TravelMemory Application Deployment Using MERN Stack

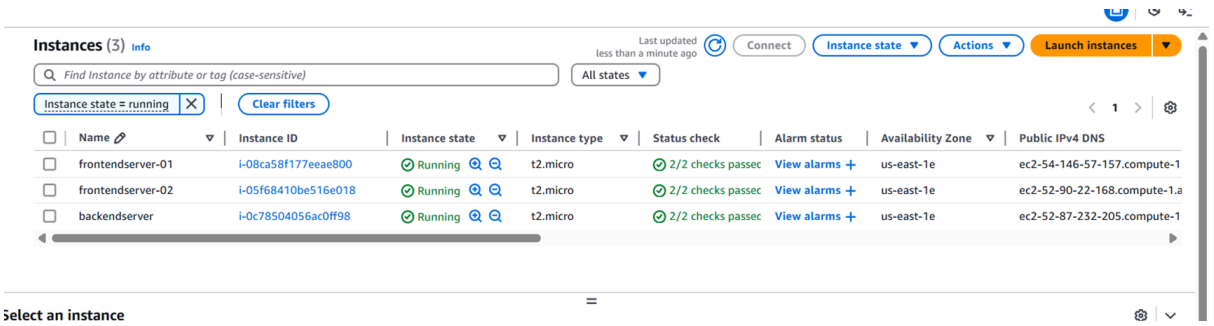
Objective;-

- Deploy the TravelMemory application using the MERN stack on AWS.
- Configure backend and frontend EC2 instances.
- Set up a secure, scalable, and high-performance architecture with a load balancer and Cloudflare integration.

Step 1: Create EC2 Instances

Create three EC2 instances on AWS with Ubuntu as the operating system:

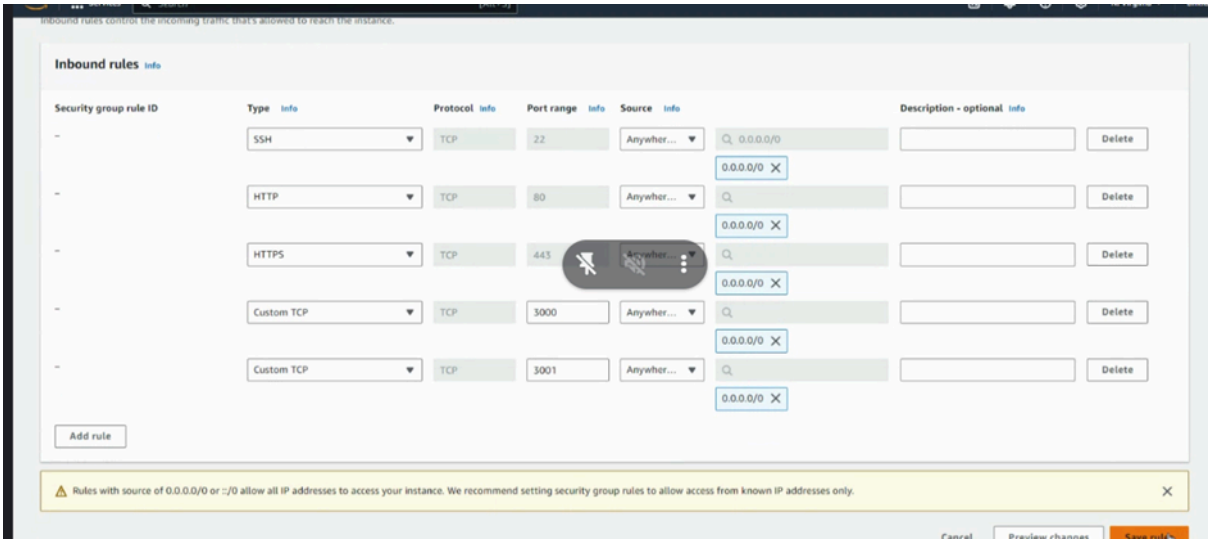
1. **Backend Server**: backendserver
2. **Frontend Servers**: frontendserver-01, frontendserver-02



Security Group Configuration:-

Configure the security group to allow the following ports:

- **SSH (Port 22):** For remote server management.
- **HTTP (Port 80):** For web traffic.
- **HTTPS (Port 443):** For secure web traffic.
- **Port 3000:** For the backend server.
- **Port 3001:** For frontend servers.



Step 2: Backend Server Setup

Install Necessary Tools

Run the following commands to install required software:

- sudo apt update -y
- sudo apt i install nginx
- sudo apt install git
- sudo apt install nodejs
- sudo apt install npm

Here are the commands to install and configure Nginx on your Ubuntu EC2 instance:

Step 1: Install Nginx

Run the following commands on the backend server to install and configure Nginx:

1. Update Package List: **sudo apt update**
2. Install Nginx: **sudo apt install -y nginx**
3. Start and Enable Nginx Service: **sudo systemctl start nginx**

   sudo systemctl enable nginx
4. Check Nginx Status: **sudo systemctl status nginx**

   Ensure it shows active (running).

Step 2: Configure Nginx

1. Edit the Default Nginx Configuration File:

sudo nano /etc/nginx/sites-available/default

2 Replace the File Content with the Following Configuration:

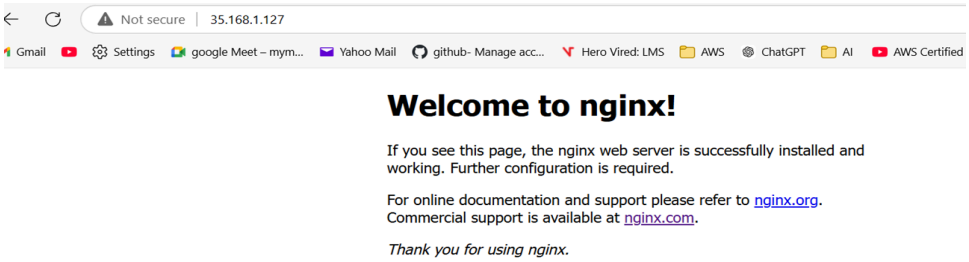
```
server {
        listen 80;
        server_name <your-ec2-public-ip>;(35.168.1.127)
        location / {
        proxy_pass http://127.0.0.1:3000;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
        }
}
```

Replace <your-ec2-public-ip> with the backend server's public IP address.

(35.168.1.127)

```
  GNU nano 7.2
server {
    listen 80;

    server_name 35.168.1.127;

    location / {
        proxy_pass http://127.0.0.1:3000;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
    }
}
```

3. **Test the Nginx Configuration:** **sudo nginx -t**
4. **Restart Nginx to Apply Changes :** **sudo systemctl restart nginx**
5. **Verify Nginx:** Open your backend EC2 public IP in a browser. You should see the Nginx default page.



Step 3: Proceed to Clone the Repository

After installing and configuring Nginx, you can now proceed with the next steps to clone the repository and set up the backend application.

1. **Clone the Backend Repository:** **sudo git clone** https://github.com/Kavi1312/TravelMemory.git

https://github.com/Kavi1312/TravelMemory

Gmail | Settings | google Meet – mym... | Yahoo Mail | github- Manage acc... | Hero Vired: LMS | AWS | ChatGPT | AI

Kavi1312 / TravelMemory

Code | Pull requests | Actions | Projects | Wiki | Security | Insights | Settings

TravelMemory Public

forked from UnpredictablePrashant/TravelMemory

Pin | Watch

main | 1 Branch | 0 Tags | Go to file | Add file | Code

This branch is 6 commits ahead of, 1 commit behind `UnpredictablePrashant/TravelMemory:main`.

Contribute | Sync fork

Kavi1312 Update .env — 922c5f9 · 2 days ago — 21 Commits

backend	Update .env	2 days ago
frontend	Update url.js	3 days ago
.gitignore	Initial commit	last year
LICENSE	Initial commit	last year
README.md	Update README.md	6 months ago
azure-pipelines.yml	Set up CI with Azure Pipelines	9 months ago

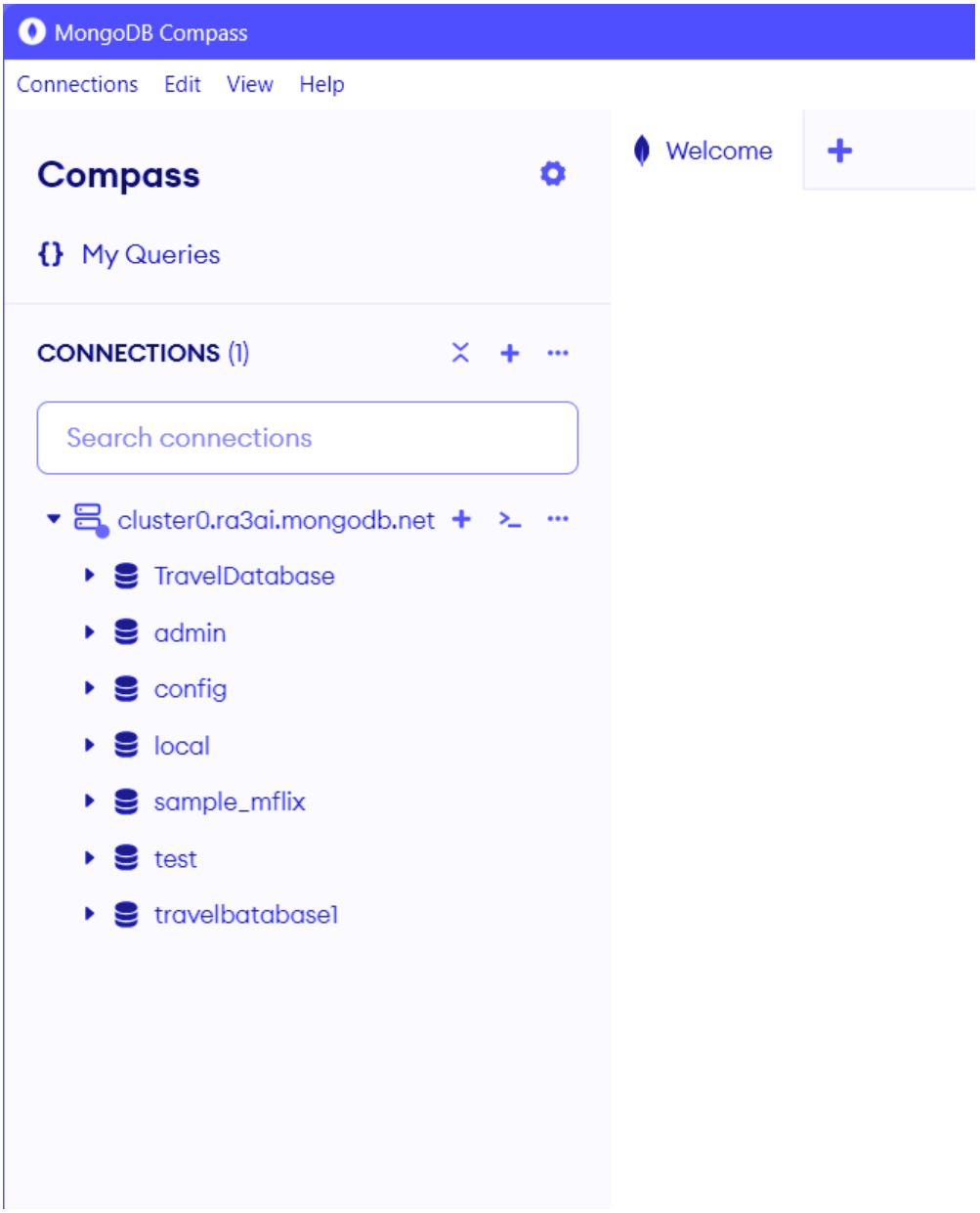
README | License

2. **Navigate to folder :** **cd TravelMemory/backend**
3. **Install Required Packages for the Backend:** **sudo apt install -y nodejs npm**
   sudo npm install
4. **Configure the Backend Environment Variables:** **sudo nano .env**

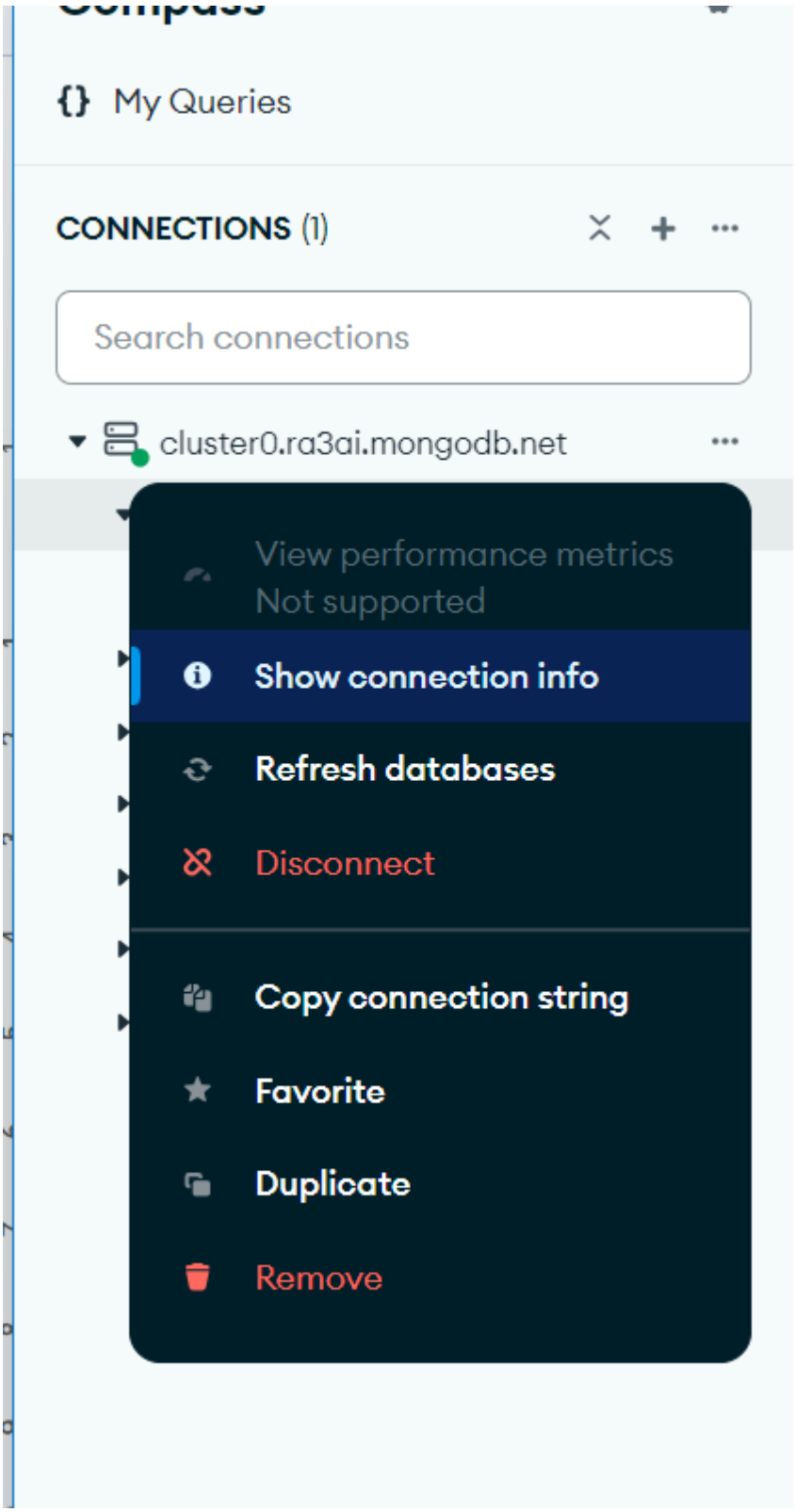
   Add the following content: makefile

```
PORT=3000

MONGO_URI="mongodb+srv://vika2k:5j4tyZQSnucMiK1h@cluster0.b7mwy.mongodb.net/TravelDatabase"
```

(your MonogoDb username password and connecting string details and your database name)



- *Copy connection string details from MongoDb Compass ;*
- *Click cluster name 3 dots and select connection details.*
- *Use this Link to create cluster by login :-* [Project Overview | Cloud: MongoDB Cloud]


Compass
My Queries
CONNECTIONS (1)
Search connections
cluster0.ra3ai.mongodb.net
View performance metrics
Not supported
Show connection info
Refresh databases
Disconnect
Copy connection string
Favorite
Duplicate
Remove

Start the Backend Application (Optional for Testing): sudo node index.js

Modify index.js to Replace localhost with Backend IP Address: sudo nano index.js

```
  GNU nano 7.2                                                    index.js
const express = require('express')
const cors = require('cors')
require('dotenv').config()

const app = express()
PORT = process.env.PORT
const conn = require('./conn')
app.use(express.json())
app.use(cors())

const tripRoutes = require('./routes/trip.routes')

app.use('/trip', tripRoutes) // http://localhost:3001/trip --> POST/GET/GET by ID

app.get('/hello', (req,res)=>{
    res.send('Hello World!')
})

app.listen(PORT, ()=>{
    console.log(`Server started at http://localhost:${PORT}`)
})
```

```
^C
ubuntu@ip-172-31-60-166:~/TravelMemory/backend$ sudo node index.js
Server started at http://35.168.1.127:3000
```

i-064f5a4b02aa9892a (backendserver)

PublicIPs: 35.168.1.127 PrivateIPs: 172.31.60.166

Search [Alt+S]

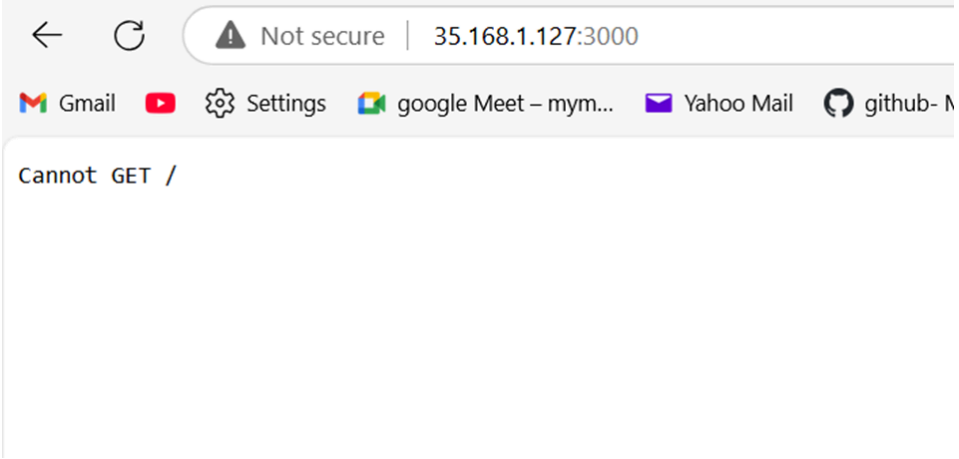
Certificate Manager | S3 | Route 53 | EC2 | CloudFront | RDS | VPC | IAM | Lambda | API Gateway

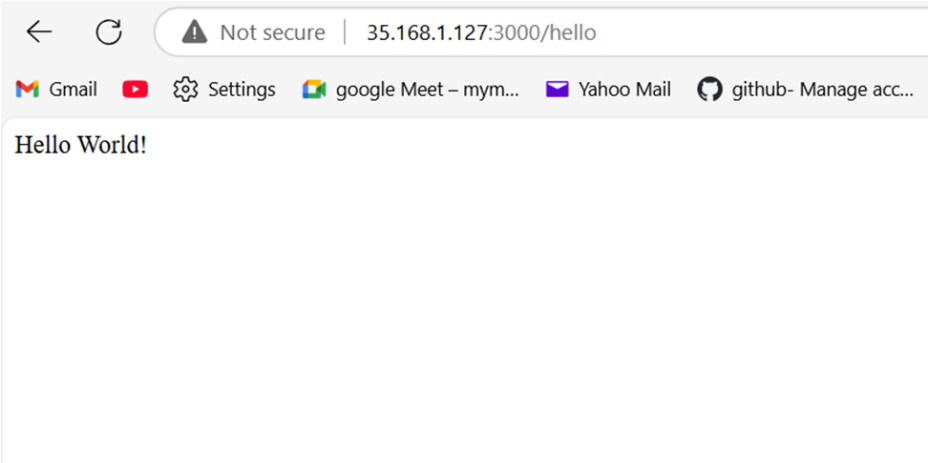
```
ubuntu@ip-172-31-60-166:~/TravelMemory/backend$ sudo npm install

up to date, audited 118 packages in 2s

13 packages are looking for funding
  run `npm fund` for details

13 vulnerabilities (3 low, 1 moderate, 8 high, 1 critical)

To address issues that do not require attention, run:
  npm audit fix

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
ubuntu@ip-172-31-60-166:~/TravelMemory/backend$ sudo node index.js
Server started at http://35.168.1.127:3000
{
  tripName: 'new trip',
  startDateOfJourney: '2024-11-30',
  endDateOfJourney: '2024-12-14',
  nameOfHotels: 'first',
  placesVisited: 'kashmir',
  totalCost: '12001',
  experience: 'nice memory',
  image: '',
  tripType: 'leisure',
  featured: true,
  shortDescription: 'happy time'
}
```

i-064f5a4b02aa9892a (backendserver)

PublicIPs: 35.168.1.127 PrivateIPs: 172.31.60.166

Update the host with the backend EC2 public IP. (check with or without port and / hello)





Ensured Backend is working fine