ASSIGNMENT 2

Aim: Implement AWS CodePipeline

Theory:

AWS CodePipeline is a fully managed [continuous delivery](https://aws.amazon.com/devops/continuous-delivery/) service that helps you automate your release pipelines for fast and reliable application and infrastructure updates. CodePipeline automates the build, test, and deploy phases of your release process every time there is a code change, based on the release model you define. This enables you to rapidly and reliably deliver features and updates. You can easily integrate AWS CodePipeline with third-party services such as GitHub or with your own custom plugin. The CodePipeline console also provides a way to quickly search for your resources, such as repositories, build projects, deployment applications, and pipelines. Choose Go to resource or press the / key, and then type the name of the resource. Any matches appear in the list. Searches are case insensitive. You only see resources that you have permissions to view. For more information, see [Viewing resources in the console](https://docs.aws.amazon.com/codepipeline/latest/userguide/security-iam-resources-console.html).

Before you can use AWS CodePipeline for the first time, you must complete the following steps.

Steps:

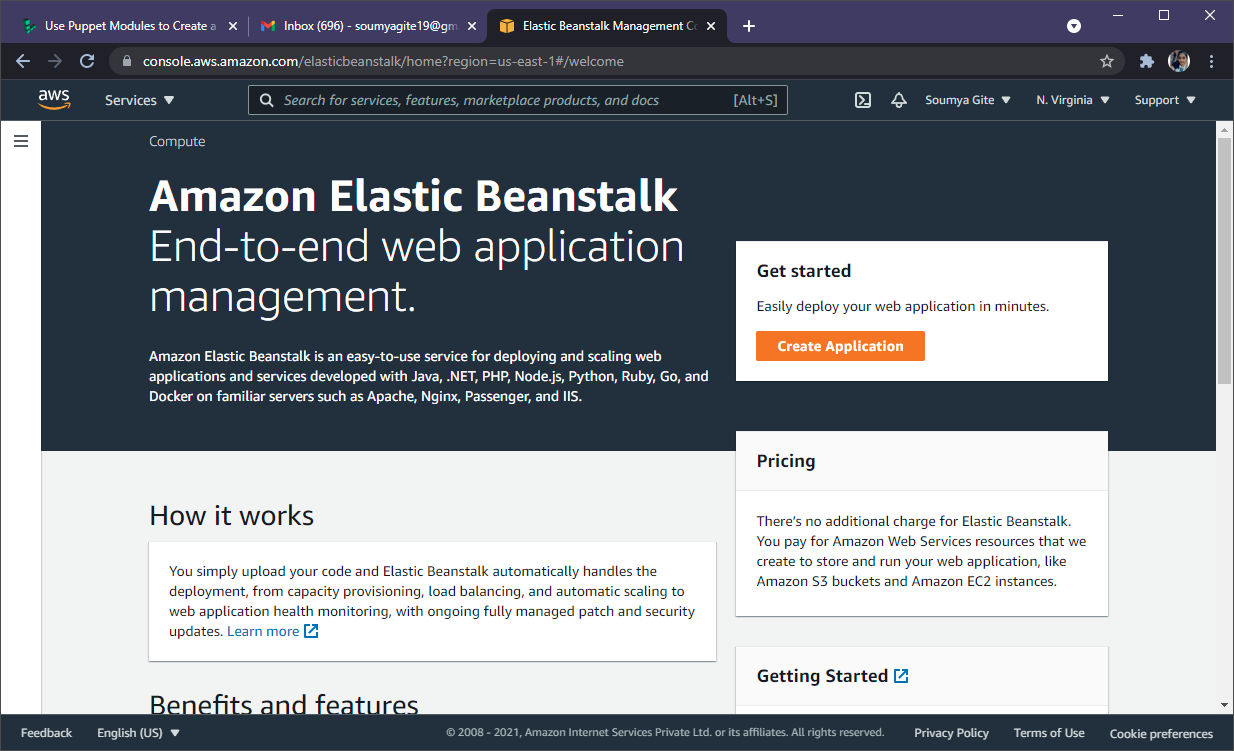
* [Step 1: Create an AWS account](https://docs.aws.amazon.com/codepipeline/latest/userguide/getting-started-codepipeline.html#create-aws-account)
* [Step 2: Create or use an IAM user](https://docs.aws.amazon.com/codepipeline/latest/userguide/getting-started-codepipeline.html#create-iam-user)
* [Step 3: Use an IAM managed policy to assign CodePipeline permissions to the IAM user](https://docs.aws.amazon.com/codepipeline/latest/userguide/getting-started-codepipeline.html#assign-permissions)
* [Step 4: Install the AWS CLI](https://docs.aws.amazon.com/codepipeline/latest/userguide/getting-started-codepipeline.html#install-cli)
* [Step 5: Open the console for CodePipeline](https://docs.aws.amazon.com/codepipeline/latest/userguide/getting-started-codepipeline.html#open-codepipeline-console)
* [Next steps](https://docs.aws.amazon.com/codepipeline/latest/userguide/getting-started-codepipeline.html#next-steps)

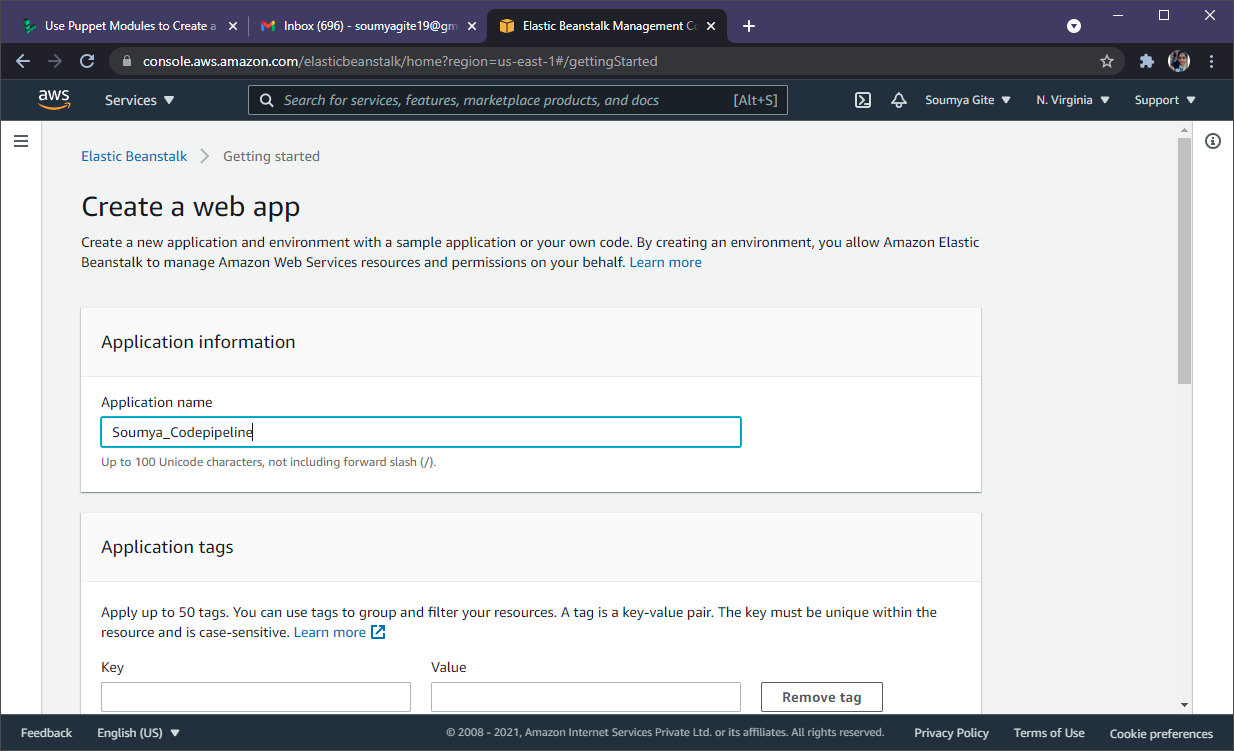
Advantages:

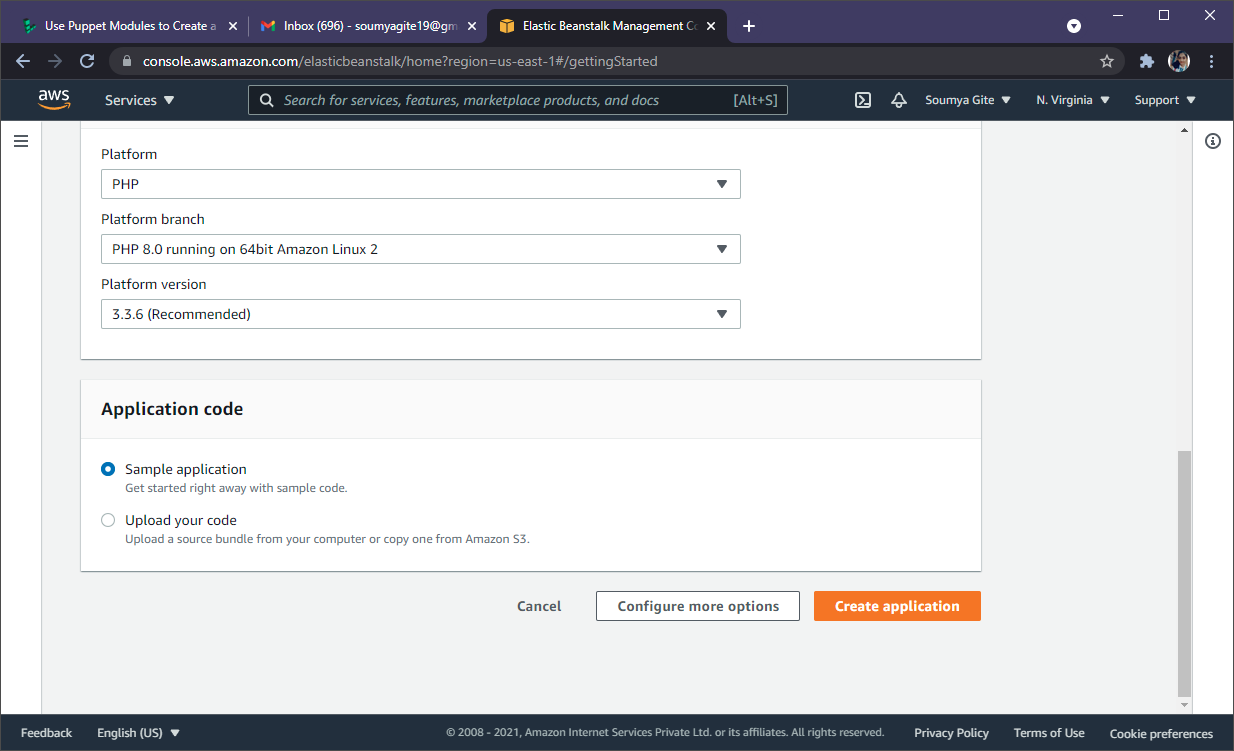
* Within the AWS architecture. You do not have to expose keys to the outside world to be able to trigger builds. If your CI/CD provider lies outside and gets hacked exposing AWS keys, the hacker can do anything with your AWS account.
* Pipeline as Code: Any Infrastructure as Code solution for AWS (such as Terraform or Cloud Formation) would also work for Codepipeline. Additionally, within the Codepipeline, AWS allows you to configure the build part with the help of yaml files.
* Managed by AWS: You do not have to install Codepipeline yourself, or even provision a set of agents. AWS does all of that for you.
* Fairly powerful: It gives you most features that other CI/CD providers have.

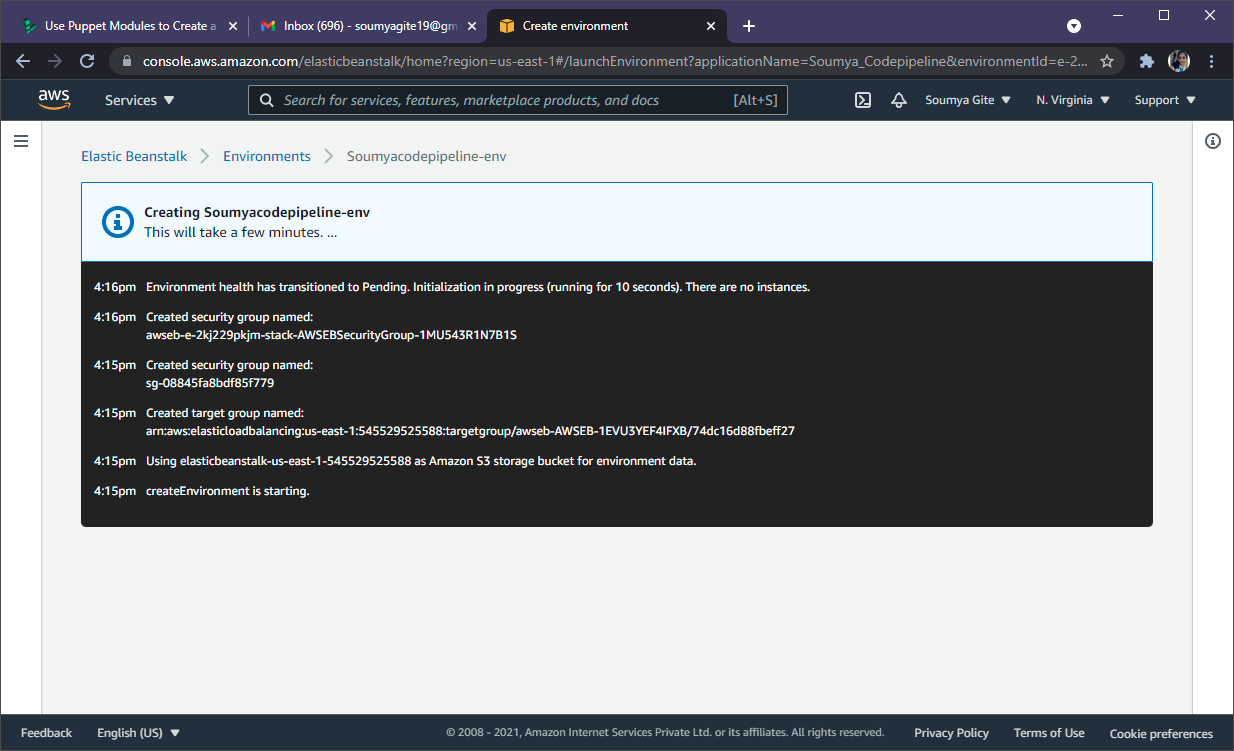
Disadvantages:

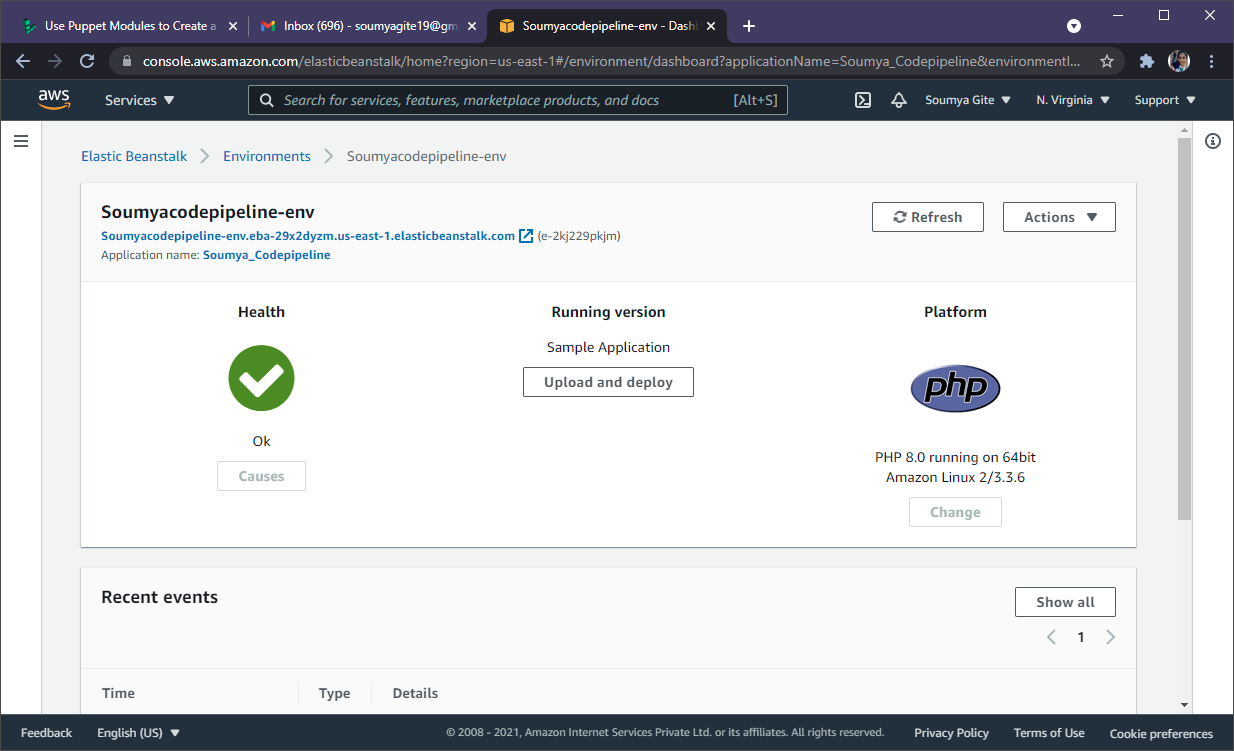
* The console UI and the overall usability is bad. With CI/CD systems, usability is often the difference between people following practices diligently and not following them.
* It forces unnecessary rules. For example, you have to specify a source, which has to be either S3, AWS CodeCommit, GitHub or a few others. If you want to pick up the source from a custom location, you have to come up with hacks such as specify a dummy source and then create a CodeBuild stage to pull the custom source.
* Codepipeline has to be composed of multiple AWS services like CodeCommit, CodeBuild, Code deploys etc. This makes it extremely complicated and overly dependent on AWS. For example, the build stage has to be a CodeBuild module. CodeBuild is another AWS service, which you can assume is similar to a build agent.
* It is a walled garden. The agents will be on AWS, the infra will be on AWS, the CI/CD server is on AWS. It makes all your infra completely dependent on AWS.

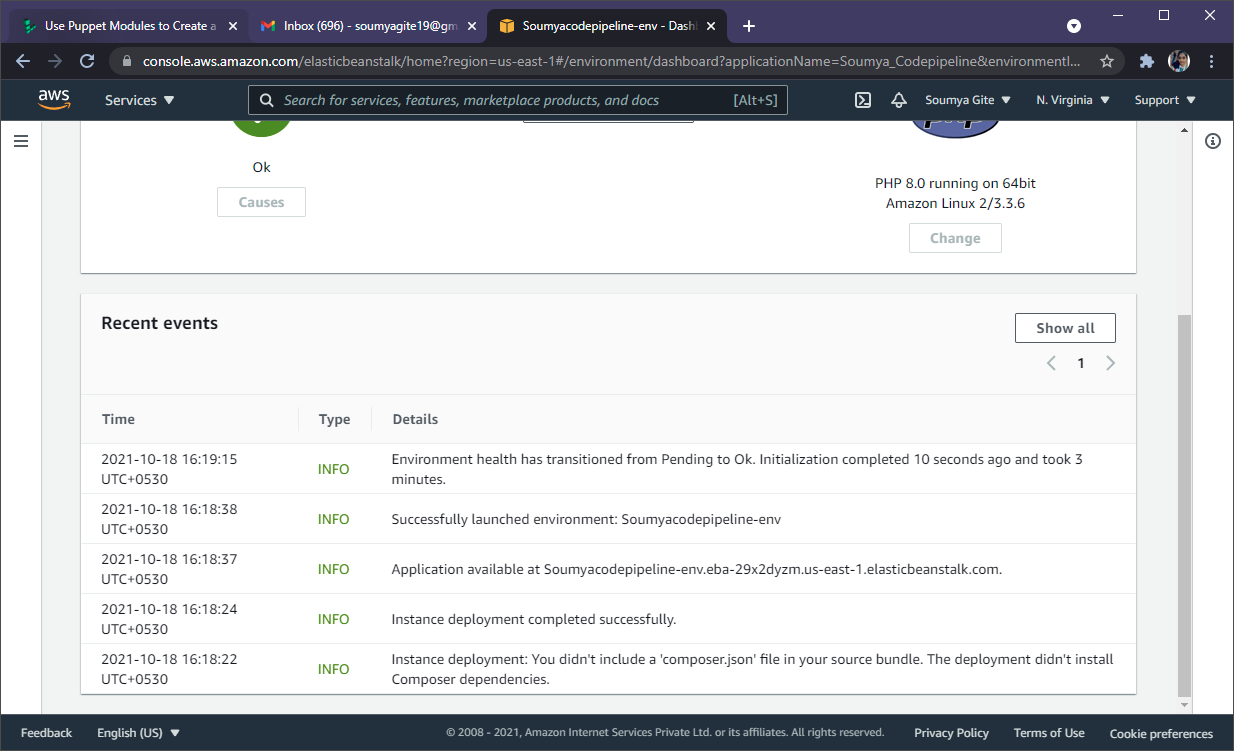


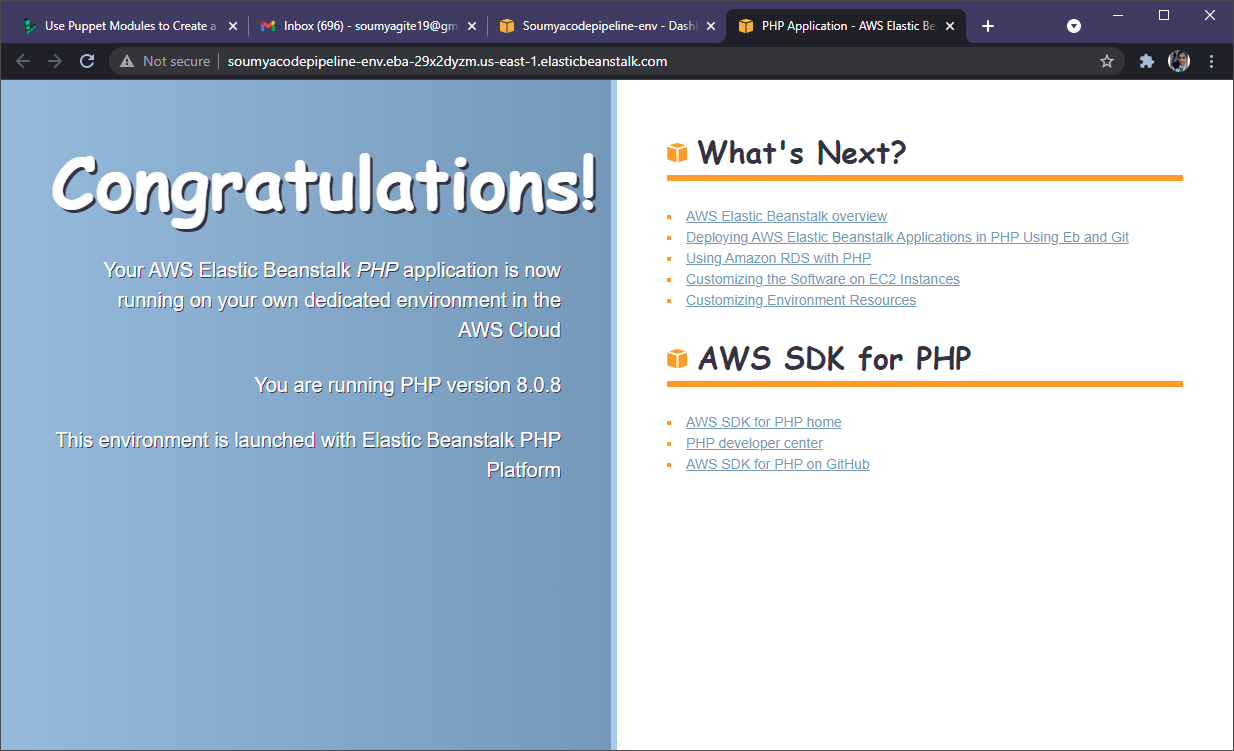


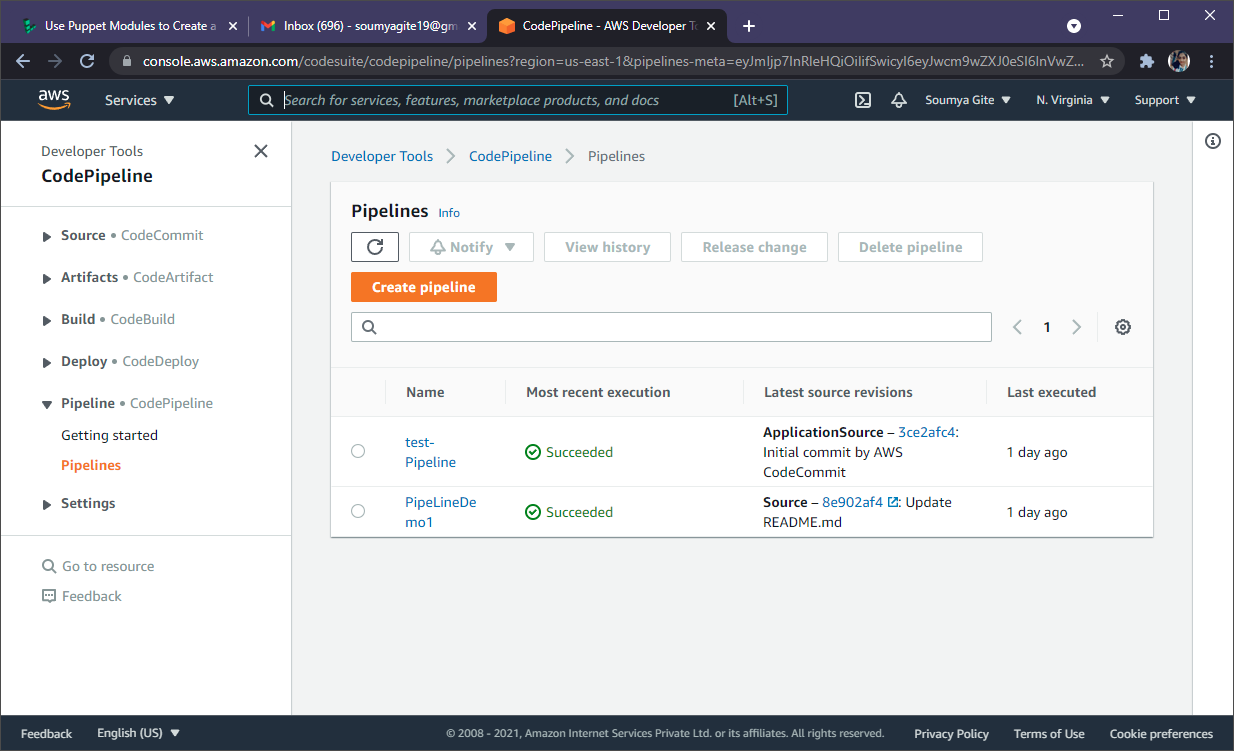


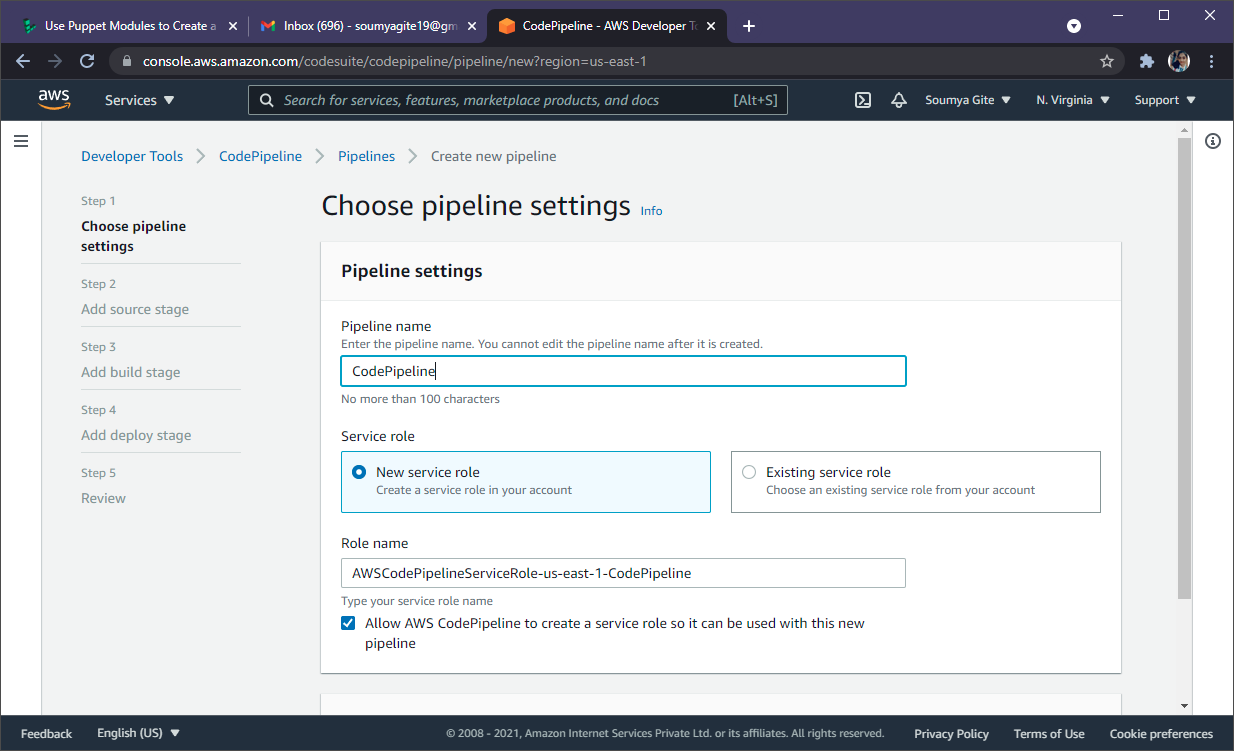


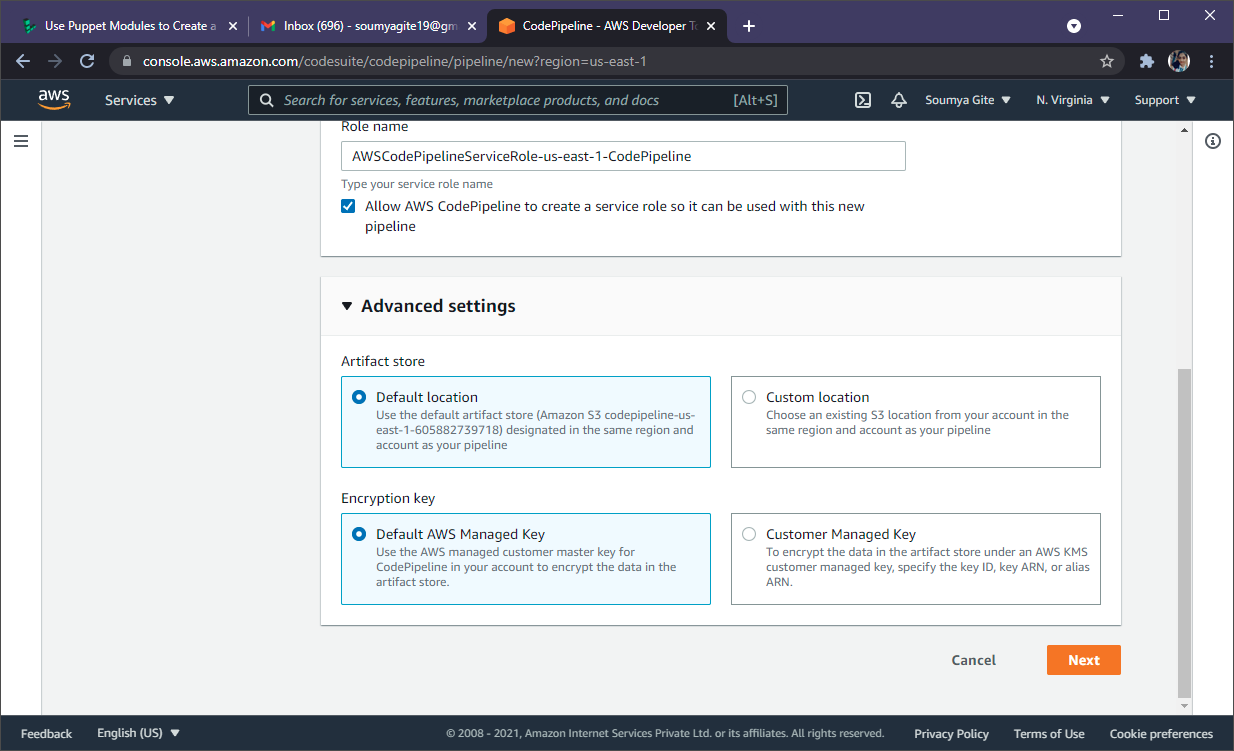


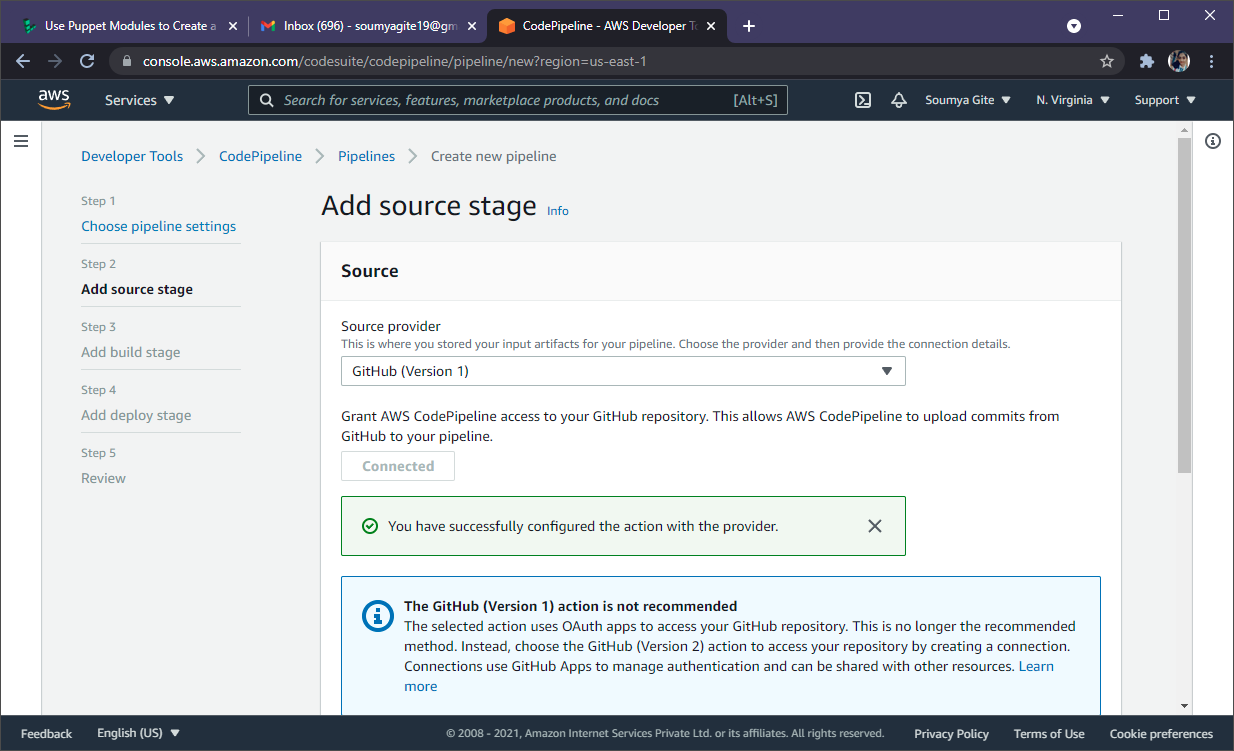


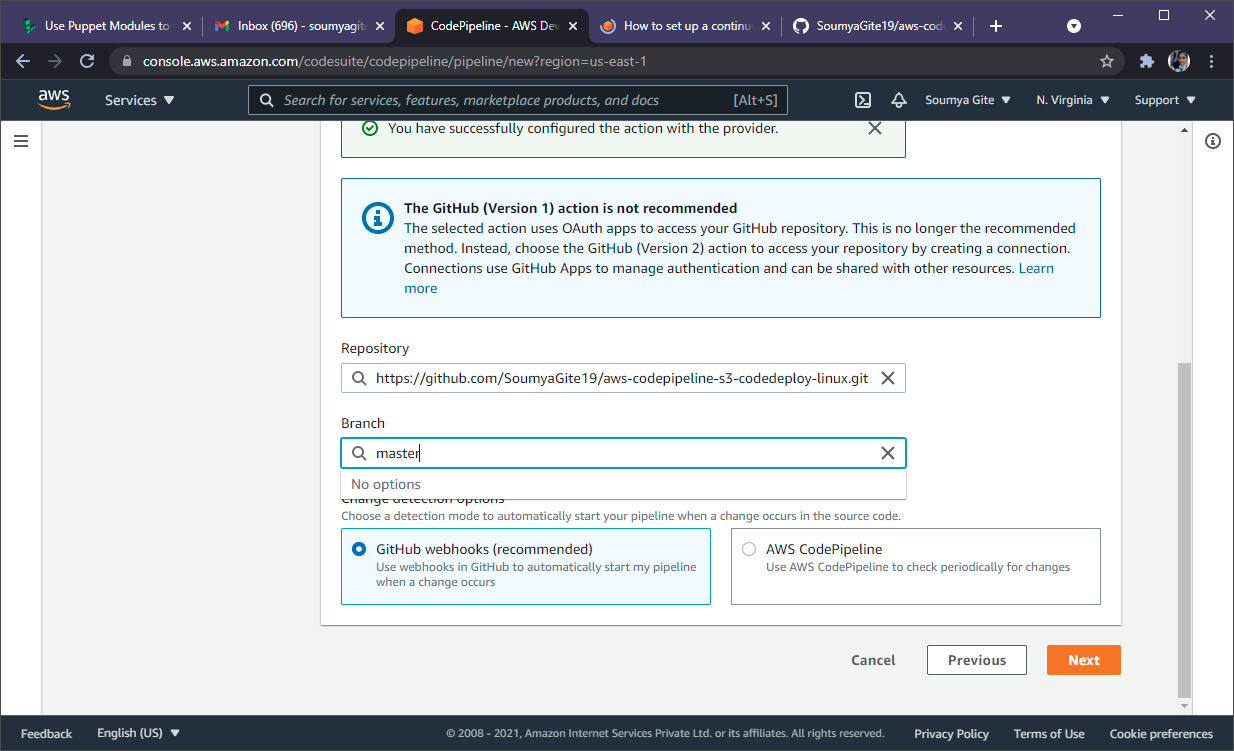


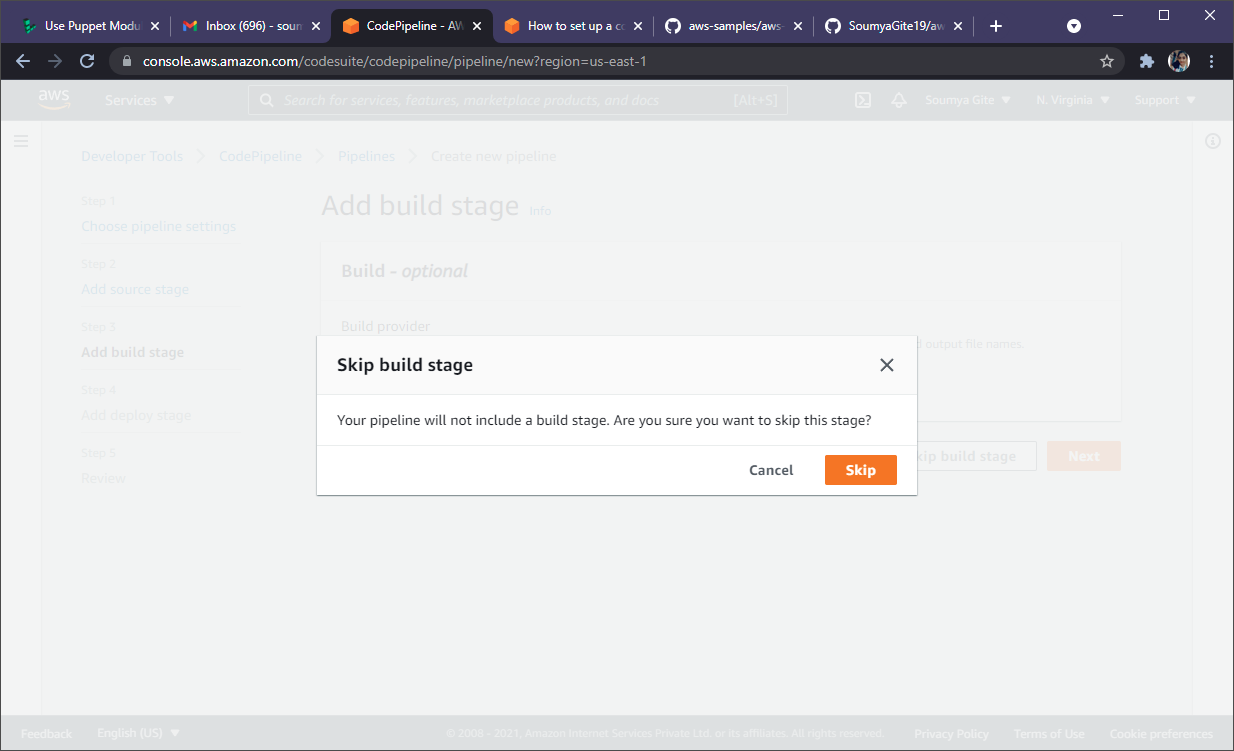


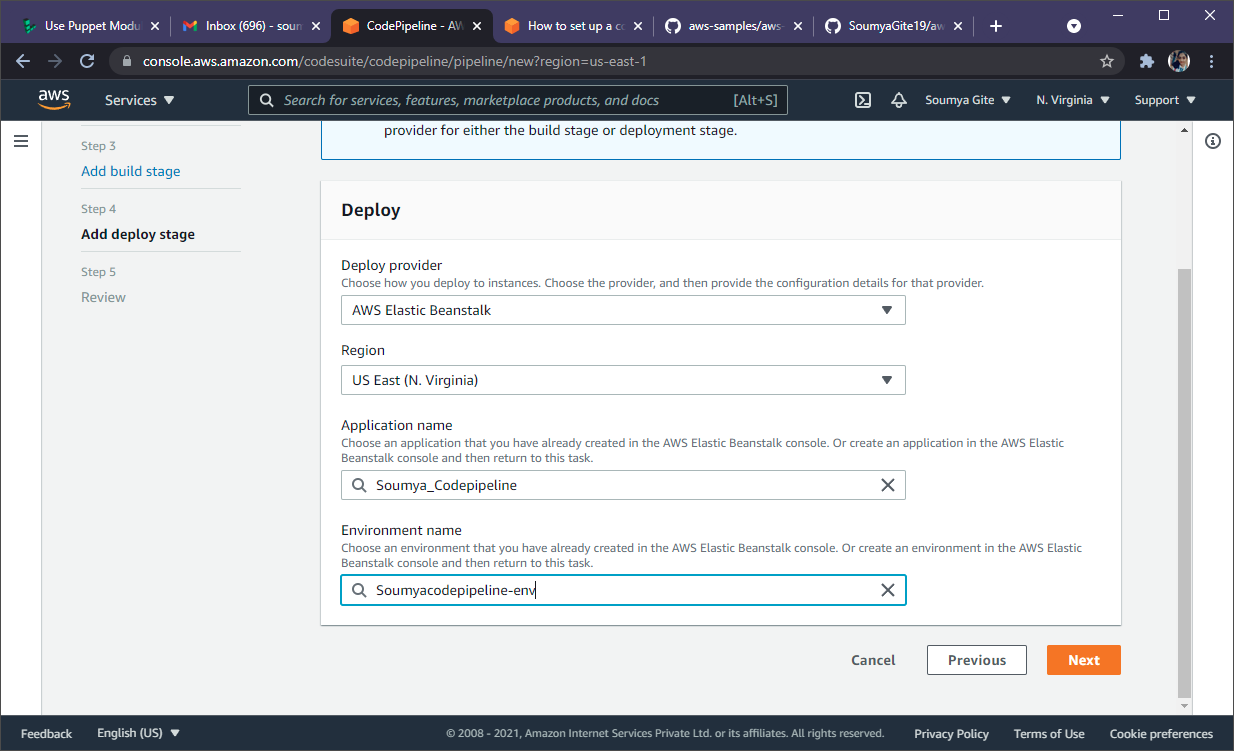


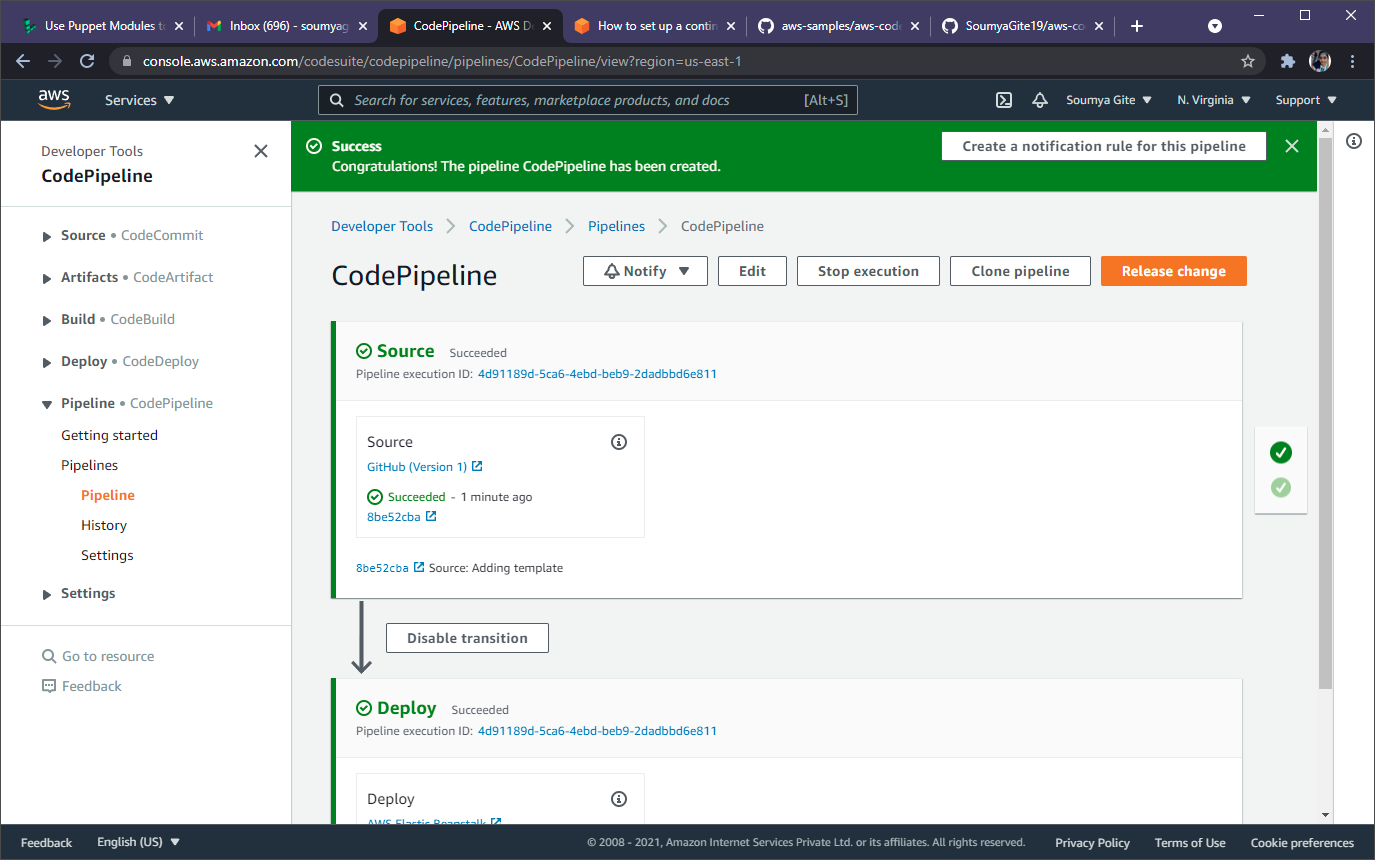


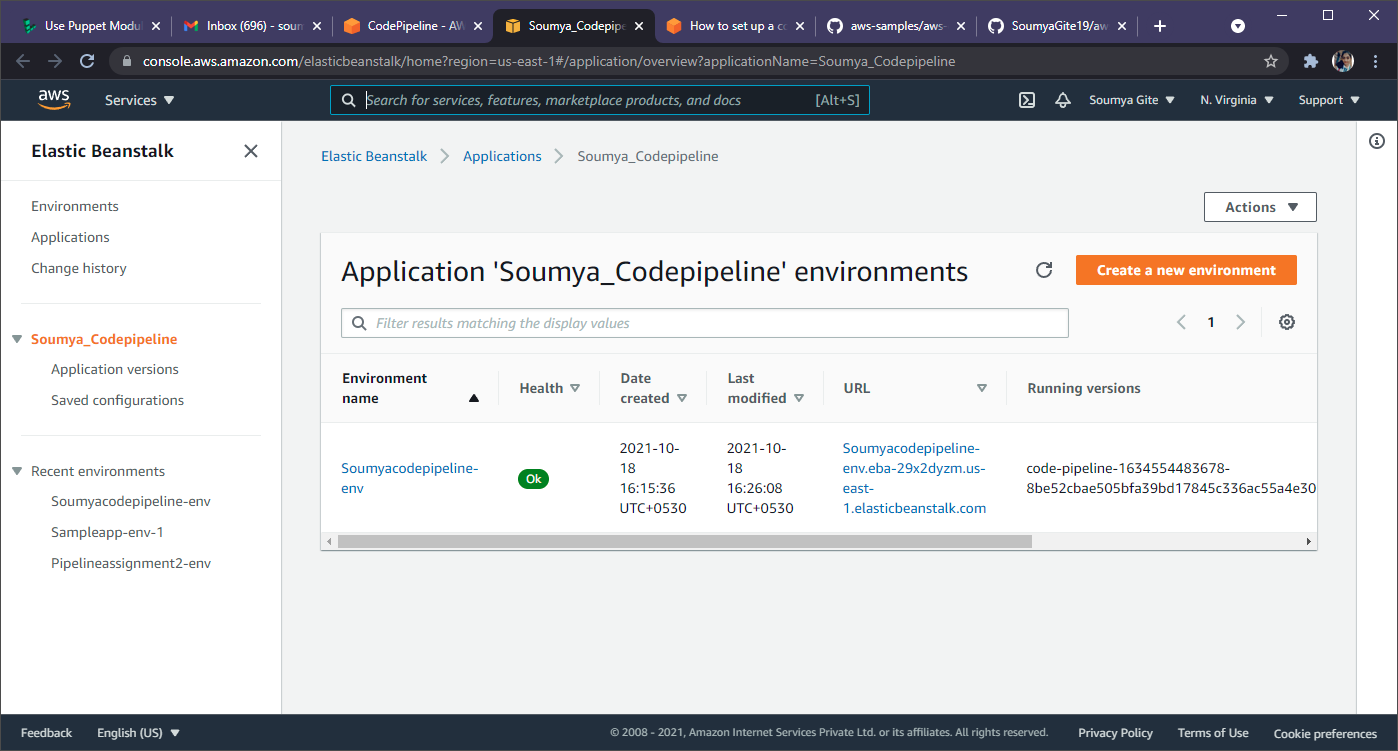


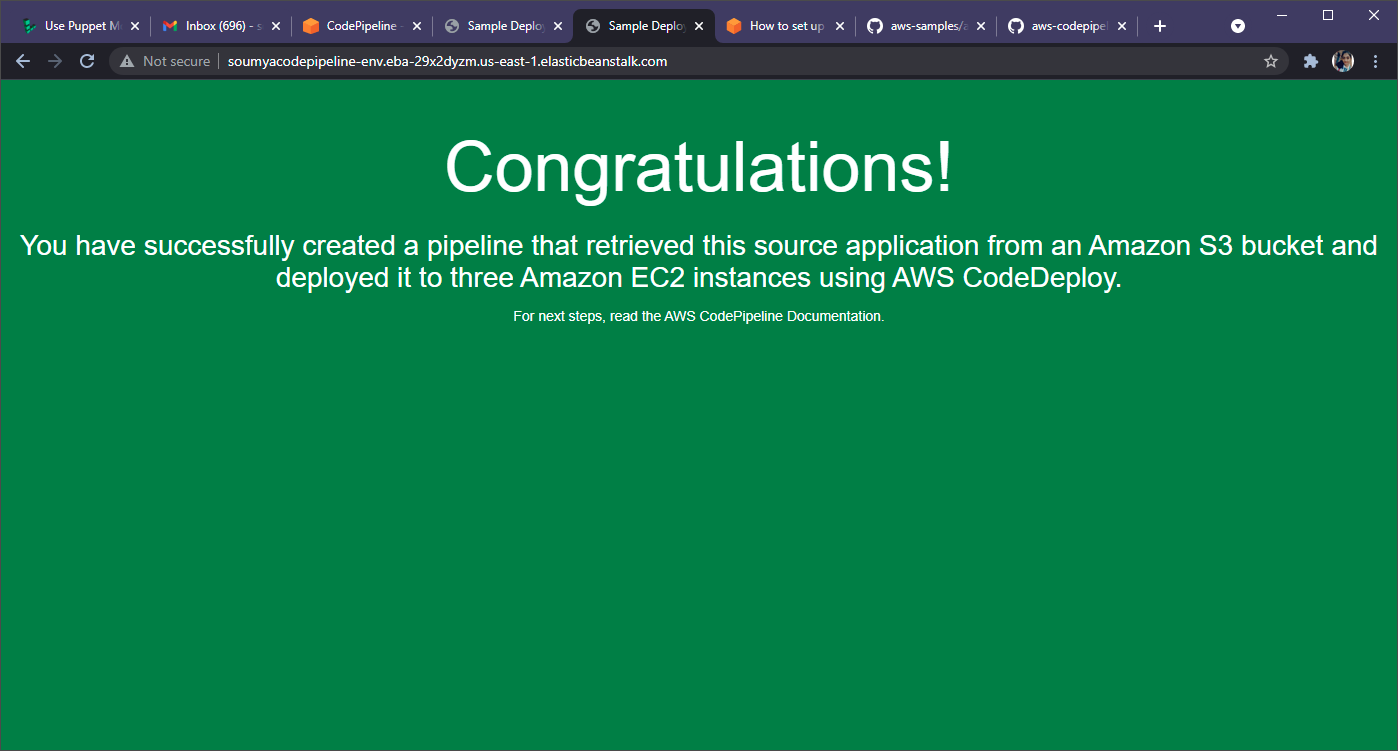


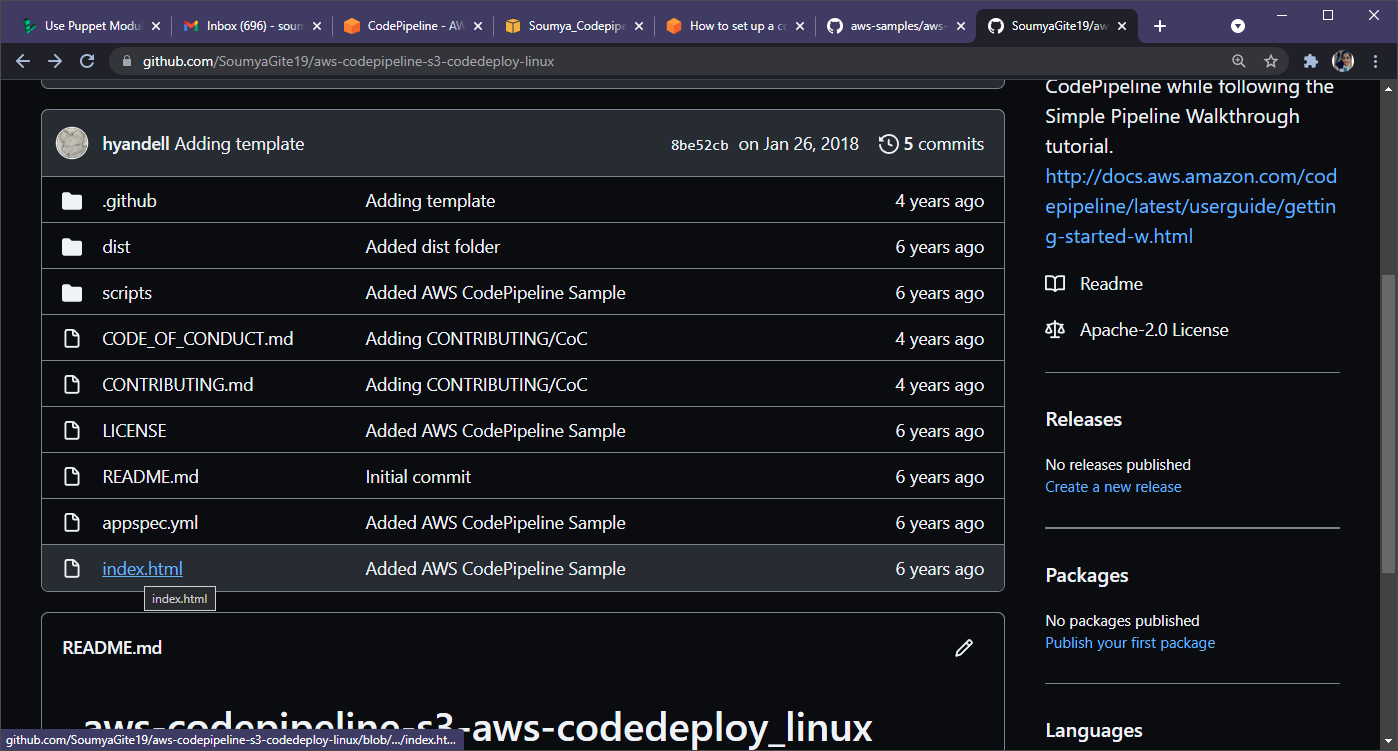


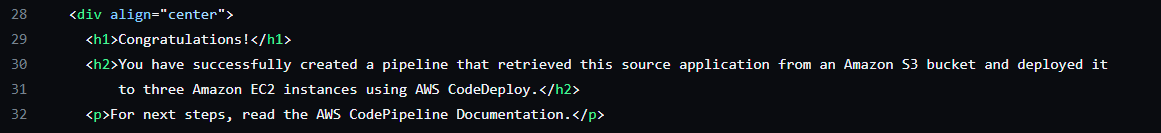


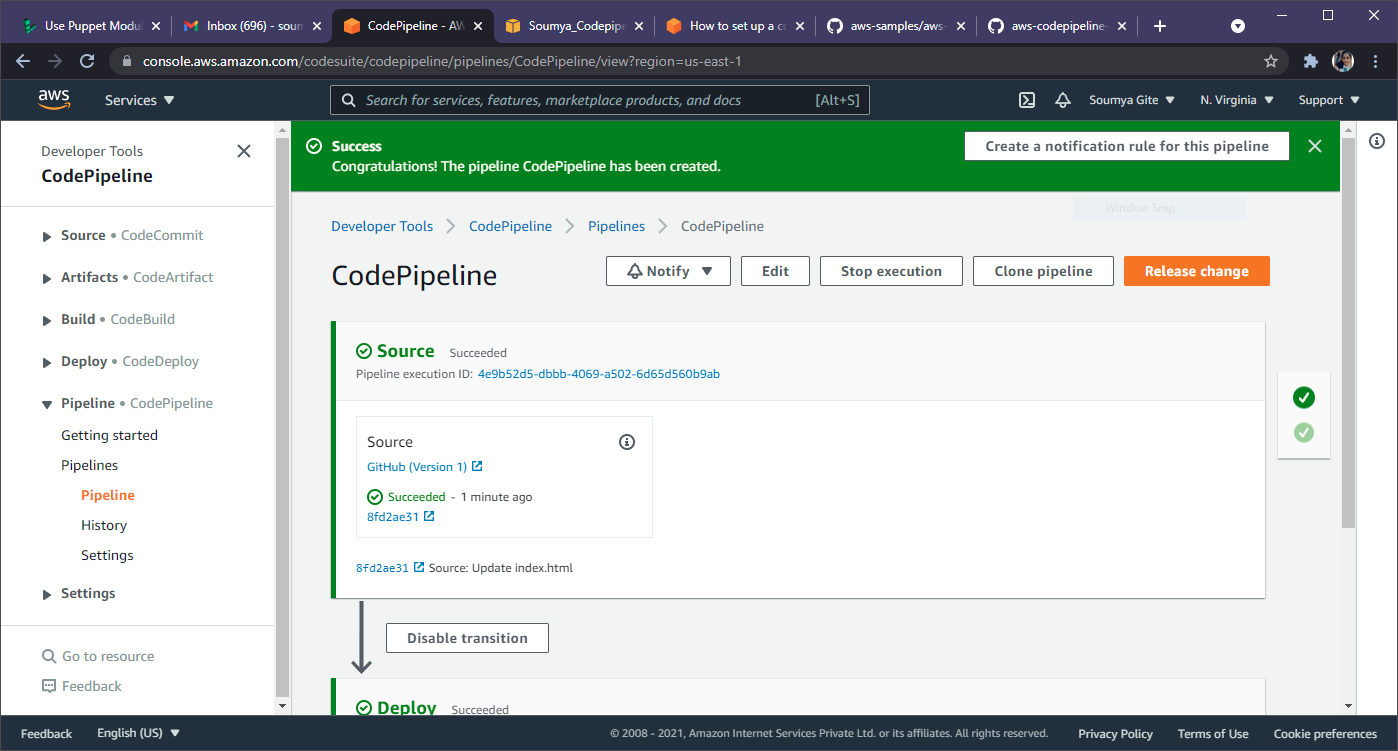


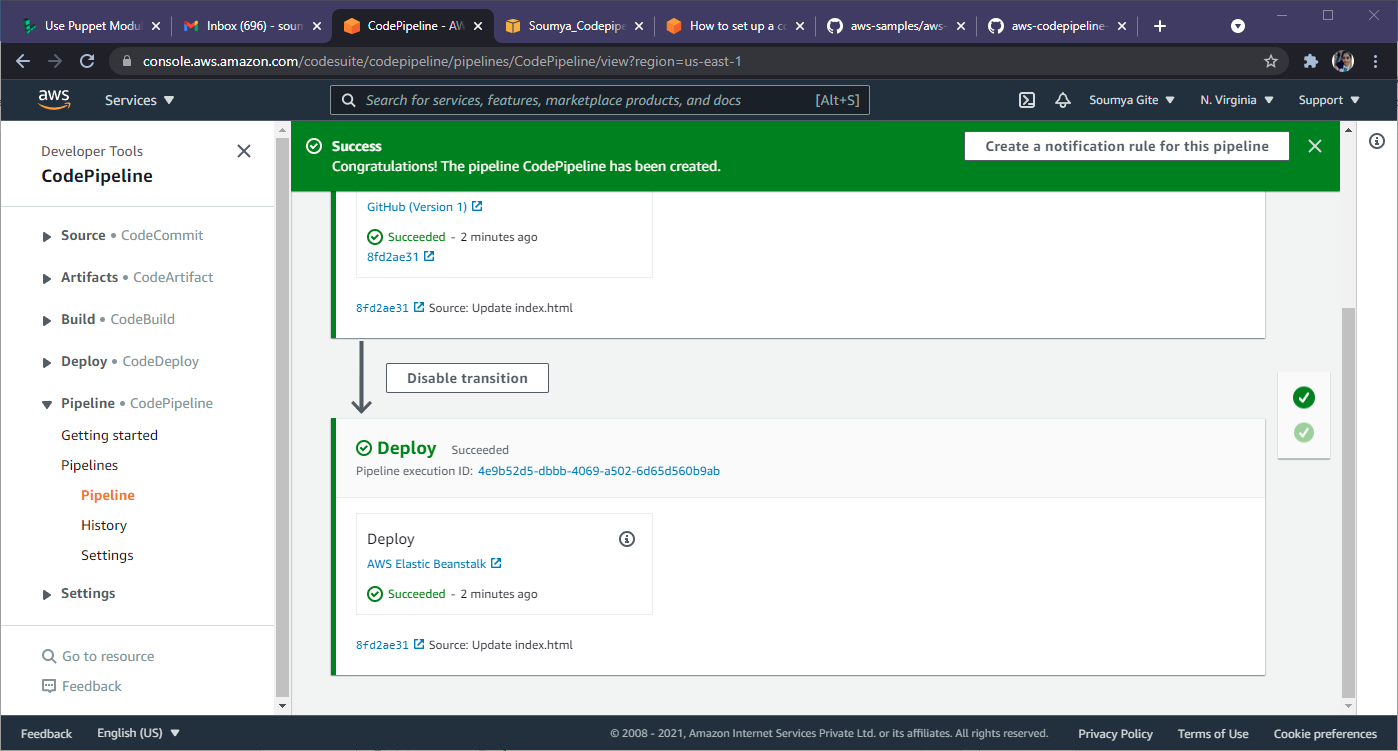




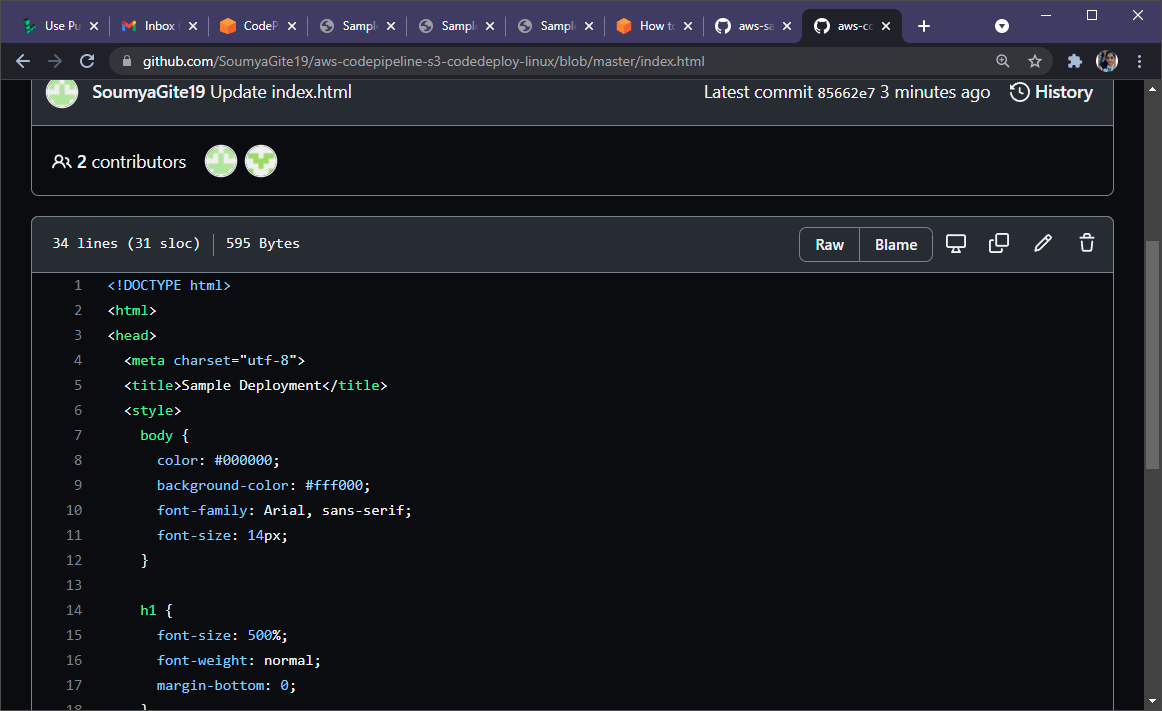


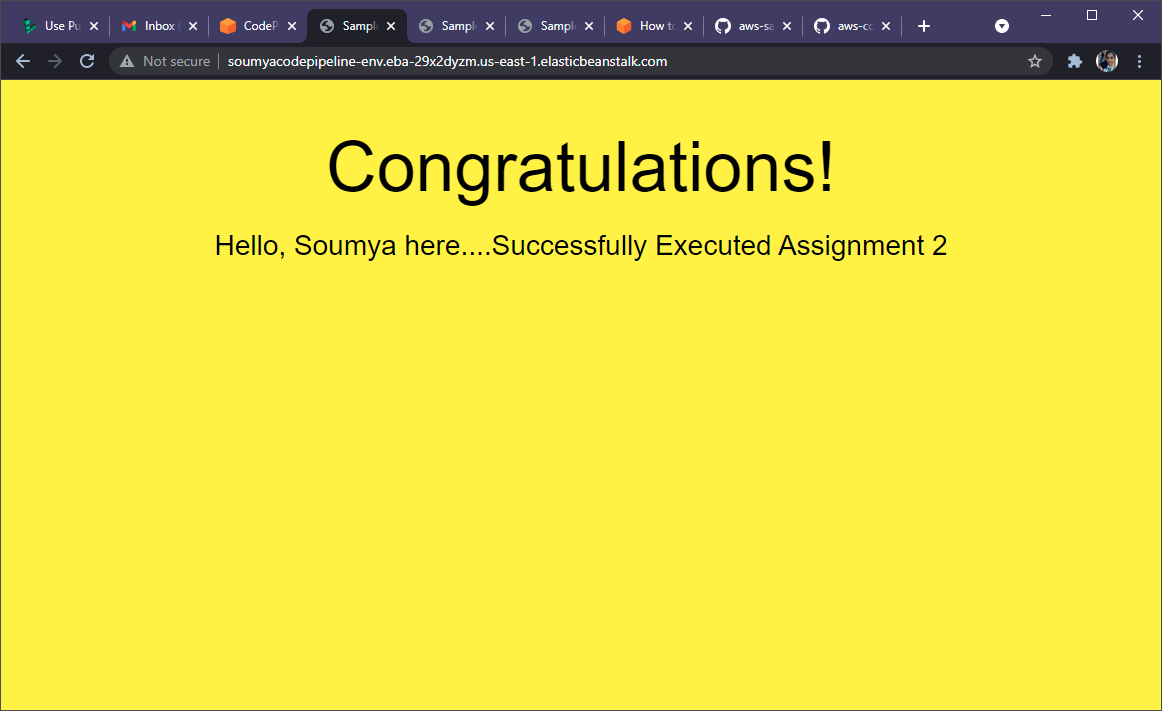






Index.html





Delete pipeline

