INVENTORY MANAGEMENT–A WEB APPLICATION TO MAINTAIN A SHOP INVENTORY

PROJECT REPORT

Submitted by

SABARI R (711619104037)

GANESH KUMAR M (711619104010)

SEENIVASAN V (711619104044)

SUBBU KUTTY B (711619104049)

In partial fulfillment for the award of the degree

Of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

KATHIR COLLEGE OF ENGINEERING

“WISDOM TREE”, NEELAMBUR, COIMBATORE-641062

ANNA UNIVERSITY: CHENNAI 600025

Project Report Format

1. **INTRODUCTION**
    a. Project Overview
    b. Purpose
2. **LITERATURE SURVEY**
    a. Existing problem
    b. References
    c. Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
    a. Empathy Map Canvas
    b. Ideation & Brainstorming
    c. Proposed Solution
    d. Problem Solution fit
4. **REQUIREMENT ANALYSIS**
    a. Functional requirement
    b. Non-Functional requirements
5. **PROJECT DESIGN**
    a. Data Flow Diagrams
    b. Solution & Technical Architecture
    c. User Stories
6. **PROJECT PLANNING & SCHEDULING**
    a. Sprint Planning & Estimation
    b. Sprint Delivery Schedule
    c. Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**

a. Feature 1

b. Feature 2

c. Database Schema (if Applicable)

8. **TESTING**

a. Test Cases

b. User Acceptance Testing

9. **RESULTS**

a. Performance Metrics

10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**

Source Code

GitHub & Project Demo Link

CHAPTER-1

INTRODUCTION

1.1 Project Overview:

This is the basic advance level of inventory management here in inventory management we can help people to manage their own shops inventory from where ever they want. They only need a required system with the software installed if they want, they can just use our web application in order to use our inventory management system. Internet is mandatory in order to access and work in your inventory management account , instead of working shifted times they can monitor their inventory form their home or from their vocation where ever the place they want .This will drastically reduce the travelling expenses of the user instead of travelling to their shop they can simply check that out from their house itself, And it also makes the work very easier it shows in the tabled manner so that the user don't need to check on the inventory system to know the current stock in the inventory .The stock needed will be mailed to the user so that they don't want to check the inventory regularly instead the warning is given to the user by the system.

1.2 Purpose:

By following this inventory management system, the work of the user is minimized to none and the comfort of the user is increased and he doesn't need to monitor the work done or the work happening in the inventory the whole work is done by the inventory management system. He only need to refill the stock in the inventory. So that the stock in the inventory doesn't exhaust and the user can peacefully carry on his business. The inventory management system not only manages the stock of the

inventory but also helps the user to stay updated in the stock and the amount of stock needed in the inventory so that the user can keep up on it. The refiling must be done manually by the user the inventory management system can only able to intimate the user, The rest is in the user's hand they need to take care of the amount of stock to be refilled and contact the required dealer in order to buy the stock and update the stock in his own inventory in order to avoid the continuous warning of the inventory management system**.**

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 main use of the inventory management is to serve the people and to make the work easier. In the current scenario, if a customer does not find the desired merchandise at one retail shop, he has a second brand to rely on. A retailer can't afford to lose even a single customer.

CHAPTER-2

LITERATURE SURVEY

A literature survey is a comprehensive summary of previous research on a topic. The literature survey surveys scholarly articles, books ,and other sources relevant to a particular area of research .The survey should enumerate, describe, summarize, objectively evaluateand clarity this previous search.

A literature survey is an overview of the previously published works on a topic. The term can refer to a full scholarly paper or a **section** of a scholarly work such as a book, or an article. Either way, a literature survey is supposed to provide the researcher/author and the audiences with a general image of the existing knowledge on the topic under question. A good literature survey can ensure that a proper research question has been asked and a proper theoretical framework and/or research methodology have been chosen. To be precise, a literature survey serves to situate the current study within the body of the relevant literature and to provide context for the reader. In such case, the survey usually precedes the methodology and results sections of the work.

2.1 Existing Problem:

The main problem in inventory management is to maintain the stock in the inventory the user finds it difficult to maintain the stock in the inventory all by himself so he needs an assistance in the work in order to maintain the amount of stock in the inventory so that the shop will rum with the stock in the inventory. The main problem arises is the trust issues in the inventory so that the owner doesn’t able to trust the employee working in the inventory so, He needs to supervise himself so that the mistakes won’t happen in the inventory and the employee may have the power to move

without the knowledge of the owner is less, this may cause delay in decision making and may lead to decrease in the efficiency of the inventory management system. And the owner may or may not be reachable at the moment may lead to the surplus of goods in the inventory management so the main problem is the owner should need an assistance an exact assistance to guide him out and carry out the inventory. This makes the work a lot more easier and reduces the mental pressure to the owner or the user, so they can work in mental peace and the efficiency of the work is increased.

2.2 The below shown are the reference of the inventorymanagement:

Abramovitz and Modigliani (1957)

They highlighted the relationship between capacity utilization and inventory investment. Existing stock of inventories was expected to adjust to the desired levels. Thus, the variable existing stock of inventories, was essential to be negatively related with the desired stock. The result was that there is positive relation among the ratio of inventory to sales and inventory investment. High ratio of stocks to sales in the past suggests requirement of high levels of inventories in the past and promising high investment in inventories in the current period also.

Krishna Murthy (1964)

Study was aggregative and dealt with inventories in the private sector of Indian economy as a whole for the period 1948-61. This study used sales to represent demand for the product and suggested the importance of accelerator. Short term rate of interest had also been found to be significant.

R.S. Chadda (1964)

Study had been made on inventory management practices of Indian companies. The analysis suggested application of modern scientific inventory control techniques like operations research. These modern scientific techniques furnish opportunities for the companies, Companies can minimize their investment in inventory but there is

continuous flow of production. He argued that industrially advanced countries, like, USA, were engaged in developing highly sophisticated mathematical models and techniques for modernizing and redefining the existing tools of inventory investment.

National Council of Applied Economic Research (NCAER) (1966)

Conducted a study in 1966 regarding working capital management of three industries namely cement, fertilizer and sugar. This study mainly devoted to ratio analysis of composition, utilization and financing of working capital for the period of 1959 to 1963. The study reveals that inventory constituted a major portion of working capital i.e. 74.06 per cent in the sugar industry followed by cement industry (63.1%) and fertilizer industry (59.58%). It was observed that inventory had not managed properly. So far as the utilization of working capital was concerned, cement and fertilizer industry had better implementation of working capital. The sugar industry had huge accumulation of stocks so there was inefficient utilization of working capital heavily.

Krishnamurty and Sastry (1970)

It is the most comprehensive study on manufacturers' inventories. They used the CMI data and the consolidated balance sheet data of public limited companies published by the RBI, in order to analyse each of the major components, like the raw materials, goods-in-process and finished goods, for 21 industries over the period ranging from 1946-62. The study was a time series one although there were some inter-industry cross-section analyses that were carried out in the analysis. The utilisation of productive capacity and price anticipations was also found to be relevant in the study.

George (1972)

It was the study on cross section analysis of balance sheet data of 52 public limited companies for the period of 1967- 70.

Accelerator, internal and external finance variables were considered in the formulation of equations for raw materials including goods-in-process inventories. However, equations for finished goods inventories conceive only output variable. Deliberation was given on accelerator and external finance variables.

Mishra (1975)

It is the study of six major public sector enterprises. He concluded that Inventory constitutes the most important component of working capital of public enterprises (ii) efficiency of working capital funds employed in receivables is terribly low in the selected enterprises and (iii) In all units both the current assets and the quick ratios are greater than their standards. Enterprises need proper control on receivables.

Lambrix and Singhvi (1979)

Adopted working capital cycle approach in working capital management, also suggested that investment in working capital can be optimized and cash flows can be improved by reducing the time frame of physical flow starting from the receipt of raw material to the shipment of finished goods, i.e. inventory management, and by improving the terms and conditions on which firm sells goods as well as receipt of cash

Lal (1981)

He studied Modi Steels Limited as a case study, his study focused on inventory management. He originated a model which involve price variable in inventory management; earlier price variable in inventory was not considered in that company. The analysis recommended solid policies, which would look after internal and external factors, ultimately it would help in bringing in efficient working capital

management.

Farzaneh (1997)

Presented a mathematical model, to assist the companies in their decision to switch from EOQ to JIT purchasing policy. He defines JIT as “to produce and deliver finished goods just in time to be sold, sub-assemblies just in time to be assembled in goods and purchased material just in time to be transformed into fabricated parts”. He highlights that the EOQ model focuses on minimizing the inventory costs rather than minimizing the inventory.

Gaur, Fisher and Raman (2005)

In their study examined firm-level inventory behaviour among retailing companies. They took a sample of 311 public-listed retail firms for the years 1987–2000 to examine the relationship of inventory turnover with gross margin, capital intensity and sales surprise. They observed that inventory turnover for retailing firms was positively related to capital intensity and sales surprise while inversely associated with gross margins. They also suggested models that yield an alternative metric of inventory productivity, adjusted inventory turnover that can be used in study of performance analysis and managerial decision-making.

S. Singh (2006)

Analysed the inventory control practices of single fertilizer company named IFFCO. He statistically examined the inventory system with consumption, sales and other variables along with growth of these variables and inventory patterns. He concluded that an increase in components of inventory lead to an increase in the proportion of inventory in current assets. A special focus was made on stores and spares in order to

calculate excess purchases resulting in loss of profit.

Pradeep singh (2008)

In his study made an attempt to examine the inventory and working capital management of Indian Farmers Fertilizer Cooperative Limited (IFFCO) and National Fertilizer Limited (NFL). He concluded that the overall position of the working capital of IFFCO and NFL is satisfactory. But there is a need for improvement in inventory in case of IFFCO. however inventory was not properly utilized and maintained bay IFFCO during study period. The management of NFL must try to properly utilize the inventory and try to maintain the inventory as per the requirements. So that liquidity will not interrupt

Capkun, Hameri and Weiss (2009)

Statistically analysed the relationship between inventory performance and financial performance in manufacturing companies using the financial information of a large sample of US-based manufacturing firms over a 26-year period, that is, 1980 to 2005. They inferred that a significant relationship existed between inventory performance along with the performance of its components and profitability. Raw material inventory performance was highly correlated to gross profit and operating profit. Work in progress inventory was highly correlated to gross profit measures while finished goods inventory performance was more correlated with operating profit measures.

Gaur and Bhattacharya (2011)

Attempted to study the linkage between the performance of the components of inventory such as raw material, work in progress and finished goods and financial performance of Indian manufacturing firms. The study revealed that finished goods inventory as inversely associated with business performance while raw material inventory and work in

progress did not have much effect on same. They emphasised that instead of focusing on total inventory, an attempt should be made to concentrate on individual components of inventory so as to adequately manage the same.

2.3 problem statement definition:

To solve the need that the shopkeepers doesn't have the systematic way to keep their records of the inventory data. In inventory system, demand is usually uncertain and the lead-time can also vary. To avoid shortages, managers often maintain a safety stock. In such situations, it is not clear what order quantities and reorder points will minimize expected total inventory cost.

The above problem statement explains about the shop keepers have the need to maintain their inventory in the easier and efficient way so they can use it according to their convenience. They may lead to loss of records or the error in the records stored in it .And also, they can handle it only if they physically present there and they needed to store the records for the future reference else the information may be altered or they may be lost , they need to be protected so that, they can use it for big data analysis and use it accordingly in order to customize the amount of stock to be refilled in the inventory according to the user and the records may help in the future inference. They do not need to store any physical records with them the inventory management itself stores the records of the user in the dedicated cloud so that the user can use the inference of the data stored in the cloud according to their needs in their desired ways so that the user can utilize every resource in the inventory management system itself he /she doesn't need to take any back up of the documents or the sales records of the data nor the stock records too. It intimates the user to refill the inventory so the user just needs to be active in order to perform that action so the inventory doesn't need to refill it manually another time.

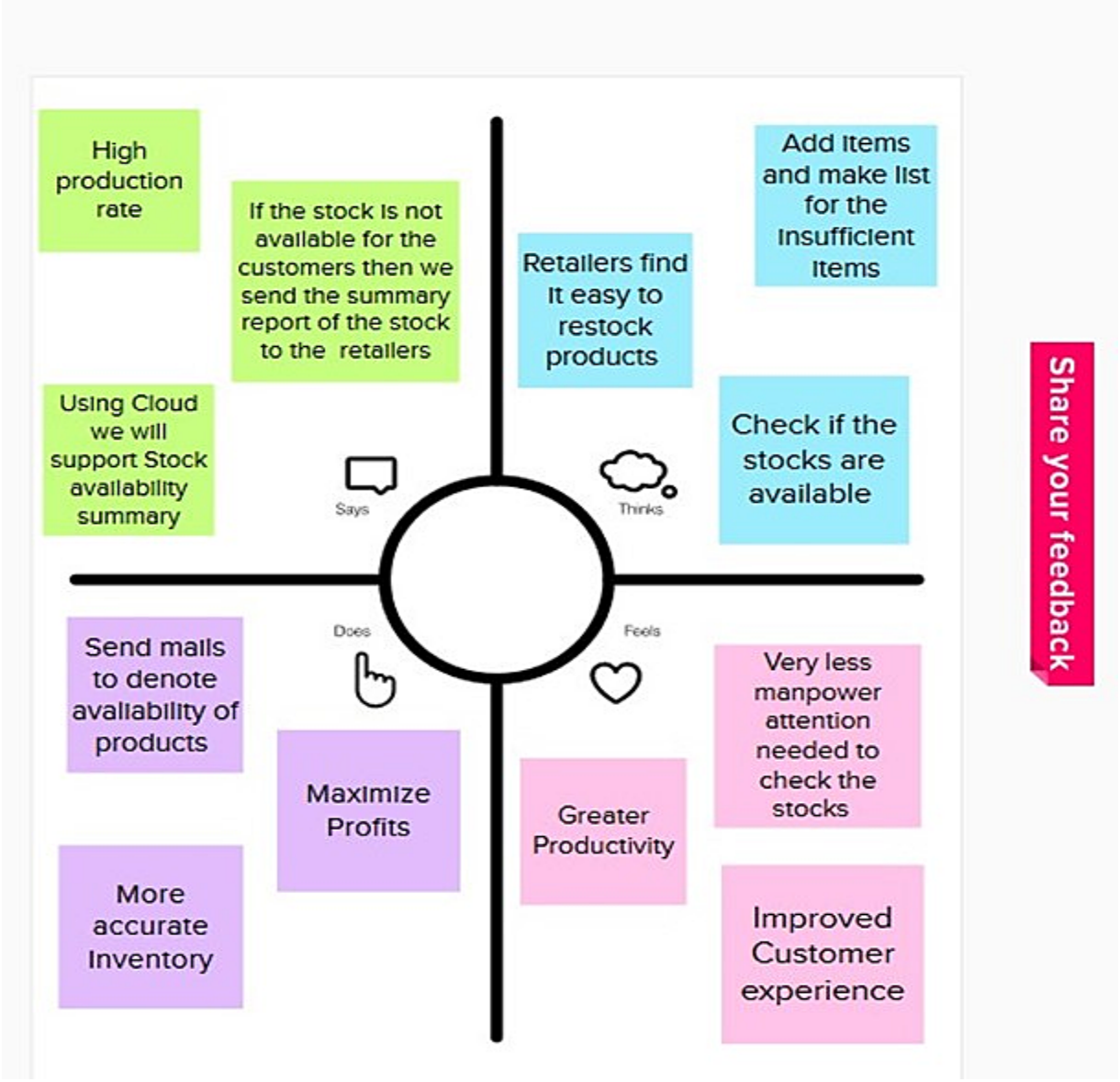
CHAPTER-3

IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas:

Empathy map is the graphical representation of the project. It can simply explain the process of the process so that the new comer may easily understand the undergoing process of the project and the personal opinion of the user.

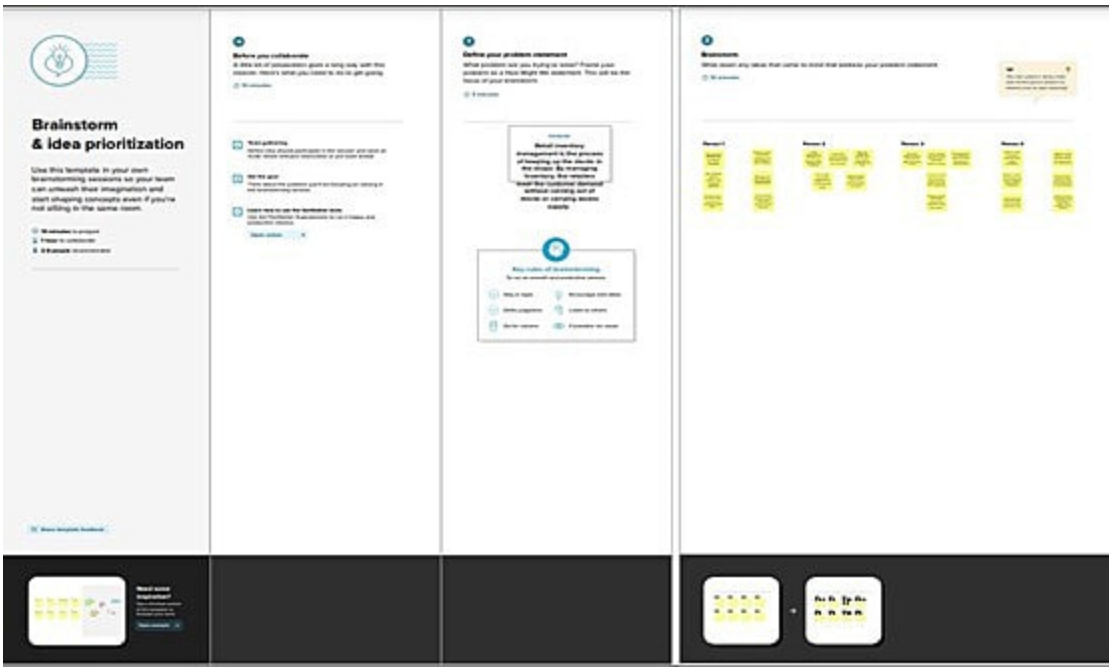
Empathy map



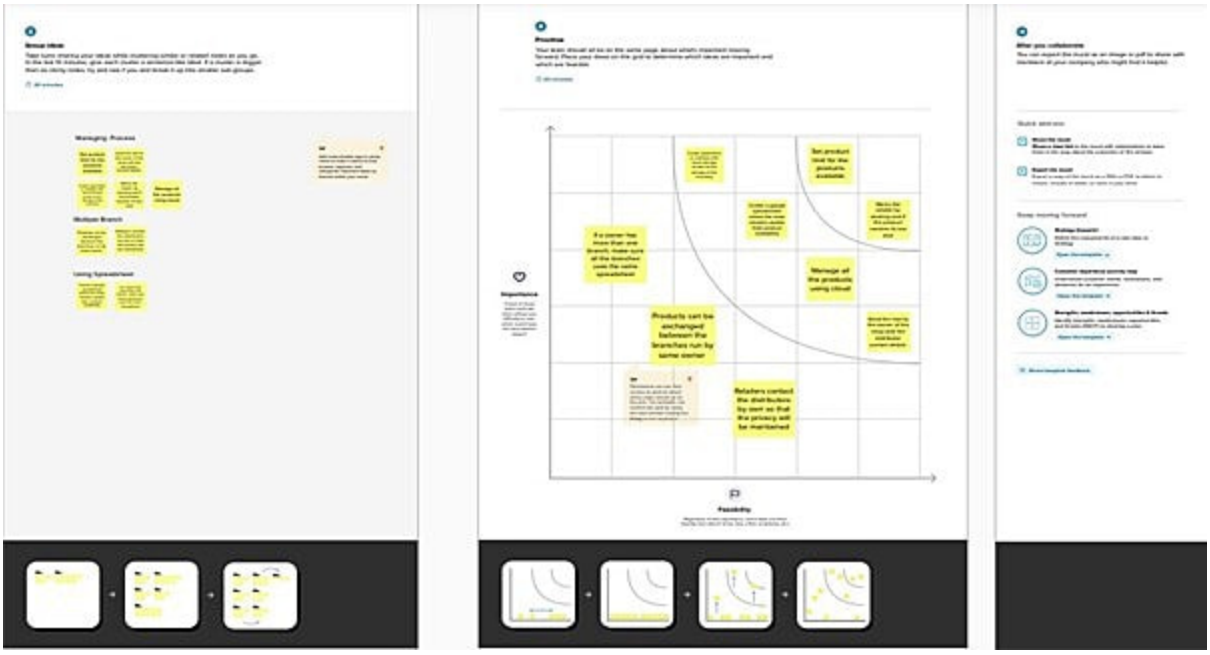
Empathy map by keeping user on their shoes

3.2 Ideation & Brainstorming:

Every plan doesn't execute as automatically every thing needs a plan in order to execute and it should be verified as it held according to the plan so one needs to be supervise the work in order to maintain the flow of the project.



Collaboration between the team members



Results after collaboration prioritize the ideas of the team members

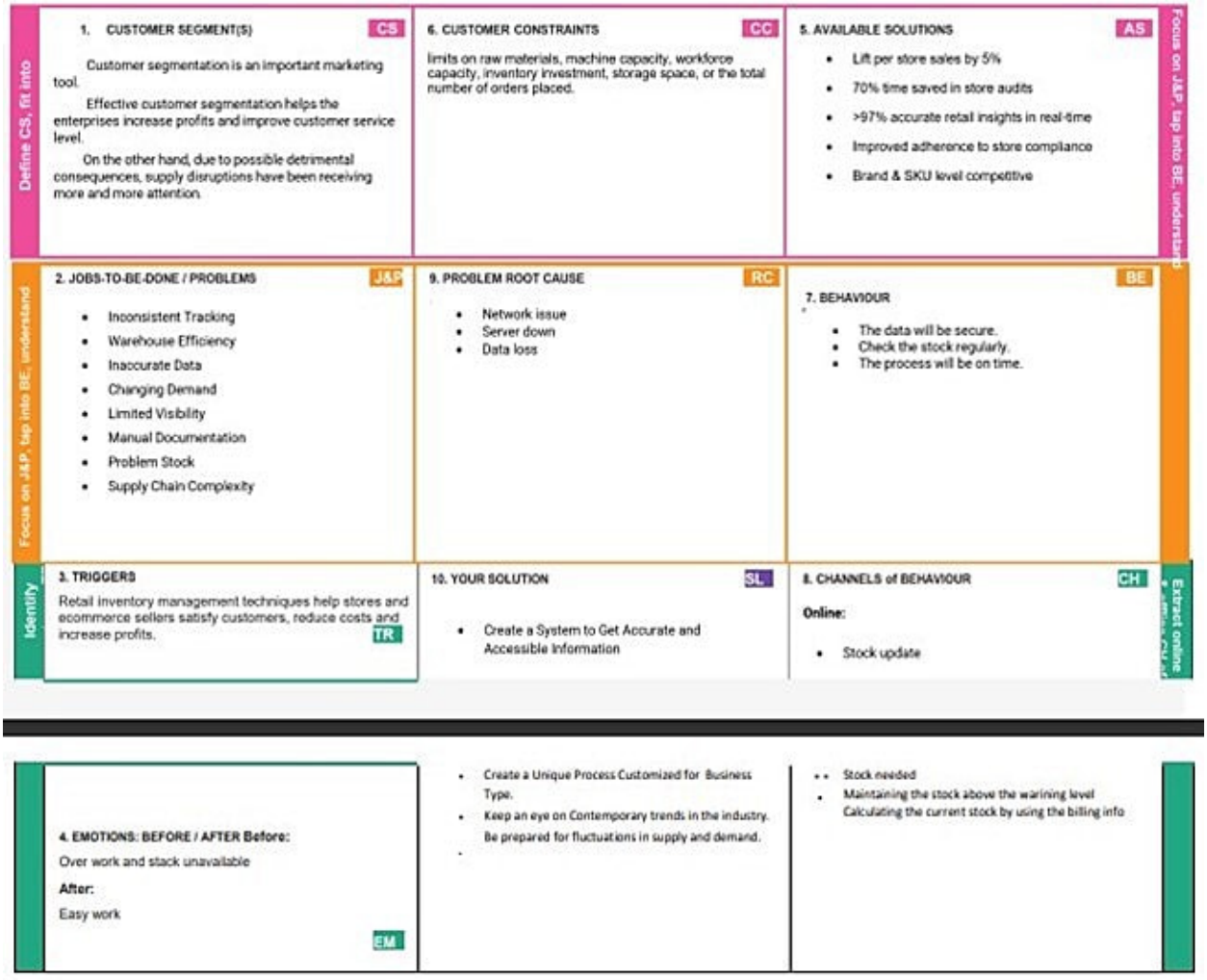
3.3Proposed solution:

Proposed solution is the rough idea of the project to give the multiple solution of the project and the best one is selected in order to obtain the best outcome of the project to produce the maximum efficiency.

S. No.	Parameter	Description
1	Problem Statement	To solve the need that the shopkeepers doesn't have the systematic way to keep their record of inventory data.
2	Idea / Proposed Solution	An application which retailers successfully log in to the application, that 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 the stock reduced to the limited amount found in the inventory. So that they can order new stock.
3	Novelty / Uniqueness	With this inventory management system, the shopkeeper not only can fill the inventory but also reduce the wastage of goods. The users can register the stocks that they need by logging in from their account.
4	Social Impact / Customer Satisfaction	Customer Satisfaction is entirely depend on the services which they expected. If the retailer's system exceeds with customer's expectation, the customers will be satisfied.

5	Business Model	With the better inventory management system, Update the inventory without any need of manpower. Retailer can live up with user's need and be on the flow with current sale products and they can update the inventory with that products.
6	Scalability of the Solution	To create a scalable inventory management system, the retailer have to 1. Keeping low inventory levels as much as possible 2. Keep an eye on Sales Projections 3. Use ODM (On-Demand Manufacturing). ODM refers to manufacture or in this case, update the products which are highly in demand.

3.4 Problem solution fit:



Innovative solutions based on the existing problem

CHAPTER-4

REQUIREMENT ANALYSIS

An easy-to-use interface that doesn't require advanced training, support or documentation. Automation for eliminating manual processes of business functions related to inventory management. A reliable, secure database that provides accurate, real-time data.

4.1 Functional Requirements:

1. **User Registration** - The Users can make the Registration through form and Registration through Email.
2. **User Confirmation** – The Users Get Confirmation via Email and Confirmation via OTP.
3. **Login –** Login to the application by entering the Email and Password.
4. **Dashboard –** By opening the Dashboard the list of products display and User can view the product availability.
5. **Add items to cart –** User as they wish to buy the product, they can select the product and add it to the cart.
6. **Stock Update** – If the desired product is unavailable, they can update the products into the list for buying products.

4.2 Non-Functional Requirements:

7. **Usability –** While usability determines how effective implementing an inventory tracking system is in your business. If it takes hours for your staff to learn the ins and outs of the software, then it’s probably not worth buying.

8. **Security -** The process of ensuring the safety and optimum management control of stored goods. 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.

9. **Reliability -** Relying on manual inventory counts to know what you have will only guarantee high inefficiencies and a loss of customers.
10. **Performance -** Creating systems to log products, receive them into inventory, track changes when sales occur, manage the flow of goods from purchasing to final sale and check stock counts.

11. **Availability -** Whether a specific item is available for customer orders. Additional information provided by retailers may include the quantity available.

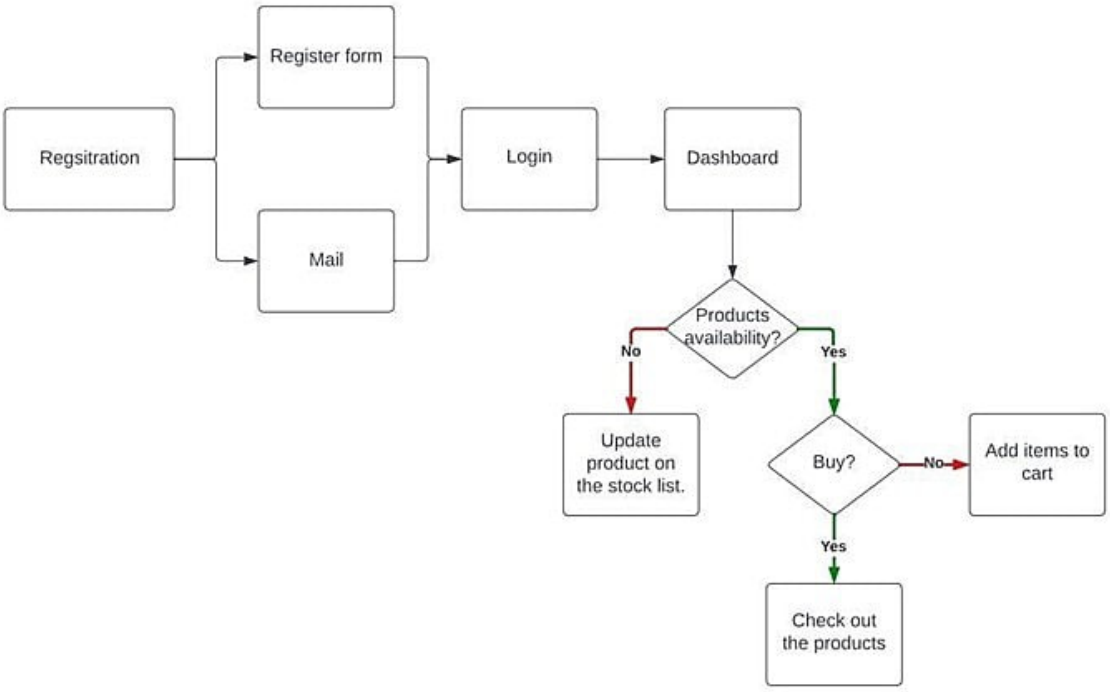
12. **Scalability -** They should use an automated inventory management system for inventory tracking. This will make your business much more scalable so that you can continue building consistent growth and take advantage of increased sales.

CHAPTER-5

PROJECT DESIGN

5.1 Data flow diagram:

The term data flow diagram shows the path of the traveling in the project done or explaining the total working of the project it also explains the path of the travel or the working process of the project and the process that held inside the project.



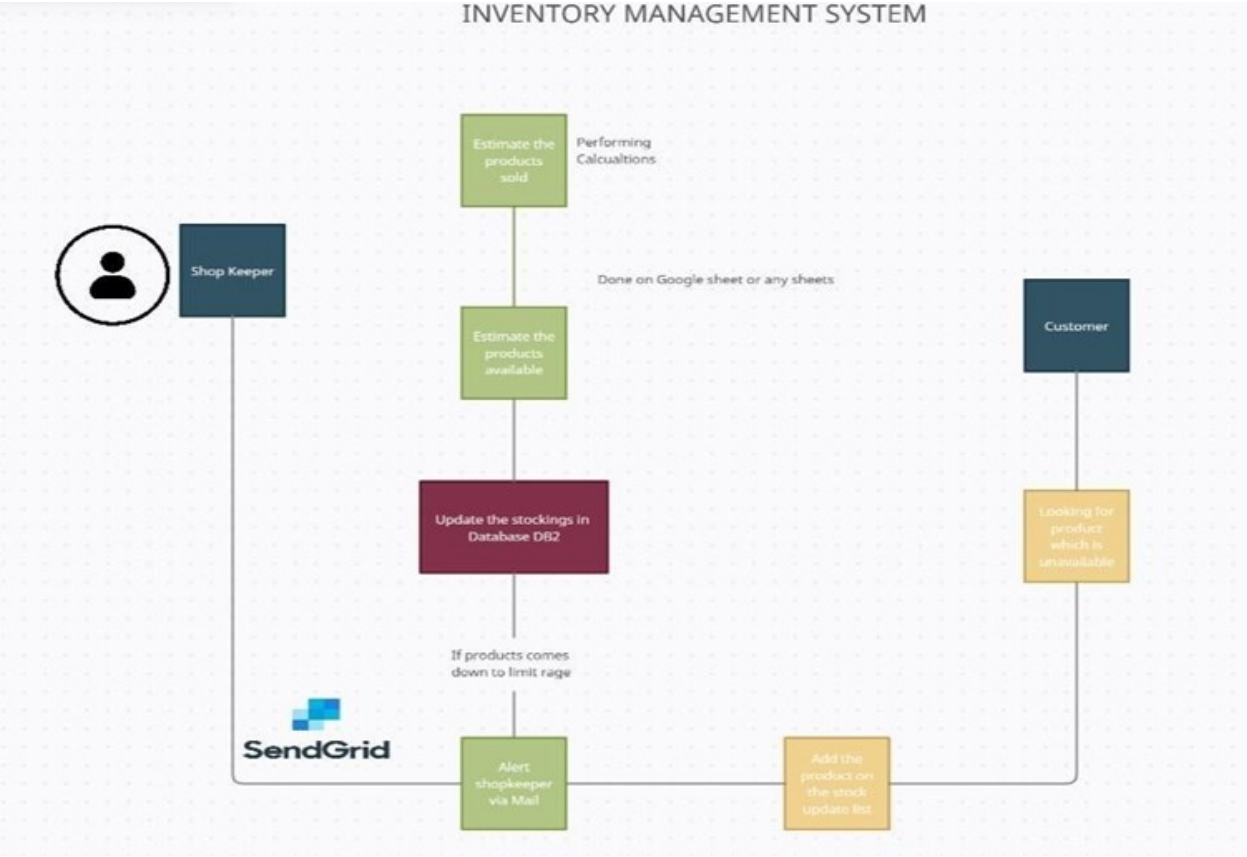
Data flow diagram for inventory management

The above shown diagram shows the data flow of the inventory

management system the user must register himself in order to work in the inventory management system. After registering himself he must login to the dashboard and check the product availability and decide whether to buy the product or not in case of buying they need to check on the product and they can add the desired product to their cart and they may proceed for buying. If they didn't find their desired product they are allowed to mention the product that they needed in order to make the product available for the next time.

5.2 Solution &Technical architecture:

The term solution and the technical architecture defined as the solution of the problem mentioned and the way that the solution that has to be implemented on the way in order to achieve the goal and to solve the problem.



Process of inventory management

The above shown diagram is the technical architecture of the inventory management system. It explains about the inventory management system so that tha user can run a shop using a send grid so according to the amount of stock given by the user the inventory management system warns the user via mail only if the product comes to the down range. After refilling the stock in the inventory it has to be updated in the cloud database db2.It has to estimate the available products after refilling it will count on the exhaustion of the product. It calculates the estimated products sold by the shop owners these calculations may be done on the google sheets or any other tabled format use to perform calculations. According with that it also do the customer survey or the product that customer wants that they didn't get now was noted they are added to the inventory with the acknowledgement of the user or the owner.

5.3 user stories:

The below mentioned are the user stories of the inventory management and the user experience of the individual so that each opinion may be taken under survey so that we can update the inventory management according to the desire of the customer so that they can even work efficiently and comfortably.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint1
		USN-2	As a user, I can register for the application through E-mail	I can access my account / dashboard	Medium	Sprint1

	Confirmation	USN-3	As a user, I will receive confirmation email once I have registered for the application	I can get confirmation for my email and password and create authenticated account.	Medium	Sprint1
	Login	USN-4	As a user, I can log into the application by entering email & password	I can log onto the application with verified email and password	High	Sprint1
	Dashboard	USN-5	As a user, I can view the products which are available	Once I log on to the application, I can view products to buy.	High	Sprint2

CHAPTER-6

PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

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	7	6 Days	24 Oct 2022	29 Oct 2022	7	29 Oct 2022
Sprint-2	9	6 Days	31 Oct 2022	05 Nov 2022	9	05 Nov 2022
Sprint-3	5	6 Days	07 Nov 2022	12 Nov 2022	5	12 Nov 2022
Sprint-4	10	6 Days	14 Nov 2022	19 Nov 2022	10	19 Nov 2022

Table of sprint planning

6.2 Sprint Delivery Schedule

This topic explains about the delivery info of the sprint and the time taken to complete the sprint separately.

6.2.1 Product Backlog, Sprint Schedule, and Estimation:

The above planned sprint must be delivered on time so the more important is to deliver the sprint correctly. so the product must satisfy the customer so a survey must be held and collect the needs of the user and priori the needs of the user and also note the requirements of the system. The above mentioned are tabulated below

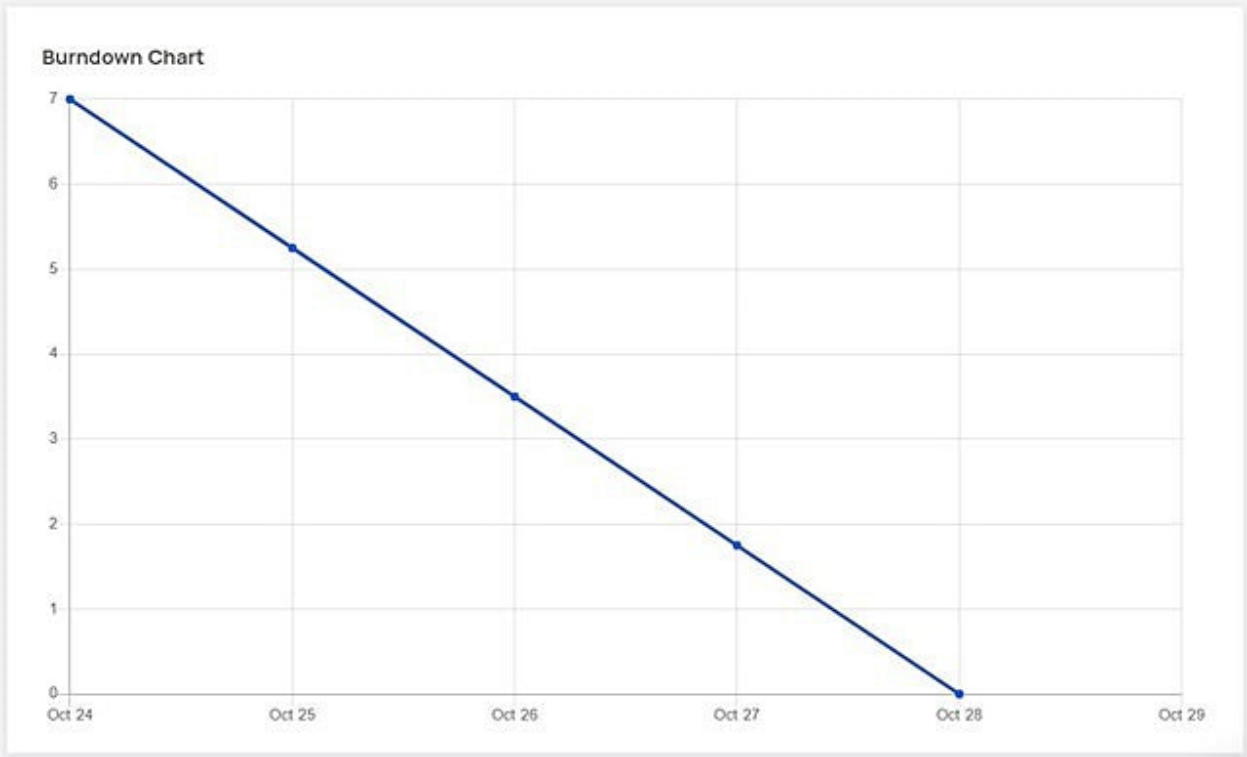
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	4
Sprint1		USN-2	As a user, I can register for the application through E-mail	1	Medium	4
Sprint1	Confirmation	USN-3	As a user, I will receive confirmation email once I have registered for the application	2	Medium	4
Sprint1	Login	USN-4	As a user, I can log into the application by entering email	2	High	4

			& password			
Sprint2	Dashboard	USN-5	As a user, I can view the products which are available	4	High	4
Sprint2	Add items to cart	USN-6	As a user, I can add the products I wish to buy to the carts.	5	Medium	4
Sprint3	Stock Update	USN-7	As a user, I can add products which are not available in the dashboard to the stock list.	5	Medium	4
Sprint4	Request to Customer Care	USN-8	As a user, I can contact the Customer Care Executive and	5	Low	4
Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Story Points**	**Priority**	**Team Members**
			request any services I want from the customer care.			

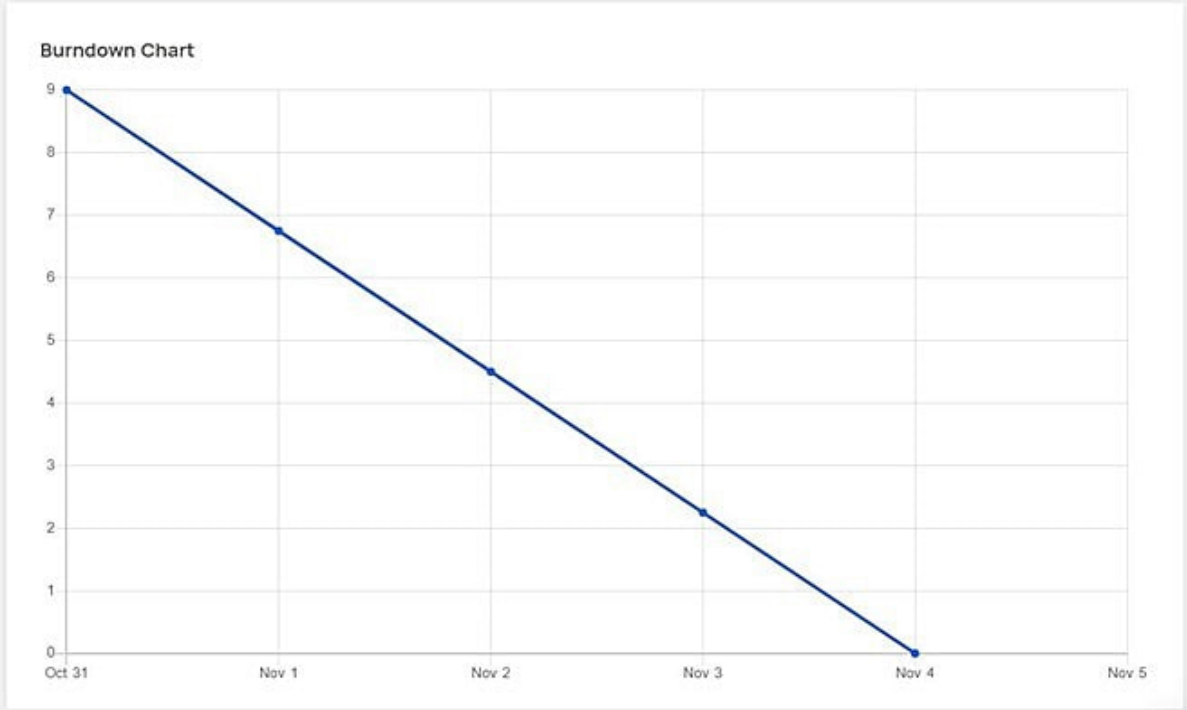
Sprint4	Contact Administrator	USN-9	I can be able to report any difficulties I experience as a report	5	Medium	4

6.2.2 Project Tracker, Velocity & Burndown Chart

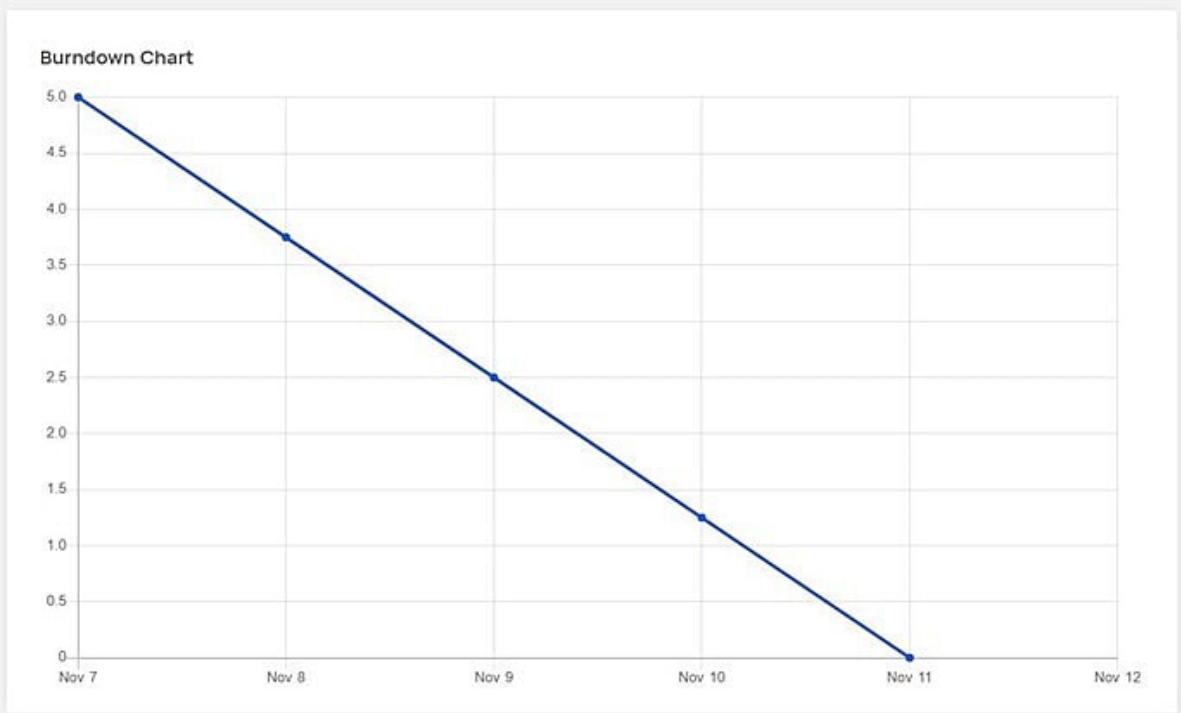
The below shown respective diagrams are the burndown charts of the sprint planning and the execution graph of the sprint.



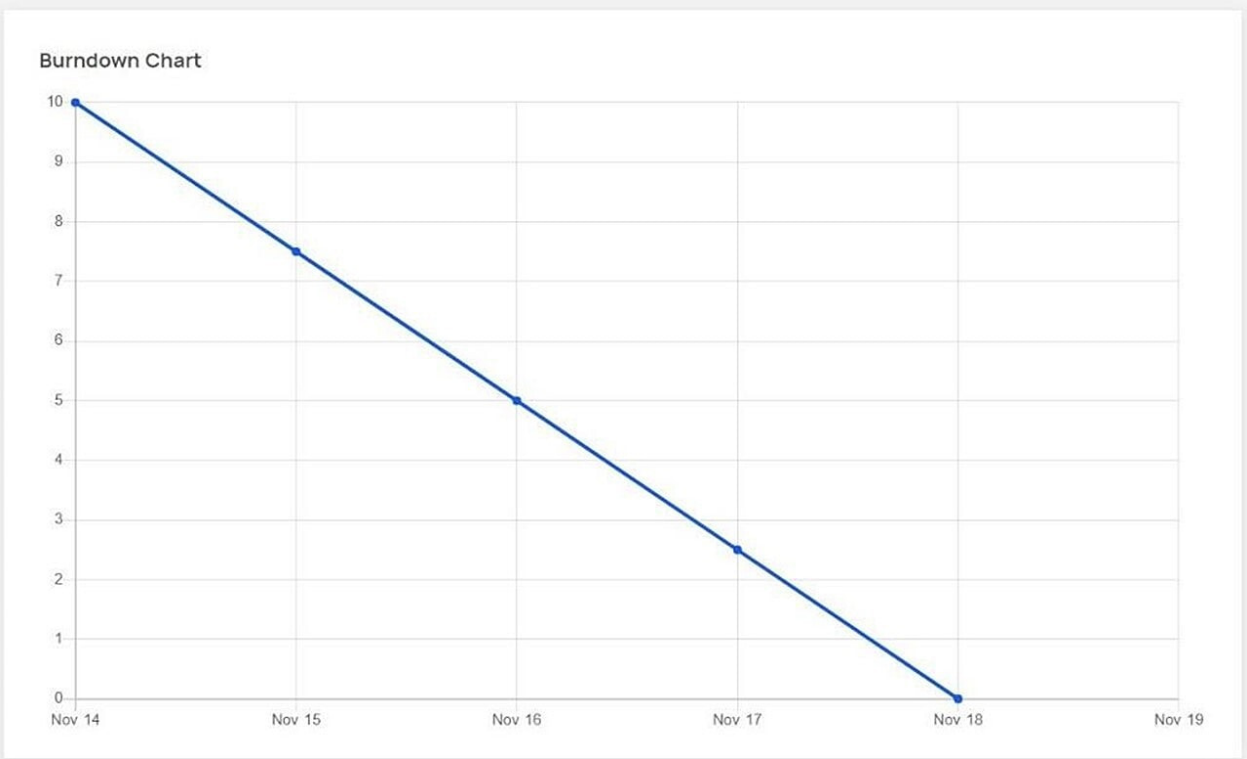
Sprint 1(registration, confirmation, login)



Sprint 2(dashboard, add items to cart)



Sprint 3(stock update)

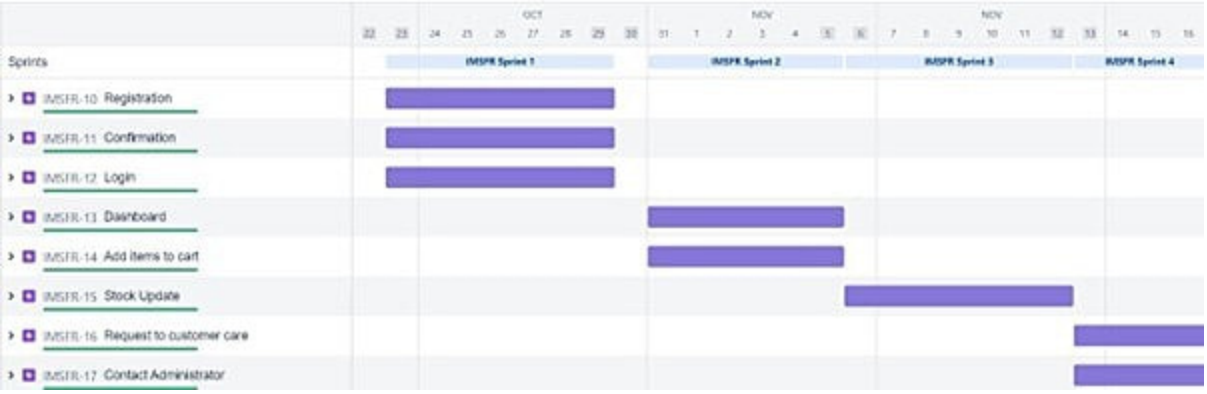


Sprint 4(request to customer care, contact administrator)

6.3 Reports from JIRA:

The planning of the sprint individually and the work done and the time taken to do that work is mentioned in the below picture.

The JIRA reports are mentioned below,



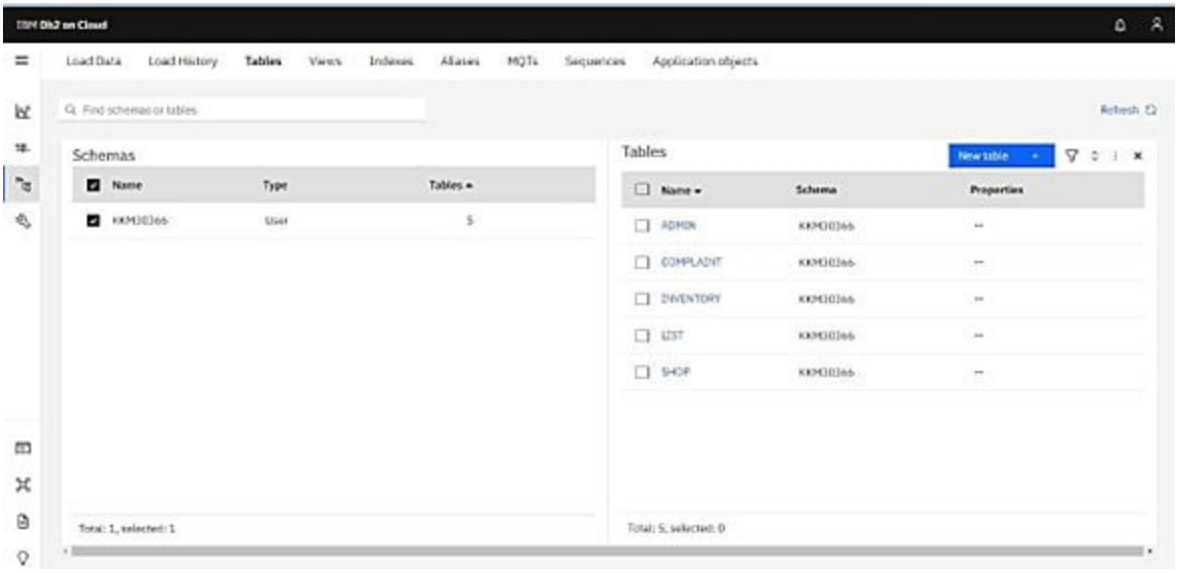
Reports for the

CHAPTER-7

CODING & SOLUTIONING

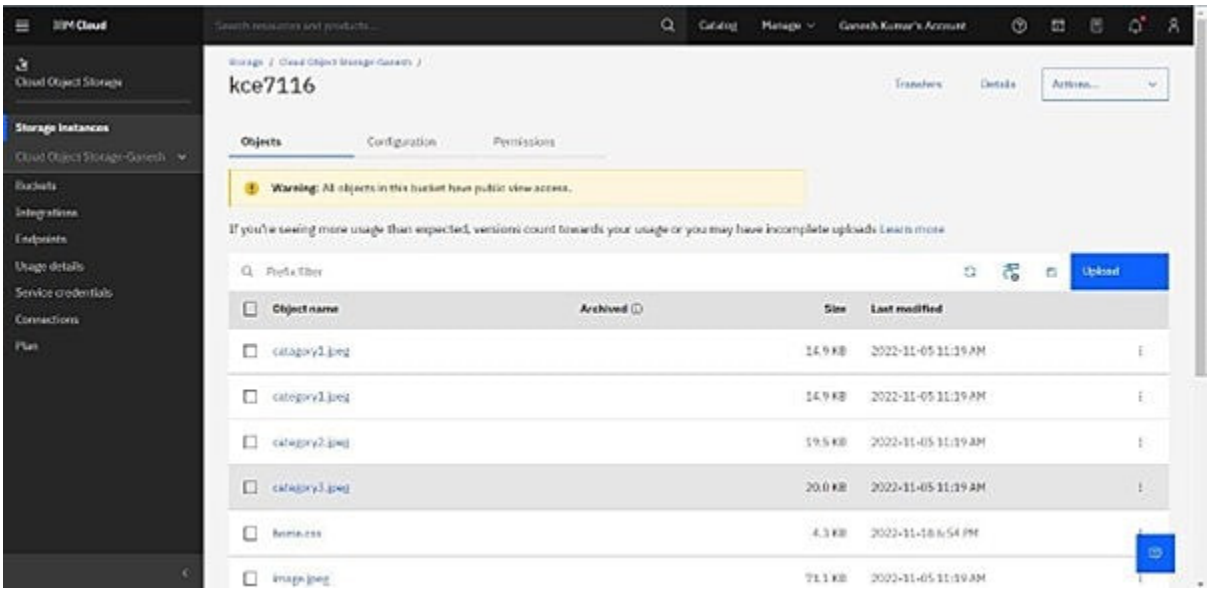
7.1 FEATURES 1:

In inventory management the mandatory need is the database so the user needs to store the info of the shop. The inventory management stores the data of the inventory in a tabled format in IBM DB2 provided by the IBM itself, which stores the information .



IBM db2 storage

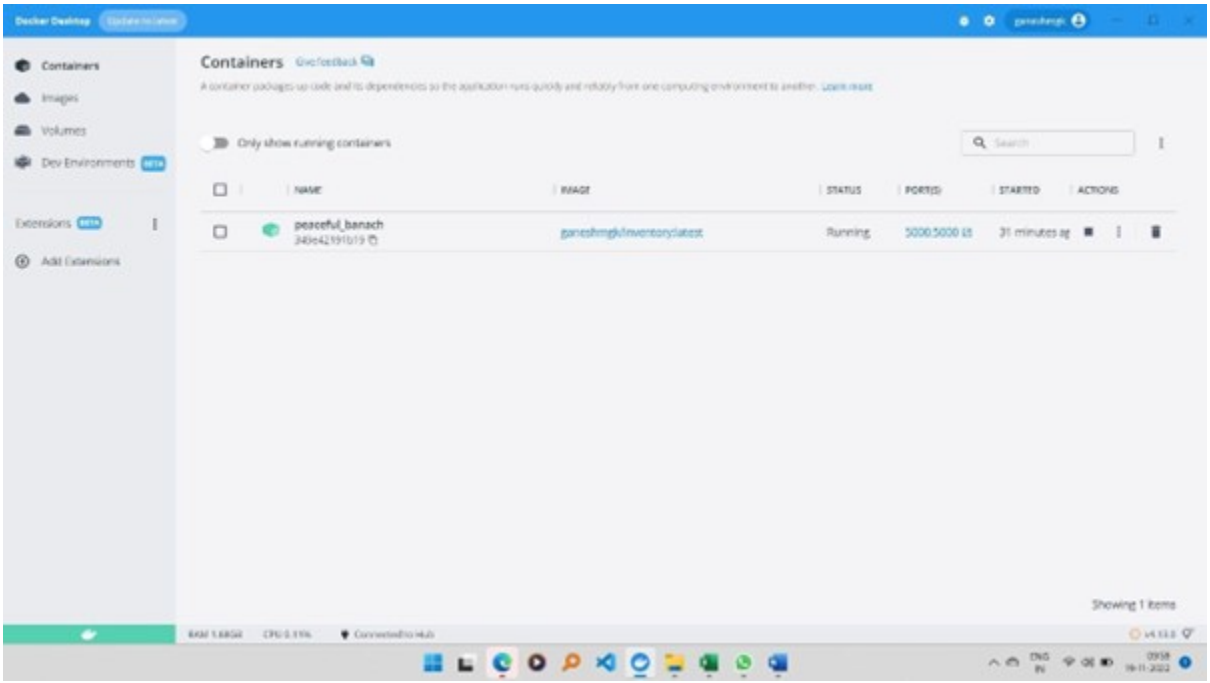
And for object storage we have used IBM Cloud Object Storage which stores images and cascading style sheets(css) files for the system.



IBM object storage

7.2 FEATURES 2:

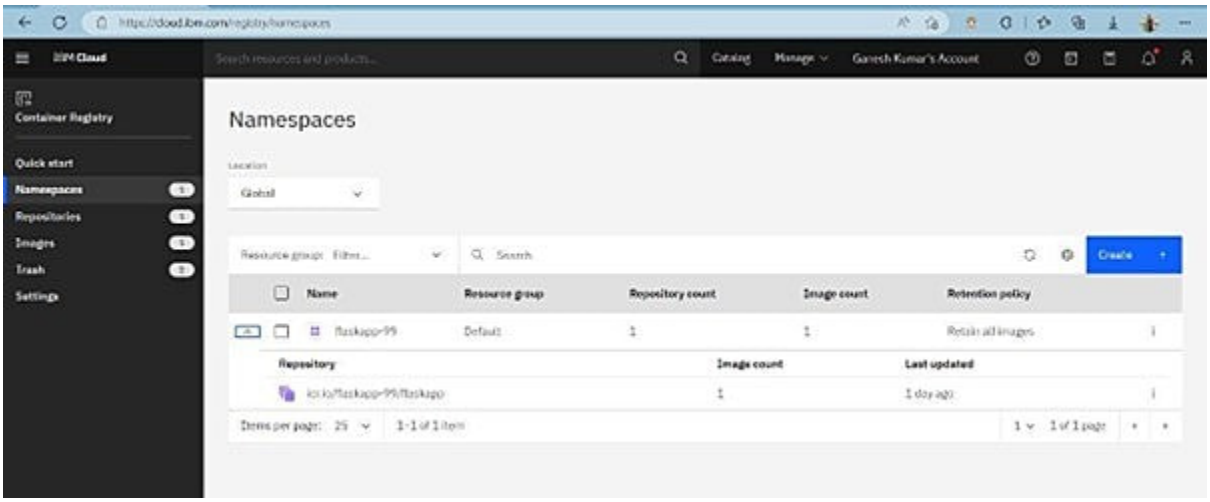
For containerize the application, we have used Docker Application to create and push Images to the docker hub.



Docker Application which contains the

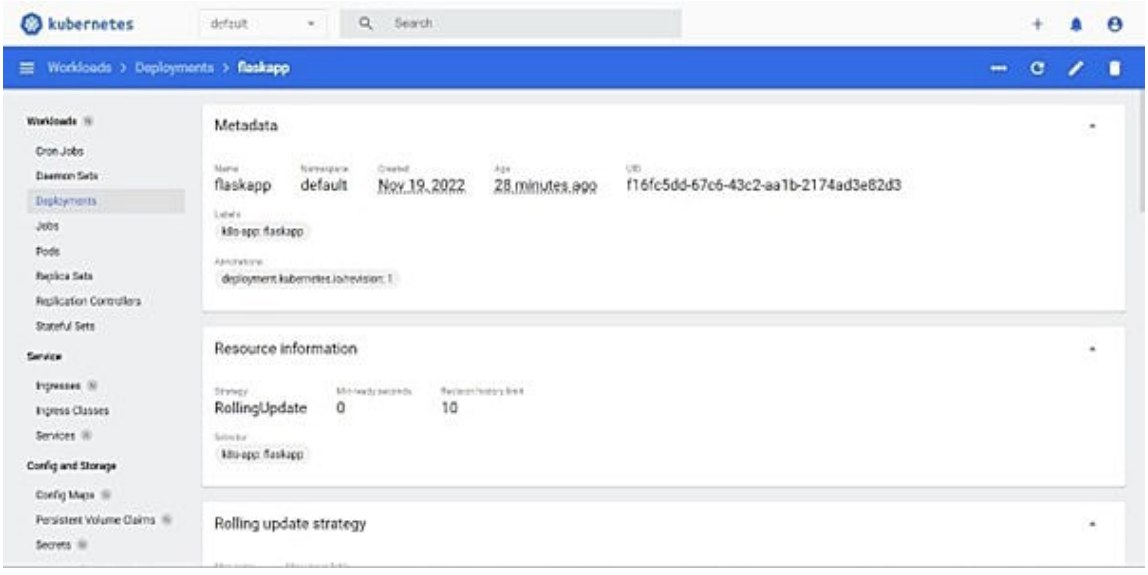
application as an image and create container with it.

From the docker hub, pull the image and store it in the IBM Container Registry Namespace through CLI.



IBM registry contains the image in its Namespace.

After uploading the image to the registry, with Kubernetes create new resource from the image that has been uploaded to the IBM namespace.



Kubernetes in the docker

After create the resource, the app will be deployed with Kubernetes and provide Internal Endpoints TCP number which combined with the public IP address provided with the namespace.(For the database usage, we have mentioned in the Feature 1)

CHAPTER-8 TESTING

8.1 TEST CASES:

Purpose of Document

The purpose of this document is to briefly explain the test cover age and open issues of the Corporate Employee Attrition Analytics project at the time of the release to User Acceptance Testing (UAT).

Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity1	Severity2	Severity3	Severity4	Subtotal
By Design	12	4	2	7	25
Duplicate	0	0	0	2	2
External	2	3	0	1	6
Fixed	11	5	4	15	35
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	0	0	1	1
Totals	35	12	8	27	72

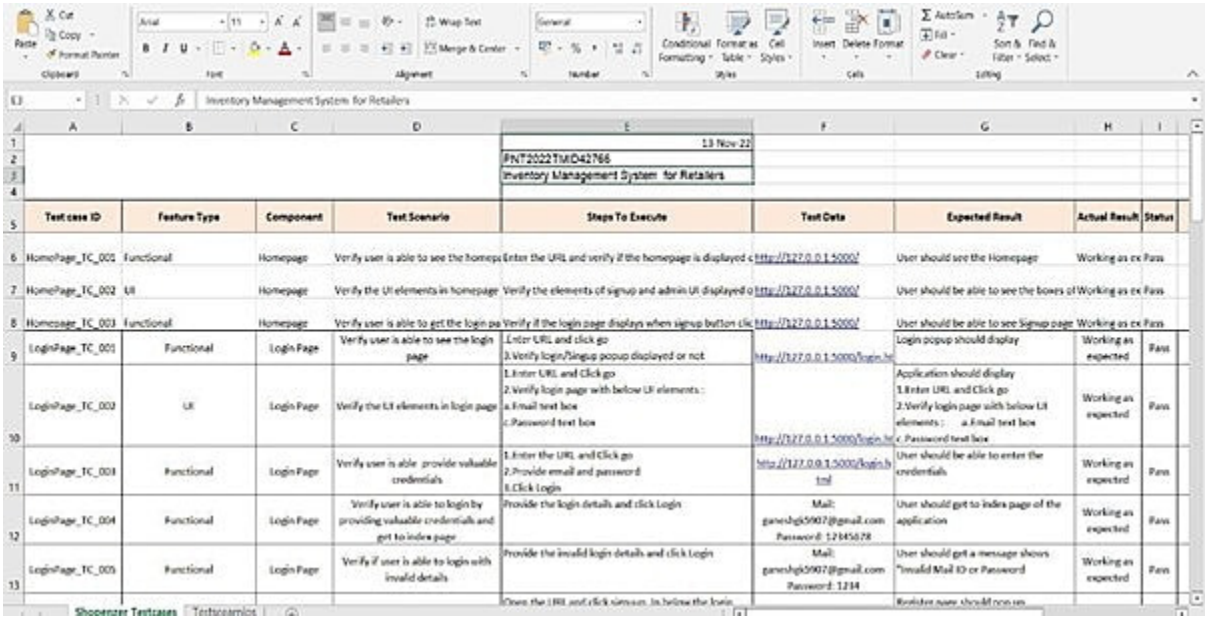
Test Case Analysis

This report shows the number of test cases that have passed, failed and untested

Section	Total Cases	Not Tested	Fail	Pass
Home page	4	0	0	4
Register Page	5	0	1	5
Login Page	5	0	0	5
Admin Login Page	4	0	0	4
Index Page	3	0	0	3
Products list page	5	0	0	3
Update Product Page	4	0	0	4
Contact Page	3	0	0	3
Complaint Page	3	0	0	3
Admin Index page	2	0	0	2
Admin Product Page	3	0	0	3
Logout from User Page	2	0	0	2
Logout from Admin page	2	0	0	2

8.2 USER ACCEPTANCE TESTING:

This is table that explaining the user's opinion about the functions of the components of the inventory management system and the detailed report is uploaded in the git hub repositories.



					13 Nov 22				
					PNT2022TMID42766				
					Inventory Management System for Retailers				
Test case ID	Feature Type	Component	Test Scenario	Steps To Execute	Test Data	Expected Result	Actual Result	Status	
HomePage_TC_001	Functional	Homepage	Verify user is able to see the homepa	Enter the URL and verify if the homepage is displayed c	http://127.0.0.1:5000/	User should see the Homepage	Working as ex	Pass	
HomePage_TC_002	UI	Homepage	Verify the UI elements in homepage	Verify the elements of signup and admin UI displayed o	http://127.0.0.1:5000/	User should be able to see the boxes of	Working as ex	Pass	
Homepage_TC_003	Functional	Homepage	Verify user is able to get the login pa	Verify if the login page displays when signup button clic	http://127.0.0.1:5000/	User should be able to see Signup page	Working as ex	Pass	
LoginPage_TC_001	Functional	Login Page	Verify user is able to see the login page	1.Enter URL and click go 3.Verify login/Singup popup displayed or not	http://127.0.0.1:5000/login.ht	Login popup should display	Working as expected	Pass	
LoginPage_TC_002	UI	Login Page	Verify the UI elements in login page	1.Enter URL and Click go 2.Verify login page with below UI elements : a.Email text box c.Password text box	http://127.0.0.1:5000/login.ht	Application should display 1.Enter URL and Click go 2.Verify login page with below UI elements : a.Email text box c. Password text box	Working as expected	Pass	
LoginPage_TC_003	Functional	Login Page	Verify user is able provide valuable credentials	1.Enter the URL and Click go 2.Provide email and password 3.Click Login	http://127.0.0.1:5000/login.html	User should be able to enter the credentials	Working as expected	Pass	
LoginPage_TC_004	Functional	Login Page	Verify user is able to login by providing valuable credentials and get to index page	Provide the login details and click Login	Mail: ganeshgk5907@gmail.com Password: 12345678	User should get to index page of the application	Working as expected	Pass	
LoginPage_TC_005	Functional	Login Page	Verify if user is able to login with invalid details	Provide the invalid login details and click Login	Mail: ganeshgk5907@gmail.com Password: 1234	User should get a message shows "Invalid Mail ID or Password	Working as expected	Pass	

User acceptance test

CHAPTER 9

RESULTS

9.1 PERFORMANCE METRICS:

Performance metrics

The above is the performance metrics evaluation of our application calculated by the Gatling.

The is the report generated by the Gatling – Professional Load Testing Tool.

CHAPTER-10

ADVANTAGES & DISADVANTAGES

10.1 Advantages

1. It helps to maintain the right amount of stocks
2. It leads to a more organized warehouse
3. It saves time and money
4. Improves efficiency and productivity
5. A well-structured inventory management system leads to improved customer retention
6. Schedule maintenance
7. Reduction in holding costs
8. Flexibility - the ability to easily adjust production levels, rawmaterial purchases, and transport capacity

10.2 Disadvantages

1. Bureaucracy - The term bureaucracy refers to a complex organization that has multi layered systems and processes.
2. Impersonal touch
3. Increased space is need to hold the inventory
4. High implementation costs
5. Complexity - Longer / less reliable lead times from global suppliers.

CHAPTER – 11

CONCLUSION

Inventory management has to do with keeping accurate records of goods that are ready for shipment. This often means having enough stock of goods to the inventory totals as well as subtracting the most recent shipments of finished goods to buyers. When the company has a return policy in place, there is usually a sub-category contained in the finished goods inventory to account for any returned goods that are reclassified or second grade quality. Accurately maintaining figures on the finished goods inventory makes it possible to quickly convey information to sales personnel as to what is available and ready for shipment at any given time by buyer. Inventory management is important for keeping costs down, while meeting regulation. Supply and demand is a delicate balance, and inventory management hopes to ensure that the balance is undisturbed. Highly trained Inventory management and high-quality software will help make Inventory management a success. The ROI of Inventory management will be seen in the forms of increased revenue and profits, positive employee atmosphere, and on overall increase of customer satisfaction.

CHAPTER – 12

FUTURE SCOPE

The scope of an inventory system can cover many needs, including valuing the inventory, measuring the change in inventory and planning for future inventory levels. The value of the inventory at the end of each period provides a basis for financial reporting on the balance sheet, store and van stock management is being revolutionised with the use of inventory and supply-chain management apps on mobile devices. However, not all app-based inventory management systems are the same. Leading-edge app-based Inventory Management systems: make use of the best technologies to manage the process, Inventory management is vital for retailers because the practice helps them increase profits. They are more likely to have enough inventory to capture every possible sale while avoiding overstock and minimizing expenses. From a strategic point of view, retail inventory management increases efficiency.

1. Invest in an inventory management system.
2. Set up stock alerts.
3. Select suppliers strategically.
4. Implement SKU management practices.
5. Optimize your order size.
6. Consider drop shipping.

CHAPTER – 13

APPENDIX

The system analyst interviewed various officials like manager of the store, dealing hand, and users to discuss present way of working and possible improvement in the system. During the discussions it was found that first user has to place a request on pre-printed form to store manager through the concerned head of the section. Depending on the availability, item will be issued to user otherwise NOC is issued. Quantity of item issued is subtracted from the balance and when item is received from the supplier quantity is added in the balance. A special ledger is maintained to records issue and receipt entries of items. When the particular item reaches below reorder level procurement procedure starts to replenish the item. ABC analysis of inventory items is performed on regular basis which is time consuming process. The management also finds fast moving and slow items from time to time so that sufficient stock of fastmoving items can be maintained and slow moving items can be reduced from the inventory. To facilitate inventory management, ABC analysis classifies the inventory items into following three classes based on the consumption value:

A. Class A: These items constitute the most important class of inventories so far as the proportion in the total value of inventory.
B. Class B: These items constitute an intermediate position, which constitute approximately 20% of the total items, accounts for approximately 20% of the total material consumption value.
C. Class C: It consists remaining 70% items, accounting only 10% of the monetary value of total material usage. Quite relaxed inventory procedures are used.

Expected outputs from the system

A. Current balance of item in stock.
B. List of items below reorder level.
C. Item wise cost of inventory.
D. List of spoiled items.
E. List of items received and issued within a specific period.
F. Ledger preparations as required by management
G. ABC analysis of inventory items.
H. Placement of orders for replenishment of items

Input requirements

A. Details of Inventory items
B. Details of items received
C. Details of items issued
D. Details of issue form
E. Details of items ordered for procurement
F. Details of potential vendors

Methods and procedure

Computing Current Balance: *Current balance = Current Balance + Receipt - Issue*

Computing Reorder Level: *Reorder Level = Daily Demand * Procurement Time*
Daily Demand = Yearly Demand / *No. of Working Days*
Procurement time is generally constant

Source code:

home.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>SHOPZY</title>
    <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.2.0/css/all.min.css" integrity="sha512-
xh6O/CkQoPOWDdYTDqeRdPCVd1SpvCA9XXcUnZS2FmJNp1coAFzvtCN9BmamE+4aHK8yyUHUSC
cJHgXloTyT2A==" crossorigin="anonymous" referrerpolicy="no-referrer" />
    <!-- CSS only -->
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.
css" rel="stylesheet" integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi"
crossorigin="anonymous">
    <!-- JavaScript Bundle with Popper -->
<link href="homecss.css" rel="stylesheet"/>
</head>
<body>
    <nav class="navbar navbar-expand-lg bg-dark">
        <div class="container-fluid">
          <a class="navbar-brand fw-bold text-info"
href="home.html"><h5>SHOPZY</h5></a>
          <button class="navbar-toggler" type="button" data-bs-
toggle="collapse" data-bs-target="#navbarSupportedContent" aria-
controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle
navigation">
            <span class="navbar-toggler-icon"></span>
          </button>
```

```
          <div class="collapse navbar-collapse"
id="navbarSupportedContent">
            <ul class="navbar-nav me-auto mb-2 mb-lg-0">
              <li class="nav-item">
                <a class="nav-link active" aria-current="page"
href="home.html"></a>
            </ul>
            <a href="login.html"><button type="button" class="btn btn-info
btn-warning" data-bs-toggle="modal" data-bs-target="#exampleModal" ><i
class="fa-solid fa-arrow-right-to-bracket"></i> SIGNIN</button></a>
            <a href="adminlogin.html"><button type="button" class="btn
btn-info" data-bs-toggle="modal" data-bs-target="#exampleModal"
style="background-color:white; "><i class="fa-solid fa-user-plus"></i>
ADMIN</button></a>
          </div>
        </div>
      </nav>
      <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle
.min.js" integrity="sha384-
OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJuaOe923+mo//f6V8Qbsw3"
crossorigin="anonymous"></script>
      <!-- My Java script -->
      <script src="index.js" ></script>
</body>
</html>
```

index.html:

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <title>Store</title>
        <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.2.0/css/all.min.css" integrity="sha512-
```

```
xh6O/CkQoPOWDdYTDqeRdPCVd1SpvCA9XXcUnZS2FmJNp1coAFzvtCN9BmamE+4aHK8yyUHUSC
cJHgXloTyT2A==" crossorigin="anonymous" referrerpolicy="no-referrer" />
        <link rel="stylesheet" href="home.css">
        <link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link
href="https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,700;1,10
0;1,900&display=swap" rel="stylesheet">
<link href="{{url_for('static', filename='home.css')}}" rel="stylesheet">
    </head>
    <body>
        <div class="header">
        <div class="container">
            <div class="navbar">
                <div class="logo">
                    <a href="cart.html"><img src="https://s3.jp-tok.cloud-
object-storage.appdomain.cloud/kce7116/logo.jpg"width="125px"></a>
            </div>
            <nav>
                <ul>
                    <li><a href=""><h3><i class="fa-sharp fa-solid fa-
house"></i> Home</h3></a></li>
                    <li><a href="products.html">Products</a></li>
                    <li class="nav-item">
                        <a class="nav-link" href="/logout">Logout</a>
                </ul>
            </nav>
            <img src="https://s3.jp-tok.cloud-object-
storage.appdomain.cloud/kce7116/inventory.jpeg" width="30px"
height="30px">
        </div>
        <div class="row">
            <div class="col2">
                <h1>
                    hello there!<br>welcome to Shopzy.
```

```
                </h1>
                <p>
                A place for ALL!. A trusted provider of goods and
services.
                </p>
                <a href=""class="btn">explore now &#8594</a>
            </div>
            <div class="col2 img">
                <img src="https://s3.jp-tok.cloud-object-
storage.appdomain.cloud/kce7116/inventory_management.jpg">
            </div>
        </div>
    </div>
<!-- -----featured catagories-------- -->
<div class="categories">
    <div class="smallcontainer">
    <div class="row">
        <div class="col3"> <img src="https://s3.jp-tok.cloud-object-
storage.appdomain.cloud/kce7116/catagory1.jpeg"></div>
        <div class="col3" id="col-i"> <img src="https://s3.jp-tok.cloud-
object-storage.appdomain.cloud/kce7116/category2.jpeg"></div>
        <div class="col3"> <img src="https://s3.jp-tok.cloud-object-
storage.appdomain.cloud/kce7116/category3.jpeg"></div></div></div></div>
<!-- -----featured catagories-------- -->
<div class="smallcontainer">
 <h2 class="title">featured products</h2>
    <div class="row">
        <div class="pro">
        <div class="col4">
            <img  src="https://s3.jp-tok.cloud-object-
storage.appdomain.cloud/kce7116/product1.jpeg" >
            <h4>Red Nail Polish</h4>
            <div class="rating">
            <i class="fa fa-star"></i>
            <i class="fa fa-star"></i>
            <i class="fa fa-star"></i>
```

```
            <i class="fa fa-star"></i>
            <i class="fa fa-star-half-stroke"></i></div>
            <p>$30.00</p></div>
        </div> <div class="col4"><div class="pro">
            <img src="https://s3.jp-tok.cloud-object-
storage.appdomain.cloud/kce7116/product2.jpeg" >
            <h4>Makeup Kit</h4>
            <div class="rating">
            <i class="fa fa-star"></i>
            <i class="fa fa-star"></i>
            <i class="fa fa-star"></i>
           <i class="fa fa-star-half-stroke"></i>
           <i class="fa-regular fa-star"></i></div>
            <p>$35.00</p></div>
        </div> <div class="col4"><div class="pro">
            <img src="https://s3.jp-tok.cloud-object-
storage.appdomain.cloud/kce7116/product3.jpeg" >
            <h4>Hair Straightning Cream</h4>
            <div class="rating">
            <i class="fa fa-star"></i>
            <i class="fa fa-star"></i>
            <i class="fa fa-star"></i>
            <i class="fa fa-star"></i>
            <i class="fa-regular fa-star"></i></div>
            <p>$40.00</p> </div></div> </div></div>
            <footer class="footer-distributed">
                <div class="footer-right">
                    <!-- <a href="www.google.com"><i class="fa fa-
facebook"></i></a>
                    <a href="#"><i class="fa fa-twitter"></i></a>
                    <a href="#"><i class="fa fa-linkedin"></i></a>
                    <a href="#"><i class="fa fa-github"></i></a> -->
```

```
                </div>

                <div class="footer-left">

                    <p class="footer-links">
                        <a class="link-1" href="index.html">Home</a>

                        <a href="contact.html">Contact Us</a>

                        <a href="complaint.html">Raise a Complaint</a>
                    </p>

                    <p>Shopzy &copy; 2022 All Rights are Reserved</p>
                </div>

            </footer>
        </body></html>
```

app.py:

```
import datetime
from datetime import datetime
from pickletools import read_unicodestring1
from turtle import st, update
import ibm_db
from flask import Flask, flash, render_template, request, session, url_for, redirect
# from flask_mail import Mail, Message
# from flask import *


from mydb import connect
```

```
app = Flask(__name__)
app.secret_key = 'your secret key'
mail = Mail(app)

app.config['MAIL_SERVER']='smtp.gmail.com'
app.config['MAIL_PORT'] = 465
app.config['MAIL_USERNAME'] = 'team06inventory@gmail.com'
app.config['MAIL_PASSWORD'] = 'pjqnmjtrdlqkjqfj'
app.config['MAIL_USE_TLS'] = False
app.config['MAIL_USE_SSL'] = True
mail = Mail(app)

@app.route("/")
def homepage():
    return render_template("home.html")

@app.route("/login.html")
def loginpage():
    return render_template("login.html")

@app.route("/adminlogin.html", methods = ['GET','POST'])
def adminlogin():
    return render_template("adminlogin.html")

@app.route('/admindata', methods=['POST', 'GET'])
def admin():
    # userdatabase = []
    if request.method == 'POST':
        email = request.form.get('adminemail')
        password = request.form.get('adminpassword')
```

```
        sql = "SELECT * FROM ADMIN WHERE EMAIL = ? AND
PASSWORD = ?"
        stmt = ibm_db.prepare(connect.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)
        print(account)
        if account:
            return render_template("admin/index.html")
    return render_template("adminlogin.html")

@app.route("/logindata", methods=['GET','POST'])
def login():
    if request.method == 'POST':
        email = request.form.get('email')
        password = request.form.get('password')

        sql = "SELECT * FROM SHOP WHERE EMAIL = ? AND
PASSWORD = ?"
        stmt = ibm_db.prepare(connect.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)
        print(account)
        if account:
```

```
            return render_template("index.html")
        else:
            return ("Invalid username or password")
    return render_template("login.html")

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

@app.route("/registerdata", methods=['GET','POST'])
def registernew():
    if request.method == 'POST':
        name = request.form['name']
        email = request.form['email']
        password = request.form['pwd']
        mobile = request.form['ph']

        sql = "SELECT * FROM SHOP WHERE EMAIL = ?"
        stmt = ibm_db.prepare(connect.conn,sql)
        ibm_db.bind_param(stmt,1,email)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
            msg = "Already existed account! Kindly Login"
            return render_template("login.html")
        else:
            sql = "INSERT INTO SHOP
(NAME,EMAIL,PASSWORD,MOBILENUMBER)
VALUES('{0}','{1}','{2}','{3}')"
```

```
        res =
ibm_db.exec_immediate(connect.conn,sql.format(name,email,password,
mobile))
        mesg = Message(
            'Hello',
            sender ='team06inventory@gmail.com',
            recipients = [email]
           )
      mesg.body = 'Welcome to Shopzy. Thank you for registering with
us.\nHappy Shop(zy)ing!!!.\nLogin id:\n
email:'+email+'\nPassword:'+password
      mail.send(mesg)
      msg = "Your account has been registered successfully!l"
      if res:
         return render_template("login.html",msg=msg)

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

@app.route("/products.html")
def dashboard():
   return render_template("products.html")

@app.route("/addproducts.html")
def addprod():
   return render_template("addproducts.html")


@app.route("/addproducts.html",methods = ['POST', 'GET'])
```

```
def addproduct():
    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)
        name =  request.form['name']

        sql = "SELECT * FROM LIST WHERE PRODUCTNAME =?"
        prep_stmt = ibm_db.prepare(connect.conn, sql)
        ibm_db.bind_param(prep_stmt,1,pname)
        ibm_db.execute(prep_stmt)
        product = ibm_db.fetch_assoc(prep_stmt)
        if product:

          if product['PRODUCTNAME']==pname:

            return render_template('addproducts.html', msg="Product already
added! Add a new product.")
        #   else:
        #     sql ="INSERT INTO LIST
(PRODUCTNAME,QUANTITY,DATE,HOLDERNAME) VALUES
(?,?,?,?);"
        #     prep_stmt = ibm_db.prepare(connect.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, str(the_time))
        #     ibm_db.bind_param(prep_stmt, 4, name)
        #     ibm_db.execute(prep_stmt)
        #     return render_template('addproducts.html', msg="Product
```

```
added")
        else:
            sql ="INSERT INTO LIST
(PRODUCTNAME,QUANTITY,DATE,HOLDERNAME) VALUES
(?,?,?,?);"
            prep_stmt = ibm_db.prepare(connect.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, str(the_time))
            ibm_db.bind_param(prep_stmt, 4, name)
            ibm_db.execute(prep_stmt)
            return render_template('addproducts.html', msg="Product
added")
    return render_template("addproducts.html")



@app.route('/contact.html')
def contact():
    return render_template("contact.html")

@app.route('/complaint.html')
def compalint():
    return render_template("complaint.html")

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

```
        complaint =  request.form['complaint']
        sql = "INSERT INTO COMPLAINT
(NAME,MAIL,COMPLAINT) VALUES (?,?,?);"
        prep_stmt = ibm_db.prepare(connect.conn, sql)
        ibm_db.bind_param(prep_stmt, 1, name)
        ibm_db.bind_param(prep_stmt, 2, mail)
        ibm_db.bind_param(prep_stmt, 3, complaint)
        ibm_db.execute(prep_stmt)
        flash("Complaint Sent", "Thank you for contacting us.")
        return render_template('complaint.html', msg = "Complaint Sent.
Thank you for contacting us.")
    return render_template("complaint.html")


# For Admin
@app.route("/updateproducts.html")
def updateprod():
    return render_template("admin/updateproducts.html")

@app.route("/updateproducts",methods = ['POST', 'GET'])
def updateproducts():
    if request.method == 'POST':
        pid = request.form['pid']
        pname = request.form['pname']
        quantity = request.form['quantity']
        price =  request.form['price']

        sql = "SELECT * FROM INVENTORY WHERE NAME =?"
        prep_stmt = ibm_db.prepare(connect.conn, sql)
        ibm_db.bind_param(prep_stmt,1,pname)
```

```
        ibm_db.execute(prep_stmt)
        product = ibm_db.fetch_assoc(prep_stmt)
        if product:

          if product['NAME']==pname:

            return render_template('admin/updateproducts.html',
msg="Product already existed! Add a new product.")
        else:
            sql ="INSERT INTO INVENTORY
(ID,NAME,QUANTITY,PRICE) VALUES (?,?,?,?);"
            prep_stmt = ibm_db.prepare(connect.conn, sql)
            ibm_db.bind_param(prep_stmt,1,pid)
            ibm_db.bind_param(prep_stmt,2,pname)
            ibm_db.bind_param(prep_stmt,3,quantity)
            ibm_db.bind_param(prep_stmt,4,price)
            ibm_db.execute(prep_stmt)
            return render_template('admin/updateproducts.html',
msg="Product added")
    return render_template("admin/updateproducts.html")


@app.route('/logout')
def logout():
    session.pop('loggedin', None)
    session.pop('id', None)
    session.pop('email', None)
    session.pop('name', None)
    return render_template("home.html")
```

```
if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5000, debug=True)
```

GIT HUB:

The repositeries of the git hub link is given below ,you can refer this below link to know more info:

https://github.com/IBM-EPBL/IBM-Project-13614-1659524091

project demo link:

https://drive.google.com/file/d/1klTtixg6i_goaBSRfNjYPvYGvY4MS8rD/view?usp=share_link