



Vivekanand Education Society's

Institute of Technology

An Autonomous Institute Affiliated to University of Mumbai,, Approved by AICTE & Recognized by Govt. of Maharashtra
Hashu Advani Memorial Complex, Collector Colony, Chembur East, Mumbai - 400074.

Department of Information Technology

A.Y. 2024-25

Advance DevOps Lab

Experiment 02

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.

Roll No.	44
Name	GANESH SANJAY PANDHRE
Class	D15B
Subject	Advance DevOps Lab
LO Mapped	LO1: To understand the fundamentals of Cloud Computing and be fully proficient with Cloud based DevOps solution deployment options to meet your business requirements.
Grade:	

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.

THEORY :

Continuous deployment is a key practice in modern DevOps, enabling organizations to streamline their software release process by automating the deployment of application updates. It allows for the seamless delivery of code revisions to production environments without requiring explicit approval from a developer, thereby reducing time-to-market and enhancing the overall efficiency of the development lifecycle.

AWS CodePipeline is a continuous integration and continuous delivery (CI/CD) service that facilitates the building, testing, and deployment of code whenever there is a change in the source code repository. By automating these steps, CodePipeline ensures that new features, bug fixes, and updates are reliably and consistently delivered to users.

One of the critical components of a continuous deployment pipeline is the deployment environment, which is typically made up of virtual servers or containers that host the application.

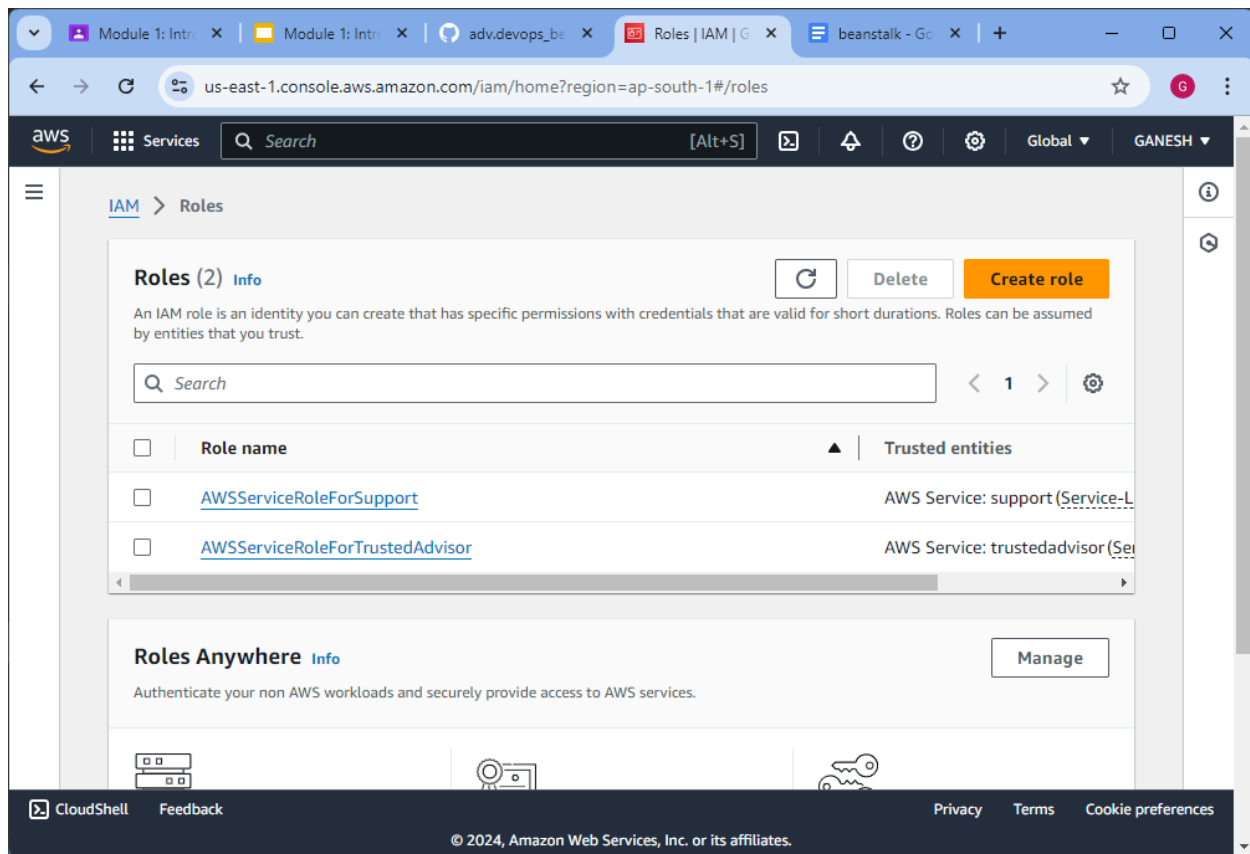
Amazon Elastic Beanstalk (EBS) is a Platform as a Service (PaaS) offering that simplifies the deployment and management of applications in the cloud. It abstracts the underlying infrastructure, such as EC2 instances, load balancers, and scaling configurations, allowing developers to focus on writing code without worrying about provisioning and maintaining the infrastructure.

In a typical AWS CodePipeline workflow, the source code for an application is stored in a version control system like GitHub, an S3 bucket, or AWS CodeCommit. The pipeline monitors this source repository for changes and triggers a series of automated actions whenever a change is detected. These actions might include building the application, running automated tests, and finally deploying the code to a live environment.

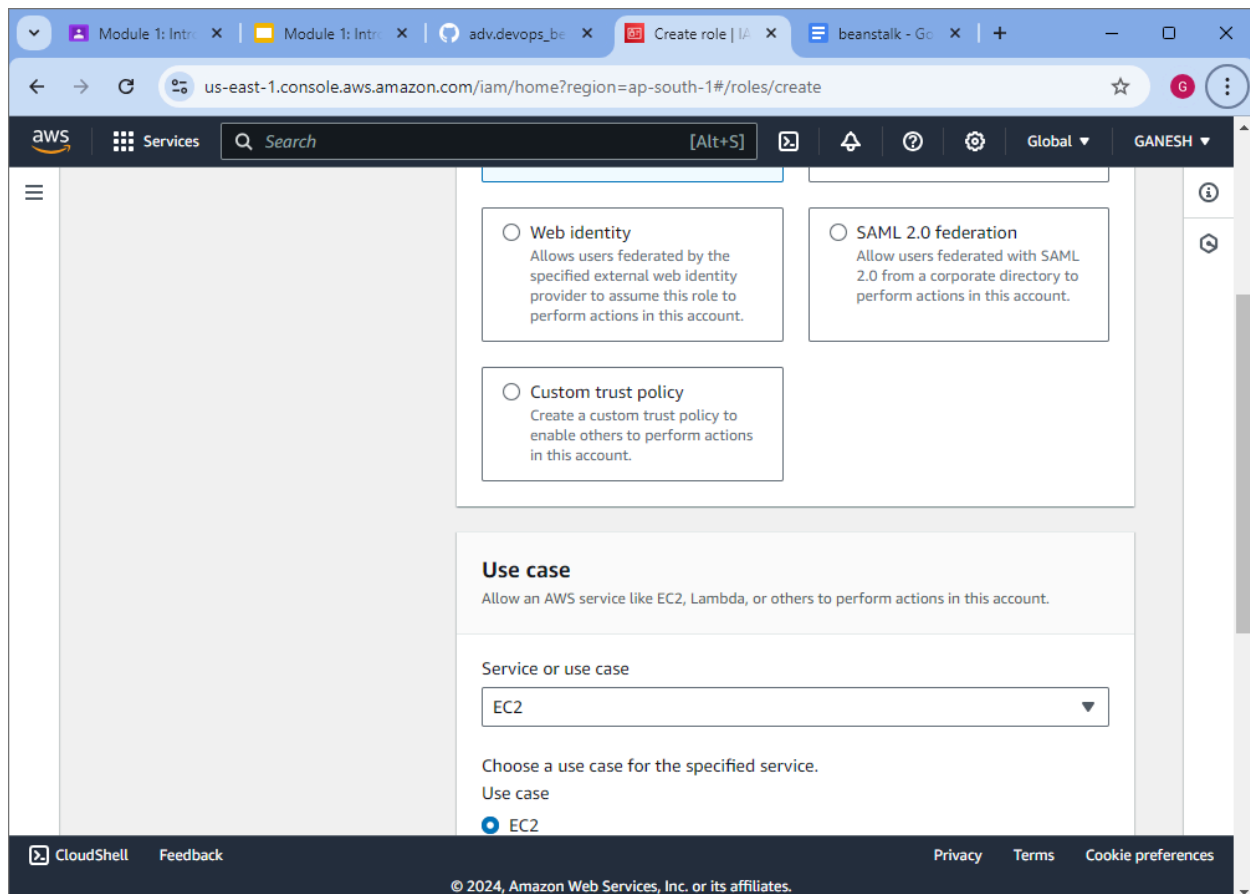
The deployment target in this setup could be an Amazon EC2 instance managed by Elastic Beanstalk, which takes care of the deployment details like setting up the necessary resources, deploying the code, and ensuring that the application is running smoothly. This integration with Elastic Beanstalk offers an out-of-the-box deployment solution that is both scalable and resilient.

AWS CodePipeline's integration with Elastic Beanstalk ensures that every code change goes through a consistent deployment process, thereby minimizing human errors and ensuring that the application remains stable and reliable. This automated process not only accelerates the development cycle but also improves the quality of the software by providing immediate feedback on the code's performance in a production-like environment.

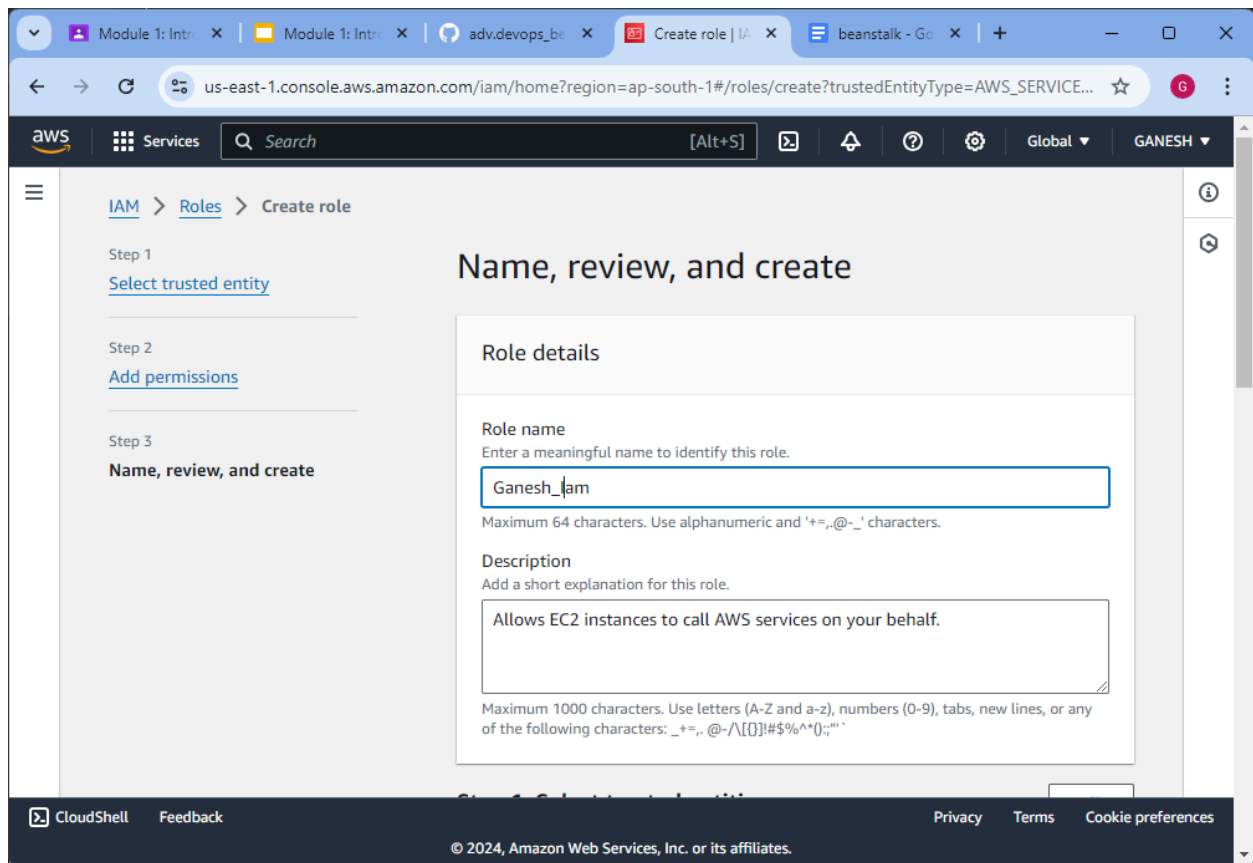
Create a role in an IAM.



Add EC2 for a service or use case.



Give name to the role.



us-east-1.console.aws.amazon.com/iam/home?region=ap-south-1#/roles/create?trustedEntityType=AWS_SERVICE...

Services Search [Alt+S] Global GANESH

Step 1
[Select trusted entity](#)

Step 2
[Add permissions](#)

Step 3
Name, review, and create

Name, review, and create

Role details

Role name
Enter a meaningful name to identify this role.

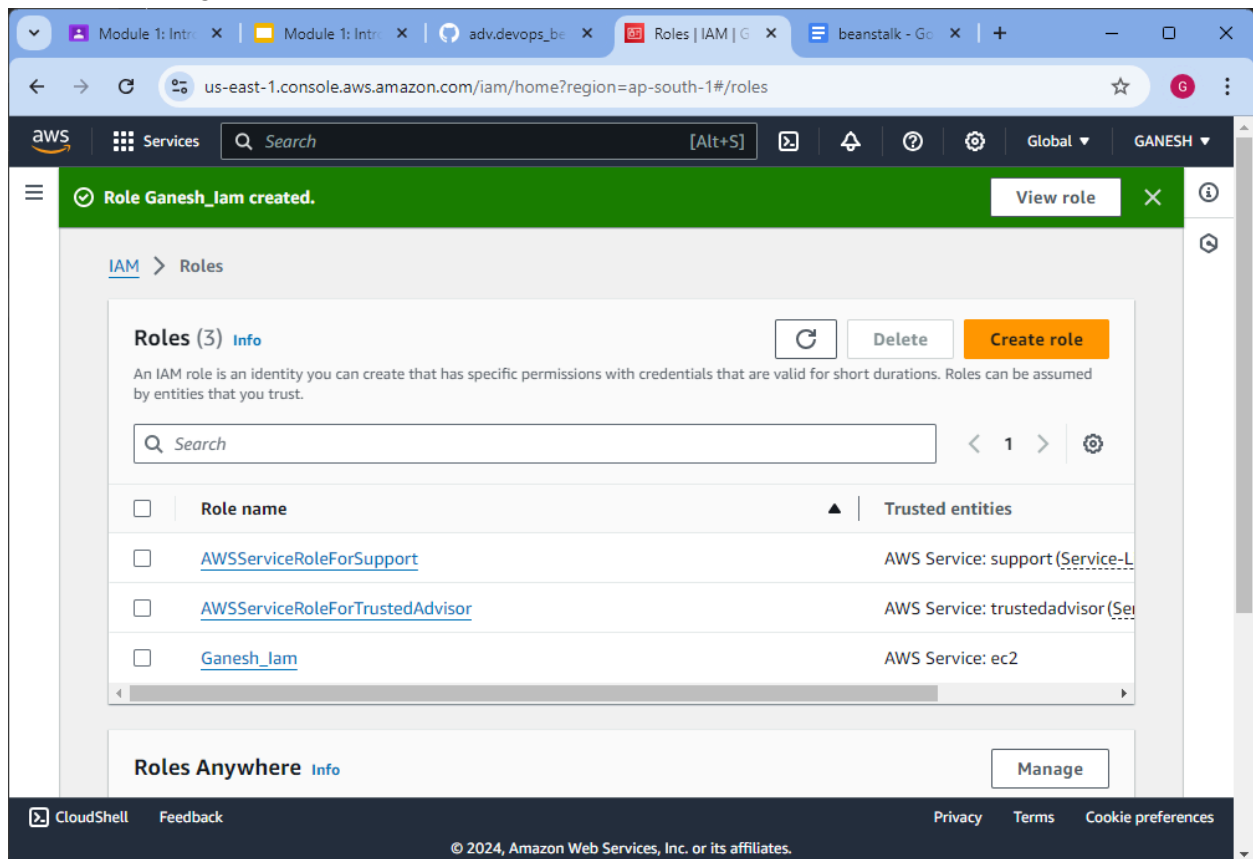
Maximum 64 characters. Use alphanumeric and '+,=, @, -, _' characters.

Description
Add a short explanation for this role.

Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: _ +=, @-/\[\]\!#\$%^&*(){};:~" ' `

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

IAM role is being created



us-east-1.console.aws.amazon.com/iam/home?region=ap-south-1#/roles

Services Search [Alt+S] Global GANESH

Role Ganesh_iam created. View role

Roles (3) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

< 1 >

<input type="checkbox"/>	Role name	Trusted entities
<input type="checkbox"/>	AWSServiceRoleForSupport	AWS Service: support (Service-L
<input type="checkbox"/>	AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Se
<input checked="" type="checkbox"/>	Ganesh_iam	AWS Service: ec2

Roles Anywhere Info

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

Go to the Elastic beanstalk and create an application. Give the appropriate name for the application.

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

Ganesh_website

Maximum length of 100 characters.

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates.

Select the platform as PHP.

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

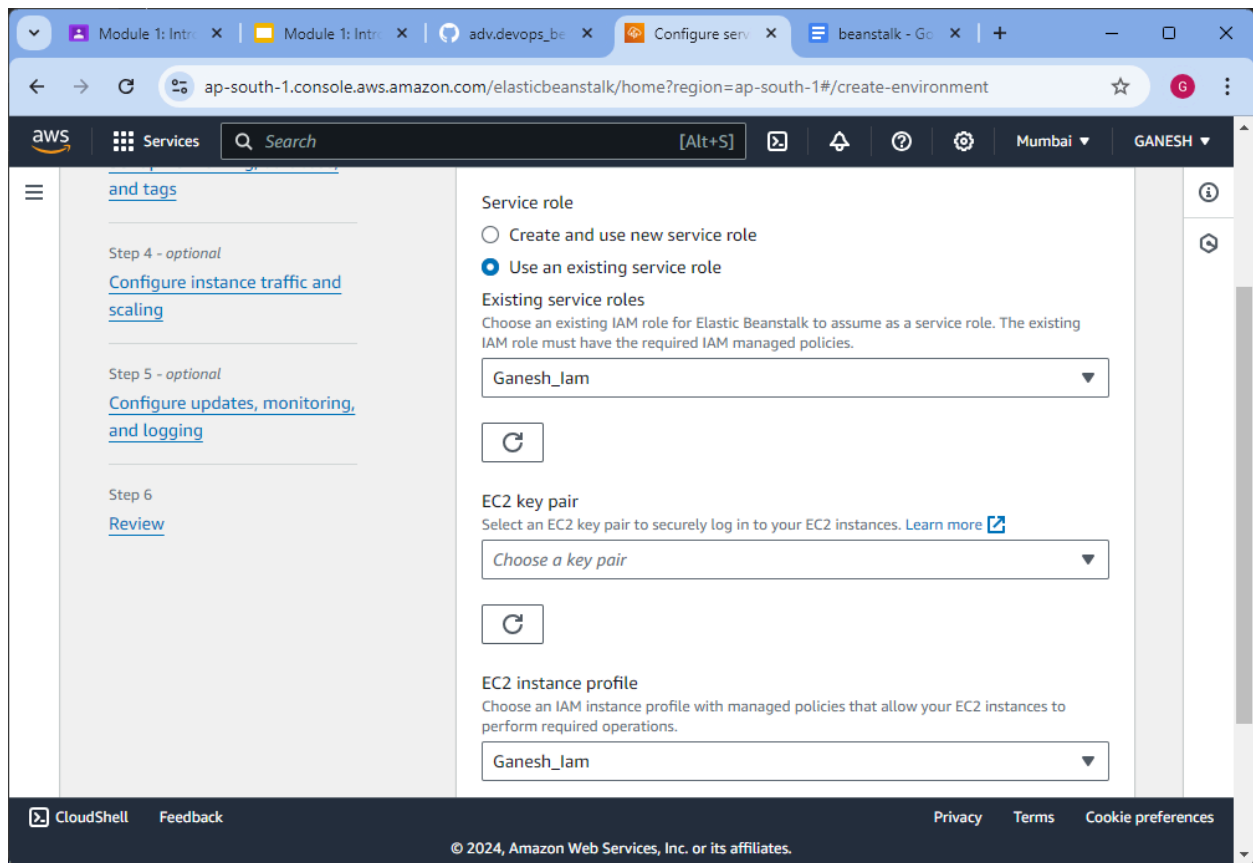
Platform branch

PHP 8.3 running on 64bit Amazon Linux 2023

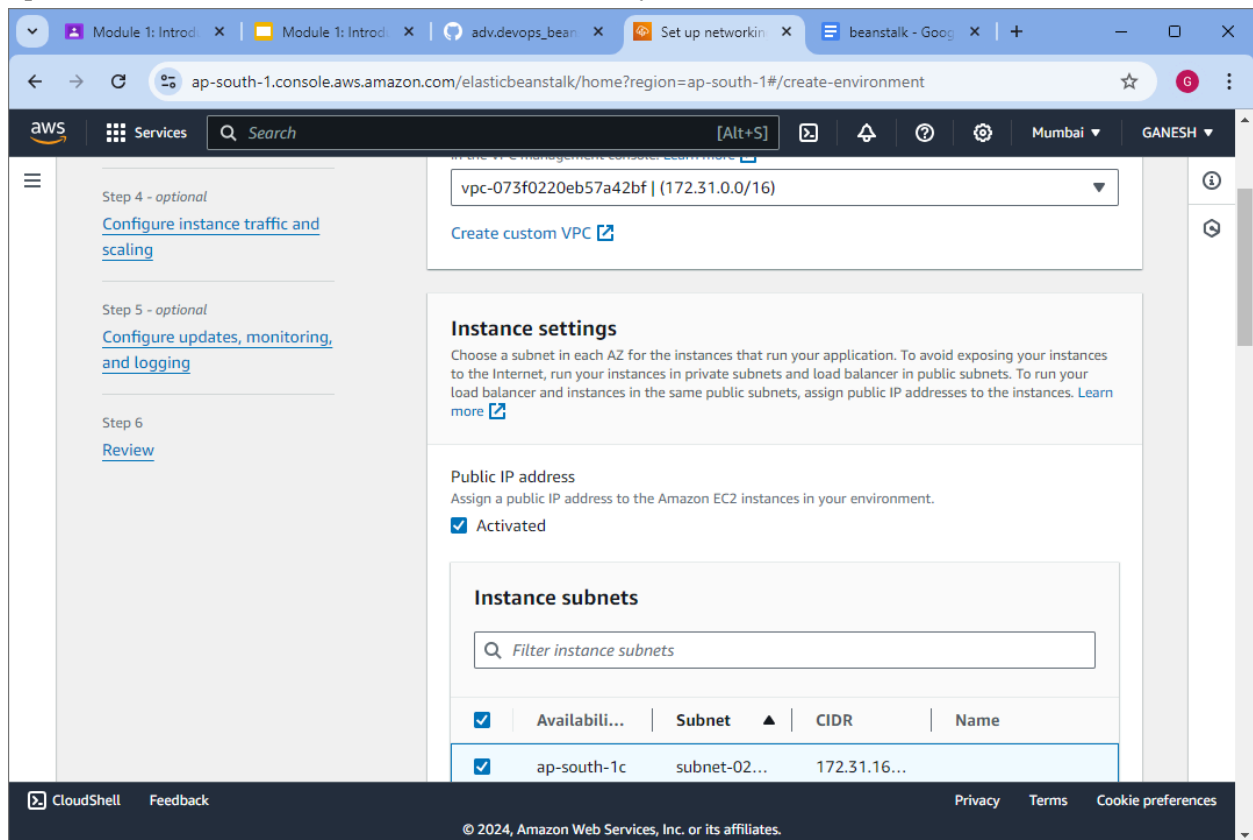
CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates.

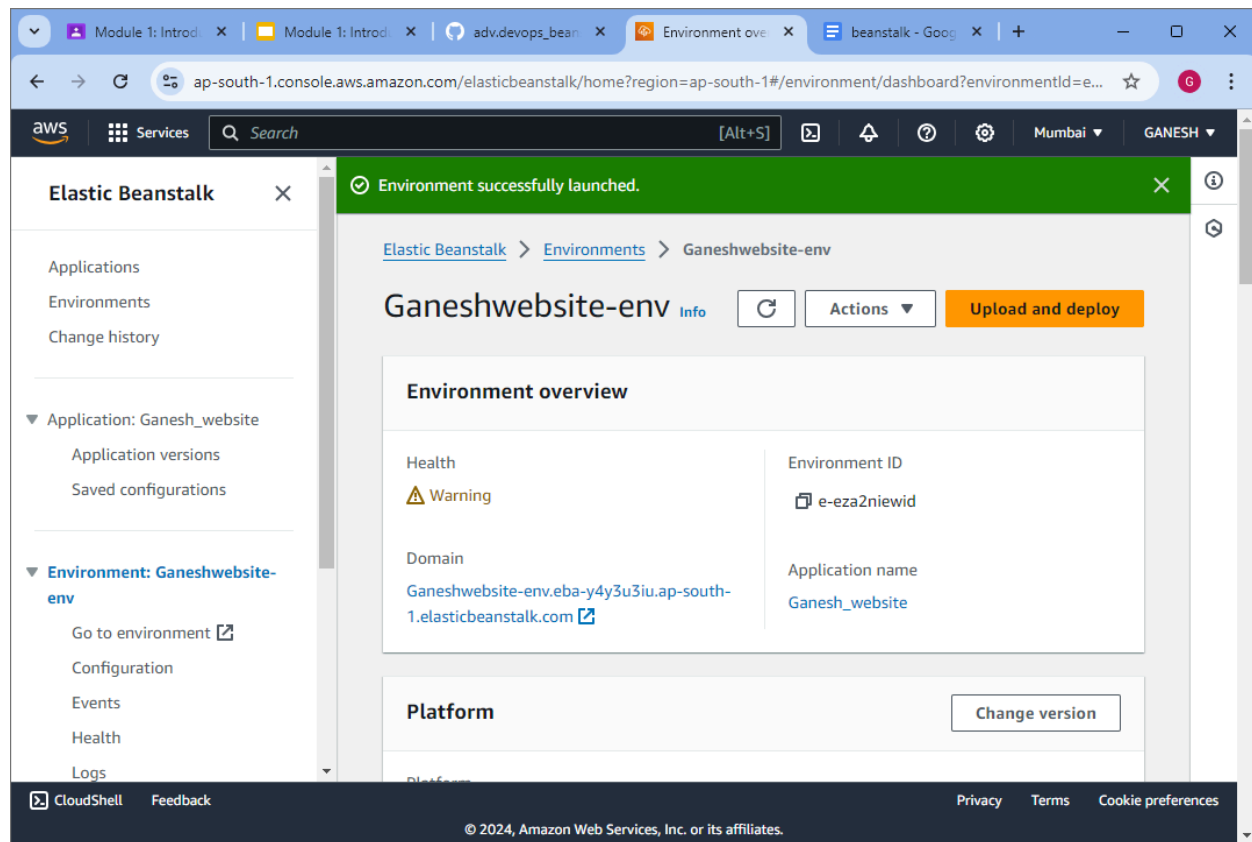
In an ec2 instance profile, select the created IAM role.



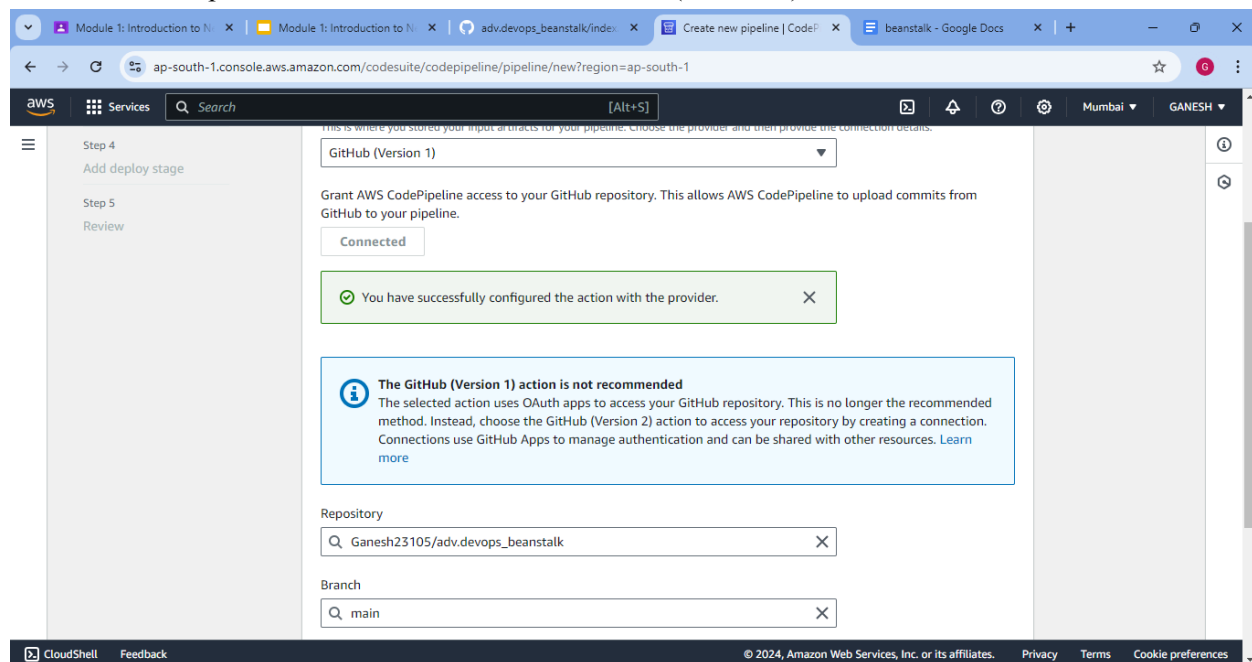
Vpc is to be selected. Public IP address and availability is to be checked.



Environment is launched successfully.



Go to the CodePipeline and select the source as GitHub (version 1).

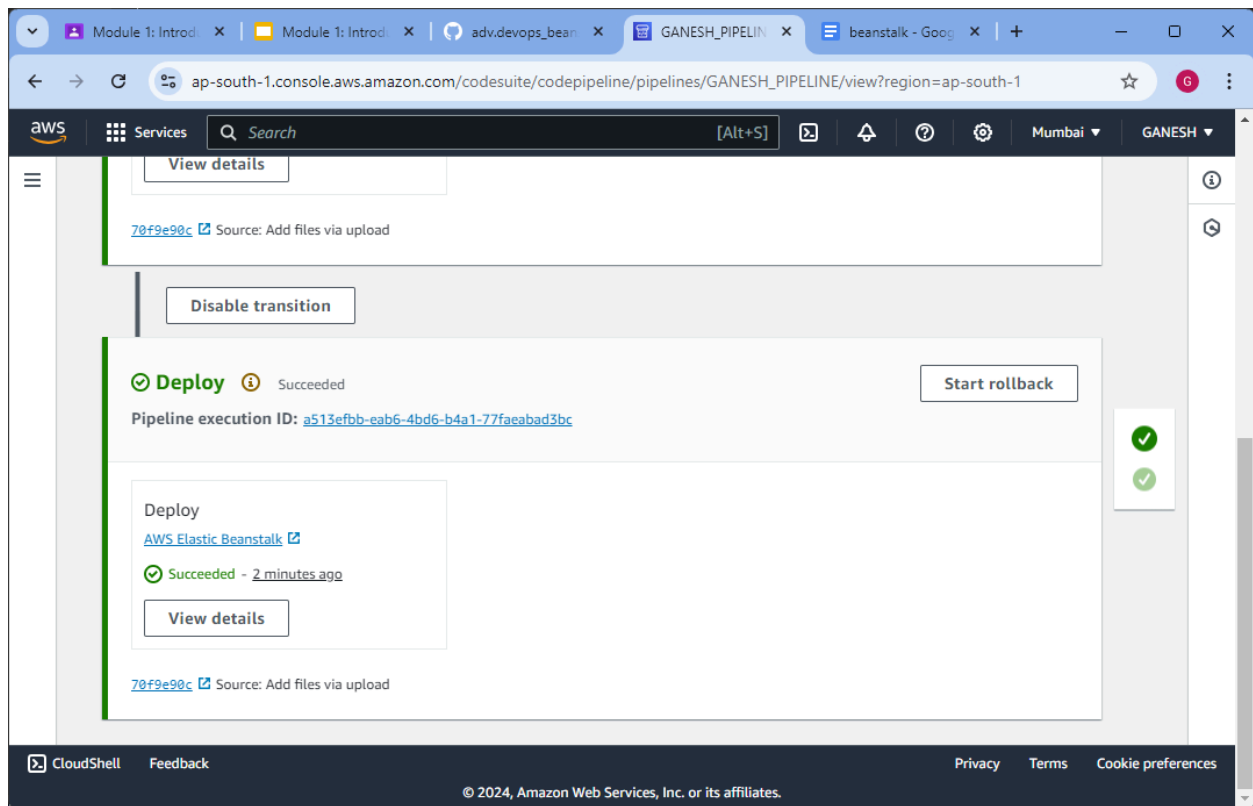


After skipping the build stage, AWS Elastic beanstalk is to be selected in the Deploy Provider. Select your recently created application name and environment name.

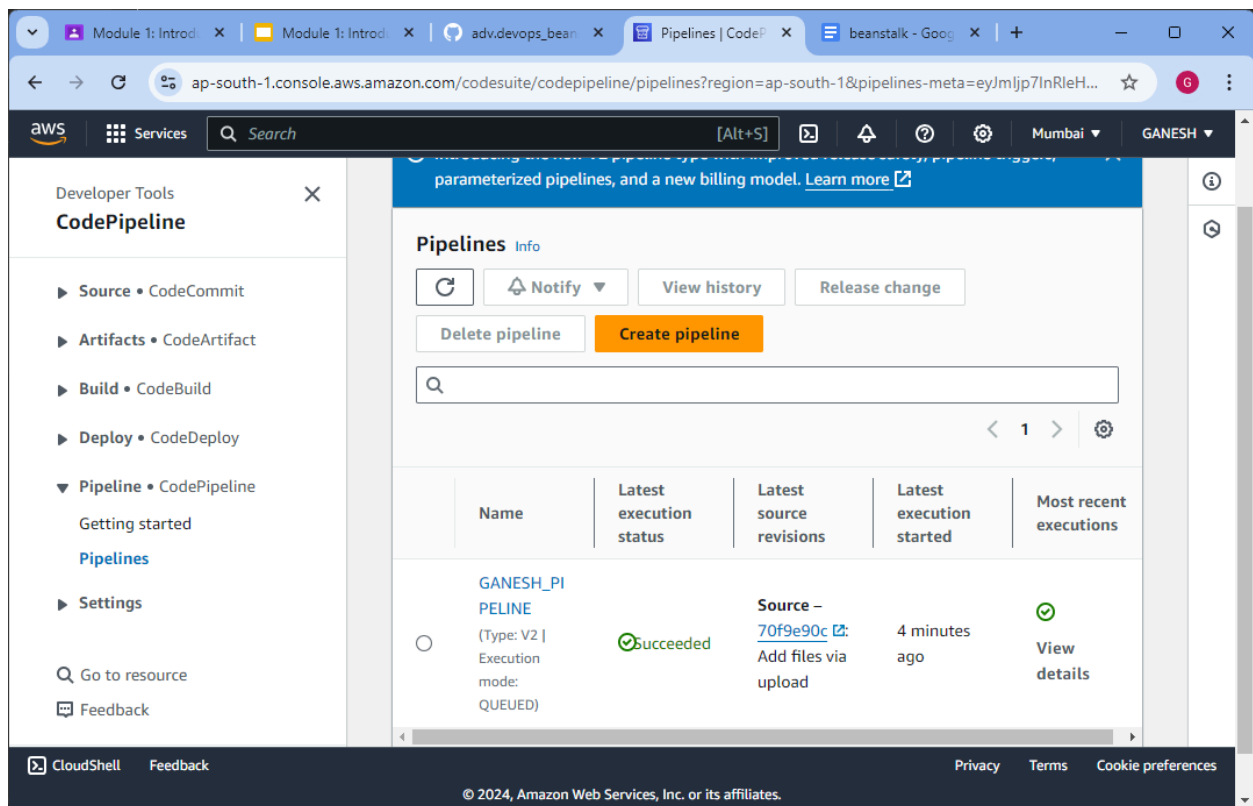
The screenshot shows the AWS CodePipeline console in the 'Deploy' step configuration. The 'Deploy provider' is set to 'AWS Elastic Beanstalk'. The 'Region' is 'Asia Pacific (Mumbai)'. The 'Input artifacts' section is empty. The 'Application name' is 'Ganesh_website' and the 'Environment name' is 'Ganeshwebsite-env'. There is a checkbox for 'Configure automatic rollback on stage failure' which is currently unchecked.

Pipeline is created. Source and Deploy section is also successful.

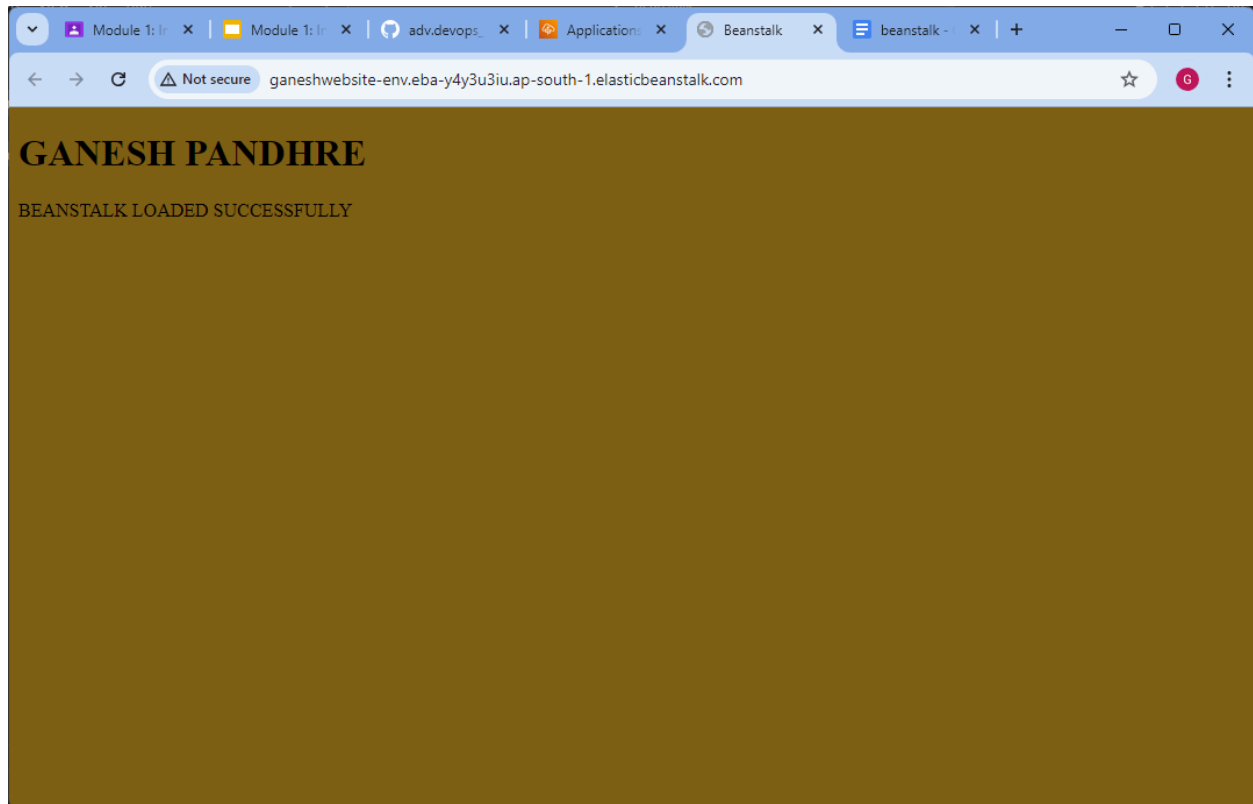
The screenshot shows the AWS CodePipeline console view for the 'GANESH_PIPELINE'. A green success banner at the top states: 'Success Congratulations! The pipeline GANESH_PIPELINE has been created.' Below the banner, the pipeline type is 'V2' and the execution mode is 'QUEUED'. The 'Source' stage is shown as 'Succeeded' with a green checkmark. The pipeline execution ID is 'a513efbb-eab6-4bd6-b4a1-77faeabad3bc'. The 'Source' stage details show it was 'Succeeded - 1 minute ago' with a link to the source code.



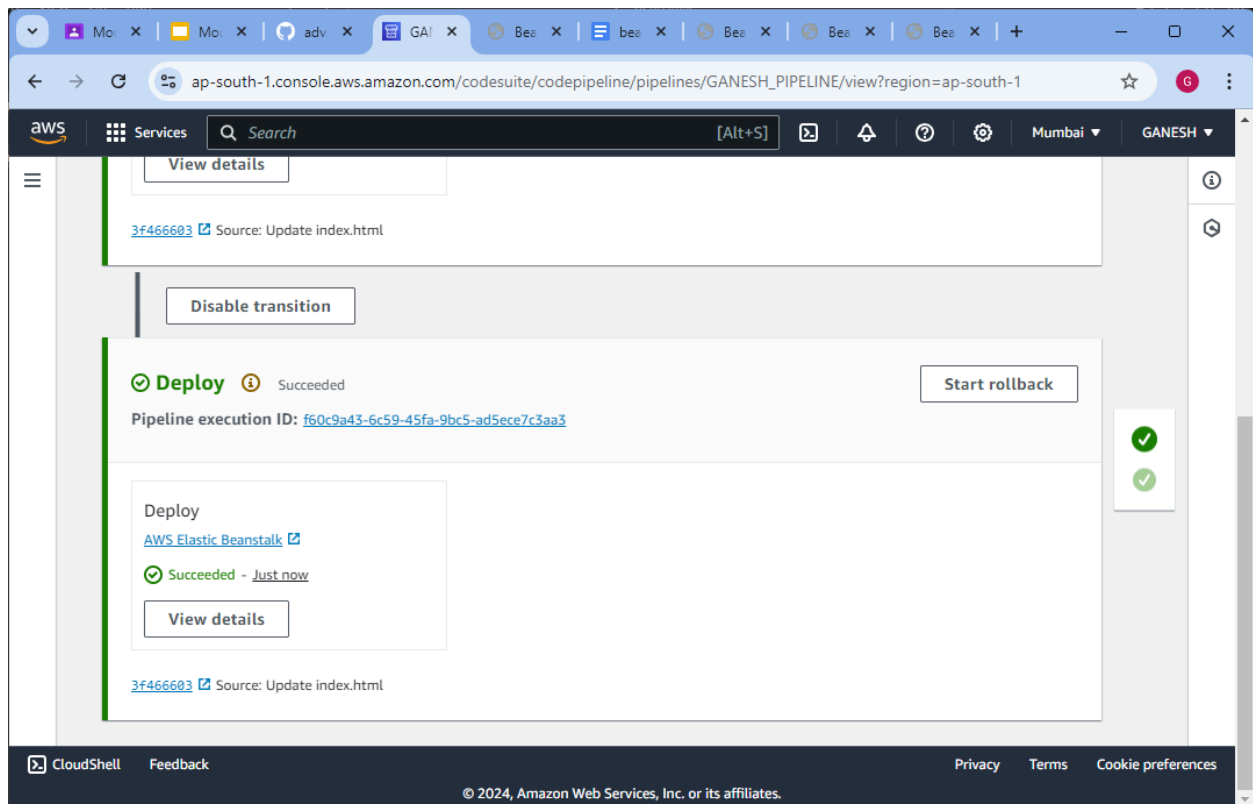
Go to the elastic beanstalk environment and click on domain.



Following output is to be generated of the code which is in the github repository.



Changes are done in the code of the Github repository and it is being directly deployed without any configurations.



Changes are to be reflected.



CONCLUSION :

Continuous deployment using AWS CodePipeline and Elastic Beanstalk represents a powerful approach to modern software development, where automation plays a crucial role in delivering high-quality software quickly and efficiently. This method supports the agile methodology by enabling rapid iterations and continuous improvements, leading to more responsive and innovative applications.