Software Requirement Specification Documentation

Mess Groceries Stock Maintenance Portal

Name	R JANA SRUTI
Roll no	7376221MC115
Seat no	377
Project ID	17
Problem Statement	Mess Groceries Stock Maintenance Portal

Technical Component: (MERN Stack)

Front End	React.js
Back End	Node.js
Database	MongoDB
API	OpenAPI

1. Introduction:

1.1 Purpose:

The purpose of the portal is to efficiently manage the stock of groceries. It helps track inventory levels, place orders for replenishment, manage suppliers, and generate reports for informed decision-making. The portal ensures that the right products are available in the right quantities, minimizing stockouts and

overstocking. It also provides insights into stock trends and supplier performance, facilitating effective inventory management.

1.2 Scope of Project:

The scope of the project includes designing and implementing a web-based portal for managing the stock of groceries. This involves developing features such as product management, supplier management, order management, user authentication and permissions, reporting and analytics, mobile compatibility, integration with POS systems. The scope also encompasses ensuring data security, user authentication, and integration with other systems if needed. The project will be developed using appropriate technologies and frameworks to meet the functional and performance requirements.

2. System Overview:

2.1 Users:

1. **Mess Managers:**

- Responsible for managing the groceries in their respective mess (girls mess, boys mess, day scholar mess).
- Can submit purchase requests for groceries needed in their mess.
- Have access to consumption reports specific to their mess.

2. **Purchase Managers:**

- Handle the procurement process, including approving purchase requests and creating purchase orders.
- Monitor stock levels and ensure timely procurement of groceries.
- Review all the reports like cost reports and analyse expenditure on groceries.

2.2 Features:

1. Item Management: Maintain a database of grocery items with details such as name, category, quantity, and expiry date.

2. Supplier Management: Manage a list of suppliers, including contact information and pricing details.

3. Purchase Request: Enable users to submit purchase requests for groceries, specifying the quantity and delivery date.

4. Approval Workflow: Implement a workflow for approving purchase requests, including notifications and tracking.

5. Purchase Order Generation: Automatically generate purchase orders based on approved requests.

6. Goods Receipt: Record the receipt of goods from suppliers and update the stock inventory accordingly.

7. Reporting and Analytics: Generate reports on stock levels, consumption patterns, and costs for analysis and decision-making.

3. Requirements:

3.1 Functional Requirement:

1. Item Management:

- Maintain a master list of grocery items with details such as name, category, unit of measure, reorder level, and expiry date.
- Allow users to add new items, edit existing items, and mark items as inactive or out of stock.

2. Supplier Management:

- Maintain a list of approved suppliers with contact information and terms.
- Link suppliers to specific items they provide.

3. Purchase Request Management:

- Enable users to create purchase requests for items needed in their mess.
- Route purchase requests for approval based on predefined workflows.

4. Approval Workflow:

- Define approval hierarchies and rules based on purchase request amount, type of item, etc.
- Notify approvers of pending requests and allow them to approve or reject requests.

5. Purchase Order Generation:

- Convert approved purchase requests into purchase orders.
- Include details such as item, quantity, price, and delivery date.

6. Goods Inward Note (GIN) Entry:

- Record receipt of goods from suppliers.
- Verify the quality and quantity of goods received.

7. Consumption Tracking:

- Record daily consumption of groceries in each mess.
- Generate consumption reports for analysis.

8. Expiry Product Management:

- Identify and flag items nearing expiration.
- Generate reports of expiring products for timely action.

9. Stock Maintenance:

- Monitor real-time stock levels.
- Conduct regular stock audits.

10. Reporting and Analytics:

- Generate various reports (e.g., consumption, stock, cost) with options for customization and filtering.

3.2 Non-functional Requirements:

1. Performance:

- System should be responsive and able to handle multiple concurrent users.
- Reports generation should be efficient, even with large datasets.

2. Reliability:

- System should be available 24/7 with minimal downtime for maintenance.
- Data integrity should be maintained through proper validation and error handling.

3. Scalability:

- System should be able to scale up to accommodate a growing number of users and items.

4. Usability:

- User interface should be intuitive and easy to use, even for non-technical users.

5. Security:

- Data should be stored securely with access controls and encryption.

- Regular security audits should be conducted to identify and mitigate vulnerabilities.

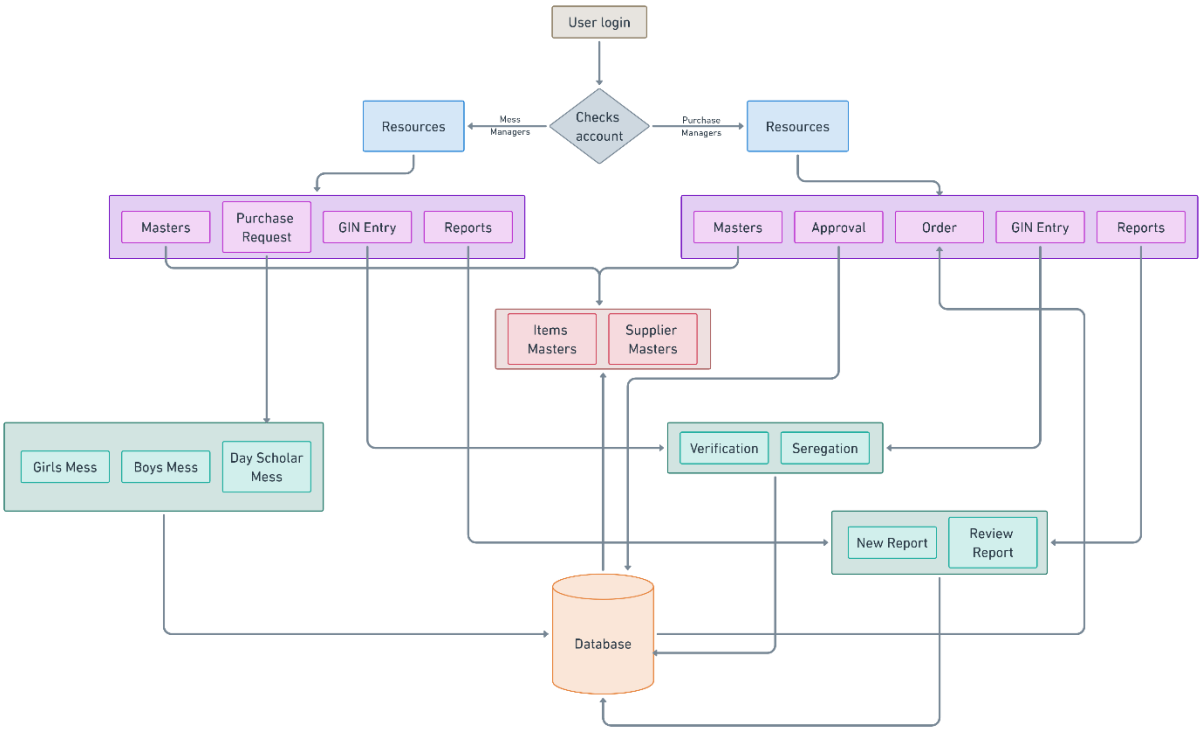
6. **Compliance:**

- System should comply with relevant data protection and privacy regulations (e.g., GDPR, HIPAA).

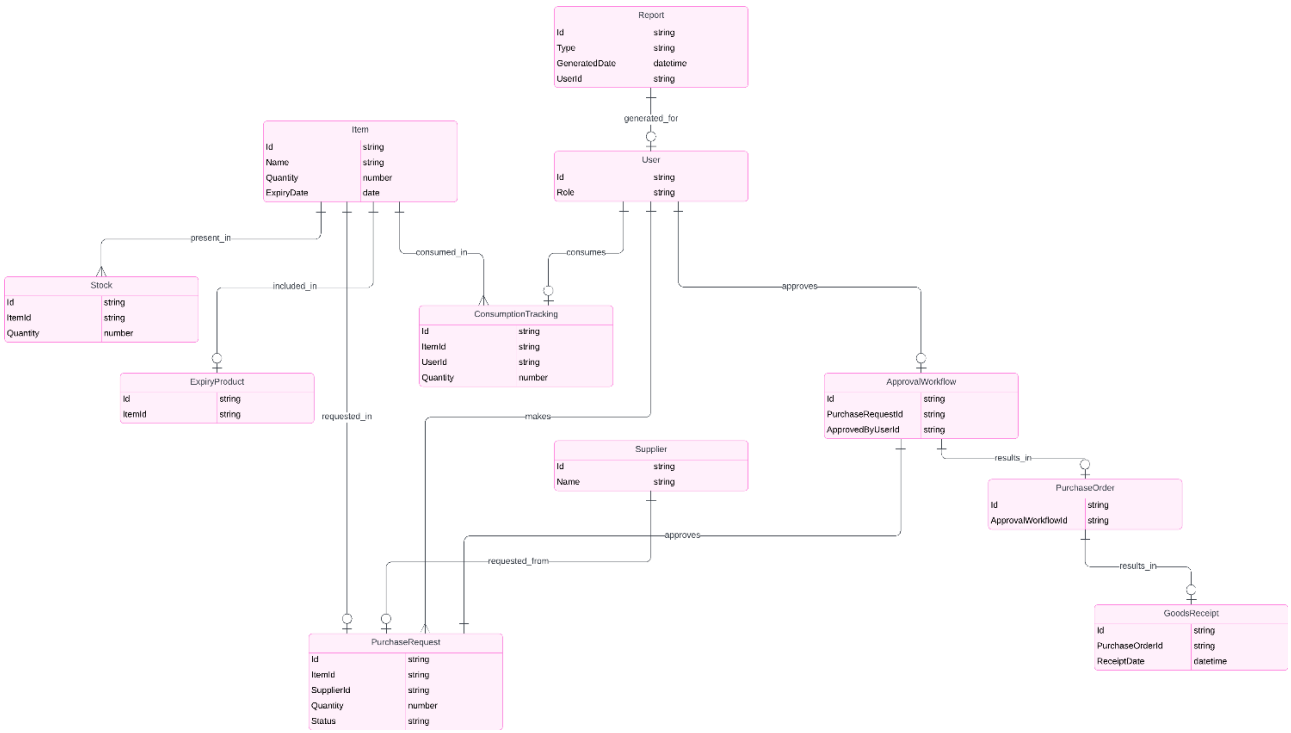
7. **Maintainability:**

- System should be easy to maintain and update with new features or changes in requirements

Flowchart:



ER Diagram:


Report
Id string
Type string
GeneratedDate datetime
UserId string
generated_for
Item
Id string
Name string
Quantity number
ExpiryDate date
User
Id string
Role string
present_in
consumed_in
consumes
approves
Stock
Id string
ItemId string
Quantity number
included_in
ConsumptionTracking
Id string
ItemId string
UserId string
Quantity number
ExpiryProduct
Id string
ItemId string
requested_in
makes
ApprovalWorkflow
Id string
PurchaseRequestId string
ApprovedByUserId string
Supplier
Id string
Name string
results_in
PurchaseOrder
Id string
ApprovalWorkflowId string
approves
requested_from
results_in
GoodsReceipt
Id string
PurchaseOrderId string
ReceiptDate datetime
PurchaseRequest
Id string
ItemId string
SupplierId string
Quantity number
Status string