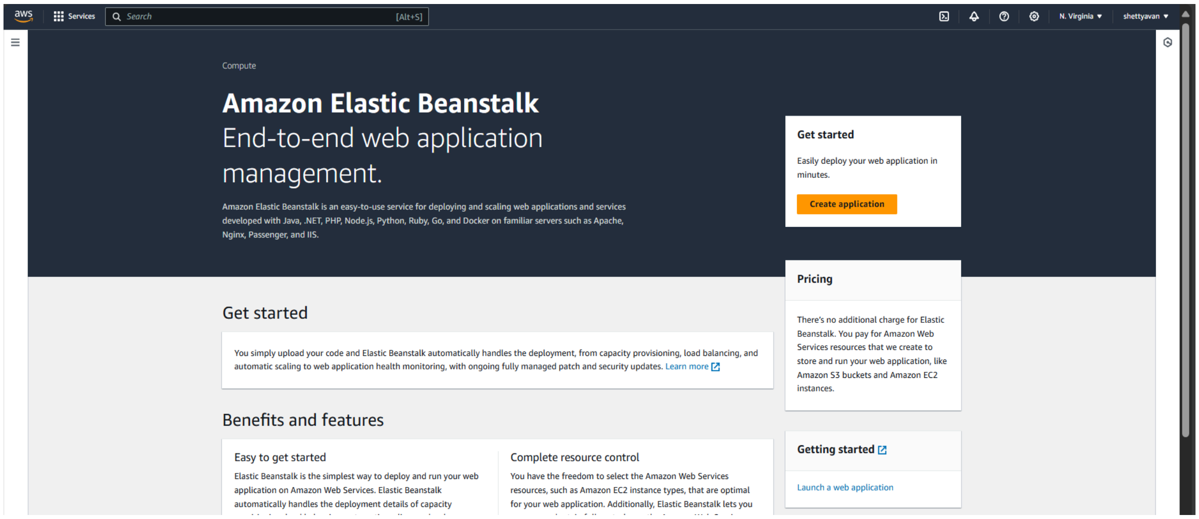
Krushikesh Shelar D15C 51

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.

1. Open the aws console and then search Elastic Beanstalk (Opens a dashboard as seen below)



Compute

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.

Create application

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

Benefits and features

Easy to get started

Elastic Beanstalk is the simplest way to deploy and run your web application on Amazon Web Services. Elastic Beanstalk automatically handles the deployment details of capacity

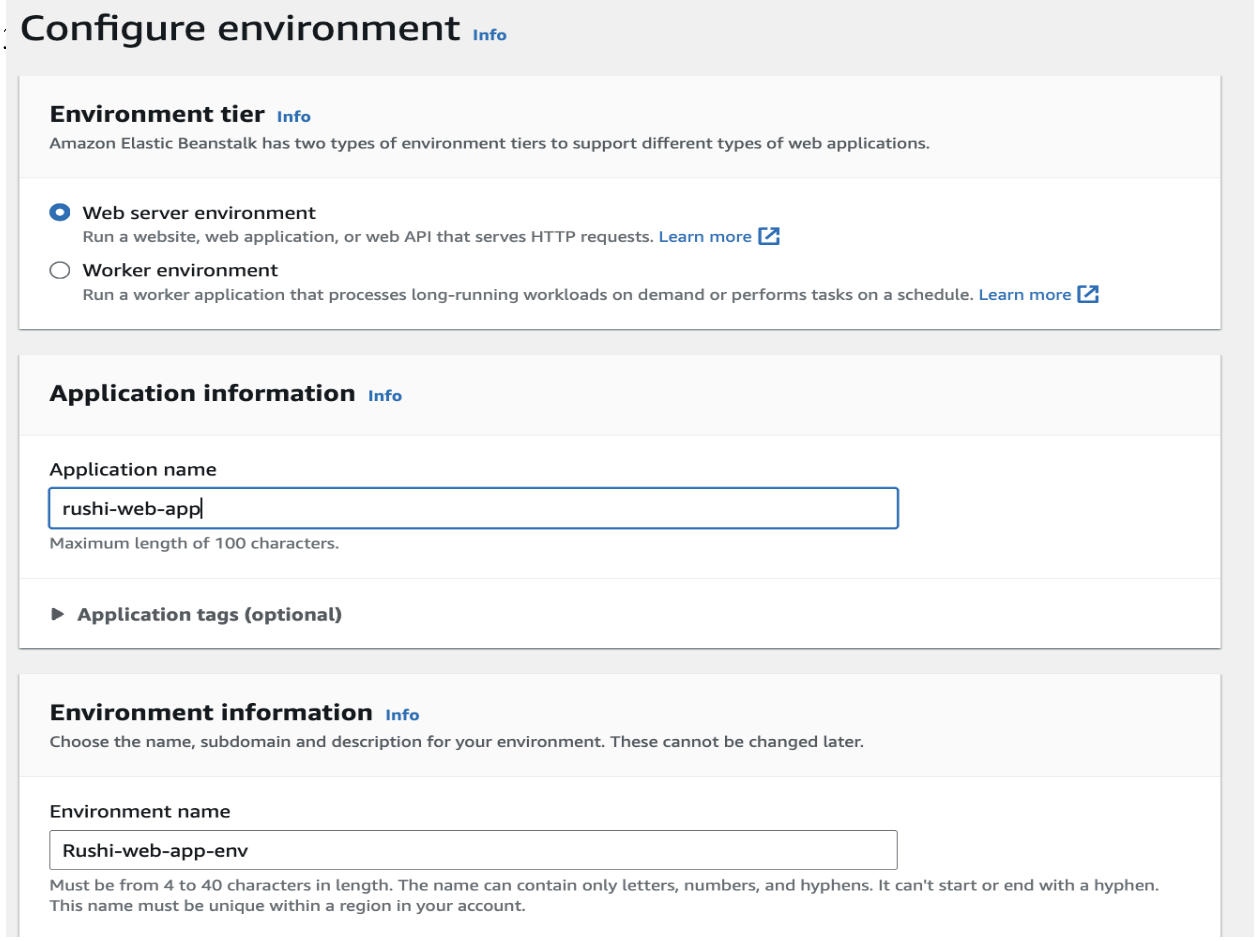
Complete resource control

You have the freedom to select the Amazon Web Services resources, such as Amazon EC2 instance types, that are optimal for your web application. Additionally, Elastic Beanstalk lets you

Getting started

Launch a web application

2. Click on create application and configure the environment by adding your application name



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

rushi-web-app

Maximum length of 100 characters.

▶ Application tags (optional)

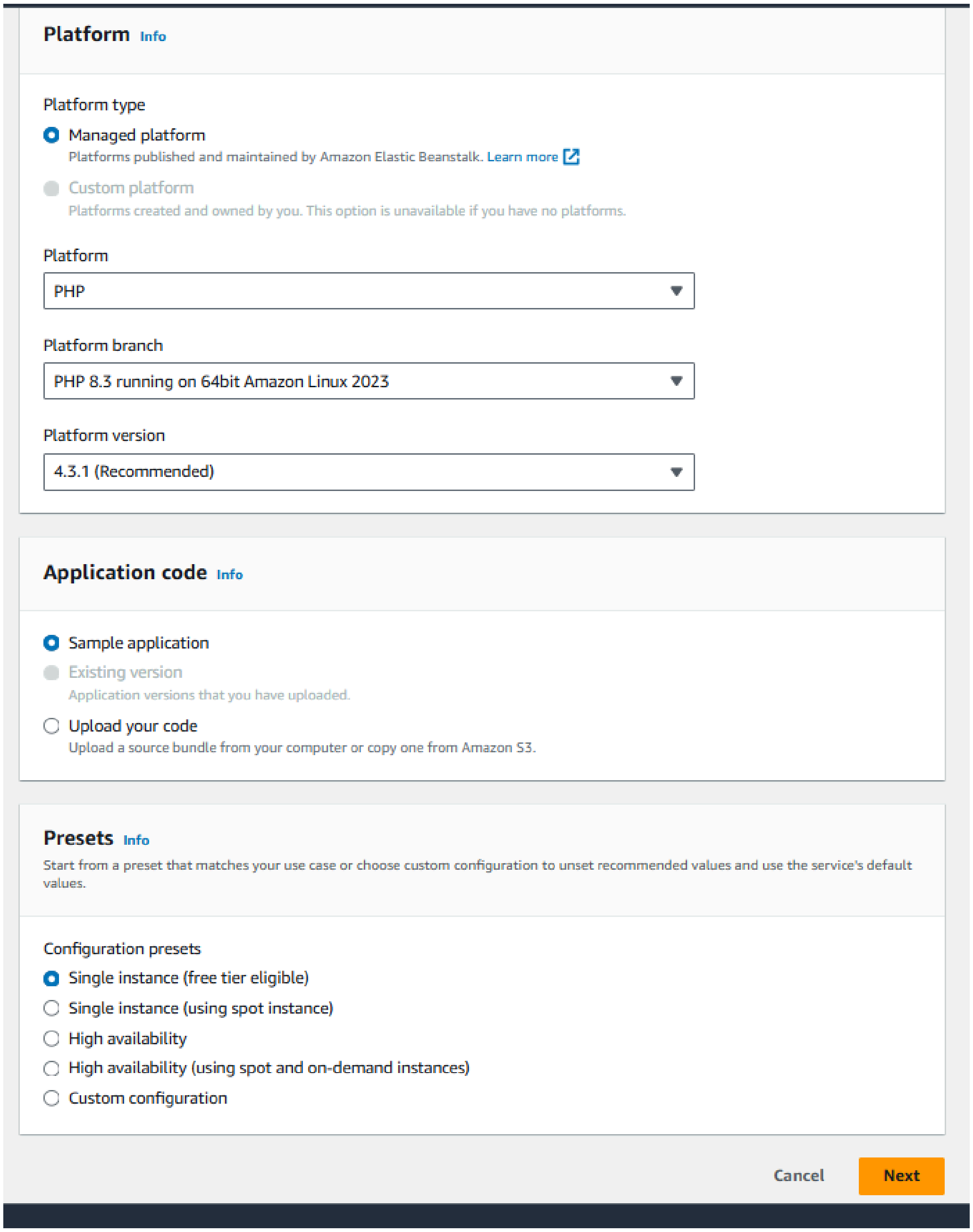
Environment information Info

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

Environment name

Rushi-web-app-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.



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)

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.

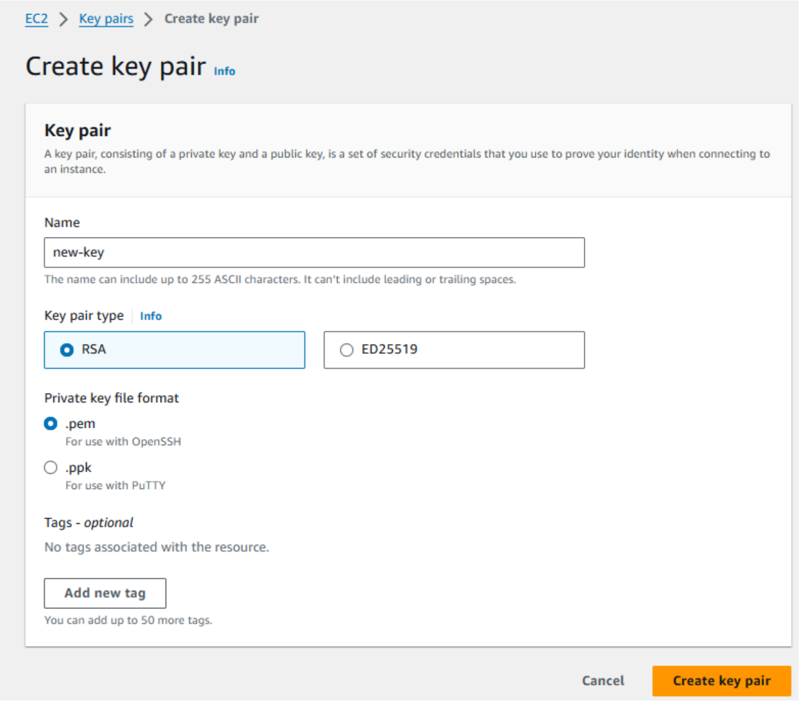
Presets Info

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

4. In this step you have to create a key pair
   a. Go to EC2 instance tab and from the left panel create a key pair. As the key pair might be useful for the further process



EC2 > Key pairs > Create key pair

Create key pair Info

Key pair

A key pair, consisting of a private key and a public key, is a set of security credentials that you use to prove your identity when connecting to an instance.

Name: new-key

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type Info

● RSA ○ ED25519

Private key file format

● .pem — For use with OpenSSH

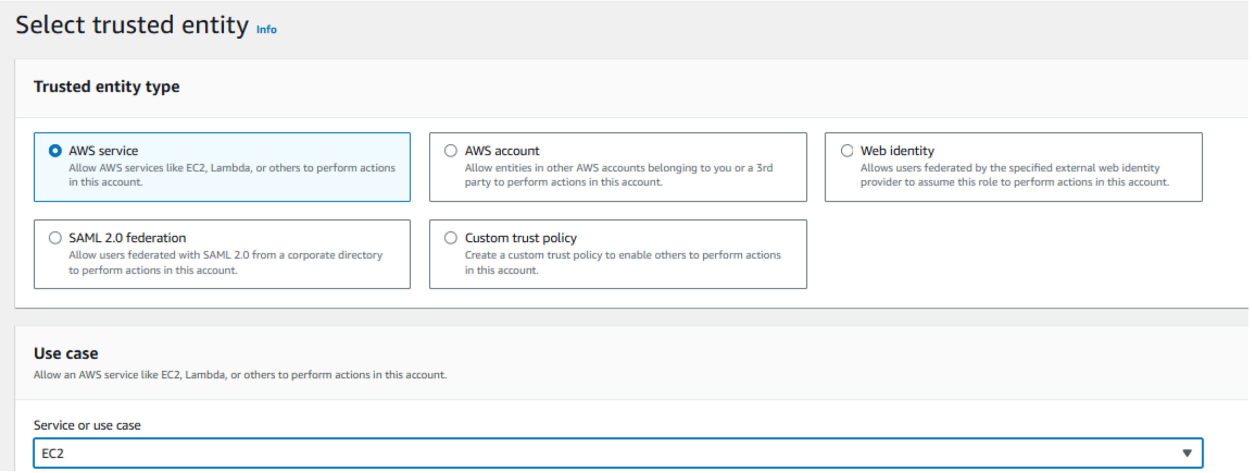
○ .ppk — For use with PuTTY

Tags - *optional*

No tags associated with the resource.

You can add up to 50 more tags.

b. In the same fashion go to IAM and then under roles section click create role and then Select AWS Service and under instances select EC2. Your EC2 Instance role is created



Select trusted entity Info

Trusted entity type

● AWS service — Allow AWS services like EC2, Lambda, or others to perform actions in this account.

○ AWS account — Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

○ Web identity — Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

○ SAML 2.0 federation — Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

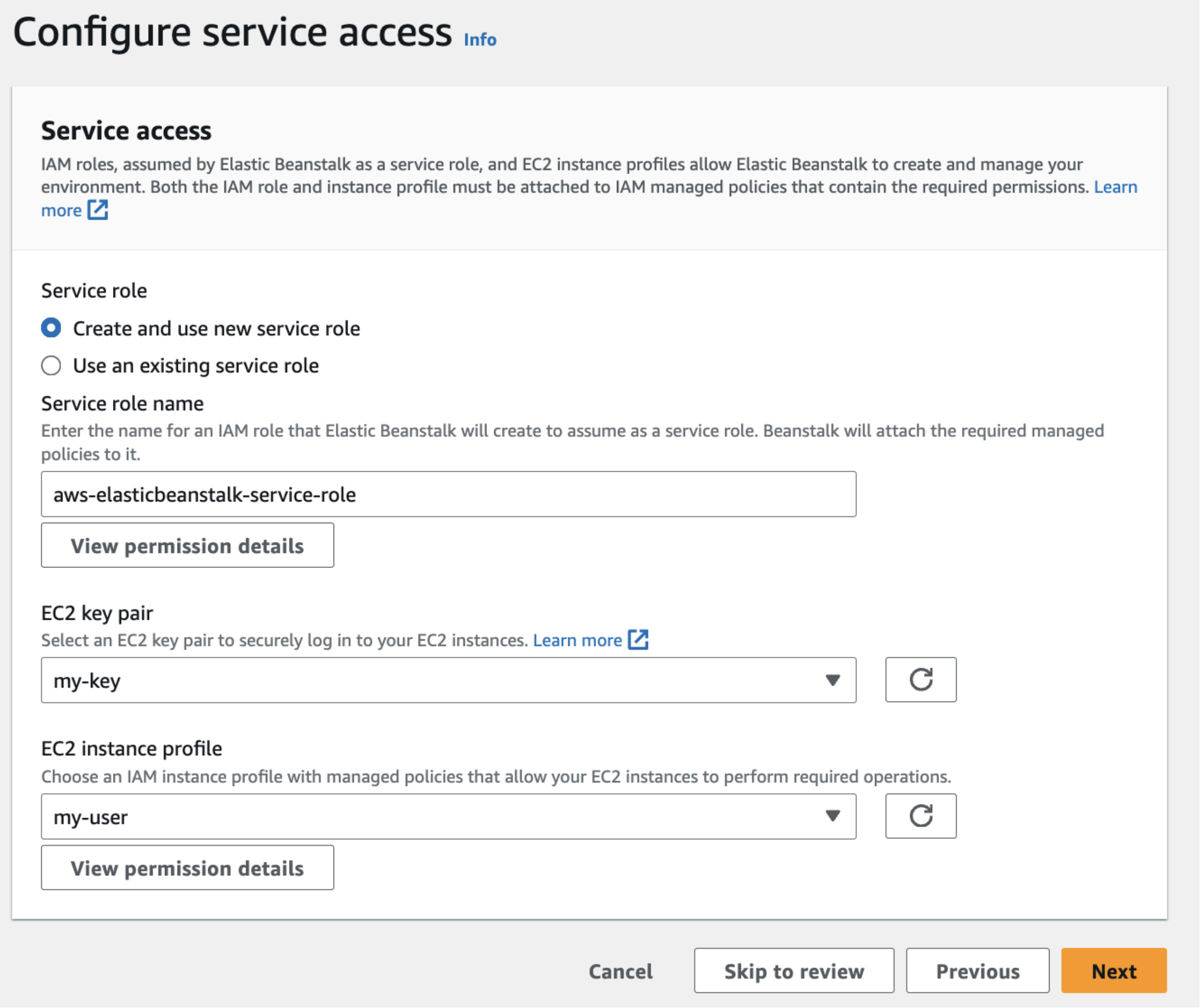
○ Custom trust policy — Create a custom trust policy to enable others to perform actions in this account.

Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case: EC2

5. Now comeback to Elastic Beanstalk page and from the drop down menu select the newly created key pair and instance profile



Configure service access

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

Service role name

Enter the name for an IAM role that Elastic Beanstalk will create to assume as a service role. Beanstalk will attach the required managed policies to it.

aws-elasticbeanstalk-service-role

EC2 key pair

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

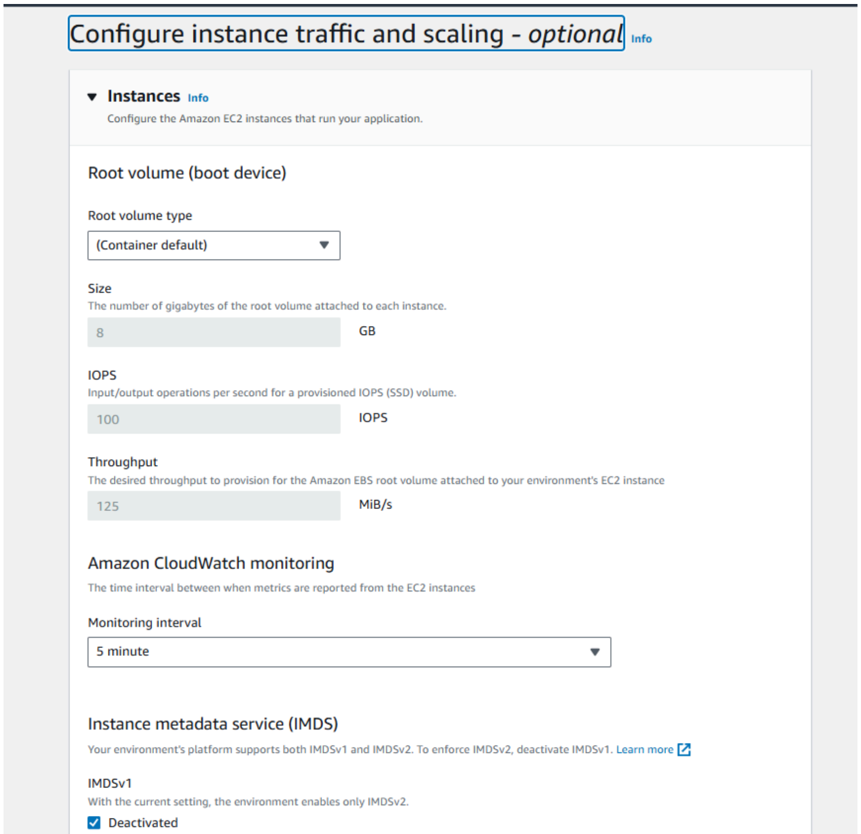
my-key

EC2 instance profile

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

my-user

6. No changes further in the Configure instance traffic section



Configure instance traffic and scaling - *optional*

▼ **Instances**

Configure the Amazon EC2 instances that run your application.

Root volume (boot device)

Root volume type

(Container default) ▼

Size

The number of gigabytes of the root volume attached to each instance.

8 GB

IOPS

Input/output operations per second for a provisioned IOPS (SSD) volume.

100 IOPS

Throughput

The desired throughput to provision for the Amazon EBS root volume attached to your environment's EC2 instance

125 MiB/s

Amazon CloudWatch monitoring

The time interval between when metrics are reported from the EC2 instances

Monitoring interval

5 minute ▼

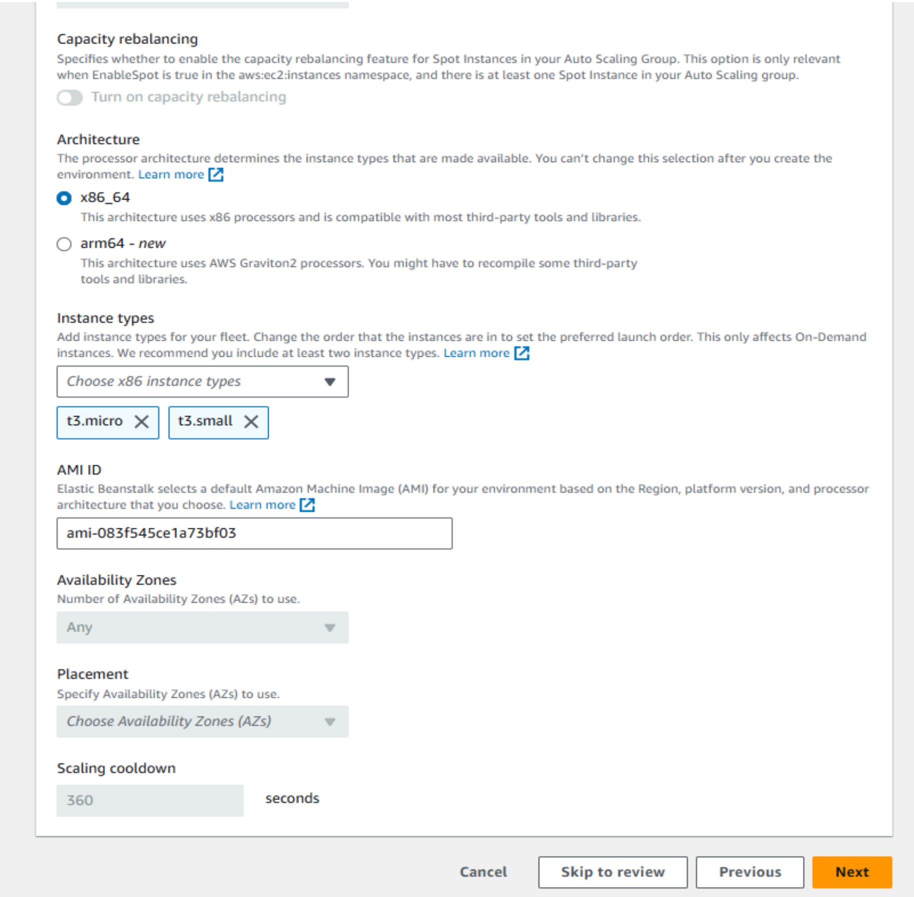
Instance metadata service (IMDS)

Your environment's platform supports both IMDSv1 and IMDSv2. To enforce IMDSv2, deactivate IMDSv1. Learn more

IMDSv1

With the current setting, the environment enables only IMDSv2.

☑ Deactivated



Capacity rebalancing

Specifies whether to enable the capacity rebalancing feature for Spot Instances in your Auto Scaling Group. This option is only relevant when EnableSpot is true in the aws:ec2:instances namespace, and there is at least one Spot Instance in your Auto Scaling group.

Turn on capacity rebalancing

Architecture

The processor architecture determines the instance types that are made available. You can't change this selection after you create the environment. Learn more

● x86_64

This architecture uses x86 processors and is compatible with most third-party tools and libraries.

○ arm64 - *new*

This architecture uses AWS Graviton2 processors. You might have to recompile some third-party tools and libraries.

Instance types

Add instance types for your fleet. Change the order that the instances are in to set the preferred launch order. This only affects On-Demand instances. We recommend you include at least two instance types. Learn more

Choose x86 instance types

t3.micro ✕ t3.small ✕

AMI ID

Elastic Beanstalk selects a default Amazon Machine Image (AMI) for your environment based on the Region, platform version, and processor architecture that you choose. Learn more

ami-083f545ce1a73bf03

Availability Zones

Number of Availability Zones (AZs) to use.

Any

Placement

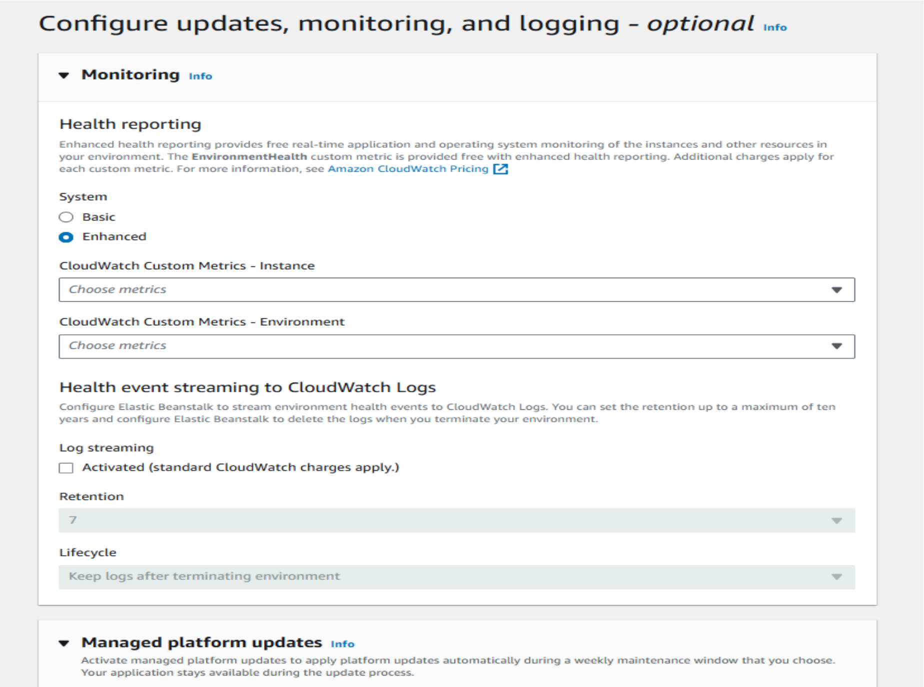
Specify Availability Zones (AZs) to use.

Choose Availability Zones (AZs)

Scaling cooldown

360 seconds

7. Again no changes in the configure updates, monitoring, and logging part just click on next.



Configure updates, monitoring, and logging - *optional* Info

▾ **Monitoring** Info

Health reporting

Enhanced health reporting provides free real-time application and operating system monitoring of the instances and other resources in your environment. The **EnvironmentHealth** custom metric is provided free with enhanced health reporting. Additional charges apply for each custom metric. For more information, see Amazon CloudWatch Pricing

System

○ Basic

● Enhanced

CloudWatch Custom Metrics - Instance

Choose metrics ▾

CloudWatch Custom Metrics - Environment

Choose metrics ▾

Health event streaming to CloudWatch Logs

Configure Elastic Beanstalk to stream environment health events to CloudWatch Logs. You can set the retention up to a maximum of ten years and configure Elastic Beanstalk to delete the logs when you terminate your environment.

Log streaming

□ Activated (standard CloudWatch charges apply.)

Retention

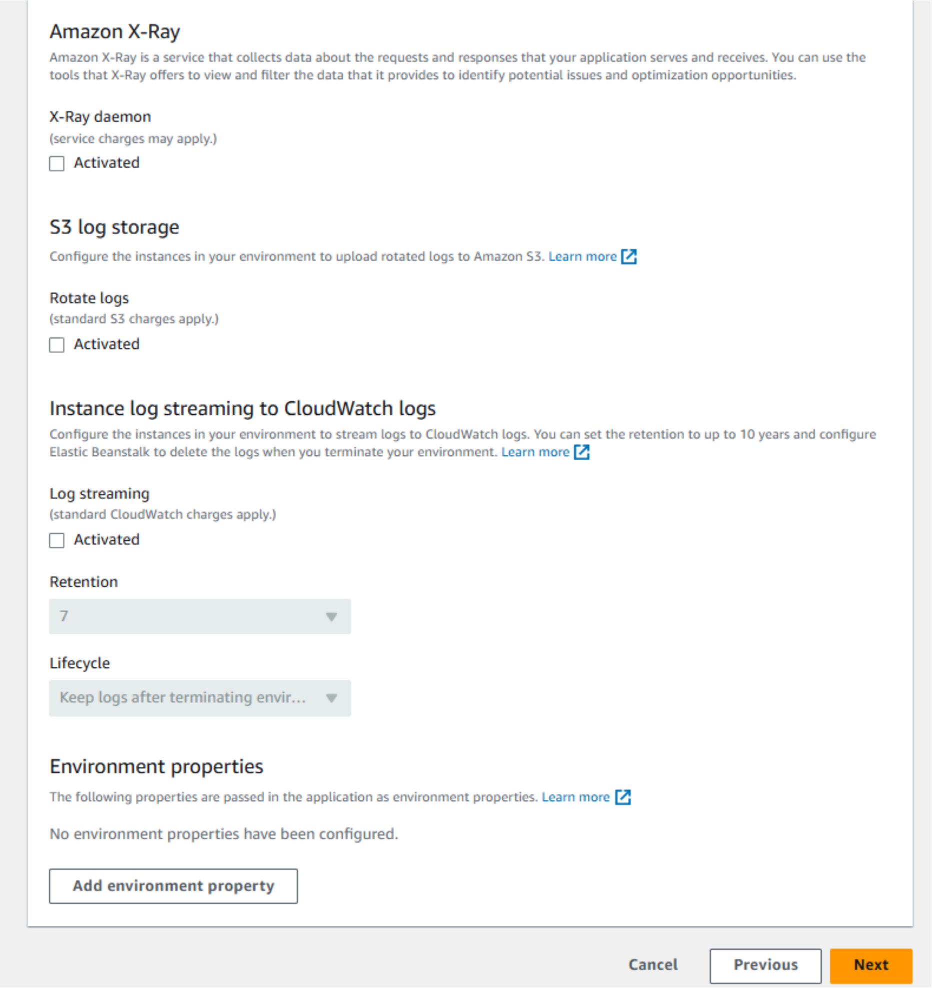
7 ▾

Lifecycle

Keep logs after terminating environment ▾

▾ **Managed platform updates** Info

Activate managed platform updates to apply platform updates automatically during a weekly maintenance window that you choose. Your application stays available during the update process.



Amazon X-Ray

Amazon X-Ray is a service that collects data about the requests and responses that your application serves and receives. You can use the tools that X-Ray offers to view and filter the data that it provides to identify potential issues and optimization opportunities.

X-Ray daemon

(service charges may apply.)

□ Activated

S3 log storage

Configure the instances in your environment to upload rotated logs to Amazon S3. Learn more

Rotate logs

(standard S3 charges apply.)

□ Activated

Instance log streaming to CloudWatch logs

Configure the instances in your environment to stream logs to CloudWatch logs. You can set the retention to up to 10 years and configure Elastic Beanstalk to delete the logs when you terminate your environment. Learn more

Log streaming

(standard CloudWatch charges apply.)

□ Activated

Retention

7 ▾

Lifecycle

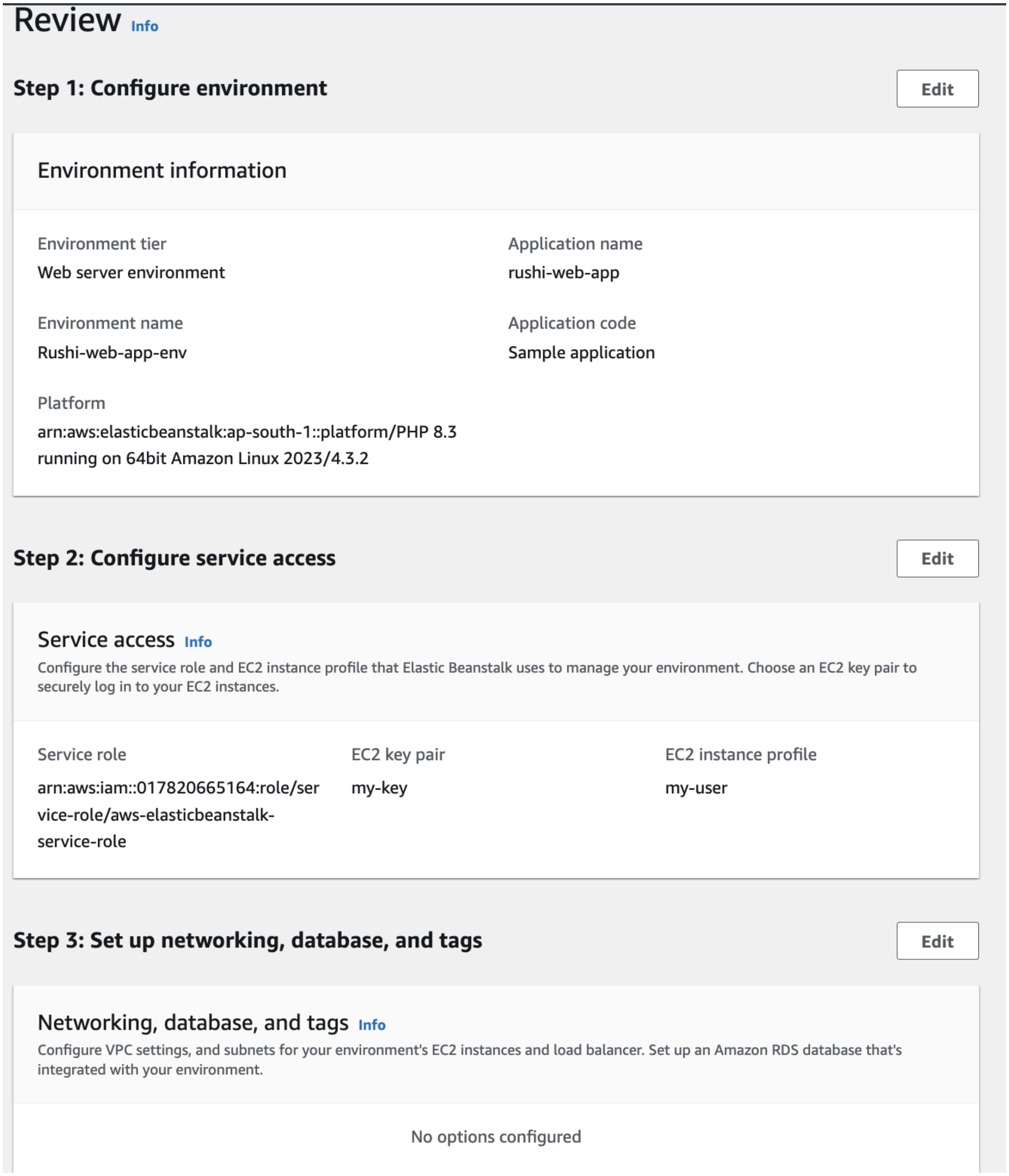
Keep logs after terminating envir... ▾

Environment properties

The following properties are passed in the application as environment properties. Learn more

No environment properties have been configured.

8. Now review the changes made and click on create application



Review

Step 1: Configure environment

Environment information

Environment tier
Web server environment

Application name
rushi-web-app

Environment name
Rushi-web-app-env

Application code
Sample application

Platform
arn:aws:elasticbeanstalk:ap-south-1::platform/PHP 8.3 running on 64bit Amazon Linux 2023/4.3.2

Step 2: Configure service access

Service access

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.

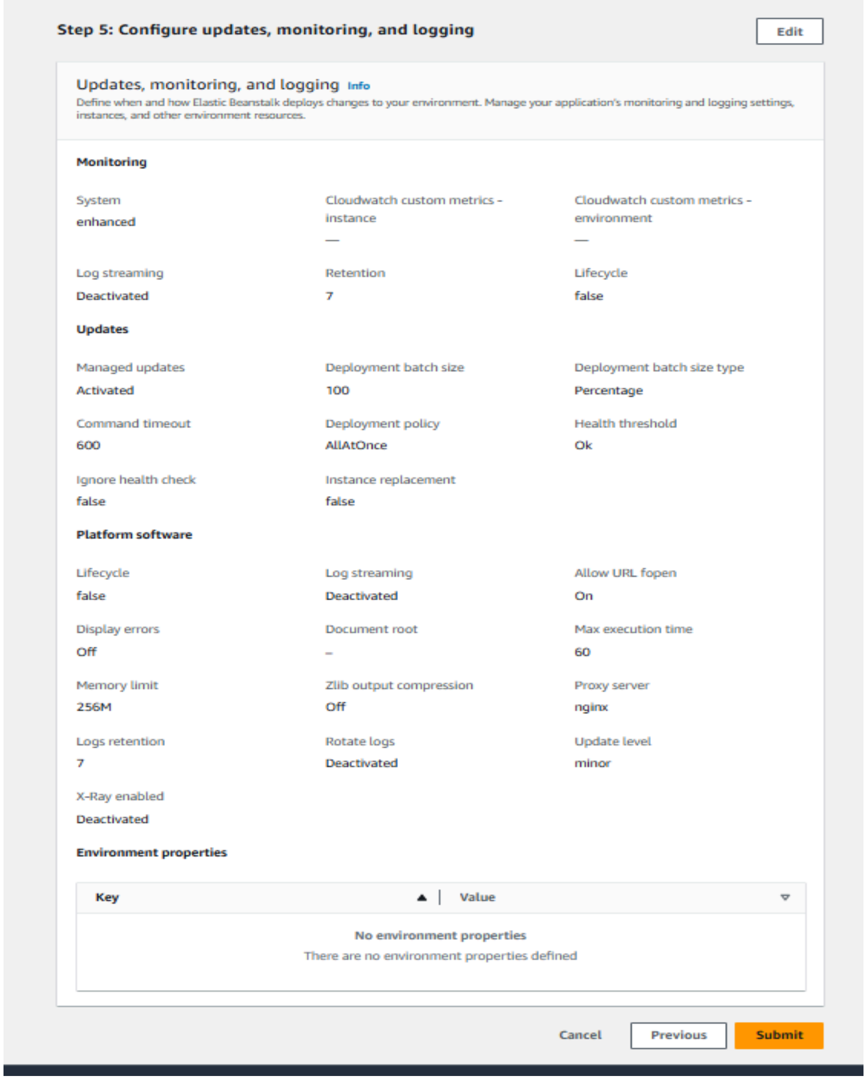
Service role	EC2 key pair	EC2 instance profile
arn:aws:iam::017820665164:role/service-role/aws-elasticbeanstalk-service-role	my-key	my-user

Step 3: Set up networking, database, and tags

Networking, database, and tags

Configure VPC settings, and subnets for your environment's EC2 instances and load balancer. Set up an Amazon RDS database that's integrated with your environment.

No options configured



Step 5: Configure updates, monitoring, and logging

Updates, monitoring, and logging Info

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 false	Instance replacement false	

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		

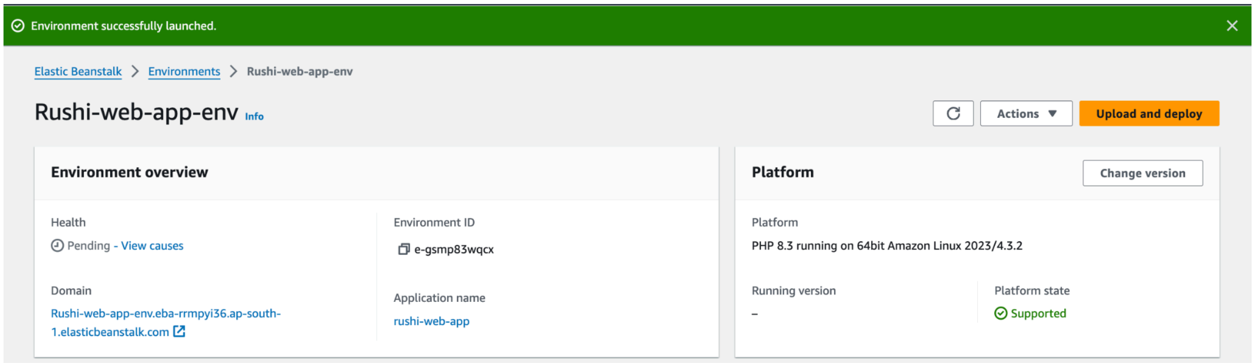
Environment properties

Key	Value

No environment properties

There are no environment properties defined

9. Your sample environment is created for you to deploy your application. By default, it creates an EC2 instance, a security group, an Auto Scaling group, an Amazon S3 Bucket, Amazon CloudWatch alarms and a domain name for your Application.



Environment successfully launched.

Elastic Beanstalk > Environments > Rushi-web-app-env

Rushi-web-app-env Info

Environment overview

Health
Pending - View causes

Environment ID
e-gsmp83wqcx

Domain
Rushi-web-app-env.eba-rrmpyi36.ap-south-1.elasticbeanstalk.com

Application name
rushi-web-app

Platform

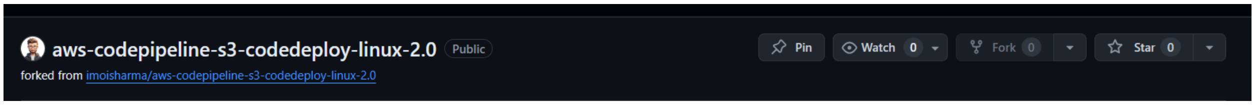
Platform
PHP 8.3 running on 64bit Amazon Linux 2023/4.3.2

Running version
–

Platform state
Supported

Pipeline creation:

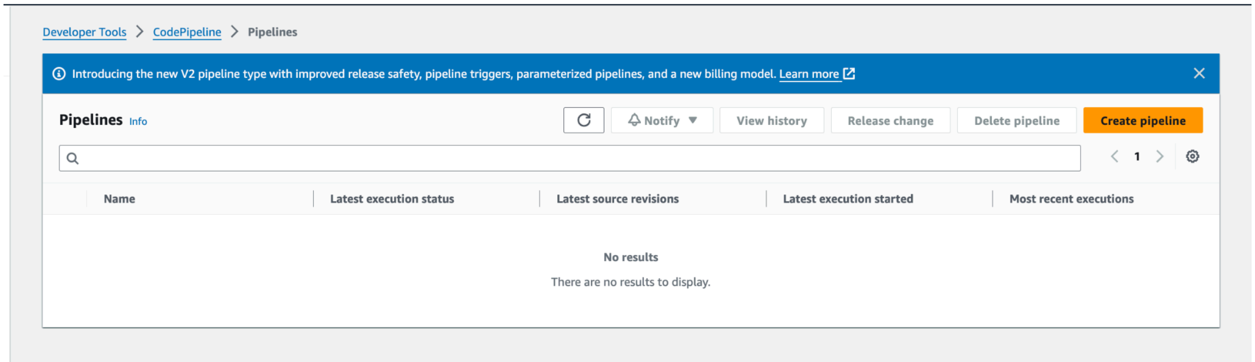
1. Fork a github repo for aws codepipeline available as The pipeline takes code from the source and then performs actions on it.We don't need to code from scratch in this manner



aws-codepipeline-s3-codedeploy-linux-2.0 Public

forked from imoisharma/aws-codepipeline-s3-codedeploy-linux-2.0

2. Go to developer tools and select CodePipeline and create a new pipeline



Developer Tools > CodePipeline > Pipelines

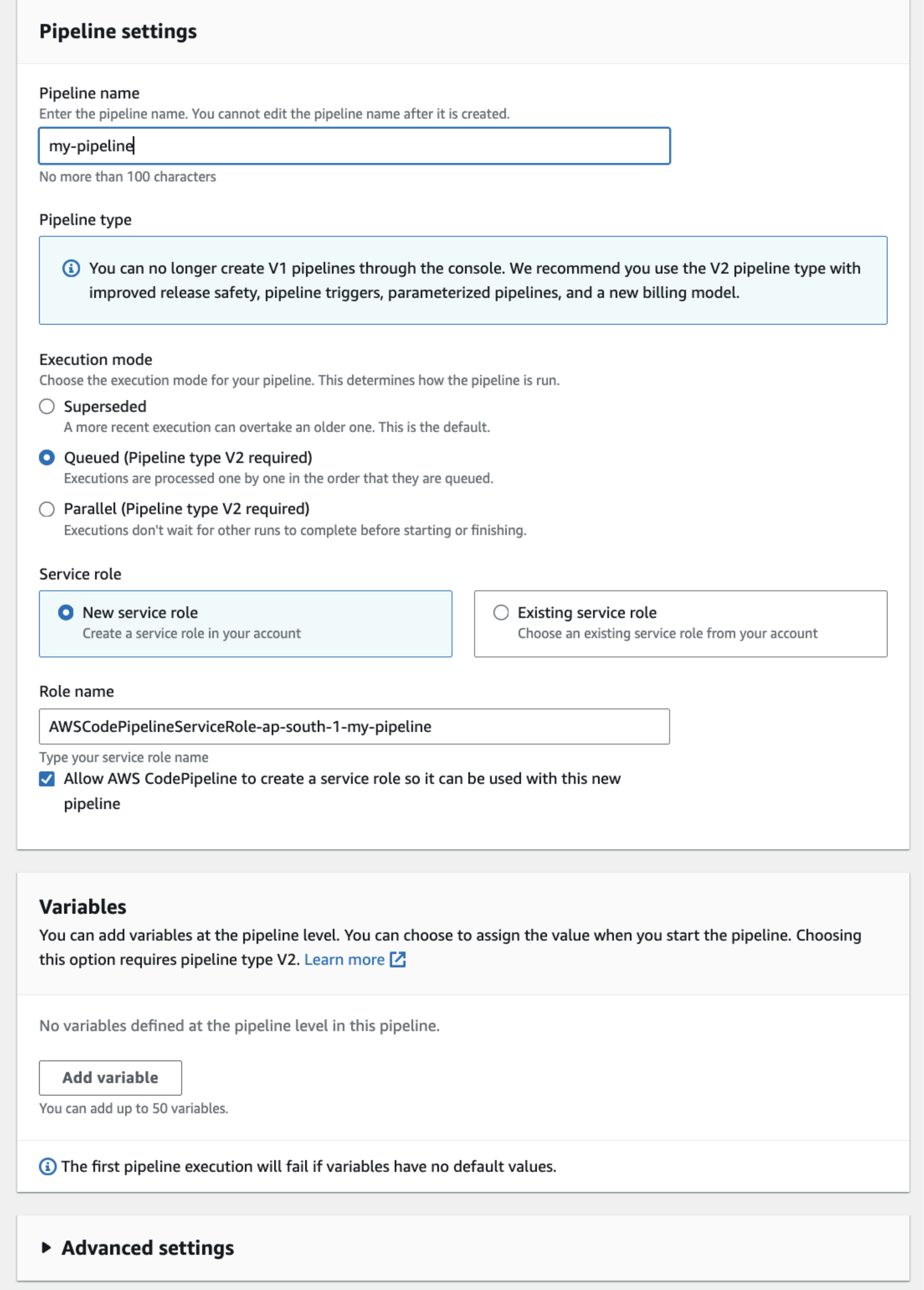
Introducing the new V2 pipeline type with improved release safety, pipeline triggers, parameterized pipelines, and a new billing model. Learn more

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

No results

There are no results to display.

3. Name your pipeline and select the desired service role



Pipeline settings

Pipeline name

Enter the pipeline name. You cannot edit the pipeline name after it is created.

my-pipeline

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
Create a service role in your account

○ Existing service role
Choose an existing service role from your account

Role name

AWSCodePipelineServiceRole-ap-south-1-my-pipeline

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.

You can add up to 50 variables.

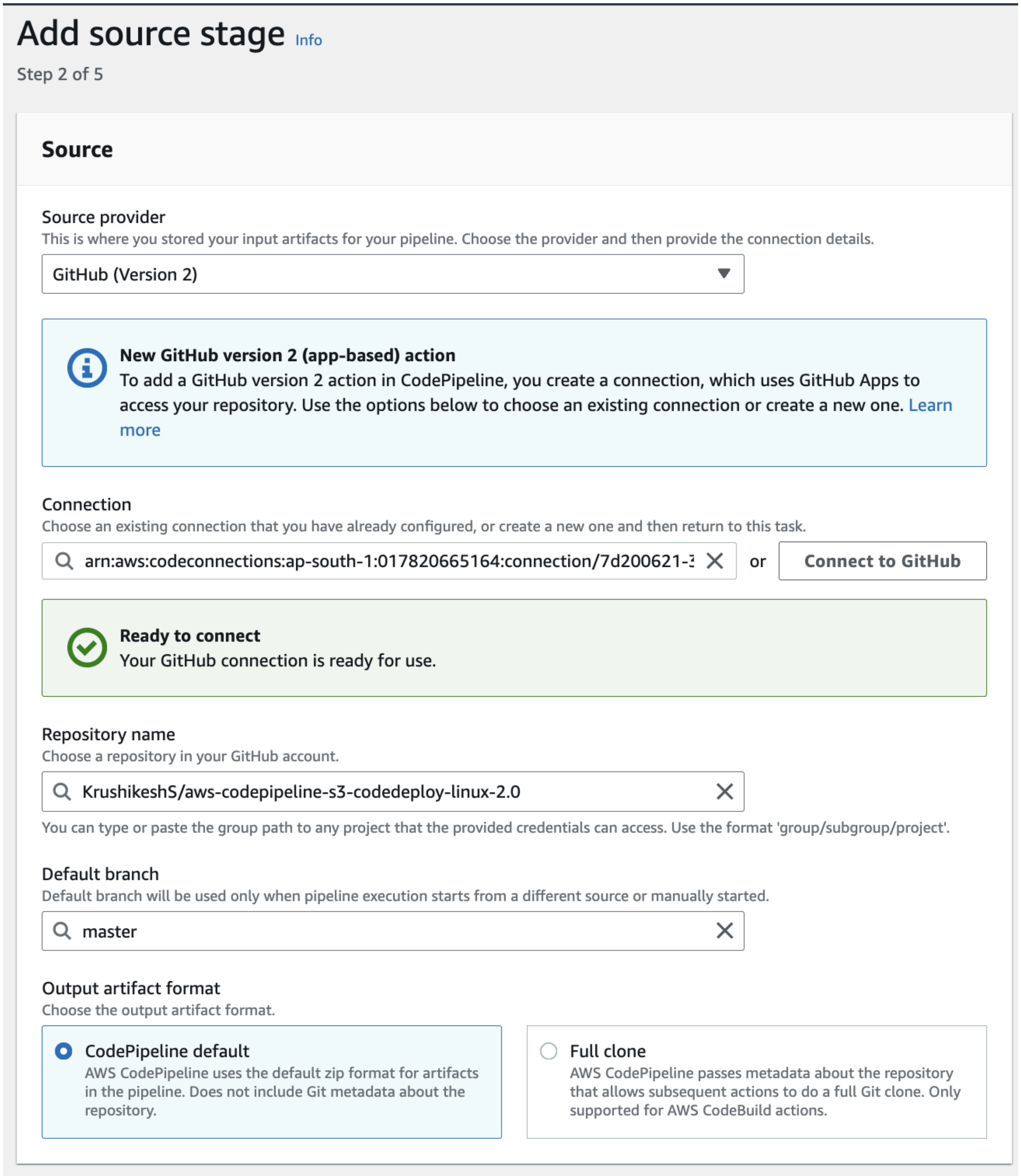
ⓘ The first pipeline execution will fail if variables have no default values.

▶ Advanced settings

4. In the source stage select Github v2 as the provider and then connect your github connect so that the pipeline can access the forked source code .

   a. For this purpose create aws github connection and with your credentials install the AWS under the forked repository

5. Once the connection is established from the drop down menu select the repository and the branch



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:ap-south-1:017820665164:connection/7d200621-3 or Connect to GitHub

Ready to connect
Your GitHub connection is ready for use.

Repository name
Choose a repository in your GitHub account.
KrushikeshS/aws-codepipeline-s3-codedeploy-linux-2.0
You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

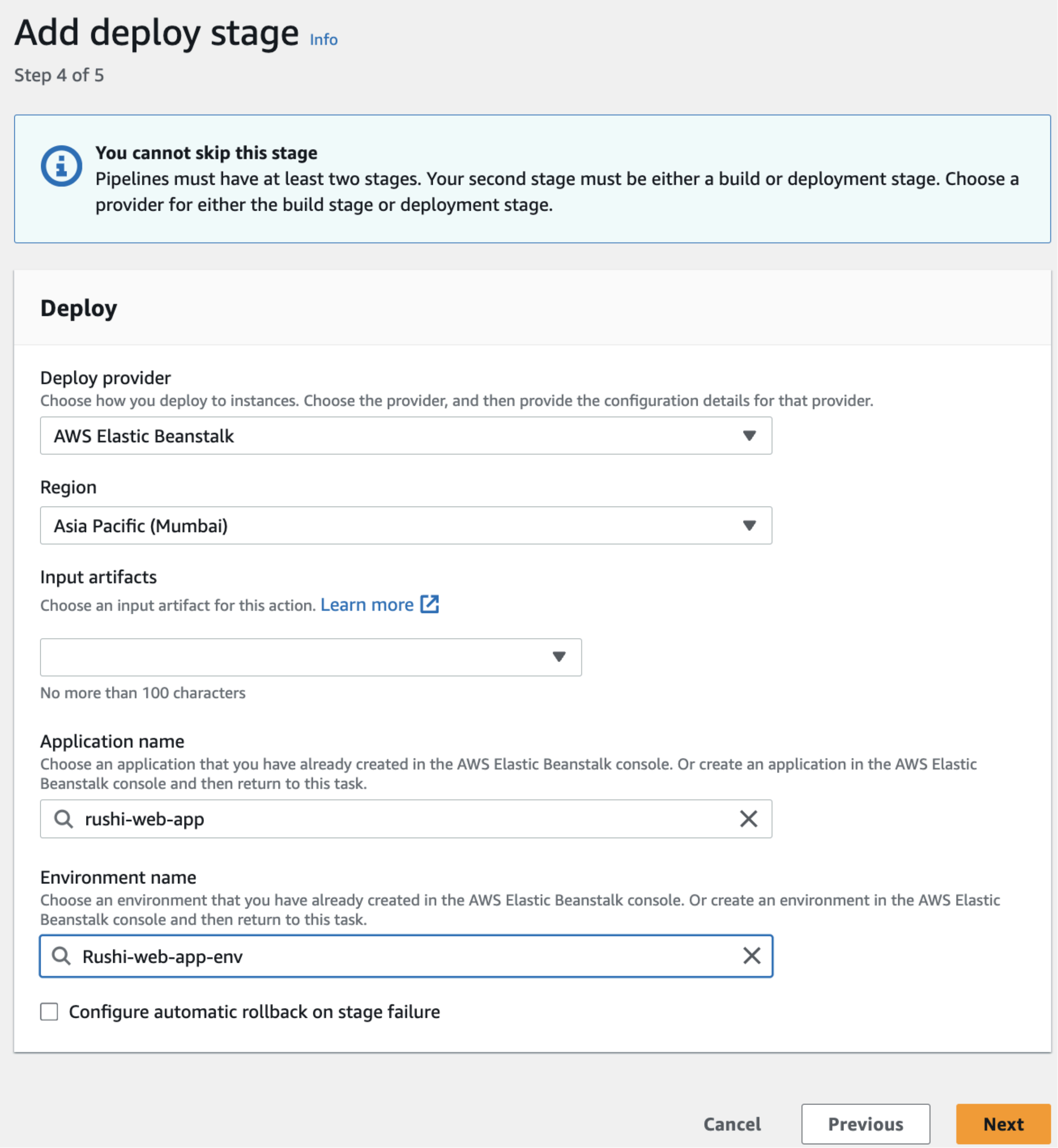
Default branch
Default branch will be used only when pipeline execution starts from a different source or manually started.
master

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.

6. Skip the build stage part as we are not plugging in any build provider and in choose Beanstalk as the Deploy Provider, same region as the Bucket and Beanstalk, name and environment name.



Add deploy stage Info

Step 4 of 5

You cannot skip this stage
Pipelines must have at least two stages. Your second stage must be either a build or deployment stage. Choose a provider for either the build stage or deployment stage.

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
Asia Pacific (Mumbai)

Input artifacts
Choose an input artifact for this action. Learn more

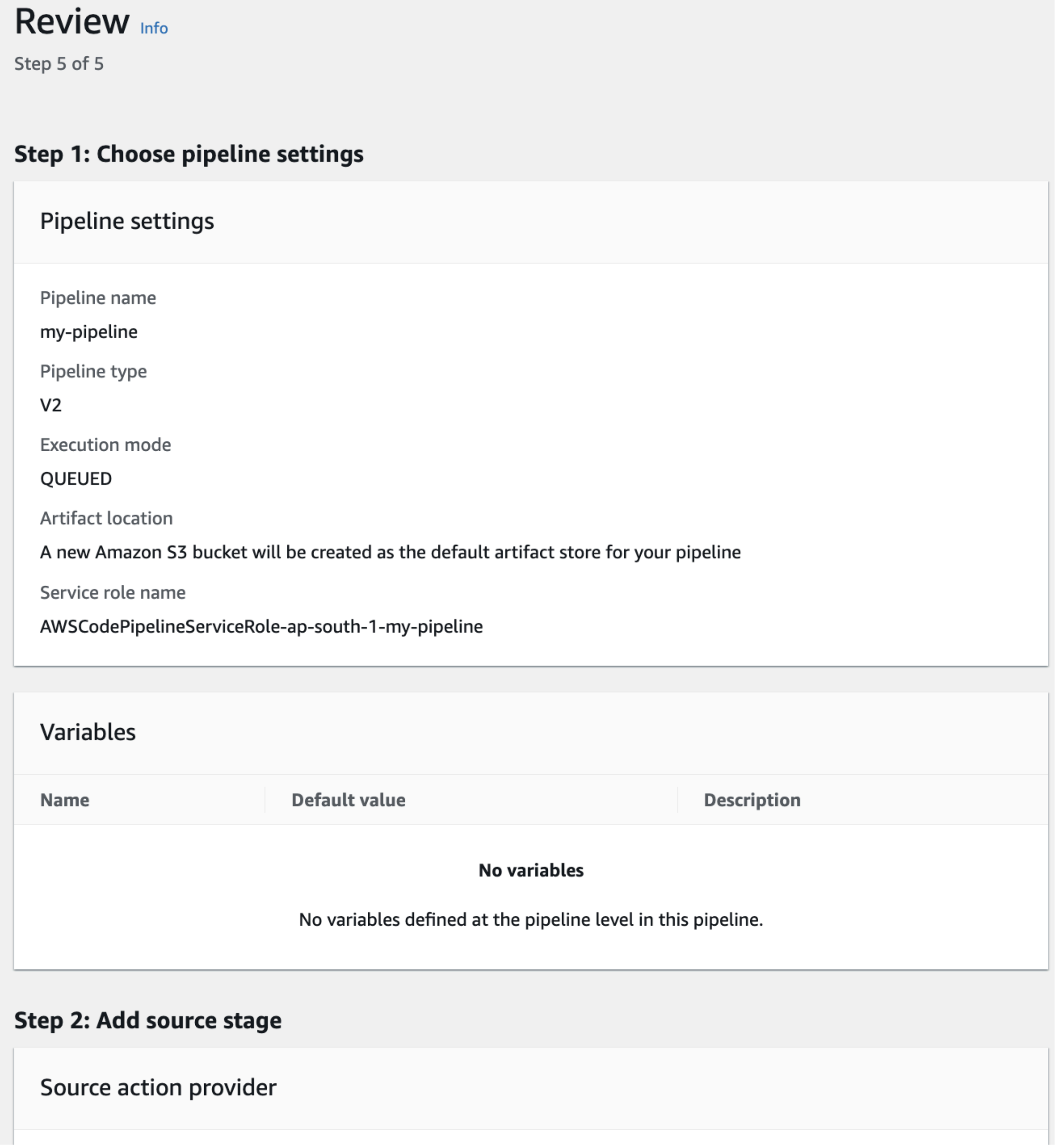
No more than 100 characters

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.
rushi-web-app

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.
Rushi-web-app-env

☐ Configure automatic rollback on stage failure

7. Review the settings and click on create pipeline



Review Info

Step 5 of 5

Step 1: Choose pipeline settings

Pipeline settings

Pipeline name
my-pipeline

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-ap-south-1-my-pipeline

Variables

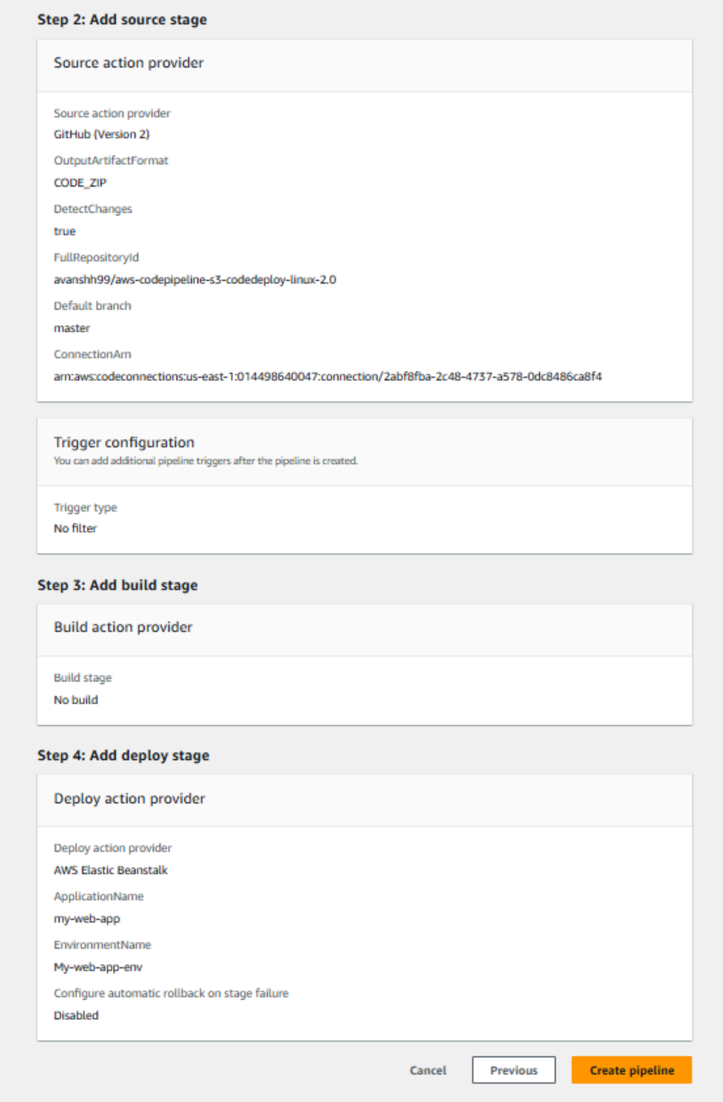
Name	Default value	Description

No variables

No variables defined at the pipeline level in this pipeline.

Step 2: Add source stage

Source action provider



Step 2: Add source stage

Source action provider

Source action provider
GitHub (Version 2)

OutputArtifactFormat
CODE_ZIP

DetectChanges
true

FullRepositoryId
avanshh99/aws-codepipeline-s3-codedeploy-linux-2.0

Default branch
master

ConnectionArn
arn:aws:codeconnections:us-east-1:014498640047:connection/2abf8fba-2c48-4737-a578-0dc8486ca8f4

Trigger configuration
You can add additional pipeline triggers after the pipeline is created.

Trigger type
No filter

Step 3: Add build stage

Build action provider

Build stage
No build

Step 4: Add deploy stage

Deploy action provider

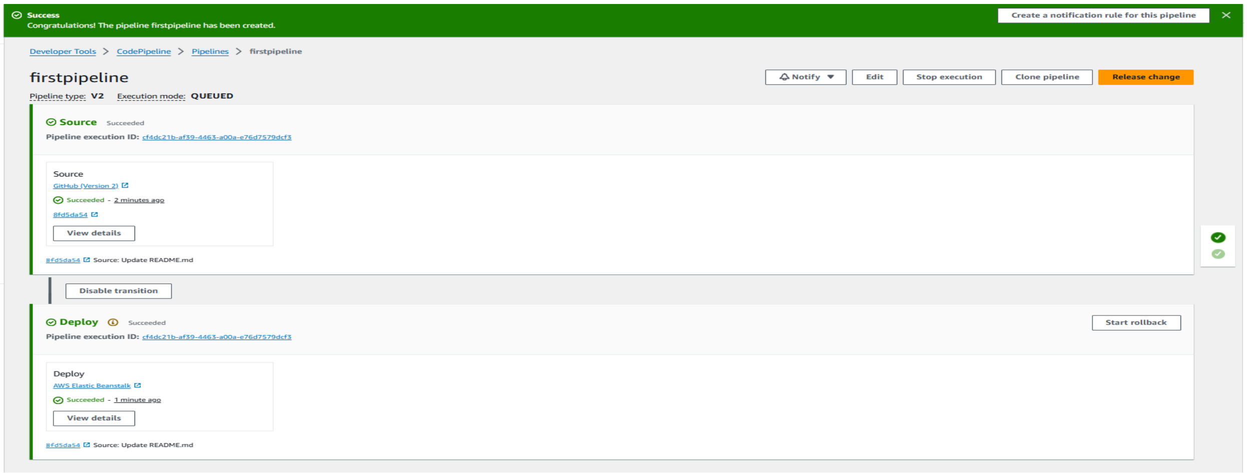
Deploy action provider
AWS Elastic Beanstalk

ApplicationName
my-web-app

EnvironmentName
My-web-app-env

Configure automatic rollback on stage failure
Disabled

8. Once the Successfully created message appears, your pipeline is created.Then go ahead and check the URL provided in the EBS environment.



Success
Congratulations! The pipeline firstpipeline has been created.

Developer Tools > CodePipeline > Pipelines > firstpipeline

firstpipeline

Pipeline type: V2 Execution mode: QUEUED

Source Succeeded
Pipeline execution ID: cf4dc21b-af39-4463-a00a-e76d7579dcf3

Source
GitHub (Version 2)
Succeeded - 2 minutes ago
8fd5da54

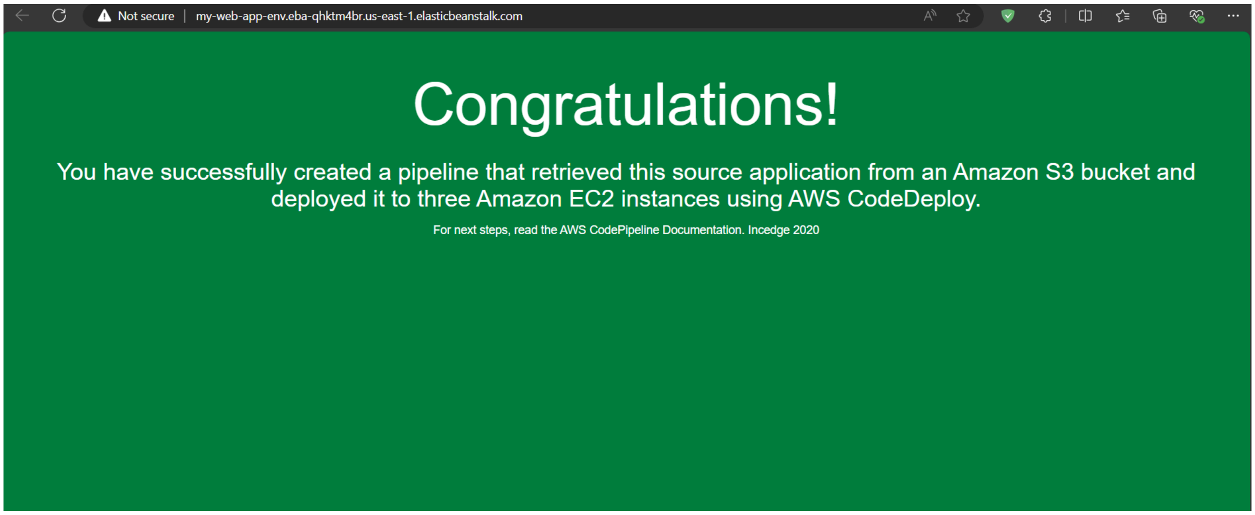
8fd5da54 Source: Update README.md

Deploy Succeeded
Pipeline execution ID: cf4dc21b-af39-4463-a00a-e76d7579dcf3

Deploy
AWS Elastic Beanstalk
Succeeded - 1 minute ago

8fd5da54 Source: Update README.md

9. This is the website hosted from that forked repo in our beanstalk environment



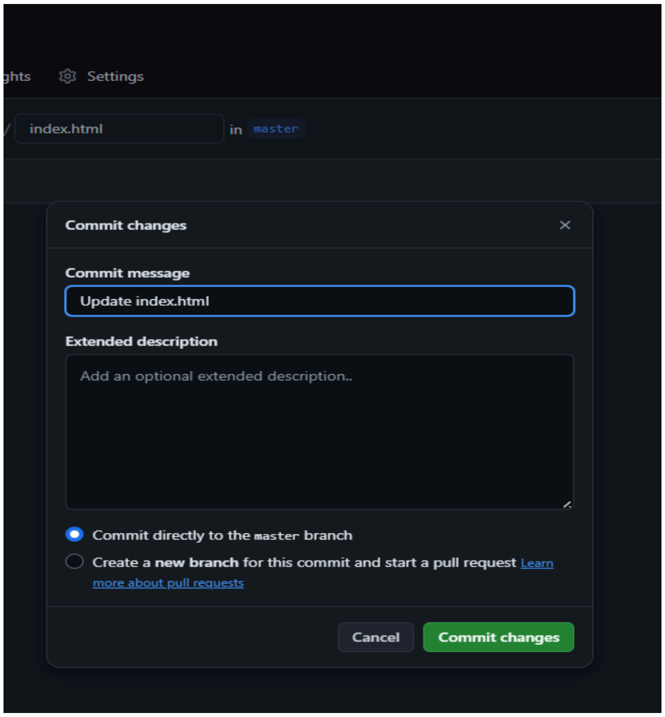
Not secure | my-web-app-env.eba-qhktm4br.us-east-1.elasticbeanstalk.com

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

10. Go to the repository and make the changes in the index.html file and commit them



Commit changes

Commit message

Update index.html

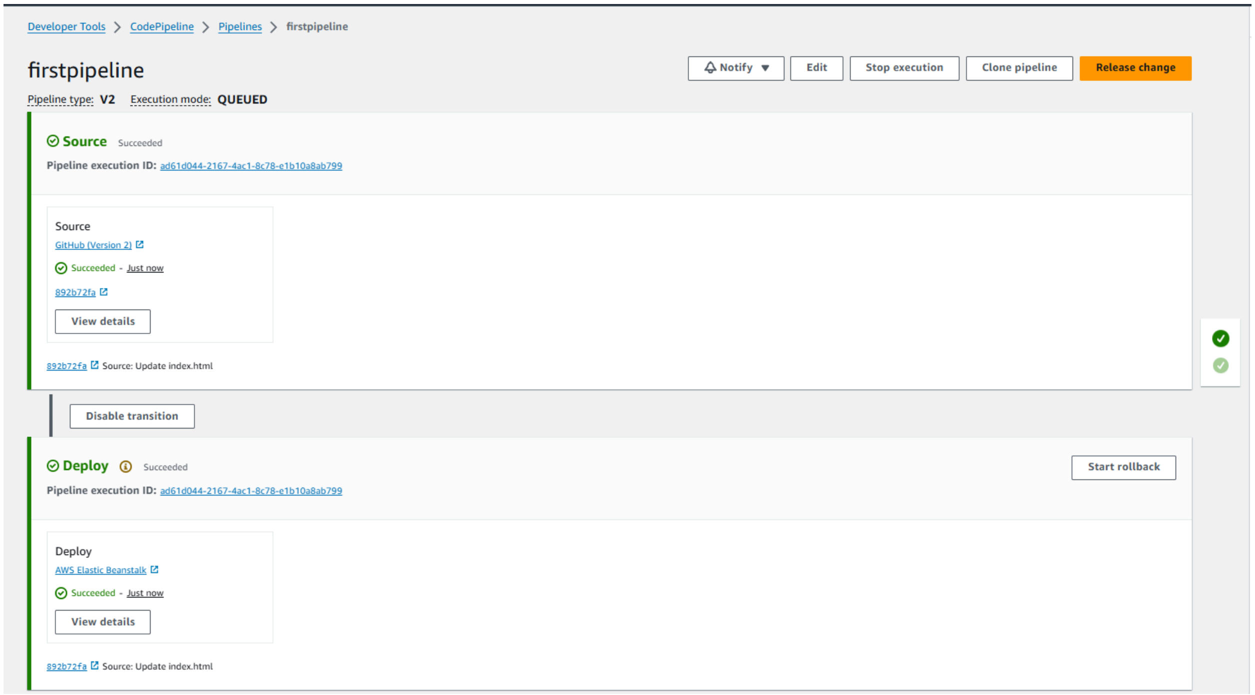
Extended description

Add an optional extended description..

● Commit directly to the master branch

○ Create a new branch for this commit and start a pull request Learn more about pull requests

11.The changes that are committed can be noticed in the source panel in real time and to view the changes check the url (refresh it) and you can view the changes once the deployment section shows success.



firstpipeline

Pipeline type: V2 Execution mode: QUEUED

Source Succeeded

Pipeline execution ID: ad61d044-2167-4ac1-8c78-e1b10a8ab799

Source
GitHub (Version 2)
Succeeded - Just now
892b72fa

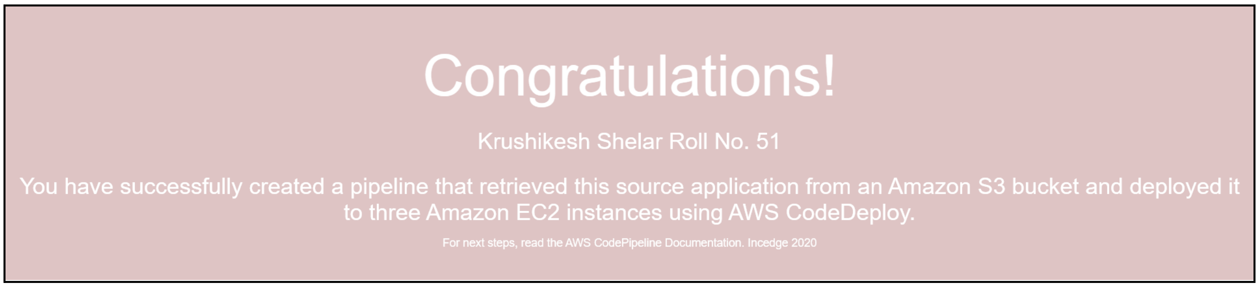
892b72fa Source: Update index.html

Deploy Succeeded

Pipeline execution ID: ad61d044-2167-4ac1-8c78-e1b10a8ab799

Deploy
AWS Elastic Beanstalk
Succeeded - Just now

892b72fa Source: Update index.html



Congratulations!

Krushikesh Shelar Roll No. 51

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