SPRINT - 2

PROJECT NAME : PLASMA DONOR APPLICATION

TEAM ID : PNT2022TMID25930

DOMAIN : CLOUD APPLICATION DEVELOPMENT

As a part of sprint 1 we delivered the frontend code. Now as a part of this phase we have worked on backend.

App.py

```python
from flask import Flask, render_template, request, redirect, url_for, session
import ibm_db
import re
app = Flask(__name__)
app.secret_key = 'a'


conn= ibm_db.connect("DATABASE=bludb;HOSTNAME=19af6446-6171-4641-8aba-9dcff8e1b6ff.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30699;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=wdr02881;PWD=awfqjNgEkItfWG3M",",")


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


@app.route('/login',methods =['GET', 'POST'])
def login():
    global userid
    msg = ''
```

```python
    if request.method == 'POST' :
        username = request.form['username']
        password = request.form['password']
        sql = "SELECT * FROM users 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)
        account = ibm_db.fetch_assoc(stmt)
        print (account)
        if account:
            session['loggedin'] = True
            session['id'] = account['USERNAME']
            userid=  account['USERNAME']
            session['username'] = account['USERNAME']
            msg = 'Logged in successfully !'

            msg = 'Logged in successfully !'
            return render_template('dashboard.html', msg = msg)
        else:
            msg = 'Incorrect username / password !'
    return render_template('login.html', msg = msg)
```

```python
@app.route('/register', methods =['GET', 'POST'])
def register():
    msg = ''
    if request.method == 'POST' :
        username = request.form['username']
        email = request.form['email']
        password = request.form['password']
        sql = "SELECT * FROM users WHERE username =?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
            msg = 'Account already exists !'
        elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):
            msg = 'Invalid email address !'
        elif not re.match(r'[A-Za-z0-9]+', username):
            msg = 'name must contain only characters and numbers !'
        else:
            insert_sql = "INSERT INTO  users 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, password)
            ibm_db.execute(prep_stmt)
```

```python
            msg = 'You have successfully registered !'
    elif request.method == 'POST':
        msg = 'Please fill out the form !'
    return render_template('register.html', msg = msg)


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


@app.route('/donate',methods =['GET', 'POST'])
def donate():
    msg = ''
    if request.method == 'POST' :
        username = request.form['username']
        email = request.form['email']


        bloodgroup= request.form['bloodgroup']
        phno = request.form['phno']
        jobs = request.form['s']
        sql = "SELECT * FROM donors WHERE username =?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account)
```

```python
        insert_sql = "INSERT INTO  RECEIVER 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.execute(prep_stmt)
        msg = 'You have successfully selected a donor !'




    elif request.method == 'POST':
        msg = 'Please select a donor !'
    return render_template('donor.html', msg = msg)


@app.route('/receiver')
def receiver():
    print(session["username"],session['id'])


    cursor = mysql.connection.cursor()
    cursor.execute('SELECT * FROM donors WHERE userid = % s', (session['id'],))
    account = cursor.fetchone()
```

```python
    print("accountdislay",account)



    return render_template('display.html',account = account)


@app.route('/logout')

def logout():
    session.pop('loggedin', None)
    session.pop('id', None)
    session.pop('username', None)
    return render_template('home.html')




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