**Inventory Management System**

**Deliverable -5**

**Code Crafters**

**Group List**

1. Sameera Ellanki
2. Varsha Jampala
3. Nikhil Kuchipudi
4. Naga Shivani Pinnamaraju
5. Jaswanth Sai Donepudi
6. Manogna Reddy Anugu
7. Atluri Varun Chowdary
8. Naga Bindu Kondaveeti

Index

[1. Requirements specification for phase one 1](#_Toc165314560)

[1.1 REQ-1 Inventory Management Requirements 1](#_Toc165314561)

[1.2 REQ-2 Warehouse Management Requirements 1](#_Toc165314562)

[1.3 REQ-5 Reporting and Analytics Requirements 1](#_Toc165314563)

[1.4 REQ-6 System Administration Requirements 2](#_Toc165314564)

[1.5 REQ-7 User Interface Requirements 2](#_Toc165314565)

[1.6 NF-5 Usability 2](#_Toc165314566)

[1.7 NF-1 Performance 2](#_Toc165314567)

[1.8 NF-3 Reliability and Availability 3](#_Toc165314568)

[1.9 NF-6 Maintenance 3](#_Toc165314569)

[2 UML Diagrams 4](#_Toc165314570)

[2.1 Class Diagram 4](#_Toc165314571)

[2.2 Sequence Diagram 5](#_Toc165314572)

[2.3 Use case diagrams 6](#_Toc165314573)

[2.3.1 Normal Case 6](#_Toc165314574)

[2.3.2 Error Case 7](#_Toc165314575)

[3 Test Cases 8](#_Toc165314576)

[4 User Manual 12](#_Toc165314577)

[5 Compilation Instructions 20](#_Toc165314578)

[6 Report Ending Feature Summary 21](#_Toc165314579)

[7 Reflection 22](#_Toc165314580)

[8 Member Contribution Table 23](#_Toc165314581)

# Requirements specification for phase one

## REQ-1 Inventory Management Requirements

* **REQ-1.3 Bin Location System**

Users must able to precisely track product locations as well as inventory locations within warehouse using the system. To enable effective inventory management this feature should have real - time location data updating and retrieval capabilities.

## REQ-2 Warehouse Management Requirements

* **REQ-2.1 Warehouse Operations**

Users should be able to manage and keep an eye on inventory in particular warehouse with the help of the system. this involve see every item that is currently on hand in the warehouse as well as the locations of item in any particular warehouse.

* **REQ-2.2 Bin Location Management**

To achieve the good possible inventory organisation the system provide comprehensive bin location management. This include features that allow you to efficiently search for and retrieve product information as well as enter and update the bin location of a registered product in the system.

* **REQ-2.3 Stock Movement**

Upon navigating to the warehouse module user should have the ability to take bulk actions to the target products like changing the sku and expiry dates.

## REQ-5 Reporting and Analytics Requirements

* **REQ-5.2 Performance Metrics**

The system should show Analytics on order fulfilment rates and supplier performance, these will be generating as a report list.

## REQ-6 System Administration Requirements

* **REQ-6.1 User Access Control**

Only one user should be able to access the organization and manage all its associated modules, we changed this requirement because of different companies being able to use different hierarchical makes it complicated to implement user rolls to access control.

* **REQ-6.2 Security Management**

The system shall ensure data integrity and access security within the system.

## REQ-7 User Interface Requirements

* **REQ-7.1 Dashboard**

User-friendly dashboard for real-time data display, like graphs for the number of sales, profit and other useful information.

* **REQ 7.2 Navigation**

Navigation within the inventory management system should be streamlined and intuitive, allowing users to easily transition between different modules and features without confusion, thereby enhancing user experience and system usability.

## NF-5 Usability

* **NF-5.1 Friendly UI UX**

The system's user interface (UI) should be designed to be simple and intuitive, organizing all features clearly to minimize user learning time and maximize efficiency. This includes clear labels, logical feature placement, and responsive design.

## NF-1 Performance

* **NF-1.1 System Performance**

The system must be optimized to handle multiple simultaneous client requests effectively without degradation in response times or performance.

* **NF-1.2 Data Handlin**

The system must be capable of efficiently managing large volumes of data with minimal impact on performance. This capability should be scalable depending on the hardware specifications of the server on which the system is deployed.

## NF-3 Reliability and Availability

* **NF-3.1 System Uptime**

The system should be designed to maximize uptime and minimize downtime, ensuring reliable access for users which is crucial for maintaining operational continuity and user satisfaction.

## NF-6 Maintenance

* **NF-6.1 Components**

The system architecture should be modular, allowing for individual components to be maintained, upgraded, or replaced independently without affecting the overall system functionality.

* **NF-6.2 Updated Documentation**

Comprehensive and up-to-date documentation should be maintained, covering all system functionalities, including minor features and updates, to support effective use, maintenance, and future enhancements of the system.

# UML Diagrams

## Class Diagram

The Phase 3 class diagram of the "Inventory Management System" lists attributes below the rectangles that represent the objects, with class names placed in the upper section. Links are used to show relationships between various objects, and lines are used to indicate the connections between these objects.

A diagram of a company

Description automatically generated

## Sequence Diagram

Sequence diagram for the "Inventory management system" in relation to the requirements for Phase 3. Objects are the entities that converse within the system. Vertical lines are used in the sequence diagram to represent them.

**Bin Location Creation sequence diagram**

A diagram of a software company

Description automatically generated

## Use case diagrams

### Normal Case

In the Normal case we can see the warehouse details without any hiccups and the bin location can be created and updated and deleted at any time also we can search for items inside of the bin locations module.

A diagram of a warehouse management system

Description automatically generated

### Error Case

* Invalid input during bin location management triggers error messages, preventing data inconsistencies.
* Attempts to delete non-empty bin locations result in errors, ensuring all items are relocated first.
* Connectivity issues or unauthorized access attempts are flagged, maintaining system security and integrity.

A diagram of inventory management system

Description automatically generated

# Test Cases

There is no explicit way to execute test cases for now we must do the test cases manually. You can follow the following things to do the actions for input and output results. **NOTE: Few sections might not be able to delete because they are used elsewhere.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Description** | **Input** | **Output** | **Test Result** |
| Check if the signup works | Name: name  Email: [email@email.com](mailto:email@email.com)  Password: 12345678  Org Name: Org | Navigates to login page | pass |
| Check if the signup works, if given a same email adress | Name: name  Email: [email@email.com](mailto:email@email.com)  Password: 12345678  Org Name: Org | Shows error | pass |
| Check if the Login works | Email: [email@email.com](mailto:email@email.com)  Password: 12345678 | Navigates to dashboard | pass |
| Check if the Login works if given wrong password | Email: [email@email.com](mailto:email@email.com)  Password: 123456 | Nothing happens | pass |
| Item add | Navigate to items and click on add new item, fill the form and create. | Item creates | pass |
| Item update | Click on the three dots on an item and select the edit icon, update the form and submit | Item updates | pass |
| Item delete | Click on the three dots on an item and select the button. | Item delete | pass |
| Customers add | Navigate to Customers and click on add new Customer, fill the form and create. | Customers creates | pass |
| Customers update | Click on the three dots on an Customers and select the edit icon, update the form and submit | Customers updates | pass |
| Customers delete | Click on the three dots on an Customer and select the button. | Customers delete | pass |
| Vendors add | Navigate to Vendors and click on add new Vendor, fill the form and create. | Vendors creates | pass |
| Vendors update | Click on the three dots on an Vendor and select the edit icon, update the form and submit | Vendors updates | pass |
| Vendors delete | Click on the three dots on an Vendor and select the button. | Vendors delete | pass |
| Create a Sales Order | Click on create sales order button and fill the form. | SO should be created and stock of the selected items should be reduced. | pass |
| Edit a Sales Order | Click on three dots select Edit change the records as needed. (some are restricted from editing) | SO should be edited accordingly. Stock is adjusted accordingly | pass |
| View a Sales Order | Click on 3 dots select the view button. | SO opens with all disabled columns. | pass |
| Delete a Sales Order | Click on 3 dots and select delete. | SO should be deleted and stock is put pack. | pass |
| Create a Purchase Order | Click on create purchase order button and fill the form. | PO should be created and stock of the selected items should be updated. | Pass |
| Edit a Purchase Order | Click on three dots, select Edit, and change the records as needed. (some are restricted from editing) | PO should be edited accordingly. Stock is adjusted accordingly. | Pass |
| View a Purchase Order | Click on three dots, select the view button. | PO opens with all disabled columns. | Pass |
| Delete a Purchase Order | Click on three dots and select delete. | PO should be deleted and stock is updated accordingly. | Pass |
| Create Invoice | Navigate to Invoice module click on create Invoice and select a customer and a sales order and hit create. | A invoice with calculated amount due should be displayed. | Pass |
| Update Invoice | If the corresponding SO is updated we can update the invoice by selecting 3 dots update button. | Invoice amount due should be updated | Pass |
| Delete Inoice | Select 3 dots hit delete button | Invoice should be deleted. | Pass |
| Create a Bill | Navigate to the Bills module, click on create Bill, select a vendor and a purchase order, then hit create. | A bill with calculated amount due should be displayed. | Pass |
| Update Bill | If the corresponding purchase order is updated, select the bill, then click on three dots and select the update button. | Bill amount due should be updated accordingly. | Pass |
| Delete Bill | Select the bill, then click on three dots and hit delete. | The bill should be deleted. | Pass |
| Create Warehouse | Navigate to the Warehouse module, click on create Warehouse, Enter Name and address and hit create. | The Warehouse should be crated | Pass |
| Edit Warehouse | On the Actions Click on edit and edit the warehouse information. | The warehouse information edited correctly | Pass |
| Delete Warehouse | Click on Actions select delete. | The warehouse should be deleted | Pass |
| View warehouse Items | Clicking on the eye button in the actions menu should display all warehouse items. | The warehouse items displayed. | Pass |
| Create Bin Locations | Navigate to the Bins module, click on create Bins, select a Warehouse and item and enter the location. | The bin location created successfully | Pass |
| Edit Bin Locations | Edit the bin location by clicking on the actions Edit button. | The bin location is edited correctly with the new information. | Pass |
| Delete Bin Locations | Click on the actions delete button. | The bin should be deleted successfully. | Pass |
| Dashboard | Click on dashboard and view the results. | The stats should be displayed. | Pass |

# User Manual

**System Access:**

Requirements: Internet connection, modern web browser.

URL: Access the system by entering the provided one of the URL after running the UI repository.

**Create Account:**

Click “Sign Up” on the login page, enter your details, and submit.

Make sure the email wasn’t used before.

A screenshot of a computer

Description automatically generated

**Logging In:**

Login: Enter your email and password on the login page and click “Login.”

A screenshot of a login form

Description automatically generated

**Accessing the Dashboard:**

Login to the system using your credentials. The dashboard will be the first screen you see, displaying key metrics and real-time data relevant to inventory and sales performance.

Various widgets and graphs will provide insights into sales trends, inventory levels, and operational metrics.

A screenshot of a graph

Description automatically generated

**Managing Items:**

Add: Go to “Items” > “Add New Item,” fill in details, and save.

Update/Delete: Find the item under Items action by clicking on the three dots, select “Edit” to update or “Delete” to remove.

A screenshot of a computer

Description automatically generated

**Managing Vendors:**

Add: Navigate to “Vendors” > “Add New Vendor,” enter details, and save.

Update/Delete: Choose a vendor action by clicking on the three dots, click “Edit” to update or “Delete” to remove.

A screenshot of a computer

Description automatically generated

**Managing Customers:**

Add: Go to “Customers” > “Add New Customer,” fill out the form, and click “Save.”

Update/Delete: Select a customer action by clicking on the three dots, “Edit” to update their information, or “Delete” to remove them.

A screenshot of a computer

Description automatically generated

**Managing Sales Orders:**

Create: Click on the “Create Sales Order” button, fill out the form with the necessary details, and submit to create a new sales order. This action will reduce the stock of selected items.

Edit: Select the sales order you wish to edit, click on the three dots, and choose “Edit” to make changes to the sales order records. Note that some fields may be restricted from editing.

View: Click on the three dots of a sales order and select “View” to open the sales order with all columns disabled for viewing purposes.

Delete: To delete a sales order, click on the three dots of the respective sales order and select “Delete.” This action will put the stock back to its previous state.

A screenshot of a computer

Description automatically generated

**Managing Purchase Orders:**

Create: Navigate to the Purchase Orders module, click on “Create Purchase Order,” fill out the form with the necessary details, and submit to create a new purchase order. This action will update the stock of selected items accordingly.

Edit: Select the purchase order you wish to edit, click on the three dots, choose “Edit,” and make the required changes. Stock adjustments will be made accordingly.

View: Click on the three dots of a purchase order and select “View” to open the purchase order with all columns disabled for viewing.

Delete: To delete a purchase order, click on the three dots of the respective purchase order and select “Delete.” This action will update the stock accordingly.

A screenshot of a computer

Description automatically generated

**Managing Invoices:**

Create: Navigate to the Invoice module, click on “Create Invoice,” select a customer and a sales order, then click “Create” to generate an invoice with the calculated amount due.

Update: If the corresponding sales order is updated, select the invoice, click on the three dots, and choose “Update” to reflect the changes in the invoice amount due.

Delete: To delete an invoice, select the invoice, click on the three dots, and hit “Delete.” This action will remove the invoice from the system.

A screenshot of a computer

Description automatically generated

**Managing Bills:**

Create: Access the Bills module, click on “Create Bill,” select a vendor and a purchase order, then click “Create” to generate a bill with the calculated amount due.

Update: If the corresponding purchase order is updated, select the bill, click on the three dots, and select “Update” to ensure the bill amount due is updated accordingly.

Delete: To delete a bill, select the bill, click on the three dots, and hit “Delete.” This action will remove the bill from the system.

A screenshot of a computer

Description automatically generated

**Warehouse Management:**

Create: Click on Create Warehouse, enter the necessary details such as name, address, and capacity, then click Create. This will add a new warehouse to your system.

Edit: Find the warehouse you wish to edit in the warehouse list, click on the Edit icon from the actions menu.

View: To view detailed information about a warehouse, simply click on the View icon in the actions menu next to the respective warehouse.

Delete: In the warehouse list, select the warehouse you want to delete and click on the Delete icon in the actions menu.

A screenshot of a computer

Description automatically generated

**Bin Locations Management**

Create: Click on “Create Bin”, select a warehouse, assign an item, and specify the location within the warehouse. Fill out all necessary fields and click Create.

Edit: To edit a bin location, navigate to the bin location list, click on the Edit icon for the location you wish to update, make the changes, and save.

Delete: To delete a bin location, click on the Delete icon for the respective location in the list. Ensure the bin is empty before attempting deletion.

A screenshot of a computer

Description automatically generated

**Logging Out:**

Logout: Click the red button on the side panel called Logout.

# Compilation Instructions

**Deploying the server**

* Navigate to the server files.
* Install Node Package Manager (npm is bundled with node) by installing node. (<https://nodejs.org/en/download>)
* Run the command “npm i”.
* Next Run “npm start”.

**Deploying the UI**

* Open a new terminal but don’t close the server one.
* Navigate to the UI files.
* Install yarn using “npm install --global yarn”.
* Run the command “yarn”.
* Next run “yarn dev”.
* Open one of the links that pops up in the terminal.

The test cases should be tested manually as per the given instructions above in the test cases section.

This can be done only after running the application successfully.

# Report Ending Feature Summary

**Successfully Implemented Features:**

* Inventory Dashboard
* Items Module
* Customers Module
* Vendors Module
* Sales Orders
* Invoices
* Purchase Orders
* Bills
* Warehouse
* Bin Locations

**Limitations and Known Issues:**

* Scalability: The current system requires further testing under increased loads to ensure scalability.
* Complex UI Features: Some advanced dashboard features have not been fully optimized for all end users particularly in varying screen resolutions.

**Future Plans and Improvements:**

* Enhanced Data Handling and Scalability: Plans to integrate more robust data management techniques to handle larger datasets efficiently or change the database completely.
* Advanced Analytics: Features introduction of more complex analytics tools to provide deeper insights into inventory trends and performance metrics.
* Mobile Optimization: Developing a mobile friendly version of the system to allow users to manage inventory remotely.
* Security Enhancements: Strengthen security measures to protect sensitive data and prevent unauthorized access.

# Reflection

The dashboard warehouses and bin locations modules have all undergone significant development and implementation which have improved the system 's overall effectiveness and user - friendliness.

**Accomplishments:**

* Dashboard: The user experience has greatly improved by the integration of intuitive navigation and real - time data displays.
* Warehouses: The system make it possible to precisely track and manage inventory across several warehouses.
* Bin Locations: To help effective management of inventory organisation within the warehouses the bin location system has be simplified.

**What Went Well:**

* One standout feature of the dashboard is user interface design, which provides a smooth experience to reduce learning curves and increase operational effectiveness.
* The team 's ability to work together has produced creative answers and efficient problem - solving, especially when it comes to integrating the bin locations with the warehouse management system.

**Areas for Improvement:**

* Incorporating User Feedback: Regular and organised feedback session with end users could improve the dashboard 's usability and functionality even more.
* Flexibility Warehouse Management: Although the current system does a good job managing standard operations it would be possible to handle a greater range of inventory types and logistical challenges by incorporate more flexible warehouse management features.
* Bin Location Optimisation: To maximise space utilisation and further cut down on retrieval time the algorithm for bin location assignments could be improved.

# Member Contribution Table

|  |  |  |
| --- | --- | --- |
| **Member Name** | **Contribution description** | **Overall Contribution (%)** |
| Sameera | Worked on first two dashboard APIs, repository maintenance and cleanup of code. | 12.5 |
| Varsha | Worked on Warehouse API, contributed to documentation. | 12.5 |
| Nikhil | Worked on Bin Locations API, Drawn the class diagram. | 12.5 |
| Jaswanth | Worked on second two Dashboard APIs, written test cases. | 12.5 |
| Varun | Worked on Dashboard UI, updated manual | 12.5 |
| Shivani | Worked on Dashboard UI, Drawn the sequence diagram and use case diagrams. | 12.5 |
| Bindu | Worked on Warehouse UI, wrote reflection and code inspection feedback. | 12.5 |
| Manogna | Worked on Bin Locations Bills UI, wrote meeting minutes. | 12.5 |