**SARAN TRAVELS**

**Documentation on BusTicketBooking:**

1. **Create the connection to MongoDB.**
2. **Create a Schema for Ticket booking.**

**Total 3 Schemas -> 1. UserSchema**

**2. BusSchema**

**3. SeatSchema**

**UserSchema = {**

**Name: username,**

**Email: user mail,**

**Password: user password,**

**Booked seats: [**

**bus\_id:**

**seat number:**

**]**

**},**

**BusSchema = {**

**Name: bus name,**

**Seats:[seatSchema]**

**},**

**SeatSchema = {**

**Seat number:**

**Status: default: false,**

**Booked by: user\_id**

**}**

1. **Make the endpoints:**
   1. **Create the bus by seats**
   2. **Get the bus details**
   3. **Create the User**
   4. **Get the User details and seats booked by him**
   5. **Booking the seat by user in the bus**

**EXAMPLE :**

1. **Bus creation input:**

{

    "name":" tirupathi bus",

    "seats”: [

        {"seat\_number":1},

        {"seat\_number":2},

        {"seat\_number":3}

        ]

}

1. **User creation input:**

{

    "name": "saran",

    "email": "saran@gmail.com",

    "password": "saran"

}

**3. Ticket Booking input:**

{

    "user\_id":"641abf4d47e5f91d87aa90db",

    "bus\_id":"641ac2b847e5f91d87aa90e3",

    "seat\_number":1

}

**Error Codes for Rest API :**

* if error occurs in request, wrong request;

**400 Bad Request**: The request could not be understood by the server due to incorrect syntax. The client SHOULD NOT repeat the request without modifications.

* if user , bus , seat not found;

**404 Not Found:** The server can not find the requested resource.