SOFTWARE REQUIREMENTS SPECIFICATION

for

Social Serving Food Delivery System (SSFDS)

Version 1.0

Prepared by

Devansha Dhanker(22CS10021)

More Aayush Babasaheb(22CS30063)

Shivam Choudhary(22CS10072)

Contents

1 Introduction

1.1 Purpose

The purpose of this project is to to develop an application for a non-profit organisation (NPO) that tracks leftover food from restaurants and hotels and uses it for several purposes - an NGO can buy the food at a higher discount rate, any user can buy the food at a comparatively discounted rate or the user can donate money to an NGO that will be used to charity workby the NGO.

1.2 Document Conventions

The following standards document conventions are used in our document: IEEE STD 830 -1998, IEEE Standard for Software Requirement Specification.

1.3 Intended Audience and Reading Suggestions

This document is intended to assist the users when they use the software and for developers and the project managers to plan their project and implement the software required.
This Software Requirement Specification document is divided into Six subsections:
Section 1: Introduction
Section 2: Overall Description of the Software giving information about functions, user classes, operating environment, constraints and documentation.
Section 3: External Interface Requirements giving a brief introduction to user, hardware, software and communications interfaces.
Section 4: Detailed functional requirements of different features.
Section 5: List of non-functional requirements
Section 6: Other requirements

1.4 Project Scope

The main goal of the project is to build an online Social Serving Food Delivery System, which will not only be helpful for the needy to get food, but also decreases the wastage of food.

1.5 References

The documents and websites referred to, are as follows:

- IEEE STD 830 -1998, IEEE Standard for Software Requirement Specification.

2 Overall Description

2.1 Product Perspective

The Online Social Serving Food Delivery System is an autonomous solution, offering a platform for restaurants to distribute surplus food to both NGOs and customers. Customers have the option to either purchase the food for themselves or donate money to NGOs for their operations.

2.2 Product Functions

The following functions are provided by the software:

• Login by different entities- Restaurant, NGO, Customer, NPO[Admin].

• The Restaurant can add the details (food item, plates, price per plate, address of the restaurant [lat, lon]).

• The NGO fills in the details (name, address [lat, lon]). They can view the available items which are at a distance of 10 km from them.

• NGO's can place the order for required amount of plates of a particular food item from a restaurant which is not more than 10km distance from their location.

• Customers(other than NGO) can see the the available items from all the restaurants which are at most 10km away.

• Customer(other than NGO) can place an order, mention their address [lat,lon] and opt for prepaid or cash on delivery option.

• For Customer, there is an option to self-pickup the food, in which case, the mode of payment should be prepaid.

• Customers have an option to donate to an NGO some amount of money [An option will be shown to select a particular NGO].

• The details of the transaction like Restaurants name, food items, date(be it by NGO or any other customer) will be updated on the NPO Portal.

2.3 User Classes and Characteristics

- Restaurant
  – Create account with login credentials.
  – Add address.
  – Add items, their respective count and price.
- Customer
  – Create account with login credentials.
  – Add address.
  – Check out various food items available from various restaurants.
  – Make order and choose whether purchasing for private use
  – Donate some money to an NGO.
  – Make payment or opt for COD.
- NGO
  – Create account with login credentials.
  – Add address.
  – Check out various food items available from various restaurants.
  – Place order make payment.
- NPO [Admin]
  – See all the registered restaurants and customers.
  – Assign time window to restaurants in which they can add details of left over food.

2.4 Operating Environment

The software is an online based portal and is thus platform independent and can be used in all well-known browsers.

2.5 Design and Implementation Constraints

- Limited Storage provided in the SQLAlchemy Database.
- Server issues by Stripe API.

2.6 User Documentation

The Webpage is user-friendly and intuitive.

2.7 Assumptions and Dependencies

The user must be familiar with the internet and capable to connect to the platform with a web browser.

- No accessibility features are provided on the platform assuming that the disabled users' device provide those if any such users use this platform.
- Connecting and using the platform requires:
  - working internet connection.
  - compatible browser

3 External Interface Requirements

3.1 User Interfaces

The product will feature a web application through which users will interact. Customers, NGOs, restaurants, and NPO administrators will each have their own distinct dashboards tailored to their specific roles and functionalities. Access to these dashboards will be granted through individual login credentials, directing users to their respective user interfaces.

- Homescreen
  – Login
  ∗ Email
  ∗ Password
  –User Signup
  ∗ Username
  ∗ Email-Address
  ∗ Address
  ∗Password
  ∗Customer or NGO
  – Restaurant-Signup
  ∗ Username
  ∗ Email-Address
  ∗ Address
  ∗Password

- NPO[Admin]
  – Personal Details
  – List of registered customers, NGOs restaurants.
  – Change time window of restaurant[when they can update the details of left over food items].
- Restaurant
  – Personal Details
  – Add a new food item
  ∗ Add title
  ∗ Add description
  ∗ Add available quantity
  ∗ Add price

– Remove a food item

– See order history

- Customer
  - Personal Details
  - List of available restaurants
    - List of food items their details
    - Add to cart option
  - Go to cart
  - Option to donate money to a NGO
  - Payment options : COD or prepaid
  - Delivery mode : Self pickup or through delivery agent
  - Purchase History
- NGO
  - Personal Details
  - List of available restaurants
    - List of food items their details
    - Add to cart option
  - Go to cart
  - Payment options : Prepaid
  - Purchase History

3.2 Hardware Interfaces

The hardware interfaces for the online portal are listed below:

- The portal can be used on any platform or PC which has a proper internet con-nection.
- Compatible web browser is required to avoid any problems.

3.3 Software Interfaces

The Software interfaces for the project are listed below:

- The Project will connect to Bootstrap 5 for Front-End Responsive Pages and MongoDB Cloud Database.
- The project is a web app and thus requires a working internet connection and a web browser only and is operating system independent.

• The Project uses Flask Framework in the server side and simple HTML,CSS and Javascript with Bootstrap in the client side. All the data coming from client side will be stored in an online MongoDB Cloud Database.

3.4 Communications Interfaces

All the communications will be done via the web-browser with the standard HTTP protocol.

4 System Features

4.1 Register a Customer/NGO/Restaurant

- Description
  – Getting on the platform requires the user to logged in to the specific profile. Different profiles have different features. This task is very important to access the all the features of a class
- Stimulus/Response Sequence
  – For Registering, User will be asked to provide specific information to make the profile.
- Functional Requirements
  – Register: As soon as the user enters they will see a general home page and there will be a link to go to the register page. There the user will enter the details in the respective boxes. After the details are authenticated/stored the user will be directed to the respective user page.
  – Login: Users will enter the credentials, created during the register process and then after authentication, they will be redirected to the respective user page.

4.2 Profile

- Description
  – User can see their profile and possibly edit their location.
- Stimulus/Response Sequence
  – Content of the profile will be shown here.
  – Edit button is available to edit location
  – Logout button is also provided to log user out
- Functional Requirements
  – This requires the use of some common Bootstrap framework to display buttons.

4.3 Menu Of The Restaurant

- Description

  – At the time of registering the restaurant will not be asked to create the menu but after the redirection, They will be allowed to add items, delete items and change the details of the menu item.For adding items to the menu the restaurant should have its location entered in the map. The menu will be displayed by request from the restaurant/customer/NGO.

- Stimulus/Response Sequence

  – The content of the menu will be shown in the card form with image at the left and other information at the right and buttons at the bottom of the information.

  – The restaurant can add/edit/delete menu items by clicking on the add/ed-it/delete button respectively.

  – The customer can select the items with number of plates from the menu and put them in the cart list by selecting the Add To Cart button. This will put the item in the cart list.

- Functional Requirements

  – This requires the use of some common Bootstrap framework to add the items as a card on the page, and using the database to store the cart.

4.4 Customer/NGO Cart

- Description

  – After Add To Cart button, Food and their number of plates are stored in the cart list. The cart list is different for every customer/NGO and can be accessed or will be displayed by request from the respective Customer/NGO. Also the user can’t add dishes from different restaurants at the same time in the cart.

- Stimulus/Response Sequence

  – The content of the cart will be shown in the card form with image at the left and other information at the right and buttons at the bottom of the information.

  – The User can add/edit/delete cart items by clicking on the add/edit/delete button respectively.

  – Total Price is to be calculated and shown here.

  – Button To Self-Pickup Option

  – Button To Make Payments is also provided.

- Functional Requirements
  – This requires the use of some common Bootstrap framework to add the items as a card on the page and to display buttons.

4.5 Customer Donate NGO

- Description

  – After Donating NGO Button, Page redirects to choose NGOs among the list.

- Stimulus/Response Sequence
  – The User can choose anyone NGOs among them.

  – A form opens where user can enter the amount to be donated(min amount is Rs. 100).

  – Button To Make Online Payments is also provided.

- Functional Requirements

  – This requires the use of some common Bootstrap framework to add the items as a card on the page and to display buttons.

4.6 Customer/NGO Payment

- Description

  – After Donating NGO Button, Page redirects to choose Mode Of Payment among the list.

- Stimulus/Response Sequence

  – All Payment Mode will be displayed in the radio button.

  – The User can choose anyone among them.

  – Button To Make Proceed is also provided.

- Functional Requirements

  – This requires the use of some common Bootstrap framework to add the modes on the page and to display buttons. After the button, we used Stripe API, to make online transaction and store Transaction Number.

4.7 Purchase/Order History

- Description

  – User can see the Order History of their profiles.

- Stimulus/Response Sequence

  – The content of the Order History will be shown in the card form with image at the left and other information at the right.

- Functional Requirements

  – This requires the use of some common Bootstrap framework to add order items in the card form.

5 Other Nonfunctional Requirements

5.1 Performance Requirements

The user should be able to query databases quickly and the results fetched must be appropriate. This can be done by finding the right balance between performance and accuracy by using SQLAlchemy.

5.2 Safety Requirements

The application runs on the web browser and hence harm to the user device is minimal, whereas a lot of data is to be read and written and hence the data storage and the server damage is possible but that too during heavy usage.

5.3 Security Requirements

The portal exchanges many critical documents between the users hence there is a multitude of safety requirements:

- A secure Login and Password for the users.
- A secured database system so that private information cannot be accessed by other people. Different storage spaces for the users so there is no information exchange between the users unwillingly

5.4 Business Rules

The software should not be outsourced to any third party without prior permission.
Maintaining more than one account or fake account is illegal.

6 Other Requirements

- Securely store the data and backup server to prevent any server crashes.
- A server capable of handling heavy workload and queries.
- Licensing requirements: Applicable.

6.1 Appendix A: Glossary

- Javascript : A Modern Multipurpose programming language.
- Flask : Flask is a micro web framework written in Python.
- SQLAlchemy: SQLAlchemy is an open-source SQL toolkit and object-relational mapper for the Python programming language

6.2 Appendix B: Analysis Models

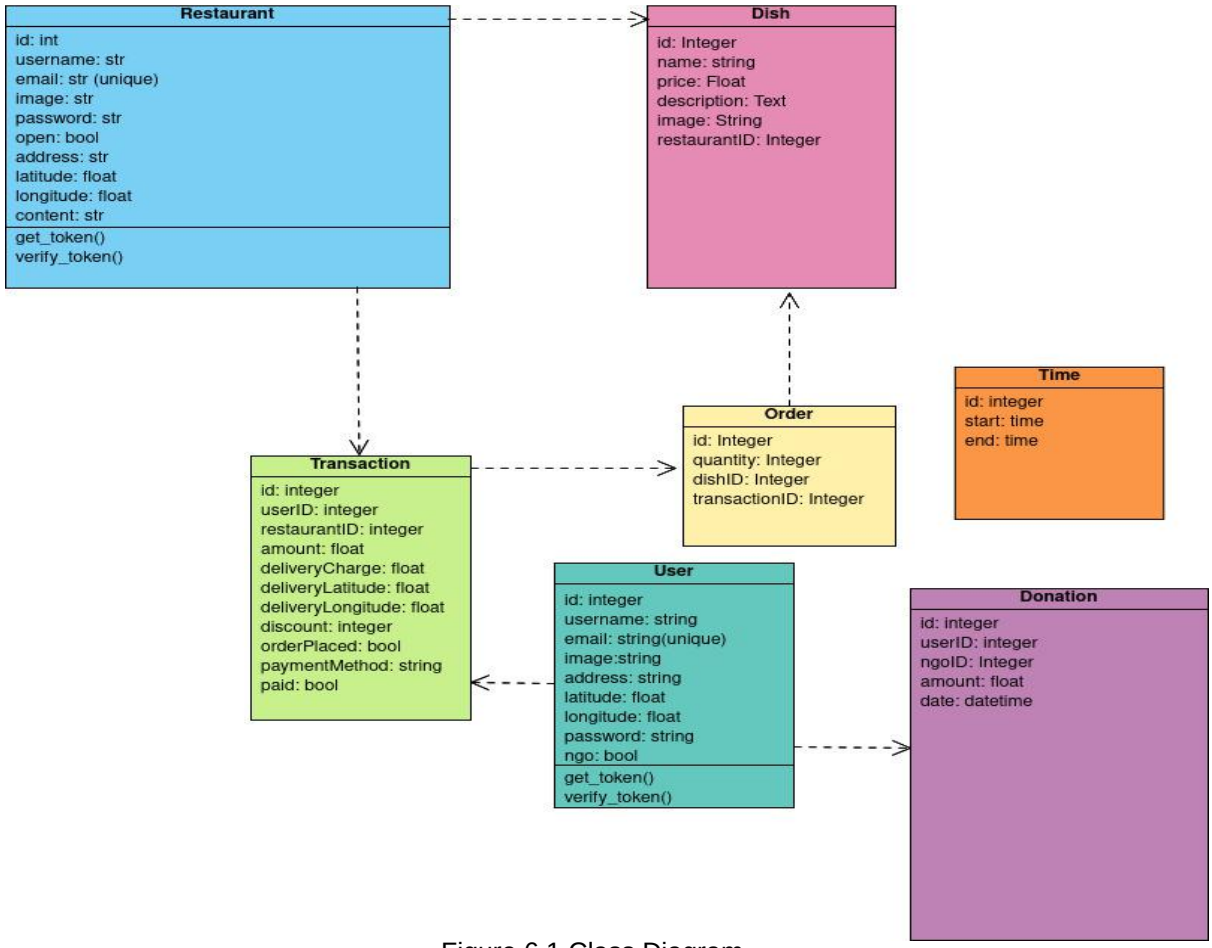

Figure 6.1 Class Diagram

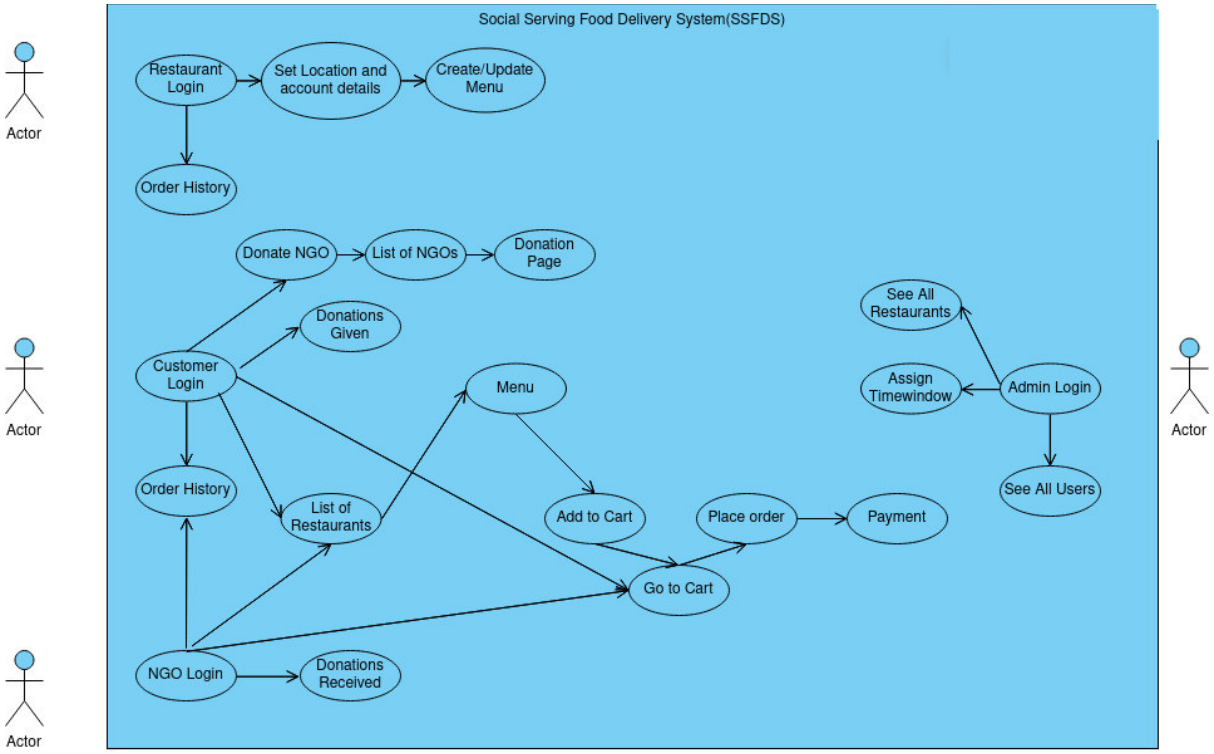

Figure 6.2 Usecase Diagram