

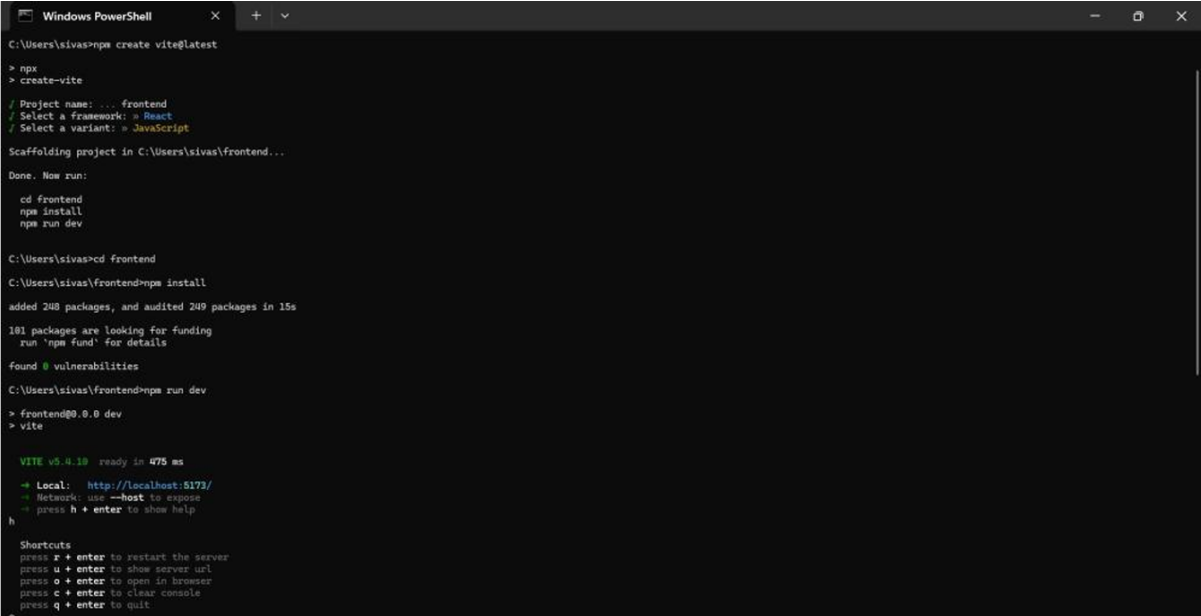
# Screenshots of Installation and Executions :

## 1. PRE-REQUISITES :

Here are the key prerequisites for developing a full-stack application using Node.js, Express.js, MongoDB, React.js:

### ✓ Vite:

Vite is a new frontend build tool that aims to improve the developer experience for development with the local machine, and for the build of optimized assets for production (go live). Vite (or ViteJS) includes: a development server with ES \_native\_ support and Hot Module Replacement; a build command based on rollup. Installation : `npm create vite@latest`



```
Windows PowerShell
C:\Users\sivas>npm create vite@latest

> npx
> create-vite

? Project name: ... frontend
? Select a framework: > React
? Select a variant: > JavaScript
Scaffolding project in C:\Users\sivas\frontend...

Done. Now run:

  cd frontend
  npm install
  npm run dev

C:\Users\sivas>cd frontend
C:\Users\sivas\frontend>npm install

added 248 packages, and audited 249 packages in 15s

181 packages are looking for funding
  run 'npm fund' for details

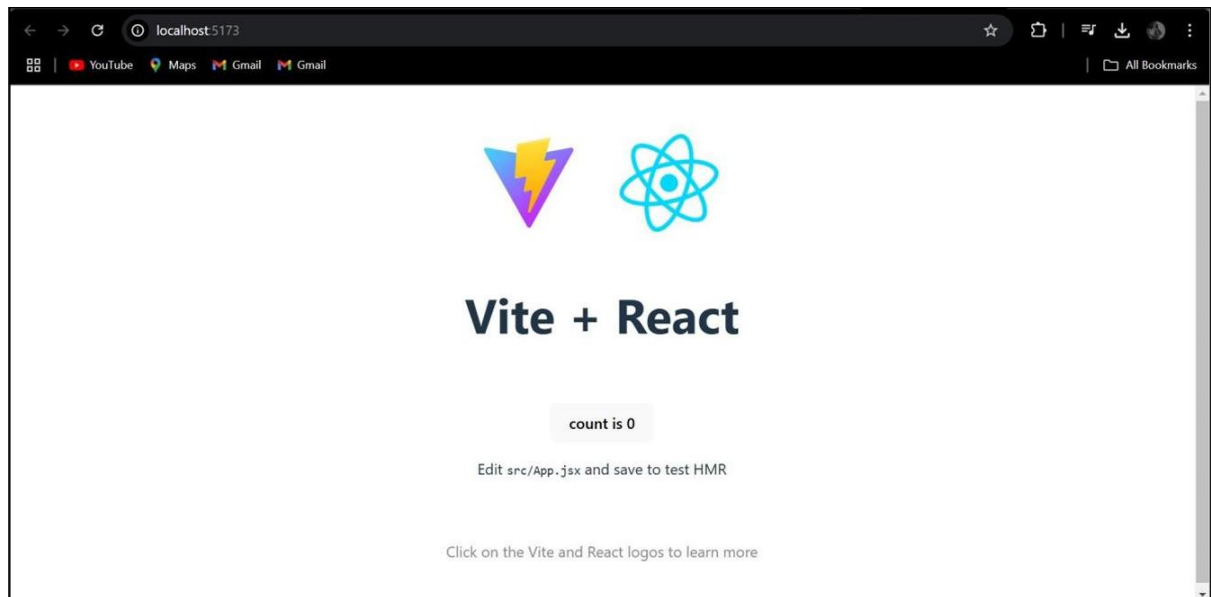
found 0 vulnerabilities

C:\Users\sivas\frontend>npm run dev

> frontend@0.0.0 dev
> vite

VITE v5.4.19 ready in 475 ms
➜ Local:   http://localhost:5173/
➜ Network: use --host to expose
➜ press h + enter to show help
h

Shortcuts
press r + enter to restart the server
press u + enter to show server url
press o + enter to open in browser
press c + enter to clear console
press q + enter to quit
q
```



### ➤ ✓ Node.js and npm:

**Node.js is a powerful JavaScript runtime environment that allows you to run JavaScript code on the server-side. It provides a scalable and efficient platform for building network applications. Install Node.js and npm on your development machine, as they are required to run JavaScript on the server-side. Download: <https://nodejs.org/en/download/> Installation instructions: <https://nodejs.org/en/download/package-manager/> Run “npm init” to get default dependencies**

### ➤ Express.js:

**Express.js is a fast and minimalist web application framework for Node.js. It simplifies the process of creating robust APIs and web applications, offering features like routing, middleware support, and modular architecture. Install Express.js, a web application framework for Node.js, which handles server-side routing, middleware, and API development. Installation: Open your command prompt or terminal and run the following command: `npm install express`**

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\BALAJI 07> npm install express

added 65 packages in 1s

13 packages are looking for funding
  run 'npm fund' for details
PS C:\Users\BALAJI 07> |
```

## ➤ ✓ MongoDB:

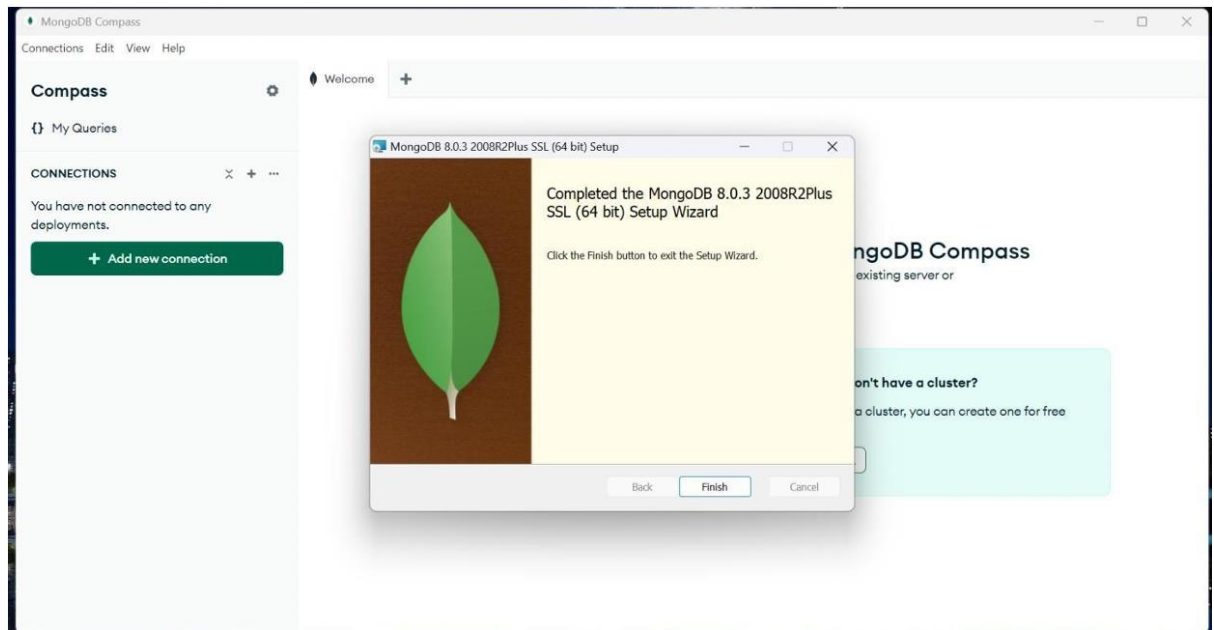
**MongoDB is a flexible and scalable NoSQL database that stores data in a JSON-like format. It provides high performance, horizontal scalability, and seamless integration with Node.js, making it ideal for handling large amounts of structured and unstructured data. Set up a MongoDB database to store your application's data.**

**Download:**

<https://www.mongodb.com/try/download/community> 2

**Installation instructions:**

<https://docs.mongodb.com/manual/installation/>



- ✓ **React.js:** React.js is a popular JavaScript library for building user interfaces. It enables developers to create interactive and reusable UI components, making it easier to build dynamic and responsive web applications. Install React.js, a JavaScript library for building user interfaces.

Follow the installation guide: <https://reactjs.org/docs/create-a-new-react-app.html>

- ✓ **HTML, CSS, and JavaScript:** Basic knowledge of HTML for creating the structure of your app, CSS for styling, and JavaScript for client-side interactivity is essential.
- ✓ **Database Connectivity:** Use a MongoDB driver or an Object-Document Mapping (ODM) library like Mongoose to connect your Node.js server with the MongoDB database and perform CRUD (Create, Read, Update, Delete) operations.
- ✓ **Front-end Framework:** Utilize React.js to build the user-facing part of the application, including entering booking room, status of the booking, and user interfaces for the admin dashboard. For making better UI we have also used some libraries like material UI and bootstrap.

Install the required dependencies by running the following commands:

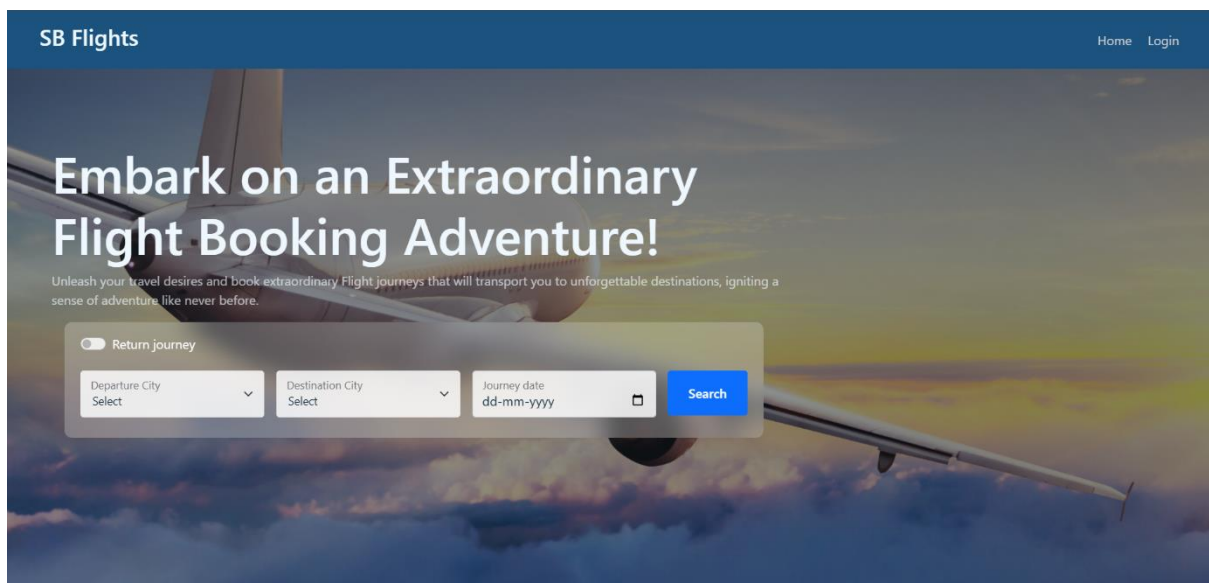
```
cd frontend || npm install cd ../backend || npm install
```

**Start the Development Server:**

- **To start the development server, execute the following command: npm start**

## SCREENSHOTS OF FLIGHT BOOKING APP:

USER PAGE, LANDING PAGE, INTERACTION PAGE:



## USER LOGIN:

### Register

Username

santhoshnew

Email address

santhoshnew@gmail.com

Password

\*\*\*\*\*

User type

▼

User type

Admin

Customer

Flight Operator

## LOGIN AS FLIGHT OPERATOR:

### Add new Flight

Flight Name

santhoshoperator

Flight Id

1

Departure City

Chennai

▼

Departure Time

11:11 PM

⌚

Destination City

Banglore

▼

Arrival time

07:01 AM

⌚

Total seats

120

Base price

17000

Add now

# CONTROLL OVER FLIGHTS AS OPERATOR:

### Add new Flight

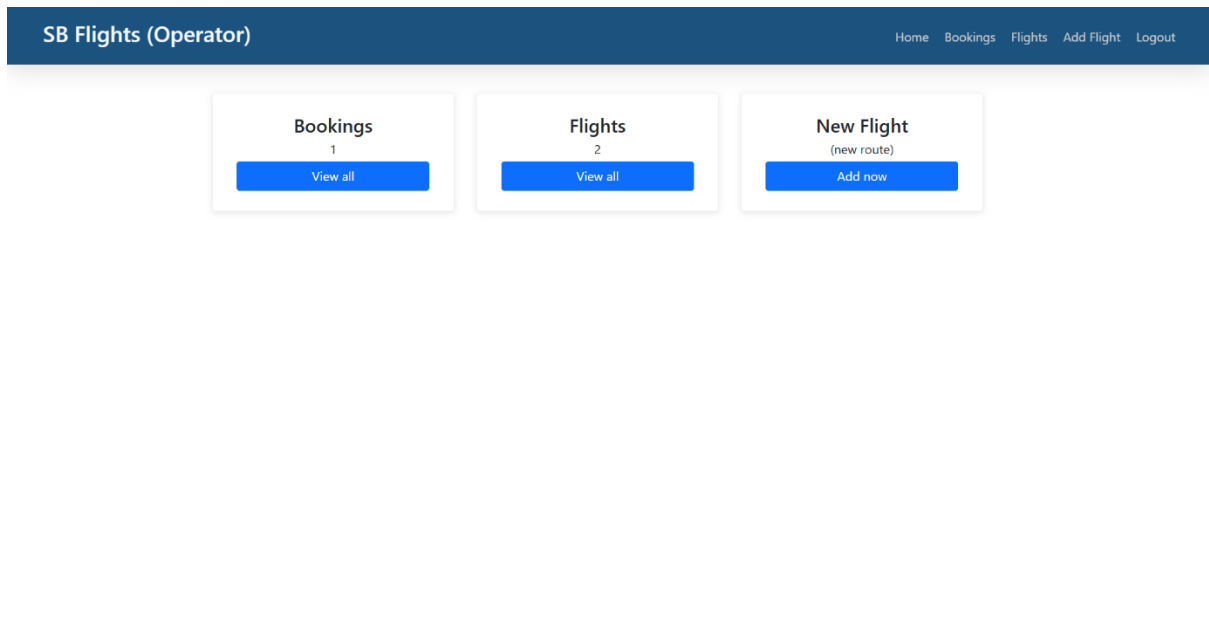
Flight Name santhoshoperator	Flight Id 1
Departure City Chennai	Departure Time 11:11 PM
Destination City Banglore	Arrival time 07:01 AM
Total seats 120	Base price 17000

Add now

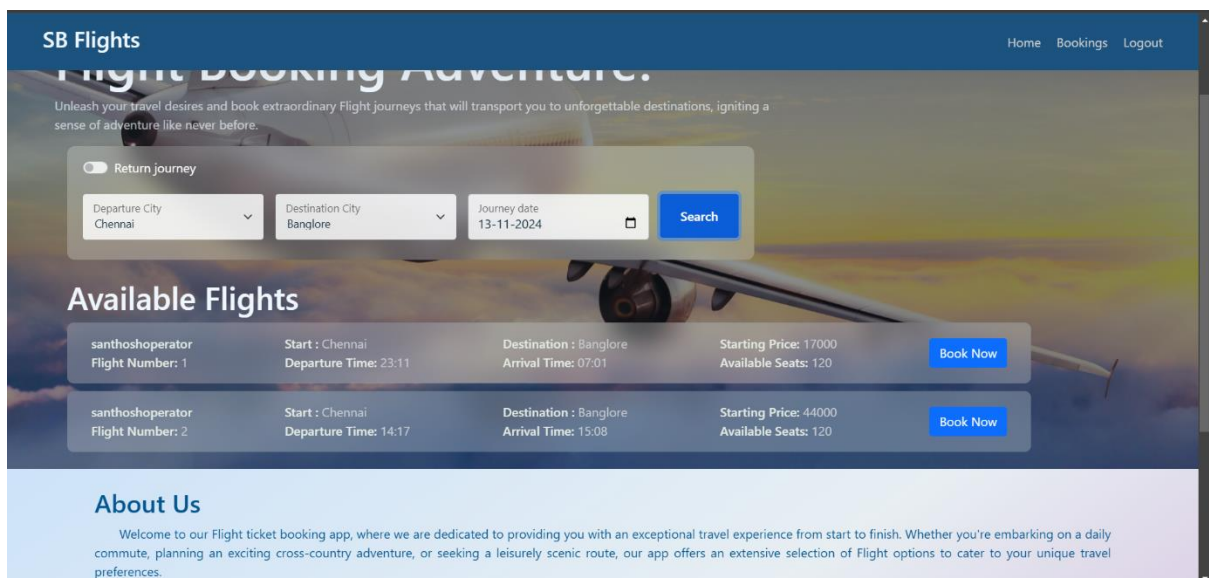
## All Flights

\_id: 67338edc494484d6fe136897  
Flight Id: 1 Flight name: santhoshoperator  
Starting station: Chennai Departure time: 23:11  
Destination: Banglore Arrival time: 07:01  
Base price: 17000 Total seats: 120

Edit details



## SEARCHING SEATS AND TICKETS:





## Book ticket

Flight Name: santhoshoperator

Flight No: 1

Base price: 17000

Email  
santhosh001@gmail.comMobile  
1112225556No of passengers  
1Journey date  
13-11-2024Seat Class  
Select

Passenger 1

Name  
santhoshAge  
25

Total price: 0

Book now

localhost:3000 says  
booking successful

OK

## Book ticket

Flight Name: santhoshoperator

Flight No: 1

Base price: 17000

Email  
santhosh001@gmail.comMobile  
1112225556No of passengers  
1Journey date  
13-11-2024Seat Class  
Select

Passenger 1

Name  
santhoshAge  
25

Total price: 0

Book now

## Bookings

Booking ID: 67338fa7494484d6fe1368ac

Mobile: 1112225556 Email: santhosh001@gmail.com

Flight Id: 1 Flight name: santhoshoperator

On-boarding: Chennai Destination: Bangalore

Passengers: Seats: undefined-1

1. Name: santhosh, Age: 25

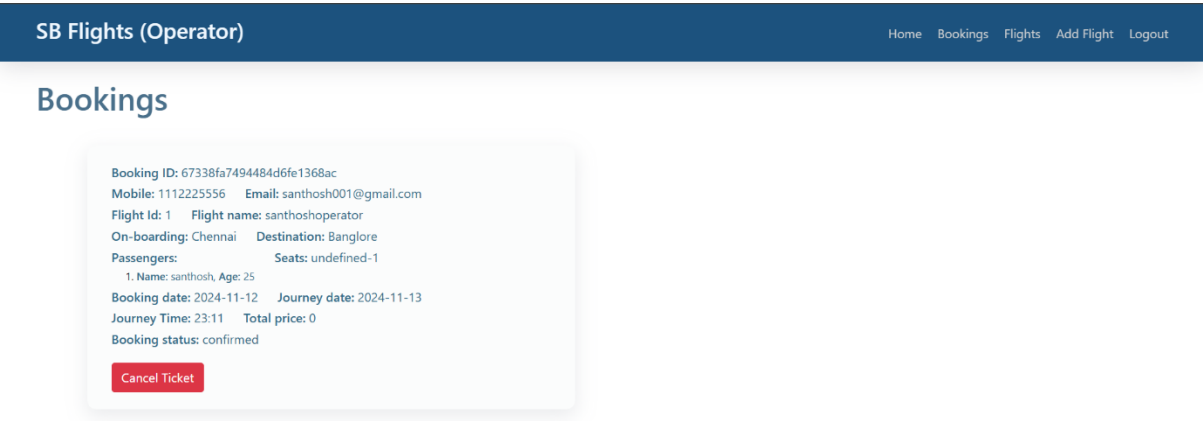
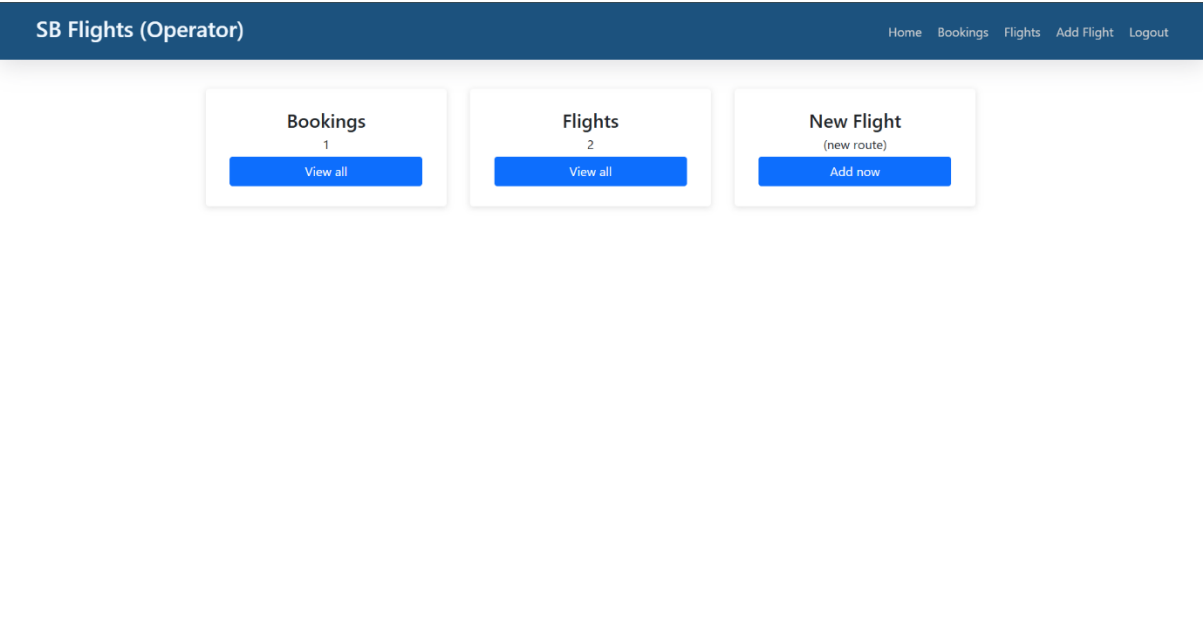
Booking date: 2024-11-12 Journey date: 2024-11-13

Journey Time: 23:11 Total price: 0

Booking status: confirmed

Cancel Ticket

# FLIGHT MANAGEMENT:



# MONGODB COMPASS DATA:

MongoDB Compass - Project Demo/FlightBookingMERN.bookings

ConnectionsEditViewCollectionHelp

Compass

My Queries

CONNECTIONS (1)

Search connections

▼Project Demo

▼FlightBookingMERN

bookings

flights

users

admin

config

local

bookingsflights+

Project Demo > FlightBookingMERN > bookings

Documents0AggregationsSchemaIndexes1Validation

Type a query: { field: 'value' } or [Generate query](#)

ADD DATA

EXPORT DATA

UPDATE

DELETE

```
_id: ObjectId('67338fa7494484d6fe1368ac')
user: ObjectId('67338968494484d6fe1367d8')
flight: ObjectId('67338edc494484d6fe136897')
flightName: "santhoshoperator"
flightId: "1"
departure: "Chennai"
destination: "Banglore"
email: "santhosh001@gmail.com"
mobile: "1112225556"
seats: "undefined-1"
passengers: Array (1)
totalPrice: 0
journeyDate: 2024-11-13T00:00:00.000+00:00
journeyTime: "23:11"
seatClass: ""
bookingStatus: "confirmed"
bookingDate: 2024-11-12T17:25:59.828+00:00
__v: 0
```

Connections Edit View Collection Help

### Compass

{ } My Queries

CONNECTIONS (1)

Search connections

- Project Demo
  - FlightBookingMERN
    - bookings
    - flights**
    - users
  - admin
  - config
  - local

Project Demo > FlightBookingMERN > flights

Documents 0 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#)

EXPLAIN Reset Find Options

ADD DATA EXPORT DATA UPDATE DELETE

25 1 - 2 of 2

```
{ "_id": ObjectId("67338edc494484d6fe136897"),
  "flightName": "santhosoperator",
  "flightId": "1",
  "origin": "Chennai",
  "destination": "Banglore",
  "departureTime": "23:11",
  "arrivalTime": "07:01",
  "basePrice": 17000,
  "totalSeats": 120,
  "__v": 0 }
```

```
{ "_id": ObjectId("67338f1f494484d6fe1368a5"),
  "flightName": "santhosoperator",
  "flightId": "2",
  "origin": "Chennai",
  "destination": "Banglore",
  "departureTime": "14:17",
  "arrivalTime": "15:08",
  "basePrice": 44000,
  "totalSeats": 120,
  "__v": 0 }
```

Connections Edit View Collection Help

### Compass

{ } My Queries

CONNECTIONS (1)

Search connections

- Project Demo
  - FlightBookingMERN
    - bookings
    - flights
    - users**
  - admin
  - config
  - local

Project Demo > FlightBookingMERN > users

Documents 4 Aggregations Schema Indexes 2 Validation

Type a query: { field: 'value' } or [Generate query](#)

EXPLAIN Reset Find Options

ADD DATA EXPORT DATA UPDATE DELETE

25 1 - 6 of 6

```
{ "_id": ObjectId("67338968494484d6fe1367d8"),
  "username": "santhosh",
  "email": "santhosh@gmail.com",
  "usertype": "customer",
  "password": "52b5185y6DyKRd9gTNCJPv8IUAB1.kyyqgvw3p11f6ytjXK7eP16oHFQ0oq2",
  "approval": "approved",
  "__v": 0 }
```

```
{ "_id": ObjectId("67338997494484d6fe1367dd"),
  "username": "santhosh1",
  "email": "santhosh1@gmail.com",
  "usertype": "admin",
  "password": "52b5185l00FHfNJD328a1Rb1WkGw2x.gjuTbRSW0poMkyxKx9s0YduInfyu",
  "approval": "approved",
  "__v": 0 }
```

```
{ "_id": ObjectId("67338aaa494484d6fe1367f6"),
  "username": "santhosh1",
  "email": "santhosh1@gmail.com",
  "usertype": "admin",
  "password": "52b5185HLh18Ej3QGG10NXcs9bkKuZ9E/bVzeauLvurcj0UumZyY248iCPu",
  "approval": "approved",
  "__v": 0 }
```

```
{ "id": ObjectId("67338b4d494484d6fe1368e3") }
```