SOFTWARE REQUIREMENTS SPECIFICATION

Food Delivery Service Management System

Prepared by:
Parth Patil (22CS30041)
Rishit Garg (22CS30045)
Sachish Singla (22CS30046)

Indian Institute of Technology Kharagpur

April 1, 2024

Contents

1. Revision History..... 4

2. Introduction.....5

2.1 Purpose..... 5
2.2 Document Conventions..... 5
2.3 Intended Audience and Reading Suggestions.....5
2.4 Product Scope..... 5
2.5 References..... 6

3 Overall Description.....7
3.1 System Environment.....7
3.2 Product Perspective.....7
3.3 Product Functions.....8
3.4 User Characteristics..... 8
3.5 Operating Environment.....9
3.6 Implementation Constraints..... 9
3.7 Assumptions and Dependencies..... 9

4 External Interface Requirements.....10
4.1 User Interfaces..... 10
4.1.1 HomeScreen.....10
4.1.2 Customer..... 10
4.1.3 Restaurant.....11
4.1.4 Delivery Agent.....11
4.1.5 Management.....12
4.2 Hardware Interfaces..... 12
4.3 Software Interfaces.....12
4.4 Communication Interface.....12

5 System Features.....13
5.1 Onboarding a new Customer/ Delivery Agent/ Restaurant.....13
5.1.1 Description.....13
5.2.2 Stimulus / Response Sequences.....13
5.1.3 Functional Requirements.....13
5.2 Restaurant Menu..... 13
5.2.1 Description.....13

5.2.2 Stimulus / Response Sequences....13
5.2.3 Functional Requirements....14
5.3 Handling Orders....14
5.3.1 Description....14
5.3.2 Stimulus / Response Sequences....14
5.3.3 Functional Requirements....14
5.4 Order Tracking....15
5.4.1 Description....15
5.4.2 Stimulus / Response Sequences....15
5.5.3 Functional Requirements....15
5.5 Promotional Offers....15
5.5.1 Description....15
5.5.2 Stimulus / Response Sequences....15
5.5.3 Functional Requirements....16
5.6 Feedback System....16
5.6.1 Description....16
5.6.2 Stimulus / Response Sequences....16
5.6.3 Functional Requirements....16
5.7 Management....16
5.7.1 Description....16
5.7.2 Stimulus / Response Sequences....16
5.7.3 Functional Requirements....17
5.8 Statistics....17
5.8.1 Description....17
5.8.2 Stimulus / Response Sequences....17
5.8.3 Functional Requirements....17
5.9 Recommendations....17
5.9.1 Description....17
5.9.2 Stimulus / Response Sequences....17
5.9.3 Functional Requirements....18

6 Use Case Diagram....19

7 Class Diagram....20

8 Non-Functional Requirements....21
8.1 Performance Requirements....21
8.2 Safety Requirements....21
8.3 Software System Attributes....21
8.3.1 Maintainability....21
8.3.2 Flexibility....21

8.3.3 Usability..... 21
8.3.4 Portability..... 21
8.4 Business Rules..... 22

1. Revision History

Name	Date	Reason for Changes	Version
Creation	April 1, 2024	Template changes and additions	1.0

2. Introduction

2.1 Purpose

The purpose of this document is to provide a detailed description of the Food Delivery Service Management Software. This Software has four main user types: Customer, Restaurant, Delivery Agent, and Management. The application is designed so that the interface, functionality, and constraints of the system will vary from user to user. For example, the restaurant can change its menu and assign delivery agents, whereas the customer can choose from several restaurants and place their order. Ratings of Restaurants, Delivery Agents, and Customers will be maintained to improve user experience. This software will benefit all its users as it increases the customer's convenience, restaurant reach, and delivery agent employment.

2.2 Document Conventions

This Software Requirement Specification Document has been written using free writing tools such as Google Docs, which are typed in Arial font for normal text and Bold Arial for the headings. The font size used is 11 for text and 16, 23 for headings. All headings are highlighted appropriately in bold. The document is prepared using the UK English convention.

2.3 Intended Audience and Reading Suggestions

This SRS targets system developers for building the system, testers to ensure its functionality, and project stakeholders (product managers, investors) to understand the system's goals and target users. 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 on Page 2.

2.4 Product Scope

The Food Delivery Management System facilitates online food ordering and delivery for restaurants, delivery agents, and customers. It provides features for menu browsing, order placement, delivery tracking, and user management. This streamlines food delivery operations, increases restaurant reach, and offers a convenient service for those unable to dine out. The

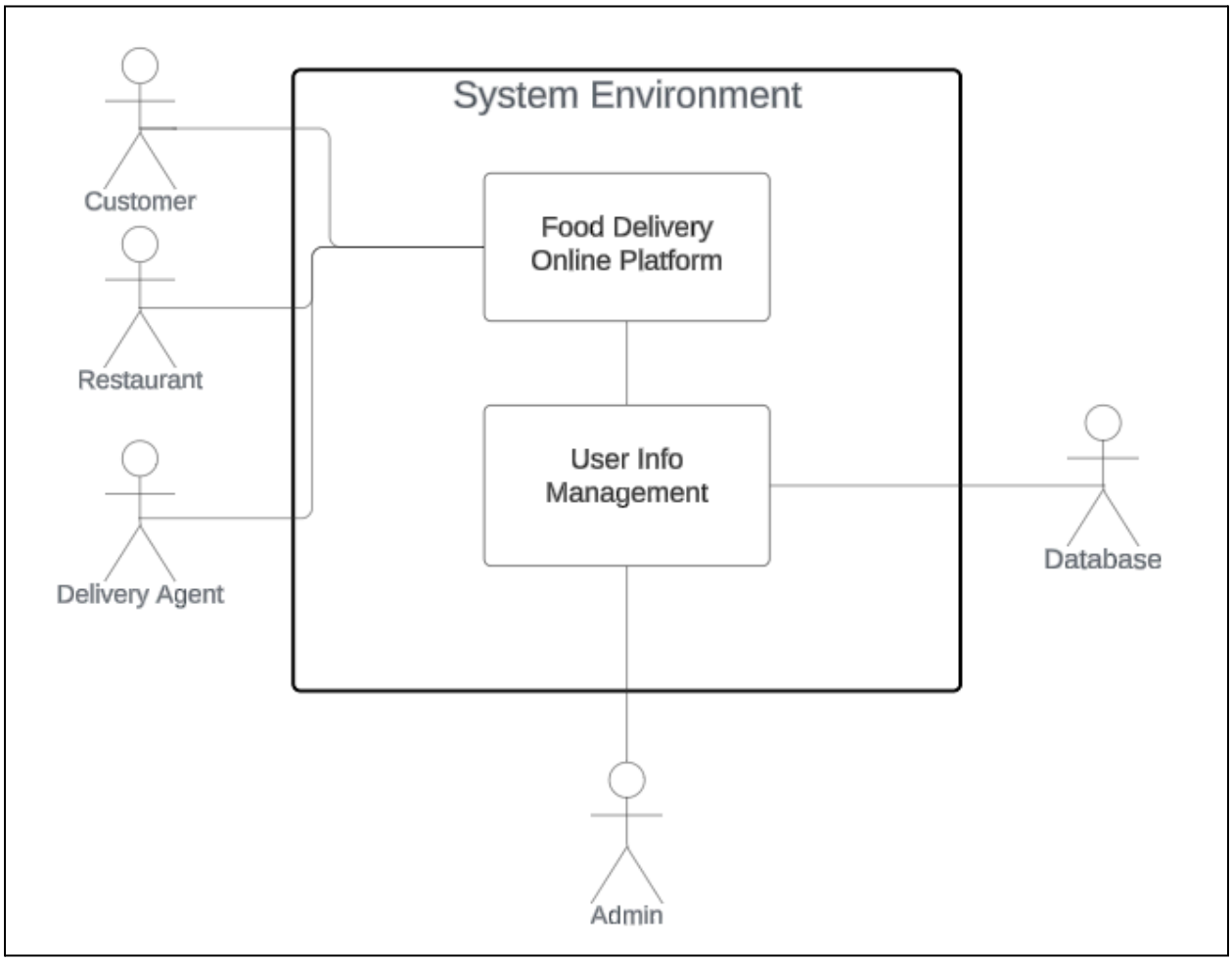
system allows managers to create promotions and offers. All users can view their past orders. Restaurants can disable dishes as required and set their working hours.

2.5 References

This document is based on the IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specification from the IEEE Computer Society in 1998.

3 Overall Description

3.1 System Environment



3.2 Product Perspective

The self-contained Food Delivery Service Management Software was developed in response to the realization that, on occasion, people find it difficult or impossible to visit restaurants because of things like discomfort, time constraints, or lack of transportation. By providing affordable

delivery costs, enticing discounts, and safe food delivery from well-known eateries, the project seeks to alleviate this problem.

3.3 Product Functions

The key functions of FDMS are:

- Facilitate food delivery to customers directly from restaurants
- Create employment opportunities for local people as delivery agents
- Provide convenience to customers by allowing them to see estimated delivery time and track pending orders
- Maintain a feedback system for reviewing restaurants and delivery agents, giving customers the option to read others' reviews

3.4 User Characteristics

The expected activities and interactions of different types of users are listed below.

- Customer:
  – View a list of different restaurants in their vicinity
  – Order from multiple restaurants simultaneously
  – View their past orders
  – Review restaurants and delivery agents
  – See and use promotional offers
  – Track orders and view the estimated time of delivery
  – Get recommendations from the platform
  – Add or remove restaurants from a list of favourites
  – Add multiple addresses to the application
  – Choose one of the saved addresses as the delivery address or enter a new address

- Restaurant:
  – View the menu and add dishes
  – Disable/Enable dishes for order temporarily
  – View pending orders and order history
  – View feedback given by customers
  – View statistics based on order history

- Delivery Agent:
  – View information related to assigned orders

– View past orders
– Enter an OTP for confirmation while delivering orders
– Update location via GPS of the device or coordinates
– Enable auto-updating of the location while working.

- Management:
  – View information of all users registered on the platform
  – Maintain ratings of delivery agents and restaurants
  – Provide restaurant recommendations to customers
  – Create and remove promotional offers for customers

3.5 Operating Environment

The client-side software is a web-based GUI implemented using React JS. It is independent of the operating system used and runs in any modern browser. The server side logic is implemented in Javascript, which uses Node JS compiler, which runs on all modern operating systems. MongoDB is used as the database for the software. It is hosted on cloud and provided by MongoDB Atlas.

3.6 Implementation Constraints

The use of TomTom API for getting location data limits the accuracy of the software used by TomTom. Using MongoDB limits the implementation to the usage policy of the database platform.

3.7 Assumptions and Dependencies

It is assumed that the user possesses a functional internet connection and an email address to access the web app. The dependencies for the project will be available in the requirements file (package.json) of the project.

4 External Interface Requirements

4.1 User Interfaces

The home page will contain links to log in pages of all four types of users: customer, restaurant, delivery agent, and management. The users will be redirected to their respective dashboards on logging in.

4.1.1 HomeScreen

- About Us
    - Contains brief information about the platform
- Login
    - Ask for username and password
    - Authenticate and navigate to the dashboard
- Sign Up
    - Input various details of the user
    - Register a new user

4.1.2 Customer

- View Available Restaurants
    - View restaurants available for delivery
    - Search for restaurants by dishes, cuisines, and name
    - Add or remove restaurants from a list of favorites

- Place Order
    - View items in a cart
    - Choose payment type (Cash or Card)
    - Provide a delivery address
    - Avail promo offers

- Provide Feedback
    - Review restaurants after an order has been delivered
    - Review delivery agent after an order has been delivered

- Profile
    - Display personal details
    - Edit personal details
    - View past and pending orders

- Order info
    - Live tracking of orders by displaying a map with the route taken by the delivery agent
    - See the estimated time of delivery for a pending order

4.1.3 Restaurant

- Handle Menu
    - Set an item name and price for a new dish
    - Add a dish image file to display
    - Set availability of existing dishes

- Profile
    - Display restaurant details
    - Edit restaurant details
    - View past and pending orders
    - Add a restaurant image file to display

4.1.4 Delivery Agent

- Set working status
    - Update Location via GPS to start working
    - Pause working temporarily
    - Automatically start working on updating the location

- Profile
    - Display personal details
    - Edit personal details
    - View past and pending orders

- Orders
    - View pending and delivered orders
    - Accept orders via OTP

4.1.5 Management

- Customer List
    - View customer details and order details
    - Provide recommendations for customers
    - Provide promotional offers

- Restaurant List
    - View restaurant details
    - Maintain feedback from customers

- Delivery Agent List
    - View agent details
    - Accept payment of collected cash
    - Maintain feedback from customers

- Offers
    - View the list of active offers
    - Delete existing offers

4.2 Hardware Interfaces

Although users do not need to meet any specific hardware interface requirements, it is advised to use a desktop computer or laptop for the best possible online content presentation.

4.3 Software Interfaces

The platform should easily be able to integrate with online payment gateways for ease of the customer. Also, customers should be able to contact the delivery agents in case of a delayed order without revealing the delivery agent's contact information. For this, integration with a third-party calling service will be required. Ultimately, the application will be made available as a web application. The hosting will be done locally for testing purposes. Because the online application is compatible with multiple operating systems and browsers, users will be able to access it easily on various platforms.

4.4 Communication Interface

All communication will occur through a web browser using the standard HTTPS protocol.

5 System Features

5.1 Onboarding a new Customer/ Delivery Agent/ Restaurant

5.1.1 Description

Joining the platform shall require a registration and logging in process. Relevant information must be taken from different kinds of users.

5.2.2 Stimulus / Response Sequences

The user shall choose the kind of user they are once they enter the website. Their information shall be taken according to their features. Logging in shall grant users access to various pages depending on the class.

5.1.3 Functional Requirements

The home page shall contain paths to log in pages of each user. New users shall have the option to go to the sign up page. If the user already exists in the database, they must be authenticated before granting access to the web pages. Additionally, restaurants shall have the option to upload an image which will be visible to the customer.

5.2 Restaurant Menu

5.2.1 Description

The restaurant must be able to create its menu after signing up. Restaurants must be able to enable or disable dishes based on availability. The menu will be displayed to the customer, from where they can place orders.

5.2.2 Stimulus / Response Sequences

The contents of the menu shall be displayed in a tabular form. Restaurants may choose to make changes in the availability and price of dishes. They may add dishes at any point, which must reflect on the customer side.
Customers can choose from different restaurants to order from. They will see a list of available dishes when choosing a particular restaurant.

5.2.3 Functional Requirements

The system shall allow restaurants to manage their menus. It shall facilitate adding the name, price, and (optionally) image of each dish. The feature shall integrate with the order management side of the website to ensure customers cannot order unavailable items.

5.3 Handling Orders

5.3.1 Description

Customers can place orders based on the available options. Each dish must have a preparation time period, during which the delivery agent will arrive for pick up. The customer shall be able to track pending orders. All types of users should be able to view relevant past and pending orders.

5.3.2 Stimulus / Response Sequences

The customer shall create a new order by choosing dishes, delivery address, promo code, and payment method. The restaurant will automatically accept orders if all the dishes are available. A delivery agent who has updated his status as working and is the closest to the mentioned restaurant will be allocated to deliver the order. Upon completion of the preparation time, the delivery agent will pickup the order and proceed to deliver it to the customer. The delivery agent will be able to view the pickup and delivery address.
The customer will be able to view the status of his order, the delivery agent assigned, the OTP for completion of the order, and the estimated delivery time.

5.3.3 Functional Requirements

The system shall enable customers to browse restaurants, select items, and place orders from multiple restaurants at the same time. A cart must be maintained for every session that the user logs in. The cart contents shall empty after the customer logs out. The customer has the option to pay via cash or card. The restaurant, delivery agent, and customers can view the pending and delivered orders. Each order must have an automatically generated OTP, which is necessary for the delivery agent to complete an order.

5.4 Order Tracking

5.4.1 Description

Each order needs to be tracked from start to end. The platform should know which orders have been completed. The customer should be able to see the status of the order and the live location of the delivery agent.

5.4.2 Stimulus / Response Sequences

On creating a new order, the customer shall be able to see the details of the pending order. They shall be able to view the order status, as well as the live location of the delivery agent. The platform may choose to provide a phone number for contacting the delivery agent in case of a delayed order. The delivery agent device shall regularly send live locations to the platform. The customer will also be able to see the estimated delivery time according to their time zone.

5.5.3 Functional Requirements

Each order should be tracked from start to end during its lifespan. Orders will never be deleted from the database. The customer side should be able to receive live location information in case of an order that is yet to be delivered. The delivery agent side should send regular location updates via GPS. The customer will also be able to see the estimated delivery time according to their time zone.

5.5 Promotional Offers

5.5.1 Description

The management may choose to give customers offers for limited use. Each customer can avail of an offer once. The offer will be applied at the time of ordering and reduce the total billed amount.

5.5.2 Stimulus / Response Sequences

These deals will be displayed in the offers area of the customer dashboard page. The customer must click on the offer to take advantage of it; once clicked, it will be immediately withdrawn from the list of offers. The management will have access to a list of offers and can add or remove them. They will make that offer available to clients who use coupon codes.

5.5.3 Functional Requirements

Each offer will be stored as a class in the database. It will have a unique coupon code the management decides while creating the order. If an offer is availed, the order will be tagged with that particular offer.

5.6 Feedback System

5.6.1 Description

After an order is delivered, customers may choose to give ratings and reviews to restaurants and delivery agents based on the food and service quality.

5.6.2 Stimulus / Response Sequences

The customer will give the restaurant and/ or delivery agent one review per order. The restaurant and other customers can view restaurant reviews.

5.6.3 Functional Requirements

An average rating of restaurants and delivery agents must be maintained. The different reviews given by customers can be viewed by management and restaurants. The reviews given to delivery agents can be viewed only by the management for monitoring misconduct. While posting a review, the customer shall choose from a range of 1 to 5 stars and optionally leave a comment.

5.7 Edit Profile

5.7.1 Description

Users should be able to modify their profile information easily to ensure their account details are up to date.

5.7.2 Stimulus / Response Sequences

The user will go to the profile page. The user will choose to edit details. The software allows the user to change all details except for email ID. The user will save the changes and all changes will be reflected in the database.

5.7.3 Functional Requirements

The users shall be provided with an intuitive interface for editing details, including change of password. There must be validation checks depending on the type of field. The restaurant in particular can also upload a restaurant image to be viewed by customers.

5.8 Management

5.8.1 Description

This section has been kept separate to describe the role and functionalities provided to the management user class.
Management will have no signup page but will have an authenticated login page. They have access to most information stored on the database, except for sensitive information like passwords and OTPs.

5.8.2 Stimulus / Response Sequences

The dashboard will have links to lists of customers, restaurants, and delivery agents. Clicking on each individual user will open a page displaying the user’s information. The management can view feedback given by customers to restaurants and delivery agents. They may post offers from time to time for customers.

5.8.3 Functional Requirements

Management users will be stored as a separate class in the database. Management user access shall be restricted from sensitive data like passwords and OTP. Each page should be secured with authentication and can be viewed only after logging in as a management user.

5.9 Statistics

5.9.1 Description

The platform's growth will make it difficult to extract statistical information from orders. For this purpose, the platform must allow data to be presented in a systematic and visual manner.

5.9.2 Stimulus / Response Sequences

The management can view statistics for different restaurants. Each restaurant can view its own statistics. The restaurant can view their most ordered dishes, frequent customers, and other relevant information.

5.9.3 Functional Requirements

The system shall generate in-depth reports containing statistical data across various platform domains, including orders, customers, and delivery agents. Management users can select different report categories to view specific data sets relevant to their needs. Additionally, the system shall present statistical data in a clear and visually appealing format (charts, graphs, tables) for easy comprehension and informed decision-making.

5.10 Recommendations

5.10.1 Description

Recommendations is a key feature for the customer while ordering from a Food Delivery application. It provides the customer with a list of restaurants to order from based on their previous orders.

5.10.2 Stimulus / Response Sequences

When the customer wishes to see recommended restaurants, a unique list of restaurants will be made for the customer based on their ordering trends in the choice of restaurants and dishes ordered. The list will be sorted on their similarity to the ordering tendencies of the customer.

5.10.3 Functional Requirements

The software should keep track of all orders for each customer. Data from these orders is then used to search restaurants offering similar types of dishes. Restaurants where orders are placed frequently will also be displayed on top of the list of recommendations.

5.11 Payment Options

5.11.1 Description

It is important to provide the customer with different payment options for convenience. Transactions must be seamless and easily traceable by the management.

5.11.2 Stimulus / Response Sequences

On checking out and placing an order, the customer will choose a payment option. In case of card or bank transfer, the software will redirect the customer to the relevant gateway. The transaction is completed on the gateway, which will send an acknowledgement back to the software. In case of cash options, payment will be received by the delivery agent. The agent enters an OTP which is visible only to the customer.

5.11.3 Functional Requirements

The software should be able to handle cash on delivery, card payment, and digital wallet payments. For cash on delivery, the status of payment should be updated by the delivery agent. The management shall be able to clear payment dues of a delivery agent after accepting payments from the agent. In case of online payments, the software shall be versatile in terms of integration with different payment gateways.

5.12 Restaurant Payment Tracking

5.12.1 Description

In addition to facilitating customer payments, the food delivery software must include functionality for administrators to track payments owed to restaurants after deducting transaction fees. This feature ensures transparency in financial transactions between the platform and partner restaurants.

5.12.2 Stimulus / Response Sequences

On receiving a payment from the customer, the platform will record the transaction fee and the total payment to be given to the restaurant after fee deduction. The software shall provide the management with details of pending payments.

5.12.3 Functional Requirements

The management shall be able to configure and customize transaction fees. The software shall automatically deduct fees on completion of an order. The restaurant shall receive a summary of pending payments. The management shall be able to view pending payments for different restaurants. Payments can be settled via the platform by using aforementioned integration of payment gateways. The management and restaurants both should be able to view a history of payments.

6 Use Case Diagram

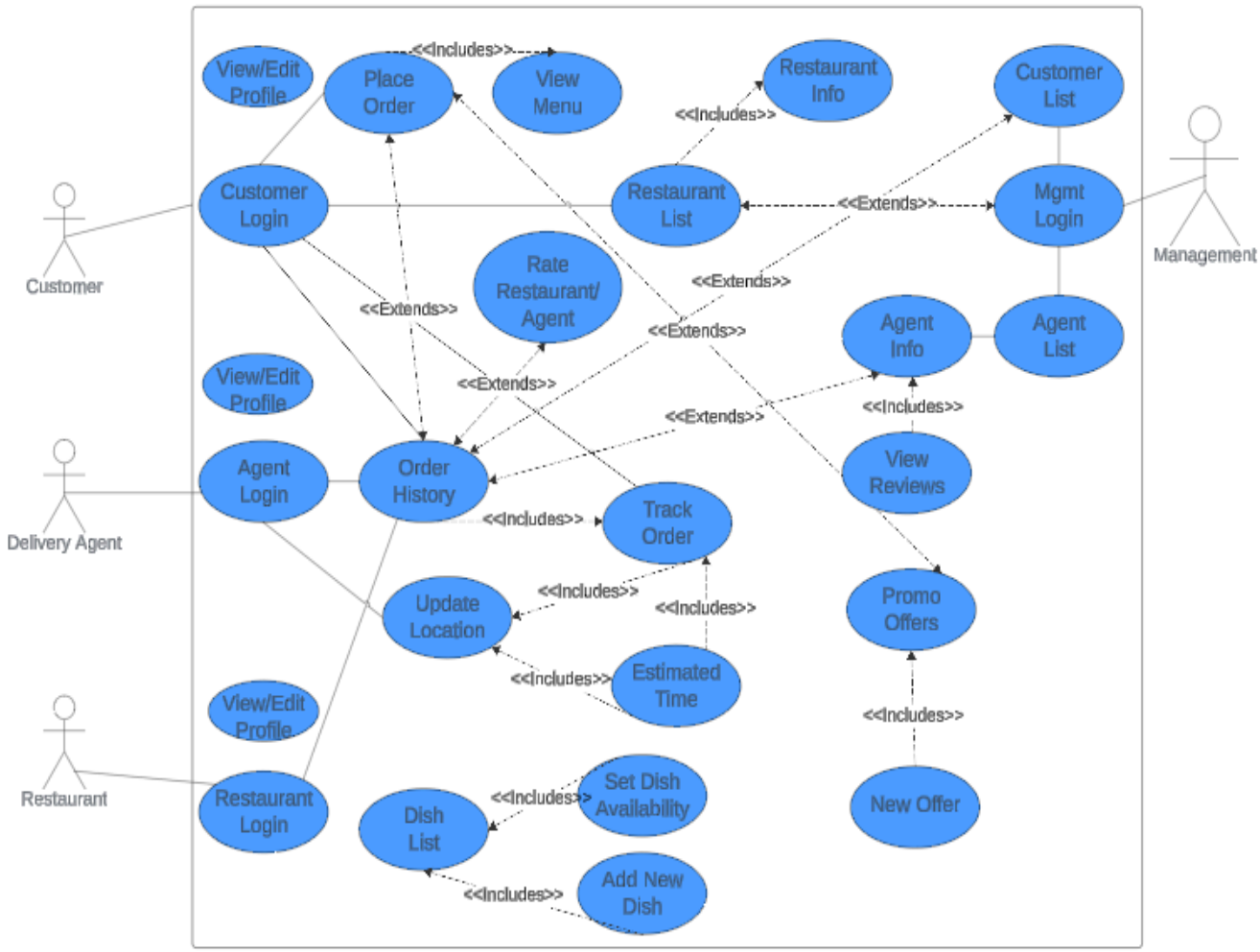
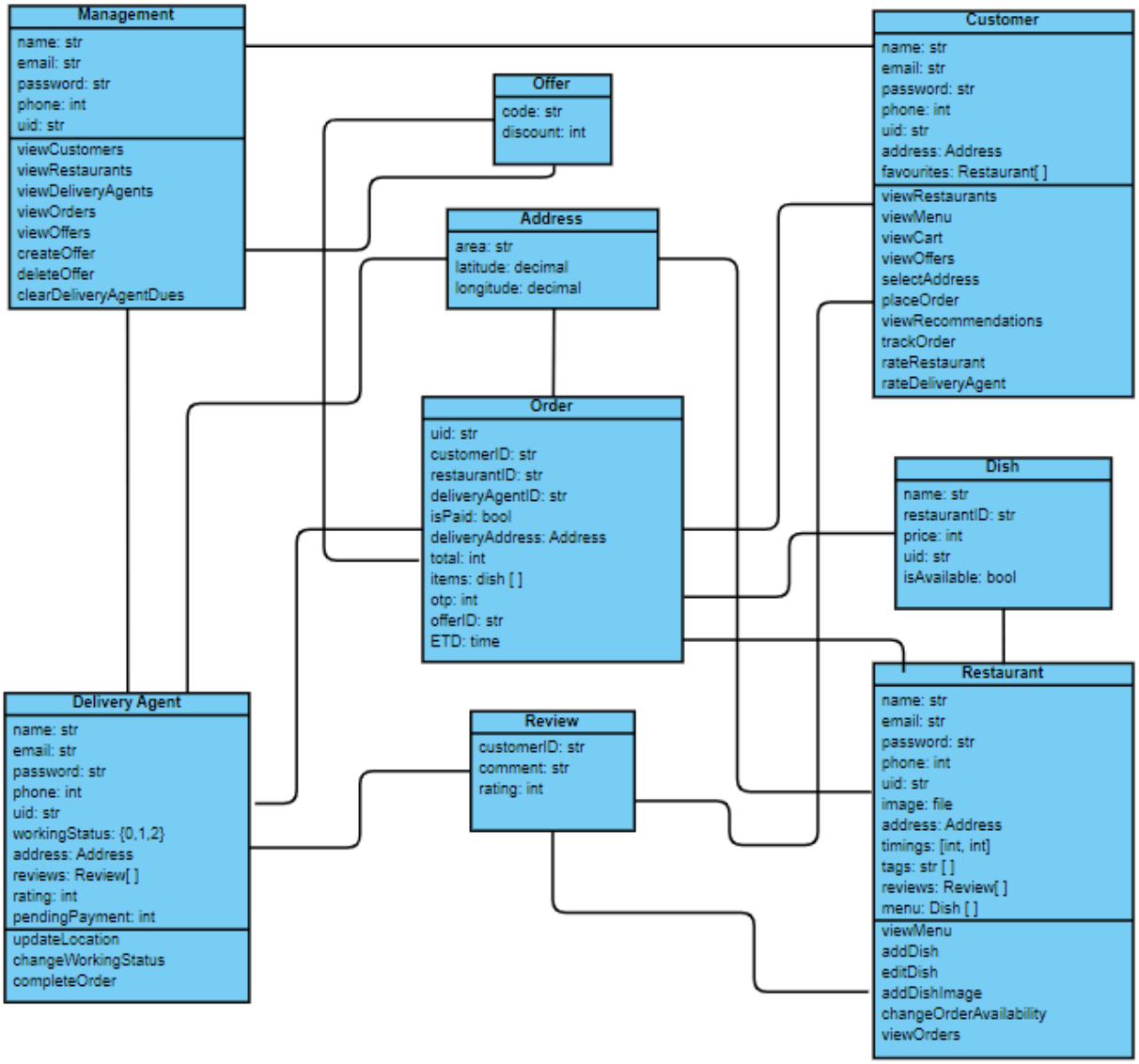


Figure 6.1: Use Case Diagram

7 Class Diagram



8 Non-Functional Requirements

8.1 Performance Requirements

The front end should be able to query data quickly by communicating with the backend. For this purpose, MongoDB can be used for a balance between speed and accuracy. In the case of tracking orders, the freely available TomTom API can be used for distance calculation, route generation, and map rendering.

8.2 Safety Requirements

The application does not provide a significant risk to the user's device because it is web-based. However, because there is a lot of reading and writing of data involved, damage to the server and data storage might only happen during times of high demand. All sensitive information of the users must be stored securely for cyber safety.

8.3 Software System Attributes

8.3.1 Maintainability

The code must have a modular design with clear interfaces for simplification of future bug fixes. A robust testing guide or framework must be provided for ease of developers.

8.3.2 Flexibility

The system must be easily configurable, and adding features must not be cumbersome. The scalability of the system must be taken into consideration while designing the software.

8.3.3 Usability

The software must have an intuitive and visually pleasing user interface. A simplistic design for ease of navigation through the website is necessary. The software must perform role-based access control, maintaining a clutter-free experience.

8.3.4 Portability

A widely used programming language must be chosen for implementation. Migration from one operating system to another must be made easy. The software must have minimal system dependencies. For the use of the developer, dependencies must be maintained in a requirements.txt file.

8.4 Business Rules

The software will be free to use for all users and the source code will be publicly hosted for free use and modification.