# PLASMA DONAR APLLICATION

## NALAIYA THIRAN PROJECT BASED LEARNING ON PROFESSIONAL READLINESS FOR INNOVATION, EMPLOYNMENT AND ENTERPRENEURSHIP

## PROJECT REPORT

**BHUVANSESH S**

**BRAJESH KUMAR V**

**CHIDAMBARAM V**

**DEEPAK B**


**BACHELOR OF ENGINEERING**

**IN**

**ELECTRONICS AND COMMUNICATION ENGINEERING**

**SAVEETHA ENGINEERING COLLEGE**

**CHENNAI-602105**

**NOVEMBER 2022**

# BONAFIDE CERTIFICATE

Certified that this project report titled **“PLASMA DONOR APPLICATION by NALAIYA THIRAN PROJECT BASED LEARNING Program”,** is the Bonafide work **BHUVANSESH S (212219060045) BRAJESH KUMAR V (212219060049), CHIDAMBARAM V (212219060058), DEEPAK B(212219060062)** who carried out the work under faculty mentor and industry mentor supervision, for the partial fulfillment of the requirements for the award of the degree of **BACHELOR OF ENGINEERING IN ELECTRONICS AND COMMUNICATION ENGINEERING**.

Certified further that to the best of my knowledge and belief, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion.

# ABSTRACT

The system will provide a complete technical solution to the Plasma Donation through Website. The main goal of our project is to design a user-friendly web application that is like a scientific vehicle from which we can help reduce mortality or help those affected by corona by donating plasma. The basic constructs of table spaces" clusters and induces have been exploited to provide higher consistency and reliability for the data storage. The database connectivity was planned using the latest SQL Connection. Plasma therapy is an experimental approach to treat those COVID-positive patients and help them recover faster. The project has been planned to be having the view of distributed architecture with centralized storage of the database. The entire project has been developed keeping in view of the distributed client server computing technology in mind.

# INDEX

**PAGE NO:**

# 1. INTRODUCTION

## 1.1 PROJECT OVERVIEW

Category: Cloud App Development

Team ID: PNT2022TMID03517

Skills Required: IBM Cloud, HTML, Java script, IBM Cloud Object Storage, Python Flask, Kubernetes, Docker, IBM DB2, IBM Container Registry.

The government is carrying out Covid vaccination campaigns. Apart from vaccination, there is another scientific method by which a covid infected person can be treated and the death risk can be reduced. This plasma therapy is an experimental approach to treat corona-positive patients and help them recover. The Blood Donation Agent is to create an Information about the donor and organization that are related. Through this app any person who is interested in donating the blood can register himself. Organization wants to register itself with this site that can also register.

This plasma therapy is considered to be safe. A person who has recovered from Corona can donate their plasma to a person who is infected with the corona virus. If any general consumer wants to make request blood online he can also take the help of this site. Admin who can do addition, deletion and modification if required.

## 1.2 PURPOSE

The main goal of our project is to design a user-friendly web application that is like a scientific vehicle from which we can help reduce mortality or help those affected by corona by donating plasma. The Blood Donation Agent is to create an Information about the donor and organization that are related to donating the blood. Through this application any person who is interested in donating the blood can register himself.

Plasma therapy is an experimental approach to treat those COVID-positive patients and help them recover faster. The project has been planned to be

having the view of distributed architecture with centralized storage of the database. The entire project has been developed keeping in view of the distributed client server computing technology in mind.

The basic constructs of table spaces" clusters and induces have been exploited to provide higher consistency and reliability for the data storage. The database connectivity was planned using the latest SQL Connection. The authentication and authorization was cross checked at all the relevant stages. If a particular person has fully recovered from Corona, they are eligible to donate their plasma.

# 2. LITERATURE SURVEY

## 2.1 EXISTING PROBLEM

## Introduction

Applying optimization methods to healthcare management and logistics is a developing research area with numerous studies. There are two types of process in the existing system: the blood donation process and the blood request process. Facility location, staff rostering, patient allocation, and medical supply transportation are the main themes analysed. When a new donor comes to donate blood, they are required to fill out their personal information during the registration process before making a donation.

After the donation, the donor is given a donor identification card with their name, blood type and a barcode to be used as a reference for future donations. The barcode is used to retrieve the donor's record containing their personal information, medical history and donation information, including blood results Only blood bank administrators have the authority to access the donor's records. This makes it difficult for donors to make changes to their personal information within the system. Having a donor ID card may be a tangible reminder to people that they are helping lives as a blood donor; however, possessing a physical card comes with drawbacks such as loss or damage.  is necessary for several treatments and surgeries, and still a limited resource.

Blood is classified into groups (A and subgroups, B, 0 or AB) and based on the Rhesus factor (Rh+ or Rh-), and each donor should be correctly matched with the patient who receives his/her blood. Moreover, as it may transmit diseases, blood must be screened before utilization.

## 2.2 REFERENCES

| S.NO | TITLE | AUTHORS | ABSTRACT | DRAWBACKS |
|---|---|---|---|---|
| 1 | Blood Bank Management Information System in India | Vikas Kulshreshtha Research Scholar, Dr. Sharad Maheshwari, Associate Professor | A blood bank is a bank of blood or blood components, gathered as a result of blood donation, stored and preserved for later use in blood transfusion. To provide web based communication there are numbers of online web based blood bank management system exists for communicating between department of blood centers and hospitals, to satisfy blood necessity, to buy, sale and stock the blood, to give information about this blood. Introduces the review of the main features, merits and demerits provided by the existing Web -Based Information System for Blood Banks. This study shows the comparison of various existing system and provide some more idea for improve the existing system. | • It does not provide the better inventory solution to the end user<br>• Requires an active internet connection |
| 2 | A Web Based Blood Bank System For Managing Records of Donors and Recipients | Manvir Kaur, Nahida Nazir, Navneet Kaur, Syed Faraz Ali, Chirag Agarwal, Ujjwal Dubey, Varun Gupta, Abid Sarwar, Manik Rakhra, Omdev Dahiya. | The Online Blood Donation Management System, the purpose of which is to act as a bridge between a person who needs blood, a patient, and a blood donor. The design of an automatic blood system has become an integral part for saving the human lives, who need the blood under different situations. | There are various drawbacks of the pre-existing system like privacy issues for the donors, which are getting reflected directly on the interface. |
| 3 | Developing a plasma donor application using Function-asa-service in AWS | Aishwarya R, Gowri Jain University, Department of MCA, computer science. | A plasma is a liquid portion of the blood, over 55% of human blood is plasma. Plasma therapy is a process where blood is donated by recovered patients in order to establish antibodies that fight the infection. In this project plasma donor application is being developed by using AWS services. The services used are AWS Lambda, API gateway, DynamoDB, AWS Elastic Compute Cloud with the help of these AWS services. | • It requires an internet connection for the working of the website.<br>• Cant handle multiple requests at the same time |
| 4 | Web Based Online Blood Donation System "2021 3rd International Conference on Advances in Computing | Rohit Kumar, Rajan Kumar, Manik Tyagi | This paper depicts a high level program to close the hole between blood givers and individuals needing blood. The Online Blood donation Administration Framework application is an approach to synchronize blood donation centres with emergency clinics with the assistance of the Web. It is a web application where enlisted clinics can check the accessibility of the necessary Blood and can send a blood solicitation | • It requires an internet connection for the working of the website.<br>• Blood donation centre can likewise send a solicitation to another blood donation centre that isn't accessible. |

| | | | | |
|---|---|---|---|---|
| 5 | A Research Paper on Blood Donation Management System | Devanjan K. Srivastava, Utkarsh Tanwar, M.G.Krishna Rao, Priya Manohar, Balraj Singh, | Blood donation and transfusion has been an ever - serious issue and the shortage of blood throughout the world has caused many people to lose their life. The lack of a centralized system for blood donation is majorly responsible for those losses. Now in the era of online and digital processes, the conventional methods of collecting blood are absolute. We have designed a SQLite database as an integral part of the integrated framework to store historical blood donation data in a centralized database for analytical processing. The proposed system would enable people to register as a donor to make themselves available whenever in need of their blood type. | • It requires Internet Connection<br>• There is no proper centralized database for registered donors |

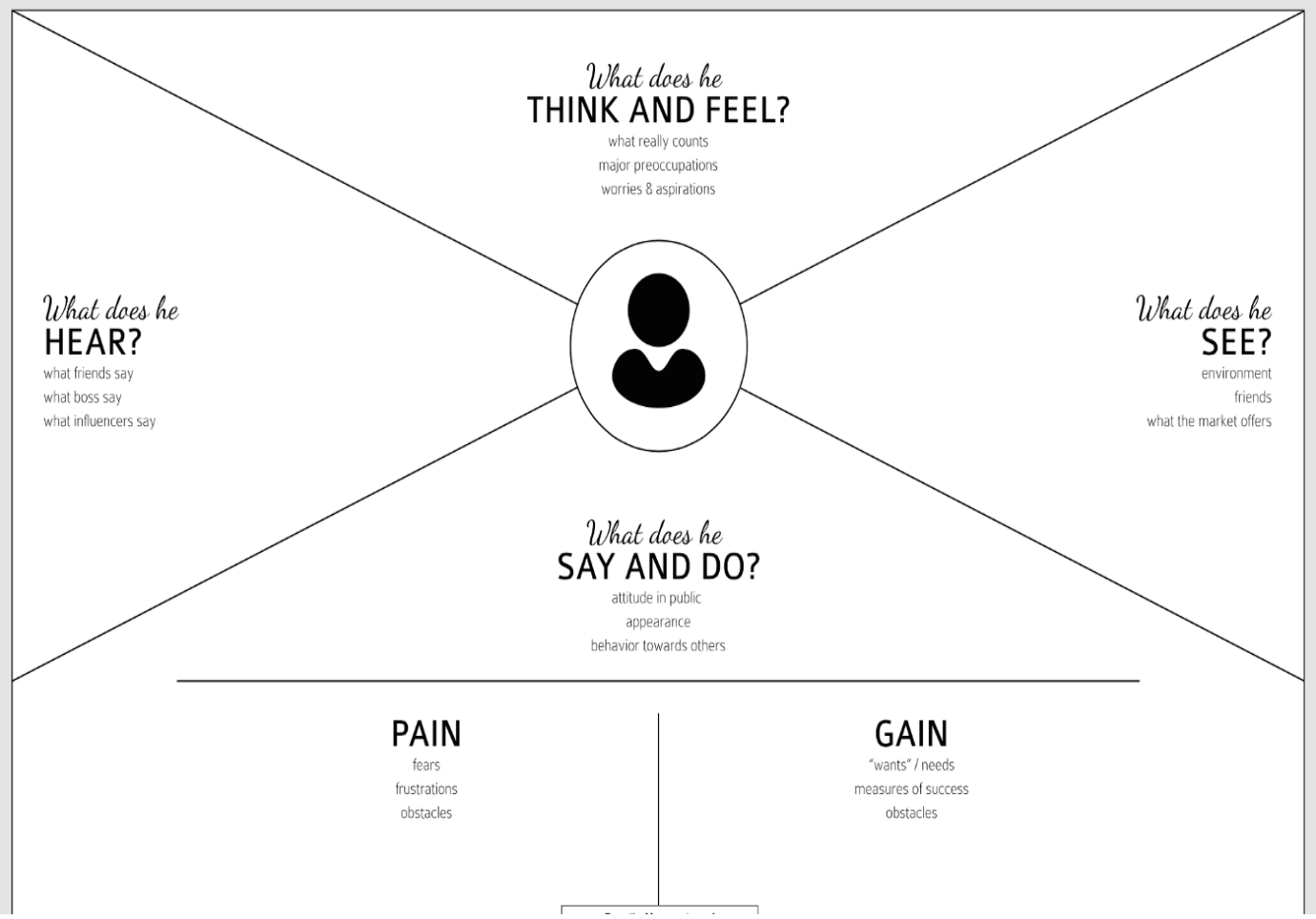
## 2.3 PROBLEM STATEMENT DEFINITION

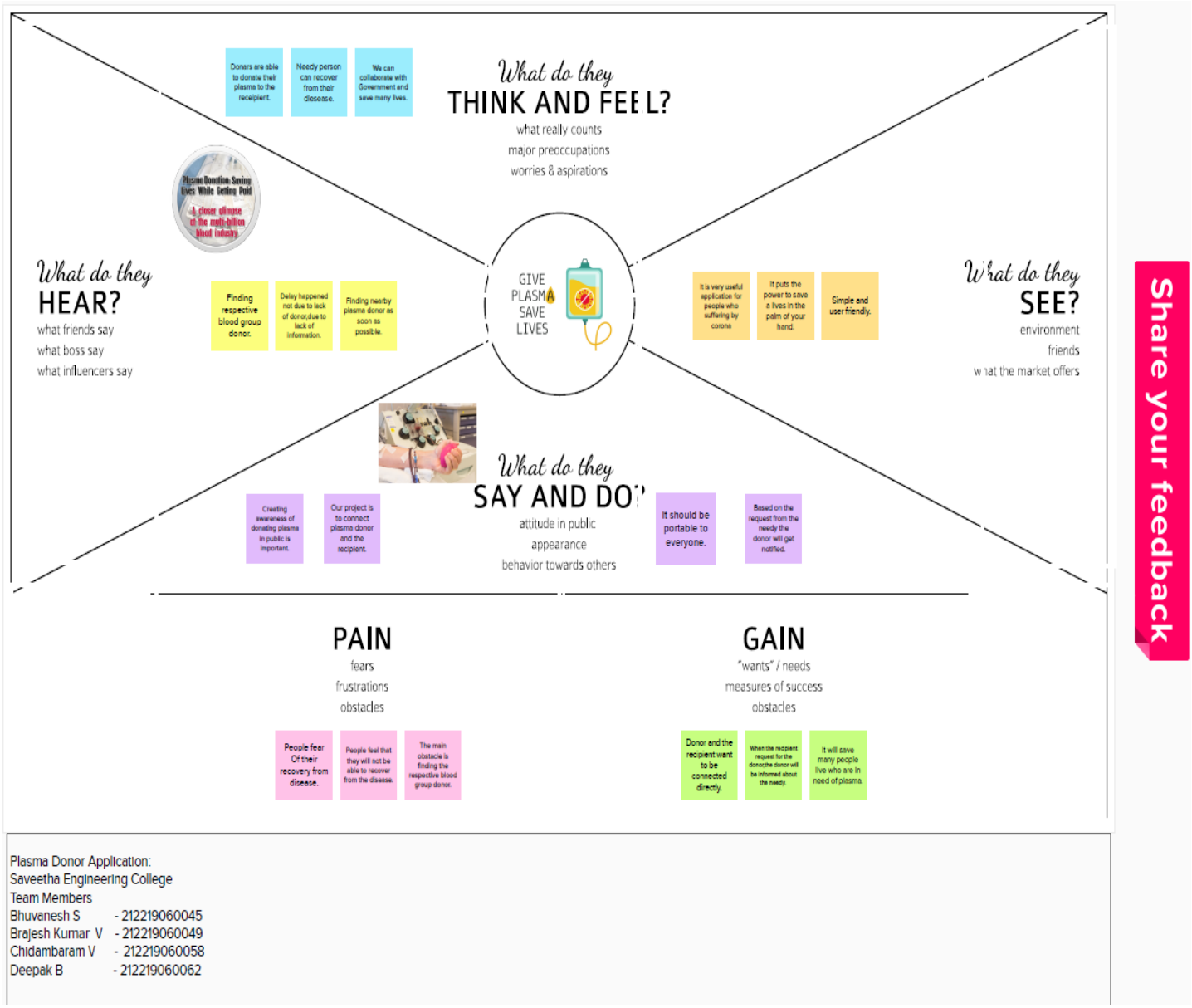
During the COVID 19 crisis, the requirement of plasma became a high priority and the donor count has become low. Plasma donation is a specific form of blood donation whereby plasma, the clear liquid component of whole blood, is collected during donation through a process known as plasmapheresis. During plasmapheresis, an apheresis machine removes and separates the plasma from the whole blood. As plasma contains nutrients, proteins, and antibodies, it can be used to develop treatments for diseases. Plasma used for these purposes is called convalescent plasma. 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. The proposed system implements a cloud based web application as a solution to this problem. The web application is implemented using Python flask and Docker software. The details of the donors are fetched using a form filling structure and later embedded with the cloud storage. The receiver of the plasma has to make a request for a particular blood group in the application itself. The applicable donor and receiver are connected anonymously and the blood is donated. The donor and receiver are connected through a unique id which keeps the identity of both donor and receiver hidden.

# 3. IDEATION & PROPOSED SOLUTION

## 3.1 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





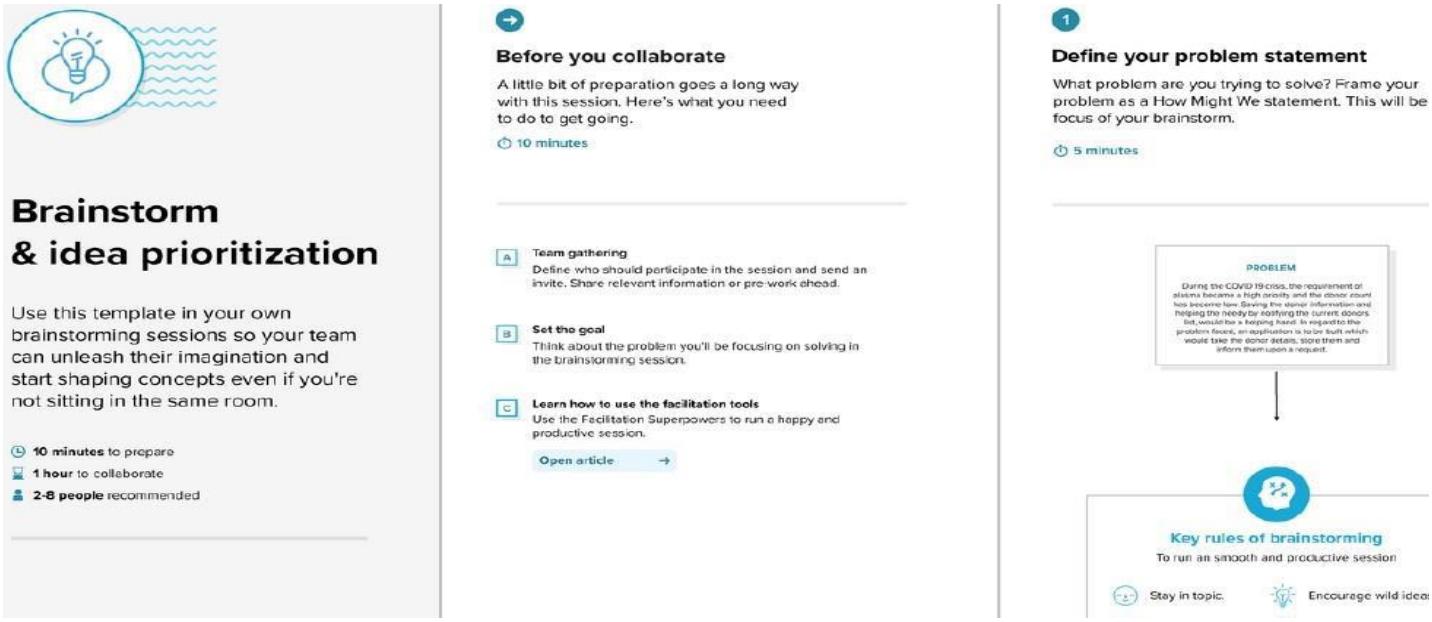

Share your feedback

Plasma Donor Application:
Saveetha Engineering College
Team Members
Bhuvanesh S - 212219060045
Brajesh Kumar V - 212219060049
Chidambaram V - 212219060058
Deepak B - 212219060062

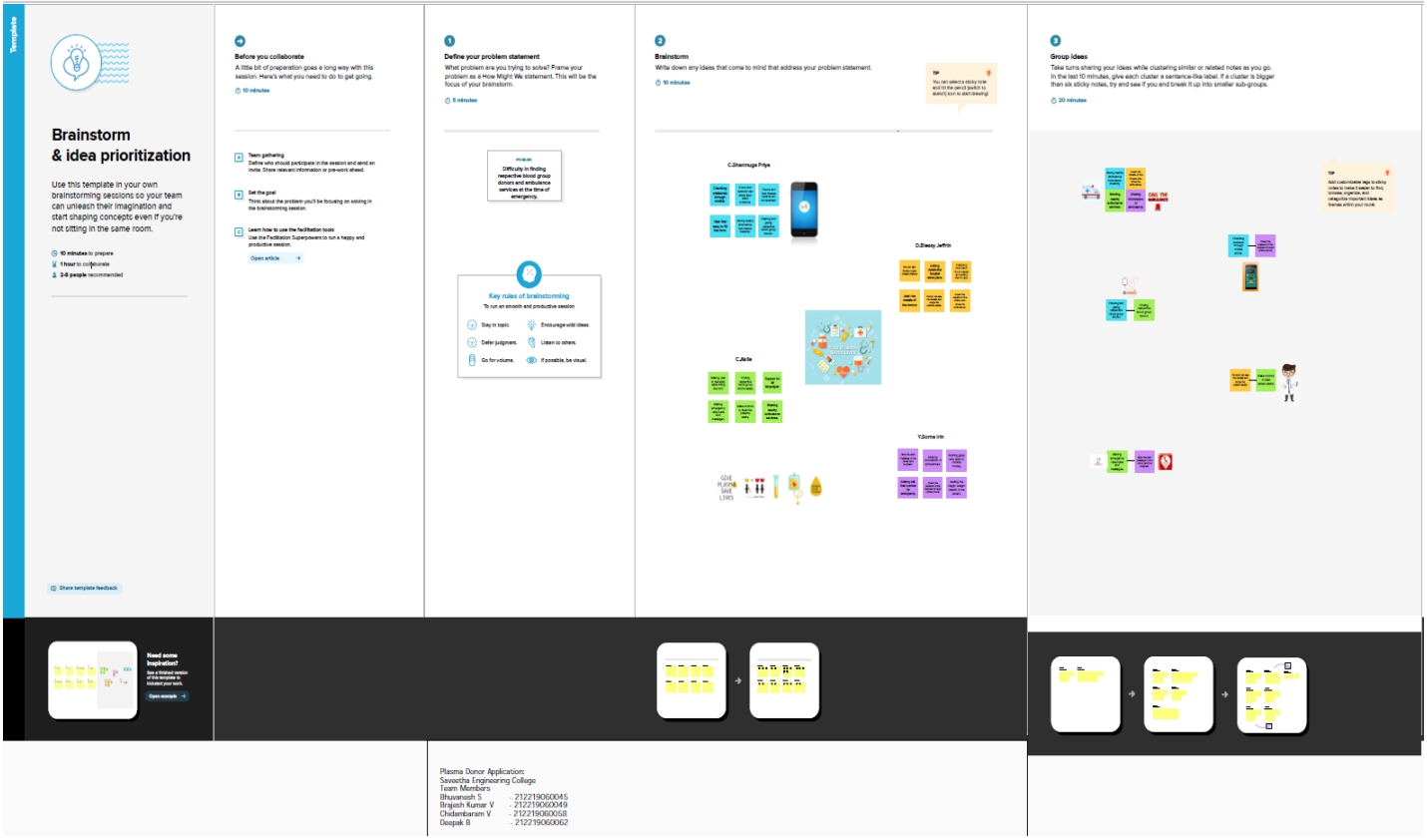
## 3.2 IDEATION & BRAINSTORMING

### Team Gathering Collaboration and Select the Problem Statement

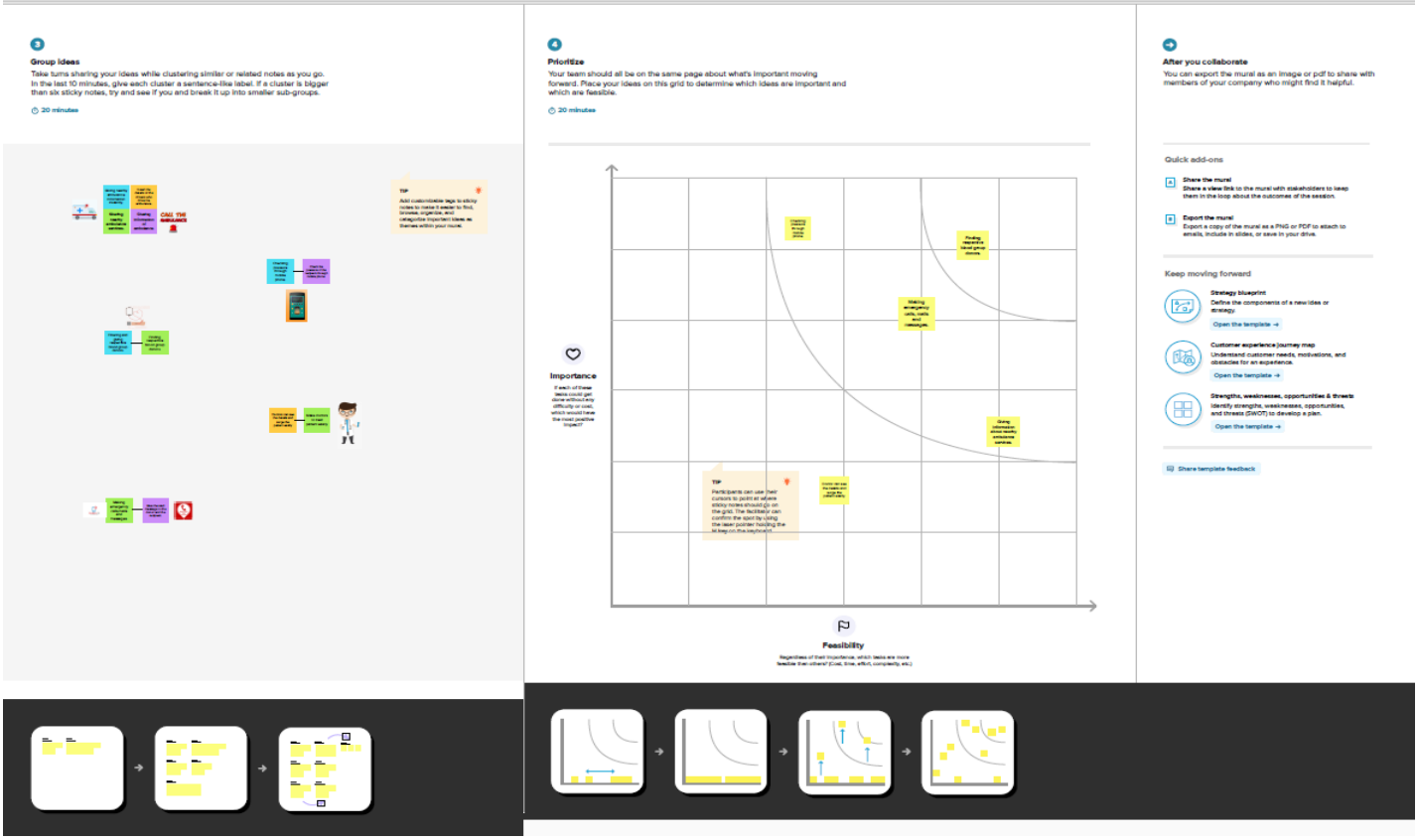
Brainstorm & Idea Prioritization Template: Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value,helping each other develop a rich amount of creative solutions.




Brain Storm, Idea Listing and Group ideas:



# Idea Prioritization




# Top Ideas

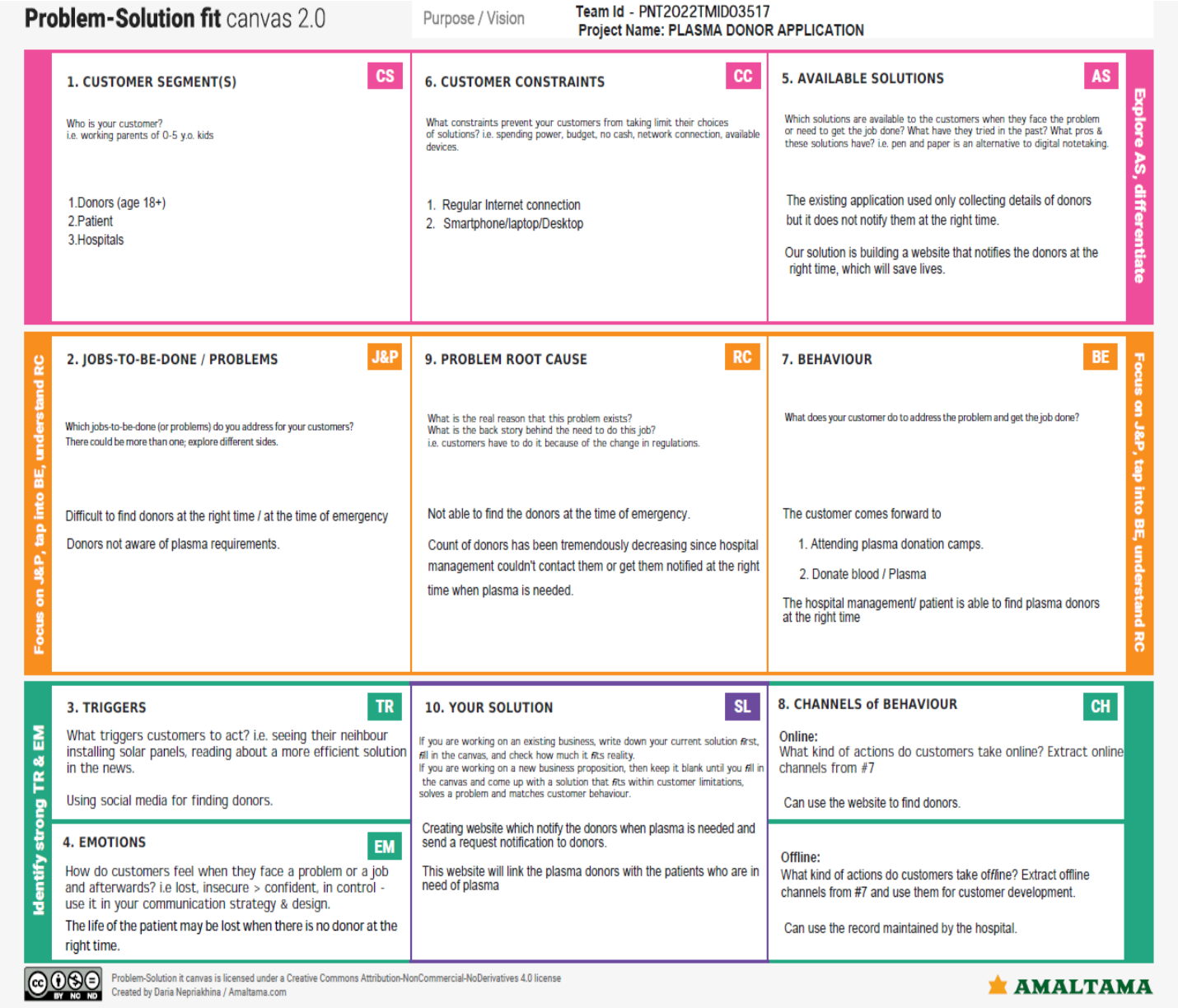
## 3.3 PROPOSED SOLUTION

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.

| S.No. | Parameter | Description |
|---|---|---|
| 1. | Problem Statement (Problem to be solved) | During the COVID 19 crisis, the requirement of plasma became high and the donor count being low. Saving the donor information and helping the need by notifying the current donors would be a helping hand. It is very difficult to find the respective blood group donors when anyone is in need. The donors also face a lot of inconvenience when using the system to donate plasma. |
| 2. | Idea / Solution description | The donors can register their details in the application. Their registered details will be saved in the database. When there is a need for plasma, the relevant donors will be alerted via their mail. |
| 3. | Novelty / Uniqueness | Our application is unique in the fact that it is very convenient for the donor. It has a unique, simple and elegant User Interface that makes it much easier to understand and navigate. |
| 4. | Social Impact / Customer Satisfaction | It assures people that plasma is available at anytime, anywhere by contacting all relevant donors by mail, largely reducing the worry of patients. |

| | | |
|---|---|---|
| 5. | Business Model (Revenue Model) | This application can collaborate with the Government and Non-Profitable Organizations where they can utilize the certain amount every year for its maintenance. |
| 6. | Scalability of the Solution | A large amount of requests for plasma donation can be processed at the same time. |

# 3.4 PROBLEM SOLUTION FIT



# 4. REQUIREMENT ANALYSIS

## 4.1 FUNCTIONAL REQUIREENTS

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<br>• Registration through LinkedIn<br>• Registration through Gmail |
| FR-2 | **USER CONFIRMATION** | • Confirmation via OTP<br>• Confirmation via Gmail |
| FR-3 | **DATA ENTRY** | • Collect the data from user<br>• Fill up the appropriate data |
| FR-4 | **ANALYSING DATA** | • Analysing data and providing result based on user request |
| FR-5 | **TECHNOLOGY USED** | • Docker<br>• DB2<br>• Flask<br>• IBM Cloud<br>• Kubernetes<br>• IBM |
| FR-6 | **PHASES** | • Response phase<br>• Request phase |

## 4.2 NON-FUNCTIONAL REQUIREENTS

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

| FR NO. | NON- FUNCTIONAL REQUIREMENT | DESCRIPTION |
|---|---|---|
| NFR-1 | **USABILITY** | • Finding donor at the right time.<br>• Saving life of people who are in need. |
| NFR-2 | **SECURITY** | • Should avoid security misconfiguration.<br>• Must Safeguard your data. |
| NFR-3 | **RELIABILITY** | • Final result will be accurate. |
| NFR-4 | **PERFORMANCE** | • Filtering of donors will be accurate. |
| NFR-5 | **AVAILABILITY** | • It can be accessed at anytime and everywhere. |
| NFR-6 | **SCALABILITY** | • Network input/output. Disk input/output Memory utilization.<br>• CPU usage. |

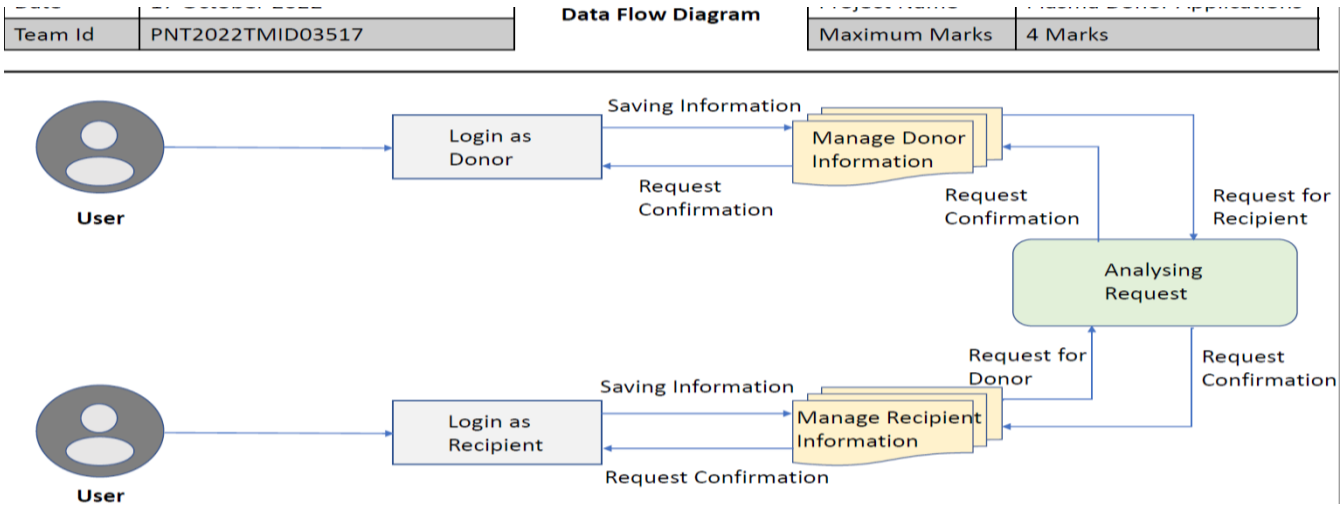
# 5. PROJECT DESIGN

## 5.1 DATA FLOW DIAGRAMS

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

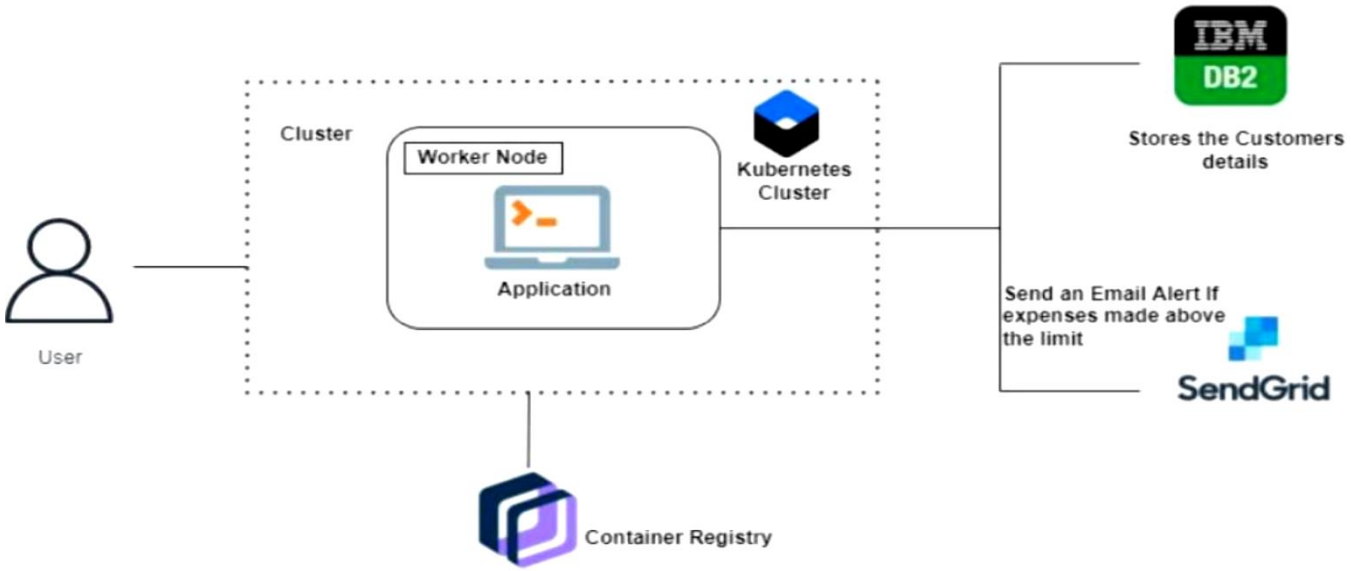
## STEPS:

1. Donor can enter their details and check their eligibility.
2. Hospital In-Charge enter their hospital details and register themselves.
3. Recipients can enter their details and book their slots.
4. After Donor's donation finished, In-charge update the details in database.
5. After Recipient's request for plasma, In-charge has to allocate the the appropriate plasma for recipient.
6. After the process finished, all users enter their feedback to their appropriate requests.
7. All the changes can enter into DB2.

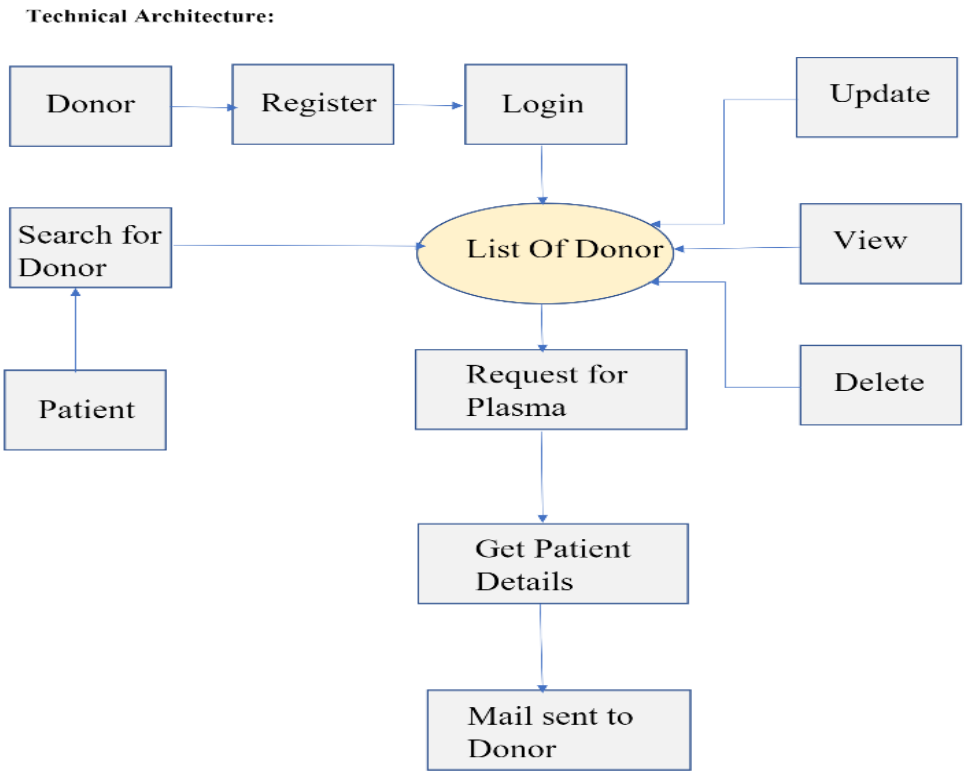


## 5.2 SOLUTION & TECHNICAL ARCHITECTURE

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




## SOLUTION ARCHITECTURE



## 5.3 USER STORIES

| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria | Priority | Release |
|---|---|---|---|---|---|---|
| Donor / Recipient / Hospital In-Charge (Mobile/Desktop user) | Registration | USN-1 | As a user, I can register for the application by entering my email, password, and confirming my password. | I can access my account / dashboard | High | Sprint-1 |
| | | USN-2 | As a user, I will receive confirmation email or SMS once I have registered for the application | I can receive confirmation email & click confirm | High | Sprint-1 |
| | | USN-3 | As a user, I can register for the application through Gmail and Phone Number. | I can register & access the dashboard with Gmail or any kind of Login | Medium | Sprint-2 |
| | Login | USN-4 | As a user, I can log into the application by entering email or phone number & password | I can Log into the Application by using Email ID and Password | High | Sprint-1 |
| Donor / Recipient / Hospital In-Charge (Web user) | Dashboard | USN-5 | As a user, I can be allowed to choose the three options like Donor, Recipient and Hospital In-Charge. | I am a Donor and need to access only Donor registration with my credentials | Medium | Sprint-3 |
| | | USN-6 | | I am a Recipient and need to access only Recipient registration with my credentials. | Medium | Sprint-3 |
| | | USN-7 | | I am a Hospital In-Charge and need to access only In-Charge registration with my hospital's credentials | Medium | Sprint-3 |
| Donor | Donor's Page | USN-8 | As a Donor, I can enter my details and check my eligibility, and book my slot for donation | I am donor, I can get the slot timings and nearby hospital details. | High | Sprint-4 |
| Recipient | Recipient's Page | USN-9 | As a Recipient, I can enter my details and book my slot in a hospital as any nearby. | I am a recipient; I can get the appropriate Plasma present in nearby areas. | High | Sprint-4 |
| Hospital In-Charge | Hospital In-Charge Page | USN-10 | As a Hospital In-Charge, I can enter my details and hospital details as per the conditions. | I am a Hospital In-Charge; I can check the user credentials and do my process | High | Sprint-4 |
| All users (Donor, Recipient, Hospital In-Charge) | At last feedback page | USN-11 | Finally, all users enter their feedback and receive feedbacks and issues. | I am a user; I can send and receive queries through feedback pages. | Medium | Sprint-4 |

# 6. PROJECT PLANNING & SCHEDULING

## 6.1 SPRINT PLANNING & ESTIMATION

| Sprint | Functional Requirement | User Story Number | User Story/ task | Story Points | Priority | Team Members |
|---|---|---|---|---|---|---|
| Sprint - 1 | Develop Flask Web Page | USN - 1 | As a user, I can register for the application by entering my email, password, and confirming my password. | 4 | High | 4 |
| | Registration | USN - 2 | As a user, I will receive confirmation email once I have registered for the application | 8 | High | 4 |
| | Login | USN - 3 | As a user, I can register for the application through google account | 8 | High | 4 |

| | | | | | | |
|---|---|---|---|---|---|---|
| Sprint - 2 | Registration for Plasma Donor | USN - 4 | I can view list represents a specific donor with donor details | 10 | High | 4 |
| | Request for Plasma | USN - 5 | As a user, I can log into the application by entering email & password | 10 | High | 4 |
| Sprint - 3 | Awareness | USN - 6 | As a user, I can log in into the application and view the dashboard for plasma information's. | 10 | Low | 4 |
| | Database | USN - 7 | As a user, I can get notifications after register for plasma donation/needy. | 10 | High | 4 |
| Sprint - 4 | Send Notification | USN - 8 | Admin can access, view, modify, update all details of the plasma donor application | 12 | High | 4 |
| | Software Testing | USN - 9 | As user want to access the application without any drawback we need to test the software before release. | 8 | High | 4 |

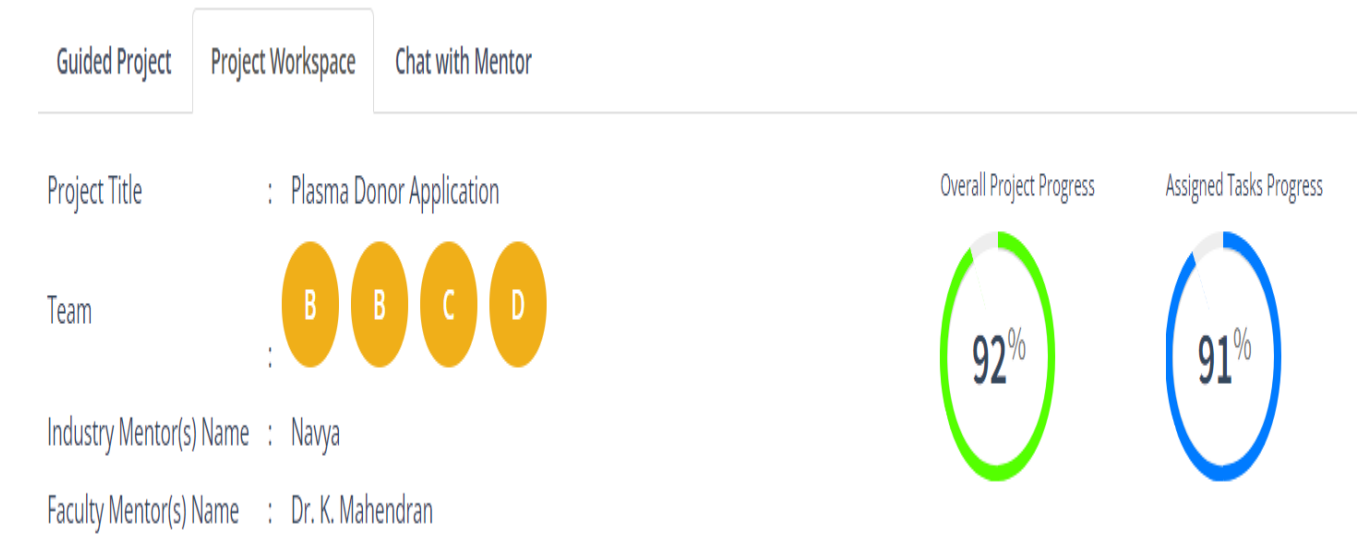
# 6.2 SPRINT DELIVERY SCHDULE

| **Sprint** | **Total Story Points** | **Duration** | **Sprint Start Date** | **Sprint End Date** | **Story Points Completed** | **Sprint Release Date** |
|---|---|---|---|---|---|---|
| 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 Oct 2022 | 12 Nov 2022 | 20 | 12 Nov 2022 |
| Sprint - 4 | 20 | 6 Days | 13 Nov 2022 | 19 Nov 2022 | 20 | 19 Nov 2022 |

**Velocity**

Sprint - 1 (AV) = Sprint Duration/Velocity = 20/6 = 3.33
Sprint - 2 (AV) = Sprint Duration/Velocity = 20/6 = 3.33
Sprint - 3 (AV) = Sprint Duration/Velocity = 20/6 = 3.33
Sprint - 4 (AV) = Sprint Duration/Velocity = 20/6 = 3.33

## 6.3 REPORTS FROM JIRA




# 7. CODING & SOLUTIONING

## 7.1 FEATURE 1

### LOGIN

```
if request.method == 'GET':
    uname = request.args.get("uname")
    psw = request.args.get("psw")

    sql = "SELECT * FROM accounts WHERE username =? AND password=
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.bind_param(stmt, 1, uname)
    ibm_db.bind_param(stmt, 2, psw)
    ibm_db.execute(stmt)
    account = ibm_db.fetch_assoc(stmt)
    print(account)

    if uname == 'admin' and psw == 'admin':
        return redirect(url_for('admin'))

    if account:
        session['loggedin'] = True
        session['id'] = account['USERNAME']
        userid = account['USERNAME']
        session['username'] = account['USERNAME']
        return redirect(url_for("about"))
```

## SIGN UP

```
msg = ''
if request.method == 'POST':
    uname = request.form['uname']
    email = request.form['email']
    name = request.form['name']
    dob = request.form['dob']
    psw = request.form['psw']
    con_psw = request.form['con_psw']

    sql = "SELECT * FROM accounts WHERE username =?"
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.bind_param(stmt, 1, uname)
    ibm_db.execute(stmt)
    account = ibm_db.fetch_assoc(stmt)
    print(account)

    if account:
        msg = 'Account already exists !'
        flash(msg)
        return redirect(url_for("signup"))
    elif psw != con_psw:
        msg = "Password and Confirm Password do not match."
        flash(msg)
        return redirect(url_for("signup"))
    else:
        insert_sql = "INSERT INTO accounts 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, dob)
        ibm_db.bind_param(prep_stmt, 4, uname)
        ibm_db.bind_param(prep_stmt, 5, psw)
        ibm_db.execute(prep_stmt)

        insert_donor = "INSERT INTO donor(Name,Username,Email,DOB,Availability) VALUES (?, ?, ?, ?, ?)"
        prep_stmt = ibm_db.prepare(conn, insert_donor)
        ibm_db.bind_param(prep_stmt, 1, name)
```

# FEATURE 7.2

## SEND MAIL TO SELECTED USER

```
if request.method == 'POST':
    if request.form['select'] == 'select':
        email = request.form["Email"]
        uname = request.form['Username']
        curr_uname = session["username"]
        name = request.form['Name']
        select = "SELECT * from requests where Username = ? and Requestuname = ?"
        stmt = ibm_db.prepare(conn, select)
        ibm_db.bind_param(stmt, 1, uname)
        ibm_db.bind_param(stmt, 2, curr_uname)
        ibm_db.execute(stmt)
        bool = ibm_db.fetch_assoc(stmt)

        print("boolean"+str(bool))
        if not bool:
            request_sql = "INSERT INTO requests VALUES (?, ?)"
            stmt = ibm_db.prepare(conn, request_sql)
            ibm_db.bind_param(stmt, 1, uname)
            ibm_db.bind_param(stmt, 2, curr_uname)
            ibm_db.execute(stmt)
            sendmail(email, 'Plasma donor App plasma request', name,'You have receive
        else:
            print(bool)
        print(email)
        print(name)
```

## SEARCH ACCORDING BLOOD TYPE AND LOCATION

```
global value
bloodgrp = request.form['bloodgrp']
city = request.form['city']

send_sql = "SELECT * FROM donor where BLOODTYPE = ? and CITY = ? and USERNAME != ? and AVAILAB
prep_stmt = ibm_db.prepare(conn, send_sql)
ibm_db.bind_param(prep_stmt, 1, bloodgrp)
ibm_db.bind_param(prep_stmt, 2, city)
ibm_db.bind_param(prep_stmt, 3, session['username'])
ibm_db.bind_param(prep_stmt, 4, 'Available')
ibm_db.execute(prep_stmt)
row = ibm_db.fetch_assoc(prep_stmt)

value = {}
ind = 0
while row != False:
    value[ind] = row
    ind += 1
    row = ibm_db.fetch_assoc(prep_stmt)
print(value)
```

# 8. TESTING

## 8.1 TEST CASES

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Team ID | PNT2022TMID03517 | | | | | | | | |
| | | | | Project Name | Project - Plasma Donor Application | | | | | | | | |
| | | | | Maximum Marks | 4 marks | | | | | | | | |
| **Test case ID** | **Feature Type** | **Component** | **Test Scenario** | **Pre-Requisite** | **Steps To Execute** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Commnets** | **TC for Automation(Y/N)** | **BUG ID** | **Executed By** |
| 1 | Functional | Login Page | Verify user is able to Login into the Application | | 1) Open the Plasma Donor Applicaion 2) Login with user Credentials<br>3) Verify logged in to user account | UserName:Bhuvi<br>Password:Testing | Login Successfully | Working as expected | Pass | | N | | Bhuvanesh |
| 2 | Functional | Sign up Page | Verify the User is able to Signup in th eapplication | | 1) Open the Plasma Donor Applicaion<br>2) Enter the Details and Create a new User<br>3) Verify if User is created and Inserted into DB Table | Username:Kumar<br>Password:Testing<br>Name:Kumar<br>DOB:03/06/2001<br>Password:Testing | Account Created Successfully | Working as expected | Pass | | N | | Brajesh Kumar |
| 3 | Functional | Personal Details page | Verify if all the user details are stored in Database | | 1) Open the Plasma Donor Applicaion 2) Enter the Details<br>3) Verify if user is created and inserted into DB Table | Username:Chidambaram<br>Password:Testing<br>Name:Chidambaram<br>DOB:20/08/2001<br>Age:21<br>Availability:Available<br>Contact No:9047353651<br>City:Chennai<br>State:Tamil Nadu<br>Country:India<br>Blood Type: A+<br>Description: Happy to Donate | User Details must be stored in the database | Working as expected | Pass | | N | | Chidambaram |
| 4 | Functional | Search page | Search users based on blood type, city and availability | | 1) Log in to Plasma Donor Application<br>2) Enter City and Blood type<br>3) All the donor details with city and blood type is displayed. | City:Chennai<br>Blood Type:A+ | Donor details with matching details must be displayed | Working as expected | Pass | | N | | Brajesh Kumar |
| 5 | Functional | Request page | Verify the Request is displayed | | 1) Log in to Plasma Donor Application<br>2) All the requests for the user is received | | All the request received by the user must be displayed | Working as expected | Pass | | N | | Deepak |

# Test Scenarios

- Verify user can able to see login page
- Verify user can able to login to application or not?
- Verify user can able to navigate to create your account page?
- Verify user can able to recovery password
- Verify login page elements

## Search

1. Verify user is able to search by entering keywords in search box
2. Verify user is able to see suggestions based on keyword entered in search box
3. Verify user is able to see related auto suggestions displaying based on keyword entered in search box
4. Verify user is able to see no matches found message when no results are matching with entered keyword
5. Verify user is able to see search detailed page when nothing entered in textbox

## 8.2 USER ACCEPTANCE TESTING

### 1. Purpose of Document

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

### 2. Defect Analysis

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

| Resolution | Severity 1 | Severity 2 | Severity 3 | Severity 4 | Subtotal |
|---|---|---|---|---|---|
| By Design | 5 | 0 | 0 | 0 | 5 |
| 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 | 10 | 0 | 0 | 0 | 10 |

### 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 | 0 | 0 | 0 | 0 |
| Client Application | 5 | 0 | 0 | 5 |
| Security | 0 | 0 | 0 | 0 |
| Outsource Shipping | 0 | 0 | 0 | 0 |
| Exception Reporting | 0 | 0 | 0 | 0 |
| Final Report Output | 5 | 0 | 0 | 5 |
| Version Control | 0 | 0 | 0 | 0 |

# 9. RESULTS

## 9.1 PERFORMANCE METRICS

Team ID PNT2022TMID03517

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 | New | Low | No Changes | Moderate | Yes, 2hrs | >10 to 30% | GREEN | |

NFT - Detailed Test Plan

| S.No | Project Overview | NFT Test approach | umptions/Dependencies/Ri | Approvals/SignOff |
|---|---|---|---|---|
| 1 | Login Page | 1) Open the Plasma Donor App<br>2) Login with User Credentials | No Risks | N/A |
| 2 | Sign Up Page | 1) Open the Plasma Donor App<br>2) Enter the Details and Create a New User | No Risks | N/A |
| 3 | Personal Details Page | 1) Log in to Plasma Donor Application<br>2) Enter all the Personal Details and availability Details | No Risks | N/A |
| 4 | Search Donor Page | 1) Log in to Plasma Application<br>2) Enter City and Blood type<br>3) All the Donor details with city and Blood type is displayed | No Risks | N/A |
| 5 | Request Page | 1) Log in to Plasma Donor Application<br>2) All the Requests for the User is Received | No Risks | N/A |
| 6 | Email Acknowledgement | 1) Mails are Sent to the requested user<br>2) Mails are sent to the reuest user | No Risks | N/A |

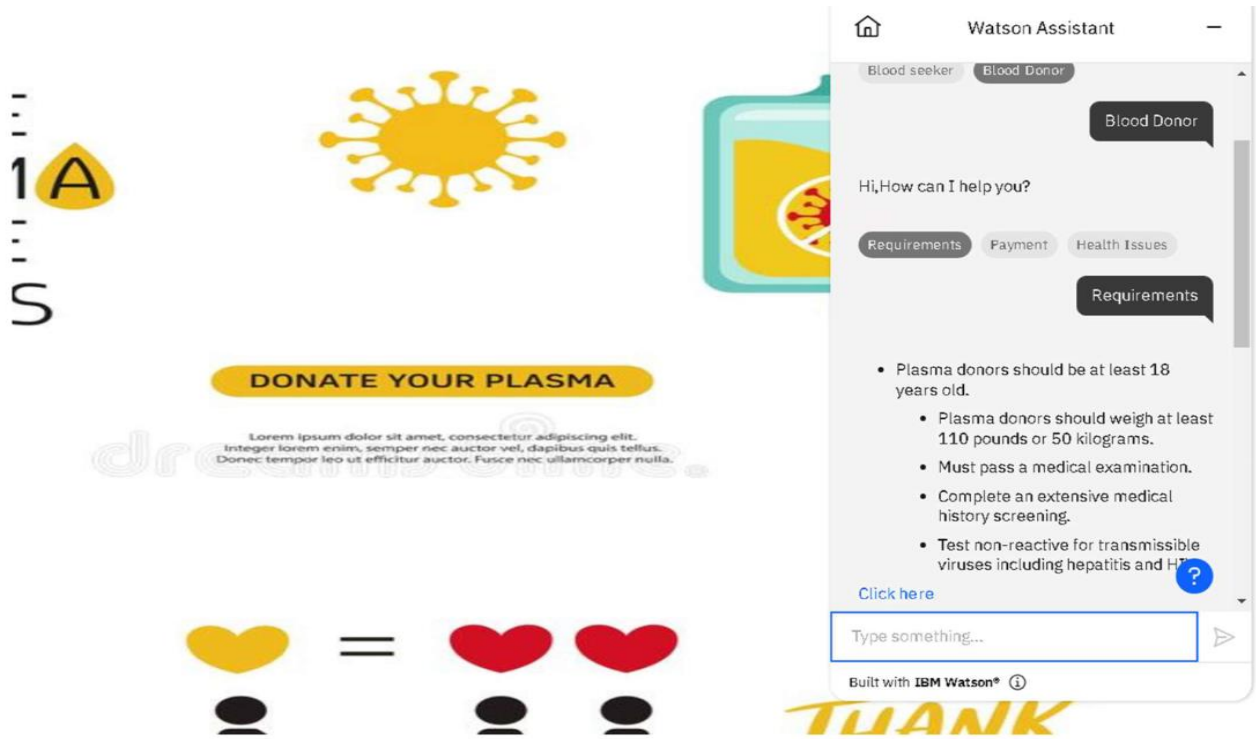
End Of Test Report

| S.No | Project Overview | NFT Test approach | NFR - Met | Test Outcome | GO/NO-GO decision | Recommendations | Identified Defects (Detected/Closed/Open) | Approvals/SignOff |
|---|---|---|---|---|---|---|---|---|
| 1 | Plasma Donor | 2) Test for all Testcases<br>3) Log out of the Plasma Donor Application | YES | Test Passed | GO | N/A | None | N/A |

| | |
|---|---|
| Team ID | PNT2022TMID03517 |
| Project Name | Project - Plasma Donor Application |
| Maximum Marks | 4 marks |

| Test case ID | Feature Type | Component | Test Scenario | Pre-Requisite | Steps To Execute | Test Data | Expected Result | Actual Result | Status | Commnets | TC for Automation(Y/N) | BUG ID | Executed By |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Functional | Login Page | Verify user is able to Login into the Application | | 1) Open the Plasma Donor Applicaion 2) Login with user Credentials<br>3) Verify logged in to user account | UserName:Bhuvi<br>Password:Testing | Login Successfully | Working as expected | Pass | | N | | Bhuvanesh |
| 2 | Functional | Sign up Page | Verify the User is able to Signup in th eapplication | | 1) Open the Plasma Donor Applicaion<br>2) Enter the Details and Create a new User<br>3) Verify if User is created and Inserted into DB Table | Username:Kumar<br>Password:Testing<br>Name:Kumar<br>DOB:03/06/2001<br>Password:Testing | Account Created Successfully | Working as expected | Pass | | N | | Brajesh Kumar |
| 3 | Functional | Personal Details page | Verify if all the user details are stored in Database | | 1) Open the Plasma Donor Applicaion 2) Enter the Details<br>3) Verify if user is created and inserted into DB Table | Username:Chidambaram<br>Password:Testing<br>Name:Chidambaram<br>DOB:20/08/2001<br>Age:21<br>Availability:Available<br>Contact No:9047353651<br>City:Chennai<br>State:Tamil Nadu<br>Country:India<br>Blood Type: A+<br>Description: Happy to Donate | User Details must be stored in the database | Working as expected | Pass | | N | | Chidambaram |
| 4 | Functional | Search page | Search users based on blood type, city and availability | | 1) Log in to Plasma Donor Application<br>2) Enter City and Blood type<br>3) All the donor details with city and blood type is displayed. | City:Chennai<br>Blood Type:A+ | Donor details with matching details must be displayed | Working as expected | Pass | | N | | Brajesh Kumar |
| 5 | Functional | Request page | Verify the Request is displayed | | 1) Log in to Plasma Donor Application<br>2) All the requests for the user is received | | All the request received by the user must be displayed | Working as expected | Pass | | N | | Deepak |


GIVE
PLASMA
SAVE
LIVES
DONATE YOUR PLASMA
Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Integer lorem enim, semper nec auctor vel, dapibus quis tellus.
Donec tempor leo ut efficitur auctor. Fusce nec ullamcorper nulla.
Close
Hi! I'm a virtual assistant. How can I help you today?
THANK YOU


DONATE YOUR PLASMA
Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Integer lorem enim, semper nec auctor vel, dapibus quis tellus.
Donec tempor leo ut efficitur auctor. Fusce nec ullamcorper nulla.
Watson Assistant
Blood seeker
Blood Donor
Blood Donor
Hi,How can I help you?
Requirements
Payment
Health Issues
Requirements
Plasma donors should be at least 18 years old.
Plasma donors should weigh at least 110 pounds or 50 kilograms.
Must pass a medical examination.
Complete an extensive medical history screening.
Test non-reactive for transmissible viruses including hepatitis and H
Click here
Type something...
Built with IBM Watson®

# 10. ADVANTAGES & DISADVANTAGES

## 1. ADVANTAGES

- The project is identified by the merits of the system offered to the user. The merits of this project are as follows:
- It is a web-enabled cloud based project.This project offers user to enter the data through simple and interactive forms.
- This is very helpful for the client to enter the desired information through so much simplicity.
- This website is fast and offers great accuracy as compared to manual registered keeping.
- Less maintenance is required.
- User Friendly. It is very easy to use and understand. It is easily workable and accessible for everyone.
- The user is mainly more concerned about the validity of the data, whatever he is entering.
- There are checks on every stages of any new creation, data entry or updations. so that the user cannot enter the invalid data, which can create problems at later date.Sometimes the user finds in the later stages of using project that he needs to update some of the information that he entered earlier.
- There are options for him by which he can update the records. More over there is restriction for his that he cannot change the primary data field.
- This keeps the validity of the data to longer e0tent.User is provided the option of monitoring the records he entered earlier. He can see the desired records with the variety of options provided by him.
- From every part of the project the user is provided with the links through framing so that he can go from one option of the project to other as per the requirement.
- This is bound to be simple and very friendly as per the user is concerned.
- Decision making process would be greatly enhanced because of faster processing of information since data collection from information available on computer takes much less time then manual system.
- Allocating of sample results becomes much faster because at a time the user can see the records of last years.
- Easier and faster data transfer through latest technology associated with the computer and communication.

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

## 2. DISADVANTAGES

Wrong inputs will affect the project outputs. Internet Connection is Required. Reports are not Verified. It cannot automatically verify the genuine users.

## 11. CONCLUSION

- This project proved good. It provided practical knowledge of not only programming in ASP.NET and VB.NET web based application and no some extent windows Application and SQL Server, but also about all handling procedure related with "Plasma Donor Application".
- It also provides knowledge about the latest technology used in developing web enabled application and client server technology that will be great demand in future. This will provide better opportunities and guidance in future in developing projects independently.

## 12. FUTURE SCOPE

Plasma Donor Application is a web application to build such a way that it should suits for all type of blood banks in future. One important future scope is availability of location-based blood bank details and extraction of location-based donor's detail, which is very helpful to the acceptant people. All the time the network facilities cannot be use. This time donor request does not reach in proper time, this can be avoided through adding some message sending procedure this will help to find proper blood donor in time. This will provide availability of blood in time.

## 13. APPENDIX

Source Code: https://github.com/IBM-EPBL/IBM-Project-27333-1660054225