

**Name: Tarunkumar SHarma**  
**Div: D15B**  
**Roll no: 56**

## **EXPERIMENT NO. 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.**

Theory:

Continuous deployment allows you to deploy revisions to a production environment automatically without explicit approval from a developer, making the entire software release process automated.

You will create the pipeline using AWS CodePipeline, a service that builds, tests, and deploys your code every time there is a code change. You will use your GitHub account, an Amazon Simple Storage Service (S3) bucket, or an AWS CodeCommit repository as the source location for the sample app's code. You will also use AWS Elastic Beanstalk as the deployment target for the sample app. Your completed pipeline will be able to detect changes made to the source repository containing the sample app and then automatically update your live sample app.

### **Step1: Create a deployment environment**

Your continuous deployment pipeline will need a target environment containing virtual servers, or Amazon EC2 instances, where it will deploy sample code. You will prepare this environment before creating the pipeline.

**1) To simplify the process of setting up and configuring EC2 instances for this tutorial, you will spin up a sample environment using AWS Elastic Beanstalk. Elastic Beanstalk lets you easily host**

web applications without needing to launch, configure, or operate virtual servers on your own. It automatically provisions and operates the infrastructure (e.g. virtual servers, load balancers, etc.) and provides the application stack (e.g. OS, language and framework, web and application server, etc.) for you.

2) Name your web app and choose PHP from the drop-down menu(or any other language you are interested in) and then click Create Application.

1

New accounts only support launch templates

Starting on October 1, 2024, Amazon EC2 Auto Scaling will no longer support the creation of launch configurations for new accounts. Existing environments will not be impacted. For more information about other situations that are impacted, including temporary option settings required for new accounts, refer to [Launch templates](#) in the Elastic Beanstalk Developer Guide.

×

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

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

expi2

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.

## Application information [Info](#)

Application name

expi2

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

Expi2-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

Python

### Platform branch

Python 3.12 running on 64bit Amazon Linux 2023

### Platform version

4.2.0 (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

## Set up networking, database, and tags - optional [Info](#)

### 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-004343f75115095e3 | (10.0.0.0/16) | defaultVPC-vpc

[Create custom VPC](#)

## Expi2-env Info



Actions ▼

Upload and deploy

### Environment overview

Health

☹ Grey

Domain

–

Environment ID

📄 e-xyt42z95ai

Application name

[expi2](#)

### Platform

Change version

Platform

Python 3.12 running on 64bit Amazon Linux 2023/4.2.0

Running version

–

Platform state

🟢 Supported

# Congratulations

Your first AWS Elastic Beanstalk Python Application is now running on your own dedicated environment in the AWS Cloud

This environment is launched with Elastic Beanstalk Python Platform

## What's Next?

- [AWS Elastic Beanstalk overview](#)
- [AWS Elastic Beanstalk concepts](#)
- [Deploy a Django Application to AWS Elastic Beanstalk](#)
- [Deploy a Flask Application to AWS Elastic Beanstalk](#)
- [Customizing and Configuring a Python Container](#)
- [Working with Logs](#)

# 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

