IBM NALAIYATHIRAN

PROJECT REPORT

SUBMITTED BY

JAYA HARI PRASATH M	962319104047
AJITH KANNA T	962319104010
ASWINI S	962319104026
BHARATHI H V	962319104033

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

AMRITA COLLEGE OF ENGINEERING AND TECHNOLOGY

ERACHAKULAM, NAGERCOIL.

ANNA UNIVERSITY::CHENNAI 600 025

JUNE 2022

CERTIFICATE OF EVALUATION

COLLEGE NAME	:	AMRITA COLLEG OF ENGINEERING AND TECHNOLOGY
BRANCH	:	COMPUTER SCIENCE AND ENGINEERING
SEMESTER	:	VII
TITLE	:	PLASMA DONOR APPLICATION
TEAM ID	:	PNT2022TMID51943

STUDENT NAMES		**REGISTRATION NUMBER**	**MENTOR**
JAYA HARI PRASATH M	-	962319104047	
AJITH KANNA	-	962319104010	**Mrs. KARPAGAVALLI C**
ASWINI S	-	962319104026	
BHARATHI HV	-	962319104033	

SUPERVISOR **EVALUATOR**

ACKNOWLEDGEMENT

First and foremost we would like to express our sincere gratitude to our respected Founder Amma , **Mata Amritanandamayi Devi** and Chairman **Mr. K S Ramasubban IAS(Retd)** for their blessings and grace in making our project great success.

We would like to place a record with deep sense of gratitude to our Honorable Principal **Dr. T Kannan** for having given us the opportunity to pursue B.E., course in this prestigious institution.

We would like to express our sincere thanks to our beloved Head of the Department **Dr. P M Siva Raja** and Project Coordinator **Mrs. S L Jothi Lakshmi**, for creating a supportive and a model environment for us to work and build up our innovative skills. We also thank our project Mentor **Mrs. Karpagavalli C** for the kind encouragement and moral support, who has been a constant source of inspiration to us.

We wish to express our sincere sense of gratitude to **Dr. P M Siva Raja**, Head, Department of Computer Science and Engineering and to our Project Guide who enabled us to complete our project successfully.

CONTENTS Page No

1 . INTRODUCTION

1.1 Project Overview

Objective: -

The main objective of developing the system is gather all donors under a single roof, and encourage to donate blood. And also create an efficient system for the donors, where they can get the notification of the blood request of the nearby blood requestor and the contact details of the requestor will be display on the application. Apart from that the donor can also donate the blood anytime by using this application.

Abstract: -

This project is aimed to developing an online Blood Donation Information. The entire project has been developed keeping in view of the distributed client server computing technology, in mind. The Blood Donation Agent is to create an e-Information about the donor and organization that are related to donating the blood. Through this application any person who is interested in donating the blood can register himself in the same way if any organization wants to register itself with this site that can also register. Moreover if any general consumer wants to make request blood online he can also take the help of this site. Admin is the main authority who can do addition, deletion, and modification if required.

1.2 Purpose: -

This system proposed here aims to create an information about the donor and the organization that are related to donating the blood. This application connecting the donors & the patients by an online application. By using this application, the users can either raise a request for plasma donation or requirement. This system is used if anyone needs a Plasma Donor.

2. LITERATURE SURVEY:

2.1 . Existing problem

In the existing system of plasma donor containing following issue that are:

- Blood shortage
- Donor and staff safety
- Handle multiple request at the same time
- It required active internet connection

2.2 References

1.https://nevonprojects.com/instant-plasma-donor-recipient-connector-android-app/

2. https://www.sciencedirect.com/topics/engineering/plasma-applications

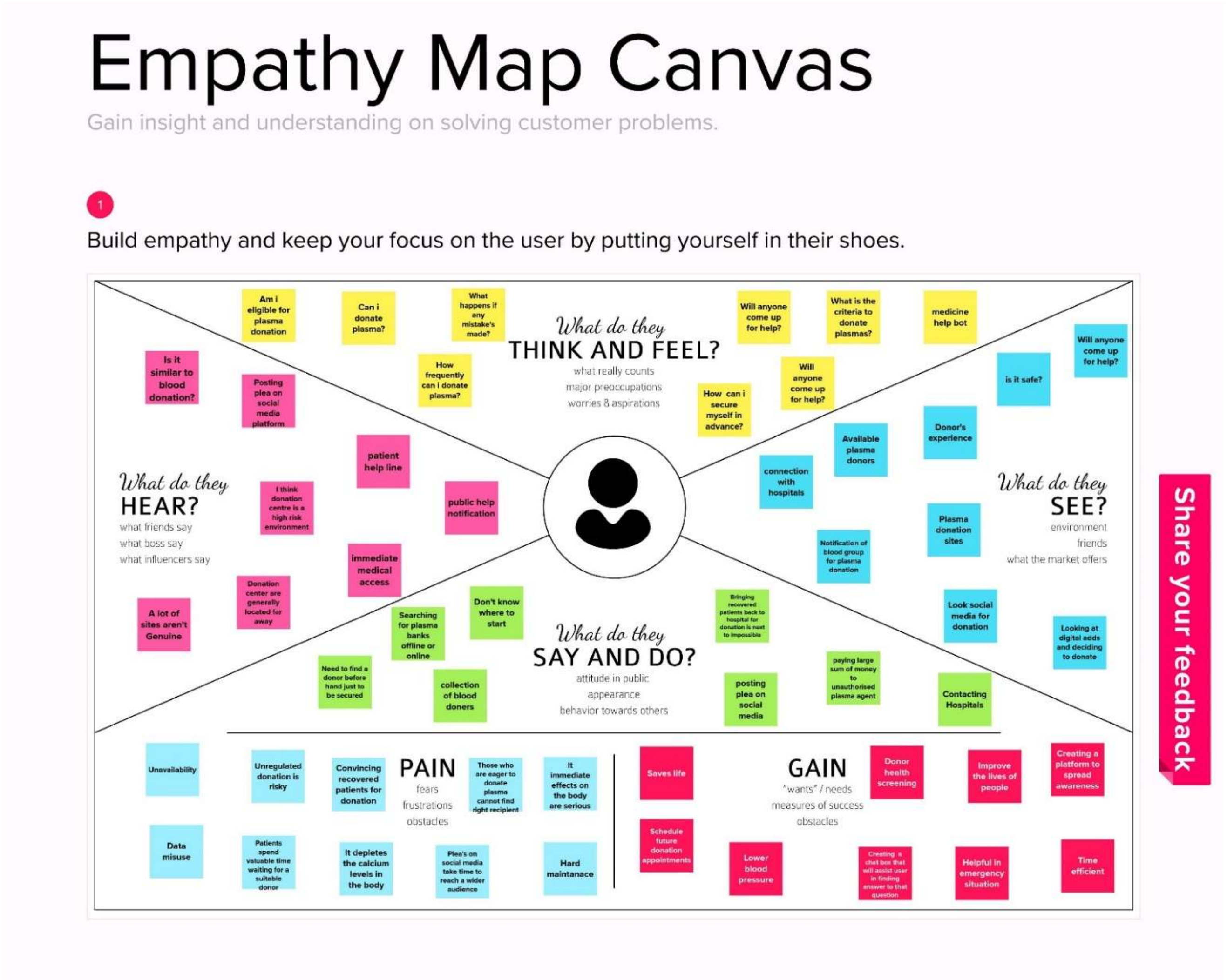
2.3 Problem Statement Definition

Plasma plays the critical role of maintaining a healthy blood pressure, blood volume and a proper pH balance. Which being the most important element for humans to survive, developing an application to interconnect plasma requesters and donors might help save lives during emergency. Making such an application which is user friendly as well as has more features for serving the people better, we are proposing a model in which we are going to connect the plasma donor and requester together in a unique way.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

- An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behaviour and attitudes.
- It is a useful tool to helps teams better understand their users.
- Creating an effective solution requires understanding the true problem and the person who is experiencing it.
- The exercise of creating the map helps participants consider things from the user’s perspective along with his or her goals and challenges.



Empathy Map Canvas

Gain insight and understanding on solving customer problems.

1

Build empathy and keep your focus on the user by putting yourself in their shoes.

What do they THINK AND FEEL?

what really counts
major preoccupations
worries & aspirations

Am I eligible for plasma donation

Can i donate plasma?

What happens if any mistake's made?

How frequently can i donate plasma?

Will anyone come up for help?

What is the criteria to donate plasmas?

medicine help bot

How can i secure myself in advance?

Will anyone come up for help?

What do they HEAR?

what friends say
what boss say
what influencers say

Is it similar to blood donation?

Posting plea on social media platform

patient help line

I think donation centre is a high risk environment

public help notification

immediate medical access

Donation center are generally located far away

A lot of sites aren't Genuine

What do they SEE?

environment
friends
what the market offers

Will anyone come up for help?

is it safe?

Donor's experience

Available plasma donors

connection with hospitals

Plasma donation sites

Notification of blood group for plasma donation

Look social media for donation

Looking at digital adds and deciding to donate

What do they SAY AND DO?

attitude in public
appearance
behavior towards others

Searching for plasma banks offline or online

Don't know where to start

Need to find a donor before hand just to be secured

collection of blood doners

Bringing recovered patients back to hospital for donation is next to impossible

posting plea on social media

paying large sum of money to unauthorised plasma agent

Contacting Hospitals

PAIN

fears
frustrations
obstacles

Unavailability

Unregulated donation is risky

Convincing recovered patients for donation

Those who are eager to donate plasma cannot find right recipient

It immediate effects on the body are serious

Data misuse

Patients spend valuable time waiting for a suitable donor

It depletes the calcium levels in the body

Plea's on social media take time to reach a wider audience

Hard maintanace

GAIN

"wants" / needs
measures of success
obstacles

Saves life

Donor health screening

Improve the lives of people

Creating a platform to spread awareness

Schedule future donation appointments

Lower blood pressure

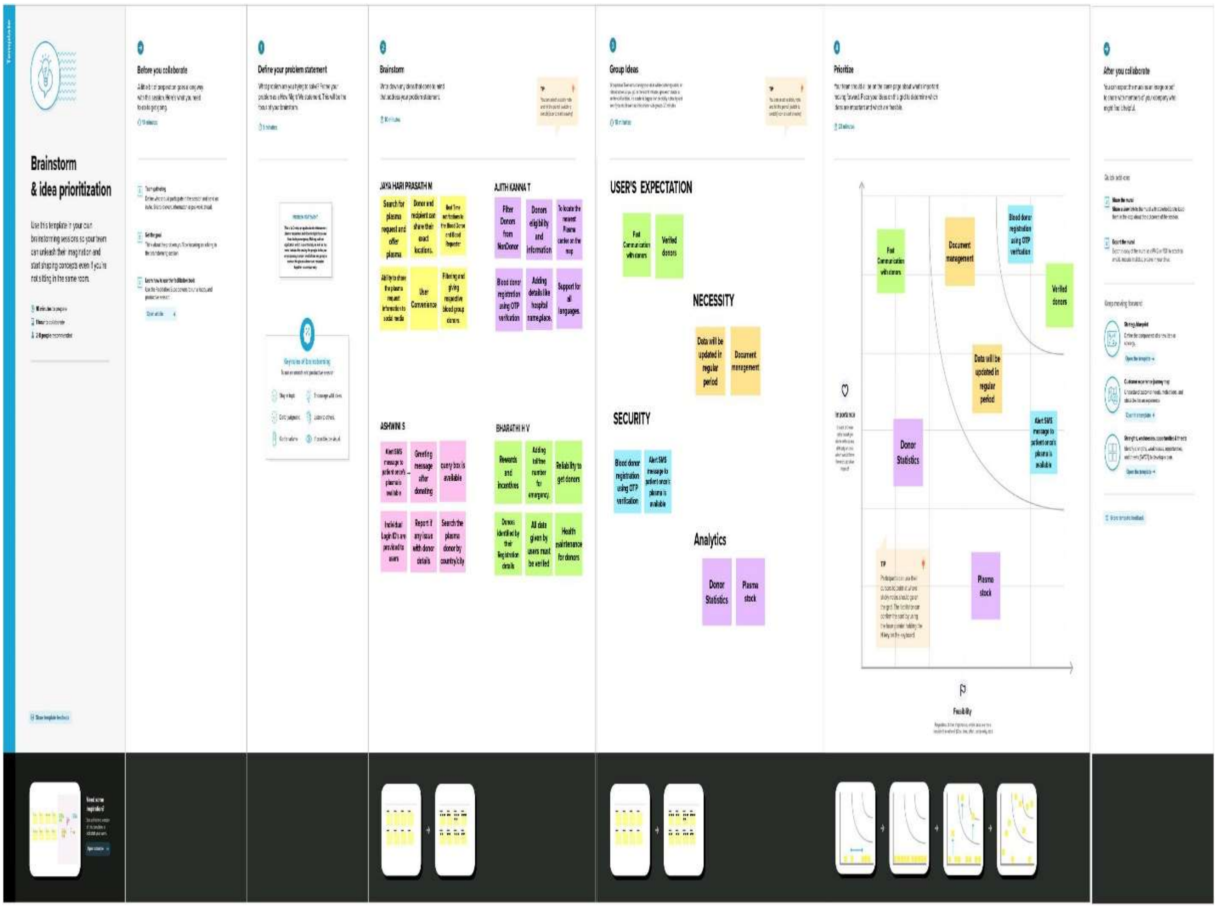
Creating a chat box that will assist user in finding answer to that question

Helpful in emergency situation

Time efficient

3.2 Ideation & Brainstorming

Brainstorming is a group problem-solving method that involves the spontaneous contribution of creative ideas and solutions. This technique requires intensive, freewheeling discussion in which every member of the group is encouraged to think aloud and suggest as many ideas as possible based on their diverse knowledge.



Brainstorm & idea prioritization
Before you collaborate
Define your problem statement
Brainstorm
Write down any ideas that come to mind that address your problem statement.
JAYA HARI PRASATH N
Search for plasma request and offer plasma
Donor and recipient can share their exact locations.
Real Time notifications to the Blood Donor and Blood Requester
Ability to share the plasma request information to social media
User Convenience
Filtering and giving respective blood group donors.
AJITH KANNA T
Filter Donors from NonDonor
Donors eligibility and information
To locate the nearest Plasma center on the map
Blood donor registration using OTP verification
Adding details like hospital name,place.
Support for all languages.
ASHWINI S
Alert SMS message to patient once's plasma is available
Greeting message after donating
query box is available
Individual Login ID's are provided to users
Report if any issue with donor details
Search the plasma donor by country/city
BHARATHI H V
Rewards and Incentives
Adding toll free number for emergency
Reliability to get donors
Donors identified by their Registration details
All data given by users must be verified
Health maintenance for donors
Group Ideas
USER'S EXPECTATION
Fast Communication with donors
Verified donors
NECESSITY
Data will be updated in regular period
Document management
SECURITY
Blood donor registration using OTP verification
Alert SMS message to patient once's plasma is available
Analytics
Donor Statistics
Plasma stock
Prioritize
Importance
Feasibility
Fast Communication with donors
Document management
Blood donor registration using OTP verification
Verified donors
Data will be updated in regular period
Donor Statistics
Alert SMS message to patient once's plasma is available
Plasma stock
TIP
After you collaborate

3.3 Proposed Solution

Solution description:

Creating a web application, which stores the donors list in a database .And connecting the donors & the patients by an online application. By using this application, the users can either raise a request for plasma donation or requirements.

- This web application always keep updating the location of the donor.
- To become a member of our system, donors need to create their profile.
- If the user wants to donate or receive they have to register with their personal details.
- After successful registration of user A successful registration email is send to the user.
- After successful registration user, will be directed to home page.
- They will be asked to press whether they will be donor or receiver.
- If the user is donor then he/she will fill the donation interest form.
- If the user is receiver then he/she can see the list of donors available and they can raise their request and contact donor directly.

Novelty:

The Donors details are stored highly secured. A User Interface is simple for users to understand. Users can submit their comments if they have any difficulties during donation Process.

Customer satisfaction:

Only the trusted donors are recommended. It can also help the people to know about the benefits of plasma donation so that their small contribution can help one person to save his/her life

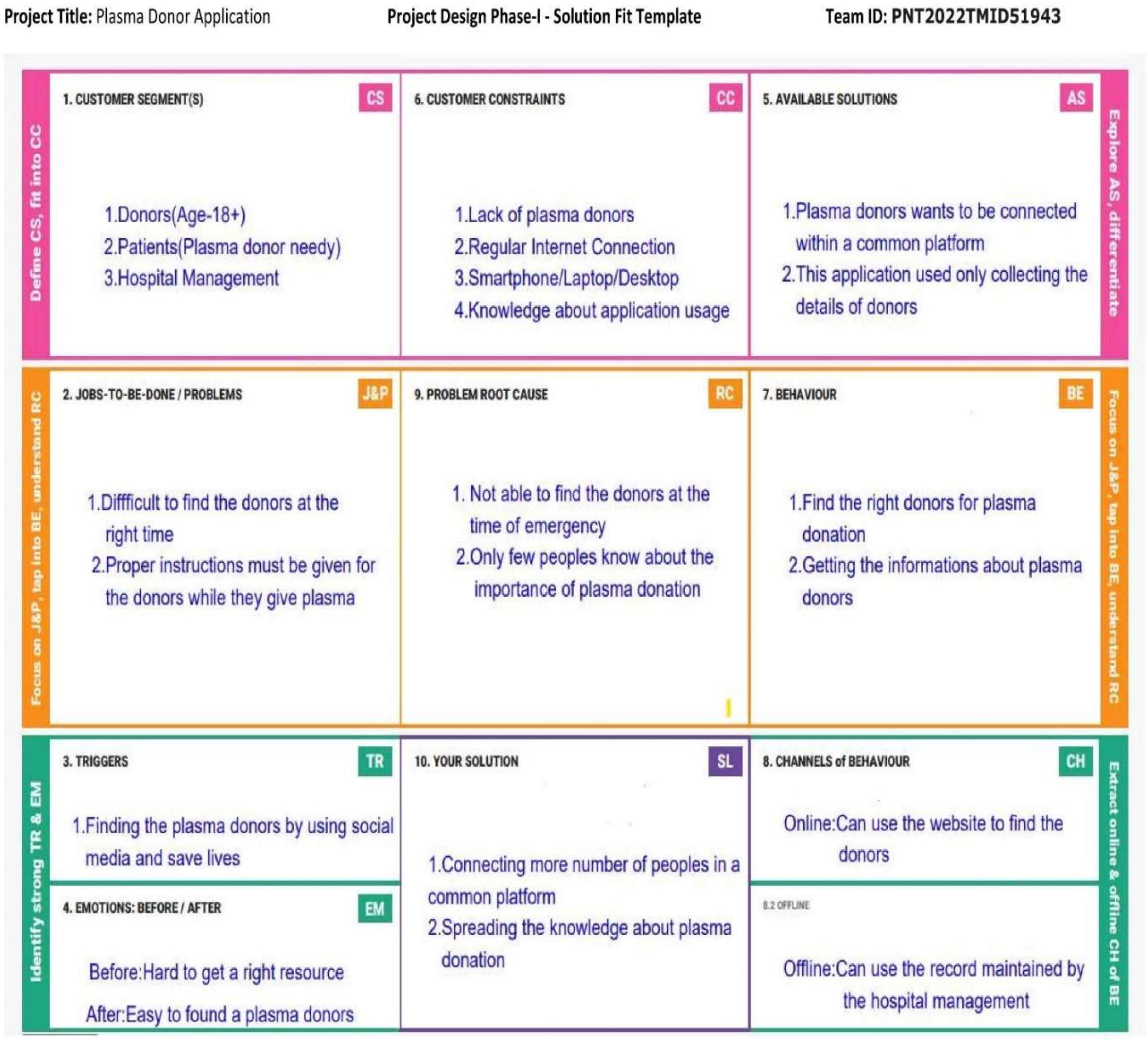
Business model:

There are many private sectors and NGOs, who organize plasma donation camps. Even collaboration with companies like Bio-life, and other pharmaceutical companies use plasma to make treatment for conditions such as immune deficiencies and bleeding disorder in order to increase revenue.

Scalability of the solution:

This application has the ability to handle more donors and provide users with good user experience. It handles the traffic, responding accurately and reacting to the growing number of requests. Chat-box facility is given. This application helps users to find plasma donors by sitting in home itself instead of searching donors everywhere

3.4 Problem Solution fit



Project Title: Plasma Donor Application **Project Design Phase-I - Solution Fit Template** **Team ID: PNT2022TMID51943**

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS
	1.Donors(Age-18+)
	2.Patients(Plasma donor needy)
	3.Hospital Management
		6. CUSTOMER CONSTRAINTS CC
		1.Lack of plasma donors
		2.Regular Internet Connection
		3.Smartphone/Laptop/Desktop
		4.Knowledge about application usage
			5. AVAILABLE SOLUTIONS AS
			1.Plasma donors wants to be connected within a common platform
			2.This application used only collecting the details of donors
				Explore AS, differentiate

Focus on J&P, tap into BE, understand RC	2. JOBS-TO-BE-DONE / PROBLEMS J&P 1.Diffficult to find the donors at the right time 2.Proper instructions must be given for the donors while they give plasma	9. PROBLEM ROOT CAUSE RC 1. Not able to find the donors at the time of emergency 2.Only few peoples know about the importance of plasma donation	7. BEHAVIOUR BE 1.Find the right donors for plasma donation 2.Getting the informations about plasma donors	Focus on J&P, tap into BE, understand RC
Identify strong TR & EM	3. TRIGGERS TR 1.Finding the plasma donors by using social media and save lives 4. EMOTIONS: BEFORE / AFTER EM Before:Hard to get a right resource After:Easy to found a plasma donors	10. YOUR SOLUTION SL 1.Connecting more number of peoples in a common platform 2.Spreading the knowledge about plasma donation	8. CHANNELS of BEHAVIOUR CH Online:Can use the website to find the donors 8.2 OFFLINE Offline:Can use the record maintained by the hospital management	Extract online & offline CH of BE

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement(Story/Sub-Task)
FR-1	User Registration	Registration through form Registration through Gmail Registration through Linkedin
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Login	Login through registration email id
FR-4	Sent Request	If plasma required then donor get the notification
FR-5	Recipient notification	Recipient get notified from Donor via message
FR-6	Confirmation	Donor information will be shared with receiver for further process.

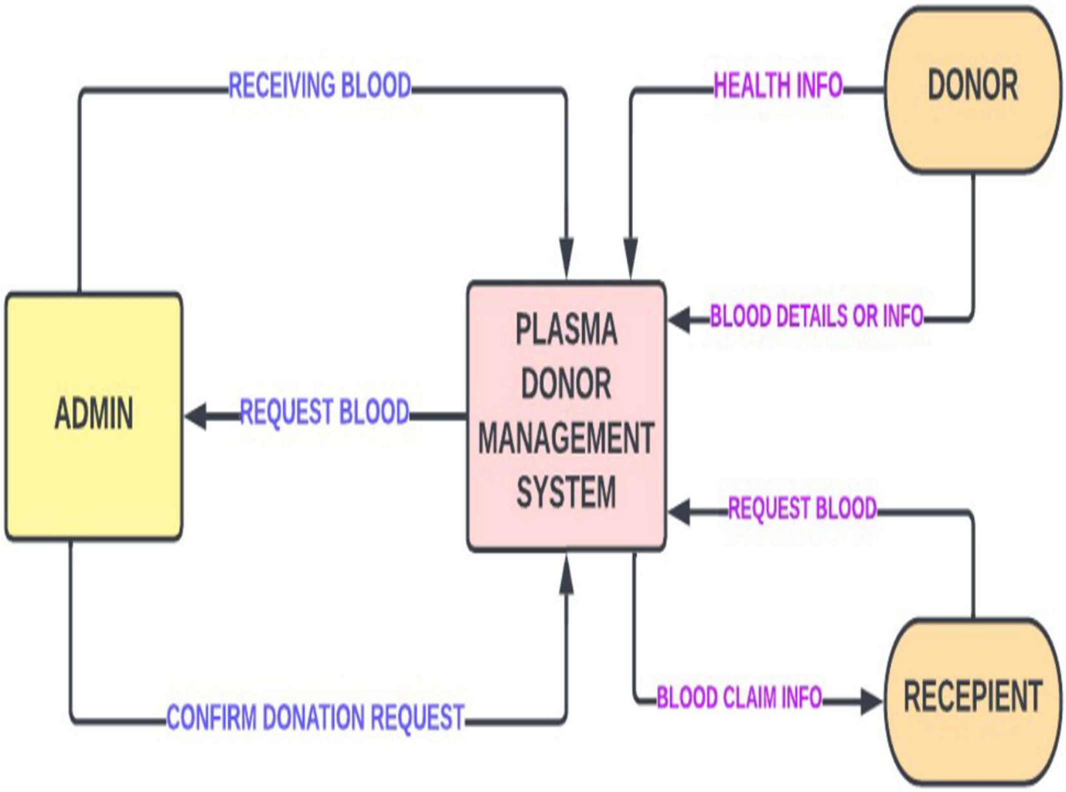
4.2. Non-Functional requirements

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The plasma Donor application is user friendly And easy to access.
NFR-2	Security	It is secured with the user email id and password
NFR-3	Reliability	Application can provide true and timely information about donors and reach receivers immediately
NFR-4	Performance	This application will help donors as well as recipients in a better manner.
NFR-5	Availability	The notification about the availability of donor was sent through SMS
NFR-6	Scalability	Major amount of donor or receiver information can be gathered and stored with a simple database structure

5. PROJECT DESIGN

5.1. Data Flow Diagram



5.2. User Stories:

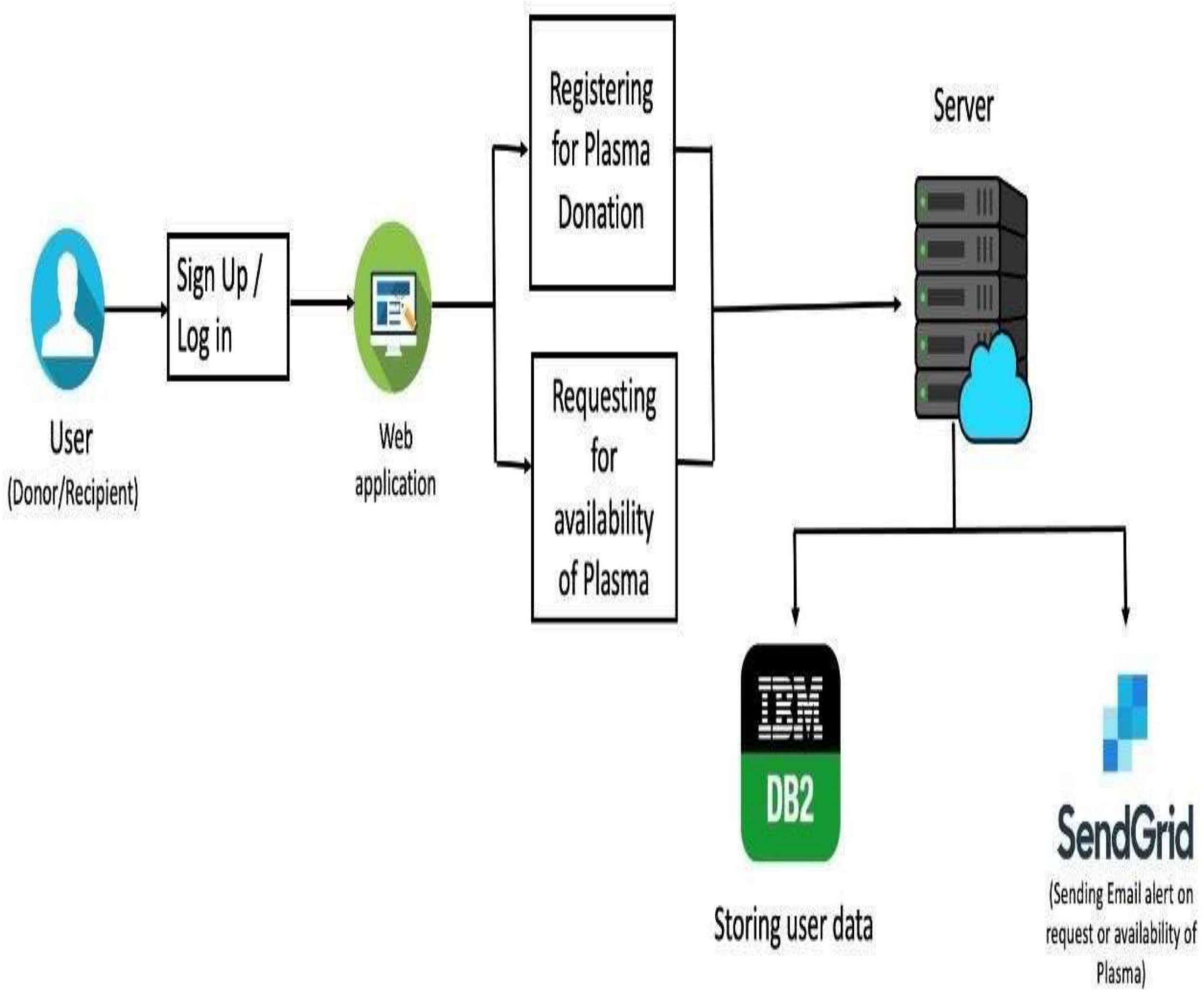
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering Mobile no, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive otp once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2

		USN-4	As a user, I can register for the application through Mobile no	I can receive confirmation otp & confirm	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering mobile no & password	I can enter into my account	High	Sprint-1
	Dashboard	USN-6	As a user ,Display all details about plasma application	I can donate/get details about the plasma	High	Sprint-2
Customer (Web user)	Application	USN-7	As a user ,I can register, login and see details about plasma	I can access the donor details and availability of plasma	High	Sprint-3

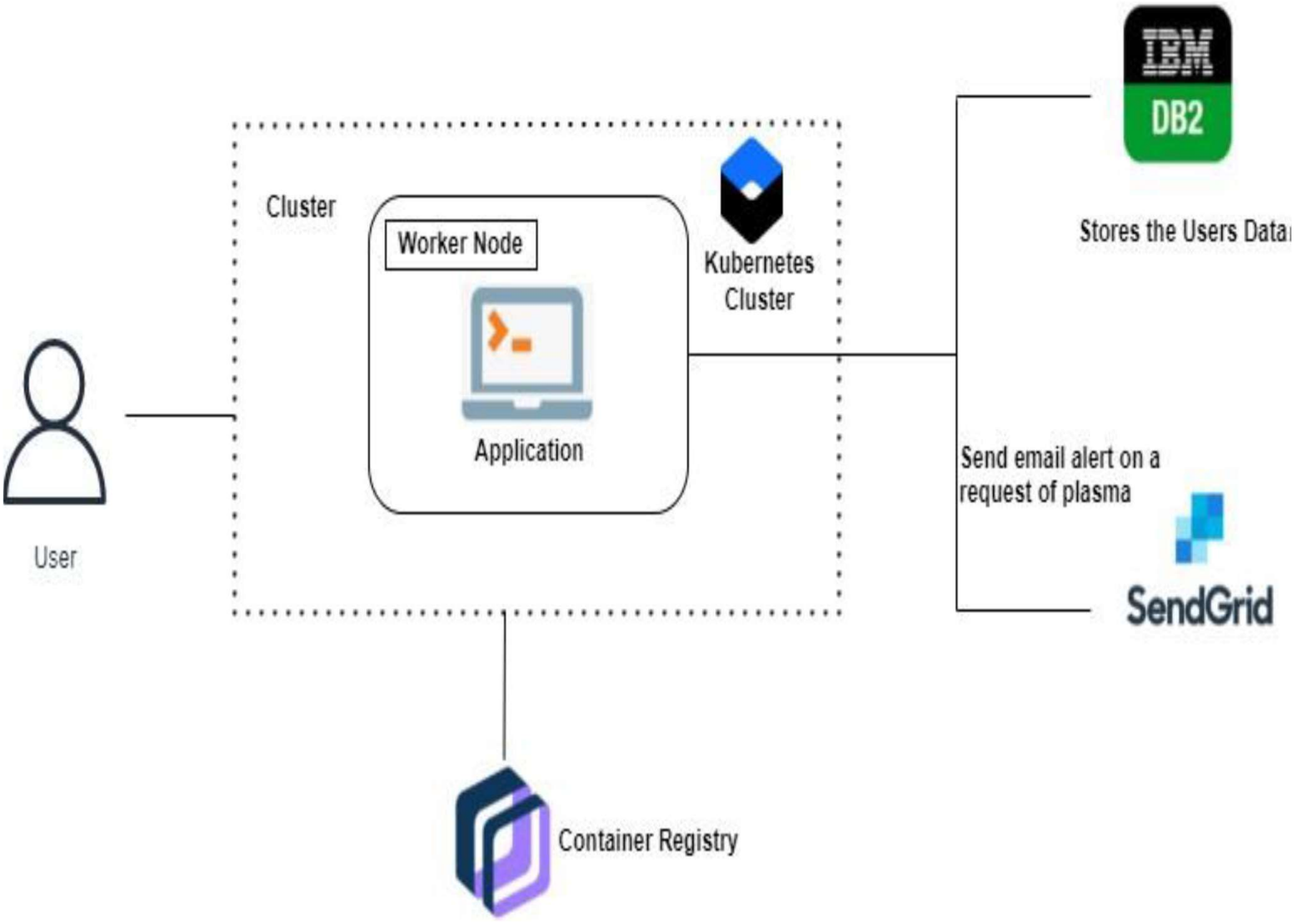
Customer Care Executive	Update Plasma storage	USN-8	Keep track the availability of the Plasma	I can provide application for customer needs	High	Sprint-4
Administrator	Verify donor details	USN-9	To add the donor plasma details in application	I can Control the all details in this application	Medium	Sprint-3
Customer Care Executive	Verify Customer Feedback	USN-10	To design the application that meets user’s desires	I can satisfy the customer expectations	Medium	Sprint-4
Customer Care Executive	Control all Plasma details	USN-11	Make sure to check the availability of plasma in application	I can alert notification through email and SMS	High	Sprint-2
Administrator	Performance of application	USN-12	To make the process more efficient	I can save time, cost by improving the Plasma	High	Sprint-4

5.3 Solution & Technical Architecture

Solution Architecture :-


User
(Donor/Recipient)
Sign Up / Log in
Web application
Registering for Plasma Donation
Requesting for availability of Plasma
Server
IBM
DB2
Storing user data
SendGrid
(Sending Email alert on request or availability of Plasma)

Technical Architecture:



6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation:

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the web app by entering my email, password,and confirming my password.	20	High	1. M.JAYA HARI PRASATH 2. ASWINI S 3. BHARATHI H V 4.AJITH KANNA T
Sprint-2	Login	USN-2	As a user, I can log into the application by entering email & password.	20	High	1. M.JAYA HARI PRASATH 2. ASWINI S 3. BHARATHI H V 4. AJITH KANNA T
Sprint-3	Donor Information	USN-3	Donors can update their personal information eg: blood group,.	20	High	1. M.JAYA HARI PRASATH 2. ASWINI S 3. BHARATHI H V 4. AJITH KANNA T
Sprint	**Functional Requirement(Epic)**	**User Story Number**	**User Story / Task**	**Story Points**	**Priority**	**Team Members**
Sprint-4	Finding thedonor	USN-4	The patient can find the donor withtheir similar blood group	20	High	1. M.JAYA HARI PRASATH 2. ASWINI S 3. BHARATHI H V 4. AJITH KANNA T

6.2 . Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story point completed (as on planned end date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

7. CODING & SOLUTIONING

CODING:

Login.html:

The login page allows a user to gain access to the application by entering their username and password.

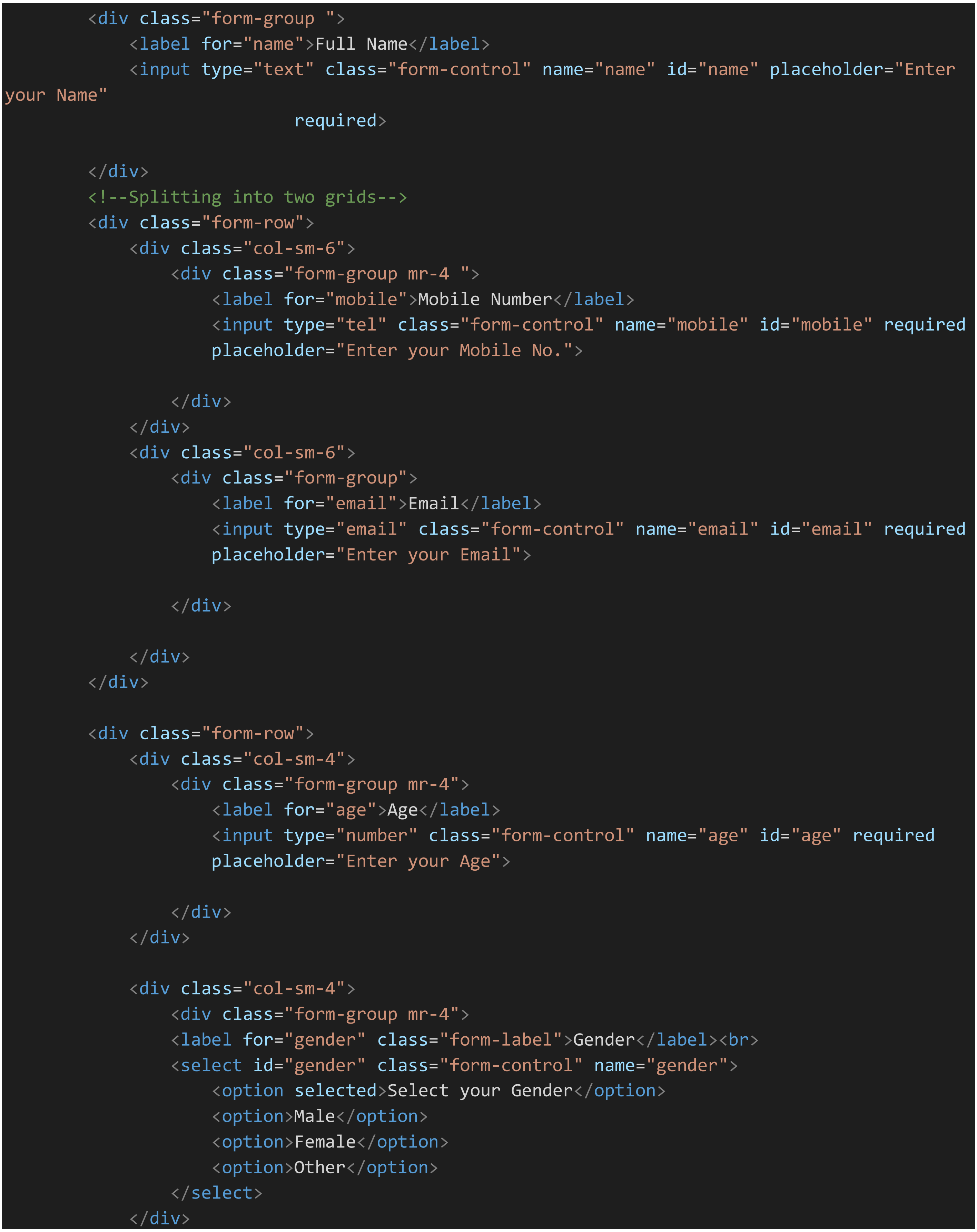
```
{% extends 'base.html'%}

<!--title tag-->
{% block title %}
<title>Admin-LogIn</title>
{% endblock %}

<!---Login Content-->
{% block content %}
<!---Registration form-->
<div class="container">
    <div class="text-center mt-5"><h2>LogIn as Admin</h2></div>

</div>

<div class="container mt-5">
    <div class="row justify-content-center">
        <div class="col-sm-6 ">
            <div class="card">
                <div class="card-body">
                    <!----Form content---->
                    <form action="/" method="post">

                        <div class="form-group">
                            <label for="email">Email</label>
                            <input type="email" class="form-control" name="" id="email"
required placeholder="Enter your Email">
                        </div>
                        <div class="form-group">
                            <label for="password">Password</label>
                            <input type="password" class="form-control" name="" id="password"
placeholder="Enter Password" required>
```

```
                        </div>






                        <!--button-->
                        <div class="form-group text-center">

                            <button type="submit" class="btn btn-success">Submit</button>
                        </div>


                    </form>
                </div>
            </div>
        </div>
    </div>
</div>

{% endblock %}
```

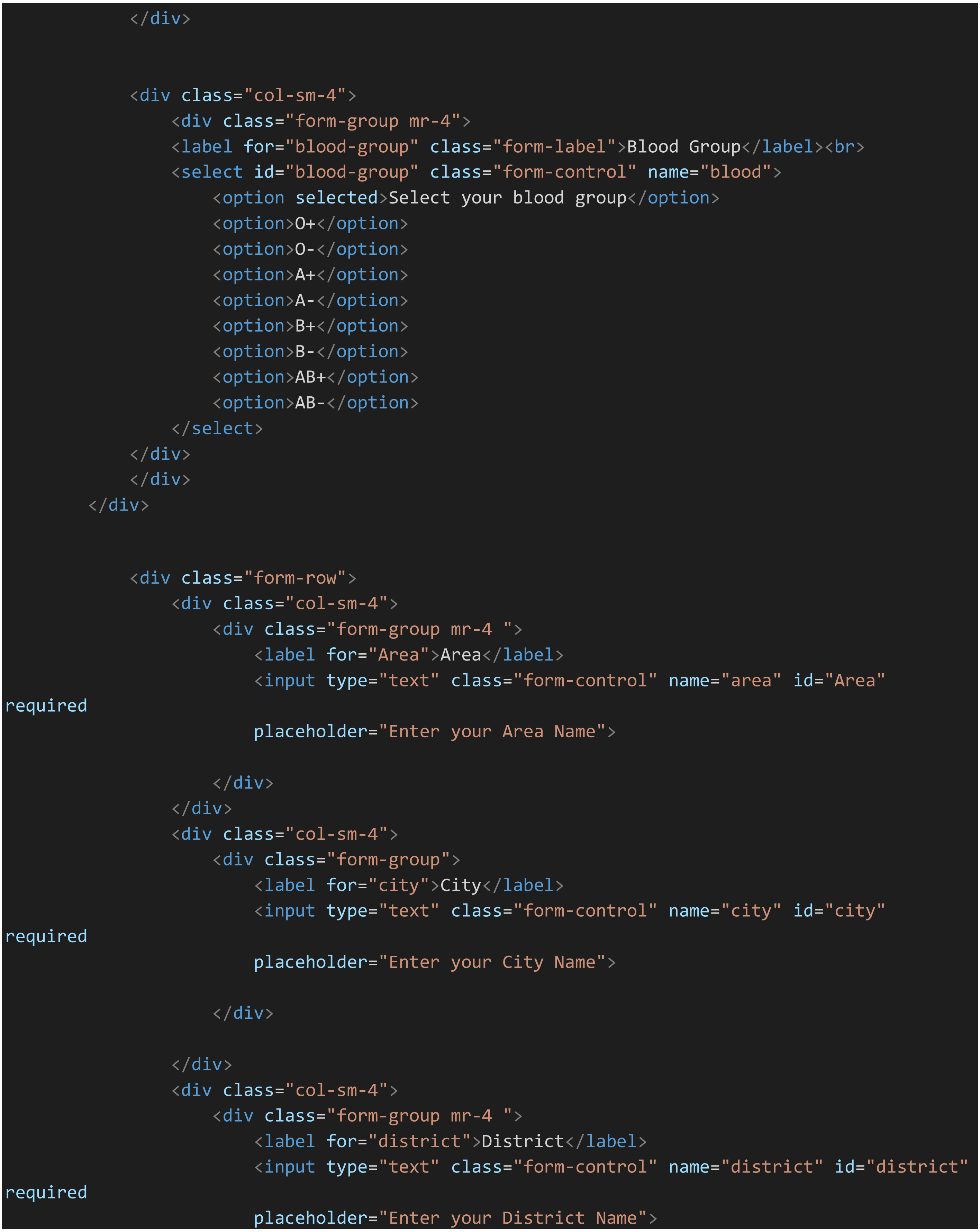
Register.html :

- ✓ Registration provides greater protection for the public.
- ✓ Every time after the initial registration, the user will log on to the system using the username and password you created

```
{% extends 'base.html' %}

<!--title tag-->
{% block title %}
<title>Plasma-Register</title>
{% endblock %}

{% block content %}

<!---Registration form-->

<div class="container mt-5<div class="row"<div class="col-sm-12"><div class="card">
<div class="card-body">
        <form action="/adddonor" method="post">
        <h4 style="text-align: center;">Donating Plasma</h4>
```

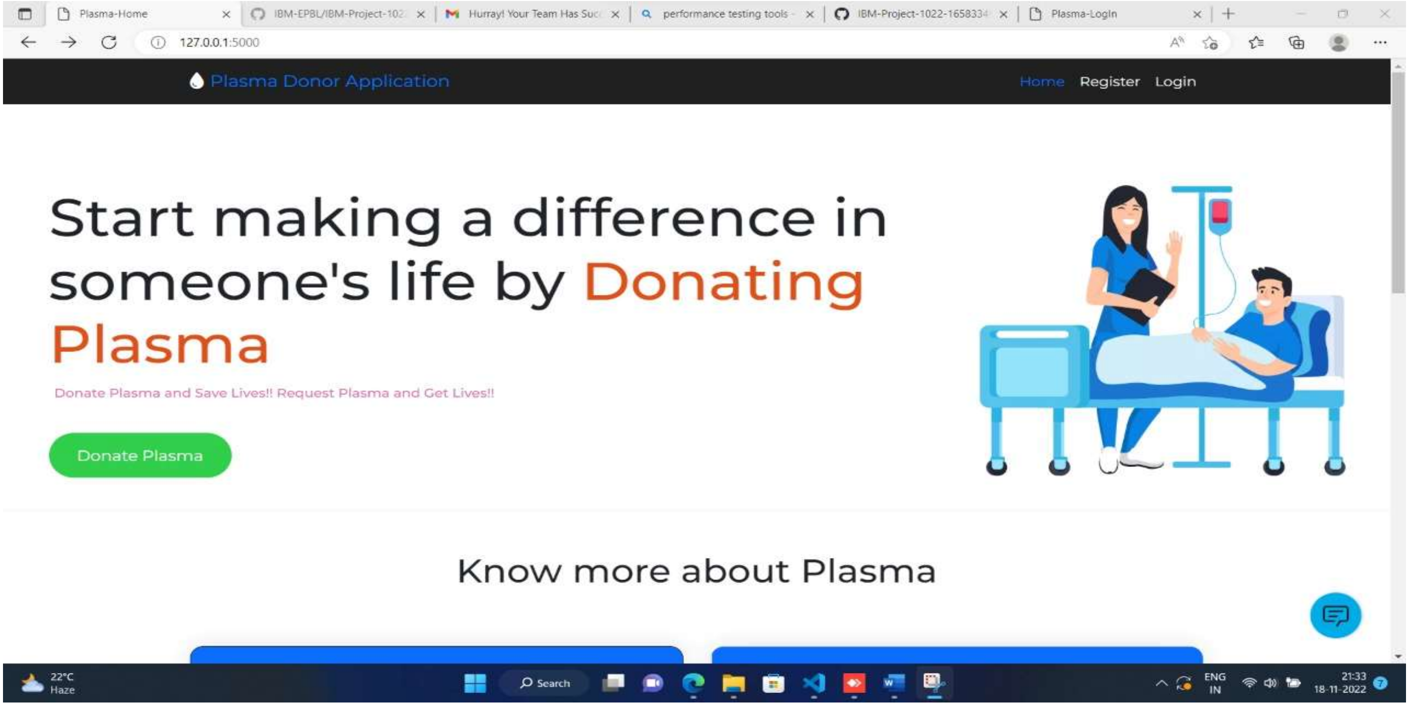


```
        <div class="form-group ">
            <label for="name">Full Name</label>
            <input type="text" class="form-control" name="name" id="name" placeholder="Enter
your Name"
                            required>

        </div>
        <!--Splitting into two grids-->
        <div class="form-row">
            <div class="col-sm-6">
                <div class="form-group mr-4 ">
                    <label for="mobile">Mobile Number</label>
                    <input type="tel" class="form-control" name="mobile" id="mobile" required
                    placeholder="Enter your Mobile No.">

                </div>
            </div>
            <div class="col-sm-6">
                <div class="form-group">
                    <label for="email">Email</label>
                    <input type="email" class="form-control" name="email" id="email" required
                    placeholder="Enter your Email">

                </div>

            </div>
        </div>

        <div class="form-row">
            <div class="col-sm-4">
                <div class="form-group mr-4">
                    <label for="age">Age</label>
                    <input type="number" class="form-control" name="age" id="age" required
                    placeholder="Enter your Age">

                </div>
            </div>

            <div class="col-sm-4">
                <div class="form-group mr-4">
                <label for="gender" class="form-label">Gender</label><br>
                <select id="gender" class="form-control" name="gender">
                    <option selected>Select your Gender</option>
                    <option>Male</option>
                    <option>Female</option>
                    <option>Other</option>
                </select>
            </div>
```



```
            </div>


            <div class="col-sm-4">
                <div class="form-group mr-4">
                <label for="blood-group" class="form-label">Blood Group</label><br>
                <select id="blood-group" class="form-control" name="blood">
                    <option selected>Select your blood group</option>
                    <option>O+</option>
                    <option>O-</option>
                    <option>A+</option>
                    <option>A-</option>
                    <option>B+</option>
                    <option>B-</option>
                    <option>AB+</option>
                    <option>AB-</option>
                </select>
            </div>
            </div>
        </div>


            <div class="form-row">
                <div class="col-sm-4">
                    <div class="form-group mr-4 ">
                        <label for="Area">Area</label>
                        <input type="text" class="form-control" name="area" id="Area"
required
                        placeholder="Enter your Area Name">

                    </div>
                </div>
                <div class="col-sm-4">
                    <div class="form-group">
                        <label for="city">City</label>
                        <input type="text" class="form-control" name="city" id="city"
required
                        placeholder="Enter your City Name">

                    </div>

                </div>
                <div class="col-sm-4">
                    <div class="form-group mr-4 ">
                        <label for="district">District</label>
                        <input type="text" class="form-control" name="district" id="district"
required
                        placeholder="Enter your District Name">
                    </div>
                </div>
            </div>


                        <!--button-->
                <div class="form-group text-center">
                    <input type="reset" value="Reset" class="btn btn-dark mr-2">
                    <input type="submit" value="Submit" class="btn btn-success">

                </div>

            </form>
        </div>
    </div>
</div>
</div>
</div>




{% endblock %}
```

7.1. Feature 1

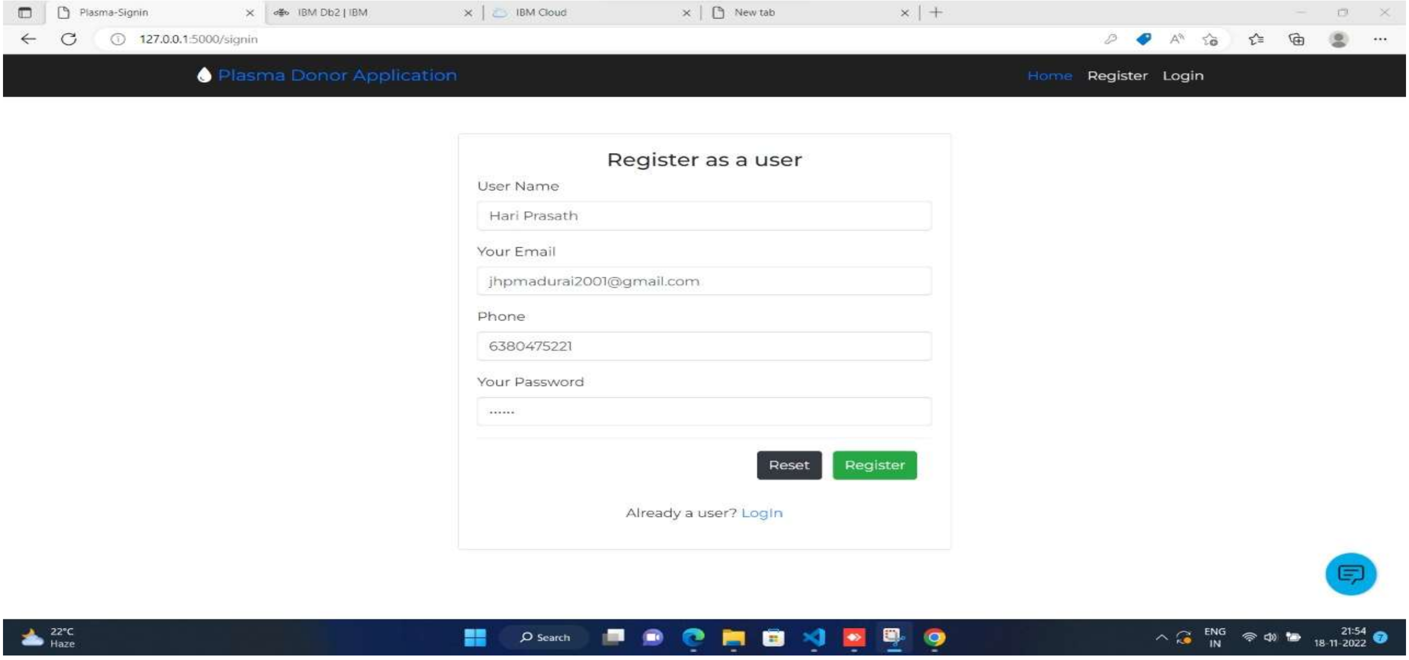


Start making a difference in someone's life by Donating Plasma

Donate Plasma and Save Lives!! Request Plasma and Get Lives!!

Know more about Plasma

7.2 Feature 2



Register as a user

User Name: Hari Prasath

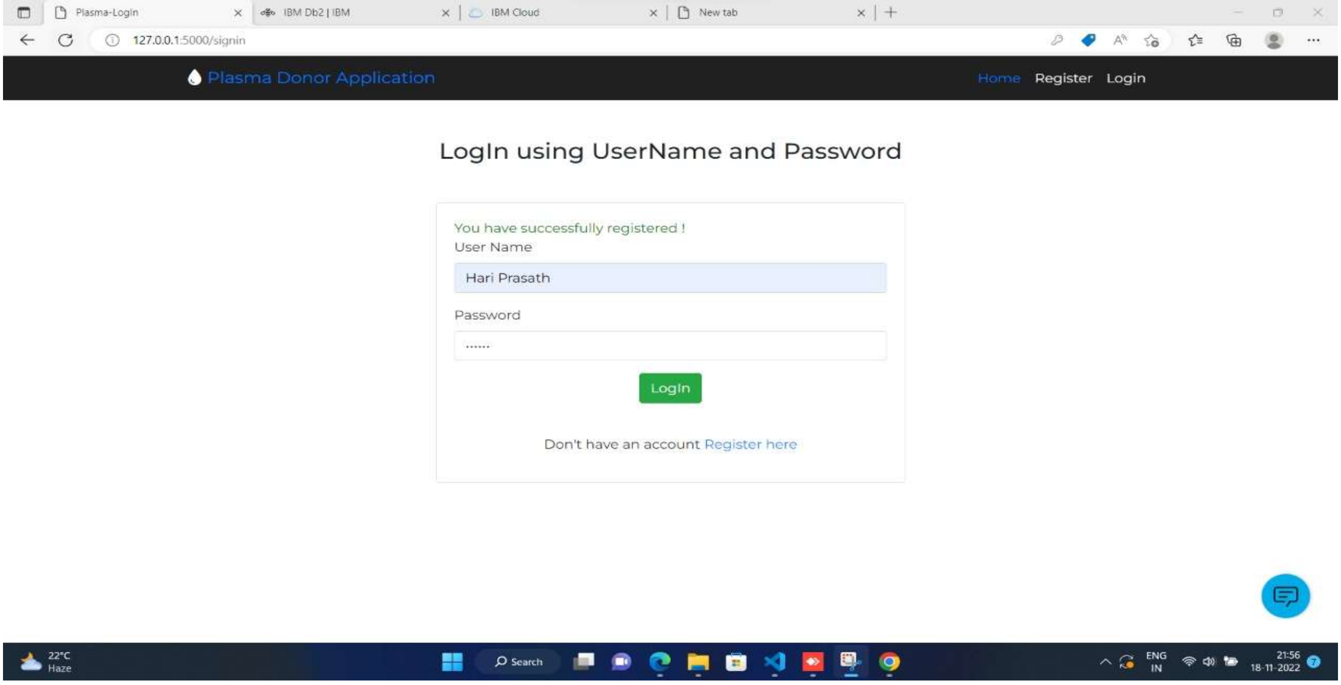
Your Email: jhpmadurai2001@gmail.com

Phone: 6380475221

Your Password: ......

Already a user? LogIn

7.3 . Feature 3



LogIn using UserName and Password

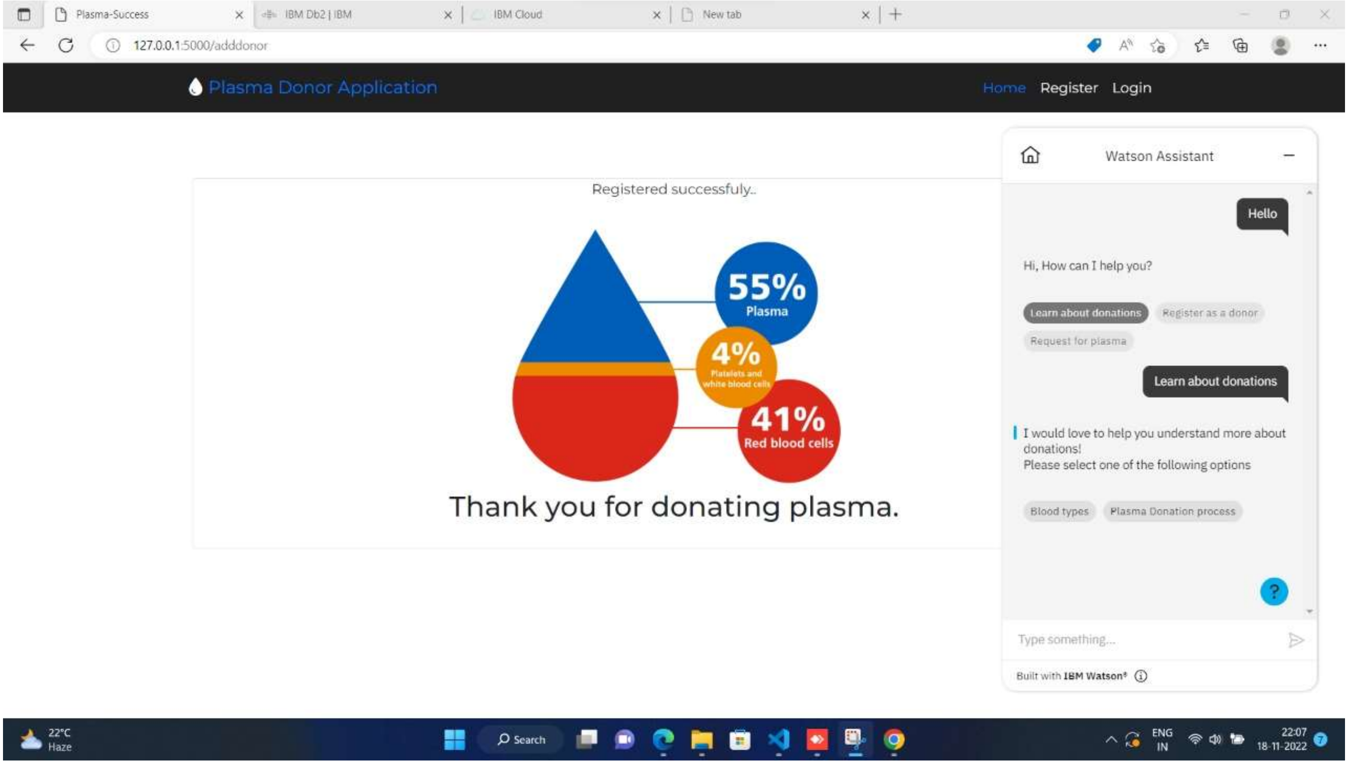
You have successfully registered !

User Name: Hari Prasath

Password: ......

Don't have an account Register here

7.4. Feature 4



Registered successfuly..

Thank you for donating plasma.

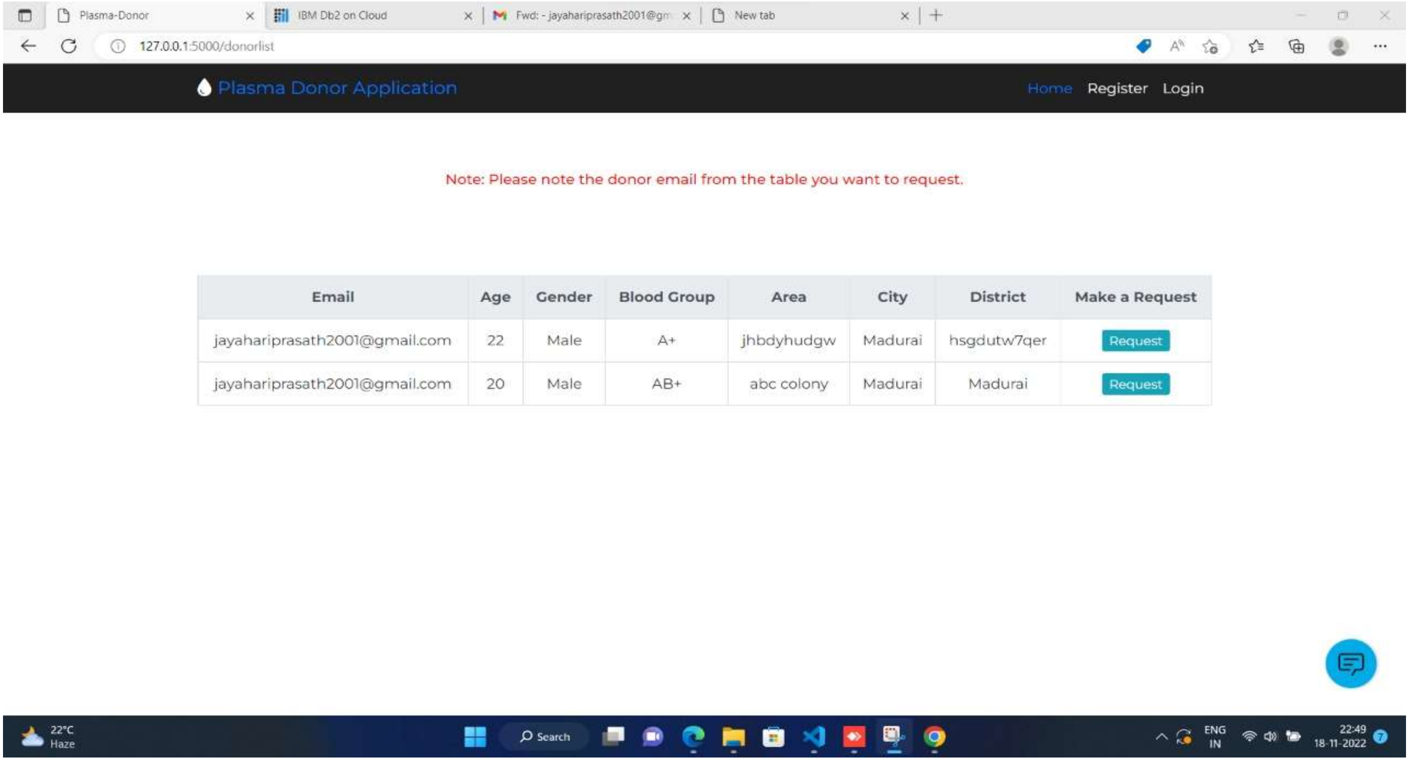
Hello

Hi, How can I help you?

Learn about donations

I would love to help you understand more about donations!
Please select one of the following options

7.5. Feature 5

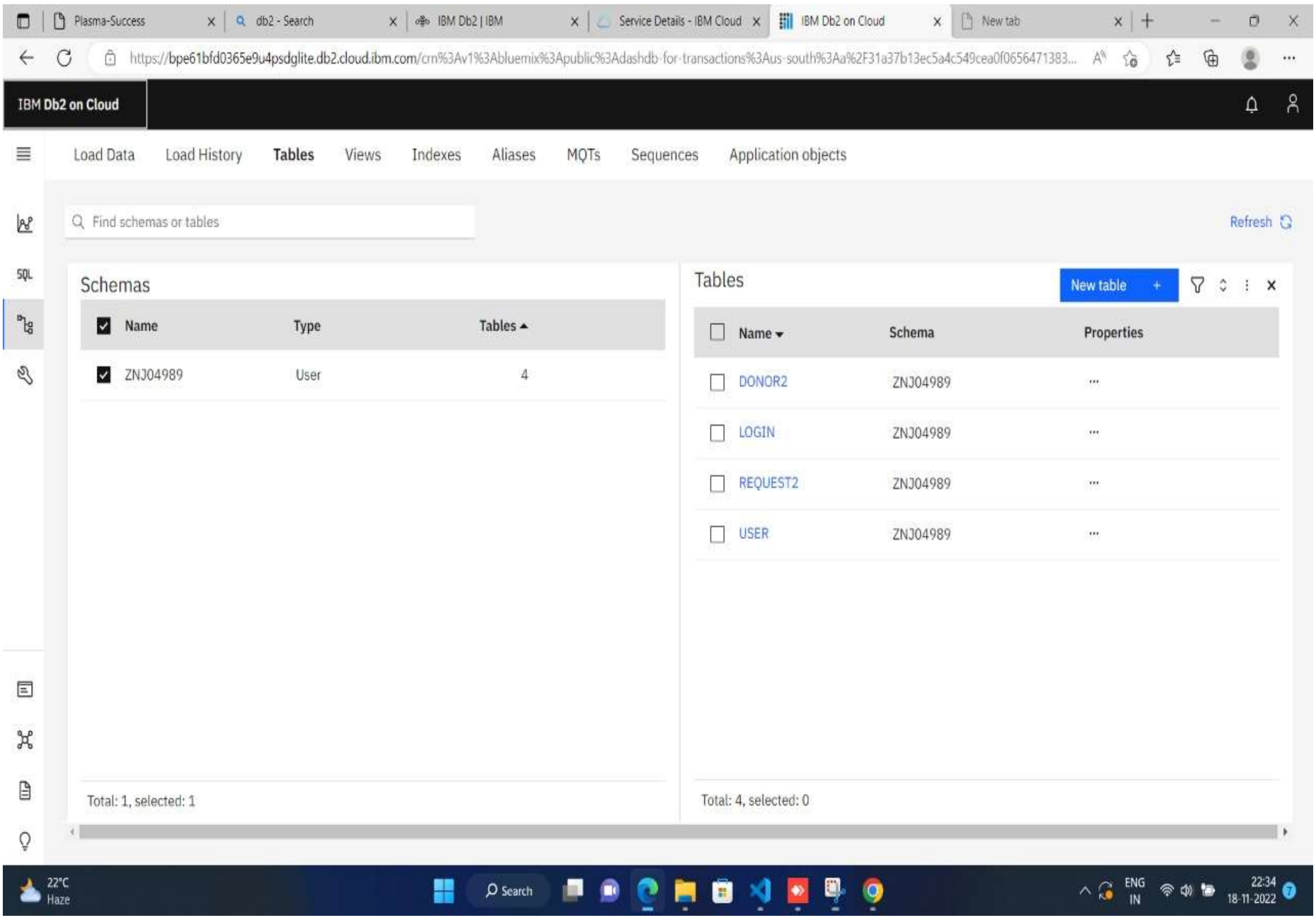


Note: Please note the donor email from the table you want to request.

Email	Age	Gender	Blood Group	Area	City	District	Make a Request
jayahariprasath2001@gmail.com	22	Male	A+	jhbdyhudgw	Madurai	hsgdutw7qer	Request
jayahariprasath2001@gmail.com	20	Male	AB+	abc colony	Madurai	Madurai	Request

7.6. Database Schema

A database schema defines how data is organized within a relational database; Schemas commonly use visual representations to communicate the architecture of the Database becoming the foundation for an organization’s data management discipline. This is inclusive of logical constraints such as, table names, fields, data types, and the relationships between these entities. This process of database schema design is also known as data modeling. These data models serve a variety of roles, such as database users, database administrators, and programmers. A database schema is considered the “blueprint” of a database which describes how the data may relate to other tables or other data models.



Schemas

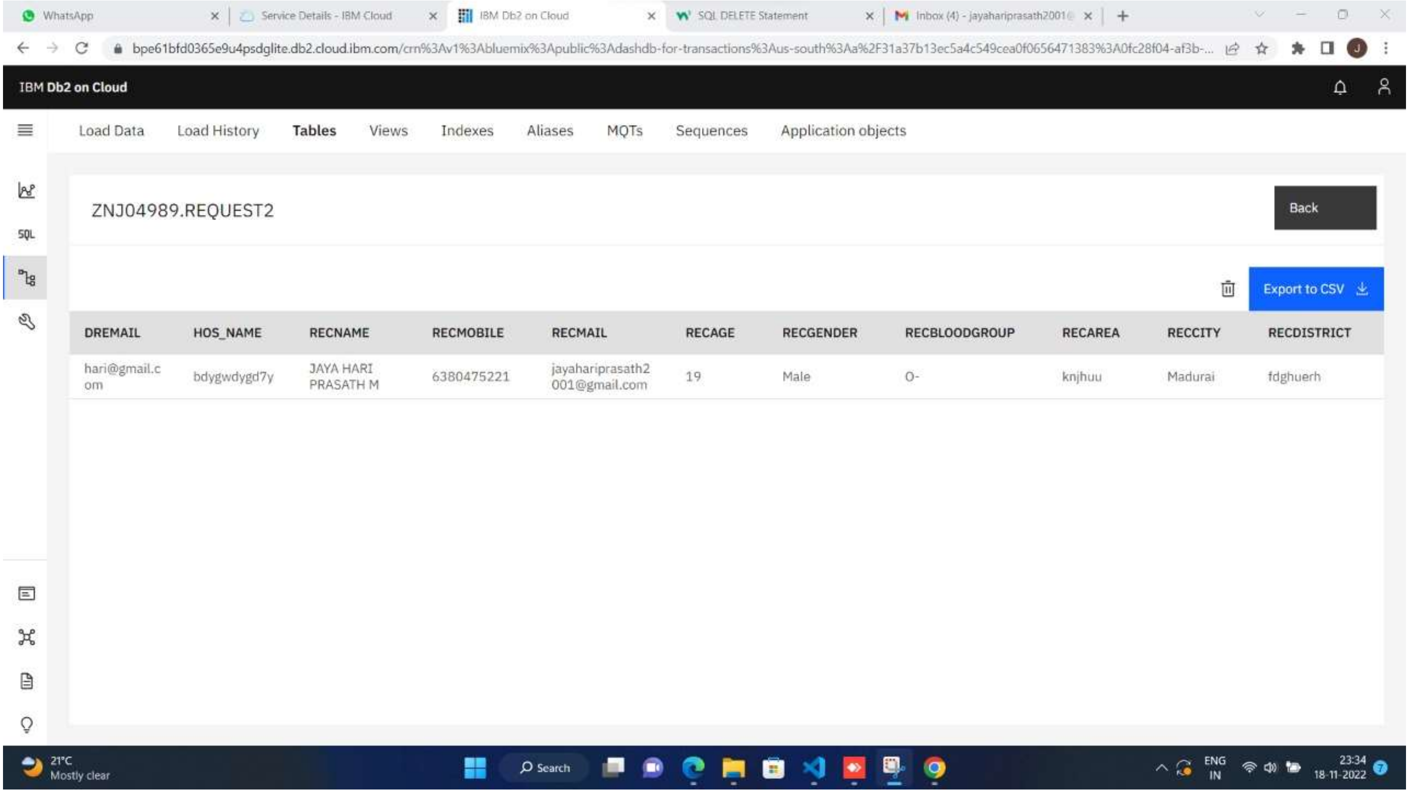
☑ Name	Type	Tables ▲
☑ ZNJ04989	User	4

Total: 1, selected: 1

Tables

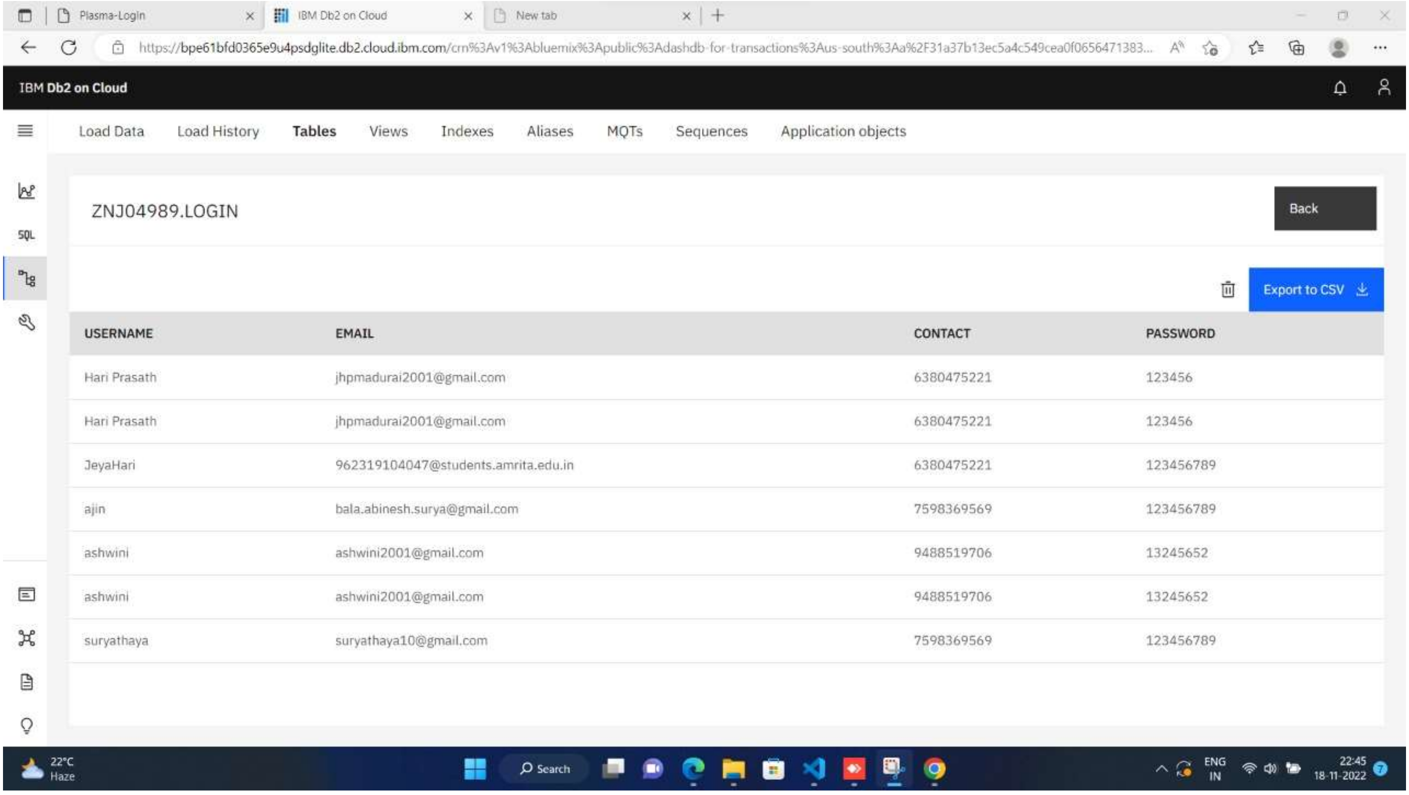
□ Name ▾	Schema	Properties
□ DONOR2	ZNJ04989	...
□ LOGIN	ZNJ04989	...
□ REQUEST2	ZNJ04989	...
□ USER	ZNJ04989	...

Total: 4, selected: 0



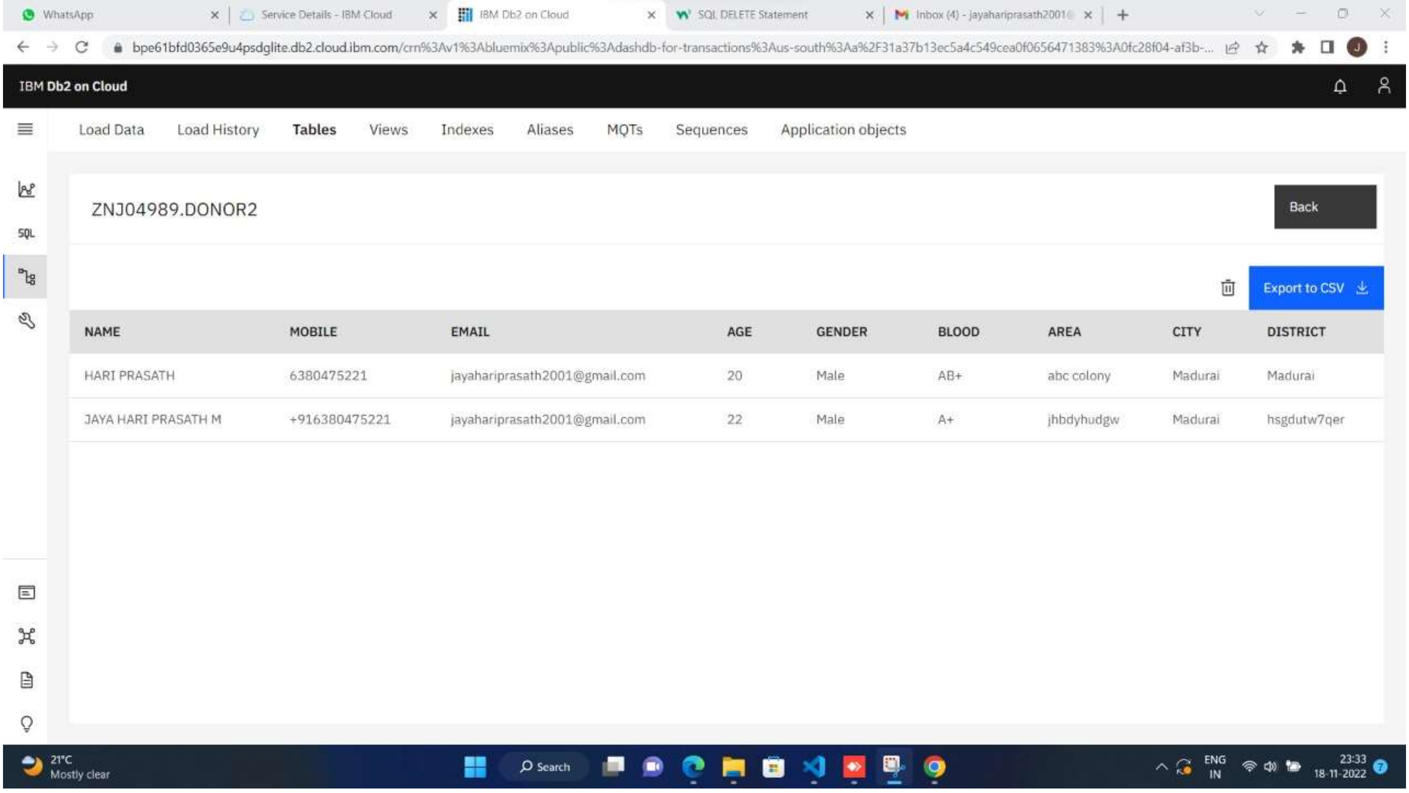
ZNJ04989.REQUEST2

DREMAIL	HOS_NAME	RECNAME	RECMOBILE	RECMAIL	RECAGE	RECGENDER	RECBLOODGROUP	RECAREA	RECCITY	RECDISTRICT
hari@gmail.com	bdygwdygd7y	JAYA HARI PRASATH M	6380475221	jayahariprasath2001@gmail.com	19	Male	O-	knjhuu	Madurai	fdghuerh



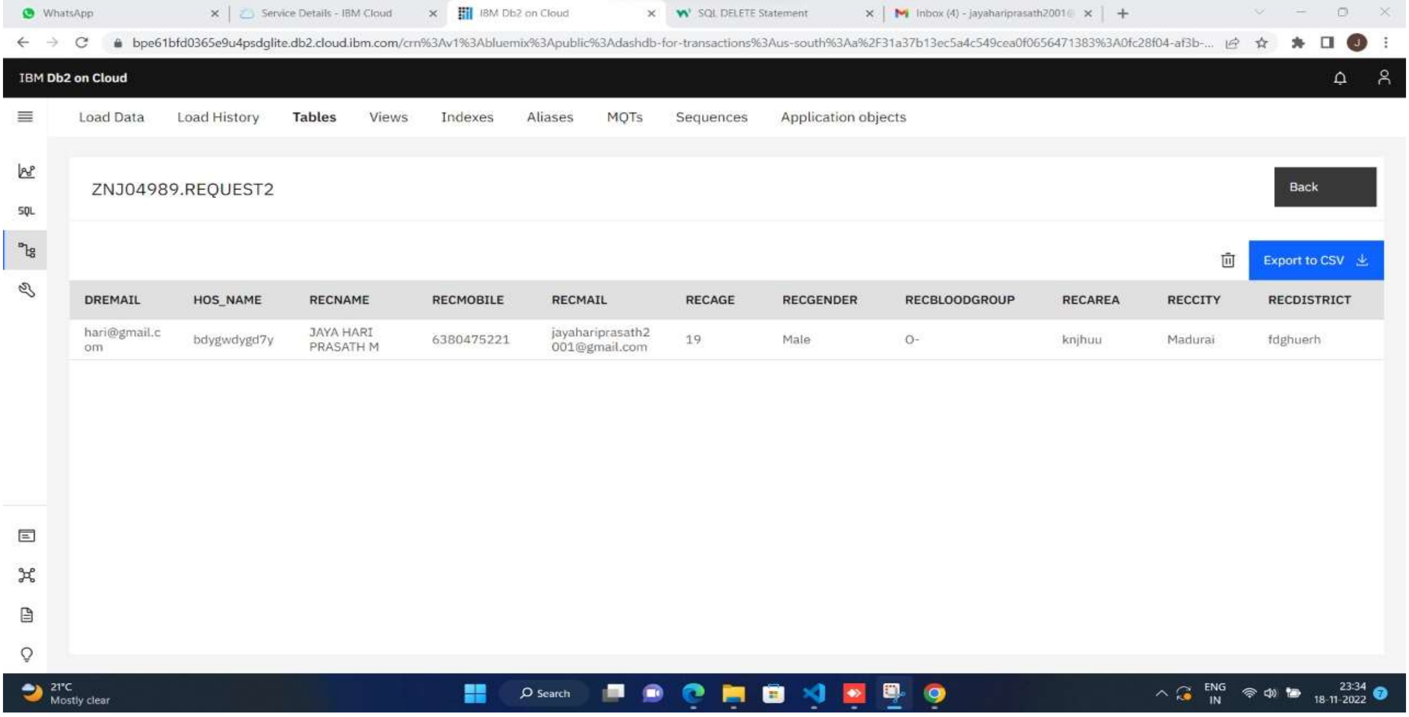
ZNJ04989.LOGIN

USERNAME	EMAIL	CONTACT	PASSWORD
Hari Prasath	jhpmadurai2001@gmail.com	6380475221	123456
Hari Prasath	jhpmadurai2001@gmail.com	6380475221	123456
JeyaHari	962319104047@students.amrita.edu.in	6380475221	123456789
ajin	bala.abinesh.surya@gmail.com	7598369569	123456789
ashwini	ashwini2001@gmail.com	9488519706	13245652
ashwini	ashwini2001@gmail.com	9488519706	13245652
suryathaya	suryathaya10@gmail.com	7598369569	123456789


WhatsApp
Service Details - IBM Cloud
IBM Db2 on Cloud
SQL DELETE Statement
Inbox (4) - jayahariprasath2001@
bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2F31a37b13ec5a4c549cea0f0656471383%3A0fc28f04-af3b-...
IBM Db2 on Cloud
Load Data
Load History
Tables
Views
Indexes
Aliases
MQTs
Sequences
Application objects
SQL
Back
Export to CSV
21°C
Mostly clear
Search
ENG
IN
23:33
18-11-2022

ZNJ04989.DONOR2

NAME	MOBILE	EMAIL	AGE	GENDER	BLOOD	AREA	CITY	DISTRICT
HARI PRASATH	6380475221	jayahariprasath2001@gmail.com	20	Male	AB+	abc colony	Madurai	Madurai
JAYA HARI PRASATH M	+916380475221	jayahariprasath2001@gmail.com	22	Male	A+	jhbdyhudgw	Madurai	hsgdutw7qer

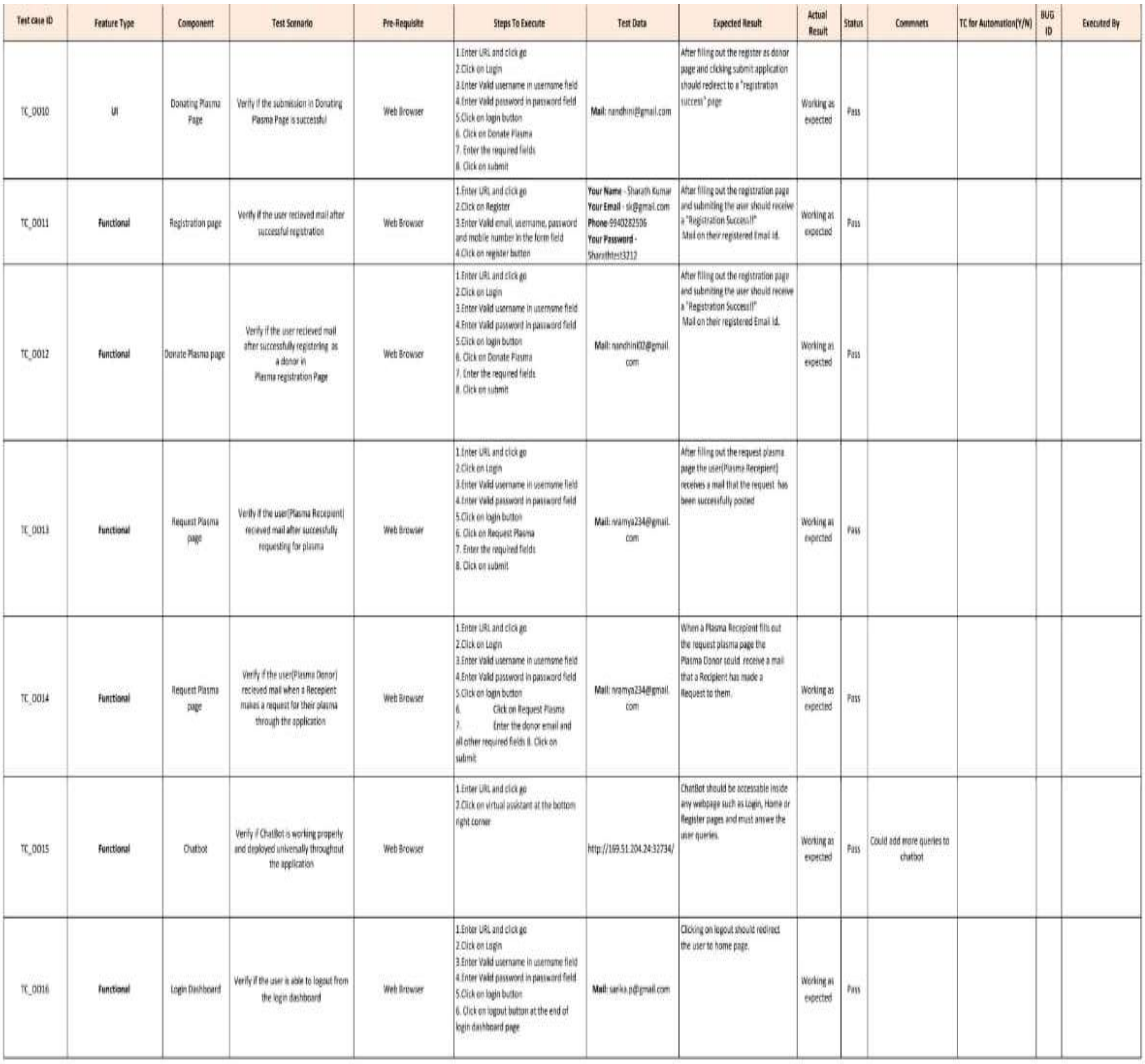

WhatsApp
Service Details - IBM Cloud
IBM Db2 on Cloud
SQL DELETE Statement
Inbox (4) - jayahariprasath2001@
bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2F31a37b13ec5a4c549cea0f0656471383%3A0fc28f04-af3b-...
IBM Db2 on Cloud
Load Data
Load History
Tables
Views
Indexes
Aliases
MQTs
Sequences
Application objects
SQL
Back
Export to CSV
21°C
Mostly clear
Search
ENG
IN
23:34
18-11-2022

ZNJ04989.REQUEST2

DREMAIL	HOS_NAME	RECNAME	RECMOBILE	RECMAIL	RECAGE	RECGENDER	RECBLOODGROUP	RECAREA	RECCITY	RECDISTRICT
hari@gmail.com	bdygwdygd7y	JAYA HARI PRASATH M	6380475221	jayahariprasath2001@gmail.com	19	Male	O-	knjhuu	Madurai	fdghuerh

8. TESTING

8.1 Test Cases

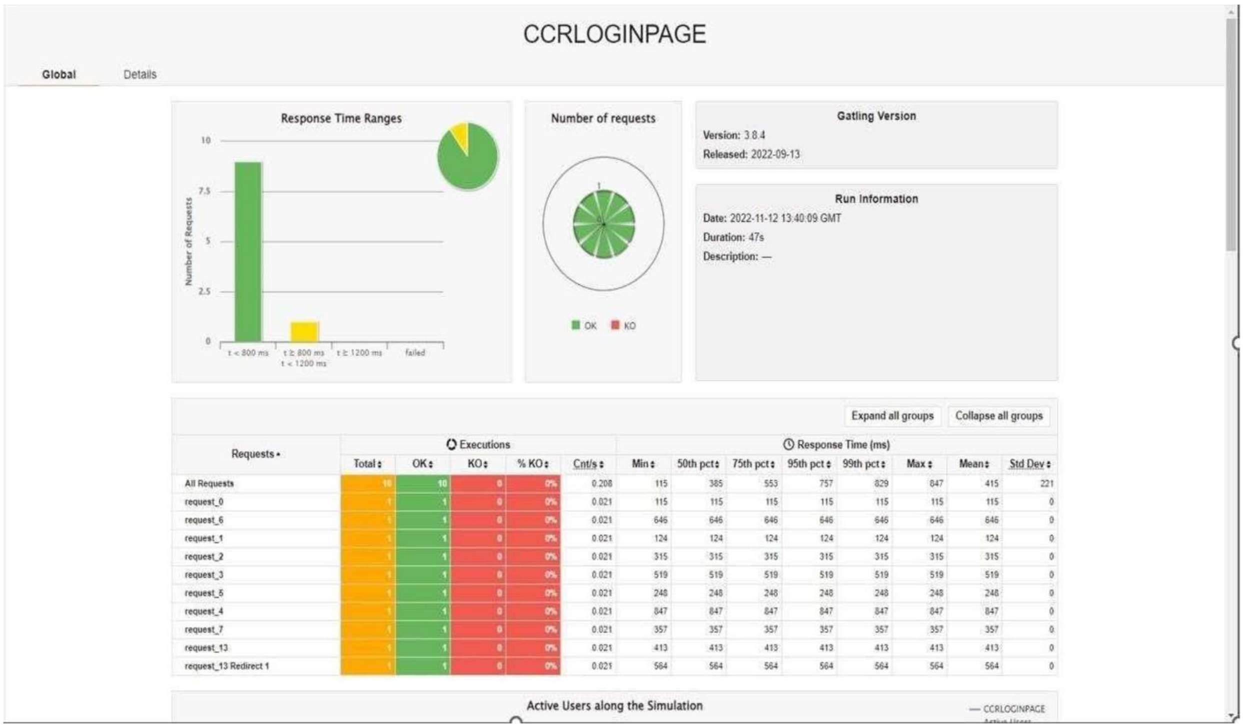


Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
TC_OO10	UI	Donating Plasma Page	Verify if the submission in Donating Plasma Page is successful	Web Browser	1.Enter URL and click go 2.Click on Login 3.Enter Valid username in username field 4.Enter Valid password in password field 5.Click on login button 6. Click on Donate Plasma 7. Enter the required fields 8. Click on submit	**Mail:** nandhini@gmail.com	After filing out the register as donor page and clicking submit application should redirect to a "registration success" page	Working as expected	Pass				
TC_OO11	**Functional**	Registration page	Verify if the user recieved mail after successful registration	Web Browser	1.Enter URL and click go 2.Click on Register 3.Enter Valid email, username, password and mobile number in the form field 4.Click on register button	**Your Name** - Sharath Kumar **Your Email** - sk@gmail.com **Phone**-9940282506 **Your Password** - Sharathtest3212	After filling out the registration page and submiting the user should receive a "Registration Success!!" Mail on their registered Email id.	Working as expected	Pass				
TC_OO12	**Functional**	Donate Plasma page	Verify if the user recieved mail after successfully registering as a donor in Plasma registration Page	Web Browser	1.Enter URL and click go 2.Click on Login 3.Enter Valid username in usernsme field 4.Enter Valid password in password field 5.Click on login button 6. Click on Donate Plasma 7. Enter the required fields 8. Click on submit	**Mail:** nandhini02@gmail.com	After filling out the registration page and submiting the user should receive a "Registration Success!!" Mail on their registered Email id.	Working as expected	Pass				
TC_OO13	**Functional**	Request Plasma page	Verify if the user(Plasma Recepient) recieved mail after successfully requesting for plasma	Web Browser	1.Enter URL and click go 2.Click on Login 3.Enter Valid username in usernsme field 4.Enter Valid password in password field 5.Click on login button 6. Click on Request Plasma 7. Enter the required fields 8. Click on submit	**Mail:** nramya234@gmail.com	After filling out the request plasma page the user(Plasma Recepient) receives a mail that the request has been successfully posted	Working as expected	Pass				
TC_OO14	**Functional**	Request Plasma page	Verify if the user(Plasma Donor) recieved mail when a Recepient makes a request for their plasma through the application	Web Browser	1.Enter URL and click go 2.Click on Login 3.Enter Valid username in usernsme field 4.Enter Valid password in password field 5.Click on login button 6. Click on Request Plasma 7. Enter the donor email and all other required fields 8. Click on submit	**Mail:** nramya234@gmail.com	When a Plasma Recepient fills out the request plasma page the Plasma Donor sould receive a mail that a Recipient has made a Request to them.	Working as expected	Pass				
TC_OO15	**Functional**	Chatbot	Verify if ChatBot is working properly and deployed universally throughout the application	Web Browser	1.Enter URL and click go 2.Click on virtual assistant at the bottom right corner	http://169.51.204.24:32734/	ChatBot should be accessable inside any webpage such as Login, Home or Register pages and must answe the user queries.	Working as expected	Pass	Could add more queries to chatbot			
TC_OO16	**Functional**	Login Dashboard	Verify if the user is able to logout from the login dashboard	Web Browser	1.Enter URL and click go 2.Click on Login 3.Enter Valid username in username field 4.Enter Valid password in password field 5.Click on login button 6. Click on logout button at the end of login dashboard page	**Mail:** sarika.p@gmail.com	Clicking on logout should redirect the user to home page.	Working as expected	Pass				

8.2. Performance Testing

Login page :

- ✓ The login page allows a user to gain access to an application by entering their username and password or by authenticating using a social media login.
- ✓ Advantages of Social Logins
  - o Improved user experience
  - o Mobile-friendly
  - o Increased data collection. .



CCRLOGINPAGE

Gatling Version

Version: 3.8.4

Released: 2022-09-13

Run Information

Date: 2022-11-12 13:40:09 GMT

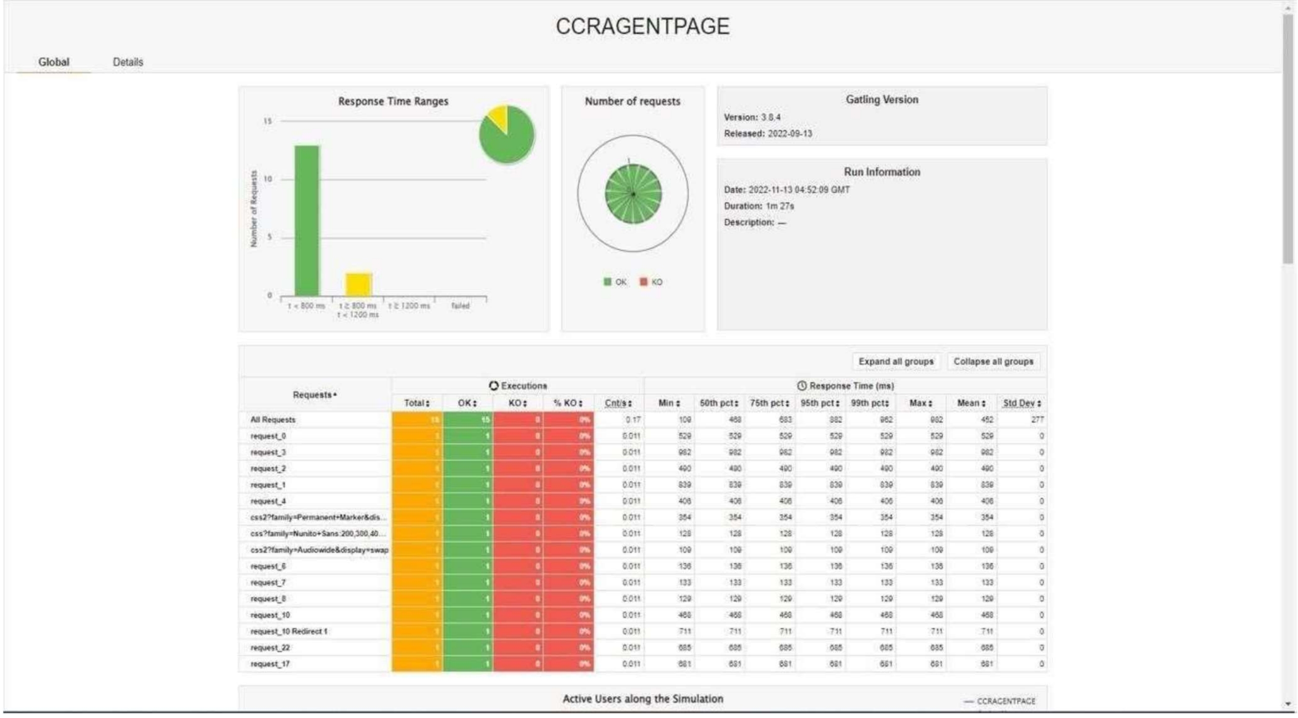
Duration: 47s

Description: —

Requests ▴	Executions					Response Time (ms)							
	Total ⇕	OK ⇕	KO ⇕	% KO ⇕	Cnt/s ⇕	Min ⇕	50th pct ⇕	75th pct ⇕	95th pct ⇕	99th pct ⇕	Max ⇕	Mean ⇕	Std Dev ⇕
All Requests	10	10	0	0%	0.208	115	385	553	757	829	847	415	221
request_0	1	1	0	0%	0.021	115	115	115	115	115	115	115	0
request_6	1	1	0	0%	0.021	646	646	646	646	646	646	646	0
request_1	1	1	0	0%	0.021	124	124	124	124	124	124	124	0
request_2	1	1	0	0%	0.021	315	315	315	315	315	315	315	0
request_3	1	1	0	0%	0.021	519	519	519	519	519	519	519	0
request_5	1	1	0	0%	0.021	248	248	248	248	248	248	248	0
request_4	1	1	0	0%	0.021	847	847	847	847	847	847	847	0
request_7	1	1	0	0%	0.021	357	357	357	357	357	357	357	0
request_13	1	1	0	0%	0.021	413	413	413	413	413	413	413	0
request_13 Redirect 1	1	1	0	0%	0.021	564	564	564	564	564	564	564	0

Register page:

- ✓ Registration provides greater protection for the public.
- ✓ Every time after the initial registration, the user will log on to the system using the username and password you created



CCRAGENTPAGE

Gatling Version

Version: 3.8.4

Released: 2022-09-13

Run Information

Date: 2022-11-13 04:52:09 GMT

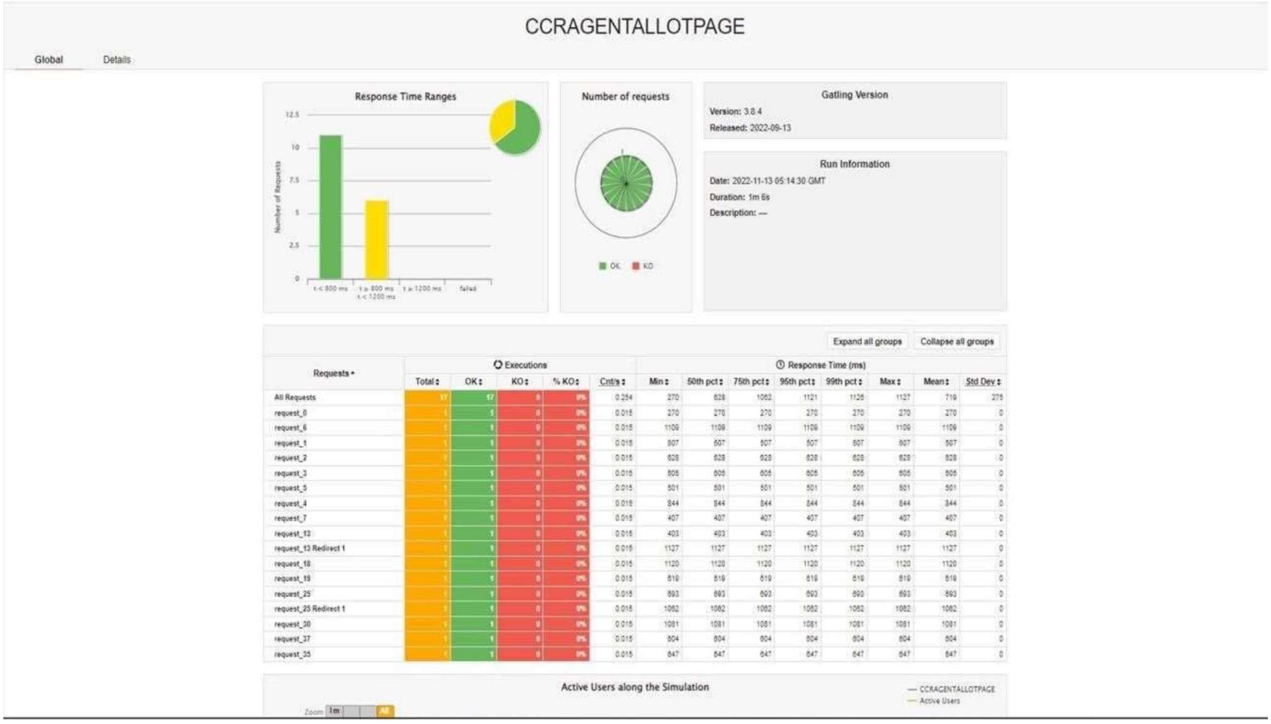
Duration: 1m 27s

Description: —

Requests •	Executions					Response Time (ms)							
	Total ⇕	OK ⇕	KO ⇕	% KO ⇕	Cnt/s ⇕	Min ⇕	50th pct ⇕	75th pct ⇕	95th pct ⇕	99th pct ⇕	Max ⇕	Mean ⇕	Std Dev ⇕
All Requests	15	15	0	0%	0.17	109	468	683	882	962	962	452	277
request_0	1	1	0	0%	0.011	529	529	529	529	529	529	529	0
request_3	1	1	0	0%	0.011	962	962	962	962	962	962	962	0
request_2	1	1	0	0%	0.011	490	490	490	490	490	490	490	0
request_1	1	1	0	0%	0.011	839	839	839	839	839	839	839	0
request_4	1	1	0	0%	0.011	406	406	406	406	406	406	406	0
css2?family=Permanent+Marker&dis...	1	1	0	0%	0.011	354	354	354	354	354	354	354	0
css?family=Nunito+Sans:200,300,40...	1	1	0	0%	0.011	128	128	128	128	128	128	128	0
css2?family=Audiowide&display=swap	1	1	0	0%	0.011	109	109	109	109	109	109	109	0
request_6	1	1	0	0%	0.011	136	136	136	136	136	136	136	0
request_7	1	1	0	0%	0.011	133	133	133	133	133	133	133	0
request_8	1	1	0	0%	0.011	129	129	129	129	129	129	129	0
request_10	1	1	0	0%	0.011	468	468	468	468	468	468	468	0
request_10 Redirect 1	1	1	0	0%	0.011	711	711	711	711	711	711	711	0
request_22	1	1	0	0%	0.011	685	685	685	685	685	685	685	0
request_17	1	1	0	0%	0.011	681	681	681	681	681	681	681	0

Donor Page:

After login the user can donate plasma or request plasma



CCRAGENTALLOTPAGE

Global Details

Gatling Version

Version: 3.8.4

Released: 2022-09-13

Run Information

Date: 2022-11-13 05:14:30 GMT

Duration: 1m 6s

Description: —

Expand all groups Collapse all groups

Requests •	Executions					Response Time (ms)							
	Total ⇕	OK ⇕	KO ⇕	% KO ⇕	Cnt/s ⇕	Min ⇕	50th pct ⇕	75th pct ⇕	95th pct ⇕	99th pct ⇕	Max ⇕	Mean ⇕	Std Dev ⇕
All Requests	17	17	0	0%	0.254	270	628	1062	1121	1126	1127	719	275
request_0	1	1	0	0%	0.015	270	270	270	270	270	270	270	0
request_6	1	1	0	0%	0.015	1109	1109	1109	1109	1109	1109	1109	0
request_1	1	1	0	0%	0.015	507	507	507	507	507	507	507	0
request_2	1	1	0	0%	0.015	628	628	628	628	628	628	628	0
request_3	1	1	0	0%	0.015	605	605	605	605	605	605	605	0
request_5	1	1	0	0%	0.015	501	501	501	501	501	501	501	0
request_4	1	1	0	0%	0.015	844	844	844	844	844	844	844	0
request_7	1	1	0	0%	0.015	407	407	407	407	407	407	407	0
request_13	1	1	0	0%	0.015	403	403	403	403	403	403	403	0
request_13 Redirect 1	1	1	0	0%	0.015	1127	1127	1127	1127	1127	1127	1127	0
request_18	1	1	0	0%	0.015	1120	1120	1120	1120	1120	1120	1120	0
request_19	1	1	0	0%	0.015	619	619	619	619	619	619	619	0
request_25	1	1	0	0%	0.015	693	693	693	693	693	693	693	0
request_25 Redirect 1	1	1	0	0%	0.015	1082	1082	1082	1082	1082	1082	1082	0
request_30	1	1	0	0%	0.015	1081	1081	1081	1081	1081	1081	1081	0
request_37	1	1	0	0%	0.015	604	604	604	604	604	604	604	0
request_35	1	1	0	0%	0.015	647	647	647	647	647	647	647	0

Active Users along the Simulation

8.3. User Acceptance Testing

- Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	4	2	5	4	15
Duplicate	3	0	2	0	5
External	4	2	0	1	7
Fixed	2	15	6	19	42
Not Reproduced	0	0	2	0	2
Skipped	0	0	3	0	3
Won't Fix	0	2	1	3	6
Totals	13	21	19	27	80

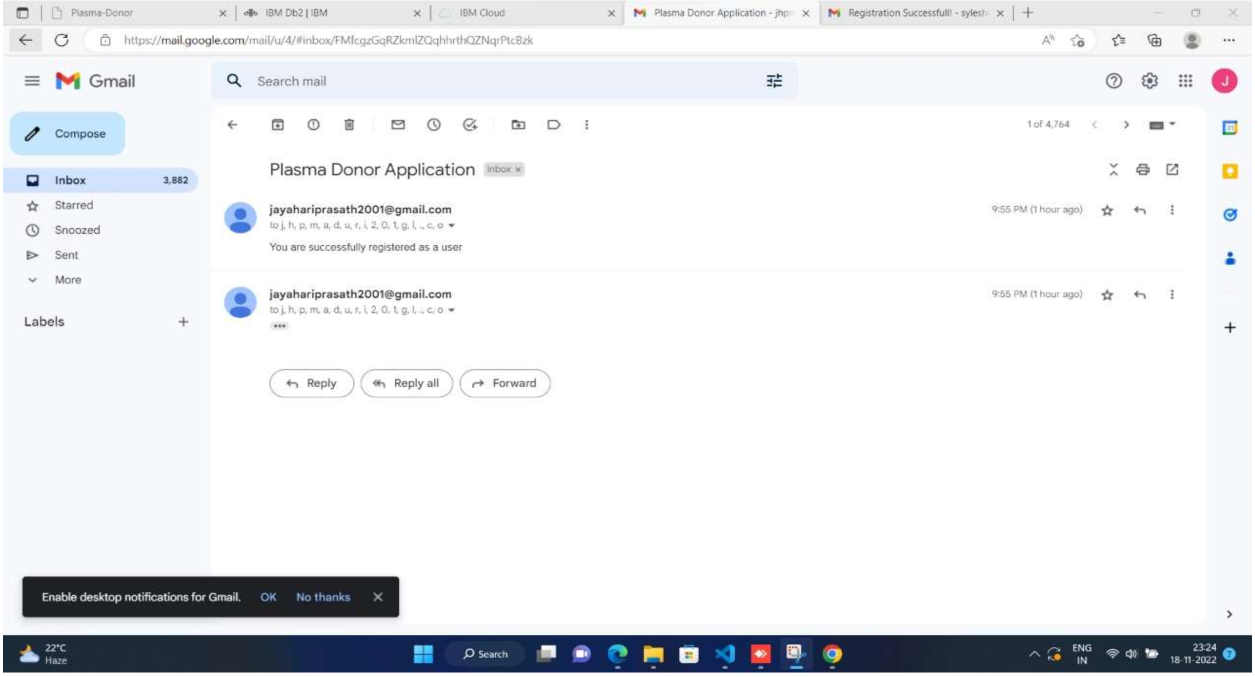
- TEST CASE ANALYSIS:

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	6	0	0	6
Client Application	49	0	0	49
Security	4	0	0	4
Outsource Shipping	5	0	0	5
Exception Reporting	7	0	0	7
Final Report Output	3	0	0	3
Version Control	1	0	0	1

9 .RESULT

9.1 Performance Metrics

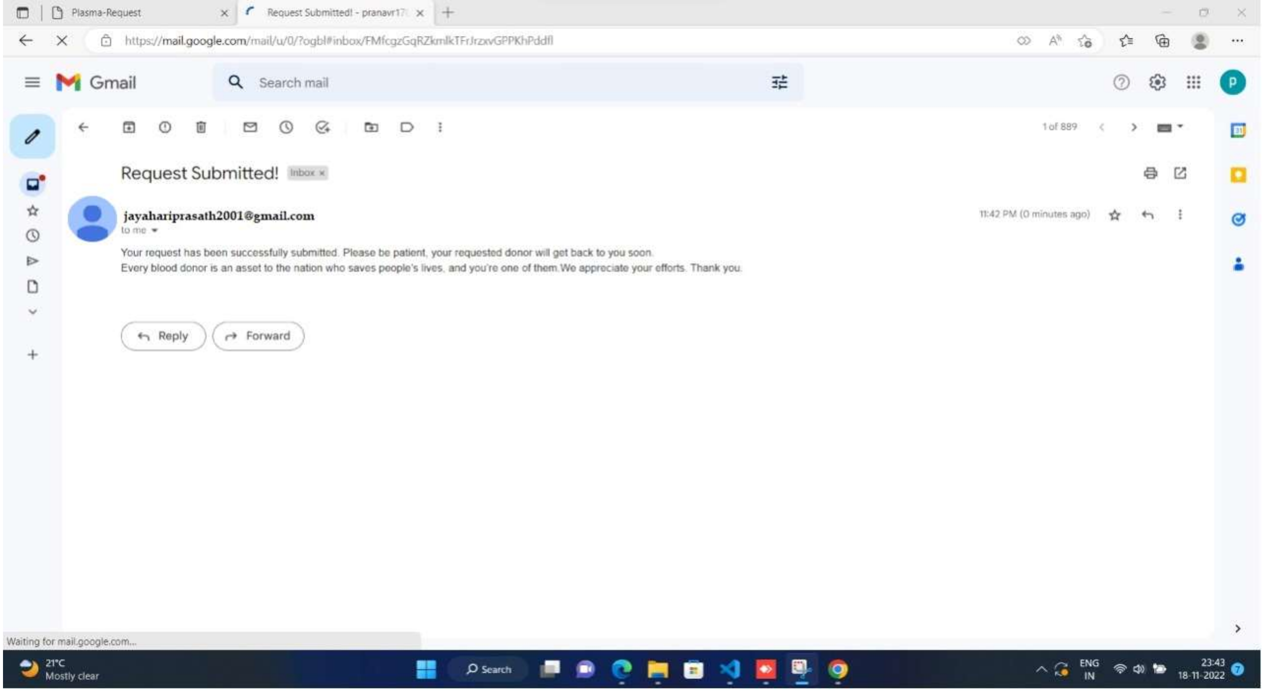

Plasma-Donor
IBM Db2 | IBM
IBM Cloud
Plasma Donor Application - jhp
Registration Successfull! - sylesh
https://mail.google.com/mail/u/4/#inbox/FMfcgzGqRZkmlZQqhhrthQZNqrPtcBzk
Gmail
Search mail
Compose
Inbox 3,882
Starred
Snoozed
Sent
More
Labels
1 of 4,764
Reply
Reply all
Forward
Enable desktop notifications for Gmail. OK No thanks
22°C Haze
Search
ENG IN
23:24 18-11-2022

Plasma Donor Application Inbox

jayahariprasath2001@gmail.com
to j, h, p, m, a, d, u, r, i, 2, 0, 1, g, l, ., c, o
9:55 PM (1 hour ago)

You are successfully registered as a user

jayahariprasath2001@gmail.com
to j, h, p, m, a, d, u, r, i, 2, 0, 1, g, l, ., c, o
9:55 PM (1 hour ago)


Plasma-Request
Request Submitted! - pranavr17
https://mail.google.com/mail/u/0/?ogbl#inbox/FMfcgzGqRZkmlkTFrJrzxvGPPKhPddfl
Gmail
Search mail
1 of 889
Reply
Forward
Waiting for mail.google.com...
21°C Mostly clear
Search
ENG IN
23:43 18-11-2022

Request Submitted! Inbox

jayahariprasath2001@gmail.com
to me
11:42 PM (0 minutes ago)

Your request has been successfully submitted. Please be patient, your requested donor will get back to you soon.
Every blood donor is an asset to the nation who saves people's lives, and you're one of them.We appreciate your efforts. Thank you.

10 .ADVANTAGES & DISADVANTAGES

ADVANTAGES:-

The application must have a simple accessible interface for donors to register and donate blood to the allocated blood bank. The users can create an account that can be used for registration and scheduling appointments at the nearest blood bank for blood donation Hospitals in need for blood plasma can request blood from blood banks. This application should have a wider reach to appeal to more potential blood donors.

- Notification of blood group for plasma donation
- connect the donors & the patients by an online application
- It will help people to find plasma easily.
- Helpful emergency situation.
-

DISADVANTAGES:-

Some of the disadvantages of the application are,

- The donor may reached their donation limit
- Internet Connection is mandatory.
- In order to donor he/she has to be a registered user
- Notification can irritate them if they are not ready to donate.
- Need to login every time while accessing the application

11. CONCLUSION

An Plasma Donor application is an exclusive suite of services for people who are in need of Plasma. It helps you track all the details about the Plasma Donors. The basic solution is to create a centralized system to keep a track on the upcoming as well as past Plasma Donation Events**.**

The Plasma donor application was developed out of a need to make finding plasma supplies or a willing donor on time and using lesser time in searching for either of the two. This system should be made available to everyone because it will help the search of plasma supplies during emergency cases faster. This helps to avoid health complication and also possible deaths due to delays in search of Plasma.

12. FUTURE SCOPE

In future, our algorithm more congenial with more features such as

- The application can be implemented using Artificial Intelligence and Deep Learning Algorithms.
- Donors last donated details can be automatically updated in the App.
- Increase efficiency and customer satisfaction with an app aligned to their needs.
- Seamlessly integrate with existing infrastructure
- The analysis such as Frequently requested zone or hospital for Plasma, Number of donors, mostly asked Blood Group, Age group of Patients need for plasma etc. can be added as additional features.
- NGOs and NCC Units information's can be made available in the application.

13. APPENDIX

13.1 Source Code:

home.css

```
@media only screen and (max-width: 500px) {
   /* For mobile phones: */
   .landing, .landing-text h1,.landing-text h3, .landing-text .btn,img , .landing-text .btn a{
     width: 100%;
     height: auto;
   }
 }
 .landing{
   margin-top: 100px;
   margin-left: 50px;
   margin-right: 50px;
 }

 .landing-text h1{
   font-size: 65px;
}


.landing-text h3{
   margin: 6px;
   font-size: 15px;
   line-height: 1.8;
   color: #e16eb5;
   margin-right: 20px;
}

.landing-text .btn{
   width: 200px;
   margin-top: 30px;
   padding: 14px 20px 12px 20px;
   background-color: #31cf4b;
   border-radius: 45px;
   text-align: center;

}
```

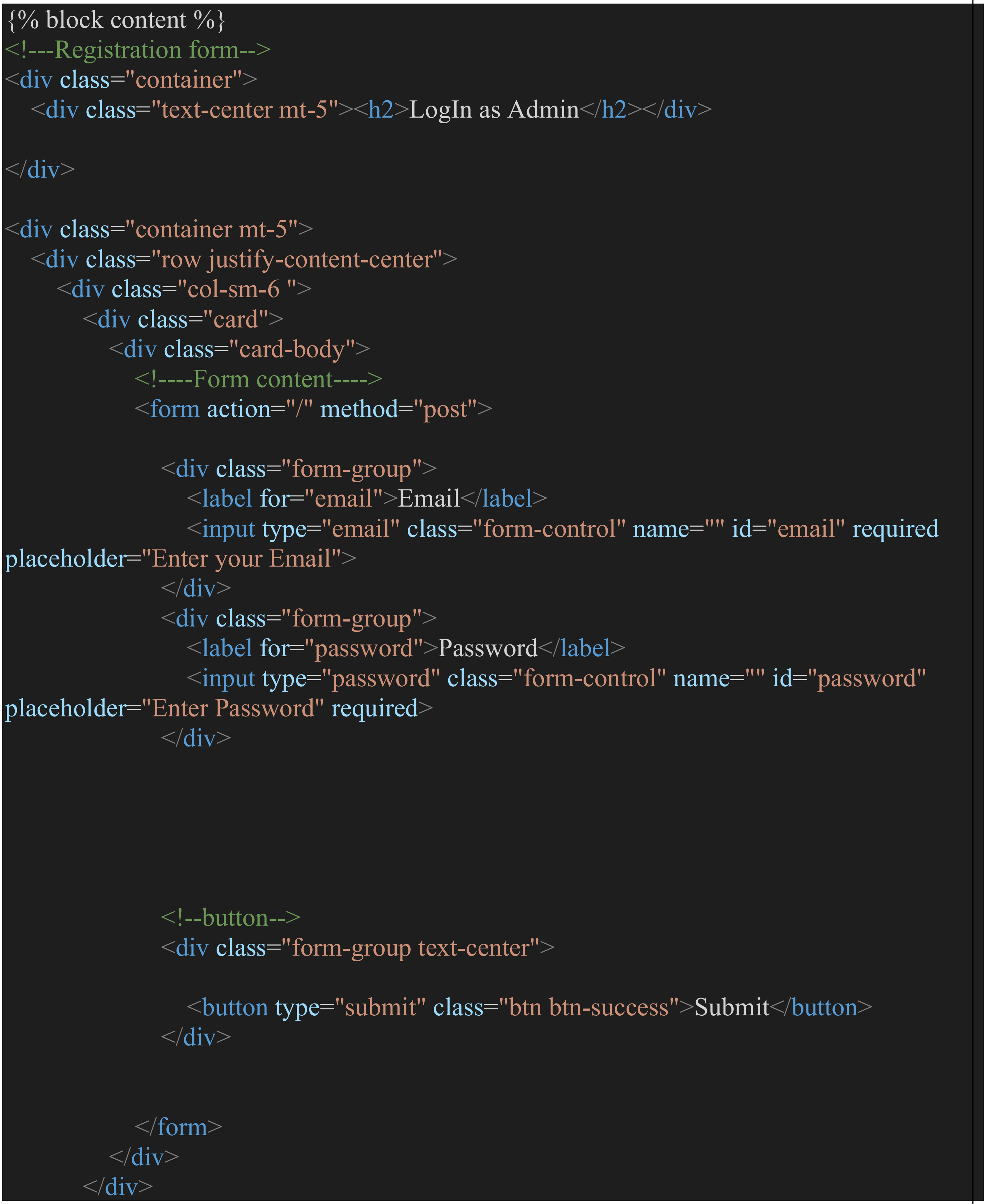
```
header{
  background-color: #212121;
}

.landing-text .btn a{
    font-size: 18px;
    color: #fff;
}

  img {
    float: right;
  }
```

Style.css

```
*{
    font-family: 'Montserrat', sans-serif;
}


#icon{
    color: rgb(239, 193, 214);
    padding-right: 2px;
}

.custom{
    background-color: #212121;
    color: white;
}
```

Admin_login.html

```
{% extends 'base.html'%}

<!--title tag-->
{% block title %}
<title>Admin-LogIn</title>
{% endblock %}

<!---Login Content-->
```



```
{% block content %}
<!---Registration form-->
<div class="container">
    <div class="text-center mt-5"><h2>LogIn as Admin</h2></div>

</div>

<div class="container mt-5">
    <div class="row justify-content-center">
        <div class="col-sm-6 ">
            <div class="card">
                <div class="card-body">
                    <!----Form content---->
                    <form action="/" method="post">

                        <div class="form-group">
                            <label for="email">Email</label>
                            <input type="email" class="form-control" name="" id="email" required
placeholder="Enter your Email">
                        </div>
                        <div class="form-group">
                            <label for="password">Password</label>
                            <input type="password" class="form-control" name="" id="password"
placeholder="Enter Password" required>
                        </div>





                        <!--button-->
                        <div class="form-group text-center">

                            <button type="submit" class="btn btn-success">Submit</button>
                        </div>


                    </form>
                </div>
            </div>
        </div>
    </div>
</div>

{% endblock %}
```

Base.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <!--font awesome-->
    <script src="https://kit.fontawesome.com/15af226b72.js"
crossorigin="anonymous"></script>

    <!-- Bootstrap CSS -->
    <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/css/bootstrap.min.css"
integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">

    <!--Google Font-->
    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Montserrat:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,700;1,800;1,900&display=swap');
    </style>
    <!--contains style for all pages-->
    <link rel="stylesheet" href="{{ url_for('static',filename='style.css')}}">
    <script src="https://kit.fontawesome.com/000fb23390.js"
crossorigin="anonymous"></script>
    {% block link %}
    {% endblock %}
    {% block title %}
    {% endblock %}
```

```
</head>
<body>

   <!-- Header -->
   <header>
<nav class="navbar navbar-expand-lg custom" style="color: white;">
   <div class="container">
   <a href="{{ url_for('index')}}" class="navbar-brand"><i class="fa-solid fa-droplet" id="icon" style="color: white;"></i>
      Plasma Donor Application</a>

   <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#myNavBar"
      aria-controls="myNavBar" aria-expanded="false" aria-label="Toggle navigation">

      <span class="navbar-toggler-icon"></span>
   </button>

   <div class="collapse navbar-collapse" id="myNavBar">
      <ul class="navbar-nav ml-auto">
         <li class="nav-item active">
            <a href="{{ url_for('home_page') }}" class="nav-link">Home</a>
         </li>


         <li class="nav-item">
            <a href="{{ url_for('signin') }}" class="nav-link" style="color:white ;">Register</a>
         </li>
         <li class="nav-item">
            <a href="{{ url_for('login') }}" class="nav-link" style="color:white ;">Login</a>
         </li>

         </ul>
      </div>
   </div>

   </nav>
</header>
   <!-- End Header -->
```

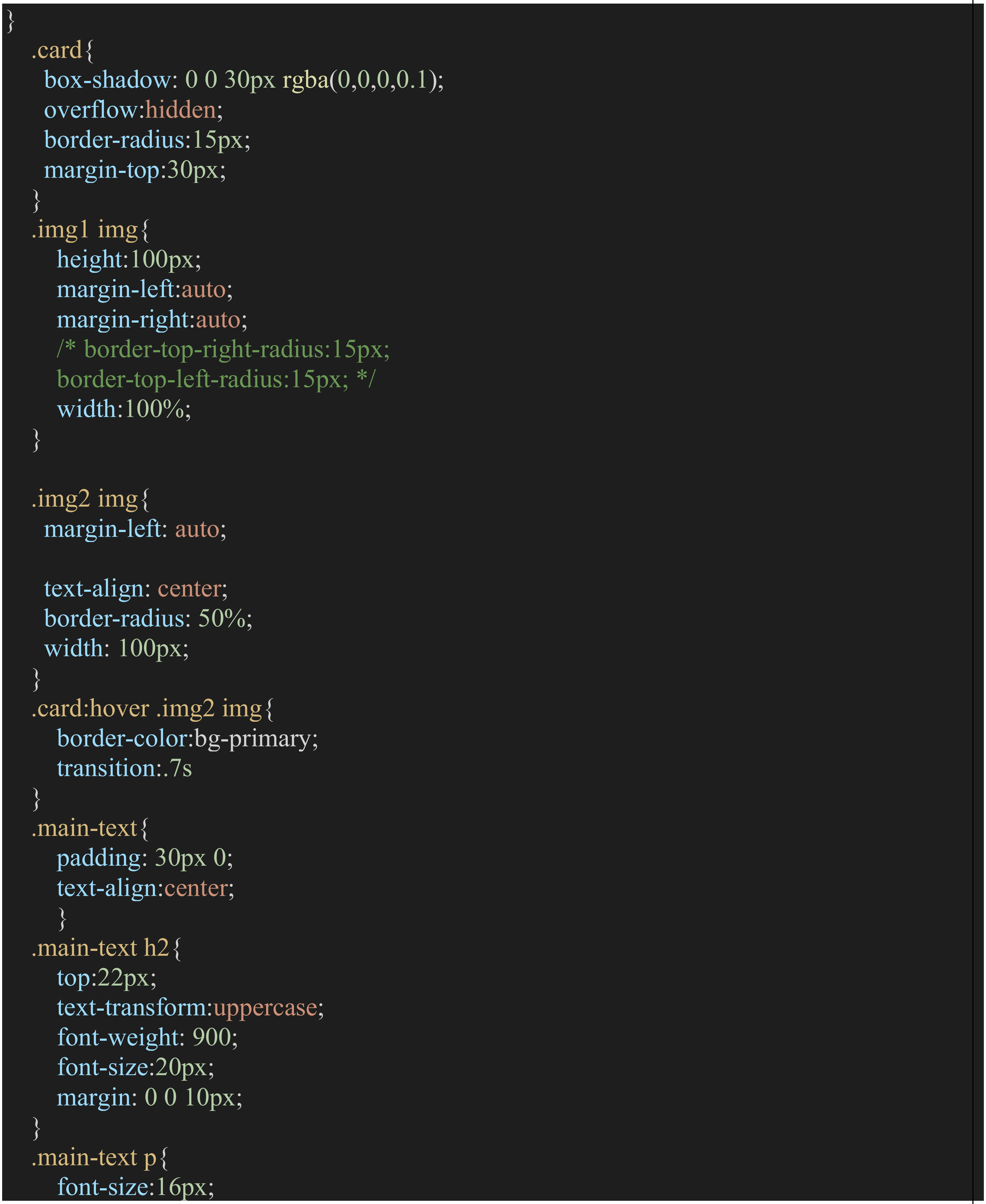
```
    <!--Future contents-->
    {% block content %}
    {% endblock %}


    <!-- Optional JavaScript -->
    <!-- jQuery first, then Popper.js, then Bootstrap JS -->
    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></script>
    <script src="https://cdn.jsdelivr.net/npm/popper.js@1.14.7/dist/umd/popper.min.js" integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM" crossorigin="anonymous"></script>
    <script>
        window.watsonAssistantChatOptions = {
          integrationID: "18cd43fb-1fde-4fb9-b9d2-a7c1095e980b", // The ID of this integration.
          region: "au-syd", // The region your integration is hosted in.
          serviceInstanceID: "05ba06eb-03f7-412f-b60f-5a4a4f83de97", // The ID of your service instance.
          onLoad: function(instance) { instance.render(); }
        };
        setTimeout(function(){
          const t=document.createElement('script');
          t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" + (window.watsonAssistantChatOptions.clientVersion || 'latest') + "/WatsonAssistantChatEntry.js";
          document.head.appendChild(t);
        });
      </script>
</body>
</html>
```

Donor.html

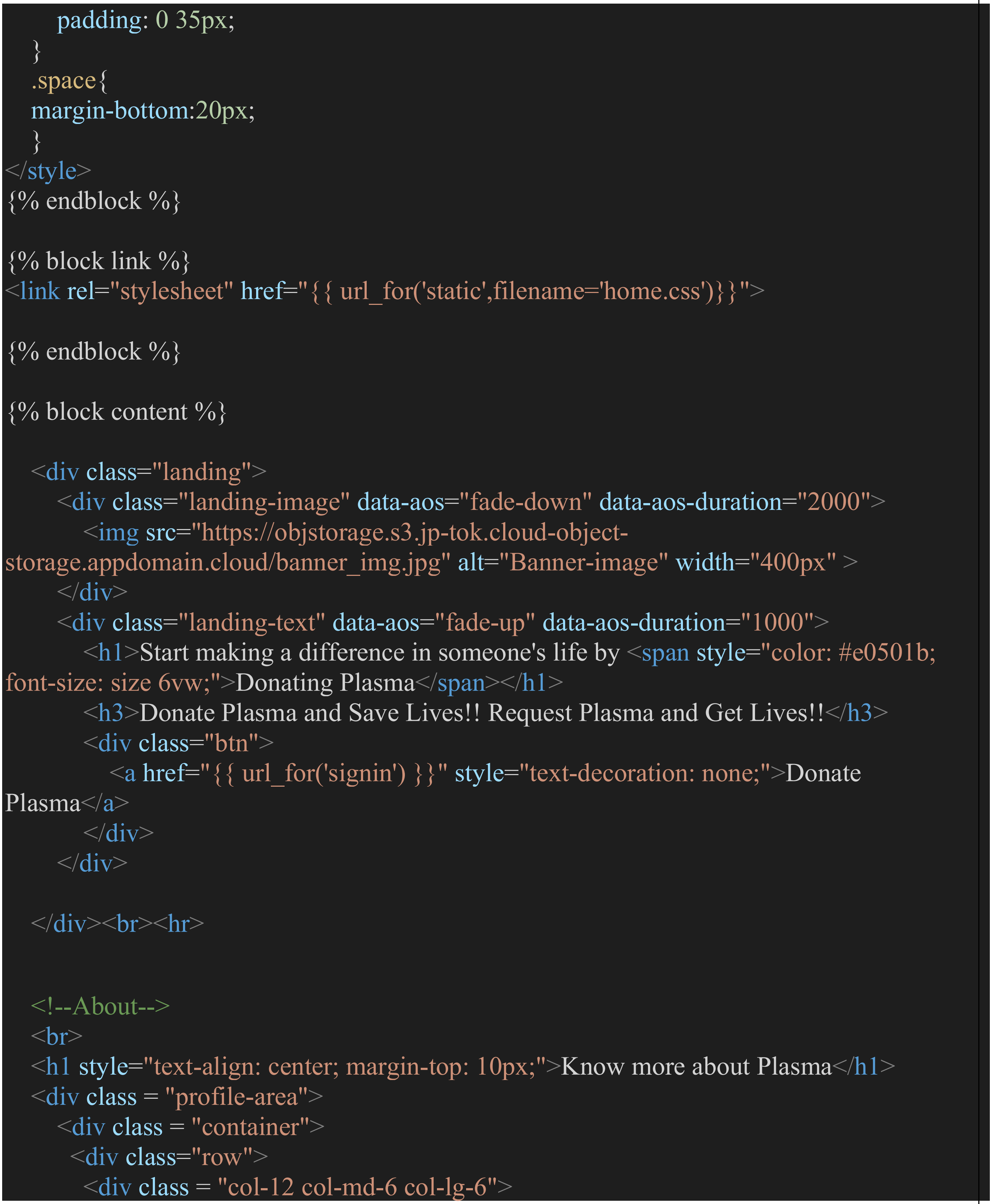
```
{% extends 'base.html'%}

<!--title tag-->
{% block title %}
<title>Plasma-Donor</title>
{% endblock %}

<!---Donor Content-->
{% block content %}
<!---Donor table-->
<div class="container mt-3">
    <div class="row justify-content-center">
        <div class="col-sm-12 ">
          <div class="msg">{{ msg }}</div>
            <div class="">
                <div class="">
                  <h6 style="text-align: center; margin-top: 50px;color: red;">Note: Please note the
donor email from the table you want to request.</h6>
  <table class="table table-hover table-bordered" style="margin:100px 0px; text-align:
center;">
    <thead class="thead-light">
      <tr>


        <th scope="col">Email</th>
        <th scope="col">Age</th>
        <th scope="col">Gender</th>
        <th scope="col">Blood Group</th>
        <th scope="col">Area</th>
        <th scope="col">City</th>
        <th scope="col">District</th>
        <th scope="col">Make a Request</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        {% for row in donor2 %}
        <td>{{ row["EMAIL"] }}</td>
        <td>{{ row["AGE"] }}</td>
        <td>{{ row["GENDER"] }}</td>
        <td>{{ row["BLOOD"] }}</td>
        <td>{{ row["AREA"] }}</td>
        <td>{{ row["CITY"] }}</td>
        <td>{{ row["DISTRICT"] }}</td>
        <td>
        <a href="{{url_for('request_page')}}" class="btn-sm btn-info" style="color: white; text-
decoration:none">Request</a>
        </td>
      </tr>
      {% endfor %}

    </tbody>
  </table>
</div>
</div>
</div>
</div>
</div>
{% endblock %}
```

Home.html

```
{% extends 'base.html' %}

<!--title tag-->
{% block title %}
<title>Plasma-Home</title>
<style>
    body{
    background:#fff;
    }

    .heading{
        padding-top: 30px;
        text-align: center;
        font-weight: 500;
    }

    .profile-area{
    padding:30px 0
```

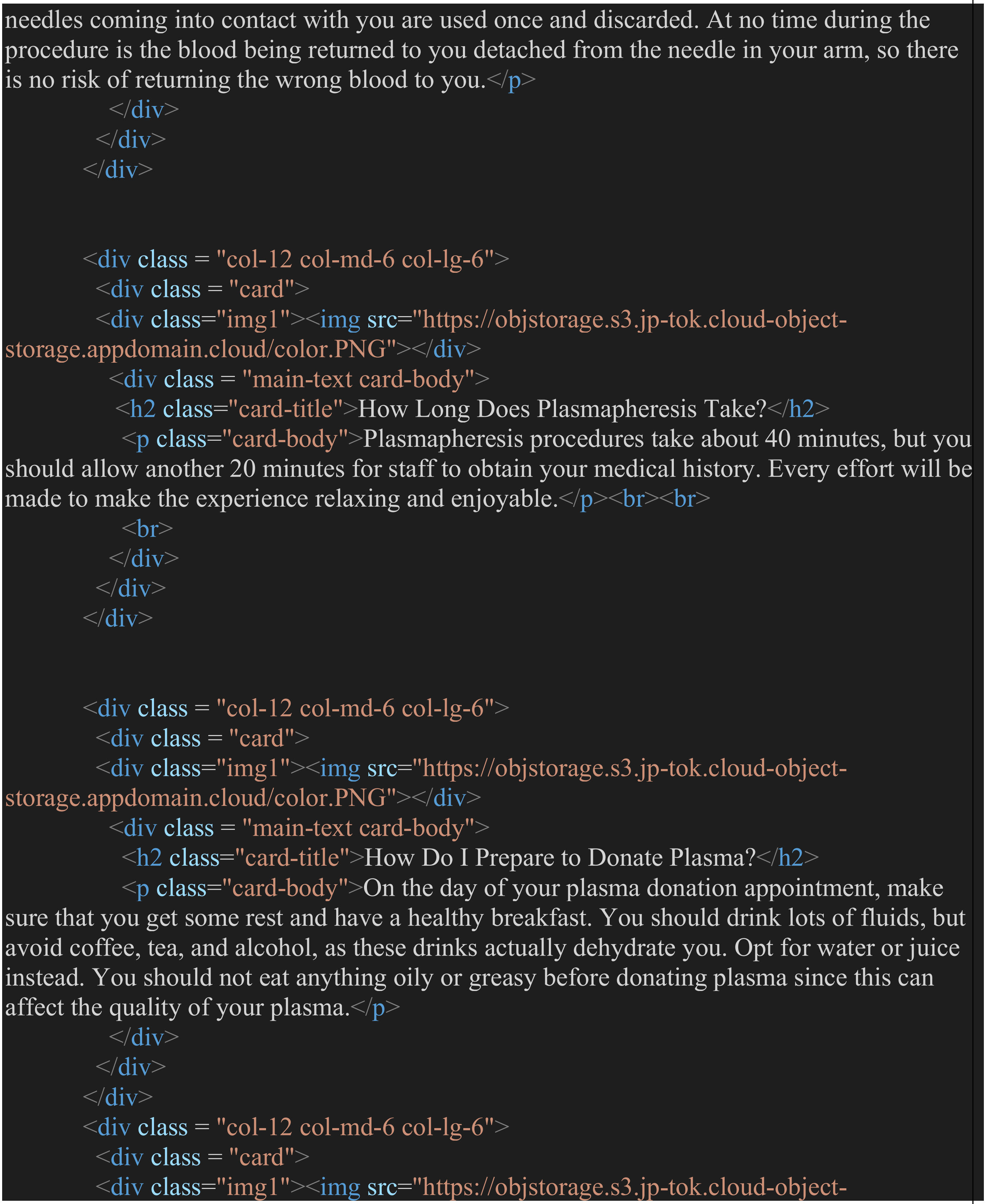


```
}
  .card{
   box-shadow: 0 0 30px rgba(0,0,0,0.1);
   overflow:hidden;
   border-radius:15px;
   margin-top:30px;
  }
  .img1 img{
    height:100px;
    margin-left:auto;
    margin-right:auto;
    /* border-top-right-radius:15px;
    border-top-left-radius:15px; */
    width:100%;
  }

  .img2 img{
   margin-left: auto;

   text-align: center;
   border-radius: 50%;
   width: 100px;
  }
  .card:hover .img2 img{
    border-color:bg-primary;
    transition:.7s
  }
  .main-text{
    padding: 30px 0;
    text-align:center;
    }
  .main-text h2{
    top:22px;
    text-transform:uppercase;
    font-weight: 900;
    font-size:20px;
    margin: 0 0 10px;
  }
  .main-text p{
    font-size:16px;
```

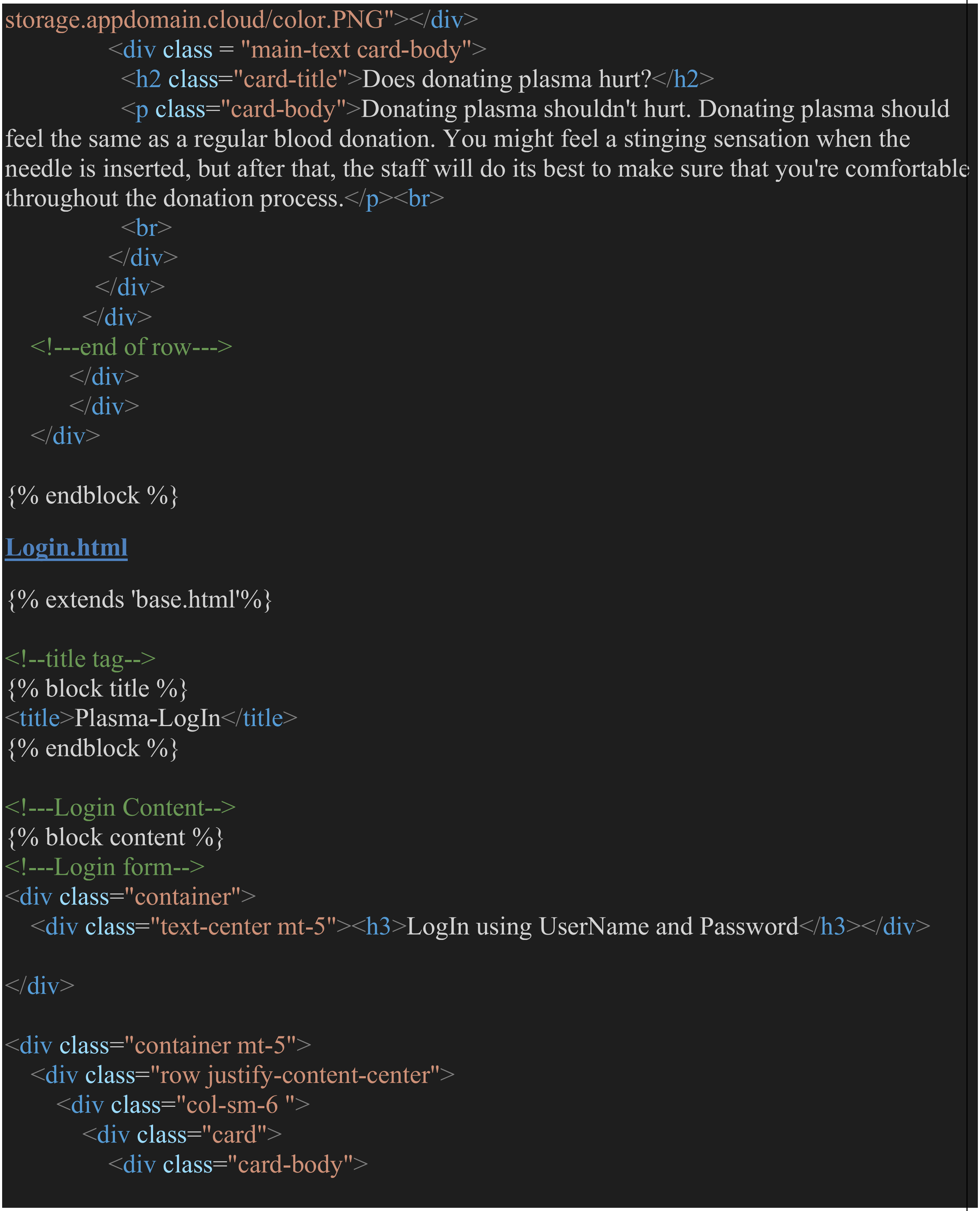


```
        padding: 0 35px;
    }
    .space{
    margin-bottom:20px;
    }
</style>
{% endblock %}

{% block link %}
<link rel="stylesheet" href="{{ url_for('static',filename='home.css')}}">

{% endblock %}

{% block content %}

    <div class="landing">
        <div class="landing-image" data-aos="fade-down" data-aos-duration="2000">
            <img src="https://objstorage.s3.jp-tok.cloud-object-
storage.appdomain.cloud/banner_img.jpg" alt="Banner-image" width="400px" >
        </div>
        <div class="landing-text" data-aos="fade-up" data-aos-duration="1000">
            <h1>Start making a difference in someone's life by <span style="color: #e0501b;
font-size: size 6vw;">Donating Plasma</span></h1>
            <h3>Donate Plasma and Save Lives!! Request Plasma and Get Lives!!</h3>
            <div class="btn">
                <a href="{{ url_for('signin') }}" style="text-decoration: none;">Donate
Plasma</a>
            </div>
        </div>

    </div><br><hr>


    <!--About-->
    <br>
    <h1 style="text-align: center; margin-top: 10px;">Know more about Plasma</h1>
    <div class = "profile-area">
        <div class = "container">
          <div class="row">
            <div class = "col-12 col-md-6 col-lg-6">
```

```
       <div class = "card custom">
         <div  class="img1"><img src="https://objstorage.s3.jp-tok.cloud-object-
storage.appdomain.cloud/color.PNG"></div>

        <div class = "main-text card-body">
         <h2 class="card-title">What is Plasma?</h2>
         <p class="card-body">Plasma is the pale yellow liquid part
          of whole blood, in which the cellular elements are suspended.  It is enriched in
proteins that help fight infection and aid the blood in clotting.  AB plasma is plasma collected
from blood group AB donors. It is considered "universal donor" plasma because it is suitable
for all recipients, regardless of blood group.</p>
        </div>
       </div>
      </div>


      <div class = "col-12 col-md-6 col-lg-6">
       <div class = "card">
        <div class="img1"><img src="https://objstorage.s3.jp-tok.cloud-object-
storage.appdomain.cloud/color.PNG"></div>
        <div class = "main-text card-body">
         <h2 class="card-title">What is Plasmapheresis?</h2>
         <p class="card-body">Plasmapheresis is the standard procedure by which plasma
is separated from whole blood and collected. Blood flows through a single needle placed in
an arm vein, into a machine that contains a sterile, disposable plastic kit. The plasma is
isolated and channeled out into a special bag, and red blood cells and other parts of the blood
are returned to you through the same needle.</p>
        </div>
       </div>
      </div>


      <div class = "col-12 col-md-6 col-lg-6">
       <div class = "card">
       <div class="img1"><img src="https://objstorage.s3.jp-tok.cloud-object-
storage.appdomain.cloud/color.PNG"></div>
        <div class = "main-text card-body">
         <h2 class="card-title">Is Plasmapheresis Safe?</h2>
         <p class="card-body">Absolutely. The machine and the procedure have been
evaluated and approved by the Food and Drug Administration (FDA), and all plastics and
```



```
needles coming into contact with you are used once and discarded. At no time during the
procedure is the blood being returned to you detached from the needle in your arm, so there
is no risk of returning the wrong blood to you.</p>
                </div>
              </div>
            </div>


            <div class = "col-12 col-md-6 col-lg-6">
              <div class = "card">
              <div class="img1"><img src="https://objstorage.s3.jp-tok.cloud-object-
storage.appdomain.cloud/color.PNG"></div>
                <div class = "main-text card-body">
                 <h2 class="card-title">How Long Does Plasmapheresis Take?</h2>
                  <p class="card-body">Plasmapheresis procedures take about 40 minutes, but you
should allow another 20 minutes for staff to obtain your medical history. Every effort will be
made to make the experience relaxing and enjoyable.</p><br><br>
                  <br>
                </div>
              </div>
            </div>


            <div class = "col-12 col-md-6 col-lg-6">
              <div class = "card">
              <div class="img1"><img src="https://objstorage.s3.jp-tok.cloud-object-
storage.appdomain.cloud/color.PNG"></div>
                <div class = "main-text card-body">
                  <h2 class="card-title">How Do I Prepare to Donate Plasma?</h2>
                  <p class="card-body">On the day of your plasma donation appointment, make
sure that you get some rest and have a healthy breakfast. You should drink lots of fluids, but
avoid coffee, tea, and alcohol, as these drinks actually dehydrate you. Opt for water or juice
instead. You should not eat anything oily or greasy before donating plasma since this can
affect the quality of your plasma.</p>
                </div>
              </div>
            </div>
            <div class = "col-12 col-md-6 col-lg-6">
              <div class = "card">
              <div class="img1"><img src="https://objstorage.s3.jp-tok.cloud-object-
```



```
storage.appdomain.cloud/color.PNG"></div>
                <div class = "main-text card-body">
                  <h2 class="card-title">Does donating plasma hurt?</h2>
                  <p class="card-body">Donating plasma shouldn't hurt. Donating plasma should feel the same as a regular blood donation. You might feel a stinging sensation when the needle is inserted, but after that, the staff will do its best to make sure that you're comfortable throughout the donation process.</p><br>
                  <br>
                </div>
              </div>
            </div>
    <!---end of row--->
          </div>
          </div>
    </div>

{% endblock %}
```

Login.html

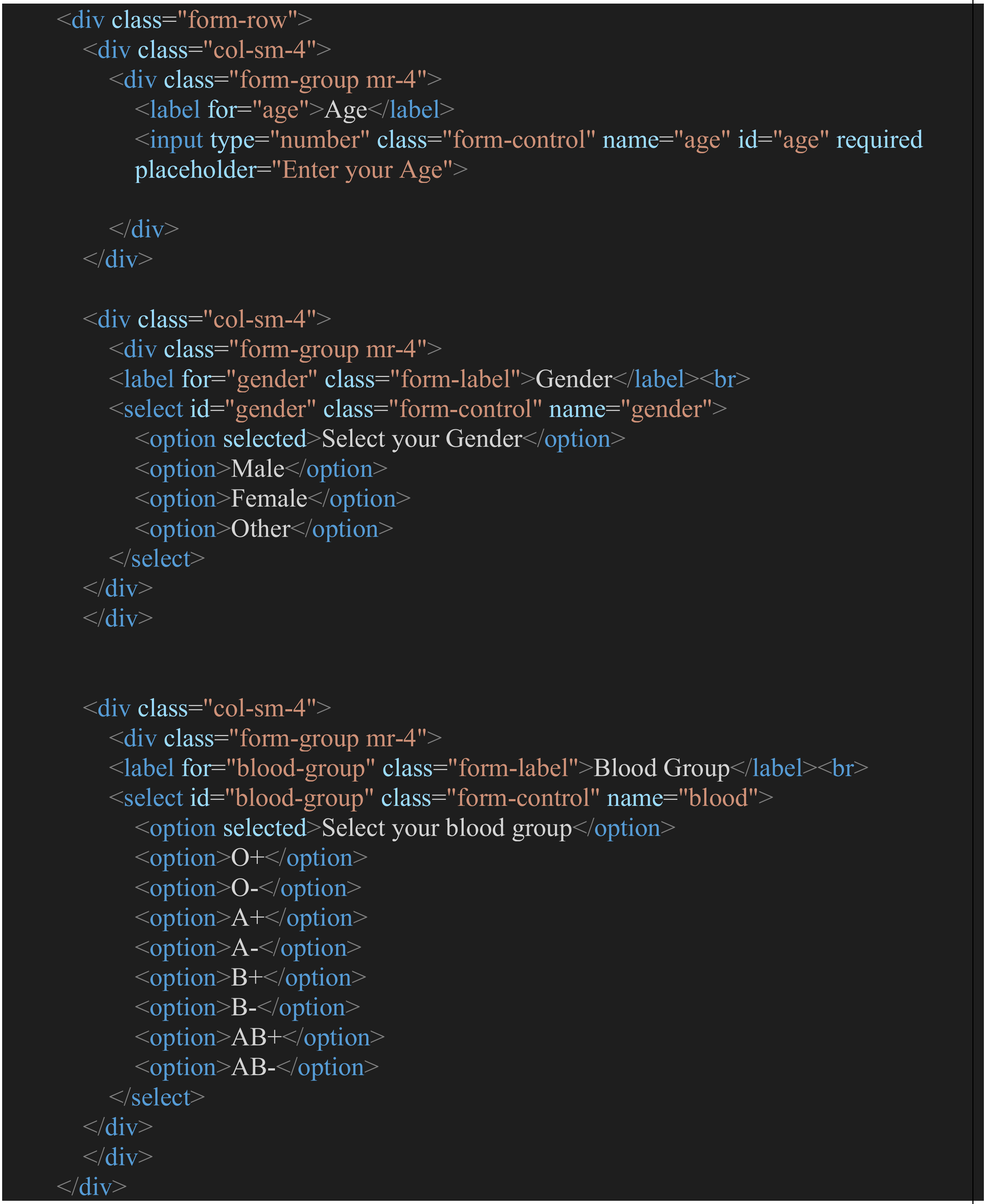
```
{% extends 'base.html'%}

<!--title tag-->
{% block title %}
<title>Plasma-LogIn</title>
{% endblock %}

<!---Login Content-->
{% block content %}
<!---Login form-->
<div class="container">
    <div class="text-center mt-5"><h3>LogIn using UserName and Password</h3></div>

</div>

<div class="container mt-5">
    <div class="row justify-content-center">
        <div class="col-sm-6 ">
            <div class="card">
                <div class="card-body">
```

```
                    <!----Form content---->
                    <form action="/login" method="POST">
                        <div class="msg" style="color: green;">{{ msg }}</div>

                        <div class="form-group">
                            <label for="username">User Name</label>
                            <input type="text" class="form-control" name="username" id="username"
required placeholder="Enter UserName">
                        </div>
                        <div class="form-group">
                            <label for="password">Password</label>
                            <input type="password" class="form-control" name="password"
id="password" placeholder="Enter Password" required>
                        </div>


                        <!--button-->
                        <div class="form-group text-center">

                        <input type="submit" value="LogIn" class="btn btn-success">
                        </div>
                        <br>
                        <div style="text-align: center;">
                            <p>Don't have an account <a href="{{ url_for('signin') }}">Register
here</a>
                        </div>

                    </form>
                </div>
            </div>
        </div>
    </div>
</div>

{% endblock %}
```

Register.html

```
{% extends 'base.html' %}

<!--title tag-->
```

```
{% block title %}
<title>Plasma-Register</title>
{% endblock %}

{% block content %}

<!---Registration form-->

<div class="container mt-5<div class="row"<div class="col-sm-12"><div class="card">
<div class="card-body">
      <form action="/adddonor" method="post">
      <h4 style="text-align: center;">Donating Plasma</h4>
      <div class="form-group ">
         <label for="name">Full Name</label>
         <input type="text" class="form-control" name="name" id="name"
placeholder="Enter your Name"
                      required>

      </div>
      <!--Splitting into two grids-->
      <div class="form-row">
         <div class="col-sm-6">
            <div class="form-group mr-4 ">
               <label for="mobile">Mobile Number</label>
               <input type="tel" class="form-control" name="mobile" id="mobile" required
               placeholder="Enter your Mobile No.">

            </div>
         </div>
         <div class="col-sm-6">
            <div class="form-group">
               <label for="email">Email</label>
               <input type="email" class="form-control" name="email" id="email" required
               placeholder="Enter your Email">

            </div>

         </div>
      </div>
```



```
<div class="form-row">
  <div class="col-sm-4">
    <div class="form-group mr-4">
      <label for="age">Age</label>
      <input type="number" class="form-control" name="age" id="age" required
      placeholder="Enter your Age">

    </div>
  </div>

  <div class="col-sm-4">
    <div class="form-group mr-4">
    <label for="gender" class="form-label">Gender</label><br>
    <select id="gender" class="form-control" name="gender">
      <option selected>Select your Gender</option>
      <option>Male</option>
      <option>Female</option>
      <option>Other</option>
    </select>
  </div>
  </div>


  <div class="col-sm-4">
    <div class="form-group mr-4">
    <label for="blood-group" class="form-label">Blood Group</label><br>
    <select id="blood-group" class="form-control" name="blood">
      <option selected>Select your blood group</option>
      <option>O+</option>
      <option>O-</option>
      <option>A+</option>
      <option>A-</option>
      <option>B+</option>
      <option>B-</option>
      <option>AB+</option>
      <option>AB-</option>
    </select>
  </div>
  </div>
</div>
```

```
        <div class="form-row">
            <div class="col-sm-4">
                <div class="form-group mr-4 ">
                    <label for="Area">Area</label>
                    <input type="text" class="form-control" name="area" id="Area" required
                    placeholder="Enter your Area Name">

                </div>
            </div>
            <div class="col-sm-4">
                <div class="form-group">
                    <label for="city">City</label>
                    <input type="text" class="form-control" name="city" id="city" required
                    placeholder="Enter your City Name">

                </div>

            </div>
            <div class="col-sm-4">
                <div class="form-group mr-4 ">
                    <label for="district">District</label>
                    <input type="text" class="form-control" name="district" id="district" required
                    placeholder="Enter your District Name">

                </div>
            </div>
        </div>


                    <!--button-->
            <div class="form-group text-center">
                <input type="reset" value="Reset" class="btn btn-dark mr-2">
                <input type="submit" value="Submit" class="btn btn-success">

            </div>

        </form>
    </div>
</div>
```

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




{% endblock %}
```

Request.html

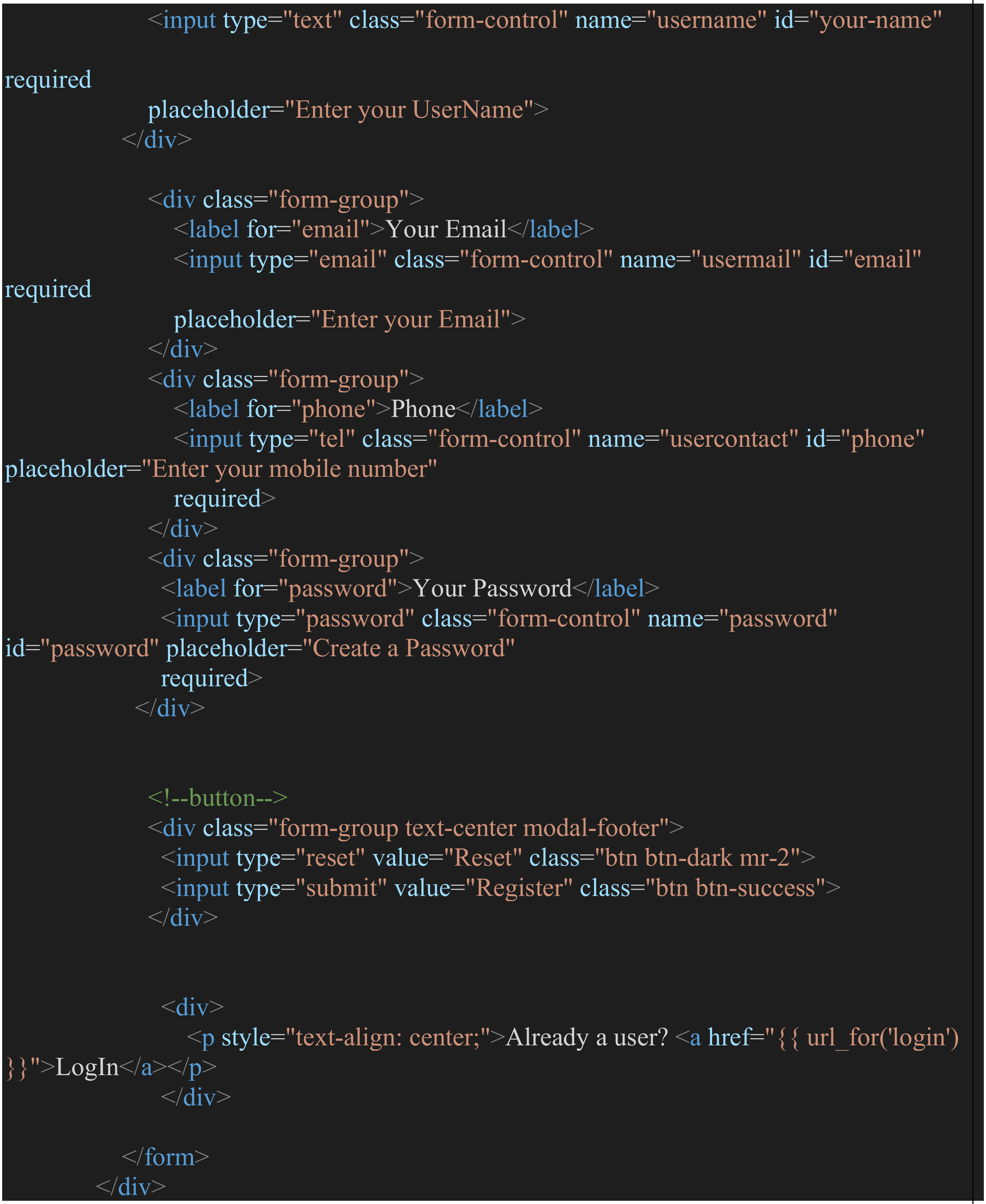
```
{% extends 'base.html'%}

<!--title tag-->
{% block title %}
<title>Plasma-Request</title>
{% endblock %}

<!---Login Content-->
{% block content %}
<!---Registration form-->


<div class="container mt-5" id="request-form">
  <div class="row justify-content-center">
      <div class="col-sm-12 ">
          <div class="card">
            <div class="msg" style="color: green;">{{ msg }}</div>
              <div class="card-body">
                <h4 style="text-align: center;">Request for Plasma</h4>
                  <!----Form content---->

                  <form action="/request_page" method="post">

                    <div class="form-row">
                        <div class="col-sm-6">
                            <div class="form-group mr-4 ">
                                <label for="drmail">Enter Donor Mail</label>
                                <input type="email" class="form-control" name="drmail" id="drmail"
required
                                placeholder="Enter Donor mail from the table">
                            </div>
                        </div>
                        <div class="col-sm-6">
                            <div class="form-group">
                                <label for="hospitalname">Hospital Name</label>
                                <input type="text" class="form-control" name="hospitalname"
id="hospitalname" required
                                placeholder="Enter Hospital Nmae">

                            </div>

                        </div>
                    </div>

                    <div class="form-row">
                        <div class="col-sm-4">
                            <div class="form-group mr-4 ">
                                <label for="fullname">FullName</label>
                                <input type="text" class="form-control" name="recname" id="fullname"
required

                                placeholder="Enter your FullName">

                            </div>
                        </div>
                        <div class="col-sm-4">
                            <div class="form-group">
                                <label for="mobile">Mobile Number</label>
                                <input type="tel" class="form-control" name="recmobile" id="mobile"
required

                                placeholder="Enter your Mobile Number">

                            </div>

                        </div>
                        <div class="col-sm-4">
                            <div class="form-group mr-4 ">
                                <label for="recmail">Your Mail</label>
                                <input type="email" class="form-control" name="recmail" id="recmail"
required

                                placeholder="Enter your Email">
```

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


    <div class="form-row">
      <div class="col-sm-4">
        <div class="form-group mr-4">
          <label for="age">Age</label>
          <input type="number" class="form-control" name="recage" id="age"
required

          placeholder="Enter your Age">

        </div>
      </div>

      <div class="col-sm-4">
        <div class="form-group mr-4">
        <label for="gender" class="form-label">Gender</label><br>
        <select id="gender" class="form-control" name="recgender">
         <option selected>Select your Gender</option>
         <option>Male</option>
         <option>Female</option>
         <option>Other</option>
        </select>
      </div>
      </div>


      <div class="col-sm-4">
        <div class="form-group mr-4">
        <label for="blood-group" class="form-label">Blood Group</label><br>
        <select id="blood-group" class="form-control" name="recbloodgroup">
         <option selected>Select your blood group</option>
         <option>O+</option>
         <option>O-</option>
         <option>A+</option>
         <option>A-</option>
         <option>B+</option>
```

```
               <option>B-</option>
               <option>AB+</option>
               <option>AB-</option>
              </select>
            </div>
            </div>
           </div>

            <div class="form-row">
              <div class="col-sm-4">
                <div class="form-group mr-4 ">
                  <label for="Area">Area</label>
                  <input type="text" class="form-control" name="recarea" id="Area"
required
                  placeholder="Enter your Area Name">

                </div>
              </div>
              <div class="col-sm-4">
                <div class="form-group">
                  <label for="city">City</label>
                  <input type="text" class="form-control" name="reccity" id="city"
required
                  placeholder="Enter your City Name">

                </div>

              </div>
              <div class="col-sm-4">
                <div class="form-group mr-4 ">
                  <label for="district">District</label>
                  <input type="text" class="form-control" name="recdistrict"
id="district" required
                  placeholder="Enter your District Name">

                </div>
              </div>
            </div>
           <!--button-->
          <div class="form-group text-center modal-footer">
            <button type="reset" class="btn btn-secondary" data-dismiss="modal">Reset</button>
            <button type="submit" class="btn btn-success">Request</button>
          </div>

         </form>
       </div>
     </div>
   </div>
 </div>
</div>



{% endblock %}
```

Signin.html

```
{% extends 'base.html'%}

<!--title tag-->
{% block title %}
<title>Plasma-Signin</title>
{% endblock %}

<!---Login Content-->
{% block content %}
<!---Registration form-->


<div class="container mt-5" id="request-form">
 <div class="row justify-content-center">
   <div class="col-sm-6 ">
     <div class="card">
       <div class="card-body">
        <h4 style="text-align: center;">Register as a user</h4>
         <!----Form content---->
         <form action="/signin" method="post">
          <div class="form-group">
           <label for="your-name">User Name</label>
```



```
                    <input type="text" class="form-control" name="username" id="your-name"

required
                    placeholder="Enter your UserName">
                </div>

                    <div class="form-group">
                        <label for="email">Your Email</label>
                        <input type="email" class="form-control" name="usermail" id="email"
required
                        placeholder="Enter your Email">
                    </div>
                    <div class="form-group">
                        <label for="phone">Phone</label>
                        <input type="tel" class="form-control" name="usercontact" id="phone"
placeholder="Enter your mobile number"
                        required>
                    </div>
                    <div class="form-group">
                      <label for="password">Your Password</label>
                      <input type="password" class="form-control" name="password"
id="password" placeholder="Create a Password"
                      required>
                  </div>


                    <!--button-->
                    <div class="form-group text-center modal-footer">
                      <input type="reset" value="Reset" class="btn btn-dark mr-2">
                      <input type="submit" value="Register" class="btn btn-success">
                    </div>



                      <div>
                          <p style="text-align: center;">Already a user? <a href="{{ url_for('login')
}}">LogIn</a></p>
                      </div>

                </form>
            </div>
```

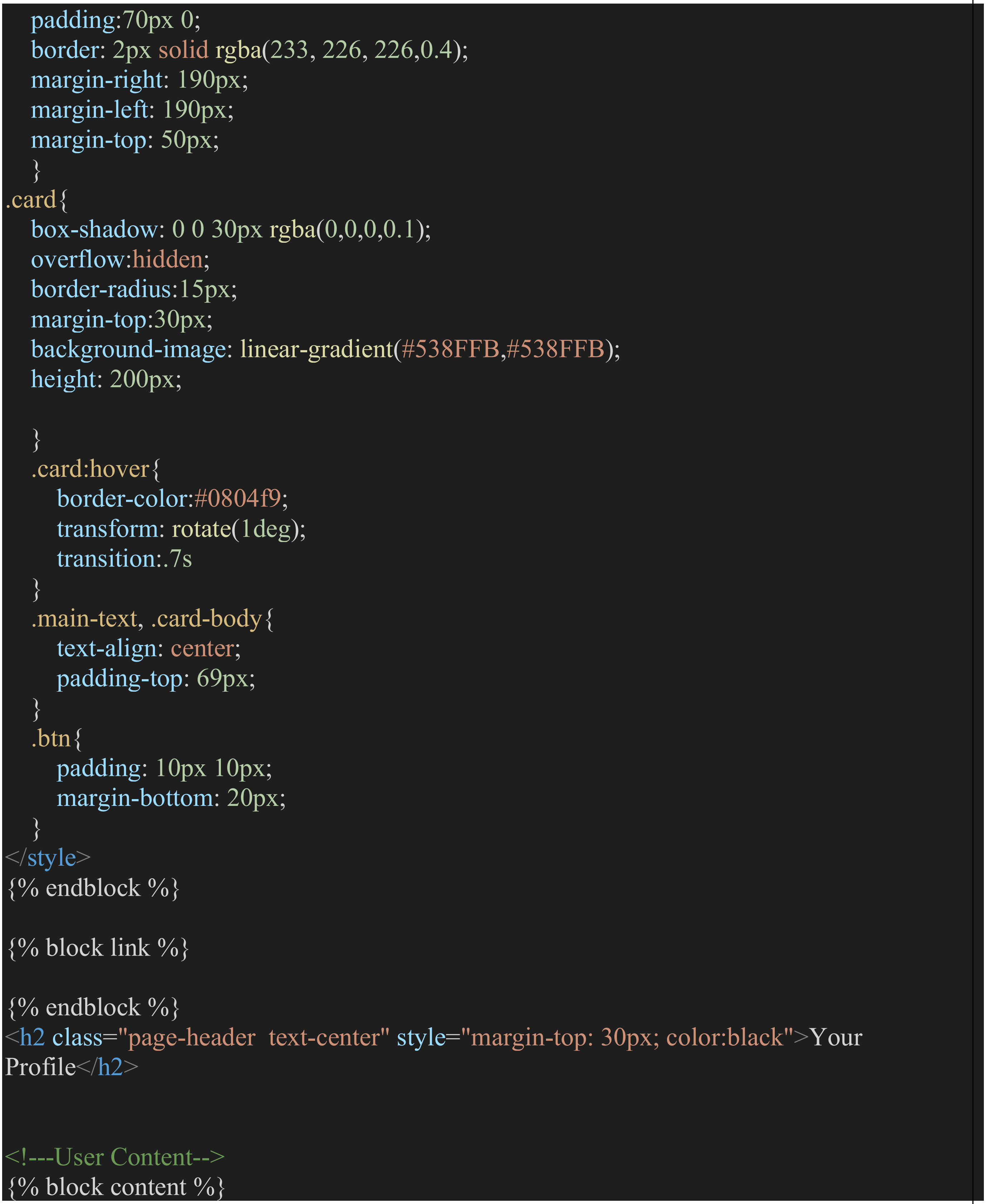
```
        </div>
      </div>
  </div>
</div>

{% endblock %}
```

Success.html

```
{% extends 'base.html'%}

<!--title tag-->
{% block title %}
<title>Plasma-Success</title>
{% endblock %}

{% block link %}
{% endblock %}

<!---User Content-->
{% block content %}



    <div class="container-fluid">
      <div class="text-center">

      <style>

      .hide {
        display: none;
      }
      .myDIV:hover + .hide {
        display: block;
        color: red;
      text-align: top;
      }


      </style>
      <div class="container">
      <div class="row p-1">
       <div class="col-sm-12">
        <div class="card" style="margin-top: 70px;">
         <div class="msg">{{ msg }}</div>

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

      <img src="https://objstorage.s3.jp-tok.cloud-object-
storage.appdomain.cloud/success.png" class="img-fluid" width="40%">

      </div> <div class="hide">You are a real superhero!</div>

          <h2>Thank you for donating plasma.</h2>
           </div>
              </div>
               </div>
                 </div>
                    </div></div> </div>


{% endblock%}
```

User_profile.html

```
{% extends 'base.html'%}

<!--title tag-->
{% block title %}
<title>Plasma-Profile</title>
<style>
  .profile-area{
```

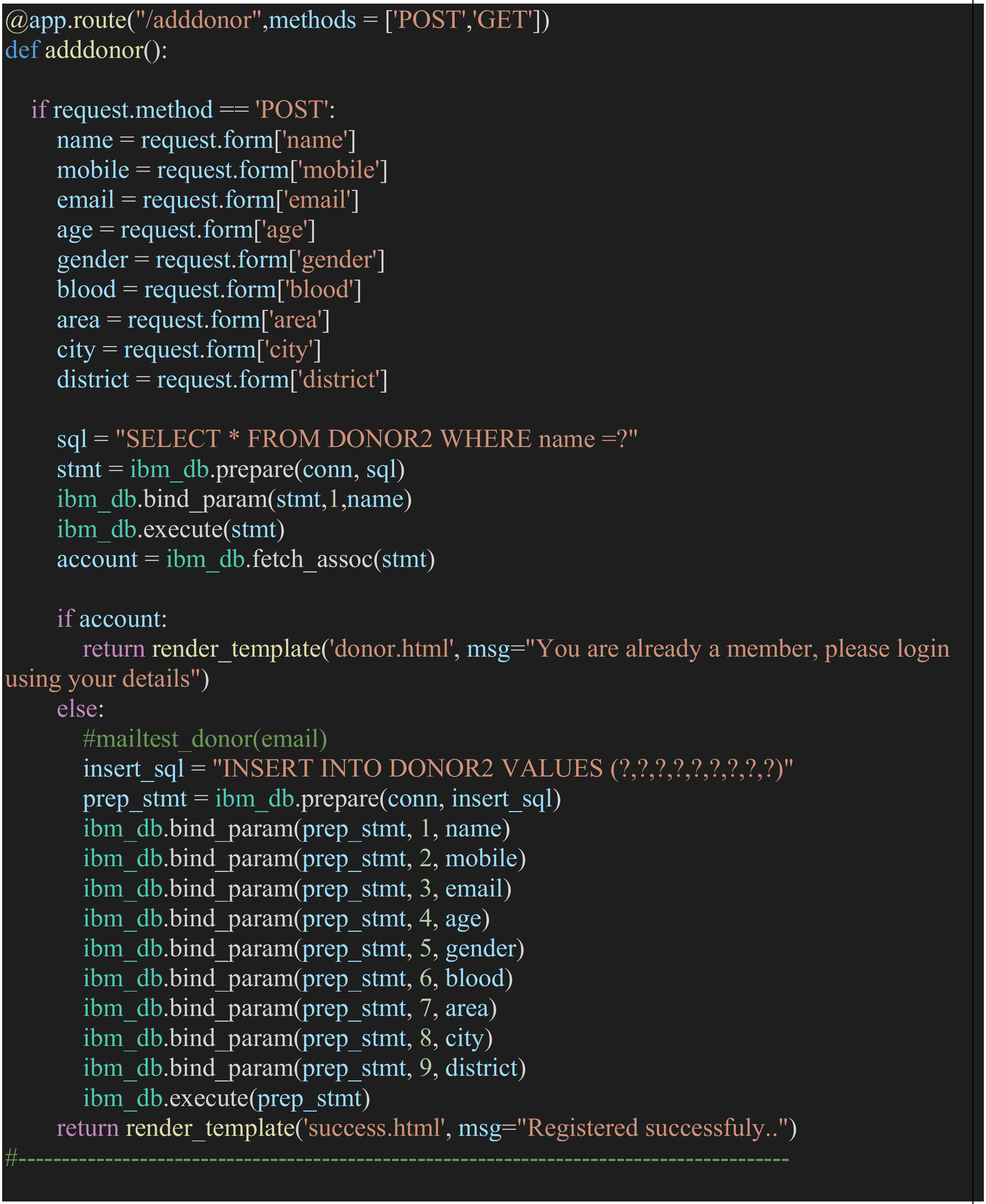


```
   padding:70px 0;
   border: 2px solid rgba(233, 226, 226,0.4);
   margin-right: 190px;
   margin-left: 190px;
   margin-top: 50px;
   }
.card{
   box-shadow: 0 0 30px rgba(0,0,0,0.1);
   overflow:hidden;
   border-radius:15px;
   margin-top:30px;
   background-image: linear-gradient(#538FFB,#538FFB);
   height: 200px;

   }
   .card:hover{
      border-color:#0804f9;
      transform: rotate(1deg);
      transition:.7s
   }
   .main-text, .card-body{
      text-align: center;
      padding-top: 69px;
   }
   .btn{
      padding: 10px 10px;
      margin-bottom: 20px;
   }
</style>
{% endblock %}

{% block link %}

{% endblock %}
<h2 class="page-header  text-center" style="margin-top: 30px; color:black">Your
Profile</h2>


<!---User Content-->
{% block content %}
```

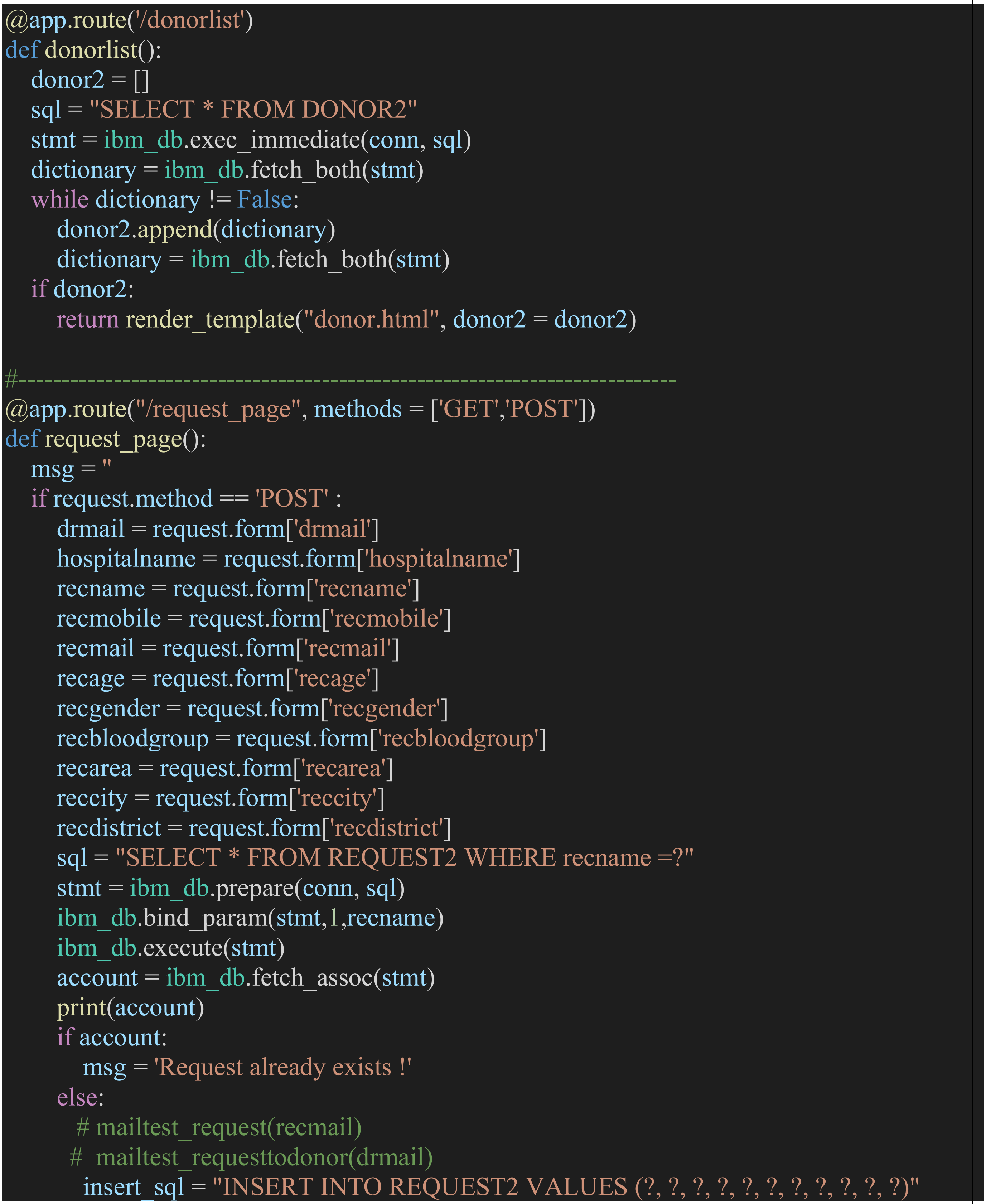
```
<div class="container">
<h3 class="text-danger" style="margin-top: 50px;">Welcome :
{{session["username"]}}!!</h3>
</div>
<div class = "profile-area">
<div class = "container">
   <div class="msg">{{ msg }}</div>
   <div class="row">

   <div class = "col-12 col-md-6 col-lg-6">
      <div class = "card">
      <div class = "main-text card-body">
         <div><a href="{{ url_for('register') }}" class="btn btn-light">Donate
Plasma</a></div>
      </div>
      </div>
   </div>
   <div class = "col-12 col-md-6 col-lg-6">
      <div class = "card">
      <div class = "main-text card-body">
         <div><a href="{{ url_for('donorlist') }}" class="btn btn-light">Request
Plasma</a></div>
      </div>
      </div>
   </div>
   </div>
   </div>
   </div><br>
   <div class="container">
   <a href="{{url_for('logout')}}" class="btn btn-danger">Log Out</a></div>



{% endblock%}
```

App.py

```
from flask import Flask,render_template,request,url_for,flash,redirect,session
import ibm_db
import sendgrid
import re
from sendgrid.helpers.mail import *
from mail import Mail

send = Mail()


app = Flask(__name__)
print(__name__)
app.secret_key="1"

conn = ibm_db.connect


@app.route("/")
def index():
    return render_template('home.html')

@app.route("/home")
def home_page():
    return render_template('home.html')
#-------------------------------------------------------

@app.route("/login",methods = ['POST', 'GET'])
def login():
    global userid
    msg = ''
    if request.method == 'POST' :
        username = request.form['username']
        password = request.form['password']
        sql = "SELECT * FROM LOGIN WHERE username =? AND password=?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.bind_param(stmt,2,password)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print (account)
        if account:
            session['loggedin'] = True
            session['id'] = account['USERNAME']
            userid=  account['USERNAME']
            session['username'] = account['USERNAME']
            msg = 'Logged in successfully !'

            return render_template('user_profile.html', msg = msg)
        else:
            msg = 'Incorrect username / password !'
    return render_template('login.html', msg = msg)

#------------------------------------------------------
# After login
@app.route('/afterlogin')
def afterlogin():
    return render_template("user_profile.html")

#----------------------------------------------------

@app.route("/signin",methods = ['POST', 'GET'])
def signin():
    msg = ''
    if request.method == 'POST' :
        username = request.form['username']
        usermail = request.form['usermail']
        usercontact = request.form['usercontact']
        password = request.form['password']
        sql = "SELECT * FROM LOGIN WHERE username =?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
            msg = 'Account already exists !'
        elif not re.match(r'[^@]+@[^@]+\.[^@]+', usermail):
            msg = 'Invalid email address !'
        elif not re.match(r'[A-Za-z0-9]+', username):
            msg = 'name must contain only characters and numbers !'
        else:
            mailtest_registration(usermail)
            insert_sql = "INSERT INTO LOGIN VALUES (?, ?, ?, ?)"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prep_stmt, 1, username)
            ibm_db.bind_param(prep_stmt, 2, usermail)
            ibm_db.bind_param(prep_stmt, 3, usercontact)
            ibm_db.bind_param(prep_stmt, 4, password)
            ibm_db.execute(prep_stmt)
            msg = 'You have successfully registered !'
            # mailtest(usermail)
            return render_template('login.html', msg = msg)
    elif request.method == 'POST':
        msg = 'Please fill out the form !'

    return render_template('signin.html', msg = msg)
#-----------------------------------------------------------------
# sendgrid integration
def mailtest_registration(to_email):
    # sg = sendgrid.SendGridAPIClient(api_key= 'apikey' )
    # from_email = Email("chinnukool72@gmail.com")
    # subject = "Registration Successfull!"
    # content = Content("text/plain", "You have successfully registered as user. Please Login using your Username and Password to donate/request for Plasma.")
    # mail = Mail(from_email, to_email, subject, content)
    # response = sg.client.mail.send.post(request_body=mail.get())
    # print(response.status_code)
    # print(response.body)
    # print(response.headers)

    send.sendEmail('Plasma Donor Application', 'You are successfully registered as a user', to_email)


#----------------------------------------------------------------------------
@app.route("/register")
def register():
    return render_template('register.html')
```



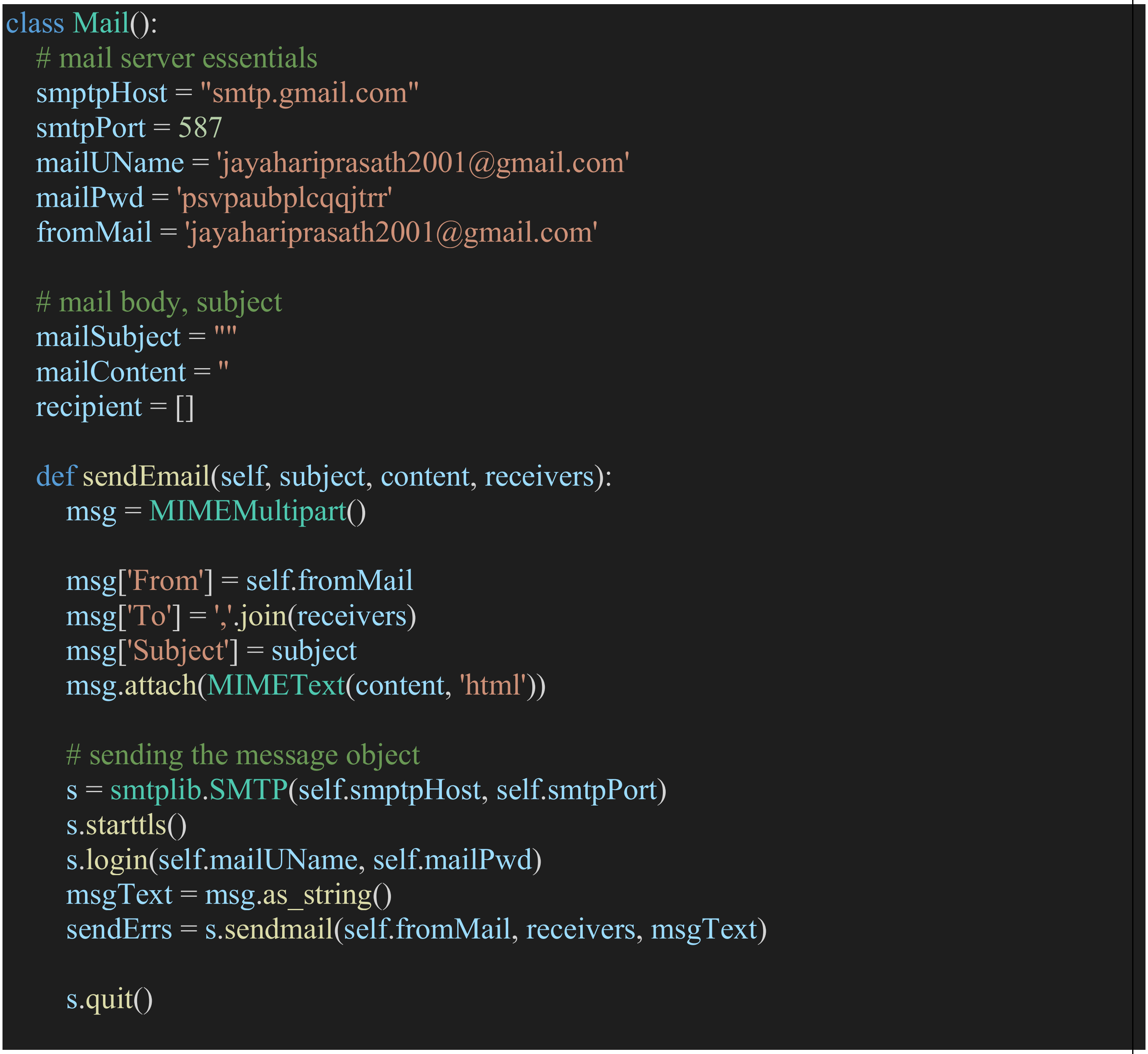
```
@app.route("/adddonor",methods = ['POST','GET'])
def adddonor():

    if request.method == 'POST':
        name = request.form['name']
        mobile = request.form['mobile']
        email = request.form['email']
        age = request.form['age']
        gender = request.form['gender']
        blood = request.form['blood']
        area = request.form['area']
        city = request.form['city']
        district = request.form['district']

        sql = "SELECT * FROM DONOR2 WHERE name =?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,name)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)

        if account:
            return render_template('donor.html', msg="You are already a member, please login
using your details")
        else:
            #mailtest_donor(email)
            insert_sql = "INSERT INTO DONOR2 VALUES (?,?,?,?,?,?,?,?,?)"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prep_stmt, 1, name)
            ibm_db.bind_param(prep_stmt, 2, mobile)
            ibm_db.bind_param(prep_stmt, 3, email)
            ibm_db.bind_param(prep_stmt, 4, age)
            ibm_db.bind_param(prep_stmt, 5, gender)
            ibm_db.bind_param(prep_stmt, 6, blood)
            ibm_db.bind_param(prep_stmt, 7, area)
            ibm_db.bind_param(prep_stmt, 8, city)
            ibm_db.bind_param(prep_stmt, 9, district)
            ibm_db.execute(prep_stmt)
        return render_template('success.html', msg="Registered successfuly..")
#--------------------------------------------------------------------------------
```



```
@app.route('/donorlist')
def donorlist():
    donor2 = []
    sql = "SELECT * FROM DONOR2"
    stmt = ibm_db.exec_immediate(conn, sql)
    dictionary = ibm_db.fetch_both(stmt)
    while dictionary != False:
        donor2.append(dictionary)
        dictionary = ibm_db.fetch_both(stmt)
    if donor2:
        return render_template("donor.html", donor2 = donor2)

#------------------------------------------------------------------------------
@app.route("/request_page", methods = ['GET','POST'])
def request_page():
    msg = ''
    if request.method == 'POST' :
        drmail = request.form['drmail']
        hospitalname = request.form['hospitalname']
        recname = request.form['recname']
        recmobile = request.form['recmobile']
        recmail = request.form['recmail']
        recage = request.form['recage']
        recgender = request.form['recgender']
        recbloodgroup = request.form['recbloodgroup']
        recarea = request.form['recarea']
        reccity = request.form['reccity']
        recdistrict = request.form['recdistrict']
        sql = "SELECT * FROM REQUEST2 WHERE recname =?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,recname)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
            msg = 'Request already exists !'
        else:
           # mailtest_request(recmail)
          #  mailtest_requesttodonor(drmail)
            insert_sql = "INSERT INTO REQUEST2 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prep_stmt, 1, drmail)
            ibm_db.bind_param(prep_stmt, 2, hospitalname)
            ibm_db.bind_param(prep_stmt, 3, recname)
            ibm_db.bind_param(prep_stmt, 4, recmobile)
            ibm_db.bind_param(prep_stmt, 5, recmail)
            ibm_db.bind_param(prep_stmt, 6, recage)
            ibm_db.bind_param(prep_stmt, 7, recgender)
            ibm_db.bind_param(prep_stmt, 8, recbloodgroup)
            ibm_db.bind_param(prep_stmt, 9, recarea)
            ibm_db.bind_param(prep_stmt, 10, reccity)
            ibm_db.bind_param(prep_stmt, 11, recdistrict)
            ibm_db.execute(prep_stmt)
            msg = 'Your request has been submitted!'
            return render_template('request.html', msg = msg)
    elif request.method == 'POST':
        msg = 'Please fill out the form !'

    return render_template('request.html', msg = msg)


#-------------------------------------------
@app.route('/logout')
def logout():
    session.clear()
    return redirect(url_for("index"))

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5000, debug=True)
```

mail.py

```
import smtplib
from email.mime.text import MIMEText
from email.mime.base import MIMEBase
from email.mime.multipart import MIMEMultipart
from email import encoders
import os
```



```
class Mail():
    # mail server essentials
    smptpHost = "smtp.gmail.com"
    smtpPort = 587
    mailUName = 'jayahariprasath2001@gmail.com'
    mailPwd = 'psvpaubplcqqjtrr'
    fromMail = 'jayahariprasath2001@gmail.com'

    # mail body, subject
    mailSubject = ""
    mailContent = ''
    recipient = []

    def sendEmail(self, subject, content, receivers):
        msg = MIMEMultipart()

        msg['From'] = self.fromMail
        msg['To'] = ','.join(receivers)
        msg['Subject'] = subject
        msg.attach(MIMEText(content, 'html'))

        # sending the message object
        s = smtplib.SMTP(self.smptpHost, self.smtpPort)
        s.starttls()
        s.login(self.mailUName, self.mailPwd)
        msgText = msg.as_string()
        sendErrs = s.sendmail(self.fromMail, receivers, msgText)

        s.quit()
```

13.2 GitHub & Project Demo Link

GitHub link: https://github.com/IBM-EPBL/IBM-Project-39532-1660455806

Project Demo link: **https://youtu.be/6X5jZXc9szI**