SPRINT-3

DATE	28 October 2022
Team ID	PNT2022TMID00855
Project Name	Plasma Donor Application

The following user stories are implemented in this sprint.

- **USN-1: As a user, I can register for the application by enteringmy email, password, and confirming my password.**
- **USN-2: As a user, I can log into the application using myregistered email & password.**

CODE:

dash.html:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Plasma Donor Application</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
    <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Raleway">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
```

```
<style>
  html,body,h1,h2,h3,h4,h5 {font-family: "Raleway", sans-serif}
  body{
  position: relative;
  }
  .modal {
  visibility: hidden;
  opacity: 0;
  position: absolute;
  top: 0;
  right: 0;
  bottom: 0;
  left: 0;
  display: flex;
  align-items: center;
  justify-content: center;
  background: rgba(77, 77, 77, .7);
  transition: all .4s;
  }
  .modal:target {
  visibility: visible;
  opacity: 1;
  }
  .modal__content {
  border-radius: 4px;
  position: relative;
  width: 500px;
  max-width: 90%;
  background: #fff;
```

```
padding: 1em 2em;
border: 2px solid yellow;
}
/* .modal__footer {
text-align: right;
a {
color: #585858;
}
i {
color: #d02d2c;
}
} */
.modal__close {
position: absolute;
top: 10px;
right: 10px;
color: #585858;
text-decoration: none;
font-size: 2rem;
}
.request-button{
height: auto;
padding: .5em 1em;
text-transform: uppercase;
background-color: transparent;
border: 1px solid #000;
border-radius: 100vmax;
}
.request-button a{
```

```
text-decoration: none;
}
form{
padding: 1em;
font-size: 1.2rem;
}
.form-control {
display: flex;
justify-content: space-between;
margin-bottom: 1.5em;
}
.form-control:last-child{
margin-bottom: 0;
}
.form-control * {
width: 100%;
}
.form-control button{
width: auto;
padding: .5em 2em;
background-color: transparent;
border: 1px solid #000;
border-radius: 100vmax;
}
.gender-field {
display: grid;
grid-auto-flow: column;
}
input{
```

```
        outline: none;
        }
        input[type=submit]{
        background-color: transparent;
        border: none;
        margin: 0;
        padding: 0;
        }
        .request-container{
        display: flex;
        justify-content: center;
        align-items: center;
        gap: 2em;
        position: absolute;
        top: 1em;
        right: 2em;
        }
        .request-container form{
        margin: 0;
        padding: 0;
        }
    </style>
</head>
<body class="w3-pale-yellow">
    <div class="request-container">
        <div class="request-button">
            <a href="#demo-modal">REQUEST</a>
        </div>
        <div class="request-button">
```

```
      <form action="/logout" method="POST">
        <input type="submit" value="LOG OUT">
      </form>
    </div>
  </div>
  <div id="demo-modal" class="modal">
    <div class="modal__content">
      <h1>Request Form</h1>
      <form action="/req" method="POST">
        <div class="form-control">
          <label for="username">Username:</label>
          <input
            type="text"
            id="username"
            name="username"
            maxlength="30"
            placeholder="username"
            required
            />
        </div>
        <div class="form-control">
          <label for="email">Email:</label>
          <input
            type="email"
            id="email"
            name="email"
            placeholder="email"
            required
            />
```

```
</div>
<div class="form-control">
  <label for="mobile">Mobile:</label>
  <input
    type="number"
    id="mobile"
    name="mobile"
    placeholder="mobile"
    required
    />
</div>
<div class="form-control">
  <label for="bloodgroup">Blood Group:</label>
  <select name="blood" id="bloodgroup">
    <option value="" selected disabled hidden>
      Choose your blood group
    </option>
    <option value="O+">O+</option>
    <option value="A+">A+</option>
    <option value="B+">B+</option>
    <option value="AB+">AB+</option>
    <option value="O-">O-</option>
    <option value="A-">A-</option>
    <option value="B-">B-</option>
    <option value="AB-">AB-</option>
  </select>
</div>
<div class="form-control">
  <button type="submit">Submit</button>
```

```
            <button type="reset">Reset</button>
          </div>
        </form>
        <a href="#" class="modal__close">&times;</a>
      </div>
    </div>
    <!-- Top container -->
    <div class="w3-bar w3-top w3-black w3-large" style="z-index:4">
      <button class="w3-bar-item w3-button w3-hide-large w3-hover-none w3-hover-text-light-grey" onclick="w3_open();"><i class="fa fa-bars"></i>  Menu</button>
      <span class="w3-bar-item w3-right"></span>
    </div>
    <!-- Sidebar/menu -->
    <nav class="w3-sidebar w3-collapse w3-amber w3-animate-left" style="z-index:3;width:300px;" id="mySidebar">
      <br>
      <div class="w3-container w3-row">
        <div class="w3-col s8 w3-bar">
          <span>Welcome, <strong>{{un}}</strong></span><br>
          <a href="#" class="w3-bar-item w3-button"><i class="fa fa-user"></i></a>
        </div>
      </div>
      <hr>
      <div class="w3-container">
        <h5>Dashboard</h5>
      </div>
      <div class="w3-bar-block">
        <a href="#" class="w3-bar-item w3-button w3-padding-16 w3-hide-large w3-dark-grey w3-hover-black" onclick="w3_close()" title="close menu"><i class="fa fa-remove fa-fw"></i>  Close Menu</a>
        <a href="#" class="w3-bar-item w3-button w3-padding w3-blue"><i class="fa fa-
```

```
users fa-fw"></i>  Overview</a>

        <a href="/index" class="w3-bar-item w3-button w3-padding"><i class="fa fa-home
fa-fw"></i>  Home </a>

      </div>

    </nav>

    <!-- Overlay effect when opening sidebar on small screens -->

    <div class="w3-overlay w3-hide-large w3-animate-opacity" onclick="w3_close()"
style="cursor:pointer" title="close side menu" id="myOverlay"></div>

    <!-- !PAGE CONTENT! -->

    <div class="w3-main" style="margin-left:300px;margin-top:43px;">

      <!-- Header -->

      <header class="w3-container" style="padding-top:22px">

        <h5><b><i class="fa fa-dashboard"></i> My Dashboard</b></h5>

      </header>

      <div class="w3-row-padding w3-margin-bottom">

        <div class="w3-quarter">

          <div class="w3-container w3-amber w3-padding-16">

            <div class="w3-left"><i class="fa fa-tint w3-xxxlarge"></i></div>

            <div class="w3-right">

              <h3>{{dicrow}}</h3>

            </div>

            <div class="w3-clear"></div>

            <h4>Plasma Availability</h4>

          </div>

        </div>

        <div class="w3-quarter">

          <div class="w3-container w3-orange w3-text-white w3-padding-16">

            <div class="w3-left"><i class="fa fa-users w3-xxxlarge"></i></div>

            <div class="w3-right">

              <h3>{{tot}}</h3>
```

```
        </div>
        <div class="w3-clear"></div>
        <h4>Application Users</h4>
      </div>
    </div>
  </div>
  <div class="w3-panel">
    <div class="w3-twothird">
      <h5>Plasma </h5>
      <table class="w3-table w3-striped w3-khaki">
        <tr>
          <td><i class="fa fa-tint"></i></td>
          <td><b>A+</b></td>
          <td><i>1</i></td>
        </tr>
        <tr>
          <td><i class="fa fa-tint"></i></td>
          <td><b>A-</b></td>
          <td><i>0</i></td>
        </tr>
        <tr>
          <td><i class="fa fa-tint"></i></td>
          <td><b>B+</td>
          <td><i>1</i></td>
        </tr>
        <tr>
          <td><i class="fa fa-tint"></i></td>
          <td><b>B-</td>
          <td><i>0</i></td>
```

```
            </tr>
            <tr>
              <td><i class="fa fa-tint"></i></td>
              <td><b>AB+</td>
              <td><i>2</i></td>
            </tr>
            <tr>
              <td><span class="fa fa-tint"></span></td>
              <td><b>AB-</td>
              <td><i>1</i></td>
            </tr>
            <tr>
              <td><i class="fa fa-tint"></i></td>
              <td><b>O+</td>
              <td><i>1</i></td>
            </tr>
            <tr>
              <td><i class="fa fa-tint"></i></td>
              <td><b>O-</td>
              <td><i>0</i></td>
            </tr>
          </table>
        </div>
      </div>
    </div>
    <hr>
    <br>
    <div class="w3-container w3-khaki w3-padding-16 w3-margin-
top=100px"style="margin-left:300px;">
      <div class="w3-row">
```

```
        <div class="w3-container w3-third">
          <h5 class="w3-bottombar w3-border-green">Demographic</h5>
          <p>Language</p>
          <p>Country</p>
          <p>City</p>
        </div>
        <div class="w3-container w3-third">
          <h5 class="w3-bottombar w3-border-red">System</h5>
          <p>Browser</p>
          <p>OS</p>
          <p>More</p>
        </div>
        <div class="w3-container w3-third">
          <h5 class="w3-bottombar w3-border-orange">Target</h5>
          <p>Users</p>
          <p>Active</p>
          <p>Geo</p>
          <p>Interests</p>
        </div>
      </div>
    </div>
    <!-- Footer -->
    <footer class="w3-container w3-padding-16 w3-pale-yellow">
      <h4>FOOTER</h4>
      <p>Powered by <a href="https://www.w3schools.com/w3css/default.asp"
target="_blank">w3.css</a></p>
    </footer>
    <!-- End page content -->
    </div>
  </body>
```

app.py:

```
from turtle import st
from flask import Flask, render_template, request, redirect, url_for, session, jsonify
from markupsafe import escape
import os
from dotenv import load_dotenv
from sendgrid import sendgrid
from sendgrid.helpers.mail import Mail, Email, To, Content
import ibm_db
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=32286;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=xnw03376;PWD=CAip5VvCHctcirIL",",")

app = Flask(__name__)
load_dotenv()   #load keys from .env
sg = sendgrid.SendGridAPIClient(api_key = os.environ.get('GmWQl2hzTKeUsU776_Xkyw'))   #set SendGrid API Key
from_email = Email("babinmon2001@gmail.com")
@app.route('/')
def home():
  sql = "select * from register where age=21 "
  stmt = ibm_db.prepare(conn, sql)
  ibm_db.execute(stmt)
  dicrow = ibm_db.fetch_assoc(stmt)
  print(dicrow)
  return render_template('index.html')

@app.route('/index')
```

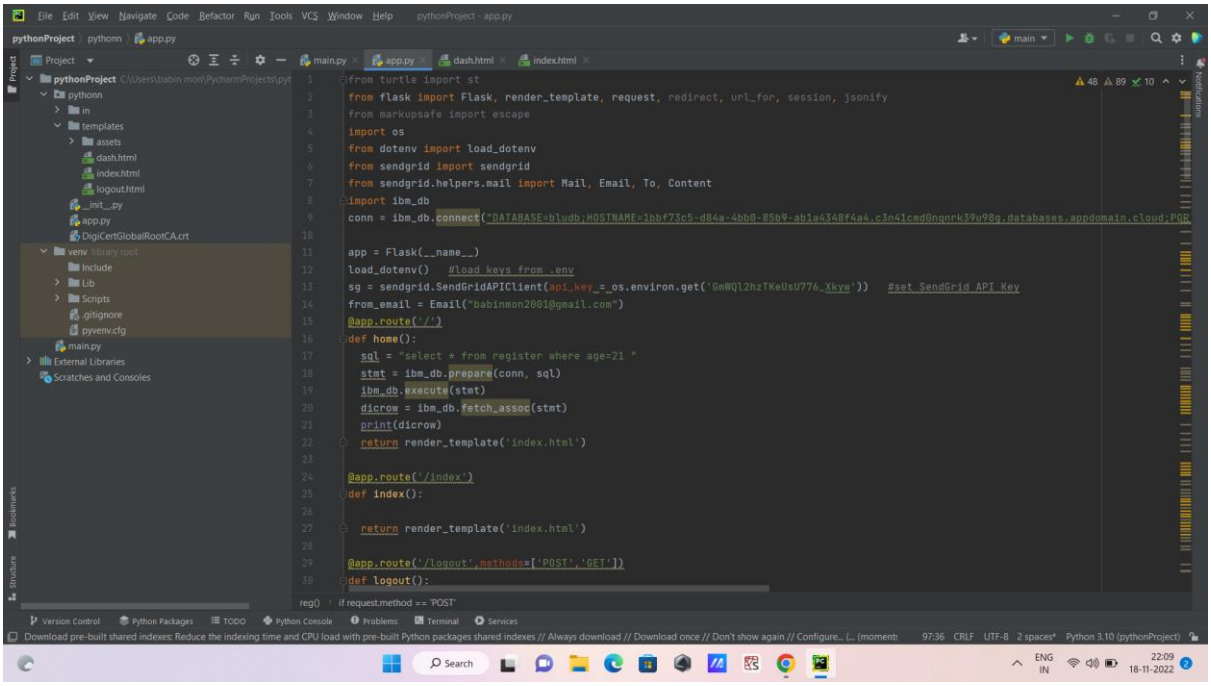
```
def index():

  return render_template('index.html')

@app.route('/logout',methods=['POST','GET'])
def logout():
  if request.method == 'POST':
    return render_template('logout.html')
  else:
    return render_template('dash.html')

@app.route('/home',methods=['POST','GET'])
def homepage():
  if request.method == 'POST':
    return render_template('index.html')
  else:
    return render_template('logout.html')


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

  if request.method == 'POST':
    username = request.form['username']
    email = request.form['email']
    phone = request.form['mobile']
    blood=request.form['blood']

    insert_sql = "INSERT INTO request VALUES (?,?,?,?)"
    prep_stmt = ibm_db.prepare(conn, insert_sql)
```

```
    ibm_db.bind_param(prep_stmt, 1, username)
    ibm_db.bind_param(prep_stmt, 2, email)
    ibm_db.bind_param(prep_stmt, 3, phone)
    ibm_db.bind_param(prep_stmt, 4, blood)

    ibm_db.execute(prep_stmt)

    return render_template('index.html',msg="saved successfully")

@app.route('/access',methods=['POST','GET'])
def accesss():

  if request.method == 'POST':
    username = request.form['username']
    email = request.form['email']
    phone = request.form['mobile']
    address=request.form['address']
    password=request.form['password']

    insert_sql = "INSERT INTO access VALUES (?,?,?,?,?)"
    prep_stmt = ibm_db.prepare(conn, insert_sql)
    ibm_db.bind_param(prep_stmt, 1, username)
    ibm_db.bind_param(prep_stmt, 5, email)
    ibm_db.bind_param(prep_stmt, 3, phone)
    ibm_db.bind_param(prep_stmt, 4, address)
    ibm_db.bind_param(prep_stmt, 2, password)

    ibm_db.execute(prep_stmt)
```

```
    return render_template('index.html',msg="saved successfully")


@app.route('/reg',methods = ['POST', 'GET'])
def reg():
  if request.method == 'POST':
    username = request.form['username']
    password = request.form['password']
    email = request.form['email']
    blood=request.form['blood-group']
    gender=request.form['gender']
    age=request.form['age']
    address=request.form['address']
    phone=request.form['mobile']
    insert_sql = "INSERT INTO register VALUES (?,?,?,?,?,?,?,?)"
    prep_stmt = ibm_db.prepare(conn, insert_sql)
    ibm_db.bind_param(prep_stmt, 1, username)
    ibm_db.bind_param(prep_stmt, 2, password)
    ibm_db.bind_param(prep_stmt, 3, email)
    ibm_db.bind_param(prep_stmt, 4, blood)
    ibm_db.bind_param(prep_stmt, 5, gender)
    ibm_db.bind_param(prep_stmt, 6, age)
    ibm_db.bind_param(prep_stmt, 7, address)
    ibm_db.bind_param(prep_stmt, 8, phone)
    ibm_db.execute(prep_stmt)
    to_email = To(email)  # set user as recipient for confirmation email
    subject = "Welcome to GetPlasma"
    content = Content("text/html",
                      "<p>Hello " + username + ",</p><p>Thank you for registering to the
GetPlasma Application!</p><p>If this wasn't you, then immediately report to our <a
href=\"mailto:getplasmaproject@gmail.com\">admin</a> or just reply to this email.</p>")
```

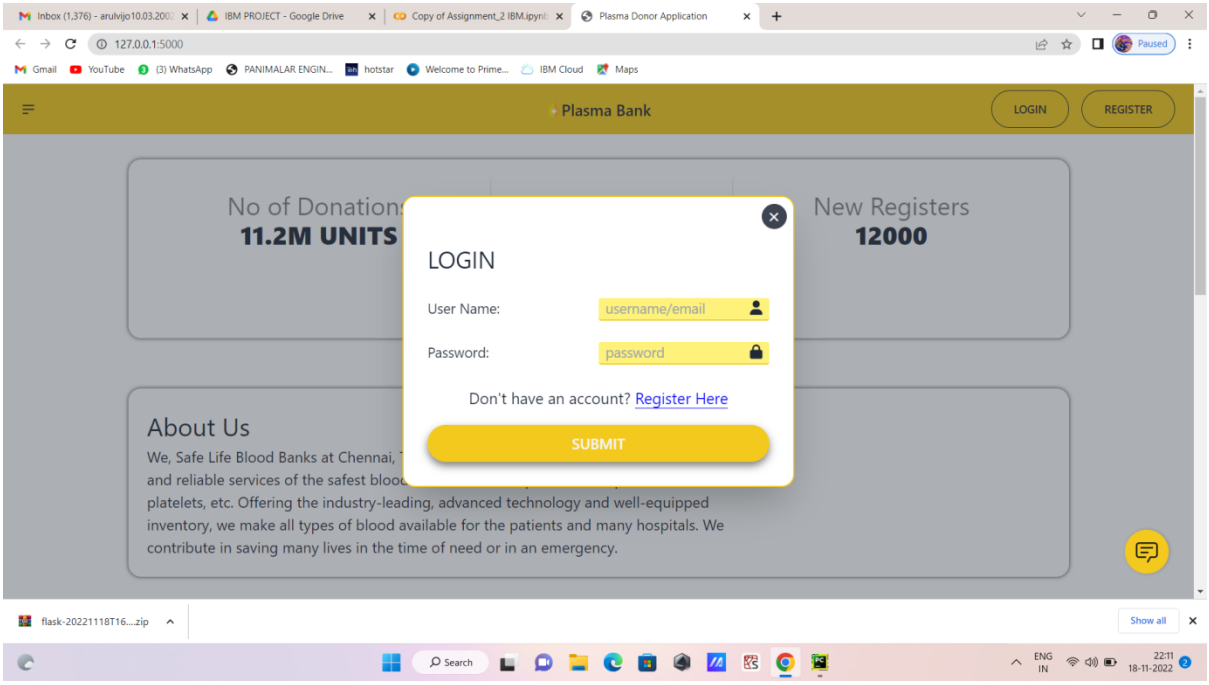
```
    email = Mail(from_email, to_email, subject, content)  # construct email format
    email_json = email.get()  # get JSON-ready representation of the mail object

  return render_template('index.html',msg="saved successfully")

@app.route('/login',methods = ['POST', 'GET'])
def login():
  if request.method == 'POST':
    username = request.form['username']
    password = request.form['password']
    sql = "select * from access where username=? and password=?"
    stmt = ibm_db.prepare(conn, sql)
    ibm_db.bind_param(stmt, 1, username)
    ibm_db.bind_param(stmt, 2, password)
    ibm_db.execute(stmt)
    dic = ibm_db.fetch_assoc(stmt)
    print(dic)
    role = str()
    requests = []
    if dic:
      un=dic['USERNAME']
      print(un)
      sql = "select count(*) from register"
      stmt = ibm_db.prepare(conn, sql)
      ibm_db.execute(stmt)
      dicrow = ibm_db.fetch_assoc(stmt)
      sql1 = "select count(*) from request"
      stmt1 = ibm_db.prepare(conn, sql1)
```

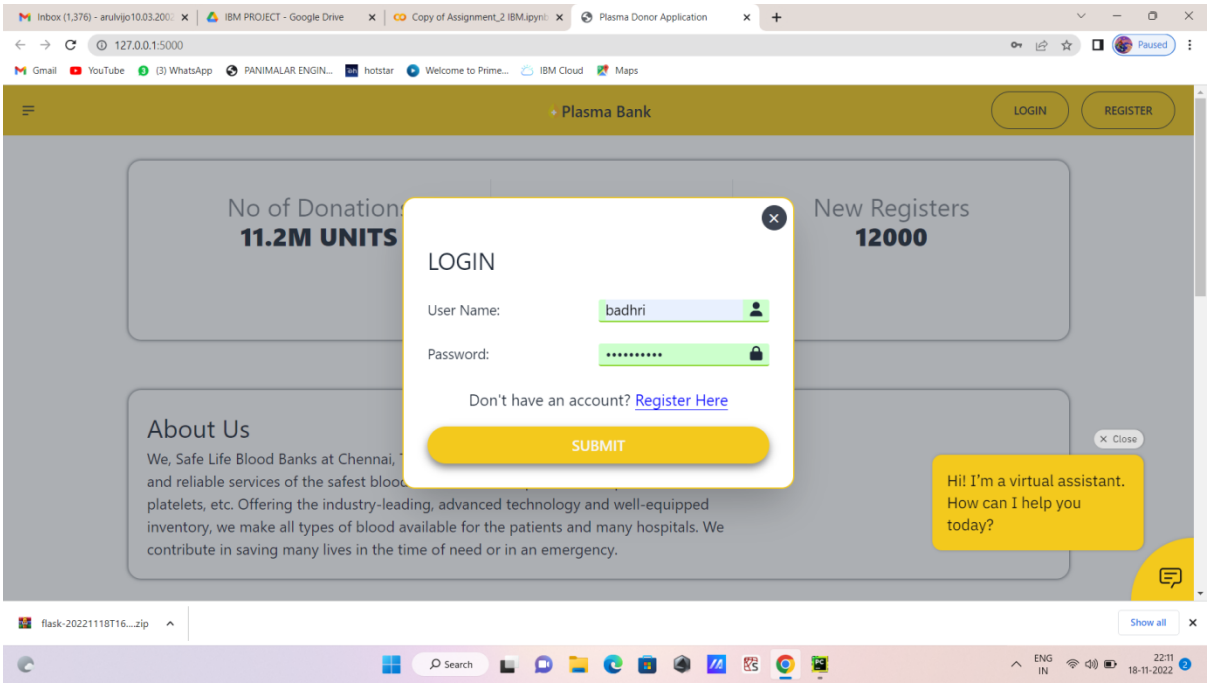
```
        ibm_db.execute(stmt1)
        dicrow1 = ibm_db.fetch_assoc(stmt1)
        sql2 = "select count(*) from access"
        stmt2 = ibm_db.prepare(conn, sql2)
        ibm_db.execute(stmt2)
        dicrow2 = ibm_db.fetch_assoc(stmt2)
        tot=(dicrow['1']+dicrow1['1']+dicrow2['1'])
        bloodA = ibm_db.fetch_assoc(stmt1)
        print(bloodA)
        return render_template('dash.html',un=un,dicrow=dicrow['1'],tot=tot)
    else:
        return render_template('index.html')
  return render_template('index.html')


if __name__ == "__main__":
    app.run(debug=True)
```


File Edit View Navigate Code Refactor Run Tools VCS Window Help
pythonProject - app.py
pythonProject > pythonn > app.py
main
Project
main.py
app.py
dash.html
index.html
pythonProject C:\Users\babin mon\PycharmProjects\pyt
pythonn
in
templates
assets
dash.html
index.html
logout.html
__init__.py
app.py
DigiCertGlobalRootCA.crt
venv library root
Include
Lib
Scripts
.gitignore
pyvenv.cfg
main.py
External Libraries
Scratches and Consoles
from turtle import st
from flask import Flask, render_template, request, redirect, url_for, session, jsonify
from markupsafe import escape
import os
from dotenv import load_dotenv
from sendgrid import sendgrid
from sendgrid.helpers.mail import Mail, Email, To, Content
import ibm_db
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;POR
app = Flask(__name__)
load_dotenv() #Load keys from .env
sg = sendgrid.SendGridAPIClient(api_key=os.environ.get('SmWQl2hzTKeUsU776_Xkyw')) #set SendGrid API Key
from_email = Email("babinmon2001@gmail.com")
@app.route('/')
def home():
  sql = "select * from register where age=21 "
  stmt = ibm_db.prepare(conn, sql)
  ibm_db.execute(stmt)
  dicrow = ibm_db.fetch_assoc(stmt)
  print(dicrow)
  return render_template('index.html')
@app.route('/index')
def index():
  return render_template('index.html')
@app.route('/logout',methods=['POST','GET'])
def logout():
reg() > if request.method == 'POST'
Version Control
Python Packages
TODO
Python Console
Problems
Terminal
Services
Download pre-built shared indexes: Reduce the indexing time and CPU load with pre-built Python packages shared indexes // Always download // Download once // Don't show again // Configure... (... (moments
97:36
CRLF
UTF-8
2 spaces*
Python 3.10 (pythonProject)
Search
ENG
IN
22:09
18-11-2022

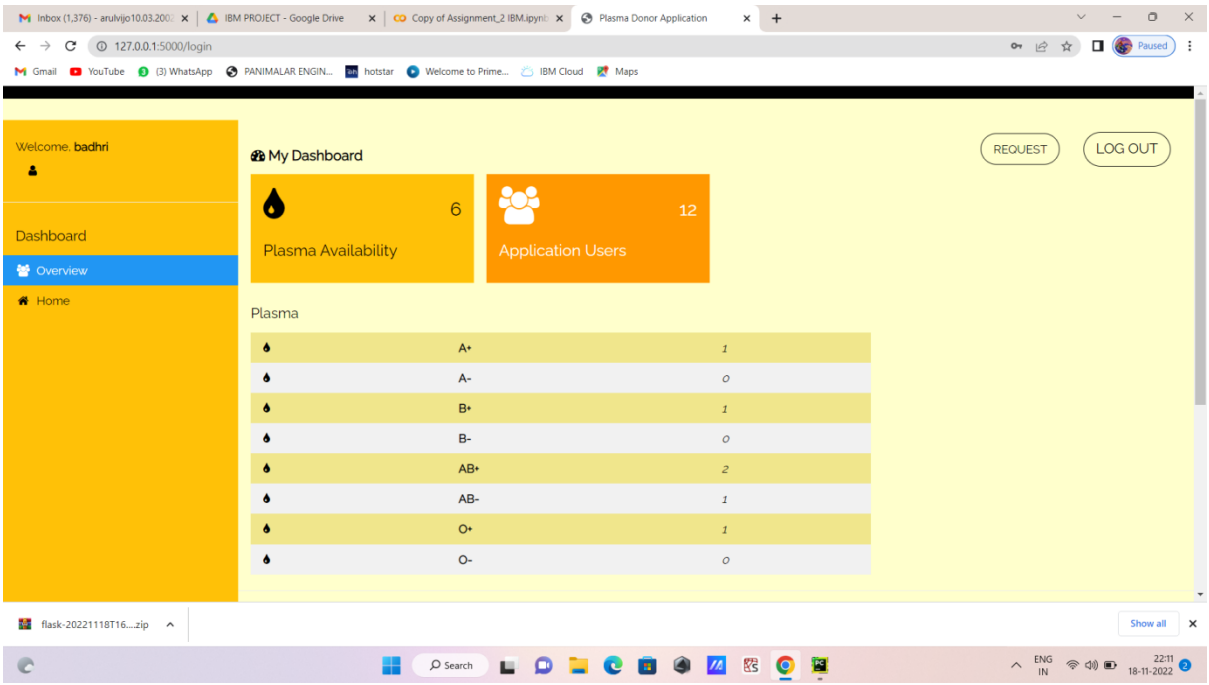
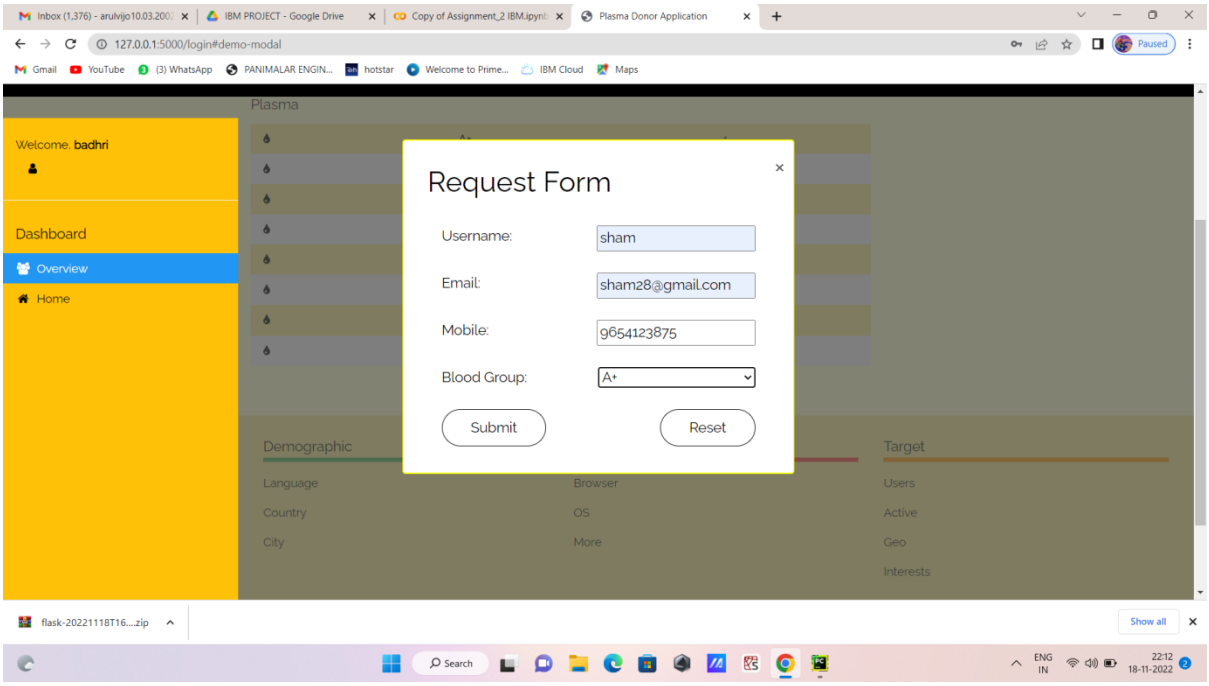
SCREENSHOTS:

Login page:





Dashboard page



Request Form