Software Requirement Specification(SRS)
FEEDIE(food donation management system)

1 Introduction :

1.1 **Purpose of this Document :** The purpose of this Software Requirements Specification (SRS) document is to define the functional and non-functional requirements for the Food Donation Management System. This document outlines the features, functions, and constraints of the system.

1.2 **Scope of this document :** The Food Donation Management System is intended to automate the donation management process for non-profit organizations. The system will allow organizations to track donations, inventory, and distribution of food items. The system will also generate reports and provide analytics to help organizations make informed decisions.

1.3 **Overview :** A software requirements specification (SRS) is a document that describes what the software will do and how it will be expected to perform. It also describes the functionality the product needs to fulfill the needs of all stakeholders.

2 General description :

The purpose of this development is to limit the wastage of food in the Society. Many restaurants and people tend to throw the leftover food at the end of the day even though the food is perfectly fine to be eaten, which means that huge amounts of food is wasted.While all that food is being wasted, some families can barely afford proper meals with their limited money. They don't get enough nutrition due to the lack of having three meals in a day. Therefore, we decided to create our application to link the restaurant and people with the unfortunate people, so instead of throwing the food, the unfortunate will be able to pick it up from the restaurant and the people at the end of the day. The application allows the restaurants and people to log in, and upload an image of the meals they have as leftovers along with a description of that meal, and the location where to pick it up.

3 Functional Requirements :

Functional requirements include various aspects of the functioning of the system. They include :

- Enter mail-id and password and sign in.
- Different types of roles are as follows NGO , Food Soldier , Donor
- Raise a food request
- Update food details.
- Sign out.

4 Interface Requirements :

This android application provides the interface between food donors and those who require the food(NGOs , people in need). In this application, food donors enter their food quantity details and addresses.

1. User Interface:

The system should have a user-friendly and intuitive interface that is easy to navigate. It should have a clean and organized layout with easy-to-use menus, buttons, and forms.

2.Login Interface:

The system should have a login interface that allows users to securely log in with their username and password and also Google login. It should also have the functionality to reset passwords and retrieve forgotten usernames.

3.Donor Interface:

The system should have a donor interface that allows donors to create and manage donation requests. It should allow them to view the status of their donation requests.

5,Admin Interface:

The system should have an administrator interface that allows administrators to manage user accounts, approve or reject Hunger spot requests.

6.NGO Interface:

The system should have an NGO interface that allows NGOs to request for food in terms of number of meals.

7.Food Soldier Interface:

The system should have a Food Soldier interface that allows food soldiers to donate food and also to assist the process of donation.

5 Performance Requirements :

1.Response Time:

The system should respond to user requests within two seconds for most actions, such as adding, modifying, or deleting records

2.Concurrent Users:

The system should be able to handle at least 50 concurrent users without significant degradation in response time or system stability.

3.Data Processing Speed:

The system should be able to process and retrieve data quickly, even when dealing with large amounts of data. The system should be able to retrieve reports within 10 seconds, even for large datasets.

4.Scalability:

The system should be designed to handle a growing number of users, records, and data without sacrificing performance. The system should be able to scale up or down based on demand.

5.Fault Tolerance:

The system should be able to recover quickly in case of any failures or disruptions. The system should have a backup mechanism for data, and the recovery process should be automated.

6 Design Constraints :

The system must provide a capacity for parallel operation and system design should not introduce scalability issues with regard to the number of surface computers, tablets or displays connected at any one time.The system must be reliable enough to run crash and glitch free more or less indefinitely, or facilitate error recovery strong enough such that glitches are never revealed to its end-users.

7 Non-Functional Attributes :

Portability : The system will be portable and can easily fit on any platform like Android, ios, etc.

Reliability : Our application can work on any operating system.

Availability : The application will work all the time except the maintenance break.

Maintainability : Most of the information will be stored in the database and it would be maintained regularly.

Security : Our application will try to provide the best security. The organization will find the donor's phone number only after accepting the request.

User-friendly : It would be a very simple application. People of all ages and professions will be able to use it because of its simple UI.

Performance : The application will work fast and smoothly.

Efficient : We have used a few data to maintain efficiency. The system won't lag.

Safety : Data in the database of the system should not be lost or damaged.

Privacy : Personal data will not be disclosed to anyone.

8 Preliminary Schedule and Budget :

BUDGET : NIL
SCHEDULE :

- Requirements Gathering and Analysis: 2 weeks
- System Design: 2 weeks
- Development: 6 weeks
- Testing and Quality Assurance: 2 weeks
- Total: 12 weeks