# **Software Requirements Specification (SRS) for Food Delivery Service Management Software (FDSMS)**

**Version:** 1.0 (Approved)

**Prepared by:**Shshank Mittal (22CS30051)  
Sujan Jain (22CS10075)  
Ashish Kumar(22CS30014)

**Organization:** Indian Institute of Technology (IIT) Kharagpur  
**Date:**

## **Table of Contents**

1. **Introduction**
   * 1.1 Purpose
   * 1.2 Document Conventions
   * 1.3 Intended Audience and Reading Suggestions
   * 1.4 Product Scope
   * 1.5 References
2. **Overall Description**
   * 2.1 Product Perspective
   * 2.2 Product Functions
   * 2.3 User Classes and Characteristics
   * 2.4 Operating Environment
   * 2.5 Design and Implementation Constraints
   * 2.6 User Documentation
   * 2.7 Assumptions and Dependencies
3. **External Interface Requirements**
   * 3.1 User Interfaces
   * 3.2 Hardware Interfaces
   * 3.3 Software Interfaces
   * 3.4 Communications Interfaces
4. **System Features**
   * 4.1 Register a Customer/Delivery Agent/Restaurant
     + 4.1.1 Description
     + 4.1.2 Stimulus/Response Sequences
     + 4.1.3 Functional Requirements
   * 4.2 Menu of a Restaurant
     + 4.2.1 Description
     + 4.2.2 Stimulus/Response Sequences
     + 4.2.3 Functional Requirements
   * 4.3 Order by a Customer
     + 4.3.1 Description
     + 4.3.2 Stimulus/Response Sequences
     + 4.3.3 Functional Requirements
   * 4.4 Promotional offers
     + 4.4.1 Description
     + 4.4.2 Stimulus/Response Sequences
     + 4.4.3 Functional Requirements
   * 4.5 Provide feedback
     + 4.5.1 Description
     + 4.5.2 Stimulus/Response Sequences
     + 4.5.3 Functional Requirements
   * 4.6 Restaurant
     + 4.6.1 See available delivery agent
       - 4.6.1.1 Description
       - 4.6.1.2 Stimulus/Response Sequences
       - 4.6.1.3 Functional Requirements
     + 4.6.2 Set the available orders for pickup in an area
       - 4.6.2.1 Description
       - 4.6.2.2 Stimulus/Response Sequences
       - 4.6.2.3 Functional Requirements
     + 4.6.3 See pending orders
       - 4.6.3.1 Description
       - 4.6.3.2 Stimulus/Response Sequences
       - 4.6.3.3 Functional Requirements
   * 4.7 Delivery Agent
     + 4.7.1 Mark their location
       - 4.7.1.1 Description
       - 4.7.1.2 Stimulus/Response Sequences
       - 4.7.1.3 Functional Requirements
     + 4.7.2 See Delivery Requests available in an area
       - 4.7.2.1 Description
       - 4.7.2.2 Stimulus/Response Sequences
       - 4.7.2.3 Functional Requirements
     + 4.7.3 Provide estimated pick-time from a restaurant and delivery time to a customer
       - 4.7.3.1 Description
       - 4.7.3.2 Stimulus/Response Sequences
       - 4.7.3.3 Functional Requirements
   * 4.8 Management
     + 4.8.1 Provide Restaurant and Food Recommendations to Customers
       - 4.8.1.1 Description
       - 4.8.1.2 Stimulus/Response Sequences
       - 4.8.1.3 Functional Requirements
     + 4.8.2 Can manage list of customers, restaurants and delivery agents
       - 4.8.2.1 Description
       - 4.8.2.2 Stimulus/Response Sequences
       - 4.8.2.3 Functional Requirements
5. **Other Nonfunctional Requirements**
   * 5.1 Performance Requirements
   * 5.2 Safety Requirements
   * 5.3 Software Quality Attributes
   * 5.4 Business Rules

### **1. Introduction**

#### **1.1 Purpose**

The aim of this document is to offer a comprehensive depiction of the Food Delivery Service Management system. It serves as a detailed guide outlining the system's purpose, functionalities, interfaces, and operational constraints. Our application caters to a diverse user base, including restaurants, delivery agents, and customers, facilitating various services such as menu selection, delivery location designation, and order management. Additionally, the system incorporates value-added features like promotional offers and user ratings, enhancing the overall user experience. Notably, this platform serves as a valuable resource for individuals unable to venture outside due to safety concerns or limited resources, while also providing restaurants with an avenue to expand their customer base..

#### **1.2 Document Conventions**

The document follows standard formatting conventions, employing Arial font for text and Times New Roman for headings. Proper indentation and numbering are used for clarity.

#### **1.3 Intended Audience and Reading Suggestions**

This document comprehensively covers both technical and non-technical aspects of the software, serving as a valuable resource for developers and end-users alike. It provides insights into the underlying motivations driving the development of the software and delves into the intricate details of its implementation. Anybody who wants to use the software can read the appropriate parts of the document, a list of which is given in the Table of Contents

**1.4 Product Scope**

FDSMS is a web-based application designed to streamline food delivery services, catering to customers, restaurants, and delivery agents. It offers features for menu management, order tracking, and user feedback.

#### **1.5 References**

The document draws inspiration from IEEE Std 830-1998, adhering to recommended practices for Software Requirements Specification.

### **2. Overall Description**

#### **2.1 Product Perspective**

FDSMS is an independent software product developed to address the growing demand for efficient food delivery services. It operates within the client-server model, where users interact with a web-based interface served by both front-end and back-end components.

#### **2.2 Product Functions**

FDSMS offers a range of functions aimed at facilitating food delivery services:

* Direct food delivery from restaurants to customers
* Third-party delivery services
* Menu management for restaurants
* Real-time order status updates
* Order history tracking
* User rating and feedback mechanisms

#### **2.3 User Classes and Characteristics**

The system caters to four primary user classes:

**Restaurant:**

* Create and manage the menu for all available items.
* Review and decide whether to accept or reject received orders, providing status updates for accepted orders and assigning estimated service times.
* Assign delivery agents to orders.
* Mark orders as "Out for Delivery" once prepared.

**Customer:**

* Browse menus of various restaurants and place orders for food items.
* Receive updates on order status, informed by both restaurant and delivery agent activities.
* Rate restaurants and delivery personnel based on service quality.

**Delivery Agent:**

* Set their current location for order pickup and view incoming delivery requests from restaurants.
* Accept delivery requests, specifying estimated delivery times, and update order statuses accordingly.

**Management:**

* Oversee the statuses of customers, restaurants, and delivery agents, managing ratings and feedback.
* Provide food and restaurant recommendations to customers.
* Offer promotions to loyal customers.

#### **2.4 Operating Environment**

FDSMS operates as a web application, accessible through standard web browsers. The front-end interface is developed using HTML, CSS, and JavaScript, while the back-end relies on the Flask framework in Python. Data is managed through a Firebase database.

#### **2.5 Design and Implementation Constraints**

The use of Firebase as the database imposes certain constraints, such as compliance with Firebase's usage policies. Any changes in these policies may impact system functionality.

#### **2.6 User Documentation**

The application provides intuitive user interfaces, supplemented by FAQs for common queries. Technical documentation for developers includes API documentation and online resources.

#### **2.7 Assumptions and Dependencies**

The system assumes users have access to a stable internet connection and possess a valid email address for registration. Dependencies include Google's Material.io for front-end design and Firebase for database management.

### **3. External Interface Requirements**

#### **3.1 User Interfaces**

The system offers separate interfaces for each user class, featuring login/sign-up functionalities and personalized dashboards tailored to specific user roles.

* **Homescreen**: Provides options for login and signup.
* **Management Dashboard**: Displays user details, lists of users, ratings, recommendations, and promotions.
* **Delivery Agent Dashboard**: Allows marking current location, viewing order lists, order details, and providing feedback.
* **Restaurant Dashboard**: Facilitates menu creation, order management, and viewing past orders.
* **Customer Dashboard**: Enables browsing restaurants, menus, placing orders, tracking order status, and accessing past orders.

#### **3.2 Hardware Interfaces**

The system does not have specific hardware requirements for users. However, the backend server requires sufficient processing power to handle incoming requests effectively.

#### **3.3 Software Interfaces**

FDSMS interacts with Google's Material.io for front-end design and Firebase for database management. It relies on the Flask framework for server-side processing and communicates with clients via standard HTTPS protocol.

#### **3.4 Communications Interfaces**

All communication within the system occurs via web browsers using HTTPS protocol, ensuring secure data transmission.

### **4. System Features**

#### **4. System Features**

#### **4.1 Register a Customer/Delivery Agent/Restaurant**

##### **4.1.1 Description**

Users must register and log in to access specific profiles, providing essential information for subsequent activities.

##### **4.1.2 Stimulus/Response Sequences**

Users select the profile they wish to register and enter relevant details, which are stored securely.

##### **4.1.3 Functional Requirements**

* **Signup**: Users navigate to the signup page, select their profile type, and enter required details. Upon authentication, they are directed to their respective user dashboard.
* **Login**: Users enter credentials created during signup for authentication, leading to access to their user dashboard.

#### **4.2 Menu of a Restaurant**

##### **4.2.1 Description**

Restaurants manage menus by adding, deleting, or modifying items. Menus are displayed upon request by restaurants or customers.

##### **4.2.2 Stimulus/Response Sequences**

Menus are presented in table format, allowing restaurants to add/delete items and modify details. Customers can select items to add to their order list.

##### **4.2.3 Functional Requirements**

* Front-end JavaScript dynamically adds menu items to the page.
* Database stores menu items and orders.

#### **4.3 Order by a Customer**

##### **4.3.1 Description**

Customers can view the status of pending orders. Restaurants update order status, which customers and delivery agents can track.

##### **4.3.2 Stimulus/Response Sequences**

Order status is updated manually by restaurants and delivery agents, with timestamps for updates. Status is displayed as a timeline with checkpoints and average time for next steps.

##### **4.3.3 Functional Requirements**

* Restaurants and delivery agents update statuses manually.
* Estimated time for next step provided by restaurants or delivery agents.

#### **4.3 Order by a Customer**

##### **4.3.4 Description**

Customers can also view their previous completed orders for reference and reordering.

##### **4.3.5 Stimulus/Response Sequences**

Customers access their past order list through the user interface, displayed in a list format.

##### **4.3.6 Functional Requirements**

* Order information stored in the database, linked to customer and restaurant IDs.
* Past orders displayed using CSS and JavaScript.

#### **4.4 Promotional Offers**

##### **4.4.1 Description**

Customers can view and apply promotional offers provided by management to avail discounts. Management can create and manage offers.

##### **4.4.2 Stimulus/Response Sequences**

Customers access the offers section to view available promotions and apply them during order placement. Management creates and manages offers through dedicated interfaces.

##### **4.4.3 Functional Requirements**

* Offers stored as class objects in the database.
* Applied offers removed from the list after use.

##### **4.4.4 Description**

Management can create and manage promotional offers for customers based on their ratings.

##### **4.4.5 Stimulus/Response Sequences**

Management accesses a list of offers, adds or deletes offers, and assigns them to customers based on their ratings.

##### **4.4.6 Functional Requirements**

* Offer management facilitated through database and JavaScript.
* Offers assigned to customers based on their ratings.

#### **4.5 Provide Feedback**

##### **4.5.1 Description**

Customers can provide feedback and ratings for restaurants and delivery agents based on their service experience.

##### **4.5.2 Stimulus/Response Sequences**

Feedback provided through a star rating system in the user interface. Feedback stored and averaged for rating calculation.

##### **4.5.3 Functional Requirements**

* Feedback captured using CSS and JavaScript.
* Ratings stored as floating-point numbers and averaged for each entity.

#### **4.6 Restaurant Features**

##### **4.6.1 See Available Delivery Agents**

###### **Description**

Restaurants can view a list of available delivery agents to assign orders for delivery.

###### **Stimulus/Response Sequences**

Restaurants access the list of available delivery agents and send delivery requests to selected agents.

###### **Functional Requirements**

Similar to "Set the available orders for pickup in an area" functionality.

##### **4.6.2 Set the Available Orders for Pickup in an Area**

###### **Description**

When orders are ready for pickup, restaurants update their status and make them available for pickup in the area.

###### **Stimulus/Response Sequences**

Upon order preparation, restaurants mark orders as available for pickup, which appear in delivery agents' dashboards as pickup orders.

###### **Functional Requirements**

Orders marked as available are added to the database and displayed in delivery agents' lists.

##### **4.6.3 See Pending Orders**

###### **Description**

Restaurants can view a list of pending orders, including order details and statuses.

###### **Stimulus/Response Sequences**

Restaurants access the list of pending orders through their dashboard, where they can update order statuses.

###### **Functional Requirements**

Orders and their details stored in the database and displayed using front-end JavaScript and CSS.

#### **4.7 Delivery Agent Features**

##### **4.7.1 Mark Their Location**

###### **Description**

Delivery agents can set their current location to indicate their working area.

###### **Stimulus/Response Sequences**

Delivery agents select their current location from a dropdown list.

##### **4.7.2 See Delivery Requests Available in an Area**

###### **Description**

Delivery agents can view a list of available delivery requests in their working area and accept them.

###### **Stimulus/Response Sequences**

Delivery agents access the list of available orders and accept them, providing estimated delivery times.

###### **Functional Requirements**

Orders and their details displayed in delivery agents' dashboards, updated in real-time.

##### **4.7.3 Provide Estimated Pickup and Delivery Time**

###### **Description**

Delivery agents provide estimated pickup time from the restaurant and delivery time to the customer.

###### **Stimulus/Response Sequences**

Delivery agents enter estimated times while accepting orders, which are updated in the order status.

###### **Functional Requirements**

Estimated times stored in the database and updated in the order status.

#### **4.8 Management Features**

##### **4.8.1 Provide Restaurant and Food Recommendations to Customers**

###### **Description**

Management provides food and restaurant recommendations to customers based on ratings.

###### **Stimulus/Response Sequences**

Management selects recommended restaurants and food items, which are displayed to customers.

###### **Functional Requirements**

Recommendations managed through database and displayed to customers based on their ratings.

##### **4.8.2 Manage List of Customers, Restaurants, and Delivery Agents**

###### **Description**

Management can view and manage lists of customers, restaurants, and delivery agents, pushing offers and recommendations.

###### **Stimulus/Response Sequences**

Management accesses user lists and interacts with them, pushing offers and recommendations as needed.

###### **Functional Requirements**

User lists managed through database interactions, with offers and recommendations assigned based on criteria.

### **5. Other Nonfunctional Requirements**

#### **5.1 Performance Requirements**

### The system should handle database queries efficiently to ensure quick response times. Balancing performance and accuracy is crucial, achieved through optimization techniques.

#### **5.2 Safety Requirements**

### The application operates within web browsers, minimizing risks to user devices. However, data storage and server integrity are vital to prevent potential damage during heavy usage.

#### **5.3 Software Quality Attributes**

### **Maintainability:** The system should be easy to maintain, allowing for seamless updates and addition of new features. Cost-effective maintenance and upgradability are essential.

### **Usability:** The application should be user-friendly, easy to learn, and navigate. Simple, intuitive design enhances user experience**.**

### **Flexibility:** The system should be flexible, allowing for modifications and adaptations to new technologies and third-party components seamlessly.

#### **5.4 Business Rules**

### The software is freely available for public use, and its source code is open for modification and use. This fosters collaboration and innovation within the development community.

### **Conclusion**

### The Food Delivery Service Management Software (FDSMS) aims to revolutionize food delivery by providing a convenient platform for users to order food from restaurants and manage deliveries efficiently. With a focus on user experience, performance, and safety, FDSMS seeks to enhance the food delivery experience for customers, restaurants, delivery agents, and management alike.

### This Software Requirements Specification serves as a comprehensive guide for developers and stakeholders, outlining the system's functionalities, interfaces, and nonfunctional requirements. By adhering to these specifications, the development team can ensure the successful implementation of FDSMS, meeting the needs and expectations of its users.

### **Appendix**

### **Glossary:** Defines technical terms and acronyms used in the document.

### **References:** Lists resources consulted during the development of this document.

### **This concludes the Software Requirements Specification for the Food Delivery Service Management Software (FDSMS). For any further inquiries or clarifications, please refer to the contact information provided in the document.**

### 