

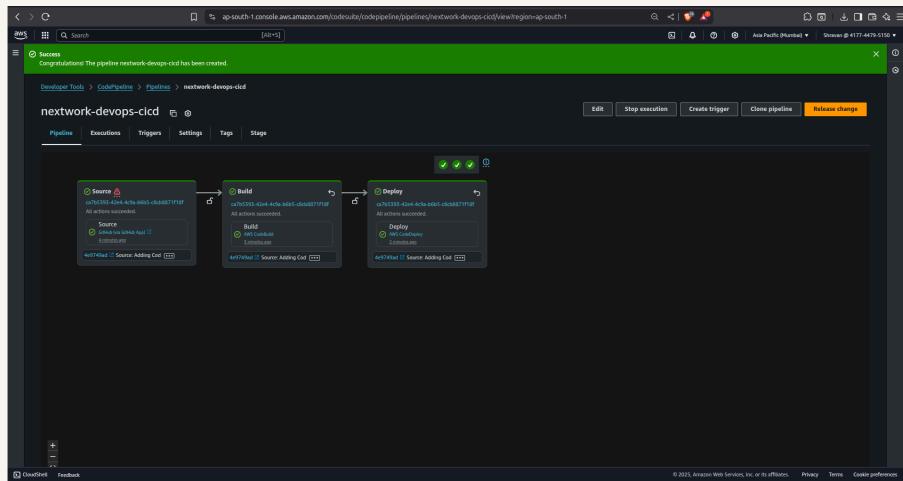


[nextrwork.org](https://nextrwork.org)

# Build a CI/CD Pipeline with AWS



Shravan Kumar Satapathy





**Shravan Kumar Satapathy**

NextWork Student

[nextwork.org](http://nextwork.org)

# Introducing Today's Project!

In this project, I will demonstrate automating CI/CD with AWS CodePipeline, connecting source, build, and deploy stages. I'm doing this to learn how CodePipeline orchestrates and automates consistent, reliable releases.

## Key tools and concepts

Services I used were AWS CodePipeline, GitHub, CodeBuild, CodeDeploy, S3, EC2, IAM, and CloudFormation. Key concepts include CI/CD pipeline orchestration, webhook events, execution modes, and automatic deployment.

## Project reflection

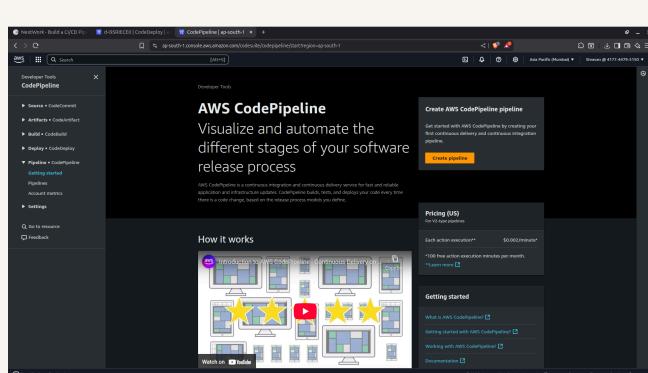
This project took me approximately 160 minutes. The most challenging part was integrating all services and troubleshooting pipeline failures. It was most rewarding to see the fully automated CI/CD pipeline deploy changes in real-time.

# Starting a CI/CD Pipeline

AWS CodePipeline is a service that automates workflows to move code changes through build and deployment stages. It ensures consistent, reliable, and automatic deployments whenever you update your code, minimizing human errors and saving time.

CodePipeline offers different execution modes based on how it handles multiple pipeline runs. I chose Superseded, so newer executions cancel older ones for latest code processing other options are Queued (sequential) and Parallel (simultaneous runs).

A service role gets created automatically during setup so CodePipeline has IAM permissions to access and manage other AWS resources like S3 or CodeBuild, allowing it to run your pipeline.

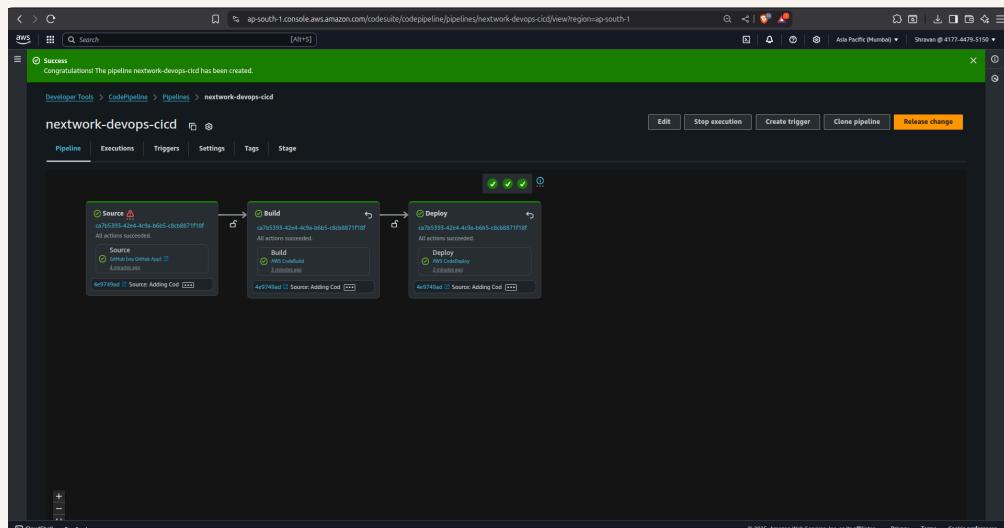




# CI/CD Stages

The three stages I've set up in my CI/CD pipeline are Source, Build, and Deploy. While setting up each part, I learnt about webhook triggers in Source, input artifacts in Build, and automatic rollback in Deploy.

CodePipeline organizes the three stages into a visual diagram. In each stage, you can see its status (grey, blue, green, red) and access execution details, unique IDs, and logs via the Executions tab.



**Shravan Kumar Satapathy**

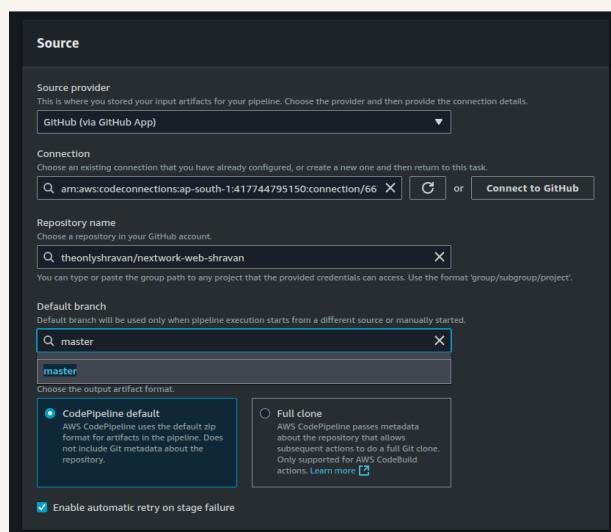
NextWork Student

[nextwork.org](http://nextwork.org)

# Source Stage

In the Source stage, the default branch tells CodePipeline which specific branch to monitor for code changes. This ensures the pipeline automatically triggers a new execution whenever there's a commit to that designated main branch, like 'master'.

The source stage is also where you enable webhook events, which automatically start your pipeline upon a GitHub code push. This makes the pipeline continuous, reacting to changes for seamless automation.



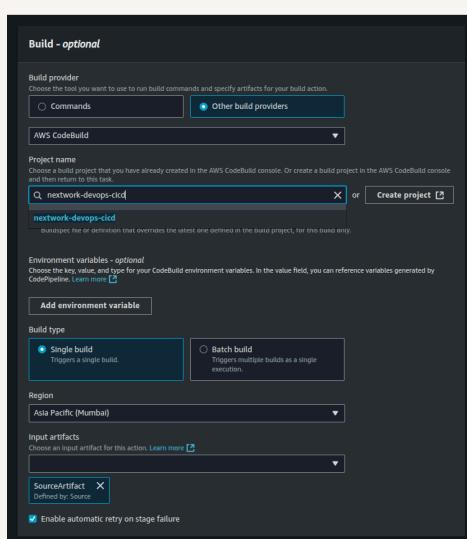
**Shravan Kumar Satapathy**

NextWork Student

[nextwork.org](http://nextwork.org)

# Build Stage

The Build stage sets up CodeBuild to compile and package code. I configured CodeBuild project nextwork-devops-cicd. The input artifact for the build stage is SourceArtifact because it's the ZIP file of source code from the previous stage.



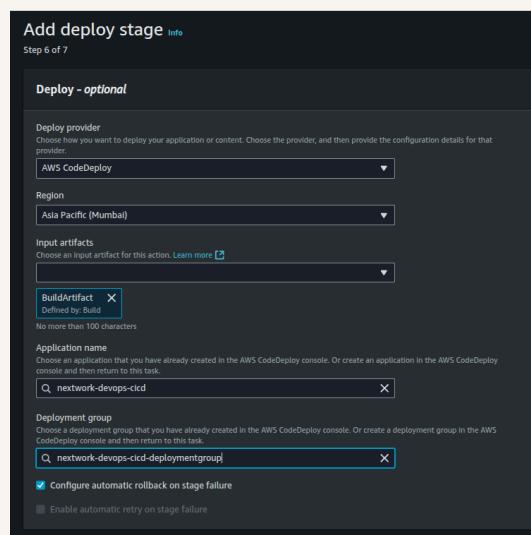
**Shravan Kumar Satapathy**

NextWork Student

[nextwork.org](http://nextwork.org)

# Deploy Stage

The Deploy stage is where application artifacts are deployed. I configured AWS CodeDeploy with BuildArtifact as input, using my existing CodeDeploy application and deployment group. I also enabled automatic rollback on stage failure for safety.





# Success!

Since my CI/CD pipeline gets triggered by code pushes to GitHub, I tested my pipeline by adding a new line to the index.jsp file in my web app. I then committed and pushed these changes to the master branch of my GitHub repository.

The moment I pushed the code change, a new pipeline execution started automatically, progressing through all stages to success. The commit message under each stage reflects the change, which I verified in GitHub.

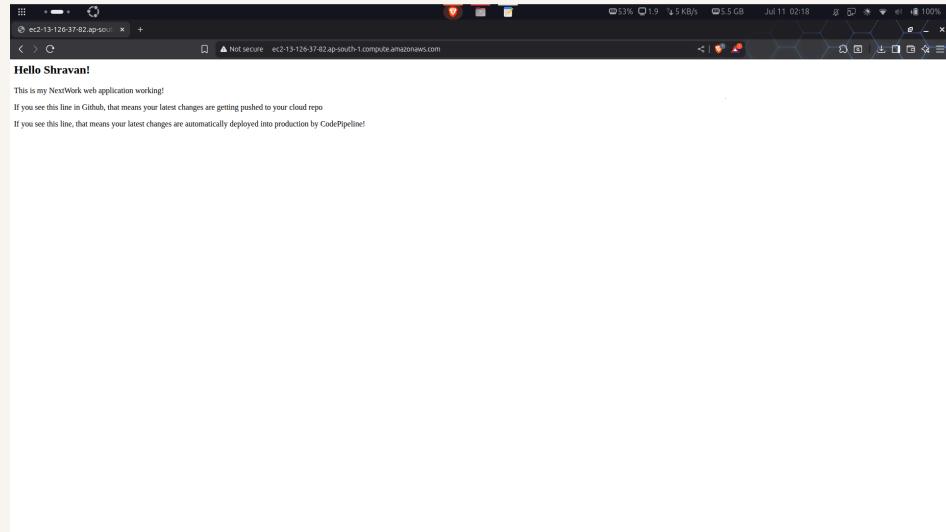
Once my pipeline executed successfully, I checked the web application live. I accessed the EC2 instance's Public IPv4 DNS and saw my new code change, confirming the CI/CD pipeline automatically deployed the updates.



**Shravan Kumar Satapathy**

NextWork Student

[nextwork.org](http://nextwork.org)





[nextwork.org](https://nextwork.org)

# The place to learn & showcase your skills

Check out [nextwork.org](https://nextwork.org) for more projects

