Software Requirements Specification (SRS)
for
Excess Food Sharing Application

Version 1.3

Prepared by

Abdullah Kasem Saad	202220203
Ahmed Mohamed Attia	202220083
Samir Atef Samir	202220196
Ahmed Hussein Ahmed	202220164

Instructor: Mohamed Reda

Course: Software Engineering

Teaching Assistant: Mohamed Karam

Table of Contents

Contents

1. Introduction

1.1 Purpose

The purpose of this document is to define the software requirements for the Excess Food Sharing Application. This application allows users to share excess food with neighbors or local charities, helping to reduce food waste in communities. This document outlines the necessary functionality, design constraints, and performance requirements for the system.

1.2 Document Conventions

This document follows standard IEEE SRS formatting and includes sections for functional and non-functional requirements.

1.3 Intended Audience and Reading Suggestions

This document is intended for:
- Developers
- Project Managers
- Quality Assurance Team
- Business Stakeholders

1.4 Product Scope

The Excess Food Sharing Application is designed to reduce food waste by enabling users to post and browse available food items in their local area. Users can share food with neighbors or donate to local charities. The application aims to facilitate local connections and minimize food waste.

1.5 References

- Lectures
- Search Engine
- Chat GPT

2. Overall Description

2.1 Product Perspective

This application will serve as a community-focused platform to minimize food waste. Users can create food listings and browse others' listings within a specified radius. Charities can also sign up to receive food donations.

2.2 Product Functions

Main functionalities include:
- User registration and login
- Food listing creation and management
- Search and filter for nearby food listings
- Notifications for nearby food offers
- Profile and settings

2.3 User Classes and Characteristics

User classes include:
- General Users: Individuals looking to share or receive food
- Charities: Organizations seeking food donations
- Administrators: Platform moderators with permissions to manage user content

2.4 Operating Environment

The application will operate on iOS and Android mobile devices and will require an active internet connection for full functionality.

2.5 Design and Implementation Constraints

The application must comply with local data privacy regulations and follow secure coding practices.

2.6 Assumptions and Dependencies

The application assumes users have internet access. The platform relies on geolocation data to filter listings.

3. External Interface Requirements

3.1 User Interfaces

The application should have a user-friendly mobile interface with features for posting, viewing, and searching food listings.

3.2 Hardware Interfaces

The application will be available on smartphones with GPS functionality.

3.3 Software Interfaces

The application will use Google Maps API for location services and a secure payment API for optional donations.

3.4 Communications Interfaces

The application will support push notifications for alerts on new food listings nearby.

4. System Features

4.1 User Registration and Authentication

Users must be able to register using an email and password or social media login. Authentication will be required to post or claim food listings.

4.2 Food Listing Creation and Management

Users can create listings for excess food, including descriptions, expiration dates, and photos.

4.3 Search and Browse Listings

The application will allow users to search for nearby food items and filter listings by type, distance, and freshness.

4.4 Notifications

The application will send notifications to users for relevant new listings or reminders about posted items.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The application should support up to 1,000 concurrent users and load pages within 2 seconds.

5.2 Security Requirements

The application will use encryption for all sensitive data and require secure login methods.

5.3 Software Quality Attributes

The application must be reliable, with uptime of 99.9%, and intuitive for new users.