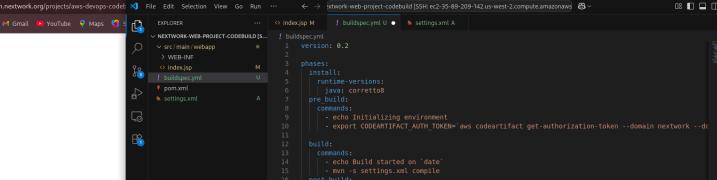




Continuous Integration with CodeBuild



Kiprono Yegon



The screenshot shows a browser window with several tabs open. The active tab is titled 'NextWork - Continuous Int...'. Below the tabs, there's a navigation bar with links like 'File', 'Edit', 'Selection', 'View', 'Run', 'Help', and 'More'. The main content area displays a Jenkins pipeline configuration for a 'nextwork-web-project' job. The pipeline script is as follows:

```
version: 2.0
phases:
  install:
    steps:
      - run: ./gradlew :main:dependencies
      - run: ./gradlew :main:correction
  pre-build:
    commands:
      - echo Initializing environment
      - export CODEARTIFACT_AUTH_TOKEN=$(aws codeartifact get-authentication-token --domain nextwork --region us-west-2)
  build:
    commands:
      - echo Build started on `date`
      - mvn --settings settings.xml compile
      - mvn --settings settings.xml package
    commands:
      - echo Build completed on `date`
      - mvn --settings settings.xml package
  artifacts:
    files:
      - target/nextwork-web-project.war
  discard-paths:
    paths: no
```

Below the script, there are tabs for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL', and 'PORTS'. The 'TERMINAL' tab shows the command 'git push' being run. The output of the terminal shows the build success message and the command being run.



Introducing Today's Project!

In this project, I will demonstrate how to use AWS CodeBuild to automate the build process in the CI/CD pipeline. Building means compressing the app's files into a single compressed file that can be deployed later.

Key tools and concepts

Services I used were. Amazon EC2, IAM, CodeArtifact, CCodeBuild, CodeConnections,S3 and Github. Key concepts I learnt include build process git issues resolving using the terminal, .buildspecyml and using CodeConnections.

Project reflection

This project took me approximately a whole day considering the various errors that occurred. The most challenging part was resolving the errors and github connection with CodeBuild. It was most rewarding to finally see success status of the codebuild.

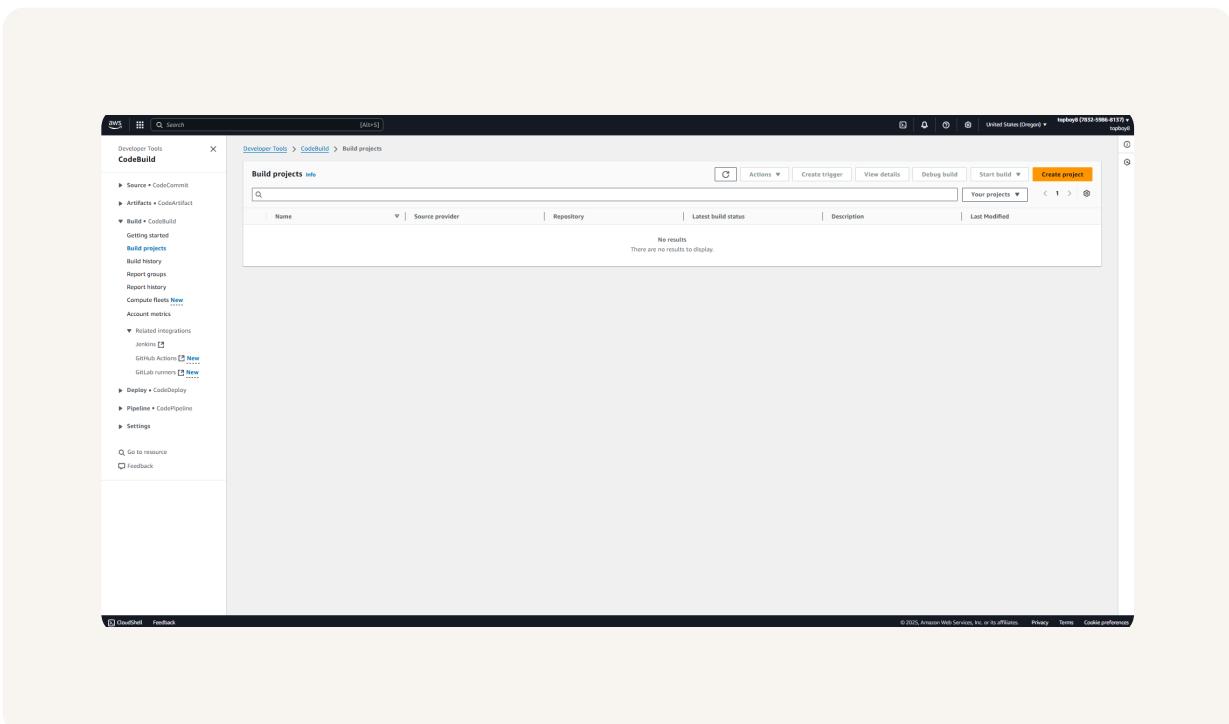
This project is part four of a series of DevOps projects where I'm building a CI/CD pipeline! I'll be working on the next project The day tomorrow.



Setting up a CodeBuild Project

CodeBuild is a continuous integration service, which means it helps you catch and fix issues early and often. Engineering teams use it because you don't manually set up and manage any build servers yourself, and you only pay for the compute time.

My CodeBuild project's source configuration means the location of the code that CodeBuild will fetch, compile, and package into a file you can deploy and I selected GitHub as source provider because that's where I've stored our web app's code.

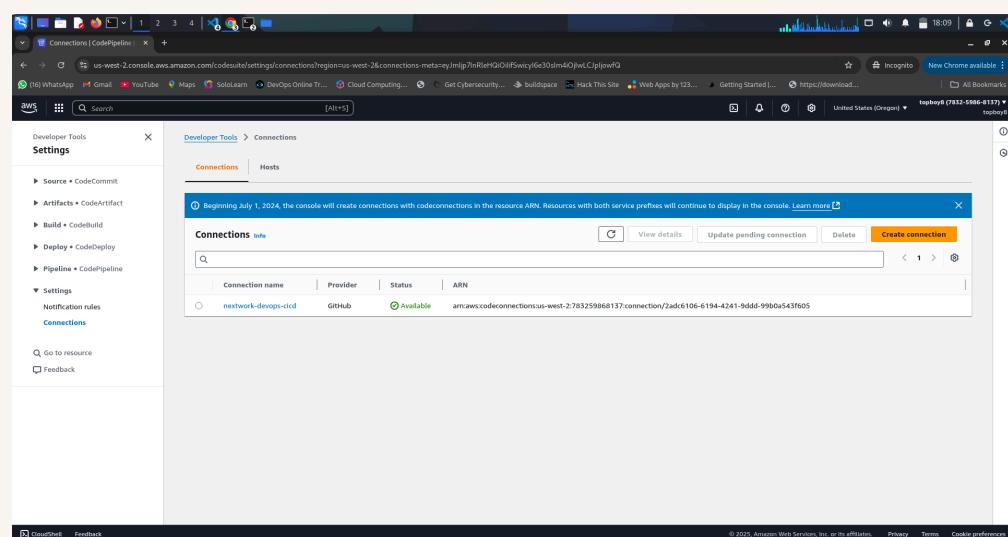




Connecting CodeBuild with GitHub

There are multiple credential types for GitHub, like Personal access token and OAuth app but I used GitHub App because it is generally the simplest and most secure option. AWS manages the application and connection, reducing the need for token key

The service that helped connect AWS and the external code repositories is CodeConnections





CodeBuild Configurations

Environment

My CodeBuild project's Environment configuration means that the environment set up for the compute power that will be compressing and compiling the files when we run our CodeBuild project it includes settings like provisioning model , compute, OS.

Artifacts

Build artifacts are tangible outputs of the build process. They're important because they are what you deploy to your servers or distribute to users. My build process creates the app To store them, I created an S3 bucket that will store artifacts

Packaging

When setting up CodeBuild, I also chose to package artifacts in a zip file because zipping reduces the size of the artifacts, which means faster uploads to S3, less storage costs, and quicker downloads when you need to use them.



Monitoring

For monitoring, I enabled CloudWatch Logs, a monitoring service that collects and tracks logs from AWS , it records everything that happens during the build process, including the commands that are run, the output of those commands and errors therein



buildspec.yml

My first build failed because Codebuild did not find the buildspec.yml file in the github repository. A buildspec.yml file is needed because CodeBuild doesn't know how to build your project and the buildspec.yml dictates how the project is built.

The first two phases in my buildspec.yml file tells the version and stages. The third phase in my buildspec.yml file installs dependencies and grabs security token. The fourth phase in my buildspec.yml file saves the output of the build file

```
version: 0.2
phases:
  install:
    runtime-versions:
      java: corretto8
  pre_build:
    commands:
      - echo Initializing environment
      - export CODEARTIFACT_AUTH_TOKEN=$(aws codestarartifact get-authorization-token --domain nextwork --dc)
  build:
    commands:
      - echo Build started on `date`
      - mvn settings.xml compile
  post_build:
    commands:
      - echo Build completed on `date`
      - mvn settings.xml package
artifacts:
  files:
    - target/nextwork-web-project.war
discard-paths: no
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 01:06 min
[INFO] Finished at: 2025-09-14T14:42:35Z
[INFO] -----
[INFO]
[INFO] * [ec2-user@ip-172-31-34-186 nextwork-web-project-codebuild]$ git push
[INFO]
[INFO] * [ec2-user@ip-172-31-34-186 nextwork-web-project-codebuild]$ git push
[INFO] Everything up-to-date
[INFO] * [ec2-user@ip-172-31-34-186 nextwork-web-project-codebuild]$
```

SSH 2025-09-14T14:42:35Z 172.31.34.186 west-2.compute.amazonaws.com

A circular profile picture of Kiprono Yegon, featuring a dark background with glowing blue and green energy-like patterns.

Success!

My second build also failed, but with a different error that said build and post-build failed. To fix this I will have to add a settings.xml file to the project. This is because the CodeBuild doesn't have the permission to access CodeArtifact.

To resolve the second error, a new policy service role that CodeBuild created when we set up our CodeBuild was attached to the role When I re-built the project again, it rebuilt with a Success status, Hurrayyyy, the project built with a success

To verify the build, I checked if the build artifact was correctly uploaded to our S3 bucket. Seeing the artifact tells me that the code was successfully compiled and packaged, the build process was completed without errors, the artifact was uploaded



Kiprono Yegon
NextWork Student

nextwork.org

The screenshot shows the AWS CodeBuild console with a successful build summary. The main title is "Build started" with a green success icon. Below it, the build ID is "nextwork-devops-cicd:4c1ede3a-26b3-4cfa-99a1-99aa173f082a". The status is "Succeeded". The initiator is "topboy0". The build ARN is "arn:aws:codebuild:us-west-2:783259868137:build/nextwork-devops-cicd:4c1ede3a-26b3-4cfa-99a1-99aa173f082a". The resolved source version is "7a015aef5a062ee10cbdbab52e9255244d65c0173f082a". The start time is "Aug 14, 2025 7:35 PM (UTC+5:00)" and the end time is "Aug 14, 2025 7:34 PM (UTC+5:00)". The build number is "3". The build logs section shows "No logs available for this build" and a "Tail logs" button. The left sidebar lists various AWS services like CodeCommit, CodeArtifact, CodeBuild, CodeDeploy, CodePipeline, and Jenkins.



nextwork.org

The place to learn & showcase your skills

Check out nextwork.org for more projects

