NALAYATHIRAN	
CLOUD APPLICATION DEVELOPMENT	
PLASMA DONOR APPLICATION	
TEAM ID	PNT2022TMID44210
TEAM LEADER	PRAGATHEESWARAN
TEAM MEMBER 1	DINAKAR A S
TEAM MEMBER 2	KAVIN KUMAR A
TEAM MEMBER 3	KARTHI S

Project Report Format

1. **INTRODUCTION**
   1.1 Project Overview
   1.2 Purpose
2. **LITERATURE SURVEY**
   2.1 Existing problem
   2.2 References
   2.3 Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
   3.1 Empathy Map Canvas
   3.2 Ideation & Brainstorming
   3.3 Proposed Solution
   3.4 Problem Solution fit
4. **REQUIREMENT ANALYSIS**
   4.1 Functional requirement
   4.2 Non-Functional requirements
5. **PROJECT DESIGN**
   5.1 Data Flow Diagrams
   5.2 Solution & Technical Architecture
   5.3 User Stories
6. **PROJECT PLANNING & SCHEDULING**
   6.1 Sprint Planning & Estimation
   6.2 Sprint Delivery Schedule
   6.3 Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
   7.1 Feature 1
   7.2 Feature 2
   7.3 Database Schema (if Applicable)
8. **TESTING**
   8.1 Test Cases
   8.2 User Acceptance Testing
9. **RESULTS**
   9.1 Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**
    Source Code
    GitHub & Project Demo Link

INTRODUCTION

Project Overview

A plasma is a liquid portion of the blood, over 55% of human blood is plasma. Plasma is used to treat various infectious diseases and it is one of the oldest methods known as plasma therapy. Plasma therapy is a process where blood is donated by recovered patients in order to establish antibodies that fights the infection. In this project plasma donor application is being developed by using IBM cloud . For instance, during COVID 19 crisis the requirement for plasma increased drastically as there were no vaccination found in order to treat the infected patients, with plasma therapy the recovery rates where high but the donor count was very low and in such situations it was very important to get the information about the plasma donors. Saving the donor information and notifying about the current donors would be a helping hand as it can save time and help the users to track down the necessary information about the donors.

Purpose

This app will help patients to get plasma and find the donors who are nearby. Donors data stored in cloud storage so users can access the data from anywhere and anytime. registration is very simple process in this app.

LITERATURE SURVEY

Applying optimization methods to healthcare management and logistics is a developing research area with numerous studies. Specifically, facility location, staff rostering, patient allocation, and medical supply transportation are the main themes analyzed. Optimization approaches have been developed for several healthcare related problems, ranging from the resource management in hospitals to the delivery of care services in a territory. However, optimization approaches can also improve other services in the health system that have been only marginally addressed, yet. One of them is the Blood Donation (BD) system, aiming at providing an adequate supply of blood to Transfusion Centres (TCs) and hospitals. Blood is necessary for several treatments and surgeries, and still a limited resource. The need for blood is about ten million units per year in the USA, 2.1 in Italy and 2 in Turkey; moreover, people still die in some countries because of inadequate supply of blood products (World Health Organization 2014). Hence, BD plays a fundamental role in healthcare systems, aiming at guaranteeing an adequate blood availability to meet the demand and save lives. In Western countries, blood is usually collected from donors, i.e., unpaid individuals who give blood voluntarily. 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.

References

S.NO	TITLE	Authors	Abstract	Drawbacks
1	Optimization of Blood Donor Information and Management System	K. Yamini, M. E(CSC), SVCET, Thirupachur, India R. Devi, Asst. Professor, SVCET, Thirupachur, India	Emergency situations, such as accidents, create an immediate, critical need for specific blood type. In addition to emergency requirements, advances in medicine have increased the need for blood in many ongoing treatments and elective surgeries. Despite increasing requirements for blood, only about 5% of the Indian population donates blood. In this paper we propose a new and the Blood donor App. efficient way to overcome such scenarios with our project. We have to create a new idea, just touch the button. Donor will be prompted to enter an individual's details, like name, phone number, and blood type. After that your contact details will appear in	The accuracy of the location displayed on the map was beyond the scope of this Project. Only Android was used as a mobile operating system to test the application

			alphabetical order on the screen; the urgent time of a blood requirement, you can quickly check for contacts matching a particular or related blood group and reach out to them via Phone Call/SMS through	
2	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. Manual systems as compared to Computer Based Information Systems are time consuming, laborious, and costly. This paper introduces the review of the main features, merits and	Do not provide the better inventory solution to the end use It requires an active internet connection.

			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. First I will give some basic introduction about blood banks then I will try to provide comparative study of some existing web based blood bank system. After that I will introduce some new idea for improving the existing techniques used in web based blood bank system and at end I will conclude this paper	
3	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	Internet Connection is mandatory There is no proper centralized database for registered donors

			conventional methods of collecting blood are absolute. An automated system is required to manage the centers and to showcase the information to the interested parties. We have developed a website that single handedly solves all these issues related to blood donation and reception. 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. We have introduced a search tab to search available people ready to donate. In our proposed system in the donor registration, health -related details would be updated in the blood management system database for all	

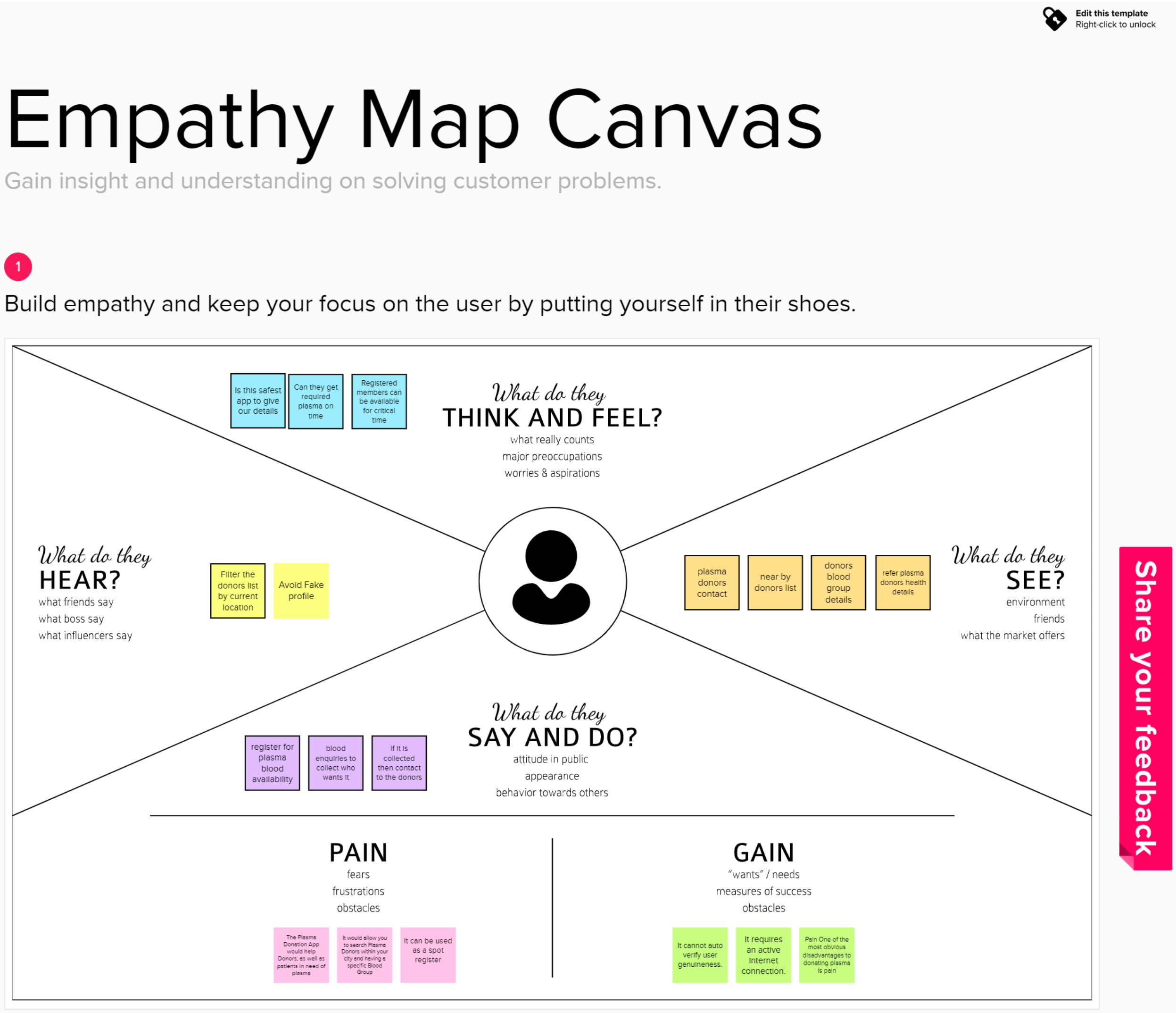
			to see	
4	A Study on Blood Bank Management	A. Clemen Teena, K Sankar S. Kannan	‘Blood Bank Information System’ will be an information management system which helps to manage the records of donors and patients at a blood bank. The system will allow the authorized blood bank officer to login using a secret password and easily manage the records of the blood donors and the patients in need of blood	No search filter available UI improvement in Login page

Problem Statement Definition

Plasma plays the critical role of maintaining a healthy blood pressure, blood volume and a proper pH balance. Which being the most important element for humans to survive, developing an application to interconnect plasma requesters and donors might help save lives during emergency. Making such an application which is user friendly as well as has more features for serving the people better, we are proposing a model in which we are going to connect the plasma donor and requester together in a unique way

IDEATION & PROPOSED SOLUTION

Empathy Map Canvas



Empathy Map Canvas

Gain insight and understanding on solving customer problems.

1

Build empathy and keep your focus on the user by putting yourself in their shoes.

What do they THINK AND FEEL?

what really counts

major preoccupations

worries & aspirations

Is this safest app to give our details

Can they get required plasma on time

Registered members can be available for critical time

What do they HEAR?

what friends say

what boss say

what influencers say

Filter the donors list by current location

Avoid Fake profile

What do they SEE?

environment

friends

what the market offers

plasma donors contact

near by donors list

donors blood group details

refer plasma donors health details

What do they SAY AND DO?

attitude in public

appearance

behavior towards others

register for plasma blood availability

blood enquiries to collect who wants it

If it is collected then contact to the donors

PAIN

fears

frustrations

obstacles

The Plasma Donation App would help Donors, as well as patients in need of plasma

It would allow you to search Plasma Donors within your city and having a specific Blood Group

It can be used as a spot register

GAIN

"wants" / needs

measures of success

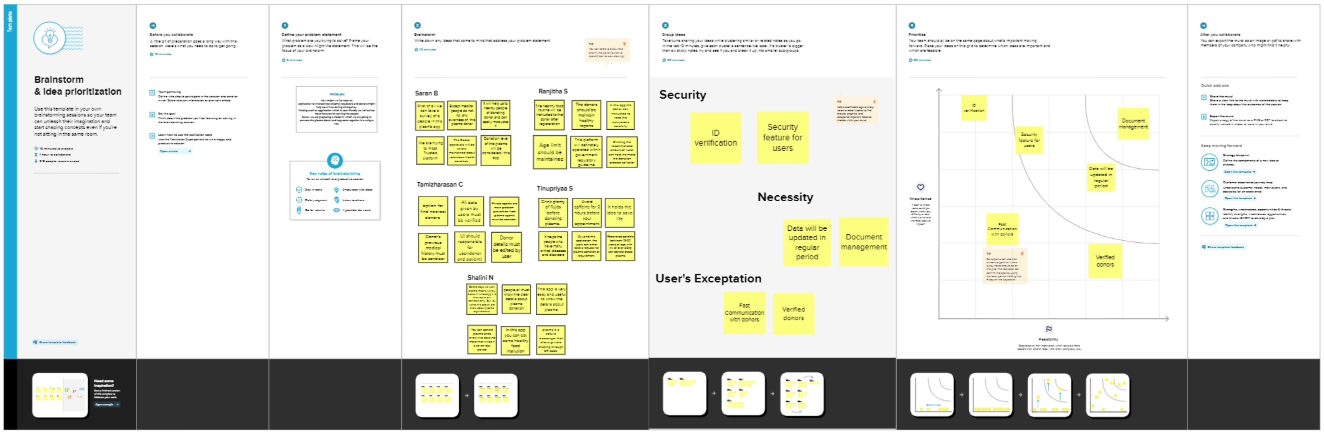
obstacles

It cannot auto verify user genuineness.

It requires an active internet connection.

Pain One of the most obvious disadvantages to donating plasma is pain

Ideation & Brainstorming



Proposed Solution

Problem Statement:

Plasma plays the critical role of maintaining a healthy blood pressure, blood volume and a proper pH balance. Which being the most important element for humans to survive, developing an application to interconnect plasma requesters and donors might help save lives during emergency. Making such an application which is user friendly as well as has more features for serving the people better, we are proposing a model in which we are going to connect the plasma donor and requester together in a unique way.

Novelty:

Cloud communicating is an emerging technology that can be integrated with traditional health management used to provide better health services. Traditional healthcare systems mainly include personal and public healthcare services, teaching and research activities. Personal healthcare services are offered at hospitals, homes and different organizations. Public healthcare services involve guidelines for drugs, food and safety policies to maintain a healthy environment. Teaching and research activities are essential for prevention, detection, tracking and treatment of diseases. Healthcare information systems are designed today for the convenience of the user who obtains its bene fits. In many emergency situations there is an immediate, critical need for specific plasma. In addition to emergency requirements, advances in medicine have increased the need for plasma in many ongoing treatments and medical surgeries. To motivate people for plasma donation and to help patients receive plasma in emergency situations, we have designed an application to overcome all the problems which the current offline as well as online systems face. If in emergency a patient requires plasma, using this application we'll not just be able to contact Blood/Plasma Bank and Hospitals but can also seek help from individual registered Donors. In developing countries, especially like India, the plasma resource

Feasibility Of Ideas:

Admin:

Admin can manage both donors & acceptors. He can add or remove any user from the system. Each member in a donor & acceptor is given a user id and password, which identifies him uniquely. From admin module use can change donor details, delete donor or change the password.

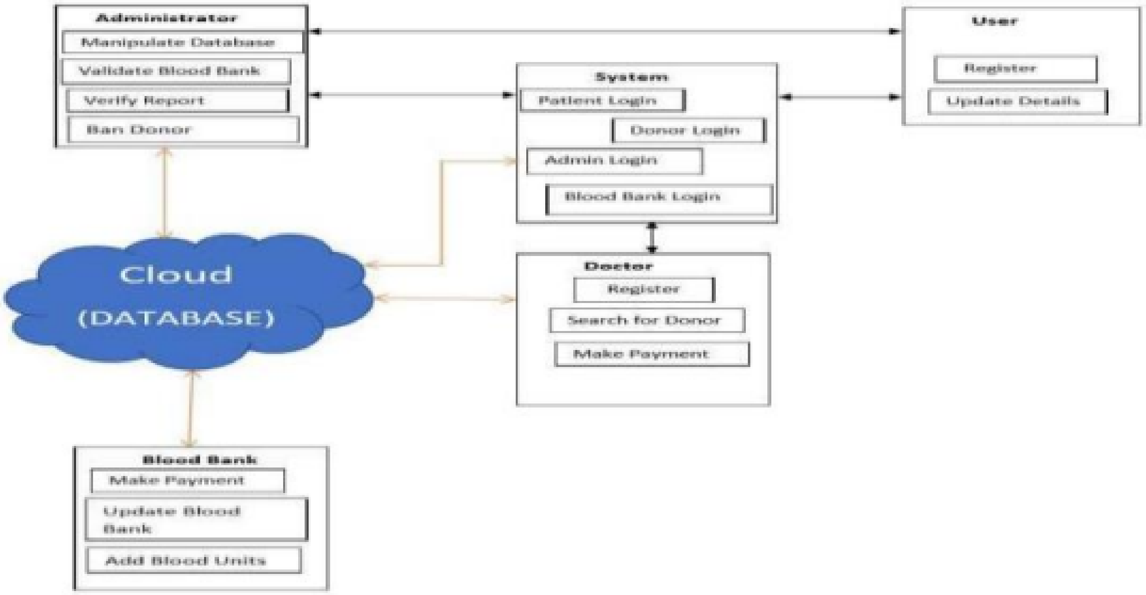
- Register
- Change Password
- Modify donor details
- Delete donor details
- Logout

Login:

To login in the system user has first register himself/herself. After successful Registration user can login into the system Plasma Donor. Acceptors: This module helps user to find plasma group. When user click on find a plasma group system ask him to enter plasma group he wants to search. After entering the plasma group, system search for the availability of the plasma group and give him the list of the donors who has the same plasma group. Whenever a user wants to change password, he can select the change password option. Then system ask the user to enter old username and password then system check the credentials and change the password. Clicking on logout button user can log out from the system.

Business Model:

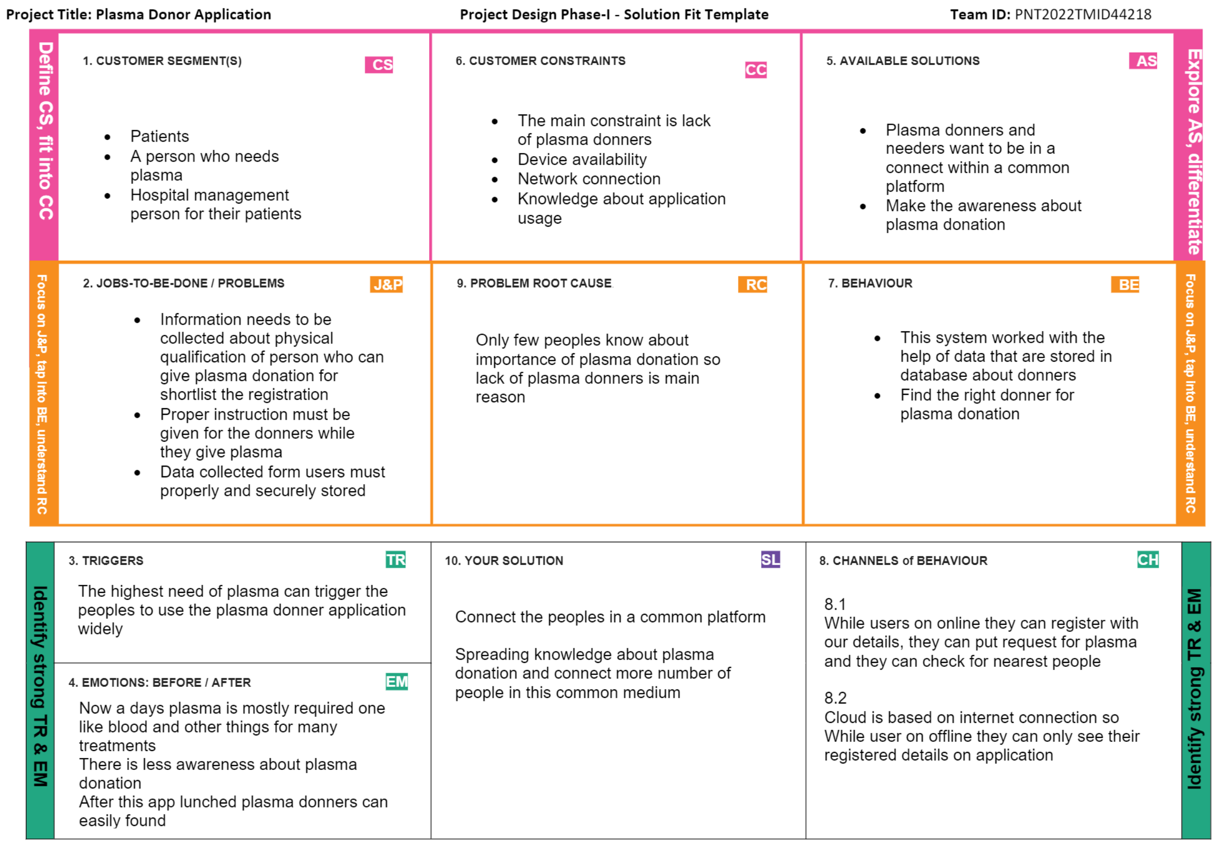
The people who are in need of plasma can search in our site for getting the details of donors having the same plasma group and within the same city or his current location. They can directly search a donor and can select a city name as well as the plasma group which he needs. He then gets the details of the donors who exist within the city and the same plasma that he has selected. If no match was are found for the city and group selected by him, he gets a message 'SORRY DONORS ARE NOT AVAILABLE WITH THE FOLLOWING PLASMA GROUP AND AREA'.



Scalability:

The aim is to build a Lifesaver E-Plasma Donation App using Cloud with advanced features that will help to overcome the barrier between plasma bank, plasma donor and patient. To build an android application that will help people to get plasma in emergency situations like natural disasters using features like geo-tagging, SMS Gateway and payment gateway. To motivate people for plasma donation and to help patients receive plasma in emergency situations, we have designed an application to overcome all the problems which the current offline as well as online systems face. If in emergency a patient requires plasma, using this application we'll not just be able to contact plasma Bank and Hospitals but can also seek help from individual registered Donors

Problem Solution fit



Project Title: Plasma Donor Application **Project Design Phase-I - Solution Fit Template** **Team ID:** PNT2022TMID44218

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS • Patients • A person who needs plasma • Hospital management person for their patients	6. CUSTOMER CONSTRAINTS CC • The main constraint is lack of plasma donners • Device availability • Network connection • Knowledge about application usage	5. AVAILABLE SOLUTIONS AS • Plasma donners and needers want to be in a connect within a common platform • Make the awareness about plasma donation	Explore AS, differentiate
Focus on J&P, tap into BE, understand RC	2. JOBS-TO-BE-DONE / PROBLEMS J&P • Information needs to be collected about physical qualification of person who can give plasma donation for shortlist the registration • Proper instruction must be given for the donners while they give plasma • Data collected form users must properly and securely stored	9. PROBLEM ROOT CAUSE RC Only few peoples know about importance of plasma donation so lack of plasma donners is main reason	7. BEHAVIOUR BE • This system worked with the help of data that are stored in database about donners • Find the right donner for plasma donation	Focus on J&P, tap into BE, understand RC
Identify strong TR & EM	3. TRIGGERS TR The highest need of plasma can trigger the peoples to use the plasma donner application widely 4. EMOTIONS: BEFORE / AFTER EM Now a days plasma is mostly required one like blood and other things for many treatments There is less awareness about plasma donation After this app lunched plasma donners can easily found	10. YOUR SOLUTION SL Connect the peoples in a common platform Spreading knowledge about plasma donation and connect more number of people in this common medium	8. CHANNELS of BEHAVIOUR CH 8.1 While users on online they can register with our details, they can put request for plasma and they can check for nearest people 8.2 Cloud is based on internet connection so While user on offline they can only see their registered details on application	Identify strong TR & EM

REQUIREMENT ANALYSIS

FR No	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Gmail
FR-2	User Confirmation	Confirmation via Email
FR-3	User Login	Login using Registered email Id
FR-4	Sent Request	If plasma is required, the receiver will contact the donor
FR-5	Contact Donor	Contact the donor directly if a phone number is given
FR-6	View donation camps	View the list of donation camps happening nearby

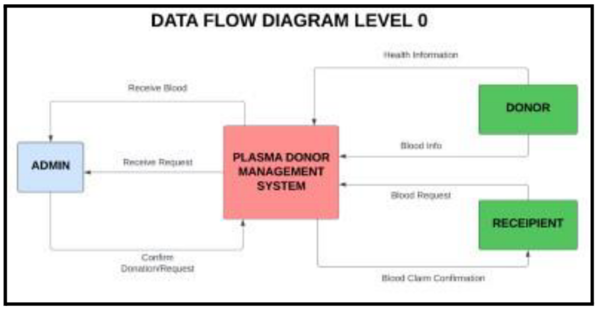
Non-functional Requirements

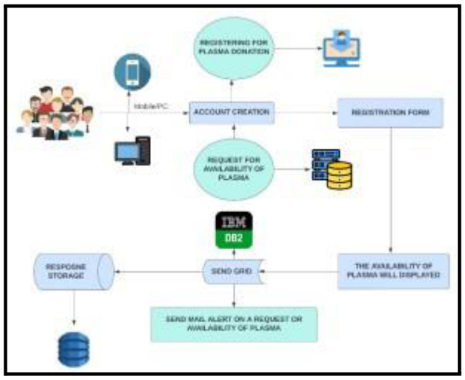
FR No	Non-Functional Requirement	Description
FR-1	Usability	The user interface of the plasma donor system must be well-designed and welcoming.
FR-2	Security	Data storage is required by

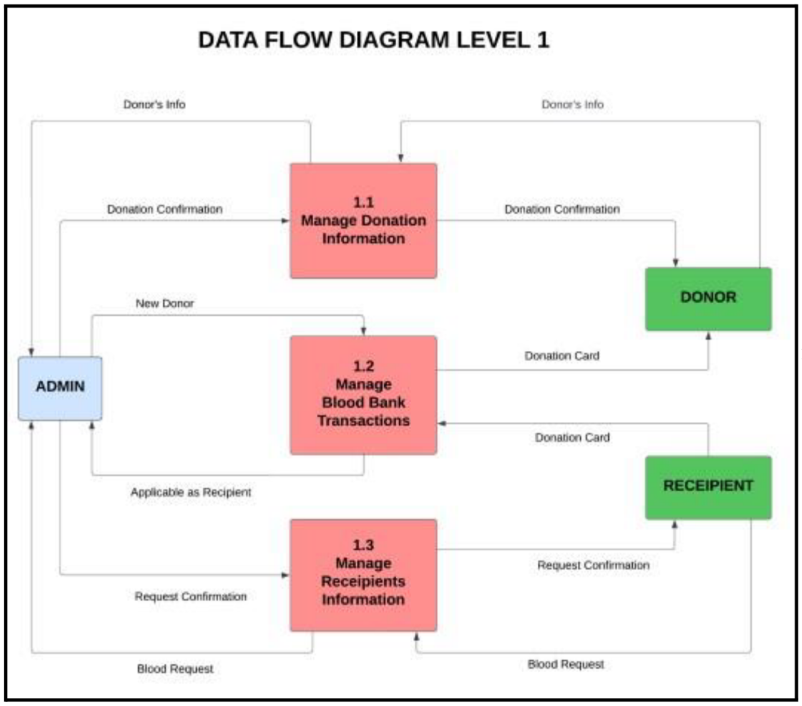
		security systems, just like it is by many other applications. Databases are able to keep all the donor information that is viewed by applications. It must be secured with email Id and password.
FR-3	Reliability	The system has the ability to work all the times without failures apart from network failure. A donor can have the faith on the system. The authorities will keep the privacy of all donors in a proper manner.
FR-4	Performance	The Plasma donor System must perform well in different scenarios. The system is interactive and delays involved are less.
FR-5	Availability	The system, including the online components, should be available 24/7.
FR-6	Scalability	The system offers the proper resources for issue solutions and is designed to protect sensitive information during all phases of operation.

PROJECT DESIGN

Data Flow Diagrams




REGISTERING FOR PLASMA DONATION
Mobile/PC
ACCOUNT CREATION
REGISTRATION FORM
REQUEST FOR AVAILABILITY OF PLASMA
IBM
DB2
RESPOSNE STORAGE
SEND GRID
THE AVAILABILITY OF PLASMA WILL DISPLAYED
SEND MAIL ALERT ON A REQUEST OR AVAILABILITY OF PLASMA

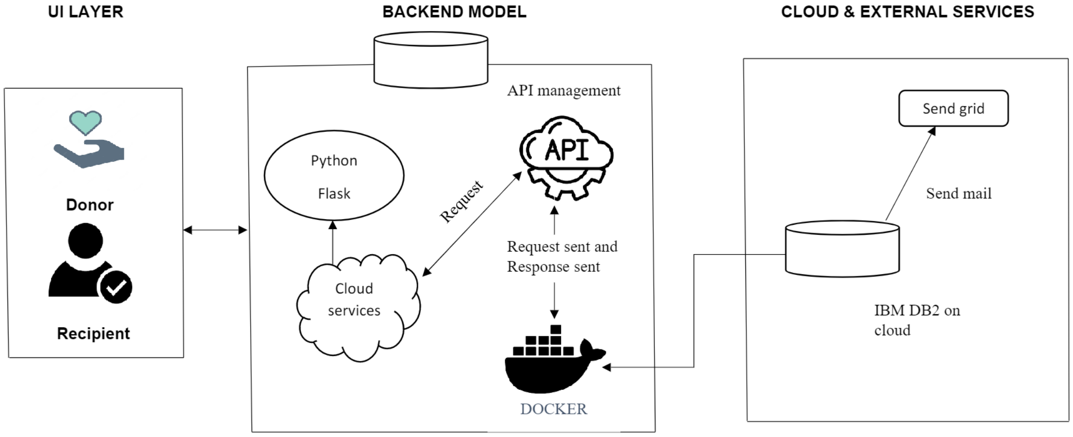


1.2

Solution & Technical Architecture

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2



User Stories

User Type	**Functional Requirement (Epic)**	**User Story Numbe r**	**User Story / Task**	**Acceptance criteria**	**Priority**	**Release**
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and	I can access my account / dashboard	High	Sprint-1

			confirming my password.			
		USN-2	As a user, I will receive confirmation on email once I have registered for the application	I can receive confirmation on email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-1
		USN-4	As a user, I can register for the application through Gmail	I can receive confirmation on email &click confirm	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can enter into my account	High	Sprint-1
	Dashboard	USN-6	As a user ,Display	I can donate/get	High	Sprint-2

			all details about plasma application	details about the plasma		
Customer (Web user)	Application	USN-7	As a user ,I can register, login and see details about plasma	I can access the donor details and availability of plasma	High	Sprint-3
Customer Care Executive	Update Plasma storage	USN-8	Keep track the availability of the Plasma	I can provide application for customer needs	High	Sprint-4
Administrato r	Verify donor details	USN-9	To add the donor plasma details in application	I can Control the all details in this application	Medium	Sprint-3
Customer Care Executive	Verify Customer Feedback	USN-10	To design the application that meets user's desires	I can satisfy the customer expectations	Medium	Sprint-4
Customer Care Executive	Control all Plasma details	USN-11	Make sure to check the availability of	I can alert notification through email and SMS	High	Sprint-2

			plasma in application			
Administrato r	Performanc e of application	USN-12	To make the process more efficient	I can save time, cost by improving the Plasma management application	High	Sprint - 4

PROJECT PLANNING & SCHEDULING

Sprint Planning & Estimation

Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Story Points**	**Priority**
Sprint 1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	20	High

Sprint 2	Login	USN-2	As a user, I can login into my account through the registered mail ID.	20	High
Sprint 3	Donor Information	USN-3	As a user, I can fill the information like blood pressure, blood group, address, mobile number and other information.	20	Low
Sprint 4	Finding the Donor	USN-4	The patient can find the donor by their blood groups, location.	20	Medium

Sprint Delivery Schedule

Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date (Actual)**
Sprint 1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint 2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint 3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022

Sprint 4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

ADVANTAGES :

- User can find plasma donor information
- User can get immediate help from donors

DISADVANTAGES :

- Internet connection required

APPENDIX

login.html

```
<!DOCTYPE html>
<html >
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>Plasma Donor App</title>
          <link rel="stylesheet" href="style.css">

<style>
.login{
top: 20%;
}
</style>
</head>

<body>
<div class="header">
```

```
<div>Plasma Donor App</div>
        <ul>
                <li><a href="register.html">Register</a></li>
                <li><a class="active" href="register.html">Request</a></li>
        </ul>
</div>
 <div class="login" >
                <div>
                </div>

    <!-- Main Input For Receiving Query to our ML -->
   <form action="{{ url_for('loginpage')}}"method="post">
        <input type="text" name="user" placeholder="Enter UserName"
required="required" style="color:black" />
                <input type="password" name="passw" placeholder="Enter Password"
required="required" style="color:black" />
      <button type="submit" class="btn btn-primary btn-block btn-large">Login</button>

   </form>
<br><br>
<div style="color:black"></div>
 </div>

</body>
</html>
```

register.html

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

```
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>Plasma Donor App</title>
        <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet'
type='text/css'>
        <link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet'
type='text/css'>
        <link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet'
type='text/css'>
        <link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300'
rel='stylesheet' type='text/css'>
        <link rel="stylesheet" href="{{ url_for('static', filename='style1.css') }}">
        <link rel="stylesheet" href="style.css">

<style>
.login{
top: 20%;
}
</style>
</head>

<body>
<div class="header">
<div>Plasma Donor App</div>
        <ul>
```

```
                    <li><a class="active" href="login.html">Home</a></li>
          </ul>
</div>
 <div class="login">

     <!-- Main Input For Receiving Query to our ML -->
    <form action="/adddonor",method="post">
          <input type="text" name="name" placeholder="Enter Your Name"
required="required" style="color:black"/>
         <input type="email" name="email" placeholder="Enter Email" required="required"
style="color:black"/>
                    <input type="text" name="phone" placeholder="Enter 10-digit mobile
number" required="required" style="color:black"/>
                    <input type="city" name="city" placeholder="Enter Your City Name"
required="required" style="color:black"/>
                    <select name="infect">

                                                    <option value="select" selected>Select COVID
infection status</option>
                                                  <option value="infected">Infected</option>
                                                  <option value="uninfected">Uninfected</option>
                    </select>
        <select name="blood">
                                                   <option value="select" selected>Choose your blood
group</option>
                                                  <option value="O Positive">O Positive</option>
                                                  <option value="A Positive">A Positive</option>
                                                  <option value="B Positive">B Positive</option>
                                                  <option value="AB Positive">AB Positive</option>
```

```
                                                <option value="O Negative">O Negative</option>
                                                <option value="A Negative">A Negative</option>
                                                <option value="B Negative">B Negative</option>
                                                            <option value="AB Negative">AB Negative</option>
                  </select>
                   <input type="password" name="passw" placeholder="Enter Password" required="required" style="color:black"/>
                               <button type="submit" class="btn btn-primary btn-block btn-large">Register</button>

   </form>

 <br><br>
<div style="color:black"></div>
 </div>




</body>
</html>
```

stats.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Plasma Donar App</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
                                                  <link                        rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
                                                                                        <script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script
>
                                                                                        <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
  <link rel="stylesheet" href="{{ url_for('static', filename='style1.css') }}">
  <link rel="stylesheet" href="style.css">
</head>
<style>
            .big{
            top:70;
            background-color:white;
            margin-top:80px;
            margin-left:550px;
            margin-right:550px;
            height:200px;
            border-radius: 25px;
            border: 3px solid #4a77d4;
            box-shadow: 6px 8px 4px grey;
            text-align:center;
            }
            .row{

            height:150px;
```

```
                }
                .col{
                        margin:10px;
                        margin-left:50px;
                        margin-right:50px;
                        border-radius: 25px;
                        border: 1px solid #4a77d4;
                        box-shadow: 0px 8px 4px grey;
                        text-align:center;
                }
                .ext{
                margin-top:25px;
                line-height:40px;
                }
                .ext1{
                margin-top:40px;
                line-height:50px;
                font-size:25px;
                color:#f95450;
                }

</style>
<body>

<div class="container-fluid">
<div class="header">
<div><b>Plasma Donar App</b></div>
<ul>
                <li><a href="/requester">Request</a></li>
```

```
            <li><a class="active" href="/login">Home</a></li>
      </ul>
</div>
 <br>
 <div class="big">
  <div class="box">
            <div class="ext1"><font
size="20px">{{b}}</font><br><b>Donors</b></div>
      </div>
 </div>
 <br>
 <div class="row">
  <div class="col" >
            <div class="ext">{{b1}}<br><b>O Positive</b></div>
      </div>
  <div class="col" >
            <div class="ext">{{b2}}<br><b>A Positive</b></div>
      </div>
  <div class="col" >
            <div class="ext">{{b3}}<br><b>B Positive</b></div>
      </div>
  <div class="col" >
            <div class="ext">{{b4}}<br><b>AB Positive</b></div>
      </div>
 </div>
 <br>
 <div class="row">
  <div class="col" >
            <div class="ext">{{b5}}<br><b>O Negative</b></div>
```

```
        </div>
    <div class="col" >
                <div class="ext">{{b6}}<br><b>A Negative</b></div>
        </div>
    <div class="col" >
                <div class="ext">{{b7}}<br><b>B Negative</b></div>
        </div>
    <div class="col" >
                <div class="ext">{{b8}}<br><b>AB Negative</b></div>
        </div>
  </div>
</div>
</body>
</html>
```

request.html

```
<!DOCTYPE html>
<html >
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>Plasma Donor App</title>
        <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet'
type='text/css'>
        <link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet'
type='text/css'>
        <link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet'
type='text/css'>
```

```
        <link   href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300'
rel='stylesheet' type='text/css'>
        <link rel="stylesheet" href="{{ url_for('static', filename='style1.css') }}">
        <link rel="stylesheet" href="style.css">

<style>
.login{
top: 20%;
}
</style>
</head>

<body>
<div class="header">
<div>Plasma Donor App</div>
        <ul>


                <li><a href="request.html">Request</a></li>
                <li><a href="register.html">Register</a></li>
                <li><a class="active" href="login.html">Home</a></li>
        </ul>
</div>
 <div class="login">
                <div>
                </div>

   <!-- Main Input For Receiving Query to our ML -->
  <form action="{{ url_for('requested')}}"method="post">
```

```
                <select name="bloodgrp">
                                          <option value="select" selected>Choose your blood
group</option>
                                         <option value="O Positive">O Positive</option>
                                         <option value="A Positive">A Positive</option>
                                         <option value="B Positive">B Positive</option>
                                         <option value="AB Positive">AB Positive</option>
                                         <option value="O Negative">O Negative</option>
                                         <option value="A Negative">A Negative</option>
                                         <option value="B Negative">B Negative</option>
                                                    <option      value="AB      Negative">AB
Negative</option>
                </select>
                <textarea rows="4" placeholder="Enter the address" required="required"
style="color:black" name="address"></textarea>
                <!input type="textarea" name="address" rows="4" placeholder="Enter the
address" required="required" style="color:black" />
          <button type="submit" class="btn btn-primary btn-block btn-large">Submit the
request</button>

   </form>



 <br><br>
<div style="color:black"></div>

 </div>
```

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

app.py

```
from flask import Flask, render_template, request, redirect, url_for, session
import ibm_db
import json
app = Flask(__name__, template_folder="template")
conn =
ibm_db.connect("DATABASE=bludb;HOSTNAME=8e359033-a1c9-4643-82ef-
8ac06f5107eb.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=30120;SECURITY=SSL
;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=msd87692;PWD=sosEU2X4p8M
PTc1e","","")

@app.route('/register')
def register():
    return render_template('register.html')

@app.route("/adddonor",methods = ['POST','GET'])
def adddonor():
    if request.method == 'POST':
        name = request.form['name']
        email = request.form['email']
        phone = request.form['phone']
        city = request.form['city']
        infect = request.form['infect']
        blood = request.form['blood']
        password = request.form['passw']
        sql = "SELECT * FROM DONOR2 WHERE name =?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,name)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)

        if account:
                return render_template('donor.html', msg="You are already a member, please login using your details")
        else:
            #mailtest_donor(email)
            insert_sql = "INSERT INTO DONOR2 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, phone)
            ibm_db.bind_param(prep_stmt, 4, city)
            ibm_db.bind_param(prep_stmt, 5, infect)
            ibm_db.bind_param(prep_stmt, 6, blood)
            ibm_db.bind_param(prep_stmt, 7, password)
            ibm_db.execute(prep_stmt)
        return render_template('success.html', msg="Registered successfuly..")



@app.route('/')
@app.route('/login')
def login():
    return render_template('login.html')

@app.route('/loginpage',methods=['GET','POST'])
def loginpage():
    user = request.form['user']
    passw = request.form['passw']
    sql = "SELECT * FROM user WHERE email =? AND password=?"
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.bind_param(stmt,1,user)
    ibm_db.bind_param(stmt,2,passw)
    ibm_db.execute(stmt)
    account = ibm_db.fetch_assoc(stmt)
    print (account)
    print(user,passw)
    if account:
            return redirect(url_for('stats'))
    else:
        return render_template('login.html', pred="Login unsuccessful. Incorrect username / password !")


@app.route('/stats')
def stats():
    '''sql = "SELECT blood FROM user group by blood"
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.execute(stmt)
    count = ibm_db.fetch_assoc(stmt)
    print(count)'''
    return render_template('stats.html',b=0,b1=0,b2=0,b3=0,b4=0,b5=0,b6=0,b7=0,b8=0)
```

```
@app.route('/requester')
def requester():
    return render_template('request.html')



@app.route('/requested',methods=['GET','POST'])
def requested():
    bloodgrp = request.form['bloodgrp']
    address = request.form['address']
    print(address)
    sql = "SELECT * FROM user WHERE blood=?"
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.bind_param(stmt,1,bloodgrp)
    ibm_db.execute(stmt)
    data = ibm_db.fetch_assoc(stmt)
    msg = "Need Plasma of your blood group for: "+address
    while data != False:
        print ("The Phone is : ", data["PHONE"])

url="https://www.fast2sms.com/dev/bulk?authorization=xCXuwWTzyjOD2ARd1EngbH
3a7tKIq5PklJ8YSf0Lh4FQZecs9iNI1dSvuqprxFwCKYJXA5amQkBE36Rl&sender_id=
FSTSMS&message="+msg+"&language=english&route=p&numbers="+str(data["PHO
NE"])
        result=requests.request("GET",url)
        print(result)
        data = ibm_db.fetch_assoc(stmt)
        return render_template('request.html', pred="Your request is sent to the concerned
people.")
```

```
if __name__ == "__main__":
  app.run(host='0.0.0.0', port=8080,debug=True)
```

style.css

```
@import url(https://fonts.googleapis.com/css?family=Open+Sans);
.btn {
	display: inline-block;
	*display: inline;
	*zoom: 1; padding:
	4px 10px 4px;
	margin-bottom: 0;
	font-size: 13px;
	line-height: 18px;
	color: #333333;
	text-align: center;
	text-shadow: 0 1px 1px rgba(255, 255, 255, 0.75);
	vertical-align: middle;
	background-color: #f5f5f5;
	background-image: -moz-linear-gradient(top, #ffffff, #e6e6e6);
	background-image: -ms-linear-gradient(top, #ffffff, #e6e6e6);
	background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#ffffff), to(#e6e6e6));
	background-image: -webkit-linear-gradient(top, #ffffff, #e6e6e6);
	background-image: -o-linear-gradient(top, #ffffff, #e6e6e6);
	background-image: linear-gradient(top, #ffffff, #e6e6e6);
	background-repeat: repeat-x;
```

```
    filter: progid:dximagetransform.microsoft.gradient(startColorstr=#ffffff, endColorstr=#e6e6e6, GradientType=0);
    border-color: #e6e6e6 #e6e6e6 #e6e6e6;
    border-color: rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.25);
    border: 1px solid #e6e6e6;
    -webkit-border-radius: 4px;
    -moz-border-radius: 4px;
    border-radius: 4px;
    -webkit-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05);
    -moz-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05);
    box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05);
    cursor: pointer; *margin-left: .3em;
    }

.btn:hover, .btn:active, .btn.active, .btn.disabled, .btn[disabled] { background-color: #e6e6e6; }

.btn-large {
    padding: 9px 14px;
    font-size: 15px;
    line-height: normal;
    -webkit-border-radius: 5px;
    -moz-border-radius: 5px;
    border-radius: 5px;
    }

.btn:hover {
	color: #333333;
	text-decoration: none;
	background-color: #e6e6e6;
	background-position: 0 -15px;
	-webkit-transition: background-position 0.1s linear;
	-moz-transition: background-position 0.1s linear;
	-ms-transition: background-position 0.1s linear;
	-o-transition: background-position 0.1s linear;
	transition: background-position 0.1s linear;
	}

.btn-primary, .btn-primary:hover {
	text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25);
	color: #ffffff;
	}

.btn-primary.active { color: rgba(255, 255, 255, 0.75); }

.btn-primary {
	background-color: #4a77d4;
	background-image: -moz-linear-gradient(top, #6eb6de, #4a77d4);
	background-image: -ms-linear-gradient(top, #6eb6de, #4a77d4);
	background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#6eb6de), to(#4a77d4));
	background-image: -webkit-linear-gradient(top, #6eb6de, #4a77d4);
	background-image: -o-linear-gradient(top, #6eb6de, #4a77d4);
	background-image: linear-gradient(top, #6eb6de, #4a77d4);
	background-repeat: repeat-x;
```

```
    filter: progid:dximagetransform.microsoft.gradient(startColorstr=#6eb6de, endColorstr=#4a77d4, GradientType=0);
    border: 1px solid #3762bc;
    text-shadow: 1px 1px 1px rgba(0,0,0,0.4);
    box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.5);
    }

.btn-primary:hover, .btn-primary:active, .btn-primary.active, .btn-primary.disabled, .btn-primary[disabled] {
    filter: none;
    background-color: #4a77d4;
    }

.btn-block { width: 100%; display:block; }

* { -webkit-box-sizing:border-box; -moz-box-sizing:border-box; -ms-box-sizing:border-box; -o-box-sizing:border-box; box-sizing:border-box; }

html { width: 100%; height:100%; overflow:hidden; }

body {
    width: 100%;
    height:100%;
    font-family: 'Open Sans', sans-serif;
    background: #ffffff;
    color: #000000;

    font-size: 18px;
    text-align:center;
```

```
          letter-spacing:1.2px;



}
.header {
                              top:0;
                              margin:0px;
                              left: 0px;
                              right: 0px;
                              position: fixed;
                              background: #4a77d4;
                              color: white;
                              box-shadow: 0px 8px 4px grey;
                              overflow: hidden;
                              padding: 15px;
                              font-size: 1.5vw;
                              width: 100%;
                              text-align: center;
                    }
.login {
          position: absolute;
          top: 70%;
          left: 50%;
          margin: -25px 0 0 -150px;
          width:400px;
          height:400px;
}
```

```
.header div { color: #fff; text-shadow: 0 0 10px rgba(0,0,0,0.3); letter-spacing:1px;
text-align:center; float:left; padding-left:150px;}

ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  padding-right:150px;
  overflow: hidden;
 }



li {
  float: right;
}

li a {
  display: block;
  color: white;
  text-align: center;
  padding: 0px 15px;
  text-decoration: none;
}



input {
        width: 100%;
        margin-bottom: 10px;
        background: rgba(255,255,255,255);
	border: none;
	outline: none;
	padding: 10px;
	font-size: 13px;
	color: black;
	text-shadow: black;
	border: 1px solid rgba(0,0,0,0.3);
	border-radius: 4px;
	box-shadow: inset 0 -5px 45px rgba(100,100,100,0.2), 0 1px 1px rgba(255,255,255,0.2);
	-webkit-transition: box-shadow .5s ease;
	-moz-transition: box-shadow .5s ease;
	-o-transition: box-shadow .5s ease;
	-ms-transition: box-shadow .5s ease;
	transition: box-shadow .5s ease;
}
input:focus { box-shadow: inset 0 -5px 45px rgba(100,100,100,0.4), 0 1px 1px rgba(255,255,255,0.2); }


select {
	width: 100%;
	margin-bottom: 10px;
	background: rgba(255,255,255,255);
	border: none;
	outline: none;
	padding: 10px;
	font-size: 13px;
	color: #000000;
```

```
        text-shadow: 1px 1px 1px rgba(0,0,0,0.3);
        border: 1px solid rgba(0,0,0,0.3);
        border-radius: 4px;
        box-shadow: inset 0 -5px 45px rgba(100,100,100,0.2), 0 1px 1px
rgba(255,255,255,0.2);
        -webkit-transition: box-shadow .5s ease;
        -moz-transition: box-shadow .5s ease;
        -o-transition: box-shadow .5s ease;
        -ms-transition: box-shadow .5s ease;
        transition: box-shadow .5s ease;
}
```

style1.css

```
@import url(https://fonts.googleapis.com/css?family=Open+Sans);
.btn {
        display: inline-block;
        *display: inline;
        *zoom: 1; padding:
        4px 10px 4px;
        margin-bottom: 0;
        font-size: 13px;
        line-height: 18px;
        color: #333333;
        text-align: center;
        text-shadow: 0 1px 1px rgba(255, 255, 255, 0.75);
        vertical-align: middle;
        background-color: #f5f5f5;
        background-image: -moz-linear-gradient(top, #ffffff, #e6e6e6);
        background-image: -ms-linear-gradient(top, #ffffff, #e6e6e6);
```

```
	background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#ffffff), to(#e6e6e6));
	background-image: -webkit-linear-gradient(top, #ffffff, #e6e6e6);
	background-image: -o-linear-gradient(top, #ffffff, #e6e6e6);
	background-image: linear-gradient(top, #ffffff, #e6e6e6);
	background-repeat: repeat-x;
	filter: progid:dximagetransform.microsoft.gradient(startColorstr=#ffffff, endColorstr=#e6e6e6, GradientType=0);
	border-color: #e6e6e6 #e6e6e6 #e6e6e6;
	border-color: rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.25);
	border: 1px solid #e6e6e6;
	-webkit-border-radius: 4px;
	-moz-border-radius: 4px;
	border-radius: 4px;
	-webkit-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05);
	-moz-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05);
	box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05);
	cursor: pointer; *margin-left: .3em;
	}

.btn:hover, .btn:active, .btn.active, .btn.disabled, .btn[disabled] { background-color: #e6e6e6; }

.btn-large {
	padding: 9px 14px;
	font-size: 15px;
	line-height: normal;
	-webkit-border-radius: 5px;
	-moz-border-radius: 5px;
	border-radius: 5px;
	}

.btn:hover {
	color: #333333;
	text-decoration: none;
	background-color: #e6e6e6;
	background-position: 0 -15px;
	-webkit-transition: background-position 0.1s linear;
	-moz-transition: background-position 0.1s linear;
	-ms-transition: background-position 0.1s linear;
	-o-transition: background-position 0.1s linear;
	transition: background-position 0.1s linear;
	}

.btn-primary, .btn-primary:hover {
	text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25);
	color: #ffffff;
	}

.btn-primary.active { color: rgba(255, 255, 255, 0.75); }

.btn-primary {
	background-color: #4a77d4;
	background-image: -moz-linear-gradient(top, #6eb6de, #4a77d4);
	background-image: -ms-linear-gradient(top, #6eb6de, #4a77d4);
```

```
        background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#6eb6de),
to(#4a77d4));
        background-image: -webkit-linear-gradient(top, #6eb6de, #4a77d4);
        background-image: -o-linear-gradient(top, #6eb6de, #4a77d4);
        background-image: linear-gradient(top, #6eb6de, #4a77d4);
        background-repeat: repeat-x;
        filter: progid:dximagetransform.microsoft.gradient(startColorstr=#6eb6de,
endColorstr=#4a77d4, GradientType=0);
        border: 1px solid #3762bc;
        text-shadow: 1px 1px 1px rgba(0,0,0,0.4);
        box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.5);
        }

.btn-primary:hover, .btn-primary:active, .btn-primary.active, .btn-primary.disabled,
.btn-primary[disabled] {
        filter: none;
        background-color: #4a77d4;
        }

.btn-block { width: 100%; display:block; }

* { -webkit-box-sizing:border-box; -moz-box-sizing:border-box;
-ms-box-sizing:border-box; -o-box-sizing:border-box; box-sizing:border-box; }

html { width: 100%; height:100%; overflow:hidden; }

body {
        width: 100%;
        height:100%;
```

```
    font-family: 'Open Sans', sans-serif;
    background: #ffffff;
    color: #000000;

    font-size: 18px;
    text-align:center;
    letter-spacing:1.2px;



}
.header {
            top:0;
            margin:0px;
            left: 0px;
            right: 0px;
            position: fixed;
            background: #4a77d4;
            color: white;
            box-shadow: 0px 8px 4px grey;
            overflow: hidden;
            padding: 15px;
            font-size: 1.5vw;
            width: 100%;
            text-align: center;
        }
.login {
    position: absolute;
    top: 70%;
    left: 50%;
```

```
            margin: -25px 0 0 -150px;
            width:400px;
            height:400px;
}

.header div { color: #fff; text-shadow: 0 0 10px rgba(0,0,0,0.3); letter-spacing:1px;
text-align:center; float:left; padding-left:150px;}

ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  padding-right:150px;
  overflow: hidden;
 }



li {
  float: right;
}

li a {
  display: block;
  color: white;
  text-align: center;
  padding: 0px 15px;
  text-decoration: none;
}
```

```
input {
	width: 100%;
	margin-bottom: 10px;
	background: rgba(255,255,255,255);
	border: none;
	outline: none;
	padding: 10px;
	font-size: 13px;
	color: black;
	text-shadow: black;
	border: 1px solid rgba(0,0,0,0.3);
	border-radius: 4px;
	box-shadow: inset 0 -5px 45px rgba(100,100,100,0.2), 0 1px 1px rgba(255,255,255,0.2);
	-webkit-transition: box-shadow .5s ease;
	-moz-transition: box-shadow .5s ease;
	-o-transition: box-shadow .5s ease;
	-ms-transition: box-shadow .5s ease;
	transition: box-shadow .5s ease;
}
input:focus { box-shadow: inset 0 -5px 45px rgba(100,100,100,0.4), 0 1px 1px rgba(255,255,255,0.2); }


textarea {
	width: 100%;
	margin-bottom: 10px;
	background: rgba(255,255,255,255);
```

```
        border: none;
        outline: none;
        padding: 10px;
        font-size: 13px;
        color: black;
        text-shadow: black;
        border: 1px solid rgba(0,0,0,0.3);
        border-radius: 4px;
        box-shadow: inset 0 -5px 45px rgba(100,100,100,0.2), 0 1px 1px
rgba(255,255,255,0.2);
        -webkit-transition: box-shadow .5s ease;
        -moz-transition: box-shadow .5s ease;
        -o-transition: box-shadow .5s ease;
        -ms-transition: box-shadow .5s ease;
        transition: box-shadow .5s ease;
}
textarea:focus { box-shadow: inset 0 -5px 45px rgba(100,100,100,0.4), 0 1px 1px
rgba(255,255,255,0.2); }

select {
        width: 100%;
        margin-bottom: 10px;
        background: rgba(255,255,255,255);
        border: none;
        outline: none;
        padding: 10px;
        font-size: 13px;
        color: #000000;
        text-shadow: 1px 1px 1px rgba(0,0,0,0.3);
```

```
    border: 1px solid rgba(0,0,0,0.3);
    border-radius: 4px;
    box-shadow: inset 0 -5px 45px rgba(100,100,100,0.2), 0 1px 1px rgba(255,255,255,0.2);
    -webkit-transition: box-shadow .5s ease;
    -moz-transition: box-shadow .5s ease;
    -o-transition: box-shadow .5s ease;
    -ms-transition: box-shadow .5s ease;
    transition: box-shadow .5s ease;
}
```