«SOA-BASED Software Development»

George David Apostolidis
MInas Charakopoulos
Panagiotis Machairas
DIonisis Lougaris

How to install and run the application

Clone the repository and navigate to the root directory using these commands:

git clone https://github.com/GeorgeApos/trip-management-system.git
cd trip-management-system

Build the project with **Docker**:

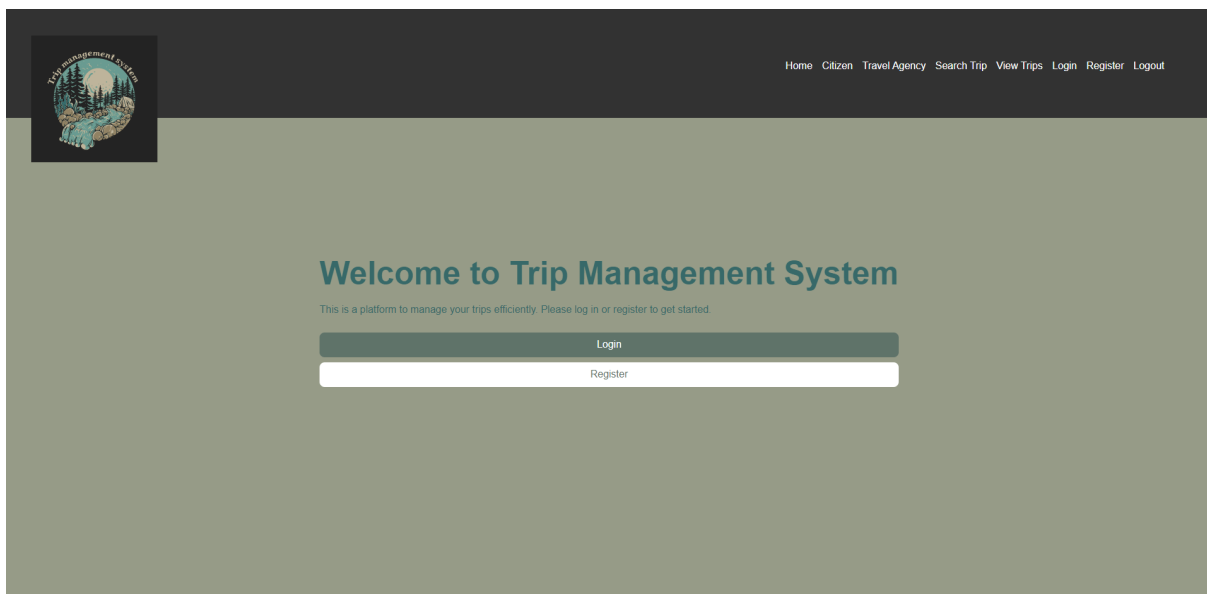docker-compose up --build

After the build process is complete, access the application by opening your web browser and going to http://localhost:3000.

Landing Page Component Documentation

Overview

The Landing Page component serves as the initial interface users encounter when accessing the Trip Management System. It provides a concise and inviting introduction to the platform, encouraging users to either log in or register to commence their journey in efficiently managing their trips.

Usage:

The LandingPage component serves as the welcoming gateway to the Trip Management System, greeting users with a bold "Welcome to Trip Management System" header. This sets a positive tone for their interaction. A concise paragraph offers a clear overview of

the platform's purpose, emphasizing efficient trip management. Prominently placed "Login" and "Register" buttons facilitate user access, with "Login" directing to "/login" for existing accounts and "Register" leading to "/register" for new account creation. This user-centric design prioritizes simplicity and accessibility, ensuring users can seamlessly engage with the platform based on their needs.

Register Component Documentation

Overview

The Register component facilitates user registration within the application, providing a form for users to input their information. It dynamically adjusts the input fields based on the selected user type (citizen or travel agency) and initiates the registration process upon clicking the "Register" button.

Register

Citizen	⌄

VAT

First Name / Company Name

Last Name

Email

Password

Register

Register

Travel Agency	⌄

VAT

First Name / Company Name
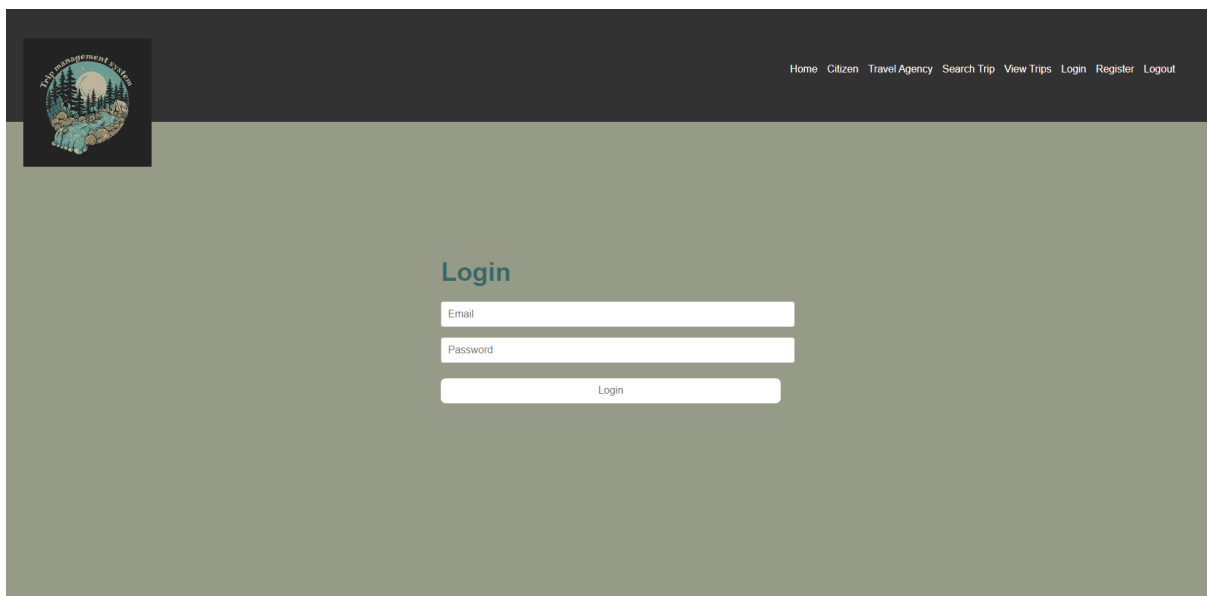
Email

Password

Register

Usage

The Register component serves as the user registration interface, featuring an intuitive form for users to input essential details. This includes selecting their user type, entering VAT information, and providing relevant name details based on their role. React's useState hook efficiently manages the component's dynamic state, accommodating various input fields like VAT, names, email, and password. The user type dropdown dynamically adjusts the form, displaying fields tailored to citizens or travel agencies. Clicking the "Register" button triggers an Axios request to the server, transmitting user credentials and details as URL parameters. Successful registrations store email and password in local storage for subsequent logins, redirecting users to the appropriate dashboard. Error handling is robust, logging encountered issues to the console for effective debugging. This multifunctional approach ensures a secure and streamlined user registration process in the application.

Login Component Documentation

Overview

The Login component is a user interface element designed to facilitate user authentication within your application. It provides a form for users to input their email and password, attempting to log in by sending a request to the server. The component is responsible for handling user input, making requests to the backend, and redirecting users based on their role.
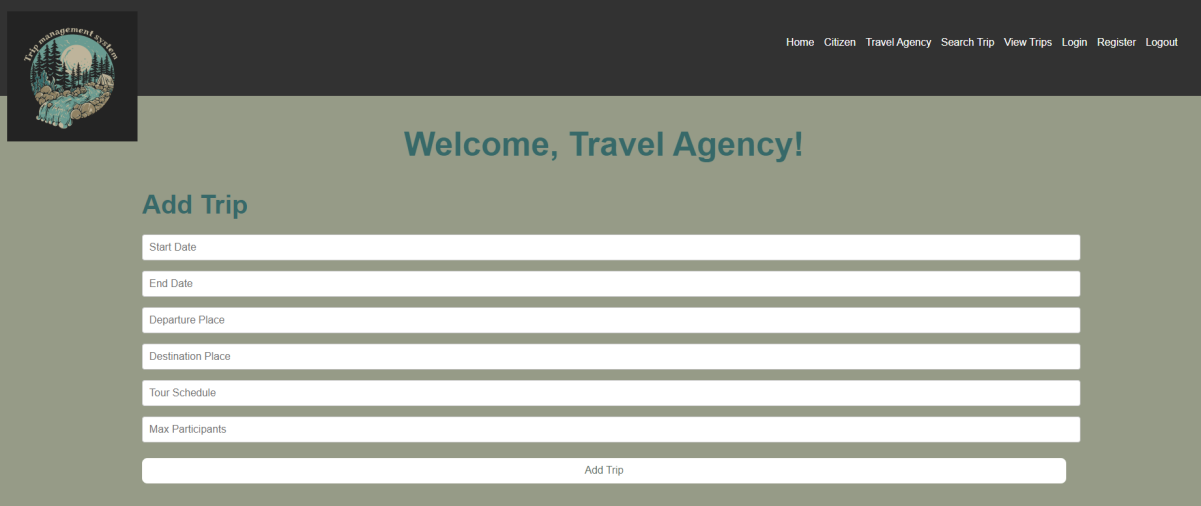

Usage

The Login component offers a user-friendly interface, allowing users to input their email address and password and initiating the login process with the "Login" button. Using React's useState hook, the component manages the state of the email and password input

fields. Upon activation, the "Login" button triggers an Axios request to the server, transmitting user credentials as a Basic Authentication header. Successful logins (HTTP status 200) prompt extraction of the user's role (either "citizen" or "travelAgency"), leading to redirection to the respective dashboard. Robust error handling logs encountered issues to the console for effective debugging. This comprehensive functionality ensures a smooth and secure user authentication experience within the application.

Travel Agency Component Documentation

Overview

The TravelAgencyDashboard component serves as the main dashboard for travel agencies within the Trip Management System. It provides travel agencies with the ability to add new trips by entering relevant trip details. The component is designed to enhance the user experience for travel agencies in managing their offered trips efficiently.
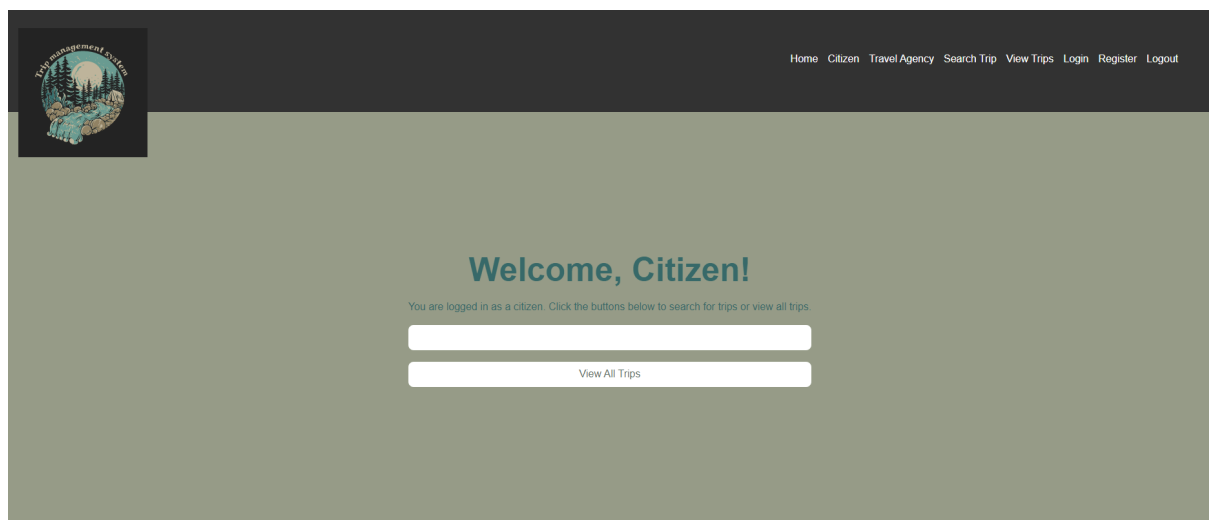

Usage

The TravelAgency component delivers a personalized experience for travel agencies with a welcoming "Welcome, Travel Agency!" heading upon rendering. The dashboard prominently features an "Add Trip" section, facilitating the streamlined addition of new trips. This section includes controlled input fields for trip details like "Start Date," "End Date," "Departure Place," "Destination Place," "Tour Schedule," and "Max Participants," all efficiently managed through React's useState hook for state management. The "Add Trip" button seamlessly integrates the trip addition functionality, triggering the handleAddTrip function. This function orchestrates the gathering of trip data, creation of an authentication header, and initiation of a POST request to the server endpoint "http://localhost:8080/travelAgency/addTrip." Robust error handling mechanisms ensure graceful management of potential issues during the trip addition process. The

TravelAgency component not only offers a welcoming interface but also empowers travel agencies to efficiently add and manage their trips within the application.

Citizen Landing Page Component Documentation

Overview

The Citizen Landing Page component serves as the welcoming interface for citizens upon successful login. It provides a user-friendly display with a personalized greeting, guiding citizens to explore available actions within the application. The primary purpose of this component is to offer quick access to key features, allowing citizens to effortlessly navigate to the trip search or view all trips sections.
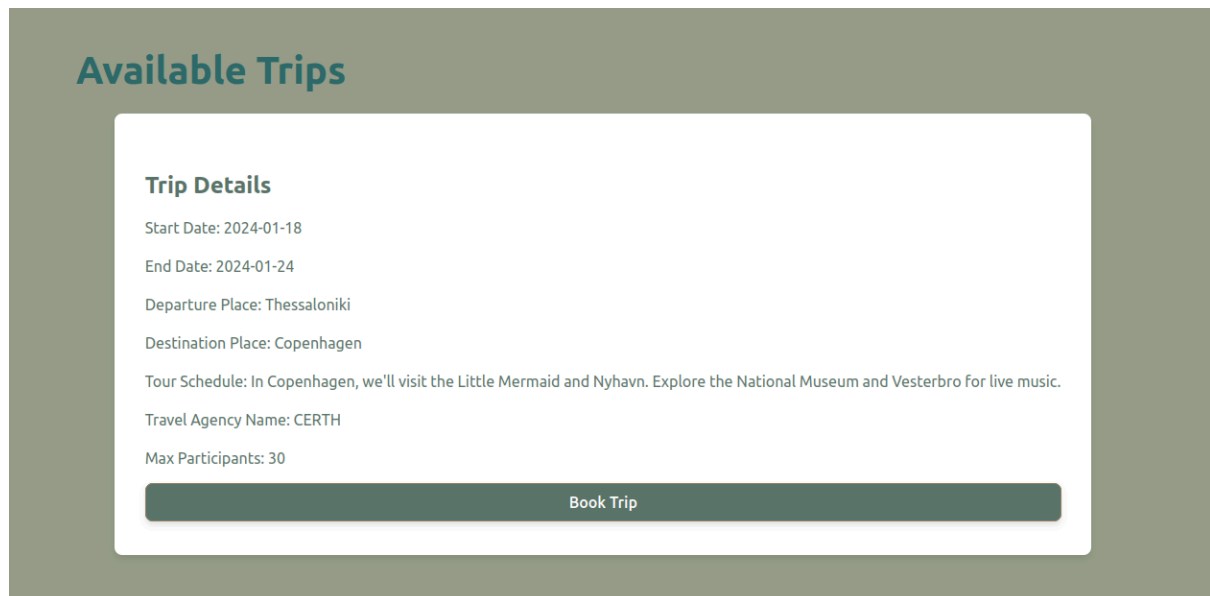

Usage

Upon rendering, the CitizenLandingPage component warmly welcomes users with a friendly "Welcome, Citizen!" heading, fostering a positive and inviting atmosphere. Following this, a descriptive paragraph informs citizens about their logged-in status and encourages them to interact with the application through prominently displayed buttons. The "Search for Trips" button, linked to the "/citizen/search-for-trip" route, facilitates a seamless trip search experience for users interested in specific travel opportunities. Additionally, the "View All Trips" button, located below the search button and linked to the "/citizen/all-trips" route, enables citizens to explore and access a comprehensive list of all available trips within the application. This user-centric design aims to guide citizens efficiently to sections catering to their travel needs.

View Trips Component Documentation

Overview

The View Trips component is responsible for presenting users with a list of available trips, allowing them to view trip details and book a trip if interested. This component is designed to enhance user engagement by providing a clear and organized display of available travel options.

Available Trips

Trip Details

Start Date: 2024-01-18

End Date: 2024-01-24

Departure Place: Thessaloniki

Destination Place: Copenhagen

Tour Schedule: In Copenhagen, we'll visit the Little Mermaid and Nyhavn. Explore the National Museum and Vesterbro for live music.

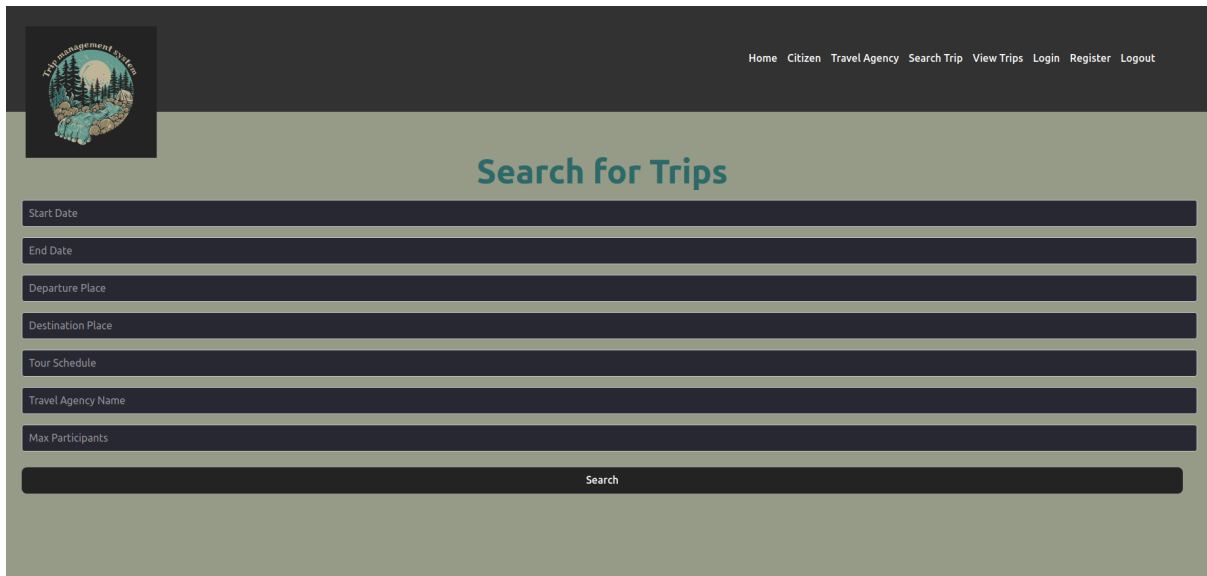Travel Agency Name: CERTH

Max Participants: 30

Book Trip

Usage

Upon rendering, the ViewTrips component initiates an HTTP GET request to the "http://localhost:8080/availableTours" endpoint, aiming to fetch available trips from the server. The component prominently starts with a "Available Trips" heading, clearly communicating its purpose to present users with a list of travel options. To efficiently handle data retrieval, the component employs the useEffect hook, executing a function to fetch trips when the component mounts. The fetched data is then stored in the trips state variable using the setTrips function, ensuring effective state management. The display of available trips is dynamically handled through the map function, generating a list that includes a separate box (styled as a div with the class "trip-box") for each trip. These boxes neatly encapsulate trip details such as destination, departure date, return date, tour schedule, and the maximum number of participants, contributing to an organized and visually appealing presentation. To enhance user interaction, each trip box features a "Book Now" button, providing users with the option to express interest in a specific trip. Clicking the button triggers the handleBookNow function, which initiates a POST request to the server for booking the selected trip. Subsequent success or failure messages are then logged to the console, offering users valuable feedback on their booking attempts. This comprehensive functionality ensures that the ViewTrips component provides users with a seamless and engaging experience in exploring and booking available trips.

Search Trips Component Documentation

Overview

The SearchTrip component is designed to allow citizens to search for trips based on various parameters. It provides a user-friendly interface with input fields for specifying search criteria and displays the search results, including detailed trip information.

Usage

Upon rendering, the SearchTrip component presents a search form and dynamically updates the displayed trips based on the specified search criteria. The search form in the SearchTrip component is prominently introduced with a clear focus on its primary purpose, inviting users to explore available trips. It features well-defined input fields controlled by React's useState hook, allowing users to input specific search criteria, including "Start Date," "End Date," "Departure Place," "Tour Schedule," "Travel Agency Name," and "Max Participants." Upon entering the desired parameters, users can engage the search functionality by clicking the "Search" button. This action triggers an HTTP POST request to the server, where the specified search criteria are sent as parameters. The search results are then dynamically displayed below the form, presenting each trip in a structured format within its own box. The detailed information provided for each trip encompasses essential details such as trip name, destination, start date, end date, departure place, tour schedule, travel agency name, and maximum participants. Notably, each displayed trip is accompanied by a "Book Trip" button; however, it currently serves as a placeholder without a defined functionality, hinting at potential future features or developments within the application.