

Project Development Phase Delivery of Sprint-1

Date
Team ID
Project Name

02 November 2022
PNT2022TMID37500
Inventory Management System For Retailer

app.py Creating flask application

```
import re
import ibm_db
from flask import Flask, jsonify, render_template, request, session
from flask_cors import CORS

app = Flask(__name__)
CORS(app)

conn = ibm_db.connect(
    "DATABASE=bludb;HOSTNAME=6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30376;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=yzk69108;PWD=vL0zrqj9PeaSutC4", '', ''
)
print(conn)
print("connection successful..")

app.secret_key = 'a'

@app.route('/')
def hello_world():
    return 'Hello World'

# post.text = request.form.get('body','')
@app.route('/sign-up', methods=['POST'])
def register():
    msg = ''
    if request.method == 'POST':
        username = request.json.get("username", None)
        email = request.json.get("email", None)
        password = request.json.get("password", None)
        sql = "SELECT * FROM users WHERE email =?"
        print(sql)
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt, 1, email)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account, "hello")
        if account:
            return {"msg": "Account already exist"}, 401
        elif not re.match(r'^@+@[^@]+\.[^@]+', email):
            return {"msg": "invalid email address"}, 401
        elif not re.match(r'[A-Za-z0-9]+', username):
```

```

        return {"msg": "name must contain only char and numbers"}, 401
    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, email)
        ibm_db.bind_param(prepare_stmt, 3, password)
        ibm_db.execute(prepare_stmt)
        msg = 'You have successfully registered !', 200
    elif request.method == 'POST':
        msg = 'Please fill out the form !'

    return {"msg":msg,"success":True}

```

```

@app.route('/sign-in', methods = ['POST'])
def login():
    global userid
    msg = ''
    if request.method == 'POST' :
        email = request.json.get("email", None)
        password = request.json.get("password", None)
        sql = "SELECT * FROM users WHERE email =? AND password=?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,email)
        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']
            return {"msg": "user loggedin successfully","success":True}, 200
        else:
            return {"msg": "Incorrect username / password!"}, 400

```

```

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

```

Frontend – Reactjs

home.js

```

import React from 'react';

function Home(props) {
    return (
        <div>
            Home
        </div>
    );
}

export default Home;

```

signup.js

```
import React, { useState } from 'react';
import { useNavigate } from 'react-router-dom';
export default function Signup() {

  const [Username, setUsername] = useState()
  const [Email, setEmail] = useState()
  const [Password, setPassword] = useState()
  const navigate = useNavigate();
  const signup = async () => {

    const postdata = { username: Username, email: Email, password: Password }
    const response = await fetch('http://127.0.0.1:5000/sign-up', {
      method: 'POST',
      headers: {
        'Content-Type': 'application/json',
      },
      body: JSON.stringify(postdata),
    })
    const data = await response.json()
    console.log(data.msg, data.success);
    if (data.success == true) {
      window.alert(data.msg)
      navigate("/signin")
    } else {
      window.alert(data.msg)
    }
  }

  return (
    <div className="relative flex flex-col justify-center min-h-screen overflow-hidden">
      <div className="w-full p-6 m-auto bg-white rounded-md shadow-md lg:max-w-xl">
        <h1 className="text-3xl font-semibold text-center text-purple-700 underline">
          Sign Up
        </h1>
        <form className="mt-6">
          <div className="mb-2">
            <label className="block text-sm font-semibold text-gray-800">username</label>
            <input type="text" onChange={(e) => setUsername(e.target.value)}
              className="block w-full px-4 py-2 mt-2 text-purple-700 bg-white border rounded-md
              focus:border-purple-400 focus:ring-purple-300 focus:outline-none focus:ring focus:ring-opacity-40"
            />
          </div>
          <div className="mb-2">
            <label className="block text-sm font-semibold text-gray-800">Email</label>
            <input type="email" onChange={(e) => setEmail(e.target.value)} className="block
              w-full px-4 py-2 mt-2 text-purple-700 bg-white border rounded-md focus:border-purple-400
              focus:ring-purple-300 focus:outline-none focus:ring focus:ring-opacity-40" />
          </div>
          <div className="mb-2">
```

```

        <label className="block text-sm font-semibold text-gray-800" >Password</label>
        <input type="password" onChange={e => setPassword(e.target.value)}
className="block w-full px-4 py-2 mt-2 text-purple-700 bg-white border rounded-md
focus:border-purple-400 focus:ring-purple-300 focus:outline-none focus:ring focus:ring-opacity-40"
/>
    </div>
    <div className="mt-6">
        <button onClick={signup} type="button" className="w-full px-4 py-2 tracking-
wide text-white transition-colors duration-200 transform bg-purple-700 rounded-md hover:bg-
purple-600 focus:outline-none focus:bg-purple-600">Create Account</button>
    </div>
</form>

<p className="mt-8 text-xs font-light text-center text-gray-700">
    {" "}
    Already have an account? {" "}
    <a href="/signin" className="font-medium text-purple-600 hover:underline">
        Sign in
    </a>
</p>
</div>
</div>
);
}

```

signin.js

```

import React, { useState } from 'react';
import { useNavigate } from 'react-router-dom';

export default function SignIn() {

    const [Email, setEmail] = useState()
    const [Password, setPassword] = useState()
    const navigate = useNavigate();
    const signin = async () => {

        const postdata = { email: Email, password: Password }
        const response = await fetch('http://127.0.0.1:5000/sign-in', {
            method: 'POST',
            headers: {
                'Content-Type': 'application/json',
            },
            body: JSON.stringify(postdata),
        })
        const data = await response.json()
        console.log(data.msg, data.success);
        if (data.success === true) {
            window.alert(data.msg)
            navigate("/home")
        } else {
            window.alert(data.msg)
        }
    }
}

```

```

    }
  }
}

```

```

return (
  <div className="relative flex flex-col justify-center min-h-screen overflow-hidden">
    <div className="w-full p-6 m-auto bg-white rounded-md shadow-md lg:max-w-xl">
      <h1 className="text-3xl font-semibold text-center text-purple-700 underline">
        Sign in
      </h1>
      <form className="mt-6">
        <div className="mb-2">
          <label className="block text-sm font-semibold text-gray-800">Email</label>
          <input onChange={(e) => setEmail(e.target.value)} type="email" className="block
w-full px-4 py-2 mt-2 text-purple-700 bg-white border rounded-md focus:border-purple-400
focus:ring-purple-300 focus:outline-none focus:ring focus:ring-opacity-40" />
        </div>
        <div className="mb-2">
          <label className="block text-sm font-semibold text-gray-800">Password</label>
          <input onChange={(e) => setPassword(e.target.value)} type="password"
className="block w-full px-4 py-2 mt-2 text-purple-700 bg-white border rounded-md
focus:border-purple-400 focus:ring-purple-300 focus:outline-none focus:ring focus:ring-opacity-40"
/>
        </div>
        {/* <a href="#" className="text-xs text-purple-600 hover:underline"> Forget
Password? </a> */}
        <div className="mt-6">
          <button onClick={signin} type="button" className="w-full px-4 py-2 tracking-
wide text-white transition-colors duration-200 transform bg-purple-700 rounded-md hover:bg-
purple-600 focus:outline-none focus:bg-purple-600"> Sign In</button>
        </div>
      </form>

      <p className="mt-8 text-xs font-light text-center text-gray-700">
        {" "}
        Don't have an account? {" "}
        <a
          href="/signup"
          className="font-medium text-purple-600 hover:underline"
        >
          Sign up
        </a>
      </p>
    </div>
  </div>
);
}

```

OUTPUT:

