

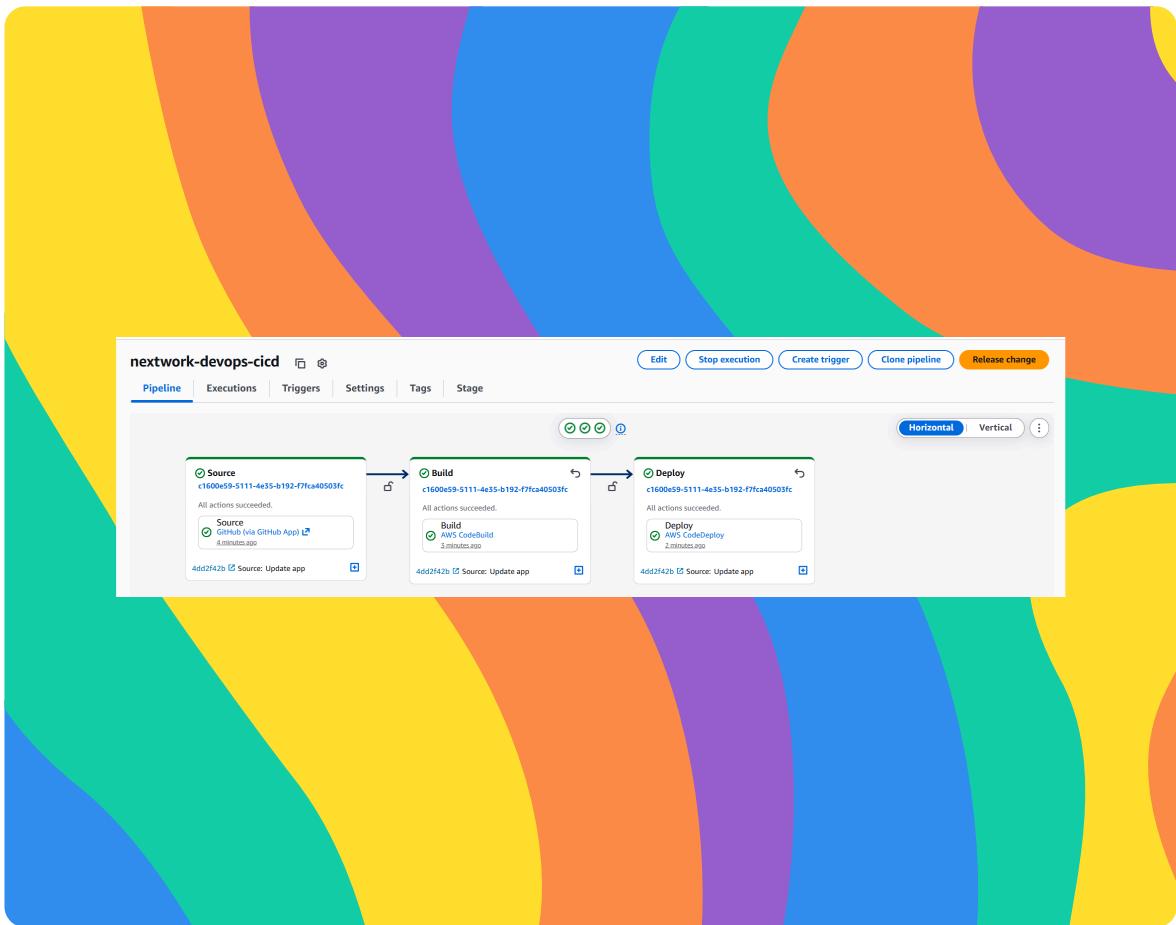


nextwork.org

Build a CI/CD Pipeline with AWS



Mohammed Furqanuddin





Mohammed Furqanud...

NextWork Student

nextwork.org

Introducing Today's Project!

In this project, I will demonstrate how to build a complete CI/CD pipeline using AWS CodePipeline by connecting source, build, and deployment stages. I'm doing this project to learn how automated deployments work, test changes, and safely roll back when needed

Key tools and concepts

The key services used were AWS CodePipeline, CodeBuild, CodeDeploy, EC2, IAM, CloudFormation, S3, and GitHub. Through this project, I learned CI/CD automation, in-place deployments, rollbacks, IAM permissions, infrastructure as code, and monitoring pipeline executions.

Project reflection

This project took me approximately 3-4 hours to complete. The most challenging part was configuring the CI/CD pipeline and resolving region and deployment issues. The most rewarding part was successfully triggering an automated deployment and rollback, which confirmed that the pipeline was working end to end.

Mohammed Furqanud...

NextWork Student

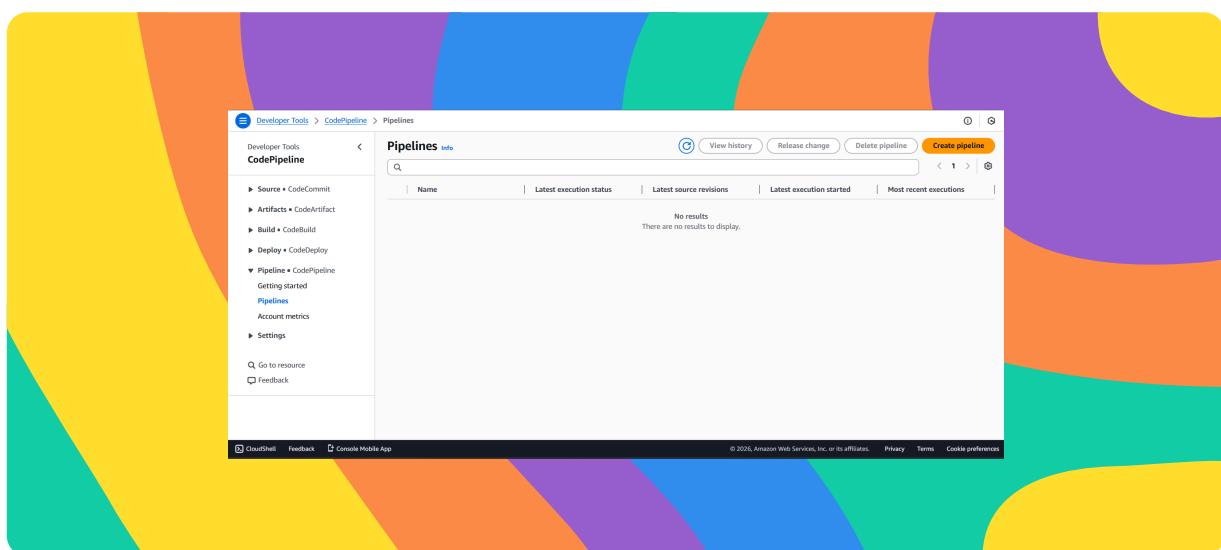
nextwork.org

Starting a CI/CD Pipeline

AWS CodePipeline is a fully managed CI/CD service that automates building, testing, and deploying applications by connecting all stages into a single workflow.

AWS CodePipeline provides multiple execution modes to control how pipeline runs behave. I chose Superseded mode so that any new change immediately replaces the currently running execution. Other options include Queued, where executions run one after another, and Parallel, where multiple executions can run at the same time.

AWS CodePipeline requires a service role to securely interact with other AWS services. This role is created automatically during setup so CodePipeline can access resources like source repositories, build services, and deployment targets without manual permission handling.



Mohammed Furqanud...

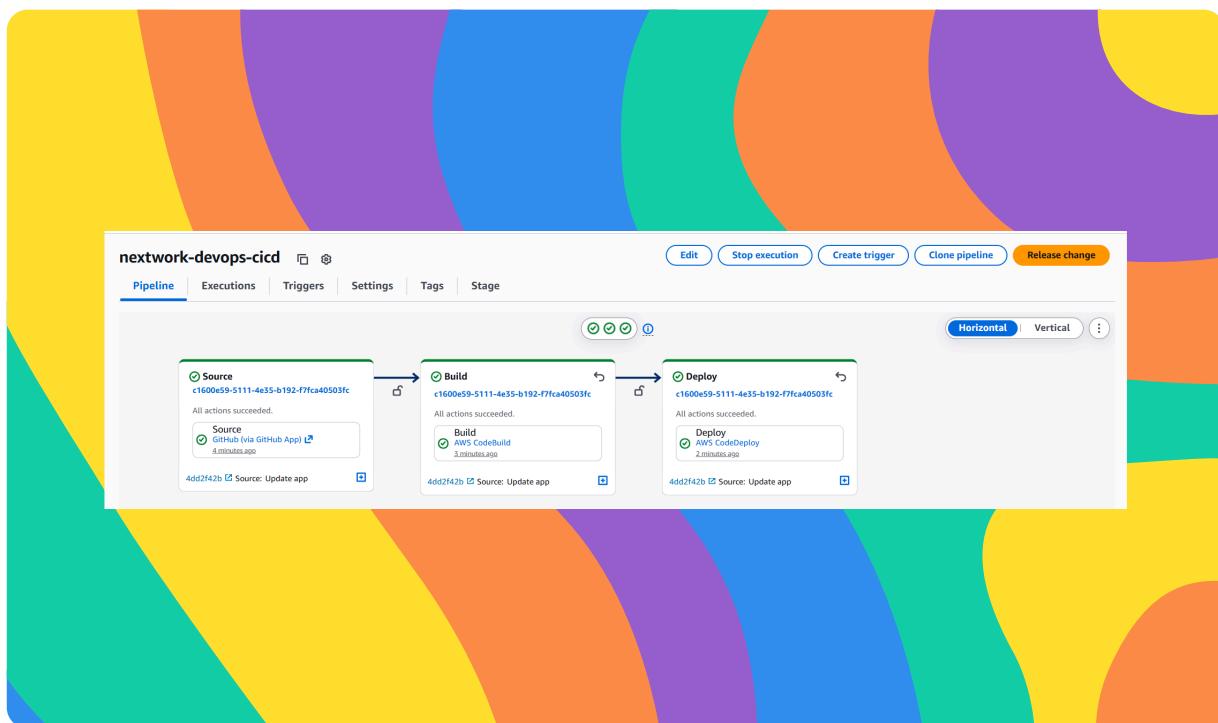
NextWork Student

nextwork.org

CI/CD Stages

I verified all pipeline settings for nextwork-devops-cicd. The pipeline uses V2 with Superseded execution mode and a new service role. The Source stage connects to GitHub with change detection on the master branch, the Build stage uses AWS CodeBuild, and the Deploy stage uses AWS CodeDeploy with automatic rollback enabled.

CodePipeline organizes the workflow into three stages: Source, Build, and Deploy. For each stage, the pipeline shows the status of execution, the service used (GitHub, CodeBuild, and CodeDeploy), and the time of the latest run. By opening a stage, I can view more details such as commit information, build logs, and deployment results, which helps in tracking progress and quickly identifying any issues in the pipeline.



Mohammed Furqanud...

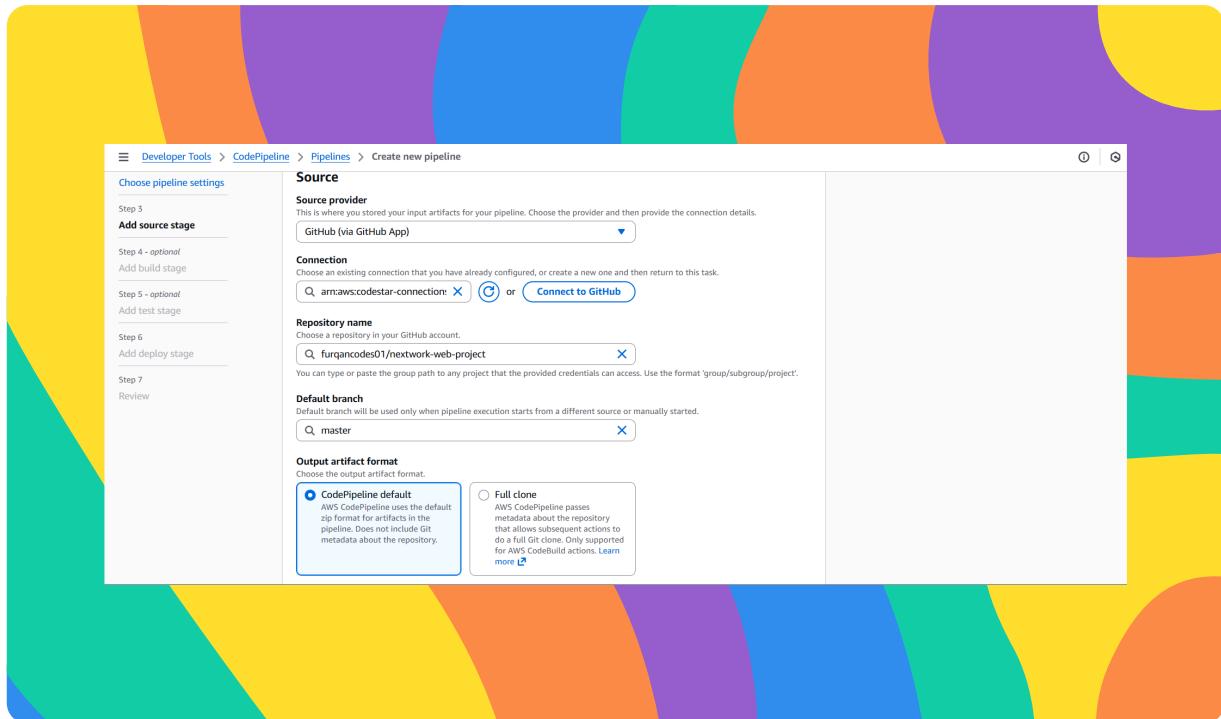
NextWork Student

nextwork.org

Source Stage

In the Source stage, the default branch tells AWS CodePipeline which branch to monitor for code changes. This ensures that the pipeline is triggered only when updates are pushed to the selected branch, allowing controlled and consistent builds and deployments.

Webhook events allow AWS CodePipeline to automatically trigger the pipeline when code is pushed to the specified branch in GitHub. GitHub sends a notification to CodePipeline on each push, enabling real-time, continuous CI/CD execution.



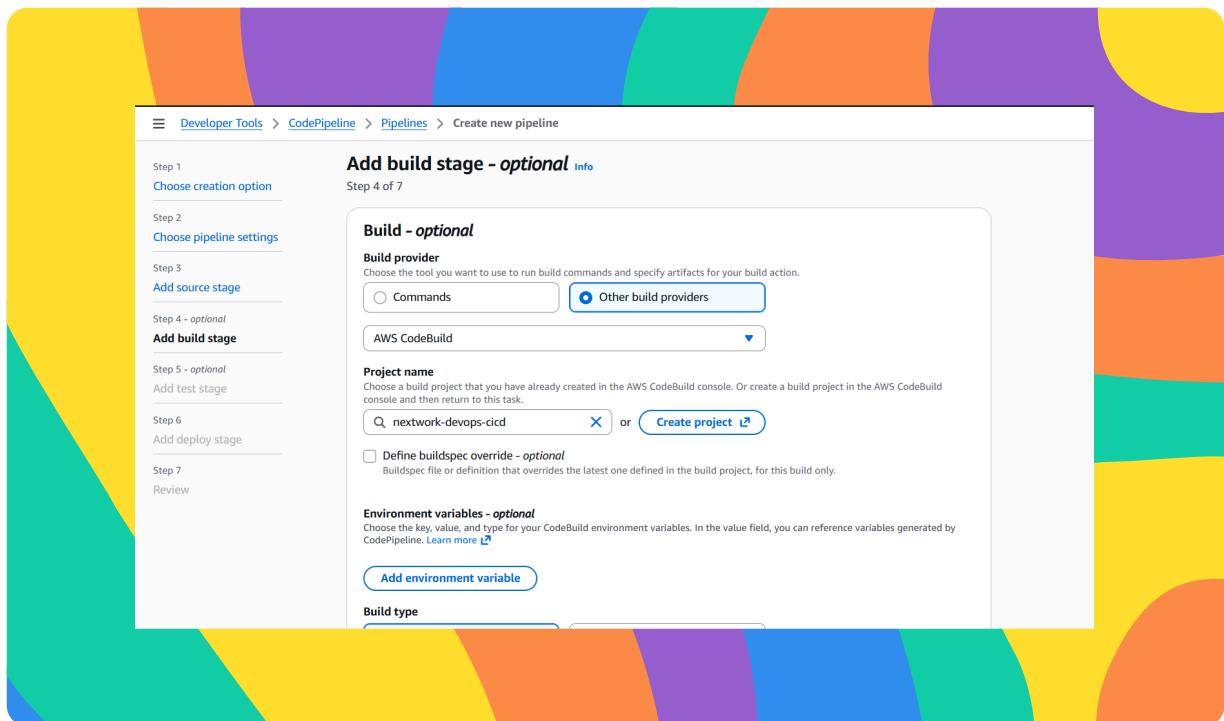
Mohammed Furqanud...

NextWork Student

nextwork.org

Build Stage

The Build stage is responsible for compiling and packaging the application into a deployable artifact. I configured AWS CodePipeline to use AWS CodeBuild for this process. The input artifact for the Build stage is SourceArtifact because it contains the source code output from the Source stage. This allows CodeBuild to access the application files and transform them into a build artifact ready for deployment.



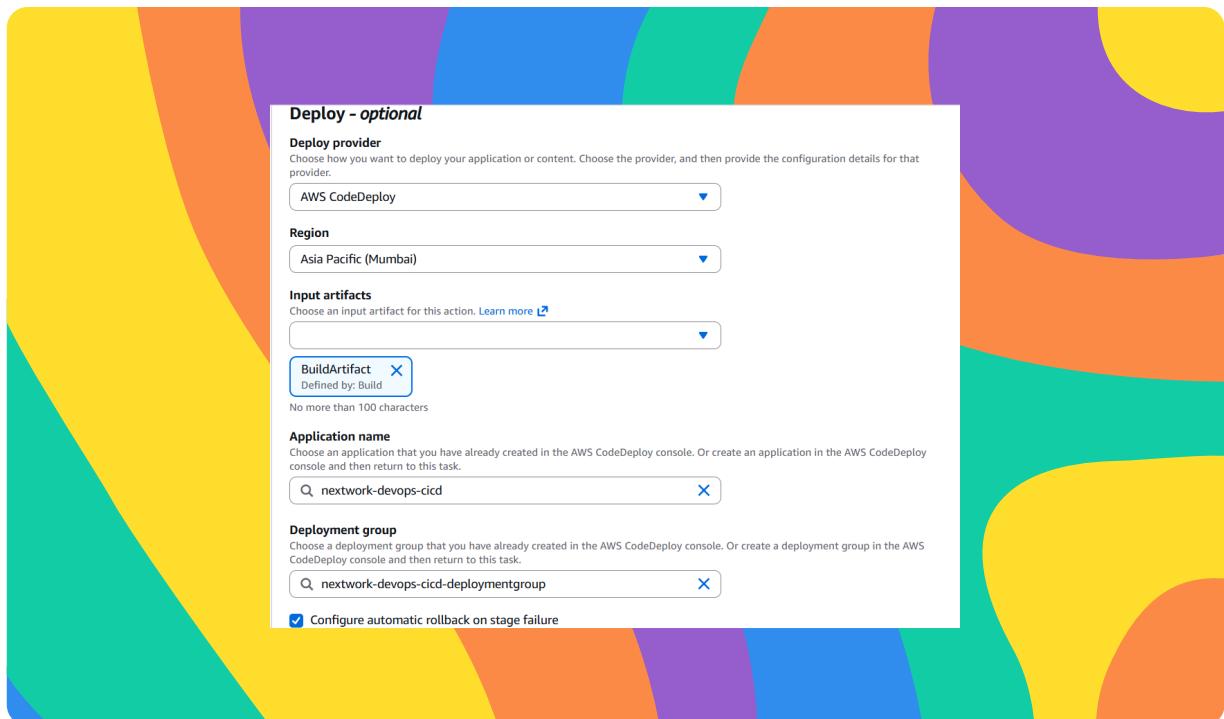
Mohammed Furqanud...

NextWork Student

nextwork.org

Deploy Stage

The Deploy stage is where the application is released to the target environment. I configured AWS CodePipeline to use AWS CodeDeploy, using the BuildArtifact from the Build stage and an existing deployment group, with automatic rollback enabled for reliability.





Mohammed Furqanud...

NextWork Student

nextwork.org

Success!

I tested the pipeline by adding a new line to the index.jsp file and pushing the change to GitHub. This triggered the CI/CD pipeline automatically and confirmed that the updated code was successfully built and deployed using CodePipeline.

The moment I pushed the code change to GitHub, the pipeline was automatically triggered. The Source stage detected the new commit, the Build stage compiled the updated code, and the Deploy stage deployed the changes to the EC2 instance. The commit message was visible under each stage, confirming that the same code version successfully passed through the entire pipeline.

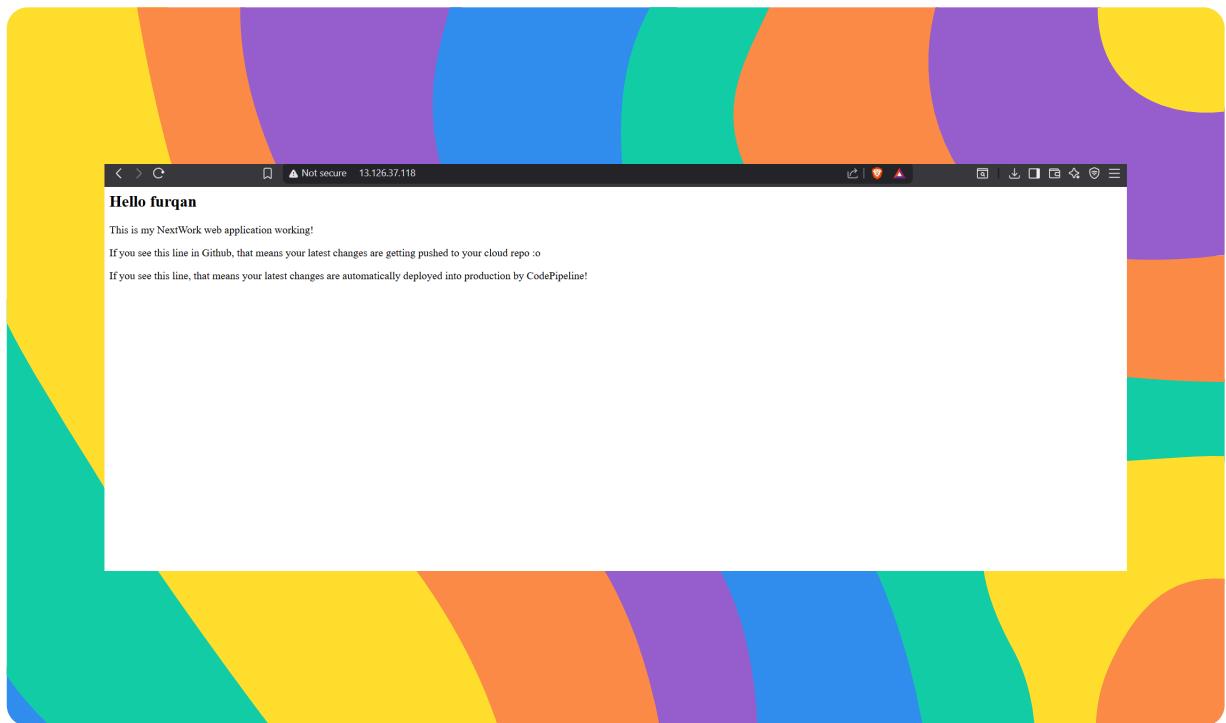
Once my pipeline executed successfully, I confirmed it was working by opening the web application using the EC2 public IP address. The newly added line from the latest code change appeared on the page, which verified that the update was automatically built and deployed through the CI/CD pipeline. I also confirmed that all stages in CodePipeline (Source, Build, and Deploy) completed successfully without errors.



Mohammed Furqanud...

NextWork Student

nextwork.org





Mohammed Furqanud...

NextWork Student

nextwork.org

Testing the Pipeline

In a project extension, I initiated a rollback during the Deploy stage of the pipeline. Automatic rollback is important because it allows the application to quickly return to a stable previous version if a deployment fails, minimizing downtime and ensuring reliability without requiring manual intervention.

During the rollback, the Source and Build stages were unaffected because rollback only applies to the Deploy stage. I verified this by noticing that Source and Build continued to show the latest commit, while the application was reverted to a previous deployed version.

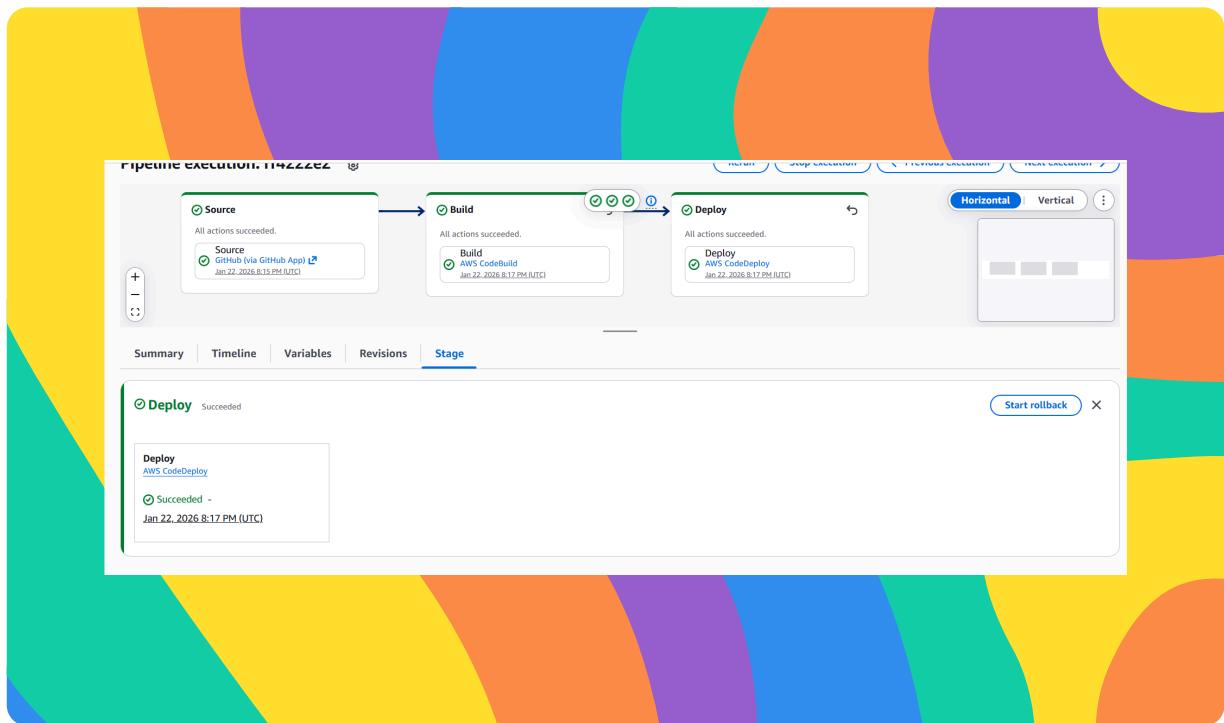
After refreshing the web application using the same Public IPv4 DNS, the newly added line was no longer visible. This confirmed that the application had been successfully reverted to the previous version, verifying that the rollback worked as expected.



Mohammed Furqanud...

NextWork Student

nextwork.org





nextwork.org

The place to learn & showcase your skills

Check out nextwork.org for more projects

