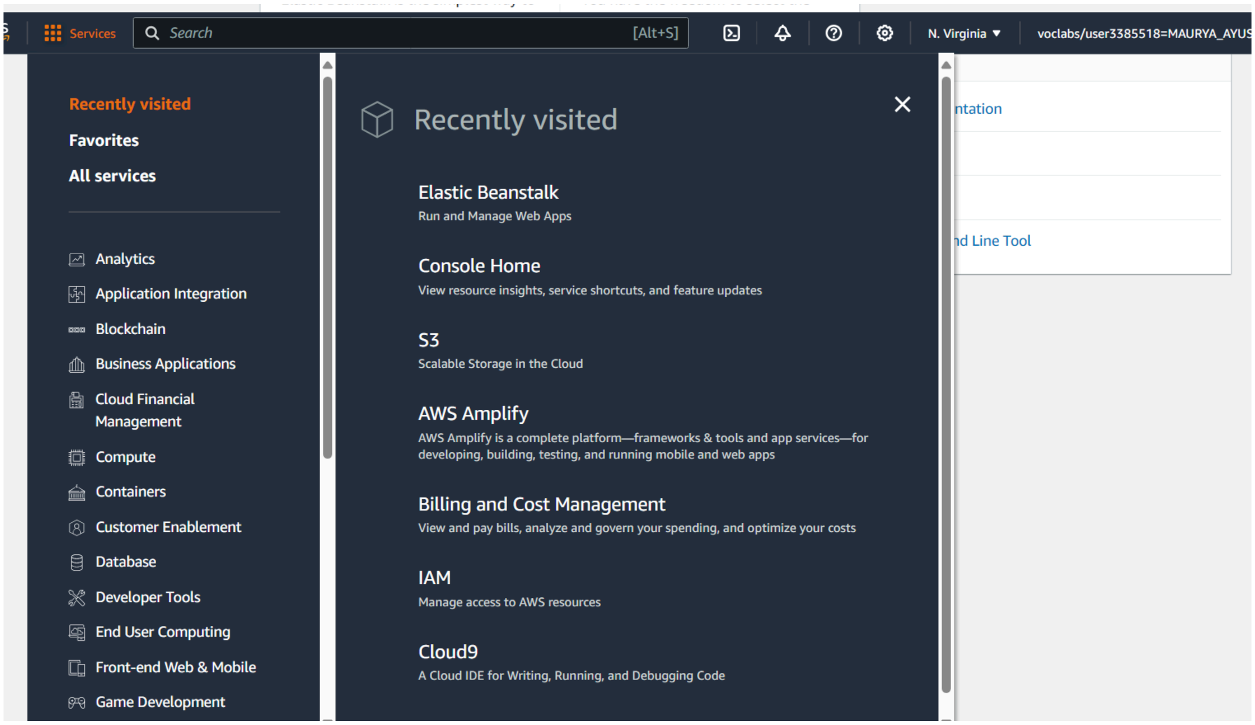
Step 1: Login to your AWS console. Search for Elastic Beanstalk in the searchbar near services.



Recently visited

Elastic Beanstalk
Run and Manage Web Apps

Console Home
View resource insights, service shortcuts, and feature updates

S3
Scalable Storage in the Cloud

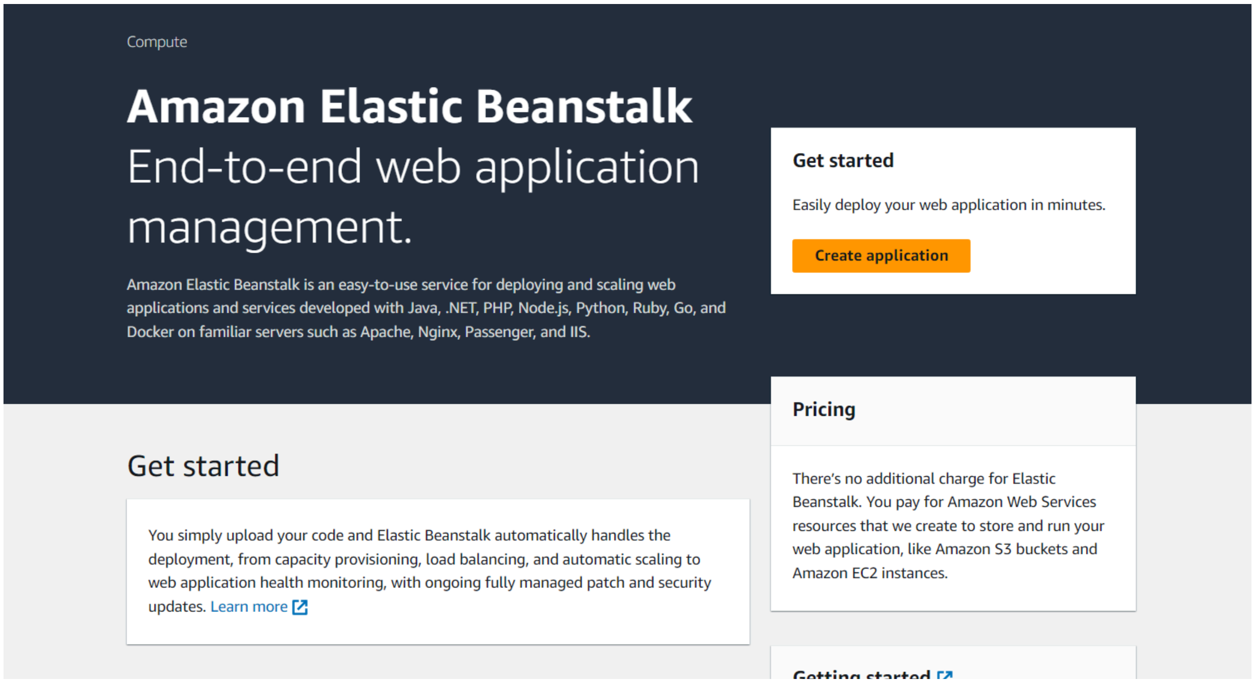
AWS Amplify
AWS Amplify is a complete platform—frameworks & tools and app services—for developing, building, testing, and running mobile and web apps

Billing and Cost Management
View and pay bills, analyze and govern your spending, and optimize your costs

IAM
Manage access to AWS resources

Cloud9
A Cloud IDE for Writing, Running, and Debugging Code

Step 2: Go to Elastic Beanstalk and click on Create Application



Amazon Elastic Beanstalk
End-to-end web application management.

Amazon Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.

Get started

Easily deploy your web application in minutes.

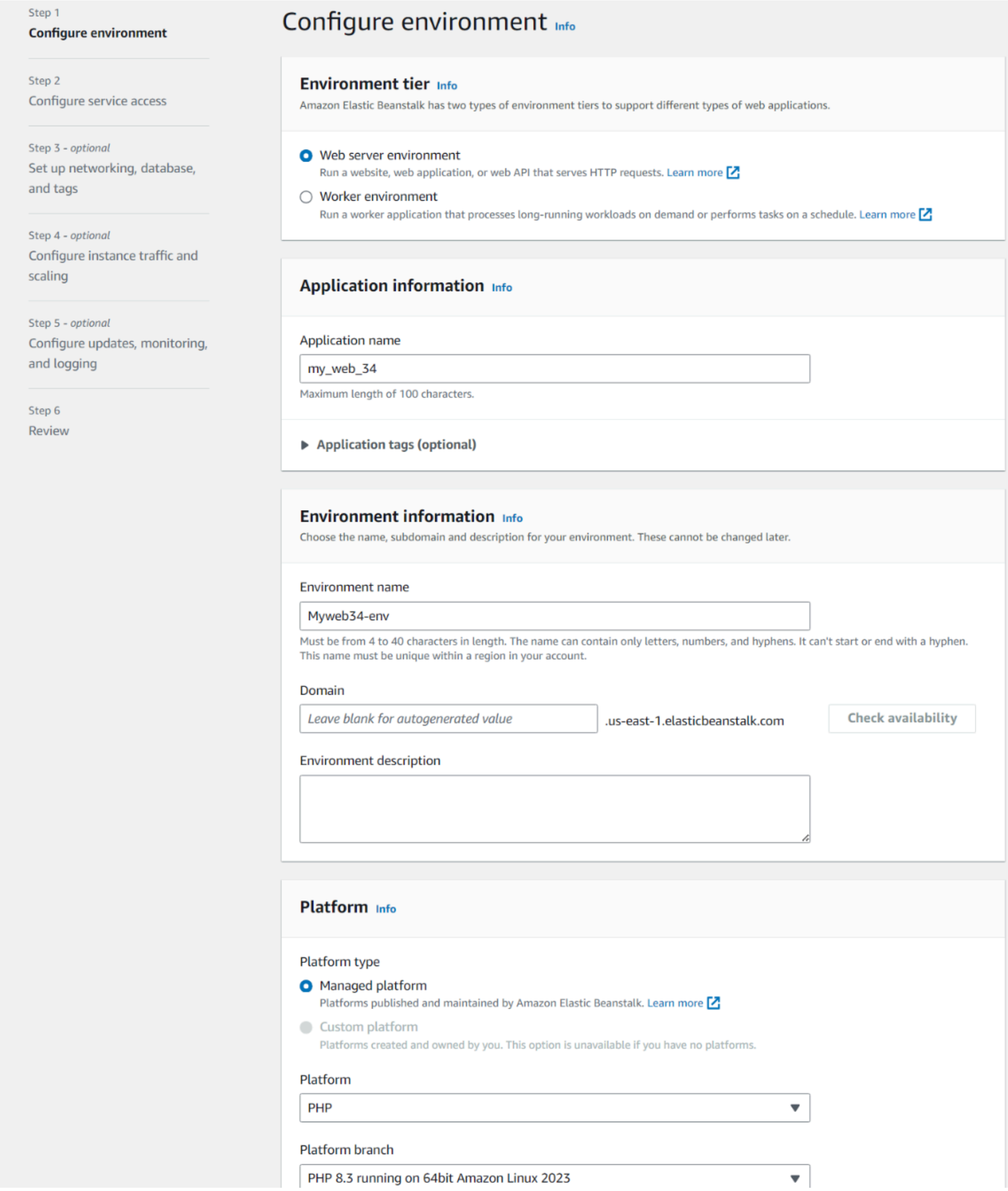
Pricing

There's no additional charge for Elastic Beanstalk. You pay for Amazon Web Services resources that we create to store and run your web application, like Amazon S3 buckets and Amazon EC2 instances.

Get started

You simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, and automatic scaling to web application health monitoring, with ongoing fully managed patch and security updates. Learn more

Step 3: Enter the name of your application. Scroll down and in the platform, select platform as PHP. Keep the application code as Sample Application. Set the instance to single instance. Click on NEXT



Configure environment Info

Environment tier Info

Amazon Elastic Beanstalk has two types of environment tiers to support different types of web applications.

● Web server environment
Run a website, web application, or web API that serves HTTP requests. Learn more

○ Worker environment
Run a worker application that processes long-running workloads on demand or performs tasks on a schedule. Learn more

Application information Info

Application name: my_web_34

Maximum length of 100 characters.

▶ Application tags (optional)

Environment information Info

Choose the name, subdomain and description for your environment. These cannot be changed later.

Environment name: Myweb34-env

Must be from 4 to 40 characters in length. The name can contain only letters, numbers, and hyphens. It can't start or end with a hyphen. This name must be unique within a region in your account.

Domain: *Leave blank for autogenerated value* .us-east-1.elasticbeanstalk.com Check availability

Environment description:

Platform Info

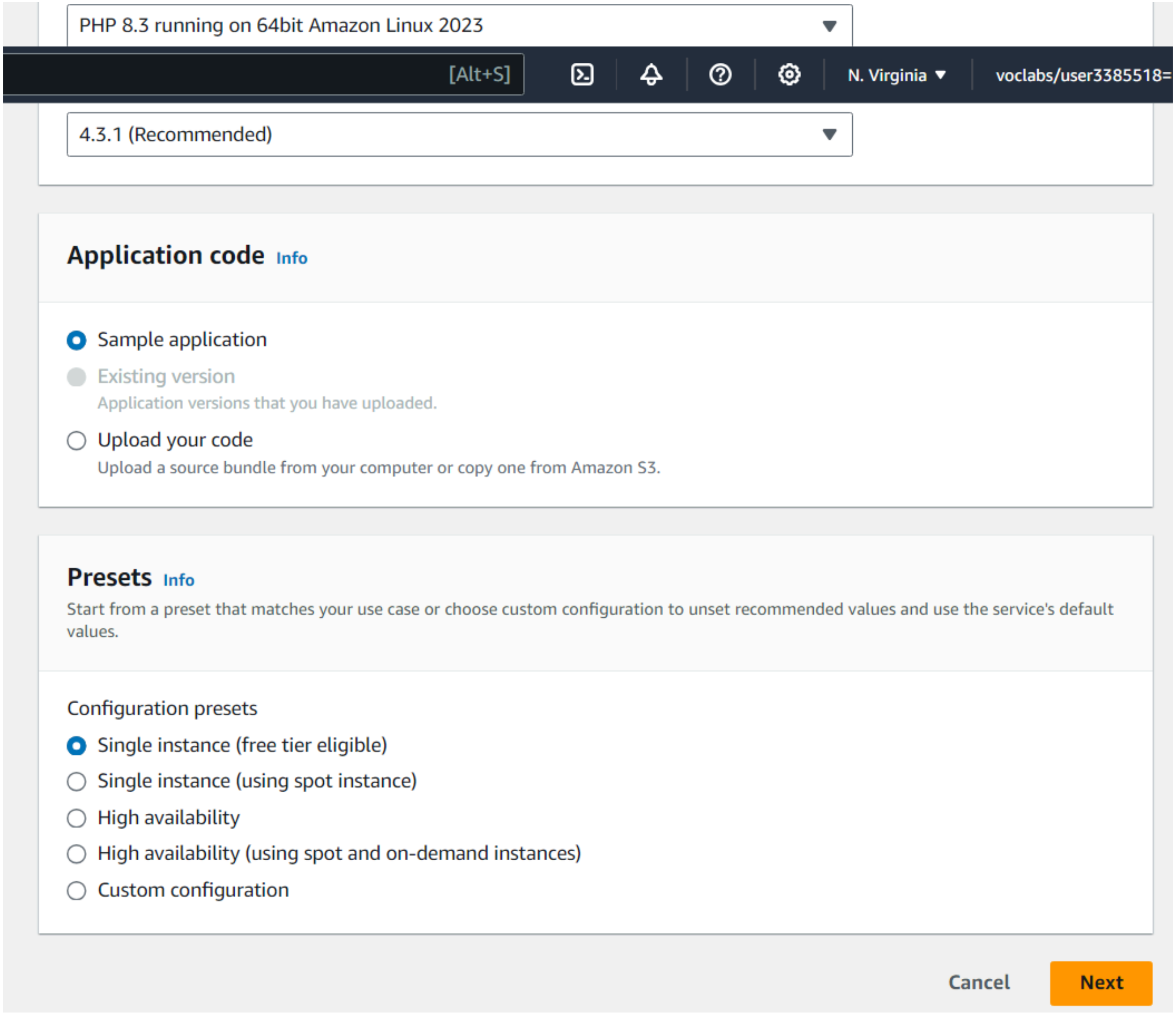
Platform type

● 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



PHP 8.3 running on 64bit Amazon Linux 2023

4.3.1 (Recommended)

Application code Info

● Sample application

○ Existing version
Application versions that you have uploaded.

○ Upload your code
Upload a source bundle from your computer or copy one from Amazon S3.

Presets Info

Start from a preset that matches your use case or choose custom configuration to unset recommended values and use the service's default values.

Configuration presets

● Single instance (free tier eligible)

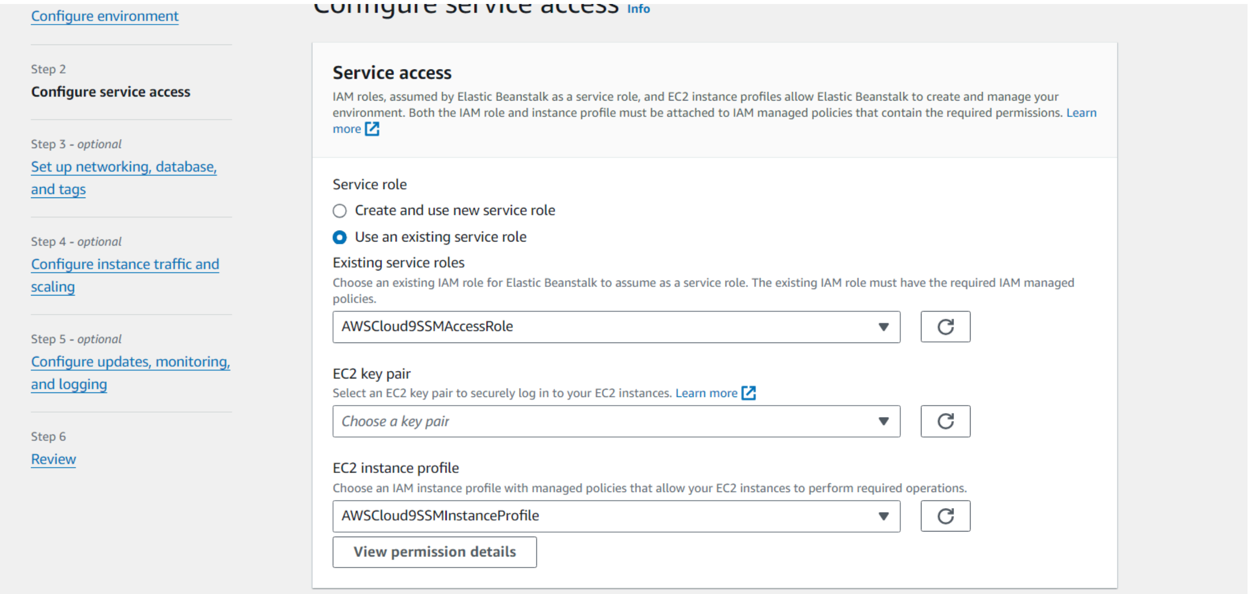
○ Single instance (using spot instance)

○ High availability

○ High availability (using spot and on-demand instances)

○ Custom configuration

Step 4 : Use an existing service role and choose whatever service role is available on your account.


Configure environment
Step 2
Configure service access
Step 3 - optional
Set up networking, database, and tags
Step 4 - optional
Configure instance traffic and scaling
Step 5 - optional
Configure updates, monitoring, and logging
Step 6
Review
Configure service access Info

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.

AWSCloud9SSMAccessRole

EC2 key pair

Select an EC2 key pair to securely log in to your EC2 instances. Learn more

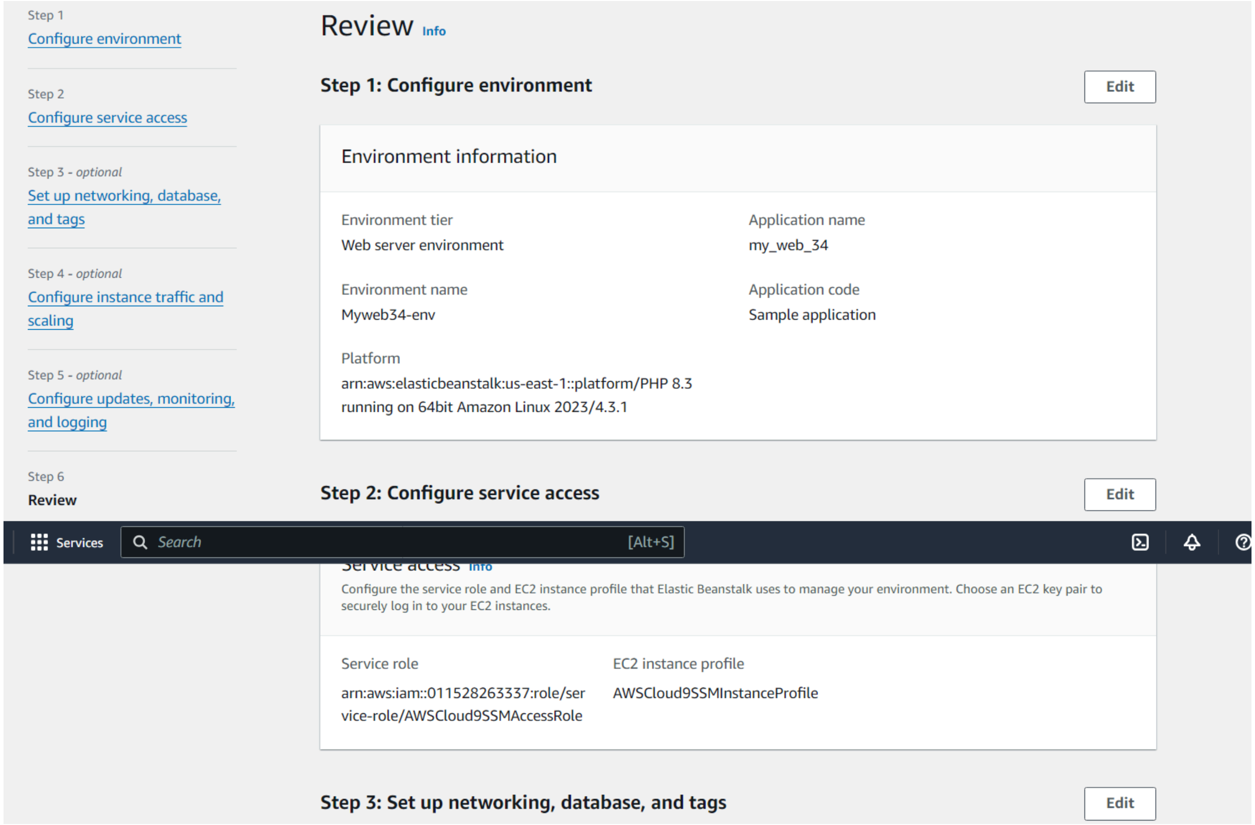
Choose a key pair

EC2 instance profile

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

AWSCloud9SSMInstanceProfile

Step 5 : Review the settings that you have set up for your application and submit your application.



Step 1: Configure environment

Environment information

Environment tier: Web server environment

Application name: my_web_34

Environment name: Myweb34-env

Application code: Sample application

Platform: arn:aws:elasticbeanstalk:us-east-1::platform/PHP 8.3 running on 64bit Amazon Linux 2023/4.3.1

Step 2: Configure service access

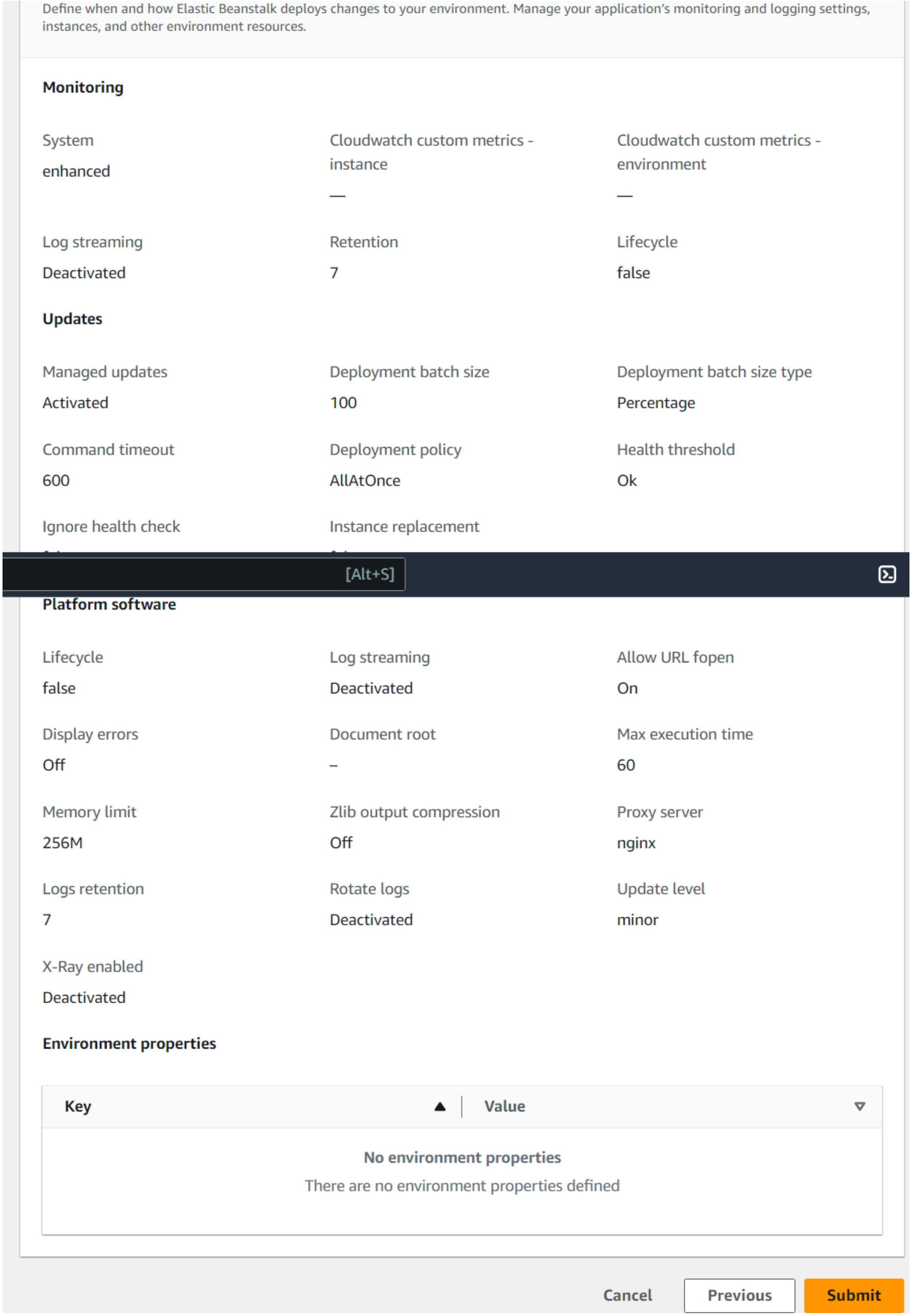
Service access Info

Configure the service role and EC2 instance profile that Elastic Beanstalk uses to manage your environment. Choose an EC2 key pair to securely log in to your EC2 instances.

Service role: arn:aws:iam::011528263337:role/service-role/AWSCloud9SSMAccessRole

EC2 instance profile: AWSCloud9SSMInstanceProfile

Step 3: Set up networking, database, and tags



Define when and how Elastic Beanstalk deploys changes to your environment. Manage your application's monitoring and logging settings, instances, and other environment resources.

Monitoring

System
enhanced

Cloudwatch custom metrics - instance
—

Cloudwatch custom metrics - environment
—

Log streaming
Deactivated

Retention
7

Lifecycle
false

Updates

Managed updates
Activated

Deployment batch size
100

Deployment batch size type
Percentage

Command timeout
600

Deployment policy
AllAtOnce

Health threshold
Ok

Ignore health check

Instance replacement

Platform software

Lifecycle
false

Log streaming
Deactivated

Allow URL fopen
On

Display errors
Off

Document root
–

Max execution time
60

Memory limit
256M

Zlib output compression
Off

Proxy server
nginx

Logs retention
7

Rotate logs
Deactivated

Update level
minor

X-Ray enabled
Deactivated

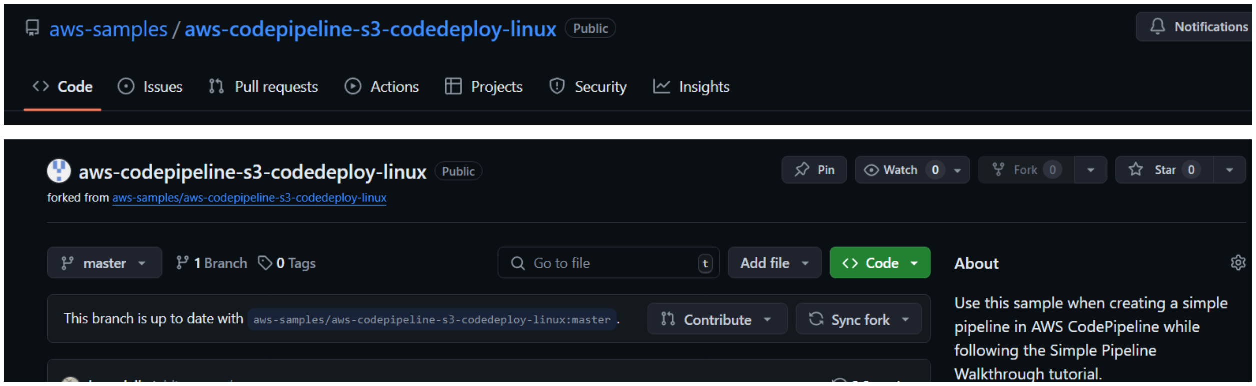
Environment properties

Key ▲	Value ▽

No environment properties

There are no environment properties defined

Step 6: Go to the github link below. This is a github with a sample code for deploying a file on AWS CodePipeline. Fork this repository into your personal github.
https://github.com/aws-samples/aws-codepipeline-s3-codedeploy-linux

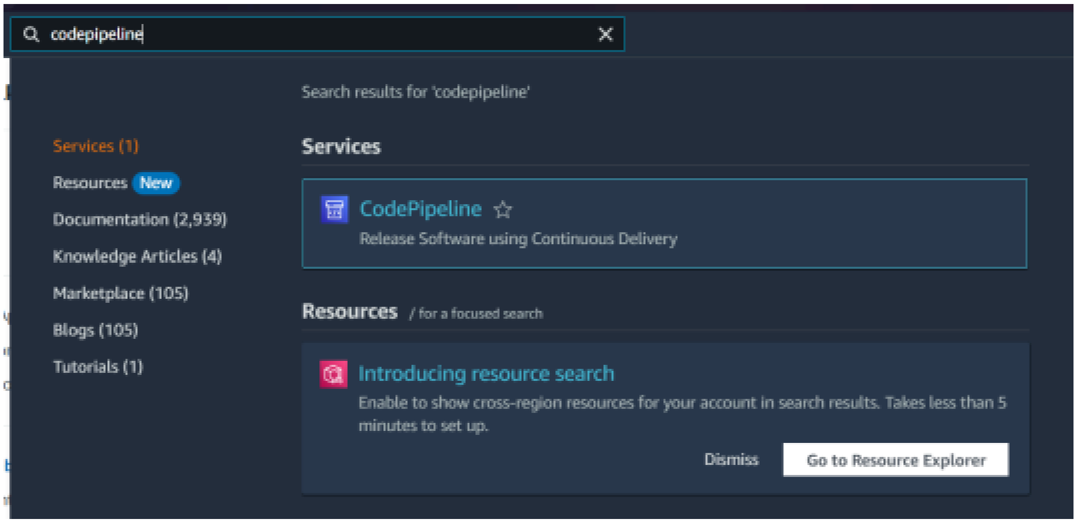


forked from aws-samples/aws-codepipeline-s3-codedeploy-linux

This branch is up to date with `aws-samples/aws-codepipeline-s3-codedeploy-linux:master`.

Use this sample when creating a simple pipeline in AWS CodePipeline while following the Simple Pipeline Walkthrough tutorial.

Step 7: Search CodePipeline in the services tab and click on it.

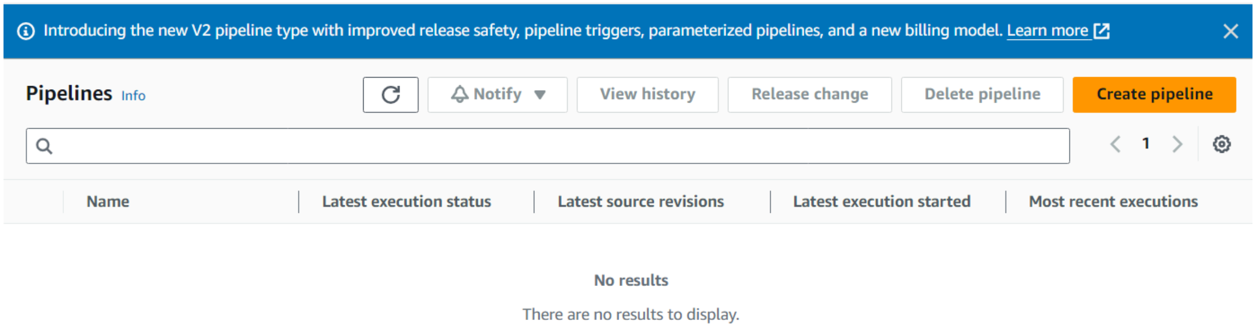


Search results for 'codepipeline'

Release Software using Continuous Delivery

Enable to show cross-region resources for your account in search results. Takes less than 5 minutes to set up.

Step 8: Click on Create Pipeline.



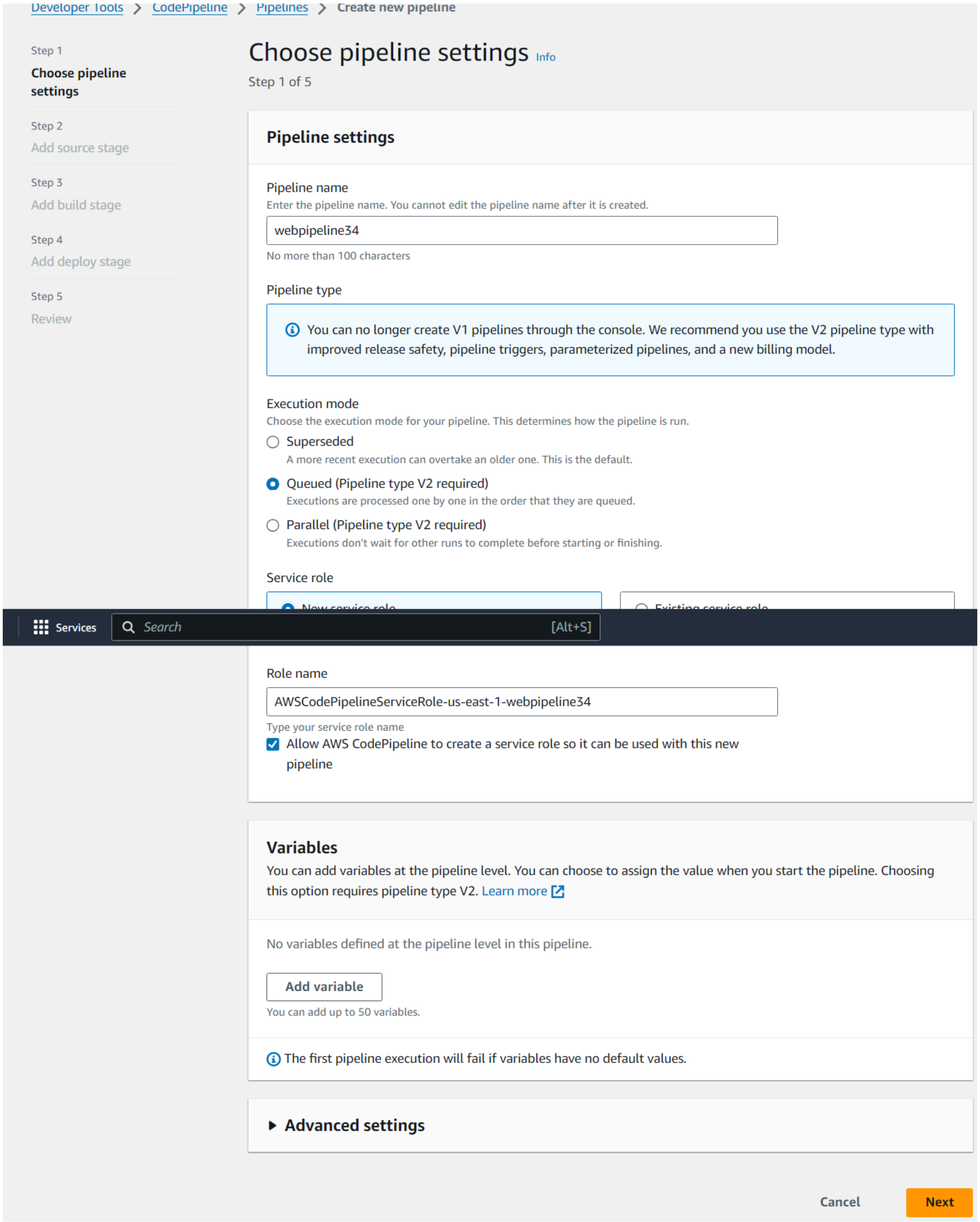
Introducing the new V2 pipeline type with improved release safety, pipeline triggers, parameterized pipelines, and a new billing model. Learn more

Name	Latest execution status	Latest source revisions	Latest execution started	Most recent executions

No results

There are no results to display.

Step 9: Give a name to your Pipeline. A new service role would be created with the name of the pipeline



Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

Choose pipeline settings Info

Step 1 of 5

Pipeline settings

Pipeline name
Enter the pipeline name. You cannot edit the pipeline name after it is created.
webpipeline34
No more than 100 characters

Pipeline type

ⓘ You can no longer create V1 pipelines through the console. We recommend you use the V2 pipeline type with improved release safety, pipeline triggers, parameterized pipelines, and a new billing model.

Execution mode
Choose the execution mode for your pipeline. This determines how the pipeline is run.

○ Superseded
A more recent execution can overtake an older one. This is the default.

● Queued (Pipeline type V2 required)
Executions are processed one by one in the order that they are queued.

○ Parallel (Pipeline type V2 required)
Executions don't wait for other runs to complete before starting or finishing.

Service role

● New service role ○ Existing service role

Role name
AWSCodePipelineServiceRole-us-east-1-webpipeline34
Type your service role name

☑ Allow AWS CodePipeline to create a service role so it can be used with this new pipeline

Variables

You can add variables at the pipeline level. You can choose to assign the value when you start the pipeline. Choosing this option requires pipeline type V2. Learn more

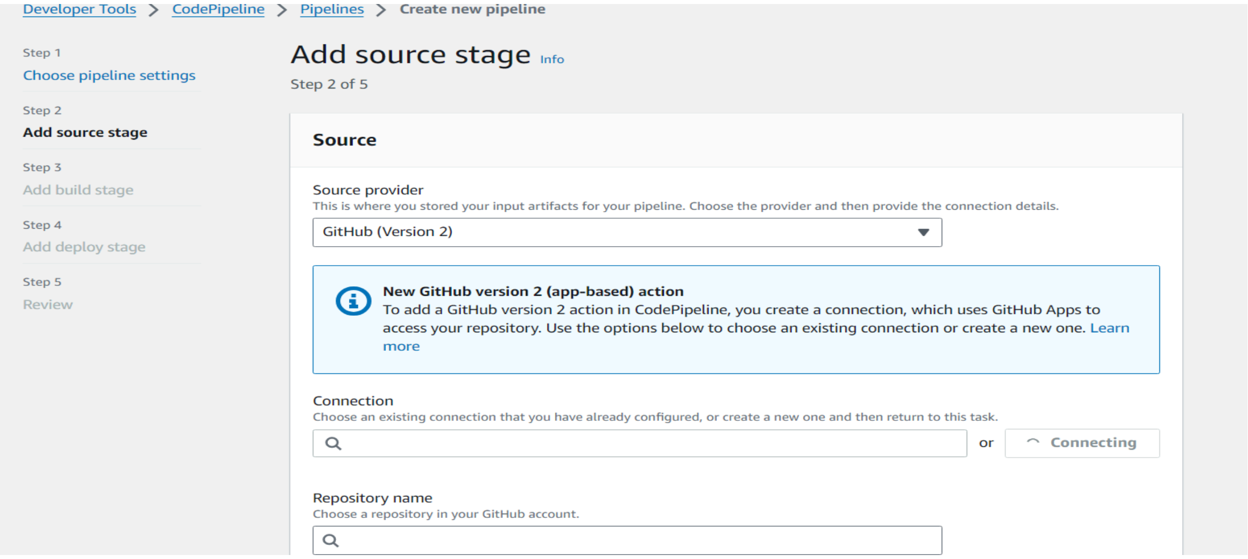
No variables defined at the pipeline level in this pipeline.

Add variable
You can add up to 50 variables.

ⓘ The first pipeline execution will fail if variables have no default values.

▸ **Advanced settings**

Step 10 : Select a source provider (as Github (Version 2)). Click on Connect to Github to connect your github.



Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

Add source stage Info

Step 2 of 5

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)

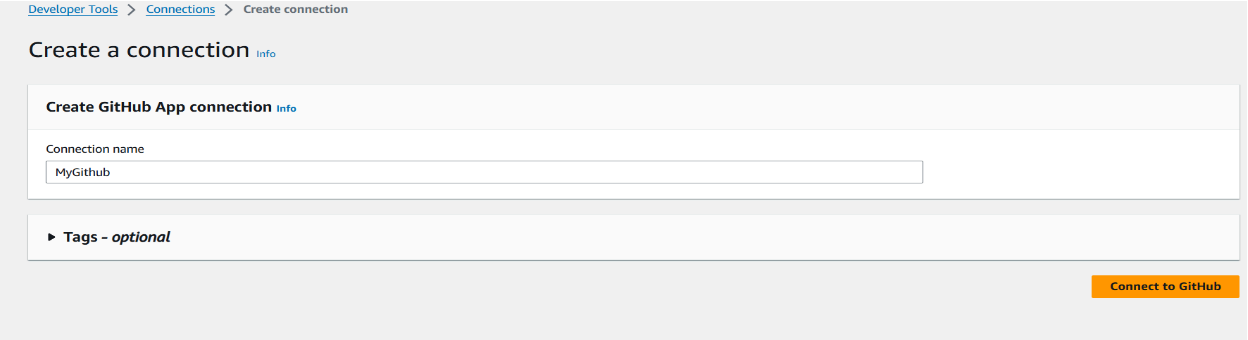
New GitHub version 2 (app-based) action
To add a GitHub version 2 action in CodePipeline, you create a connection, which uses GitHub Apps to access your repository. Use the options below to choose an existing connection or create a new one. Learn more

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

or Connecting

Repository name
Choose a repository in your GitHub account.

Step 11: Give a name to your GitHub app Connection and click on Connect. This will give you a prompt to either to select a GitHub app or to install a new app. If it is your first time, click on Install a new app



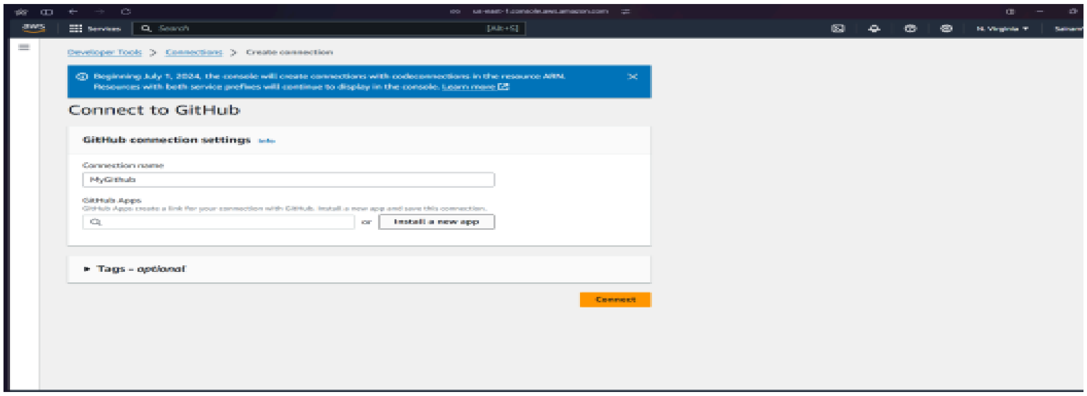
Developer Tools > Connections > Create connection

Create a connection Info

Create GitHub App connection Info

Connection name
MyGithub

Tags - *optional*



Developer Tools > Connections > Create connection

Beginning July 1, 2024, the console will create connections with codeconnections in the resource ARN. Resources with both service prefixes will continue to display in the console. Learn more

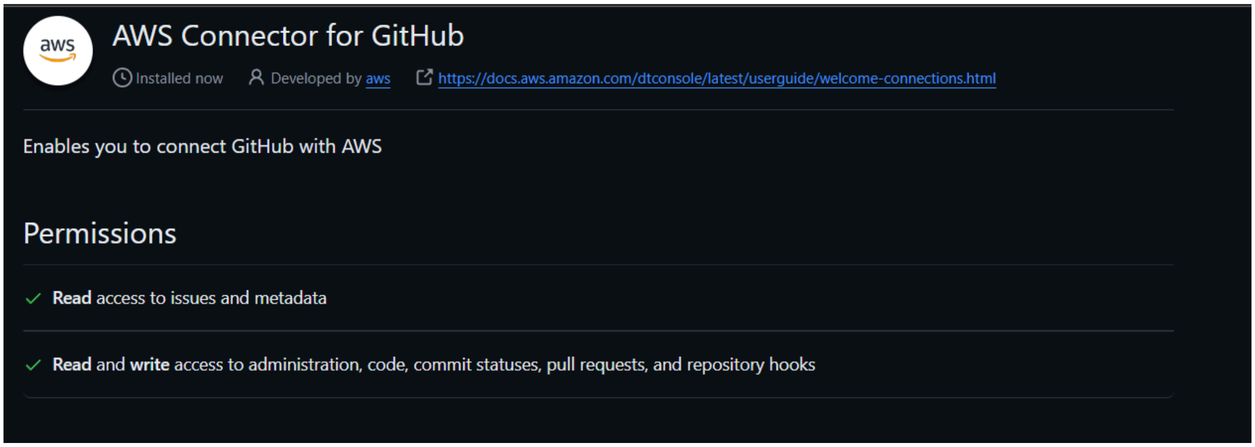
Connect to GitHub

GitHub connection settings Info

Connection name
MyGithub

GitHub Apps
GitHub Apps create a link for your connection with GitHub. Install a new app and save this connection.

Step 12 : This will direct you to install AWS Connector On Your GitHub. Install it to your account and give it its permissions.



AWS Connector for GitHub

Installed now · Developed by aws · https://docs.aws.amazon.com/dtconsole/latest/userguide/welcome-connections.html

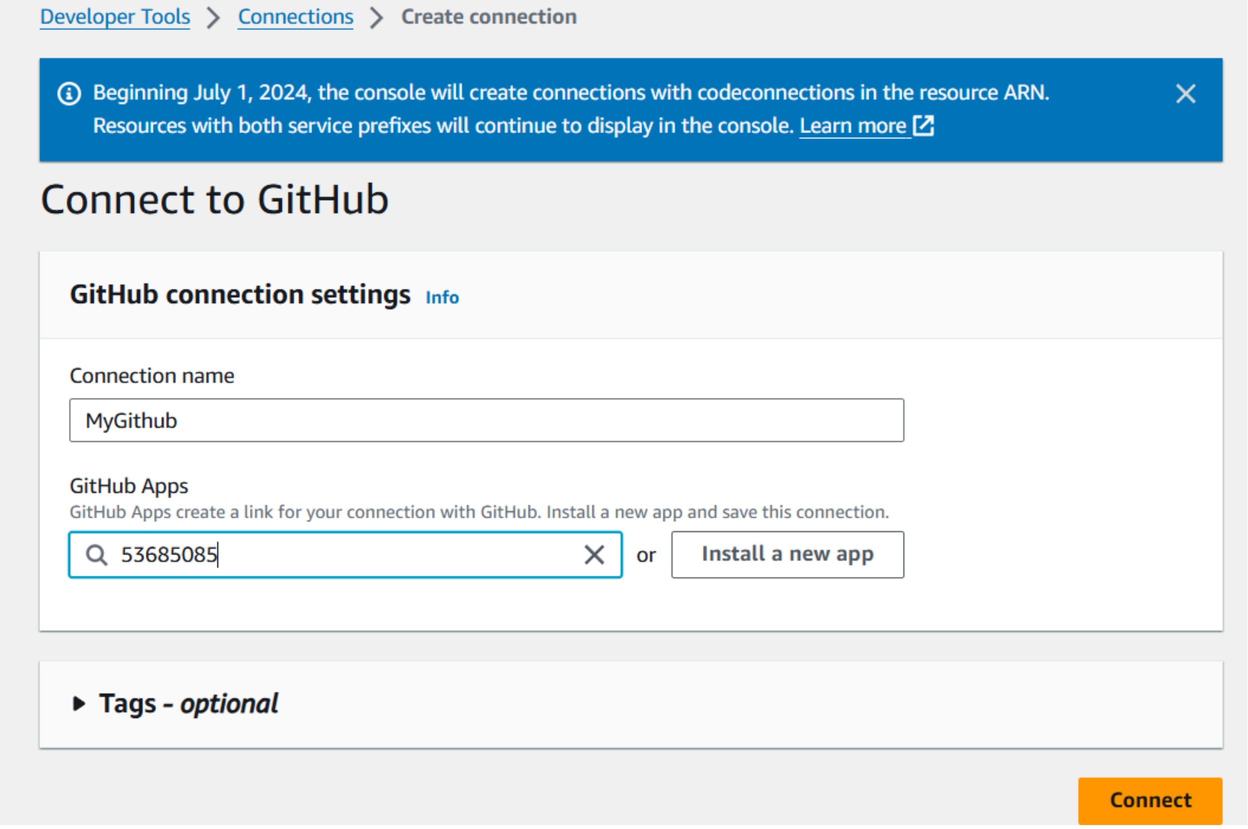
Enables you to connect GitHub with AWS

Permissions

✓ **Read** access to issues and metadata

✓ **Read** and **write** access to administration, code, commit statuses, pull requests, and repository hooks

Step 13: After the app is set up, it gives the number in the text field. Click on Connect. After clicking on connect, the link is shown in the connection field and AWS shows that GitHub connection is ready to use



Developer Tools > Connections > Create connection

Beginning July 1, 2024, the console will create connections with codeconnections in the resource ARN. Resources with both service prefixes will continue to display in the console. Learn more

Connect to GitHub

GitHub connection settings Info

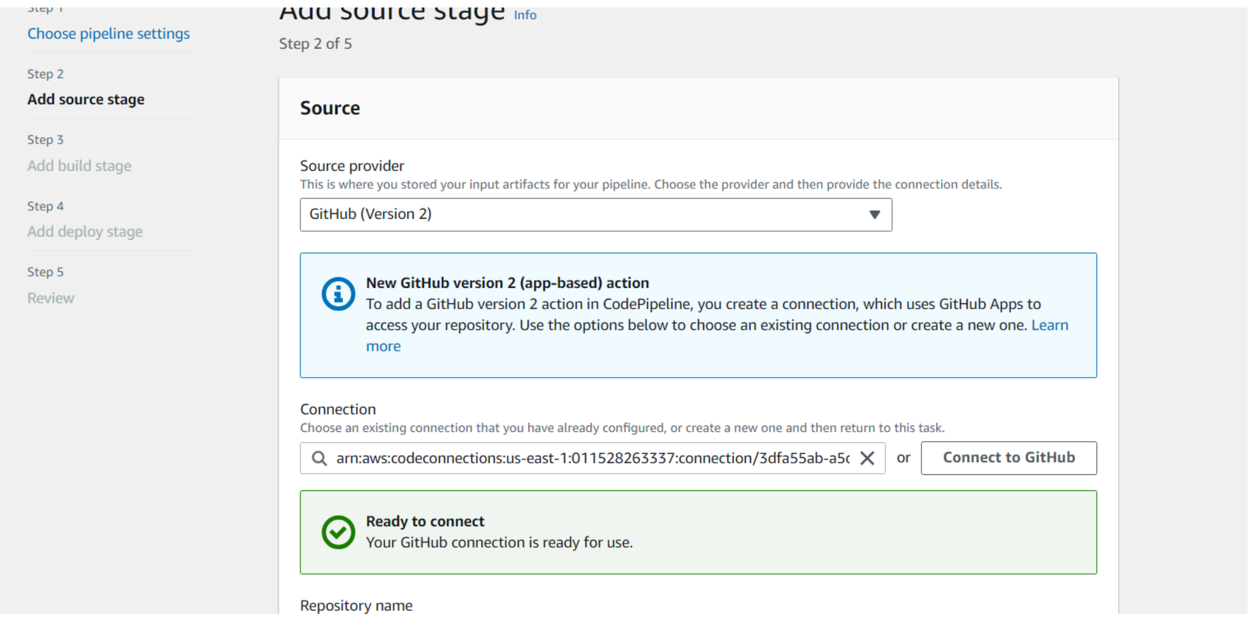
Connection name

MyGithub

GitHub Apps

GitHub Apps create a link for your connection with GitHub. Install a new app and save this connection.

53685085



Step 1

Choose pipeline settings

Step 2

Add source stage

Step 3

Add build stage

Step 4

Add deploy stage

Step 5

Review

Add source stage Info

Step 2 of 5

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.

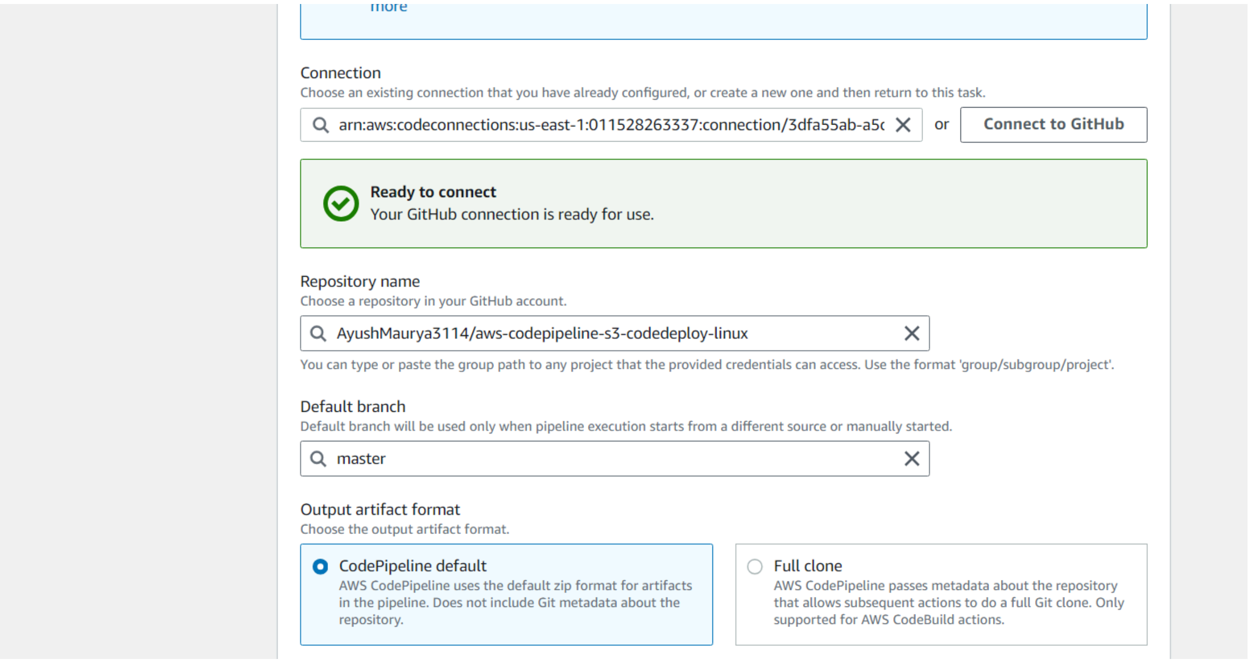
arn:aws:codeconnections:us-east-1:011528263337:connection/3dfa55ab-a5c

Ready to connect

Your GitHub connection is ready for use.

Repository name

Step 14: Select the repository that you had forked to your GitHub. After that select the branch on which the files are present (default is Master).



more

Connection

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

arn:aws:codeconnections:us-east-1:011528263337:connection/3dfa55ab-a5c

Ready to connect

Your GitHub connection is ready for use.

Repository name

Choose a repository in your GitHub account.

AyushMaurya3114/aws-codepipeline-s3-codedeploy-linux

You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

Default branch

Default branch will be used only when pipeline execution starts from a different source or manually started.

master

Output artifact format

Choose the output artifact format.

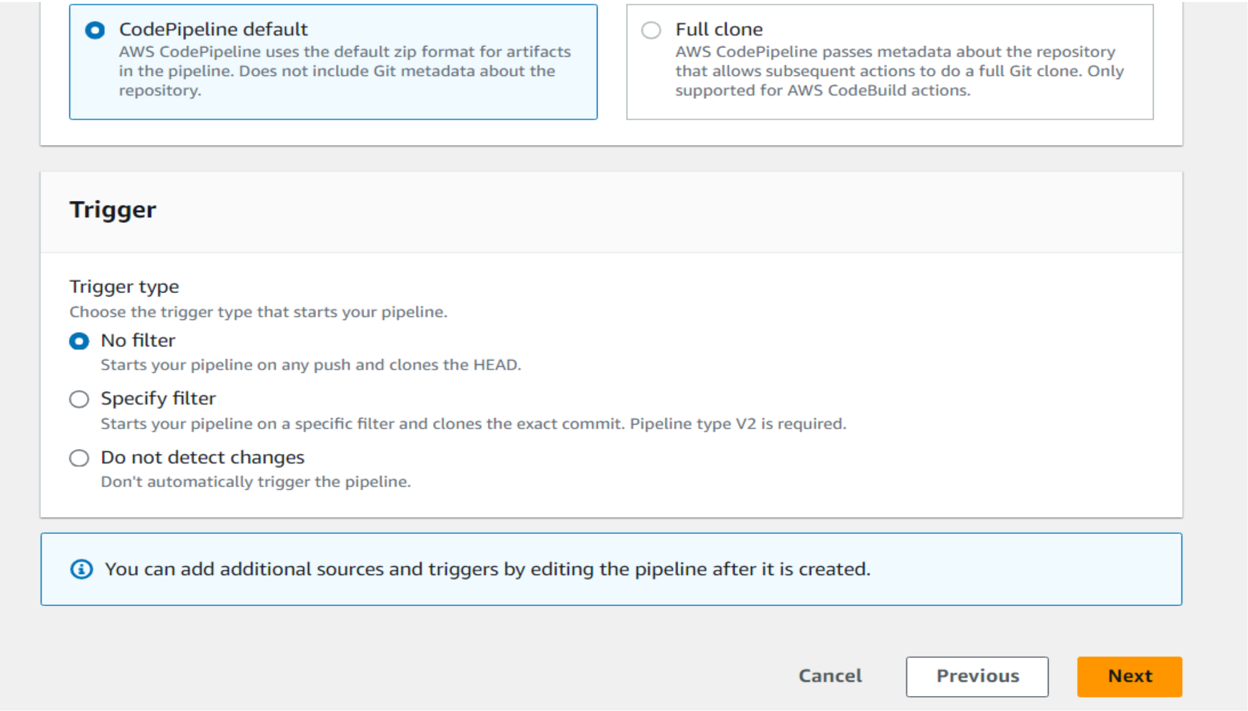
● CodePipeline default

AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include Git metadata about the repository.

○ Full clone

AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full Git clone. Only supported for AWS CodeBuild actions.

Step 15: Set the Trigger type as no filter. This would allow it to the website to update as soon as some change is made in the github.



● CodePipeline default
AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include Git metadata about the repository.

○ Full clone
AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full Git clone. Only supported for AWS CodeBuild actions.

Trigger

Trigger type
Choose the trigger type that starts your pipeline.

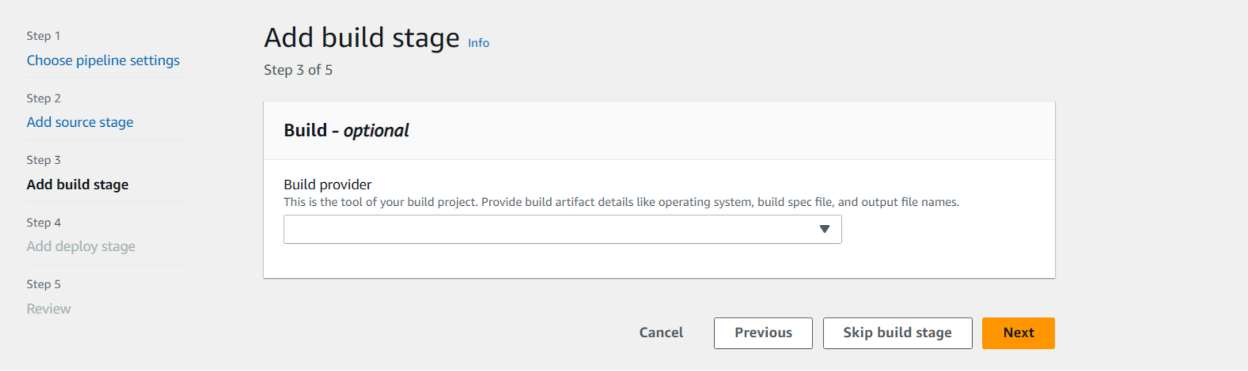
● No filter
Starts your pipeline on any push and clones the HEAD.

○ 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.

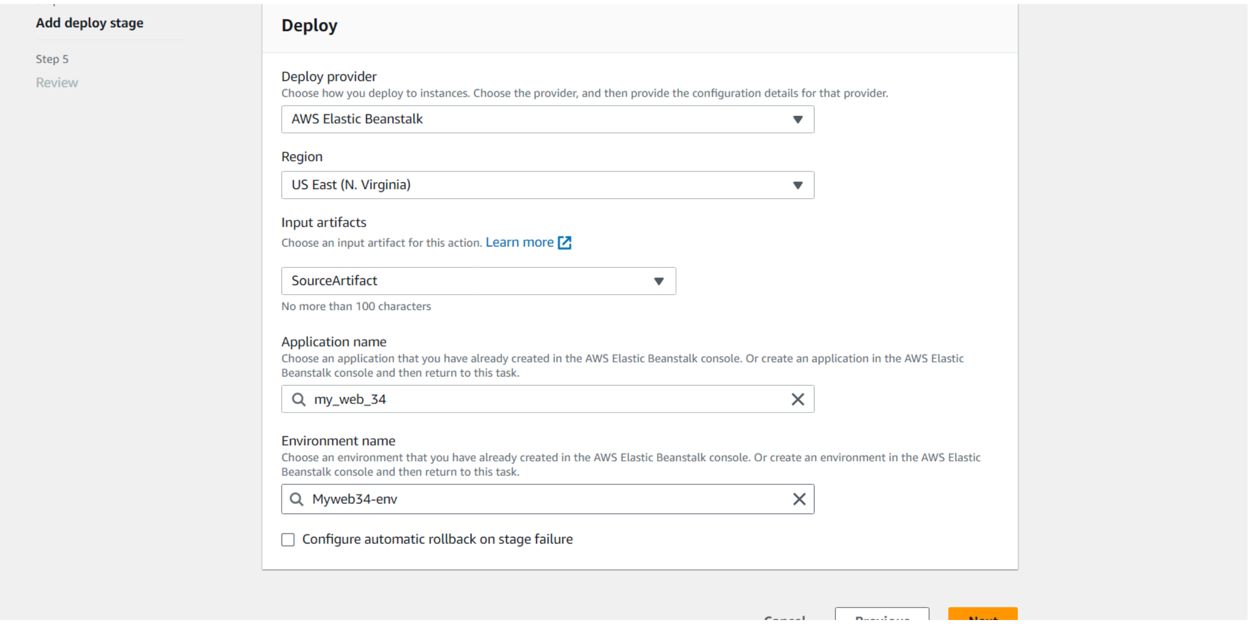
Step 16: Skip the build stage and go to Deploy. Select the deploy provider as AWS Elastic Beanstalk and Input Artifact as SourceArtifact. The application name would be the name of your Elastic Beanstalk. Then click on next.



Add build stage Info
Step 3 of 5

Build - *optional*

Build provider
This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output file names.



Deploy

Deploy provider
Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.
AWS Elastic Beanstalk

Region
US East (N. Virginia)

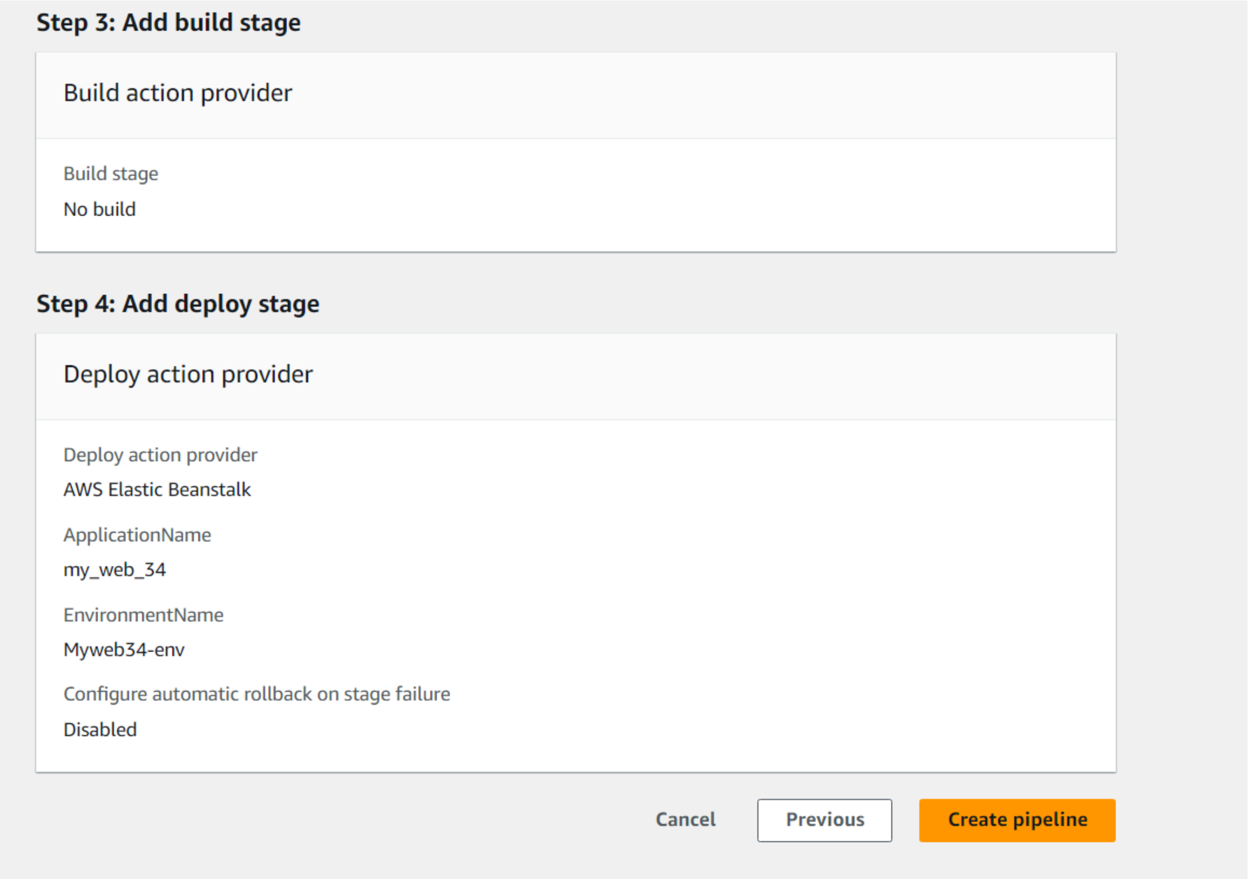
Input artifacts
Choose an input artifact for this action. Learn more
SourceArtifact
No more than 100 characters

Application name
Choose an application that you have already created in the AWS Elastic Beanstalk console. Or create an application in the AWS Elastic Beanstalk console and then return to this task.
my_web_34

Environment name
Choose an environment that you have already created in the AWS Elastic Beanstalk console. Or create an environment in the AWS Elastic Beanstalk console and then return to this task.
Myweb34-env

☐ Configure automatic rollback on stage failure

Step 17: Check all the information and click on create Pipeline.



Step 3: Add build stage

Build action provider

Build stage
No build

Step 4: Add deploy stage

Deploy action provider

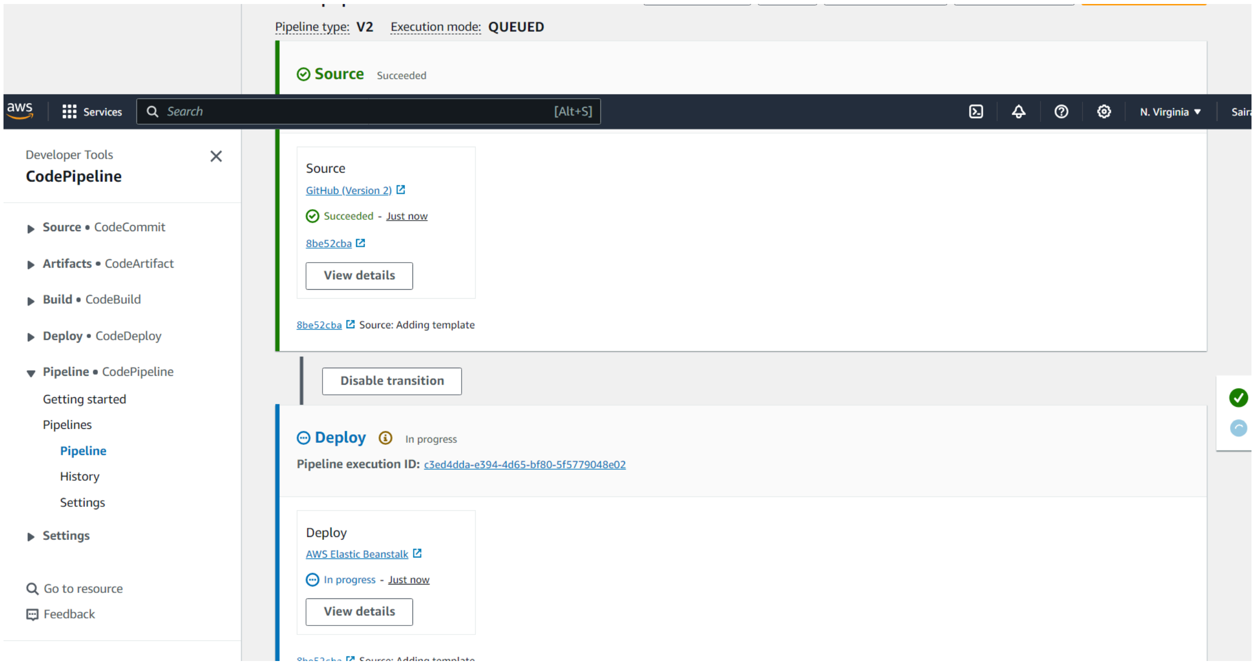
Deploy action provider
AWS Elastic Beanstalk

ApplicationName
my_web_34

EnvironmentName
Myweb34-env

Configure automatic rollback on stage failure
Disabled

Step 18: If the pipeline is successfully deployed, this screen comes up where the source is set up and then it is transitioned to deploy.



Pipeline type: V2 Execution mode: QUEUED

Source Succeeded

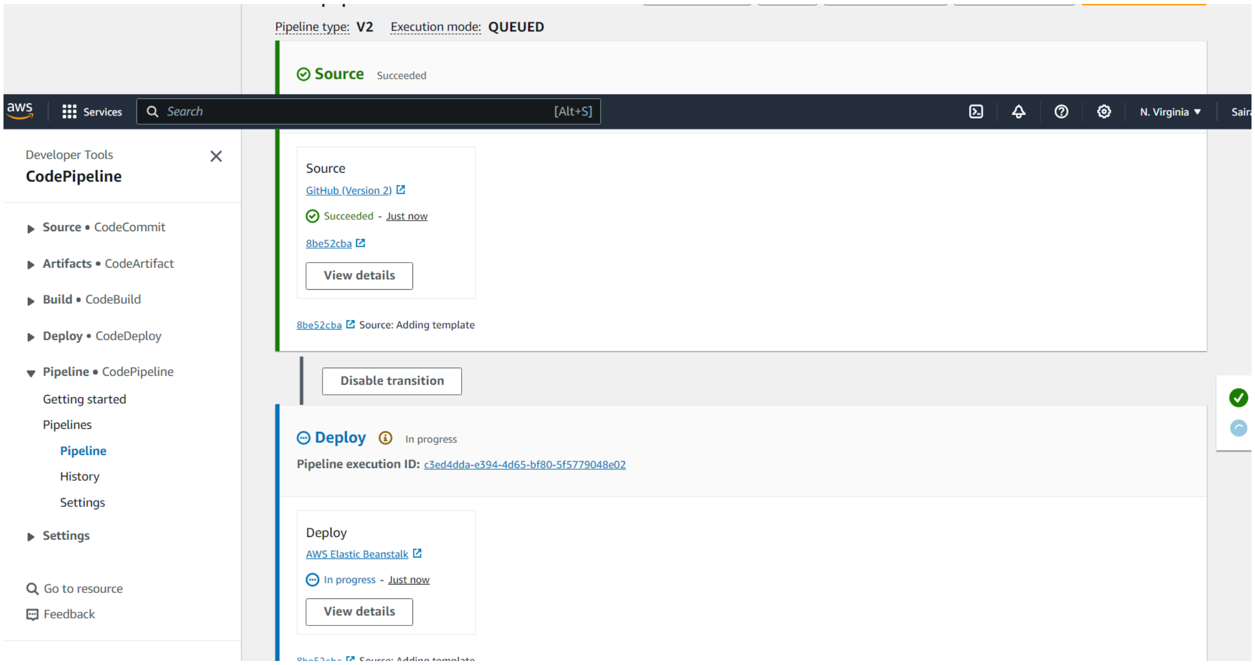
Source
GitHub (Version 2)
Succeeded - Just now
8be52cba

8be52cba Source: Adding template

Deploy In progress
Pipeline execution ID: c3ed4dda-e394-4d65-bf80-5f5779048e02

Deploy
AWS Elastic Beanstalk
In progress - Just now

Step 19: Once the deployment is complete, click on the AWS Elastic Beanstalk under Deploy.



Pipeline type: V2 Execution mode: QUEUED

Source Succeeded

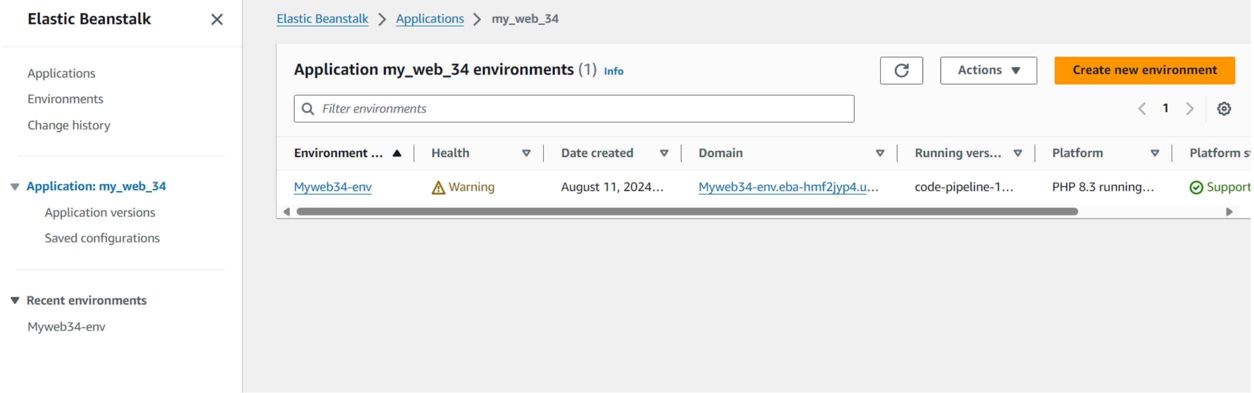
Source
GitHub (Version 2)
Succeeded - Just now
8be52cba

8be52cba Source: Adding template

Deploy In progress
Pipeline execution ID: c3ed4dda-e394-4d65-bf80-5f5779048e02

Deploy
AWS Elastic Beanstalk
In progress - Just now

Step 20: This will redirect you to the application screen of Elastic Beanstalk. Click on the link shown under Domain.



Elastic Beanstalk ✕

Applications
Environments
Change history

▼ Application: my_web_34
Application versions
Saved configurations

▼ Recent environments
Myweb34-env

Elastic Beanstalk > Applications > my_web_34

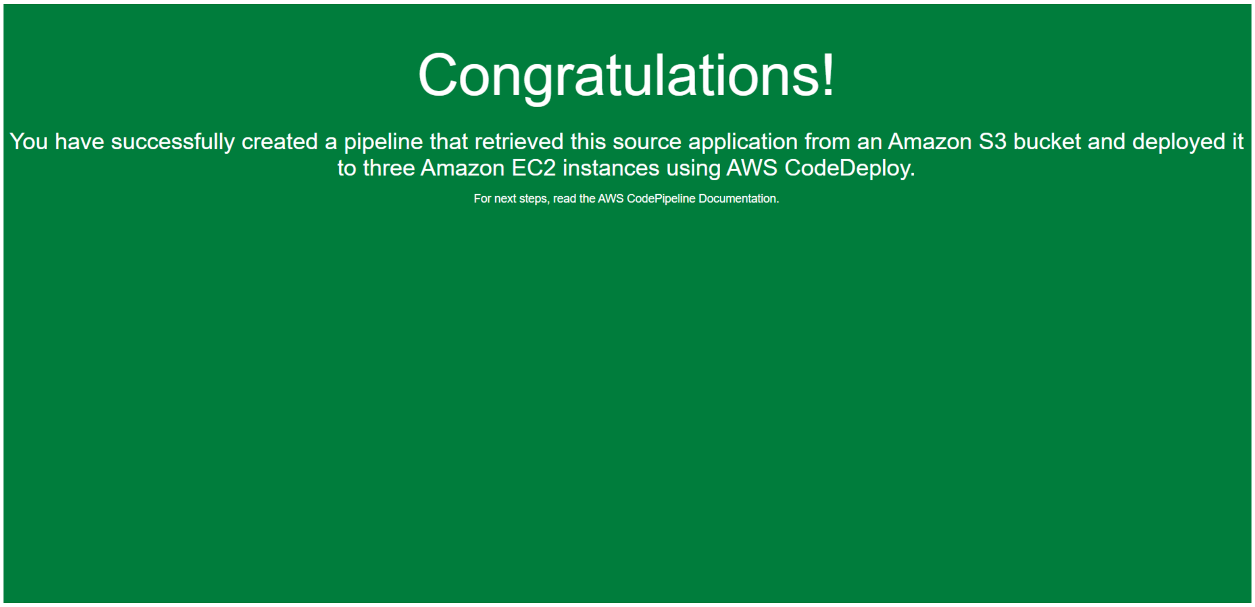
Application my_web_34 environments (1) Info

Actions ▼ | Create new environment

Filter environments

Environment ... ▲	Health	Date created	Domain	Running vers...	Platform	Platform s
Myweb34-env	⚠ Warning	August 11, 2024...	Myweb34-env.eba-hmf2jyp4.u...	code-pipeline-1...	PHP 8.3 running...	✓ Support

Step 21: This will successfully show the sample website hosted.

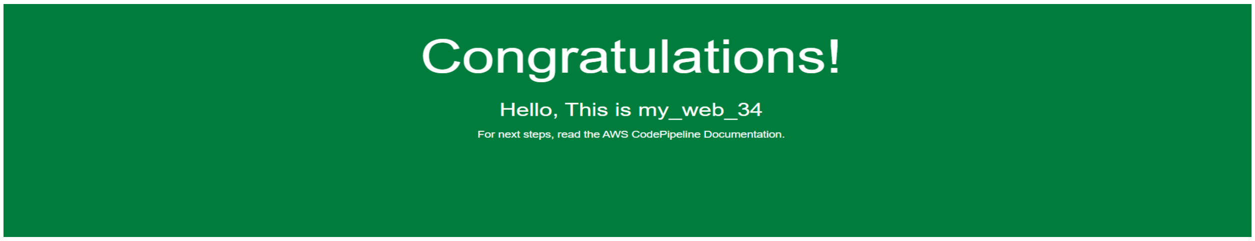


Congratulations!

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.

Step 22: Now, we make some changes to the index.html file in the github.
For eg: If you make some changes to the <h2>tag.
Once the changes are committed, when the website is refreshed, the changes can be seen.



Congratulations!

Hello, This is my_web_34

For next steps, read the AWS CodePipeline Documentation.