PLASMA DONOR APPLICATION

Team id	**PNT2022TMID49331**
Project Name	Plasma Donor Application

Team Members	P.SARAVANAN(923119104015)
	R.YASOTHA(923119104020),
	M.VANITHA(923119104019),
	C.PRASANTH(923119104012)

Table Of Contents

INTRODUCTION

1.1 PROJECT OVERVIEW:

The main goal of our project is to design a user-friendly web application that is like a scientific vehicle from which we can help reduce mortality or help those affected by COVID19 by donating plasma from patients who have recovered without approved antiretroviral therapy planning for a deadly COVID19 infection, plasma therapy is an experimental approach to treat those COVID-positive patients and help them recover faster.

Therapy, which is considered reliable and safe. If a particular person has fully recovered from COVID19, they are eligible to donate their plasma. As we all know, the traditional methods of finding plasma, one has to find out for oneself by looking at hospital records and contacting donors have been recovered, sometimes may not be available at home and move to other places. In this type of scenario, the health of those who are sick becomes disastrous. Therefore, it is not considered a rapid process to find plasma.

1.2 PURPOSE:

During the COVID 19 crisis, the requirement of plasma became a high priority and the donor count has become low.

The Purpose of this Application is Saving the donor information and helping the needy by notifying the current donors list, would be a helping hand. In regard to the problem faced, This application is to be built which would take the donor details, store them and inform them upon a request.

2 LITERATURE SURVEY

2.1 EXISTING PROBLEM:

- Cannot Upload and Download the latest updates.
- No use of Web Services and Remoting.
- Risk of mismanagement and of data when the project is under development.
- Less Security.
- No proper coordination between different Applications and Users.
- Fewer Users – Friendly

2.2 REFERENCE:

[1] R. C. Gojko Adzic, "Serverless computing: Economic and architectural impact," ESEC/FSE, 2017.

[2] P. C. P. C. a. V. I. M. Yan, "Building a chatbot with server less computing," IBM watson research center, 2016.

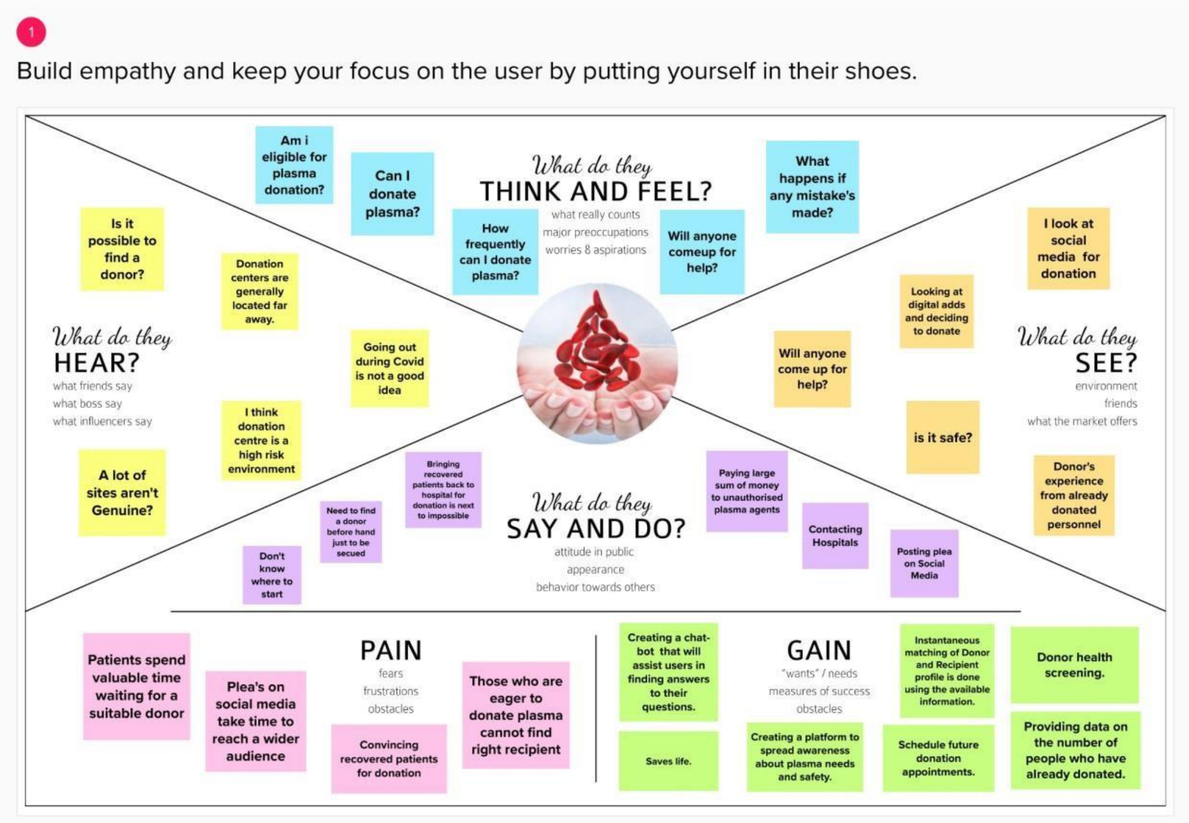
[3] S. E. a. B. J. J. Short, "“Cloud Event Programming Paradigms: Applications and Analysis,”," 9th IEEE International Conference on Cloud Computing (CLOUD), pp. pp. 400-406, 2017.

[4] Z. Al-Ali, "“Making Server less Computing More Server less,”," IEEE 11th International Conference on Cloud Computing (CLOUD), pp. pp. 456-459, 2018., 2018.

[5] A. S. a. S. Jindal, "“EMARS: Efficient Management and Allocation of Resources in Serverless,”," IEEE 11th International Conference on Cloud Computing (CLOUD), pp. pp. 827-830, 2018.

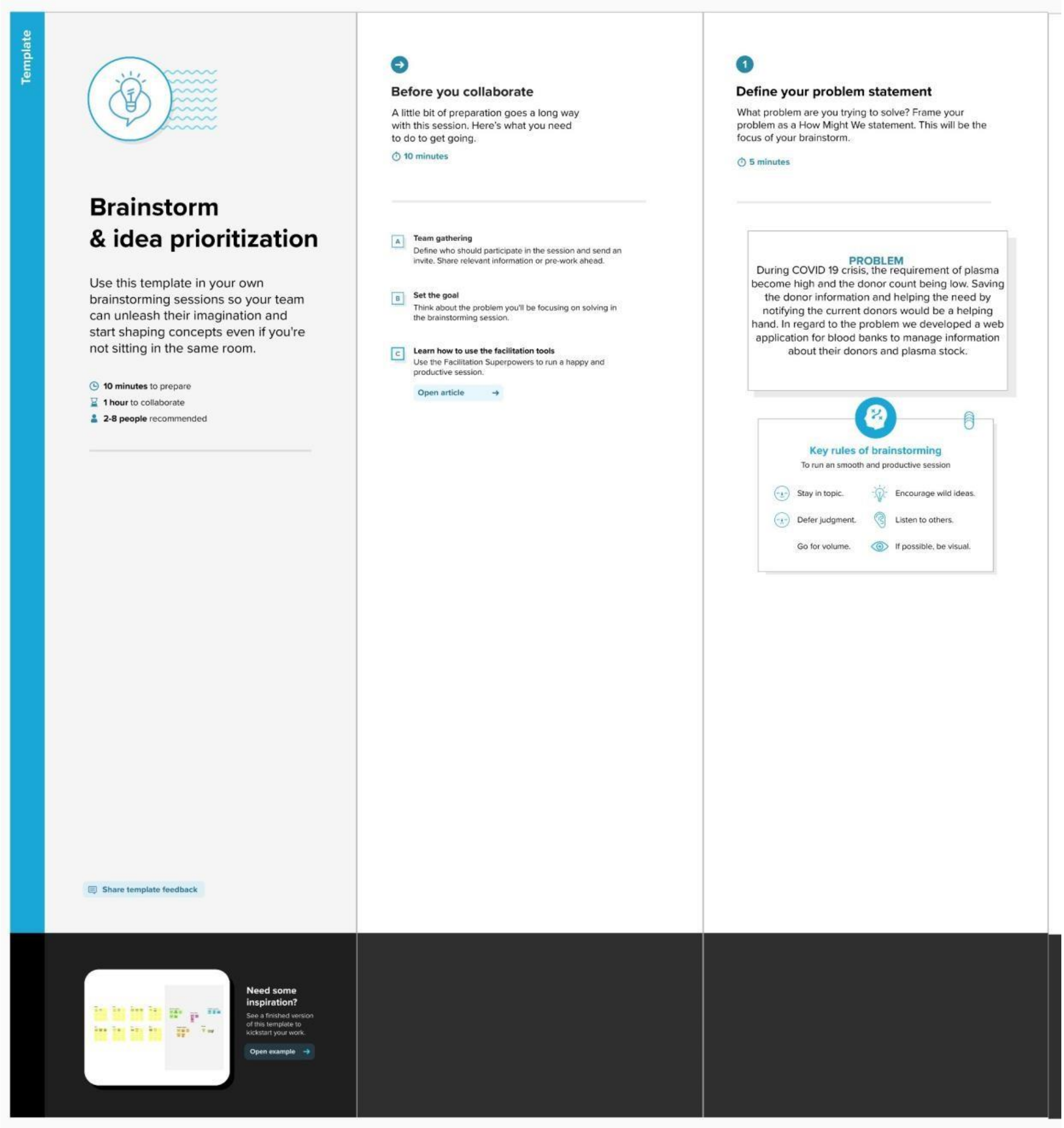
3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas:


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 8 aspirations
Am i eligible for plasma donation?
Can I donate plasma?
How frequently can I donate plasma?
Will anyone comeup for help?
What happens if any mistake's made?
What do they
HEAR?
what friends say
what boss say
what influencers say
Is it possible to find a donor?
Donation centers are generally located far away.
Going out during Covid is not a good idea
I think donation centre is a high risk environment
A lot of sites aren't Genuine?
What do they
SEE?
environment
friends
what the market offers
I look at social media for donation
Looking at digital adds and deciding to donate
Will anyone come up for help?
is it safe?
Donor's experience from already donated personnel
What do they
SAY AND DO?
attitude in public
appearance
behavior towards others
Don't know where to start
Need to find a donor before hand just to be secued
Bringing recovered patients back to hospital for donation is next to impossible
Paying large sum of money to unauthorised plasma agents
Contacting Hospitals
Posting plea on Social Media
PAIN
fears
frustrations
obstacles
Patients spend valuable time waiting for a suitable donor
Plea's on social media take time to reach a wider audience
Convincing recovered patients for donation
Those who are eager to donate plasma cannot find right recipient
GAIN
"wants" / needs
measures of success
obstacles
Creating a chat-bot that will assist users in finding answers to their questions.
Saves life.
Creating a platform to spread awareness about plasma needs and safety.
Instantaneous matching of Donor and Recipient profile is done using the available information.
Schedule future donation appointments.
Donor health screening.
Providing data on the number of people who have already donated.

3.2 Brainstorm & Idea Prioritization Template:

Step-1: Team Gathering, Collaboration and Select the Problem Statement



Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

- **10 minutes** to prepare
- **1 hour** to collaborate
- **2-8 people** recommended

Need some inspiration?

See a finished version of this template to kickstart your work.

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A. **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B. **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C. **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

1 Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

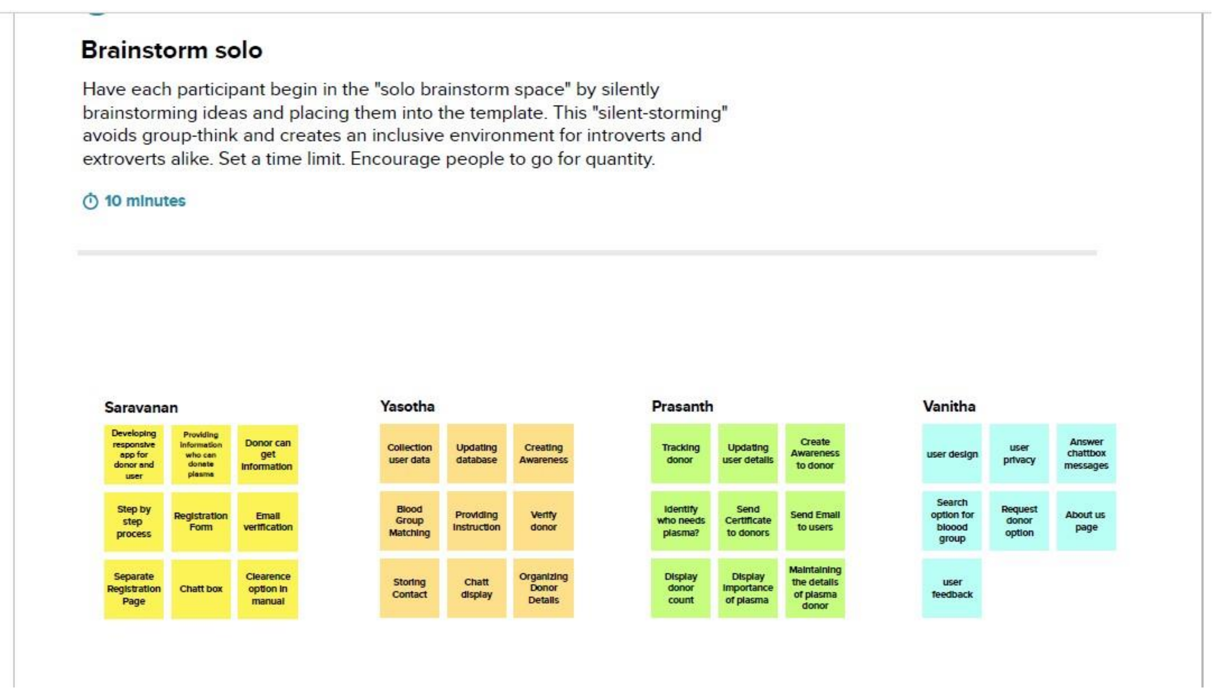
5 minutes

PROBLEM
During COVID 19 crisis, the requirement of plasma become high and the donor count being low. Saving the donor information and helping the need by notifying the current donors would be a helping hand. In regard to the problem we developed a web application for blood banks to manage information about their donors and plasma stock.

Key rules of brainstorming
To run an smooth and productive session

- Stay in topic.
- Encourage wild ideas.
- Defer judgment.
- Listen to others.
- Go for volume.
- If possible, be visual.

Step-2: Brainstorm, Idea Listing and Grouping



Brainstorm solo

Have each participant begin in the "solo brainstorm space" by silently brainstorming ideas and placing them into the template. This "silent-storming" avoids group-think and creates an inclusive environment for introverts and extroverts alike. Set a time limit. Encourage people to go for quantity.

⏱ 10 minutes

Saravanan

- Developing responsive app for donor and user
- Providing information who can donate plasma
- Donor can get information
- Step by step process
- Registration Form
- Email verification
- Separate Registration Page
- Chatt box
- Clearence option in manual

Yasotha

- Collection user data
- Updating database
- Creating Awareness
- Blood Group Matching
- Providing instruction
- Verify donor
- Storing Contact
- Chatt display
- Organizing Donor Details

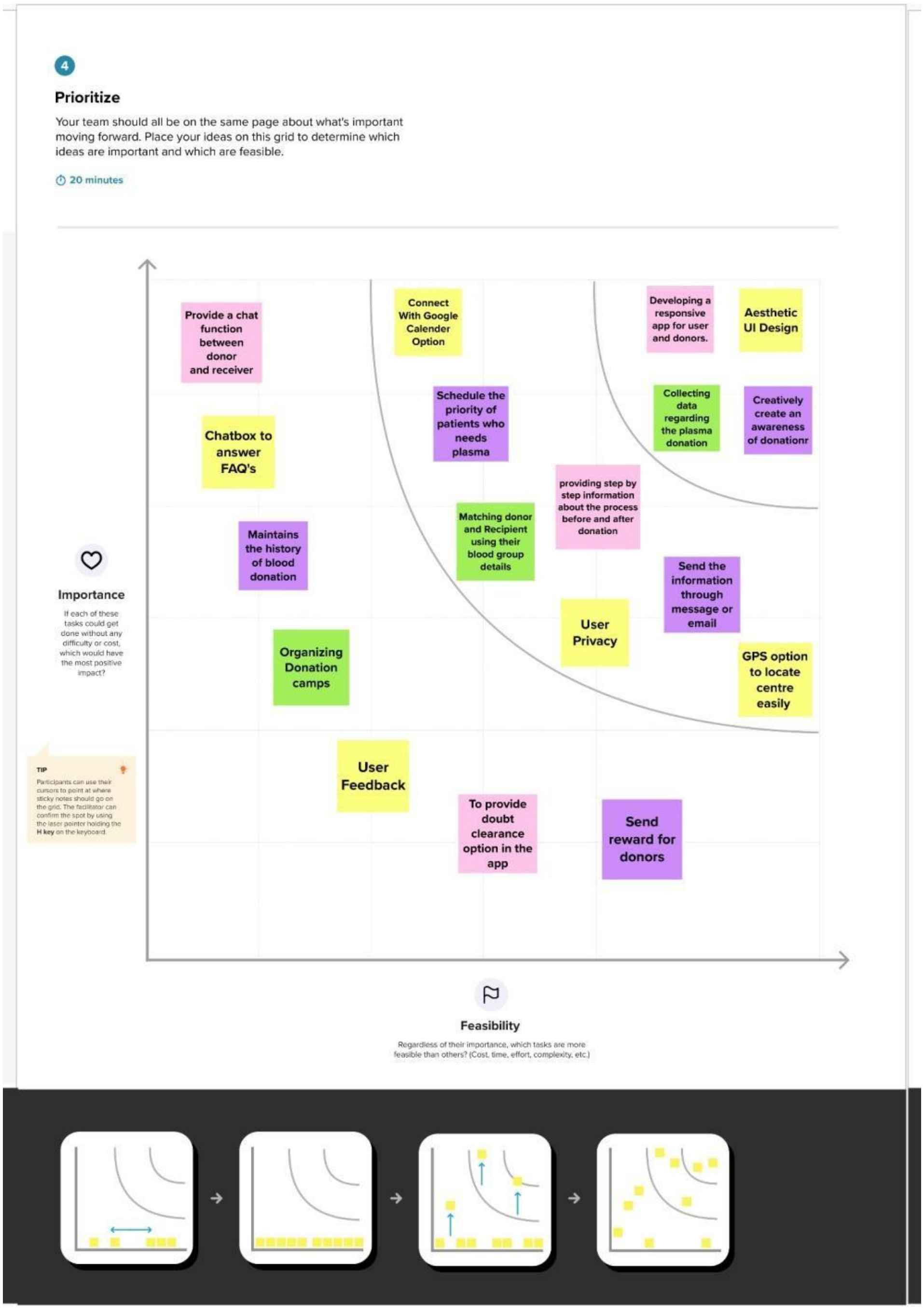
Prasanth

- Tracking donor
- Updating user details
- Create Awareness to donor
- Identify who needs plasma?
- Send Certificate to donors
- Send Email to users
- Display donor count
- Display importance of plasma
- Maintaining the details of plasma donor

Vanitha

- user design
- user privacy
- Answer chattbox messages
- Search option for bloood group
- Request donor option
- About us page
- user feedback

Step-3: Idea Prioritization



4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

Provide a chat function between donor and receiver

Connect With Google Calender Option

Developing a responsive app for user and donors.

Aesthetic UI Design

Chatbox to answer FAQ's

Schedule the priority of patients who needs plasma

Collecting data regarding the plasma donation

Creatively create an awareness of donationr

providing step by step information about the process before and after donation

Maintains the history of blood donation

Matching donor and Recipient using their blood group details

Send the information through message or email

User Privacy

Organizing Donation camps

GPS option to locate centre easily

User Feedback

To provide doubt clearance option in the app

Send reward for donors

Importance

If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

TIP

Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.

Feasibility

Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

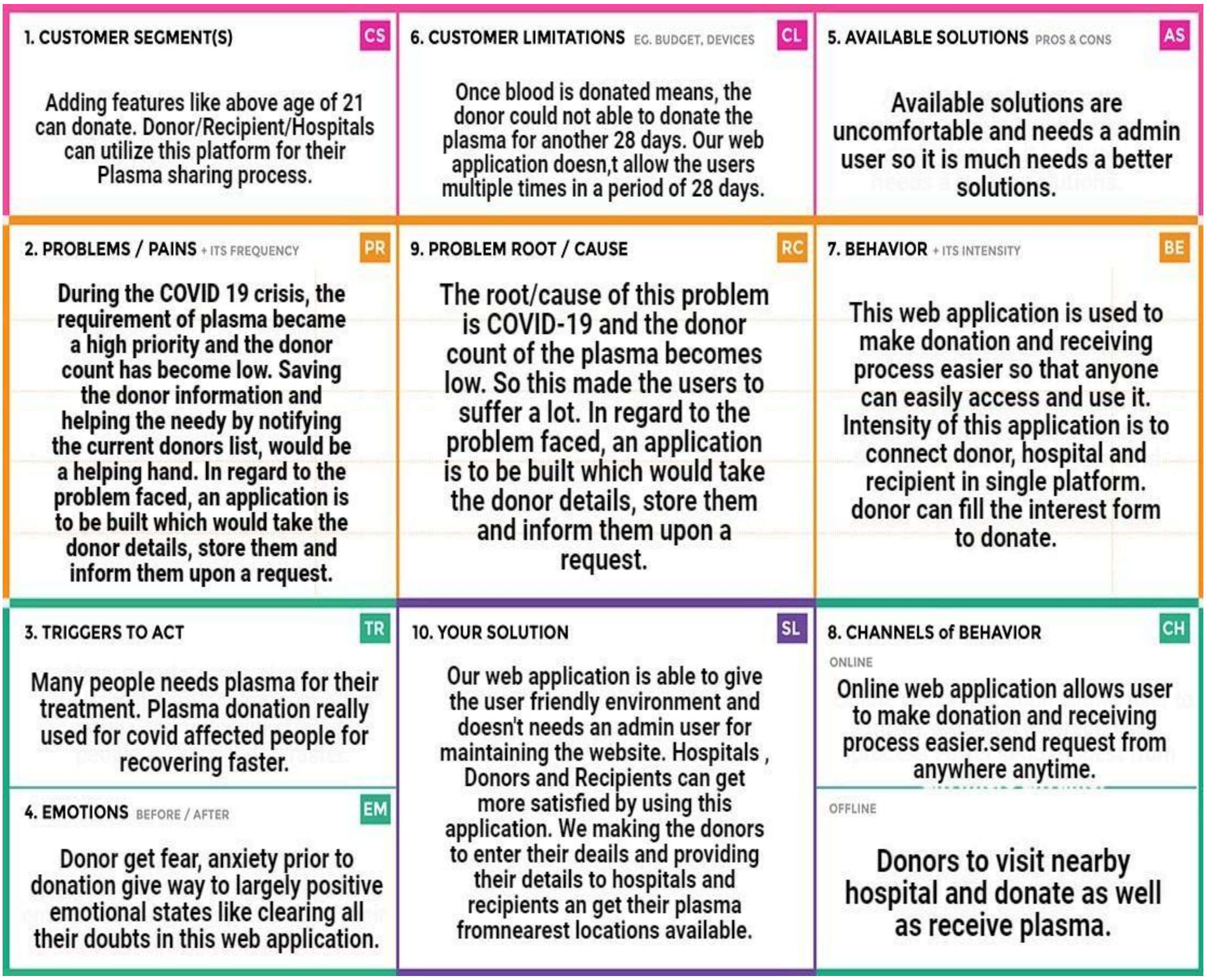
3.3 Proposed Solution Template:

Project team shall fill the following information in proposed solution template

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To help the plasma donor and seeker by developing a cloud-based application.
2.	Idea/Solution description	In day-to-day life requirement for plasma became high, especially during the COVID-19 crisis. But the donor count was low. Saving the donor information and helping the needy by notifying the current donors would be a helping hand. It is very difficult to find the respective blood group donors when anyone is in need. Regarding the problem faced, an application is to be built which would take the donor details store them and inform them upon request. And also for plasma donation centre, it is Easy to find donors.
3.	Novelty/ Uniqueness	We help the donor to access the location of a blood centre which is nearby him/her. We Notify them by sending a confirmation emails after they get registered for the plasma donation and alsowe notify them once the appointment is fixed in the centre. Further , more the GPS map option is available to direct The donor to the centre.
4.	Social Impact / Customer Satisfaction	By using this application, the user will experience a user-friendly and responsive interface and they get satisfaction by Saving thousand so people's life.

5.	Business Model(Revenue Model)	Donating Plasma with the help of an application makes our idea realistic. The user's information is encrypted. We maintain this app by automation for saving admin and user time. Users get profited as we take care of them even after the plasma donation by giving them hospitality details. Also, we use the Chabott answer FAQs ,asset helps the user to get immediate Answer to their doubts.
6.	Scalability of the Solution	Whatever the requirements, the application provides a clear solution for the requirements. It can handle more users who use the application at the same time .

3.4 PROBLEM SOLUTION FIT:



1. CUSTOMER SEGMENT(S) CS

Adding features like above age of 21 can donate. Donor/Recipient/Hospitals can utilize this platform for their Plasma sharing process.

6. CUSTOMER LIMITATIONS EG. BUDGET, DEVICES CL

Once blood is donated means, the donor could not able to donate the plasma for another 28 days. Our web application doesn,t allow the users multiple times in a period of 28 days.

5. AVAILABLE SOLUTIONS PROS & CONS AS

Available solutions are uncomfortable and needs a admin user so it is much needs a better solutions.

2. PROBLEMS / PAINS + ITS FREQUENCY PR

During the COVID 19 crisis, the requirement of plasma became a high priority and the donor count has become low. Saving the donor information and helping the needy by notifying the current donors list, would be a helping hand. In regard to the problem faced, an application is to be built which would take the donor details, store them and inform them upon a request.

9. PROBLEM ROOT / CAUSE RC

The root/cause of this problem is COVID-19 and the donor count of the plasma becomes low. So this made the users to suffer a lot. In regard to the problem faced, an application is to be built which would take the donor details, store them and inform them upon a request.

7. BEHAVIOR + ITS INTENSITY BE

This web application is used to make donation and receiving process easier so that anyone can easily access and use it. Intensity of this application is to connect donor, hospital and recipient in single platform. donor can fill the interest form to donate.

3. TRIGGERS TO ACT TR

Many people needs plasma for their treatment. Plasma donation really used for covid affected people for recovering faster.

4. EMOTIONS BEFORE / AFTER EM

Donor get fear, anxiety prior to donation give way to largely positive emotional states like clearing all their doubts in this web application.

10. YOUR SOLUTION SL

Our web application is able to give the user friendly environment and doesn't needs an admin user for maintaining the website. Hospitals , Donors and Recipients can get more satisfied by using this application. We making the donors to enter their deails and providing their details to hospitals and recipients an get their plasma fromnearest locations available.

8. CHANNELS of BEHAVIOR CH

ONLINE

Online web application allows user to make donation and receiving process easier.send request from anywhere anytime.

OFFLINE

Donors to visit nearby hospital and donate as well as receive plasma.

4. REQUIREMENT ANALYSIS:

4.1 FUNCTIONAL REQUIREMENTS:

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 (WebApp)
FR-2	User Confirmation	Confirmation via EmailConfirmation via OTP
FR-3	Certification	After the donor donates plasma, we will give them a certificate of appreciation and authentication.
FR-4	Statistical data	The availability of plasma is given in the page as stats, which will be helpful for the users.
FR-5	User Plasma Request	Users can request to donate plasma by filling out the request form on the page. Once the request is submitted, they will get an email
FR-6	Searching/reporting requirements	Users can use the search bar to look up information about camps and other topics.
FR-7	Virtual Assistants	A virtual assistant is a software agent that can carry out tasks or provide services on behalf of a person in response to commands or inquiries. When users enter their inquiries, the system will respond with pertinent information about plasma and details of plasma donation.

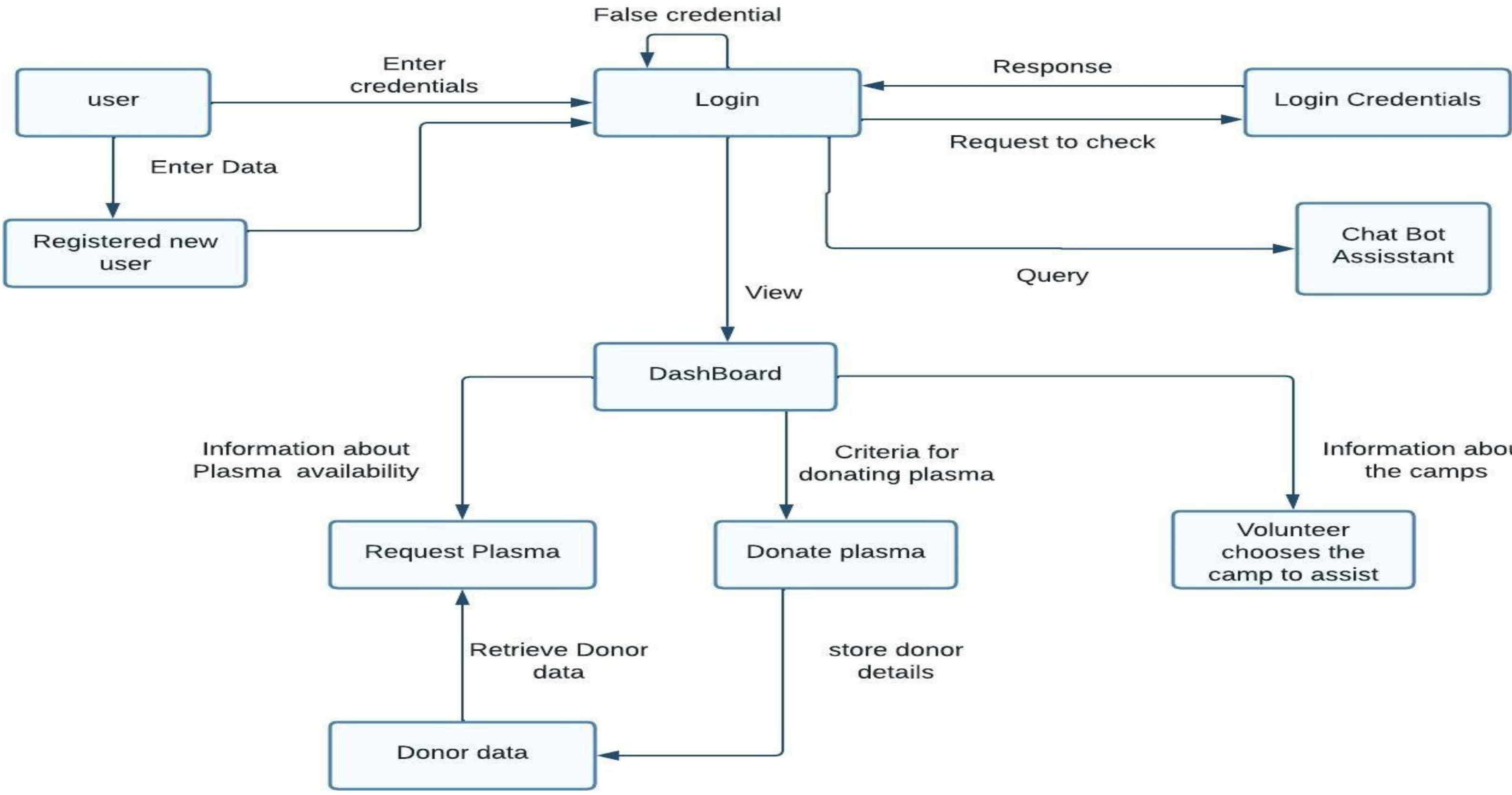
4.2 NON-FUNCTIONALREQUIREMENTS:

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

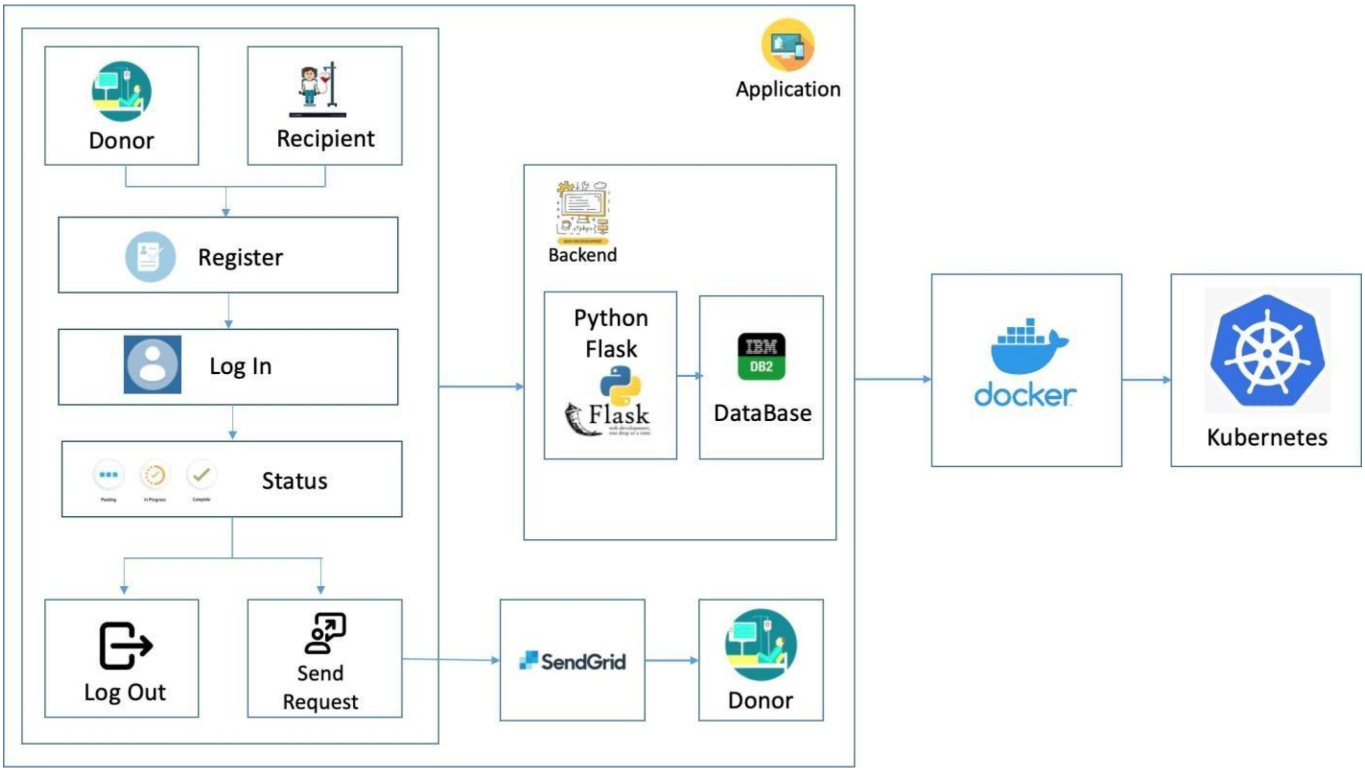
NFR No.	Non-Functional Requirement	Description
NFR-1	Usability	Must have a good-looking User-friendly interface.
NFR-2	Security	It must be secured with the proper username and password.
NFR-3	Reliability	The system should be made in such a way that it is reliable in its operations and for securing the sensitive details.

5. PROJECT DESIGN

5.1 Data Flow Diagram:



5.2 Solution & Technical Architecture:



5.3 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 my email, password, and confirmingmy password.	I can access my account /dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation emailonce I have registered for the application	I can receive confirmationemail & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Gmail	I can receive confirmation notifications through Gmail	Medium	Sprint-1
	Login	USN-4	As a user, I can log into the application by entering email & password	I can access into my User profile and view details indashboard	High	Sprint-1
	Dashboard	USN-5	As a user, I can send the proper requests todonate and obtain plasma.	I can receive appropriate notifications through email	High	Sprint-1
Customer (Web user)	Login	USN-6	As a user, I can register and log into the application by entering email & password toview the profile	I can access into my User profile and view details indashboard	High	Sprint-1
	Dashboard	USN-7	As a user, I can send the proper requests todonate and obtain plasma.	I can receive appropriate notifications through email	High	Sprint-1
Customer Care Executive	Application	USN-8	As a customer care executive, I can try to address user's concerns and questions	I can view and address their concerns	Medium	Sprint-2
Administrator	Application	USN-9	As an administrator I can help with user-facing aspects of a website, like its appearance, navigation and use of media.	I can change appearance friendly manner	Medium	Sprint-3

		USN-10	As an administrator, I can involve working withthe technical side of websites.	I can help with such as troubleshooting issues, setting up web hosts, ensuring users have access and programmingservers	Medium	Sprint-1

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	Plasma Donor Application	User Story / Task	Story Points	Priority	Team members
Sprint-1	Registration	PDA-1	As a user, I can register for the application by entering my Name, email, password, confirming my password, Age, BloodGroup.	3	High	Yasotha
Sprint-3		PDA-2	As a user, I will receive confirmation email once I have registered for the application	3	Medium	Prasanth
Sprint-2		PDA-3	As a user, I can register for the application through Gmail	5	Medium	Saravanan
Sprint-1	Login	PDA-4	As a user, I can log into the application by entering email and password	2	High	Yasotha, Saravanan
Sprint-3		PDA-5	As a user, I can reset my password using Forgot Password option	4	Medium	Prasanth
Sprint-4		PDA-6	As a user, I can view my past requests for plasma donation	3	Low	Vanitha
Sprint-4		PDA-7	As a user, I can close past requests I made for plasma	2	Low	Saravanan

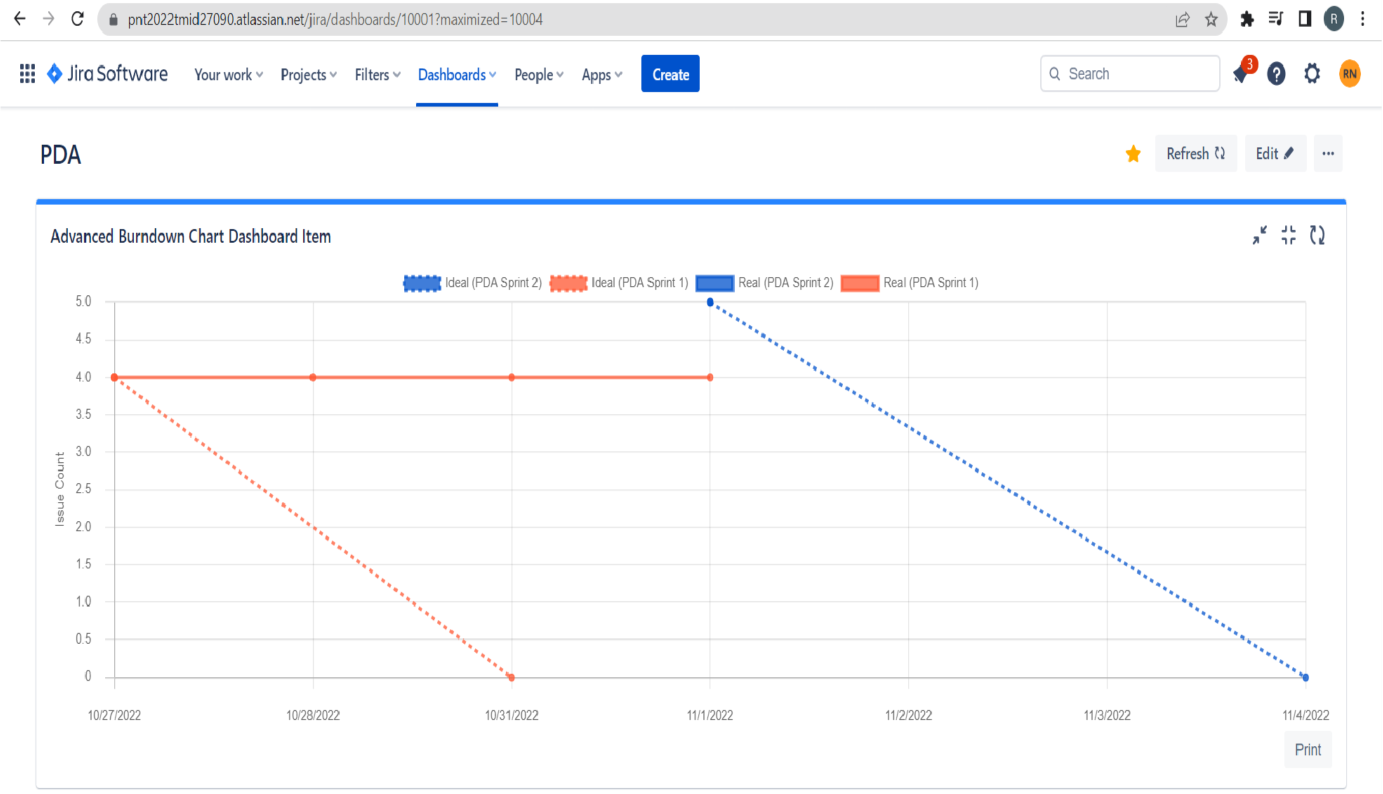
Sprint-1	Home Page	PDA-8	As a user, I can view the homepage of the website	2	Medium	Yasotha
Sprint - 1	About Page	PDA-9	As a user, I can view the about page on the website and get information related to Plasma Donation	2	Medium	Vanitha
Sprint - 2	Register as Donor	PDA-11	As a user, I can register as a donor by submitting a form and uploading certificate of recovery from Covid-19	3	High	Yasotha
Sprint	Functional Requirement(Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members

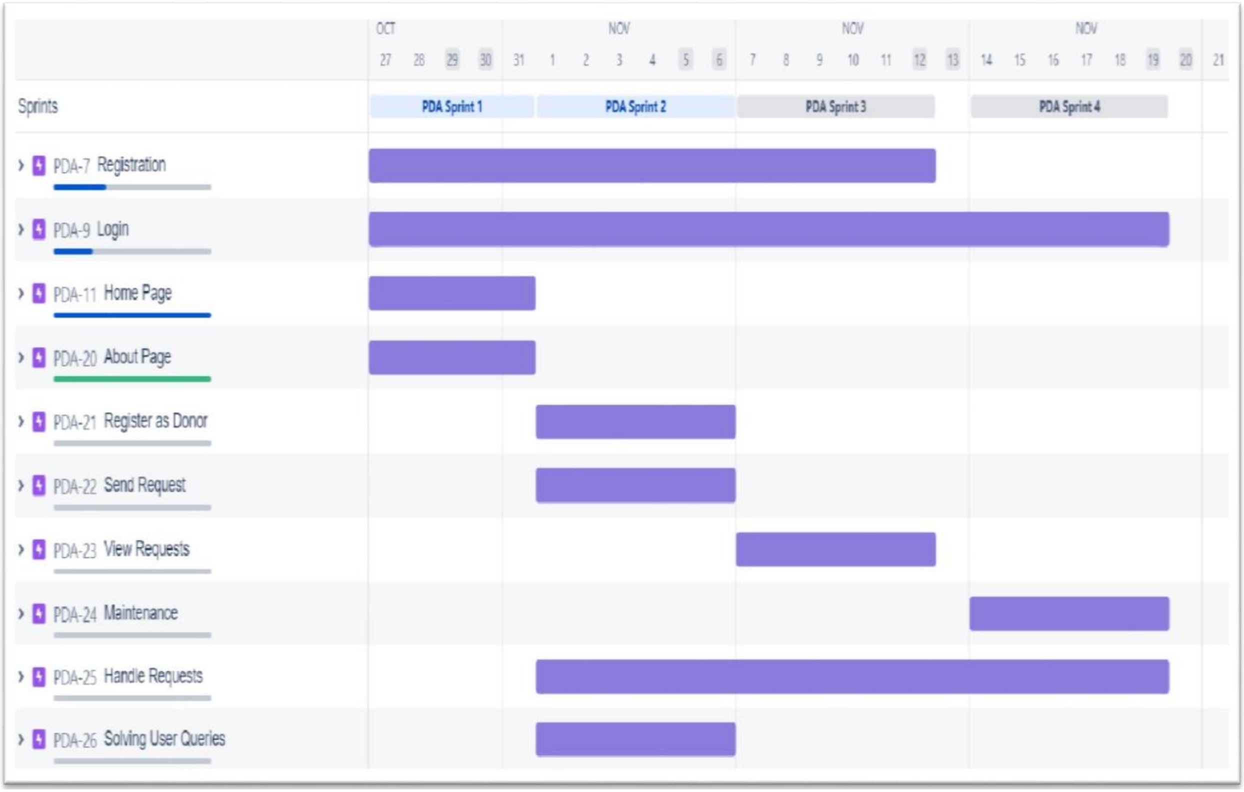
Sprint-2	Send Request	PDA-12	As a user, I can raise a request for plasma donation withspecific requirements through the request page.	2	High	Prasanth
Sprint-3	View Requests	PDA-13	As a user, I can view requests for plasma donation verified by admin	4	Medium	Saravanan
Sprint-4	Maintenance	PDA-14	As an admin, I can maintain the databases involved	2	Medium	Yasotha
Sprint-2	Handle Requests	PDA-15	As an admin, I can view all requests for plasma donation	1	High	Saravanan, Prasanth
Sprint-4		PDA-16	As an admin, I can delete requests that are past some timeperiod or have been closed	3	Low	Prasanth
Sprint-2	Solving User Queries	PDA-17	Creating a ChatBot that helps to solve the queries of the user.	2	High	Vanitha, Prasanth

6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	8	5 Days	27 Oct 2022	31 Nov 2022	8	03 Nov 2022
Sprint-2	13	4 Days	01 Nov 2022	06 Nov 2022	12	07 Nov 2022
Sprint-3	11	5 Days	07 Nov 2022	12 Nov 2022	11	09 Nov 2022
Sprint-4	9	5 Days	14 Nov 2022	19 Nov 2022	8	15 Nov 2022

6.3 Reports from JIRA





7 CODING & SOLUTIONING

7.1 FEATURE 1:

Python

It is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation.[33]

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming.

It is often described as a "batteries included" language due to its comprehensive standard library.[34][35]

Guido van Rossum began working on Python in the late 1980s as a successor to the ABC programming language and first released it in 1991 as Python 0.9.0.[36]

Python 2.0 was released in 2000 and introduced new features such as list comprehensions, cycle-detecting garbage collection, reference counting, and Unicode support. Python 3.0, released in 2008, was a major revision that is not completely backward-compatible with earlier versions. Python 2 was discontinued with version 2.7.18 in 2020.[37]

Python consistently ranks as one of the most popular programming languages

7.2 FEATURE 2:

Flask

Flask is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries.[2]

It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions.

However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

7.3 Database Schema

IBM Db2 -

a hybrid ANSI-compliant data virtualization tool for accessing, querying and summarizing data across the enterprise which:

- Provides a massively parallel processing (MPP) architecture Exploits Hive, HBase and Apache Spark concurrently for best-in-class analytic capabilities

- Requires only a single database connection or query to connect disparate sources such as HDFS, RDMS, NoSQL databases, object stores and Web HDFS

- Provides low latency support for ad-hoc and complex queries, high performance, and federation capabilities

- Understands dialects from other vendors and various products from Oracle, IBM® Db2® and IBM Netezza®

- Enables advanced row and column security

KUBERNATES-

Kubernetes — also known as “k8s” or “kube” — is a container orchestration platform for scheduling and automating the deployment, management, and scaling of containerized applications.

Kubernetes was first developed by engineers at Google before being open sourced in 2014. It is a descendant of Borg, a container orchestration platform used internally at Google. Kubernetes is Greek for *helmsman* or *pilot*, hence the helm in the Kubernetes logo (link resides outside IBM).

Today, Kubernetes and the broader container ecosystem are maturing into a general-purpose computing platform and ecosystem that rivals — if not surpasses — virtual machines (VMs) as the basic building blocks of modern cloud infrastructure and applications.

This ecosystem enables organizations to deliver a high-productivity Platform-as-a-Service (PaaS) that addresses multiple infrastructure-related and operations-related tasks and issues surrounding cloud-native development so that development teams can focus solely on coding and innovation.

8 TESTING

8.1 TESTING CASE:

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product.

It provides a way to check the functional it your components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectation and does not fail in an unacceptable manner.

There are various types of test. Each test type addresses a specific testing requirement

8.2 ACCEPTANCE TESTING

Acceptance Testing
UAT Execution & Report Submission

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the Plasma Donor Application project at the time of the release to User Acceptance Testing (UAT).

2 .Defect Analysis

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

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

3. 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	7	0	0	7
Client Application	51	0	0	51
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

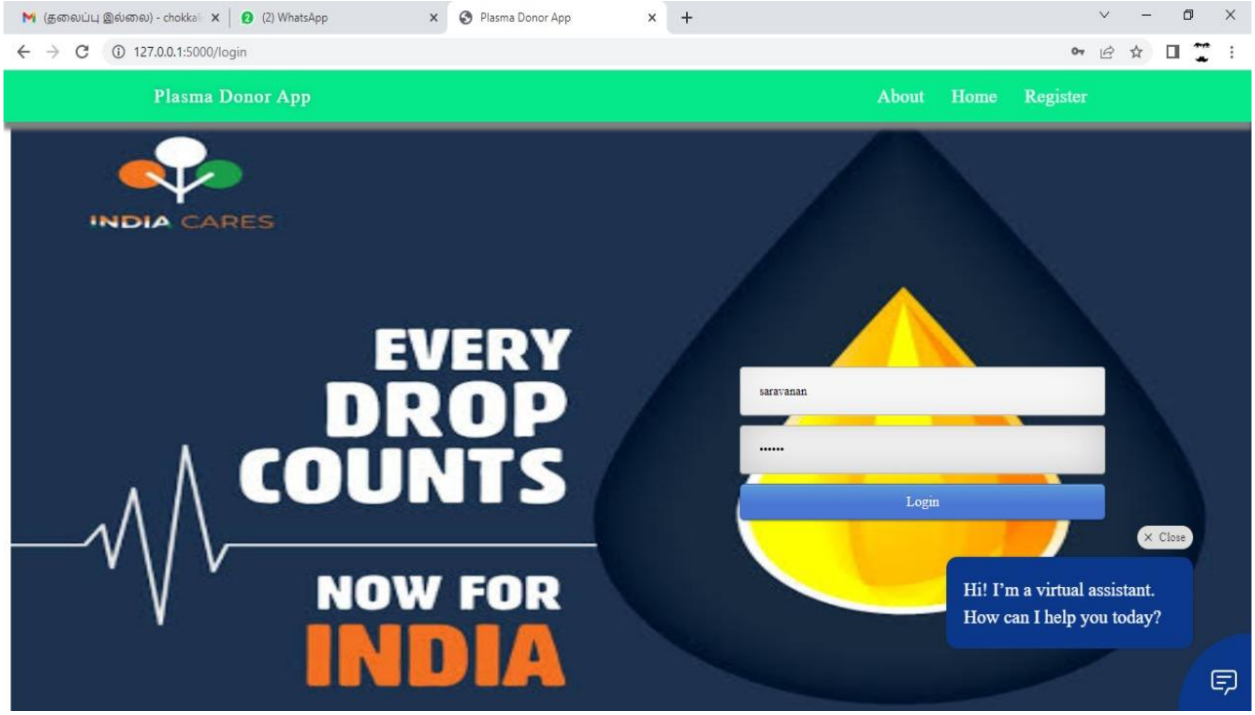
9 RESULTS

9.1 PERFORMANCE METRICS:

- Project metrics are used to track the progress and performance of a project.
- Monitoring parts of a project like **productivity, scheduling, and scope** make it easier for team leaders to see what's on track.
- As a project evolves, managers need access to changing deadlines or budgets to meet their client's expectations

OUTPUT SCREENS:

Login Page

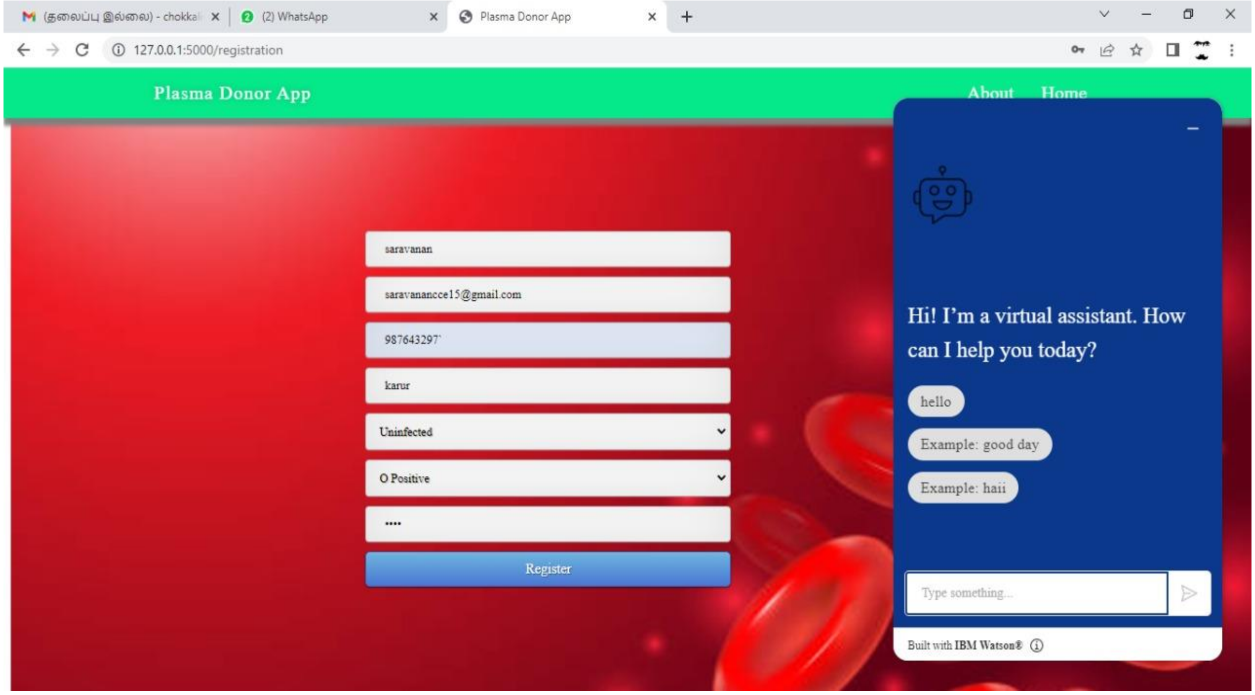


saravanan

••••••

Hi! I'm a virtual assistant.
How can I help you today?

Register Page:



saravanan

saravanancce15@gmail.com

987643297`

karur

Uninfected

O Positive

••••

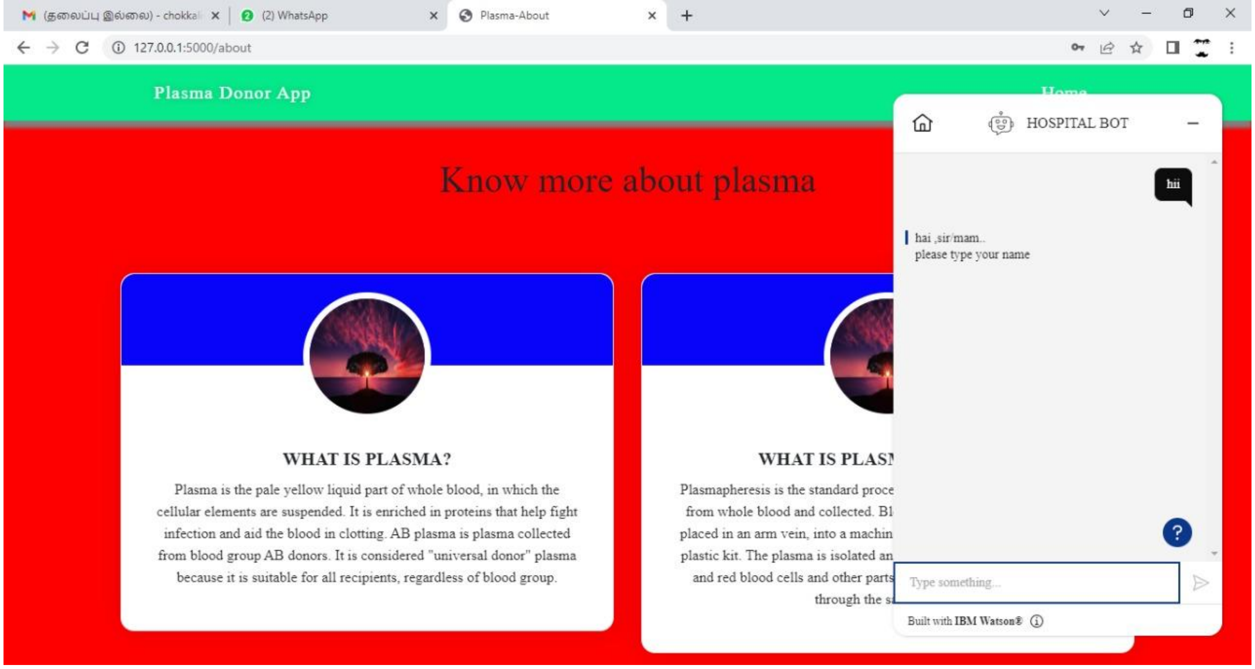
Hi! I’m a virtual assistant. How can I help you today?

hello

Example: good day

Example: haii

About us:



Know more about plasma

WHAT IS PLASMA?

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.

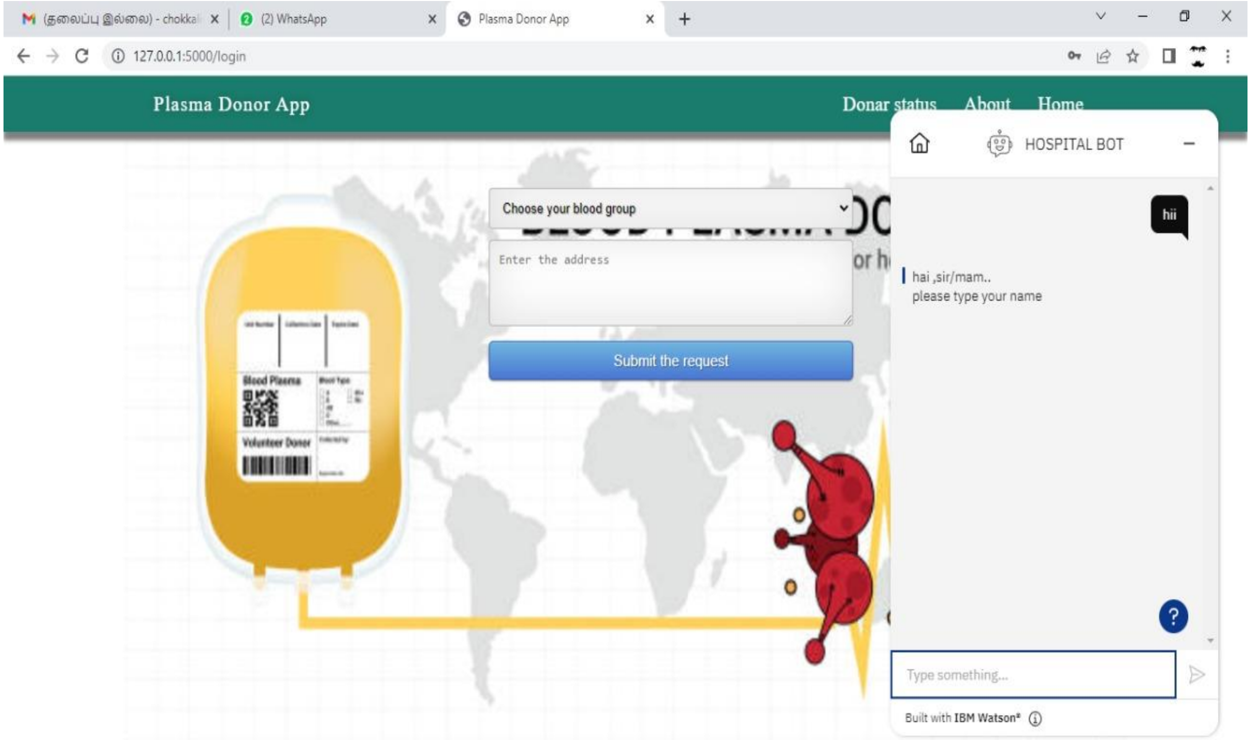
WHAT IS PLASI

Plasmapheresis is the standard proce
from whole blood and collected. Bl
placed in an arm vein, into a machin
plastic kit. The plasma is isolated an
and red blood cells and other parts
through the s

hii

hai ,sir/mam..
please type your name

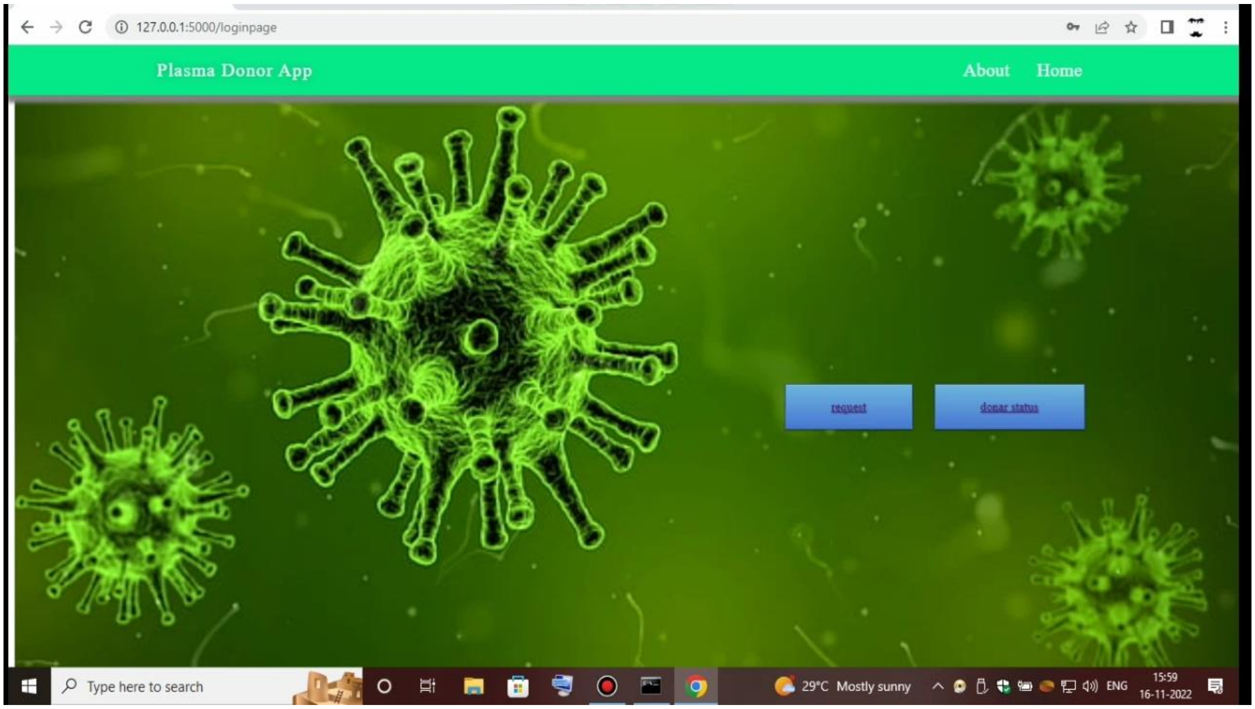
Request Page:



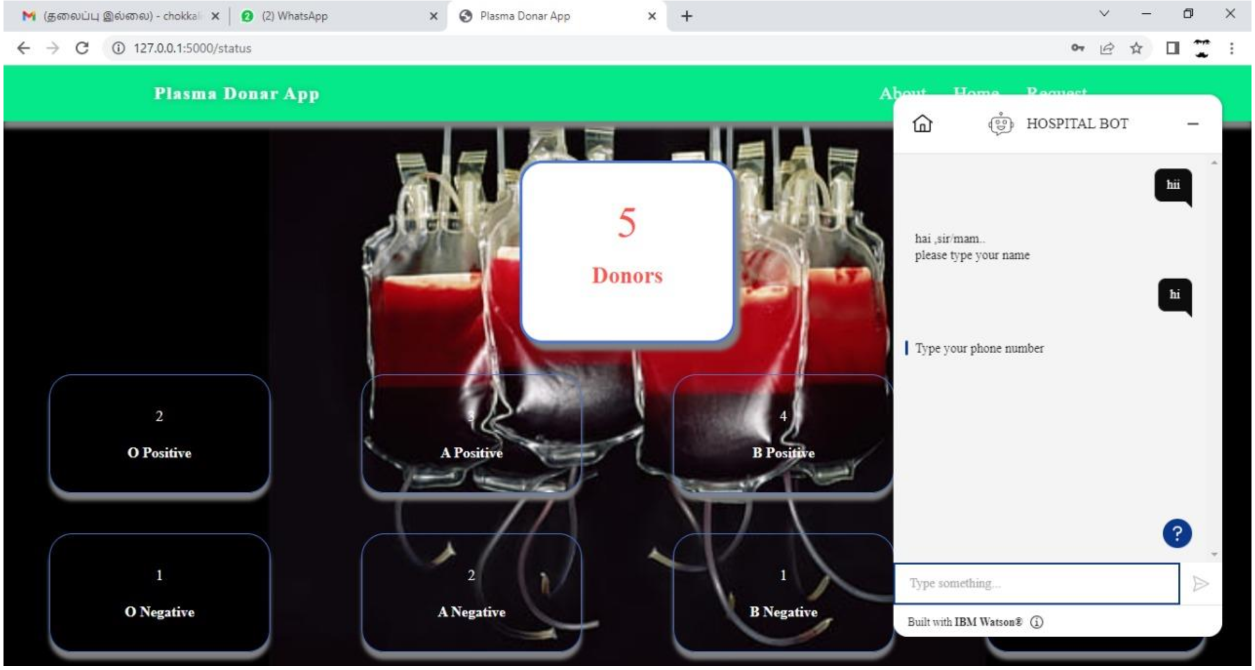
hii

hai ,sir/mam..
please type your name

Dashboard Page:



Plasma Donor Page



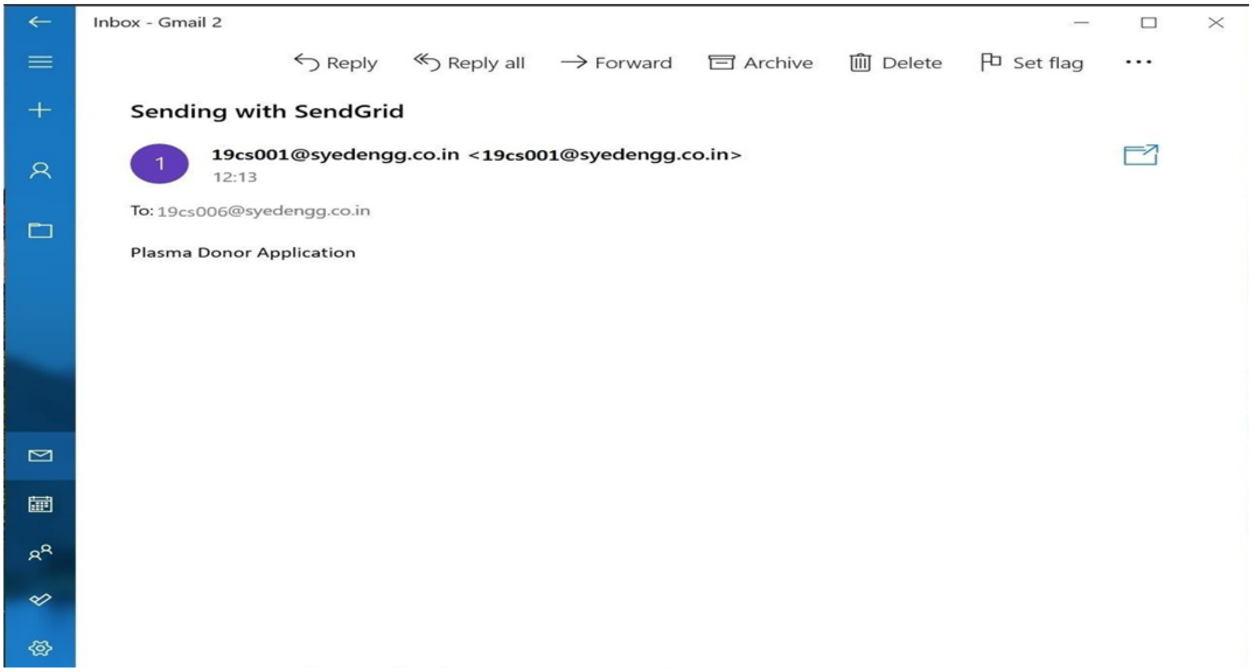
hii

hai ,sir/mam..
please type your name

hi

Type your phone number

Send grid:



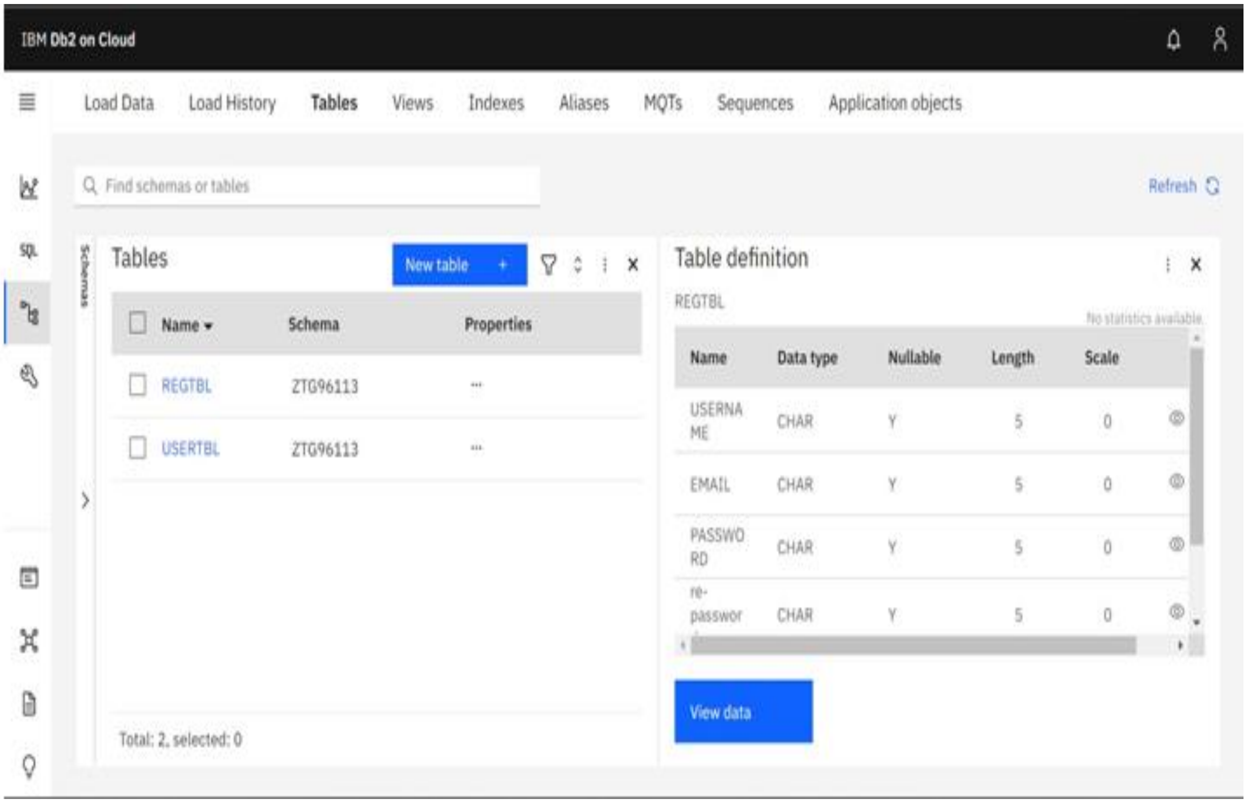
Sending with SendGrid

19cs001@syedengg.co.in <19cs001@syedengg.co.in>
12:13

To: 19cs006@syedengg.co.in

Plasma Donor Application

IBM Db 2



Tables

□ Name ▾	Schema	Properties
□ REGTBL	ZTG96113	···
□ USERTBL	ZTG96113	···

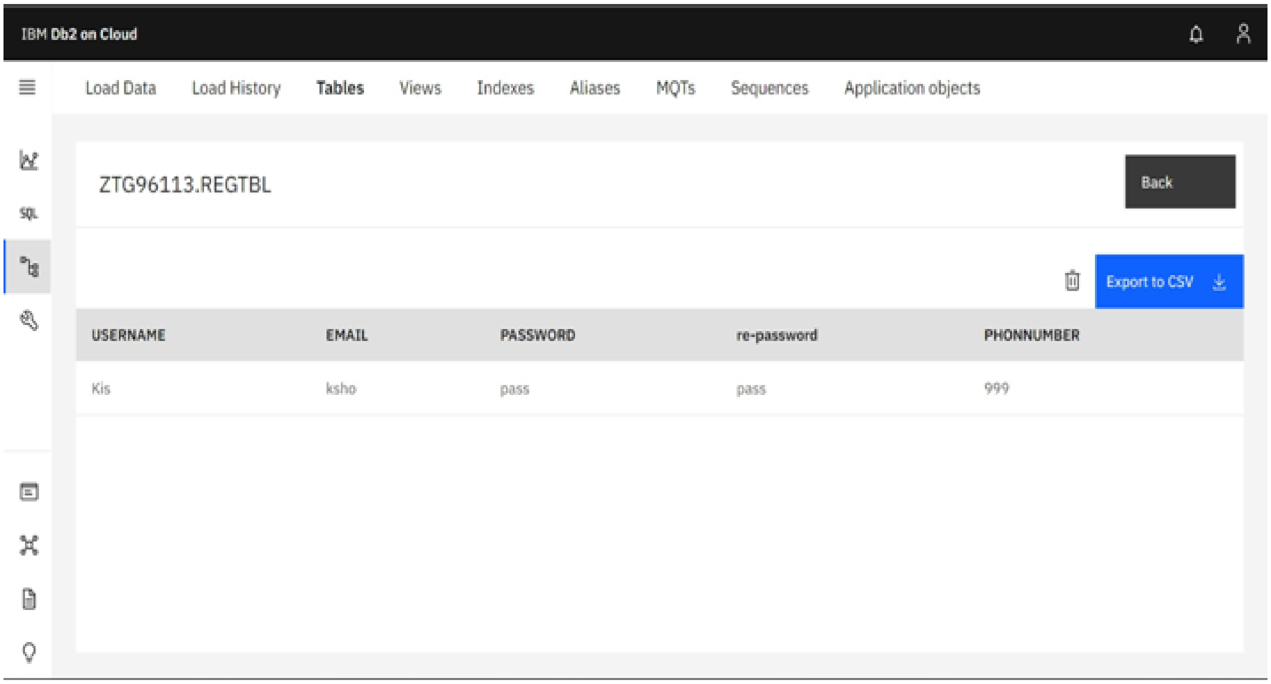
Total: 2, selected: 0

Table definition

REGTBL

No statistics available

Name	Data type	Nullable	Length	Scale
USERNAME	CHAR	Y	5	0
EMAIL	CHAR	Y	5	0
PASSWORD	CHAR	Y	5	0
re-passwor	CHAR	Y	5	0



ZTG96113.REGTBL

USERNAME	EMAIL	PASSWORD	re-password	PHONNUMBER
Kis	ksho	pass	pass	999

10 ADVANTAGES & DISADVANTAGES

ADVANTAGES:

- **Speed**: This website is fast and offers great accuracy as compared to manual registered keeping.
- **Maintenance** : Less maintenance is required
- **User Friendly**: It is very easy to use and understand. It is easily workable and accessible for everyone.
- **Fast Results**: It would help you to provide plasma donors easily depending upon the availability of it.

DISADVANTAGES:

- **Internet**: It would require an internet connection for the working of the website.
- **Auto- Verification**: It cannot automatically verify the genuine users.

.

11 CONCLUSIONS

The efficient way of finding plasma door for the infected people is implemented using the plasma donor website that is hosted on IBM Cloud platform.

To ensure the smooth functioning of the web site operation. I have hosted the website in IBM Db2 & Kubernates Cluster to make sure the operations are running successfully Cloud lambda function is used and to deploy the application IBM Db2 service is used.

12 FUTURE ENHANCEMENTS

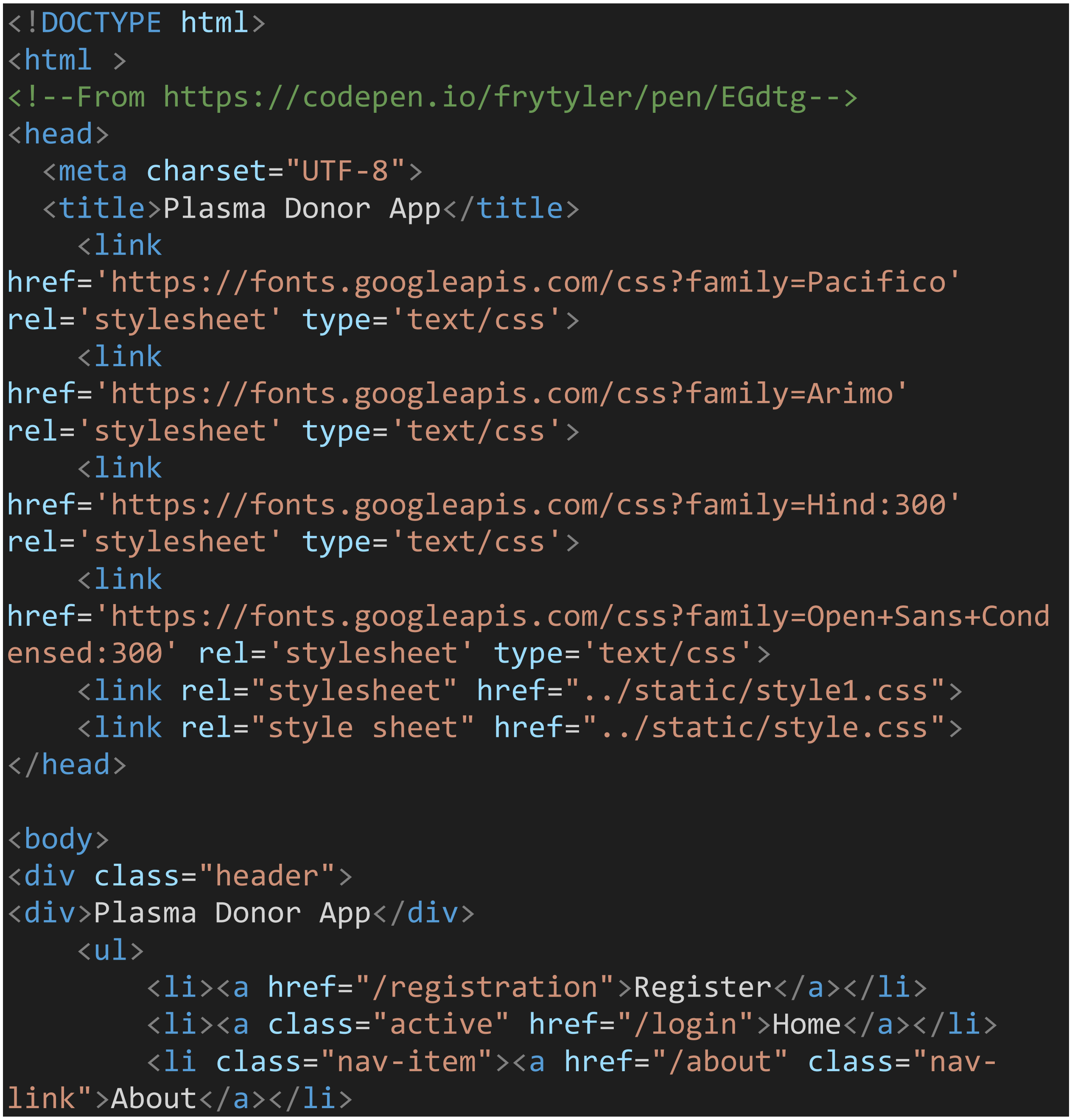
Upgrading the UI that is more user-friendly which will help many users to access the website and also ensures that many plasma donors can be added into the community.

Using elastic load balancer, it helps to handle multiple requests at the same time which will maintain the uptime of the website with negligible downtime.

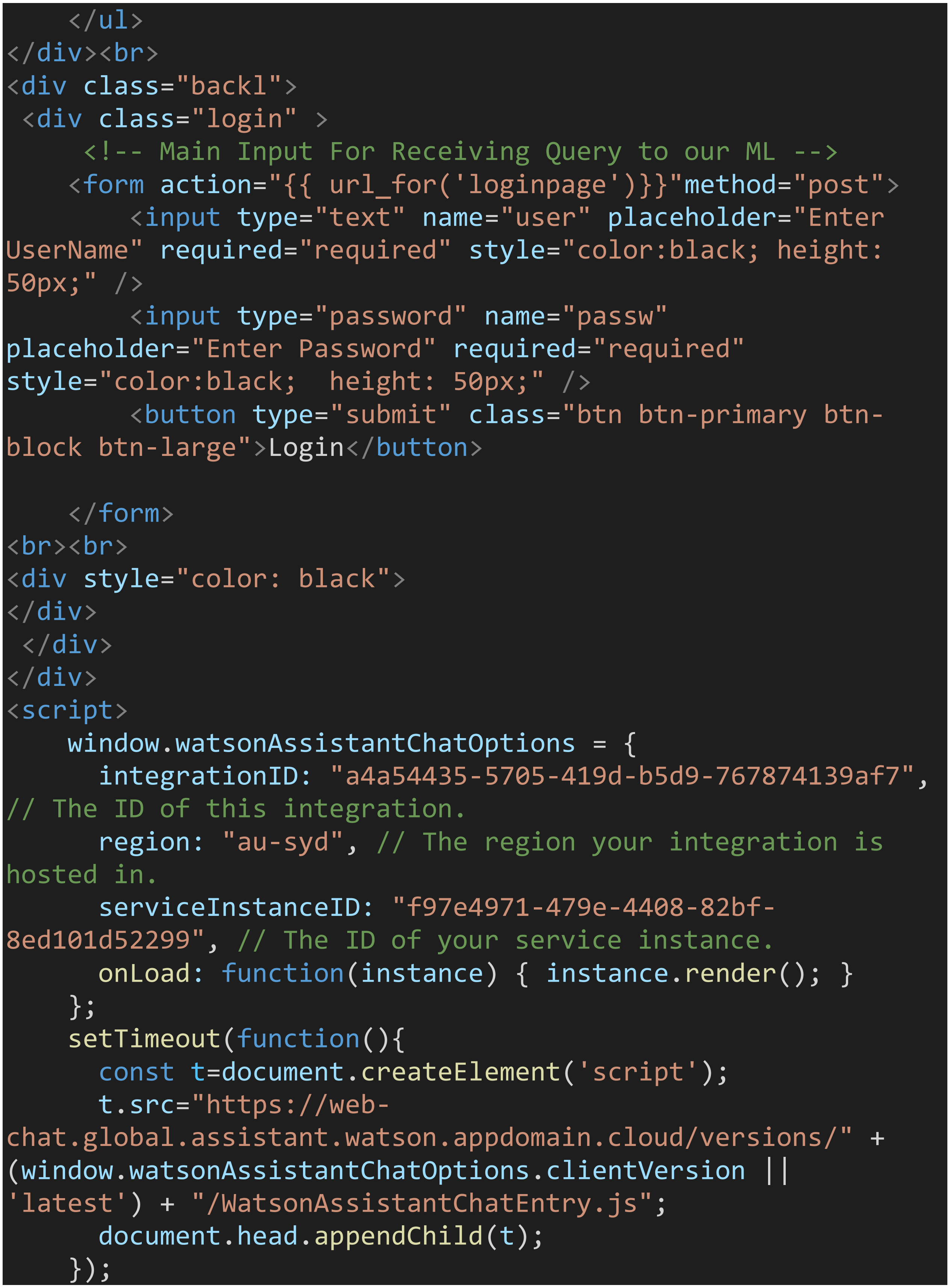
13 APPENDIXES

13.1 SOURCE CODE:

LOGIN.HTML



```
<!DOCTYPE html>
<html >
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>Plasma Donor App</title>
    <link
href='https://fonts.googleapis.com/css?family=Pacifico'
rel='stylesheet' type='text/css'>
    <link
href='https://fonts.googleapis.com/css?family=Arimo'
rel='stylesheet' type='text/css'>
    <link
href='https://fonts.googleapis.com/css?family=Hind:300'
rel='stylesheet' type='text/css'>
    <link
href='https://fonts.googleapis.com/css?family=Open+Sans+Cond
ensed:300' rel='stylesheet' type='text/css'>
    <link rel="stylesheet" href="../static/style1.css">
    <link rel="style sheet" href="../static/style.css">
</head>

<body>
<div class="header">
<div>Plasma Donor App</div>
    <ul>
        <li><a href="/registration">Register</a></li>
        <li><a class="active" href="/login">Home</a></li>
        <li class="nav-item"><a href="/about" class="nav-
link">About</a></li>
```

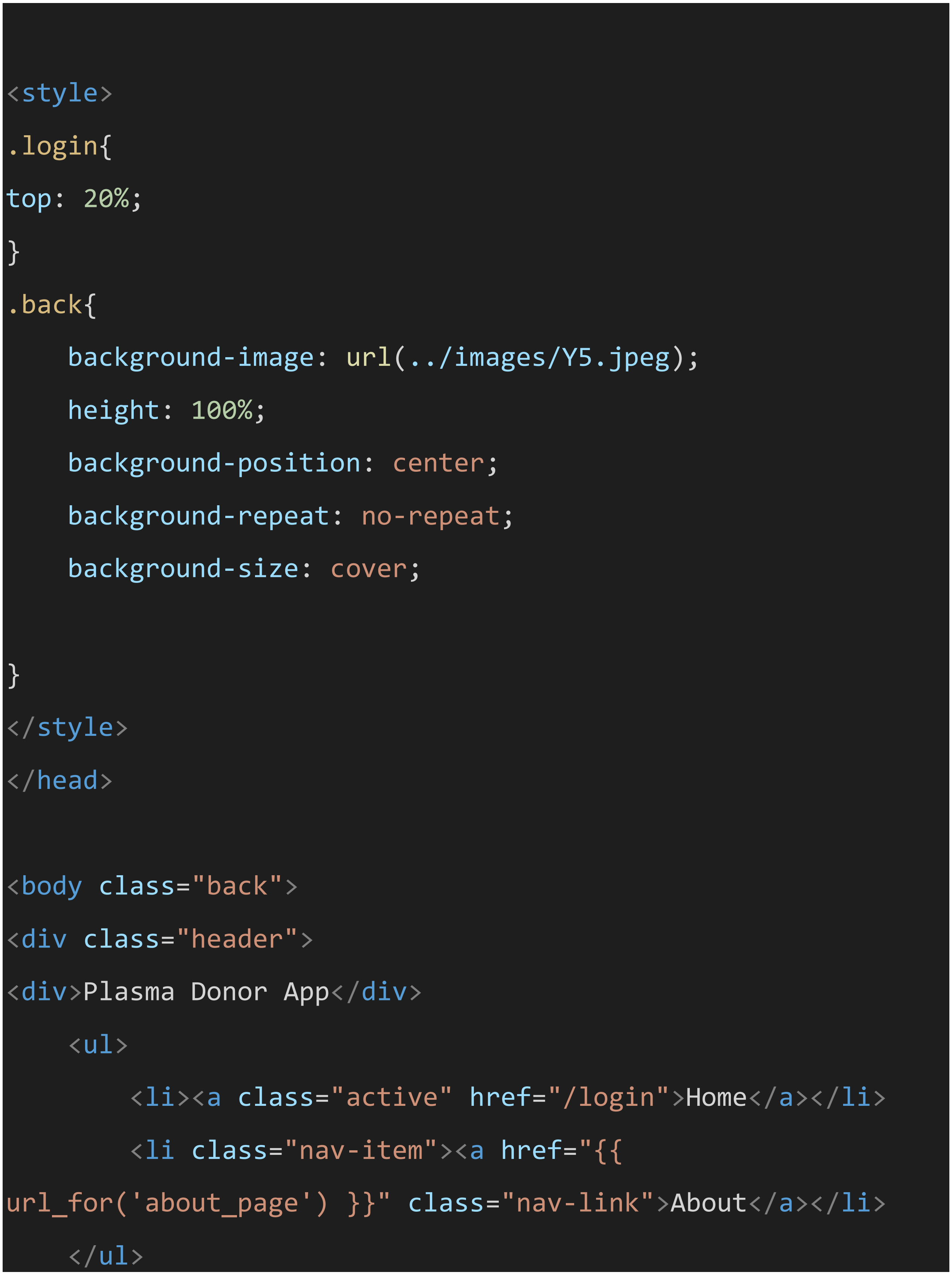


```
    </ul>
</div><br>
<div class="backl">
 <div class="login" >
     <!-- Main Input For Receiving Query to our ML -->
    <form action="{{ url_for('loginpage')}}"method="post">
        <input type="text" name="user" placeholder="Enter
UserName" required="required" style="color:black; height:
50px;" />
        <input type="password" name="passw"
placeholder="Enter Password" required="required"
style="color:black;  height: 50px;" />
        <button type="submit" class="btn btn-primary btn-
block btn-large">Login</button>

    </form>
<br><br>
<div style="color: black">
</div>
 </div>
</div>
<script>
    window.watsonAssistantChatOptions = {
      integrationID: "a4a54435-5705-419d-b5d9-767874139af7",
// The ID of this integration.
      region: "au-syd", // The region your integration is
hosted in.
      serviceInstanceID: "f97e4971-479e-4408-82bf-
8ed101d52299", // 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><!--end chatbot-->
</body>
</html>
```

REGISTER.HTML

```
<!DOCTYPE html>
<html >
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>Plasma Donor App</title>
    <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='style sheet' type='text/css'>
    <link href='https://fonts.googleapis.com/css?family=Arimo' rel='style sheet' type='text/css'>
    <link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
    <link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet' type='text/css'>
    <link rel="stylesheet" href="{{url_for('static', filename='style1.css') }}">
    <link rel="stylesheet" href="../static/style.css">
```



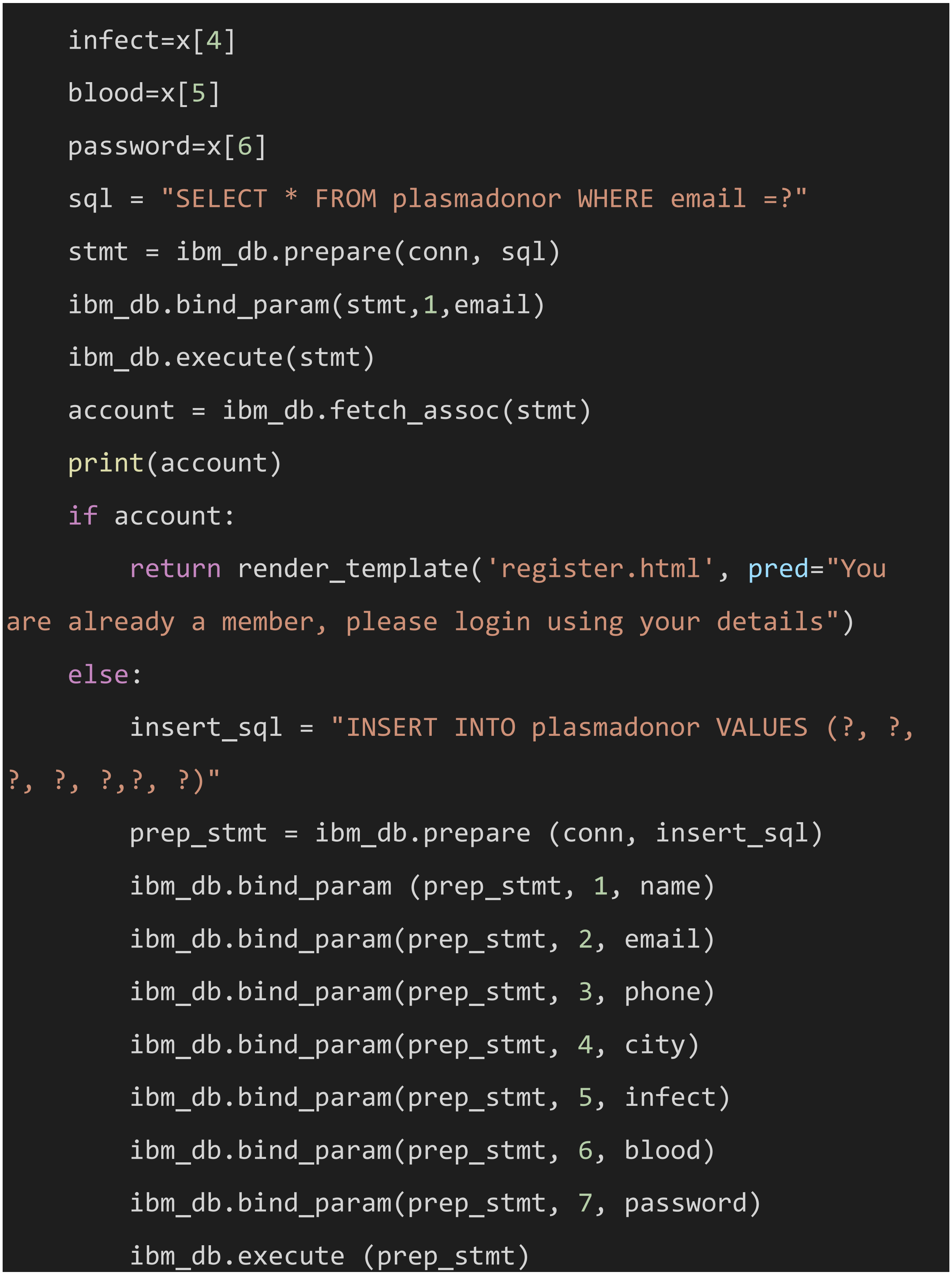
```
<style>
.login{
top: 20%;
}
.back{
    background-image: url(../images/Y5.jpeg);
    height: 100%;
    background-position: center;
    background-repeat: no-repeat;
    background-size: cover;

}
</style>
</head>

<body class="back">
<div class="header">
<div>Plasma Donor App</div>
    <ul>
        <li><a class="active" href="/login">Home</a></li>
        <li class="nav-item"><a href="{{
url_for('about_page') }}" class="nav-link">About</a></li>
    </ul>
```

```
</div>
<div class="backr">
 <div class="register">

     <!-- Main Input For Receiving Query to our ML -->
    <form action="{{ url_for('register')}}"method="post">
        <input type="text" name="name" placeholder="Enter
Your Name" required="required" style="color:black"/>
        <input type="email" name="email" placeholder="Enter
Email" required="required" style="color:black"/>
        <input type="text" name="phone" placeholder="Enter
10-digit mobile number" required="required"
style="color:black"/>
        <input type="city" name="city" placeholder="Enter
Your City Name" required="required" style="color:black"/>
        <select name="infect">

                      <option value="select" selected>Select
COVID infection status</option>
                      <option
value="infected">Infected</option>
                      <option
value="uninfected">Uninfected</option>
        </select>
        <select name="blood">
                        <option value="select" selected>Choose
your blood group</option>
                        <option value="O Positive">O
Positive</option>
                        <option value="A Positive">A
Positive</option>
                        <option value="B Positive">B
Positive</option>
                        <option value="AB Positive">AB
Positive</option>
                        <option value="O Negative">O
Negative</option>
                        <option value="A Negative">A
Negative</option>
                        <option value="B Negative">B
Negative</option>
                        <option value="AB Negative">AB
Negative</option>
        </select>
        <input type="password" name="passw"
placeholder="Enter Password" required="required"
style="color:black"/>
```

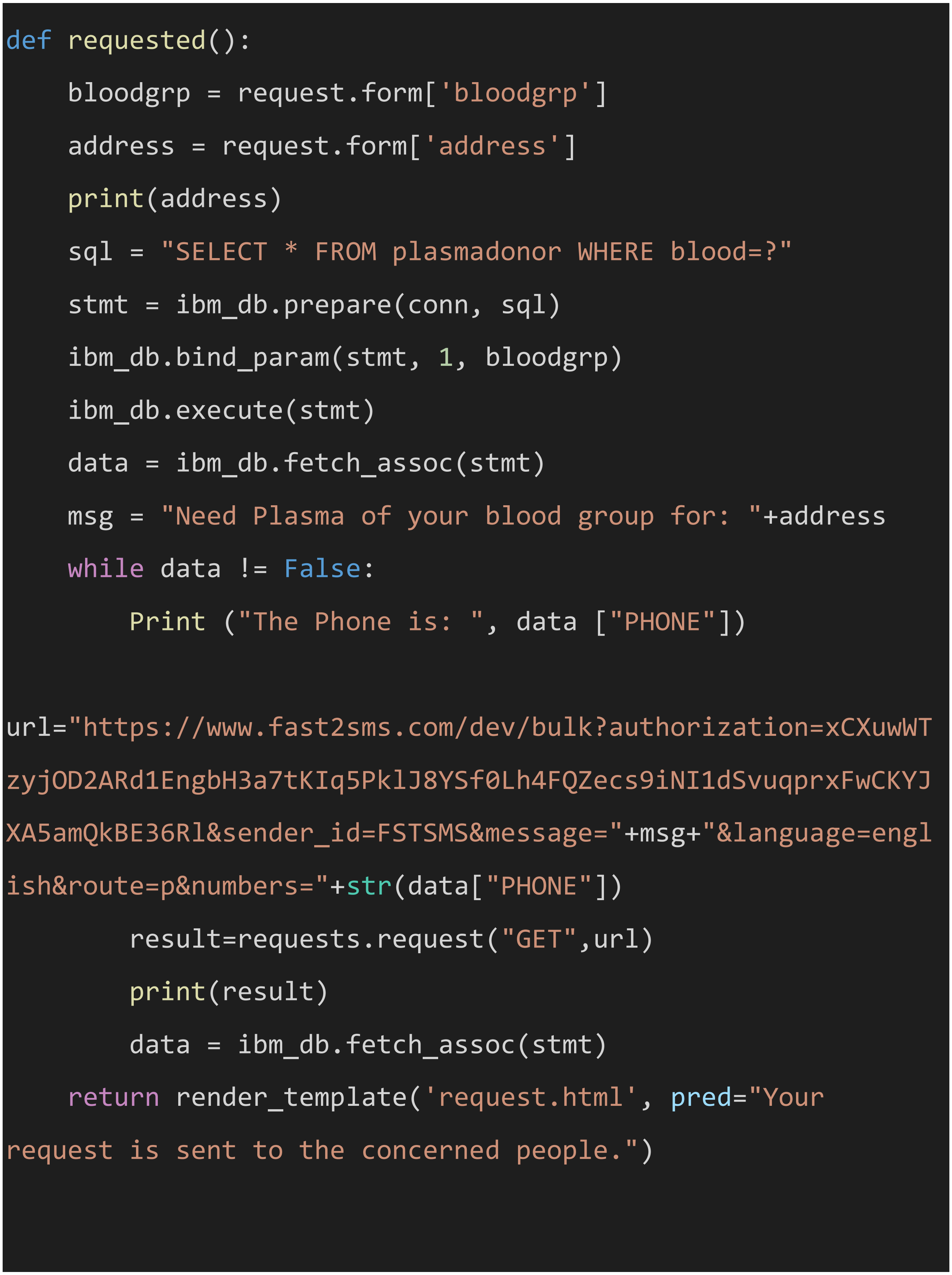
```
        <button type="submit" class="btn btn-primary btn-block btn-large">Register</button>

    </form>

 <br><br>
<div style="color:black">
</div>
 </div>
</div>
<script>
    window.watsonAssistantChatOptions = {
      integrationID: "a4a54435-5705-419d-b5d9-767874139af7", // The ID of this integration.
      region: "au-syd", // The region your integration is hosted in.
      serviceInstanceID: "f97e4971-479e-4408-82bf-8ed101d52299", // 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><!--end chatbot-->

</body>
</html>
```

APP.PY

```
from flask import Flask, render_template, request, redirect,
url_for, session
import ibm_db
import json
import requests
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=3883e7e4-
18f5-4afe-be8c-
fa31c41761d2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;
PORT=31498;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalR
ootCA.crt;UID=sch83401;PWD=j7QZUHGAtUGbPhns",'','')
app = Flask(__name__)
@app.route('/registration')
def home():
    return render_template('register.html')

@app.route('/register',methods=['POST'])
def register():
    x = [x for x in request.form.values()]
    print(x)
    name=x[0]
    email=x[1]
    phone=x[2]
    city=x[3]
```



```
    infect=x[4]
    blood=x[5]
    password=x[6]
    sql = "SELECT * FROM plasmadonor WHERE email =?"
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.bind_param(stmt,1,email)
    ibm_db.execute(stmt)
    account = ibm_db.fetch_assoc(stmt)
    print(account)
    if account:
        return render_template('register.html', pred="You are already a member, please login using your details")
    else:
        insert_sql = "INSERT INTO plasmadonor VALUES (?, ?, ?, ?, ?,?, ?)"
        prep_stmt = ibm_db.prepare (conn, insert_sql)
        ibm_db.bind_param (prep_stmt, 1, name)
        ibm_db.bind_param(prep_stmt, 2, email)
        ibm_db.bind_param(prep_stmt, 3, phone)
        ibm_db.bind_param(prep_stmt, 4, city)
        ibm_db.bind_param(prep_stmt, 5, infect)
        ibm_db.bind_param(prep_stmt, 6, blood)
        ibm_db.bind_param(prep_stmt, 7, password)
        ibm_db.execute (prep_stmt)
        Return render_template ('register.html',
pred="Registration Successful, please login using your
details")
@app.route ("/about")
Def about page():
    Return render_template ('about.html')

@app.route('/')
@app.route ('/login')
Def login ():
    Return render_template ('login.html')

@app.route ('/loginpage', methods= ['POST'])
Def loginpage():
    User = request.form ['user']
    Passw = request.form ['passw']
    Sql = "SELECT * FROM plasmadonor WHERE email =? AND
password=?"
    Stmt = ibm_db.prepare (conn, sql)
    ibm_db.bind_param (stmt, 1, user)
    ibm_db.bind_param (stmt, 2, passw)
    ibm_db.execute(stmt)
    Account = ibm_db.fetch_assoc (stmt)
    Print (account)
    Print (user, passw)
    If account:
            Return redirect (url_for ('stats'))
    Else:
        Return render_template ('login.html', pred="Login unsuccessful. Incorrect username / password!")


@app.route ('/status')
Def status ():
    '''sql = "SELECT blood FROM user group by blood"
    Stmt = ibm_db.prepare (conn, sql)
    ibm_db.execute (stmt)
    Count = ibm_db.fetch_assoc (stmt)
    Print (count)'''
    return render_template('status.html',b=5,b1=2,b2=3,b3=4,b4=2,b5=1,b6=2,b7=1,b8=1)
@app.route ('/requester')
def requester ():
    return render_template ('request.html')


@app.route('/requested',methods=['POST'])
```



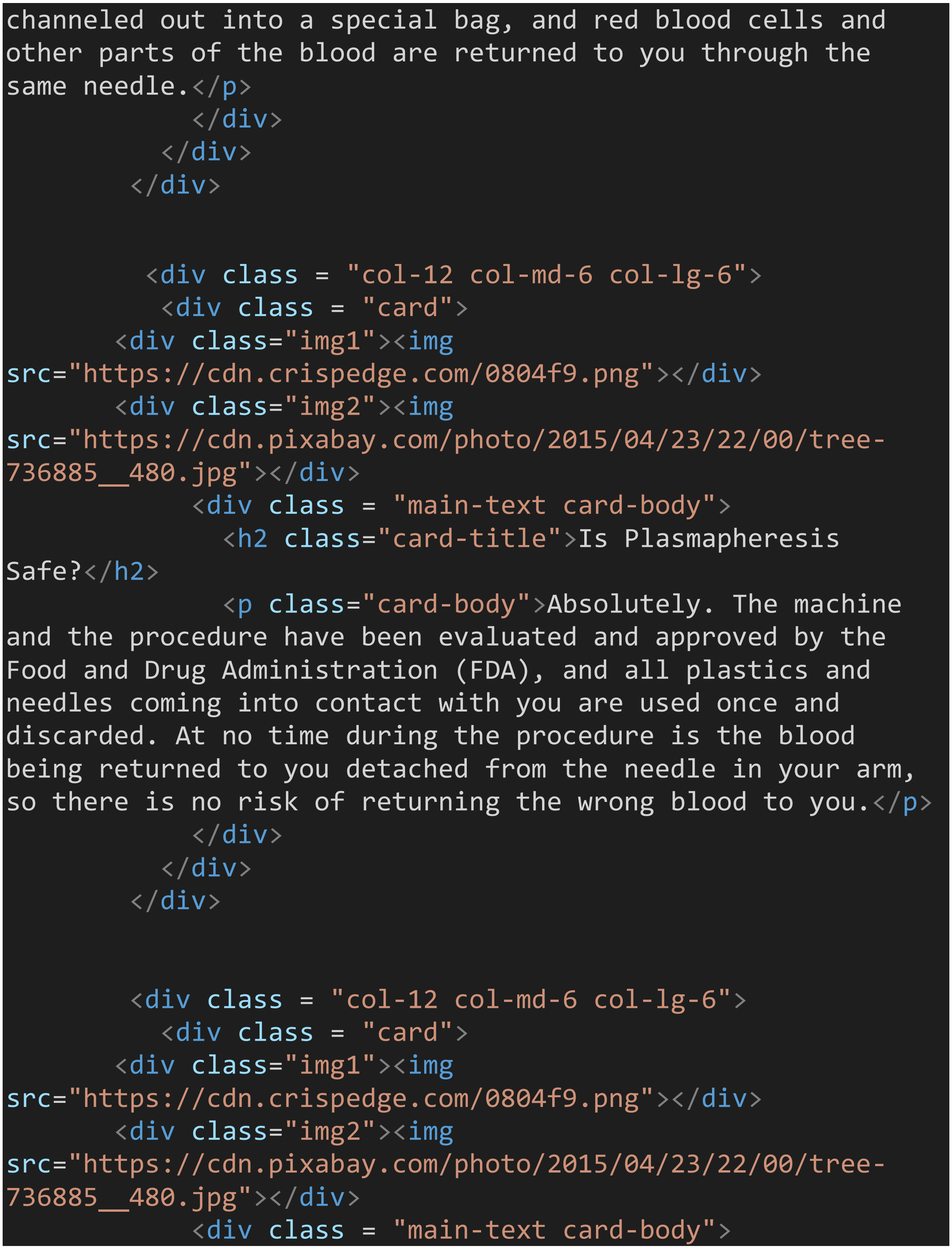
```python
def requested():
    bloodgrp = request.form['bloodgrp']
    address = request.form['address']
    print(address)
    sql = "SELECT * FROM plasmadonor WHERE blood=?"
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.bind_param(stmt, 1, bloodgrp)
    ibm_db.execute(stmt)
    data = ibm_db.fetch_assoc(stmt)
    msg = "Need Plasma of your blood group for: "+address
    while data != False:
        Print ("The Phone is: ", data ["PHONE"])

url="https://www.fast2sms.com/dev/bulk?authorization=xCXuwWTzyjOD2ARd1EngbH3a7tKIq5PklJ8YSf0Lh4FQZecs9iNI1dSvuqprxFwCKYJXA5amQkBE36Rl&sender_id=FSTSMS&message="+msg+"&language=english&route=p&numbers="+str(data["PHONE"])
        result=requests.request("GET",url)
        print(result)
        data = ibm_db.fetch_assoc(stmt)
    return render_template('request.html', pred="Your request is sent to the concerned people.")
```

```
if __name__ == "__main__":
    app.run(host='0.0.0.0', port=8080)
```

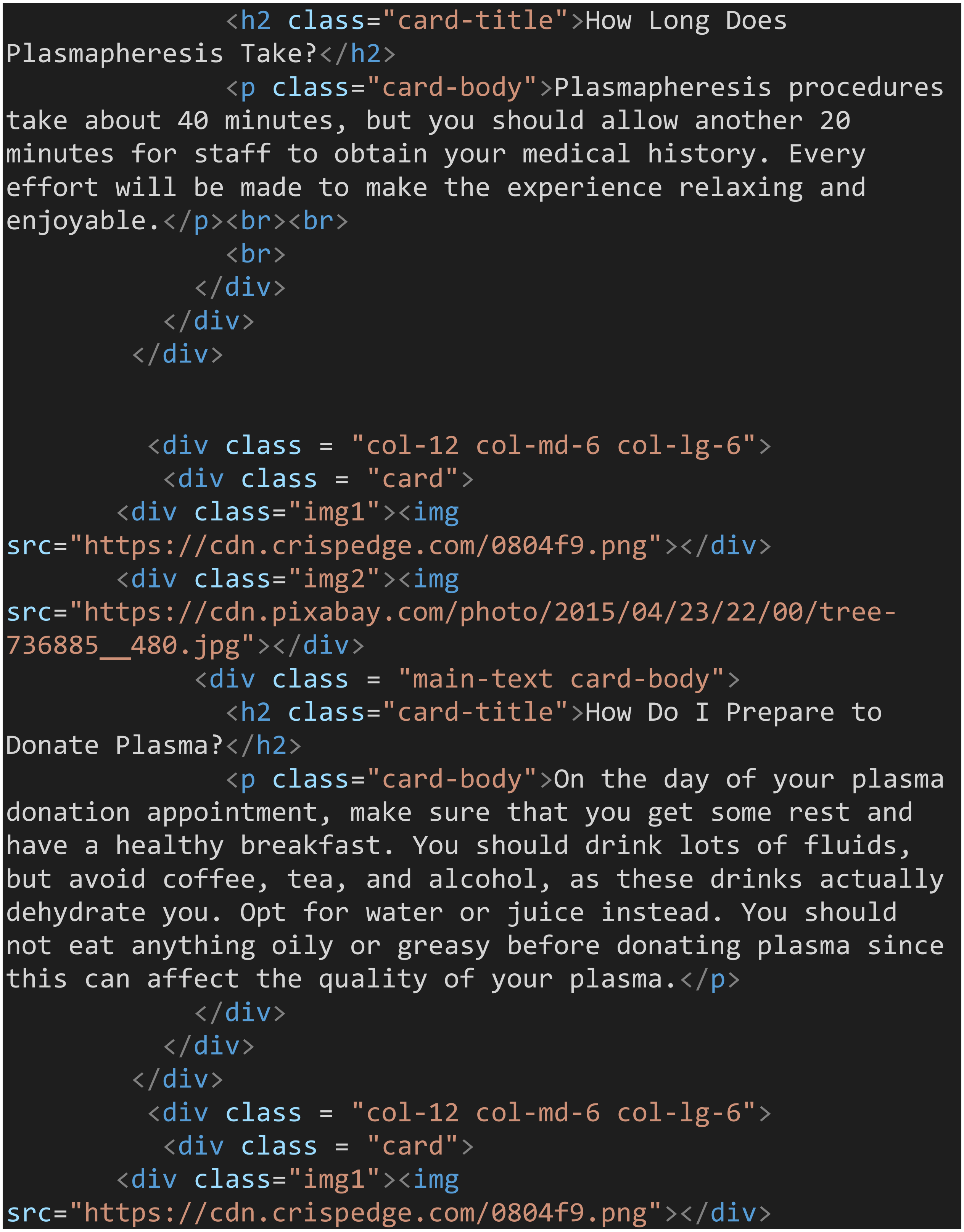
ABOUT PAGE:

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

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

{% block link %}

<link rel="style sheet" href="./static/about.css">
<style>
  body{
    background-color: red;
  }
</style>
{% endblock %}

<!---About Content-->
{% block content %}
<div class="header">
  <div>Plasma Donor App</div>
    <ul>
        <li><a class="active" href="/login">Home</a></li>
    </ul>
  </div><br><br><br>
<div class="container"><h1 class="heading">Know more about 
plasma</h1></div>
```

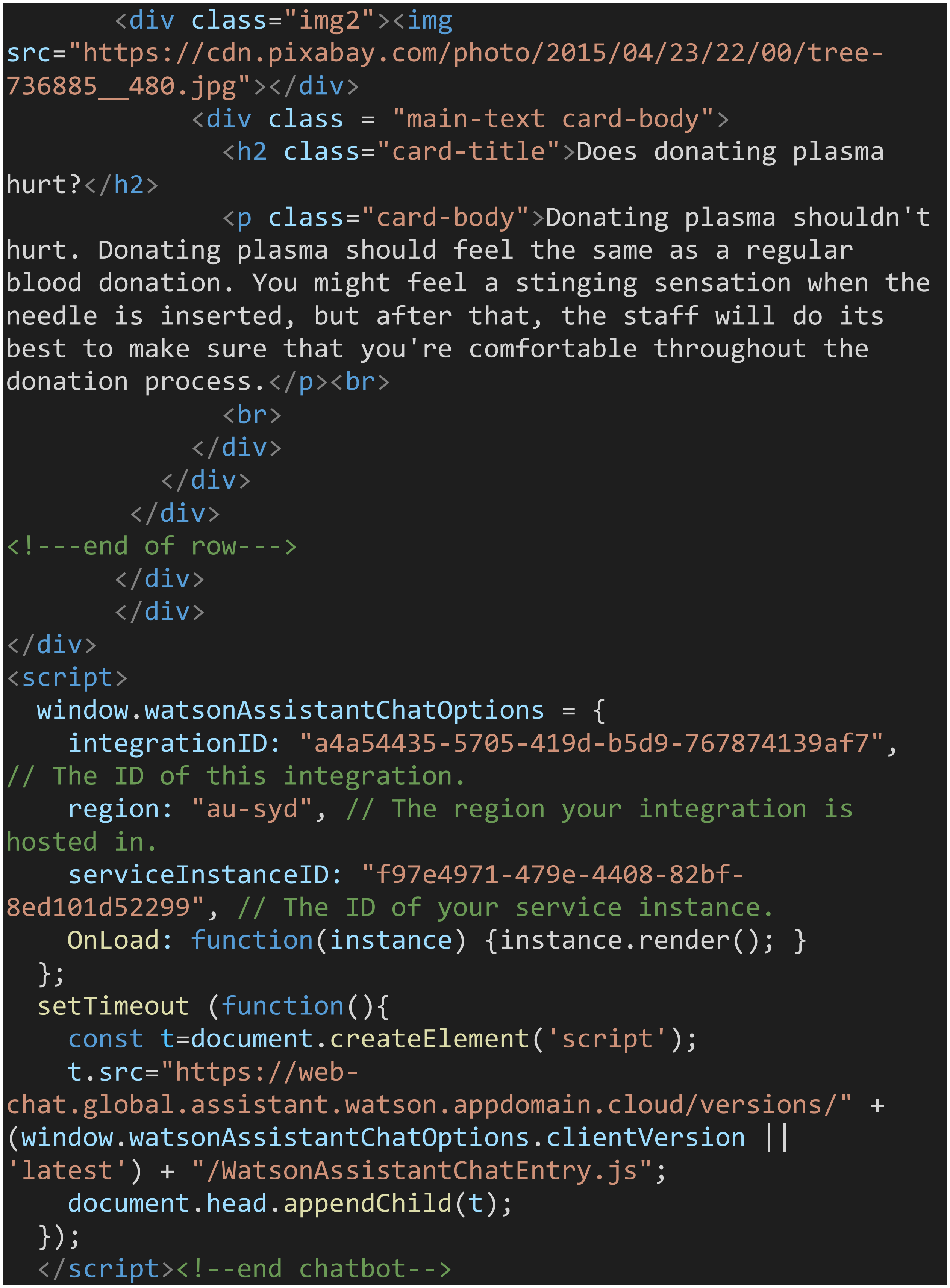
```
   <div class = "profile-area">
    <div class = "container">
      <div class="row">
        <div class = "col-12 col-md-6 col-lg-6">
          <div class = "card">
               <div class="img1"><imp
src="https://cdn.crispedge.com/0804f9.png"></div>
               <div class="img2"><img
src="https://cdn.pixabay.com/photo/2015/04/23/22/00/tree-
736885__480.jpg"></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://cdn.crispedge.com/0804f9.png"></div>
        <div class="img2"><img
src="https://cdn.pixabay.com/photo/2015/04/23/22/00/tree-
736885__480.jpg"></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://cdn.crispedge.com/0804f9.png"></div>
       <div class="img2"><img
src="https://cdn.pixabay.com/photo/2015/04/23/22/00/tree-
736885__480.jpg"></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://cdn.crispedge.com/0804f9.png"></div>
       <div class="img2"><img
src="https://cdn.pixabay.com/photo/2015/04/23/22/00/tree-
736885__480.jpg"></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://cdn.crispedge.com/0804f9.png"></div>
       <div class="img2"><img
src="https://cdn.pixabay.com/photo/2015/04/23/22/00/tree-
736885__480.jpg"></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://cdn.crispedge.com/0804f9.png"></div>
```

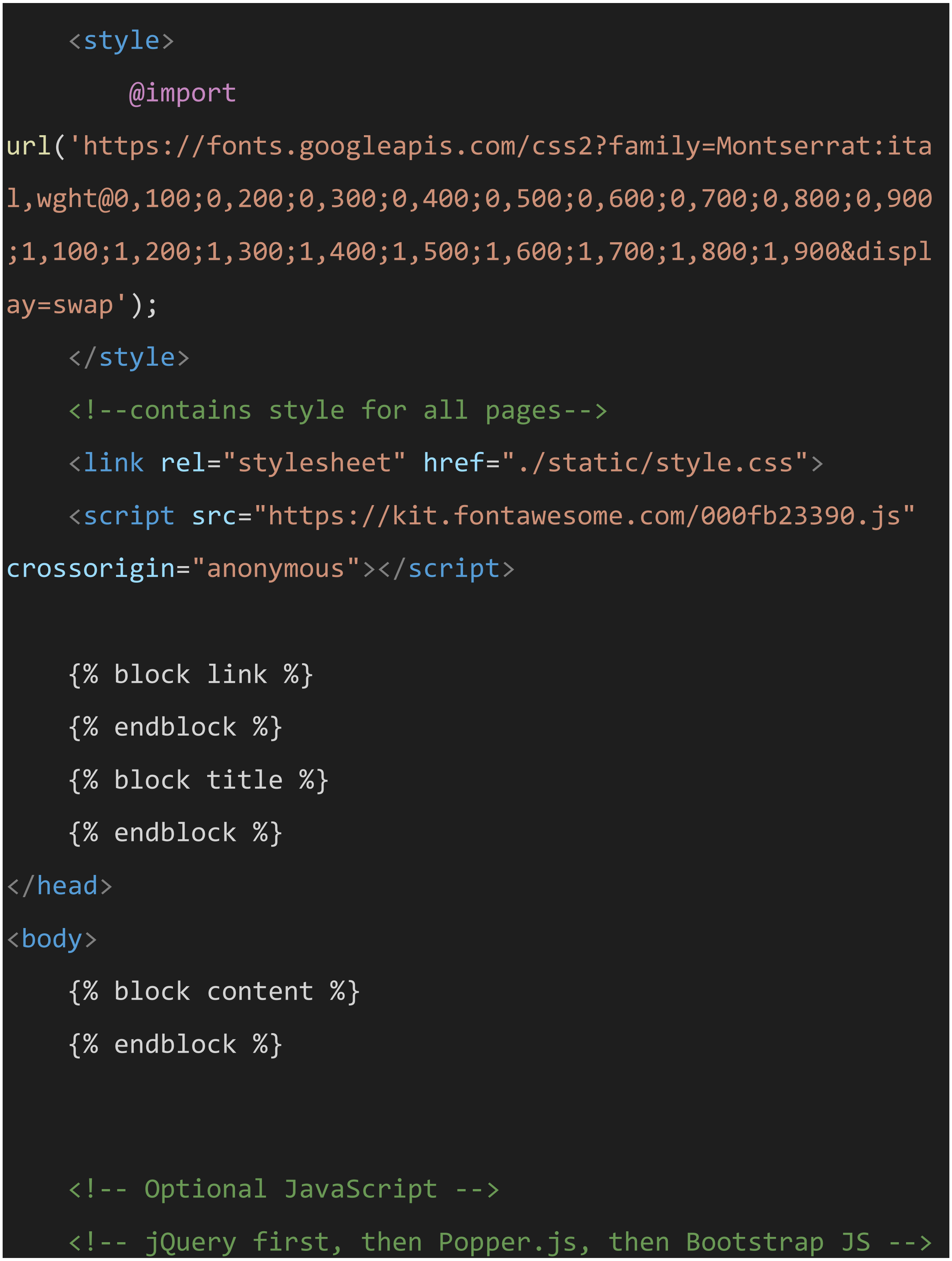


```
       <div class="img2"><img
src="https://cdn.pixabay.com/photo/2015/04/23/22/00/tree-
736885__480.jpg"></div>
            <div class = "main-text card-body">
              <h2 class="card-title">Does donating plasma
hurt?</h2>
              <p class="card-body">Donating plasma shouldn't
hurt. Donating 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>
<script>
  window.watsonAssistantChatOptions = {
    integrationID: "a4a54435-5705-419d-b5d9-767874139af7",
// The ID of this integration.
    region: "au-syd", // The region your integration is
hosted in.
    serviceInstanceID: "f97e4971-479e-4408-82bf-
8ed101d52299", // 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><!--end chatbot-->
```

```
{% 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="./static/style.css">
    <script src="https://kit.fontawesome.com/000fb23390.js"
crossorigin="anonymous"></script>

    {% block link %}
    {% endblock %}
    {% block title %}
    {% endblock %}
</head>
<body>
    {% 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>

</body>
</html>
```

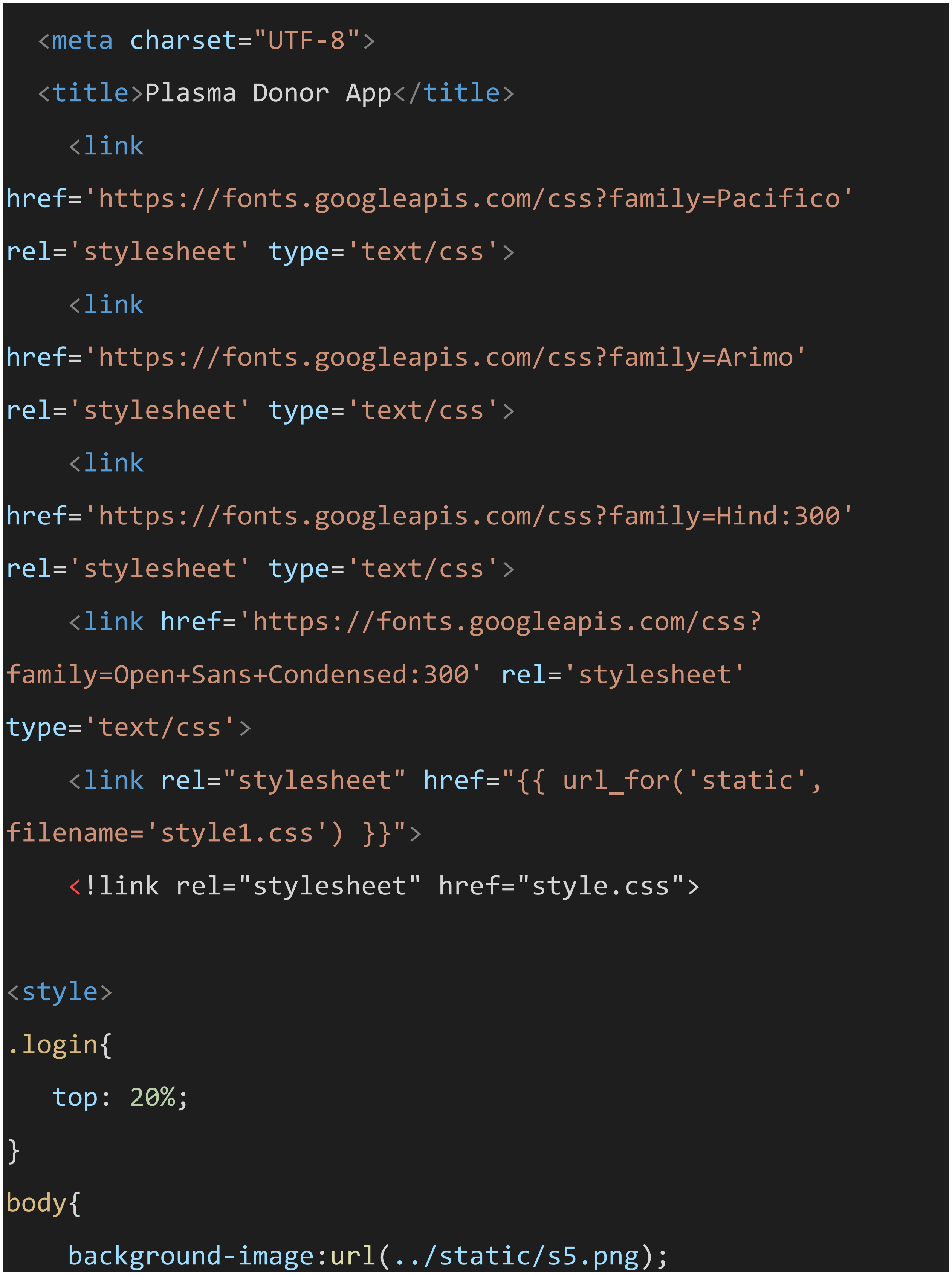
REQUEST AND RESPONSE.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>plasma-donar-application</title>
    <link rel="stylesheet" href="{{ url_for('static',
filename='style1.css') }}">
    <link rel="stylesheet" href="../static/style.css">

<style>
.login{
top: 20%;
}
.back{
    background-image: url(../static/s7.png);
    height: 100%;
    background-position: center;
    background-repeat: no-repeat;
    background-size:cover;

}
.r{
    background-color: #4a77d4;
    background-image: -moz-linear-gradient(top, #6eb6de,
#4a77d4);
    background-image: -ms-linear-gradient(top, #6eb6de, #4a77d4);
    background-image: -webkit-gradient(linear, 0 0, 0 100%,
from(#6eb6de), to(#4a77d4));
    background-image: -webkit-linear-gradient(top, #6eb6de,
#4a77d4);
    background-image: -o-linear-gradient(top, #6eb6de, #4a77d4);
    background-image: linear-gradient(top, #6eb6de, #4a77d4);
    background-repeat: repeat-x;
    filter:
progid:dximagetransform.microsoft.gradient(startColorstr=#6eb6de,
endColorstr=#4a77d4, GradientType=0);
    border: 1px solid #3762bc;
    text-shadow: 1px 1px 1px rgba(0,0,0,0.4);
    box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px
rgba(0, 0, 0, 0.5);
    margin-top: 350px;
    margin-left: 850px;
   padding: 15px 50px;
}
.s{
    background-color: #4a77d4;
    background-image: -moz-linear-gradient(top, #6eb6de,
#4a77d4);
    background-image: -ms-linear-gradient(top, #6eb6de, #4a77d4);
    background-image: -webkit-gradient(linear, 0 0, 0 100%,
from(#6eb6de), to(#4a77d4));
    background-image: -webkit-linear-gradient(top, #6eb6de,
#4a77d4);
    background-image: -o-linear-gradient(top, #6eb6de, #4a77d4);
    background-image: linear-gradient(top, #6eb6de, #4a77d4);
    background-repeat: repeat-x;
    filter:
progid:dximagetransform.microsoft.gradient(startColorstr=#6eb6de,
endColorstr=#4a77d4, GradientType=0);
    border: 1px solid #3762bc;
    text-shadow: 1px 1px 1px rgba(0,0,0,0.4);
    box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px
rgba(0, 0, 0, 0.5);
    margin-top: 350px;
    margin-left: 20px;
   padding: 15px 50px;
}
```

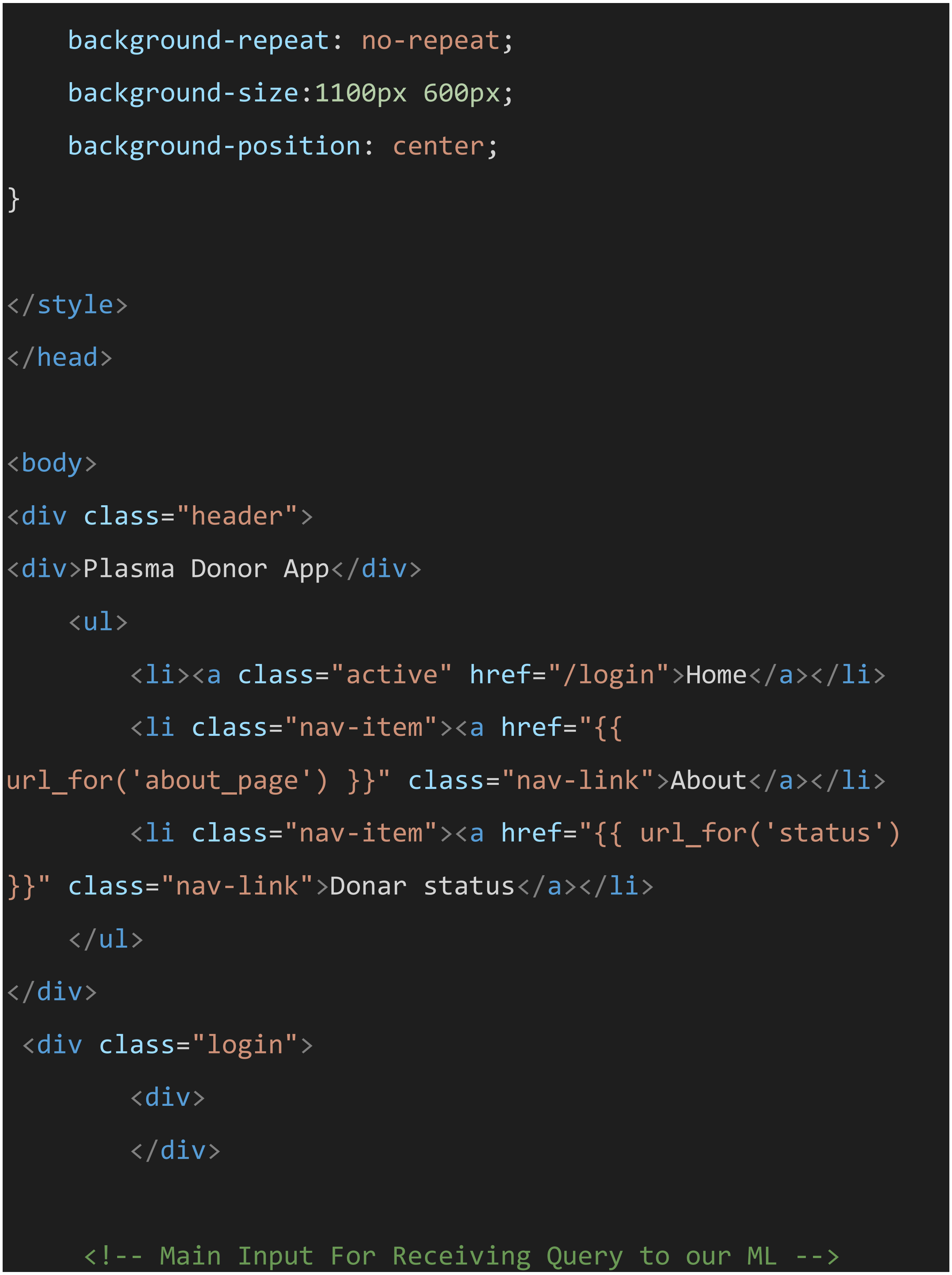
```
</style>
</head>
<body>
    <div class="header">
        <div>Plasma Donor App</div>
            <ul>
                <li><a href="/login">Home</a></li>
                <li class="nav-item"><a href="/about" class="nav-
link">About</a></li>
            </ul>
        </div>
    <div class="back">
        <button class="r"><a
href="/requester">request</a></button>
        <button class="s"><a href="/status">donar
status</a></button>

    </div>
</body>
</html>
```

REQUEST.HTML

```
<!DOCTYPE html>

<html >

<!--From https://codepen.io/frytyler/pen/EGdtg-->

<head>
```



```
  <meta charset="UTF-8">
  <title>Plasma Donor App</title>
    <link
href='https://fonts.googleapis.com/css?family=Pacifico'
rel='stylesheet' type='text/css'>
    <link
href='https://fonts.googleapis.com/css?family=Arimo'
rel='stylesheet' type='text/css'>
    <link
href='https://fonts.googleapis.com/css?family=Hind:300'
rel='stylesheet' type='text/css'>
    <link href='https://fonts.googleapis.com/css?
family=Open+Sans+Condensed:300' rel='stylesheet'
type='text/css'>
    <link rel="stylesheet" href="{{ url_for('static',
filename='style1.css') }}">
    <!link rel="stylesheet" href="style.css">

<style>
.login{
   top: 20%;
}
body{
    background-image:url(../static/s5.png);
```

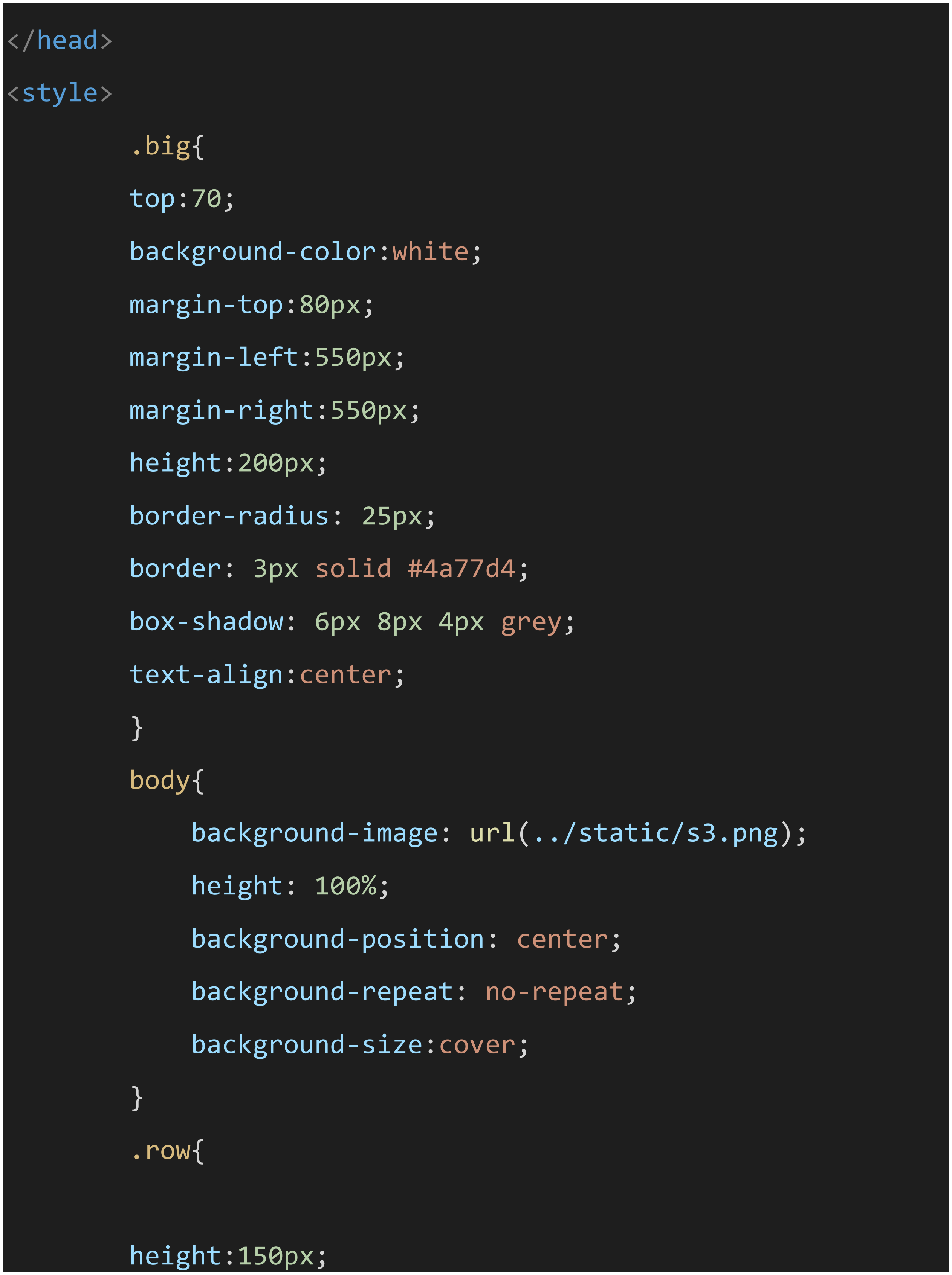


```
    background-repeat: no-repeat;
    background-size:1100px 600px;
    background-position: center;
}

</style>
</head>

<body>
<div class="header">
<div>Plasma Donor App</div>
    <ul>
        <li><a class="active" href="/login">Home</a></li>
        <li class="nav-item"><a href="{{
url_for('about_page') }}" class="nav-link">About</a></li>
        <li class="nav-item"><a href="{{ url_for('status')
}}" class="nav-link">Donar status</a></li>
    </ul>
</div>
 <div class="login">
        <div>
        </div>

     <!-- Main Input For Receiving Query to our ML -->
    <form action="{{ url_for('requested')}}"method="post">
        <select name="bloodgrp">
                      <option value="select" selected>Choose
your blood group</option>
                      <option value="O Positive">O
Positive</option>
                      <option value="A Positive">A
Positive</option>
                      <option value="B Positive">B
Positive</option>
                      <option value="AB Positive">AB
Positive</option>
                      <option value="O Negative">O
Negative</option>
                      <option value="A Negative">A
Negative</option>
                      <option value="B Negative">B
Negative</option>
                      <option value="AB Negative">AB
Negative</option>
        </select>
        <textarea rows="4" placeholder="Enter the address"
required="required" style="color:black"
name="address"></textarea>
```

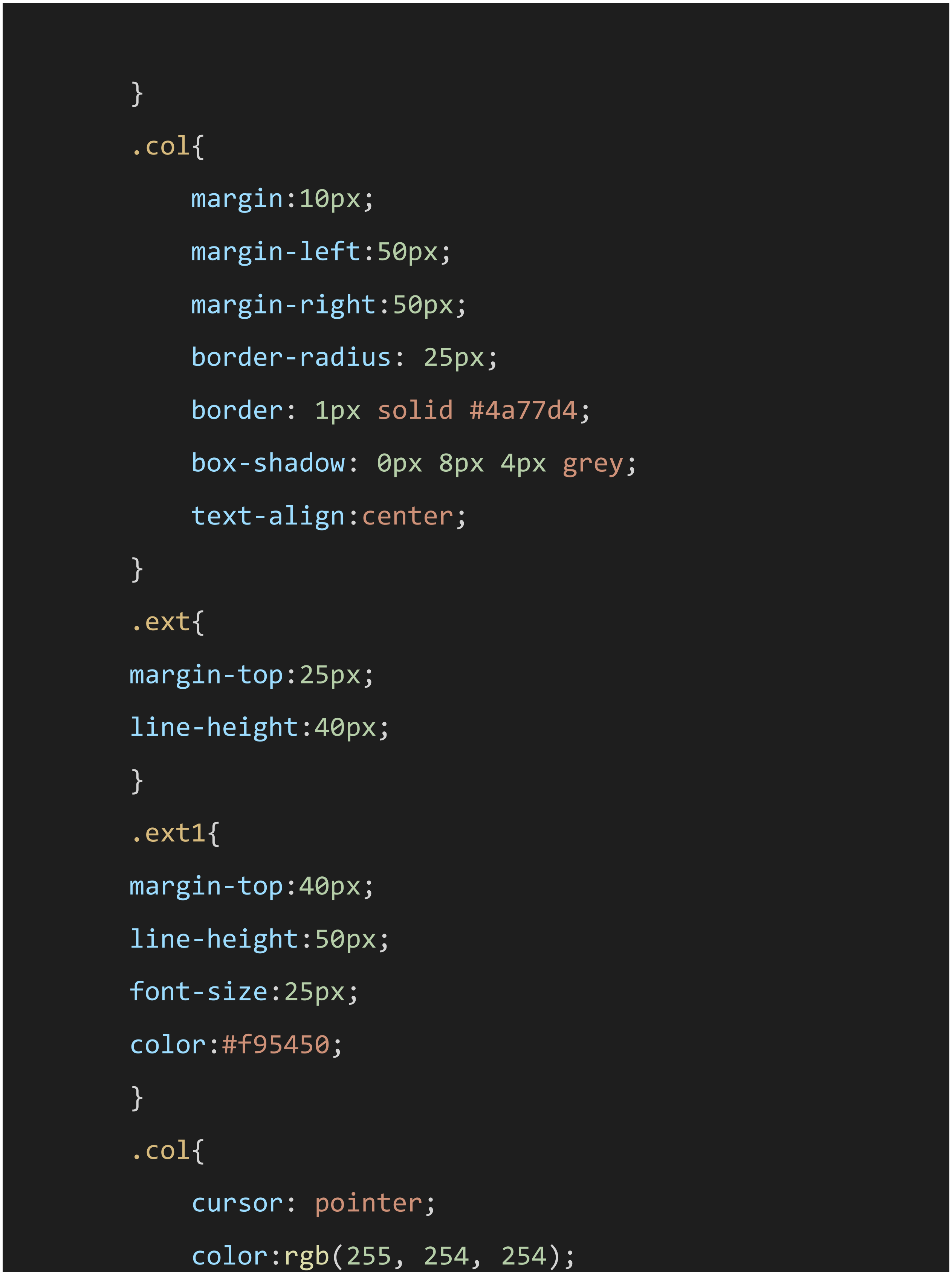
```
        <!input type="textarea" name="address" rows="4"
placeholder="Enter the address" required="required"
style="color:black" />
        <button type="submit" class="btn btn-primary btn-
block btn-large">Submit the request</button>

    </form>


 <br><br>
<div style="color:black">
 {{ pred }}</div>

 </div>
 <script>
    window.watsonAssistantChatOptions = {
      integrationID: "a4a54435-5705-419d-b5d9-767874139af7",
// The ID of this integration.
      region: "au-syd", // The region your integration is
hosted in.
      serviceInstanceID: "f97e4971-479e-4408-82bf-
8ed101d52299", // 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><!--end chatbot-->

</body>
</html>
```

PLASMA DONOR.HTML

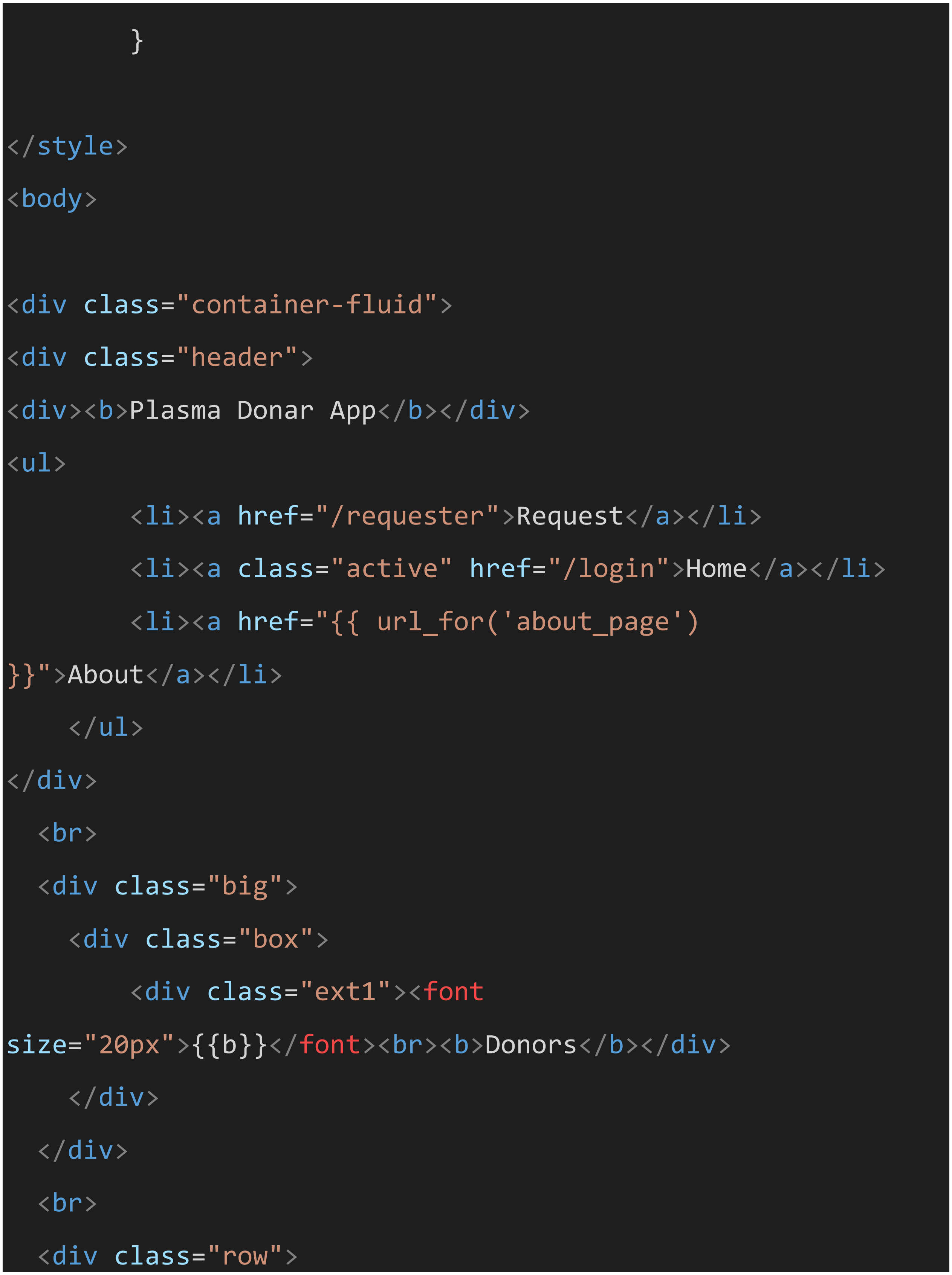
```html
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Plasma Donar App</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
  <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
  <link rel="stylesheet" href="style.css">
```



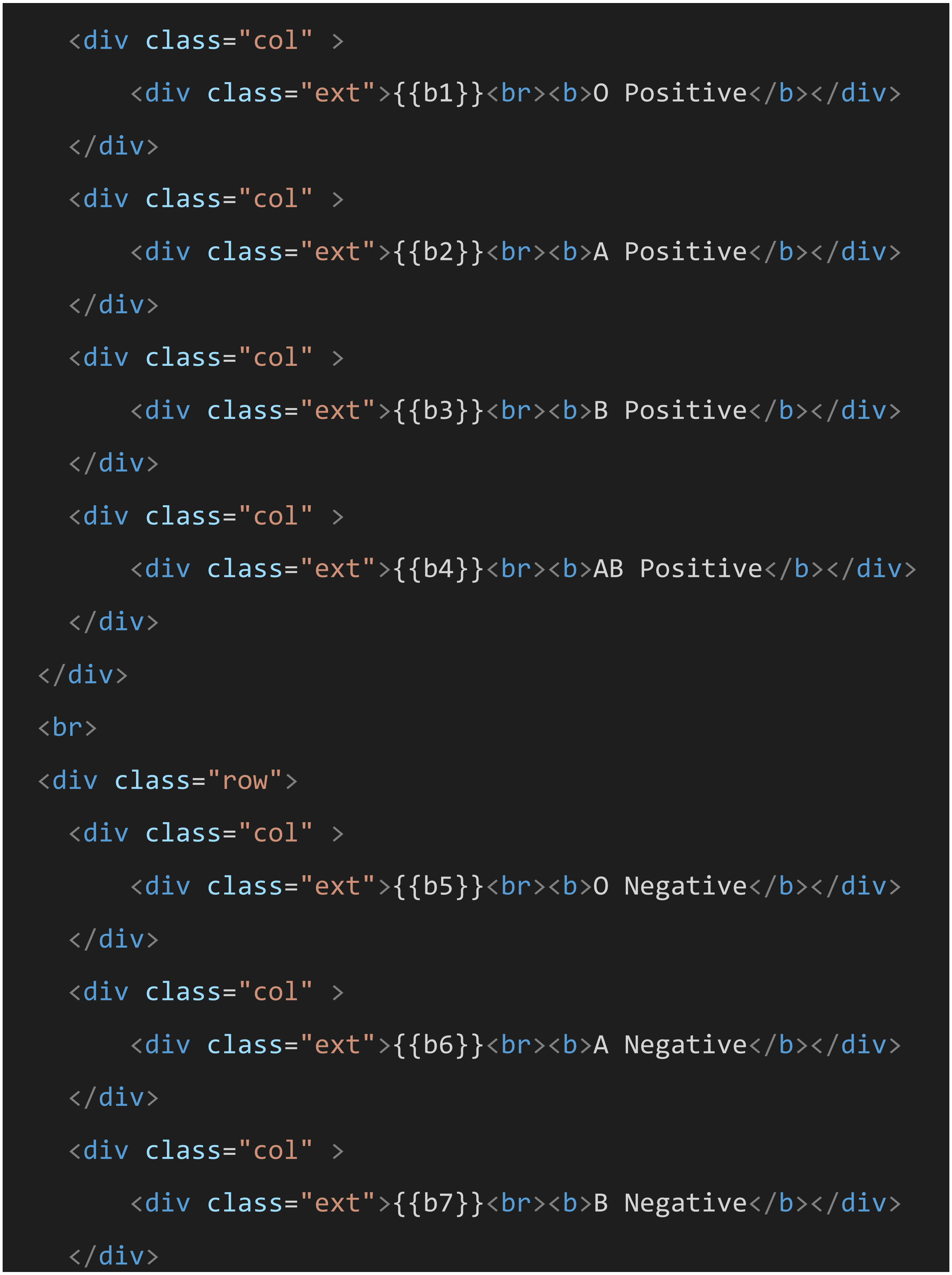
```
</head>
<style>
        .big{
        top:70;
        background-color:white;
        margin-top:80px;
        margin-left:550px;
        margin-right:550px;
        height:200px;
        border-radius: 25px;
        border: 3px solid #4a77d4;
        box-shadow: 6px 8px 4px grey;
        text-align:center;
        }
        body{
            background-image: url(../static/s3.png);
            height: 100%;
            background-position: center;
            background-repeat: no-repeat;
            background-size:cover;
        }
        .row{

        height:150px;
```



```
}
.col{
    margin:10px;
    margin-left:50px;
    margin-right:50px;
    border-radius: 25px;
    border: 1px solid #4a77d4;
    box-shadow: 0px 8px 4px grey;
    text-align:center;
}
.ext{
margin-top:25px;
line-height:40px;
}
.ext1{
margin-top:40px;
line-height:50px;
font-size:25px;
color:#f95450;
}
.col{
    cursor: pointer;
    color:rgb(255, 254, 254);
```



```
        }

</style>
<body>

<div class="container-fluid">
<div class="header">
<div><b>Plasma Donar App</b></div>
<ul>
        <li><a href="/requester">Request</a></li>
        <li><a class="active" href="/login">Home</a></li>
        <li><a href="{{ url_for('about_page')
}}">About</a></li>
    </ul>
</div>
  <br>
  <div class="big">
    <div class="box">
        <div class="ext1"><font
size="20px">{{b}}</font><br><b>Donors</b></div>
    </div>
  </div>
  <br>
  <div class="row">
```



```
  <div class="col" >
      <div class="ext">{{b1}}<br><b>O Positive</b></div>
  </div>
  <div class="col" >
      <div class="ext">{{b2}}<br><b>A Positive</b></div>
  </div>
  <div class="col" >
      <div class="ext">{{b3}}<br><b>B Positive</b></div>
  </div>
  <div class="col" >
      <div class="ext">{{b4}}<br><b>AB Positive</b></div>
  </div>
</div>
<br>
<div class="row">
  <div class="col" >
      <div class="ext">{{b5}}<br><b>O Negative</b></div>
  </div>
  <div class="col" >
      <div class="ext">{{b6}}<br><b>A Negative</b></div>
  </div>
  <div class="col" >
      <div class="ext">{{b7}}<br><b>B Negative</b></div>
  </div>
```

```
    <div class="col" >
        <div class="ext">{{b8}}<br><b>AB Negative</b></div>
    </div>
  </div>
</div>
<script>
    window.watsonAssistantChatOptions = {
      integrationID: "a4a54435-5705-419d-b5d9-767874139af7", // The ID of this integration.
      region: "au-syd", // The region your integration is hosted in.
      serviceInstanceID: "f97e4971-479e-4408-82bf-8ed101d52299", // 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><!--end chatbot-->
```

```
</body>

</html>
```

13.2 GITHUB

https://github.com/IBM-EPBL/IBM-Project-54945-1663221870

PROJECT DEMO LINK

https://vimeo.com/771570408