|  |  |
| --- | --- |
| Internship Project Title | Operations Management-Inventory Module using MERN Stack |
| Name of the Company | TCS ion |
| Name of the Industry Mentor | Debashis Roy |
| Name of the Institute | ICT Academy of Kerala |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Start Date | End Date | Total Effort (hrs.) | Project Environment | Tools used |
| 11-04-2023 | 08-06-2023 | 125 hrs. | MERN Stack (Mongo DB, Express JS, React JS, Node JS) | VS Code, npm, create-react-app, MongoDB Compass, postman, Chrome developer tools |

**TABLE OF CONTENT**

* Acknowledgements
* Objective
* Abstract
* Introduction
* Module description
* Internship Activities
* Approach / Methodology
* Data Flow Diagram(DFD)
* Flow Chart
* Project Milestone
* System Implementation
* Hardware Requirement
* Software Requirement
* Screen shots
* Reflections on the Internship
* Recommendations
* Outcome / Conclusion
* Enhancement Scope
* Link to Code
* Link to Demo Video

**Operations Management-Inventory Module**

**Acknowledgement**

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I am highly indebted to ICT Academy of Kerala for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in doing the project.

I would like to express my special thanks to TCS, who give me the golden opportunity to do this wonderful project on the topic of Operations Management. I am also very grateful to the industry mentor Mr. Debashis Roy for being mentor of my project.

Finally, I would like to than my relatives and close friends for their kind co-operation and encouragement which help me in completion of this project

**Objective**

        The primary goal of this system is to design or develop an application to manage the inventory and order for all type of business. This system enables an easy-to-use environment to carry out their duties in accordance with the circumstances.

**Abstract**

     The Operations Management is a easy to use yet powerful and flexible inventory and order management system designed for businesses of all sizes to meet their end-to-end inventory management need. It helps us to track the progress of inventory, manages sales order for created customer, manage packaging and Shipment of order, Generating delivery challans,

Help to raise new invoice, manage received payments for customer. Handle Sales return, and create its credit note. In addition to sales module, it also handles purchase module.it include, Vendor creation Raising Purchase order for created vendor, Tracking bills and payment made to vendor and generate vendor credit notes. It also featured with a dash bord of analysis and reports.

**Introduction**

     The Operation Management is a powerful inventory management application that lets you manage all you order. You can manage your contact bills and reports and generate report easily. It helps us to track the progress of inventory, manages sales and purchase orders. The inventory automatically synchronizes stock across all channels. Any change made in one channel will instantly appear across all others, eliminating the risk of listing errors. Featured with a content management Dashboards, Reports and analysis.

**Module Description**

This project contains four modules:

1. Inventory
2. Sales
3. Purchase
4. Dashboard and reports

**Inventory module**

1. Item Group.
2. Items
3. Inventory Adjustment

**Sales Module**

1. Customer
2. Sales Order
3. Packages
4. Delivery Challans
5. Invoices
6. Payment Received
7. Sales Return
8. Credit Notes

**Purchase Module**

1. Vendor
2. Purchase Order
3. Bills and Payment
4. Vendor Credit

**Dash board and Report Module**

1. Dashboard
2. Inventory Summery Report
3. Inventory Aging Summary Report
4. Product Sales Report
5. Sales by Items/Customer

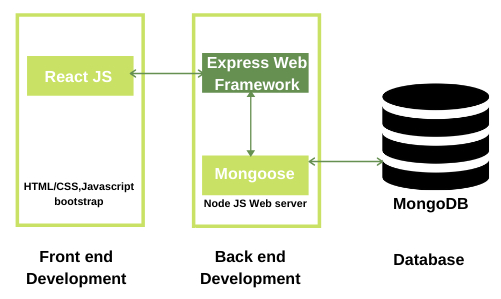
**Internship Activities**

* The Internship activity started with self-study of the system. Then made a rough idea of Form and Database design.
* Then created a backend project using Express.js and Node.js.Then npm packages Initialized, Installed dependencies and set the entry point.
* Created Mongo DB and established connection with the application
* Created Restful APIs using MERN stack
* Created Database Models
* Created backend functions required
* Created a React App
* Installed Packages as per requirement
* Started Front end coding
* Created css files as per requirement
* After completion done proper testing
* Daily activity reports, Interim reports created and sent
* Project final Report Created
* Uploaded the Project contents

**Approach/Methodology**

MERN stack stands for mongo db as database, express framework on node js as the backend server and react is for creating the frontend. MERN stack has the advantage that it uses the same script at all the stacks i.e. JavaScript. So knowing JavaScript alone is sufficient to make web applications using the MERN stack. So MERN stack is selected for doing this project.

An ODM *mongoose* is used to make data operations with mongo db simpler. Express framework is used to make setting up http server for the backend which acts as a REST API easier. React js which uses components makes it easier to design the frontend for single page applications and the create-react-app framework makes developing react applications easier. Bootstrap is used, so that the web application can be made accessible on many devices having varied screen sizes.



MERN Architecture

The above figure shows the basic MERN architecture and this is what is used in doing this industry project. The front end consists of the data entry forms as well as reports and tables for showing data delivered from backend. Asynchronous requests are made between the front end and backend using the axios package. At the backend express helps in handling the http requests as well as responses. Before sending data from data entry form, there is a primary validation and at the backend also the data is validated before sending to mongo db. Mongoose does validations based on the schema given for the data model. The end points usually used for submitting data use the post verb, while that for getting data use the get verb. HTTP put is also used, in case of updating an existing entry. Express also handles the routing for the various API calls made from the frontend.

For multipart-form requests, *multer* is used. *Multer* is used in this project for uploading item images. In case of item images, they are uploaded to a public folder on the node server with a dynamically created filename and its filename is saved in mongo db. When getting item details, the path of the image is created from its filename stored in the database.

The html-pdf package is used to generate pdf files from HTML template. For each report, I made a HTML template and it was converted to pdf using the create function of this package. The create function consists of two parameters namely the HTML string and options object. In options, it is possible to mention the orientation as well as paper size. The created pdf was then streamed to front end using binary buffer and specifying response type as pdf.

The front end consists of many data entry forms, tables and reports. All the data entry forms follow the same logic as shown in the flowchart. Each form will be explained in the succeeding sections.

**Data Flow Diagram (DFD)**

**Level 0**

User

**Level 1**

User

Inventory Management

Sales Management

Purchase Management

Dash board and Reports

DB

**Level 2**

User

User

Customer

Sales Order

Packages

Delivery Challan

Invoice

Payment Received

Sales Return

Credit Note

Vendors

Purchase Order

Bill

Payment

Vendor Credit

Item Group

Items

Inventory Adjustment

DB

Product Sale Report

Customer wise Sale

Item wise Sale

Inventory Summary

Dash Board

Inventory Aging Summary

**Flow Chart**

Start

Dash Board

Customer

Item Group

Sales Order

Item

Inventory Adjustment

Packages

Vendor

Delivery Challan

Purchase Order

Bill

Sales Return

Payment

Payment Received

Vendor credit

Invoice

Credit Note

Stop

**Project Milestones**

 This Project have two milestones

Milestone 1 of this project includes two Major Tasks

* **Install and configure necessary software required for MERN stack**
* **Complete Inventory Module**

Milestone 2 of this project includes two Major Tasks

* **Complete Sales Module**
* **Complete Purchase Module**
* **Complete Dashboard and Report Module**

**System Implementation**

**Project Title: Operations Management-Inventory Module**

**Description**

        The Operations Management is an easy to use yet powerful and flexible inventory and order management system designed for businesses of all sizes to meet their end-to-end inventory management need. It helps us to track the progress of inventory, manages sales order for created customer, manage packaging and Shipment of order, Generating delivery challans,

Help to raise new invoice, manage received payments for customer. Handle Sales return, and create its credit note. In addition to sales module, it also handles purchase module.it include, Vendor creation Raising Purchase order for created vendor, Tracking bills and payment made to vendor and generate vendor credit notes. It also featured with a dash bord of analysis and reports.

**Hardware Requirement**

* Processor: Pentium 4 (or equivalent)
* 4 GB RAM
* Hard disk space: 20 GB
* A projecting device (for the instructor only)
* A connection to the internet
* Keyboard and mouse or another pointing device

**Software Requirement**

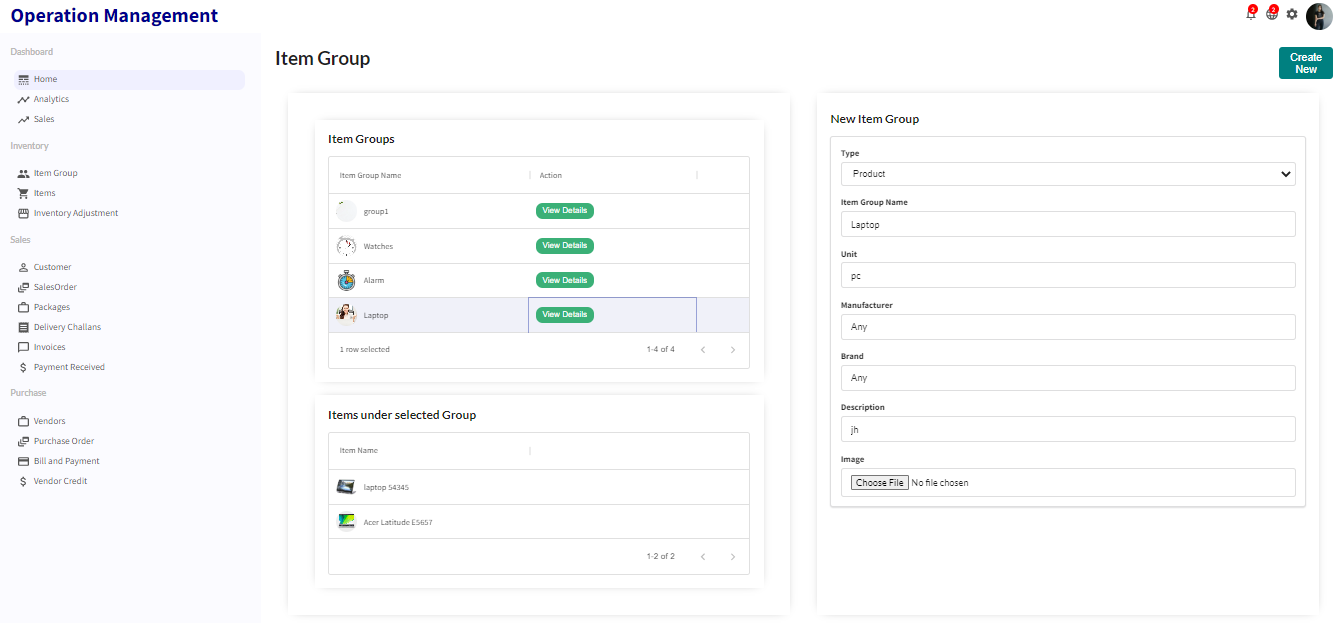
* Operating System: Windows 10 or above
* Node.js
* Supported Internet browser: Chrome - Latest version, or the penultimate version

**Screenshots**

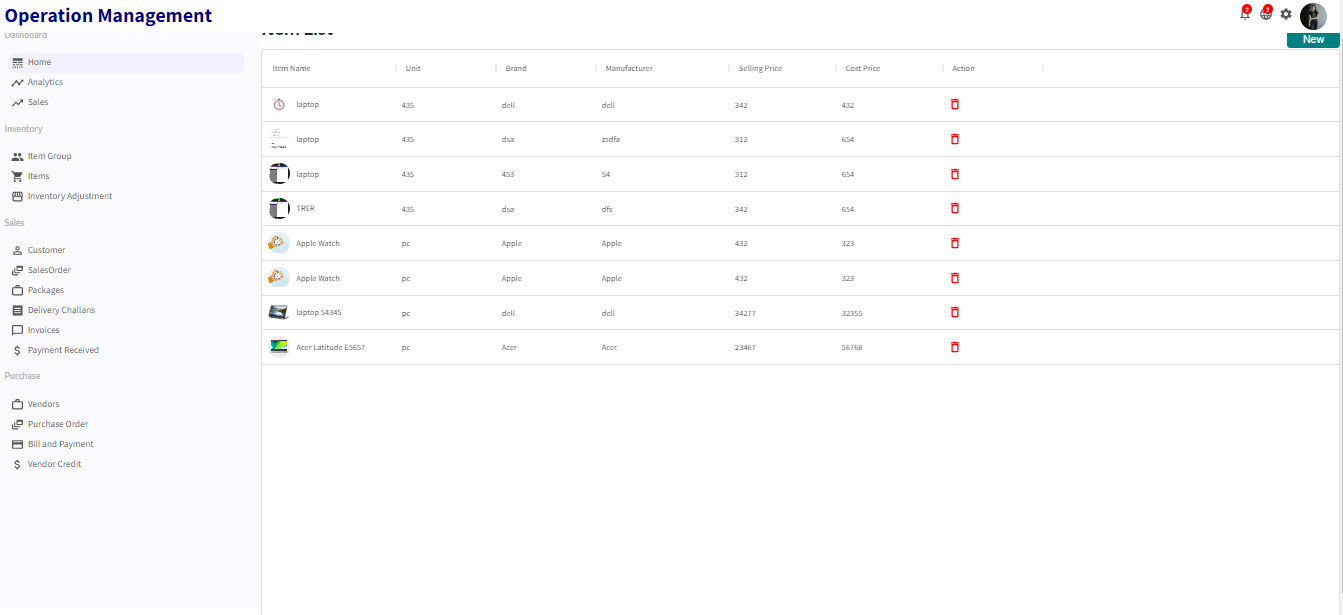
**Action Term1-Inventory Module**

**Item Groups**

An item group is a collection of products which share similar attributes like color, production, features, or usage. Item groups can also be formed based on the markets in which they’re sold or if they’re similar in price.  In this web application

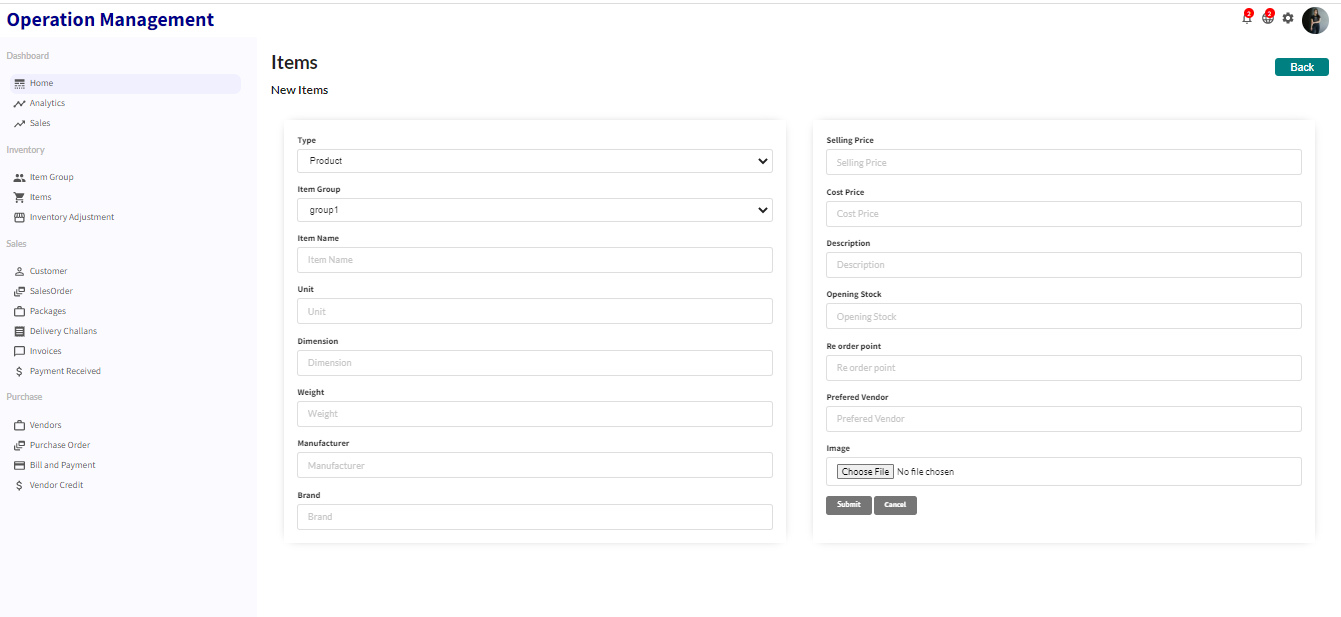


**Item List**

It list all created items. User can delete a specific item on clicking the delete icon for the specific item.

**New Item**

This form is used for entering inventory items. In this form, it is possible to assign an item group to each item. Based on the item, other details can be filled. The file input is used to upload the item image. This form is a multipart form and its posting is handled by multer as a middleware at the backend before inserting to database. Updating certain fields like cost price, selling price, reorder point, preferred vendor and description is possible.

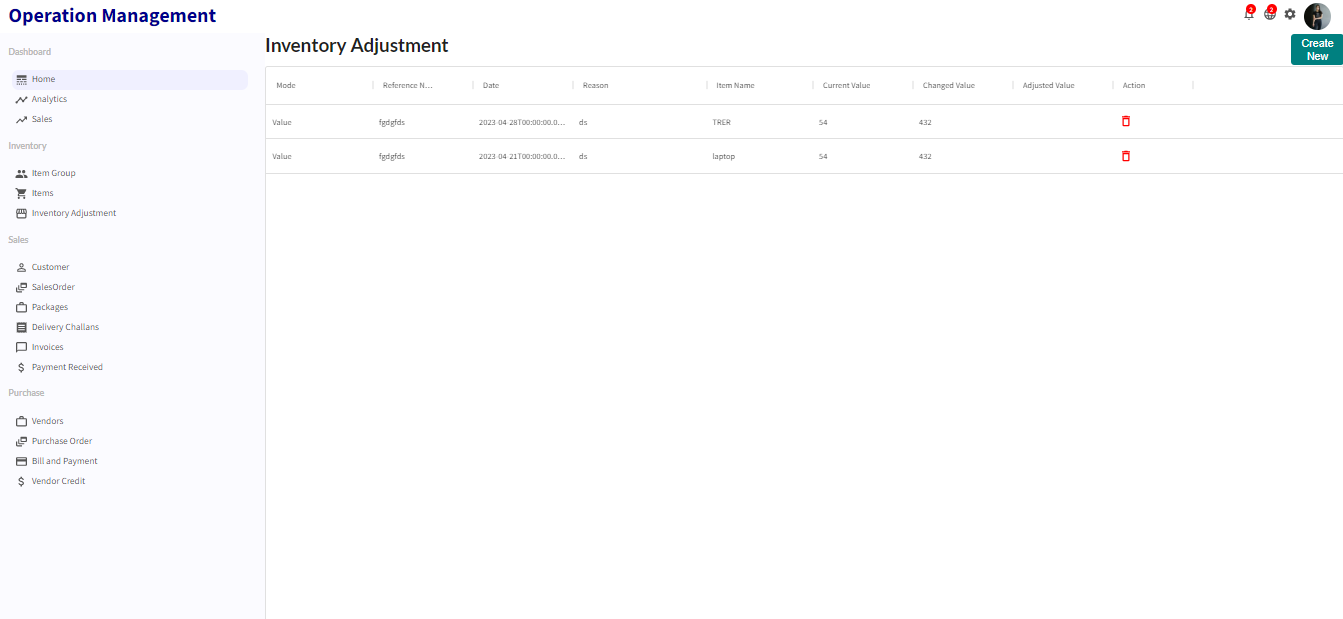


**Inventory Adjustment**

Inventory adjustments are corrections of inventory or stock records to bring them into agreement with the findings of the actual physical inventory. Inventory adjustments are increases or decreases made in inventory to account for theft, loss, breakages, and errors in the amount or number of items received. In this form inventory adjustment mode includes two options namely increase or decrease. The item can be selected and the quantity for adjustment of inventory can be input.

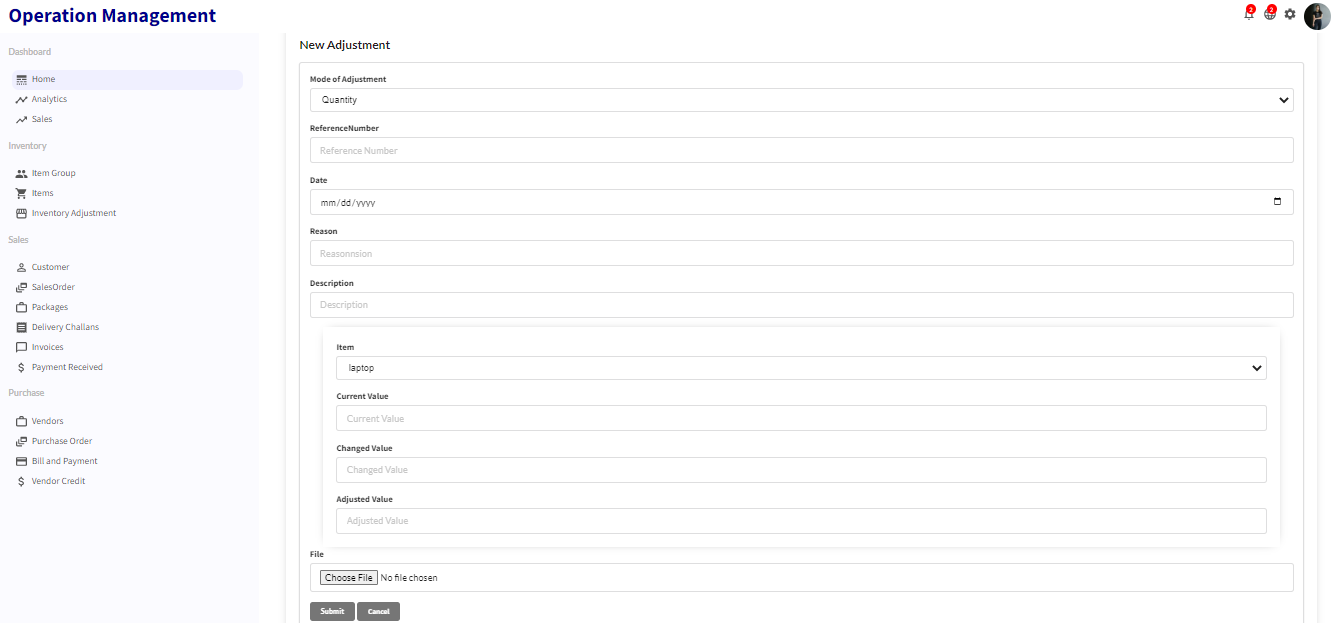
**Inventory Adjustment List**

It list all created Inventory Adjustments. User can delete a specific adjustment on clicking the delete icon for the specific row.



**New Inventory Adjustment**

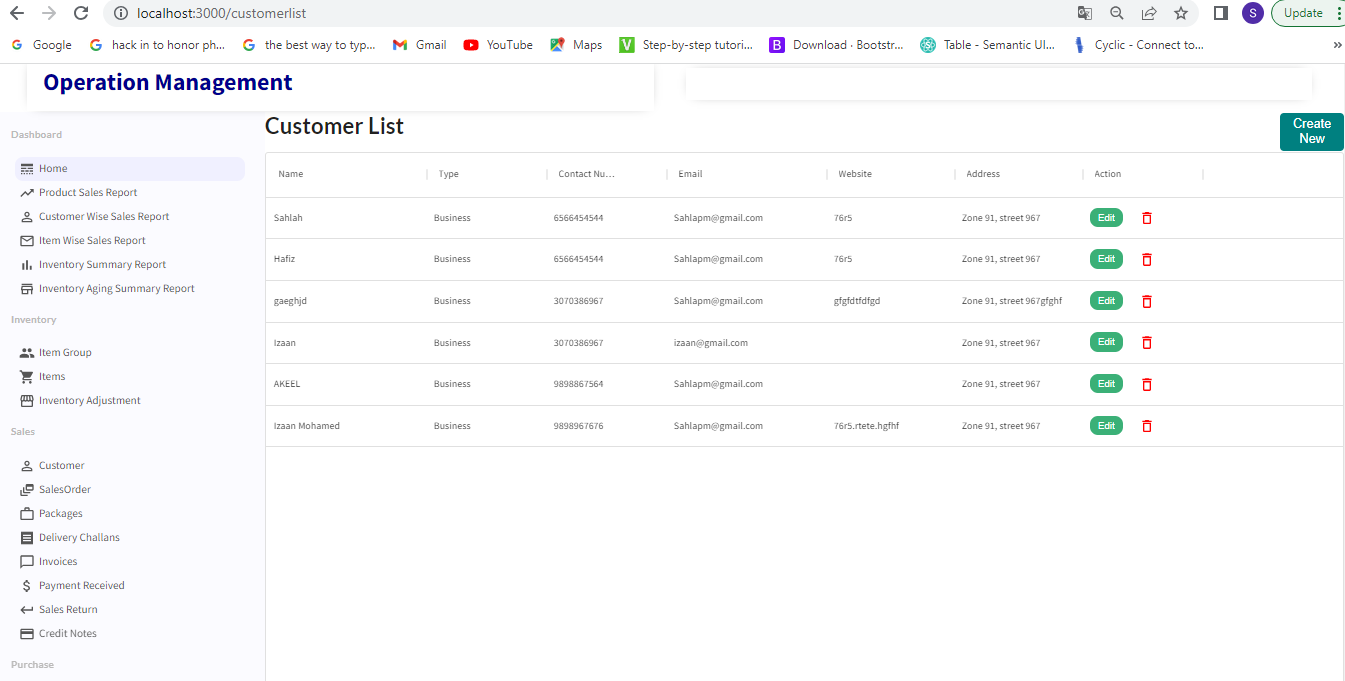
User can create new Inventory Adjustments from this page



**Action Term2-Sales Module**

**Customer List**

It list all created Customer. User can delete a specific customer on clicking the delete icon for the specific row and can edit the row by clicking in edit button.

****

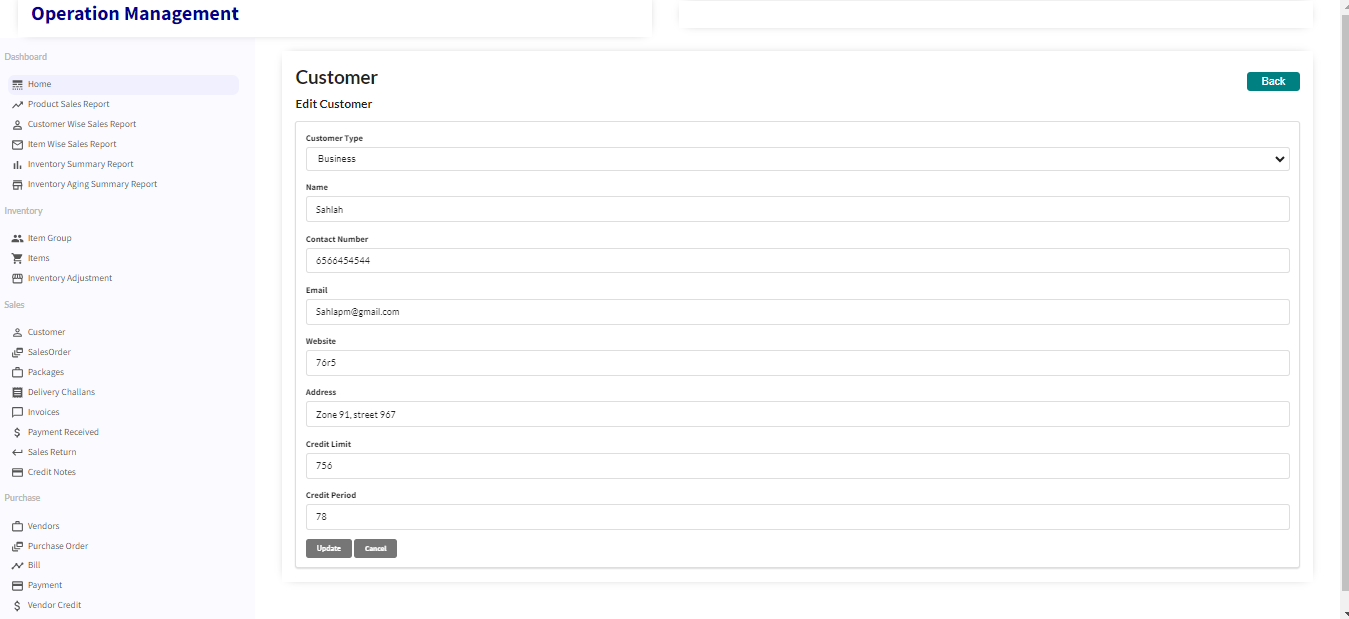
**New Customer**

This form is used to enter details of sales customers. Sales customers can be individuals as well as institutions.

****

**Customer Edit**

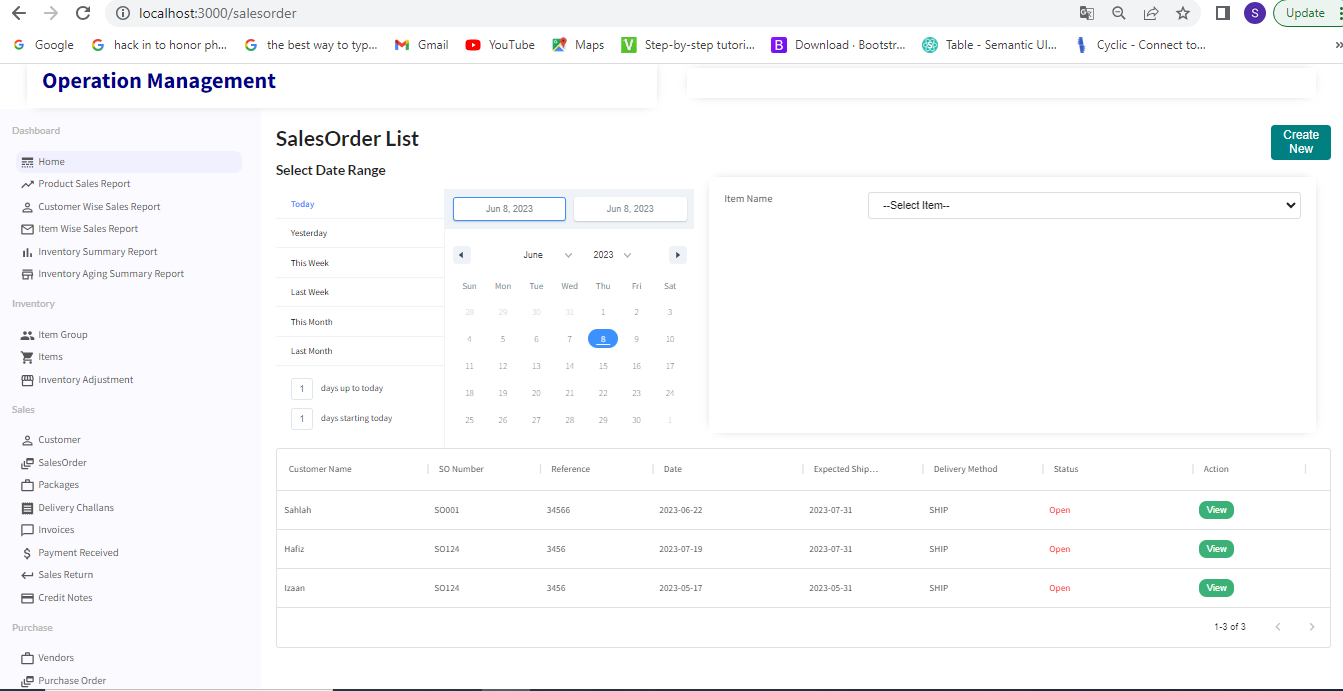
For editing selected customer details

****

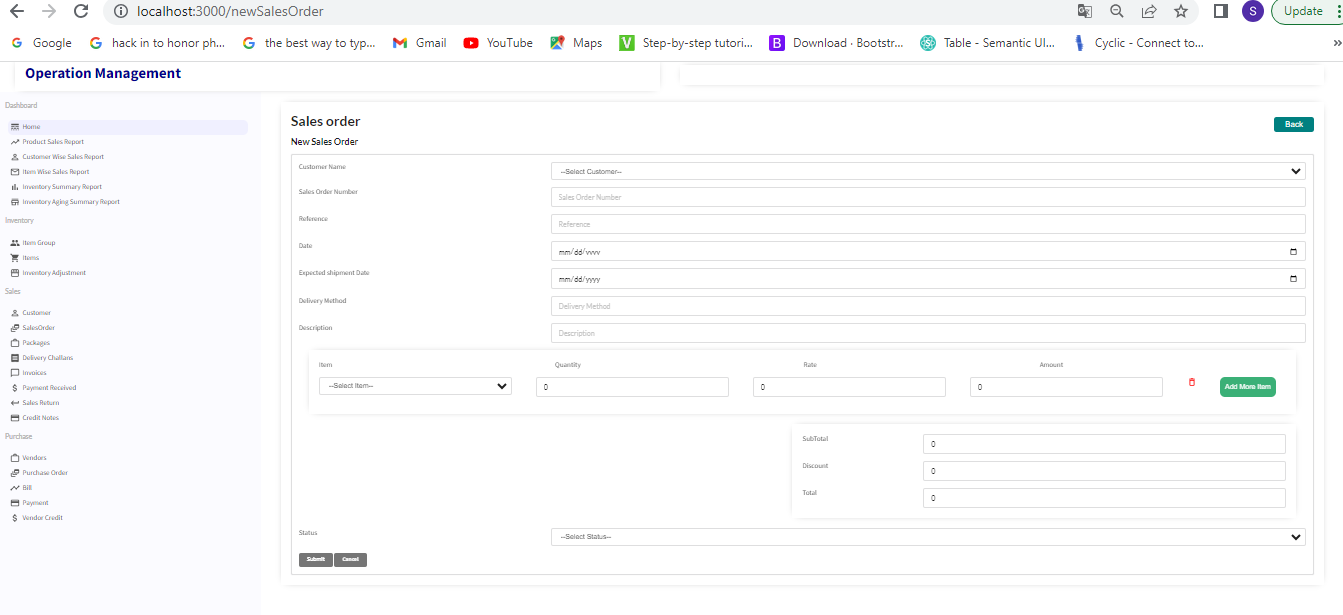
**Sales Order**

A sales order is a commercial document -- prepared by a seller and issued to a customer -- confirming the sale of goods or services involved in a given transaction. The document contains details about the sale, including the quantity, quality, and price of any goods or services exchanged

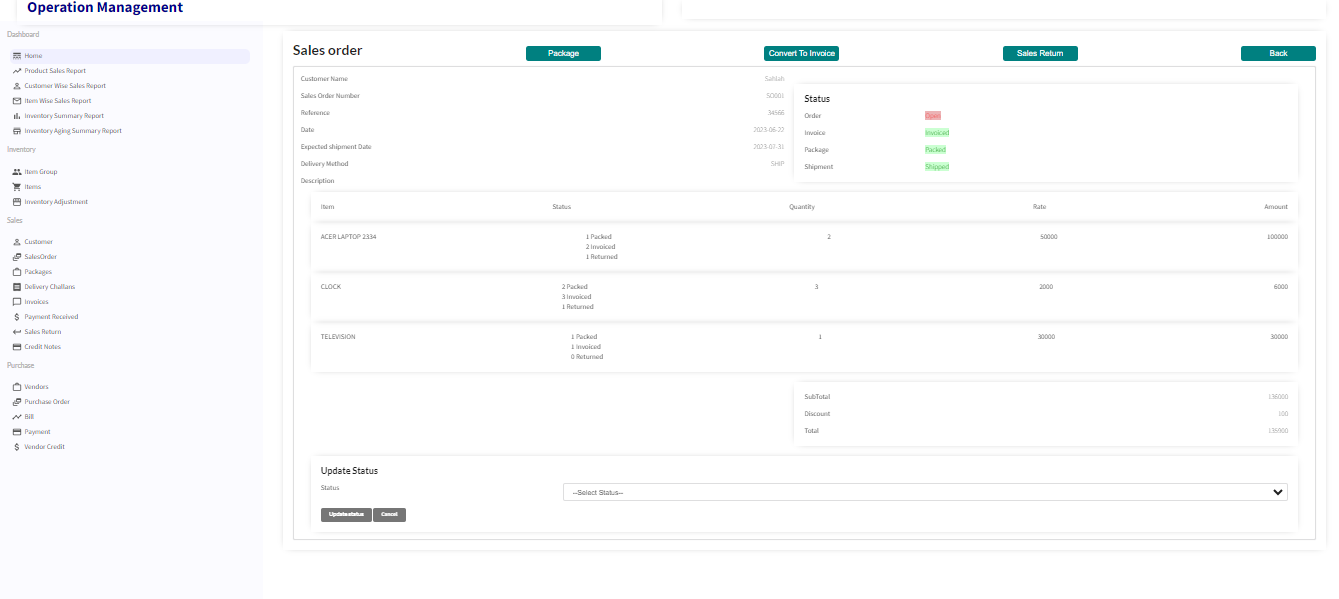
**Sales Order List**



**New Sales Order**

****

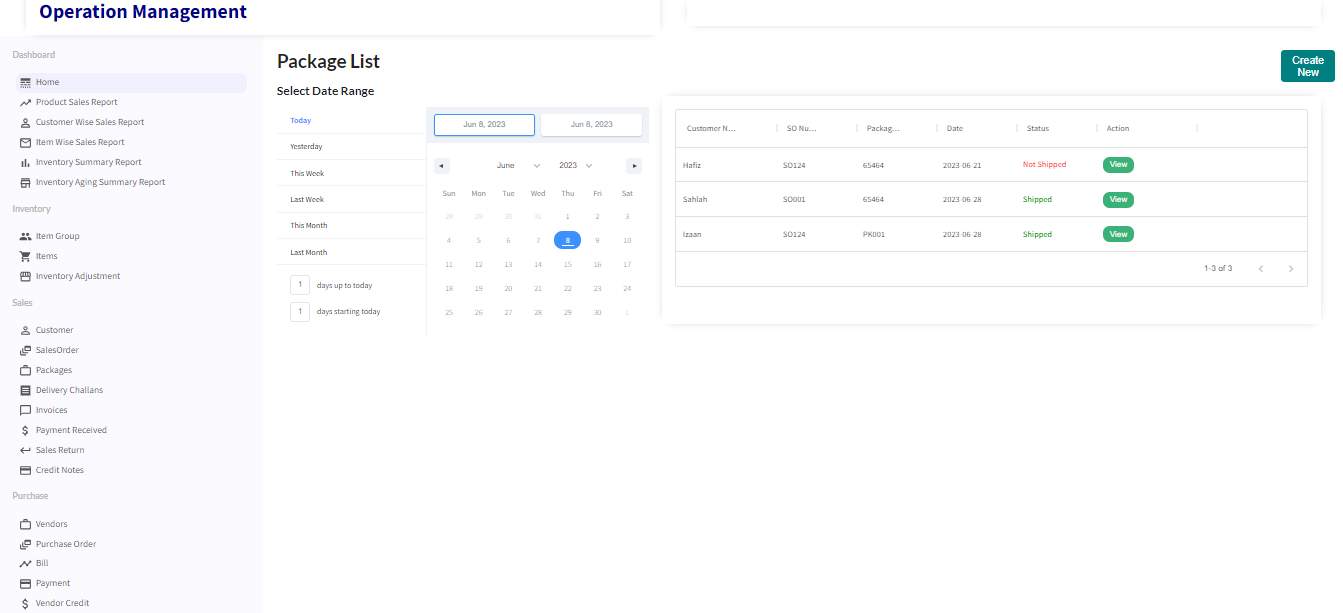
**Sales Order View**

****

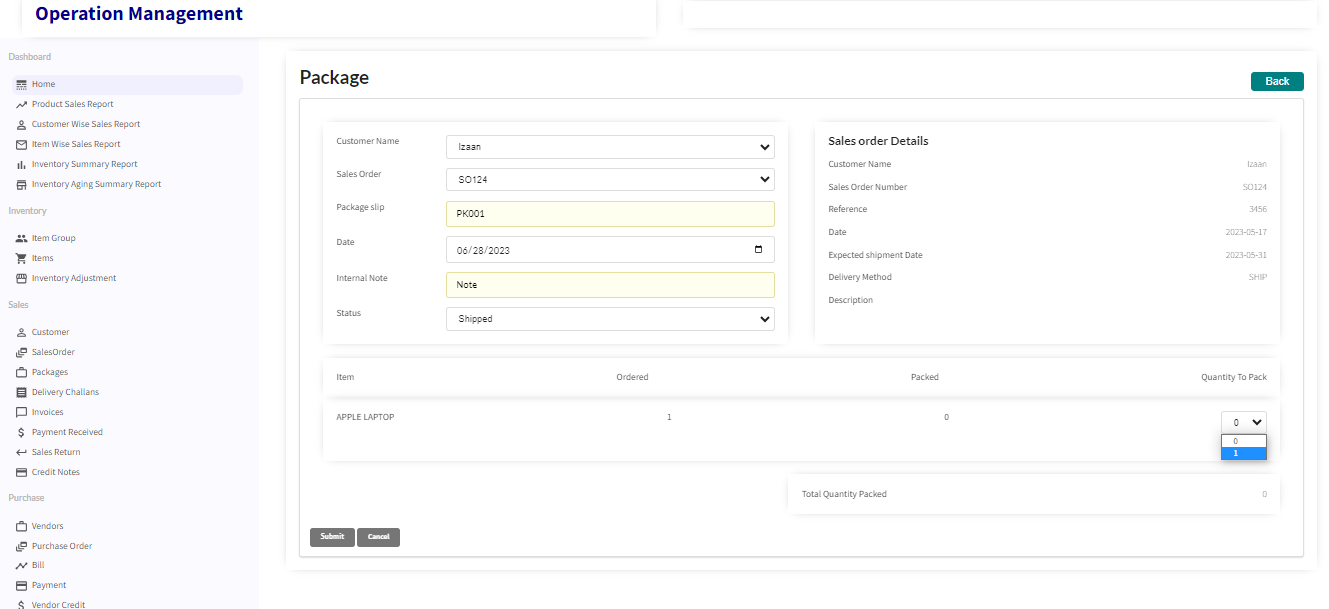
**Packages**

The Packages module allows you to create and assign package slip numbers to the items in your sales orders and keep track of your consignments when they are shipped

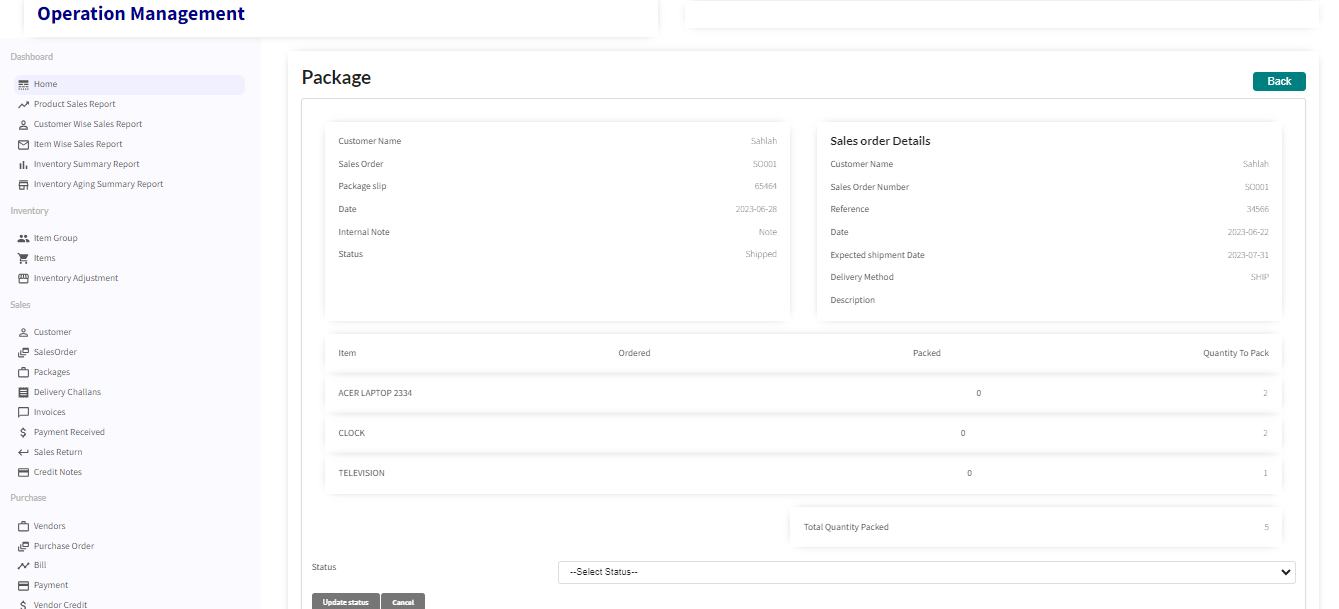
**Package List**

****

**New Package**

****

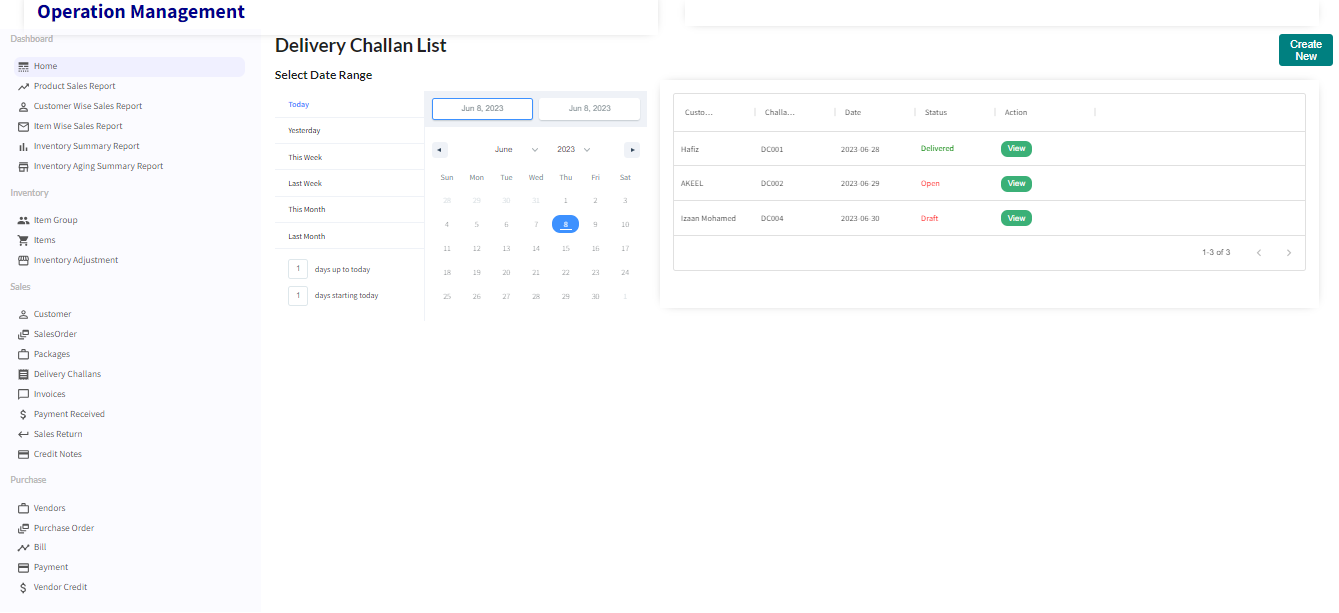
**Package View**

****

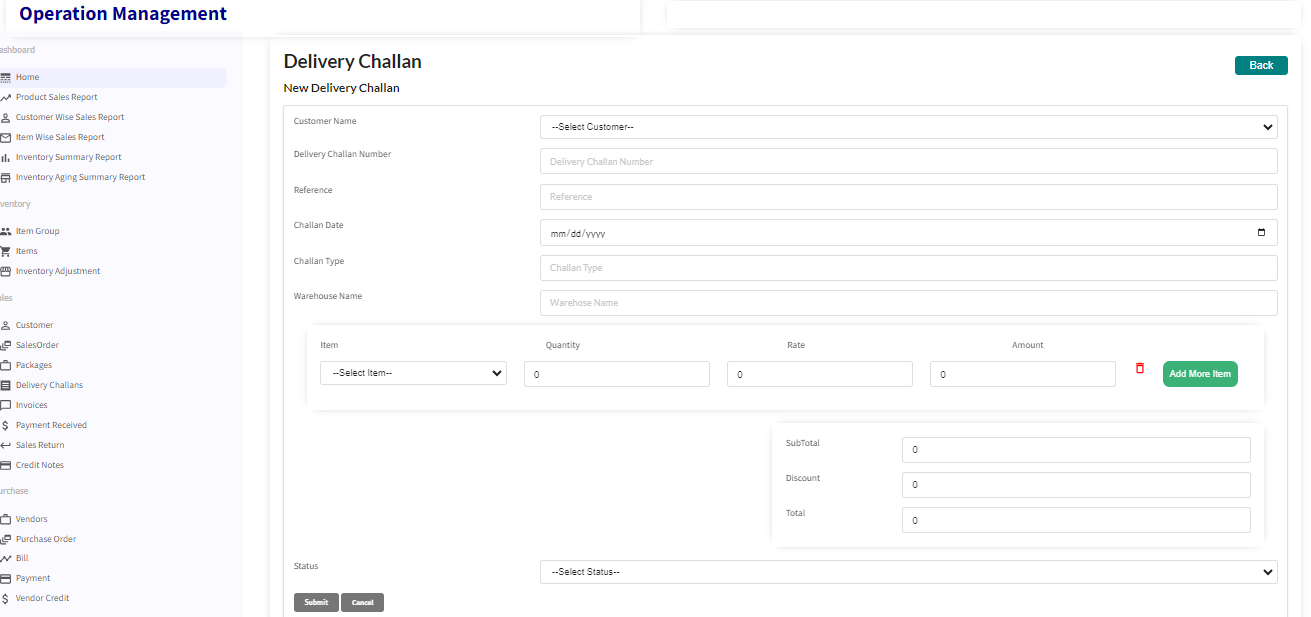
**Delivery Challan**

A **Delivery Challan** is created in situations where goods are being transported from one place to another which may or may not result in sales. Example: Transfer of goods from the Head Office to its branches. Delivery Challans are sent along with the shipment of goods. It contains the details of items and quantity of goods being delivered.

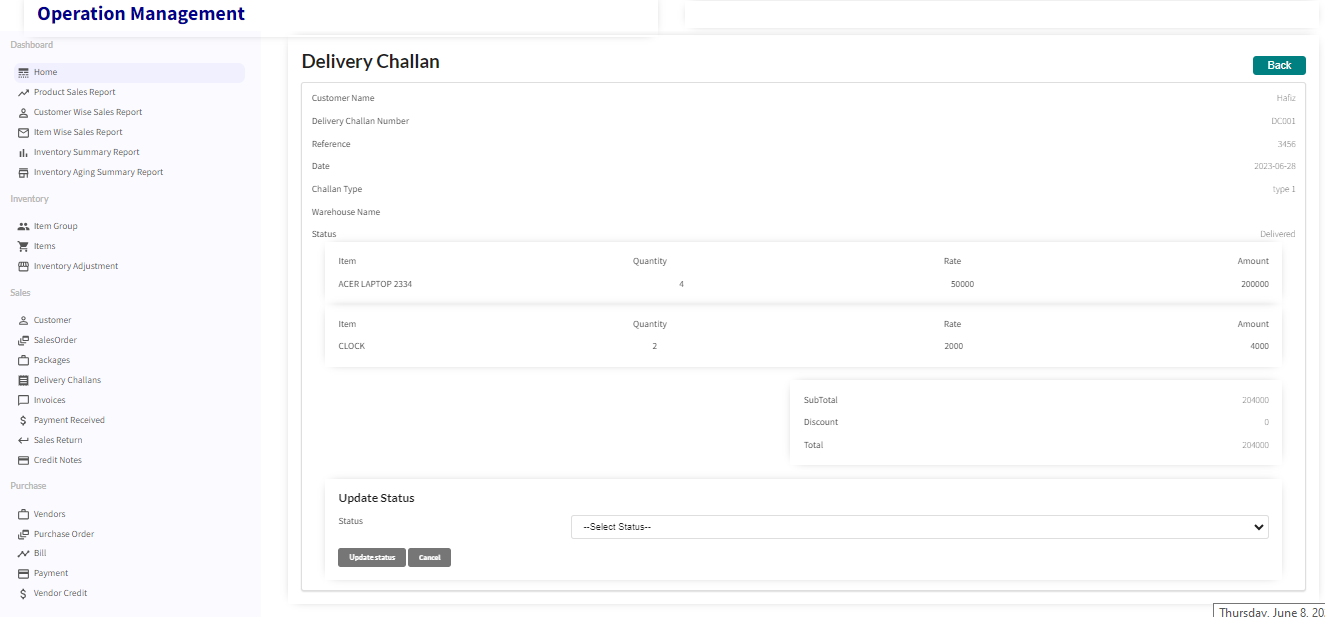
**Delivery Challan List**

****

**Create Delivery Challan**

****

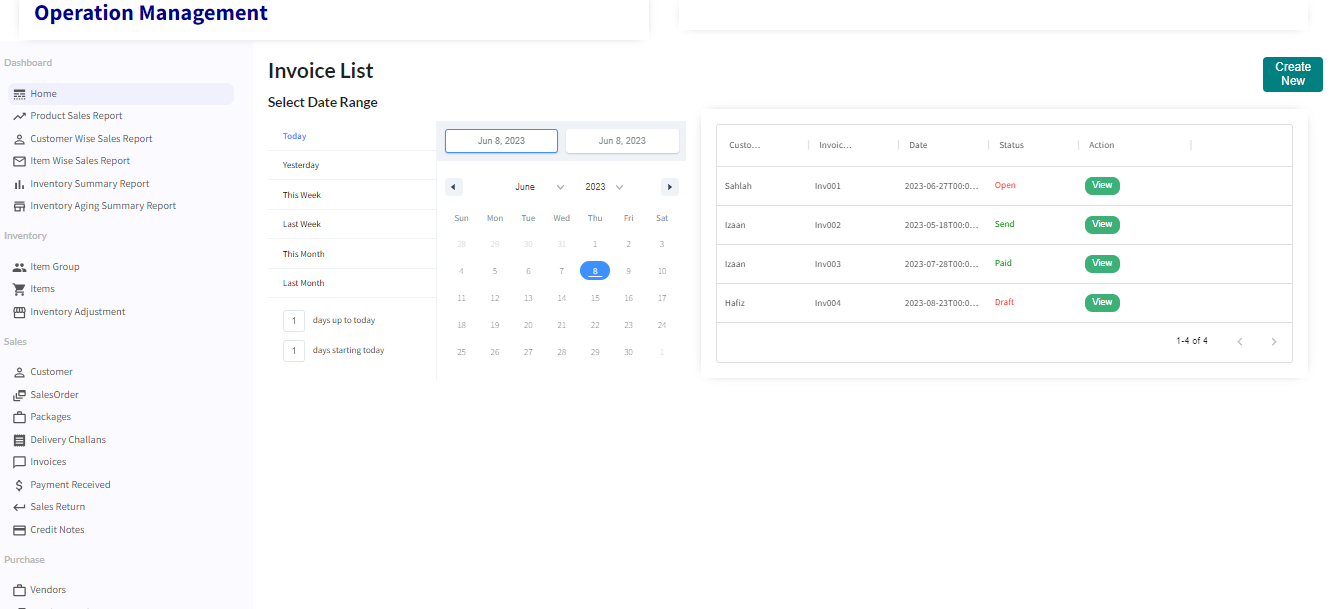
**Delivery Challan View**

****

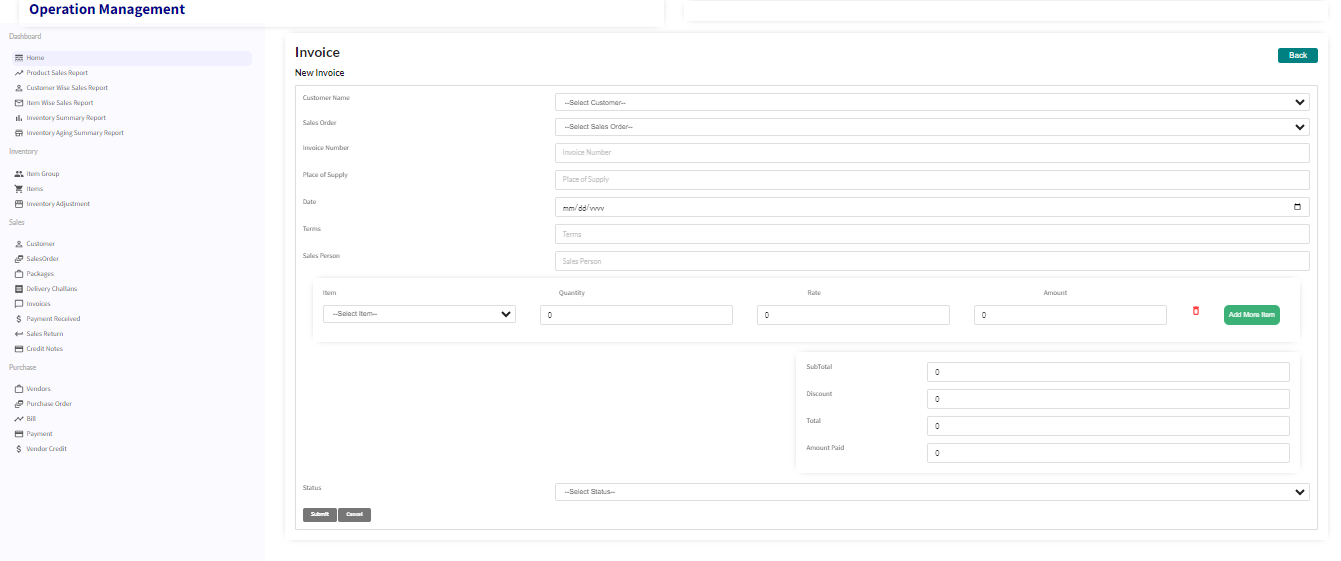
**Invoice**

An invoice is a commercial document issued by a seller to a buyer relating to a sale transaction and indicating the products, quantities, and agreed-upon prices for products or services the seller had provided the buyer.

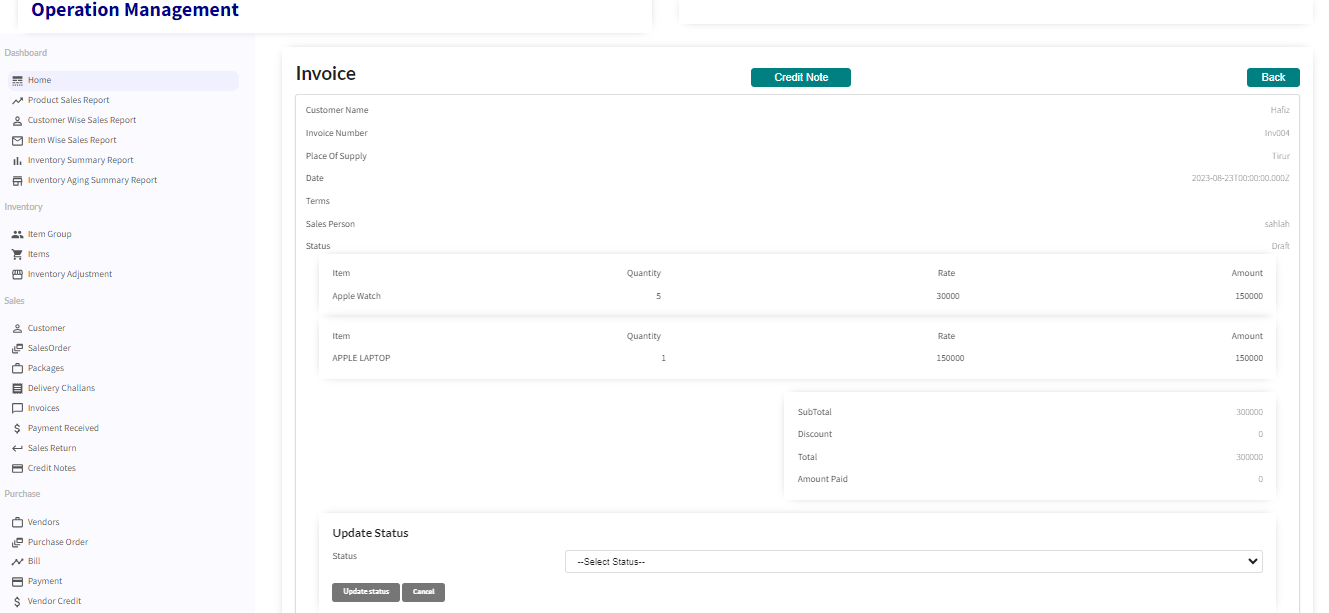
**Invoice List**

****

**Create Invoice**

****

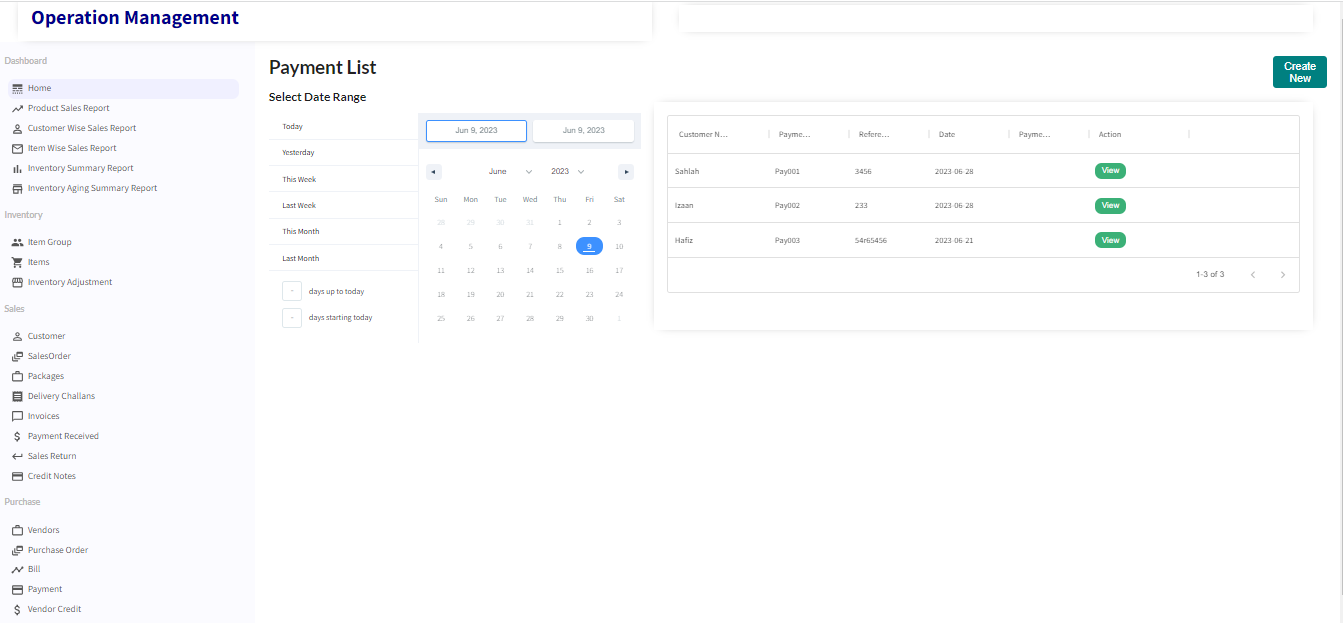
**Invoice View**

****

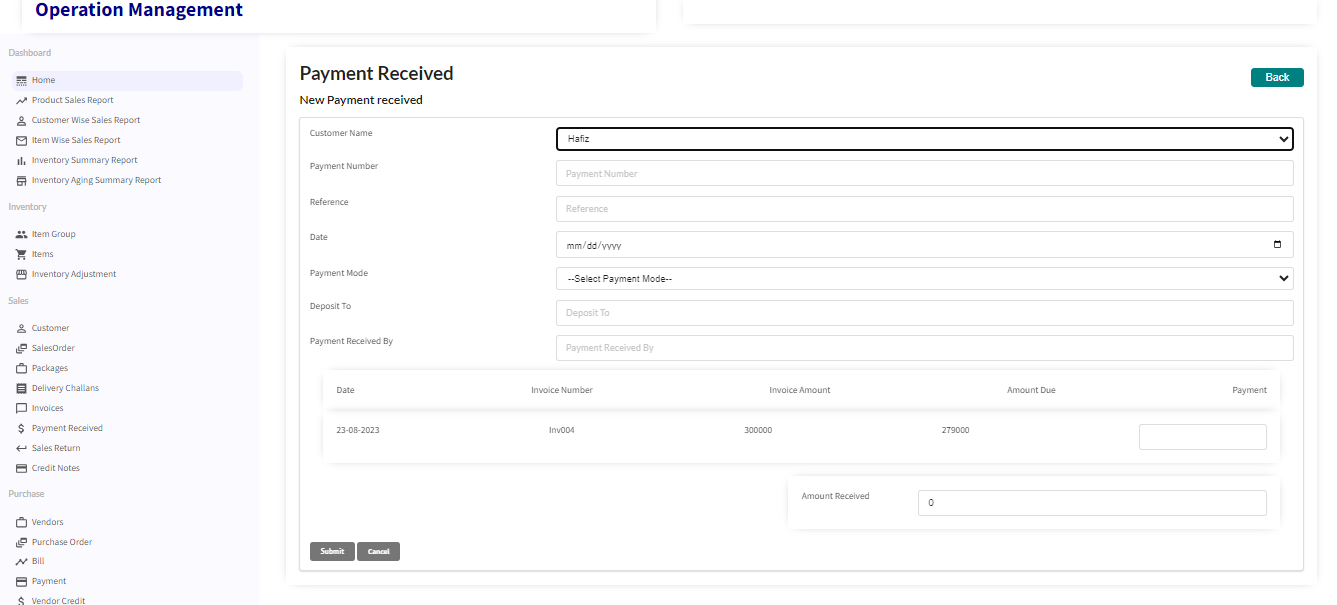
**Payment Received**

This form is used to record receipts from customers for invoice. The invoice can be selected from the picklist and the details like amount is automatically filled. Mode of payment includes by cash, cheque, debit/credit card and digital payment

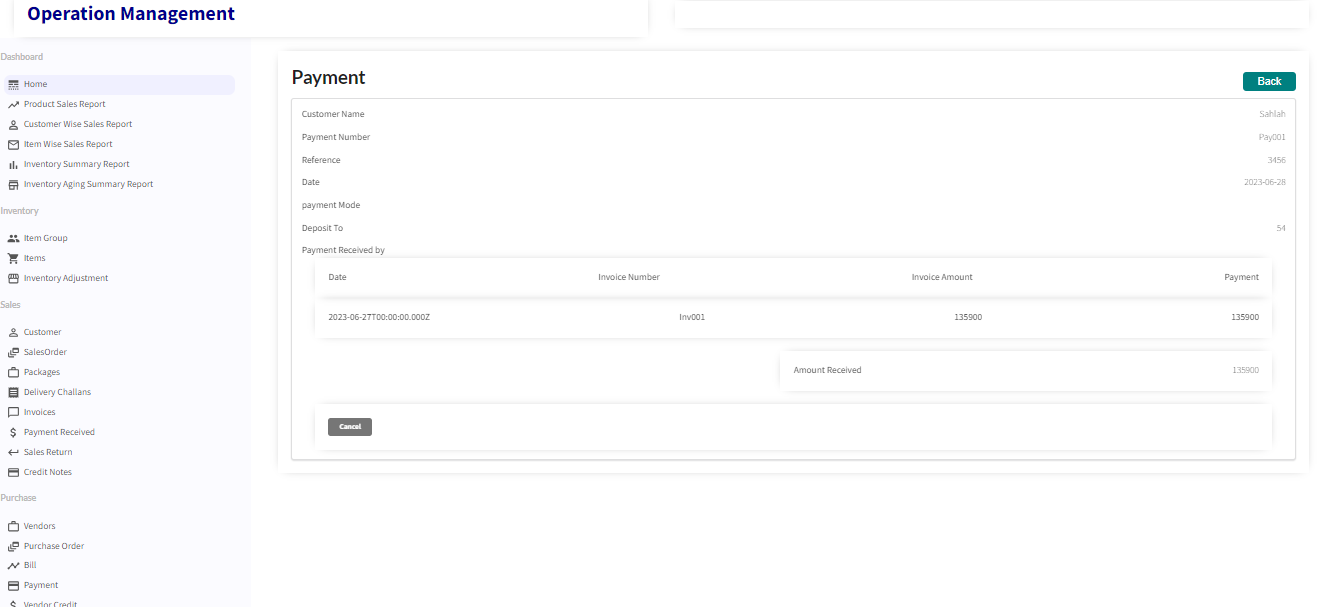
**Payment Received List**

****

**Create Payment Received**

****

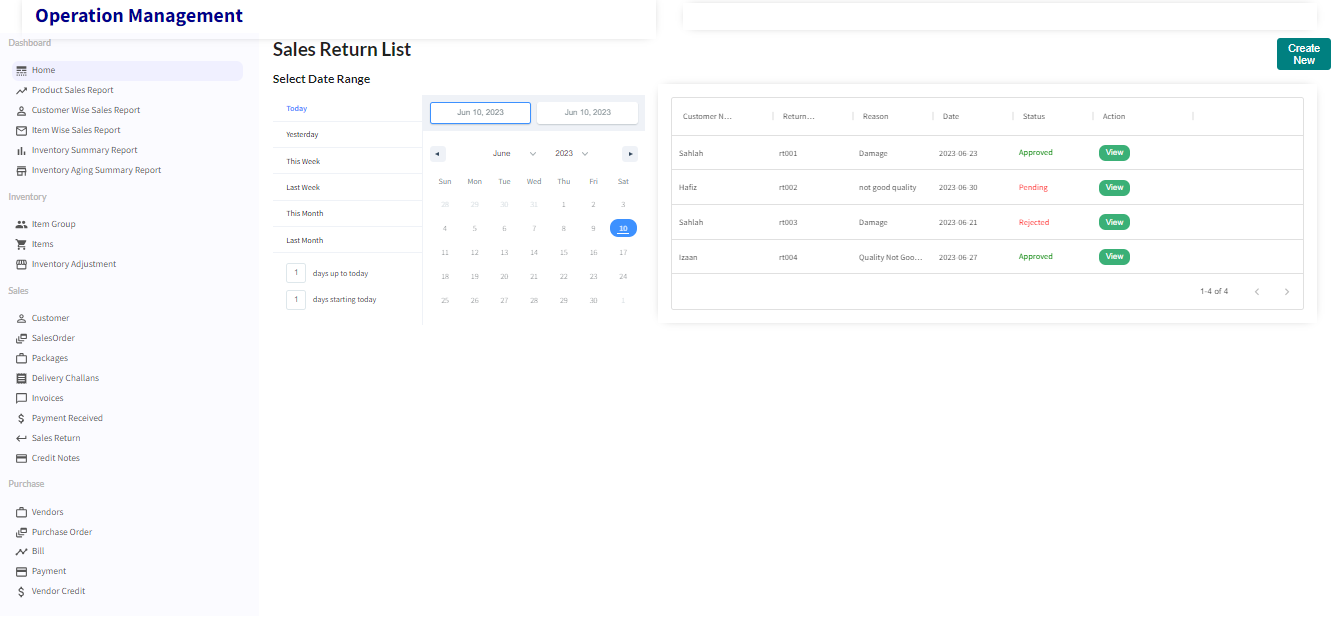
**Payment Received View**

****

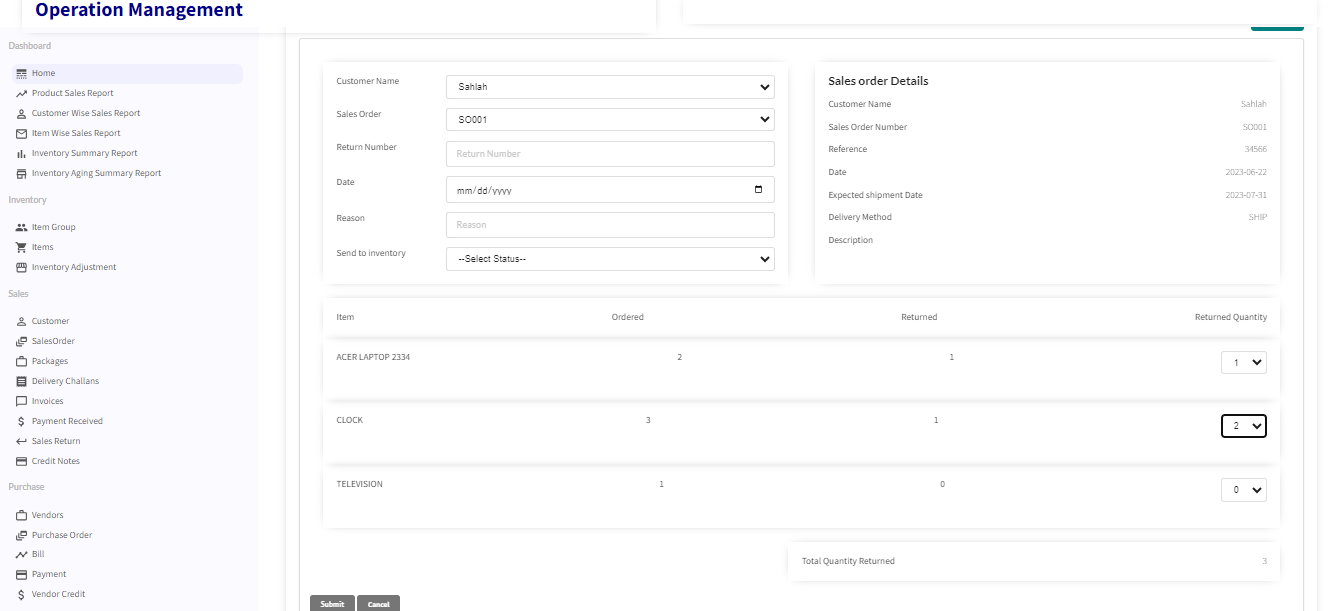
**Sales Return**

A sales return is merchandise sent back by a buyer to the seller. The return is usually because an excess quantity was either ordered or shipped, or due to defective goods.

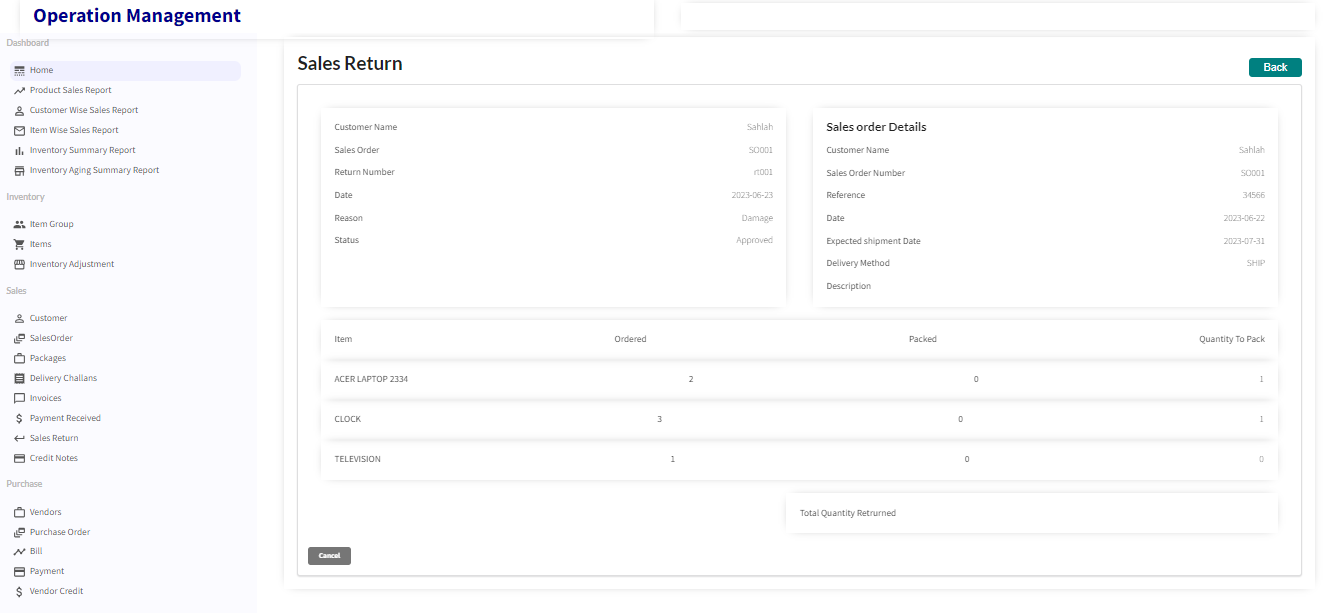
**Sales Return List**

****

**Create Sales Return**

****

**Sales Return View**

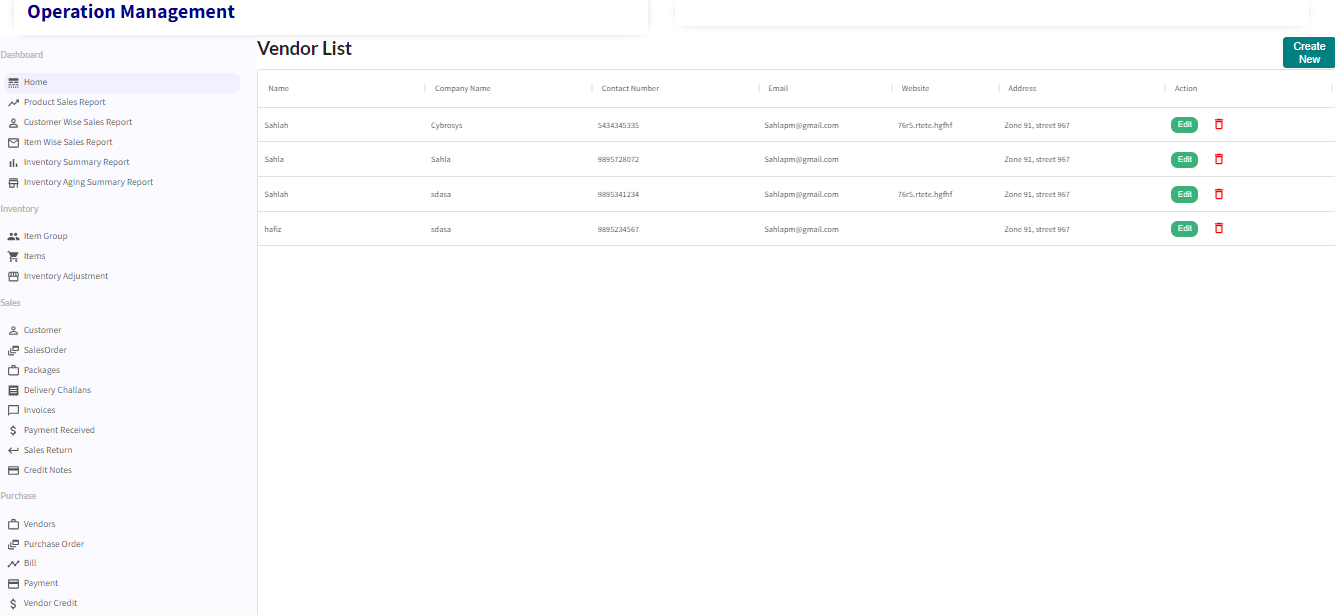
****

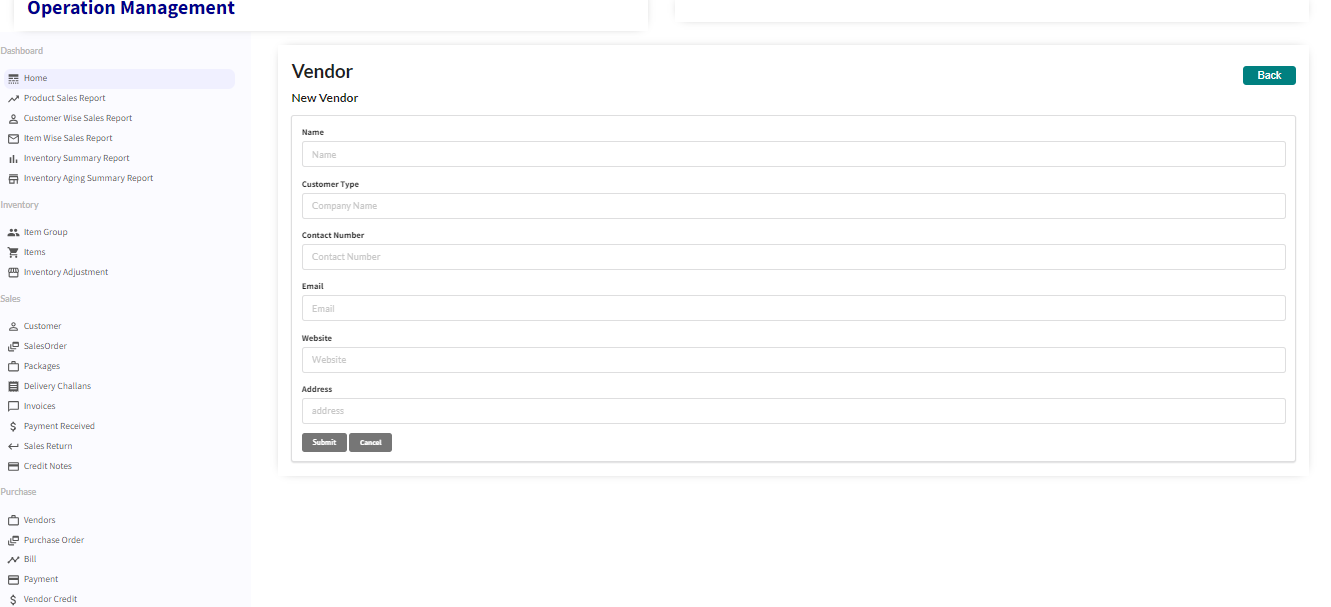
**Action Term3-Purchase Module**

**Vendor**

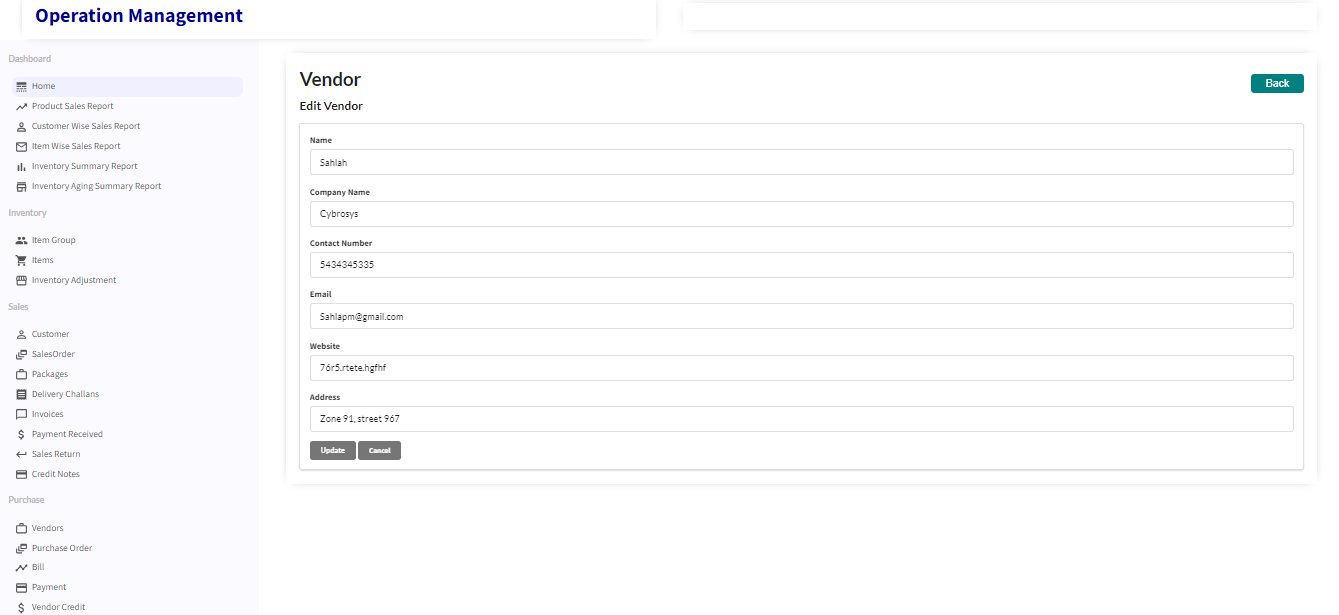
Purchases of goods/services are made from vendors. The vendor form is used to enter the details about the vendor which includes contact information, the goods and services they sell and about the point of contact.

**Vendor List**

****

**Create Vendor**

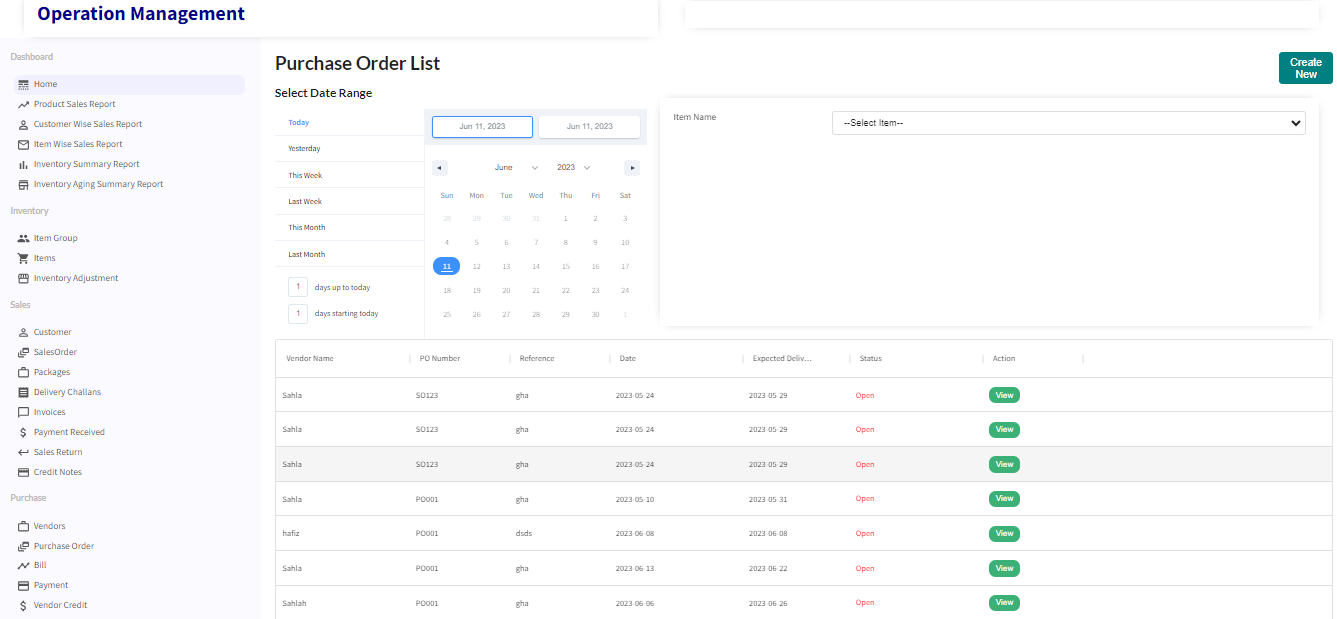
**Edit Vendor**

****

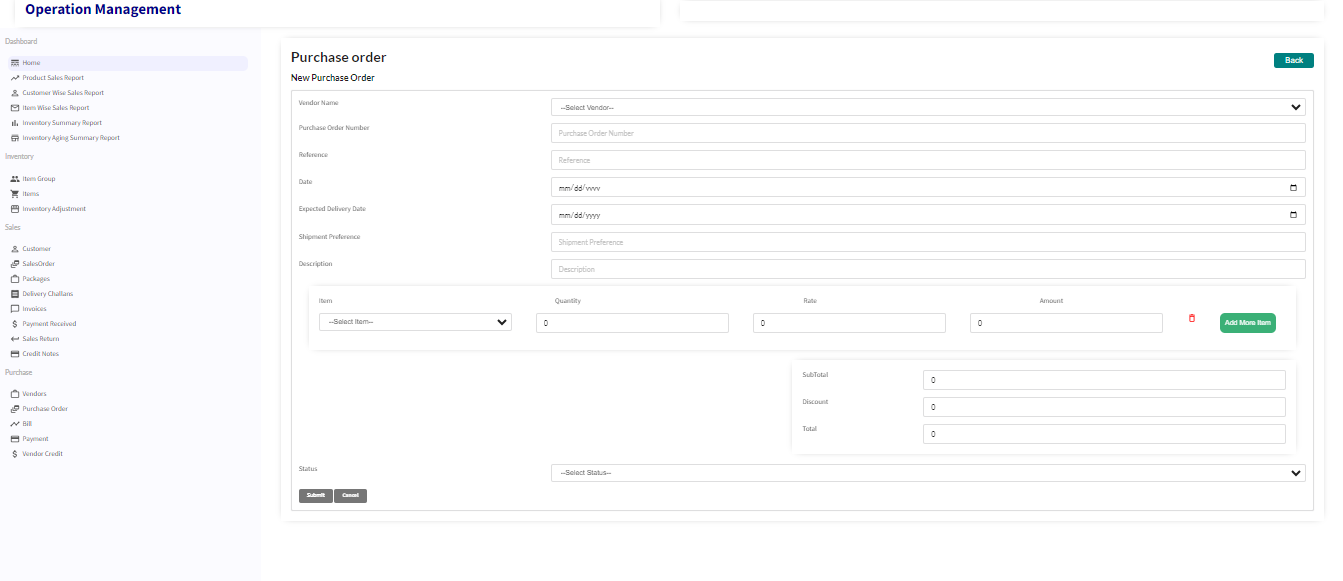
**Purchase Order**

A purchase order (PO) is a commercial document and first official offer issued by a buyer to a seller, indicating types, quantities, and agreed prices for products or services. It is used to control the purchasing of products and services from external suppliers

**Purchase Order List**

****

**Create Purchase Order**

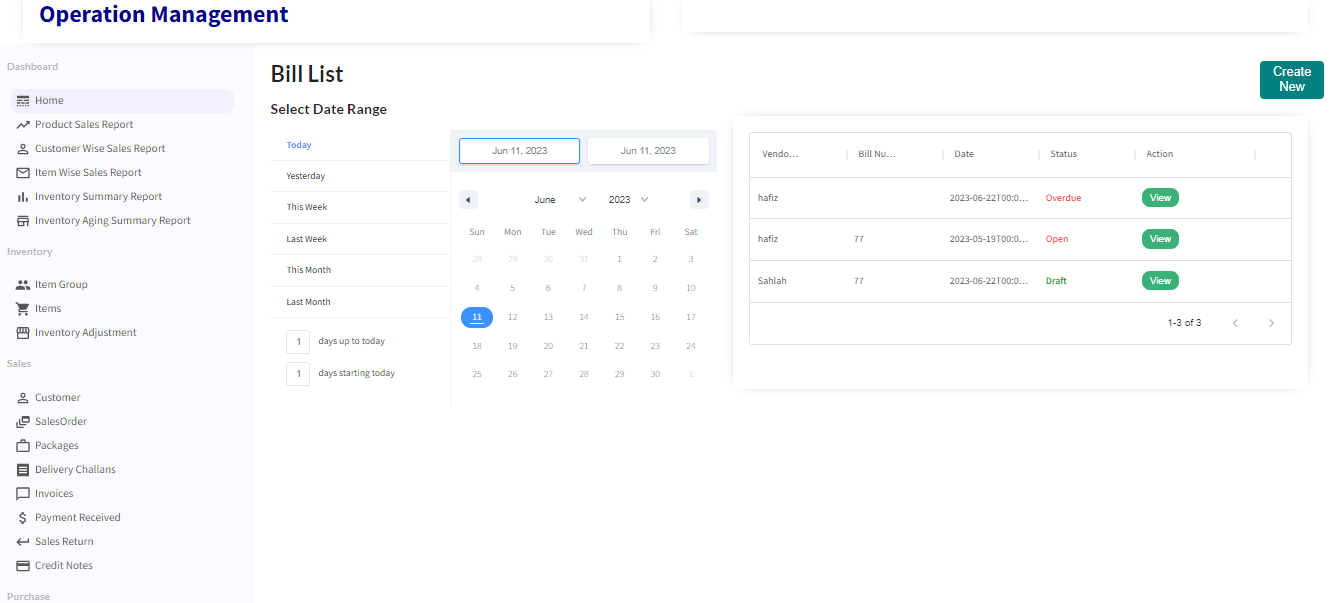
****

**Purchase Order View**

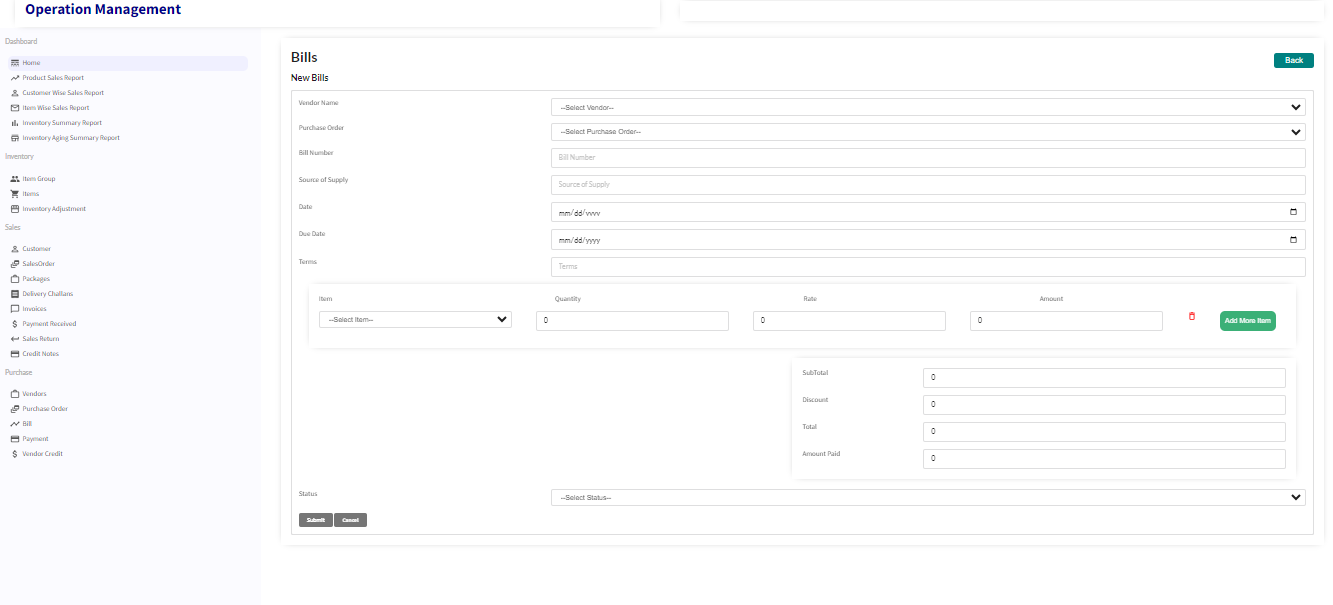
**Bills**

Vendor bills are requests for payment for products and services. Vendor bills might represent a bill for ongoing services, or they can be based on purchase orders for specific items and services. Bill details can be entered for a specific vendor by selecting vendor or for a purchase order by selecting the purchase order

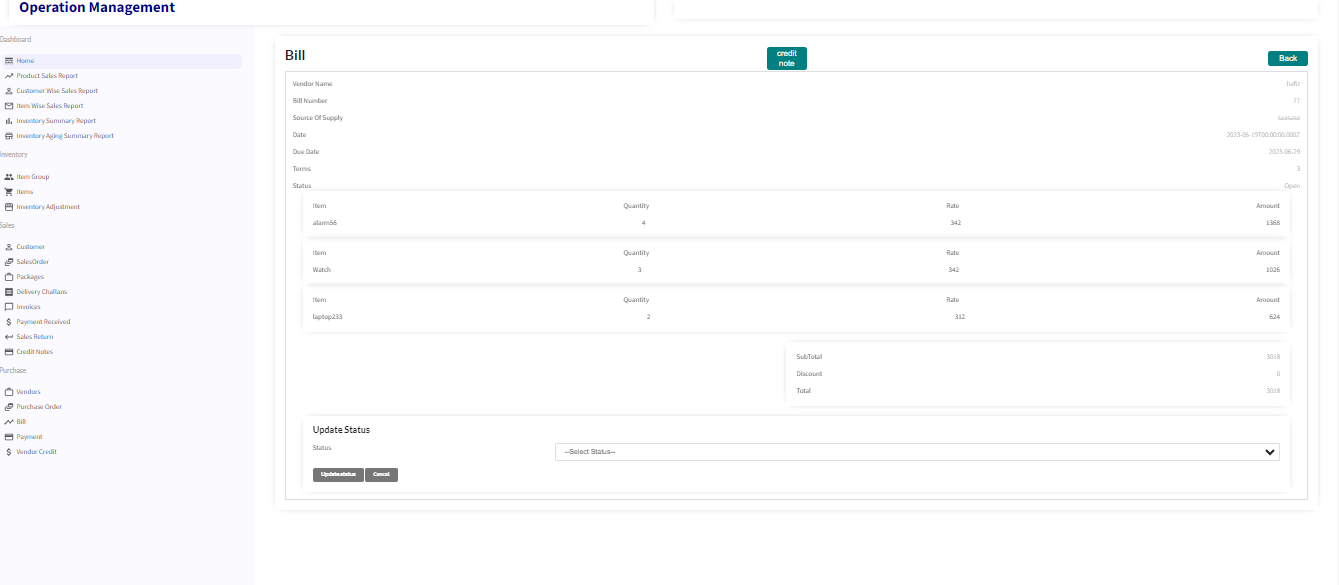
**Bill List**

****

**Create Bill**

****

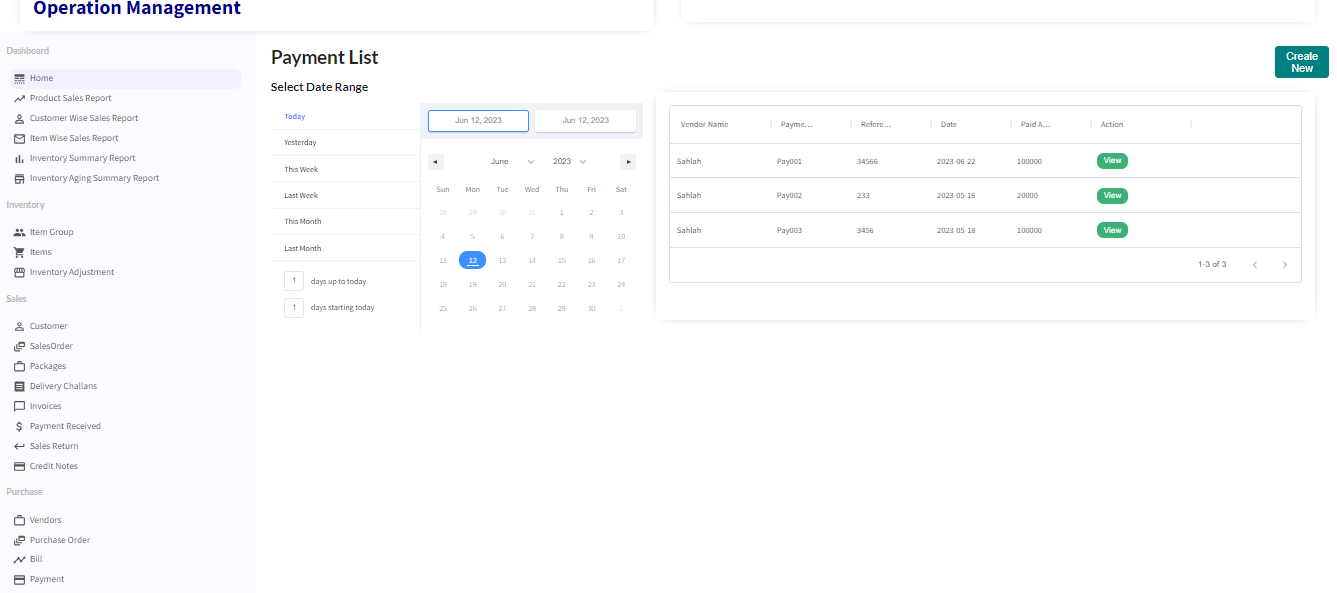
**View Bill**

****

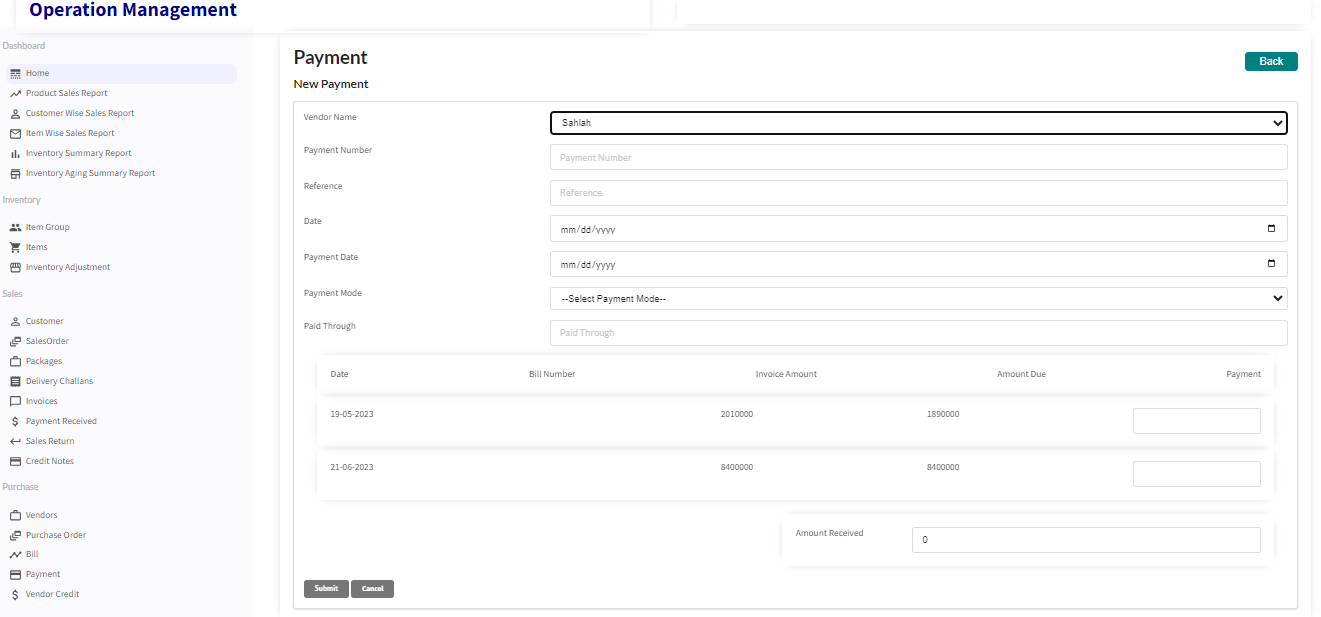
**Payment**

A bill payment is the last step in the purchase to pay cycle, when a company pays an outside vendor for purchased goods or services. The payment can be recorded in this data entry form. The bill has to be selected and amount will be automatically filled.

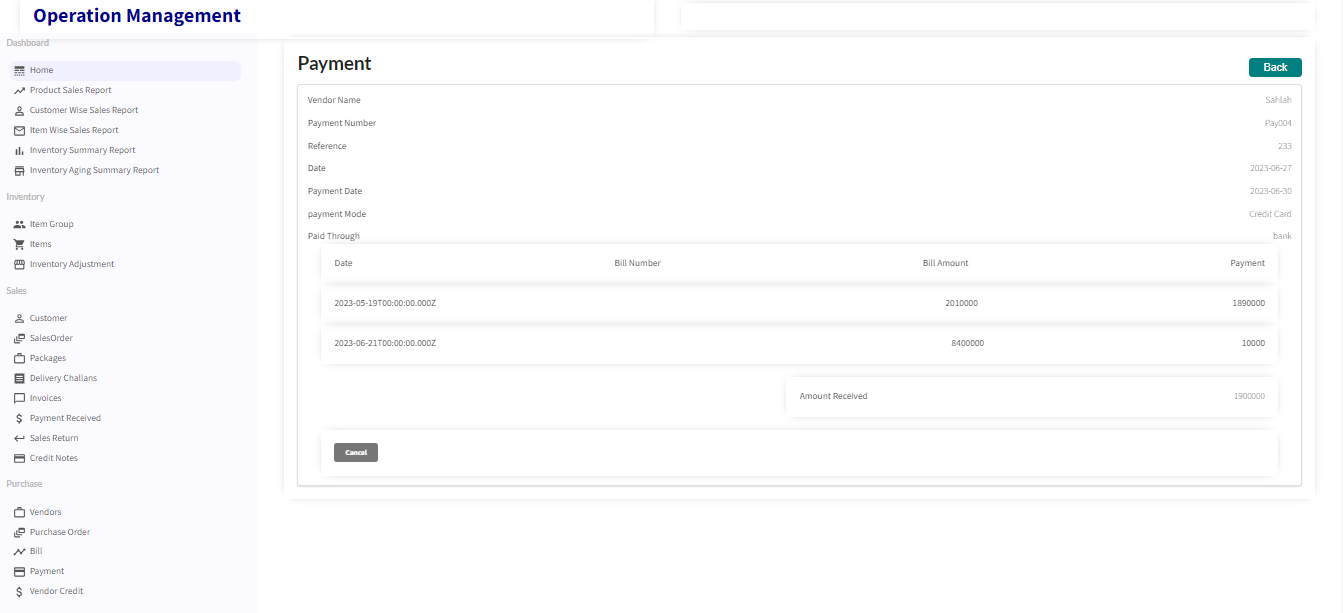
**Payment List**

****

**Create Payment**

****

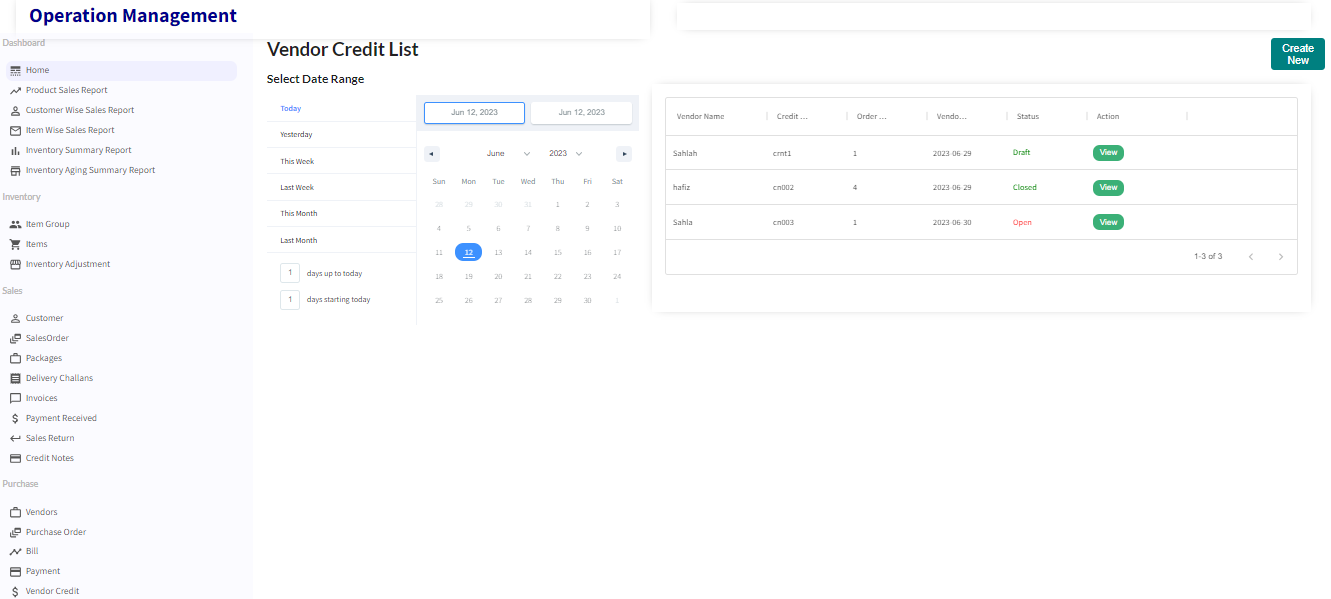
**Payment View**

****

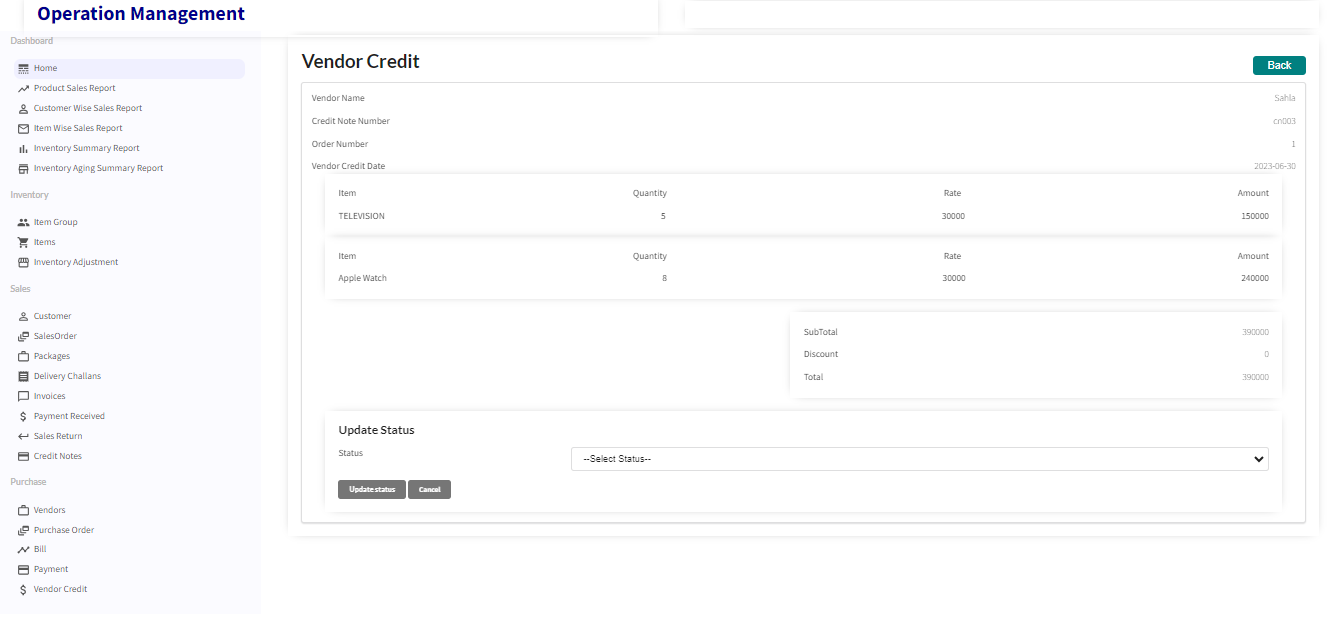
**Vendor Credit**

Vendor Credits are credits that you receive from your vendor, and is treated as an equivalent of physical cash that the **vendor owes you**. This helps you track the money you’re owed until it is either paid by said vendor at a later date i.e. refunded, or subtracted from any future bill amount due to that vendor.

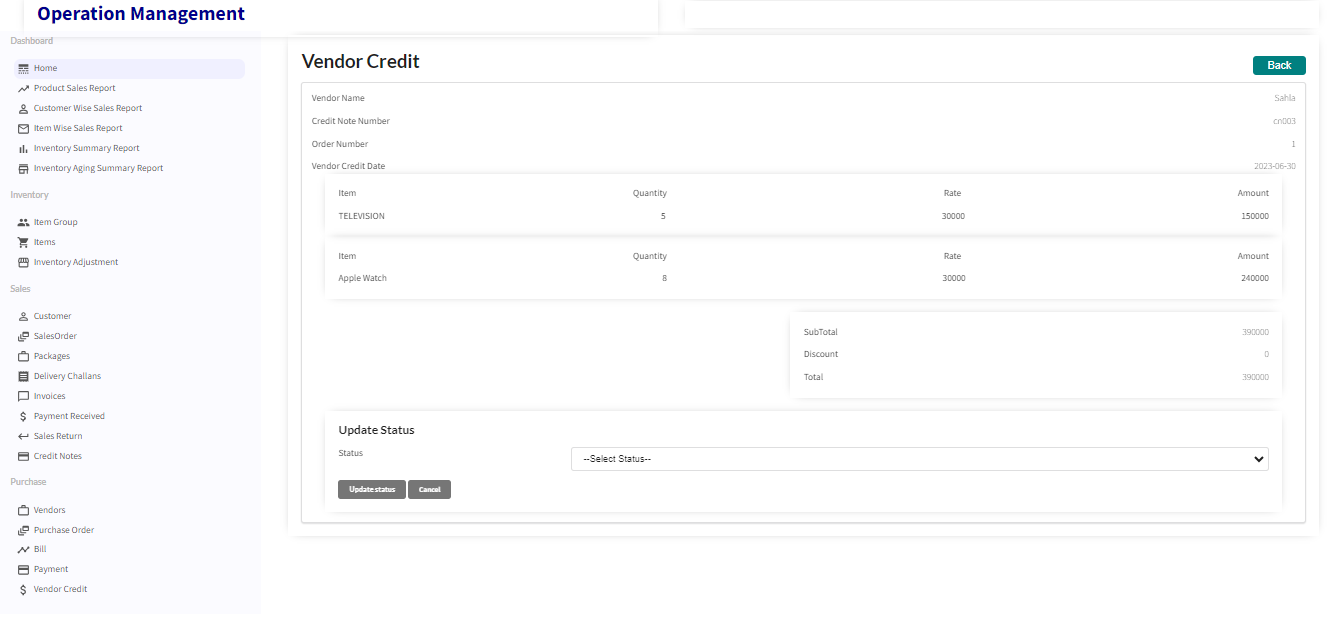
**Vendor Credit List**

****

**Create Vendor Credit**

****

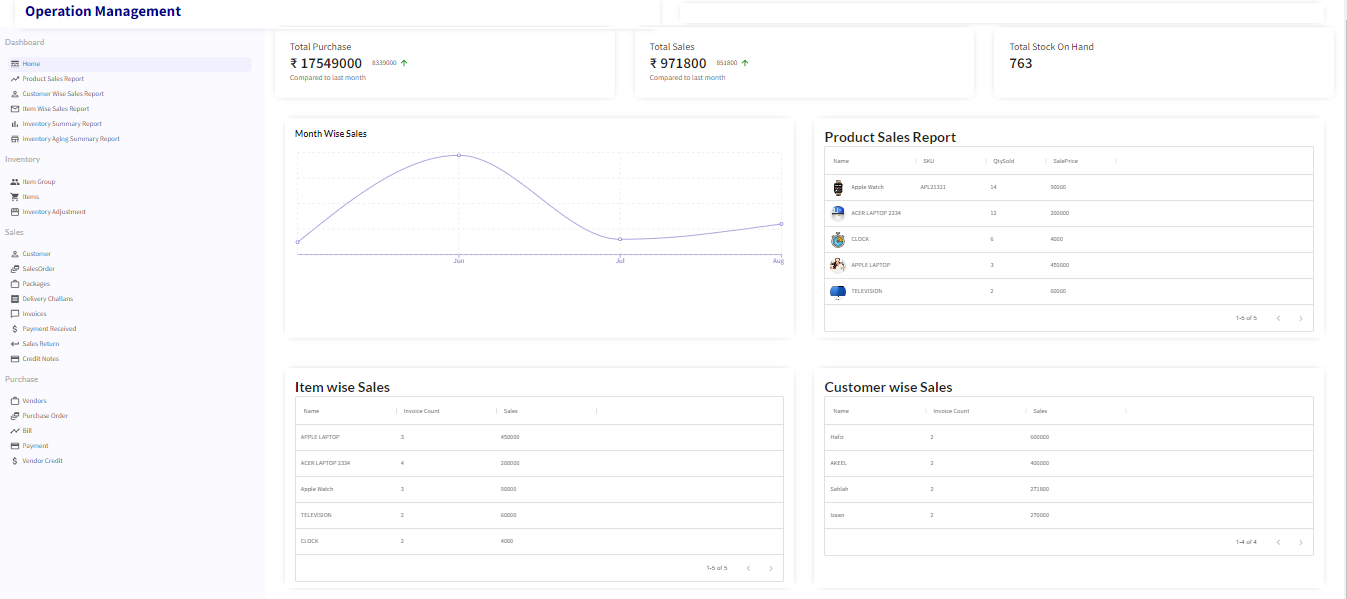
**Vendor Credit View**

****

**Action Term4- Dash board and Reports**

**Dash Board**

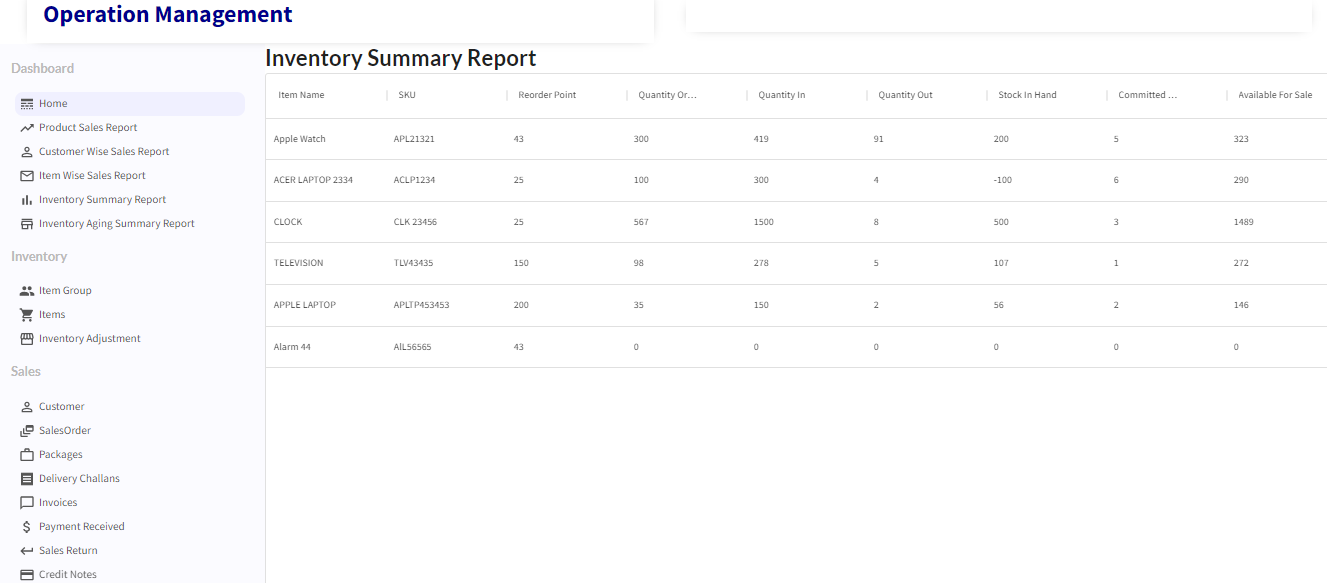
This is the Landing Page of this application. This Dash board will give us a quick view of the entire system. It displays total Monthly Sales, Monthly Purchase, Total Stock In Hand, Month wise sale graph, Product sale report, and item and customer wise reports.



**Reports**

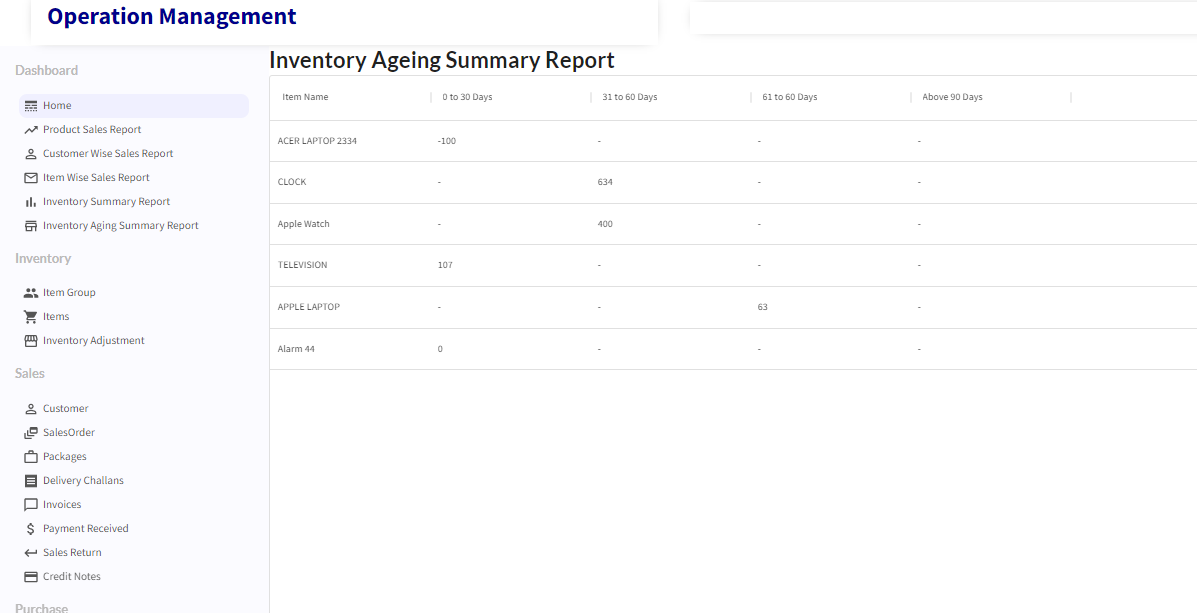
**Inventory Summary Report**

This report is a summary of the movement of items in and out of your organization and their quantities. This report contains the fields like Item Name, SKU, Reorder Level, Quantity Ordered (Purchase Order Quantity in Open and Close Status), Quantity In (Bills in Open, Overdue and Paid Status + Credit Note Open and Close Status), Quantity Out (Invoices in sent, overdue and paid + Vendor credit open and closed), Stock In Hand (Items in Open Bills - Items in Open Invoices), committed stock (Sales order in Open and close status) and Available for sale(Quantity in - Quantity out - Committed stock)

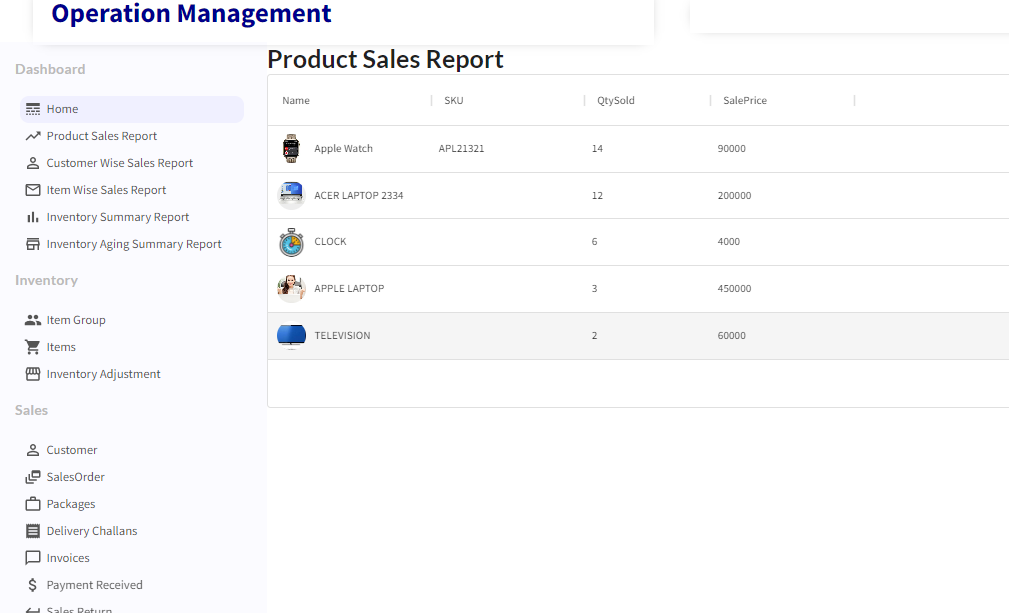
****

**Inventory Aging Summary Report**

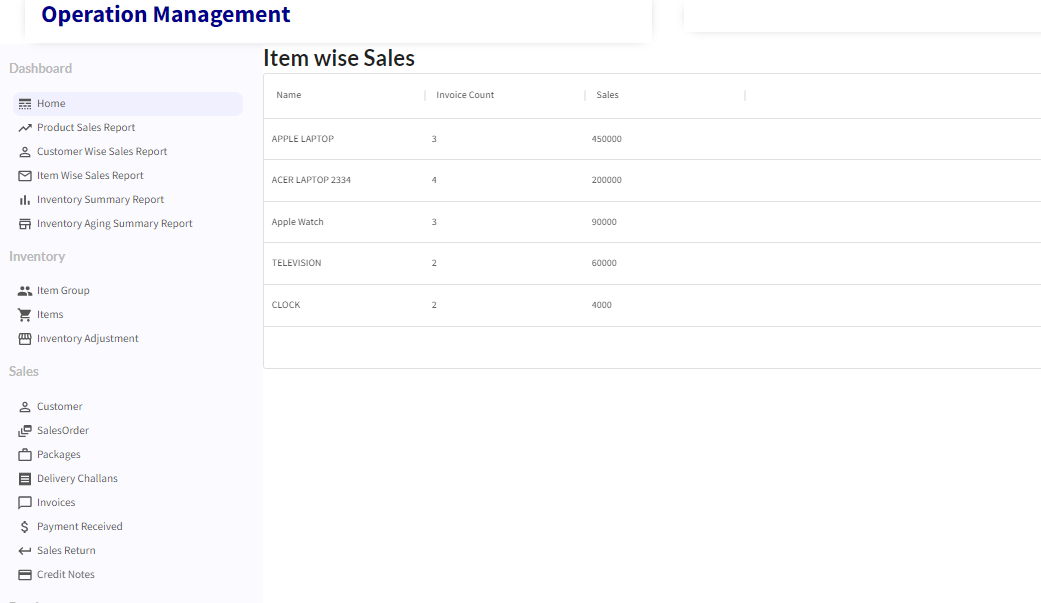
An inventory aging report, also called an aged inventory report or stock aging report, provides key metrics on how quickly your inventory moves. It's essentially a list of the items on hand grouped by the length of time in inventory. In this first stock on hand is calculated and then stock in (based on purchase bills) on each day is calculated starting backwards from today and is subtracted from the stock on hand till its value becomes zero.

****

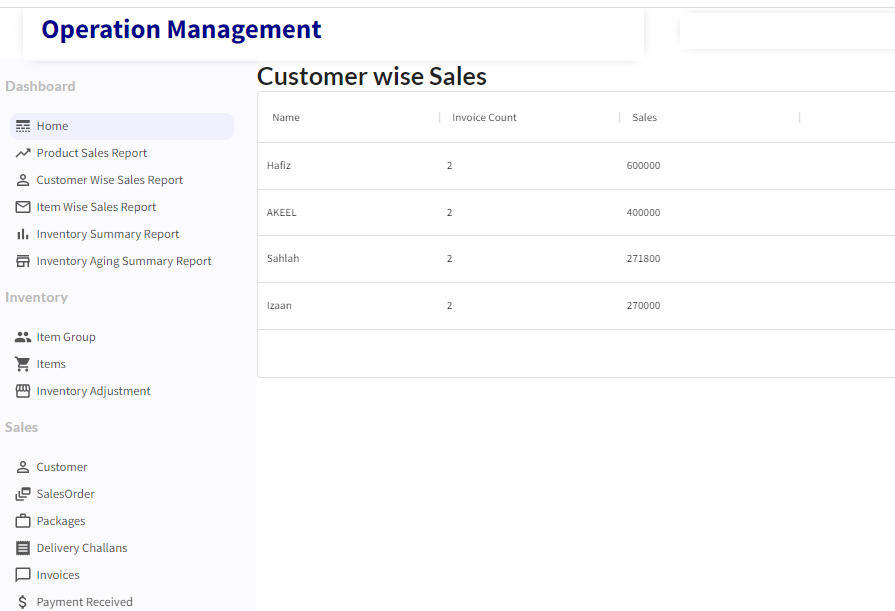
**Product Sale Report**

****

**Sales by Item Report**

****

**Sales by Customer Report**

****

**Reflections on the Internship**

On Finishing this Internship I got a good experience of using the MERN stack for doing industry level projects. The project *inventory module* helped me learn many new things and to hone my programming skills in JavaScript. I was also able to learn writing many types of mongo db queries for getting the required data as per the project requirements.

Even though the starting was a bit difficult, later I could able to write code for getting the expected output. I searched many sites to learn the techniques needed to satisfy the requirements. This project task was challenging and I enjoyed doing it. I would say this internship was a worthwhile experience.

TCS ion has provided a great platform for doing internships. They have provided many helpful references as well as guidance in the form of industry mentors to make the internship a very fruitful one. Since activity reports have to be written daily, it helped me to be more systematic in my work. This internship experience has created a good work habit, to better schedule jobs and to get the work done on time.

The experience gained from this internship will be very useful for me in pursuing a job as a MERN full stack developer. I hope that I can advance my career further by using the inputs gained from doing this internship.

**Recommendation**

The experience got from this internship is very valuable and hope it can help me in advancing my career. But certain thinks I felt to improve from your side are.

* First thing is about the project description document. It would be good if the document explain more about the data flow and the domain
* I would have liked to have more interaction with the industrial mentors and effective discussion in digital discussion room

**Outcome/Conclusion**

The internship was on the topic to develop automated software for inventory management module of operations management using MERN stack. I have completed the application on time by creating a complete solution which satisfies all requirements. Also I included additional features to make the application more user friendly.

The experience gained from this internship is very valuable and hope it can help me in advancing my career

**Enhancement Scope**

* User Registration and Authentication
* User Authorization and setting different access rights for different categories of users
* Customization settings for each field
* Automatic code generation for each transaction with settings to set suffix and prefix for the code
* Barcode generation settings
* Multiple item saving in a single click using excel
* Report emailing facility.
* Customer registration and online ordering facility
* Payment Gateway Integration

**Link to Demo Video**

<https://www.loom.com/share/32175082428a480c92f6f4230345c9a1>