INVENTORY MANAGMENT SYSTEM FOR RETAILERS

GITHUB ID	IBM-EPBL/IBM-Project-37029-1660299701
TEAM ID	PNT2022TMID44584
TEAM LEADER	JEEVITHA K (731719205007)
TEAM MEMBER 1	DEVI T (731719205007)
TEAM MEMBER 2	SOLAIMATHI T (731719205019)
TEAM MEMBER 3	VIGNESH M (731719205023)
INDUSTRY MENTOR NAME	VASUDEVA HANUSH
FACULTY MENTOR NAME	SUJITHA A

TABLE OF CONTENTS

1.INTRODUCTION

1.1 PROJECT OVERVIEW

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.

In practice, effective retail inventory management results in lower costs and a better understanding of sales patterns. Retail inventory management tools and methods give retailers more information on which to run their businesses. Applications have been developed to help retailers track and manage stocks related to their own products. The System will ask retailers to create their accounts by providing essential details. Retailers can access their accounts by logging into the application.

Once retailers successfully log in to the application they can update their inventory details, also users will be able to add new stock by submitting essential details related to the stock. They can view details of the current inventory. The System will automatically send an email alert to the retailers if there is no stock found in their accounts. So that they can order new stock.

1.2 PURPOSE

Retail inventory management tools and methods **give retailers more information with which to run their businesses**, including: Product locations. Quantities of each product type. Which stock sells well and which doesn't, by location and sales channel.

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

2. LITERATURE SURVEY

Paper 1:

Study on manufacture inventories

Publication Year: (1970)
\
Author: Krishnamurty and Sastry

\ 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 Accelerator represented by change in sales, bank finance and shortterm interest rate was found to be an important determinant. The utilisation of productive capacity and price anticipations was also found to be relevant in the study.

Paper 2

Publication Year: (1972)

Author: George

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

Paper 3:

Study on manufacture inventories Publication

Year: (1970) Author: Krishnamurty and Sastry

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 Accelerator represented by change in sales, bank finance and shortterm interest rate was found to be an important determinant. The utilisation of productive capacity and price anticipations was also found to be relevant in the study.

2.1 EXISTING PROBLEM

The digital revolution may be upon us, but vast numbers of companies large and small still sell (and, in many cases, manufacture) physical products. These products, and the materials used to produce them, create the need for one of the most complex and challenging areas of potential value creation and loss for any business inventory management.

Theft, fraud, human error, and other problems make preventing lost value as important as gaining it through revenue.

Fortunately, technology, paired with strategic thinking, make it easier for both small businesses and large corporations to stop the bleeding and protect their profitability and productivity while meeting the needs of their customers.

2.2 REFERENCES

References numbers are used to identify and track various items in Inventory, such as inventory adjustments, location transfers, and manual counts. Reference numbers are location specific, and you can use them to search for various items.

2.3 PROBLEM STATEMENT DEFINITION

Who does the problem affect?

Common retailers who run their business with large scale or small scale stocks. What is the issue? It is crucial for an organization today to understand its

inventory to achieve both efficient and fast operations, that too, at an affordable cost. Lack of the right inventory at the right time can mean back orders, excess inventory, etc. These drive up costs. Late delivery due to stock-outs is bound to give you a bad reputation. Inaccurate calculations of stock and price.

When does the issue occur?

Late deliveries are due to late planning. Poor tracking may lead to back orders. Overstocking of discounted products and neglecting the trends of seasonal sales may result in excess inventory.

Where is the issue coming?

Due to human error, Lack of interest and consciousness. Amount of data which is beyond the limits of human power to be calculated manually

Why is it important that we fix the problem?

Nearly 81% of consumers experienced an "out- of-stock" situation in the past 12 months, resulting in lost sales for retailers and lots of disappointment for instore shoppers. Globally, retailers recorded losses of a whopping $1.75 trillion due to mismanaged inventory. Therefore considering the economic crisis of the retailers and to reduce the manpower efficiently while handling data, it is very important to have a best inventory management system for retailers.

3. IDEATION & PROPOSED SOLUTION

Proposed Solution Template:

Project team shall fill the following information in proposed solution template S.No. Parameter Description

1. Problem Statement

(Problem to be solved Irrespective of the size of the business,inventory management is one of the most challenging processes in the retail sector.In this industry, the efficiency of inventory management directly impacts customer satisfaction. As retail is a fast paced, and customer-facing sector, customer satisfaction is core to its business growth. The inventory process involves multiple intricate aspects that drive accurate product delivery. Even a single error in the process can have expensive and long-term consequences. This will eventually affect the company's growth and reputation.The inventory issue

refers to the general issue of deciding how much inventory to keep on hand in expectation of possible demand.Loss occurs when a business is unable to meet demand or when commodities are stocked for which there is no demand.

2. Idea / Solution description

So to provide a solution to this problem of retailers, an inventory management system with easy to operate and access mechanism can be used to track the inventory of a single store or to manage the delivery of stock between several branches of a larger franchise. However, the system merely records sales and restocking data and provides warning of low stock at any location through email at a specified interval.

3. Novelty / Uniqueness

The right products at the right time. To set automatic reorder points based on preset stock levels and current availability to avoid overselling. To use multi-location warehouse management features to track and control expanding inventories. To use inventory control processes like blind receiving with barcodes and mobile scanners to prevent human error, inventory manipulation and shrinkage due to theft or negligence. To introduce dashboards with simple interfaces that show real time inventory data. To use RFID reader and manage products with the RIOT application to generate accurate data

4. Social Impact / Customer Satisfaction

When product is returned because it is damaged or dead on arrival, and it is still under warranty, you can arrange with the manufacturer to do an instant swap of the product to keep the customer happy. To reduce the amount of time that products sit on your shelves. When you don't carry extra inventory for extended periods of time, your inventory costs decrease. This is a savings that you can pass on to clients in the form of lower pricing. By being able to give clients accurate inventory information, you improve the image of your company and add one more element to customer retention. To have popular items in stock and ready to instantly fulfill any customer’s order.

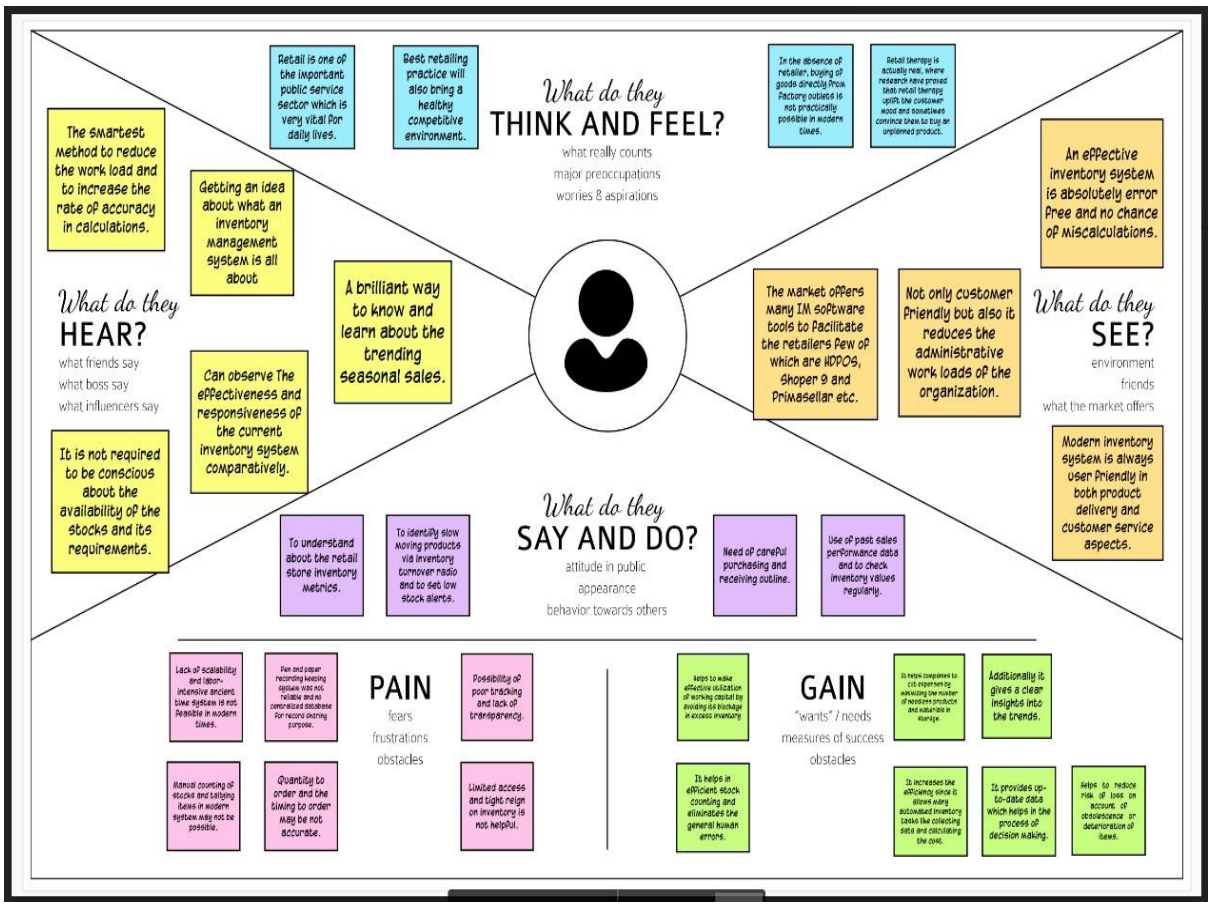
5. Business Model (Revenue Model)

It helps companies to identify which and how much stock to order at what time. It tracks inventory from purchase to sale of goods. It is suited for situations where a business is expecting to grow. It starts with adopting the right inventory technology to allow for more effective supply chain management, cutting down on costs, reducing waste and reducing the overall carbon footprint.

6. Scalability of the Solution

Give employees the right inventory tools for the job. They need software to replace manual inventory documentation, and paperless transactions for invoices and purchase orders. Add images with product descriptions in your inventory database to improve purchasing and receiving processes, enhance accuracy and prevent misplaced inventory. Inventory management ensures control over customers demands thereby resulting to customer satisfaction and increase financial performance. Thus the proposed system provides to be a user friendly and makes it cheaply available.

3.1 EMPATHY MAP CANVAS



3.2 IDEATION & BRAINSTORMING

3.3 PROPOSED SOLUTION

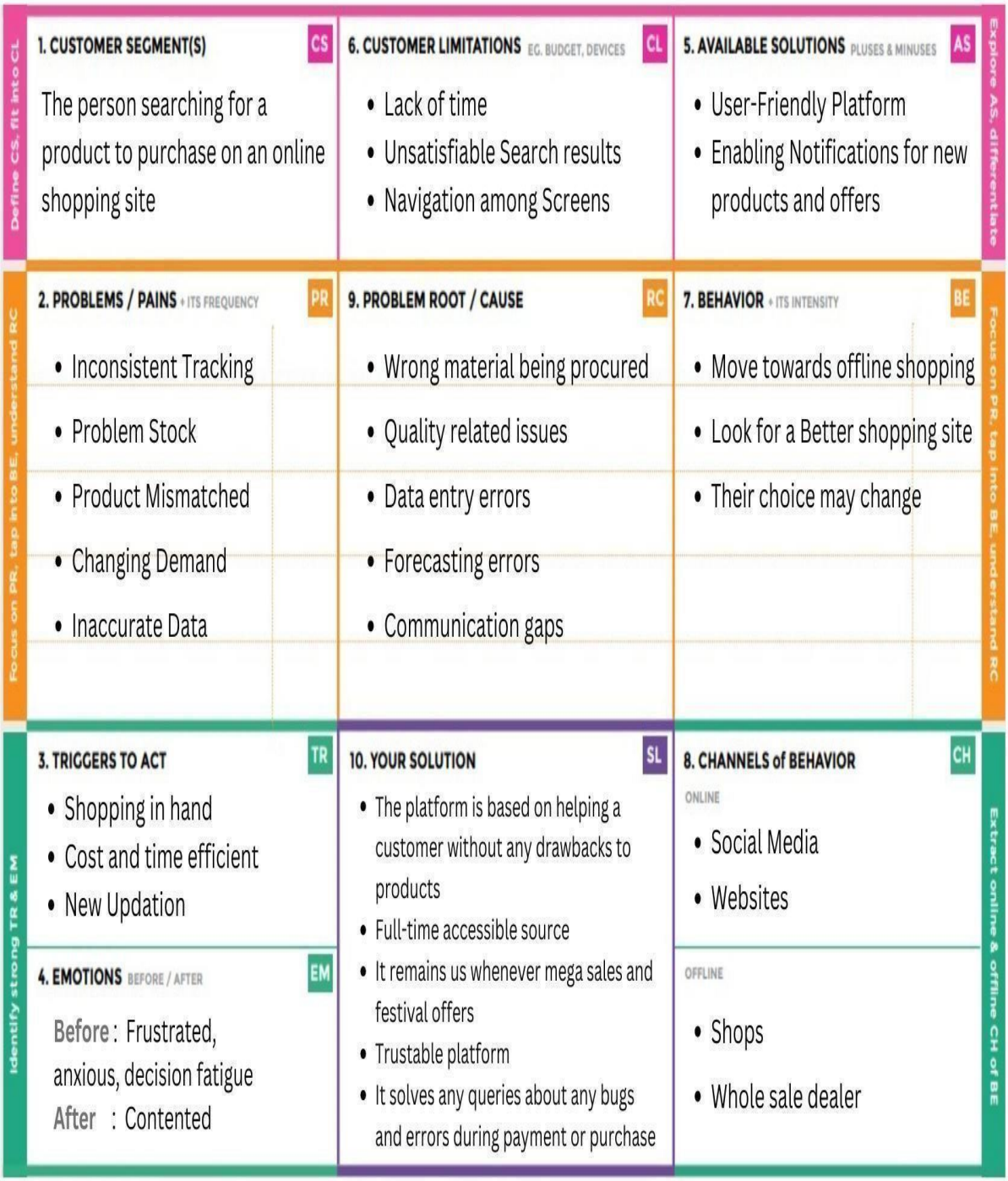
Project team shall fill the following information in proposed solution template

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved	Irrespective of the size of the business,inventory management is one of the most challenging processes in the retail sector.In this industry, the efficiency of inventory management directly impacts customer satisfaction. As retail is a fast paced, and customer-facing sector, customer satisfaction is core to its business growth. The inventory process involves multiple intricate aspects that drive accurate product delivery. Even a single error in the process can have expensive and long-term consequences. This will eventually affect the company's growth and reputation.The inventory issue refers to the general issue of deciding how much inventory to keep on hand in expectation of possible demand.Loss occurs when a business is unable to meet demand or when commodities are stocked for which there is no demand.
2.	Idea / Solution description	So to provide a solution to this problem of retailers, an inventory management system with easy to operate and access mechanism can be used to track the inventory of a single store or to manage the delivery of stock between several branches of a larger franchise. However, the system merely records sales and restocking data and provides warning of low stock at any location through email at a specified interval.
3.	Novelty / Uniqueness	The right products at the right time. To set automatic reorder points based on preset stock levels and current

		availability to avoid overselling. To use multi-location warehouse management features to track and control expanding inventories. To use inventory control processes like blind receiving with barcodes and mobile scanners to prevent human error, inventory manipulation and shrinkage due to theft or negligence. To introduce dashboards with simple interfaces that show real time inventory data. To use RFID reader and manage products with the RIOT application to generate accurate data
4.	Social Impact / Customer Satisfaction	When product is returned because it is damaged or dead on arrival, and it is still under warranty, you can arrange with the manufacturer to do an instant swap of the product to keep the customer happy. To reduce the amount of time that products sit on your shelves. When you don't carry extra inventory for extended periods of time, your inventory costs decrease. This is a savings that you can pass on to clients in the form of lower pricing. By being able to give clients accurate inventory information, you improve the image of your company and add one more element to customer retention. To have popular items in stock and ready to instantly fulfill any customer's order.
5.	Business Model (Revenue Model)	It helps companies to identify which and how much stock to order at what time. It tracks inventory from purchase to sale of goods. It is suited for situations where a business is expecting to grow. It starts with adopting the right inventory technology to allow for more effective supply chain management, cutting down on costs, reducing waste and reducing the overall carbon footprint.

6.	Scalability of the Solution	Give employees the right inventory tools for the job. They need software to replace manual inventory documentation, and paperless transactions for invoices and purchase orders. Add images with product descriptions in your inventory database to improve purchasing and receiving processes, enhance accuracy and prevent misplaced inventory. Inventory management ensures control over customers demands thereby resulting to customer satisfaction and increase financial performance. Thus the proposed system provides to be a user friendly and makes it cheaply available.

3.4 PROBLEM SOLUTION FIT



4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	**User Registration**	Registration through Form Registration through Gmail
FR-2	**User Confirmation**	Confirmation via Email Confirmation via OTP
FR-3	**User Login**	Username/Email-ID Login with Password
FR-4	**Admin Login**	Login with Username/Email-ID Login with Password
FR-5	**Inventory Management**	Track quantity of products present in inventory at any instant
FR-6	**Tracking of stock**	Notifications through Email

4.2 NON-FUNCTIONAL REQUIREMENT

Following are the non-functional requirements of the proposed solution

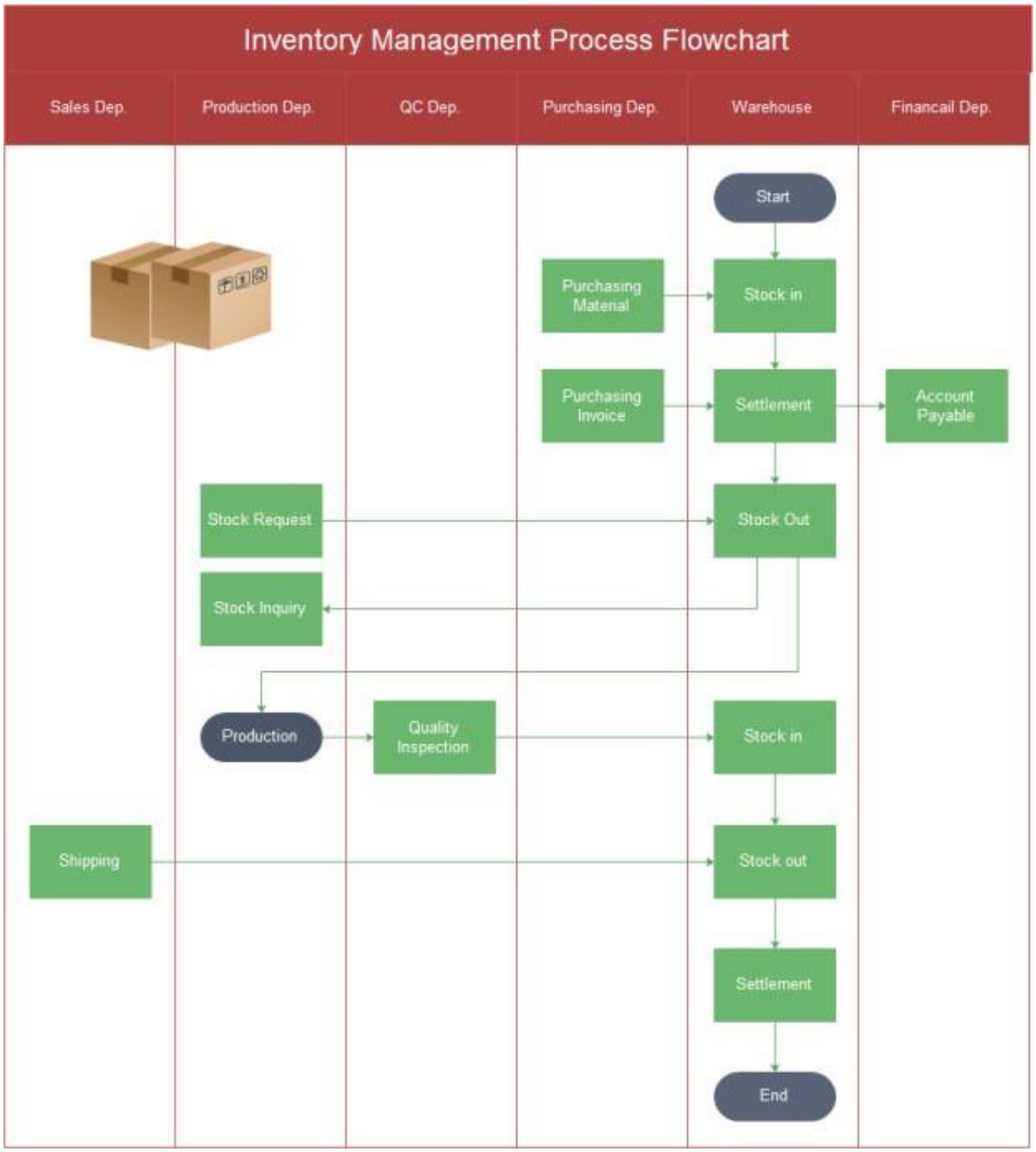
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	This cloud web application makes the process of inventory management a lot easier which saves money and time both. This system is highly responsive to both desktop and mobile users.
NFR-2	**Security**	Inventory security aims to prevent inventory losses – for example, due to incorrect storage, theft, or incorrect incoming goods inspection – so that the correct stock is always available.
NFR-3	**Reliability**	The availability of products should be properly updated for customer satisfaction. The out-of-stock information should be notified. The system must give accurate inventory status to the user continuously.
NFR-4	**Performance**	The companies have to design and operate materials management and product distribution functions effectively. Inventory control systems enable a business to determine and maintain an optimum level of investment in inventory in order to achieve the required operational performance.

NFR-5	**Availability**	The software will be available only to the administrator of the organization and the product, as well as customer details, will be recorded by him. He can manage the inventory.
NFR-6	**Scalability**	The System can manage large inventory and provides quick access to the inventory in no time.

5. PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



DATA FLOW DIAGRAM FOR IVENTORY MANAGEMENT SYSTEM FOR RETAILERS

5.2 SOLUTION & TECHNICAL ARCHITECTURE

Solution Architecture:

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed, and delivered.

Example - Solution Architecture Diagram:

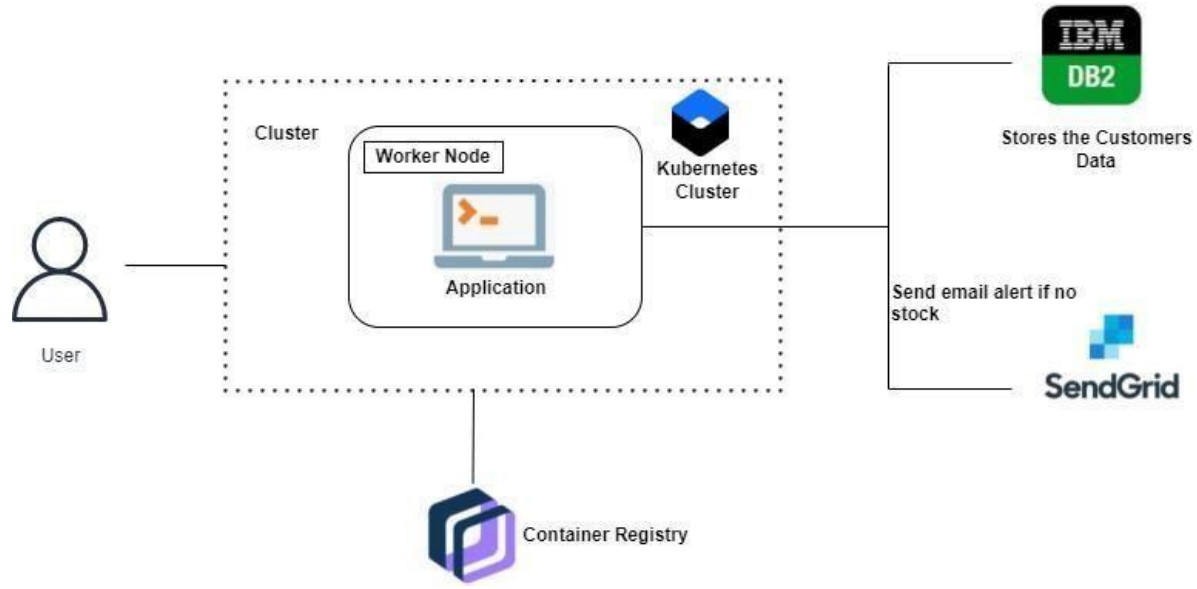


Figure 1: Architecture and data flow of the Inventory Management System for Retailers

5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can register & application Through Gmail	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can access my account	High	Sprint-1
	Dashboard	USN-6	As a user,i can log into my account for the mobile	I can access my account /Dashboard	High	Sprint-1
Customer (Webuser)	Registration	USN-7	As a user,I can register for the application by entering my email, password, and confirming my password	I can access my account/Dashboard	High	Sprint-1
		USN-8	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-9	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-10	As a user ,I can upload a Profile photo and add my name to my account	I can upload my Profile photo/Name in my account	Medium	Sprint-1
Customer Care Executive	Customer Support	USN-11	As a user,I can support for customers to handle queries and complaints from their	I can support for customers to clear	High	Sprint-1

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
			customers	complaints		
Administrator	Responsibility	USN-12	As a system administrator I want to be able to add new users when required so that	I Can add new users	High	Sprint -1

6. PROJECT PLANNING & SCHEDULING

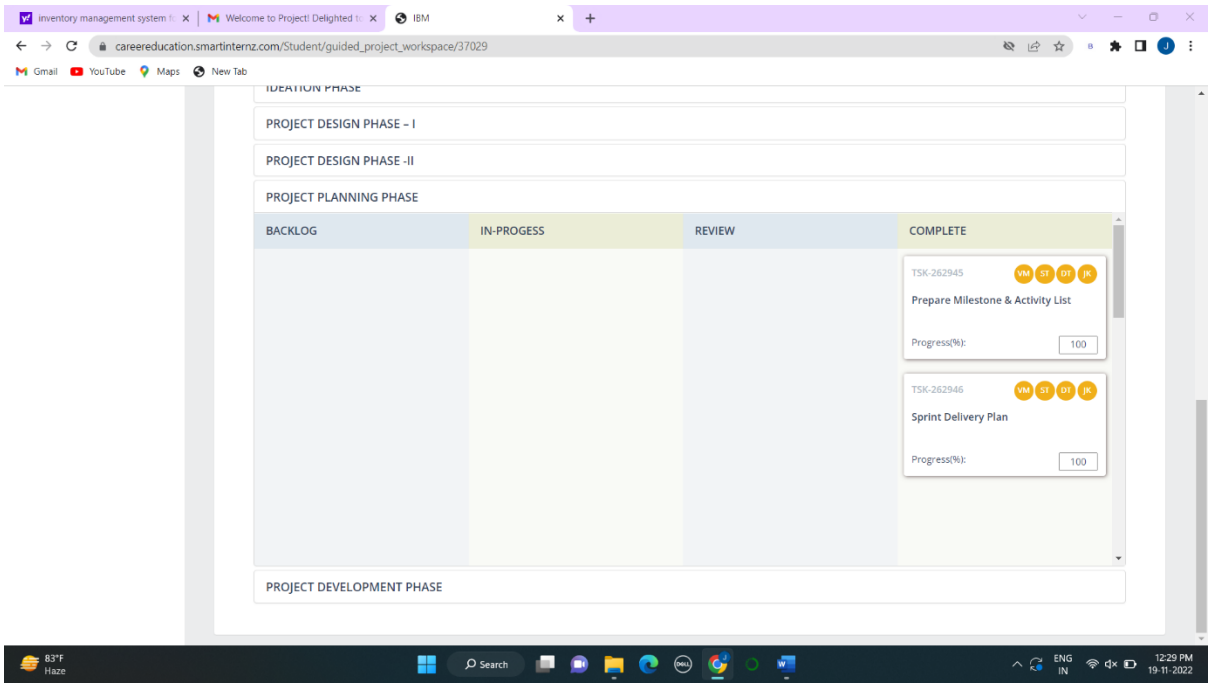
6.1 SPRINT PLANNING & ESTIMATION

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

6.2 SPRINT DELIVERY SCHEDULE

Sprint	Total Story Points	Duration	Sprint StartDate	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

6.3 REPORT FROM JIRA



7. CODING & SOLUTIONING

7.1 FEATURE 1

Inventory Control: Inventory control handles products that are already in stock at the warehouse and plays a key role in supply chain management. Inventory control tools can categorize products by type, location and SKU (or serial number), audit data, generate reports in real-time and search, filter and view products. For more on inventory control methods, including expert advice, see our"Essential guide to inventory control" .

Inventory Management: Inventory management features govern the data from other parts of the system, like inventory control. That's not all; inventory management also handles business processes that occur before the stock arrives at a warehouse and how the inventory reaches other destinations. These features include tools for multi-location warehouse management and integrations with other software or enterprise resource planning platforms .

7.2 FEATURE 2

Inventory Tracking: To implement inventory controls, you need to track the status of products and materials in the supply chain. Perpetual inventory tracking features help automate manual tasks. For example, the system automatically generates a tracking number when it creates a receipt or invoice. Tracking integrations with third-party logistics (3PL) providers combined with an email solution lets businesses better manage customer relationships by sharing what's in stock or shipping times.

Inventory Barcoding: Barcoding software helps eliminate data entry errors and automate business functions that require communication with other parts of the system. Collecting, storing and organizing digital inventory data makes inventory operations faster and more accurate. Barcoding software can integrate with digital documentation and reporting features (for example, touchscreen signatures and paperless invoicing). Mobile barcode scanning devices improve inventory accuracy, speed up back-office processes and stock replenishment and enable paperless documentation.

Inventory Optimization: Take inventory planning to a higher level with inventory optimization. Add sophistication to a basic inventory plan. Instead of standard ordering formulas and a basic ordering process, you can use tools that provide automated reports, inventory trends and a view of changes across the entire supply chain. This information enables a closer match in supply and demand so you can optimize the inventory on-hand.

8. TESTING

8.1 TEST CASES

Sprints	Sprint Duration	Velocity	Actual Velocity
Sprint-1	6	7	0.85
Sprint-2	6	9	0.66

Sprint-3	6	5	1.2
Sprint-4	6	10	0.6

9. RESULTS

9.1 PERFORMANCE METRICS

Good Inventory Performance is the most important point for successfully managing any business, as many seek to achieve good inventory management but face a large number of problems that may cause them huge losses, but by measuring and monitoring Inventory Performance on an ongoing basis, merchants will be able to avoid problems that may It results in a loss of customer confidence or lack of profits, and thus negatively affects your business, and there are a set of important inventory performance metrics that will help you effectively in improving your inventory performance, and in this article we will show you the most important inventory performance metrics that you should use.

There are a number of important things to consider when choosing stock performance metrics, as each indicator you include in your business will clearly impact all of your operations.

Therefore, you need to choose inventory performance metrics that synergize your employees and help them achieve the end goal, so you can achieve your organization's goals in an orderly fashion.

You also need to make inventory performance metrics measurable, specific, and relevant to your goals to achieve them in a timely manner.

Avoid inventory performance metrics that will not improve your inventory performance. You will also need to use an easy-to-use and customizable inventory software to be able to measure and track inventory performance metrics effectively and periodically.

10. ADVANTAGES &DISADVANTAGES

ADVANTAGES:

- Materials availability.
- Better level of customer service.
- Maintaining sufficient stock.
- Keeping wastage and losses to a minimum.
- Optimizing product sales.
- Optimizing storage cost.
- Manage inventory.
- Better understanding of sales patterns.
- Improve productivity.
- Increase profits.
- Regular supply at reasonable prices.

DISADVANTAGES:

Even with an efficient inventory management method, you can control but not eliminate business risk.

The control of inventory is complex because of the many functions it performs. It should thus be viewed as a shared responsibility.

Holding inventory can result to a greater risk of loss to devaluation (changes in price).

11. CONCLUSION

An effective inventory management system helps to reduce stock-related costs such as warehousing, carrying, and ordering costs.

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.

From another perspective, retail inventory costs consist of more than the wholesale price of the merchandise. You have to factor in costs such as storage, insurance, transportation, shrinkage, handling and more.

12. FUTURE SCOPE

For many companies, inventory management is where all the elements of the supply chain converge.

Inventory may be seen as the bloodstream of any competitive business. Its unobstructed flow is critical to the well being of any organization that trades in tangible goods. Like the life-sustaining “oxygen” that is carried by the blood to various parts of the human body, inventory carries value for manufacturing, marketing, and sales. Like the doctor who extracts a few samples of blood from a patient’s arm in order to evaluate health and wellness indicators, a view of inventory can say a lot about a business. Accumulation of unnecessary inventory is frequently symptomatic of bigger problems that lurk somewhere in a company’s systems infrastructure and processes. Its significance cannot be overstated.

The forces of technology, globalization, and consumer empowerment have profoundly influenced the way that industries have managed inventory in the past thirty years or more.

Tools that range from the ubiquitous but powerful Excel spreadsheet to Warehouse Management Systems (WMS) to Demand Planning and Forecasting Systems to multifunctional and integrative Enterprise Resource Planning (ERP) Systems have helped us to make inventory-related computations at exceedingly high speeds, and have improved inventory record accuracy substantially.

13. APPENDIX

SOURCE CODE

Login.html

```
<html>
    <head>
        <title>login</title>
        <link rel="stylesheet" href="color.css">
        </head>
    <style>
        .border{
    height: 30rem;
    width: 25rem;
    border-radius: .5rem;
    margin: 2rem;
    cursor: pointer;
    overflow: hidden;
    position: relative;
    box-shadow: 0 0 .5rem #333;
    left: 300px;
    top: 45px;
}
.border h1{
 color: white;
}
.border h2{
color: rgb(248, 246, 246);
}
.border p{
        color:white;
}
.border a{
    color: rgb(244, 242, 246);
}
.border button {
 outline: none;
 border: none;
 background-color: blue;
 color: white;
 padding: 10px;
 font-size: 17px;
 width: 150px;
```

```
}
.border button a{
    color:white;
}
body {
      background-image: url('picture/pic.png');
      background-repeat: no-repeat;
      background-attachment: fixed;
      background-size: 100% 100%;
    }
    </style>
<body>
<center >
<div class="border">
    <h1>LOGIN</h1>
    <form method="post">
    <b><p><h2>Username</h2></p></b>
  <input type="text" name="username" autofocus="" autocapitalize="none"
autocomplete="username" required="" id="id_username"><br>
  <b><p><h2>Password</h2></p></b>
  <input type="password" name="password" autocomplete="current-password"
required="" id="id_password"><br>
  <i class="far fa-eye" id="togglePassword" style="margin-left: -30px; cursor:
pointer;"></i><br><br>
  <button type="submit"  id="button"><a href="nav.html" >Login</a></button>
  <p>Already don't have an account?</p>  <a
href="register.html">Register</a>
</form>
</div></center>
</body>
</html>
```

Register.html

```
<html>
    <head>
        <link rel="stylesheet" href="co.css">
    </head><body>
    <div class="container">
<form id="contact">
            <h1>Registration Form</h1>
            <h3>Fill the form below and press the submit button!</h3>
            <div class="row">
                <!-- first column -->
```

```
                <div class="column">
                    <fieldset>
                        <input type="text" placeholder="Full Name *"
name="name" >
                    </fieldset>
                    <fieldset>
                        <input type="email" placeholder="Email Id *"
name="email">
                    </fieldset>
                    <fieldset>
                        <input type="password" placeholder="password *"
name="password" >
                    </fieldset>
                    <!-- adding all country code list -->
                    <fieldset>
                        <input type="text" placeholder="Phone number *"
name="phone" id="phone">
                    </fieldset>
                </div>
                <!-- second column -->
                <div class="column">
                    <fieldset>
                        <input type="text" placeholder="Address *"
name="address" id="address" >
                    </fieldset>
                    <fieldset>
                        <input type="text" placeholder="Land Mark *"
name="Land Mark" id="land mark" >
                    </fieldset>
                    <fieldset>
                        <input type="text" placeholder="District *"
name="district" id="address" ></fieldset>
                         <fieldset>
                        <input type="text" placeholder="Pin code *"
name="pincode" id="pincode" >
                    </fieldset>
                   </div>
            </div>
            <!-- submit button -->
            <fieldset>
                <button type="submit"  id="button"><a href="lo.html" >submit
now</a></button>
                </fieldset></div>
</form>
   </body>
    </html>
```

Common.css

```
* {
    padding: 0;
    margin: 0;
    box-sizing: border-box;
   }
   body {
    font-family: 'Open Sans', sans-serif, helvetica, Arial;
    font-weight: 400;
    font-size: 14px;
    color: black;
    /* body background image */
    background-image: linear-gradient(to bottom, rgba(128, 128, 128, 0.541),
rgba(171, 37, 238, 0.637)), url(image.jpg);
    background-attachment: fixed;
    background-size: cover;

   }
   .container {
    max-width: 800px;
    width: 100%;
    margin: 0 auto;
   }
   #contact {
    background-color: rgba(25, 23, 25, 0.7);
    padding: 20px;
    margin: 50px 0;
   }
   #contact input, button {
    font: 400 15px 'Open Sans', sans-serif, helvetica, Arial;
   }
   #contact h1 {
    font-size: 35px;
    font-weight: bold;
    text-align: center;
    color: rgb(246, 246, 250);
   }
   #contact h3 {
    margin: 5px 0px 15px;
    text-align: center;
    color:rgb(150, 150, 150)
   }
   .row {
    display: flex;
    width: 100% !important;
    padding: 20px 0px;
```

```
}
.row .column {
 margin: 0px 20px;
 width: 50%;
}
fieldset {
 border: medium none !important;
 margin:  0 0 10px;
 min-width: 100%;
 width: 100%;
}
#contact input {
 width: 100%;
 border: 1px solid rgb(150, 150, 150);
 background-color: white;
 padding: 10px;
 margin: 5px 0;
}
input[type = "radio"] {
 width: 10% !important;
}
#contact .row .radio {
 border: 1px solid rgb(150, 150, 150);
 background-color: white;
 margin: 7px 0 10px;
 padding: 5px;
}

#contact input:hover {
 transition: border-color 0.3s ease-in-out;
 border: 1px solid rgb(68, 68, 68);
}
#contact button {
 outline: none;
 border: none;
 background-color: rgb(176, 176, 185);
 color: rgb(25, 24, 24);
 margin: 0 0 5px 40%;
 padding: 10px;
 font-size: 17px;
 width: 150px;

}

#contact button:hover {
 background-color: rgba(0, 0, 255, 0.8);
}
```

DB

```
Const config =
require("../config/auth.config"); const db =
require("../models"); const User = db.user;
const Role = db.role;
 var jwt =
require("jsonwebtoken"); var
bcrypt = require("bcryptjs");
```

```
exports.signup = (req, res) => { const user = new
  User({ username: req.body.username, email:
  req.body.email, password:
  bcrypt.hashSync(req.body.password, 8),
  });

  user.save((err, user) => { if (err) {
    res.status(500).send({ message: err });
    return;
    }

    if (req.body.roles) {
      Role.find(
        { name: { $in: req.body.roles
          },
        },
        (err, roles) => { if (err) {
          res.status(500).send({ message: err });
          return;
          }

          user.roles = roles.map((role) =>
          role._id); user.save((err) => { if (err)
          { res.status(500).send({ message: err });
          return;
            }

            res.send({ message: "User was registered successfully!" });
          });
        }
      );
    } else {
      Role.findOne({ name: "user" }, (err, role)
        => { if (err) { res.status(500).send({
        message: err }); return;
        }

        user.roles = [role._id]; user.save((err)
        => { if (err) { res.status(500).send({
        message: err }); return;
```

```
          }

          res.send({ message: "User was registered successfully!" });
        });
      });
    }
  });
};

exports.signin = (req, res) => {
  User.findOne({ username:
    req.body.username,
  })
    .populate("roles", "-_v")
    .exec((err, user) => { if (err) {
      res.status(500).send({ message: err });
      return;
      }

      if (!user) { return res.status(404).send({ message: "User
        Not found." });
      }

      var passwordIsValid =
        bcrypt.compareSync(
        req.body.password, user.password
      );

      if (!passwordIsValid) { return res.status(401).send({
        message: "Invalid Password!" });
      }

      var token = jwt.sign({ id: user.id }, config.secret, {
        expiresIn: 86400, // 24 hours
      }); var authorities

      = [];

      for (let i = 0; i < user.roles.length; i++) {
        authorities.push("ROLE_" +
        user.roles[i].name.toUpperCase());
      } req.session.token =

      token;

      res.status(200).send({
        id: user._id,
```

```
        username:
        user.username,
```

```
            email: user.email,
roles: authorities,
      });
    });
};

exports.signout = async (req, res) => { try { req.session = null;
  return res.status(200).send({ message: "You've been signed out!"
  });
  } catch (err) {
    this.next(err);
  }
};
```

Navigationbar.html

```
<html>
    <head>
        <link rel="stylesheet" href="nav.css">
    </head>
    <style>
        body {
      background-image: url('picture/th.jpg');
      background-repeat: no-repeat;
      background-attachment: fixed;
      background-size: 100% 100%;
    }
    </style><header id="header">
    <nav>
        <ul>


            <li><a href="dase.html">HOME</a></li>
            <li><a href="contact.html">CONTACT</a></li>
            <li><a href="about.html">ABOUT US</a></li>
            <li><a href="lo.html">LOGOUT</a></li>

        </ul>
    </nav>
    </header>
```

```
    </html>
```

Navigation.css

```
#header{
    width: 100%;
    position: fixed;
    top: 0;
    left: 0;
    display: flex;
    align-items: center;
    justify-content: space-between;

    z-index: 1000;
    background: #333;
    height: 3rem;
    box-shadow: 1rem 0 .5rem #000;

}
     ul{
            display:flex;
            align-items: center;
            justify-content: space-between;
            list-style: none;
            left: 0;
        }

    li{
                margin-left: 2rem;
                padding: 20px;
            }


                a{
                    color: #fff;
                    font-size: 1rem;
                    right: 0px;
                }

                    a:hover{
                        color:#FFF200;
                        text-transform: none;
                    }
                }
            }
        }
```

```
    }
    body{
        color:#FFFFF0;
        position: relative;
    }
}
}
```

HOME PAGE

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
body {
  font-family: "Lato", sans-serif;
  background-image: url('picture/12.jpeg');
  background-repeat: no-repeat;
      background-attachment: fixed;
      background-size: 100% 100%;
    }
.sidenav {
  height: 100%;
  width: 250px;
  position: fixed;
  z-index: 1;
  top: 0;
  left: 0;
  background-color: rgb(107, 12, 239);
  overflow-x: hidden;
  padding-top: 20px;
}
.sidenav a {
  padding: 6px 8px 6px 16px;
  text-decoration: none;
  font-size: 25px;
  color:white;
  display: block;
}
.sidenav a:hover {
  color: black;
}
.sidenav button a{
  color:darkblue;
```

```
}
.main {
  margin-left: 160px; /* Same as the width of the sidenav */
  font-size: 28px; /* Increased text to enable scrolling */
  padding: 0px 10px;
}
@media screen and (max-height: 450px) {
  .sidenav {padding-top: 15px;}
  .sidenav a {font-size: 18px;}
}
</style>
</head>
<body>
<div id="mySidenav" class="sidenav">
  <a href="product.html">PRODUCTS</a><br><br>
  <a href="addproduct.html">ADD PRODUCT</a><br><br>
    <a href="sales.html">SALES REPORT</a><br><br>
  <a href="suppliers.html">SUPPLIERS</a><br><br>
  <a href="addsuppliers.html">ADD SUPPLIYERS</a><br><br><br>
  <button type="submit"  id="button"><a href="nav.html" >Back</a></button>
</div>
<div class="main">
</div>
</body>
</html>
```

CONTACT PAGE

```
<!DOCTYPE html>
<head>
    <title>Contact Form</title>
</head>
<style>
    body {
  background-image: url('picture/th (8).jpg');
  background-repeat: no-repeat;
  background-attachment: fixed;
  background-size: 100% 100%;
}
</style><header id="header">
<body>
    <center>
    <h3><b>Contact us</b></h3>
```

```
    <form id="fcf-form-id" class="fcf-form-class" method="post"
action="contact-form-process.php">
            <label for="Name" class="fcf-label">Your name</label><br>
            <div class="fcf-input-group">
                <input type="text" id="Name" name="Name" class="fcf-form-
control" required>
            </div><br>
        <div class="fcf-form-group">
            <label for="Email" class="fcf-label">Your email
address</label><br>
            <div class="fcf-input-group">
                <input type="email" id="Email" name="Email" class="fcf-form-
control" required>
            </div><br>
        <div class="fcf-form-group">
            <label for="Message" class="fcf-label">Your message</label><br>
            <div class="fcf-input-group">
                <textarea id="Message" name="Message" class="fcf-form-control"
rows="6" maxlength="3000" required></textarea>
            </div><br>
        </div>
        <div class="fcf-form-group">
            <button type="submit" id="fcf-button" class="fcf-btn fcf-btn-
primary fcf-btn-lg fcf-btn-block">Send</button>
        </div>
        </div></form>
    <p>contact number : 12345678901<br>09876543211</p>
<p>E-Mail : inventoryretailer@gmail.com</p></center>
</body>
</html>
```

ABOUT PAGE

```
<html>
    <head>
        <title>about us</title>
    </head>
    <style>
        .flp{
```

```
            font-size: 25px;
            font-weight: 100;
        }
        body {
  background-image: url('picture/io.webp');
  background-repeat: no-repeat;
  background-attachment: fixed;
  background-size: 100% 100%;
        }

    </style>
    <body>
        <center>
            <h2>ABOUT US</h2>
        <h1>INVENTORY MANAGEMENT SYSTEM FOR RETAILERS</h1>
        </center>
        <div class="flp">
        <p>
            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.<br><br>


In practice, effective retail inventory management results in lower costs and
a better understanding of sales patterns. Retail inventory management tools
and methods give retailers more information on which to run their businesses.
Applications have been developed to help retailers track and manage stocks
related to their own products. The System will ask retailers to create their
accounts by providing essential details. Retailers can access their accounts
by logging into the application.<br><br>


Once retailers successfully log in to the application they can update their
inventory details, also users will be able to add new stock by submitting
essential details related to the stock. They can view details of the current
inventory. The System will automatically send an email alert to the retailers
if there is no stock found in their accounts.  So that they can order new
stock.<br><br>
        </p></div>
        <script>
            window.watsonAssistantChatOptions = {
              integrationID: "1f065f6e-3057-489f-a846-2a97db7d5633", // The ID
of this integration.
              region: "au-syd", // The region your integration is hosted in.
              serviceInstanceID: "84d5b9e5-b4d1-498a-9596-14b250464018", //
The ID of your service instance.
              onLoad: function(instance) { instance.render(); }
            };
```

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

Products

```
<!DOCTYPE html>
<html>
<head>
<style>
table {
  font-family: arial, sans-serif;
  border-collapse: collapse;
  width: 100%;
}

td, th {
  border: 1px solid #dddddd;
  text-align: left;
  padding: 8px;
}

tr:nth-child(even) {
  background-color: #dddddd;
}
</style>
</head>
<body>
  <center>  <h2>PRODUCTS</h2></center>

  <table>
    <tr>
      <th>S.NO</th>
      <th>PRODUCT CODE</th>
      <th>PRODUCT TITLE</th>
      <th>TOTAL STOCK</th>
      <th>COST PER ITEM</th>
```

```
      <th>TOTAL COST</th>
      </tr>
    <tr>
      <td>01</td>
      <td>551</td>
      <td>wood</td>
      <td>65</td>
      <td>₹10000</td>
      <td>₹650000</td>
      </tr>
    <tr>
      <td>02</td>
      <td>552</td>
      <td>metals</td>
      <td>12</td>
      <td>₹5000</td>
      <td>₹60000</td>
    </tr><tr>
        <td>03</td>
        <td>553</td>
        <td>leather</td>
        <td>20</td>
        <td>₹15000</td>
        <td>₹300000</td>
        </tr>
    </table></body></html>
```

Addproduct

```
<html>
    <head>
        <title>add product</title>
        <link rel="stylesheet" href="co.css">
    </head>
    <body>
        <div class="container">
            <form id="contact"><center>
        <h1>ADD PRODUCTS</h1>
        <div class="column">
            <fieldset>
                <input type="text" placeholder="product code *" name="product
code" >
            </fieldset>
            <fieldset>
                <input type="text" placeholder="product title *" name="product
title">
            </fieldset>
```

```
            <fieldset>
                <input type="text" placeholder="total stock *" name="total
stock" >
            </fieldset>


            <!-- adding all country code list -->
            <fieldset>
                <input type="text" placeholder="cost per item *" name="cost">
            </fieldset>
        </div>
        <!-- second column -->
        <div class="column">


            <fieldset>
                <input type="text" placeholder="total cost *" name="total
cost"  >
            </fieldset>
            <fieldset>
                <input type="text" placeholder="supplier id *" name="supplier
id">
            </fieldset>

            <fieldset>
                <input type="text" placeholder="product description *"
name="subject"></fieldset>



        </div>
        <button type="submit"  id="button">ADD&SAVE</button><br>
        <button type="submit"  id="button">CANCEL</button>

    </div></center></div>
</form>
    </body>
</html>
```

Salesreport

```
<!DOCTYPE html>
<html>
<head>
<style>
table {
  font-family: arial, sans-serif;
```

```
  border-collapse: collapse;
  width: 100%;
}

td, th {
  border: 1px solid #dddddd;
  text-align: left;
  padding: 8px;
}

tr:nth-child(even) {
  background-color: #dddddd;
}
</style>
</head>
<body>
    <center><h2>SALES REPORT</h2></center>

  <table>
    <tr>
      <th>S.NO</th>
      <th>PRODUCT CODE</th>
      <th>PRODUCT TITLE</th>
      <th>SALES ITMS</th>
      <th>DISCOUNT</th>
      <th>TOTAL COST</th>
      </tr>
    <tr>
      <td>01</td>
      <td>551</td>
      <td>wood</td>
      <td>10</td>
      <td>3%</td>
      <td>₹100000</td>
      </tr>
    <tr>
      <td>02</td>
      <td>552</td>
      <td>metals</td>
      <td>8</td>
      <td>0%</td>
      <td>₹40000</td>
    </tr><tr>
        <td>03</td>
        <td>553</td>
        <td>leather</td>
        <td>5</td>
        <td>2%</td>
```

```
        <td>₹75000</td>
        </tr>
    </table></body></html>
```

Suppliers

```
<!DOCTYPE html>
<html>
<head>
<style>
table {
  font-family: arial, sans-serif;
  border-collapse: collapse;
  width: 100%;
}

td, th {
  border: 1px solid #dddddd;
  text-align: left;
  padding: 8px;
}

tr:nth-child(even) {
  background-color: #dddddd;
}
</style>
</head>
<body>
    <center><h2>SUPPLIERS</h2></center>

  <table>
    <tr>
      <th>S.NO</th>
      <th>SUPPLIER ID</th>
      <th>SUPPLIER NAME</th>
      <th>PHONE NO</th>
      <th>EMAIL ADDRESS</th>
      <th>PRODUCT CODE</th>
      </tr>
    <tr>
      <td>01</td>
      <td>101</td>
      <td>Devi</td>
      <td>6575859510</td>
      <td>devi@gmail.com</td>
      <td>551</td>
      </tr>
```

```
    <tr>
      <td>02</td>
      <td>102</td>
      <td>Solaimathi</td>
      <td>3434242556</td>
      <td>solaimathi@gmail.com</td>
      <td>552</td>
    </tr><tr>
        <td>03</td>
        <td>103</td>
        <td>Vignesh</td>
        <td>8767543209</td>
        <td>vignesh@gmail.com</td>
        <td>553</td>
        </tr>
    </table></body></html>
```

Addsuppliers

```
<html>
    <head>
        <title>add product</title>
        <link rel="stylesheet" href="co.css">
    </head>
    <body>
        <div class="container">
            <form id="contact"><center>
        <h1>ADD SUPPLIERS</h1>
        <div class="column">
            <fieldset>
                <input type="text" placeholder="supplier id *" name="product
code" >
            </fieldset>
            <fieldset>
                <input type="text" placeholder="supplier name *" name="product
title">
            </fieldset>
            <fieldset>
                <input type="text" placeholder="phone no *" name="total stock"
>
            </fieldset>


            <!-- adding all country code list -->
            <fieldset>
                <input type="email" placeholder="email *" name="cost">
            </fieldset>
```

```
        </div>
        <!-- second column -->
        <div class="column">


            <fieldset>
                <input type="text" placeholder="product code *" name="total
cost"  >
            </fieldset>

        <button type="submit"  id="button">ADD&SAVE</button><br>
        <button type="submit"  id="button">CANCEL</button>

    </div></center></div>
</form>
    </body>
</html>
```

GITHUB & PROJECT DEMO LINK

IBM-EPBL/**IBM-Project-37029-1660299701**

https://youtu.be/94Nggaoc_SY