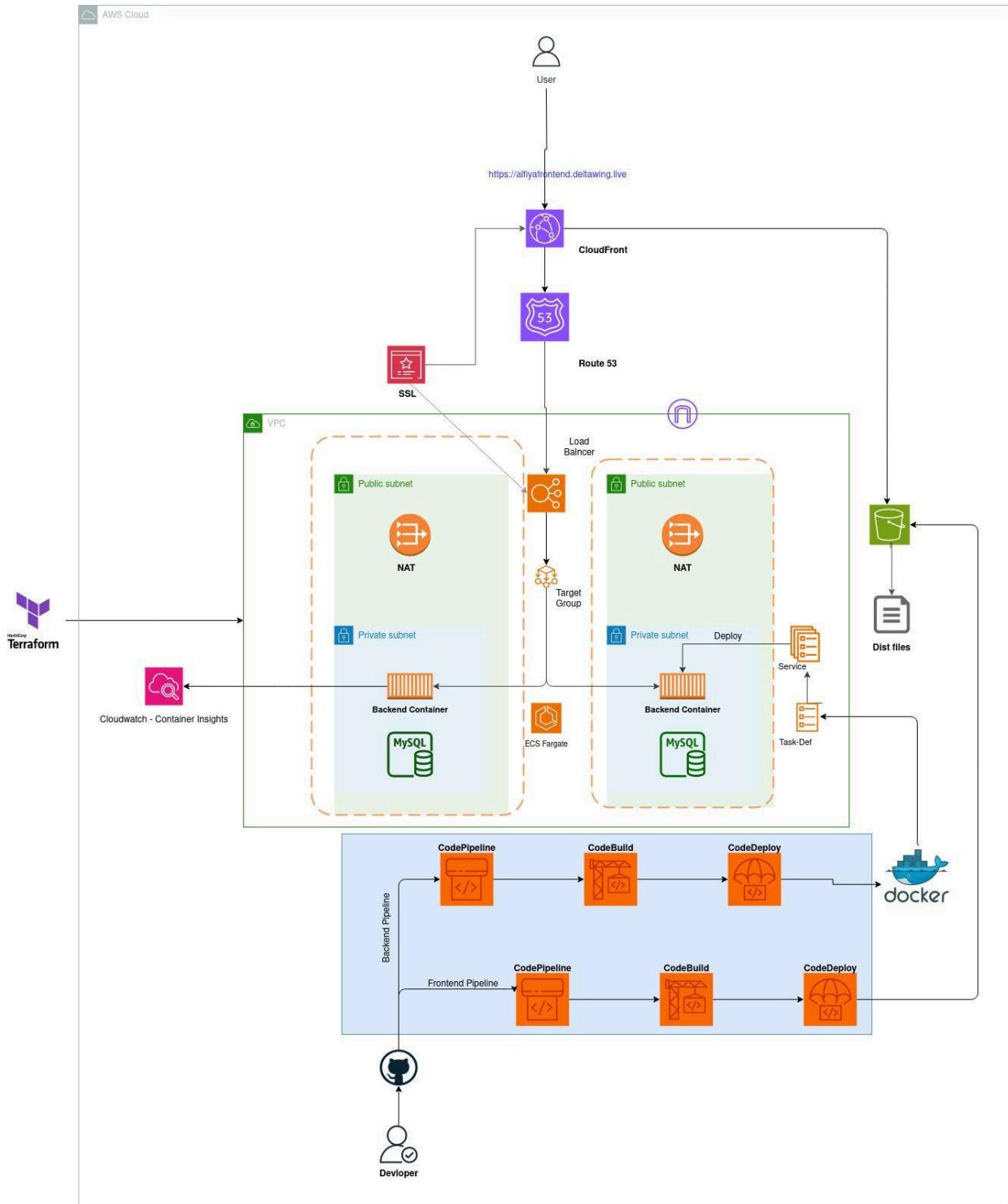


AWS MINI PROJECT-1

Problem Statement

1. Use of terraform for creation of infrastructure and Deploy an CRUD application on AWS
2. Front end - S3.
3. Back end - ECS.
4. Database - RDS.
5. Perform cost optimization
6. Ensure best security practices
7. Create self explanatory documentation
8. Enable monitoring of infrastructure
9. CI/CD using jenkins/AWS Code pipeline(SCM, Github)(terraform)
10. Expose domain along with SSL certification
11. Architecture diagram



Steps:

1. Created infrastructure through terraform
Link: <https://github.com/Alfiya-git/3-tier-infra.git>

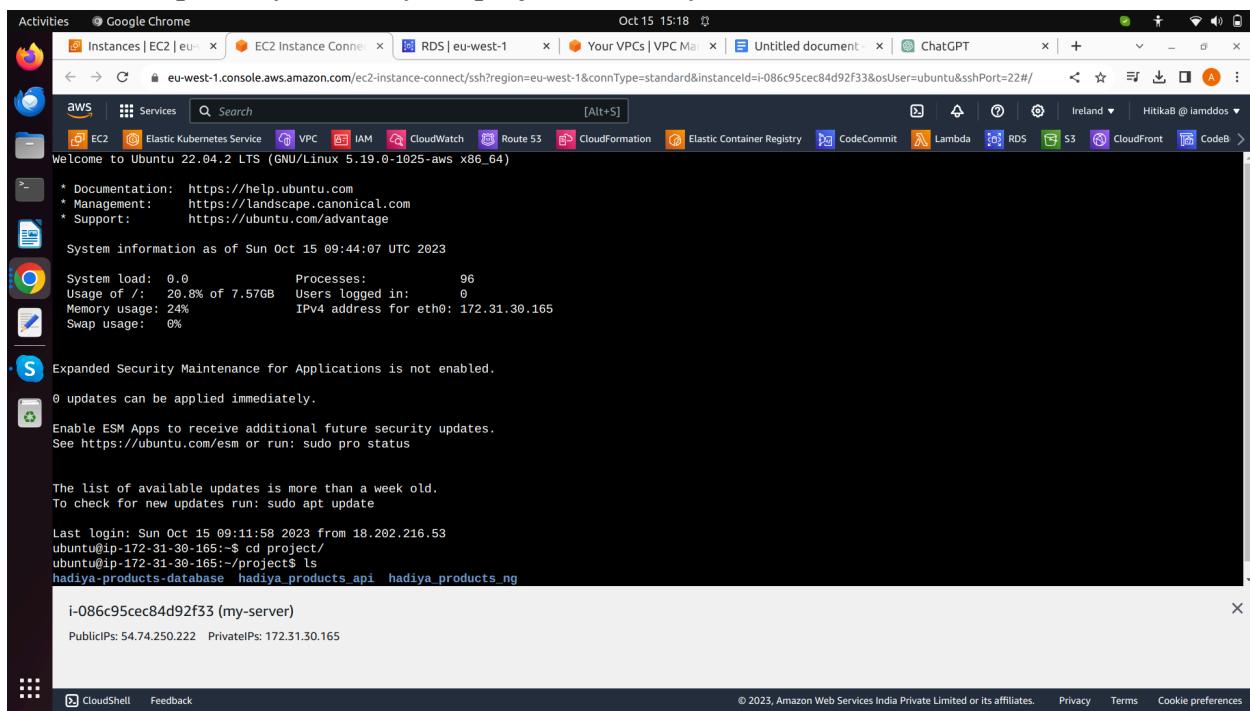
2. Firstly , created VPC and its components then RDS in the same VPC in private subnet
3. Open Your Terminal and Open your application Navigate to Your Project Directory, Use the cd command to navigate to the directory where you want to clone the GitHub repository.

Backend : <https://github.com/Alfiya-git/backend.git>

Frontend : <https://github.com/Alfiya-git/frontend.git>

Database : <https://github.com/Alfiya-git/hadiya-products-database.git>

4. Clone the GitHub Repository: Use the git clone command to clone the repository from GitHub. Replace repository_url with the URL of the GitHub repository you want to clone: git clone repository_url
5. Verify the Cloning Process , Once the cloning process is complete, you should see the repository files in your project directory.



```
Oct 15 15:18 ⓘ
Activities Google Chrome Oct 15 15:18 ⓘ
Instances | EC2 | eu-west-1 × EC2 Instance Connect × RDS | eu-west-1 × Your VPCs | VPC M × Untitled document × ChatGPT × + ⓘ
eu-west-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=eu-west-1&connType=standard&instanceId=i-086c95cec84d92f33&osUser=ubuntu&sshPort=22# ⓘ
aws ⓘ Services ⓘ Search ⓘ [Alt+S]
EC2 ⓘ Elastic Kubernetes Service ⓘ VPC ⓘ IAM ⓘ CloudWatch ⓘ Route 53 ⓘ CloudFormation ⓘ Elastic Container Registry ⓘ CodeCommit ⓘ Lambda ⓘ RDS ⓘ S3 ⓘ CloudFront ⓘ CodeBuild ⓘ
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.19.0-1025-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

System information as of Sun Oct 15 09:44:07 UTC 2023

System load: 0.0 Processes: 96
Usage of /: 20.8% of 7.57GB Users logged in: 0
Memory usage: 24% IPv4 address for eth0: 172.31.30.165
Swap usage: 0%

* Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Sun Oct 15 09:11:58 2023 from 18.202.216.53
ubuntu@ip-172-31-30-165:~$ cd project/
ubuntu@ip-172-31-30-165:~/project$ ls
hadiya-products-database hadiya_products_api hadiya_products_ng

i-086c95cec84d92f33 (my-server)
PublicIPs: 54.74.250.222 PrivateIPs: 172.31.30.165

CloudShell Feedback ⓘ © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences
```

6. Go into the backend hadiya_products_api and run command ll there must be .env.development file . *In a Node.js project, the .env.development file is typically used to store environment variables specific to the development environment. Environment variables are values that can be accessed within your Node.js application and are used to configure various aspects of the application. These*

variables can include database connection strings, API keys, feature toggles, and more.

7. Run command cp .env.development .env and edit in .env using nano editor
8. Testing Locally : Create a database ‘hadiya’ and a user ‘admin’ with password ‘password’ and change in the .env file for testing and enter command mysql -u admin -p -D hadiya

```
ubuntu@ip-172-31-30-165:~$ sudo mysql
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 17
Server version: 8.0.34-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input

mysql> CREATE USER 'admin'@'localhost' IDENTIFIED BY '12345678';
Query OK, 0 rows affected (0.02 sec)

mysql> GRANT ALL PRIVILEGES ON *.* TO 'admin'@'localhost';
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> CREATE DATABASE hadiya;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+-----+
| Database      |
+-----+
| hadiya        |
| information_schema |
| mysql          |
| performance_schema |
| sys            |
+-----+
```

```

ubuntu@ip-172-31-30-165:~$ mysql -u admin -p -D hadiya
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 18
Server version: 8.0.34-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its

```

9. Now make a rds on console with same user password and database and enter its endpoint into .env file and test it

10. Restore the tables and its values into database using command

```

ubuntu@ip-10-0-2-213:~/project/hadiya-products-database/sql$ mysql -h test.cd9x5jnkt7u0.eu-central-1.rds.amazonaws.com -u admin -p -P 3306 -D hadiya < V1_Create_Pro
ducts_Table.sql
Enter password:

```

```

ubuntu@ip-10-0-2-213:~/project/hadiya-products-database/sql$ mysql -h test.cd9x5jnkt7u0.eu-central-1.rds.amazonaws.com -u admin -p -P 3306 -D hadiya < V2_Insert_Dum
my_Products.sql
Enter password:

```

11. Cross check by logging in the database if there are database tables named Products and have values.

```

ubuntu@ip-10-0-2-213:~/project/hadiya-products-database/sql$ mysql -u admin -h test.cd9x5jnkt7u0.eu-central-1.rds.amazonaws.com -p -D hadiya
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MySQL connection id is 20
Server version: 8.0.33 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [hadiya]> select * from hadiya;
ERROR 1146 (42S02): Table 'hadiya.hadiya' doesn't exist
MySQL [hadiya]> select * from Products;
+----+-----+-----+-----+-----+
| id | uuid           | name          |          | price    | currency | imageURL
|----+-----+-----+-----+-----+
+----+-----+-----+-----+-----+
| 1 | c1dc5702-58aa-4c17-abea-d5d54641362c | Dell Inspiron 7415 AMD R5 5500U 14 inches(35cm) FHD Touch Display 2 in 1 Business Laptop (8GB/512GB SSD/Intel Integrate d Graphics/WIn 10 + MSO/Backlit KB + FPR + Active Pen /Pebble Metal Color, 1.5kg) | 7100.00 | INR      | https://m.media-amazon.com/images/I/514FlU4YKBS._SL1080_.jp
g | 2 | c3cb85e4-528f-44f0-9dd4-b5cdab445697 | Acer Aspire C24 23.8 inch Full HD IPS All in One Desktop I Intel Core i3 1005G1 I 8GB DDR4 I 512GB SSD I Windows 11 Hom

```

12. Dockerize the backend by going into the directory using docker command

```

ubuntu@ip-10-0-2-213:~/project/hadiya_products_api$ sudo docker build -t hadiya_products.api .
Sending build context to Docker daemon 557.1kB
Step 1/7 : FROM node:16-alpine
--> 2573171e0124
Step 2/7 : WORKDIR /app/products
--> Running in 3f89674b948c
Removing intermediate container 3f89674b948c
--> 15964b4a9a10
Step 3/7 : COPY package.json package-lock.json ./
--> 0a118861b358
Step 4/7 : RUN npm install
--> Running in 617db2d0da5f
npm WARN deprecated querystring@0.2.0: The querystring API is considered Legacy. new code should use the URLSearchParams API instead.
npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.
npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.
npm WARN deprecated core-js@2.6.12: core-js@<3.23.3 is no longer maintained and not recommended for usage due to the number of issues. Because of the V8 engine whims, feature detection in old core-js versions could cause a slowdown up to 100x even if nothing is polyfilled. Some versions have web compatibility issues. Please, upgrade your dependencies to the actual version of core-js.
added 450 packages, and audited 451 packages in 15s

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

9 vulnerabilities (2 moderate, 5 high, 2 critical)

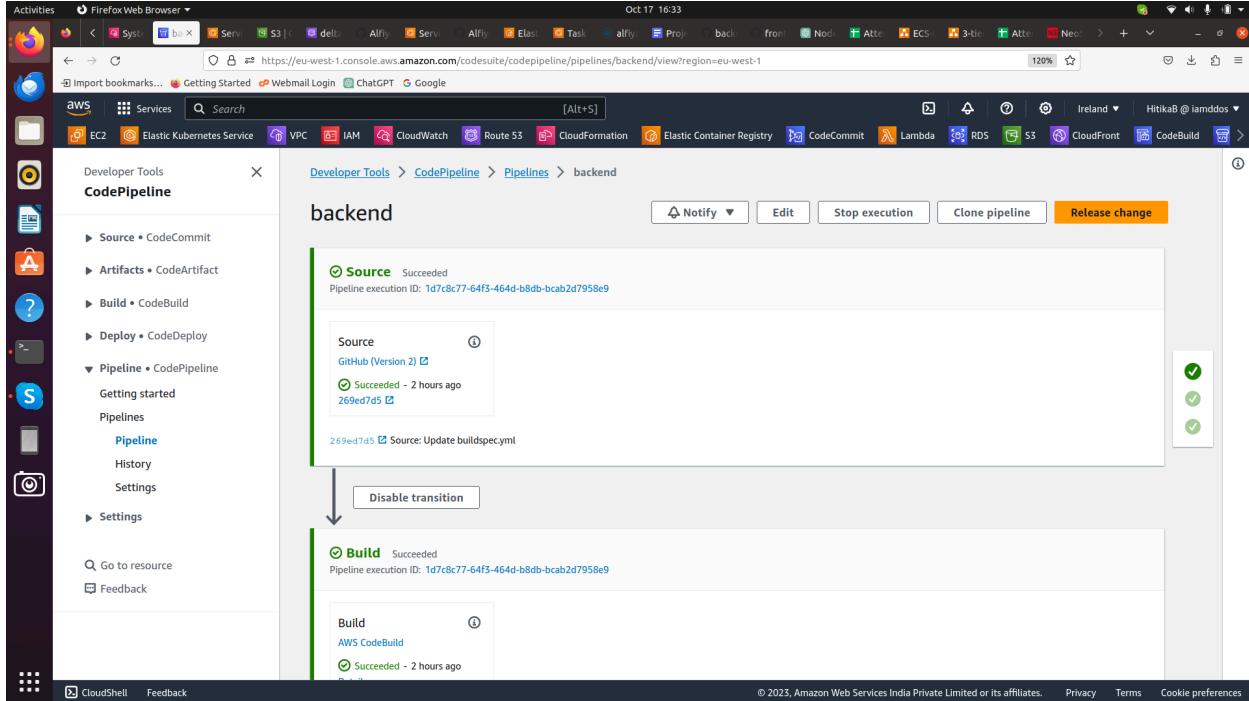
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
hadiya_products.api	latest	db7b59245354	About a minute ago	196MB
node	16-alpine	2573171e0124	2 months ago	118MB

13. Push image to ECR or docker hub

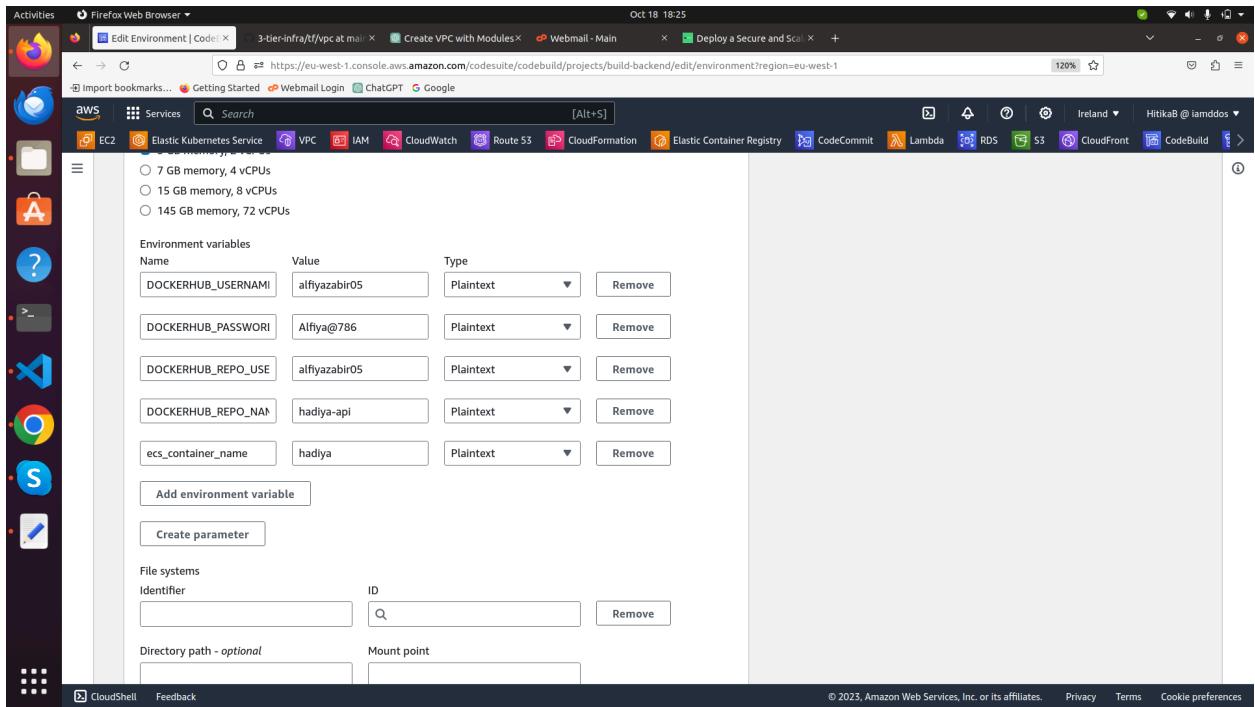
14. Create ECS and its components through terraform check if the service is running or not

15. Create Code Pipeline for continuous deployment for backend



(in the build stage - add builspec.yml in the repo or you can enter during the configuration and set environment variables)

Environment variables:



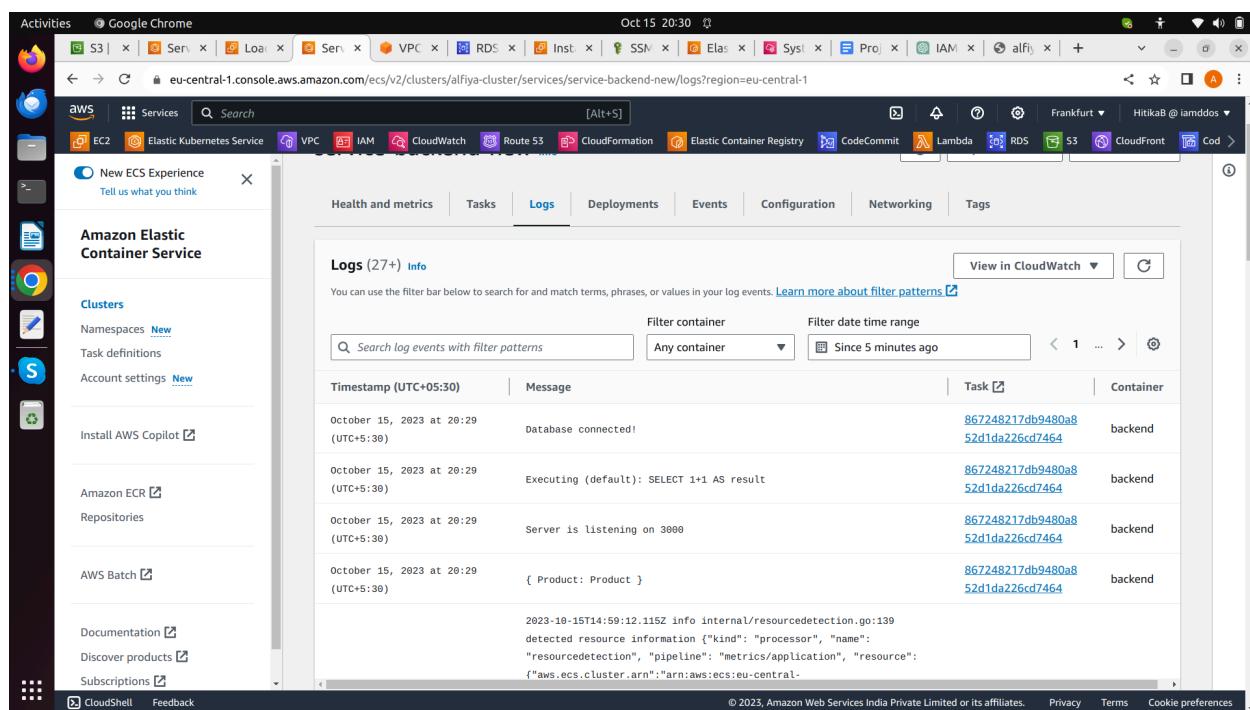
version: 0.2

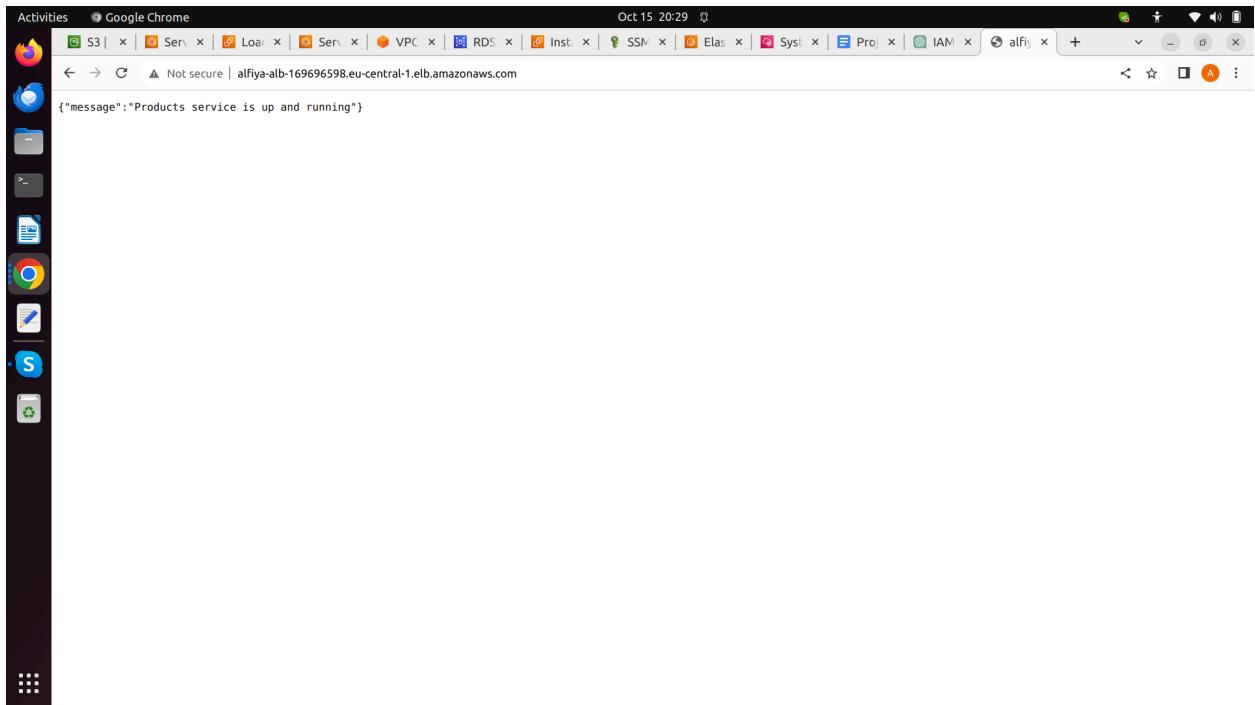
```

phases:
  pre_build:
    commands:
      - echo Logging in to DockerHub...
      - docker login -u $DOCKERHUB_USERNAME -p
$DOCKERHUB_PASSWORD
      -
REPOSITORY_URI=$DOCKERHUB_REPO_USERNAME/$DOCKERHUB_REPO_N
AME
      - COMMIT_HASH=$(echo $CODEBUILD_RESOLVED_SOURCE_VERSION |
cut -c 1-7)
      - IMAGE_TAG=${COMMIT_HASH:=latest}
  build:
    commands:
      - echo Build started on `date`
      - echo Building the Docker image...
      - docker build -t $REPOSITORY_URI:latest .
      - docker tag $REPOSITORY_URI:latest $REPOSITORY_URI:$IMAGE_TAG

```

```
post_build:  
  commands:  
    - echo Build completed on `date`  
    - echo Pushing the Docker images to DockerHub...  
    - docker push $REPOSITORY_URI:latest  
    - docker push $REPOSITORY_URI:$IMAGE_TAG  
    - echo Writing image definitions file...  
    - printf '[ {"name":"'ecs_container_name","imageUri":"%s"} ]'  
$REPOSITORY_URI:$IMAGE_TAG > imagedefinitions.json  
artifacts:  
  files: imagedefinitions.json
```





16. Add listener 443

A screenshot of the AWS CloudWatch Metrics interface. A modal dialog box is open, prompting for the creation of a new CloudWatch Metrics stream. The stream name is 'tf-target-group-alifiya'. The interface includes fields for 'Metric Name' (tf-target-group-alifiya), 'Dimensions' (None), 'Unit' (None), and 'Metric Type' (CloudWatch Metrics). Below the input fields, there is a note: 'Metrics are collected every 1 minute. Metrics older than 15 days are deleted.' At the bottom of the dialog are 'Create' and 'Cancel' buttons.

17. Request certificate from ACM and add it to the listener 443

The screenshot shows the AWS Certificate Manager (ACM) interface. On the left, there's a sidebar with options like 'List certificates', 'Request certificate', and 'Import certificate'. The main area displays a table titled 'Certificates (1)'. The table has columns for 'Certificate ID', 'Domain name', 'Type', 'Status', 'In use', 'Renewal eligibility', and 'Key algorithm'. A single row is shown, representing a certificate for the domain '*.deltawing.live' issued by Amazon.

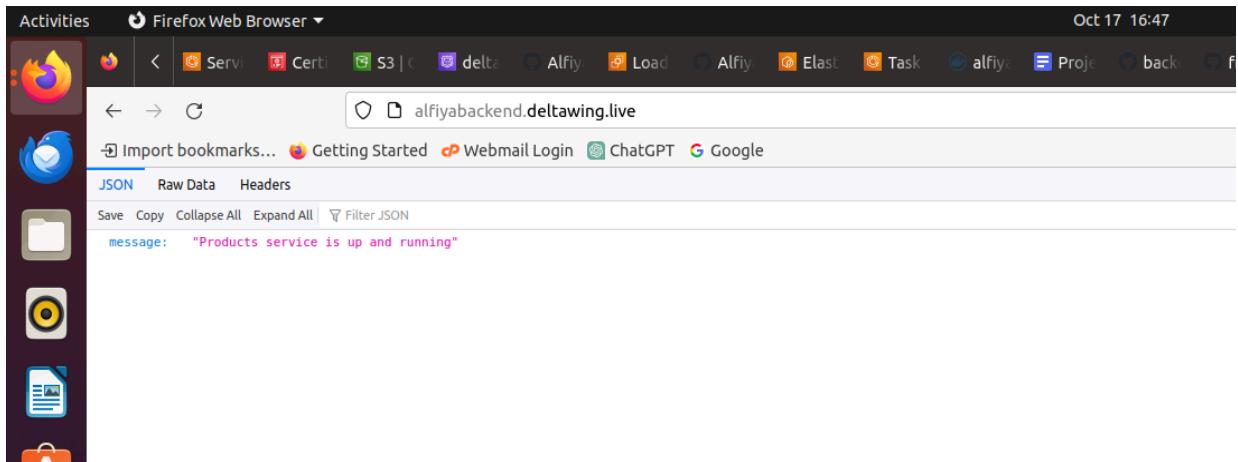
ch

18. Open port 443 in security groups of ALB

19. Add record name in the route 53

The screenshot shows the AWS Route 53 console. On the left, there's a sidebar with various navigation options. The main area shows a 'Hosted zone details' page for the domain 'deltawing.live'. In the bottom right corner, a modal window titled 'Edit record' is open. It shows a form where 'Record name' is set to 'alftybackend' and 'Value/Route traffic to' is set to 'dualstack.tf-alb-alfty-749650111.eu-west-1'. Other fields like 'Record type' (set to 'A'), 'Alias' (checked), and 'Route traffic to' (set to 'Alias to Application and Classic Load Balancer') are also visible. The modal has a 'Save' button at the bottom right.

20. Add hit the domain name on the web browser



21. Update the domain name in the frontend files for that go to the directory in src/environments/environments.prod.ts file and update in the URL

```
ubuntu@ip-10-0-2-213:~$ cd project/
ubuntu@ip-10-0-2-213:~/project$ ls
hadiya-products-database hadiya_products_api hadiya_products_ng
ubuntu@ip-10-0-2-213:~/project$ cd hadiya_products_ng/
ubuntu@ip-10-0-2-213:~/project/hadiya_products_ng$ ls
README.md angular.json karma.conf.js package-lock.json package.json src tsconfig.app.json tsconfig.json tsconfig.spec.json
ubuntu@ip-10-0-2-213:~/project/hadiya_products_ng$ cd src/
ubuntu@ip-10-0-2-213:~/project/hadiya_products_ng/src$ cd environments/
ubuntu@ip-10-0-2-213:~/project/hadiya_products_ng/src/environments$ ls
environment.prod.ts environment.ts
ubuntu@ip-10-0-2-213:~/project/hadiya_products_ng/src/environments$ sudo nano environment.prod.ts
```

```
Alifiya-git / frontend
Code Issues Pull requests Actions Projects Wiki Security Insights Settings
main / frontend / src / environments / environment.prod.ts
Alifiyazabir files
Code Blame 4 lines (4 loc) · 108 Bytes Code 55% faster with GitHub Copilot
1 export const environment = {
2   production: true,
3   apiGatewayURL: 'https://alifiyabackend.deltawing.live'
4 };
```

22. Create S3 and Cloudfront using terraform

The screenshot shows the AWS S3 console interface. On the left, there's a sidebar with various services like EC2, VPC, IAM, CloudWatch, etc. The main area is titled 'Amazon S3 > Buckets > tf-service-alfiya'. It shows a table of objects with columns for Name, Type, Last modified, Size, and Storage class. The objects listed are:

Name	Type	Last modified	Size	Storage class
3rdpartylicenses.txt	txt	October 17, 2023, 14:06:15 (UTC+05:30)	12.7 KB	Standard
favicon.ico	ico	October 17, 2023, 14:06:15 (UTC+05:30)	948.0 B	Standard
index.html	html	October 17, 2023, 14:06:15 (UTC+05:30)	1.3 KB	Standard
main.46685c4e99ad6435.js	js	October 17, 2023, 14:06:15 (UTC+05:30)	271.7 KB	Standard

23. Update the S3 permissions if it is not

The screenshot shows the 'Permissions' tab in the AWS S3 console for the 'tf-service-alfiya' bucket. The 'Publicly accessible' option is selected. In the 'Block public access (bucket settings)' section, 'Block all public access' is set to 'Off'. Below that, there's a link to 'Individual Block Public Access settings for this bucket'. At the bottom, there's a 'Bucket policy' section with an 'Edit' button.

```
{
  "Version": "2012-10-17",
  "Statement": [
    ...
  ]
}
```

24. Update the bucket policy

```
{
  "Version": "2012-10-17",
  "Statement": [
    ...
  ]
}
```

```

    {
        "Sid": "PublicReadGetObject",
        "Effect": "Allow",
        "Principal": "*",
        "Action": "s3:GetObject",
        "Resource": "arn:aws:s3:::tf-service-alfiya/*"
    },
    {
        "Sid": "2",
        "Effect": "Allow",
        "Principal": {
            "AWS": "arn:aws:iam::cloudfront:user/CloudFront Origin Access Identity
E25OVSW93YSJBV"
        },
        "Action": "s3:GetObject",
        "Resource": "arn:aws:s3:::tf-service-alfiya/*"
    }
]
}

```

25. Enable the static website hosting

The screenshot shows the 'Static website hosting' configuration for a bucket. It includes fields for 'Static website hosting' (set to 'Enabled'), 'Hosting type' (set to 'Bucket hosting'), and 'Bucket website endpoint' (set to <http://alfiya-frontend.s3-website.eu-central-1.amazonaws.com>). There is also a note about the Region-specific website endpoint.

26. In the cloudfront which is created using terraform update the domain name and add SSL certificate and invalidation

The screenshot shows the AWS CloudFront console with the distribution configuration for E1VOOQ38TFS023. The General tab is selected, displaying the following details:

- Distribution domain name:** d5ve0xflae1fw.cloudfront.net
- ARN:** arn:aws:cloudfront::31898877498.distribution/E1VOOQ38TFS023
- Last modified:** October 16, 2023 at 12:27:52 PM UTC

Settings:

- Description:** My CloudFront Distribution Comment
- Alternate domain names:** alfyawfrontend.deltawing.live
- Standard logging:** Off
- Price class:** Use all edge locations (best performance)
- Custom SSL certificate:** deltawing live
- Cookie logging:** Off
- Supported HTTP versions:** HTTP/2, HTTP/1.1, HTTP/1.0
- Security policy:** TLSv1_2_2021
- Default root object:** -

Security - Web Application Firewall (WAF): Core protections are disabled.

27. Create a pipeline for the frontend build and deploy it into s3

The screenshot shows the AWS CodePipeline console with the 'frontend' pipeline configuration. The pipeline consists of two stages:

- Source:** GitHub (Version 2) - Succeeded (Pipeline execution ID: e9b6493d-9224-4ff6-ad78-b732f481ca0b)
- Build:** AWS CodeBuild - Succeeded (Pipeline execution ID: e9b6493d-9224-4ff6-ad78-b732f481ca0b)

The pipeline is currently disabled, as indicated by the 'Disable transition' button.

(Upload buildspec.yml in the repository)

version: 0.2

```
env:  
variables:  
  BUILD_FOLDER: "dist"
```

```
phases:  
install:  
  runtime-versions:  
    nodejs: 18  
  commands:  
    - echo Installing source NPM dependencies...  
    - npm install  
    - npm install -g @angular/cli
```

```
build:  
  commands:  
    - echo Build started  
    - ng build
```

```
artifacts:  
files:  
  - '**/*'  
base-directory: 'dist'  
discard-paths: yes
```

Oct 15 22:18

s3.console.aws.amazon.com/s3/buckets/alfiya-frontend?region=eu-central-1&tab=objects

Amazon S3 > Buckets > alfiya-frontend

alfiya-frontend Info

Publicly accessible

Objects (7)

Name	Type	Last modified	Size	Storage class
3rdpartylicenses.txt	txt	October 15, 2023, 22:16:41 (UTC+05:30)	12.7 KB	Standard
favicon.ico	ico	October 15, 2023, 22:16:41 (UTC+05:30)	948.0 B	Standard
index.html	html	October 15, 2023, 22:16:41 (UTC+05:30)	1.3 KB	Standard

28. Add the record name in the route 53

Oct 17 17:13

https://us-east-1.console.aws.amazon.com/route53/v2/hostedzones?region=eu-west-1#ListRecordSets/Z0570330D9HKG2IU051

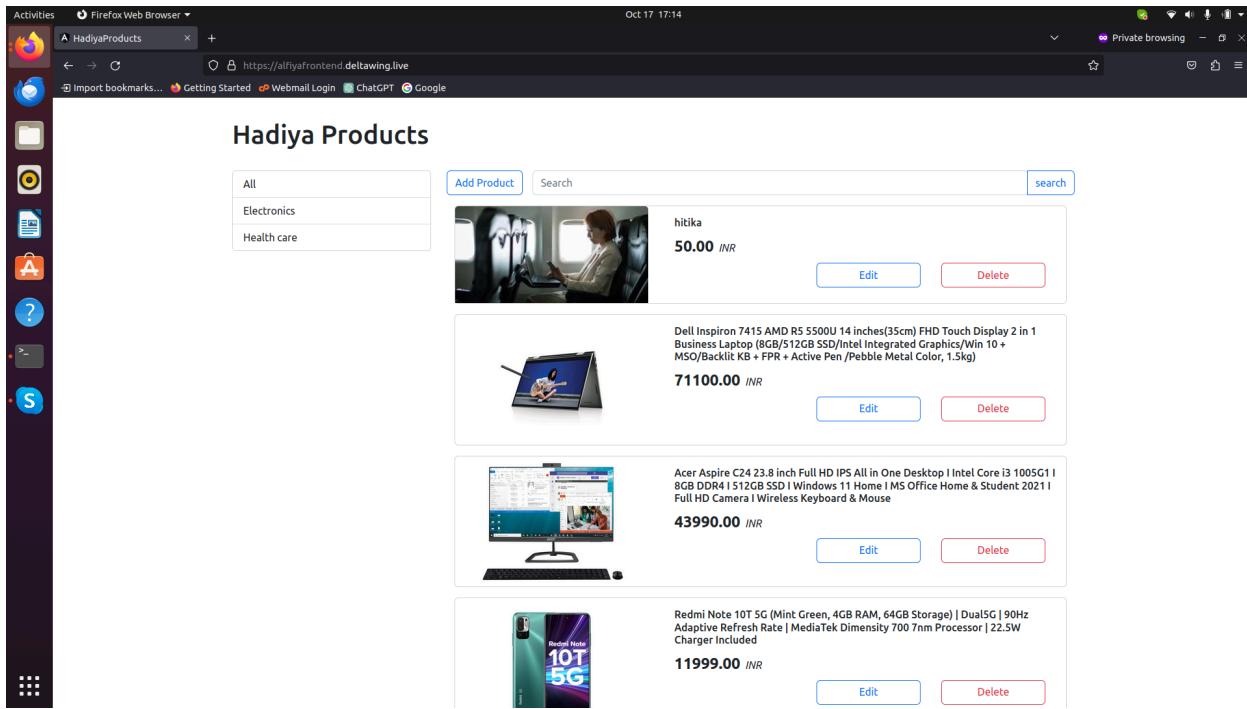
Route 53 > Hosted zones > deltawing.live

deltawing.live Info

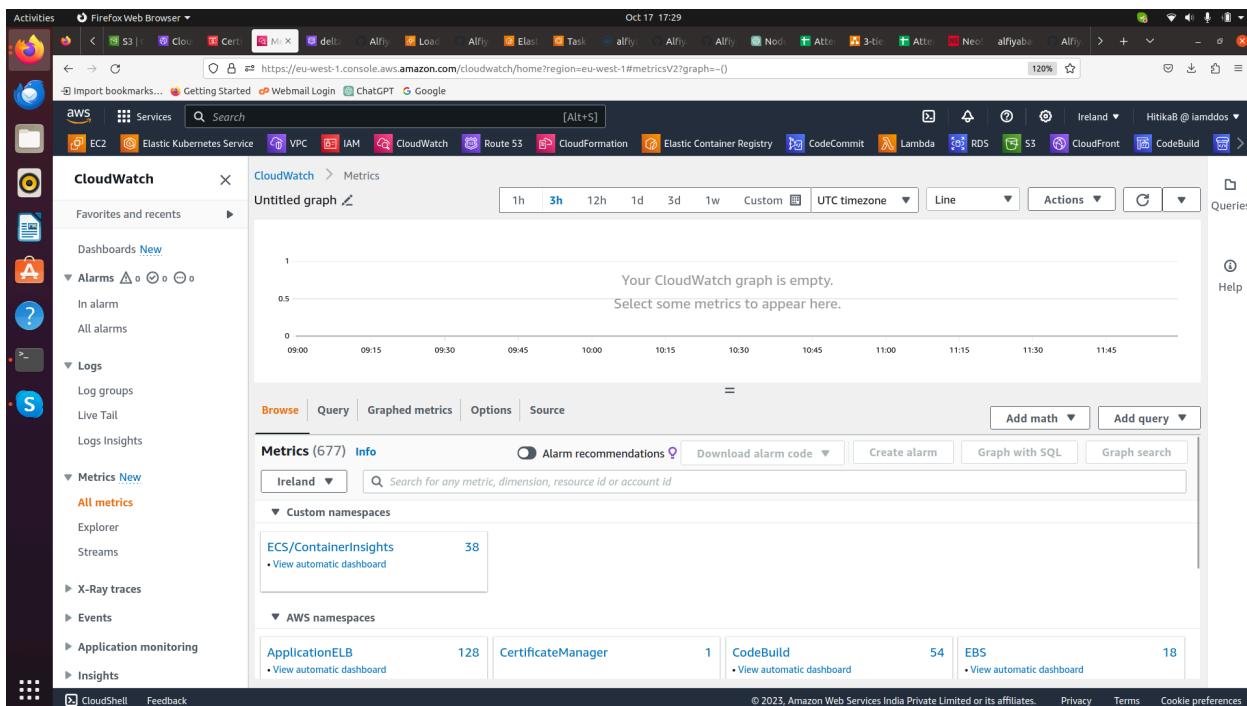
Records (7)

Record name	Type	Routing policy	Alias	Value/Route traffic to	TTL (s)
deltawing.live	NS	Simple	-	No ns-493.awsdns-61.com. ns-1197.awsdns-21.org. ns-2015.awsdns-59.co.uk. ns-878.awsdns-45.net.	17280
deltawing.live	SOA	Simple	-	No ns-493.awsdns-61.com. awsdns...	900
_dd9c2f26c6abb7d4177ac7ef6...	CNAME	Simple	-	No _b654eeaa74397348531e58d...	300
alfiyabackend.deltawing.live	A	Simple	-	Yes dualstack.tf-alb-alfiya-74963...	-
alfiyafrontend.deltawing.live	A	Simple	-	Yes d3veoxfe18fw.cloudfront...	-
backend.deltawing.live	A	Simple	-	Yes dualstack.tf-alb-hitika-1565...	-
frontend.deltawing.live	A	Simple	-	Yes dxoa620w0pzgn.cloudfront...	-

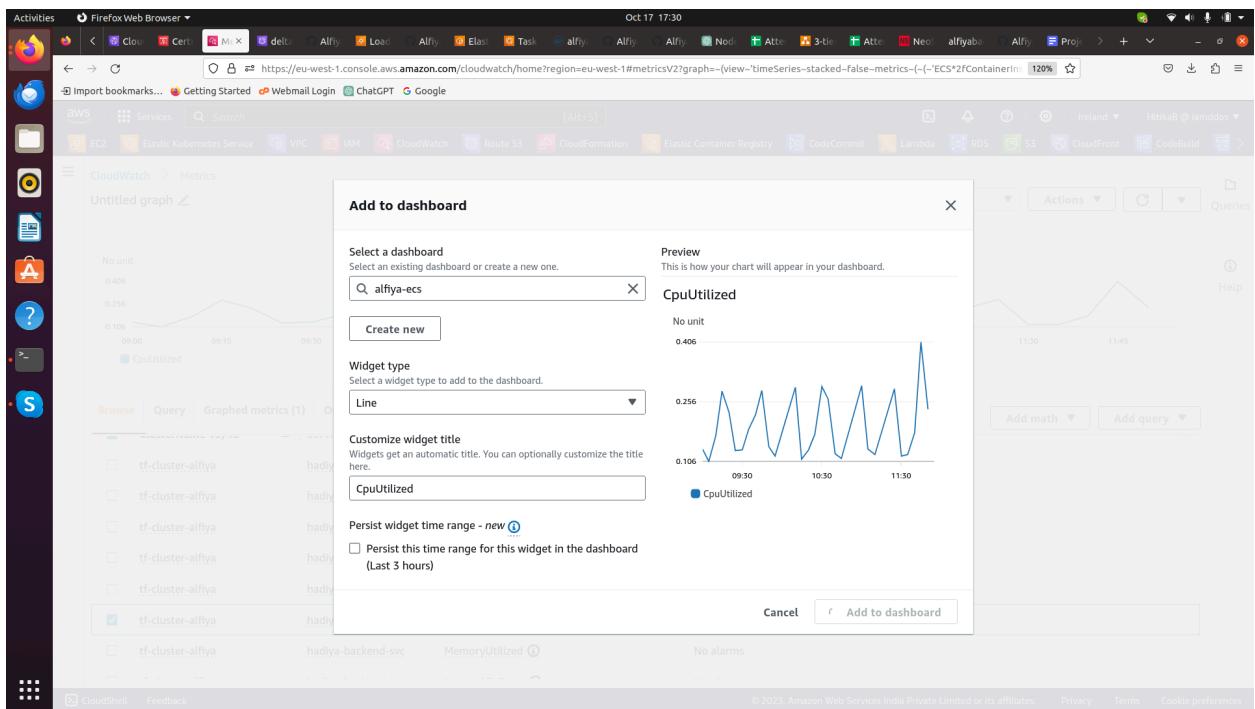
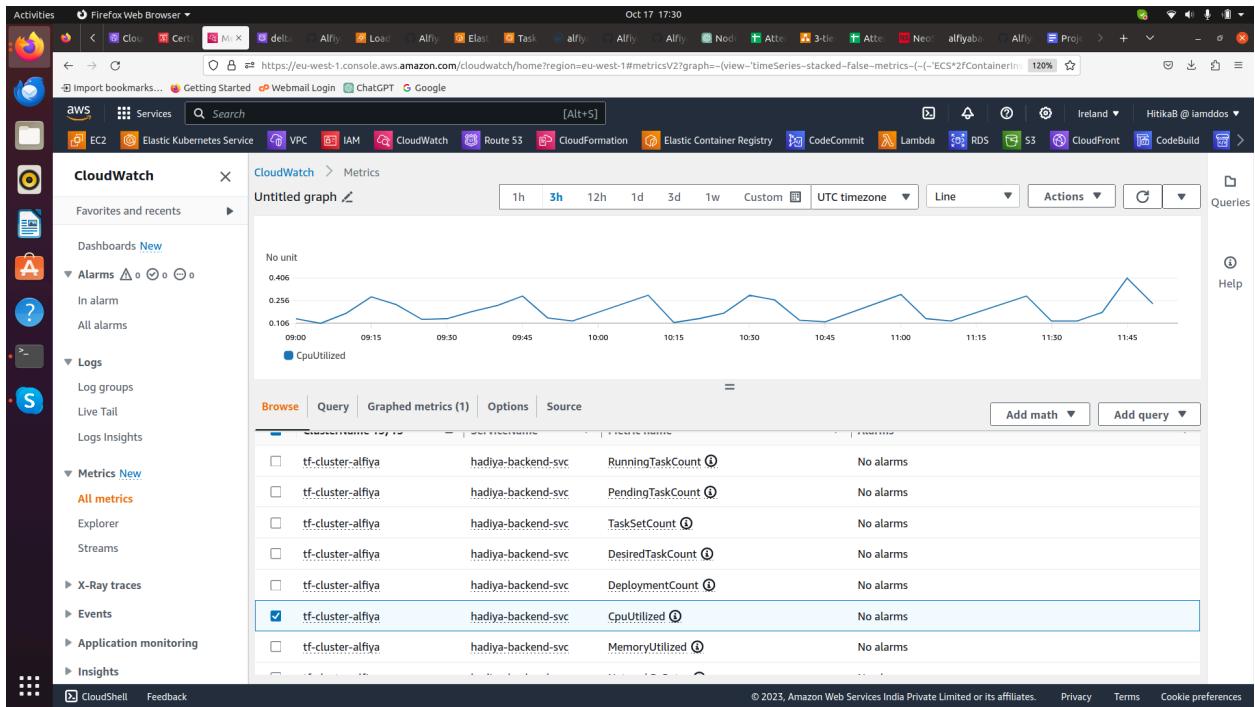
29. Hit the domain name on the web browser

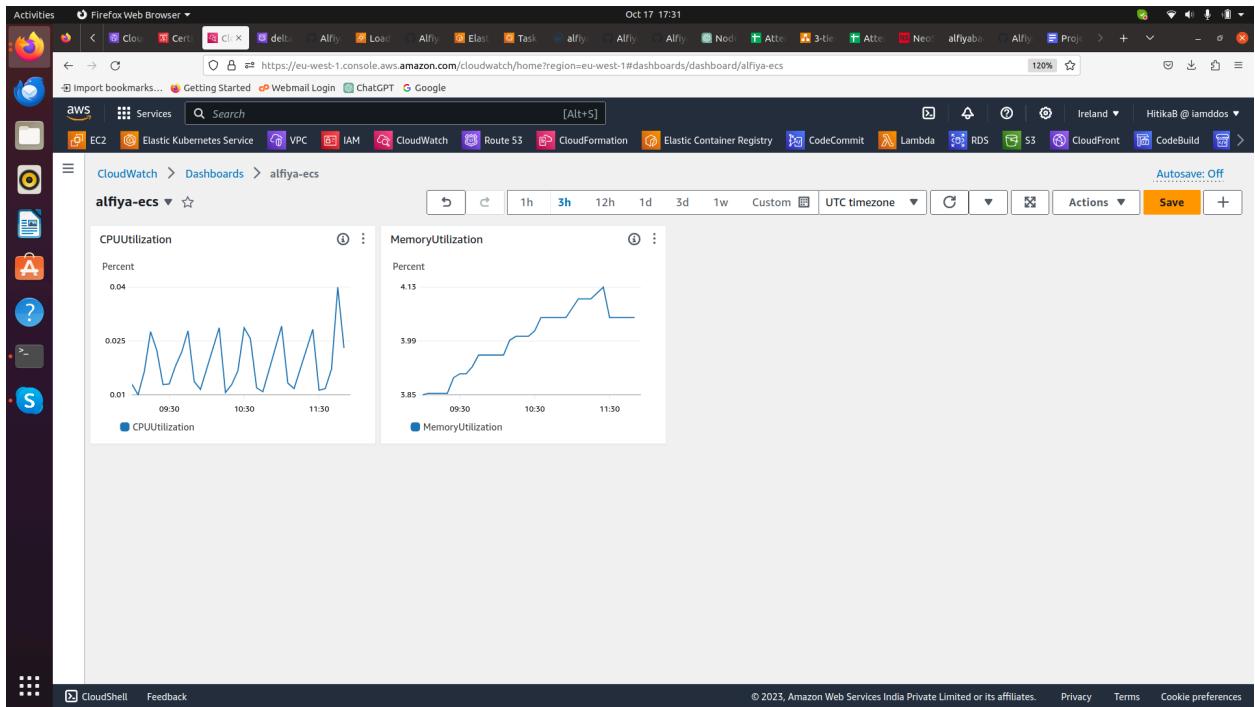


30. For monitoring you can update the cluster and enable monitoring if not done through terraform



31. Configure the metrics you want on the dashboard





```
docker run --rm -it -v $(pwd):/app -w /app node:16-alpine npm install then docker
run --rm -it -v $(pwd):/app -w /app node:16-alpine npm run build
```