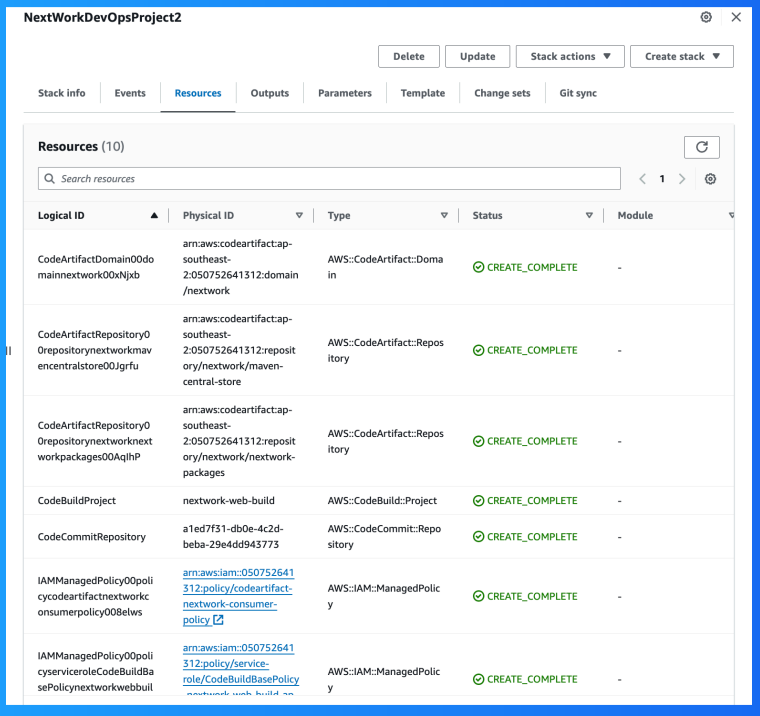
NextWork.org

Automate with Cloud Formation

tiubenedict@gmail.com



tiubenedict@gmail.com
NextWork Student

Introducing today's project!

What is AWS CloudFormation?

AWS CloudFormation is an Infrastructure as Code service, allowing you to create aws resource stacks programmatically from a text file template. It helps developers recreate stacks reliably.

How I used CloudFormation in this project

We used CloudFormation's IaC generator to scan our account for resources that it can base the configs from, and we used CloudFormation to re-launch those same resources using a template.

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

I didn't think that a CloudFormation template would be a bit difficult to set-up without using the IaC generator.

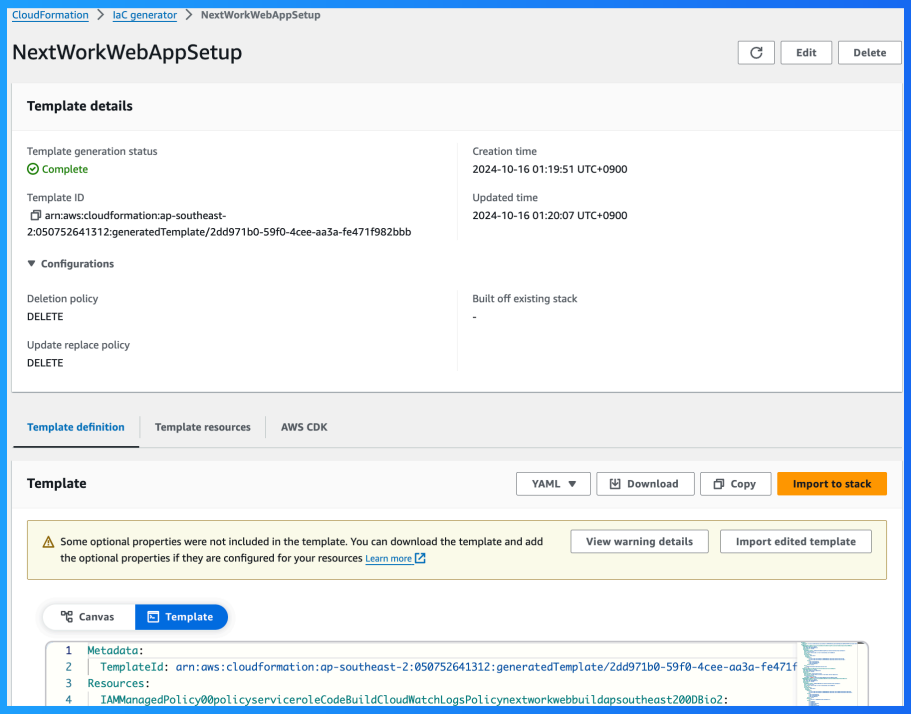
This project took me...

This project took me 60-90 minutes.

tiubenedict@gmail.com
NextWork Student

CloudFormation templates

A CloudFormation template is a text file that contains the configurations necessary for creating a resource stack in AWS.



tiubenedict@gmail.com
NextWork Student

IaC generator

I created a CloudFormation template using the IaC generator, which scans my account for resources that I can draw the configurations from.

Not all resources could be added to my template

The resources I couldn't add to a template were the source core respository (setting up a CodeCommit repo) and CodeBuild project.

The resources that I could add to my template includes the S3 bucket, IAM roles and policies, and the CodeArtifact domain and associated repositories.

tiubenedict@gmail.com
NextWork Student

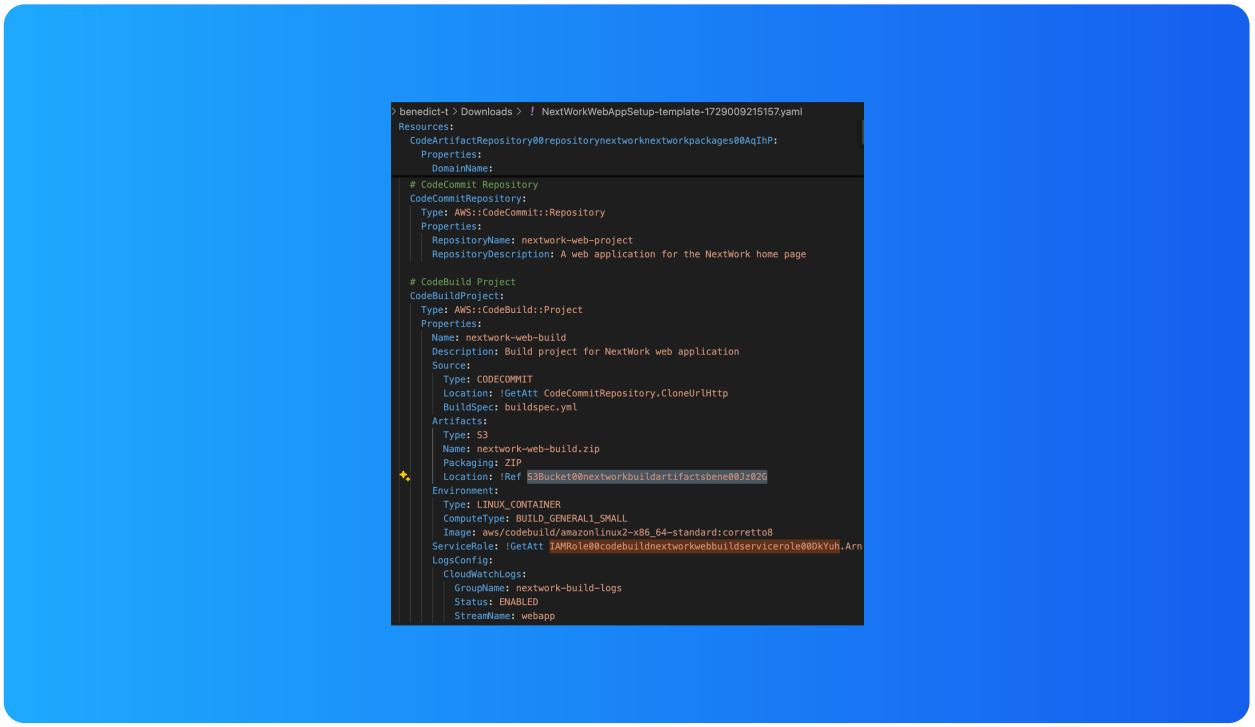
NextWork.org

Manually adding resources

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

I had to manually define these because setting up a repo can be more complicated such as Git configs or user permissions, while configurations also need to be set for the build environment.

I also had to make sure the references were consistent in this template, so I made two edits: the S3 bucket for hosting the build artifact, and the specific IAM role amazon resource name for the CodeBuild project.



tiubenedict@gmail.com
NextWork Student

Testing my template

Before testing my template, I deleted those same resources that were included in the template, so that there wouldn't be the redundant resources sharing the same name, which would prevent CloudFormation from creating them.

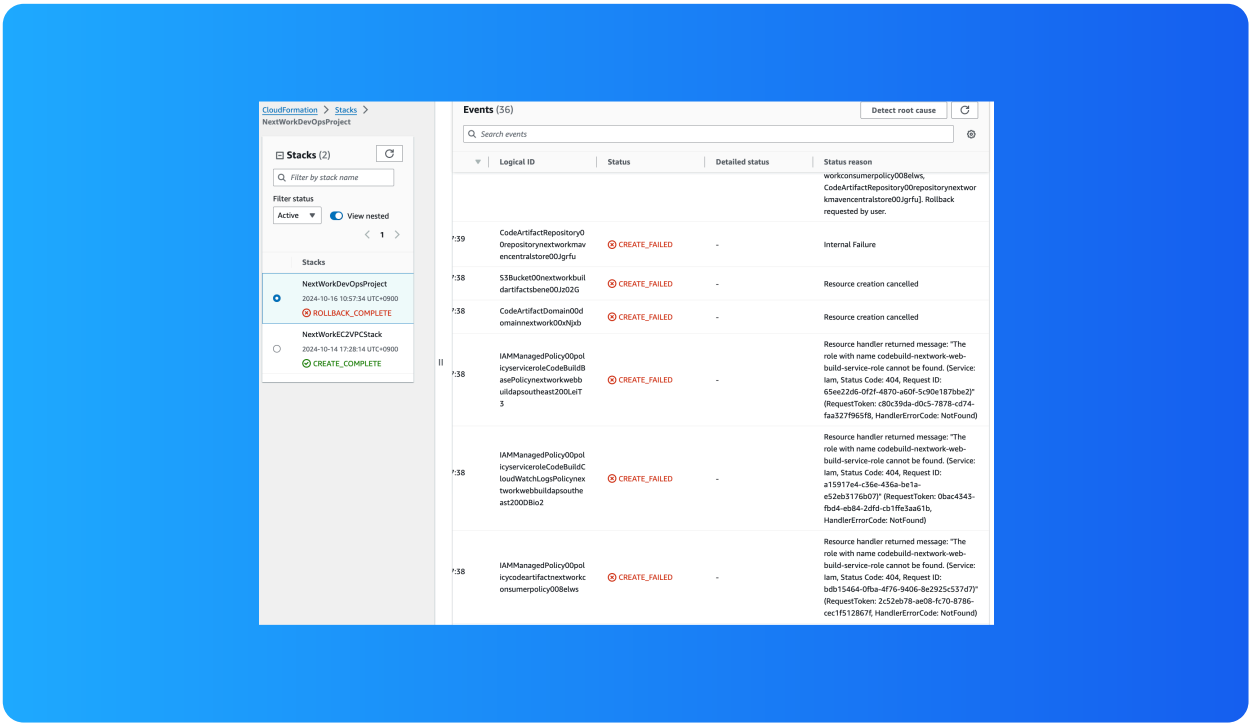
A stack is a collection of AWS resources, created together using the CloudFormation template.

The result of my first test was: an error from creating the IAM policies for CodeArtifact and CodeBuild.

tiubenedict@gmail.com
NextWork Student

Unpacking the first error

My first template test failed because the necessary permission template was not available yet to attach to the IAM policy, as it was also just in the process of being created in this template.



To fix this error, I edited my CloudFormation template.

I added the line DependsOn: IAMRole so that that role gets created first before the IAM managed policies, which need that role, are created.

tiubenedict@gmail.com
NextWork Student

Fixing the first error

The DependsOn attribute means that this resource has a dependency on another resource, so it must wait for that dependency to be created first to prevent errors.

The DependsOn line was added to four different parts of my template: the 3 IAM policies (1 for CodeArtifact, 2 for CodeBuild), and another for the CodeBuild project itself.

```
127   # IAM Policy for CodeArtifact
128   IAMManagedPolicy00policycodeartifactnextworkconsumerpolicy008elws:
129     UpdateReplacePolicy: "Delete"
130     Type: "AWS::IAM::ManagedPolicy"
131     DeletionPolicy: "Delete"
132     DependsOn: "IAMRole00codebuildnextworkwebbuildservicerole00DkYuh"
133     Properties:
134       ManagedPolicyName: "codeartifact-nextwork-consumer-policy"
135       Path: "/"
136       Description: "Provides permissions to read from CodeArtifact"
137       Groups: []
138       PolicyDocument:
139         Version: "2012-10-17"
140         Statement:
141         - Resource: "*"
142           Action:
143           - "codeartifact:GetAuthorizationToken"
144           - "codeartifact:GetRepositoryEndpoint"
145           - "codeartifact:ReadFromRepository"
146           Effect: "Allow"
147         - Condition:
148             StringEquals:
149               sts:AWSServiceName: "codeartifact.amazonaws.com"
150           Resource: "*"
151           Action: "sts:GetServiceBearerToken"
152           Effect: "Allow"
153       Roles:
154       - "codebuild-nextwork-web-build-service-role"
155       Users: []
156   # IAM Role for CodeBuild
157   IAMRole00codebuildnextworkwebbuildservicerole00DkYuh:
```

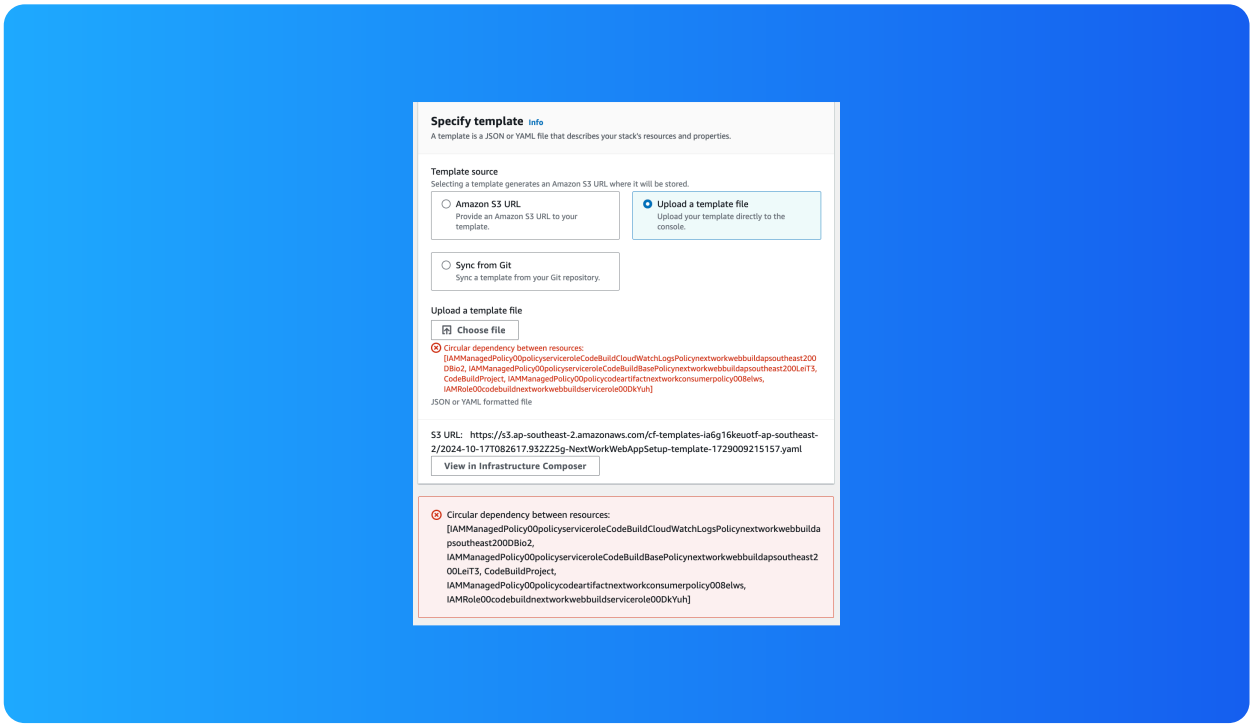
tiubenedict@gmail.com
NextWork Student

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 two or more resources reference each other in the config, so CloudFormation doesn't know which one needs to be created first. Here, the necessary role was needed for the policies, but role creation also references the policyies.

To fix this error, I removed the reference to the policies from the role creation section.



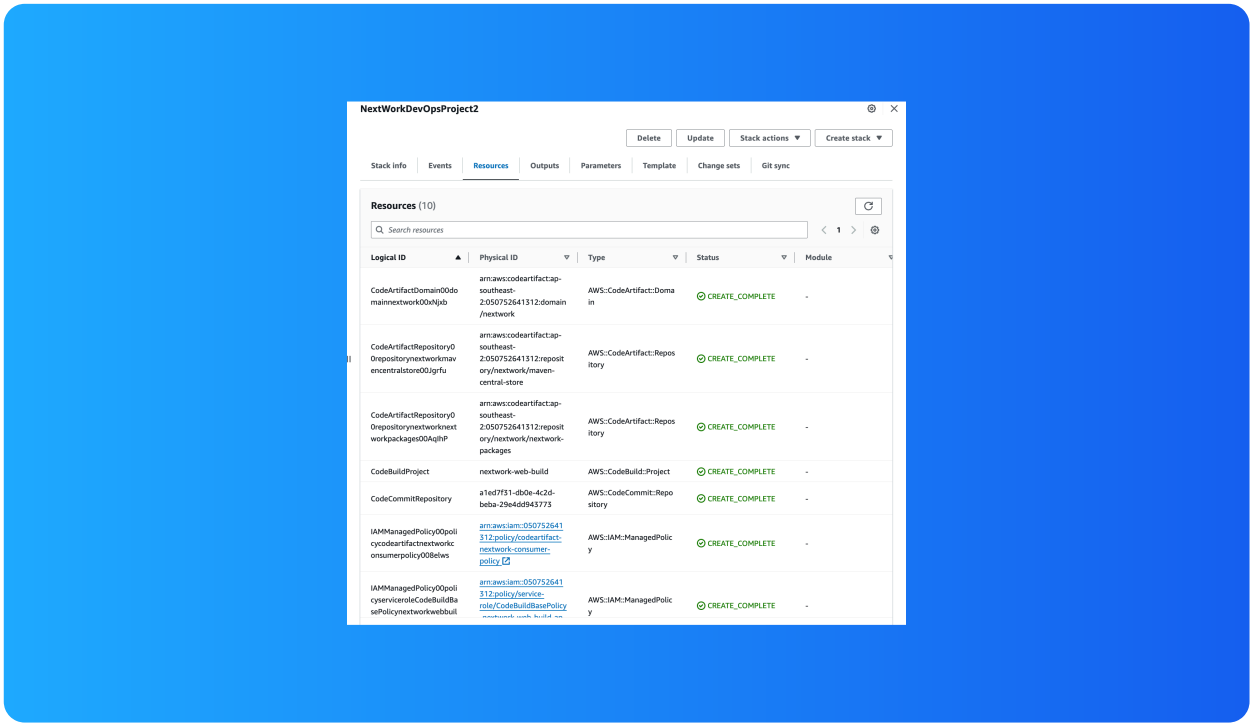
tiubenedict@gmail.com
NextWork Student

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 from within the CloudFormation stack.

Not all the resources in the list had a shortcut URL, because they were deployed to a specific region, not universally to my IAM account.



Everyone should be in a job they love.

Check out nextwork.org for more projects