

Building a CI/CD Pipeline with AWS CloudPipeline to deploy a static website on AWS S3

-Kavitha Bangalore

AWS CodePipeline provides a native AWS continuous deployment pipeline to manage web application deployments from the source code repository to the deployment of the web application.

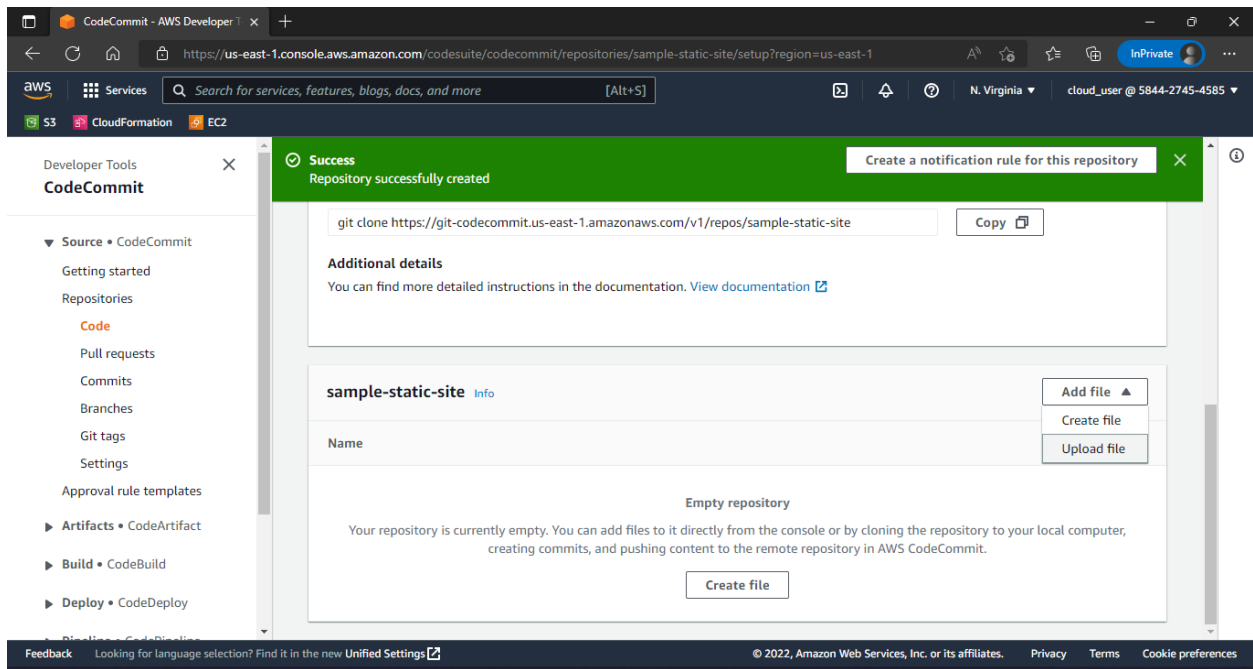
Below is a high-level architecture diagram of how the CI/CD Pipeline with AWS CodePipeline to Deploy a Static Website on AWS:



Building a CI/CD Pipeline with AWS CloudPipeline to deploy a static website on AWS S3

-Kavitha Bangalore

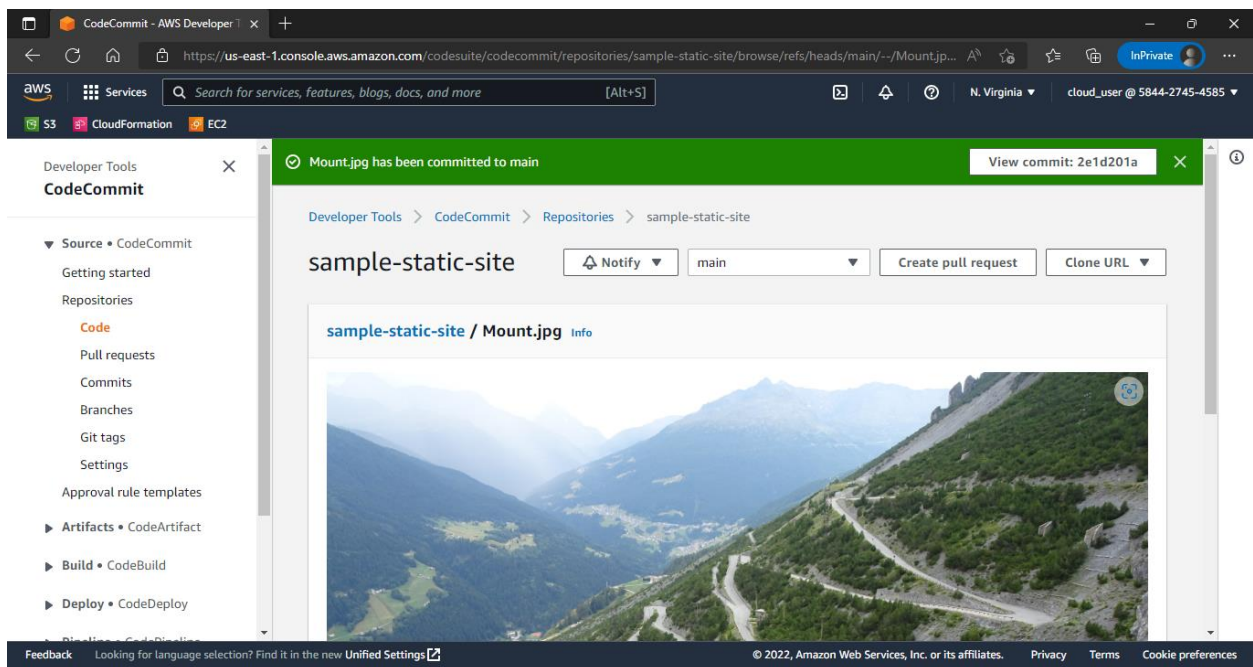
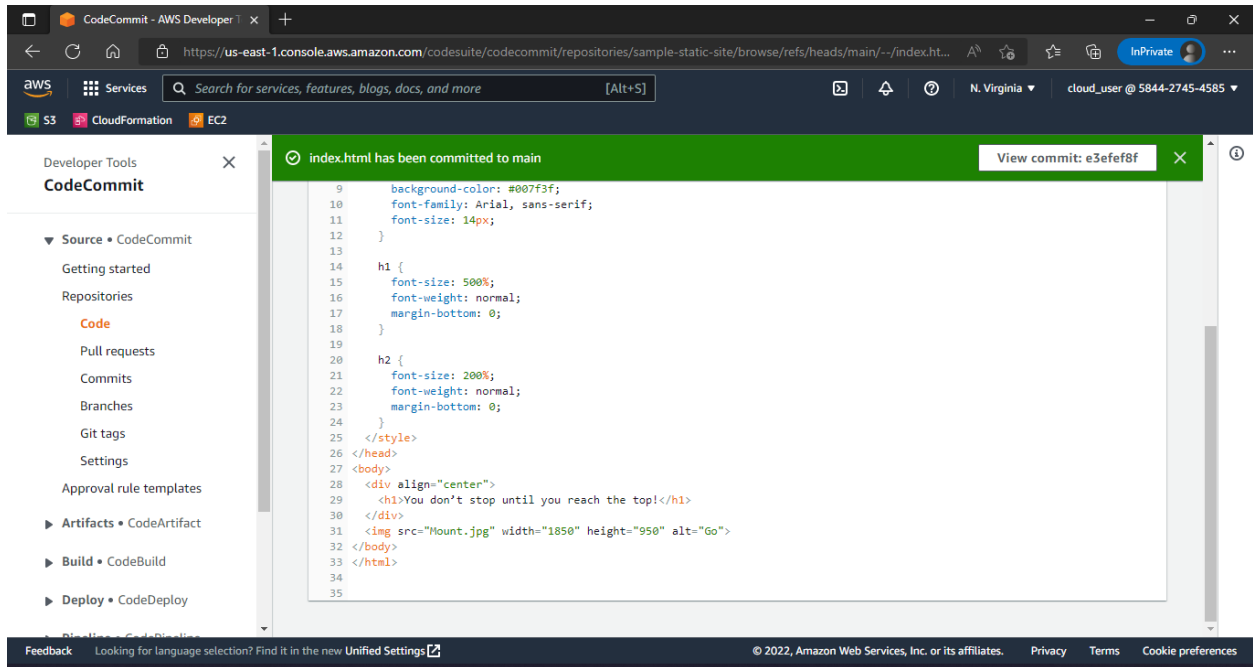
STEP 1: First, create a Repo using CodeCommit from the AWS Management Console.



STEP 2: Upload files using CodeCommit – index.html and mount.jpg

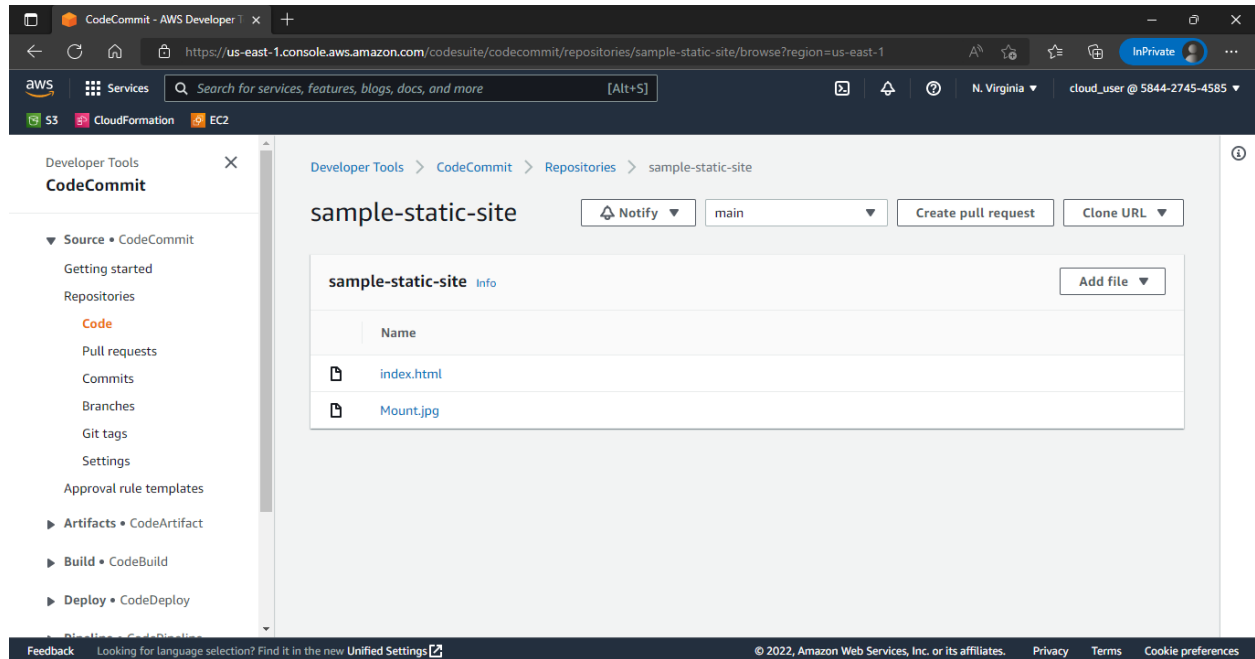
Building a CI/CD Pipeline with AWS CloudPipeline to deploy a static website on AWS S3

-Kavitha Bangalore



Building a CI/CD Pipeline with AWS CloudPipeline to deploy a static website on AWS S3

-Kavitha Bangalore

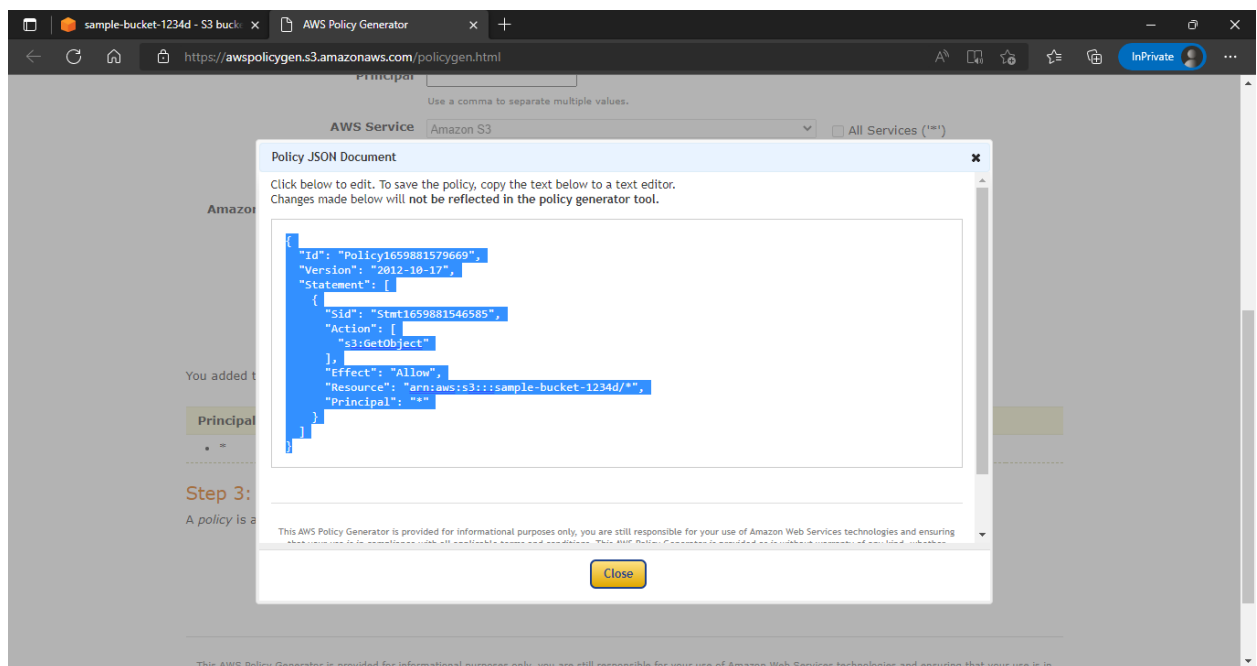
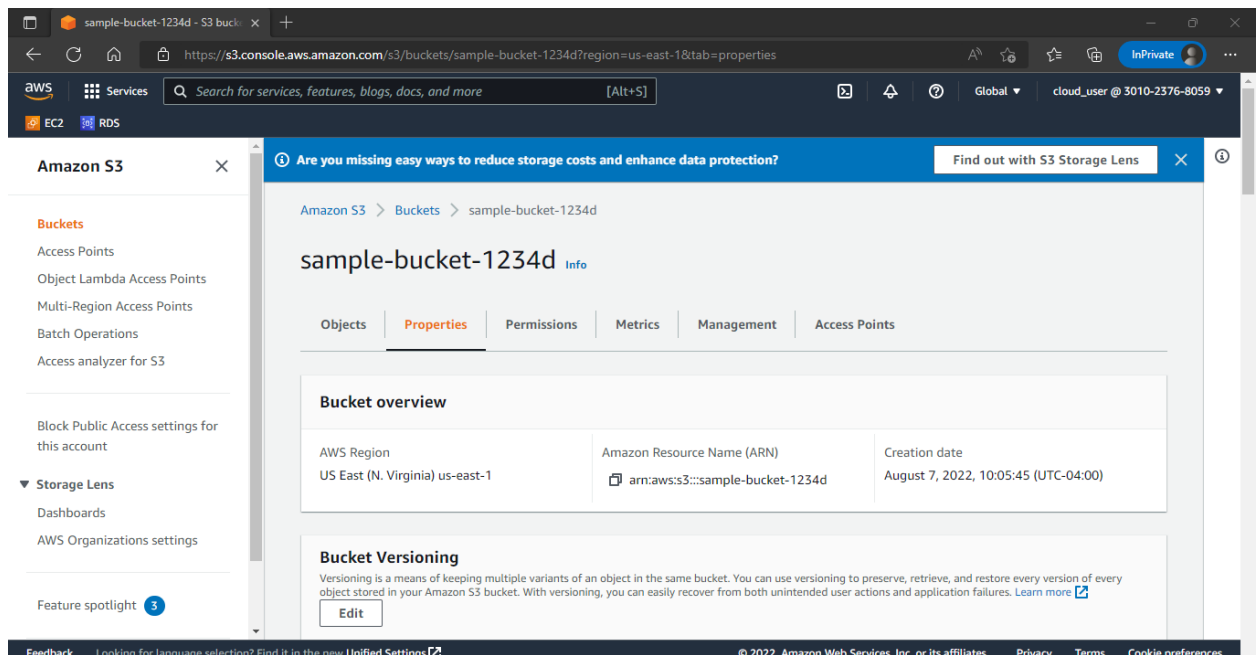


STEP 3: Next setup the S3 bucket and the bucket policy

1. When creating the S3 bucket, enable Static Website Hosting.

Building a CI/CD Pipeline with AWS CloudPipeline to deploy a static website on AWS S3

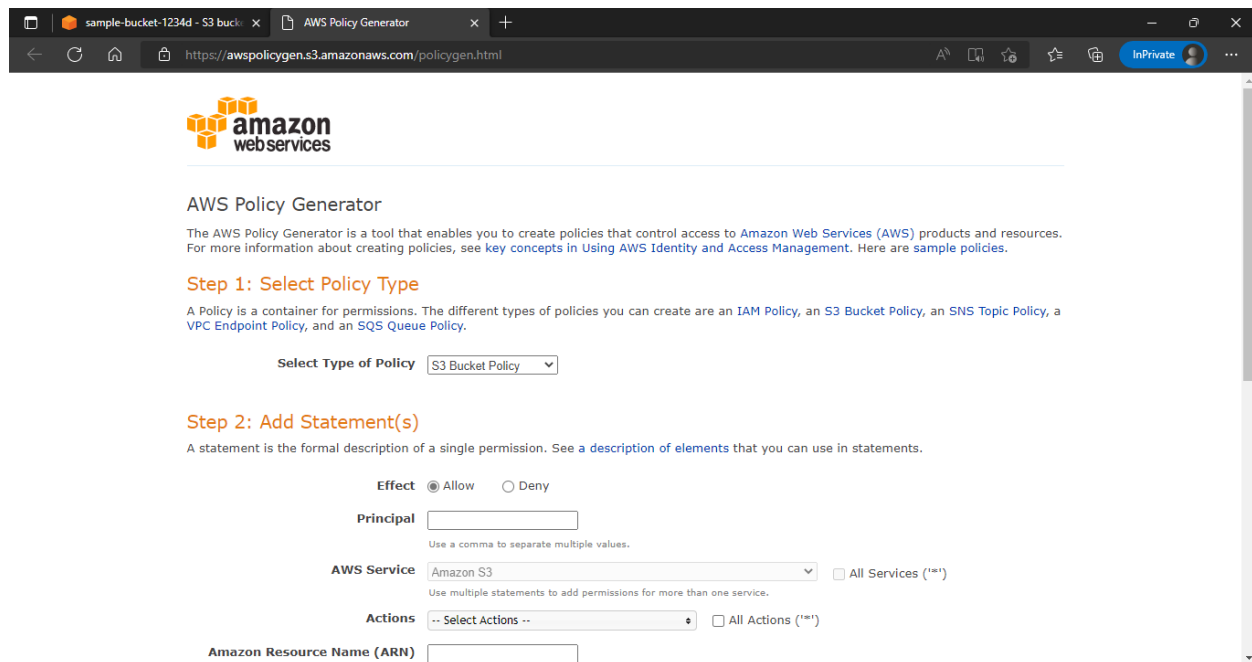
-Kavitha Bangalore



2. In the S3 Bucket under permissions, select the Policy Generator and the Action: GetObject, The Bucket ARN number can be got from the Bucket main page.

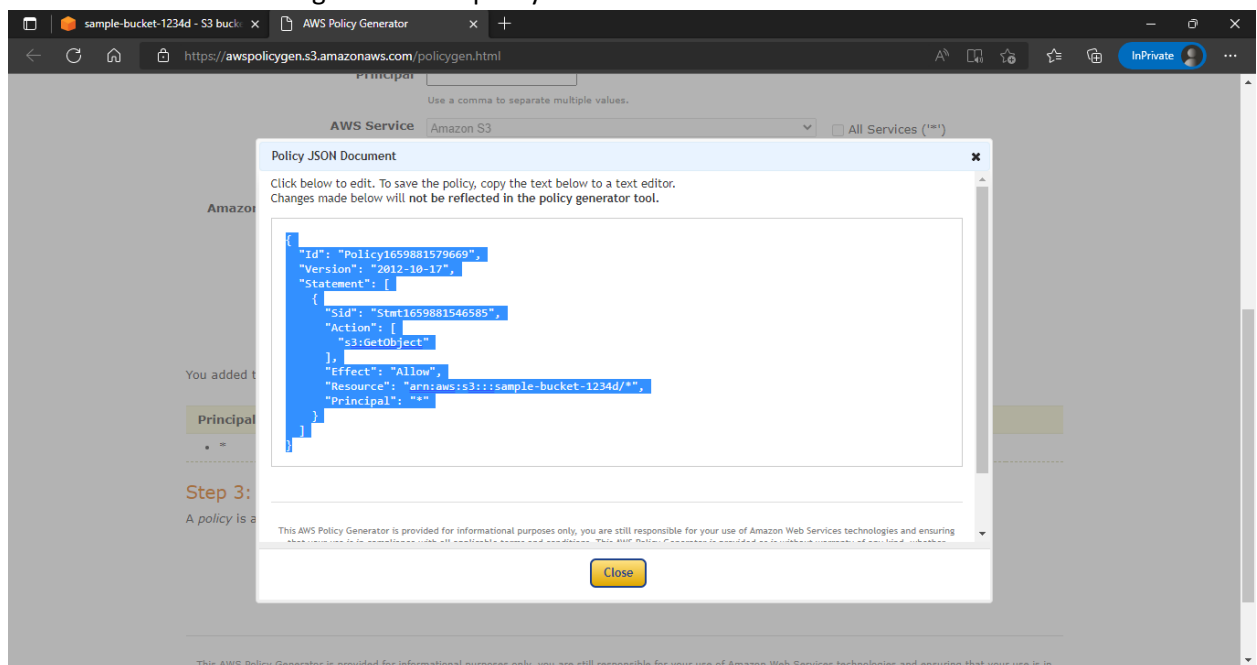
Building a CI/CD Pipeline with AWS CloudPipeline to deploy a static website on AWS S3

-Kavitha Bangalore



The screenshot shows the AWS Policy Generator interface. At the top, the Amazon Web Services logo is displayed. Below it, the title "AWS Policy Generator" is followed by a brief description: "The AWS Policy Generator is a tool that enables you to create policies that control access to Amazon Web Services (AWS) products and resources. For more information about creating policies, see [key concepts in Using AWS Identity and Access Management](#). Here are [sample policies](#)." The interface is divided into two main steps. **Step 1: Select Policy Type** includes a dropdown menu labeled "Select Type of Policy" with "S3 Bucket Policy" selected. **Step 2: Add Statement(s)** includes fields for "Effect" (radio buttons for "Allow" and "Deny", with "Allow" selected), "Principal" (a text input field), "AWS Service" (a dropdown menu with "Amazon S3" selected and a checkbox for "All Services (*)"), "Actions" (a dropdown menu with "-- Select Actions --" and a checkbox for "All Actions (*)"), and "Amazon Resource Name (ARN)" (a text input field).

3. Then Add Statement and generate the policy.



The screenshot shows the AWS Policy Generator interface with a modal window titled "Policy JSON Document" open. The modal contains the following JSON text:

```
{
  "Id": "Policy1659881579669",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1659881546585",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::sample-bucket-1234d/*",
      "Principal": "*"
    }
  ]
}
```

 Below the JSON text, there is a disclaimer: "This AWS Policy Generator is provided for informational purposes only, you are still responsible for your use of Amazon Web Services technologies and ensuring that your use is in accordance with all applicable laws and regulations. This tool does not generate a policy document that is ready to be used in the AWS IAM console." A "Close" button is located at the bottom right of the modal. The background of the screenshot shows the "Step 3: Add Statement" section of the AWS Policy Generator, which includes a "Principal" dropdown menu and a "Statement" list.

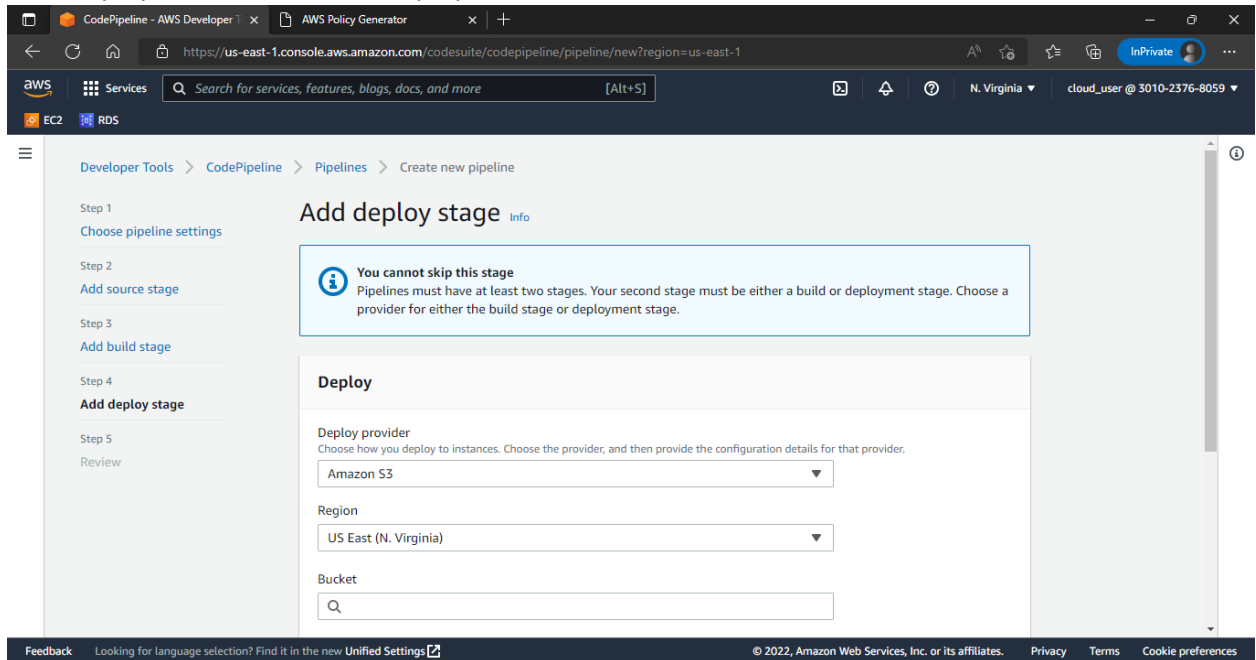
4. Copy the policy and paste it in the S3 bucket policy editor.

STEP 4 : Next create the CodePipeline

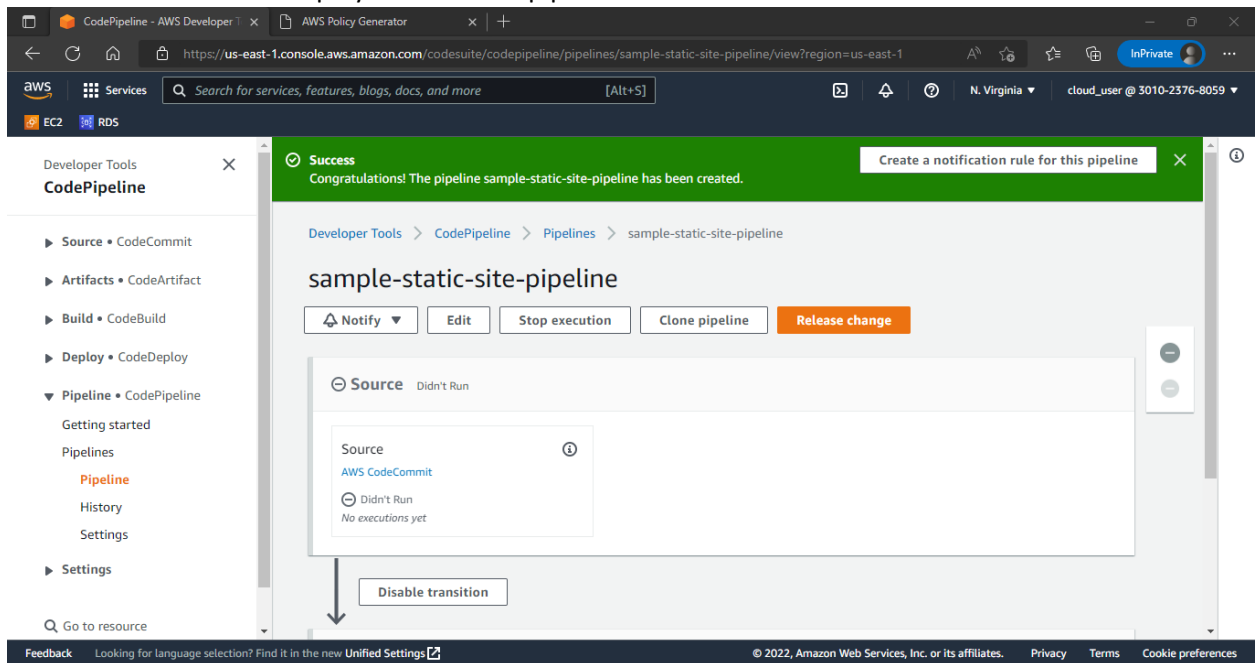
Building a CI/CD Pipeline with AWS CloudPipeline to deploy a static website on AWS S3

-Kavitha Bangalore

1. The CodePipeline as indicated in the architecture diagram uses the CodeCommit and CodeDeploy services of AWS to deploy the static website.



2. Then extract the file to deploy and create a pipeline.



3. Click the bucket website endpoint URL to view the static website.

Building a CI/CD Pipeline with AWS CloudPipeline to deploy a static website on AWS S3

-Kavitha Bangalore

