

IBM – NALAIYA THIRAN PROJECT

INVENTORY MANAGEMENT SYSTEM

INDUSTRY MENTOR : VASUDEVA HANUSH

FACULTY MENTOR : KOUSIKA N

TEAM ID : PNT2022TMID02851

TEAM LEAD : YATHEESWAR KARTHICK R

TEAM MEMBER : VAISHNAV M A

TEAM MEMBER : VIGNESH M

TEAM MEMBER : YASH P A

TABLE OF CONTENT

CHAPTER	CONTENTS	PAGE NO
1	INTRODUCTION 1.1 PROJECT OVERVIEW 1.2 PURPOSE	04
2	LITERATURE SURVEY 2.1 EXISTING PROBLEM 2.2 REFERENCES 2.3 PROBLEM STATEMENT DEFINITION	05
3	IDEATION & PROPOSED SOLUTION 3.1 EMPATHY MAP CANVAS 3.2 IDEATION & BRAINSTROMING 3.3 PROPOSED SOLUTION 3.4 PROBLEM SOLUTION FIT	08
4	REQUIREMENT ANALYSIS 4.1 FUNCTIONAL REQUIREMENT 4.2 NON-FUNCTIONAL REQUIREMENTS	11
5	PROJECT DESIGN 5.1 DATA FLOW DIAGRAMS 5.2 SOLUTION & TECHNICAL ARCHITECTURE 5.3 USER STORIES	14
6	PROJECT PLANNING & SCHEDULING 6.1 SPRINT PLANNING & ESTIMATION 6.2 SPRINT DELIVERY SCHEDULE 6.3 REPORTS FROM JIRA	18

7	CODING & SOLUTIONING 7.1 FEATURE 1 7.2 FEATURE 2 7.3 DATABASE SCHEMA	23
8	TESTING 8.1 TEST CASES 8.2 USER ACCEPTANCE TESTING	26
9	RESULTS 9.1 PERFORMANCE METRICS	28
10	ADVANTAGES & DISADVANTAGES	31
11	CONCLUSION	33
12	FUTURE SCOPE	34
13	APPENDIX SOURCE CODE GITHUB & PROJECT DEMO LINK	34

1.INTRODUCTION:

Inventory management information system is high performance software, which speed up the business operation of the organization . Every organization , which deals with the raw materials, put its great effort in the efficient utilization of its raw, material according to its need and requirement . The organization has to perform number of tasks and operations in order to run its business in manual system .

For example From NaavebUROM

- Estimation of new raw material required.
- Preparation of purchase order.

Preparation of inward sale invoice This Software “Inventory Management System” , is used for recording the information about the day to day transaction of stock of an organization. It stores purchase information of the products with credit/debit information form the supplier. Similarly, it stores sales information with credit/debit about the customer. If a product is purchased, then the related information is stored in stocks , that is , stocks are up to date. Another part I it prepare sales report after product it sold. in the sales information, the information about who sold the product is also kept, so there is no problem for misunderstandings in future.

1.1. PROJECT OVERVIEW:

Inventory management system is an application which is helpful for business operate. Inventory management is a challenging problem area in supply chain management. Companies need to have inventories in warehouses in order to fulfil customer demand, meanwhile these inventories have holding costs and this is frozen fund that can be lost. Therefore, the task of inventory management is to find the quantity of inventories that will fulfil the demand, avoiding overstocks. This paper presents a case study for the assembling company on inventory management. It is proposed to use inventory management in order to decrease stock levels and to apply an agent system for automation of inventory management processes. Inventory management system (IMS) use for a departmental store.

1.2. PURPOSE:

A case study at 'Guckenheimer' (an on-site corporate restaurant management and catering company) cited issues regarding a basic resources requirement list that has to be maintained manually by the staff. To keep track of their inventory levels they have to calculate a list of the groceries utilized during a course of time, calculate and analyze the requirements for the future, and place their next order to the vendors if needed. This process takes up a lot of time and human effort, and is also prone to human error. This poses a problem of a situation that the staff at 'Guckenheimer,' as well as many other restaurants faces. It takes up a lot of time to manually keep track of sales and place correct orders to vendors, wasting useful labor in trivial works. A product which would assist in tackling the above mentioned problems would prove to be fruitful to clients such as 'Guckenheimer' and similar enterprises as this product would help convert the unproductive time to something more useful, by removing the unnecessary error prone complications and efforts.

2.LITERATURE SURVEY:

2.1. EXISTING PROBLEM:

Products are considered as the business resources for the organization. This includes managing the product with appropriate way to review any time as per the requirement. Therefore it is important to have a computer based IMS which has the ability to generate reports, maintain the balance of the stock, details about the purchase and sales in the organization. Before developing this application we came up with 2Inventory Management System existing in the market, which helps to give the knowledge for the development of our project. These application software are only used by the large organization but so we came up with the application which can be used by the small company for the management of their stock in the production houses. After analyzing the other inventory management system we decided to include some of common and key features that should be included in every inventory management system. So we decided to include those things that help the small organization in away or other.

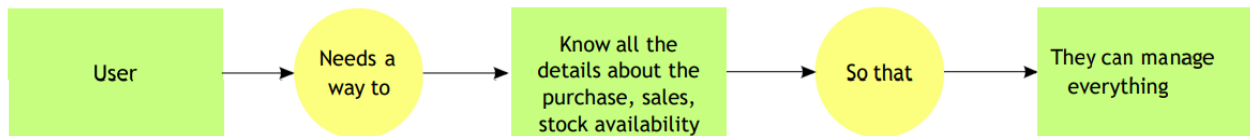
2.2. REFERENCES:

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

- [1] <https://www.camcode.com/asset-tags/what-is-an-inventory-management-system/>
- [2] Jimmy Wales, online encyclopedia WiKipedia ,
<http://www.wikipedia.org>
- [3] James Gosling. Java (Programming Language) ,
<http://www.java.com>
- [4] Names Allaire, Netbeans-Fully-featured Java IDE,
<http://www.netbeans.org>
- [5] James Gosling , Welcome to java world.com:
how-to feature and columns by Java expert; news;
Java applets; sample code ; tips ,
<http://www.javaworld.com>
- [6] Pressman, Roger S.
“Software Engineering A Practitioner” Approach
- [7] John Osborn , JavaBeans:
Developing Component Software in Java
- [8] Doug Lea Concurrent Programming in Java:
Design Principles and Pattern, Addison-Wesley , November,1996
- [9] Design Report, submitted 9th November 2012.
<https://skydrive.live.com/redir?resid=2CEDE9F7F5F99604!196&authkey=!AO5IghTCML6xAk8>
- [10] Testing Document, submitted 26th November 2012.
<https://skydrive.live.com/redir?resid=2CEDE9F7F5F99604!161&authkey=!AC8P0Lqe9amxSeM>

2.3. PROBLEM STATEMENT DEFINITION:

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



Who does the problem affect?	Retailers and Customers
Why is it important to use?	Greater Insights about stocks and Increased productivity
What are the Benefits?	Better Inventory Accuracy and avoiding Stockouts and Excess Stock
How is it better than the others?	Accessible by retailer and at any time and more Organized Warehouse
When to use?	To get rid of obsolete and out of date inventory items. To evaluate movement of specific items. When a company wants a dynamic and systematic system to record and keep their inventory data.

3. IDEATION & PROPOSED SOLUTION:

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

3.1. EMPATHY MAP CANVAS:

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



3.2. IDEATION & BRAINSTORMING:

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

Team Lead		Team Member 1		Team Member 2		Team Member 3	
<i>Managing Finances</i>	<i>Tracking Inventory</i>	<i>Avoiding Late Deliveries</i>	<i>Managing Time and Effort</i>	<i>Predicting Future Sales</i>	<i>Enhancing Customer Loyalty</i>	<i>inventory prioritize</i>	<i>Feedback and Rating</i>
<i>Flexibility</i>	<i>Scheduling</i>	<i>Information are secured</i>	<i>Enhanced user interface</i>	<i>Storing the information about customer</i>	<i>Storing the information about product</i>	<i>Increased information transparency</i>	<i>Checking availability of product</i>

3.3. PROPOSED SOLUTION:

PREPROCESSING PHASES:

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

Sales: This is Provide Sales information And Sales Page it is contain sales_id, Product_code , Product_name , Quantity, Revenue, Sold by etc.

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

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

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

Edit: Many Module contains Edit facilities Which has control of editing value from data base directly and insert new value etc.

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

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

3.4. PROBLEM SOLUTION FIT:

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

4.REQUIREMENT ANALYSIS:

4.1. FUNCTIONAL REQUIREMENTS:

- The user must have, at disposal, functions for managing the inventory efficiently.
- The functions for inventory management should allow the user to know which ingredients in the inventory are below their threshold levels and need attention.
- The system must include functions that will allow the user to add a recipe, ingredient, vendor to the database.
- The user should also be able to delete any recipe from the database when not needed.
- The system must allow the user to create orders for the ingredients that are below threshold.
- The system must include a mechanism for the user so that the user can just update the sales of the day in the system and the system deducts the corresponding amount of ingredient quantity from the inventory. Thus keeping a track of ingredients.
- The system must also include functions for the user to add special days in the system when the inventory usage will be more than usual or less than usual and thus provide a way to alert the user of the possibility of over usage or under usage or certain ingredients.
- The system also must provide a prediction function to the user where the system will give the user the predicted usage of inventory of certain pre-set days.

4.2.NON-FUNCTIONAL REQUIREMENTS:

Usability

- i.** The system must be easy to use by both managers and chefs such that they do not need to read an extensive amount of manuals.
- ii.** The system must be quickly accessible by both managers and chefs.

Reliability

- i.** The System must give accurate inventory status to the user continuously. Any inaccuracies are taken care by the regular confirming of the actual levels with the levels displayed in the system.
- ii.** The System must successfully add any recipe, ingredients, vendors or special occasions given by the user and provide estimations and inventory status in relevance with the newly updated entities.
- iii.** The system must provide a password enabled login to the user to avoid any foreign entity changing the data in the system.

Performance

- i.** The system must not lag, because the workers using it don't have down-time to wait for it to complete an action.
- ii.** The system must complete updating the databases, adding of recipe, ingredient, vendor and occasions successfully every time the user requests such a process.

Supportability

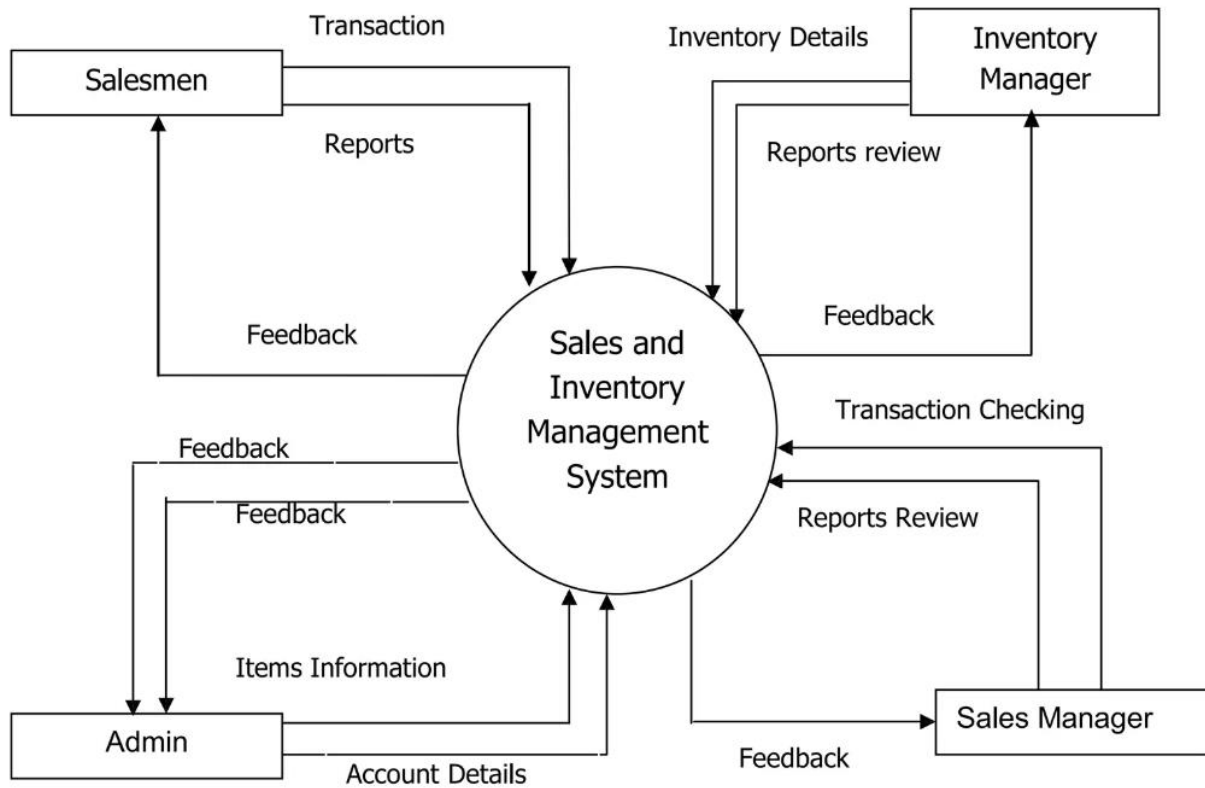
- i.** The software is designed such that it works even on systems having the minimum configuration.
- ii.** The system is adaptable even if additional plugins or modules are added at a later point.
- iii.** The data can be exported to the manager so as to make the system more portable.

Packaging

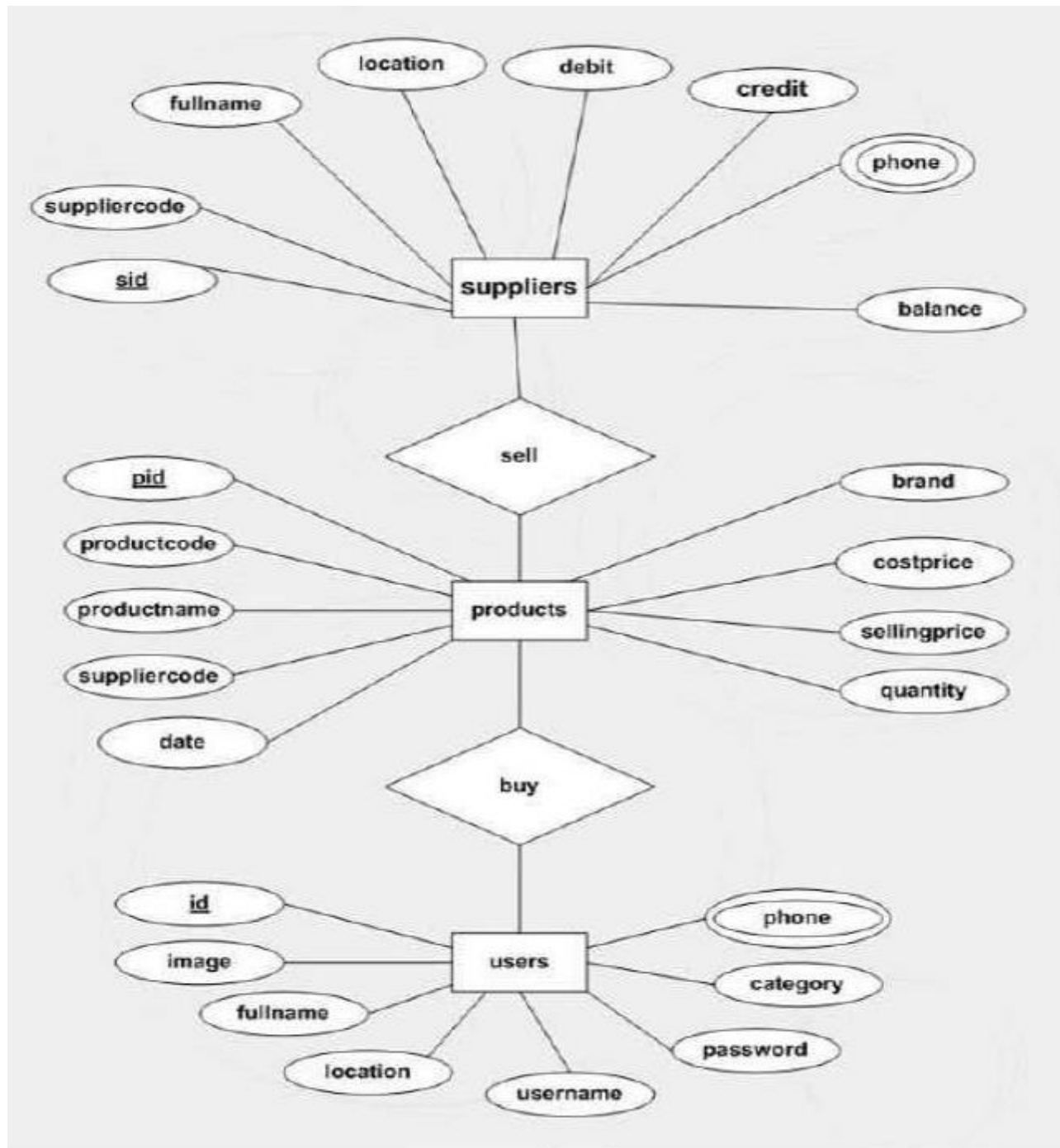
- i.** The system must be able to run on the Windows operating systems beginning with Windows XP, and must be able to run on future releases such as the upcoming Windows 8
- ii.** The software must incorporate a license key authentication process.
- iii.** The packaging must come with a manual that details the use of the system, and also the instructions on how to use the program. This manual may be included either in a booklet that comes with the software, or on the disc that the software itself is on.

5.PROJECT DESIGN:

5.1.DATA FLOW DIAGRAMS:

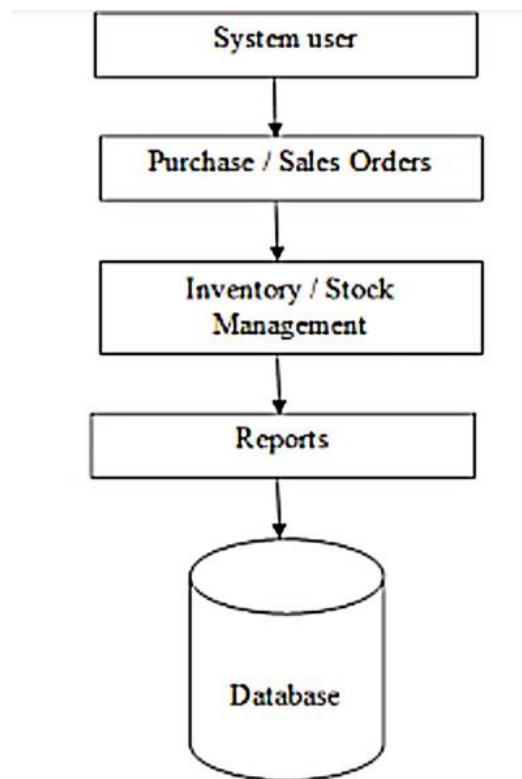


Flow Data Diagram:



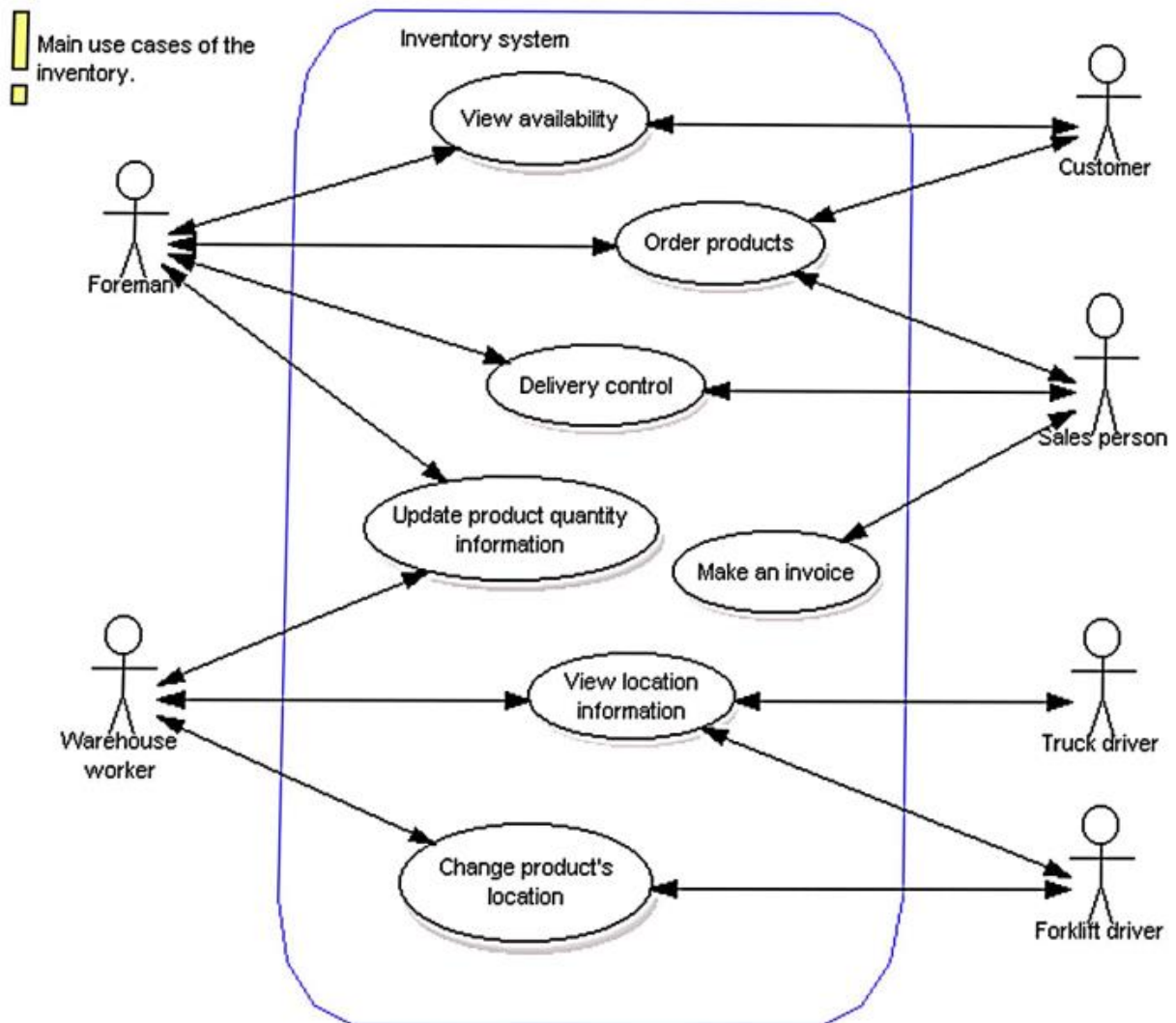
5.2. SOLUTION & TECHNICAL ARCHITECTURE:

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



5.3.USER STORIES:

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



6. PROJECT PLANNING & SCHEDULING:

6.1. SPRINT PLANNING & ESTIMATION:

Sprint 1:

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

Sprint 2:

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

Sprint 3:

- 1.Integration of Send grid into our application

Sprint 4:

- 1.Deploying the application using Docker and Kubernetes

6.2. SPRINT DELIVERY SCHEDULE:

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

Date	31 October 2022
Project Name	Inventory Management System for Retailers

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

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Yatheeswar Karthick R Vaishnav M A Vignesh M Yash P A
Sprint-1		USN-2	As a user, I can register for the application through E-mail	1	Medium	Yatheeswar Karthick R Vaishnav M A Vignesh M Yash P A
Sprint-1	Confirmation	USN-3	As a user, I will receive confirmation email once I have registered for the application	2	Medium	Yatheeswar Karthick R Vaishnav M A Vignesh M Yash P A

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Login	USN-4	As a user, I can log into the application by entering email & password	2	High	Yatheeswar Karthick R Vaishnav M A Vignesh M Yash P A
Sprint-2	Dashboard	USN-5	As a user, I can view the products which are available	4	High	Yatheeswar Karthick R Vaishnav M A Vignesh M Yash P A
Sprint-2	Add items to cart	USN-6	As a user, I can add the products I wish to buy to the carts.	5	Medium	Yatheeswar Karthick R Vaishnav M A Vignesh M Yash P A
Sprint-3	Stock Update	USN-7	As a user, I can add products which are not available in the dashboard to the stock list.	5	Medium	Yatheeswar Karthick R Vaishnav M A Vignesh M Yash P A

Sprint-4	Request to Customer Care	USN-8	As a user, I can contact the Customer Care Executive and request any services I want from the customer care.	5	Low	Yatheeswar Karthick R Vaishnav M A Vignesh M Yash P A
Sprint-4	Contact Administrator	USN-9	I can be able to report any difficulties I experience as a report	5	Medium	Yatheeswar Karthick R Vaishnav M A Vignesh M Yash P A

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

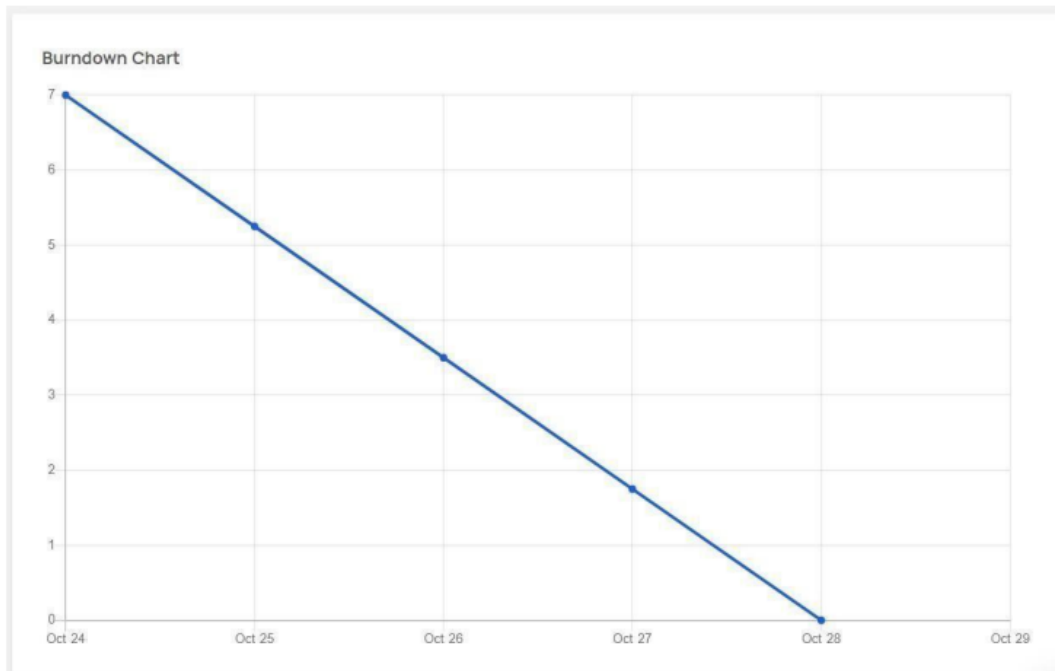
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	7	6 Days	27 Oct 2022	29 Oct 2022	7	29 Oct 2022
Sprint-2	9	6 Days	01 Nov 2022	05 Nov 2022	9	05 Nov 2022
Sprint-3	5	6 Days	07 Nov 2022	12 Nov 2022	5	12 Nov 2022
Sprint-4	10	6 Days	14 Nov 2022	19 Nov 2022	10	19 Nov 2022

Velocity:

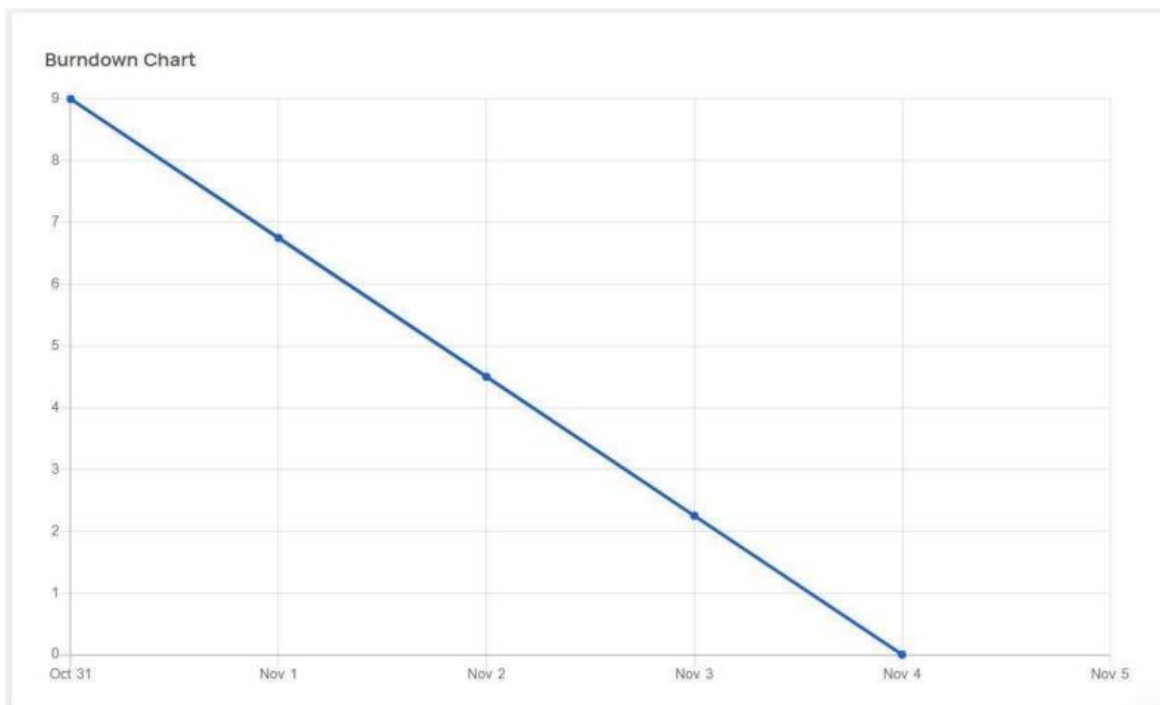
Sprints	Sprint Duration	Velocity	Actual Velocity
Sprint-1-1	6 6	7 7	0.855
Sprint-2-2	6 6	9 9	0.666
Sprint-3-3	6 6	5 5	1.2 2
Sprint-4-4	6 6	10 0	0.6 6

Burndown Chart:

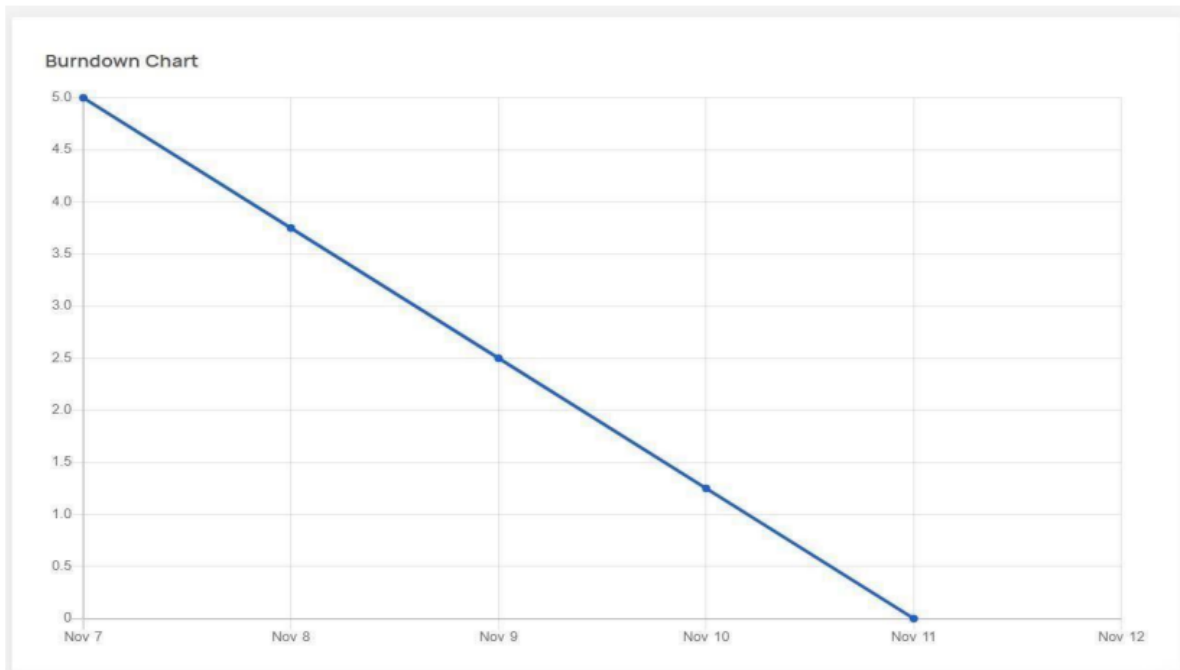
Sprint - 1



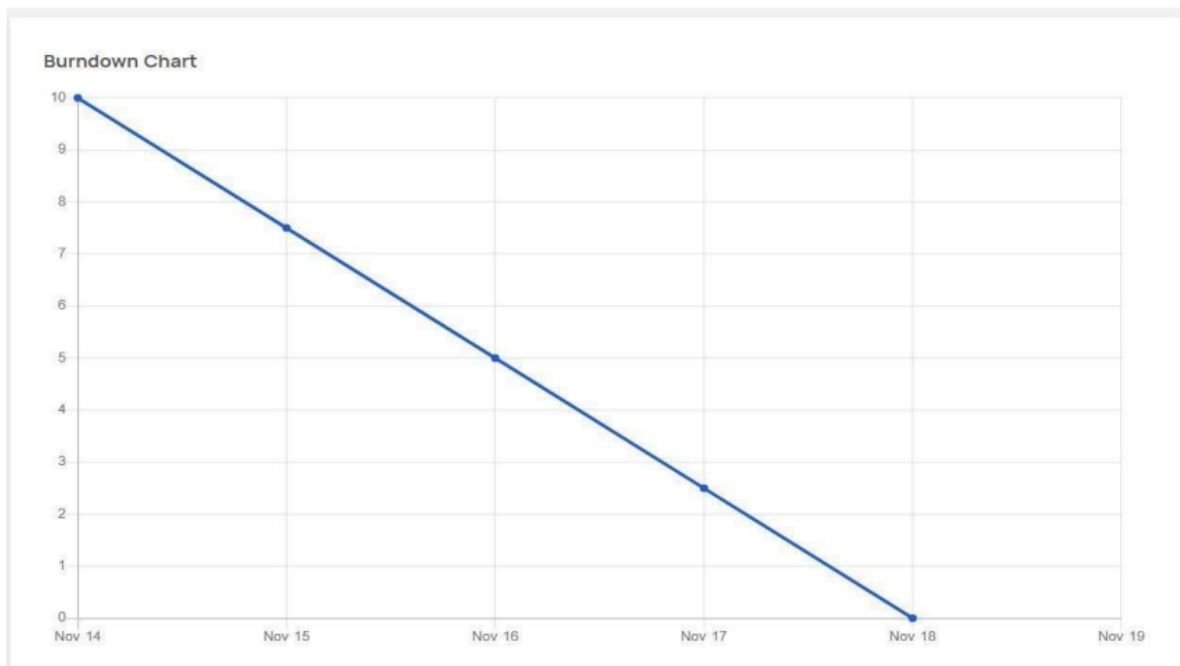
Sprint - 2



Sprint - 3



Sprint – 4 Project Tool: JIRA



6.3. REPORTS FROM JIRA:

IT organizations have the challenge of ensuring system uptime, supporting users, and managing inventory of both hardware and software. IT teams gain significant efficiencies when one tool can support multiple business operations. According to Gartner, mastering the discipline of effective asset management is a huge cost savings for companies.

7.CODING & SOLUTIONING:

7.1. FEATURE 1:



Add Product

Product Name *

Product Id *

Price *

Quantity *

ADD

7.2. FEATURE 2:



Update Product

UPDATE

7.3. DATABASE SCHEMA:

USERS TABLE:

Table definition					
USERS					
No statistics available.					
Name	Data type	Nullable	Length	Scale	
USER_ID	BIGINT	N		0	👁
USERNAME	VARCHAR	N	32	0	👁
EMAIL_ID	VARCHAR	N	32	0	👁
PASSWORD	VARCHAR	N	32	0	👁

PRODUCT TABLE:

Table definition

INVENTORY PRODUCTS

No statistics available.

Name	Data type	Nullable	Length	Scale	
PRODUCT_ID	BIGINT	N		0	👁
PRODUCT_NAME	VARCHAR	N	32	0	👁
PRODUCT_PRICE	DOUBLE	N		0	👁
PRODUCT_QUANTITY	INTEGER	Y		0	👁
PRODUCT_UPDATE_DATE	TIMESTAMP	N	10	6	👁

8. TESTING:

8.1.TEST CASES:

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

8.2.USER ACCEPTANCE TESTING:

Test case : Testing the Add Recipe Interface and its functioning

Case 1: Testing the Quantity input field.

Case 2: Testing the Recipe Name field.

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

Case 4: Testing the available ingredients list.

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

Test Case : Check Threshold Interface

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

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

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

Test Case : Testing the Update after sales interface

Case 1: Test the Recipe list box.

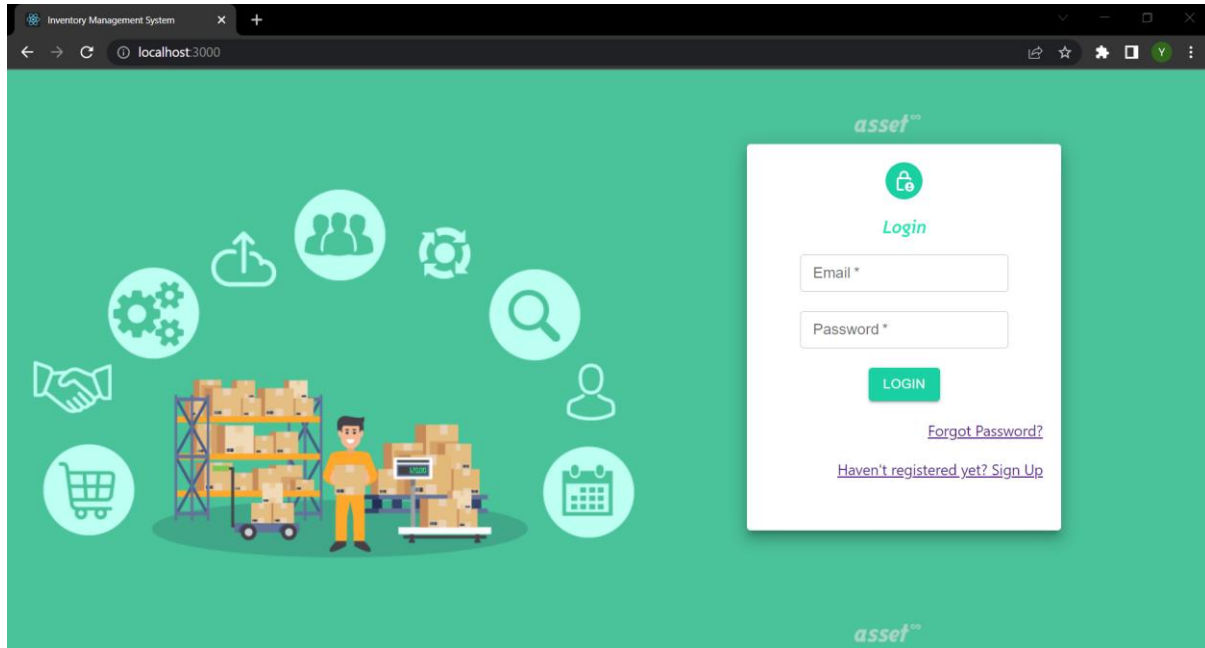
Case 2: Test the quantity text field.

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

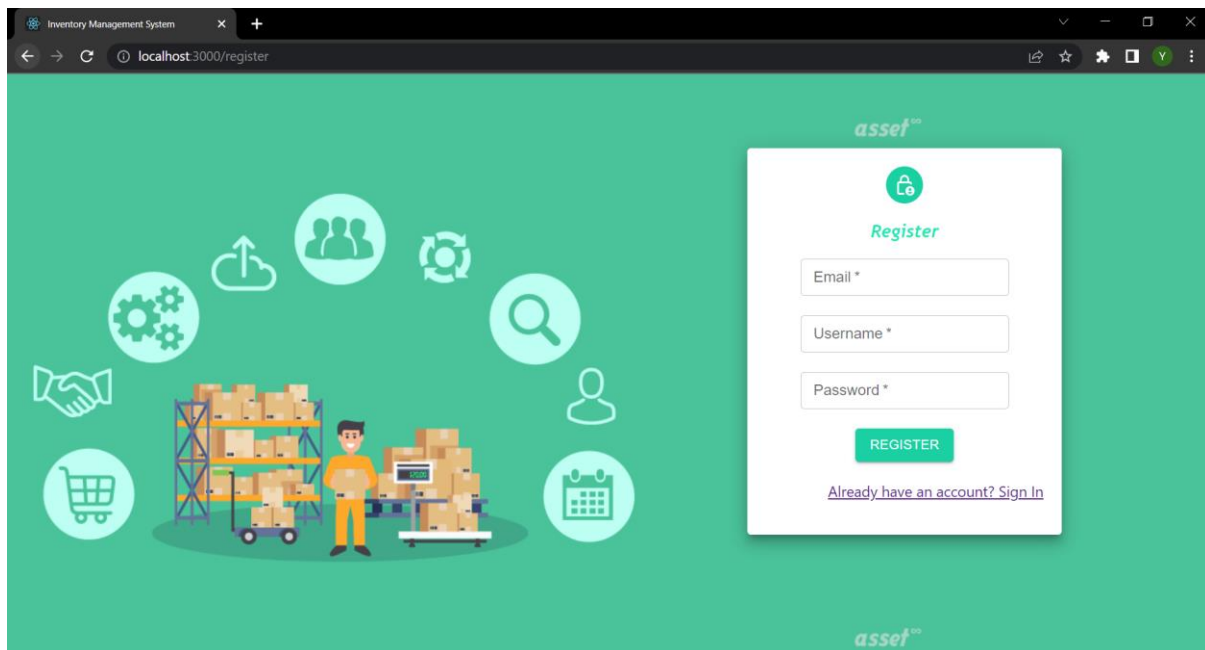
Case 4: Test if the details are updated to the database when requested.

9. RESULTS:

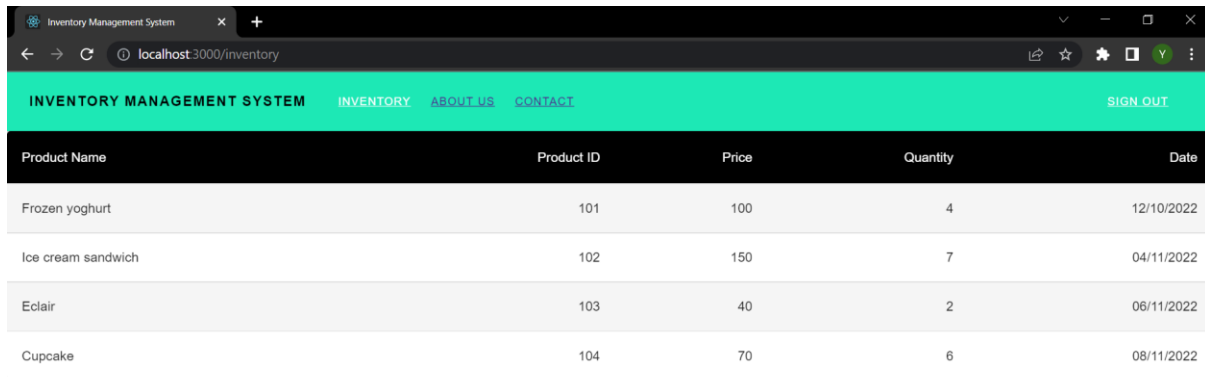
LOGIN:



REGISTER:



INVENTORY:



The screenshot shows a web browser window with the title 'Inventory Management System'. The address bar shows 'localhost:3000/inventory'. The page has a teal header with the title 'INVENTORY MANAGEMENT SYSTEM' and navigation links 'INVENTORY', 'ABOUT US', 'CONTACT', and 'SIGN OUT'. Below the header is a table with the following data:

Product Name	Product ID	Price	Quantity	Date
Frozen yoghurt	101	100	4	12/10/2022
Ice cream sandwich	102	150	7	04/11/2022
Eclair	103	40	2	06/11/2022
Cupcake	104	70	6	08/11/2022

[ADD](#) [UPDATE](#) [DELETE](#)

ADD PRODUCT:



The screenshot shows a web browser window with the title 'Inventory Management System'. The address bar shows 'localhost:3000/addform'. The page displays a form titled 'Add Product' with the following fields:

Add Product

Product Name *

Product Id *

Price *

Quantity *

[ADD](#)

UPDATE PRODUCT:



Update Product

Product Name *

Product Id *

Price *

Quantity *

UPDATE

DELETE PRODUCT:



Delete Product

Product Name *

Product Id *

DELETE

9.1. PERFORMANCE METRICS:

Inventory Performance is a measure of how effectively and efficiently inventory is used and replenished. The goal of inventory performance metrics is to compare actual on-hand dollars versus forecasted cost of goods sold. Many Lean practitioners claim that inventory performance is the single best indicator of the overall operational performance of a facility.

Inventory performance looks at and is measured using either Inventory Days On-Hand (DOH) or Inventory Turns.

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

10.ADVANTAGES & DISADVANTAGES:

ADVANTAGES:

1. **It helps to maintain the right amount of stocks:** contrary to the belief that is held by some people, inventory management does not seek to reduce the amount of inventory that you have in stock, however, it seeks to maintain an equilibrium point where your inventory is working at a maximum efficiency and you do not have to have many stocks or too few stocks at hand at any particular point in time. The goal is to find that zone where you are never losing money in your inventory in either direction. With the aid of an efficient inventory management strategy, it is easy to improve the accuracy of inventory order.
2. **It leads to a more organized warehouse:** with the aid of a good inventory management system, you can easily organize your warehouse. If your warehouse is not organized, you will find it very difficult to manage your inventory. A lot of businesses choose to optimize their warehouse by putting the items that have the highest sales together in a place that is easy to access in the warehouse. This ultimately helps to speed up order fulfilment and keeps clients happy.

DISADVANTAGES:

1. **Bureaucracy:** even though inventory management allows employees at every level of the company to read and manipulate company stock and product inventory, the infrastructure required to build such a system adds a layer of bureaucracy to the whole process and the business in general. In instances where inventory control is in-house, this includes the number of new hires that are not present to regulate the warehouse and facilitate transactions. In instances where the inventory management is in the hands of a third party, the cost is a subscription price and a dependence on another separate company to manage its infrastructure. No matter the choice you go for, it translates to a higher overhead cost and more layers of management between the owner and the customer. From the view point of the customer, a problem that requires senior management to handle will take a longer period of time before it will be trashed out.
2. **Impersonal touch:** another disadvantage of inventory management is a lack of personal touch. Large supply chain management systems make products more accessible across the globe and most provide customer service support in case of difficulty, but the increase in infrastructure can often mean a decrease in the personal touch that helps a company to stand out above the rest. For instance, the sales manager of a small manufacturing company that sells plumbing supplies to local plumbers can throw in an extra box of washers or elbows at no charge to the customer without raising any alarms. This is done for the sake of customer relations and often makes the customer feel like he is special. While free materials can also be provided under inventory management, processing time and paper work make obtaining the material feel more like a chore for the customer or even an entitlement.

11.CONCLUSION:

The project “Inventory Management System for Calculation and Ordering of Available and Processed Resources” mainly as the name suggests deals with the calculation of the available and processed resources for an accurate inventory control and process management for a domain specific client who are related to the subject of food chains/outlets. This enables the inventory to be applied at every level in the hierarchy of the products and its complex combinations of recipes. A system that accurately calculates the atomic ingredients used for making a recipe then automatically performs the back end operation pertaining to a database of many relational tables onto which the changes are being made with each and every operation performed on the front end and which also shows up if at the time of retrieval.

The most important part of Inventory controlling is its ability to check for threshold levels and alert the manager to replenish the stock before it reaches a danger zone. So as when an ingredient level goes below the threshold level then it routes an alert to the manager. Then if needed accordingly an automated order form is produced so as to each specific vendor along with the quantities needed for replenishment. As a part of the standard maintaining a drill of risk management is done in order to sustain during the days of special occasion or holidays when the demand reaches to rather more different scale as compared to other days. These occasions call on for special inclusions into the menu which reflects on the recipes and in turn reflects the ingredients being used up eventually. Thus was provided the liberty of adding special recipe to the menu for some special occasion and is regarded as a key feature.

12.FUTURE SCOPE:

- The Fourth Industrial Revolution will continue to drive technological change that will impact the way that we manage inventories.
- Successful companies will view inventory as a strategic asset, rather than an aggravating expense or an evil to be tolerated.
- Collaboration with supply chain partners, coupled with a holistic approach to supply chain management, will be key to effective inventory management.

13.APPENDIX:

SOURCE CODE:

```
import ibm_db
from flask import Flask
from flask import request
from flask_cors import CORS, cross_origin
app = Flask(__name__)
cors = CORS(app)
app.config['CORS_HEADERS'] = 'Content-Type'
global conn
try:
    conn = ibm_db.connect('DATABASE=bludb;HOSTNAME=0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=31198;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=wsj61209;PWD=38i48sUIICxGGsN2', "", "")
```

```

    print("conn success")
except:
    print("conn error ")

@app.route("/login", methods=['POST'])
@cross_origin()
def login():
    email = request.form['email']
    password = request.form['password']
    stmt = ibm_db.exec_immediate(conn, "Select password,user_id from
wsj61209.users where email_id='"+email+"'")
    result = {"message": "fail", "user_id": ""}
    if ibm_db.fetch_row(stmt) != False and ibm_db.result(stmt,0)==password:
        result["message"] = "success"
        result["user_id"] = ibm_db.result(stmt,1)
    print(result)
    return result

@app.route("/signup", methods=['POST'])
@cross_origin()
def signup():
    result = {}
    email = request.form['email']
    if mailExists(email):
        result["message"] = "fail"
    else:

```

```

        insert = "insert into wsj61209.users(name,email,password)
values('"+request.form['name']+"','"+request.form['email']+"','"+request.form['pass
word']")"

        ibm_db.exec_immediate(conn,insert)

        result["message"] = "success"

    return result

```

```

def mailExists(email):

    result = False

    stmt = ibm_db.exec_immediate(conn,"Select user_id from TLF61984.users
where email_id='"+email+"'")

    if ibm_db.fetch_row(stmt) != False:

        result = True

    return result

```

```

@app.route("/getstocks", methods=['POST'])
@cross_origin()
def getstocks():

    stocks = []

    useremail = request.form['useremail']

    print(useremail)

    stmt = ibm_db.exec_immediate(conn,"Select * from wsj61209.stock where
useremail='"+useremail+"'")

    while ibm_db.fetch_row(stmt) != False:

        stock = { }

        stock["stockid"] = ibm_db.result(stmt,0)

```

```
stock["name"] = ibm_db.result(stmt,1)
stock["quantity"] = ibm_db.result(stmt,2)
stock["minvalue"] = ibm_db.result(stmt,3)
stock["price"] = ibm_db.result(stmt,4)
stock["category"] = ibm_db.result(stmt,5)
stock["useremail"] = ibm_db.result(stmt,6)
stocks.append(stock)

return stocks
```

```
@app.route("/addstock", methods=['POST'])
@cross_origin()
def addstock():
    result = {}
    name = request.form['name']
    useremail = request.form['useremail']
    insert = "insert into
wsj61209.stock(name,quantity,minvalue,price,category,useremail)
values('"+name+"','"+request.form['quantity']+"','"+request.form['minvalue']+"','"+re
quest.form['price']+"','"+request.form['category']+"','"+useremail+"')"
    ibm_db.exec_immediate(conn,insert)
    result["message"] = "Stock Created Successfully"
    return result

@app.route("/deletestock", methods=['POST'])
@cross_origin()
```

```

def deletestock():
    result = { }
    stockid = request.form['stockid']
    ibm_db.exec_immediate(conn,"delete from TLF61984.stock where
stockid="+stockid)
    result["message"] = "Stock deleted Successfully"
    return result

@app.route("/updatestock", methods=['POST'])
@cross_origin()
def updatestock():
    result = { }
    stockid = request.form['stockid']
    name = request.form['name']
    ibm_db.exec_immediate(conn,"update wsj61209.stock set
name='"+name+"',quantity="+request.form['quantity']+" ,minvalue="+request.form
['minvalue']+" ,price="+request.form['price']+" ,category='"+request.form['category'
]+"' where stockid="+stockid)
    result["message"] = "Stock updated Successfully"
    return result

```

GITHUB : <https://github.com/IBM-EPBL/IBM-Project-21671-1659787519>

DEMOLINK:

<https://drive.google.com/drive/folders/1MGLrx4rIQbRfBH3q3UsF5yPBc6UZ3cu->