NAME:VAISHNAL MALI
DIV:D15A
ADVANCED DEV-OPS 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.

Theory:

AWS Elastic Beanstalk (Elastic Beanstalk)

AWS Elastic Beanstalk is a fully managed service that makes it easy to deploy, manage, and scale web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker. It abstracts the underlying infrastructure management (like servers, networking, and storage) so that developers can focus on writing code.

Key Features:

- Ease of Use: You can simply upload your code, and Elastic Beanstalk automatically handles deployment, from capacity provisioning, load balancing, and auto-scaling to application health monitoring.
- Platform Flexibility: Elastic Beanstalk supports a variety of application platforms, making it a versatile choice for different types of projects.
- Customization: While Elastic Beanstalk manages the infrastructure, you still have full control over the AWS resources that power your application.
- Environment Management: Elastic Beanstalk allows you to create and manage multiple environments, making it easier to test and deploy new versions of your application.
- Cost-Effectiveness: You pay only for the AWS resources needed to run your applications, and there is no additional charge for Elastic Beanstalk.
- Elastic Beanstalk is ideal for developers who want to focus on writing code rather than managing infrastructure. It is commonly used to quickly deploy web applications or RESTful APIs.

AWS CodePipeline

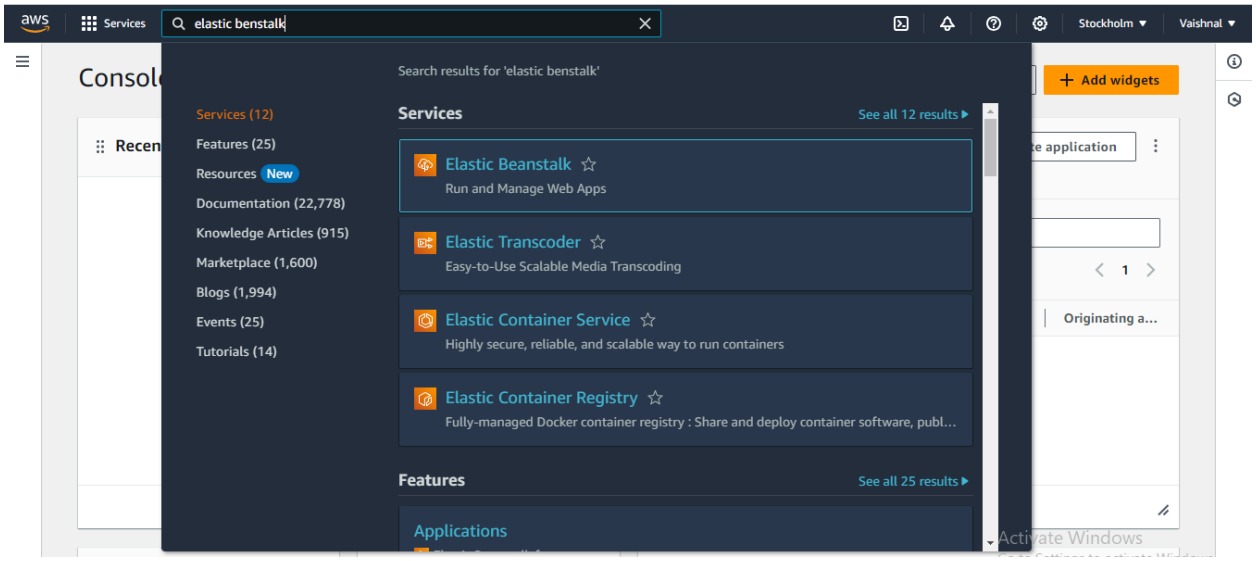
AWS CodePipeline is a fully managed continuous integration and continuous delivery (CI/CD) service that automates the build, test, and deployment phases of your release process. It enables you to model and automate the steps required to release your software.

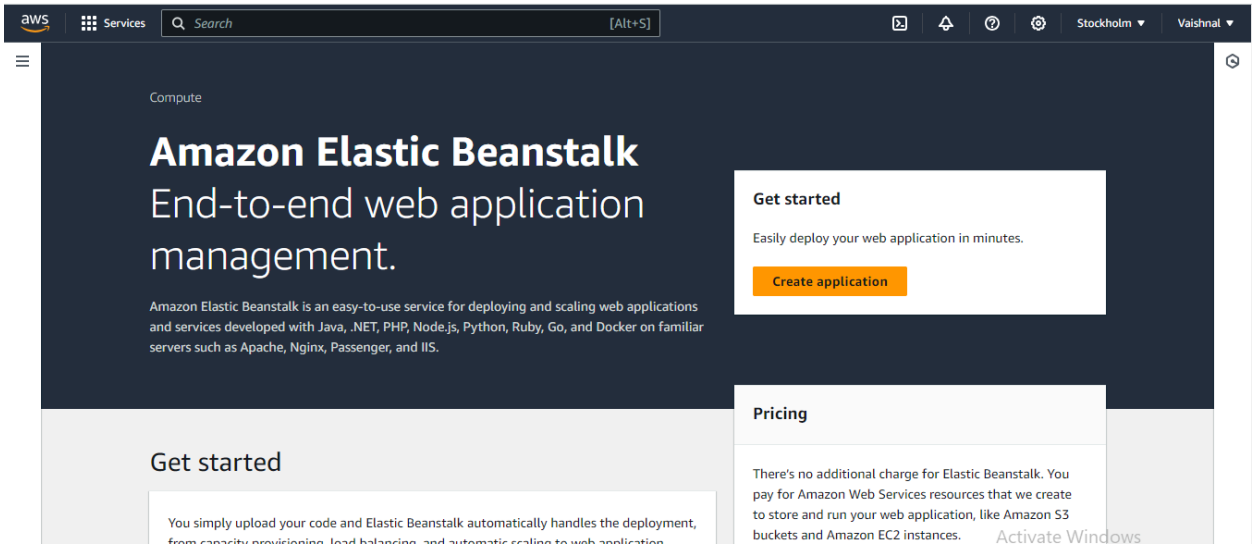
Key Features:

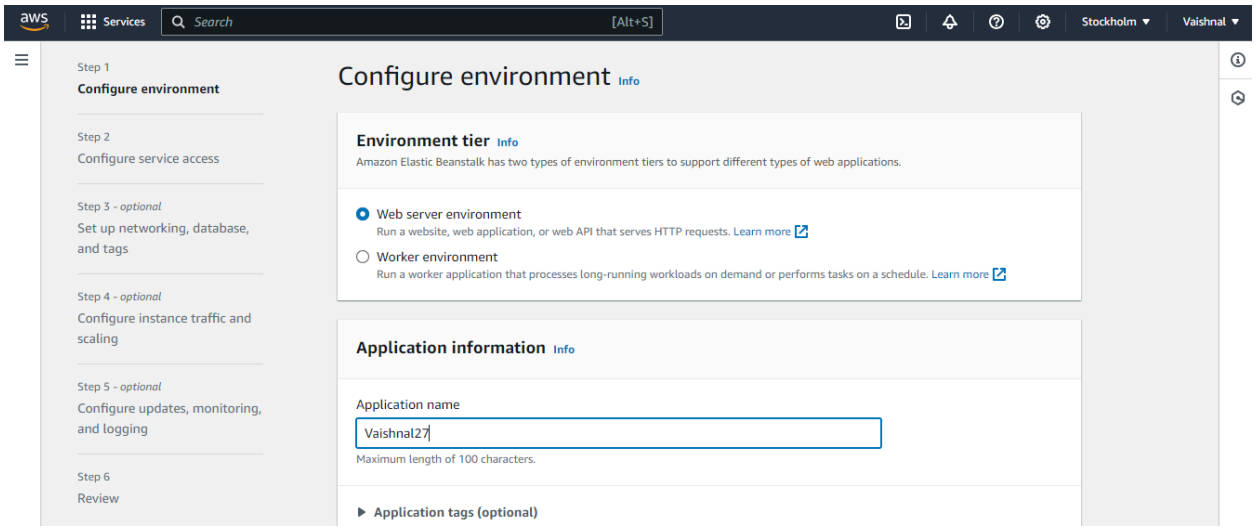
- Continuous Integration and Delivery: CodePipeline automates the release process, enabling you to rapidly deliver updates and new features to your users.
- Integration with Other AWS Services: CodePipeline integrates seamlessly with other AWS services such as CodeBuild, CodeDeploy, Elastic Beanstalk, and S3, providing a powerful and cohesive toolset for CI/CD.

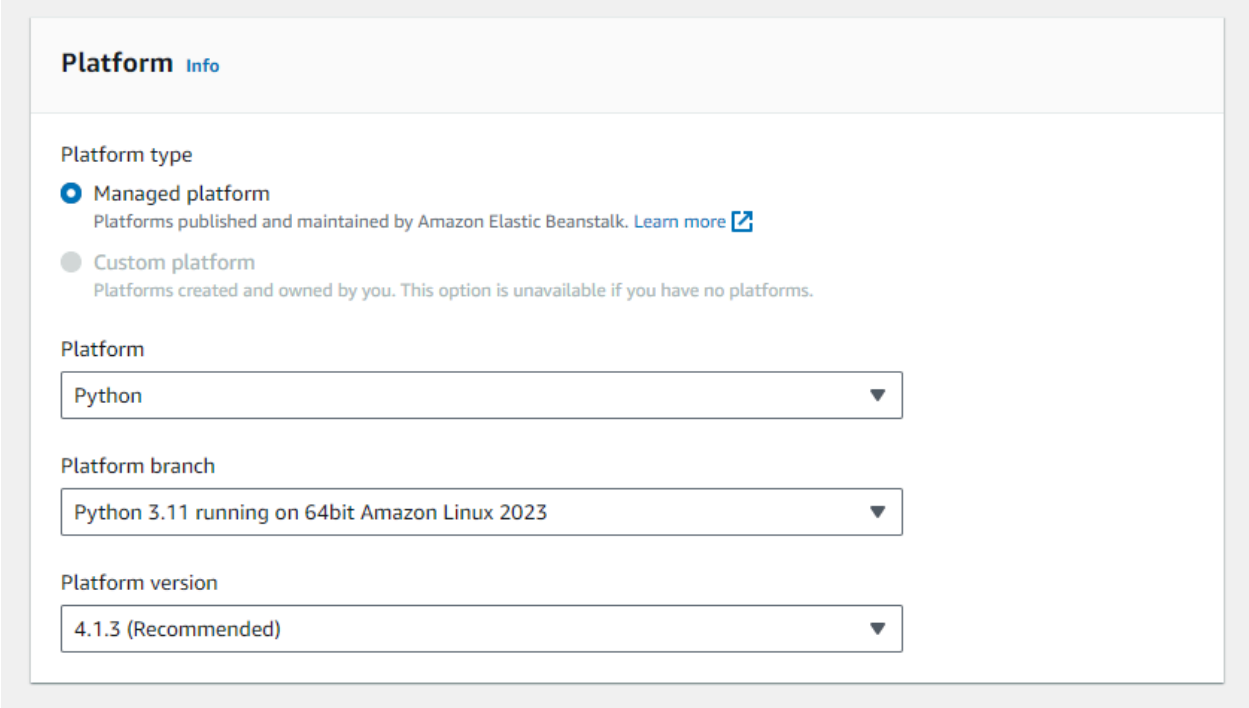
- Customizable Workflows: You can define your workflow and customize the steps for building, testing, and deploying your application.
- Third-Party Integration: CodePipeline can integrate with third-party tools such as GitHub, Jenkins, and Bitbucket, allowing you to use existing tools and workflows.
- Parallel Execution: CodePipeline can execute multiple stages in parallel, which speeds up the deployment process.
- AWS CodePipeline is ideal for teams looking to implement CI/CD practices. It is used to automate the entire release process, ensuring that your application is tested and deployed consistently and reliably every time there is a change in the code.

Implementation:






aws
Services
Search
[Alt+S]
Stockholm
Vaishnal
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
Vaishnal27
Maximum length of 100 characters.
Application tags (optional)


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 3.11 running on 64bit Amazon Linux 2023
Platform version
4.1.3 (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.

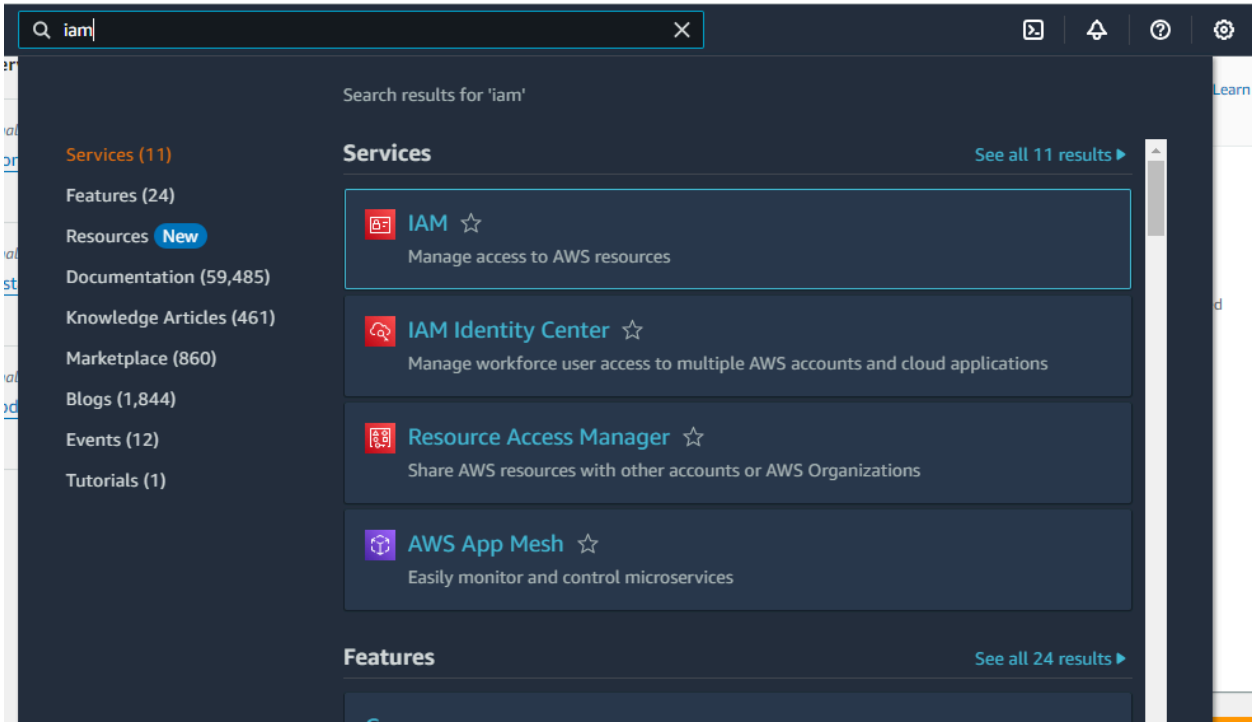
Presets Info

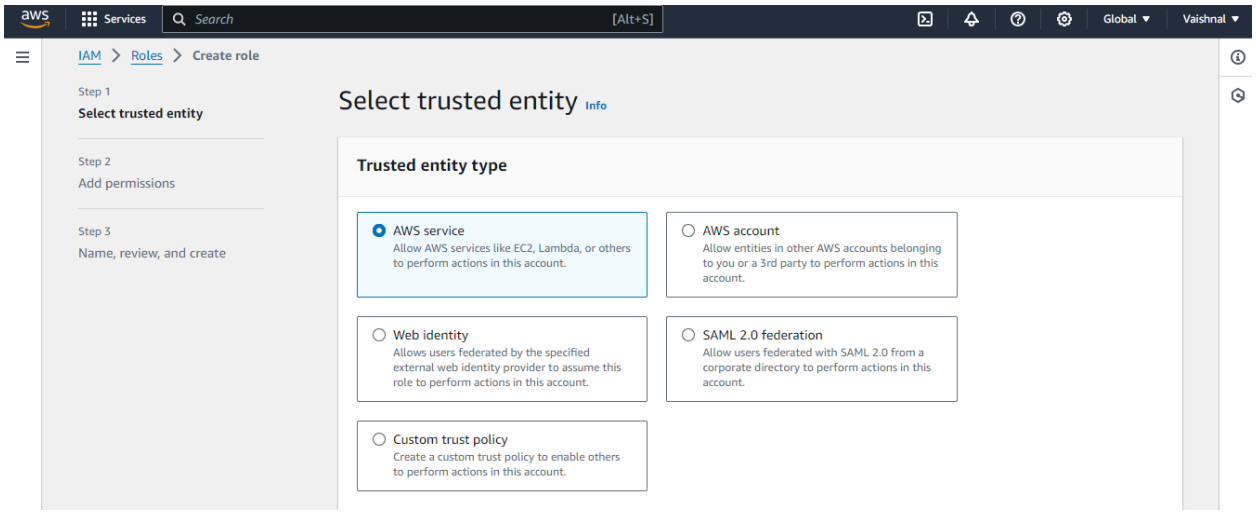
Start from a preset that matches your use case or choose custom configuration to unset recommended values and use the service's default values.

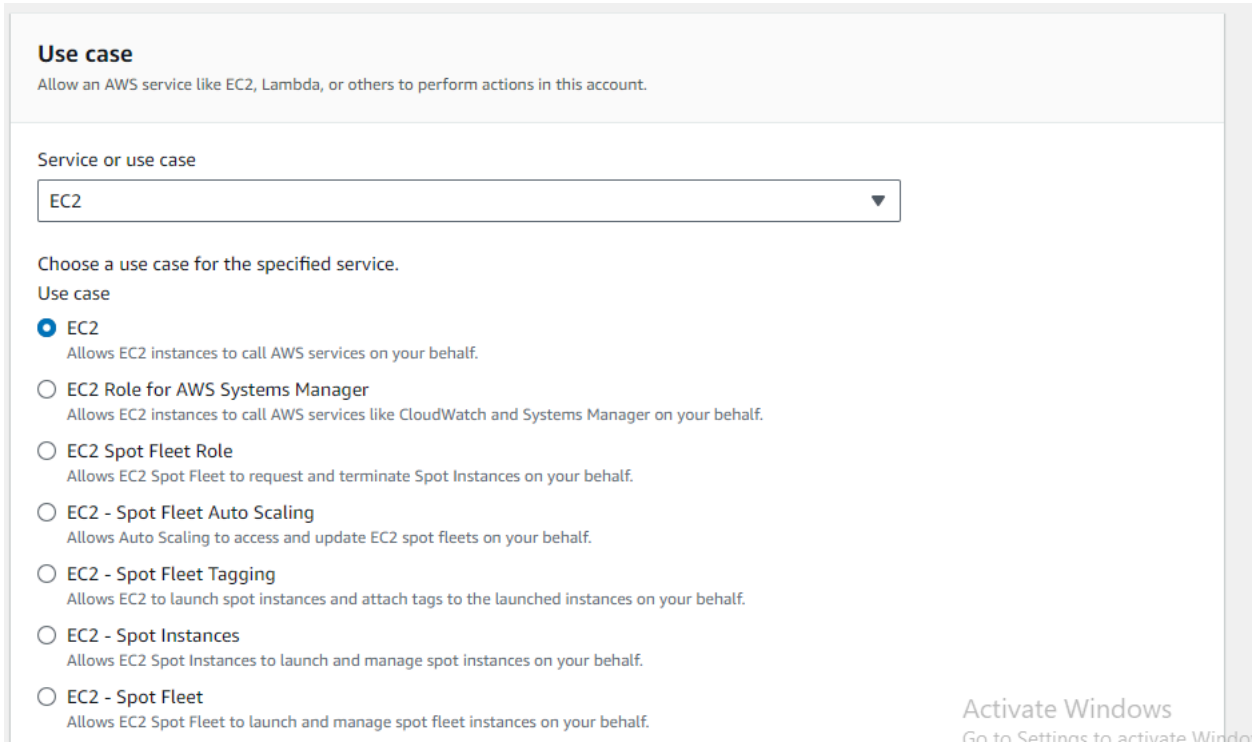
Configuration presets

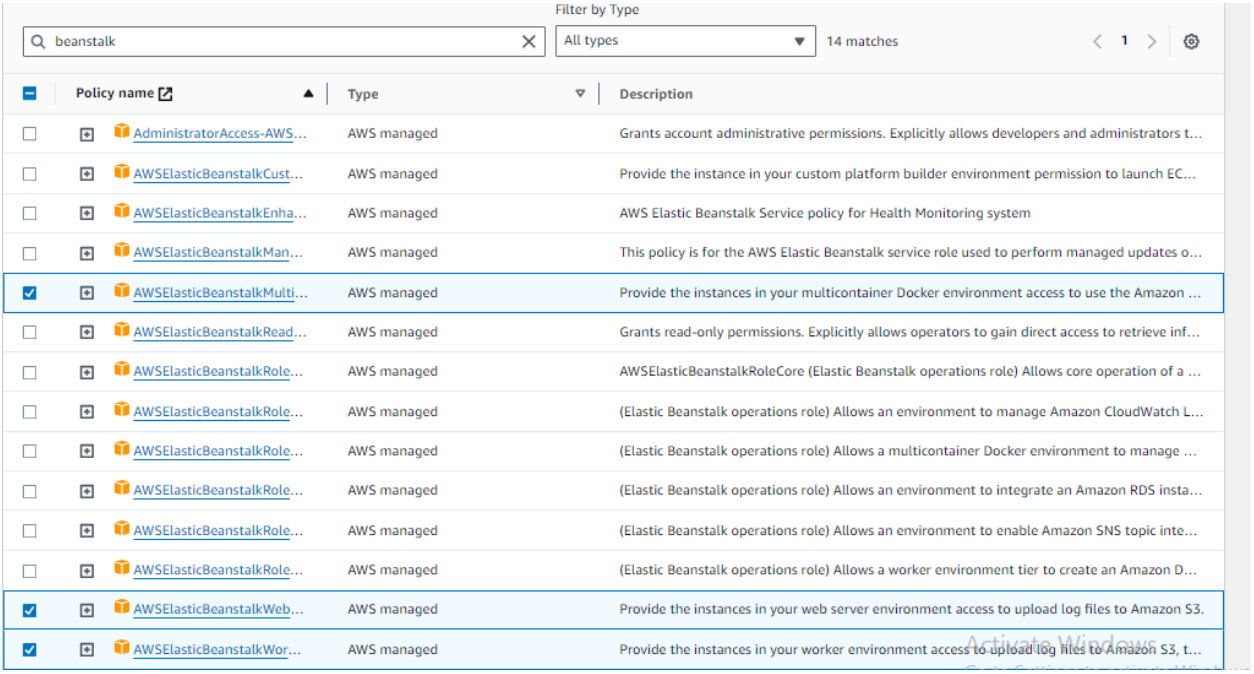
- (•) Single instance (free tier eligible)
- ( ) Single instance (using spot instance)
- ( ) High availability
- ( ) High availability (using spot and on-demand instances)
- ( ) Custom configuration

Cancel Next




aws
Services
Search
[Alt+S]
Global
Vaishnal
IAM
Roles
Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Select trusted entity
Info
Trusted entity type
AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
AWS account
Allow entities in other AWS accounts 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.
Use case
EC2
Allows EC2 instances to call AWS services on your behalf.
EC2 Role for AWS Systems Manager
Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.
EC2 Spot Fleet Role
Allows EC2 Spot Fleet to request and terminate Spot Instances on your behalf.
EC2 - Spot Fleet Auto Scaling
Allows Auto Scaling to access and update EC2 spot fleets on your behalf.
EC2 - Spot Fleet Tagging
Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.
EC2 - Spot Instances
Allows EC2 Spot Instances to launch and manage spot instances on your behalf.
EC2 - Spot Fleet
Allows EC2 Spot Fleet to launch and manage spot fleet instances on your behalf.
Activate Windows



Name, review, and create

Role details

Role name
Enter a meaningful name to identify this role.

aws-elaticbeanstalk-role

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: _+=,. @-/\[{}]!#$%^*();:"'`

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.

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.

aws-elaticbeanstalk-role

View permission details

Cancel **Skip to review** **Previous** **Next**

Elastic Beanstalk > Environments > Vaishnal27-env

Vaishnal27-env Info

Actions **Upload and deploy**

Environment overview

Health
Pending

Environment ID
e-5gujkvgkpa

Domain
–

Application name
Vaishnal27

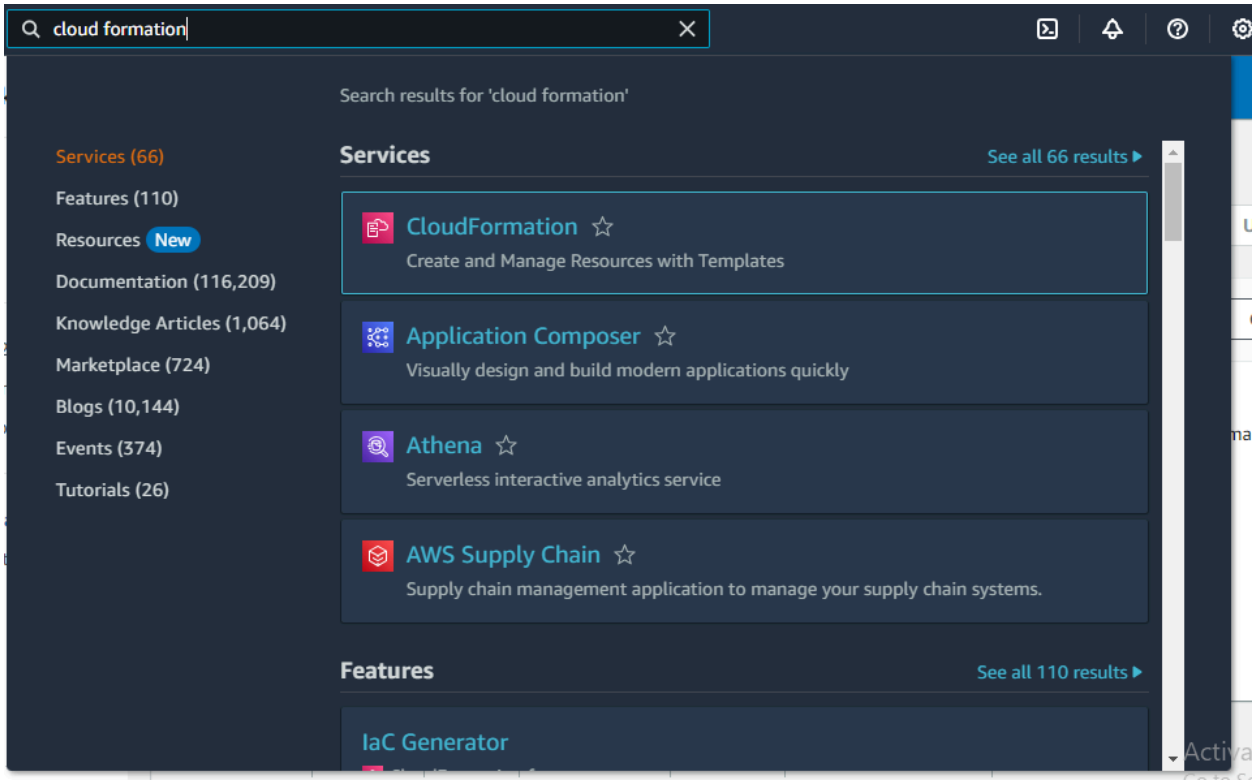
Platform

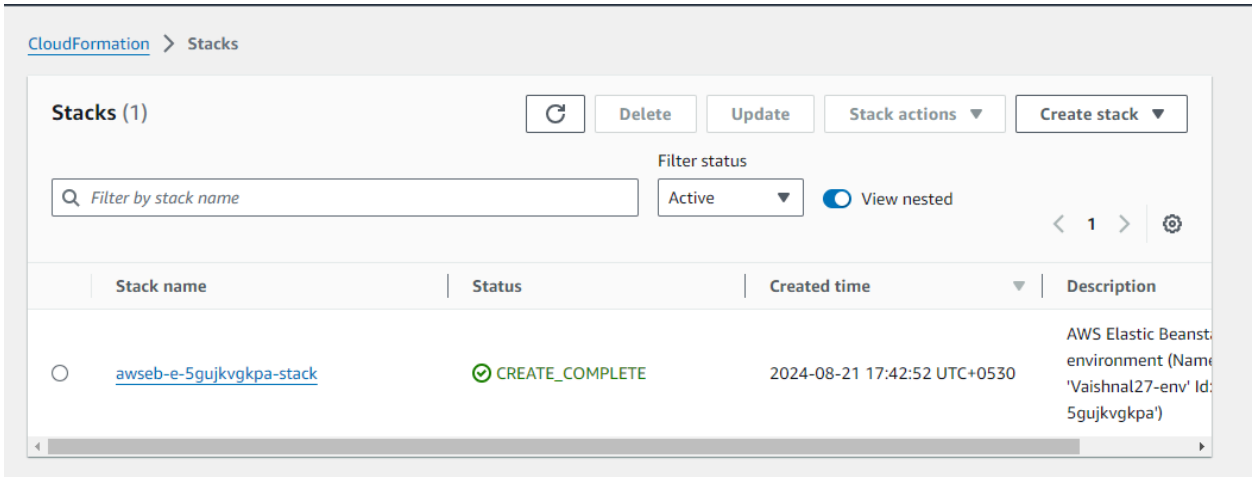
Change version

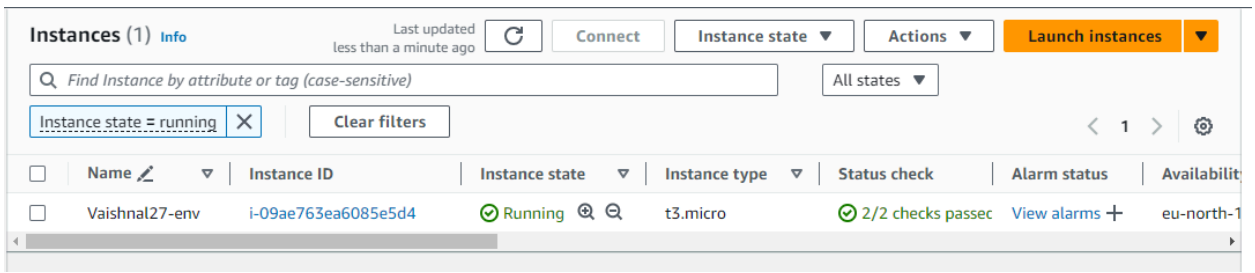
Platform
Python 3.11 running on 64bit Amazon Linux 2023/4.1.3

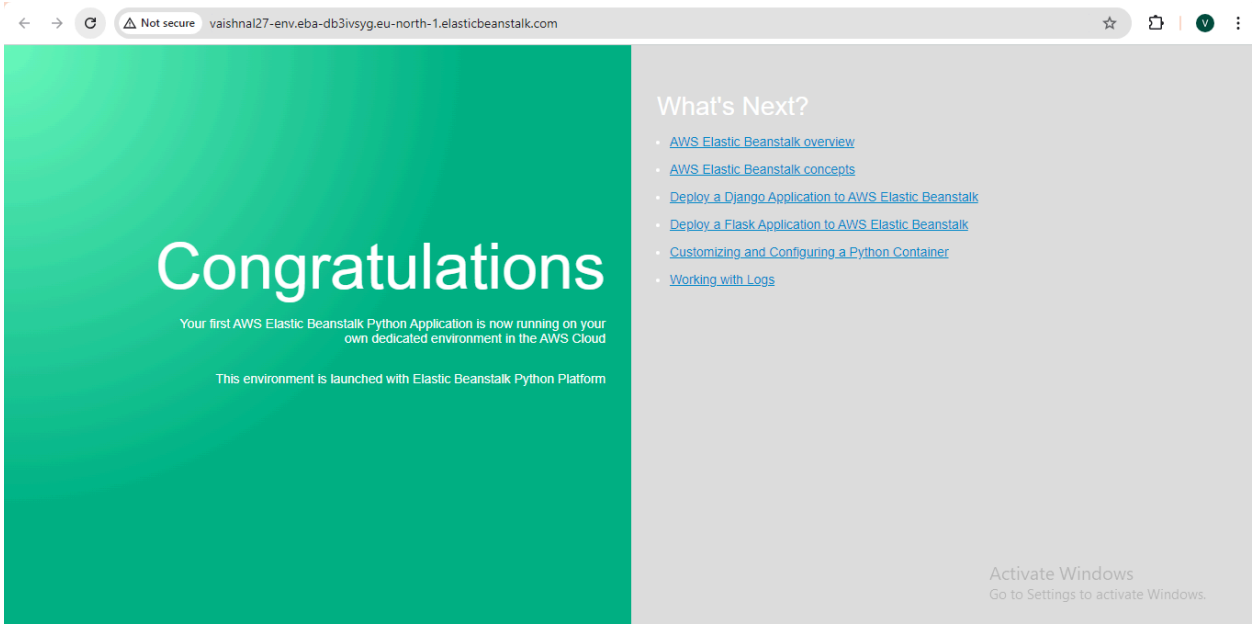
Running version
–

Platform state
Supported

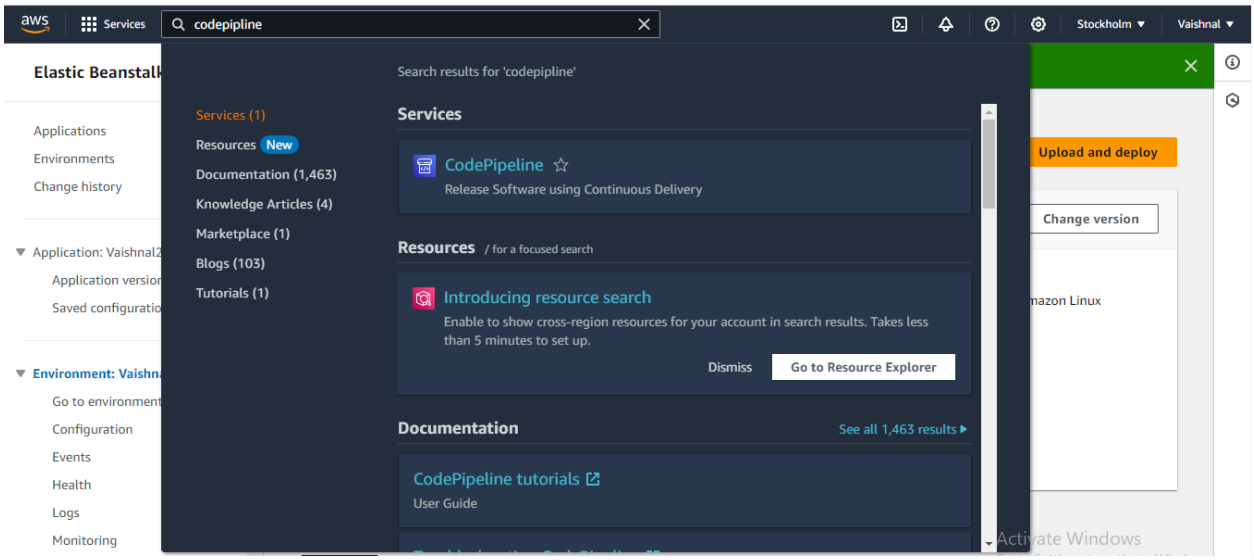

cloud formation
Search results for 'cloud formation'
Services (66)
Features (110)
Resources New
Documentation (116,209)
Knowledge Articles (1,064)
Marketplace (724)
Blogs (10,144)
Events (374)
Tutorials (26)
Services
See all 66 results
CloudFormation
Create and Manage Resources with Templates
Application Composer
Visually design and build modern applications quickly
Athena
Serverless interactive analytics service
AWS Supply Chain
Supply chain management application to manage your supply chain systems.
Features
See all 110 results
IaC Generator

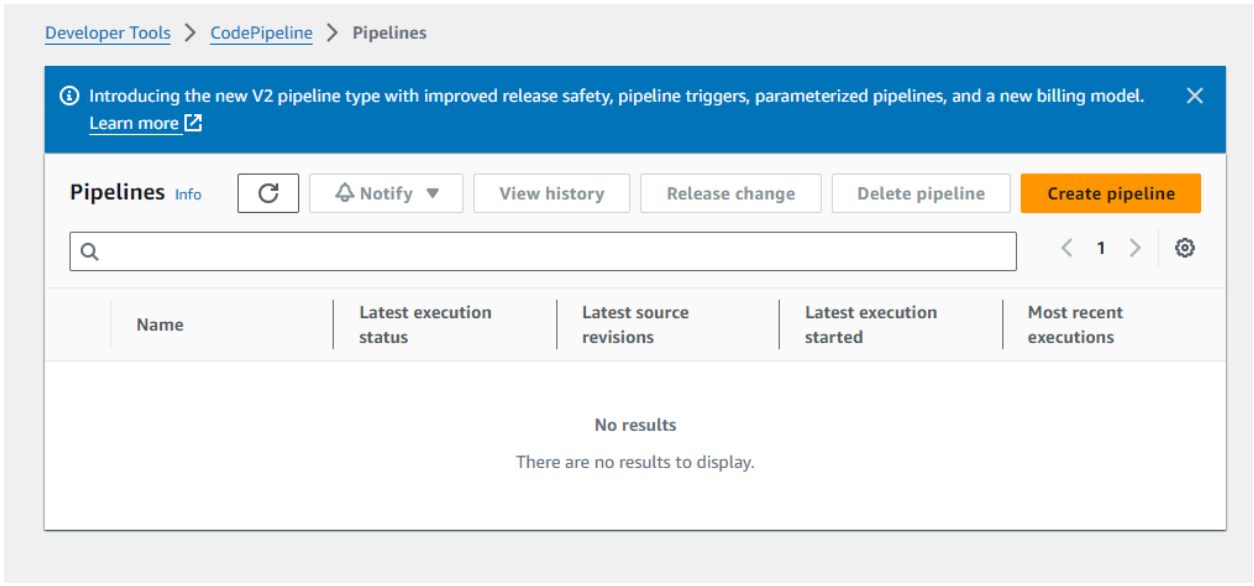

CloudFormation
Stacks
Stacks (1)
Delete
Update
Stack actions
Create stack
Filter status
Filter by stack name
Active
View nested
Stack name
Status
Created time
Description
awseb-e-5gujkvgkpa-stack
CREATE_COMPLETE
2024-08-21 17:42:52 UTC+0530

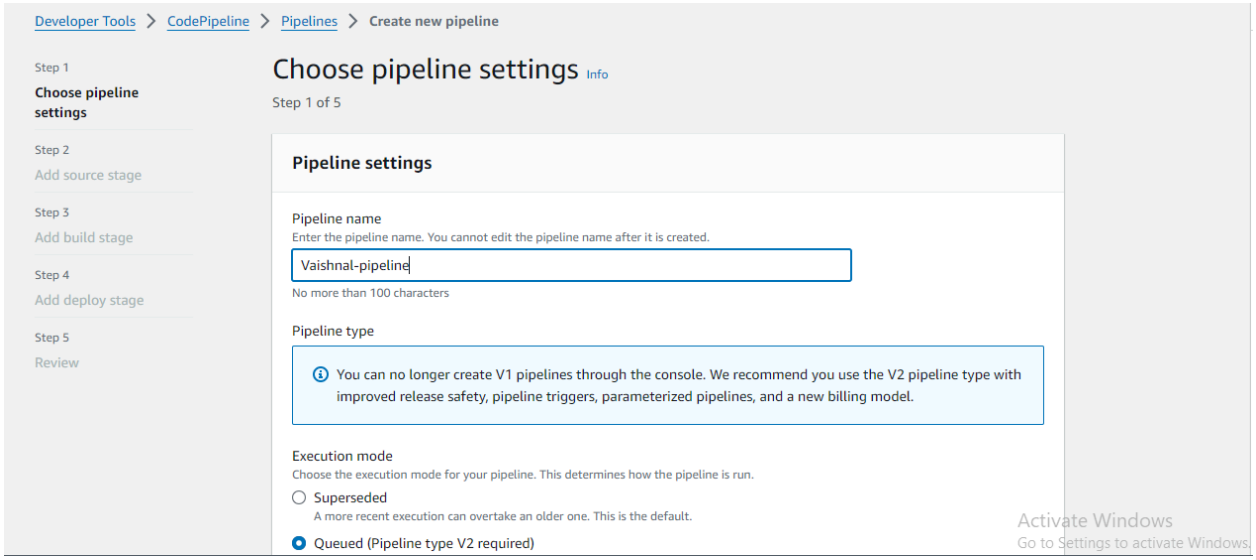

Instances (1) Info
Last updated
less than a minute ago
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Instance state = running
Clear filters
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Vaishnal27-env
i-09ae763ea6085e5d4
Running
t3.micro
2/2 checks passed
View alarms

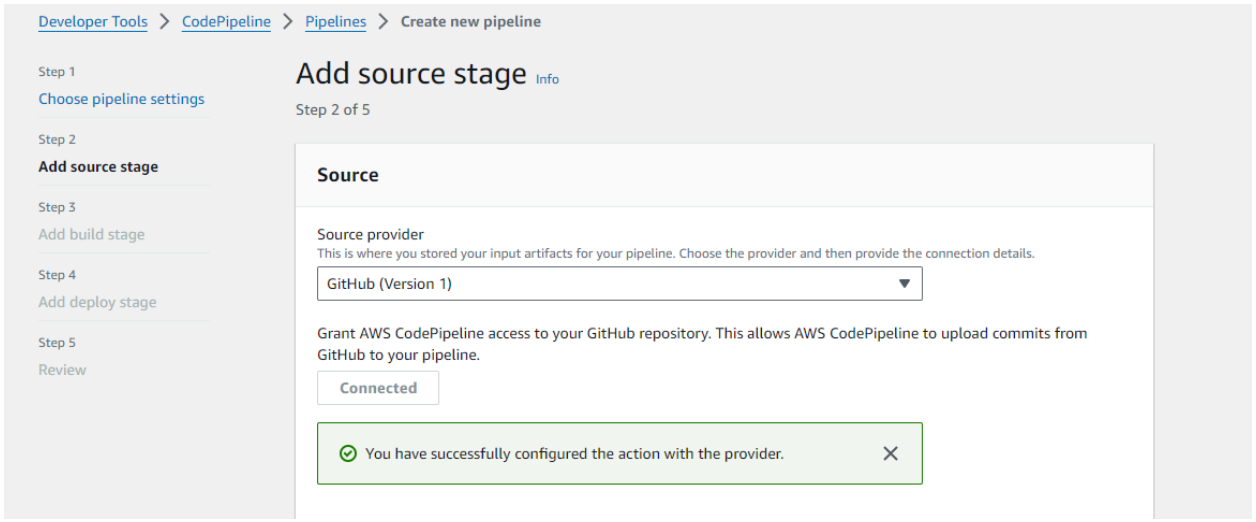

Not secure
vaishnal27-env.eba-db3ivsyg.eu-north-1.elasticbeanstalk.com
Congratulations
Your first AWS Elastic Beanstalk Python Application is now running on your own dedicated environment in the AWS Cloud
This environment is launched with Elastic Beanstalk Python Platform
What's Next?
AWS Elastic Beanstalk 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
Activate Windows
Go to Settings to activate Windows.

Code Deployment using Codepipeline






Developer Tools > CodePipeline > Pipelines > Create new pipeline
Step 1
Choose pipeline settings
Step 2
Add source stage
Step 3
Add build stage
Step 4
Add deploy stage
Step 5
Review
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.
Vaishnal-pipeline
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.
Execution mode
Choose the execution mode for your pipeline. This determines how the pipeline is run.
Superseded
A more recent execution can overtake an older one. This is the default.
Queued (Pipeline type V2 required)
Activate Windows
Go to Settings to activate Windows.


Developer Tools > CodePipeline > Pipelines > Create new pipeline
Step 1
Choose pipeline settings
Step 2
Add source stage
Step 3
Add build stage
Step 4
Add deploy stage
Step 5
Review
Add source stage Info
Step 2 of 5
Source
Source provider
This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.
GitHub (Version 1)
Grant AWS CodePipeline access to your GitHub repository. This allows AWS CodePipeline to upload commits from GitHub to your pipeline.
Connected
You have successfully configured the action with the provider.

Repository

vaishnal16/IP-lab-Exp-2

Branch

main

main

Choose a detection mode to automatically start your pipeline when a change occurs in the source code.

GitHub webhooks (recommended)
Use webhooks in GitHub to automatically start my pipeline when a change occurs

AWS CodePipeline
Use AWS CodePipeline to check periodically for changes

Cancel | Previous | Next

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

Add build stage Info

Step 3 of 5

Build - *optional*

Build provider
This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output file names.

Cancel | Previous | Skip build stage | Next

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

Add deploy stage Info

Step 4 of 5

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.

Deploy

Deploy provider
Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.

AWS Elastic Beanstalk

Region

Europe (Stockholm)

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.

Vaishnal27

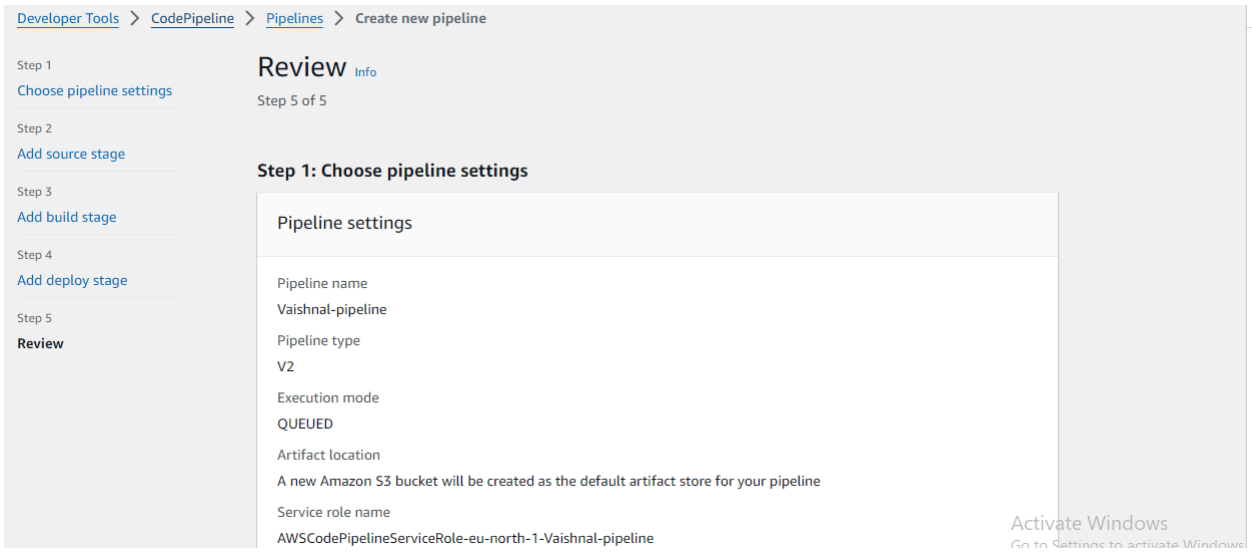
Environment name

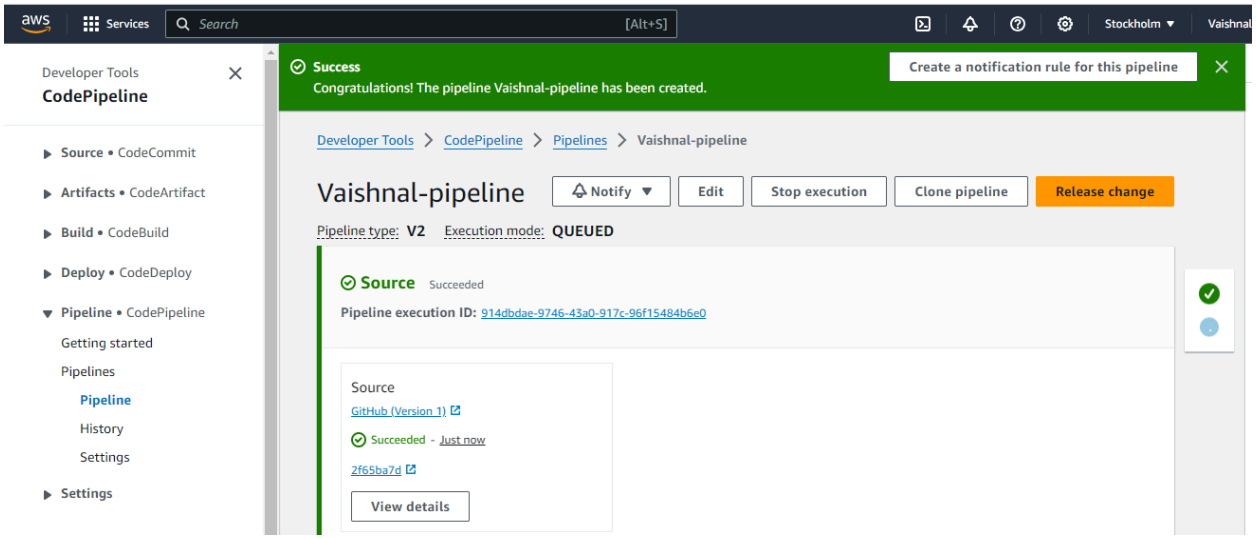
Choose an environment that you have already created in the AWS Elastic Beanstalk console. Or create an environment in the AWS Elastic Beanstalk console and then return to this task.

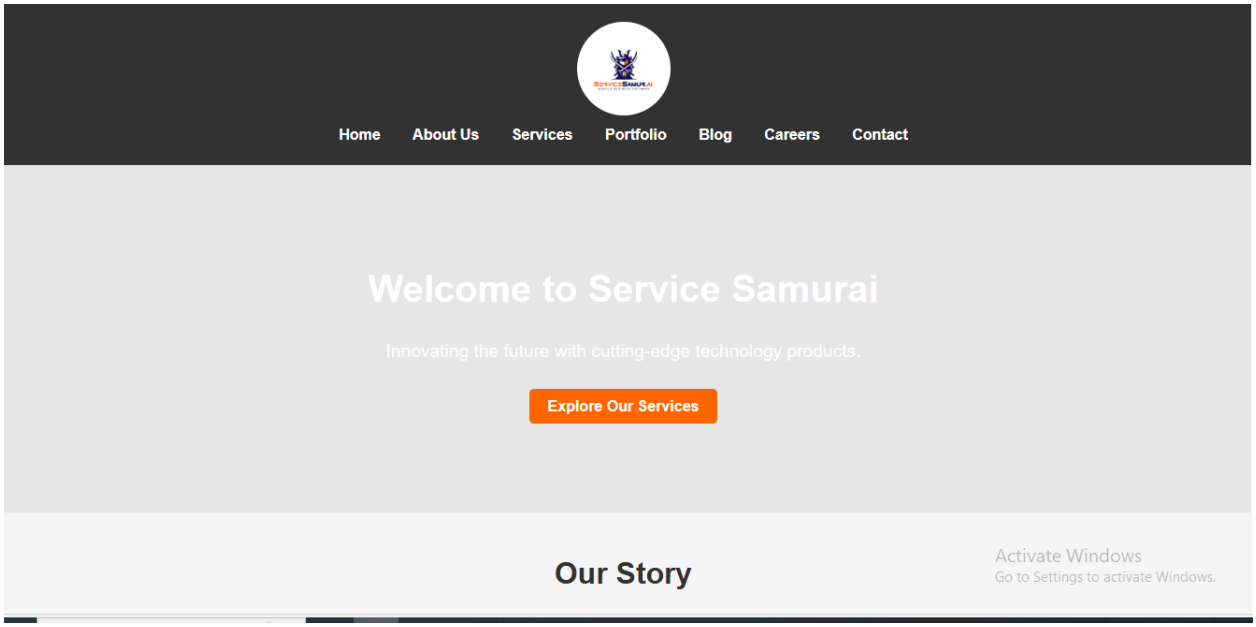
Vaishnal27-env

Vaishnal27-env

Cancel Previous Next




aws
Services
Search
[Alt+S]
Stockholm
Developer Tools
CodePipeline
Source • CodeCommit
Artifacts • CodeArtifact
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Getting started
Pipelines
Pipeline
History
Settings
Settings
Success
Congratulations! The pipeline Vaishnal-pipeline has been created.
Create a notification rule for this pipeline
Developer Tools > CodePipeline > Pipelines > Vaishnal-pipeline
Vaishnal-pipeline
Notify
Edit
Stop execution
Clone pipeline
Release change
Pipeline type: V2
Execution mode: QUEUED
Source
Succeeded
Pipeline execution ID: 914dbdae-9746-43a0-917c-96f15484b6e0
Source
GitHub (Version 1)
Succeeded - Just now
2f65ba7d
View details


Home
About Us
Services
Portfolio
Blog
Careers
Contact
Welcome to Service Samurai
Innovating the future with cutting-edge technology products.
Explore Our Services
Our Story
Activate Windows
Go to Settings to activate Windows.