**WAREHOUSING**

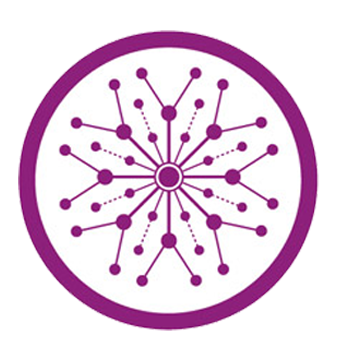
(AI based inventory Management system)

Final Year Project

**Session 2019-2023**

A project submitted in partial fulfillment of the degree of

BS in Software Engineering



Department of Software Engineering

Faculty of Computer Science & Information Technology

The Superior University, Lahore

Spring 2023

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Type (Nature of project) | | | [ ✓ ] **D**evelopment [ ] **R**esearch [ ] **R**&**D** | | |
| Area of specialization | | |  | | |
| FYP ID | | | **BSSE-FYP-S22-012** | | |
| **Project Group Members** | | | | | |
| Sr.# | Reg. # | Student Name | | Email ID | \*Signature |
| (i) | BSEM-S19-007 | Samiullah | | [bsem-s19-007@superior.edu.pk](mailto:bsem-s19-007@superior.edu.pk) | Sami |
| (ii) | BSEM-S19-014 | Ali Raza | | [bsem-s19-014@superior.edu.pk](mailto:bsem-s19-014@superior.edu.pk) | Ali |
| (iii) |  |  | |  |  |

\*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to work of others

# Plagiarism Free Certificate

This is to certify that, I **Samiullah** S/D of **Saif-ur-Rehman**, group leader of FYP under registration no **BSEM-S19-007** at Software Engineering Department, The Superior College, Lahore. I declare that my FYP report is checked by my supervisor.

Date: 20 Jun, 2022 Name of Group Leader: Samiullah Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_

Name of Supervisor: Mr. Muhammad Ahmed Co-Supervisor: Mr. Aqib

Designation: Senior Lecturer Designation: Senior Lecturer

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

HoD: Dr. Tehreem Masood

Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Warehousing**

**Change Record**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Author(s)** | **Version** | **Date** | **Notes** | **Supervisor’s Signature** |
|  | 1.0 |  | <Original Draft> |  |
|  |  |  | <Changes Based on Feedback from Supervisor> |  |
|  |  |  | <Changes Based on Feedback From Faculty> |  |
|  |  |  | <Added Project Plan> |  |
|  |  |  | <Changes Based on Feedback from Supervisor> |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**APPROVAL**

|  |  |
| --- | --- |
| **Project Supervisor** | |
| Comments: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | |
| Name:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | |
| Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |

|  |  |
| --- | --- |
| **Project Manager** | |
| Comments: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
|  |  |
| Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

|  |  |
| --- | --- |
| **Head of the Department** | |
| Comments: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
|  |  |
| Date:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

# Dedication

This study is whole heartedly dedicated to our beloved parents and teachers, who have been our source of inspiration and gave us strength when we think of giving up, who continually provide their moral, spiritual, emotional, and financial reports. I dedicate this project to God Almighty, my creator, my strong pillar, my source of inspiration, wisdom, knowledge, and understanding. He has the source of my strength throughout this program and only have I soared in his wings. I also dedicate this work to HoD Dr. Tehreem Masood my respected Teacher Mr. Muhammad Ahmed and Mr. Aqib and all the professors of superior who has encouraged me all the way and whose encouragement have made sure that I give it all it takes to finish that which I have started. I also dedicate this to my fellows who supported me throughout the process.

# Acknowledgements

I would like to thank Mr. Muhammad Ahmed And Mr. Aqib for their expert advices and encouragement on this difficult project as well as Mr. Muhammad Ahmed and all superior professors and teachers for his brilliance in the class and he taught me how to write a report on a relevant topic. This enhanced my skills in writing the report and further research papers and articles etc. I would like to thank those people who wrote reports on this topic in the past and I got many concepts and ways of writing the report from those reports and I am very thankful to Google too. This project is impossible without networks and search engines.

# Executive Summary

Artificial intelligence is intelligence demonstrated by machines that can leveraged to keep track of transactions over the internet. The technology provides transparency and traceability that can used in the management of inventory. When it comes to the formalization of inventory, AI technology promises to authenticate owners and other users of warehouse, and provides a fixed ledger of transactions. At present, AI technology explored as concept in several countries to manage inventory. We extend the idea to the granting of inventory Artificial intelligence is intelligence demonstrated by machines. This technology can offer an effective means to manage inventory a warehouse, provide digital documentation to users in the market and reduce inefficiency in inventory systems. However, the uptake of the technology in inventory administration limited by human related factors. These limitations include, but are not limited to, the accuracy of data entered into the system, the ability of the system to facilitate data preservation, pre-existing institutional and the digital divide across warehouses. Part of overcoming these barriers requires the warehouses owners and governments to invest in digital technologies and develop institutional capacities to overcome current limitations to bring inventory management into the industry.

# Table of Contents

[Dedication v](#_Toc125468174)

[Acknowledgements vi](#_Toc125468175)

[Executive Summary vii](#_Toc125468176)

[Table of Contents viii](#_Toc125468177)

[List of Tables x](#_Toc125468178)

[List of Figures xi](#_Toc125468179)

[Chapter 1 1](#_Toc125468188)

[Introduction 1](#_Toc125468189)

[1.1. Background 2](#_Toc125468190)

[1.2. Motivations and Challenges 2](#_Toc125468191)

[1.3. Goals and Objectives 3](#_Toc125468192)

[1.4. Literature Review/Existing Solutions 3](#_Toc125468193)

[1.5. Gap Analysis 3](#_Toc125468194)

[1.6. Proposed Solution 3](#_Toc125468195)

[1.7. Project Plan 4](#_Toc125468196)

[1.7.1. Work Breakdown Structure 4](#_Toc125468197)

[1.7.2. Roles & Responsibility Matrix 6](#_Toc125468198)

[1.7.3. Gantt Chart 7](#_Toc125468199)

[1.8. Report Outline 9](#_Toc125468200)

[1.9. Empathy Map 10](#_Toc125468201)

[Chapter 2 11](#_Toc125468202)

[Software Requirement Specifications 11](#_Toc125468203)

[2.1. Introduction 12](#_Toc125468205)

[2.1.1. Purpose 12](#_Toc125468206)

[2.1.2. Document Conventions 12](#_Toc125468207)

[2.1.3. Intended Audience and Reading Suggestions 12](#_Toc125468222)

[2.1.4. Product Scope 13](#_Toc125468223)

[2.1.5. References 13](#_Toc125468224)

[2.2. Overall Description 14](#_Toc125468225)

[2.2.1. Product Perspective 14](#_Toc125468226)

[2.2.1.1 Inventory Management 14](#_Toc125468227)

[2.2.1.2 Inventory Tracking 14](#_Toc125468228)

[2.2.1.3 Reporting 15](#_Toc125468229)

[2.2.1.4 Inventory Forecasting 15](#_Toc125468231)

[2.2.1.5 Inventory Alert 15](#_Toc125468233)

[2.2.1.6 Inventory Security and Backups 16](#_Toc125468234)

[2.2.2. Operating Environment 16](#_Toc125468235)

[2.2.3. Design and Implementation Constraints 16](#_Toc125468236)

[2.2.4. Assumptions and Dependencies 16](#_Toc125468237)

[2.3. External Interface Requirements 17](#_Toc125468238)

[2.3.1. User Interfaces 17](#_Toc125468239)

[2.3.2. Hardware Interfaces 17](#_Toc125468240)

[2.3.3. Software Interfaces 17](#_Toc125468241)

[2.3.4. Communications Interfaces 18](#_Toc125468242)

[2.4. System Features 18](#_Toc125468244)

[2.5. Nonfunctional Requirements 19](#_Toc125468245)

[2.5.1. Performance Requirements 19](#_Toc125468246)

[2.5.2. Safety Requirements 19](#_Toc125468247)

[2.5.3. Security Requirements 19](#_Toc125468248)

[2.5.4. Software Quality Attributes 20](#_Toc125468249)

[2.5.5. Business Rules 20](#_Toc125468250)

[2.6. Other Requirements 21](#_Toc125468251)

[Chapter 3 22](#_Toc125468252)

[Use Case Analysis 22](#_Toc125468253)

[Use Cases Description 23](#_Toc125468254)

[Chapter 4 24](#_Toc125468255)

[System Design 24](#_Toc125468256)

[4.1. Domain Model 25](#_Toc125468257)

[4.2. Entity Relationship Diagram with data dictionary 26](#_Toc125468258)

[4.3. Class Diagram 27](#_Toc125468259)

[4.4. Activity Diagram 28](#_Toc125468260)

[4.5. Sequence / Collaboration Diagram 29](#_Toc125468261)

[4.6. Deployment Diagram 30](#_Toc125468262)

[Chapter 5 31](#_Toc125468263)

[Implementation 31](#_Toc125468264)

[5.2. Important Flow Control/Pseudo codes 32](#_Toc125468265)

[5.3. Components, Libraries, Web Services and stubs 32](#_Toc125468266)

[5.4. Deployment Environment 33](#_Toc125468267)

[5.5. Tools and Techniques 33](#_Toc125468268)

[5.6. Best Practices / Coding Standards 33](#_Toc125468269)

[5.7. Version Control 33](#_Toc125468270)

[Chapter 6 34](#_Toc125468271)

[Testing and evaluations 34](#_Toc125468272)

[Testing Evaluation 35](#_Toc125468273)

[6.1 Testing Use Case 35](#_Toc125468274)

[6.2 Performance Testing 38](#_Toc125468275)

[6.3 Unit Testing 39](#_Toc125468276)

[6.4 Equivalence Partitioning 39](#_Toc125468277)

[6.5 Boundary Value Analysis 39](#_Toc125468279)

[6.6 Stress Testing 39](#_Toc125468280)

[6.7 Data Flow Testing 40](#_Toc125468281)

[Chapter 7 41](#_Toc125468283)

[Summary, Conclusion & Future Enhancements 41](#_Toc125468284)

[7.1 Project Summary 42](#_Toc125468285)

[7.2 Achievements and Improvements 43](#_Toc125468286)

[7.3 Critical Review 44](#_Toc125468287)

[7.4 Lessons Learnt 44](#_Toc125468288)

[7.5 Future Enhancements/Recommendations 45](#_Toc125468292)

[Reference and Bibliography 46](#_Toc125468293)

# List of Tables

[Table 1 # 1.7.2: Roles & Responsibility Matrix 6](#_Toc125468128)

[Table 2 # 1.7.3: Gant chart 7](#_Toc125468129)

[Table 3 # 2.1.2: Document Convention 12](#_Toc125468130)

[Table 4 # 2.3.3: Software interface 17](#_Toc125468131)

[Table 5 # 5.3: Components, Libraries, Web Services and stubs 32](#_Toc125468132)

[Table 6 # 6.1: Test Case 1: Login 35](#_Toc125468133)

[Table 7 # 6.2: Test Case 2: Dashboard 35](#_Toc125468134)

[Table 8 # 6.3: Test Case 3: Check Records 36](#_Toc125468135)

[Table 9 # 6.4: Test Case 4: Update Data 36](#_Toc125468136)

[Table 10 # 6.5: Test Case 5: Check Warehouse 37](#_Toc125468137)

[Table 11 # 6.6: Test Case 6: Logout 37](#_Toc125468138)

[Table 12 # 6.7: Test Case 7: Registration 38](#_Toc125468139)

[Table 13 # 7.1: Project Summary 42](#_Toc125468140)

# List of Figure

[Figure 1 # 1.7: Project Plan 4](#_Toc125468628)

[Figure 2 # 1.8: Report Outline 9](#_Toc125468629)

[Figure 3 # 1.9: Empathy Map 10](file:///D:\BSSE\7th%20Semester\FYP\Warehousing%20Project%20Report%20(FYP).docx#_Toc125468630)

[Figure 4 # 2.3.1: User Interface 17](#_Toc125468631)

[Figure 5 # 3.1: Use Case Model 23](#_Toc125468632)

[Figure 6 # 4.1.: Domain Model 25](#_Toc125468633)

[Figure 7 # 4.2.: ERD 26](#_Toc125468634)

[Figure 8 # 4.3.: Class Diagram 27](#_Toc125468635)

[Figure 9 # 4.4: Activity Diagram 28](#_Toc125468636)

[Figure 10 # 4.5: Sequence / Collaboration Diagram 29](#_Toc125468637)

[Figure 11 # 4.6: Deployment Diagram 30](#_Toc125468638)

# Chapter 1

# Introduction

**Chapter 1:** Introduction

The warehouse management process is known to be a multi-step process because it requires the involvement of all stakeholders directly or indirectly involved in warehouse management. The warehouse / inventory storage systems in use today pose major problems in terms of data fraud, sensitive data security, and the risk of system failure due to natural disasters. If the server used for data storage fails. Compared to current approaches and practices for warehouse management and data storage, AI is a state-of-the-art technology and database that can fully address the problems that plague current systems. The basic and most important aspect of AI technology is intelligence demonstrated by machines, as opposed to the natural intelligence displayed by humans.

## Background

The main problem, which is associated with the inventory management, is the usage of the printed document. In this context therefore, it is easy to see the relevance of a AI based inventory management system to developed and more so to developing economies. Paper documents can easily replicated, tampered or even destroyed.

## Motivations and Challenges

Some countries are already turning to AI for inventory management, including Sweden and Ukraine. Georgia is a great example for that where many warehouses have already on AI. The sale process now takes minutes, rather than days, with operational cost reductions of up to 90 percent. With these efficiencies, it is surely only a matter of time before other countries start to leverage the potential of AI for inventory.

Centralized servers for inventory data face many problems. Data loss due to natural disasters biggest and most realistic problem is that the inventory manger that handles the data can also tamper with the data, our motivation and goal is to find a solution to the shortcomings of warehousing / inventory management systems using Artificial intelligence is intelligence demonstrated by machines.

## Goals and Objectives

* **Less Storage**
* **Improve Productivity**
* **Increase Profits**
* Inventory Adjustment

## Literature Review/Existing Solutions

* We gathered requirements of Pakistani peoples / warehouses owner and provide solution AI based Inventory management system name Warehousing.
* There is a need of this kind of platform for Warehouses.
* We also need to spread awareness through social platforms. This AI based Warehousing web app help the people to manage their warehouses.

## Gap Analysis

Gap analysis can be applied to many aspects such as marketing, process, productivity, supply control, etc., but here we will focus on inventory. Warehouse management gap analysis may include finding areas with a high risk of error or slow pick times. Once these opportunities are identified, you can set viable goals to maximize your business potential.

## Proposed Solution

AI provides a potential solution for many of the challenges of inventory management. This use case for AI extends beyond a pure database, leveraging the opportunity to create a permanent, unbreakable record of inventory data. The simplest implementation of a AI based inventory system could enable the ownership to recorded and assigned to the owner’s user account. If there are some changes in the data, these can added to the AI data base, and if the goods sold, all the relevant data of inventory can be update. Every transaction is traceable, timestamped, and indisputable. Used in this way, AI could provide a highly secure record of inventory that cannot manipulated. For example, if the owner wants to take visit on the warehouse inventory, it can grant access rights. This can done whenever needed, unconstrained by office working hours. No paper-based system can provide such flexibility, resilience and durability.

## Project Plan

Project plan includes Work Breakdown Structure (WBS) and Gantt Chart. WBS describes complete project decomposition by breaking down its structure. And Gantt chart will describe the timeline and time strategies about the whole project deliverables.



Figure 1 # 1.7: Project Plan

## Work Breakdown Structure

1. **Requirements**
   1. Requirement Gathering
      1. Literature Review
      2. Informal Requirements
   2. Analysis

1.2.1 Analysis Process

1.2.2 Formal Documentation

* 1. Software Requirement Specifications (SRS)

1. **Design**
   1. Software Architecture
      1. UML Diagram
      2. Class Diagram
      3. Deployment Diagram
      4. Activity Diagram
      5. Use case Diagram
      6. ER-Diagram
      7. Sequence Diagram
      8. Package Diagram
   2. Network Diagram

2.2.1 System Architecture (Service Oriented)

* 1. User Interface View

2.3.1 Web Application Interface.

1. **Implementation**
   1. Web Front End
      1. HTML
      2. CSS
   2. Web Back End
      1. Laravel
   3. Database Management System
      1. My-SQL
      2. SQL Server
2. **Testing**
   1. Module testing
   2. System testing
3. **Deployment**
   1. Web Front End
      1. React
      2. HTML
      3. CSS
   2. Web Back End
      1. Node
   3. Database Management System
      1. My-SQL
      2. SQL Server

## Roles & Responsibility Matrix

Table 1 # 1.7.2: Roles & Responsibility Matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **WBS #** | **WBS Deliverable** | **Activity #** | **Activity to Complete the Deliverable** | **Duration**  **(# of Days)** |
| 1 | Literature Review | 1.1.1 | - | 2 |
| 2 | Informal Requirements | 1.1.2 | 1.1.1 | 3 |
| 3 | Analysis Process | 1.2.1 | 1.1.2 | 2  (1st Week) |
| 4 | Formal Documentation | 1.2.2 | 1.2.1 | 3 |
| 5 | SRS | 1.3 | 1.2.2 | 4  (2nd Week) |
| 6 | UML Diagram | 2.1.1 | 1.3 | 3 |
| 7 | Class Diagram | 2.1.2 | 1.3 | 2 |
| 8 | Deployment Diagram | 2.1.3 | 1.3 | 3  (3rd Week) |
| 9 | Activity Diagram | 2.1.4 | 1.3 | 3 |
| 10 | Use case Diagram | 2.1.5 | 1.3 | 3  (4th Week) |
| 11 | ER-Diagram | 2.1.6 | 1.3 | 3 |
| 12 | Sequence Diagram | 2.1.7 | 1.3 | 3 |
| 13 | Package Diagram | 2.1.8 | 1.3 | 2  (5th Week) |
| 14 | System Architecture | 2.3.1 | 1.3 | 3 |
| 15 | Web UI Interface | 2.2.1 | 1.3 | 3  (6th Week) |
| 17 | Web App front end | 3.1 | 1.3 , 2.1 , 2.2 , 2.3.1 | 14  (8th Week) |
| 18 | Web App back end | 3.2 | 1.3 , 2.1 , 2.2 , 2.3.1 | 28  (20th Week) |
| 21 | Database Management | 3.5 | 1.3 , 2.1.6 | 21  (23rd Week) |
| 22 | Module testing | 4.1 | 3.1 , 3.2 , 3.3 , 3.4 | 3 |
| 23 | System testing | 4.2 | 4.1 | 7  (25th Week) |
| 24 | Live web on server | 5.1 | 4.2 | 2  (26th Week) |

## Gantt Chart

Table 2 # 1.7.3: Gant chart

|  |  |  |
| --- | --- | --- |
| **Milestone 1** | | |
| **Name** | **Start Date** | **End Date** |
| **Requirement Gathering** | **01/02/22** | **05/02/22** |
| 1. Literature Review | 01/02 | 02/02 |
| 1. Informal Requirements | 03/02 | 05/02 |

|  |  |  |
| --- | --- | --- |
| **Milestone 2** | | |
| **Name** | **Start Date** | **End Date** |
| **Analysis** | **08/02/22** | **018/02/22** |
| 1. Analysis Process | 08/02 | 09/02 |
| 1. Formal Documentation | 12/02 | 12/02 |
| 1. SRS | 15/02 | 18/02 |

|  |  |  |
| --- | --- | --- |
| **Milestone 3** | | |
| **Name** | **Start Date** | **End Date** |
| **Software Architecture** | **19/02/22** | **20/03/22** |
| 1. UML | 19/02 | 23/02 |
| 1. Class Diagram | 24/02 | 26/02 |
| 1. Deployment Diagram | 26/02 | 29/02 |
| 1. Activity Diagram | 1/03 | 05/03 |
| 1. Use Cases | 06/03 | 08/03 |
| 1. ER Diagram | 09/03 | 13/03 |
| 1. Sequence Diagram | 14/03 | 16/03 |
| 1. Package Diagram | 19/03 | 20/03 |

|  |  |  |
| --- | --- | --- |
| **Milestone 4** | | |
| **Name** | **Start Date** | **End Date** |
| **System Architecture** | **21/03/22** | **23/03/22** |
| 1. Network Diagram | 21/03 | 23/03 |

|  |  |  |
| --- | --- | --- |
| **Milestone 5** | | |
| **Name** | **Start Date** | **End Date** |
| **User Interface** | **26/03/22** | **05/04/22** |
| 1. Web UI | 26/03 | 05/04 |

|  |  |  |
| --- | --- | --- |
| **Milestone 6** | | |
| **Name** | **Start Date** | **End Date** |
| **Implementation** | **03/05/22** | **14/09/22** |
| 1. Web Frontend | 03/05 | 30/06 |
| 1. Web backend | 01/07 | 14/09 |

|  |  |  |
| --- | --- | --- |
| **Milestone 7** | | |
| **Name** | **Start Date** | **End Date** |
| **Database Management System** | **15/09/22** | **12/10/22** |
| 1. MySQL/SQL Server | 15/09 | 12/10 |

|  |  |  |
| --- | --- | --- |
| **Milestone 8** | | |
| **Name** | **Start Date** | **End Date** |
| **Testing** | **13/10/22** | **26/10/22** |
| 1. Module Testing | 13/10 | 17/10 |
| 1. System Testing | 18/10 | 26/10 |

|  |  |  |
| --- | --- | --- |
| **Milestone 9** | | |
| **Name** | **Start Date** | **End Date** |
| **Deployment** | **27/10/22** | **31/10/22** |
| 1. Live web in server | 27/10 | 28/10 |
| 1. Launch Android app on server | 29/10 | 31/10 |

## Report Outline



Figure 2 # 1.8: Report Outline

## Empathy Map

Figure 3 # 1.9: Empathy Map















# Chapter 2

# Software Requirement Specifications

**Chapter 2:** Software Requirement Specifications



## Introduction

## Purpose

AI provides a potential solution for many of the challenges of inventory management. This use case for block-chain extends beyond a pure database, leveraging the opportunity to create a permanent, unbreakable record of inventory data. The simplest implementation of a AI based inventory system could enable the ownership to recorded and assigned to the owner’s user account. If there are some changes in the data, these can added to the AI data base, and if the goods sold, all the relevant data of inventory can be update.

## Document Conventions

Table 3 # 2.1.2: Document Convention

|  |  |
| --- | --- |
| Name | Meaning |
| DDB | Distributed Database |
| ER | Entity Relationship |
| DB | Database |
| API | Application Programming Interface, which is a software intermediary that allows two applications to talk to each other. |
| Web Services | A web service is a piece of code that is available on through internet from a specific server and work as a standardized XML messaging system. The XML is used to encode all communication to web services. |
| Artificial intelligence | Artificial intelligence is intelligence demonstrated by machines that can be leveraged to keep track of transactions over the internet. The technology provides transparency and traceability that can be used in the management of land rights. |

## Intended Audience and Reading Suggestions

The project has implemented under the guidance of university and professors and Entrepreneurship professors. This project is useful for the people who are facing several issues in their property. This document is also readable for Investors this Document and project are not only for FYP it is also readable for Several Companies and Warehouses. This document also helps out the developer and project manager to implement the future requirements and future upgradations.

## Product Scope

Inventory management helps to manage the stock of the company. It provides proper details of the products what kind of raw material, what are the sizes we require etc. to the purchasing department. When the inventory management provides proper information to management, they buy according to them, which helps the company to store fewer products. Inventory management helps to improve the productivity of the machines and man’s power. Employees are aware of stocks and the quantity that require producing. Inventory management helps to improve the profits of the company. It helps to provide proper information about stocks that saves the unnecessary expenses on stocks.

## References

* **System Requirement Specification**

This project is based on a web application that can connect the libraries and some external services the other feature like Requirement, Safety, and Security Requirements, and the quality attributes of this project are described below.

* **Source**

IEEE Format for Software Requirement Specification

* **Document help Reference**

IEEE Version 3.0

## Overall Description

## Product Perspective

An inventory web app can have distinctive features, depending on the businesses it is serving to. You can integrate end number of elements as per your industry and use. However, there are some standard features which every inventory management app should have in place.

## Inventory Management

The module or feature is all meant to keep your essential warehouse functions centralized. This helps in tracking every single inventory details like stock level, product history, and other product specifications. More importantly, the data syncs with all other modules in the inventory system. This helps in accurately operating inventory which enhances productivity by promoting a sense of collaboration among team members working from different locations on their systems.

* Categorization of products in stock
* Measurements of products
* Products history
* Cycle Counting
* Live Stock reports
* Vendor Management

## Inventory Tracking

Warehouse requires a workforce to keep a tab at every step. Barcode, which we have understood earlier in this article, plays a significant role in letting you track your inventory accurately. Assets serial number, RFID, and other things important here as these are the things which let you know real-time data of everything inventory.

* Product tracking
* Tagging
* Reports
* Invertor tracking solutions

## Reporting

## For any inventory business to be efficient, you need to be updated with real-time data regarding product status, driver’s whereabouts, order status, shipment and much more. An efficient Inventory Management app comes with the capabilities of integrating various reporting tools and features that makes the reporting process easier. This is a crucial feature for any hardcore inventory business.

## Inventory Forecasting

## You would never want to disappoint your customers with their favorite products out of the stocks, would you? Inventory forecast, as the name suggests, lets you find out which products are going to out of stock soon and what’s in abundance. This is a great trick to have up your sleeve for serving an excellent user experience to your customers. The biggest advantage of forecasting is it gives you the control and wisdom of spending your resources wisely. You should be more alert about sales attributes such as size, color, material, scent and other features to be better prepared. This gives you a better judgement of purchase quantity and a better understanding of what to purchase when.

* Powerful future insights
* Reduced risk stock out
* Lower holding costs
* Increase turnover rates

## Inventory Alert

Gone are the days when you had to waste your valuable business time and money on manually monitoring the critical stock data. With the right inventory alert feature, you will be able to find out the possible issues which might occur in the future directly via email or SMS alert. What goes out; what comes in; status of a shipment, issues alerts, etc. important aspects of running a business accurately will be easily managed by getting a reliable inventory alerts.

* Reduced stock waste
* Optimized inventory levels
* Drive sales

## Inventory Security and Backups

No matter what business you are in, having the right security for your inventory is essential. If you have Inventory Management Apps, the proper protection should be put in place. Right from protecting your assets to the data it accumulates should be safe and adequately backed up. Generally, the inventory should be securely optimized, to protect our data from hacking.

## Operating Environment

* Web Services database
* Operating system
* Windows
* Database
* Firebase database
* Platform
* AI System

## Design and Implementation Constraints

* There is also a limitation of online network is available to devices so that is why the application runs properly.
* The application also has the limitation of Constraints are design limitations. An obvious example is a budget. Money affords labor, tools, and to some extent, time.

## Assumptions and Dependencies

* A Request for the same identification is not given by the system No Same User exist. It generates System Error so the system should avoid this.
* A Request for the same identification is not given by the system no same operator id exists. It generates System Error so the system should avoid this.

## External Interface Requirements

## User Interfaces

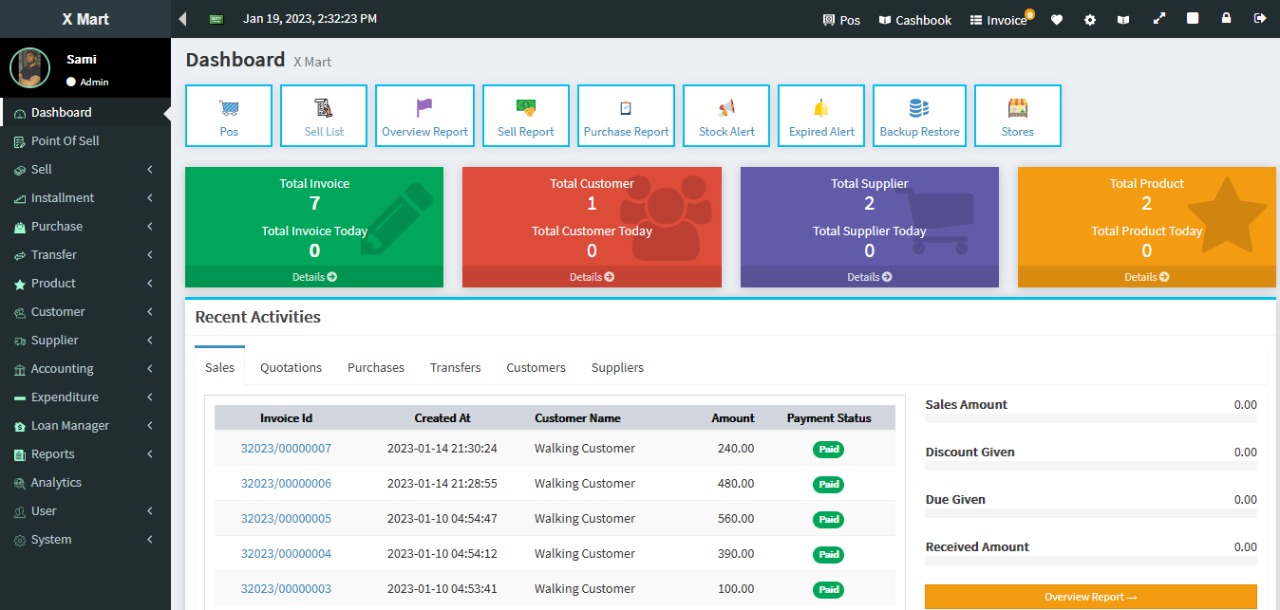


Figure 4 # 2.3.1: User Interface

## Hardware Interfaces

As this system is online so it only require a web browser and can run with any system specifications.

## Software Interfaces

Table 4 # 2.3.3: Software interface

|  |  |
| --- | --- |
| **Software used** | **Description** |
| **Operating system** | We have chosen Windows for this platform |
| **Database** | To save the data of warehouse inventory. |
| **XML** | Used to create front end sale page, inventory, Homepage, Record page And other pages. |
| **Activities** | Activity shows the front more effective and speedily working. |
| **Fragment** | The fragment is used to work and transact between activities. |
| **ReactJS** | To implement the project, we use react with JavaScript |
| **Firebase Database** | Firebase database used to store images and inventory data |
| **Authentication** | Authentication Used for Google, Email, etc. |
| **Fire Storage** | Fire storage used storing images |
| **Remote Config** | Remote Config is used on an admin side |
| **Notification Manager** | To notify the warehouse manger |
| **Firebase Real-time** | Firebase real-time used to store Uri and string, int, float data |
| **Web Services** | Web services used in it to add statistics in an application and connect with retailers. |

## Communications Interfaces

## Communication Standards

HTTP

FTP

* Google map uses HTTP protocol and secure protocol.
* API uses HTTP protocol to connect the user.
* Firebase Server communication needs an internet connection on any browser.
* Frontend:
* Javascript
* React Framework
* CSS
* Metamask Chrome Extension
* Backend:
* Artificial intelligence

## System Features

This system provides multiple features such as:

* Inventory management
* Uploading products pictures
* Buyer Request
* Report
* Quality check of product

## Nonfunctional Requirements

## Performance Requirements

* The application should load and be usable within 3 seconds.
* The database should be normalized to prevent redundant data and improve performance.
* The retrieving power of the application should be good.
* The database should be distributed to prevent outages.
* The data saving efficiently in the database.
* When moving from one to second activity system performance not disturb.

## Safety Requirements

* Databases should use sharing to be redundant to prevent loss of data.
* Backups of the databases should be done hourly and be kept for one week.
* Data should be store in the arrangement of AI.
* Images stored in the database also in the backend AI.
* System shutdown in the case of a cyber-attack.
* In any draw, the back system should tell
* Authentication used when the email enter by the user is safe.
* Authentication used by Google is safe

## Security Requirements

In order to evaluate the security requirement of our application. We evaluate our application with the security point of view. The following are the different security aspects. Security has three aspects Confidentiality, Integrity and Availability

* **Confidentiality:**

Confidentiality means that the message between the two users through a WEB Socket will be readable to the other user. All the chat between the users will be encrypted stored into the database. A message is encrypted by AI technology and can be easily readable to another user.

* **Integrity:**

This ensure that the messages chat between the two users will not be edited or change during the transfer of it. We will use hashing function which will map an encrypted message to fixed size length integer. A has function is a one-way function if somebody has an output of has it cannot be reversed.

* **Availability:**

100% availability 24/7 because our web server is running 24/7

## Software Quality Attributes

* Because this application is critical to business communication, we will have a goal of four nines (99.99%) availability.
* The interface should be easy to learn without a tutorial and allow users to accomplish their goals without errors.
* The interface can use by anyone.

## Business Rules

* Application is efficient in use and user should be satisfied.
* Working on data is efficient.
* Working on web services data is sufficient.
* Working on application loading is efficient.
* The application should never allow anyone to read messages or discussions not intended for that person.
* Emails should be sent with a latency of no greater than 12 hours.
* Each request should be processed within 10 seconds.
* The application should load in 3 seconds when the number of simultaneous users is> 10000

## Other Requirements

* The application should use continuous integration so that features and bug fixes can be deployed quickly without downtime.
* Function – function must be reliable for the user.
* Environment – use conditions and profile details.
* Duration – duration must be not enough to boor the user.
* Probability – The likelihood of successfully functioning over the duration.

# 

# Chapter 3

# Use Case Analysis

**Chapter 3:** Use Case Analysis

## Use Cases Description

This Use Case Diagram is a graphic depiction of the interactions among the elements of Inventory Management System. It represents the methodology used in system analysis to identify, clarify, and organize system requirements of Inventory Management System.

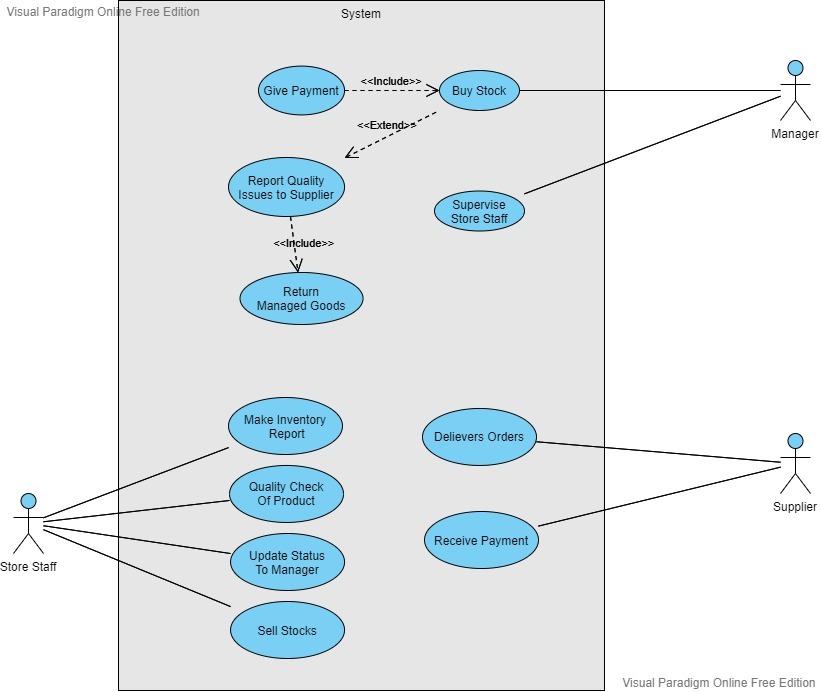


Figure 5 # 3.1: Use Case Model

# Chapter 4

# System Design

**Chapter 4:** System Design

## Domain Model

In software engineering, a domain model is a conceptual model of the domain that incorporates both behavior and data. In ontology engineering, a domain model is a formal representation of a knowledge domain with concepts, roles, datatypes, individuals, and rules, typically grounded in a description logic.

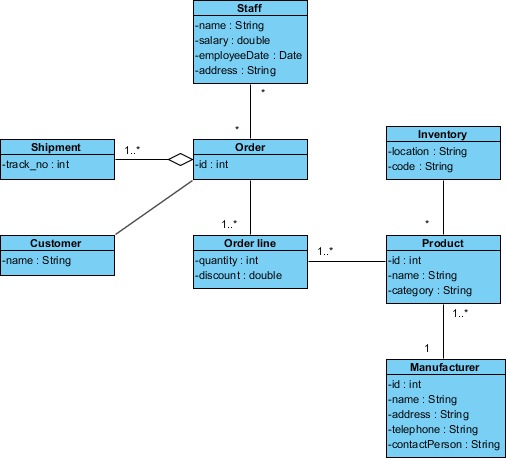


Figure 6 # 4.1.: Domain Model

## Entity Relationship Diagram with data dictionary

ER Diagram is a graphical representation of a data model using entities, their attributes and relationships between those entities. It has a form of a diagram.

Data Dictionary is a list of data elements (entity/table and attribute/column) with their attributes and descriptions. It has a form of a set of tables.

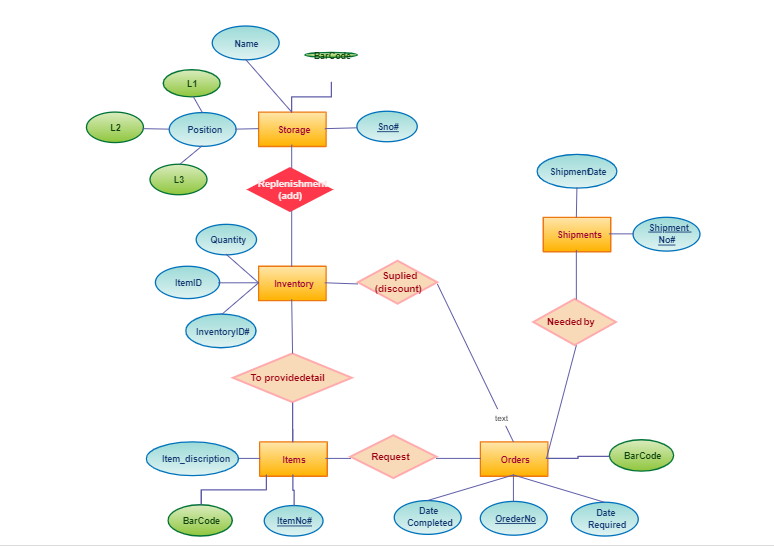


Figure 7 # 4.2.: ERD

## Class Diagram

Class diagrams are used to describe the static view of an application: the main constituents are classes and their relationships. A class is a description of a concept, and may have attributes and operations associated with it. Classes are represented as rectangles. A relationship between two classes is drawn as a line. Inheritance relationships indicate that one class can be inherited by other class, without needing to be explicitly represented in the subclasses themselves.

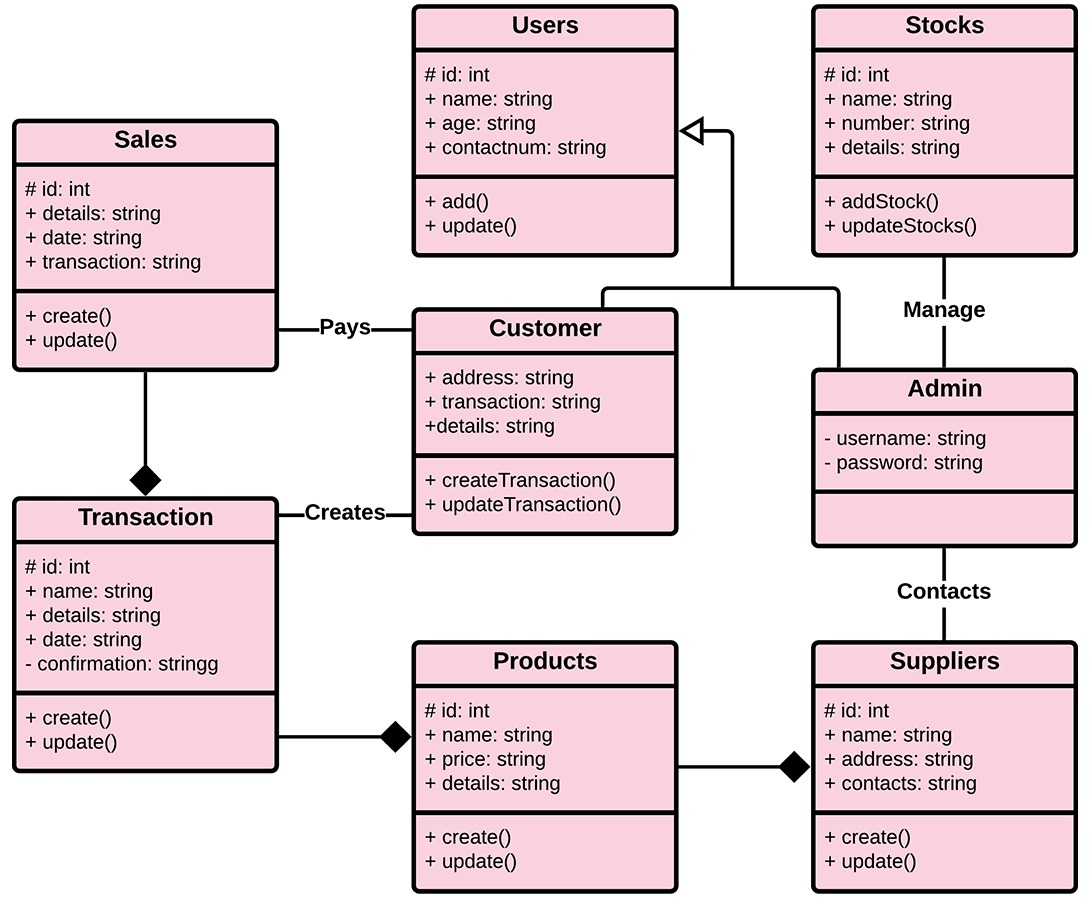


Figure 8 # 4.3.: Class Diagram

## Activity Diagram

Activity diagrams are graphical representations of [workflows](https://en.wikipedia.org/wiki/Workflow) of stepwise activities and action with support for choice, iteration and concurrency. In the [Unified Modeling Language](https://en.wikipedia.org/wiki/Unified_Modeling_Language), activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.

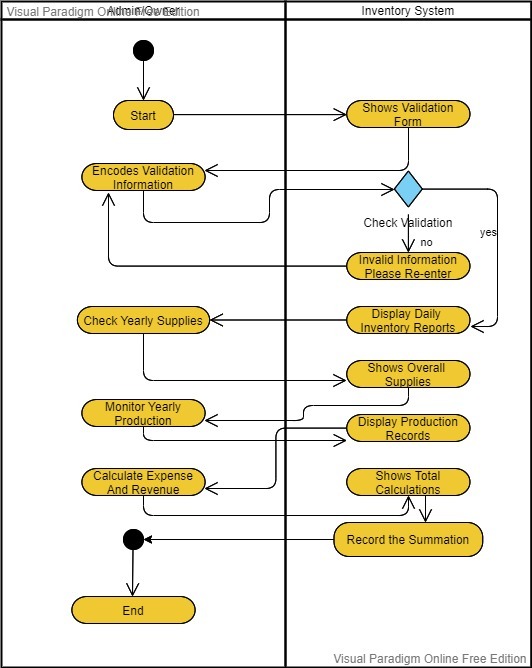


Figure 9 # 4.4: Activity Diagram

## Sequence / Collaboration Diagram

Sequence / Collaboration diagrams and sequence diagrams are alternate representations of an interaction. A collaboration diagram is an interaction diagram that shows the sequence of messages that implement an operation or a transaction. Collaboration diagrams show objects, their links, and their messages.

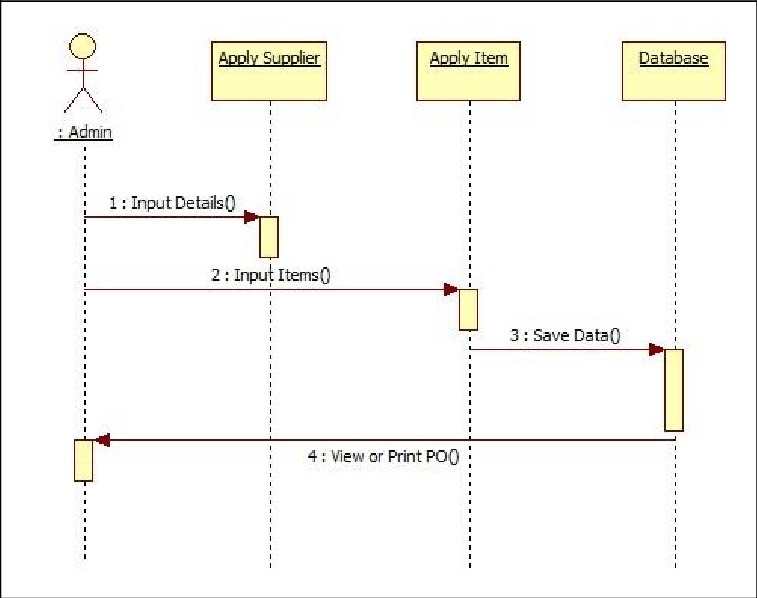


Figure 10 # 4.5: Sequence / Collaboration Diagram

## Deployment Diagram

Deployment diagrams are used to visualize the hardware processors/ nodes/ devices of a system, the links of communication between them and the placement of software files on that hardware.

In this [UML deployment diagram](https://creately.com/diagram-type/uml-diagram/) tutorial, we will cover what is a deployment diagram, deployment diagram notations and how to draw one. You can use one of the editable [deployment diagram examples](https://creately.com/diagram-community/popular/t/deployment-diagram) to start right away.

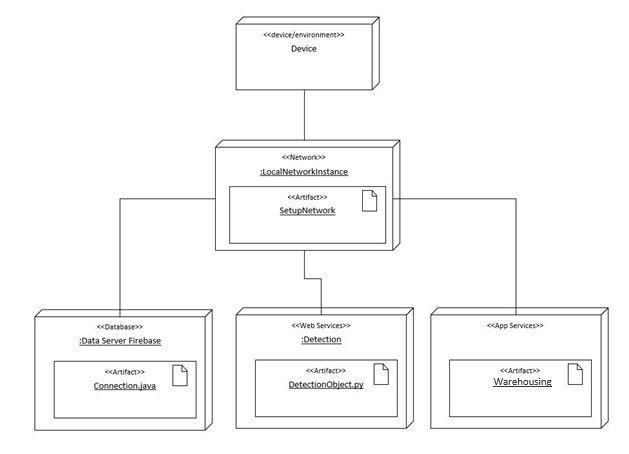


Figure 11 # 4.6: Deployment Diagram

A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them.

Deployment diagrams are typically used to visualize the physical hardware and software of a system. Using it you can understand how the system will be physically deployed on the hardware.

# Chapter 5

# Implementation

**Chapter 5:** Implementation

## Important Flow Control/Pseudo codes

* In application, all code is written as per line.
* In application all code is capitalize initial keyword.
* The hierarchy and structures are well defined
* End multi line Structure supported.
* A logical plan is used to programming in this application.
* Stand raised code and easy to understand and easy to develop again
* Security is used in development also.
* Structured should be implementable in this app
* Accessing the data in the standard form.
* Name or fields and variable name is easy to identify.
* Contains no space in the code
* Additional work is with capital form.
* Consistent use of names.

## Components, Libraries, Web Services and stubs

Table 5 # 5.3: Components, Libraries, Web Services and stubs

|  |  |
| --- | --- |
| **Firebase Database** | Firebase database used to store images and inventory data |
| **Authentication** | Using authentication |
| **Fire Storage** | Fire storage used storing images |
| **Notification Manager** | To notify the warehouse manger |
| **Firebase Real-time** | Firebase real-time used to store Uri and string, int, float data |

## Deployment Environment

* Windows Operating System

## Tools and Techniques

* ReactJS
* Javascript
* Metamask Chrome Extension
* Artificial intelligence
* Ganache
* API

## Best Practices / Coding Standards

A coding standard’s purpose is to restrict use of problematic areas of the programming language. Therefore, using coding standards prevents undefined or unspecified behavior. In addition, it limits the use of error-prone constructs, such as "goto" And using coding standards improves the code’s readability, maintainability, and portability.

* [Maintainability](https://en.wikipedia.org/wiki/Maintainability)
* [Dependability](https://en.wikipedia.org/wiki/Dependability)
* [Efficiency](https://en.wikipedia.org/wiki/Efficiency)
* [Usability](https://en.wikipedia.org/wiki/Usability)

## Version Control

* API use http protocol to connect the user.
* Firebase Server communication needs internet connection on any browser.
* GitHub version control

# Chapter 6

# Testing and evaluations

## Testing Evaluation

Testing & Evaluation is the system analysis in which system components are compared to requirements and specification through testing. We used this process to test our system components. The results are evaluated to assess the progress of design, performance, supportability etc. By this approach we discover defects/ Bugs in the system.

## 6.1 Testing Use Case

Table 6 # 6.1: Test Case 1: Login

|  |  |
| --- | --- |
| Test Case ID | C001 |
| Test Case Name | Login |
| Test Case Description | To verify the user logged in with valid credentials or not |
| Purpose | The purpose to this test case is to verify that the user easily logged in into the system or not |
| Prerequisites | Must have a Wi-Fi connection.  Must have a valid credentials for logged in |
| Test Procedure | Remotely |
| Expected Result | Users can log in into the system with valid credentials |
| Actual Result | Users can enter into the system with valid credentials |
| Status | Pass. |

Table 7 # 6.2: Test Case 2: Dashboard

|  |  |
| --- | --- |
| Test Case ID | C002 |
| Test Case Name | Dashboard |
| Test Case Description | To verify the user logged in application to see dashboard |
| Purpose | The purpose to this test case is to verify that the user sees the dashboard or not. |
| Prerequisites | Must have a Wi-Fi connection.  Must have a Valid credentials for logged in |
| Test Procedure | Remotely |
| Expected Result | Users can log in into the system |
| Actual Result | Users enter into the system |
| Status | Pass. |

Table 8 # 6.3: Test Case 3: Check Records

|  |  |
| --- | --- |
| Test Case ID | C004 |
| Test Case Name | Check Records |
| Test Case Description | To verify the users can checks the records. |
| Purpose | The purpose to this test case is to verify that the user can checks and get the accurate details of warehousing. |
| Prerequisites | Must have a Wi-Fi connection. Must logged in into the system. |
| Test Procedure | Remotely |
| Expected Result | Users can easily see the details on the screen |
| Actual Result | Users check the history |
| Status | Pass. |

Table 9 # 6.4: Test Case 4: Update Data

|  |  |
| --- | --- |
| Test Case ID | C005 |
| Test Case Name | Update Data |
| Test Case Description | To verify that the seller can update data with correct/new information |
| Purpose | The purpose to this test case is to verify that the seller can update the data into the records by adding correct/new information |
| Prerequisites | Must have a Wi-Fi connection. Must logged in into the system  Must have an idea which information wants to update |
| Test Procedure | Remotely |
| Expected Result | Data Updated successfully |
| Actual Result | Data Updated successfully |
| Status | Pass. |

Table 10 # 6.5: Test Case 5: Check Warehouse

|  |  |
| --- | --- |
| Test Case ID | C006 |
| Test Case Name | Check Land Papers |
| Test Case Description | To verify that the user gets warehouse items count or no |
| Purpose | The purpose to this test case is to verify that the seller and buyer both have received the notification. |
| Prerequisites | Must have a Wi-Fi connection. Must logged in into the system |
| Test Procedure | Remotely |
| Expected Result | Date Checks Successfully |
| Actual Result | Date Checks Successfully |
| Status | Passed |

Table 11 # 6.6: Test Case 6: Logout

|  |  |
| --- | --- |
| Test Case ID | C007 |
| Test Case Name | Logout |
| Test Case Description | To verify that the user session is saved after clicking on logout |
| Purpose | The purpose to this test case is to verify the session are saved when user clicked on logout button |
| Prerequisites | Must have a Wi-Fi connection. Must logged in into the system  Must Clicked on Logout Button |
| Test Procedure | Remotely |
| Expected Result | User Logout |
| Actual Result | Users Logout |
| Status | Passed |

Table 12 # 6.7: Test Case 7: Registration

|  |  |
| --- | --- |
| Test Case ID | C009 |
| Test Case Name | Registration |
| Test Case Description | To verify that owner of warehouse be able to register new employ into the system. |
| Purpose | The purpose to this test case is to verify the registration verified when owner clicked on register button. |
| Prerequisites | Must have a Wi-Fi connection. Must have valid data that accept the system. Must be registered in system |
| Test Procedure | Remotely |
| Expected Result | Registered in System |
| Actual Result | Registered in System |
| Status | Passed |

## 6.2 Performance Testing

Performance testing is the type of testing to test the system speed, throughput, responsiveness, and workload. Stability, Reliability, Scalability and other resources that are used in the system. The performance of the software is determined by these factors

* Speed
* Scalability
* Stability

This testing tool test our web application performance. It checks the speed that how much loading speed when web application runs on the browser. It also checks that the application is able to handle the work load on the system and application is stable and error free or not. The record system is working fine and its performance according to the expectation and all the modules are working properly. We tested the Volume testing to test the database.

## 6.3 Unit Testing

This testing is to test the individual components of the application. We divided all the components into the modules and test the modules and testing each module.

## 6.4 Equivalence Partitioning

Using this Equivalence Partitioning method test cases are divided into the three categories or set of input data is classes. Each test cases represent the class so we divide our test case into the three equivalence classes that are used to test the test case with valid and invalid inputs.

### **Login:**

User can be logged by enter valid username and key/password to enter into the system. When user entered the username and key/password system checks that the entered data is already exist in database or not. If it exists it redirect to the dashboard and if it’s not existed in the database the system saves the data into the database and redirect to dashboard.

## 

## 6.5 Boundary Value Analysis

Boundary Value analysis is to analyze and measure the boundary value in the range by the system. The system set the value range for the system.

* Only Authorize person can logged in into the system
* 0 – 100 users can enter into the system

## 6.6 Stress Testing

Stress testing is the type of testing which test how system handles the extreme workload and high traffic or data processing. We used this testing to check that our system handles lots of incoming messages from the server or not. We used Meter tool for the stress testing of our application. This testing mainly determines the system on its robustness and handle extremely heavy load conditions. This ensure that our application works fine under the incredibly hefty burden conditions

## 6.7 Data Flow Testing

Data flow testing is used to analyze the flow of data into the system. It’s the process to analyze that the how-to data flow into the system. Data Flow testing centers focuses on what factors get values and the points at these values are used in which we characterize numerous flows of our application additionally depict how the system flowing in our system.

### **User 1 Data Flow**

The user first turns on his/her Wi-Fi to use the application to check data with someone. Once application is opened user enters the valid username and key/password into the fields. When user entered the username and key/password system checks that the entered data is already exist in database or not. If it exists it redirect to the dashboard and if it’s not existed in the database the system saves the data into the database and redirect to dashboard.

# 

# Chapter 7

# Summary, Conclusion & Future Enhancements

## 7.1 Project Summary

This document is the report about our FYP project; it will give us a detailed view of the data flow and the roles and responsibilities of the actors and the system. There are seven Chapters in this project and the content in them is mentioned below Table.

Table 13 # 7.1: Project Summary

|  |  |  |
| --- | --- | --- |
| ***#*** | ***Chapters*** | **Document Usage** |
| **1.** | ***Introduction*** | This section covers the   * Summary and background of this project. * Project scope, goals, and objectives are also mentioned * Project and work breakdown structure |
| **2.** | ***Software Requirement Specifications*** | This section covers the   * Description of the product * Functional and non-functional requirements * Product interfaces and business rules |
| **3.** | ***Use Case analysis*** | This section covers   * Use cases that are intended for developers, testers, and other technical readers   *“Use case is a list of actions or event steps typically defining the interactions between a role and a system to achieve a goal”* |
| **4.** | ***System Design*** | * This Section is solely intended for the developers and the project managers * Detailed diagrams are the part of this section describing the system architect and the data flow |
| **5.** | ***Implementation*** | This section describes   * The tools and techniques used for the development * Development environment components and best practices are also mentioned. |
| **6.** | ***Testing and Evaluation*** | This section describes   * This section describes the types of testing through which the System has passed |
| **7.** | ***Summary, Conclusion, and Future Enhancements*** | This section describes   * The overall summary of the chapters * The conclusions made from this Project   The future enhancements that are to be made in it |
| **8.** | ***Appendix*** | This section includes any additional or supplementary  information on the book's topic |

## 

## 7.2 Achievements and Improvements

By working on this project, we got hands-on practical experience on AI and web development knowledge that we gained during our 4 years of degree this project cleared our basic to a high level of concepts or any confusion that we had. We now have a thorough understanding of how does the AI and database work. By doing this project we not only technologically but as a team learned many things. We learned to do teamwork and how to manage and divide work and save time by doing so. Teamwork is the collaborative effort of a group to achieve a common goal or to complete a task most effectively and efficiently. This concept is seen within the greater framework of a team, which is a group of interdependent individuals who work together towards a common goal. Contribute to Open-Source Projects, Teach Others What You Know, Challenge Yourself with a New Skill, Schedule Regular Downtime.

Our Improvement Involves increase in technical abilities these skills include working with modern-day technologies like

* Frontend & Backend **skills**.
* Responsive design **skills**.
* AI **skills**.
* Testing and debugging **skills**.

Amplifying the need for good communication skills is the fact that coding is something that is very hard to grasp for non-developers. You must be able to make other people understand technical problems, which is something that can be tough. As a way of improving Communication skills, a great idea is to have regular talks with people you know outside of the office and try to make them understand what you are working on.

## 7.3 Critical Review

The warehouse management process is known to be a multi-step process because it requires the involvement of all stakeholders directly or indirectly involved in warehouse management. The warehouse / inventory storage systems in use today pose major problems in terms of data fraud, sensitive data security, and the risk of system failure due to natural disasters. If the server used for data storage fails. Compared to current approaches and practices for warehouse management and data storage, AI is a state-of-the-art technology and database that can fully address the problems that plague current systems. The basic and most important aspect of AI technology is intelligence demonstrated by machines, as opposed to the natural intelligence displayed by humans.

## 7.4 Lessons Learnt

This section describes the lesson learned in the process of making this software working which is mentioned below

### **7.4.1 Applying Knowledge**

The project provided us the opportunity to apply the knowledge that we learned during our 4 years of the degree, and to self-explore and learn new technologies that would prepare us for over future.

### **7.4.2 Collaboration**

It’s incredibly important to have close collaboration with the full project team at the very early stages of the project before the budget is set. Do not make key decisions without your project team already on board.

Collaborating early in the project will save you both time and heartache.

### **7.4.3 Be very patient**

Having trust in each other and on Allah being patient is one of the most important things that we learned from this project do not give up, keep working hard you will achieve your goal.

## 7.5 Future Enhancements/Recommendations

This software was a very basic one, saving multiple issue and time, and demonstrates the idea we had. For better and more accurate results, preferable system that a user giving instruction through view to save time and life. We can conclude that this project of ours has a vast scope in future application.

# Reference and Bibliography

**Reference and Bibliography**

**What is Artificial Intelligence and why we use it.**

Data Science Central: <https://www.datasciencecentral.com>

**Three ways AI can help solve inventory management challenges**

IBM: <https://www.ibm.com/blogs/supply-chain/>

**Artificial Intelligence in Warehousing:**

[Freshservice](https://freshservice.com/): <https://freshservice.com>

Inventory Planner: <https://www.inventory-planner.com>

Unleashed: <https://www.unleashedsoftware.com>

In Data Labs: <https://indatalabs.com/blog/ai-inventory-management>

Matello: <https://www.matellio.com/ai-based-inventory-management-software>