

Acad. Year
2020-21

Blood Bank Management System

Blood Bank Management System



**INDIAN INSTITUTE OF INFORMATION
TECHNOLOGY GUWAHATI**

Gurnoor Singh 1801065

Kumar Badal 1801094

Guided By: Dr.Rakesh Matam

Dr.Angshuman Jana

COMPUTER SCIENCE ENGINEERING DEPARTMENT

Contents

- 1. Abstract**
- 2. Introduction**
- 3. Functional Requirements**
- 4. Non-Functional Requirements**
- 5. Data Flow Diagram**
- 6. Use Case Diagram**
- 7. E-R Diagram**
- 8. Sequence of Application**
- 9. Implementation**
- 10. Technology Stack**
- 11. Future Works**
- 12. Limitations**
- 13. Constraints**
- 14. Testing methods**
- 15. Contributions**

ABSTRACT

The manual system of managing the blood is quite obsolete. So, to reduce the turnaround time, this app creates a nexus between the Donor and Recipient digitally. In case of emergency, this application can save many lives as it ensures proper maintenance of records of donors and recipients and shows the description of all Blood Banks in the vicinity of 50kms radius from your present location.

INTRODUCTION

Introduction contain the following sub categories

Existing System

The present system is a manual system. Manual system involves paper work in the form of maintaining various files and manuals. Maintaining critical information in the files and manuals is full of risk and a tedious process. Here, the nexus between Donor and recipient doesn't exist, every time Recipient/Donor has to locate manually the Blood Bank and check that the required Blood is available/required or not.

A manual system has following disadvantages

- Searching of particular information is critical it takes lot of time.
- Proper safety Donor/Seeker might not be there, amid ongoing pandemic
- In the manual system total turnaround time will be very high.
- The existing system need to travel a location check the availability of the Blood
- The existing system needs to save the information in the form of files.
- Limited sharing is possible if the data is in the form of paper or disk drives.
- The manual system gives us less security for saving data, some data may be lost due to mismanagement that means less transparency will be there.

Proposed System

This web application aims to serve the need for accessing blood services amid ongoing pandemic. There are many advantages to everyone involved in this whole ecosystem. The system after careful analysis has been identified to be presented with the following modules:

- **Donor Registration:** Donors can register for donation of the blood.
- **Recipient Registration:** Recipients can register to contact the donors.
- **Search:** By specifying the Blood Group and Pin Code clients can find the Donor list
- **Donor:** List of donors will be specified.
- **Record Storage:** The user information files should be stored in a centralized database which can be maintained by the system.

- **Blood Banks Nearby:** Shows all the Blood Banks in the vicinity of 50kms radius from your present location.
- **Administration :** Administration would be able to keep an eye on the records of Donors/Recipients.
- **Authentication:** Authentication of these donor details will be provided for only registered members.

Advantages for Blood Seekers

Advantages for Blood Seekers are explained below:

- **Ease of Accessibility:** At a time when the country is facing such a pandemic, this Web Application will provide succor to all those who direly require blood.
- **Donor Repository:** Recipient can directly access the repository of Donor which will help in case of emergency which in turn will reduce the turnaround time of the whole process.
- **Single Window System and Transparent :** This Web Application allow blood seeker to obtain blood & brings the added advantage of complete transparency & single window access to the service.

Advantages for Donors

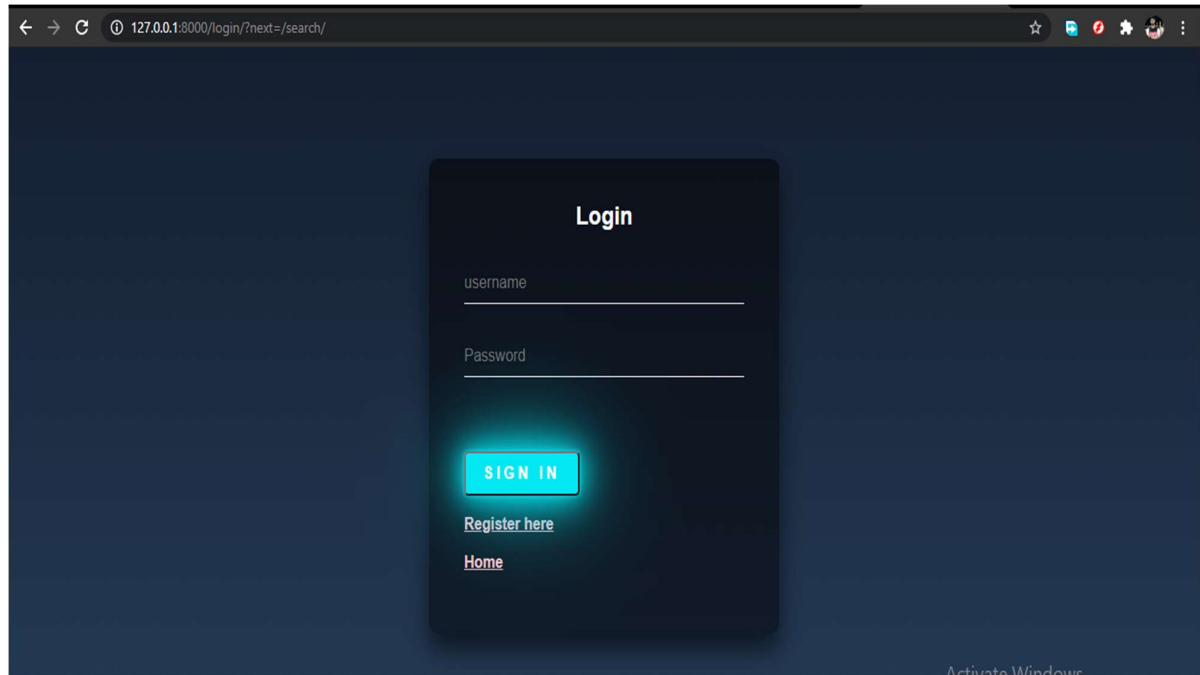
Advantages for Donors are explained below:

- **Restrict professional Donors:** Only registered Donors can participate in the process which will in turn restrict the professional donors to mint money.
- **Safe and adequate Blood supply:** All the Blood samples from Donors will be first tested and then designated accordingly to the Recipient. This will maintain a safe Concatenation with adequate Blood supply.
- **Help in following COVID protocols:** Donor repositories will be maintained , which will help in collecting Blood units from their respective locations

FUNCTIONAL REQUIREMENTS

Functional requirements of our system are explained below.

1. The admin requires Username and password to access all the details of Donors and Blood seekers



2. Donor/Recipient can register on the portal by entering their required credentials, all their Data will be saved.

REGISTRATION FORM

[Home](#)

Username* Required. 150 characters or fewer. Letters, digits and @/!+!-_ only.

Password*

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation* Enter the same password as before, for verification.

Name*

Usertype* donor

Email*

Phone no*

Pincode*

Address*

Blood type* A+

Allergies* Yes

Cardiac disease* Yes

Bleeding disorder* Yes

Activate Windows
Go to Settings to activate Windows.

- Recipients can select blood group and location to get the donor list of that particular area.

SEARCH

[Home](#)

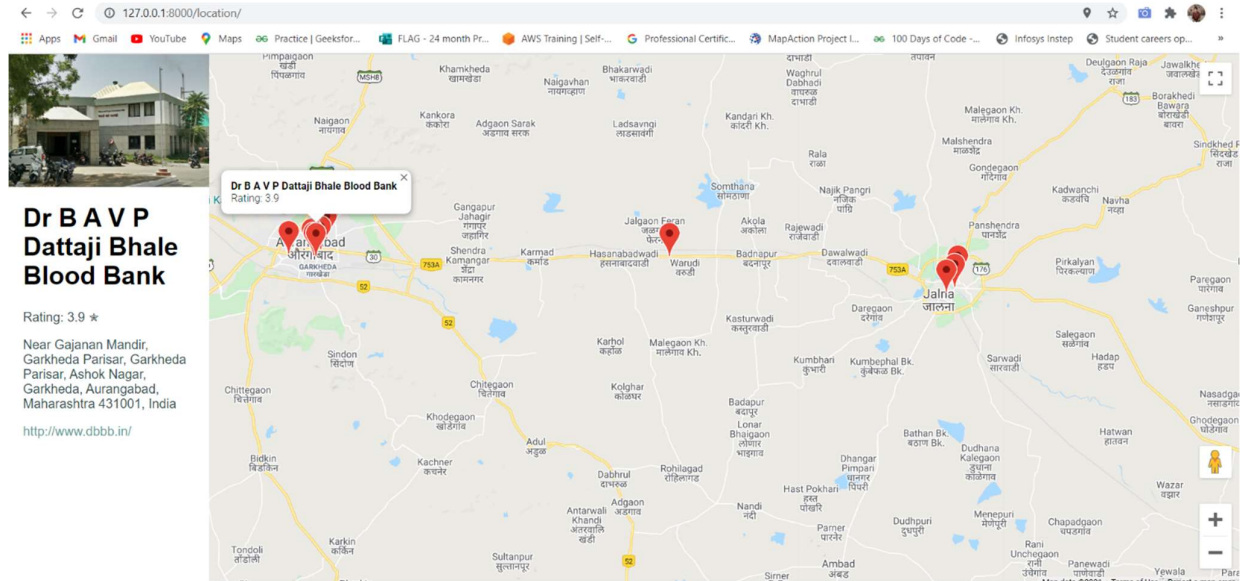
Blood type ----- Pincode

DONOR LIST

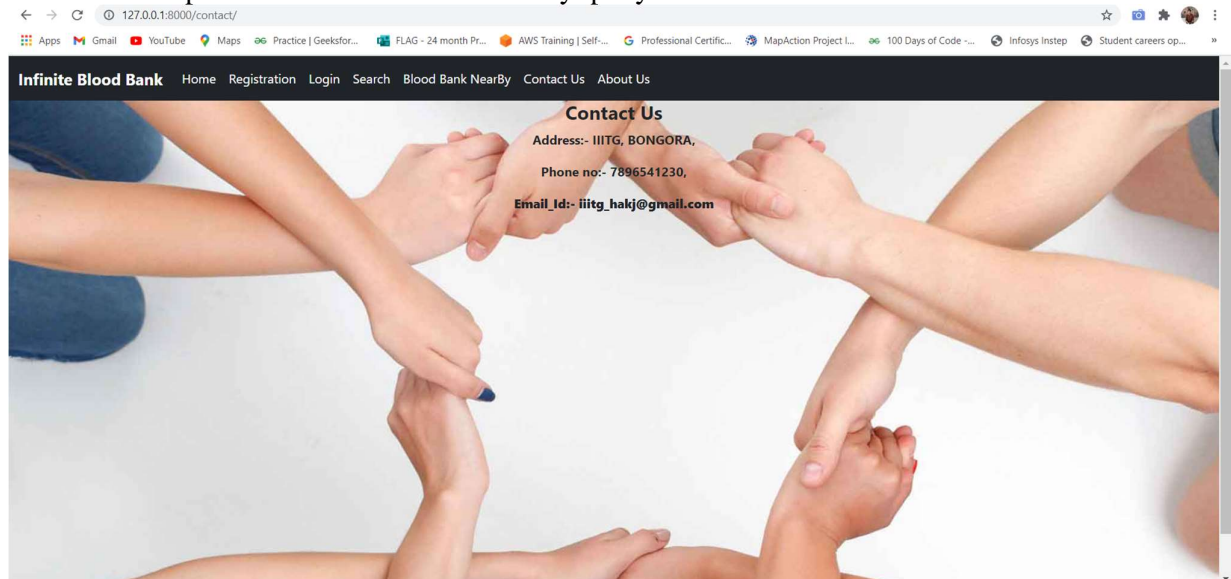
| Name | Emails | Blood_group | Phone_no | pincode | address | allegeries | cardiac_disease | bleeding_disorder |
|---------|-----------------------|-------------|------------|---------|---------|------------|-----------------|-------------------|
| raka | raka@gmail.com | A+ | 9512364780 | 781015 | liitg | yes | yes | yes |
| vanu | vanu@gmail.com | A+ | 7410258963 | 781015 | liitg | yes | yes | yes |
| ranu | ranu@gmail.com | A+ | 7410258963 | 781015 | liitg | yes | yes | yes |
| abhi | kumar@gmail.com | O- | 0369852147 | 781015 | liitg | yes | yes | yes |
| gurnoor | gurnoor_123@gmail.com | A+ | 0789654123 | 851015 | liitg | yes | yes | yes |
| def | def@gmail.com | AB- | 0369852147 | 440001 | nagpur | no | no | no |

Activate Windows

- All the Blood Banks with their credentials will be displayed on one of the web page on Google Maps



- There will be one webpage displaying the contact details of the Developer/Administrator in case of any query.



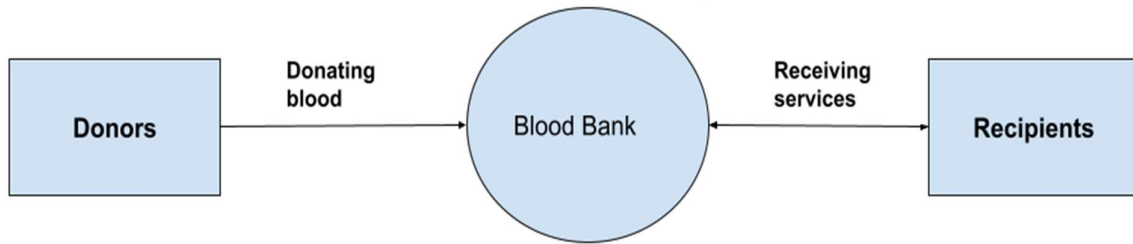
NON-FUNCTIONAL REQUIREMENTS

Requirement that specifies criteria that can be used to judge the operation of a system are called non-functional requirements. Non-functional requirements of our system are mentioned below:

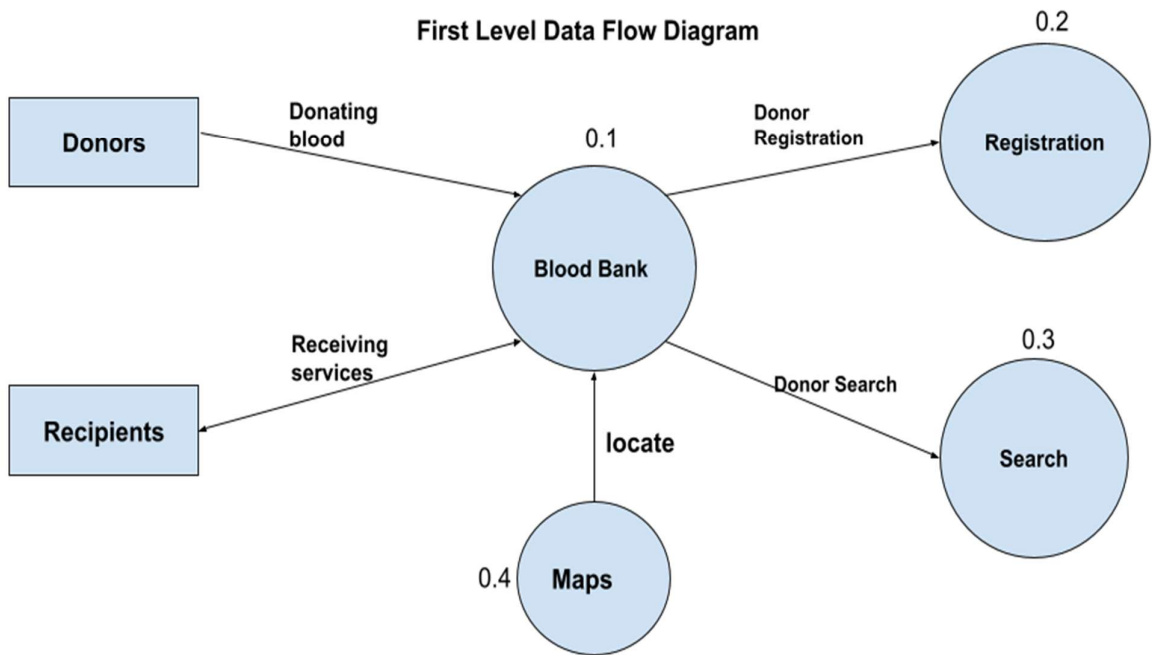
1. **Possess reliability:** Administrator will maintain privacy in case if any client has disease.
2. **Flexible:** will be easy to amend if there's a need of extension and addition of new features in future
3. Recipient details can be accessed by Administrator only.

DATA FLOW DIAGRAM

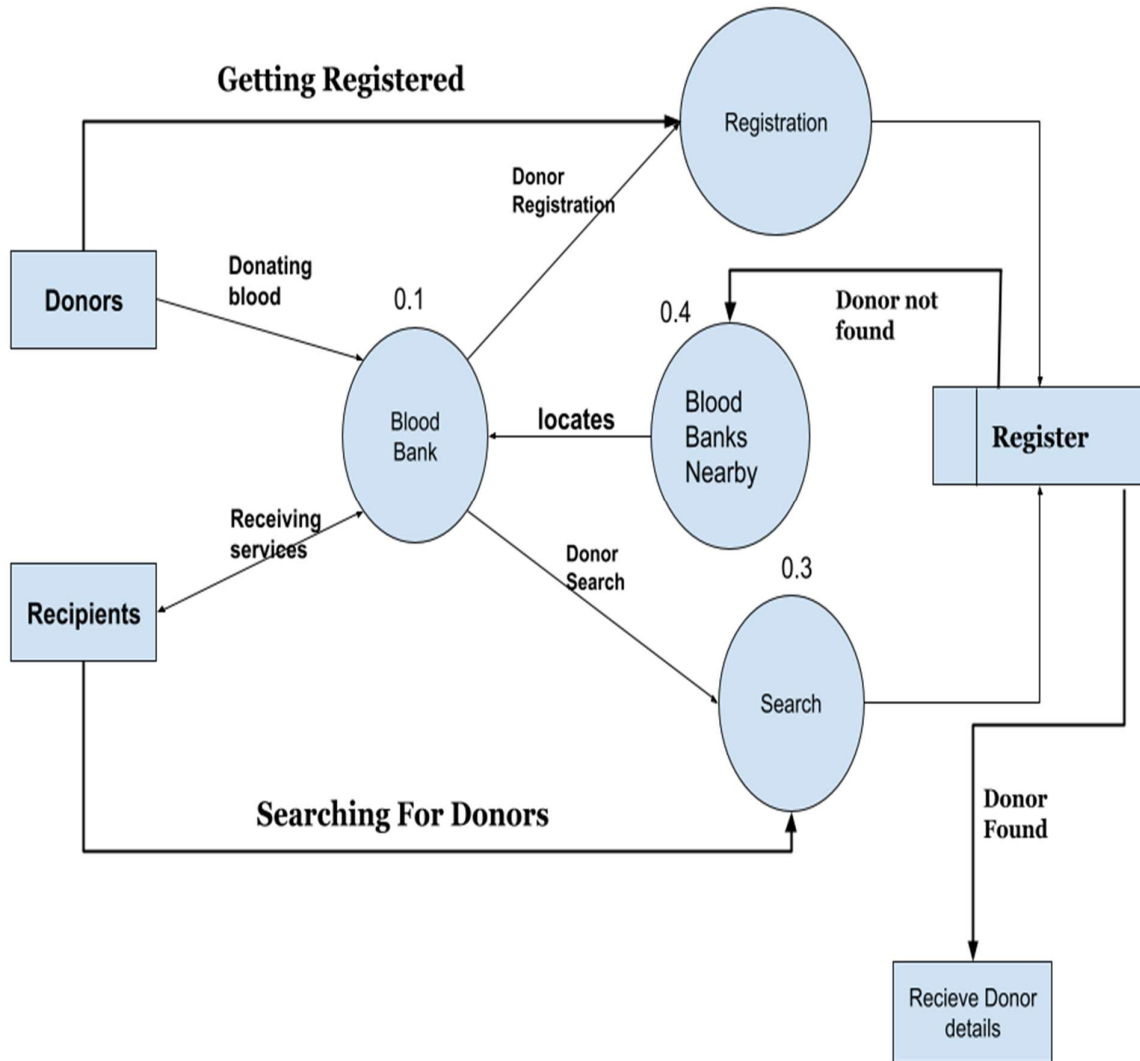
Zero Level Data Flow Diagram



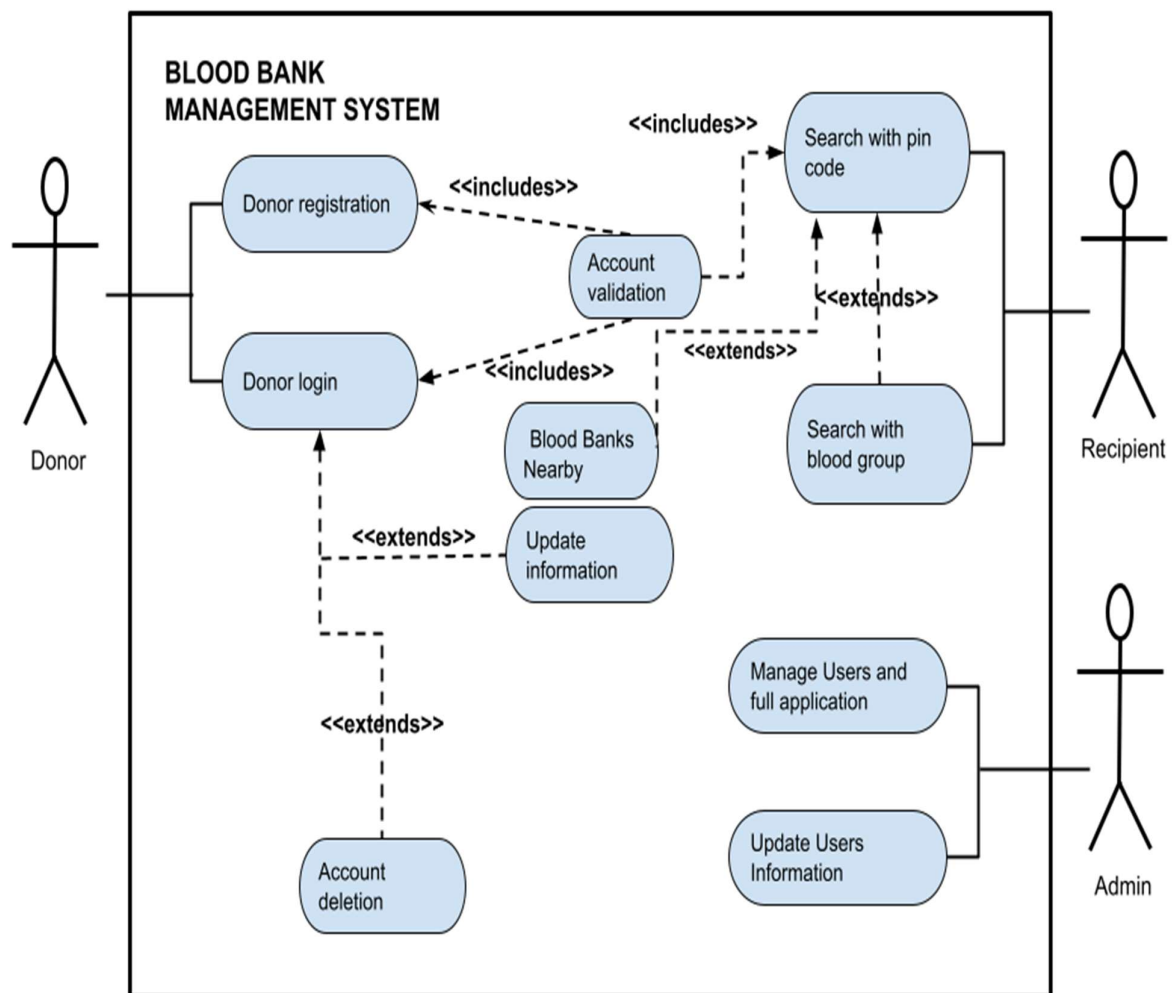
First Level Data Flow Diagram



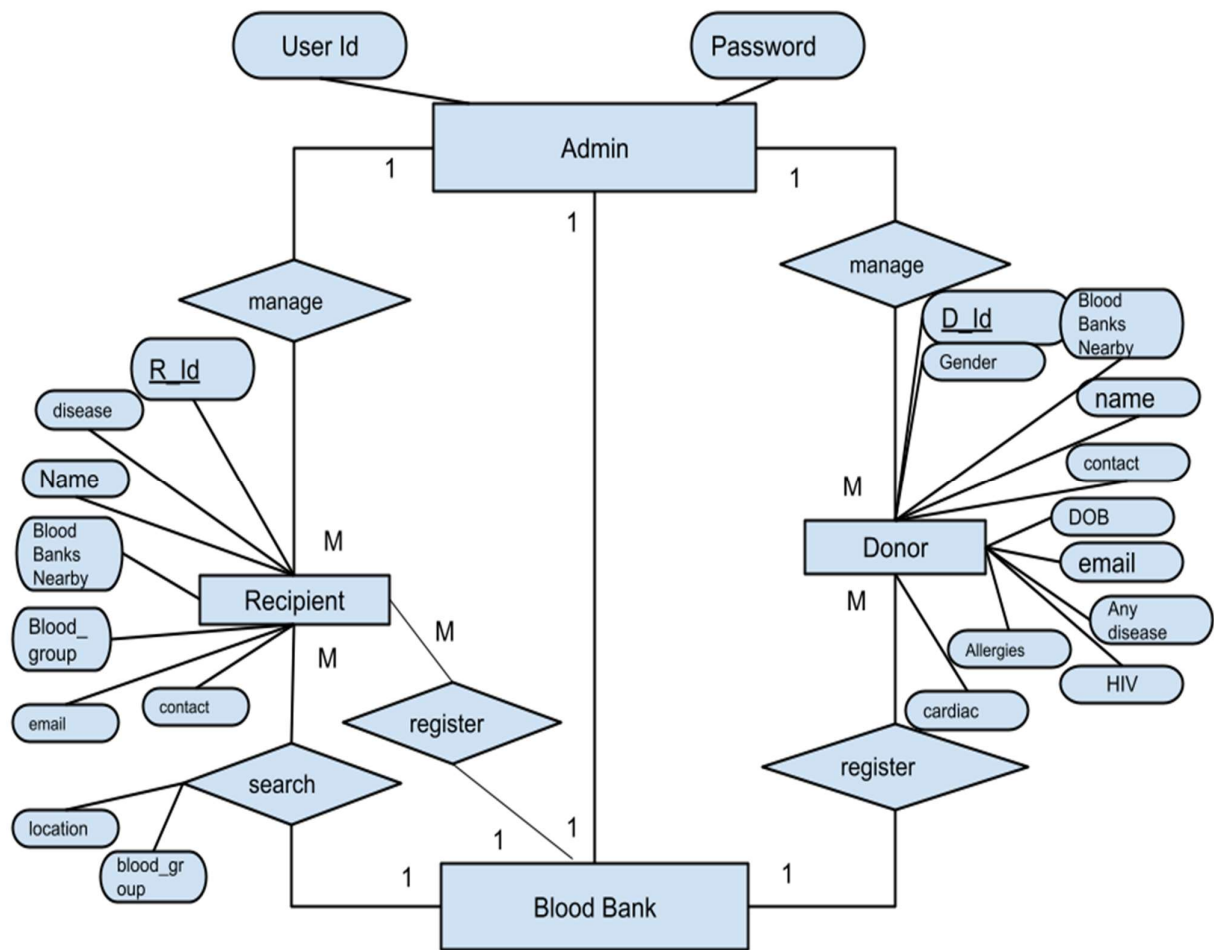
Second Level Data Flow Diagram 0.2



USE CASE DIAGRAM



E-R DIAGRAM



SEQUENCE OF APPLICATION

This application is user friendly with attractive features:

1. Starts with a homepage which gives an overview of all the features inside this application
2. After that we can go to registration page for both Donors and Recipients.
3. Then if already registered, you have an option to Login with username and password. and search the details of all Donors, filtered by specific blood group and Pin Code.
4. If you're searching for a blood bank near your current location then select the "Blood Banks Nearby" which will display all the blood banks around your area.
5. There's one Admin page which shows all the data of both Recipients and Donors which can only be accessed by the Administrators.
6. In case of any discrepancies and suggestions you can go to "Contact us" option.
7. "About us" page gives an insight about the company and its operations.

IMPLEMENTATION

Implementation is based on the Waterfall Model

1. Initially in first phase, all the requirements, were gathered and with proper selection of Software Tools i.e. SRS document was prepared.
2. Then in the second phase, system design of the project was prepared i.e. the Use Case and the DFD diagram of the project.
3. In the third phase, the app was implemented in small units with each unit being developed and tested for its functionality simultaneously i.e. Unit Testing.
4. In the fourth phase, all the units developed were integrated and tested for any faults or failures

After the functionals and non- functional requirements were complete, the project was ready to be deployed.

TECHNOLOGY STACK

Database Server: Sqlite

Client: Any web browser

Development Tools: Microsoft Visual Studio, Google Cloud

Programming Language: Python

Framework: Django

FUTURE WORKS

In near future we can add more feature applications:

1. We can prepare an Android/IOS version of this application.
2. We can add a feature to
 - ✓ check the availability of stock in a Blood Bank.
 - ✓ organise Blood Donation Camps
 - ✓ generate an E-pass for those who want to donate blood in this pandemic
3. Using cloud services every Donor/ Recipient receives a push notification about the status.

LIMITATIONS

1. This web application is not yet hosted on webserver
2. May not be user friendly for a layman as entry work is cumbersome and tedious and not simple and crisp.

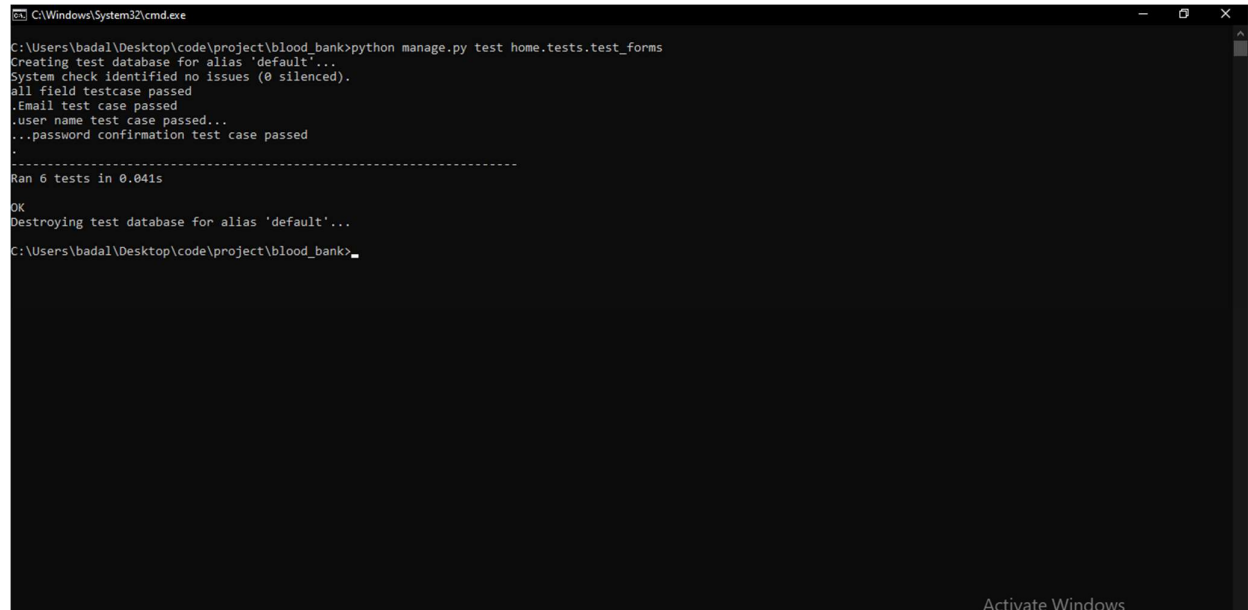
TESTING METHODS USED

All the units were tested manually at every instance initially.

Further unit testing was done :

1. Unit Testing of forms:

All the test values of forms were taken, depending upon the parameters assert condition was set i.e assertTrue() or assertFalse(), parameters were taken as input to return a Boolean value. This was how forms were tested.

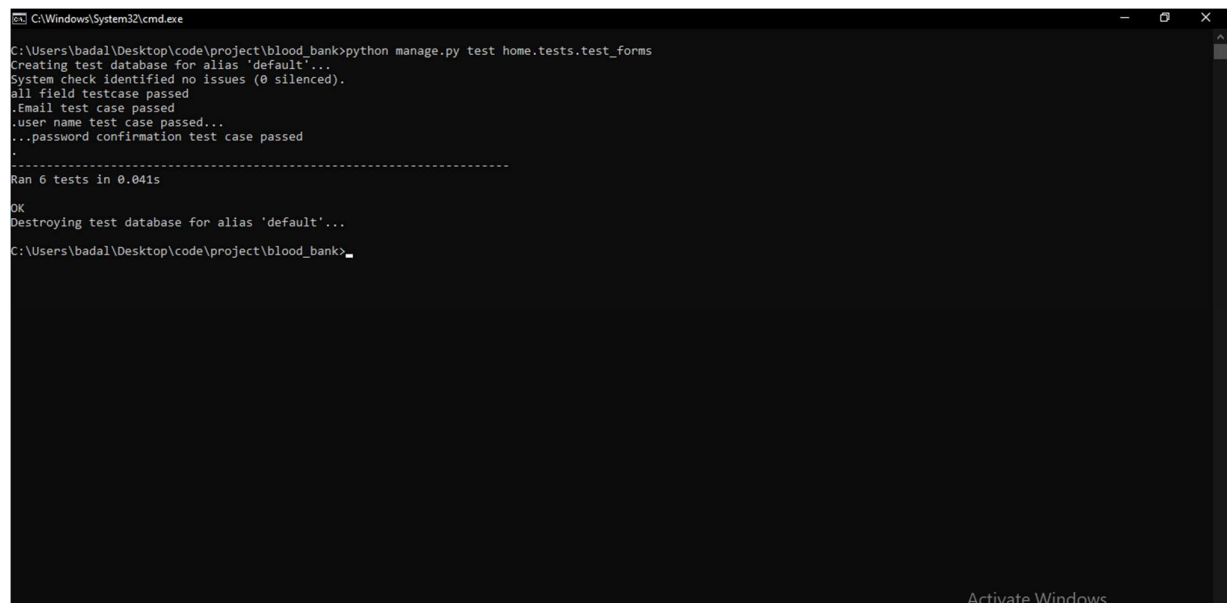


```
C:\Windows\System32\cmd.exe
C:\Users\badal\Desktop\code\project\blood_bank>python manage.py test home.tests.test_forms
Creating test database for alias 'default'...
System check identified no issues (0 silenced).
.
all field testcase passed
.Email test case passed
..user name test case passed...
...password confirmation test case passed
.
-----
Ran 6 tests in 0.041s

OK
Destroying test database for alias 'default'...
C:\Users\badal\Desktop\code\project\blood_bank>
```

2. Unit Testing of models:

All the test values of models were taken ,depending upon the parameters assert condition was set i.e. assertEquals() or assertNotEqual(), parameters were taken as Input as if both values were equal assertEquals() return true.

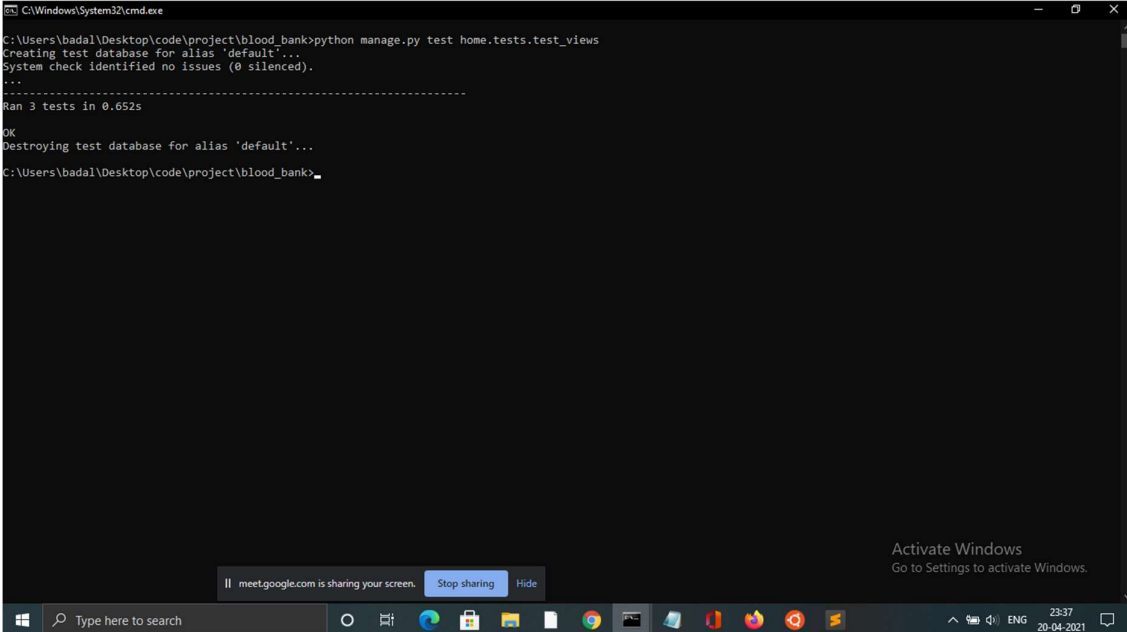


```
C:\Windows\System32\cmd.exe
C:\Users\badal\Desktop\code\project\blood_bank>python manage.py test home.tests.test_forms
Creating test database for alias 'default'...
System check identified no issues (0 silenced).
.
all field testcase passed
.Email test case passed
..user name test case passed...
...password confirmation test case passed
.
-----
Ran 6 tests in 0.041s

OK
Destroying test database for alias 'default'...
C:\Users\badal\Desktop\code\project\blood_bank>
```

3. Unit Testing of view:

Here we checked the behavior as it is experienced by a user through a web browser. Again we used `assertEqual()` to check if both parameters are equal then a response code 200 will be generated i.e. response is successful.



```
C:\Windows\System32\cmd.exe
C:\Users\badal\Desktop\code\project\blood_bank>python manage.py test home.tests.test_views
Creating test database for alias 'default'...
System check identified no issues (0 silenced).
...
-----
Ran 3 tests in 0.652s

OK
Destroying test database for alias 'default'...
C:\Users\badal\Desktop\code\project\blood_bank>
```

The screenshot shows a Windows command prompt window with a black background and white text. The title bar at the top reads "C:\Windows\System32\cmd.exe". The command prompt shows the execution of the command `python manage.py test home.tests.test_views`. The output indicates that a test database was created for the alias 'default', a system check identified no issues, and 3 tests were run successfully in 0.652 seconds. The prompt then shows the user's current directory as `C:\Users\badal\Desktop\code\project\blood_bank>`. At the bottom of the window, there is a taskbar with various application icons and a system tray showing the time as 23:37 on 20-04-2021. A watermark for "meet.google.com" is visible in the center of the screen.

CONTRIBUTIONS

Kumar Badal (1801094):

- Designed the frontend of the Home Page, Login Page and Contact Us page using HTML, CSS and JavaScript
- Implemented the back-end using Django of
 - ✓ Login Page of the User and the Admin
 - ✓ Search Page
- Designed/Modified and added additional features in the frontend of “Blood Banks Nearby” webpage using HTML and CSS.
- Testing and Debugging of the application
- Prepared all the documents of the Project

Gurnoor Singh (1801065):

- Designed the frontend of the Registration page, Search page and About us using HTML CSS and bootstrap
- Implemented the back-end using Django of
 - ✓ Registration Page of the User
- Built a webpage that displays a Google map centered on the user's location, finds nearby blood banks, and displays the blood banks as clickable markers to show more details about each blood bank using Java Script , Google Cloud and Google Maps Platform APIs.
- Testing and Debugging of the application
- Prepared all the documents of the Project