PROJECT REPORT

Date	**19.11.2022**
Team ID	**PNT2022TMID27449**
Project Name	**Inventory Management System For Retailers**
Team Members	**Ashik Varghese (311119104012)** **Asmir Khan (311119104014)** **Chris Sasan Doss (311119104015)** **Jawin Arivumathi (311119104031)**

INDEX

1. INTRODUCTION

Project Overview

The objective of this system is to manage the items in an inventory such as tracking orders, placing orders to other suppliers and checking the items in the inventory. The system allows the admin to maintain the items in the inventory.

Whenever the item levels go low, the system places an order to the supplier. The supplier gets the notification of these orders as soon as they are placed and can send the items to the inventory. There are two login pages each for the admin and supplier.

The software has been developed using the most powerful and secured backend Python and IBM Cloud for the databases and most widely accepted frontend JavaScript with HTML and CSS coding

Purpose

The primary purpose of inventory management is to ensure there is enough goods or materials to meet demand without creating overstock, or excess inventory

Retail management refers to the process of helping customers find products in your store. It includes everything from increasing your customer pool to how products are presented, and how you fulfill a customer's needs. A good store manager helps customers leave the store with a smile.

2. LITERATURE SURVEY

1)Paper: Inventory management for retail companies: A literaturereview and current trends

Author: Cinthya Vanessa Munoz Macas,Jorge Andres Espinoza Aguirre,

Rodrigo Arcentales-Carrion,Mario Pena

Year: 2021

Description:

To analyze and present an extensive literature concerning inventory management, containing multiple definitions and fundamental concepts for the retail sector. The primary outcomes ofthis study are the leading inventory management systems and models, the Key Performance Indicators (KPIs) for their correct management, and the benefits and challenges for choosing or adopting an efficient inventory control and management system.

Future Work and Analysis:

To reduce the cost and maintenance and make available for all the companies.The need to analyze their KPIs becomes highly significant,as well as their different systems, methodologies, and tools used within inventory management and optimization.

2) **Paper:** Retail inventory management with stock-out based dynamicdemand substitution

Author: Baris TanSelcuk Karabati

Year: 2013

Description: To study an inventory management problem in a retail setting with stock-out based substitutions and multiple items in a product category and propose an approximate solution to determine theorder-up-to levels to maximize the expected profit subject to service level constraints. The method uses demand parameters including the substitution probabilities estimated from the point-of-sales data. The method provides a practical tool for retailers to manage their inventory.

Future Work and Analysis:

Through a computational study, by explicitly accounting for substitutions, the performance of the inventory system can be improved. The amount of improvement depends on the minimum direct service level requirement as well as the correlation between the market share and the profit margin of the products. By combining the method we presented in an earlier study to estimate the demand and customer choice parameters, the method we presented in thisstudy can be used to manage inventory in a better way in retailing.

3)Paper: Robust inventory management with stock-out substitution

Author: Zhaolin LiGrace Fu

Year: 2017

Description: Stock-out substitution is a well-documented phenomenon that occurs when customers seek a different product as a substitute for their first-choice item if it runs out of stock. We consider asingle-period inventory model with limited information regarding the external demands (i.e., mean, variance, and covariance) and focus on identifying the inventory levels that maximize the worst-case expected profit. We formulate a two-stage optimization model: the second stage characterizes the worst-case joint demand distribution by treating the inventory levels as input parameters, and the first stage identifies the optimal inventory levels based on the results of the second stage.

Future Work and Analysis:

Our approach makes use of the limited information on product demands and is suitable for the circumstance in which exact demand distributions can not be accurately estimated. After formulating the optimization model as a two-stage model, we find that the closed-form solution of the second stage is intractable except for two special cases. We develop a heuristic solutionbased on these two special cases. An extensive numerical study indicates that the performance of the heuristic solution is nearly optimal over a wide range of parameters. Investigating the effects of robust decision rule changes could offer interesting

4)Paper: Managing demand uncertainty: Probabilistic selling versusinventory substitution

Author: Yi ZhangGuowei Hua

Shoyang Wang Juliang Zhang Vicenc

Fernandez

Year: 2018

Description: To combat demand uncertainty, both strategies of inventory substitution and probabilistic selling can be used. Although the two strategies differ in operation, we believe that they share a common feature in combating demand uncertainty by encouraging some customers to give up some specific demand for the product to enable demand substitution. It is interesting to explore which strategy ismore advantageous to the retailer.

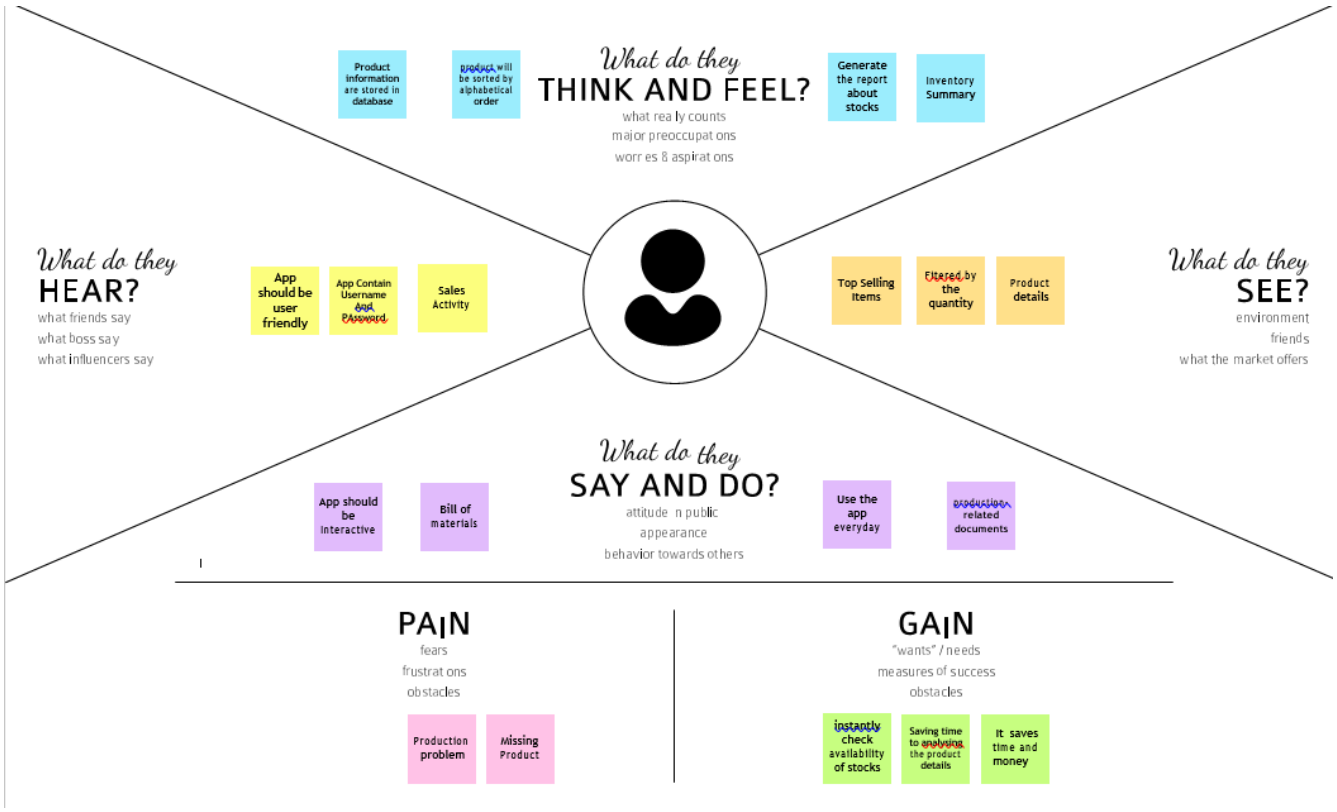
Future Work and Analysis:

inventory substitution is the better choice for the retailer when the product similarity is higher.The price of the probabilistic product is an exogenous variable. Future research mayextend our work by combining the pricing and inventory decisions. It is also worth considering PS in a supply chain setting. For example, it is interesting to explore the conditions under which a retailer's probabilistic selling will benefit the supplier, the retailer, and both.

3. IDEATION & PROPOSED SOLUTION

Empathy Map Canvas

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

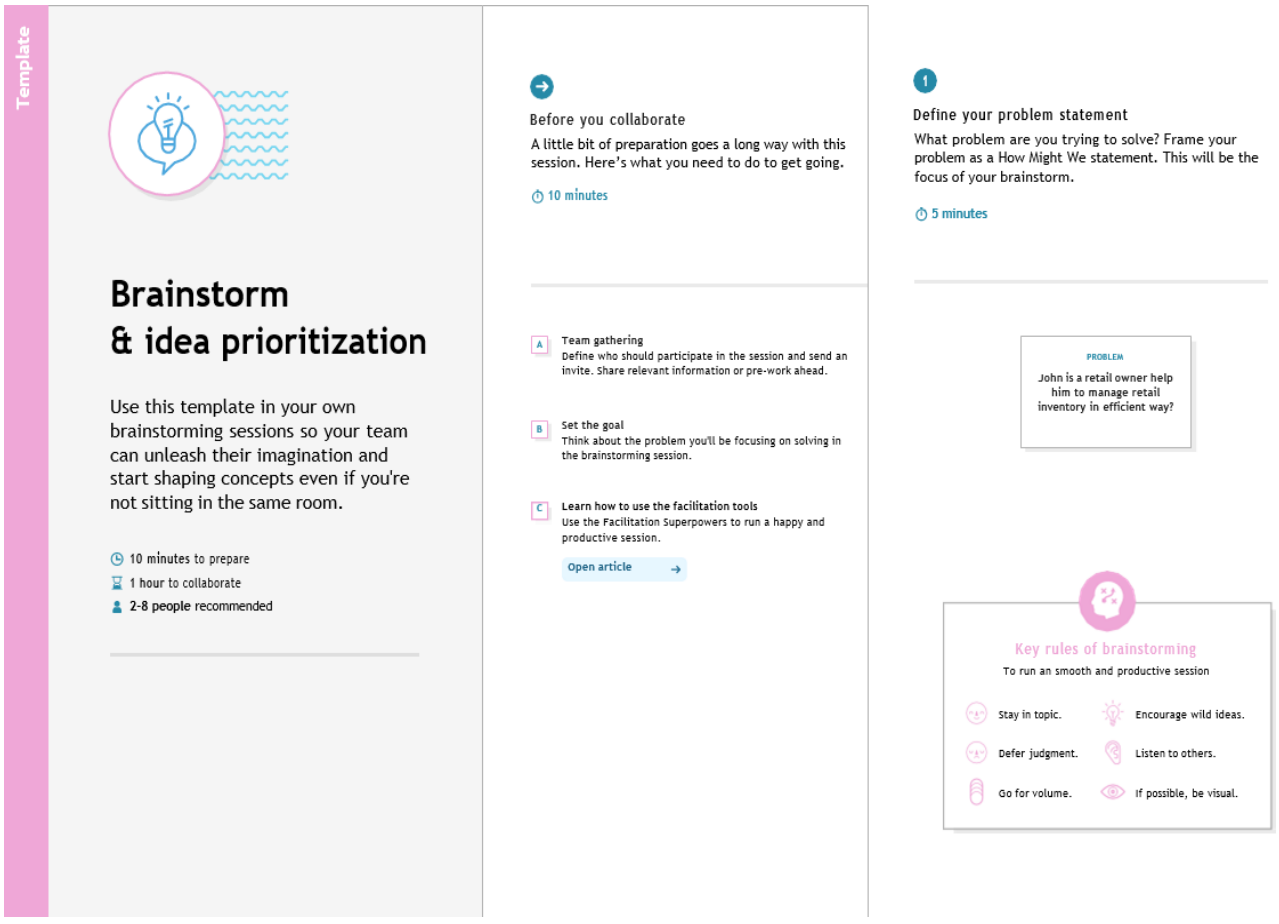


Ideation & Brainstorming

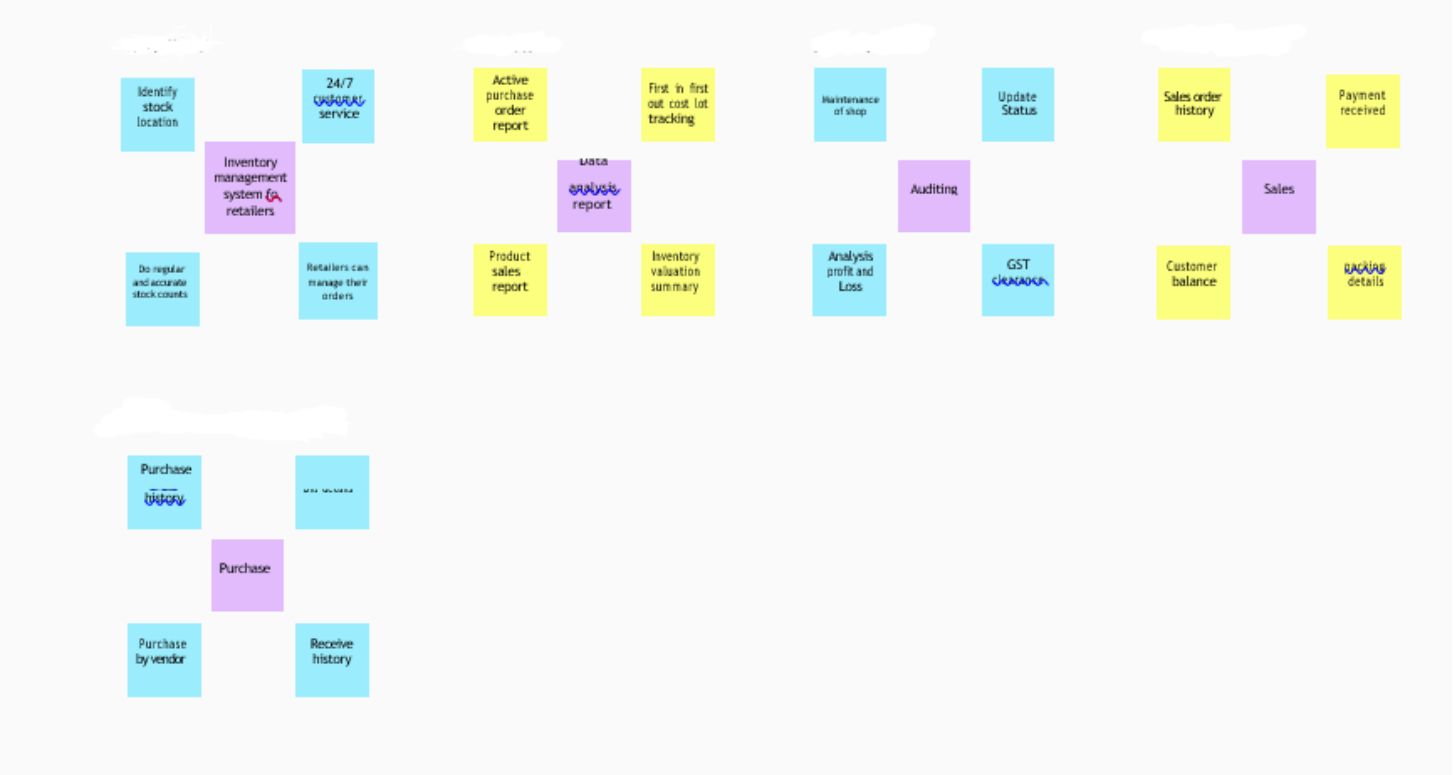
Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

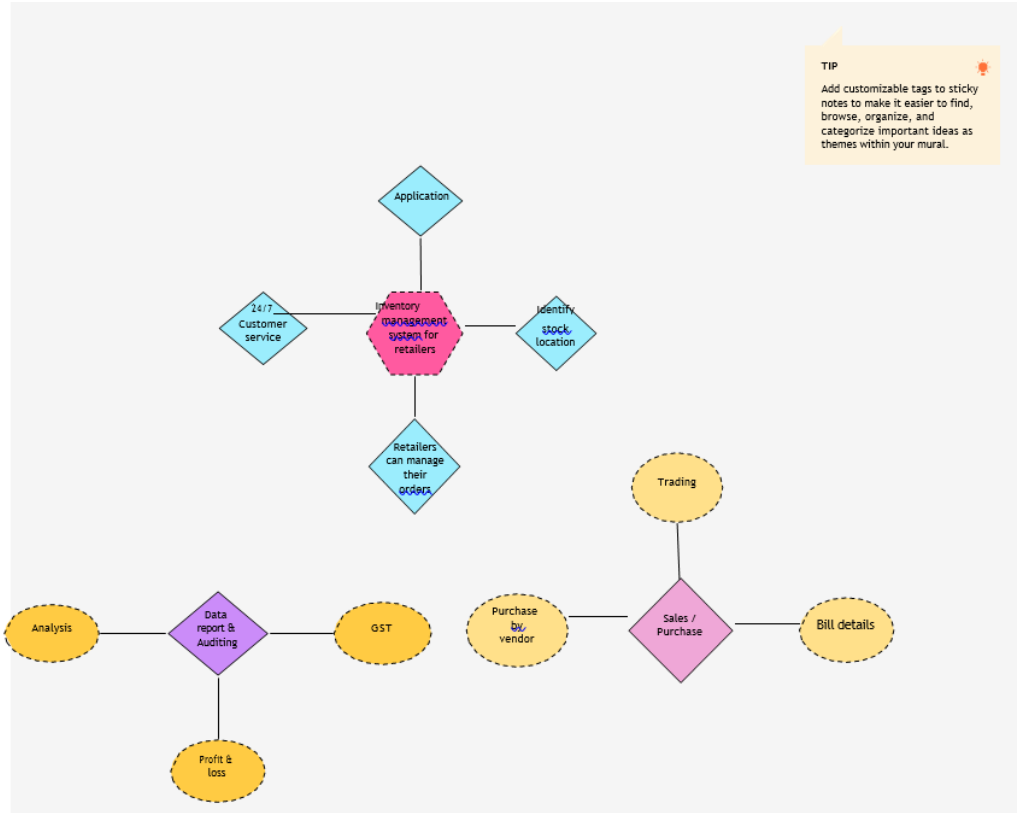
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Step-1: Team Gathering, Collaboration and Select the Problem Statement

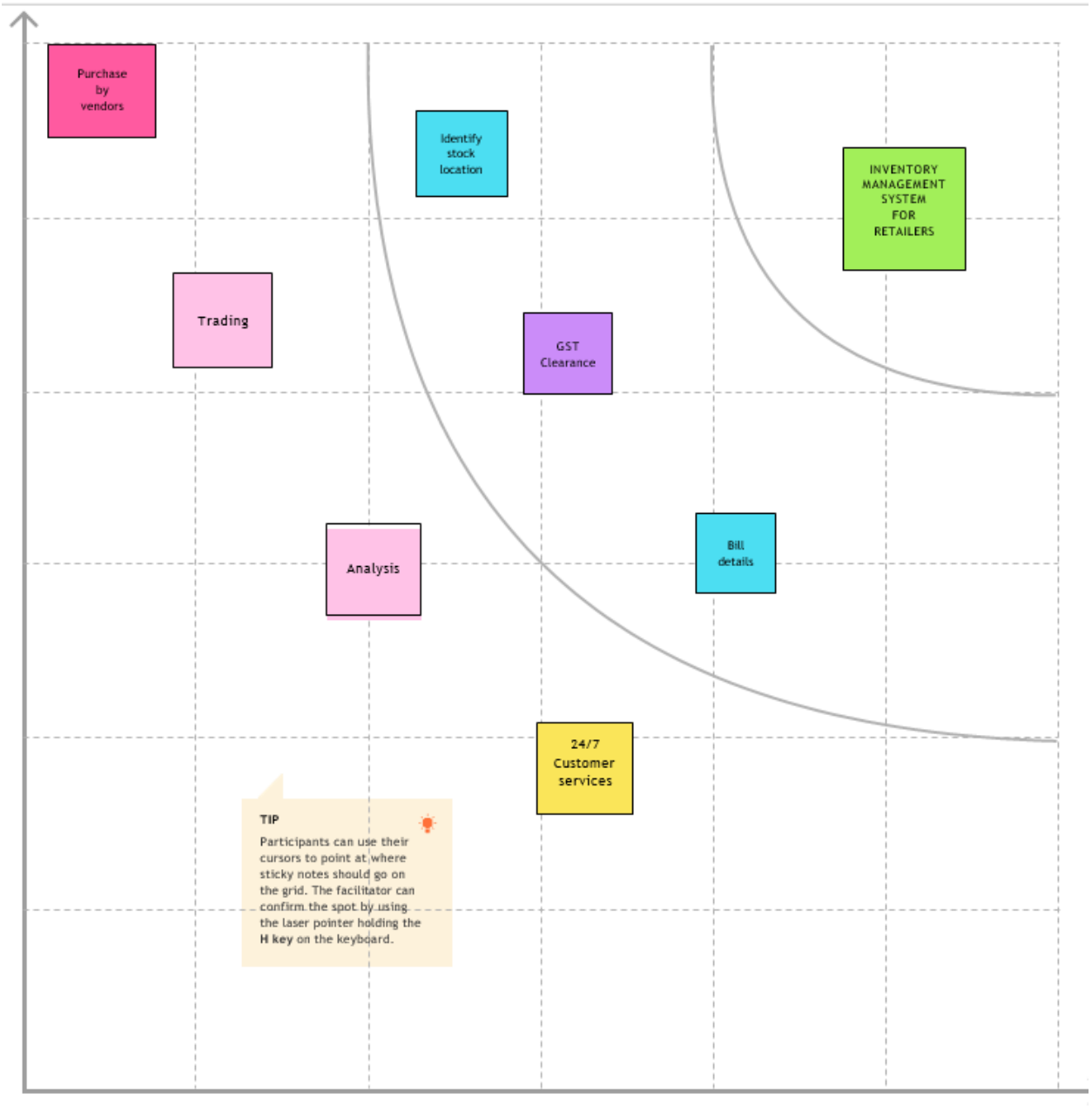


Step-2: Brainstorm, Idea Listing and Grouping





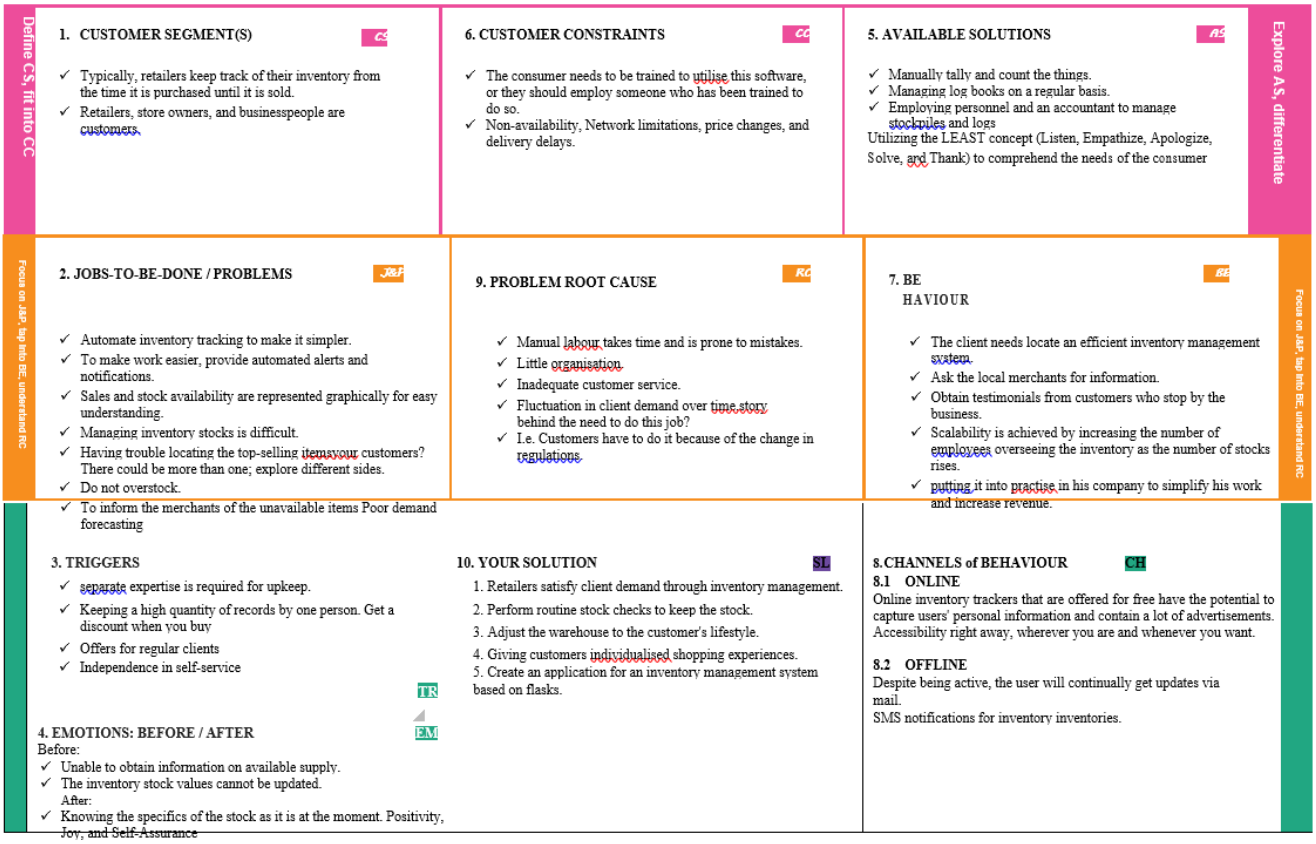
Step-3: Idea Prioritization



Proposed Solution

S. No	Parameter	Description
1.	Problem Statement (Problem to be solved)	➢ Retailers only keep track of inventory data in a logbook that is poorly organised, so they lack a systematic system for doing so promptly and safely.
2.	Idea / Solution description	➢ Creating a cloud-based inventory management system to efficiently maintain and manage the goods for retailers.
3.	Novelty / Uniqueness	➢ The dashboard displays the week's trending stocks. ➢ A graphic representation makes it simple to analyse the sale, allowing us to order and maintain stock properly.
4.	Social Impact / Customer Satisfaction	➢ Customers will receive additional options as a result of the abundance of products. ➢ Retailers will gather and examine customer feedback to identify any changes.
5.	Business Model (Revenue Model)	➢ Retailers may discern the true demands of their customers and place orders accordingly to prevent stock waste.

Problem Solution fit



4. REQUIREMENT ANALYSIS

Functional requirement

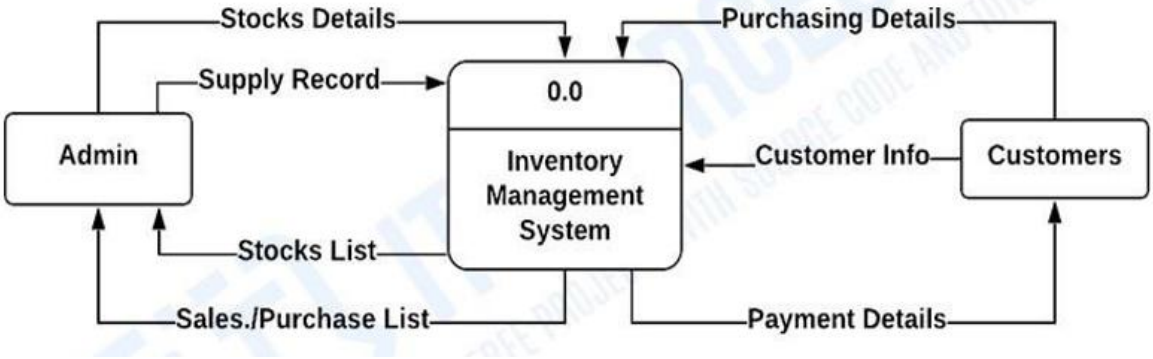
- The System aims at providing an efficient interface to the user for managing of inventory, it shall also provide the user varied options for managing the inventory through various functions at hand. The ingredient levels are continuously monitored based on their usage and are checked for the threshold levels in the inventory and accordingly the user is alerted about low levels of certain ingredients. The design is such that the user does not have to manually update the inventory every time, the System does if for the user.
- The System calculates and predicts the amount of usage for specific set days that are pre-set by the user(admin) , it also alerts the user of an impending action to order ingredients before the specific day set by the user. Therefore the user never has to worry about manually calculating the estimated usage of the ingredients as the System does it for the user.
- The simple interface of the System has functions like adding a recipe, removing or updating the recipe. It also extends to functions such as adding a vendor for an ingredient,, removing the vendor, checking threshold levels, processing orders, altering processed orders etc.

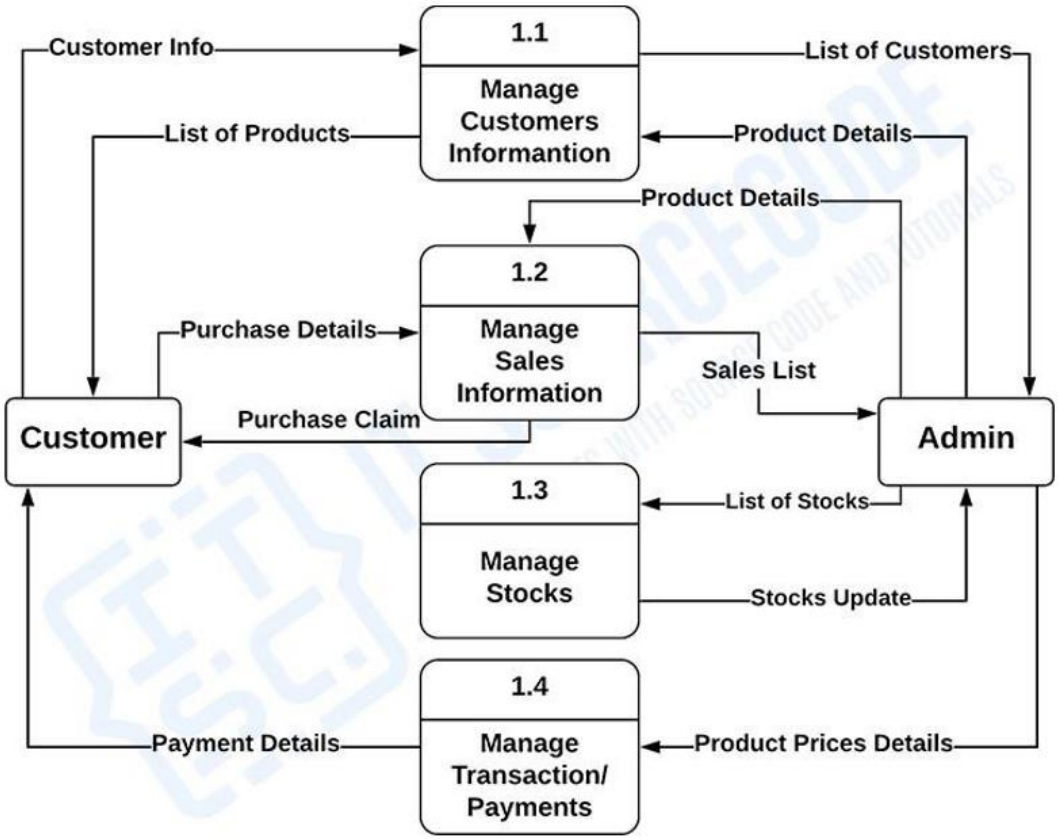
Non-Functional requirements

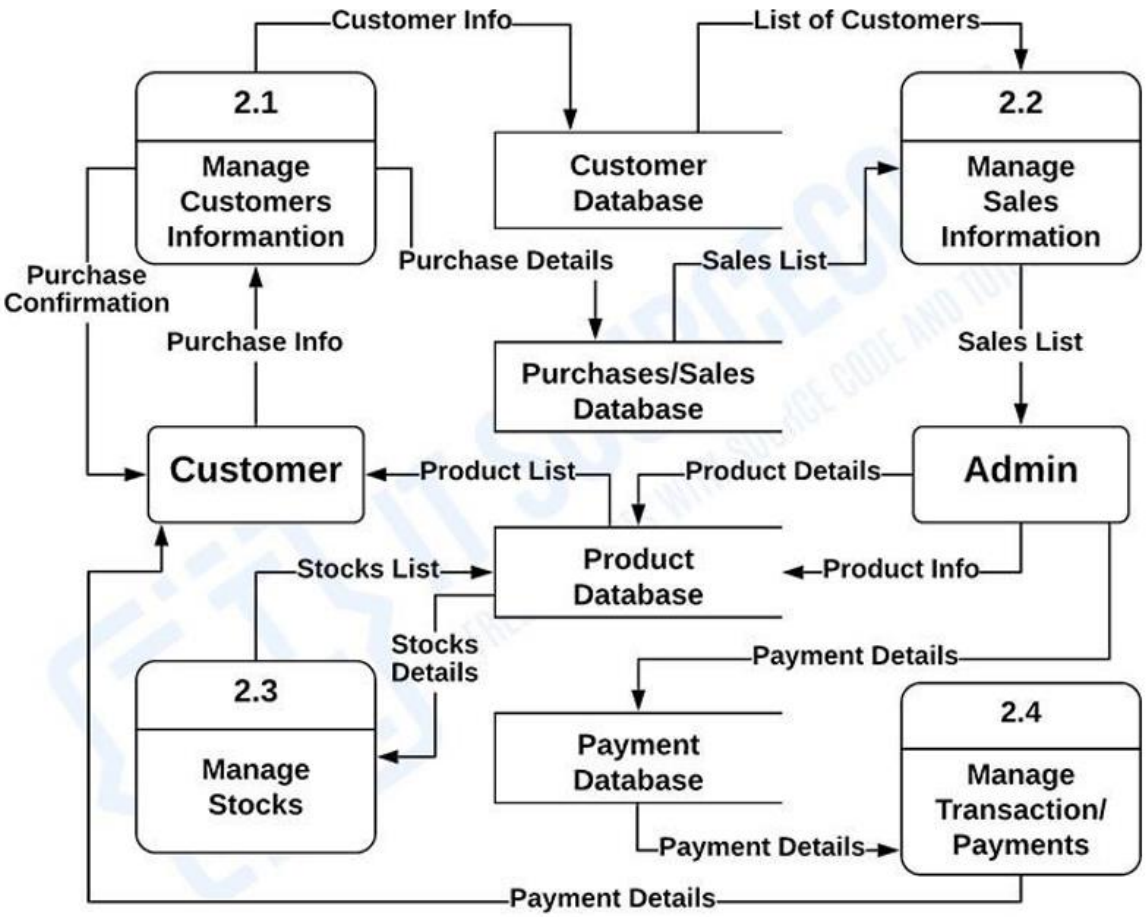
- The system must not lag, because the workers using it don't have down-time to wait for it to complete an action.
- The system must complete updating the databases, adding of recipe, ingredient, vendor and occasions successfully every time the user requests such a process.
- All the functions of the system must be available to the user every time the system is turned on.
- The calculations performed by the system must comply according to the norms set by the user and should not vary unless explicitly changed by the user
- The System must give accurate inventory status to the user continuously. Any inaccuracies are taken care by the regular confirming of the actual levels with the levels displayed in the system.
- The System must successfully add any recipe, ingredients, vendors or special occasions given by the user and provide estimations and inventory status in relevance with the newly updated entities.

5. PROJECT DESIGN

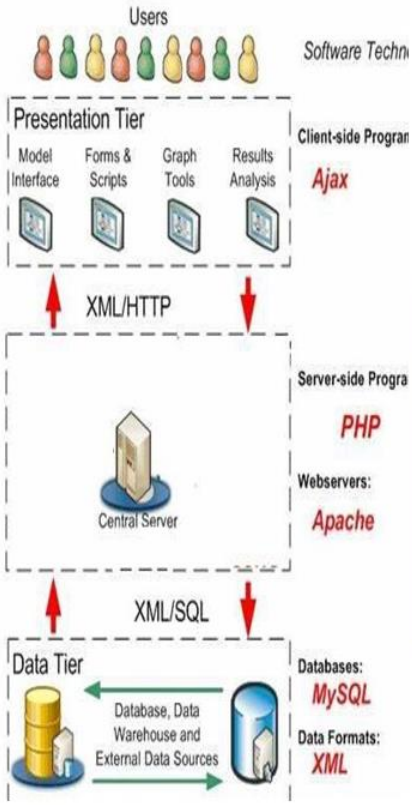
Data Flow Diagrams

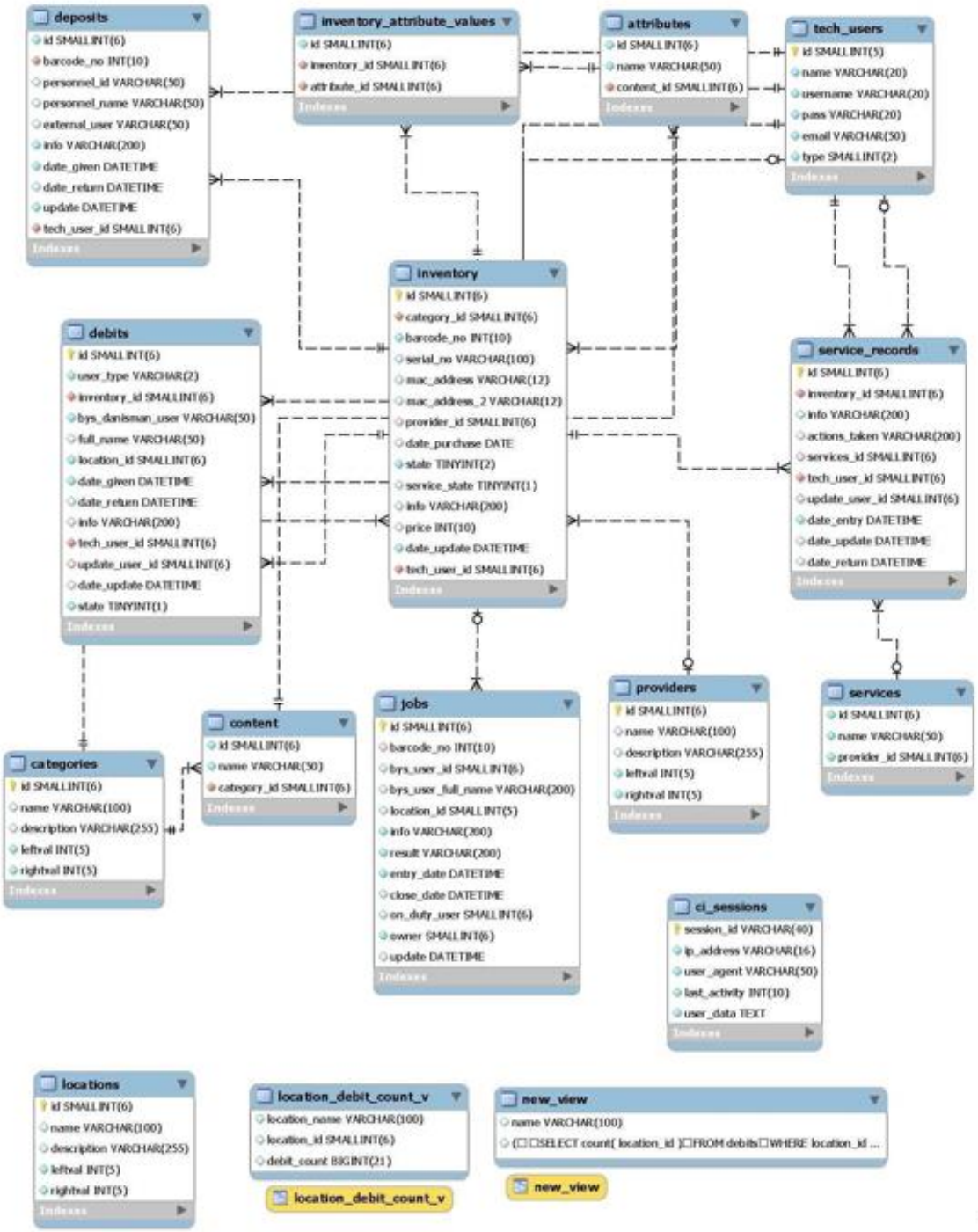





Customer Info
List of Customers
2.1
Manage Customers Informantion
Customer Database
2.2
Manage Sales Information
Purchase Confirmation
Purchase Details
Sales List
Purchase Info
Purchases/Sales Database
Sales List
Customer
Product List
Product Details
Admin
Stocks List
Product Database
Product Info
Stocks Details
Payment Details
2.3
Manage Stocks
Payment Database
2.4
Manage Transaction/ Payments
Payment Details
Payment Details

Solution & Technical Architecture




deposits
id SMALLINT(6)
barcode_no INT(10)
personnel_id VARCHAR(50)
personnel_name VARCHAR(50)
external_user VARCHAR(50)
info VARCHAR(200)
date_given DATETIME
date_return DATETIME
update DATETIME
tech_user_id SMALLINT(6)
Indexes
inventory_attribute_values
id SMALLINT(6)
inventory_id SMALLINT(6)
attribute_id SMALLINT(6)
Indexes
attributes
id SMALLINT(6)
name VARCHAR(50)
content_id SMALLINT(6)
Indexes
tech_users
id SMALLINT(5)
name VARCHAR(20)
username VARCHAR(20)
pass VARCHAR(20)
email VARCHAR(50)
type SMALLINT(2)
Indexes
inventory
id SMALLINT(6)
category_id SMALLINT(6)
barcode_no INT(10)
serial_no VARCHAR(100)
mac_address VARCHAR(12)
mac_address_2 VARCHAR(12)
provider_id SMALLINT(6)
date_purchase DATE
state TINYINT(2)
service_state TINYINT(1)
info VARCHAR(200)
price INT(10)
date_update DATETIME
tech_user_id SMALLINT(6)
Indexes
debits
id SMALLINT(6)
user_type VARCHAR(2)
inventory_id SMALLINT(6)
bys_danisman_user VARCHAR(50)
full_name VARCHAR(50)
location_id SMALLINT(6)
date_given DATETIME
date_return DATETIME
info VARCHAR(200)
tech_user_id SMALLINT(6)
update_user_id SMALLINT(6)
date_update DATETIME
state TINYINT(1)
Indexes
service_records
id SMALLINT(6)
inventory_id SMALLINT(6)
info VARCHAR(200)
actions_taken VARCHAR(200)
services_id SMALLINT(6)
tech_user_id SMALLINT(6)
update_user_id SMALLINT(6)
date_entry DATETIME
date_update DATETIME
date_return DATETIME
Indexes
content
id SMALLINT(6)
name VARCHAR(50)
category_id SMALLINT(6)
Indexes
jobs
id SMALLINT(6)
barcode_no INT(10)
bys_user_id SMALLINT(6)
bys_user_full_name VARCHAR(200)
location_id SMALLINT(5)
info VARCHAR(200)
result VARCHAR(200)
entry_date DATETIME
close_date DATETIME
on_duty_user SMALLINT(6)
owner SMALLINT(6)
update DATETIME
Indexes
providers
id SMALLINT(6)
name VARCHAR(100)
description VARCHAR(255)
leftval INT(5)
rightval INT(5)
Indexes
services
id SMALLINT(6)
name VARCHAR(50)
provider_id SMALLINT(6)
Indexes
categories
id SMALLINT(6)
name VARCHAR(100)
description VARCHAR(255)
leftval INT(5)
rightval INT(5)
Indexes
ci_sessions
session_id VARCHAR(40)
ip_address VARCHAR(16)
user_agent VARCHAR(50)
last_activity INT(10)
user_data TEXT
Indexes
locations
id SMALLINT(6)
name VARCHAR(100)
description VARCHAR(255)
leftval INT(5)
rightval INT(5)
Indexes
location_debit_count_v
location_name VARCHAR(100)
location_id SMALLINT(6)
debit_count BIGINT(21)
location_debit_count_v
new_view
name VARCHAR(100)
(SELECT count(location_id) FROM debits WHERE location_id ...
new_view

User Stories

User Type	Functional Requirement(Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Retailer	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	Medium	Sprint-1
	Login	USN-3	As a user, I can log into the application by entering email & password	I can access my account /dashboard	High	Sprint-1
	Dashboard	USN-4	As a user, I can view the stock list and suppliers list	Once I log in to the system, I can able to view the stocks	Medium	Sprint-1
	Items	USN-5	As a user, I can add the items.	I can create a new type of item	High	Sprint-2
		USN-6	As a user, I can see the items	I can be able to see the items that can be added to the inventory	Low	Sprint-2
	Inventory	USN-7	As a user, I can add the items to inventory.	I can add items to the inventory with quantity	High	Sprint-2
		USN-8	As a user, I can see the items in the inventory.	I can see the inventory items with quantity	Low	Sprint-2
	Indication	USN-9	As a user, I can be able to receive indication	I receive a notification when the stock running low	High	Sprint-3
	Location	USN-10	As a user, I can be able to see items from a particular store location	I can be able to make purchase from a particular location	Medium	Sprint-3
		USN-11	As a user, I can add a new location of my store	I can be able to add new store locations	Medium	Sprint - 3
Customer	Purchase	USN -12	As a customer, I can be able to purchase good from the particular location of the store	I can able to purchase from the store	High	Sprint - 4
Retailer & Customer	Deployment	USN-13	As a user, I can access the software in the web	I can access the software in web	High	Sprint -4

6. PROJECT PLANNING & SCHEDULING

Sprint Planning & Estimation

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 entering my email, password, and confirmingmy password.	5	High	Ashik Varghese Chris Sasan Doss Jawin Ariyumathi
Sprint-2		USN-2	As a user, I will receive confirmation email onceI have registered for the application	3	Medium	Ashik Varghese Chris Sasan Doss Jawin Ariyumathi
Sprint-4		USN-3	As a user, I can register for the application through Face book	8	Low	Ashik Varghese Chris Sasan Doss Jawin Ariyumathi
Sprint-3		USN-4	As a user, I can register for the application through Gmail	8	High	Asmir Khan Ashik Varghese Chris Sasan Doss Jawin Ariyumathi
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	5	High	Asmir Khan Ashik Varghese Jawin Ariyumathi Chris Sasan Doss
Sprint-2		USN-4	As a user, I can login into the application through Google one Tap Sign in	3	Medium	Asmir Khan Ashik Varghese Jawin Ariyumathi
Sprint-1	Dashboard	USN-5	As a user, I must be able to see my details onthe dashboard.	3	High	Asmir Khan Ashik Varghese Chris Sasan Doss
Sprint-2		USN-6	As a user, I should be able to change password whenever I prefer.	2	Medium	Asmir Khan Chris Sasan Doss Jawin Ariyumathi
Sprint-1	Inventory	USN-7	As a retailer, I should be able to alter product details in the app	2	Medium	Asmir Khan Ashik Varghese Chris Sasan Doss
Sprint-2		USN-8	As a retailer, I should be able to add or remove quantity of products in the app.	3	Medium	Asmir Khan Ashik Varghese Jawin Ariyumathi
Sprint-2		USN-9	As a retailer, I should get alert on stock shortage or unavailability.	5	Medium	Asmir Khan Ashik Varghese
Sprint-1	Order	USN-7	As a user, I should be able to order items on theapp	2	High	Ashik Varghese Chris Sasan Doss Jawin Ariyumathi
Sprint-1		USN-8	As a user, I should be able to verify and pay in a secure payment gateway	3	High	Asmir Khan Ashik Varghese Chris Sasan Doss Jawin Ariyumathi
Sprint-3		USN-9	As a user, I should be able to get the product ontime.	5	Low	Ashik Varghese Chris Sasan Doss Jawin Ariyumathi
Sprint-4	Maintenance	USN-1	As a administrator, I should be able to edit details of the users of the app.	8	High	Asmir Khan Ashik Varghese Chris Sasan Doss Jawin Ariyumathi
Sprint-4		USN-2	Termination user accounts temporarily or permanently if needed.	5	Low	Ashik Varghese Chris Sasan Doss
Sprint-2	Feedback	USN-1	As a customer care team member, I should beable to get feedback from the users.	2	High	Asmir Khan Ashik Varghese Chris Sasan Doss Jawin Ariyumathi
Sprint-3		USN-2	As a customer care team member, I should be available 24/7 to increase customer base	8	Medium	Asmir Khan Ashik Varghese Chris Sasan Doss Jawin Ariyumathi

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	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	18	6 Days	31 Oct 2022	05 Nov 2022	18	05 Nov 2022
Sprint-3	21	6 Days	07 Nov 2022	12 Nov 2022	21	12 Nov 2022
Sprint-4	21	6 Days	14 Nov 2022	19 Nov 2022	21	19 Nov 2022

Milestone And Activities

TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring the, technical papers, research publications etc.	28 SEPTEMBER 2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements	24 SEPTEMBER 2022
Ideation	List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	25 SEPTEMBER 2022
Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	23 SEPTEMBER 2022
Problem Solution Fit	Prepare problem - solution fit document.	30 SEPTEMBER 2022
Solution Architecture	Prepare solution architecture document.	28 SEPTEMBER 2022

Customer Journey	Prepare the customer journey maps to understand the user interactions & experiences with the application (entry to exit).	20 OCTOBER 2022
Functional Requirement	Prepare the functional requirement document.	8 OCTOBER 2022
Data Flow Diagrams	Draw the data flow diagrams and submit for review.	9 OCTOBER 2022
Technology Architecture	Prepare the technology architecture diagram.	10 OCTOBER 2022
Prepare Milestone & Activity List	Prepare the milestones & activity list of the project.	22 OCTOBER 2022
Project Development - Delivery of Sprint-1, 2, 3 & 4	Develop & submit the developed code by testing it.	IN PROGRESS..

7. CODING&SOLUTIONING

App.py

```
from turtle import st
from flask import Flask, render_template, request, redirect, url_for, session from markupsafe import escape

import ibm_db
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=54a2f15b-5c0f-46df-8954-
    7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32733;SECURITY=S
    SL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=lkc93724;PWD=zAzNGa6DaNk6xvle",' ','')

import smtplib, ssl

## email.mime subclasses
from email.mime.multipart import MIMEMultipart from email.mime.text import MIMEText
## The pandas library is only for generating the current date, which is not necessary for sending emails
import pandas as pd



from datetime import datetime from flask import Flask

app = Flask( name ) var_list = []

app.secret_key = 'your secret key'



@app.route('/') def home():
if not session.get("name"):
return render_template('home.html')
return render_template('home.html', session = session)



@app.route('/register') def new_student():
return render_template('Register.html')




@app.route('/addrec',methods = ['POST', 'GET']) def addrec():
msg=''
if request.method == 'POST':

fname = request.form['fname'] lname = request.form['lname'] cname = request.form['cname'] state = request.form['state'] city =
    request.form['city']
mobileno = request.form['mobileno'] emailid = request.form['emailid'] password = request.form['password'] pincode =
    request.form['pincode']
```

```
sql = "SELECT * FROM Users WHERE EMAILID =?"
stmt = ibm_db.prepare(conn, sql) ibm_db.bind_param(stmt,1,emailid) ibm_db.execute(stmt)
account = ibm_db.fetch_assoc(stmt)


if account:
msg="You are already a member, please login using your details" return render_template('Register.html', msg = msg)
else:


var_list.append(fname) var_list.append(lname) var_list.append(cname) var_list.append(state) var_list.append(city)
    var_list.append(mobileno) var_list.append(emailid) var_list.append(password) var_list.append(pincode)

html= '''<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML 1.0 Transitional //EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
    transitional.dtd"><html

xmlns="http://www.w3.org/1999/xhtml" xmlns:v="urn:schemas-microsoft-com:vml" xmlns:o="urn:schemas-microsoft-
    com:office:office"><head><!--[if gte mso 9]><xml>
<o:OfficeDocumentSettings>
<o:AllowPNG/>
<o:PixelsPerInch>96</o:PixelsPerInch>
</o:OfficeDocumentSettings>
</xml>
<![endif]-->
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta name="x-apple-disable-message-reformatting">
<!--[if !mso]><!-->
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<!--<![endif]-->
<title></title>


<style type="text/css">
@media only screen and (min-width: 620px) {
.u-row {
width: 600px !important;
}
.u-row .u-col { vertical-align: top;
}
.u-row .u-col-100 {
width: 600px !important;
}
}
```

```
@media (max-width: 620px) {
.u-row-container {
max-width: 100% !important; padding-left: 0px !important; padding-right: 0px !important;

}
.u-row .u-col {
min-width: 320px !important; max-width: 100% !important; display: block !important;
}
.u-row {
width: calc(100% - 40px) !important;
}
.u-col {
width: 100% !important;
}
.u-col>div { margin: 0 auto;
}
}


body { margin: 0;
padding: 0;
}


table, tr,
td {
vertical-align: top;
border-collapse: collapse;
}


p {
margin: 0;
}


.ie-container table,

.mso-container table { table-layout: fixed;
}


* {
line-height: inherit;
```

```
}

a[x-apple-data-detectors='true'] { color: inherit !important;
text-decoration: none !important;
}

table, td {
color: #000000;
}

#u_body a { color: #0000ee;
text-decoration: underline;
}
</style>

<!--[if !mso]><!-->
<link href="https://fonts.googleapis.com/css?family=Cabin:400,700" rel="stylesheet" type="text/css">
<!--<![endif]-->

</head>

<body class="clean-body u_body" style="margin: 0;padding: 0;-webkit-text-size-adjust: 100%;background-color: #f9f9f9;color: #000000">
<!--[if IE]><div class="ie-container"><![endif]-->
<!--[if mso]><div class="mso-container"><![endif]-->
<table id="u_body" style="border-collapse: collapse;table-layout: fixed;border-spacing: 0;mso-table-lspace: 0pt;mso-table-rspace: 0pt;vertical-align: top;min-width: 320px;Margin: 0 auto;background-color: #f9f9f9;width:100%" cellpadding="0" cellspacing="0">
<tbody>
<tr style="vertical-align: top">
<td style="word-break: break-word;border-collapse: collapse !important;vertical-align:
top">
<!--[if (mso)|(IE)]><table width="100%" cellpadding="0" cellspacing="0"
border="0"><tr><td align="center" style="background-color: #f9f9f9;"><![endif]-->

<div class="u-row-container" style="padding: 0px;background-color: transparent">
<div class="u-row" style="Margin: 0 auto;min-width: 320px;max-width: 600px;overflow-wrap: break-word;word-wrap: break-
```

```
    word;word-break: break-word;background- color: transparent;">
<div style="border-collapse: collapse;display: table;width: 100%;height: 100%;background-color: transparent;">
<!--[if (mso)|(IE)]><table width="100%" cellpadding="0" cellspacing="0" border="0"><tr><td style="padding: 0px;background-
    color: transparent;" align="center"><table cellpadding="0" cellspacing="0" border="0" style="width:600px;"><tr
    style="background-color: transparent;"><![endif]-->

<!--[if (mso)|(IE)]><td align="center" width="600" style="width: 600px;padding: 0px;border-top: 0px solid transparent;border-
    left: 0px solid transparent;border-right: 0px solid transparent;border-bottom: 0px solid transparent;" valign="top"><![endif]--
    >
<div class="u-col u-col-100" style="max-width: 320px;min-width: 600px;display: table-cell;vertical-align: top;">
<div style="height: 100%;width: 100% !important;">
<!--[if (!mso)&(!IE)]><!-->

<div style="height: 100%; padding: 0px;border-top: 0px solid transparent;border-left: 0px solid transparent;border-right: 0px
    solid transparent;border-bottom: 0px solid transparent;">
<!--<![endif]-->



<table    style="font-family:'Cabin',sans-serif;"          role="presentation" cellpadding="0" cellspacing="0" width="100%"
    border="0">
<tbody>
<tr>
<td  style="overflow-wrap:break-word;word-break:break- word;padding:10px;font-family:'Cabin',sans-serif;" align="left">




wrap: break-word;">

<div style="color: #afb0c7; line-height: 170%; text-align: center; word-



<p style="font-size: 14px; line-height: 170%;"><span  style="font-size:

14px; line-height: 23.8px;">View Email in Browser</span></p>
</div>



</td>
</tr>
</tbody>
</table>



<!--[if (!mso)&(!IE)]><!-->
</div>
<!--<![endif]-->
```

```
</div>
</div>
<!--[if (mso)|(IE)]></td><![endif]-->
<!--[if (mso)|(IE)]></tr></table></td></tr></table><![endif]-->
</div>
</div>
</div>




<div class="u-row-container" style="padding: 0px;background-color: transparent">
<div class="u-row" style="Margin: 0 auto;min-width: 320px;max-width: 600px;overflow-wrap: break-word;word-wrap: break-word;word-break: break-word;background- color: #ffffff;">
<div style="border-collapse: collapse;display: table;width: 100%;height: 100%;background-color: transparent;">
<!--[if (mso)|(IE)]><table width="100%" cellpadding="0" cellspacing="0" border="0"><tr><td style="padding: 0px;background-color: transparent;" align="center"><table cellpadding="0" cellspacing="0" border="0" style="width:600px;"><tr style="background-color: #ffffff;"><![endif]-->


<!--[if (mso)|(IE)]><td align="center" width="600" style="width: 600px;padding: 0px;border-top: 0px solid transparent;border-left: 0px solid transparent;border-right: 0px solid transparent;border-bottom: 0px solid transparent;" valign="top"><![endif]-->
<div class="u-col u-col-100" style="max-width: 320px;min-width: 600px;display: table-cell;vertical-align: top;">
<div style="height: 100%;width: 100% !important;">
<!--[if (!mso)&(!IE)]><!-->
<div style="height: 100%; padding: 0px;border-top: 0px solid transparent;border-left: 0px solid transparent;border-right: 0px solid transparent;border-bottom: 0px solid transparent;">
<!--<![endif]-->



<table style="font-family:'Cabin',sans-serif;" role="presentation" cellpadding="0" cellspacing="0" width="100%" border="0">
<tbody>
<tr>
<td style="overflow-wrap:break-word;word-break:break- word;padding:20px;font-family:'Cabin',sans-serif;" align="left">


<table width="100%" cellpadding="0" cellspacing="0" border="0">
<tr>


<td style="padding-right: 0px;padding-left: 0px;" align="center">



<img align="center" border="0" src="https://assets.unlayer.com/projects/111476/1667819721582-Secure%20login-rafiki.png" alt="Image" title="Image" style="outline: none;text-decoration: none;-ms-interpolation-mode: bicubic;clear: both;display: inline-block !important;border: none;height: auto;float: none;width: 100%;max-width: 560px;"
width="560" />
```

```
</td>
</tr>
</table>


</td>
</tr>
</tbody>
</table>


<!--[if (!mso)&(!IE)]><!-->
</div>
<!--<![endif]-->
</div>
</div>
<!--[if (mso)|(IE)]></td><![endif]-->
<!--[if (mso)|(IE)]></tr></table></td></tr></table><![endif]-->
</div>
</div>
</div>




<div class="u-row-container" style="padding: 0px;background-color: transparent">


<div class="u-row" style="Margin: 0 auto;min-width: 320px;max-width: 600px;overflow-wrap: break-word;word-wrap: break-word;word-break: break-word;background- color: #003399;">
<div style="border-collapse: collapse;display: table;width: 100%;height: 100%;background-color: transparent;">
<!--[if (mso)|(IE)]><table width="100%" cellpadding="0" cellspacing="0" border="0"><tr><td style="padding: 0px;background-color: transparent;" align="center"><table cellpadding="0" cellspacing="0" border="0" style="width:600px;"><tr style="background-color: #003399;"><![endif]-->


<!--[if (mso)|(IE)]><td align="center" width="600" style="width: 600px;padding: 0px;border-top: 0px solid transparent;border-left: 0px solid transparent;border-right: 0px solid transparent;border-bottom: 0px solid transparent;" valign="top"><![endif]-->
<div class="u-col u-col-100" style="max-width: 320px;min-width: 600px;display: table-cell;vertical-align: top;">
<div style="height: 100%;width: 100% !important;">
<!--[if (!mso)&(!IE)]><!-->
<div style="height: 100%; padding: 0px;border-top: 0px solid transparent;border-left: 0px solid transparent;border-right: 0px solid transparent;border-bottom: 0px solid transparent;">
<!--<![endif]-->
```

```
<table    style="font-family:'Cabin',sans-serif;"          role="presentation" cellpadding="0" cellspacing="0" width="100%"
     border="0">
<tbody>
<tr>
<td style="overflow-wrap:break-word;word-break:break-word;padding:40px 10px 10px;font-family:'Cabin',sans-serif;"
     align="left">

<table width="100%" cellpadding="0" cellspacing="0" border="0">
<tr>
<td style="padding-right: 0px;padding-left: 0px;" align="center">

<img      align="center"                  border="0" src="https://cdn.templates.unlayer.com/assets/1597218650916-xxxxc.png"
          alt="Image" title="Image" style="outline: none;text-decoration: none;-ms-interpolation-mode: bicubic;clear: both;display:
     inline-block !important;border: none;height: auto;float: none;width: 26%;max-width: 150.8px;"
width="150.8" />



</td>
</tr>
</table>



</td>
</tr>
</tbody>
</table>



<table    style="font-family:'Cabin',sans-serif;"          role="presentation" cellpadding="0" cellspacing="0" width="100%"
     border="0">
<tbody>
<tr>
<td  style="overflow-wrap:break-word;word-break:break- word;padding:10px;font-family:'Cabin',sans-serif;" align="left">




wrap: break-word;">

<div style="color: #e5eaf5; line-height: 140%; text-align: center; word-



<p style="font-size: 14px; line-height: 140%;"><strong>T H A N K S  F

O R S I G N I N G U P !</strong></p>
</div>
```

```
</td>
</tr>
</tbody>
</table>

<table    style="font-family:'Cabin',sans-serif;"          role="presentation" cellpadding="0" cellspacing="0" width="100%"
      border="0">
<tbody>
<tr>
<td  style="overflow-wrap:break-word;word-break:break-word;padding:0px 10px 31px;font-family:'Cabin',sans-serif;"
      align="left">



wrap: break-word;">

<div style="color: #e5eaf5; line-height: 140%; text-align: center; word-



<p style="font-size: 14px; line-height: 140%;"><span  style="font-size:

28px; line-height: 39.2px;"><strong><span style="line-height: 39.2px; font-size: 28px;">Confirm Your E-mail Address
      </span></strong>
</span>
</p>
</div>



</td>
</tr>
</tbody>
</table>



<!--[if (!mso)&(!IE)]><!-->
</div>
<!--<![endif]-->
</div>
</div>
<!--[if (mso)|(IE)]></td><![endif]-->
<!--[if (mso)|(IE)]></tr></table></td></tr></table><![endif]-->
</div>
</div>
</div>
```

```
<div class="u-row-container" style="padding: 0px;background-color: transparent">
<div class="u-row" style="Margin: 0 auto;min-width: 320px;max-width: 600px;overflow-wrap: break-word;word-wrap: break-word;word-break: break-word;background- color: #ffffff;">
<div style="border-collapse: collapse;display: table;width: 100%;height: 100%;background-color: transparent;">
<!--[if (mso)|(IE)]><table width="100%" cellpadding="0" cellspacing="0" border="0"><tr><td style="padding: 0px;background-color: transparent;" align="center"><table cellpadding="0" cellspacing="0" border="0" style="width:600px;"><tr style="background-color: #ffffff;"><![endif]-->

<!--[if (mso)|(IE)]><td align="center" width="600" style="width: 600px;padding: 0px;border-top: 0px solid transparent;border-left: 0px solid transparent;border-right: 0px solid transparent;border-bottom: 0px solid transparent;" valign="top"><![endif]-->
<div class="u-col u-col-100" style="max-width: 320px;min-width: 600px;display: table-cell;vertical-align: top;">
<div style="height: 100%;width: 100% !important;">
<!--[if (!mso)&(!IE)]><!-->
<div style="height: 100%; padding: 0px;border-top: 0px solid transparent;border-left: 0px solid transparent;border-right: 0px solid transparent;border-bottom: 0px solid transparent;">
<!--<![endif]-->



<table  style="font-family:'Cabin',sans-serif;"  role="presentation" cellpadding="0" cellspacing="0" width="100%" border="0">
<tbody>
<tr>
<td style="overflow-wrap:break-word;word-break:break-word;padding:33px 55px;font-family:'Cabin',sans-serif;" align="left">




word;">

<div style="line-height: 160%; text-align: center; word-wrap: break-



<p style="font-size: 14px; line-height: 160%;"><span  style="font-size:

22px; line-height: 35.2px;">Hi, </span></p>

<p style="font-size: 14px; line-height: 160%;"><span style="font-size: 18px; line-height: 28.8px;">You're almost ready to get started. Please click on the button below to confirm your email address and experience the awesome Inventory Management Service!</span></p>
</div>



</td>
</tr>
</tbody>
</table>
```

<table style="font-family:'Cabin',sans-serif;" role="presentation" cellpadding="0" cellspacing="0" width="100%" border="0">
<tbody>
<tr>
<td style="overflow-wrap:break-word;word-break:break- word;padding:10px;font-family:'Cabin',sans-serif;" align="left">

<!--[if mso]><style>.v-button {background: transparent
!important;}</style><![endif]-->

<!--[if mso]><v:roundrect xmlns:v="urn:schemas-microsoft-com:vml" xmlns:w="urn:schemas-microsoft-com:office:word" href="http://localhost:5000/confirm" style="height:46px; v-text-anchor:middle; width:157px;" arcsize="8.5%" stroke="f" fillcolor="#9ecbe4"><w:anchorlock/> <![endif]-->

CONFIRM

<!--[if mso]> </v:roundrect><![endif]-->

</td>
</tr>
</tbody>
</table>

<table style="font-family:'Cabin',sans-serif;" role="presentation" cellpadding="0" cellspacing="0" width="100%" border="0">
<tbody>
<tr>
<td style="overflow-wrap:break-word;word-break:break-word;padding:33px 55px 60px;font-family:'Cabin',sans-serif;" align="left">

wrap: break-word;">

<div style="color: #3598db; line-height: 160%; text-align: center; word-

```
<p style="line-height: 160%; font-size: 14px;"><span style="font-size:

18px; line-height: 28.8px;">Once again, Thanks for signing up with us!</span></p>
</div>


</td>
</tr>
</tbody>
</table>


<!--[if (!mso)&(!IE)]><!-->
</div>

<!--<![endif]-->
</div>
</div>
<!--[if (mso)|(IE)]></td><![endif]-->
<!--[if (mso)|(IE)]></tr></table></td></tr></table><![endif]-->
</div>
</div>
</div>



<!--[if (mso)|(IE)]></td></tr></table><![endif]-->
</td>
</tr>
</tbody>
</table>
<!--[if mso]></div><![endif]-->
<!--[if IE]></div><![endif]-->
</body>


</html>'''



# Set up the email addresses and password. Please replace below with your email address and password
email_from = 'padhu10a@gmail.com' epassword = 'rbjibzkssszsbrjo' email_to = emailid

# Generate today's date to be included in the email Subject date_str = pd.Timestamp.today().strftime('%Y-%m-%d')
```

```
# Create a MIMEMultipart class, and set up the From, To, Subject fields email_message = MIMEMultipart()
email_message['From'] = email_from

email_message['To'] = email_to email_message['Subject'] = f'Report email - {date_str}'

# Attach the html doc defined earlier, as a MIMEText html content type to the MIME message
email_message.attach(MIMEText(html, "html")) # Convert it as a string
email_string = email_message.as_string()


# Connect to the Gmail SMTP server and Send Email context = ssl.create_default_context()
with smtplib.SMTP_SSL("smtp.gmail.com", 465, context=context) as server: server.login(email_from, epassword)
server.sendmail(email_from, email_to, email_string) return render_template('notify.html')




@app.route('/confirm') def confirnation():
insert_sql = "INSERT INTO Users (FIRSTNAME, LASTNAME, COMPANYNAME, STATE, CITY, MOBILENO, EMAILID, PASSWORD,
    PINCODE) VALUES (?,?,?,?,?,?,?,?,?)"
prep_stmt = ibm_db.prepare(conn, insert_sql) ibm_db.bind_param(prep_stmt, 1, var_list[0])
ibm_db.bind_param(prep_stmt, 2, var_list[1])
ibm_db.bind_param(prep_stmt, 3, var_list[2])
ibm_db.bind_param(prep_stmt, 4, var_list[3])
ibm_db.bind_param(prep_stmt, 5, var_list[4])
ibm_db.bind_param(prep_stmt, 6, var_list[5])
ibm_db.bind_param(prep_stmt, 7, var_list[6])
ibm_db.bind_param(prep_stmt, 8, var_list[7])
ibm_db.bind_param(prep_stmt, 9, var_list[8]) ibm_db.execute(prep_stmt)
var_list.clear()

return render_template('confirm.html')



@app.route('/login') def login():
return render_template('Login.html')



@app.route('/loginrec', methods =['POST', 'GET']) def loginrec():
msg = ''
if request.method == 'POST' and 'email' in request.form and 'password' in request.form: email = request.form['email']
password = request.form['password']
```

```
sql = "SELECT * FROM Users WHERE EMAILID =? AND PASSWORD =?"
stmt = ibm_db.prepare(conn, sql) ibm_db.bind_param(stmt,1,email) ibm_db.bind_param(stmt,2,password) ibm_db.execute(stmt)
account = ibm_db.fetch_assoc(stmt)


if account: session['loggedin'] = True session['id'] = account['ID']
session['email'] = account['EMAILID'] session['name'] = account['FIRSTNAME']

return render_template('dashboard/dashboard.html') else:
msg = 'Incorrect email / password !'
return render_template('login.html', msg = msg)


@app.route('/dashboard') def dashboard():

if session['loggedin'] == True:
return render_template('dashboard/dashboard.html') else:
return redirect(url_for('home'))


@app.route('/addproduct') def addproduct():
if session['loggedin'] == True:
return render_template('dashboard/addproduct.html') else:
return redirect(url_for('home'))


@app.route('/movement') def movement():
if session['loggedin'] == True:
products = []
sql = "SELECT * FROM Products WHERE HOLDERNAME = ?"
prep_stmt = ibm_db.prepare(conn, sql) ibm_db.bind_param(prep_stmt, 1, session['name']) ibm_db.execute(prep_stmt)
dictionary = ibm_db.fetch_both(prep_stmt) while dictionary != False:
# print ("The Name is : ", dictionary) products.append(dictionary)
dictionary = ibm_db.fetch_both(prep_stmt)


if products:
return render_template("dashboard/movement.html", products = products , session = session)
else:
return render_template("dashboard/movement.html") else:
return redirect(url_for('home'))

@app.route('/moveproc',methods = ['POST', 'GET']) def moveproc():
if request.method == 'POST': pname = request.form['pname']
```

```
quantityout = request.form['quantityout'] tow = request.form['to']

insert_sql = "UPDATE products SET QUANTITYOUT = ?, TO = ? WHERE PRODUCTNAME = ? AND HOLDERNAME = ?;"
prep_stmt = ibm_db.prepare(conn, insert_sql) ibm_db.bind_param(prep_stmt, 1,quantityout) ibm_db.bind_param(prep_stmt, 2,
    tow)
ibm_db.bind_param(prep_stmt, 3, pname)
ibm_db.bind_param(prep_stmt, 4, session['name']) ibm_db.execute(prep_stmt)
select_sql="SELECT QUANTITYIN from PRODUCTS WHERE PRODUCTNAME = ?
AND HOLDERNAME = ?;"
pre_stmt = ibm_db.prepare(conn, select_sql) ibm_db.bind_param(pre_stmt, 1, pname)
ibm_db.bind_param(pre_stmt, 2, session['name']) ibm_db.execute(pre_stmt)
outofstock = ibm_db.fetch_both(pre_stmt)
if outofstock['QUANTITYIN'] <= int(quantityout):



html= ''' <!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML 1.0 Transitional //EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
    transitional.dtd">
<html    xmlns="http://www.w3.org/1999/xhtml" xmlns:v="urn:schemas-microsoft-com:vml" xmlns:o="urn:schemas-microsoft-
    com:office:office">

<head>
<!--[if gte mso 9]>
<xml>
<o:OfficeDocumentSettings>
<o:AllowPNG/>

<o:PixelsPerInch>96</o:PixelsPerInch>
</o:OfficeDocumentSettings>
</xml>
<![endif]-->
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta name="x-apple-disable-message-reformatting">
<!--[if !mso]><!-->
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<!--<![endif]-->
<title></title>



<style type="text/css">
@media only screen and (min-width: 620px) {
.u-row {
width: 600px !important;
}
.u-row .u-col { vertical-align: top;
}
```

```
.u-row .u-col-100 {
width: 600px !important;
}
}


@media (max-width: 620px) {
.u-row-container {
max-width: 100% !important; padding-left: 0px !important; padding-right: 0px !important;
}
.u-row .u-col {
min-width: 320px !important; max-width: 100% !important;

display: block !important;
}
.u-row {
width: calc(100% - 40px) !important;
}
.u-col {
width: 100% !important;
}
.u-col>div { margin: 0 auto;
}
}


body { margin: 0;
padding: 0;
}


table, tr,
td {
vertical-align: top;
border-collapse: collapse;
}


p {
margin: 0;
}


.ie-container table,
.mso-container table { table-layout: fixed;
```

```
}

* {
line-height: inherit;
}


a[x-apple-data-detectors='true'] { color: inherit !important;
text-decoration: none !important;
}


table, td {
color: #000000;
}


@media (max-width: 480px) { #u_column_3 .v-col-background-color { background-color: #3598db !important;
}
}
</style>




<!--[if !mso]><!-->
<link      href="https://fonts.googleapis.com/css?family=Cabin:400,700"        rel="stylesheet" type="text/css">
<!--<![endif]-->


</head>


<body class="clean-body u_body" style="margin: 0;padding: 0;-webkit-text-size-adjust: 100%;background-color: #f9f9f9;color:
      #000000">
<!--[if IE]><div class="ie-container"><![endif]-->
<!--[if mso]><div class="mso-container"><![endif]-->

<table style="border-collapse: collapse;table-layout: fixed;border-spacing: 0;mso-table- lspace: 0pt;mso-table-rspace: 0pt;vertical-
      align: top;min-width: 320px;Margin: 0 auto;background-color: #f9f9f9;width:100%" cellpadding="0" cellspacing="0">
<tbody>
<tr style="vertical-align: top">
<td style="word-break: break-word;border-collapse: collapse !important;vertical-align:
top">
<!--[if     (mso)|(IE)]><table           width="100%"    cellpadding="0"  cellspacing="0"
```

```
border="0"><tr><td align="center" style="background-color: #f9f9f9;"><![endif]-->



<div class="u-row-container" style="padding: 0px;background-color: transparent">
<div class="u-row" style="Margin: 0 auto;min-width: 320px;max-width: 600px;overflow-wrap: break-word;word-wrap: break-word;word-break: break-word;background- color: #ffffff;">
<div style="border-collapse: collapse;display: table;width: 100%;height: 100%;background-color: transparent;">
<!--[if (mso)|(IE)]><table width="100%" cellpadding="0" cellspacing="0" border="0"><tr><td style="padding: 0px;background-color: transparent;" align="center"><table cellpadding="0" cellspacing="0" border="0" style="width:600px;"><tr style="background-color: #ffffff;"><![endif]-->

<!--[if (mso)|(IE)]><td align="center" width="600" class="v-col-background-color" style="width: 600px;padding: 0px;border-top: 0px solid transparent;border-left: 0px solid transparent;border-right: 0px solid transparent;border-bottom: 0px solid transparent;" valign="top"><![endif]-->
<div class="u-col u-col-100" style="max-width: 320px;min-width: 600px;display: table-cell;vertical-align: top;">
<div class="v-col-background-color" style="height: 100%;width: 100%

!important;">



<!--[if (!mso)&(!IE)]><!-->
<div style="height: 100%; padding: 0px;border-top: 0px solid

transparent;border-left: 0px solid transparent;border-right: 0px solid transparent;border-bottom: 0px solid transparent;">

<!--<![endif]-->



<table style="font-family:'Cabin',sans-serif;" role="presentation" cellpadding="0" cellspacing="0" width="100%" border="0">
<tbody>
<tr>
<td style="overflow-wrap:break-word;word-break:break- word;padding:0px;font-family:'Cabin',sans-serif;" align="left">

<table width="100%" cellpadding="0" cellspacing="0" border="0">
<tr>
<td style="padding-right: 0px;padding-left: 0px;" align="right">



<img align="right" border="0" src="https://assets.unlayer.com/projects/111476/1668447888843-Warning-cuate.png" alt="Image" title="Image" style="outline: none;text-decoration: none;-ms-interpolation-mode: bicubic;clear: both;display: inline-block !important;border: none;height: auto;float: none;width: 100%;max-width: 600px;"
width="600" />
```

```
</td>
</tr>
</table>


</td>
</tr>
</tbody>
</table>


<!--[if (!mso)&(!IE)]><!-->
</div>
<!--<![endif]-->
</div>
</div>

<!--[if (mso)|(IE)]></td><![endif]-->
<!--[if (mso)|(IE)]></tr></table></td></tr></table><![endif]-->
</div>
</div>
</div>




<div class="u-row-container" style="padding: 0px;background-color: transparent">
<div class="u-row" style="Margin: 0 auto;min-width: 320px;max-width: 600px;overflow-wrap: break-word;word-wrap: break-word;word-break: break-word;background- color: #003399;">
<div style="border-collapse: collapse;display: table;width: 100%;height: 100%;background-color: transparent;">
<!--[if (mso)|(IE)]><table width="100%" cellpadding="0" cellspacing="0" border="0"><tr><td style="padding: 0px;background-color: transparent;" align="center"><table cellpadding="0" cellspacing="0" border="0" style="width:600px;"><tr style="background-color: #003399;"><![endif]-->

<!--[if (mso)|(IE)]><td align="center" width="600" class="v-col-background-color" style="background-color: #3598db;width: 600px;padding: 0px;border-top: 0px solid transparent;border-left: 0px solid transparent;border-right: 0px solid transparent;border-bottom: 0px solid transparent;" valign="top"><![endif]-->
<div id="u_column_3" class="u-col u-col-100" style="max-width: 320px;min-width: 600px;display: table-cell;vertical-align: top;">
<div class="v-col-background-color" style="background-color: #3598db;height: 100%;width: 100% !important;">
<!--[if (!mso)&(!IE)]><!-->
<div style="height: 100%; padding: 0px;border-top: 0px solid transparent;border-left: 0px solid transparent;border-right: 0px solid transparent;border-bottom: 0px solid transparent;">
<!--<![endif]-->

<table    style="font-family:'Cabin',sans-serif;"          role="presentation" cellpadding="0" cellspacing="0" width="100%" border="0">
```

```
<tbody>
<tr>
<td style="overflow-wrap:break-word;word-break:break-word;padding:40px 10px 10px;font-family:'Cabin',sans-serif;"
    align="left">

<table width="100%" cellpadding="0" cellspacing="0" border="0">
<tr>
<td style="padding-right: 0px;padding-left: 0px;" align="center">



<img                align="center"                border="0" src="https://assets.unlayer.com/projects/111476/1668448081560-
    vecteezy_warning-icon-png- transparent_9663941_592.png" alt="Image" title="Image" style="outline: none;text-decoration:
    none;-ms-interpolation-mode:          bicubic;clear:     both;display:     inline-block     !important;border: none;height:
    auto;float: none;width: 30%;max-width: 174px;"
width="174" />



</td>
</tr>
</table>



</td>
</tr>
</tbody>
</table>



<table    style="font-family:'Cabin',sans-serif;"         role="presentation" cellpadding="0" cellspacing="0" width="100%"
    border="0">
<tbody>
<tr>
<td  style="overflow-wrap:break-word;word-break:break- word;padding:10px;font-family:'Cabin',sans-serif;" align="left">



wrap: break-word;">

<div style="color: #e5eaf5; line-height: 140%; text-align: center; word-



<p  style="font-size:  14px;  line-height:  140%;"><span  style="color:

#000000; font-size: 20px; line-height: 28px;">WARNING</span></p>
</div>
```

```
</td>
</tr>
</tbody>
</table>


<table    style="font-family:'Cabin',sans-serif;"          role="presentation" cellpadding="0" cellspacing="0" width="100%"
    border="0">
<tbody>
<tr>
<td  style="overflow-wrap:break-word;word-break:break-word;padding:0px 10px 31px;font-family:'Cabin',sans-serif;"
    align="left">



wrap: break-word;">

<div style="color: #e5eaf5; line-height: 140%; text-align: center; word-


<p style="font-size: 14px; line-height: 140%;"><span  style="font-size:

24px; line-height: 33.6px;"><strong>You are <span style="color: #000000; font-size: 24px; line- height: 33.6px;">Out of Stock
    !</span></strong>
</span>
</p>
</div>


</td>
</tr>
</tbody>
</table>


<!--[if (!mso)&(!IE)]><!-->
</div>

<!--<![endif]-->
</div>
</div>
<!--[if (mso)|(IE)]></td><![endif]-->
<!--[if (mso)|(IE)]></tr></table></td></tr></table><![endif]-->
</div>
</div>
</div>
```

```
<div class="u-row-container" style="padding: 0px;background-color: transparent">
<div class="u-row" style="Margin: 0 auto;min-width: 320px;max-width: 600px;overflow-wrap: break-word;word-wrap: break-word;word-break: break-word;background- color: #ffffff;">
<div style="border-collapse: collapse;display: table;width: 100%;height: 100%;background-color: transparent;">
<!--[if (mso)|(IE)]><table width="100%" cellpadding="0" cellspacing="0" border="0"><tr><td style="padding: 0px;background-color: transparent;" align="center"><table cellpadding="0" cellspacing="0" border="0" style="width:600px;"><tr style="background-color: #ffffff;"><![endif]-->

<!--[if (mso)|(IE)]><td align="center" width="600" class="v-col-background-color" style="background-color: #000000;width: 600px;padding: 0px;border-top: 0px solid transparent;border-left: 0px solid transparent;border-right: 0px solid transparent;border-bottom: 0px solid transparent;" valign="top"><![endif]-->
<div class="u-col u-col-100" style="max-width: 320px;min-width: 600px;display: table-cell;vertical-align: top;">
<div class="v-col-background-color" style="background-color: #000000;height: 100%;width: 100% !important;">
<!--[if (!mso)&(!IE)]><!-->
<div style="height: 100%; padding: 0px;border-top: 0px solid transparent;border-left: 0px solid transparent;border-right: 0px solid transparent;border-bottom: 0px solid transparent;">

<!--<![endif]-->



<table    style="font-family:'Cabin',sans-serif;"          role="presentation" cellpadding="0" cellspacing="0" width="100%" border="0">
<tbody>
<tr>
<td style="overflow-wrap:break-word;word-break:break-word;padding:33px 55px;font-family:'Cabin',sans-serif;" align="left">




wrap: break-word;">

<div style="color: #000000; line-height: 160%; text-align: center; word-



<p style="font-size: 14px; line-height: 160%;"><span  style="font-size:

18px; line-height: 28.8px; color: #ecf0f1;">Please order <span style="color: #3598db; font-size: 18px; line-height: 28.8px;">new stocks </span>to get rid of the <span style="color: #3598db; font-size: 18px; line-height: 28.8px;">out-of-stock</span>.</span>
</p>
</div>



</td>
```

```
</tr>
</tbody>
</table>


<table    style="font-family:'Cabin',sans-serif;"          role="presentation" cellpadding="0" cellspacing="0" width="100%"
    border="0">
<tbody>
<tr>
<td style="overflow-wrap:break-word;word-break:break-word;padding:33px 55px 60px;font-family:'Cabin',sans-serif;"
    align="left">

<div style="color: #3598db; line-height: 160%; text-align: center; word-
wrap: break-word;">

<p style="line-height: 160%; font-size: 14px; text-align: center;"><span style="font-size: 18px; line-height: 28.8px; color:
    #3598db;">Post queries in the Contact Support for further clearance!</span></p>
<p style="line-height: 160%; font-size: 14px; text-align: center;"> </p>
<p style="line-height: 160%; font-size: 14px; text-align: center;"><span style="font-size: 18px; line-height: 28.8px; color:
    #3598db;">Thank you!</span></p>
</div>



</td>
</tr>
</tbody>
</table>



<!--[if (!mso)&(!IE)]><!-->
</div>
<!--<![endif]-->
</div>
</div>
<!--[if (mso)|(IE)]></td><![endif]-->
<!--[if (mso)|(IE)]></tr></table></td></tr></table><![endif]-->
</div>
</div>
</div>



<!--[if (mso)|(IE)]></td></tr></table><![endif]-->
</td>
</tr>
</tbody>
```

```
</table>
<!--[if mso]></div><![endif]-->
<!--[if IE]></div><![endif]-->
</body>

</html>


'''



# Set up the email addresses and password. Please replace below with your email address and password
email_from = 'padhu10a@gmail.com' epassword = 'rbjibzkssszsbrjo' email_to = session['email']

# Generate today's date to be included in the email Subject date_str = pd.Timestamp.today().strftime('%d-%m-%Y')

# Create a MIMEMultipart class, and set up the From, To, Subject fields email_message = MIMEMultipart()
email_message['From'] = email_from email_message['To'] = email_to
email_message['Subject'] = f'Warning!!! {pname} - Out Of Stock - {date_str}'




message

# Attach the html doc defined earlier, as a MIMEText html content type to the MIME


email_message.attach(MIMEText(html, "html")) # Convert it as a string
email_string = email_message.as_string()


# Connect to the Gmail SMTP server and Send Email context = ssl.create_default_context()
with smtplib.SMTP_SSL("smtp.gmail.com", 465, context=context) as server: server.login(email_from, epassword)
server.sendmail(email_from, email_to, email_string) products = []
sql = "SELECT * FROM Products WHERE HOLDERNAME = ?"
prep_stmt = ibm_db.prepare(conn, sql)

ibm_db.bind_param(prep_stmt, 1, session['name']) ibm_db.execute(prep_stmt)
dictionary = ibm_db.fetch_both(prep_stmt) while dictionary != False:
# print ("The Name is : ", dictionary) products.append(dictionary)
dictionary = ibm_db.fetch_both(prep_stmt)
return render_template('dashboard/movement.html', msg = "Product movement noted!", products = products)
else:
```

```
products = []
sql = "SELECT * FROM Products WHERE HOLDERNAME = ?"
prep_stmt = ibm_db.prepare(conn, sql) ibm_db.bind_param(prep_stmt, 1, session['name']) ibm_db.execute(prep_stmt)
dictionary = ibm_db.fetch_both(prep_stmt) while dictionary != False:
# print ("The Name is : ", dictionary) products.append(dictionary)
dictionary = ibm_db.fetch_both(prep_stmt)
return render_template('dashboard/movement.html', msg = "Product movement noted!", products = products)



@app.route('/report') def report():
if session['loggedin'] == True:
products = [] stockonhand = []
sql = "SELECT * FROM Products WHERE HOLDERNAME = ?"
prep_stmt = ibm_db.prepare(conn, sql) ibm_db.bind_param(prep_stmt, 1, session['name']) ibm_db.execute(prep_stmt)

dictionary = ibm_db.fetch_both(prep_stmt) while dictionary != False:
# print ("The Name is : ", dictionary) products.append(dictionary)
dictionary = ibm_db.fetch_both(prep_stmt)


for i in products:
calc = int((i['QUANTITYIN'])) - int(i['QUANTITYOUT'])
stockonhand.append(str(calc))


return render_template('dashboard/report.html', row_row1 =zip(products,stockonhand)) else:
return redirect(url_for('home'))


@app.route('/stockupdate') def stock():
if session['loggedin'] == True:
products = []
sql = "SELECT * FROM Products WHERE HOLDERNAME = ?"
prep_stmt = ibm_db.prepare(conn, sql) ibm_db.bind_param(prep_stmt, 1, session['name']) ibm_db.execute(prep_stmt)
dictionary = ibm_db.fetch_both(prep_stmt) while dictionary != False:
# print ("The Name is : ", dictionary) products.append(dictionary)
dictionary = ibm_db.fetch_both(prep_stmt)


if products:
return render_template("dashboard/stockupdate.html", products = products , session = session)
else:
return render_template("dashboard/stockupdate.html") else:
```

```
return redirect(url_for('home'))



@app.route('/proc_delete', methods = ['POST', 'GET']) def proc_delete():
id = request.args.get('pid')
delete_sql = "DELETE FROM products WHERE ID = ? AND HOLDERNAME = ?;" prep_stmt = ibm_db.prepare(conn, delete_sql)
ibm_db.bind_param(prep_stmt, 1, id)
ibm_db.bind_param(prep_stmt, 2, session['name']) ibm_db.execute(prep_stmt)


products = []
sql = "SELECT * FROM Products WHERE HOLDERNAME = ?"
prep_stmt = ibm_db.prepare(conn, sql) ibm_db.bind_param(prep_stmt, 1, session['name']) ibm_db.execute(prep_stmt)
dictionary = ibm_db.fetch_both(prep_stmt) while dictionary != False:
# print ("The Name is : ", dictionary) products.append(dictionary)
dictionary = ibm_db.fetch_both(prep_stmt)
return render_template('dashboard/stockupdate.html', msg='Product successfully deleted!' , products = products)

@app.route('/proc_update', methods = ['POST', 'GET']) def proc_update():
if request.method == 'POST':
pname = request.form['pname'] quantityin = request.form['quantityin'] pid = request.form['pid']
update_sql = "UPDATE products SET PRODUCTNAME = ?, QUANTITYIN = ?
WHERE ID = ? AND HOLDERNAME = ?;"

prep_stmt = ibm_db.prepare(conn, update_sql) ibm_db.bind_param(prep_stmt, 1, pname)
ibm_db.bind_param(prep_stmt, 2, quantityin)
ibm_db.bind_param(prep_stmt, 3, pid)
ibm_db.bind_param(prep_stmt, 4, session['name']) ibm_db.execute(prep_stmt)


products = []
sql = "SELECT * FROM Products WHERE HOLDERNAME = ?"
prep_stmt = ibm_db.prepare(conn, sql) ibm_db.bind_param(prep_stmt, 1, session['name']) ibm_db.execute(prep_stmt)
dictionary = ibm_db.fetch_both(prep_stmt) while dictionary != False:
# print ("The Name is : ", dictionary) products.append(dictionary)
dictionary = ibm_db.fetch_both(prep_stmt)
return render_template('dashboard/stockupdate.html', msg='Product successfully updated!' , products = products)

@app.route('/addproc',methods = ['POST', 'GET']) def addproc():
if request.method == 'POST': pname = request.form['pname'] quantity = request.form['quantity'] the_time = datetime.now()
the_time = the_time.replace(second=0, microsecond=0)
```

```
sql = "SELECT * FROM Products WHERE HOLDERNAME =?"
stmt = ibm_db.prepare(conn, sql) ibm_db.bind_param(stmt,1,session['name']) ibm_db.execute(stmt)
product = ibm_db.fetch_assoc(stmt)

if product:


if product['PRODUCTNAME']==pname:


return render_template('dashboard/addproduct.html', msg="Product already added!
Add a new product.")
else:
sql ="INSERT INTO Products (PRODUCTNAME,QUANTITYIN,QUANTITYOUT,TO,DATE,HOLDERNAME) VALUES (?,?,?,?,?,?);"
prep_stmt = ibm_db.prepare(conn, sql) ibm_db.bind_param(prep_stmt, 1, pname)
ibm_db.bind_param(prep_stmt, 2, quantity)
ibm_db.bind_param(prep_stmt, 3, '0')
ibm_db.bind_param(prep_stmt, 4, '')
ibm_db.bind_param(prep_stmt, 5, str(the_time))
ibm_db.bind_param(prep_stmt, 6, session['name']) ibm_db.execute(prep_stmt)
return render_template('dashboard/addproduct.html', msg="Product added") else:
sql ="INSERT INTO Products (PRODUCTNAME,QUANTITYIN,QUANTITYOUT,TO,DATE,HOLDERNAME) VALUES (?,?,?,?,?,?);"
prep_stmt = ibm_db.prepare(conn, sql) ibm_db.bind_param(prep_stmt, 1, pname)
ibm_db.bind_param(prep_stmt, 2, quantity)
ibm_db.bind_param(prep_stmt, 3, '0')
ibm_db.bind_param(prep_stmt, 4, '')
ibm_db.bind_param(prep_stmt, 5, str(the_time))
ibm_db.bind_param(prep_stmt, 6, session['name']) ibm_db.execute(prep_stmt)
return render_template('dashboard/addproduct.html', msg="Product added")

@app.route('/productlist') def productlist():
if session['loggedin'] == True:
products = []
sql = "SELECT * FROM Products WHERE HOLDERNAME = ?"
prep_stmt = ibm_db.prepare(conn, sql) ibm_db.bind_param(prep_stmt, 1, session['name']) ibm_db.execute(prep_stmt)
dictionary = ibm_db.fetch_both(prep_stmt) while dictionary != False:
# print ("The Name is : ", dictionary) products.append(dictionary)
dictionary = ibm_db.fetch_both(prep_stmt)


if products:
return render_template("dashboard/productlist.html", products =  products , session = session)
else:
return render_template("dashboard/productlist.html") else:
return redirect(url_for('home'))
```

```
@app.route('/contactsupport') def contactsupport():
if session['loggedin'] == True:
return render_template('dashboard/contactsupport.html') else:
return redirect(url_for('home'))



@app.route('/contactsup', methods = ['POST','GET']) def contactsup():
if request.method == 'POST': name = request.form['name']

mobileno = request.form['mobileno'] emailid = request.form['emailid'] query = request.form['query']

html = "<h1>Query from, </h1><br/><b>Name: </b>"+name+"<br/><b>Email ID:
</b>"+emailid+"<br/><b>Contact no: </b>"+mobileno+"<br/><b>Query: </b><b>"+query+"</b>"


# Set up the email addresses and password. Please replace below with your email address and password
email_from = 'padhu10a@gmail.com' epassword = 'rbjibzkssszsbrjo' email_to = 'imsa3258@gmail.com'

# Generate today's date to be included in the email Subject date_str = pd.Timestamp.today().strftime('%Y-%m-%d')

# Create a MIMEMultipart class, and set up the From, To, Subject fields email_message = MIMEMultipart()
email_message['From'] = email_from email_message['To'] = email_to email_message['Subject'] = f'Query email - {date_str}'

# Attach the html doc defined earlier, as a MIMEText html content type to the MIME message
email_message.attach(MIMEText(html, "html")) # Convert it as a string
email_string = email_message.as_string()



# Connect to the Gmail SMTP server and Send Email context = ssl.create_default_context()
with smtplib.SMTP_SSL("smtp.gmail.com", 465, context=context) as server: server.login(email_from, epassword)
server.sendmail(email_from, email_to, email_string)

return render_template('dashboard/contactsupport.html', msg = "We have mailed your query to our Support team! Soon they will
    reach you.")

@app.route('/feedback') def feedback():
if session['loggedin'] == True:
return render_template('dashboard/feedback.html') else:
return redirect(url_for('home'))



@app.route('/feedbackadd', methods = ['POST','GET']) def feedbackadd():
```

```
if request.method == 'POST': interface = request.form['interface']
availability = request.form['availability'] userfriendly = request.form['userfriendly'] chatbot = request.form['chatbot'] suggest = request.form['suggest']

sql = "SELECT * FROM Feedback WHERE NAME =?" stmt = ibm_db.prepare(conn, sql) ibm_db.bind_param(stmt,1,session['name']) ibm_db.execute(stmt)
account = ibm_db.fetch_assoc(stmt)



if account:
return render_template('dashboard/feedback.html', msg = "Your feedback was submitted already.")
else:
ins_sql = "INSERT INTO Feedback (interface,availability,userfriendly,chatbot,suggest,name) VALUES (?,?,?,?,?,?);"
prep_stmt = ibm_db.prepare(conn, ins_sql) ibm_db.bind_param(prep_stmt, 1, interface)
ibm_db.bind_param(prep_stmt, 2, availability)

ibm_db.bind_param(prep_stmt, 3, userfriendly)
ibm_db.bind_param(prep_stmt, 4, chatbot)
ibm_db.bind_param(prep_stmt, 5, suggest)
ibm_db.bind_param(prep_stmt, 6, session['name']) ibm_db.execute(prep_stmt)

return render_template('dashboard/feedback.html', msg = "Your feedback was submitted.")

@app.route('/logout') def logout():
session['loggedin'] = False session.pop('id', None) session.pop('email', None) session.pop('name', None) return redirect(url_for('home'))

if name == " main ":
app.run(debug=True, host="0.0.0.0", port=5000)
```

Dockerfile

```
FROM python:3.10.6 WORKDIR /app
COPY requirements.txt ./
RUN pip install -r requirements.txt COPY . .
EXPOSE 5000
CMD ["python","./app.py"]
```

requirements.txt

```
flask ibm_db pandas
```

flask_service.yaml:

apiVersion: v1 kind: Service metadata:
name: flask-app-service spec:
type: NodePort ports:
- port: 5000 selector:
app: ims-final

flask_ingress.yaml:

apiVersion: networking.k8s.io/v1 kind: Ingress
metadata:
name: flask-app-ingress annotations: kubernetes.io/ingress.class: nginx
nginx.ingress.kubernetes.io/ssl-redirect: "false"

spec:
ingressClassName: nginx rules:
- http:
paths:

- backend: service:
name: flask-app-service port:
number: 5000 path: /
pathType: Prefix

ibm_deployment.yaml:

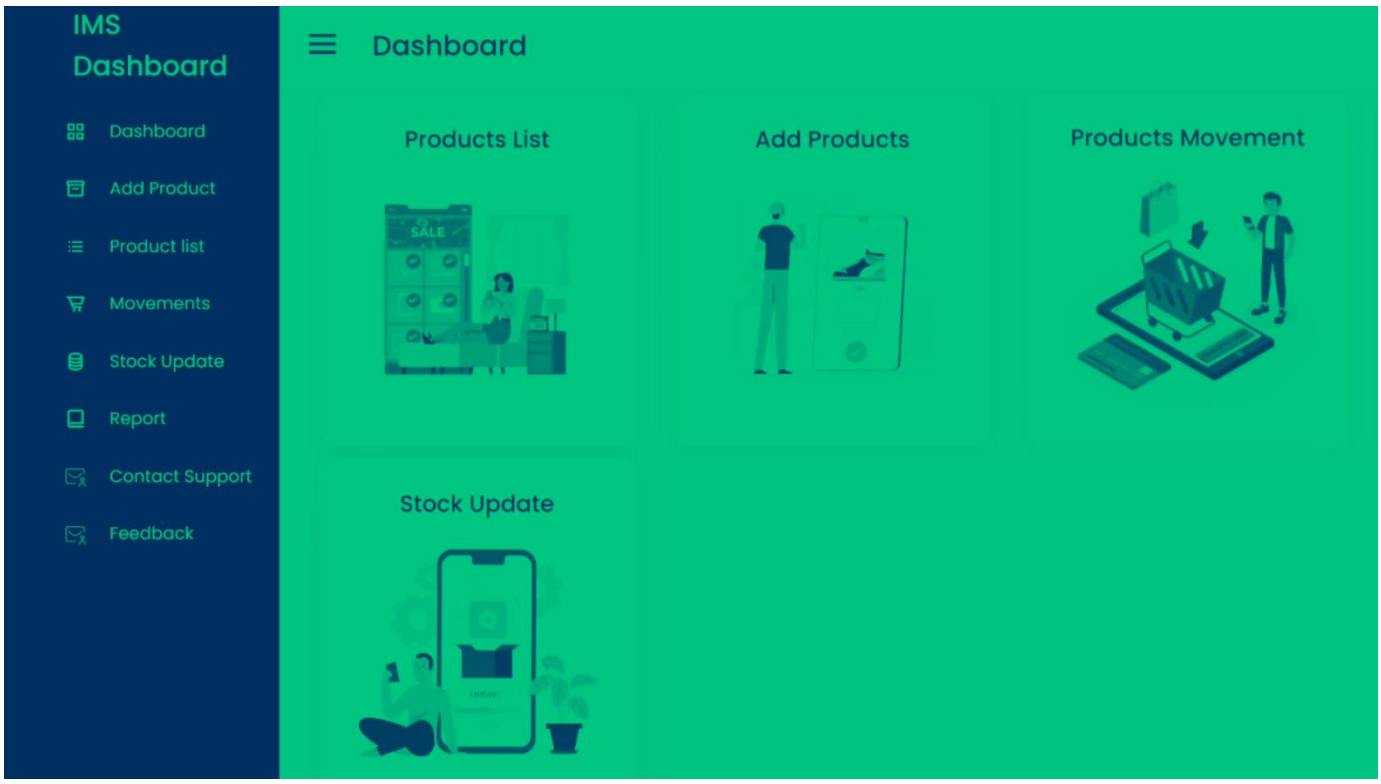
apiVersion: apps/v1 kind: Deployment metadata:
name: ims-final

spec:
replicas: 3 selector:
matchLabels:
app: ims-final template:
metadata:

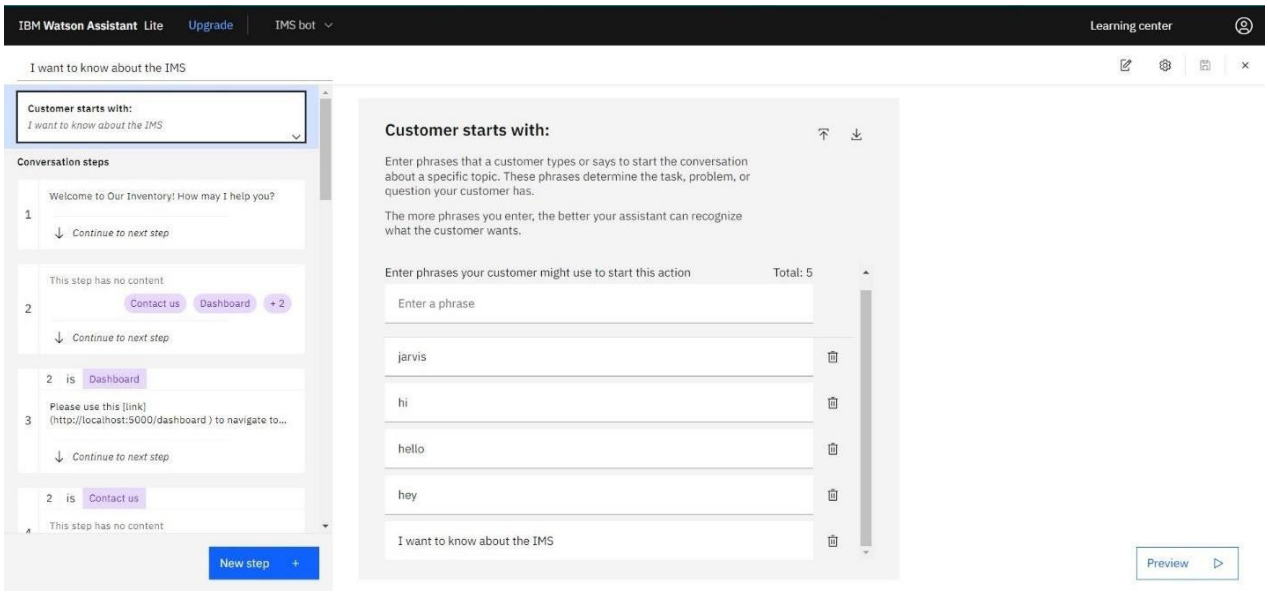
labels:
app: ims-final

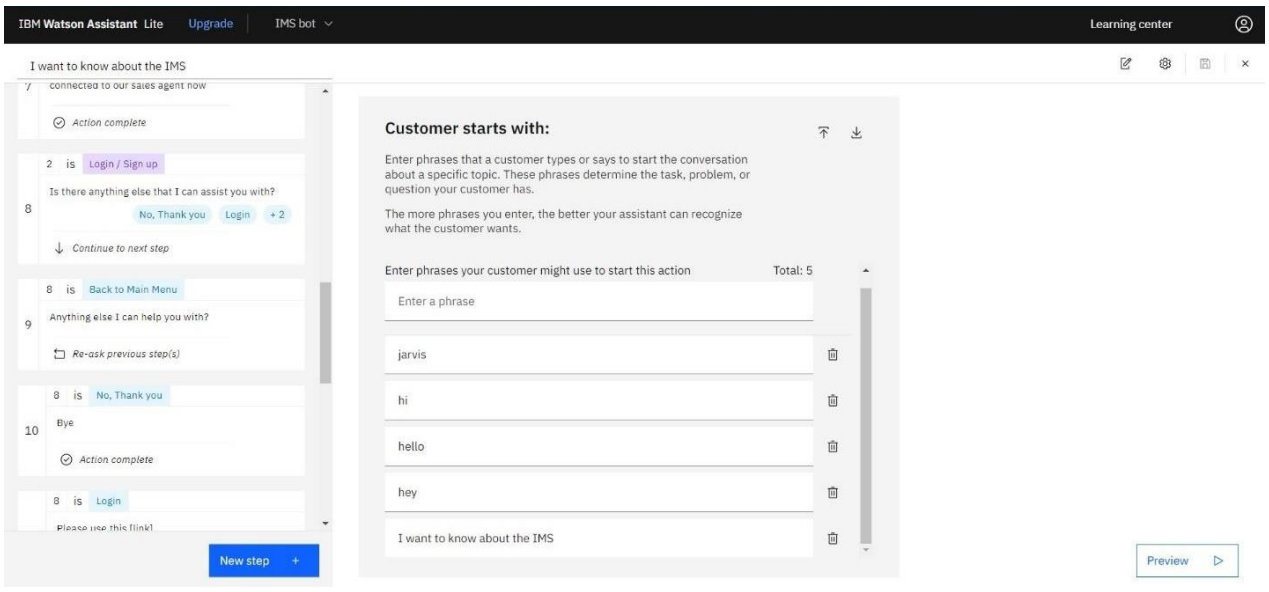
spec:
containers:
- name: job-portal-container
image: jp.icr.io/padmanaban/ims-final imagePullPolicy: Always
ports:
- containerPort: 5000 protoco

8. RESULTS & OUTPUTS:

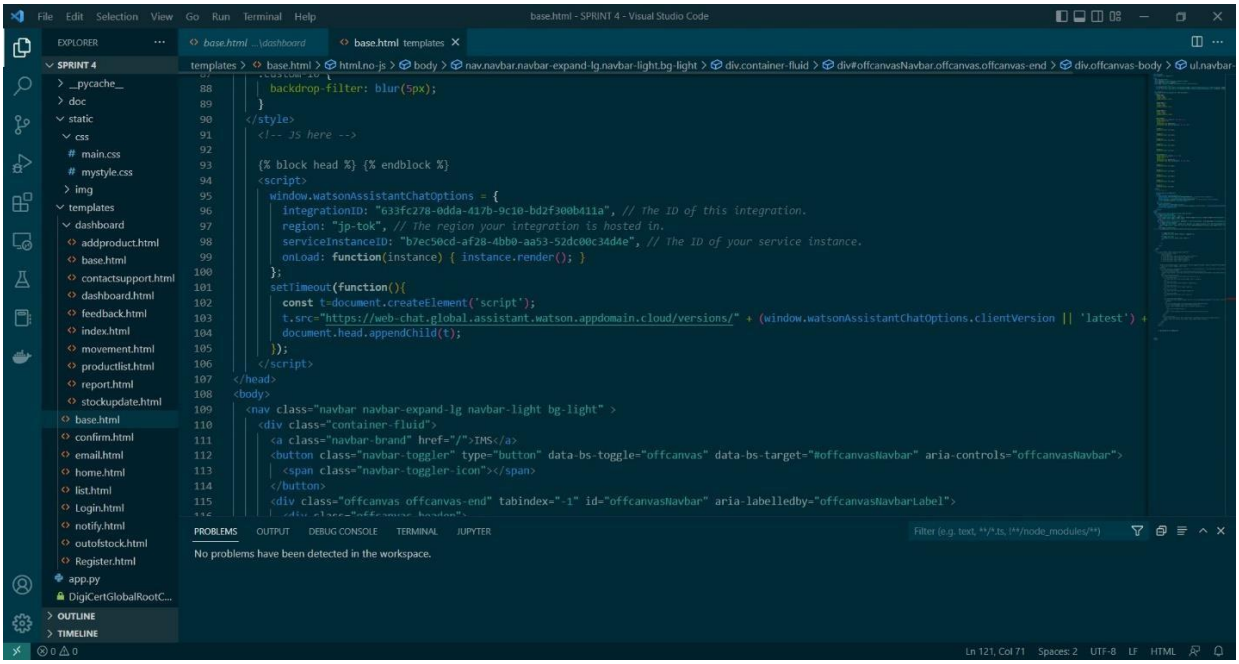


Setting Actions to IBM Watson Assistant:

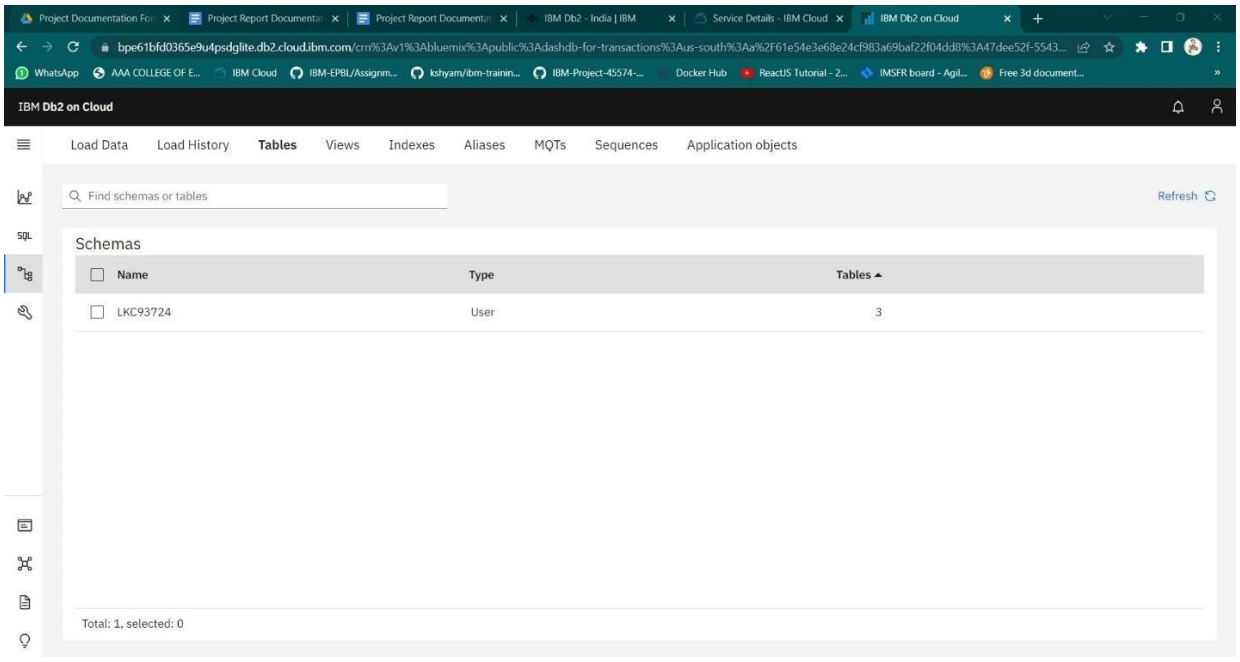


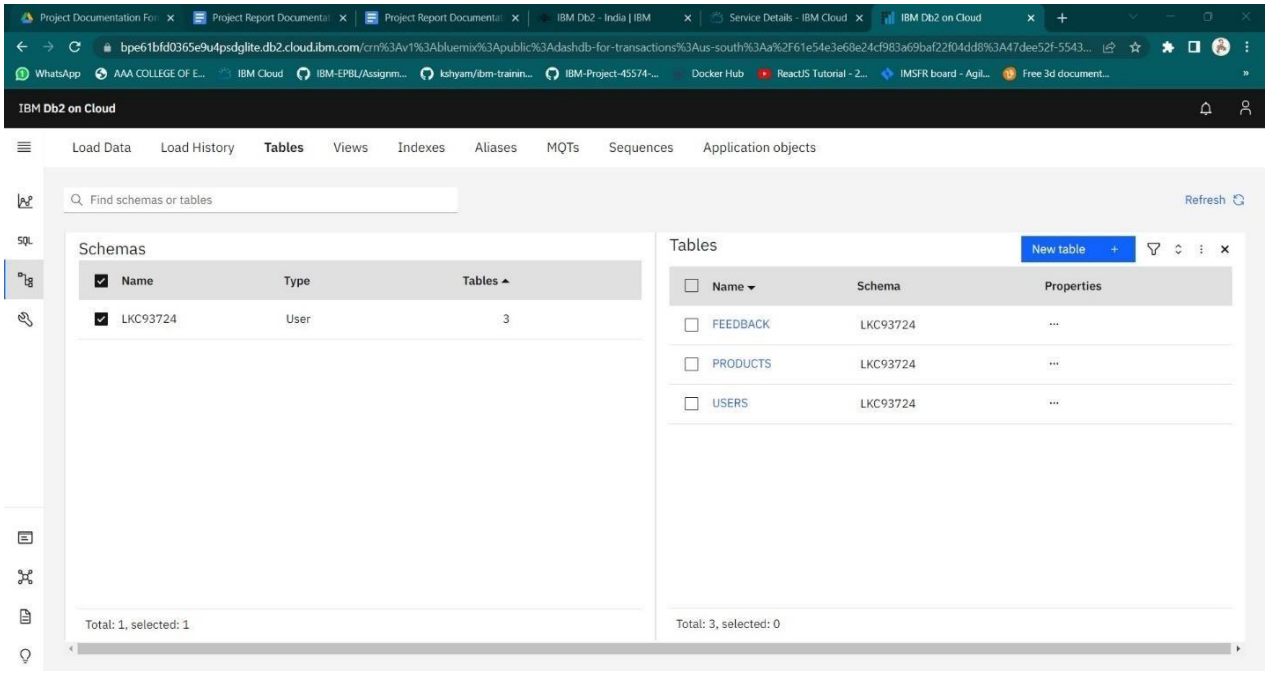


Integrating the IBM Watson Assistant in the base template:

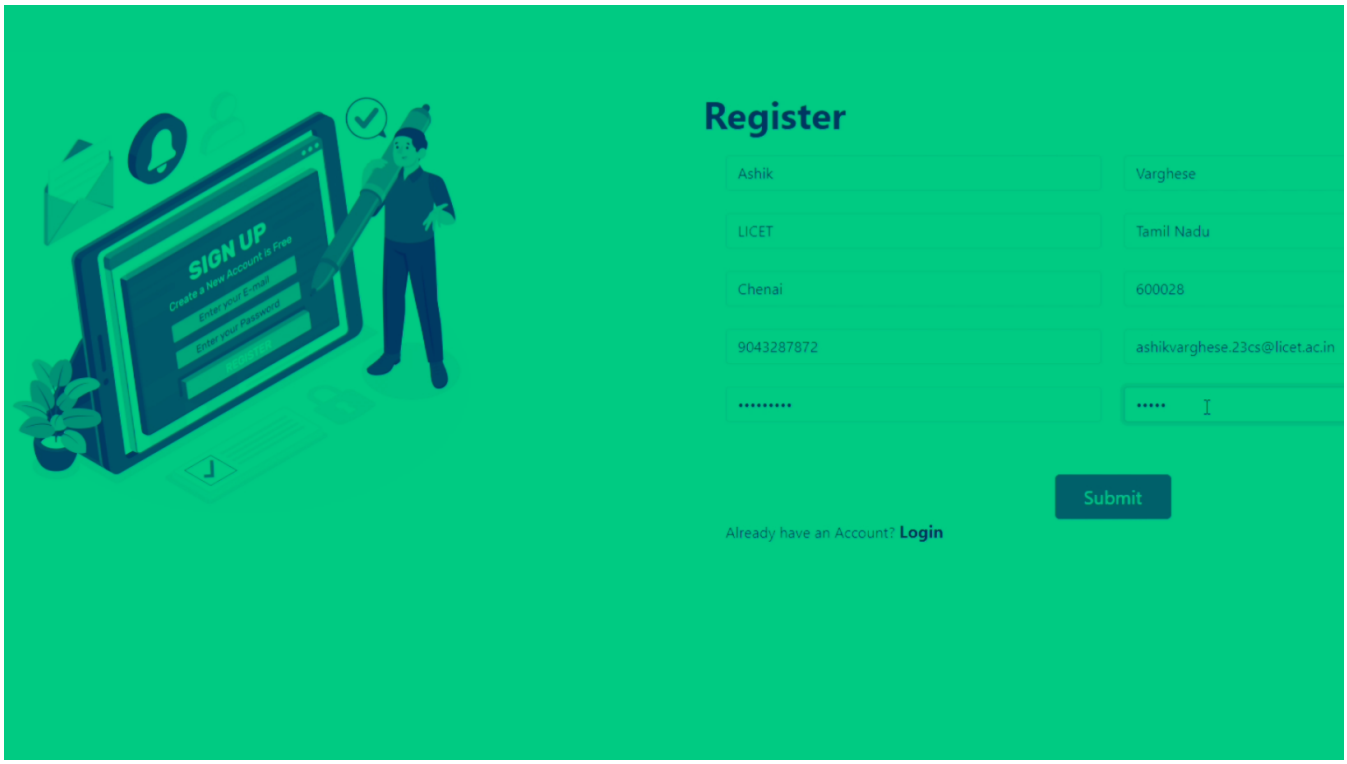


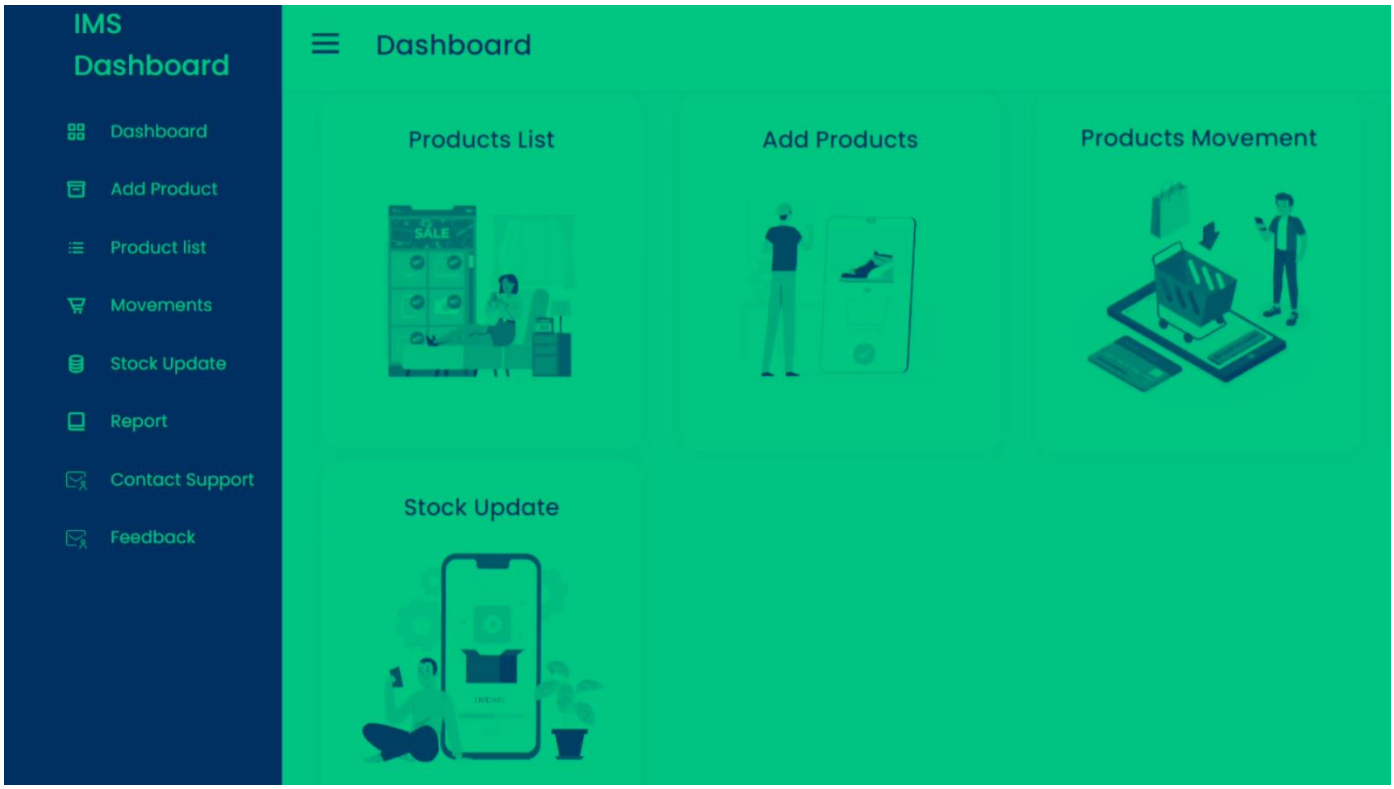
Database Schema(IBM DB2)

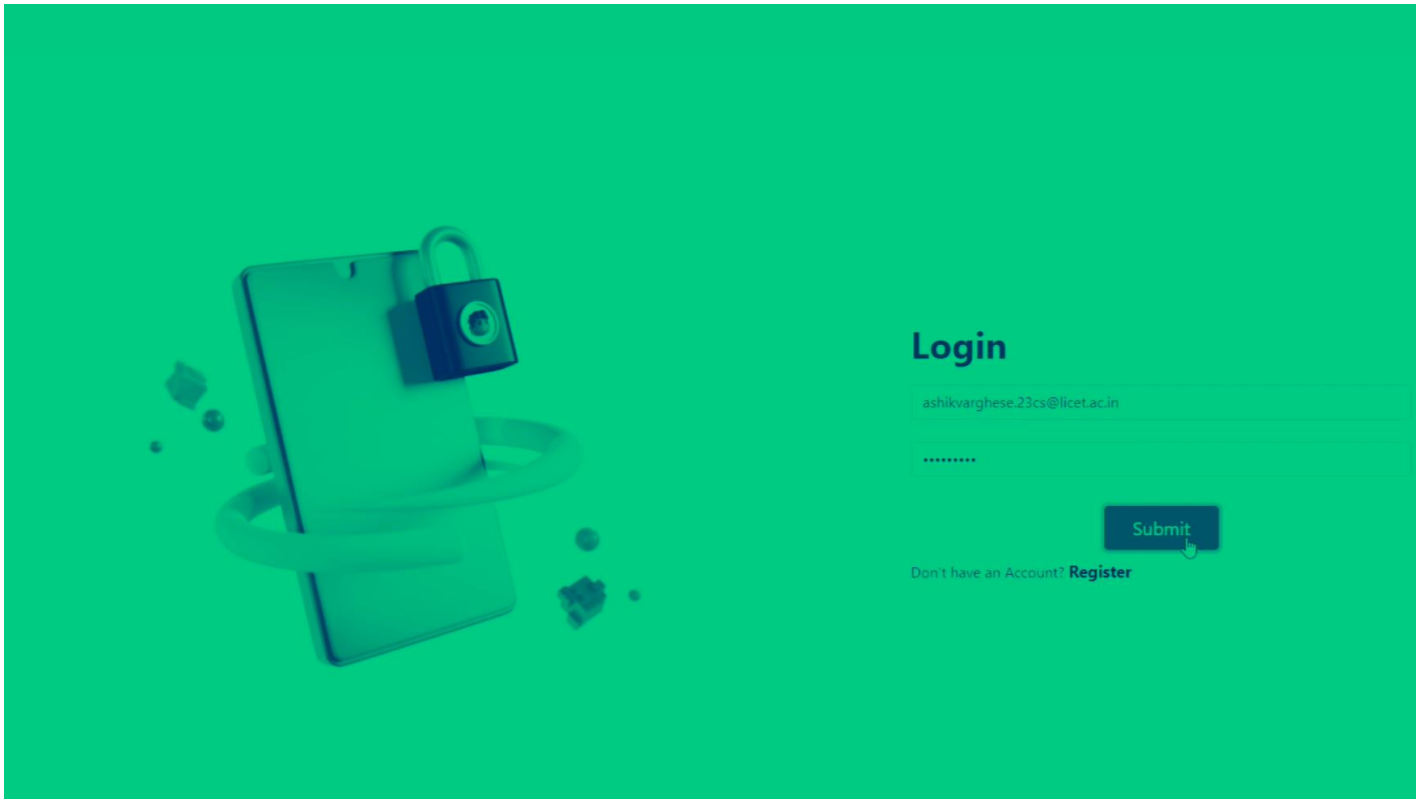


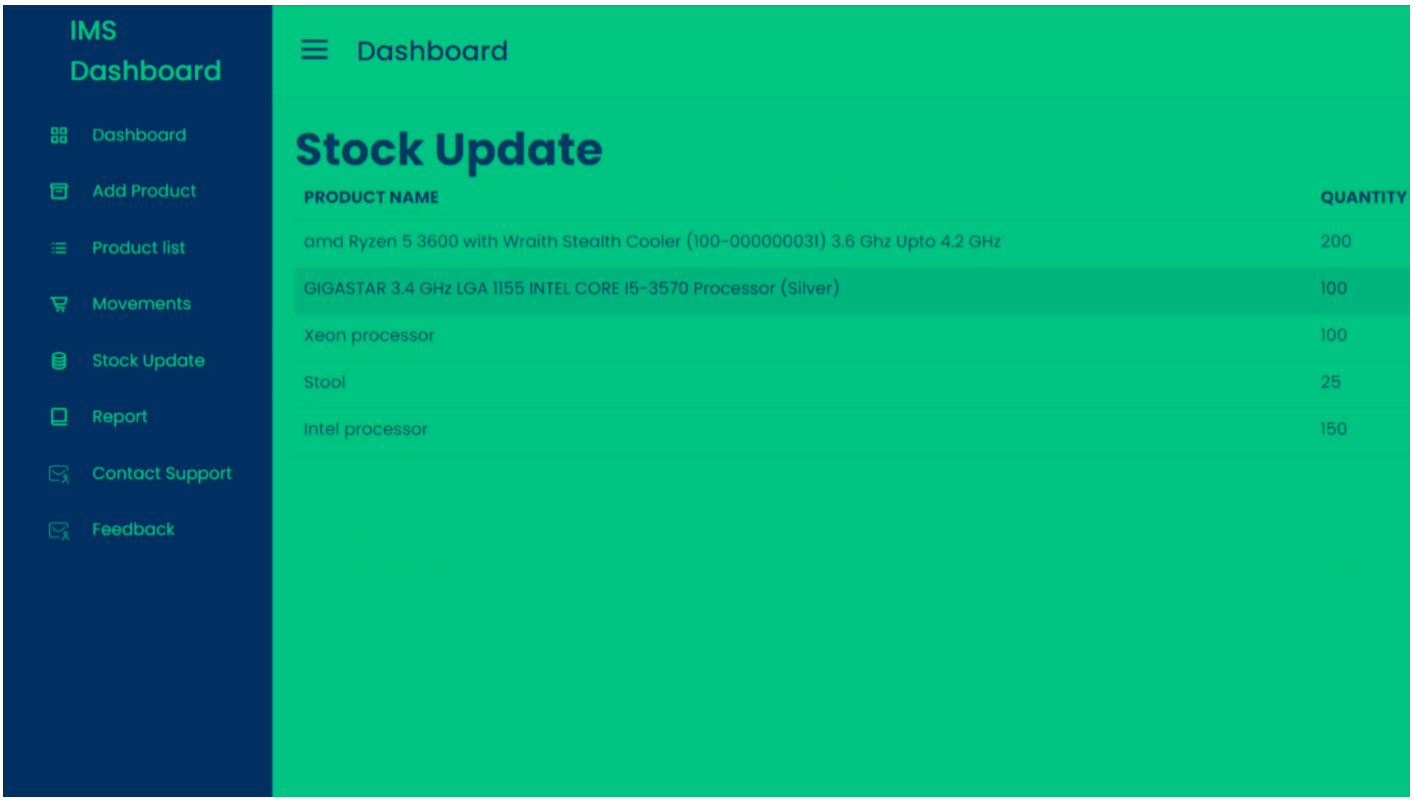


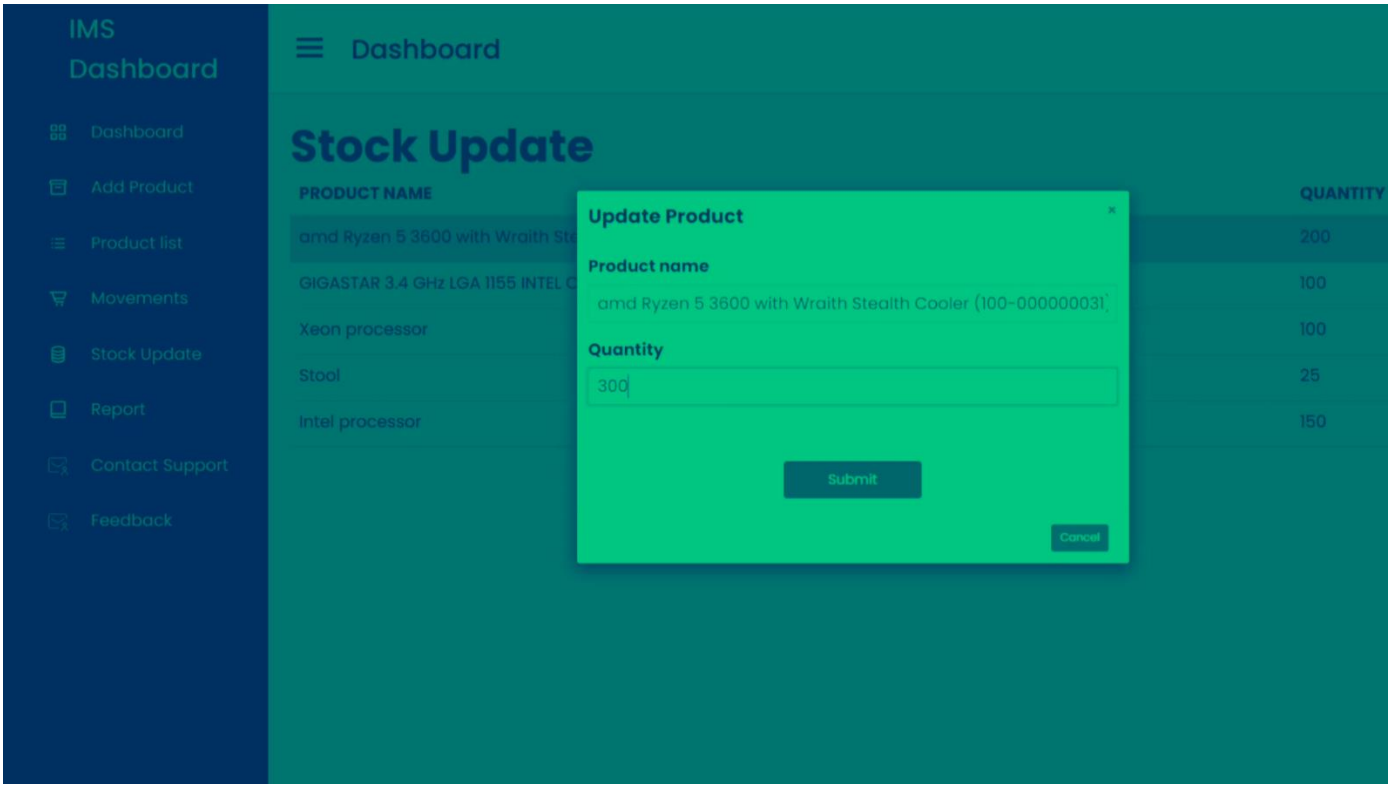
Other application images

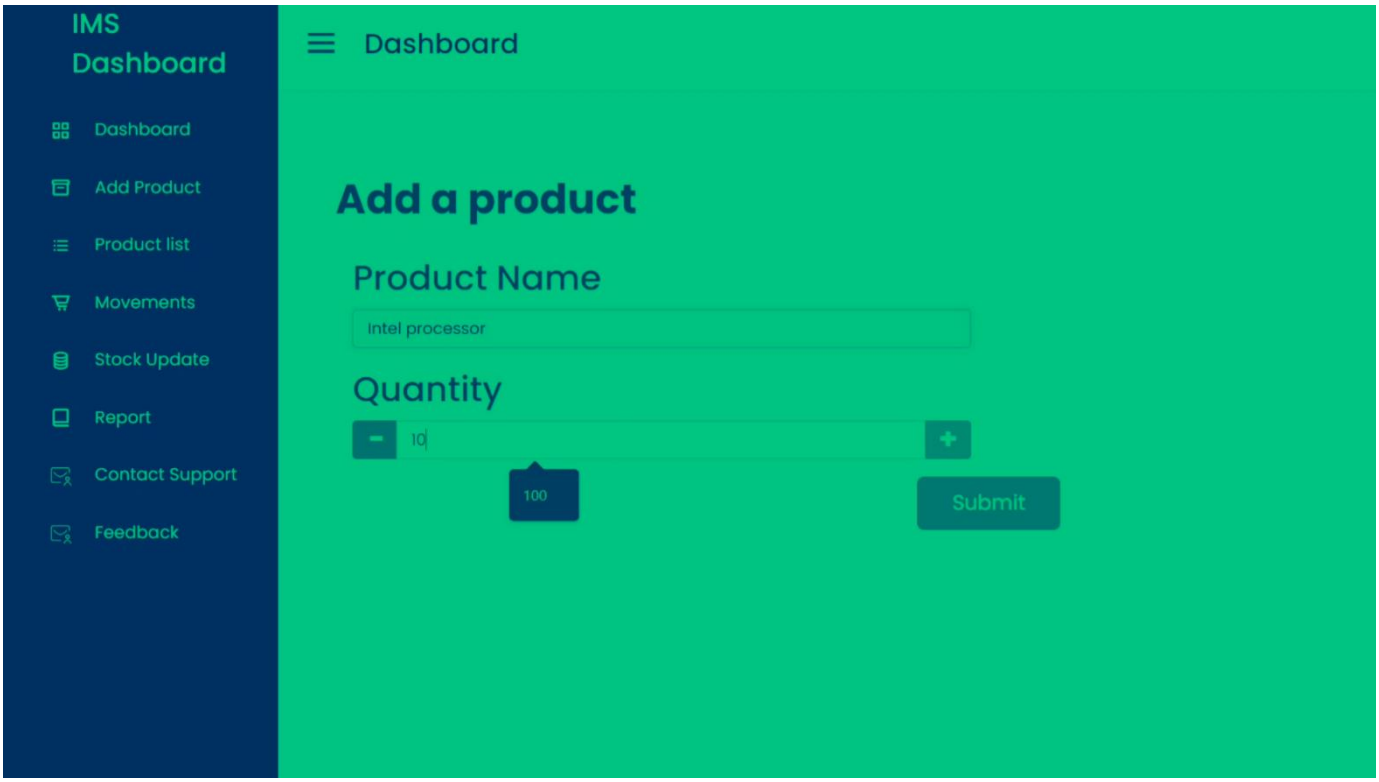


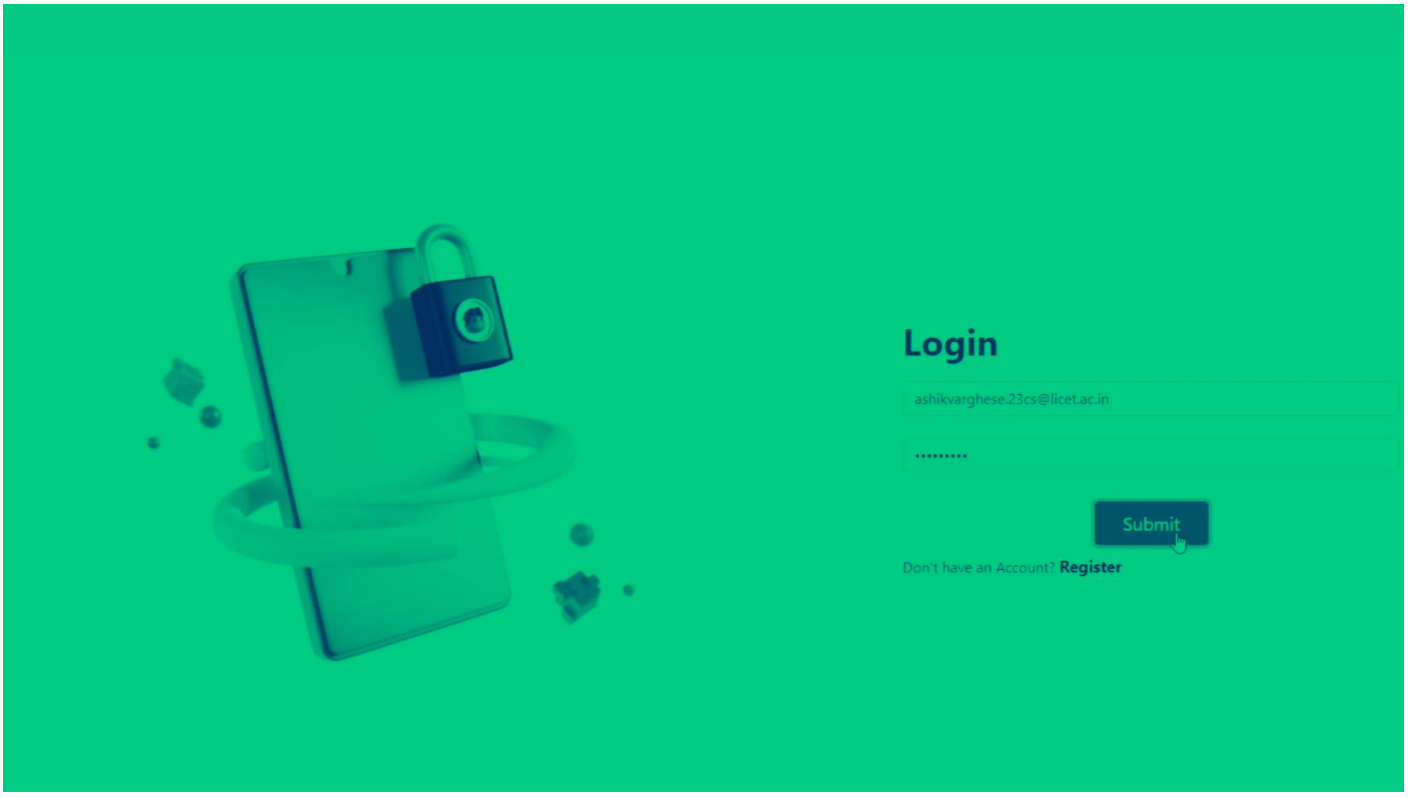

IMS
Dashboard
Dashboard
Add Product
Product list
Movements
Stock Update
Report
Contact Support
Feedback
Dashboard
Products List
SALE
Add Products
Products Movement
Stock Update


Login
ashikvarghese.23cs@licet.ac.in
Submit
Don't have an Account? Register


IMS Dashboard
Dashboard
Add Product
Product list
Movements
Stock Update
Report
Contact Support
Feedback
Dashboard
Stock Update
PRODUCT NAME
QUANTITY
amd Ryzen 5 3600 with Wraith Stealth Cooler (100-00000031) 3.6 Ghz Upto 4.2 GHz
200
GIGASTAR 3.4 GHz LGA 1155 INTEL CORE I5-3570 Processor (Silver)
100
Xeon processor
100
Stool
25
Intel processor
150


IMS Dashboard
Dashboard
Add Product
Product list
Movements
Stock Update
Report
Contact Support
Feedback
Dashboard
Stock Update
PRODUCT NAME
QUANTITY
Update Product
Product name
amd Ryzen 5 3600 with Wraith Stealth Cooler (100-00000031)
Quantity
300
Submit
Cancel
200
100
100
25
150


IMS
Dashboard
Dashboard
Add Product
Product list
Movements
Stock Update
Report
Contact Support
Feedback
Dashboard
Add a product
Product Name
Intel processor
Quantity
10
100
Submit


Login
ashikvarghese.23cs@licet.ac.in
Submit
Don't have an Account? Register

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 Leanpractitioners claim that inventory performance is the single best indicator of the overall operational performance of a facility.

10. ADVANTAGES & DISADVANTAGES

- Paper-based retail inventory management can take a lot of time and effort. The retail inventory management software can cut short your in-store inventory process cycles through automation. Automation would give you time to focus on other productive business tasks.
- Inventory management is one of the crucial retail processes. Thus, any discrepancy in the inventory control would impact all other operations in your company. The retail inventory software can streamline the inventory processes, which would, in turn, improve the efficiency of your entire business
- Manual inventory control would increase your labor and process costs. The software would not only help you save time, but it would also help you reduce costs. As a result, the profitability of your business would improve. Also, you can invest the excess funds in activities that promote your business growth.
- One of the biggest problems with any computerized system is the potential for a system crash. A corrupt hard drive, power outages and other technical issues can result in the loss of needed data. At the least, businesses are interrupted when they are unable to access data they need. Business owners should back up data regularly to protect against data loss.
- Hackers look for any way to get company or consumer information. An inventory system connected to point-of-sale devices and accounting is a valuable resource to hack into in search of potential financial information or personal details of owners, vendors or clients. Updating firewalls and anti-virus software can mitigate this potential issue.
- When everything is automated, it is easy to forego time-consuming physical inventory audits. They may no longer seem necessary when the computers are doing their work. However, it is important to continue to do regular audits to identify loss such as spoilage or breakage. Audits also help business owners identify potential internal theft and manipulation of the computerized inventory system.

11. CONCLUSION

Inventory management is a very complex but essential part of the supply chain. An effective inventory management system helps to reduce stock-related costs such as warehousing, carrying, and ordering costs. As you have read above, there are different techniques that businesses can utilize to simplify and optimize stock management processes and control systems.

12. FUTURE SCOPE

In summary, successful companies will embrace the challenges of inventory management in the 21st century by levering the technology that is being offered through the Fourth Industrial Revolution. More important, companies will look at inventory as a strategic asset, that when properly deployed will deliver increased value and competitive advantage. Effective collaboration between supply chain partners will take on increased importance. The intensifying risks inherent with global sourcing in combination with a better appreciation of TCO will motivate companies to rethink their global inventory strategies.

13. REFERENCES

- Aggarwal, S.: A review of current inventory theory and its applications. International Journalof Production Research 12, 443–472 (1974)
- Anily, S., Federgruen, A.: One warehouse multiple retailer systems with vehicle routing costs. Management Science 36, 92–114 (1990)
- Beckmann, M.: An inventory model for arbitrary interval and quantity distributions of demand. Management Science 8, 35–57 (1961)
- Hamann, T., Proth, J.: Inventory control of repairable tools with incomplete information. International Journal of Production Economics 31, 543–550 (1993)

14. APPENDIX

GitHub link

https://github.com/IBM-EPBL/IBM-Project-38082-1660370377

Demo Video Link:

https://licetacin-my.sharepoint.com/:v:/g/personal/ashikvarghese_23cs_licet_ac_in/EVSWj9vB_StHjHIIQRzCLs4BDzE-liHb9jsXrYYfuPU93A?e=06YBri