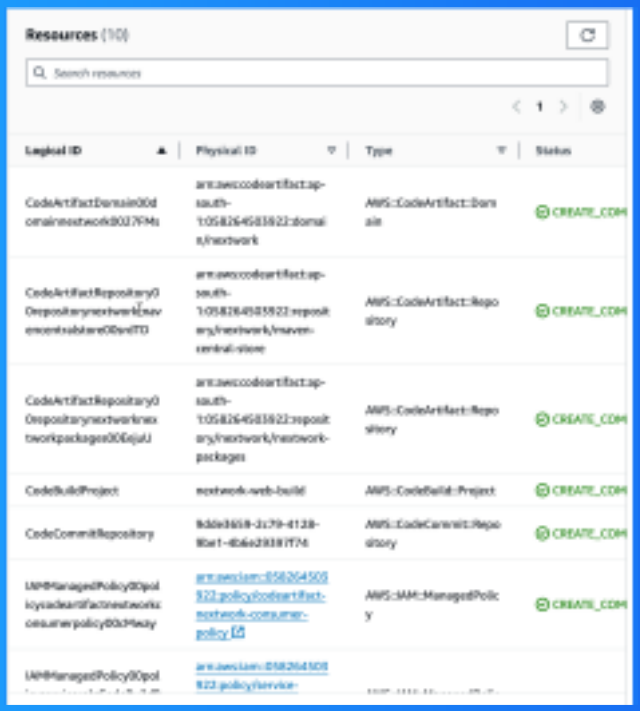
NextWork.org

Automate with Cloud Formation

ZA zandiletsh20@gmail.com



ZA
zandiletsh20@gmail.com
NextWork Student
NextWork.org

Introducing today's project!

What is AWS CloudFormation?

AWS CloudFormation allows you to define and manage AWS infrastructure using code. It automates resource management, ensuring consistent and repeatable deployments, and integrates with other AWS services for efficient infrastructure management.

How I used CloudFormation in this project

I created a CloudFormation template for my WebApp, using the IaC generator for some resources and manually adding others.

One thing I didn't expect in this project was...

I initially didn't expect CloudFormation to not rearrange dependencies automatically, but later understood the reason behind that.

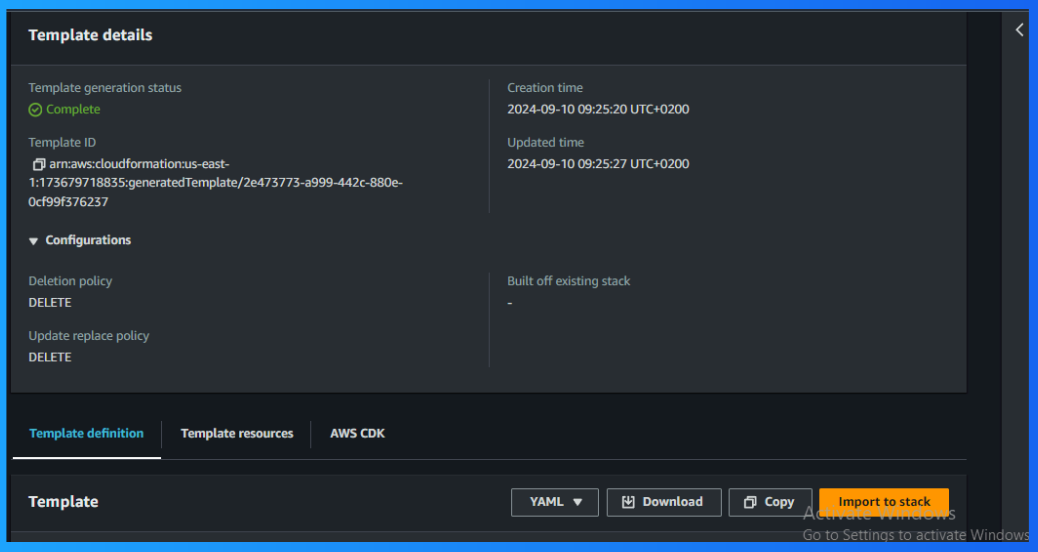
This project took me...

This project took me about 2 hours to completely understand and document.

ZA
zandiletsh20@gmail.com
NextWork Student

CloudFormation templates

A CloudFormation template is a JSON or YAML file used in AWS CloudFormation to define and automate the creation and management of AWS resources. It includes sections like: resources, mappings, parameters, outputs, conditions, metadata and transform.



ZA
zandiletsh20@gmail.com
NextWork Student
NextWork.org

IaC generator

I created a CloudFormation template using the IaC generator, which automates the creation, management, and provisioning of AWS resources by defining them in templates written in JSON or YAML.

Not all resources could be added to my template

The resources I couldn't add to a template were Cloud9, CodeCommit repository, and CodeBuild project.

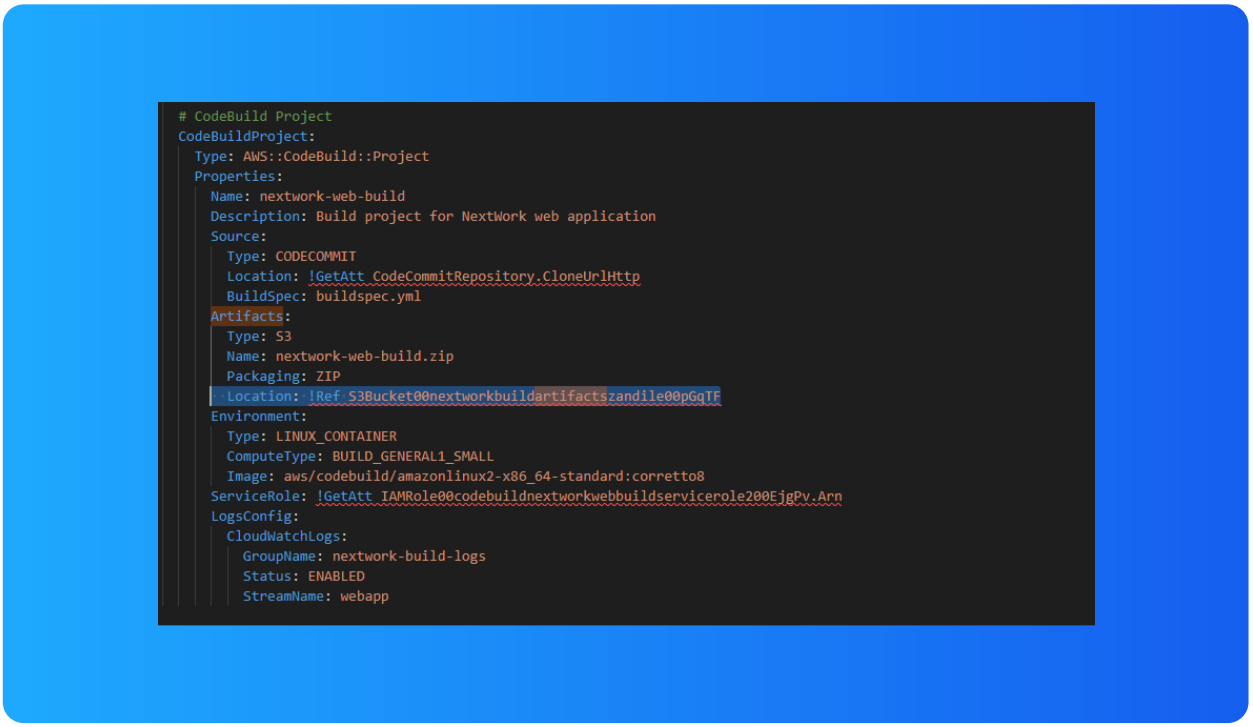
The resources that I could add to my template includes: CodeArtifact domain, local CodeArtifact repository, upstream CodeArtifact repository, IAM policy, IAM service role an S3 bucket.

Manually adding resources

After downloading the generated template, I manually defined two more resources: a CodeCommit repo and a CodeBuild project.

I had to manually define these because the configurations were too complex or required manual setup, making it impossible to generate them using an Infrastructure as Code (IaC) generator.

I also had to make sure the references were consistent in this template, so I made two edits for the configuration name of the build artifacts bucket and IAM service role for CodeBuild.



```
# CodeBuild Project
CodeBuildProject:
  Type: AWS::CodeBuild::Project
  Properties:
    Name: nextwork-web-build
    Description: Build project for NextWork web application
    Source:
      Type: CODECOMMIT
      Location: !GetAtt CodeCommitRepository.CloneUrlHttp
      BuildSpec: buildspec.yml
    Artifacts:
      Type: S3
      Name: nextwork-web-build.zip
      Packaging: ZIP
      Location: !Ref S3Bucket00nextworkbuildartifactszandile00pGqTF
    Environment:
      Type: LINUX_CONTAINER
      ComputeType: BUILD_GENERAL1_SMALL
      Image: aws/codebuild/amazonlinux2-x86_64-standard:corretto8
    ServiceRole: !GetAtt IAMRole00codebuildnextworkwebbuildservicerole200EjgPv.Arn
    LogsConfig:
      CloudWatchLogs:
        GroupName: nextwork-build-logs
        Status: ENABLED
        StreamName: webapp
```

Testing my template

Before testing my template, I removed potential conflicts and overlaps between existing AWS resources in the account and the new resources I wanted to deploy.

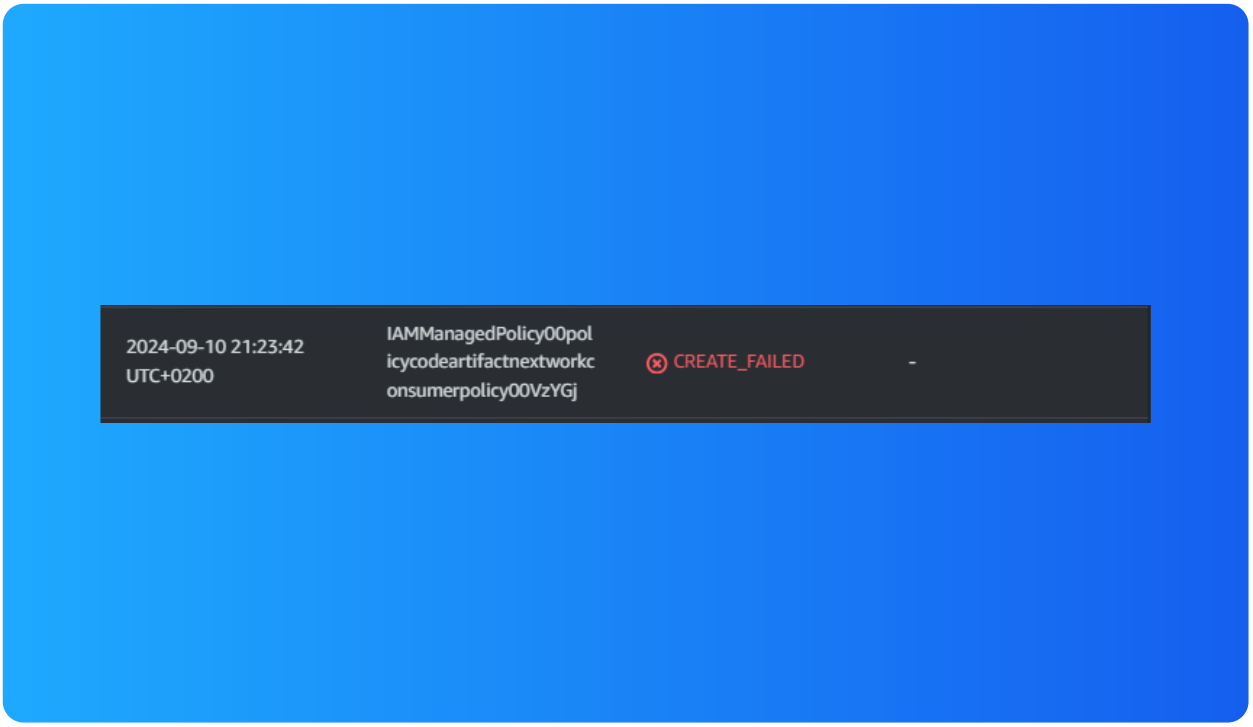
A stack is a collection of resources that can be deployed or deleted as a single unit.

The result of my first test was create failed.

ZA
zandiletsh20@gmail.com
NextWork Student

Unpacking the first error

My first template test failed because CloudFormation tried to attach IAM policies to the CodeBuild service role before the role was created. Since the role didn't exist yet, the policies could not be attached, causing the failure.



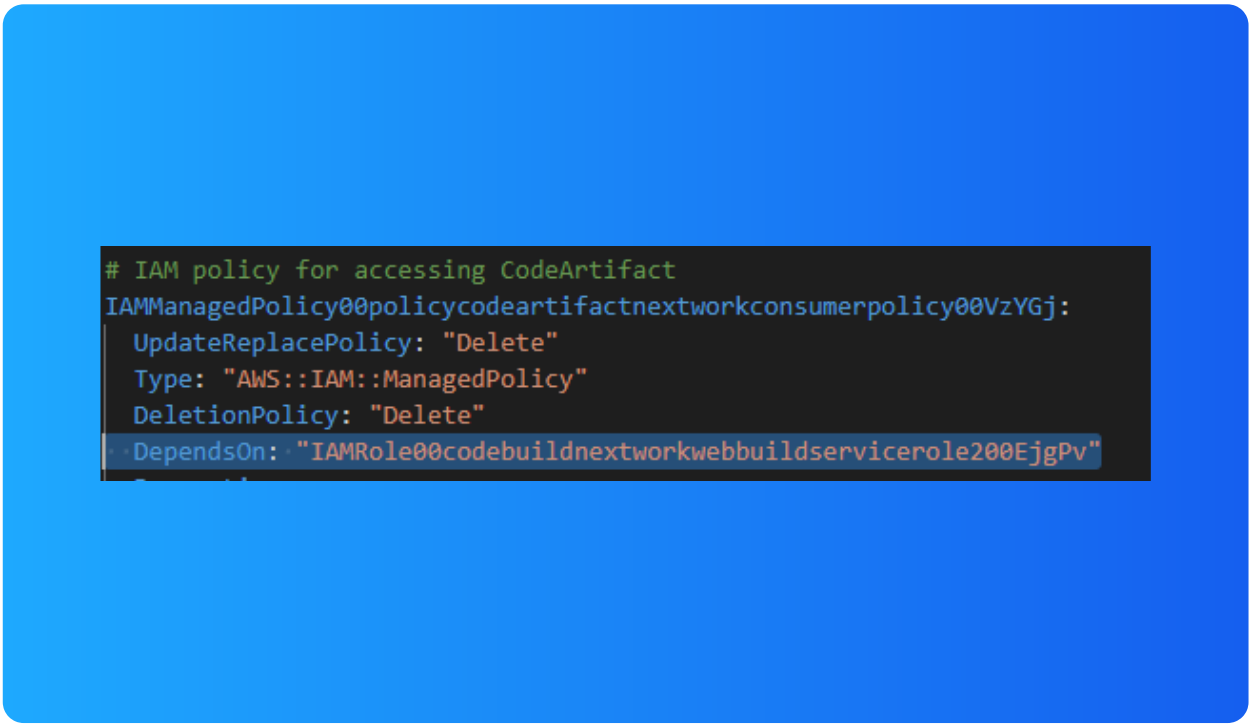
To fix this error, I edited my CloudFormation template.

I added the line DependsOn: "MY CLOUDBUILD IAM ROLE's NAME IN THE CLOUDFORMATION TEMPLATE" in the configuration of the IAM policies, which tells cloudformation to create CodeBuild service role first before creating IAM policies.

Fixing the first error

The DependsOn attribute means one resource is created or deleted only after another specified resource is successfully created or deleted.

The DependsOn line was added to four different parts of my template: the 3 IAM policies and the CodeBuild project.

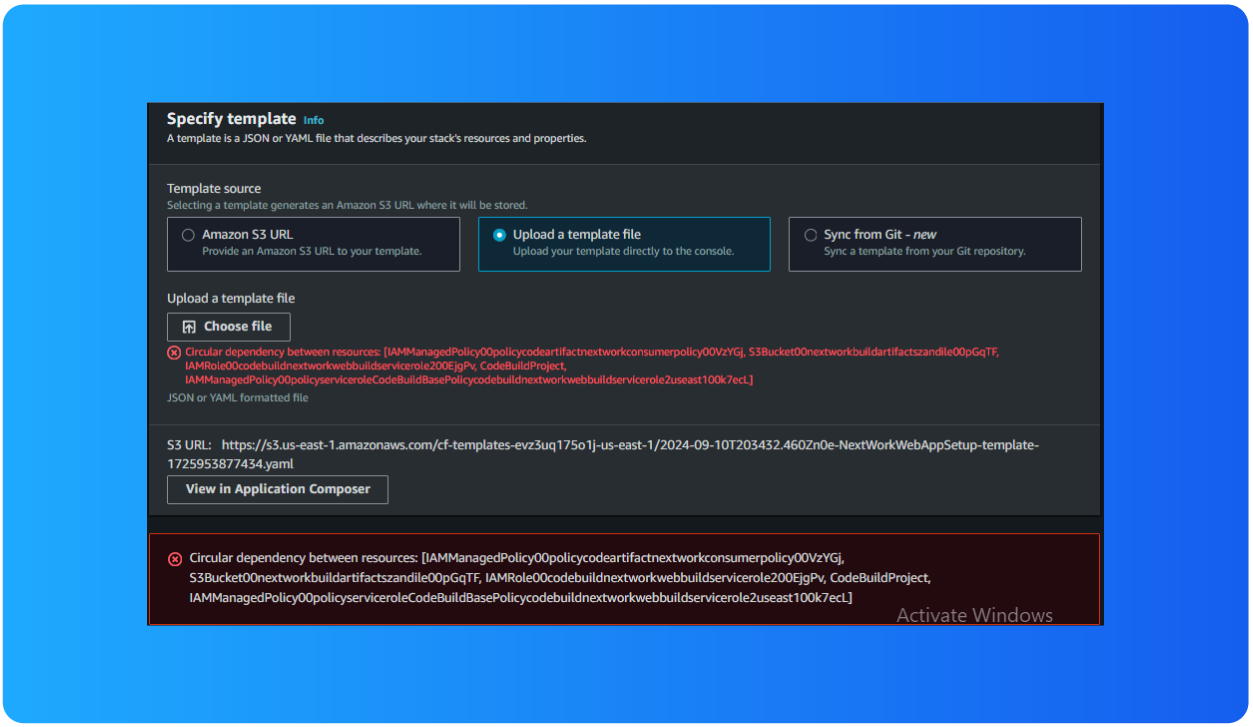


Second template test

I gave my CloudFormation template another test! But this time, I couldn't create the stack because of an error. This is known as a circular dependency.

This error means that resources in a CloudFormation template have circular dependencies, meaning they reference each other as dependencies. This confuses CloudFormation about which resource to create first.

To fix this error, I deleted references to the IAM policies of the IAM CodeBuild service role configuration.



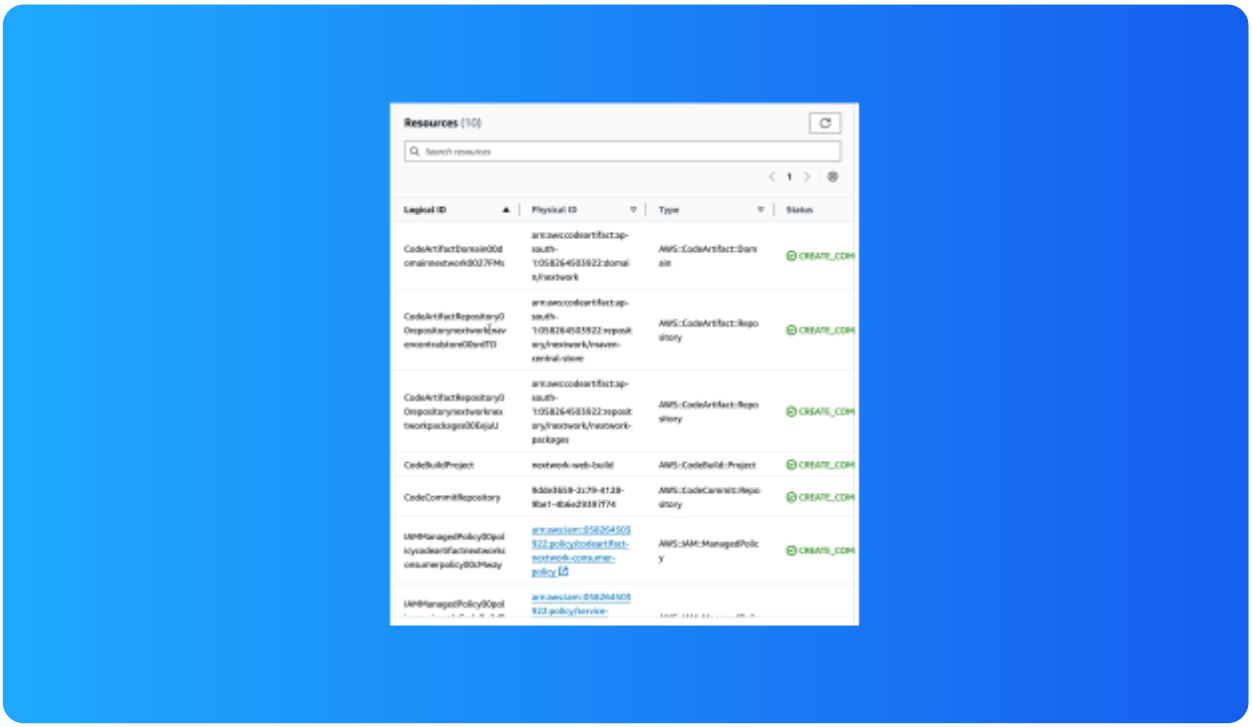
ZA
zandiletsh20@gmail.com
NextWork Student

My final template test

In my final test, creating the new stack was a great success

I could verify all the deployed resources by visiting the Resources tab in the CloudFormation Stack shows a list of all resources created and deployed from the template.

Not all the resources in the list had a shortcut URL, because CloudFormation cannot create the same shortcut link for resources deployed in a specific region.



Everyone should be in a job they love.

Check out nextwork.org for more projects