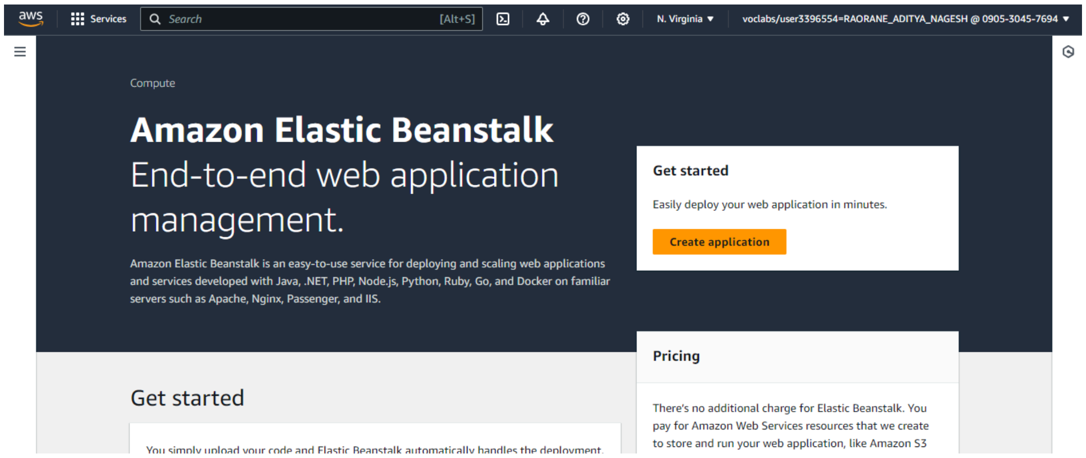
Name: Aditya Nagesh Raorane   Class: D15C[Batch B]   Roll No: 44

2]Application Deployment with AWS Elastic Beanstalk and AWS CodePipeline

Step 1:To open AWS Elastic Beanstalk, log in to the AWS Management Console, navigate to Elastic Beanstalk under "**Compute**," and click “**Create Application**” to start a new project


Compute

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

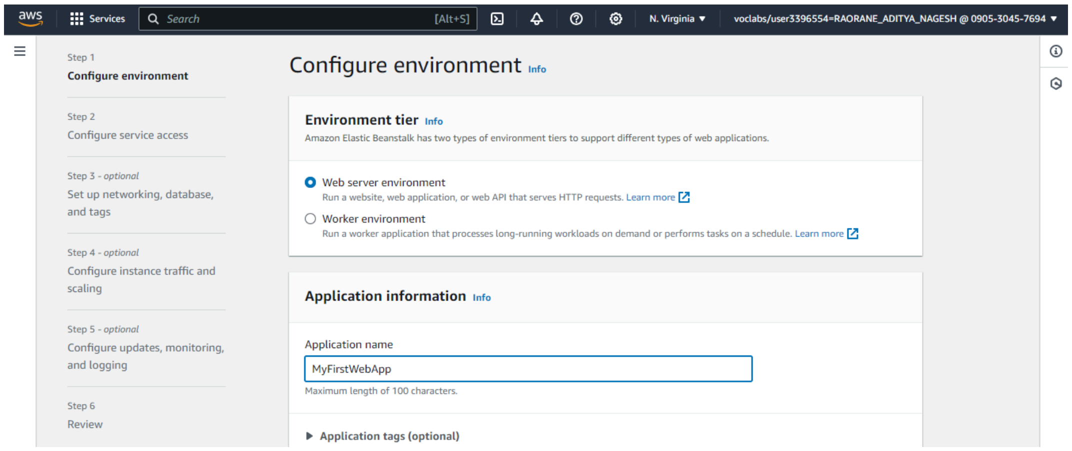
Amazon Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.

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

Get started
You simply upload your code and Elastic Beanstalk automatically handles the deployment

2]After clicking "Create Application" in Elastic Beanstalk, you'll choose between two environment tiers:
Web Server Environment: Handles HTTP requests for web applications with built-in load balancing, auto-scaling, and a pre-configured web server.
Worker Environment: Manages background tasks asynchronously, processing jobs from an SQS queue without direct user interaction.
Click on “Web server Environment”.

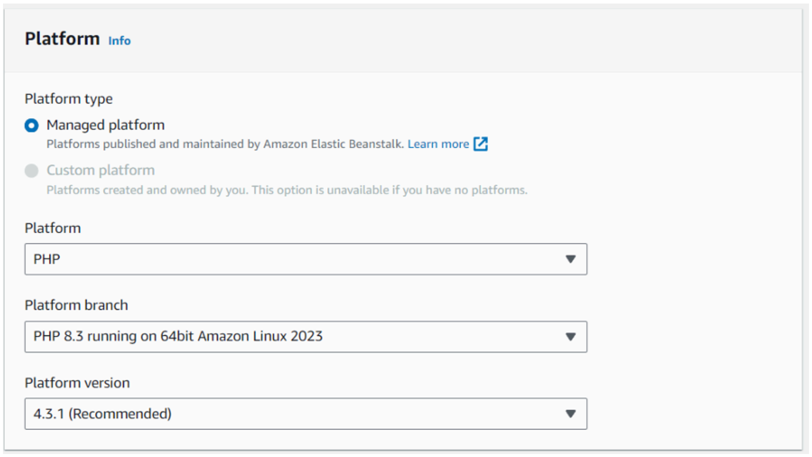

Step 1
Configure environment
Step 2
Configure service access
Step 3 - optional
Set up networking, database, and tags
Step 4 - optional
Configure instance traffic and scaling
Step 5 - optional
Configure updates, monitoring, and logging
Step 6
Review

Configure environment

Environment tier
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.
Worker environment
Run a worker application that processes long-running workloads on demand or performs tasks on a schedule.

Application information
Application name: MyFirstWebApp
Maximum length of 100 characters.
Application tags (optional)

Step 3:Under Platform, you can choose between Managed Platform and Custom Platform. Selected Managed Platform and configured it with PHP for ease of use and automatic updates.
Managed Platform: AWS manages and updates the environment, providing pre-configured platforms like PHP, making deployment easier and maintenance automated.
Custom Platform: Allows you to create and manage your own platform, giving full control over the environment configuration and software stack.



Platform Info

Platform type

[✓ Managed platform]
Platforms published and maintained by Amazon Elastic Beanstalk. Learn more

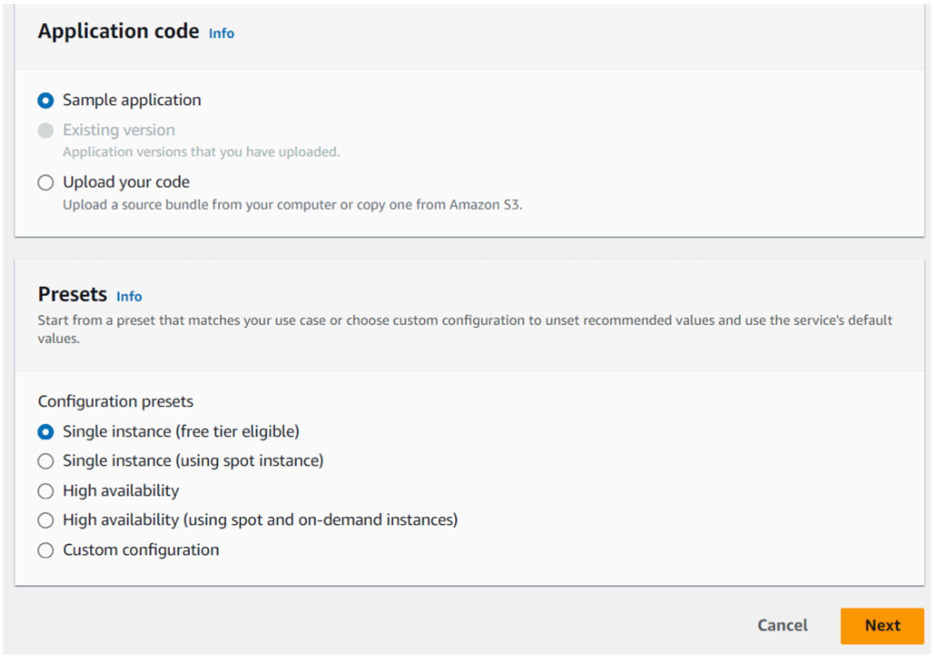
Custom platform
Platforms created and owned by you. This option is unavailable if you have no platforms.

Platform
PHP

Platform branch
PHP 8.3 running on 64bit Amazon Linux 2023

Platform version
4.3.1 (Recommended)

Step 4:**Presets** offer different deployment configurations to match your needs.
Single Instance (Free Tier Eligible) runs your application on one EC2 instance, suitable for development or low-traffic scenarios and eligible for the AWS Free Tier.
Single Instance deploys on one EC2 instance but without Free Tier limitations, ideal for small-scale projects.
High Availability uses multiple instances and a load balancer to ensure fault tolerance and reliability for production environments.
Using Spot Instances leverages unused EC2 capacity at reduced costs, suitable for applications that can handle interruptions and require cost savings.



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.

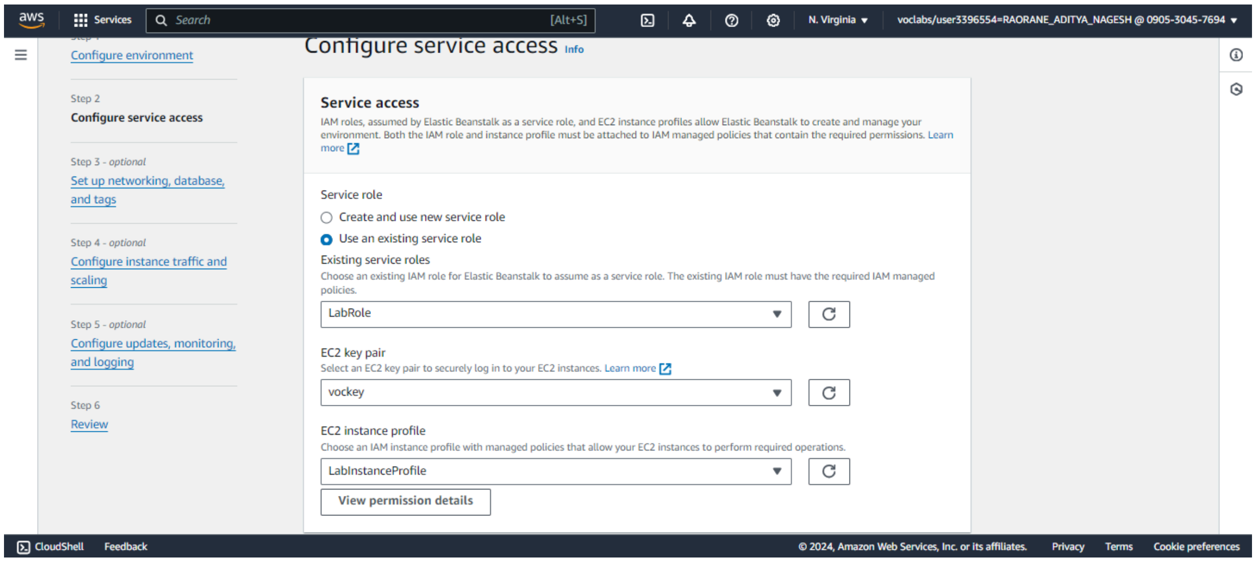
Presets Info

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 5:Define your application's service configuration, including settings for load balancing, auto-scaling, and health checks.Each EC2 instance can be accessed using its corresponding key pair.Choosing an EC2 key pair as a "vockey" (or a "virtual key") is typically done to enable secure access to your EC2 instances; allowing you to securely connect to your instances using SSH (Linux), ensuring that only authorized users can access the server.


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]

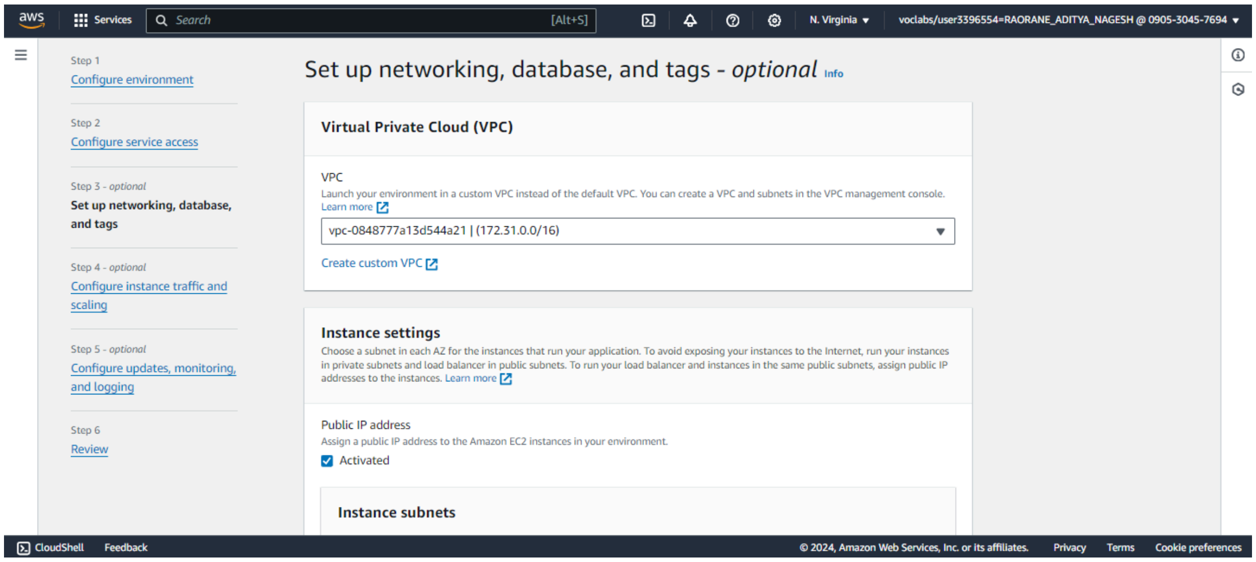
Existing service roles
Choose an existing IAM role for Elastic Beanstalk to assume as a service role. The existing IAM role must have the required IAM managed policies.
LabRole

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.
LabInstanceProfile

aws
Services
Search
[Alt+S]
N. Virginia
voclabs/user3396554=RAORANE_ADITYA_NAGESH @ 0905-3045-7694
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
View permission details
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Step 6: Configure VPC, subnets, and security groups for network access. Set up your environment's database if needed and apply tags for resource management.


Set up networking, database, and tags - optional

Virtual Private Cloud (VPC)

VPC
Launch your environment in a custom VPC instead of the default VPC. You can create a VPC and subnets in the VPC management console. Learn more
vpc-0848777a13d544a21 | (172.31.0.0/16)

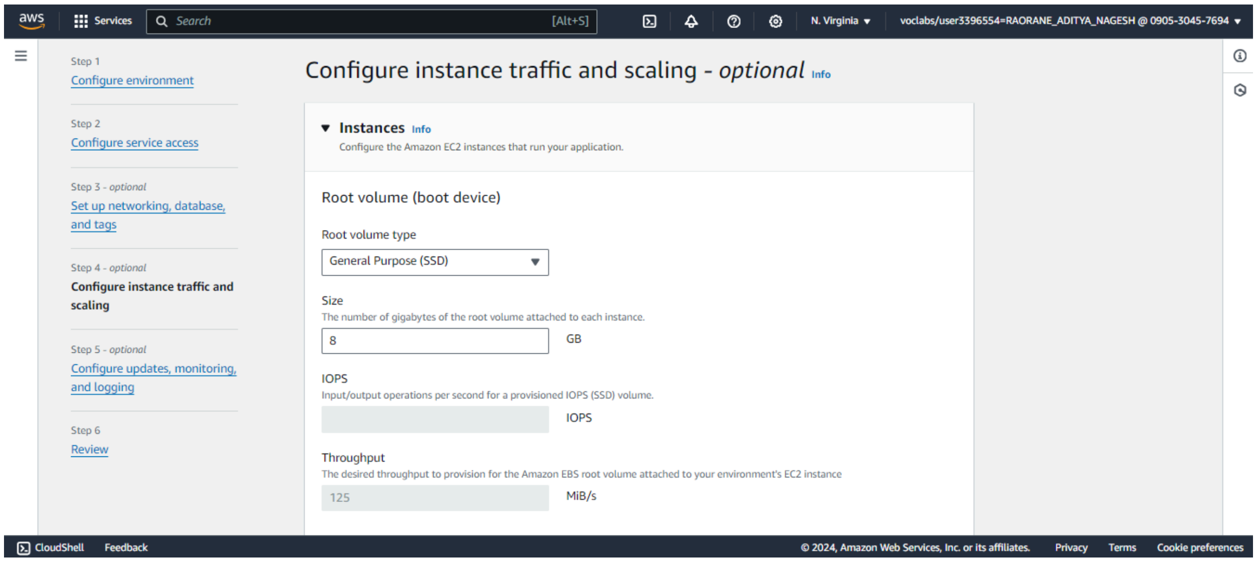
Instance settings
Choose a subnet in each AZ for the instances that run your application. To avoid exposing your instances to the Internet, run your instances in private subnets and load balancer in public subnets. To run your load balancer and instances in the same public subnets, assign public IP addresses to the instances. Learn more

Public IP address
Assign a public IP address to the Amazon EC2 instances in your environment.
[✓ Activated]

Instance subnets

aws
Services
Search
[Alt+S]
N. Virginia
voclabs/user3396554=RAORANE_ADITYA_NAGESH @ 0905-3045-7694
Step 1
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
Create custom VPC
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Step 7:Adjust load balancer settings to manage incoming traffic and configure auto-scaling rules to handle changes in traffic volume.The **root volume type** is the primary storage for the OS, and selecting **General Purpose (SSD)** offers balanced performance and durability for most applications.


aws
Services
Search [Alt+S]
N. Virginia
voclabs/user3396554=RAORANE_ADITYA_NAGESH @ 0905-3045-7694
Step 1
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
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Configure instance traffic and scaling - *optional* Info

Instances Info

Configure the Amazon EC2 instances that run your application.

Root volume (boot device)

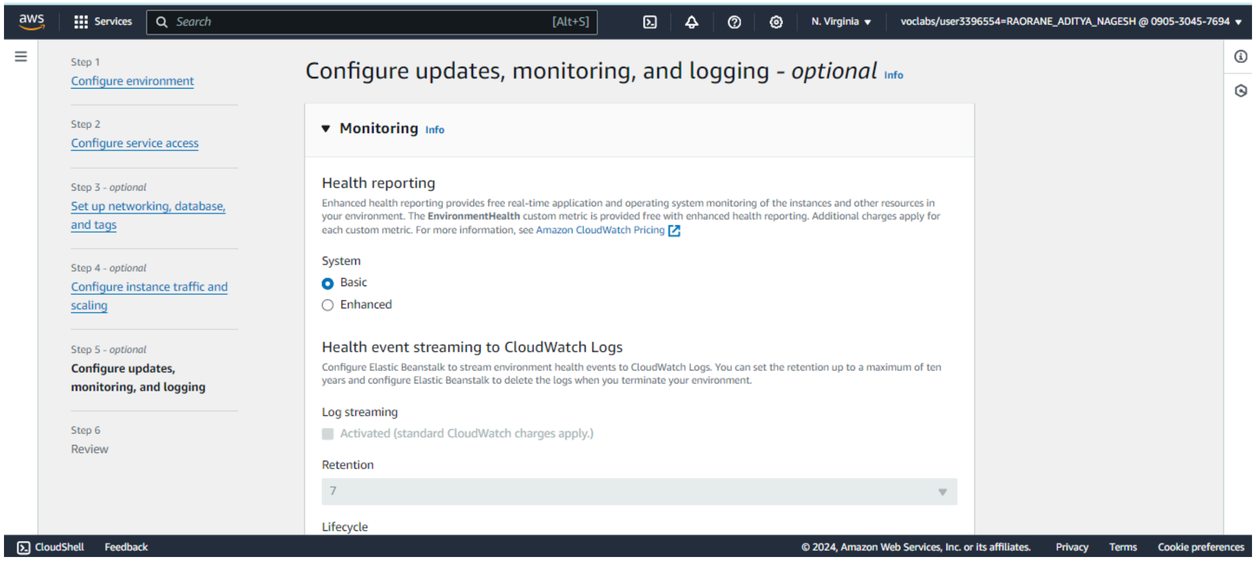
Root volume type: General Purpose (SSD)

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.
IOPS

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

Step 8:Set up deployment policies for updates, enable monitoring for performance metrics, and configure logging to capture and analyze application logs.


aws
Services
Search [Alt+S]
N. Virginia
voclabs/user3396554=RAORANE_ADITYA_NAGESH @ 0905-3045-7694
Step 1
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
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

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

Health event streaming to CloudWatch Logs

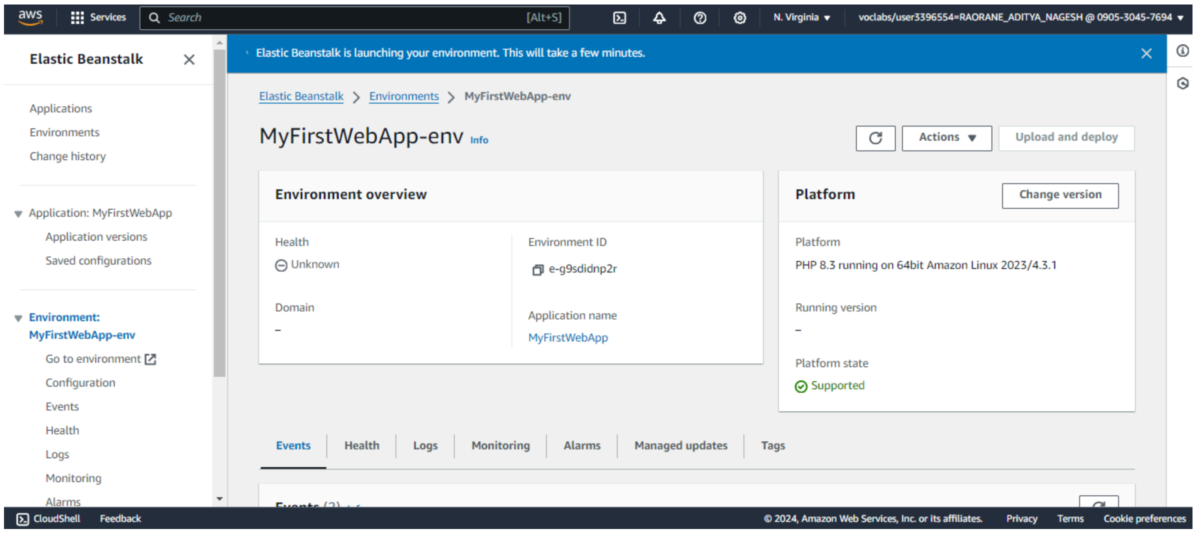
Configure Elastic Beanstalk to stream environment health events to CloudWatch Logs. You can set the retention up to a maximum of ten years and configure Elastic Beanstalk to delete the logs when you terminate your environment.

Log streaming
☐ Activated (standard CloudWatch charges apply.)

Retention
7

Lifecycle

Step 9:This creates our environment .We can now check our sample application uploaded if our environment is successfully deployed.



Elastic Beanstalk is launching your environment. This will take a few minutes.

Elastic Beanstalk > Environments > MyFirstWebApp-env

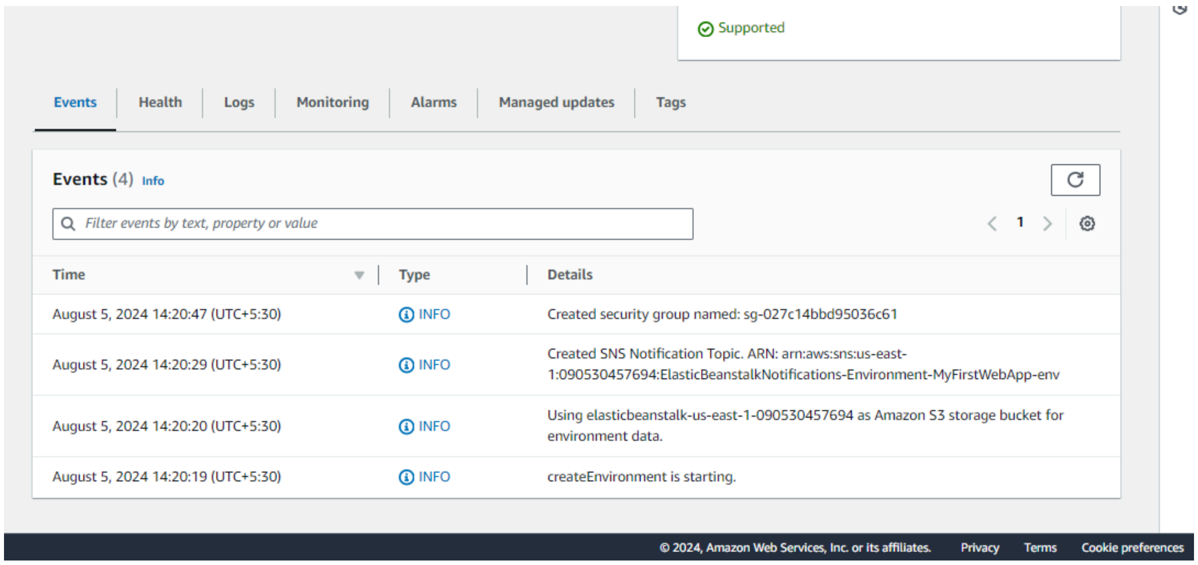
MyFirstWebApp-env

Environment overview

Health: Unknown
Environment ID: e-g9sdidnp2r
Domain: –
Application name: MyFirstWebApp

Platform

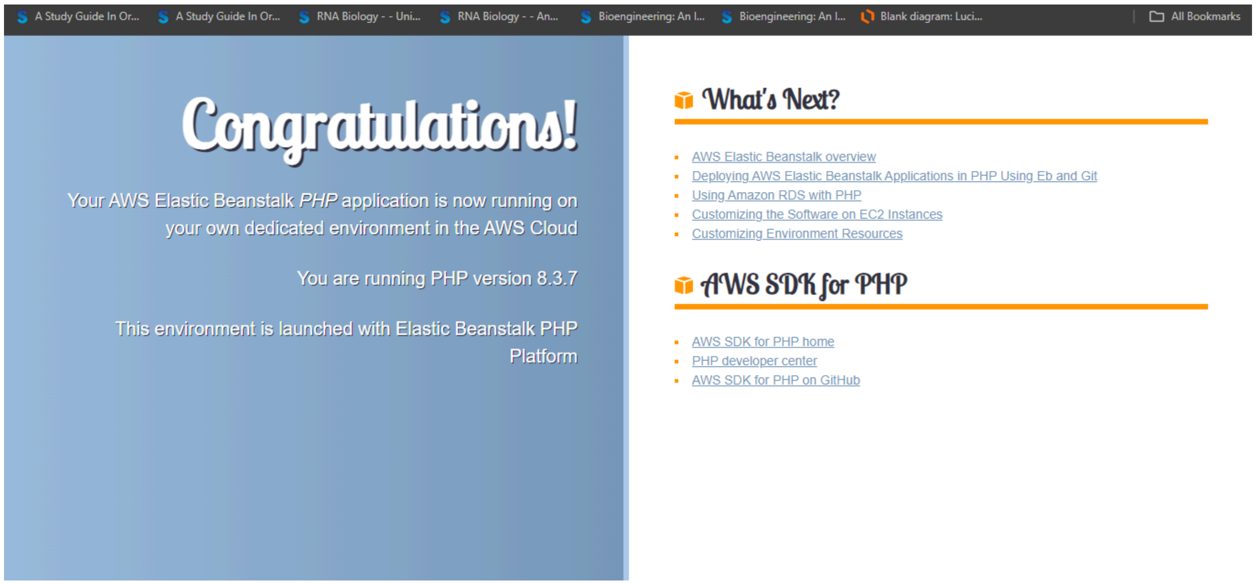
Platform: PHP 8.3 running on 64bit Amazon Linux 2023/4.3.1
Running version: –
Platform state: Supported



Events (4)

Time	Type	Details
August 5, 2024 14:20:47 (UTC+5:30)	INFO	Created security group named: sg-027c14bbd95036c61
August 5, 2024 14:20:29 (UTC+5:30)	INFO	Created SNS Notification Topic. ARN: arn:aws:sns:us-east-1:090530457694:ElasticBeanstalkNotifications-Environment-MyFirstWebApp-env
August 5, 2024 14:20:20 (UTC+5:30)	INFO	Using elasticbeanstalk-us-east-1-090530457694 as Amazon S3 storage bucket for environment data.
August 5, 2024 14:20:19 (UTC+5:30)	INFO	createEnvironment is starting.

Name: Aditya Nagesh Raorane Class: D15C[Batch B] Roll No: 44


A Study Guide In Or... A Study Guide In Or... RNA Biology - - Uni... RNA Biology - - An... Bioengineering: An I... Bioengineering: An I... Blank diagram: Luci... All Bookmarks

Congratulations!

Your AWS Elastic Beanstalk *PHP* application is now running on your own dedicated environment in the AWS Cloud

You are running PHP version 8.3.7

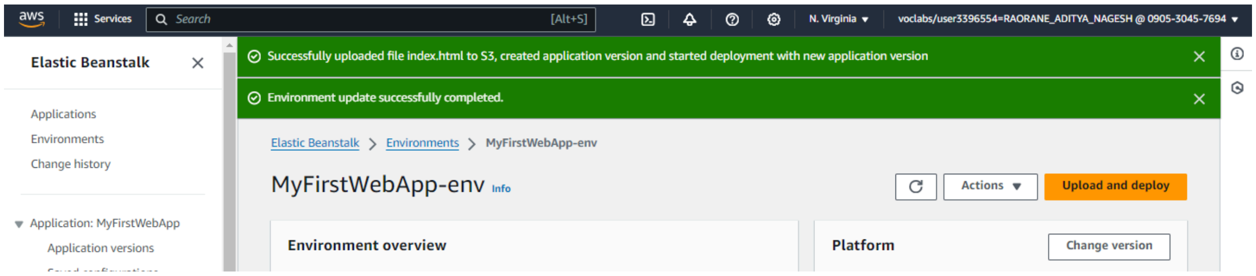
This environment is launched with Elastic Beanstalk PHP Platform

What's Next?

- AWS Elastic Beanstalk overview
- Deploying AWS Elastic Beanstalk Applications in PHP Using Eb and Git
- Using Amazon RDS with PHP
- Customizing the Software on EC2 Instances
- Customizing Environment Resources

AWS SDK for PHP

- AWS SDK for PHP home
- PHP developer center
- AWS SDK for PHP on GitHub

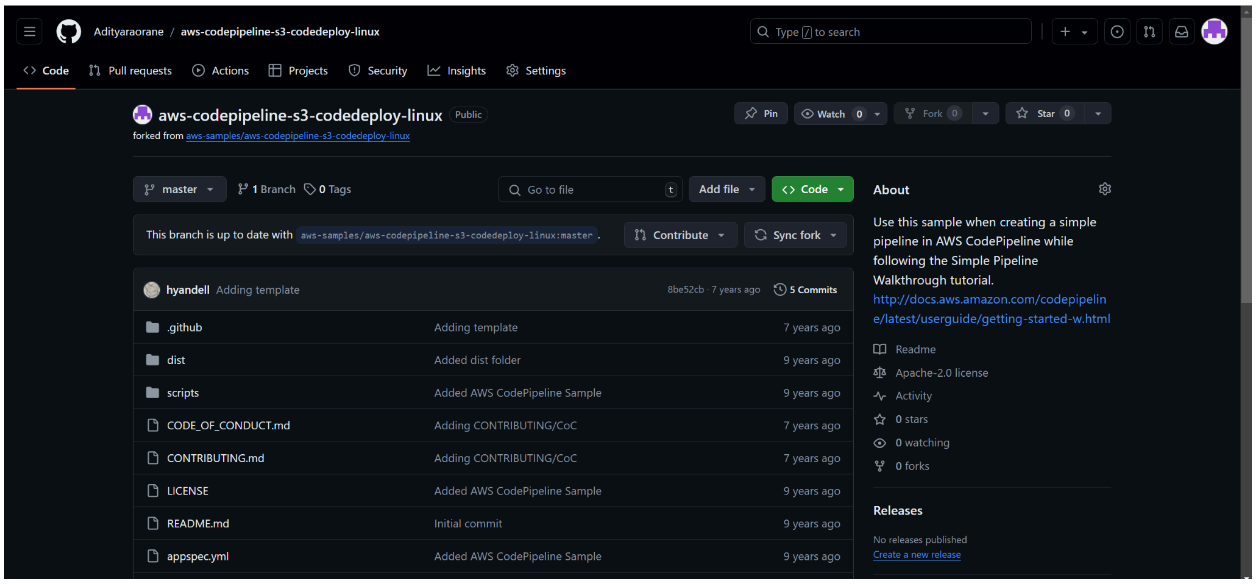

aws Services Search [Alt+S] N. Virginia voclabs/user3396554=RAORANE_ADITYA_NAGESH @ 0905-3045-7694
Elastic Beanstalk
Applications
Environments
Change history
Application: MyFirstWebApp
Application versions

Successfully uploaded file index.html to S3, created application version and started deployment with new application version

Environment update successfully completed.

Elastic Beanstalk > Environments > MyFirstWebApp-env
MyFirstWebApp-env Info
Actions
Upload and deploy
Environment overview
Platform
Change version

<u>Step 10:</u> Go to the github link below. This is a github with a sample code for deploying a file on AWSCodePipeline. Fork this repository into your personal github.
https://github.com/aws-samples/aws-codepipeline-s3-codedeploy-linux

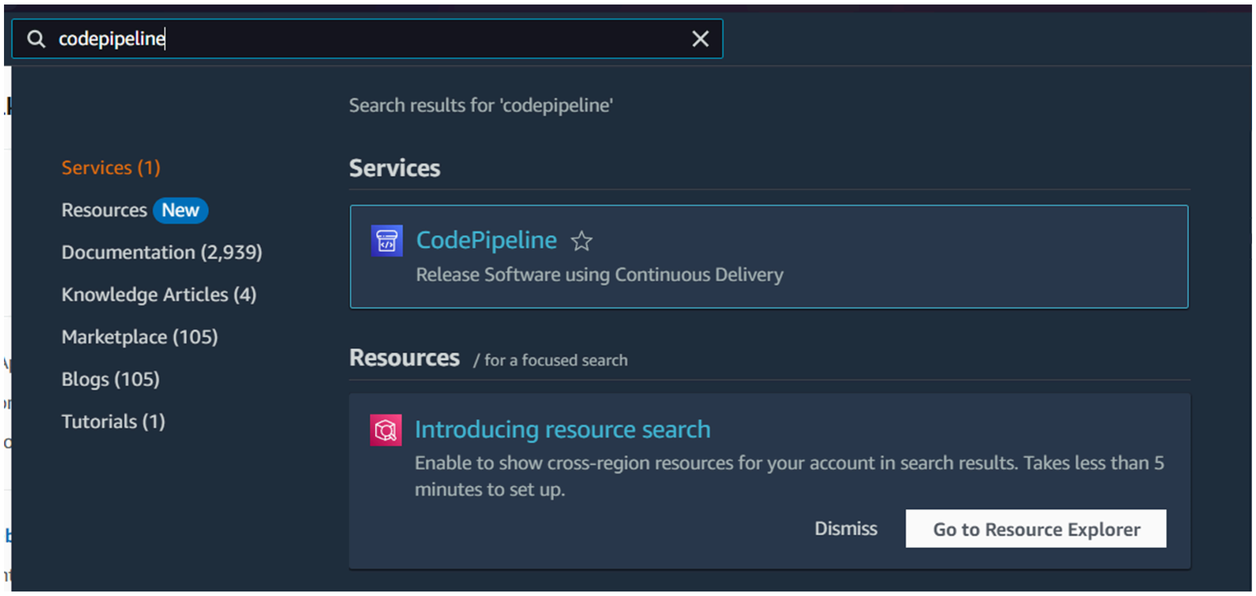

Adityaraorane / aws-codepipeline-s3-codedeploy-linux
Type / to search
Code Pull requests Actions Projects Security Insights Settings
aws-codepipeline-s3-codedeploy-linux Public
forked from aws-samples/aws-codepipeline-s3-codedeploy-linux
Pin Watch 0 Fork 0 Star 0
master 1 Branch 0 Tags Go to file Add file Code
This branch is up to date with aws-samples/aws-codepipeline-s3-codedeploy-linux:master .
Contribute Sync fork
hyandell Adding template 8be52cb · 7 years ago 5 Commits

.github	Adding template	7 years ago
dist	Added dist folder	9 years ago
scripts	Added AWS CodePipeline Sample	9 years ago
CODE_OF_CONDUCT.md	Adding CONTRIBUTING/CoC	7 years ago
CONTRIBUTING.md	Adding CONTRIBUTING/CoC	7 years ago
LICENSE	Added AWS CodePipeline Sample	9 years ago
README.md	Initial commit	9 years ago
appspec.yml	Added AWS CodePipeline Sample	9 years ago

About

Use this sample when creating a simple pipeline in AWS CodePipeline while following the Simple Pipeline Walkthrough tutorial.
http://docs.aws.amazon.com/codepipeline/latest/userguide/getting-started-w.html

Step 11:Search **CodePipeline** in the “services” tab and click on it.



codepipeline

Search results for 'codepipeline'

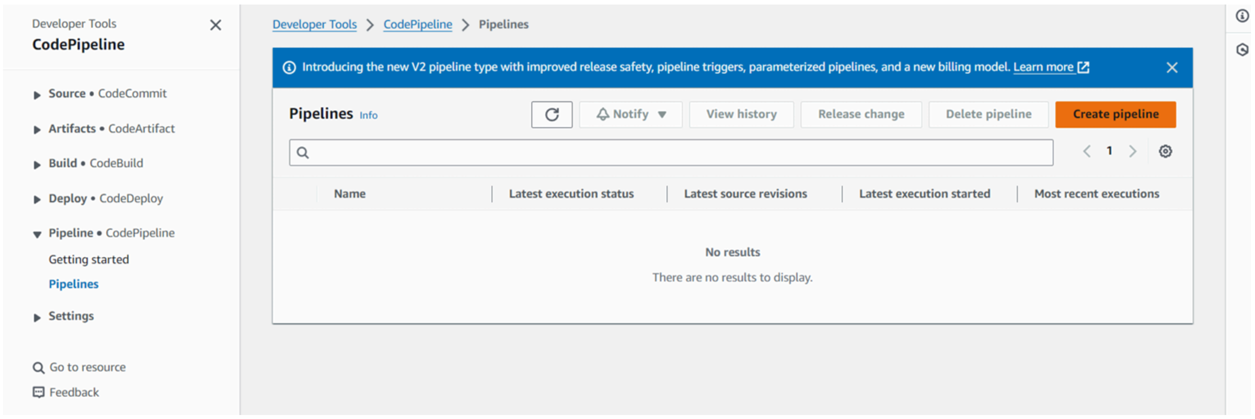
Services

CodePipeline
Release Software using Continuous Delivery

Resources / for a focused search

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

Step 12: After navigation to CodePipeline under “Developer Tools,” and click “Create pipeline” to start setting up your CI/CD.



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

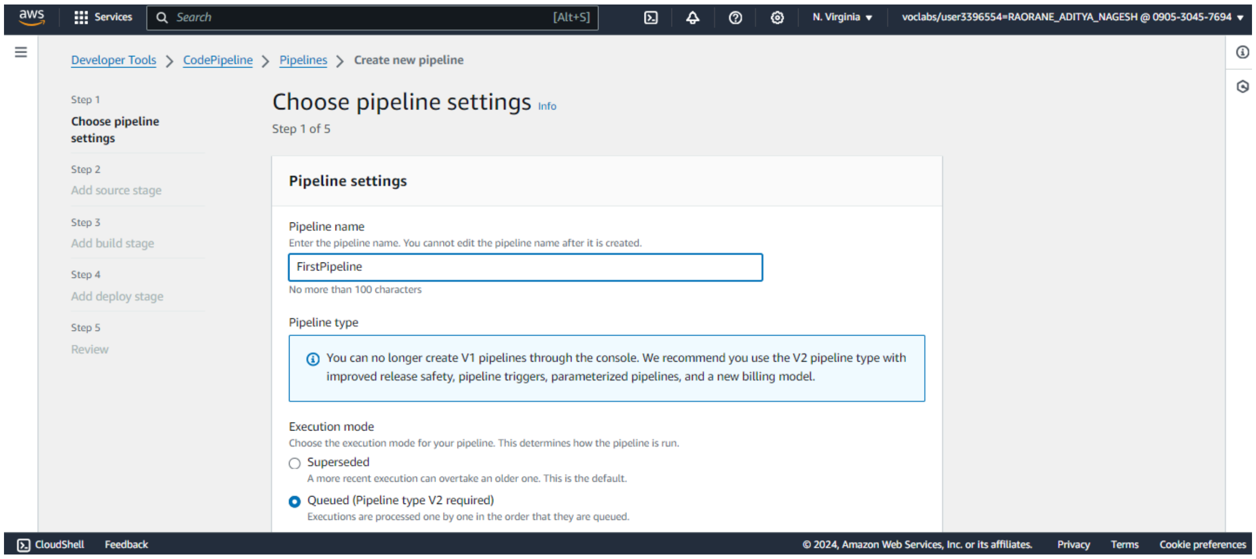
No results
There are no results to display.

Step 13: Give a name to your Pipeline. A new service role would be created with the name of the pipeline. The **pipeline type** options are:

- **Suspended**: Pauses the pipeline, stopping all execution of actions and deployments until resumed.
- **Queued**: Holds pipeline executions in a queue, executing them sequentially. This is useful for managing limited resources and preventing overload.
- **Parallel**: Runs multiple pipeline executions concurrently, allowing for faster processing but potentially higher resource usage.

Selecting **Queued** helps manage resource allocation by ensuring that pipelines run one at a time, avoiding conflicts and optimizing resource usage.Finally, click on “**Next**” at the bottom.

Name: Aditya Nagesh Raorane Class: D15C[Batch B] Roll No: 44



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

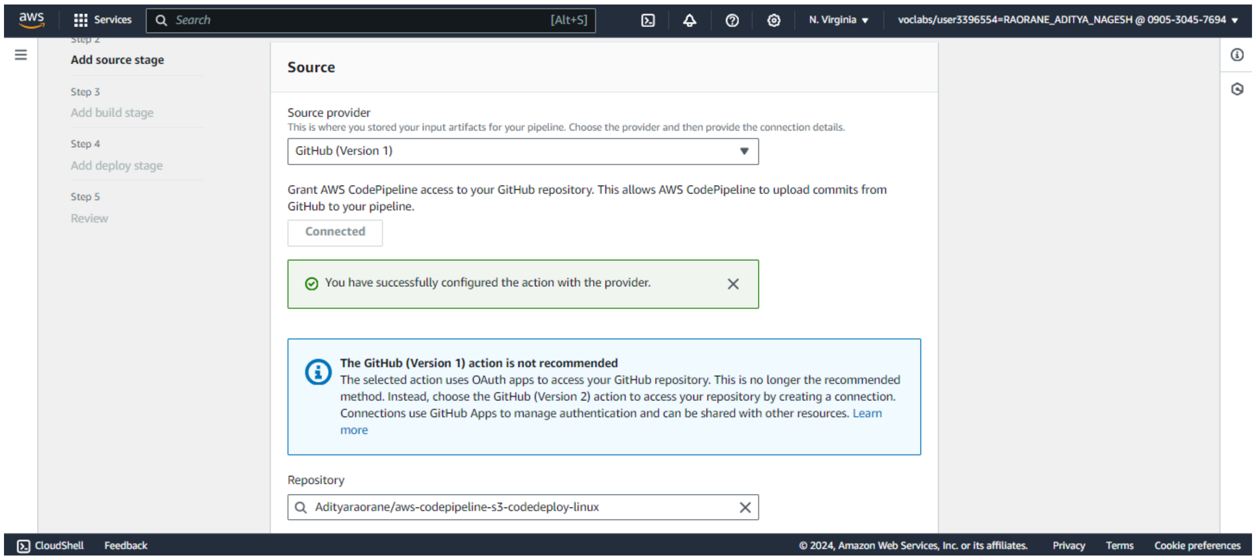
Pipeline settings

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

Step 14: In the Source stage of AWS CodePipeline, select GitHub(Version 1) as your source provider, then connect your GitHub account and choose the repository and branch you want to use for the pipeline(by default the branch is **master**). Then click on "**Next**".



Step 2 Add source stage
Step 3 Add build stage
Step 4 Add deploy stage
Step 5 Review

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 1)

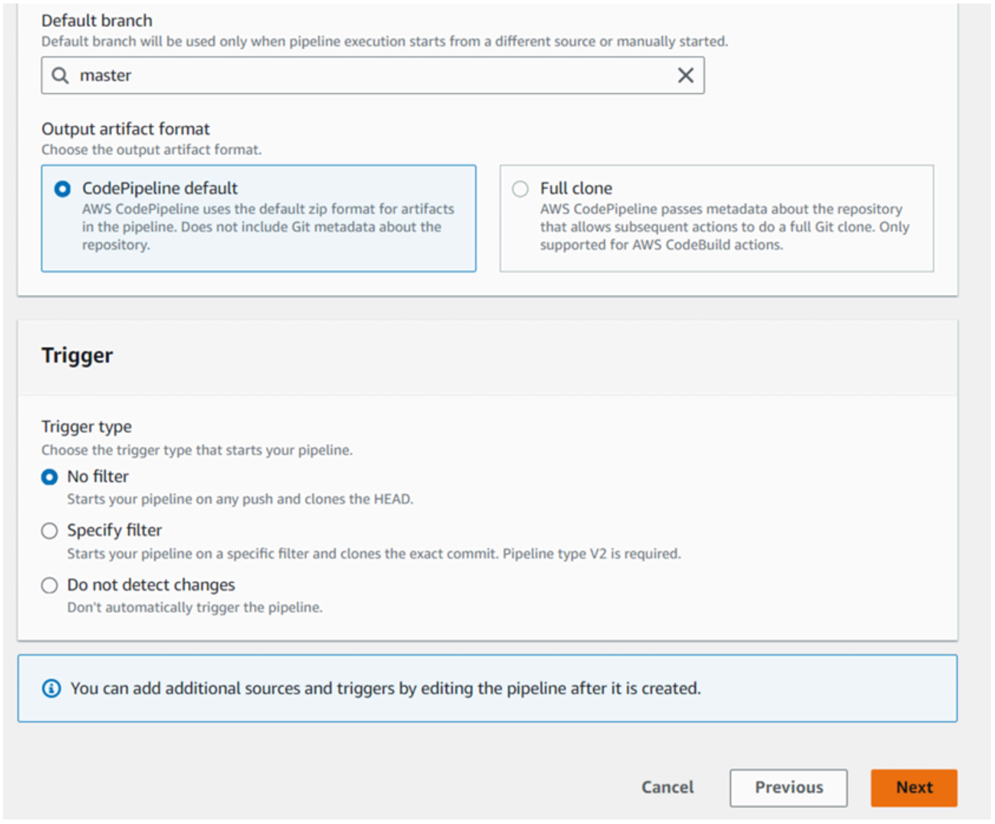
Grant AWS CodePipeline access to your GitHub repository. This allows AWS CodePipeline to upload commits from GitHub to your pipeline.
Connected

You have successfully configured the action with the provider.

The GitHub (Version 1) action is not recommended
The selected action uses OAuth apps to access your GitHub repository. This is no longer the recommended method. Instead, choose the GitHub (Version 2) action to access your repository by creating a connection. Connections use GitHub Apps to manage authentication and can be shared with other resources. Learn more

Repository
Adityaraorane/aws-codepipeline-s3-codedeploy-linux

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.


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

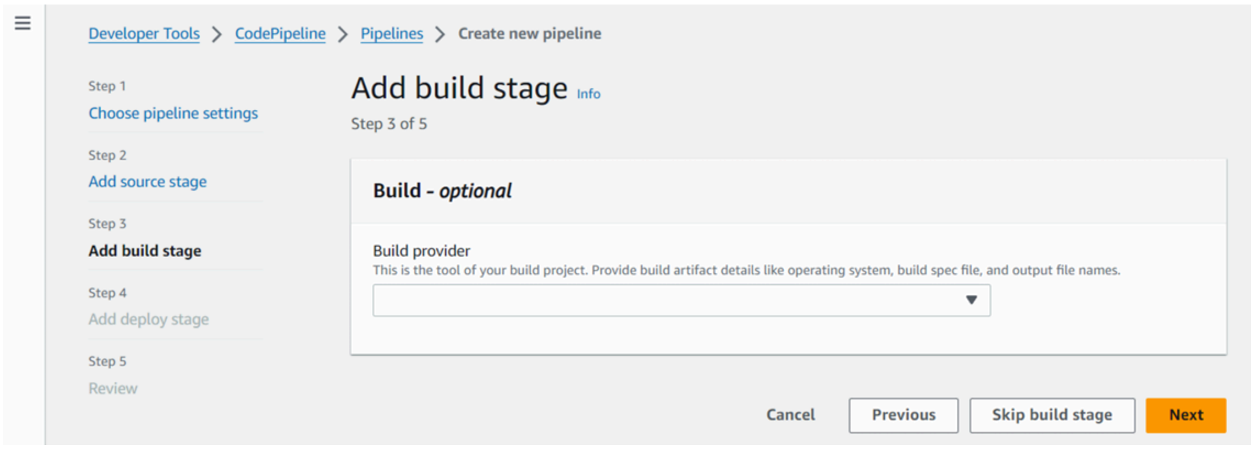
Output artifact format
Choose the output artifact format.
- [✓ CodePipeline default] AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include Git metadata about the repository.
- Full clone: AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full Git clone. Only supported for AWS CodeBuild actions.

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.



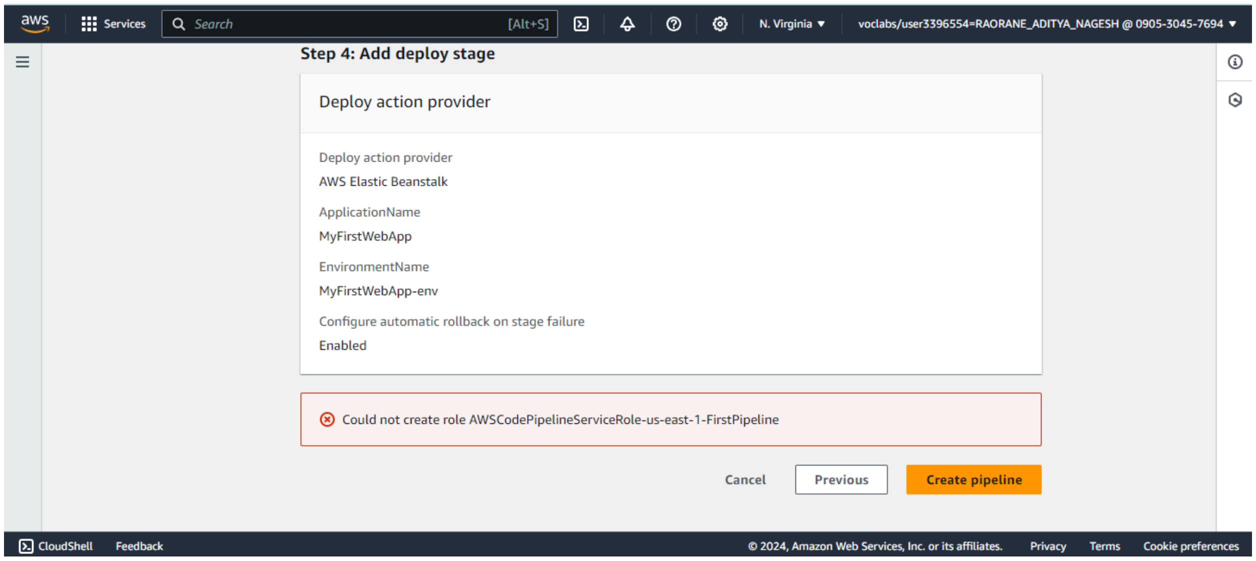
Add build stage Info
Step 3 of 5

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.

Step 17:The error occurred during the deployment stage because AWS Academy accounts often have **restricted IAM permissions**, preventing the creation of roles needed for the pipeline. This limitation can cause issues when configuring pipelines that require specific roles for execution. To resolve the error, deploy the pipeline using your personal AWS account, which typically has broader permissions and can create the necessary roles for successful deployment.

Name: Aditya Nagesh Raorane Class: D15C[Batch B] Roll No: 44


Services
Search [Alt+S]
N. Virginia
voclabs/user3396554=RAORANE_ADITYA_NAGESH @ 0905-3045-7694

Step 4: Add deploy stage

Deploy action provider

Deploy action provider
AWS Elastic Beanstalk

ApplicationName
MyFirstWebApp

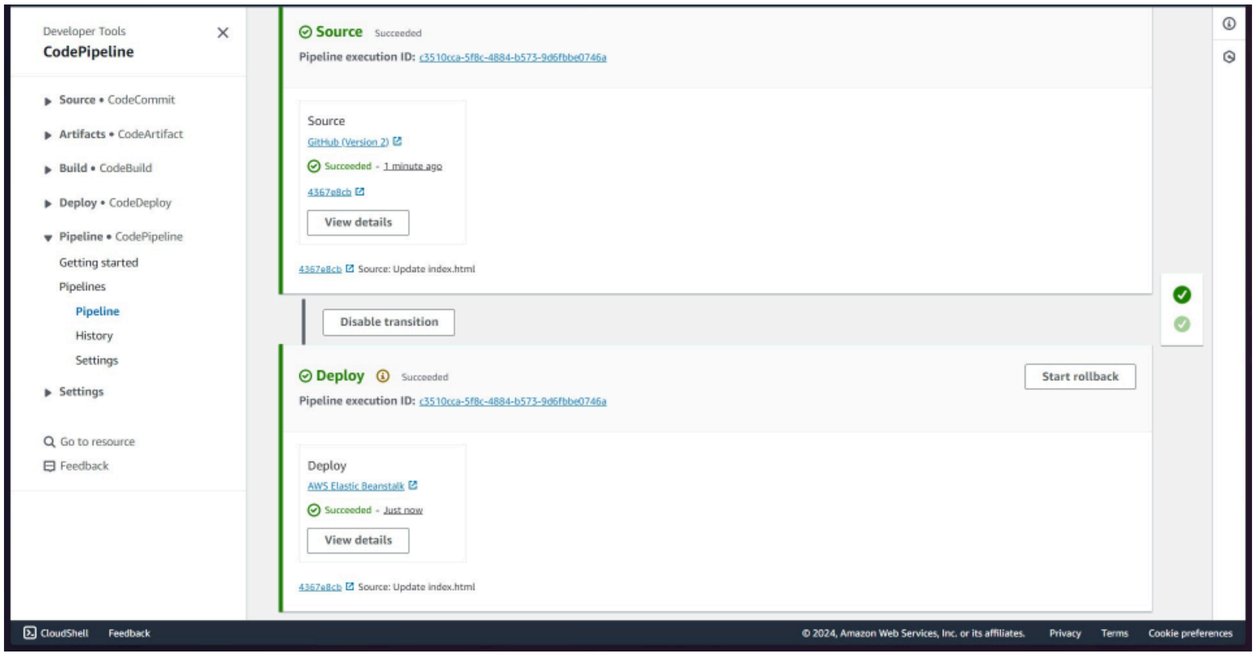
EnvironmentName
MyFirstWebApp-env

Configure automatic rollback on stage failure
Enabled

Could not create role AWSCodePipelineServiceRole-us-east-1-FirstPipeline

Cancel
Previous
Create pipeline
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

11]Once the source and deploy stages are successfully completed in AWS CodePipeline, you will see a message indicating "Review" to verify and confirm the pipeline setup before finalizing the deployment.



Source Succeeded
Pipeline execution ID: c3510cca-5f8c-4884-b573-9d6fbbe0746a

Source
GitHub (Version 2)
Succeeded - 1 minute ago
4367e8cb

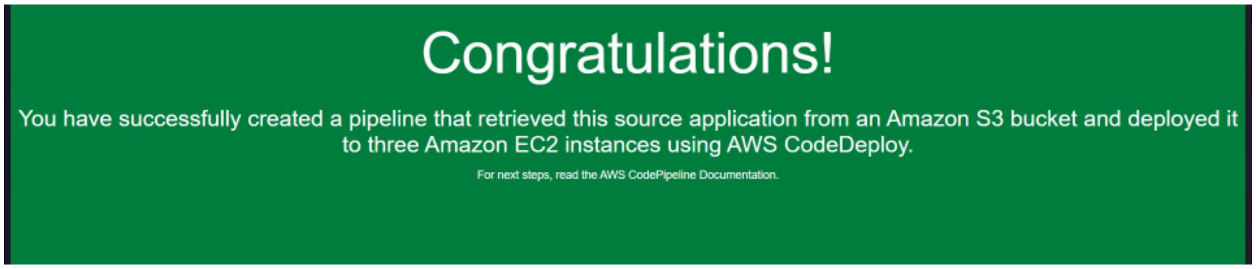
4367e8cb Source: Update index.html

Deploy Succeeded
Pipeline execution ID: c3510cca-5f8c-4884-b573-9d6fbbe0746a

Deploy
AWS Elastic Beanstalk
Succeeded - Just now

4367e8cb Source: Update index.html

12]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.

<u>Note</u>:We can make some changes to the index.html file in the github.Once the changes are committed, when the website is refreshed, the changes can be seen.