

MERN Project

Create EC2 instance for Frontend and Backend Servers

Frontend Server: tilak-travelapp_frontend

Backend Server: tilak-travelapp_backend

| Instances (3) Info | | | | |
|--|--------------------------|---------------------|----------------------|---------------|
| <input type="text" value="Find instance by attribute or tag (case-sensitive)"/> | | | | |
| Key name = tilak-keypair <input type="button" value="X"/> <input type="button" value="Clear filters"/> | | | | |
| <input type="checkbox"/> | Name | Instance ID | Instance state | Instance type |
| <input type="checkbox"/> | tilak-travelapp_frontend | i-00858c238fa0b2f7b | Running | t2.micro |
| <input type="checkbox"/> | tilak-travelapp_backend | i-05ed8b286822d797f | Running | t2.micro |

Install updates on Frontend and Backend servers and install the required applications

#apt update

Install Nginx Application

#apt install nginx -y

Install Node JS and NPM

Followed the [link](#) to install NodeJS

curl -s https://deb.nodesource.com/setup_18.x | sudo bash

sudo apt install nodejs -y

node -v

Please find the below screenshot for the installation

The image shows two terminal windows side-by-side. The left window is titled 'root@frontend:/opt' and shows the output of 'apt update' and 'apt install nginx -y'. The right window is titled 'root@backend:/opt' and shows the output of 'apt update' and 'curl -s https://deb.nodesource.com/setup_18.x | sudo bash'. Both windows show successful installation of updates and applications.

```
root@frontend:/opt# apt update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [900 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [145 kB]
Get:7 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 c-n-f Metadata [528 B]
Get:8 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [41.6 kB]
Get:9 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 c-n-f Metadata [472 B]

root@backend:/opt# apt update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Hit:4 https://deb.nodesource.com/node_18.x jammy InRelease
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1012 kB]
Get:7 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [227 kB]
Get:8 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [15.6 kB]
Get:9 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [900 kB]
Get:10 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [145 kB]
```

```

root@frontend:/opt/TravelMemory/frontend# curl -s https://deb.nodesource.com/setup_18.x | sudo bash
=====
                        SCRIPT DEPRECATION WARNING
=====

This script, located at https://deb.nodesource.com/setup_X, used to
install Node.js is deprecated now and will eventually be made inactive.

Please visit the NodeSource distributions Github and follow the
instructions to migrate your repo.
https://github.com/nodesource/distributions

The NodeSource Node.js Linux distributions GitHub repository contains
information about which versions of Node.js and which Linux distributions
are supported and how to install it.
https://github.com/nodesource/distributions

                        SCRIPT DEPRECATION WARNING
=====
=====
=====
TO AVOID THIS WAIT MIGRATE THE SCRIPT

```

Till now we have gone with the required application installation now will go with the Cloning of the code from Github.

Now will make few manual configurations on the backend server.

Create a new .env file in the Backend application directory

MONGO_URI="mongodb+srv://tilakpechetti:xxxxxx%4012@cluster0.nitiwty.mongodb.net/"

PORT="3000"

Backend Server configuration

```

root@backend: /opt/TravelMemory X + v
MONGO_URI='mongodb+srv://tilakpechetti:xxxxxx%4012@cluster0.nitiwty.mongodb.net/travel'
PORT='3000'

```

Simple configuration right!!! That's it no other changes are required.

Now start the application using below commands

#npm install

#node index.js (which will start the application) Please see the screenshot for reference.

```
root@backend: /opt/TravelMemory/ backend# npm install

up to date, audited 118 packages in 2s

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

5 vulnerabilities (4 moderate, 1 critical)

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

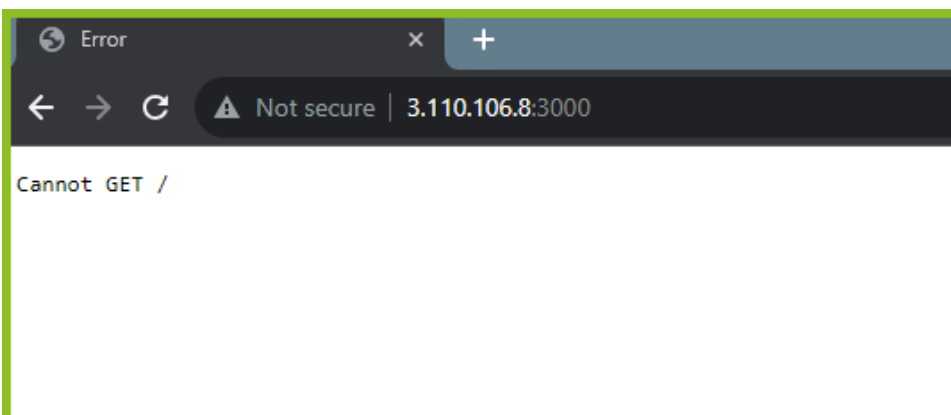
To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
root@backend: /opt/TravelMemory/ backend# |
```

```
root@backend: /opt/TravelMemory/ backend# node index.js
Server started at http://localhost:3000
```

To Test if the application is running or not

Use the Server IP or else custom domain name or else an API URL.



Configure Nginx to bypass proxy on backend server.

Below is the configuration steps that needs to be updated

```
#sudo unlink /etc/nginx/sites-enabled/default
```

```
#cd /etc/nginx/sites-available/
```

```
#sudo nano custom_server.conf
```

```
server { listen 80;
```

```
location / {
```

```
proxy_pass http://my_server;
```

```
}}
```

```
#sudo service nginx configtest
```

```
#sudo service nginx restart
```

Now we will configure the Frontend server

After the required applications are installed follow the below steps

Clone the code

```
# sudo git clone https://github.com/UnpredictablePrashant/TravelMemory
```

```
## Installing the NodeSource Node.js 18.x repo ...

## Populating apt-get cache ...

+ apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1012 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [227 kB]
Get:7 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [15.6 kB]
Get:8 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [901 kB]
Get:9 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [145 kB]
Get:10 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [987 kB]
Get:11 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [215 kB]
Get:12 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [21.9 kB]
Get:13 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [41.6 kB]
Get:14 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [24.3 kB]
Get:15 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [640 B]
Get:16 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [785 kB]
```

Check the node version is latest, I have v18.17.1

```
root@frontend: /opt/TravelM  X + v
root@frontend:/opt/TravelMemory/frontend# node --version
v18.17.1
root@frontend:/opt/TravelMemory/frontend#
```

```
root@frontend: /opt/TravelM X + v
root@frontend:/opt/TravelMemory/frontend# node --version
v18.17.1
root@frontend:/opt/TravelMemory/frontend# npm --version
9.6.7
root@frontend:/opt/TravelMemory/frontend#
```

Next step is to update the backend URL in the frontend server on below path

My Application path is:

/opt/TravelMoemory/frontend/src/url.js

```
root@frontend: /opt/TravelM X + v
root@frontend:/opt/TravelMemory/frontend# ll
total 740
drwxr-xr-x  5 root root  4096 Sep  3 07:28 ./
drwxr-xr-x  5 root root  4096 Sep  2 06:20 ../
-rw-r--r--  1 root root   310 Sep  2 06:20 .gitignore
-rw-r--r--  1 root root  3359 Sep  2 06:20 README.md
drwxr-xr-x 835 root root 36864 Sep 26 16:21 node_modules/
-rw-r--r--  1 root root 685471 Sep 26 16:21 package-lock.json
-rw-r--r--  1 root root   973 Sep  2 17:28 package.json
drwxr-xr-x  2 root root  4096 Sep  2 06:20 public/
drwxr-xr-x  3 root root  4096 Sep 28 12:08 src/
root@frontend:/opt/TravelMemory/frontend#
```

```
root@frontend: /opt/TravelM X + v
export const baseUrl = "http://3.110.106.8"
~
~
~
```

After the configuration is done,

#npm install

Then

#npm start

```
root@frontend: /opt/TravelM X + v
root@frontend:/opt/TravelMemory/frontend/src# npm install

up to date, audited 1502 packages in 5s

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

9 vulnerabilities (3 moderate, 6 high)

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.
root@frontend:/opt/TravelMemory/frontend/src#
```

Command: npm start

```
root@frontend: /opt/TravelM X + v
Compiled successfully!

You can now view frontend in the browser.

Local:            http://localhost:80
On Your Network:  http://172.31.34.35:80

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

Which will allow you to launch and you should be able to connect to frontend server

Test:

Curl -V localhost

It should give you the result

Once are you able to access the URL, please the record and it should be visible in the database.

MongoDB Compass - tilak/travel.tripdetails

Connect Edit View Collection Help

tilak Documents travel.tripdetails

My Queries

Databases

Search

- admin
- local
- megashop
- sample_airbnb
- sample_analytics
- sample_geospatial
- sample_guides
- sample_mflix
- sample_restaurants
- sample_supplies
- sample_training
- sample_weatherdata
- travel

tripdetails

travel.tripdetails

Documents Aggregations Schema Indexes Validation

Filter Type a query: { field: 'value' }

ADD DATA EXPORT DATA

```
createdAt: 2023-09-26T16:21:03.306+00:00
__v: 0

_id: ObjectId('651576c55227fc9241e99dc0')
tripName: "krishna"
startDateOfJourney: "2023-09-29"
endDateOfJourney: "2023-09-30"
nameOfHotels: "Assignment Hotel"
placesVisited: "Delhi"
totalCost: 10000
tripType: "leisure"
experience: "Assignment"
image: "http://13.127.63.68/"
shortDescription: "Assignment"
featured: false
createdAt: 2023-09-28T12:35:16.922+00:00
__v: 0
```

Error React App React App

Not secure | 13.127.63.68/addexperience

Travel Memory Add Experience

Trip Name

krishna

Trip Date

29-09-2023



30-09-2023

Name of Hotels

Assignment Hotel

Trip Type

Leisure



Total Cost

10000

Places Visited

Delhi

Featured Trip?

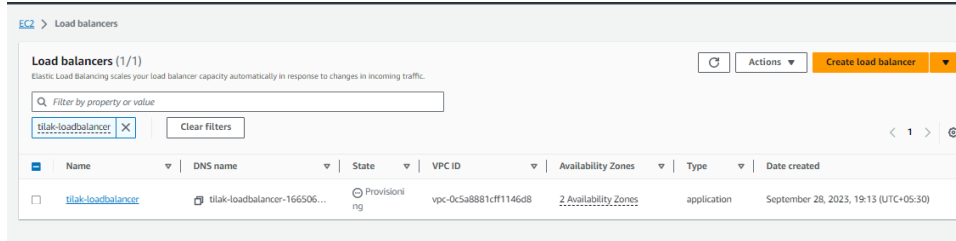
☐ True

☒ False

Configuring Load-Balancer

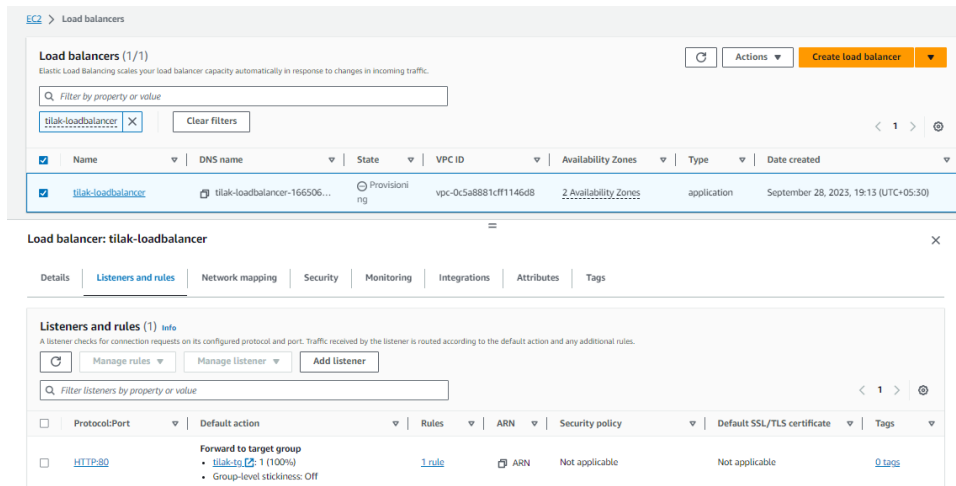
In AWS Create a New application loadbalancer and create a listener port no:80 and attach the target group as shown below.

Make sure the Target group is **healthy**



| | Name | DNS name | State | VPC ID | Availability Zones | Type | Date created |
|-------------------------------------|------------------------------------|------------------------------|--------------|----------------------|----------------------|-------------|---------------------------------------|
| <input checked="" type="checkbox"/> | tilak-loadbalancer | tilak-loadbalancer-166506... | Provisioning | vpc-0c5a8881cf1146d8 | 2 Availability Zones | application | September 28, 2023, 19:13 (UTC+05:30) |

Attached target group



| | Protocol/Port | Default action | Rules | ARN | Security policy | Default SSL/TLS certificate | Tags |
|-------------------------------------|-------------------------|---|------------------------|-----|-----------------|-----------------------------|------------------------|
| <input checked="" type="checkbox"/> | HTTP:80 | Forward to target group <ul style="list-style-type: none">tilak-tg (100%)Group-level stickiness: Off | 1 rule | ARN | Not applicable | Not applicable | 0 tags |

Target group is **Healthy**

Below are the details of the load balancer configured as shown

EC2 > Target groups > tilak-tg

tilak-tg Actions

Details
 arn:aws:elasticloadbalancing:ap-south-1:295397158094:targetgroup/tilak-tg/af4842e7fec32685

| | | | |
|-------------------------|----------------------------------|---------------------------|-----------------------------|
| Target type Instance | Protocol : Port HTTP: 80 | Protocol version HTTP1 | VPC vpc-0c5a8881cf114648 |
| IP address type IPv4 | Load balancer None associated | | |

| | | | | | |
|---------------|---------|-----------|--------|---------|----------|
| Total targets | Healthy | Unhealthy | Unused | Initial | Draining |
| 2 | 1 | 0 | 0 | 0 | 1 |

► **Distribution of targets by Availability Zone (AZ)**
 Select values in this table to see corresponding filters applied to the Registered targets table below.

Targets | Monitoring | Health checks | Attributes | Tags

Registered targets (1/2) Refresh Deregister Register targets

Filter resources by property or value

| Instance ID | Name | Port | Zone | Health status | Health status details |
|---------------------|--------------------------|------|-------------|---------------|-----------------------|
| i-00858c238fa0b277b | tilak-travelapp_frontend | 80 | ap-south-1a | Healthy | |

Load_Balancer DNS Name:

tilak-loadbalancer-1665065052.ap-south-1.elb.amazonaws.com

Opened website with Load Balancer URL

← → ↻ ⚠ Not secure | tilak-loadbalancer-1665065052.ap-south-1.elb.amazonaws.com

Travel Memory Add Experience

Featured

tilak
business
hello

More Details

Featured

tilak
business
hello

More Details

Updating the Load balancer DNS Name in Cloud Flair as CNAME record

CLOUDFLARE Add site Support English Profile

← tilakpechetti.co.in Add filter Search Add record

| Type | Name | Content | Proxy status | TTL | Actions |
|-------|---------------------|---------------------------------|--------------|------|---------|
| A | tilakpechetti.co.in | 34.102.136.180 | Proxied | Auto | Edit |
| A | travel | 13.127.63.68 | Proxied | Auto | Edit |
| CNAME | _domainconnect | _domainconnect.gd.domaincont... | Proxied | Auto | Edit |

Cloudflare

tilakpechetti.co.in

Overview

Analytics & Logs

DNS

Records

Settings

Email

SSL/TLS

Security

Access

Speed

Caching

Search DNS Records

Add filter

Search

Add record

| Type | Name | Content | Proxy status | TTL | Actions |
|-------|---------------------|---|--------------|------|---------|
| A | tilakpechetti.co.in | 34.102.136.180 | Proxied | Auto | Edit |
| A | travel | 13.232.160.175 | Proxied | Auto | Edit |
| CNAME | _domainconnect | _domainconnect.gd.domaincont... | Proxied | Auto | Edit |
| CNAME | email | email.secureserver.net | Proxied | Auto | Edit |
| CNAME | travel-api | tilak-mern-project-766167796.a... | Proxied | Auto | Edit |
| Type | Name (required) | Target (required) | Proxy status | TTL | |
| CNAME | travel-api | tilak-loadbalancer-1665065052.ap-south- | Proxied | Auto | |
| | Use @ for root | | | | |

Record Attributes

Documentation

The information provided here will not impact DNS record resolution and is only meant for your reference.

Custom domain name for URL: Travel-api.tilakpechetti.co.in

← → ↺

Not secure | travel-api.tilakpechetti.co.in

Travel Memory

Add Experience

Featured

tilak

business

hello

More Details

Done