Version 1.0

Name.  
Date.

Software Requirements Specification

Online Food Delivery Application

# Table of Contents

**Table of Contents.......................................................................................................................... ii Revision History............................................................................................................................ ii**

**1. Introduction............................................................................................................................. 1**

1.1 Purpose............................................................................................................................................. 1

1.2 Document Conventions.................................................................................................................... 1

1.3 Intended Audience and Reading Suggestions.................................................................................. 1

1.4 Product Scope...................................................................................................................................1

1.5 References........................................................................................................................................ 1

1. **Overall Description................................................................................................................. 1**
   1. Product Perspective .......................................................................................................................... 1
   2. Product Functions.............................................................................................................................2
   3. User Classes and Characteristics......................................................................................................2
   4. Operating Environment.................................................................................................................... 2
   5. Design and Implementation Constraints.......................................................................................... 3
   6. User Documentation.........................................................................................................................3
   7. Assumptions and Dependencies.......................................................................................................3
2. **External Interface Requirements...........................................................................................3**
   1. User Interfaces..................................................................................................................................3
   2. Hardware Interfaces..........................................................................................................................3
   3. Software Interfaces...........................................................................................................................3
   4. Communications Interfaces..............................................................................................................4
3. **System Use Cases.....................................................................................................................4**
   1. U1...................................................................................................
   2. U2..................................................................................................................................................... 5

4.2 U3..................................................................................................................................................... 6

1. **Other Nonfunctional Requirements............................................................................................. 8**
   1. Performance Requirements.............................................................................................................. 8
   2. Safety Requirements.........................................................................................................................8
   3. Security Requirements......................................................................................................................8
   4. Software Quality Attributes..............................................................................................................8 **6. System Requirements Chart...................................................................................................8**

**Appendix A: Glossary...................................................................................................................9**

**Appendix B: Analysis Models...................................................................................................... 9**

**Appendix C: To Be Determined List.........................................................................................10**

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Member | 16-07-24 | Rechecking the document | 1.0 |
|  |  |  |  |

# 1. Introduction

## 1.1 Purpose:

This document outlines the software requirements for a food delivery application, aiming to facilitate the delivery of food to customers within a city. It covers the entirety of the application.

## 1.2 Document Conventions:

The SRS follows the conventions of using Times 11pt font for regular text and Times 14pt bold for headings. Requirements are identified using a two-part numbering system, with each requirement having its own explicitly stated priority level.

## 1.3 Intended Audience and Reading Suggestions:

This document is intended for developers, project managers, marketing staff, users, testers, and documentation writers. It contains an overview of the software, detailed requirements, and prioritization.

Readers are suggested to begin with the overview sections and then proceed to sections pertinent to their role.

## 1.4 Product Scope:

The food delivery application aims to streamline the process of ordering and delivering food to customers within a city. The software's purpose is to provide a convenient platform for users to select and order food from various restaurants, facilitating efficient delivery services. The software aligns with business goals of enhancing customer satisfaction and increasing operational efficiency.

## 1.5 References:

1. Reference 1
2. Reference 2

# 2. Overall Description

## 2.1 Product Perspective:

This food delivery application is a standalone product designed to facilitate the ordering and delivery of food from various restaurants within a city. It serves as a platform connecting customers with restaurants and delivery personnel. The application interacts with external systems such as restaurant databases, payment gateways, and mapping services to provide its services.

## 2.1 Product Functions:

**1.**User registration and authentication

**2.**Browse and search restaurants and menus

**3**.Place food orders

**4.**Track order status in real-time

**5.**Payment processing

**6**.Provide feedback and ratings for orders and delivery experience

## 2.2 User Classes and Characteristics:

**1.Customers:** Regular users who order food for delivery. They have varying technical expertise and may use the application frequently.

**2.Delivery Personnel:** Users responsible for delivering orders. They require access to order information and navigation features to reach the delivery location efficiently.

## 2.3 Operating Environment:

The food delivery application will operate on both web and mobile platforms. Supported environments include:

**Web:** Google Chrome (latest version), Mozilla Firefox (latest version), Safari (latest version)

**Mobile:**

* Android: Devices running Android 7.0 (Nougat) and above.
* iOS: iPhone 6 and above, running iOS 12 and above.

## 2.4 Design and Implementation Constraints:

The design and implementation of the food delivery application are subject to several constraints:

Firstly, compliance with data protection regulations such as GDPR and HIPAA is imperative. The application must ensure the privacy and security of user data, adhering to the guidelines outlined in these regulations to protect sensitive information.

Secondly, integration with existing payment gateways is essential for facilitating secure transactions within the application. The integration must meet industry standards for encryption and authentication to ensure the confidentiality and integrity of financial transactions.

Additionally, the application must utilize scalable and efficient database technologies to effectively manage large volumes of orders and user data. The chosen database solution should be capable of handling increasing loads as the user base expands, without compromising on performance or data integrity.

## 2.5 User Documentation:

1. User documentation for the food delivery application will include user manuals, on-line help, and tutorials.
2. User manuals will provide step-by-step guidance on using the app, while on-line help will offer quick assistance for specific tasks. Tutorials will offer interactive learning experiences.
3. All documentation will be accessible digitally through the app and website for user convenience.

## 2.6 Assumptions and Dependencies:

Assumptions and dependencies play a crucial role in the development and functionality of the food delivery application.

Assumptions include the availability of a stable internet connection for users to access the application seamlessly. Additionally, continuous access to restaurant menus and pricing information through APIs is assumed to ensure accurate ordering.

Dependencies involve the integration with third-party mapping services for real-time location tracking of delivery personnel. Furthermore, the application relies on payment gateway APIs for secure transaction processing, emphasizing the importance of reliable external services for core functionalities.

# 3. External Interface Requirements

## 3.1 User Interfaces:

The user interface of the food delivery application adheres to specific design standards for consistency and ease of use. All interface elements follow a standardized layout with a main area for displaying restaurant listings and order details, a menu section for navigation, and a status bar for feedback and notifications. Screen resolutions supported include standard mobile device resolutions such as 320x480, 375x667, and 414x736 pixels for iOS and similar resolutions for Android devices. The application aims to comply with Section 508 accessibility standards to ensure usability for users with disabilities.

## 3.2 Hardware Interfaces:

The food delivery application primarily operates as a software product and does not directly interact with hardware components. Therefore, there are no specific hardware interfaces required for the application.

## 3.3 Software Interfaces:

The application interfaces with various software components, including databases, operating systems, and external APIs. It relies on databases to store user information, order details, and restaurant menus. The operating system compatibility extends to both iOS and Android platforms. Additionally, the application integrates with third-party APIs for services such as mapping for real-time location tracking of delivery personnel and payment gateway APIs for secure transaction processing. While the specifics of the integration protocols are not detailed here, the application relies on these interfaces for essential functionalities.

## 3.4 Communications Interfaces:

Communication functions within the food delivery application include interactions with external systems such as email services for order confirmations and notifications. The application communicates over standard protocols such as HTTP for web-based interactions and possibly SMTP for email communications. Communication security is ensured through encryption mechanisms to protect sensitive data such as payment information during transactions. Data transfer rates and synchronization mechanisms are optimized to ensure timely and reliable communication between the application and external services.

# 4. System Use Cases

**4.1 Food Delivery Application Working (U1)**

1. **Objective** - The objective of this use case is to enable users to place food orders through the app, facilitating convenient and efficient food delivery services.
2. **Priority** – High
3. **Source** – John Smith (End-user)
4. **Actors** – Customer

Application (service)

Admin

1. **Flow of Events**

a. **Basic Flow**:

* + - * Customer opens the application and logs in or sign ups (if not already) by entering his name, phone number and address.
      * Admin manages the application
      * Customer then selects item/s and places the order.
      * Application calculates the total price to be paid by the customer.
      * Customer then pays the bills.
      * Application chooses the delivery personnel.
      * Customer can also track the delivery ride.

b. **Alternative Flow(s)**

* + - * If an item is unavailable, the customer may choose a replacement or remove it from the cart.
      * If the user experiences connectivity issues during checkout, they are prompted to retry or use alternative payment methods.

c. **Exception Flow(s)**:

If the delivery personnel encounter unexpected delays, the user is notified and provided with revised delivery estimates.

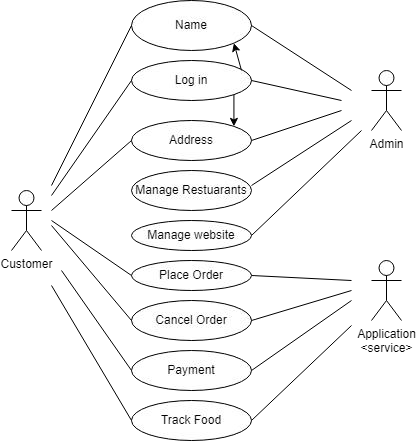
1. **Preconditions** –

The customer is logged into their account.

The app has access to the user's location for delivery purposes.

1. **Post conditions** -

The order is successfully placed and visible in the customer's order history The food is delivered to the customer in time.



**4.2 Placing an Order (U2)**

1. **Objective** – The objective of this use case is to enable users to place food orders through the app, facilitating convenient and efficient food delivery services.
2. **Priority** – High
3. **Source** – Carl Gnome (End-user)
4. **Actors** – Customer

Restaurant

1. **Flow of Events**

a. **Basic Flow**

* + - * Customer manages the order by first selecting item/s or deals to order by their choice by adding them to cart.
      * Then the customer chooses the amount of money needed.
      * Customer confirms the choice.
      * System validates the amount.
      * System asks central bank computer to debit the customer’s account.
      * System issues money to the user.

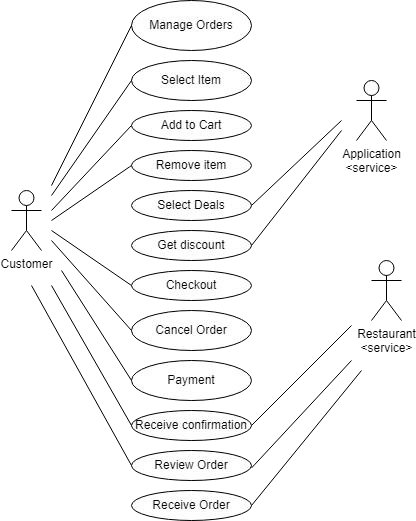
b. **Alternative Flow**:

* + - * If an item is out of stock, the user is notified and prompted to remove it from the cart or select a replacement.
      * If the user encounters connectivity issues during checkout, they are prompted to retry or use alternative payment methods.

c. **Exception:**

If the delivery address provided by the user is invalid or unreachable, the user is prompted to provide a valid address.

1. **Preconditions** – User is logged in the app has access to the user's location for accurate restaurant suggestions and delivery options
2. **Post conditions** – The user receives a confirmation of the order.



**4.3 Deliver the order (U3)**

1. **Objective** - The ultimate objective of this use-case is to facilitate the successful delivery of food orders from restaurants to customers' specified delivery locations.
2. **Priority** – High
3. **Source** – John Smith (Delivery Personnel)
4. **Actors** – Customer

Application (service)

Restaurant (service)

1. **Flow of Events**
   1. **Basic Flow**:
      * After customer has places the order, the application alerts the chosen restaurant about the order to be prepared.
      * Restaurant prepares the order and sends it to the delivery personnel who was assigned by application service and was most suitable.
      * Customer is also notified about the order and delivery details.
   2. **Alternative Flow(s)**
      * If the delivery personnel encounters traffic or unexpected delays to the delivery location, they provide updated estimated arrival times to the user via the app.
      * If the delivery personnel is unable to locate the delivery address, they contact the user for clarification or assistance.
   3. **Exception Flow(s)**:

If the delivery personnel encounter unexpected delays, the user is notified and provided with revised delivery estimates.

If the delivered order is incorrect or damaged, the delivery personnel informs the user and contacts support.

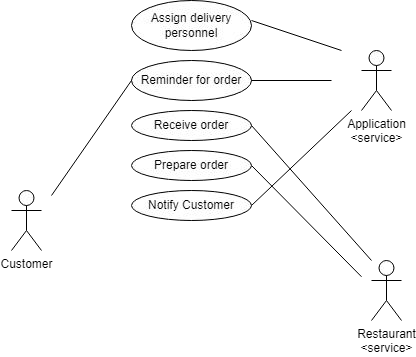
1. **Preconditions** –

The customer is logged into their account.

The app has access to the user's location for delivery purposes.

1. **Post conditions** -

The order is successfully placed and visible in the customer's order history The food is delivered to the customer in time.



# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

1. The app must respond to user interactions.
2. Orders must be processed and confirmed within 30 seconds of being placed by the user.
3. Estimated Time of Arrival (ETA) for food deliveries must be accurate within a margin of error of +/5 minutes.

## 5.2 Safety Requirements

1. Implement end-to-end encryption for all user data, including personal information, payment details, and order history, to prevent unauthorized access or data breaches.
2. Payment transactions must be processed securely. The app must ensure secure payment processing.

## 5.3 Security Requirements

1. All sensitive user data transmitted between the app and servers must be encrypted.
2. The app must implement robust authentication mechanisms, such as multi-factor authentication (MFA) or biometric authentication, to verify the identity of users during login and account access.
3. The app must provide users with a clear and comprehensive privacy policy that outlines how their data is collected, used, and shared, as well as their rights and options regarding data privacy.

## 5.4 Software Quality Attributes

1. The app must be compatible with major mobile platforms.
2. The app could be updated in terms of new restaurants, deals or other bugs.
3. The app is reliable and usable with task such as success rate, efficiency, and user satisfaction.

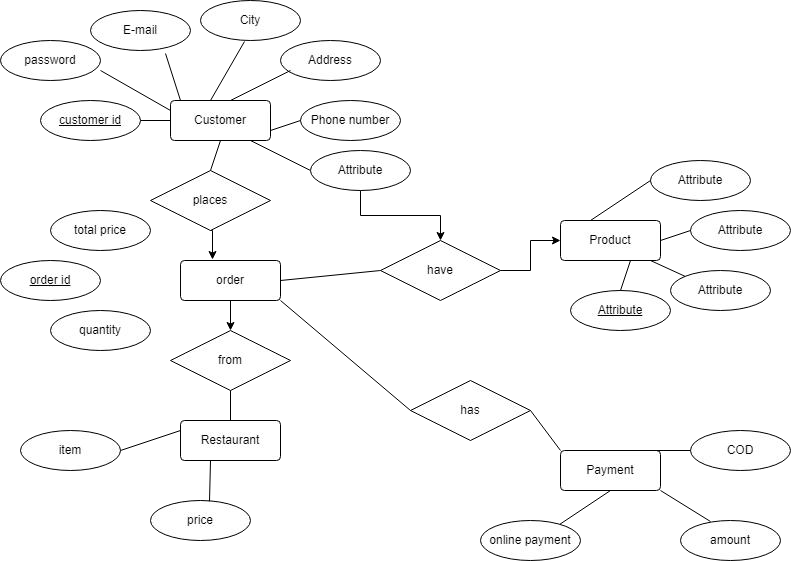
# 6. System Requirements Chart

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Priority** | **Type** | **Source** | **Contained in Use Case(s)** | **Description** |
| 1 | High | F | John Smith Customer | U1, U2, U3 | The system shall allow users to place food orders from available restaurants. |
| 2 | High | F | John Smith Customer | U2, U3 | The system shall provide real-time tracking of order status and delivery. |
| 3 | Medium | NF | Stakeholders | N/A | The system shall maintain a response time of under 2 seconds for all user interactions. |
| 4 | High | NF | Security Team | N/A | The system shall encrypt all sensitive user data and payment information using industry-standard encryption algorithms. |
| 5 | High | F | John Smith Customer | U1, U3 | The system shall allow users to search for restaurants based on location, cuisine, and other filters. |
| **ID** | **Priority** | **Type** | **Source** | **Contained in Use Case(s)** | **Description** |
| 6 | High | NF | Compliance Team | N/A | The system shall comply with GDPR and other relevant data protection regulations. |
| 7 | Medium | NF | Developers | N/A | The system shall be scalable to handle a 100% increase in concurrent users during peak hours without performance degradation. |

# Appendix A: Glossary

1. Customer/User: an individual who uses the food delivery app to order food.
2. Delivery personnel: person who delivers the order/ food to the customer.
3. COD: Cash on Delivery.

# Appendix B: Analysis Models



# Appendix C: To Be Determined List

1. TBD: Localization and internationalization requirements for supporting multiple languages and regions.
2. TBD: Specific legal agreements and compliance requirements, including privacy policies and terms of use.
3. TBD: Reusable components, modules, or libraries to be integrated into the project for efficiency and cost savings.
4. TBD: Error handling mechanisms and logging practices for capturing and reporting system errors and exceptions.
5. TBD: Clarification on any additional requirements or specifications needed for the successful implementation of the SRS.

