IBM PROJECT REPORT

TEAM ID PNT2022TMID21617

DOMAIN NAME

CLOUD APP DEVELOPMENT

PROJECT NAME

PLASMA DONOR APPLICATION

TEAM MEMBERS

ARTHIKA K	(142219104001)
DAKSHNAMURTHY S	(142219104018)
DEEPAN SURYA RAJ S V	(142219104021)
DURGA S P	(142219104024)

Github repository link

https://github.com/IBM-EPBL/IBM-Project-11144-1659270573

Demo link

https://drive.google.com/drive/folders/1P235Hv6VA9pHls0mhqb0XGWKLGXwRTer?usp=sharing

CHAPTER -1

INTRODUCTION

1. INTRODUCTION

1.1 Project Overview:

During COVID-19 crisis, the plasma requirement became high and donor count was low because of unawareness of donor blood at the required hospital. So, to track all the donors who can give blood and their blood type can be accessed by this application whenever they need it. The donor information is saved in the database and helping the users by notifying when a person requests for a particular blood type.

With rapid increase in the usage of social networks sites across the world, there is also a steady increase in blood donation requests as being noticed in the number of posts on these sites such as Facebook and twitter seeking blood donors. There are some blood donor finder applications in the market such as Blood app by Red Cross. However, more reliable applications that meet the needs of users are prompted.

We have several pages for various other functionalities. Our pages serve these following properties: Donor Registration Search for a donor, Registration requires an individual to fill up important personal details like email, phone no, etc. and then the record is added to the database. Next Page has provided the facility to look for a particular donor and adds several filters to be performed on our data based on one's requirements.

1.2 Purpose

The initiative is aimed at helping to bridge the gap between the donors and the patients who require blood. Plasma plays the critical role of maintaining a healthy blood pressure, blood volume and a proper pH balance. Without plasma, our body would not be supplied with many of the proteins that are necessary to support blood clotting and our immune system responses. In addition, plasma carries many of the electrolytes that our muscles need to function properly and support our activities of daily living. So, in order to get plasma whenever there is a need for that blood group using this application, we can send an email to all the donors who have that particular blood group. The purpose for plasma rapidly increased during COVID pandemic. This application will easily find outs the donor information to the recipient. By storing the information of the donor and sending the email by providing the information about the recipient by making the communication easier between the donor and recipient.

If a patient needs a plasma at a clinic, plasma donors in vicinity can be contacted through using a clinic management service provided in this application. Registered donors will get notification for the blood requests only if their blood group is compatible with the requested blood type and in the same city/region. Then matching blood donors can go to the requesting clinic and donate.

The purpose for creating this application is to make it hassle-free for the people who are in need of a certain group. It makes it easy for look for a certain blood group. Moreover, it maintains a dataset of all samples collected from different people and in similar way it is utilised by other needy people by making use of search donor functionality which is rather effective and time saver.

CHAPTER -2
LITERATURE SURVEY

2.1 EXISTING PROBLEM

The user must have a device with android operating system with an active internet connection to interact with this application. The user interface can be better than now the user given details are maintained unverified. The android mobile user will not be able to insert or view details if the server goes down. Thus, there is disadvantage of single point failure. The android mobile user will not be able to insert or view details if the server goes down. Thus, there is disadvantage of single point failure. This is system is closed for general plasma donation and mainly focused on COVID-19 patients for plasma donation.

2.2 REFERNCES

[1] Kalpana Devi Guntoju, Tejaswini Jalli, Sreeja Uppala, Sanjay Mallisetti (2022) Instant Plasma Donor Recipient connector web application

[2] S Periyanayagi, A Manikandan,M Muthukrishnan, and M Ramakrishnan (2021) Blood Donation Application using Android Studio

[3] Rishab Chakrabarti, Asha Darade, Neha Jadhav, Prof. S. M. Chitalkar(2020) Lifesaver E-Blood Donation App Using Cloud

[4] M.R. Anish Hamlin, J. Albert (2018) Mayan Blood donation and life saver-blood donation app

[5]Aishwarya R Gowri (2020) Developing a plasma donor application using Function-as-aservice in AWS

[6] Rishab Chakrabarti, Asha Darade, Neha Jadhav, Prof. S. M. Chitalkar2020Lifesaver E-Blood Donation App Using Cloud

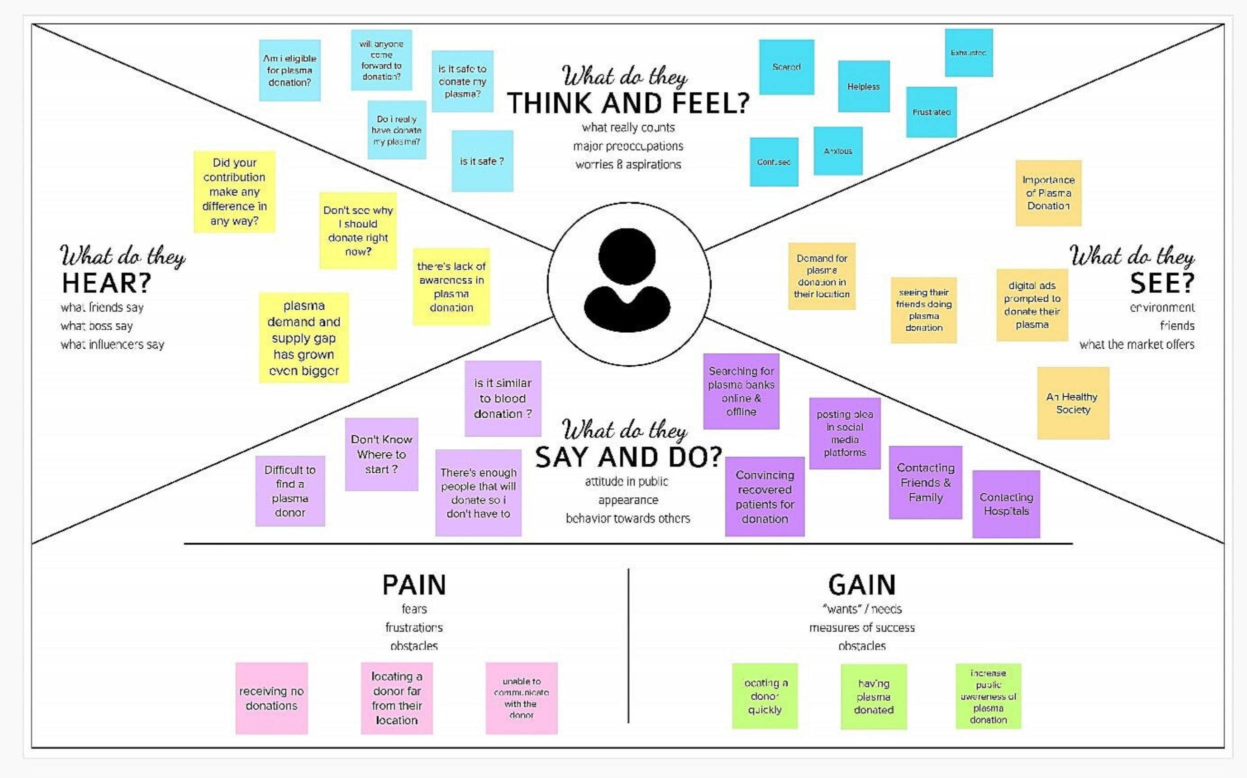
2.3 PROBLEM STATEMENT DEFINITION

The Donor needs to upload their recovered COVID-19 Certificate and it required to verified by the blood bank. It is a user-friendly application. It will help people to find plasma easily. The Donor details are verified before they allow to donate and have to authorised by institution. The Verification and validation are done in email base Reduction in the errors of blood bank using most eligible donor method. Direct Communication Between donor and the person in need of blood During the Emergency situation. The efficient way of finding plasma donor for the infected people. Aws lambda function is used and to deploy the application AWS EC2 service is used. When the giver gives the blood, it will naturally evacuate the contributor detail for next three months. It additionally confirms with the Department of Health and Welfare to guarantee the benefactor medical case history. Accessibility and availability are the criteria on which an application is designed for its success in the IT market. This application helps health care centres to provide the blood as quick as possible when their stocks are insufficient. The application sends periodically actual location information of available donors to main system and the blood requests to the donors.

CHAPTER -3

IDEATION AND PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS


What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
Am i eligible for plasma donation?
will anyone come forward to donation?
is it safe to donate my plasma?
Do i really have donate my plasma?
is it safe ?
Scared
Helpless
Exhausted
Frustrated
Confused
Anxious
What do they HEAR?
what friends say
what boss say
what influencers say
Did your contribution make any difference in any way?
Don't see why I should donate right now?
there's lack of awareness in plasma donation
plasma demand and supply gap has grown even bigger
What do they SEE?
environment
friends
what the market offers
Importance of Plasma Donation
Demand for plasma donation in their location
seeing their friends doing plasma donation
digital ads prompted to donate their plasma
An Healthy Society
What do they SAY AND DO?
attitude in public
appearance
behavior towards others
is it similar to blood donation ?
Don't Know Where to start ?
Difficult to find a plasma donor
There's enough people that will donate so i don't have to
Searching for plasma banks online & offline
posting plea in social media platforms
Convincing recovered patients for donation
Contacting Friends & Family
Contacting Hospitals
PAIN
fears
frustrations
obstacles
receiving no donations
locating a donor far from their location
unable to communicate with the donor
GAIN
"wants" / needs
measures of success
obstacles
ocating a donor quickly
having plasma donated
increase public awareness of plasma donation

3.2 IDEATION & BRAINSTORMING

https://github.com/IBM-EPBL/IBM-Project-11144-1659270573/blob/main/Pre-Development/Ideation%20phase/Ideation_Plasma_Donor_Application.pdf

3.3 PROPOSED SOLUTION

Many major medical conditions are treated by plasma. For this reason, blood drives are held to solicit donations of plasma and blood. One of the most wellknown techniques known as plasma treatment, plasma is used to cure various incurable diseases. Due to the lack of vaccines available to treat patients who had been exposed to the coronavirus, the need for plasma increased significantly. Plasma therapy had a high success rate but a very low donor count during these times, thus it was crucial to learn more about the donors. It would be helpful to save the contributor information and let clients know about the recurring donors because it can help them find the crucial information more quickly. This system's goal is to use an online application to link donors and patients. The users of this application can post requests for plasma donations or requirements.

The fundamental solution is to establish a centralised system to keep track of current and previous Plasma Donation Events. The suggested remedy is as follows: This application has two roles:

- User
- Admin

User:

Users must register with their personal information in order to give or receive. Following a user's successful registration.

- The user receives an email upon successful registration.
- The user will be taken to the home page after a successful registration. If they want to be a donor or a recipient, they must press.
- If the user is a donor, they must fill out the donation interest form with their name, blood group information, location, the date they last contributed, a phone number, and an email address.
- The donor will be routed to a page where they may get the E-Certificate after completing the donation form.
- If the user is a recipient, he or she can raise a request and get in touch with the donor immediately after viewing the list of possible donors.

Admin:

- With their login information, admin can log in.
- The request may be amended by admin.
- The request may be revoked by admin.
- Volunteers can be added by admin.

Users can easily grasp a user interface. The application is available anytime, anywhere. The user can use this application to raise a request and directly contact the donor to ask them to donate the plasma if they urgently need it for their treatment but the plasma is not available in the nearby hospitals. Hospitals may also put out a call for donors. Someone who wishes to donate blood and plasma but is unsure how to do so uses this programme, which is easy to use and will help save many lives. Nowadays, a lot of them have smartphones on which they can download this programme and use it to save lives.

3.4 PROBLEM SOLUTION FIT

Anyone above the age of 21 can donate We working on plasma therapy is process where blood is donated and received. You can donate plasma every 28 days, up to 13 times per year. While the FDA does not allow donors to give plasma more frequently. Limited no of users can use it at the same time. It allows people to help others. It is a relatively safe process.The process can be very uncomfortable and It depletes the calcium levels in the body. Localized allergic reaction Air embolism and Hemolysis. Bruising and discomfort. The side effects of plasma donation include nausea and dizziness and fainting in some cases. You may develop a raised bump or experience continued bleeding and bruising at the needle site too. Some people might experience pain and physical weakness after donating plasma. This app is used to make donation and receiving process easier so that anyone can easily access and use it. Intensity of this application is to connect donor and receiver in single platform donor can fill the interest form to donate. Many people needs plasma for their treatment. Plasma donation really used for covid affected people for recovering faster. Donor get fear,anxiety prior to donation give way to largely positive emotional states like relaxation following donation our app allows the user to request and donate plasma to requested person. Receiver can directly contact the donor and receive plasma. When you donate plasma, the blood that's drawn from your arm goes through a special machine to separate the different parts of your blood. Then we get plasma which can be used for transfusion. Application is to connect donor and receiver in single platform. donor can fill the interest form to donate. Online app allows user to make donation and receiver process easier send request from anywhere anytime users to visit nearby camp or hospital and donate as well as receive plasma.

CHAPTER -4
REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Access Website	Software operator should be capable to access web- application through either an application browser or similar on the pc.
FR-2	Software operator Registration	The software operator should be able to register through the web-application. The donor software operator must provide user name, gender, blood/plasma group, location, contact.
FR-3	Login/logout/update details	The login information will be stored on the database for future use.
FR-4	Search for donor	Each element in the list represents a specific donor with the donor details.
FR-5	User plasma request	Users can request to donate plasma by filling out the request form on the page. Once the request is submitted, they will get an email.
FR-6	View distribution details	The plasma bank should be able to view the status of the distribution details.

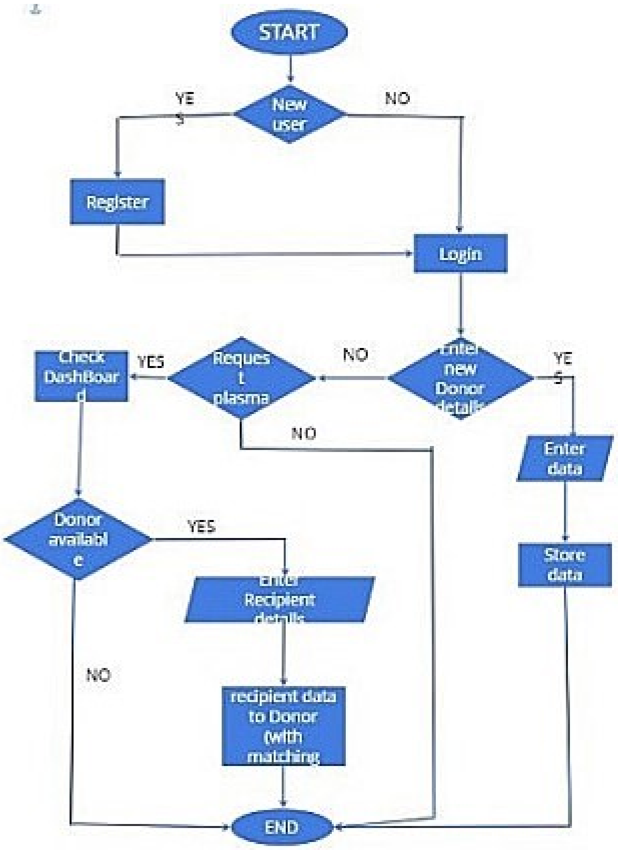
4.2 NON-FUNCTIONAL REQUIREMENTS

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The plasma donor application must have a good looking user friendly interface.
NFR-2	Security	The plasma donor application must be secured with proper username and passwords.
NFR-3	Reliability	The plasma donor application should work properly,even when faults occur.
NFR-4	Performance	The plasma donor application must perform well in different scenarios.
NFR-5	Availability	The plasma donor application must be available 24 hours a day with no bandwidth issues.
NFR-6	Scalability	The plasma donor application should able to increase or decrease in performance and cost in response to changes in application and system processing demands.

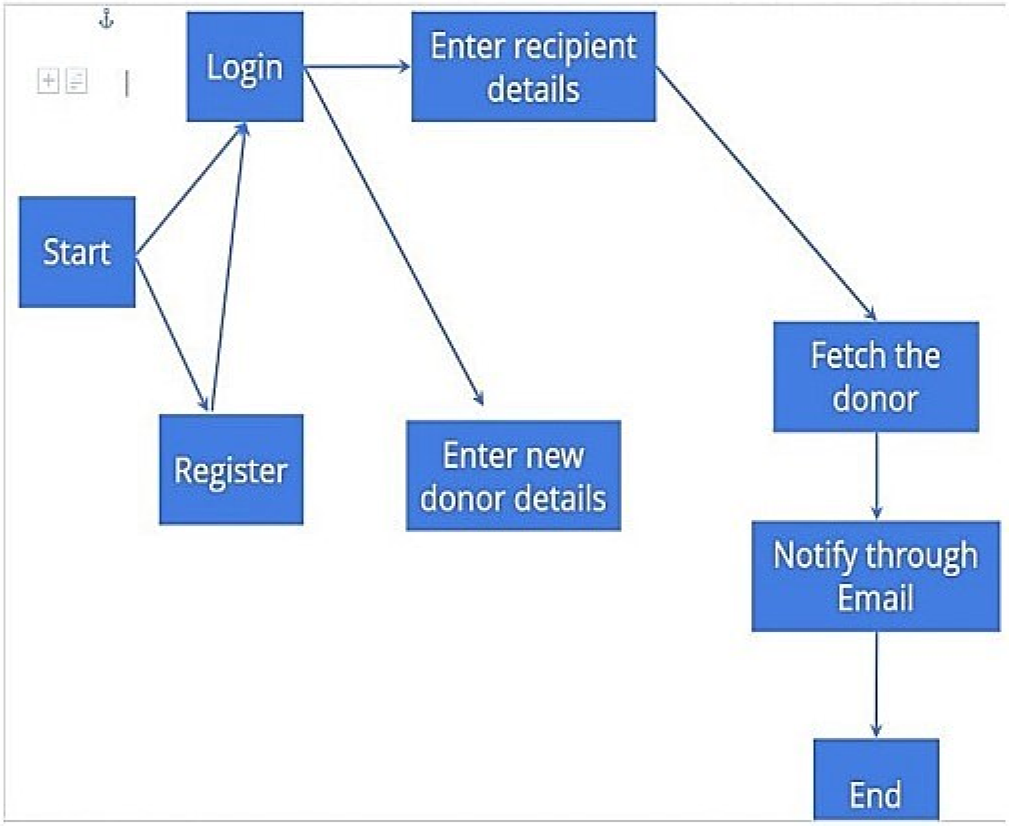
CHAPTER -5

PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS



5.2 SOLUTION & TECHNICAL ARCHITECTURE



5.2 SOLUTION & TECHNICAL ARCHITECTURE

SN O	Component Description	Description	Technology
1	User Interface	The interaction between the user and application e.g., Web UI, Mobile App, Chatbot	HTML, CSS, JavaScript / Bootstrap etc.
2	Application Logic-1	Framework used for designing the application.	Python, Python - Flask

3	Application Logic-2	Accessing the cloud and storing details of the users both donors and patients.	IBM Cloud, IBM DB2
4	Application Logic-3	Docker is an open-source platform for building, deploying, and managing containerized application	Docker
5	Database	Data Type, Configurations etc.	SQL.
6	Cloud Database	Database Service on Cloud	BM Cloud and IBM DB2
7	File Storage	File storage requirements	IBM Block Storage or NO Storage Service or Local File System

5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register 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 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 Facebook	I can register & access the dashboard	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can receive confirmation 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 all details about plasma application	I can donate/get details about the plasma	High	Sprint-2
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
Administrator	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 plasma in application	I can alert notification through email	High	Sprint-2
Administrator	Performance of applicatio n	USN-12	To make the process more efficient	I can save time, cost by improving the Plasma management application	High	Sprint-4

CHAPTER -6

PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

As a user, I can register in the donor application by entering my name, phone_no, Email id, blood group ,aadhar noAs a admin, I can log into the application by entering email & passwordAs a user I can ask query in chatbot. As a user, I can receive confirmation mail. As a user, I can view dashboard and selectAs a user, I can view all the donor list and contact them directlyAs a user, I can search for the donorAs a User, I can view the about us page which contains all contact informationAs a admin, I can modify the User data. As a user, I can send mail to donors using sendgrid. As a user I can view the home page and select the desired option. As a user I can ask my query through emailAs a admin I can download the user data

Sprint-1

- Donor
- Registration
- Login
- Chatbot

Sprint -2

- Confimation
- Dashboard
- View Donor List
- Search Donor

Sprint-3

- About us
- Modify data
- Send mail
- Home page

Sprint -4

- Send Query
- Download data

6.2 SPRINT DELIVERY SCHEDULE

Velocity

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

Sprint	**Sprint Start Date**	**Sprint End Date (Planned)**
Sprint-1	24 Oct 2022	29 Oct 2022
Sprint-2	31 Oct 2022	05 Nov 2022
Sprint-3	07 Nov 2022	12 Nov 2022
Sprint-4	14 Nov 2022	19 Nov 2022

CHAPTER -7

CODING AND SOLUTION

7.1 FEATURE 1

https://github.com/IBM-EPBL/IBM-Project-11144-1659270573/tree/main/Project%20Development%20Phase/sprint%201

7.2 FEATURE 2

https://github.com/IBM-EPBL/IBM-Project-11144-1659270573/tree/main/Project%20Development%20Phase/sprint%202

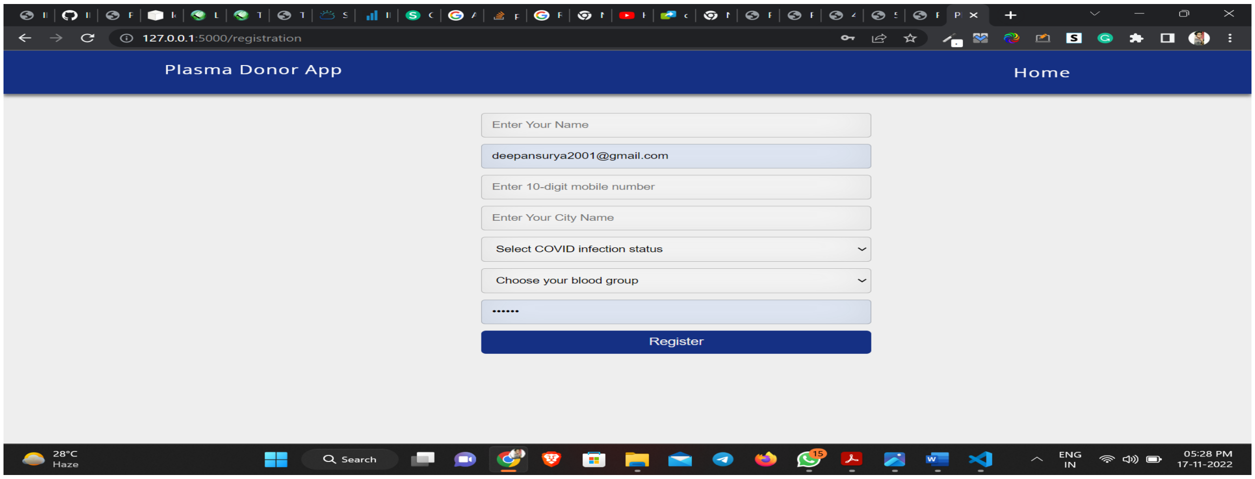
7.3 FEATURE 3

https://github.com/IBM-EPBL/IBM-Project-11144-1659270573/tree/main/Project%20Development%20Phase/sprint%203

7.4 FEATURE 4

https://github.com/IBM-EPBL/IBM-Project-11144-1659270573/tree/main/Project%20Development%20Phase/sprint%204

7.5 OUTPUT:



127.0.0.1:5000/registration

Plasma Donor App

Home

Enter Your Name

deepansurya2001@gmail.com

Enter 10-digit mobile number

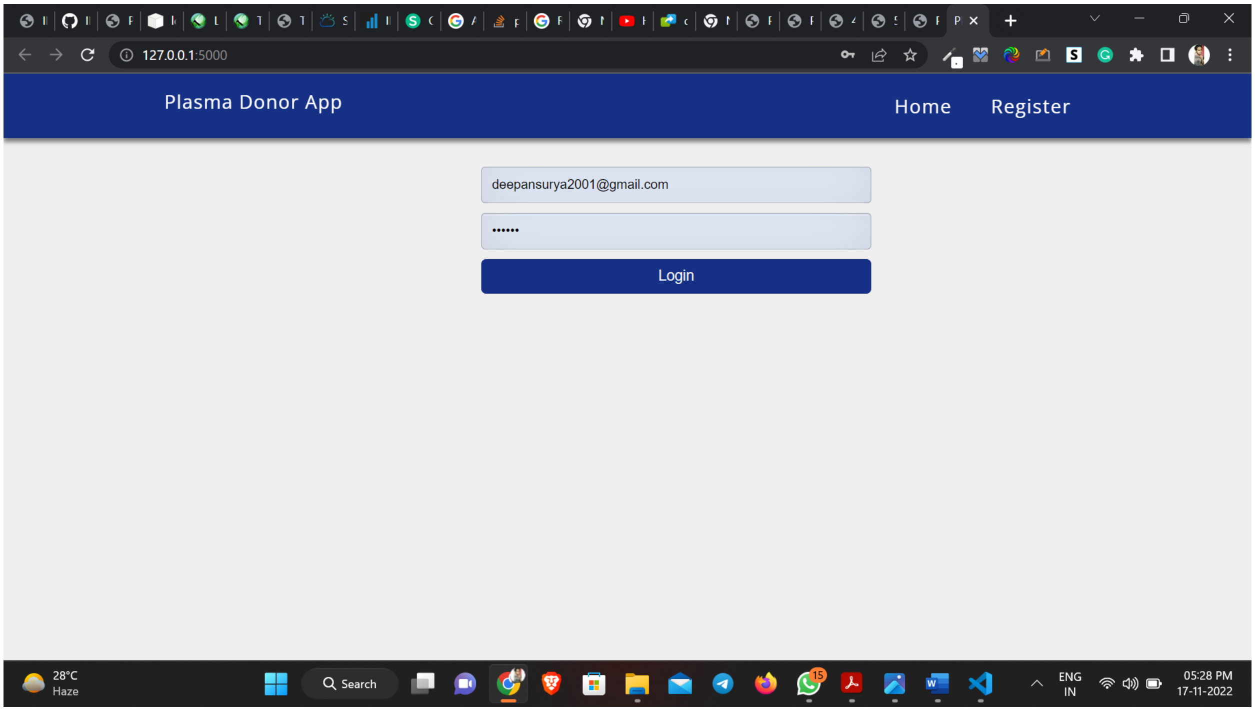
Enter Your City Name

Select COVID infection status

Choose your blood group

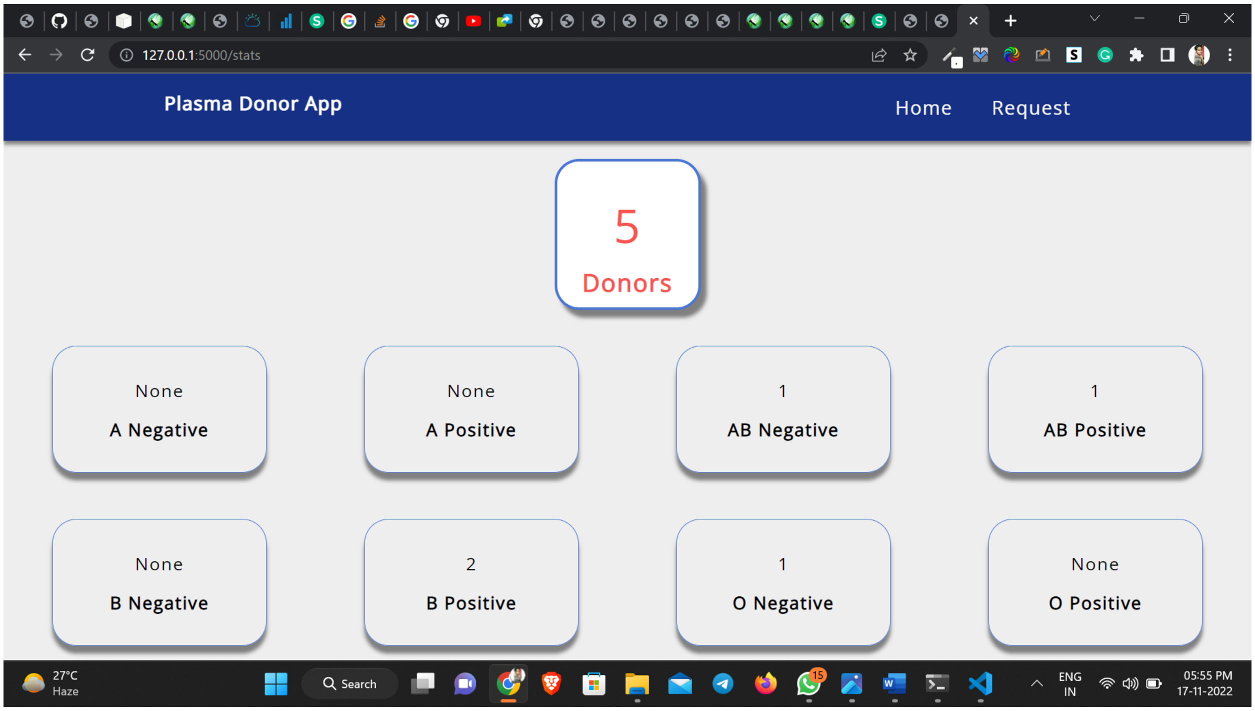
••••••

Register



deepansurya2001@gmail.com

••••••



5

Donors

None

A Negative

None

A Positive

1

AB Negative

1

AB Positive

None

B Negative

2

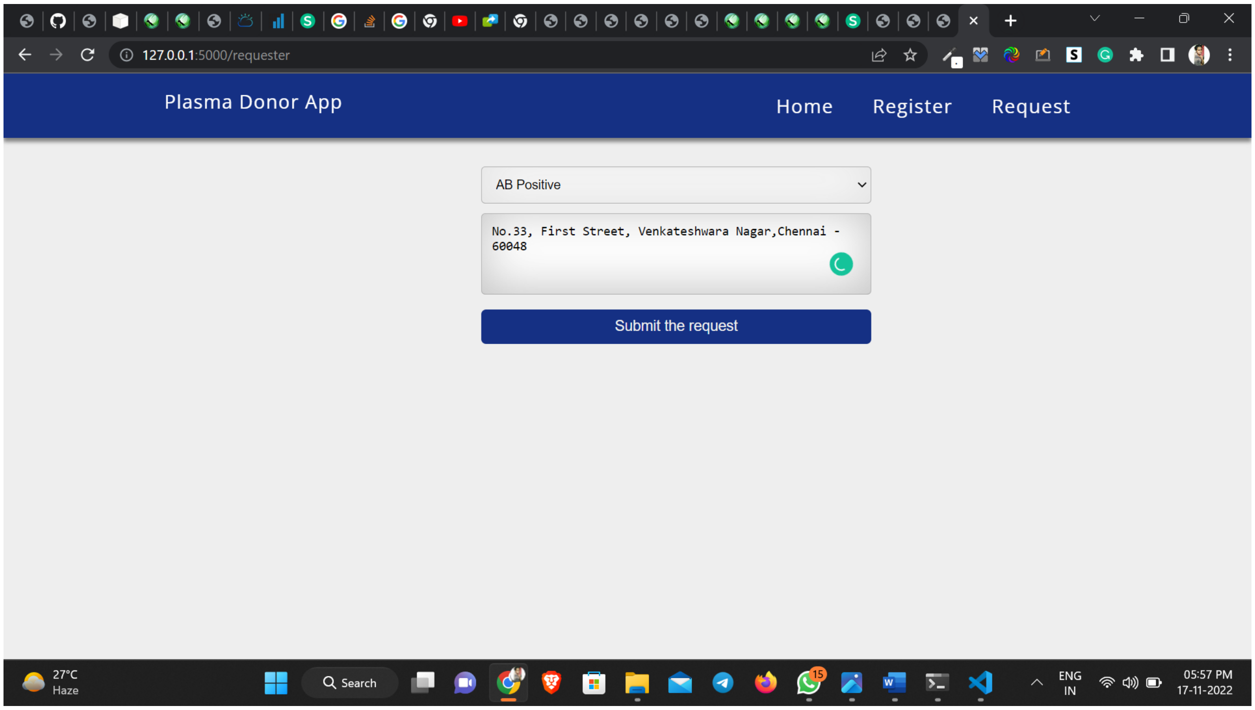
B Positive

1

O Negative

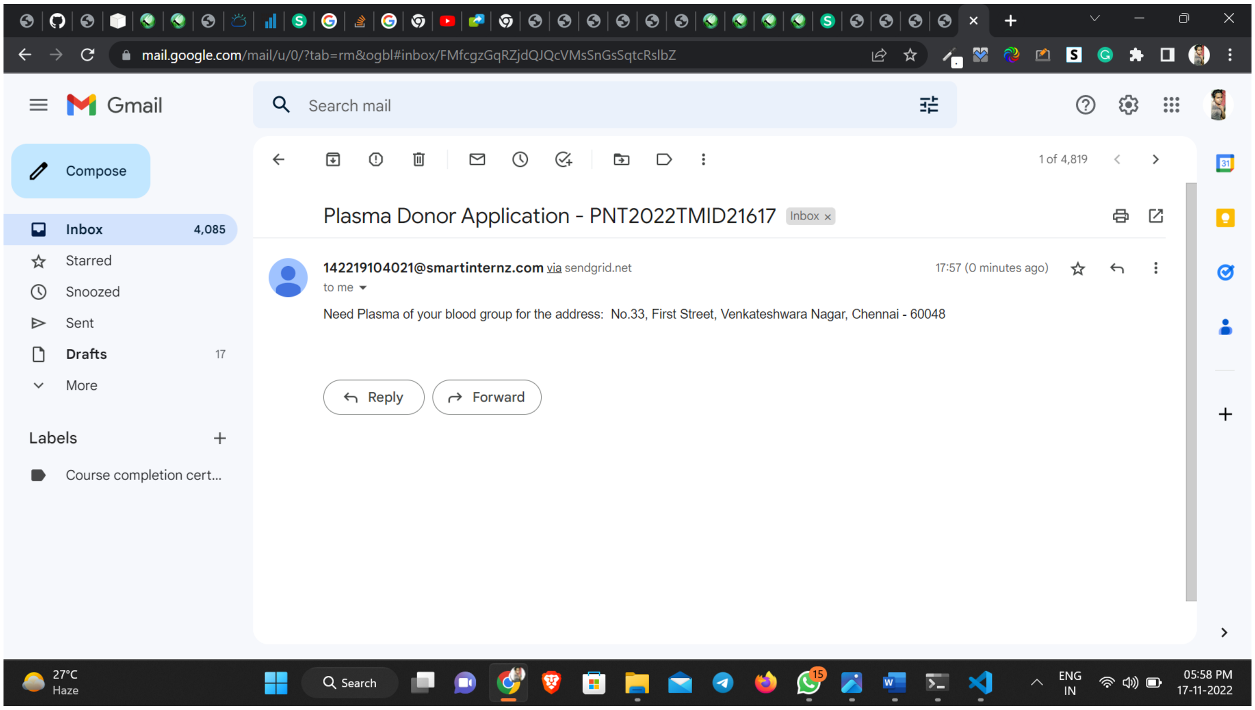
None

O Positive



AB Positive

```
No.33, First Street, Venkateshwara Nagar,Chennai -
60048
```



Plasma Donor Application - PNT2022TMID21617

142219104021@smartinternz.com via sendgrid.net 17:57 (0 minutes ago)
to me

Need Plasma of your blood group for the address: No.33, First Street, Venkateshwara Nagar, Chennai - 60048

CHAPTER -8

TESTING

8.1 TEST CASES

TEST CASES FOR PERFOMANCE TESTING

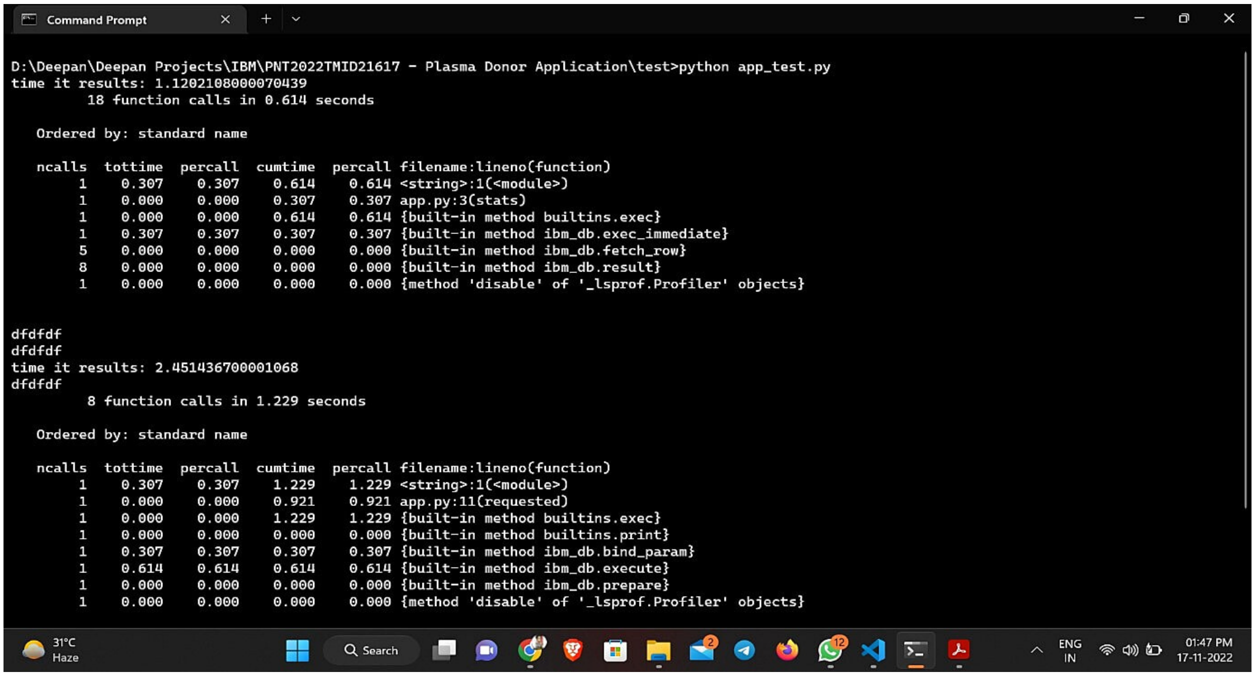
CODE TESTING-TEST CASES

```
import line_profiler
import app
import timeit
import cProfile
import time

print("time it results:", timeit.Timer(app.stats).timeit(number=2))
cProfile.runctx('app.stats()',globals(),locals(),filename=None)

# print("time it results:", timeit.Timer(app.requested).timeit(number=2))
# cProfile.runctx('app.requested()',globals(),locals(),filename=None)
```

PERFORMANCE TESTING - II

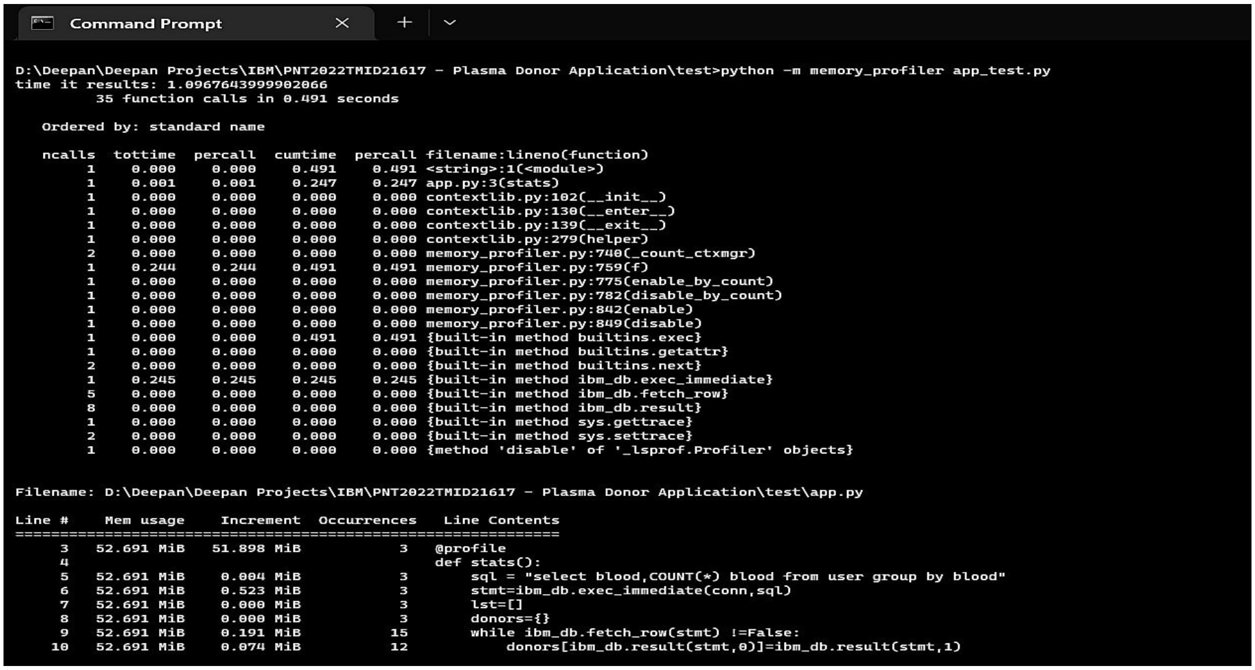
CODE TESTING

TESTING USING CPROFILE AND TIMEIT



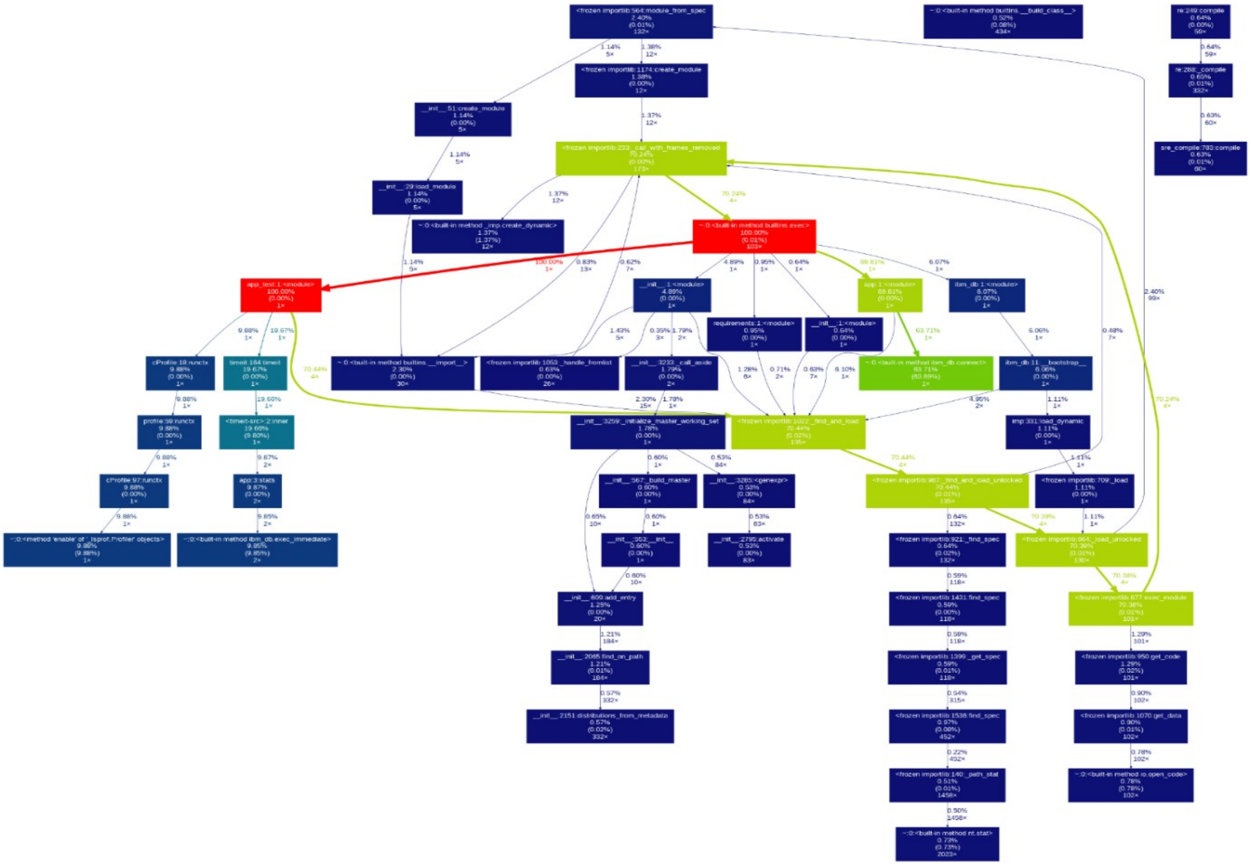
```
D:\Deepan\Deepan Projects\IBM\PNT2022TMID21617 - Plasma Donor Application\test>python app_test.py
time it results: 1.1202108000070439
         18 function calls in 0.614 seconds

   Ordered by: standard name

   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
        1    0.307    0.307    0.614    0.614 <string>:1(<module>)
        1    0.000    0.000    0.307    0.307 app.py:3(stats)
        1    0.000    0.000    0.614    0.614 {built-in method builtins.exec}
        1    0.307    0.307    0.307    0.307 {built-in method ibm_db.exec_immediate}
        5    0.000    0.000    0.000    0.000 {built-in method ibm_db.fetch_row}
        8    0.000    0.000    0.000    0.000 {built-in method ibm_db.result}
        1    0.000    0.000    0.000    0.000 {method 'disable' of '_lsprof.Profiler' objects}


dfdfdf
dfdfdf
time it results: 2.451436700001068
dfdfdf
         8 function calls in 1.229 seconds

   Ordered by: standard name

   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
        1    0.307    0.307    1.229    1.229 <string>:1(<module>)
        1    0.000    0.000    0.921    0.921 app.py:11(requested)
        1    0.000    0.000    1.229    1.229 {built-in method builtins.exec}
        1    0.000    0.000    0.000    0.000 {built-in method builtins.print}
        1    0.307    0.307    0.307    0.307 {built-in method ibm_db.bind_param}
        1    0.614    0.614    0.614    0.614 {built-in method ibm_db.execute}
        1    0.000    0.000    0.000    0.000 {built-in method ibm_db.prepare}
        1    0.000    0.000    0.000    0.000 {method 'disable' of '_lsprof.Profiler' objects}
```

TESTING USING MPROFILE



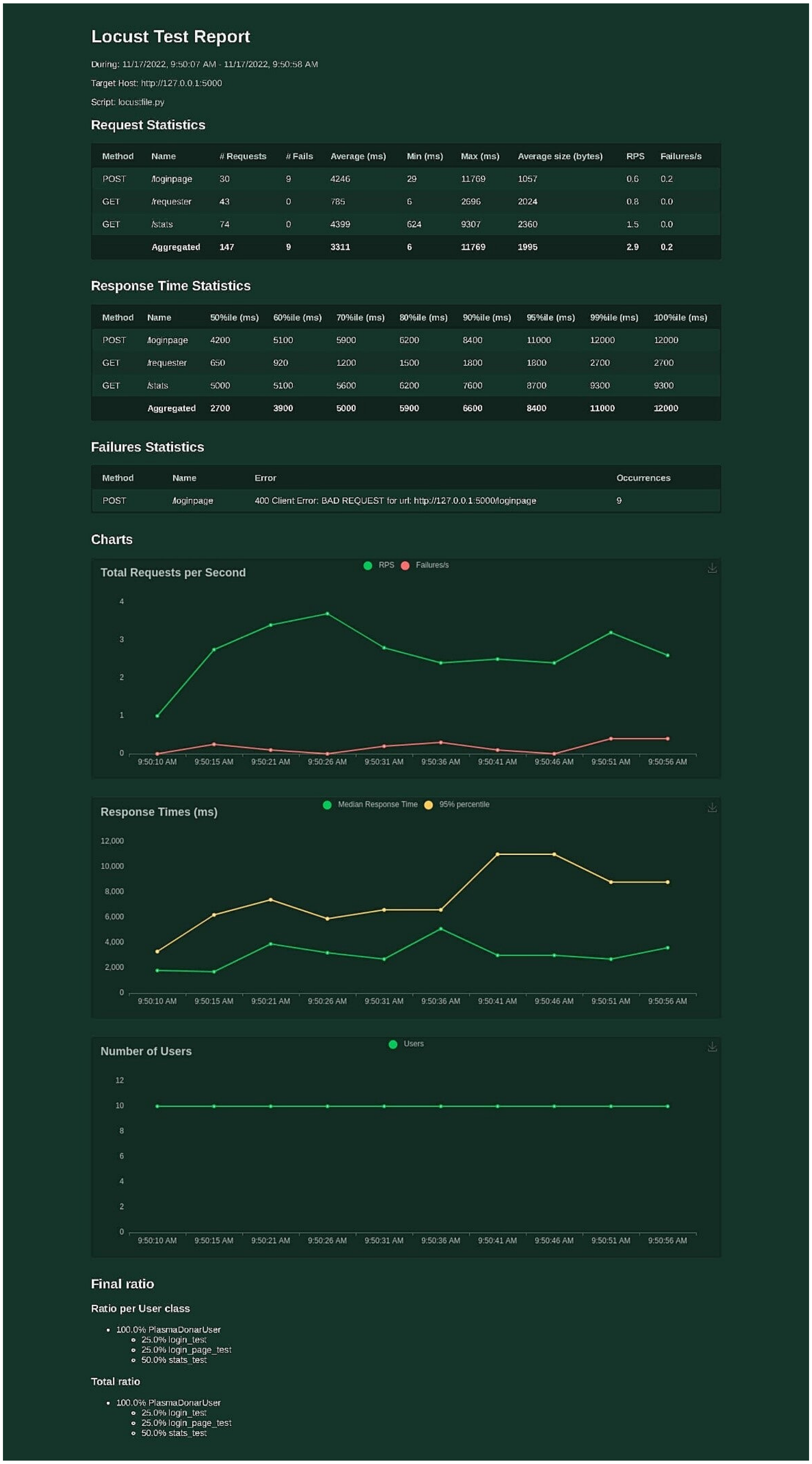
```
D:\Deepan\Deepan Projects\IBM\PNT2022TMID21617 - Plasma Donor Application\test>python -m memory_profiler app_test.py
time it results: 1.0967643999902066
         35 function calls in 0.491 seconds

   Ordered by: standard name

   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
        1    0.000    0.000    0.491    0.491 <string>:1(<module>)
        1    0.001    0.001    0.247    0.247 app.py:3(stats)
        1    0.000    0.000    0.000    0.000 contextlib.py:102(__init__)
        1    0.000    0.000    0.000    0.000 contextlib.py:130(__enter__)
        1    0.000    0.000    0.000    0.000 contextlib.py:139(__exit__)
        1    0.000    0.000    0.000    0.000 contextlib.py:279(helper)
        2    0.000    0.000    0.000    0.000 memory_profiler.py:740(_count_ctxmgr)
        1    0.244    0.244    0.491    0.491 memory_profiler.py:759(f)
        1    0.000    0.000    0.000    0.000 memory_profiler.py:775(enable_by_count)
        1    0.000    0.000    0.000    0.000 memory_profiler.py:782(disable_by_count)
        1    0.000    0.000    0.000    0.000 memory_profiler.py:842(enable)
        1    0.000    0.000    0.000    0.000 memory_profiler.py:849(disable)
        1    0.000    0.000    0.491    0.491 {built-in method builtins.exec}
        1    0.000    0.000    0.000    0.000 {built-in method builtins.getattr}
        2    0.000    0.000    0.000    0.000 {built-in method builtins.next}
        1    0.245    0.245    0.245    0.245 {built-in method ibm_db.exec_immediate}
        5    0.000    0.000    0.000    0.000 {built-in method ibm_db.fetch_row}
        8    0.000    0.000    0.000    0.000 {built-in method ibm_db.result}
        1    0.000    0.000    0.000    0.000 {built-in method sys.gettrace}
        2    0.000    0.000    0.000    0.000 {built-in method sys.settrace}
        1    0.000    0.000    0.000    0.000 {method 'disable' of '_lsprof.Profiler' objects}


Filename: D:\Deepan\Deepan Projects\IBM\PNT2022TMID21617 - Plasma Donor Application\test\app.py

Line #    Mem usage    Increment  Occurrences   Line Contents
=============================================================
     3   52.691 MiB   51.898 MiB           3   @profile
     4                                         def stats():
     5   52.691 MiB    0.004 MiB           3       sql = "select blood,COUNT(*) blood from user group by blood"
     6   52.691 MiB    0.523 MiB           3       stmt=ibm_db.exec_immediate(conn,sql)
     7   52.691 MiB    0.000 MiB           3       lst=[]
     8   52.691 MiB    0.000 MiB           3       donors={}
     9   52.691 MiB    0.191 MiB          15       while ibm_db.fetch_row(stmt) !=False:
    10   52.691 MiB    0.074 MiB          12           donors[ibm_db.result(stmt,0)]=ibm_db.result(stmt,1)
```

GENERATING GRAPH


<frozen importlib:564:module_from_spec
~:0:<built-in method builtins.__build_class__>
re:249:compile
<frozen importlib:1174:create_module
re:288:_compile
__init__:51:create_module
sre_compile:783:compile
<frozen importlib:233:_call_with_frames_removed
__init__:29:load_module
~:0:<built-in method _imp.create_dynamic>
~:0:<built-in method builtins.exec>
app_test:1:<module>
__init__:1:<module>
app:1:<module>
ibm_db:1:<module>
requirements:1:<module>
__init__:1:<module>
cProfile:18:runctx
timeit:164:timeit
~:0:<built-in method builtins.__import__>
<frozen importlib:1053:_handle_fromlist
__init__:3233:_call_aside
~:0:<built-in method ibm_db.connect>
ibm_db:11:__bootstrap__
profile:59:runctx
<timeit-src>:2:inner
__init__:3259:_initialize_master_working_set
<frozen importlib:1022:_find_and_load
imp:331:load_dynamic
cProfile:97:runctx
app:3:stats
__init__:567:_build_master
__init__:3265:<genexpr>
<frozen importlib:987:_find_and_load_unlocked
<frozen importlib:709:_load
~:0:<method 'enable' of '_lsprof.Profiler' objects>
~:0:<built-in method ibm_db.exec_immediate>
__init__:553:__init__
__init__:2795:activate
<frozen importlib:921:_find_spec
<frozen importlib:664:_load_unlocked
__init__:609:add_entry
<frozen importlib:1431:find_spec
<frozen importlib:677:exec_module
__init__:2065:find_on_path
<frozen importlib:1399:_get_spec
<frozen importlib:950:get_code
__init__:2151:distributions_from_metadata
<frozen importlib:1536:find_spec
<frozen importlib:1070:get_data
<frozen importlib:140:_path_stat
~:0:<built-in method io.open_code>
~:0:<built-in method nt.stat>

TEST CASES FOR APPLICATION TESTING

```
from locust import HttpUser, task
import random
import time
data = ({'user': ' deepansuryarajsv@gmail.com' , 'passw': 'deepan'}, { 'name' : ' yuvankaargilrajsv@gmail.com' , 'passw' : 'yuvan'}, {'user' : ' rajasaro2001@rediffmail.com' , 'passw' : 'deepa'})
post_headers={'Content-Type': 'application/x-www-form-urlencoded'}

class PlasmaDonarUser(HttpUser):

    @task(10)
    def login_test(self):
        self.client.get("/requester")


    # @task(10)
    # def login_page_test(self):
    #     self.client.post("/loginpage",data=data[random.randint(0,2)], headers=post_headers)

    # @task(20)
    # def stats_test(self):
    #     self.client.get("/stats")

    @task(10)
    def stats_test(self):
        self.client.get("/registration")

    @task(20)
    def stats_test(self):
        self.client.get("/requested")
```



Locust Test Report

During: 11/17/2022, 9:50:07 AM - 11/17/2022, 9:50:58 AM

Target Host: http://127.0.0.1:5000

Script: locustfile.py

Request Statistics

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/loginpage	30	9	4246	29	11769	1057	0.6	0.2
GET	/requester	43	0	785	6	2696	2024	0.8	0.0
GET	/stats	74	0	4399	624	9307	2360	1.5	0.0
	Aggregated	**147**	**9**	**3311**	**6**	**11769**	**1995**	**2.9**	**0.2**

Response Time Statistics

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/loginpage	4200	5100	5900	6200	8400	11000	12000	12000
GET	/requester	650	920	1200	1500	1800	1800	2700	2700
GET	/stats	5000	5100	5600	6200	7600	8700	9300	9300
	Aggregated	**2700**	**3900**	**5000**	**5900**	**6600**	**8400**	**11000**	**12000**

Failures Statistics

Method	Name	Error	Occurrences
POST	/loginpage	400 Client Error: BAD REQUEST for url: http://127.0.0.1:5000/loginpage	9

Charts

Total Requests per Second

Response Times (ms)

Number of Users

Final ratio

Ratio per User class

- 100.0% PlasmaDonarUser
  - 25.0% login_test
  - 25.0% login_page_test
  - 50.0% stats_test

Total ratio

- 100.0% PlasmaDonarUser
  - 25.0% login_test
  - 25.0% login_page_test
  - 50.0% stats_test

8.3 USER ACCEPTANCE TESTING

1. Purpose of Document

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

2. Defect Analysis

This report showsthe 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	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

3. Test Case Analysis

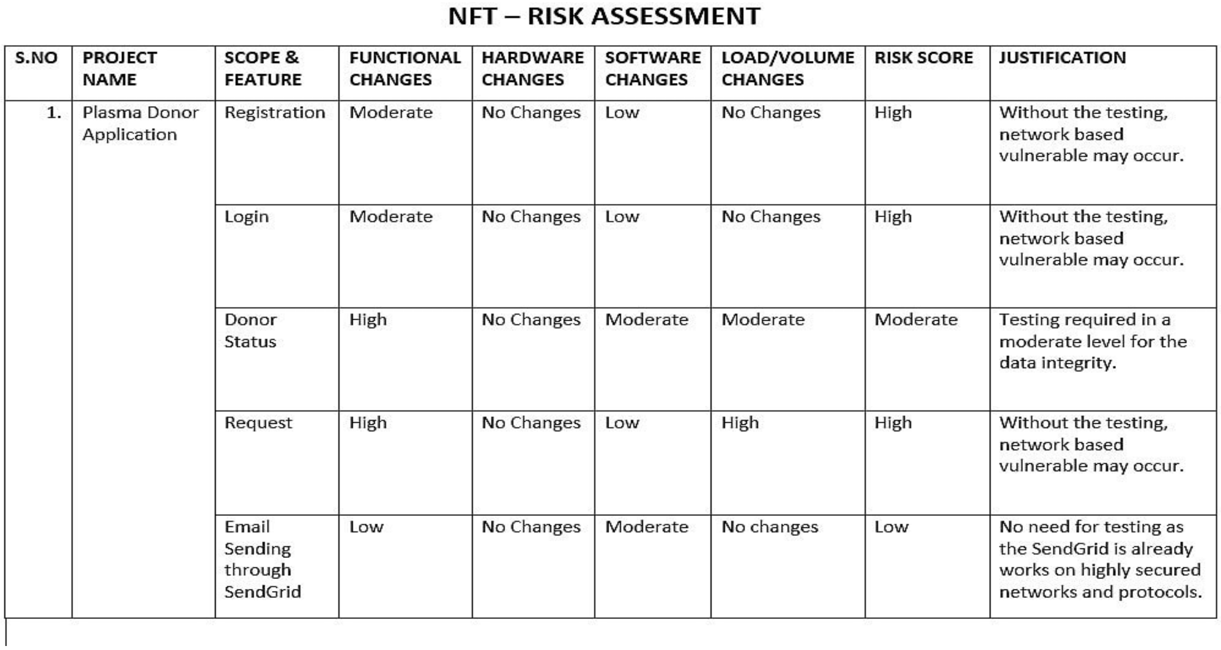
This report shows the number of test cases that have passed, failed,and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	51	0	0	51
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

CHAPTER -9

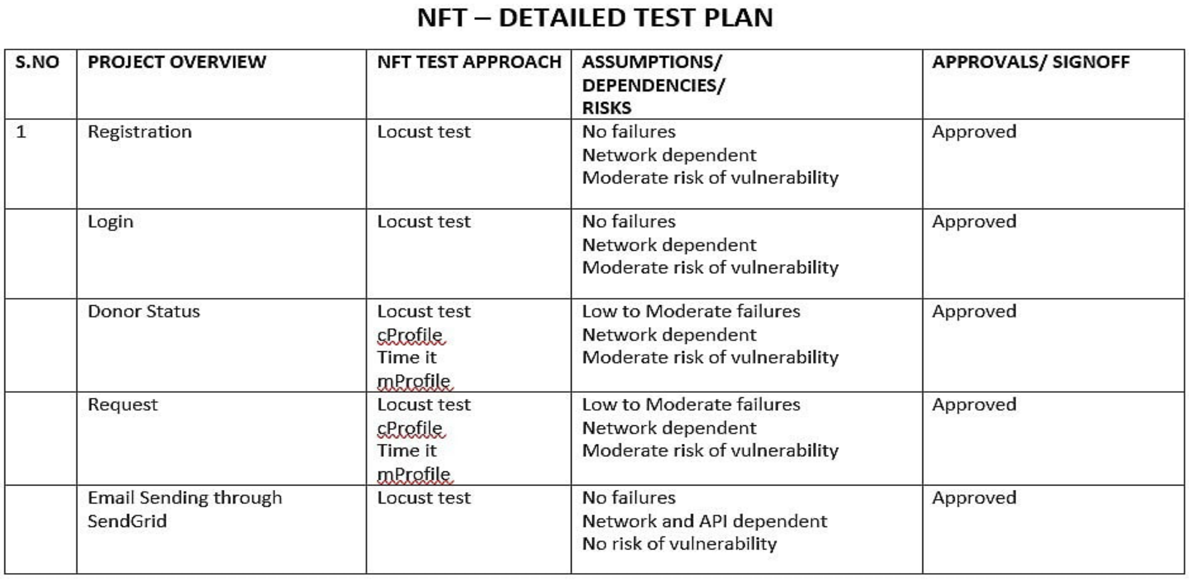
RESULTS

9.1 PERFORMANCE METRICS



NFT – RISK ASSESSMENT

S.NO	PROJECT NAME	SCOPE & FEATURE	FUNCTIONAL CHANGES	HARDWARE CHANGES	SOFTWARE CHANGES	LOAD/VOLUME CHANGES	RISK SCORE	JUSTIFICATION
1.	Plasma Donor Application	Registration	Moderate	No Changes	Low	No Changes	High	Without the testing, network based vulnerable may occur.
		Login	Moderate	No Changes	Low	No Changes	High	Without the testing, network based vulnerable may occur.
		Donor Status	High	No Changes	Moderate	Moderate	Moderate	Testing required in a moderate level for the data integrity.
		Request	High	No Changes	Low	High	High	Without the testing, network based vulnerable may occur.
		Email Sending through SendGrid	Low	No Changes	Moderate	No changes	Low	No need for testing as the SendGrid is already works on highly secured networks and protocols.



NFT – DETAILED TEST PLAN

S.NO	PROJECT OVERVIEW	NFT TEST APPROACH	ASSUMPTIONS/ DEPENDENCIES/ RISKS	APPROVALS/ SIGNOFF
1	Registration	Locust test	No failures Network dependent Moderate risk of vulnerability	Approved
	Login	Locust test	No failures Network dependent Moderate risk of vulnerability	Approved
	Donor Status	Locust test cProfile Time it mProfile	Low to Moderate failures Network dependent Moderate risk of vulnerability	Approved
	Request	Locust test cProfile Time it mProfile	Low to Moderate failures Network dependent Moderate risk of vulnerability	Approved
	Email Sending through SendGrid	Locust test	No failures Network and API dependent No risk of vulnerability	Approved

END OF TEST REPORT

S.NO	PROJECT OVERVIEW	NFT TEST APPROACH	NFR - MET	TEST OUTCOME	GO/NO GO DECISION	RECOMMEDATIONS	IDENTIFIED DEFECTS	APPROVALS/SIGNOFF
1.	Registration	Locust test	Met	Very Good	GO	Improved Security	Open	Approved
	Login	Locust test	Met	Very Good	GO	Improved Security	Open	Approved
	Donor Status	Locust test cProfile Time it mProfile	Met upto 80%	Good	GO	Improved Security Improved Performance Reduced latency	Closed	Approved
	Request	Locust test cProfile Time it mProfile	Met upto 80%	Good	GO	Improved Security Uninterrupted connection	Closed	Approved
	Email Sending through SendGrid	Locust test	Met	Excellent	GO	No comments	Open	Approved

CHAPTER -10

ADVANTAGES AND DISADVANTAGES

10.1 APPLICATIONS

Donate Plasma application provides a reliable platform to connect local Plasma donors with patients. Donate Plasma creates a communication channel through authenticated clinics whenever a patient needs plasma. It can be a useful tool to find compatible plasma donors who can receive plasma request posts in their local area. Clinics can use this web application to maintain the plasma donation activity

● Availability lookup for different blood groups.

● Hassle free and time saving.

● Makes emergency situation feasible.

10.2 ADVANTAGES

● This application acts as bridge between plasma recipients and donors.

● It is a user-friendly application. ● Anyone can access app and know information about the plasma donors. ● It stores data about plasma donors.

● Plasma Receiver can check the availability of the plasma donors based on matching blood group.

● Any plasma donor can give their details for donating plasma.

● A Better User Experience – With responsive design, it’s a lot easier and user friendly cross multiple platforms and various screen sizes.

● Client Secure Login – Impress clients with a modern web portal and improve customer service with automated processes.

● Easy Sign up/ Registration – It takes a few minutes to sign up

10.3 Disadvantages

- This application is used only for plasma donation.
- Internet reliance – 4G & Wi-Fi internet access is available in many locations, if you are unable to connect, you will not be able to access the application.
- Reduced Speed – It’s likely that a web application will operate at a slightly slower speed than one hosted on a server locally.
- It doesn't store detailed information about medical complications.
- It cannot auto verify user genuineness.
- It requires an active internet connection.

CHAPTER -11
CONCLUSION

11.1 CONCLUSION

This application serves as a bridge between the needy and cure. It is mainly developed for the people living in rural areas to make use of it in maintaining records of groups available in a region. Hence, it is quite beneficial for people to know more about it and how it helps in serving a broad category of patients in needy of plasma.

In recent days, it is noticed the increase in plasma request posts on social media such as Facebook, Twitter, and Instagram. Interestingly there are many people across the world interested in donating plasma when there is a need, but those donors don’t have an access to know about the plasma donation requests in their local area. This is because that there is no platform to connect local plasma donors with patients. DONATE PLASMA solves the problem and creates a communication channel through authorized clinics whenever a patient needs plasma donation.

CHAPTER -11
FUTURE SCOPE

11.1 FUTURE SCOPE

Enhancements that can be made in future with slight modifications we can make this application easy to operate and user friendly. This "DONATE PLASMA" can be developed to meet the requirements of modern-day system operation, new options and their respective roles will be applied and implemented.

● In future, we will try to verify the genuine users.
● Improving the UI.
● Quality records management.
● Can be added detailed locations of availability. ● Making it more efficient to deal a large collection base.