**TABLES OF CONTENTS**

[LIST OF FIGURES](#_2486me7g86pq) 3

[LIST OF TABLES](#_gjd2forb07lv) 4

[LIST OF ABBREVIATIONS](#_mucgg0elghk) 5

[**CHAPTER 1: INTRODUCTION**](#_cfbxzfwzsagg) **6**

[1.1. Overview](#_v9a8eq4ic3ij) 6

[1.2. Problem Statement](#_q95si2fh0xo3) 7

[1.3. Objectives](#_kdowziffqp8d) 7

[1.4. Methodology](#_eee85797foqp) 8

[1.5. Feasibility Analysis](#_2zj01u3fub1c) 8

[1.5.1. Technical Feasibility](#_lf5uqpzh1zni) 9

[1.5.2 Economic Feasibility](#_hnep7gbxl4p) 9

[1.5.3 Operational Feasibility](#_axnn8uwm3o0q) 9

[1.5.4 Schedule Feasibility](#_96pjwnov2nve) 9

[1.6. System Requirements](#_2gxbgw3uq0kj) 10

[1.6.1. Development Side](#_z27d86309fys) 10

[1.6.2. Client Side](#_4buznxrhk5nv) 10

[1.6.3. Server Side](#_symthjhq94i2) 11

[**CHAPTER 2: LITERATURE REVIEW**](#_440pv5q1oqs4) **12**

[2.1. Background](#_798gm22j0pb3) 12

[2.2. Review of Related Works and Tools Used](#_9szpunmf8ujc) 13

[**CHAPTER 3: SYSTEM DEVELOPMENT**](#_kzvxylbk33hj) **15**

[3.1. Project WorkFlow and Schedule](#_rb08bpue7wr) 15

[3.1.1. Project Team](#_r6jifzyav372) 15

[3.1.2. Responsibilities of supervisor and team member](#_qm0j0fwx55s1) 16

[3.1.2.1. Responsibilities of supervisor](#_3t1xugy8kxdq) 16

[3.1.2.2. Responsibilities of team member](#_loj86lgzx2h8) 16

[3.2. System Analysis](#_p5yf2o2ftux8) 16

[3.2.1. Requirement Analysis](#_11tn89xzxfs7) 17

[3.3. Architecture Design](#_buqlhsvxw8ha) 19

[3.4.1. Data Design](#_pk7hka29wsml) 20

[3.4.2. Process Design](#_qze95b3gli2g) 21

[3.4.2.1. Context Diagram](#_l2fqkfru8yym) 21

[3.4.2.2. DFD](#_zeewmeuukmty) 22

[**CHAPTER 4: IMPLEMENTATION AND TESTING**](#_uhvtdoxhmvku) **23**

[4.1. Implementation:](#_5zw62fj983o0) 23

[4,1.1. Overview of Development Methodology:](#_ufbk4uxfhi73) 23

[4.2. Testing](#_wzjpx7ih01lk) 24

[4.2.1. Unit Testing:](#_zg9g63lf6v6f) 24

[4.2.2. Integration Testing](#_obvlucrwqx23) 27

[4.2.3. System Testing:](#_7gpcylmmmqht) 28

[**CHAPTER 5: MAINTENANCE AND SUPPORT**](#_rboc7s8uo0hc) **31**

[5.1. Maintenance](#_7ghm2ypdecuo) 31

[5.2. Support](#_kw2zh2jnu9hk) 31

[**CHAPTER 6: LIMITATIONS AND FUTURE ENHANCEMENTS**](#_gan6ce1a07zi) **32**

[6.1 Limitations](#_ebx17o85v6gb) 32

[6.2 Future Enhancement](#_1ofc63udn958) 32

[6.3 Conclusion](#_40cmjfclepqm) 32

[**CHAPTER 7: APPENDIX**](#_icwj9j70kf4j) **33**

[**REFERENCES**](#_x69xq1rsln7d) **41**

# 

# 

## LIST OF FIGURES

Figure 1: Gantt Chart 9

Figure 2: Use case Diagram 18

Figure 3: Architecture Design 19

Figure 4: Er-Diagram 20

Figure 5: System Context Diagram 21

Figure 6: DFD 22

Figure 7: V-Model of development 24

**Appendix**

Figure 8: App in drawer 32

Figure 9: App Log in/Sign up page 32

Figure 10: Login with email 33

Figure 11: Completing login with password 33

Figure 12: Login with Google 34

Figure 13: Main entry page 34

Figure 14: Request Blood Form 35

Figure 15: Donate Blood Form 35

Figure 16: Type of request feed 36

Figure 17: Request Feed 36

Figure 18: Donate Feed 37

Figure 19: Maps showing both requests 37

Figure 20: Image showing list of registered users’ info 38

Figure 21: Edit Profile Form 38

Figure 22: Send Feedback form 39

## LIST OF TABLES

### 

Table 1: Software Development Requirements 10

Table 2: Client-Side Requirements: 11

Table 3: Project TeamTable 15

Table 4: User registration 25

Table 5: User login test 1 25

Table 6: User login test 2 26

Table 7: Request Blood 26

Table 8: View the Request Feed 27

Table 9:Request Blood 28

Table 10: Check if system works or not in different conditions 28

Table 11: Check if users can see database of users 29

# 

## LIST OF ABBREVIATIONS

CPU Central Processing Unit

RAM Random Access Memory

SDK Software Development Kit

API Application Programming Interface

OS Operating System

IDE Integrated Development Environment

AVD Android Virtual Device

HTTP Hypertext Transfer Protocol

ERD Entity Relationship Diagram

DFD Data Flow Diagram

GB GigaByte

# 

# 

# 

# CHAPTER 1: INTRODUCTION

## 1.1. Overview

Blood is an essential fluid that keeps humans alive. It is the fuel that runs our bodies. Daily, there are cases in which blood transfusion is required, such as serious injuries from natural disasters and accidents, illnesses like anemia, leukemia. No other artificial substance can fulfill all the functionality of the true blood. Some artificial substitutes are available which are used for the fulfilment of some functions of biological blood [1]. Generous donors are the only source of true blood during the transfusion. The blood transfusion demands have outstripped the supply of the blood in the recent years. With the increasing population, the demand for blood is always uprising [2]. There are even more demand during the time of emergency such as accidents and natural disasters. Blood banks also face the acute shortage of blood during the festive season [3]. The scarcity of blood during such emergency may lead to the loss of many precious lives. A quick and effective way to call the donors immediately and arrange blood would be able to save many precious lives.

This is where the project entitled “Blood Bank” aims to function in order to minimize the problems related to the blood transfusion process. It is an Android Mobile application which acts as a mediator between the eligible donors and the patients. This application simplifies the search for the volunteer donors. The eligible donors are registered in the system database. People in need are able to search the blood donor database. They are able to contact the donors using various means. The search operation selects the best possible donors from all the donors and suggests those donors to the seeker. Best possible donors are filtered from the set of all the donors by analyzing their distance with the patient’s location. This application also has the records such as last blood donation performed by the user. It is helpful in notifying the users about their next eligible blood donation date. Further, this application helps to improve the participation in blood donation. According to Red Cross, people who are at least 16 years of age are eligible for blood donation [4]. Comparing this data with the latest census of Nepal, [5] it is estimated that around 60% of the total population are the eligible blood donors. The active participation of these eligible blood donors helps to solve the uprising scarcity of blood donation [6]. This application also provides informative materials related to the blood transfusion thus increasing the participation in blood donation programs.

## 1.2. Problem Statement

The existing blood bank management systems are specifically targeted for the management of the blood available in blood banks. They provide information about the available blood groups. Similarly, there are other systems that provide data about the available donors. The location of the donors is not considered in those systems. It is inefficient if the system shows all the available donors regardless of their location. If there are two donors, then selecting the nearest donor is the no brainer choice during the time of emergency. This application will show the donors on the basis of their location, and so now the user can be able to decide for themselves. The application will ask for donor’s address during registration. Later when the search is performed, the location is taken into account and the eligible donors will be sorted according to the nearest location.

Another problem is the flow of information during the time of emergency. Whenever there is the requirement of blood, people usually post it into social networking sites. Sometimes the same post are repeated multiple times by different users. This app will provide the platform to contact the specific volunteer donors. As the donors are contacted directly, it will help to manage the blood in a very effective way [6].

## 1.3. Objectives

The objectives of this project are enlisted below:

* To provide an online platform for the volunteer blood donors.
* To locate the nearest donor during the emergency and help in quick response.
* To keep track of the records of donors such as their last donation date, next possible donation date.

1.3.1 Significance of the Project

This project will be significant in addressing different issues of blood transfusion services. It will be helpful for minimizing the communication gap between the donor and seekers. As a result, this project will be helpful in saving precious lives during the emergency. Similarly, this project will act as an online platform where the interested donors can register themselves and help the victims during the need.

## 1.4. Methodology

The existing systems about Blood Bank Management system will be reviewed. The problems that exist in those systems will be analyzed and goals for further enhancement will be set. The secondary data for this project will be collected through the websites of Nepal Red Cross Society and Official portal of Government of Nepal [7].

Further data will be collected from the registered users. The users who want to participate in blood donation are required to register to the system database using their valid information. The data thus collected will be divided into different categories according to the blood type and location of the users. Similarly, the blood demands made in the system will be accounted. Those data will be used to analyze the trend of blood demands.

## 1.5. Feasibility Analysis

We checked if the project was feasible or not before started working on the project [8]. We checked different types of feasibility for e.g. Technical, Economic etc. Since, it seemed feasible to develop this project, we decided to give this project a go and develop a working system.

## 1.5.1. Technical Feasibility

BloodBank for Android system is to be developed for Android devices as the name suggests, we are going to develop the software for mobile devices and for that purpose we require knowledge of Java programming language, a powerful enough computer to run Android Studio and knowledge of Firebase technologies and their APIs.

We have knowledge of Java, and computers powerful enough to handle Android Studio, the project seems Technically feasible [9].

## 1.5.2 Economic Feasibility

The hardware tools we need to develop this project are already available to us, and the software tools that we are going to use are Android Studio and Firebase technologies which are free of cost. We also used Google Maps API, Geocoder API which is required to have a billing account but don’t require any money to use if the usage is under the cost of $200 [9].

## 1.5.3 Operational Feasibility

BloodBank for Android is designed to provide a way to connect people who need blood and people who want to donate. The software will have simple user interface so the user can navigate easily. It can be easily used by normal users who knows how interaction in touch screen works and how to navigate in those systems [9].

## 1.5.4 Schedule Feasibility

Gantt chart of the Android Blood Bank Application is follows:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Weeks/  Project Activities | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Planning |  |  |  |  |  |  |  |  |  |  |
| Designing |  |  |  |  |  |  |  |  |  |  |
| Coding |  |  |  |  |  |  |  |  |  |  |
| Testing |  |  |  |  |  |  |  |  |  |  |
| Implementation |  |  |  |  |  |  |  |  |  |  |

Figure 1: Gantt Chart

The schedule of the Android Blood Bank Application Project has six major activities [10]. Planning and design is scheduled to be completed within the first week of starting the project. Development phase starts in second week. V model is selected for system development. So, development, evaluation and review will continue in loop until seventh week. Testing is to be done in the eighth week and implementation is to be done in the ninth week.

## 1.6. System Requirements

There are some necessary specifications that are required for the smooth running of this application.

### 1.6.1. Development Side

Table 1: Software Development Requirements:

|  |  |
| --- | --- |
| Operating System | Windows 10 |
| Application | Android Studio |
| Programming Language | Java |
| Processor | Intel® Core™ i5-4200 CPU @ 1.60GHz |
| RAM | 4GB |
| Disk Space | 2GB |
| Internet Connection | At least 512 Kbps |
| Database | Firebase Realtime Database |

### 1.6.2. Client Side

The project is an Android based application that is going to be developed for mobile phones, Hence the user requires a smartphone and a decent internet connection.

Table 2: Client-Side Requirements:

|  |  |
| --- | --- |
| Operating System | Android |
| Android version | 5.0 or Above |

### 1.6.3. Server Side

Firebase Technologies will be used as the server(backend) for this application. We used Firebase Authentication for authentication purposes and Firebase Realtime Database for data storing purpose. Google Maps server will be used to implement maps features.

# 

# 

# CHAPTER 2: LITERATURE REVIEW

## 2.1. Background

The idea of taking blood from one individual to infuse it into another was developed by the ancient Egyptians. The word “transfusion” stems from the ancient Latin “transfundo”, which initially meant to pour from one vessel to another. Today, blood transfusion is one aspect of human solidarity, and millions of blood donations are made every year worldwide. Various blood components can be prepared from these donations, including fresh frozen plasma (FFP), platelet concentrates (PCs), red blood cell components (RBCCs), whole blood (WB), white blood cells (neutrophils), and blood-derived preparations. Since 2013, blood products have been included on the list of essential medicines established by the World Health Organization [11].

In the context of Nepal, blood donation was started by the Nepal Red Cross Society in the year 1996, i.e. 3 years after the inception of the Society itself. During the initial years the service was available only for the people of Kathmandu but over the years blood banks have been established in 68 places of 48 districts of the country. In the initial years the service was made possible through collection of blood from professional donors but since 1982 collection of blood was emphasized from voluntary non-remunerated donors only. In the meanwhile, serious efforts were made and are being made to collect blood from institutionalized sectors like colleges, universities, industries, clubs, governmental and non-governmental offices [12].

Recently, there have been various developments to promote blood donation campaigns using various online platforms such as Facebook. Nepal Red Cross Society has even collaborated with online apps and blood databases like Blood Donor Plus and Youth For Blood. These online platforms facilitate the users to sign up for voluntary blood donor membership, provide information about the upcoming campaigns, make requests for blood, track the blood donation records of the members, aware the youths about the benefits of blood donation. Despite these amazing features provided by the online platforms, something is still missing such as finding the nearest donor during the time of the emergency and integrating the blood request with the popular social media like Facebook, Twitter [13] [14].

During our survey, we found the maximum blood request on social media like Facebook, Twitter etc. Let’s be accurate, we found many requests on Facebook group named Men’s Room Reloaded and Blood on request, and we collected the data of blood request. During the survey we observed the problems of the general public who could not find a donor on the need and many people misuse the information. Many fake donor and requests were seen over the social media and that makes the availability of the blood quite difficult.

While working on literature survey, we studied many existing systems. Mostly there are many existing mobile applications which provide the services related to blood on request, donor tracker etc. E-Blood Donor is the mobile based application which only provide the information about the donor and receiver service where we can request for blood in any state and city of India [15]. Blood Member Nepal is also a mobile based application which asks for blood group and district in Nepal [16] and searches for available donors in the following district from Blood Donor Association Nepal (BLODAN)’s database [17]. Eblood is also mobile based application which stores the donor information such as location, blood group, email and last donated date [18]. Blood Nearby is mobile based android application which tracks donors and search for the nearby blood banks in certain radius [19].

## 2.2. Review of Related Works and Tools Used

* Java

There are two obvious choices for programming language when you want to develop an Android application that are recognized by Google itself. They are Kotlin and Java [20]. Since we were already familiar with the Java programming language, we opted for it. We used Firebase SDK and APIs along with Maps API codes that were available in Java in our application.

* Android

We were developing a software for Android devices so we needed a device running Android OS [21].

* Google Maps APIs

When we needed to implement location services we used Google Maps servers and their API to interact with them [22]. It is used to show requests in the map, and user to be able to interact with it.

* Android Studio

Android Studio is a tool based on Jetbrains technology provided by Google itself for Android developers [23]. Another choice can be Eclipse IDE. But we opted for Android Studio since we were familiar with Jetbrains products. And Android Studio provided support for AVD, where most of the running and testing of application was done..

* Git & GitHub

We used Git for version control management of our project and Github to store the git repository so our team were able to see the updates instantly without being present at the same location [24].

* Firebase Authentication

Firebase Authentication is used to authenticate users for this project[25]. It provides easier management of user and provides many services through which a user can authenticate themselves.

* Firebase Realtime Database

Firebase Realtime Database, a NoSQL database, is used to store data generated from this project.

* Geocoder API

Geocoder API is used to convert address into Latitude and Longitude value, which is used when searching for places when using Activities which include Google Maps [26].

# 

# CHAPTER 3: SYSTEM DEVELOPMENT

Systems development is the process of defining, designing, testing and implementing a new software application or program. It can include the internal development of customized systems, the creation of database systems or the acquisition of third party developed software [27].

## 3.1. Project WorkFlow and Schedule

* Team Size: 4
* Team Effective Project Duration: 10 weeks
* Effort Required per person: 35 hours per week

### 3.1.1. Project Team

The table 3 shows the various personnel associated with the project.

Table 3: Project Team

|  |  |
| --- | --- |
| **Resources** | **Role** |
| Bikash Balami | Supervisor |
| Aasutosh Kumar Jha | Developer/Designer |
| Bharat KC | Developer/Designer |
| Kishor Bam | Developer/Designer |
| Uttam Joshi | Developer/Designer |

### 3.1.2. Responsibilities of supervisor and team member

#### 3.1.2.1. Responsibilities of supervisor

* Project Planning
* Schedule Project
* Schedule Tracking
* Documentation monitoring

#### 3.1.2.2. Responsibilities of team member

* Background Research
* Design and analysis
* Development and Testing
* System Installation and Implementation
* Project Documentation and Report Submission

## 3.2. System Analysis

System Analysis is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components. System analysis is conducted for the purpose of studying a system or its parts in order to identify its objectives. It is a problem-solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish their purpose [28].

System Analysis has 4 main activities. They are:

* Software Specification
* Software Design and Implementation
* Software Validation
* Software Evolution.

### 3.2.1. Requirement Analysis

* Functional Requirement Analysis

We have gathered requirements of our proposed application which are depicted as given in the use case diagram in the next page[29]. We have two types of actors they are, Normal user and administrator. The Administrator has access to everything related to the application such as user database, required list etc. An administrator needs to login before he/she can make use of their privileges.A normal user also has to login before they can use the services that are available in the application. A user can make requisite for blood , view users database, view available request in the map.

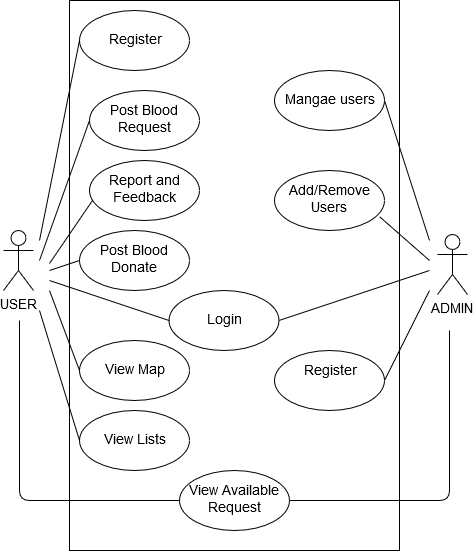
****

Figure 2: Use case Diagram

* Non Functional Analysis

We also gathered non functional requirements for our proposed application [30]. We will use firebase authentication system provided by google to authenticate user in our application. the firebase authentication system is robust and have almost zero down time.

We will use firebase realtime database to store different data produced from our application. It is synchronized between different instances of the application almost instantly so there will be very little lag.

## 3.3. Architecture Design

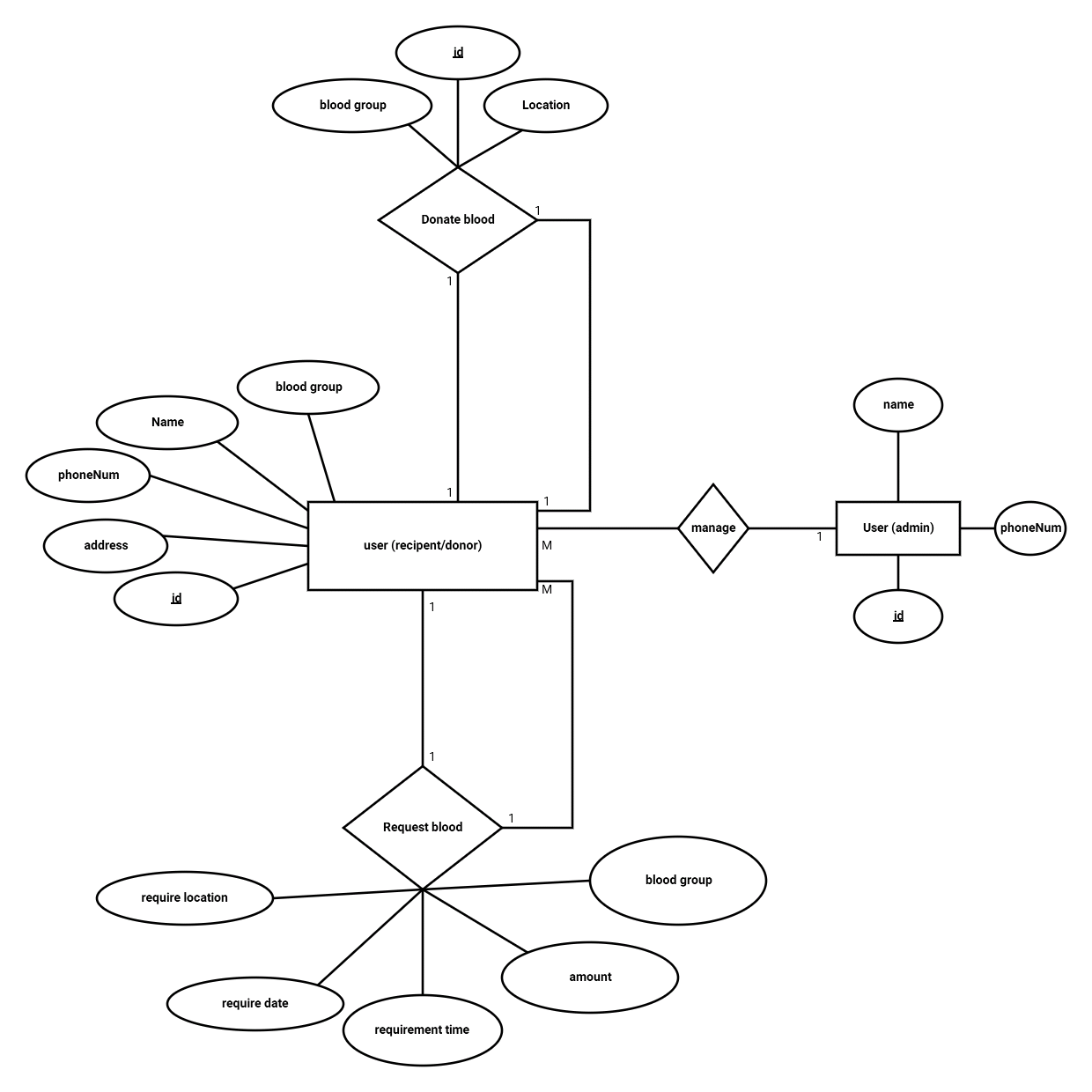
# 

Figure 3: Architecture Design

The figure shows the basic system architecture of Android Blood Bank application. The architecture consists of five major components. The user uses the android application through an android device to connect with the Firebase server. The android application interacts with the server by HTTP request- response method. Whenever we are required to work with location, the application shows an activity that requests map data from the Google Map server. The Google Map server returns the map data. The user can choose the location where request should be posted, and then the application stores the request data in Firebase Realtime Database using Firebase API.

3.4. System Design

### 3.4.1. Data Design

Figure 4: ERD

Blood Bank Android Application has four entities: user ,blood, donor and user(admin) where user has different attributes such as name, blood group ,phoneNum, address and id. Blood has attribute blood group, Donor has attributes id, name, phoneNum, blood group, address and admin has attributes such as name, id, phoneNum. The ERD clearly shows that user(receiver) requests for required blood, donor donates blood. Admin manages and maintains the registers data, and if needed, contacts the users.

### 3.4.2. Process Design

#### 3.4.2.1. Context Diagram

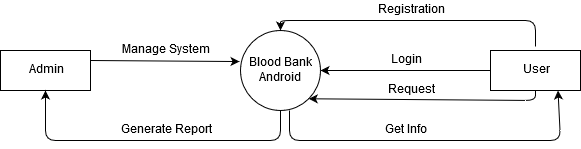
****

Figure 5: System Context Diagram

The diagram in figure shows the system context diagram. It describes the overview functionalities required by the external entities. A context diagram, sometimes called a

level 0 data-flow diagram, is drawn in order to define and clarify the boundaries of the software system. It identifies the flows of information between the system and external entities. The entire software system is shown as a single process [31].

In the above figure the entities are represented using the rectangle. User and Admin are the two entities in the system. The overall system is represented by a single process in the circle. The data flow is shown using the arrows.

#### 3.4.2.2. DFD

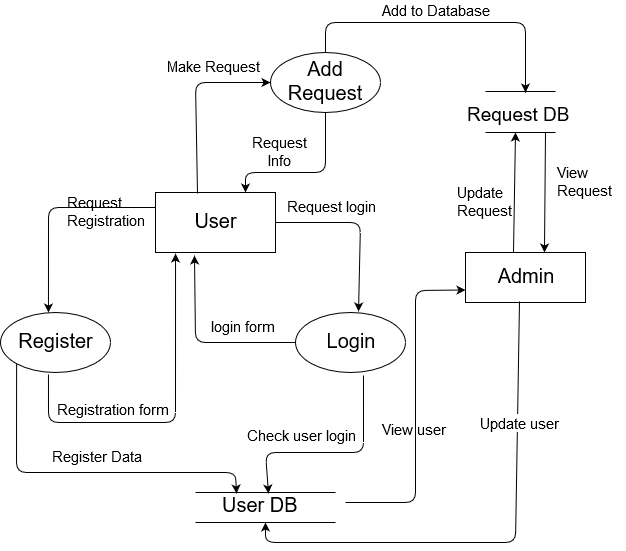


Figure 6: DFD

# CHAPTER 4: IMPLEMENTATION AND TESTING

## 4.1. Implementation:

Implementation phase is one of the important phases of project development [32]. In this phase

we implement our conceptual design into the working program by using various tools. The successful implementation of project is nearer steps towards the project completion. Project implementation was not an easy step to us as we encountered various issues related to the programming logic as challenges.

Our project is completely based on V model so if the requirement changes then there is no big deal in changing the system’s part. The application should maintain a database that contains the list of donors and requests which consists of fields: Name, Blood group, Location and Number if required. Whenever there is blood request, the software searches the database for the list of available donors based on the location (may be defined via hospital name/location).

Google map API is integrated with the system so that user can search nearby events.

### 4,1.1. Overview of Development Methodology:

System “Blood Bank” is developed based on V Model [33]. The first phase of V model is requirement analysis and gathering. After gathering the requirements feasibility study of the system was studied. After feasibility, we start designing the entire system as per the requirement. When the designing is done we start creating the module for each component. The creation of module is done in coding part. Once coding is done we start testing the individual modules and testing is known as unit testing. Unit testing of each component must be done and are completed. Some of the unit testing are registering the new user with form validation, login page and many more. Once unit testing is completed, we start integrating the systems that are linked with each other. Once the whole system is integrated we start the integration testing. For example, we register a member and also request the certain amount of blood. Once the integration testing is done we test the whole system and make sure whether overall system is working correctly or not. During the coding and testing phase we found many errors while conducting the project. To minimize such errors, V model is used for development as we can do the verification and validation of the system together.

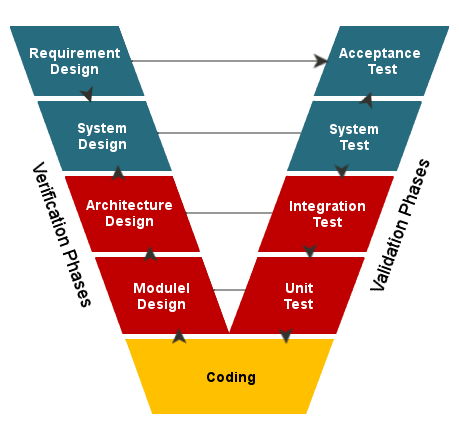


Figure 7: V-Model of development[34]

## 4.2. Testing

The testing phase can be carried out manually or by using automated testing tools to ensure

each component works fine[35]. After the project is ready we tested its various components in

terms of quality, performance to make it error free and remove any sort of technical jargons.

### 4.2.1. Unit Testing:

During the coding phase each individual module was tested to check whether it works

properly or not. Different errors found during unit testing were debugged. Some of the

major test cases are listed below:

Test Case I:

Table 4: User registration

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Steps | Test Step | Test Data | Expected  Results | Actual  Results | Status  (Pass/Fail) |
| 1. | Click on the Sign in with email button |  | App should show email entry form | App shows email entry form | Pass |
| 2. | Input valid email | Email: bkcraaj@gmail.com | Should accept valid email | Accepts valid email | Pass |
| 3. | Input first and last name | Name: Bharat KC | Should accept non empty name | Accepts non empty name | Pass |
| 4. | Input Password | Password: 111111 | Should accept password more than 6 characters | Accepts password more than 6 characters | Pass |
| 5. | Click on the save button |  | Should save data to database. | Saves data to database. | Pass |

Test Case II:

Table 5: User login test 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Steps | Test Step | Test Data | Expected Results | Actual Results | Status  (pass/fail) |
| 1. | Click on the Sign in with email button |  | App should show email entry form | App shows email entry form | Pass |
| 2. | Input Email | Email: bkcraaj@gmail.com | Should accept the email as it is already registered. | Accepts the email. | Pass |
| 3. | Input Password | Password:(left empty) | Should not accept invalid passwords. | Doesn’t accept. | Pass |
| 4. | Click on the Sign In button |  | Should not accept invalid passwords. | Can’t sign in. | Fail |

Test Case III

Table 6: User login test 2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Steps | Test Step | Test Data | Expected Results | Actual Results | Status  (pass/fail) |
| 1. | Click on the Sign in with email button |  | App should show email entry form | App shows email entry form | Pass |
| 2. | Input Email(Valid Email) | Email: bkcraaj@gmail.com | Should accept the email as it is already registered. | Accepts the email. | Pass |
| 3. | Input Password(Valid Password) | Password: 111111 | Should accept the valid password | Accepts the password | Pass |
| 4. | Click on the Sign In button |  | Should Log in since password is correct. | Successful Login | Pass |

Test Case IV:

Table 7: Request Blood

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Steps | Test Step | Test Data | Expected Results | Actual Results | Status  (pass/fail) |
| 1 | Click on request blood |  | Request form should be displayed | Shows the form | Pass |
| 2. | Enter full name | Bharat KC | Should accept valid name | Accepts | Pass |
| 3. | Input the amount of blood required. | 2 pints | Should accept valid amount | Accepts | Pass |
| 4. | Input Phone number | 9876543210 | Should accept valid number | Accepts | Pass |
| 5. | Click on post my request button |  | Entering valid data in form should post the data to database. | Posts the data | Pass |

### 4.2.2. Integration Testing

Integration testing was done after unit testing by combining different individual modules.

Some of the major test cases are listed below:

Test case 1:

Table 8: View the Request Feed

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Steps | Test Step | Test Data | Expected Results | Actual Results | Status  (pass/fail) |
| 1. | User Login | Valid credentials | Main activity layout should be shown. | Shows main activity layout. | Pass |
| 2. | Click on show list button |  | Should show two buttons. Each for different type of requests | Shows two buttons. Each for different type of requests | Pass |
| 3. | Click on request feed button |  | Should show the feed of information of users requesting blood. | Show the feed of information of users requesting blood. | Pass |

Test case 2:

Table 9:Request Blood

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Steps | Test step | Test Data | Expected  Results | Actual Results | Status  (pass/fail) |
| 1 | User login | Valid credentials | Should open Profile Page | Shows profile entry form if the user logged in for the first time. | Pass |
| 2. | Create Profile | Valid user information | User should be able to see the main activity. | Shows main activity | Pass |
| 3. | Post Request | Valid form for Request blood | Should be able to post the request. | Posts the post to database. | Pass |

### 4.2.3. System Testing:

System testing was done after integrating testing in order to ensure that the whole system functions properly. After the integration testing the whole system working process was checked. The output was as per the system specifications and hence the system was found to work properly.

Test case 1:

Table 10: Check if system works or not in different conditions

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Steps | Test Step | Test Data | Expected Results | Actual Results | Status  (pass/fail) |
| 1. | Disable Internet access |  | App should not open | Doesn’t open | Pass |
| 2. | Disable GPS |  | Maps should not show their current location. | Doesn’t show. | Pass |

Test case 2:

Table 11: Check if users can see database of users

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Steps | Test Step | Test Data | Expected Results | Actual Results | Status(pass/fail) |
| 1. | Log in using valid data |  | Should Log in | Logs user in | Pass |
| 2. | View Database of users |  | Should be able to view database of users | Shows database of users. Users can filter results based on district. | Pass |

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# CHAPTER 5: MAINTENANCE AND SUPPORT

## 5.1. Maintenance

Android blood bank application uses the Firebase server. Our system is implemented within 2 server i.e Firebase server which is mainly used for authentication purposes and Firebase Realtime Database server which is used when we need to work with stored data on the database server. When Google makes changes to the Firebase API we need to update our application to reflect the API changes proposed by Google. When we encounter the bugs and problems in application we need to update our code. Also, admin can edit, add and delete the event details, this can surely maintain the memory as event details can contain images and images have high memory.

## 5.2. Support

We have used the advanced version of Waterfall Model i.e. V Model. V model focuses on verification and validation so we can quickly change the requirement of the system. If any changes need to be made then we can make it for sure in a short period of time as V model allows us to do so. Admin can support the general user as user can send their feedback to admin directly. Support to general user is very important, so we allocate the access of admin panel to blood bank. If needed then we need to take cloud service for the data storage but this should be followed under economic feasibility of the project.

# CHAPTER 6: LIMITATIONS AND FUTURE ENHANCEMENTS

## 6.1 Limitations

• Internet access is required to run the application.

• There is no fixed way to validate the donor’s accurate information.

• There can be fake requests.

• The application depends upon the static location provided by the user during registration. There is no real-time location tracking.

## 6.2 Future Enhancement

* Event and Blood Request Notification.
* Support for facebook circle.

## 6.3 Conclusion

This project emphasizes on creating an effective blood donor management system. The main objective of this project is to narrow the communication gap with the donors and make volunteer donors available in minimum response time. “BloodBankAndroid: A System to connect with blood donors” was developed to help donors and recipients to be connected using the same platform.

The project wouldn’t have been completed if series of research weren’t conducted at various levels. To get more idea on the topic, we started with the research on different kinds of blood donation activities carried out in the Kathmandu valley. Similarly, the author researched about the distribution of blood trends in the youths of Kathmandu. The author studied the availability of volunteer donors in the valley. The research gave a broader knowledge on how the author could carry out this project more effectively. Thus, the author came up with the idea of enhancing the features that existed in online blood bank management systems.

# 

# CHAPTER 7: APPENDIX

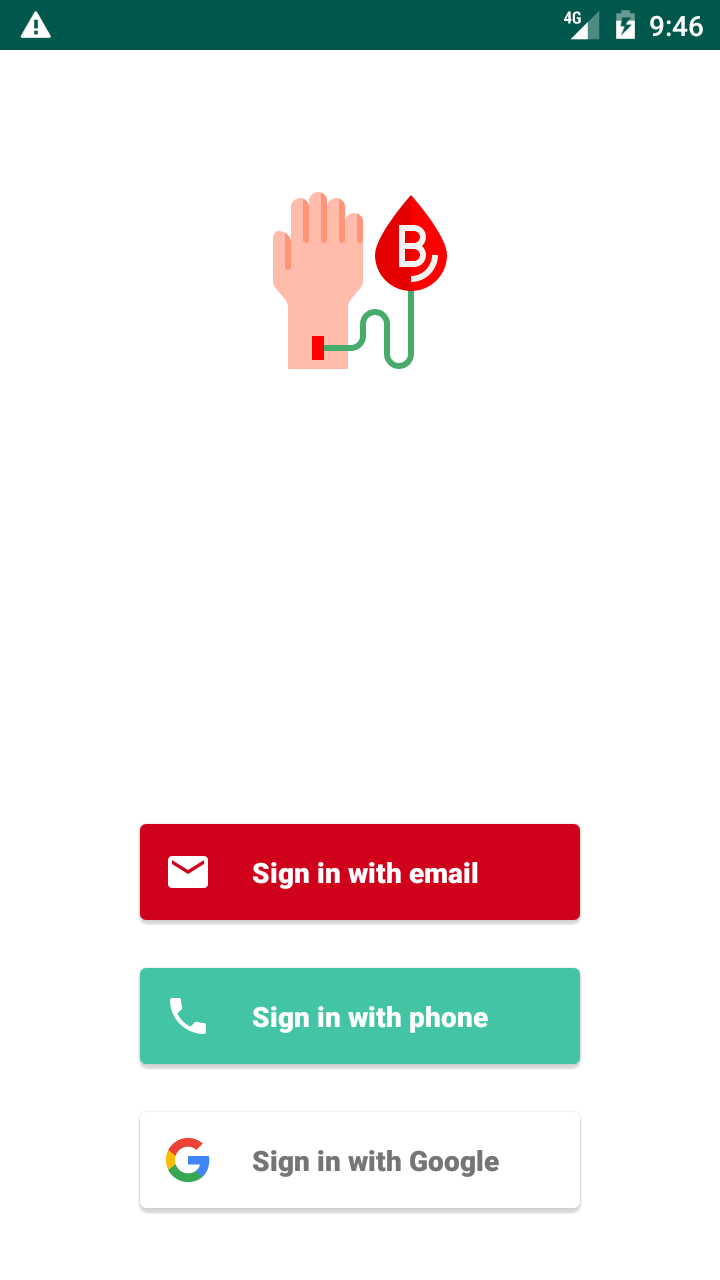
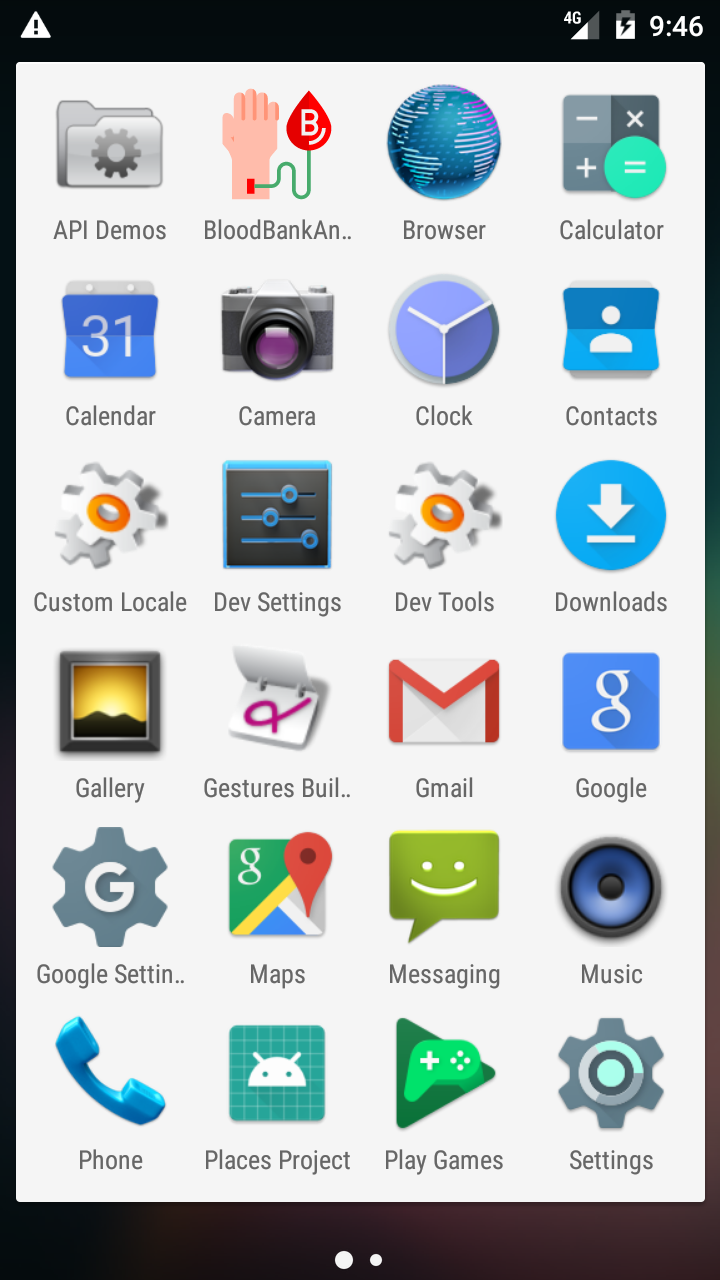


Figure 8: App icon in drawer Figure 9: App Log in/Sign up page

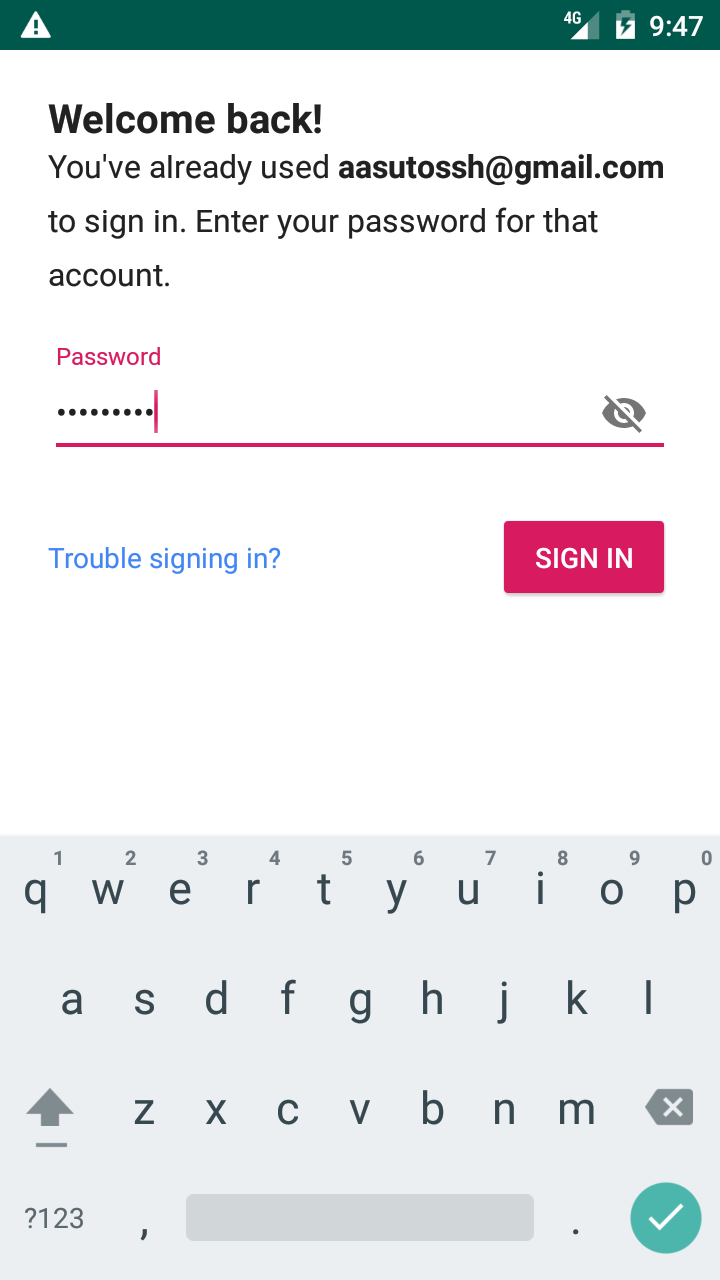
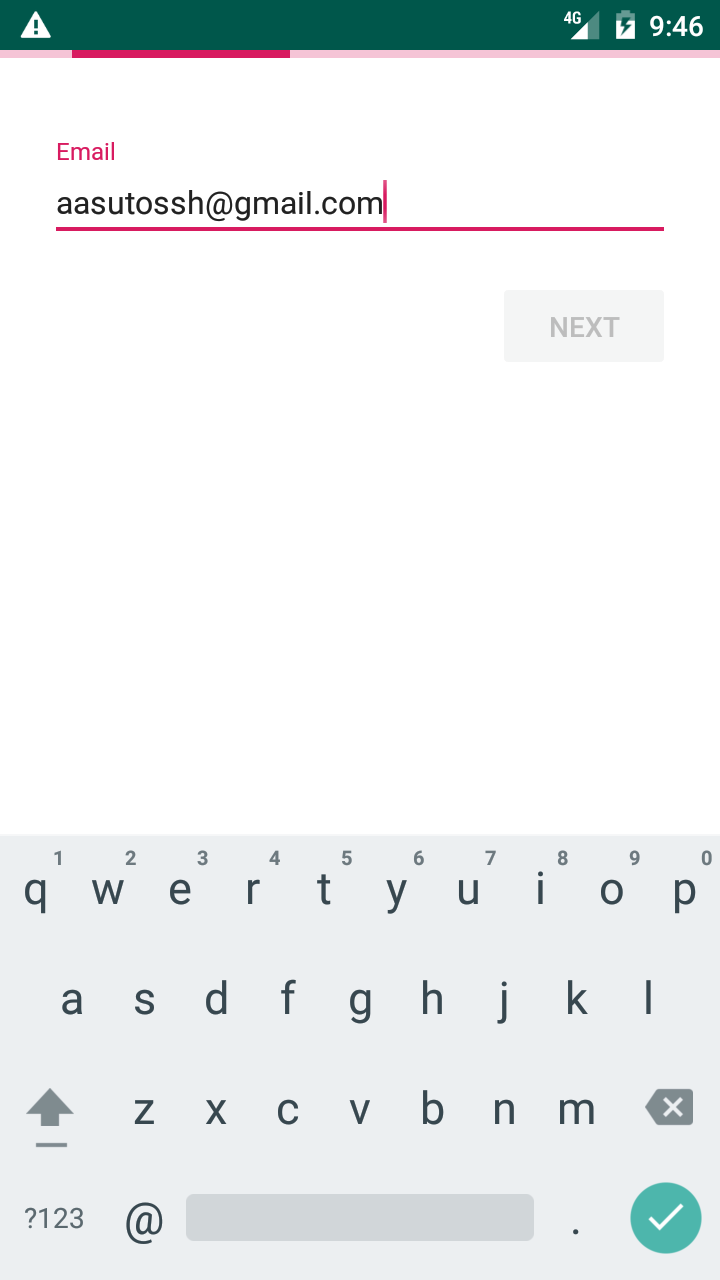


Figure 10: Login with email Figure 11: Completing login with password

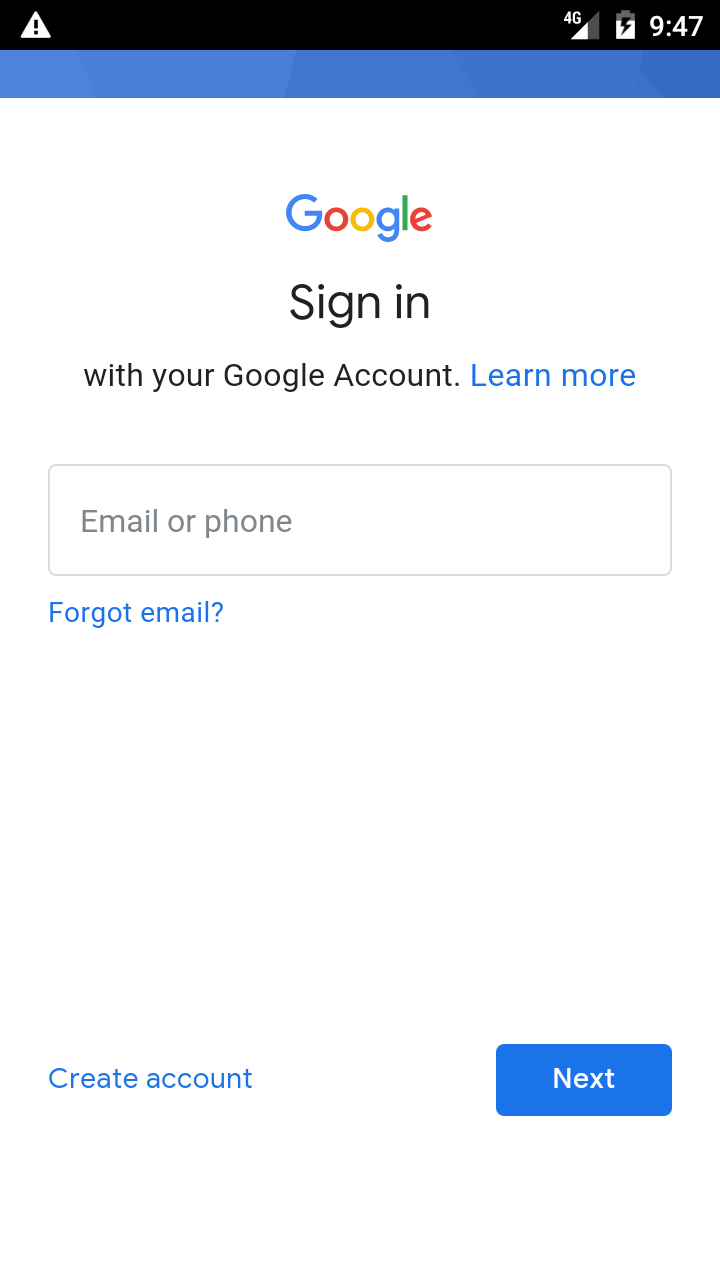
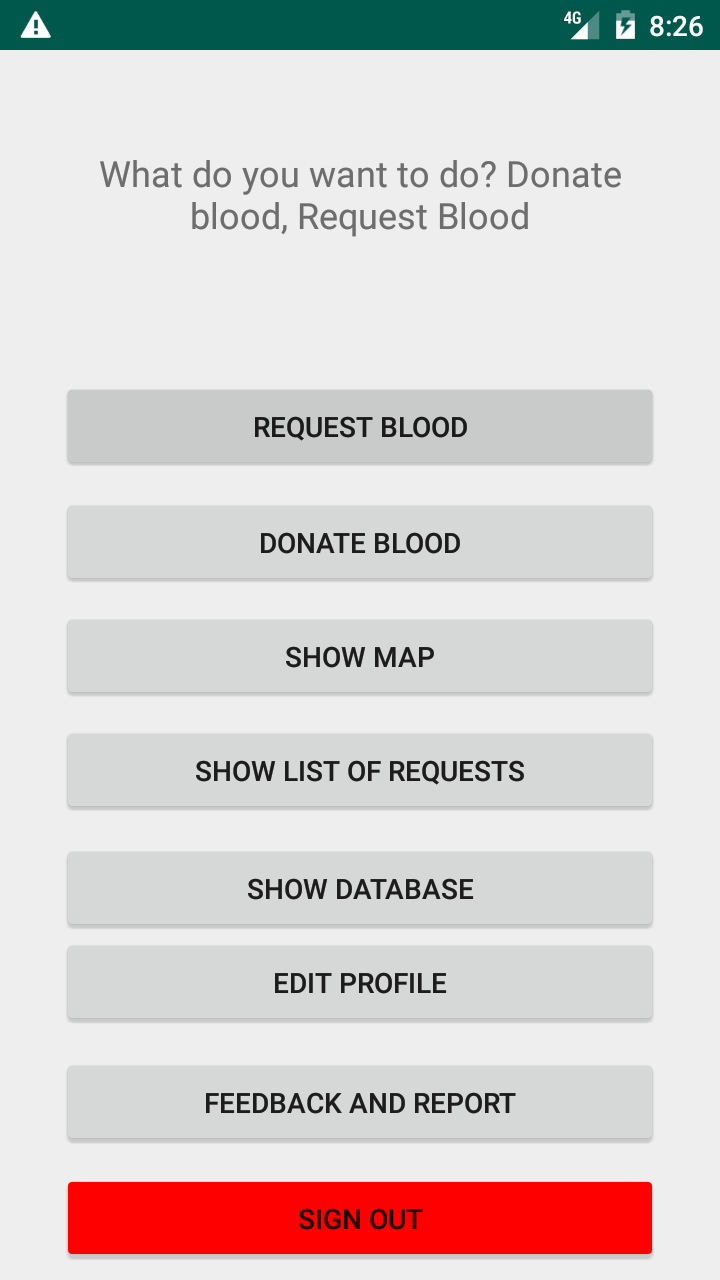
 

Figure 12: Login with Google Figure 13: Main entry page

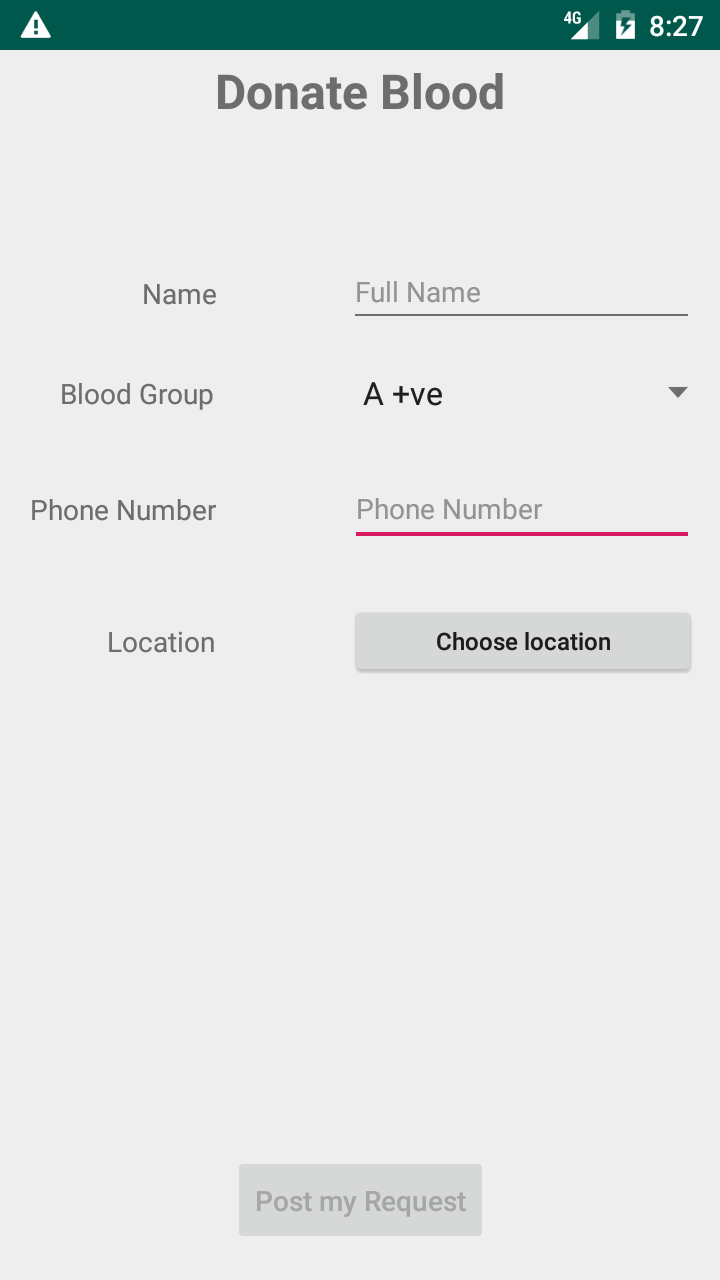
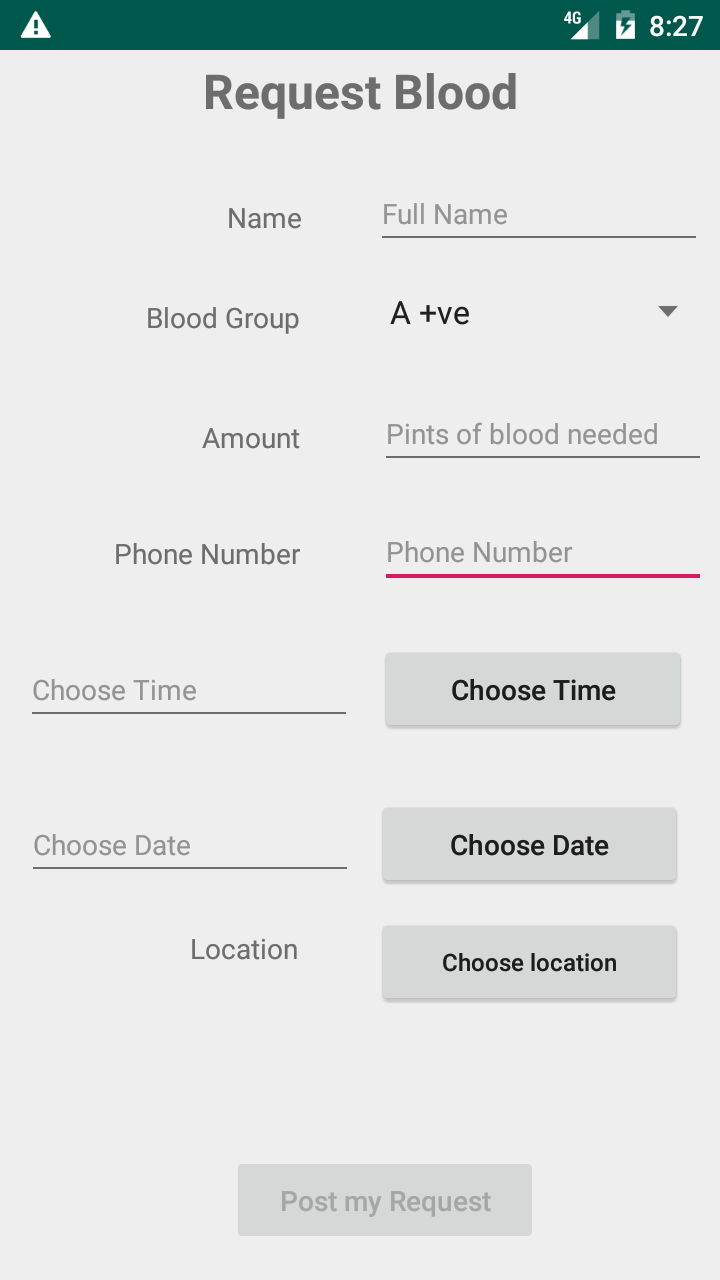


Figure 14: Request Blood Form Figure 15: Donate Blood Form

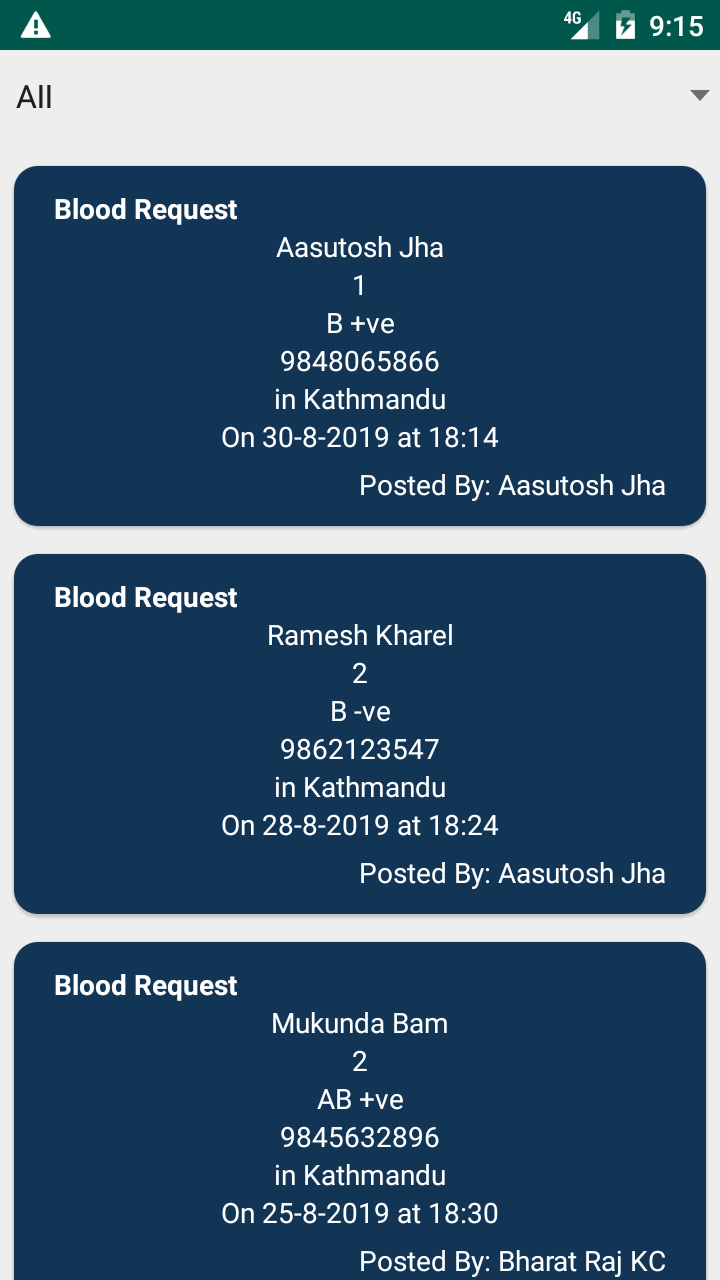
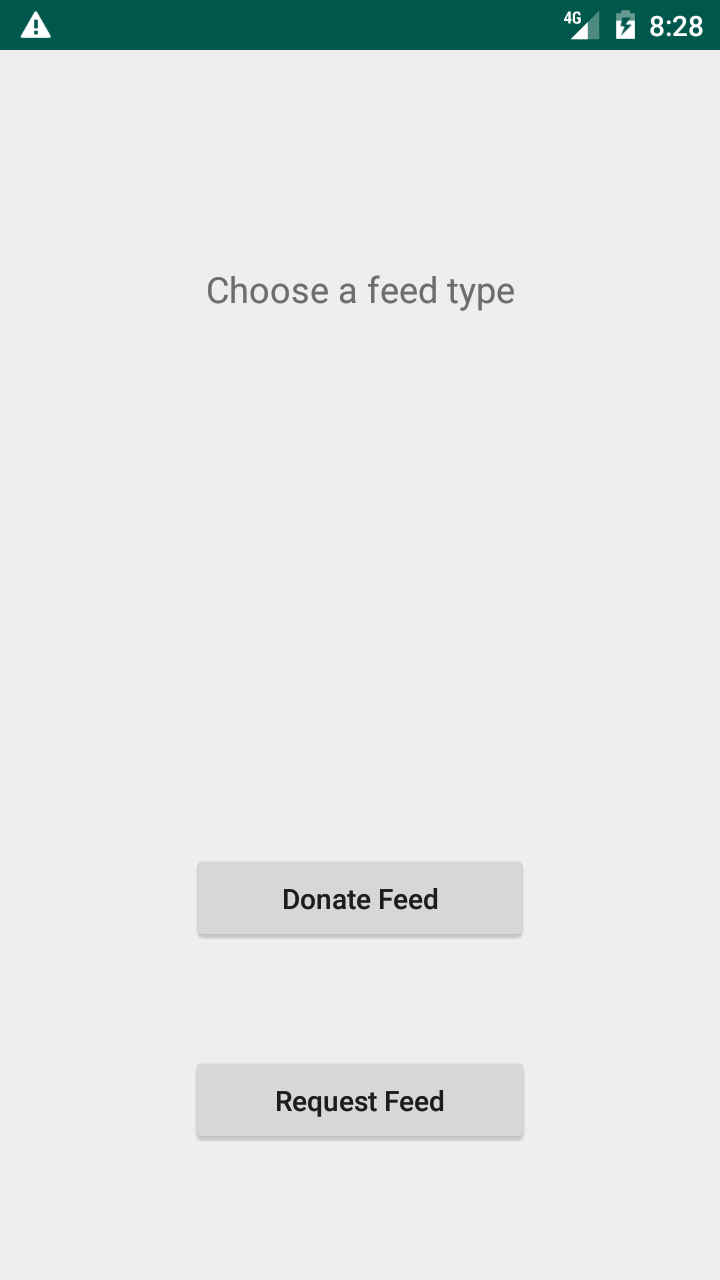


Figure 16: Type of feed Figure 17: Request Feed

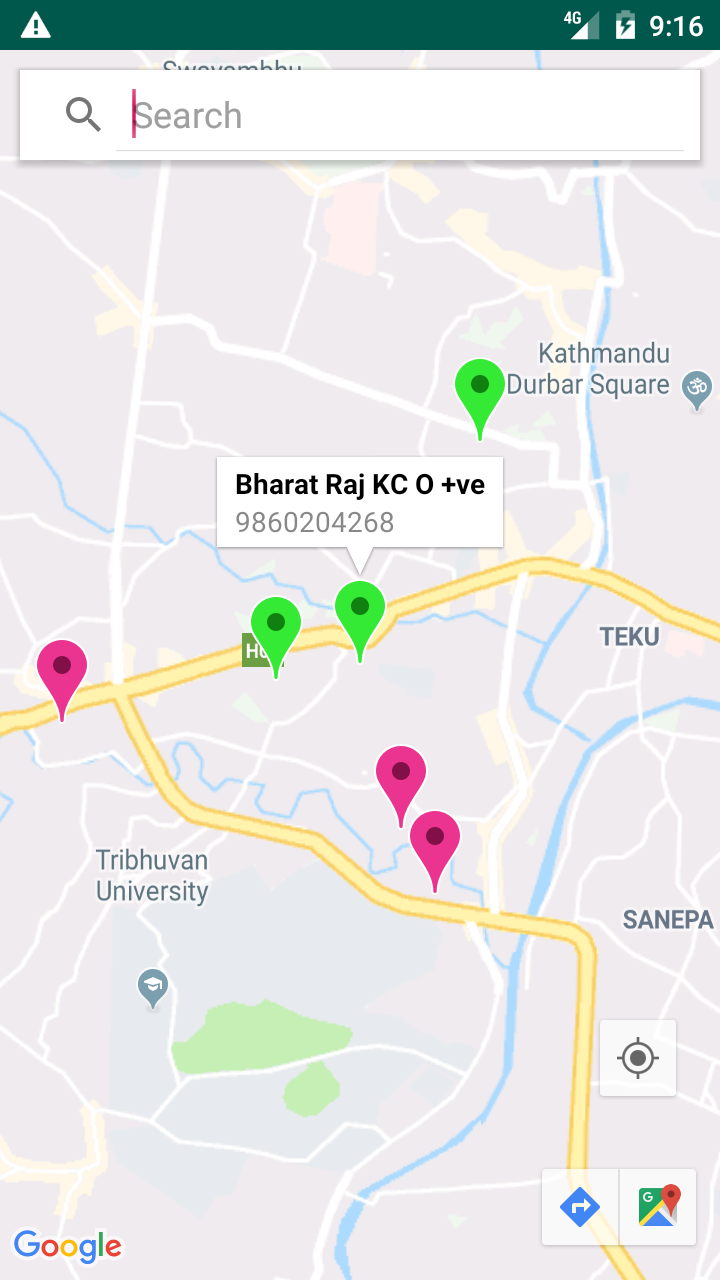
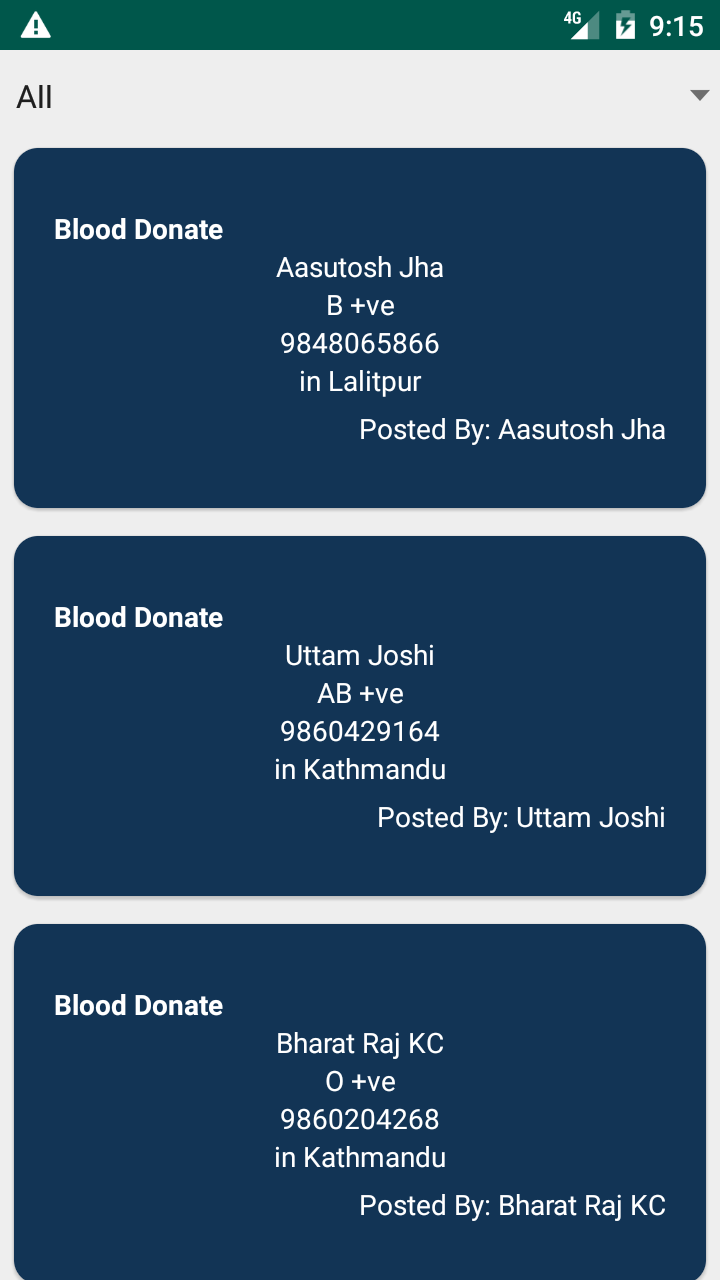


Figure 18: Donate Feed Figure 19: Maps showing both requests

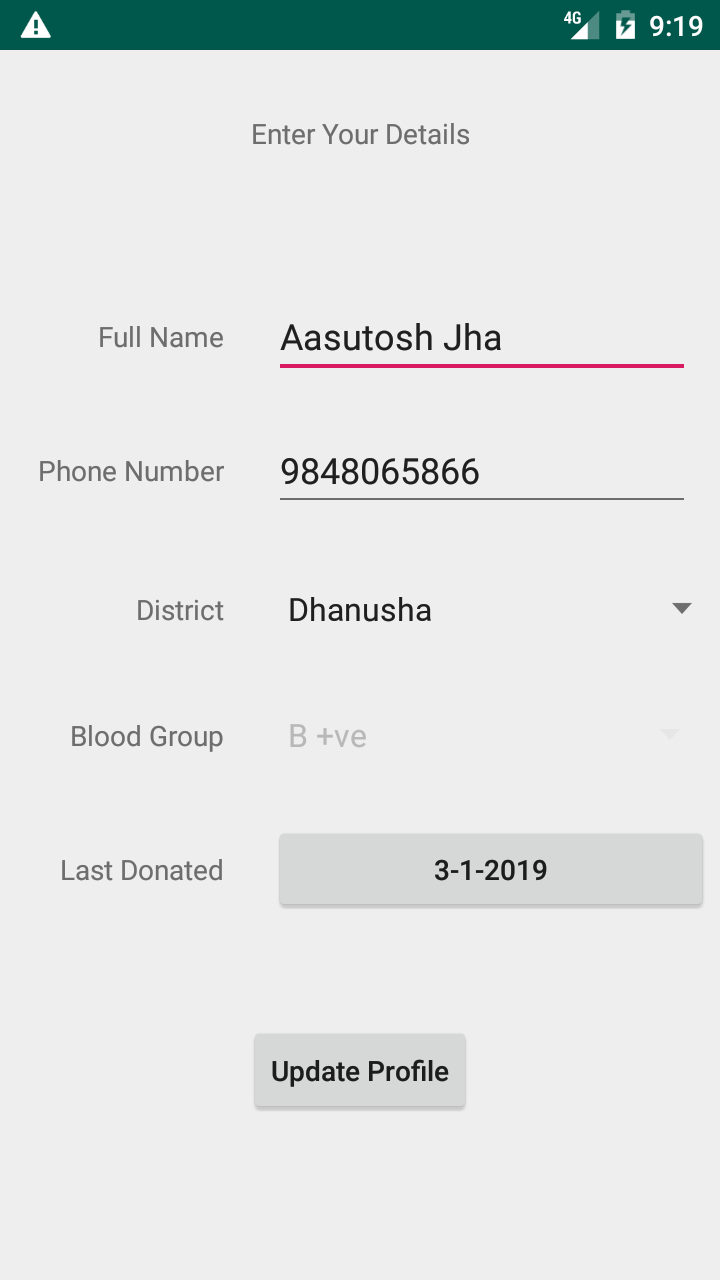


Figure 20: Image showing list of registered users’ info Fig 21: Edit Profile Form

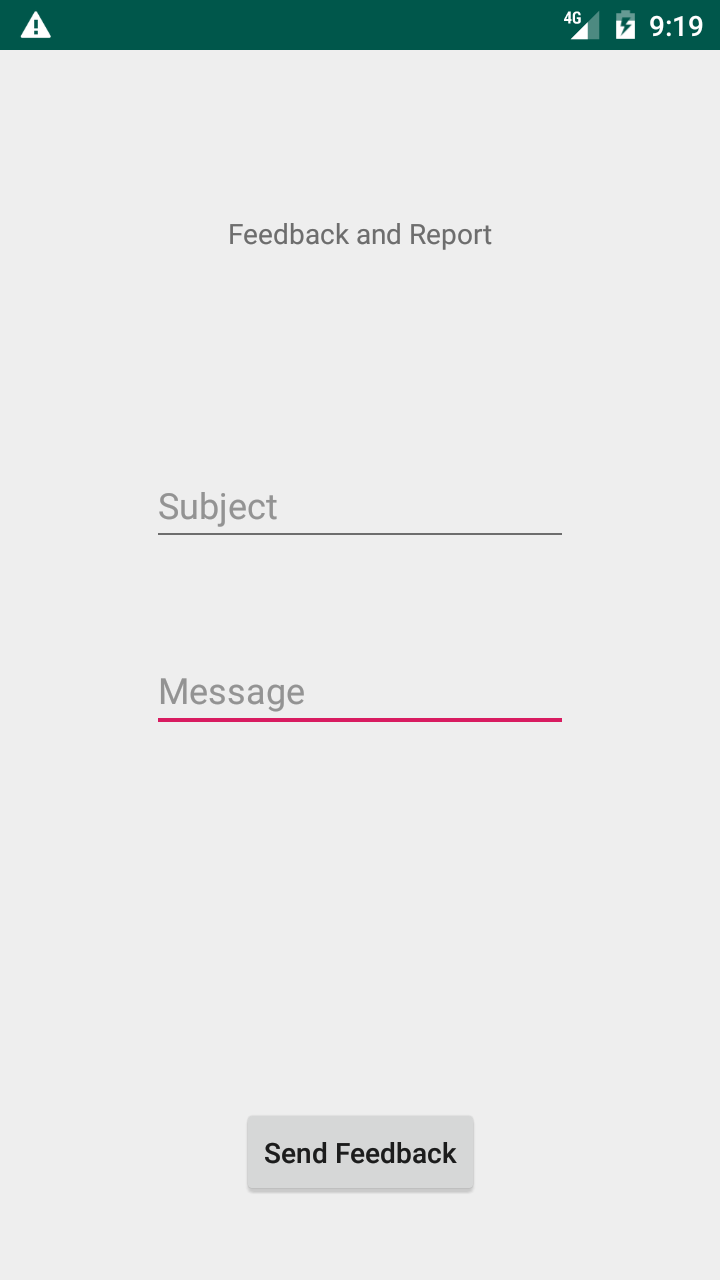


Fig 22: Send Feedback form

# REFERENCES

[1] S. Sarkar, "Artificial blood," [Online]. Available:

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2738310/>

[2] "Demand outstrips supply of blood - The Himalayan Times," [Online]. Available:

<https://thehimalayantimes.com/kathmandu/demand-outstrips-supply-of-blood/>

[3] "Blood demand increasing in Bhaktapur," [Online]. Available:

<https://thehimalayantimes.com/kathmandu/blood-demand-increasing-bhaktapur/>

[4] "Blood Donor Eligibility: Blood Pressure, Pregnancy, Disease & More," [Online]. Available:

<https://www.redcrossblood.org/donate-blood/how-to-donate/eligibility-requirements/eligibility-criteria-alphabetical.html>

[5] "National Population and Housing Census 2011," [Online]. Available:

<http://cbs.gov.np/image/data/Population/Population%20projection%202011-2031/PopulationProjection2011-2031.pdf>

[6] "Giving blood in a time of crisis", World Health Organization, 2018," [Online]. Available:

<https://www.who.int/en/news-room/feature-stories/detail/giving-blood-in-a-time-of-crisis>

[7] "Blood Banks”, The Official portal of Government of Nepal,2018," [Online]. Available:

<https://www.nepal.gov.np/NationalPortal/view-page?id=45>

[8] "Feasibility Study," [Online]. Available:

<https://www.investopedia.com/terms/f/feasibility-study.asp>

[9] "Types of Feasibility Study," [Online]. Available:

<http://ecomputernotes.com/software-engineering/feasibilitystudy>

[10] "What is Schedule Feasibility?," [Online]. Available:

<http://www.taskmanagementguide.com/glossary/what-is-schedule-feasibility.php>

[11] "Blood Donation Research Paper," no.

<https://research-paper.essayempire.com/examples/argumentative/blood-donation-research-paper/>

[12] "Blood Services in Nepal | Nepal Red Cross Society," [Online]. Available:

<http://www.nrcs.org/program/blood-services>

[13] "Starting of Youth For Blood," [Online]. Available:

<http://youthforblood.org/?reqp=1&reqr=>

[14] "History of Blood Donor Plus Mobile App," [Online]. Available:

<http://blooddonorplus.com/>

[15] E-Blood Donor, [Online]. Available:

<https://play.google.com/store/apps/details?id=com.obo.edonor&hl=en>

[16] Blood Members Nepal, [Online]. Available:

<https://play.google.com/store/apps/details?id=c25.niroj.bloodmembers&hl=en>

[17] Blood Donor’s Association Nepal(BOLDAN), [Online]. Available:

<http://bloodmembers.org/>

[18] eBlood, [Online]. Available:

<https://play.google.com/store/apps/details?id=com.eBlood&hl=en>

[19] Blood Nearby, [Online]. Available:

<https://play.google.com/store/apps/details?id=com.bloodnearby&hl=en>

[20] Java | Oracle

https://www.java.com/en/

[21] Android (operating system), [Online]. Available:

<https://en.wikipedia.org/wiki/Android_(operating_system)>

[22] What is Google Maps API? How it is used? [Online]. Available:

<https://github.com/RefugeRestrooms/refugerestrooms/wiki/What-is-the-Google-Maps-API%3F-How-isit-used%3F>

[23] Android Studio, [Online]. Available:

<https://en.wikipedia.org/wiki/Android_Studio>

[24] Git and GitHub, [Online]. Available:

<https://codeburst.io/git-and-github-in-a-nutshell-b0a3cc06458f>

[25] Detail about firebase:

<https://firebase.google.com/>

[26] What is Geocoding?

<https://developers.google.com/maps/documentation/geocoding/intro>

[27] "Systems Development | Software Application | Development," [Online]. Available:

<https://study.com/academy/lesson/systems-development-methods-and-tools.html>

[28] "System Analysis and Design Overview," [Online]. Available:

<https://www.tutorialspoint.com/system_analysis_and_design/system_analysis_and_design_overview>

[29] "Use case diagram," [Online]. Available:

<https://en.wikipedia.org/wiki/Use_case_diagram>

[30] Non-functional requirements, [Online]. Available:

<https://en.wikipedia.org/wiki/Non-functional_requirement>

[31] "Context diagrams," [Online]. Available:

<https://www.cs.uct.ac.za/mit_notes/software/htmls/ch06s06.html>

[32] Implementation:

<https://en.wikipedia.org/wiki/Implementation>

[33] V model:

<https://www.tutorialspoint.com/sdlc/sdlc_v_model.htm>

[34] V model image source:

<http://www.professionalqa.com/assets/images/v-model.png>

[35] Testing and testing types:

<https://www.codeproject.com/Tips/351122/What-is-software-testing-What-are-the-different-ty>