

1. Log In to MongoDB Atlas

1. Visit the [MongoDB Atlas website](#) and log in.
 2. Navigate to your project and click on the **Cluster** where you want to connect your application.
-

2. Create a Database User

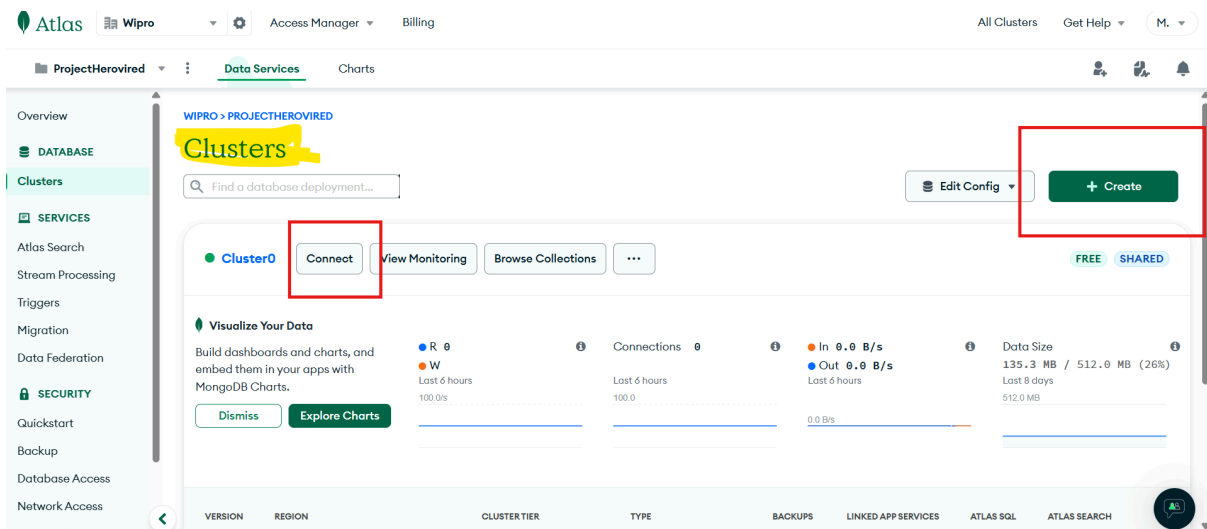
1. Go to the **Database Access** tab in the left sidebar.
2. Click on **Add New Database User**.
3. Fill in the details:
 - **Username:** Enter a username (e.g., vika2k).
 - **Password:** Use the auto-generate feature or manually create a secure password (e.g., 5j4tyZQSnucMiK1h).
 - Save this password securely.
4. **Assign Roles:**
 - Choose **Read and Write to Any Database** or a custom role like `readWrite` for your specific database (e.g., `travelDatabase`).
5. Click **Add User** to create the database user.

3. Whitelist Your IP Address

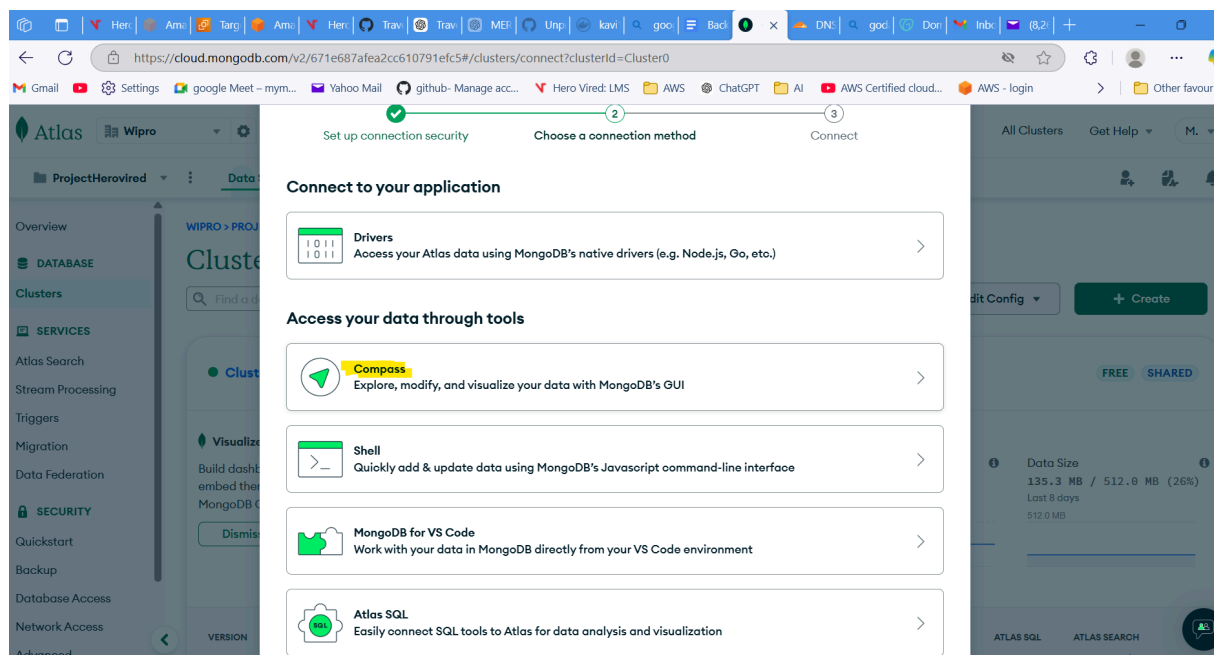
1. Go to the **Network Access** tab in the left sidebar.
 2. Click on **Add IP Address**.
 3. Add the IP address(es) from which you'll connect to the database:
 - **Your Local Machine:** Click **Add My Current IP Address**.
 - **EC2 Instance:** Use the public IP of your EC2 instance.
 - For universal access (not recommended for production), you can use `0.0.0.0/0`.
 4. Save changes and wait for the update to complete.
-

4. Obtain the Connection String

1. Go to the **Clusters** tab and click **Connect** next to your cluster.
2. Choose **Connect Your Application**.
3. Copy the connection string template:



mongodb+srv://<username>:<password>@cluster0.b7mwy.mongodb.net/<database>?retryWrites=true&w=majority



4. Replace the placeholders:

- <username>: Your database username (e.g., vika2k).
- <password>: Your database password (e.g., 5j4tyZQSnucMiK1h).
- <database>: The name of your database (e.g., traveldatabase).

5. Choose options as per needed and connect.

Connect to Cluster0



Connecting with MongoDB Compass

I don't have MongoDB Compass installed

I have MongoDB Compass installed

1. Choose your version of Compass

1.38 or later ▼

See your Compass version in "About Compass"

2. Copy the connection string, then open MongoDB Compass

Use this connection string in your application

```
mongodb+srv://vika2k:<db_password>@cluster0.ra3ai.mongodb.net/
```



Replace **<db_password>** with the password for the **vika2k** user. Ensure any options are [URL encoded](#). You can edit your database user password in [Database Access](#).

Example:

```
mongodb+srv://vika2k:5j4tyZQSnucMiK1h@cluster0.b7mwy.mongodb.net/travel  
database?retryWrites=true&w=majority
```

5. Use the Connection String in the Application

1. Open your `.env` file in the backend directory:

```
Copy code  
nano .env
```

2. Add the connection string:

```
MONGO_URI="mongodb+srv://vika2k:5j4tyZQSnucMiK1h@cluster0.b7mwy.mongod  
b.net/traveldatabase"  
PORT=3000
```

3. Save the file and restart your backend server:

```
node index.js
```

Ensure your data given in frontend server are available here.

The screenshot displays the MongoDB Atlas web interface. The top navigation bar includes links for Gmail, Settings, google Meet, Yahoo Mail, github, Manage acc..., Hero Vired: LMS, AWS, ChatGPT, AI, AWS Certified cloud..., AWS - login, and Other favourites. The Atlas logo is on the left, followed by a dropdown menu showing 'Wipro' and 'Access Manager'. The main header shows 'ProjectHerovired' and 'Data Services' tabs. The left sidebar contains navigation options: Overview, DATABASE, Clusters, SERVICES, Atlas Search, Stream Processing, Triggers, Migration, Data Federation, SECURITY, Quickstart, Backup, Database Access, Network Access, and Advanced. The main content area is divided into tabs: Overview, Real Time, Metrics, Collections (selected), Atlas Search, Performance Advisor, Online Archive, and Cmd Line Tools. The 'Collections' tab shows a list of collections: sample_mflix, test, and traveldatabase1. The 'traveldatabase1' collection is selected, showing its details: Storage Size: 36KB, Logical Data Size: 125KB, Total Documents: 4, and Indexes Total Size: 36KB. The 'tripdetails' collection is highlighted. Below the collection name, there are tabs for Find, Indexes, Schema Anti-Patterns, Aggregation, and Search Indexes. A search bar is present with the text 'Type a query: { field: 'value' }'. The 'QUERY RESULTS: 1-4 OF 4' section shows a single document with the following fields: _id, tripName, startDateOfJourney, endDateOfJourney, nameOfHotels, placesVisited, totalCost, and tripTime.

Overview

Overview Real Time Metrics Collections Atlas Search Performance Advisor Online Archive Cmd Line Tools

DATABASE

Clusters

SERVICES

Atlas Search

Stream Processing

Triggers

Migration

Data Federation

SECURITY

Quickstart

Backup

Database Access

Network Access

Advanced

ProjectHerovired

Data Services

Charts

Access Manager

Billing

All Clusters

Get Help

M.

+ Create Database

Search Namespaces

TravelDatabase

tripdetails

sample_mflix

test

traveldatabase1

STORAGE SIZE: 36KB LOGICAL DATA SIZE: 125KB TOTAL DOCUMENTS: 4 INDEXES TOTAL SIZE: 36KB

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

Generate queries from natural language in Compass

INSERT DOCUMENT

Filter Type a query: { field: 'value' } Reset Apply Options

QUERY RESULTS: 1-4 OF 4

```
{
  "_id": ObjectId('6745cf3887db2979a4d129e8'),
  "tripName": "Dec Trip",
  "startDateOfJourney": "2024-12-07",
  "endDateOfJourney": "2024-12-06",
  "nameOfHotels": "Jolly",
  "placesVisited": "Bhadhrinath",
  "totalCost": 14300,
  "tripTime": "14300"
}
```