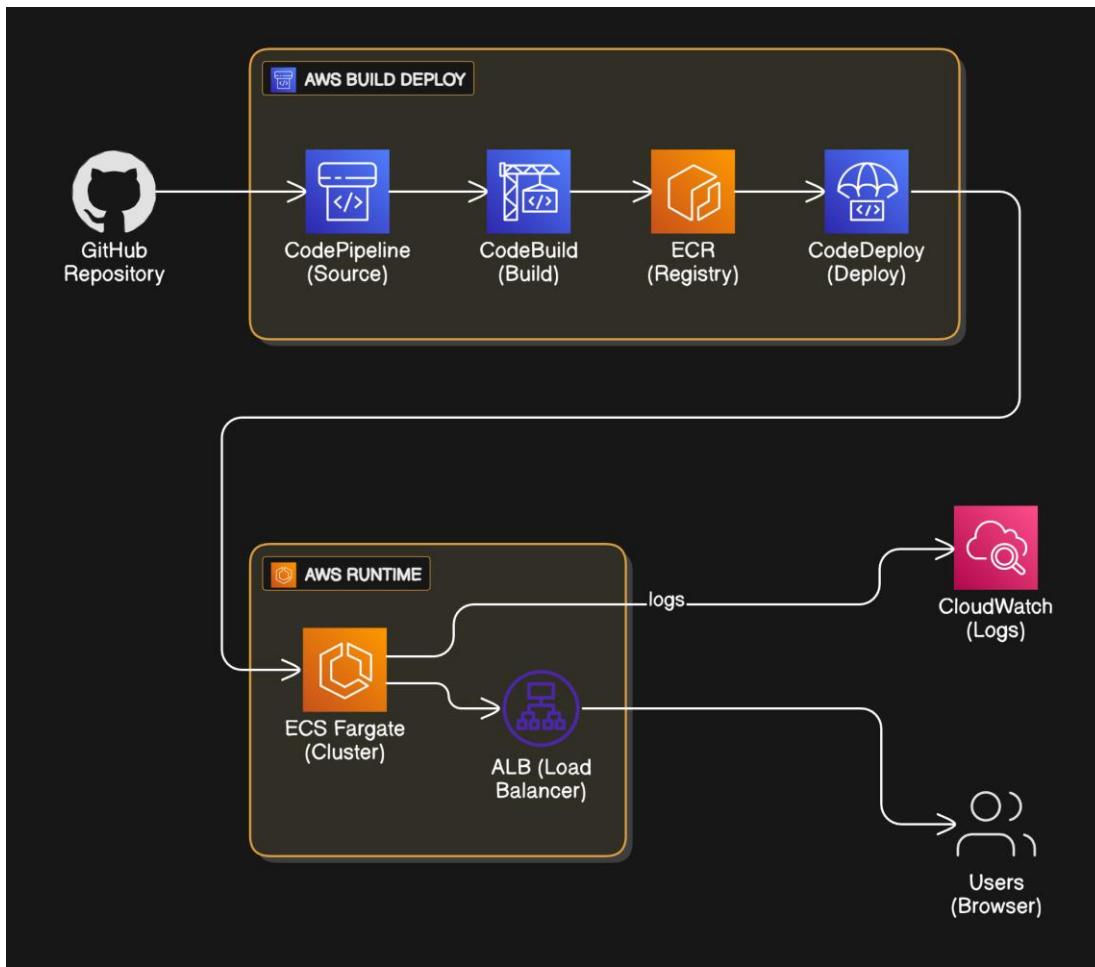


AWS DEVOPS CI/CD PIPELINE

A **production-ready, fully automated CI/CD pipeline** built on AWS using Terraform, featuring containerized deployment with ECS Fargate, automated builds with Code Pipeline, and a modern responsive web interface.

THE FULL ARCHITECTURE OF THE PROJECT:



key Components:

- **VPC:** Isolated network with public/private subnets across 2 AZs
- **ECS Fargate:** Serverless container orchestration
- **Application Load Balancer:** Traffic distribution with health checks
- **ECR:** Private Docker image registry
- **Code Pipeline:** Automated CI/CD workflow
- **Code Build:** Docker image building
- **CloudWatch:** Centralized logging and monitoring

DEPLOYMENT TASKS:

us-east-1.console.aws.amazon.com/ecr/private-registry/repositories?region=us-east-1

Amazon ECR > Private registry > Repositories

Amazon Elastic Container Registry

- ▼ Private registry
 - Repositories
 - Features & Settings
- ▼ Public registry
 - Repositories
 - Settings
- ECR public gallery
- Amazon ECS
- Amazon EKS
- Getting started
- Documentation

Private repositories (1)

Repository name	URI	Created at	Tag immutability	Encryption type
aws-devops-app	731733338203.dkr.ecr.us-east-1.amazonaws.com/aws-dev-ops-app	12 December 2025, 12:17:16 (UTC-05)	Mutable	AES-256

CloudShell Feedback Console Mobile App © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ECS CLUSTER:

us-east-1.console.aws.amazon.com/ecs/v2/clusters/aws-devops-cluster/services?region=us-east-1

Amazon Elastic Container Service

- Express Mode New
- Clusters
- Namespaces
- Task definitions
- Account settings
- Amazon ECR
- Repositories
- AWS Batch
- Documentation
- Discover products
- Subscriptions

aws-devops-cluster

Last updated 12 December 2025, 12:46 (UTC-5:00)

Cluster overview		Status		CloudWatch monitoring		Registered container instances	
ARN	arn:aws:ecs:us-east-1:731733338203:cluster/aws-devops-cluster	Status	Active	Container Insights	View in CloudWatch	-	-
Services		Tasks					
Draining	-	Active	1	Pending	-	Running	2

Services (1) Info Last updated 12 December 2025, 12:46 (UTC-5:00)

Service name	ARN	Status	Schedule...	Lau...	Task de...	Deployments and tasks
aws-devops-service	arn:aws:ecs:us-east-1:731733338203:service/aws-devops-service	Active	REPLICA	FARG...	aws-devo...	2/2 tasks

ALB:

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LoadBalancers:

EC2 > Load balancers

Load balancers (1/1) What's new?

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Name	State	Type	Scheme	IP address type	VPC ID	Availability Zones
aws-devops-alb	Active	application	Internet-facing	IPv4	vpc-0bc3a0b56256db8bb	2 Availability Zones

Load balancer: aws-devops-alb

Details

Load balancer type	Status	Load balancer IP address type
Application	Active	IPv4
Scheme	Hosted zone	Date created
Internet-facing	735SXDOIRO7X7K	December 12, 2025, 12:17:16 (UTC-05:00)

S3 BUCKET:

The screenshot shows the AWS S3 console interface. On the left, the navigation pane is open with sections like 'Buckets', 'Access management and security', and 'Storage management and insights'. The main area displays the 'aws-devops-pipeline/' bucket. It shows two objects: 'build_outp/' and 'source_out/'. There are buttons for 'Copy S3 URI', 'Copy URL', 'Download', 'Open', 'Delete', 'Actions', 'Create folder', and 'Upload'.

PIPELINE DEPLOYMENT

The screenshot shows the AWS CodePipeline console. It displays a pipeline named 'aws-devops-pipeline' with three stages: 'Source', 'Build', and 'Deploy'. Each stage has a green checkmark icon indicating success. The 'Source' stage shows a GitHub action. The 'Build' stage shows an AWS CodeBuild action. The 'Deploy' stage shows an Amazon ECS action. There are buttons for 'Edit', 'Stop execution', 'Create trigger', 'Clone pipeline', and 'Release change'.

Access Your Application

After deployment completes, Terraform will output:

```
=====
AWS DevOps Pipeline Deployment Complete!
=====

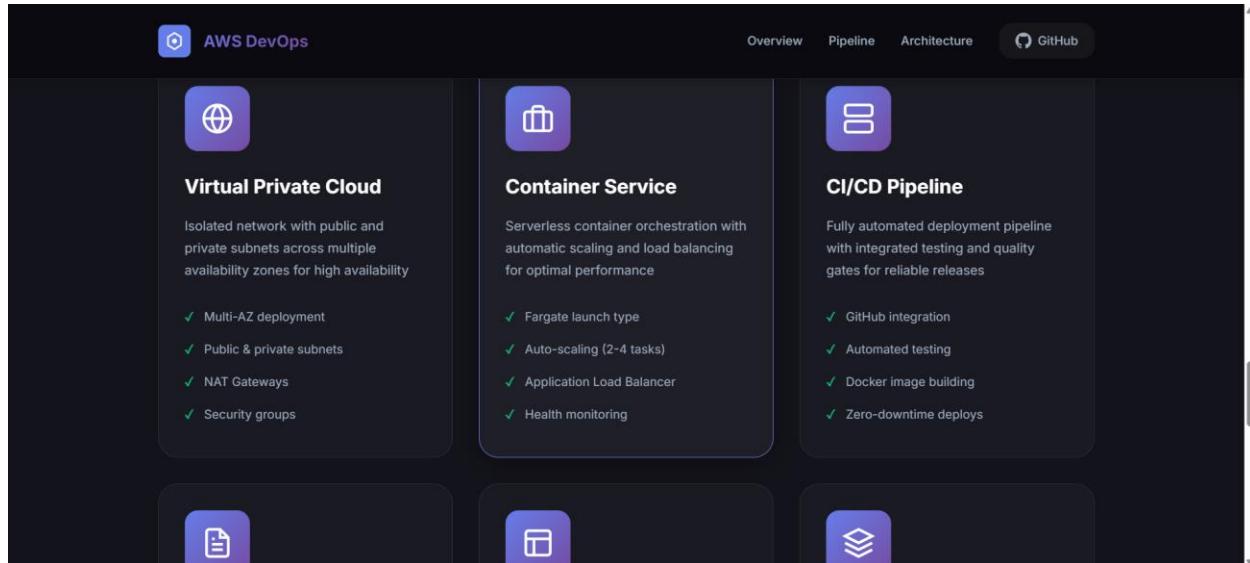
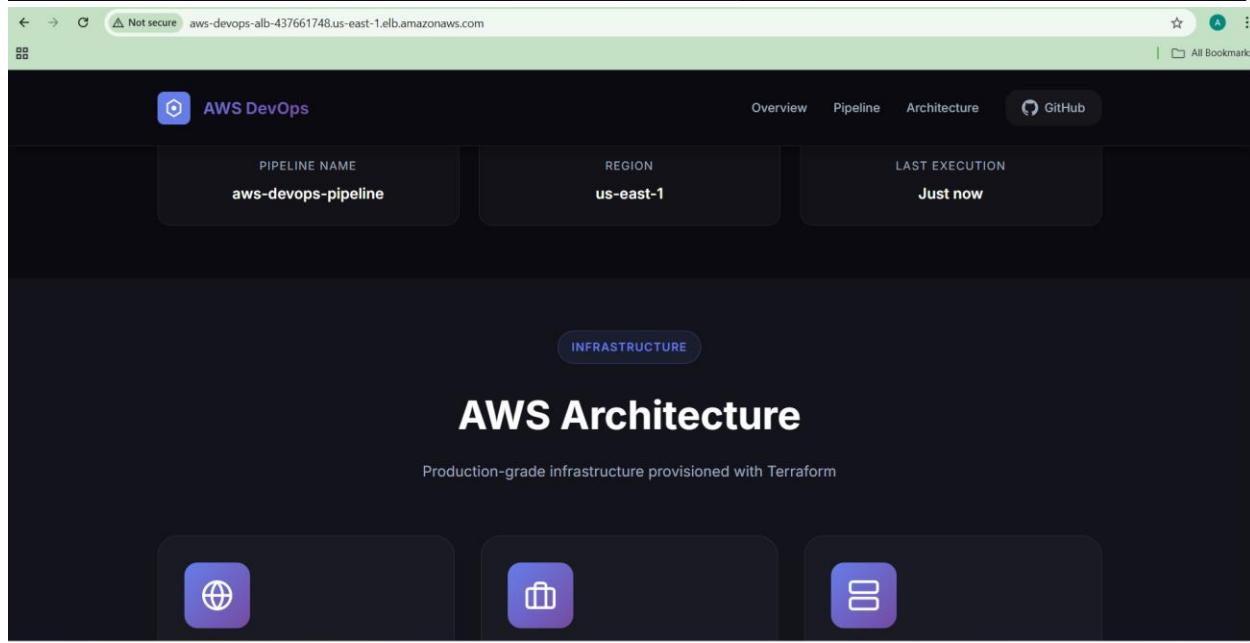
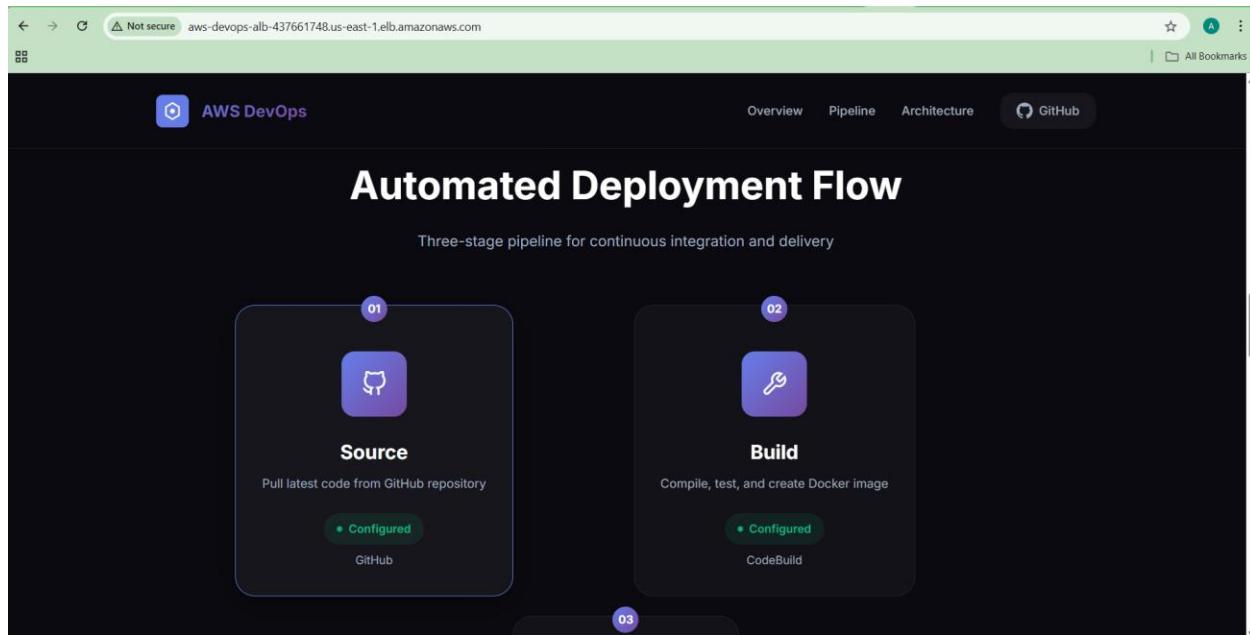
Application URL: http://aws-devops-alb-437661748.us-east-1.elb.amazonaws.com
Pipeline Console: https://us-east-1.console.aws.amazon.com/codesuite/codepipeline/pipelines/aws-devops-pipeline/view
ECS Console: https://us-east-1.console.aws.amazon.com/ecs/home?region=us-east-1#/clusters/aws-devops-cluster/services
```

RUNNING APPLICATION SCREENSHOT:

The screenshot shows the AWS DevOps landing page. At the top, there's a navigation bar with links for Overview, Pipeline, Architecture, and GitHub. A prominent button labeled "Production Ready" is visible. The main heading reads "Automated CI/CD Pipeline Built on AWS". Below the heading, a subtext states: "Enterprise-grade continuous integration and deployment pipeline using AWS CodePipeline, CodeBuild, ECR, and ECS Fargate. Infrastructure provisioned with Terraform for complete automation." Two buttons at the bottom are "View Pipeline" and "Architecture".

The screenshot shows the AWS DevOps landing page with a "Complete DevOps Solution" heading. It features four main components: "Automated Deployments" (push to deploy with zero manual intervention), "High Availability" (Multi-AZ deployment with auto-scaling), "Container Orchestration" (ECS Fargate for serverless containers), and "Infrastructure as Code" (Complete Terraform automation). A "SYSTEM OVERVIEW" button is located below these components.

The screenshot shows the AWS DevOps landing page with a focus on the CI/CD pipeline components. It includes "Source Control" (GitHub integration with webhook triggers), "Build Process" (AWS CodeBuild compiles code, runs tests, and creates Docker images with optimized caching), "Container Registry" (Amazon ECR stores Docker images with automated scanning and lifecycle management policies), and "Deployment" (ECS Fargate deploys containers with zero-downtime blue/green deployments and health monitoring). Each component has its own sub-sections for integration, trigger, service, and output.



Not secure aws-devops-alb-437661748.us-east-1.elb.amazonaws.com

AWS DevOps

Overview Pipeline Architecture GitHub

Monitoring & Logging

Comprehensive observability with CloudWatch for metrics, logs, and alerts to ensure system health

- ✓ CloudWatch Logs
- ✓ Container Insights
- ✓ Custom metrics
- ✓ Automated alerts

Container Registry

Private Docker registry with automated vulnerability scanning and lifecycle management for security

- ✓ Private repository
- ✓ Image scanning
- ✓ Lifecycle policies
- ✓ Encryption at rest

Infrastructure as Code

Complete infrastructure automation using Terraform for consistent and repeatable deployments

- ✓ Terraform modules
- ✓ Version controlled
- ✓ Automated provisioning
- ✓ State management

TECHNOLOGY STACK

Built With Modern Tools

Not secure aws-devops-alb-437661748.us-east-1.elb.amazonaws.com

AWS DevOps

Overview Pipeline Architecture GitHub

AWS DevOps Pipeline

Production-ready CI/CD infrastructure built with AWS services and Terraform for automated deployments.

Quick Links

- GitHub Repository
- AWS CodePipeline
- Terraform Docs

Resources

- Overview
- Pipeline
- Architecture

© 2025 Amitabh. Built with ❤ for DevOps

GitHub • LinkedIn