**Personal Blog Using IBM Cloud**

**(Abhiraj Singh)**

**PHASE 3**

**Features :**

1. **User Authentication**: Allow users to create accounts and log in. This enables you to have a personalized experience and manage user-specific content.

2. **User Profiles**: Let users create and edit their profiles, including avatars, bios, and other personal information.

3. **Content Management**: Implement a content management system (CMS) to write, edit, and delete blog posts. You can also categorize posts or create tags to organize content.

4. **Markdown Support**: Offer Markdown support for writing blog posts. Markdown is a lightweight markup language that allows you to format text without complex HTML.

5. **WYSIWYG Editor**: Include a What You See Is What You Get (WYSIWYG) editor for users who aren't comfortable with Markdown.

6. **Comments**: Allow readers to comment on blog posts. This encourages engagement and interaction.

**6. User Interaction:**

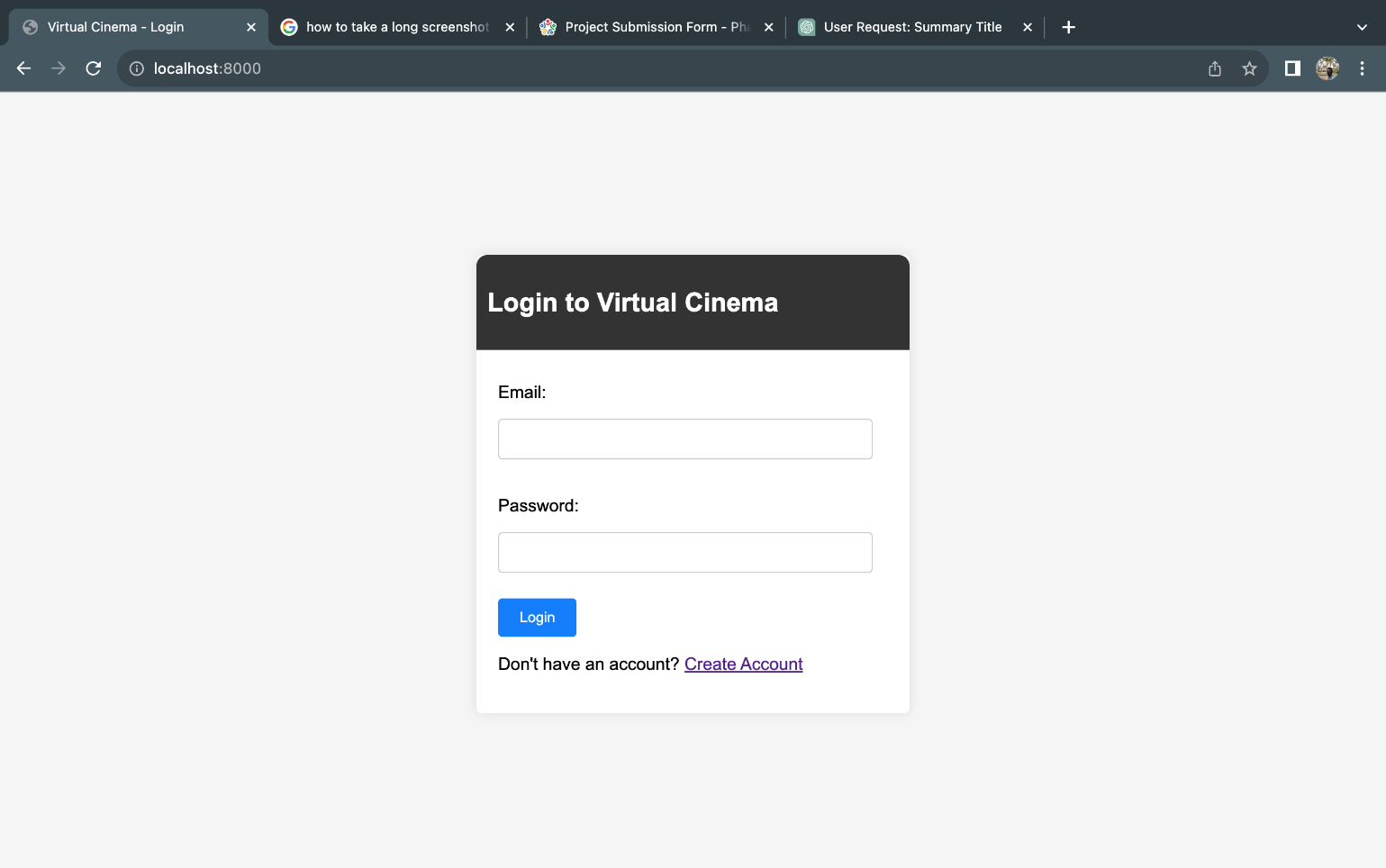
**Comments and Reviews:** Enable users to leave comments and reviews on movies. This can foster community engagement and help others decide what to watch.

**Ratings**: Allow users to rate movies on a scale or with a star system. Aggregate these ratings to provide average scores and help users make informed choices.

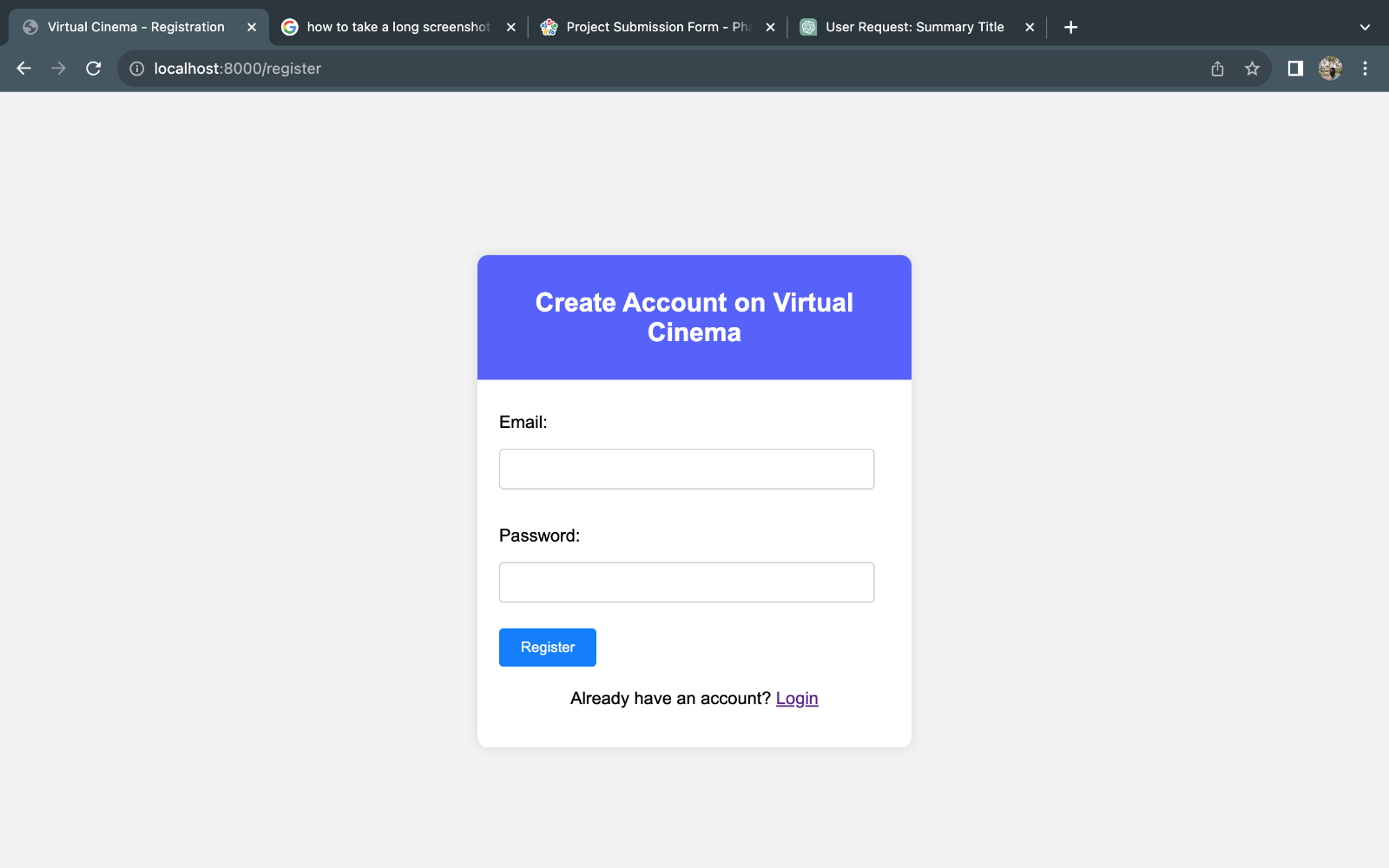
**7. Payment Processing:**

Set up payment gateways to handle subscription payments and one-time movie rentals. Ensure secure and reliable payment processing.

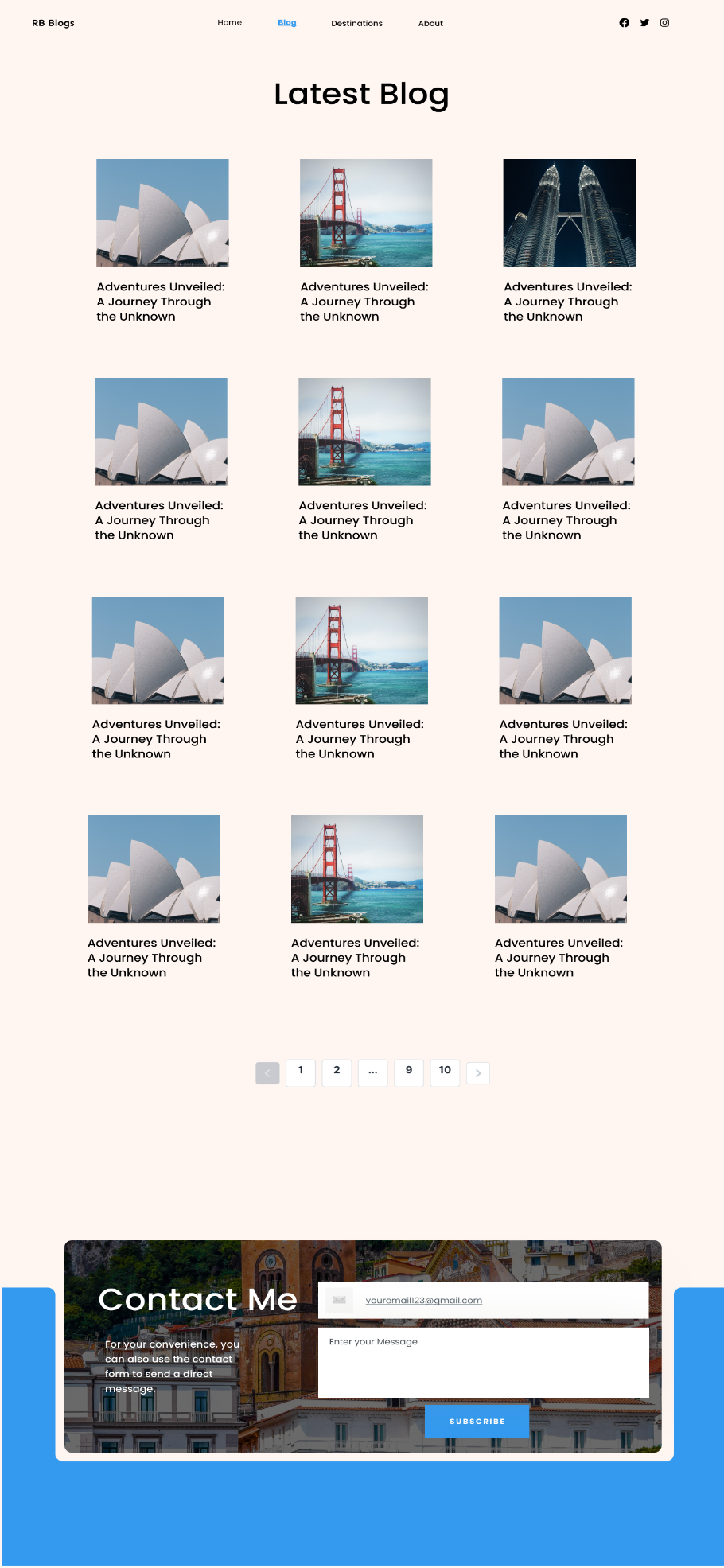
**Login Page :**

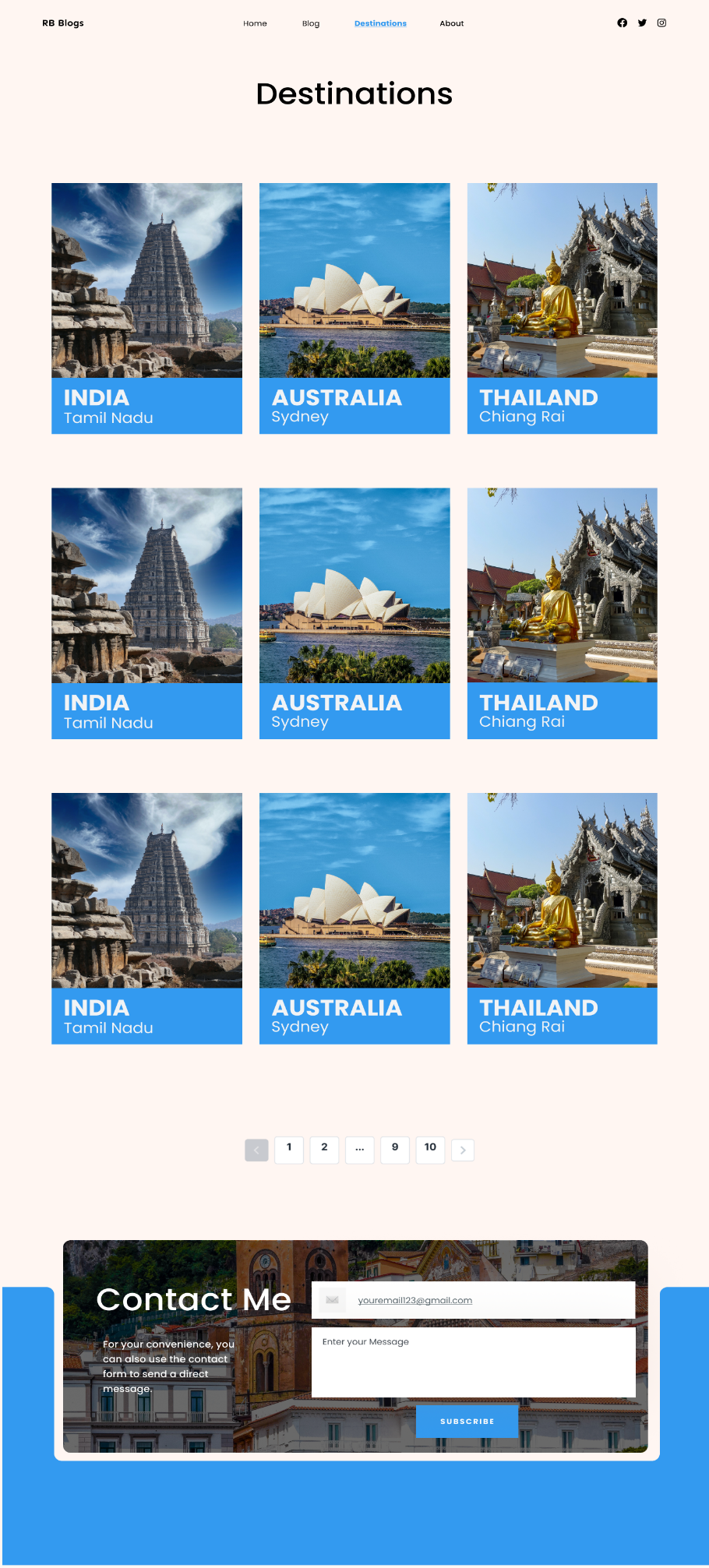


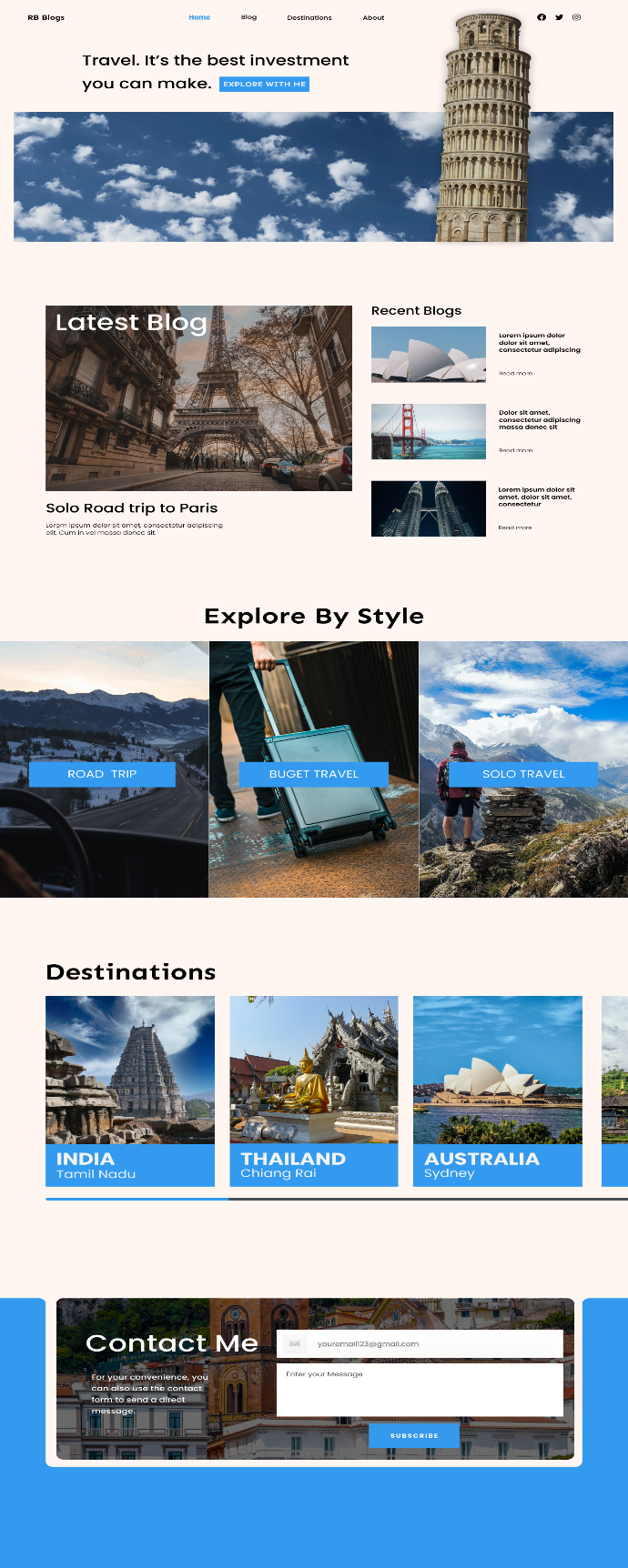
**Registration Page :**



**Home page :**







**Server Code using Flask :**

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

from werkzeug.security import generate\_password\_hash, check\_password\_hash

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

app.secret\_key = 'your\_secret\_key' # Replace with a strong secret key

# Mock user database (you should use a database like SQLite or SQLAlchemy)

users = {

'Barath': {

'username': 'Barath',

'password': generate\_password\_hash('barath007')

}

}

@app.route('/')

def initial():

return render\_template('login.html')

@app.route('/home')

def home():

return render\_template('home.html')

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

def register():

if request.method == 'POST':

username = request.form['username']

password = request.form['password']

if username in users:

flash('Username already exists. Please choose another.')

else:

users[username] = {

'username': username,

'password': generate\_password\_hash(password)

}

flash('Registration successful. Please log in.')

return redirect(url\_for('login'))

return render\_template('register.html')

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

def login():

if request.method == 'POST':

username = request.form['username']

password = request.form['password']

if username in users and check\_password\_hash(users[username]['password'], password):

session['user'] = username

return redirect(url\_for('home'))

else:

flash('Invalid username or password. Please try again.')

return "hello world"

@app.route('/dashboard')

def dashboard():

if 'user' in session:

return f'Hello, {session["user"]}. This is your dashboard.'

return 'You are not logged in.'

@app.route('/logout')

def logout():

session.pop('user', None)

return render\_template('login.html')

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

app.run(port=8000)

**Login.html** :

<!DOCTYPE html>

<html>

<head>

<title>Virtual Cinema - Login</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #f5f5f5;

margin: 0;

padding: 0;

}

.container {

width: 400px;

margin: 150px auto;

background-color: #fff;

border-radius: 5px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

#small {

height: auto;

background-color: #333;

border-radius: 10px 10px 0px 0px;

padding: 10px;

}

.container1 {

padding: 20px;

}

h2 {

color: #fff;

}

form {

text-align: left;

}

label {

display: block;

margin: 10px 0;

}

input[type="text"],

input[type="password"] {

width: 90%;

padding: 10px;

margin: 5px 0;

border: 1px solid #ccc;

border-radius: 4px;

}

input[type="submit"] {

background-color: #007bff;

color: #fff;

padding: 10px 20px;

border: none;

border-radius: 4px;

cursor: pointer;

}

input[type="submit"]:hover {

background-color: #0056b3;

}

</style>

</head>

<body>

<div class="container">

<div id="small">

<h2>Login to Virtual Cinema</h2>

</div>

<div class="container1">

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

<label for="email">Email:</label>

<input type="text" id="email" name="username" required><br><br>

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

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

<input type="submit" value="Login">

<p>Don't have an account? <a href="/register">Create Account</a></p>

</form>

</div>

</div>

<script>

const form = document.querySelector('form');

form.addEventListener('submit', (e) => {

const fs = new FormData(form);

const urlEncoded = new URLSearchParams(fs);

fetch("http://localhost:8000/login", {

method: 'POST',

body: urlEncoded,

headers: {

'Content-Type': 'application/x-www-form-urlencoded'

}

})

});

</script>

</body>

</html>

**Register.html :**

<!DOCTYPE html>

<html>

<head>

<title>Virtual Cinema - Registration</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #f2f2f2;

}

.container {

width: 400px;

margin: 0 auto;

background-color: #ffffff;

border-radius: 5px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

text-align: center;

margin-top: 150px;

border-radius: 10px;

}

#small {

height: auto;

background-color: #555CFF;

border-radius: 10px 10px 0px 0px;

padding: 10px;

}

.container1 {

padding: 20px;

}

h2 {

color: #ffffff;

}

form {

text-align: left;

}

label {

display: block;

margin: 10px 0;

}

input[type="text"], input[type="password"] {

width: 90%;

padding: 10px;

margin: 5px 0;

border: 1px solid #ccc;

border-radius: 4px;

}

input[type="submit"] {

background-color: #007bff;

color: #fff;

padding: 10px 20px;

border: none;

border-radius: 4px;

cursor: pointer;

}

input[type="submit"]:hover {

background-color: #0056b3;

}

.login-link {

margin-top: 20px;

}

</style>

</head>

<body>

<div class="container">

<div id="small">

<h2>Create Account on Virtual Cinema</h2>

</div>

<div class="container1">

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

<label for="email">Email:</label>

<input type="text" id="email" name="username" required><br><br>

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

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

<!-- Additional registration fields go here -->

<input type="submit" value="Register">

</form>

<p class="login-link">Already have an account? <a href="/">Login</a></p>

</div>

<script>

const form = document.querySelector('form');

form.addEventListener('submit', (e) => {

const fs = new FormData(form);

const urlEncoded = new URLSearchParams(fs);

fetch("http://localhost:8000/register", {

method: 'POST',

body: urlEncoded,

headers: {

'Content-Type': 'application/x-www-form-urlencoded'

}

})

});

</script>

</div>

</body>

</html>

**Home.html :**

<!DOCTYPE html>

<html>

<head>

<title>StreamiGo - Media Streaming</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #f5f5f5;

margin: 0;

padding: 0;

}

#header {

background-color: #333;

color: white;

padding: 20px 0;

text-align: center;

}

#navbar {

background-color: #444;

overflow: hidden;

}

#navbar a {

float: left;

display: block;

color: white;

text-align: center;

padding: 10px 20px;

text-decoration: none;

}

#navbar a:hover {

background-color: #666;

}

/\* Flex container for movie cards \*/

#movie-container {

display: flex;

flex-wrap: wrap;

justify-content: space-around;

padding: 20px;

}

/\* Movie card styling \*/

.movie-card {

flex: 0 0 calc(30% - 20px);

border: 1px solid #ddd;

margin: 10px;

padding: 10px;

background-color: #fff;

border-radius: 5px;

box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);

transition: transform 0.2s;

}

.movie-card:hover {

transform: scale(1.02);

}

.movie-poster {

width: 100%;

height: auto;

}

.movie-title {

font-weight: bold;

margin-top: 10px;

}

.movie-description {

margin-top: 10px;

}

/\* Trailer section styling \*/

#trailer-section {

background-color: #111; /\* Dark background color \*/

color: white;

padding: 40px 0;

text-align: center;

}

#trailer-heading {

font-size: 24px;

font-weight: bold;

margin-bottom: 20px;

}

#trailer-video-container {

display: flex;

justify-content: center; /\* Center the iframe horizontally \*/

}

#trailer-video {

width: 80%; /\* Make the video wider \*/

}

.user-dropdown select {

appearance: none;

-webkit-appearance: none;

-moz-appearance: none;

background: transparent;

background-image: url('path-to-your-icon.png'); /\* Replace with your icon URL \*/

background-size: 20px; /\* Adjust icon size as needed \*/

background-repeat: no-repeat;

background-position: right center;

padding: 10px;

}

</style>

</head>

<body>

<div id="header">

<h1>StreamiGo</h1>

</div>

<div id="navbar">

<a href="#">Home</a>

<a href="#">Movies</a>

<a href="#">Trailers</a>

<a href="#">UserProfile</a>

<a href="/logout">Logout</a>

<!-- Add more navigation links as needed -->

</div>

<div id="main-content">

<!-- Trailer Section -->

<div id="trailer-section">

<h2 id="trailer-heading">Upcoming Movie Trailers</h2>

<div id="trailer-video-container">

<iframe id="trailer-video" width="800" height="450" src="https://www.youtube.com/embed/Po3jStA673E?si=3JxCQE36C\_T2lz2Y" title="YouTube video player" frameborder="0" allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture; web-share" allowfullscreen></iframe>

</div>

</div>

<!-- Flex container for movie cards -->

<div id="movie-container">

<!-- Recommended Movies -->

<!-- Movie cards go here -->

<div class="movie-card">

<img class="movie-poster" src="/static/leo.jpeg" alt="Movie 1 Poster">

<div class="movie-details">

<h2 class="movie-title">Movie Title 1</h2>

<p class="movie-description">A thrilling action film with a compelling story.</p>

</div>

</div>

<div class="movie-card">

<img class="movie-poster" src="/static/leo.jpeg" alt="Movie 2 Poster">

<div class="movie-details">

<h2 class="movie-title">Movie Title 2</h2>

<p class="movie-description">An emotional drama that will leave you in tears.</p>

</div>

</div>

<!-- Add more movie cards for other recommended movies -->

<!-- Add more movie cards for other recommended movies -->

<div class="movie-card">

<img class="movie-poster" src="/static/leo.jpeg" alt="Movie 3 Poster">

<div class="movie-details">

<h2 class="movie-title">Movie Title 3</h2>

<p class="movie-description">A captivating mystery that will keep you guessing.</p>

</div>

</div>

<div class="movie-card">

<img class="movie-poster" src="/static/leo.jpeg" alt="Movie 4 Poster">

<div class="movie-details">

<h2 class="movie-title">Movie Title 4</h2>

<p class="movie-description">A heartwarming family film for all ages.</p>

</div>

</div>

<div class="movie-card">

<img class="movie-poster" src="/static/leo.jpeg" alt="Movie 5 Poster">

<div class="movie-details">

<h2 class="movie-title">Movie Title 5</h2>

<p class="movie-description">A mind-bending sci-fi adventure that will leave you questioning reality.</p>

</div>

</div>

<div class="movie-card">

<img class="movie-poster" src="/static/leo.jpeg" alt="Movie 6 Poster">

<div class="movie-details">

<h2 class="movie-title">Movie Title 6</h2>

<p class="movie-description">An epic fantasy film with stunning visuals and an unforgettable journey.</p>

</div>

</div>

</div>

<script>

// JavaScript to handle the dropdown change

document.getElementById("user-dropdown").addEventListener("change", function() {

var selectedValue = this.value;

if (selectedValue === "logout") {

window.location.href = "/logout"; // Redirect to the logout page

} else if (selectedValue === "user") {

// Handle user profile link

// You can add your logic here

}

});

</script>

</div>

</body>

</html>