

Domain – <http://mernstack.danielb3.shop>

## Create an EC2 Instance

The screenshot shows the AWS Management Console interface for an EC2 instance. The left sidebar contains navigation links for EC2 Dashboard, EC2 Global View, Events, Console-to-Code, and various EC2 services like Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, and Network & Security. The main content area displays the 'Instance summary for i-084a61213e8dd6df0 (DanielMern)'. The summary includes fields for Instance ID, IPv6 address, Hostname type, Answer private resource DNS name, Auto-assigned IP address, IAM Role, IMDSv2, Public IPv4 address, Instance state, Private IP DNS name, Instance type, VPC ID, Subnet ID, Private IPv4 addresses, Public IPv4 DNS, Elastic IP addresses, and AWS Compute Optimizer finding. The instance is currently in a 'Pending' state. Below the summary, there are tabs for Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags.

Field	Value
Instance ID	i-084a61213e8dd6df0 (DanielMern)
IPv6 address	-
Hostname type	IP name: ip-10-2-10-132.ec2.internal
Answer private resource DNS name	-
Auto-assigned IP address	54.89.98.9 [Public IP]
IAM Role	-
IMDSv2	Required
Public IPv4 address	54.89.98.9 [open address]
Instance state	Pending
Private IP DNS name (IPv4 only)	ip-10-2-10-132.ec2.internal
Instance type	t2.micro
VPC ID	vpc-08eebeeb425d3cd37
Subnet ID	subnet-070f6751946cf9645
Private IPv4 addresses	10.2.10.132
Public IPv4 DNS	-
Elastic IP addresses	-
AWS Compute Optimizer finding	User: amaws:iam::295397358094:user/Daniel_Gonsalves is not authorized to perform: compute-optimizer:GetEnrollmentStatus on resource: * because no identity-based policy allows the compute-optimizer:GetEnrollmentStatus action [Retry]
Auto Scaling Group name	-

## Attach Elastic IP for Static IP hosting

The screenshot shows the AWS Management Console interface for an Elastic IP address. The left sidebar contains navigation links for EC2 Dashboard, EC2 Global View, Events, Console-to-Code, and various EC2 services like Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, and Network & Security. The main content area displays the 'Elastic IP address summary for 18.213.249.54'. The summary includes fields for Allocated IPv4 address, Type, Allocation ID, Reverse DNS record, Association ID, Scope, Associated instance ID, Private IP address, Network interface ID, Network interface owner account ID, Public DNS, NAT Gateway ID, Address pool, and Network Border Group. The IP address is currently associated with an instance. Below the summary, there are tabs for Summary, Tags, and Actions. The 'Tags' tab shows no tags associated with this resource.

Field	Value
Allocated IPv4 address	18.213.249.54
Type	Public IP
Allocation ID	eipalloc-0e82f64771a1d5b6c
Reverse DNS record	-
Association ID	-
Scope	VPC
Associated instance ID	-
Private IP address	-
Network interface ID	-
Network interface owner account ID	-
Public DNS	-
NAT Gateway ID	-
Address pool	Amazon
Network Border Group	us-east-1

## Associate Elastic IP address

Choose the instance or network interface to associate to this Elastic IP address (18.213.249.54)

Elastic IP address: 18.213.249.54

### Resource type

Choose the type of resource with which to associate the Elastic IP address.

- ☒ Instance  
☐ Network interface

**⚠** If you associate an Elastic IP address with an instance that already has an Elastic IP address associated, the previously associated Elastic IP address will be disassociated, but the address will still be allocated to your account. [Learn more](#)

If no private IP address is specified, the Elastic IP address will be associated with the primary private IP address.

### Instance

### Private IP address

The private IP address with which to associate the Elastic IP address.

### Reassociation

Specify whether the Elastic IP address can be reassociated with a different resource if it already associated with a resource.

- ☐ Allow this Elastic IP address to be reassociated

Cancel

Associate

Using CloudFlare with a domain address- danielb3.shop, we shall connect the elastic ip to the domain

spoofing. [New Alert](#)

---

**DNS management for danielb3.shop** [Import and Export](#) [Dashboard Display Settings](#)

Review, add, and edit DNS records. Edits will go into effect once saved.

Search DNS Records

[Add filter](#)

Type	Name	Content	Proxy status	TTL	Actions
A	danielb3.shop	18.213.249.54	Proxied	Auto	<a href="#">Edit</a>

---

**Cloudflare Nameservers**

To use Cloudflare, ensure your authoritative DNS servers, or nameservers have been changed. These are your assigned Cloudflare nameservers.

## Server Configuration

1) Upgrade and Update Ubuntu

2) Install Node.js

```
sudo apt-get install nodejs
```

```
sudo apt install npm
```

### 3) Clone Git

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

### 4) Install Nginx and Initiate

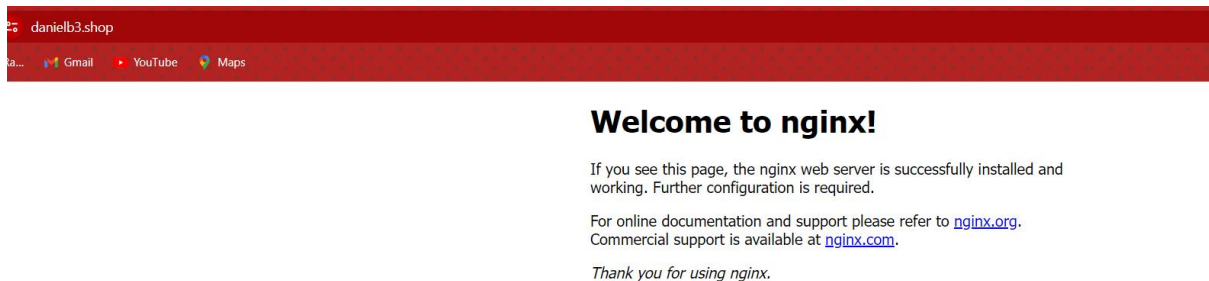
```
sudo apt install nginx
```

```
sudo nginx -t
```

```
sudo systemctl restart nginx
```

```
sudo systemctl status nginx
```

Verify if nginx is initiated



### 5) Install React

```
sudo npm install react
```

### 6) Install Dependencies

Navigate to backend and frontend directories respectively and run the command

```
sudo npm install
```

### 7) Fix Dependencies

Incase of certain conflicting dependencies run the command

```
npm audit fix --force
```

### 8) Navigate to the backend directory and initiate a .env file

In the .env file type:

```
MONGO_URI= mongodb+srv://username:password@travelmemorydb.uid.mongodb.net
```

### 9) Run the backend using pm2

```
pm2 start index.js
```

### 10) Update TravelMemory/frontend/src/url.js to include port number of the env file

### 11) Start the nginx configuration, configure sites enabled as :

```

server {
    listen 3000;
    server_name 54.172.244.145;

    location / {
        proxy_pass http://localhost: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;
    }
}

```

## DEPLOYMENT CONFIGURATION

```

ubuntu@ip-10-2-8-240:~/TravelMemory/frontend/src$ nano url.js
ubuntu@ip-10-2-8-240:~/TravelMemory/frontend/src$ ubuntu@ip-10-2-8-240:~/TravelMemory/frontend/src$ cd ..
ubuntu@ip-10-2-8-240:~/TravelMemory/frontend$ cd ..
ubuntu@ip-10-2-8-240:~/TravelMemory$ cd backend
ubuntu@ip-10-2-8-240:~/TravelMemory/backend$ node index.js
Server started at http://localhost:3000
^C
ubuntu@ip-10-2-8-240:~/TravelMemory/backend$ pm2 start index.js
[PM2] Applying action restartProcessId on app [index](ids: [ 0 ])
[PM2] [index](0) 0
[PM2] Process successfully started

```

id	name	namespace	version	mode	pid	uptime	▯	status	cpu	mem	user	watching
0	index	default	1.0.0	fork	2281	0s	46	online	0%	7.0mb	ubuntu	disabled

```

ubuntu@ip-10-2-8-240:~/TravelMemory/backend$ pm2 list

```

id	name	namespace	version	mode	pid	uptime	▯	status	cpu	mem	user	watching
0	index	default	1.0.0	fork	2281	5s	46	online	0%	74.8mb	ubuntu	disabled

```

ubuntu@ip-10-2-8-240:~/TravelMemory/backend$ pm2 list

```

id	name	namespace	version	mode	pid	uptime	▯	status	cpu	mem	user	watching
0	index	default	1.0.0	fork	2281	12s	46	online	0%	68.6mb	ubuntu	disabled

## Start the Front End to test it

```

ubuntu@ip-10-2-8-240:~/TravelMemory$ cd frontend
ubuntu@ip-10-2-8-240:~/TravelMemory/frontend$ npm start

> frontend@0.1.0 start
> react-scripts start

```

```
ubuntu@ip-10-2-8-240: ~/travelMemory/frontend
Compiled successfully!Compiled successfully!

You can now view frontend in the browser.

Local:          http://localhost:3002
On Your Network: http://10.2.8.240:3002

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

webpack compiled successfully
```

Deploy React after testing.

