R0845035

Roji Pun

BACKEND-AIRBNB Project

Project Summary Overview  
I created a web API for a booking system with LiteDB as the database. The API allows you to manage users, bookings, camping places, availability, and comment ratings. The API is built with ASP.NET Core and handles HTTP requests using a RESTful method.  
  
Key Components  
1. Controllers: User Controller  
  
LogIn: I implemented user authentication to deliver a token for valid credentials.  
Register: I established a registration endpoint to ensure that new users' passwords and confirmations match.  
Getuserinfo: This method returns all users with admin privileges.  
GetAll: I created this API to list all users with admin access.  
Get: Using this function, I retrieve user information by ID.  
Update: I handle user information updates based on ID.  
Delete: I included feature to delete users.

GetCommentsRatingsBySpotId: I created this endpoint to retrieve comments and ratings for specific camping areas.  
AddCommentRating: I developed a way for adding new comments and ratings.

CampingSpotController  
  
GetAllCampingSpots: I created a method for retrieving a list of all camping locations.  
GetCampingSpotById: This endpoint returns information about a camping spot by ID.  
CreateCampingSpot: I developed this endpoint to allow administrators to create new camping spaces.

BookingController  
  
BookCampingSpot: I included functionality for booking a camping spot, including conflict resolution if the spot is already reserved.  
GetBookingsByUserId: This method returns the bookings for a specified user.  
GetBookingById: I created a method to retrieve booking information by ID.  
UpdateBooking: I included the ability for administrators to amend booking details.  
DeleteBooking: This technique enables administrators to delete bookings.  
AvailabilityController  
  
Create availability: I designed this endpoint Administrators can add availability records.  
GetAvailabilityBySpotId: This method returns availability information for a given camping spot.  
UpdateAvailability: I created functionality for updating availability records.  
Delete Availability: I provided a technique for administrators to erase availability records.  
Data Context  
I utilized the IDataContext interface to communicate with the LiteDB database. Each controller uses this interface to conduct CRUD actions for various models.  
  
Models  
User: This class represents users and contains information about their authentication and profiles.  
Comment Rating: This represents comments and ratings for camping locations.  
Camping Spot: Describes several camping areas, including location and amenities.  
Booking: A reservation made by a user for a camping spot.  
Availability: Indicates the availability of a camping area.  
Security: I used role-based and policy-based authorization to manage access. Certain tasks, such as making and modifying Authorization is required for creating or removing records.  
Error Handling: I incorporated error handling to manage exceptions and respond appropriately to various conditions.  
I used RESTful endpoints with normal HTTP methods (GET, POST, PUT, DELETE) for resource interactions.  
Authorization is enforced to guarantee that only authorized users can carry out sensitive operations.  
Conclusion  
I developed a powerful API for managing bookings and related entities in a camping reservation system. The project uses LiteDB for data storage and ASP.NET Core to create and secure the API.