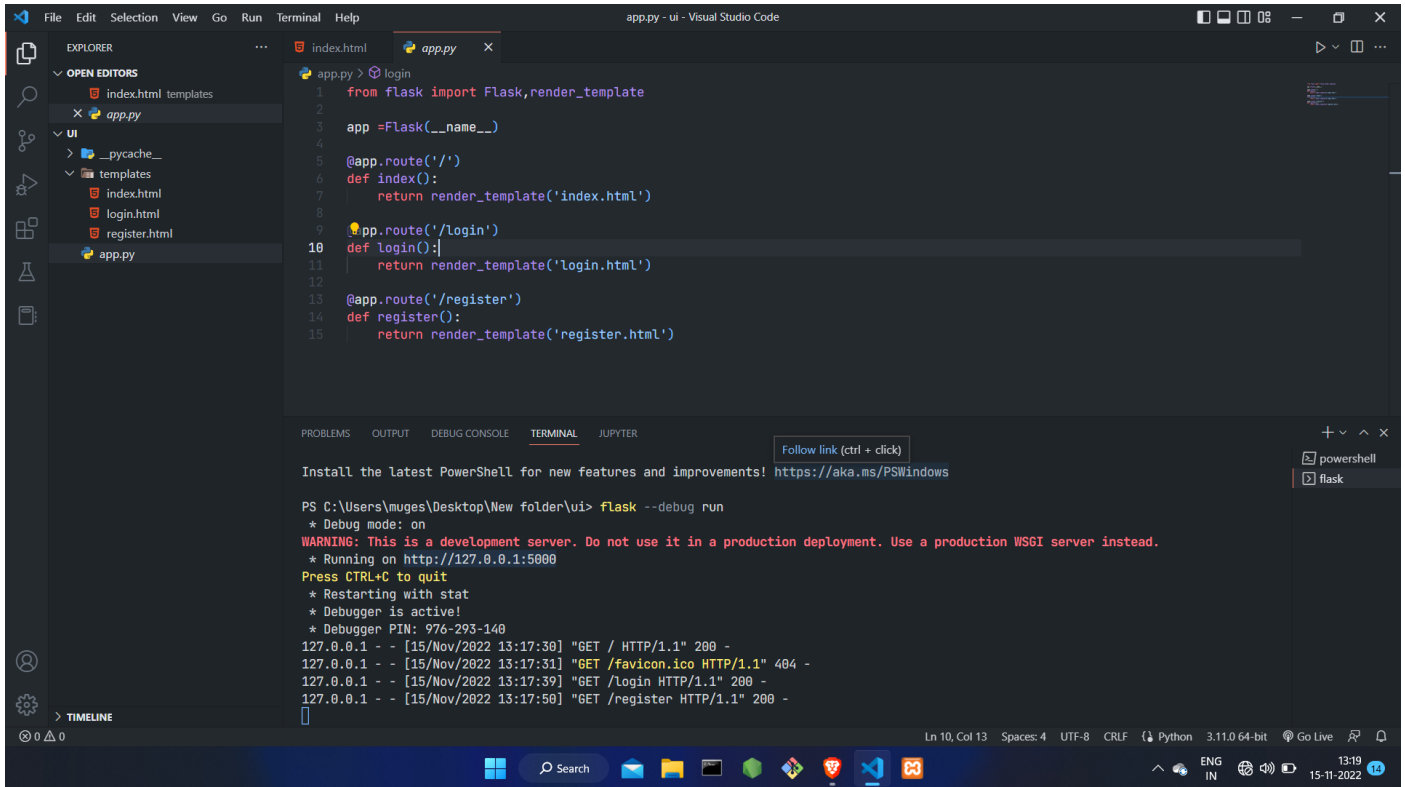


IMPLEMENTING WEB APPLICATION

Create UI to interact with Application

Date	05.11.2022
Team ID	PNT2022TMID48220
Project Name	Inventory Management System for Retailers

Coding for Simple User Interface



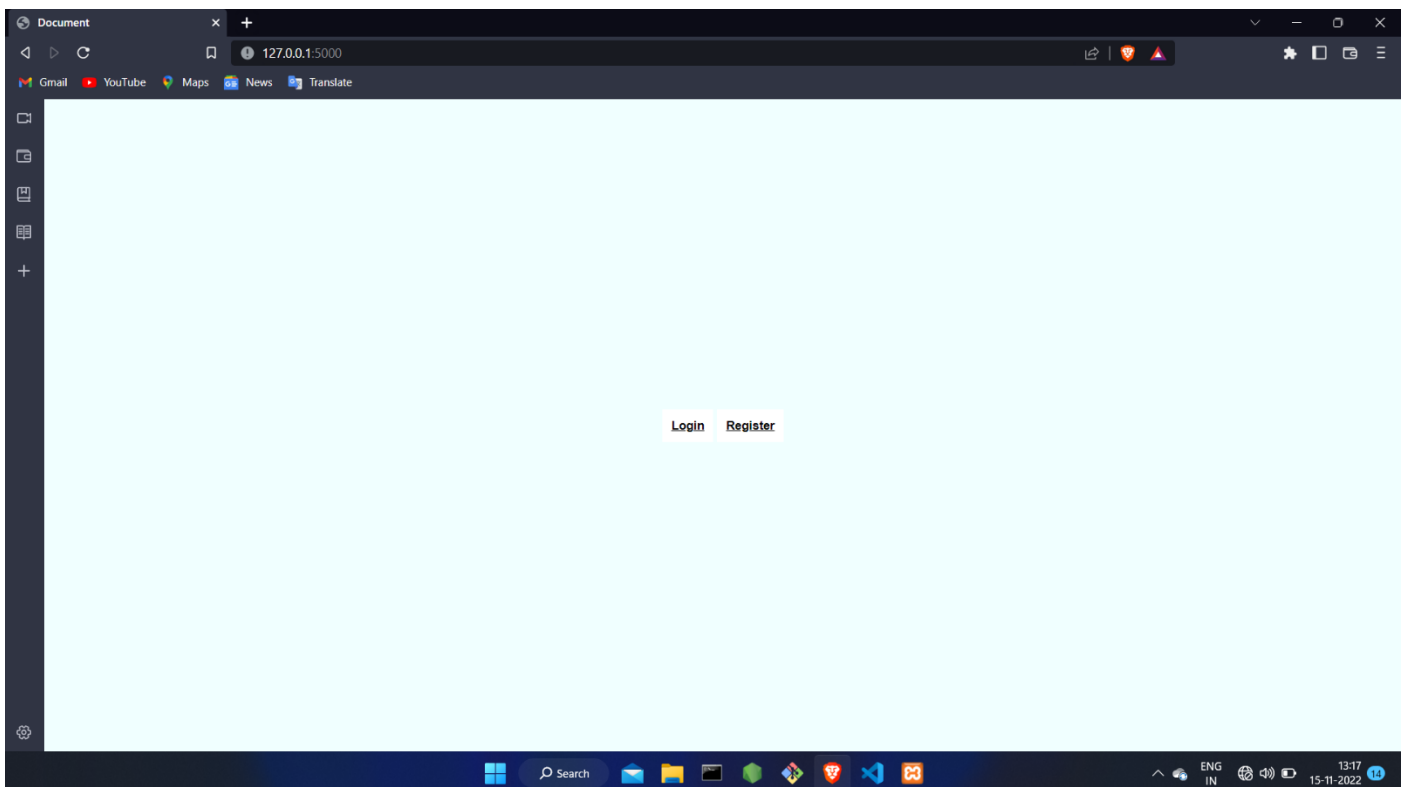
The screenshot shows the Visual Studio Code editor with a Flask application. The Explorer pane on the left shows the project structure: `index.html`, `templates`, `app.py`, `login.html`, and `register.html`. The main editor displays the `app.py` file with the following code:

```
1 from flask import Flask, render_template
2
3 app = Flask(__name__)
4
5 @app.route('/')
6 def index():
7     return render_template('index.html')
8
9 @app.route('/login')
10 def login():
11     return render_template('login.html')
12
13 @app.route('/register')
14 def register():
15     return render_template('register.html')
```

The TERMINAL pane at the bottom shows the command prompt output:

```
PS C:\Users\muges\Desktop\New folder\ui> flask --debug run
* 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 stat
* Debugger is active!
* Debugger PIN: 976-293-140
127.0.0.1 - - [15/Nov/2022 13:17:30] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [15/Nov/2022 13:17:31] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [15/Nov/2022 13:17:39] "GET /login HTTP/1.1" 200 -
127.0.0.1 - - [15/Nov/2022 13:17:50] "GET /register HTTP/1.1" 200 -
```

Index page



Registration Page

Document

127.0.0.1:5000/register

Gmail YouTube Maps News Translate

Email

Username

Pasword

Register

Search

ENG IN 13:17 15-11-2022

Login Page

Document

127.0.0.1:5000/login

Gmail YouTube Maps News Translate

Username

Pasword

Login

Search

ENG IN 13:17 15-11-2022