CHAPTER 1

INTRODUCTION

Sustaining the optimal level of stock so little neither too much—of requested goods is the aim of retail inventory management. If retailers manage their inventory well, they can satisfy customers requirements before running out of supplies. Affordable pricing inventory tracking cuts costs and promotes practical understanding of sales trends. Methodologies for retail inventory management give organisations more data with which to operate their business. Applications have been created to assist shops in keeping track of and managing the supply of their own products. Retailers will be prompted by the System to register their accounts by providing necessary information. Retailers can log into the programme to access their accounts. Retailers can change their inventory details once they have successfully logged into the application. Additionally, users will be able to add new stock by providing the necessary stock-related information. They have access to the current inventory's specifics. If there is no stock found in the accounts of the retailers, the System will automatically send an email alert to them. in order for them to order new stock.

1.1 PROJECT OVERVIEW

Systems for managing inventory keep track of all goods coming in and going out of a physical store or online. Time spent on stock management activities like cycle counts, physical inventories, and manual refill computations is reduced with an maintain stock products. Through essential zero correction, back-stock guiding and monitoring, and stored replenishment, RFID inventory software increases on-shelf availability. Retailers can save their overheads by using inventory management systems to gather data, do computations, and generate records automatically. Along with time and money savings, this also improves business productivity. Distributors, wholesalers, manufacturers, and retailers may improve their warehouses with the aid of inventory management software.

1.2 PURPOSE

Applications for inventory management are growing in popularity. An online ordering system's major idea is to give customers a simple means of ordering a product through an internet-based application. The extent to which a shop will benefits from or be able to implement an informant system The issue's feasibility is evaluated. As we've been creating the project inventory control system for trailers, we've centred on the following two main categories of feasibility research. It evaluates solution as well as the better for technical resource and expertise availability. What is practical and reasonable is referred to as "achievability." It largely addresses their main worries.

CHAPTER 2

LITERATURE SURVEY

[1] TITLE: Supply Chain Inventory Collaborative Management and Information

AUTHOR: Fuan Zhang & Zhenzhi Gong & 2019.

The study paper examines how economic globalisation is progressing and how businesses are becoming more and more competitive. In order to achieve supply chain integration, businesses need to share information often. This article intends to investigate how supply chain inventory is managed collaboratively and information is shared using network based things and cloud computing. This article discusses the bullwhip phenomenon and the problem of information transfer their continuous chain before introducing the theory and practises of cooperative exchange mechanism. Six aspects are then examined from the forecasting, supply chain structure, time lag, and shortage game. Analysis is done on the origin of the bullwhip phenomena. Additionally, the supply chain's total advantages have increased commensurately, and the market's competitiveness has been improved, since each node enterprise's operating costs and risks have decreased by 12% in comparison to the no sharing scenario.

[2]TITLE: Implementation of Inventory Management System

AUTHOR: Keshav Srivastava & Dilip Kumar Choubey & 2020.

The research article talks about how every E-Commerce department shop receives a tonne of data every day, which can occasionally lead to missing items, poor inventory management, and losing sight of their database. The customers also play a significant role in creating this situation by altering the products in the cart, leaving the cart with things at any time, which causes issues at checkout, and frequently cancelling the orders. A system that not only stores this changing data, but also manages it effectively, is absolutely necessary. Because it uses MongoDB to store data, this system keeps good track of all the details regarding the dealer, supplier, manufactured goods, and raw materials. In order to give a good Graphical User Interface (GUI), the data on the backend and the frontend are constructed using Java on NetBeans. This allows anyone without any technical experience to access the inventory. The current work could contribute to a high standard of agreed-upon customer service. It might encourage us to choose flexible capacity and give us the ability to handle demand peaks and troughs.

[3]TITLE: inventory management for smart system using IOT.

AUTHOR: Souvik Paul Atrayee Chatterjee; Digbijay Guha &2021.

The study paper discusses how traditional inventory management models can't meet growing business needs for product diversity because of their high workload and ineffectiveness.This essay outlines the concepts and design of a cutting-edge, IoT-based intelligent inventory management system. We anticipate positive development because this system has several advantages over the usual practice and because inventory management is an essential element of customer service and cost-cutting in every production system. Inventory management for businesses that have gone global and have thousands of components and hundreds of warehouses is a time-consuming process that requir es a lot of effort. traditional robotic arm systems for the foundation for inventory pick-up and delivery has been the marking and tracking of warehouse areas.

[4]TITLE: Cloud Based Inventory Management System.

AUTHOR: Vivek Sharma; Aamod Krishna Tiwari; Ayush Srivastava; Divyanshu Srivastava & 2021.

In fact, the e-commerce sector has been controlled by businesses like Amazon, eBay, FedEx, and DHL, according to this comprehensive research paper. Worldwide shipping industries have expanded their operations in response to the demand for their services. Costs for maintenance and web-based management solutions have also gone up as a result. We might think about creating some cloud computing projects with improved functionality and lower prices. Cloud-based shipping management systems may speed up every step of the process, from placing the order to getting the product successfully delivered to the customer and receiving positive feedback. To boost automation and data centralization, the efficiency of the data is improved. Significant cost savings are achieved for the company, and increased visibility and greater scope for the entire business. It enables us to more effectively monitor processes in real-time and create precise models for merging the entire process. Automation that takes place through cloud-based networks mostly depends on the ability of the full team of operations to be accessed from anywhere at any time. Additionally, it links all participants in the supply chain, providing a critical strategic approach to inventory building. It aids in delivery network monitoring and gives slow-moving packages priority. Internal hardware is somewhat eliminated from the automation process as a whole and is even more affordable to acquire and operate. The best practises, which provide flexibility, security, and scalability, are available for businesses of all sizes. Additionally, functionality enhancements that improve connectivity, dependability, and overall ROI.

[5]TITLE: IoT based interactive shopping ecosystem.

AUTHOR: Abhirup Khanna; Ravi Tomar &2018

The goal of the research project on the Internet of Things is to connect networked information systems to physical entities. Through the internet, it links items like smart phones, sensors, LED (Light Emitting Diode) displays, and even clothing so they can communicate and share information. IoT has practically found use in every aspect of life today, and inventory management is no different. As the variety and quantity of both items and customers increase exponentially, inventory management is currently becoming more and more difficult. Optimizing inventory management in order to improve sales and decrease operating expenses is the largest challenge faced by store owners. With such massive product selections, retail owners find it incredibly challenging to track and keep an eye on a product's performance in terms of sales, shelf life, cost, and consumer response. IoT serves as a solution to this issue by making it easier to employ Wireless Sensor Networks (WSN) to connect all the different players in a logistic system. This enormous network of linked devices produces enormous amounts of data that are challenging to store and process. In this case, cloud computing acts as a facilitator and offers significant assistance in resolving issues with storage and processing capacity. In this article, we offer an Interactive Shopping Model and an Automated Inventory Intelligent Administration System (AIMS), which use the advantages of IoT and the cloud to provide real-time monitoring, tracking, and management of products. We also offer an algorithm that shows how our system functions. The iFogSim simulator is used to model both the proposed system and algorithm. In order to achieve this, we provide examples of the simulated outcomes and the operation of our suggested system.

2.1 EXISTING PROBLEM

In inventory management systems, demand is sometimes unpredictable and lead times might change. To prevent shortages, managers usually store a safety stock. It can be challenging to predict what order sizes and reorder intervals will result with in lowest overall inventory investment in such circumstances. The general problem of determining that however much inventories to depend on stock in anticipation of potential demand is referred to as the inventory issue. Loss happens when a stocks cannot meet demand (such as when a retailer has a later a products and whenever soldiers in a conflict run out the of ammunition) or when products are stockpiled even if there is no market for them.

2.2 REFERENCES

[1] D. Plinere, L. Aleksejeva&2020"Agent system application as a tool for inventory management improvement," in 8th Int. Conf. on Soft Computing, Computing with Words and Perceptions in System Analysis.

[2] D.C.U. Cadavid, C.C. Zuluaga&2020 "A framework for decision support system in inventory management area," Ninth LACCEI Latin American and Caribbean Conf., Medellin, Colombia.

[3] D. Dhoka, Y.L. Choudary &2019"ABC Classification for Inventory Optimization," IOSR Journal of Business and Management.

[4] S.Angel Raphella, S.Gomathi Nathan and G.Chitra&2018 "Inventory Management- A Case Study", International Journal of Emerging Research in Management &Technology.

[5] Abhirup Khanna; Ravi Tomar &2018. IoT based interactive shopping ecosystem.

[6] Weijun Yang; Yuqiang Chen; Yeh-Cheng Chen; Ku-Ce Yeh & 2021. Intelligent Agent-Based Predict System With Cloud Computing.

[7] Fuan Zhang & Zhenzhi Gong & 2019. Supply Chain Inventory Collaborative Management and Information.

[8] Keshav Srivastava & Dilip Kumar Choubey & 2020. Implementation of Inventory Management System.

[9] Souvik Paul Atrayee Chatterjee; Digbijay Guha & 2021. Study of smart inventory management system based on the internet of things.

[10] Vivek Sharma; Aamod Krishna Tiwari; Ayush Srivastava; Divyanshu Srivastava & 2021. Cloud Based Inventory Management System.

2.3 PROBLEM STATEMENT DEFINITION

Demand is generally a moving target in inventory systems, and lead times might change as well. Mangers retain a safety stock to prevent shortages. It is unclear in these circumstances what order sizes and reorder points will reduce the anticipated total inventory cost. This issue can be solved through simulation models. Poor inventory management results in difficulties because you don't have reliable real-time information on how much product you have, in addition to having too many or too few inventory. This raises the possibility of ordering merchandise from suppliers incorrectly or of selling fake inventory.

CHAPTER 3
IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

An example of such a system is a simple picture that conveys concepts on a user's characteristics and behaviour. It is a helpful tool that enables teams to comprehend their users most fully.
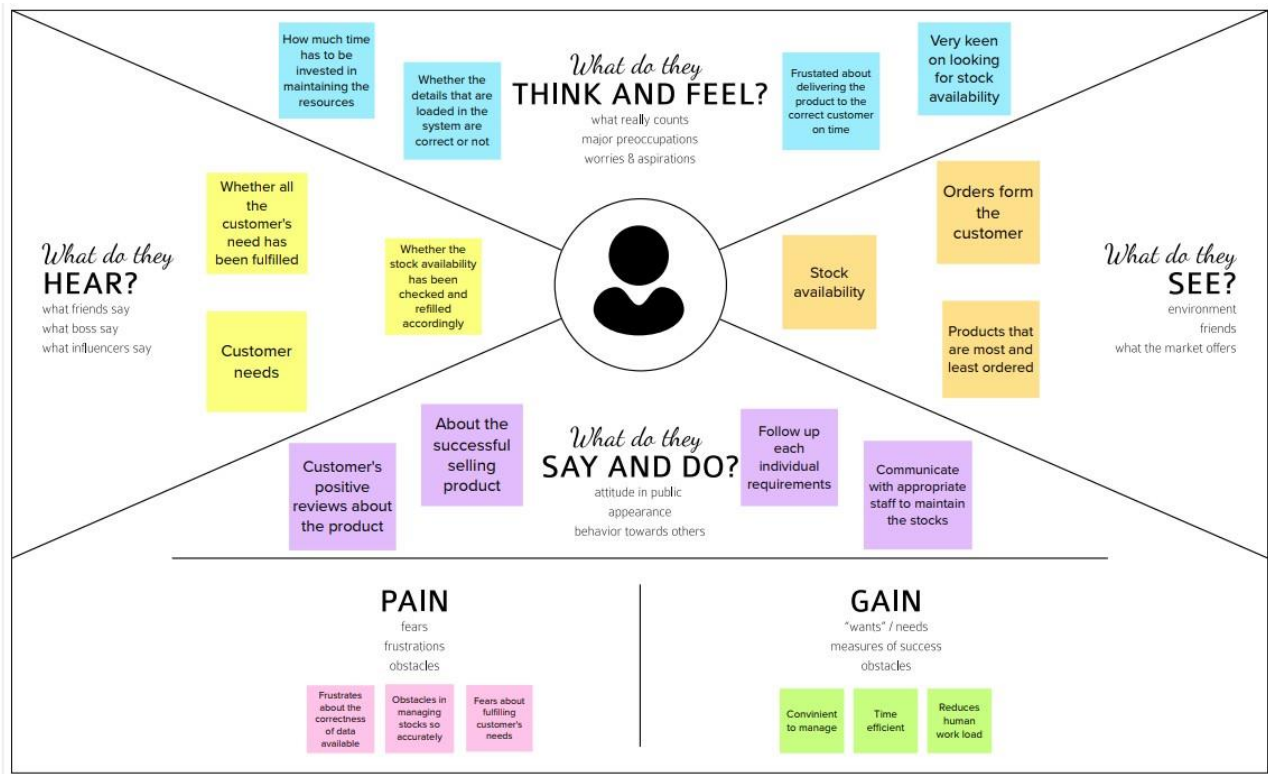
Fig 3.1.1 EMPATHY MAP CANVAS

3.1 INDEATION & BRAINSTORMING

During a brainstorming process, everyone on a team is invited to take an active part of original thought that results in problem solving. By prioried product under qualities, we urge our participants to work together and generate a wealth of innovative ideas.
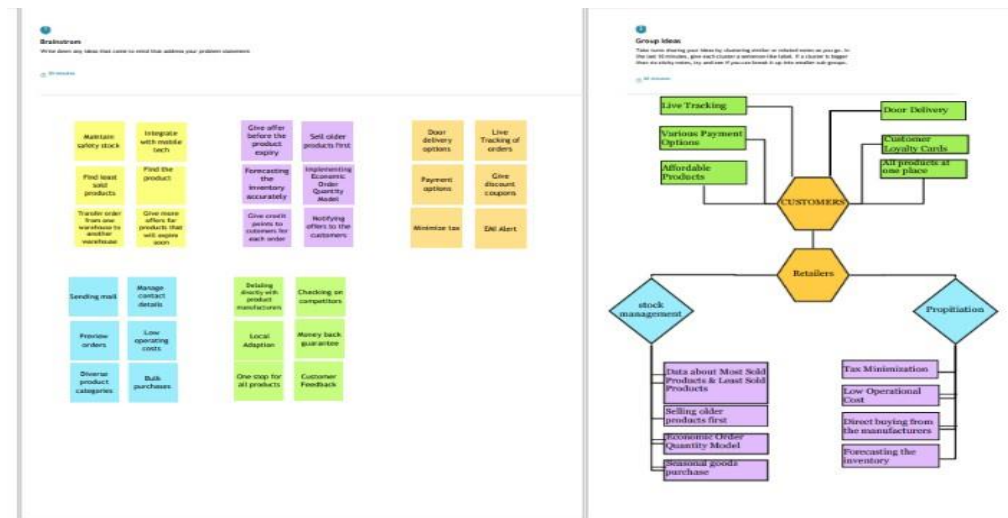
Fig 3.2.1 BRAINSTORMING

3.2 PROPOSED SOLUTION

The goal of the inventory control system was to produce as little idle, waste, surplus, scrap, and outdated items as possible. to view inventory as a dangerous investment. to reduce costs. Making procuring, distributing, storing, and utilising inventory as easy and effective as possible for businesses is the main focus on product management. If you effectively manage your inventory, you'll always be aware of what items are available, how many they are, and where they are. The basic purpose of an inventory system is to keep record of your goods and supplies. You must record when you purchased the inventory, when you sold it, and how much you still have on hand. In order to handle the merchandise, it also notifies you where your inventory .

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Retail inventory management is the process of ensuring you carry merchandise that shoppers want, with neither too little nor too much on hand. By managing inventory, retailers meet customer demand without running out of stock or carrying excess supply . The System will ask retailers to create their accounts by providing essential details. Retailers can access their accounts by logging into the application. Once retailers successfully log in to the application they can update their inventory details, also users will be able to add new stock by submitting essential details related to the stock. They can view details of the current inventory. The System will automatically send an email alert to the retailers if there is no stock found in their accounts. So that they can order new stock.
2.	Idea / Solution description	Inventory management software for the retail industry focuses on recording the stock levels that a retail company has within an inventory database that updates in real-time. These solutions streamline order management as they enable businesses to automatically reorder stock that is running low before it runs out, ensuring no sales opportunities are missed. On the other hand, this also means that overstocking can be avoided if certain products aren't selling as expected.
3.	Novelty / Uniqueness	Applications have been developed to help retailers track and manage stocks related to their own products. Reduces the time for managing inventory by keeping record in place.
4.	Social Impact / Customer Satisfaction	The results indicate that higher levels of inventory management practice can lead to an enhanced competitive advantage and improved organizational performance.
5.	Business Model (Revenue Model)	Inventory management system makes it so much easier to keep an updated overview of stocking levels regardless of how many locations you manage.
6.	Scalability of the Solution	Switch from spreadsheets to a cloud-based automated inventory management solution. Sync online orders, Scan barcodes, Manage multiple warehouses, Packages and Shipments.

3.3 PROBLEM SOLUTION FIT

The lot of businesses we encounter have fantastic ideas that they are so passionate about that they secretly wish they could force down their customer' throats. Even some of these business owners asked us if there was a way they could educate their clients so they would need what they offered.we genuinely don't believe that this is the ideal strategy to develop profitable ventures. Finding an existing problem and coming up with a helpful and fulfilling solution is the ideal way to put an entrepreneur's drive, resourcefulness, creativity, and technical expertise to use.

Fig 3.4.1 PROBLEM SOLUTION FIT

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

The essential to be established product is having requirements that are well specified. They provide a formal understanding that a client and a supplier are together pursuing the same objective. High-quality, exhaustive specifications aid in reducing financial risks and maintaining the project's timeline. Requirements, as defined by the Business Analysis Body of Knowledge, are a practical illustration of a need.

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Easy Inventory Managementsystem	The software makes the process of inventory management a lot easier which saves time. It assists to automate the business processes and guides to make stock calculation. this project is used to product count management system.
FR-2	Inventory Reports	The software is meant to generate automated report, inventory validation report and inventory forecast report. If any product count is zero ,automatically email alert go to admin or retailers.
FR-3	Productivity and Efficiency	Inventory management software enables us to increase productivity and efficiency by implementing automated daily manual tasks. The software saves uncountable hours and gives the opportunity to print shipping labels, process and dispatch orders, manage stock, create and update the listing on the system.
FR-4	Avoid Stock-outs and Over-stock	When it comes to maintaining the balance sheet of inventories and its management, it is a difficult and challenging task to handle. Case of less stock leads to stock-out which not only disrupt customer relation but cause a possible loss whereas in case of over-stock its storage creates a problem.
FR-5	Inventory Alerts	Various modules trigger inventory alerts that help reduce waste, optimize inventory financials and managecustomer expectations. These features include automated email that alert you to low inventory levels ,shipment and supply chain delays. . If any product count is zero ,automatically email alert go to admin or retailers.

4.2 NON-FUNCTIONAL REQUIREMENT

System describe a common features in the requirement. They are sometimes referred to as qualities. What a business must do in order to successfully transition from its current state to its intended state with the new product is outlined in a separate set of requirements. They are only necessary during the short time that the shift is taking place. Examples include "users must receive global networks" and "prior data must be moved to the cloud storage."

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

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Usability examines the effectiveness of the inventory software. If it takes hours for your staff to learn the software, then it's not worth it. You should remember to choose a solution that simplifies inventory management.
NFR-2	**Security**	Inventory security is the process of ensuring the safety and security. It is of central importance for optimum warehouse management because the performance of a company stands or falls with the safety and efficiency of a warehouse.
NFR-3	**Reliability**	Improving inventory reliability is important for several reasons. Your delivery reliability depends on it, but also consider the costs involved. your data is important to our store so data privacy is reliable.
NFR-4	**Performance**	Inventory Performance is a measure of how effectively and efficiently inventory is used and replenished and productivity. The goal of inventory performance metrics is to compare actual on-hand dollars versus forecasted cost of goods sold.
NFR-5	**Availability**	Inventory availability refers to whether a specific item is available for customer orders. Additional information provided by retailers may include the quantity available. If the item is not in stock, the retailer may indicate its status. after update the stock who are visit that product receive mail from retailers your product stock is ready to buy.
NFR-6	**Scalability**	The inventory management software or app you choose be able to grow as your business does? The last thing you want is to have to manually re-enter all your inventory if you outgrow your current system. retailer to avoid paper pen calculation ,easily install that software and easily update the stock details .only admin or retailers can do it.

CHAPTER 5
PROJECT DESIGN

5.1 DATA FLOW DIAGRAM

A data flow diagram shows how information flows through a process or system. This includes data input/output, data storage, and various sub processes through which data moves. DFDs are created using standardized symbols and notations to describe various entities and their relationships.
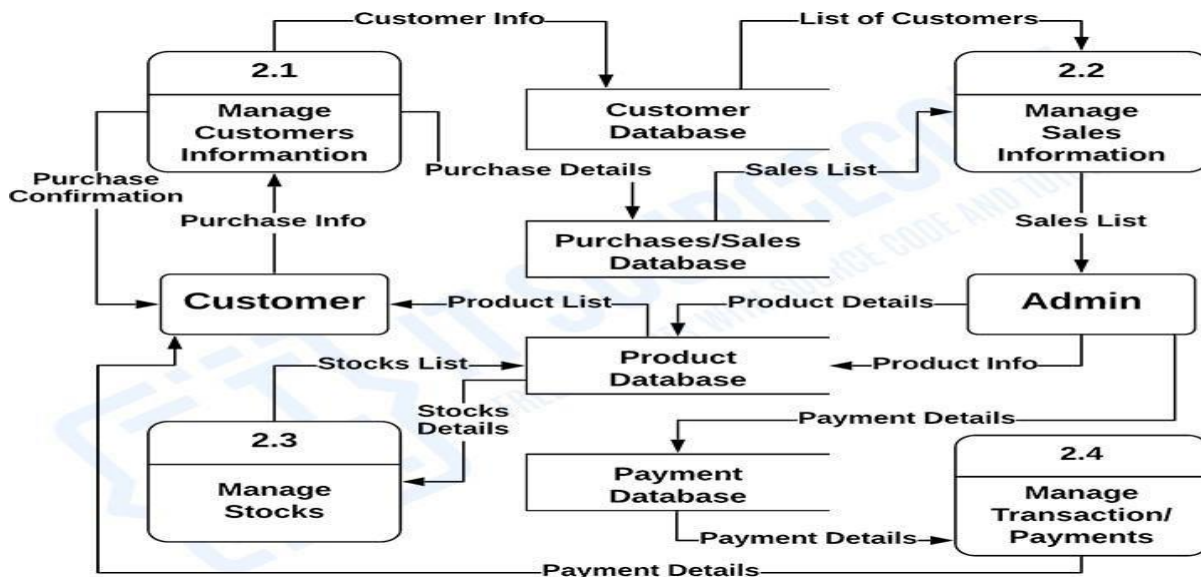
Fig 5.1.1 DATA FLOW DIAGRAM

5.2 SOLUTION & TECHNICAL ARCHITECTURE

The data can be stored in database and user can retrieve the data whenever required. If enter the product name automatically shows the inventory stock count. Information of the stocks will be stored and hosted on the IBM cloud. Requirements to store files.if the product count zero,it will automatically notify. SendGrid used in application will send the email alert if there is less number or no stock to the user.

Fig 5.2.1 TECHNICAL ARCHITECTURE

5.3 USER STORIES

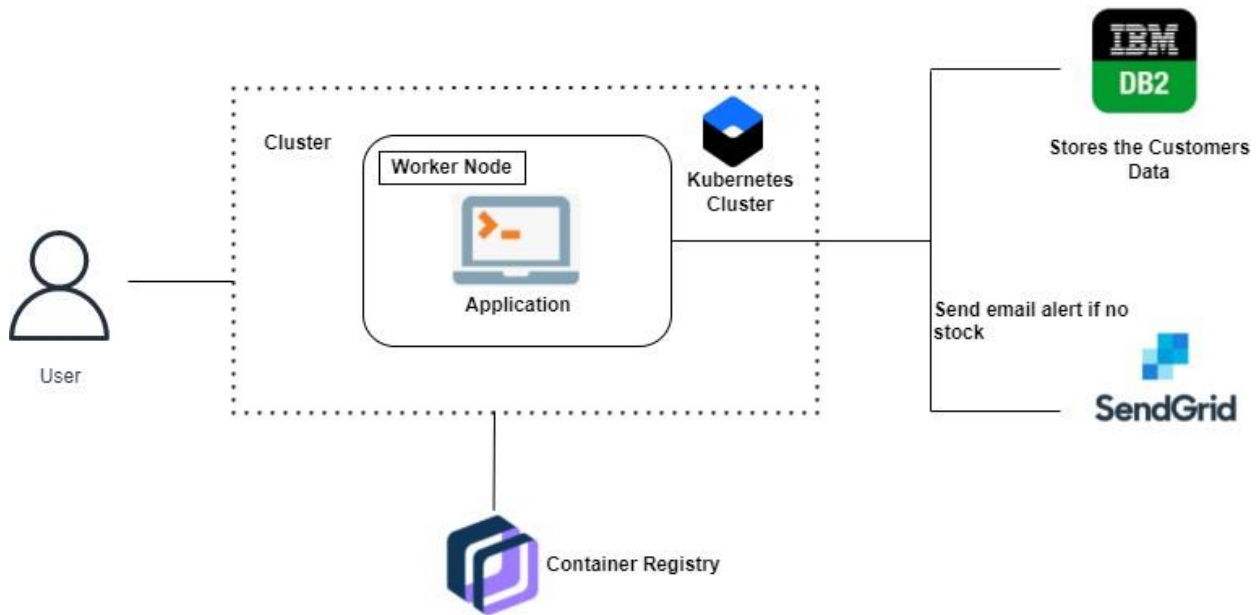| User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	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
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-3
		USN-4	As a user, I can register for the application through Gmail	I can register for the application through Gmail	Medium	Sprint-2
	Login	USN-5	As a user, I can log into the application by entering email & password	I can log in by entering Gmail & password	High	Sprint-1
	Dashboard	USN-6	As a user, I can track data of sales of products and inventory levels	I can track data of sales of products and inventory levels.	High	Sprint-1
Customer (Web user)	Registration	USN-7	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-8	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
		USN-9	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-3
		USN-10	As a user, I can register for the application through Gmail	I can register for the application through Gmail	Medium	Sprint-2
	Login	USN-11	As a user, I can log into the application by entering email & password	I can log in by entering Gmail & password	High	Sprint-1
	Dashboard	USN-12	As a user, I can track data of sales of products and inventory levels	I can track data of sales of products and inventory levels.	High	Sprint-1
Customer Care Executive	Support	USN-13	As a Executive, I Provide answers for the queries asked by users.	I provide the answers for the queries asked by the users.	High	Sprint-1

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Sprint	Functional Requirement(Epic)	User Story Number	User Story / Task	Story point	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	1	High	Surya, Naveenraja, Vigneshwaran, Pradeepan
Sprint-1		USN-2	As a user, I can register for the application through E-mail	4	Medium	Surya, Naveenraja, Vigneshwaran, Pradeepan
Sprint-1	Confirmation	USN-3	As a user, I will receive confirmation email once I have registered for the Application	2	Medium	Surya, Naveenraja, Vigneshwaran, Pradeepan

Sprint-1	Login	USN-4	As a user, I can log into the application by entering email & password	2	High	Surya, Naveenraja, Vigneshwaran ,Pradeepan
Sprint-2	Dashboard	USN-5	As a user, I can view the productswhich are available	10	High	Surya, Naveenraja, Vigneshwaran ,Pradeepan
Sprint-2	Add items to cart	USN-6	As a user, I can add the products Iwish to buy to the carts.	10	Medium	Surya, Naveenraja, Vigneshwaran Pradeepan
Sprint-3	Stock Update	USN-7	As a user, I can add products which are not available in the dashboard tothe stock list.	20	Medium	Surya, Naveenraja, Vigneshwaran ,Pradeepan
Sprint-4	Request to Customer Care and email alert	USN-8	As a user, I can contact the Customer Care Executive and requestany services I want from the customer care.	10	Low	Surya, Naveenraja, Vigneshwaran ,Pradeepan
Sprint-4	Contact Administrator	USN-9	I can be able to report any difficultiesI experience as a report	10	Medium	Surya, Naveenraja, Vigneshwaran ,Pradeepan

6.2 SPRINT DELIVERY SCHEDULE

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	27 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	1 Nov 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

6.3 REPORTS FROM JIRA

Jira generally defined together with everything from software development and customer service to start-ups and companies. Jira assists teams in planning, assigning, recording, documenting, and controlling work.
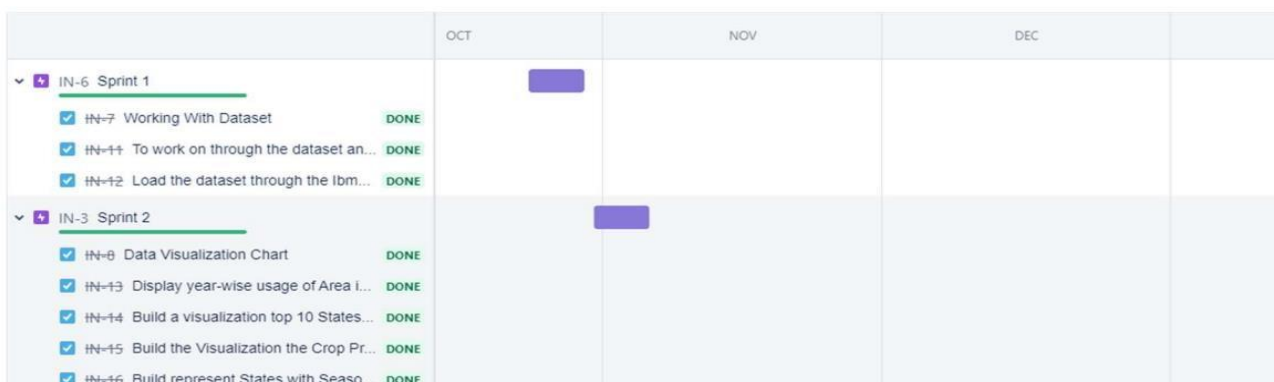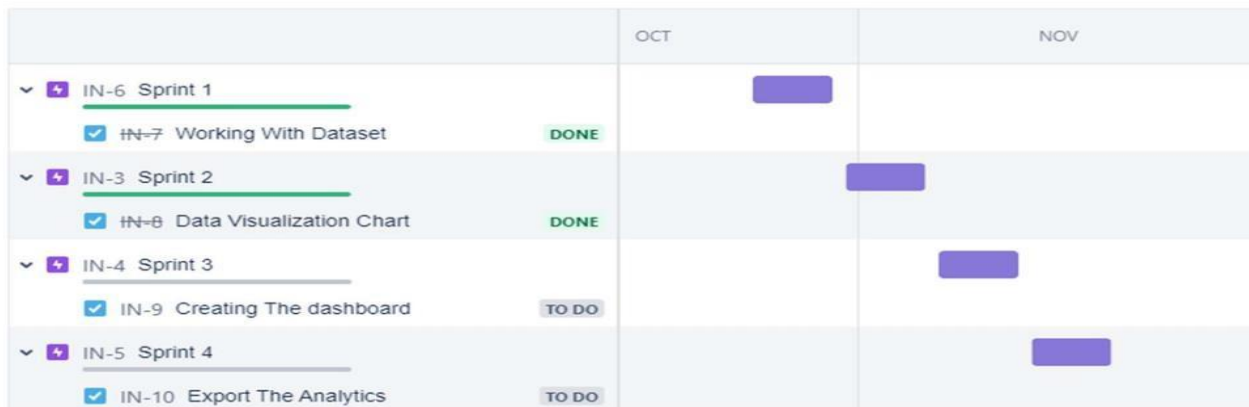

Fig 6.3.1 REPORTS FROM JIRA

CHAPTER 7

CODING & SOLUTION

7.1 CODE

from flask import Flask, render_template, flash, redirect, url_for, session, request, logging

from wtforms import Form, StringField, TextAreaField, PasswordField, validators, SelectField, IntegerField

import ibm_db

from passlib.hash import sha256_crypt

from functools import wraps

from sendgrid import *

#creating an app instance

app=Flask(name_)

app.secret_key='a'

conn=ibm_db.connect("DATABASE=

bludb;HOSTNAME=19af6446-6171-

4641-8aba-

9dcff8e1b6ff.c1ogj3sd0tgtu0lqde00.dat

abases.appdomain.cloud;PORT=30699

;SECURITY= SSL;

SSLServerCertificateDigiCertGlobalRootCA.crt;PROTOCOL=TCPIP;UID=ncl66473;PWD
=r2Ciii0CiSVri1qq;", "", "")

#Index

@app.route('/')

def index():

```python
    return render_template('home.html')


#Products

@app.route('/products')

def products():

    sql = "SELECT * FROM products"

    stmt = ibm_db.prepare(conn, sql)

    result=ibm_db.execute(stmt)

    products=[]

    row = ibm_db.fetch_assoc(stmt)

    while(row):

        products.append(row)

        row = ibm_db.fetch_assoc(stmt)

    products=tuple(products)

    #print(products)

    if result>0:

        return render_template('products.html', products = products)

    else:

        msg='No products found'

        return render_template('products.html', msg=msg)
    #Locations
```

```python
@app.route('/locations')

def locations():

sql = "SELECT * FROM locations"

stmt = ibm_db.prepare(conn, sql)

result=ibm_db.execute(stmt)

locations=[]

row = ibm_db.fetch_assoc(stmt)

while(row):

 locations.append(row)

row = ibm_db.fetch_assoc(stmt)

locations=tuple(locations)

#print(locations)

 if result>0:

     return render_template('locations.html', locations = locations)

   else:

     msg='No locations found'

     return render_template('locations.html', msg=msg)

#Product Movements

@app.route(/product_movements)

def product_movements():
```

```python
sql = "SELECT * FROM productmovements"

stmt = ibm_db.prepare(conn, sql)

result=ibm_db.execute(stmt)

movements=[]

row = ibm_db.fetch_assoc(stmt)while(row):

movements.append(row)

row = ibm_db.fetch_assoc(stmt)

movements=tuple(movements)
#print(movements)

    if result>0:

    return render_template('product_movements.html', movements = movements)

  else:

    msg='No product movements found'

    return render_template('product_movements.html', msg=msg)
#Register Form  Class

  RegisterForm(Form):
#print(name)

    name = StringField('Name', [validators.Length(min=1, max=50)])

  username = StringField('Username', [validators.Length(min=1,

  max=25)])email = StringField('Email', [validators.length(min=6,

  max=50)])
```

```python
        password = PasswordField('Password', [ validators.DataRequired()]),

        validators.EqualTo('confirm', message='Passwords do not match')])

      confirm = PasswordField('Confirm Password')

#user register

  @app.route('/register', methods=['GET','POST'])

  def register():

      form = RegisterForm(request.form)

      if request.method == 'POST' and form.validate():

        name = form.name.data

        email = form.email.data

        username = form.username.data

        password = sha256_crypt.encrypt(str(form.password.data))

sql1="INSERT INTO users(name, email, username, password) VALUES(?,?,?,?)" stmt1

= ibm_db.prepare(conn, sql1)

        ibm_db.bind_param(stmt1,1,name)

        ibm_db.bind_param(stmt1,2,email)

        ibm_db.bind_param(stmt1,3,username)

        ibm_db.bind_param(stmt1,4,password)

        ibm_db.execute(stmt1)

        #for flash messages taking parameter and the category of message to be flashed
```

```python
        flash("You are now registered and can log in", "success")

    #when registration is successful redirect to homereturn

    redirect(url_for('login'))

        return render_template('register.html', form = form)

    #User login

      @app.route('/login', methods = ['GET', 'POST'])

      def login():

        if request.method == 'POST':

    #Get form fields

    username = request.form['username']

    password_candidate = request.form['password']

    sql1="Select * from users where username = ?"stmt1

    = ibm_db.prepare(conn, sql1)

    ibm_db.bind_param(stmt1,1,username)

    result=ibm_db.execute(stmt1)

    d=ibm_db.fetch_assoc(stmt1)

        if result > 0:

#Get the stored hashdata = d

    password = data['PASSWORD']
```

```python
    #compare passwords

if sha256_crypt.verify(password_candidate, password):

#Passed session['logged_in'] = True session['username'] = username

    flash("you are now logged in","success")

return redirect(url_for('dashboard'))

 else:

    error = 'Invalid Login'

    return render_template('login.html', error=error)

 #Close connection

    cur.close()

    else:

      error = not found

    return render_template('login.html', error=error)return

    render_template('login.html')
```

CHAPTER 8

TESTING

8.1 TEST CASES

Testing is done to detect for problems. Testing is the process of looking for any flaws or shortcomings in a piece of work. It offers a means of checking whether parts, sub-assemblies, assemblies, and/or a finished product perform properly. It is the way of verifying software to make sure that it satisfies user expectations and meets requirements without failing in an unacceptable way.
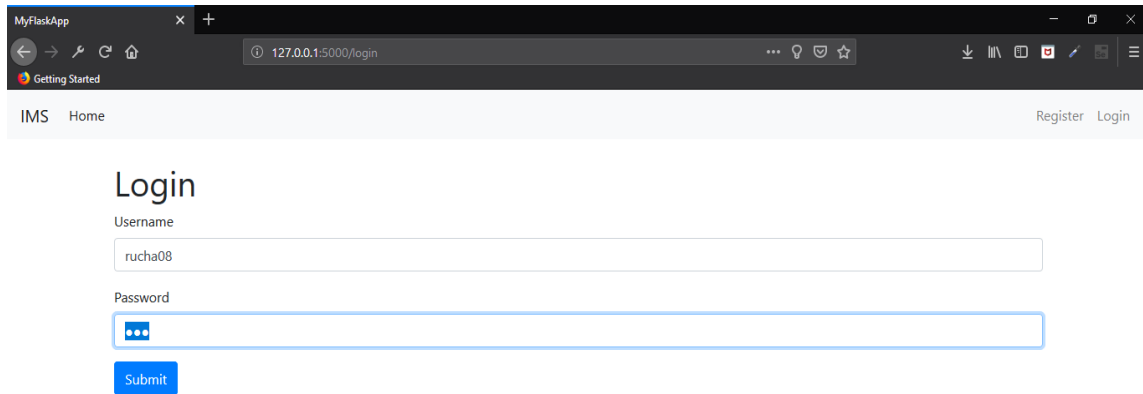
8.2 USER ACCEPTANCE TESTING

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets thefunctional requirements. All test cases are run at this point to ensure that the program is right and complete. The test must be completed successfully before the program can be accepted by the customer. The customer formally approves the delivery of this system after customer workers have checked that the preliminary production statistics load is correct and that the test suite has been achieved with perfect results.
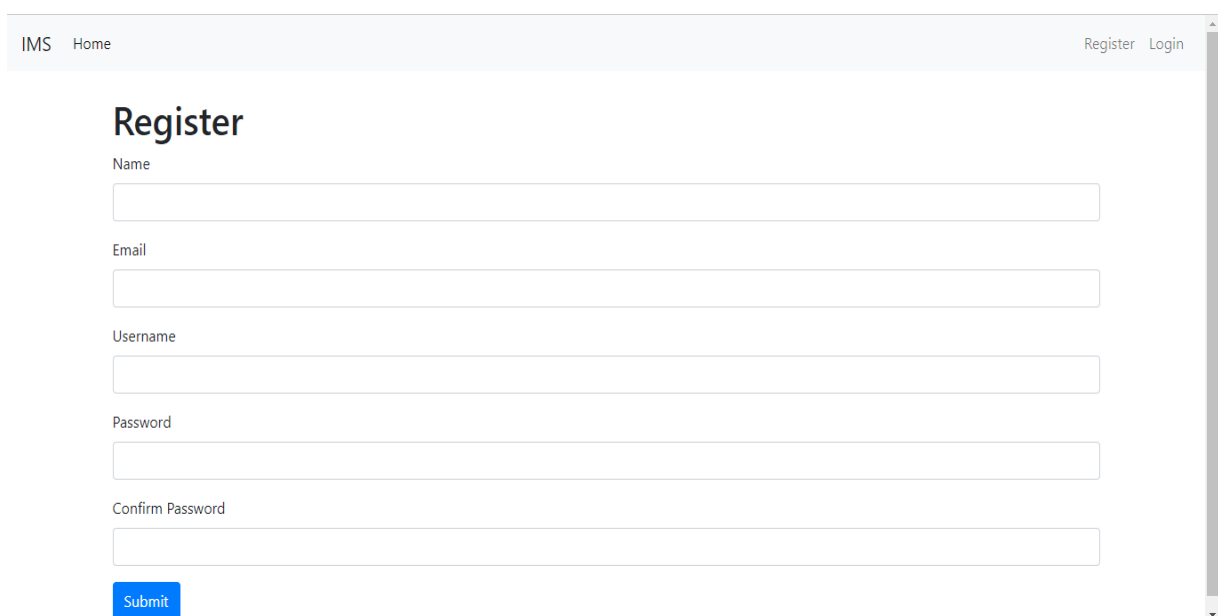
CHAPTER 9

RESULTS

9.1 OUTPUT
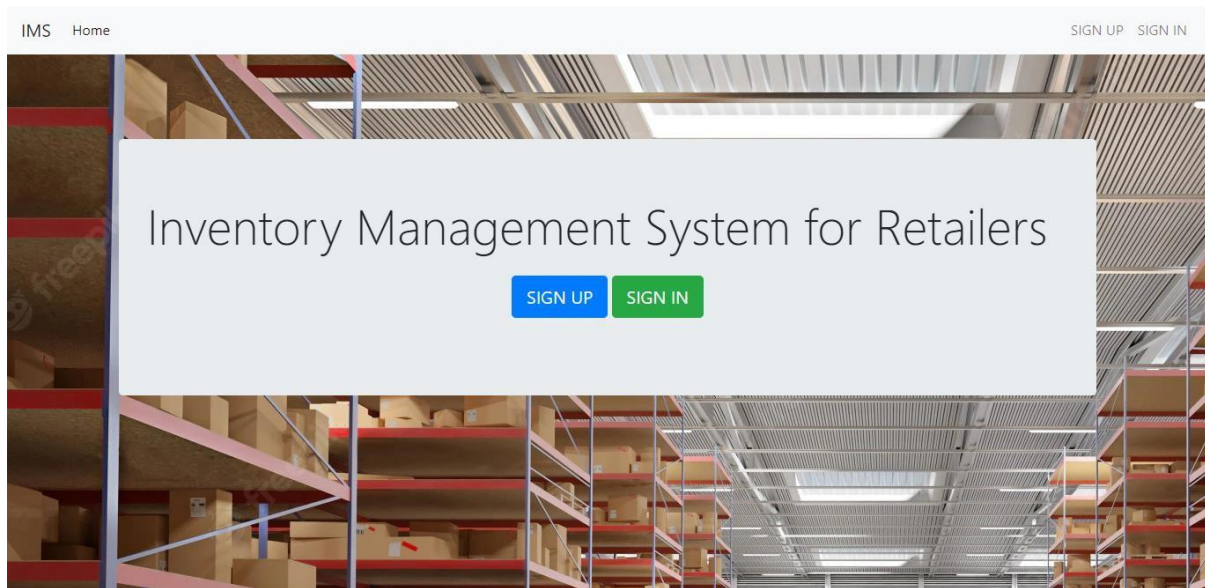

Fig 9.1.1 LOGIN

Fig 9.1.2 REGISTRATION PAGE
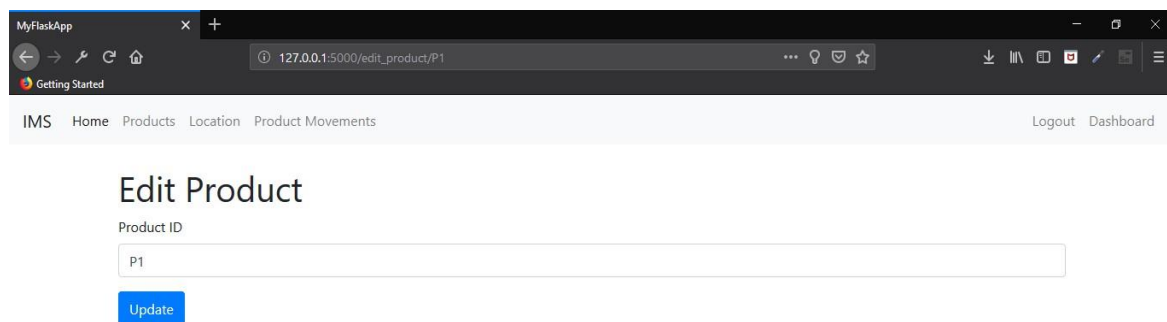
Fig 9.1.3 HOME PAGE
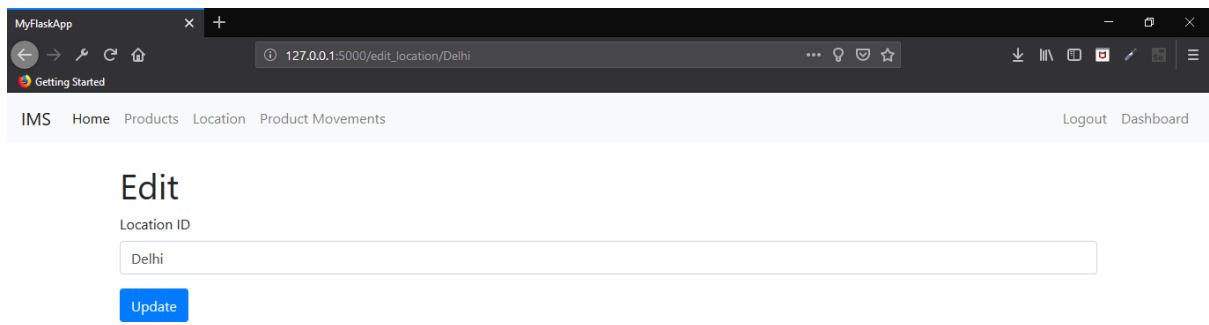

Fig 9.1.4 ADD PRODUCT
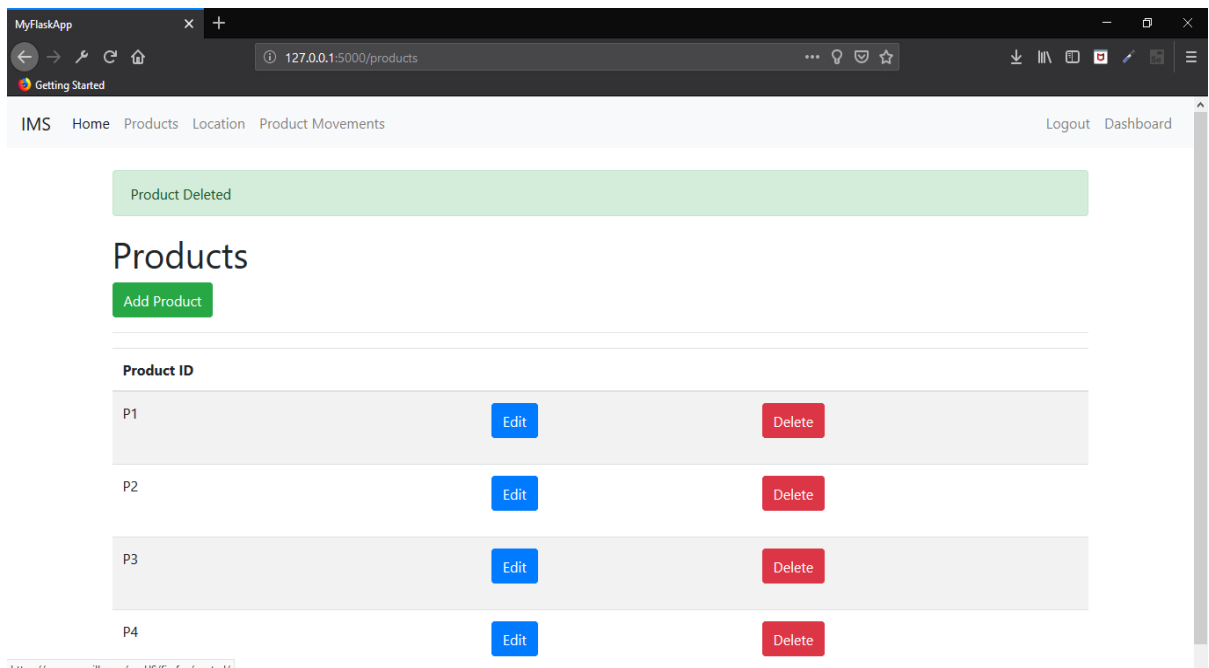
Fig 9.1.5 EDIT LOCATION
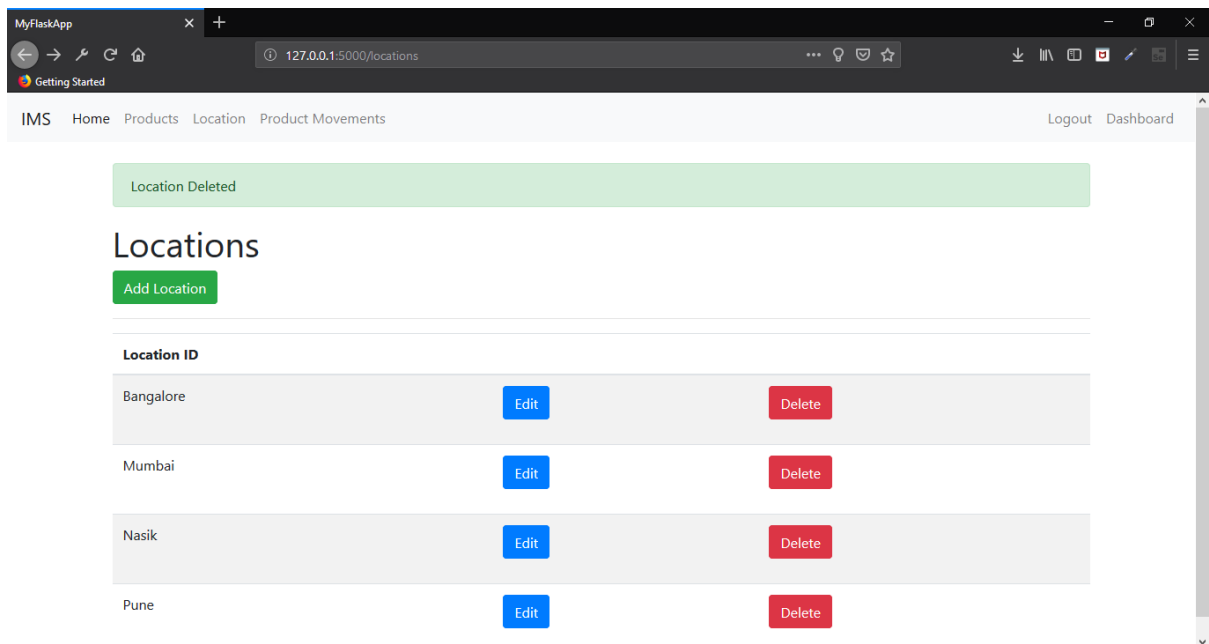

Fig 9.1.6 DELETE PRODUCT
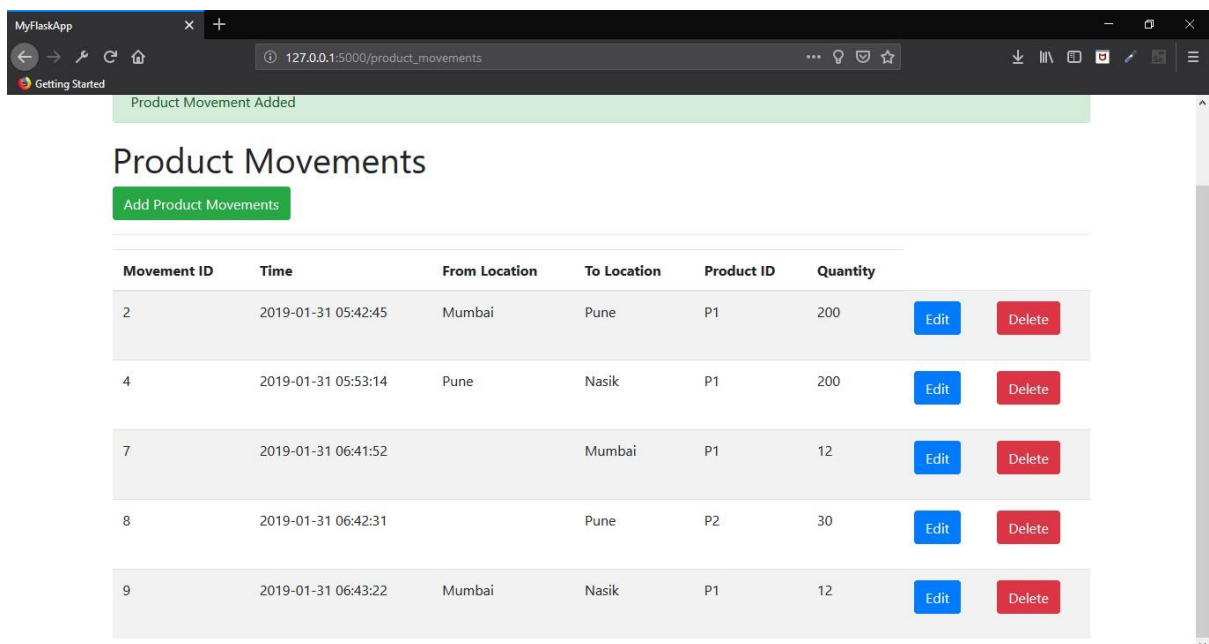
Fig 9.1.7 DELETE LOCATION

Fig 9.1.8 NEW PRODUCT

CHAPTER 10

10.1 ADVANTAGES

- Despite common assumption, inventory management does not aim to minimise the amount of inventory, which aids in maintaining the proper level of inventories.
- It results in a more organised warehouse since you can easily organise your warehouse with the help of an effective inventory management system.
- A successful inventory management system can result in time and financial savings for the company.

10.2 DISAVANTAGES

- The absence of a personal touch is another drawback of inventory management.
- Despite the fact that inventory management enables staff members at all levels of the organisation to view and change company stock and product inventory, the infrastructure needed to create such a system adds an additional layer of bureaucracy to the process in general and the business in particular.
- Inventory management can show you how much merchandise you have on hand and how much has been sold off, but it can also conceal manufacturing issues that could have disastrous effects on customer service.

CHAPTER 8

CONCLUSION

It goes without saying that the manual inventory system needs to be replaced with an automated inventory system that can streamline the procedure, reduce human error, and boost inventory effectiveness. The software of choice can also fulfil the requirements laid out in order to resolve the problem. The function and feature of the system that was used to generate the software are regarded to be the best ways to achieve the project's objective. The hardware system, which is the cash machine, must be integrated with the supply chain for its use to be coordinated with the cash machine. In the future, the system can be improved by adding a section for store papers and bringing a new transaction.

CHAPTER 12

FUTURE SCOPE

Inventory management functions as the point of convergence for the supply chain for many businesses. Any competitive business can be regarded as having inventory as its lifeblood. Any business that deals in physical commodities needs it to flow freely in order to function effectively. Inventory has value for manufacturers, marketing, and sales just like the blood provides the life-sustaining "oxygen" to various regions of the human body. A look at the inventory of a corporation can reveal a lot about it, much like the expert who takes a few blood samples from a patient's arm to assess their health and wellness. Unnecessary inventory build up is frequently a sign of more serious issues in a company's infrastructure and operational procedures.

CHAPTER 13

APPENDIX

13.1 SOURCE CODE

```html
<!DOCTYPE html>
<html lang="en" dir="ltr">
 <head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title> inventory</title>
      <style>
      @import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@200;300;400;500;600
;700&display=swap');
*{
 margin: 0;
 padding: 0;
 box-sizing: border-box;
 font-family: 'Poppins',sans-serif;
}
::selection{
 color: #000;
 background: #fff;
}
nav{
 position: fixed;
 background: #1b1b1b;
 width: 100%;
 padding: 10px 0;
 z-index: 12;
}
nav .menu{
 max-width: 1250px;
```

```css
  margin: auto;
  display: flex;
  align-items: center;
  justify-content: space-between;
  padding: 0 20px;
}
.menu .logo a{
  text-decoration: none;
  color: #fff;
  font-size: 35px;
  font-weight: 600;
}
.menu ul{
  display: inline-flex;
}
.menu ul li{
  list-style: none;
  margin-left: 7px;
}
.menu ul li:first-child{
  margin-left: 0px;
}
.menu ul li a{
  text-decoration: none;
  color: #fff;
  font-size: 18px;
  font-weight: 500;
  padding: 8px 15px;
  border-radius: 5px;
  transition: all 0.3s ease;
}
.menu ul li a:hover{
  background: #fff;
  color: black;
```

```css
}
.img{
 background: url('img3.jpg')no-repeat;
 width: 100%;
 height: 100vh;
 background-size: cover;
 background-position: center;
 position: relative;
}
.img::before{
 content: '';
 position: absolute;
 height: 100%;
 width: 100%;
 background: rgba(0, 0, 0, 0.4);
}
.center{
 position: absolute;
 top: 52%;
 left: 50%;
 transform: translate(-50%, -50%);
 width: 100%;
 padding: 0 20px;
 text-align: center;
}
.center .title{
 color: #fff;
 font-size: 55px;
 font-weight: 600;
}
.center .sub_title{
 color: #fff;
 font-size: 52px;
 font-weight: 600;
```

```css
}
.center .btns{
 margin-top: 20px;
}
.center .btns button{
 height: 55px;
 width: 170px;
 border-radius: 5px;
 border: none;
 margin: 0 10px;
 border: 2px solid white;
 font-size: 20px;
 font-weight: 500;
 padding: 0 10px;
 cursor:  pointer;
 outline: none;
 transition: all 0.3s ease;
}
.center .btns button:first-child{
 color: #fff;
 background: none;
}
.btns button:first-child:hover{
 background: white;
 color: black;
}
.center .btns button:last-child{
 background: white;
 color: black;
}
</style>
   <script src="https://kit.fontawesome.com/a076d05399.js"></script>
  </head>
```

```
<body>
 <nav>

  <div class="menu">
   <div class="logo">

    <a href="#">IMS & CO</a>
   </div>
   <ul>
    <li><a href="login.html">Log In</a></li>

    <li><a href="reg.html">sign Up</a></li>

    <li><a href="home.html">Log Out</a></li>

    <li><a href="#">Contact</a></li>
   </ul>
  </div>
 </nav>
 <div class="img"></div>
 <div class="center">
  <div class="title">Inventory management system for retailers</div>
  <div class="sub_title">Stock Management</div>
  <div class="btns">
   <button>Learn More</button>
  </div>
 </div>
</body>
</html>
```

13.2 GITHUB & PROJECT DEMO LINK

https://github.com/IBM-EPBL/IBM-Project-1185-1658377429