NAME: SANJAYRAM A M

ROLL NUM:7376221EC293

SEAT NO: 421

PROJECT TITLE: INVENTORY SYSTEM

INVENTORY SYSTEM

DISCRIPTION:

Real-time stock availability updates empower users with immediate insights into inventory levels, that will help in quick decision making. The comprehensive reporting system generates consolidated, yearly, monthly, weekly, and daily reports, providing invaluable analytics for strategic planning. Timely reminders keep users informed about low stock levels and pending orders, enhancing proactive inventory control. Integration with messaging services enables effortless sharing of reports and reminders via WhatsApp or other messaging platforms, ensuring seamless communication with web users. With a user-friendly interfacing environment helps to reduce the stress to the user and saves a considerable amount of money and helps in business development.

PROBLEM STATEMENT:

Inefficient inventory management processes hinder business operations, leading to stockouts, excess inventory, and missed sales opportunities. Existing systems lack real-time visibility into inventory levels, making it challenging to make informed decisions regarding purchases, sales, and overall stock control. Manual entry processes are time-consuming and always leads to errors, resulting in discrepancies and inefficiencies. Additionally, there's a lack of timely notifications for low stock levels and pending orders, causing disruptions in supply chain management. These challenges underscore the need for a comprehensive inventory management solution that offers seamless purchase and sales tracking, real-time stock availability updates, comprehensive reporting capabilities, and proactive reminders to optimize inventory control and streamline operations.

TECHNICAL COMPONENTS INVOLVED

COMPONENTS	TECHSTACK
FRONT END	React (JS Library for building user interfaces)
BACK END	Node.js with Express.js
DATABASE	MongoDB(NOSQL Database)
API	OpenAPI

PROJECT INFO

1. Need of this web application

inventory management application provides real-time tracking of stock levels, generates detailed reports for informed decision-making, and sends proactive reminders to prevent stockouts and streamline operations, enhancing efficiency and profitability for businesses.

1.1. Scope

We can use this application for various application. It is not limited to a particular job i.e. if this application is used in a super market, it can keep track of various good's entry and as well as its sales. Like wise we can use it for medical shops, electronics items, and even a large warehouse where it reduces time and labour cost

2. Rescissory requirement for the software

➢ **Purchase Entry Portal**

The Purchase Entry Portal allows users to input details of new inventory purchases, including product information, quantity, and supplier details. Through an intuitive interface built with React, users can efficiently record purchases, triggering updates to the MongoDB database

➢ **Sales Entry Portal**

The Sales Entry Portal facilitates the recording of sales transactions with ease, enabling users to input customer details, product information, and quantities sold. Node.js with Express.js handles backend logic to process sales data and update MongoDB in real-time

➢ **Availability of Stocks**

The stock availability feature provides real-time visibility into inventory levels, allowing users to quickly check the quantity of each product in stock. Utilizing React for the frontend interface, Node.js with Express.js manages backend operations to retrieve stock data from MongoDB.

➢ **Reporting System**

The reporting system generates comprehensive reports on inventory performance, offering insights into sales trends, stock levels, and product movement. It provides a weekly, monthly, or yearly

If it was a bio-based food product and it was not sold, the web application will provide notification for its disposal.

➢ **Get Remainders about the status**

The reminder feature sends proactive notifications about critical inventory statuses, such as low stock levels or pending orders. Inventory thresholds and trigger reminders, ensuring timely action to maintain optimal stock levels and streamline operations.

It provides remainders for the disposal of the expired products based on the manufacturing date.

➢ **Notification**

It dispatches messages to the user's WhatsApp or to his mobile phone and even send messages to the goods dealer about the need of re-stock

WORKFLOW

1. **Start Process**: Initiate inventory management.	End Process: Conclude the inventory management workflow. (MongoDB)
Receive Purchase Order: Receive and process new purchase orders (React)	**Send Notifications**: Send alerts for low stock levels or pending orders. (API)
Update Inventory with New Stock: Add purchased items to the inventory (MongoDB)	**Generate Reports**: Create reports on inventory status and sales data. (API)
Process Sales: Handle sales transactions. (React)	**Adjust Inventory after Sales**: Update inventory levels post-sales.

APPLICATION WORK FLOW

```
                                              data base mongoDB
              user Login



not approved  no     authentication        yes

              yes

              home page                              each and every
                                                     information is taken
                                                     from databse

Purchase Entry  Purchase Entry   Purchase Entry   Purchase Entry
Portal          Portal           Portal           Portal



              gathering all the
              information to
              homepage
                              suggestion box
                              about website

              notification        to the user
```