Burger Shop Customer Management System

Requirement Specification Document

MOS Burgers

2024/06/30

Dilshani Prabodha

1. Introduction

- Project Overview : Develop a system for managing customer information.

- Objectives : Enable the cashier to add customer information and the owner to manage (add, update, delete, display) customer details.

2. Functional Requirements

User Roles

- *Cashier:*
  * Add, update, delete, display customer information.
  * Add, update, delete, display food item details.
  * Add, update, delete, display order details.

- *Owner:*
  * Add, update, delete, display customer information.
  * Add, update, delete, display food item details.
  * Add, update, delete, display order details.
  * View monthly sales report, beat customer reports, annual sales report and items count report.

Store Management

- View Items
- Add Items
- Update Items
- Expirable Items

Order Management

- Place Order
- View Order
- Update Order
- Delete Order

Customer Management

- Add Customer
- View Customer
- Update Customer
- Delete Customer
- Display Customer
- Customer History

Report Management

- Monthly Sales Report
- Annual Sales Report
- Item Count Report
- Best Customer Report

3. Non-functional Requirements

- Performance : Quick data processing.
- Usability : Intuitive interface for both roles.
- Security : Secure access control.
- Scalability : Handle increasing data volume.
- Availability : Available during shop hours.

4. Priorities

- High Priority : User Authentication, Customer Information Management (Add, Update, Delete, Display).

- Medium Priority: User Role Management.

- Low Priority : Reporting and Analytics.

5. Dependencies

- **Authentication System** : Differentiate between cashier and owner roles.
- **Database** : Store customer data.
- **Technology stack** : HTML, CSS, JavaScript, MySQL.

6. Technical Solutions

- **Frontend** : HTML, CSS, JavaScript.
- **Backend** : Node.js with Express.js.
- **Database** : MySQL.

7. Timeline

- **June 20 – June 29** : Requirements gathering.

  Frontend planning using UML diagrams, wire frame and UI/UX design with Figma.

- **June 29 – July 14** : Frontend development.

- **July 14 – July 19** : Development interactivity and functionality to the frontend.

  Implement client side validations for user inputs.

- **July 19 – July 21** : Finalize the development of the project the web application.

  Launch the project.

8. Feasibility Assessment

- **Budget** :
- **Time** : Realistic timeline.
- **Skills** : HTML, CSS, JavaScript, MySQL.