
Software Requirements Specification

for

RESTAURANT APP

Version 1.0 approved

Prepared by

Thakur College of Engineering & Technology

February 3, 2025



Table of Contents

Table of Contents	ii
Revision History	ii
1. Introduction	1
1.1	Purpose
1.2 Document	1
1.3 Intended Audience and Reading	Conventions
1.4 Product	1
1.5	Suggestions
2. Overall Description	1
2.1 Product	Scope
2.2 Product	1
2.3 User Classes and	References
2.4 Operating Environment	1
2.5 Design and Implementation	2
2.6 User	Perspective
2.7 Assumptions and	2
3. External Interface Requirements	Functions
3.1 User	2
3.2 Hardware	Characteristics
3.3 Software	2
3.4 Communications	Environment
4. System Features	2
5. Other Nonfunctional Requirements	Constraints
5.1 Performance	3
5.2 Safety	Documentation
5.3 Security	3
	Dependencies
	3
	3
	Interfaces
	3
	Interfaces
	4
	Interfaces
	4
	Interfaces
	5
	5
	6
	Requirements
	6
	Requirements
	8
	Requirements
	9



5.4Software	Quality	Attributes
		9
5.5Business		Rules
		10
6. Other Requirements		11
Appendix A: Glossary		12
Appendix B: Analysis Models.....		12
Appendix C: To Be Determined List.....		12

Revision History

Name	Date	Reason For Changes	Version



1. Introduction

1.1 Purpose

The purpose of this SRS document is to provide a detailed description of the Restaurant Application for users. This document will outline the system's functionality, constraints, and requirements to ensure a seamless dining experience for customers.

1.2 Document Conventions

NA

1.3 Intended Audience and Reading Suggestions

This project is a prototype for the restaurant management system. This has been implemented under the guidance of college professors. This project is useful for the restaurant management team and as well as to the customers.

1.4 Product Scope

- A Restaurant App is a comprehensive online platform designed to enhance the dining experience for restaurant customers by offering features such as table booking, digital menus, order placement, and bill generation—all accessible within a single app.
- For customers, the app streamlines the dining process by enabling convenient seat reservations, contactless ordering, and instant bill payments directly from their table, eliminating the need for waiter assistance. This not only saves time but also provides a seamless, hassle-free experience.
- For restaurants, the app delivers multiple benefits. It reduces the need for one-to-one catering, minimizes crowding in dining areas, and ensures efficient order tracking. By automating tasks like menu updates and billing, it allows staff to focus on providing better service, ultimately boosting customer satisfaction and operational efficiency.

1.5 References

NA



2. Overall Description

2.1 Product Perspective

The Restaurant Application is a mobile and web-based platform that operates as an extension of the restaurant's RMS. It will interact with various components such as the booking system, order management system, billing system, and feedback system.

2.2 Product Functions

- User Management: Facilitates user account creation and login, enabling customers to manage their profiles seamlessly.
- Table Management: Dynamically manages table availability based on real-time vacancy and customer requirements. Tracks occupancy status, ensuring efficient utilization of seating arrangements.
- Booking Management: Handles customer booking data, including details such as table number, booking date, and time, ensuring smooth reservation processes.
- Menu Management: Maintains an up-to-date menu, including food items, pricing, and special offers, allowing for easy modifications and improved customer experience.
- Order Management: Tracks customer orders, including food items, associated prices, and order placement times, ensuring accurate and efficient service.
- Bill Generation: Automates bill generation for customers, providing a transparent and error-free invoicing experience.
- Feedback/Ratings Management
- Admin Management

2.3 User Classes and Characteristics

Users of the system should be able to retrieve table availability and manage orders based on the given date/time and party size.

The system will support two types of user privileges: Customer and Employee. Customers will have access to reservation and ordering functions, while Employees will have access to both customer-facing and restaurant management functions.



Customer Functions

The customer will have the ability to perform the following actions within the app:

1. Make a New Reservation
 - Choose a date/time and specify the party size.
 - Select a seating preference (e.g., indoor/outdoor).
 - Confirm the reservation.
2. Place an Order
 - View a digital menu.
 - Place food and drink orders directly from the table using the app.
 - Modify or cancel orders before they are prepared.

3. View Order Status

Track the status of the order (e.g., order received, being prepared, ready for serving).

4. Generate and Pay Bill
 - Automatically generate a bill based on the items ordered.
 - Make payment directly through the app, eliminating the need for manual bill processing.
5. Cancel or Modify an Existing Reservation
 - Customers can modify or cancel their reservations up to a certain point before their dining time.

Employee Functions

Employees (staff members) will have access to the following restaurant management functionalities:

1. Customer Order Management
 - View orders placed by customers at each table.
 - Modify or cancel orders on request, based on the current status.
2. Reservation Management
 - View and manage customer reservations, including party sizes and seating preferences.
 - Monitor table availability and assign tables accordingly.
3. Menu Management
 - Update and modify the digital menu, including pricing, descriptions, and availability of items.
4. Billing Management
 - Review and finalize bills for customers.
 - Assist in processing payments if needed.
5. Customer Interaction
 - View customer feedback and ratings.
 - Access customer preferences (e.g., dietary restrictions) for a more personalized dining experience.

Administrative Functions

Administrators (restaurant managers or owners) will have access to broader management functionalities:



1. Add/Delete Items from the Menu
 - Add new items to the menu or remove outdated ones.
2. Manage Reservation Times
 - Set or adjust restaurant hours, define time slots for reservations, and manage seating capacities.
3. Monitor Sales and Performance
 - View sales data, including total billings and popular menu items.
4. Manage Staff Schedules
 - Assign staff to tables, track shift schedules, and ensure proper coverage during peak hours.

2.4 Operating Environment

- Centralized Database
- Client/ server system
- OS: android / IOS.
- Database: SQL

2.5 Design and Implementation Constraints

1. **Platform Compatibility**
 - Must be compatible with both iOS and Android devices. A web version should also be available for desktop users.
2. **Network Connectivity**
 - Requires internet for real-time updates, but must function in offline mode for viewing menus and making future reservations.
3. **Data Privacy and Security**
 - Must comply with data protection regulations (e.g., GDPR, CCPA) and ensure secure payment transactions via encrypted connections.
4. **Scalability**
 - Must handle growing user traffic and support multiple restaurant locations without performance issues.
5. **Performance Requirements**
 - The app must load in 2-3 seconds and provide real-time order and reservation updates.
6. **User Interface (UI) Design**
 - Should be intuitive, responsive, and accessible, with features like high contrast and screen reader support.
7. **Integration with Existing Systems**
 - Must integrate with the restaurant's POS and kitchen management systems for order synchronization.
8. **Third-Party Integrations**



- Should support integrations with payment gateways and reservation systems, ensuring security and reliability.
- 9. **Localization and Multilingual Support**
 - Must support multiple languages, currencies, and regional pricing.
- 10. **Compliance with Local Regulations**
- 11. **Maintenance and Updates**
 - Should allow easy menu updates, bug fixes, and system changes without interrupting app use.

2.6 User Documentation

NA

2.7 Assumptions and Dependencies

1. **Reliable Internet Infrastructure**

The application is dependent on the availability and reliability of internet services to ensure consistent user access and engagement. Users must have a stable internet connection for real-time updates, order processing, and payment transactions.
2. **Device Compatibility and Performance**

The application assumes that users will have devices (smartphones, tablets, desktops) with sufficient hardware and software capabilities to run the app smoothly. The app's performance may vary based on the device's operating system version and available resources.
3. **Third-Party Services Availability**

The app relies on third-party services, such as payment gateways (e.g., Stripe, PayPal) and reservation management tools. The functionality of these integrations depends on their uptime and proper integration with the app's backend.

3. External Interface Requirements

3.1 User Interfaces

The system shall have an intuitive and user-friendly interface for both mobile and web platforms.

3.2 Hardware Interfaces

The system shall interface with POS terminals and printers for billing and order processing.



3.3 Software Interfaces

Frontend Development:

HTML/CSS/JavaScript: Using these standard web development technologies for building the user interface of your website.

React or Angular or Vue.js: Choosing JavaScript framework to build dynamic and responsive user interfaces.

Styling and responsive design (Bootstrap or Tailwind CSS): Ensuring that your website is mobile-friendly and has a responsive design.

Backend Development:

Server-side Language (e.g., Node.js, Python, Ruby): Choosing a server-side language to handle the business logic and server-side operations.

Web Framework (e.g., Express.js for Node.js, Flask/Django for Python/Spring boot): Using a web framework to streamline backend development.

Database (e.g., MongoDB, MySQL, PostgreSQL): Choosing a database to store user data, exercise information, and other relevant data.

User Authentication (OAuth or JWT): Implementing secure user authentication mechanisms to protect user data.

Payment Gateway (Stripe, PayPal, etc.): Website offers premium features or services, integrate a secure payment gateway for transactions.

Hosting and Deployment (AWS, Heroku, Netlify, or similar services): Choosing a reliable hosting service for deploying and scaling your application.

3.4 Communications Interfaces:

The system shall support communication protocols for online reservations, order updates, and notifications.

4. System Features

4.1 User management:

4.1.1 Description and Priority:

This feature facilitates user account creation and login, enabling customers to manage their profiles seamlessly.



Priority: High

4.1.2 Functional Requirements

4.1.2.1 REQ-1: User Registration

- Users can register with a unique username and password.
- The system validates the uniqueness of the username.
- Mandatory fields include username, password, age, and gender.
- Email verification is sent upon successful registration.

4.1.2.2 REQ-2: User Authentication

- Users can securely log in with their registered credentials.
- The system employs encryption for storing and transmitting passwords.
- Failed login attempts trigger account lockout after three consecutive failures.
- Users can reset their passwords through a secure email verification process.

4.1.2.3 REQ-3: Profile Setup

- After registration, users are prompted to set up their profiles.
- Mandatory profile information includes name, address, age, payment information.
- User can optionally add more details such as their avatar, favorite food items, offers, timings ,etc.

4.1.2.4 REQ-4: Profile Editing

- Users can update their personal information such as name, address, payment options, avatars, etc.

4.1.2.5 REQ-5: GDPR Compliance

- The system adheres to GDPR regulations regarding user data protection and privacy.
- Users can request and download their data, and request account deletion.

5. Other Nonfunctional Requirements

5.1 Performance Requirements:

- The system must be able to handle a minimum of 500 concurrent users without performance degradation.
- Table availability and menu updates should be reflected in real-time (within 2 seconds).
- Order processing and bill generation should not take more than 5 seconds per transaction.

5.2 Safety Requirements

Food Allergy Disclaimer:



- **Requirement:** Users must acknowledge and agree to a food allergy disclaimer during account creation or at the time of ordering, stating they are aware of possible allergens in the food.
- **Rationale:** This ensures user safety and mitigates the risk of allergic reactions due to lack of information.

Age Restrictions for Alcohol Orders:

- **Requirement:** Users under the age of 18 (or as per regional laws) must be restricted from ordering alcoholic beverages, with a verification process in place.
- **Rationale:** Ensures compliance with legal regulations and prevents underage alcohol consumption.

Emergency Contact for Health Incidents:

- **Requirement:** The app should allow restaurants to record emergency contact details for customers in case of health-related incidents.
- **Rationale:** Provides a safety net for addressing unexpected emergencies.

Clear Food Labels:

- **Requirement:** All menu items must include clear labels for allergens, nutritional information, and dietary preferences (e.g., vegan, gluten-free).
- **Rationale:** Helps users make informed decisions about their orders.

5.3 Security Requirements

Data Encryption:

- **Requirement:** All user data, including personal information, payment details, and order history, must be encrypted using industry-standard protocols (e.g., AES-256).
- **Rationale:** Safeguards user information from unauthorized access and breaches.

User Authentication:

- **Requirement:** Implement secure authentication mechanisms (e.g., strong passwords, two-factor authentication) to protect user accounts.
- **Rationale:** Prevents unauthorized access to user accounts and sensitive data..

Role-Based Access Control (RBAC):

- **Requirement:** Restrict access to administrative features (e.g., menu updates, order management) based on roles (e.g., admin, staff, customer).
- **Rationale:** Ensures only authorized personnel can access sensitive functionalities.



Privacy Policy Compliance:

- **Requirement:** The app must include a comprehensive privacy policy detailing how user data is collected, stored, and used, with explicit user consent required.
- **Rationale:** Builds user trust and ensures compliance with data protection regulations.

Data Access Logging:

- **Requirement:** Implement access logs to track user and admin activities within the system, providing traceability in case of security incidents.
- **Rationale:** Enhances accountability and facilitates security investigations.

5.4 Software Quality Attributes

Usability:

- **Requirement:** The app must achieve a System Usability Scale (SUS) score of 85 or higher, ensuring an intuitive and user-friendly interface.
- **Rationale:** Enhances user experience, encouraging repeat use and customer satisfaction.

Availability:

- **Requirement:** The system must guarantee 99.9% uptime, ensuring consistent accessibility for customers and restaurants.
- **Rationale:** Minimizes disruptions to operations and customer frustration.

Scalability:

- **Requirement:** The system must be capable of scaling to support 1,000+ concurrent users as business demands grow.
- **Rationale:** Ensures the app remains efficient as customer and restaurant traffic increases.

Reliability:

- **Requirement:** The system should process 95% of orders without errors under peak load conditions.
- **Rationale:** Demonstrates robustness and reliability during high-demand periods.

Flexibility:

- **Requirement:** The system should allow the seamless addition of new features, such as loyalty programs or payment methods, without significant code changes.
- **Rationale:** Future-proofs the platform against evolving industry trends.

Robustness:



- **Requirement:** The system must handle invalid inputs gracefully and provide meaningful error messages (e.g., when a table is unavailable or payment fails).
- **Rationale:** Prevents crashes and improves user satisfaction by offering clear guidance.

5.5 Business Rules

Booking Confirmation and Approval:

- **Rule:** Customers must confirm their table bookings within a specified time frame, or the reservation will be released.
- **Implication:** Requires an automated booking confirmation system with reminders for users.

Age Verification for Alcohol Orders:

- **Rule:** Alcoholic beverages can only be ordered upon successful age verification.
- **Implication:** Requires integration of an age-verification module for legal compliance.

Payment and Billing:

- **Rule:** Bills must be generated automatically after the order is marked as served, and payments should be processed through secure gateways.
- **Implication:** Requires an automated billing and payment system with support for multiple payment methods.

Menu Updates:

- **Rule:** Only authorized restaurant staff can update the menu, including pricing, availability, and offers.
- **Implication:** Requires role-based access and an admin dashboard for menu management.

Order Cancellation Policy:

- **Rule:** Orders can only be canceled within a specified time after placement; otherwise, cancellation fees may apply.
- **Implication:** Requires a cancellation policy framework and logic for refund processing.

Customer Feedback Collection:

- **Rule:** Customers must be prompted to provide feedback after completing their dining experience.
- **Implication:** Requires a feedback module to gather and analyze customer reviews.



Appendix A: Glossary

NA

Appendix B: Analysis Models

NA

Appendix C: To Be Determined List

NA

For Faculty Use

Correction Parameters	Formative Assessment [40%]	Timely completion of Practical [40%]	Attendance / Learning Attitude [20%]	
Marks Obtained				