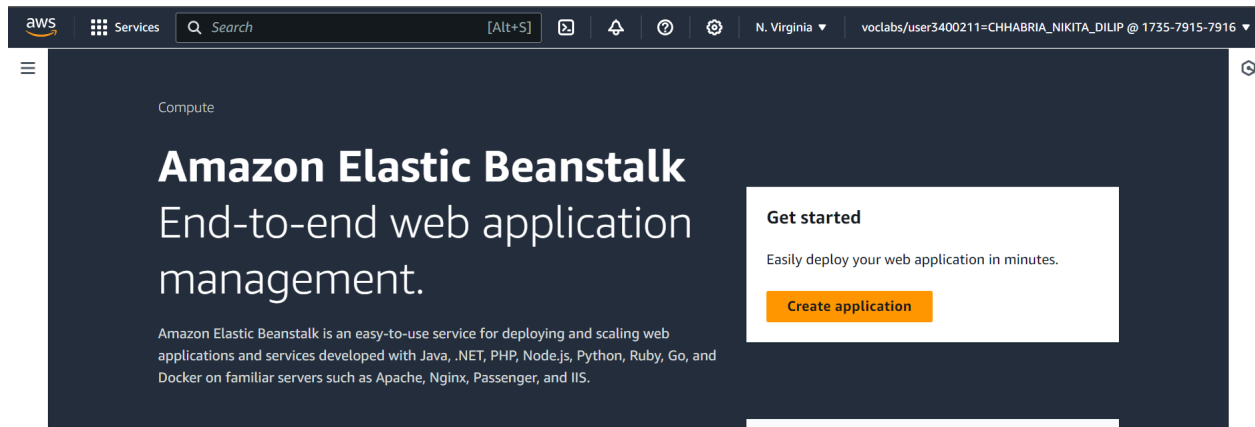
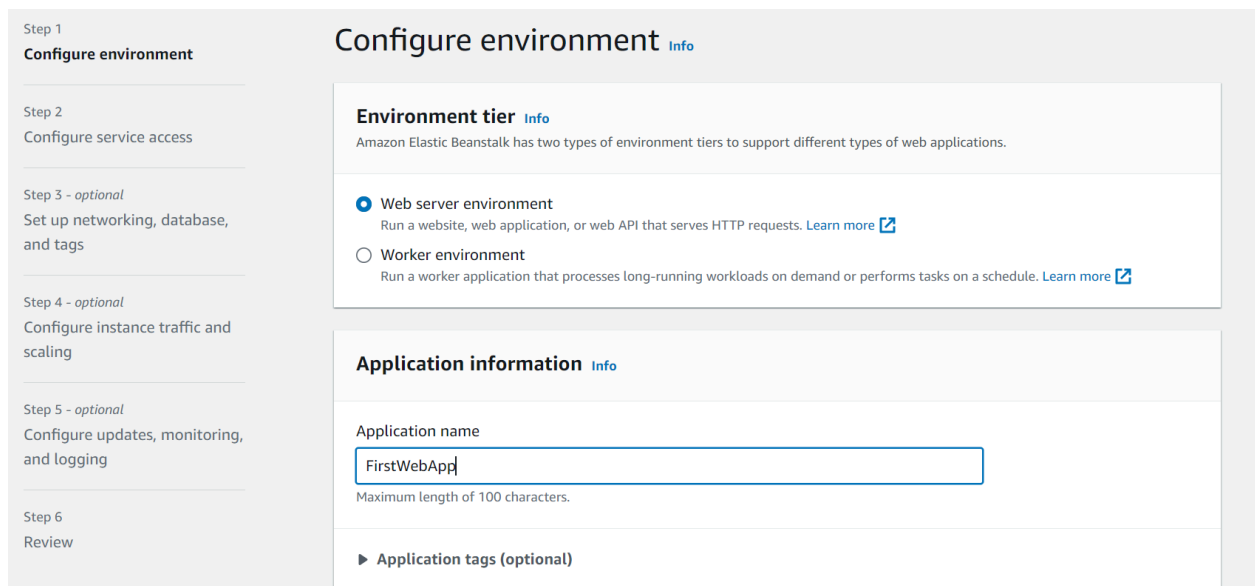


Aim: To Build Your Application using AWS CodeBuild and Deploy on S3 / SEBS using AWS CodePipeline, deploy Sample Application on EC2 instance using AWS CodeDeploy.

Go to your amazon academy account and search for <Elastic Beanstalk>



Click on create application and configure the environment



Now, manage the platform type as shown below

Platform [Info](#)

Platform type

☒ **Managed platform**
Platforms published and maintained by Amazon Elastic Beanstalk. [Learn more](#)

☐ **Custom platform**
Platforms created and owned by you. This option is unavailable if you have no platforms.

Platform

PHP

Platform branch

PHP 8.3 running on 64bit Amazon Linux 2023

Platform version

4.3.1 (Recommended)

A few more customizations

Service access

IAM roles, assumed by Elastic Beanstalk as a service role, and EC2 instance profiles allow Elastic Beanstalk to create and manage your environment. Both the IAM role and instance profile must be attached to IAM managed policies that contain the required permissions. [Learn more](#)

Service role

☐ Create and use new service role

☒ Use an existing service role

Existing service roles

Choose an existing IAM role for Elastic Beanstalk to assume as a service role. The existing IAM role must have the required IAM managed policies.

EMR_EC2_DefaultRole

EC2 key pair

Select an EC2 key pair to securely log in to your EC2 instances. [Learn more](#)

vockey

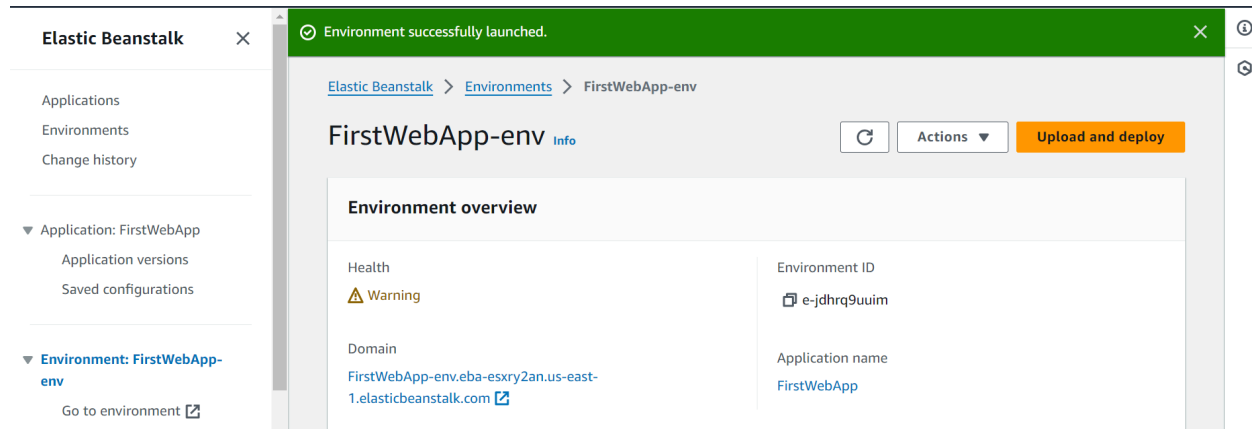
EC2 instance profile

Choose an IAM instance profile with managed policies that allow your EC2 instances to perform required operations.

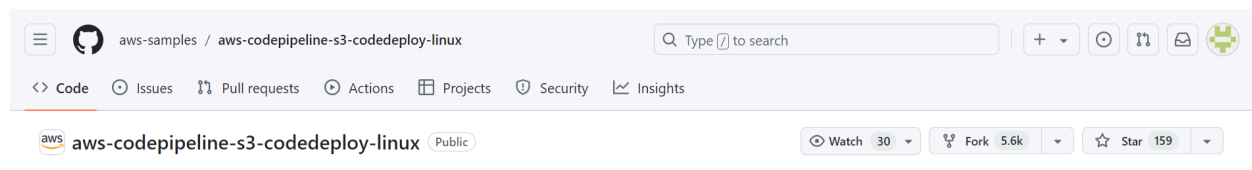
EMR_EC2_DefaultRole

[View permission details](#)

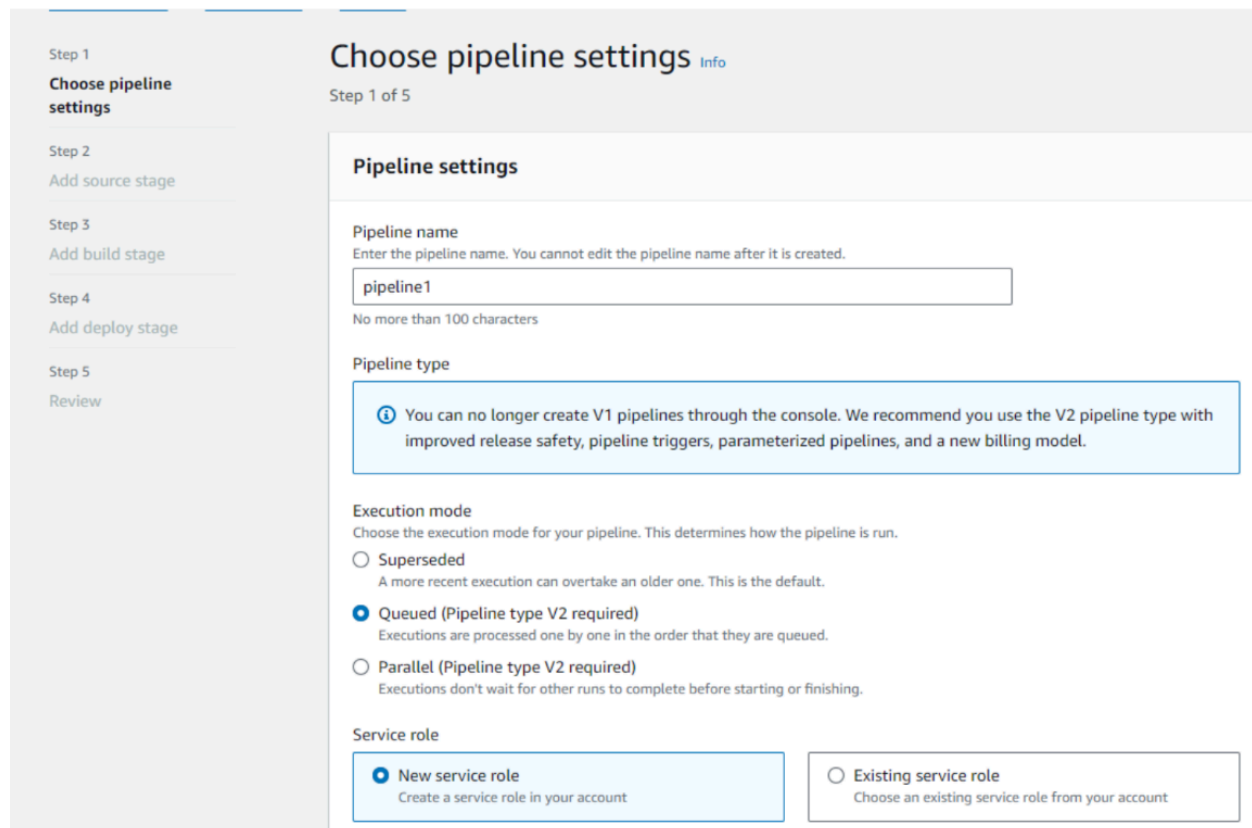
And environment created successfully!



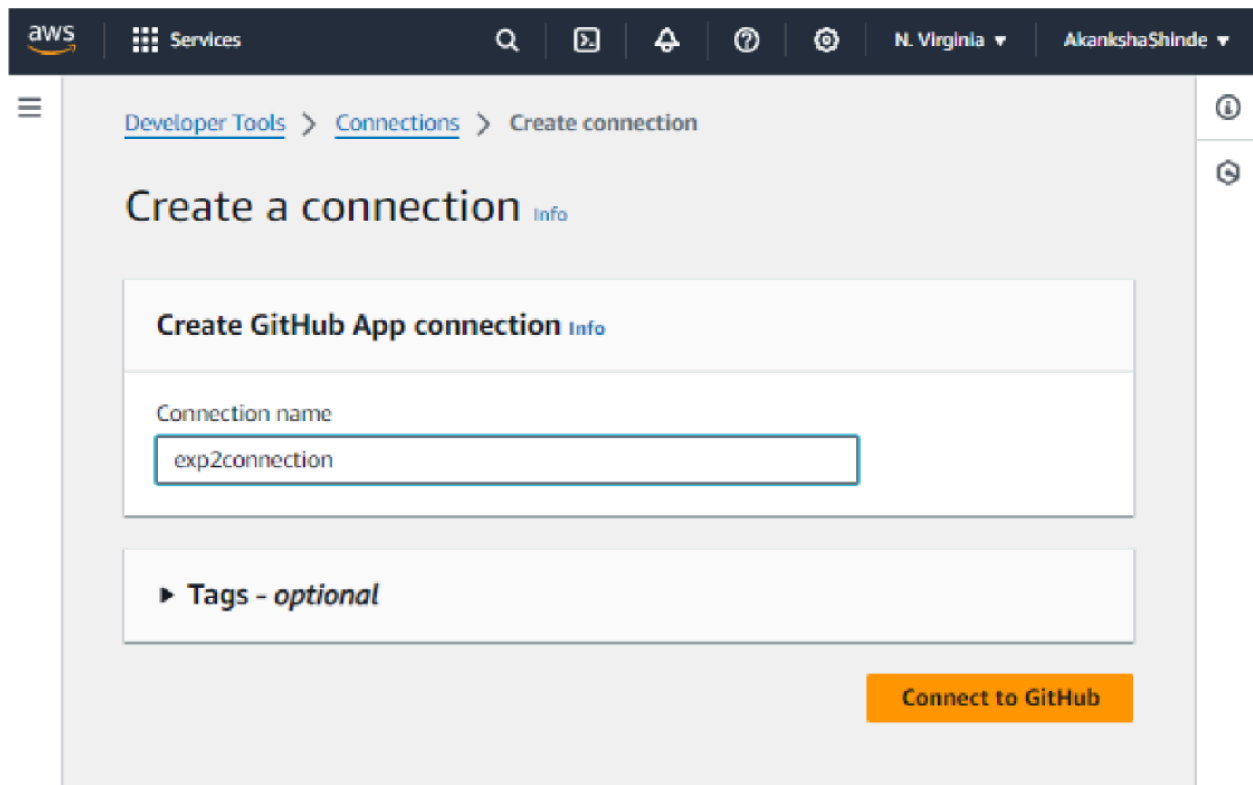
Now, go to github> <https://github.com/aws-samples/aws-codepipeline-s3-codedeploy-linux>



Fork this Repository and come back to the account and go to code pipeline



Follow through all the steps given




The screenshot shows the AWS IAM console interface. The top navigation bar includes the AWS logo, 'Services', a search bar, and icons for help, notifications, and settings. The user's profile 'AkankshaShinde' is visible in the top right. The breadcrumb trail indicates the path: 'Developer Tools > Connections > Create connection'. The main heading is 'Create a connection' with an 'Info' link. Below this, there is a section titled 'Create GitHub App connection' with an 'Info' link. A form field for 'Connection name' contains the text 'exp2connection'. Below the form, there is a section for 'Tags - optional' with a right-pointing arrow. At the bottom right, there is an orange button labeled 'Connect to GitHub'.

2. In the source stage, choose GitHub v2 as the provider, then connect your GitHub account to AWS by creating a connection. You'd need your GitHub credentials and then you'd need to authorize and install AWS on the forked GitHub Repository.

Source


Source provider
This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.

GitHub (Version 2) ▼

 **New GitHub version 2 (app-based) action**
To add a GitHub version 2 action in CodePipeline, you create a connection, which uses GitHub Apps to access your repository. Use the options below to choose an existing connection or create a new one. [Learn more](#)

Connection
Choose an existing connection that you have already configured, or create a new one and then return to this task.

or

 **Ready to connect**
Your GitHub connection is ready for use.

Repository name
Choose a repository in your GitHub account.


Trigger

Trigger type
Choose the trigger type that starts your pipeline.

☒ **No filter**
Starts your pipeline on any push and clones the HEAD.

☐ **Specify filter**
Starts your pipeline on a specific filter and clones the exact commit. Pipeline type V2 is required.

☐ **Do not detect changes**
Don't automatically trigger the pipeline.

 You can add additional sources and triggers by editing the pipeline after it is created.

3. Then, simply choose this forked repository and the branch which you will be able to find in the search box. After that, click Continue and skip the build stage. Proceed to the Deployment stage.

The screenshot shows the 'Add build stage' step, which is Step 3 of 5. The title 'Add build stage' is followed by an 'Info' link. Below the title, the text 'Step 3 of 5' is displayed. The main content area is titled 'Build - optional'. Under this, there is a section for 'Build provider' with a description: 'This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output file names.' Below the description is a dropdown menu. At the bottom of the form, there are four buttons: 'Cancel', 'Previous', 'Skip build stage', and 'Next' (which is highlighted in orange).

Review all the settings

The screenshot shows the 'Review' step, which is Step 5 of 5. The title 'Review' is followed by an 'Info' link. Below the title, the text 'Step 5 of 5' is displayed. The main content area is titled 'Step 1: Choose pipeline settings'. Below this, there is a section for 'Pipeline settings' which contains the following information: Pipeline name: firstPipeline, Pipeline type: V2, Execution mode: QUEUED, Artifact location: A new Amazon S3 bucket will be created as the default artifact store for your pipeline, Service role name: AWSCodePipelineServiceRole-us-east-1-firstPipeline. At the bottom of the form, there is a section for 'Variables'.

The pipeline is created successfully

Step 3: Add build stage

Build action provider

Build stage

No build

Step 4: Add deploy stage

Deploy action provider

Deploy action provider

AWS Elastic Beanstalk

ApplicationName

firstWebApp

EnvironmentName

FirstWebApp-env

Configure automatic rollback on stage failure

Disabled

Cancel

Previous

Create pipeline

Now, you will be able to see the code deployed

The screenshot shows the AWS CodePipeline console for a pipeline named 'pipeline1'. The pipeline type is V2 and the execution mode is QUEUED. The pipeline execution ID is 9c6d9b58-bd56-4c62-804e-90d7021f06fe. The pipeline consists of two stages: Source and Deploy. The Source stage is successful, using the 'Source' provider with the action 'Update README.md'. The Deploy stage is also successful, using the 'AWS Elastic Beanstalk' provider. A 'Disable transition' button is visible between the stages. The 'Release changes' button is highlighted in orange.

pipeline1 [Notify] [Edit] [Stop execution] [Clone pipeline] [Release changes]

Pipeline type: V2 Execution mode: QUEUED

Source Succeeded

Pipeline execution ID: 9c6d9b58-bd56-4c62-804e-90d7021f06fe

Source

[GitHub \(Version 2\)](#)

Succeeded - 1 minute ago

[8fd5da54](#)

[View details](#)

[8fd5da54](#) Source: Update README.md

[Disable transition](#)

Deploy Succeeded

Pipeline execution ID: 9c6d9b58-bd56-4c62-804e-90d7021f06fe

[Start rollback](#)

Deploy

[AWS Elastic Beanstalk](#)

Succeeded - just now

[View details](#)

Now we can see the URL provided for our application, we can simply click on it to lead us to our app

The screenshot shows the AWS Elastic Beanstalk console 'Environments' page. It displays a table with two environments: 'FirstApp-env' and 'MyfirstWebApp-env (terminating)'. The 'FirstApp-env' environment is healthy (Ok) and has a domain URL provided. The 'MyfirstWebApp-env' environment is in a terminating state (Unknown).

[Elastic Beanstalk](#) > Environments

Environments (2) [Info](#) [Refresh] [Actions] [Create environment]

	Environment name	Health	Application name	Platform	Domain	Running versions	Tier name	Date created
<input type="radio"/>	FirstApp-env	Ok	firstApp	PHP 8.1 running on...	FirstApp-env.eba-n34wxyhg.us...	code-pipeline-1723...	WebServer	August 14, 2024 22
<input type="radio"/>	MyfirstWebApp-env (terminating)	Unknown	myfirstWebApp	PHP 8.1 running on...	-	-	WebServer	August 14, 2024 22

Congratulations!

You have successfully created a pipeline that retrieved this source application from an Amazon S3 bucket and deployed it to three Amazon EC2 instances using AWS CodeDeploy.

For next steps, read the [AWS CodePipeline Documentation](#). Incedge 2020