# Project Based Learning Activity

On

“Travel Mate – a travel buddy finder website”

A Report Submitted for a **Project Based Learning** in 2nd Semester of Second Year Computer Engineering

Academic Year 2023-24 Submitted by-

|  |  |  |
| --- | --- | --- |
| **Sr.No.** | **NAME** | **ROLL NO.** |
| 1. | Arhaan Shaikh | S213027 |
| 2. | Imam Shaikh | S213028 |
| 3. | Sakshi Shelar | S213029 |
| 4. | Sairaj Shilimkar | S213032 |
| 5. | Om Shinde | S213033 |



Zeal Education Society’s

Zeal College of Engineering & Research Narhe, Pune-411041.

## Zeal Education Society’s

**Zeal College of Engineering & Research Department of Computer Engineering**



## CERTIFICATE

This is to certify that Project **Travel Mate – a travel buddy finder**

|  |  |  |
| --- | --- | --- |
| **Sr.NO.** | **NAME** | **ROLL NO.** |
| 1. | Arhaan Shaikh | S213027 |
| 2. | Imam Shaikh | S213028 |
| 3. | Sakshi Shelar | S213029 |
| 4. | Sairaj Shilimkar | S213032 |
| 5. | Om Shinde | S213033 |

are Bonafide students of this institute and the work has been carried out by them under the supervision of **Prof. Komal Bhosale** and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of Second Year Engineering (Computer Engineering). It is certified that all corrections/suggestions indicated for internal assignment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor of Engineering Degree.

## Prof. Komal Bhosale Prof. A. V. Mote

Project Guide H. O. D

# ACKNOWLEDGEMENT

We take this opportunity to thank our project guide **Prof. Komal Bhosale** and Head of Department **Prof. A. V. Mote** for their valuable guidance and for providing all the necessary facilities, which were indispensable in the completion of this project report. We are also thankful to all the staff members of the Computer Engineering Department for their valuable time, support, comments, suggestions and persuasion. We would also like to thank the institute for providing the required facilities, Internet access and important books.

# INDEX

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **CONTENT** | **Page No.** |
| 1. | Abstract | 05 |
| 2. | Software Requirement | 06 |
| 3. | Introduction | 07 |
| 4. | Problem Statement | 08 |
| 5. | Objective and Outcome | 08 |
| 6. | Theory | 08- |
| 7. | Input & Output | 11-26 |
| 8. | Conclusion | 27 |
| 9. | References | 28 |

**ABSTRACT**

Travel Mate is a dynamic platform that revolutionizes the way people find travel companions for their adventures around the globe. By leveraging social networking principles, Travel Mate brings together individuals with similar interests and travel plans, facilitating connections and fostering friendships beyond geographical boundaries. Through advanced search algorithms and interactive features like liking and commenting, users can discover potential travel buddies, exchange ideas, and plan memorable trips together.

Whether exploring remote destinations or embarking on urban escapades, Travel Mate enhances the travel experience by creating a vibrant community of like-minded explorers. With its intuitive interface and robust functionalities, Travel Mate is poised to become the go-to platform for connecting travelers and igniting new adventures worldwide.

# SOFTWARE REQUIREMENT

* *Operating System: Windows, Linux, macOS*
* *Programming languages: HTML, CSS, JavaScript, Mern Stack*
* *Database Management System: MongoDB*
* *Software: VS Code, firebase, GitHub*
* *System: Windows 11*

# INTRODUCTION

Welcome to Travel Mate, the ultimate hub for adventurers and wanderers alike. Our platform is more than just a website; it's a thriving community of passionate travelers eager to connect, explore, and share their adventures.

At Travel Mate, we understand that the journey is as important as the destination, and that's why we've curated a range of features to enhance every aspect of your travel experience. From creating personalized profiles to engaging in lively discussions, planning group excursions to discovering hidden gems, Travel Mate is your go-to destination for all things travel-related.

Travel Mate is where travelers come together to forge meaningful connections and plan unforgettable trips. Our platform offers a diverse array of features designed to cater to every type of traveler. Whether you're a solo adventurer seeking companionship or a seasoned globetrotter looking to expand your network, Travel Mate provides the tools and resources you need to make your travel dreams a reality. With a user-friendly interface and an active community of like-minded explorers, Travel Mate is the perfect companion for your next journey.

From creating personalized profiles to engaging in lively discussions, planning group excursions to discovering hidden gems, Travel Mate offers a range of features to enhance your travel experience. Explore destinations, connect with fellow travellers, and plan your next adventure with ease.

Travel Mate stands out as the premier platform for travel enthusiasts thanks to its community-driven approach, interactive features, and commitment to safety and security. Join Travel Mate today and unlock a world of possibilities as you embark on your next adventure.

# PROBLEM STATEMENT:

Many travelers face the challenge of finding compatible travel companions and planning trips efficiently, often resulting in missed opportunities for meaningful connections and suboptimal travel experiences. Travel Mate addresses this issue by providing a comprehensive platform where travelers can connect with like-minded individuals, plan trips collaboratively, and share their experiences.

# OBJECTIVE:

The main objectives of the tourism blog website are:

* + Foster connections between travelers based on shared interests and destinations.
  + Provide tools for collaborative trip planning, making it easier to organize and coordinate travel.
  + Promote active participation through liking, sharing, and commenting features.
  + Prioritize user experience and security to enhance overall satisfaction.
  + Stay updated with industry trends and improve the platform to meet evolving needs.

# OUTCOME:

Travel Mate facilitates meaningful connections among travelers, enhancing their journeys and fostering a sense of community.

With collaborative tools and engaged users, Travel Mate streamlines trip planning processes, resulting in more enjoyable and well-coordinated travel experiences.

# THEORY:

The MERN stack, comprised of MongoDB, Express.js, React.js, and Node.js, is an ideal choice for developing the Travel Mate platform due to its versatility, scalability, and efficiency. Let's delve into each component and its role in creating a robust and dynamic travel community platform:

# MongoDB(Database)

* MongoDB, a NoSQL database, offers flexibility and scalability, making it suitable for handling diverse data types and accommodating the growing user base of Travel Mate.
* It stores user profiles, travel preferences, trip itineraries, and interaction data, allowing for efficient retrieval and manipulation of information.
* MongoDB's document-based structure aligns well with the variable nature of travel data, enabling seamless integration with the platform's features.

# Express.js(Backend Framework)

* Express.js, a lightweight Node.js framework, serves as the backend of the Travel Mate platform, handling server-side logic, routing, and API interactions.
* It facilitates the creation of RESTful APIs for user authentication, profile management, trip planning, and interaction functionalities.
* Express.js seamlessly integrates with MongoDB, enabling smooth data retrieval, manipulation, and storage operations.

# React.js(Frontend Library)

* React.js, a powerful JavaScript library, powers the frontend of Travel Mate, providing a dynamic and interactive user interface.
* It enables the creation of reusable UI components, enhancing development efficiency and maintainability.
* React.js facilitates real-time updates and seamless navigation, offering users a smooth and engaging experience while interacting with the platform's features.

# Node.js(Runtime Environment)

* Node.js serves as the runtime environment for Travel Mate, enabling server-side JavaScript execution and event-driven architecture.
* It provides high scalability and performance, allowing Travel Mate to handle concurrent user requests and deliver responsive user experiences.
* Node.js integrates seamlessly with Express.js and MongoDB, forming a cohesive backend ecosystem for the platform.

# Advantages of MERN Stack for Travel Mate

* **Unified Development Environment:** The MERN stack offers a unified development environment, allowing developers to use JavaScript throughout the entire stack, streamlining development and reducing code complexity.
* **Scalability and Performance:** Leveraging Node.js and MongoDB's scalability features, Travel Mate can easily scale to accommodate growing user demands and deliver high-performance experiences.
* **Real-time Updates:** With React.js and Node.js, Travel Mate can implement real-time updates and notifications, enhancing user engagement and fostering dynamic interactions within the travel community.
* **Community Support and Resources:** The MERN stack boasts a large and active developer community, providing ample resources, libraries, and frameworks to expedite development and troubleshoot issues effectively.

## Implementing a Website:

1. Define the Purpose and Goals: Determine the purpose of your website and identify the goals you want to achieve. This will guide the content, design, and functionality of your site.
2. Plan the Structure and Content: Create a site map that outlines the main sections and pages of your website. Determine the content you want to include on each page, such as text, images, videos, and forms.
3. Design the User Interface: Create a visual design for your website, considering factors like branding, color scheme, typography, and overall layout. Use design tools or hire a web designer to create a visually appealing and user-friendly interface.
4. Develop the Front-End: Write the HTML, CSS, and JavaScript code to implement the user interface and interactivity of your website. Structure your HTML according to the site map, apply CSS styles for visual presentation, and use JavaScript for dynamic functionality.
5. Implement Back-End Functionality: If your website requires server-side processing, choose a back-end technology such as PHP, Node.js, or Python. Develop the necessary code to handle data storage, user authentication, form submissions, database interactions, and other server-side functionalities.
6. Create and Optimize Content: Write compelling and relevant content for each page, considering search engine optimization (SEO) best practices. Optimize images and other media files for faster loading times.
7. Test and Debug: Thoroughly test your website across different browsers, devices, and screen sizes to ensure compatibility and responsiveness. Debug any issues or errors that may arise during testing.
8. Deploy and Host: Choose a reliable web hosting provider and deploy your website to a live server. Configure domain settings, set up SSL certificates for secure connections if necessary, and ensure proper backup and security measures are in place.
9. Monitor and Maintain: Regularly monitor your website's performance, security, and user feedback. Make updates and improvements as needed to keep your site up-to- date and user-friendly.

# IMPLEMENTATION CODE

## File Name – app.js

import logo from './logo.svg';

import React,{createContext ,useState} from 'react';

import './App.css';

import Navbar from './components/Navbar';

import { BrowserRouter,Routes,Route, unstable\_useViewTransitionState } from 'react-router-dom';

import Home from './components/Home';

import SignUp from './components/SignUp';

import Signin from './components/Signin';

import Profile from './components/Profile';

import { ToastContainer} from 'react-toastify';

import 'react-toastify/dist/ReactToastify.css';

import Makeatrip from './components/Makeatrip';

import Modal from './components/modal';

import { Logincontext } from './context/logincontext';

import { useContext } from 'react';

import UserProfile from './components/UserProfile';

import MyFolliwngPost from './components/MyFollowingPost';

import Travelmate from './components/mainhome';

import Start from './home/Start'

function App() {

const [userLogin, setUserLogin]=useState(false);

return (

<BrowserRouter>

<div className="App">

<Logincontext.Provider value={{setUserLogin}} >

<Navbar login={userLogin} />

<Routes>

{/<Route path='/travelmate' element={<Start/>}></Route>/}

<Route path='/' element={<Home/>}></Route>

<Route path='/signup' element={<SignUp/>}></Route>

<Route path='/signin' element={<Signin/>}></Route>

<Route exact path='/profile' element={<Profile/>}></Route>

<Route path='/makeatrip' element={<Makeatrip/>}></Route>

<Route path='/profile/:userid' element={<UserProfile/>}></Route>

<Route path='/followingpost' element={<MyFolliwngPost/>}></Route>

</Routes>

<ToastContainer theme='colored'/>

</Logincontext.Provider>

</div>

</BrowserRouter>

);

}

export default App;

# File name: index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<meta name="theme-color" content="#000000" />

<meta

name="description"

content="Web site created using create-react-app"

/>

<link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />

<link rel="stylesheet" href="https://fonts.googleapis.com/css2?family=Material+Symbols+Outlined:opsz,wght,FILL,GRAD@20..48,100..700,0..1,-50..200" />

<!--

manifest.json provides metadata used when your web app is installed on a

user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/

-->

<link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />

<!--

Notice the use of %PUBLIC\_URL% in the tags above.

It will be replaced with the URL of the public folder during the build.

Only files inside the public folder can be referenced from the HTML.

Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC\_URL%/favicon.ico" will

work correctly both with client-side routing and a non-root public URL.

Learn how to configure a non-root public URL by running npm run build.

-->

<title>Travelmate</title>

</head>

<body>

<noscript>You need to enable JavaScript to run this app.</noscript>

<div id="root"></div>

<!--

This HTML file is a template.

If you open it directly in the browser, you will see an empty page.

You can add webfonts, meta tags, or analytics to this file.

The build step will place the bundled scripts into the <body> tag.

To begin the development, run npm start or yarn start.

To create a production bundle, use npm run build or yarn build.

-->

</body>

</html>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8" />

<link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />

<meta name="viewport" content="width=device-width, initial-scale=1" />

<meta name="theme-color" content="#000000" />

<meta

name="description"

content="Web site created using create-react-app"

/>

<link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />

<link rel="stylesheet" href="https://fonts.googleapis.com/css2?family=Material+Symbols+Outlined:opsz,wght,FILL,GRAD@20..48,100..700,0..1,-50..200" />

<!--

manifest.json provides metadata used when your web app is installed on a

user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/

-->

<link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />

<!--

Notice the use of %PUBLIC\_URL% in the tags above.

It will be replaced with the URL of the public folder during the build.

Only files inside the public folder can be referenced from the HTML.

Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC\_URL%/favicon.ico" will

work correctly both with client-side routing and a non-root public URL.

Learn how to configure a non-root public URL by running npm run build.

-->

<title>Travelmate</title>

</head>

<body>

<noscript>You need to enable JavaScript to run this app.</noscript>

<div id="root"></div>

<!--

This HTML file is a template.

If you open it directly in the browser, you will see an empty page.

You can add webfonts, meta tags, or analytics to this file.

The build step will place the bundled scripts into the <body> tag.

To begin the development, run npm start or yarn start.

To create a production bundle, use npm run build or yarn build.

-->

</body>

</html>

# File Name: index.css

body {

margin: 0;

font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',

'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',

sans-serif;

-webkit-font-smoothing: antialiased;

-moz-osx-font-smoothing: grayscale;

}

code {

font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',

    monospace;

}

# File Name: navbar.js

import React ,{useContext} from "react";

import "./Navbar.css";

import { Link } from "react-router-dom";

import logo from "../image/logo.jpeg"

import { Logincontext } from "../context/logincontext";

import { useNavigate } from "react-router-dom";

import { toast } from 'react-toastify';

export default function Navbar({login}){

const navigate=useNavigate()

const notifyA=(msg)=>toast.success(msg)

const loginstatus=()=>{

const token =localStorage.getItem("jwt")

if(token || login){

return[

<>

<Link to="/"><li className="nav">Book a Trip</li></Link>

<Link to="/followingpost"><li className="nav">Buddies' Trips</li></Link>

<Link to="/makeatrip"><li className="nav">Make a Trip</li></Link>

<Link to="/profile"><li className="nav">Profile</li></Link>

<Link to={""}>

<button className="primaryBtn" onClick={()=>{

if(window.confirm("Do you really want to log out?")){

notifyA("Logged out successfully");

navigate("https://travelmate-a4e99.web.app");

localStorage.clear();

}

}}>Log Out</button>

</Link>

</>

]

}

else{

return[

<>

<Link to="https://travelmate-a4e99.web.app"><li className="nav">HOME</li></Link>

<Link to="/signup"><li className="nav">SignUp</li></Link>

<Link to="/signin"><li className="nav">Signin</li></Link>

</>

]

}

}

return(

<div className="navbar">

<img src={logo} alt="travelmate" />

<ul className="nav-menu">

{loginstatus()}

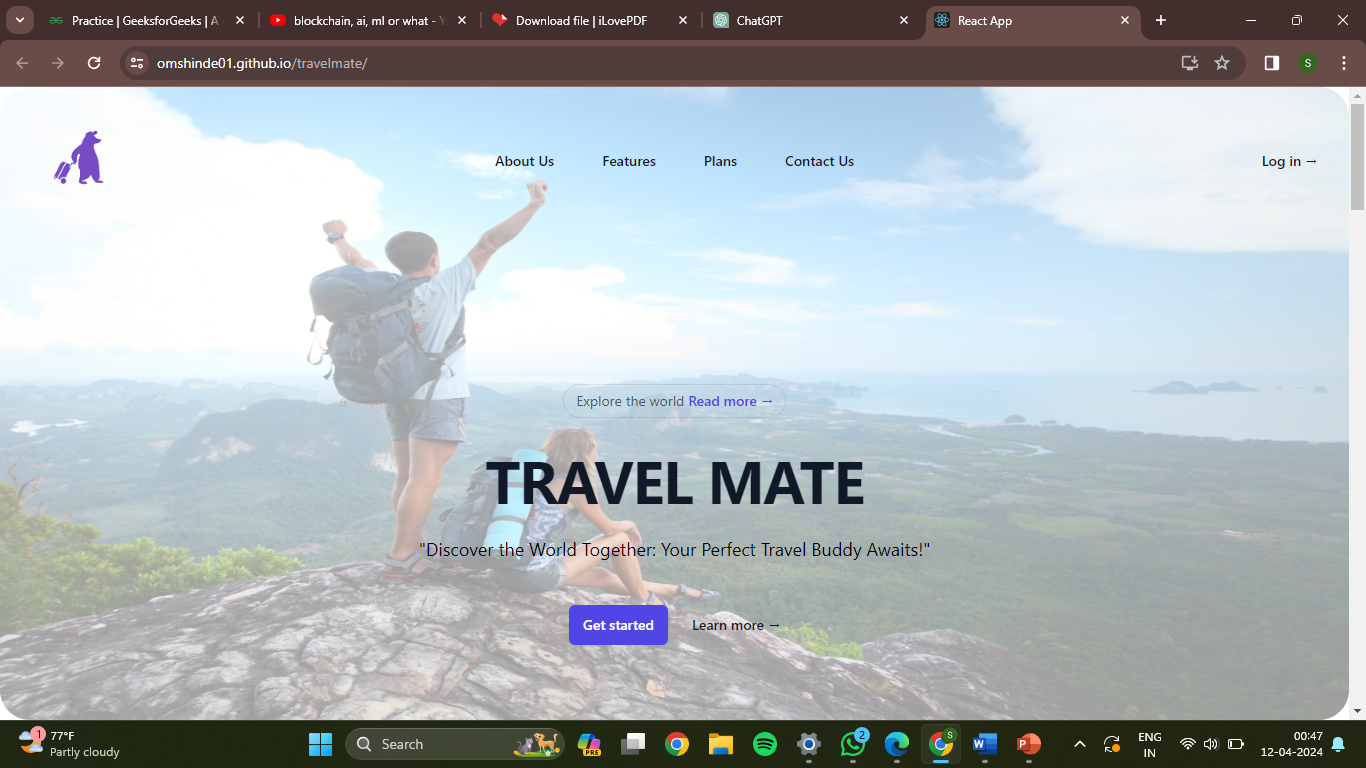
</ul>

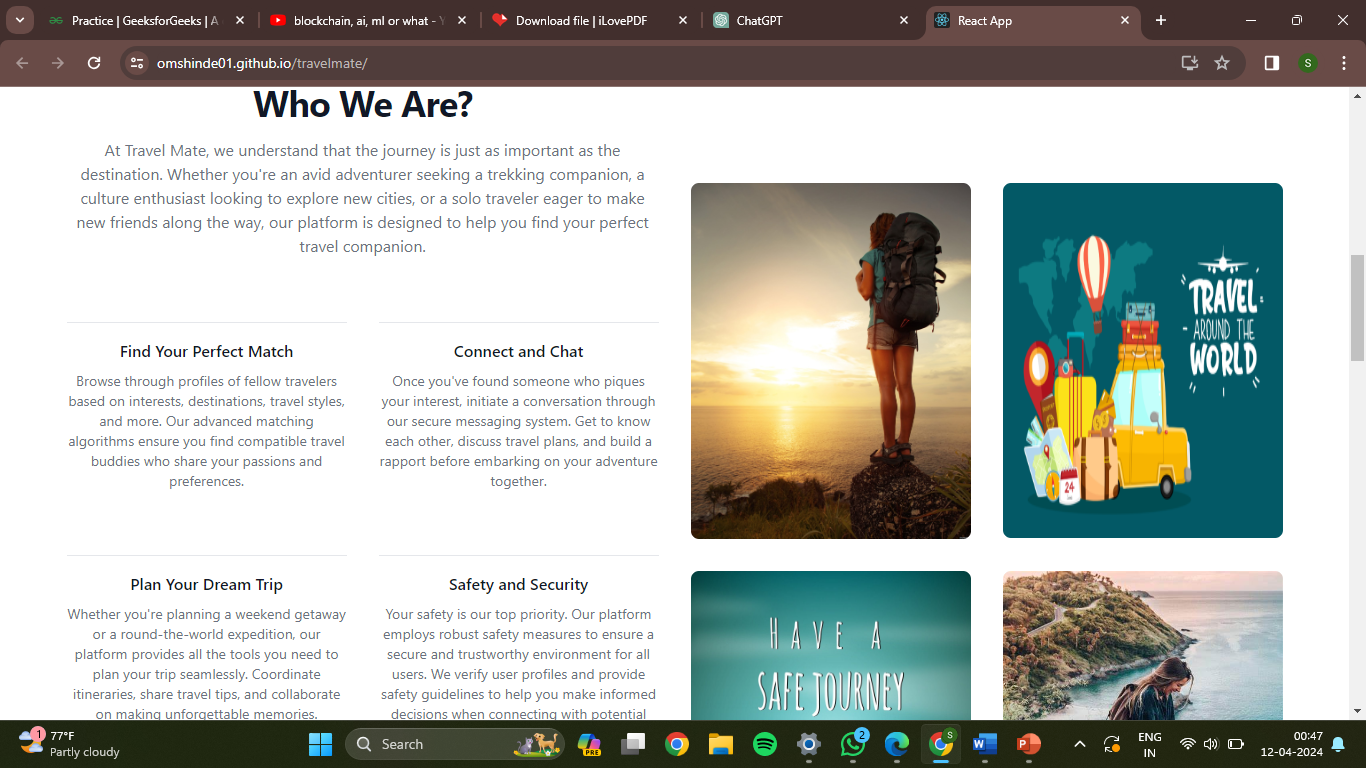
</div>

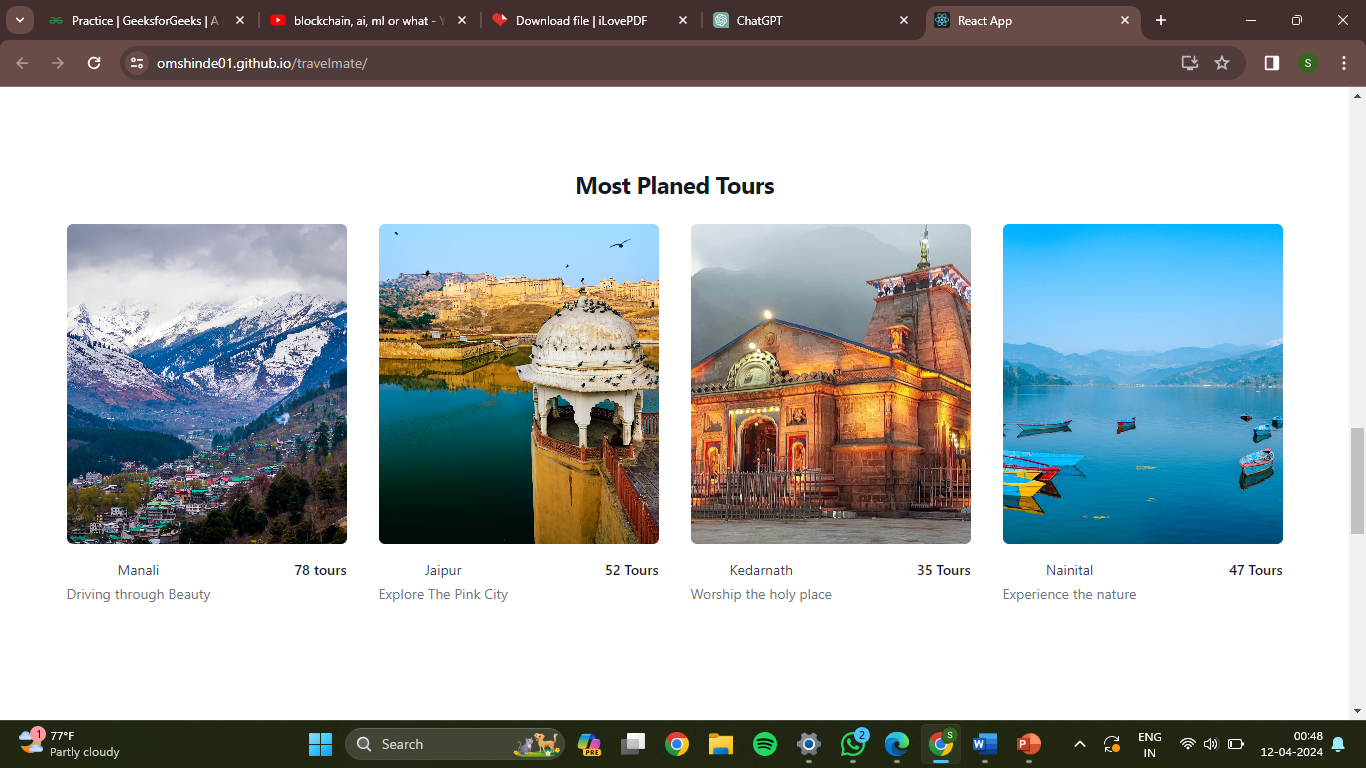
)

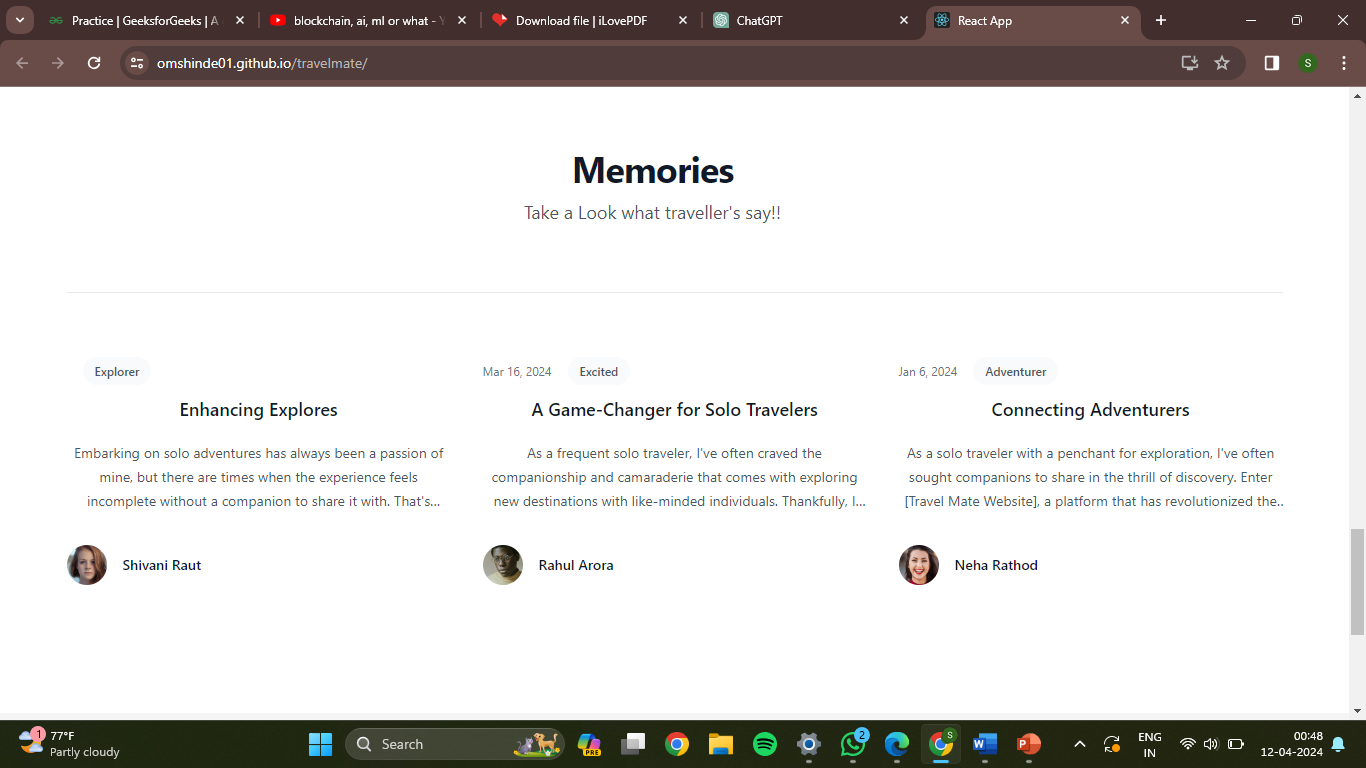
}

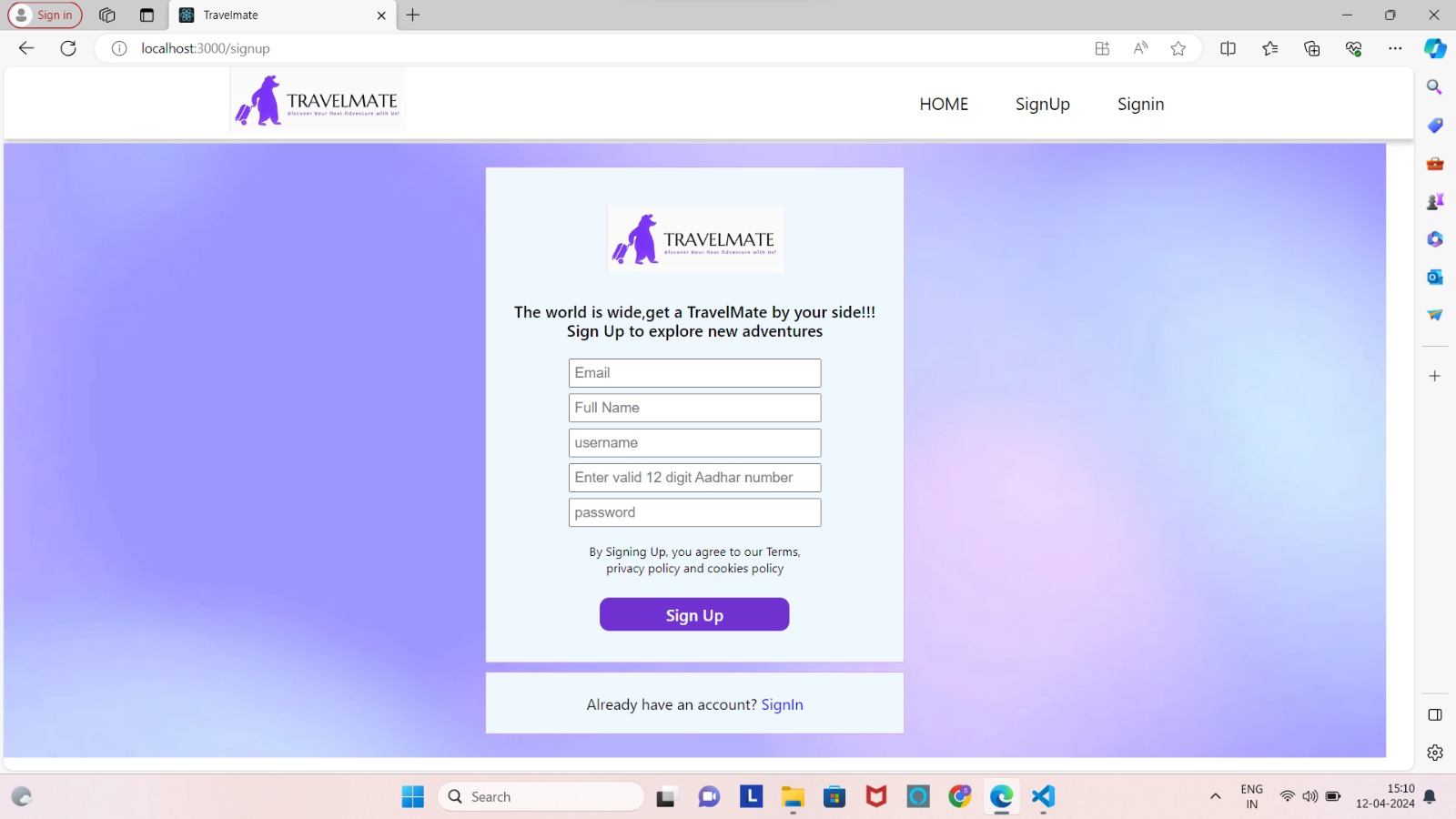
# OUTPUT:

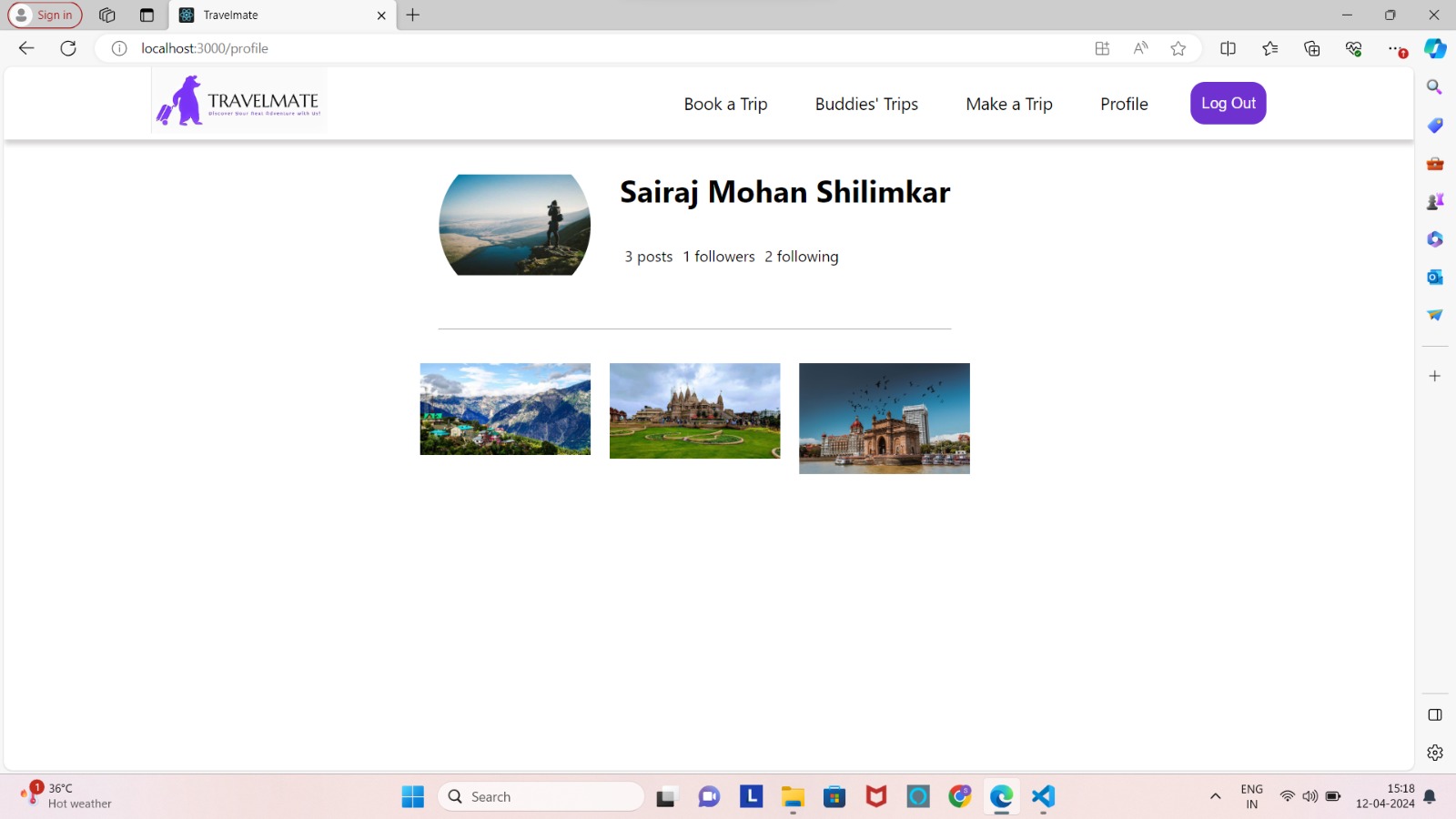
****

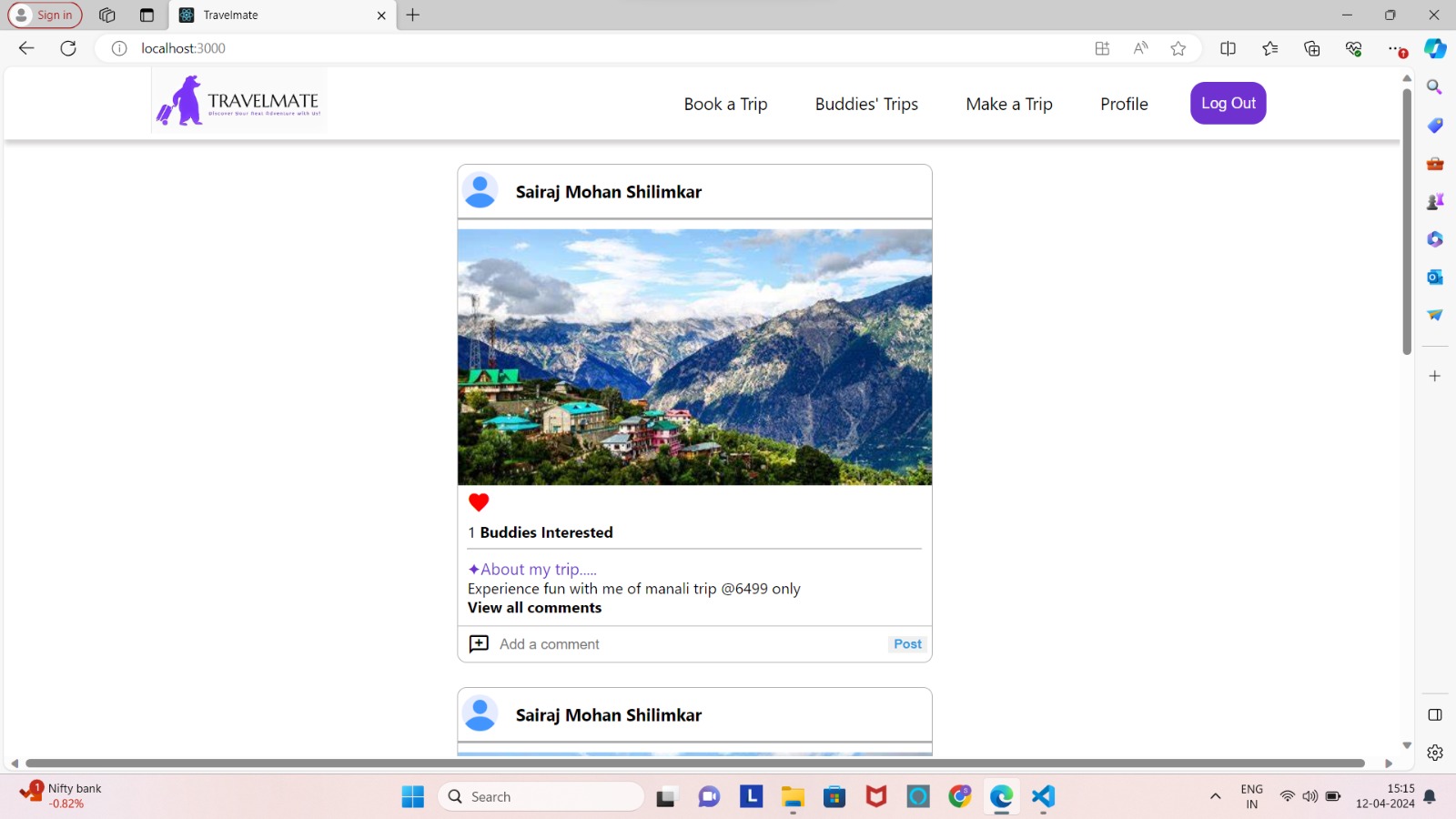
****



****

****

****

****

**CONCLUSION**

* **Empowering Travelers:** Travel Mate empowers travelers by providing a platform where they can connect, collaborate, and explore the world together, enriching their experiences and creating lifelong memories.
* **Building Bridges:** Through its emphasis on community-building and collaboration, Travel Mate transcends geographical boundaries, fostering cross-cultural understanding and promoting global citizenship.
* **Continual Evolution:** As Travel Mate continues to evolve and innovate, it remains committed to its mission of connecting travelers worldwide and making every journey an adventure to remember.

# REFERENCES

* <https://www.researchgate.net/publication/285682764_Social_networks_and_travel_Some_hypotheses>
* <https://www.researchgate.net/publication/45158767_Trust_perceptions_of_online_travel_information_by_different_content_creators_Some_social_and_legal_implications>
* <https://www.academia.edu/14459851/Safety_and_security>
* <https://ieeexplore.ieee.org/document/10467991/>
* <https://ieeexplore.ieee.org/document/10439090/>