs.

**4.3.2 Business Tier**

The business tier is often referred to as the logical or intermediate layer. In this tier, the data gathered at the presentation layer is processed. It offers the system's essential functions as well as communication to the data tier, which simplifies activities performed by the presentation tier.

**4.3.3 Data Tier**

The data layer, commonly referred to as the back end, is the location of the application's data. The database backend stores data that may be accessed using MySQL Database Connectivity. MySQL database connections are used to send complex database queries between the business tier and the backend database.



**Figure 4.3 Architectural Tier 3 Designs**

Source: <https://weblogs.asp.net/fredriknormen/using-web-services-in-a-3-tier-architecture>

**4.4 System Design Goal**

* To make the system dependable, comprehensible, and affordable.
* To make the system compactable, and remove the old system's limitations.
* Make system user-friendly.

**4.5 Process or Data Flow Diagram (DFD)**

A data flow diagram, sometimes known as a flowchart, is a graphical representation of how the automated inventory system should work. It displays the techniques used as well as the data flow. It is also a useful way for modelling process features in a system with high-level detail by describing the types of data that will be used and how they will be inputted into and outputted from the system. The data flow diagram reflected our process model.The data Flow Diagramis illustrated below:

USER LOGIN

REMOVE INVENTORY

DOWNLOAD RECIEPT

PRINT INVOICE

ADD/MANAGE ORDER

SYSTEM DASHBOARD

CATEGORIES

LOG OUT

EDIT INVENTORY

BRAND

ADD INVENTORY

Product

ORDER

MANAGE RECORDS

ORDER

REPORT RECORD

ADD USER

MANAGE USER

USERS DETAILS STOREED IN DATABASE

BOOKING DETAILS

**Figure 4.5: IMS Process flow diagram**

**4.6 Use Case Diagram**

A use case diagram (UCD) shows visually how a user could interact with software. The many use cases and user categories for the system are shown in a use case diagram. The main objective of a use case diagram is to show which system functions are executed for which actors.

CATEGORIES

C

USER

**Figure 4.6: IMS Use case diagram**

**4.7 Entity Relationship Diagram**

An entity-relationship diagram (ERD) is a data modelling method that employs visuals to depict the entities and interactions in an information system. It represents how data is saved in the system's databases and files. The Entity Relationship Diagram is shown below

Order Item

Order\_item\_id: int(11)\*

Order\_id: int(11)

Product\_id: int(11)

quantity: varchar(255)

rate: varchar(255)

total: varchar(255)

Order\_item\_status int(11)

Product quantity

USER LOGIN

User\_id: int (11)\*

Username: varchar(255)

Password: varchar (255)

Email: varchar (255)

Inventory Orders information

orders\_id: int(11)\*

user\_id: int(11)

client\_name: varchar (255)

client\_contact: varchar (255)

sub\_total: varchar (255)

vat: varchar(255)

total\_amount: varchar (255)

Discount: varchar (255)

grand\_total: varchar (255)

paid: varchar(255)

due: varchar(255)

payment\_type: int(11)

payment\_status:int(11)

payment\_place: int(11)

gstn: varchar(255)

order\_status: int(11)

order\_date:date

Product quantity

Product

Product\_id:int (11)\*

Product\_name: varchar (255)

Product\_image: text

Categories\_id: int(11)

Quantity: varchar(255)

Rate: varchar(255)

Active: int(11)

Status: int(11)

Product quantity

Inventory Categories

Categories\_id: int (11)\*

Categories name: varchar (255)

Categories active: int(11)

Categories status: int(11)

Product quantity

Inventory Brands

Brand\_id: int (11)\*

Brand name: varchar(255)

Brand active: int(11)

Brand status: int(11)

Admin

User\_id

User\_name

User\_Address

User\_Telephone

**Figure 4.7: IMS Entity Relationship Diagram**

**4.8 prototype software**

The Inventory Management System is window desktop software. Because the application's interfaces have been simplified, even people with less technical knowledge may utilise it.

**4.9 User Interface (UI)**

User Interface improves the user experience by creating user interfaces for hardware and software found in computers, home appliances, mobile phones, and other electronic devices. To fulfil user goals, user interface design attempts to make user interactions as simple and effective as possible. A poor interface design will either increase or hinder a user's ability to use the functionality of an interface. Good user interface design changes how the user performs some interactions and improves the design's aesthetic appeal. Design requires the balancing of technical utility and aesthetic components. To do this, a system must be created that is not just functional but also useable and flexible to changing user expectations.

**4.10 Project Login Page**

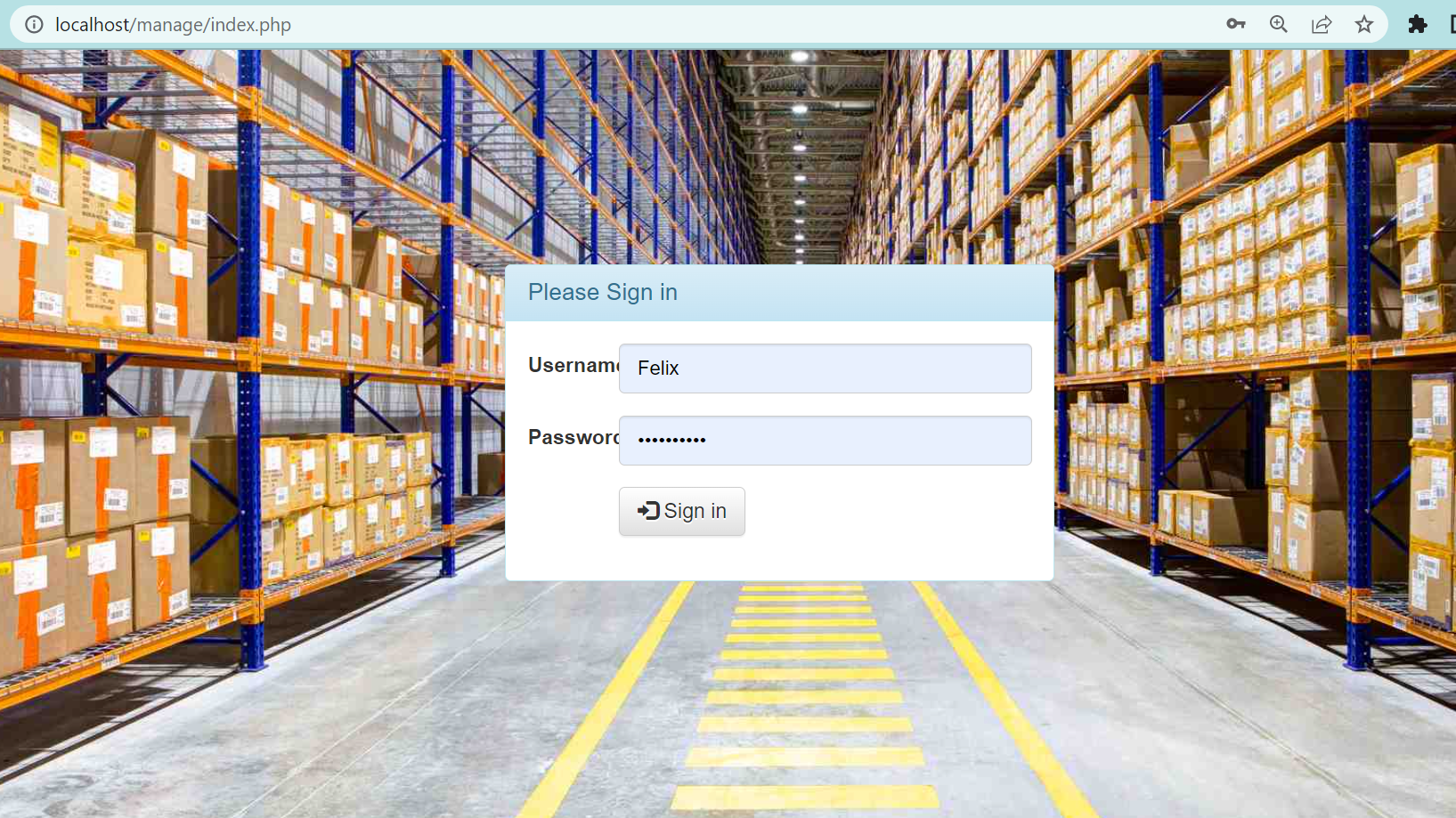
This is the primary user interface and is necessary for security. The system has a

visually appealing design and a background picture that immediately gives it a

professional appearance. The system is only accessible to authorised persons. Before

being allowed access, the user is requested for their username and password; if an

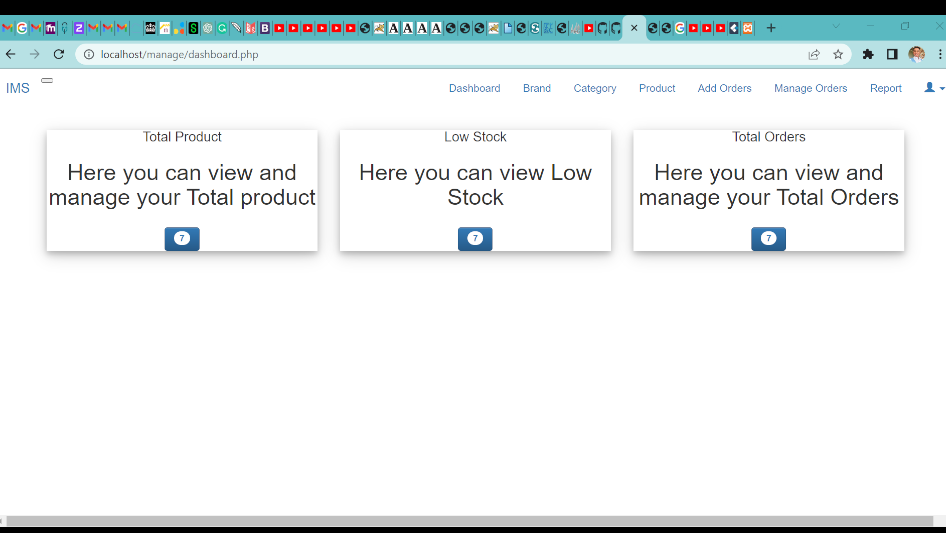
account has not yet been created; the user is unable to access the system.



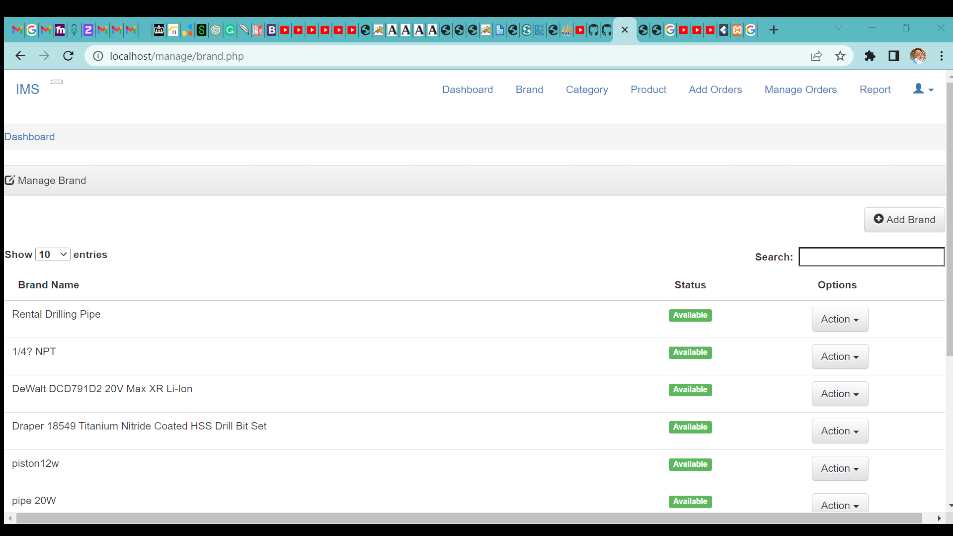
**Figure 4.10 Login page of the inventory management system**

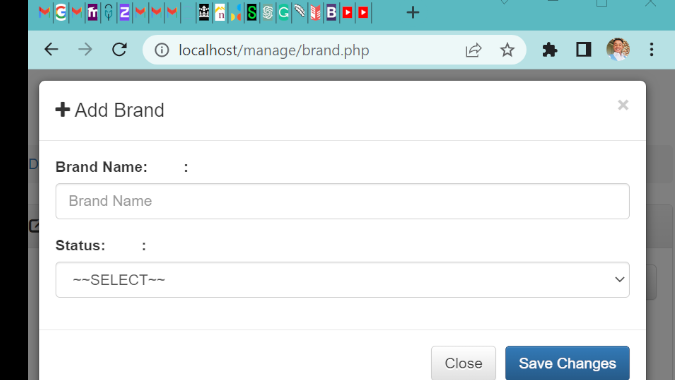
**4.11 Project Dashboard**

After a successful login attempt, the user will be sent to the application dashboard page, where they may select from the menu options of brand, categories, product, order, total product, low stock, and total order. The index menu is where brands, categories, goods, and orders are managed and added. The user can add, update, or remove inventory. When needed, the user may also make product orders, generate invoices, and generate sales reports. With the assistance of the admin, a new user can change and manage their password at their space, and vice versa.

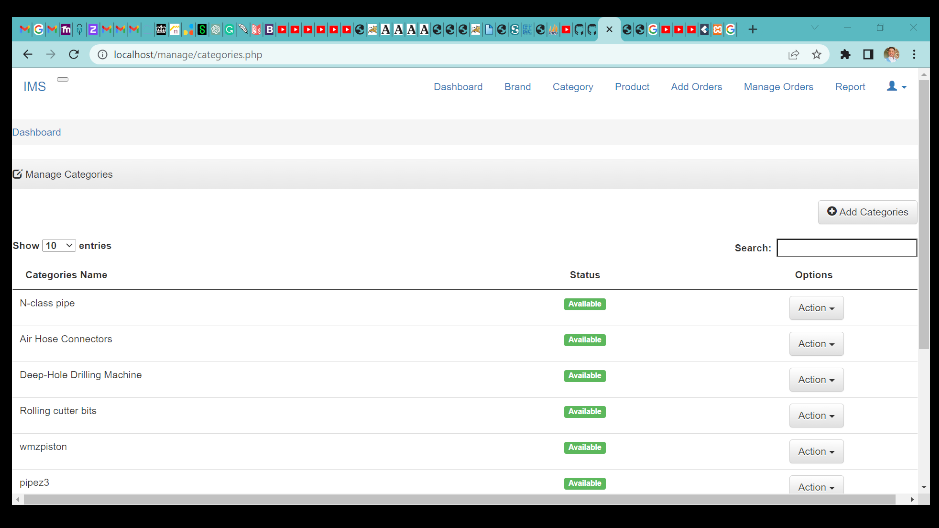


**Figure 4.11 Dashboard page of the inventory management system**

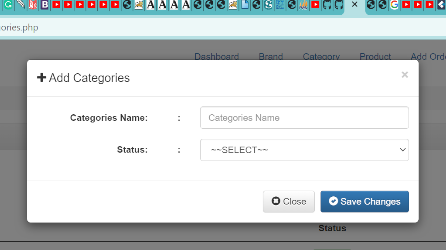
**Figure 4.11 Manage Brand page**

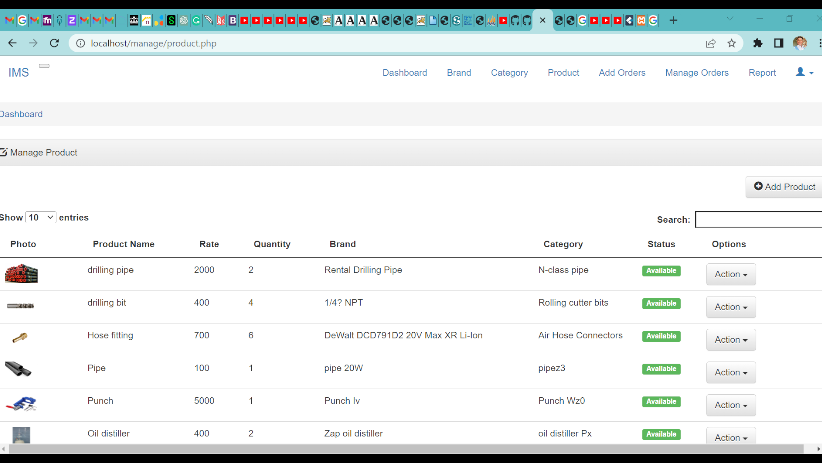
****

**Figure 4.11 Add Brand page**

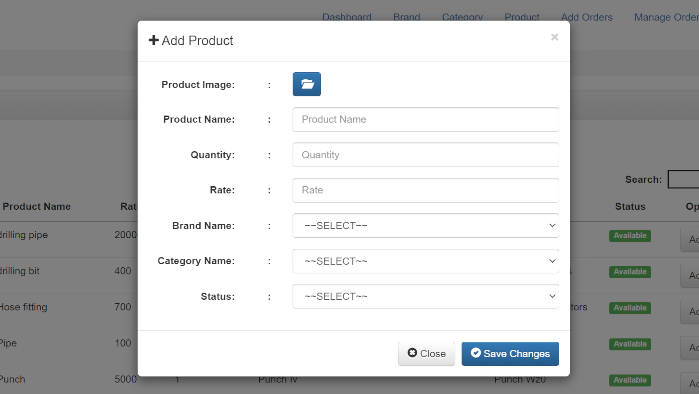
****

**Figure 4.11 Manage Categories page**

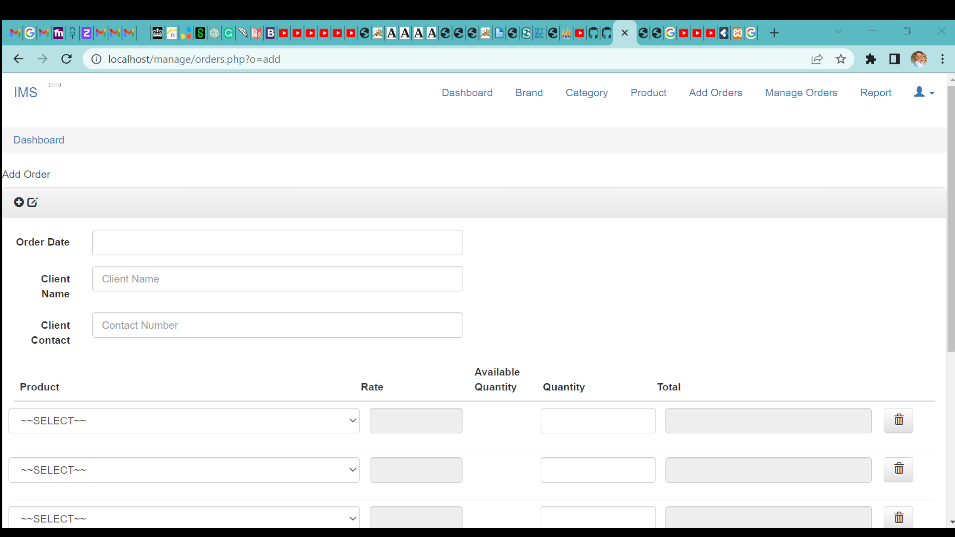
** Figure 4.11 Add Categories page**

****

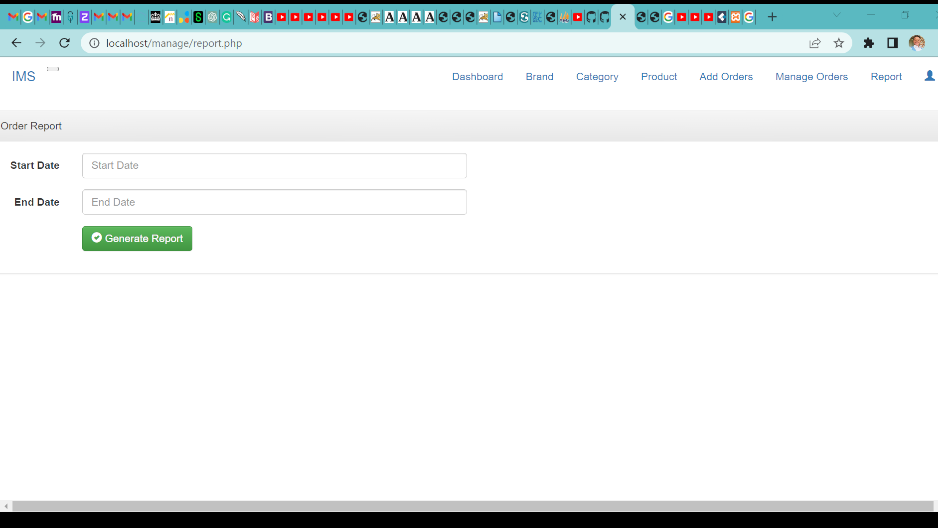
**Figure 4.11 Manage product page**

****

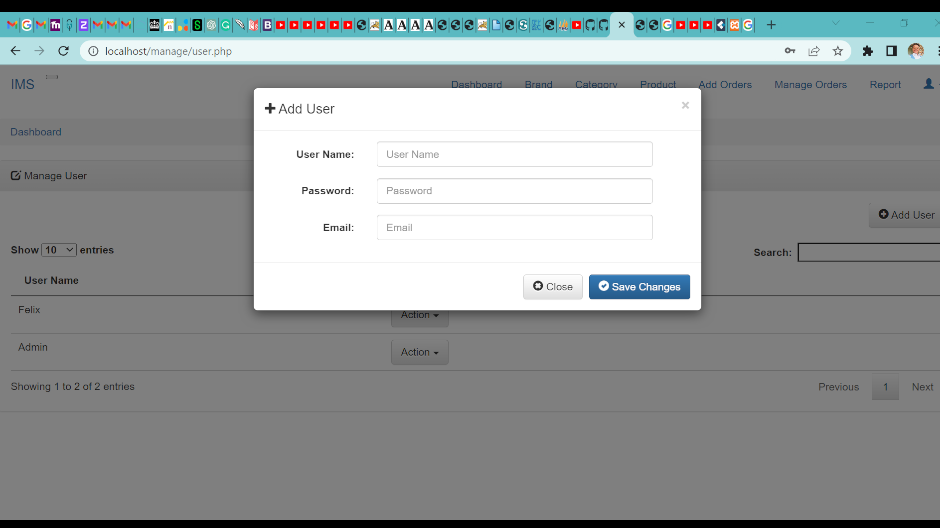
**Figure 4.11 Add Product page**

****

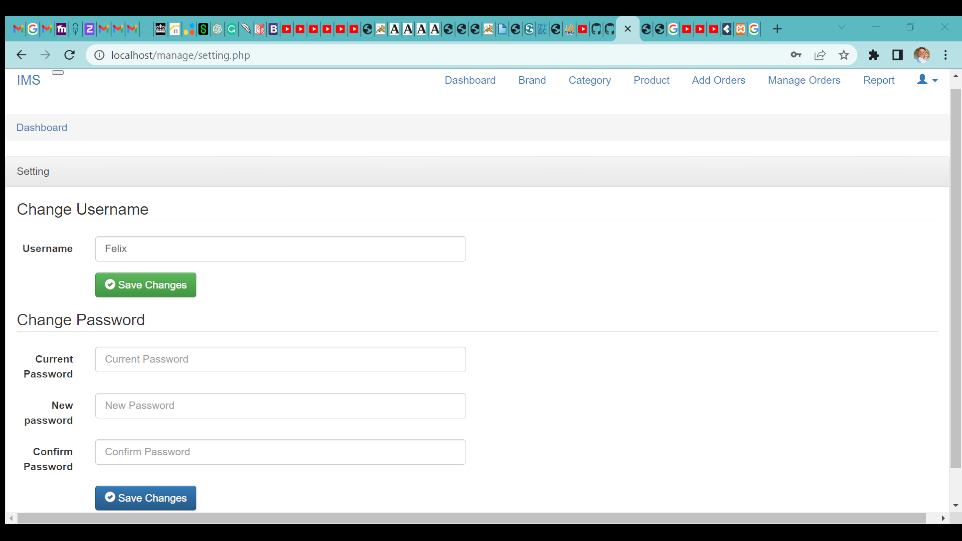
**Figure 4.11 Order Item page**

****

**Figure 4.11 Generate report page**

****

**Figure 4.11 Mange and Add user page**

****

**Figure 4.11 Change password page**

**4.12 Summary**

At this phase, a planned system is executed, and the inventory management system's analysis and design are extensively scrutinised. This phase describes several important system interface components. This plan describes the techniques used to create the system, the resources and equipment used, and the additional resources obtained to put the new system into action.

**CHAPTER 5.0**

**TESTING AND DEBUGGING**

**5.1 Purpose for Testing and Debugging**

Software testing entails determining how effectively the software fits user and system requirements. Testing is also a reliable predictor of the dependability and quality of software. Both software development life cycle stages and the modules for programme code are levels where testing takes place. The system is tested to see whether it has satisfied the objectives established during the early stages of system development. Validation and verification are two methods of software testing.

Debugging, on the other hand, entails identifying, isolating, and fixing software defects, errors, or other issues. Debugging is typically carried out after the following testing when issues are discovered that the test missed. Before making changes to the code or system settings to remedy the problem, debugging sometimes requires reviewing log files, error messages, or other system outputs to identify the issue's primary cause.

**Validation**

Validation frequently comprises testing software to ensure that it works as intended, meets user requirements, and is free of errors and defects. Functional, performance, and security testing may be necessary. It occurs at the conclusion of the life cycle of software development. Software is validated if it meets the standards for which it was created. Validation is important because it helps to ensure the reliability, effectiveness, and safety of the systems and items we use. Errors can be prevented, costs reduced, and overall quality improved.

**Verification**

System verification is the procedure of ensuring that a system or product complies with the demands and needs of the company. This procedure includes testing the system under a variety of settings and scenarios to ensure that it functions as intended and satisfies the quality criteria. The goal of system verification is to guarantee that the system is dependable, efficient, and secure, and it serves the demands of its users.

**5.2 Goals and objectives of testing and debugging**

The primary goal of testing is to ensure that a software system or application operates as expected and satisfies the functional and non-functional requirements, uncover faults, mistakes, or other issues that may develop in the programme, and guarantee that they are corrected before the system is released to production. Testing may help to enhance software quality by detecting problems early in the development process, reducing the likelihood of defects and errors in the final product, and ensuring that the system meets the needs of the user.

On the other hand, debugging seeks to identify, isolate, and resolve software defects, errors, or other issues. Debugging can help to improve the user experience, reduce maintenance costs and downtime, and boost programme performance and dependability.

To summarise, testing and debugging are crucial components of the software development process and play an important role in ensuring software quality and reliability.

**5.3 Type of testing**

There are several forms of testing, depending on the context and goal of the testing. These are only a few examples of testing procedures that may be employed. Unit testing is the specific form of testing used typically to test the functionality of this programme.

* Unit Testing
* Integration testing
* System testing
* Acceptance testing
* Performance testing
* Security testing
* Regression testing
* User acceptance testing.

**5.4 Unit Testing**

This type of testing is performed by the programmer. Unit testing is used to ensure that individual bits of code function correctly, early error detection in the development process, and to ensure that changes to the code do not introduce new issues. Throughout the unit testing process, I completed several testing activities, including reflecting the unit data on the database and its interface, creating reports, and printing invoices.

Moreover, the testing covers logical, syntactic, and functional concerns. This was done to ensure that each software unit was functioning correctly. The testing, however, turned up no significant problems.

**5.5 Test Cases**

A software programme is tested to check if it works correctly under a specific set of conditions or variables known as a test case. It is a specific use case or scenario developed to test a specific component of the product presently in development, such as a feature or capability. The test examples are listed below, along with the results

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test  Scenario | | Description | Test Steps | Test Data | Expected Result | Status |
| 1 | Verify the Login  functionality of the system | Enter a valid username and password | 1 Enter valid username   1. Enter valid password 2. Click on login button | Username: felix  Password:  felix12345 | Successful login | passed |
| 2 | Verify the Login  functionality of the system | Enter an invalid username and a valid password | 1 Enter invalid username  2 Enter valid password  3 Click on login button | Username: felixity  Password:  felix12345 | Error message “User doesn’t exist” | passed |
| 3 | Verify the Login  functionality of the system | Enter a valid username and an invalid password | 1 Enter valid username  2 Enter invalid password  3 Click on login button | Username: felix  Password:  felix123456 | Error message  “Invalid username/password combination” | passed |
| 4 | Verify the Login  functionality of the system | Enter an empty username and a valid password | 1 Enter empty username  2 Enter valid password  3 Click on login button | Username:  Password:  felix12345 | Error message  “Username is required” | passed |
| 5 | Verify the Login  functionality of the system | Enter a valid username and empty password | 1 Enter a valid username  2 Enter empty password  3 Click on login button | Username:  felix  Password: | Error message  “Password is required” | passed |
| 6 | Verify the Login  functionality of the system | Enter an empty username and empty password | 1 Enter an empty username  2 Enter empty password  3 Click on login button | Username:  Password: | Error message  “Username and password is required” | passed |

**Table 5.4 User Login Test**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Scenario | Test Steps | Description | Expected Result | Status |
| Inventory Brand | 1  2  3 | Add Brand  Edit Brand  Removed Brand | User add brand  Brand edited successfully  Brand successfully removed | Passed  Passed  passed |
| Inventory  Categories | 1  2  3 | Add categories  Edit categories  Removed categories | Categories added successfully  Categories edited successfully  Categories removed successfully | Passed  Passed  passed |
| Product | 1  2  3 | Add Product  Edit Product  Removed Product | Product images are added successfully  Product edited successfully  Product removed from the system successfully | Passed  Passed  passed |

**Table 5.4 Inventory Sales Test**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Scenario | Test Steps | Description | Expected Result | Actual result | Status |
| Order | 1 | Add order | User will be able to add product to list of order | Product are successfully added to the other list | passed |
| Add row | 2 | Add row | User is able to add row to order page list | Row added to the order page | passed |
| Reset | 3 | Reset | User is able to reset order page | Order page reset | passed |
| Save changes | 4 | Save changes | User is able to save changes | Changes is successfully save | passed |

**Table 5.4 Product Order Test**

|  |  |  |  |
| --- | --- | --- | --- |
| Test Scenario | Description | Expected Result | Status |
| Report | Client information and purchases | Records can be retrieved for future reference for client purchases. | failed |

**Table 5.4 Users report Test**

|  |  |  |  |
| --- | --- | --- | --- |
| Test scenario | Description | Expected Result | Status |
| Print invoice | Print out invoice for product order | Invoices may be printed as pdf files from the system for future reference. | failed |

**Table 5.4 Print invoice Test**

|  |  |  |  |
| --- | --- | --- | --- |
| Test Scenario | Description | Expected Result | Status |
| Add User | The system will generate a username and password that may be used to login. The user is responsible for adding, modifying, and deleting user information. | Once the user information is correctly filled in on the add user form, it will notify you that the user was successfully added. If not, it will notify user unsuccessfully added | passed |

**Table 5.4 Add User Test**

|  |  |  |  |
| --- | --- | --- | --- |
| Test Scenario | Description | Expected Result | Status |
| Search Item | Search for available item | If the item is discovered during the search, the system will display item found; otherwise, it will display item not found. | Passed |

**Table 5.4 Search Item Test**

|  |  |  |  |
| --- | --- | --- | --- |
| Test scenario | Description | Expected result | Status |
| Logout system | Close the system | If the user clicks the logout button, he or she will be removed from the system; otherwise, the user will remain in the system. | passed |

**Table 5.4 Logout the system**

**5.6 Conclusion**

The system was tested using the concepts of System Analysis and Design. Due to several failed test cases, the system prototype may not be the best. Nonetheless, the system codes can be accessed and updated based on the needs of the user.

**CHAPTER 6.0**

**CONCLUSION**

**6.1 Summary**

In conclusion, the inventory Management System was developed as a simple desktop-based solution to meet all the criteria appropriate for SMEs. The developed software will allow the user to keep track of all inventories. The application will update, add, manage, and delete items according to the requirements. This application also search item from time to time when required to aid the decision-making process and progress of the user. Although this system was developed utilising concepts from system analysis and design, the prototype has some significant benefits and issues, which are listed below.

**6.2 System Success**

* The prototype can provide the user with an interface via which he can recreate the data he desires.
* The prototype developed is user-friendly and secure.
* The prototype can keep track of inventory sales.
* The prototype is capable of managing client data.
* The prototype is capable of creating and managing user accounts.
* The prototype can order, add, edit, and as well as remove item.

**6.3 System Limitation**

Project limitations are features or restrictions that may affect how a project is carried out, finished, or its outcomes. Since this is a prototype, it has certain limitations. Owing to project time restrictions and the lack of sufficient programming knowledge across several disciplines, I was unable to complete some functions in the system compared to the proposed system objectives. According to expectations, these constraints will be considerable. The following are some system limitations:

* Reports are not supported.
* The prototype is not appropriate for large organisations.
* Show a message informing users of the stock inventory that is available.
* The system cannot generate receipt.
* Limited area cover
* Security limitation

**6.4 Lesson Learned**

Spending a long time planning and implementing this prototype is usually an excellent lesson. The information, skills, and concepts learned have enhanced my grasp of software development, time management abilities, effective communication and creativity, and the capacity to work under pressure. In addition, I have learnt about the inventory management process, Visual Studio Code (VS Code) technology involved, and Microsoft database administration.

**6.5 Future Recommendation**

In the development of this inventory management system, it is advised or recommended that this project work be adopted or implemented in small and medium-scale businesses. However, there is always room for improvement. Some of the scopes we can modify for the usability and effectiveness of the prototype are listed below:

* Making the system responsive and adaptable in any scenario.
* It is possible to install an online payment system.
* An alert system for when new and low stocks are added to the system can be installed.
* Adding payment invoices to the client's email address may be encouraged.
* Mobile application.

**REFERENCES**

1. A., A.O., FATOBA, B. and ABISOYE, B.O., 2023. Design of a computerized inventory management system for supermarkets. [online]. Edu.ng:8080. Available from: <http://repository.futminna.edu.ng:8080/xmlui/bitstream/handle/123456789/6048/Design%20of%20a%20Computerized%20Inventory%20Management%20System%20for%20Supermarkets.pdf?sequence=1&isAllowed=y> [Accessed 24 Mar 2023].
2. ALEEM, R.B.A., 2023. SALES AND INVENTORY MANAGEMENT SYSTEM. [online]. Edu.my. Available from: <http://utpedia.utp.edu.my/id/eprint/13591/1/Rahmet%20Bee_13743.pdf> [Accessed 24 Mar 2023].
3. ALEF, Y., 2017. Towards an inventory for archaeological heritage management in Israel. *ISPRS - International Archives of the Photogrammetry Remote Sensing and Spatial Information Sciences*, XLII-2/W5, pp. 21–25. Available from: <https://pdfs.semanticscholar.org/f2ee/0555481e17298b154c1c7d2bd6e0d088dbd3.pdf>.
4. ASGHAR, A.R. et al., 2023. *The impact of analytical assessment of requirements prioritization models: An empirical study*. [online]. Semanticscholar.org. Available from: https://pdfs.semanticscholar.org/76cc/82376a564b7413bdb0bab73fe2e91c0d0630.pdf [Accessed 16 Apr 2023].
5. ATIEH, A.M. et al., 2016. Performance improvement of inventory management system processes by an automated warehouse management system. *Procedia CIRP*, 41, pp. 568–572. Available from: <https://www.sciencedirect.com/science/article/pii/S2212827115012019>.
6. CAMPBELL, S.M., 2010. Punched card systems and the early information explosion, 1880-1945 by Lars Heide. (Baltimore: Johns Hopkins university press, 2009. 369 p. ISBN 978-0-8018-9143-4 65$ US). *Scientia Canadensis*, 33(2), p. 146. Available from: https://www.erudit.org/en/journals/scientia/1900-v1-n1-scientia1821474/1006163ar.pdf.
7. CHILA, M. and SUSI, L.C., 2019. Implementing a web-based inventory tracking system: A quality improvement initiative. *Journal of Radiology Nursing*, 38(4), pp. 277–280. Available from: https://www.sciencedirect.com/science/article/pii/S1546084319301725.
8. GILL, S., KHULLAR, P. and PAL SINGH, N., 2016. A review on various approaches of spare parts inventory management system. Indian Journal of Science and Technology, 9(48). [online]. Available from: <https://sciresol.s3.us-east-2.amazonaws.com/IJST/Articles/2016/Issue-48/Article61.pdf>.
9. HEIDE, L., 1997. Shaping a technology: American punched card systems 1880-1914. *IEEE Annals of the History of Computing*, 19(4), pp. 28–41. Available from: https://d1wqtxts1xzle7.cloudfront.net/85389513/85.62789720220503-1-163ppmo-libre.pdf?1651559774=&response-content-disposition=inline%3B+filename%3DShaping\_a\_technology\_American\_punched\_ca.pdf&Expires=1681524659&Signature=VYjtJr8FvRSqmolgnDnSIQGEY6kWDPvfI-xr~Vk-VXP-cr2YExE3nTkUWOHg2D8yuIYSbxC2wHoqOJhc1hJyrzv04cKvIytEEcUp4fNShZ5qMNZhTHqHrfAqrjMn43uBAP71nH9DuWZLWy8DObuhitvrXr2kNtgcCxPJlgXPIbl1LWzzV81yJkTYly3yl7pGZ7yZXamSTB8QnMLisl2MqI7ff1vJmFLDk373U~cAurisyzLc7OXCfHbFHcalSC6gKTxWi7w38vU7hUy5rYZEbZRue-mT36m9YPFEiJD3hSYYME5SoI0nLTyDZzo6SOAYZYK5v-B11ZLU1BAf-muYBA\_\_&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA [Accessed 15 Apr 2023].
10. JONES, C.M. et al., 2023. NBER WORKING PAPER SERIES. [online]. Nber.org. Available from: <https://www.nber.org/system/files/working_papers/w8494/w8494.pdf> [Accessed 24 Mar 2023].
11. MARTIN, A. and LYNCH, M., 2009. Counting things and people: The practices and politics of counting. *Social Problems*, 56(2), pp. 243–266. Available from: <https://d1wqtxts1xzle7.cloudfront.net/45406677/Martin.2009-libre.pdf?1462542009=&response-content-disposition=inline%3B+filename%3DCounting_Things_and_People_The_Practices.pdf&Expires=1681522611&Signature=LiEp3sCEhFTCwtwIhwnVLr--JMeC3bp-hcz8ZriXOHuKWNB5J24rA8b38zUjZ-6XUnTW8UwU2cmeCPZj6hBjrRBECMz1z5LkBY0nJXqdhXnU6Nf3TnqaOnggEMG~4Q1NIc0KlwGpCNuWOXNLhXFSZQOdVQYhFdSaLn3x2lZnWrz4LIvSZfKaP5Mgn6cI8tcoO4hmP8LVvLOgyO47crmzQRSrcZtXzIMKTs58FKowDKnlx9qWapEK-LrwiNMr2a~xCcPtSWqCFHUhCCtdWv8WvbVVPi22BfL3EnBBztSPSxzr16ulcfbLmjRveAlL7o-hghTRr4QH2EqvaKeHVuaLfg__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA>.
12. MELANIE, 2017. *Factors affecting Inventory Management – Inventory Control 101*. [online]. Unleashed Software. Available from: https://www.unleashedsoftware.com/blog/factors-affecting-inventory-management [Accessed 15 Apr 2023].
13. MOGOROSI, T., 2023. An analysis of the inventory management system and controls at glencore: A case of rustenburg smelter. [online]. [Nwu.ac.za](http://nwu.ac.za/). Available from: <https://dspace.nwu.ac.za/bitstream/handle/10394/25356/Mogorosi_TM.pdf?sequence=1&isAllowed=y> [Accessed 23 Mar 2023].
14. MUYUMBA, T. and PHIRI, J., 2017. A web based inventory control system using cloud architecture and barcode technology for Zambia air force. *International Journal of Advanced Computer Science and Applications : IJACSA*, 8(11). [online]. Available from: https://pdfs.semanticscholar.org/5f66/5fd000b32def9190a47263c7209899b00e37.pdf [Accessed 15 Apr 2023].
15. OLADELE, T.O. et al., 2021. Development of an inventory management system using association rule. Indonesian Journal of Electrical Engineering and Computer Science, 21(3), p. 1868. Available from: <https://d1wqtxts1xzle7.cloudfront.net/82844958/22115_46460_1_PB-libre.pdf?1648518304=&response-content-disposition=inline%3B+filename%3DDevelopment_of_an_inventory_management_s.pdf&Expires=1679663519&Signature=exFshKl7w9I6AAclo7-V2uCaw92YW9qQYVcPuAXSxKWa8vtBLnoViw4Tzu15cnF4gn03F1hA~hwXwhybPd68nX2s57ck~bAwkUD8NALTe~IF4GtzKE5x9pijmUOmug1xYPgczd48kdAuQzF6JquxSETmtisYWaFoA3NPw1KecmTnX6nanSieHg80F6w9AAZiBE820x-xHGWyRkqLRrA4oZBggDkpJJ8o0W0Jd14YSjG7yVAZF817N40QGTdTjHnwPRWskof9iCiDy98oE54KIPUv2TpJ0gZi1UIEA4AAMQS3GBoE9YfZTkF7XNMtQVS5yUbt9-EiVxEmR8LqjVZP6Q__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA>.
16. RADIO FREQUENCY IDENTIFICATION and READER, T., 2023. *Radio Frequency Identification (RFID)*. [online]. Otago.ac.nz. Available from: https://ourarchive.otago.ac.nz/bitstream/handle/10523/1247/RFID\_Pre\_Publication.pdf?sequence=5 [Accessed 15 Apr 2023].
17. DE SILVA, N.P., 2023. SALES AND INVENTORY MANAGEMENT SYSTEM for IMPERIAL AUTO CARE. [online]. [Cmb.ac.lk](http://cmb.ac.lk/). Available from: <https://dl.ucsc.cmb.ac.lk/jspui/bitstream/123456789/3985/1/0808784.pdf> [Accessed 24 Mar 2023].
18. SUNOL, H., 2023. How warehouse Inventory management can boost productivity. [online]. Cyzerg.com. Available from: <https://articles.cyzerg.com/warehouse-inventory-management-techniques-to-increase-productivity> [Accessed 24 Mar 2023].
19. TEJESH, B.S.S. and NEERAJA, S., 2018. Warehouse inventory management system using IoT and open source framework. *Alexandria Engineering Journal*, 57(4), pp. 3817–3823. Available from: https://www.sciencedirect.com/science/article/pii/S1110016818301765.