KONGUNADU COLLEGE OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

HX 8001-PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY AND ENTREPRENEURSHIP

PLASMA DONOR APPLICATION

NALAIYA THIRAN PROJECT REPORT 2022

Submitted by

BARATH S	**621319104009**
KOWSIK V	**621319104028**
SIVABALAN T	**621319104053**
SURENDHAR S	**621319104059**

Team ID: PNT2022TMID13250

NOVEMBER 2022

TABLE OF CONTENT

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

The most crucial element in a human body is blood. A person is unable to survive without it. Blood is essential, but its diverse components are also priceless resources for the human circulatory system. Years of experience in surviving and defending ourselves demonstrate that plasma's need is more critical than we initially believed.

Someone needs blood and blood components like RBC and plasma every minute of every day. However, the lack of a platform to match nearby blood donors with patients meant that the need went unmet. Our research aims to remove obstacles that patients must overcome in order to find a suitable donor because of the distance between patients, which makes finding a donor difficult. The plasma donor application will be used by the patients to send requests, and when a request is made, the programme will notify the registered donors.

1.2 PURPOSE

Finding a plasma donor has become increasingly difficult with medical advancements. An application that links the patient with nearby donors has been created with this goal in mind. The donor can visit the requester by visiting our application, which will then notify the donor of the requester's need for plasma. The major goal is to get in touch with donors who are in the same area as the requester since if the donor is far away, he or she might not be able to visit the requester.

CHAPTER 2

LITERATURE SURVEY

2.1 EXISTING SYSTEMS

- The requirement for the patient to look for donors nearby so they may verify the donors' identities.
- The patients may not find a donor because they find it difficult to look for donors in other places.
- If the patient hasn't identified any donors in a while, they could lose hope. The search for a plasma donor may cause anxiety in the patient's family.
- The patient may become exhausted when looking for a donor because there isn't an internet option; instead, they must visit each hospital to see if there are any donors available.

TITLE	**AUTHOR & YEAR**	**JOURNAL NAME**	**REMARKS**
Instant plasma donor recipient connector web application	Kalpana Devi Guntoju, Tejsvini Jalli, Sreejauppla & 2022	International Research Journal Of Modernization in engineering Technology and Science	In this proposed system, a donor who wants to donate plasma and can donate the plasma to a blood bank. The blood bank after checking the donor certificate can make a request to the donor.

TITLE	**AUTHOR & YEAR**	**JOURNAL NAME**	**REMARKS**
Mobile phone based system to promote eye donation	Waraporn Chumkasian, RitinFernand, KhinThanWin, HeidiLord & 2021	Science Direct	Eye Donor Form generates form using any smartphone .The donation form input is saved electronically and can be forwarded to your family members, healthcare.
Blood donation smartphone app	Afaf AliBatis, Ahmed Albarrak & 2021	Science Direct	The top rated features were the ability to request for blood donors, and the ability to locate the nearest blood center on the map.The preferred method of contact was found to be SMS
Android application for Medicine donation	Netra Shigwan, Pratiksha Chaudhari, Shweta Pawar, Anuja Gote & 2020	International Research Journal of Engineering and Technology (IRJET)	In this system, there are three entities,admin,Volunteer,andusers. Users can post Available medicine with the medicine name, medicine description, QRcode and medicine expired date. When Admin assign him medicines, he can open the user’s address location on the Google map and go to that location and bring the medicines

2.2 REFERENCES

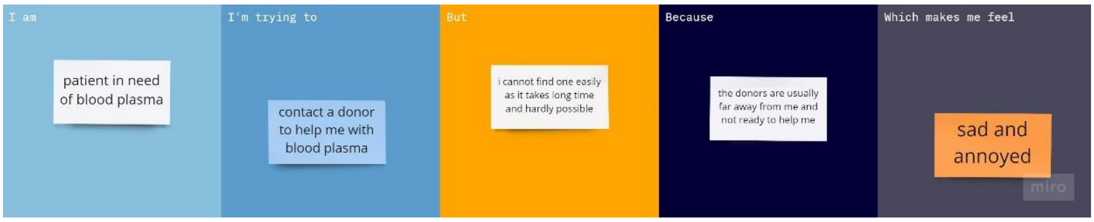
1. Kalpana Devi Guntoju, Tejaswini Jalli, Sreeja Uppala, Sanjay Mallisetti, International Research Journal of Modernization in Engineering Technology and Science, Instant plasma donor recipient connector web application, Volume:04/Issue:06/June-2022.
2. Kavita shirsat, Iqra shaikh,Pradnya Deshmukh,Mayuri lambhate4,Food donation application: food share, International research journal of engineering and technology (IRJET), volume: 08 issue: 05 | May 2021
3. Altahir Saad ,Ahmed Saad,Lars Rune Christensen, Blood Donation & Blood Banks through Android Mobile App and Web Application System , International Journal of Computer Science Trends and Technology (IJCST) ,Mar - Apr 2019.
4. A.Beurel, F.Terrade, J.-P.Lebaudy, B.Danic, Science direct, Determinants of plasma donation: A review of the literature, Volume 24, Issue 3, September 2017.
5. Vamsi Krishna Tatikonda and Hosam El-Ocla. BLOODR: blood donor and requester mobile application. mHealth on sep 18 2017.
6. Nikita M. Lunawat, Chetan D. Kshirsagar , Ashish A. Gawhande , Rohini M. Rathod,Blood and organ for patient using andriod application, International Journal of Research in Engineering and Technology,May-2016.
7. Blood App, American Red Cross, 2016. link: http://www.redcrossblood.org/ bloodapp.
8. Blood, organ and tissue donation -The need of blood donation in Canada, link:http://healthycanadians.gc.ca/diseases-conditions-maladiesaffections/ donation-contribution-eng.php.

2.3 PROBLEM STATEMENT DEFINITION

A problem statement is an explanation of the problem that need to be solved. A problem statement definition is a key that can assist guarantee that everyone working on a project is aware of the issue they need to solve and the significance of the project.

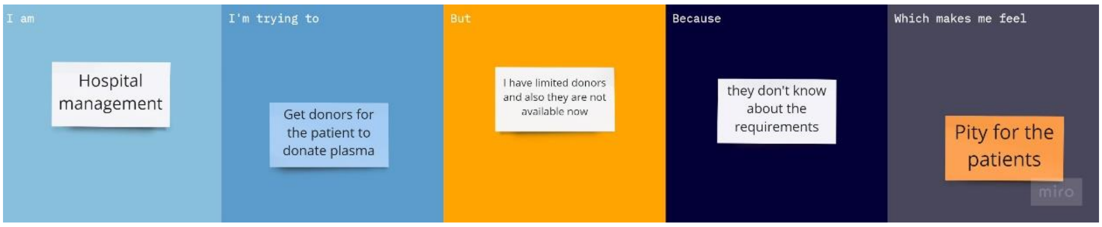
The problem statement states that the patient is having trouble locating a plasma donor in their area. As a result, the patient may have to wait a long time for one or may not be able to locate any donors nearby who are within driving distance. These are the issues that the patients are dealing with when trying to find a donor, even though they have located a donor in a remote place. These issues include the donor's potential inability to arrive in time for the donation.

Problem Statement 1



I am	I'm trying to	But	Because	Which makes me feel
patient in need of blood plasma	contact a donor to help me with blood plasma	i cannot find one easily as it takes long time and hardly possible	the donors are usually far away from me and not ready to help me	sad and annoyed

problem statement 2



I am	I'm trying to	But	Because	Which makes me feel
Hospital management	Get donors for the patient to donate plasma	I have limited donors and also they are not available now	they don't know about the requirements	Pity for the patients

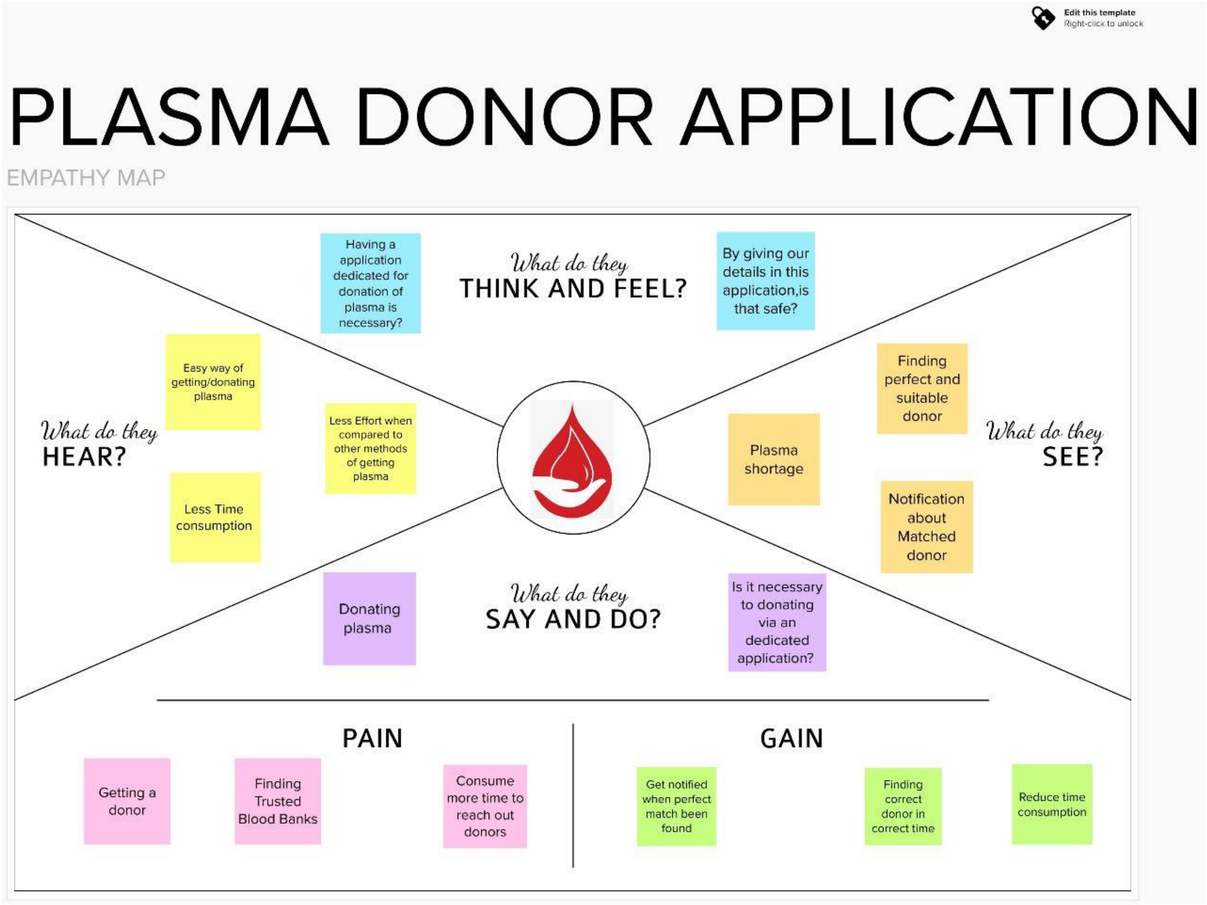
CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

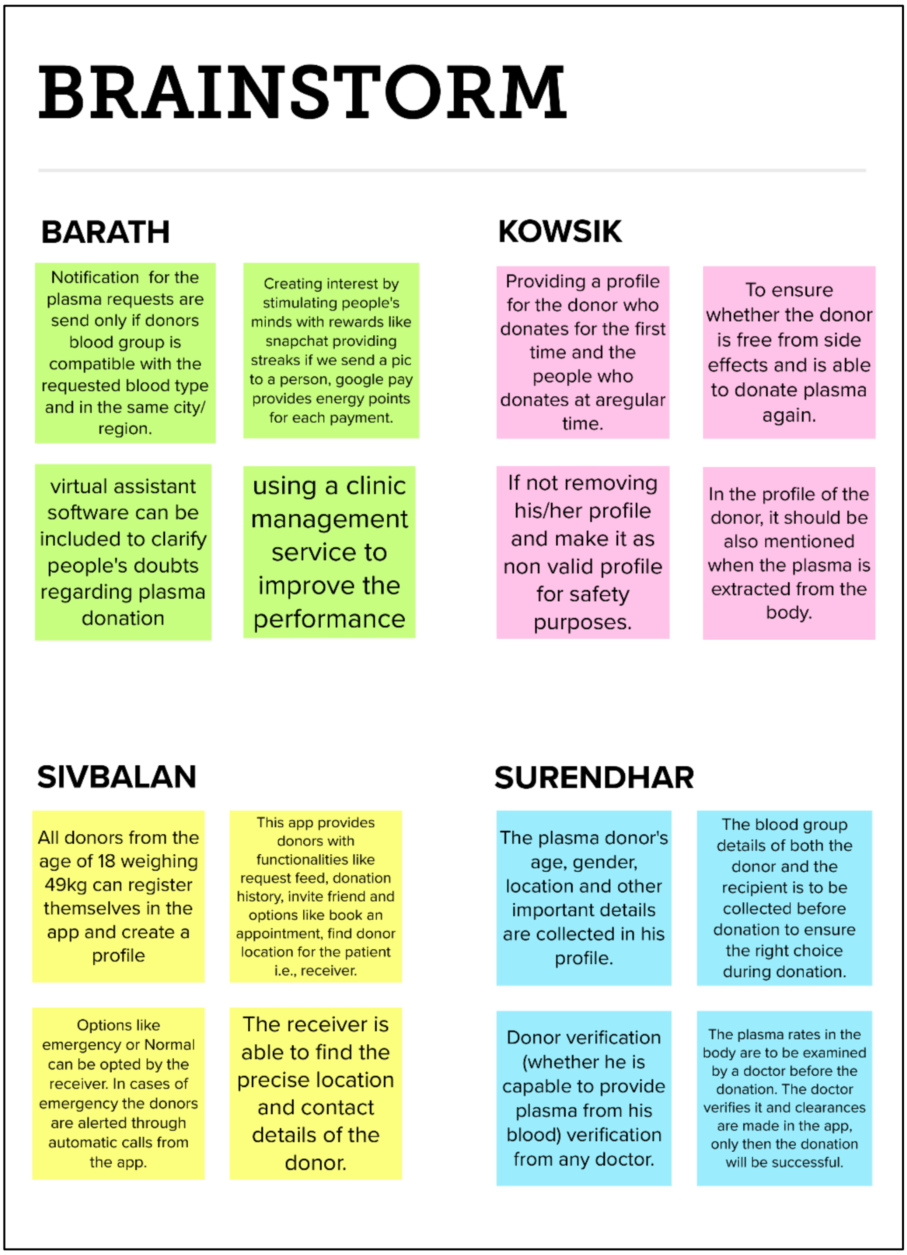
A collaborative tool known as empathy map that enables groups to learn more about their customers. Empathy maps can depict user groupings like consumer segments, much like user personas can. A variety of user opinions were gathered and included in the empathy map.

A variety of user opinions were gathered and included in the empathy map. Real-world users were surveyed to get the information that is most likely to meet user needs throughout implementation.


Edit this template
Right-click to unlock
PLASMA DONOR APPLICATION
EMPATHY MAP
What do they THINK AND FEEL?
Having a application dedicated for donation of plasma is necessary?
By giving our details in this application,is that safe?
What do they HEAR?
Easy way of getting/donating pllasma
Less Effort when compared to other methods of getting plasma
Less Time consumption
What do they SEE?
Finding perfect and suitable donor
Plasma shortage
Notification about Matched donor
What do they SAY AND DO?
Donating plasma
Is it necessary to donating via an dedicated application?
PAIN
Getting a donor
Finding Trusted Blood Banks
Consume more time to reach out donors
GAIN
Get notified when perfect match been found
Finding correct donor in correct time
Reduce time consumption

3.2 IDEATION & BRAINSTORMING

A group problem-solving technique that involves the sharing of original ideas. This method calls for a lengthy, rambunctious conversation in which each group member is urged to think aloud and offer as many ideas as they can based on their varied knowledge. Brainstorming blends informal problem-solving techniques with lateral thinking, a technique for coming up with fresh ideas and approaches to problems.



BRAINSTORM

BARATH

Notification for the plasma requests are send only if donors blood group is compatible with the requested blood type and in the same city/ region.

Creating interest by stimulating people's minds with rewards like snapchat providing streaks if we send a pic to a person, google pay provides energy points for each payment.

virtual assistant software can be included to clarify people's doubts regarding plasma donation

using a clinic management service to improve the performance

KOWSIK

Providing a profile for the donor who donates for the first time and the people who donates at aregular time.

To ensure whether the donor is free from side effects and is able to donate plasma again.

If not removing his/her profile and make it as non valid profile for safety purposes.

In the profile of the donor, it should be also mentioned when the plasma is extracted from the body.

SIVBALAN

All donors from the age of 18 weighing 49kg can register themselves in the app and create a profile

This app provides donors with functionalities like request feed, donation history, invite friend and options like book an appointment, find donor location for the patient i.e., receiver.

Options like emergency or Normal can be opted by the receiver. In cases of emergency the donors are alerted through automatic calls from the app.

The receiver is able to find the precise location and contact details of the donor.

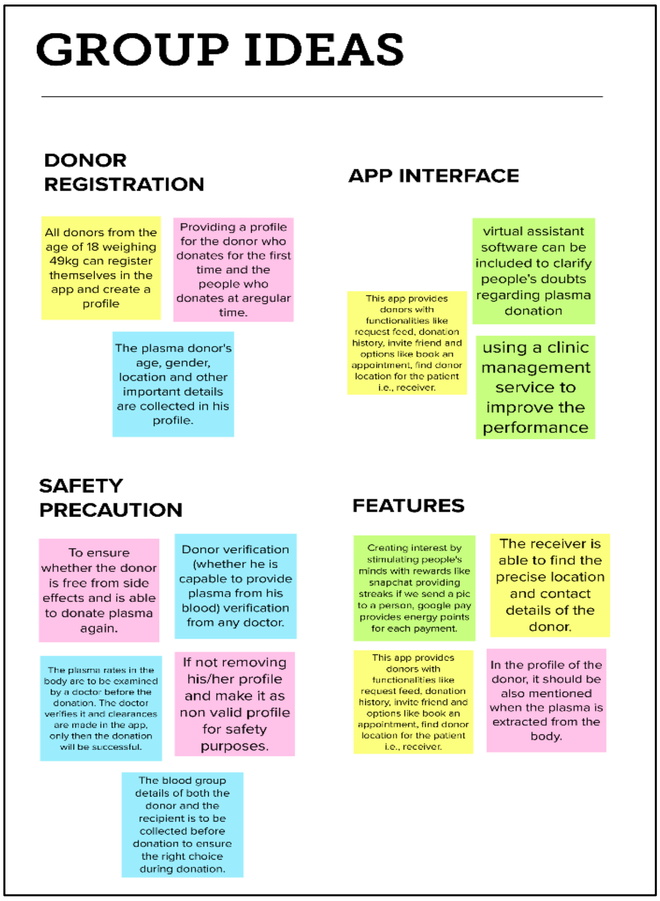
SURENDHAR

The plasma donor's age, gender, location and other important details are collected in his profile.

The blood group details of both the donor and the recipient is to be collected before donation to ensure the right choice during donation.

Donor verification (whether he is capable to provide plasma from his blood) verification from any doctor.

The plasma rates in the body are to be examined by a doctor before the donation. The doctor verifies it and clearances are made in the app, only then the donation will be successful.



GROUP IDEAS

DONOR REGISTRATION

All donors from the age of 18 weighing 49kg can register themselves in the app and create a profile

Providing a profile for the donor who donates for the first time and the people who donates at aregular time.

The plasma donor's age, gender, location and other important details are collected in his profile.

APP INTERFACE

virtual assistant software can be included to clarify people's doubts regarding plasma donation

This app provides donors with functionalities like request feed, donation history, invite friend and options like book an appointment, find donor location for the patient i.e., receiver.

using a clinic management service to improve the performance

SAFETY PRECAUTION

To ensure whether the donor is free from side effects and is able to donate plasma again.

Donor verification (whether he is capable to provide plasma from his blood) verification from any doctor.

The plasma rates in the body are to be examined by a doctor before the donation. The doctor verifies it and clearances are made in the app, only then the donation will be successful.

If not removing his/her profile and make it as non valid profile for safety purposes.

The blood group details of both the donor and the recipient is to be collected before donation to ensure the right choice during donation.

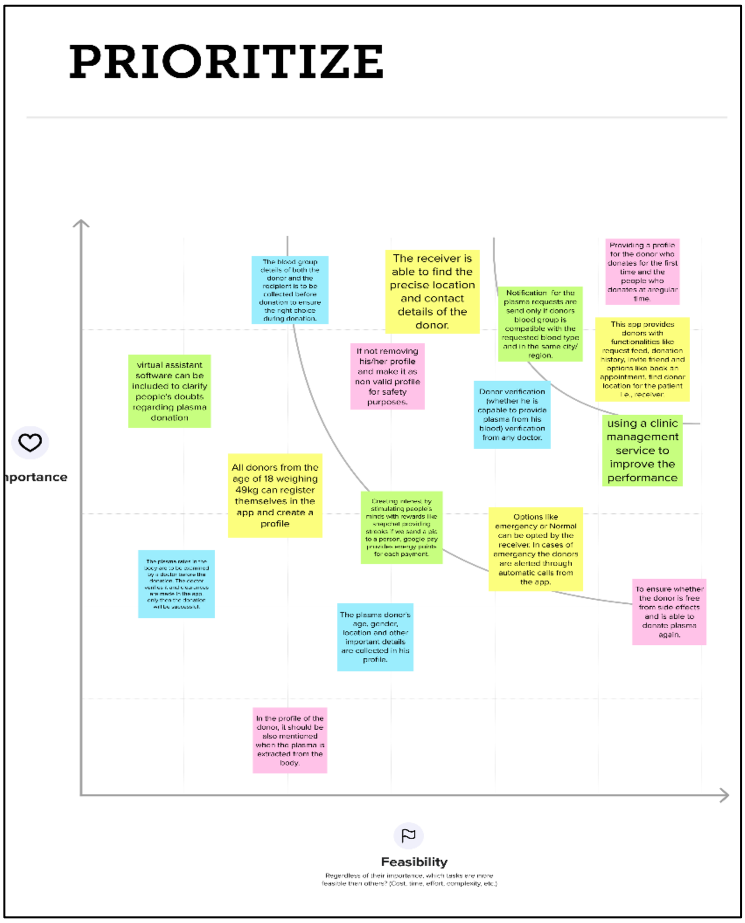
FEATURES

Creating interest by stimulating people's minds with rewards like snapchat providing streaks if we send a pic to a person, google pay provides energy points for each payment.

The receiver is able to find the precise location and contact details of the donor.

This app provides donors with functionalities like request feed, donation history, invite friend and options like book an appointment, find donor location for the patient i.e., receiver.

In the profile of the donor, it should be also mentioned when the plasma is extracted from the body.



PRIORITIZE

The blood group details of both the donor and the recipient is to be collected before donation to ensure the right choice during donation.

The receiver is able to find the precise location and contact details of the donor.

Providing a profile for the donor who donates for the first time and the people who donates at aregular time.

Notification for the plasma requests are send only if donors blood group is compatible with the requested blood type and in the same city/ region.

This app provides donors with functionalities like request feed, donation history, invite friend and options like book an appointment, find donor location for the patient i.e., receiver.

virtual assistant software can be included to clarify people's doubts regarding plasma donation

If not removing his/her profile and make it as non valid profile for safety purposes.

Donor verification (whether he is capable to provide plasma from his blood) verification from any doctor.

using a clinic management service to improve the performance

All donors from the age of 18 weighing 49kg can register themselves in the app and create a profile

Creating interest by stimulating people's minds with rewards like snapchat providing streaks if we send a pic to a person, google pay provides energy points for each payment.

Options like emergency or Normal can be opted by the receiver. In cases of emergancy the donors are alerted through automatic calls from the app.

The plasma rates in the body are to be examined by a doctor before the donation. The doctor verifies it and clearances are made in the app, only then the donation will be successful.

To ensure whether the donor is free from side effects and is able to donate plasma again.

The plasma donor's age, gender, location and other important details are collected in his profile.

In the profile of the donor, it should be also mentioned when the plasma is extracted from the body.

3.3 PROPOSED SOLUTION

S. No	Parameter	Description
1.	Problem Statement (Problem to be solved)	The need for plasma became urgent, and the number of donors has declined. It would be helpful to save the donor information and assist the less fortunate by informing the list of current donors.
2.	Idea / Solution description	To solve the issue, a programme will be created that will collect donor information, store it, and provide information upon request.
3.	Novelty / Uniqueness	The originality of the concept is its new, responsive interface and quick reaction when notifying donors and recipients. Users will communicate with a chatbot that is completely dedicated to that application environment.
4.	Social Impact / Customer Satisfaction	By connecting the donor and the beneficiary in an effortless manner, plasma donation will raise awareness among society's citizens. With its services, this programme will undoubtedly satisfy users.
5.	Business Model (Revenue Model)	The suggested model can be implemented in conjunction with non-profit organisations, and it will generate income for each and every request made by the user.
6.	Scalability of the Solution	The model can be deployed in a large scale based on the requirement and usage by the users.

3.4 PROBLEM SOLUTION FIT

Define CS, fit	**1.CUSTOMER SEGMENT CS** - The recipient who are in need of plasma. - The NGO's & hospital managements.	**6.CUSTOMER CONSTRAINTS CC** - There is no connection details between the customers. - Unavailability of plasma at the needed time.	**5.AVAILABLE SOLUTIONS AS** - Seeking help through social media. - Existing system involves, only the collection of donor data and will not notify the about the recipient.	Explore AS, differentiate

Focus on J&P, tap into BE, understand RC	**2.JOBS TO BE DONE/PROBLEMS J&P** - Establish a connection between the donor and the recipient. - Notify donors at the correct time. - Demand has increased.	**9.PROBLEM ROOT CAUSE RC** - 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.	**7.BEHAVIOUR BE** - The recipient will get the plasma at the right time. - The donors whose details, stored in database during registration will be notified.	Focus on J&P, tap into BE, understand RC

Identify strong TR & EM	**3.TRIGGERS TR** - We can advertise the web app through the NGO's and through the pharmaceutical companies. **4.EMOTIONS: BEFORE/AFTER EM** - Before : Anxiety, Stress, Scared - After : Relaxed, Happy	**10.YOUR SOLUTION SL** - Finding the respective donor and notify them through email for the requests.	**8.CHANNELS OF BEHAVIOUR CH** - The donor will register and they will be notified through the mail. - It will acts as a communication channel.	Identify strong TR & EM

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through website
FR-2	User Confirmation	Confirmation via Email
FR-3	User Login	Login through registered email id
FR-4	Send Request	If plasma required then donor get the notification
FR-5	Contact Donor	Contact donor directly if emergency

4.1 NON - FUNCTIONAL REQUIREMENT

NFR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The plasma Donor application is user friendlyand easy to access
NFR-2	**Security**	The users/donor details are stored in thecloud and it is secured with the user email id andpassword
NFR-3	**Reliability**	The system have the ability to work all thetimes without failure apart from network failure. The contact list of the donor are provided
NFR-4	**Performance**	The plasma donor application works well in every emergency situation. The easy interactive with the user and less interrupts
NFR-5	**Availability**	The plasma Application is an online webapplication and it monitor 24/7
NFR-6	**Scalability**	The application offers multiple users and it is designed to protect the users information and details.

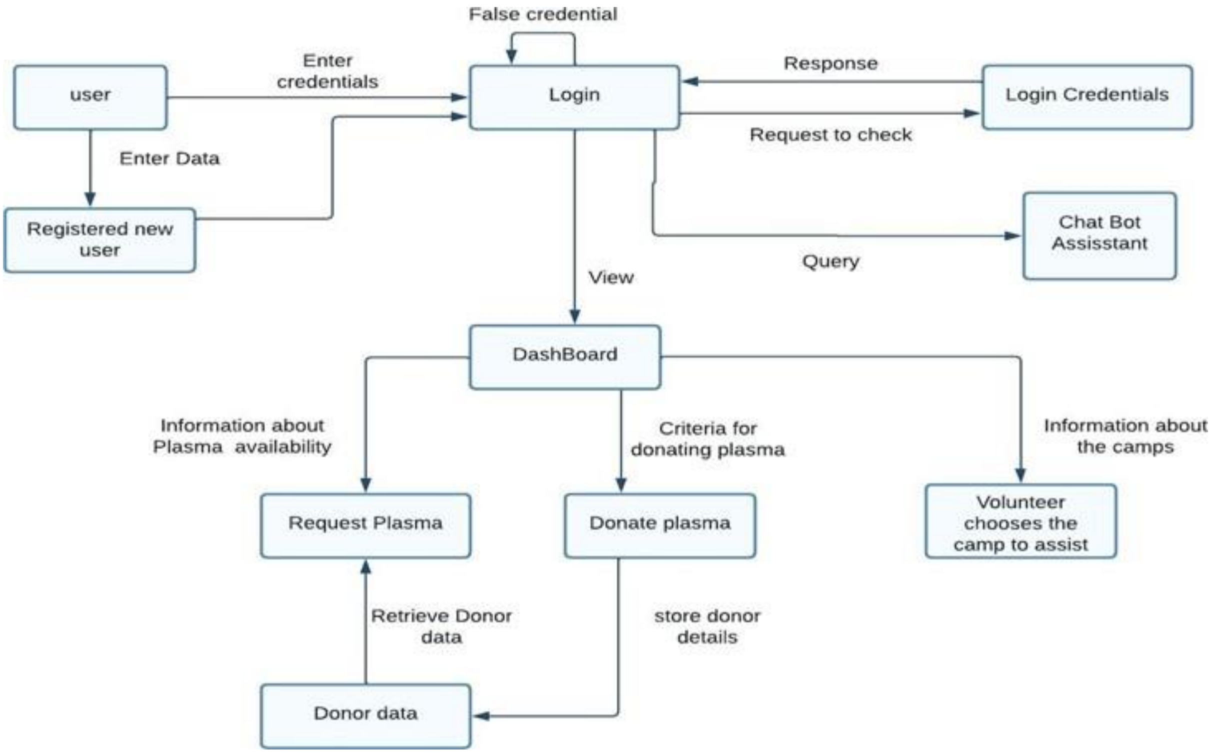
CHAPTER 5

PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

A data flow diagram (DFD) shows how the flow of information is done in an application it includes all the login details and other data that can be done by the user on the application how the data is being handled.

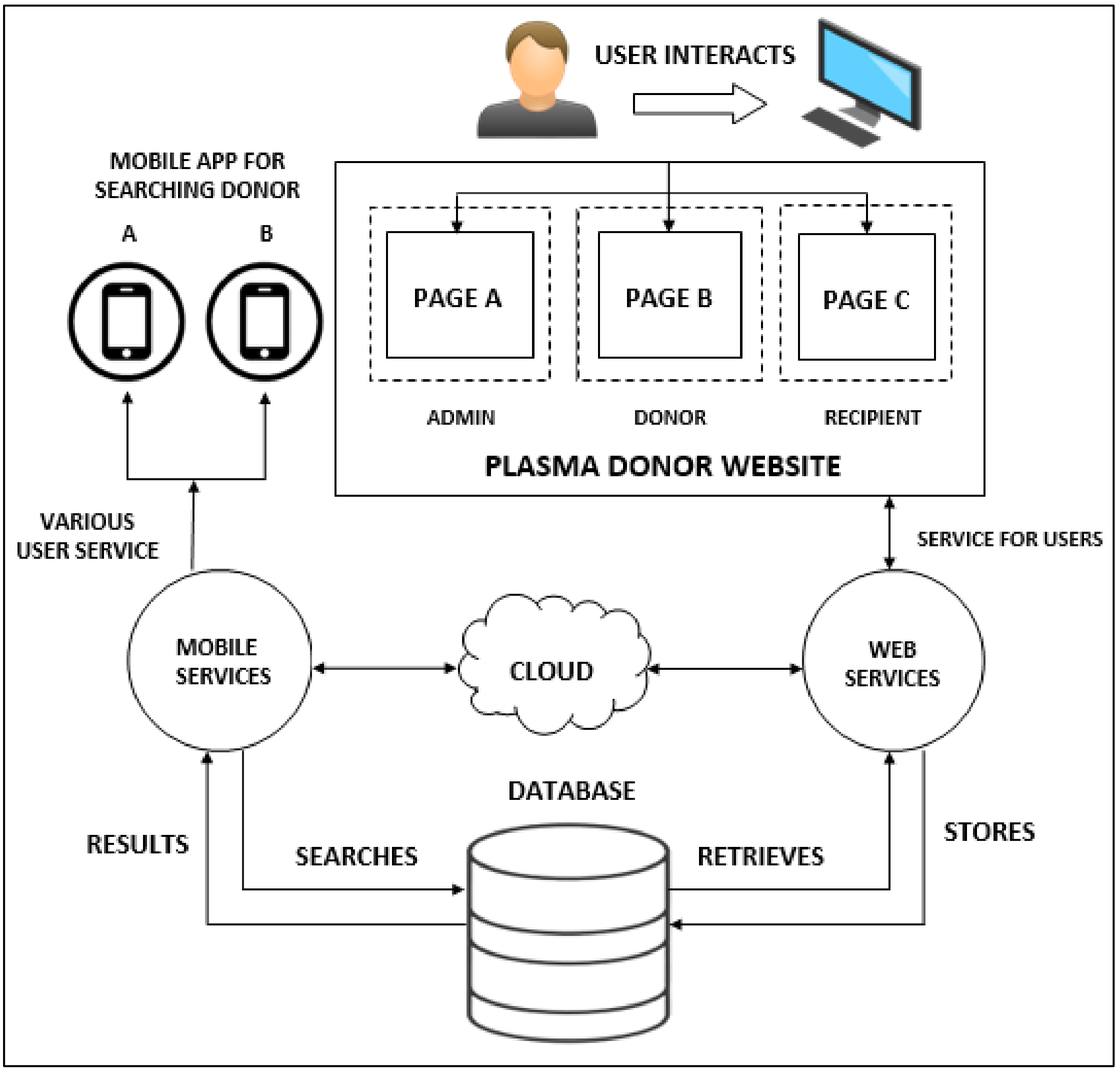
Simple, multi-level DFDs that gradually delve deeper into the data processing process are one type of data flow diagram, while more elaborate, hand-drawn process overviews can be found. A project's flow can be understood using the model that has been created. The IBM database and cloud services will be used to carry out the connections between them.



5.2 ARCHITECTURES

5.2.1 SOLUTION ARCHITECTURE

With the aim of ensuring that the created solution is compatible with the enterprise architecture in terms of information architecture, system portfolios, integration needs, and other factors, solution architecture is the process of creating solutions using established procedures, rules, and best practises. In order to address specific business objectives, requirements, or challenges, applications and information systems can be built and developed using a combination of roles, procedures, and documentation.

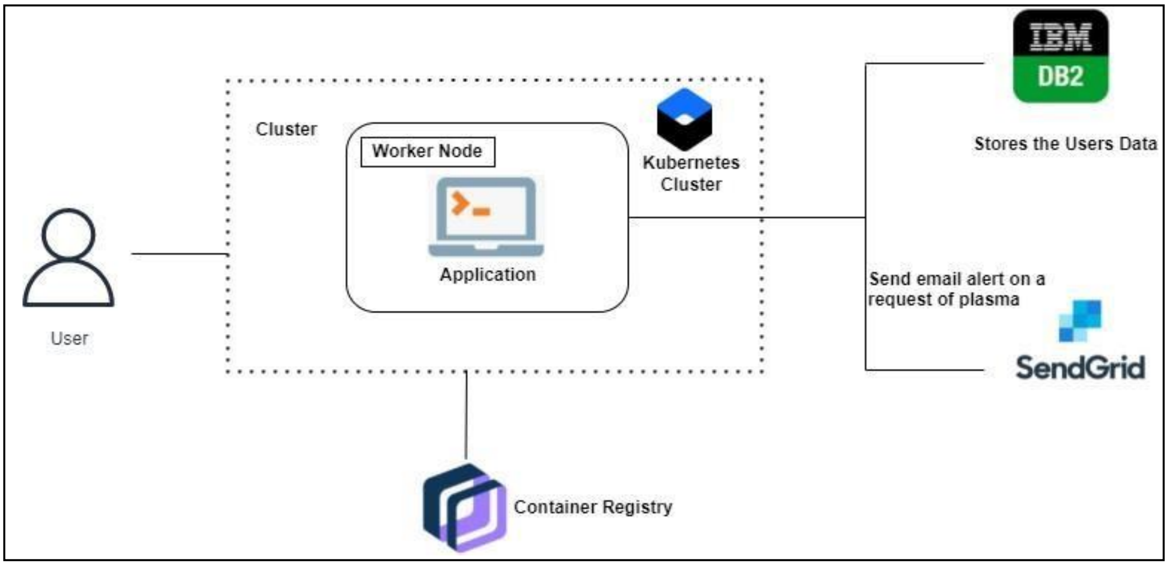


5.2.2 TECHNICAL ARCHITECTURE

For the purpose of designing computer systems, technical architecture (TA), a type of IT architecture, is employed. In order to satisfy system-relevant requirements, it entails creating a technological blueprint for the positioning, interacting, and interdependence of all pieces. Within the field of information technology, the word "architecture" has gained widespread usage over the previous ten years. This shouldn't come as a surprise given that the majority of businesses have to restructure their IT infrastructure to incorporate contemporary trends like cloud computing and software as a service (SaaS).

The container registry, which will be utilised to hold the entire process model, as well as Kubernetes and Dockers have been used as part of the technical architecture in this case. The SendGrid has been utilised to alert the user when plasma is required at the appropriate time.

The entire model must also be distributed by the cluster service into the public IP domain where it has previously been deployed in Kubernetes.



5.3 USER STORIES

User Type	**Functional Requirement(Epic)**	**User Story Number**	**User Story / Task**
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.
		USN-2	As a user, I will receive confirmation emailonce I have registered for the application
		USN-3	As a user, I can register for the applicationthrough Gmail
	Login	USN-4	As a user, I can log into the application by entering email & password
	Dashboard	USN-5	As a user,I can send the proper requests todonate and obtain plasma.
Customer (Web user)	Login	USN-6	As a user, I can register and log into the application by entering email & password toview the profile
	Dashboard	USN-7	As a user, I can send the proper requests todonate and obtain plasma.
Customer Care Executive	Application	USN-8	As a customer care executive, I can try to address user's concerns and questions
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.
		USN-10	As an administrator, I can involve working withthe technical side of websites.
User Type	Functional Requirement(Epic)	User Story Number	User Story / Task
Chatbot	Dashboard	USN-11	In addition the Customer careexecutive,chatbot can try to address user's concerns and questions

CHAPTER 6

PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority
Sprint - 1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	8	High
		USN-2	As a user, I will receive confirmation email once I have registered for the application	4	Low
	Login	USN-3	As a user, I can log into the application by entering email & password	8	High
Sprint - 2	Register for plasma donation	USN-4	As a donor, I have to register to intimate the users that I am interested in donating the plasma.	10	High
	Request for Plasma	USN-5	As a recipient, I have to request the plasma from the donors.	10	High
Sprint - 3	Homepage	USN-6	As a public, I will expect a good UI as front page.	4	Low
	Database	USN-7	The user data has to be saved and needs to maintained in the database.	10	High
	Chatbot	USN-8	As a user, I can get clarify my doubts with the help of the Watson assistant.	6	Moderate
Sprint - 4	Send Notification	USN-9	As a donor, myself have to be notified for donation of the plasma when it was in need.	12	High
	Deployment	USN-10	As a developer, I would like to deploy it in public domain	8	High

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	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Sprint velocity

Sprint 1 = 20/6 = 3.66

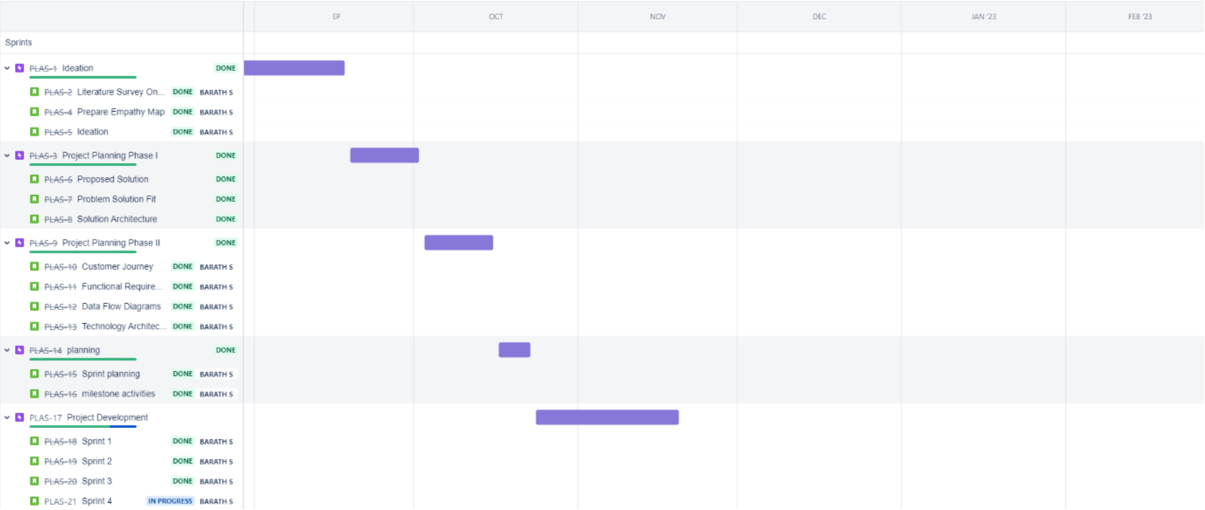
Sprint 2 = 20/6 = 3.66

Sprint 3 = 20/6 = 3.66

Sprint 4 = 20/6 = 3.66

6.3 REPORTS FROM JIRA

Jira is a piece of software for managing projects and keeping track of issues. Agile development teams frequently utilise the programme, which was created by the Australian software company Atlassian, to keep track of bugs, stories, epics, and other activities.



Sprints

- ~~PLAS-1~~ Ideation DONE
  - ~~PLAS-2~~ Literature Survey On... DONE BARATH S
  - ~~PLAS-4~~ Prepare Empathy Map DONE BARATH S
  - ~~PLAS-5~~ Ideation DONE BARATH S
- ~~PLAS-3~~ Project Planning Phase I DONE
  - ~~PLAS-6~~ Proposed Solution DONE
  - ~~PLAS-7~~ Problem Solution Fit DONE
  - ~~PLAS-8~~ Solution Architecture DONE
- ~~PLAS-9~~ Project Planning Phase II DONE
  - ~~PLAS-10~~ Customer Journey DONE BARATH S
  - ~~PLAS-11~~ Functional Require... DONE BARATH S
  - ~~PLAS-12~~ Data Flow Diagrams DONE BARATH S
  - ~~PLAS-13~~ Technology Architec... DONE BARATH S
- ~~PLAS-14~~ planning DONE
  - ~~PLAS-15~~ Sprint planning DONE BARATH S
  - ~~PLAS-16~~ milestone activities DONE BARATH S
- PLAS-17 Project Development
  - ~~PLAS-18~~ Sprint 1 DONE BARATH S
  - ~~PLAS-19~~ Sprint 2 DONE BARATH S
  - ~~PLAS-20~~ Sprint 3 DONE BARATH S
  - PLAS-21 Sprint 4 IN PROGRESS BARATH S

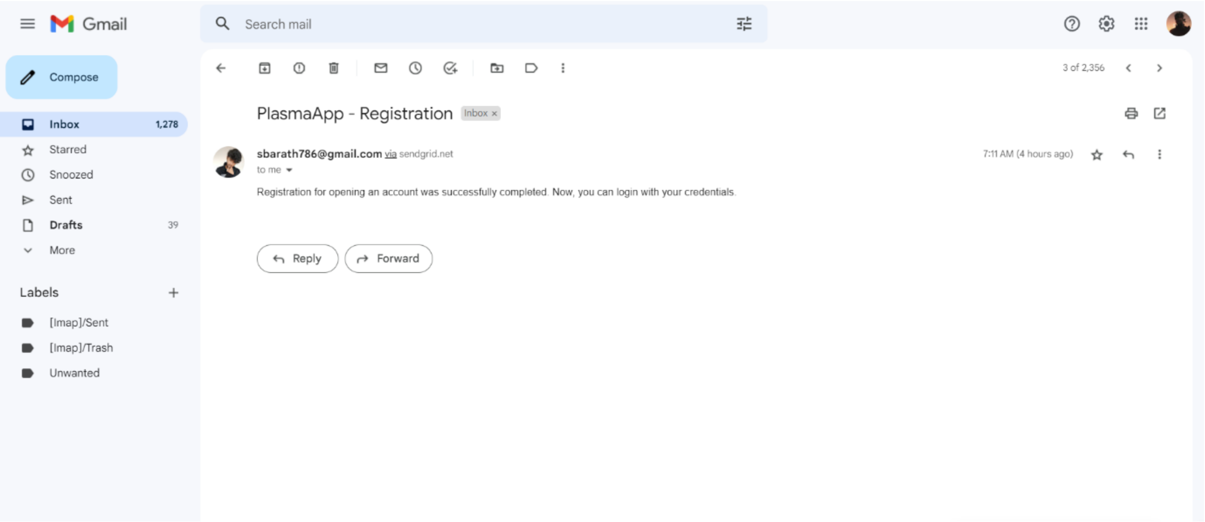
CHAPTER 7

CODING & SOLUTIONING

7.1 FEATURE 1

Users who register, request plasma, or donate plasma will receive notifications regarding such activities.

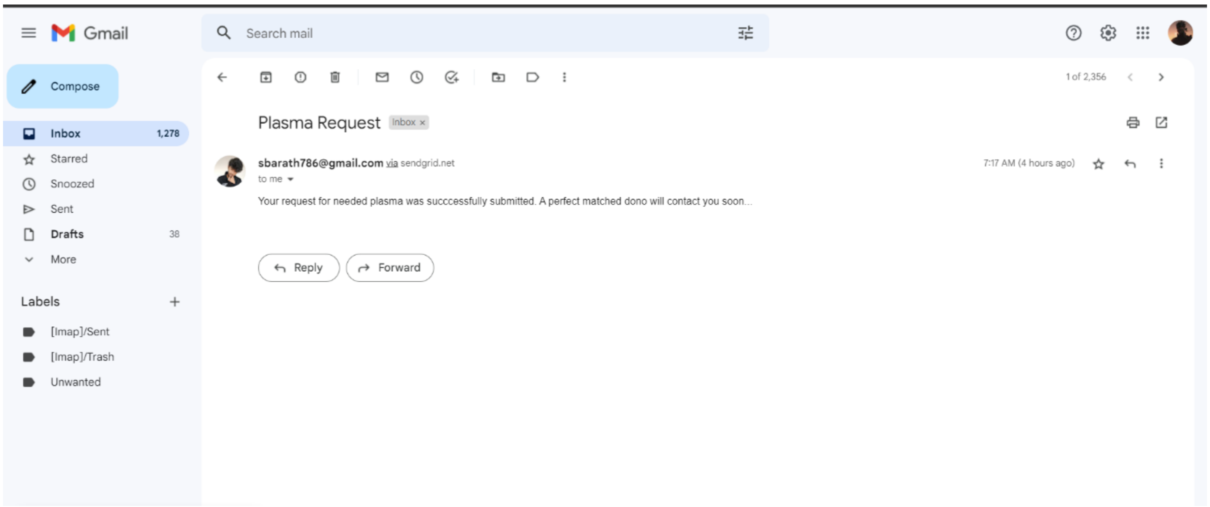
Sent mails



PlasmaApp - Registration

sbarath786@gmail.com via sendgrid.net
to me
7:11 AM (4 hours ago)

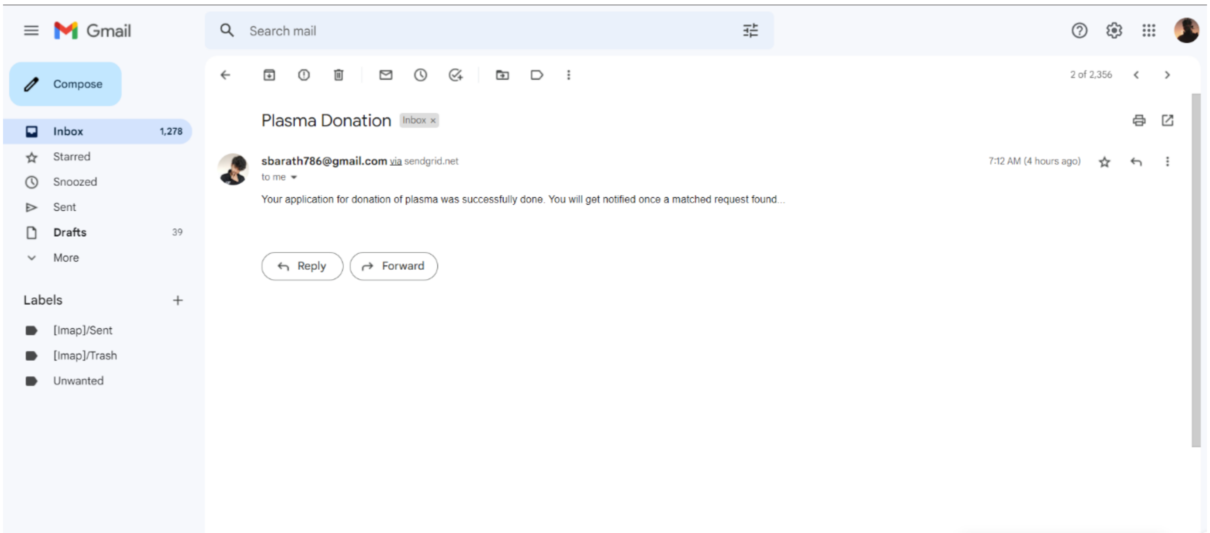
Registration for opening an account was successfully completed. Now, you can login with your credentials.



Plasma Request

sbarath786@gmail.com via sendgrid.net
to me
7:17 AM (4 hours ago)

Your request for needed plasma was succcessfully submitted. A perfect matched dono will contact you soon...



Plasma Donation Inbox ×

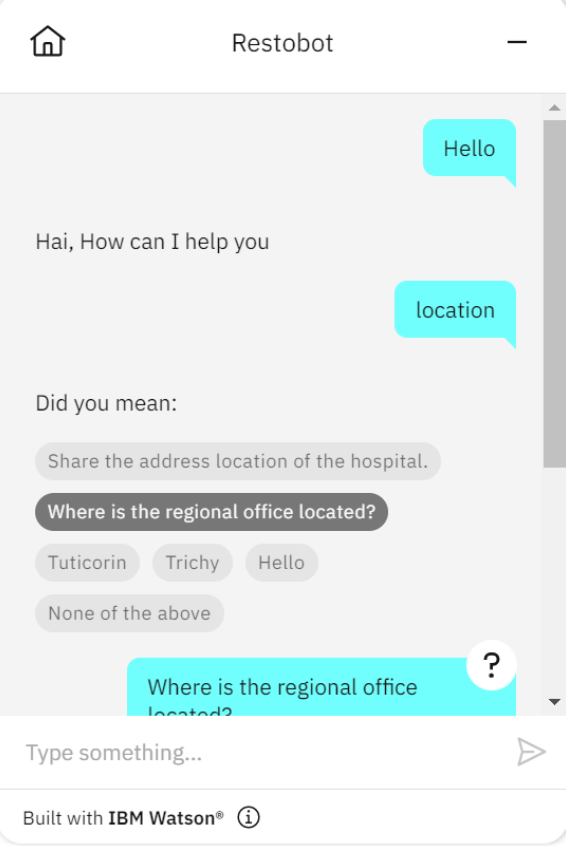
sbarath786@gmail.com via sendgrid.net
to me

7:12 AM (4 hours ago)

Your application for donation of plasma was successfully done. You will get notified once a matched request found...

7.2 FEATURE 2

For the benefit of the users, chatbots have been introduced into the model and trained on FAQs.



Restobot

Hello

Hai, How can I help you

location

Did you mean:

Share the address location of the hospital.

Where is the regional office located?

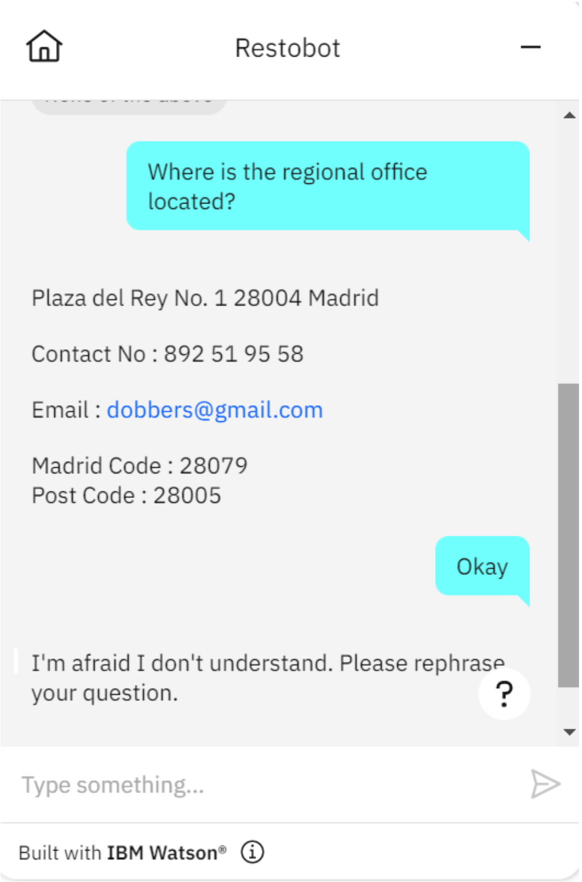
Tuticorin

Trichy

Hello

None of the above

Where is the regional office located?



Restobot

Where is the regional office located?

Plaza del Rey No. 1 28004 Madrid

Contact No : 892 51 95 58

Email : dobbers@gmail.com

Madrid Code : 28079
Post Code : 28005

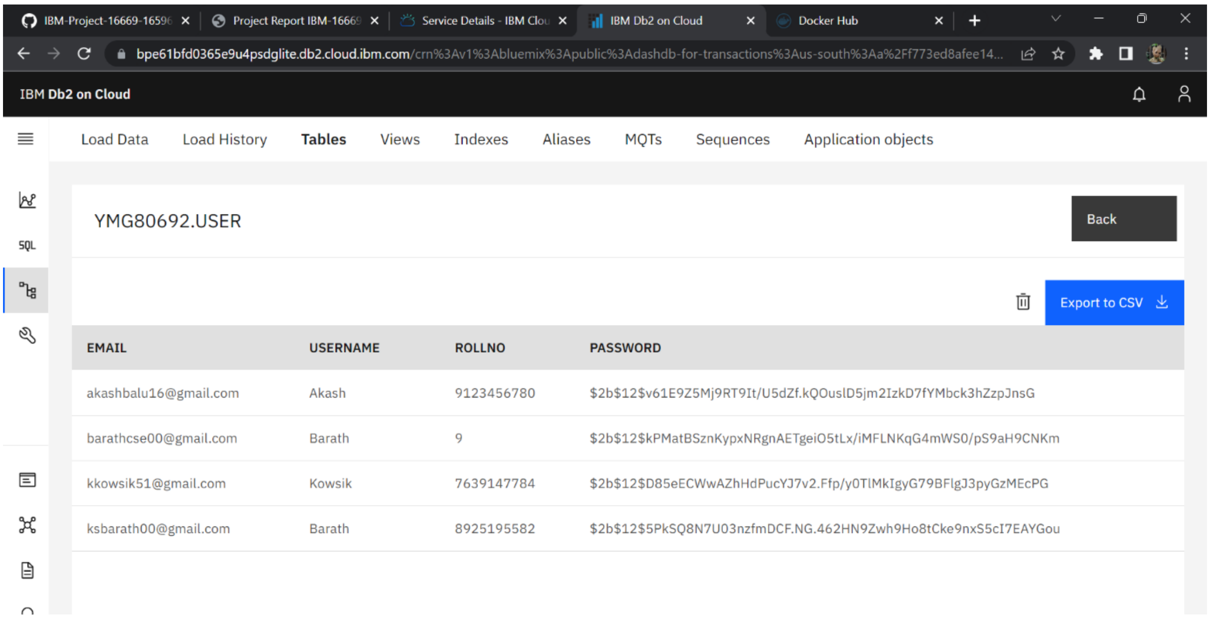
Okay

I'm afraid I don't understand. Please rephrase your question.

7.3 DATABASE SCHEMA

User details

create table USER(EMAIL VARCHAR(30),USERNAME CHAR(20), ROLLNO bigint, PASSWORD VARCHAR(100));



YMG80692.USER

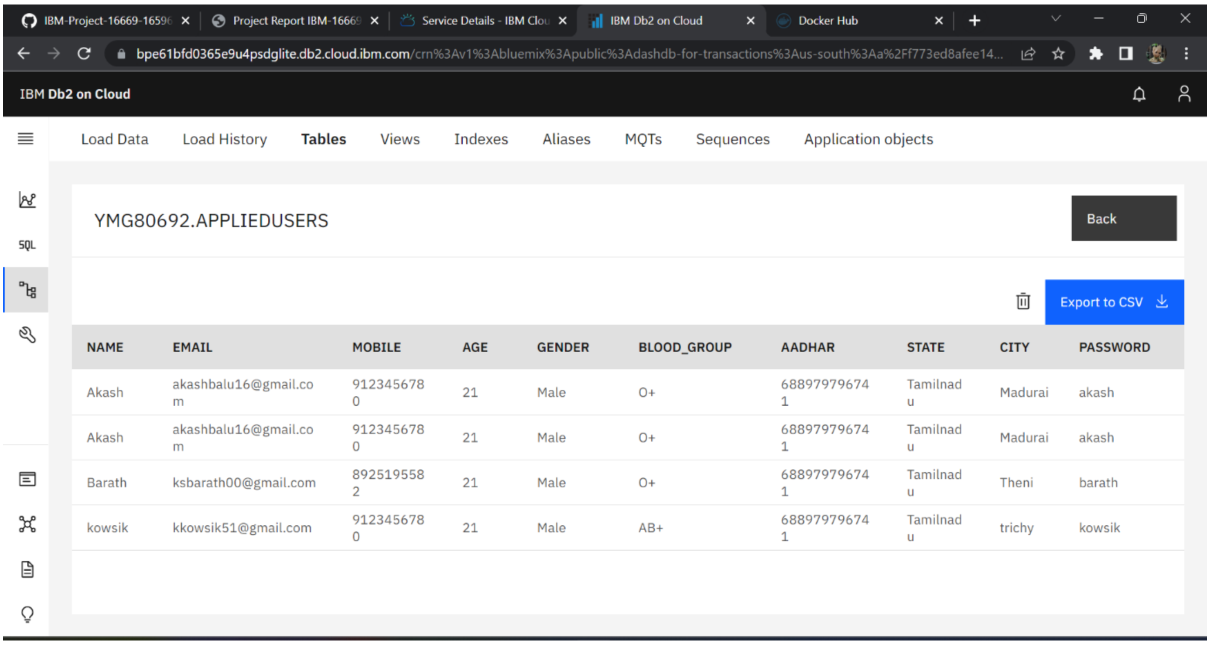
EMAIL	USERNAME	ROLLNO	PASSWORD
akashbalu16@gmail.com	Akash	9123456780	$2b$12$v61E9Z5Mj9RT9It/U5dZf.kQOuslD5jm2IzkD7fYMbck3hZzpJnsG
barathcse00@gmail.com	Barath	9	$2b$12$kPMatBSznKypxNRgnAETgeiO5tLx/iMFLNKqG4mWS0/pS9aH9CNKm
kkowsik51@gmail.com	Kowsik	7639147784	$2b$12$D85eECWwAZhHdPucYJ7v2.Ffp/y0TlMkIgyG79BFlgJ3pyGzMEcPG
ksbarath00@gmail.com	Barath	8925195582	$2b$12$5PkSQ8N7U03nzfmDCF.NG.462HN9Zwh9Ho8tCke9nxS5cI7EAYGou

Applied users

CREATE TABLE appliedusers(NAME CHAR(30),EMAIL CHAR(50), MOBILE bigint,AGE integer, GENDER CHAR(20), BLOOD_GROUP CHAR(10), AADHAR bigint, STATE CHAR(25), CITY CHAR(25), PASSWORD VARCHAR(50));

Value insertion

INSERT INTO appliedusers(NAME, EMAIL, MOBILE, AGE, GENDER, BLOOD_GROUP, AADHAR, STATE, CITY, PASSWORD)VALUES('Barath','barathcse00@gmail.com',8925195582,21,'Male','O+',688979796741,'Tamilnadu','Theni','barath');



YMG80692.APPLIEDUSERS

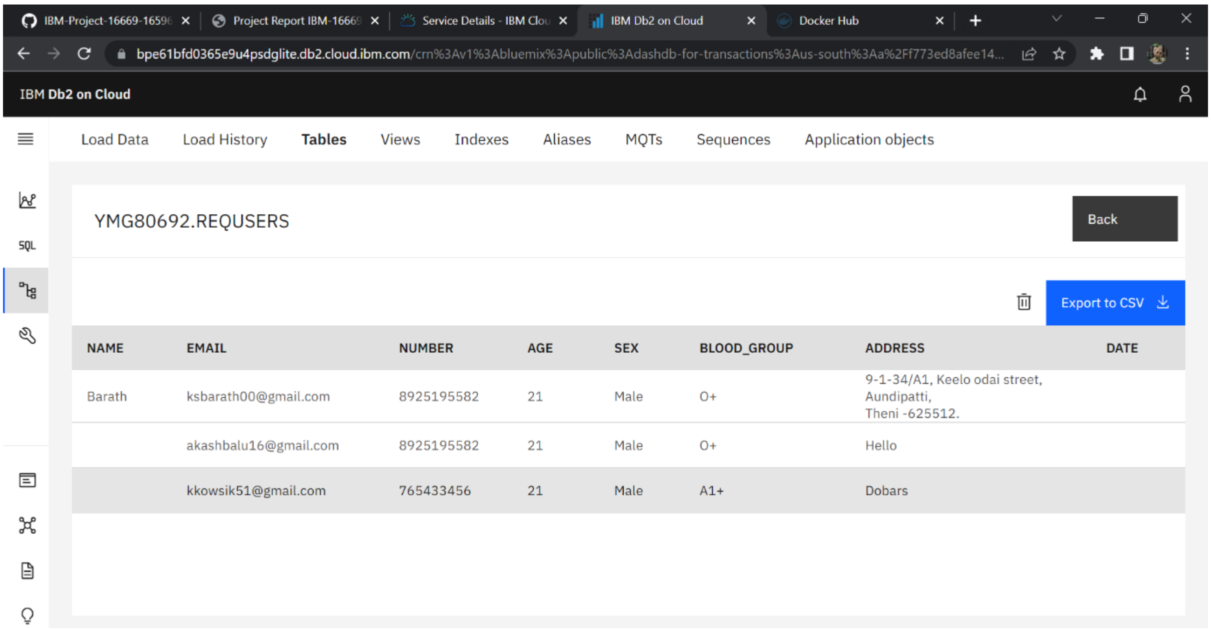
NAME	EMAIL	MOBILE	AGE	GENDER	BLOOD_GROUP	AADHAR	STATE	CITY	PASSWORD
Akash	akashbalu16@gmail.com	9123456780	21	Male	O+	688979796741	Tamilnadu	Madurai	akash
Akash	akashbalu16@gmail.com	9123456780	21	Male	O+	688979796741	Tamilnadu	Madurai	akash
Barath	ksbarath00@gmail.com	8925195582	21	Male	O+	688979796741	Tamilnadu	Theni	barath
kowsik	kkowsik51@gmail.com	9123456780	21	Male	AB+	688979796741	Tamilnadu	trichy	kowsik

Requested Users

create table requsers (name char(30),email char(50),number bigint,age int,sex char(25),blood_group char(10),address char(100),date char(20));

Value insertion

insert into requsers (name,email,number,age,sex,blood_group,address, date) values ('Barath','barathcse00@gmail.com',8925195582,21,'Male','O+','9-1-34/A1, keela odai street, Aundipatti, Theni','2022-11-3');



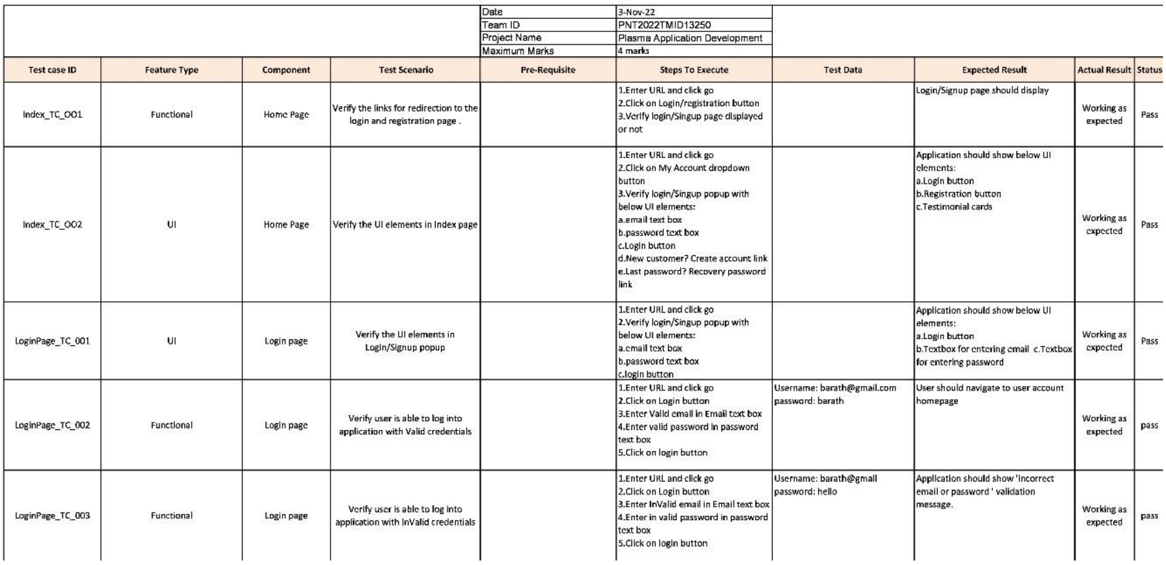
YMG80692.REQUSERS

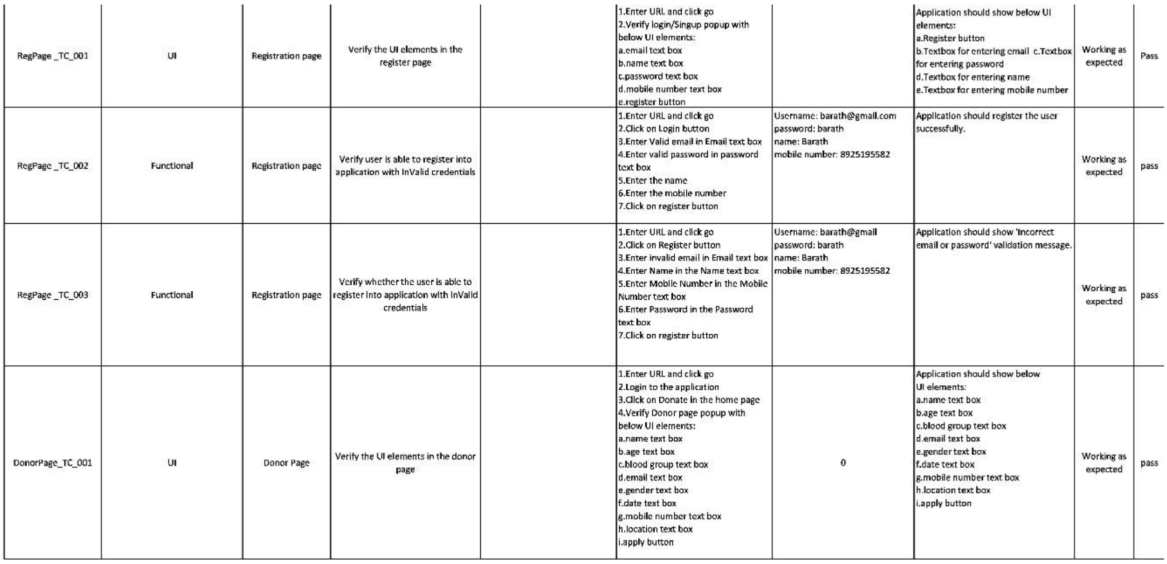
NAME	EMAIL	NUMBER	AGE	SEX	BLOOD_GROUP	ADDRESS	DATE
Barath	ksbarath00@gmail.com	8925195582	21	Male	O+	9-1-34/A1, Keelo odai street, Aundipatti, Theni -625512.	
	akashbalu16@gmail.com	8925195582	21	Male	O+	Hello	
	kkowsik51@gmail.com	765433456	21	Male	A1+	Dobars	

CHAPTER 8

TESTING

8.1 TEST CASES



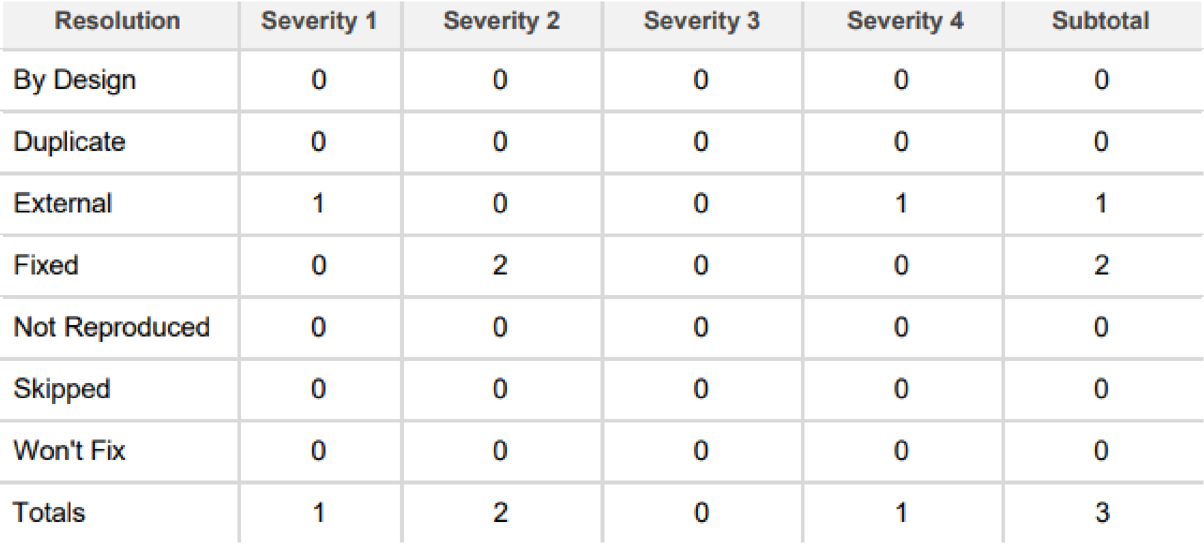


Date	3-Nov-22
Team ID	PNT2022TMID13250
Project Name	Plasma Application Development
Maximum Marks	4 marks

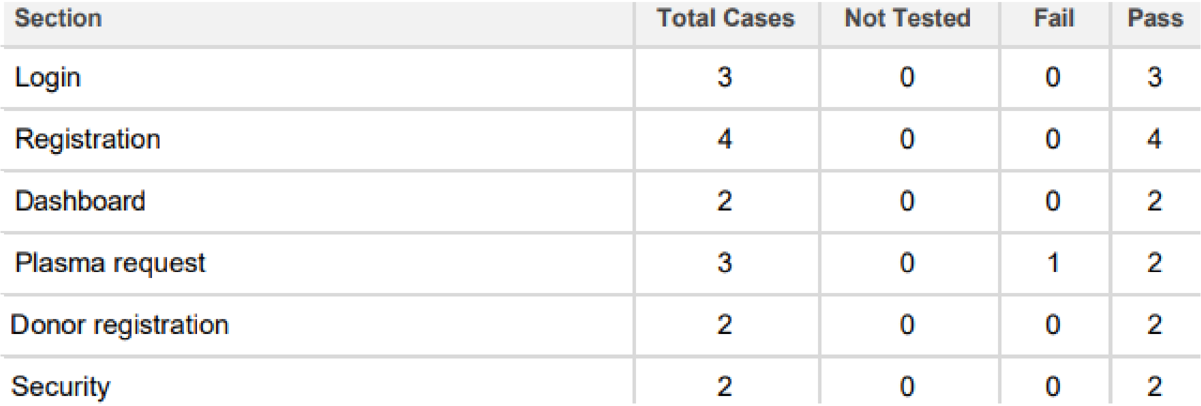
Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status
Index_TC_OO1	Functional	Home Page	Verify the links for redirection to the login and registration page .		1.Enter URL and click go 2.Click on Login/registration button 3.Verify login/Singup page displayed or not		Login/Signup page should display	Working as expected	Pass
Index_TC_OO2	UI	Home Page	Verify the UI elements in Index page		1.Enter URL and click go 2.Click on My Account dropdown button 3.Verify login/Singup popup with below UI elements: a.email text box b.password text box c.Login button d.New customer? Create account link e.Last password? Recovery password link		Application should show below UI elements: a.Login button b.Registration button c.Testimonial cards	Working as expected	Pass
LoginPage_TC_001	UI	Login page	Verify the UI elements in Login/Signup popup		1.Enter URL and click go 2.Verify login/Singup popup with below UI elements: a.email text box b.password text box c.login button		Application should show below UI elements: a.Login button b.Textbox for entering email c.Textbox for entering password	Working as expected	Pass
LoginPage_TC_002	Functional	Login page	Verify user is able to log into application with Valid credentials		1.Enter URL and click go 2.Click on Login button 3.Enter Valid email in Email text box 4.Enter valid password in password text box 5.Click on login button	Username: barath@gmail.com password: barath	User should navigate to user account homepage	Working as expected	pass
LoginPage_TC_003	Functional	Login page	Verify user is able to log into application with InValid credentials		1.Enter URL and click go 2.Click on Login button 3.Enter InValid email in Email text box 4.Enter in valid password in password text box 5.Click on login button	Username: barath@gmail password: hello	Application should show 'Incorrect email or password ' validation message.	Working as expected	pass
RegPage _TC_001	UI	Registration page	Verify the UI elements in the register page		1.Enter URL and click go 2.Verify login/Singup popup with below UI elements: a.email text box b.name text box c.password text box d.mobile number text box e.register button		Application should show below UI elements: a.Register button b.Textbox for entering email c.Textbox for entering password d.Textbox for entering name e.Textbox for entering mobile number	Working as expected	Pass
RegPage _TC_002	Functional	Registration page	Verify user is able to register into application with InValid credentials		1.Enter URL and click go 2.Click on Login button 3.Enter Valid email in Email text box 4.Enter valid password in password text box 5.Enter the name 6.Enter the mobile number 7.Click on register button	Username: barath@gmail.com password: barath name: Barath mobile number: 8925195582	Application should register the user successfully.	Working as expected	pass
RegPage _TC_003	Functional	Registration page	Verify whether the user is able to register into application with InValid credentials		1.Enter URL and click go 2.Click on Register button 3.Enter invalid email in Email text box 4.Enter Name in the Name text box 5.Enter Mobile Number in the Mobile Number text box 6.Enter Password in the Password text box 7.Click on register button	Username: barath@gmail password: barath name: Barath mobile number: 8925195582	Application should show 'Incorrect email or password' validation message.	Working as expected	pass
DonorPage_TC_001	UI	Donor Page	Verify the UI elements in the donor page		1.Enter URL and click go 2.Login to the application 3.Click on Donate in the home page 4.Verify Donor page popup with below UI elements: a.name text box b.age text box c.blood group text box d.email text box e.gender text box f.date text box g.mobile number text box h.location text box i.apply button	0	Application should show below UI elements: a.name text box b.age text box c.blood group text box d.email text box e.gender text box f.date text box g.mobile number text box h.location text box i.apply button	Working as expected	pass

DonorPage_TC_002	Functional	Donor Page	Verify whether the user able to register for donation		1.Enter URL and click go 2.Login to the application 3.Click on Donate in the home page 4.Enter name in name text box 5.Enter age in age text box 6.Enter blood group in blood group text box 7.Enter email in email text box 8.Enter gender in gender text box 9.Enter date in date text box 10.Enter mobile number in mobile number text box 11.Enter location inlocation text box 12.Click on apply button	Username: barath@gmail.com name: Barath age: 21 blood group: O+ve gender: Male Date: 2022-11-03 mobile number: 8925195582 location: Theni	The user will get Successfully applied and also get a mail from the application.	Working as expected	Pass
DonorPage_TC_003	Functional	Donor Page	Verify whether the user able to register for donation using invalid credentials		1.Enter URL and click go 2.Login to the application 3.Click on Donate in the home page 4.Enter name in name text box 5.Enter age in age text box 6.Enter blood group in blood group text box 7.Enter invalid email in email text box 8.Enter gender in gender text box 9.Enter date in date text box 10.Enter mobile number in mobile number text box 11.Enter location inlocation text box 12.Click on apply button	Username: barath@gmail name: Barath age: 21 blood group: O+ve gender: Male Date: 2022-11-19 mobile number: 8925195582 location: Theni	Application should show 'Incorrect email or password ' validation message.	Working as expected	pass
RequestPage_TC_001	UI	Request page	Verify the UI elements in the request page		1.Enter URL and click go 2.Login to the application 3.Click on Request in the home page 4.Verify Request page popup with below UI elements: a.name text box b.age text box c.blood group text box d.email text box e.gender text box f.date text box g.mobile number text box h.location text box i.password text box j.apply button		Application should show below UI elements: a.name text box b.age text box c.blood group text box d.email text box e.gender text box f.date text box g.mobile number text box h.location text box i.password text box j.apply button	Working as expected	Pass
RequestPage_TC_002	Functional	Request page	Verify whether the user able to request for donation		1.Enter URL and click go 2.Login to the application 3.Click on Donate in the home page 4.Enter name in name text box 5.Enter age in age text box 6.Enter blood group in blood group text box 7.Enter email in email text box 8.Enter gender in gender text box 9.Enter date in date text box 10.Enter mobile number in mobile number text box 11.Enter location in location text box 12. Enter password in password text box 13.Click on request button	Username: barath@gmail.com name: Barath age: 21 blood group: O+ve gender: Male Date: 2022-11-03 mobile number: 8925195582 location: Theni password: barath	The user will get Successfully applied and also get a mail from the application.	Working as expected	Pass
RequestPage_TC_003	Functional	Request page	Verify whether the user able to request for donation using invalid credentials		1.Enter URL and click go 2.Login to the application 3.Click on Donate in the home page 4.Enter name in name text box 5.Enter age in age text box 6.Enter blood group in blood group text box 7.Enter invalid email in email text box 8.Enter gender in gender text box 9.Enter date in date text box 10.Enter mobile number in mobile number text box 11.Enter location in location text box 12. Enter invalid password in password text box 13.Click on request button	Username: barath@gmail name: Barath age: 21 blood group: O+ve gender: Male Date: 2022-11-19 mobile number: 8925195582 location: Theni password: hello	Application should show 'Incorrect email or password ' validation message.	Working as expected	Pass

8.2 USER ACCEPTANCE TESTING



Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	0	0	0	0	0
Duplicate	0	0	0	0	0
External	1	0	0	1	1
Fixed	0	2	0	0	2
Not Reproduced	0	0	0	0	0
Skipped	0	0	0	0	0
Won't Fix	0	0	0	0	0
Totals	1	2	0	1	3



Section	Total Cases	Not Tested	Fail	Pass
Login	3	0	0	3
Registration	4	0	0	4
Dashboard	2	0	0	2
Plasma request	3	0	1	2
Donor registration	2	0	0	2
Security	2	0	0	2

CHAPTER 9

RESULTS

9.1 PERFORMANCE METRICS

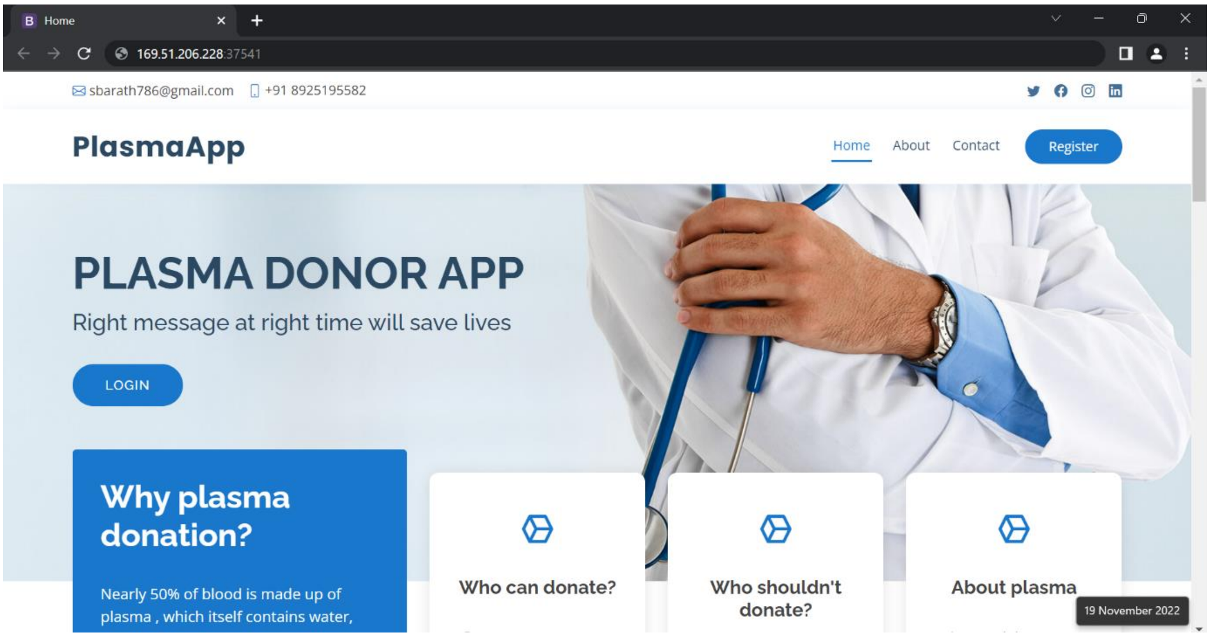
			NFT - Risk Assessment						
S.No	Project Name	Scope/feature	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	Load/Voluem Changes	Risk Score	Justification
1	Plasma Donor	Existing	Low	No Changes	Moderate		>5 to 10%	GREEN	As we have seen the changes
2	Plasma Donor	New	Low	No Changes	Moderate		>5 to 10%	ORANGE	As we have seen the changes
3	Plasma Donor	New	Low	No Changes	Moderate		>5 to 10%	GREEN	As we have seen the changes
4	Plasma Donor	Existing	Low	No Changes	High		>5 to 10%	GREEN	As we have seen the changes
5	Plasma Donor	New	Low	No Changes	Moderate		As we have seen the changes	As we have seen the changes	As we have seen the changes

NFT - Detailed Test Plan				
S.No	Project Overview	NFT Test approach	Assumptions/Dependencies/Risks	Approvals/SignOff
1	Plasma Donor Login Page	LOAD	Page slow down.It may not be accessible	Barath S , Kowsik V
2	Donor page	STRESS	Page slow down.It may not be accessible	Barath S , Sivabalan T
3	Request Page	STRESS	Page slow down.It may not be accessible	Barath S , Surendhar S

S.No	Project Overview	NFT Test approach	NFR - Met	Test Outcome	GO/NO-GO decision	Identified Defects (Detected/Closed/Open)	Approvals/SignOff
1	Plasma Donor Login Page	LOAD	Not met because of login the user	PASS	NO-GO	CLOSED	Barath S , Kowsik V
2	Donor Page	STRESS	Not met because donor able to register	PASS	NO-GO	CLOSED	Barath S , Sivabalan T
3	Request Page	STRESS	Not met because requester is able to rewquest	PASS	NO-GO	CLOSED	Barath S , Surendhar S

OUPTUTS

Homepage

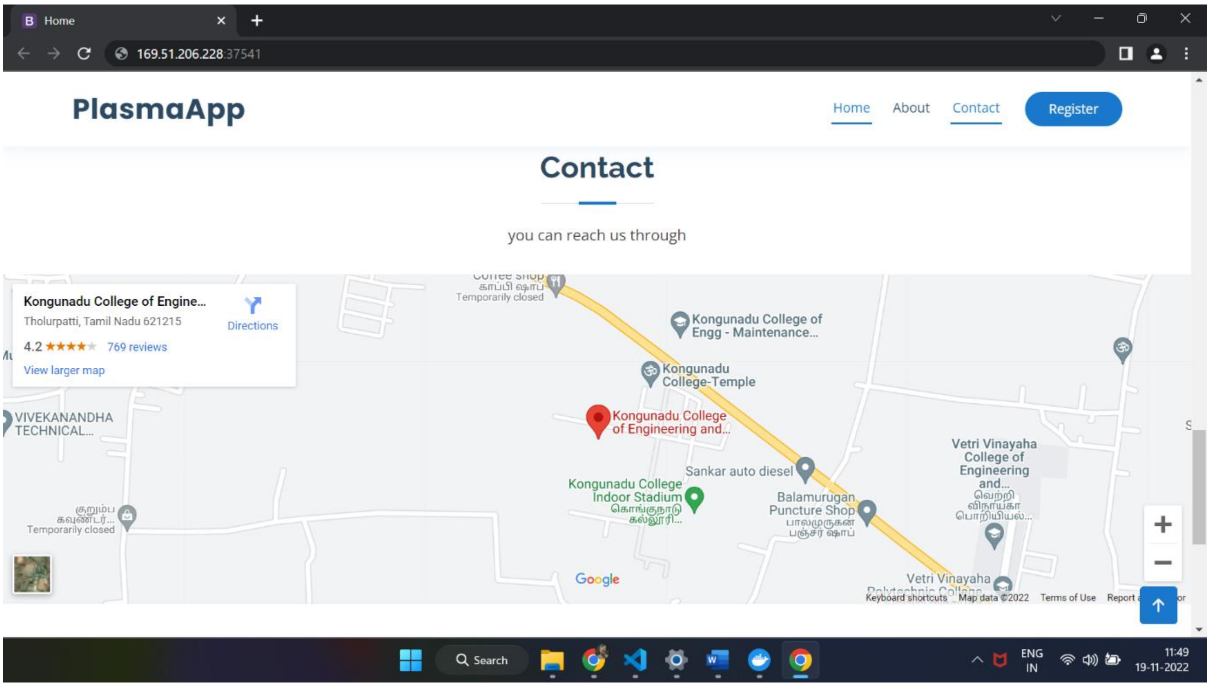


PLASMA DONOR APP

Right message at right time will save lives

Why plasma donation?

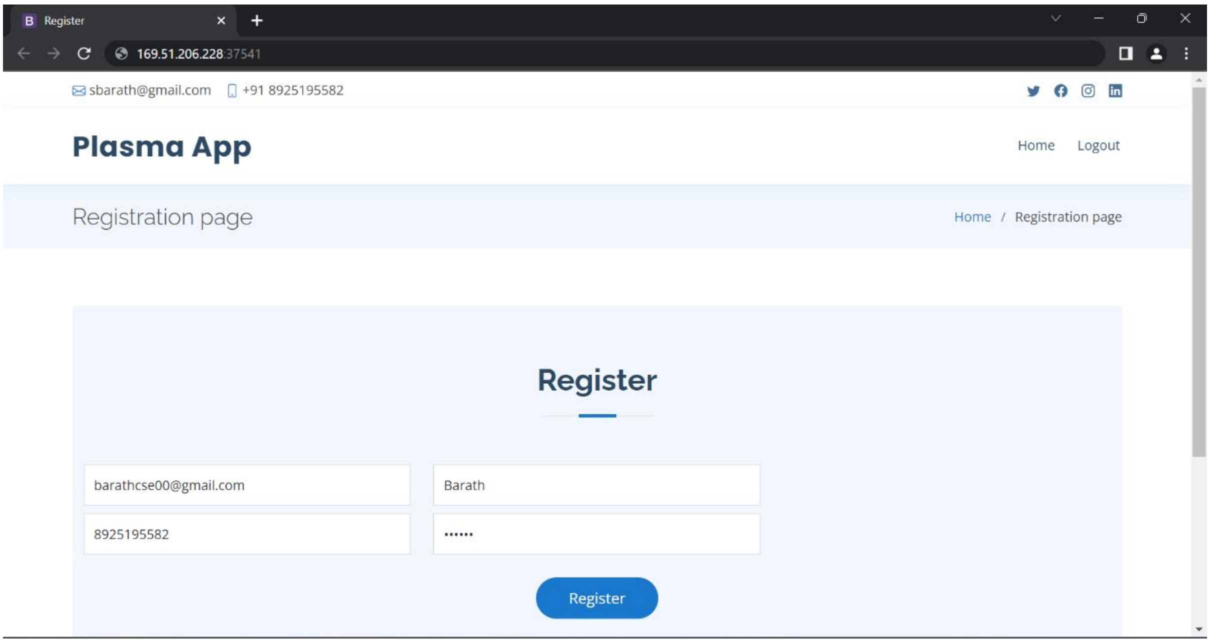
Nearly 50% of blood is made up of plasma , which itself contains water,



Contact

you can reach us through

Registration page



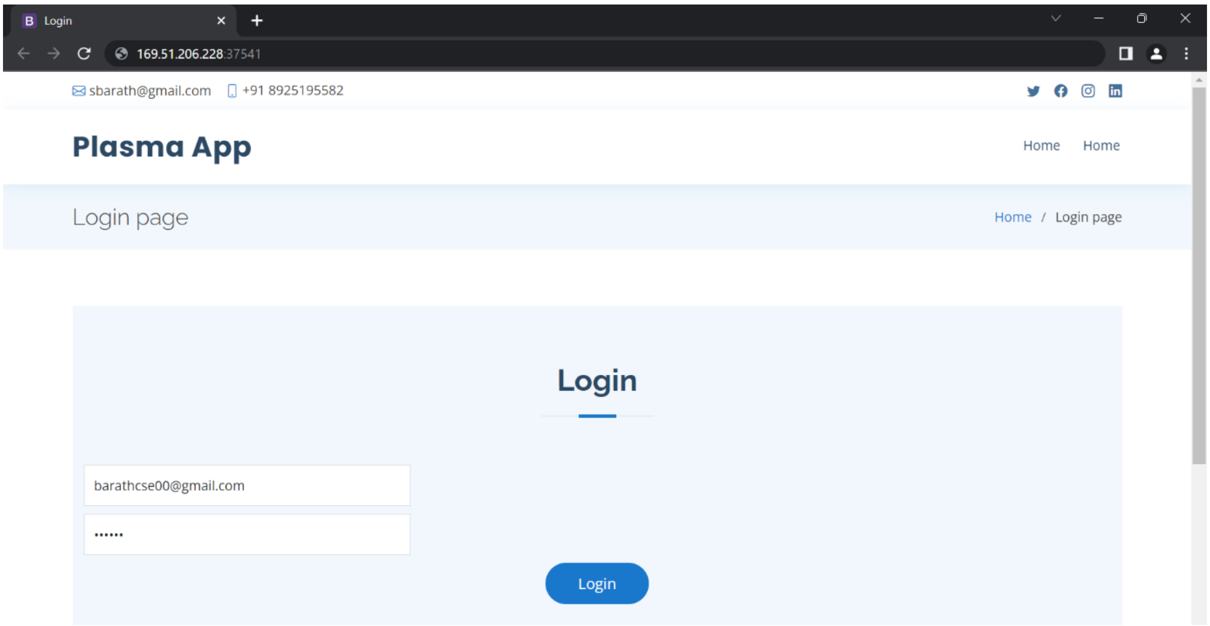
barathcse00@gmail.com

Barath

8925195582

••••••

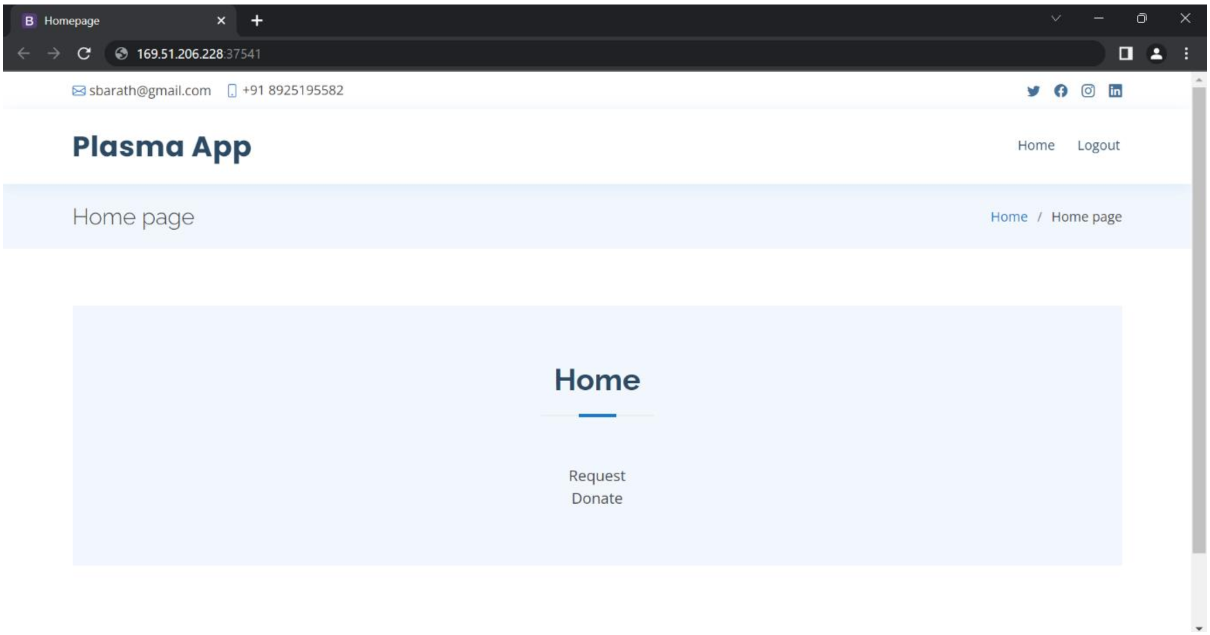
Login page



barathcse00@gmail.com

••••••

Homepage



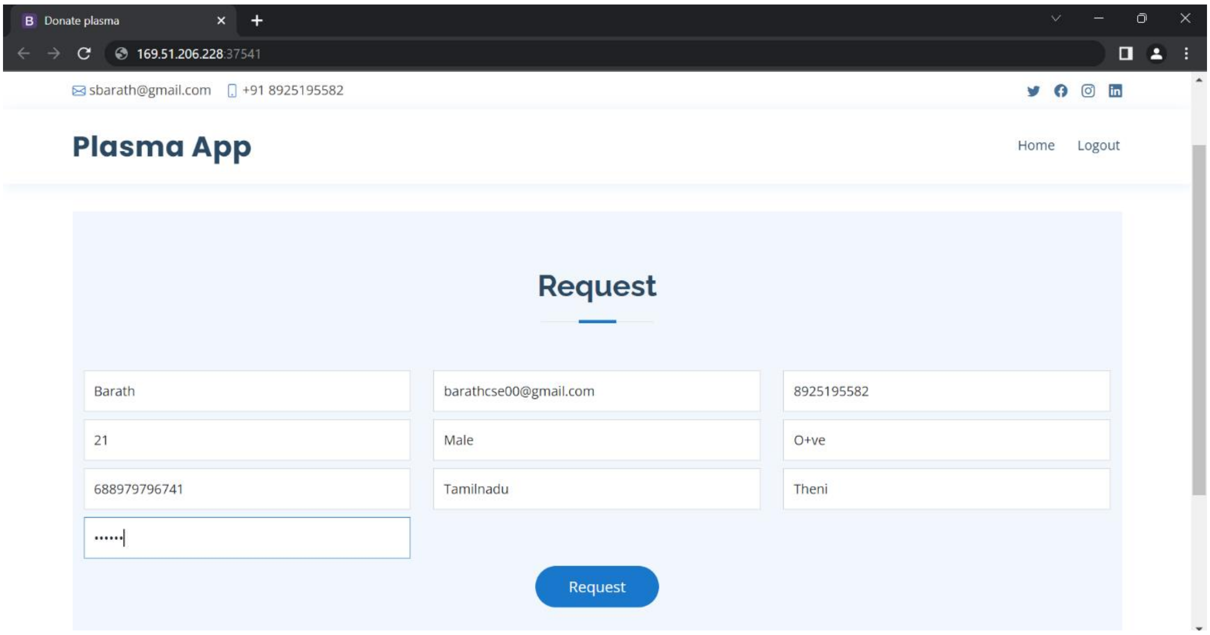
Home page

Home / Home page

Home

Request
Donate

Requesting plasma page

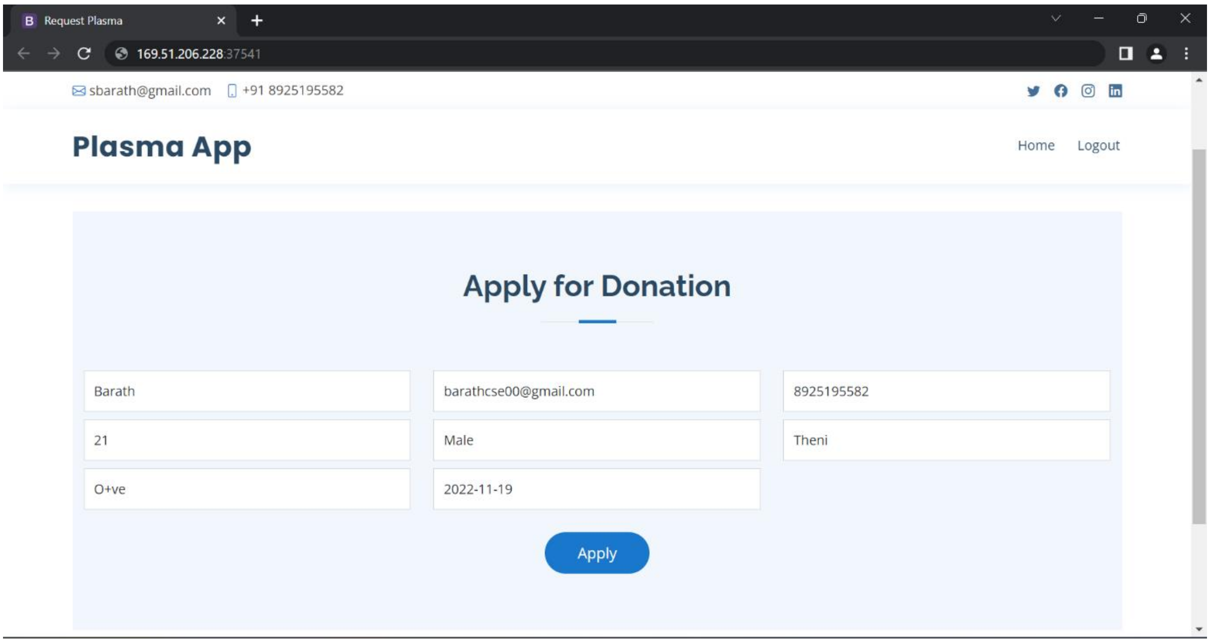


Request

Barath	barathcse00@gmail.com	8925195582
21	Male	O+ve
688979796741	Tamilnadu	Theni
••••••		

Request

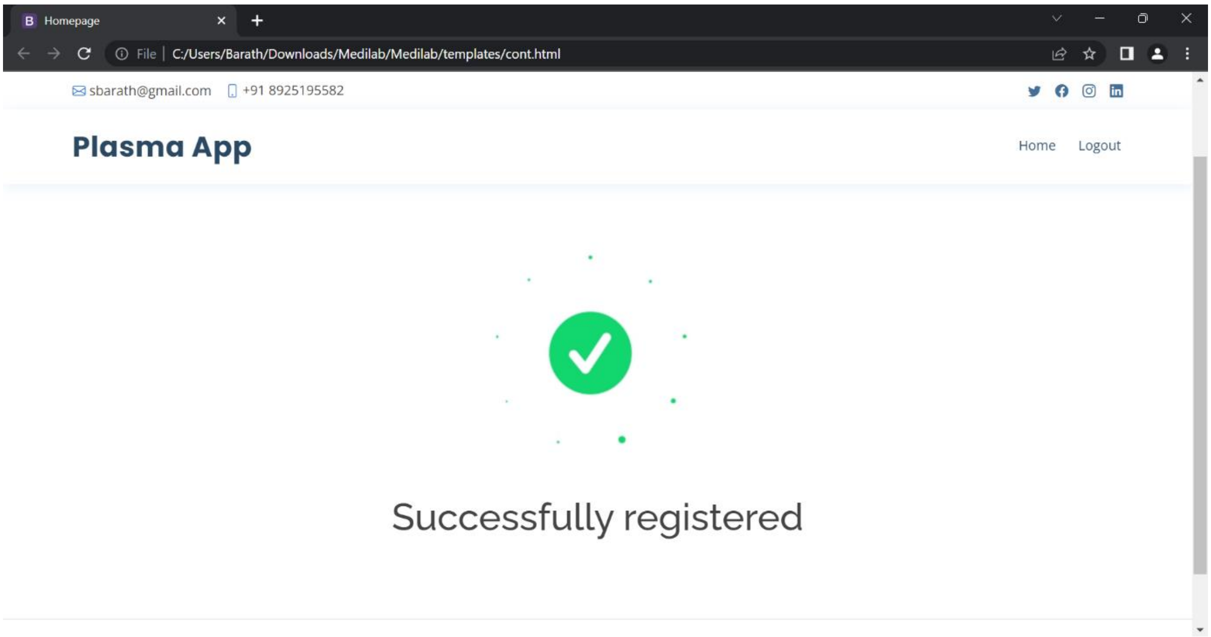
Apply for donation



Apply for Donation

Barath
barathcse00@gmail.com
8925195582
21
Male
Theni
O+ve
2022-11-19

Confirmation page



Homepage
File | C:/Users/Barath/Downloads/Medilab/Medilab/templates/cont.html
sbarath@gmail.com
+91 8925195582
Plasma App
Home
Logout

Successfully registered

CHAPTER 10

ADVANTAGES & DISADVANTAGES

10.1 ADVANTAGES

- The patient can use this plasma donor application to locate several donors.
- The donors will get notified if there is any need for plasma.
- Patients can use the application to obtain plasma; they are not required to go to the donors' location to receive assistance.
- The request process is completely an online one.

10.2 DISADVANTAGES

- Despite what people may believe given the prevalence of false donation requests, trust cannot be easily established.

CHAPTER 11

CONCLUSION

Plasma request posts have increased in frequency recently on social media platforms like Facebook, Twitter, and Instagram. It's interesting that many people want to donate plasma when there is a need, but they lack access to information about plasma donation requests in their neighbourhood. This is because there isn't a platform that would connect patients with nearby blood donors. Our application handles the situation and establishes contact with authorised hospitals when a patient needs plasma donation. This tool is useful for locating compatible plasma donors who can accept plasma request posts locally. Clinics can use this web application to carry on with the plasma donation procedure. Data gathered through this application can be used to analyse the ratio of regional requests to donations in order to hold donation camps to increase public awareness. The web service will alert the registered donors if a request for plasma is made. The notice lets the donor know if there are any requests in the application, which saves time and lets the donor know about the request. The donor might go to the clinic to donate plasma after getting the notification. By making the donor search online, the application streamlines and improves efficiency while saving the requester's time and enabling them to locate many donors. With the tagline "The correct message at the right moment will change the lives of many," the project model worked on the process of further developing the application for organ donation as well. The project will function in accordance with the needs and requirements of the client that were ascertained from them during the customer journey map and the concepts that were gathered during the ideation phase.

CHAPTER 12

FUTURE SCOPE

If a patient wants plasma, the application's e-mail feature can be used to notify the donor by email, informing them of the request in the system. Similarly, a follow-up email might be sent to ensure that the donor is easy to recall. The application can be expanded with other features like contribution camp updates and a donation leftover.

CHAPTER 13

APPENDIX

13.1 SOURCE CODE

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=98538591-7217-4024-b027-8baa776ffad1.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=30875;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;PROTOCOL=TCPIP;UID=ymg80692;PWD=dS5CZPeX9CN20vpY ","","")

# url_for('static', filename='style.css')

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

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

@app.route("/home",methods=['GET'])
def home():
    if 'email' not in session:
      return redirect(url_for('login'))
    return render_template('home.html')

@app.route("/register",methods=['GET','POST'])
def register():
  if request.method == 'POST':
    email = request.form['email']
    username = request.form['username']
    rollNo = request.form['rollNo']
    password = request.form['password']

    if not email or not username or not rollNo or not password:
      return render_template('register.html',error='Please fill all fields')

    hash=bcrypt.hashpw(password.encode('utf-8'),bcrypt.gensalt())

    query = "SELECT * FROM USER WHERE email=? OR rollNo=?"
    stmt = ibm_db.prepare(conn, query)
    ibm_db.bind_param(stmt,1,email)
    ibm_db.bind_param(stmt,2,rollNo)
    ibm_db.execute(stmt)
    isUser = ibm_db.fetch_assoc(stmt)

    if not isUser:
      insert_sql = "INSERT INTO USER(EMAIL, USERNAME, ROLLNO,
PASSWORD) VALUES (?,?,?,?)"
      prep_stmt = ibm_db.prepare(conn, insert_sql)
      ibm_db.bind_param(prep_stmt, 1, email)
      ibm_db.bind_param(prep_stmt, 2, username)
      ibm_db.bind_param(prep_stmt, 3, rollNo)
      ibm_db.bind_param(prep_stmt, 4, hash)
      ibm_db.execute(prep_stmt)
      return render_template('register.html',success="You can login")
    else:
      return render_template('register.html',error='Invalid Credentials')

  return render_template('register.html',name='Home')

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

      if not email or not password:
        return render_template('login.html',error='Please fill all fields')
      query = "SELECT * FROM USER WHERE email=?"
      stmt = ibm_db.prepare(conn, query)
      ibm_db.bind_param(stmt,1,email)
      ibm_db.execute(stmt)
      isUser = ibm_db.fetch_assoc(stmt)
      print(isUser,password)

      if not isUser:
        return render_template('login.html',error='Invalid Credentials')

      isPasswordMatch = bcrypt.checkpw(password.encode('utf-8'),isUser['PASSWORD'].encode('utf-8'))
      if not isPasswordMatch:
        return render_template('login.html',error='Invalid Credentials')

      session['email'] = isUser['EMAIL']
      return redirect(url_for('home'))

    return render_template('login.html')

@app.route("/apply",methods=['GET','POST'])
def apply():
  if request.method == 'POST':
    name = request.form.get('name')
    email = request.form['email']
    mobile = request.form['mobile']
    age = request.form['age']
    gender = request.form['gender']
    blood_group = request.form['blood_group']
    aadhar = request.form['aadhar']
    state = request.form['state']
    city = request.form['city']
    password = request.form['password']

    if not name or not email or not mobile or not age or not gender or not
blood_group or not aadhar or not state or not city or not password:
      return render_template('apply.html',error='Please fill all fields')

    insert_sql = "INSERT INTO appliedusers(NAME, EMAIL, MOBILE, AGE,
GENDER, BLOOD_GROUP, AADHAR, STATE, CITY, PASSWORD)
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, mobile)
    ibm_db.bind_param(prep_stmt, 4, age)
    ibm_db.bind_param(prep_stmt, 5, gender)
    ibm_db.bind_param(prep_stmt, 6, blood_group)
    ibm_db.bind_param(prep_stmt, 7, aadhar)
    ibm_db.bind_param(prep_stmt, 8, state)
    ibm_db.bind_param(prep_stmt, 9, city)
    ibm_db.bind_param(prep_stmt, 10, password)
    ibm_db.execute(prep_stmt)
    return redirect(url_for('confirm'))
  return render_template('apply.html')

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


@app.route("/rqst",methods=['GET','POST'])
def rqst():
  if request.method == 'POST':
    name = request.form.get('name')
    email = request.form['email']
    mobile = request.form['mobile']
    age = request.form['age']
    sex = request.form.get('sex')
    blood_group = request.form.get('blood_group')
    address = request.form['address']
    date = request.form.get('date')

    insert_sql = "insert into requsers
(name,email,number,age,sex,blood_group,address,date) 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, mobile)
    ibm_db.bind_param(prep_stmt, 4, age)
    ibm_db.bind_param(prep_stmt, 5, sex)
    ibm_db.bind_param(prep_stmt, 6, blood_group)
    ibm_db.bind_param(prep_stmt, 7, address)
    ibm_db.bind_param(prep_stmt, 8, date)
    ibm_db.execute(prep_stmt)
    return redirect(url_for('confirm'))
  return render_template('rqst.html')

@app.route('/logout')
def logout():
    session.pop('email', None)
    return redirect(url_for("index"))

@app.route('/newapply',methods=['GET','POST'])
def newapply():
  if request.method == 'POST':
    name=request.form['name'];
    email=request.form['email'];
    mobile=request.form['mobile'];
    age=request.form['age'];
    gender=request.form['gender'];
    blood_group=request.form['blood_group'];
    aadhar=request.form['aadhar'];
    address=request.form['address'];
    covid=request.form['covid'];
    nearby=request.form['nearby'];

    query = "insert into donors
(name,email,mobile,age,gender,blood_group,aadhar,address,covid,nearby)
values(?,?,?,?,?,?,?,?,?,?);"
    prep_stmt = ibm_db.prepare(conn, query)
    ibm_db.bind_param(prep_stmt, 1, name)
    ibm_db.bind_param(prep_stmt, 2, email)
    ibm_db.bind_param(prep_stmt, 3, mobile)
    ibm_db.bind_param(prep_stmt, 4, age)
    ibm_db.bind_param(prep_stmt, 5, gender)
    ibm_db.bind_param(prep_stmt, 6, blood_group)
    ibm_db.bind_param(prep_stmt, 7, aadhar)
    ibm_db.bind_param(prep_stmt, 8, address)
    ibm_db.bind_param(prep_stmt, 9, covid)
    ibm_db.bind_param(prep_stmt, 10, nearby)
    ibm_db.execute(prep_stmt)
    return redirect(url_for('confirm'))
  return render_template('newapply.html')

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

Mail.py

```
import sendgrid
import os
from sendgrid.helpers.mail import *


def register_notify(email):
    sg =
sendgrid.SendGridAPIClient(api_key=os.environ.get('SENDGRID_API_KEY')
)
    from_email = Email("sbarath786@gmail.com")
    to_email = To(email)
    subject = "PlasmaApp - Registration"
    content = Content("text/plain", "Registration for opening an account was
successfully completed. Now, you can login with your credentials.")
    mail = Mail(from_email, to_email, subject, content)
    response = sg.client.mail.send.post(request_body=mail.get())
    print(response.status_code)
    print(response.body)
    print(response.headers)



def apply_notify(email):
    sg =
sendgrid.SendGridAPIClient(api_key=os.environ.get('SENDGRID_API_KEY')
)
    from_email = Email("sbarath786@gmail.com")
    to_email = To(email)
    subject = "Plasma Donation"
    content = Content("text/plain","Your application for donation of plasma was successfully done. You will get notified once a matched request found...")
    mail = Mail(from_email, to_email, subject, content)
    response = sg.client.mail.send.post(request_body=mail.get())
    print(response.status_code)
    print(response.body)
    print(response.headers)



def request_notify(email):
    sg = sendgrid.SendGridAPIClient(api_key=os.environ.get('SENDGRID_API_KEY'))
    from_email = Email("sbarath786@gmail.com")
    to_email = To(email)
    subject = "Plasma Request"
    content = Content("text/plain", "Your request for needed plasma was succcessfully submitted. A perfect matched dono will contact you soon...")
    mail = Mail(from_email, to_email, subject, content)
    response = sg.client.mail.send.post(request_body=mail.get())
    print(response.status_code)
    print(response.body)
    print(response.headers)
```

13.2 GITHUB/PROJECT DEMO LINK

PROJECT LINK:

https://github.com/IBM-EPBL/IBM-Project-2677-1658481006

DEMO LINK:

https://youtu.be/Kdz_HOOjeOo