

Cloud Deployment with Automation

Introduction

This case study explores the concepts of AWS CodePipeline, EC2 and S3. In this we build a simple HTML WebApp using AWS CodeBuild, and automatically upload it to an S3 bucket to then automatically deploy it to an EC2 instance using CodeDeploy.

The tools and concepts used for this case study are :-

AWS CodeBuild

Key Features:

Build Automation: Fully managed service that compiles source code, runs tests, and produces software packages.

Custom Build Environments: Supports Docker images for custom build environments.

Scalability: Automatically scales to meet build demand.

Pay-as-you-go Pricing: You only pay for the compute resources you use.

Practical Uses:

Continuous Integration (CI) for automated testing and building of applications.

Integrating with other AWS services for a seamless development pipeline.

Building and packaging applications for deployment.

AWS CodePipeline

Key Features:

Continuous Delivery: Automates the software release process using defined workflows.

Integration with Other AWS Services: Works seamlessly with CodeBuild, CodeDeploy, and third-party tools.

Customizable Workflows: Easily define stages for building, testing, and deploying applications.

Practical Uses:

Automating the release process from code commit to deployment.

Creating pipelines for microservices or multi-environment setups.

Enabling rapid and reliable application delivery.

Amazon S3 (Simple Storage Service)

Key Features:

Scalable Storage: Virtually unlimited storage capacity.

Durability and Availability: Designed for 99.999999999% durability and high availability.

Security Features: Supports access control, encryption, and versioning.

Practical Uses:

Storing build artifacts and deployment packages.

Hosting static websites and serving assets for web applications.

Backup and archival storage.

Amazon EC2 (Elastic Compute Cloud)

Key Features:

Flexible Computing: Provides resizable compute capacity in the cloud.

Variety of Instance Types: Different instance types for various workloads.

Auto Scaling: Automatically adjusts capacity based on demand.

Practical Uses:

Hosting applications and services in a scalable manner.

Running batch processing and data analytics workloads.

Deploying web applications or back-end services.

AWS CodeDeploy

Key Features:

Automated Deployments: Automatically deploys applications to EC2, Lambda, or on-premises servers.

Blue/Green Deployments: Reduces downtime and risks during application updates.

Monitoring and Rollback: Monitors deployment status and can roll back if issues are detected.

Practical Uses:

Managing updates and deployments for applications running on EC2 instances.

Facilitating microservices deployments with minimal downtime.

Integrating with CI/CD pipelines to automate the deployment process.

Steps

Creating an S3 Bucket

General configuration

AWS Region
US East (N. Virginia) us-east-1

Bucket type [Info](#)

☒ **General purpose**
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

☐ **Directory**
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name [Info](#)

jai-61

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

[Choose bucket](#)

Format: s3://bucket/prefix

Allowing public access


☐ **Block all public access**
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

☐ **Block public access to buckets and objects granted through new access control lists (ACLs)**
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

☐ **Block public access to buckets and objects granted through any access control lists (ACLs)**
S3 will ignore all ACLs that grant public access to buckets and objects.

☐ **Block public access to buckets and objects granted through new public bucket or access point policies**
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

☐ **Block public and cross-account access to buckets and objects through any public bucket or access point policies**
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

 **Turning off block all public access might result in this bucket and the objects within becoming public**
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

☒ I acknowledge that the current settings might result in this bucket and the objects within becoming public.

Enabling static website hosting

Static website hosting

Use this bucket to host a website or redirect requests. [Learn more](#)


[Edit](#)

S3 static website hosting
Enabled

Hosting type
Bucket hosting

Bucket website endpoint

When you configure your bucket as a static website, the website is available at the AWS Region-specific website endpoint of the bucket. [Learn more](#)

 <http://jai-61.s3-website-us-east-1.amazonaws.com>

Creating a Build project

Create build project

Project configuration

Project name

jai-61

A project name must be 2 to 255 characters. It can include the letters A-Z and a-z, the numbers 0-9, and the special characters - and _.

Public build access - *optional*

Public build access allows you to make the build results, including logs and artifacts, for this project available for the general public.

☒ Enable public build access**Public build access enabled**

Your build results, including logs and artifacts, are accessible to the general public. Downloading logs and/or artifacts will increase your AWS costs. [Learn more](#)

Public build service role

The public build service role is used to provide read access to your logs and artifacts for public builds. You can let CodeBuild create a new role, or you can choose an existing role.



New service role

Create a service role in your account



Existing service role

Choose an existing service role from your account

Using existing service role and allowing AWS to modify it

Public build service role

The public build service role is used to provide read access to your logs and artifacts for public builds. You can let CodeBuild create a new role, or you can choose an existing role.



New service role

Create a service role in your account



Existing service role

Choose an existing service role from your account

Service role

arn:aws:iam::311141519473:role/service-role/codebuild- X

☒ Allow AWS CodeBuild to modify this service role so it can be used with this build project

arn:aws:iam::311141519473:role/service-role/codebuild-jai-61-service-role

► Additional configuration

Description, Build badge, Concurrent build limit, tags

Configuring source for CodeBuild project as a GitHub repository

Source 1 - Primary

Source provider

GitHub

Credential

☐ Default source credential
Use your account's default source credential to apply to all projects

☒ Custom source credential
Use a custom source credential to override your account's default settings

Credential type

☒ GitHub App
Connect project to GitHub using an AWS managed GitHub App

☐ OAuth app
Connect project to GitHub using an OAuth app

☐ Personal access token
Connect project to GitHub using a personal access token

Connection

You can [create a new GitHub connection](#) by using an AWS managed GitHub App

arn:aws:codeconnections:us-east-1:311141519473:connection/ X

Repository

☒ Repository in my GitHub account

☐ Public repository

☐ GitHub scoped webhook

Repository

https://github.com/D15A-Jai-61/ADO X

Using an Ubuntu instance to build the WebApp

Environment image

☒ Managed image
Use an image managed by AWS CodeBuild

☐ Custom image
Specify a Docker image

Compute

☒ EC2
Optimized for flexibility during action runs

☐ Lambda
Optimized for speed and minimizes the start up time of workflow actions

Operating system

Ubuntu

Runtime(s)

Standard

Image

aws/codebuild/standard:7.0

Image version

Always use the latest image for this runtime version

Using the BuildSpec file in the GitHub repository

Buildspec

Build specifications

☐ Insert build commands
Store build commands as build project configuration

☒ Use a buildspec file
Store build commands in a YAML-formatted buildspec file

Buildspec name - optional

By default, CodeBuild looks for a file named buildspec.yml in the source code root directory. If your buildspec file uses a different name or location, enter its path from the source root here (for example, buildspec-two.yml or configuration/buildspec.yml).

buildspec.yml

Configuring the created S3 bucket to store the built project files

Artifacts

Add artifact

Artifact 1 - Primary

Type

Amazon S3 ▼

You might choose no artifacts if you are running tests or pushing a Docker image to Amazon ECR.

Bucket name

Q jai-61 X

Build project created

Developer Tools > CodeBuild > Build projects > jai-61

jai-61

Actions ▼ Create trigger Edit Clone Debug build Start build with overrides Start build

Configuration

Source provider GitHub	Primary repository D15A-Jai-61/ADO 🔗	Artifacts upload location jai-61	Service role arn:aws:iam::311141519473:role/service-role/codebuild-jai-61-service-role
Public builds Enabled	Public project URL 🔗 Go to public project 🔗		

Build process succeeded and finished

Developer Tools > CodeBuild > Build projects > jai-61 > jai-61:36120006-8118-4486-aed8-419ac0fbb27f

jai-61:36120006-8118-4486-aed8-419ac0fbb27f

Stop build

Retry build

Build status

Status	Initiator	Build ARN	Resolved source version
<div><div></div>Succeeded</div>	root	<div><div></div>arn:aws:codebuild:us-east-1:311141519473:build/jai-61:36120006-8118-4486-aed8-419ac0fbb27f</div>	7fb191ad0723de0e6cda3e22017822e476c1f5b9
Start time	End time	Build number	
Oct 23, 2024 10:00 PM (UTC+5:30)	Oct 23, 2024 10:01 PM (UTC+5:30)	1	

Success

jai-61:36120006-8118-4486-aed8-419ac0fbb27f

Build status

Status

Succeeded

Start time

Oct 23, 2024 10:00 PM (UTC+5:30)

The built project files have been successfully uploaded to the created S3 bucket

jai-61

Info

Objects

Properties

Permissions

Metrics

Management

Access Points

Objects (2)

Info

Copy S3 URI

Copy URL

Download

Open

Delete

Actions

Create folder

Upload

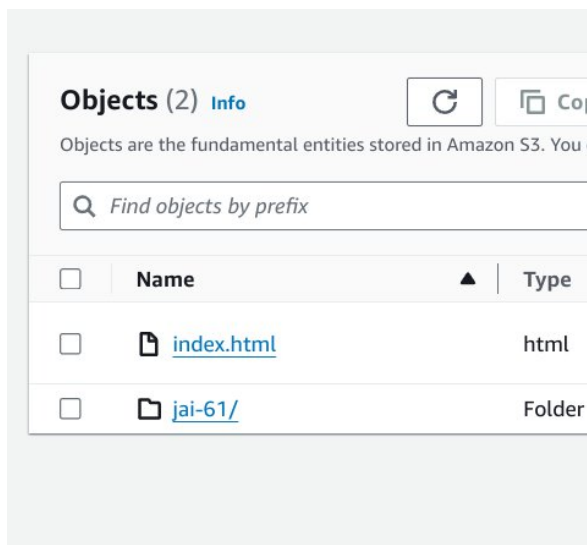
Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Find objects by prefix

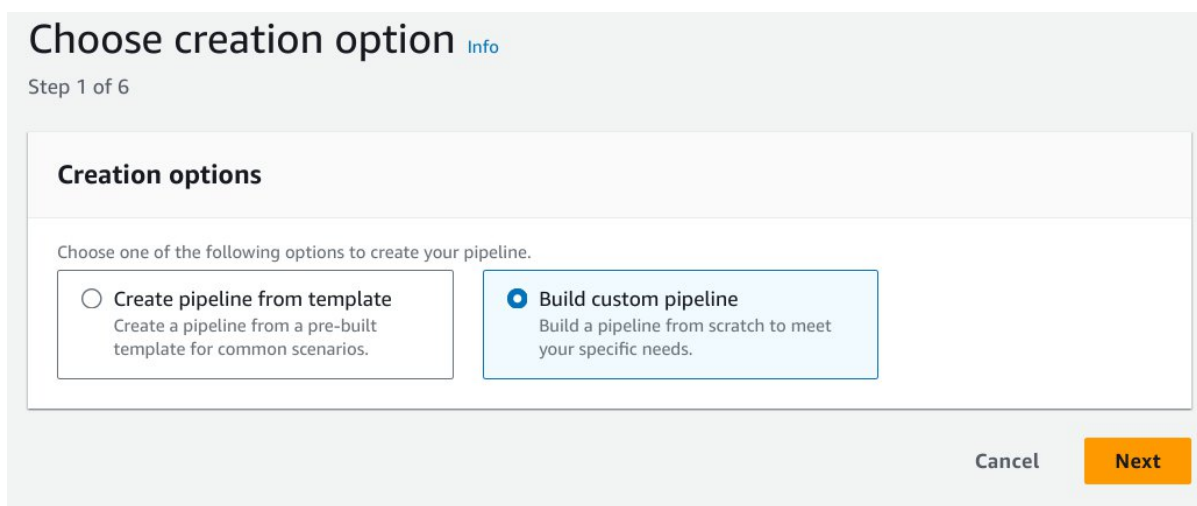
< 1 >

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	<div><div></div>index.html</div>	html	October 23, 2024, 22:01:20 (UTC+05:30)	146.0 B	Standard
<input type="checkbox"/>	<div><div></div>jai-61/</div>	Folder	-	-	-

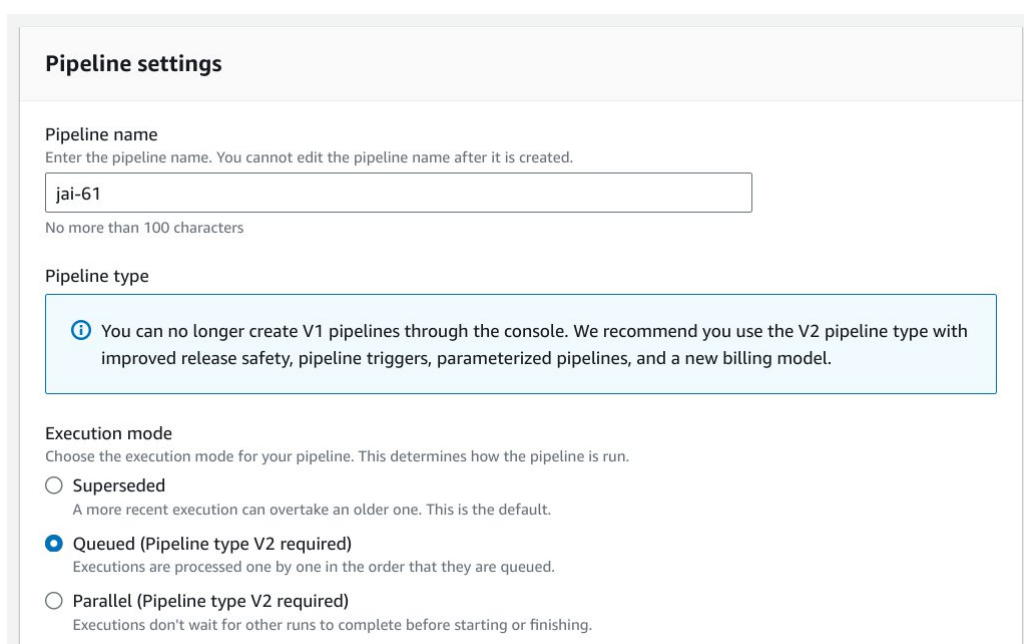
All built project files



Creating a custom Pipeline to deploy the built project



Pipeline settings



Using a new service role for the Pipeline

Service role



New service role

Create a service role in your account



Existing service role

Choose an existing service role from your account

Role name

AWSCodePipelineServiceRole-us-east-1-jai-61

Type your service role name

- ☒ Allow AWS CodePipeline to create a service role so it can be used with this new pipeline

Configuring the source for the Pipeline as the same repository as the CodeBuild project (the issue here is, even though the connection ID and repository link along with the branch name have been specified correct, the service isn't able to find it)

Source

Source provider

This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.

GitHub (Version 2)



New GitHub version 2 (app-based) action

To add a GitHub version 2 action in CodePipeline, you create a connection, which uses GitHub Apps to access your repository. Use the options below to choose an existing connection or create a new one. [Learn more](#)

Connection

Choose an existing connection that you have already configured, or create a new one and then return to this task.



arn:aws:codeconnections:us-east-1:311141519473:connection/7e82bf50-91



or

[Connect to GitHub](#)

Repository name

Choose a repository in your GitHub account.


https://github.com/D15A-Jai-61/ADO



An unspecified error occurred. Check your network connectivity, and then check to see if there are any issues with the service at the [Service Health Dashboard](#). [\(Click here to retry\)](#)


You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

Unable to find the source even though correctly typed

 An unspecified error occurred. Check your network connectivity, and then check to see if there are any issues with the service at the [Service Health Dashboard](#). (Click here to retry)

You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

Default branch
Default branch will be used only when pipeline execution starts from a different source or manually started.

 Not Found (Click here to retry)

Output artifact format
Choose the output artifact format.

☒ **CodePipeline default**
AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include Git metadata about the repository.

☐ **Full clone**
AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full Git clone. Only supported for AWS CodeBuild actions.

☒ Enable automatic retry on stage failure

Trigger


Pipeline created but failed execution due to above mentioned issue

Developer Tools > CodePipeline > Pipelines > SimpleDockerService

SimpleDockerService

Notify Edit Stop execution Clone pipeline Release change


Pipeline type: V2 Execution mode: QUEUED

 **Source** Failed

Retry stage Retry failed actions

Pipeline execution ID: [90d91d22-8f92-4fb7-8ef3-2be104e9c691](#)

CodeConnections
[GitHub \(Version 2\)](#)

 Failed - 49 minutes ago
 View details

Execution details include the error message that specifies the above mentioned issue

Action execution details

Action name: CodeConnections Status: Failed

Summary

Input

Status

⊗ Failed

Last updated

50 minutes ago

Action execution ID

712fc602-12ec-4d95-8d87-d523b6d30543

Error code

Action execution failed

Error message

[GitHub] No Branch [main] found for FullRepositoryName [https://github.com/D15A-Jai-61/ADO]

Done

Conclusion

The case study performance was unsuccessful due to presumably a server side issue from AWS, not from performed steps being incorrect.

The project was built successfully and uploaded to S3 bucket automatically successfully, but not deployed to EC2 instance successfully.