**Bus Booking System API Documentation**

**Base URL:** https://bus-booking-api-ezw2.onrender.com/api

**Authentication Endpoints**

**1. User Registration**

**Endpoint:** POST: /api/auth/register  
**Description:** Registers a new user.  
**Request Body (JSON):**

{

"name": "John Doe",

"email": "john@example.com",

"password": "SecurePassword123"

}

**Response (Success - 201):**

{

"message": "User registered successfully",

"user": {

"\_id": "65a8e1c2d2e5b2c6c4d52b98",

"name": "John Doe",

"email": "john@example.com",

"role": "user"

}

}

**Response (Error - 400):**

{

"error": "User already exists"

}

**2. User Login**

**Endpoint:** POST: /api/auth/login  
**Description:** Logs in a user and returns an access token.  
**Request Body (JSON):**

{

"email": "john@example.com",

"password": "SecurePassword123"

}

**Response (Success - 200):**

{

"message": "Login successful",

"token": "eyJhbGciOiJIUzI1NiIsInR5...",

"refreshToken": "3a4bc5de6f7g8h9ijk..."

}

**Response (Error - 401):**

{

"error": "Invalid credentials"

}

**3. User Logout**

**Endpoint:** POST: /api/auth/logout  
**Description:** Logs out the user by invalidating the refresh token.  
**Headers:**

{

"Authorization": "Bearer <your\_access\_token>"

}

**Response (Success - 200):**

{

"message": "User logged out successfully"

}

**Response (Error - 401):**

{

"error": "Unauthorized"

}

**Bus Management Endpoints**

**4. Get All Buses**

**Endpoint:** GET: /api/buses  
**Description:** Retrieves a list of all available buses.  
**Headers:**

{

"Authorization": "Bearer <your\_access\_token>"

}

**Response (Success - 200):**

[

{

"\_id": "65a8e1c2d2e5b2c6c4d52b98",

"name": "Bus1",

  "number": "TN-45-7788",

  "capacity": 100,

"availableSeats": 100,

"routeId": "65a8e1c2d2e5b2c6c4d52a12"

}

]

**5. Add a New Bus (Admin Only)**

**Endpoint:** POST: /api/buses  
**Description:** Admin can add a new bus.  
**Headers:**

{

"Authorization": "Bearer <admin\_access\_token>"

}

**Request Body (JSON):**

{

"name": "Bus1",

  "number": "TN-45-7788",

  "capacity": 100,

  "availableSeats": 100,

"routeId": "65a8e1c2d2e5b2c6c4d52a12"

}

**Response (Success - 201):**

{

"message": "Bus added successfully",

"bus": {

"\_id": "65a8e1c2d2e5b2c6c4d52b98",

"name": "Bus1",

  "number": "TN-45-7788",

  "capacity": 100,

  "availableSeats": 100,

}

}

**6. Update a Bus (Admin Only)**

**Endpoint:** PUT: /api/buses/:busId  
**Description:** Updates a bus’s details.  
**Headers:**

{

"Authorization": "Bearer <admin\_access\_token>"

}

**Request Body (JSON):**

{

"name": "Bus1",

  "number": "TN-45-7788",

  "capacity": 60,

  "availableSeats": 60,

"routeId": "65a8e1c2d2e5b2c6c4d52a12"

}

**Response (Success - 200):**

{

"message": "Bus updated successfully"

}

**7. Delete a Bus (Admin Only)**

**Endpoint:** DELETE: /api/buses/:busId  
**Description:** Deletes a specific bus.  
**Headers:**

{

"Authorization": "Bearer <admin\_access\_token>"

}

**Response (Success - 200):**

{

"message": "Bus deleted successfully"

}

**Route Management Endpoints**

**8. Get All Routes**

**Endpoint:** GET: /api/routes  
**Description:** Retrieves all available bus routes.  
**Headers:**

{

"Authorization": "Bearer <your\_access\_token>"

}

**Response (Success - 200):**

[

{

"\_id": "65a8e1c2d2e5b2c6c4d52a12",

"source": "Chennai",

    "destination": "Bangalore",

    "stops": ["Vellore", "Krishnagiri", "Hosur"]

}

]

**9. Add a Route (Admin Only)**

**Endpoint:** POST: /api/routes  
**Description:** Admin can add a new route.  
**Headers:**

{

"Authorization": "Bearer <admin\_access\_token>"

}

**Request Body (JSON):**

{

"source": "Chennai",

    "destination": "Bangalore",

    "stops": ["Vellore", "Krishnagiri", "Hosur"]

}

**Response (Success - 201):**

{

"message": "Route added successfully"

}

**Update a Route (Admin Only)**

**Endpoint:** PUT: /api/routes/:routeId  
**Description:** Updates details of an existing bus route.  
**Headers:**

{

"Authorization": "Bearer <admin\_access\_token>"

}

**Request Body (JSON):**

{

"source": "Chennai",

    "destination": "Hosur",

    "stops": ["Vellore", "Krishnagiri"]

}

**Response (Success - 200):**

{

"message": "Route updated successfully"

}

**Response (Error - 404):**

{

"error": "Route not found"

}

**Booking Management Endpoints**

**10. Get All Bookings**

**Endpoint:** GET: /api/bookings  
**Description:** Retrieves all bookings for the logged-in user.  
**Headers:**

{

"Authorization": "Bearer <your\_access\_token>"

}

**Response (Success - 200):**

[

{

"\_id": "67b7481933ae534114b71148",

    "user": "67b72cffb6236ef436786bf7",

    "bus": null,

    "seatNumber": 5,

    "status": "cancelled",

}

]

**11. Book a Seat**

**Endpoint:** POST: /api/bookings/book  
**Description:** Books a seat on a bus.  
**Headers:**

{

"Authorization": "Bearer <your\_access\_token>"

}

**Request Body (JSON):**

{

"busId": "67b746d133ae534114b7113f",

  "seatNumber": 5

}

**Response (Success - 201):**

{

"message": "Booking confirmed",

"bookingId": "65a8e1c2d2e5b2c6c4d52f67"

}

**12. Cancel a Booking**

**Endpoint:** POST: /api/bookings/cancel/:bookingId  
**Description:** Cancels a booking.  
**Headers:**

{

"Authorization": "Bearer <your\_access\_token>"

}

**Response (Success - 200):**

{

"message": "Booking canceled successfully"

}