

**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.

### Elastic Beanstalk:

Amazon Elastic Beanstalk is a fully managed AWS service that streamlines the deployment and scaling of web applications. Developers simply upload their code, and Elastic Beanstalk automatically handles tasks like resource provisioning, load balancing, auto-scaling, and application health monitoring. It supports multiple programming languages and frameworks, offering flexibility and control over the underlying AWS resources if needed. This enables rapid application deployment without the complexities of managing infrastructure.

### CodePipeline:

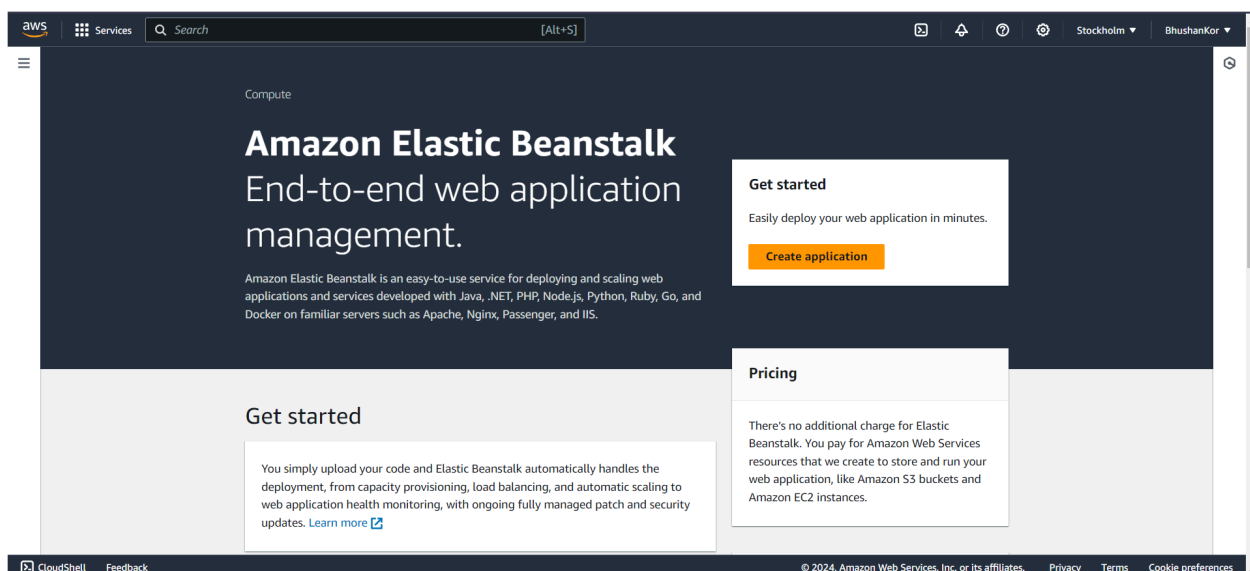
A CodePipeline is an AWS service that automates the build, test, and deploy phases of your release process. It enables continuous integration and continuous delivery (CI/CD) by defining the workflow of code changes from source to production. CodePipeline integrates with various AWS services and third-party tools to streamline and accelerate the release of new features.

**Prerequisites:** Before you start go to the end of this document and do the steps of New learning because it is necessary for step 5 and codePipeline.

## A) Elastic Beanstalk

**Step 1:** Open Your **personal AWS** Account Because Elastic Beanstalk can be created in AWS Academy but connecting GitHub with the code pipeline requires an IAM policy that is not present in the AWS Academy account.

In services search for Elastic Beanstalk and click on it.



Name: Bhushan Mukund Kor

Academic Year: 2024-2025

Division: D15C

Roll No: 28

**Step 2:** Click on Create a new application and give a name to your application.

aws Services Search [Alt+S] N. Virginia BhushanKor

Bhushan\_App\_3 application is being deleted (2)

Elastic Beanstalk > Create application

### Create new application [Info](#)

**Application information**

Application name  
Bhushan\_App\_4  
Maximum length of 100 characters.

Description

**Tags**

Apply up to 50 tags. You can use tags to group and filter your resources. A tag is a key-value pair. The key must be unique within the resource and is case-sensitive. [Learn more](#)

No tags associated with the resource.

[Add new tag](#)  
You can add 50 more tags.

[Cancel](#) [Create](#)

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

**Step 3:** After creation of the application click on Create Environment and select Web Server Environment.

aws Services Search [Alt+S] N. Virginia BhushanKor

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 [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  
Bhushan\_App\_4  
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  
BhushanApp4-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

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

Name: Bhushan Mukund Kor

Academic Year: 2024-2025

Division: D15C

Roll No: 28

**Step 4:** Select the platform as PHP and application code to sample or if you have you can upload and keep others to default.

This screenshot shows the 'Create new environment' page in the AWS Management Console. The 'Environment name' is 'BhushanApp4-env'. The 'Domain' is '.us-east-1.elasticbeanstalk.com'. The 'Platform' section is expanded, showing 'Managed platform' selected. The 'Platform' dropdown is set to 'PHP', the 'Platform branch' is 'PHP 8.3 running on 64bit Amazon Linux 2023', and the 'Platform version' is '4.3.1 (Recommended)'.

Environment name  
BhushanApp4-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

Platform version  
4.3.1 (Recommended)

This screenshot shows the 'Create new environment' page in the AWS Management Console, continuing from the previous step. The 'Application code' section is expanded, showing 'Sample application' selected. The 'Presets' section is also expanded, showing 'Single instance (free tier eligible)' selected. The 'Next' button is visible at the bottom right.

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

Cancel Next

Name: Bhushan Mukund Kor

Academic Year: 2024-2025

Division: D15C

Roll No: 28

**Step 5:** Select the use an existing service role option because create and use new service role might not work according to the new policy of AWS (Refer to the New learnings of this document).

Select the existing roles and instances and keys.

The screenshot shows the 'Configure service access' step in the AWS Elastic Beanstalk console. The left sidebar lists the steps: Step 1: Configure environment, Step 2: Configure service access (active), 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, and Step 6: Review. The main content area is titled 'Configure service access' and includes a 'Service access' section with two radio buttons: 'Create and use new service role' (unselected) and 'Use an existing service role' (selected). Below this, the 'Existing service roles' section shows a dropdown menu with 'Bhushan\_EC2\_new' selected. The 'EC2 key pair' section shows a dropdown menu with 'Bhushan\_EC2\_key' selected. The 'EC2 instance profile' section shows a dropdown menu with 'Bhushan\_EC2\_new' selected. At the bottom, there are buttons for 'Cancel', 'Skip to review', 'Previous', and 'Next'.

**Step 6:** Review and click on submit.

The screenshot shows the 'Review' step in the AWS Elastic Beanstalk console. The left sidebar lists the steps: Step 1: Configure environment, Step 2: Configure service access (active), 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, and Step 6: Review (active). The main content area is titled 'Review' and includes a 'Review' section with two sub-sections: 'Step 1: Configure environment' and 'Step 2: Configure service access'. The 'Step 1: Configure environment' section shows 'Environment tier' as 'Web server environment', 'Application name' as 'Bhushan\_App\_4', 'Environment name' as 'BhushanApp4-env', 'Application code' as 'Sample application', and 'Platform' as 'amazon-elasticbeanstalk-us-east-1:platform/PHP 8.3 running on 64bit Amazon Linux 2023/4.3.1'. The 'Step 2: Configure service access' section shows 'Service role' as 'arn:aws:iam::010928205712:role/Bhushan\_EC2\_new', 'EC2 key pair' as 'Bhushan\_EC2\_key', and 'EC2 instance profile' as 'Bhushan\_EC2\_new'. At the bottom, there are buttons for 'Edit' and 'Next'.

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

Tags

Key	Value
No tags	
There are no tags defined	

Step 4: Configure instance traffic and scaling

Instance traffic and scaling

Customize the capacity and scaling for your environment's instances. Select security groups to control instance traffic. Configure the software that runs on your environment's instances by setting platform-specific options.

Instances

IMDSv1  
Deactivated

Capacity

Step 4: Configure instance traffic and scaling

Instance traffic and scaling

Customize the capacity and scaling for your environment's instances. Select security groups to control instance traffic. Configure the software that runs on your environment's instances by setting platform-specific options.

Instances

IMDSv1  
Deactivated

Capacity

Environment type	Fleet composition	On-demand base
Single instance	On-Demand instance	0
On-demand above base	Capacity rebalancing	Scaling cooldown
0	Deactivated	360
Processor type	Instance types	AMI ID
x86_64	t3.micro,t3.small	ami-083f545ce1a73bf03
Availability Zones	Metric	Statistic
Any	NetworkOut	Average
Unit	Period	Breach duration
Bytes	5	5
Upper threshold	Scale up increment	Lower threshold
6000000	1	2000000

Step 5: Configure updates, monitoring, and logging

Updates, monitoring, and logging

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	Retention	Lifecycle
Deactivated	7	false

Updates

Managed updates	Deployment batch size	Deployment batch size type
Activated	100	Percentage
Command timeout	Deployment policy	Health threshold
600	AllAtOnce	Ok

Name: Bhushan Mukund Kor

Academic Year: 2024-2025

Division: D15C

Roll No: 28

600

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 defined

There are no environment properties defined

Cancel Previous Submit

**Step 7:** Done Elastic BeanStalk environment is created successfully, click on the link to view the preview of your sample code or uploaded code.

Elastic Beanstalk

Applications

Environments

Change history

Application: Bhushan\_App\_4

Application versions

Saved configurations

Environment: BhushanApp4-env

Go to environment

Configuration

Events

Health

Logs

Monitoring

Alarms

Managed updates

Tags

Recent environments

BhushanApp4-env

BhushanApp2-env

Environment successfully launched.

Elastic Beanstalk > Environments > BhushanApp4-env

BhushanApp4-env Info

Actions Upload and deploy

Environment overview

Health Warning

Environment ID e-zaagrygpc

Domain BhushanApp4-env.eba-2mccvqpx.us-east-1.elasticbeanstalk.com

Application name Bhushan\_App\_4

Platform

Platform PHP 8.3 running on 64bit Amazon Linux 2023/4.3.1

Running version -

Platform state Supported


Events (11) Info

Filter events by text, property or value

Time	Type	Details
August 11, 2024 21:11:41 (UTC+5:30)	INFO	Successfully launched environment: BhushanApp4-env
August 11, 2024 21:11:38 (UTC+5:30)	INFO	Added instance [i-049e1dd5eb5428d26] to your environment.
August 11, 2024 21:11:38 (UTC+5:30)	WARN	Environment health has transitioned from Pending to Warning. Initialization completed 16 seconds ago and took 2 minutes. Unable to assume role "arn:aws:iam:010928205712:role/Bhushan_EC2_new". Verify that the role exists and is configured correctly.

**Academic Year:2024-2025**

**Roll No: 28**



← → ↺ ⌂ Not secure bhusanapp4-env.eba-2mcvqgpx.us-east-1.elasticbeanstalk.com

Apps Explore UNESCO W... My Courses 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](#)

Name: Bhushan Mukund Kor

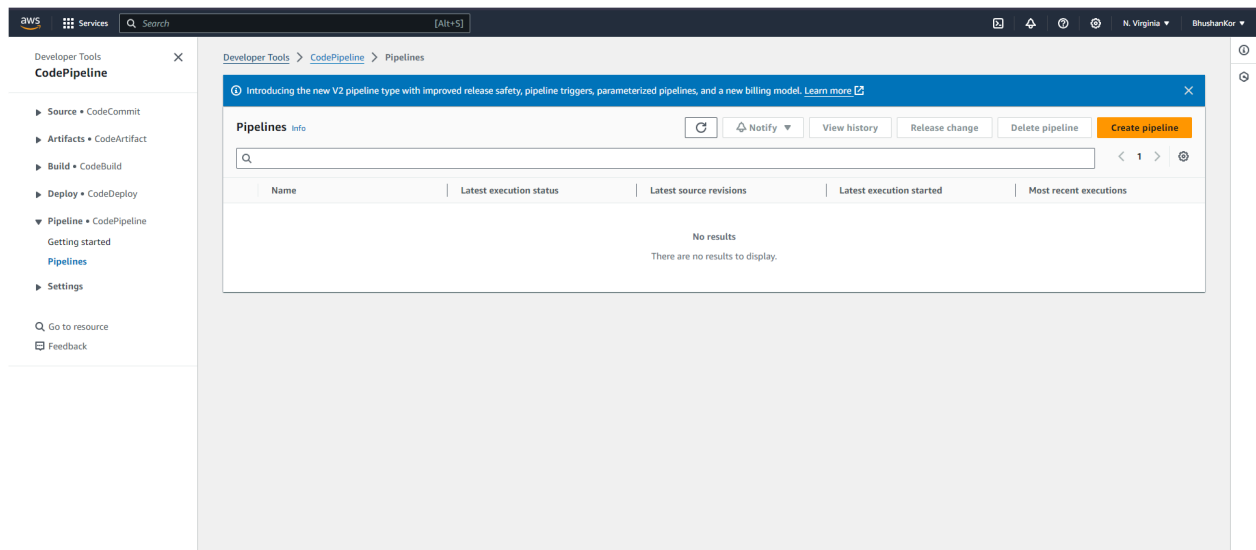
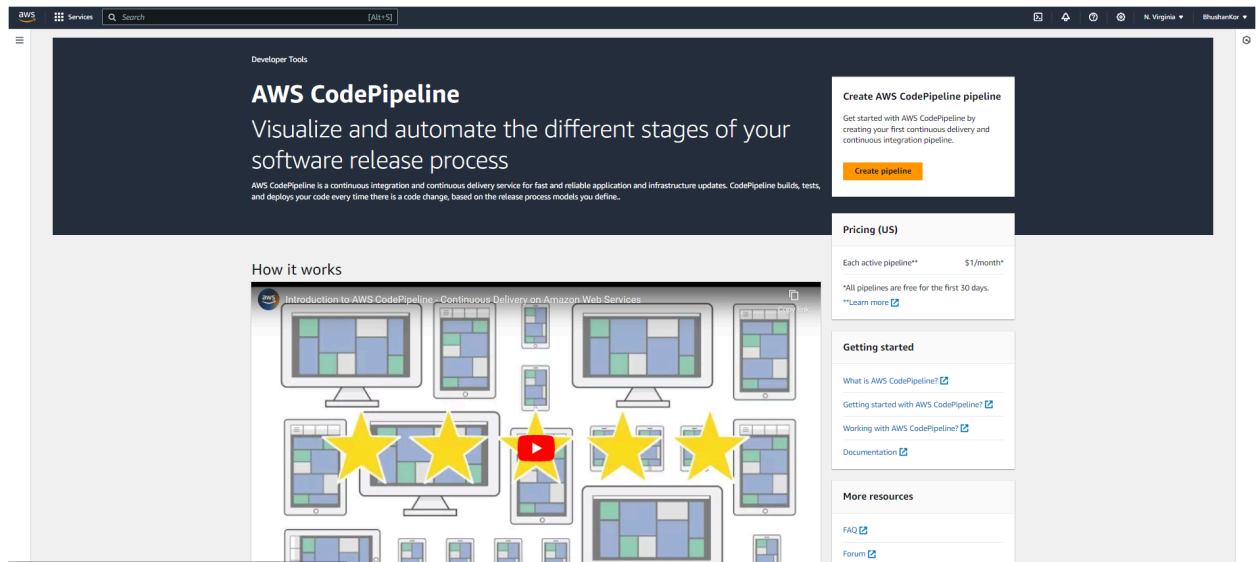
Academic Year: 2024-2025

Division: D15C

Roll No: 28

## B) CodePipeline

Step 1: Search for codePipeline in services and click on create pipeline.





Name: Bhushan Mukund Kor

Academic Year: 2024-2025

Division: D15C

Roll No: 28

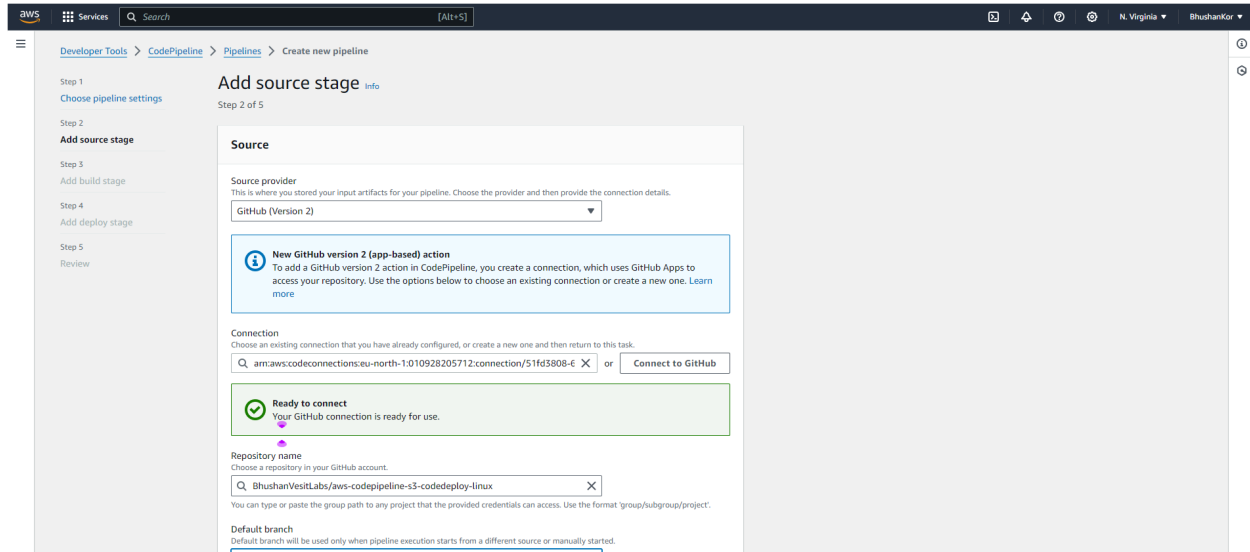
**Step 2:** Give the name to the pipeline, select a new service role, and keep the rest all to default.

The screenshot shows the 'Choose pipeline settings' page in the AWS CodePipeline console. The page is titled 'Choose pipeline settings' and is Step 1 of 5. The left sidebar shows the steps: Step 1 (Choose pipeline settings), Step 2 (Add source stage), Step 3 (Add build stage), Step 4 (Add deploy stage), and Step 5 (Review). The main content area is divided into sections: Pipeline settings, Pipeline type, Execution mode, and Service role. In the Pipeline settings section, the Pipeline name is 'Bhushan\_PipeLines'. In the Pipeline type section, the 'Queued (Pipeline type V2 required)' option is selected. In the Execution mode section, the 'Queued (Pipeline type V2 required)' option is selected. In the Service role section, the 'New service role' option is selected, and the Role name is 'AWSCodePipelineServiceRole-us-east-1-Bhushan\_PipeLines'. There is a checkbox 'Allow AWS CodePipeline to create a service role so it can be used with this new pipeline' which is checked.

The screenshot shows the 'Variables' and 'Advanced settings' page in the AWS CodePipeline console. The page is titled 'Variables' and is Step 2 of 5. The left sidebar shows the steps: Step 1 (Choose pipeline settings), Step 2 (Add source stage), Step 3 (Add build stage), Step 4 (Add deploy stage), and Step 5 (Review). The main content area is divided into sections: Service role, Role name, Variables, and Advanced settings. In the Service role section, the 'New service role' option is selected, and the Role name is 'AWSCodePipelineServiceRole-us-east-1-Bhushan\_PipeLine'. There is a checkbox 'Allow AWS CodePipeline to create a service role so it can be used with this new pipeline' which is checked. In the Variables section, there is a button 'Add variable' and a note 'The first pipeline execution will fail if variables have no default values.' In the Advanced settings section, there is a button 'Advanced settings'.

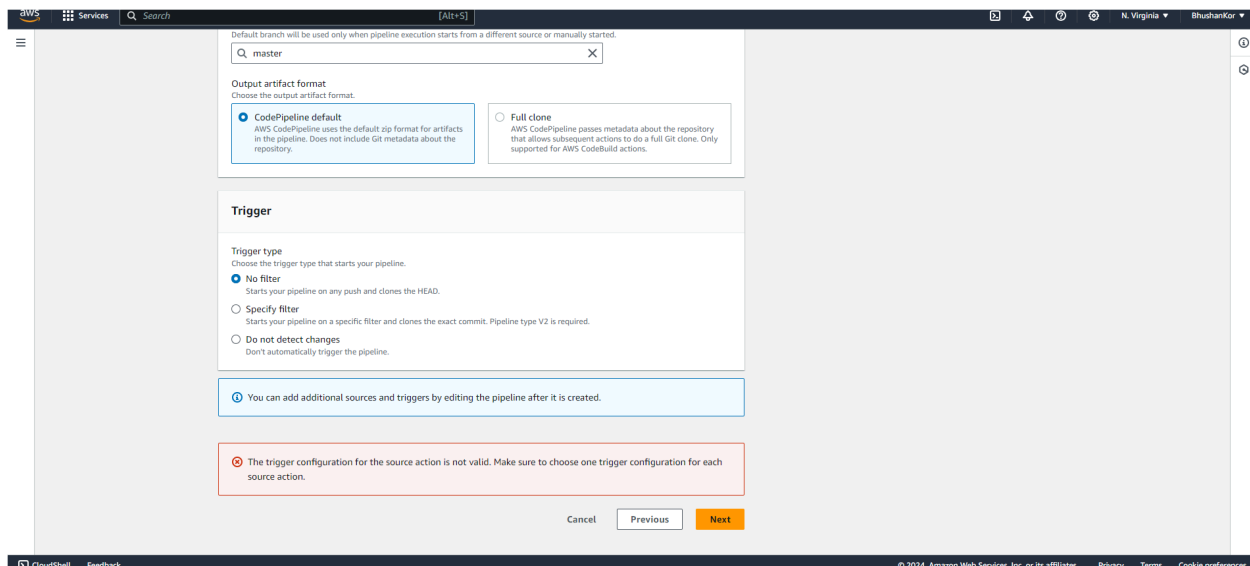
Before Step 3 Fork or clone the below repository to your GitHub account.  
<https://github.com/aws-samples/aws-codepipeline-s3-codedeploy-linux.git>

**Step 3:** Now in the Add source stage Select GitHub (Version 2) [This will work only in personal AWS academy], then connect your GitHub account and select repository name, branch to main/master.



The screenshot shows the AWS CodePipeline console interface. The left sidebar lists the steps: Step 1 (Choose pipeline settings), Step 2 (Add source stage), Step 3 (Add build stage), Step 4 (Add deploy stage), and Step 5 (Review). The main panel is titled 'Add source stage' and shows the configuration for a new source stage. The 'Source provider' is set to 'GitHub (Version 2)'. A message indicates that a new GitHub version 2 (app-based) action is being added. The 'Connection' section shows a dropdown with 'am:aws:codeconnections:eu-north-1:010928205712:connection/51fd3808-6' and a 'Connect to GitHub' button. The 'Repository name' is set to 'BhushanVesitLabs/aws-codepipeline-s3-codedeploy-linux'. The 'Default branch' is set to 'main'.

**Step 4:** Select Trigger to No filter otherwise it will give an error.



The screenshot shows the 'Trigger' configuration section of the AWS CodePipeline console. The 'Trigger type' is set to 'No filter'. The 'Output artifact format' is set to 'CodePipeline default'. A message indicates that the trigger configuration for the source action is not valid. The 'Next' button is highlighted in orange.

Name: Bhushan Mukund Kor

Academic Year: 2024-2025

Division: D15C

Roll No: 28

**Step 5:** In the Depoly stage select AWS ElasticBeanstalk as the deploy provider, Select a region to US East (N.Virginia) or any with N.Virginia. Input artifacts to default and the Application name and environment name of your Elastic Beanstalk.

The screenshot shows the AWS CodePipeline console interface. The left sidebar indicates the current step is 'Add deploy stage' (Step 4 of 5). The main panel is titled 'Add deploy stage' and contains a warning message: '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.' Below the warning, the 'Deploy' section is active, showing the following configuration:

- Deploy provider:** AWS Elastic Beanstalk
- Region:** US East (N. Virginia)
- Input artifacts:** SourceArtifact
- Application name:** Bhushan\_App\_4
- Environment name:** BhushanApp4-env

The bottom of the console shows the footer with '© 2024, Amazon Web Services, Inc. or its affiliates.' and links for 'Privacy', 'Terms', and 'Cookie preferences'.

This screenshot shows the same AWS CodePipeline console interface as the previous one, but with the 'Configure automatic rollback on stage failure' checkbox checked. The configuration details are identical:

- Deploy provider:** AWS Elastic Beanstalk
- Region:** US East (N. Virginia)
- Input artifacts:** SourceArtifact
- Application name:** Bhushan\_App\_4
- Environment name:** BhushanApp4-env
- Configure automatic rollback on stage failure:** ☒

At the bottom of the configuration panel, there are three buttons: 'Cancel', 'Previous', and 'Next'.

Name: **Bhushan Mukund Kor**

Academic Year: **2024-2025**

Division: **D15C**

Roll No: **28**

**Step 6:** Review and Click on Create Pipeline.

The screenshot shows the AWS CodePipeline console in the 'Review' step of creating a new pipeline. The left sidebar lists the steps: Step 1: Choose pipeline settings, Step 2: Add source stage, Step 3: Add build stage, Step 4: Add deploy stage, and Step 5: Review. The main content area is titled 'Review' and 'Step 1: Choose pipeline settings'. It displays the following pipeline settings:

- Pipeline name: Bhushan\_PipeLines
- Pipeline type: V2
- Execution mode: QUEUED
- Artifact location: codepipeline-us-east-1-934567252759
- Service role name: AWSCodePipelineServiceRole-us-east-1-Bhushan\_PipeLines

Below the settings, there is a 'Variables' section with a table:

Name	Default value	Description
No variables		

A note at the bottom of the variables section states: 'No variables defined at the pipeline level in this pipeline.'

The screenshot shows the AWS CodePipeline console in the 'Add source stage' step. The left sidebar lists the steps: Step 1: Choose pipeline settings, Step 2: Add source stage, Step 3: Add build stage, Step 4: Add deploy stage, and Step 5: Review. The main content area is titled 'Step 2: Add source stage'. It displays the following configuration:

- Source action provider: GitHub (Version 2)
- OutputArtifactFormat: CODE\_ZIP
- DetectChanges: true
- ConnectionArn: arn:aws:codeconnections:eu-north-1:010928205712:connection/51fd3808-6436-478b-b88b-030b391fb258
- FullRepositoryId: BhushanVestLabs/aws-codepipeline-s3-codedeploy-linux
- Default branch: master

Below the configuration, there is a 'Trigger configuration' section with the following details:

- Trigger type: No filter

Below the trigger configuration, there is a 'Step 3: Add build stage' section with the following details:

- Build action provider: [Not specified]

Name: Bhushan Mukund Kor

Division: D15C

Academic Year: 2024-2025

Roll No: 28

The screenshot shows the 'Create pipeline' wizard in the AWS CodePipeline console. It is at 'Step 4: Add deploy stage'. The 'Deploy action provider' is set to 'AWS Elastic Beanstalk'. The 'ApplicationName' is 'Bhushan\_App\_4', the 'EnvironmentName' is 'BhushanApp4-env', and 'Configure automatic rollback on stage failure' is 'Enabled'. A red error message at the bottom states: 'The trigger configuration for the source action is not valid. Make sure to choose one trigger configuration for each source action.' Navigation buttons at the bottom are 'Cancel', 'Previous', and 'Create pipeline'.

**Step 7:** Your Pipeline is Created Successfully. Also it is successfully connected to the source and successfully Deployed.

The screenshot shows the 'Bhushan\_PipeLines' pipeline in the 'Succeeded' state. A green banner at the top says 'Success: Congratulations! The pipeline Bhushan\_PipeLines has been created.' The pipeline type is 'V2' and the execution mode is 'QUEUED'. The 'Source' stage is shown as 'Succeeded' with a 'View details' button. The 'Deploy' stage is also shown as 'Succeeded' with a 'View details' button and a 'Start rollback' button. The left sidebar shows the navigation menu with 'Pipeline' selected under 'CodePipeline'.

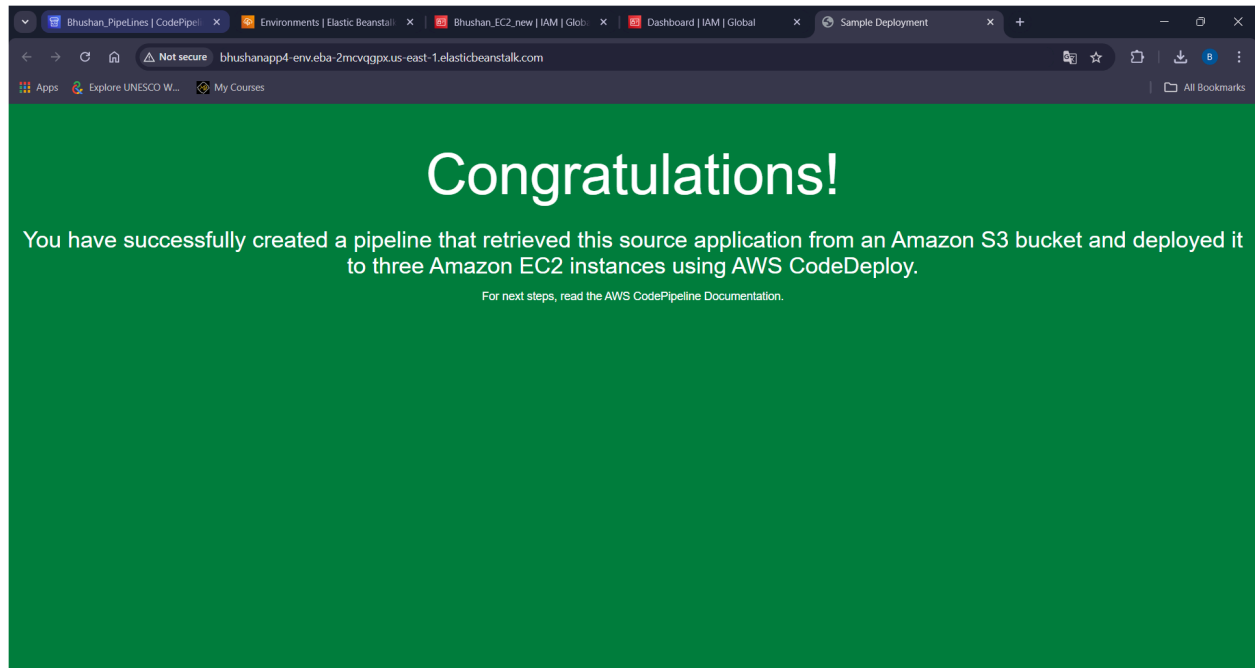
Name: **Bhushan Mukund Kor**

Academic Year: **2024-2025**

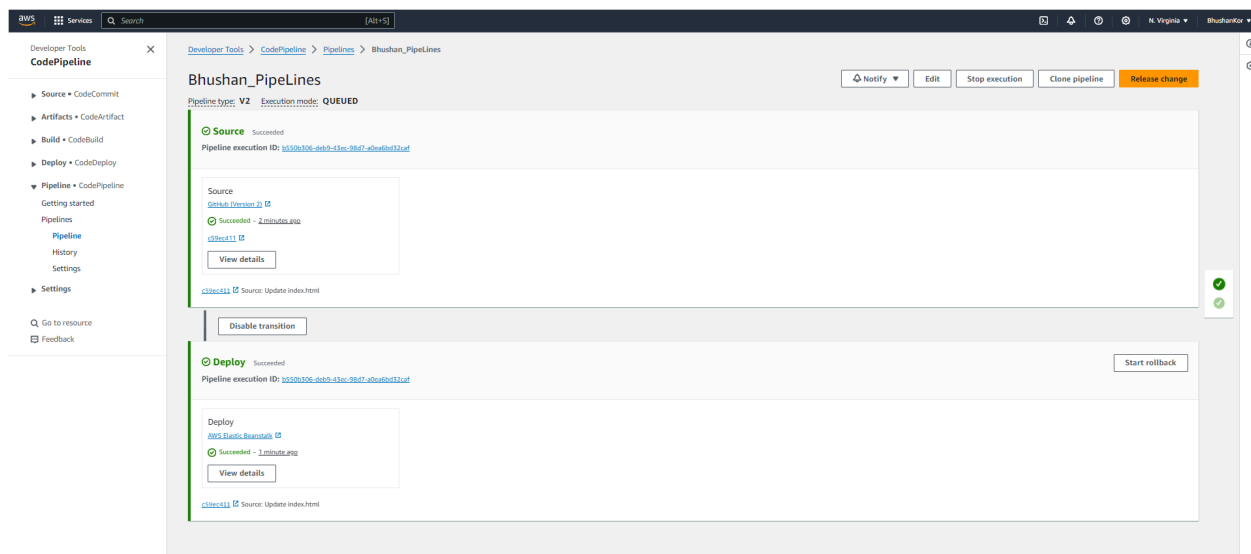
Division: **D15C**

Roll No: **28**

**Step 8:** Click on the given link and you will see the Result.



**Step 9:** Make Some changes in the Source code file on GitHub and commit changes and then deploy the changes in pipeline.



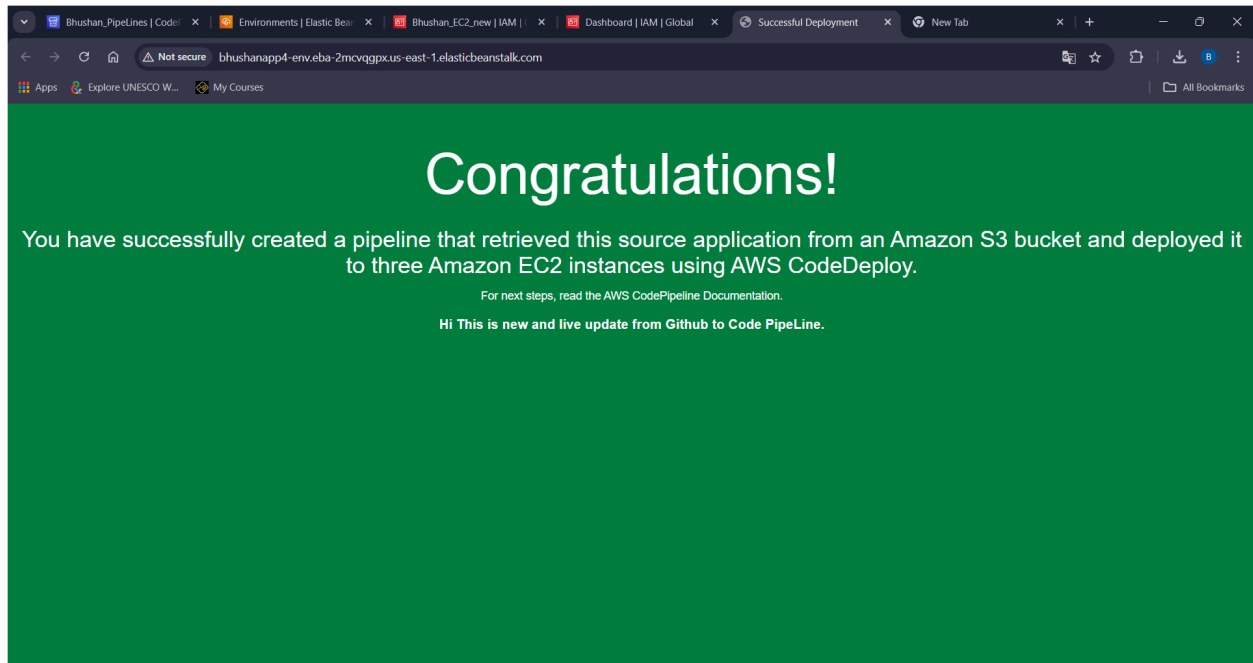
**Name: Bhushan Mukund Kor**

**Academic Year: 2024-2025**

**Division: D15C**

**Roll No: 28**

**Step 10:** Name of website is changed also one we line is added at the end.



**New Learnings in this experiment:****Note:**

Previously Elastic Beanstalk created a default EC2 instance profile named `aws-elasticbeanstalk-ec2-role` the first time an AWS account created an environment. This instance profile included default managed policies. If your account already has this instance profile, it will remain available for you to assign to your environments.

However, recent AWS security guidelines don't allow an AWS service to automatically create roles with trust policies to other AWS services, EC2 in this case. Because of these security guidelines, Elastic Beanstalk no longer creates a default `aws-elasticbeanstalk-ec2-role` instance profile.

The screenshot shows the AWS Management Console interface for configuring service access. The sidebar on the left lists steps 1 through 6, with step 2, 'Configure service access', being the current step. The main content area is titled 'Configure service access' and contains a 'Service access' section. This section has two radio buttons: 'Create and use new service role' (which is unselected) and 'Use an existing service role' (which is selected). Below these radio buttons is a section for 'Existing service roles' with a dropdown menu showing 'Bhushan\_EC2\_new'. There are also sections for 'EC2 key pair' and 'EC2 instance profile', both with dropdown menus showing 'Bhushan\_EC2\_key' and 'Bhushan\_EC2\_new' respectively. At the bottom of the main content area are buttons for 'Cancel', 'Skip to review', 'Previous', and 'Next'.

**Because of the above problem, we have chosen the existing role and for that, we have to create a new role with certain policies.**



**To create an instance profile**

1. Open the Roles page in the IAM console.
2. Choose Create role.
3. Under the Trusted entity type, choose AWS service.
4. Under Use case, choose EC2.
5. Choose Next.
6. Attach the appropriate managed policies provided by Elastic Beanstalk and any additional policies that provide permissions that your application needs.
7. Choose Next.
8. Enter a name for the role.
9. (Optional) Add tags to the role.
10. Choose Create role.

**For Point 6 Do the below Steps****To add managed policies to the role attached to the default instance profile**

1. Type `AWSElasticBeanstalk` to filter the policies.
2. Select the following policies, and then choose Attach policy:
  - `AWSElasticBeanstalkWebTier`
  - `AWSElasticBeanstalkWorkerTier`
  - `AWSElasticBeanstalkMulticontainerDocker`
3. Also, add `AmazonS3FullAccess` or `AmazonDynamoDBFullAccess`.
4. Choose Attach policy.

**Add the Trust relationship policy for EC2**

To allow the EC2 instances in your environment to assume the required role, the instance profile must specify Amazon EC2 as a trusted entity in the trust relationship policy, as follows.

```
{
  "Version": "2008-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
```

Name: Bhushan Mukund Kor

Academic Year: 2024-2025

Division: D15C

Roll No: 28

```
{
  "Service": "ec2.amazonaws.com"
},
{
  "Action": "sts:AssumeRole"
}
]
}
```

To customize permissions, you can add policies to the role attached to the default instance profile or create your own instance profile with a restricted set of permissions.

## Screenshot for reference

### Policies:

Policy name	Type	Attached entities
AdministratorAccess-AWSElasticBeanstalk	AWS managed	1
AmazonS3FullAccess	AWS managed	1
AWSCloud9EnvironmentMember	AWS managed	1
AWSCloud9SSMInstanceProfile	AWS managed	2
AWSElasticBeanstalkCustomPlatformforEC2Role	AWS managed	1
AWSElasticBeanstalkEnhancedHealth	AWS managed	1
AWSElasticBeanstalkManagedUpdatesCustomerRolePolicy	AWS managed	1
AWSElasticBeanstalkMulticontainerDocker	AWS managed	1
AWSElasticBeanstalkWebTier	AWS managed	1
AWSElasticBeanstalkWorkerTier	AWS managed	1

### Trust Relationship:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "ec2.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

**Academic Year:2024-2025**

**Roll No: 28**

**IAM Role:**

[IAM](#) > [Roles](#) > Bhushan\_EC2\_new

## Bhushan\_EC2\_new




Info

Delete

Allows EC2 instances to call AWS services on your behalf.

Summary

Edit

<div>Creation date</div> <div>August 11, 2024, 21:00 (UTC+05:30)</div>	<div>ARN</div> <div> arn:aws:iam::010928205712:role/Bhushan_EC2_new</div>	<div>Instance profile ARN</div> <div> arn:aws:iam::010928205712:instance-profile/Bhushan_EC2_new</div>
<div>Last activity</div> <div> 3 days ago</div>	<div>Maximum session duration</div> <div>1 hour</div>	

### EC2 Instance with attached IAM Role:

**AWS** | **Services** | **Search** [Alt+S] | **N. Virginia** | **Bhushan Kor**

---

EC2 Dashboard

EC2 Global View

Events

Console-to-Code [Preview](#)

**Instances**

**Instances**

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

**Images**

AMIs

AMI Catalog

**Elastic Block Store**

Volumes

Snapshots

Lifecycle Manager

**Network & Security**

Security Groups

Elastic IPs

Placement Groups

+ Successfully attached Bhushan\_EC2\_new to instance i-0e721f56644c4f090

### Instances (1/1) Info

All states ▾

<input checked="" type="checkbox"/>	Name ↗	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
<input checked="" type="checkbox"/>	Bhushan_EC2	i-0e721f56644c4f090	⏸ Stopped	t2.micro	-	<a href="#">View alarms +</a>	us-east-1c	-	-	-

### i-0e721f56644c4f090 (Bhushan\_EC2)

[Details](#)
[Status and alarms](#)
[Monitoring](#)
[Security](#)
[Networking](#)
[Storage](#)
[Tags](#)

**▼ Instance summary Info**

Instance ID  
[i-0e721f56644c4f090 \(Bhushan\\_EC2\)](#)

IPv6 address  
-

Hostname type  
IP name: ip-172-31-84-117.ec2.internal

Answer private resource DNS name  
IPv4 (A)  
-

Auto-assigned IP address  
-

IAM Role  
[Bhushan\\_EC2\\_new](#)

IMDSv2  
Required

Public IPv4 address  
-

Instance state  
⏸ Stopped

Private IP DNS name (IPv4 only)  
[ip-172-31-84-117.ec2.internal](#)

Instance type  
t2.micro

VPC ID  
[vpc-0893b22aad0bd85a5](#)

Subnet ID  
[subnet-0b21b9439d73960f](#)

Instance ARN  
[arn:aws:ec2:us-east-1:010928205712:instance/i-0e721f56644c4f090](#)

Private IPv4 addresses  
[172.31.84.117](#)

Public IPv4 DNS  
-

Elastic IP addresses  
-

Elastic Compute Optimizer finding  
[Opt-in to AWS Compute Optimizer for recommendations.](#) | [Learn more](#)

Auto Scaling Group name  
-

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences