NextWork.org

CI/CD Pipeline with AWS CodePipeline

Sadeesha Perera


Sadeesha Perera
linkedin.com/sadeesha-perera

Introducing AWS CodePipeline!

What it does & how it’s useful

AWS CodePipeline is an AWS service that automates the steps that DevOps/cloud engineers take to release application updates. Developers and teams use AWS CodePipeline because it basically automates the whole development process once it has been setup

How I’m using it in today’s project

I’m using AWS CodePipeline in this project to create an automated pipeline using the CodeCommit, CodeBuild and CodeDeploy components created in previous projects.

This project took me...

The project took me 150 minutes, due to unexpected issues while creating my deployment group, and documentation to me 30 minutes. In total I spent 3 hours on this project.


Sadeesha Perera
linkedin.com/sadeesha-perera

Set Up a CI/CD Pipeline

- A CI/CD pipeline is a practice of continues integration, and continuous deployment. This makes sure that changes made to the source code are constantly being integrated, and updated in a shared git repository so that other developers working from the same code base are always using the latest version of the code (Continues Integration). It also assures that end users are always seeing the latest version and updates to the application (Continues deployment)

To set up a CI/CD in CodePipeline, I configured three stages:

- The **source** stage, which means the source code for my web app, this is currently stored in CodeCommit

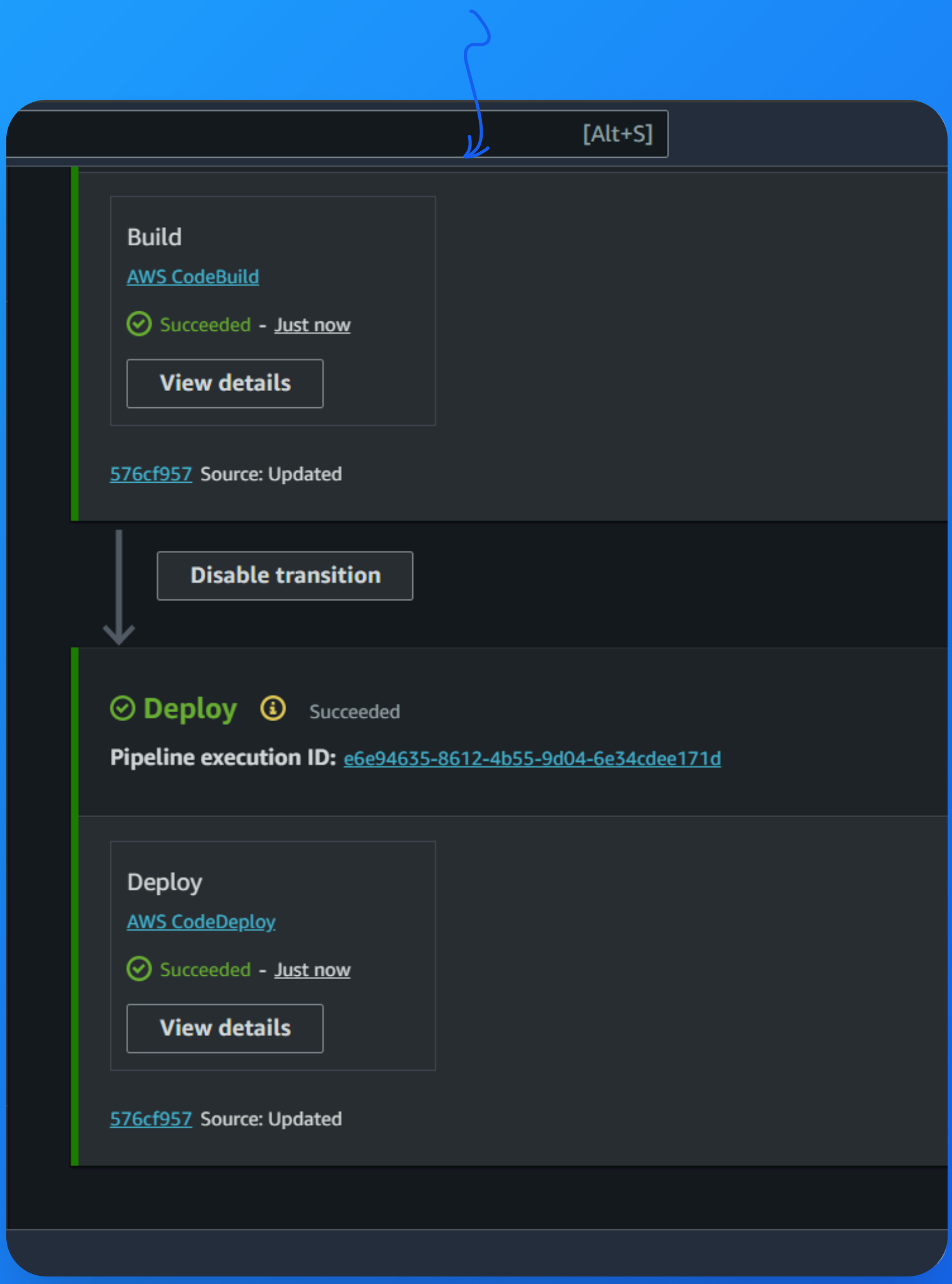
- The **build** stage, which means s the service that is managing the build process for my web app i.e. packaging and compiling my web app code into a WAR file. This is CodeBuild.

- The **deploy** stage, which means the service that is managing the deployment process for my webapp, i.e. making it available to the world. This is CodeDeploy

Sadeesha Perera
linkedin.com/sadeesha-perera

NextWork.org

A peek into my pipeline set up in CodePipeline!




Sadeesha Perera
linkedin.com/sadeesha-perera

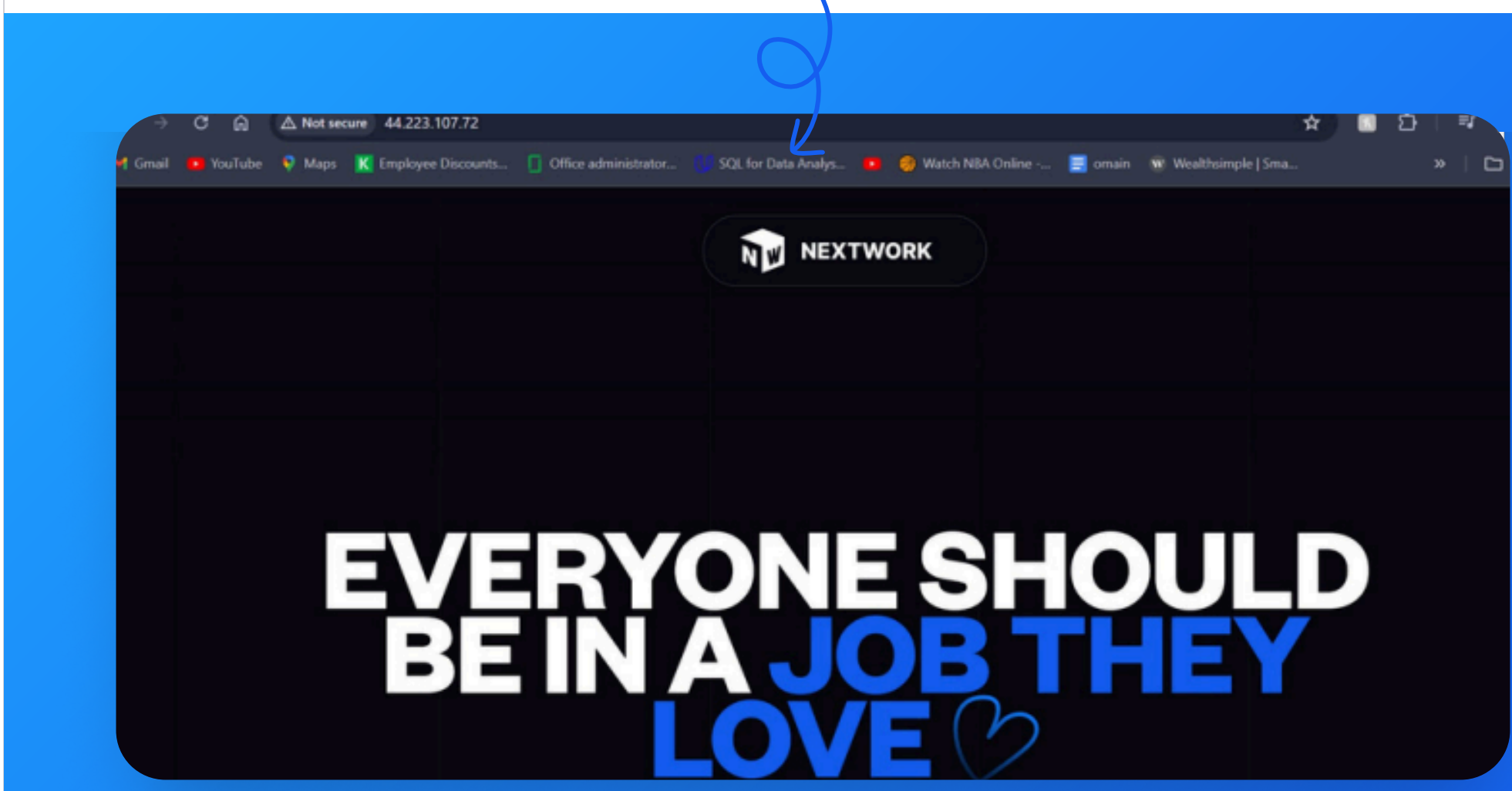
Release a Change

- My CI/CD pipeline gets triggered by a commit in my local working environment (Cloud9), which updates my CodeCommit repository.

- I tested this by making two updates to my web app's source code. These changes were an edit to index.jsp, and uploading a folder of image assets in the web app folder of my project files

- Once my pipeline executed successfully, I checked the IPv4 address of my web server that is hosting my web app

My web app automatically updated after I committed a change.




Sadeesha Perera
linkedin.com/sadeesha-perera

Trigger A Rollback

- A rollback in a pipeline means reverting the version of our code that a stage of our pipeline is referring to.

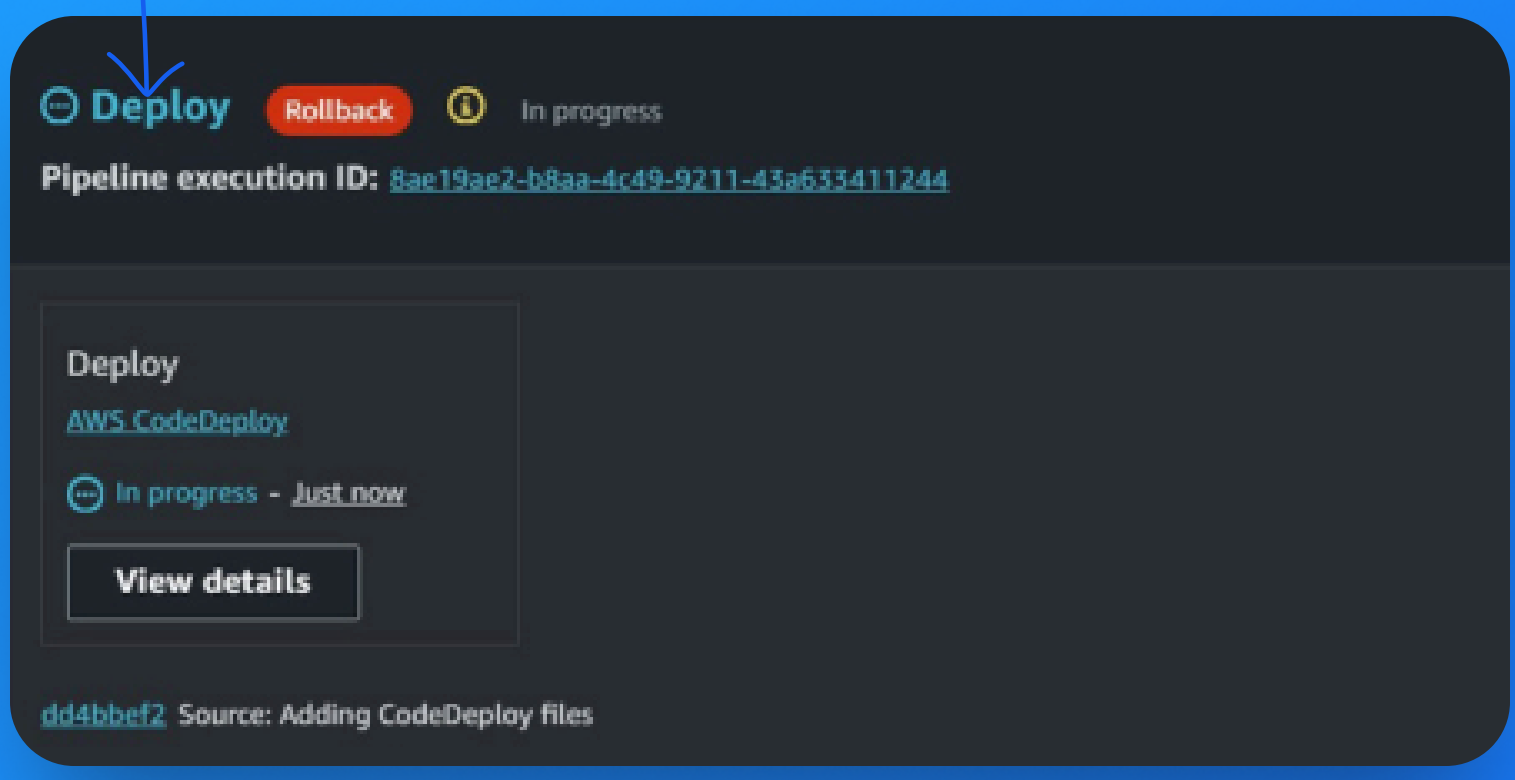
- I initiated a rollback on the deployment stage. I checked the source stage and learnt that the source stage was unaffected by the rollback, and stayed using the latest version of my source code.

- Once the rollback was complete, my web app reverted back to reflect the simple index.jsp file, i.e. before I updated my file with new image assets.

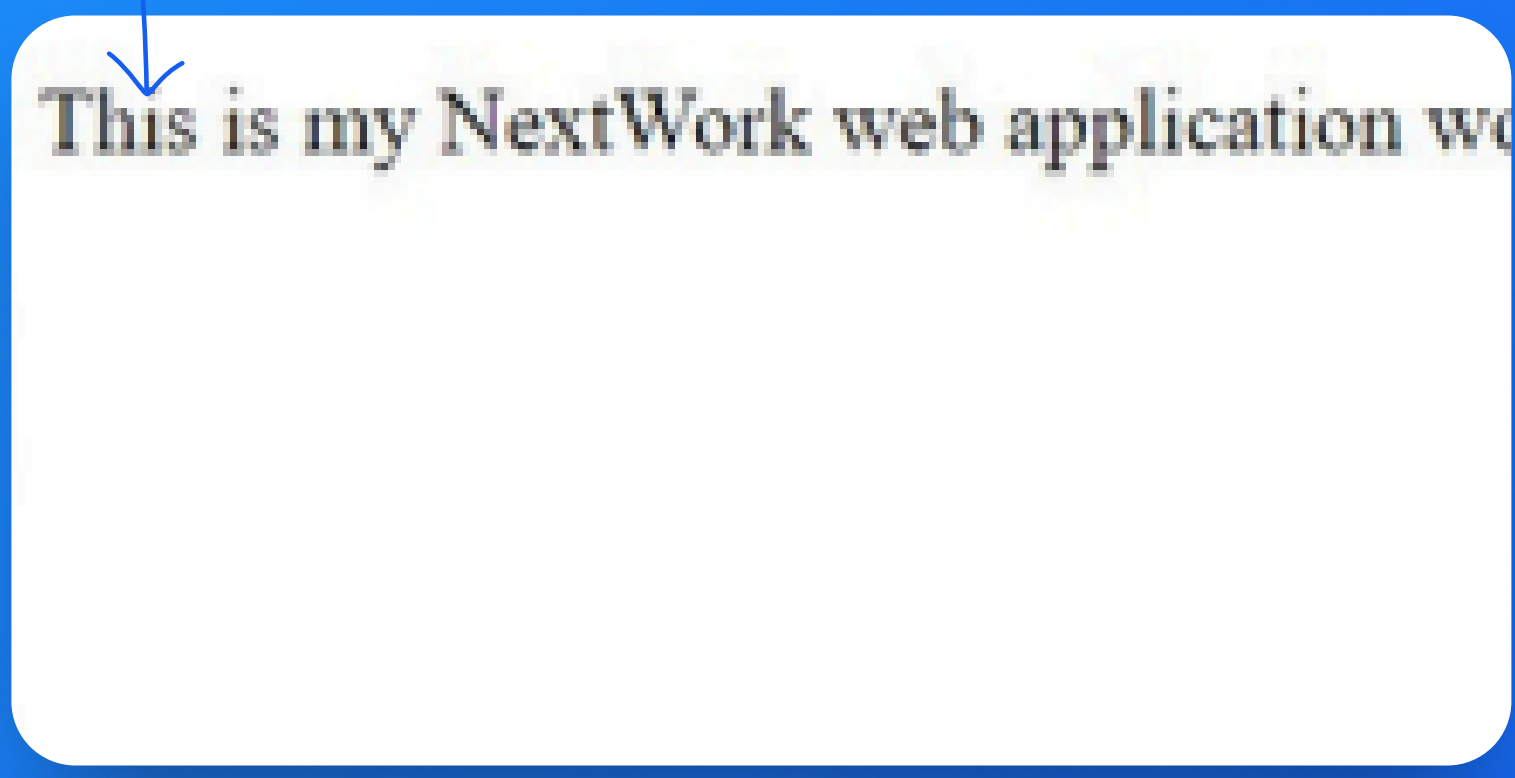
- To update my Deploy stage back to the latest version of my source code, I released the change in my CodePieline pipeline. Releasing a change means that the latest version of my source code gets referenced across the Source, Build and Deploy stages of my pipeline.

Sadeesha Perera
linkedin.com/sadeesha-perera

My CodeDeploy panel showing a successful rollback.



My web app returned to its previous state after the rollback.




Sadeesha Perera
linkedin.com/sadeesha-perera

NextWork.org

My CI/CD Pipeline

1. **AWS Cloud9** is responsible for writing, debugging and running my source code.

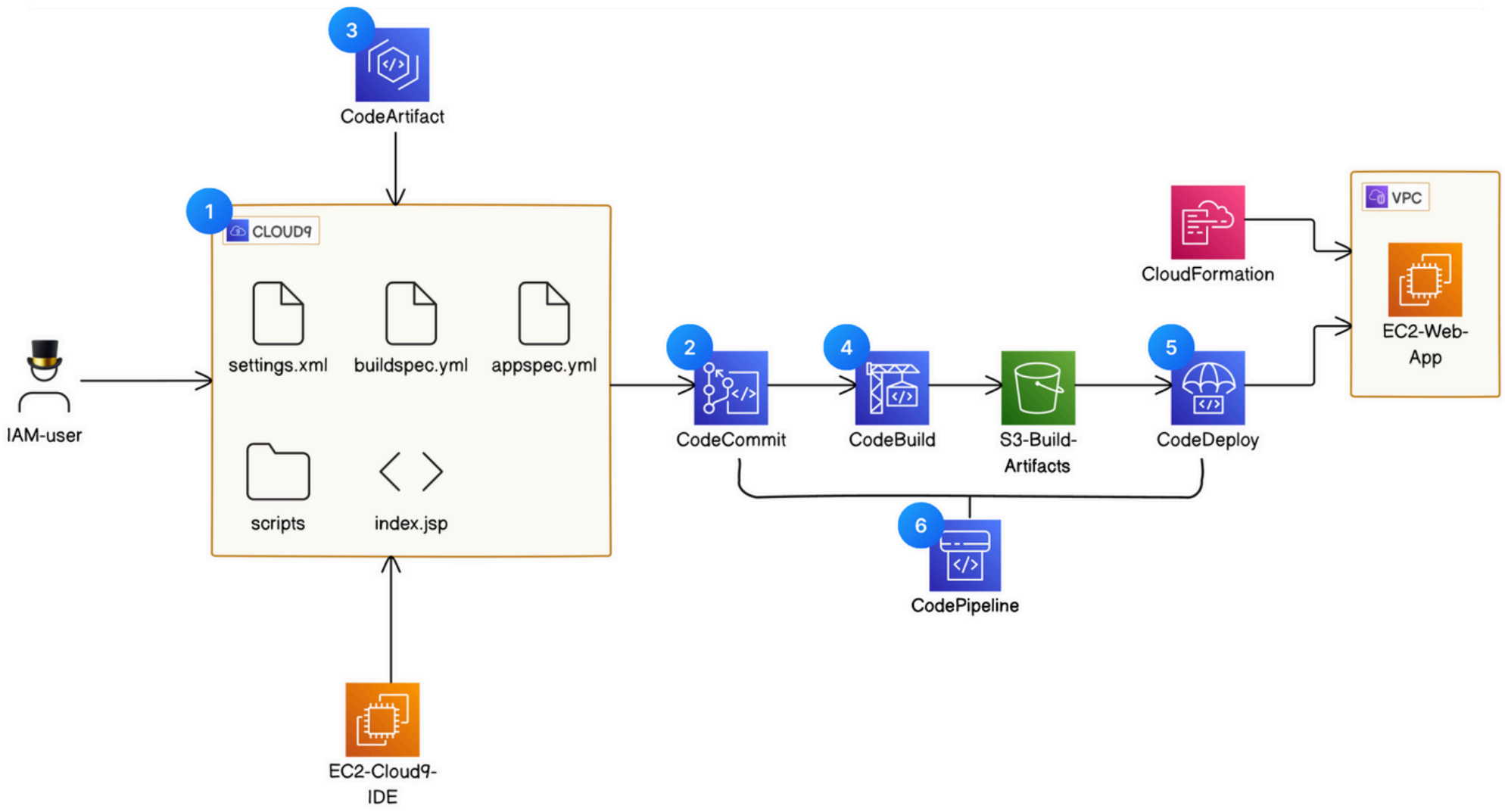
2. **AWS CodeCommit** is responsible for storing repositories of my source code from Cloud9.

3. **AWS CodeArtifact** is responsible for storing packages and dependencies for my web application.

4. **AWS CodeBuild** is responsible for building my source code and storing my artefacts in an S3 bucket.

5. **AWS CodeDeploy** is responsible for automating the deployment of my web application to my EC2 instance.

6. **AWS CodePipeline** is responsible for modelling the full release process for building my code, deploying, testing my web application and releasing afterwards.




Sadeesha Perera
linkedin.com/sadeesha-perera

My key learnings

1. CI/CD means Continuous integration and continuous deployment. It the process of streamlining and accelerating the software development lifecycle.

2. To set up a CI/CD pipeline for my web app project, i configured three stages:
Source
Build
Deploy

3. Example scenarios of triggering a rollback in the deploy stage are reverting my web application to its previous version, or reverting to the period before the source code was last updated.

4. One thing I didn’t expect was how the pipeline instantly updates the web app after making changes to the source code and running the commit command. I now see how this can be very convenient and beneficial for developers. especially with the security of reverting to the previous version if something crashes after updating the source code.

NEXTWORK

Everyone should be in a job they love. yes!

Check out community.nextwork.org for more free projects