**Final Report**

**PLASMA DONOR APPLICATION**

**Team Members:**

**Gokul R**

**DanielVijayaraj P**

**Mukesshkumar G**

**Jayaprakash M**

**Project Report Format**

1. **INTRODUCTION**
   1. Project Overview
   2. Purpose
2. **LITERATURE SURVEY**
   1. Existing problem
   2. References
   3. Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
   1. Empathy Map Canvas
   2. Ideation & Brainstorming
   3. Proposed Solution
   4. Problem Solution fit
4. **REQUIREMENT ANALYSIS**
   1. Functional requirement
   2. Non-Functional requirements
5. **PROJECT DESIGN**
   1. Data Flow Diagrams
   2. Solution & Technical Architecture
   3. User Stories
6. **PROJECT PLANNING & SCHEDULING**
   1. Sprint Planning & Estimation
   2. Sprint Delivery Schedule
   3. Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
   1. Layer between Donor and Recipient
   2. Chat Bot Integration
   3. SendGrid Output
   4. Database Schema (if Applicable)
8. **TESTING**
   1. Test Cases
   2. User Acceptance Testing
9. **RESULTS**
   1. Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**

Source Code

GitHub & Project Demo Link

# INTRODUCTION

## Project Overview:

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.

## Purpose:

The main purpose of the proposed system, the donor can donate the plasma to the blood bank, the blood bank can apply for the donor and once the donor has accepted the request, the blood bank can add the units they need and the admin can also send the request to the blood bank that urgently needs the plasma for the patient and can take the plasma from the blood bank.

# LITERATURE SURVEY

## Existing problem

There are many people who are willing to donate plasma and who need plasma.

But there is not any accessible way to help them to find plasma donation centers in real- time. So, the problem is not the lack of donors, but finding the right sponsor at the right time. If someone needs plasma, they seek plasma first from family members, then from hospitals and the nearest plasma bank. If they can't process plasma in these ways, it's very difficult for them to contact another for a short-term plasma draw. This is a problem that Iwant to solve through this application. Instead of just providing plasma to people in need with an outdated list of regular plasma donors who may or may not be available to help, This application reaches the right people the moment users find Out.

## References

1. We have referred from researchers who have previously researched on this topic and literature
2. surveys are,
3. 1. https://onlinelibrary.wiley.com/doi/full/10.1111/j.1423-0410.2010.01360.x
4. 2. R. Kumar, R. Kumar and M. Tyagi, "Web Based Online Blood Donation System," 2021
5. 3rd International Conference on Advances in Computing, Communication Control and
6. Networking (ICAC3N), 2021, pp. 1630-1632, doi: 10.1109/ICAC3N53548.2021.9725558.
7. 3. P. U. Pratyusha, K. Chaitanya, A. Saranam, K. Manideep and S. Kranthi, "Smart
8. Intelligent Web based Online Blood Donation System," 2021 2nd International
9. Conference on
10. Smart Electronics and Communication (ICOSEC), 2021, pp. 1813-1819,
11. doi: 10.1109/ICOSEC51865.2021.9591811.
12. 4. M. Kaur et al., "A Web-based Blood Bank System for Managing Records of Donors and
13. Receipts," 2022 International Conference on Computational Intelligence and
14. Sustainable Engineering Solutions (CISES), 2022, pp. 459-464, doi:
15. 10.1109/CISES54857.2022.9844389.
16. 5. G. Dudani, Tanushree, K. Singh and A. Singh Chauhan, "A Systematic Review & Design of
17. Web-Based Blood Management System," 2021 3rd International Conference on
18. Advances in Computing, Communication Control and Networking (ICAC3N), 2021,
19. pp. 1896-1900, doi: 10.1109/ICAC3N53548.2021.9725479.
20. 6. R. S. Ali, T. F. Hafez, A. B. Ali and N. Abd-Alsabour, "Blood bag: A web application to
21. manage all blood donation and transfusion processes," 2017 International
22. Conference on Wireless Communications, Signal Processing and Networking
23. (WiSPNET), 2017, pp. 2125-2130, doi: 10.1109/WiSPNET.2017.8300136.
24. 7. E. H. Kaplan, "Evaluating plasma holds in the presence of multiple infections," in
25. Mathematical Medicine and Biology: A Journal of the IMA, vol. 18, no. 3, pp. 215-224,
26. Sept. 2001, doi: 10.1093/imammb/18.3.215.
27. 8. M. Arif, S. Sreevas, K. Nafseer and R. Rahul, "Automated online Blood bank
28. database," 2012 Annual IEEE India Conference (INDICON), 2012, pp. 012-017, doi:

## 10.1109/INDCON.2012.6420581.

* 1. **Problem Statement Definition**

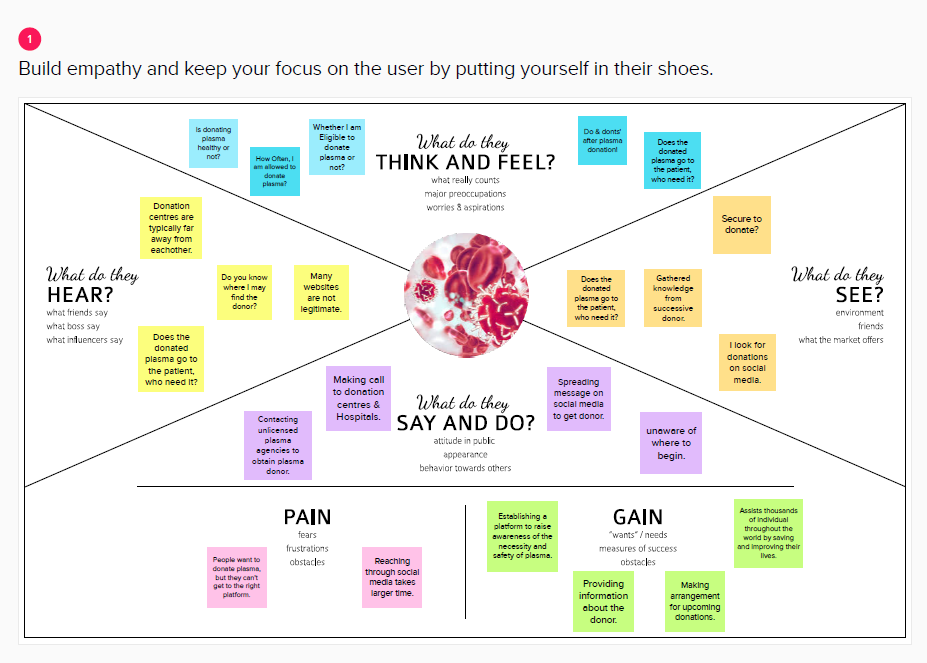
This system aims at connecting the donors & the patients by an online application. By using this application, the users can either raise a request for plasma donation or requirement. Similar to blood donors there also exist plasma donors where there exists problems like in case of emergency needs the most important life saver necessity is plasma.

# IDEATION & PROPOSED SOLUTION

## Empathy Map Canvas

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

**Empathy Map:**

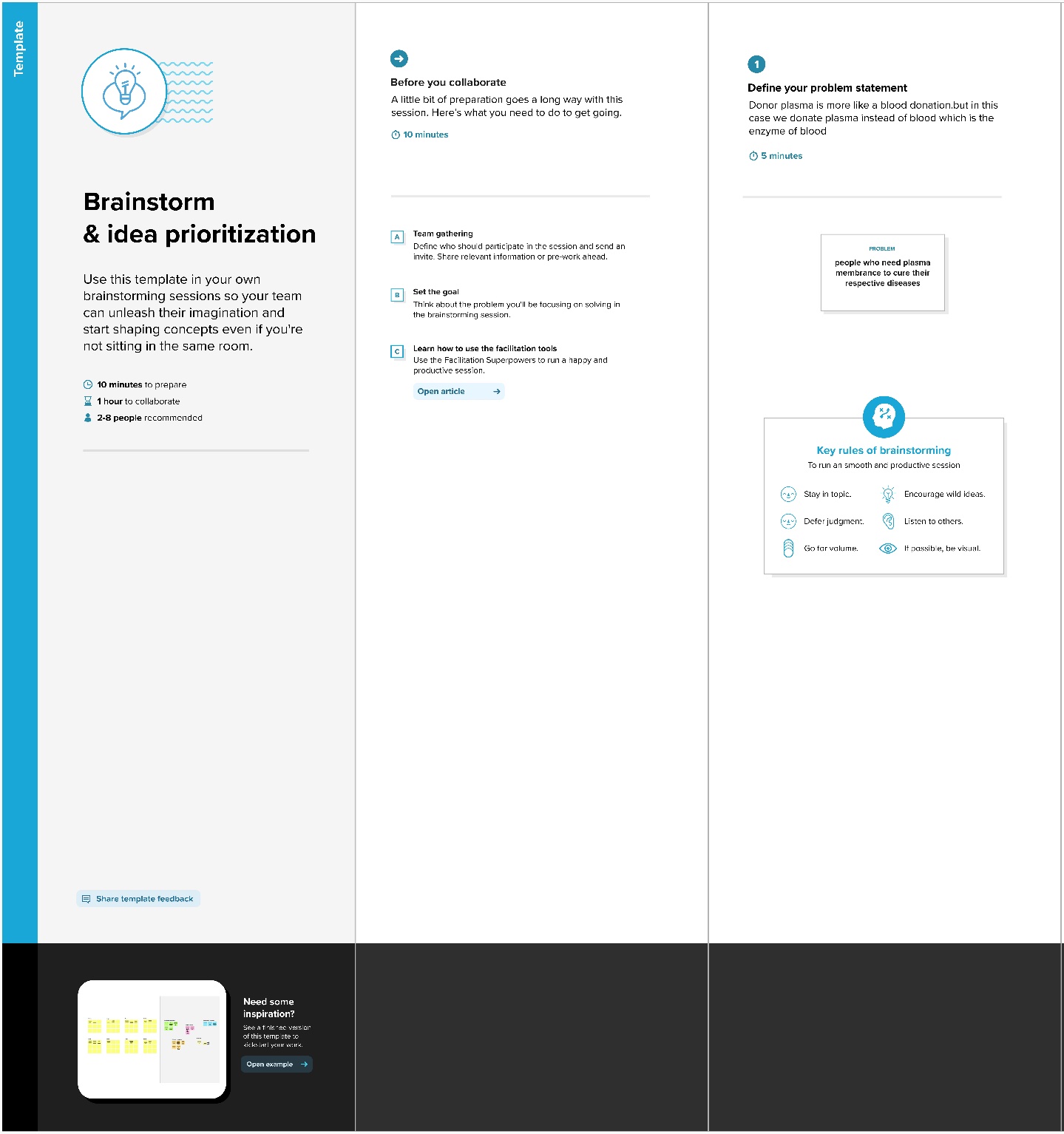
****

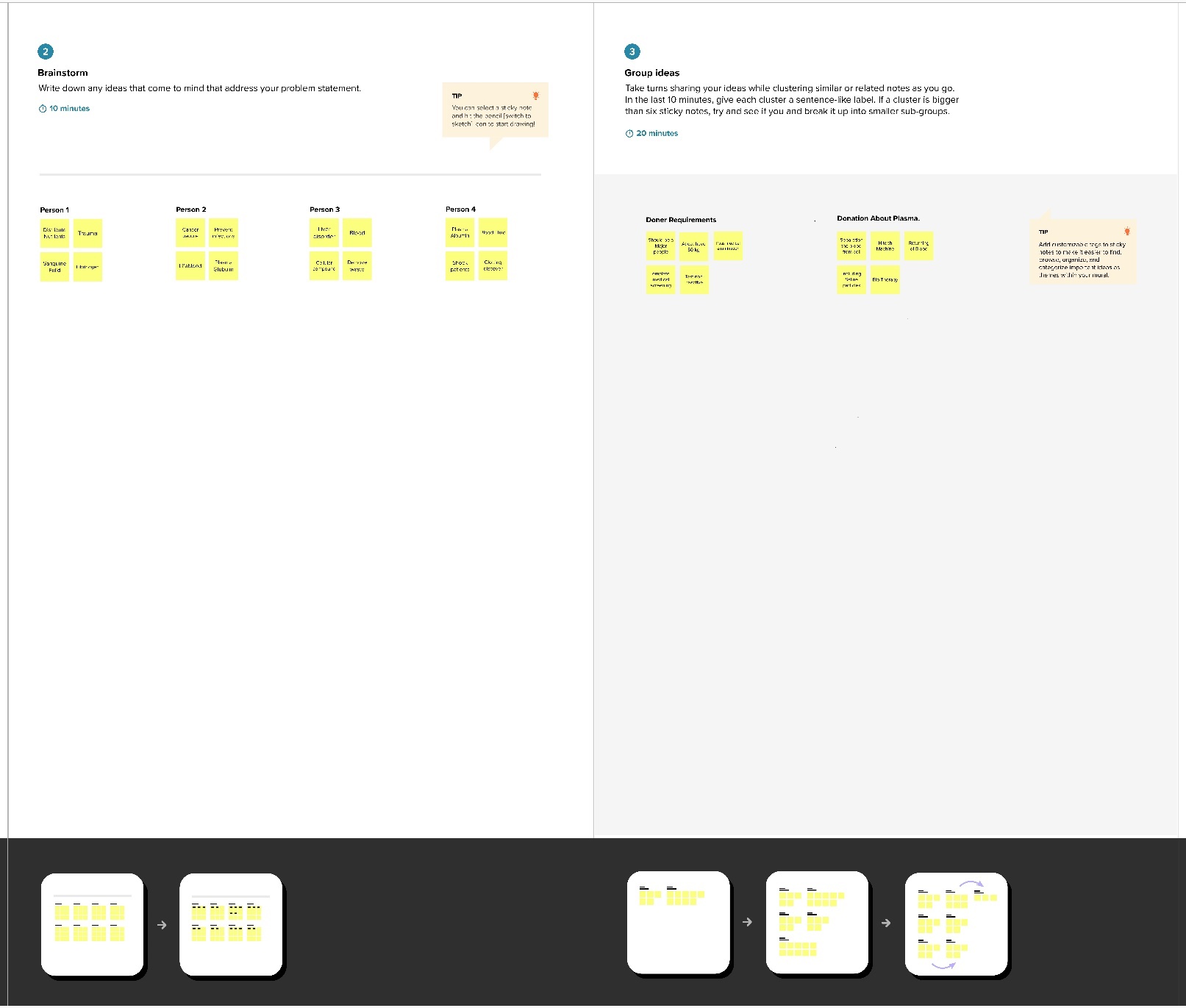
## Ideation & Brainstorming

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich number of creative solutions.

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

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

****



**Step-2: Brainstorm, Idea Listing and Grouping**

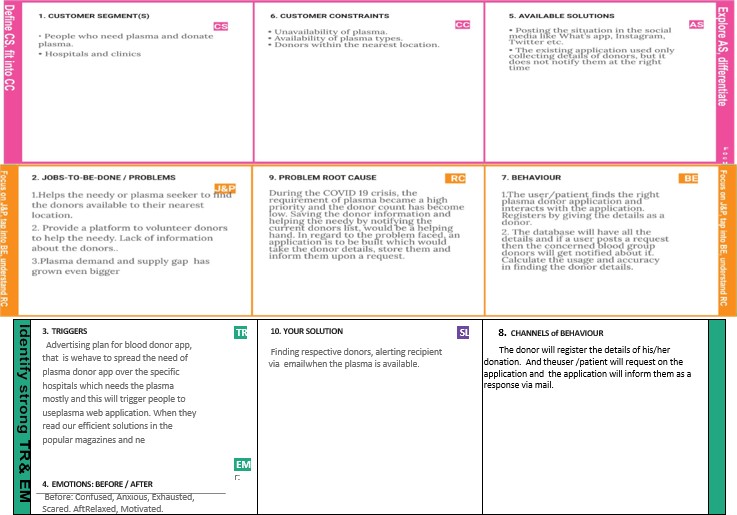
**Step-3: Idea Prioritization**

****

## Proposed Solution

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Parameter** | **Description** |
| 1. | Problem Statement | A web application which uses cloud  infrastructure to store the details of the donors and the location of the plasma available for donation. This Application aims to reduce the time to find the correct donors and their location. This stores the details of the donors so that when a request is made, we could easily track down the donor. This is proposed to be a web application which can be viewed on both mobile devices as well as computers |
| 2. | Idea / Solution description | The application is providing the facility to approach nearby plasma donors so that it will become much easier to search rare types in the hour of need. |

|  |  |  |
| --- | --- | --- |
| 3. | Uniqueness | A User Interface is simple for users to  understand. We can use the application anywhere anytime. The user immediately need the plasma for their treatment but the plasma is not available in nearby hospitals, then user can use this application to raise request and directly contact the donor , request them to donate the plasma. Hospitals can also raise request donors for donation. Somebody wants to donate blood and plasma but they don’t know the way to donate then they use this application which will simple to use and it will save lives of many people. Today many of them have mobile phones they can install this application and use it to save the lives of people. |
| 4. | Customer Satisfaction | Effect of donor motivation on donor satisfaction and loyalty is variable due to the influence of common donorship attitudes prevailing in donor population, impact of social marketing programs, focused on promotion of donor commitment and deliberate donorship. Thus, we have predicted that effect of donor motivation on donor relationship satisfaction and loyalty change |
| 5. | Business Model (Revenue Model) | This application is accessible by everyone . It is free. Because of the trouble in finding givers who match a specific blood bunch, this application empowers clients to enlist individuals who wish to give plasma and keep their data in a data set. Nowadays the need for plasma increases. Anyone with basic knowledge can access this app. This can be used anywhere anytime. working with the government we can utilize an application to help those needing plasma. |
| 6. | Scalability of the Solution | This application helps users to find plasma donors by sitting in home itself instead of searching donors everywhere. When there is a emergency then plasma request to send to everyone. Once the donor is ready to donate receiver is notified about donation. Receiver can contact the donor. With this app donor can know the eligibility to donate and making it easier to locate suitable donor at right time. |

* 1. **Problem Solution fit**

# REQUIREMENT ANALYSIS

## Functional requirement

Following are the functional requirements of the proposed solution.

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story / Sub-Task)** |
| FR-1 | Access Website | Should be able to access web applications on anydevice. |
| FR-2 | User Registration | Registration through Form  Registration through Gmail  Registration through Website  Registration through Application |
| FR-3 | User Login | Login through the registered email id. |
| FR-4 | Send Request | Users can request plasma and donors will be notified. |
| FR-5 | Contact donor | Donor details will be provided to the recipient. |

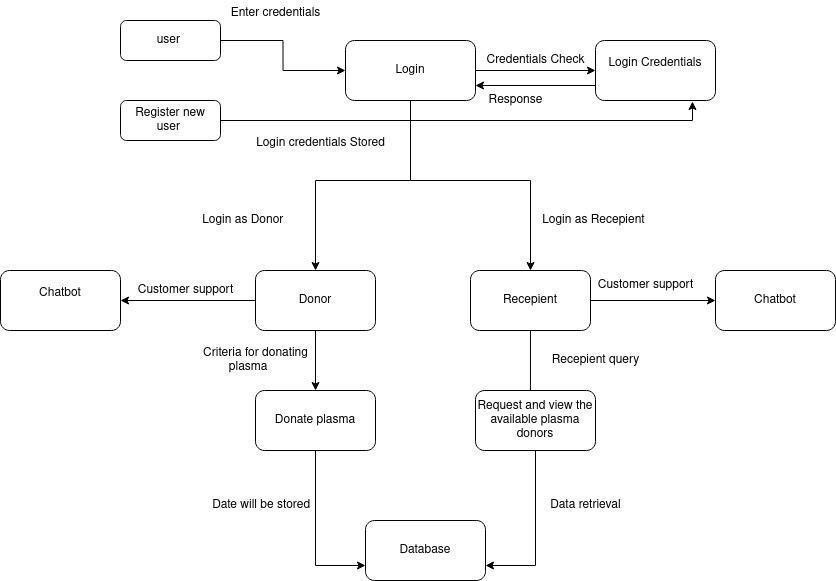
## Non-Functional requirements

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

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Non-Functional Requirement** | **Description** |
| NFR-1 | **Usability** | Plasma donor Application is very useful to the  emergency situation patient, because that  application gives the information of the nearby  plasma donors and request to donate their plasma  to patient via email , SMS etc. |
| NFR-2 | **Security** | Very secured website and application that  provides various security features like 2 step  verification , Email Verification , OTP password  etc.. |
| NFR-3 | **Reliability** | It gave the reliable information to the user , because  the register donors are well reliable person .So reliability is high. |
| NFR-4 | **Performance** | Carrying out an evaluation to quantify empirically  the recommendation abilities of two state-of-the-art  methods, considering different configurations,  within the proposed framework. |
| NFR-5 | **Availability** | The plasma donor application is an 24/7 online web application. |
| NFR-6 | **Scalability** | The application can be extended to provide plasma donor availability based on the recipient's location. |

* 1. **Data Flow Diagrams**

# PROJECT DESIGN



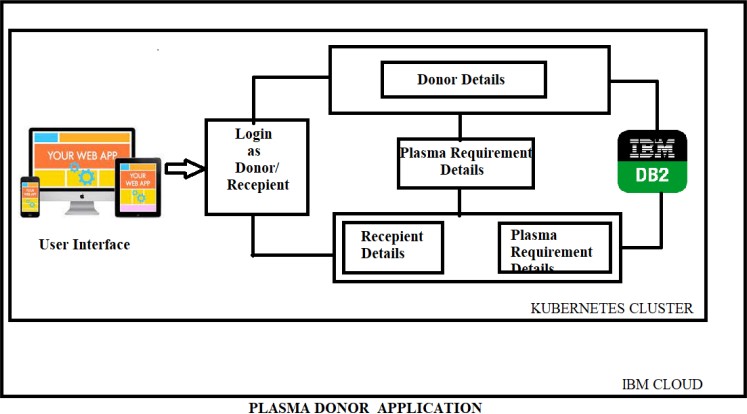
## Solution & Technical Architecture

**Solution Architecture:**

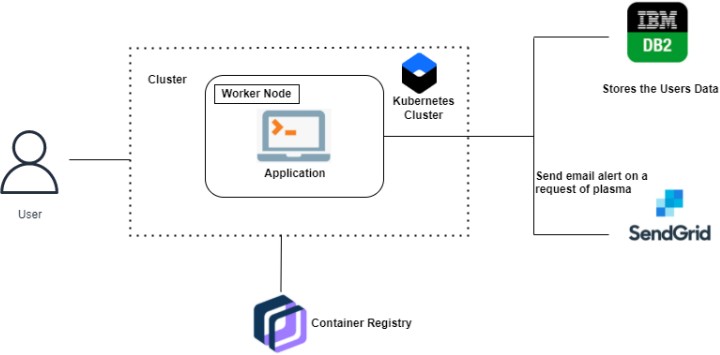
Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions.

Its goals are to:

* Find the best tech solution to solve existing business problems.
* Describe the structure, characteristics, behaviour, and other aspects of the software to project stakeholders.
* Define features, development phases, and solution requirements.
* Provide specifications according to which the solution is defined, managed, and delivered.



**Technology Architecture:**



**Table-1: Components & Technologies:**

|  |  |  |  |
| --- | --- | --- | --- |
| **SNO** | **Component** | **Description** | **Technology** |
| 1. | User Interface | How user interacts with application e.g.Web UI, Mobile App, Chatbot etc. | HTML, CSS, JavaScript |
| 2. | Application Logic | Logic for a process in the application | Python flask |
| 3. | Cloud Database | Database Service on Cloud | IBM DB2 |
| 4. | File Storage | File storage requirements | IBM Block Storage |
| 5. | External API | Trigger notification | Send grid |

**Table-2: Application Characteristics:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | Backend is built using flask and frontend usinghtml,Css,Javascript. | Html , Css , Javascript , Flask |
| 2. | Security Implementations | Database storage and access would be limited. | IBM DB2 |
| 3. | Scalable Architecture | Docker containers allows multiple containers to bedeployed at the same time. | Docker |
| 4. | Availability | Multiple kubernetes containers will be deployed. | Kubernetes |
| 5. | Performance | Backend would be able to handle multiple clients | Flask |

* 1. **User Stories**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **User Type** | **Functional**  **Requirement (Epic)** | **User Story Number** | **User Story / Task** | **Acceptance criteria** | **Priority** | **Release** |
| Customer (Mobile user) | Registration | USN-1 | As a user, I can register as donor or recipient by entering my email, password, and confirming  my password. | I can access my account / dashboard | High | Sprint-1 |
|  |  | USN-2 | As a user, I will receive confirmation email once I have registered for the  application | I can receive confirmation email | Medium | Sprint-4 |
| Donor | Donor’s  Login | USN-3 | As a donor, I can login into thedonor’s page | As a donor, I can access  into my  profile and donate plasma | High | Sprint-1 |
| Recipient | Recipient’s  Login | USN-4 | As a recipient, I can login into recipientpage | As a recipient, I can access into my profile and  request  plasma | High | Sprint-1 |
| Chat bot | Dashboard | USN-5 | For the customer convenience, There is a chat bot for  the queries | I can get the related queries from the bot | Medium | Sprint-2 |
| Administrator | Administration | USN-6 | As an administrator I can enumerate theuserdata and manage  them | I can manage the data which provides data integrity to  the user | High | Sprint-3 |

# PROJECT PLANNING & SCHEDULING

## Sprint Planning & Estimation

**Project Tracker, Velocity & Burndown Chart:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **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 |

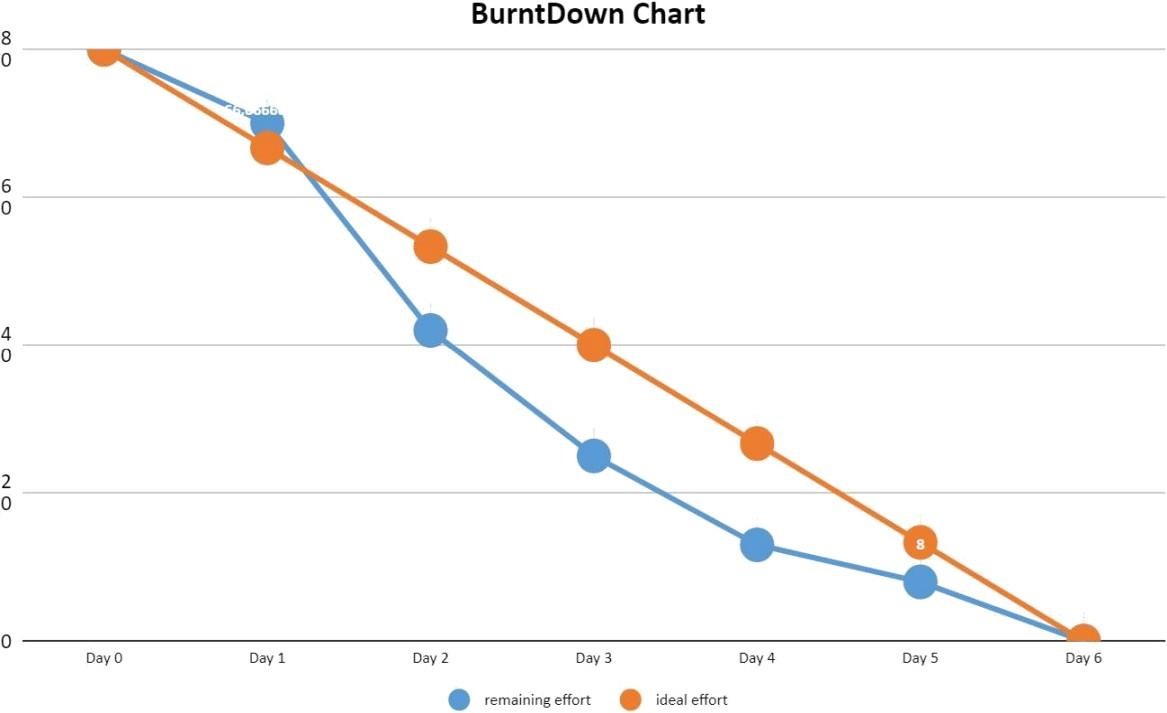
**Velocity:**

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) per iteration unit (story points per day) AV = Sprint Duration / Velocity

= 20/6 = 4

***Burndown Chart:***

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile [software development m](https://www.visual-paradigm.com/scrum/what-is-agile-software-development/)ethodologies such as [Scrum.](https://www.visual-paradigm.com/scrum/scrum-in-3-minutes/) However, burn down charts can be applied to any project containing measurable progress over time.

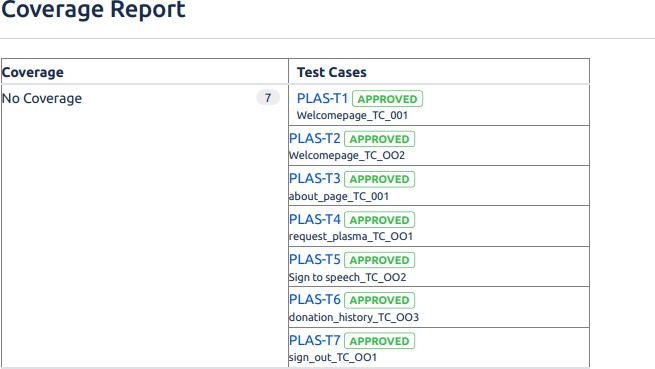


## Sprint Delivery Schedule

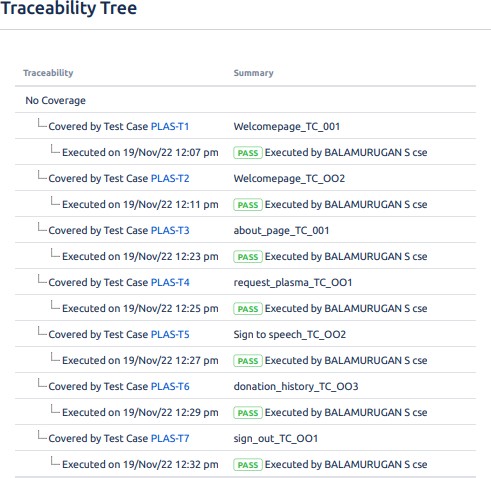
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Functional Requirement**  **(Epic)** | **User Story Number** | **User Story / Task** | **Story Points** | **Priority** | **Team Members** |
| Sprint-1 | Registration | USN-1 | As a user, I can register as donor or recipient by entering my email, password and confirming  my password | 10 | High | Gokul |
| Sprint-4 | Registration | USN-2 | As a user, I will receive confirmation email once I have registered for the  application | 20 | Medium | Hariprasad P |
| Sprint-1 | Donor’s Login | USN-3 | As a donor, I can login into the  donor’spage | 5 | High | Balamurugan S |
| Sprint-1 | Recipient’s  Login | USN-4 | As a recipient, I can login  into recipient  page | 5 | High | Dharanish V |
| Sprint-1 | Recipient’s  Login | USN-4 | As a recipient, I can login  into recipient  page | 5 | High | Dharanish V |
| Sprint-3 | Administration | USN-6 | As an administrator I can enumerate the user data and manage  them | 20 | High | Hariprasad P |

* 1. **Reports from JIRA**

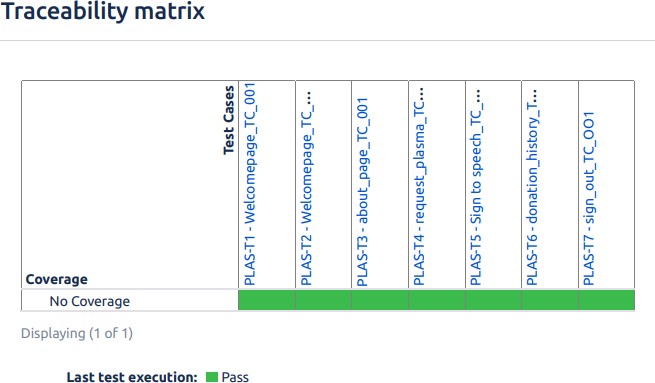
## COVERAGE REPORT

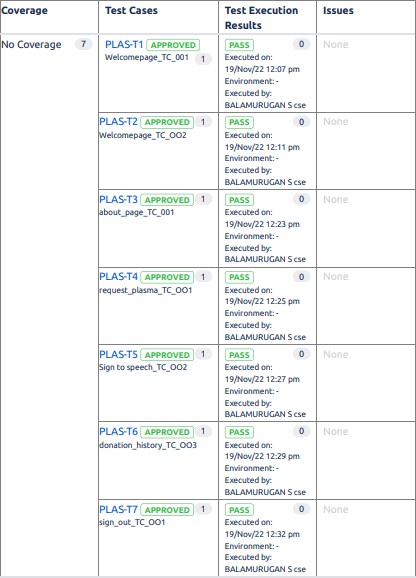


**Traceability Tree Report**



**Traceability Matrix Report**



**Traceability Report**

# CODING & SOLUTIONING

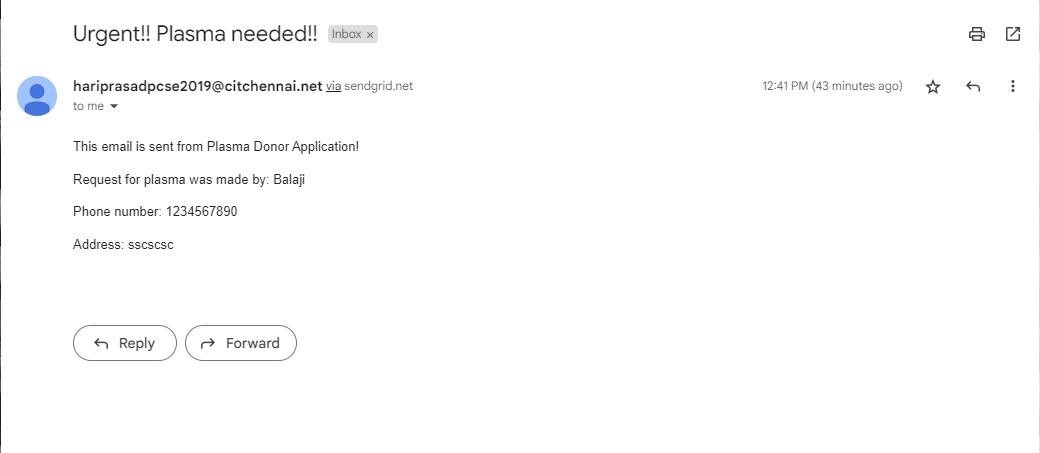
## Layer between Donor and Recipient

The application servers as a layer between donor and the recipient, donor can expose their information to the application, and they will be notified via email if there’s a request for plasma with the same blood group

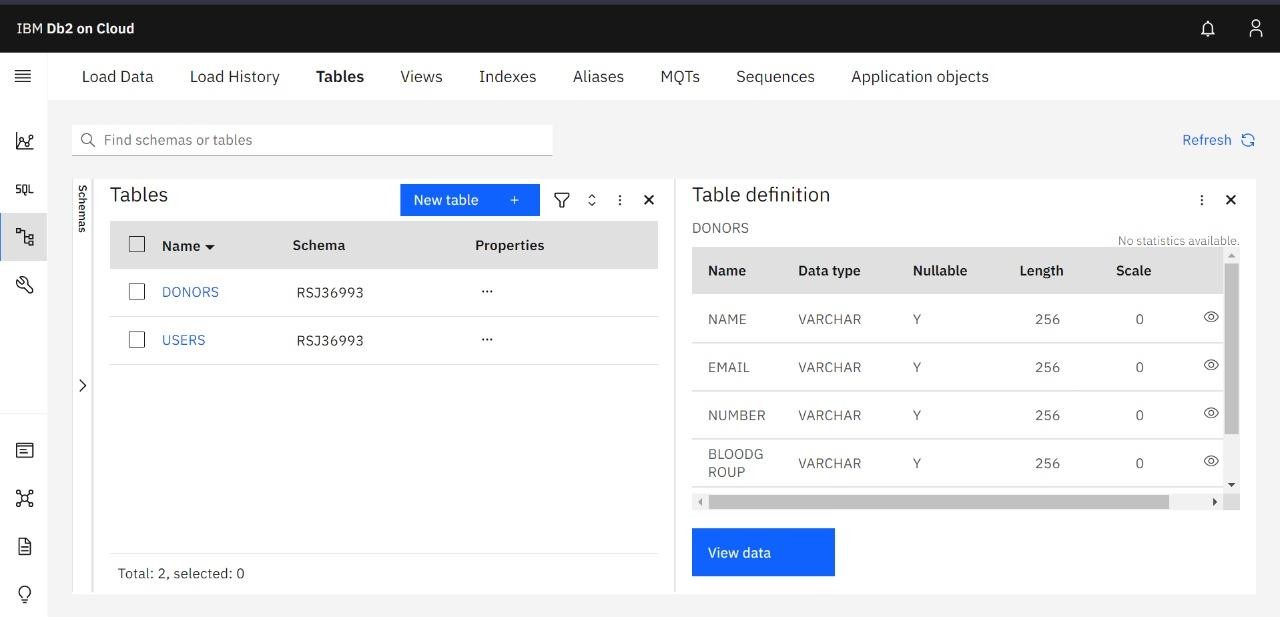
## Chat bot integration

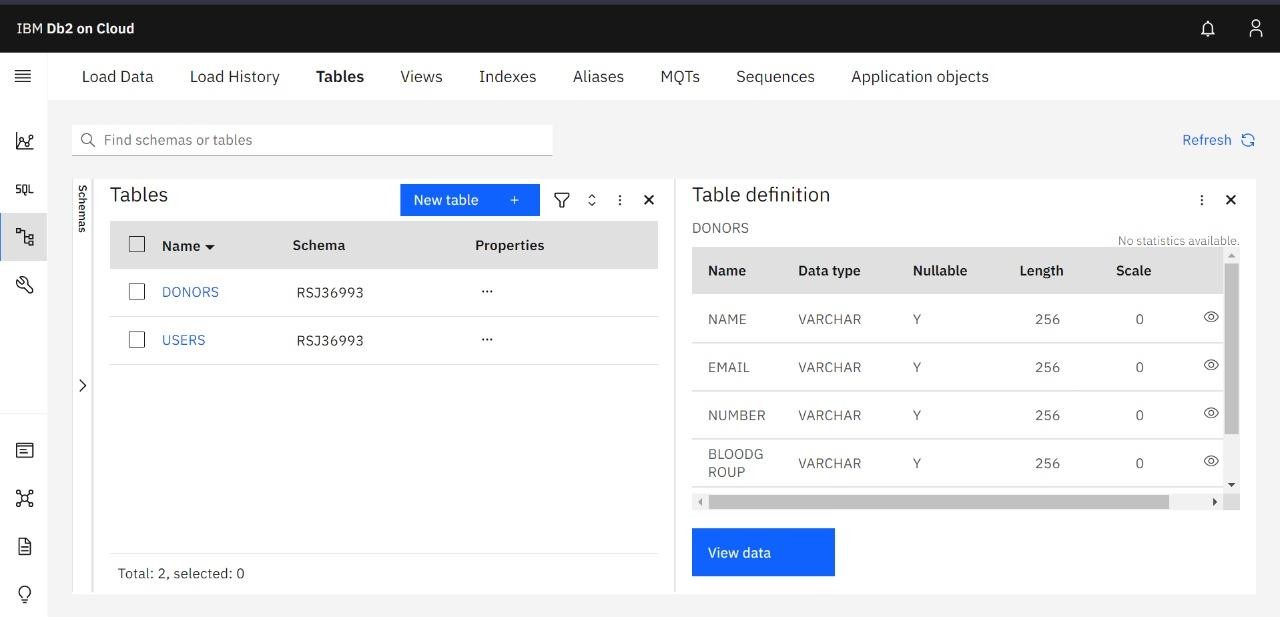
The application has a chatbot integrated with it to help with basic user queries and to interact with the user. The chatbot feature is added to the application by using IBM Watson assistant in IBM cloud. This chatbot can interact with the user and guide them for simple queries.

## SendGrid output



* 1. **Database Schema**





# .

* 1. **Test Cases**

# TESTING

|  |
| --- |
| * Verify if the buttons in web page are responsive |
| * Verify if the UI elements are getting displayed properly |
| * Verify if the user can upload files from his system |
| * Verify if the output is displayed |
| * Verify if the user can login using his credentials |
| * Verify if the model predicts the input accurately |
| * Verify if the user is getting redirected to home page after sign in |
| * Verify if the UI elements are being displayed |
| * Verify if the user can navigate to other pages in navigation bar |
| * Verify if the user can exit the home to sign page |

## User Acceptance Testing

**Purpose of Document**

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

**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 | 2 | 0 | 0 | 0 | 2 |
| Duplicate | 1 | 0 | 0 | 0 | 1 |
| External | 0 | 0 | 0 | 0 | 0 |
| Fixed | 3 | 0 | 0 | 0 | 3 |
| Not Reproduced | 2 | 0 | 0 | 0 | 2 |
| Skipped | 0 | 0 | 0 | 0 | 0 |
| Won't Fix | 0 | 0 | 0 | 0 | 0 |
| Totals | 8 | 0 | 0 | 0 | 8 |

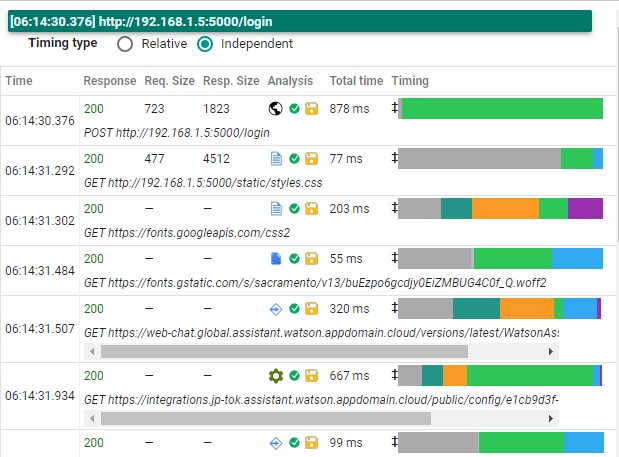
**Test Case Analysis**

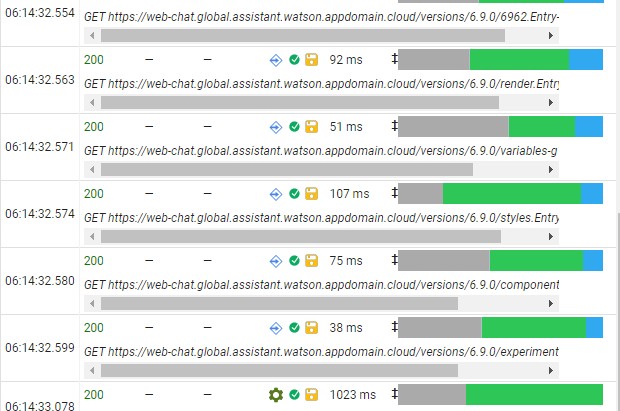
This report shows the number of test cases that have passed, failed, and untested

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Section** | **Total Cases** | **Not Tested** | **Fail** | **Pass** |
| Print Engine | 0 | 0 | 0 | 0 |
| Client Application | 3 | 0 | 0 | 3 |
| Security | 0 | 0 | 0 | 0 |
| Outsource Shipping | 0 | 0 | 0 | 0 |
| Exception Reporting | 0 | 0 | 0 | 0 |
| Final Report Output | 0 | 0 | 0 | 0 |
| Version Control | 0 | 0 | 0 | 0 |

* 1. **Performance Metrics**

# Results





**Advantages:**

# Advantages and Disadvantages

* + - Easy user interaction.
    - Faster loading of news.
    - Easy login process.
    - Servers an layer between donor and recipient

**Disadvantages:**

* + - Reports are not verified.
    - Wrong information will affect the output
    - No authentication.

# Conclusion

Although the government is conducting large-scale Covid immunisation efforts, the volume of vaccines produced is insufficient to vaccinate the whole population at this time. With the number of corona positive cases increasing by the day, preserving lives has become the top priority. According to WHO estimates, more than 3 million individuals have died as a result of the coronavirus. Aside from immunisation, there is another scientific approach for treating a covid infected individual and lowering the chance of mortality. This plasma treatment is an experimental strategy to treating and recovering corona-positive individuals. This plasma treatment is thought to be both safe and promising. This plasma therapy is considered to be safe & promising. A person who has recovered from Covid can donate his/her plasma to a person who is infected with the coronavirus. This technique suggested here tries to connect donors and patients using an internet application. Users can use this application to make a request for plasma donation or a necessity. Both parties have the option to accept or reject the request. To donate plasma, the user must provide a Covid Negative report. If somebody need a Plasma Donor, this system is employed. Blood and plasma donation is a type of citizen's social duty in which a person can voluntarily donate blood/plasma using our app. This application was built with the idea of ensuring that donors contribute blood/plasma to the community. This approach is designed to be user-friendly so that anybody may access and manage his or her account. This application will disrupt the blood/plasma supply chain and assist the poor in finding free donors. This project will assist new blood and plasma banks in improving their services and transitioning from traditional to user-friendly frameworks.

# Future Scope

The following features can be added in the application in the future:

* + - To add location of the donor on request
    - Implement industry standards Oauth protocols
    - Requesting donor within the neighbouring location

# Appendix

## Source Code App.py:

from *from* flask *import* Flask, render\_template, request, redirect, url\_for, session

*from* flask\_mail *import* Mail, Message

*import* ibm\_db

*import* os

conn = ibm\_db.connect("DATABASE=bludb; HOSTNAME=b1bc1829-6f45-4cd4-bef4-10cf081900bf.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32304;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=nrv74690;PWD=tTMbaQMjnEHnpUz6",'','')

app = Flask( name ) app.config['SECRET\_KEY'] = 'top-secret!'

app.config['MAIL\_SERVER'] = 'smtp.sendgrid.net'

app.config['MAIL\_PORT'] = 587 app.config['MAIL\_USE\_TLS'] = True app.config['MAIL\_USERNAME'] = 'apikey'

app.config['MAIL\_PASSWORD'] = os.environ.get('SENDGRID\_API\_KEY') app.config['MAIL\_DEFAULT\_SENDER'] = os.environ.get('MAIL\_DEFAULT\_SENDER') mail = Mail(app)

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

print(os.environ.get('MAIL\_DEFAULT\_SENDER')) if 'email' not in session:

return redirect(url\_for('login'))

return render\_template('home.html',name='Home')

@app.route("/about",methods=['GET'])

## GitHub Link:

[**https://github.com/IBM-EPBL/IBM-Project-45949-1660733660**](https://github.com/IBM-EPBL/IBM-Project-45949-1660733660)

**Project Demo Link:**

## https://drive.google.com/drive/folders/1on7u7U3EA8bTLSmxmOEZHKMxTMrVdJcS?usp=share\_link