

## Sprint 2

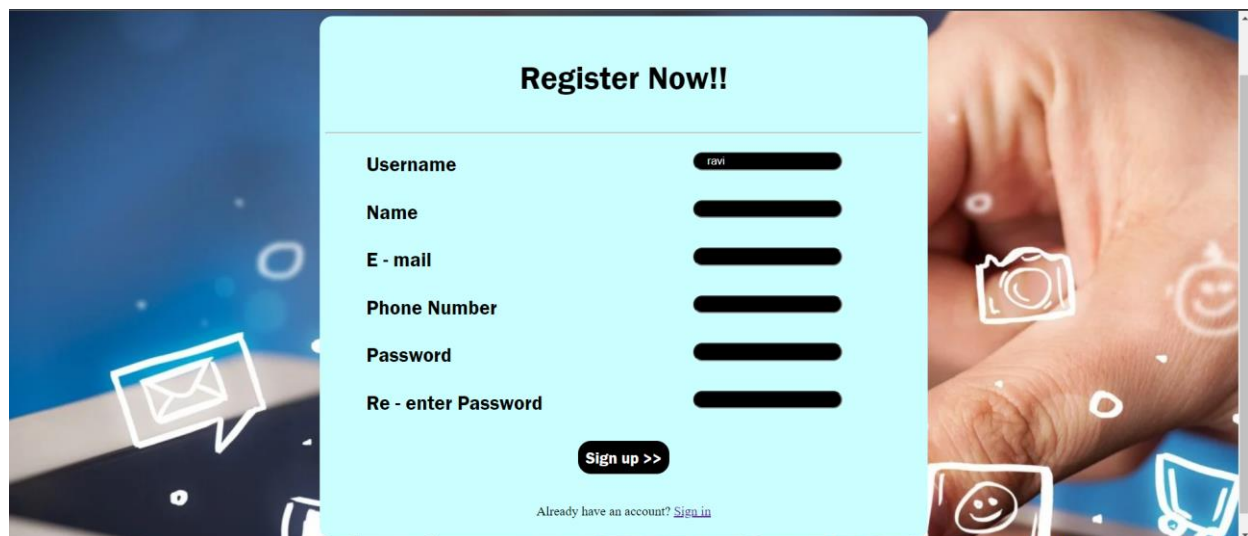
Team Mates:  
VISHALI R  
VISHAGAN C SU  
SOWMIYAA S  
ASHOK KUMAR CV

Team ID: PNT2022TMID21475

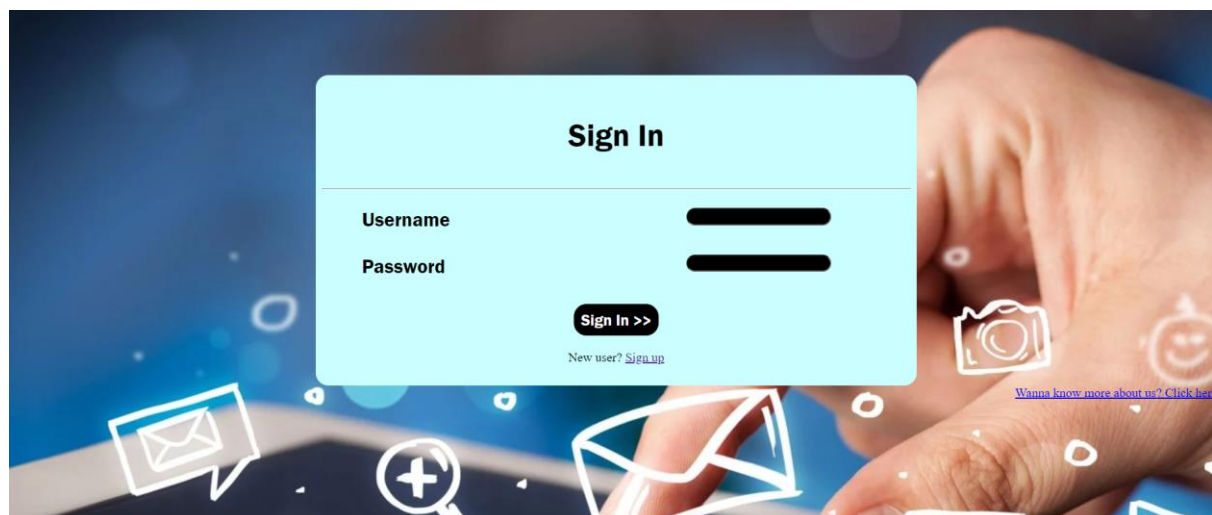
### Completed Tasks:

- Login page
- Sign Up page
- Dashboard page
- Add tickets page
- Sendgrid integration

### 1. Create an UI to interact with Application



A screenshot of a web form titled "Register Now!!". The form is light blue and contains the following fields: Username (with the value "ravi" entered), Name, E-mail, Phone Number, Password, and Re-enter Password. Each field has a corresponding input box. Below the fields is a "Sign up >>" button. At the bottom, there is a link: "Already have an account? [Sign in](#)". The background of the form is a blurred image of hands with various icons overlaid.



A screenshot of a web form titled "Sign In". The form is light blue and contains the following fields: Username and Password. Each field has a corresponding input box. Below the fields is a "Sign In >>" button. At the bottom, there is a link: "New user? [Sign up](#)". The background of the form is a blurred image of hands with various icons overlaid.

### 2. Connect with IBM

DB2Agent Table:

The screenshot shows the IBM Db2 on Cloud console. The top navigation bar includes 'Load Data', 'Load History', 'Tables', 'Views', 'Indexes', 'Aliases', 'MQTs', 'Sequences', and 'Application objects'. The 'Tables' tab is selected. Below the navigation bar is a search bar labeled 'Find schemas or tables' and a 'Refresh' button. The main content area is divided into two panels: 'Schemas' and 'Tables'.

**Schemas Panel:**

Name	Type	Tables
FBB91466	User	3

Total: 1, selected: 1

**Tables Panel:**

Name	Schema	Properties
AGENTS	FBB91466	...
COMPLAINTS	FBB91466	...
USERS	FBB91466	...

Total: 3, selected: 0

## Customer Table:

The screenshot shows the 'Customer Table' in the IBM Db2 on Cloud console. The table has five columns: USERNAME, NAME, EMAIL, PHN, and PASSWORD. There is one row of data. An 'Export to CSV' button is visible in the top right corner.

USERNAME	NAME	EMAIL	PHN	PASSWORD
subash12	Subash	subash12@gmail.com	7010566424	subash

## Code:

## App.py

:

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

app = Flask(__name__)

# for connection
# conn= ""

app.secret_key = 'a'
dsn_hostname = "0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud" # e.g.: "54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgu0lqde00.databases.appdomain.cloud"
dsn_uid = "fbb91466" # e.g. "abc12345"
dsn_pwd = "FCkG6iJmug58yjCR" # e.g. "7dBZ3wWt9XN6$o0J"

dsn_driver = "{IBM DB2 ODBC DRIVER}"
dsn_database = "BLUDB" # e.g. "BLUDB"
dsn_port = "31198" # e.g. "32733"
dsn_protocol = "TCPIP" # i.e. "TCPIP"
dsn_security = "SSL" # i.e. "SSL"
```

```
dsn = (
    "DRIVER={0};"
    "DATABASE={1};"
    "HOSTNAME={2};"
    "PORT={3};"
    "PROTOCOL={4};"
    "UID={5};"
    "PWD={6};"
    "SECURITY={7};").format(dsn_driver, dsn_database, dsn_hostname, dsn_port, dsn_protocol, dsn_uid,
dsn_pwd,dsn_security)

conn = ibm_db.connect(dsn, "", "")

print("connection successful...")


@app.route('/signup', methods=['GET', 'POST'])
def signup():
    global userid
    msg = ""
    if request.method == 'POST':
        username = request.form['username']
        name = request.form['name']
        email = request.form['email']
        phn = request.form['phn']
```

```

password = request.form['pass']repass =
request.form['repass']print("inside
checking") print(name)
if len(username) == 0 or len(name) == 0 or len(email) == 0 or len(phn)
== 0 or len(password) == 0 or len(repass) == 0: msg = "Form is not
filled completely!!"print(msg)
return render_template('signup.html', msg=msg)elif password
!= repass:
    msg = "Password is not matched"
    print(msg)
    return render_template('signup.html', msg=msg)elif not
re.match(r'[a-z]+' , username):
    msg = 'Username can contain only small letters and numbers'print(msg)
    return render_template('signup.html', msg=msg)elif not
re.match(r'^@[^@]+\.[^@]+' , email):
    msg = 'Invalid email'
    print(msg)
    return render_template('signup.html', msg=msg)elif not
re.match(r'[A-Za-z]+' , name):
    msg = "Enter valid name"
    print(msg)
    return render_template('signup.html', msg=msg)elif not
re.match(r'[0-9]+' , phn):
    msg = "Enter valid phone number"
    print(msg)
    return render_template('signup.html', msg=msg)

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'else:
    insert_sql = "insert into users values(?,?,?,?)"prep_stmt =
    ibm_db.prepare(conn, insert_sql) ibm_db.bind_param(prepare_stmt, 1,
    username)
    ibm_db.bind_param(prepare_stmt, 2, name)
    ibm_db.bind_param(prepare_stmt, 3, email)
    ibm_db.bind_param(prepare_stmt, 4, phn)
    ibm_db.bind_param(prepare_stmt, 5, password)
    ibm_db.execute(prepare_stmt) print("successs")

```

```

        msg = "succesfully signed up"
        return render_template('dashboard.html', msg=msg, name=name)else:
        return render_template('signup.html')

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

@app.route('/login', methods=["GET", "POST"])def login():
    global useridmsg
    = "
    if request.method == 'POST':
        username = request.form['username']password
        = request.form['pass']
        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'
            return render_template('dashboard.html', name=account['name'])else:
            msg = 'Incorrect user credentials'
            return render_template('dashboard.html', msg=msg)

    else:
        return render_template('login.html')

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

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

```









```

        Password
    </div>
    <div class="textinformright">
        <input type="password" name="pass">
    </div>
</div>

    <br>
    <div>
        <button class="forbutton" type="submit"> Sign In
>></button>
    </div>
</form>
<br>

    <div>
        New user? <a href="/signup">Sign up</a>
    </div>
    <br>
</div>

</div>
</div>

{% endblock %}

```

### Dashboard.html:

```

{% extends 'base.html' %}

{% block head %}

<title>
    Dashboard
</title>

{% endblock %}

{% block body %}

<!-- things

    div 1
welcome jetson,    sign out

```

```
div 2
your complaints status

add new complaint -->
<br>
<div class="fordashboardtop">
  <div class="fordashboardtopelements1">Welcome
    Ravi,
  </div>
  <div class="fordashboardtopelements2">
    <a href="/login"><button class="forbutton">Sign out</button></a>
  </div>
</div>
<br>
<div class="outerofdashdetails">

  <div class="fordashboarddetails">
    <br>
    <!-- table of customers complaints -->
    <table class="fortable">
      <thead>
        <th class="pad">Complaint Detail</th>
        <th>Assigned Agent</th>
        <th>Status</th>
        <th>More details</th>
      </thead>
      <tbody>
        <tr>
          <td class="pad">
            Network Issue
          </td>
          <td>
            Dr.Raj
          </td>
          <td>
            Not Completed
          </td>
          <td>
            <a href="">Click Here >></a>
          </td>
        </tr>
      </tbody>
    </table>
    <br>
```

```

        <center>
            <div class="fordashboardtopelements2">
                <a href="/login"><button class="forbutton2">Add new complaint
+ </button></a>
            </div>
        </center>
    </div>

</div>

{% endblock %}

```

### Base.html:

```

<!DOCTYPE html>

<head>

    <link rel="stylesheet" href="static/css/main.css"/>
    {% block head %} {% endblock %}
</head>

<body>
    {% block body %}

    {% endblock %}
</body>

</html>

```