MIT-WPU

॥ विश्वशान्तिर्ध्रुवं ध्रुवा ॥

FULL STACK DEVELOPMENT

MINI-PROJECT

Rental Application

Project Members

Amar Khakhkhar PE-57

Adeeb Khan PE-54

Anshjeet Mahir PE-55

Pratham Acharya PE-56

Under the Guidance of

Prof. Dipali Baviskar

School of Computer Engineering and Technology
MIT World Peace University, Kothrud,
Pune 411 038, Maharashtra - India
2023-2024

Overview of technologies used

1. MERN Stack:

1.1 JavaScript :

The object-oriented, cross-platform programming language is called JavaScript. JavaScript can be used to establish methods for controlling objects in the host environment.

Standard object libraries like Date, Math, and Array are included in JavaScript. It also includes the basic building elements of programming languages, like statements, control structures, and instance managers.

By adding objects, JavaScript could be expanded to include a wide range of ideas, such as:

• Client-side JavaScript: Objects that control the DOM and browser are implemented to produce this kind of JavaScript. Client-side extensions, for instance, let an application update elements on an HTML page and react to user interaction such as form submissions, mouse hovers, and page changes.

• JavaScript generated server-side entails putting in place the additional objects required for server-side JavaScript operation. For example, this server-side extension enables an application to connect to a database, run with a distinct function file on the server, and transfer data regularly between requests inside the application.

The DOM controls how your scripts display HTML elements. Check out the JavaScript technology analysis page for a preview of the special innovations used in JavaScript programming.

1.2 NodeJS :

An open-source system programme that acts as a server environment is called Node.js. We can easily and quickly create network apps with Nodejs, an independent development platform built on top of Chrome's JavaScript Runtime. The Google V8 JavaScript engine powers Node.js applications. Furthermore, a large number of required modules are written in JavaScript 6.

Apps can operate as a web server without the need for extra software like Nginx, IIS, or Apache HTTP Server because Node.js includes an integrated library.
Event-driven, non-blocking input and output techniques are provided by Node.js. It optimises the programme overall and is very extensible. Node.js uses asynchronous programming in its operations. This means that Node.js handles all task execution and processing in the background.

Applications using NodeJS:

- WebSocket server
- Notification system
- Applications that need to upload files on the client.
- Other real-time data applications.

1.3 Express.js :

Express.js is a framework built on top of Node.js. It provides robust features for online and mobile application development.

The HTTP protocols and middleware are supported by Express.js, which makes the API very versatile and easy to use.

Express enhances NodeJS's capability to provide developers with a more favourable development environment rather than decreasing its speed.

Significantly, Express.js is a fundamental component of well-known NodeJS frameworks like Sails.js and MEAN.

1.4 MongoDB :

The most widely used NoSQL (*) database, MongoDB, is currently used by millions of users and is open source. It is written in one of the most popular programming languages accessible today. In addition, MongoDB is a cross-platform data system that offers high availability, high performance, and ease of extension by relying on the concepts of collections and documents.

(*) Transact-SQL is not used by NoSQL, a source database format that was developed with the JavaScript Framework and the JSON data type.

The shortcomings of the RDBMS relational data model have been fixed since its introduction to improve caching, model scalability, operation speed, and other factors.

Additionally, MongoDB is a cross-platform database that offers quick production, high availability, and straightforward scaling. It runs on both Collection and Document methods.

Commonly used terms in MongoDB:

- _id
- Collection

- Cursor
- Database
- Document
- Field
- JSON
- Index

1.5 ReactJS:

1.5.1 Virtual-DOM

A JavaScript object called Virtual-DOM has all the information needed to create a DOM. When data changes, it calculates the difference between the object and the actual tree, which aids in optimising the DOM tree's re-rendering. It goes without saying that a virtual model can handle customer data.

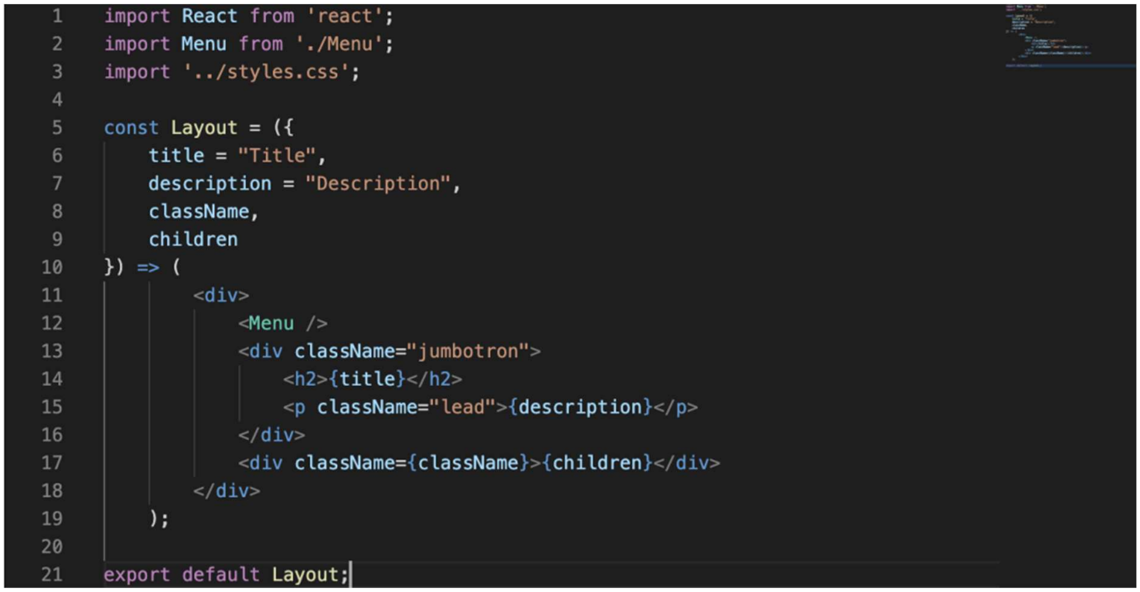
1.5.2 Component

React is not built around templates like other frameworks; instead, it is centred around components. You can build a component by using the create Class function of the React object, which is where you start when you visit this library.

ReactJS creates HTML tags in a different way than we usually write them, but it renders them as layered objects by enclosing them in Components.
The most important part of a React application is the render function. It is a feature that controls the generation of HTML tags and demonstrates how to use Virtual-DOM for data processing. Any time there is a data change, Virtual-DOM will instantaneously process and update it.

1.5.3 Props

Props: are not controlled by Component, actually stands for Properties



```
import React from 'react';
import Menu from './Menu';
import '../styles.css';

const Layout = ({
    title = "Title",
    description = "Description",
    className,
    children
}) => (
        <div>
            <Menu />
            <div className="jumbotron">
                <h2>{title}</h2>
                <p className="lead">{description}</p>
            </div>
            <div className={className}>{children}</div>
        </div>
    );

export default Layout;
```

The title = "Title" line creates a name attribute with the value "Title". It appears to be a function call. It is correct that props are sent to the component just like an argument is sent to a function.

Furthermore, a component can have default props set, meaning that it will continue to function even if no props are provided.

1.6 Redux:

Redux is an open-source JavaScript tool that is used to consistently manage the state of an application. While it is commonly employed alongside React, it can also be used separately in JavaScript applications or in combination with other view libraries or frameworks.

The core concepts of Redux are based on the Flux design, which emphasises a one-way data flow. An application's state is stored by Redux in a single JavaScript object known as the "store." The store's state can only be changed by dispatching operations; otherwise, it is read-only. To express state changes, simple JavaScript objects called actions are utilised. Reducers are pure functions that provide the state that an operation should modify.

1.6.1 Store

The store is the hub of a Redux application. All the state of your application is kept in a single JavaScript object. This state represents a specific instance of your application at that given time. The following are the store's three main responsibilities:

Holding State: The store keeps track of the present state of your application.

Offering a way for people to contact the state: You may acquire the current state by using the getState function.

Allowing the state to be updated: To update the state, send actions to the store.

1.6.2 Reducers:

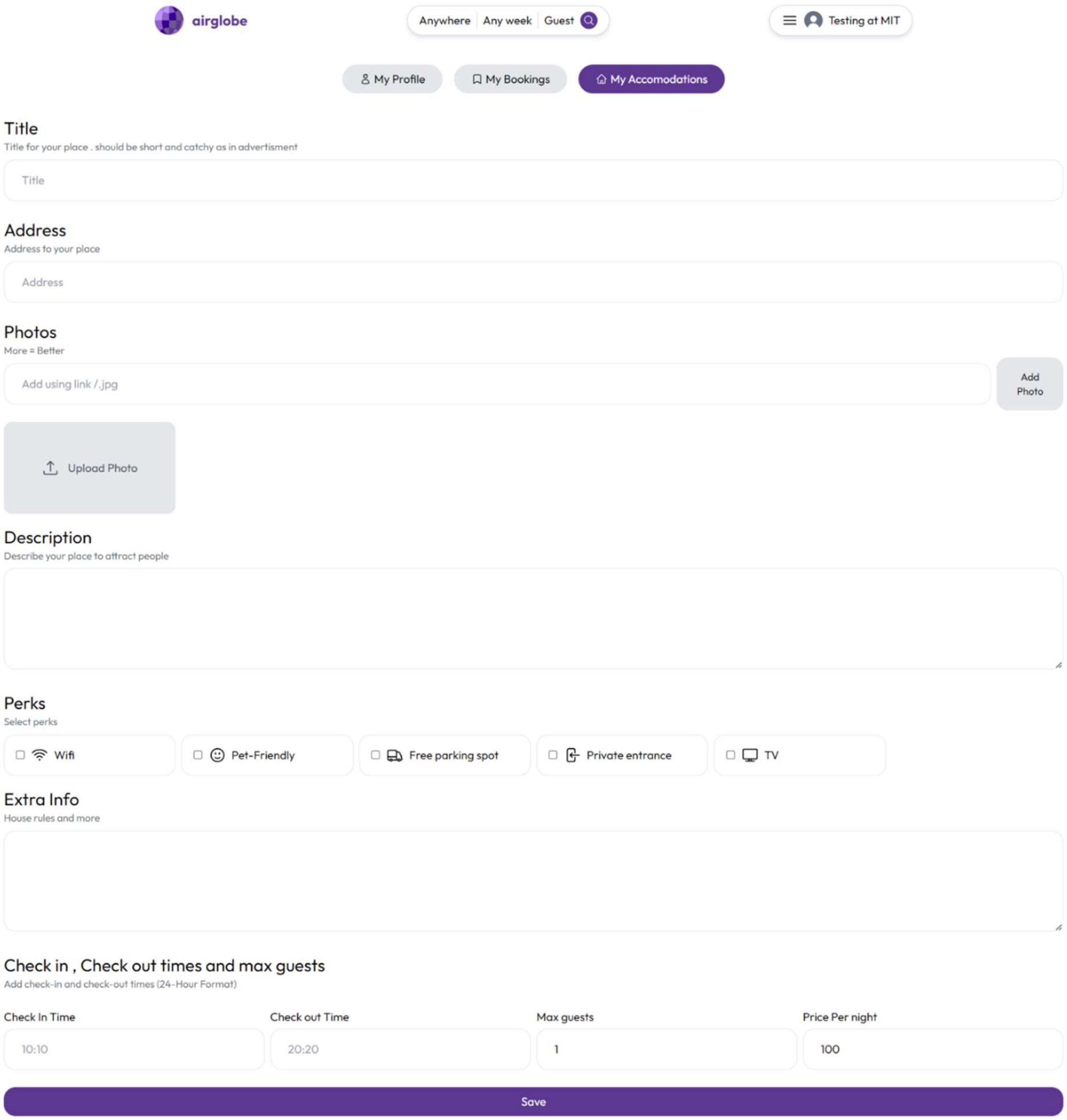
Functions called reducers are in charge of defining how the state of the application changes in reaction to user input. They return the next state after accepting the argument of the current state and an action. Reducers have to be pure functions, which means they have to return a new state object rather than alter the state directly.

1.6.3 Actions:

Information payloads from your application are sent to your store as actions. For the store, they are the only source of information. Simple JavaScript objects with a type property designating the kind of action being carried out are called actions.

2. Workflow/ architecture diagram with explanation

i. Rent place adding page



Title

Title for your place . should be short and catchy as in advertisment

Title

Address

Address to your place

Address

Photos

More = Better

Add using link /.jpg

Description

Describe your place to attract people

Perks

Select perks

□ Wifi □ Pet-Friendly □ Free parking spot □ Private entrance □ TV

Extra Info

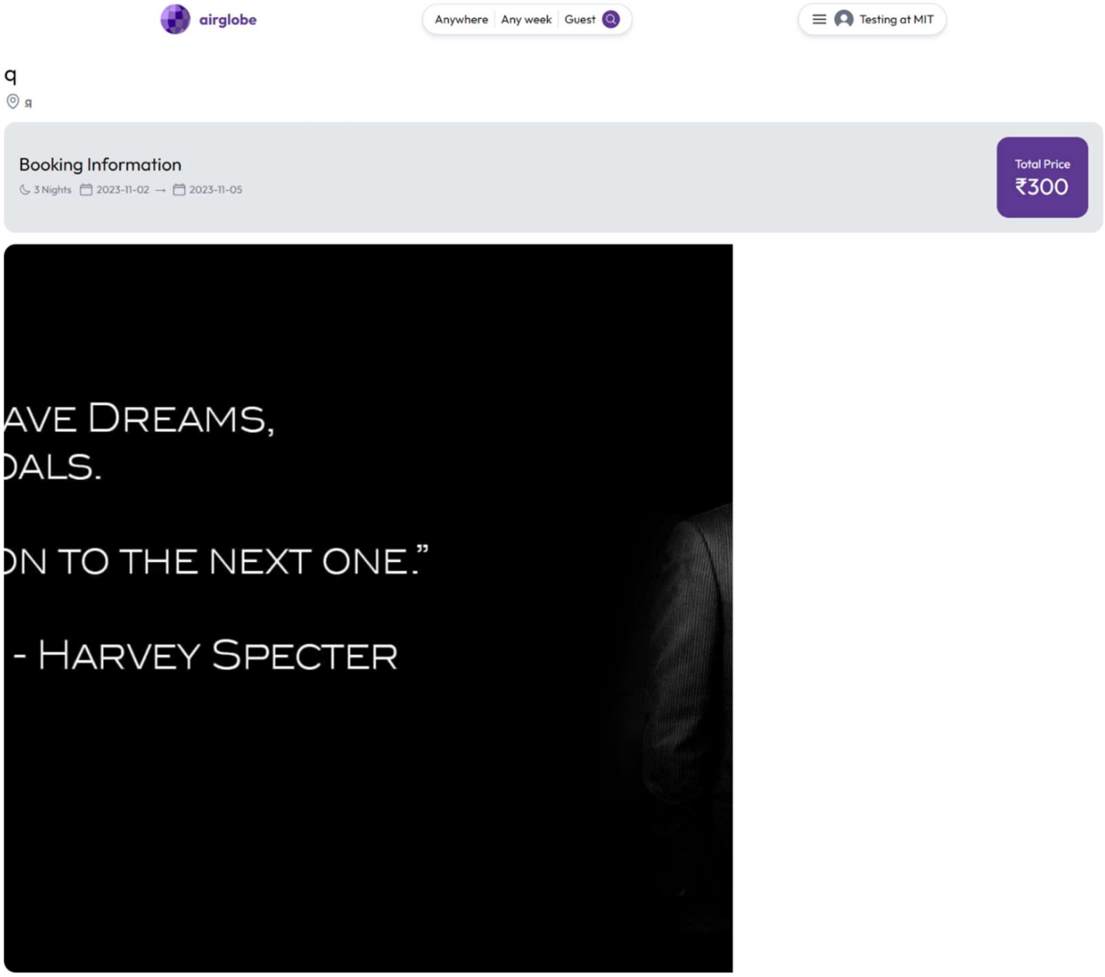
House rules and more

Check in , Check out times and max guests

Add check-in and check-out times (24-Hour Format)

Check In Time	Check out Time	Max guests	Price Per night
10:10	20:20	1	100

ii. After booking Page



q

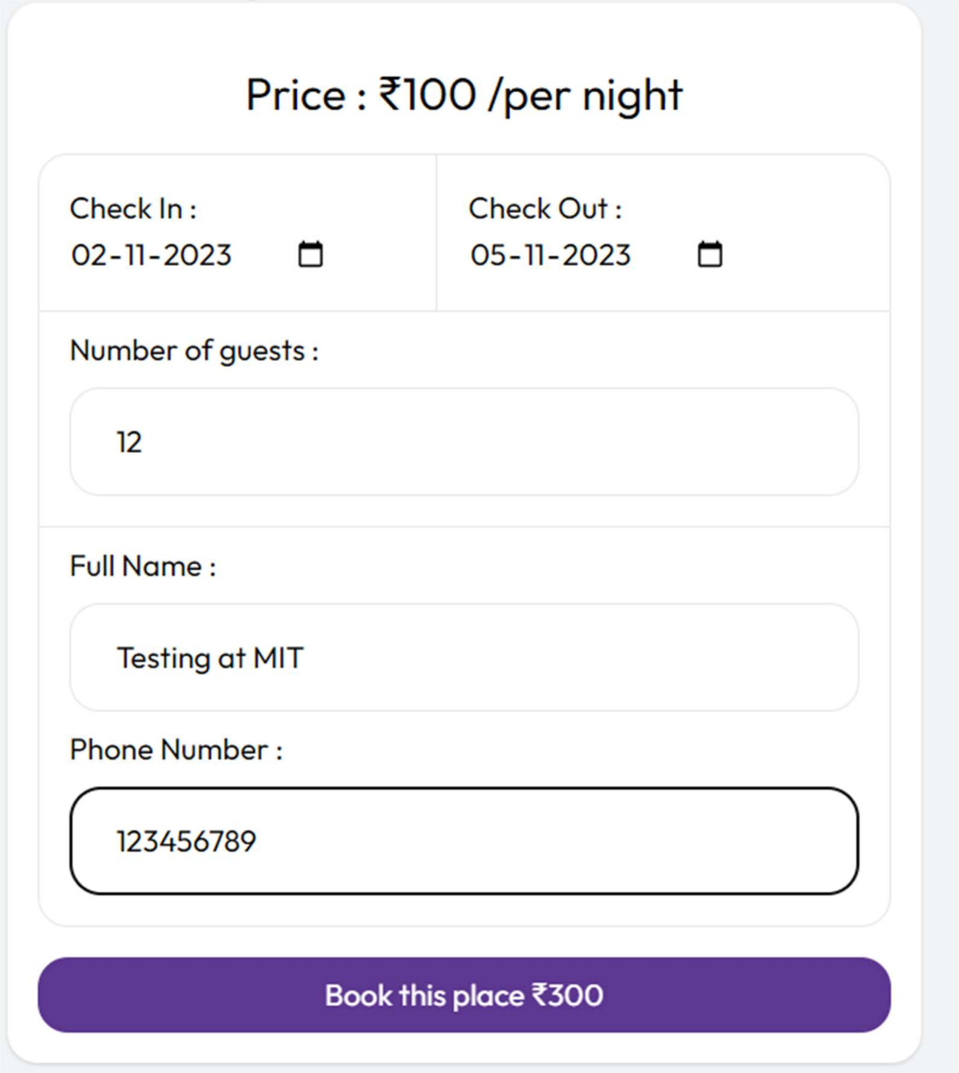
q

Booking Information

3 Nights 2023-11-02 → 2023-11-05

Total Price
₹300

iii. Booking Details



Price : ₹100 /per night

Check In : 02-11-2023

Check Out : 05-11-2023

Number of guests :

12

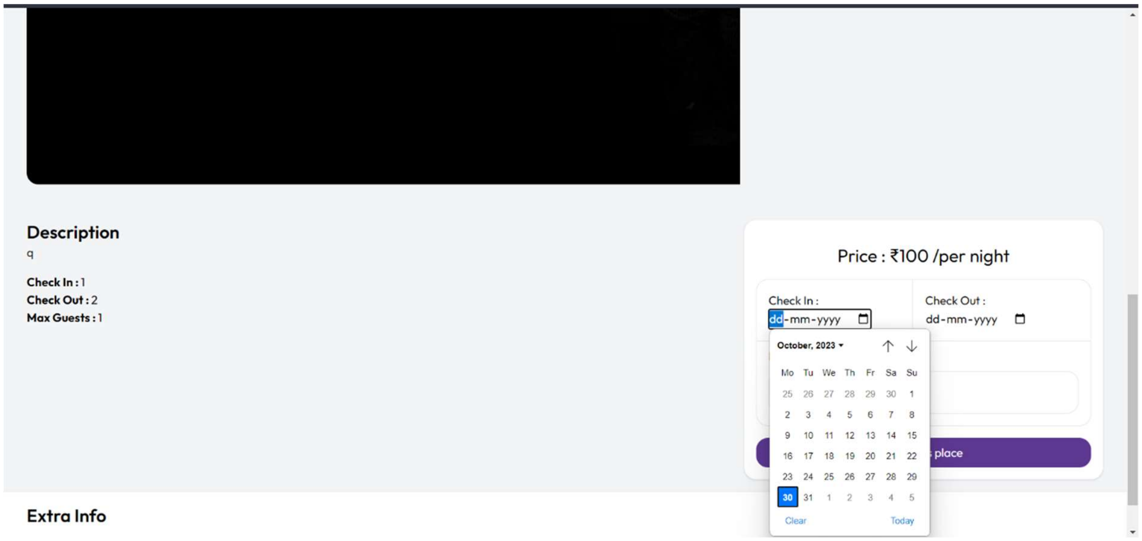
Full Name :

Testing at MIT

Phone Number :

123456789

iv. House info page



Description

q

Check In : 1
Check Out : 2
Max Guests : 1

Extra Info

Price : ₹100 /per night

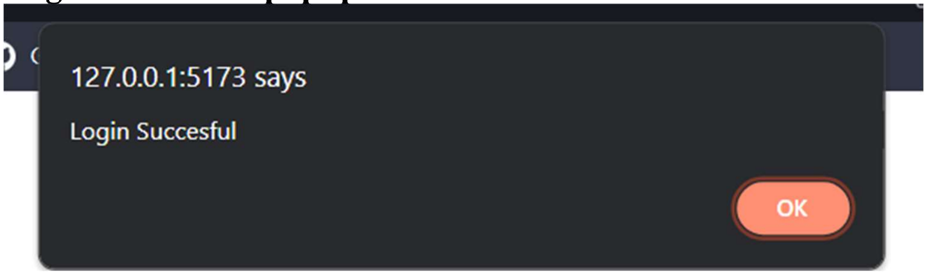
Check In : dd-mm-yyyy

Check Out : dd-mm-yyyy

October, 2023

Mo	Tu	We	Th	Fr	Sa	Su
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

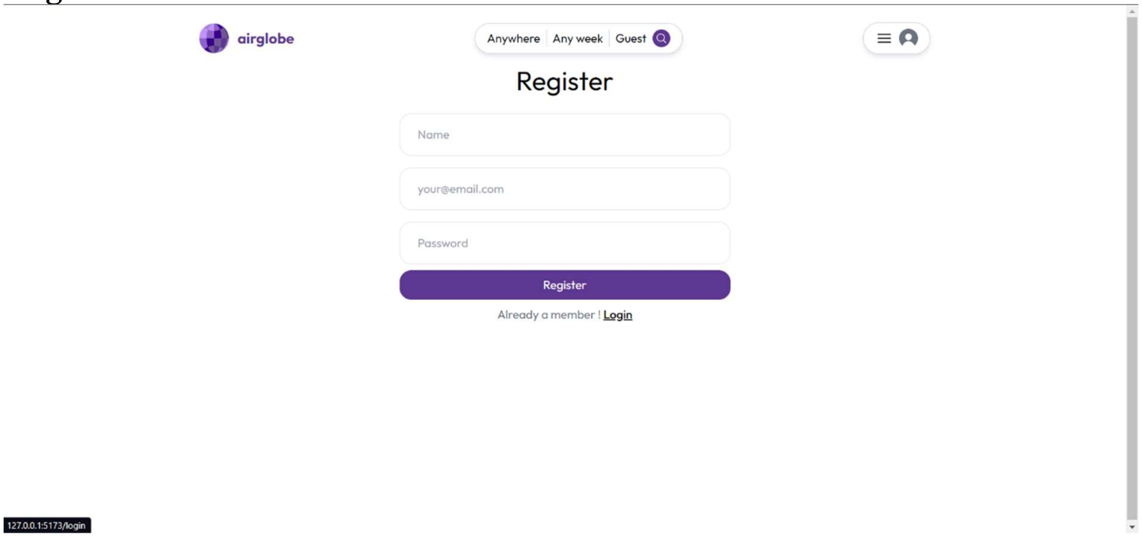
Login Successful popup



127.0.0.1:5173 says

Login Succesful

Registration Form



Register

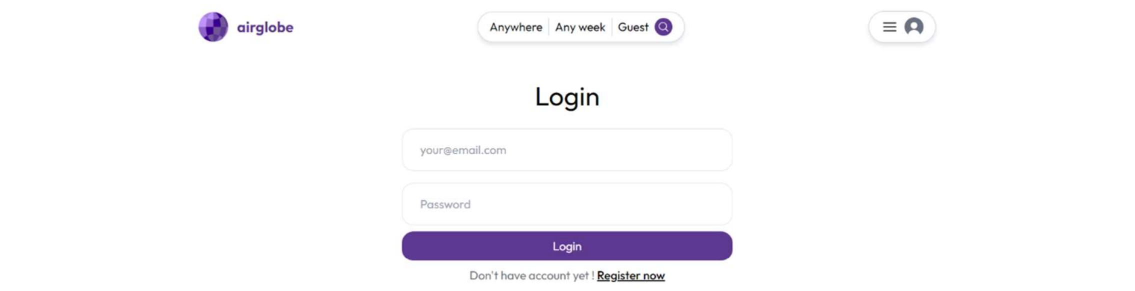
Name

your@email.com

Password

Register

Already a member ! Login



Login

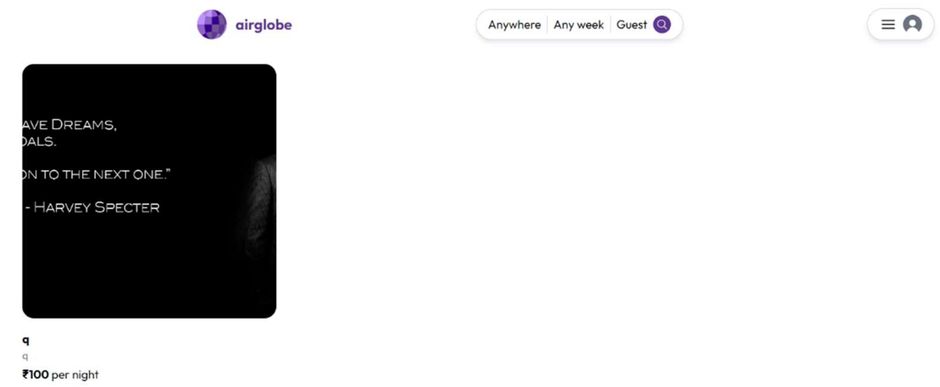
your@email.com

Password

Login

Don't have account yet ! Register now

Home Page



Conclusion:

In the contemporary landscape of the sharing economy and online hospitality, Airbnb stands as a trailblazer. Its platform revolutionized the way people find accommodations when traveling and the way individuals monetize their unused living spaces. The success of Airbnb has inspired numerous entrepreneurs and developers to venture into the world of online accommodations.

The Airbnb Clone project is a full-stack web application developed using the MERN (MongoDB, Express, React, Node.js) stack. This project aims to provide a seamless online platform for users to discover, book, and manage accommodations in various locations. Inspired by the popular Airbnb platform, our application offers a range of features including user registration and authentication, accommodation listings, booking functionality, and accommodation management. This report provides an overview of the project's goals, functionality, and the technologies employed.

The Airbnb Clone project aspires to offer a high-quality, reliable, and feature-rich web application that closely mirrors the functionality of Airbnb itself. Our project report will delve into the intricacies of our development process, exploring the architecture, implementation, and the challenges we encountered along the way. We are committed to providing a detailed account of our journey, and we invite you to join us in discovering the inner workings of our Airbnb-inspired application