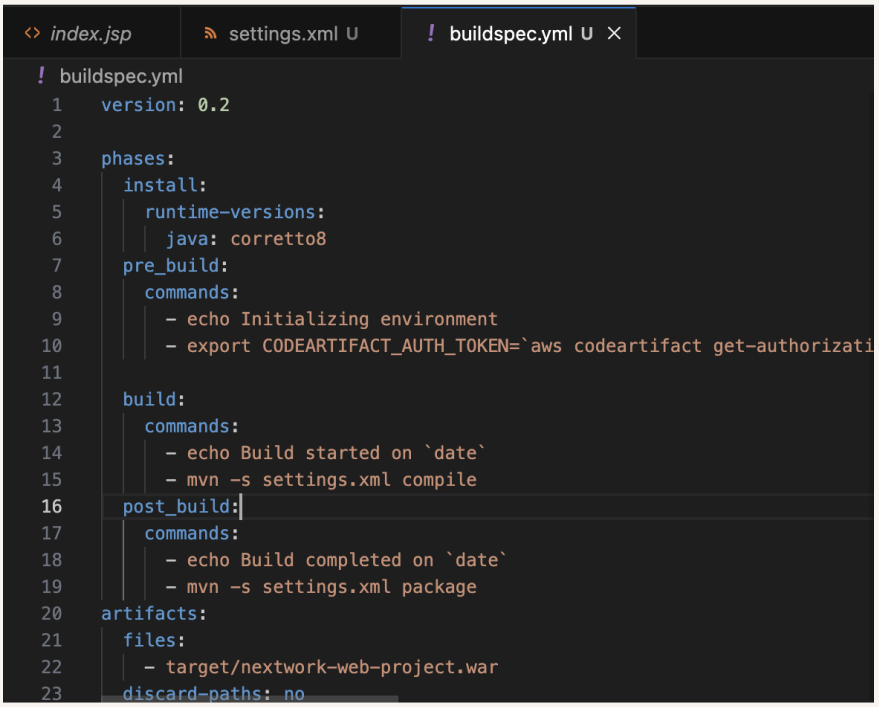
nextwork.org

Continuous Integration with CodeBuild

Joseph Cole-Showers



Joseph Cole-Showers
NextWork Student

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. i.e Compressing web app files into a single compress file like a zip file that I can later deploy.

Key tools and concepts

Services I used include: CodeBuild, CodeConnection, CodeArtifact, S3, EC2, Github and VS Code. Key concepts I learnt include the build process, buildspec.yml, using Codeconnections to connect AWS with Github.

Project reflection

This project took me approximately 2hrs to do including troubleshooting time. The most challenging part was resolving connections to Github from AWS.

This project is part four of a series of DevOps projects where I'm building a CI/CD pipeline.

Joseph Cole-Showers
NextWork Student

Setting up a CodeBuild Project

CodeBuild is a continuous integration service, which means it helps to compile and package code. It turns the web app code into something computers can run. Engineering teams use it because CI services automates this process.

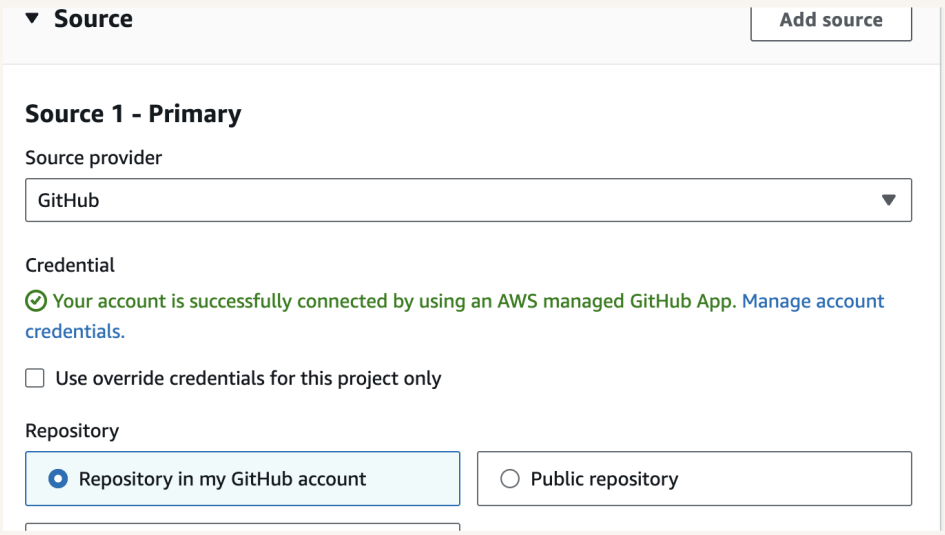
My CodeBuild project's source configuration means where the code lives and I selected GitHub. That is where the web app source code repo is stored.

Joseph Cole-Showers
NextWork Student

Connecting CodeBuild with GitHub

There are multiple credential types for GitHub, like GitHub App, Personal Access Tokens and OAuth app. I used GitHub App because it is the most simple, yet the most secure way between GitHub and AWS. AWS handles all of the credentials e.g the tokens

The service that helped connect is AWS CodeConnections. CodeConnections helps to connect environment with other third-party platforms.



J
Joseph Cole-Showers
NextWork Student

CodeBuild Configurations

Environment

My CodeBuild project's Environment configuration means the environment setup for the compute power i.e the EC2 instance that will be compiling and compressing the files when I run the CodeBuild project.

Artifacts

Build artifacts are files that get created as part of the build process. They're important because they represent artifacts that will be used later in the CI/CD pipeline. My build process will create a compressed file for the webapp. Stored it in S3.

Packaging

When setting up CodeBuild, I also chose to package artifacts into a zip file. This helps with package management as files are all organized neatly in a single executable file.

J
Joseph Cole-Showers
NextWork Student
NextWork.org

Monitoring

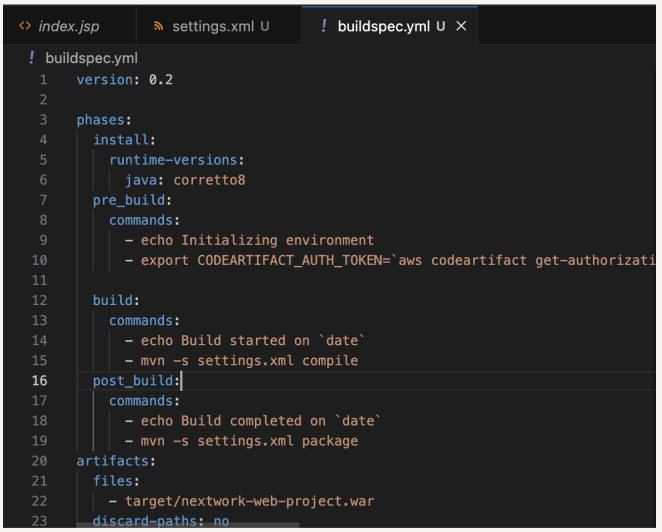
For monitoring, I enabled CloudWatch Logs, which is a service that will log commands that run and any errors that happen. This is helpful for troubleshooting.

Joseph Cole-Showers
NextWork Student

buildspec.yml

My first build failed because CodeBuild could not find the buildspec.yml file in the source code root directory. A buildspec.yml file is needed because it tells CodeBuild how to run the build process.

The first 2 phases in my buildspec.yml file installs Java and gets access to CodeArtifact. The third phase in my buildspec.yml file compiles the web app. The fourth phase in my buildspec.yml file packages the web app into a single compressed artifact



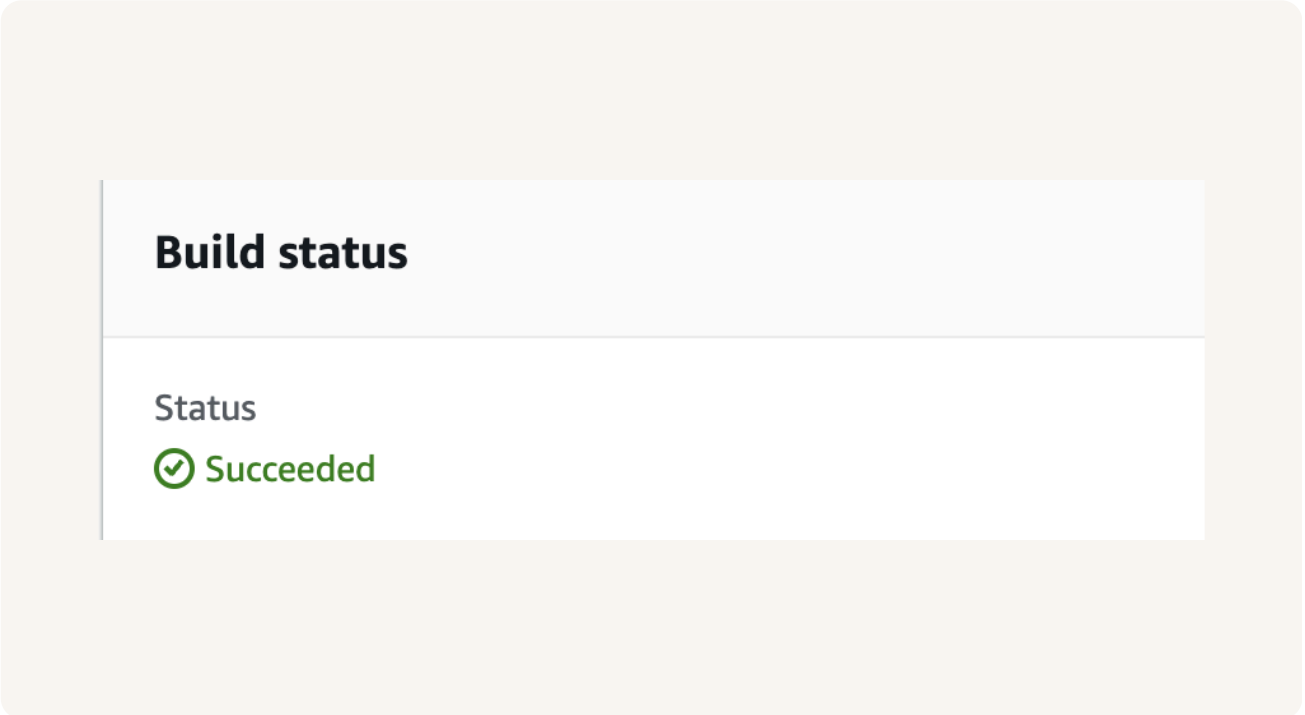
Joseph Cole-Showers
NextWork Student

Success!

My second build also failed, but with a different error that said I failed to run an error while executing the command 'mvn -s settings.xml' to compile the web app.

To resolve the second error, I attached a policy for access CodeArtifact to CodeBuild role. When I built my project again, I saw a success. With the priviledges to CodeArtifact granted by the role, Codebuild now has access to the repo.

To verify the build, I checked S3 bucket for the artifact. Seeing the artifact tells me that Codebuild was a success. The build process compiled the code from the source repo, compressed it to a file and stored the file in S3 bucket.



NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for more projects