

Requirements Report

Menu Mangment System

College of Computer Sciences & Information Technology

Kingdom of Saudi Arabia

Ministry of Higher Education

King Faisal University



Table of Contents

1. ***Introduction***
2. ***Objectives***
3. ***User Requirements Definition*** *3.1. Customer‑Facing Requirements  
    3.1.1. Menu Navigation  
    3.1.2. Order Modification  
    3.1.3. Customer Reviews  
   3.2. Staff‑Facing Requirements  
    3.2.1. Order Visibility for Waitstaff  
    3.2.2. Menu Management (Manager Access)*
4. ***System Architecture*** *4.1. Components  
    4.1.1. Client‑Side Application  
    4.1.2. Server‑Side Application  
    4.1.3. Database  
    4.1.4. APIs  
   4.2. Additional Integrations  
    4.2.1. Analytics & Reporting*
5. ***System Requirements Specification*** *5.1. Functional Requirements  
    5.1.1. Menu Navigation  
    5.1.2. Order Modification  
    5.1.3. Order Visibility for Waitstaff  
    5.1.4. Menu Management  
    5.1.5. Customer Reviews  
    5.1.6. Reporting & Analytics  
   5.2. Nonfunctional Requirements  
    5.2.1. Performance  
    5.2.2. Security  
    5.2.3. Scalability  
    5.2.4. Reliability  
    5.2.5. Usability  
    5.2.6. Maintainability*
6. ***Process Model***

# 1. Introduction

This document outlines the requirements for a new computer system designed to manage our restaurant's menu. Developed for a small team of about four individuals, the system is intended to simplify the process of adding, updating, or removing menu items. It will support item categorization (e.g., appetizers, main courses, desserts) and allow managers to include detailed, pricing, name, and more. The primary goal is to streamline menu management and order processing, enhancing the overall customer and staff experience.

# 2. Objectives

• Ease of Management: Enable managers to update and maintain the menu quickly and accurately.

• Enhanced Customer Experience: Provide an intuitive interface for customers to navigate the menu, customize orders, and leave feedback.

• Improved Service Efficiency: Equip waitstaff with real-time order details to ensure prompt and accurate service.

• Scalability & Reliability: Develop a system that can scale as the business grows while ensuring high performance and security.

# 3. User Requirements Definition

3.1. Customer-Facing Requirements

* Menu Navigation:
* Requirement: The system shall present a user-friendly interface where customers can browse the menu organized into clearly labeled sections (e.g., appetizers, mains, desserts).
* Improvement: Incorporate search and filtering capabilities to help customers quickly locate items by name, price, categorize.
* Order Modification:
* Requirement: Customers shall be able to easily add or remove items from their orders, adjust quantities, and review their selections before finalizing the order.
* Customer Reviews:
* Requirement: Customers shall be able to provide ratings and written reviews for menu items, which will be available for managers to review and respond to.
* Improvement: Include moderation tools for managing inappropriate content and analytics to gauge item popularity.

3.2. Staff-Facing Requirements

* Order Visibility for Waitstaff:
* Requirement: The system shall provide waitstaff with real-time access to current orders along with relevant customer details to support timely service.
* Menu Management (Manager Access):
* Requirement: Managers shall have a secure backend interface to update menu items, name, prices, and categorize.
* Improvement: Add version control or audit logs so managers can track changes over time.//

# 4. System Architecture

4.1. Components

* Client-Side Application:
* Provides a responsive, intuitive interface for both customers and staff (e.g., web and mobile versions).
* Includes features for menu browsing, order customization, and review submission.
* Server-Side Application:
* Handles core business logic, such as order processing, menu updates, and review management.
* Implements real-time updates (using technologies like WebSockets) to keep waitstaff informed.
* Database:
* A relational database that stores menu items, orders, customer data, reviews, and statistical data.
* Improvement: Incorporate backup and failover strategies to ensure data integrity and availability.
* APIs: // تحت مراجعة
* RESTful APIs will facilitate communication between the client-side interface, server-side application, and database.
* Improvement: Ensure APIs are well-documented and secured with appropriate authentication and rate-limiting measures.

4.2. Additional Integrations

* Analytics & Reporting:
* Generate reports on orders, popular items, and customer feedback for continuous improvement.

# 5. System Requirements Specification

5.1. Functional Requirements

1. Menu Navigation:

* Provide a user-friendly, searchable interface with clearly defined sections.
* Support filtering options by category, ingredient.

1. Order Modification:

* Allow customers to add, remove, and update order items.
* Display a dynamic order summary with pricing details before final confirmation.

1. Order Visibility for Waitstaff:

* Offer real-time updates of incoming orders along with essential customer details.
* Support notifications for new or modified orders.

1. Menu Management:

* Enable managers to update item details (name, price,categorize) via a secure admin panel.
* Include audit logs to track changes over time.//

1. Customer Reviews:

* Allow customers to submit ratings.

1. Reporting & Analytics (Additional):

* Generate statistical reports on sales that the manager can display total sales and profits along with the average and number of ratings.
* Display total sales and profit for each item.

5.2. Nonfunctional Requirements

1. Performance:

* The system should respond promptly to user interactions, even under multiple concurrent users.

1. Security:

* Data must be securely stored and transmitted (e.g., using encryption).
* Implement role-based access control to protect sensitive areas of the system.

1. Scalability:

* Design the system to easily accommodate increases in menu items, orders, and user volume.

1. Reliability:

* Ensure high availability with regular backups, fault tolerance, and disaster recovery plans.

1. Usability:

* The interface should be intuitive, require minimal training, and be accessible on multiple devices.
* Improvement: Adhere to accessibility standards (such as WCAG) to ensure usability for all users.

1. Maintainability:

* The system should be modular to allow for easy updates and enhancements.
* Improvement: Include comprehensive documentation for developers and administrators.

# 7. Process Model //

Our team will embark on a journey to create a fantastic system for managing our restaurant's menu. Imagine a group enthusiastic individuals working together to make menu management a breeze for everyone. Here's a simplified breakdown of our approach:

