# A Project Report on
# BLOOD BANK MANAGEMENT SYSTEM


**Submitted by**

A Udayasree R170096
A Kavya R170137
K Cherishma R170158
M Bhuvana Rekha R171174

**Submitted to**
Under the supervision of
**Mr. Satyanandaram Nandigam**
Assistant Professor in Computer Science and Engineering Department
RGUKT, RK Valley
as a part of
**Mini Project in E3**

# Acknowledgement

We would like to express my sincere gratitude to **Mr. Satyanandaram Nandigam** our project internal guide for valuable suggestions and keen interest throughout the progress of our course of research.

We are grateful to **P.Harinath HOD CSE** , for providing excellent computing facilities and a congenial atmosphere for progressing with our project.

At the outset, we would like to thank **Rajiv Gandhi University of Knowledge and Technologies, R.K Valley** for providing all the necessary resources for the successful completion of our course work. At last, but not the least, we thank our colleagues, our classmates and other students for their physical and moral support.

**With sincere regards,**
A Udaya Sree,
A Kavya,
K Cherishma,
M Bhuvana Rekha.


# Certificate

This is to certify that the report entitled **"Blood Bank Management System"** submitted by A Udaya Sree,A Kavya,K Cherishma,M Bhuvana Rekha in partial fulfillment of therequirement for the award of Bachelor of Technology in Computer Science and Engineering is a bona fide work carried out by us under his supervision and guidance.

The report hasn't been submitted previously in part or in full to this or any other university or institution for the award of any degree.

**Mr. Satyanandaram Nandigam**,
Project internal guide,
Computer Science and Engineering,
RGUKT,R.K Valley.

**Mr.P Harinath,**
Head of thedepartment,
Computer Science and Engineering,
RGUKT, R.K Valley.Project.

# Declaration

We **A Udaya Sree**, **A Kavya** , **K Cherishma** , **M Bhuvana Rekha** here by declare that this report entitled **“Blood Bank Management System”** submitted by us under the guidance and supervision of Mr.P.Harinath, is a bonafide work. We also declare that it has not been submitted previously in part or in full to this university or other university or institution for the award of any degree or diploma.

Date: __-09-2022
Place: RK Valley

A Udaya Sree (R170096)
A Kavya (R170137)
K Cherishma (R170158)
M Bhuvana Rekha (R171174)

# Abstract

Blood Bank Management System (BBMS) is a browser based system that is designed to store, process, retrieve and analyze information concerned with the admin within a blood bank. This project aims at maintaining all the information pertaining to blood donors, different blood groups available in each blood bank and help them manage in a better way. Aim is to provide transparency in this field, make the process of obtaining blood from a blood bank hassle free and corruption free and make the system of blood bank management effective. The donors who are interested in donating blood has to register in the database. The requirement of the blood has to be requested and we supply the Blood. Then blood units status is updated whether they are available or not.

# 1. Introduction

## 1.1 Problem Statement

On the day following a blooddonation, the Blood Bank Testing Unit tests all blood for blood type and potential viral agents. They send the results of these tests to the Blood Inventory (another unit of the Centre). For each tested blood unit, if the tests indicate that the blood may be contaminated with a viral agent, the blood unit is destroyed. This is indicated on the test form.

The Blood Bank distributes blood to various patients requesting blood. Requests usually come in for specific blood types. The Blood Bank prepares refrigerated containers of these units and distributes them to the patient when they place the order. The BloodBank receives request specific units of blood to supply to the patient from the Blood Bank. When the order is filled, the Blood Bank Manager accepts the order and sends blood to the Patients.

## 1.2 Purpose

The Blood Bank Management system is a great project. This project is designed for successful completion of a project on blood bank management system.The basic building aim is to provide blood donation service to the city recently. Blood Bank Management system is a web-based application that is designed to store, process, retrieve and analyse information concerned with the administrative and inventory management within a Blood Bank.

This project aims at maintaining all the information pertaining to blood donors, different blood groups available in every Blood Bank and help them manage ina better way.

Project aim is to provide transparency in this field, make the process of obtaining blood from a Blood Bank hassle-free and corruption-free and make the system of Blood Bank Management effective

### 1.3 Scope

The specification builds on the experience of IT technology in blood transfusion that is currently
available and informs both Connecting for Health (CfH) and commercial companies producing both hardware and software.

The main objective of this specification is to support theautomated tracking of blood products from the initial collection of the blood unit to the final ordering and purchase of the units by hospitals.

# 2. Overall Description

### 2.1 Product Perspective

This system includes online components. The collection of blood will be manual through Blood Bank. The donor can either register on the Blood Bank website on his own or can visit the Blood Bank An online database is maintained with all the information about the donors. Once the blood is collected it stored in a safe place. An online Blood Inventory Database is maintained as well for the Blood Units collected.

The blood samples are tested to determine whether they are fit to be used or not..Hospitals place orders from this Blood Bank. A record of the order and payment is maintained by both parties. Once the order is placed the Blood Bank Manager send the receipt. Once the hospital makes the payment the order is delivered by the Blood Bank staff.

## 2.2 Product Function

According to this product, a Donors can create an account back at home or register themselves at the spot of blood camp before donating blood. The Hospital Manger has to register the Hospital which act as an acceptor here. The details of the blood inventory i.e., the availability of a particular type of blood is regularly updated and maintained by the ADMIN. It is
a confidential data so the access is only with the administrators. The registered hospital can place an online order. The order is processed by the Inventory Manager who can check the database of the blood units. If the required blood type and the amount is available, it notifies the corresponding hospitals. When the Hospital Manager confirms the Order, the details are
being sent to it.

## 2.3 User Characteristics

There are mainly three users interacting with each other in this system: Donor, Patient, Admin.

The Donors register through the website or at the spot of blood donation. The Admin assists them
and maintain the details of the Donors. The Admin updates the donor database.
The Admin who collects the blood units from the donors with proper procedure. After the
collection and packaging of the blood. The Doctor sends the samples to the Lab Technician for Blood testing and the rest of the blood units to the blood inventory. The Lab Technician carries out test on eachblood sample. He sends the blood report to the Inventory Manager.

The Admin discards the unfit blood and store the healthy blood in the respective places.
Then he/she updates the required changes in the blood inventory database. He/she keeps a track of
the date of the blood was newly stored in the inventory and its expiry date.
The Patient places an order for the blood units as per requirement. The Admin go throughs the order.
Admin checks and maintains the details.

### 2.4 Constraints

The Donor and the acceptor are constrained to create an account first to avail the services.The internet connection is also a constraint for this web application.The web application is also constrained by the database capacity so it works well with a smaller number of donors .
The access to manage the databases are different for different people. The Admin is given the access to maintain the database of the registered donors and Patients.

### 2.5 Assumptions and Dependencies

It is assumed that the users have enough resources to run the web application i.e a mobile phone or a computer that supports the required functions.

The web application uses Django for creating and managing the database.

The front end is designed with the help of HTML and CSS.

# 3. Requirements

### 3.1 Functional Requirements

*1. Access Website:*
User should be able to access web-application through either an application browser or similar service on the mobile phone or computer. There should not be any limitation to access web-application.

*2. User Registration:*

Given that user has accessed web-application, then the user should be able to register through the web-application. The donor user must provide first name, gender, blood group, location, contact , username and password.

*3. User log-in:*

Given that the user has registered, then the user should be able to login to the web-application. The login information will be stored on the database for future use.

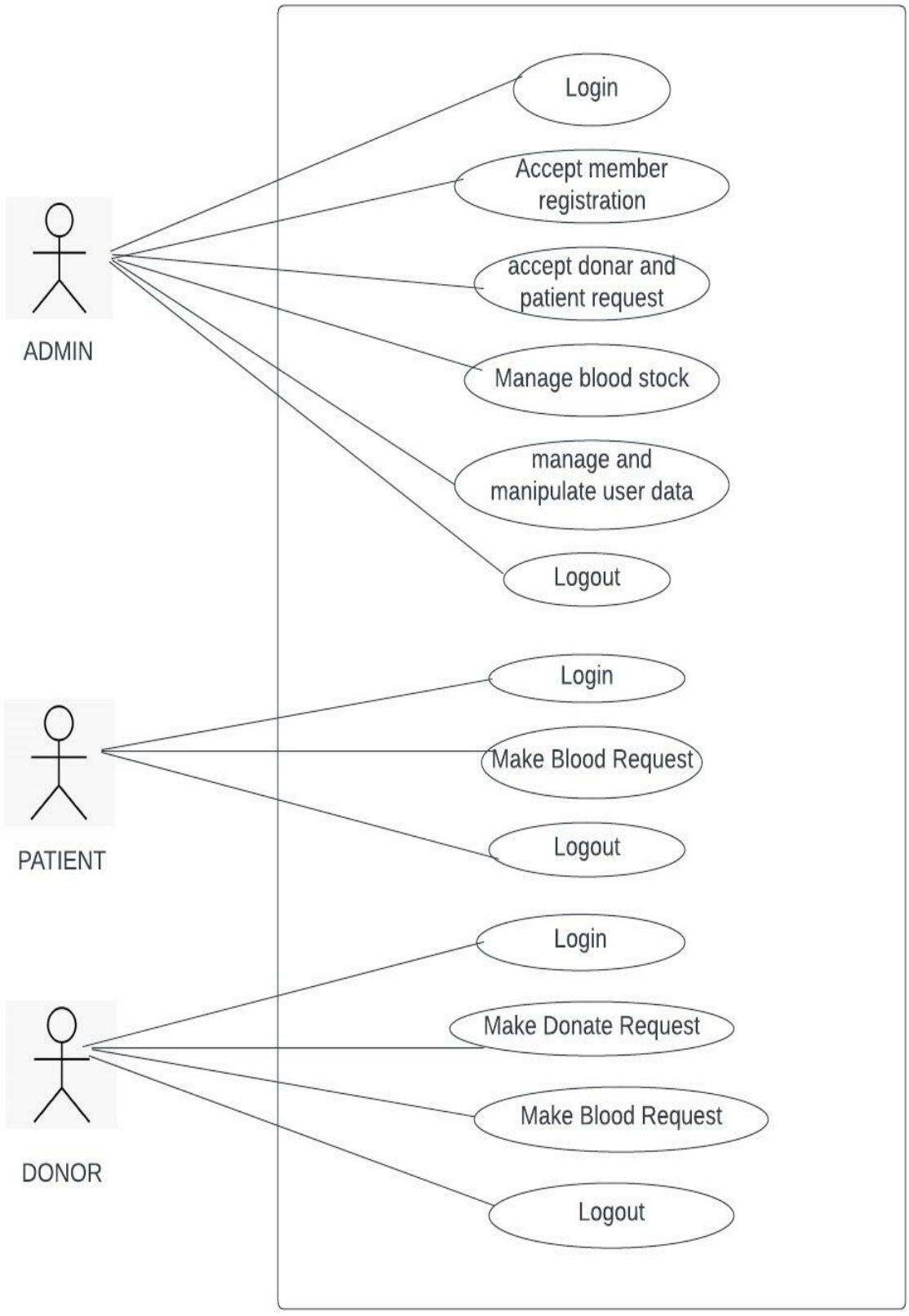
*4. Request Blood:*

User(Patient) should be able to request for blood at emergency situation, user need to define blood
group, location, required date, contact. The order requested will be sent to blood bank and then to the Inventory to check the availability. If available, the requested blood will be sent to the requested
donor(Hospital).

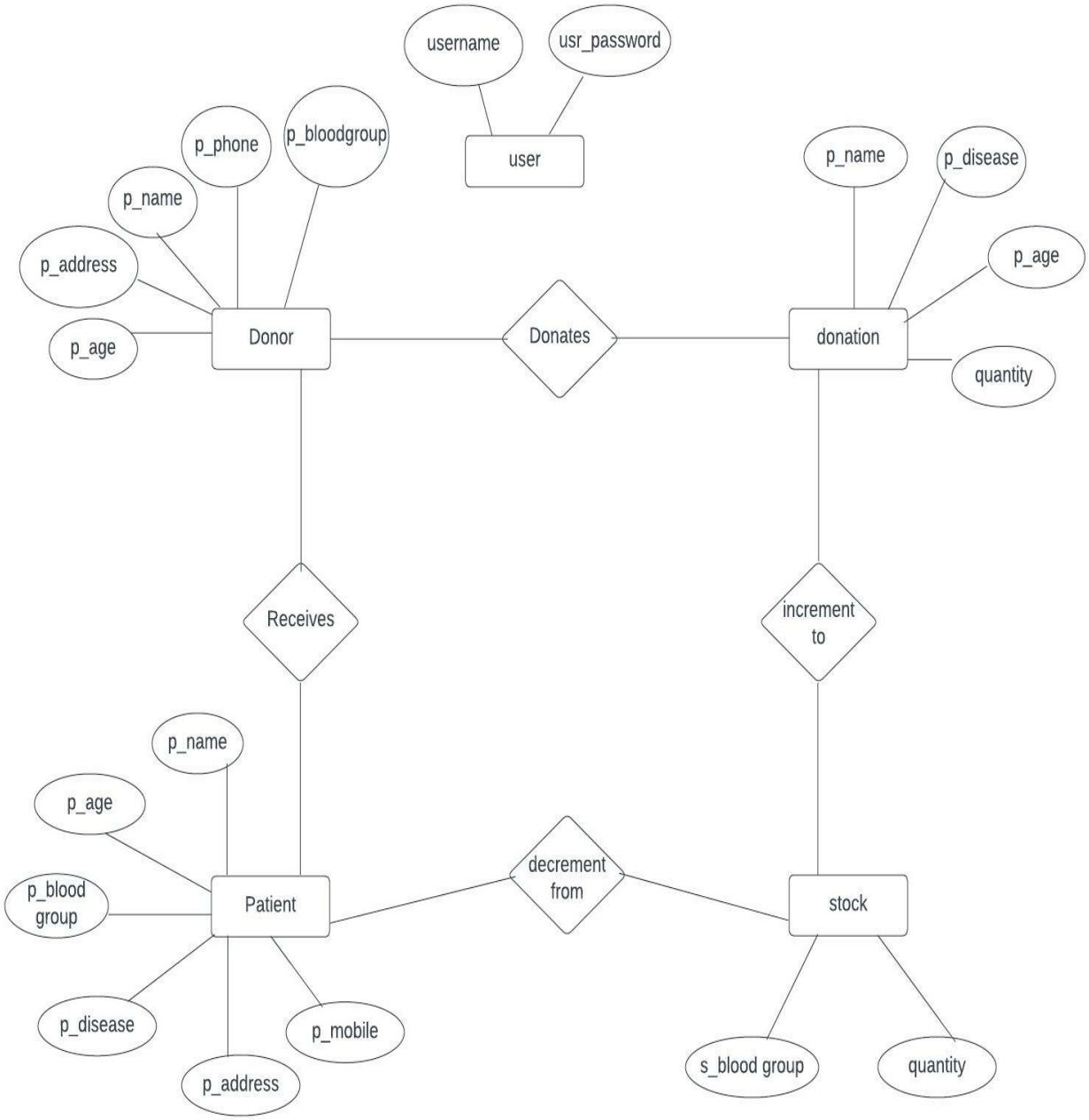
*5. View Request:*

The Blood Bank should be able to view received request and then respond to them and can search
requests by selecting two options select blood group and provision.

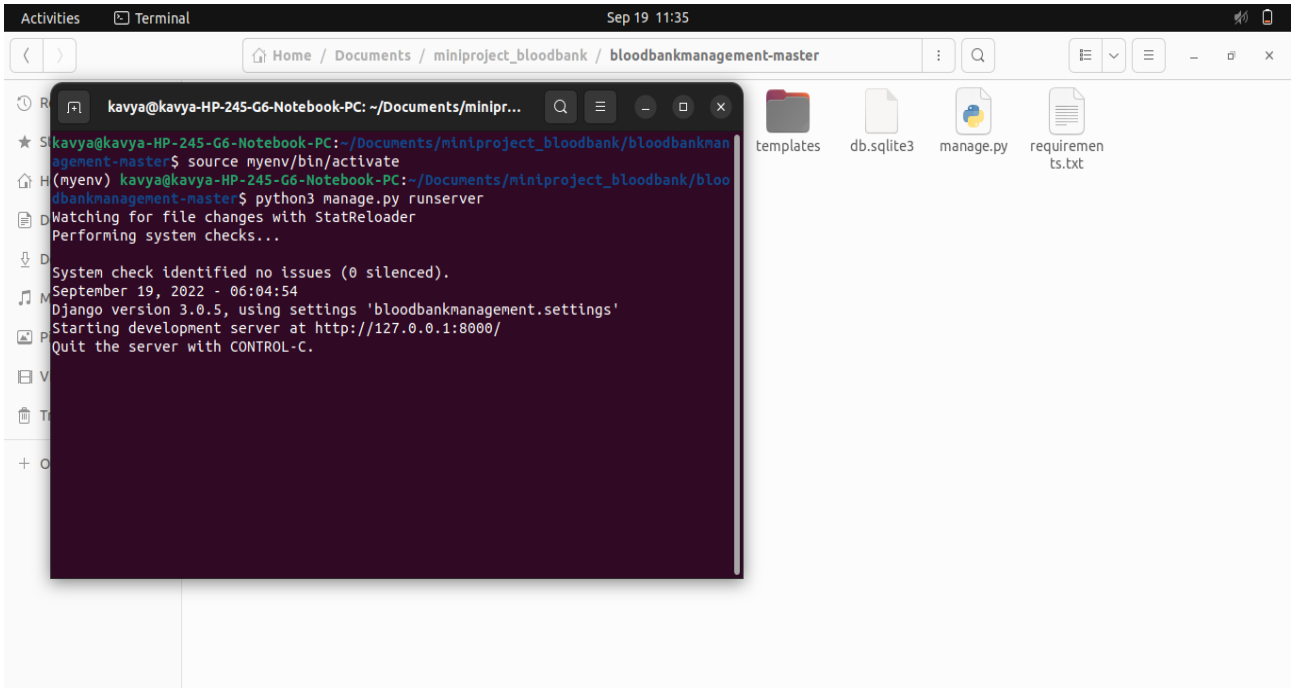
# 4.Usecase Diagrams:



# 5.ER DIAGRAM:


username
usr_password
user
p_phone
p_bloodgroup
p_name
p_address
p_age
Donor
Donates
p_name
p_disease
p_age
donation
quantity
Receives
increment to
p_name
p_age
p_blood group
Patient
decrement from
stock
p_disease
p_mobile
p_address
s_blood group
quantity

# 6.CODE SNIPPETS




Activities Text Editor Sep 19 11:33

Open views.py ~/Documents/miniproject_bloodbank/bloodbankmanagement-master/patient Save

```
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 blood import forms as bforms
from blood import models as bmodels


def patient_signup_view(request):
    userForm=forms.PatientUserForm()
    patientForm=forms.PatientForm()
    mydict={'userForm':userForm,'patientForm':patientForm}
    if request.method=='POST':
        userForm=forms.PatientUserForm(request.POST)
        patientForm=forms.PatientForm(request.POST,request.FILES)
        if userForm.is_valid() and patientForm.is_valid():
            user=userForm.save()
            user.set_password(user.password)
            user.save()
            patient=patientForm.save(commit=False)
            patient.user=user
            patient.bloodgroup=patientForm.cleaned_data['bloodgroup']
            patient.save()
            my_patient_group = Group.objects.get_or_create(name='PATIENT')
            my_patient_group[0].user_set.add(user)
        return HttpResponseRedirect('patientlogin')
    return render(request,'patient/patientsignup.html',context=mydict)

def patient_dashboard_view(request):
    patient= models.Patient.objects.get(user_id=request.user.id)
    dict={
```

Bracket match found on line: 63 Python 2 Tab Width: 8 Ln 63, Col 85 INS

Activities Text Editor Sep 19 11:33

Open views.py ~/Documents/miniproject_bloodbank/bloodbankmanagement-master/donor Save

```
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 blood import forms as bforms
from blood import models as bmodels
from donor import models as dmodels

def donor_signup_view(request):
    userForm=forms.DonorUserForm()
    donorForm=forms.DonorForm()
    mydict={'userForm':userForm,'donorForm':donorForm}
    if request.method=='POST':
        userForm=forms.DonorUserForm(request.POST)
        donorForm=forms.DonorForm(request.POST,request.FILES)
        if userForm.is_valid() and donorForm.is_valid():
            user=userForm.save()
            user.set_password(user.password)
            user.save()
            donor=donorForm.save(commit=False)
            donor.user=user
            donor.bloodgroup=donorForm.cleaned_data['bloodgroup']
            donor.save()
            my_donor_group = Group.objects.get_or_create(name='DONOR')
            my_donor_group[0].user_set.add(user)
        return HttpResponseRedirect('donorlogin')
    return render(request,'donor/donorsignup.html',context=mydict)


def donor_dashboard_view(request):
    donor= models.Donor.objects.get(user_id=request.user.id)
```

Python 2 Tab Width: 8 Ln 43, Col 68 INS

Activities Text Editor Sep 19 11:32

Open views.py Save

~/Documents/miniproject_bloodbank/bloodbankmanagement-master/blood

```
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+"
```

Python 2 Tab Width: 8 Ln 9, Col 39 INS

Activities Text Editor Sep 19 11:31

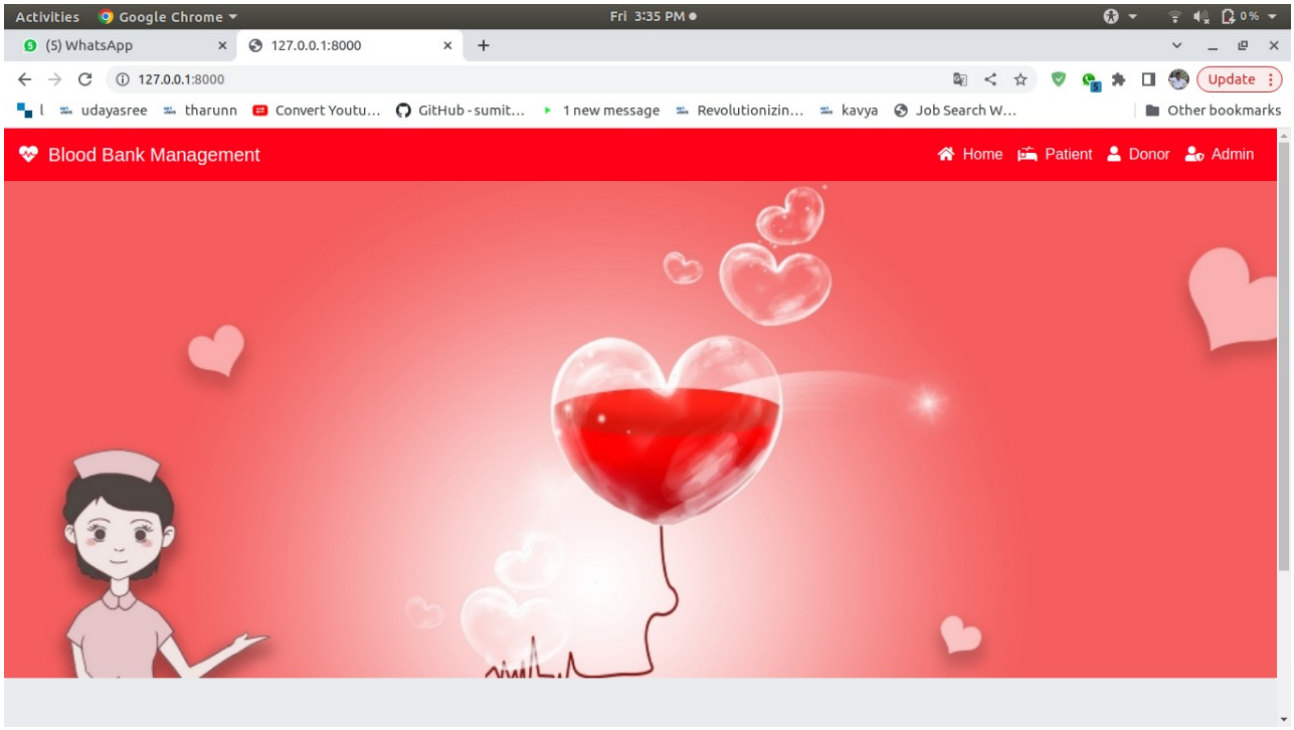
Open manage.py Save

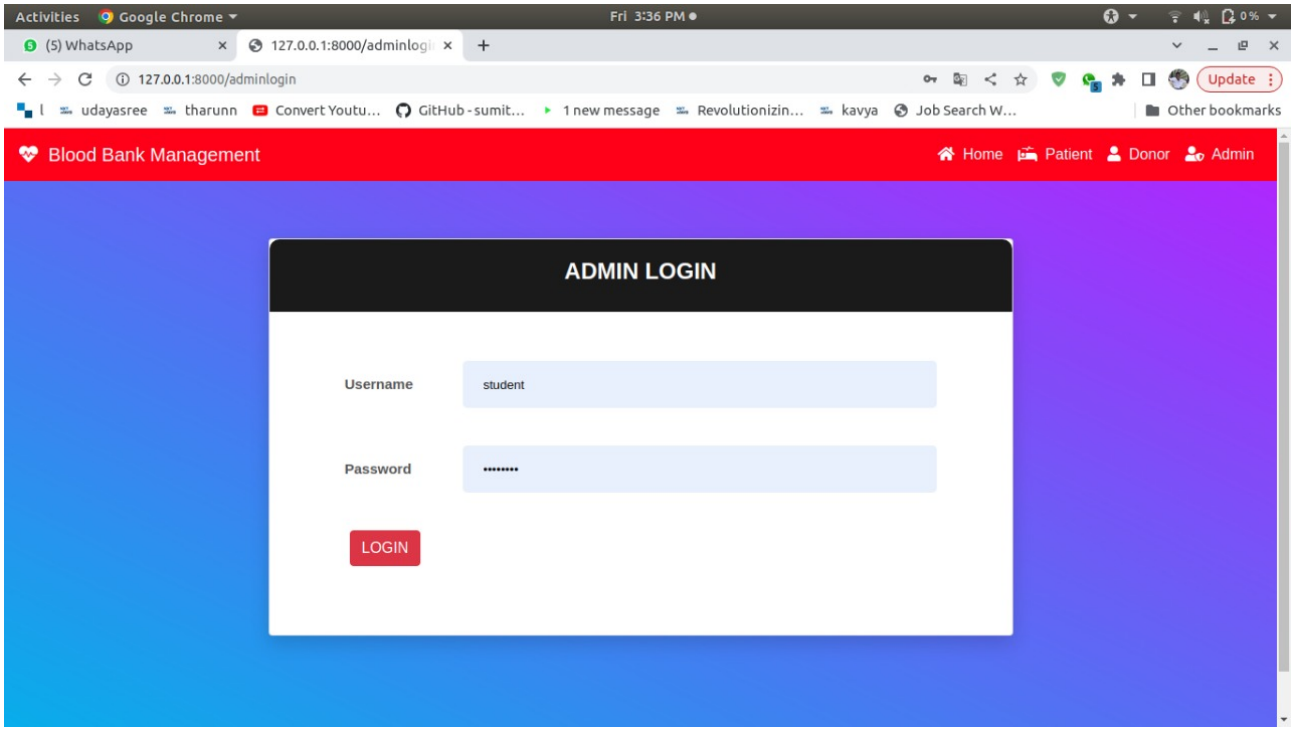
~/Documents/miniproject_bloodbank/bloodbankmanagement-master

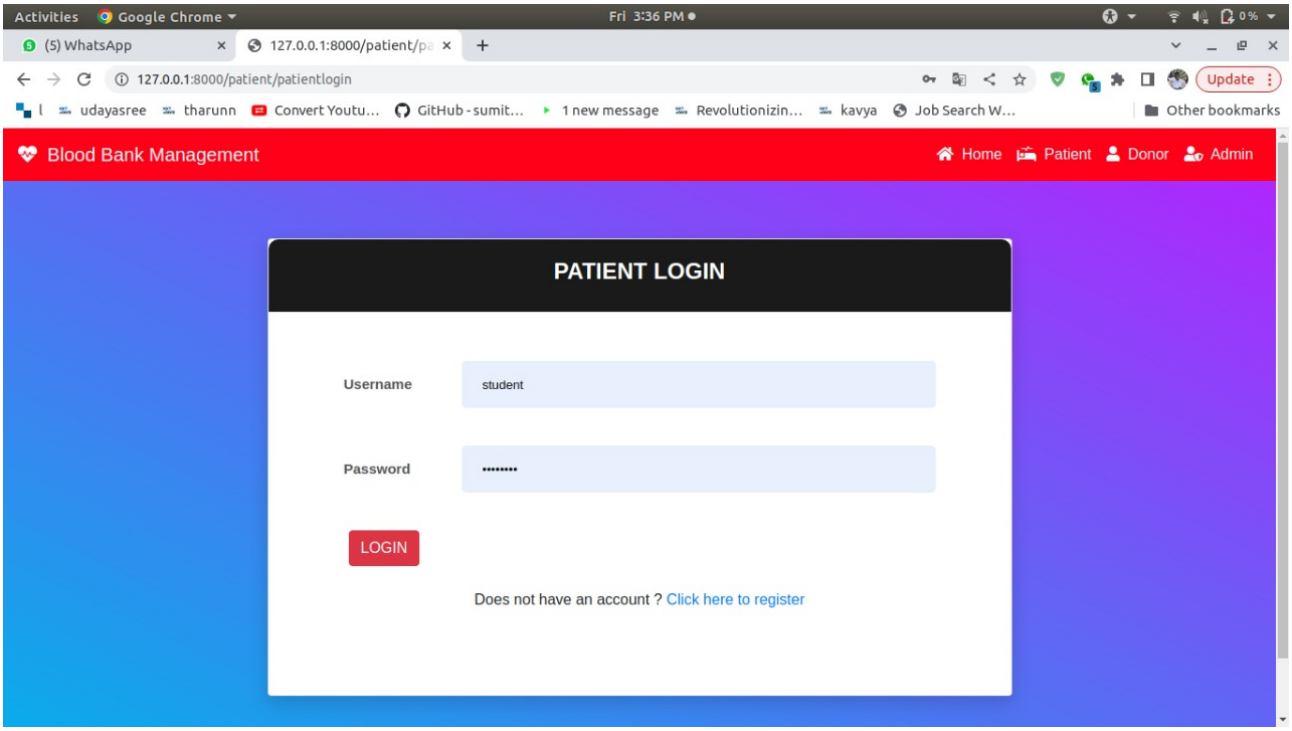
```
#!/usr/bin/env python
"""Django's command-line utility for administrative tasks."""
import os
import sys


def main():
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'bloodbankmanagement.settings')
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)


if __name__ == '__main__':
    main()
```

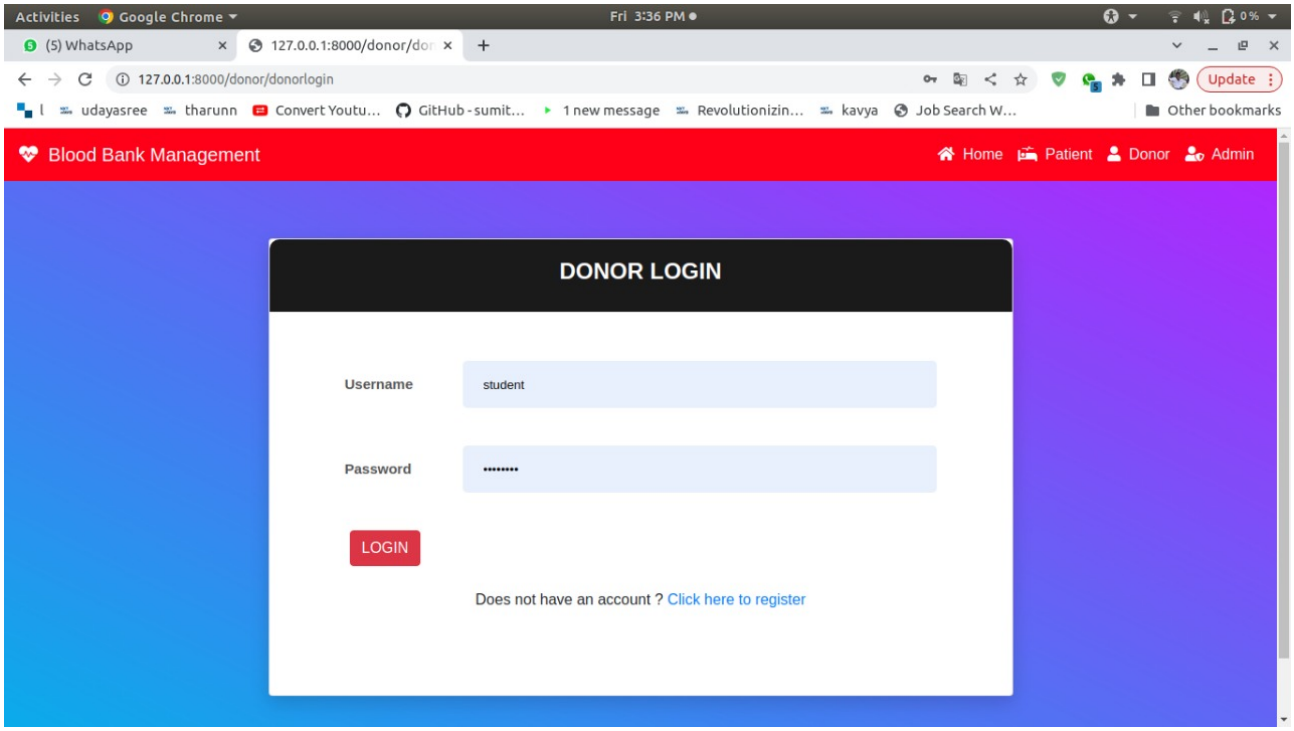
Bracket match found on line: 21 Python 2 Tab Width: 8 Ln 21, Col 11 INS

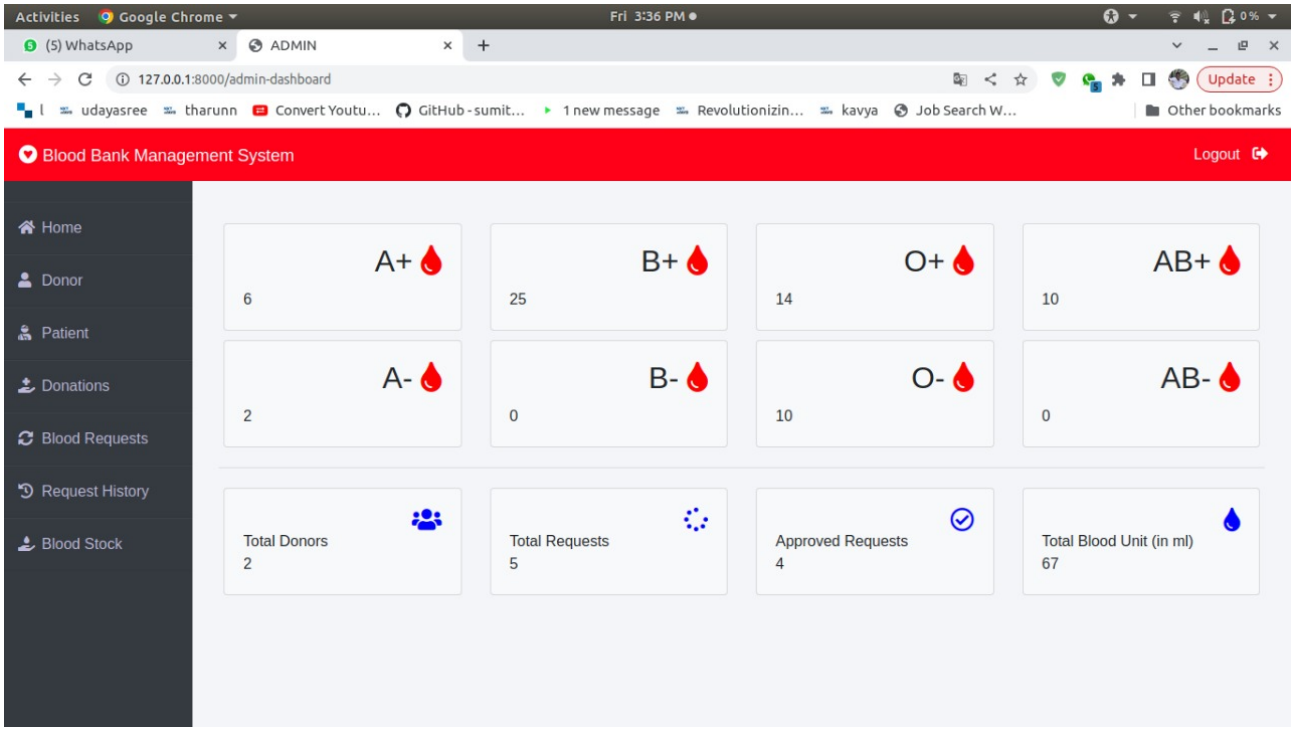
# 7.PROJECT IMAGES:

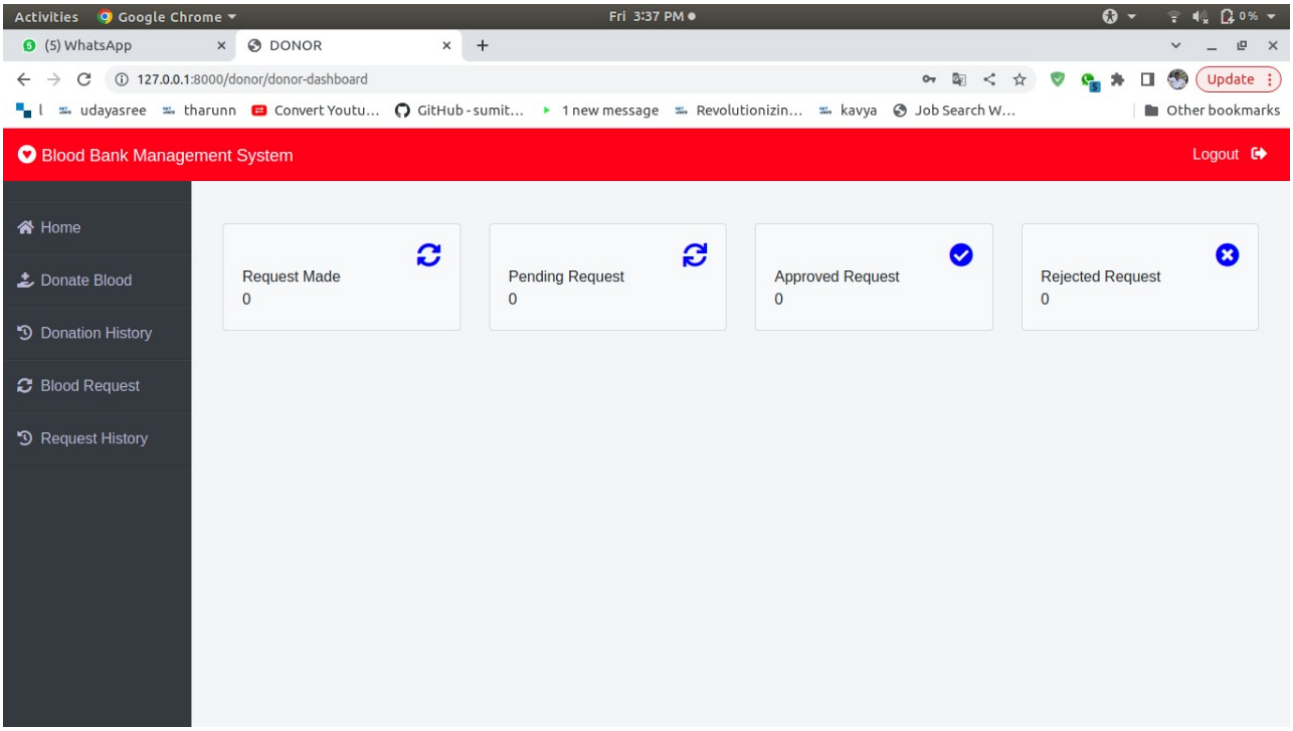


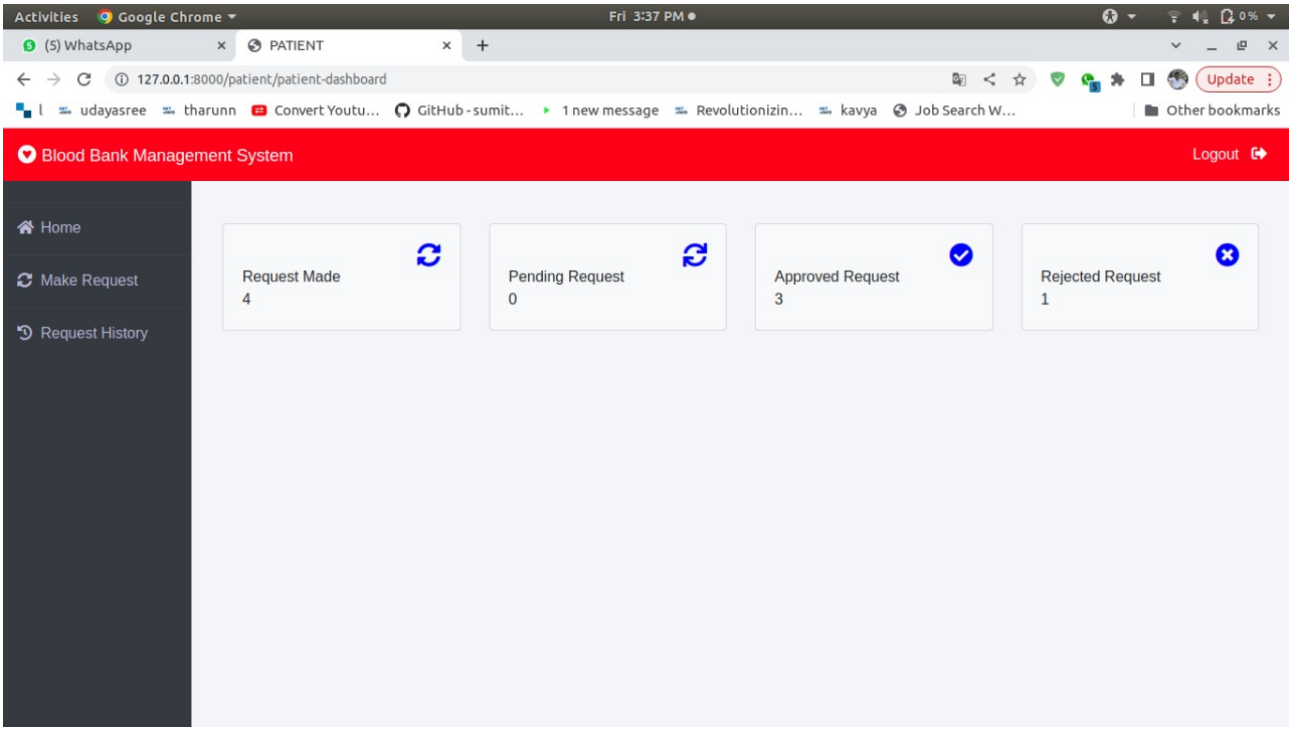



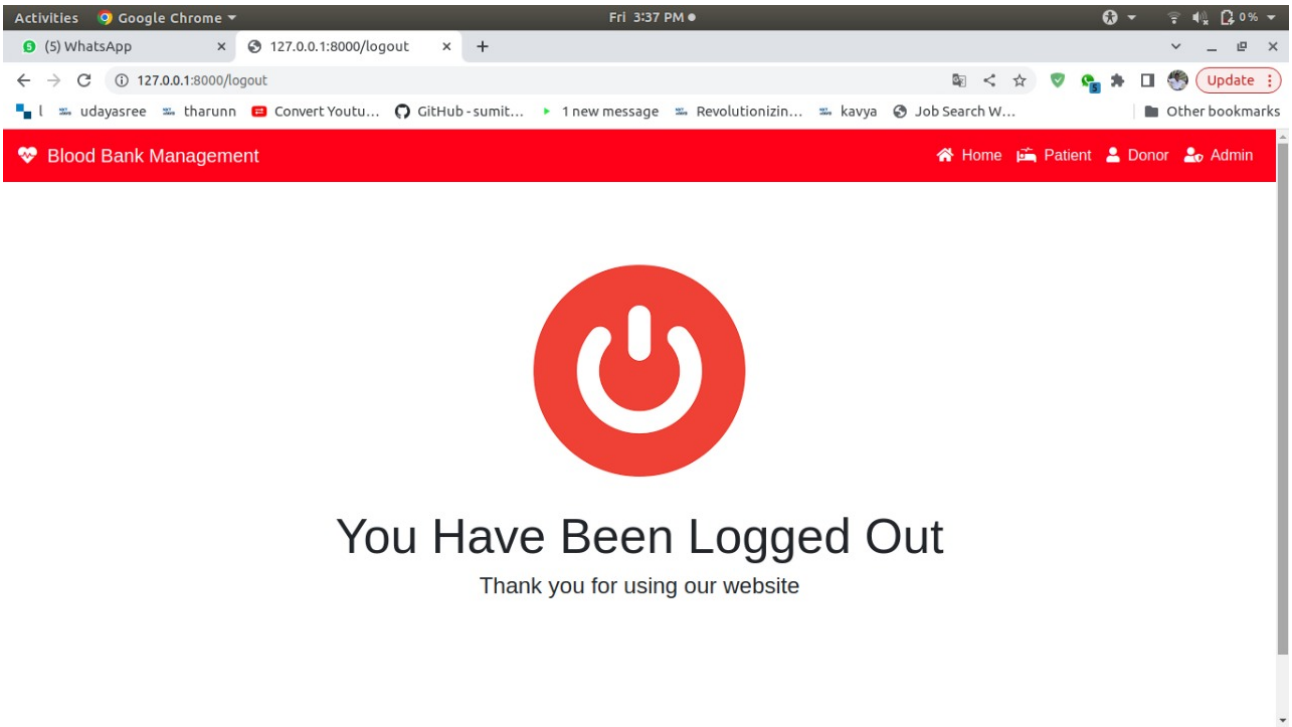

Activities
Google Chrome
Fri 3:36 PM
(5) WhatsApp
127.0.0.1:8000/patient/pa
127.0.0.1:8000/patient/patientlogin
Update
udayasree
tharunn
Convert Youtu...
GitHub - sumit...
1 new message
Revolutionizin...
kavya
Job Search W...
Other bookmarks
Blood Bank Management
Home
Patient
Donor
Admin
PATIENT LOGIN
Username
student
Password
LOGIN
Does not have an account ? Click here to register


Activities
Google Chrome
Fri 3:36 PM
(5) WhatsApp
127.0.0.1:8000/donor/do
127.0.0.1:8000/donor/donorlogin
Update
udayasree
tharunn
Convert Youtu...
GitHub - sumit...
1 new message
Revolutionizin...
kavya
Job Search W...
Other bookmarks
Blood Bank Management
Home
Patient
Donor
Admin
DONOR LOGIN
Username
student
Password
LOGIN
Does not have an account ? Click here to register


Activities
Google Chrome
Fri 3:36 PM
(5) WhatsApp
ADMIN
127.0.0.1:8000/admin-dashboard
Update
udayasree
tharunn
Convert Youtu...
GitHub - sumit...
1 new message
Revolutionizin...
kavya
Job Search W...
Other bookmarks
Blood Bank Management System
Logout
Home
Donor
Patient
Donations
Blood Requests
Request History
Blood Stock
A+
6
B+
25
O+
14
AB+
10
A-
2
B-
0
O-
10
AB-
0
Total Donors
2
Total Requests
5
Approved Requests
4
Total Blood Unit (in ml)
67


Activities
Google Chrome
Fri 3:37 PM
(5) WhatsApp
DONOR
127.0.0.1:8000/donor/donor-dashboard
Update
udayasree
tharunn
Convert Youtu...
GitHub - sumit...
1 new message
Revolutionizin...
kavya
Job Search W...
Other bookmarks
Blood Bank Management System
Logout
Home
Donate Blood
Donation History
Blood Request
Request History
Request Made
0
Pending Request
0
Approved Request
0
Rejected Request
0


Activities
Google Chrome
Fri 3:37 PM
(5) WhatsApp
PATIENT
127.0.0.1:8000/patient/patient-dashboard
Update
udayasree
tharunn
Convert Youtu...
GitHub - sumit...
1 new message
Revolutionizin...
kavya
Job Search W...
Other bookmarks
Blood Bank Management System
Logout
Home
Make Request
Request History
Request Made
4
Pending Request
0
Approved Request
3
Rejected Request
1


Activities
Google Chrome
Fri 3:37 PM
(5) WhatsApp
127.0.0.1:8000/logout
127.0.0.1:8000/logout
Update
udayasree
tharunn
Convert Youtu...
GitHub - sumit...
1 new message
Revolutionizin...
kavya
Job Search W...
Other bookmarks
Blood Bank Management
Home
Patient
Donor
Admin
You Have Been Logged Out
Thank you for using our website

## 8.CONCLUSION:

This proposed Blood Bank Management System gives a reliable platform for both donors and acceptors. The BMMS is a web-based application that helps to minimize human errors and problems pertainingto data redundancy. It is a fast-paced and efficient way tocommunicate without any security threats as the data entered will be verified and frequently updated thereby increasing the probability of saving one's life.

## 9.REFERENCES:

www.internshala.com
www.github.com
www.youtube.com
www.fullstack.com