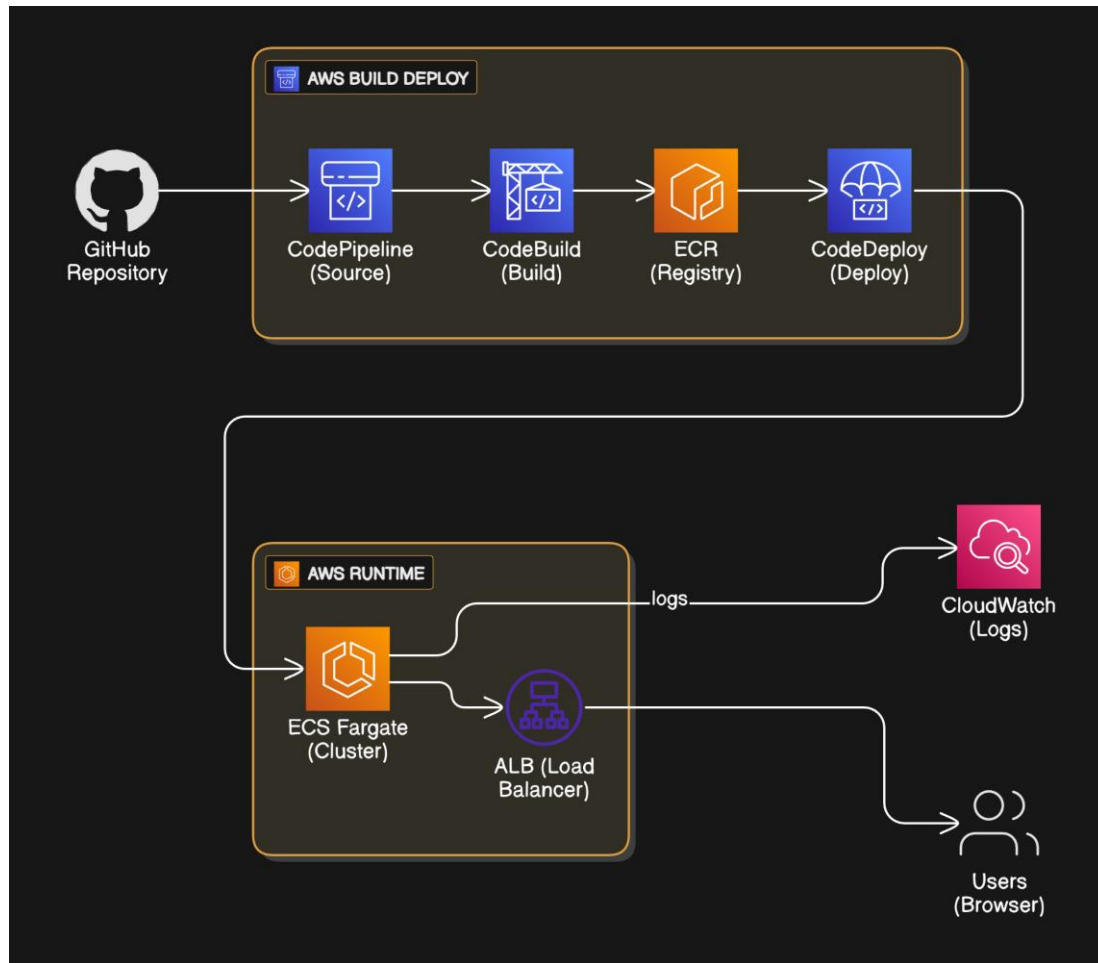


# AWS DEVOPS CI/CD PIPELINE

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 [↗](#)

**Managed signing now available**  
 Automatically sign your container images upon push to verify authenticity and ensure supply chain security. [Configure image signing](#)

**Private repositories (1)**

Search by repository substring

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 > Clusters > aws-devops-cluster > Services

**aws-devops-cluster**

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

**Cluster overview**

ARN: arn:aws:ecs:us-east-1:731733338203:cluster/aws-devops-cluster

Status: Active

CloudWatch monitoring: Container Insights | [View in CloudWatch](#)

Registered container instances: -

**Services**

Draining: - Active: 1 Pending: - Running: 2

**Services (1)**

Filter services by value

Filter launch type: Any launch type

Filter scheduling strategy: Any scheduling strategy

Filter resource management type: Any resource management type

Service name	ARN	Status	Schedu...	Lau...	Task de...	Deployments and tasks
aws-devops-service	arn:aws:ecs:us-e	<span style="color: green;">Active</span>	REPLICA	FARG...	aws-devo...	2/2 tasks

CloudShell Feedback Console Mobile App

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## 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.

Filter load balancers

Name	State	Type	Scheme	IP address type	VPC ID	Availability Zones
aws-devops-alb	<span style="color: green;">Active</span>	application	Internet-facing	IPv4	vpc-0bc3a0b56256db8bb	2 Availability Zones

**Load balancer: aws-devops-alb**

[Details](#) | [Listeners and rules](#) | [Network mapping](#) | [Resource map](#) | [Security](#) | [Monitoring](#) | [Integrations](#) | [Attributes](#) | [Capacity](#)

**Details**

Load balancer type: Application

Status: Active

VPC: [vpc-0bc3a0b56256db8bb](#)

Scheme: Internet-facing

Hosted zone: [Z355XD0T807X7K](#)

Availability Zones: [subnet-0f31c618c30fe1de0](#) [us-east-](#)

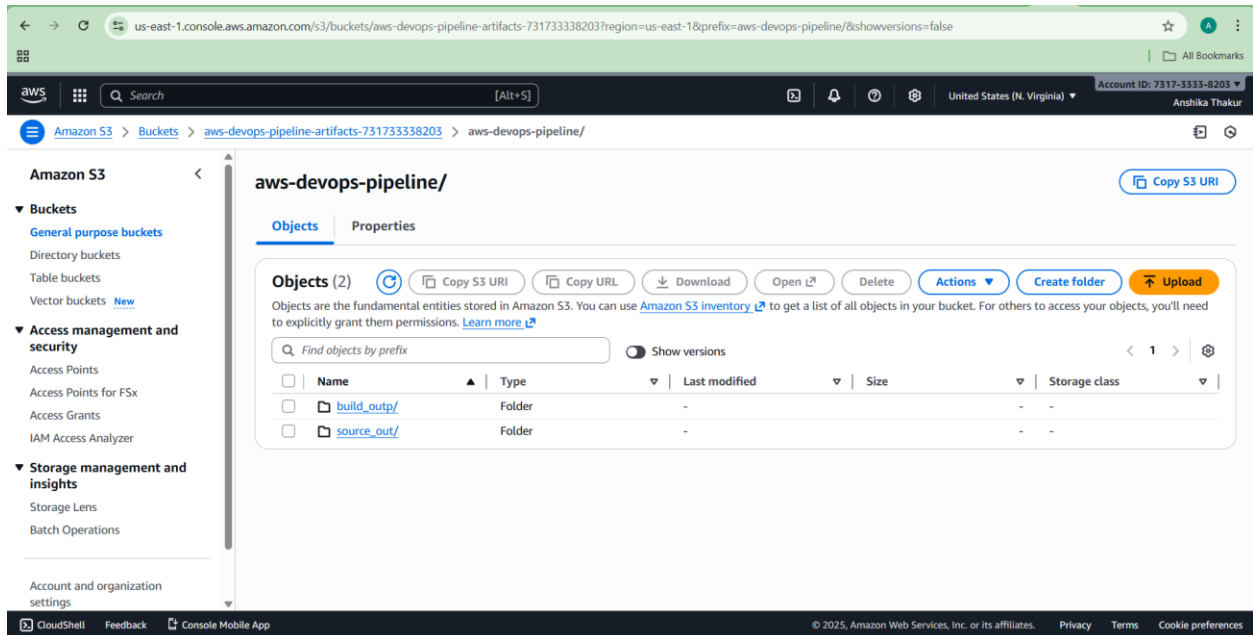
Load balancer IP address type: IPv4

Date created: December 12, 2025, 12:17 (UTC-05:00)

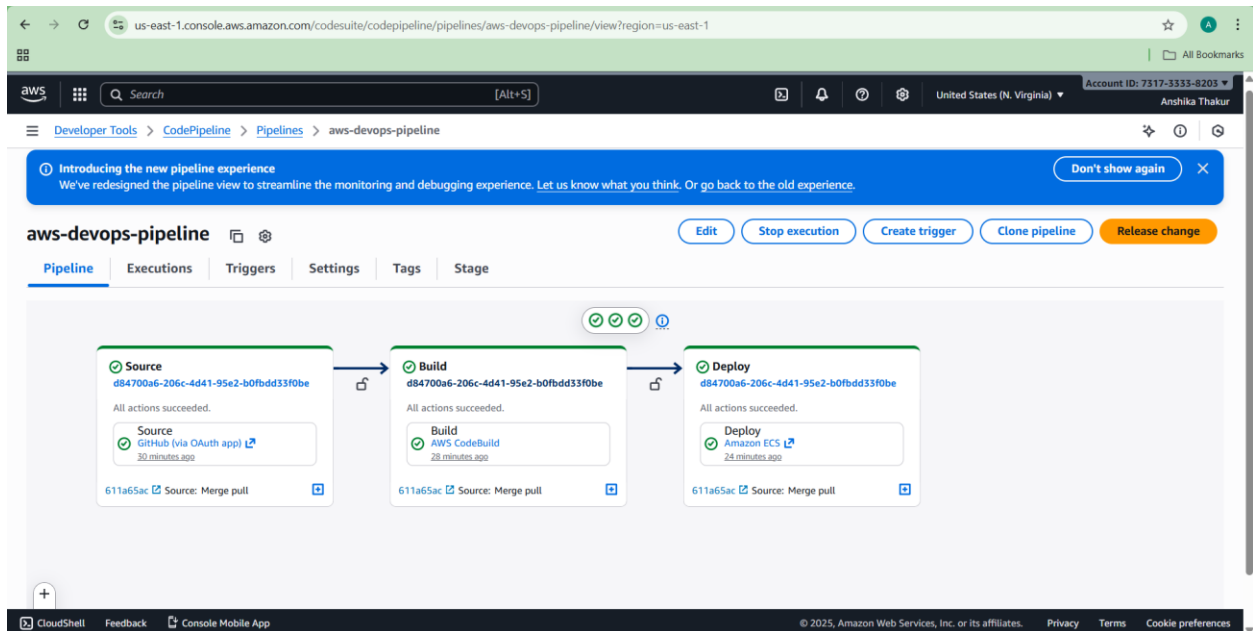
CloudShell Feedback Console Mobile App

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## S3 BUCKET:



## PIPELINE DEPLOYMENT



## 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 image displays three sequential screenshots of the AWS DevOps console, illustrating the setup and components of an automated CI/CD pipeline.

**Screenshot 1: Automated CI/CD Pipeline Built on AWS**

The top screenshot shows the main landing page of the AWS DevOps console. The header includes the AWS DevOps logo and navigation links for Overview, Pipeline, Architecture, and GitHub. A "Production Ready" status indicator is visible. The main heading is "Automated CI/CD Pipeline Built on AWS". Below this, a description 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 are present: "View Pipeline" and "Architecture".

**Screenshot 2: Complete DevOps Solution**

The middle screenshot shows the "SYSTEM OVERVIEW" section. It features four key capabilities:

- Automated Deployments:** Push to deploy with zero manual intervention.
- High Availability:** Multi-AZ deployment with auto-scaling.
- Container Orchestration:** ECS Fargate for serverless containers.
- Infrastructure as Code:** Complete Terraform automation.

Below these capabilities is a "SYSTEM OVERVIEW" button and the heading "Complete DevOps Solution".

**Screenshot 3: Detailed Pipeline Components**

The bottom screenshot provides a detailed breakdown of the pipeline components:

- Source Control:** GitHub Integration with webhook triggers for automatic pipeline execution on code changes.

INTEGRATION	TRIGGER
GitHub	Webhook
- Build Process:** AWS CodeBuild compiles code, runs tests, and creates Docker images with optimized caching.

SERVICE	OUTPUT
CodeBuild	Docker Image
- Container Registry:** Amazon ECR stores Docker images with automated scanning and lifecycle management policies.

SERVICE	SECURITY
ECR	Image Scan
- Deployment:** ECS Fargate deploys containers with zero-downtime blue/green deployments and health monitoring.

