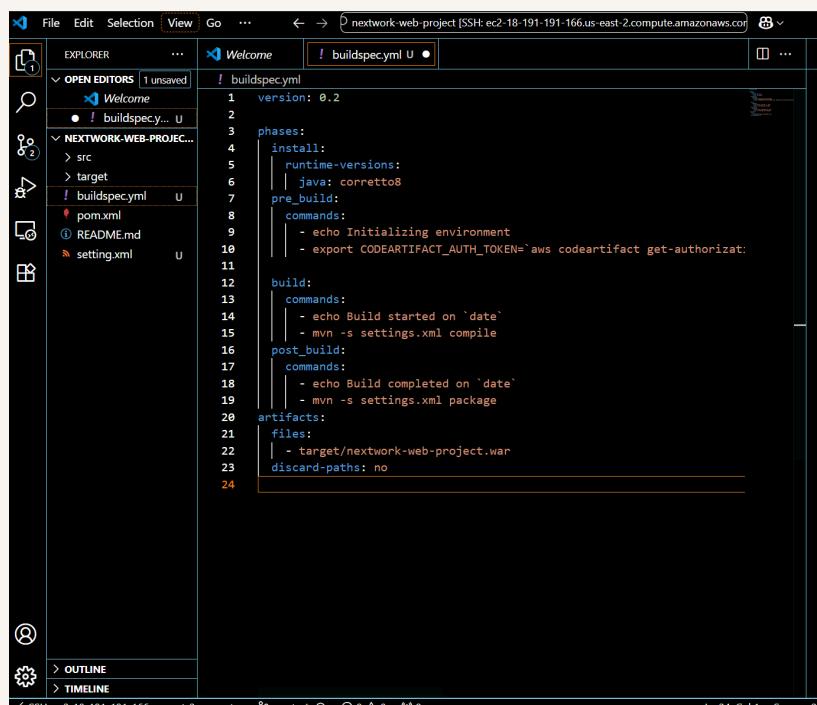




Continuous Integration with CodeBuild



naveen msd



```
File Edit Selection View Go ... ← →  nextwork-web-project [SSH: ec2-18-191-191-166.us-east-2.compute.amazonaws.com] ...  
EXPLORER ... Welcome buildspec.yml U ...  
OPEN EDITORS 1 unsaved  
Welcome buildspec... U  
NEXTWORK-WEB-PROJECT...  
> src  
> target  
buildspec.yml U  
pom.xml README.md setting.xml U  
OUTLINE  
TIMELINE  
buildspec.yml  
1 version: 0.2  
2  
3 phases:  
4 | install:  
5 | | runtime-versions:  
6 | | | java: corretto8  
7 | pre_build:  
8 | | commands:  
9 | | | - echo Initializing environment  
10 | | | - export CODEARTIFACT_AUTH_TOKEN=`aws codeartifact get-authorizat:  
11 | build:  
12 | | commands:  
13 | | | - echo Build started on `date`  
14 | | | - mvn -s settings.xml compile  
15 | post_build:  
16 | | commands:  
17 | | | - echo Build completed on `date`  
18 | | | - mvn -s settings.xml package  
19 artifacts:  
20 | files:  
21 | | - target/nextwork-web-project.war  
22 | discard-paths: no  
23  
24
```



Introducing Today's Project!

In this project, I will create and configure a CodeBuild project from scratch. Connect your CodeBuild project to your GitHub repository. Define your build process using a buildspec.yml file. Automate testing using CodeBuild.

Key tools and concepts

Services I used were AWS code build, IAM roles adn permissions, Key concepts I learnt was on configuring the code build, providing the nesseccary permission to access code artifact and ready for deployment.

Project reflection

This project took me approximately 2hrs The most challenging part was troubleshooting the build errors. It was most rewarding to see the build was successful.

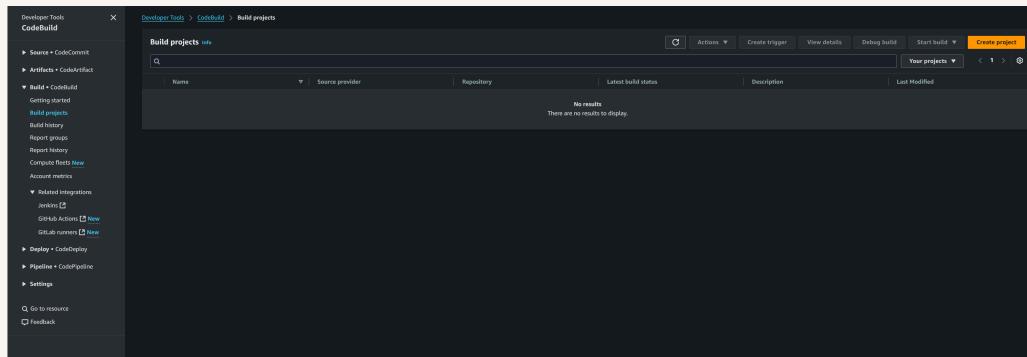
This project is part four of a series of DevOps projects where I'm building a CI/CD pipeline! I'll be working on the next project tomo



Setting up a CodeBuild Project

Continuous Integration is like having a quality control checkpoint that automatically kicks in whenever anyone on the team makes changes to the code.

My CodeBuild project's source configuration means basically telling where the project code lives in my case I have the code in the git hub and I selected that to tell AWS code build its over there.





Connecting CodeBuild with GitHub

This is generally the simplest and most secure option. AWS manages the application and connection, reducing the need for you to handle tokens or keys directly.

The service that helped connect to my GitHub. Instead of dealing with the headache of managing API keys, tokens or SSH credentials, CodeConnections handles all that authentication complexity.

Source 1 - Primary

Source provider

GitHub

Credential

✓ Your account is successfully connected by using an AWS managed GitHub App. [Manage account credentials](#).

Use override credentials for this project only

Repository

Repository in my GitHub account

Public repository

GitHub scoped webhook

Repository

Q

C



CodeBuild Configurations

Environment

My CodeBuild project's Environment configuration on demand it includes settings like provisioning model, Environmental Image, Compute, Running mode, OS, Runtime, image, version and service role.

Artifacts

Build artifacts are the tangible outputs of my build process. That's why storing in S3 is Important. They're important. It's a highly reliable and scalable storage solution that's also in our AWS Environment.

Packaging

When setting up CodeBuild, I also chose to package artifacts in Zip packaging because Zip are much smaller in size as they compress larger files and it's simplicity nature.

Monitoring

For monitoring, I enabled CloudWatch Logs, which will record everything that happens during the build process, including the commands that are run, the output of those commands, and any errors that occur.



buildspec.yml

My first build failed because there was no buildspec.yml file in the git repo. A buildspec.yml file is needed because it's like a step-by-step instruction manual. It goes through each phase in order - install, pre_build, build, then post_build

The first two phases in my buildspec.yml file is prepare and build. The third phase in my buildspec.yml file... The fourth phase in my buildspec.yml file post_build, is the finishing touches after the main build is done.

```
File Edit Selection View Go ... ← → ⌘ nextwork web-project [SSH: ec2-18-191-191-166.us-east-2.compute.amazonaws.com] ⌘ ...  
EXPLORER ... Welcome buildspec.yml U  
OPEN EDITORS (1 unsaved)  
• ! buildspec_y...  
NEXTWORK-WEB-PROJECT...  
> src  
> target  
buildspec.yml U  
pom.xml  
README.md  
setting.xml U  
OUTLINE  
TIMELINE  
version: 0.2  
phases:  
install:  
| runtime-versions:  
| | java: corretto8  
pre_build:  
commands:  
| - echo Initializing environment  
| - export CODEARTIFACT_AUTH_TOKEN=$(aws codeartifact get-authorization-token --region us-east-2 --repository-name nextwork-web-project --profile default)  
build:  
commands:  
| - echo Build started on `date`  
| - mvn -s settings.xml compile  
post_build:  
commands:  
| - echo Build completed on `date`  
| - mvn -s settings.xml package  
artifacts:  
files:  
| target/nextwork-web-project.war  
discard-paths: no
```



Success!

My second build also failed, but with a different error that said mvn -s settings.xml is not accessible. To fix have to give IAM role permission to CodeBuild to access the code artifact.

To resolve the second error, I added permission to access codeartifact. When I built my project again, I saw the build was successful.

To verify the build, I checked the object in the S3 bucket. Seeing the artifact tells me that the build was successful, as there were no errors found and the artifact is available for download.

The screenshot shows the AWS CodeBuild console with a successful build log. The build ID is `nextwork-devops-cicd:af33506d-9e0e-4f04-aeb3-20269bb48ed5`. The build status is "Build started" and "Success". The build duration was 9 seconds. The build log table lists the following phases and their statuses:

Name	Status	Context	Duration	Start time	End time
SUBMITTED	Success	-	<1 sec	Sep 6, 2025 6:26 PM (UTC-7:00)	Sep 6, 2025 6:26 PM (UTC-7:00)
QUEUED	Success	-	<1 sec	Sep 6, 2025 6:26 PM (UTC-7:00)	Sep 6, 2025 6:26 PM (UTC-7:00)
PROVISIONING	Success	-	5 secs	Sep 6, 2025 6:26 PM (UTC-7:00)	Sep 6, 2025 6:26 PM (UTC-7:00)
DOWNGLOAD_SOURCE	Success	-	3 secs	Sep 6, 2025 6:26 PM (UTC-7:00)	Sep 6, 2025 6:26 PM (UTC-7:00)
INSTALL	Success	-	<1 sec	Sep 6, 2025 6:26 PM (UTC-7:00)	Sep 6, 2025 6:26 PM (UTC-7:00)
PRE_BUILD	Success	-	5 secs	Sep 6, 2025 6:26 PM (UTC-7:00)	Sep 6, 2025 6:26 PM (UTC-7:00)
BUILD	Success	-	16 secs	Sep 6, 2025 6:26 PM (UTC-7:00)	Sep 6, 2025 6:27 PM (UTC-7:00)
POST_BUILD	Success	-	30 secs	Sep 6, 2025 6:27 PM (UTC-7:00)	Sep 6, 2025 6:27 PM (UTC-7:00)
UPLOAD_ARTIFACTS	Success	-	<1 sec	Sep 6, 2025 6:27 PM (UTC-7:00)	Sep 6, 2025 6:27 PM (UTC-7:00)
FINALIZING	Success	-	<1 sec	Sep 6, 2025 6:27 PM (UTC-7:00)	Sep 6, 2025 6:27 PM (UTC-7:00)
COMPLETED	Success	-	-	Sep 6, 2025 6:27 PM (UTC-7:00)	-



nextwork.org

The place to learn & showcase your skills

Check out nextwork.org for more projects

