INTEGRATING SENDGRID SERVICE

Team ID	PNT2022TMID13363
Project Name	Plasma donor application

Definition:

   SendGrid is **a cloud-based SMTP provider that allows you to send email without having to maintain email servers**. SendGrid manages all of the technical details, from scaling the infrastructure to ISP outreach and reputation monitoring to whitelist services and real time analytics.

Code:

```
from flask import Flask, render_template, request, redirect, url_for, session
import ibm_db
import re

app = Flask(__name__)

hostname = '2f3279a5-73d1-4859-88f0-a6c3e6b4b907.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud'
uid = 'dfw40988'
pwd = '6gng8NbeWBZg8jwW'
driver = "{IBM DB2 ODBC DRIVER}"
```

```python
db_name = 'Bludb'
port = '30756'
protocol = 'TCPIP'
cert = "Certificate.crt"
dsn = (
    "DATABASE ={0};"
    "HOSTNAME ={1};"
    "PORT ={2};"
    "UID ={3};"
    "SECURITY=SSL;"
    "PROTOCOL={4};"
    "PWD ={6};"
).format(db_name, hostname, port, uid, protocol, cert, pwd)
connection = ibm_db.connect(dsn, "", "")
print()
# query = "SELECT username FROM USER1 WHERE username=?"
# stmt = ibm_db.prepare(connection, query)
# ibm_db.bind_param(stmt, 1, username)
# ibm_db.execute(stmt)
# username = ibm_db.fetch_assoc(stmt)
# print(username)
app.secret_key = 'a'
```

```python
@app.route('/', methods=['GET', 'POST'])
@app.route('/register', methods=['GET', 'POST'])
def register():
    msg = " "
    if request.method == 'POST':
        username = request.form['username']
        email_id = request.form['email_id']
        phone_no = request.form['phone_no']
        password = request.form['password']
        query = "SELECT * FROM USER1 WHERE username=?;"
        stmt = ibm_db.prepare(connection, query)
        ibm_db.bind_param(stmt, 1, username)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        if (account):

            msg = "Account already exists!"
            return render_template('register.html', msg=msg)
        # elif not re.match(r'[^@]+@[^@]+\.[^@]+', email_id):
        #     msg = "Invalid email addres"
        # elif not re.match(r'[A-Za-z0-9+', username):
        #     msg = "Name must contain only characters and numbers"
        else:
            query = "INSERT INTO USER1 values(?,?,?,?)"
```

```python
            stmt = ibm_db.prepare(connection, query)

            ibm_db.bind_param(stmt, 1, username)

            ibm_db.bind_param(stmt, 2, email_id)

            ibm_db.bind_param(stmt, 3, phone_no)

            ibm_db.bind_param(stmt, 4, password)

            ibm_db.execute(stmt)

            msg = 'You have successfully Logged In!!'

            return render_template('login.html', msg=msg)

        else:

          msg = 'PLEASE FILL OUT OF THE FORM'

          return render_template('register.html', msg=msg)




@app.route('/login', methods=['GET', 'POST'])

def login():

    global userid

    msg = ' '

    if request.method == "POST":

        username = request.form['username']

        password = request.form['password']

        query = "select * from user1 where username=? and password=?"

        stmt = ibm_db.prepare(connection, query)

        ibm_db.bind_param(stmt, 1, username)

        ibm_db.bind_param(stmt, 2, password)
```

```python
        ibm_db.execute(stmt)

        account = ibm_db.fetch_assoc(stmt)

        print(account)

        if account:

            session['Loggedin'] = True

            session['id'] = account['USERNAME']

            session['username'] = account['USERNAME']

            msg = 'Logged in Successfully'

            return render_template('welcome.html', msg=msg,
username=str.upper(username))

        else:

            msg = 'Incorrect Username or Password'

            return render_template('login.html', msg=msg)

    else:

        msg = 'PLEASE FILL OUT OF THE FORM'

        return render_template('login.html', msg=msg)


@app.route('/welcome', methods=['GET', 'POST'])
def welcome():
    if request.method == 'POST':

        username = request.form['username']

        print(username)

        return render_template('welcome.html', username=username)

    else:
```

```python
    return render_template('welcome.html', username=username)


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