

Dojo App Deployment to ECS Using TF Advanced

Overview

This App consists of two components. Frontend and Backend.

A React-based frontend for the DevOps Learning Platform. This application provides an interactive interface for learning DevOps concepts and taking quizzes.

A Flask-based REST API backend for the DevOps Learning Platform. This application provides endpoints for managing DevOps topics and quizzes.

Terraform:

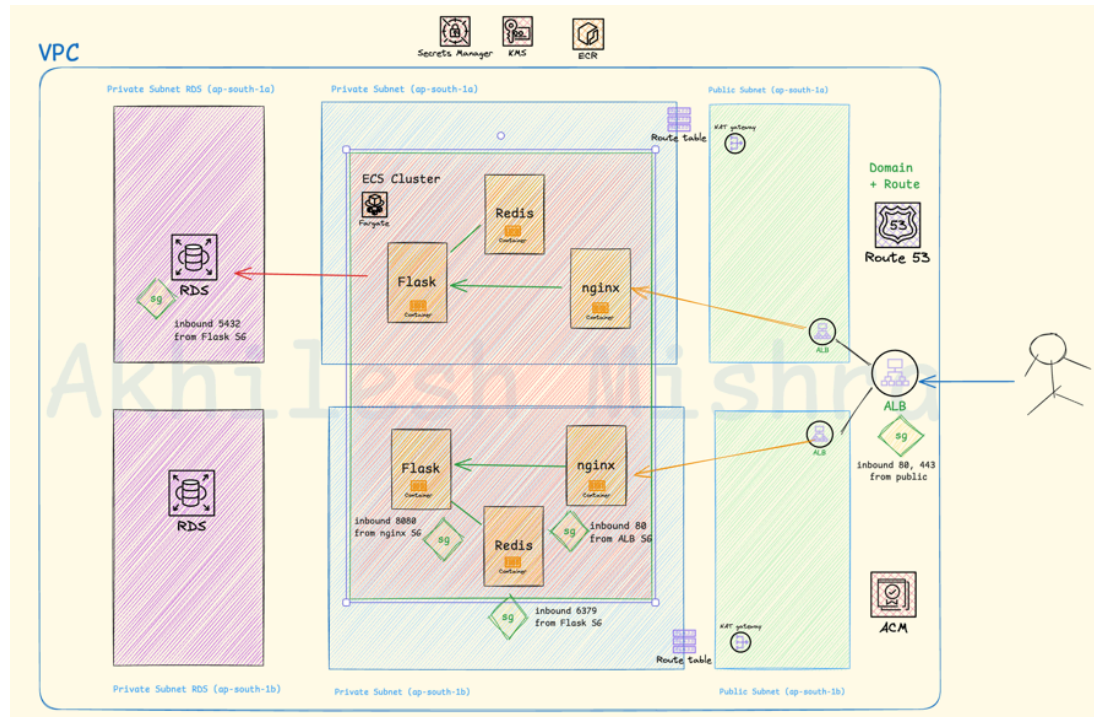
Infrastructure Deployment with Terraform

<https://github.com/rajeshchandranaws3/bootcamp-august-rajesh/tree/main/class14-assignment/infrastructure>

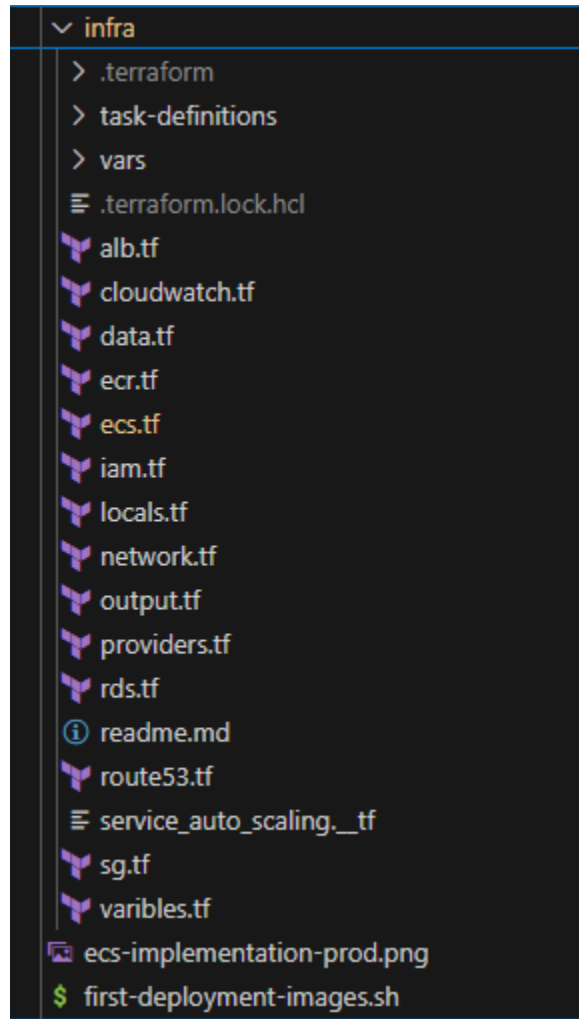
Purpose:

The Terraform code is used to create aws vpc, public subnets, private subnets, igw, nat, rds, ecs, alb etc..

Architecture Diagram



Code Structure




AWS Resources:

S3 Backend


[Buckets](#) > [state-bucket-307946636515](#) > [class14/](#) > [dev/](#)

dev/

Objects | Properties

Objects (1)  [Copy S3 URI](#)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3](#) permissions. [Learn more](#)

<input type="checkbox"/>	Name	Type
<input type="checkbox"/>	 terraform.tfstate	tfstate

Network:

[vpc-01edfed02bc7b7e65](#)

[acl-0348daa3998ec2395](#) | No | 10.0.0.0/16 | [Owner ID](#)
[IPv6 CIDR \(Network border group\)](#) | [Network Address Usage metrics](#) | [Route 53 Resolver DNS Firewall rule groups](#) | [307946636515](#)
- | Disabled | -

[Resource map](#) | [CIDRs](#) | [Flow logs](#) | [Tags](#) | [Integrations](#)

Resource map [info](#) [Show all details](#)

VPC

Your AWS virtual network

dev-vpc

Subnets (6)

Subnets within this VPC

us-east-1a

- ECS Fargate Public Subnet 0
- RDS Private Subnet 0
- ECS Fargate Private Subnet 0

us-east-1b

- ECS Fargate Public Subnet 1
- RDS Private Subnet 1
- ECS Fargate Private Subnet 1

Route tables (3)

Route network traffic to resources

- ECS EC2 Private Route Table 1
- rtb-0821ad7c2e9ceddba
- ECS EC2 Private Route Table 0

Network Connections (3)

Connections to other networks

- igw-0ce5ee5c203cf8400
- nat-08a5919fd4507a217
- nat-0e7b4b6e1ba05274a

Security Groups:

SG for ALB:

sg-0ee8096b1a316ea45 - dev-lb-sg

Details

Security group name
dev-lb-sg

Security group ID
sg-0ee8096b1a316ea45

Description
controls access to the Application Load Balancer (ALB)

VPC ID
vpc-01edfed02bc7b7e65

Owner
307946636515

Inbound rules count
2 Permission entries

Outbound rules count
1 Permission entry

Inbound rules | Outbound rules | Sharing - new | VPC associations - new | Tags

Inbound rules (2)

	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source
<input type="checkbox"/>	-	sgr-07ed821e632c3d34b	IPv4	HTTP	TCP	80	0.0.0.0/0
<input type="checkbox"/>	-	sgr-0a6993d0e42859b3d	IPv4	HTTPS	TCP	443	0.0.0.0/0

SG for Frontend (React.js):

sg-0b93f7000469bec42 - dev-ecs-tasks-react-sg

Details

Security group name
dev-ecs-tasks-react-sg

Security group ID
sg-0b93f7000469bec42

Description
allow inbound access from the ALB only

VPC ID
vpc-01edfed02bc7b7e65

Owner
307946636515

Inbound rules count
1 Permission entry

Outbound rules count
1 Permission entry

Inbound rules | Outbound rules | Sharing - new | VPC associations - new | Tags

Inbound rules (1)

	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source
<input type="checkbox"/>	-	sgr-0f3f1d7533f662466	-	HTTP	TCP	80	sg-0ee8096b1a316ea45 / dev-lb-sg

SG for Backend (Flask App):

sg-0b007ca84e785256a - dev-ecs-tasks-flask-sg

Details

Security group name
dev-ecs-tasks-flask-sg

Security group ID
sg-0b007ca84e785256a

Description
allow inbound access from the ALB only

VPC ID
vpc-01edfed02bc7b7e65

Owner
307946636515

Inbound rules count
1 Permission entry

Outbound rules count
1 Permission entry

Inbound rules | Outbound rules | Sharing - new | VPC associations - new | Tags

Inbound rules (1)

	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source
<input type="checkbox"/>	-	sgr-0cb76e687da2b9774	-	Custom TCP	TCP	8000	sg-0b93f7000469bec42 / dev-ecs-tasks-react-sg

SG for RDS (Postgresql):

sg-0d66f2a3607a8b9c1 - dev-rds-sg

Details

Security group name dev-rds-sg	Security group ID sg-0d66f2a3607a8b9c1	Description allow inbound access from the ECS only	VPC ID vpc-01edfed02bc7b7e65
Owner 307946636515	Inbound rules count 2 Permission entries	Outbound rules count 1 Permission entry	

[Inbound rules](#) | [Outbound rules](#) | [Sharing - new](#) | [VPC associations - new](#) | [Tags](#)

Inbound rules (2)

<input type="checkbox"/>	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source
<input type="checkbox"/>	-	sgr-0e3c0dfd4583eb786	-	PostgreSQL	TCP	5432	sg-0b007ca846785256a / dev-ecs-tasks-flask-sg

Database:

dev-app-db

Summary

DB identifier dev-app-db	Status Available	Role Instance	Engine PostgreSQL
CPU 6.98%	Class db.t3.small	Current activity 0 Connections	Region & AZ us-east-1a

[Connectivity & security](#) | [Monitoring](#) | [Logs & events](#) | [Configuration](#) | [Zero-ETL integrations](#) | [Maintenance](#)

Connectivity & security

Endpoint & port

Endpoint
dev-app-db.cizic4iqc955.us-east-1.rds.amazonaws.com

Port
5432

Networking

Availability Zone
us-east-1a

VPC
dev-vpc (vpc-01edfed02bc7b7e65)

Subnet group
dev-app-db-subnet-group

Security

VPC security groups
dev-rds-sg (sg-0d66f2a3607a8b9c1)
Active

Publicly accessible
No

Certificate authority
Info

<	Connectivity & security	Monitoring	Logs & events	Configuration	Zero-ETL integrations	Maintenance & backups	Data migrations	T
Instance								
Configuration			Instance class		Primary storage		Monitoring	
DB instance ID dev-app-db			Instance class db.t3.small		Encryption Enabled		Monitoring type Database Insights - Standard	
Engine version 14.17			vCPU 2		AWS KMS key dev-rds-kms-key		Performance Insights Disabled	
RDS Extended Support Enabled			RAM 2 GB		Storage type General Purpose SSD (gp3)		Enhanced Monitoring Disabled	
DB name mydb			Availability		Storage 30 GiB		Logs published to Cloudwatch PostgreSQL log Upgrade log	
License model Postgresql License			Master username postgres		Provisioned IOPS 3000 IOPS			
Option groups default:postgres-14 In sync			Master password *****		Storage throughput 125 MiBps			
Amazon Resource Name (ARN) arn:aws:rds:us-east-1:307946636515:db:dev-app-db			IAM DB authentication Not enabled		Storage autoscaling Enabled			
			Multi-AZ No		Maximum storage threshold 100 GiB			

ECR

	Repository name ▲	URI
<input type="radio"/>	dev-app-flask	307946636515.dkr.ecr.us-east-1.amazonaws.com/dev-app-flask
<input type="radio"/>	dev-app-react	307946636515.dkr.ecr.us-east-1.amazonaws.com/dev-app-react

Images:

Execute the [first-image-deployment.sh](#) to update the images into ecr.

```

class14-assignment > $ first-deployment-images.sh
4  AWS_REGION="us-east-1"
5  AWS_ACCOUNT_ID="307946636515"
6  environment="dev"
7  app_name="app"
8  REPOSITORY_NAME_APP="${environment}-${app_name}-flask"
9  REPOSITORY_NAME_FRONTEND="${environment}-${app_name}-react"
10
11 # Login to ECR
12 aws ecr get-login-password --region $AWS_REGION | docker login --username AWS --password-stdin $AWS_ACCOUNT_ID.dkr.ecr.$AWS_REGION.amazonaws.com
13
14 # Function to build and push Docker images
15 build_and_push() {
16     local dockerfile_path=$1
17     echo "Dockerfile path: $dockerfile_path"
18     local ecr_repository=$2
19     echo "ECR Repository: $ecr_repository"
20     local ecr_base_url=$AWS_ACCOUNT_ID.dkr.ecr.$AWS_REGION.amazonaws.com
21     echo "ECR Base URL: $ecr_base_url"
22     local tag="latest"
23     local image_final_path=$ecr_base_url/$ecr_repository:$tag
24     echo "Final image path: $image_final_path"
25
26
27     echo "Building $service image..."
28     docker buildx build --platform linux/amd64 -t $image_final_path $dockerfile_path
29     if [ $? -ne 0 ]; then
30         echo "Docker build failed for $dockerfile_path"
31         exit 1
32     fi
33
34     echo "Pushing $dockerfile_path image to ECR..."
35     docker push $image_final_path
36     if [ $? -ne 0 ]; then
37         echo "Docker push failed for $image_final_path"
38         exit 1
39     else
40         echo "Successfully pushed $image_final_path"
41     fi
42 }
43
44 # Build and push images
45 build_and_push "app/backend" $REPOSITORY_NAME_APP
46 build_and_push "app/frontend" $REPOSITORY_NAME_FRONTEND
47
48 echo "All images have been built and pushed successfully."

```

ECS:

Cluster:

dev-app-cluster

Last updated

November 1, 2025, 13:14 (UTC+5:30)

Update cluster

Delete cluster

Cluster overview

ARN

arn:aws:ecs:us-east-1:307946636515:cluster/dev-app-cluster

Status

Active

CloudWatch monitoring

Default

Registered container instances

-

Services

Draining

-

Active

2

Tasks

Pending

-

Running

4

Task Definition:

Task definition	Status of last revision
dev-app-flask	ACTIVE
dev-app-react	ACTIVE

JSON for Flask:

```
{
  "compatibilities": [
    "EC2",
    "FARGATE",
    "MANAGED_INSTANCES"
  ],
  "containerDefinitions": [
    {
      "command": [
        "bash",
        "-c",
        "sleep 10 && ./migrate.sh && gunicorn --bind 0.0.0.0:8000 run:app"
      ],
      "cpu": 0,
      "environment": [
        {
          "name": "ALLOWED_ORIGINS",
          "value": "*"
        },
        {
          "name": "FLASK_APP",
          "value": "run.py"
        },
        {
          "name": "FLASK_DEBUG",
          "value": "1"
        },
        {
          "name": "DATABASE_URL",
          "value": "postgresql://postgres:9xtlPd85js@dev-app-db.cizic4iqc955.us-east-1.rds.amazonaws.com:5432/mydb"
        },
        {
          "name": "DB_USERNAME",
          "value": "postgres"
        },
        {
          "name": "DB_PORT",
          "value": "5432"
        },
        {
          "name": "SECRET_KEY",
          "value": "f4a83d7e15e9f8b1c23b4b73f9b09b8827f501bbdce1a3dca7d23d8c991e3f45"
        },
        {
          "name": "DB_NAME",

```

```

        "value": "mydb"
      },
      {
        "name": "ENV",
        "value": "dev"
      },
      {
        "name": "DB_HOST",
        "value": "dev-app-db.cizic4iqc955.us-east-1.rds.amazonaws.com"
      },
      {
        "name": "DB_PASSWORD",
        "value": "9xtlPd85js"
      }
    ],
    "essential": true,
    "image": "307946636515.dkr.ecr.us-east-1.amazonaws.com/dev-app-flask:latest",
    "logConfiguration": {
      "logDriver": "awslogs",
      "options": {
        "awslogs-group": "/aws/ecs/dev-flask",
        "awslogs-region": "us-east-1",
        "awslogs-stream-prefix": "/aws/ecs/dev-flask-service"
      }
    },
    "mountPoints": [],
    "name": "flask-app",
    "portMappings": [
      {
        "appProtocol": "http",
        "containerPort": 8000,
        "hostPort": 8000,
        "name": "app",
        "protocol": "tcp"
      }
    ],
    "systemControls": [],
    "volumesFrom": []
  },
  {
    "cpu": "1024",
    "executionRoleArn": "arn:aws:iam::307946636515:role/dev-ecs-task-execution-role",
    "family": "dev-app-flask",
    "memory": "2048",
    "networkMode": "awsvpc",
    "placementConstraints": [],
    "registeredAt": "2025-11-01T07:39:46.362Z",
    "registeredBy": "arn:aws:iam::307946636515:user/user1",
    "requiresAttributes": [

```

```

    {
      "name": "com.amazonaws.ecs.capability.logging-driver.awslogs"
    },
    {
      "name": "ecs.capability.execution-role-awslogs"
    },
    {
      "name": "com.amazonaws.ecs.capability.ecr-auth"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.19"
    },
    {
      "name": "ecs.capability.execution-role-ecr-pull"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.18"
    },
    {
      "name": "ecs.capability.task-eni"
    }
  ],
  "requiresCompatibilities": [
    "FARGATE"
  ],
  "revision": 7,
  "status": "ACTIVE",
  "taskDefinitionArn": "arn:aws:ecs:us-east-1:307946636515:task-definition/dev-app-flask:7",
  "volumes": [],
  "tags": [
    {
      "key": "Environment",
      "value": "dev"
    },
    {
      "key": "Application",
      "value": "flask"
    }
  ]
}

```

JSON for React:

```

{
  "compatibilities": [
    "EC2",
    "MANAGED_INSTANCES",
    "FARGATE"
  ]
}

```

```

    ],
    "containerDefinitions": [
      {
        "cpu": 0,
        "environment": [
          {
            "name": "ENV",
            "value": "dev"
          },
          {
            "name": "BACKEND_URL",
            "value": "http://app:8000"
          }
        ],
        "essential": true,
        "image": "307946636515.dkr.ecr.us-east-1.amazonaws.com/dev-app-react:latest",
        "logConfiguration": {
          "logDriver": "awslogs",
          "options": {
            "awslogs-group": "/aws/ecs/dev-react",
            "awslogs-region": "us-east-1",
            "awslogs-stream-prefix": "/aws/ecs/dev-react-service"
          }
        },
        "mountPoints": [],
        "name": "react",
        "portMappings": [
          {
            "appProtocol": "http",
            "containerPort": 80,
            "hostPort": 80,
            "name": "react",
            "protocol": "tcp"
          }
        ],
        "systemControls": [],
        "volumesFrom": []
      }
    ],
    "cpu": "1024",
    "executionRoleArn": "arn:aws:iam::307946636515:role/dev-ecs-task-execution-role",
    "family": "dev-app-react",
    "memory": "2048",
    "networkMode": "awsvpc",
    "placementConstraints": [],
    "registeredAt": "2025-11-01T07:39:46.343Z",
    "registeredBy": "arn:aws:iam::307946636515:user/user1",
    "requiresAttributes": [
      {

```

```

      "name": "com.amazonaws.ecs.capability.logging-driver.awslogs"
    },
    {
      "name": "ecs.capability.execution-role-awslogs"
    },
    {
      "name": "com.amazonaws.ecs.capability.ecr-auth"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.19"
    },
    {
      "name": "ecs.capability.execution-role-ecr-pull"
    },
    {
      "name": "com.amazonaws.ecs.capability.docker-remote-api.1.18"
    },
    {
      "name": "ecs.capability.task-eni"
    }
  ],
  "requiresCompatibilities": [
    "FARGATE"
  ],
  "revision": 4,
  "status": "ACTIVE",
  "taskDefinitionArn": "arn:aws:ecs:us-east-1:307946636515:task-definition/dev-app-react:4",
  "volumes": [],
  "tags": [
    {
      "key": "Environment",
      "value": "dev"
    },
    {
      "key": "Application",
      "value": "react"
    }
  ]
}

```

Services:

Services									
Services	Tasks	Infrastructure	Updated	Metrics	Scheduled tasks	Configuration	Updated	Event history	New
Tags									
Services (2) Info									
Filter services by value									
Filter launch type Any launch type									
Filter scheduling strategy Any scheduling strategy									
< 1 >									
Service name	ARN	Status	Schedul...	Launch ...	Task de...	Deployments and tasks	Last deploy		
dev-app-flask-service	arn:aws:ecs:us-e	Active	REPLICA	FARGATE	dev-app-fl...	2/2 Tasks running	Comple...		
dev-app-react-service	arn:aws:ecs:us-e	Active	REPLICA	FARGATE	dev-app-re...	2/2 Tasks running	Comple...		

Tasks:

Services						
Tasks						
Infrastructure						
Updated						
Metrics						
Scheduled tasks						
Configura						
Tasks (4)						
Filter tasks by property or value						
Filter desired status Any desired status						
Task	Last status	Desired st...	T...	Health sta.		
110d4c43e3e344858cc97...	Running	Running	dev-...	Healthy		
1cf17cb53085495eb7d98...	Running	Running	dev-...	Healthy		
bbebe99991b840df87d7...	Running	Running	dev-...	Healthy		
caa7f30369fb4605b306e...	Running	Running	dev-...	Healthy		

Route 53:

Hosted zones (1)				
Automatic mode is the current search behavior optimized for best filter results. To change modes go to settings.				
Filter records by property or value				
Hosted zone name	Type	Created by	Record count	
rajeshapps.site	Public	Route 53	4	

Public **rajeshapps.site** Info

Delete zone

► Hosted zone details

Records (4)

DNSSEC signing

Hosted zone tags (0)

Records (4) Info

⌂ Delete record

Automatic mode is the current search behavior optimized for best filter results. [To change modes go to settings](#)

🔍 Filter records by property or value

Type ▼

Routing p... ▼

<input type="checkbox"/>	Record name ▼	Type ▼	Routin... ▼	Differ... ▼	Alias ▼
<input type="checkbox"/>	rajeshapps.site	NS	Simple	-	No
<input type="checkbox"/>	rajeshapps.site	SOA	Simple	-	No
<input type="checkbox"/>	dev.rajeshapps.site	A	Simple	-	Yes
<input type="checkbox"/>	_1a77d0313aa206fc17...	CNAME	Simple	-	No

Prerequisites

GitHub Secrets Required

The following secrets to be configured in GitHub repository settings:

1. AWS_ACCESS_KEY_ID: AWS IAM access key for authentication
2. AWS_SECRET_ACCESS_KEY: AWS IAM secret key for authentication

AWS Permissions Required

The IAM user/role must have permissions for:

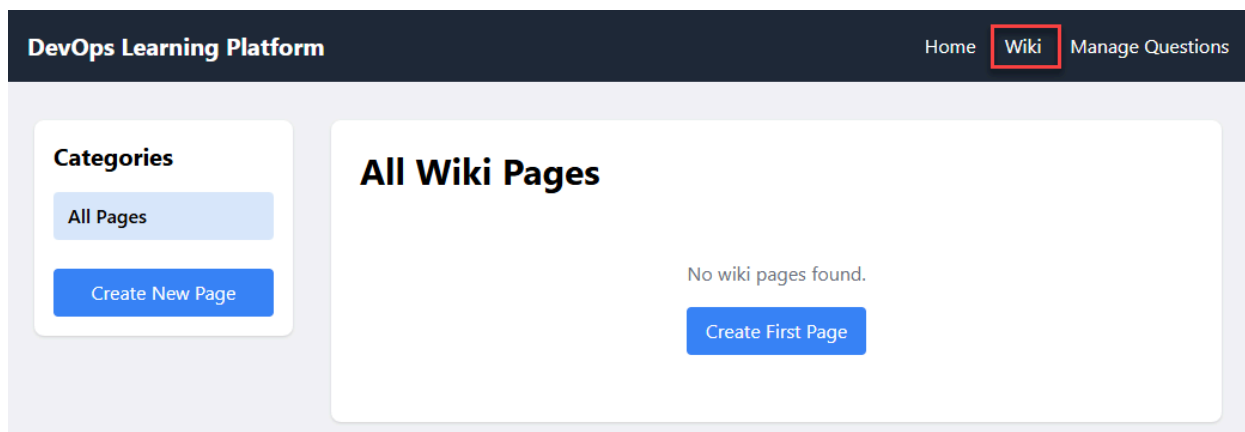
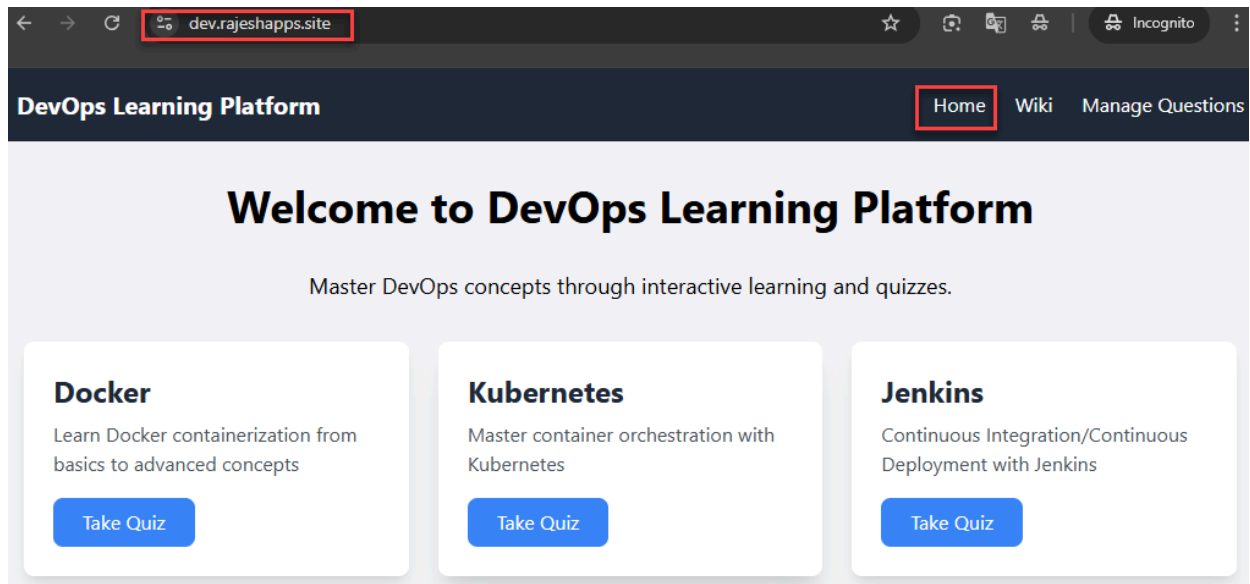
- Terraform Workflow
 - EC2 (VPC, Security Groups, Subnets)

- ECS (Cluster, Service, Task Definition)
- RDS (Database instances)
- IAM (Roles and policies)
- S3 (Terraform state backend)
- ECR (Repository creation)
- Build & Deploy Workflow
 - ECR (Push images, authentication)
 - ECS (Describe/update task definitions and services)

Terraform State Backend

S3 bucket for storing Terraform state (configured in dev.tfbackend)

Output



Question Management

Add Single Question

Topic Slug:

Question:

Options:

Bulk Upload Questions

Upload CSV File:

Format:

topic_slug,question_text,option1,option2,option3,option4,correct_answer

No file chosen

CSV Template:

```
topic_slug,question_text,option1,option2,option3,option4,correct_answer
docker,What is Docker?,A containerization platform
```

