

Project 8: AWS Elastic Beanstalk

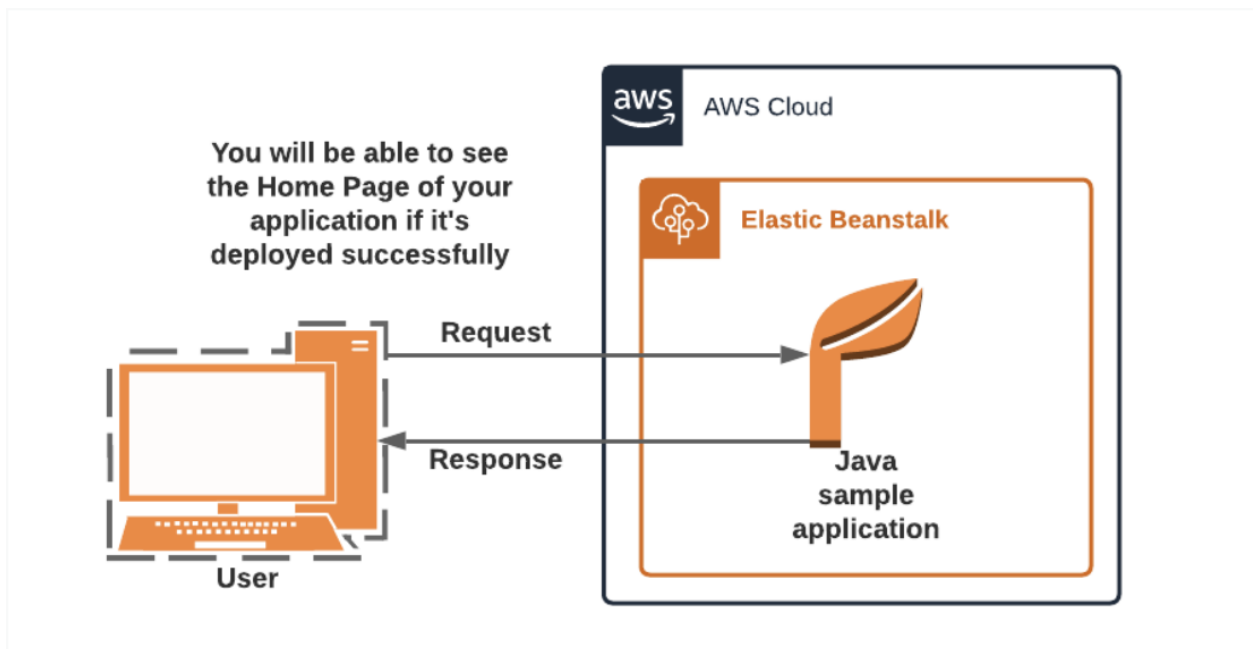
What is AWS Elastic Beanstalk?

- AWS 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 services such as Apache, Nginx, Passenger, and IIS.
- You can simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, auto-scaling to application health monitoring. At the same time, you retain full control over the AWS resources powering your application and can access the underlying resources at any time.

Objective of this project:

- We will be quickly deploying and managing a Java application in the AWS Cloud without worrying about the infrastructure that runs those applications.

Architecture Diagram



Step 1: Navigate to Elastic Beanstalk and to the Application Section

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](#)

Step 2: Create an Application

- With this section, we create a new application and environment with a sample application or your own code. With this you create an environment that can allow Elastic Beanstalk to manage Amazon Web resources and services on your behalf.

We will call this Sample Application and using the Python Platform.

Application information

Application name

Up to 100 Unicode characters, not including forward slash (/).

Platform

Platform

Python

Platform branch

Python 3.8 running on 64bit Amazon Linux 2

Platform version

3.3.2 (Recommended)

Application code

☒ **Sample application**
Get started right away with sample code.

☐ **Upload your code**
Upload a source bundle from your computer or copy one from Amazon S3.

Cancel

Configure more options

Create application

Step 3: Test the Application to See if it's working

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](#)

Sampleapplication-env

[Sampleapplication-env.eba-ybwgxwnu.us-east-1.elasticbeanstalk.com](#) (e-qgpyajk9n2)

Application name: **Sample Application**

Refresh

Actions ▼

Health



Ok

Causes

Running version

Sample Application

Upload and deploy

Platform



Python 3.8 running on 64bit
Amazon Linux 2/3.3.2

Change