Order Processing,

Store Management,

Bill Issuing,

Customer Management.

"MOS Burgers" shop offers a variety of food items and beverages, each listed with corresponding prices in annex 01

The system users should have the capability to add new items, manage the existing item list, update price lists, and receive notifications regarding item expiration dates.

User should be able to view all food items. All food items within the system should be presented to the user categorically.The Comprehensive list of food items and their respective categories is provided in the annex at the end of the document. When displaying the food items, the list should include the price, quantity on hand, expiration dates for packaged foods, and the unique food item code. • User should be able to update and delete food items. • User should be able to add items to the system. • System should notify user about the expired food items and should be able to remove those food items marked as expired.

**Milestone 01: Project Planning and Analysis Tasks and Submissions:**

1. Develop a use case diagram depicting interactions between actors and the system.

2. Develop wireframes for the web application to outline basic structure and layout.

3. Finalize the prototype design using [Figma/Adobe XD].

4. Analyze detailed requirements, including functional and non-functional aspects.

5. Document additional research findings on best practices for web development.

To get started on your project for "MOS Burgers," let's break down each of the tasks in Milestone 01:

**1. Develop a Use Case Diagram**

The use case diagram will depict the interactions between actors (users) and the system. The actors typically include the customer, store manager, and possibly staff members.

**Actors:**

* Customer
* Store Manager
* Staff

**Use Cases:**

* View Food Items
* Add/Update/Delete Food Items
* Place Order
* View Order Details
* Generate Receipt
* Add/Update/Delete Customer Details
* Generate Reports

**Example Use Case Diagram:**

**2. Develop Wireframes**

Wireframes provide a visual guide to the skeletal framework of the web application. Here are some essential wireframes:

**Wireframes:**

1. **Home Page:**
   * Navigation bar (Home, Menu, Orders, Customers, Reports)
   * Quick access links to popular features
2. **Menu Management Page:**
   * List of food items with details (name, price, quantity, expiration date)
   * Buttons to add, update, delete items
3. **Order Management Page:**
   * Search bar to find items by name, code, or category
   * Cart display with item details (price, quantity, discount)
   * Order summary and place order button
4. **Customer Management Page:**
   * List of customers with details (name, contact, order history)
   * Buttons to add, update, delete customers
5. **Report Generation Page:**
   * Options to generate monthly, annual reports
   * List of top customers and sales statistics

**3. Finalize Prototype Design**

Using design tools like Figma or Adobe XD, create a high-fidelity prototype based on the wireframes. The prototype should include all the necessary UI elements and interactions.

**Steps to create a prototype:**

1. Open Figma or Adobe XD.
2. Create frames for each page based on the wireframes.
3. Design UI elements (buttons, forms, tables).
4. Link pages to simulate navigation and interactions.
5. Share the prototype for feedback.

**4. Analyze Detailed Requirements**

**Functional Requirements:**

* Ability to manage food items (CRUD operations).
* Order processing including discounts and receipts.
* Customer management with order history.
* Report generation for sales and customer data.

**Non-Functional Requirements:**

* User-friendly interface.
* Responsive design for various devices.
* Secure handling of customer data.
* Efficient and scalable to handle growing number of transactions.

**5. Document Additional Research Findings**

Research best practices for web development, focusing on technologies and methodologies that will ensure the system is robust and scalable.

**Research Findings:**

* **Technology Stack:** Consider using modern frameworks like React.js for the frontend, Node.js for the backend, and MongoDB for the database.
* **Security:** Implement authentication and authorization, secure data transmission using HTTPS, and protect against common web vulnerabilities (e.g., SQL injection, XSS).
* **Performance:** Optimize queries, use caching strategies, and ensure efficient database indexing.
* **Usability:** Follow UI/UX best practices for navigation, form design, and accessibility.

**Example Use Case: Place Order**

1. **Actor:** Customer
2. **Precondition:** Customer is logged in.
3. **Main Flow:**
   * Customer selects items from the menu.
   * System displays item details and adds them to the cart.
   * Customer reviews the cart and applies discount if any.
   * Customer confirms the order.
   * System generates and displays a receipt.
4. **Postcondition:** Order is saved, and receipt is generated.

**Tools and Resources**

* **Design Tools:** Figma, Adobe XD
* **Development Tools:** VS Code, GitHub, Node.js, Express.js, MongoDB
* **Libraries and Frameworks:** React.js, Bootstrap
* **Security:** OWASP guidelines, HTTPS

By following these steps and guidelines, you'll be well on your way to developing a comprehensive and efficient digital system for "MOS Burgers."