

PROJECT DEVELOPMENT PHASE

SPRINT-2

Date	16 Nov 2022
Team ID	PNT2022TMID49820
Project Name	Smart Fashion Recommender Application
Team Members	M.Karthick Raja R.Hirthic Shyam A.Edwin kirubakaran R.Muthu Karthick

The screenshot displays a web browser window with multiple tabs open. The address bar shows the URL 127.0.0.1:5000. The browser's taskbar at the bottom includes icons for Gmail, YouTube, Maps, Docker, and WhatsApp. The web application has a red navigation bar with links for Home, About, and Register. The main content area is light blue and contains two login forms: 'Admin Login' and 'User Login'. Both forms are purple boxes with 'Username' and 'Password' input fields and a 'Submit LOGIN' button. A security warning for 'DigiCertGlobalRoot....crt' is visible in the bottom left, and a 'Show all' link is in the bottom right.

ibm data | Python T | Service D | Getting s | Getting s | IBM Db2 | 500 Inter | HTML Re | Inbox (16 | Smart Fa | User Pan | +

127.0.0.1:5000

Gmail YouTube Maps Get Docker | Docke... (29) WhatsApp

Home Home About Register

Admin Login

Username :
Password :
Submit LOGIN

User Login

Username :
Password :
Submit LOGIN

DigiCertGlobalRoot....crt

Show all

Admin Should Enter the Correct Username & Password:

ibm data: x Python T: x Service D: x Getting s: x Getting s: x IBM Db2: x 500 Inter: x HTML Re: x Inbox (18: x Smart Fa: x User Pan: x +

127.0.0.1:5000

Gmail YouTube Maps Get Docker | Docke... (29) WhatsApp

Home Home About Register

Admin Login

Username :

Password :

LOGIN

User Login

Username :

Password :

LOGIN

DigiCertGlobalRoot.... crt Show all x

View User Cart:

Form: x Blocl: x Othe: x How: x GitH: x Hack: x medi: x unbl: x GitH: x IBM: x githu: x Paym: x canv: x Abou: x CART: x Cont: x Cont: x +

file:///C:/Users/hirthic/Pictures/IBM-Project-33153-1660215117-main/Project%20Developm

bugbounty Itprojects Amazon.in AliExpress Suggested Sites HttpSession in Serv...

SHOPPING CART

Item: Quantity:

Shopping List

Item	Value
empty	empty

1:12 AM 11/17/2022

Database Connection:

```
from flask import *
```

```
from flask_cors import CORS
```

```
import ibm_db
```

```
import re
```

```
app = Flask(__name__)
```

```
CORS(app)
```

```
cors =CORS(app, resources={
```

```
    r"/*":{
```

```
        "origins": "*"
```

```
    }
```

```
})
```

```
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=0c77d6f2-5da9-48a9-81f8-
```

```
86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=3119
```

```
8;SECURITY=SSL;SSLServiceCertificate=DigiCertGlobalRootCA.crt;UID=kj  
f40222;PWD=mbSqNCJVKq3Ecbgi",",")
```

```
@app.route("/")
```

```
def hello():
```

```
    return render_template('index.html')
```

```

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

def login():

    uname=request.form['username']

    passwd=request.form['password']

    sql = "SELECT * FROM users WHERE USERNAME =? AND
PASSWORD=?"

    try:

        stmt = ibm_db.prepare(conn, sql)

        ibm_db.bind_param(stmt,1,uname)

        ibm_db.bind_param(stmt,2,passwd)

        ibm_db.execute(stmt)

        account = ibm_db.fetch_assoc(stmt)

        print (account)

        if account["USERNAME"] == uname and account["PASSWORD"] ==
passwd:

            msg = 'Logged in successfully !'

            print(msg)

            return redirect(url_for("user"))

        else:

            msg = 'Incorrect username / password !'

            print(msg)

```

```
        return render_template('index.html')

except:

    print("Error")

    return redirect(url_for("success"))

else:

    print("complete")
```

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

```
def register():
```

```
    msg = " "
```

```
    if request.method == 'POST' :
```

```
        username = request.form['username']
```

```
        password = request.form['password']
```

```
        try:
```

```
            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 !'
```

```
    print(msg)
```

```
    return redirect(url_for("reg"))
```

```
else:
```

```
    print("running")
```

```
    print(username)
```

```
    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, password)
```

```
    ibm_db.execute(prep_stmt)
```

```
    msg = 'You have successfully registered !'
```

```
    print(msg)
```

```
    return redirect(url_for("user"))
```

```
except:
```

```
    print("error")
```

```
    return redirect(url_for("reg"))
```

```
return render_template('register.html')
```

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

```
def reg():
```

```
    return render_template('register.html')
```

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

```
def about():
```

```
    return render_template('About.html')
```

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

```
def admin():
```

```
    return render_template('Admin.html')
```

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

```
def cart():
```

```
    return render_template('Cart.html')
```

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

```
def contact():
```

```
    return render_template('Contact.html')
```

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

```
def detail():  
    return render_template('Detail.html')  
  
@app.route('/items')  
def items():  
    return render_template('Items.html')  
  
@app.route('/success')  
def success():  
    return render_template('Success.html')  
  
@app.route('/user')  
def user():  
    return render_template('User.html')  
  
if __name__ == "__main__":  
    app.run()
```