



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.	42
Name	Naikwadi Yash Shivdas
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 an IAM Role

Services See all 26 results ▶

IAM ☆
Manage access to AWS resources

1)

others to perform actions in this account.

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 ▼

Choose a use case for the specified service.

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

aws Services Search [Alt+S]

Step 2
[Add permissions](#)

Step 3
Name, review, and create

Role details

Role name
Enter a meaningful name to identify this role.
Yash-Naikwadi-42
Maximum 64 characters. Use alphanumeric and '+,=,@,_,-' characters.

Description
Add a short explanation for this role.
Allows EC2 instances to call AWS services on your behalf.
Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: _+=, @-/\[\]\#\\$\%^&*()~`''

Step 1: Select trusted entities Edit

Trust policy

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

Creating an environment using elastic beanstalk.

Search [Alt+S]

Global Yash_Naikwadi

Role Yash-Naikwadi-42 created. View role

Search

1

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

Yash-Naikwadi-42

Maximum length of 100 characters.

► Application tags (optional)

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.

Yash-Naikwadi-42



EC2 key pair

Select an EC2 key pair to securely log in to your EC2 instances. [Learn more](#)

Choose a key pair



EC2 instance profile

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

Yash-Naikwadi-42



[View permission details](#)

Public IP address

Assign a public IP address to the Amazon EC2 instances in your environment.

☒ Activated**Instance subnets**

<input type="checkbox"/>	Availability Zone	Subnet	CIDR	Name
<input type="checkbox"/>	ap-south-1a	subnet-042aa0c03...	172.31.32.0/20	
<input type="checkbox"/>	ap-south-1b	subnet-0b3307ce1...	172.31.0.0/20	
<input checked="" type="checkbox"/>	ap-south-1c	subnet-0c75485f6...	172.31.16.0/20	

Successful creation of Environment.

[Alt+S] 🔍 🔔 ? ⚙️ Mumbai ▼ Yash_Naikwadi ▼

✓ Environment successfully launched. ✕ ℹ️

[Elastic Beanstalk](#) > [Environments](#) > Yash-Naikwadi-42-env

Yash-Naikwadi-42-env Info

🔄 Actions ▼ Upload and deploy

Environment overview

Health
⚠️ Warning

Domain
[Yash-Naikwadi-42-env.eba-tax8ggku.ap-south-1.elasticbeanstalk.com](#) [🔗](#)

Environment ID
📄 e-2kpnxzyic3

Application name
Yash-Naikwadi-42

Creating pipeline using CodePipeline service from AWS.

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.

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.

[Alt+S]



Mumbai ▾

Yash_Naikwadi ▾

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.

Q

Yash-Naikwadi-42

X

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.


Q

Yash-Naikwadi-42-env

X

☐ Configure automatic rollback on stage failure

Successful creation of CodePipeline.


 **Success**

Congratulations! The pipeline Yash-Naikwadi-42 has been created.

Create a notification rule for this pipeline

[Developer Tools](#) > [CodePipeline](#) > [Pipelines](#) > Yash-Naikwadi-42

Yash-Naikwadi-42

 Notify ▼


Edit

Stop execution

Clone pipeline


Release change


Pipeline type: **V2** Execution mode: **QUEUED**


 **Source** Succeeded

Pipeline execution ID: [0289a572-c5ab-4242-9fc4-d41ed80bf696](#)



Source

[GitHub \(Version 1\)](#) 

 Succeeded - [Just now](#)

[7365ad78](#) 


Disable transition


 **Deploy**  Succeeded

[Start rollback](#)


Pipeline execution ID: [0289a572-c5ab-4242-9fc4-d41ed80bf696](#)

Deploy

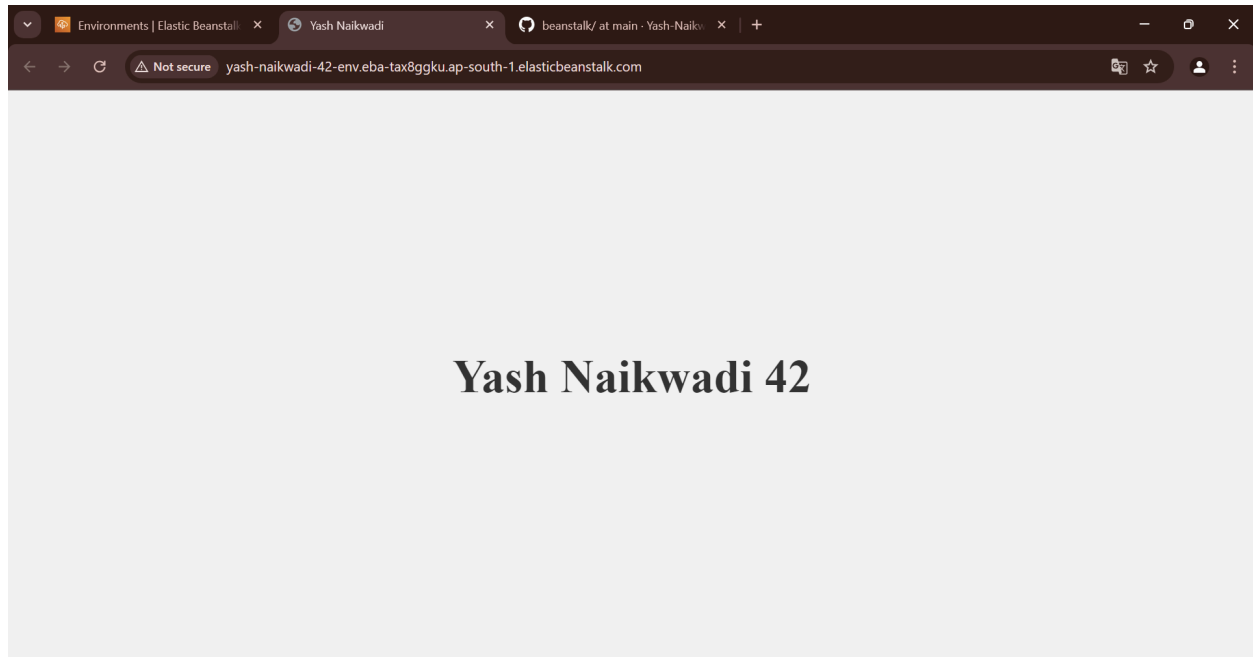
[AWS Elastic Beanstalk](#) 

 Succeeded - [Just now](#)

[View details](#)

[7365ad78](#)  Source: Create index.html

Selection domain from the specified environment and observing the webpage created, that is linked with GitHub.



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.