

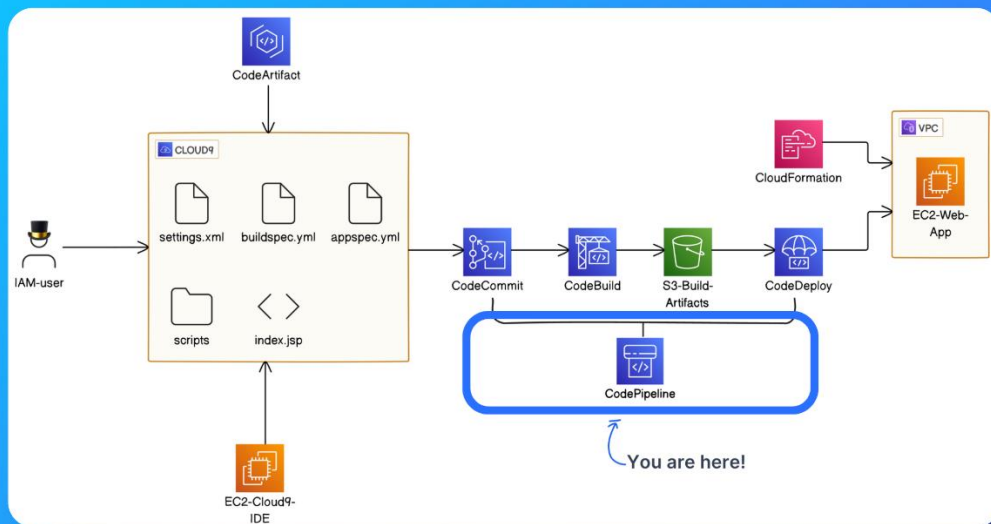


CI/CD with CodePipeline



Prajit Venkatachalam

<https://www.linkedin.com/in/prajit-venkatachalam/>





Introducing Today's Project!

What is AWS CodePipeline?

AWS CodePipeline is an AWS service that automates the steps DevOps or cloud engineers take to release application updates. With CodePipeline, I can automate the steps typically performed manually in CodeCommit, CodeBuild, and CodeDeploy as soon as I commit and push changes to the main branch. This makes the process faster and reduces the chances of errors.

How I used CodePipeline in this project

In this project, I used CodePipeline to create an automated pipeline using the CodeCommit, CodeBuild and CodeDeploy components created earlier in Step 2,4, and 5.

This project took me...

This Project took me 4 hours to complete include report writing and pre-setups.

Pre-setups to be done before starting with AWS CodePipeline

Check out my previous projects on setting up the below steps!

STEP 1: [Setting up a Web App using Visual Studio Code IDE](#)

STEP 2: [Setting up a Git Repository with AWS CodeCommit](#)

STEP 3: [Setting up AWS CodeArtifact](#)

STEP 4: [Package an App with CodeBuild](#)

STEP 5: [Deploy an App with AWS CodeDeploy](#)



CI/CD Pipeline

A CI/CD pipeline is a practice of continuous integration and continuous deployment. This ensures that changes made to the source code are continuously integrated and updated in a shared Git repository, so other developers working on the same codebase have the latest version of the code (continuous integration). It also ensures that end users are always seeing the most recent version of the application (continuous deployment).

My CI/CD pipeline has three stages

1. The **source stage** refers to the source code for my web app, which is currently stored in CodeCommit.
2. The **build stage** refers to the service that manages the build process for my web app. This process is handled by CodeBuild.
3. The **deploy stage** refers to the service that manages the deployment process for my **webapp** (i.e., making my web app available to the world). This process is handled by CodeDeploy.

The screenshot displays the AWS CodePipeline console for a pipeline named 'nextwork-web-pipeline'. The pipeline type is 'V2' and the execution mode is 'SUPERSEDED'. The pipeline is currently in a 'Succeeded' state, indicated by a green checkmark. The pipeline execution ID is 'cd40c47-08d2-4814-8b65-1d8e5c98d5ba'. The pipeline consists of two stages: 'Source' and 'Build'. The 'Source' stage is 'Succeeded' and was executed 31 minutes ago. It uses 'AWS CodeCommit' as the provider. The 'Build' stage is also 'Succeeded' and was executed 29 minutes ago. It uses 'AWS CodeBuild' as the provider. The console shows the 'Disable transition' button for the 'Source' stage and the 'Start rollback' button for the 'Build' stage. The 'Release change' button is highlighted in orange.



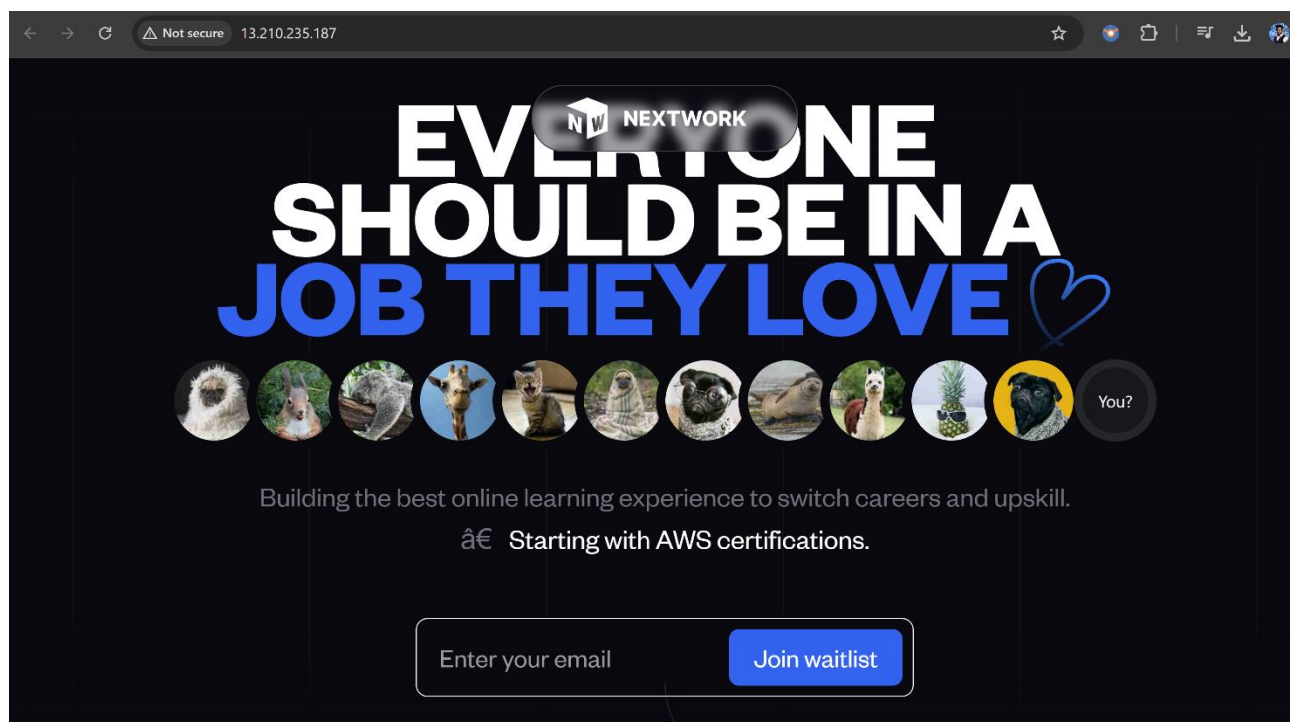
Releasing a Change

My CI/CD pipeline gets triggered when I commit changes in my local working environment (Visual Studio Code). This updates my CodeCommit repository.

I tested this by making **two** updates to my web app's source code: editing my index.jsp file and uploading a folder of image assets to the webapp folder of my project files. Once the two updates are completed, I made a commit and pushed the changes to the main repository by running the commands.

```
cd ~/nextwork-web-project
git add .
git commit -m "Update index.jsp and upload image assets"
git push -u origin main
```

My pipeline executed successfully as soon as I committed and pushed the changes, I verified it by opening the public IPv4 address of my web server that hosts my web app.





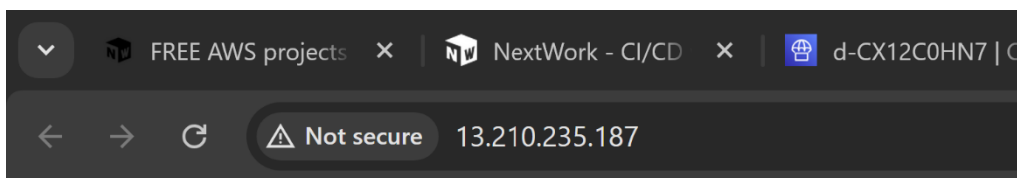
Trigger A Rollback

A rollback means reverting to an earlier, working version of my application if the latest changes cause any problems.

In this case, I only rolled back the deploy stage. This means my source code and build remain the same as the latest version, but the application version deployed to the server is reverted to how it was before my latest commit.

The screenshot shows the AWS CodePipeline console. At the top, there's a status bar with a green checkmark, the word 'Deploy', a red 'Rollback' button, and the word 'Succeeded'. To the right is a 'Start rollback' button. Below this, the 'Pipeline execution ID' is displayed as '18d0c36f-1b29-403c-9d4a-d8e8f6f49908'. The main section shows a 'Deploy' stage using 'AWS CodeDeploy'. It has a green checkmark and says 'Succeeded - 9 minutes ago'. There is a 'View details' button. At the bottom, it says 'a9f7fb70 Source: Adding CodeDeploy files'.

Once the rollback was completed, my web app's appearance reverted to reflect the previous version of my index.jsp file, i.e., before I updated my index.jsp file and uploaded new image assets.



Hello Prajit!

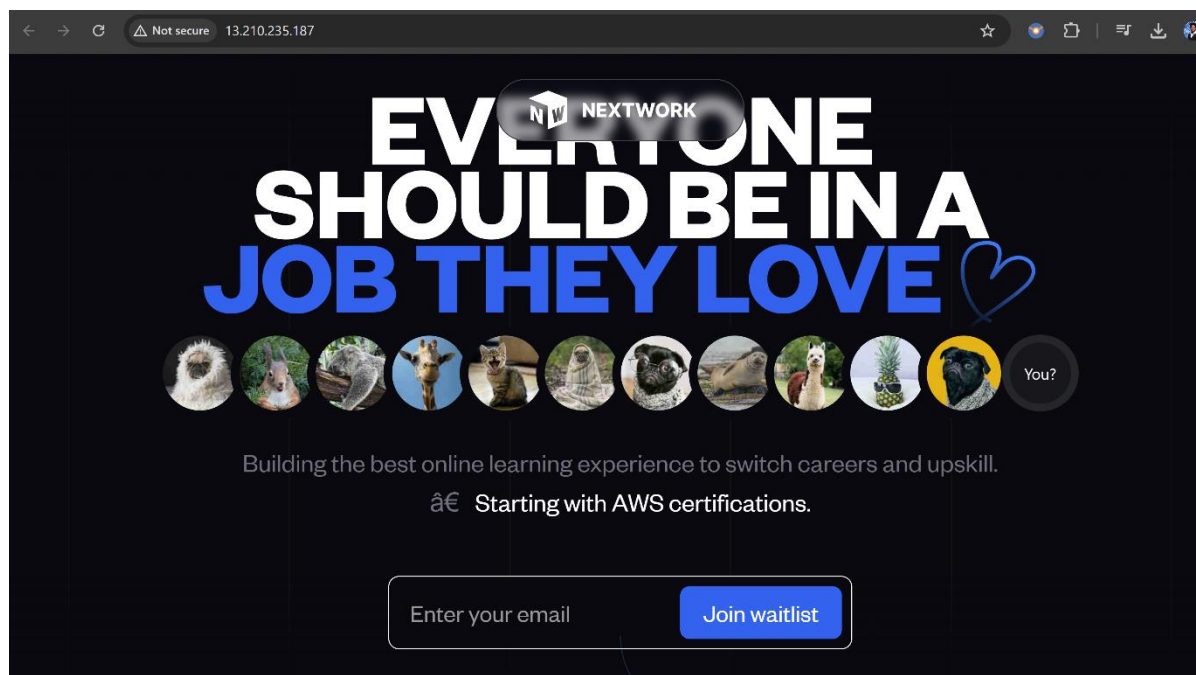
This is my NextWork web application working!



Reverting a Rollback

To revert the deploy stage back to the latest version of my source code, I need to revert the rollback that I did in the earlier step. To make it happen, I released a change again in my CodePipeline. Releasing a change will reflect the latest version my source code in all three stages of the pipeline: Source, Build, and Deploy.

Now, I reloaded the webpage again, and it shown me the latest version of the source code and web page.





NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

