**System Requirements**

**P11: Blood Donating System**

**Muhamad Bilal 22100224**

**Ahmad Aslam 22100221**

**Muhammad Saad Azam 22100082**

**Table of Contents**

1. Introduction 3
2. System Actors 4
3. Functional Requirements 7
4. Non-functional Requirements / Quality Attributes 9
5. Who Did What? 12
6. Review checklist 12

# Introduction

Blood Donating System is an Android, iOS and web-based app that provides a platform to blood donors and receivers for successful blood donation. It is going to make blood donation accessible to the one in need. The potential users of the app are blood donors, NGOs and blood recipients.

The overall objectives of the app are:

* Expedite the process of blood donation.
* Encourage NGOs to provide blood donors to the system.
* Through a system of ratings, points and rewards, encourage more and more donors to use the app.
* Provide an easy communication between blood donors and blood receivers.
* Make blood donation easily accessible to the one in need.

In the app, a recipient is going to ask for a donation of blood. The request will be pushed to a newsfeed where a donor will willingly respond to the request. A chat module will open between the donor and the receiver where they can communicate regarding the blood donation. On successful blood donation, the donor will be rewarded with points.

The app is going to be android, iOS and web-based. The main technologies that we are going to use for our app are Ruby on Rails, React and React Native.

# System Actors

|  |  |  |
| --- | --- | --- |
| **Actor** | **Description** | |
| Admin | Admin has full knowledge and control of the system. Admin acts as a moderator and has direct access to the whole system. | |
| **Functionalities** | | |
| Admin can:   * Register NGOs * Authorize NGOs * Sign in to his/her account * View feedbacks/complaints/reports regarding the app * View logs in order to detect any system failure or error * Log out of the app * Make internal changes in the app. | | |
| NGOs | NGOs will provide blood donors to the system which will in result assist in blood donation. In result of their services, they will receive more points, ratings and popularity in the app. | |
| **Functionalities** | | |
| NGOs can:   * Register their accounts * If they are registered, sign in their accounts * Provide data about their blood donors to the systems and provide its blood donors their login details. * Create, Read, Update and Delete data of their blood donors * Notify blood donors about blood donations request according to their eligibility and location * Respond to association requests if a donor wants to join that NGO * View user activity of their blood donors * Edit their own profile * If they want to drop their services, request the admin to remove them from the system | | |
| Blood donors | Blood donors are the actual main service provider to our system. They are going to donate blood to someone in need of blood. On fulfilment of a request, they will receive rewards and points. | |
| **Functionalities** | | |
| Blood donors can:   * Sign up their accounts * If they are registered, sign in their accounts * Respond to blood requests on newsfeed * Chat with a blood receiver whose request they have accepted * View/Edit their profile * Request an NGO to join with it * Look up NGOs according to their names * Log out of their accounts * View requests of blood receivers | | |
| Blood receiver | | They are the beneficiary of our service as they are the one who are in need of donation of blood. |
| **Functionalities** | | |
| Blood receivers can:   * Sign up their accounts * If they are signed up, sign in their accounts * Push notification for blood request on newsfeed * Chat with blood donors * View/Edit their profile * View notifications regarding the blood donors who have accepted their request | | |

# Functional Requirements

|  |  |  |
| --- | --- | --- |
| **Sr#** | **Actor** | **Requirement** |
| 1 |  | There will be login/logout system for users (admin, blood donors, blood receivers). |
| 2 | System will show updated news feed every time new request is made. |
| 3 | System will send push notifications to nearest donors of recipient, depending upon location. |
| 4 | System will automatically give points to NGOs and blood donors, on handling each request depending upon reviews/feedback/ratings. |
| 5 | **As an admin,** | I want to register and authorize NGOs to check authenticity of NGO. |
| 6 | I want to view feedback, complaints and logs to make constructive updates. |
| 7 | I want to notice any system failure/error to revise system issues. |
| 8 | I want to monitor users (NGO, blood donors, blood receivers) activity for security constraints. |
| 9 | I want to make internal changes in app, for any updates if required. |
| 10 | I want to handle termination requests of NGOs to clear their data. |
| 11 | **As an NGO,** | I want to CRUD blood donors so that I can handle updated blood donor data depending upon their availability. |
| 12 | I want to send login details to my registered blood donors so that they can respond to blood donation requests. |
| 13 | I want to respond to Onboarding requests, to handle associations by analyzing their health report. |
| 14 | I want to view user activity of my associated blood donors, so that check their eligibility for blood donation. |
| 15 | I want to view/edit my user profile to keep my profile up-to-date. |
| I want to request admin to terminate my account in case I want to end my services for the app. |
| 16 | **As a blood donor,** | I want to see blood donor information and respond to blood request on newsfeed/notification to respond blood requests. |
| 17 | I want to chat with blood recipient to stay in touch with the recipient. |
| 18 | I want to view and make changes to my profile to keep it up-do-date. |
| 19 | I want to look up for an NGO if I want to associate myself to it. |
| 20 | **As a blood recipient,** | I want to put a request for blood depending upon location. |
| 21 | I want to chat with blood donor to communicate about further details. |
| 22 | I want to view and edit my profile for any updates, if required. |
| 23 | I want to see notification of accepted blood requests. |
| 24 | I want to see blood donor information, when donor accepts the request. |
| 25 | I want to give feedback/reviews/rating to blood donor. |

# Non-functional Requirements / Quality Attributes

|  |  |
| --- | --- |
| **Sr#** | **Requirements** |
| 1 | App will be available for Android and IOS users. Moreover, web-based App will also increase the availability of the system. Moreover, MTBF (Mean Time Between Failures) will be more than 48 hours which means that system should not fail more than 4 times a week and MTTR (Mean Time To Repair) will be less than 5 hours which means system should be operational within 5 hours after the failure is detected. |
| 2 | Download size of the Application will not be more than 150 Mbs. |
| 3 | Any requested page should not take more than 7 seconds to load with average internet speed of 10 Mbps. |
| 4 | Upon entering the correct login details, user should be able to access the app within 5 seconds. |
| 5 | System should be scalable which means it should be capable of handling at least 10000 users simultaneously. |
| 6 | System will force user to create strong password, at least 8 character long containing combination of alphabet, number and character, to ensure account security. System will use Password authenticator to determine the strength of the given passwords based on the number of alphabets, characters and digits and display pop up message accordingly. |
| 7 | System will display correct and understandable error messages to user if user is doing something wrong. For Example: entering wrong blood group. |
| 8 | Any personal data of the user sent to server will be encrypted. This will ensure that the data is not usable in case of any data theft. |
| 9 | System will be able to correctly declare log about module crashing. |
| 10 | System will be robust against cyber security attacks. Input validation attacks like SQL injection, Buffer overflow and XSS attacks will be prevented using centralized validation approach rather than relying on client-side validation. |
| 11 | Current state of the database will be backed up bi-weekly and there will also be option for emergency backup in case on any security threat or hardware failure. |
| 12 | Reusability of the code will be ensured with different techniques like modularity, high cohesion and loose coupling. |
| 13 | Since the users of the app will be average internet user, usability of the system will be given prime importance. An average user will be able to post, search and create blood donations request within 5-steps. Moreover, use of consistency standards will help understanding the content easily but help and documentation will also be available. |
| 14 | For the early stages of the software launch portability of the software will be limited to only Android, IOS apps and chrome web-app. Later, the support for other web browsers like Safari and Firefox will be available. |
| 15 | Chat between the blood donor and blood receiver will be encrypted so that the users feel safe while sharing contact information in the chat. |

# Who Did What?

|  |  |
| --- | --- |
| **Name of the Team Member** | **Tasks done** |
| Muhammad Bilal | Functional Requirements |
| Ahmad Aslam | Non-Functional Requirements |
| Muhammad Saad Azam | Introduction, System Actors |

# Review checklist

|  |  |
| --- | --- |
| **Section** **Title** | **Reviewer Name(s)** |
| Introduction | Muhammad Saad Azam |
| Actors | Muhammad Saad Azam, Muhammad Bilal |
| Functional Requirements | Muhammad Bilal, Ahmad Aslam |
| Non-functional requirements | Ahmad Aslam, Muhammad Saad Azam |