**Software Requirements Specification (SRS)**

**Project Title:** Online Food Delivery System  
**Date:** October 04, 2024  
**Version:** 1.0  
**Author:** Mr.Anurag Sanjay Jadhav  
**Approvers:** Prof.Dr.Dipali Sale.

**Table of Contents**

1. [Introduction](#1-introduction)  
   1.1 [Purpose](#11-purpose)  
   1.2 [Scope](#12-scope)  
   1.3 [Definitions, Acronyms, and Abbreviations](#13-definitions-acronyms-and-abbreviatio)  
   1.4 [References](#14-references)
2. [Overall Description](#2-overall-description)  
   2.1 [Product Perspective](#21-product-perspective)  
   2.2 [Product Features](#22-product-features)  
   2.3 [User Classes and Characteristics](#23-user-classes-and-characteristics)  
   2.4 [Operating Environment](#24-operating-environment)  
   2.5 [Assumptions and Dependencies](#25-assumptions-and-dependencies)
3. [System Features](#3-system-features)  
   3.1 [Feature 1: User Registration and Authentication](#31-feature-1-user-registration-and-auth)  
   3.2 [Feature 2: Restaurant Search and Menu Browsing](#32-feature-2-restaurant-search-and-menu)  
   3.3 [Feature 3: Order Placement and Tracking](#33-feature-3-order-placement-and-tracki)  
   3.4 [Feature 4: Payment Integration](#34-feature-4-payment-integration)  
   3.5 [Feature 5: Notifications](#35-feature-5-notifications)
4. [External Interface Requirements](#4-external-interface-requirements)  
   4.1 [User Interfaces](#41-user-interfaces)  
   4.2 [Hardware Interfaces](#42-hardware-interfaces)  
   4.3 [Software Interfaces](#43-software-interfaces)
5. [Other Non-Functional Requirements](#5-other-non-functional-requirements)  
   5.1 [Performance Requirements](#51-performance-requirements)  
   5.2 [Security Requirements](#52-security-requirements)  
   5.3 [Usability Requirements](#53-usability-requirements)
6. [Appendix](#6-appendix)

**1. Introduction**

**1.1 Purpose**

The purpose of this document is to define the software requirements for the **Online Food Delivery System**. This application enables users to browse restaurants, view menus, place orders, and track delivery in real-time. It serves as a guide for developers, testers, and stakeholders to ensure the project is implemented correctly.

**1.2 Scope**

The system allows users to register, log in, search for nearby restaurants, browse menus, place orders, and make payments. It will also allow real-time tracking of orders and notifications for order status updates. The platform supports multiple payment options and operates across web and mobile devices.

**1.3 Definitions, Acronyms, and Abbreviations**

* **SRS:** Software Requirements Specification
* **UI:** User Interface
* **API:** Application Programming Interface
* **CRUD:** Create, Read, Update, Delete
* **OTP:** One-Time Password

**1.4 References**

* IEEE Standard for SRS Documentation
* Payment Gateway API Documentation
* [Links to related documentation]

**2. Overall Description**

**2.1 Product Perspective**

The **Online Food Delivery System** is a web and mobile-based platform that connects users to restaurants. Users can search for restaurants, browse menus, place orders, and pay through the system. The platform also tracks deliveries and notifies users of order status updates.

**2.2 Product Features**

* User registration and authentication
* Restaurant and menu browsing
* Order placement and tracking
* Integrated payment gateway
* Notifications for order updates and promotions
* Reviews and ratings for restaurants and dishes

**2.3 User Classes and Characteristics**

* **General Users:** People who want to browse restaurants, place orders, and track delivery.
* **Restaurant Owners:** Manage menus, order processing, and promotions.
* **Delivery Personnel:** Track and manage order deliveries in real-time.
* **Admin Users:** Oversee system operations, monitor transactions, and handle disputes.

**2.4 Operating Environment**

* **Frontend:** React Native for mobile and React.js for web
* **Backend:** Node.js with Express
* **Database:** MongoDB for user and order data
* **Operating Systems Supported:** Windows, macOS, Android, iOS
* **Browsers Supported:** Chrome, Firefox, Safari, Edge

**2.5 Assumptions and Dependencies**

* Reliable third-party APIs for payment processing (e.g., Stripe, PayPal).
* Stable internet connection is required to use the platform.
* Restaurants will maintain up-to-date menus and inventory.

**3. System Features**

**3.1 Feature 1: User Registration and Authentication**

**Description:**  
Users must be able to register using email or social media accounts (Google, Facebook) and log in securely. They should be able to recover passwords via email and verify their identity using OTP.

**Functional Requirements:**

* Users can register using an email address or social media accounts.
* OTP verification via SMS or email during account creation.
* Secure login with password encryption.
* Password reset option via email.

**3.2 Feature 2: Restaurant Search and Menu Browsing**

**Description:**  
Users should be able to search for nearby restaurants, filter by cuisine type, and view available menus.

**Functional Requirements:**

* Users can search restaurants by name, location, or cuisine.
* Restaurant pages display menus with prices and dish descriptions.
* Filters for price, rating, and delivery time.

**3.3 Feature 3: Order Placement and Tracking**

**Description:**  
Users should be able to add items to a cart, place an order, and track it in real-time.

**Functional Requirements:**

* Add multiple dishes to a shopping cart.
* View order summary before checkout.
* Place orders and track delivery progress (preparing, out for delivery, delivered).
* Option to cancel the order within a certain time frame.

**3.4 Feature 4: Payment Integration**

**Description:**  
Users should be able to make payments using credit/debit cards, digital wallets, or cash on delivery.

**Functional Requirements:**

* Support for multiple payment methods (Stripe, PayPal, digital wallets).
* Option for cash on delivery.
* Integration of a secure payment gateway with transaction logs.

**3.5 Feature 5: Notifications**

**Description:**  
The system should notify users about the order status, delivery times, and promotions.

**Functional Requirements:**

* Push notifications for order status updates (placed, out for delivery, delivered).
* Email notifications for new promotions and discounts.
* SMS alerts for delivery confirmations.

**4. External Interface Requirements**

**4.1 User Interfaces**

The system will have an intuitive and responsive UI on both web and mobile platforms, with the following pages:

* **Login and Registration Pages:** For user authentication.
* **Restaurant Search Page:** Search, browse, and filter restaurants.
* **Order Management Page:** Review the cart, place orders, and track delivery.

**4.2 Hardware Interfaces**

The application is web-based and mobile-based, requiring a computer or smartphone with an internet connection.

**4.3 Software Interfaces**

* **Payment Gateway API:** Integration with Stripe or PayPal for secure payment processing.
* **SMS API:** Integration for OTP verification and order status updates.
* **Map API:** Google Maps or similar services for delivery tracking.

**5. Other Non-Functional Requirements**

**5.1 Performance Requirements**

* The system should load in under 3 seconds for standard internet connections.
* Must support up to 50,000 concurrent users.
* Orders should be processed within 2 seconds after confirmation.

**5.2 Security Requirements**

* Secure payment gateways with PCI DSS compliance.
* Data encryption for sensitive user information (passwords, payment details).
* Role-based access control for users, restaurants, and admins.

**5.3 Usability Requirements**

* The application should be easy to navigate for non-technical users.
* The mobile app must be fully responsive and optimized for different screen sizes.
* Accessibility standards should follow WCAG 2.1 guidelines.

**6. Appendix**

* Glossary of terms used in the system.
* User stories and personas.
* Use case diagrams for key system interactions.

This SRS outlines all key features, functionalities, and requirements necessary for the development of the **Online Food Delivery System**. Each section provides a clear and structured approach for developers, stakeholders, and testers to follow during the lifecycle of the project