

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
1	INTRODUCTION	
1.1	Project Overview	
1.2	Purpose	
2	LITERATURE SURVEY	
2.1	Existing Problem	
2.2	References	
2.3	Problem Statement Definition	
3	IDEATION AND PROPOSED SOLUTION	
3.1	Empathy Map Canvas	
3.2	Ideation And Brainstorming	
3.3	Proposed solution	
4	REQUIREMENT ANALYSIS	
4.1	Functional Requirement	
4.2	Non Functional Requirement	
5	PROJECT DESIGN	
5.1	Dataflow Diagrams	
5.2	Solution And Technical Architecture	
5.3	User Stories	
6	PROJECT PLANNING AND SCHEDULING	
6.1	Sprint Delivery And Schedule	

7	CODING AND SOLUTIONING
7.1	Feature 1
7.2	Feature 2
8	TESTING
8.1	Test cases
8.2	User Acceptance Testing
9	RESULTS
9.1	Performance Metrics
10	ADVANTAGES AND DISADVANTAGES
11	CONCLUSION
12	FUTURE SCOPE
13	APPENDIX
	Source Code
	Github And Project Demo Link

INTRODUCTION

1.1 Project overview:

Plasma is used by pharmaceutical companies to make plasma-derived medicinal products (PDMPs). PDMPs are used to treat conditions such as immune deficiencies and bleeding disorders. Several PDMPs are included in the WHO Model Lists of Essential Medicines. According to the WHO, self-sufficiency driven by voluntary (non-remunerated) plasma donations is an important national goal to ensure an adequate supply is secured to meet the needs of the population. Australia, New Zealand, the UK, the Netherlands, and France only allow public or not-for-profit sectors to collect plasma for fractionation. Each of the 5 countries have toll or contract agreements with 1 private commercial plasma fractionator to manufacture PDMPs from the plasma collected within their respective countries. None of these countries pay plasma donors. Donors are only permitted to donate every 2 weeks (24 to 26 times per year) in these 5 countries. Austria, the Czech Republic, Germany, and the US allow both public and non-for-profit sectors, as well as commercial private plasma collection centres, to operate in the country. Private, not-for-profit, or public plasma collection centres in these 4 countries offer monetary compensation and other in-kind incentives to plasma donors. While the Czech Republic limits plasma donation to every 2 weeks, a much higher frequency of donation is allowed in other countries; up to 50 times per year in Austria, 60 times per year in Germany, and more than 100 times per year in the US. Austria, the Czech Republic, Germany, and the US (which allow commercial private plasma collectors to operate and pay donors) are 100% self-sufficient in immune globulins. These 4 countries collect the most plasma, which is from paid donors. In 2017, Austria, the Czech Republic, Germany, and the US collected 75 litres per 1,000 people, 45 litres per 1,000 people, 36 litres per 1,000 people, and 113 litres per 1,000 people of plasma for fractionation, respectively. Countries that do not pay donors including Australia, New Zealand, the UK, the Netherlands, and France are dependent to some extent on US and European Union donors who are paid for the supply of plasma or imported PDMPs.

1.2 Purpose:

To satisfy the Applicant by Donating the plasma who are under the critical situation .This may save many people lives who utilize this application in an efficient manner.The limited literature search conducted for the Environmental Scan did not identify publications on events of disease transmission through PDMPs manufactured from either paid or non-renumerated donors' plasma, the impact of plasma collection centres (including those that do or do not pay donors) on the collection of whole blood or other blood components, or the long-term costs associated with plasma self-sufficiency on the health care system. Registered hospitals can check the availability of required plasma and can send request to the nearby banks and also order for donors whose blood type match up with the required blood type. The location of nearest plasma donation centers, the banks and hospitals which request for the availability of plasma are traced by GPS tracking.

2.LITERATURE SURVEY

2.1 Existing problem

ABOUT THE SYSTEM

The system which was manual that is based on paper card to collect Plasma donor data, keep record of Plasma donors and disseminate results to Plasma donors, had weakness that needed IT based solutions. The system was characterized by delays and sometime failure to access historical records; errors were witnessed in entry and manual analysis of results, secrecy and confidentiality of records lacked because unauthorized persons could easily access the records. In regard to the existing plasma management system, we have taken into account a system called Raster's web-based Plasma Management System. Plasma banks play an important role in the process of collecting blood and managing blood stocks, approving Plasma requests, updating donations and updating available Plasma. Raster's web-based BBMS will address the issues and problems encountered in collecting information about donors, Plasma camps, inventories of blood bags, and Plasma transfusion services, etc., including donor screening, inventory management, Plasma ordering, Plasma usage review and compatibility testing. Plasma Management system will greatly increase the safety and quality of the Plasma supply as well as provide logistics data for the optimal supply chain management.

DRAWBACKS OF THE EXISTING SYSTEM

- § Less Security: A proper validation interface is not available in this system which in turn brings a dilemma among the donors in regard to their details and intervention of third parties.
- § Not a reliable service system: It is user friendly where the GUI is not much interactive the users, this a major drawback of the system.

- § Has some troubleshooting problems: If the user inputs some misinformation, there is no valid gateway to cross check the provided details.
- § Tackling fake donors: Since a dynamic algorithm for choosing the most eligible donor is followed, which takes into account the willingness factor of each donor, fake donors will automatically be relegated to the bottom of the donor list.
- § Handling large user traffic: To handle large user traffic parallel programming is employed. When a higher demand for Plasma is needed, by pressing a limit of 5 donors is kept for people with constant demand for Plasma, like people with blood cancer, a separate registry of their phone numbers is kept thereby avoiding intermediaries from creating artificial scarcity.

REFERENCE:

[1] Javed Akhtar Khan and M.R. Alony, “A New Concept of Blood Bank Management System using Cloud Computing for Rural Area (INDIA)”, TIT Group of Institute of Engineering, Bhagwant University Ajmer, (RJ) INDIA, International Journal of Electrical, Electronics.

[2] A. Clemen Teena, K. Sankar and S. Kannan, “A Study on Blood Bank Management”, Department of MCA, Bharath University, Selaiyur, Chennai-73, Tamil Nadu, India, Middle-East Journal of Scientific Research 19 (8): 1123-1126, 2014, ISSN 1990-9233, DOI: 10.5829/idosi.mejsr.2014.19.8.11202

[3] K M Akkas Ali, Israt Jahan, Md. Ariful Islam, Md. Shafaat Parvez, “Blood Donation Management System”, Institute of Information Technology, Jahangirnagar University, Dhaka, Bangladesh, Department of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh.

[4] Aware Sachin B, Arshad Rashid, Ansari Adil, Bombale R.R., “Web Based Blood Donation System”.

[5] André Smith, Ralph Matthews, Jay Fiddler, “Blood Donation and Community: Exploring the Influence of Social Capital”, International Journal of Social Inquiry, Volume 4, Number 1, 2011 pp. 45-63.

[6] Shyam Sundaram, Santhana, “Real-Time Blood Donor Management Using Dashboards Based on Data Mining Models”, Dept. of Computer Science, DG Vaishnav College Chennai 600106, Tamil Nadu, India.

[7] Sultan Turhan, “An Android Application for Volunteer Blood Donors”, Department of Computer Engineering, Galatasaray University, Istanbul, TURKEY, url:sturhan@gsu.edu.tr

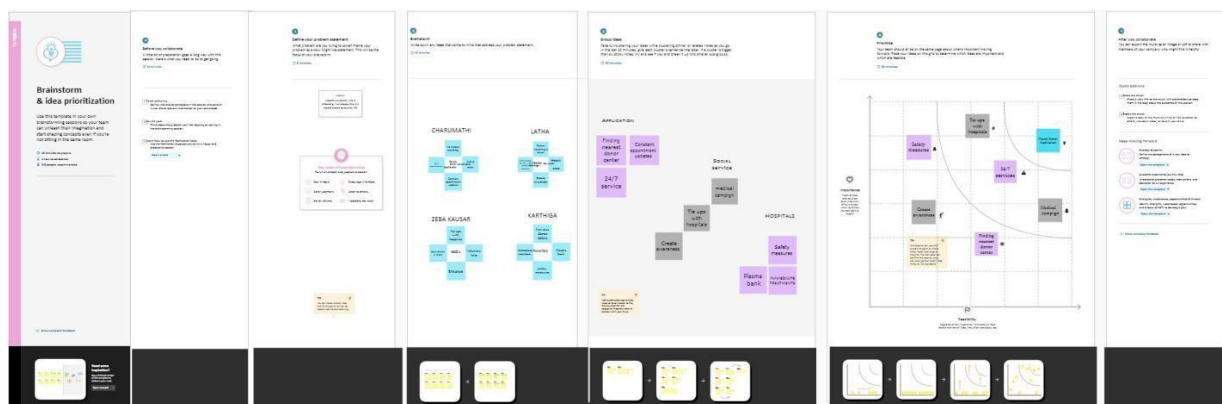
PROBLEM STATEMENT

This project is to interconnect all the blood banks, hospitals, donors into a single network, validation, store various data and information of plasma of each individual. This system is used to store data over a centralized server which consists of database where the individual's information cannot be accessed by a third party. Plasma donation is not only a noble service, it is also good for our health and does not have any side effects on the body. For people who donate blood regularly, the chances of getting heart diseases and high blood pressure are greatly reduced. It also helps in countering the problem of obesity, the minister added. No other work is more sacred than this. Blood donation is an excellent service .

3.2

IDEATION

AND



3.3 Proposed Solution :

Project team shall fill the following information in proposed solution template.

S. N o.	parameter	Description
1.	Problem Statement (Problem to be solved)	To develop a cloud based application to take plasma donor details, store them and inform them upon a request.
2.	Idea / Solution description	The project idea is to store user details upon registration and using that find eligible, verified and nearby donor. May include hospital details to perform the donation .
3.	Novelty / Uniqueness	Nearby and verified donors are shown according to particular user. Progress of the donation process is shown.
4.	Social Impact / Customer Satisfaction	Facilitates more plasma treatment counts to reduce covid-19. By the presence of only covid- negative certified donors, the user seeking plasma treatment has better trust and confidence.

5.	Business Model (Revenue Model)	Revenue generation can be through user registration under the non-donor field.
6.	Scalability of the Solution	Cloud Application allows flexible horizontal and vertical scaling.

4. REQUIREMENT ANALYSIS

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 Registration through Gmail Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Login	Enter user name and password
FR-4	Adding User essential data	Link Bank account Enter Hard cash expenses Set budget limit
FR-5	Alerts	Email Alert

Non-functional Requirements:

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

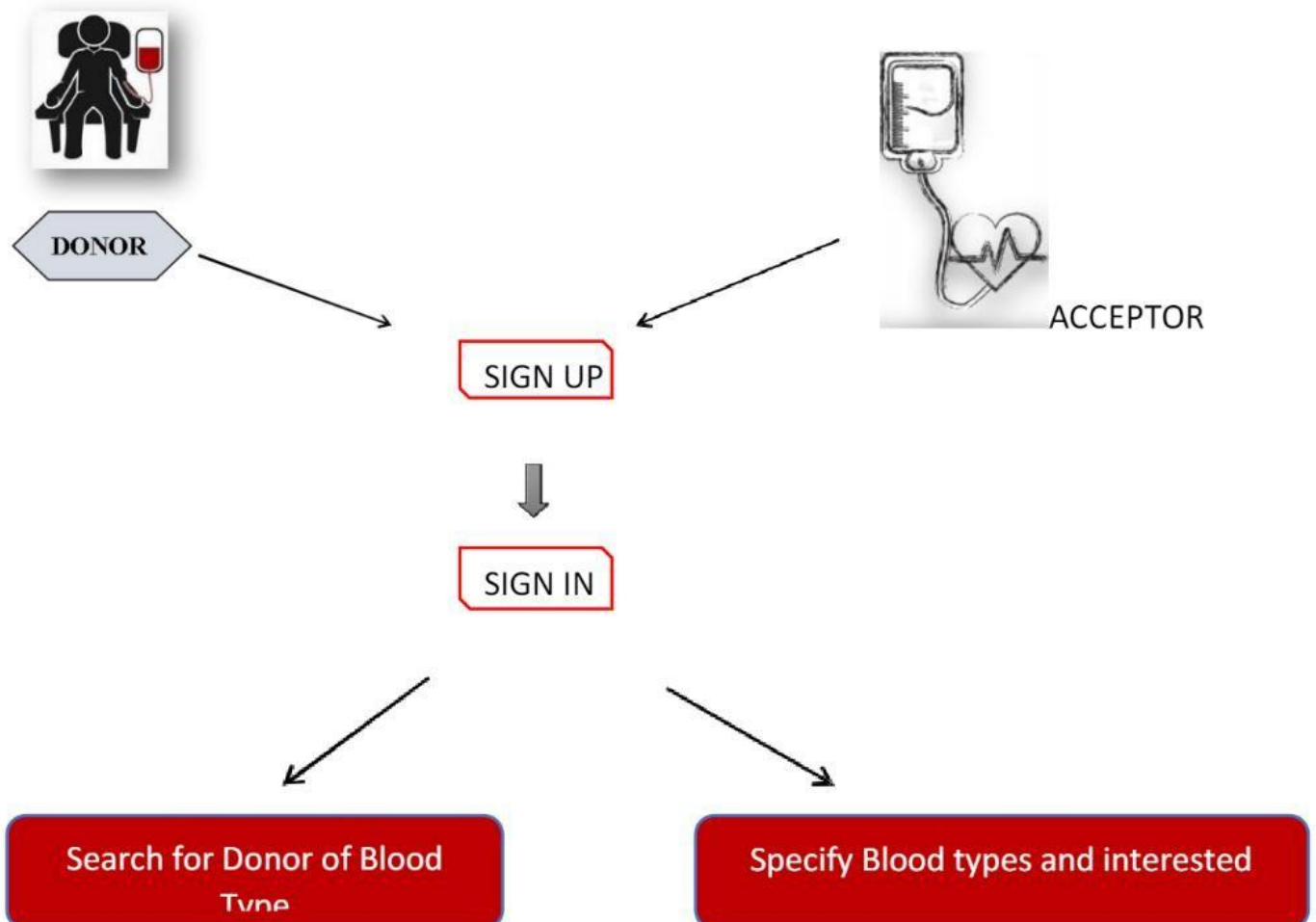
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Fast and Precise Budget Planning
NFR-2	Security	Resilient and data are immutable
NFR-3	Reliability	Failure-free operations and proper alerts are generated.
NFR-4	Performance	Instant e-mail alerts and generation of accurate expense-time graph.
NFR-5	Availability	User can access all the functionality of the application for 3 months after that they need to pay for premium account with all facility.
NFR-6	Scalability	This application can be further upscaled by having a separate section for Auditing/Accounting.

5. PROJECT DESIGN

Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is set.

DATA FLOW DIAGRAM





CALL DONARS



FEEDBACK



UPDATE PROFILE



RECEIVE CALL

TECHNOLOGY ARCHITECTURE:

SNO	Component Description	Description	Technology
1	User Interface	The interaction between the user and application	HTML, CSS, JavaScript / Bootstrap etc.
2	App Logic-1	Framework used for designing the application.	Python, Python - Flask
3	App Logic-2	Accessing the cloud and storing details of the users both donors' patients.	IBM Cloud, IBM DB2
4	App Logic-3	Docker is an open-source platform for building, deploying, and managing	Docker

User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Prioriy	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Spint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Spri nt-1

		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Spri nt-2
--	--	-------	--	---	-----	-----------

		USN-4	As a user, I can register for the application through G mail	I can receive confirmati on email and click confirm	Medi um	Spri nt-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can enter into my account	High	Spri nt-1
	Dashboard	USN-6	As a user, Display all details about plasma application	I can donate or get details about the plasma	High	Spri nt-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	Can keep track of the availability of Plasma	I can provide applications for customer needs	High	Sprint-4
Administrator	Verify donor details	USN-9	To add the donor plasma details in application	Can control 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	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 the application	Can alert notification through email and SMS	High	Sprint-2

Administrat or	Performan ce of applicatio n	USN-1 2	To make the process more efficient	Can save time, cost by improving the Plasma management application	High	Sprint-4
----------------	---------------------------------------	------------	--	--	------	----------

6. PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

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	30	6 Days	25 Oct 2022	30 Oct 2022	30	30 Oct 2022
Sprint-2	30	6 Days	1 Nov 2022	6 Nov 2022	30	6 Nov 2022
Sprint-3	30	6 Days	8 Nov 2022	13 Nov 2022	30	13 Nov 2022
Sprint-4	30	6 Days	15Nov 2022	20 Nov 2022	30	20 Nov 2022

6.2 REPORTS FROM JIRA:

When JIRA sends either standard notifications or user invitations to a mail server, they are listed as when the applicant in need of plasma.





















The mail server is receiving a constant stream of concurrent multiple email notifications from the same sender which, in turn, triggers security measures on the server which handle these messages, And this application is widely used in Health care industry.

Resolution

- Check the base url of JIRA to see if it is set as a direct ip address with port number.
- Example: **http://10.10.10.10:8080**
- Some email servers (such as Microsoft Outlook) will consider messages from non-DNS urls as phishing attempts. You can correct this behaviour by setting JIRA's base url to a url address such as **http://my-jira.com**.
- Sometimes when certain mail servers receive multiple emails from the same sender security measures are triggered that will then list those emails as phishing messages. For this, it is best to check with the local mail server administrator for further assistance and confirmation.

Open issues in JIRA

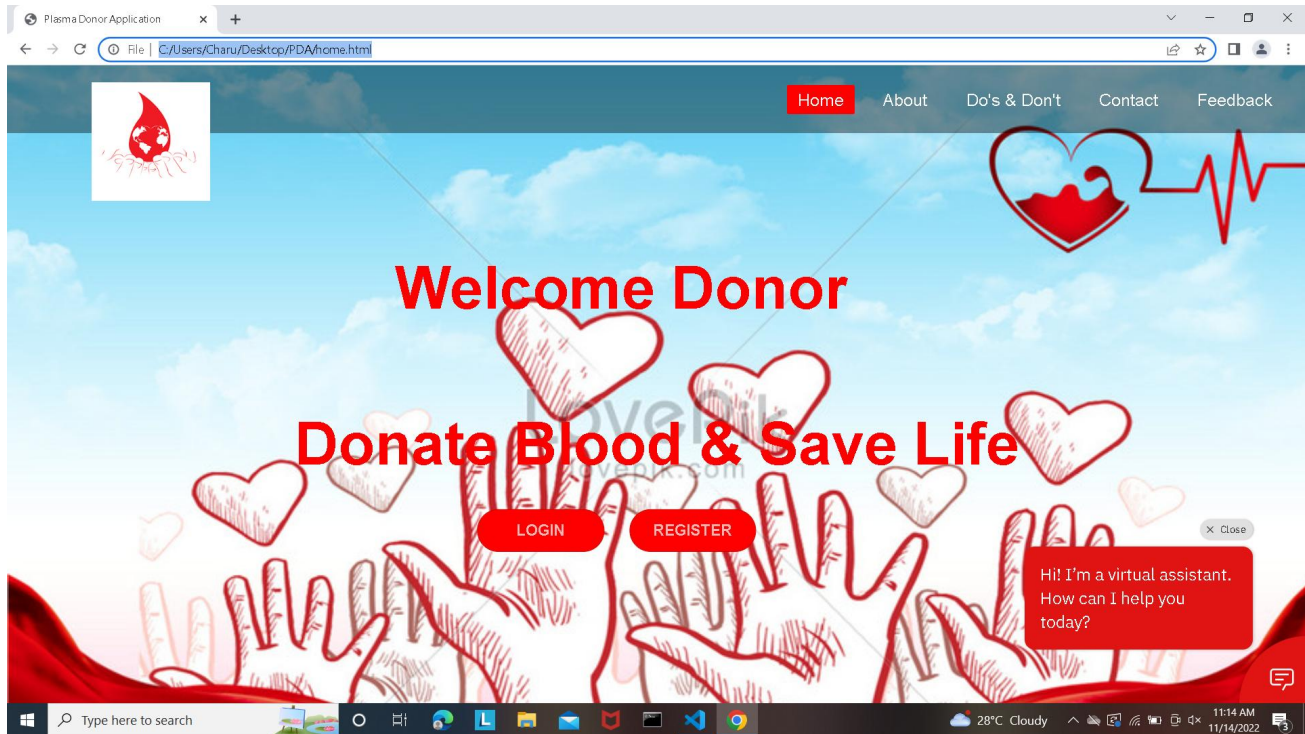
Created by Atlassian OnDemand [Administrator], last modified 3 minutes ago

Key	Summary	T	Created	Updated	Due	Assignee	P	Status	Resolution
WTP-10	Back end		Dec 30, 2016	Dec 30, 2016		Unassigned		TO DO	Unresolved
WTP-9	Front end		Dec 30, 2016	Dec 30, 2016		Unassigned		TO DO	Unresolved
WTP-8	Investigate vendor options		Dec 22, 2016	Dec 30, 2016		Unassigned		TO DO	Unresolved
WTP-7	Gather user metrics		Dec 22, 2016	Dec 30, 2016		Unassigned		TO DO	Unresolved
WTP-6	Website's color scheme is wrong		Dec 22, 2016	Dec 30, 2016		Unassigned		TO DO	Unresolved
WTP-5	Back button leads to 404 page		Dec 22, 2016	Dec 30, 2016		Unassigned		TO DO	Unresolved
WTP-4	Create new home page		Dec 22, 2016	Dec 30, 2016		Unassigned		TO DO	Unresolved
WTP-3	As a user, the first page I see is the home page		Dec 22, 2016	Dec 30, 2016		Jose		TO DO	Unresolved
WTP-2	As an external client, I can navigate to the pages most important to me through my profile		Dec 22, 2016	Dec 30, 2016		Alana		TO DO	Unresolved
WTP-1	Hide the context menu when user right-clicks on an issue		Dec 22, 2016	Dec 30, 2016		Will		TO DO	Unresolved

10 issues  Refresh

7. CODING AND SOLUTIONING

7.1 FEATURE 1



EXPLANATION:

The Chatbot is integrated with the home page so that the user can pass their queries in it and it will answered for the necessary request.

7.1 FEATURE 2

First Name	Last Name	Age	Gender	Blood Group	Email Address	phone number	Address
samba	rafiq	36	Female	O	rafiqsamba1902@gmail.com	623-598-1255	Mahalakshmi Nagar
Joseph	Austin	40	Male	B	joaus56@gmail.com	987-123-6548	Agaramthen
Catherine	Gerald	25	Female	B	cath22@gmail.com	895-263-7745	Selayur
Charu	Kumar	20	Female	A	charu1902@gmail.com	959-795-9722	Chromepet

EXPLANATION:

When the user request for plasma for the needed blood group, the available donor details are sent to the user as a notification.

8. TESTING

8.1 TEST CASE:

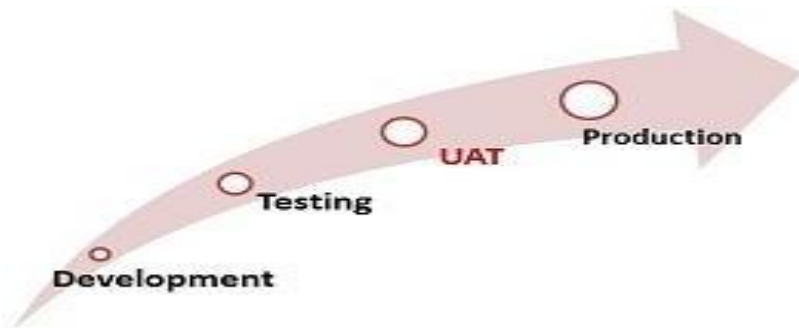
For the URL verifier module in the ISOT phishing detection system, phishing detection is done using 16 different heuristic rules. In the system, 11 main classes were defined, and 1 class was defined with 5 sub-classes. This covers all 16 heuristic rules. To test the system, 15 test cases were designed using assertion methods. Ten test cases were designed to test the 10 main classes and 5 test cases were designed to test the class with five sub-classes. The getter-setter method was used to test the class with five sub-classes. The getter method is used to obtain or retrieve a variable value from the class, and the setter method is used to store the variables.

The class with five sub-classes checks the 5 different heuristic rules, length of the URL, number of dots and slashes in the URL, presence of @ symbols in the URL, IP address mentioned in the URL, and the presence of special character such as ',', '_', ';' in the URL. Initially, only a single test case was created for the class with five sub-classes, but it was failing as this class has five methods as shown in Figure 4.11. After applying the getter setter method, all the test cases passed without any issues. The test results are shown in Figure 4.12. `assert Not Null()` is used to check if the input URL is not empty, and `assert Array Equals()` is used to compare the result from the detection method with the expected result.

[illegible]

8.2 USER ACCEPTANCE TESTING:

User Acceptance Testing (UAT) is a type of testing performed by the end user or the client to verify/accept the software system before moving the software application to the production environment. UAT is done in the final phase of testing after functional, integration and system testing is done.



The main **Purpose of UAT** is to validate end to end business flow. It does not focus on cosmetic errors, spelling mistakes or system testing. User Acceptance Testing is carried out in a separate testing environment with production-like data setup. It is kind of black box testing where two or more end-users will be involved.

UAT is performed by –

- Client
- End users

2. 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	11	4	2	2	19
Duplicate	1	1	2	0	4
External	1	4	0	1	6
Fixed	10	2	2	20	34
Not Reproduced	0	0	2	0	2
Skipped	0	0	2	1	3
Won't Fix	0	5	2	1	8
Totals	23	16	12	25	76

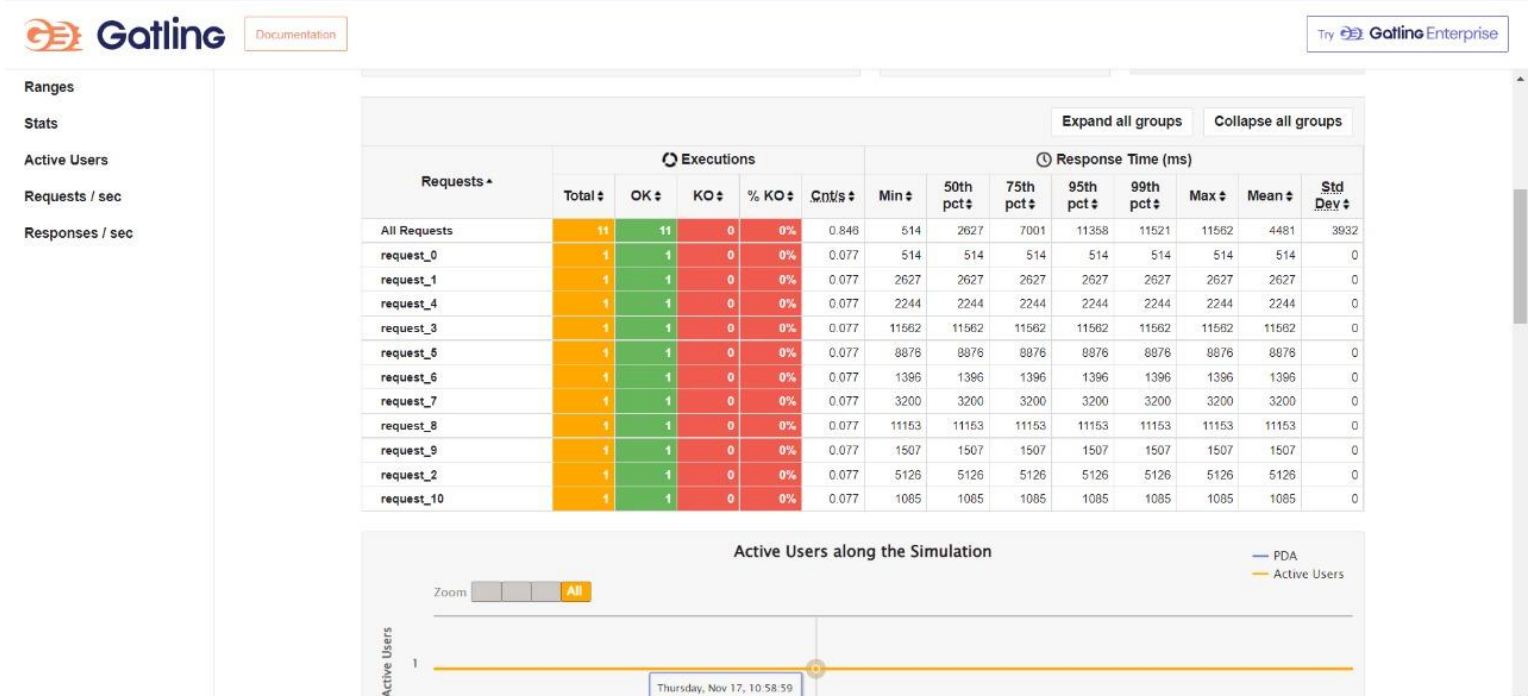
3. Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	5	0	1	4
Client Application	46	0	2	40
Security	3	0	0	3
Outsource Shipping	2	0	0	2
Exception Reporting	11	0	2	9
Final Report Output	5	0	0	5
Version Control	2	0	1	2

9. RESULTS

9.1 PERFORMANCE METRICS



request_0

request_1

request_4

request_3

request_5

request_6

request_7

request_8

request_9

request_2

request_10

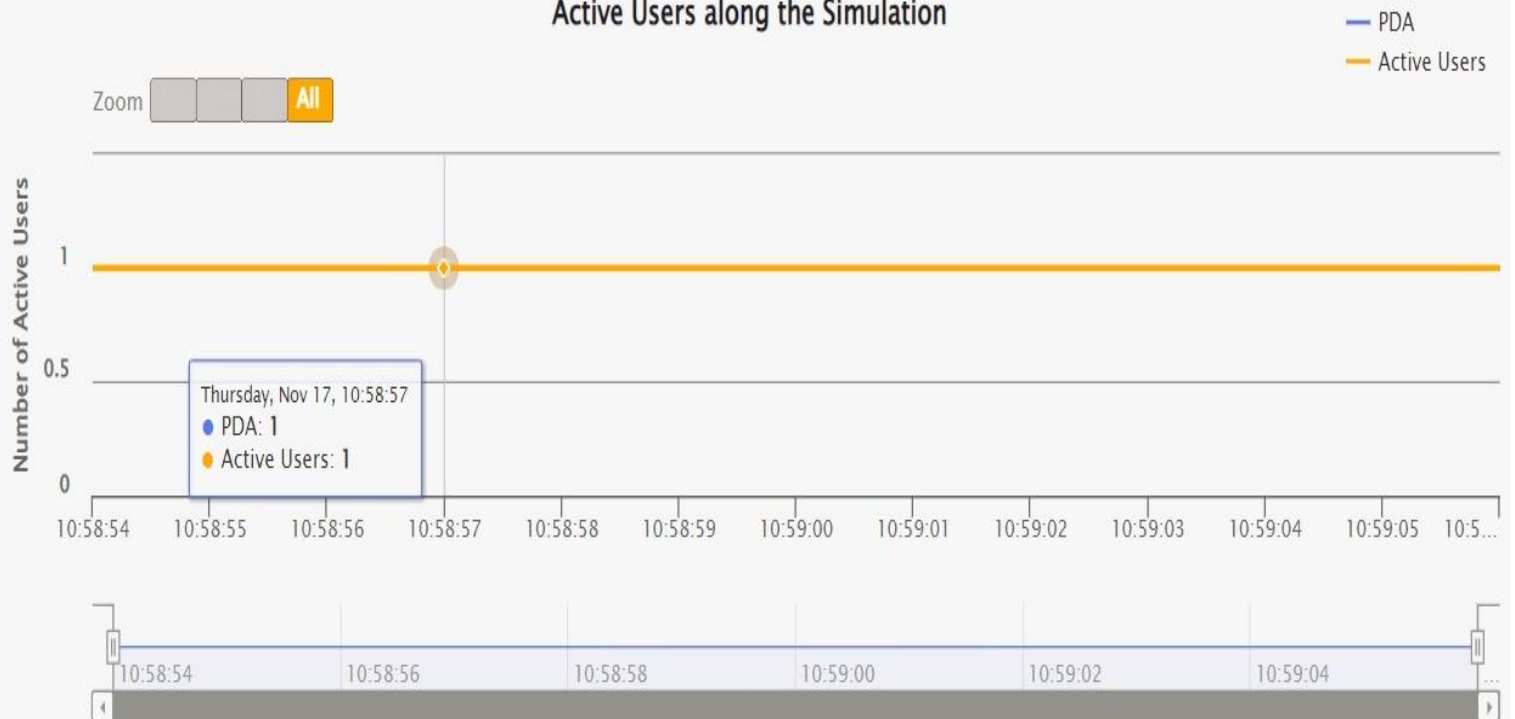
Response Time Distribution



Response Time Percentiles over Time (OK)



Active Users along the Simulation



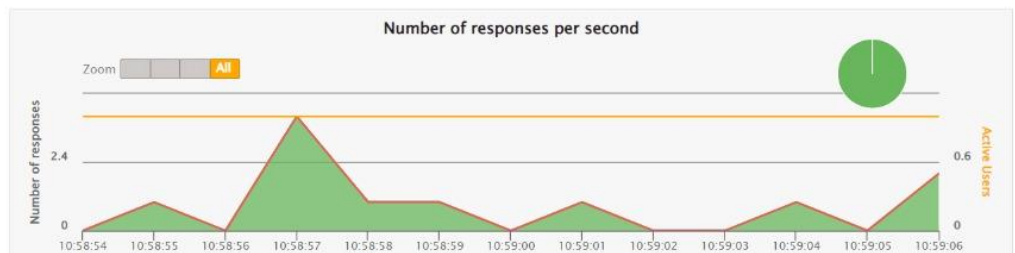
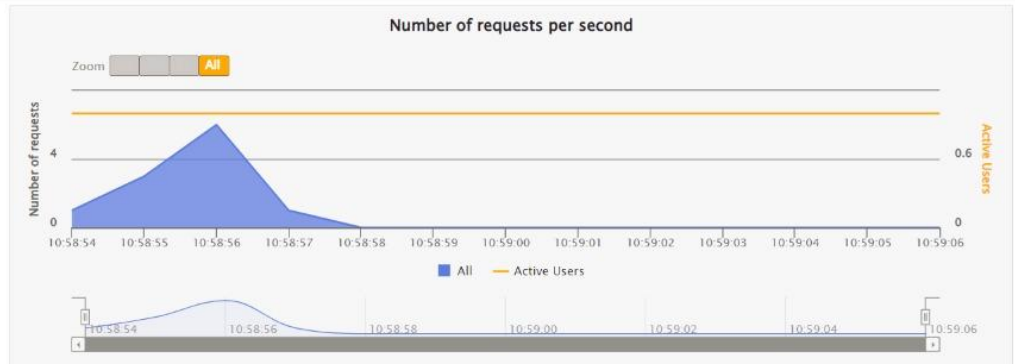
Ranges

Stats

Active Users

Requests / sec

Responses / sec



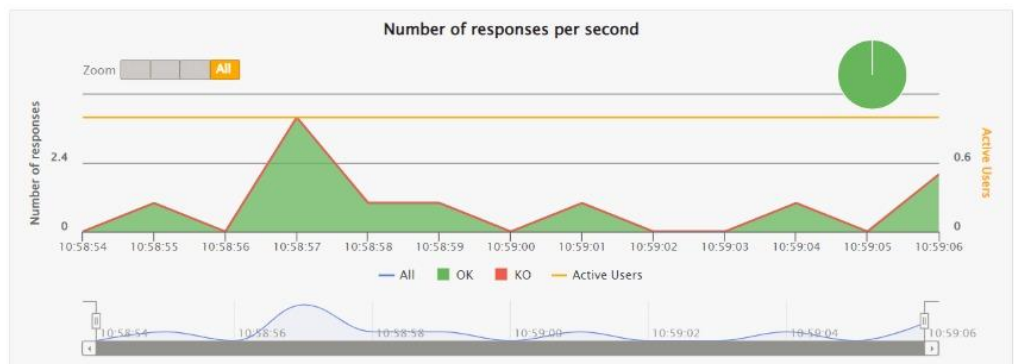
Ranges

Stats

Active Users

Requests / sec

Responses / sec



request_0

request_1

request_4

request_3

request_5

request_6

request_7

request_8

request_9

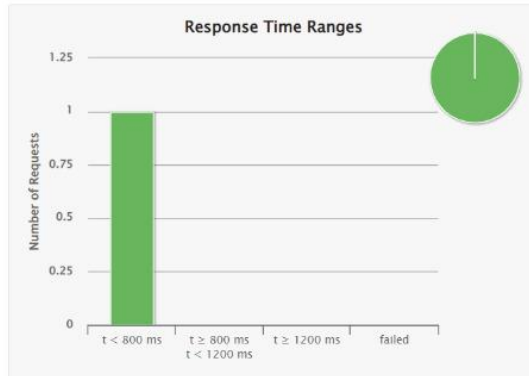
request_2

request_10

PDA

Global

Details



Stats

Executions

	Total	OK	KO
Total count	1	1	0
Mean count/s	0.077	0.077	-

Response Time (ms)

	Total	OK	KO
Min	514	514	-
50th percentile	514	514	-
75th percentile	514	514	-
95th percentile	514	514	-
99th percentile	514	514	-
Max	514	514	-
Mean	514	514	-
Standard Deviation	0	0	-



request_0

request_1

request_4

request_3

request_5

request_6

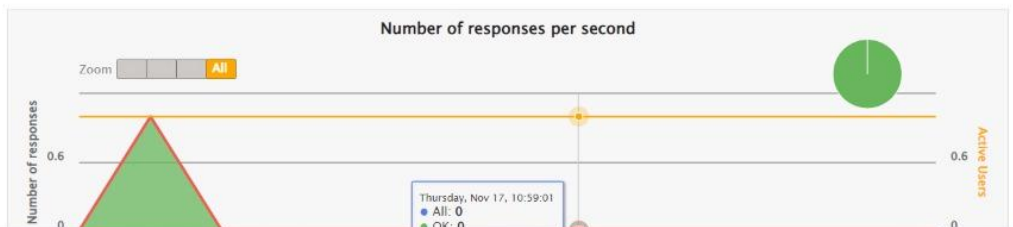
request_7

request_8

request_9

request_2

request_10



10.ADVANTAGES

- The project is identified by the merits of the system offered to the user. The merits of this project are as follows: -
- It's a web-enabled project.
- This project offers user to enter the data through simple and interactive forms. This is very helpful for the client to enter the desired information through so much simplicity.
- The user is mainly more concerned about the validity of the data, whatever he is entering. There are checks on every stages of any new creation, data entry or updation so that the user cannot enter the invalid data, which can create problems at later date.
- Sometimes the user finds in the later stages of using project that he needs to update some of the information that he entered earlier. There are options for him by which he can update the records. Moreover there is restriction for his that he cannot change the primary data field. This keeps the validity of the data to longer extent.

- User is provided the option of monitoring the records he entered earlier. He can see the desired records with the variety of options provided by him. From every part of the project the user is provided with the links through framing so that he can go from one option of the project to other as per the requirement. This is bound to be simple and very friendly as per the user is concerned. That is, we can say that the project is user friendly which is one of the primary concerns of any good project.
- Data storage and retrieval will become faster and easier to maintain because data is stored in a systematic manner and in a single database:
- The size of the database increases day-by-day, increasing the load on the database back up and data maintenance activity.
- Training for simple computer operations is necessary for the users working on the system. .
- Decision making process would be greatly enhanced because of faster processing of information since data collection from information available on computer takes much less time than manual system.
- Allocating of sample results becomes much faster because at a time the user can see the records of last years.
- Easier and faster data transfer through latest technology associated with the computer and communication.

- Through these features it will increase the efficiency, accuracy and transparency,

DISADVANTAGES:

The size of the database increases day-by-day, increasing the load on the database back up and data maintenance activity.

- Training for simple computer operations is necessary for the users working on the system.

11 .CONCLUSION

It has been a great pleasure for me to work on this exciting and challenging project. This project proved good for me as it provided practical knowledge of not only programming in ASP.NET and VB.NET web based application and no some extent Windows Application and SQL Server, but also about all handling procedure related with **“Blood Bequeath Federal”**. It also provides knowledge about the latest technology used in developing web enabled application and client server technology that will be great demand in future. This will provide better opportunities and guidance in future in developing projects independently. Technology is introducing new innovations day by day, thus reducing the time required to do things. The proposed system can be used to reduce the time required to deliver required blood to the needy in cases of emergency. The Web application can be used by the people interested in donating their blood by locating their nearest blood bank. The web application provides a way of communication and synchronization between the hospitals and the blood banks. It also provides them with the facility of communicating with the nearby donors in emergency. The database is a vital aspect of the system. The database of the hospitals and the blood banks must be checked for consistency on regular basis for smooth working of the system.

12. FUTURE ENHANCEMENT

The proposed system uses Google Maps which provides the user with an efficient way of locating the nearby donors/blood banks. The Web application is developed using Xampp server which is an open-source software, while the web application for the hospitals and the blood banks is also developed using open-source tools, hence the system developed is quite feasible. A module is a software component or part of a program that contains one or more routines. One or more independently developed modules make up a program. An enterprise-level software application may contain several different modules, and each module serves unique and separate business operations. Modules make a programmer's job easy by allowing the programmer to focus on only one area of the functionality of the software application. Modules are typically incorporated into the program (software) through interfaces.

Software applications include many different tasks and processes that cohesively serve all paradigms within a complete business solution. Early software versions were gradually built from an original and basic level, and development teams did not yet have the ability to use prewritten code. For example, Systems, Applications and Products in

Data Processing (SAP) - an enterprise resource planning (ERP) software - is comprised of several large modules (for example, finance, supply chain and payroll, etc.), which may be implemented with little or no customization. A classic example of a module-based application is Microsoft Word, which contains modules incorporated from Microsoft Paint that help users create drawings or figures. This module consists of the process of how the requests from recipients for the required blood are served. The Blood Bank first checks whether the request is a valid one. After validation it checks the hospital's database to ensure that the required amount of blood is not available in that hospital and after the request is served. The blood bank module also consists of requesting the blood when urgently needed from other banks and from the registered donors who have kept their status as available for further contact.

- Supporting visual impaired persons
- Online testing human healthy such as blood pressure, temperature, etc.
- Supporting all nation and nationalities languages.
- To do using digital signature technologies to make the system more secure.
- The system has to include video demo

13. APPENDIX

SOURCE CODE:

HTML FILES:

about.html:

```
<!DOCTYPE html>
<html>
<head>
  <title>About Us Section</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="https://pro.fontawesome.com/releases/v5.10.0/css/all.css">
  <link rel="stylesheet" type="text/css" href="about.css">
</head>
<body>
  <div class="section">
    <div class="container">
      <div class="content-section">
        <div class="title">
          <h1>About Us</h1>
        </div>
        <div class="content">
          <h3>PLASMA DONAR APPLICATION</h3>
          <p>A Donation of Blood[PLASMA] means a few minutes to you,
            but a Lifetime for somebody else <br> Plasma Donation is The Act Of Giving Blood To Someone Who Needs It.
            It Is Not Just About Giving Blood, But It Is An Act Of Kindness That Saves The Lives Of Hundreds Of People. These Fifteen Minutes
            Of Your Life Can Save Someone's Entire Life. You Can't Even Imagine That Donating One Bag Of Blood Can Be So Beneficial To The
            Human Race. <br>Donating The Blood Without Expecting Or Asking For Any Money Or Gesture Is A Great Act Of Kindness, And If You
            Are 18 Years Old Or Above, You Should Definitely Take Part In This Act Of Kindness.</p>
          <div class="button">
            <a href="home.html">GO BACK</a>
          </div>
        </div>
        <div class="social">
          <a href=""><i class="fab fa-facebook-f"></i></a>
          <a href=""><i class="fab fa-twitter"></i></a>
          <a href=""><i class="fab fa-instagram"></i></a>
        </div>
      </div>
      <div class="image-section">
        
      </div>
    </div>
  </div>
</body>
</html>
```

Contact.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=">
  <title>Contact Us Page</title>
<link rel="stylesheet" href="contact.css">
  <script src="https://kit.fontawesome.com/c32adfdcda.js" crossorigin="anonymous"></script>
</head>
<body>
<form action="home.html" method="POST">

  <section>

    <div class="section-header">
      <div class="container">
        <h2>Contact Us</h2>
        <p>Welcome Donor!! <br> “We make a living by what we get, but we make a life by what we give.” – Winston Churchill <br>
Do you feel you don’t have much to offer? You have the most precious resource of all: the ability to save a life by donating
blood! Help share this invaluable gift with someone in need.</p>
      </div>

      <div class="container">
        <div class="row">

          <div class="contact-info">
            <div class="contact-info-item">
              <div class="contact-info-icon">
                <i class="fas fa-home"></i>
              </div>

              <div class="contact-info-content">
                <h4>Address</h4>
                <p>1222 clkz road,<br/> sandar, ponmar, <br/>55060</p>
              </div>
            </div>

            <div class="contact-info-item">
              <div class="contact-info-icon">
                <i class="fas fa-phone"></i>
              </div>

              <div class="contact-info-content">
                <h4>Phone</h4>
                <p>959-795-9722</p>
              </div>
            </div>

            <div class="contact-info-item">
              <div class="contact-info-icon">
                <i class="fas fa-envelope"></i>
              </div>
```

```

        <div class="contact-info-content">
            <h4>Email</h4>
            <p>clkz1902@gmail.com</p>
        </div>
    </div>
</div>

<div class="contact-form">
    <form action="" id="contact-form">
        <h2>Send Message</h2>
        <div class="input-box">
            <input type="text" required="true" name="">
            <span>Full Name</span>
        </div>

        <div class="input-box">
            <input type="email" required="true" name="">
            <span>Email</span>
        </div>

        <div class="input-box">
            <textarea required="true" name=""></textarea>
            <span>Type your Message...</span>
        </div>

        <div class="input-box">
            <input type="submit" value="Send" name="">
        </div>
    </form>
</div>

</div>
</div>
</section>
</form>
</body>
</html>

```

dd.html:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link href="https://fonts.googleapis.com/css2?family=Roboto:wght@400;700&display=swap" rel="stylesheet">
    <title>Tabs</title>
</head>
<body>
    <div class="mytabs">

```



```

<input type="radio" id="tabfree" name="mytabs" checked="checked">
<label for="tabfree">DO's</label>
<div class="tab">
  <h2>DO's</h2>
  <p>*Get a full night's sleep.
<br>*Have adequate food, including iron-rich foods as they help to maintain the hemoglobin level.
<br>*Make sure you're drinking enough water before donating blood.
<br>*Make sure you haven't had a cold or flu in at least 72 hours.
<br>*Don't smoke for at least two hours prior to donating blood.
<br>*Don't consume alcohol a day before blood donation.
<br>*Review the most recent qualifying conditions for donors.
<br>*Carry your donor identity card or any other form of identification.<br>*Eat well-balanced meals for the next 24 hours to
replenish your body. <br>*Wait at least five minutes after the process is completed before getting up.<br>
Ensure to eat after blood donation. *Let the stomach be not empty.<br>
*Increase fluid intake for the next 24 to 48 hours as it might take around 24 hours for the body to get the fluids retrieved post
blood donation.<br>
</p>
</div>

  <input type="radio" id="tabsilver" name="mytabs">
  <label for="tabsilver">DON'Ts</label>
  <div class="tab">
    <h2>DON'Ts</h2>
    <p>*Avoid consuming fizzy beverages or aerated drinks. <br>*Do not stand for long periods of time.<br>*For roughly five
hours, avoid strong lifting or pulling with the donated arm.<br>
*Avoid smoking for four hours, and alcohol for 24 hours.<br>
*Ensure to share your medical history to safeguard the patient who will receive the blood.</p>
  </div>

</div>
</body>

```

```

<style>
  body {
    background: url(https://media.istockphoto.com/id/1256555401/vector/blood-donation-
concept.jpg?s=612x612&w=0&k=20&c=OKES1lI31Ny0H4CTABQgayI230R2o4tvCEE8RkSULpI=);
    font-family: 'Roboto', sans-serif;
  }
  .mytabs {
    display: flex;
    flex-wrap: wrap;
    max-width: 600px;
    margin: 50px auto;
    padding: 25px;
  }
  .mytabs input[type="radio"] {
    display: none;
  }
  .mytabs label {
    padding: 25px;
    background: #e2e2e2;
    font-weight: bold;
  }

```

```

  .mytabs .tab {
    width: 100%;
    padding: 20px;
    background: #fff;
    order: 1;
    display: none;
  }

```

```

}
.mytabs .tab h2 {
  font-size: 3em;
}

```

```

.mytabs input[type='radio']:checked + label + .tab {
  display: block;
}

```

```

.mytabs input[type="radio"]:checked + label {
  background: #fff;
}
</style>
</html>

```

feedback.html:

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <link rel="stylesheet" href="feedback.css" />
  </head>
  <body>
    <!-- Design by Ildiesign - https://dribbble.com/shots/7292664-Survey-UI-Design -->
    <!-- Icons by - https://www.flaticon.com/authors/roundicons -->
    <div id="panel" class="panel-container">
      <strong>
        >How satisfied are you with our<br />
        application's performance?</strong>
      >
      <div class="ratings-container">
        <div class="rating">
          
          <small>Unhappy</small>
        </div>
        <div class="rating">
          
          <small>Neutral</small>
        </div>
        <div class="rating active">
          

```

```

        <small>Satisfied</small>
    </div>
</div>
<button class="btn" id="send">Send review</button>
</div>
</body>
<script src="feedback.js"></script>
</html>

```

home.html:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Plasma Donor Application</title>
    <link rel="stylesheet" href="{{ url_for('static', filename='CSS/style.css') }}">
</head>
<body>
    <script>
        window.watsonAssistantChatOptions = {
            integrationID: "3b214ae8-6eba-404e-85be-63772e4c2fef", // The ID of this integration.
            region: "eu-gb", // The region your integration is hosted in.
            serviceInstanceID: "3b4bf0dd-e329-4199-99b4-25f9ba1ee227", // 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>
    <div class="wrapper">
        <nav class="navbar">
            
            <ul>
                <li><a class="active" href="#">Home</a></li>
                <li><a href="{{url_for('about')}}">About</a></li>
                <li><a href="{{url_for('dd')}}">Do's & Don't</a></li>
                <li><a href="{{url_for('contact')}}">Contact</a></li>
                <li><a href="{{url_for('feedback')}}">Feedback</a></li>
            </ul>
        </nav>
        <div class="center">
            <h1>Welcome Donor</h1>
            <h2>Donate Blood & Save Life</h2>
            <div class="buttons">
                <a href="{{url_for('login')}}"><button>LOGIN</button></a>
                <a href="{{url_for('register')}}"><button class="btn2">REGISTER</button></a>
            </div>
        </div>
    </body>
</html>

```

last.html:

```
<!DOCTYPE html>
<html>
<head>

<meta name="viewport" content="width=device-width,initial-scale=1.0">
<title>Thank you!</title>
<link rel="stylesheet" href="last.css">
</head>
<body>
<center>
<form action="home.html" method="POST">
<div class="popup">

<h2>Thank you!</h2>
<p>Your Details have been successfully submitted.Thanks!</p>
<a href="home.html"><button type="button">OK</button></a>
</div>
</div>
</form>
</body>
</center>
</html>
```

login.html:

```
<!DOCTYPE html>
<html>
<head>
<title>Login Form</title>
<link rel="stylesheet" type="text/css" href="fontawesome/css/all.min.css">
<link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='CSS/Login.css') }}">
</head>
<script>
    window.watsonAssistantChatOptions = {
        integrationID: "3b214ae8-6eba-404e-85be-63772e4c2fef", // The ID of this integration.
        region: "eu-gb", // The region your integration is hosted in.
        serviceInstanceID: "3b4bf0dd-e329-4199-99b4-25f9ba1ee227", // 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>
<form action="signin.html" method="POST">
<div class="container">
```

```

<div class="header">
<h1>Login</h1>
</div>
<div class="main">
<form>
<span>
<i class="fa fa-user"></i>
<input type="text" placeholder="Username" name="">
</span><br>
<span>
<i class="fa fa-lock"></i>
<input type="password" placeholder="password" name="">
</span><br>
<a href="{{url_for('signin')}}"><button>SIGN IN</button></a>
</form>
</form>
</div>
</div>
</body>
</html>

```

msgsent.html:

```

<!DOCTYPE html>
<html>
<head>
<center>
<meta name="viewport" content="width=device-width,initial-scale=1.0">
<title>Thank you!</title>
<link rel="stylesheet" href="msgsent.css">
</head>
<body>
<form action="home.html" method="POST">
<div class="popup">

<h2>Be Calm & Wait Patiently...</h2><br>
<p>Message been sent successfully!!</p>
<a href="home.html"><button type="button">OK</button></a>
</div>
</div>
</form>
</body>
</center>
</html>

```

regdonor.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Registration Form</title>
  <link rel="stylesheet" href="regdonor.css">
</head>
<body>

<div class="wrapper">
  <div class="title">
    DONOR DETAILS
  </div>
  <div class="form">
    <div class="inputfield">
      <label>First Name</label>
      <input type="text" class="input">
    </div>
    <div class="inputfield">
      <label>Last Name</label>
      <input type="text" class="input">
    </div>
    <div class="inputfield">
      <label>Age</label>
      <input type="Age" class="input">
    </div>

    <div class="inputfield">
      <label>Gender</label>
      <div class="custom_select">
        <select>
          <option value="">Select</option>
          <option value="male">Male</option>
          <option value="female">Female</option>
        </select>
      </div>
    </div>
  </div>
```

```
    <div class="inputfield">
      <label>BloodGroup</label>
      <div class="custom_select">
        <select>
          <option value="">Select</option>
          <option value="O">O</option>
          <option value="A">A</option>
          <option value="B">B</option>
        </select>
      </div>
    </div>
```

```
    <div class="inputfield">
      <label>Email Address</label>
      <input type="text" class="input">
    </div>
```

```

<div class="inputfield">
  <label>Phone Number</label>
  <input type="text" class="input">
</div>
<div class="inputfield">
  <label>Address</label>
  <textarea class="textarea"></textarea>
</div>
<div class="inputfield">
  <label>Postal Code</label>
  <input type="text" class="input">
</div>
<div class="inputfield terms">
  <label class="check">
    <input type="checkbox">
    <span class="checkmark"></span>
  </label>
  <p>Agreed to terms and conditions</p>
</div>
<div class="inputfield">
  <a href="last.html"><input type="submit" value="Register" class="btn"></a>
</div>
</div>
</div>
</body>
</html>

```

register.html:

```

<!DOCTYPE html>
<html>
<head>
<title>Sign Up</title>
<link rel="stylesheet" type="text/css" href="{{ url_for('static', filename='CSS/register.css') }}">
</head>
<script>
  window.watsonAssistantChatOptions = {
    integrationID: "3b214ae8-6eba-404e-85be-63772e4c2fef", // The ID of this integration.
    region: "eu-gb", // The region your integration is hosted in.
    serviceInstanceID: "3b4bf0dd-e329-4199-99b4-25f9ba1ee227", // 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>
  <form action="home.html" method="POST">
<form>
<h1>Sign Up</h1>
<input type="text" name="" placeholder="Username">

```

```

<input type="password" name="" placeholder="Password">
<input type="password" name="" placeholder="Confirm Password">
<input type="text" name="" placeholder="Gender">
<input type="text" name="" placeholder="Age">
<input type="email" name="" placeholder="Email">
<input type="phone" name="" placeholder="Mobile no">

<a href="{{url_for('home')}}"><button>SIGN UP</button></a>
</form>
</form>
</body>
</html>

```

request.html:

```

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Donor Details</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" type="text/css" href="request.css">
</head>

<body>
  <form action="msgsent.html" method="POST">

```

```

<h1 align="center">DONOR DETAILS</h1>

```

```

<table table>
  <tr align="center">
    <th>First Name</th>
    <th>Last Name</th>
    <th>Age</th>
    <th>Gender</th>
    <th>Blood Group</th>
    <th>Email Address</th>
    <th>phone number</th>
    <th>Address</th>
  </tr>

  <tr align=center>
    <td>samba</td>
    <td>rafiq</td>
    <td>36</td>
    <td>Female</td>
    <td>O</td>
    <td>rafiqsamba1902@gmail.com</td>
    <td>623-598-1255</td>
    <td>Mahalakshmi Nagar</td>
    <td>
      <div class="container">
        <a href="msgsent.html"><button class=".btn">Message</button></a>
      </div>
    </td>
  </tr>

```



```

        </td>
    </tr>
    <tr align=center>
        <td>Joseph</td>
        <td>Austin</td>
        <td>40</td>
        <td>Male</td>
        <td>B</td>
        <td>joaus56@gmail.com</td>
        <td>987-123-6548</td>
        <td>Agaramthen</td>
        <td>
            <div class="container">
                <a href="msgsent.html"><button class=".btn">Message</button></a>
            </div>
        </td>
    </tr>
    <tr align=center>
        <td>Catherine</td>
        <td>Gerald</td>
        <td>25</td>
        <td>Female</td>
        <td>B</td>
        <td>cath22@gmail.com</td>
        <td>895-263-7745</td>
        <td>Selaiyur</td>
        <td>
            <div class="container">
                <a href="msgsent.html"><button class=".btn">Message</button></a>
            </div>
        </td>
    </tr>
    <tr align=center>
        <td>Charu</td>
        <td>Kumar</td>
        <td>20</td>
        <td>Female</td>
        <td>A1</td>
        <td>charu1902@gmail.com</td>
        <td>959-795-9722</td>
        <td>Chromepet</td>
        <td>
            <div class="container">
                <a href="msgsent.html"><button class=".btn">Message</button></a>
            </div>
        </td>
    </tr>
</table>

```

```

<br><br>

```

```

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

```

signin.html:

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device, initial-scale=1.0" />
    <link rel="stylesheet" href="signin.css">
    <title></title>
  </head>
  <style>
    body{
      background-image: url('https://png.pngtree.com/background/20210710/original/pngtree-blood-donation-poster-background-
material-picture-image_1005808.jpg');
      background-repeat: no-repeat;
      background-attachment: fixed;
      background-size: 100%, 100%;
    }
  </style>

  <body>
    <div class="container">
      <a href="regdonor.html"><button class="btn btn--primary btn1--block">REGISTER AS DONOR</button></a>
      <a href="request.html"><button class="btn btn--secondary btn2--block">REQUEST FOR PLASMA</button></a>
    </div>
  </body>
</html>
```

CSS FILES:

about.css

```
@import url('https://fonts.googleapis.com/css2?family=Poppins:wght@200;300;400&display=swap');
*{
  margin:0px;
  padding:0px;
  box-sizing: border-box;
  font-family: 'Poppins', sans-serif;
}
.section{
  width: 100%;
  min-height: 100vh;
  background-color: red;
}
.container{
  width: 80%;
  display: block;
  margin:auto;
  padding-top: 100px;
}
.content-section{
  float: left;
  width: 55%;
}
.image-section{
  float: right;
  width: 40%;
}
.image-section img{
  width: 100%;
  height: auto;
}
.content-section .title{
  text-transform: uppercase;
  font-size: 28px;
}
.content-section .content h3{
  margin-top: 20px;
  color:black;
  font-size: 21px;
}
.content-section .content p{
  margin-top: 10px;
  font-family: handwritten;
  font-size: 18px;
  line-height: 1.5;
}
.content-section .content .button{
  margin-top: 30px;
}
.content-section .content .button a{
  background-color: #3d3d3d;
  padding:12px 40px;
  text-decoration: none;
  color:#fff;
  font-size: 25px;
```

```

        letter-spacing: 1.5px;
    }
    .content-section .content .button a:hover{
        background-color: black;
        color:#fff;
    }
    .content-section .social{
        margin: 40px 40px;
    }
    .content-section .social i{
        color:black;
        font-size: 30px;
        padding:0px 5px;
    }
    .content-section .social i:hover{
        color:#3d3d3d;
    }
    @media screen and (max-width: 768px){
        .container{
            width: 80%;
            display: block;
            margin:auto;
            padding-top:50px;
        }
        .content-section{
            float:none;
            width:100%;
            display: block;
            margin:auto;
        }
        .image-section{
            float:none;
            width:100%;
        }
        .image-section img{
            width: 100%;
            height: auto;
            display: block;
            margin:auto;
        }
        .content-section .title{
            text-align: center;
            font-size: 19px;
        }
        .content-section .content .button{
            text-align: center;
        }
        .content-section .content .button a{
            padding:9px 30px;
        }
        .content-section .social{
            text-align: center;
        }
    }
}

```

contact.css

```
@import url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;600;700&display=swap');
```

```
* {  
  padding: 0;  
  margin: 0;  
  box-sizing: border-box;  
  font-family: 'Poppins', sans-serif;  
}
```

```
body {  
  min-height: 100vh;  
  width: 100%;  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  background-image: url(https://tse3.mm.bing.net/th?id=OIP.7D6IFnQ5eYDNiEDr-AQ9WQHads&pid=Api&P=0);  
  background-size: cover;  
  background-position: center;  
  position: relative;  
}
```

```
body::after {  
  content: "";  
  position: absolute;  
  top: 0;  
  left: 0;  
  height: 100%;  
  width: 100%;  
  background-color: rgba(0,0,0,0.8);  
}
```

```
section {  
  position: relative;  
  z-index: 3;  
  padding-top: 50px;  
  padding-bottom: 50px;  
}
```

```
.container {  
  max-width: 1080px;  
  margin-left: auto;  
  margin-right: auto;  
  padding-left: 20px;  
  padding-right: 20px;  
}
```

```
.section-header {  
  margin-bottom: 50px;  
  text-align: center;  
}
```

```
.section-header h2 {  
  color: #FFF;  
  font-weight: bold;  
  font-size: 3em;  
  margin-bottom: 20px;  
}
```

```
.section-header p {
```

```
color: #FFF;
}
```

```
.row {
  display: flex;
  flex-wrap: wrap;
  align-items: center;
  justify-content: space-between;
}
```

```
.contact-info {
  width: 50%;
}
```

```
.contact-info-item {
  display: flex;
  margin-bottom: 30px;
}
```

```
.contact-info-icon {
  height: 70px;
  width: 70px;
  background-color: white;
  text-align: center;
  border-radius: 50%;
}
```

```
.contact-info-icon i {
  font-size: 30px;
  line-height: 70px;
}
```

```
.contact-info-content {
  margin-left: 20px;
}
```

```
.contact-info-content h4 {
  color: red;
  font-size: 1.4em;
  margin-bottom: 5px;
}
```

```
.contact-info-content p {
  color: white;
  font-size: 1em;
}
```

```
.contact-form {
  background-color: black;
  padding: 40px;
  width: 45%;
  padding-bottom: 20px;
  padding-top: 20px;
}
```

```
.contact-form h2 {
  font-weight: bold;
  font-size: 2em;
  margin-bottom: 10px;
  color: black;
}
```

```
.contact-form .input-box {  
  position: relative;  
  width: 100%;  
  margin-top: 10px;  
}
```

```
.contact-form .input-box input,  
.contact-form .input-box textarea{  
  width: 100%;  
  padding: 5px 0;  
  font-size: 16px;  
  margin: 10px 0;  
  border: none;  
  border-bottom: 2px solid #333;  
  outline: none;  
  resize: none;  
}
```

```
.contact-form .input-box span {  
  position: absolute;  
  left: 0;  
  padding: 5px 0;  
  font-size: 16px;  
  margin: 10px 0;  
  pointer-events: none;  
  transition: 0.5s;  
  color: #666;  
}
```

```
.contact-form .input-box input:focus ~ span,  
.contact-form .input-box textarea:focus ~ span{  
  color: #e91e63;  
  font-size: 12px;  
  transform: translateY(-20px);  
}
```

```
.contact-form .input-box input[type="submit"]  
{  
  width: 100%;  
  background: red;  
  color: black;  
  border: none;  
  cursor: pointer;  
  padding: 10px;  
  font-size: 18px;  
  border: 1px solid red;  
  transition: 0.5s;  
}
```

```
.contact-form .input-box input[type="submit"]:hover  
{  
  background: #FFF;  
  color: red;  
}
```

```
@media (max-width: 991px) {  
  section {  
    padding-top: 50px;  
    padding-bottom: 50px;  
  }  
  
  .row {
```

```

    flex-direction: column;
  }

  .contact-info {
    margin-bottom: 40px;
    width: 100%;
  }

  .contact-form {
    width: 100%;
  }
}

```

feedback.css

```

@import url("https://fonts.googleapis.com/css?family=Muli&display=swap");
@import url("https://fonts.googleapis.com/css?family=Montserrat&display=swap");

* {
  box-sizing: border-box;
}

```

```

body {
  background-color: #fef9f2;
  font-family: "Montserrat", sans-serif;
  display: flex;
  justify-content: center;
  align-items: center;
  height: 100vh;
}

```

```

.panel-container {
  background-color: #fff;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.7);
  border-radius: 4px;
  font-size: 90%;
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
  padding: 30px;
  max-width: 400px;
  text-align: center;
}

```

```

.panel-container strong {
  line-height: 20px;
}

```

```

.panel-container p {
  margin: 25px 0;
  line-height: 20px;
}

```

```

.ratings-container {
  display: flex;
}

```



```
margin: 20px 0;
border: 0;
border-radius: 4px;
color: black;
cursor: pointer;
padding: 12px 30px;
transition: 5s;
}
```

```
.rating {
  cursor: pointer;
  flex: 1;
  padding: 20px;
  margin: 10px 5px;
}
```

```
.rating.active,
.rating:hover {
  border-radius: 4px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
color: red;
  cursor: pointer;
  padding: 12px 30px;
transition: 5s;
}
```

```
.rating img {
  width: 40px;
}
```

```
.rating small {
  color: #555;
  display: inline-block;
  margin: 10px 0 0;
}
```

```
.rating.active small,
.rating:hover small {
  color: black;
}
```

```
.btn {
  background-color: #302d2b;
  border: 0;
  border-radius: 4px;
  color: #fff;
  cursor: pointer;
  padding: 12px 30px;
transition: 5s;
}
```

```
.fa-heart {
  color: red;
  font-size: 30px;
  margin-bottom: 10px;
}
```

last.css

```
*{
margin: 0;
padding: 0;
box-sizing: border-box;
font-family: 'poppins',sans-serif;
background-image: url();
}

.container{
width: 100%;
height: 100vh;
background: #3c5077;
display: flex;
align-items: center;
justify-content: center;
}
```

```
.btn{
padding: 10px 60px;
background: #fff;
border: 0;
outline: none;
cursor: pointer;
font-size: 22px;
font-weight: 500;
border-radius: 30px;
}
```

```
.popup{
width: 400px;
background: #fff;
border-radius: 6px;
position: absolute;
top: 50%;
left: 50%;
transform: translate(-50% , -50%);
text-align: center;
padding: 0 30px 30px;
color: #333;
}
```

```
.popup img{
width:100px;
margin-top: -50px;
border-radius: 100%;
align-items: center;
box-shadow: 0 2px 5px rgba(0,0,0,0.2);
}
```

```
.popup h2{
align-items: center;
font-size: 38px;
font-weight: 500;
margin: 30px 0 10px;
}
```

```
.popup button{
width: 100%;
margin-top: 50px;
```

```
padding: 10px 0;
background: red;
color: #fff;
border: 0;
outline: none;
font-size: 18px;
border-radius: 4px;
align-items: center;
cursor: pointer;
box-shadow: 0 5px 5px rgba(0,0,0,0.2);
}
```

login.css

```
body{
font-family: sans-serif;
background-image: url(https://static.vecteezy.com/system/resources/previews/007/119/937/non_2x/world-blood-donor-concept-free-vector.jpg);
background-repeat: no-repeat;
overflow: hidden;
background-size: cover;
}
```

```
.container{
width: 380px;
margin: 7% auto;
border-radius: 25px;
background-color: rgba(0,0,0,0.2);
box-shadow: 0 0 17px #333;
}
```

```
.header{
text-align: center;
padding-top: 75px;
}
```

```
header h1{
color: red;
font-size: 45px;
margin-bottom: 80px;
}
```

```
.main{
text-align: center;
}
```

```
.main input, button{
width: 300px;
height: 40px;
border: none;
outline: none;
padding-left: 40px;
box-sizing: border-box;
font-size: 15px;
color: #333;
margin-bottom: 48px;
}
```

```
.main button{
padding-left:0;
background-color: red;
letter-spacing: 1px;
font-weight: bold;
margin-bottom: 50px;
}
```

```
.main button:hover{
box-shadow: 2px 2px 5px #555;
background-color: #7799d4;
}
```

```
.main input:hover{
box-shadow: 2px 2px 5px #555;
background-color: red;
}
```

```
.main span{
position: relative;
}
```

```
.main i{
position: absolute;
left: 15px;
color: #333;
font-size: 16px;
top: 2px;
}
```

msgsent.css

```
{
margin: 50;
padding: 50;
box-sizing: border-box;
background-image: url(https://static.vecteezy.com/system/resources/previews/001/874/132/original/abstract-geometric-white-
background-free-vector.jpg);
font-family: 'poppins',sans-serif;
}
```

```
.container{
width: 50%;
height: 50vh;
background: #3c5077;
display: flex;
align-items: center;
justify-content: center;
}
```

```
.btn{
padding: 10px 60px;
background: #fff;
border: 0;
outline: none;
cursor: pointer;
```

```
font-size: 22px;
font-weight: 500;
border-radius: 30px;
}
```

```
.popup{
width: 400px;
background: #fff;
border-radius: 6px;
position: absolute;
top: 50%;
left: 50%;
transform: translate(-50% , -50%);
text-align: center;
padding: 0 30px 30px;
color: #333;
}
```

```
.popup img{
width: 250px;
margin-top: -50px;
border-radius: 50%;
align-items: center;
box-shadow: 0 2px 5px rgba(0,0,0,0.2);
}
```

```
.popup h2{
align-items: center;
font-size: 38px;
font-weight: 500;
margin: 30px 0 10px;
}
```

```
.popup button{
width: 100%;
margin-top: 50px;
padding: 10px 0;
background: grey;
color: #fff;
border: 0;
outline: none;
font-size: 18px;
border-radius: 4px;
align-items: center;
cursor: pointer;
box-shadow: 0 5px 5px rgba(0,0,0,0.2);
}
```

regdonor.css

```
@import url('https://fonts.googleapis.com/css?family=Montserrat:400,700&display=swap');

*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: 'Montserrat', sans-serif;
}

body{
  padding: 0 10px;
  background-image: url('https://png.pngtree.com/thumb_back/fh260/back_our/20190621/ourmid/pngtree-aids-red-ribbon-banner-poster-
background-image_182671.jpg');
  background-repeat: no-repeat;
  background-attachment: fixed;
  background-size: 100%, 100%;
  margin: 0;
  padding: 0;
}

.wrapper{
  max-width: 500px;
  width: 100%;
  background: transparent;
  margin: 20px auto;
  box-shadow: 1px 1px 2px rgba(0,0,0,0.125);
  padding: 30px;
}
```

```
.wrapper .title{
  font-size: 24px;
  font-weight: 700;
  margin-bottom: 25px;
  color: red;
  text-transform: uppercase;
  text-align: center;
}
```

```
.wrapper .form{
  width: 100%;
}
```

```
.wrapper .form .inputfield{
  margin-bottom: 15px;
  display: flex;
  align-items: center;
}
```

```
.wrapper .form .inputfield label{
  width: 200px;
  color: black;
  margin-right: 10px;
  font-size: 14px;
}
```

```
.wrapper .form .inputfield .input,
.wrapper .form .inputfield .textarea{
  width: 100%;
}
```

```
outline: none;
border: 1px solid red;
font-size: 15px;
padding: 8px 10px;
border-radius: 3px;
transition: all 0.3s ease;
}
```

```
.wrapper .form .inputfield .textarea{
width: 100%;
height: 125px;
resize: none;
}
```

```
.wrapper .form .inputfield .custom_select{
position: relative;
width: 100%;
height: 37px;
}
```

```
.wrapper .form .inputfield .custom_select:before{
content: "";
position: absolute;
top: 12px;
right: 10px;
border: 8px solid;
border-color: red transparent transparent transparent;
pointer-events: none;
}
```

```
.wrapper .form .inputfield .custom_select select{
-webkit-appearance: none;
-moz-appearance: none;
appearance: none;
outline: none;
width: 100%;
height: 100%;
border: 0px;
padding: 8px 10px;
font-size: 15px;
border: 1px solid red;
border-radius: 3px;
}
```

```
.wrapper .form .inputfield .input:focus,
.wrapper .form .inputfield .textarea:focus,
.wrapper .form .inputfield .custom_select select:focus{
border: 1px solid black;
}
```

```
.wrapper .form .inputfield p{
font-size: 14px;
color: black;
}

.wrapper .form .inputfield .check{
width: 15px;
height: 15px;
position: relative;
display: block;
cursor: pointer;
}
```

```

.wrapper .form .inputfield .check input[type="checkbox"]{
  position: absolute;
  top: 0;
  left: 0;
  opacity: 0;
}
.wrapper .form .inputfield .check .checkmark{
  width: 15px;
  height: 15px;
  border: 1px solid black;
  display: block;
  position: relative;
}
.wrapper .form .inputfield .check .checkmark:before{
  content: "";
  position: absolute;
  top: 1px;
  left: 2px;
  width: 5px;
  height: 2px;
  border: 2px solid;
  border-color: transparent transparent black black;
  transform: rotate(-45deg);
  display: none;
}
.wrapper .form .inputfield .check input[type="checkbox"]:checked ~ .checkmark{
  background: black;
}

```

```

.wrapper .form .inputfield .check input[type="checkbox"]:checked ~ .checkmark:before{
  display: block;
}

```

```

.wrapper .form .inputfield .check{
  width: 15px;
  height: 15px;
  position: relative;
  display: block;
  cursor: pointer;
}
.wrapper .form .inputfield .check input[type="checkbox"]{
  position: absolute;
  top: 0;
  left: 0;
  opacity: 0;
}
.wrapper .form .inputfield .check .checkmark{
  width: 15px;
  height: 15px;
  border: 1px solid black;
  display: block;
  position: relative;
}
.wrapper .form .inputfield .check .checkmark:before{
  content: "";
  position: absolute;
  top: 1px;
  left: 2px;
  width: 5px;
}

```



```
height: 2px;
border: 2px solid;
border-color: transparent transparent black black;
transform: rotate(-45deg);
display: none;
}
.wrapper .form .inputfield .check input[type="checkbox"]:checked ~ .checkmark{
background: red;
}
```

```
.wrapper .form .inputfield .check input[type="checkbox"]:checked ~ .checkmark:before{
display: block;
}
```

```
.wrapper .form .inputfield .btn{
width: 100%;
padding: 8px 10px;
font-size: 15px;
border: 0px;
background: red;
color: black;
cursor: pointer;
border-radius: 3px;
outline: none;
}
```

```
.wrapper .form .inputfield .btn:hover{
background: #ffd658;
}
```

```
.wrapper .form .inputfield:last-child{
margin-bottom: 0;
}
```

```
@media (max-width:420px) {
.wrapper .form .inputfield{
flex-direction: column;
align-items: flex-start;
}
.wrapper .form .inputfield label{
margin-bottom: 5px;
}
.wrapper .form .inputfield.terms{
flex-direction: row;
}
}
```

register.css

```
/*{
margin: 0;
padding: 0;
font-family: sans-serif;
box-sizing: border-box;
}

body{
display: flex;
justify-content: center;
align-items: center;
flex-direction: column;
    height: 100%;
    width: 100%;
    background-image: linear-
gradient(rgba(0,0,0,0.5),rgba(0,0,0,0.5)),url(https://png.pngtree.com/thumb_back/fh260/background/20220217/pngtree-red-simple-
illustration-public-welfare-publicity-background-of-world-blood-donation-image_952530.jpg);
    background-position: cover;
    background-size: cover;
    position: absolute;
}
```

```
form{
margin-top: 50px;
text-align: center;
}
```

```
input{
display: block;
width: 350px;
height: 40px;
margin: 20px;
border: none;
outline: none;
font-size: 20px;
border-bottom: 1px solid black;
background-size: cover;
background: transparent;
color: black;
}
```

```
button{
width: 350px;
height: 40px;
font-size: 20px;
background-color: red;
border-radius: 30px;
color: black;
margin-top: 20px;
transition: .5s;
}

.buttons button:hover{
    background: #cc0000;
}
```

```
form h1{
    margin-bottom: 30px;
    color: black;
}
```

```
::placeholder{
color: black;
}
```

request.css

```
*,
{
margin: 110;
padding: 110;
font-family: sans-serif;
box-sizing: border-box;
}

body{
display: flex;
justify-content: center;
align-items: center;
flex-direction: column;
height: 100%;
width: 100%;
background-image: linear-
gradient(rgba(0,0,0,1),rgba(0,0,0,1)),url(https://static.vecteezy.com/system/resources/previews/007/849/061/original/world-blood-
donor-background-free-vector.jpg);
background-position: absolute;
background-size: cover;
position: absolute;
}

table{
margin-top: 50px;
text-align: left;
width: 40px;
border-spacing: 40px;
}

.btn{
width: 5%;
height: 20px;
font-size: 15px;
margin: 1rem 0;
background-color: red;
border-radius: 30px;
color: white;
cursor: pointer;
display: inline-block;
position: fixed;
transition: .5s
}

.btn .btn--block:hover{
background: white;
transition: 0.2s;
}

table h1{
```

```
margin-bottom: 50px;
color: white;
}

::placeholder{
color: white;
}
```

signin.css

```
body{
margin: 0;
padding: 0;
}

.btn{
border-radius: 40px;
border: 0;
font-size: 1.8rem;
font-weight: 600;
margin: 1rem 0;
padding: 2rem 3rem;
text-transform: uppercase;
white-space: nowrap;
cursor: pointer;
}

.btn--primary{
background: transparent;
color: rgb(0, 15, 0);
}

.btn--primary:hover{
background: transparent ;
}

.btn--secondary{
background: transparent;
color: rgb(0, 0, 1);
}

.btn--secondary:hover{
background: transparent;
}

.btn1--block:hover{
color: #faffffa;
}

.btn2--block:hover{
color: #faffffa;
}

.btn1--block{
width: 50%;
display: inline-block;
position: fixed;
left: 5%;
top: 37%;
}

.btn2--block{
width: 50%;
display: inline-block;
```

```

    position: fixed;
    left: 50%;
    top: 37%;
}
#container{
    text-align: center;
}
@media screen and (min-width: 1024px) {
    .btn {
        font-size: 1.8rem;
    }
}

```

style.css

```

@import url('https://fonts.googleapis.com/css?family=Roboto:250&display=swap');
*{
    padding: 0;
    margin: 0;
}
.wrapper{
    background: url(https://img.lovepik.com//back_pic/05/64/03/535b618c44b9dde.jpg_wh860.jpg) no-repeat;
    background-size: cover;
    height: 100vh;
}
.navbar{
    position: fixed;
    height: 80px;
    width: 100%;
    top: 0;
    left: 0;
    background: rgba(0,0,0,0.4);
}
.navbar .logo{
    width: 140px;
    height: auto;
    padding: 20px 100px;
}
.navbar ul{
    float: right;
    margin-right: 20px;
}
.navbar ul li{
    list-style: none;
    margin: 0 8px;
    display: inline-block;
    line-height: 80px;
}
.navbar ul li a{
    font-size: 20px;
    font-family: 'Roboto', sans-serif;
    color: white;
    padding: 6px 13px;
    text-decoration: none;
    transition: .4s;
}
.navbar ul li a.active,
.navbar ul li a:hover{

```

```
background: red;
border-radius: 2px;
}

.wrapper .center{
  position: absolute;
  left: 50%;
  top: 55%;
  transform: translate(-50%, -50%);
  font-family: sans-serif;
  user-select: none;
}

.center h1{
  color: red;
  font-size: 70px;
  width: 900px;
  font-weight: bold;
  text-align: center;
margin-top: 1px;
}

.center h2{
  color: red;
  font-size: 70px;
  font-weight: bold;
  margin-top: 100px;
  width: 985px;
  text-align: center;
}

.center .buttons{
  margin: 35px 280px;
}

.buttons button{
  height: 50px;
  width: 150px;
  font-size: 18px;
  font-weight: 600;
  color: #ffb3b3;
  background: red;
  outline: none;
  cursor: pointer;
  border: 1px solid #cc0000;
  border-radius: 25px;
  transition: .4s;
}

.buttons .btn2{
  margin-left: 25px;
}

.buttons button:hover{
  background: #cc0000;
}
```

PHP FILES:

register.php

```
<?php
if (isset($_POST['signup'])) {
    if (isset($_POST['Username']) && isset($_POST['Password']) &&
        isset($_POST['Gender']) && isset($_POST['Age']) &&
        isset($_POST['Email']) && isset($_POST['Mobile no'])) {

        $username = $_POST['Username'];
        $password = $_POST['Password'];
        $gender = $_POST['Gender'];
        $age = $_POST['Age'];
        $email = $_POST['Email'];
        $phoneCode = $_POST['Mobile no'];
        $host = "localhost";
        $dbUsername = "root";
        $dbPassword = "";
        $dbName = "test";
        $conn = new mysqli($host, $dbUsername, $dbPassword, $dbName);
        if ($conn->connect_error) {
            die('Could not connect to the database.');
```

```
        }
        else {
            $Select = "SELECT email FROM register WHERE email = ? LIMIT 1";
            $Insert = "INSERT INTO register(Username, Password, Gender, Age, Email, Mobile no) values(?, ?, ?, ?, ?, ?)";
            $stmt = $conn->prepare($Select);
            $stmt->bind_param("s", $email);
            $stmt->execute();
            $stmt->bind_result($resultEmail);
            $stmt->store_result();
            $stmt->fetch();
            $rnum = $stmt->num_rows;
            if ($rnum == 0) {
                $stmt->close();
                $stmt = $conn->prepare($Insert);
                $stmt->bind_param("ssssii",$Username, $Password, $Gender, $Age, $Email, $Mobile no);
                if ($stmt->execute()) {
                    echo "New record inserted sucessfully.";
                }
                else {
                    echo $stmt->error;
                }
            }
            else {
                echo "Someone already registers using this email.";
            }
            $stmt->close();
            $conn->close();
        }
    }
    else {
        echo "All field are required.";
        die();
    }
}
else {
    echo "Signup button is not set";
}
?>
```

JS FILES:

feedback.js

```
const ratingsEl = document.querySelectorAll(".rating");
const sendBtn = document.querySelector("#send");
const panel = document.querySelector("#panel");
```

```
ratingsEl.forEach((el) => {
  el.addEventListener("click", () => {
    ratingsEl.forEach((innerEl) => {
      innerEl.classList.remove("active");
    });
  });
});
```

```
    el.classList.add("active");
  });
});
```

```
sendBtn.addEventListener("click", () => {
  panel.innerHTML = `
    <i class="fas fa-heart"></i>
    <strong>Thank you, Donor!</strong>
    <p>We'll use your feedback to improve our application's performance.</p>
    <a href="home.html"><button class="btn">Done</button></a>
  `;
});
```

PYTHON FILES:

app.py

```
from flask import Flask, render_template, request, redirect, url_for, session

import ibm_db
import bcrypt
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=19af6446-6171-4641-8aba-
9dcff8e1b6ff.clogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30699;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;
UID=dkz79110;PWD=VnQyUFTQ0JNLLLGq",'','')
```

```
app = Flask(__name__)
app.secret_key=b'_5#y2L"F4Q8z\n\xec]/'
```

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

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

```
@app.route('/login',methods=['POST','GET'])
def login():
    if request.method=='POST':
        username = request.form['username']
        password = request.form['password']
```



```
print("entered into post")
```

```
if not username or not password:
    return render_template('login.html', error='please fill all fields')
sql = "select * from user 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)
dic = ibm_db.fetch_assoc(stmt)
print(login)
if login:
    return redirect(url_for('signin'))
```

```
else:
    return redirect(url_for('login'))
```

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

```
if request.method=='POST':
    username = request.form['username']
    password = request.form['password']
    confirmpassword = request.form['confirm password']
    gender = request.form['gender']
    age = request.form['age']
    email = request.form['email']
    mobileno = request.form['mobile no']
```

```
if not username or password or confirmpassword or gender or age or email or mobileno:
    return render_template('register.html', error = 'Please fill all the fields')
hash=bcrypt.hashpw(password.encode('utf-8'),bcrypt.gensalt())
```

```
query = "SELECT * FROM user WHERE username=? OR password=?"
stmt = ibm_db.prepare(conn,query)
ibm_db.bind_param(stmt,1,username)
ibm_db.bind_param(stmt,2,password)
ibm_db.execute(stmt)
isuser=ibm_db.fetch_assoc(stmt)
if not isuser:
    sql = "INSERT INTO user(username, password, confirmpassword, gender, age, email, mobileno) VALUES(?,?,?,?,?,?,?)"
    prep_stmt = ibm_db.prepare(conn,sql)
    ibm_db.bind_param(prepare_stmt,1,username)
    ibm_db.bind_param(prepare_stmt,2,password)
    ibm_db.bind_param(prepare_stmt,3,confirmpassword)
    ibm_db.bind_param(prepare_stmt,4,gender)
    ibm_db.bind_param(prepare_stmt,5,age)
    ibm_db.bind_param(prepare_stmt,6, email)
    ibm_db.bind_param(prepare_stmt,7, mobileno)
    ibm_db.execute(prepare_stmt)
    return redirect(url_for('home'))
else:
    return render_template('register.html',error='Invalid details')
return render_template('login.html', name='home')
```

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

```
    if request.method=='POST':
        firstname = request.form['firstname']
        lastname = request.form['lastname']
        age1 = request.form['age1']
        bloodgroup = request.form['bloodgroup']
        gender1 = request.form['gender1']
        donatedbefore = request.form['donatedbefore']
        address1 = request.form['address1']
        anyhealthissues = request.form['anyhealthissues']
        sql1 = "insert into DONORDETAILS values(?,?,?,?,?,?,?,?)"
        prep_stmt1 = ibm_db.prepare(conn,sql1)
        ibm_db.bind_param(prepare_stmt1,1,firstname)
        ibm_db.bind_param(prepare_stmt1,2,lastname)
        ibm_db.bind_param(prepare_stmt1,3,age1)
        ibm_db.bind_param(prepare_stmt1,4,bloodgroup)
        ibm_db.bind_param(prepare_stmt1,5,gender1)
        ibm_db.bind_param(prepare_stmt1,6, donatedbefore)
        ibm_db.bind_param(prepare_stmt1,7, address1)
        ibm_db.bind_param(prepare_stmt1,8, anyhealthissues)
        ibm_db.execute(prepare_stmt1)
        return redirect(url_for('home'))
    elif request.method=='GET':
        return render_template('register')
```

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

```
    if request.method=='POST':
        username = request.form['username']
        gender1 = request.form['gender1']
        age1 = request.form['age1']
        bloodgroup = request.form['bloodgroup']
        email = request.form['email']
        mobileno = request.form['mobileno']
```

```
    if not username or gender1 or age1 or bloodgroup or email or mobileno:
        return render_template('request.html', error = 'Please fill all the fields')
```

```
    query = "select * from donordetails where username=? or bloodgroup=?"
    stmt = ibm_db.prepare(conn,query)
    ibm_db.bind_param(stmt,1,username)
    ibm_db.bind_param(stmt,2,bloodgroup)
    ibm_db.execute(stmt)
    isuser=ibm_db.fetch_assoc(stmt)
    if not isuser:
        return render_template('request.html', success='Request sent successfully.')
    else:
        return render_template('request.html',error='Invalid details')
    return render_template('home.html', name='home')
if (__name__)=='__main__':
    app.run(debug=True)
```

db.py

```
import ibm_db
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=19af6446-6171-4641-8aba-
9dcff8e1b6ff.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30699;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;
UID=dkz79110;PWD=VnQyUFTQ0JNLLGq",'','')
print(conn)
print("connection successful..")
```

GITHUB LINK:

<https://github.com/IBM-EPBL/IBM-Project-11953-1659361281>

PROJECT DEMO LINK:

https://drive.google.com/file/d/1JwcBhWgpzSKdRmzf65wvTIVOIVneMXy7/view?usp=share_link

