

PROJECT DEVELOPMENT PHASE

SPRINT 1

Date	10 November 2022
Team ID	PNT2022TMID54054
Project Name	Smart Fashion Recommender Application
Maximum Marks	20 Marks

SPRINT 1:

1. Admin Registration
2. User Registration
3. Admin Login Page
4. User Login Page

CONNECT REGISTRATION PAGE WITH DB2:

STEP 1: Create IBM DB2

STEP 2: Connect Python Code with DB2

STEP 3: Database Table of Register / Login Details

STEP 4: Directed to the Home Page

CODE

FILE 1: Register.html

```
dashboard.html login.html register.html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <link rel="stylesheet" href="/static/style.css">
8   <link rel="stylesheet" href="{{url_for('static',filename='style.css')}}" type="text/css">
9   <title>Login Page</title>
10 </head>
11 <body>
12   <div class="container">
13     <div class="title">Login</div>
14     <div class="content">
15       <form method="post" action="login">
16         <div class="user-details">
17           <div class="input-box">
18             <span class="details">Phone Number</span>
19             <input type="text" placeholder="Enter your number" name="phone" value="{{request.form['phone']}}" required>
20           </div>
21           <br>
22           <div class="input-box">
23             <span class="details">Password</span>
24             <input type="password" placeholder="Enter your password" name="password" value="{{request.form['password']}}" required>
25           </div>
26         </div>
27         <div class="button">
28           <input type="submit" value="Login">
29         </div>
30       </form>
31     </div>
32   </div></body>
33 </html>
```

FILE 2: Login.html

```
dashboard.html login.html register.html
1 <head>
2   <meta name="viewport" content="width=device-width, initial-scale=1.0">
3   <link rel="stylesheet" href="/static/style.css">
4   <title>Registration page </title>
5 </head>
6 <body>
7   <div class="container">
8     <div class="title">Registration</div>
9     <div class="content">
10       <form method="post">
11         <div class="user-details">
12           <div class="input-box">
13             <span class="details">Username</span>
14             <input type="text" placeholder="Enter your username" name="username" value="{{request.form['username']}}" required>
15           </div>
16           <div class="input-box">
17             <span class="details">Phone Number</span>
18             <input type="text" placeholder="Enter your number" name="phone" value="{{request.form['phone']}}" required>
19           </div>
20           <div class="input-box">
21             <span class="details">Password</span>
22             <input type="password" placeholder="Enter your password" name="password" value="{{request.form['password']}}" required>
23           </div>
24           <div class="input-box">
25             <span class="details">Confirm Password</span>
26             <input type="password" placeholder="Confirm your password" name="cpassword" value="{{request.form['cpassword']}}" required>
27           </div>
28         </div>
29         <div class="button"><input type="submit" value="Register"></div>
30       </form>
31     </div>
32   </div>
33 </body>
34 </html>
```

FILE 3: Style.css

```
1 *{
2   margin: 0;
3   padding: 0;
4   box-sizing: border-box;
5   font-family: 'Poppins', sans-serif;
6 }
7 body{
8   height: 100vh;
9   display: flex;
10  justify-content: center;
11  align-items: center;
12  padding: 10px;
13  background: linear-gradient(135deg, #71b7e6, #9b59b6);
14 }
15 .container{
16   max-width: 700px;
17   width: 100%;
18   background-color: #fff;
19   padding: 25px 30px;
20   border-radius: 5px;
21   box-shadow: 0 5px 10px rgba(0,0,0,0.15);
22 }
23 .container .title{
24   font-size: 25px;
25   font-weight: 500;
26   position: relative;
27 }
28 .container .title::before{
29   content: "";
30   position: absolute;
31   left: 0;
32   bottom: 0;
33   height: 3px;
34   width: 30px;
35   border-radius: 5px;
36   background: linear-gradient(135deg, #71b7e6, #9b59b6);
37 }
```

FILE 4: View Page.html

```
File Edit View Selection Find Packages Help
index.html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Purple Sparks</title>
8   <style>
9     body {background-color: #9966CC;
10      display: flex;
11      align-items: center;
12      justify-content: center;
13      flex-direction: column;}
14     img {height: 240px;
15      border-radius: 10px;
16      margin: 10px;}
17     h1 {color: white;}
18   </style>
19 </head>
20 <body>
21   <h1>Welcome To Purple Sparks</h1>
22   <div>
23     
24     
25     
26     
27     
28   </div>
29 </body>
30 </html>
```

FILE 5: PYTHON – App.py

```
1 from flask import Flask, render_template, request, redirect, url_for
2 import ibm_db
3
4 dsn_hostname = "61bc1829-6f45-4cd4-bef4-10cf081900bf.clogj3s0tgu0lqde00.databases.appdomain.cloud"
5 dsn_uid = "app314278" # e.g. "abc12345"
6 dsn_pwd = "2Xp9m66682uf5Rgt" # e.g. "78BZ3hdt50X06500"
7 dsn_driver = "{IBMDB2CL1}"
8 dsn_database = "b1udb" # e.g. "BLUDB"
9 dsn_port = "22304" # e.g. "22233"
10 dsn_protocol = "TCP/IP" # i.e. "TCP/IP"
11 dsn_security = "SSL" # i.e. "SSL"
12 dsn_cert = "DigitalCertificateRootCA.crt"
13 dsn = ("DRIVER={0};" "DATABASE={1};" "HOSTNAME={2};" "PORT={3};" "PROTOCOL={4};" "UID={5};" "PWD={6};" "SECURITY={7};" "SSLServerCertificate={8};").format(dsn_driver, dsn_database, dsn_hostname, dsn_port, dsn_protocol, dsn_uid, dsn_pwd, dsn_security, dsn_cert)
14 conn = ibm_db.connect(dsn, "", "")
15 app = Flask(__name__)
16 @app.route("/", methods=['GET', 'POST'])
17 def repage():
18     if request.method == 'POST':
19         USERNAME = request.form['username']
20         PHONE = request.form['phone']
21         PASSWORD = request.form['password']
22         CPASSWORD = request.form['cpassword']
23         InsertQuery = "INSERT INTO USER VALUES ('"+USERNAME+"', '"+PHONE+"', '"+PASSWORD+"', '"+CPASSWORD+"')"
24         InsertStmt = ibm_db.prepare(conn, InsertQuery)
25         ibm_db.execute(InsertStmt)
26         return redirect(url_for('loginpage'))
27     return render_template("register.html")
28 @app.route("/login", methods=['GET', 'POST'])
29 def loginpage():
30     if request.method == 'POST':
31         phone = request.form['phone']
32         password = request.form['password']
33         query = "select COUNT(*) from user where phone='"+phone+"' and password='"+password+"'"
34         stmt5 = ibm_db.exec_immediate(conn, query)
35         row = ibm_db.fetch_tuple(stmt5)
36         if (row[0] == 1):
37             return redirect(url_for('home_page'))
38     return render_template("login.html")
39 def home_page():
40     return render_template("dashboard.html")
41 if __name__ == '__main__':
42     app.run()
```

Run App.py on Command Prompt

```
C:\Windows\System32\cmd.exe - python app.py
S:\Nalaiya Thiran\SFERS>python app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with watchdog (windowsapi)
* Debugger is active!
* Debugger PIN: 997-134-709
127.0.0.1 - - [14/Nov/2022 22:06:13] "GET / HTTP/1.1" 302 -
127.0.0.1 - - [14/Nov/2022 22:06:15] "GET /index HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 22:06:20] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [14/Nov/2022 22:06:22] "GET /static/about.png HTTP/1.1" 304 -
127.0.0.1 - - [14/Nov/2022 22:06:22] "GET /static/cart.png HTTP/1.1" 304 -
127.0.0.1 - - [14/Nov/2022 22:06:22] "GET /static/chatbot.png HTTP/1.1" 304 -
127.0.0.1 - - [14/Nov/2022 22:06:23] "GET /static/background.png HTTP/1.1" 304 -
127.0.0.1 - - [14/Nov/2022 22:07:25] "GET /login HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 22:07:27] "GET /static/styles/style1.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 22:07:27] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [14/Nov/2022 22:07:33] "GET /register HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 22:07:34] "GET /static/styles/style.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 22:39:39] "GET /static/styles/style.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 22:39:39] "GET /static/styles/style.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 22:39:39] "GET /static/styles/style.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 22:39:39] "GET /static/styles/style.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 22:56:40] "POST /register HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 22:56:59] "GET /static/styles/style.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 22:57:01] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [14/Nov/2022 22:58:34] "GET /static/styles/style.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 22:58:34] "GET /static/styles/style.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 22:58:34] "GET /static/styles/style.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 23:05:43] "GET /login HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 23:05:48] "GET /static/styles/style1.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 23:05:50] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [14/Nov/2022 23:06:32] "GET /static/styles/style1.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 23:06:32] "GET /static/styles/style1.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 23:06:32] "GET /static/styles/style1.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 23:06:33] "GET /static/styles/style1.css HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 23:09:05] "GET /admin HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 23:09:08] "GET /static/style1.css HTTP/1.1" 404 -
127.0.0.1 - - [14/Nov/2022 23:09:08] "GET /static/styles/style1.css HTTP/1.1" 304 -
127.0.0.1 - - [14/Nov/2022 23:09:08] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [14/Nov/2022 23:10:40] "GET /login HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 23:10:46] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [14/Nov/2022 23:12:03] "GET /adminlogin HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 23:12:05] "GET /static/styles/style1.css HTTP/1.1" 304 -
127.0.0.1 - - [14/Nov/2022 23:12:14] "GET /admin HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 23:12:16] "GET /static/style1.css HTTP/1.1" 404 -
```

TEST SCENARIOS – ADMIN

1. Create Admin Table on DB2

The screenshot shows the IBM Db2 on Cloud console. The 'Tables' section is active, displaying a list of tables. The 'ADMIN' table is selected, and its definition is shown on the right. The table has columns: USERNAME (CHAR, 64, N, 0), EMAIL (VARCHAR, 64, N, 0), and PASSWORD (VARCHAR, 32, N, 0). A 'View data' button is visible.

Name	Data type	Nullable	Length	Scale
USERNAME	CHAR	N	64	0
EMAIL	VARCHAR	N	64	0
PASSWORD	VARCHAR	N	32	0

2. Admin Registration Page

The screenshot shows the Admin Registration page. The page has a blue and purple gradient background. A white form titled 'Admin Registration' contains fields for Username, Email, and Password. There are links for 'Forgot password' and 'SIGN IN', and a 'Signup' button.

Admin Registration

Username
Enter your username

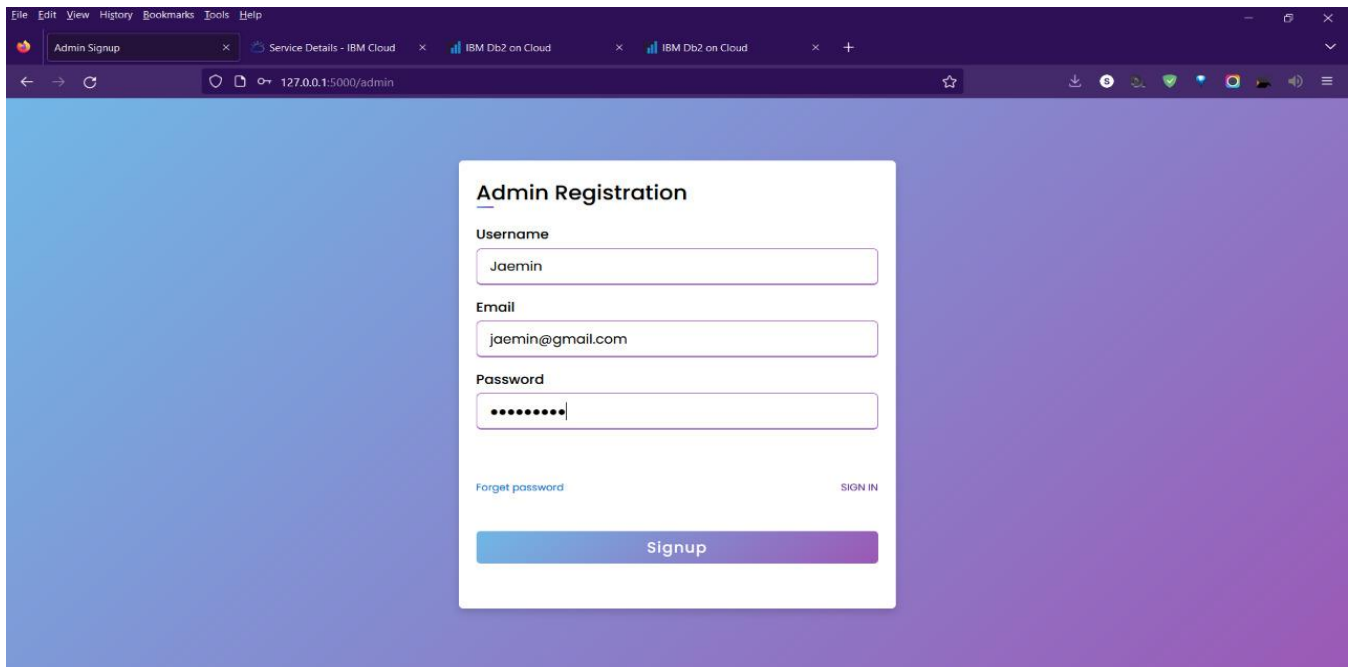
Email
Enter your Email

Password
Enter your password

[Forgot password](#) [SIGN IN](#)

Signup

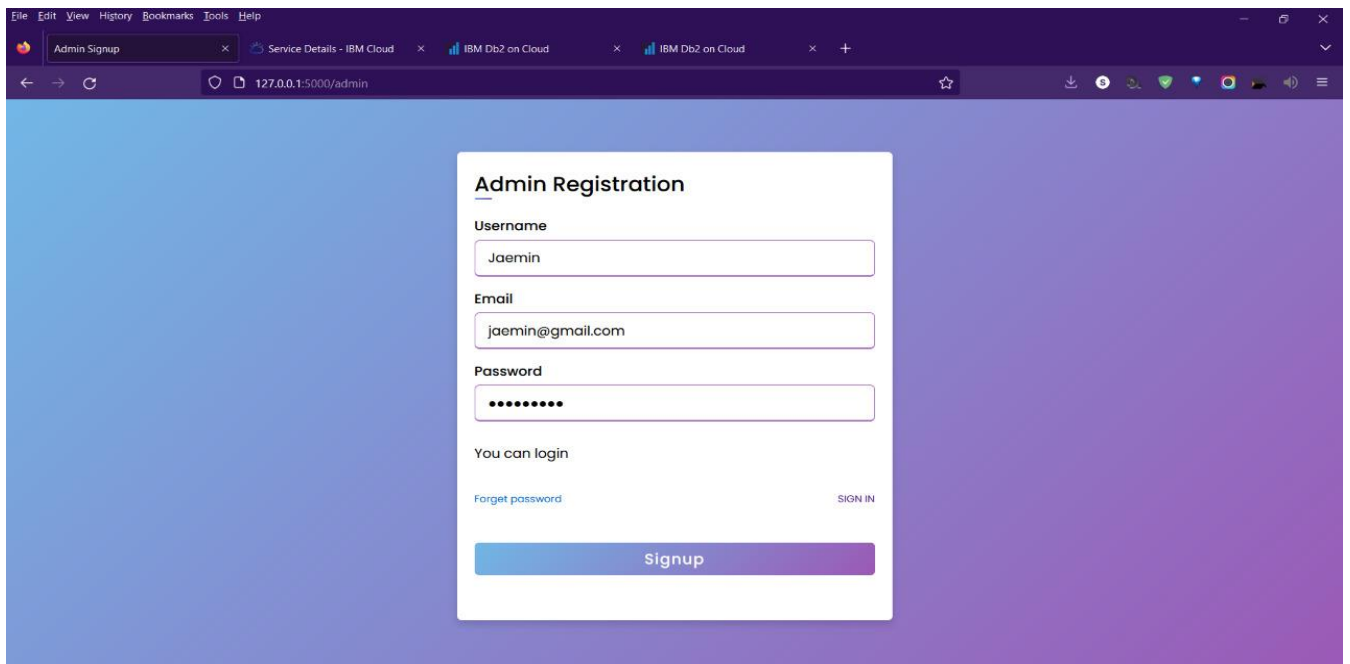
3. Enter Valid Entries of Admin Registration



A screenshot of a web browser window showing the 'Admin Registration' form. The browser's address bar displays '127.0.0.1:5000/admin'. The form is centered on a purple-to-blue gradient background. It contains the following fields and elements:

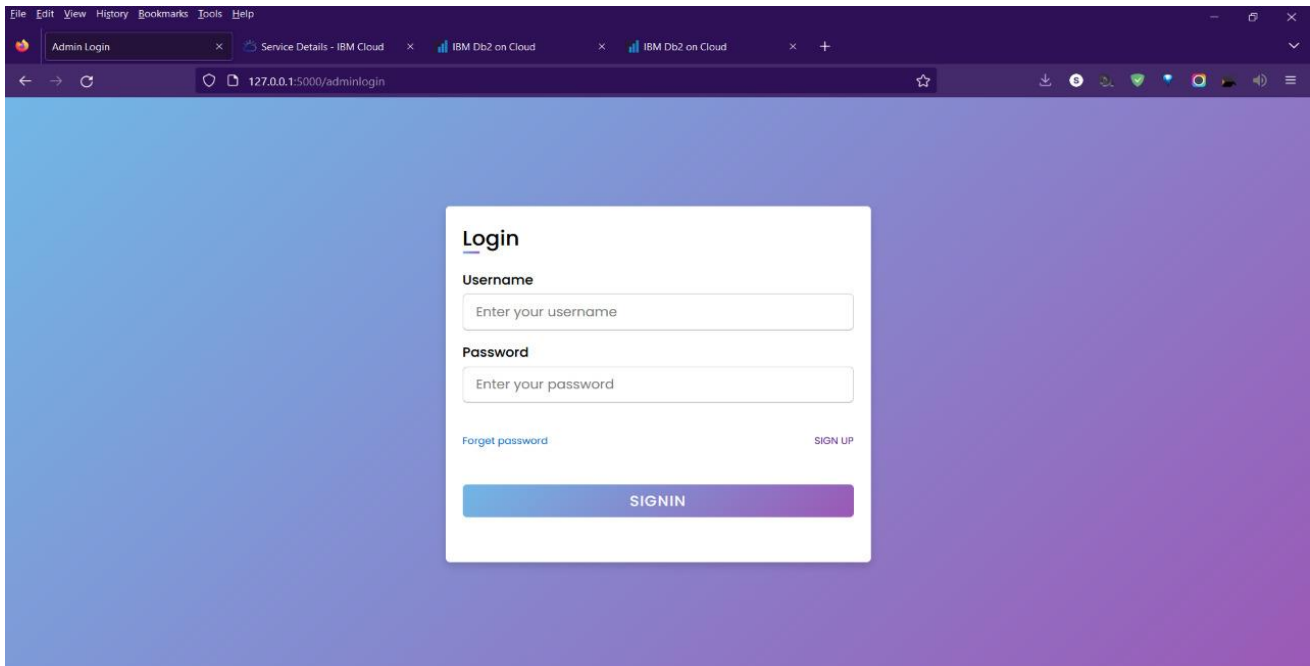
- Username:** A text input field containing 'Jaemin'.
- Email:** A text input field containing 'jaemin@gmail.com'.
- Password:** A password input field with masked characters '.....'.
- Links:** A blue link labeled 'Forget password' on the left and a blue link labeled 'SIGN IN' on the right.
- Button:** A large blue button with the text 'Signup'.

4. Successful Admin Registration



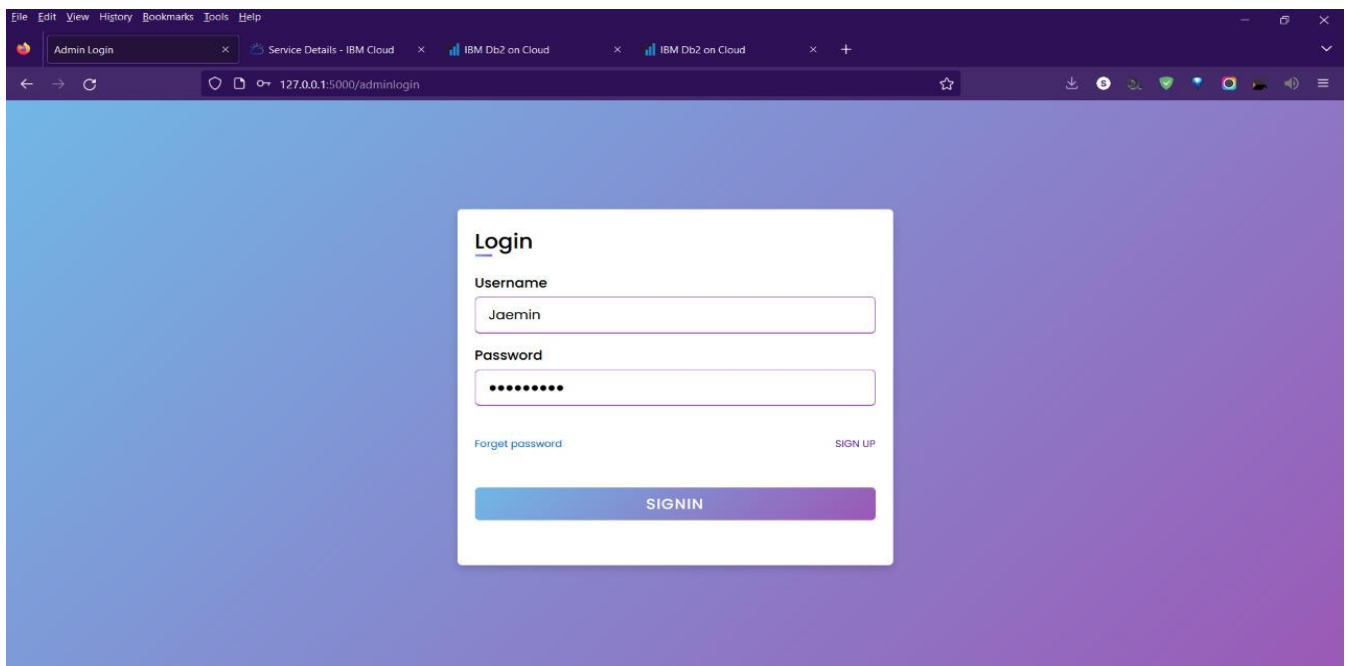
A screenshot of the same web browser window and 'Admin Registration' form as in the previous image. The form fields and links remain the same, but the 'Signup' button is now disabled, appearing in a lighter blue color. A new message, 'You can login', has been added below the password field, indicating that the registration was successful.

5. Admin Login Page



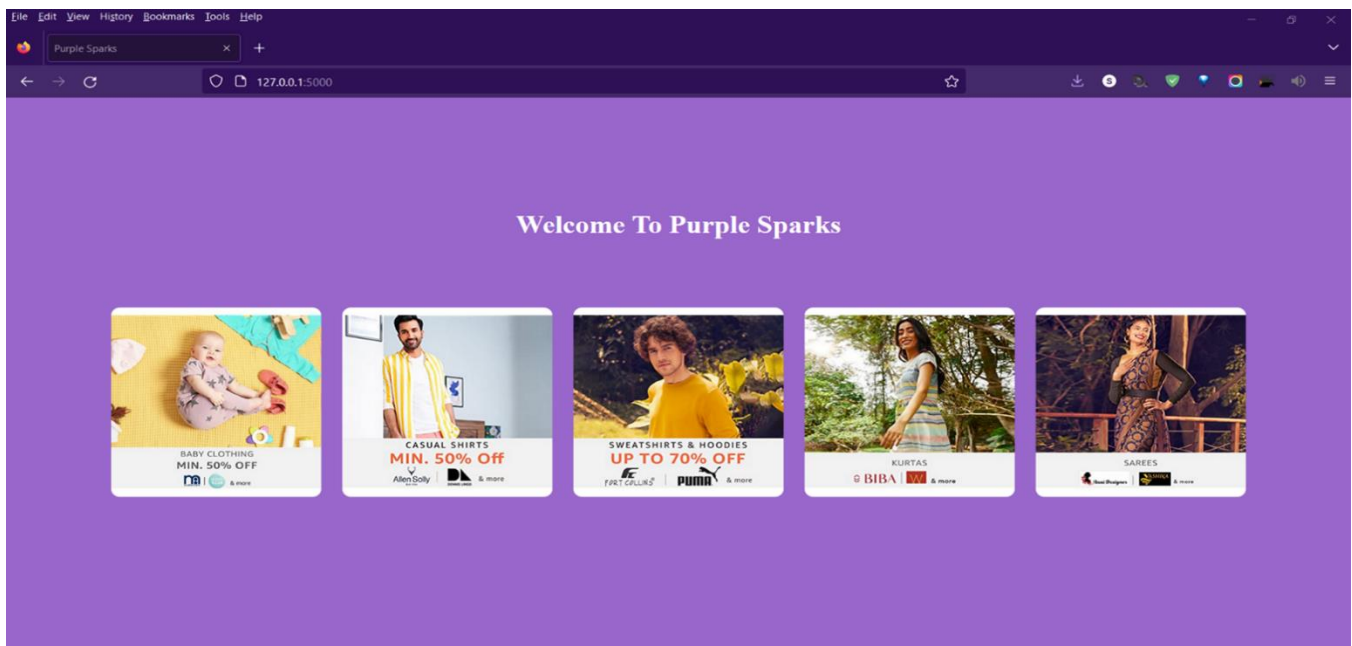
A screenshot of a web browser displaying the Admin Login page. The browser's address bar shows the URL `127.0.0.1:5000/adminlogin`. The page features a white login form centered on a blue-to-purple gradient background. The form is titled "Login" and contains two input fields: "Username" with the placeholder text "Enter your username" and "Password" with the placeholder text "Enter your password". Below the password field, there is a link for "Forgot password" and a "SIGN UP" link. At the bottom of the form is a large "SIGNIN" button.

6. Enter Valid Entries of Admin Login



A screenshot of the same Admin Login page, but with valid entries entered into the form. The "Username" field now contains the text "Jaemin", and the "Password" field is filled with ten dots, representing a masked password. The "SIGNIN" button remains visible at the bottom of the form.

7. Successful Admin Login (Dashboard of Admin)



8. Admin Database

A screenshot of the IBM Db2 on Cloud Admin Login page. The browser's address bar shows the URL 'https://bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/cm%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3A...'. The page has a dark header with the text 'IBM Db2 on Cloud'. Below the header, there is a navigation bar with tabs: 'Load Data', 'Load History', 'Tables', 'Views', 'Indexes', 'Aliases', 'MQTs', 'Sequences', and 'Application objects'. The 'Tables' tab is selected. The main content area shows a table with the title 'BPV31478.ADMIN'. The table has three columns: 'USERNAME', 'EMAIL', and 'PASSWORD'. There are two rows of data: one for 'Jaemin' and one for 'cathy'. A 'Back' button is located in the top right corner of the table area. An 'Export to CSV' button is located in the bottom right corner of the table area.

USERNAME	EMAIL	PASSWORD
Jaemin	jaemin@gmail.com	Jeonmin7@
cathy	cathy@gmail.com	Cathy7

TEST SCENARIOS – USER

1. Create User Table on DB2

The screenshot shows the IBM Db2 on Cloud web interface. The 'Tables' tab is selected, and the 'USER' table is highlighted in the list. The 'Table definition' panel on the right shows the table structure:

Name	Data type	Nullable	Length	Scale
USERNAME	CHAR	N	64	0
PHONE	VARCHAR	N	10	0
EMAIL	VARCHAR	N	64	0
PASSWORD	VARCHAR	N	32	0

The interface also shows a 'New table' button and a 'View data' button. The table definition panel indicates 'Approximate 6 rows (32.0 KB)' and 'Updated on 2022-11-12 15:52:38'.

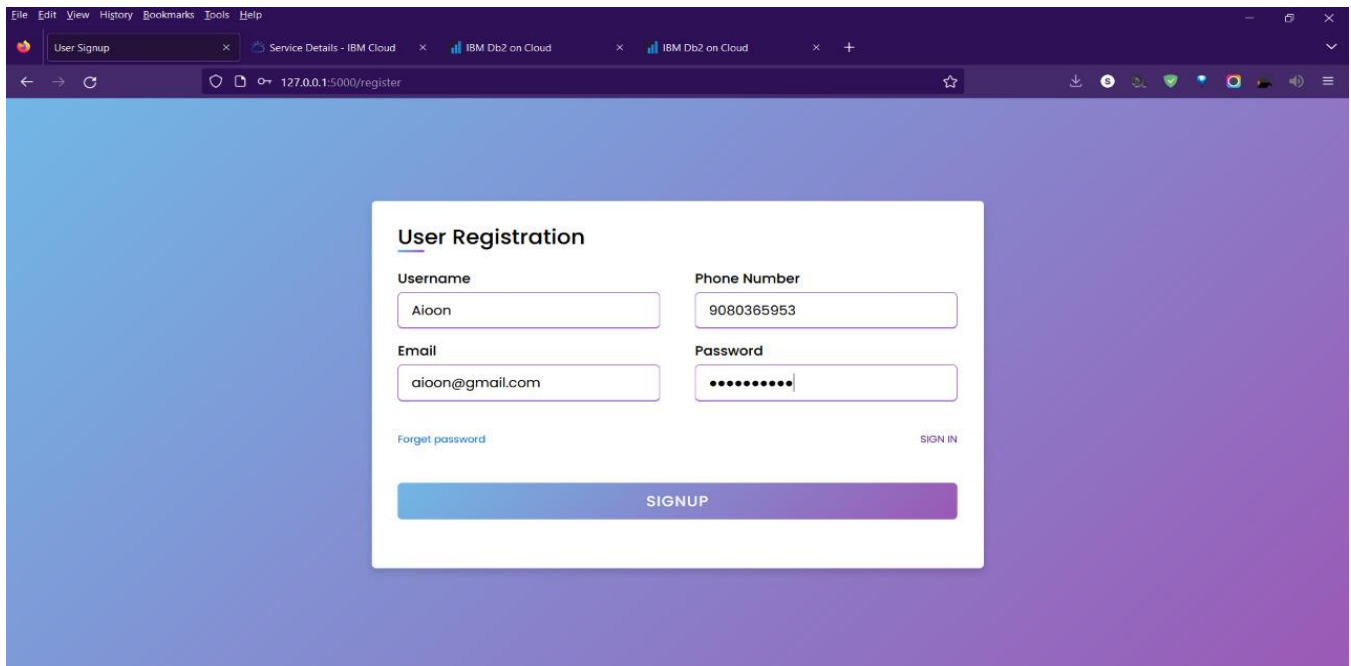
2. User Registration Page

The screenshot shows the 'User Registration' page. The page has a purple gradient background. The registration form is centered and contains the following fields:

- Username**: Enter your username
- Phone Number**: Enter your phone number
- Email**: Enter your Email
- Password**: Enter your password

Below the form, there is a 'Forgot password' link and a 'SIGN IN' link. At the bottom of the form, there is a large 'SIGNUP' button.

3. Enter Valid Entries of User Registration

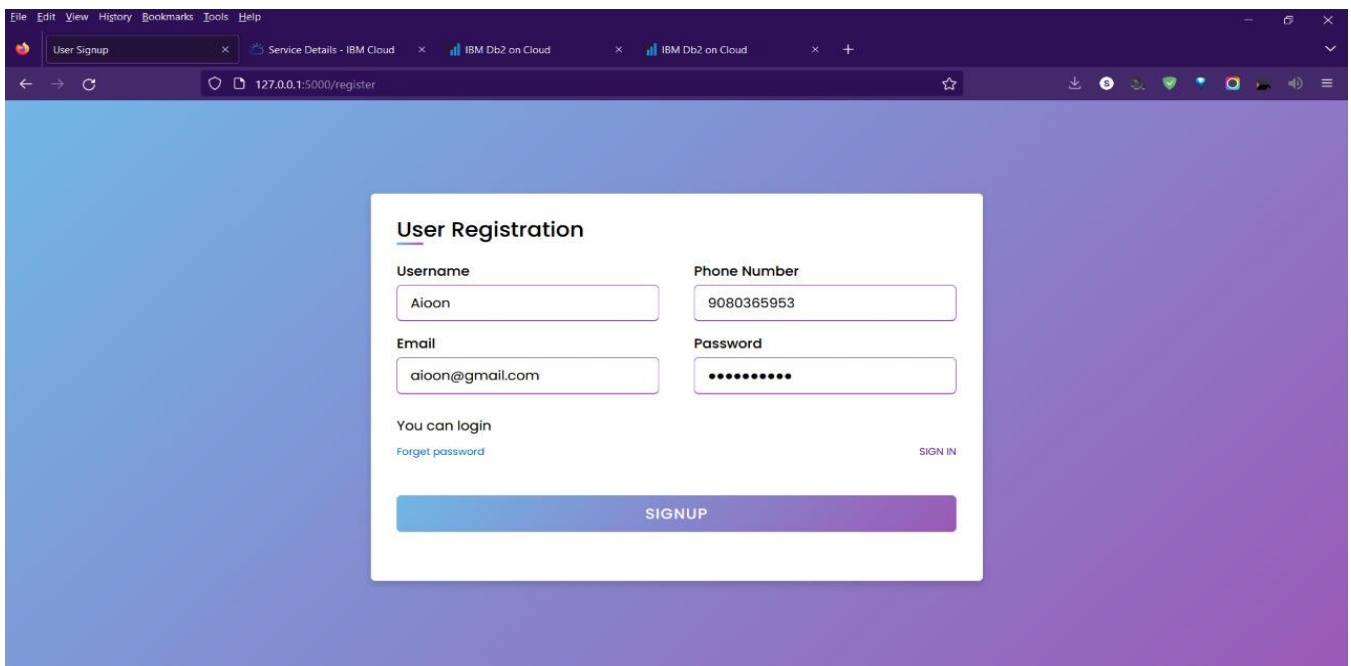


The screenshot shows a web browser window with the address bar displaying "127.0.0.1:5000/register". The page features a "User Registration" form with the following fields and values:

- Username:** Aloon
- Phone Number:** 9080365953
- Email:** aloon@gmail.com
- Password:** (masked with dots)

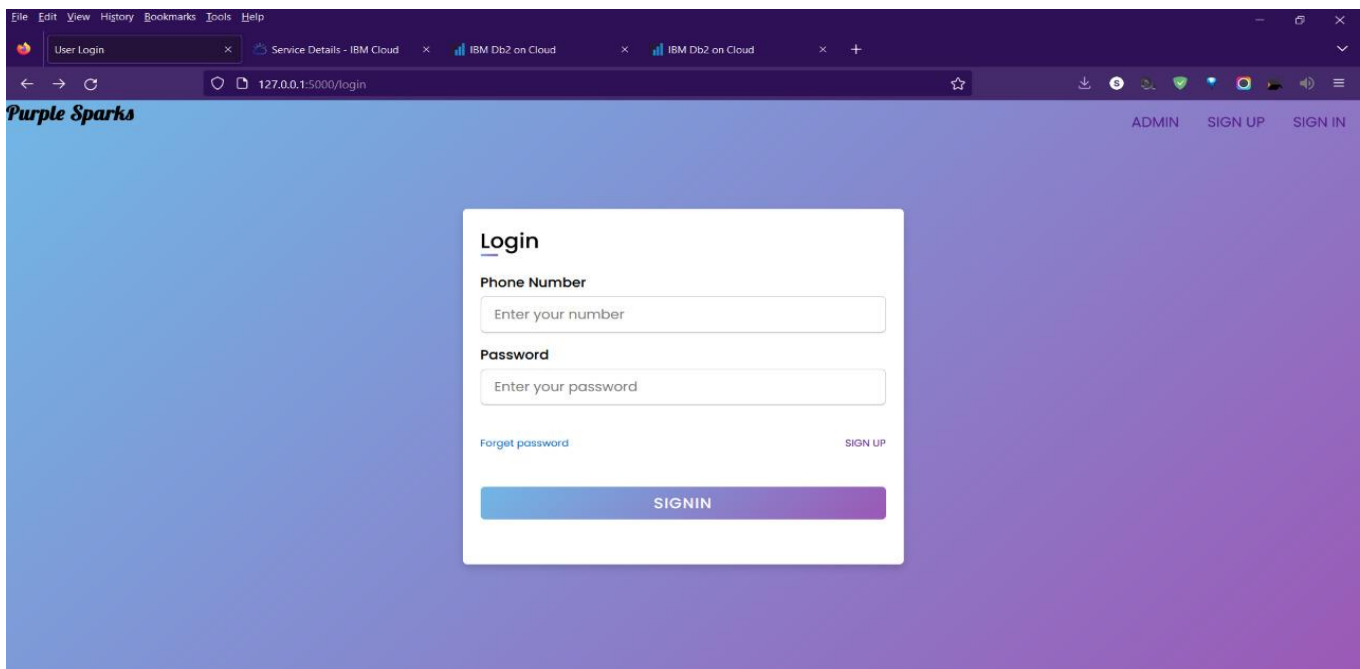
Below the form, there is a "Forget password" link and a "SIGN IN" link. A large blue "SIGNUP" button is positioned at the bottom of the form.

4. Successful User Registration

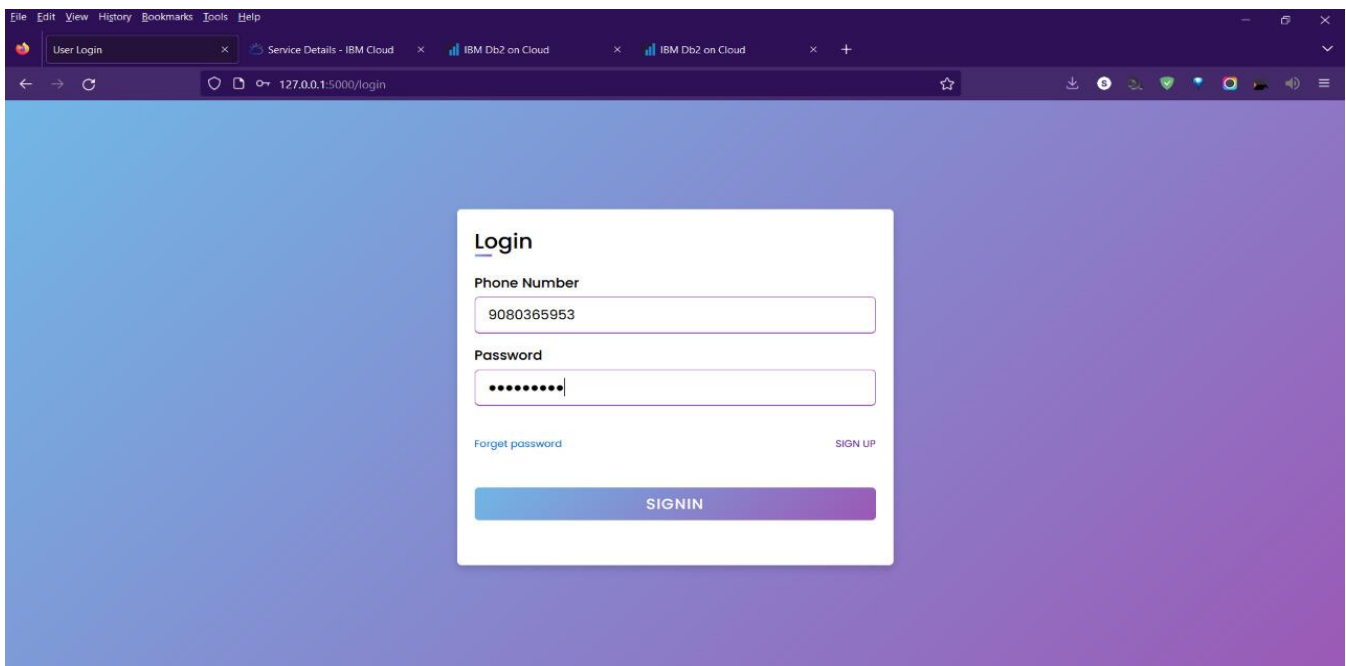


The screenshot shows the same web browser window as in the previous image, but with an additional message below the form: "You can login". The "SIGNUP" button remains visible at the bottom of the form.

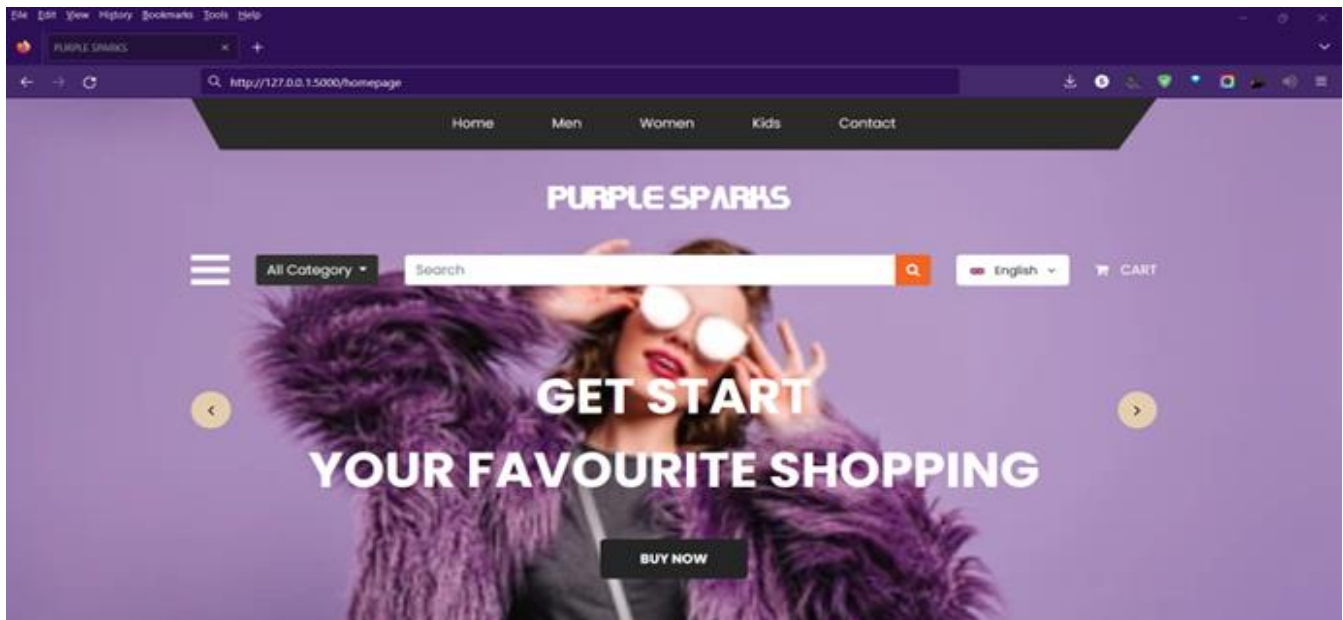
5. User Login Page



6. Enter Valid Entries of User Login



7. Successful User Login (Dashboard of User)



8. User Database

The screenshot shows the IBM Db2 on Cloud console interface. The browser's address bar displays a URL starting with 'https://bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/'. The console shows the 'BPV31478.USER' table. The table has four columns: 'USERNAME', 'PHONE', 'EMAIL', and 'PASSWORD'. There are two rows of data. The first row has values 'Aioon', '9080365953', 'aioon@gmail.com', and 'Aioontan7@'. The second row has values 'shayu', '7897897895', 'shayu@gmail.com', and 'Shayu32'. The console also includes a 'Back' button and an 'Export to CSV' button.

USERNAME	PHONE	EMAIL	PASSWORD
Aioon	9080365953	aioon@gmail.com	Aioontan7@
shayu	7897897895	shayu@gmail.com	Shayu32