**App.py**

from flask import Flask, render\_template, request, redirect, url\_for, session

from flask\_socketio import SocketIO, join\_room, leave\_room

app = Flask(\_\_name\_\_)

app.secret\_key = 'your\_secret\_key'

socketio = SocketIO(app)

# In-memory storage for chat rooms and users (Replace with a database in production)

chat\_rooms = {}

users = {}

# Route for home page (login or create account)

@app.route('/')

def home():

return render\_template('home.html')

# User registration route

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

def register():

username = request.form.get('username')

password = request.form.get('password')

# Store user information (In production, use a secure method)

users[username] = {'password': password}

return redirect('/login')

# User login route

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

def login():

if request.method == 'POST':

username = request.form.get('username')

password = request.form.get('password')

# Check if the user exists and the password is correct

if username in users and users[username]['password'] == password:

session['username'] = username

return redirect('/chatrooms')

else:

return redirect('/')

return render\_template('login.html')

# Chat room list route

@app.route('/chatrooms')

def chatrooms():

if 'username' in session:

return render\_template('chatrooms.html', chat\_rooms=chat\_rooms)

else:

return redirect('/chat')

# Chat room route

@app.route('/chat/<room\_id>')

def chat(room\_id):

if 'username' in session:

if room\_id in chat\_rooms:

return render\_template('chat.html', room\_id=room\_id)

else:

return redirect('/chatrooms')

return redirect('/')

# WebSocket event for joining a chat room

@socketio.on('join\_room')

def handle\_join\_room(data):

username = session['username']

room\_id = data['room\_id']

join\_room(room\_id)

socketio.emit('user\_join', {'username': username}, room=room\_id)

# WebSocket event for leaving a chat room

@socketio.on('leave\_room')

def handle\_leave\_room(data):

username = session['username']

room\_id = data['room\_id']

leave\_room(room\_id)

socketio.emit('user\_leave', {'username': username}, room=room\_id)

# WebSocket event for sending messages

@socketio.on('send\_message')

def handle\_send\_message(data):

username = session['username']

room\_id = data['room\_id']

message = data['message']

socketio.emit('receive\_message', {'username': username, 'message': message}, room=room\_id)

if \_\_name\_\_ == '\_main\_':

socketio.run(app, debug=True)

app.run()

**login.html**

<!DOCTYPE html>

<html>

<head>

<title>Login</title>

</head>

<body>

<h1>Login</h1>

<form action="/login" method="POST">

<label for="username">Username:</label>

<input type="text" name="username" required>

<label for="password">Password:</label>

<input type="password" name="password" required>

<button type="submit">Login</button>

</form>

</body>

</html>

**Home.html**

<!DOCTYPE html>

<html>

<head>

<title>Chat App</title>

</head>

<body>

<h1>Welcome to Chat App</h1>

<h2>Create an account or log in</h2>

<form action="/register" method="POST">

<label for="username">Username:</label>

<input type="text" name="username" required>

<label for="password">Password:</label>

<input type="password" name="password" required>

<button type="submit">Register</button>

</form>

<form action="/login" method="POST">

<label for="username">Username:</label>

<input type="text" name="username" required>

<label for="password">Password:</label>

<input type="password" name="password" required>

<button type="submit">Login</button>

</form>

</body>

</html>

**Chatrooms.html**

<DOCTYPE html>

<html>

<head>

<title>Chat Rooms</title>

</head>

<body>

<h1>Chat Rooms</h1>

<ul>

{% for room\_id in chat\_rooms %}

<li>

<a href="/chat/{{ room\_id }}">Join {{ room\_id }}</a>

</li>

{% endfor %}

</ul>

</body>

</html>

**Chat.html**

<!DOCTYPE html>

<html>

<head>

<title>Chat Room</title>

</head>

<body>

<h1>Chat Room: {{ room\_id }}</h1>

<div id="chat-messages"></div>

<textarea id="message-input" placeholder="Type your message..."></textarea>

<button id="send-button">Send</button>

<script src="https://cdnjs.cloudflare.com/ajax/libs/socket.io/4.0.1/socket.io.js"></script>

<script>

var socket = io.connect('http://' + document.domain + ':' + location.port);

// Join the chat room

socket.emit('join\_room', { room\_id: "{{ room\_id }}" });

// Handle received messages

socket.on('receive\_message', function(data) {

var message = data.username + ": " + data.message + "<br>";

document.getElementById('chat-messages').innerHTML += message;

});

// Send a message when clicking the "Send" button

document.getElementById('send-button').onclick = function() {

var message = document.getElementById('message-input').value;

socket.emit('send\_message', { room\_id: "{{ room\_id }}", message: message });

document.getElementById('message-input').value = '';

};

// Handle pressing Enter key to send a message

document.getElementById('message-input').addEventListener('keydown', function (e) {

if (e.key === 'Enter') {

document.getElementById('send-button').click();

}

});

// Leave the chat room when closing the tab

window.onbeforeunload = function() {

socket.emit('leave\_room', { room\_id: "{{ room\_id }}" });

};

</script>

</body>

</html>