# Inventory Management System

Introduction:

In the rapidly evolving landscape of technology, computers have become indispensable, influencing every facet of our daily lives. However, the conventional manual systems employed in various sectors face inherent challenges, particularly in the domain of task and inventory management. These challenges range from the cumbersome process of data manipulation to concerns about security, data integrity, and overall system reliability. Recognizing the imperative for a more streamlined and secure approach to task and inventory handling, our project endeavours to introduce an Integrated Inventory Management System that not only mitigates these challenges but also sets a new standard for efficiency and data integrity.

The limitations of manual systems are evident in their inability to handle the complexities of modern data management, such as insertion, deletion, and modification, while ensuring data consistency and security. Moreover, the challenges extend to inventory management, where tracking stock levels, managing orders, and ensuring timely replenishment become increasingly complex with manual methods.

Our project is driven by the need to transcend these limitations and provide a comprehensive solution to address the intricate demands of task and inventory management in a technologically advanced era. By integrating inventory management into a single, cohesive system, we aim to enhance overall operational efficiency and provide a holistic solution to the challenges faced by organizations.

The integrated system will offer features such as real-time task tracking, automated inventory updates, order management, and secure data handling. Through the implementation of advanced technologies, including data encryption and access controls, we strive to ensure the security and integrity of both task and inventory data.

This initiative represents a paradigm shift in how organizations approach their operational processes, moving away from siloed systems to a more interconnected and efficient solution. Our goal is to empower businesses to navigate the complexities of modern operations seamlessly, ultimately contributing to increased productivity, reduced errors, and enhanced decision-making capabilities.

Objectives:

The primary objective of our Inventory Management System is to revolutionize the way inventory is handled by harnessing the capabilities of a web-based platform. This project is designed to achieve the following key objectives:

## User-Centric Inventory Management:

## Efficient Inventory Tracking:

## Empower users to effortlessly manage inventory through an intuitive web interface. Incorporate recent improvements to enhance user-friendliness and operational efficiency.

## Real-Time Inventory Optimization:

## Implement features for real-time tracking of inventory levels, ensuring timely replenishment. Provide alerts for low stock levels, preventing stockouts and optimizing inventory management.

## Enhanced User Experience:

Easy to grasp the information for novice users.

Adjust colour schemes to ensure optimal readability and visual appeal, especially in low-light environments.

Incorporate a toggle button or setting to activate Dark Mode within the application.

## Optimized Resource Utilization:

Integrate features to enhance time management and productivity within the inventory management workflow.

By focusing on user-centric inventory management and incorporating features such as real-time optimization, movement tracking, and purchase order management, our project aims to provide organizations with a comprehensive solution that addresses the intricacies of modern inventory operations. This approach sets a new standard for efficiency, transparency, and user satisfaction in the realm of Inventory Management Systems, such as:

*i)Pomodoro Integration:* Implement a Pomodoro Timer within the system to facilitate time management and productivity.

## Allow users to start, reset, pause, and resume Pomodoro sessions based on their workflow.

## Encourage a balanced work-rest cycle to improve productivity and user satisfaction.

## *ii)Dark Mode Implementation:*

## Incorporate a toggle button or setting to activate Dark Mode within the application.

## Adjust colour schemes, ensuring readability and visual appeal in low-light environments.

# Project Category:

Enterprise Resource Planning (ERP) Systems

The project falls under the category of "Enterprise Resource Planning (ERP) Systems" with a specific focus on "Inventory Management.

Front-end Technologies:

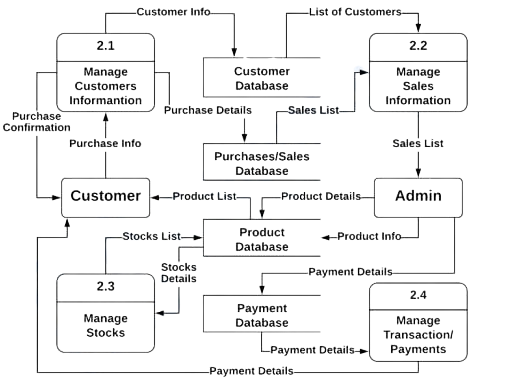
HTML/CSS/JavaScript: Fundamental technologies for creating the structure, styling, logic building and interactivity of the system.

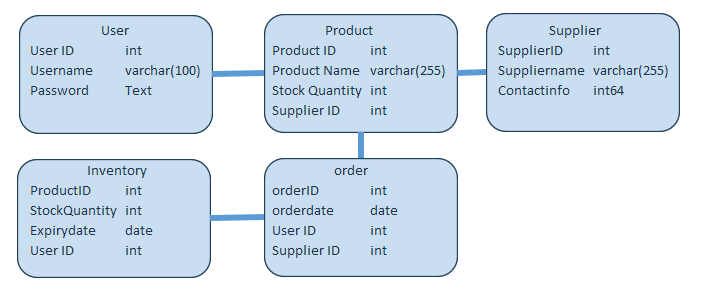
Back-end:

Nodejs: The project integrates Node.js as the backend server to handle business logic and communication between the client and MySQL database.

Database:

This project employs MySQL as the backend database to store and manage data efficiently. MySQL, a widely used relational database management system, facilitates structured data storage and retrieval. Leveraging its robust features, the backend seamlessly handles data transactions, ensuring reliability and scalability. Through SQL queries and transactions, the system interacts with MySQL to perform tasks such as data insertion, retrieval, and updates. This choice of backend technology aims to provide a secure, scalable, and well-organized foundation for the project's data management needs.

Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
Ink Drawings
USER
*User ID (PK)
Username
Password
Product 
*Product ID(PK)
Product Name
Stock Quantity
Company
**Supplier ID
Supplier 
**SupplierID(PK)
Suppliername
Contactinfo
Inventory
*Product ID
StockQuantity
Experydate
User ID
Ink Drawings
Order
orderIDorderdate
User ID
Supplier ID
Ink Drawings
Ink Drawings
DFD( Data Flow Diagram) and ER Entity Relationship Diagram)



**Number of Modules & Their Description: -**

The application consists of number of modules and sub modules, of which, the most important are discussed briefly…

The system consists of four modules. Those are…

1. User management module
2. Product management module
3. Order processing module
4. Supplier management module

1.User Management Module:

1.1 Description:

The User Management Module is responsible for handling user authentication, registration, and profile management. It ensures a secure and user-friendly experience by managing user credentials and profiles within the system.

1.2 Data Structures:

User Table:

userID (Primary Key)

username (Unique, Not Null)

password (Not Null)

2. Product Management Module:

2.1 Description:

The Product Management Module plays a central role in managing products throughout their lifecycle. It facilitates the addition, deletion, and modification of products, ensuring accurate and up-to-date information.

2.2 Data Structures:

Product Table:

productID (Primary Key)

productName (Not Null)

stockQuantity (Not Null)

3. Order Processing Module:

3.1 Description:

The Order Processing Module handles the entire lifecycle of customer orders. It ensures a seamless process from order creation to fulfillment, allowing for effective tracking of order status.

3.2 Data Structures:

Order Table:

orderID (Primary Key)

orderDate (Not Null)

userID (Foreign Key referencing User.userID)

OrderDetails Table:

orderID (Foreign Key referencing Order.orderID)

productID (Foreign Key referencing Product.productID)

quantity (Not Null)

(Primary Key: orderID, productID)

4. Supplier Management Module:

4.1 Description:

The Supplier Management Module focuses on maintaining crucial information about product suppliers. It ensures effective communication and collaboration with suppliers for a streamlined inventory process.

4.2 Data Structures:

Supplier Table:

supplierID (Primary Key)

supplierName (Not Null)

contactPerson

User Management Module:

1.1 User Table:

sql

Copy code

CREATE TABLE User (

userID INT PRIMARY KEY,

username VARCHAR(255) UNIQUE NOT NULL,

password VARCHAR(255) NOT NULL

);

1.2 Session Table:

sql

Copy code

CREATE TABLE Session (

sessionID INT PRIMARY KEY,

userID INT REFERENCES User(userID),

loginTime TIMESTAMP,

logoutTime TIMESTAMP

);

2. Product Management Module:

2.1 Product Table:

sql

Copy code

CREATE TABLE Product (

productID INT PRIMARY KEY,

productName VARCHAR(255) NOT NULL,

stockQuantity INT NOT NULL

);

2.2 Supplier Table:

sql

Copy code

CREATE TABLE Supplier (

supplierID INT PRIMARY KEY,

supplierName VARCHAR(255) NOT NULL,

contactPerson VARCHAR(255)

);

3. Order Processing Module:

3.1 Order Table:

sql

Copy code

CREATE TABLE Order (

orderID INT PRIMARY KEY,

orderDate DATE NOT NULL,

userID INT REFERENCES User(userID)

);

3.2 OrderDetails Table:

sql

Copy code

CREATE TABLE OrderDetails (

orderID INT REFERENCES Order(orderID),

productID INT REFERENCES Product(productID),

quantity INT NOT NULL,

PRIMARY KEY (orderID, productID)

);

4. Supplier Management Module:

4.1 Supplier Table:

sql

Copy code

CREATE TABLE Supplier (

supplierID INT PRIMARY KEY,

supplierName VARCHAR(255) NOT NULL,

contactPerson VARCHAR(255)

);

5. Inventory Tracking Module:

5.1 Inventory Table:

sql

Copy code

CREATE TABLE Inventory (

inventoryID INT PRIMARY KEY,

productID INT REFERENCES Product(productID),

stockQuantity INT NOT NULL,

movementType VARCHAR(50),

date

);

# Process Logic for each module:

## User Management Module:

1.1 User Authentication, Registration, and Profile Management:

User Authentication:

Verify user credentials against the database.

Grant access upon successful authentication.

Log authentication events in the session table.

User Registration:

Collect user information.

Check for username uniqueness.

Insert new user data into the User table.

Profile Management:

Allow users to update their profile information.

Ensure data consistency in the User table.

## Product Management Module:

Product Addition, Deletion, and Modification:

Product Addition:

Collect product information.

Generate a unique product ID.

Insert new product data into the Product table.

Product Deletion:

Identify the product to be deleted.

Remove corresponding records from the Product table.

Product Modification:

Retrieve and display existing product information.

Allow users to update product details.

Update records in the Product table.

## Order Processing Module:

Order Creation and Status Tracking:

Order Creation:

Collect order details, including product IDs and quantities.

Generate a unique order ID.

Insert new order data into the Order table.

Status Tracking:

Provide real-time status updates for orders.

Track order fulfillment and delivery status.

Log status changes in the Order table.

## Supplier Management Module:

Supplier Information Management:

Supplier Addition:

Collect supplier information.

Generate a unique supplier ID.

Insert new supplier data into the Supplier table.

Supplier Modification:

Retrieve and display existing supplier information.

Allow users to update supplier details.

Update records in the Supplier table.

Inventory Tracking Module:

Inventory Management:

Inventory Update:

Track product movements (e.g., additions, deletions).

Update stock quantities in the Inventory table.

Log movement details, including type and date.

## Reporting Module:

Generate Reports:

Define queries or views to extract relevant data.

Generate various reports, such as sales reports, inventory reports, etc.

Present reports based on user requests.

7. Security Module:

Implement Security Measures:

Apply encryption algorithms for sensitive data.

Manage user access control based on roles and permissions.

Log security-related events for auditing.

Integration Module:

Data Flow Between Modules:

Establish APIs or data transfer mechanisms.

Ensure seamless communication and data synchronization between modules.

User Interface (UI) Module:

UI Interaction:

Develop user-friendly interfaces for each module.

Handle user inputs and interactions.

Communicate with the backend for data retrieval and updates.

## Testing and Quality Assurance Module:

Define Test Cases:

Specify test scenarios for each module.

Include positive and negative test cases.

Execute Tests:

Conduct testing to validate module functionalities.

Address and resolve any identified issues.

# Tools/ Platforms

# Hardware Requirement Specification

## Hardware Requirements

## Minimum:

## OS: Microsoft® Windows® 2000/XP,7,8,8.1,10,11 or Any Linux Distribution.

## Processor: 1 GHz Intel Pentium III or AMD Athlon processor or equivalent

## Memory: 2 GB RAM

## DirectX Version: Microsoft DirectX® 8.1

## Hard Drive: 2.3 GB uncompressed free hard disk space.

## Recommended:

## OS: Microsoft® Windows® 2000/XP,7,8,8.1,10,11 or Any Linux Distribution.

## Processor: 2GHz Intel Pentium 4 or AMD Athlon XP 2000+ processor (or better) or equivalent

## Memory: 2GB RAM

## Sound Card: DirectX 8.1 compatible sound card with hardware positional sound.

## Hard Drive: 2.3 GB uncompressed free hard disk space.

# Software Requirement Specification

Google Chrome, Mozilla Firefox, Microsoft Edge, or Safari.

Description: Use a modern web browser for testing and debugging your client-side code. Chrome Developer Tools, Firefox Developer Tools, and other browser-specific tools will be valuable for debugging.

Recommendation: MySQL

Description: MySQL is a popular relational database management system. It integrates seamlessly with Node.js using the MYSQL package for interacting with the database.

Database GUI (Optional):

Recommendation: MySQL Workbench 8.0 CE, DBeaver

Description: A database GUI can be useful for visually managing and interacting with your MySQL database during development.

Security:

Follow best practices for securing your Node.js server and MySQL database. Use environment variables for sensitive information and employ secure coding practices.

# Defining Challenges, Specifications and Project Planning

# Challenges Identification:

Data Inaccuracy:

Incorrect data entry or manual errors may result in inaccurate inventory levels, leading to issues such as overstocking or stockouts.

Integration Issues:

## Challenges may arise when integrating the IMS with existing systems like accounting or order processing, causing data synchronization problems.

Technological Obsolescence:

## Rapid technological advancements can render the current IMS obsolete over time, necessitating frequent updates to keep pace with industry standards.

Supply Chain Disruptions:

External factors, such as natural disasters, political instability, or global crises, can disrupt the supply chain, impacting inventory levels and distribution.

Costs and Budget Constraints:

## Implementing and maintaining an IMS involves costs related to software, hardware, and ongoing support, posing challenges for organizations with budget constraints.

Scalability Challenges:

As businesses grow, the inventory management system must be scalable to accommodate increased data volume, transaction complexity, and user requirements.

# Requirements Specifications:

## Functional Requirements:

## Inventory Management:

Users can perform CRUD operations (Create, Read, Update, Delete) for articles/inventory items.

Inventory can be categorized for better organization.

Inventory status (e.g., out of stock, expiry date reached) can be updated.

## User Authentication:

Implement secure user authentication to ensure data privacy.

Enable user account creation with password protection.

## Non-functional Requirements:

## User Interface:

## Design an intuitive and responsive interface for a seamless user experience.

## Ensure compatibility across various devices and browsers.

## Performance:

## Aim for quick loading times to facilitate efficient inventory management.

## Minimize latency during task updates.

## Security:

Implement data encryption to protect user information.

Conduct regular backups to prevent data loss.

## Resources:

## Development Tools:

## HTML, CSS, JavaScript for front-end.

## Backend framework (as applicable).

## Database for data storage.

## Testing:

## Manual testing for user interface and functionality.

## Automated testing for performance.

## Deployment:

## Web server for hosting.

## By adhering to this project plan, we aim to deliver a robust and user-friendly inventory management system that addresses the identified challenges concerning inventory/articles and enhances overall productivity.

## Analysis (DFDs, ER Diagrams/Class Diagrams, etc.)

## The analysis phase for the Task Management System revolves around critical factors such as the availability of decision-making information and realistic time constraints. The System Requirement Specification (SRS) was meticulously crafted through extensive discussions with stakeholders from the "OSCM." The Software Project Management process commences with comprehensive PROJECT PLANNING activities, covering estimation of financial resources, effort, human resources, and time required for system development.

## The phases Covered in this analysis include:

## Pre-Analysis Studies:

## Identify issues in the existing system.

## Conduct investigations to formulate effective solutions.

## System Analysis:

## Develop a Software Requirement Specification (SRS) to guide the system analysis process.

## System Design:

## Plan solutions for the identified problems outlined in the requirements document.

## Project Coding:

## Translate the system design into executable program code using tools like Visual Studio or Java.

## Project Testing:

## Conduct comprehensive testing to eliminate errors introduced during the coding phase.

## Implementation & Documentation:

## Execute activities to ensure the system's continued operation post-software installation.

## Document all aspects of the system to facilitate future understanding and maintenance.

## This structured approach ensures a thorough understanding of the task management system's requirements, leading to an effective design and implementation process. The emphasis on testing and documentation guarantees a reliable and sustainable solution for users and stakeholders alike.

## Future Scope:

## Integration with E-commerce Platforms:

## The system might explore integration possibilities with popular e-commerce platforms, facilitating seamless order processing and inventory management across different sales channels.

## Cloud Integration:

## Consideration might be given to migrating the system to a cloud-based infrastructure, offering increased scalability, flexibility, and accessibility.

## Localization and Multilingual Support:

## The system might incorporate support for multiple languages and localization, catering to users in different regions and enhancing its versatility.

## Enhanced User Experience (UX) Design:

## Continuous refinement of the user interface and experience might take place based on user feedback and evolving design trends, ensuring a modern and intuitive system.

## Supply Chain Visibility:

## The system might be extended to provide visibility into the entire supply chain, allowing users to track products from suppliers to customers.

## Chatbot Integration for User Assistance:

## The implementation of a chatbot might be explored to provide immediate assistance to users, addressing queries related to product availability, order status, and more.

## Mobile Application Development:

## A mobile application might be considered to enhance user accessibility, allowing them to manage inventory and track orders conveniently on their mobile devices.

## Advanced Reporting and Analytics:

## Advanced analytics tools might be incorporated to enrich reporting capabilities, offering insights into sales trends, product performance, and supplier interactions.

## Automated Reorder System:

## An automated reorder system might be implemented, leveraging historical data analysis to trigger reorder requests for products that are running low on stock.

## Enhanced User Roles and Permissions:

## User roles and permissions might be further refined to provide more granular control over access to various modules and functionalities.

## Real-time Collaboration Features:

## The system might introduce real-time collaboration features, allowing multiple users to work simultaneously on tasks like order processing and inventory updates.

## Machine Learning for Demand Forecasting:

## The exploration of machine learning algorithms might be considered to predict demand patterns, optimizing inventory levels and preventing stockouts This will require a major/complete overhaul for the system.