 **Final Year Project Report**

**“Blood Bucket (Blood Bank Management System)”**

**Bachelor of Science**

in Software Engineering

**Submitted by**

Aiman Mohsin EP1650006

Hareem Zubair EP1650020

Khadija Yousuf EP1650027

Supervised by

**DR. PROF. SADIQ ALI KHAN**

**DEPARTMENT OF COMPUTER SCIENCE**

**UNIVERSITY OF KARACHI**

**2019**

**Abstract**

A report submitted to the Department of Computer Science in partial fulfilment of the requirements for the Degree Bachelor of Science in Software Engineering.

By,

Hareem Zubair

Khadija Yousuf

Aiman Mohsin

Application for Blood Bank Management System is a way to synchronize Blood banks and Hospitals with the help of Internet. It is a Web Application and an Android Application through which Registered Hospitals can check the availability of required Blood and can send Request for blood to the nearest blood bank or donor matching with blood requirement and can be ordered online as and when required. Blood bank can also send a request to another blood bank for unavailable blood. Person willing to donate blood can find out nearest blood banks using Blood Bank Management Android Application. The location of the blood bank can also be traced using maps. The Android application can be accessed only by the needer to search the blood donation centers and the requesting blood banks and hospitals to search the nearest blood banks and donors.

This report explains the fundamental features their description and scope. It has modules name Donor, Needer, Organizations, Hospitals and Admin and whole project is also represented through UML Diagrams and TestCases.

Finally, The Report also includes some possible future enhancement that will be considered to achieved. Blood Bucket is designed to help the Blood Bank administrator to meet the demand of Blood by sending and/or serving the request for Blood as and when required.

# CERTIFICATE OF COMPLETION

**This is to certify that the following students**

Aiman Mohsin EP1650006

Hareem Zubair EP1650020

Khadija Yousuf EP1650027

**Have successfully completed their final year project named**

**Blood Bucket (Blood Bank Management System)**

**In the partial fulfillment of the Degree of Bachelor of Science in Computer Science**

Signature & Seal of Supervisor

|  |  |
| --- | --- |
| Name: |  |
|  |  |
| Department |  |
|  |  |
| Email |  |
|  |  |
| Mobile No. |  |
|  |  |
| Date |  |

# ACKNOWLEDGEMENT

Firstly, I would like to thank Allah for all his mercies upon us, our final year project that we started with a little idea and here we are about to accomplish it. This task has been done with all hard work and efforts of my group members even though there were a few problems among us while carrying this project but we overcome them wisely.

Secondly, I would like to express my appreciation to my academic supervisor, our HOD **Sir Sadiq Ali Khan**, who is a great teacher. He always gives us support, his advice and guidance allowed us to perform to our fullest potential. On the other hand, I also thank to our assistant professor **Sir Muhammad Naseem** who provided us with valuable comments and feedback at various stages of this project.

Finally, thanks to all team members who always stick together and also work hard to produce a successful project with all effort and responsibility. I hope all our effort will give benefits to us and also to our project.

Table of Contents

[Abstract…………………………………………………...………………………..ii](#_Toc521097670)

[CERTIFICATE OF COMPLETION………………………………………........iii](#_Toc521097671)

[ACKNOWLEDGEMENT………………………………………………………..i](#_Toc521097672)v

[Listof Figures …………………………………………………………………......viii](#_Toc521097673)

[List of Tables………………………………………………………………...……viii](#_Toc521097673)

[List of Keywords……………………………………………………………..……ix](#_Toc521097675)

[1 Introduction](#_Toc521097676) ………………………………………………………………..1

[1.1 Motivation](#_Toc521097677) 1

[1.2 Aim and Objective](#_Toc521097678) 1

[1.3 Scope of the Project](#_Toc521097679) 1

[1.4 Chapter Outline](#_Toc521097679) 2

[2 Approach Used](#_Toc521097680) 3

[2.1 Methodology](#_Toc521097681) 3

[2.1.1 Agile Methodology………………………………………………….](#_Toc521097681) 3

[2.2 Operating System](#_Toc521097682) 4

[2.2.1 HTML .](#_Toc521097684) 4

[2.2.2 CSS.](#_Toc521097684) 4

[2.2.3 Javascript.](#_Toc521097684) 5

[2.2.4 Bootstrap.](#_Toc521097684) 5

[2.3 Database Technology](#_Toc521097683) 5

[2.3.1 SQL.](#_Toc521097684) 5

[2.4 Functional Requirements 6](#_Toc521097683)

[2.4.1 FRs. 6,7](#_Toc521097691)

[2.5 Non-Functional Requirements 7](#_Toc521097683)

[2.5.1 Performance.](#_Toc521097691) 7

[2.5.2 Security. 8](#_Toc521097691)

[2.5.3 Avaialability. 8](#_Toc521097691)

[2.5.4 Safety. 8](#_Toc521097691)

[2.5.5 Software Quality. 9](#_Toc521097691)

2.6 [Chapter Summary](#_Toc521097686) 9

[3 Main Page…………………………………………………………………](#_Toc521097687) 10

[4 Login Page…………………………………………………………………](#_Toc521097687) 11

[4.1 Registration Page 11,12](#_Toc521097689)

[4.2 Admin Panel 12](#_Toc521097690)

[4.3 Chapter Summary 12](#_Toc521097690)

[5 User and Admin Profiles…………………………………………...](#_Toc521097687) 13

[5.1 User Profile 13](#_Toc521097689)

[5.1.1 Account.](#_Toc521097691) 13

[5.1.2 See Needers.](#_Toc521097691) 14

[5.1.3 See Donors.](#_Toc521097691) 14

[5.1.4 Needer Registration.](#_Toc521097691) 15

[5.1.5 Donor Registration.](#_Toc521097691) 15

[5.2 Admin Profile 15](#_Toc521097689)

[5.2.1 Admin Panel.](#_Toc521097691) 16

[5.2.2 See Needers.](#_Toc521097691) 16

[5.2.3 See Donors.](#_Toc521097691) 17

[5.2.4 Patient Details.](#_Toc521097691) 17

[5.2.5 Hospital Details.](#_Toc521097691) 18

[5.2.6 Organization Details.](#_Toc521097691) 18

[5.2.7 Self-Donor Details.](#_Toc521097691) 19

[5.2.8 Messaging Service.](#_Toc521097691) 19

[5.3 Chapter Summary 19](#_Toc521097689)

[6 Needer Panel – Android…………………………………………………..](#_Toc521097687) 20

[6.1 Home Screen 20](#_Toc521097689)

[6.2 Profile 21](#_Toc521097689)

[6.3 Nearby Blood Donor 21](#_Toc521097689)

[6.4 Donor’s Profile 22](#_Toc521097689)

[6.4 Needer’s Profile 22](#_Toc521097689)

[7 UML Diagrams……………………………………………………..](#_Toc521097687) 23

[7.1 User Flow Diagram 23](#_Toc521097689)

[7.2 Admin Flow Diagram 24](#_Toc521097689)

[7.3 ERD of Donor + Admin 25](#_Toc521097689)

[7.4 ERD of Needer + Admin 25](#_Toc521097689)

[7.5 Class Diagram 26](#_Toc521097689)

[7.6 Use Case Diagram 26](#_Toc521097689)

[8 Conclusion and Future Enhancements…………………………………..](#_Toc521097687) 27

[8.1 Conclusion 27](#_Toc521097689)

[8.2 Future Enhancements 27](#_Toc521097689)

[References 2](#_Toc521097712)8

[Appendix – A: User Manual 29](#_Toc521097713)

[Appendix – B: Coding 30-29](#_Toc521097714)

List of Figures

[**Figure 2.1:** Agile Development Methodology 3](#_Toc521097715)

[**Figure 7.1:** User Flow Diagram 19](#_Toc521097715)

[**Figure 7.2:** Admin Flow Diagram 20](#_Toc521097715)

[**Figure 7.3:** ERD of Donor + Admin 21](#_Toc521097715)

[**Figure 7.4:** ERD of Needy + Admin 21](#_Toc521097715)

[**Figure 7.5:** Class Diagram 22](#_Toc521097715)

[**Figure 7.6:** Use Case Diagram 22](#_Toc521097715)

List of Tables

[**Table 2.1:** Minimum Requirement of Operating Environment 6](#_Toc521097715)

# List of Keywords

SQL – Structured Query Language

HTML – Hypertext Markup Language

CSS – Cascading Style Sheets

GUI – Graphical User Interface

AndroidOS – Android Operating System

# Introduction

The population of the world is multiplying with each coming year and so are the diseases and health issues. With an increase in the population there is an increase in the need of blood. The growing population of the world results in a lot of potential blood donors. But in spite of this not more than 10% of the total world population participates in blood donation. With the growing population and the advancement in medical science the demand for blood has also increased. Due to the lack of communication between the blood donors and the blood recipients, most of the patients in need of blood do not get blood on time and hence lose their lives.

## Motivation

## There is a dire need of synchronization between the blood donors and hospitals and the blood banks. This improper management of blood leads to wastage of the available blood inventory. Improper communication and synchronization between the blood banks and hospitals leads to wastage of the blood available. These problems can be dealt with by automating the existing manual blood bank management system. A high-end, efficient, highly available and scalable system must be developed to bridge the gap between the donors and the recipients and to reduce the efforts required to search for blood donors.

## Aim and Objective

This applied research aims to design, develop and implement online blood bank management system that provides platform to everyone so they can donate blood, get blood, work as an organization or hospital.

## Scope of the Project

## . It includes,

• Ensure that all the functionalities of a manual blood bank are covered.

• To include all the blood banks at least within a city.

• Make sure the program is simple and easy to use.

## Outline of the Project

The project Description of various activities from the Android application and web application is provided.

Chapter 1**: Introduction** – A Home page that contain About, Contact and all other description of Blood Bucket. It has access to login as well which proceeds further whether you want to login as a patient, hospital, organization and donor or you want to login as an admin or you are new member so want to register yourself.

Chapter 2: **Approach Used** – On a registration page you will have 4 options whether you want to register as a Donor, Organization, Needer or Hospital which proceeds further.

Chapter 3: **Main** – It shows main page of website.

Chapter 4: **Login** – It cover login page that takes you to two panels.

1. Admin
2. User

Chapter 5: **Registration --** This chapter covers all kind of Registrations.

Chapter 6: **Needer Panel-Android –** A needer panel is designed/developed using android operation system, so the blood required in case of emergency is easily accessible

Chapter 7: **Conclusion and Future Enhancement** - Its presents Technology is introducing new innovations day by day, thus reducing the time required to do things. The proposed system can be used to reduce the time required to deliver required blood to the needy in cases of emergency.

# Approach Used

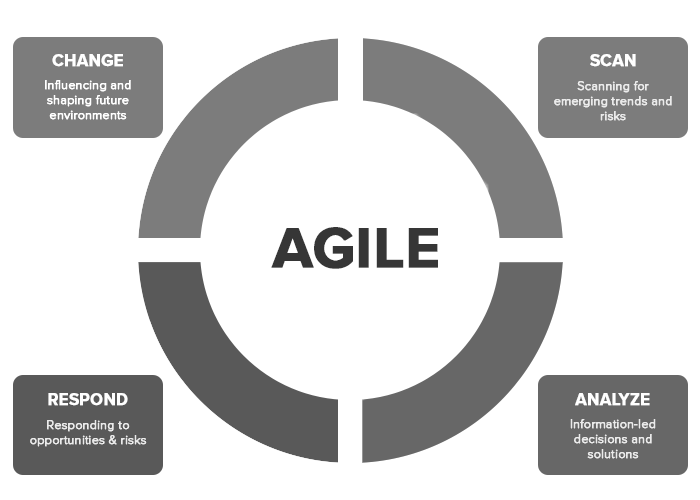
This chapter helps in evaluating the methodology used to execute the project against the other standard methodologies. This contains the details in brief about the experimentations carried out to complete the project. It also includes some flow charts or figures to make it easier for the reader to understand the methodology.

## Methodologies

In the Methodologies we’ve used Agile which is a software development approach under which requirements and solutions evolve through the collaborative effort of self-organizing and cross-functional teams and their customer/end user. Methodology is essentially a set of guiding principles and processes for managing a project because it helps to organize the development process.

### Agile Method

In the Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. Agile method was very helpful for us to in developing project through which we analyze; implement and change the requirement are mention in Figure 2.1.



**Figure 2.1:** Agile development Methodology

On the first phase we analyze the problem gather the information and then move toward the second one by concluding the solution in the form of web portal and staring to work on it.

## Operating Environment

The programming work was carried out on one computer which ran the Windows 7 Ultimate system. The home version was not chosen since it did not support the IIS web server. The features of IIS 7.5 were introduced in Chapter 2. In the final implementation, ran Windows 7 and, and it was tested on IE, Chrome, and Firefox Client-side and server-side technologies

### Hypertext Markup Language (HTML)

In **Hypertext Markup Language** (**HTML**) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items.

### Cascading Style Sheets*(CSS)*

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

### JavaScript

Alongside HTML and CSS, JavaScript is one of the three core technologies of the World Wide Web. JavaScript enables interactive web pages and thus is an essential part of web applications. The vast majority of websites use it, and all major web browsers have a dedicated JavaScript engine to execute it As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles.

### Bootstrap

Bootstrap is a free and open-source front-end framework (library) for designing websites and web applications. It contains HTML and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. Unlike many web frameworks, it concerns itself with front-end development only.

## Database technologies

#### 2.3.1 SQL

SQL is a free open source software, widely available and with good technical support. It is compatible with many Operating Systems including Windows, Linux and Mac and can be used with both Apache and Windows web servers. It can also handle large amounts of data and provides security through user authorization and access privileges. SQL merges perfectly with PHP, Perl and ASP making it ideal for managing the content of a web application. These server side scripts can therefore be used to interact with a SQL database to retrieve and add information to a database. This experience could prove vital during the development stage.

A brief about the operating environment mention in Table 2.1

Table 2.1: Min requiremet of Operating Enviroment

|  |  |
| --- | --- |
| **Option** | **Description** |
| Processor | Intel premium 233 GHz or better performance |
| Operating system | Windows 7,8 and 10 |
| Memory | 2 GB Ram |
|  |  |

## Functional Requirements

**2.4.1 Functional Requirements**

* **FR-1 Admin Login**

Description: The System provides facility to login into the web based application system.

Input: Enter username and password

Output: User Profile page.

Processing: The system will check the input of user and if valid then login is done.

Otherwise user will be asked to re-enter the username and password.

* **FR-2 Admin Add, Update, Delete**

Description: The System provides facility to add, update, and delete data the system.

Input: Buttons clicked for respective action by Admin.

Output: Action performed.

Processing: Database updated.

* **FR-3 Needer Request**

Description: The System provides facility to request through online mobile application on AndroidOS smartphones.

Input: Enter User ID and Password. Client’s reason to request.

Output: Request successfully made.

Processing: The system will check the input of user and if valid; then compare with Database. Then the request will be processed and added to verified user’s account in company’s database.

Otherwise user will be asked to re-enter the username and password.

## Non Functional Requirements

**2.5.1 PERFORMANCE**

* **Response Time:**

The system shall give responses in 1 second after checking the user’s query.

* **Capacity:**

The System must support 100000 people at a time.

* **User Interface:**

The User Interface screen shall respond within 5 seconds.

* **User Satisfaction:** -

The system is such that it stands up to the user expectations.

* **Error Handling:** -

Response to user errors and undesired situations has been taken care of to ensure that the system operates without halting.

* **Safety and Robustness: -**

The system is able to avoid or tackle disastrous action. In other words, it should be foul proof. The system safeguards against undesired events, without human intervention.

* **Portable: -**

The system should not be architecture specific. It should be easily transferable to other platforms if needed.

* **User friendliness: -**

The system is easy to learn and understand. A native user can also use the system effectively, without any difficulties.

**2.5.2 SECURITY:**

* **Modification**

Any modification (Insert, Delete, and Update) for the Database shall be synchronized and only be done by the admin.

* **Admin Rights**

Admin will be able to view and modify all information in the database system.

**2.5.3 AVAILABILITY:**

The system shall be available all the time. But Internet is must to make any request.

**2.5.4 SAFETY:**

Humans are error-prone, but the negative effects of common errors should be limited.

E.g., users should realize that a given command will delete data, and be asked to confirm their intent or have the option to undo.

**2.5.5 SOFTWARE QUALITY:**

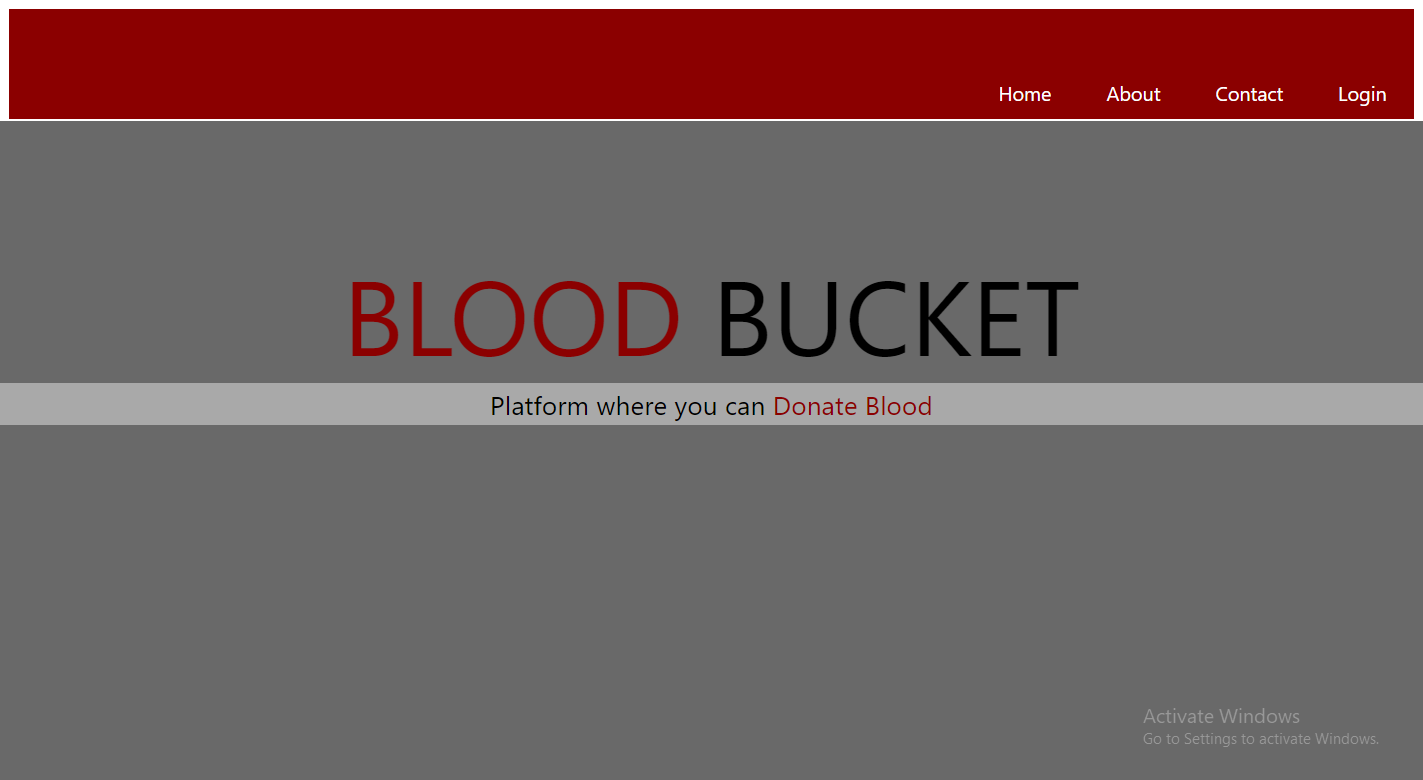
* Good quality of the framework produces robust, bug free system which contains all necessary requirements Customer satisfaction.
* Is part of the code going to be used elsewhere produces simple and independent code modules that can be reused.

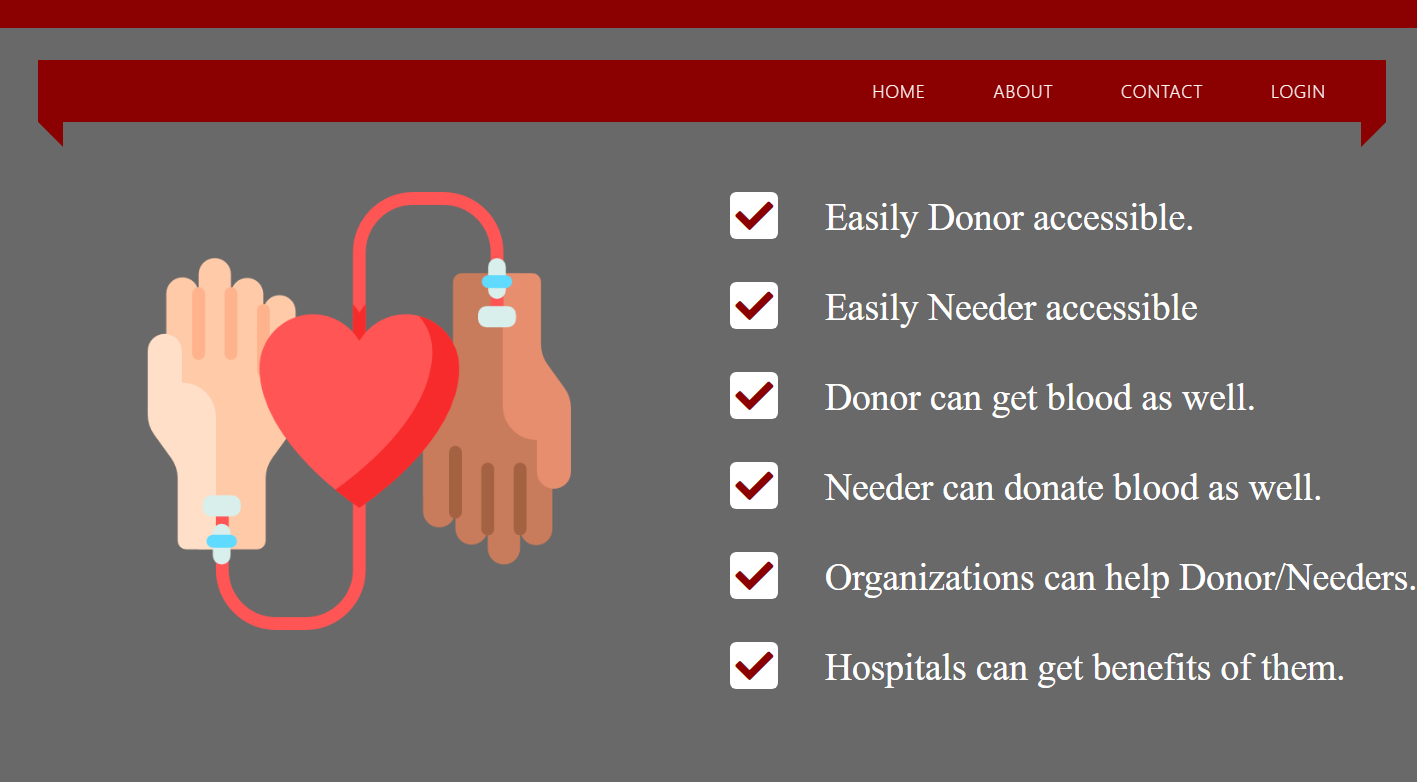
## Chapter Summary

Chapter 2 -we’ve discussed the methodology used to execute the project against the other standard methodologies. It also includes the operating environment, project background, languages used to create this project.

# Main Page

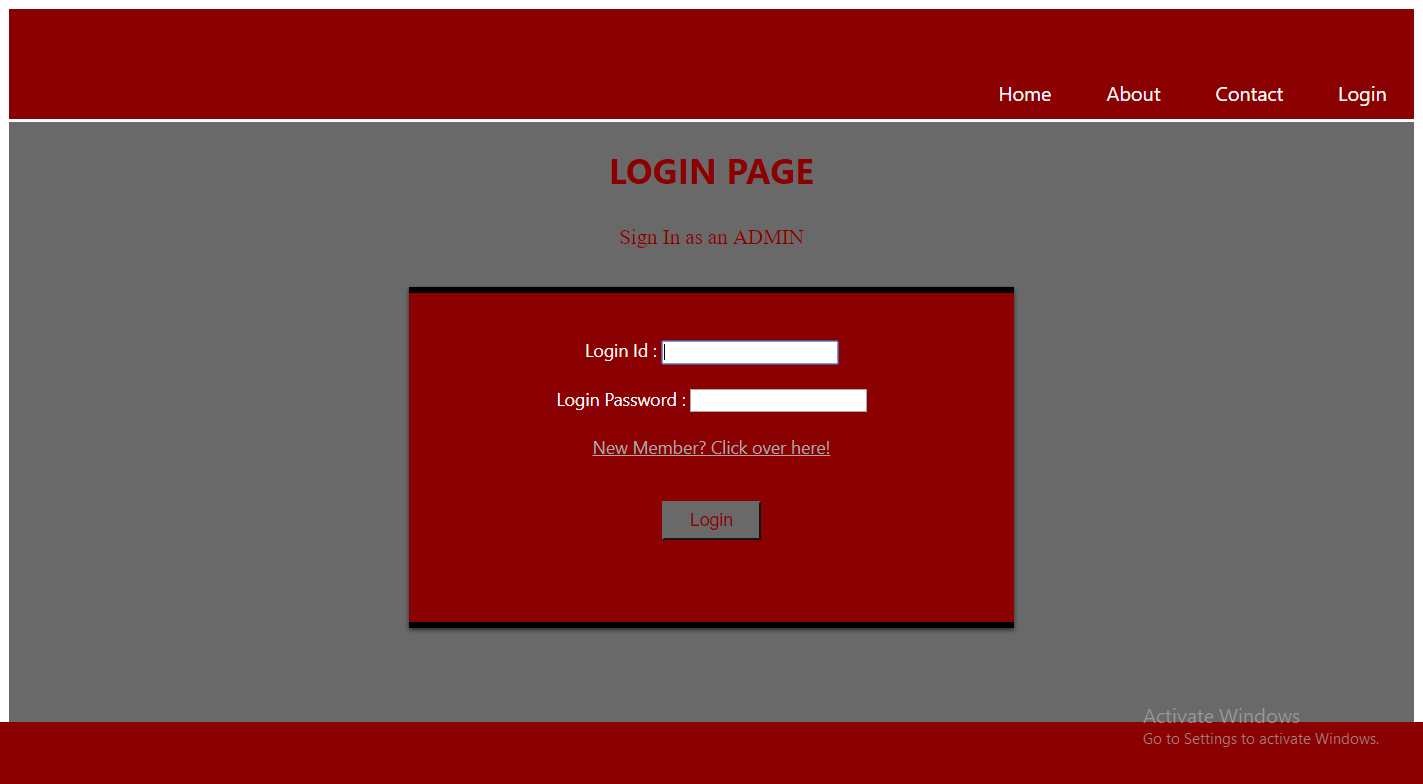
This chapter covers 3 main section that is Home, About and Contact. It further expands below.





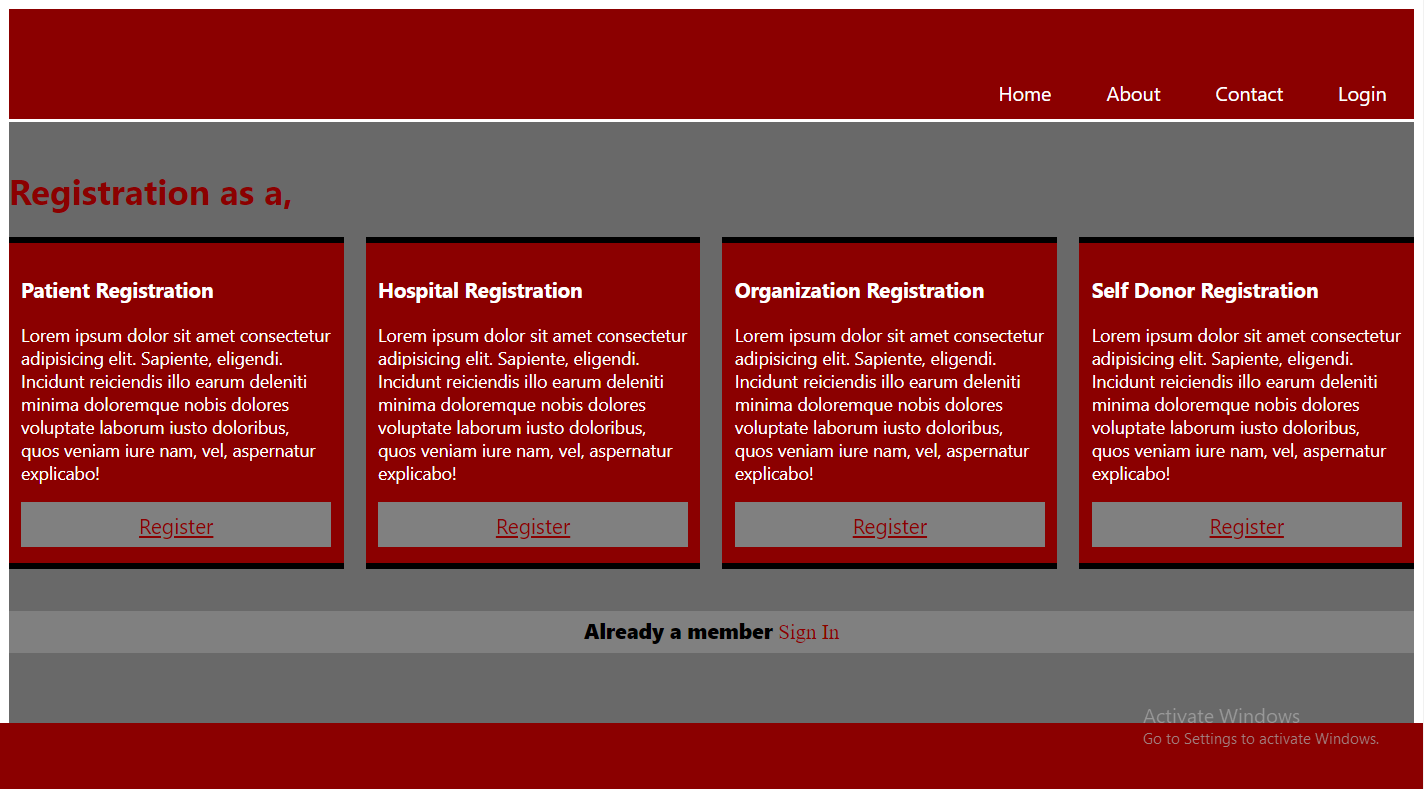
# Login

The hospitals and the blood banks will be provided an id through which they can login into the system. The hospitals and the blood banks registered into the system will be allowed access to the system and they will be directed to the home page.



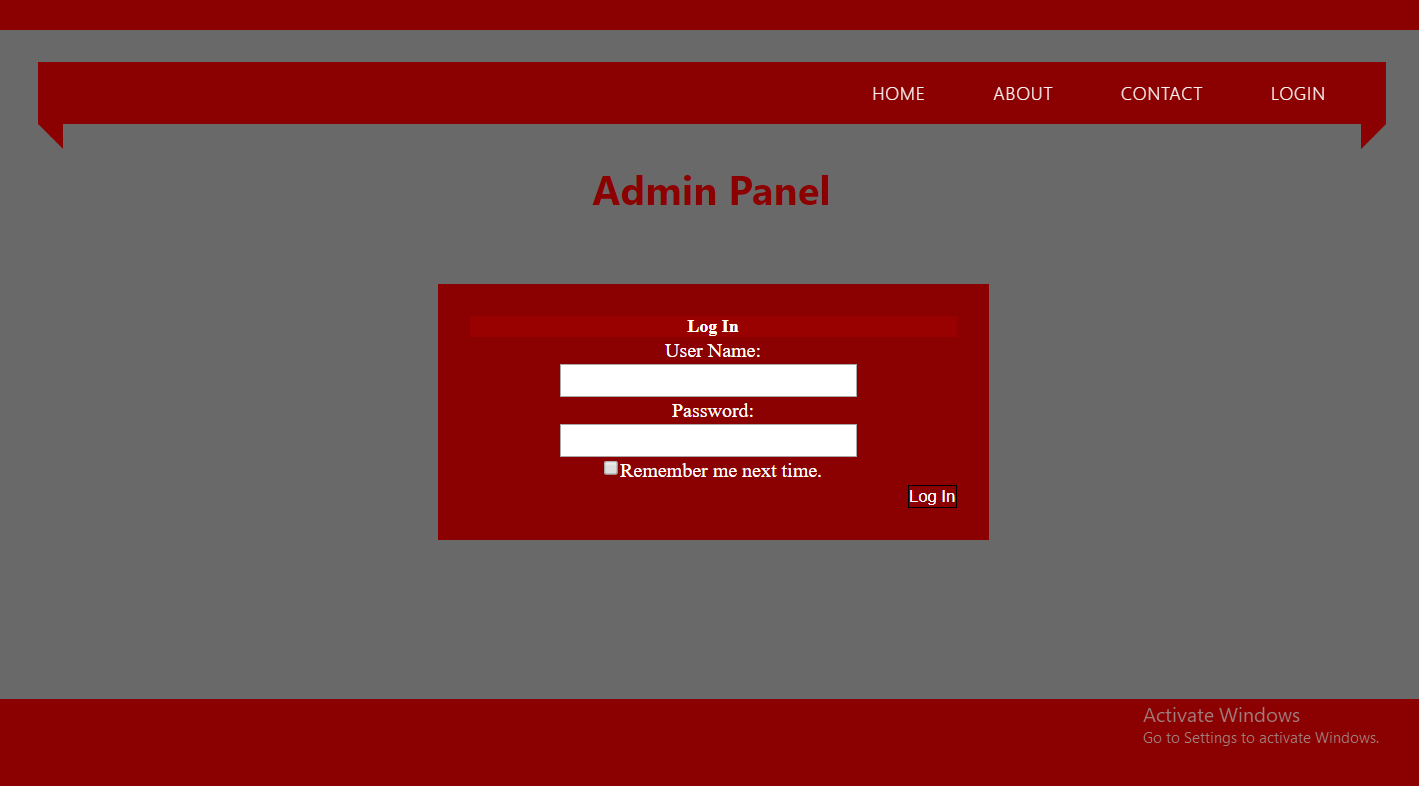
## Registration

Where user can register



## Admin Panel

Admin can access through



## Chapter Summary

Chapter 2 covers login portal, that showing two panels

1. Admin
2. User

# User & Admin Profiles

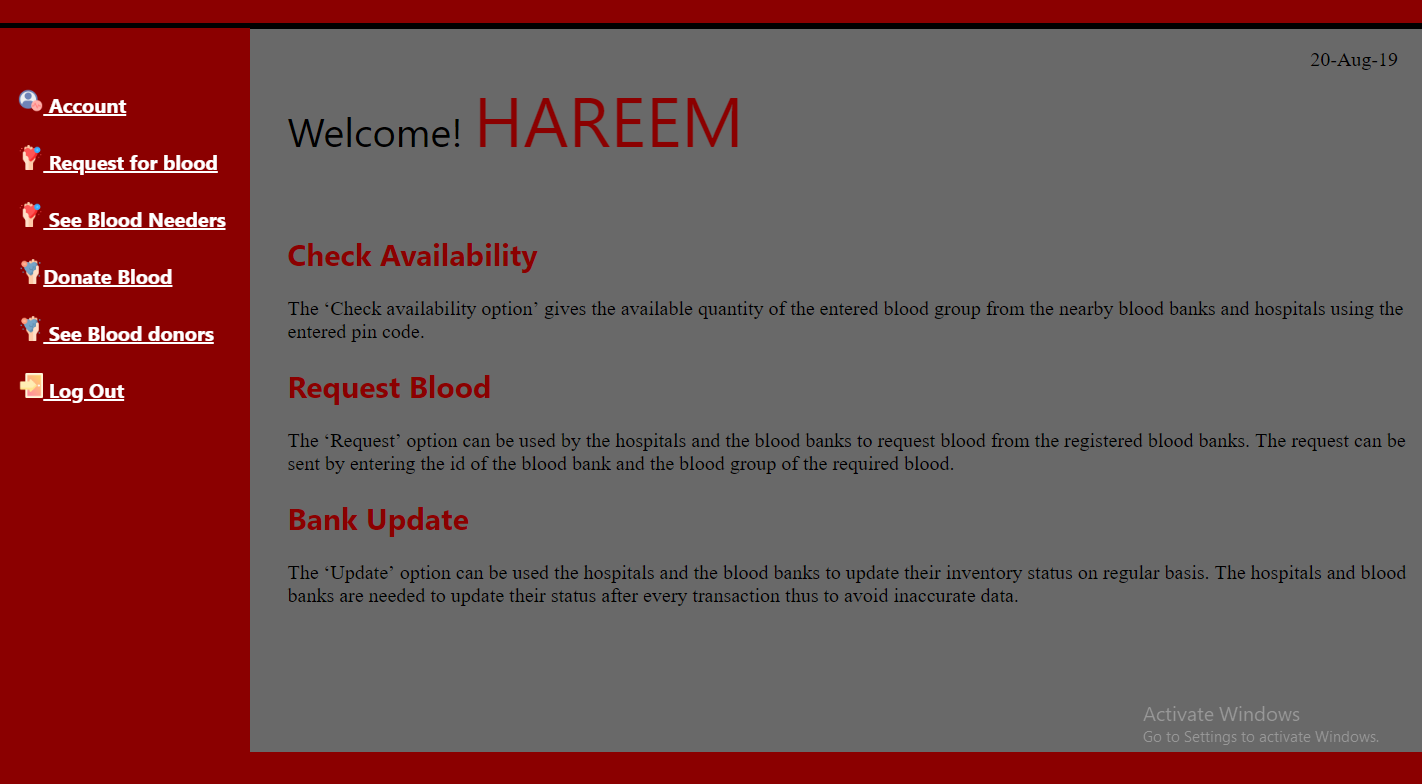
This chapter covers Admin and User Profiles.

All users that can be Patients, Organization, Hospitals or Self Donors Profiles snapshots are below

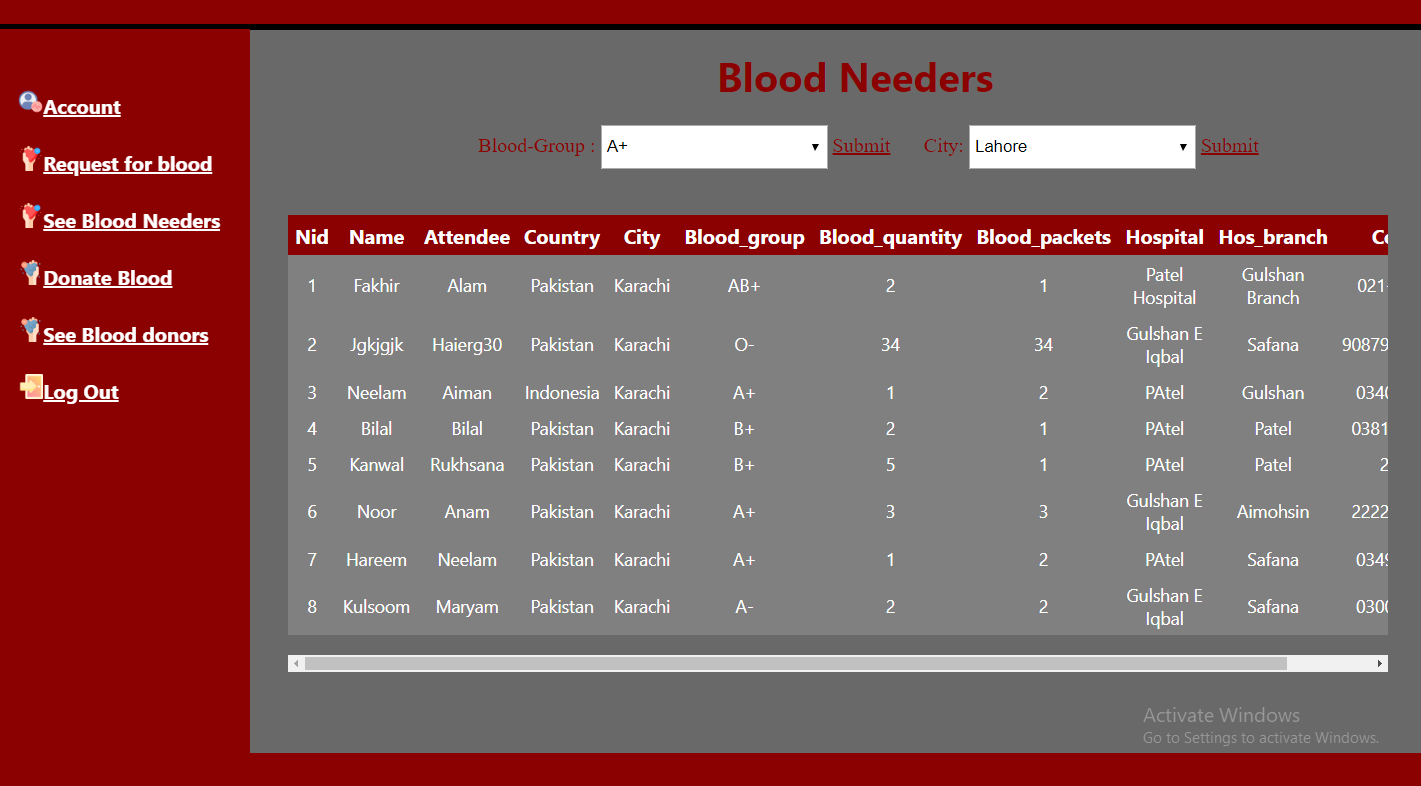
## User Profile

All users that can be Patients, Organization, Hospitals or Self Donors Profiles snapshots are below,

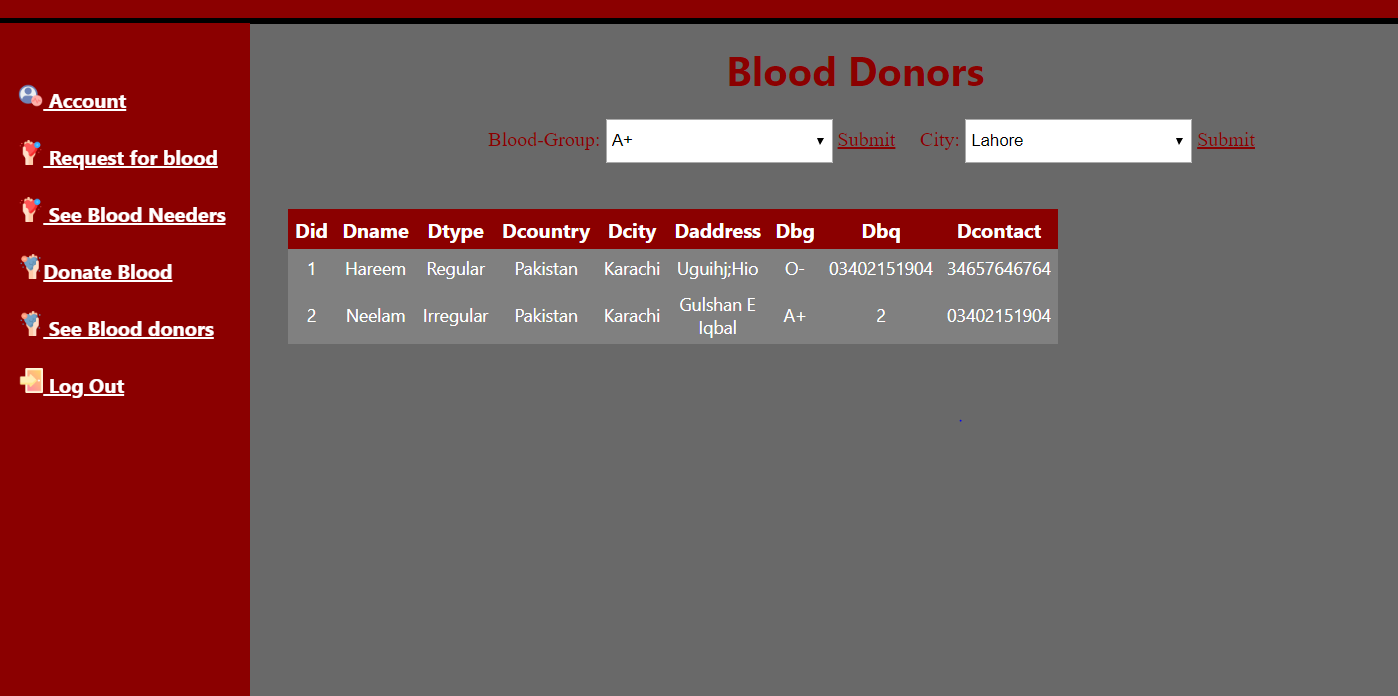
### Account



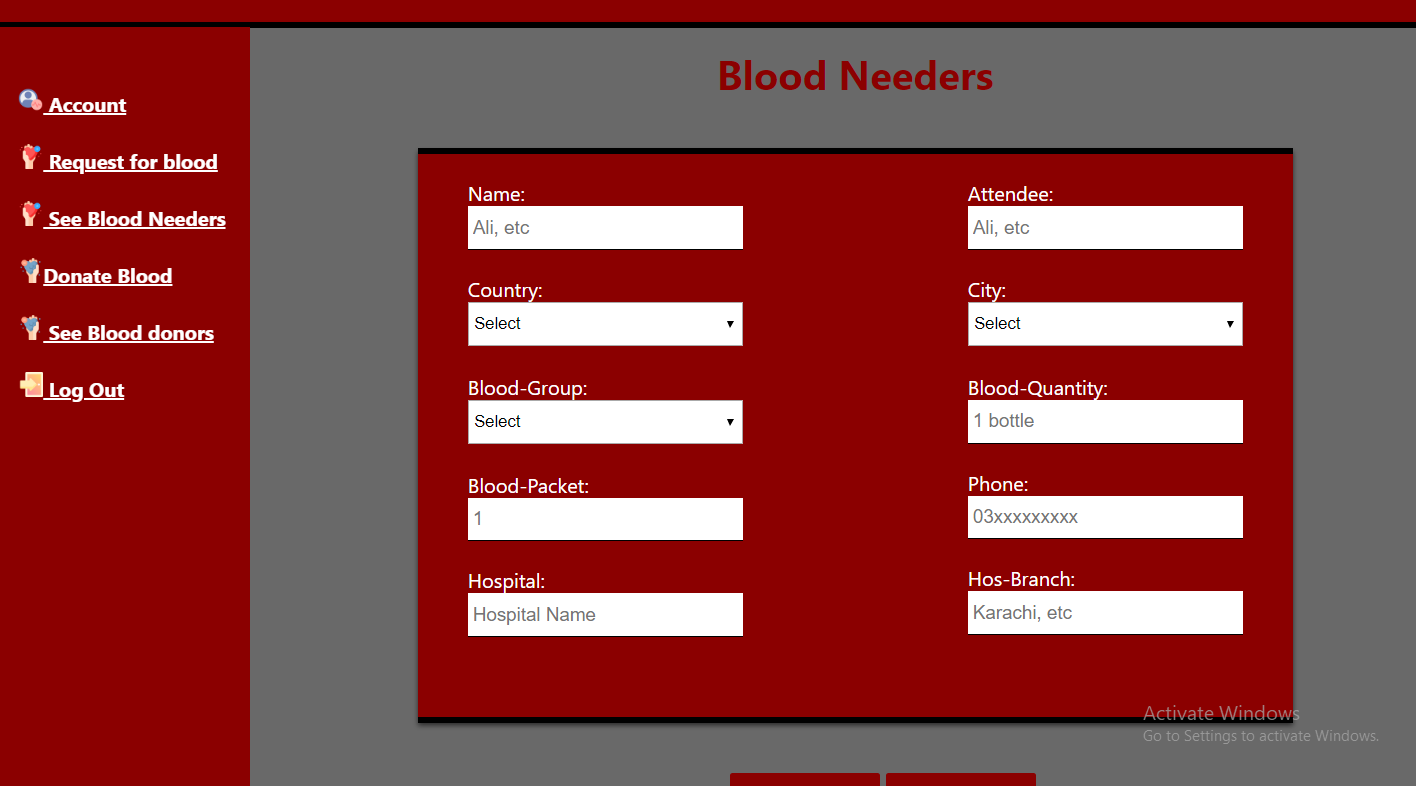
### See Needers



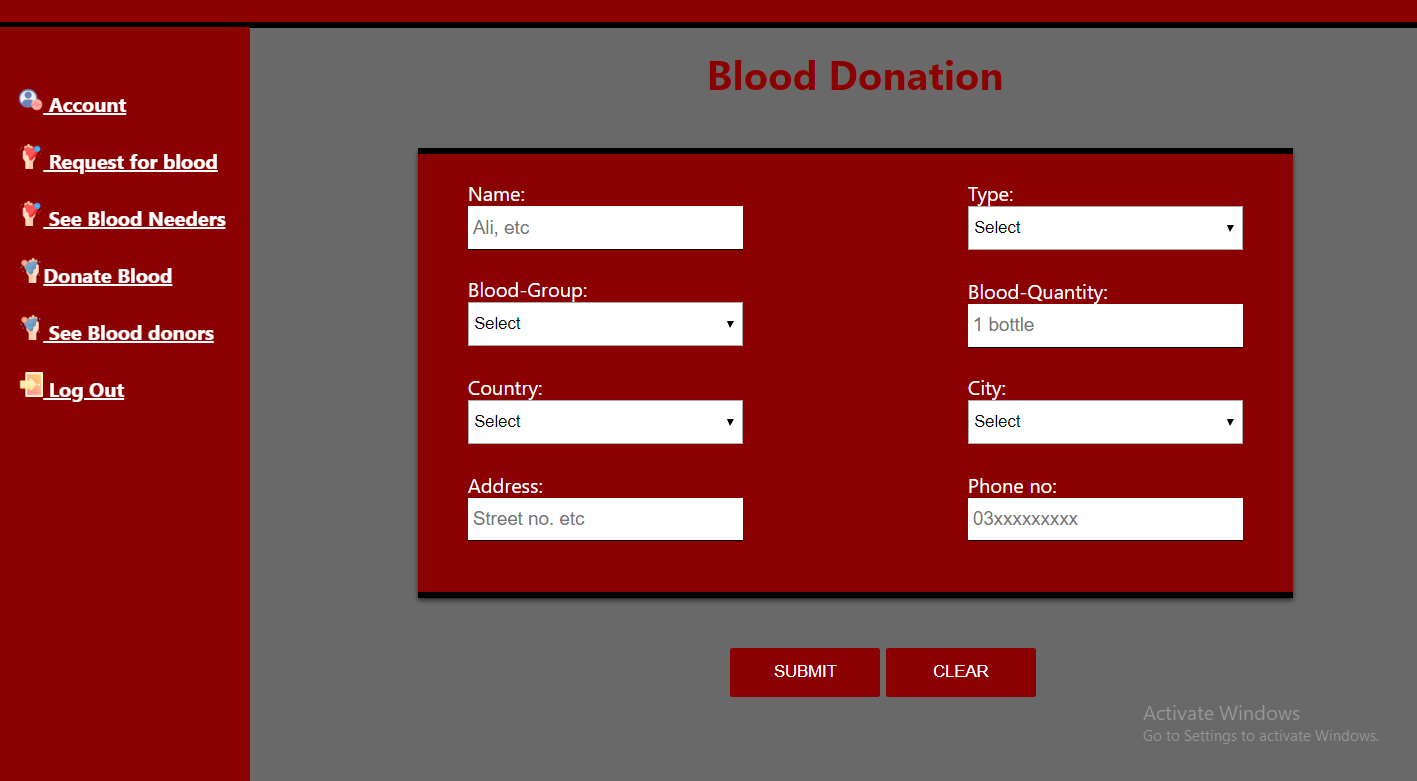
### See Donors



### Needer Registration



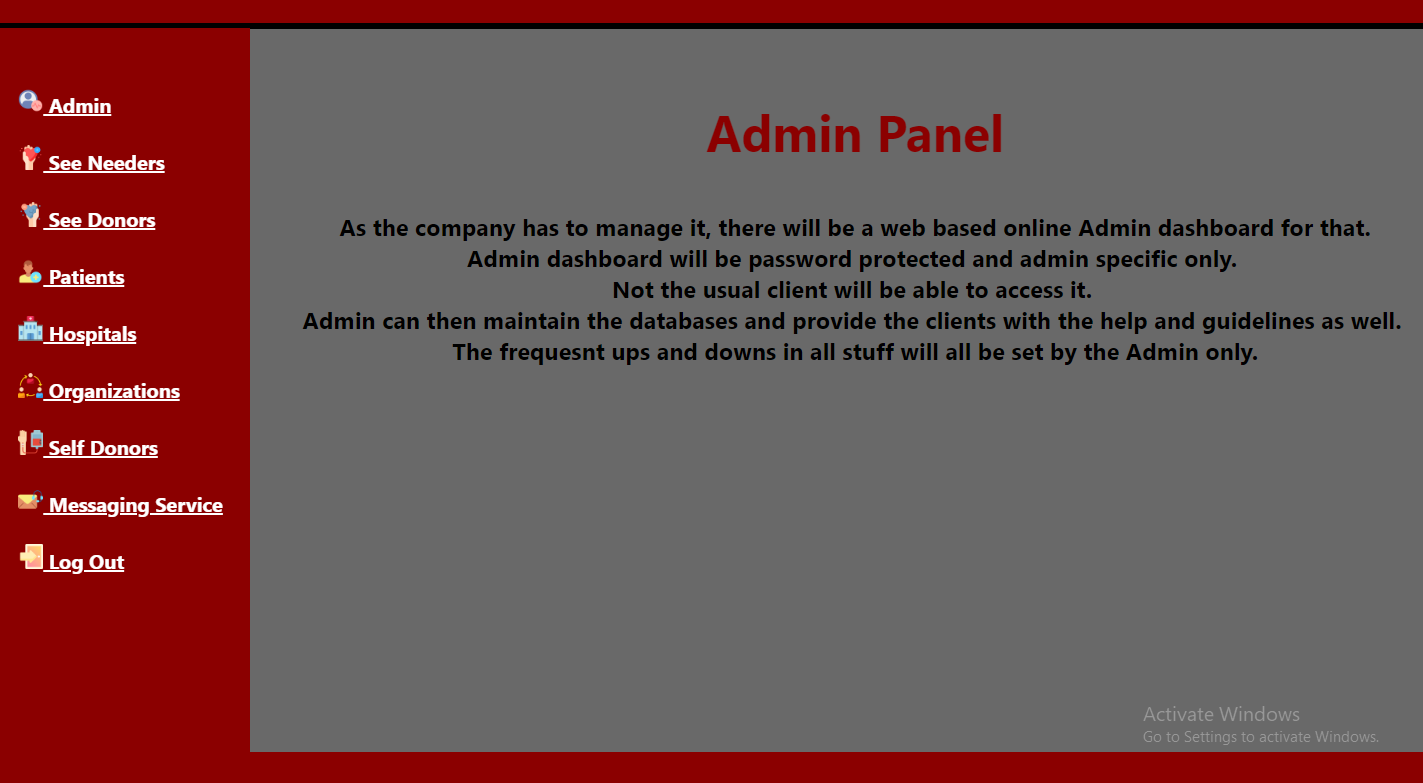
### Donor Registration



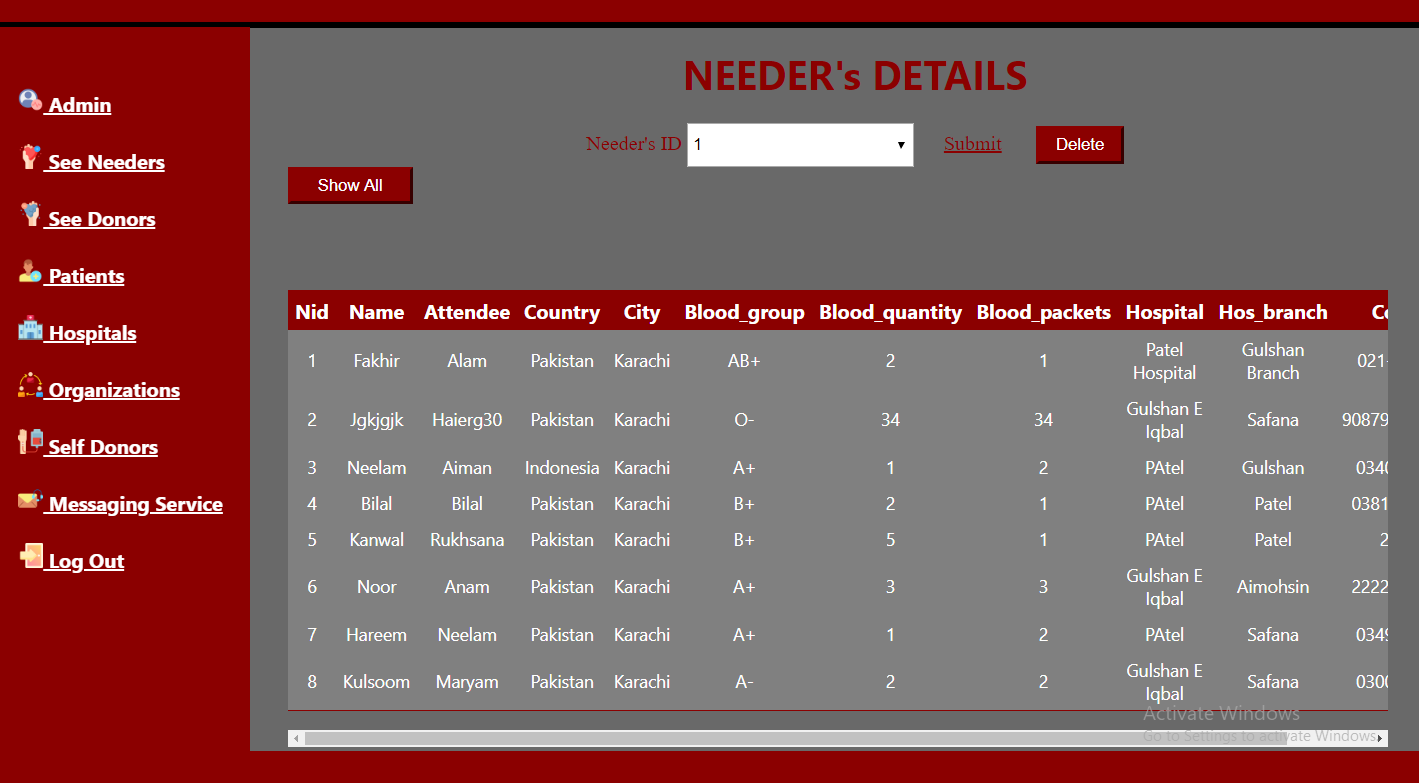
## Admin Profile

Admin will be able to view and modify all information in the database system.

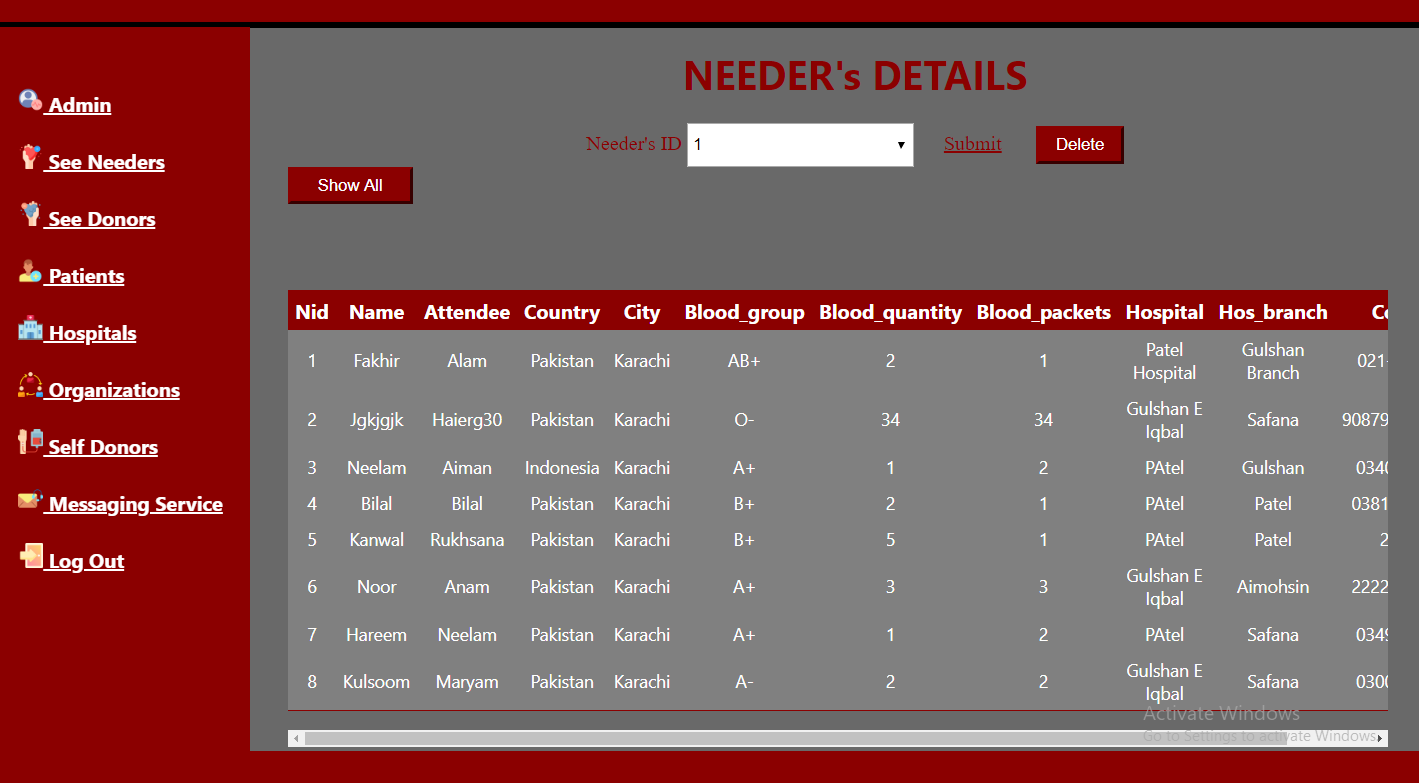
### Admin Panel



### See Needers



### See Donors



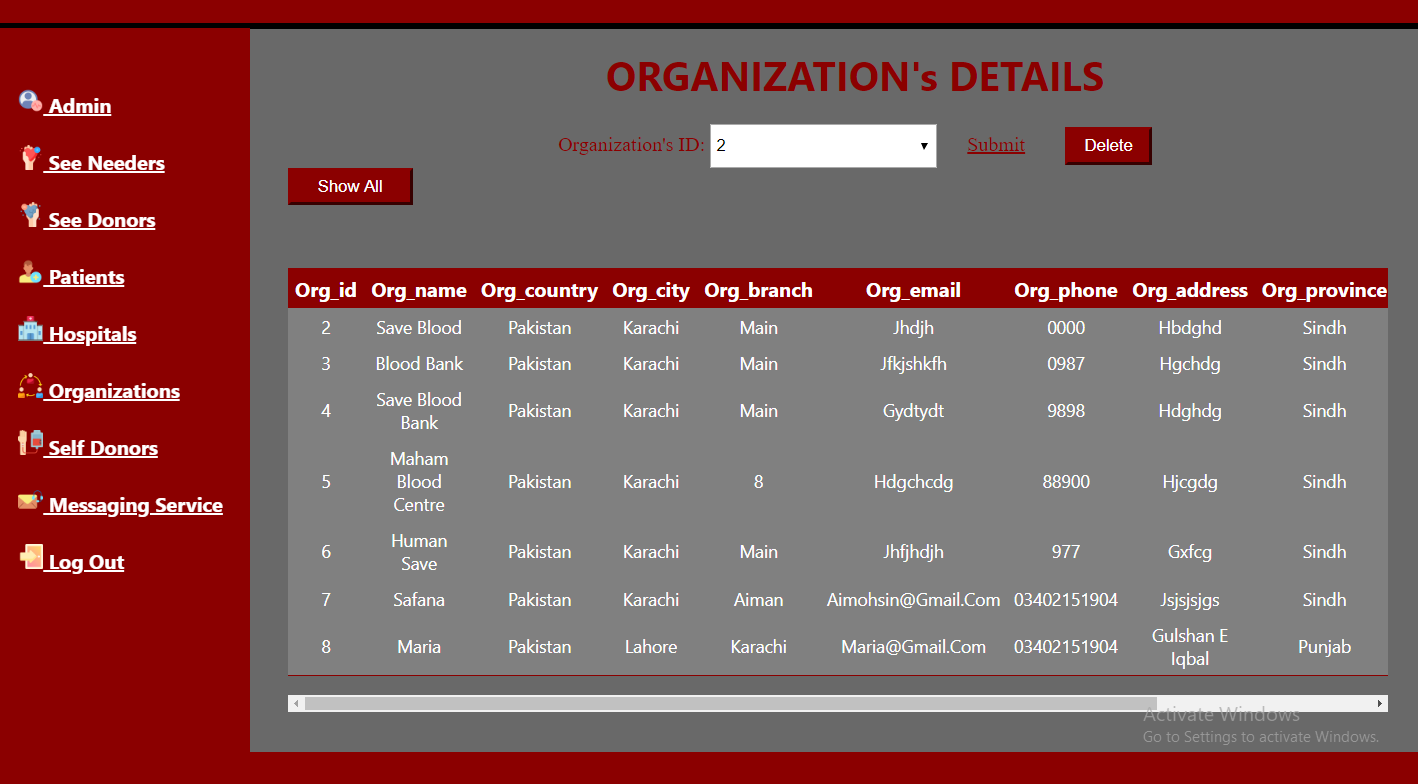
### Patient Details



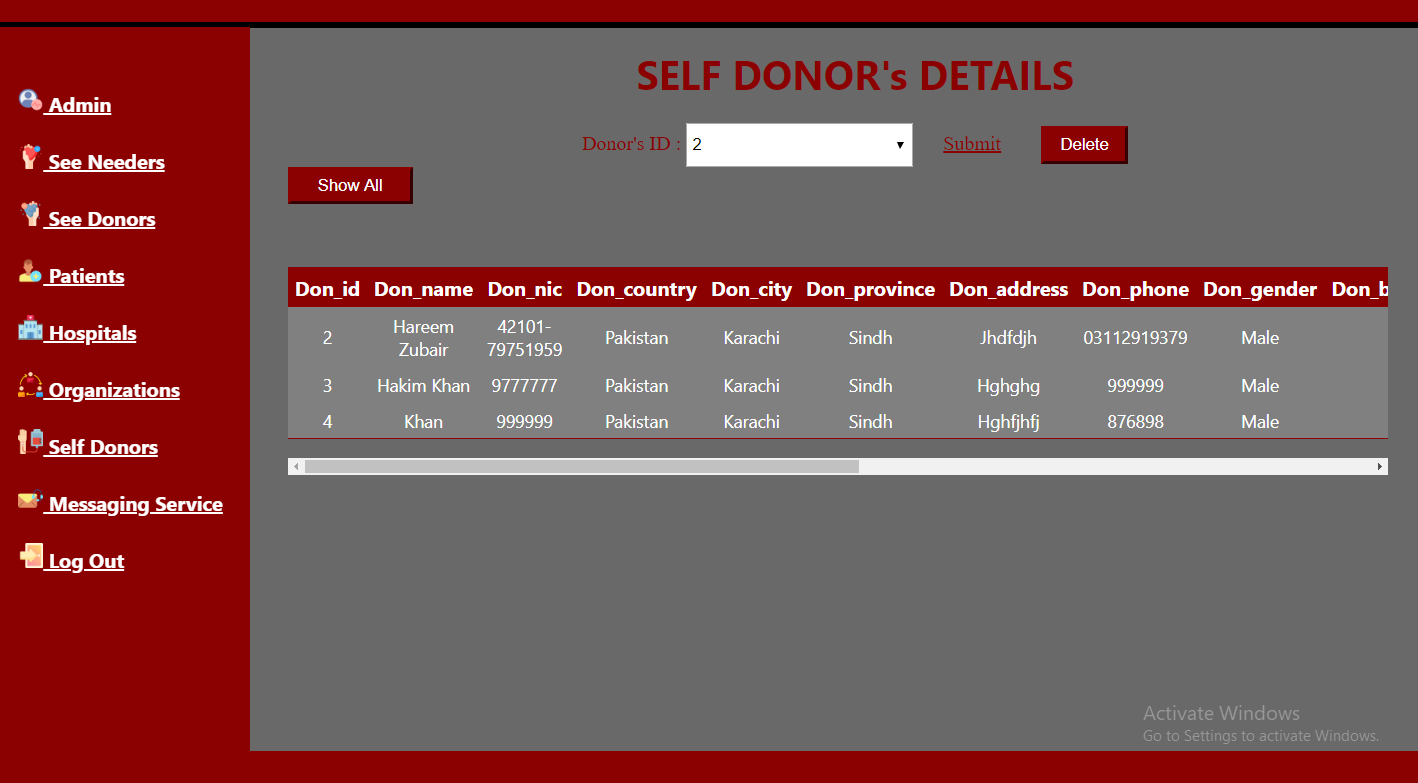
### Hospital Details



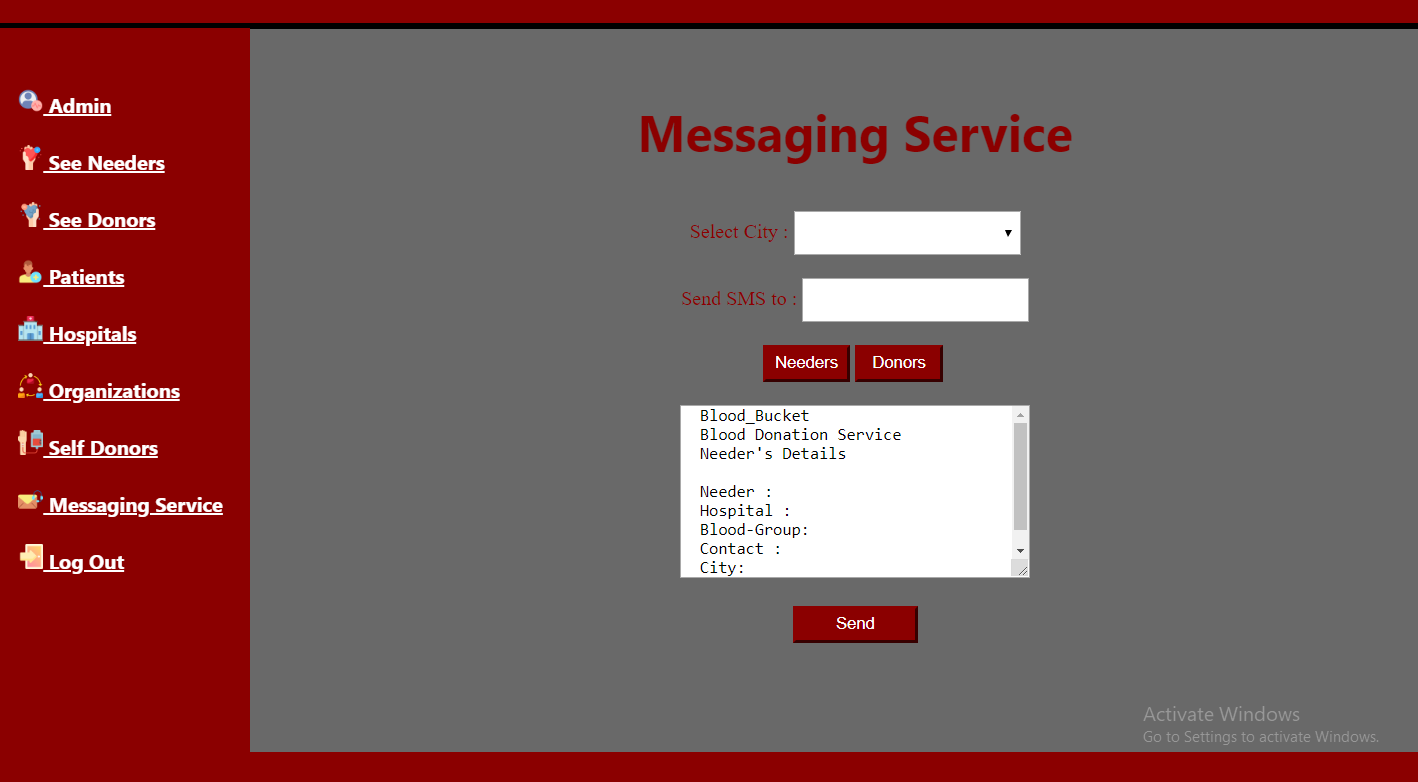
### Organization Details



### Self-Donor Details



### Messaging Service



## Chapter Summary

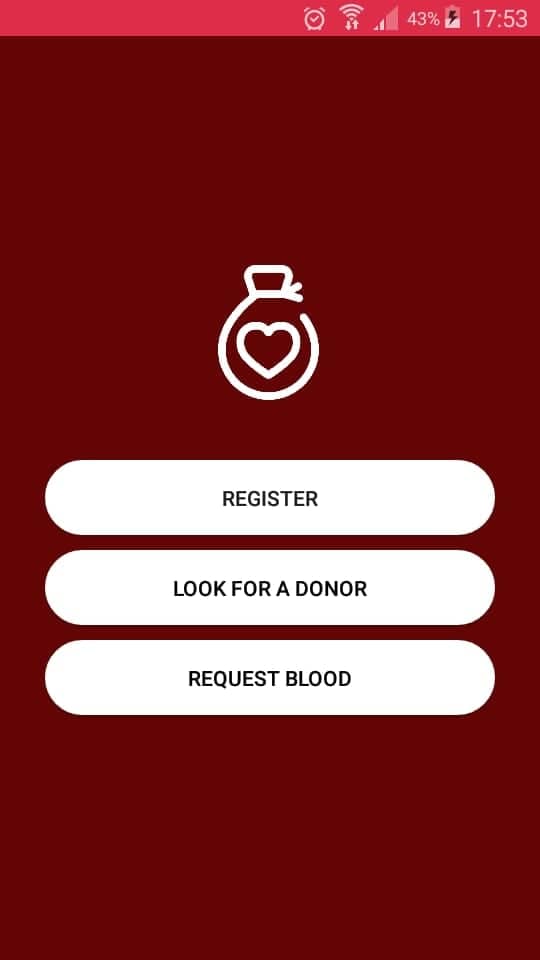
Chapter 5 covers all kind of registrations.

# Needer Panel - Android

The proposed system can be used to reduce the time required to deliver required blood to the needy in cases of emergency

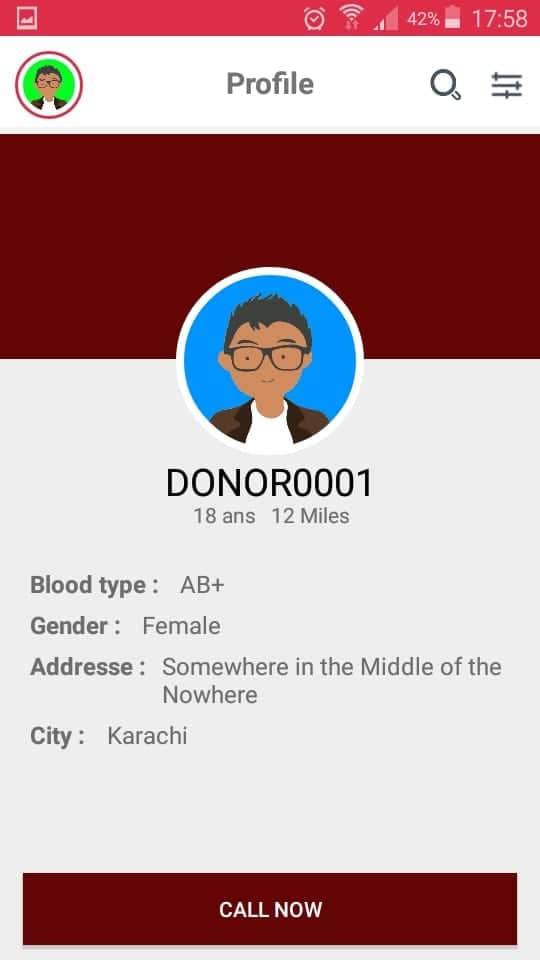
## Home Screen

Home Screen for Needer Panel looks like

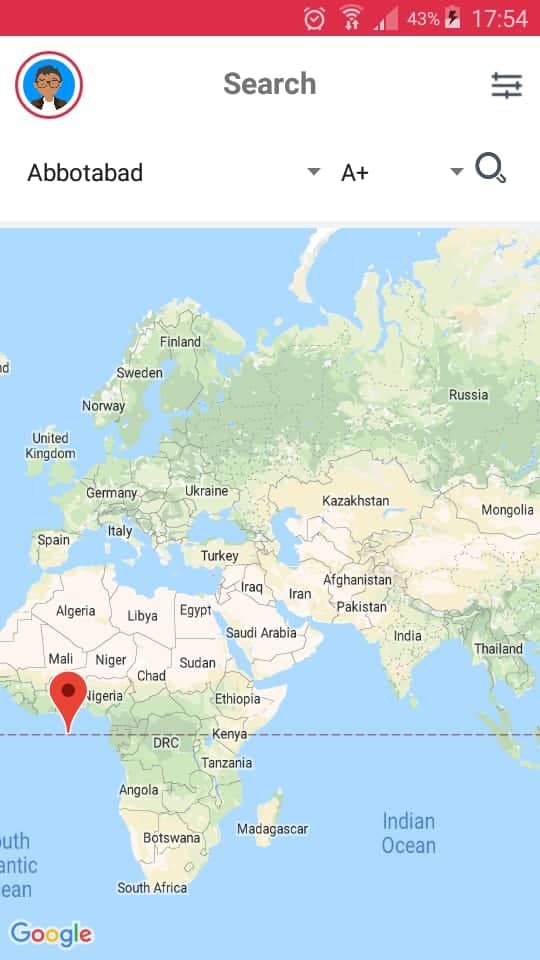
## Profile

Profile for users look like this,

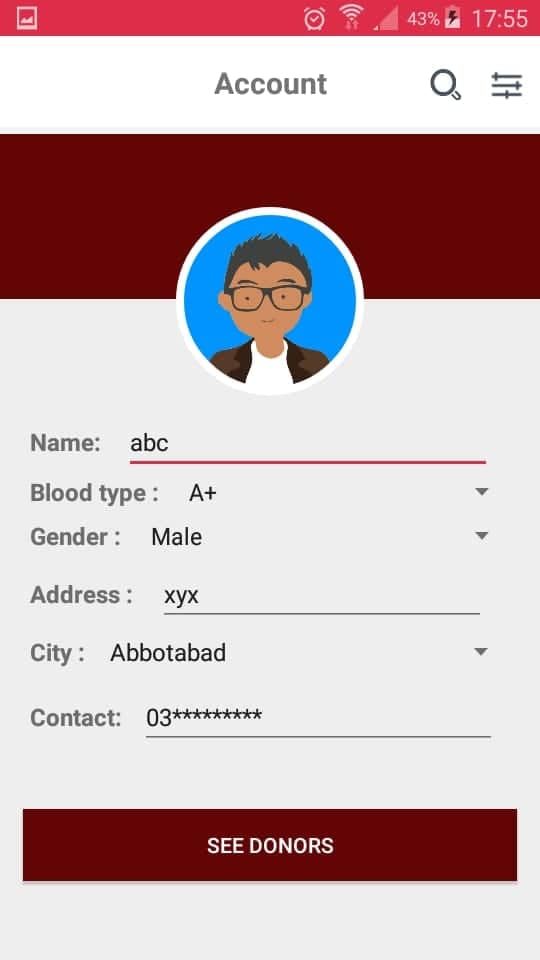
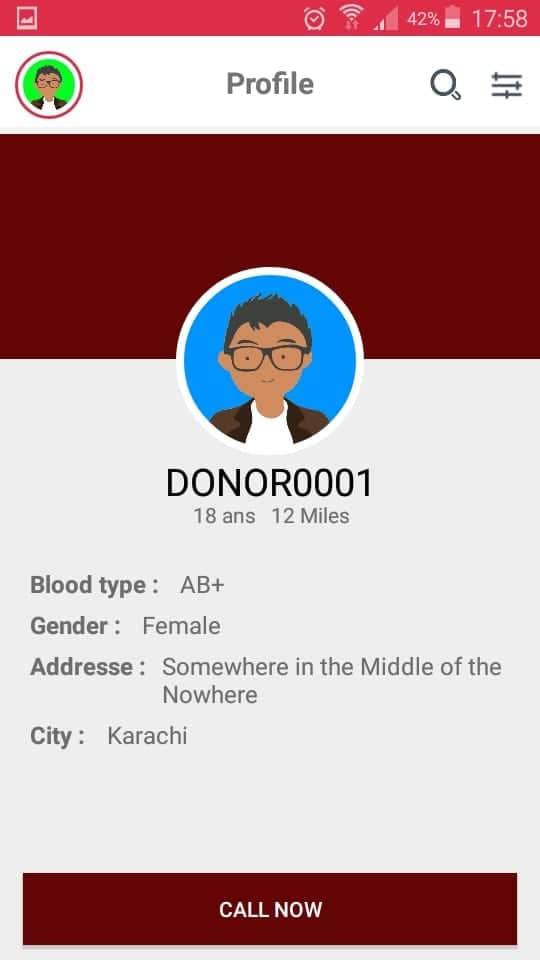


## Nearby Blood Donors using Google Map

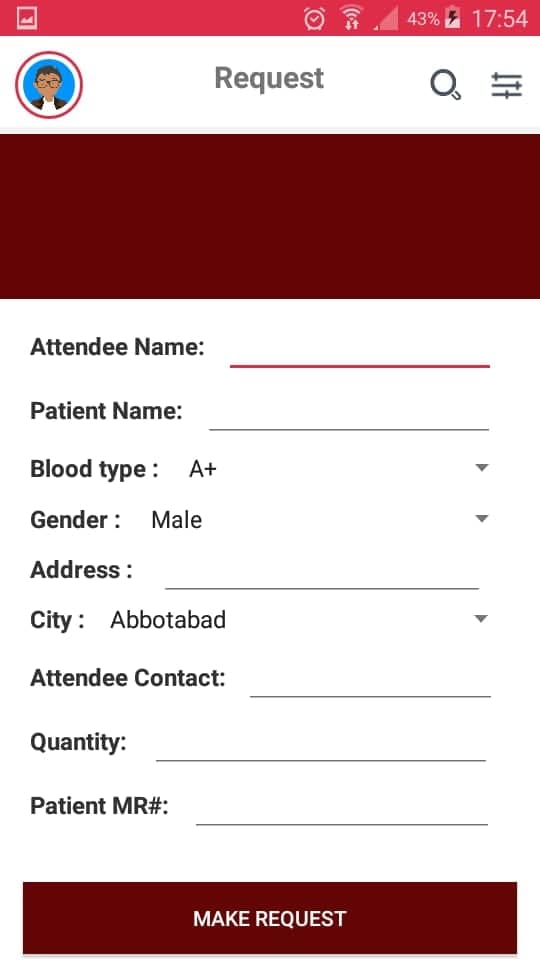
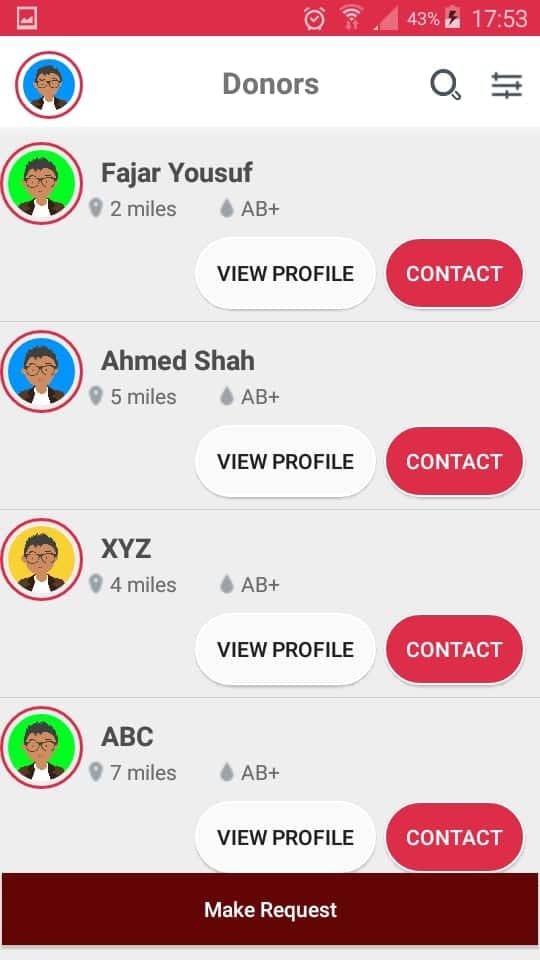
The ‘Nearby Blood Banks’ option opens Google Maps. From this the user can locate the blood banks through red markers. The blue marker gives the user’s current location.



## Donor Profile

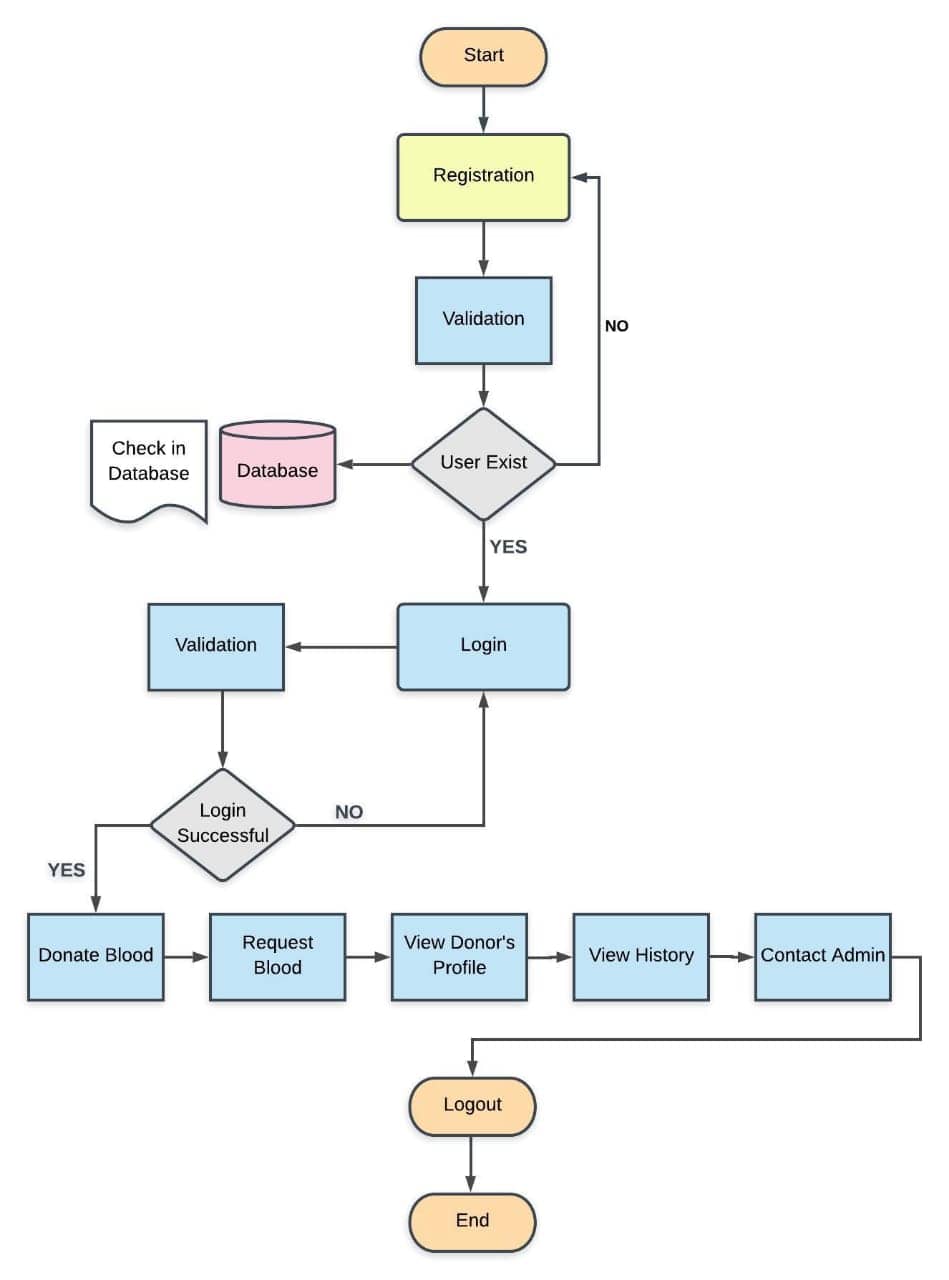
 

## Needer Profile

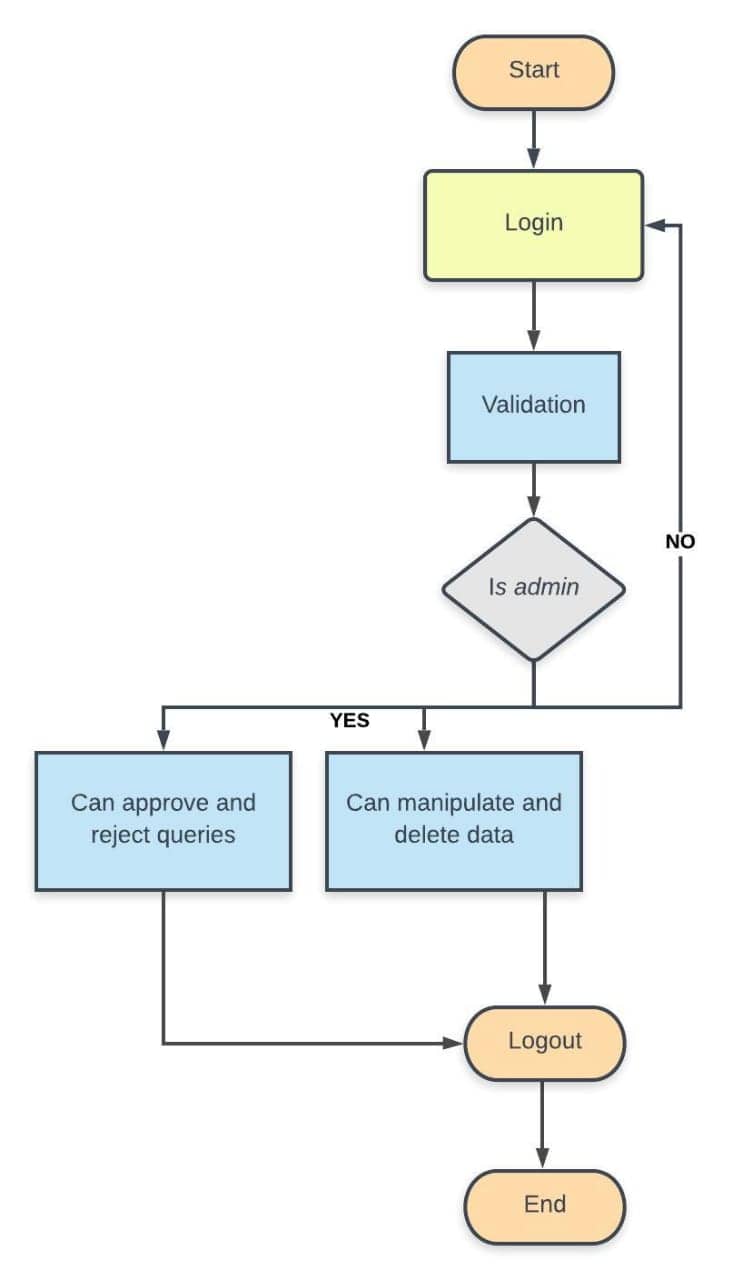
 

# UML Diagrams

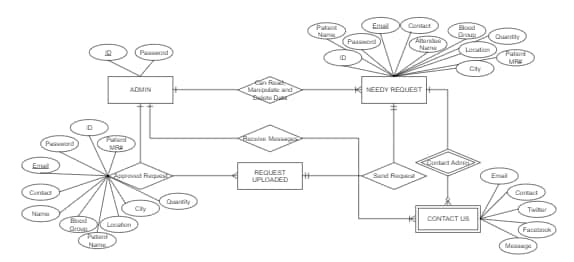
## User Flow Diagram



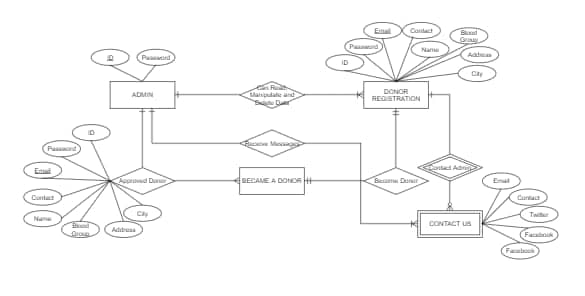
## Admin Flow Diagram



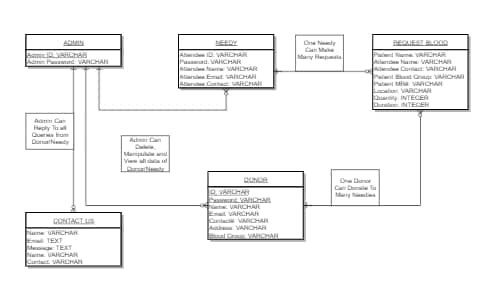
## ERD of Donor + Admin



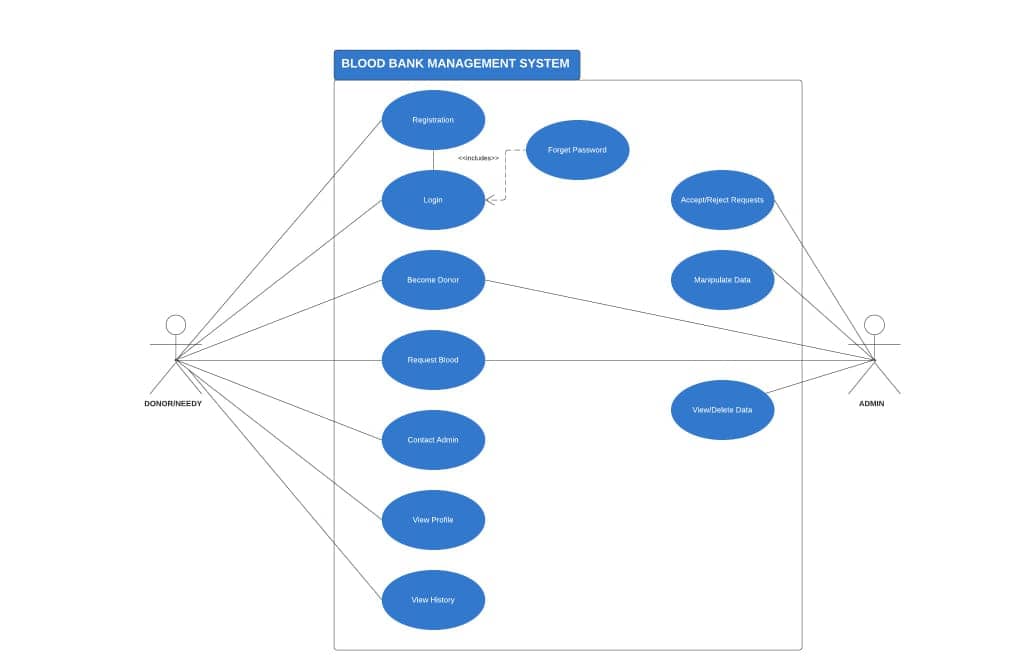
## ERD of Needy + Admin



## Class Diagram



## Use Case Diagram



# Conclusion and Future Enhancement

The proposed system can be used to reduce the time required to deliver required blood to the needy in cases of emergency

## Conclusion

## The Android application can be used by the people interested in donating their blood by locating their nearest blood bank. The web application provides a way of communication and synchronization between the hospitals and the blood banks. It also provides them with the facility of communicating with the nearby donors in emergency. The database is a vital aspect of the system. The database of the hospitals and the blood banks must be checked for consistency on regular basis for smooth working of the system. The proposed system uses Google Maps which provides the user with an efficient way of locating the nearby donors/blood banks. The Android application is developed using Android Studio which is an open source software, while the web application for the hospitals and the blood banks is also developed using open source tools, hence the system developed is quite feasible.

## Future Enhancement

Future Enhancement includes some possible future enhancement that will be considered to achieved. Blood Bucket is designed to help the Blood Bank administrator to meet the demand of Blood by sending and/or serving the request for Blood as and when required.

# References

* Ebling, M. R. (2016). Pervasive Computing and the Internet of Things. *IEEE Pervasive Computing*, *15*(1), 2–4.
* https://doi.org/10.1109/MPRV.2016.7
* <https://www.academia.edu/29746156/Proposal_on_BLOOD_BANK_MANAGEMENT_SYSTEM>
* <https://www.researchgate.net/publication/328305822_Blood_Bank_System>
* <http://portal.scitech.au.edu/darun/wp-content/uploads/2017/07/Senior_ReportFINAL.pdf>
* <https://www.projecttopics.org/design-implementation-online-blood-bank-management-system.html>
* <https://www.shifa.com.pk/blood-bank/>

# Appendix – A: User Manual

Starting Constrain Device Type CD0 TCM = Bastion A1C4

# Appendix – B: Coding

**Website**

**Site.Master:**

<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="Site1.master.cs" Inherits="blood\_bucket.Site1" %>

<!DOCTYPE html>

<html>

<head runat="server">

<title>Blood Bucket</title>

<asp:ContentPlaceHolder ID="head" runat="server">

<link rel="stylesheet" href="css/font-awesome.min.css">

<link rel="stylesheet" href="css/bootstrap.css">

<link href="style/StyleSheet1.css" rel="stylesheet" />

</asp:ContentPlaceHolder>

</head>

<body>

<form id="form1" runat="server">

<div id="cont" style="background:dimgray; height:100%; width:100%;">

<div id="header" class="fixed-top" style="top:0; height: 60px; width: 100%; background-color: darkred;"></div>

<div id="navigation" class="fixed-top" style="border-bottom: white 3px solid; margin: 0; padding:0; height:40px; width:100%; background-color:darkred;">

<ul class="navbar-nav ml-auto" style="nav-left:auto">

<li class="nav-item">

<a href="#home" class="nav-link">Home</a>

</li>

<li class="nav-item">

<a href="#explore-head-section" class="nav-link">About</a>

</li>

<li class="nav-item">

<a href="#create-head-section" class="nav-link">Contact</a>

</li>

<li class="nav-item">

<a href="#share-head-section" class="nav-link">Login</a>

</li>

</ul>

<div id="menu" style="float: right; font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;">

<asp:Menu ID="Menu1" runat="server" DataSourceID="SiteMapDataSource1" Height="40px" Orientation="Horizontal" StaticDisplayLevels="2" Width="100%">

<StaticMenuItemStyle Font-Size="Large" ForeColor="White" HorizontalPadding="25px" VerticalPadding="5px" />

</asp:Menu>

<asp:SiteMapDataSource ID="SiteMapDataSource1" runat="server" />

</div>

</div>

<div id="mainContent">

<asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">

</asp:ContentPlaceHolder>

</div>

<div id="bodyfooter" style="padding: -10px; padding-top:60px;"></div>

<div id="footer" style="position: absolute; right: 0; bottom: 10; left: 0; height: 150px; width: 100%; background-color: darkred;" ></div>

</div>

</div>

</form>

<script src="js/jquery.min.js"></script>

<script src="js/popper.min.js"></script>

<script src="js/bootstrap.min.js"></script>

</body>

</html>

**Login:**

<%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master" AutoEventWireup="true" CodeBehind="frmlogin.aspx.cs" Inherits="blood\_bucket.frmlogin" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<header>

<link rel="stylesheet" href="css/bootstrap.css">

<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.6.1/css/all.css" integrity="sha384-gfdkjb5BdAXd+lj+gudLWI+BXq4IuLW5IT+brZEZsLFm++aCMlF1V92rMkPaX4PP" crossorigin="anonymous">

</header>

<h1 style="color:darkred; font-family:'Segoe UI', Tahoma, Geneva, Verdana, sans-serif; text-align:center;">LOGIN PAGE</h1>

<a href="/frmAdmin.aspx" class="btn-light">

<i class="fas" ></i>Sign In as an ADMIN

</a>

<div id="login-box">

<br /><br />

Login Id :

<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox><br /><br />

Login Password :

<asp:TextBox ID="TextBox2" runat="server"></asp:TextBox><br /><br />

<asp:LinkButton ID="LinkButton1" runat="server" ForeColor="DarkGray" OnClick="LinkButton1\_Click1">New Member? Click over here!</asp:LinkButton><br />

<div class="btnRegister">

<asp:Button ID="btnlogin" runat="server" OnClick="btnlogin\_Click" Text="Login" BackColor="DimGray" ForeColor="DarkRed" Font-Size="Medium" BorderColor="DimGray" Height="35px" Width="90px" /><br /></div>

<asp:Label ID="lblmessage" runat="server" ForeColor="YellowGreen"></asp:Label>

</div>

</asp:Content>

**Registration:**

<%@ Page Title="" Language="C#" MasterPageFile="~/Site1.Master" AutoEventWireup="true" CodeBehind="frmpat\_reg.aspx.cs" Inherits="blood\_bucket.frmpat\_reg" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<style>

.btnRegister{

position: absolute;

top: 850px;

left: 530px;

text-align: center;

}

</style>

<h1 style="color:darkred; font-family:'Segoe UI', Tahoma, Geneva, Verdana, sans-serif; text-align:center;">PATIENT REGISTRATION</h1>

<div id="login-box">

<div class="left-box">

MR No# :

<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>

Patient User ID :

<asp:TextBox ID="TextBox8" runat="server"></asp:TextBox>

Patient User Password:

<asp:TextBox ID="TextBox9" runat="server"></asp:TextBox>

CNIC :

<asp:TextBox ID="TextBox5" runat="server" ></asp:TextBox>

City :<br />

<asp:DropDownList ID="DropDownList2" runat="server" Height="35px" Width="100%">

<asp:ListItem>Karachi</asp:ListItem>

<asp:ListItem>Islamabad</asp:ListItem>

<asp:ListItem>Lahore</asp:ListItem>

<asp:ListItem>Peshawar</asp:ListItem>

<asp:ListItem>Quetta</asp:ListItem>

<asp:ListItem></asp:ListItem>

</asp:DropDownList><br /><br />

Hospital :

<asp:TextBox ID="TextBox6" runat="server"></asp:TextBox>

Patient Type :<br />

<asp:DropDownList ID="DropDownList3" runat="server" Height="35px" Width="100%">

<asp:ListItem>Regular</asp:ListItem>

<asp:ListItem>Irregular</asp:ListItem>

<asp:ListItem></asp:ListItem>

</asp:DropDownList><br /><br />

Address :

<asp:TextBox ID="TextBox3" runat="server"></asp:TextBox>

</div>

<div class="right-box">

Name

<asp:TextBox ID="TextBox2" runat="server"></asp:TextBox>

Re-Type User ID:

<asp:TextBox ID="TextBox11" runat="server"></asp:TextBox>

Re-Type User Password:

<asp:TextBox ID="TextBox12" runat="server"></asp:TextBox>

Phone :

<asp:TextBox ID="TextBox4" runat="server" MaxLength="11">

</asp:TextBox>

Country :<br />

<asp:DropDownList ID="DropDownList1" runat="server" Height="35px" Width="100%">

<asp:ListItem>Pakistan</asp:ListItem>

<asp:ListItem>China</asp:ListItem>

<asp:ListItem>Indonesia</asp:ListItem>

<asp:ListItem>United Kingdom</asp:ListItem>

<asp:ListItem>America</asp:ListItem>

<asp:ListItem>UAE</asp:ListItem>

<asp:ListItem></asp:ListItem>

</asp:DropDownList><br /><br />

Hospital Branch :

<asp:TextBox ID="TextBox7" runat="server"></asp:TextBox>

Attendee Name :

<asp:TextBox ID="TextBox10" runat="server"></asp:TextBox>

</div>

<p>

<asp:Label ID="Label1" runat="server"></asp:Label>

</p>

<p>

<asp:Label ID="Label2" runat="server"></asp:Label>

</p>

</div>

<div class="btnRegister">

<asp:Button ID="btnsubmit" runat="server" OnClick="btnsubmit\_Click" Text="Submit" />

<asp:Button ID="btnclear" runat="server" Text="Clear" OnClick="btnclear\_Click" />

</div>

</asp:Content>