

UCS1617-MINI PROJECT MODEL PRACTICALS

Name: S.Nachammai Devi Pooja

Roll No: 185001096

Date:4/5/2021

Class & Sec: CSE-B

Problem Statement:

The reason to donate blood is very simple it helps to save lives. This project will help us to gather blood from various sources and distribute it to the needy people who have high requirements. The need for a blood bank is to manage blood is to manage blood groups which are rare and unavailable during emergency. This management system aims to make all transactions on computer system instead of paperwork. Thus the transferring of blood process is made easier. It checks the availability of the required blood group, provides necessary information about the volunteer blood donors who have registered in the database. The software is designed to handle the daily transactions of the blood bank and search the details when required. It also helps to register the details of donors, blood collection details as well as blood issued reports. It is designed in such a manner that it can suit the needs of all the blood bank requirements in the course of future.

A Donor registration form is created and once filled is registered, their blood samples of 5 units is collected and it is updated in the database for future use.

The Requestor is also given a form and registered if the donor is available then the blood samples requested are given if the donor is not available, the donor details are shown So the requestor can contact them for more information. Once the blood samples requested are given and database is updated based on the reduction of blood Units of the relevant blood groups.

States:

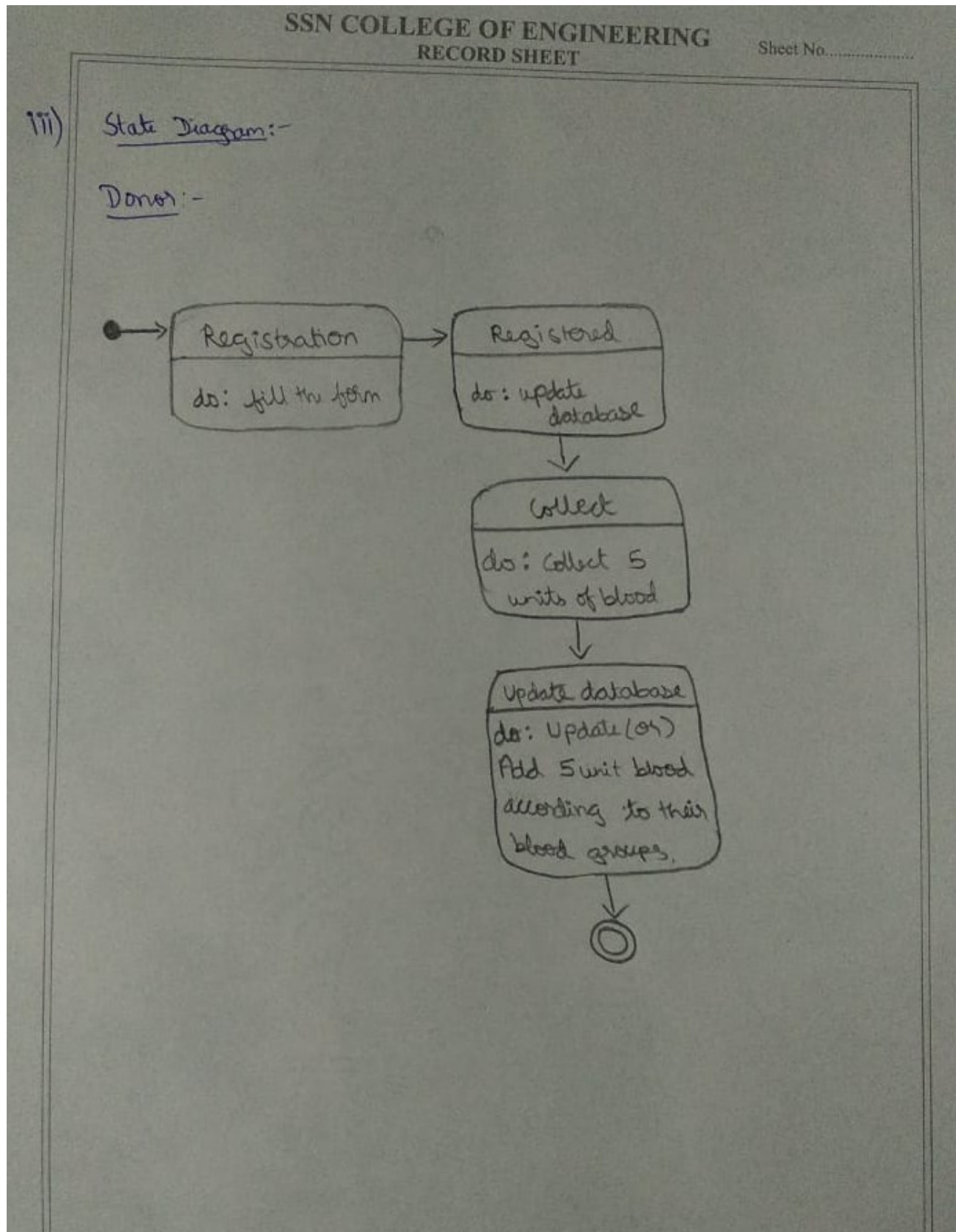
- Registration
- Registered
- Collect blood
- Give blood
- Update database
- Add blood respectively
- Notify blood donated
- Notify blood collected
- Show Donor details

Activities:

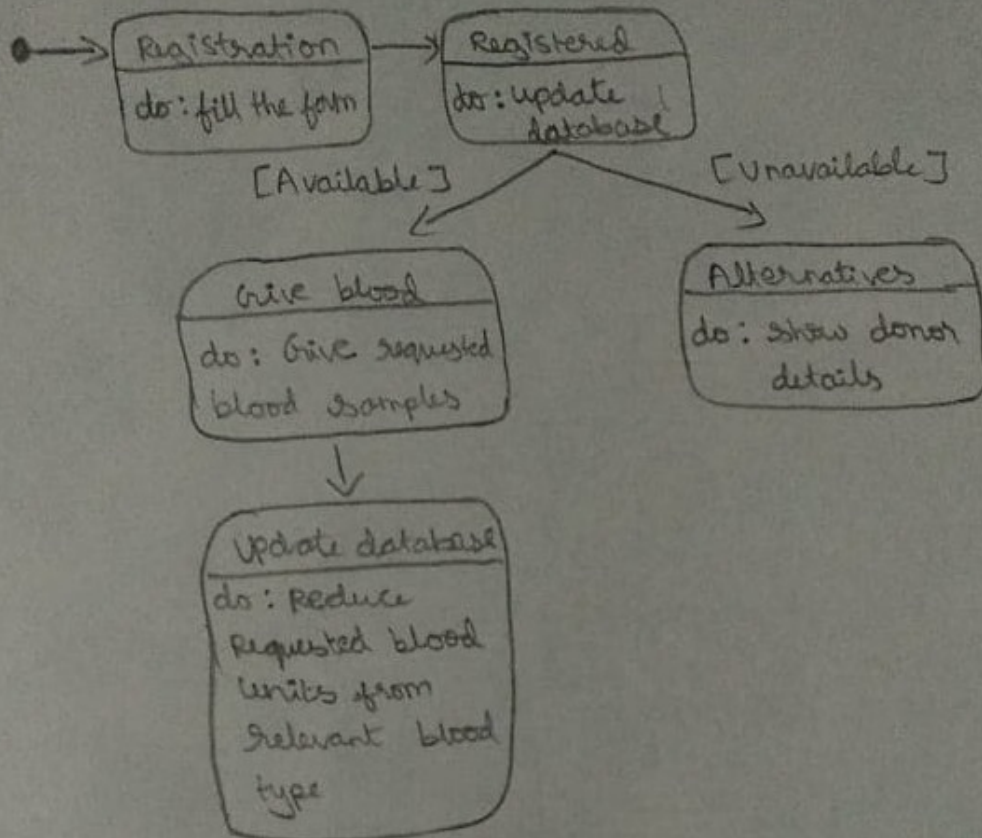
- * Select user type
- * Fill form details
- * Register details
- * Donate blood if donor
- * Check availability if requestor

- * Give needed blood if available
- * Show donor details
- * Update database
- * Notify user

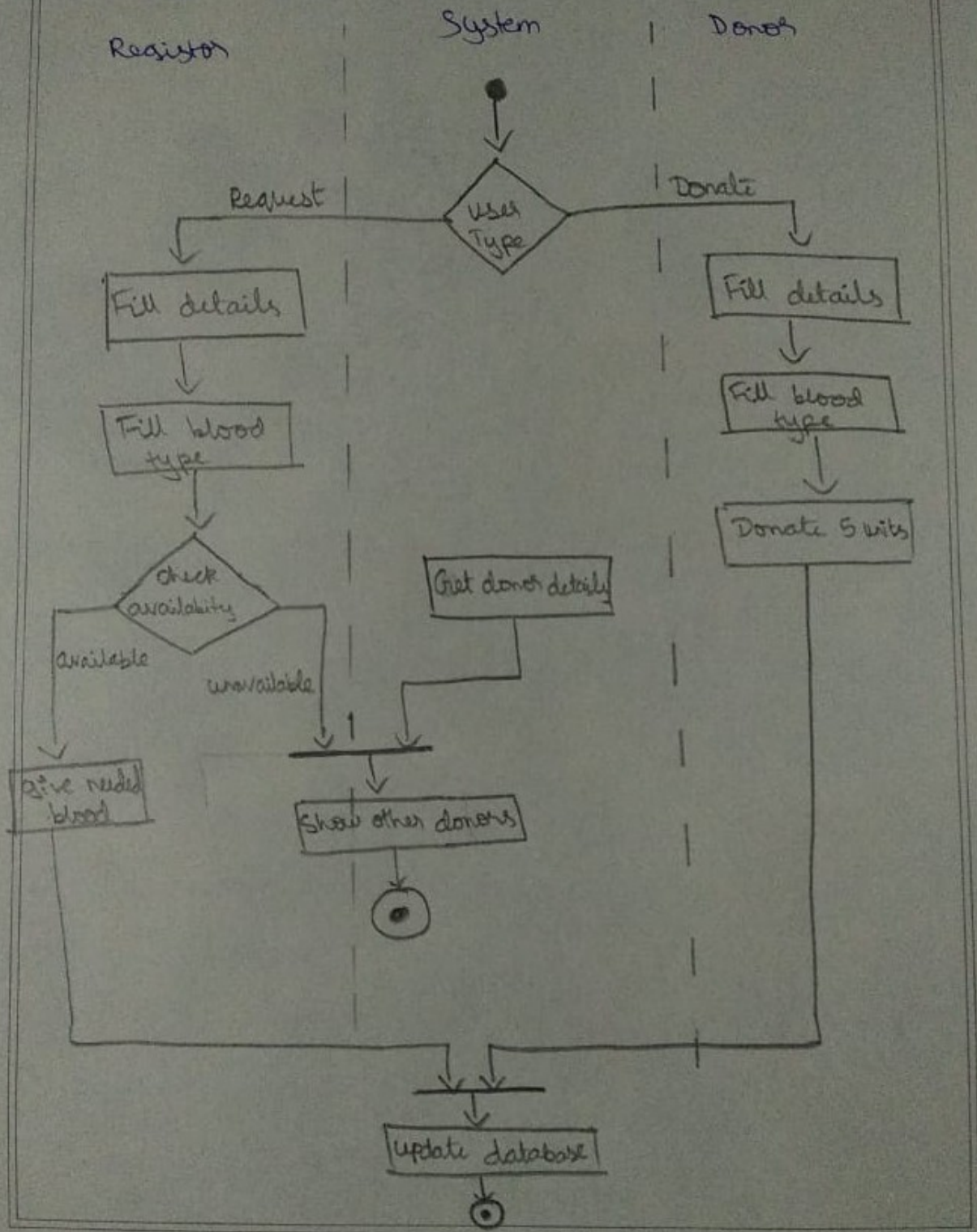
State Diagram & Activity Diagram:



Requestor :-



Activity Diagram: -



IMPLEMENTATION:

app.py:

```
from operator import add
from re import A
import re
from flask import Flask, render_template, url_for, request, redirect, session, g
from flask.helpers import send_file
from flask_sqlalchemy import SQLAlchemy
from sqlalchemy.orm import backref
app = Flask(__name__)
app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///bloodbank.db'
app.static_folder = 'static'
app.config['SECRET_KEY'] = '185001096'
db = SQLAlchemy(app)
```

```
class Blood(db.Model):
    __tablename__ = 'blood'
    id = db.Column(db.Integer, primary_key=True)
    bloodtype = db.Column(db.String(5))
    volume = db.Column(db.Integer)
```

```
class Donor(db.Model):
    __tablename__ = 'donor'
    id = db.Column(db.Integer, primary_key=True)
    Name = db.Column(db.String(50), nullable=False)
    Email = db.Column(db.String(200), nullable=False)
    Contact = db.Column(db.String(15))
    Location = db.Column(db.String(30))
    bloodtype = db.Column(db.String(5))
```

```
class Requester(db.Model):
    __tablename__ = 'request'
    id = db.Column(db.Integer, primary_key=True)
    Name = db.Column(db.String(50), nullable=False)
    Email = db.Column(db.String(200), nullable=False)
    Contact = db.Column(db.String(15))
    Location = db.Column(db.String(30))
    bloodtype = db.Column(db.String(5))
    volume = db.Column(db.Integer)
```

```
@app.route('/', methods=['POST', 'GET'])
def index():
    return render_template('index.html')
```

```
@app.route('/requestor', methods=['POST', 'GET'])
def requestor():
    if request.method == 'POST':
        name = request.form["name"]
        email = request.form["email"]
        contact = request.form["contact"]
```



```

location = request.form["location"]
bloodtype = request.form["bloodtype"]
unit = int(request.form["unit"])

req=Requester(Name=name,Email=email,Contact=contact,Location=location,bloodtype=bloodtype
,volume=unit)
    try:
        db.session.add(req)
        db.session.commit()
    except:
        return "error adding values to database"
rem=Blood.query.filter_by(bloodtype=bloodtype).first()
if rem.volume-req.volume > 0:
    rem.volume =rem.volume - req.volume
    try:
        db.session.add(req)
        db.session.commit()
    except:
        return "error adding values to database"
    return render_template('available.html')
else:
    donors= Donor.query.filter_by(bloodtype=bloodtype).all()
    return render_template('table.html',donors=donors)
else:
    return render_template('request.html')

@app.route('/donate', methods=['POST', 'GET'])
def donate():
    if request.method == 'POST':
        name = request.form["name"]
        email = request.form["email"]
        contact = request.form["contact"]
        location = request.form["location"]
        bloodtype = request.form["bloodtype"]

donor=Donor(Name=name,Email=email,Contact=contact,Location=location,bloodtype=bloodtype)
    blood=Blood.query.filter_by(bloodtype=bloodtype).first()
    blood.volume += 5
    try:
        db.session.add(donor)
        db.session.add(blood)
        db.session.commit()
    except:
        return "error adding values to database"
        return "You have Successfully donated. Thankyou!!"
    else:
        return render_template('donor.html')

if __name__ == "__main__":
    app.run(debug=True)

```

index.html:

```
<html>
<head>

</head>
<body>
<center>
<h1>Welcome to RED CROSS BLOOD BANK</h1>
<h3>Select who you are</h3>
<a href="/requestor">Requestor</a><br>
<a href="/donate">Donor</a><br>
</center>
</body>
</html>
```

donor.html:

```
<html>
<head>

</head>
<body>
  <center>
    <form method="post" action="/donate">
      <p>Enter Name:</p>
      <input type="text" name="name" >
      <p>Enter email:</p>
      <input type="email" name="email" >
      <p>Enter Phone number:</p>
      <input type="phone" name="contact" >
      <p>Enter Blood type:</p>
      <input type="radio" id="opositive" name="bloodtype" value="o+ve">
      <label for="opositive">O+ve</label><br><br>
      <input type="radio" id="onegative" name="bloodtype" value="o-ve">
      <label for="onegative">O-ve</label><br><br>
      <input type="radio" id="abpositive" name="bloodtype" value="AB+ve">
      <label for="abpositive">AB+ve</label><br><br>
      <p>Enter Location:</p>
      <input type="text" name="location" >
      <br><br>
      <input type="submit" value="submit">
    </form>
  </center>
</body>
</html>
```

request.html

```
<html>
  <head>
```

```

</head>
<body>
  <center>
    <form method="post" action="/requestor">
      <p>Enter Name:</p>
      <input type="text" name="name" >
      <p>Enter email:</p>
      <input type="email" name="email" >
      <p>Enter Phone number:</p>
      <input type="phone" name="contact" >
      <p>Enter Blood type:</p>
      <input type="radio" id="opositive" name="bloodtype" value="o+ve">
      <label for="opositive">O +ve</label><br><br>
      <input type="radio" id="onegative" name="bloodtype" value="o-ve">
      <label for="onegative">O -ve</label><br><br>
      <input type="radio" id="abpositive" name="bloodtype" value="AB+ve">
      <label for="abpositive">AB +ve</label><br><br>
      <p>Enter Location:</p>
      <input type="text" name="location" >
      <br>
      <p>Enter Number of units required:</p>
      <input type="text" name="unit" >
      <br><br>
      <input type="submit" value="submit">
    </center>
  </body>
</html>

```

table.html

```

<html>
  <head>

  </head>
  <body>
    <style>
      table {
        font-family: arial, sans-serif;
        border-collapse: collapse;
        width: 100%;
      }
      td, th {
        border: 1px solid #dddddd;
        text-align: left;
        padding: 8px;
      }

      tr:nth-child(even) {
        background-color: #dddddd;
      }
    </style>
    <center> <h2>The request Blood is not available with us.</h2>

```



```

    <h2>Try contacting below donors</h2><h2>List of Available Donors</h2> </center>
<table>
  <tr>
    <th>Donor Name:</th>
    <th>Contact</th>

  </tr>
  {%for donor in donors%}
  <tr>
    <td>{{ donor.Name }}</td>
    <td>{{ donor.Contact }}</td>

  </tr>
  {%endfor%}
  <tr>
    <td>Raja</td>
    <td>9894465024</td>

  </tr>
  <tr>
    <td>Siva</td>
    <td>9941330536</td>

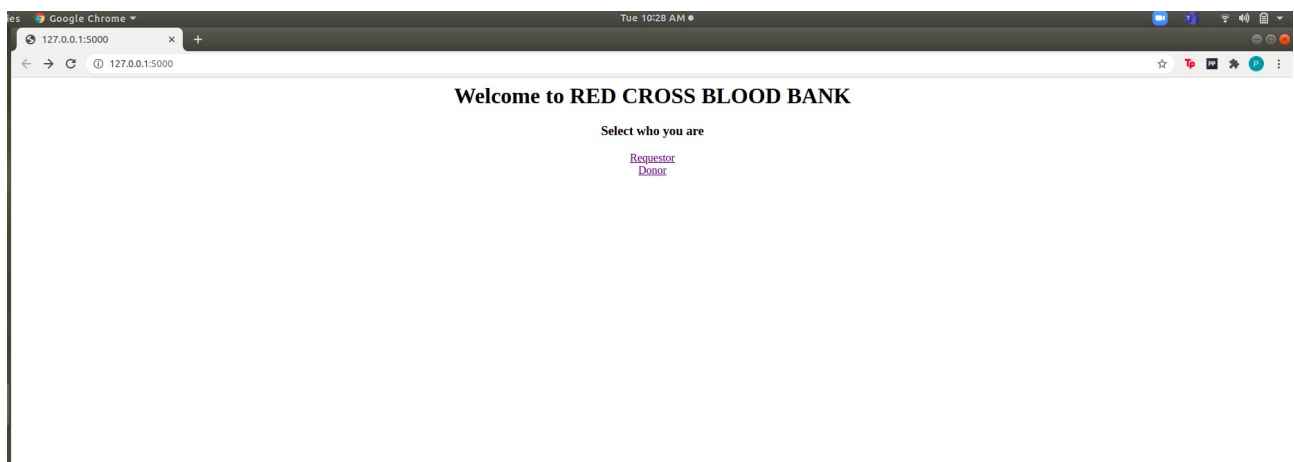
  </tr>

</table>
</body>
</html>

```

OUTPUTS:

Main page:



Requestor:

127.0.0.1:5000/requestor x +

127.0.0.1:5000/requestor

Enter Name:

Enter email:

Enter Phone number:

Enter Blood type:
☐ O +ve
☒ O -ve
☐ AB +ve

Enter Location:

Enter Number of units required:

127.0.0.1:5000/requestor x +

127.0.0.1:5000/requestor

The requested Blood Type is available with us. Please collect it from our hospital :

Donor:

127.0.0.1:5000/donate x +

127.0.0.1:5000/donate

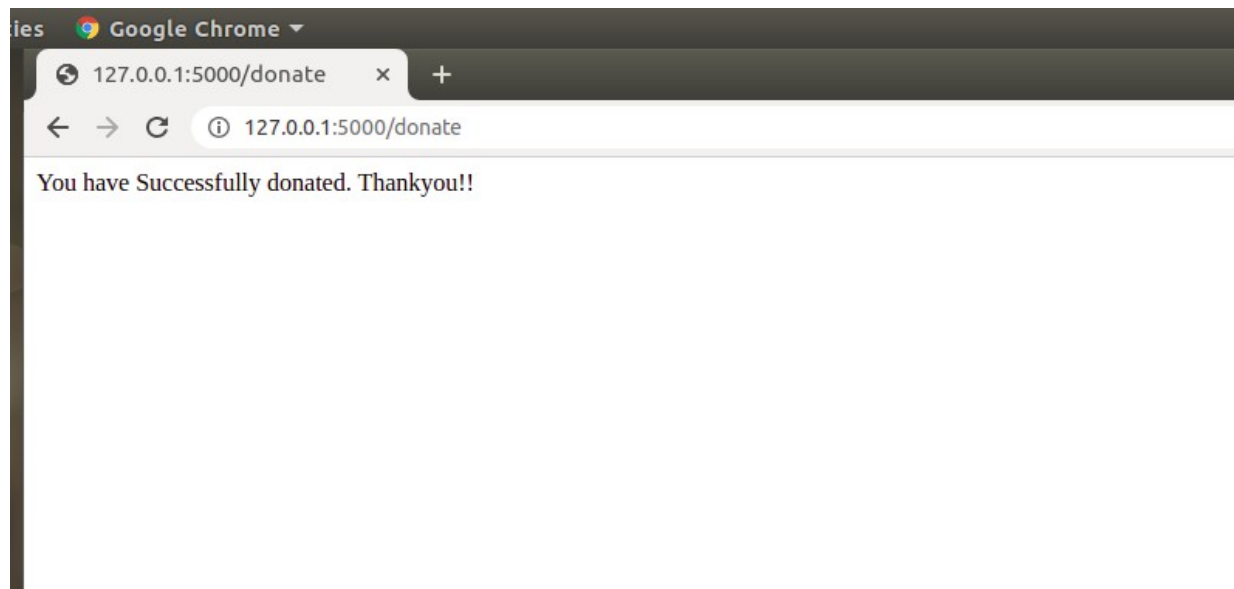
Enter Name:

Enter email:

Enter Phone number:

Enter Blood type:
☒ O+ve
☐ O-ve
☐ AB+ve

Enter Location:



Database:

File Edit View Help

New Database Open Database Write Changes Revert Changes

Database Structure Browse Data Edit Pragmas Execute SQL

Table: donor

	id	Name	Email	Contact	Location	bloodtype
Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	Pooja	ndpooja030...	9941330536	Chennai	o+ve
2	2	Praveen	vpraveenpl...	9894440357	Chennai	o-ve
3	3	Prabha	ramaprabha...	9952092232	Chennai	AB+ve
4	4	nachammai	nach@gmai...	9999999999	Trichy	o-ve

Insufficient:

127.0.0.1:5000/requestor
+
127.0.0.1:5000/requestor

Enter Name:

Enter email:

Enter Phone number:

Enter Blood type:
☐ O +ve
☐ O -ve
☒ AB +ve

Enter Location:

Enter Number of units required:

Database:

4	4	Prathyush S	prathyush1...	9655692510	Chennai	AB+ve	70
5	5	devip	devip@gma...	9710217584	madhurai	o-ve	6
6	6	poo	poo@gmail....	9444648788	Trichy	AB+ve	70

127.0.0.1:5000/requestor
+
127.0.0.1:5000/requestor

The request Blood is not available with us.
Try contacting below donors
List of Available Donors

Donor Name:	Contact
Prabha	9952092232
Raja	9894465024
Siva	9941330536

Test Cases:

Test case id	Test scenario	Testing steps	Test data	Expected outcome	Actual outcome	Pass/Fail
T101	Wrong name	Click request and enter name	Name:xy123	Invalid	valid	Fail
T102	Name of the requestor	Click request and enter name	Name:Pooja	Valid	Valid	Pass
T201	Phone number of Requestor	Click request and enter Phone number	Name:pooja Phone no:123	Invalid	valid	Fail
T202	Phone number of Requestor	Click request and enter Phone number	Name:pooja Phone no:9941330536	Valid	Valid	Pass
T301	Request blood greater than present in bank	Click request and enter details and type units required	Name:Pooja Phone no:9941339536 Requested unit: 60units Units present in bank :50 units	Invalid	Invalid	Fail
T302	Request blood less than present in bank	Click request and enter details and type units required	Name:Pooja Phone no:9941339536 Requested unit: 60units Units present in bank :70 units	valid	valid	Pass
T401	Fill details in request form	Click request and enter details	Name:Pooja Phoneno: Requestedunits:5 unit	Invalid	Valid	Fail
T402	Fill details in request form	Click request and enter details	Name:Pooja Phoneno:9941330536 Requestedunits:5 unit	valid	Valid	Pass
T501	Donor Filling type of blood	Click donate and enter details	Name:Pooja Bloodtype:O+	Valid	Valid	Pass
T502	Donor Filling type of blood	Click donate and enter details	Name:Pooja Bloodtype:	Inalid	Valid	Fail

Improvements:

In this Assignment, with the help of test cases I ensured that if different features within an application are working as expected. It validates whether the software is free of defects and is working as per the expectations of the end users.

With the help of test cases I ensured that if different features within an application are working as expected. It validates whether the software is free of defects and is working as per the expectations of the end users.