Software Requirements Specification

for

Donation Collection Application

Version 1.0 Approved

Prepared by

Uzair Khan K152183

FAST-NUCES

September 23, 2017

Contents

Revision History

Name	Date	Reason for Changes	Version

1 Introduction

1.1 Purpose

This documents aims to specify the requirements of a donation collection application. The current revision number is 1.0 .The product scope includes features such as finding the nearest collection centers for donation, allowing users to be volunteer collectors, and issuing alerts and donation requests to all users in case of an emergency.

1.2 Document Conventions

No special document conventions.

1.3 Intended Audience and Reading Suggestions

This document is intended for developers and project management. It contains specifications regarding the requirements of this project. It can be read in any sequence that the reader desires. Sections 1 and 2 provide a general overview of the requirements, with the later sections delving into more detail.

1.4 Product Scope

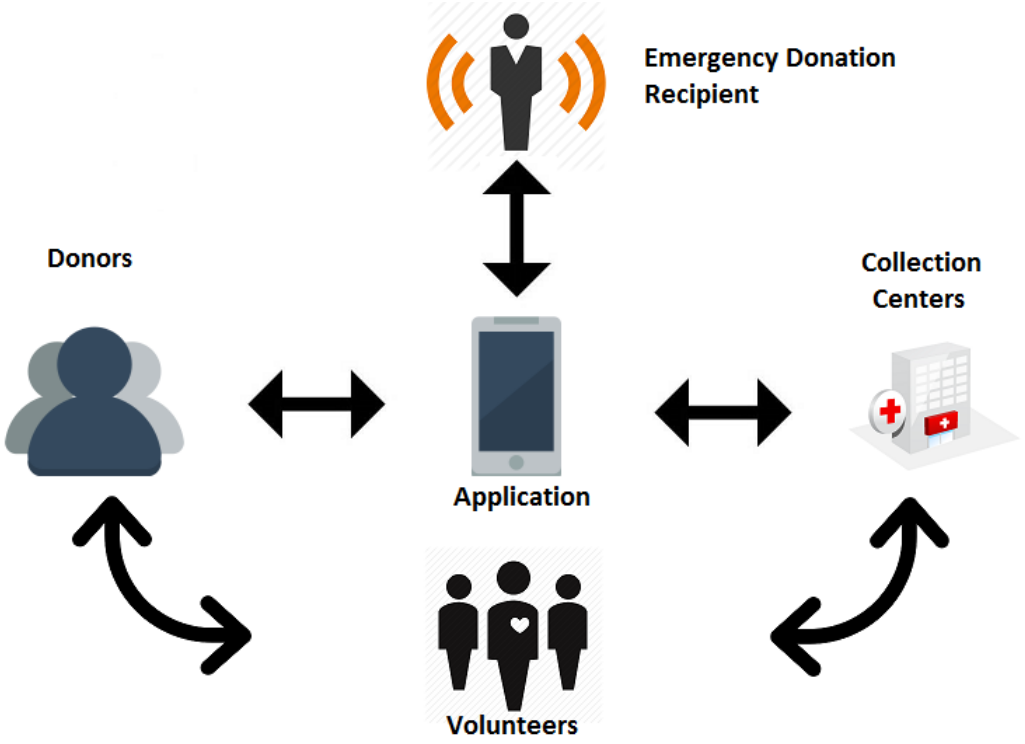
This project aims to implement a donation collection system. The idea is to connect willing donors with donation collectors to implement an efficient system that allows for recurring donations.

1.5 References

This document does not refer to any other document. It is self-contained.

2 Overall Description

2.1 Product Perspective



2.2 Product Functions

1. End-Users
    - Locate nearest donation centers.
    - Enlist to be a volunteer.
    - Request a volunteer collector.
    - Request and receive emergency donations.
2. Administrators
    - Verify emergency donations
    - Send system-wide alerts and notifications

2.3 User Classes and Characteristics

1. End-Users of this application are expected to have a diverse background of technological expertise. Since there is no barrier of entry for donation besides owning an Android device, users are expected to have the capability of operating an Android device. A small subset of users are expected to be enlisted as volunteer collectors. Additional data of such users will be stored.

2. Administrators are expected to be those who maintain this product, as well as a small subset of users who have increased privileges within the system such as that of issuing system-wide alerts. These users will have increased security and privilege levels. Administrators

2.4 Operating Environment

The project will use Android as the targeted platform. MySQL will be used for the database. A webpage is expected to be created using HTML for administrative users.

2.5 Design and Implementation Constraints

- Specific technologies: MySQL and Android.
- Hardware: Older Android platforms targeted(Android 4.2).
- Interfaces: A PHP script will be required for the Android application to communicate with the database.
- Security Considerations: User data is stored securely and ensuring that user authentication is safe.

2.6 User Documentation

The application is expected to be simple enough that additional tutorials will be unnecessary.

2.7 Assumptions and Dependencies

No such dependencies on external factors are present in the project.

3 External Interface Requirements

3.1 User Interfaces

Screen Limitations are the size of a typical Android devices.

3.2 Hardware Interfaces

Supported device types for end-users are Android devices. For Administrators, a server to host the database is required. An interface for Administrators to interact with the database will be included.

3.3 Software Interfaces

The Android application will be utilizing the Google Maps API. It will be interacting with a MySQL database with a PHP script being used for communication between the two systems.

3.4 Communications Interfaces

HTTP for communication over internet and IP protocol for intranet communications. Google Maps API will also be used.

4 System Features

4.1 Locating nearest collection center

4.1.1 Description and Priority

- High Priority
- User requests the location of the nearest collection center.

4.1.2 Stimulus/Response Sequences

The user will select the type of donation that he wishes to donate. The application will then use the users current location to determine the nearest collection center that will fulfill the user's request. Additional application response: The user can request a volunteer to come and collect the donation. The user will be able to track the volunteer on their phone and vice versa.

4.1.3 Functional Requirements

Functional requirements of this feature is the implementation of the Google Maps API that will allow the application to track and display the nearest point of interest according to the user. For the execution of the user case, a working internet connection is required on the user device. In the absence of an internet connection, the application will display an error message and request the user to select a working internet connection.

4.1.4 Use Case

- Priority: High
- Actors: User. Volunteers.
- Summary: Showing the user their current location and there intended destination based on what they want to donate. Volunteer users can also come pick up the donation if possible.
- Preconditions: The user has a working internet connection on their device and wants to find a collection center.
- Postconditions: The user has arrived at the destination based on the location provided through the application, or the donation was collected by a volunteer.
- Includes: Select Donation Type, Show Location
- Extends: Request Volunteer, Cancel
- Main Flow:
    1. The user selects the type of donation he wants to do from the main screen.
    2. The user views the nearest centre with respect to their current location.
       Exception: 1
       Alternative Path: 1, 2

3. The user reaches the destination.

- Alternative Path:

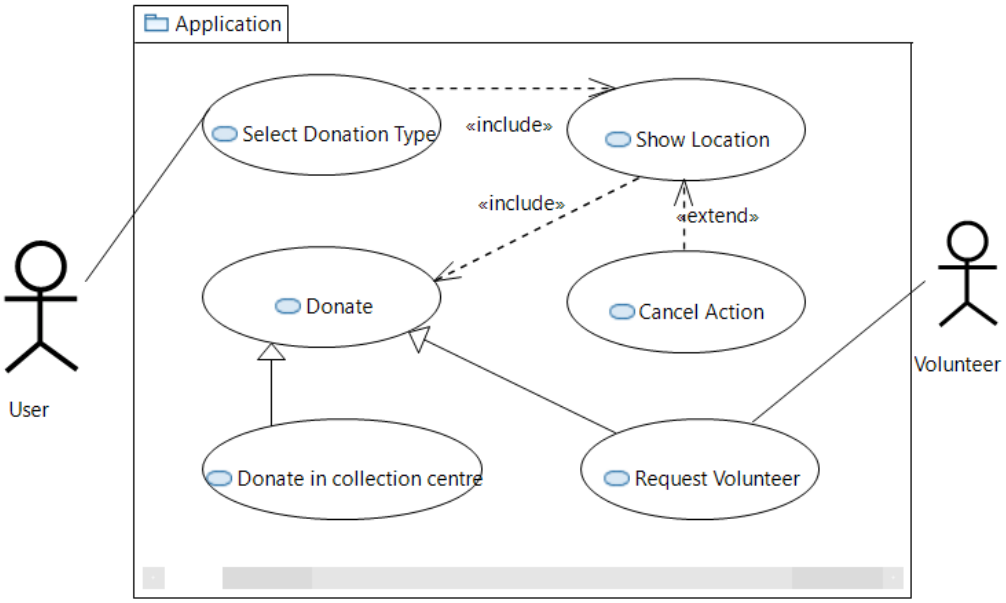
1. The user requests a volunteer.
   (a) The user specifies the details of the donation they wants to make.
   (b) The system checks whether the donation specifications are viable enough to send a volunteer for.
   (c) If the donation is not viable, the user is prompted to increase the donation specifications, thus restarting this alternative flow.
   Alternative Path: 2
   (d) If the donation is viable, a volunteer is dispatched to the user's location.
   (e) The volunteer collects the donation and takes it to the collection centre.
2. The user cancels the donation, ending this use case.

- Exceptions:

1. The user has no internet connection, and this use case is terminated.

- Assumptions:

1. A volunteer is present to collect the donation if required.
2. There are donation standards present which must be met to dispatch volunteers.



4.2 Enlisting to be a volunteer

4.2.1 Description and Priority

- Medium Priority
- User requests the service to grant them volunteer status.

4.2.2 Stimulus/Response Sequences

The user will select the option of enlisting as a volunteer. On doing so, the application will prompt the user on entering additional data such as the user's availability hours, their blood type and their method of transportation. On successful entry, the application will either grant or reject the user request depending upon the situation.

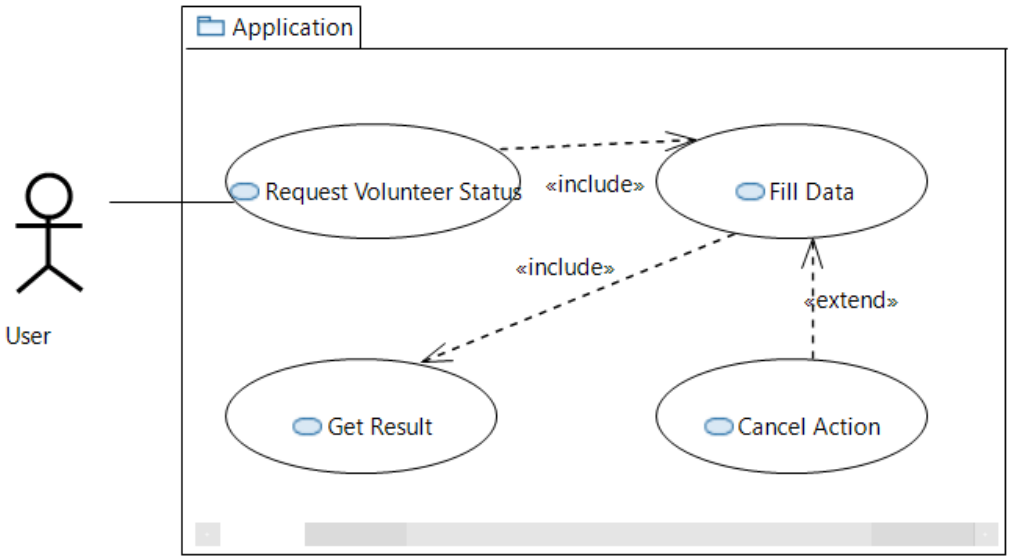
4.2.3 Functional Requirements

A working internet connection is required on the user device. Absence of an internet connection will prompt an error message. Software requirements of this feature are basic as it is merely the querying and updating of stored data in the database.

4.2.4 Use Case

- Priority: Medium
- Actors: User.
- Summary: Approving or rejecting a user's request to become a volunteer. The outcome of the user request depends upon various factors.
- Preconditions: The user has a working internet connection on their device and wants to become a volunteer. The user must be capable to perform volunteer duties.
- Postconditions: The user's application to be a volunteer is either accepted or rejected.
- Includes: Request Volunteer Status
- Extends: Cancel
- Main Flow:
    1. The user selects the option of becoming a volunteer from the main screen.
    2. The user is requested to fill data regarding their blood type, their method of transport, their area of residence and their available working hours.
       Exception: 1
       Alternative Path: 1
    3. The system processes the user's information.
    4. Volunteer status is granted to the user if the user's input data is satisfactory according to the system's requirements. The request is rejected otherwise.

- Alternative Path:
    1. The user cancels the request and does not enter the requested data, ending this use case.
- Exceptions:
    1. The user has no internet connection, and this use case is terminated.
- Assumptions:
    1. There is a volunteer criteria according to which a user's request will be judged.



4.3 Requesting emergency donation

4.3.1 Description and Priority

- Low Priority
- User requests the application service to issue a donation request on the user's behalf.
- Emergency donation requests can also be issued by administrators in the event of a genuine emergency/danger situation.

4.3.2 Stimulus/Response Sequences

The user will select the donation request button on the main screen. The application will prompt the user to enter preliminary data into the application to start the process. Upon successful data entry, the application will, if the data entered seems genuine, request the, user to meet with a verified donation collection administrator.

Upon verification of the user's case, that administrator can issue a donation request notification to all application users on the user's behalf.

Administrators also have the privilege of issuing donation requests if there is an emergency/danger situation in the city.

4.3.3 Functional Requirements

For the initial request and data entry, the application will require a working internet connection. Absence of an internet connection will prompt an error message. Software requirements of this feature are basic as it is merely the querying and updating of stored data in the database. User case assumes that there is a verified donation collection administrator present for user verification.

4.3.4 Use Case

- Priority: Low
- Actors: User, Administrator
- Summary: Approving or rejecting a user's request for emergency donation reception. The outcome of the user request depends upon user situation verification. Approval of user request will send a notification to all application users. Administrator's can also send emergency donation requests to all users in case of an emergency/danger situation.
- Preconditions: The user has a working internet connection on their device and wants to request a donation. There is an administrator present who is capable of verifying a user's case.
- Postconditions: The user's application for a donation is either accepted or rejected. If accepted, a notification is sent to all users.
- Includes: Request Emergency Donation, User Verification
- Extends: Issue Notification, Cancel
- Main Flow:
    1. The user selects the option of requesting a donation from the main screen.
       Alternative Path: 1
    2. The user is requested to fill data regarding their donation requirement such as blood type in case of a blood donation, cash required in case of a financial donation etc.
       Exception: 1
       Alternative Path: 2
    3. The system processes the user's information.
    4. If the user case seems genuine, a donation collection administrator is dispatched in coordination with the user to verify the user's case.
       Alternative Path: 3

5. If the donation collection administrator verifies the donation request as genuine, a notification is sent to all users about the donation request. Alternative Path: 4

- Alternative Path:

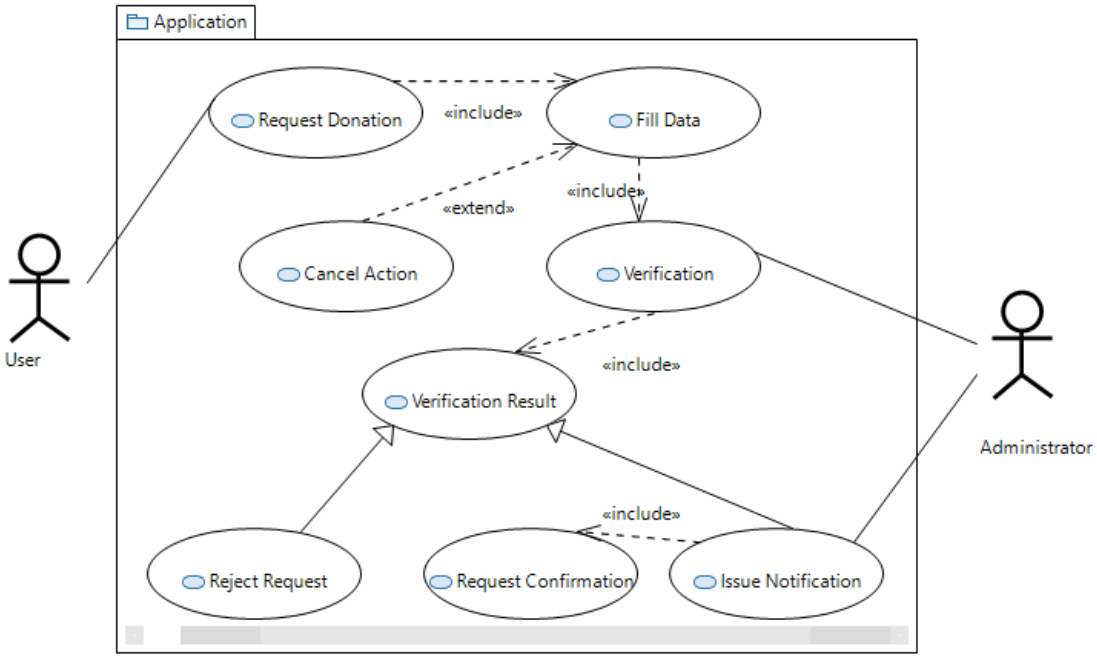
1. An administrator will send emergency donation requests to all users in case of an emergency/danger situation.
2. The user cancels the request and does not enter the requested data, ending the use case.
3. The user request does not meet certain criteria and is rejected, ending the use case.
4. The user request does not seem genuine according to the donation collection administrator and is rejected, ending the use case.

- Exceptions:

1. The user has no internet connection, and this use case is terminated.

- Assumptions:

1. There is a donation request criteria according to which a user's request will be judged.
2. There is a donation collection administrator present to verify a user's case if required.



5 Other Nonfunctional Requirements

5.1 Performance Requirements

Viewing the users location and nearest collection spot or volunteer is dependent on the internet connection of the user. Beyond that, the application should be as responsive as possible. As the application does not contain any heavy computations or factors which can increase time required, it is expected that the application will run well on older devices as well.

5.2 Safety Requirements

It is not expected that the usage of this application can result in user danger.

5.3 Security Requirements

As the application deals heavily with user data and location, certain security requirements must be fulfilled. Communication between devices over the internet will be using secure protocols. Data will be protected via passwords and safeguards.

5.4 Software Quality Attributes

Qualities of the product that the users would prefer are

- Reliability: The product must be robust and reliable enough that it will not crash when the user uses it.
- Availability: As the application requires relatively little hardware requirements, users will appreciate the availability of the application on older devices.
- Ease of Use: The application will be easy to operate.
- Correctness: The application will process user queries efficiently and without errors.

5.5 Business Rules

- Administrators: Can issue emergency alerts in case of an emergency situation that has occurred.
- Administrators: Can elevate donation requests to higher priority if the situation is extremely critical.

6 Other Requirements

Any requirements that the project needs have already been covered.

A Appendix A: Glossary

B Appendix B: Analysis Models

C Appendix C: To Be Determined List