Project Report

Team ID	PNT2022TMID53020
Project Name	**Project-Plasma Donor Application**

1. **INTRODUCTION**
    1. Project Overview

Plasma is the liquid part of blood and more than 55% of human blood is plasma. Plasma is used to treat a variety of infections and is one of the oldest methods known as plasma therapy. Plasma therapy is a procedure in which blood is donated from a recovered patient to produce antibodies that fight infections. For example, during the COVID-19 crisis, the demand for plasma increased significantly as no vaccine was found to treat infected patients. Although the recovery rate was high with plasma therapy, the number of donors was very low and in such a situation it was very important to have information about plasma donors. Storing and Notifying about the current donors is helpful because it saves time and helps users track the information they need about their donors

The main objective is to create an easy-to-use web application that can be used by donors to donate their blood to blood banks. Hospitals in need for blood plasma can request blood from blood banks. This application should have a wider reach to appeal to more potential blood donors. In case of emergency SOS, donation of blood can be done in minuscule time.

2. Purpose

There has been an increase in demand for blood plasma among hospitals and blood banks as they are additionally used in an experimental treatment for COVID-19. Hence there is a requirement for new infrastructure to facilitate donors, blood banks and hospitals for easier donation and access of blood plasma that could potentially satisfy the excess demand for it to be used for treatment.

2. **LITERATURE SURVEY**
    1. Existing problem

The plasma donation and finding a donor has been very hard without a proper application which is specifically designed for it. During COVID 19 crisis, the requirement of plasma was very high and the donor count was low. At that point, there wasn't and application or a website which would save the donor information and help the needy by notifying the current donors list.

1. In 2015, a IEEE paper on Mobile Based Healthcare Management using Artificial Intelligence was authored by Amiya Kumar Tripathy, Rebeck Carvalho, Keshav Pawaskar, Suraj Yadav, Vijay Yadav. In this paper, the health-care management system is proposed which will consist of mobile based heart rate measurement so that the data can be transferred and diagnosis based on heart rate can be provided quickly with a click of button. The system will consist of video conferencing to connect remotely with doctor. The system will also consist of Doc-Bot and an online Blood Bank. In this implemented project, heart rate calculation differs from actual one due to noise present in input signal. So the performance is not efficient in practical. Methodology used Clustering, Text Mining, Pattern Matching, Support Vector

Machine, Partitioning Algorithm and DonorHART tool used in collecting donor reaction information. Limitations are Difficulty in handling emergency situation and No proper security for personal details misuse

2. In year 2015, a IEEE paper on A Health-IoT Platform Based on the Integration of Intelligent Packaging, Unobtrusive Bio-Sensor and Intelligent Medicine Box was authored by Geng Yang, Li Xie, Matti Mantysalo, Xiaolin Zhou, Zhibo Pang, Li Da Xu, Sharon KaoWalter, Qiang Chen, Lirong Zheng. In this paper, an intelligent home-based healthcare platform is proposed and implemented. It involves iMedBox with connectivity, iMedPack with communication capability enabled by RFID, Bio-Patch and SOC. It fuses with IoT. The body-worn Bio-Patch can detect and transmit the users bio-signals to the iMedBox in real time. The only limitations are, comprehensive platform missing. And the Physical size, rigid nature and short battery become limitation for long term use.

3. In 2016, an IEEE paper was authored on Data Mining for Better Healthcare: A Path towards Automated Data Analysis? By Tania Cerquitelli,Elena Baralis, Lia Morra and Silivia Chiusano. This paper addresses the mining activity from the medical database perspective. The mining system should be able to devise which knowledge could be most interesting to the user extract actionable knowledge from large medical dataset with minimal user intervention. System should be capable of yielding actionable knowledge extracting manageable sets. Large parameter spaces need to be explored at abstraction level to envision a system capable of evaluating and comparing many data-mining technique configurations at a time.

2. References

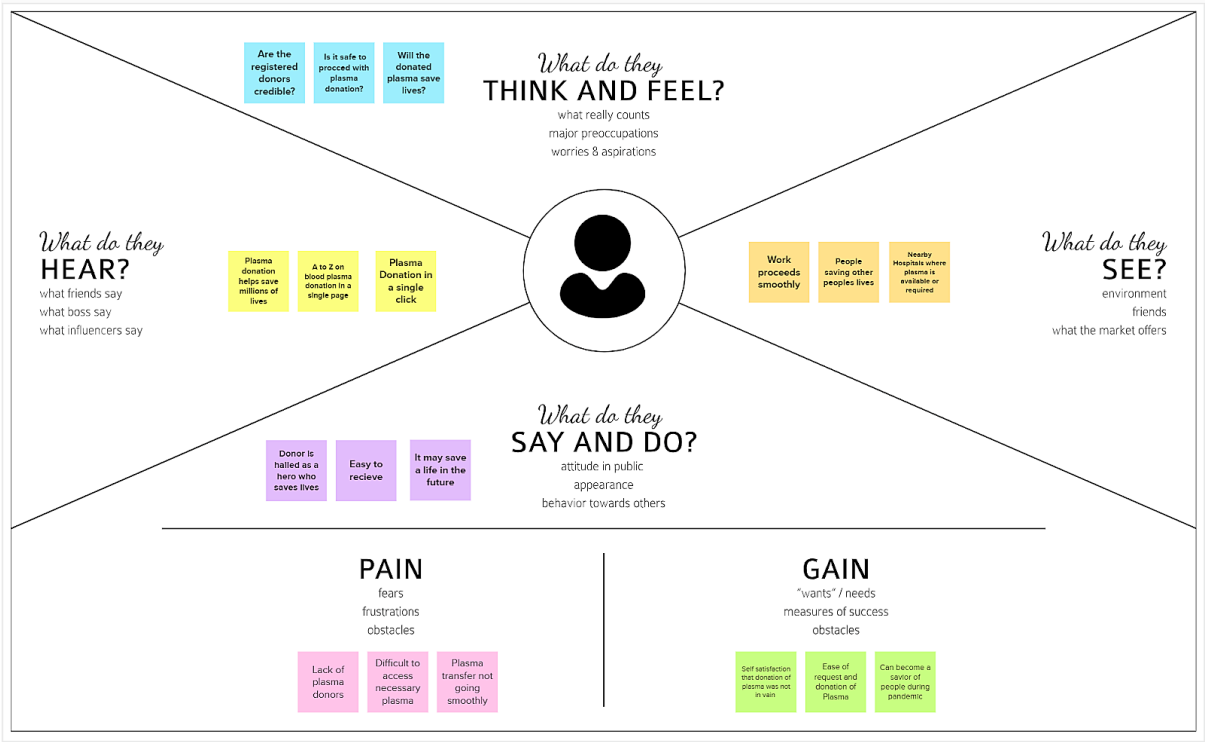
    1. France, Christopher & France, Janis & Himawan, Lina. (2022). What would it take to convince you to donate? A survey study of the relationship between motivators, barriers, and payment for whole blood, plasma, and platelet donation. Transfusion. 62.10.1111/trf.16886.
    2. Godin, Gaston & Germain, Marc. (2014). How to Motivate Whole Blood Donors to Become Plasma Donors. Journal of blood transfusion. 2014. 752182.10.1155/2014/752182.
    3. In “Short message service (SMS) based plasma” by G. Muddu Krishna & S. Nagaraju(2016)[1]. They proposed a system in which services of blood bank will be accessed via SMS. If someone needed blood then they have to request for blood via SMS and then packet count module of their system will check for availability of blood and response will be given by data processing module.
    4. Woodfield, Graeme & Ramirez, A & Everard, A & Kent, J. (1985). Volunteer blood donors for a cell pheresis programme. The New Zealand medical journal. 98. 904-6.
    5. Thorpe, Rachel & Masser, Barbara & Nguyen, Lilly & Gemelli, Carley & Davison, Tanya. (2020). Bringing new plasma donors back: testing the efficacy of telephone and e-mail communications with first-time donors. Transfusion. 60.10.1111/trf.15787.

3. Problem Statement Definition

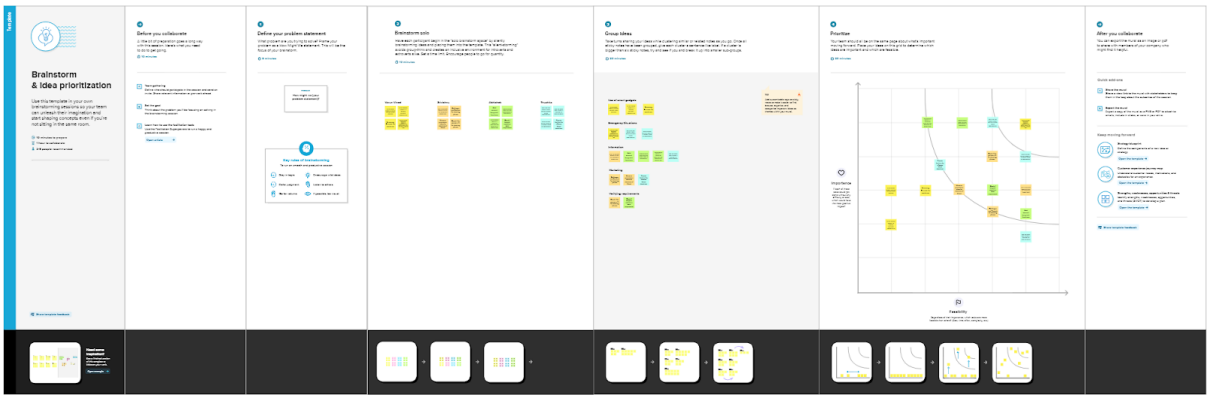
During the COVID 19 crisis, the requirement of plasma became a high priority and the donor count has become low. Saving the donor information and helping the needy by notifying the current donors list, would be a helping hand. In regard to the problem faced, an application is to be built which would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

3. IDEATION & PROPOSED SOLUTION

1. Empathy Map Canvas



2. Ideation & Brainstorming



3. Proposed Solution

S .No.	Parameter	Description

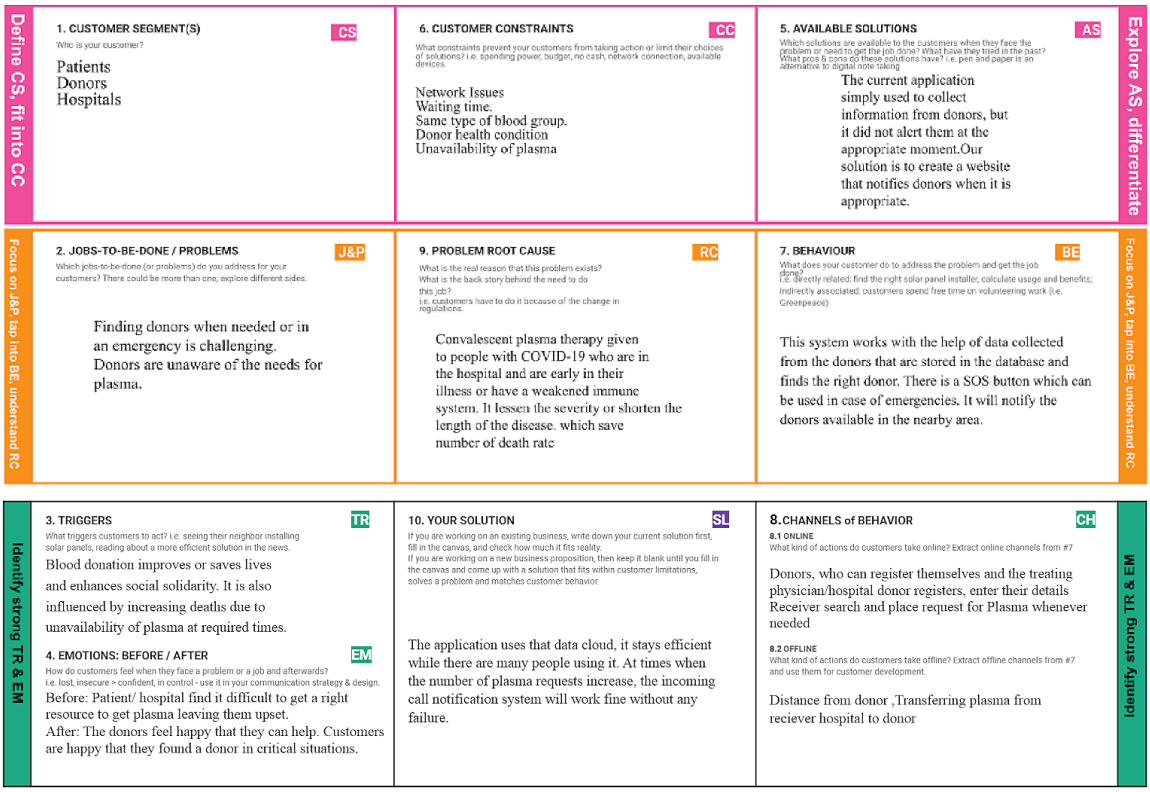
1.	Problem Statement (Problem to be solved)	To develop a web application that would allow users to donate their plasma to people who will most likely need it during an emergency. An automatic mail will also be sent to donors nearby regarding the same.
2.	Idea / Solution description	The application will allow users to register themselves on the portal for plasma donation so that recipients who require it can view their information and obtain the plasma. In addition, the hospital will automatically send a message to the closest donor to make the process easier.
3.	Novelty / Uniqueness	The app contains an entire medical unit in a single tap. Apart from normal blood donation, it maintains the person's medical records and history and intimates them once they are eligible for the next blood donation. It also sends an automated mail to the donors in case of an emergency in a hospital nearby for blood. Incase the donor accepts, his/her details are sent to the family/person requesting the same. Presence of a SOS feature for emergency plasma requirement
4.	Social Impact / Customer Satisfaction	Anyone with a basic understanding of mobile/computer applications may use the program. A lesson will also be available for convenience of access in addition to that. The software easily connects the plasma donor, the recipient, and the hospitals that need or want plasma.

5.	Business Model (Revenue Model)	The application is under the healthcare category and is free to use. It benefits those who desire to provide plasma to others in need. Data may be kept in the cloud using IBM DB2, which lowers the overall cost of creating the application.
6.	Scalability of the Solution	The programme will remain effective even when used by many users since its data will be stored on the cloud. The mail notification system will continue to function normally even when the number of requests for plasma rises.

4. Problem Solution fit

Project Title: Plasma Donor Application

Project Design Phase-I - Solution Fit



4. **REQUIREMENT ANALYSIS**
    1. Functional requirement

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)

FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Objective	Describe what the product does.
FR-4	End result	Define product features.
FR-5	Focus	Focus on user requirements.
FR-6	Documentation	Captured in use case.

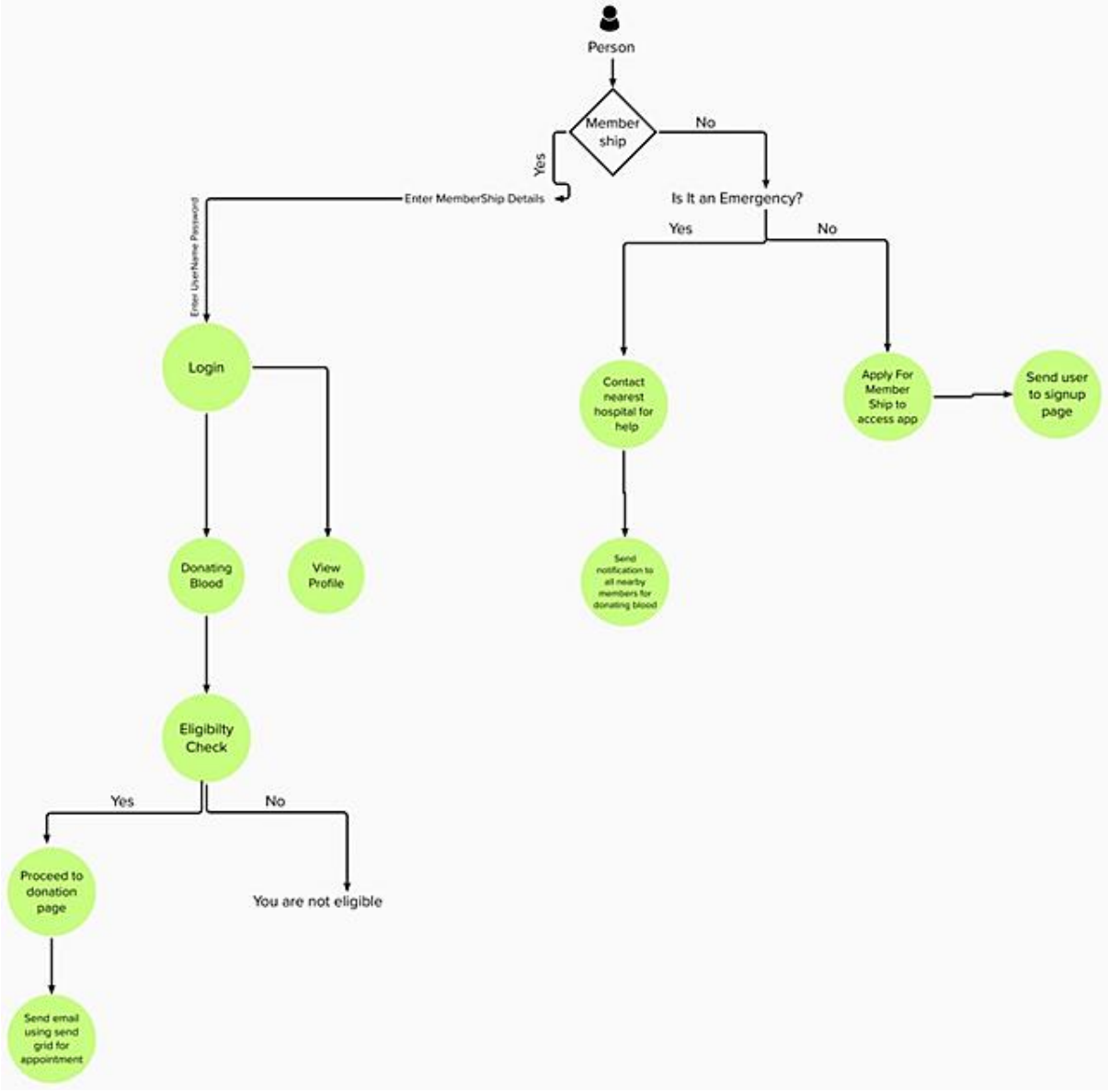
2. Non-Functional requirements

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

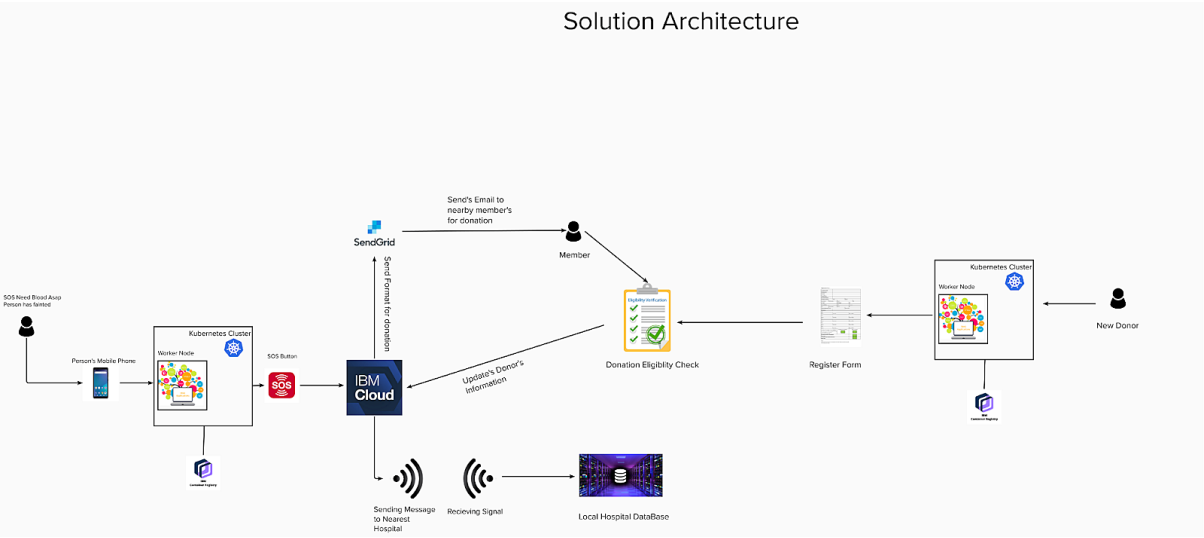
FR No.	**Non-Functional Requirement**	**Description**
NFR-1	Usability	Human Factors, overall aesthetics , consistency and documentation.
NFR-2	Security	Ability of a system to prevent unauthorised users from using it or changing their behaviour while still delivering service to authorised users.
NFR-3	Reliability	Frequency and severity of failure, recoverability, predictability , accuracy and mean time between failures(MTBF).
NFR-4	Performance	Processing speed , response time ,resource consumption, throughput and efficiency.

NFR-5	Availability	How long the system is not currently offline due to any outages or maintenance tasks. We may describe system availability using metrics such as Mean Time Between Failure (MTBF).
NFR-6	Scalability	The capability of a solution or system to improve customer service capacity and/or processing speeds to keep up with demand.

5. **PROJECT DESIGN**
    1. Data Flow Diagrams



    2. Solution & Technical Architecture



3. User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Plasma Donor)	Registration	USN-1	As a donor, I can register for the application by entering my email /Phone number, password, and confirming my password.	I can create donor account	High	Sprint -1
	Login	USN-2	Registered donor can log into the application by entering donor email & password	I can access my account / dashboard	High	Sprint -1

	Verification	USN-3	As a donor, I can enter my details to check the donor eligibility criteria including aadhar card;	I can check my eligibility to donate plasma	Medium	Sprint-2
	Dashboard	USN-4	User can provide their personal details and location	I can complete my donor profile	Low	Sprint-3
	Acceptance	USN-5	User can accept their willingness to donate plasma	I can receive confirmation e mail & click confirm	Medium	Sprint-4
Customer (Plasma Receiver)	Registration	USN-1	As a receiver, I can register for the application by entering my email /Phone number, password, and confirming my password	I can create receiver account	High	Sprint-1

	Login	USN-2	Registered receiver can log into the application by entering receiver email & password	I can access my account / dashboard	High	Sprint -1
	Verification	USN-3	As a receiver, I can enter my details to check the receiver eligibility criteria	I can check my eligibility to receive plasma	Medium	Sprint -2
	Dashboard	USN-4	User can search the list of available donor	I can choose donor according to my convenience	Low	Sprint -3
	Access	USN-5	User can access the available donors list ,then they can choose the donor who is nearby to receiver	I can receive confirmation e mail & click confirm	Medium	Sprint -4

Customer(Hospital)	Registration	USN-1	Third Party user can register for the application by entering my email /Phone number, password, and confirming my password.	I can create Third Party account	High	Sprint-1
	Login	USN-2	Registered user can log into the application by entering user email & password	I can access my account	High	Sprint-1
	Query System	USN-3	User can ask their queries via Chabot which is available 24/7 to sort user issues	I can do interrogation	Medium	Sprint-3

SOS	Verification	USN-1	For emergencies the user can send important information like blood group live location and accident proof for requesting blood without login.	I can access my application	High	Sprint -1
Administrator	Login	USN-1	Admin can log into the application by entering email & password	I can access my account / dashboard	High	Sprint -1
	Dashboard	USN-2	Admin can modify, add and remove features from the database and application	I can modify, add and remove features from the database and application	Low	Sprint -3

6. PROJECT PLANNING & SCHEDULING

1. Sprint Planning & Estimation

Here, the receivers are the hospitals who request if there is any shortage of plasma in their respective blood bank.

Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Story Point**	**Priority**	**Team Members**

Sprint-1	Registration	USN-1	As a donor, I can register for the application by entering my email Phone number, password, and confirming my password.	3	High	Varun, Srivishnu
Sprint-1	Login	USN-2	Registered donor can log into the application by entering donor email & password	3	High	Varun, Srivishnu
Sprint-2	Verification	USN-3	As a donor, I can enter my details to check the donor eligibility criteria including aadhar card;	2	Medium	Abhishek
Sprint-3	Dashboard	USN-4	User can provide their personal details and location	3	High	Thushita
Sprint-4	Acceptance	USN-5	User can accept their willingness to donate plasma	2	Medium	Thushita , Abhishek
Sprint-1	Registration	USN-1	As a hospital receptionist, I can register for the application by entering my email /Phone number, password, and confirming my password	3	High	Thushita

Sprint-2	Login	USN-2	Registered hospital receptionist can log into the application by entering receiver email & password	3	High	Thushita
Sprint-2	Verification	USN-3	As a hospital reciptionist, I can enter my details to check the receiver eligibility criteria	2	Medium	Abhishek
Sprint-4	Dashboard	USN-4	User can search the list of available donor	3	High	Varun ,Abhishek
Sprint-3	Access	USN-5	User can access the available donors list ,then they can choose the donor who is nearby to receiver	2	Medium	Thushita
Sprint-1	Query System	USN-3	User can ask their queries via Chabot which is available 24/7 to sort user issues	1	Low	Abhishek

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	10	6 Days	24 Oct 2022	29 Oct 2022	12	29 Oct 2022

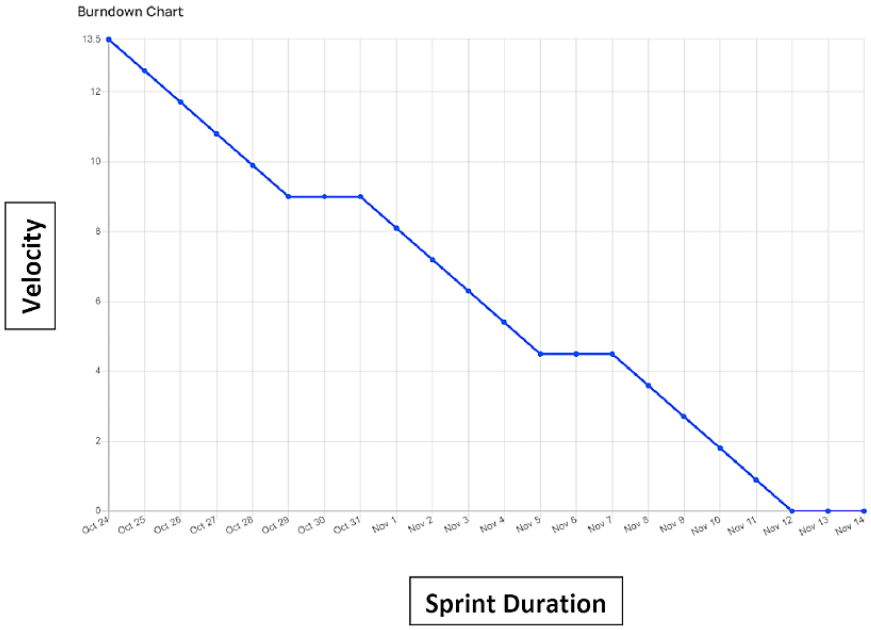
Sprint-2	7	6 Days	31 Oct 2022	05 Nov 2022	12	05 Nov 2022
Sprint-3	5	6 Days	07 Nov 2022	12 Nov 2022	12	12 Nov 2022
Sprint-4	5	6 Days	14 Nov 2022	19 Nov 2022	12	19 Nov 2022

3. Reports from JIRA

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) periteration unit (story points per day)

AV= sprint duration / velocity = 13.5/6 = 2.25



7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
    1. Feature 1

Login and Register Donor (User) and Hospital and check eligibility criteria.

Login Donor:

```
<!DOCTYPE html>
<html>
<head>
   <link rel="stylesheet" href="{{ url_for('static', filename='css/stylelsu.css') }}">
   <meta charset="utf-8">

</head>
<body>

<div class="page">
   <div class="container">
      <div class="left">
         <div class="login">Login</div>
         <div class="eula">By logging in you agree to the ridiculously long terms that
you didn't bother to read
         </div>
      </div>
      <div class="right">
         <svg viewBox="0 0 320 300">
            <defs>
               <linearGradient inkscape:collect="always" id="linearGradient" x1="13"
y1="193.49992" x2="307"
                  y2="193.49992" gradientUnits="userSpaceOnUse">
                  <stop style="stop-color:#ff00ff;" offset="0" id="stop876" />
                  <stop style="stop-color:#ff0000;" offset="1" id="stop878" />
               </linearGradient>
            </defs>
            <path
               d="m 40,120.00016 239.99984,-3.2e-4 c 0,0 24.99263,0.79932
25.00016,35.00016 0.008,34.20084 -25.00016,35 -25.00016,35 h -239.99984 c 0,-
0.0205 -25,4.01348 -25,38.5 0,34.48652 25,38.5 25,38.5 h 215 c 0,0 20,-0.99604 20,-
25 0,-24.00396 -20,-25 -20,-25 h -190 c 0,0 -20,1.71033 -20,25 0,24.00396 20,25 20,25
h 168.57143" />
         </svg>
         <form action="{{ url_for('login') }}" method="POST">
            <label for="email">Email</label>
            <input type="email" id="email" name="email" autocomplete="off"
required>
            <label for="pass">Password</label>
            <input id="pass" type="password" name="pass" autocomplete="off"
required />

            <input id="submit" type="submit" value="Log in">
         </form>
      </div>
   </div>
</div>
<script src="https://cdnjs.cloudflare.com/ajax/libs/animejs/2.0.2/anime.js"></script>
```

```
<script>
  var current = null;
  document.querySelector('#email').addEventListener('focus', function (e) {
    if (current) current.pause();
    current = anime({
      targets: 'path',
      strokeDashoffset: {
        value: 0,
        duration: 700,
        easing: 'easeOutQuart'
      },
      strokeDasharray: {
        value: '240 1386',
        duration: 700,
        easing: 'easeOutQuart'
      }
    });
  });
  document.querySelector('#pass').addEventListener('focus', function (e) {
    if (current) current.pause();
    current = anime({
      targets: 'path',
      strokeDashoffset: {
        value: -336,
        duration: 700,
        easing: 'easeOutQuart'
      },
      strokeDasharray: {
        value: '240 1386',
        duration: 700,
        easing: 'easeOutQuart'
      }
    });
  });
  document.querySelector('#submit').addEventListener('focus', function (e) {
    if (current) current.pause();
    current = anime({
      targets: 'path',
      strokeDashoffset: {
        value: -730,
        duration: 700,
        easing: 'easeOutQuart'
      },
      strokeDasharray: {
        value: '530 1386',
        duration: 700,
        easing: 'easeOutQuart'
      }
    });
  });
```

```
</script>
</body>
</html>
```

Register Donor:

```
<!DOCTYPE html>
<html lang="en">

<head>
   <meta charset="UTF-8">
   <title>sign up </title>
   <link rel="stylesheet" href="{{ url_for('static', filename='css/normalize.min.css') }}">
   <link rel="stylesheet" href="{{ url_for('static', filename='css/styler.css') }}">
   <script src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"></script>
   <script>window.jQuery || document.write('<script src="{{ url_for('static', filename='jquery.js') }}">\x3C/script>')</script>
   <script type=text/javascript src="{{ url_for('static', filename='js/script.js') }}"></script>
</head>

<body>
   <!-- partial:index.partial.html -->
   <div class="container">
      <div class="navigation">
         <ol>
            <li><a href="#" data-ref="name">Name</a></li>
            <li><a href="#" data-ref="email">Email</a></li>
            <li><a href="#" data-ref="dob">DOB</a></li>
            <li><a href="#" data-ref="pin">PIN</a></li>
            <li><a href="#" data-ref="viewpswd">Password</a></li>
            <li><a href="#" data-ref="phone">Phone</a></li>
         </ol>
      </div>
      <form id="sign-form" class="sign-form" action="{{ url_for('register') }}" method="POST">
         <ol class="questions">
            <li>
               <span><label for="name">Hi, What is your Name?</label></span>
               <input class="active" id="name" name="name" type="text" placeholder="Enter your full name" autofocus/>
            </li>

            <li>
               <span><label for="email">Enter you email</label></span>
               <input id="email" name="email" type="text" placeholder="Email" autofocus/>
            </li>
            <li>
               <span><label for="dob">Enter you dob</label></span>
```

```
            <input id="dob" name="dob" type="date" placeholder="dob"  autofocus/>
          </li>
          <li>
            <span><label for="city">Where do you reside?</label></span>
            <input id="city" name="city" type="text" placeholder="Enter your city" autofocus/>
          </li>
          <li>
            <span><label for="addr">More details please..?</label></span>
            <input id="addr" name="addr" type="text" placeholder="Enter your address" autofocus/>
          </li>
          <li>
            <span><label for="pin">Your pincode?</label></span>
            <input id="pin" name="pin" type="text" placeholder="Enter your pincode" autofocus/>
          </li>
          <li>
            <span><label for="viewpswd">Choose a password</label></span>
            <input id="viewpswd" name="password" type="text" placeholder="this your password"/>
            <input id="password" name="password" type="password" placeholder="make sure you dont forget" autofocus/>
            <span id="show-pwd" class="view"></span>
          </li>
          <li>
            <span><label for="phone">Enter your phone number</label></span>
              <select name="countryCode" class="country"  >
                <option data-countryCode="GB" value="91">(+91) INDIA </option>
              </select>
              <input id="phone" name="phone" type="text" autofocus/>
          </li>
          <li><p style="margin-top: 45px;font-size: 32pt;float:right">
            <button href="#" type="submit" style="color:white;text-decoration:none;background-color: #B7BED8;border: none;" id="signup">sign up</a></p>  </li>


        </ol>
        <div id="next-page" alt="Kiwi standing on oval"></div>
        <div class="error-message"></div>

      </form>
      <h1 id="wf" style="opacity:0;width: 600px; margin-top:1.1em;display:none; margin-bottom:1em">Thank you</h1>
      <!-- partial -->
      <!--                                                                    <script src='https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.0/jquery.min.js'></script> -->
```

```
</body>

</html>
```

Eligibility (Health Card):

```
<html>

<head>
  <title>Health Card</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="{{ url_for('static', filename='css/style2.css') }}">
</head>
<style>
  .rad {
    display: block;
    position: relative;
    padding-left: 35px;
    margin-bottom: 12px;
    cursor: pointer;
    font-size: 22px;
    -webkit-user-select: none;
    -moz-user-select: none;
    -ms-user-select: none;
    user-select: none;
  }

  /* Hide the browser's default radio button */
  .rad input {
    position: absolute;
    opacity: 0;
    cursor: pointer;
  }

  /* Create a custom radio button */
  .checkmark {
    position: absolute;
    top: 0;
    left: 0;
    height: 25px;
    width: 25px;
    background-color: #ddd;
    border-radius: 50%;
  }

  /* On mouse-over, add a grey background color */
  .rad:hover input~.checkmark {
    background-color: #ccc;
  }
```

```
/* When the radio button is checked, add a blue background */
.rad input:checked~.checkmark {
  background-color: #2196F3;
}

input[type=text],
select,
textarea {
  width: 100%;
  padding: 12px;
  color: azure;
  background-color: transparent;
  border: 1px solid rgb(201, 192, 192);
  border-radius: 4px;
  resize: vertical;
}

input[type=email],
select,
textarea {
  width: 100%;
  padding: 12px;
  color: azure;
  background-color: transparent;
  border: 1px solid rgb(200, 191, 191);
  border-radius: 4px;
  resize: vertical;
}

input[type=number],
select,
textarea {
  width: 100%;
  padding: 12px;
  color: azure;
  background-color: transparent;
  border: 1px solid rgb(200, 191, 191);
  border-radius: 4px;
  resize: vertical;
}

label {
  padding: 12px 12px 12px 0;
  display: inline-block;
}

input[type=submit] {
  background-color: rgb(0, 0, 0);
  color: beige;
  padding: 12px 20px;
```

```
  border: 1px solid #45a049;
  border-radius: 4px;
  cursor: pointer;
  float: right;
}

input[type=submit]:hover {
  background-color: #45a049;
  border: none;
}

input[type=reset] {
  background-color: rgb(0, 0, 0);
  color: beige;
  padding: 12px 20px;
  border: 1px solid #e48204;
  border-radius: 4px;
  cursor: pointer;
  float: left;
}

input[type=reset]:hover {
  background-color: #e45700;
  border: none;
}

input[type=date] {
  background-color: azure;
  width: 15%;
  border-radius: 10%;
  color: black;
  border: none;
  padding: 12px 20px;
}

select,
textarea {
  background-color: transparent;
  color: aliceblue;
}

option {
  color: aliceblue;
  background-color: black;
  line-height: 8px;
  font-size: medium;
}

.droptarget {
  float: left;
```

```
    width: 100px;
    height: max-content;
    background-color: #45a049;
    margin: 15px;
    border-radius: 5px;
    padding: 10px;
    border: 1px solid #aaaaaa;
  }

  .dropsrc {
    float: left;
    width: 100px;
    height: max-content;
    background-color: rgb(32, 10, 96);
    border-radius: 5px;
    margin: 15px;
    padding: 10px;
    border: 1px solid #aaaaaa;
  }

  .container {
    border-radius: 5px;
    background-color: rgba(10, 4, 58, 0.5);
    transition: all 4s ease;

    padding: 20px;
  }
</style>

<body style="text-align: left;font-size:large;font-family:Arial, Helvetica, sans-serif;">

  <h1 style="font-size: 50px;color: azure;">Health Card</h1>

  <div class="container">
    <form action="{{ url_for('eligibilty') }}" method="POST"
enctype="multipart/form-data">

      <h2>These details are recorded and user can view them from their dashboard. All
the data are secured and privacy
        is assured</h2>
      Gender:<br><br>
      <label class="rad">male<input type="radio" value="male" name="gender"
checked><span
          class="checkmark"></span></label>
      <label class="rad">female<input type="radio" value="female"
name="gender"><span class="checkmark"></span></label>
      <label class="rad">other<input type="radio" value="other" name="gender"
/><span class="checkmark"></span></label>
      <br><br><br>
      Blood Group:<br><br>
```

```
    <label class="rad">A+<input type="radio" value="A+" name="bg" checked><span class="checkmark"></span></label>
    <label class="rad">A-<input type="radio" value="A-" name="bg"><span class="checkmark"></span></label>
    <label class="rad">B+<input type="radio" value="B+" name="bg" /><span class="checkmark"></span></label>
    <label class="rad">AB+<input type="radio" value="AB+" name="bg"><span class="checkmark"></span></label>
    <label class="rad">AB-<input type="radio" value="AB-" name="bg"><span class="checkmark"></span></label>
    <label class="rad">B-<input type="radio" value="B-" name="bg" /><span class="checkmark"></span></label>
    <label class="rad">O+<input type="radio" value="O+" name="bg"><span class="checkmark"></span></label>
    <label class="rad">O-<input type="radio" value="O-" name="bg"><span class="checkmark"></span></label>
    <br><br><br>
    <label>Weight</label>
    <input style="width: fit-content;" type="number" name="weight" placeholder="Enter Weight...">

    <label>        Height</label>
    <input style="width: fit-content;" type="number" name="height" placeholder="Enter Height...">
    <br><br><br>
    Do you have cholestorol?<br>
    <label>Yes<input type="radio" id="cchk" name="cchk" value="Yes" onclick="td('chol')" checked> </label>
    <label>No<input type="radio" id="cchk" name="cchk" value="No"onclick="hd('chol')"></label>
    <br><br>
    <container id="chol">
      <label>Enter Cholestrol: </label>
      <input style="width: fit-content;display: block;" type="number" name="chol"
        placeholder="Enter Cholesterol value...">
    </container>
    <br><br>

    <label>Blood Pressure: </label>
    <input type="number" name="bp" placeholder="Enter BP...." style="width: fit-content;display: block">
    <br><br>
    <br><br>Have you donated blood?
    <br><br>
    <label>Yes<input type="radio" id="bchk" name="bchk" value="Yes" onclick="td('blood')" checked> </label>
    <label>No<input type="radio" id="bchk" name="bchk" value="No" onclick="hd('blood')"></label>
    <br><br>
    <container id="blood">
```

```
      <label>Last Date of Blood Donation:</label><br>
      <input name="blood" type="date" value="NULL" placeholder="Enter Date">
      <br>
      <label>Blood Donation Certificate:(optional)</label> <input type="file" name="certi" accept=".pdf" />
      <br><br>
    </container>

    <label>In the last six months have you had any of the following?</label>
    <br>
    <label>Tatooing<input type="checkbox" name="check1" value="Tatooing"></label><br>
    <label> Ear Piercing <input type="checkbox" name="check1" value="Ear_Piercing"></label><br>
    <label>Dental Extraction<input type="checkbox" name="check1" value="Dental_Extraction"></label><br>
<br>
    <label>Is there any history of surgery or blood transfusion in the past six months?</label>
    <br>
    <label>Antibiotics<input type="checkbox" name="check2" value="Antibiotics"></label><br>
    <label> Steroids <input type="checkbox" name="check2" value="Steroids"></label><br>
    <label>Aspirin<input type="checkbox" name="check2" value="Aspirin"></label><br>
    <label>Vaccinations<input type="checkbox" name="check2" value="Vaccinations"></label><br>
    <label> Alcohol <input type="checkbox" name="check2" value="Alcohol"></label><br>
    <label>Dog bite Rabies vaccine (1 year)<input type="checkbox" name="check2" value="Rabies"></label><br>
<br><br>
<label>Is there any history of surgery or blood transfusion in the past six months?</label>
<br>
<label>Major<input type="checkbox" name="check3" value="Major"></label><br>
<label>Minor<input type="checkbox" name="check3" value="Minor"></label><br>
<label>Blood Transfusion<input type="checkbox" name="check3" value="Blood Transfusion"></label><br>
    <input type="submit" value="Skip/Submit">
    <input type="reset">

  </form>
 </div>
 <script type="text/javascript">
  function alphanumeric(ele) {
   var data = ele.value;
   var pattern = /^[a-zA-Z0-9\n]*$/;
   if (!pattern.test(data)) {
```

```
      alert("text area alpha numeric only");
    }
  }
  function td(y) {
    var x = document.getElementById(y);
    x.style.display = "block";

  }
  function hd(y) {
    var x = document.getElementById(y);

    x.style.display = "none";
  }
 </script>
</body>

</html>
```

Login Hospital:

```
<!DOCTYPE html>
<html>

<head>
  <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">
  <link
href="https://fonts.googleapis.com/css2?family=Jost:wght@500&display=swap"
rel="stylesheet">

  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Log In</title>
  <script>
    function otp(){
      document.getElementById("otp").style.display='block';
      document.getElementById("pass").style.display='none';
    }
  </script>
</head>

<body>
  <div class="header">
    <h1>VAST</h1>
  </div><br>
  <div class="signup">
    <form action="{{ url_for('hospitallogin') }}" method="POST">
      <!-- {{msg}} -->
      <input type="text" id="uid" name="uid" placeholder="uid" required>
      <input id="pass" type="password" placeholder="Enter Password"
name="pass" required />
```

```
        <!-- <input style="display:none" id="otp" type="number" placeholder="Enter OTP" name="otp" required /> -->
        <button id ="submit" type="submit">Log in</button>
        <!-- <button onclick="otp();" id="login">Log in via OTP</button>
        <a href="/register">
        <button id="signuip">Sign up</button></a> -->
      </form>
    </div>
</body>

</html>
```

2. Feature 2
   Hospitals will request plasma for their patients and the Donor will be accepting the request if they are eligible and the Hospital will be notified that the Donor has accepted the request.

   Hospital Request plasma for Patient:

```
<!DOCTYPE html>
<html>

<head>
  <title>New Request</title>
</head>

<body>
  <form action="{{ url_for('hospitalrequest') }}" method="POST" enctype="multipart/form-data">
      <input type="text" name="pname" placeholder="Patient name">
      Blood Group:<br>
      <label class="rad">A+<input type="radio" value="A+" name="bg" checked><span class="checkmark"></span></label>
      <label class="rad">A-<input type="radio" value="A-" name="bg"><span class="checkmark"></span></label>
      <label class="rad">B+<input type="radio" value="B+" name="bg" /><span class="checkmark"></span></label>
      <label class="rad">AB+<input type="radio" value="AB+" name="bg"><span class="checkmark"></span></label>
      <label class="rad">AB-<input type="radio" value="AB-" name="bg"><span class="checkmark"></span></label>
      <label class="rad">B-<input type="radio" value="B-" name="bg" /><span class="checkmark"></span></label>
      <label class="rad">O+<input type="radio" value="O+" name="bg"><span class="checkmark"></span></label>
      <label class="rad">O-<input type="radio" value="O-" name="bg"><span class="checkmark"></span></label>
      <br><br>
      SOS?<br>
      <label class="rad">Yes<input type="radio" value="high" name="priority" checked><span class="checkmark"></span></label>
```

```
      <label class="rad">No<input type="radio" value="low" name="priority"><span class="checkmark"></span></label>
      Blood Camp?<br>
      <label class="rad">Yes<input type="radio" value="yes" name="blood_camp" checked><span class="checkmark"></span></label>
      <label class="rad">No<input type="radio" value="no" name="blood_camp"><span class="checkmark"></span></label>
      <input type="submit" value="Submit">
     <input type="reset">
   </form>
</body>

</html>
```

Donor Accept Request:

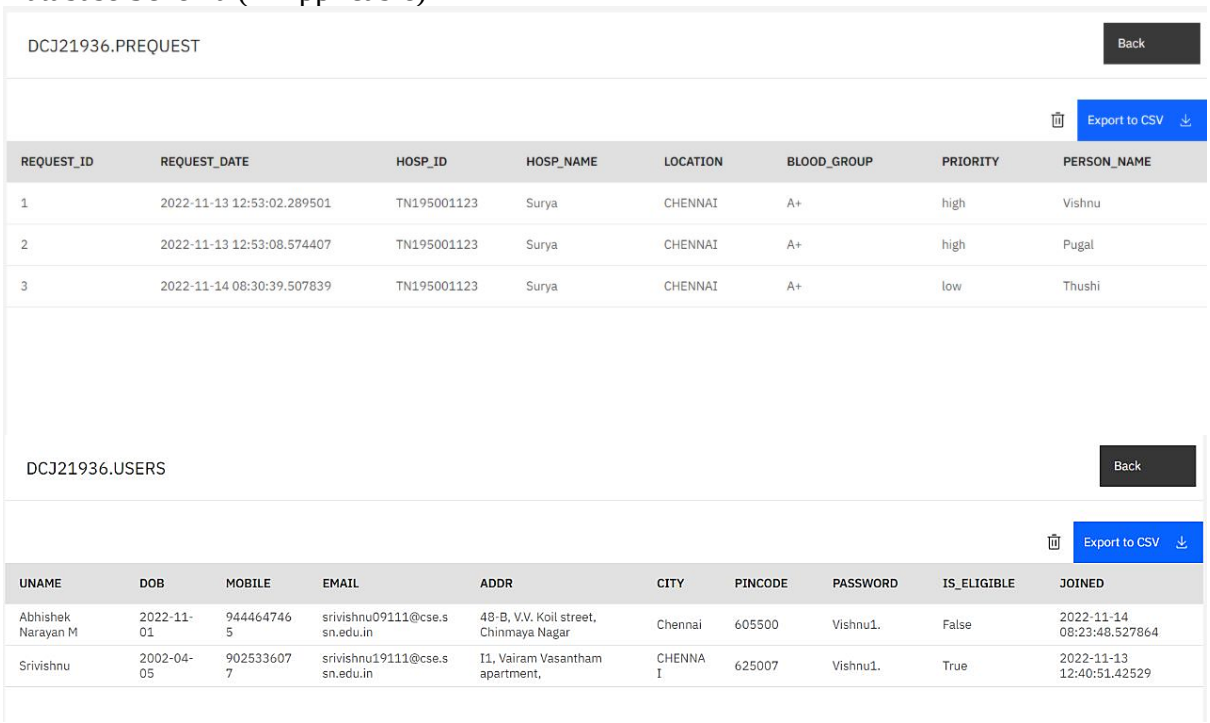
```
<html>

<head>
   <title>Donate</title>
   <meta name="viewport" content="width=device-width, initial-scale=1">

   <link href="https://fonts.googleapis.com/css?family=Roboto" rel="stylesheet">
   <link rel="stylesheet" href="https://fonts.googleapis.com/css2?family=Material+Symbols+Outlined:opsz,wght,FILL,GRAD@48,400,0,0" />
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/meyer-reset/2.0/reset.min.css">
   <link rel="stylesheet" href="{{ url_for('static', filename='css/stylen.css') }}">
</head>
<style>
   a:visited {
      color: #848697;
   }

   a {
      color: #848697;
      text-decoration: none;
   }
</style>

<body>
   <div class="center">
      <div class="left">
         <div class="logo">
            <img src="{{url_for('static', filename='6.png')}}" height="100" width="100">
         </div>
         <div class="company">
            <div class="company-name">Plasma donor Application</div>
            <div class="company-description">Donate plasma, save lives</div>
         </div>
```

```
        <div class="navigation">
          <ul>
            <a href="\dashboard">
              <li><i                                    class="material-symbols-
outlined">home</i><span>Home</span></li>
            </a>
            <a href="#">
              <li      style="color:      blue;">      <i      class="material-symbols-
outlined">bloodtype</i><span>Donate</span>
              </li>
            </a>
            <li>                    <i                    class="material-symbols-
outlined">medical_information</i><span>Health Card</span></li>
            <li><i          class="material-symbols-outlined">logout</i><span>Log
Out</span></li>

          </ul>
        </div>
      </div>
      <div class="right">
        <div class="title">Available Requests</div>
        <br><br>
        <span class="remove" onclick="rem('SOS','r1')"><i
            class="material-symbols-outlined">e911_emergency</i>Emergency
Requests
          <i id="r1" class="material-symbols-outlined">expand_less</i></span>
          <form action="{{ url_for('approved') }}" method="POST">
        <div id="SOS"><br><br>
          <input type="date" name="dondate" placeholder="Date of donation"
required>
          <table>
            <tr>
              <th>Patient Name</th>
              <th>Hospital Name</th>
              <th>Hospital Location</th>
              <th>Accept/deny</th>
            </tr>
            <br><br>{% for i in items %}
            <tr>
              <td>{{i[7]}}</td>
              <td>{{i[3]}}</td>
              <td>{{i[4]}}</td>
              <td><button                type="submit"                name="chec"
value="{{i[0]}}">Accept</button></td>
            </tr>{% endfor %}
          </table>
        </div></form>

        <br><br>
        <span class="remove1" onclick="rem('req','r2')"><i
```

```
            class="material-symbols-outlined">medication_liquid</i>Other Requests
          <i id="r2" class="material-symbols-outlined">expand_less</i></span>
        <div id="req"><br><br>
          <form action="{{ url_for('approved') }}" method="POST">
          <input type="date" name="dondate2" placeholder="Date of donation"
required>
          <table>
            <tr>
              <th>Patient Name</th>
              <th>Hospital Name</th>
              <th>Hospital Location</th>
              <th>Accept/deny</th>
            </tr>
            <br><br>{% for i in items %}
            <tr>
              <td>{{i[7]}}</td>
              <td>{{i[3]}}</td>
              <td>{{i[4]}}</td>
              <td><button type="submit" name="chec"
value="{{i[0]}}">Accept</button></td>
            </tr>{% endfor %}
          </table>
        </div></form>
        <br><br>
        <span class="remove2" onclick="rem('cam','r3')"><i class="material-symbols-
outlined">campaign</i>Blood Camps
          <i id="r3" class="material-symbols-outlined">expand_less</i></span>
        <div id="cam"><br><br>
          <form action="{{ url_for('locate') }}" method="POST">
          <table>
            <tr>
              <th>Camp name</th>
              <th>Organised by</th>
              <th>Address</th>
              <th>Location</th>
            </tr>
            <br><br>{% for i in items %}
            <tr>
              <td>{{i[7]}}</td>
              <td>{{i[3]}}</td>
              <td>{{i[4]}}</td>
              <td><button type ="submit" name="map" style="color:blue"
value="{{i[0]}}"><i class="material-symbols-outlined">near_me</i></a></td>
            </tr>{% endfor %}
          </table>
        </div>
      </div>
    </div>
</body>
<script>
```

```
    function rem(y, z) {
        var x = document.getElementById(y);
        var a = document.getElementById(z)
        if (!(x.style.display == "none"))
            x.style.display = "none";
        else
            x.style.display = "block";
        if ((a.innerHTML == "expand_less"))
            a.innerHTML = "expand_more";
        else
            a.innerHTML = "expand_less";
    }
</script>

</html>
```

3. Database Schema (if Applicable)



DCJ21936.PREQUEST

Back

Export to CSV

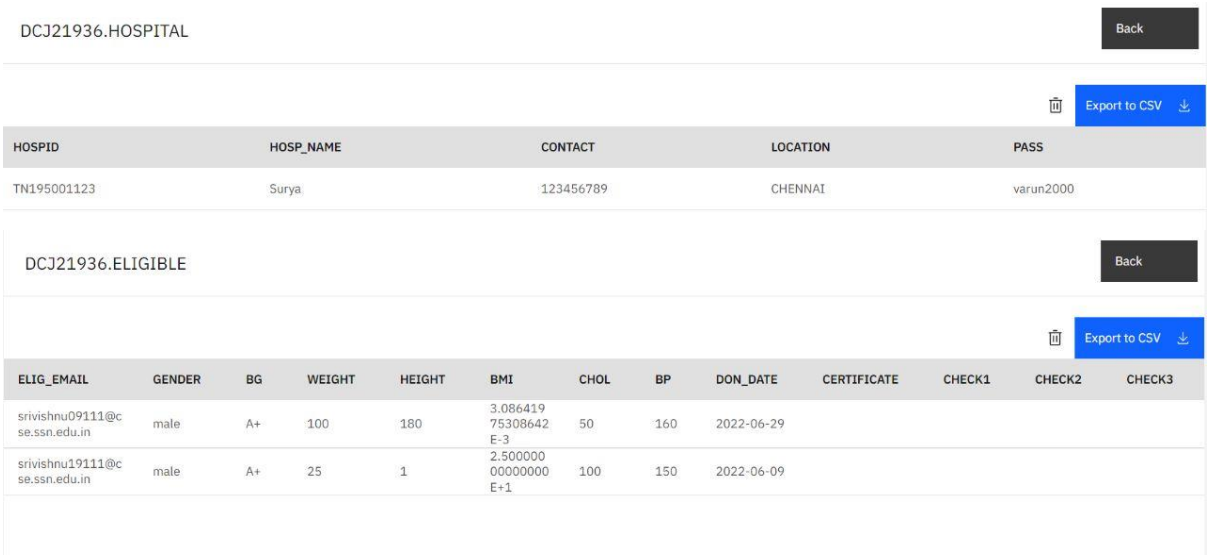
REQUEST_ID	REQUEST_DATE	HOSP_ID	HOSP_NAME	LOCATION	BLOOD_GROUP	PRIORITY	PERSON_NAME
1	2022-11-13 12:53:02.289501	TN195001123	Surya	CHENNAI	A+	high	Vishnu
2	2022-11-13 12:53:08.574407	TN195001123	Surya	CHENNAI	A+	high	Pugal
3	2022-11-14 08:30:39.507839	TN195001123	Surya	CHENNAI	A+	low	Thushi

DCJ21936.USERS

Back

Export to CSV

UNAME	DOB	MOBILE	EMAIL	ADDR	CITY	PINCODE	PASSWORD	IS_ELIGIBLE	JOINED
Abhishek Narayan M	2022-11-01	9444647465	srivishnu09111@cse.ssn.edu.in	48-B, V.V. Koil street, Chinmaya Nagar	Chennai	605500	Vishnu1.	False	2022-11-14 08:23:48.527864
Srivishnu	2002-04-05	9025336077	srivishnu19111@cse.ssn.edu.in	I1, Vairam Vasantham apartment,	CHENNAI	625007	Vishnu1.	True	2022-11-13 12:40:51.42529



DCJ21936.HOSPITAL

Back

Export to CSV

HOSPID	HOSP_NAME	CONTACT	LOCATION	PASS
TN195001123	Surya	123456789	CHENNAI	varun2000

DCJ21936.ELIGIBLE

Back

Export to CSV

ELIG_EMAIL	GENDER	BG	WEIGHT	HEIGHT	BMI	CHOL	BP	DON_DATE	CERTIFICATE	CHECK1	CHECK2	CHECK3
srivishnu09111@cse.ssn.edu.in	male	A+	100	180	3.086419 75308642 E-3	50	160	2022-06-29				
srivishnu19111@cse.ssn.edu.in	male	A+	25	1	2.500000 00000000 E+1	100	150	2022-06-09				

8. **TESTING**
    1. Test Cases

Tabulation of Test Cases

TEST CASE ID	TEST SCENARIO	TESTING STEPS	TEST DATA	EXPECTED OUTCOME	ACTUAL OUTCOME	PASS OR FAIL
T01	User authentication	Enter login credentials like user id and password, submit the credentials	userid: admin@ssn password: 1	Invalid (incorrect password)	Valid	Fail
T02	User authentication	Enter login credentials like user id and password, submit the credentials	userid: admin@ssn password: 123	Valid	Valid	Pass
T03	New user registration	Click register button, enter name id, hospital name, blood, emergency and submit the form	name id, hospital name, blood, emergency.	Not Eligible	Eligible and email is sent	Fail

T04	New user registration	Click register button, enter name id, hospital name, blood, emergency and submit the form	name id, hospital name, blood, emergency.	Eligible and email is sent	Eligible and email is sent	Pass
T05	View Eligibility (Health Care Details)	Navigate to Donor Information section, see the Health Care information	Donation donated less than 3 months ago	Not Eligible	Eligible	Fail
T06	View Eligibility (Health Care Details)	Navigate to Donor Information section, see the Health Care information	Donation has donated more than 3 months ago	Eligible	Eligible	Pass
T07	Update Health Care Details and receive email	Choose “Update” button and the Health Card information can be edited and saved	address,phone,health conditions	Not updated	Successfully updated details email is sent	Fail
T08	Update Health Care Details and receive email	Choose “Update” button and the Health Card information can be edited and saved	address,phone,health conditions	Successfully updated details email is sent	Successfully updated details email is sent	Pass

T09	Patient Requests	click donate tab in dashboard and see requests for plasma	Donor and Patient are not in the same city or have same blood group	Requests of patients are not shown	Requests of patients are shown	Fail
T10	Patient Requests	click donate tab in dashboard and see requests for plasma	Donor and Patient are not in the same city or have same blood group	Requests of patients with same city and blood group as donor is shown	Requests of patients with same city and blood group as donor is shown	Pass
T11	Accept to donate email	Click the accept button to accept the request and email is sent	accepted button is clicked	Successfully accepted and email sent.	Not Successfully accepted and no email sent.	Fail
T12	Accept to donate email	Click the accept button to accept the request and email is sent	accepted button is clicked	Successfully accepted and email sent.	Successfully accepted and email sent.	Pass
T13	Display dashboard for acceptance	click dashboard and see accepted	Donor not accepted	Accepted Not visible	Accepted visible	Fail
T14	Display dashboard for acceptance	click dashboard and see accepted	Donor accepted	Accepted visible	Accepted visible	Pass
T15	Requests to donor	Send requests to donor.	request button is clicked	Request sent successfully	Request not sent successfully	Fail
T16	Requests to donor	Send requests to donor.	request button is clicked	Request sent successfully	Request sent successfully	Pass
T17	View accept	View the accept	accepted button is clicked by donor	Accepted message is visible in dashboard.	Accepted message is not visible in dashboard.	Fail

T18	View accept	View the accept	accepted button is clicked by donor	Accepted message is visible in dashboard.	Accepted message is visible in dashboard.	Pass

2. User Acceptance Testing

8.2.1 Purpose of Document

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

8.2.2 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	8	3	2	2	15
Duplicate	1	0	3	1	5
External	2	2	1	1	6
Fixed	11	5	6	4	26
Not Reproduced	0	1	0	0	1
Skipped	0	1	0	0	1
Won't Fix	0	0	0	0	0
Totals	11	12	12	4	39

8.2.3 Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	36	0	0	36
Security	6	0	0	6
Outsource Shipping	4	0	0	4
Exception Reporting	6	0	0	6
Final Report Output	5	0	0	5

Version Control	2	0	0	2

RESULTS

1. Performance Metrics

These performance can be performed in our system

- View eligibility to donate
- Accept to donate
- Help people in need by viewing emergency requirements
- View donation camps nearby
- Save blood donation certificate
- View and update your health card
- You'll get nearby hospital information on where to donate with the location

10. **ADVANTAGES & DISADVANTAGES**

1. ADVANTAGES

1. **Spreading Awareness:** Vast can be very useful to check eligibility to donate, accept to donate, help people in need by viewing emergency requirements, view donation camps nearby,
2. **Accessibility:** Save blood donation certificates, view and update your health card, and get nearby hospital information on where to donate with the location,etc.
3. **Speed:** This website is fast and offers great accuracy as compared to manual registered keeping.
4. **Maintenance:** Less maintenance is required, Easy to Request and Accept donations.
5. **User Friendly:** It is very easy to use and understand. It is easily workable and accessible for everyone.
6. **Fast Results:** It would help you to provide plasma donors easily depending upon the availability of it

10.2 DISADVANTAGES

10.2.1 **Internet**: It would require an internet connection for the working of the website

10.2.2 **Device Intensive:** It would require a device that could connect to the internet for it to perform properly

11. **CONCLUSION**

Vast is an easy-to-use web application that can be used by donors to donate their blood to blood banks. Hospitals in need for blood plasma can request blood from blood banks. This application should have a wider reach to appeal to more potential blood donors. In case of emergency SOS, donation of blood can be done in minuscule time.

12. **FUTURE SCOPE**

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

When such scenarios arise next time in the form of any other virus or biological crisis where a person needs to be cured by using plasma, our VAST application can be used and it would be very helpful to have a user friendly application.
Also in future VAST can be developed as a software app.

13. **APPENDIX**

Source Code
https://github.com/IBM-EPBL/IBM-Project-13968-1659537802/tree/main/Project%20Development%20Phase

GitHub & Project Demo Link
https://github.com/IBM-EPBL/IBM-Project-13968-1659537802