

THE BLOOD ALLIANCE
A MINI PROJECT REPORT

Submitted by

THAMMIREDIGARI HEMANTH KRISHNA

111720102160

VETRIVEL M

111720102172

SHREE RANGANATHAN S

111720102310

*in partial fulfillment for the award of the
degree of*

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING

R.M.K. ENGINEERING COLLEGE

(An Autonomous Institution)

R.S.M. Nagar, Kavaraipettai-601 206



November 2022

R.M.K. ENGINEERING COLLEGE

(An Autonomous Institution)

R.S.M. Nagar, Kavaraipettai-601 206

BONAFIDE CERTIFICATE

Certified that this project report “**The Blood Alliance**” is the bonafide work of **Thammiredigari Hemanth Krishna 111720102160, Vetrivel M 111720102172, Shree Ranganathan S 111720102310** who carried out the **20CS513 Mini Project and Design Thinking** work under my supervision.

SIGNATURE

Dr. T. Sethukarasi, M.E., M.S. Ph.D.,
Professor and Head
Department of Computer Science and
Engineering
R.M.K. Engineering College
R.S.M. Nagar, Kavaraipettai,
Tiruvallur District– 601206.

SIGNATURE

Dr. J. Sathiamoorthy
Supervisor
Associate Professor (GR-II)
Department of Computer Science and
Engineering
R.M.K. Engineering College
R.S.M. Nagar, Kavaraipettai,
Tiruvallur District–601206.

Submitted for the Project Viva –Voce held on at **R.M.K. Engineering College, Kavaraipettai, Tiruvallur District– 601206.**

INTERNAL EXAMINER

ACKNOWLEDGEMENT

We earnestly portray our sincere gratitude and regard to our beloved **Chairman Shri. R. S. Munirathinam, our Vice Chairman, Shri. R. M. Kishore** and **our Director, Shri. R. Jyothi Naidu**, for the interest and affection shown towards us throughout the course.

We convey our sincere thanks to our **Principal, Dr. K. A. Mohamed Junaid**, for being the source of inspiration in this college.

We reveal our sincere thanks to our **Professor and Head of the Department, Computer Science and Engineering, Dr. T. Sethukarasi**, for her commendable support and encouragement for the completion of our project.

We would like to express our sincere gratitude for our Mini Project Coordinator **Dr. J. Sathiamoorthy, Associate Professor** and project guide **Dr. J. Sathiamoorthy, Associate Professor (GR-II)** for their valuable suggestions towards the successful completion for this project in a global manner.

We take this opportunity to extend our thanks to all faculty members of Department of Computer Science and Engineering, parents and friends for all that they meant to us during the crucial times of the completion of our project.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE NO
	ABSTRACT	5
	LIST OF FIGURES	6
1	INTRODUCTION	
	1.1 Problem Statement	7
	1.2 Project Scope and Objectives	7
2	OVERALL DESCRIPTION	
	2.1 Project Specification	8
	2.2 Use Case Diagram	9
	2.3 Logical Diagram	10
	2.4 System Architecture	11
3	EXTERNAL INTERFACE REQUIREMENTS	
	3.1 User Interfaces	12
	3.2 Hardware Interfaces	12
	3.3 Software Interfaces	12
4	TESTING	
	4.1 Test Plan	13
	4.2 Test Procedure	14
5	FUTURE ENHANCEMENTS	15
6	CONCLUSION	16
	REFERENCES	17
	SAMPLE CODING	18
	SCREENSHOTS	29

ABSTRACT

The Blood Alliance is designed to simplify and automate the process of searching for blood in case of emergency, maintain the records of blood requests, blood donors, recipients, blood donation programs, and blood stocks in the bank. The blood bank centers can utilize the system to digitally transform their operations. The blood bank can manage records of blood stocks, blood requests, blood donations, blood donors, and recipients all in one place. Blood recipients or patients and blood donors can also utilize the system to process blood requests and blood donations, respectively.

LIST OF FIGURES

S.NO	FIGURE NAME	PAGE NO
1.	Case Diagram	9
2.	Logical Diagram	10
3.	Architecture Diagram	11

CHAPTER 1

INTRODUCTION

1.1 Problem Statement

Due to lack of Guidance students can't able to decide which career path is suitable for them and as result they can't able to succeed in their Goals. Thus Our Project Model helps the user (students) to identify the strong points and weak spots and helps according to it to achieve their goals.

1.2 Project Scope and Objective:

1.2.1 Scope of the Project:

This study focus on the development of the Blood Alliance. The researchers mainly focus on designing and developing a system that will electronically streamline searching for blood and blood requests, recording blood information and blood units availability, recording and managing blood donors and recipients. This project is intended for blood banks management and blood requests and donation transactions. Blood banks, staff, blood donors, and patients will participate as respondents to the study.

1.2.2 Objective of the Project:

The researchers generally aim to implement a Blood Bank Management System to automate the management and daily operations of blood banks.

CHAPTER 2

OVERALL DESCRIPTION

2.1. PROJECT SPECIFICATION

2.1.1 EXISTING SYSTEM

There are two types of process in the existing system:

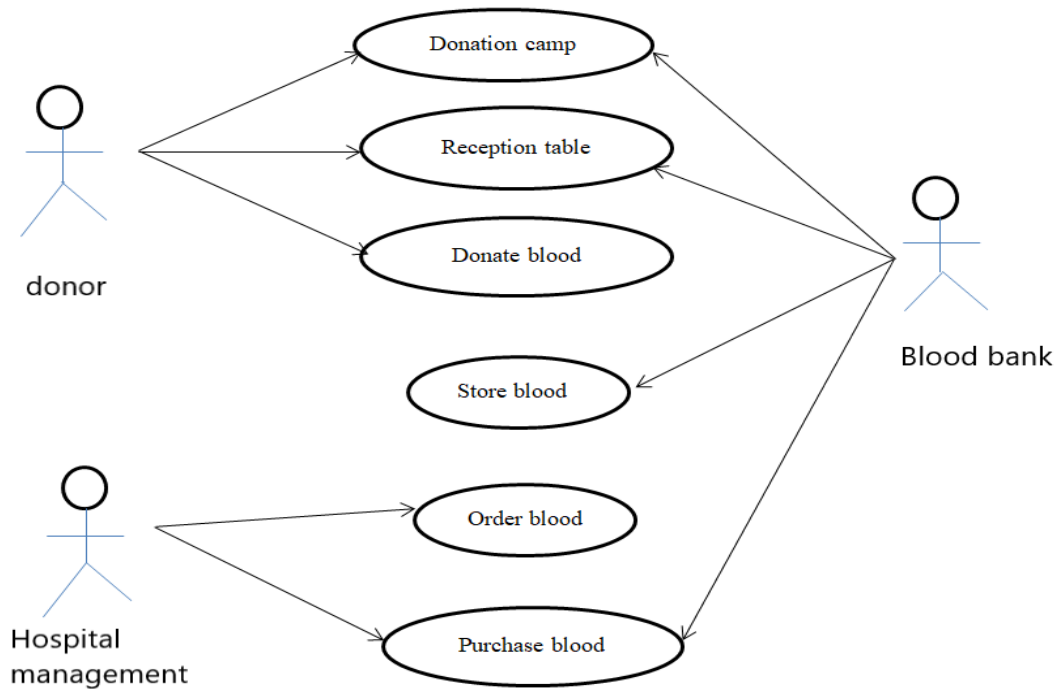
- One is the blood donation by donors.
- Another one is the blood request process by Hospitals.

2.1.1 PROPOSED SYSTEM

The proposed Blood Bank Management System helps the Blood Bank Admin to easily monitor the blood requests and users database. The proposed system takes a systematic approach of how to bridge the gap between Recipients, Donors, and Blood Banks. It improves the existing system by providing a common ground to ease the process of blood donation and reception.

The database mainly consists of the registered donor's information and inquiries managed by an Admin. It also comprises the records of available blood group samples.

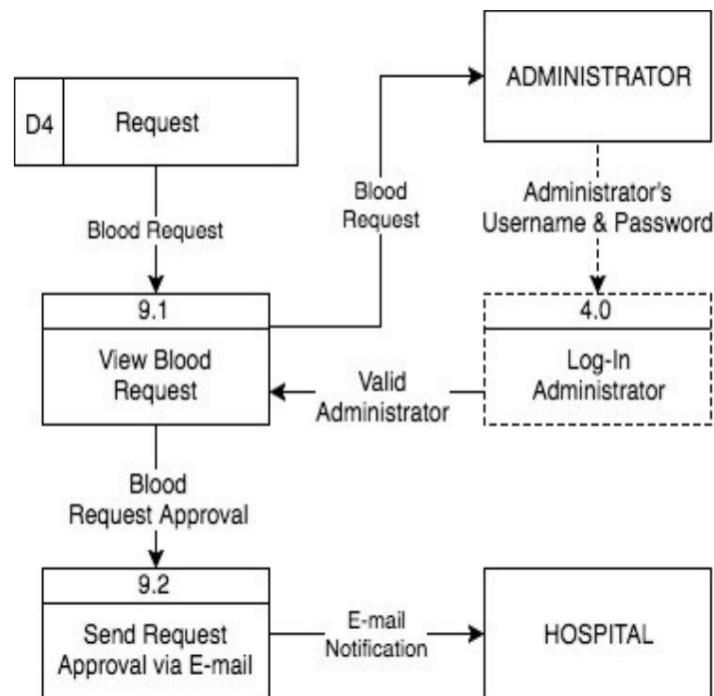
2.2. USE CASE DIAGRAM



Thus we can see in the use case diagram the student the cane able to access the quiz, report and courses whereas the technical team can also able to access the details of the Blood Alliance.

2.3. LOGICAL DESIGN

Data Flow Diagram for Blood Alliance.

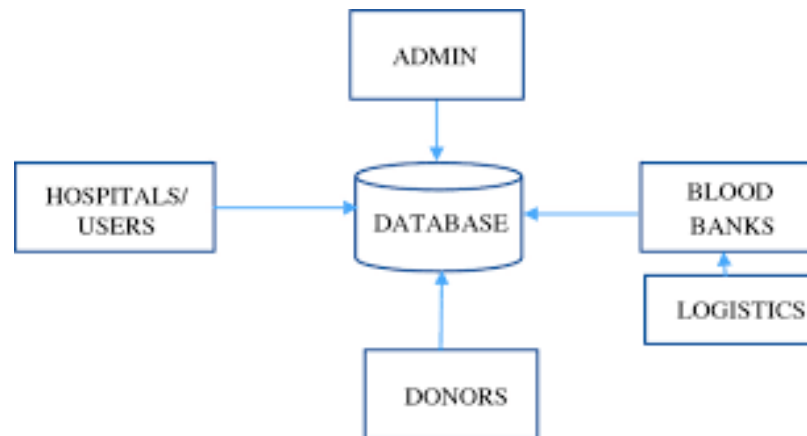


Here the request for donating the blood or for approval of the required blood for the receptors are sent from one to another respective panels.

Thus the blood request is sent to the admin who verify it by checking the details about the receptors where he tends to check the availability of the respective bloods. The respond is sent back to the receptors whether the blood is available or not with their blood bank. This process is held by the adminstator who handles all the request and respond process.

2.4 SYSTEM ARCHITECTURE

An architectural diagram is a visual representation that maps out the physical implementation for components of a software system. It shows the general structure of the software system and the associations, limitations, and boundaries between each element.



- In our project we have a very simple architecture. As an initial step Hospitals/users are able to open the software.
- After opening the app they can able to access the features available in the system
- In the technical team side they are used to manage the software as well as the records of blood stocks, blood requests, blood donations, blood donors, and recipients all in one place.
- The Database is act one of the major role which help to connect the software and the users as well as Technical teams like donors, blood bank.

CHAPTER 3

EXTERNAL INTERFACE REQUIREMENTS

3.1 USER INTERFACE

Any popular OS that will allow the use of a browser to view and access web pages. In order to make the user comfortable, we have displayed all the main details in our menu bar itself.

3.2. HARDWARE INTERFACE

Any kind of internet connection like WIFI, modem data etcetera, to allow the browser interfaces to connect to the website. The website can be accessed through any devices like computer, laptop, tablet, etc.

Hardware Requirements:

- CPU: Intel i5 processor with 64-bit operating system
- RAM: 8 GB
- STORAGE: 1 TB Storage
- INTERNET: wireless adapter (Wi-Fi)

3.3. SOFTWARE INTERFACE

Some of the software interfaces which you can use to access our website are

- Opera
- Google chrome
- Visual Studio Code
- Mysql

CHAPTER 4

TESTING

4.1 TEST PLAN

Scope - The scope of the testing is to make the user to go through all the web pages easily.

4.2 TEST PROCEDURE

The test plan is shown below

Name of the test	Things to be tested
API testing	User's comfort with website should be tested.
Unit testing	Every page of the website should be tested.
Integration testing	Connection between the webpages are tested.
System testing	The functional flow of the webpages are tested.

TEST DELIVERABLES

Test Deliverables – An enjoyable and wrathful destination will be provided as the outcome.

To understand the Test Deliverables we need to understand what all the test cases are there for the software

TEST CSE ID	TEST NAME	EXPECTED RESULTS	ACTUAL RESULTS	STATUS
1.	While opening ,the site should open	Site should open	As expected	Pass
2.	Options should be opened while clicking	Options should open	As expected	Pass
3.	On pressing donor/admin/receptor button	Buttons should navigate	As expected	Pass
4.	On pressing request button	Admin should reply for the request	As expected	Pass

CHAPTER 5

FUTURE ENHANCEMENTS

The present Blood Bank project may be further developed for more complex transactions and to meet the requirements of modern day dynamic System Operation. New options and their respective implementation may be done for this purpose. The size of the database may increase exponentially. This proposes a Blood Alliance which we believe will bring remarkable changes.

CHAPTER 6

CONCLUSION

A Blood bank is a center where supplies of blood used for blood transfusions are stored and preserved. This study was conducted to introduce a platform that will advance the system used in managing blood banks. Thus we developed a Blood Alliance. The result of the study showed that the Blood Alliance met the predefined project requirements. The system will streamline the transaction between the blood banks, blood recipients, and donors. The result of the study showed that the developed system is efficient and reliable to use.

“The Gift of blood is a gift to someone’s life.”

REFERENCES

- Vikas Kulshreshtha, Dr. Sharad Maheshwari, “Blood Bank Management Information System in India,
- A. Clemen Teena, K. Sankar, and S. Kannan, “A Study on Blood Bank Management,” in Middle-East Journal of Scientific Research
- Ravi Kumar, Shubham Singh, V Anu Ragavi, “ Blood Bank Management System,”

SAMPLE CODING

```
# Generated by Django 3.0.5 on 2021-01-17 13:08
from django.db import migrations, models
class Migration(migrations.Migration):
    initial = True
    dependencies = [
    ]
    operations = [
        migrations.CreateModel(
            name='Stock',
            fields=[
                ('id', models.AutoField(auto_created=True, primary_key=True, serialize=False,
verbose_name='ID')),
                ('bloodgroup', models.CharField(max_length=10)),
                ('unit', models.PositiveIntegerField(default=0)),
            ],
        ),
    ]
```

```
# Generated by Django 3.0.5 on 2021-01-18 16:17
```

```
from django.db import migrations, models

class Migration(migrations.Migration):

    dependencies = [
        ('blood', '0001_initial'),
    ]

    operations = [
        migrations.CreateModel(
            name='BloodRequest',
            fields=[
                ('id', models.AutoField(auto_created=True, primary_key=True, serialize=False,
verbose_name='ID')),
                ('name', models.CharField(max_length=30)),
                ('age', models.PositiveIntegerField()),
                ('mobile', models.CharField(max_length=20)),
                ('disease', models.CharField(max_length=100)),
                ('reason', models.CharField(max_length=500)),
                ('bloodgroup', models.CharField(max_length=10)),
                ('unit', models.PositiveIntegerField(default=0)),
                ('status', models.CharField(default='Pending', max_length=20)),
            ],
        ),
    ]
```

```

    ),
]
# Generated by Django 3.0.5 on 2021-02-13 05:23

from django.db import migrations, models
import django.db.models.deletion

class Migration(migrations.Migration):

    dependencies = [
        ('patient', '0001_initial'),
        ('donor', '0001_initial'),
        ('blood', '0002_bloodrequest'),
    ]

    operations = [
        migrations.RenameField(
            model_name='bloodrequest',
            old_name='age',
            new_name='patient_age',
        ),
        migrations.RenameField(
            model_name='bloodrequest',
            old_name='name',
            new_name='patient_name',
        ),
        migrations.RemoveField(
            model_name='bloodrequest',
            name='disease',
        ),
        migrations.RemoveField(
            model_name='bloodrequest',
            name='mobile',
        ),
        migrations.AddField(
            model_name='bloodrequest',
            name='request_by_donor',
            field=models.ForeignKey(null=True, on_delete=django.db.models.deletion.CASCADE,
to='donor.Donor'),
        ),
        migrations.AddField(
            model_name='bloodrequest',
            name='request_by_patient',

```

```

        field=models.ForeignKey(null=True, on_delete=django.db.models.deletion.CASCADE,
to='patient.Patient'),
    ),
]

```

Create your tests here.

```

from django.shortcuts import render,redirect,reverse
from . import forms,models
from django.db.models import Sum,Q
from django.contrib.auth.models import Group
from django.http import HttpResponseRedirect
from django.contrib.auth.decorators import login_required,user_passes_test
from django.conf import settings
from datetime import date, timedelta
from django.core.mail import send_mail
from django.contrib.auth.models import User
from donor import models as dmodels
from patient import models as pmodels
from donor import forms as dforms
from patient import forms as pforms

```

```

def home_view(request):
    x=models.Stock.objects.all()
    print(x)
    if len(x)==0:
        blood1=models.Stock()
        blood1.bloodgroup="A+"
        blood1.save()

        blood2=models.Stock()
        blood2.bloodgroup="A-"
        blood2.save()

        blood3=models.Stock()
        blood3.bloodgroup="B+"
        blood3.save()

        blood4=models.Stock()
        blood4.bloodgroup="B-"
        blood4.save()

        blood5=models.Stock()
        blood5.bloodgroup="AB+"
        blood5.save()

```

```

blood6=models.Stock()
blood6.bloodgroup="AB-"
blood6.save()

```

```

blood7=models.Stock()
blood7.bloodgroup="O+"
blood7.save()

```

```

blood8=models.Stock()
blood8.bloodgroup="O-"
blood8.save()

```

```

if request.user.is_authenticated:
    return HttpResponseRedirect('afterlogin')
return render(request,'blood/index.html')

```

```

=====
=====

```

```

#RESET

```

```

=====
===== */

```

```

/**

```

```

* A very simple reset that sits on top of Normalize.css.
*/

```

```

body,
h1, h2, h3, h4, h5, h6,
blockquote, p, pre,
dl, dd, ol, ul,
figure,
hr,
fieldset, legend {
    margin: 0;
    padding: 0;
}

```

```

/**

```

```

* Remove trailing margins from nested lists.
*/

```

```

li > ol,
li > ul {
    margin-bottom: 0;
}

```

```

/**

```

```

* Remove default table spacing.
*/
table {
  border-collapse: collapse;
  border-spacing: 0;
}

/**
 * 1. Reset Chrome and Firefox behaviour which sets a `min-width: min-content;`
 *    on fieldsets.
 */
fieldset {
  min-width: 0;
  /* [1] */
  border: 0;
}

button {
  outline: none;
  background: none;
  border: none;
}

/*
=====
=====

#PAGE WRAPPER

=====
===== */
.page-wrapper {
  min-height: 100vh;
}

body {
  font-family: "Open Sans", "Arial", "Helvetica Neue", sans-serif;
  font-weight: 400;
  font-size: 14px;
}

h1, h2, h3, h4, h5, h6 {
  font-weight: 400;
}

h1 {
  font-size: 36px;

```

```

}

h2 {
  font-size: 30px;
}

h3 {
  font-size: 24px;
}

h4 {
  font-size: 18px;
}

h5 {
  font-size: 15px;
}

h6 {
  font-size: 13px;
}

/*
=====
=====
#BACKGROUND
=====
===== */
.bg-blue {
  background: #2c6ed5;
}

.bg-red {
  background: #fa4251;
}

.bg-gra-01 {
  background: -webkit-gradient(linear, left bottom, left top, from(#fbc2eb), to(#a18cd1));
  background: -webkit-linear-gradient(bottom, #fbc2eb 0%, #a18cd1 100%);
  background: -moz-linear-gradient(bottom, #fbc2eb 0%, #a18cd1 100%);
  background: -o-linear-gradient(bottom, #fbc2eb 0%, #a18cd1 100%);
  background: linear-gradient(to top, #fbc2eb 0%, #a18cd1 100%);
}

.bg-gra-02 {

```

```

background: -webkit-gradient(linear, left bottom, right top, from(#fc2c77), to(#6c4079));
background: -webkit-linear-gradient(bottom left, #fc2c77 0%, #6c4079 100%);
background: -moz-linear-gradient(bottom left, #fc2c77 0%, #6c4079 100%);
background: -o-linear-gradient(bottom left, #fc2c77 0%, #6c4079 100%);
background: linear-gradient(to top right, #fc2c77 0%, #6c4079 100%);
}

```

```

.bg-gra-03 {
background: -webkit-gradient(linear, left bottom, right top, from(#08aeea), to(#b721ff));
background: -webkit-linear-gradient(bottom left, #08aeea 0%, #b721ff 100%);
background: -moz-linear-gradient(bottom left, #08aeea 0%, #b721ff 100%);
background: -o-linear-gradient(bottom left, #08aeea 0%, #b721ff 100%);
background: linear-gradient(to top right, #08aeea 0%, #b721ff 100%);
}

```

```
{% endblock content %}
```

```
<!--
```

```
developed By : sumit kumar
```

```
facebook : fb.com/sumit.luv
```

```
youtube : youtube.com/lazycoders
```

```
-->
```

```
{% extends 'blood/adminbase.html' %}
```

```
{% block content %}
```

```
<br><br>
```

```
<div class="container">
```

```
<div class="row">
```

```
<div class="col-sm-3">
```

```
<div class="card bg-light">
```

```
<div class="card-body">
```

```
<div class="blood">
```

```
<h2>A+ <i class="fas fa-tint"></i></h2>
```

```
</div><br><br>
```

```
<div>
```

```
{{ A1.unit }}
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="col-sm-3">
```

```
<div class="card bg-light">
```

```
<div class="card-body">
```

```
<div class="blood">
```

```
<h2>B+ <i class="fas fa-tint"></i></h2>
```

```
</div><br><br>
```



```

        <div>
            {{ B1.unit }}
        </div>
    </div>
</div>
</div>
<div class="col-sm-3">
    <div class="card bg-light">
        <div class="card-body">
            <div class="blood">
                <h2>O+ <i class="fas fa-tint"></i></h2>
            </div><br><br>
            <div>
                {{ O1.unit }}
            </div>
        </div>
    </div>
</div>
<div class="col-sm-3">
    <div class="card bg-light">
        <div class="card-body">
            <div class="blood">
                <h2>AB+ <i class="fas fa-tint"></i></h2>
            </div><br><br>
            <div>
                {{ AB1.unit }}
            </div>
        </div>
    </div>
</div>
</div>
</div>
<div class="row">
    <div class="col-sm-3">
        <div class="card bg-light">
            <div class="card-body">
                <div class="blood">
                    <h2>A- <i class="fas fa-tint"></i></h2>
                </div><br><br>
                <div>
                    {{ A2.unit }}
                </div>
            </div>
        </div>
    </div>
</div>
</div>

```

```

<div class="col-sm-3">
  <div class="card bg-light">
    <div class="card-body">
      <div class="blood">
        <h2>B- <i class="fas fa-tint"></i></h2>
      </div><br><br>
      <div>
        { { B2.unit } }
      </div>
    </div>
  </div>
</div>
<div class="col-sm-3">
  <div class="card bg-light">
    <div class="card-body">
      <div class="blood">
        <h2>O- <i class="fas fa-tint"></i></h2>
      </div><br><br>
      <div>
        { { O2.unit } }
      </div>
    </div>
  </div>
</div>
<div class="col-sm-3">
  <div class="card bg-light">
    <div class="card-body">
      <div class="blood">
        <h2>AB- <i class="fas fa-tint"></i></h2>
      </div><br><br>
      <div>
        { { AB2.unit } }
      </div>
    </div>
  </div>
</div>
</div>
<hr>
<div class="row">
  <div class="col-sm-3">
    <div class="card bg-light">
      <div class="card-body">
        <div class="blood">
          <i class="fas fa-users"></i>
        </div><br>

```

```

        <div>
            Total Donors <br>
            {{totaldonors}}
        </div>
    </div>
</div>
</div>
</div>
<div class="col-sm-3">
    <div class="card bg-light">
        <div class="card-body">
            <div class="blood">
                <i class="fas fa-spinner"></i>
            </div><br>
            <div>
                Total Requests <br>
                {{totalrequest}}
            </div>
        </div>
    </div>
</div>
<div class="col-sm-3">
    <div class="card bg-light">
        <div class="card-body">
            <div class="blood">
                <i class="far fa-check-circle"></i>
            </div><br>
            <div>
                Approved Requests <br>
                {{totalapprovedrequest}}
            </div>
        </div>
    </div>
</div>
<div class="col-sm-3">
    <div class="card bg-light">
        <div class="card-body">
            <div class="blood">
                <i class="fas fa-tint xyz"></i>
            </div><br>
            <div>
                Total Blood Unit (in ml) <br>
                {{totalbloodunit}}
            </div>
        </div>
    </div>
</div>

```

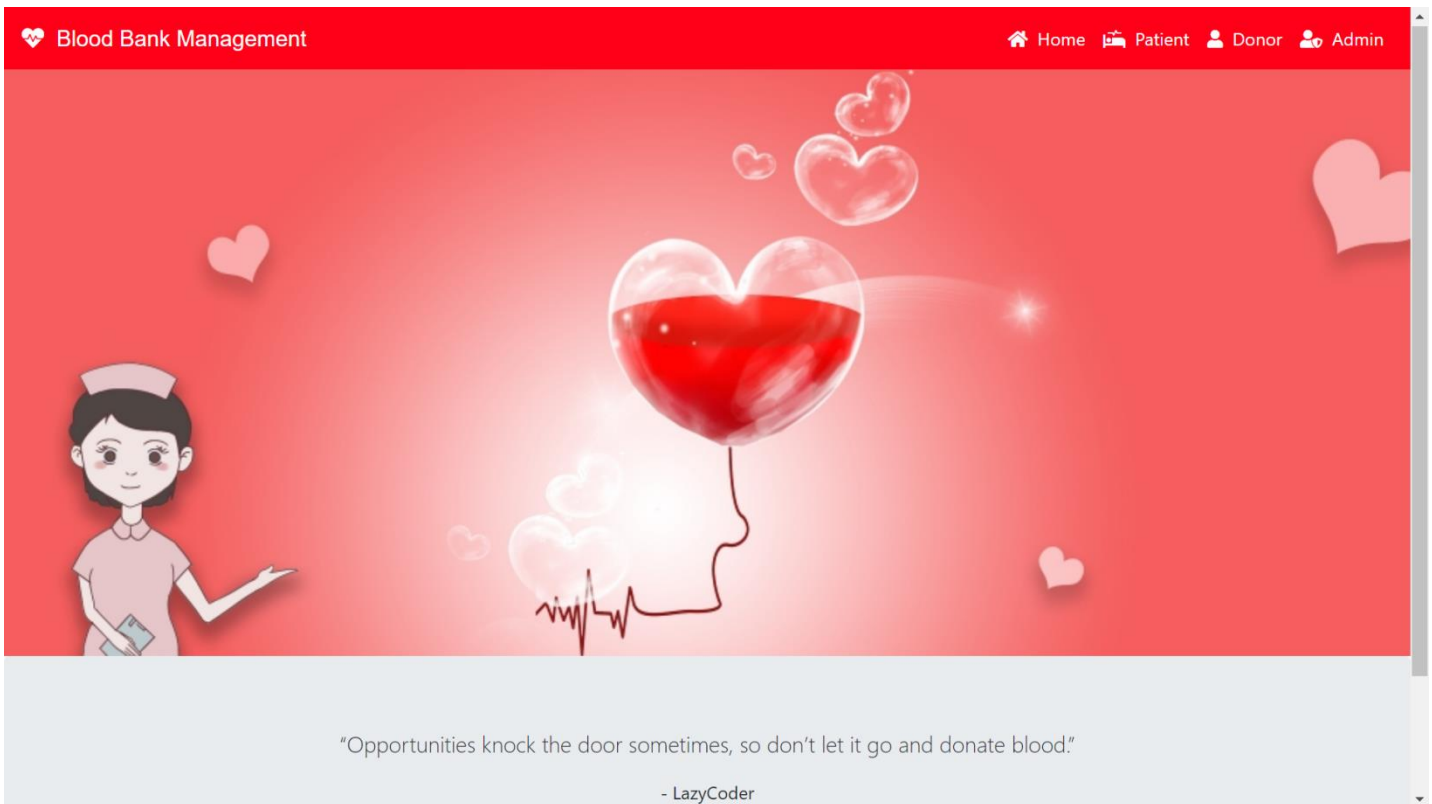
</div>

</div>

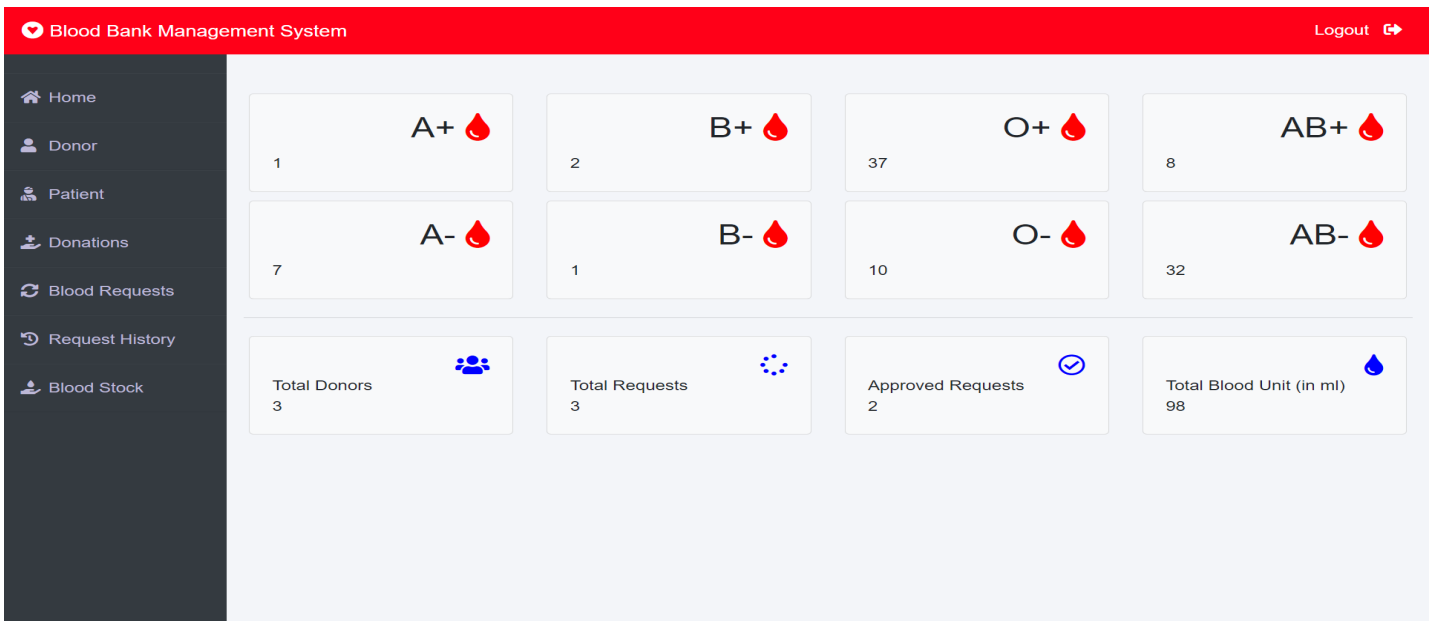
</div>

SCREENSHOTS

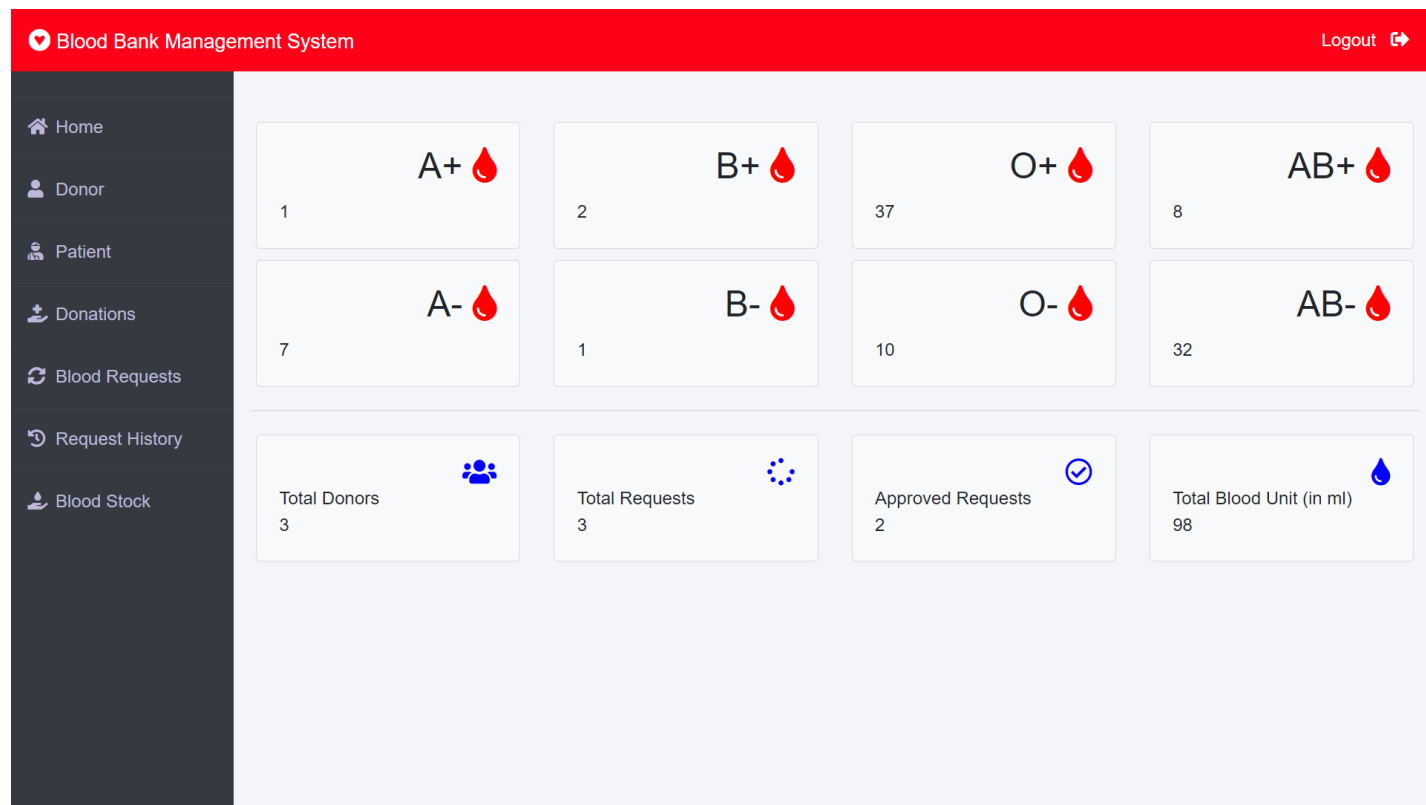
HOMEPAGE



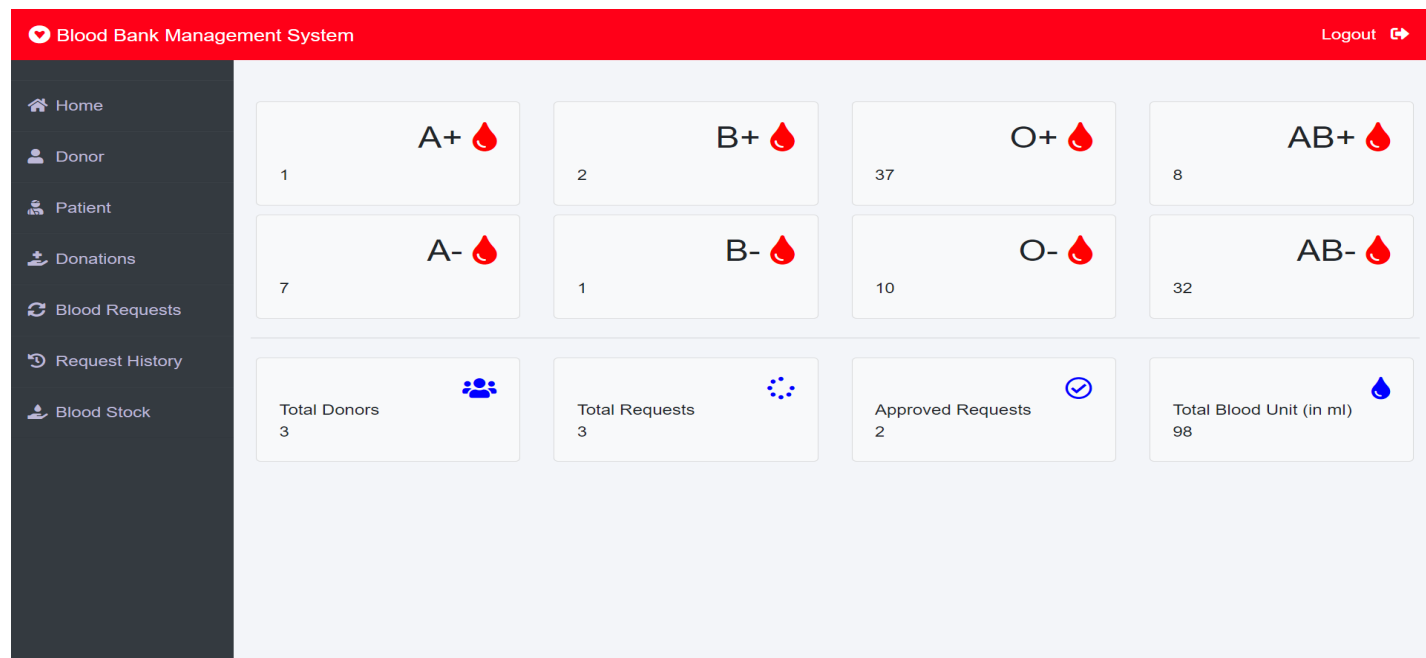
ADMIN DASHBOARD



BLOOD DONATION



BLOOD REQUEST



LOGOUT

Blood Bank Management System

Logout

Home

Donor

Patient

Donations

Blood Requests

Request History

Blood Stock

1

A+

2

B+

37

O+

8

AB+

7

A-

1

B-

10

O-

32

AB-

Total Donors

3

Total Requests

3

Approved Requests

2

Total Blood Unit (in ml)

98