

# **INVENTORY MANAGEMENT SYSTEM FOR RETAILERS**

**TEAM ID : PNT2022TMID08254**

## **TEAM MEMBERS:**

ROHINI S

DIVYA R

POORANI S

VAISHNAVI A

GIFTLIN PRIYANKA R

**BACHELOR OF TECHNOLOGY  
IN  
INFORMATION TECHNOLOGY**

**ADHIYAMAAN COLLEGE OF ENGINEERING  
HOSUR(AUTONOMOUS)600025  
NOVEMBER 2022**

## **TABLE OF CONTENT**

<b>CHAPTER</b>	<b>CONTENTS</b>	<b>PAGE NO</b>
	<b>INTRODUCTION</b>	
1	1.1 PROJECT OVERVIEW	04
	1.2 PURPOSE	05
	<b>LITERATURE SURVEY</b>	
2	2.1 EXISTING PROBLEM	06
	2.2 REFERENCE	07
	2.3 PROBLEM STATEMENT DEFINITION	08
	<b>IDEATION &amp; PROPOSED SOLUTION</b>	
	3.1 EMPATHY MAP CANVAS	09
	3.2 IDEATION & BRAINSTROMING	10
3	3.3 PROPOSED SOLUTION	10
	3.4 PROBLEM SOLUTION FIT	12
	<b>REQUIREMENT ANALYSIS</b>	
4	4.1 FUNCTIONAL REQUIREMENT	13
	4.2 NON-FUNCTIONAL REQUIREMENTS	14
	<b>PROJECT DESIGN</b>	
5	5.1 DATA FLOW DIAGRAMS	15
	5.2 SOLUTION & TECHNICAL	17

	ARCHITECTURE	
	5.3 USER STORIES	18
6	<b>PROJECT PLANNING &amp; SCHEDULING</b>	
	6.1 SPRINT PLANNING & ESTIMATION	19
	6.2 SPRINT DELIVERY SCHEDUL	20
	6.3 REPORTS FROM JIRA	24
7	<b>CODING &amp; SOLUTIONING</b>	
	7.1 FEATURE1	26
	7.2 FEATURE2	26
	7.3 DATABASE SCHEMA	27
	<b>TESTING</b>	
8	8.1 TEST CASES	28
	8.2 USER ACCEPTANCE TESTING	30
9	<b>RESULTS</b>	
	9.1 PERFORMANCE METRICS	37
10	<b>ADVANTAGES &amp; DISADVANTAGES</b>	38
11	<b>CONCLUSION</b>	40
12	<b>FUTURE SCOPE</b>	41

## **APPENDIX**

13	SOURCE CODE	42
	GITHUB & PROJECT DEMO LINK	53

## **1.INTRODUCTION**

Inventory management helps companies identify which and how much stock to order at what time . It tracks inventory from purchase to the sale of goods . The practice identifies and responds to trends to ensure there always enough stock to fulfil the customer orders and proper warning of a shortage.Once sold ,inventory becomes revenue. Before it sells, inventory (although reported as an asset on the balance sheet) ties up cash. Therefore , too much stock costs money and reduces cash flow.One measurement of good inventorymanagement is inventory turnover. An accounting measurement,inventory turnover reflect how often stock is sold in period. A business does not want more stock than sales. Poor inventory turn over can lead to deadstock, or unsold stock.

If a product is purchased, then the related information is stored in stocks , that is , stocks are up to date. Another part I it prepare sales report after product it sold. in the sales information, the information about who sold the product is also kept, so there is no problem for misunderstandings in future.

## **1.1. PROJECT OVERVIEW**

Inventory ,A structured, categorized and complete collection of items or objects the give clear insights about the qualitative and quantitative attributes of these objects. In this modern age where the e-commerce websites are breaking the internet and consumption for the most products have exponentially increased,it is of utmost the importance to maintain a robust inventory system which is capable of maintaining a record like products , quantities and various other transitive data needed to run a business smoothly. Hence,we have developed a robust the Inventory management system which runs on flask and is connected to a MySQL database .The front end has been developed using bootstrap and connected to the backend using flask.The system will be login enabled . The managers will be able to add , update , delete , retrieve it and display it on the front-end.

## **1.2. PURPOSE:**

The main purpose of inventory management is to help businesses easily and efficiently manage the ordering, stocking, storing, and using of inventory. By effectively managing your inventory, you'll always know what items are in stock, how many of them there are, and where they are located.

Plus, practicing strong inventory management allows you to understand how you use your inventory—and how demand changes for it over time. You can zero in on exactly what you need, what's not so important, and what's just a waste of money. That's using the inventory management to practice inventory control. By the way, inventory control is the balancing act of always having enough stock to meet demand, while spending as little as possible on ordering and carrying inventory.

## **2.LITERATURE SURVEY**

### **2.1. EXISTING PROBLEM**

Businesses are quickly realising that inventory control is absolutely necessary to run an efficient business and make money in the process. Especially in today's competitive marketplace, business owners simply can't afford to have money go down the drain.

- High cost of inventory
- Consistent stock outs
- Low rate of inventory turnover
- High amount of obsolete inventory
- High amount of working capital
- High cost of storage



## REFERENCES

We have referred several documentations for the purpose of development phases.

- [1] <https://www.camcode.com/asset-tags/what-is-an-inventory-management-system/>
  
- [2] Jimmy Wales, online encyclopedia WiKipedia ,  
<http://www.wikipedia.org>
  
- [3] James Gosling. Java (Programming Language) ,  
<http://www.java.com>
  
- [4] Names Allaire, Netbeans-Fully-featured Java IDE,  
<http://www.netbeans.org>
  
- [5] James Gosling , Welcome to java world.com:  
how-to feature and columns by Java expert; news; Java applets;  
sample code ; tips ,  
<http://www.javaworld.com>
  
- [6] Pressman, Roger S.  
“Software Engineering A Practitioner” Approach
  
- [7] John Osborn , JavaBeans:  
Developing Component Software in Java
  
- [8] Doug Lea Concurrent Programming in Java:  
Design Principles and Pattern, Addison-Wesley , November,1996
  
- [9] Design Report, submitted 9th November 2012.  
[https://skydrive.live.com/redir?resid=2CEDE9F7F5F99604!196&authkey=!AO5I\\_ghTCML6xAk8](https://skydrive.live.com/redir?resid=2CEDE9F7F5F99604!196&authkey=!AO5I_ghTCML6xAk8)
  
- [10] Testing Document, submitted 26th November 2012.  
<https://www.camcode.com/asset-tags/what-is-an-inventory-management->

## 2.2. PROBLEM STATEMENT DEFINITION`

The problem statement aims to make desktop application for retailers and to track all areas of IMS like purchase details, sales details, stock management. The application helps the retailer to have complete insights about the products stored in the inventory and can manage them flexibly.

Project Name: Inventory Management system for retailers



Problem Statement (PS)	I am a(Retailers)	I'm trying to	But	Because	Which makes me feel
PS-1	Shopkeeper	Sell the product with some profit	I can't do it	I'm wasting the stocks in inventory with excess supply	Very depressed
PS-2	Retailer	Sell the products in demand	I can't do it my own	I didn't know the amount of stocks in inventory	Very distressed
PS-3	Retailer	Sell the product effectively	I can't do it	I don't know to manage the stocks in inventory	Very sad

### 3.IDEATION & PROPOSED SOLUTION

We have analyzed different systems and proposed an ideation phase of our developed management system.

#### 3.1. EMPATHY MAP CANVAS

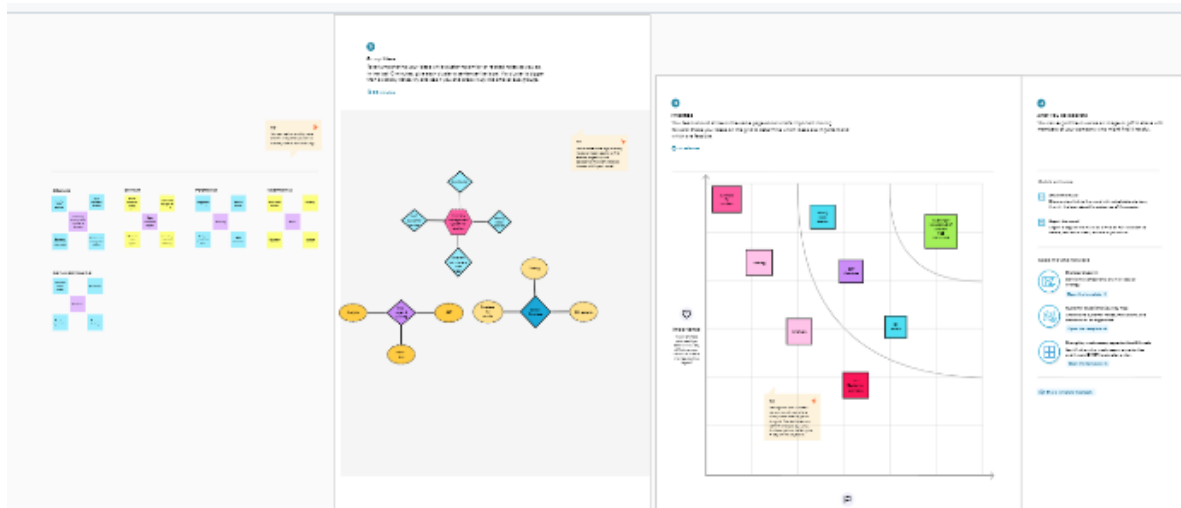
An empathy map canvas helps brands provide a better experience for users by helping teams understand the perspectives and mindset of their customers. Using a template to create an empathy map canvas reduces the preparation time and standardizes the process so you create empathy map canvases of similar quality.



<https://app.mural.co/invitation/mural/inventorysystem8245/1662990247889?sender=uc8206468e212748e2eea2294&key=b38a607b-b295-4de9-ab1d-1c6c5deba47b>

### 3.2. IDEATION & BRAINSTORMING

Noting down any ideas that come to mind that address your problem statement.



<https://app.mural.co/invitation/mural/inventorymanagementsystem0923/1663999065743?sender=uc8206468e212748e2eea2294&key=6cb9805a-42b7-42db-89f0-7900f1f71c90>

### 3.3. PROPOSED SOLUTION PREPROCESSING PHASES

#### Home:

This first module manages Home Screen Which is Provide A Home Page of my Software. After clicking home button . button will provide Welcome Screen of the Software etc.

#### Sales

This is Provide Sales information And Sales Page it is contain sales\_id, Product\_code , Product\_name , Quantity, Revenue, Sold by etc

#### Suppliers

Suppliers page contain suppliers details and its hold basic value with attributes it is provide a suppliers code, full name ,location ,phone etc Products:

It is hold the details of product with product code, product name, cost price selling price brand etc.

## **Purchase**

This is contain detail about purchase . It will provide purchase screen which is hold some value like purchase id ,product code ,product name ,quantity ,total cost etc And Each page has refresh facilities And search facilities and Direct input value interface etc .

## **Edit**

Many Module conatins Edit facilities Which has control of editing value from data base diretly and insert new value etc.

## **Clear and Delete**

Clear and delete is provide advance facilities of this software Because it is Provide a deletion and clear data process etc

In the proposed system, all the business operations will be automated. Some of the features which the new system will provide are Auto generation of Daily Demand report, Auto generation of Purchase Order of various raw materials. As everything is auto generated, the production delays are avoided. It makes the system more secure as only authenticated users can access the system. Also, there are privileges in which we can authorize a particular user for accessing system or particular modules of the application.

### **3.4. PROBLEM SOLUTION FIT**

- To develop a system that will enhance the monitoring of the sales and inventory
- To develop a module that can generate monthly sales and inventory report.
- To develop security in terms of keeping the records of the inventory
- To develop a system that can monitor the stocks inventory in a fast and efficient manner.
- To accurately record, compute and produce a report of sales.

## 4.REQUIREMENT ANALYSIS

### 4.1. FUNCTIONAL REQUIREMENTS

#### Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through User Name with password
FR-2	User Confirmation	Confirmation Email Verification using OTP
FR-3	Sign In	Sign in to application using Gmail Enter username Enter password
FR-4	Dashboard	View the stock
FR-5	Ordering	Add to cart Ordering the product
FR-6	Delivery	Product deliver to customer Cash on delivery (or)online payment



## 4.2 NON-FUNCTIONAL REQUIREMENTS

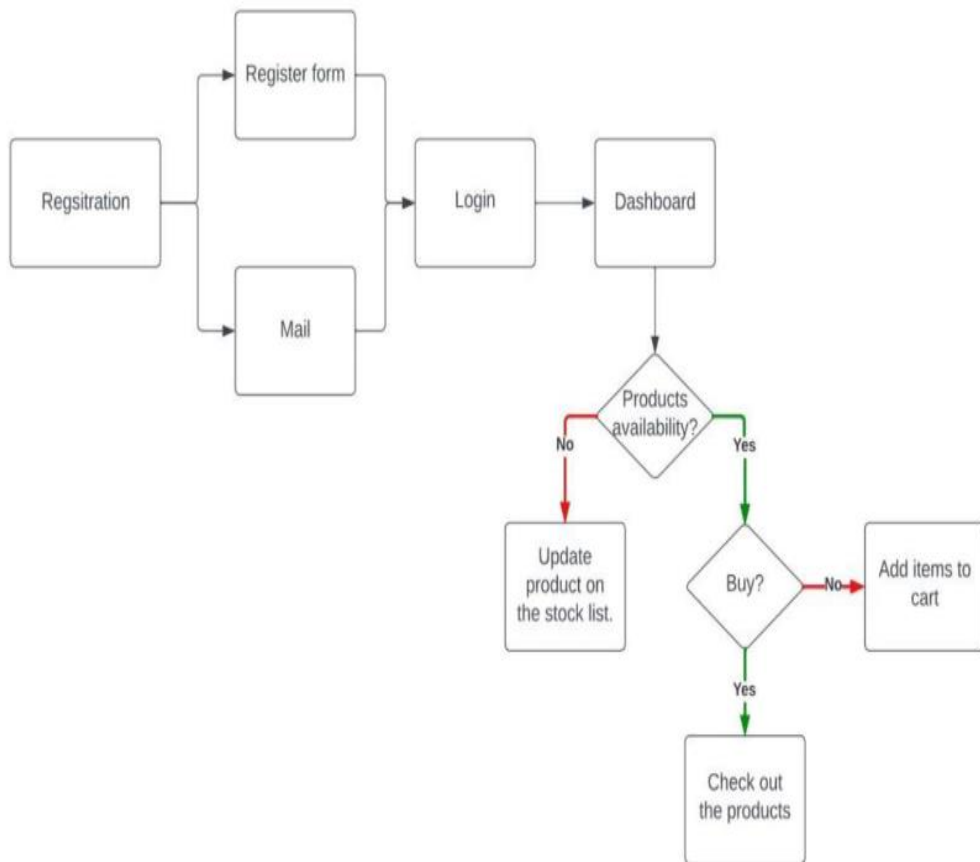
### Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	<b>Usability</b>	The web-page look's attractive. The navigate web site for the user is used easily.
NFR-2	<b>Security</b>	Security can be secure by using OTP. The site can be secure by based on cookies. Use strong password to secure the device.
NFR-3	<b>Reliability</b>	When browsing the web page is busy due to server down It should be available in all times
NFR-4	<b>Performance</b>	It is mainly used to reduce cost,time period,predict the best product This makes the Business more preductive for retailers
NFR-5	<b>Availability</b>	This uses IBM DB2 to store the data It is highly availability of database servers

## 5.PROJECT DESIGN

### 5.1.DATAFLOWDIAGRAMS

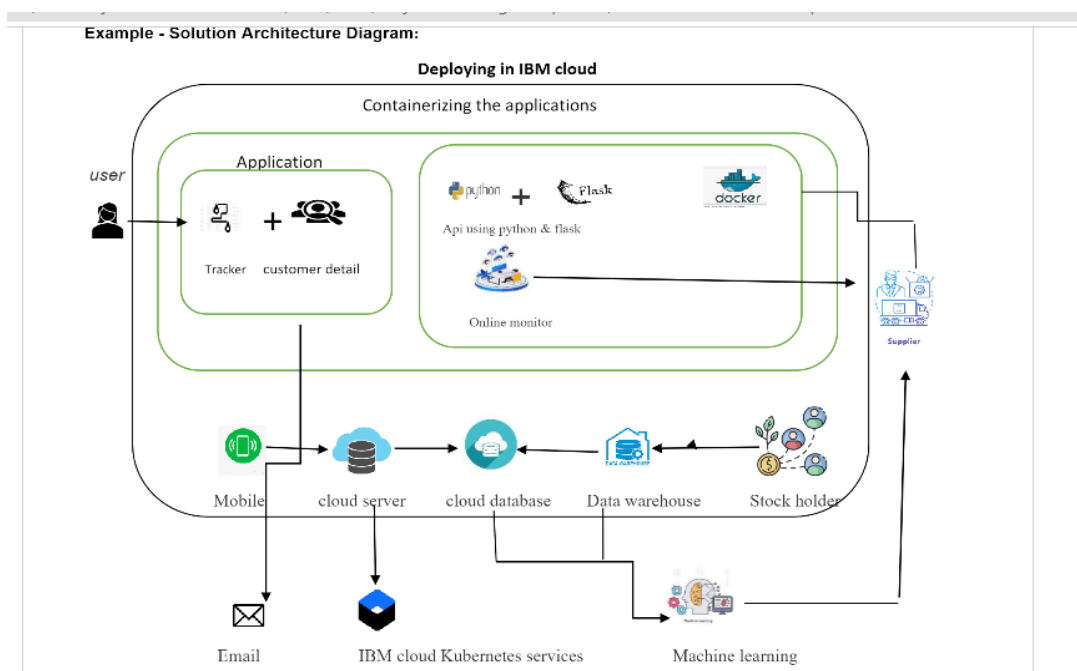


LINK-

<https://app.mural.co/t/inventorymanagementsystem0923/m/inventorymanagementsystem0923/1664960409961/8cbc41047cb30e69c69d9a7fe1a3e594bc5891e1?sender=ufaa8829900151ad6bb0c3652>

## 5.2. SOLUTION & TECHNICAL ARCHITECTURE

- There was no efficient solution available in many companies these days.
- Every process was based on paper work.
- Human fault rate were high.
- Tracing the inventory losses were not possible
- There was no efficient login system.
- After the computer age, every process is started to be integrated into computer environment.
- Now, we have qualified technology to implement new solution to these problems.



<https://app.mural.co/t/inventorymanagementsystem0923/m/inventorymanagementsystem0923/1664949289401/f06301abdbcca5e1f3940d03bbdcd01f086dce72?sender=ufaa8829900151ad6bb0c3652>

## 5.3. USER STORIES

A user story is the smallest unit of work in an agile framework. It's an end goal, not a feature, expressed from the software user's perspective.

### User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (web user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
	Conformation	USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
	Login	USN-4	As a user, I can login into the application by entering email & password	I ca with login into the application with verified email and password	Medium	Sprint-1
	Dashboard	USN-5	As a user, I can view the products which are available	Once I log on to the application, I can view products to buy	High	Sprint-2
	Add items to cart	USN-6	As a user, I can add the products I wish to buy to the carts	As a user, I can buy any products or add it to my cart for buying it later	Medium	Sprint-2
	Stock update	USN-7	As a user, I can add the products which are not available in the dashboard to the stock list	If any of the products which are not available, as a user I can update the inventory and send mail to the owner	Medium	Sprint-3
Customer Care Executive	Request to customer care	USN-8	As a user, I can contact the customer care executive and request any services I want from the customer care	As a user, I can contact the customer care and get support from them	Low	Sprint-4
Administrator	Contact Administrator	USN-9	I can able to report any difficulties I experience as a report	As a user I can give my support in my possible ways to administrator and the administration	Medium	Sprint-4

## **6. PROJECT PLANNING & SCHEDULING**

### **6.1. SPRINT PLANNING & ESTIMATION**

#### **Sprint 1:**

1. We created a Flask Project.
2. Added all the routes needed for our project
3. Created Tables in IBM Cloud.

#### **Sprint 2:**

1. We added all the html templates needed for our project.
2. We styled those pages using CSS and Bootstrap
3. We wrote Queries to connect IBM Cloud Database
4. Finished all the Fetching and Posting Stuff of IBM Cloud Database Integration.

#### **Sprint 3:**

1. Integration of Send grid into our application

#### **Sprint 4:**

1. Deploying the application using Docker and Kubernetes

## 6.2. SPRINT DELIVERY SCHEDULE

### Project Planning Phase

#### Project Planning Template (Product Backlog, Sprint Planning, Stories, Story points)

Date	22 October 2022
Team ID	PNT2022TMID08254
Project Name	Inventory management system for retailers
Maximum Marks	8 Marks

#### Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule

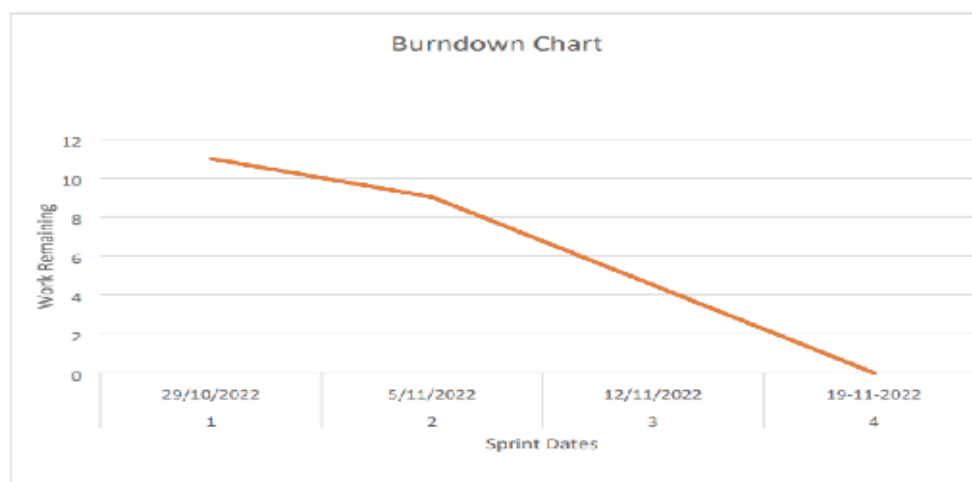
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by using my email & password and confirming my login credentials.	3	High	Rohini S Divya R Poorani s Vaishnavi A Giftlin prinyanka R
Sprint-1		USN-2	As a user, I can login through my E-mail.	3	Medium	Rohini S Divya R Poorani s Vaishnavi A Giftlin prinyanka R
Sprint-1	Confirmation	USN-3	As a user, I can receive my confirmation email once I have registered for the application.	2	Medium	Rohini S Divya R Poorani s Vaishnavi A Giftlin prinyanka R
Sprint-1	Login	USN-4	As a user, I can log in to the authorized account by entering the registered email and password.	3	Medium	Rohini S Divya R Poorani s Vaishnavi A Giftlin prinyanka R
Sprint-2	Dashboard	USN-5	As a user, I can view the products that are available currently.	4	High	Rohini S Divya R Poorani s Vaishnavi A Giftlin prinyanka R
Sprint-2	Stocks update	USN-6	As a user, I can add products which are not available in the inventory and restock the products.	3	Medium	Rohini S Divya R Poorani s Vaishnavi A

Sprint-2	Stocks update	USN-6	As a user, I can add products which are not available in the inventory and restock the products.	3	Medium	Rohini S Divya R Poorani S Vaishnavi A Gifulin prinyanka R
Sprint-3	Sales prediction	USN-7	As a user, I can get access to sales prediction tool which can help me to predict better restock management of product	6	Medium	Rohini S Divya R Poorani S Vaishnavi A Gifulin prinyanka R

Sprint - 4	Request for customer care	USN-9	As a user, I am able to request customer care to get in touch with the administrators and enquire the doubts and problems.	4	Medium	Rohini S Divya R Poorani S Vaishnavi A Gifulin prinyanka R
d	Giving feedback	USN-9	As a user, I am able to send feedback forms reporting any ideas for improving or resolving any issues I am facing to get it resolved.	3	Medium	Rohini S Divya R Poorani S Vaishnavi A Gifulin prinyanka R

### Project Tracker, Velocity & Burndown Chart: (4 Marks)

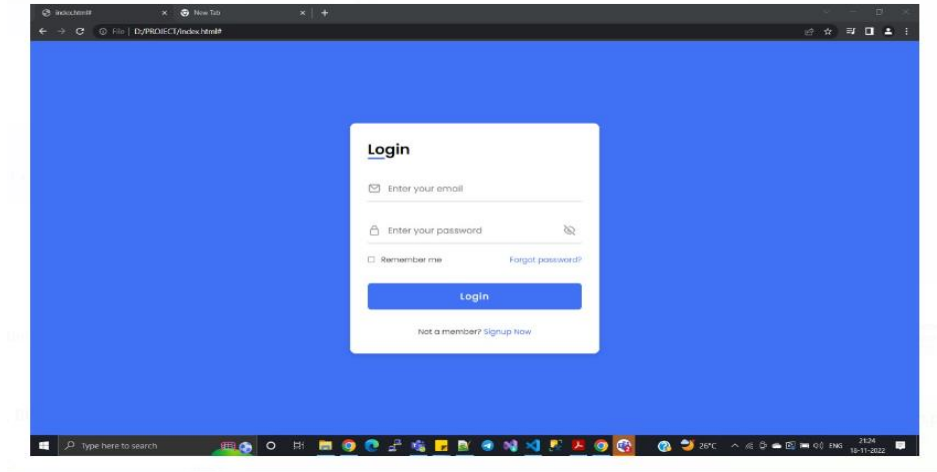
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	11	6 Days	24 Oct 2022	24 Oct 2022	11	24 Oct 2022
Sprint-2	7	6 Days	31 Oct 2022	05 Nov 2022	7	05 Nov 2022
Sprint-3	6	6 Days	07 Nov 2022	12 Nov 2022	6	12 Nov 2022
Sprint-4	7	6 Days	14 Nov 2022	19 Nov 2022	7	19 Nov 2022



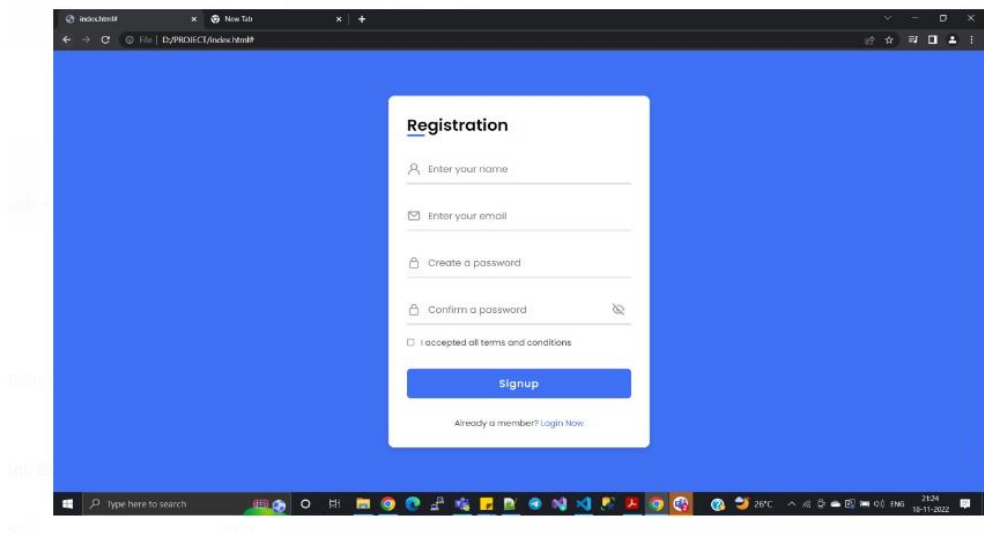


# CODING & SOLUTIONING

## 7.1.FEATURE1



## 7.2. FEATURE 2





## 7. TESTING

### 8.1. TEST CASE

Features to be tested	Test Description
Login to the system	This tests the login interface of the system.
Adding a Recipe to database	This test is conducted to verify if a recipe is successfully added to the database. This will check if the recipe is added to its header table and also check if the recipe details are added to the recipe details table.
Adding an Ingredient to database	This tests checks if new ingredient is added correctly to the database with the specified details.
Adding a Vendor to the database	This test checks if the newly added vendor is correctly added to the database with the specified details.
Checking the threshold levels	This test is conducted to verify if the ingredients that are below the threshold levels are listed by the function when called by the user. The verification is done by referring to the database.
Updating the sales for the day	This test is conducted to test the sales update in the database. The test checks if the database is updated with the correct ingredient values based on the sales data input to the system.
Updating the order reception to database	This test is conducted to test the correct updating of the database after receiving the order from the vendor.
Create Orders	This test is conducted to check the order creation capability of the system. The list of ingredients that is generated for order must comply with the set conditions of threshold levels

## **8.2. USER ACCEPTANCE TESTING**

### **Test case : Testing the Add Recipe Interface and its functioning**

Case 1: Testing the Quantity input field.

Case 2: Testing the Recipe Name field.

Case 3: Testing the Ingredients in recipe list and Quantity of ingredient list.

Case 4: Testing the available ingredients list.

Case 5: Testing the all the above cases together and checking if the entries are updated to the tables in database.

### **Test Case : Check Threshold Interface**

Case 1: Check if the Ingredients under the threshold values are shown in the Ingredients below threshold list.

Case 2: Check if the Create order button asks the user to enter values for all the ingredients listed under the ingredients below threshold list.

Case 3: Check if pressing the Process Order button creates a file with the order details in it.

### **Test Case : Testing the Update after sales interface**

Case 1: Test the Recipe list box.

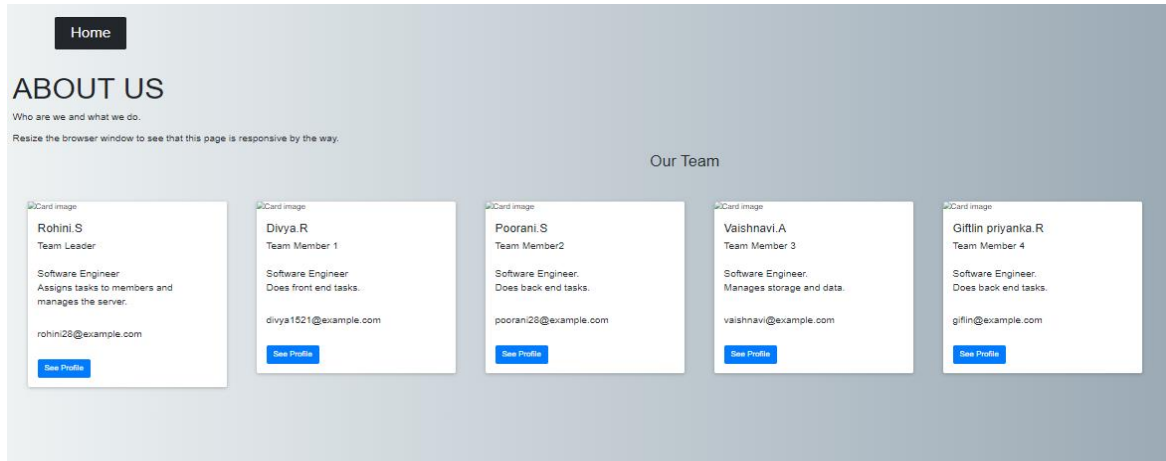
Case 2: Test the quantity text field..

Case 3: Test the recipe sold list box quantity sold list box.

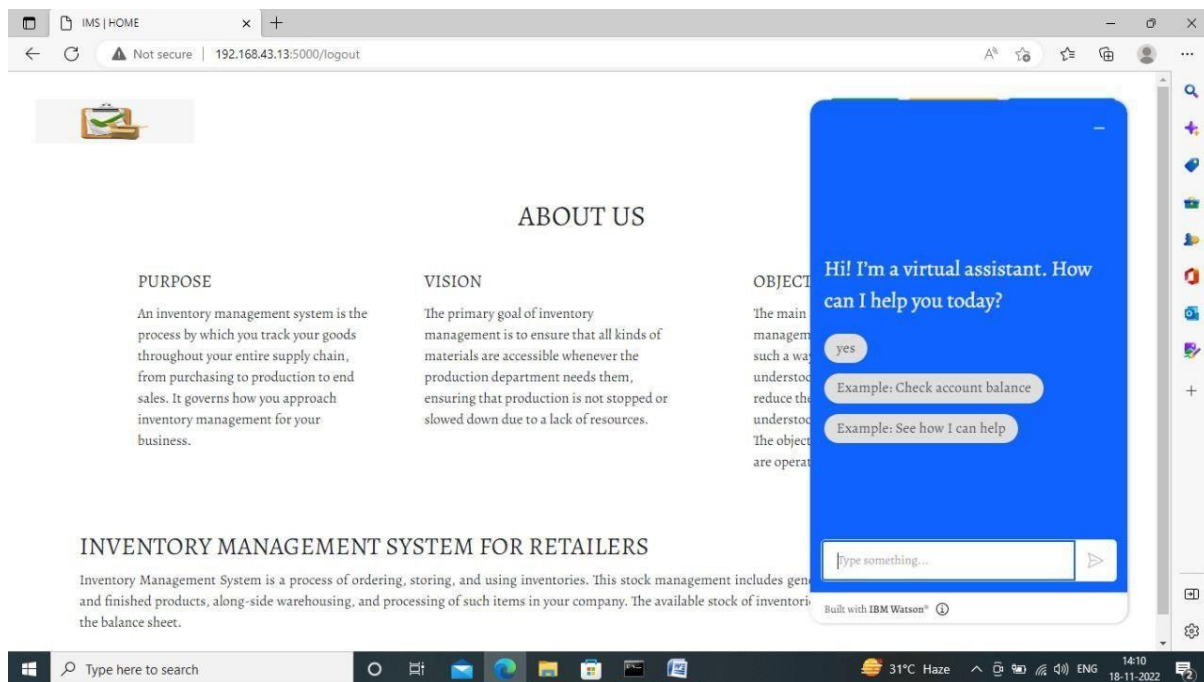
Case 4: Test if the details are updated to the database when requested.

## 8. RESULTS

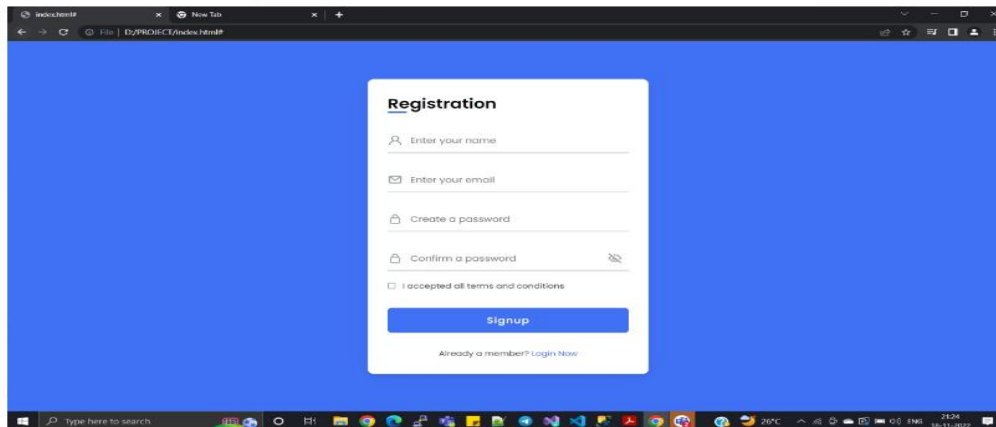
### HOME PAGE



### HOME PAGE WITH VIRTUAL ASSISTANT



## REGISTRATION FORM

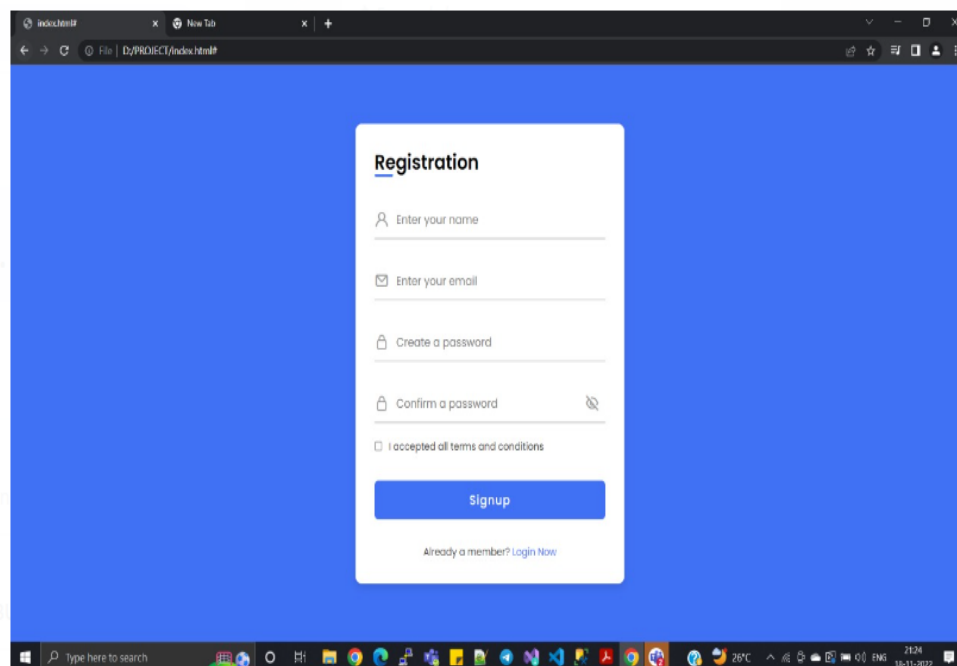


The screenshot shows a web browser window with a blue background. In the center is a white registration form titled "Registration". The form contains the following fields and elements:

- A text input field with the placeholder "Enter your name" and a person icon.
- A text input field with the placeholder "Enter your email" and an envelope icon.
- A text input field with the placeholder "Create a password" and a lock icon.
- A text input field with the placeholder "Confirm a password" and a lock icon.
- A checkbox labeled "I accepted all terms and conditions".
- A blue "Signup" button.
- A link below the button that says "Already a member? Login Now".

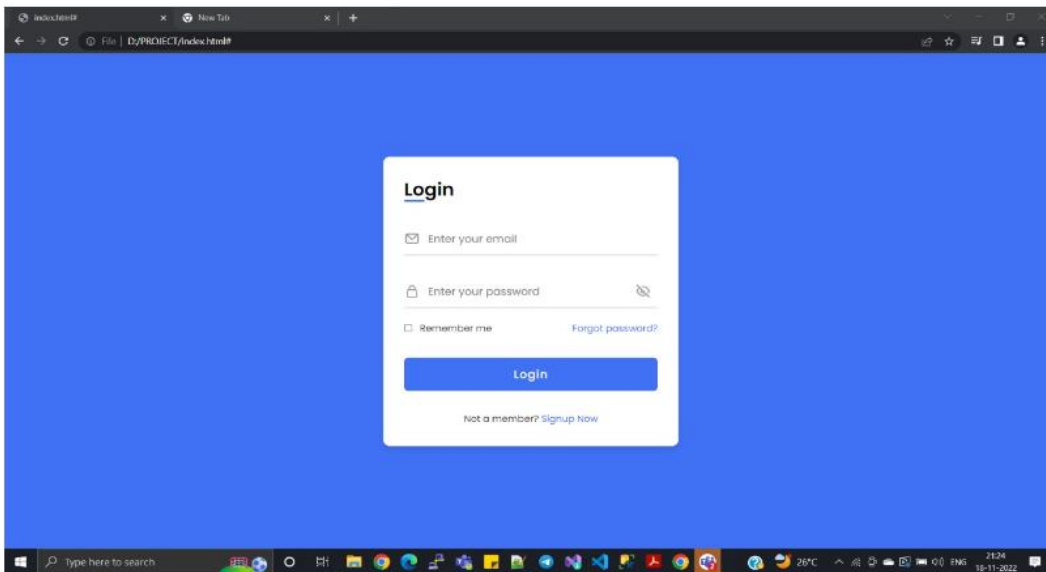
The browser's address bar shows "File | D:\PROJECT\index.html#". The Windows taskbar at the bottom displays the search bar, various application icons, and system information including the date "10-11-2022".

## REGISTRATION SUCCESSFULLY



This screenshot is identical to the one above, showing the same registration form on a blue background. The form fields, buttons, and links are the same, and the browser window and Windows taskbar are also identical.

# LOGIN FORM

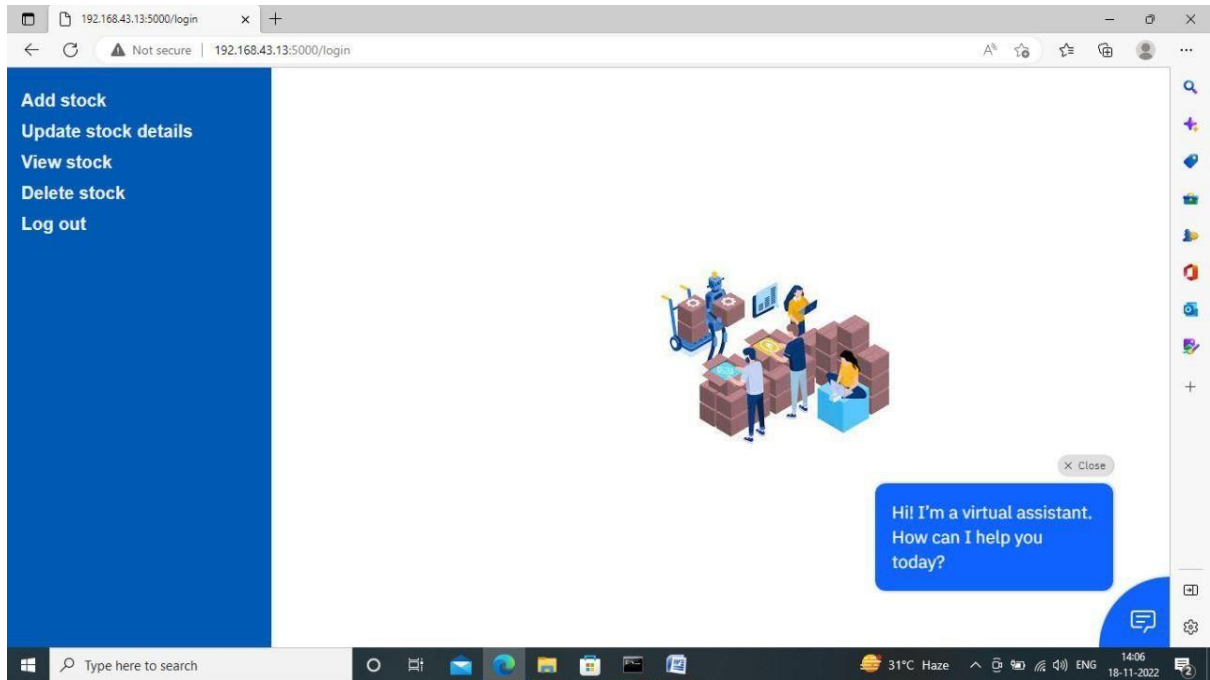


The image shows a web browser window with a blue background. In the center is a white login form titled "Login". The form contains the following elements:

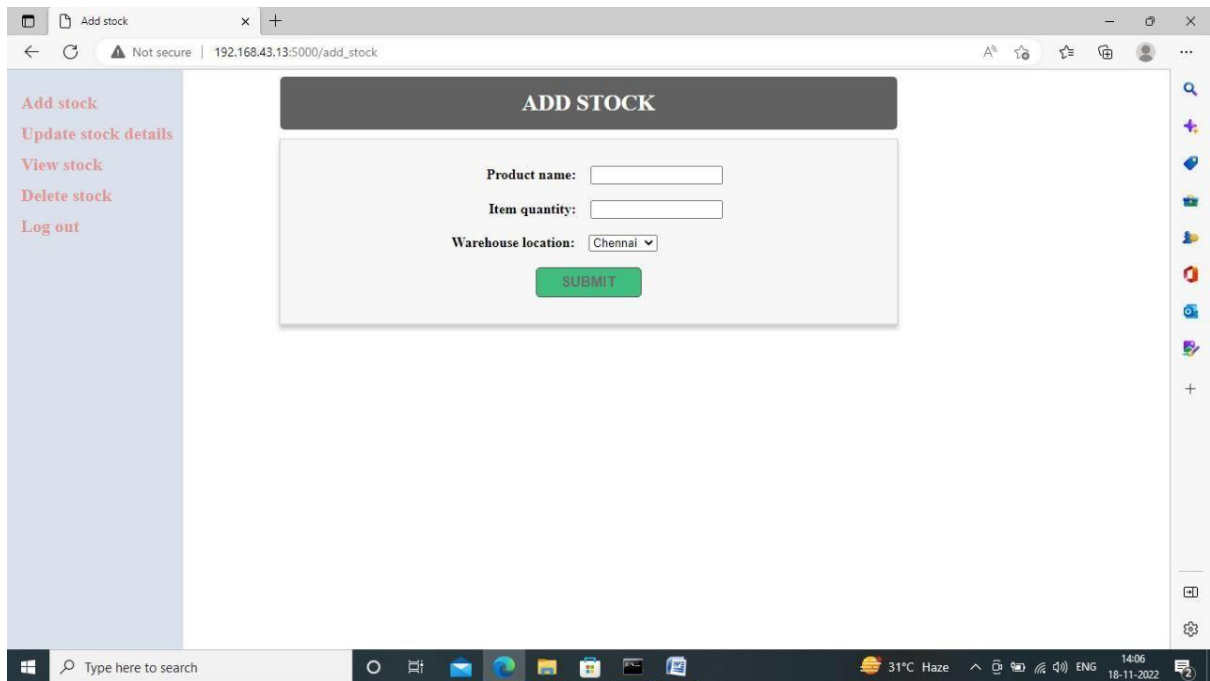
- An email input field with a placeholder "Enter your email" and an envelope icon.
- A password input field with a placeholder "Enter your password", a lock icon, and a toggle icon.
- A checkbox labeled "Remember me" and a link "Forgot password?".
- A blue "Login" button.
- A link "Not a member? Signup now" at the bottom.

The browser's address bar shows "File | D:/PROJECT/index.html". The Windows taskbar at the bottom displays various application icons, a search bar, and system information including the date "18-11-2022".

## DASHBOARD



## ADDSTOCK





## VIEW STOCK

The screenshot shows a web browser window with the address bar displaying '192.168.43.13:5000/view\_stock'. The page has a blue sidebar on the left with the following menu items: 'Add stock', 'Update stock details', 'View stock' (which is highlighted), 'Delete stock', and 'Log out'. The main content area displays a table with the following data:

Product name	Item quantity	Warehouse location
tomoto	9	Chennai
milk	20	Chennai

In the bottom right corner, there is a blue chat bubble with the text: 'Hi! I'm a virtual assistant. How can I help you today?'. The Windows taskbar at the bottom shows the search bar, task view button, and several application icons. The system tray on the right indicates a temperature of 31°C, 'Haze' weather, and the date '18-11-2022'.

## UPDATE STOCK

The screenshot shows the 'UPDATE STOCK' form in the web application. The browser address bar shows '192.168.43.13:5000/update'. The blue sidebar on the left remains the same, with 'Update stock details' now highlighted. The main content area features a form titled 'UPDATE STOCK' with the following fields:

- 'Product name:' with a text input field containing 'milk'.
- 'Item quantity:' with a numeric input field containing '20'.
- 'Warehouse location:' with a dropdown menu showing 'Chennai'.
- An 'UPDATE' button below the fields.

The same virtual assistant chat bubble is present in the bottom right corner. The Windows taskbar and system tray are consistent with the previous screenshot.

## UPDATESTOCKDETAILS

The screenshot shows a web browser window with the address bar displaying '192.168.43.13:5000/view\_stock'. The page has a blue sidebar on the left with the following menu items: 'Add stock', 'Update stock details', 'View stock', 'Delete stock', and 'Log out'. The main content area displays a table with the following data:

Product name	Item quantity	Warehouse location
tomoto	9	Chennai
milk	20	Chennai

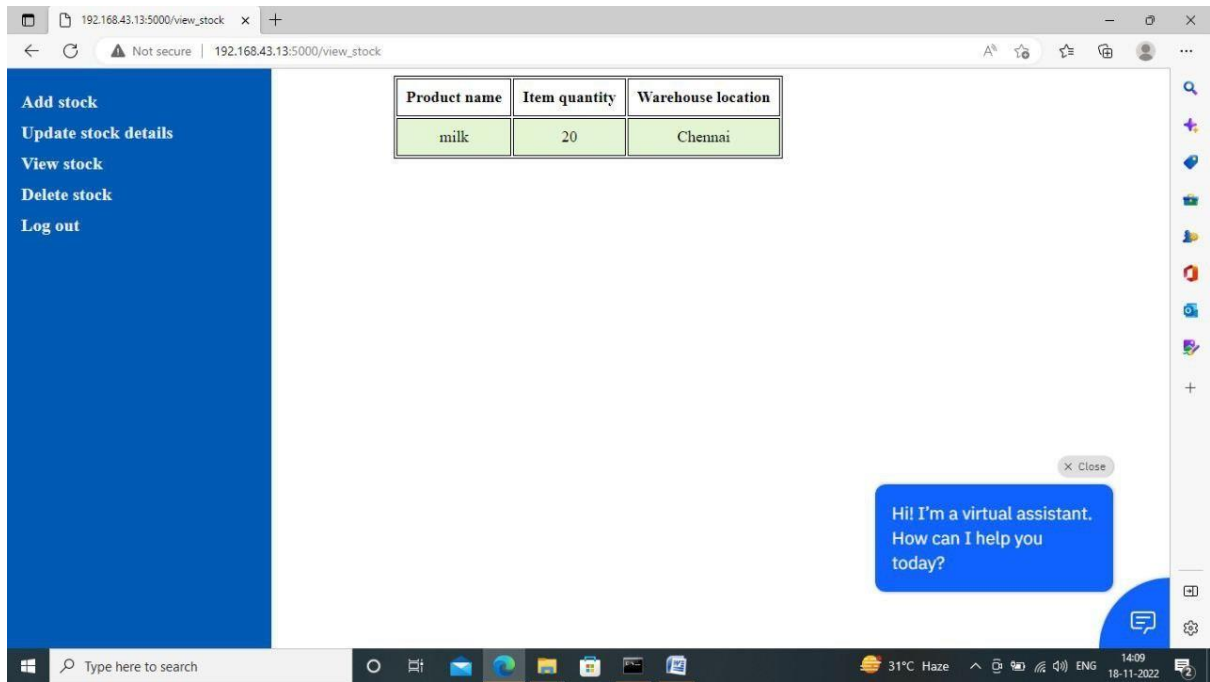
At the bottom right, there is a blue chat bubble with the text: 'Hi! I'm a virtual assistant. How can I help you today?'. The Windows taskbar at the bottom shows the search bar, task icons, system tray with weather (31°C Haze), and date/time (14:08, 18-11-2022).

## DELETE STOCK

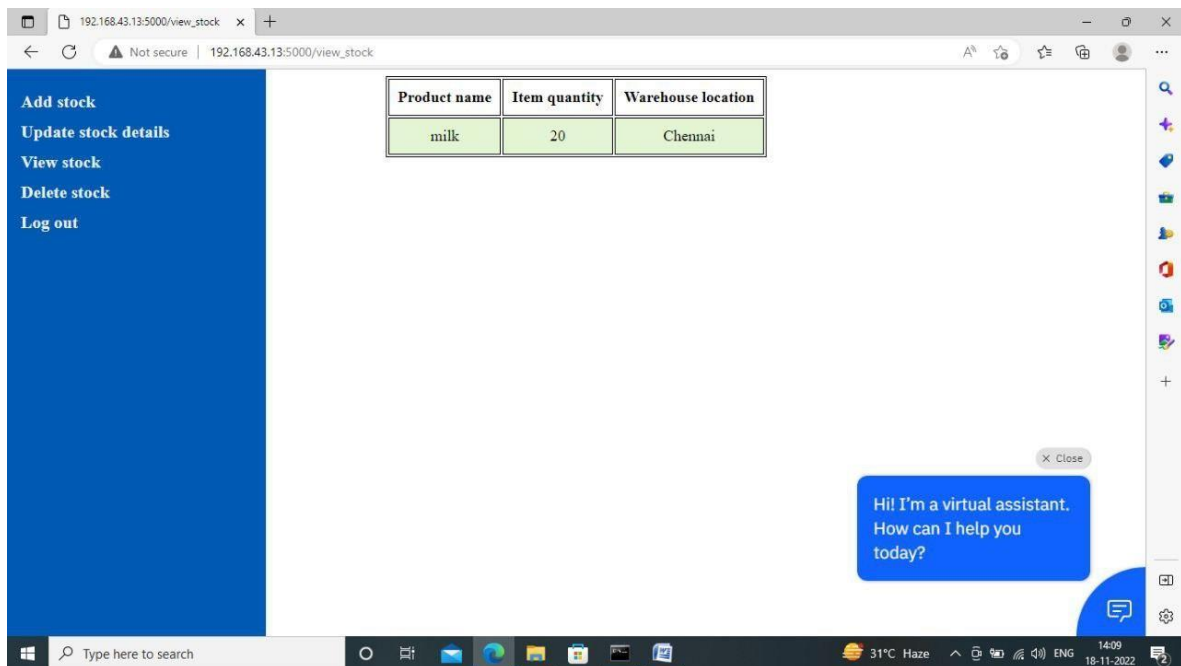
The screenshot shows a web browser window with the address bar displaying '192.168.43.13:5000/delete'. The page has a blue sidebar on the left with the following menu items: 'Add stock', 'Update stock details', 'View stock', 'Delete stock', and 'Log out'. The main content area features a 'Delete stock' form with the following fields:

Product name:

At the bottom right, there is a blue chat bubble with the text: 'Hi! I'm a virtual assistant. How can I help you today?'. The Windows taskbar at the bottom shows the search bar, task icons, system tray with weather (31°C Haze), and date/time (14:09, 18-11-2022). A status bar at the bottom left of the application window displays the text: 'Waiting for integrations.au-syd.assistant.watson.appdomaincloud...'.



## LOGOUT



## 9.1. PERFORMANCE METRICS

Inventory Performance is a measure of how effectively and efficiently inventory is used and replenished. The goal of inventory performance metrics is to compare actual on-hand dollars versus forecasted cost of goods sold. Many Lean practitioners claim that inventory performance is the single best indicator of the overall operational performance of a facility. Inventory performance looks at and is measured using either Inventory Days OnHand (DOH) or Inventory Turns.

- **Inventory Days On-Hand:** The number of days it would take to consume current on-hand inventory. Always measure multiple inventory item numbers in terms of currency (i.e. COGS).
- **Inventory Turns:** The number of times inventory is replaced in a year.

## **9. ADVANTAGES & DISADVANTAGES**

### **ADVANTAGES**

1. Better Inventory Accuracy.
2. Reduced risk of overselling.
3. Cost Savings.
4. Avoiding stockouts and Excess stock.
5. More Productivity.
6. Increased Profits.
7. Better Customer Experience.

## **DISADVANTAGES**

1. Expensive for small businesses.
2. Risk of system crashes.
3. Malicious Hacks.
4. Works for only single company

## **11. CONCLUSION**

The project “Inventory Management System for Retailers” mainly as the name suggests deals with the calculation of the available and processed resources for an accurate inventory control and process management for a domain specific client who are related to the subject of food chains/outlets. This enables the inventory to be applied at every level in the hierarchy of the products and its complex combinations of recipes. A system that accurately calculates the atomic ingredients used for making a recipe then automatically performs the back end operation pertaining to a database of many relational tables onto which the changes are being made with each and every operation performed on the front end and which also shows up if at the time of retrieval. The most important part of Inventory controlling is its ability to check for threshold levels and alert the manager to replenish the stock before it reaches a danger zone. So as when an ingredient level goes below the threshold level then it routes an alert to the manager. Then if needed accordingly an automated order form is produced so as to each specific vendor along with the quantities needed for replenishment. As a part of the standard maintaining a drill of risk management is done in order to sustain during the days of special occasion or holidays when the demand reaches to rather more different scale as compared to other days. These occasions call on for special inclusions into the menu which reflects on the recipes and in turn reflects the ingredients being used up eventually. Thus was provided the liberty of adding special recipe to the menu for some special occasion and is regarded as a key feature.

## **12.FUTURE SCOPE**

Future work can be done to add a sales page and payments page to manage stocks and product sells simultaneously. The customers can buy products from sales page with secured payment gateway in order to avoid fraudulents and scams



**App.py**

```
from flask import  
Flask,render_template,  
flash,redirect,  
url_for,request
```

```
app = Flask(__name__)  
app.secret_key="hello"
```

```
@app.route("/")  
def index():  
    return render_template("index.html")
```

```
@app.route("/about")  
def about():  
    return render_template("about.html")
```

```
@app.route("/list")  
def list():  
    return render_template("list.html")
```

```
@app.route('/signin',methods=["post","get"])  
def signin():  
    return render_template('signin.html')
```

```
@app.route('/signup',methods=["post","get"])  
def signup():
```

```
return render_template('signup.html')
```

## **home.html**

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

```
<link    rel="stylesheet"    href="https://cdnjs.cloudflare.com/ajax/libs/font-  
awesome/4.7.0/css/font-awesome.min.css">
```

```
<style>
```

```
body {
```

```
    font-family: "Lato", sans-serif;
```

```
}
```

```
/* Fixed sidenav, full height */
```

```
.sidenav {  
    height: 100%;  
    width: 300px;  
    position: fixed;  
    z-index: 1;  
    top: 0;  
    left: 0;  
    background-color: #0059b3;  
    overflow-x: hidden;  
    padding-top: 20px;  
}
```

```
/* Style the sidenav links and the dropdown button */
```

```
.sidenav a{  
    padding: 6px 8px 6px 16px;  
    text-decoration: none;  
    font-size: 20px;  
    color: rgb(239, 239, 239);  
    display: block;  
    border: none;  
    background: none;  
    width: 100%;  
    text-align: left;  
    cursor: pointer;  
    outline: none;  
}
```

```
/* On mouse-over */
```

```
.sidenav a:hover{
  color: #111;
}
```

```
/* Some media queries for responsiveness */
```

```
@media screen and (max-height: 450px) {
  .sidenav {padding-top: 15px;}
  .sidenav a {font-size: 18px;}
}
```

```
</style>
```

```
</head>
```

```
<body>
```

> Deeps:

```
<div class="sidenav">
```

```
  <a href="{ {url_for('add_stock') } }"><strong>Add stock<strong></a>
```

```
  <a href="{ {url_for('update') } }"><strong>Update stock details<strong></a>
```

```
  <a href="{ {url_for('view_stock') } }"><strong>View stock<strong></a>
```

```
  <a href="{ {url_for('delete') } }"><strong>Delete stock<strong></a>
```

```
  <a href="{ {url_for('logout') } }"><strong>Log out<strong></a>
```

```
  </div>
```

```
<nav>
```

```
  <script>
```

```
    window.watsonAssistantChatOptions = {
```

```
      integrationID: "4bd6f313-33d4-4e87-8825-22b90b8e3c2c", // The ID of
this integration.
```

```
      region: "au-syd", // The region your integration is hosted in.
```

serviceInstanceID: "60e1396a-421f-4091-b39a-a23a546843e8", // The ID  
of your service instance.

```
onLoad: function(instance) { instance.render(); }  
};  
setTimeout(function(){  
    const t=document.createElement('script');  
    t.src="https://web-  
chat.global.assistant.watson.appdomain.cloud/versions/" +  
(window.watsonAssistantChatOptions.clientVersion || 'latest') +  
"/WatsonAssistantChatEntry.js";  
    document.head.appendChild(  
});  
</script>  
</nav>  
  
</body>  
</html>
```

**GITHUB:**

[IBM-EPBL/IBM-Project-10050-1659089402](#)