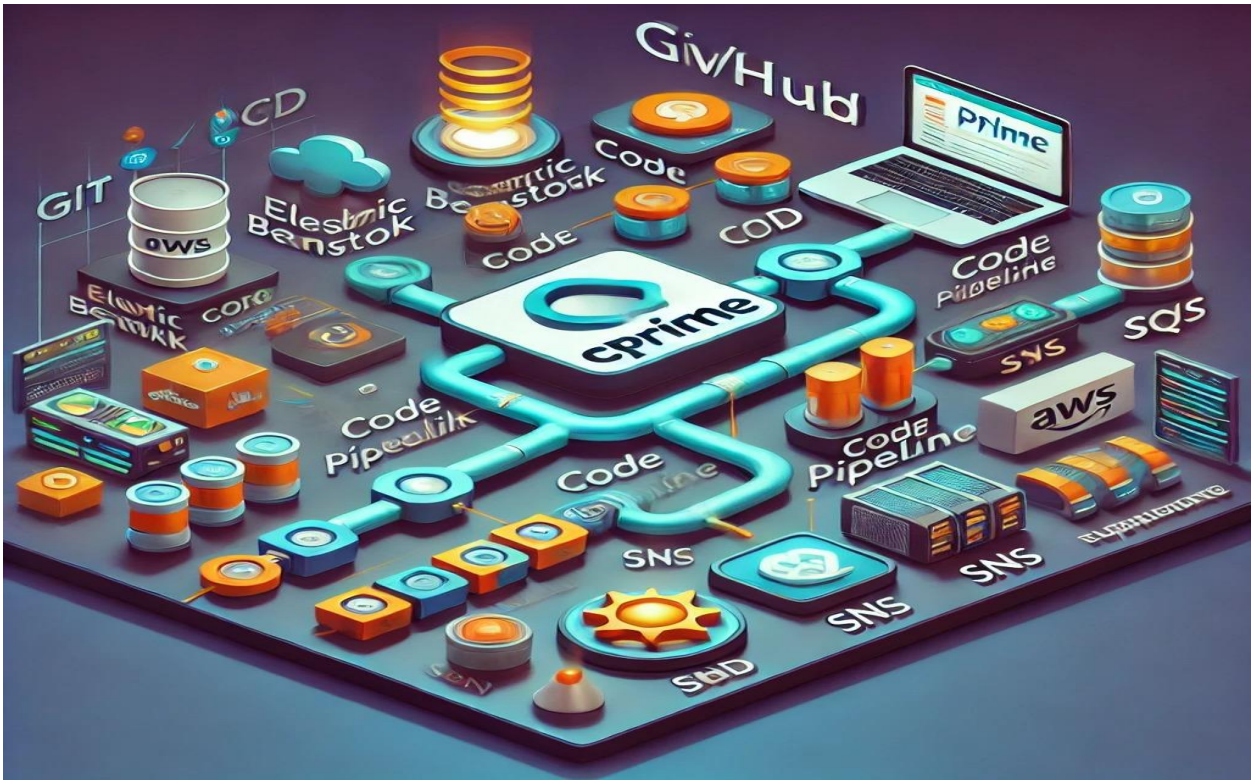
# CI/CD Pipeline with Manual Approval Using AWS and GitHub Integration

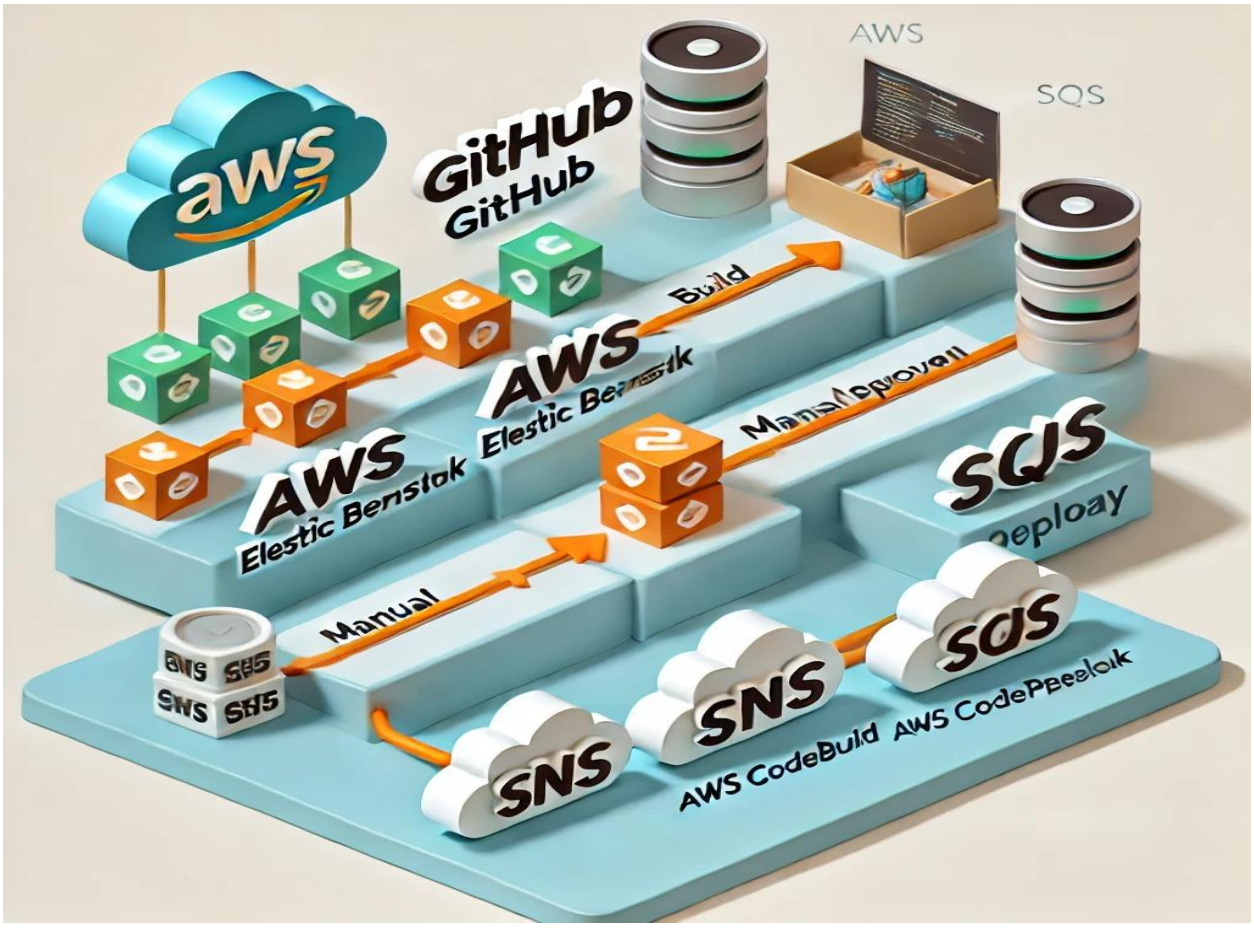



**NAME : T MANOJ**

**EMP ID : LYAKE2KHS**

## Overview

This project demonstrates how to set up a CI/CD pipeline using GitHub, AWS Elastic Beanstalk, AWS CodeBuild, AWS CodePipeline, SNS, and SQS. The pipeline automatically detects code changes, builds the application, and deploys it to AWS Elastic Beanstalk. A manual approval stage is integrated to ensure critical changes are reviewed before deployment, enhancing control and security.



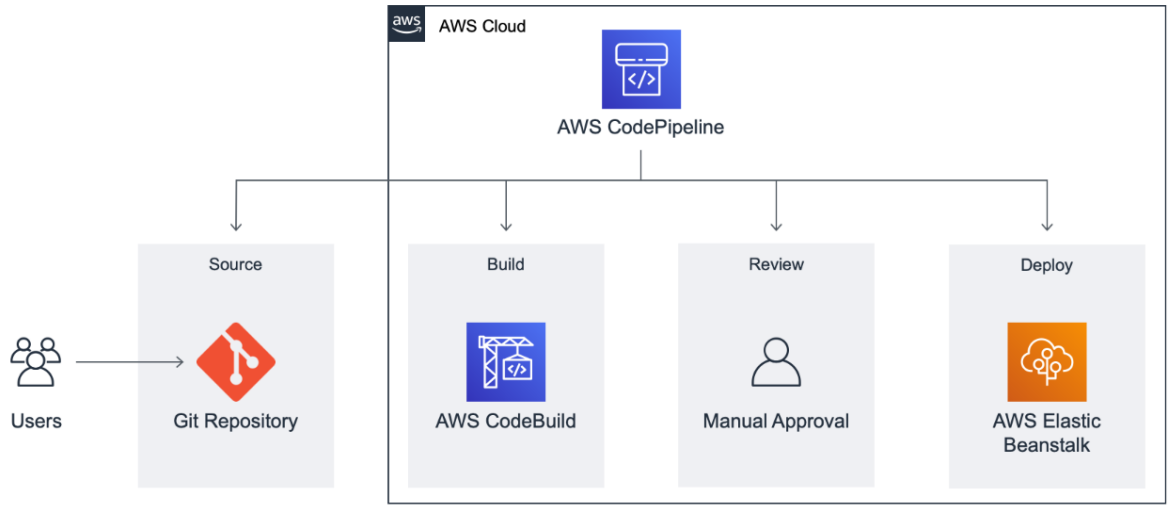
## Introduction

Modern software development requires rapid and reliable deployment of applications. Continuous Integration (CI) and Continuous Delivery (CD) pipelines automate this process, reducing manual errors, improving productivity, and ensuring quick delivery of new features.

In this project, you will:

- Host your code in GitHub for easy access and version control.
- Deploy a sample Node.js application using AWS Elastic Beanstalk, which abstracts the underlying infrastructure management.
- Use AWS CodeBuild to compile and test the code automatically.
- Configure AWS CodePipeline to orchestrate the build and deployment processes.
- Add a manual approval step using SNS and SQS to ensure that deployments are reviewed before going live.

**CI/CD ACHITECTURE :**



### Prerequisites

- GitHub Account
- Git Installation
- AWS Account
- AWS CLI (Configured)
- Text Editor/IDE (VS Code)
- Node.js Runtime
- Elastic Beanstalk Access
- CodeBuild Knowledge
- CodePipeline Integration
- IAM Roles & Permissions
- SNS Topic Setup
- SQS Queue Setup
- OAuth GitHub Integration
- Internet Connection
- Modern Browser (Chrome, Firefox)
- Basic CI/CD Understanding

## Project Phases :

- Setting Up a GitHub Repository
- Deploy Web App using AWS Elastic Beanstalk
- Create Build Project using AWS CodeBuild
- Set Up Continuous Delivery Pipeline using AWS CodePipeline
- Adding Manual Approval with SNS & SQS

# Setting Up a GitHub Repository

## 🎯 Objective:

- Create a GitHub repository to store your application code.
- Prepare the repository for integration with AWS services.

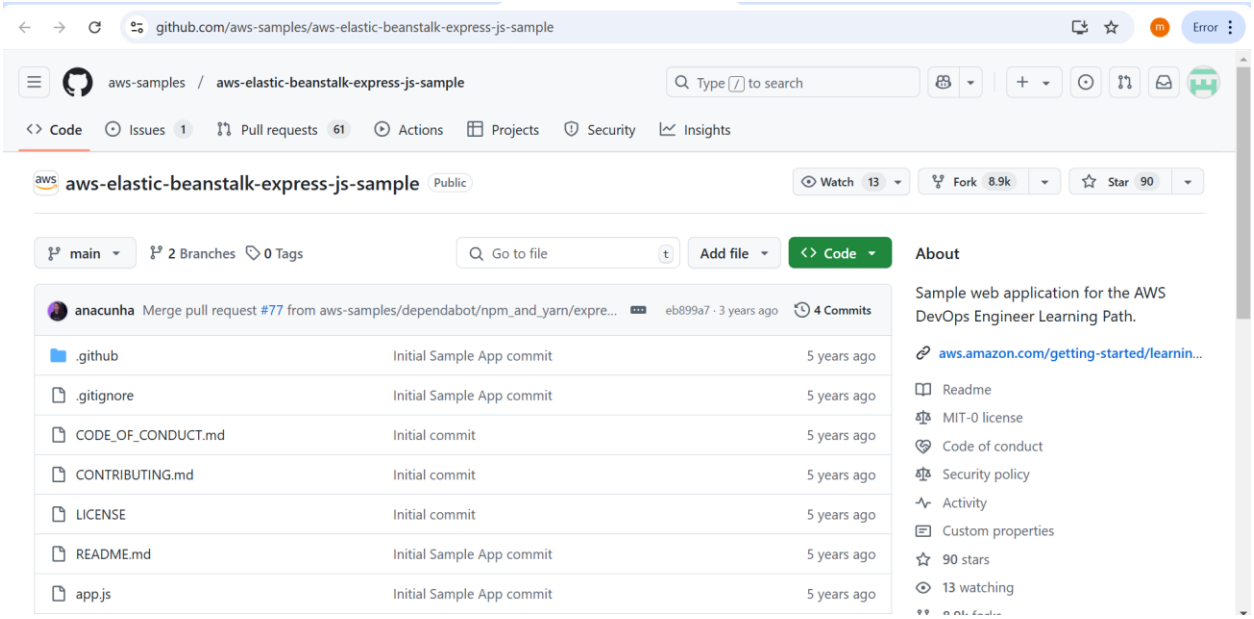

## 📌 Terminologies:

1. GitHub: A web-based platform for version control using Git.
2. Git: Tracks and manages code changes.
3. Repository (Repo): Storage space for your project, including code and version history.
4. Fork: Copying an existing repo to your GitHub account.
5. Clone: Copying the repo from GitHub to your local machine.
6. Commit: Saving changes to the version history.
7. Push: Sending local commits to the GitHub repository.

## ⚙️ Steps to Set Up GitHub Repo:

### *Step 1: Fork a Sample Repository*

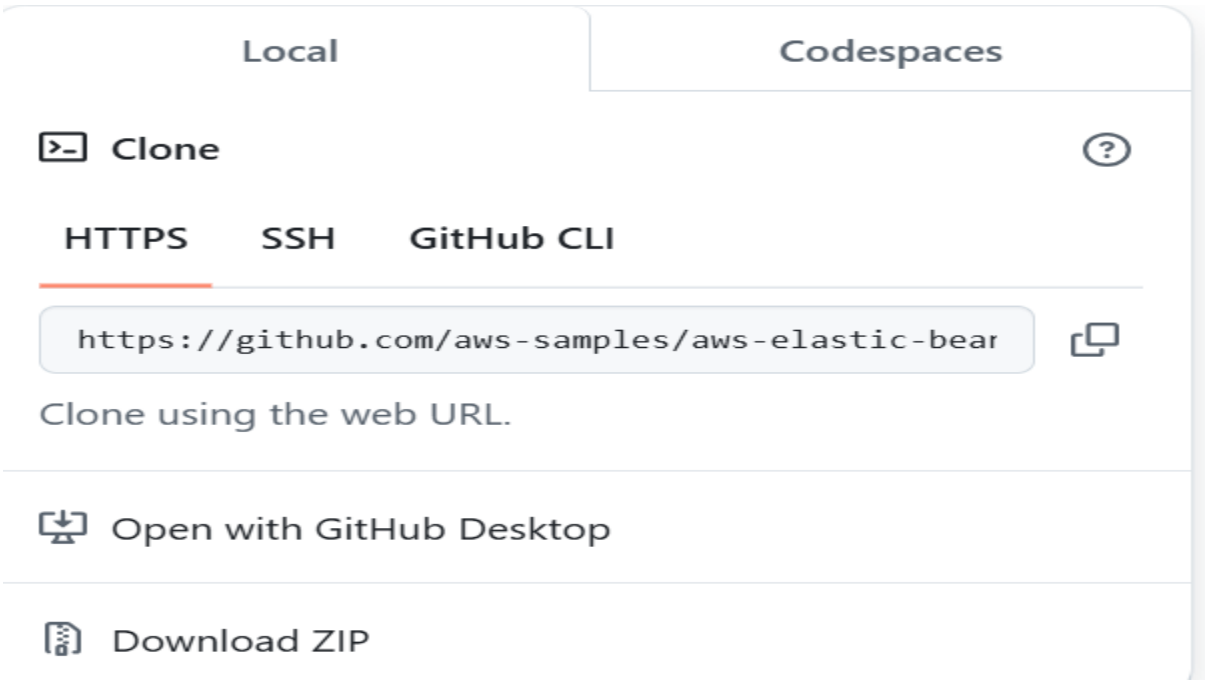
- **Navigate to GitHub and sign in.**
- **Search for the repository: `aws-elastic-beanstalk-express-js-sample`.**
- **Click the Fork button (top-right corner) to copy it to your account.**

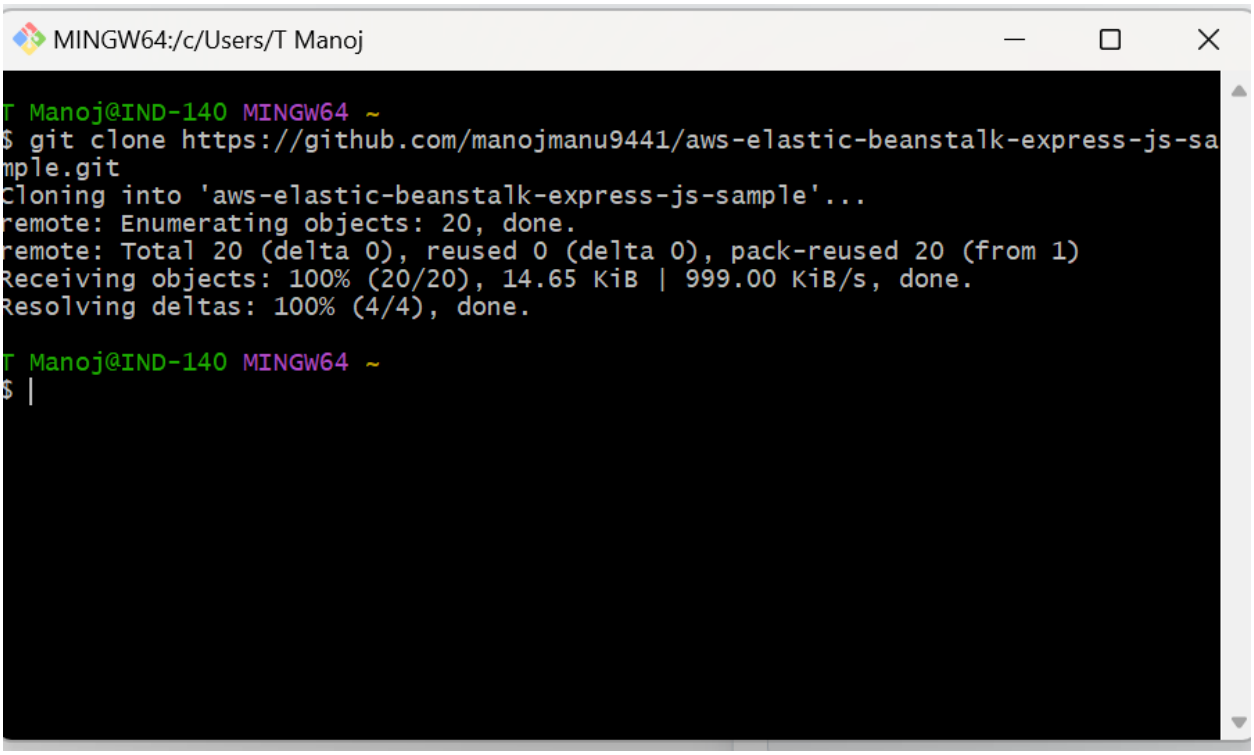



### *Step 2: Clone the Repository Locally*

1. Go to your forked repository on GitHub.
2. Click the Code button and copy the HTTPS URL.
3. Open your terminal and run:

```
git clone  aws-elastic-beanstalk-express-js-sample
```



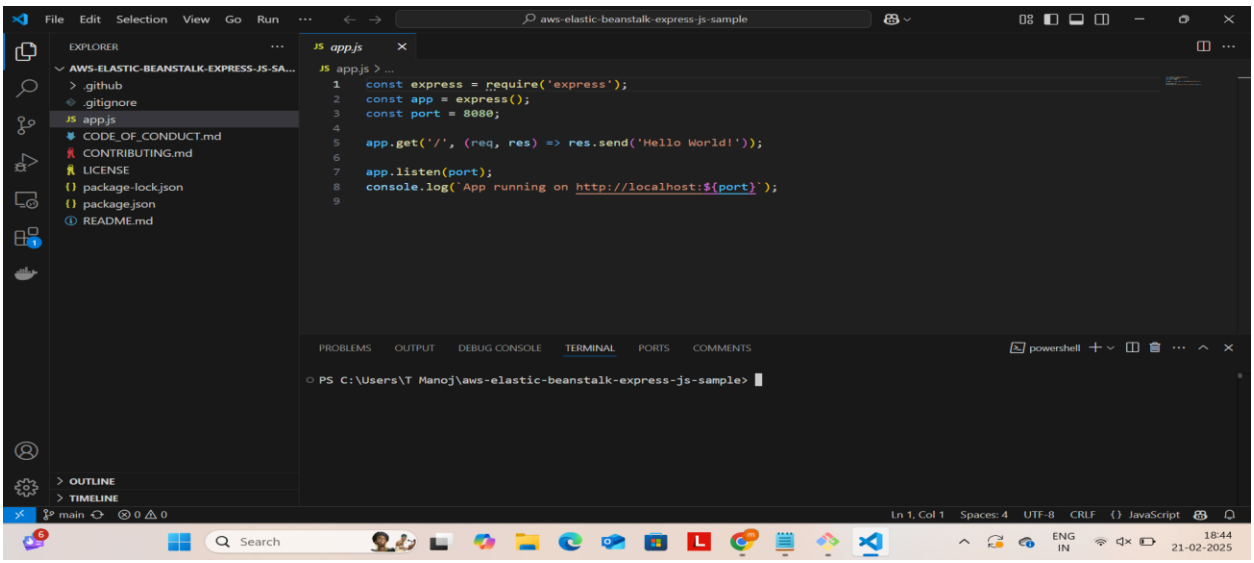




### *Step 3: Make Code Changes*

1. Open the project in Visual Studio Code (or your preferred editor).

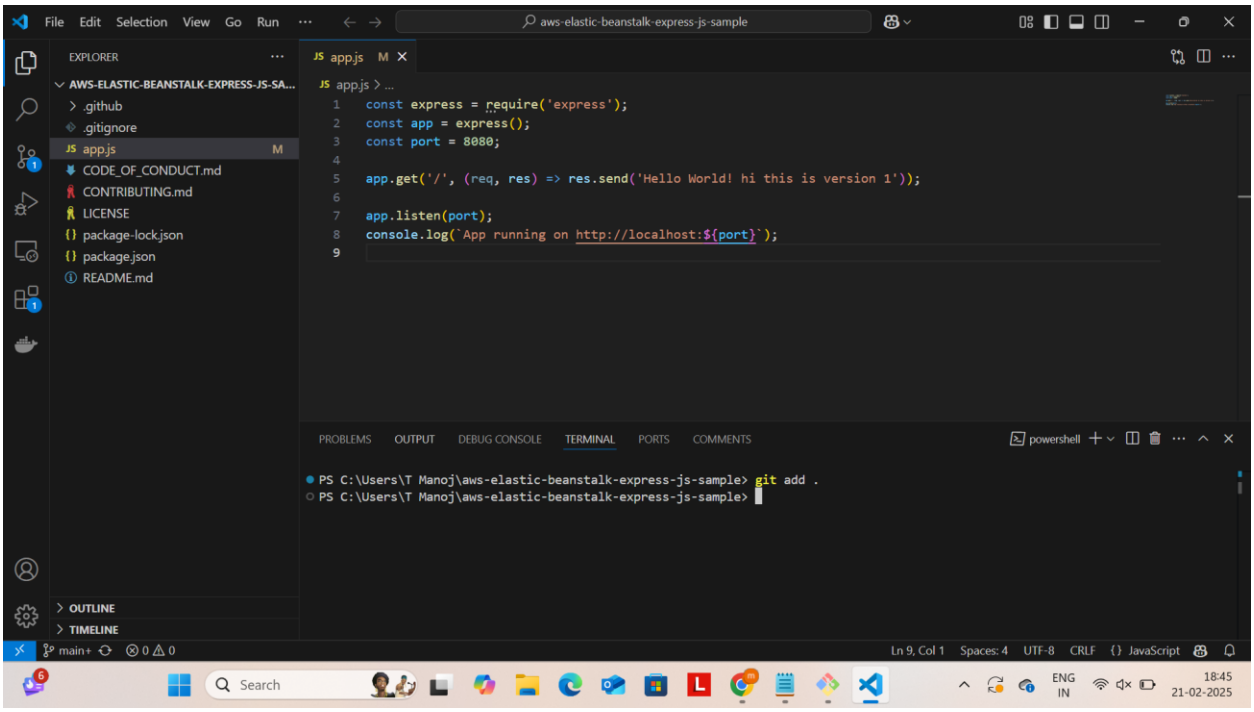
2. Modify app.js by adding:

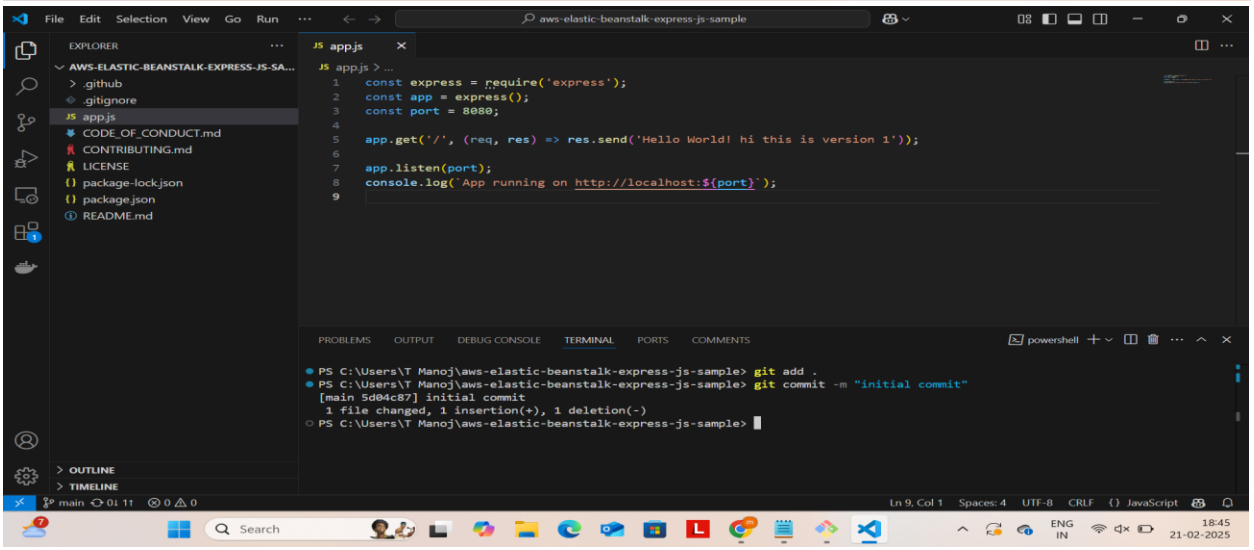
```
console.log("Hello World! This is the version 1");
```

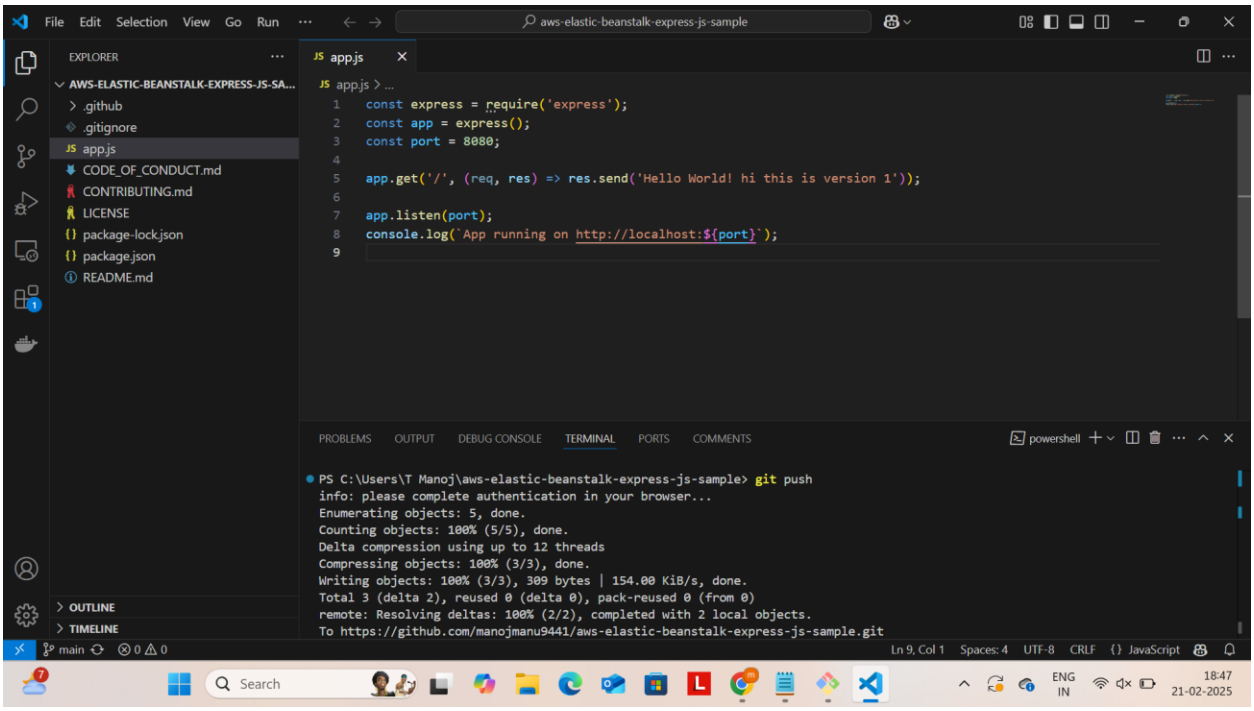



### *Step 4: Commit and Push Changes*

```
git add .
git commit -m "Intial Commit"
git push
```


File Edit Selection View Go Run
aws-elastic-beanstalk-express-js-sample
EXPLORER
AWS-ELASTIC-BEANSTALK-EXPRESS-JS-SA...
.github
.gitignore
app.js M
CODE_OF_CONDUCT.md
CONTRIBUTING.md
LICENSE
package-lock.json
package.json
README.md
app.js M
app.js > ...
1 const express = require('express');
2 const app = express();
3 const port = 8080;
4
5 app.get('/', (req, res) => res.send('Hello World! hi this is version 1'));
6
7 app.listen(port);
8 console.log(`App running on http://localhost:${port}`);
9
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
powershell
PS C:\Users\T Manoj\aws-elastic-beanstalk-express-js-sample> git add .
PS C:\Users\T Manoj\aws-elastic-beanstalk-express-js-sample>
OUTLINE
TIMELINE
main+
Ln 9, Col 1 Spaces: 4 UTF-8 CRLF {} JavaScript
Search
ENG IN
18:45 21-02-2025


File Edit Selection View Go Run
aws-elastic-beanstalk-express-js-sample
EXPLORER
AWS-ELASTIC-BEANSTALK-EXPRESS-JS-SA...
.github
.gitignore
app.js
CODE_OF_CONDUCT.md
CONTRIBUTING.md
LICENSE
package-lock.json
package.json
README.md
app.js
app.js > ...
1 const express = require('express');
2 const app = express();
3 const port = 8080;
4
5 app.get('/', (req, res) => res.send('Hello World! hi this is version 1'));
6
7 app.listen(port);
8 console.log(`App running on http://localhost:${port}`);
9
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
powershell
PS C:\Users\T Manoj\aws-elastic-beanstalk-express-js-sample> git add .
PS C:\Users\T Manoj\aws-elastic-beanstalk-express-js-sample> git commit -m "initial commit"
[main 5d04c87] initial commit
1 file changed, 1 insertion(+), 1 deletion(-)
PS C:\Users\T Manoj\aws-elastic-beanstalk-express-js-sample>
OUTLINE
TIMELINE
main 0↓ 1↑
Ln 9, Col 1 Spaces: 4 UTF-8 CRLF {} JavaScript
Search
ENG IN
18:45 21-02-2025




### *Step 5: Verify Changes*

- Go to GitHub → Your Repository → Check the `app.js` file for the update.


# Deploy Web App using AWS Elastic Beanstalk

## Objective:

- Deploy the Node.js web app using AWS Elastic Beanstalk.

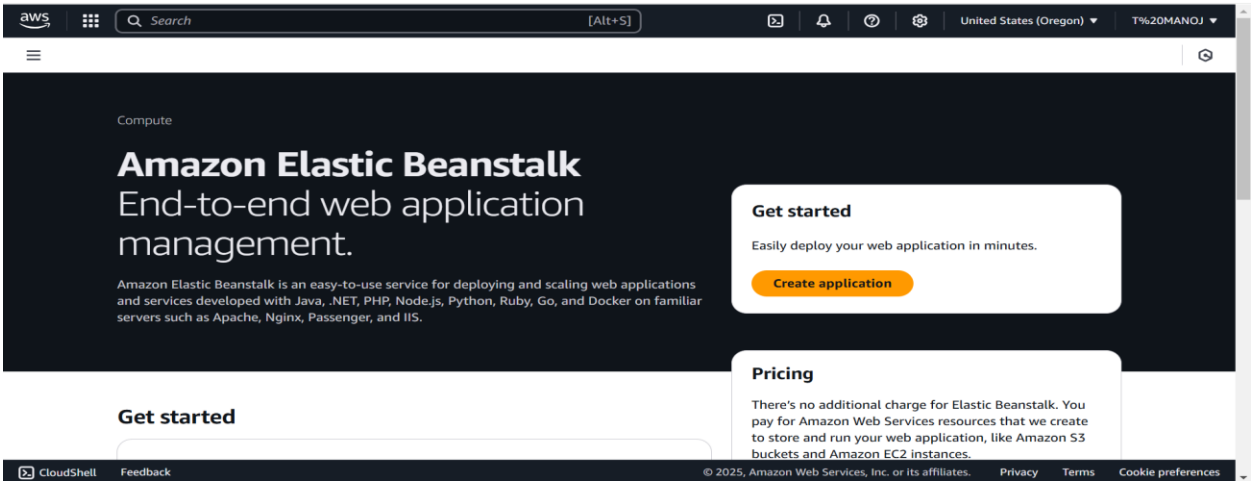

## Terminologies:

1. Elastic Beanstalk (EB): Platform-as-a-service (PaaS) that deploys apps without managing infrastructure.
2. Environment: Set of AWS resources (like EC2, S3, Load Balancers) to run your application.
3. Platform: Runtime environment (Node.js, Python, etc.) for the app.
4. EC2 Instance: Virtual machine running your application.

## Steps to Deploy Using Elastic Beanstalk:

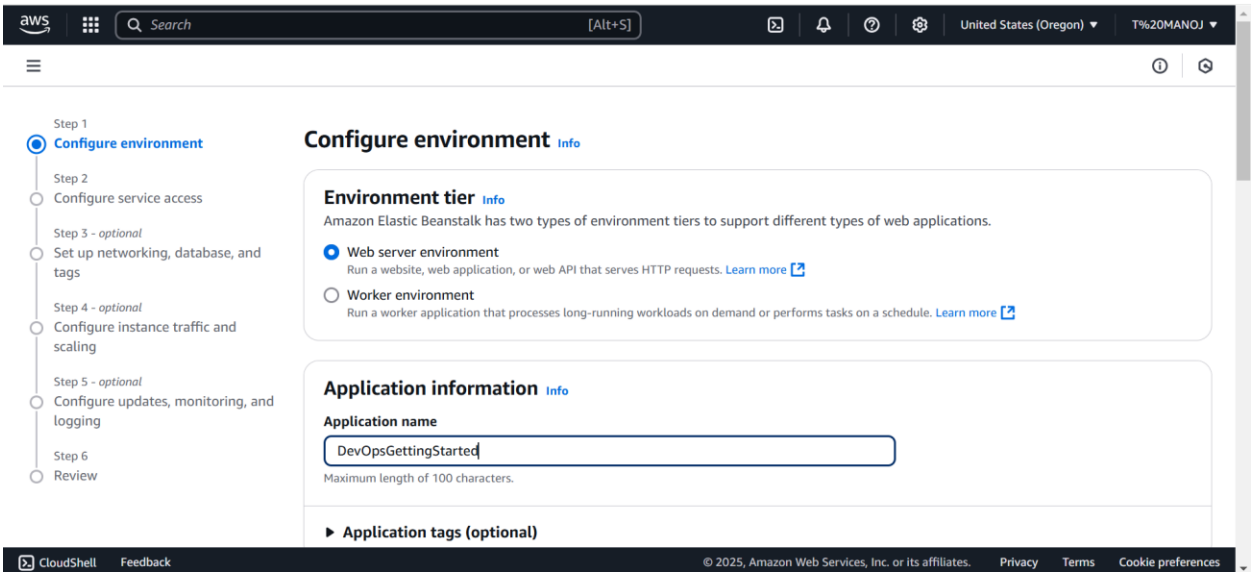
### *Step 1: Open Elastic Beanstalk Console*

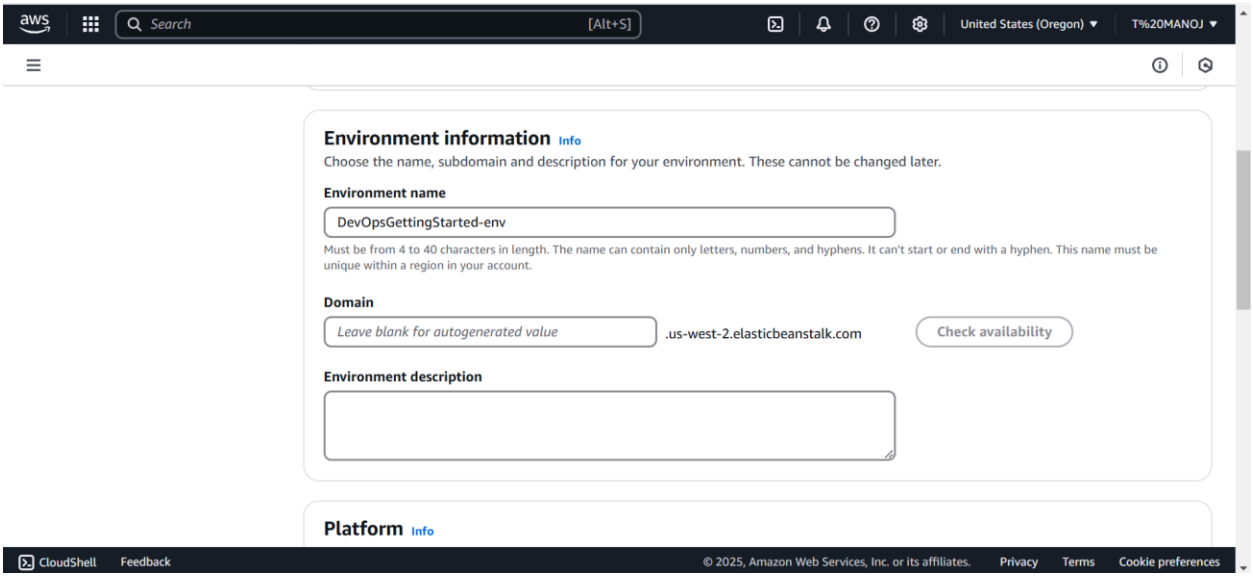
- Sign in to the AWS Management Console.
- Search and select Elastic Beanstalk.

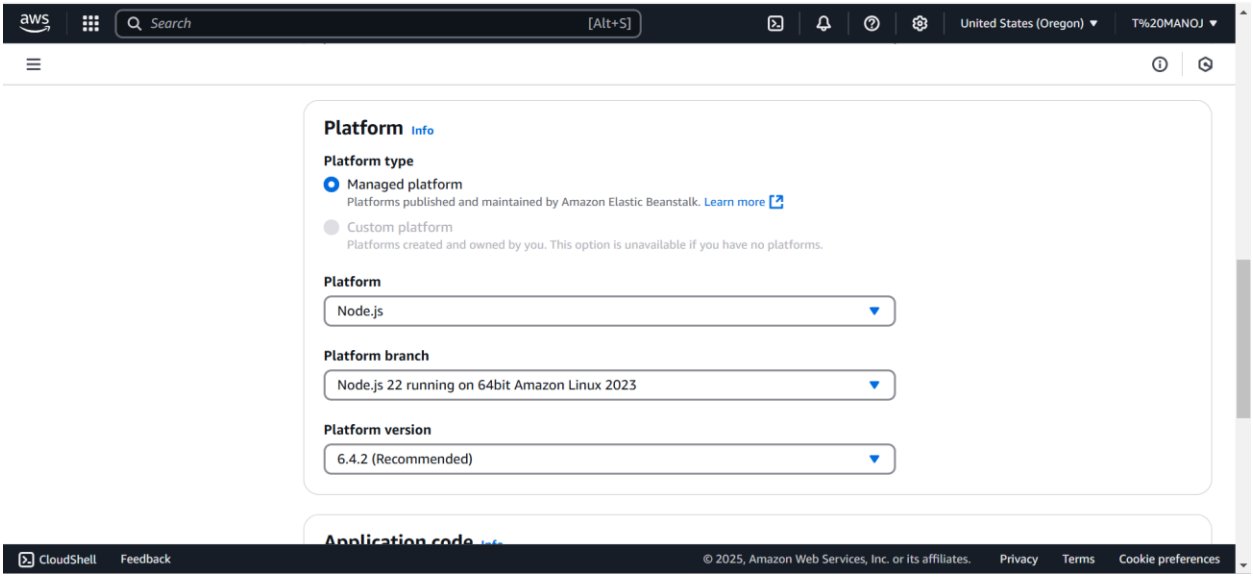


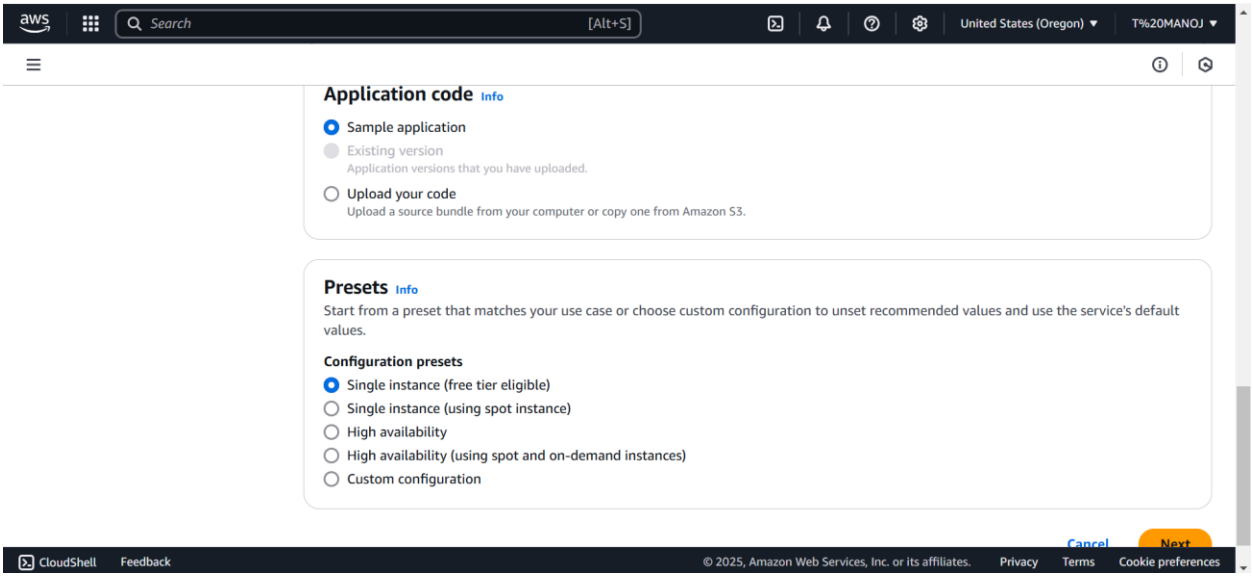

### *Step 2: Create a New Application*

1. Click Create Application.
2. Application Name: `DevOpsGettingStarted`.
3. Platform: Choose Node.js.
4. Application Code: Select Sample application.
5. Environment: Select Single instance (free tier eligible).
6. Click Next.


aws
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
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
DevOpsGettingStarted
Maximum length of 100 characters.
Application tags (optional)
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Environment information Info
Choose the name, subdomain and description for your environment. These cannot be changed later.
Environment name
DevOpsGettingStarted-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-west-2.elasticbeanstalk.com
Check availability
Environment description
Platform Info
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

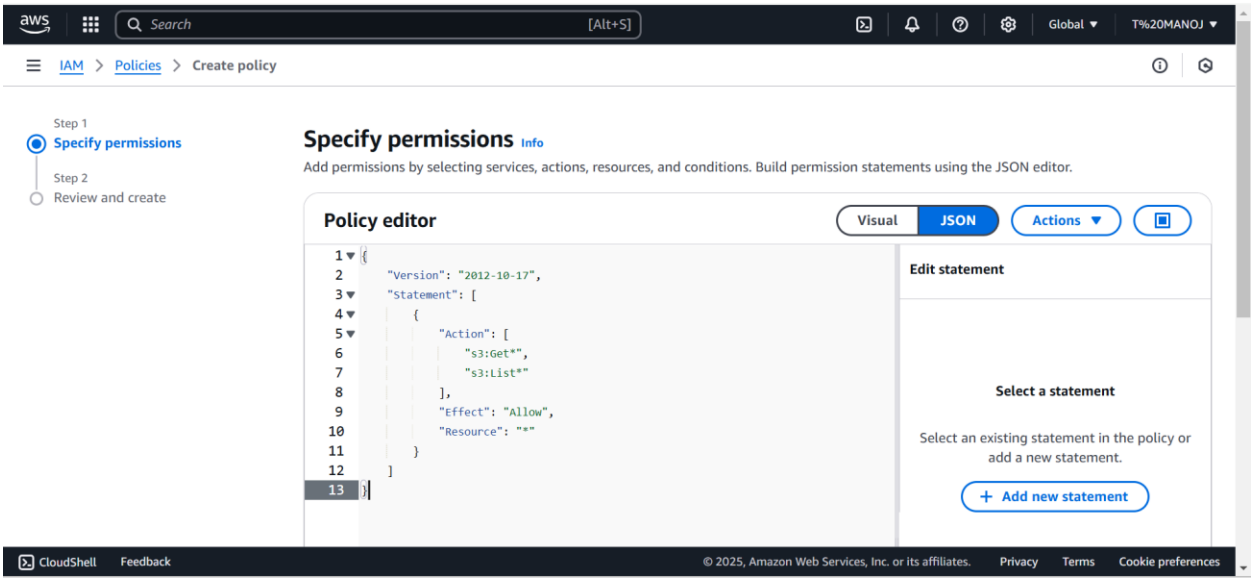


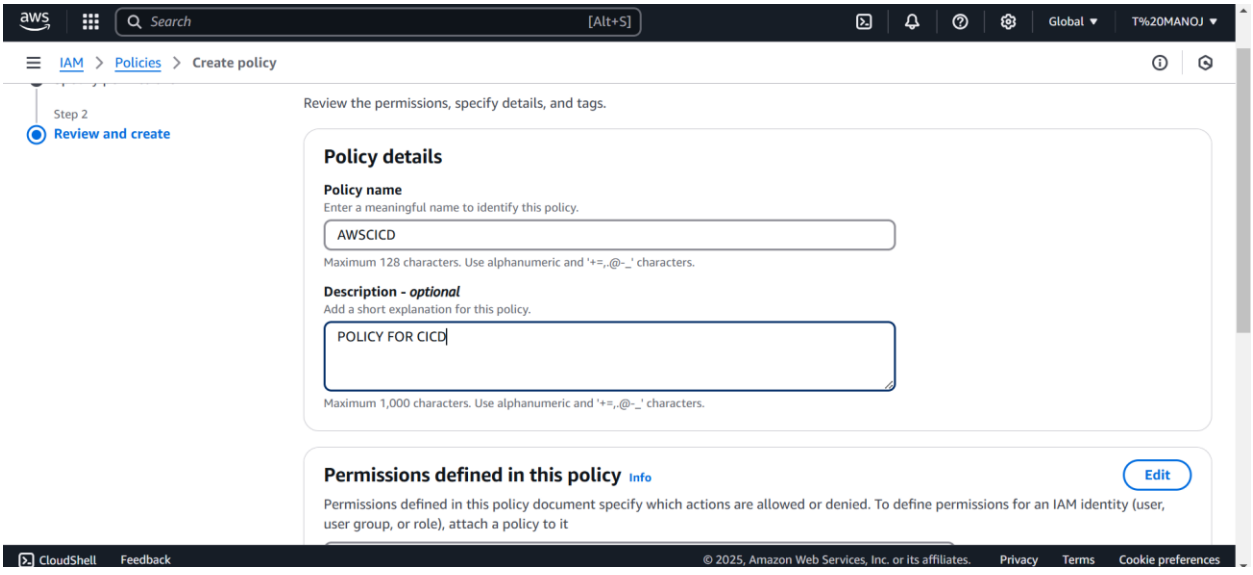



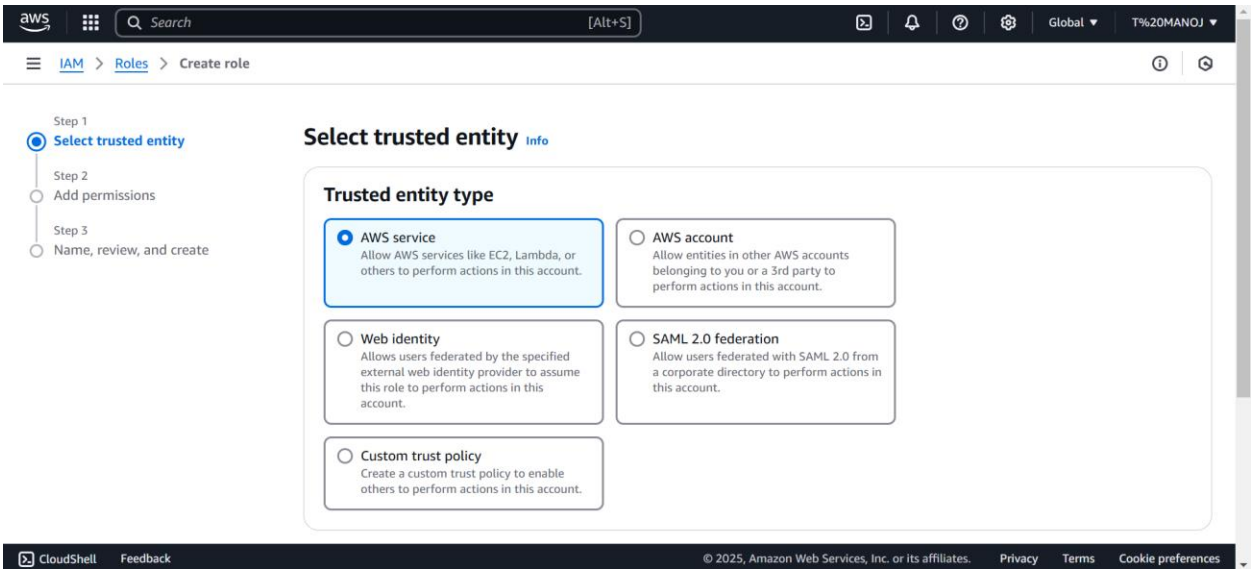

### *Step 3: Configure Service Access*

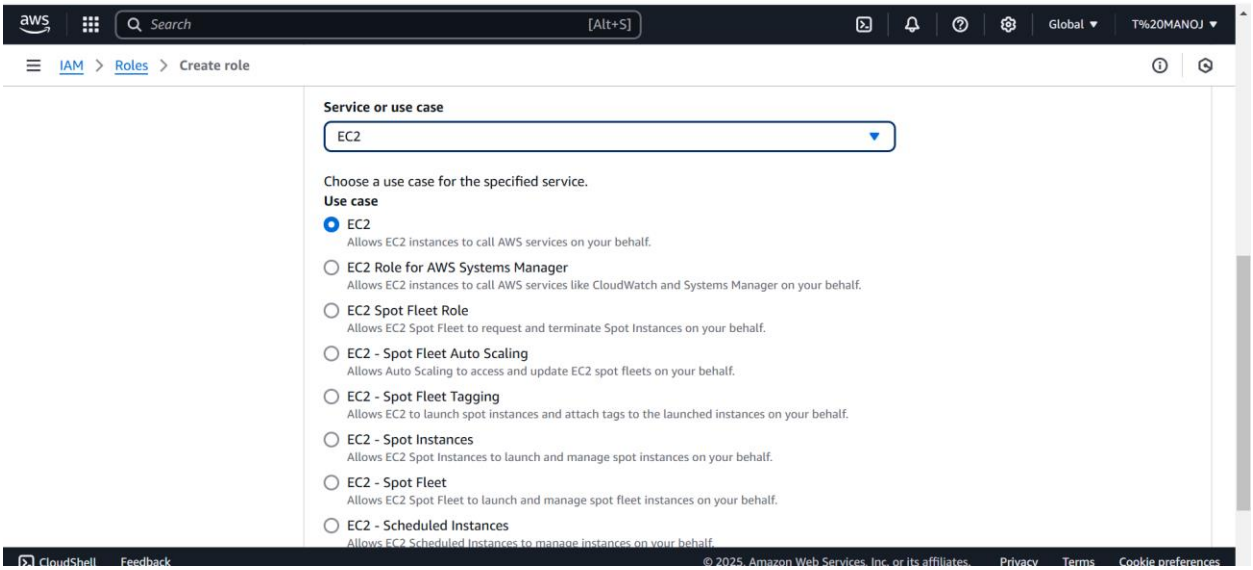
1. Service Role: Choose aws-elasticbeanstalk-service-role (create if not available).
2. EC2 Instance Profile: Select aws-elasticbeanstalk-ec2-role.
3. Key Pair: Create a new one or select existing for SSH access.

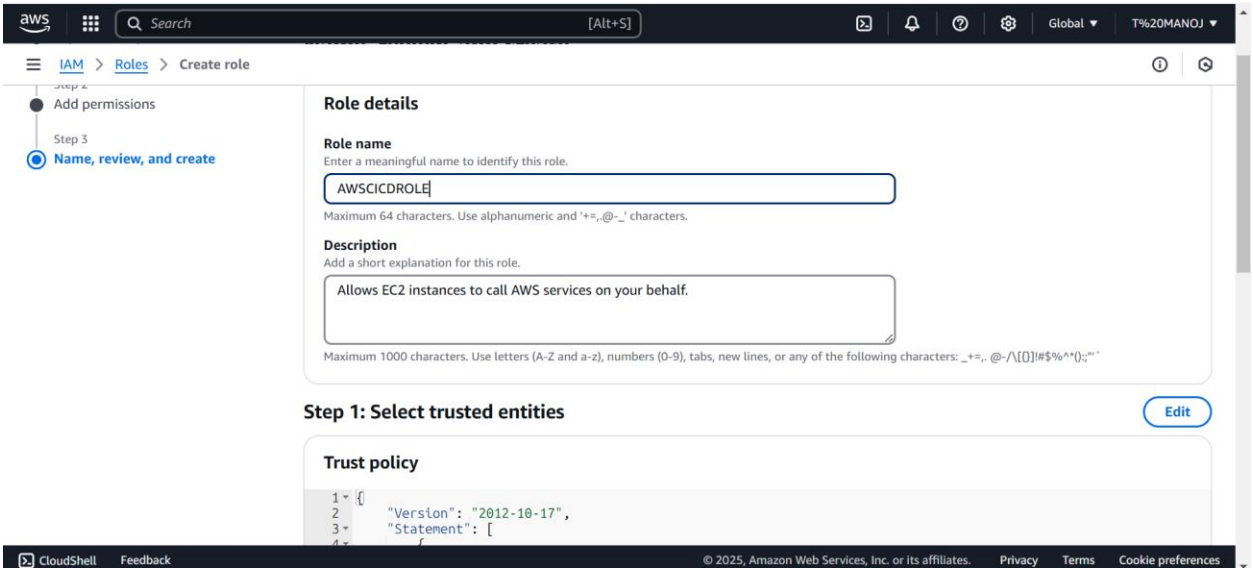

Actions
Delete
Create policy

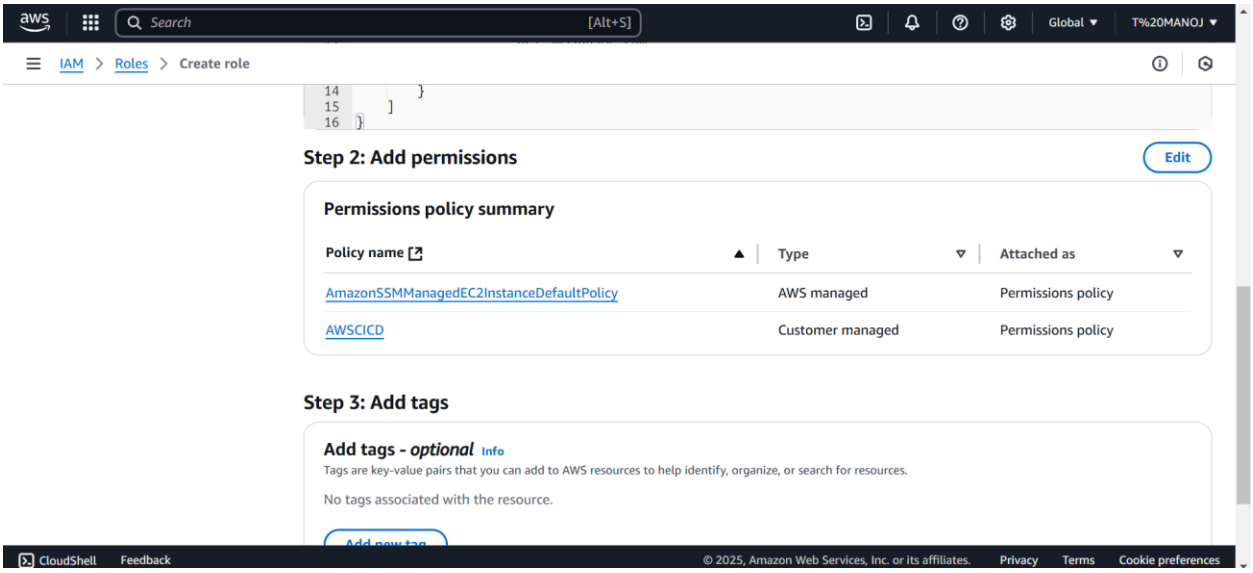

aws
Search
[Alt+S]
Global
T%20MANOJ
IAM > Policies > Create policy
Step 1
Specify permissions
Step 2
Review and create
Specify permissions Info
Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.
Policy editor
Visual
JSON
Actions
{
"Version": "2012-10-17",
"Statement": [
{
"Action": [
"s3:Get*",
"s3:List*"
],
"Effect": "Allow",
"Resource": "*"
}
]
}
Edit statement
Select a statement
Select an existing statement in the policy or add a new statement.
+ Add new statement
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Search
[Alt+S]
Global
T%20MANOJ
IAM > Policies > Create policy
Step 2
Review and create
Review the permissions, specify details, and tags.
Policy details
Policy name
Enter a meaningful name to identify this policy.
AWSCICD
Maximum 128 characters. Use alphanumeric and '+=,.@-_' characters.
Description - optional
Add a short explanation for this policy.
POLICY FOR CICD
Maximum 1,000 characters. Use alphanumeric and '+=,.@-_' characters.
Permissions defined in this policy Info
Edit
Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Search
[Alt+S]
Global
T%20MANOJ
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.
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

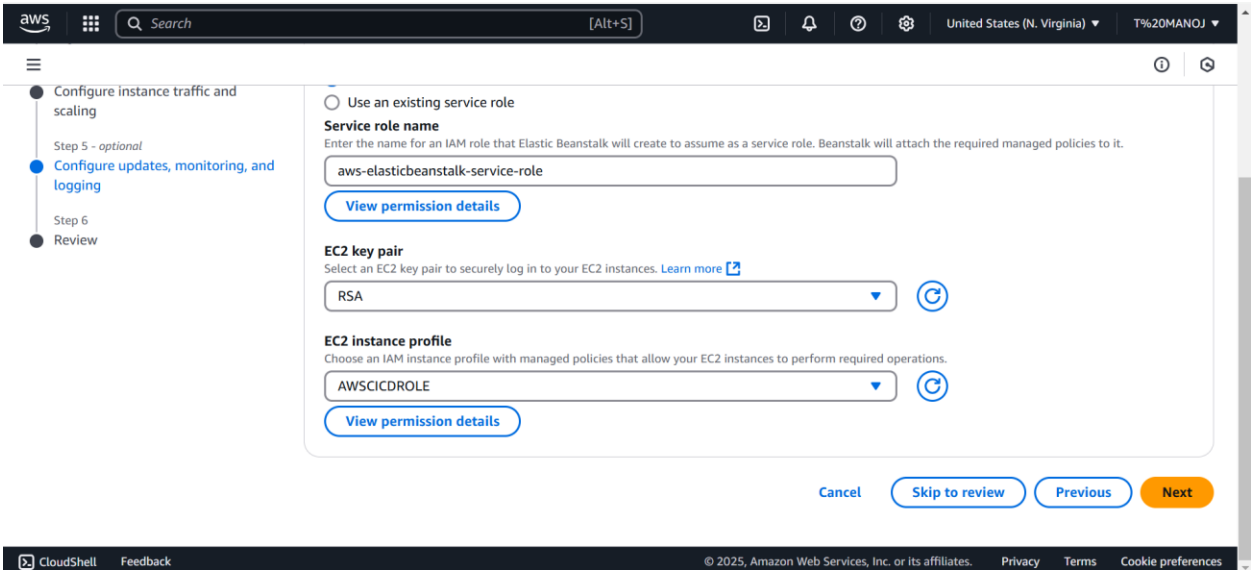

Search
[Alt+S]
Global
T%20MANOJ
IAM > Roles > Create role
Service or use case
EC2
Choose a use case for the specified service.
Use case
EC2
Allows EC2 instances to call AWS services on your behalf.
EC2 Role for AWS Systems Manager
Allows EC2 instances to call AWS services like CloudWatch and Systems 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.
EC2 - Scheduled Instances
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

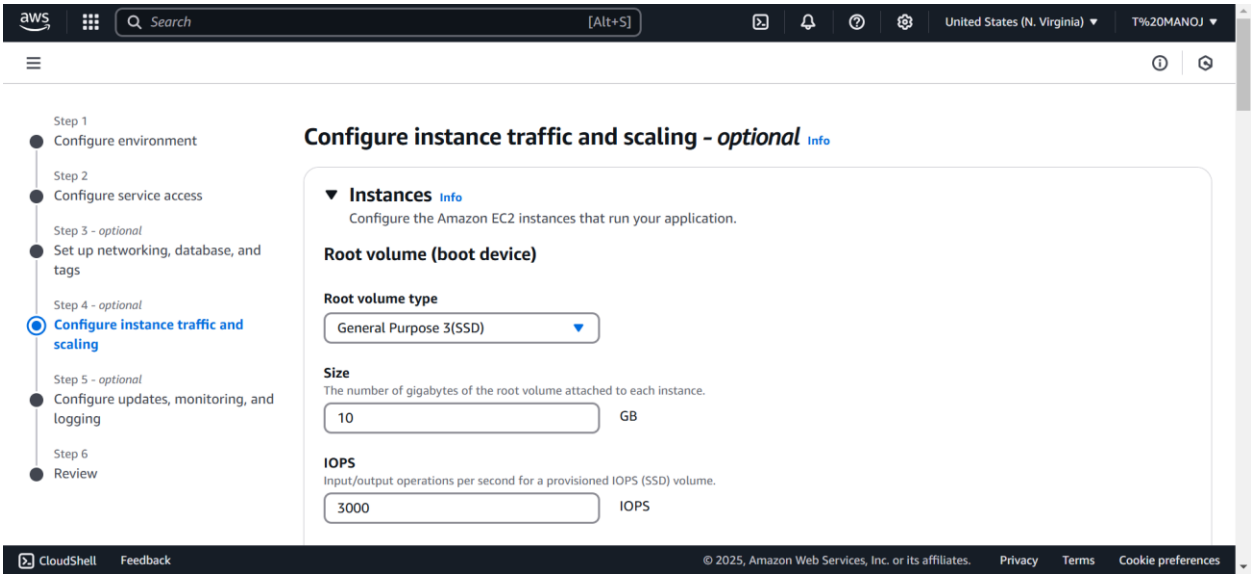


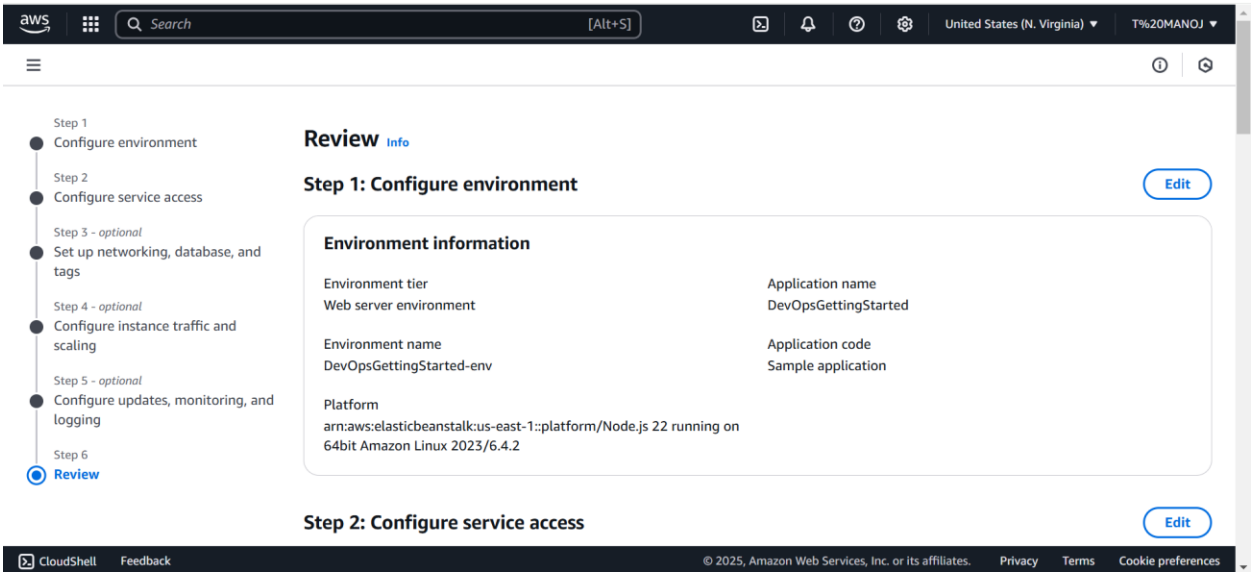



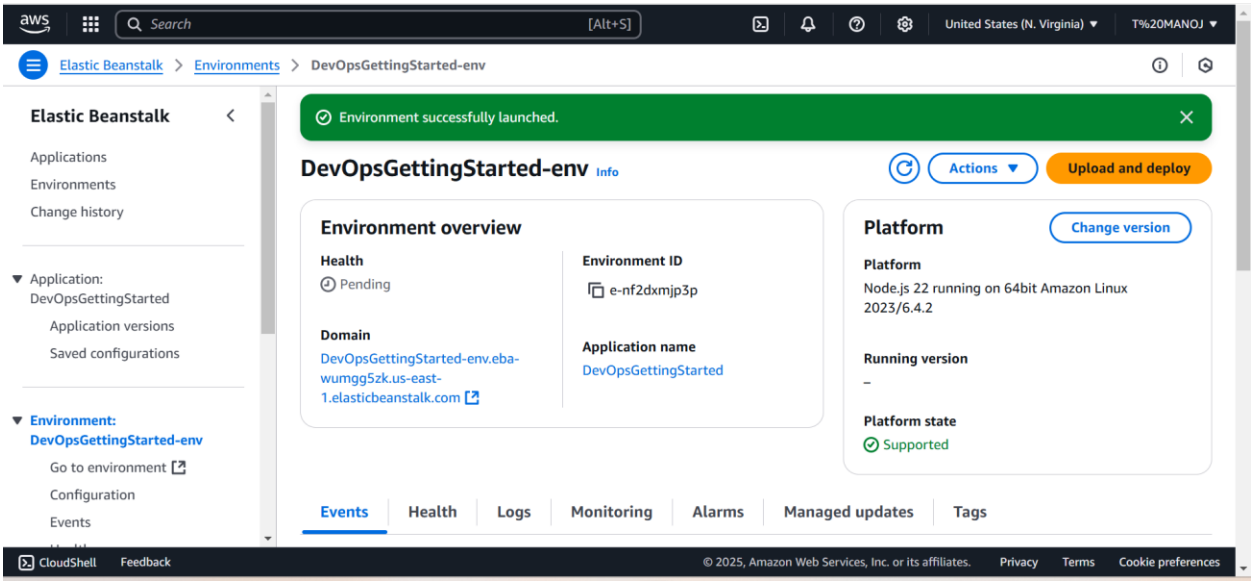

### *Step 4: Deploy and Monitor*

1. Click Submit.
2. Monitor deployment (takes ~5 minutes).
3. On success, a green checkmark appears, and a URL is generated.


aws
Search
[Alt+S]
United States (N. Virginia)
T%20MANOJ
Configure instance traffic and scaling
Step 5 - optional
Configure updates, monitoring, and logging
Step 6
Review
Use an existing service role
Service role name
Enter the name for an IAM role that Elastic Beanstalk will create to assume as a service role. Beanstalk will attach the required managed policies to it.
aws-elasticbeanstalk-service-role
View permission details
EC2 key pair
Select an EC2 key pair to securely log in to your EC2 instances. Learn more
RSA
EC2 instance profile
Choose an IAM instance profile with managed policies that allow your EC2 instances to perform required operations.
AWSCICDROLE
View permission details
Cancel
Skip to review
Previous
Next
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

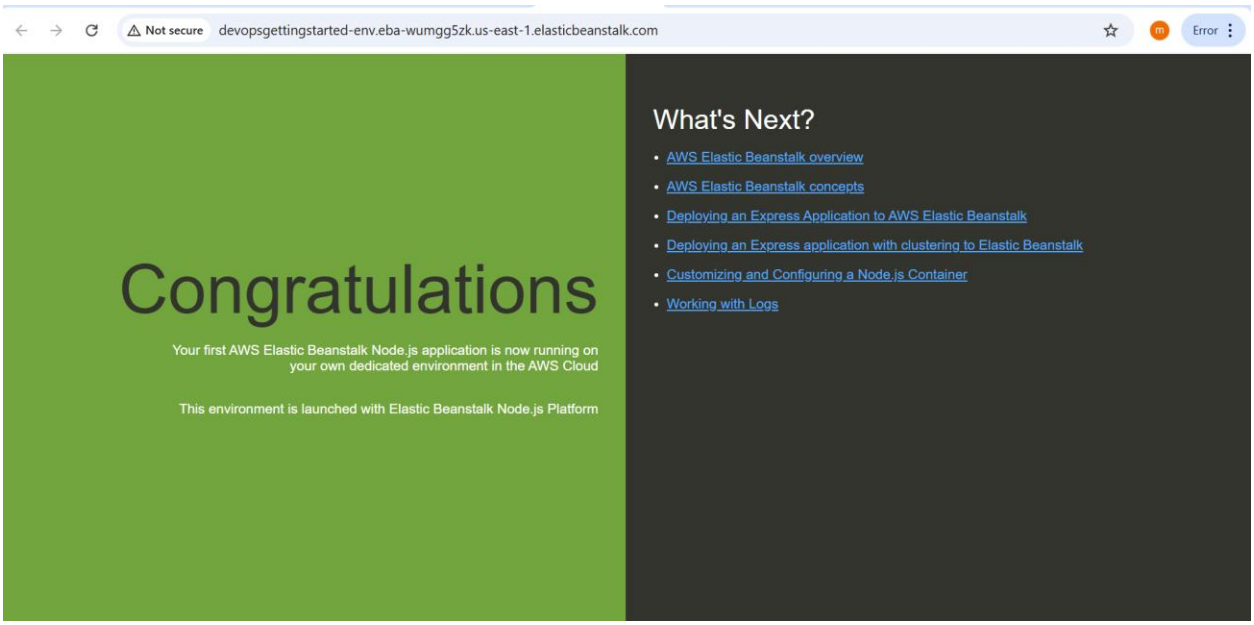

aws
Search
[Alt+S]
United States (N. Virginia)
T%20MANOJ
Step 1
Configure environment
Step 2
Configure service access
Step 3 - optional
Set up networking, database, and tags
Step 4 - optional
Configure instance traffic and scaling
Step 5 - optional
Configure updates, monitoring, and logging
Step 6
Review
Configure instance traffic and scaling - optional
Info
Instances
Info
Configure the Amazon EC2 instances that run your application.
Root volume (boot device)
Root volume type
General Purpose 3(SSD)
Size
The number of gigabytes of the root volume attached to each instance.
10
GB
IOPS
Input/output operations per second for a provisioned IOPS (SSD) volume.
3000
IOPS
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences




### *Step 5: Verify the Application*

- Open the generated URL in your browser; you should see the default Elastic Beanstalk page.

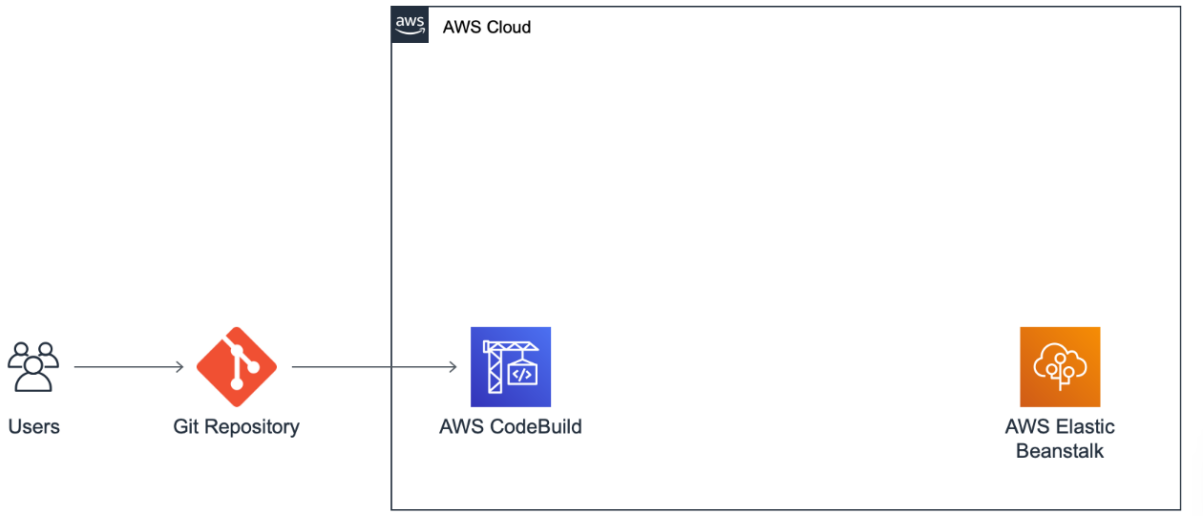



VERIFIED SAMPLE OUTPUT

# Create Build Project using AWS CodeBuild

## 🎯 Objective:

- Automate building the codebase from GitHub using AWS CodeBuild.



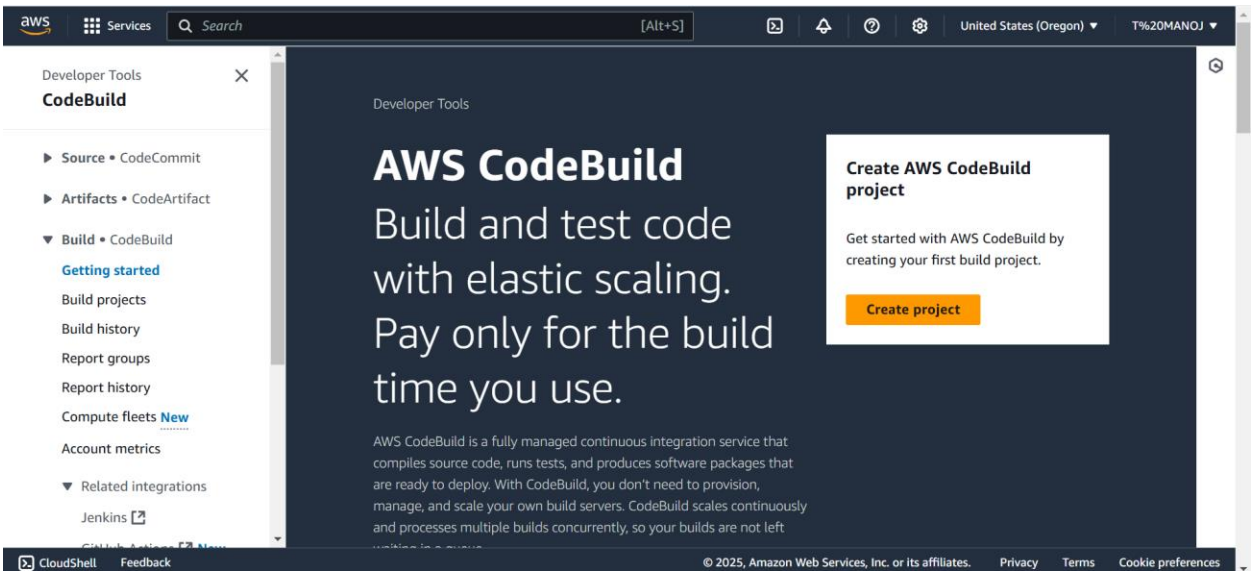

## 📌 Terminologies:

1. AWS CodeBuild: Fully managed build service that compiles source code, runs tests, and produces ready-to-deploy packages.
2. Buildspec File: `buildspec.yml` file that defines build commands.
3. Continuous Integration (CI): Regular merging of code changes into the main branch with automated builds.

## ⚙️ Steps to Create Build Project:

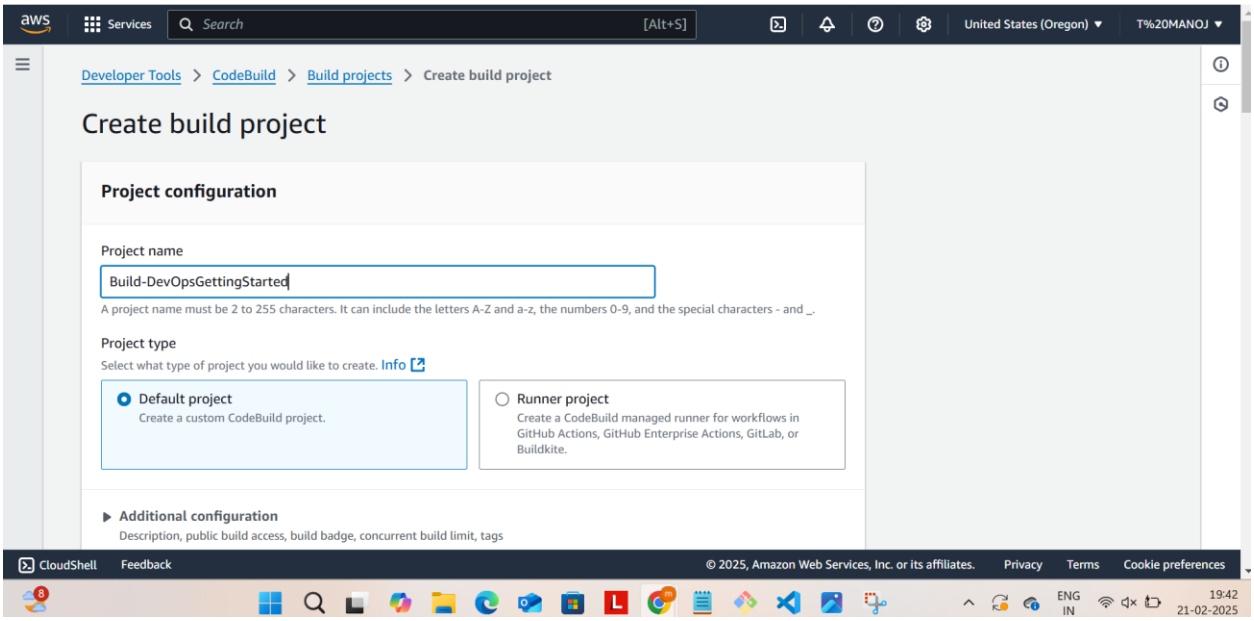
*Step 1: Open AWS CodeBuild Console*

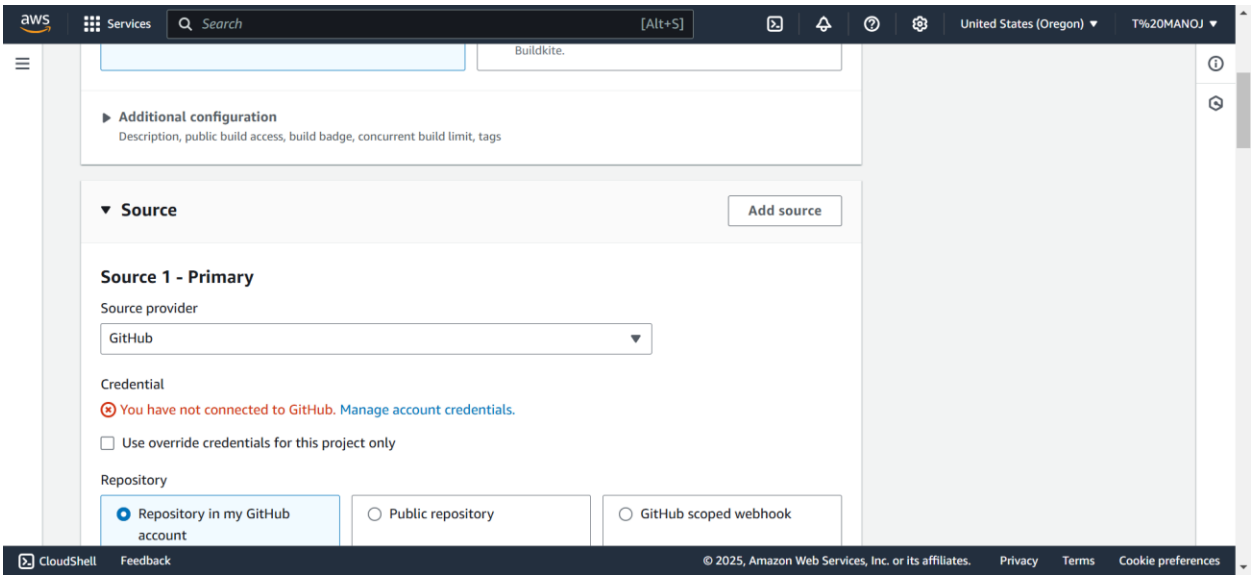
- Search for AWS CodeBuild in the AWS console.

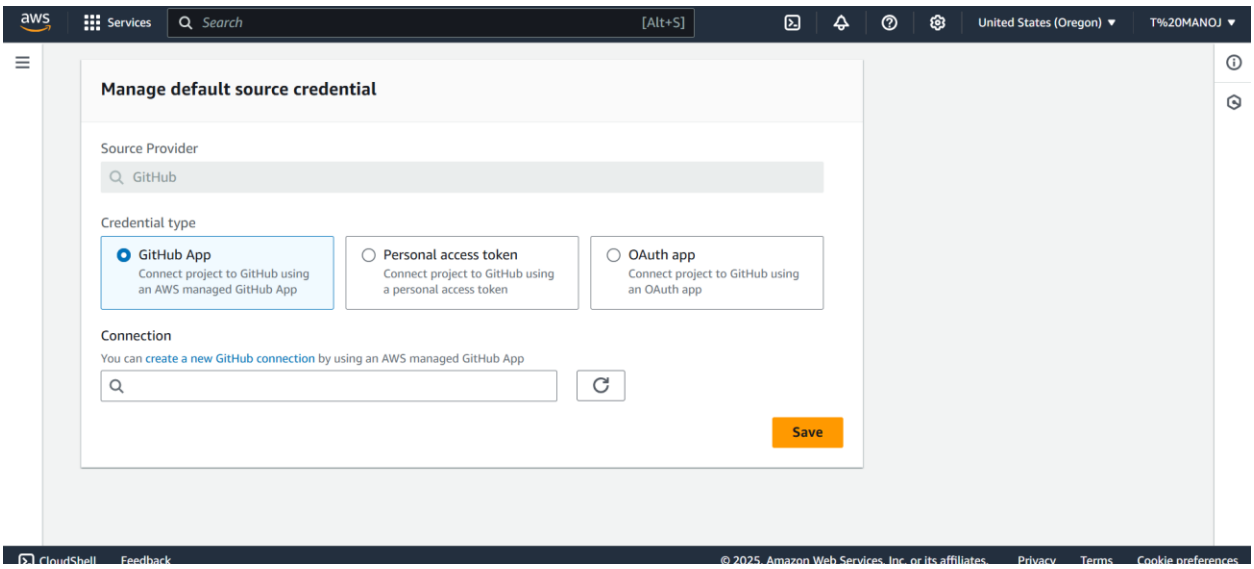


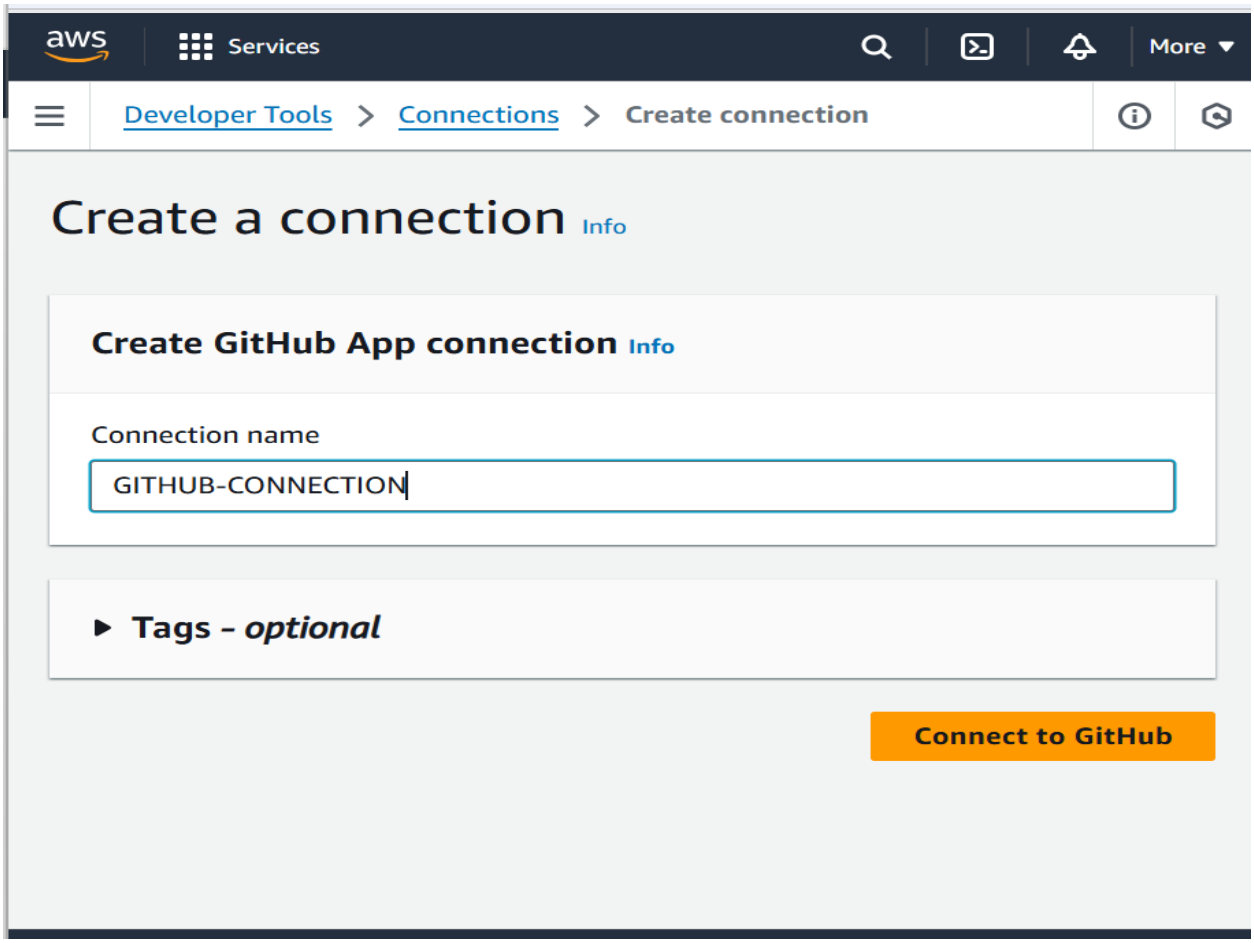

### *Step 2: Create a New Build Project*

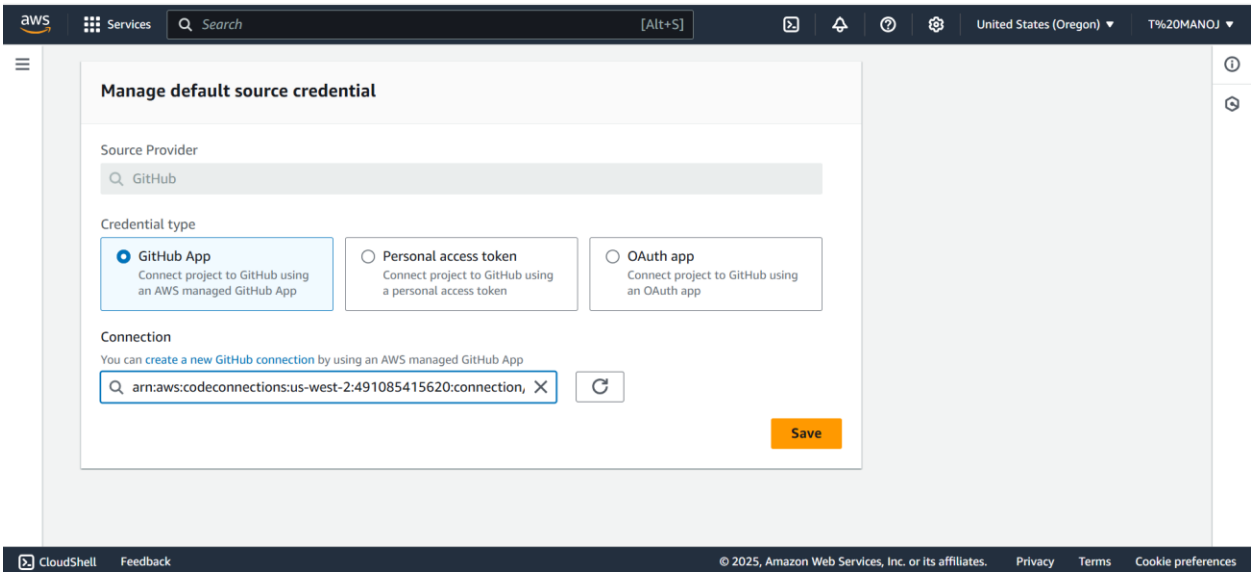
1. Click Create Build Project.
2. Project Name: `BuildDevOpsGettingStarted`.
3. Source Provider: GitHub.
4. GitHub Connection:
    a. Click Connect to GitHub via OAuth.
    b. Select the forked repository.

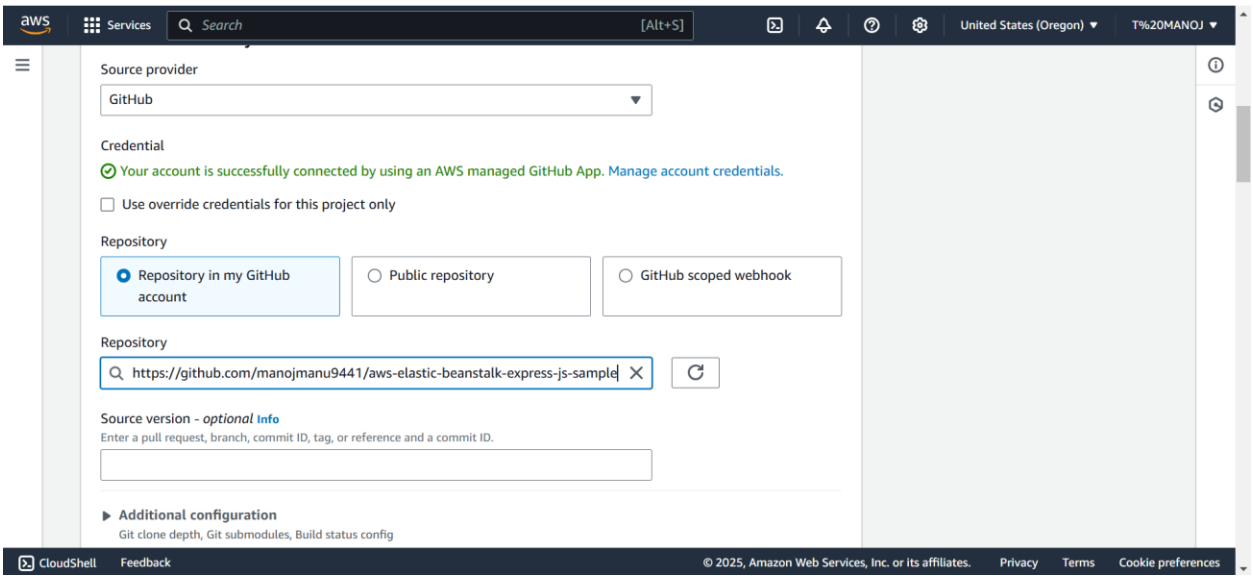


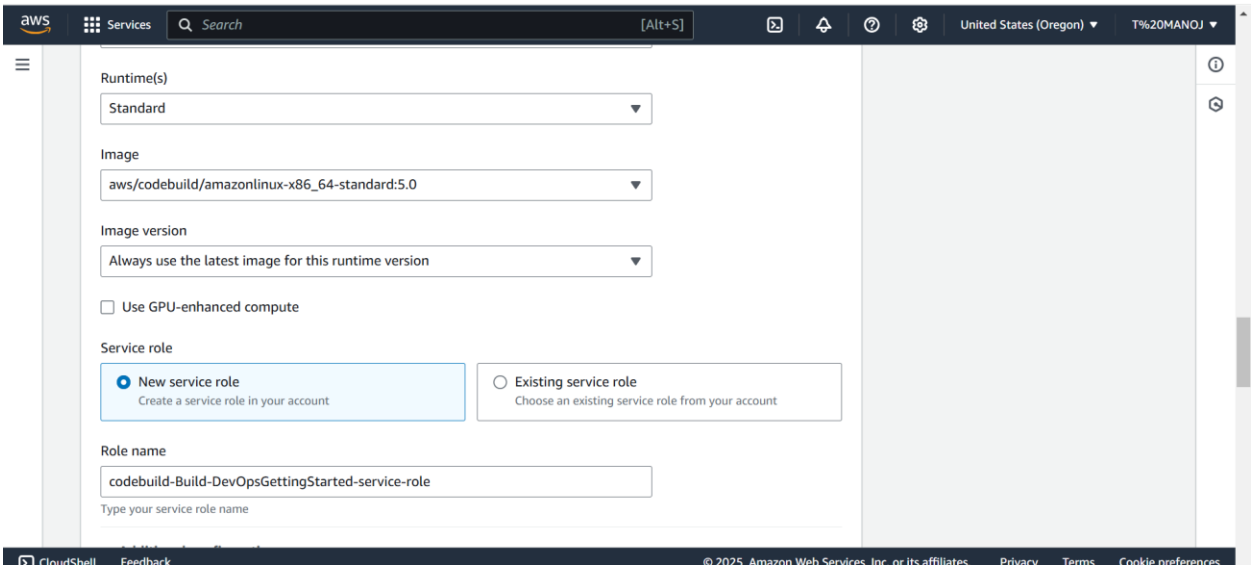

aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Buildkite.
Additional configuration
Description, public build access, build badge, concurrent build limit, tags
Source
Add source
Source 1 - Primary
Source provider
GitHub
Credential
You have not connected to GitHub. Manage account credentials.
Use override credentials for this project only
Repository
Repository in my GitHub account
Public repository
GitHub scoped webhook
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Manage default source credential
Source Provider
GitHub
Credential type
GitHub App
Connect project to GitHub using an AWS managed GitHub App
Personal access token
Connect project to GitHub using a personal access token
OAuth app
Connect project to GitHub using an OAuth app
Connection
You can create a new GitHub connection by using an AWS managed GitHub App
Save
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
More
Developer Tools
Connections
Create connection
Create a connection Info
Create GitHub App connection Info
Connection name
GITHUB-CONNECTION
Tags – optional
Connect to GitHub


aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Manage default source credential
Source Provider
GitHub
Credential type
GitHub App
Connect project to GitHub using an AWS managed GitHub App
Personal access token
Connect project to GitHub using a personal access token
OAuth app
Connect project to GitHub using an OAuth app
Connection
You can create a new GitHub connection by using an AWS managed GitHub App
arn:aws:codeconnections:us-west-2:491085415620:connection,
Save
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences



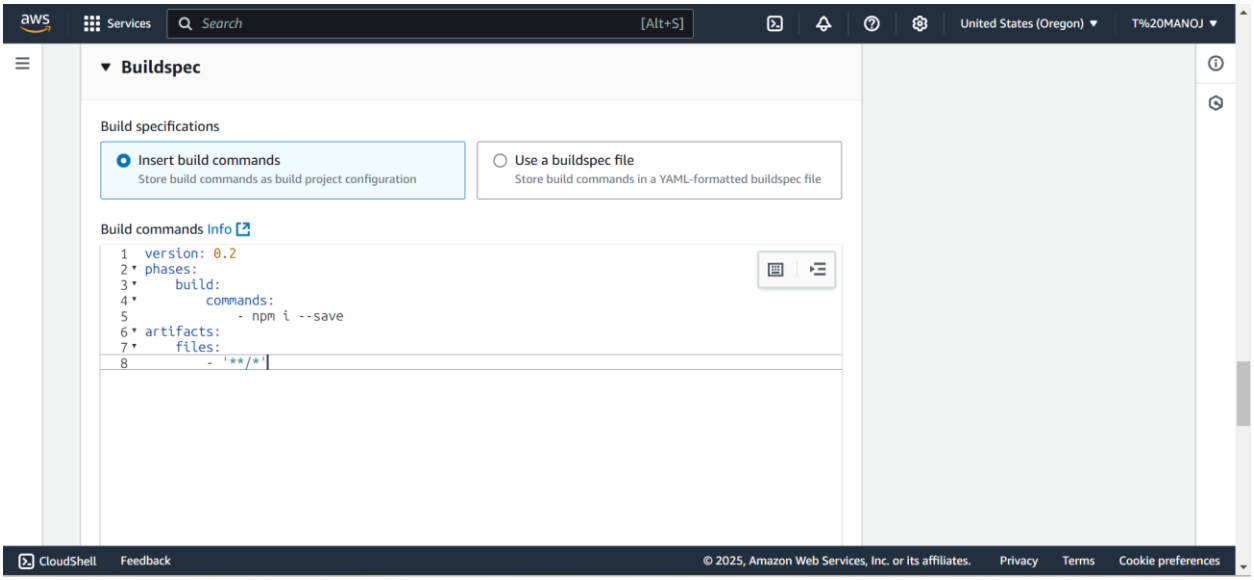

### *Step 3: Configure Build Environment*

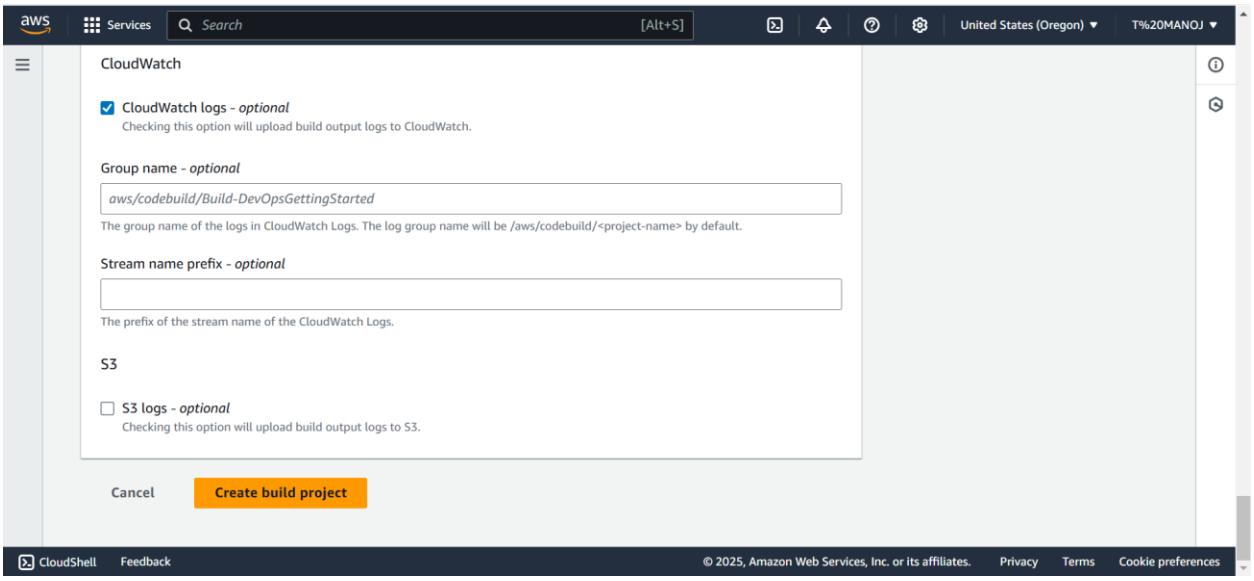
1. Environment Image: Managed image.
2. Operating System: AmazonLinux 5.0.
3. Runtime: Node.js (Choose version 14+).
4. Buildspec: Create a file named `buildspec.yml` in your repository:

JSON :

```
version: 0.2
    phases:
```

```
        build:
            commands:
                - npm i --save
artifacts:
    files:
        - '**/*'
```

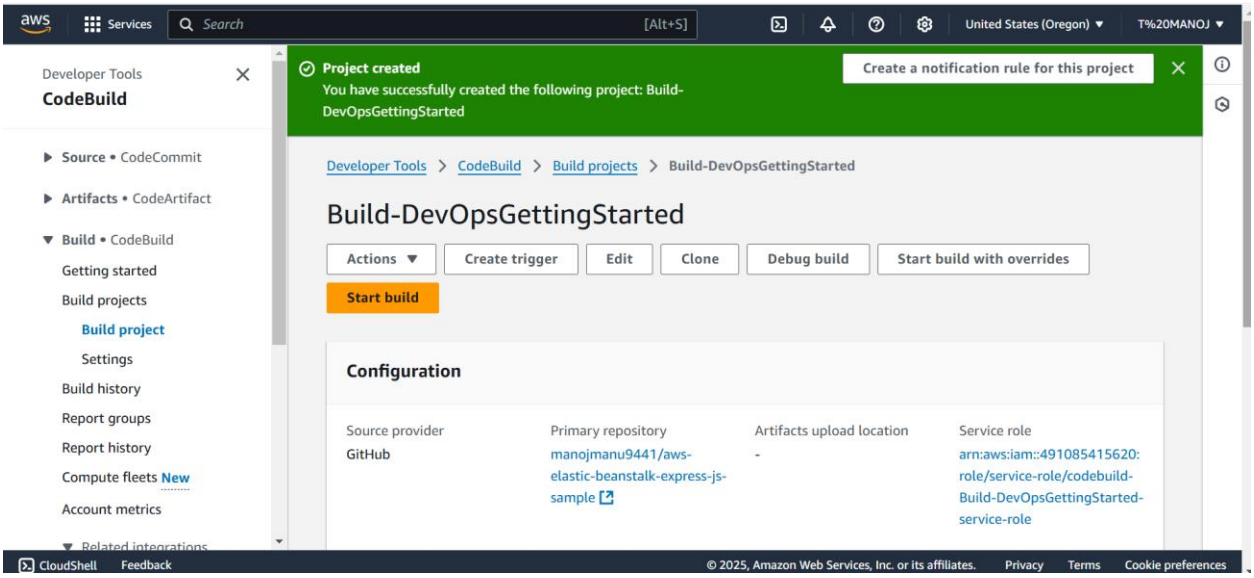


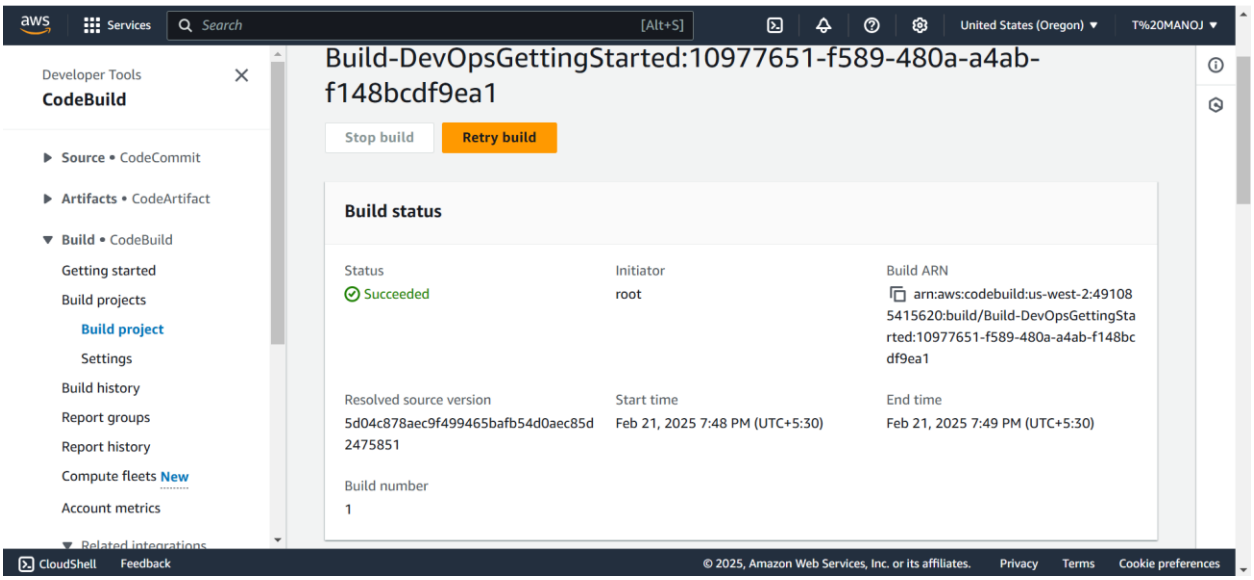




### *Step 4: Build the Project*

- Click Start Build.

- Confirm the build runs successfully (Green ✓ checkmark).

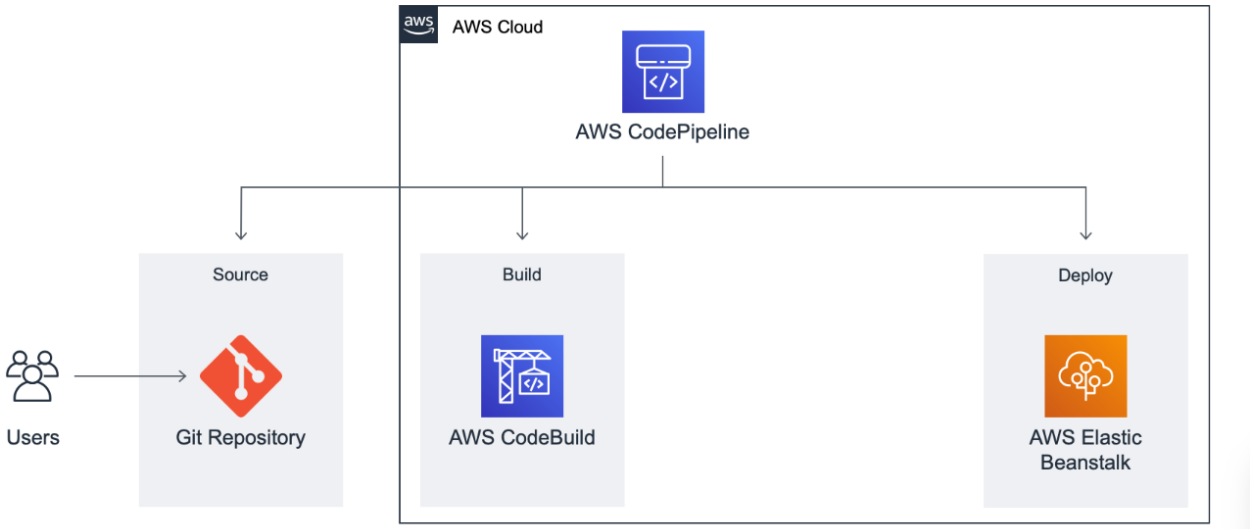


# Set Up Continuous Delivery Pipeline using AWS CodePipeline

## Objective:

- Create a CI/CD pipeline using CodePipeline for automated deployment.




## Terminologies:

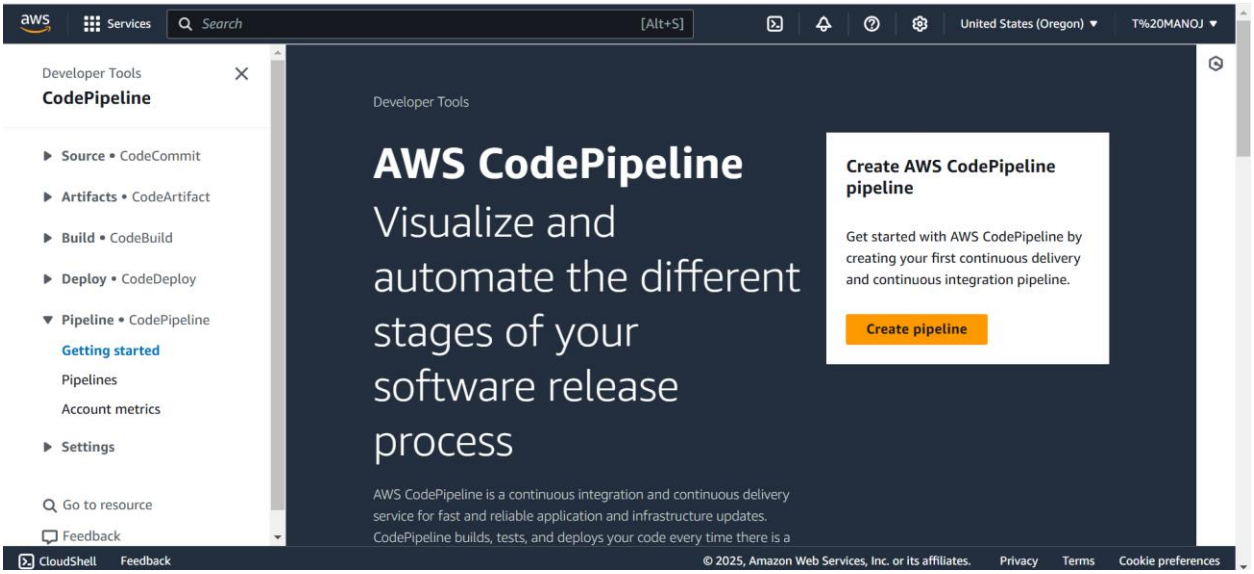
1. AWS CodePipeline: Orchestrates continuous integration and deployment workflows.
2. Pipeline Stages:
    a. Source: Code retrieval.
    b. Build: Code compilation (CodeBuild).
    c. Deploy: Code deployment (Elastic Beanstalk).

## Steps to Create CI/CD Pipeline:

### *Step 1: Access CodePipeline Console*

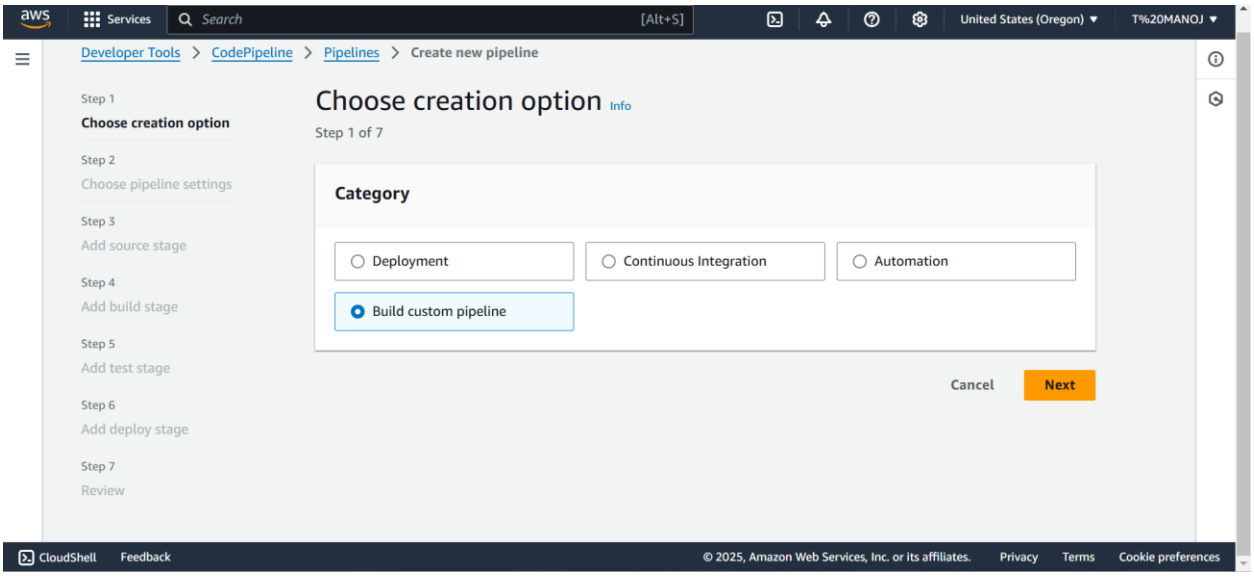
- Navigate to AWS CodePipeline in the AWS console.

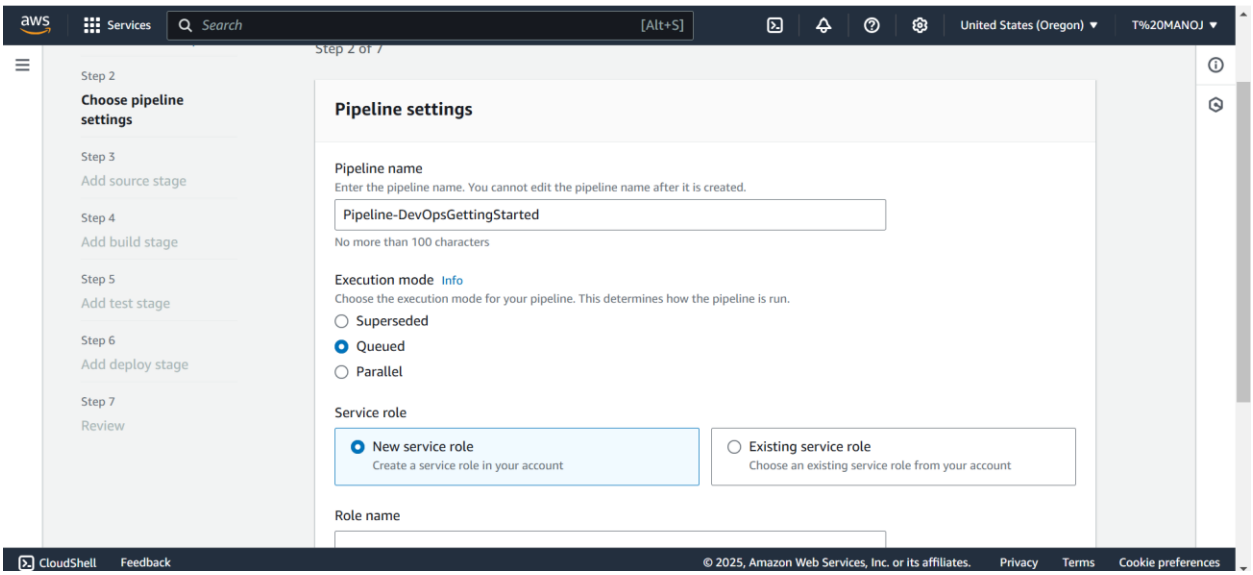
- Click Create Pipeline.




### *Step 2: Configure Pipeline*

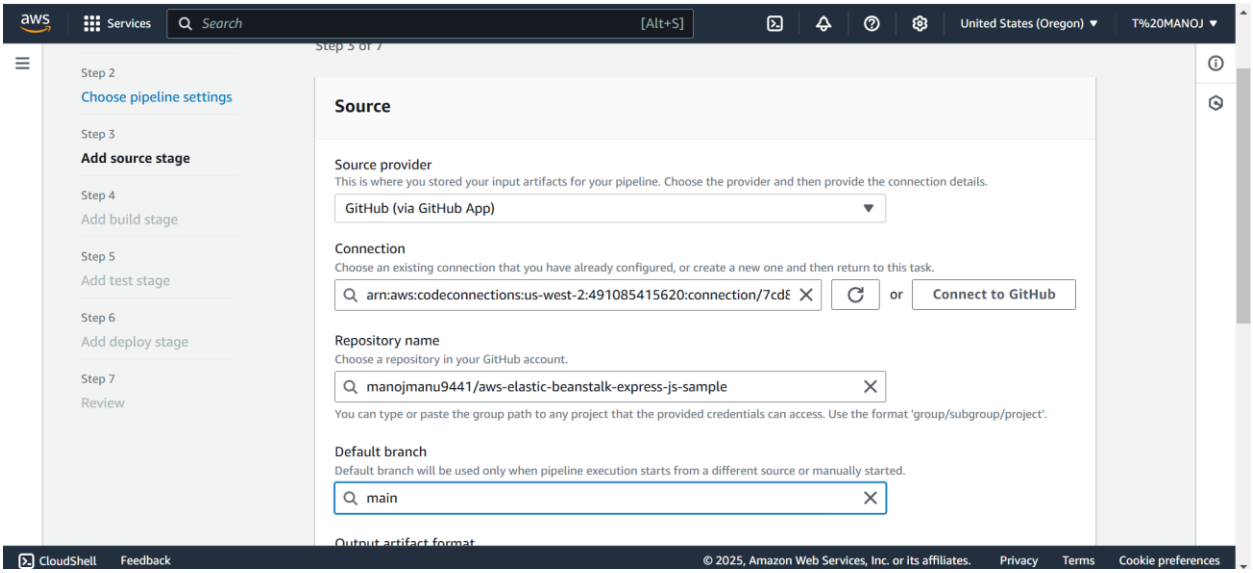
1. Pipeline Name: Pipeline-DevOpsGettingStarted.
2. Select New service role or existing service role (`AWSCodePipelineServiceRole`).





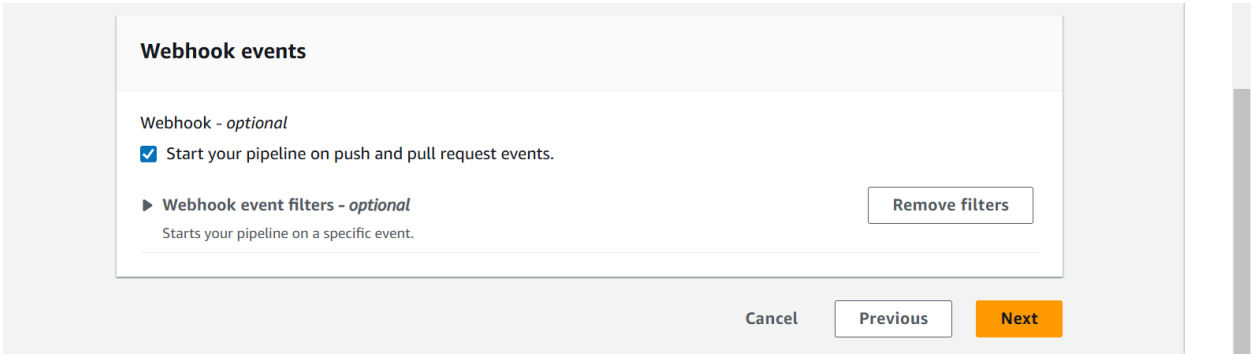

### *Step 3: Add Source Stage*

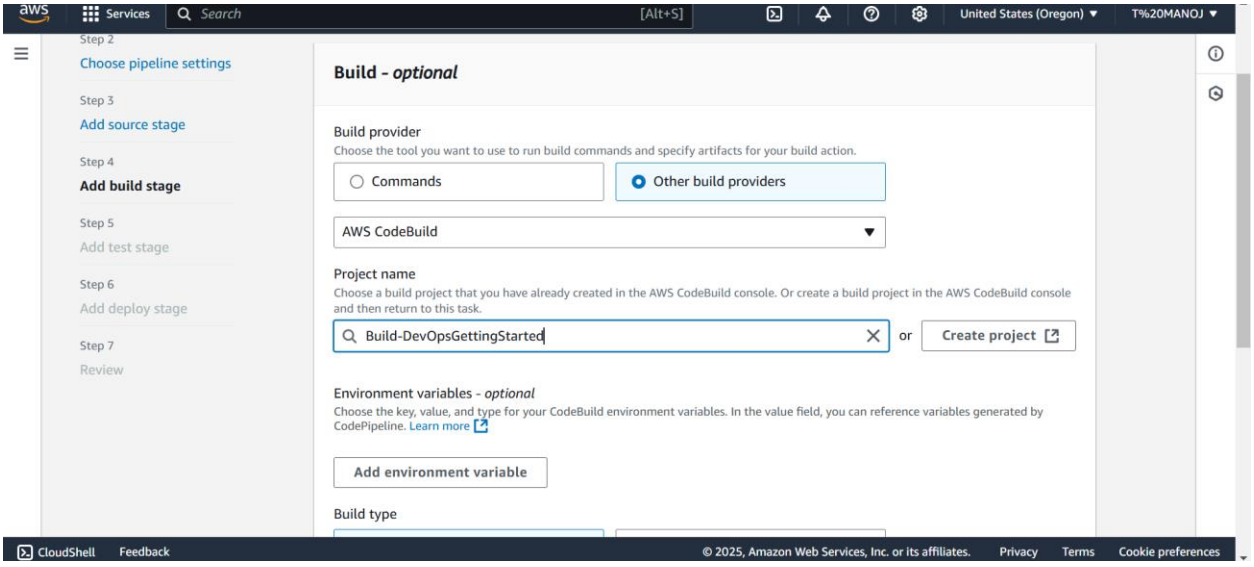
1. Source Provider: GitHub.
2. Repository: Select your GitHub repo.
3. Branch: `main`.
4. Change Detection: Select GitHub webhooks (auto-triggers builds).

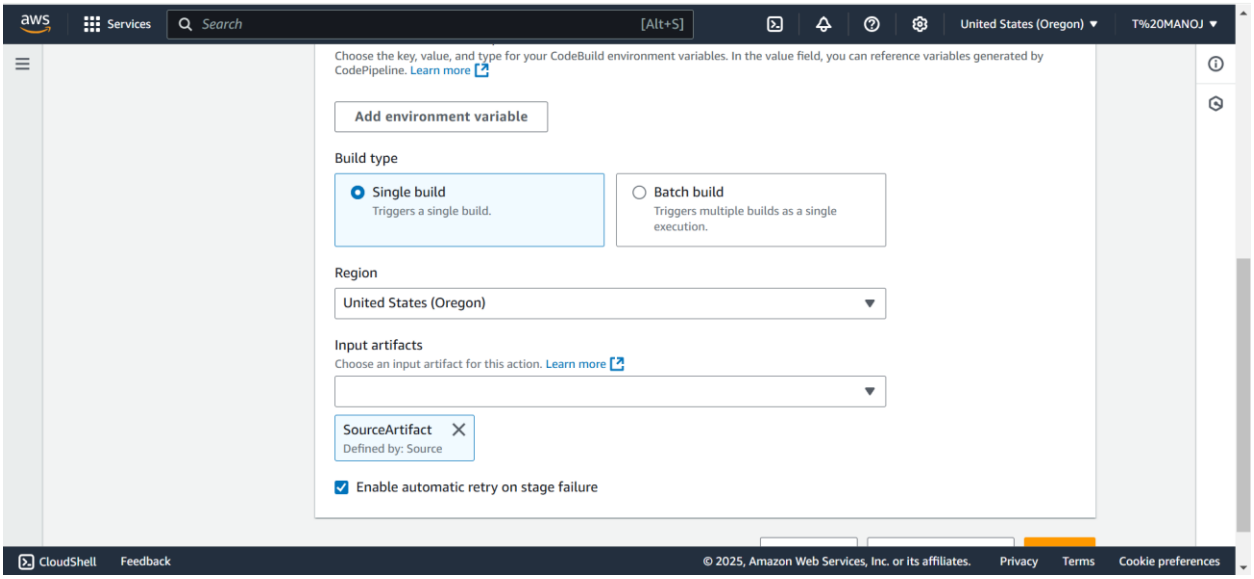



### *Step 4: Add Build Stage*

1. Build Provider: AWS CodeBuild.
2. Project Name: DevOpsGettingstarted.

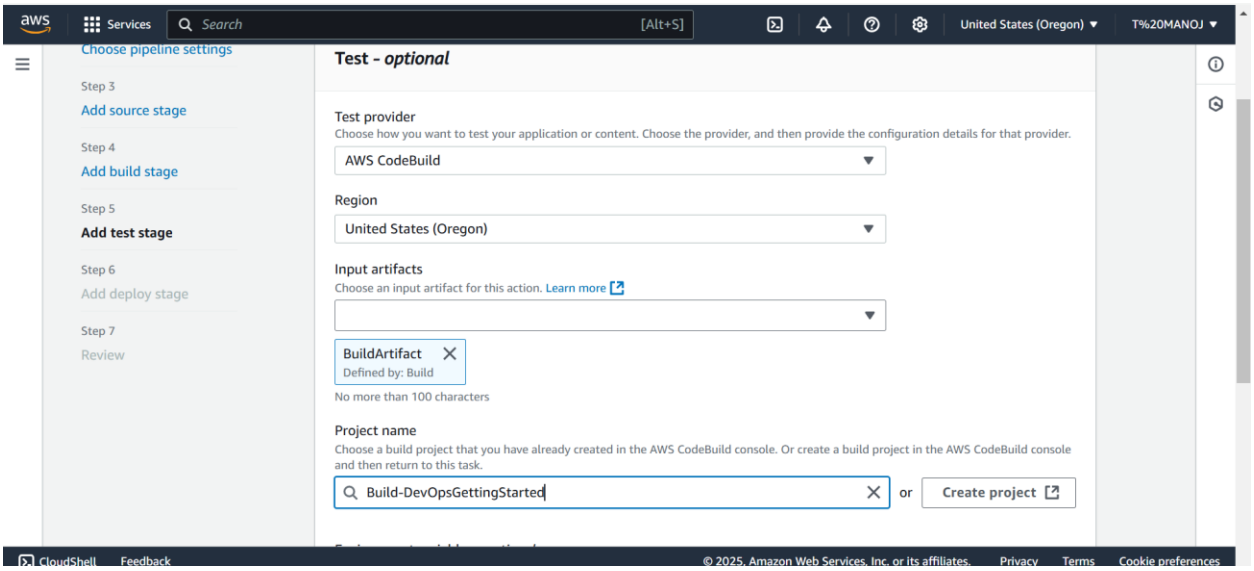

Webhook events
Webhook - optional
Start your pipeline on push and pull request events.
Webhook event filters - optional
Starts your pipeline on a specific event.
Remove filters
Cancel
Previous
Next

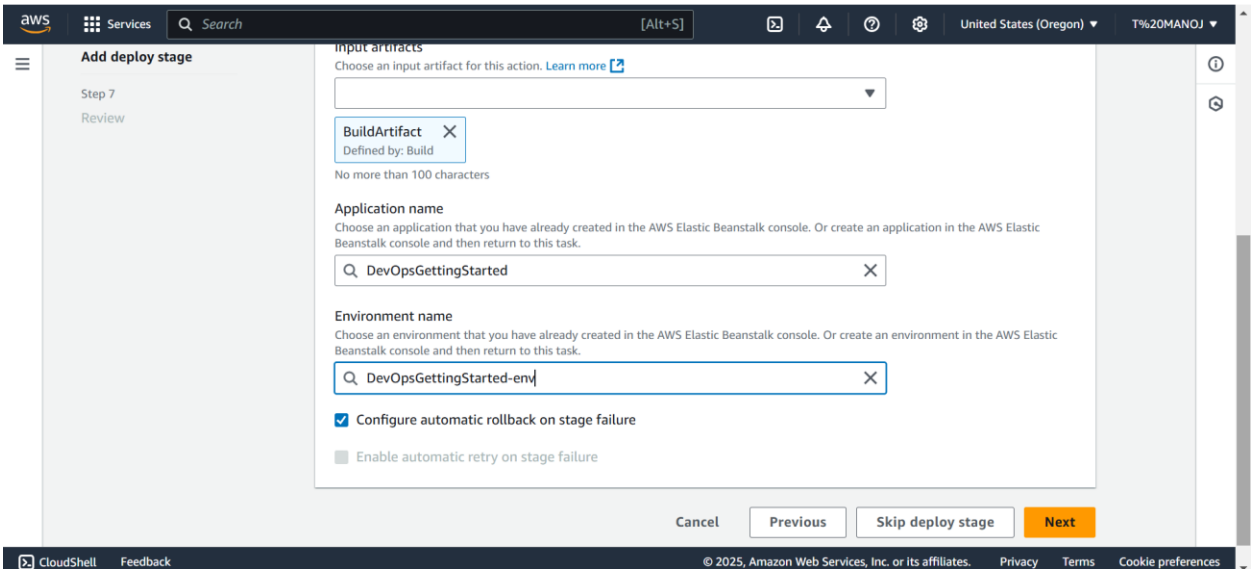

aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Step 2
Choose pipeline settings
Step 3
Add source stage
Step 4
Add build stage
Step 5
Add test stage
Step 6
Add deploy stage
Step 7
Review
Build - optional
Build provider
Choose the tool you want to use to run build commands and specify artifacts for your build action.
Commands
Other build providers
AWS CodeBuild
Project name
Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.
Build-DevOpsGettingStarted
or
Create project
Environment variables - optional
Choose the key, value, and type for your CodeBuild environment variables. In the value field, you can reference variables generated by CodePipeline. Learn more
Add environment variable
Build type
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Choose the key, value, and type for your CodeBuild environment variables. In the value field, you can reference variables generated by CodePipeline. Learn more
Add environment variable
Build type
Single build
Triggers a single build.
Batch build
Triggers multiple builds as a single execution.
Region
United States (Oregon)
Input artifacts
Choose an input artifact for this action. Learn more
SourceArtifact
Defined by: Source
Enable automatic retry on stage failure
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

### *Step 5: Add Deploy Stage*

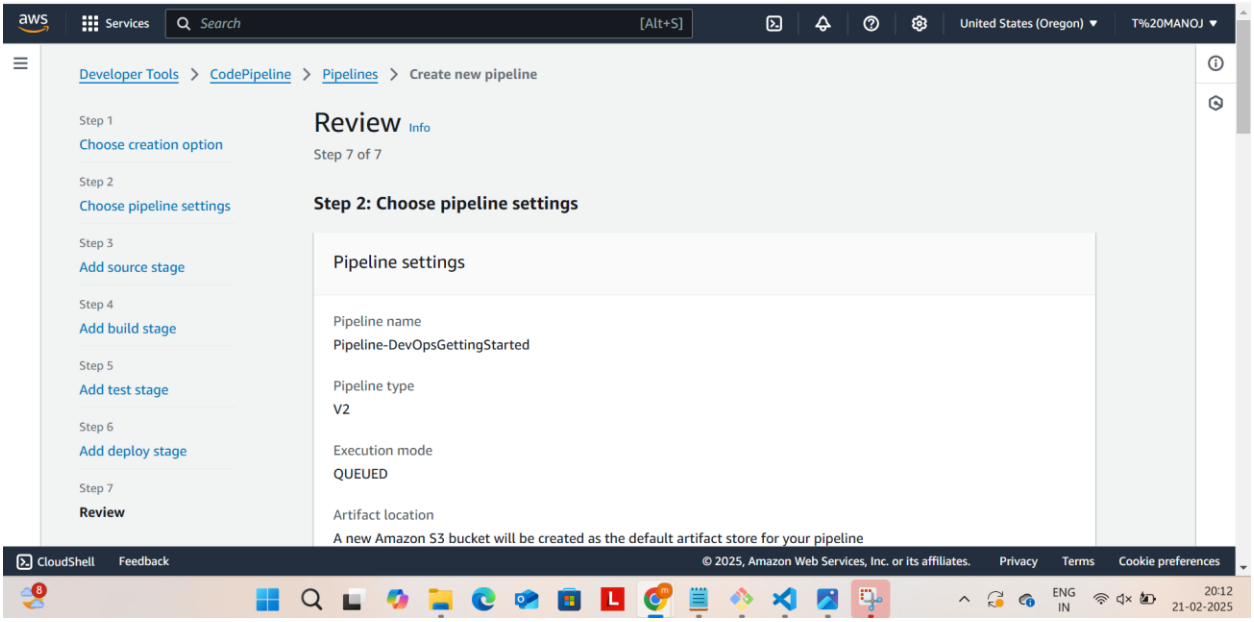
1. Deploy Provider: AWS Elastic Beanstalk.
2. Application Name: `DevOpsGettingStarted`.
3. Environment Name: Select the environment created earlier.

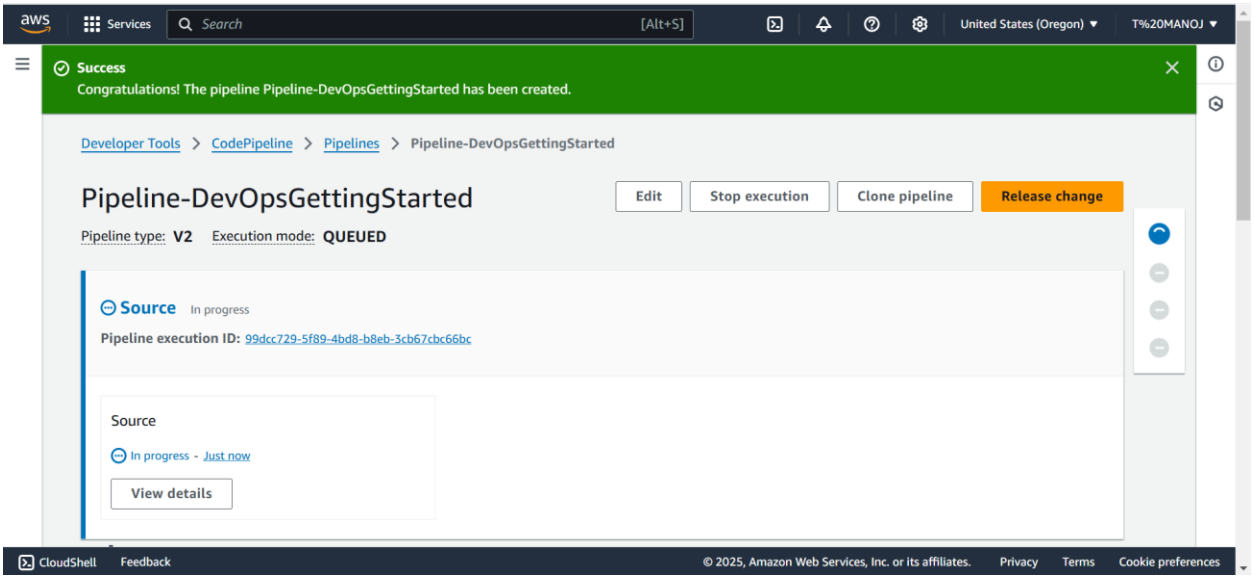


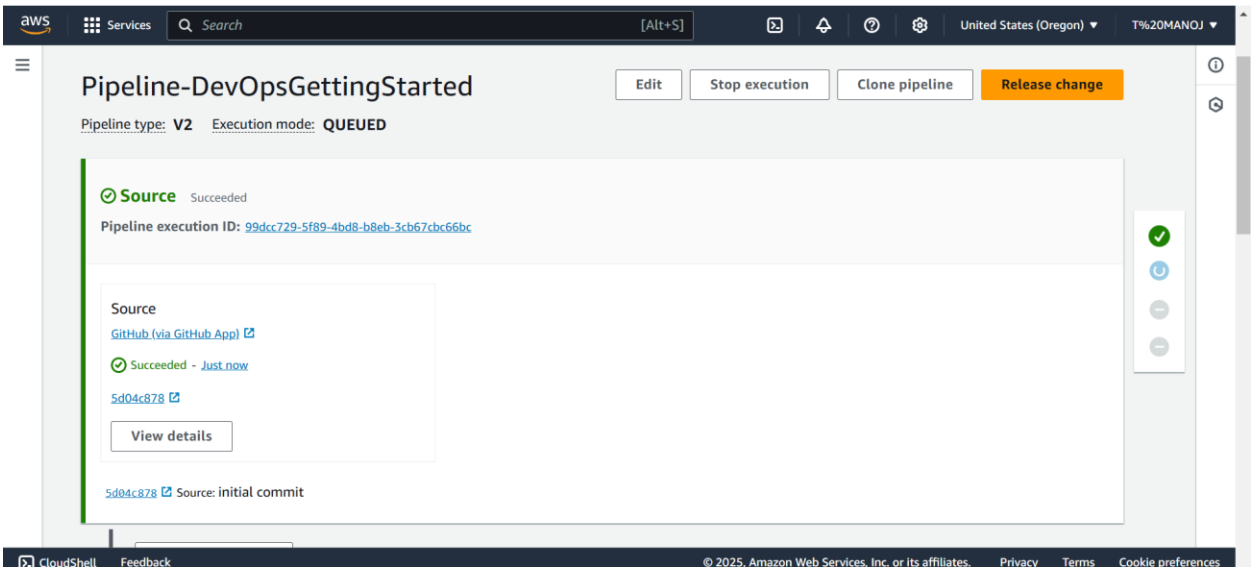



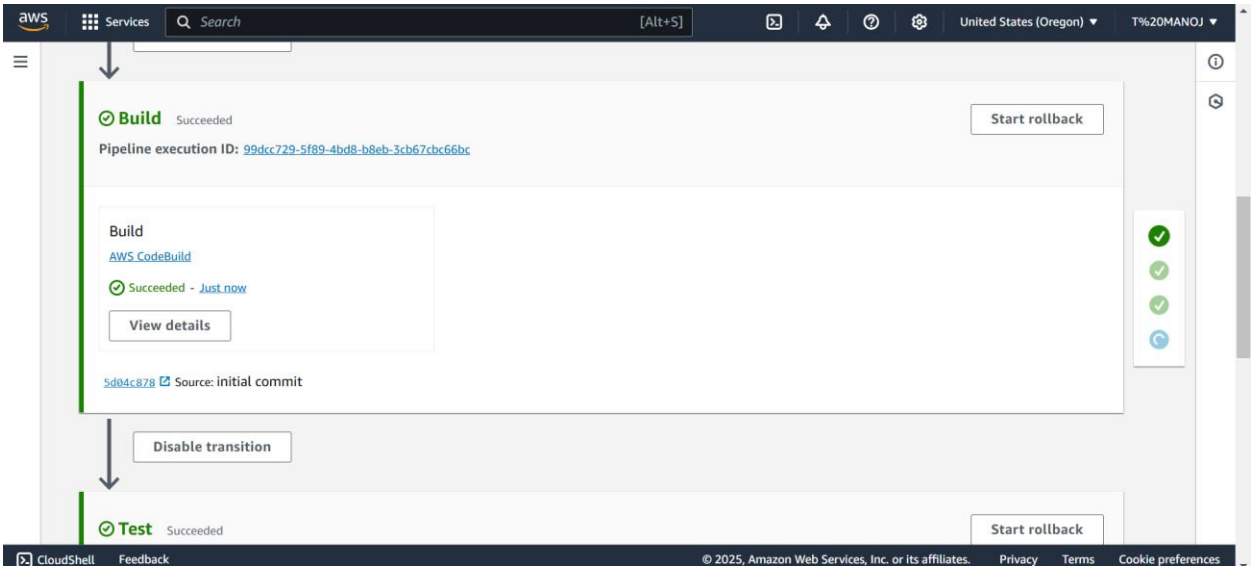

### *Step 6: Finalize and Execute*

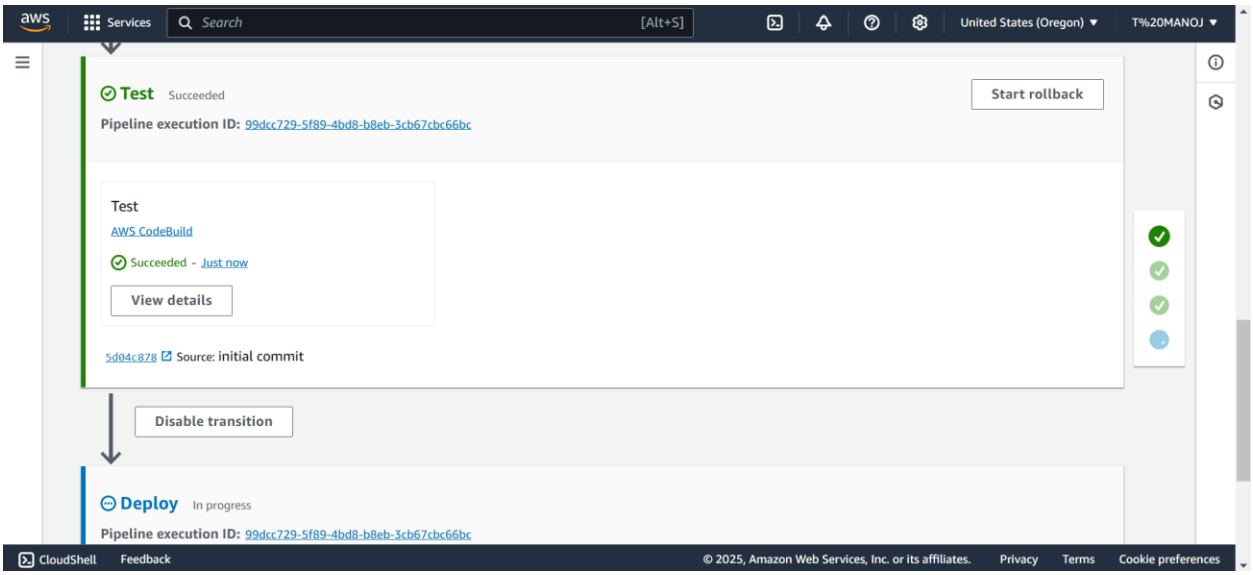
- Click Create Pipeline.
- The pipeline runs automatically on code pushes.

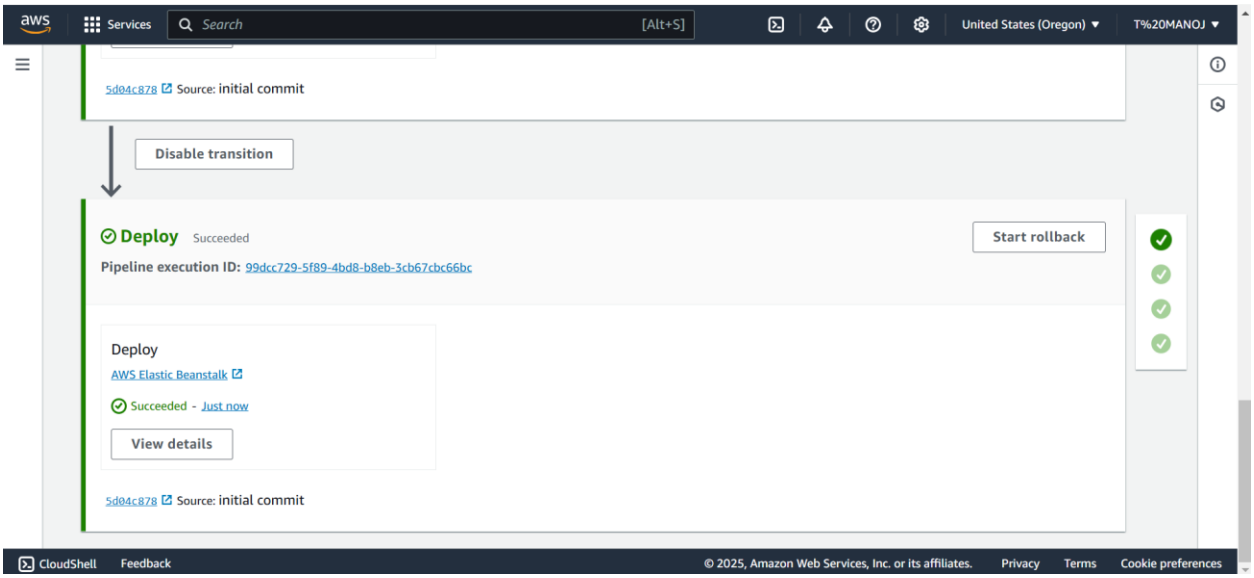

aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Developer Tools > CodePipeline > Pipelines > Create new pipeline
Step 1
Choose creation option
Step 2
Choose pipeline settings
Step 3
Add source stage
Step 4
Add build stage
Step 5
Add test stage
Step 6
Add deploy stage
Step 7
Review
Review Info
Step 7 of 7
Step 2: Choose pipeline settings
Pipeline settings
Pipeline name
Pipeline-DevOpsGettingStarted
Pipeline type
V2
Execution mode
QUEUED
Artifact location
A new Amazon S3 bucket will be created as the default artifact store for your pipeline
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
ENG
IN
20:12
21-02-2025

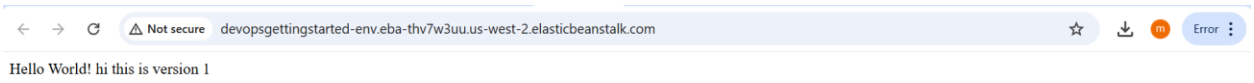

aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Success
Congratulations! The pipeline Pipeline-DevOpsGettingStarted has been created.
Developer Tools > CodePipeline > Pipelines > Pipeline-DevOpsGettingStarted
Pipeline-DevOpsGettingStarted
Edit
Stop execution
Clone pipeline
Release change
Pipeline type: V2
Execution mode: QUEUED
Source
In progress
Pipeline execution ID: 99dcc729-5f89-4bd8-b8eb-3cb67cbc66bc
Source
In progress - Just now
View details
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Pipeline-DevOpsGettingStarted
Edit
Stop execution
Clone pipeline
Release change
Pipeline type: V2
Execution mode: QUEUED
Source
Succeeded
Pipeline execution ID: 99dcc729-5f89-4bd8-b8eb-3cb67cbc66bc
Source
GitHub (via GitHub App)
Succeeded - Just now
5d04c878
View details
5d04c878 Source: initial commit
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Build
Succeeded
Start rollback
Pipeline execution ID: 99dcc729-5f89-4bd8-b8eb-3cb67cbc66bc
Build
AWS CodeBuild
Succeeded - Just now
View details
5d04c878 Source: initial commit
Disable transition
Test
Succeeded
Start rollback
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Test
Succeeded
Start rollback
Pipeline execution ID: 99dcc729-5f89-4bd8-b8eb-3cb67cbc66bc
Test
AWS CodeBuild
Succeeded - Just now
View details
5d04c878
Source: initial commit
Disable transition
Deploy
In progress
Pipeline execution ID: 99dcc729-5f89-4bd8-b8eb-3cb67cbc66bc
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
5d04c878
Source: initial commit
Disable transition
Deploy
Succeeded
Start rollback
Pipeline execution ID: 99dcc729-5f89-4bd8-b8eb-3cb67cbc66bc
Deploy
AWS Elastic Beanstalk
Succeeded - Just now
View details
5d04c878
Source: initial commit
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

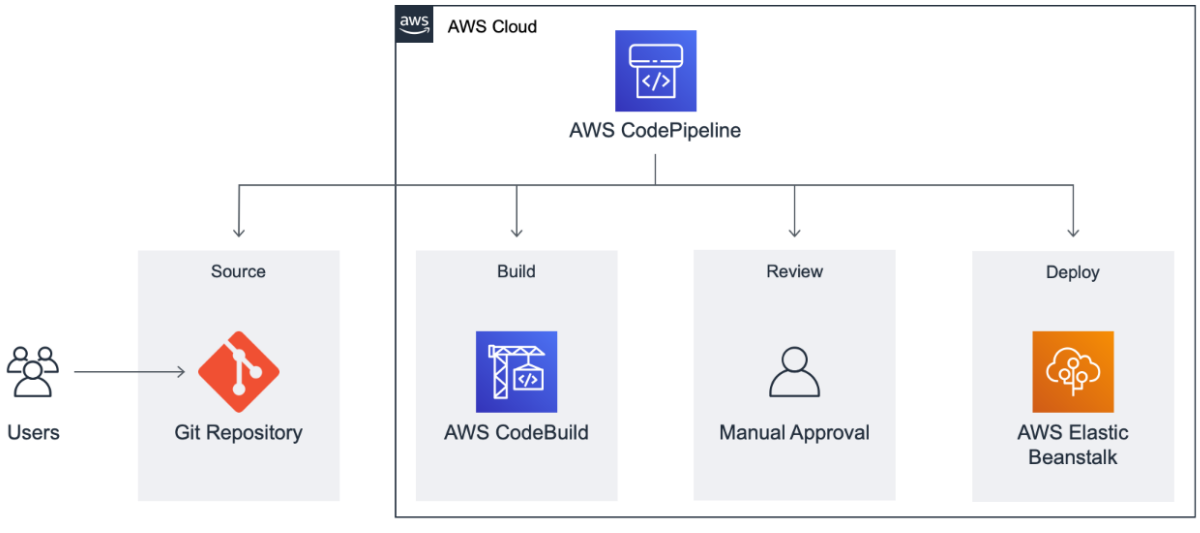



---

# Adding Manual Approval with SNS & SQS

## Objective:

- Introduce a manual approval step before deployment using SNS and SQS.



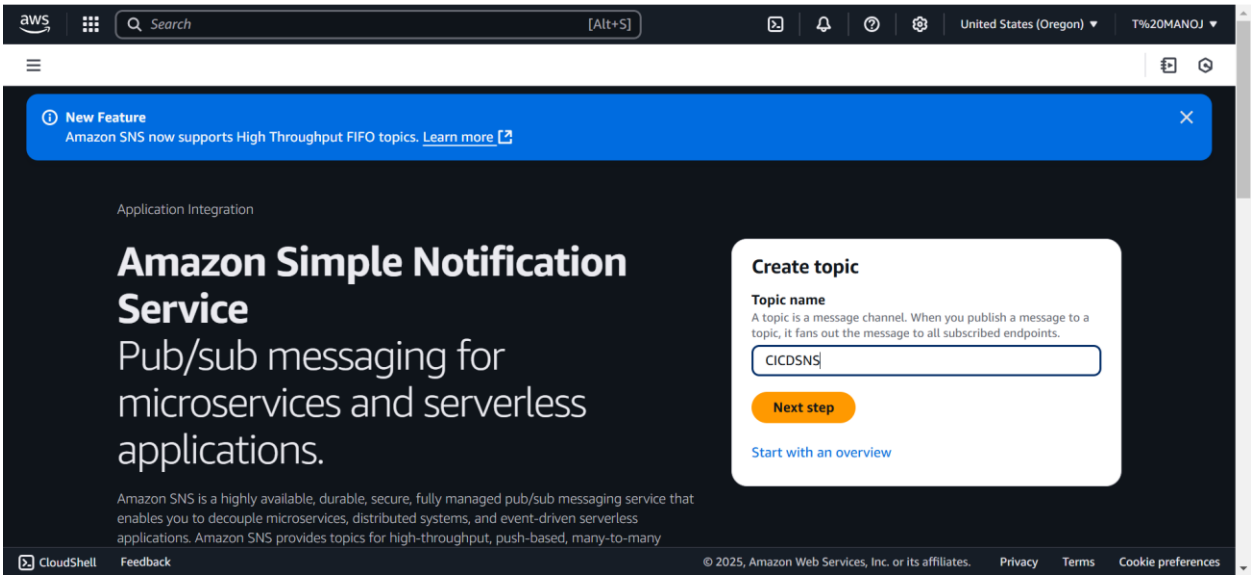
## 📌 Terminologies:

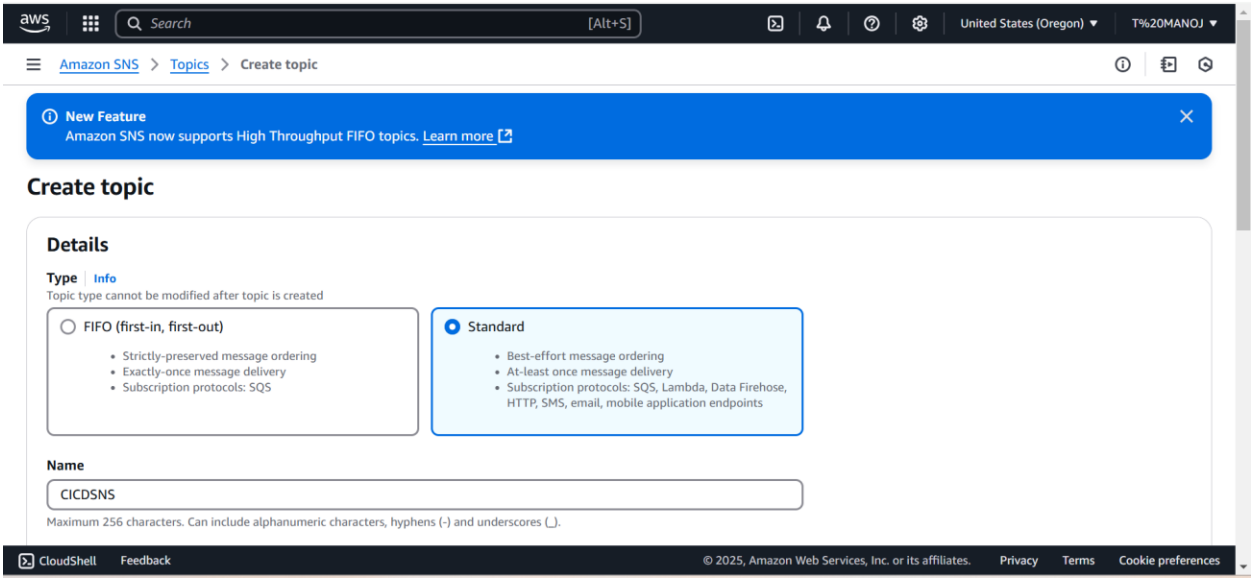
1. Manual Approval Stage: Pauses the pipeline for human validation.
2. SNS (Simple Notification Service): Sends notifications (email, SMS).
3. SQS (Simple Queue Service): Stores approval requests.
4. Rollback: Pipeline reverses changes if approval is denied.

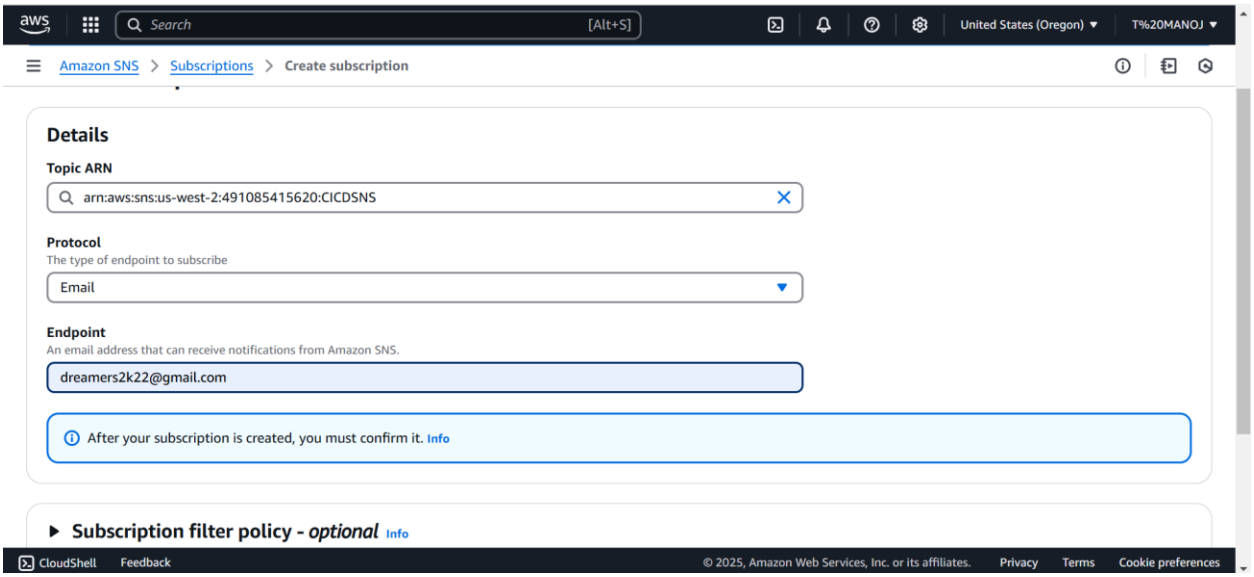
## ⚙️ Steps for Manual Approval & Rollback:

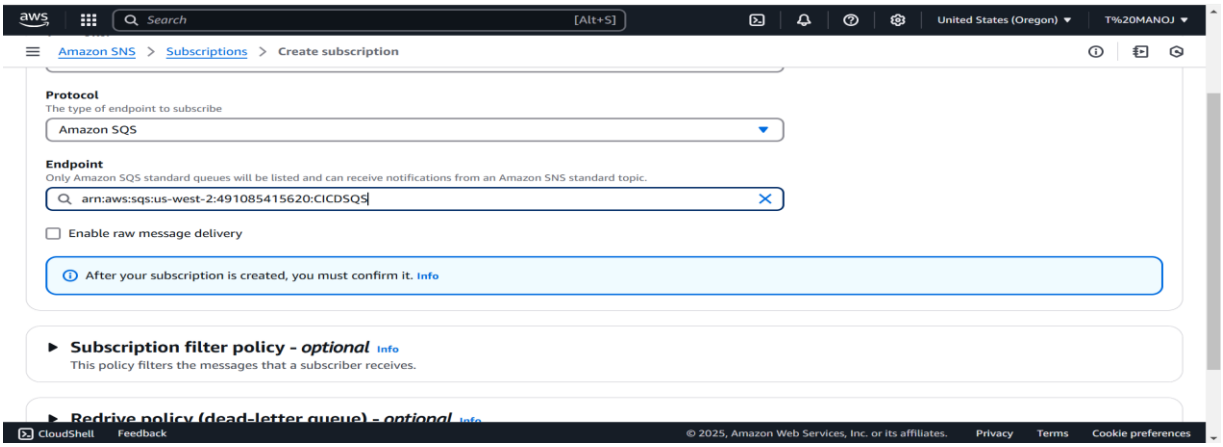
### *Step 1: Create SNS Topic*

1. Go to SNS in the AWS console.
2. Click Create Topic.
3. Type: Standard.
4. Name: CICDSNS.
5. Click Create Topic.
6. Subscribe with your email.



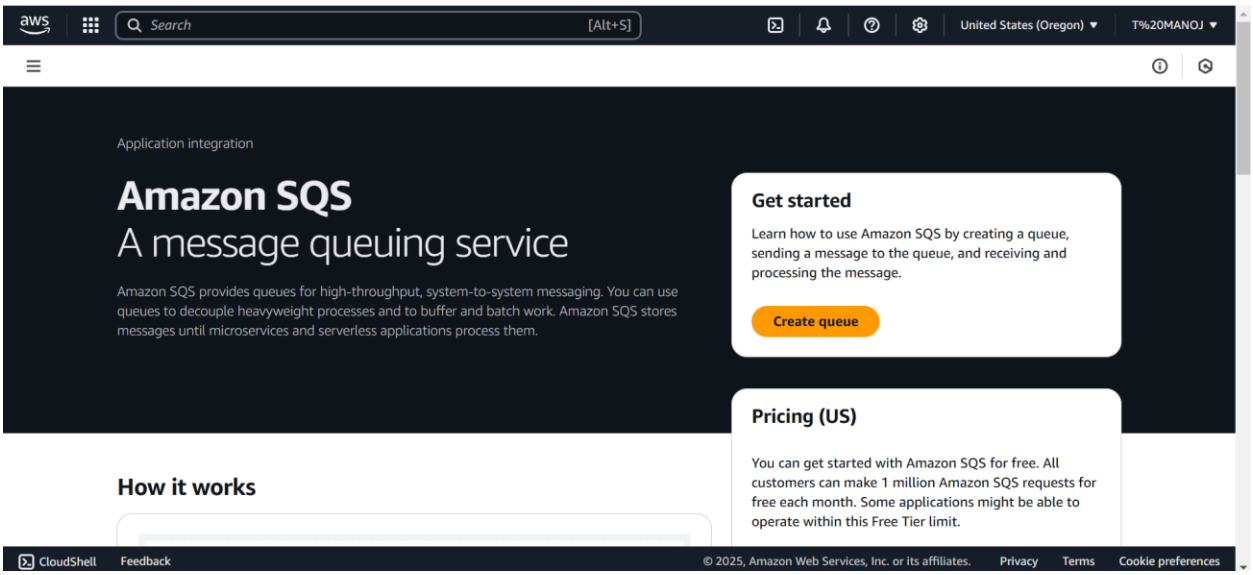

Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Amazon SNS > Topics > Create topic
New Feature
Amazon SNS now supports High Throughput FIFO topics. Learn more
Create topic
Details
Type Info
Topic type cannot be modified after topic is created
FIFO (first-in, first-out)
Strictly-preserved message ordering
Exactly-once message delivery
Subscription protocols: SQS
Standard
Best-effort message ordering
At-least once message delivery
Subscription protocols: SQS, Lambda, Data Firehose, HTTP, SMS, email, mobile application endpoints
Name
CICDSNS
Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (_).
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

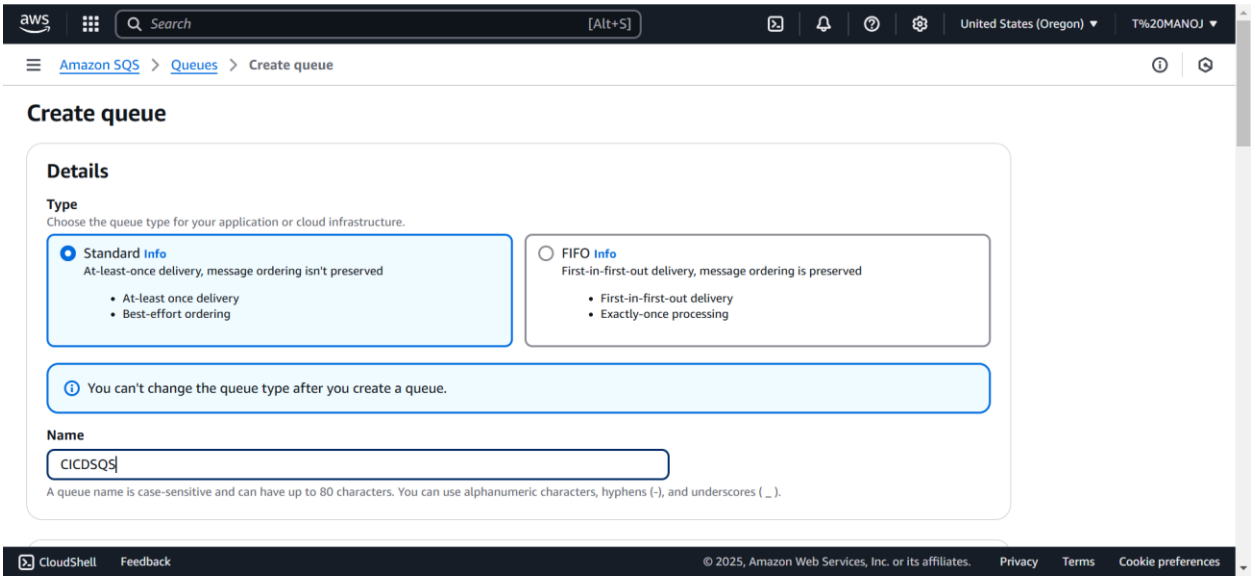

Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Amazon SNS > Subscriptions > Create subscription
Details
Topic ARN
arn:aws:sns:us-west-2:491085415620:CICDSNS
Protocol
The type of endpoint to subscribe
Email
Endpoint
An email address that can receive notifications from Amazon SNS.
dreamers2k22@gmail.com
After your subscription is created, you must confirm it. Info
Subscription filter policy - optional Info
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

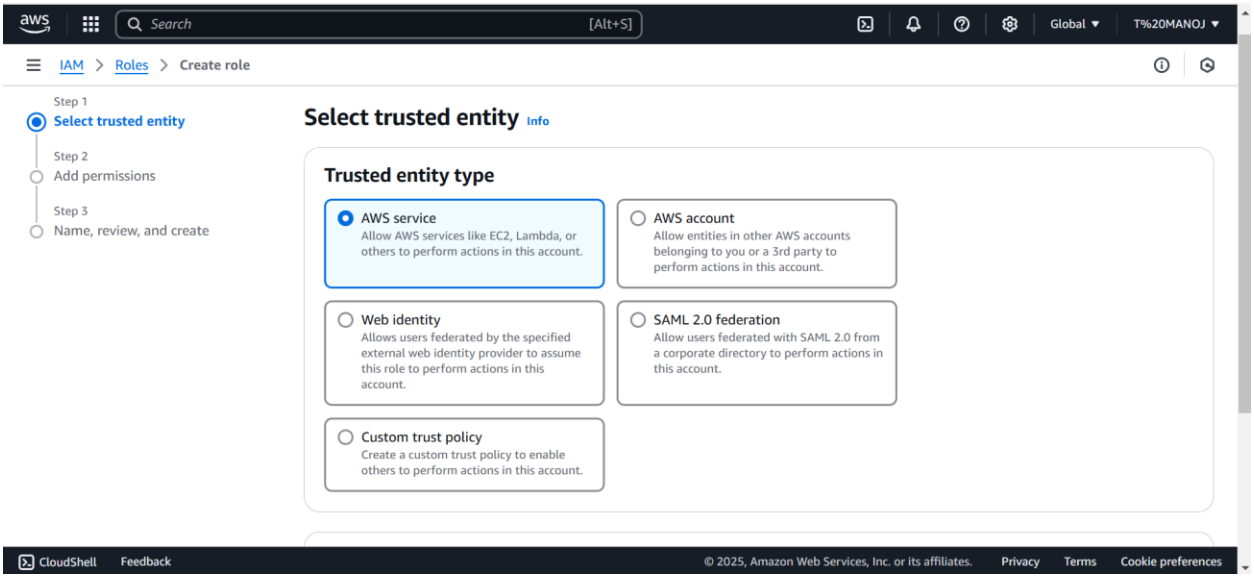



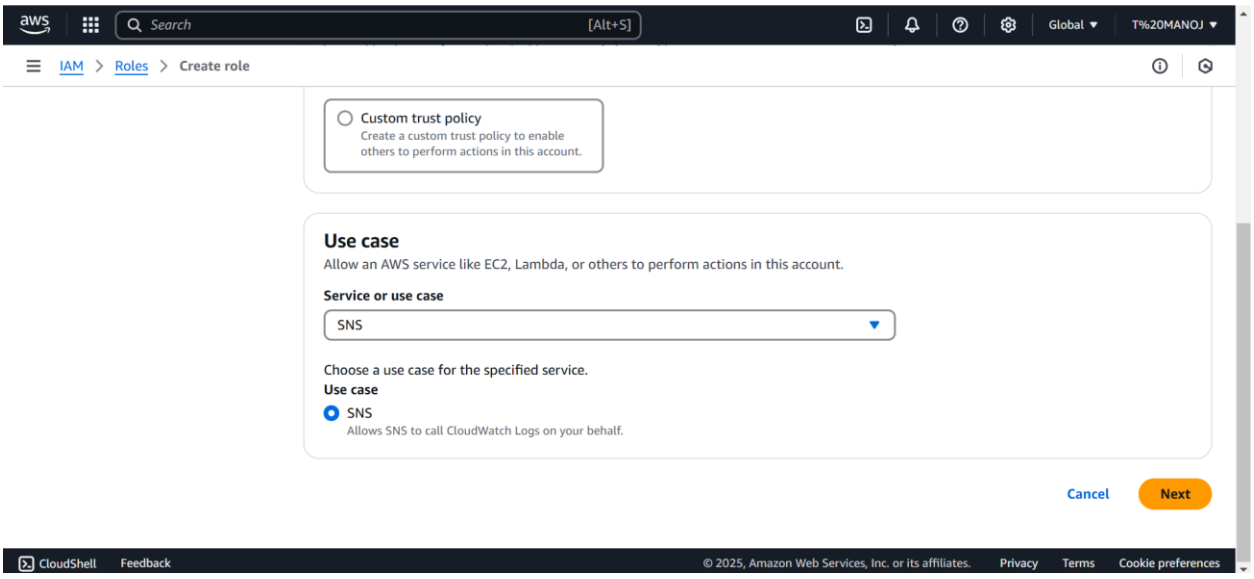

### *Step 2: Create SQS Queue*

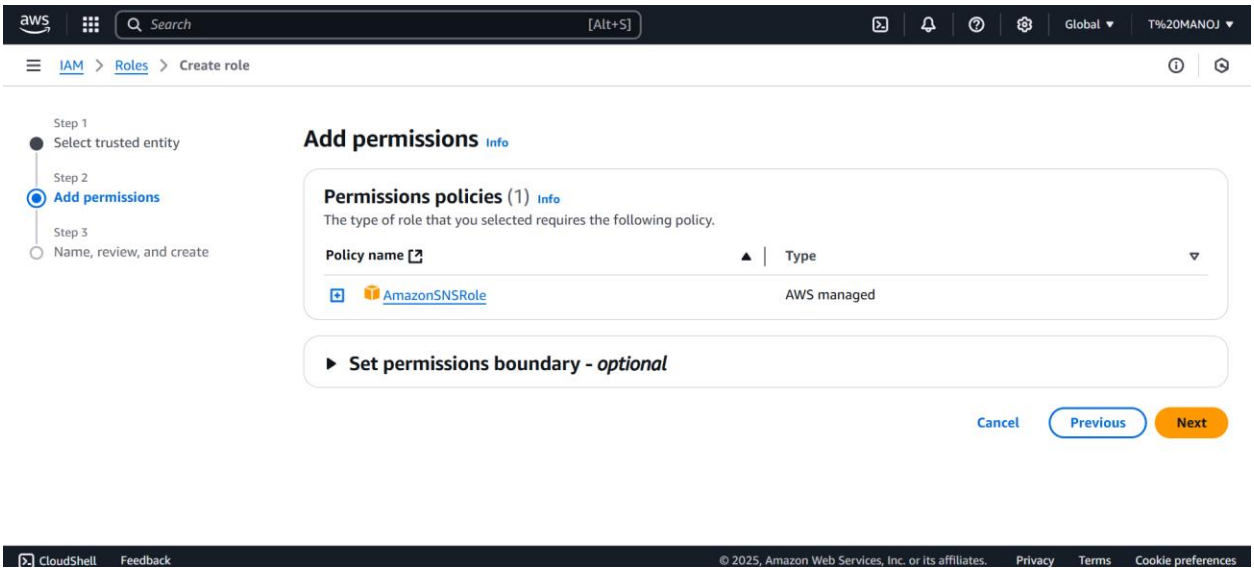
1. Navigate to SQS.
2. Click Create Queue → Standard Queue.
3. Create the role to communicate with SNS
4. Name: CICDSQS.
5. Use default settings and click Create Queue.

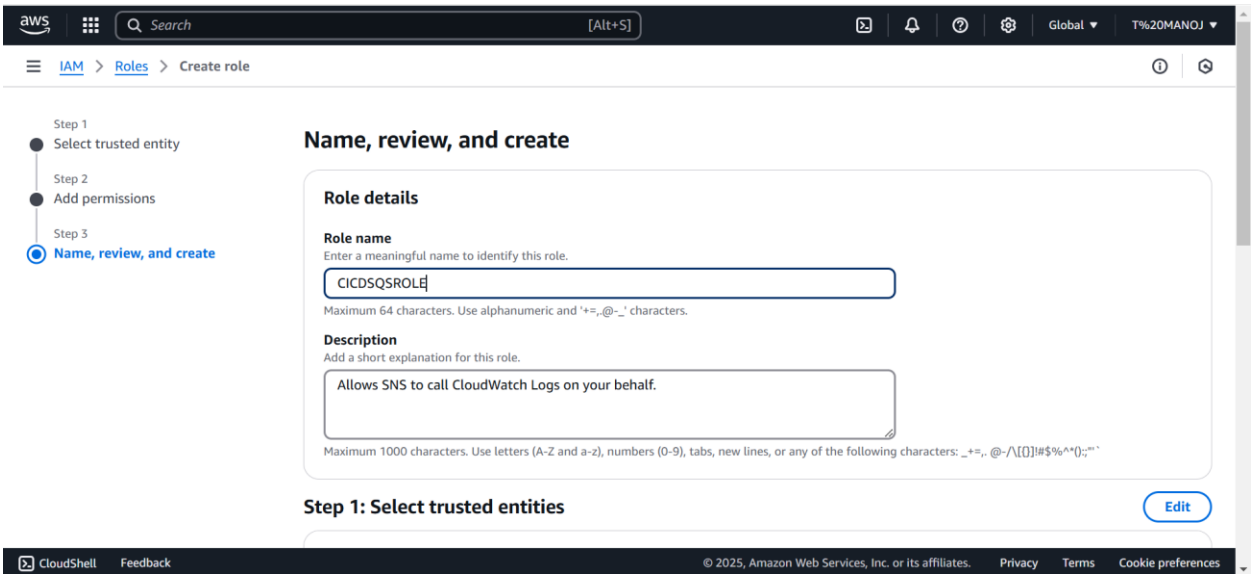

aws
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Application integration
Amazon SQS
A message queuing service
Amazon SQS provides queues for high-throughput, system-to-system messaging. You can use queues to decouple heavyweight processes and to buffer and batch work. Amazon SQS stores messages until microservices and serverless applications process them.
Get started
Learn how to use Amazon SQS by creating a queue, sending a message to the queue, and receiving and processing the message.
Create queue
Pricing (US)
You can get started with Amazon SQS for free. All customers can make 1 million Amazon SQS requests for free each month. Some applications might be able to operate within this Free Tier limit.
How it works
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

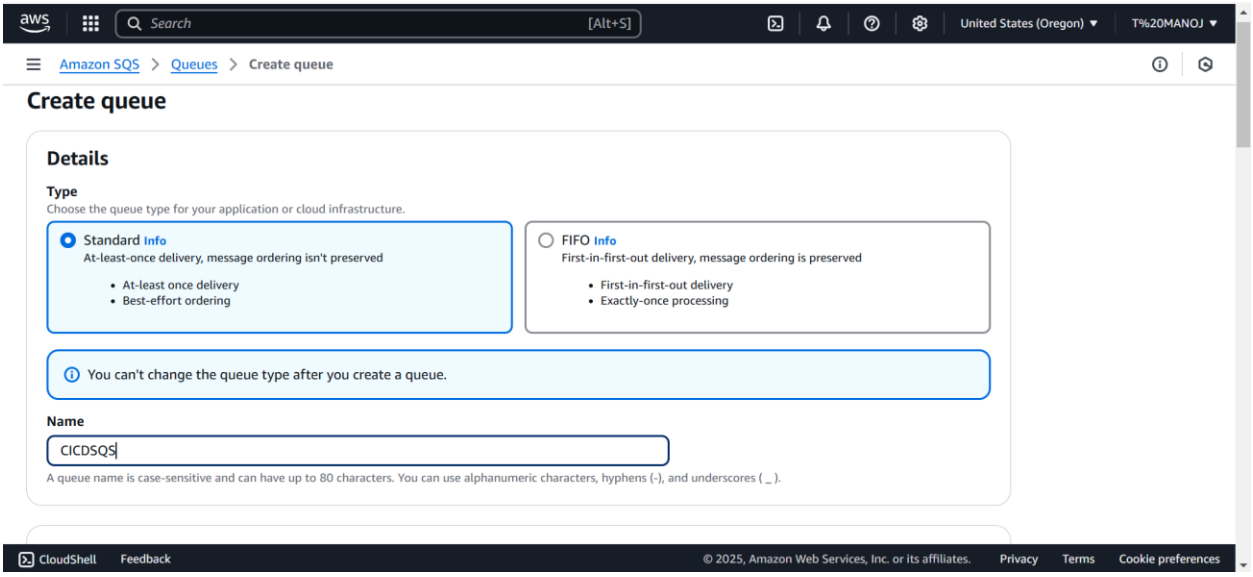

aws
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Amazon SQS
Queues
Create queue
Create queue
Details
Type
Choose the queue type for your application or cloud infrastructure.
Standard Info
At-least-once delivery, message ordering isn't preserved
At-least once delivery
Best-effort ordering
FIFO Info
First-in-first-out delivery, message ordering is preserved
First-in-first-out delivery
Exactly-once processing
You can't change the queue type after you create a queue.
Name
CICDSQS
A queue name is case-sensitive and can have up to 80 characters. You can use alphanumeric characters, hyphens (-), and underscores ( _ ).
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

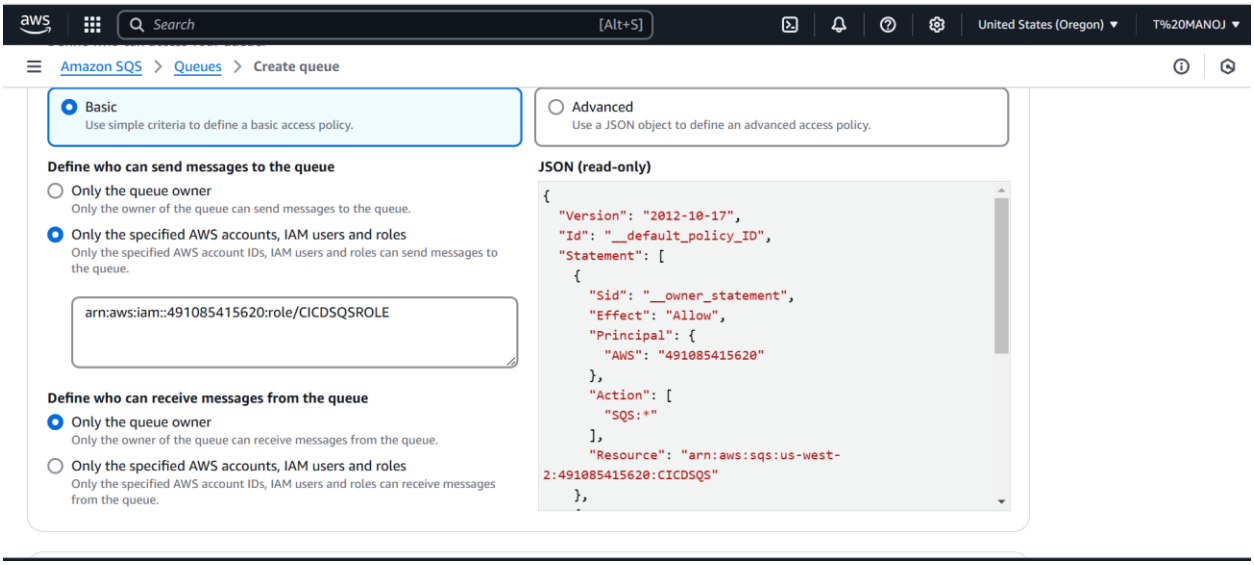

aws
Search
[Alt+S]
Global
T%20MANOJ
IAM
Roles
Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Select trusted entity
Info
Trusted entity type
AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
AWS account
Allow 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.
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

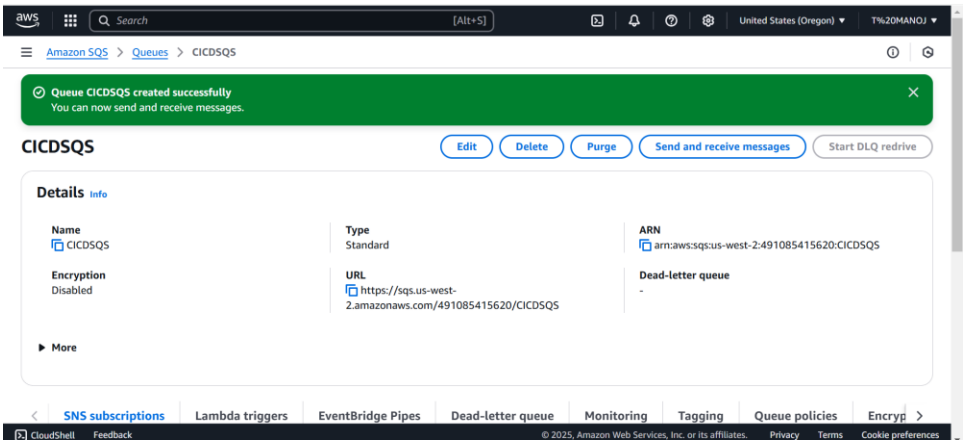

aws
Search
[Alt+S]
Global
T%20MANOJ
IAM
Roles
Create role
Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.
Use case
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.
Service or use case
SNS
Choose a use case for the specified service.
Use case
SNS
Allows SNS to call CloudWatch Logs on your behalf.
Cancel
Next
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Search
[Alt+S]
Global
T%20MANOJ
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Add permissions Info
Permissions policies (1) Info
The type of role that you selected requires the following policy.
Policy name
Type
AmazonSNSRole
AWS managed
Set permissions boundary - optional
Cancel
Previous
Next
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Search
[Alt+S]
Global
T%20MANOJ
IAM > Roles > Create role
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.
CICDSQSROLE
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.
Description
Add a short explanation for this role.
Allows SNS to call CloudWatch Logs 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
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

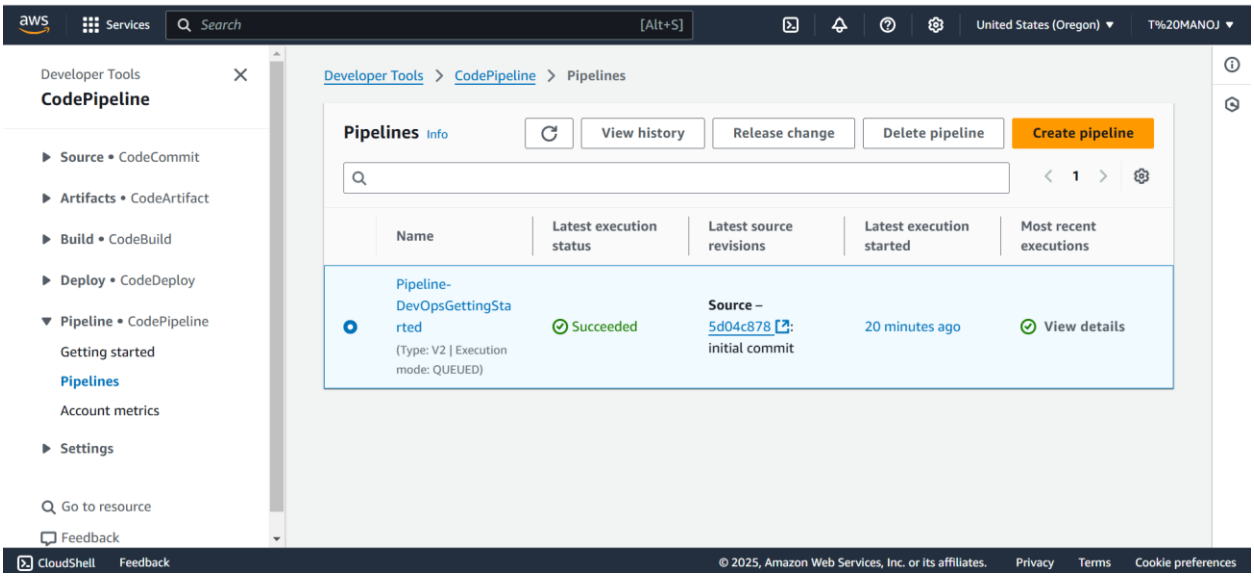

Search [Alt+S]
United States (Oregon)
T%20MANOJ
Amazon SQS > Queues > Create queue
Create queue
Details
Type
Choose the queue type for your application or cloud infrastructure.
Standard Info
At-least-once delivery, message ordering isn't preserved
At-least once delivery
Best-effort ordering
FIFO Info
First-in-first-out delivery, message ordering is preserved
First-in-first-out delivery
Exactly-once processing
You can't change the queue type after you create a queue.
Name
CICDSQS
A queue name is case-sensitive and can have up to 80 characters. You can use alphanumeric characters, hyphens (-), and underscores ( _ ).
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

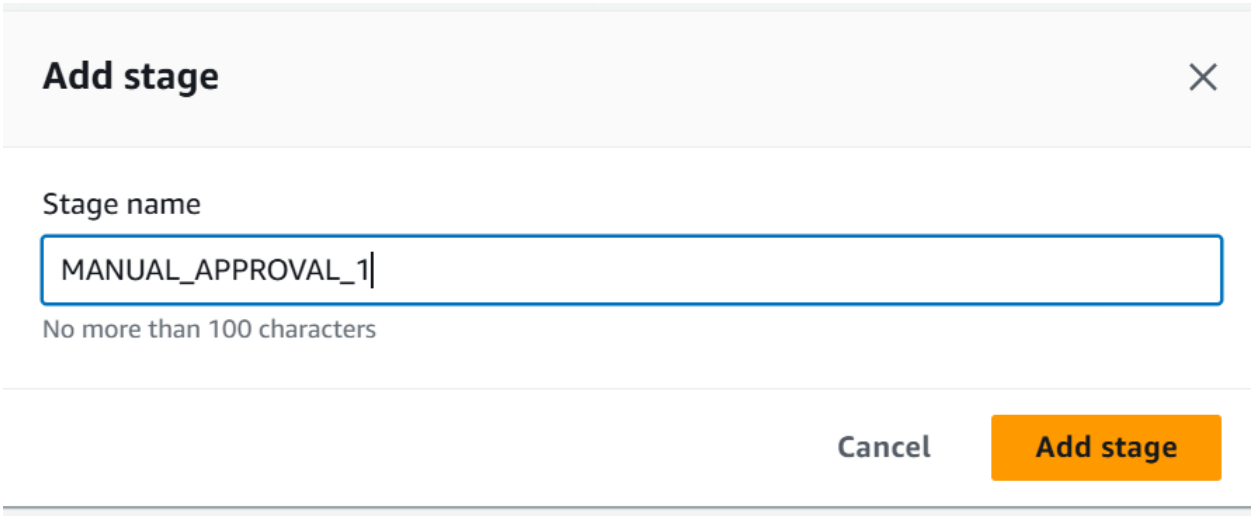

Search [Alt+S]
United States (Oregon)
T%20MANOJ
Amazon SQS > Queues > Create queue
Basic
Use simple criteria to define a basic access policy.
Advanced
Use a JSON object to define an advanced access policy.
Define who can send messages to the queue
Only the queue owner
Only the owner of the queue can send messages to the queue.
Only the specified AWS accounts, IAM users and roles
Only the specified AWS account IDs, IAM users and roles can send messages to the queue.
arn:aws:iam::491085415620:role/CICDSQSROLE
Define who can receive messages from the queue
Only the queue owner
Only the owner of the queue can receive messages from the queue.
Only the specified AWS accounts, IAM users and roles
Only the specified AWS account IDs, IAM users and roles can receive messages from the queue.
JSON (read-only)
{
"Version": "2012-10-17",
"Id": "__default_policy_ID",
"Statement": [
{
"Sid": "__owner_statement",
"Effect": "Allow",
"Principal": {
"AWS": "491085415620"
},
"Action": [
"SQS:*"
],
"Resource": "arn:aws:sqs:us-west-2:491085415620:CICDSQS"
},


Search [Alt+S]
United States (Oregon)
T%20MANOJ
Amazon SQS > Queues > CICDSQS
Queue CICDSQS created successfully
You can now send and receive messages.
CICDSQS
Edit
Delete
Purge
Send and receive messages
Start DLQ redrive
Details Info
Name
CICDSQS
Type
Standard
ARN
arn:aws:sqs:us-west-2:491085415620:CICDSQS
Encryption
Disabled
URL
https://sqs.us-west-2.amazonaws.com/491085415620/CICDSQS
Dead-letter queue
-
More
SNS subscriptions
Lambda triggers
EventBridge Pipes
Dead-letter queue
Monitoring
Tagging
Queue policies
Encryp
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

### *Step 3: Add Manual Approval Stage*

1. Go to AWS CodePipeline → Select Pipeline-DevOpsGettingStarted → Edit.
2. Add a Stage before Deploy named `ApprovalStage`.
3. Add an Action of type Manual Approval.
4. Select the SNS topic for notification.



## Edit action

Action name

Choose a name for your action

MANUALAPPROVAL

No more than 100 characters

Action provider

Manual approval

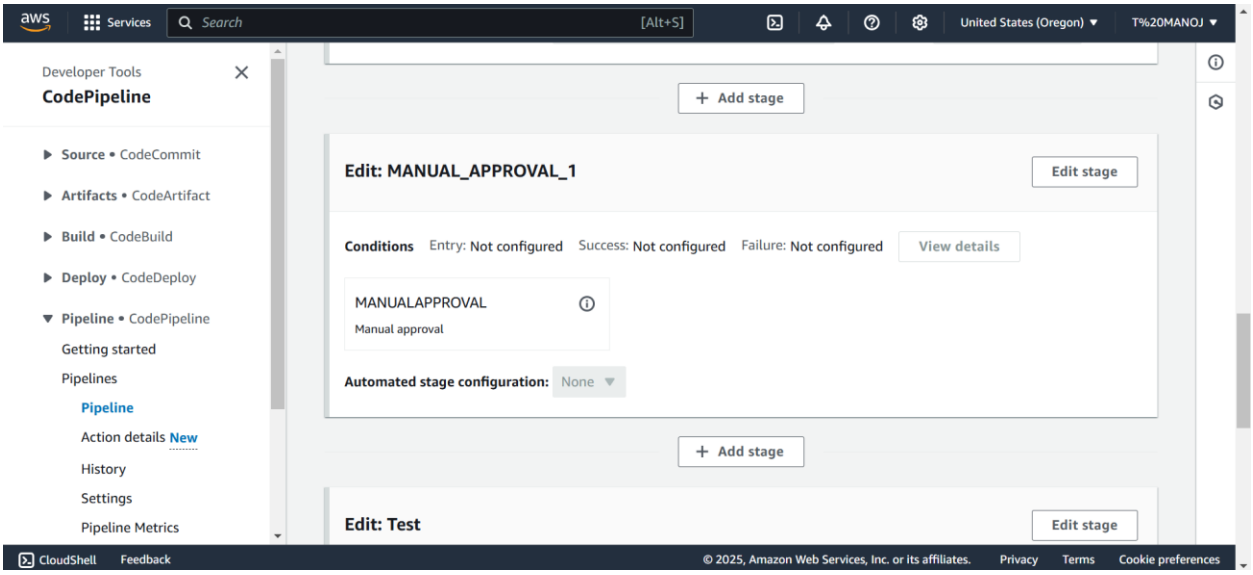
Configure the approval request.

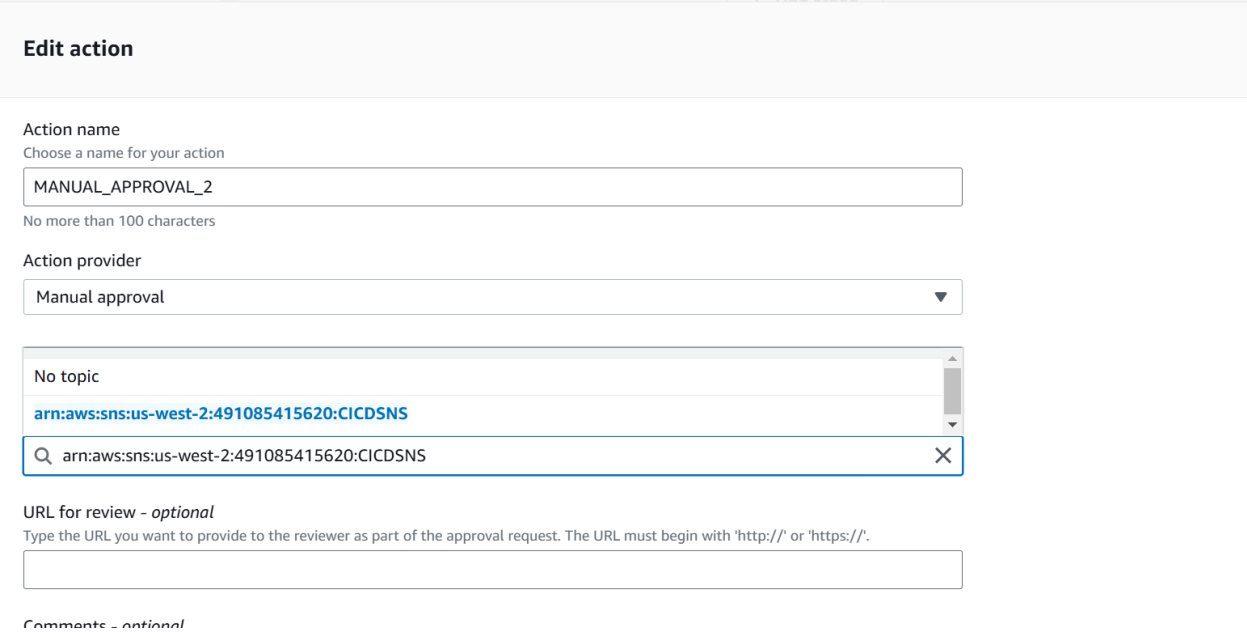
SNS topic ARN - *optional*

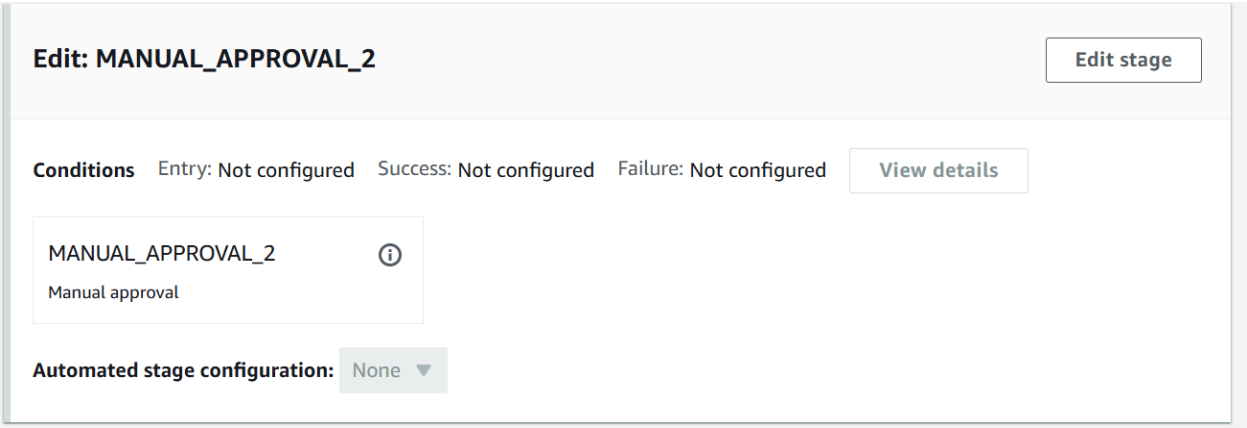
arn:aws:sns:us-west-2:491085415620:CICDSNS

URL for review - *optional*

Type the URL you want to provide to the reviewer as part of the approval request. The URL must begin with 'http://' or 'https://'.

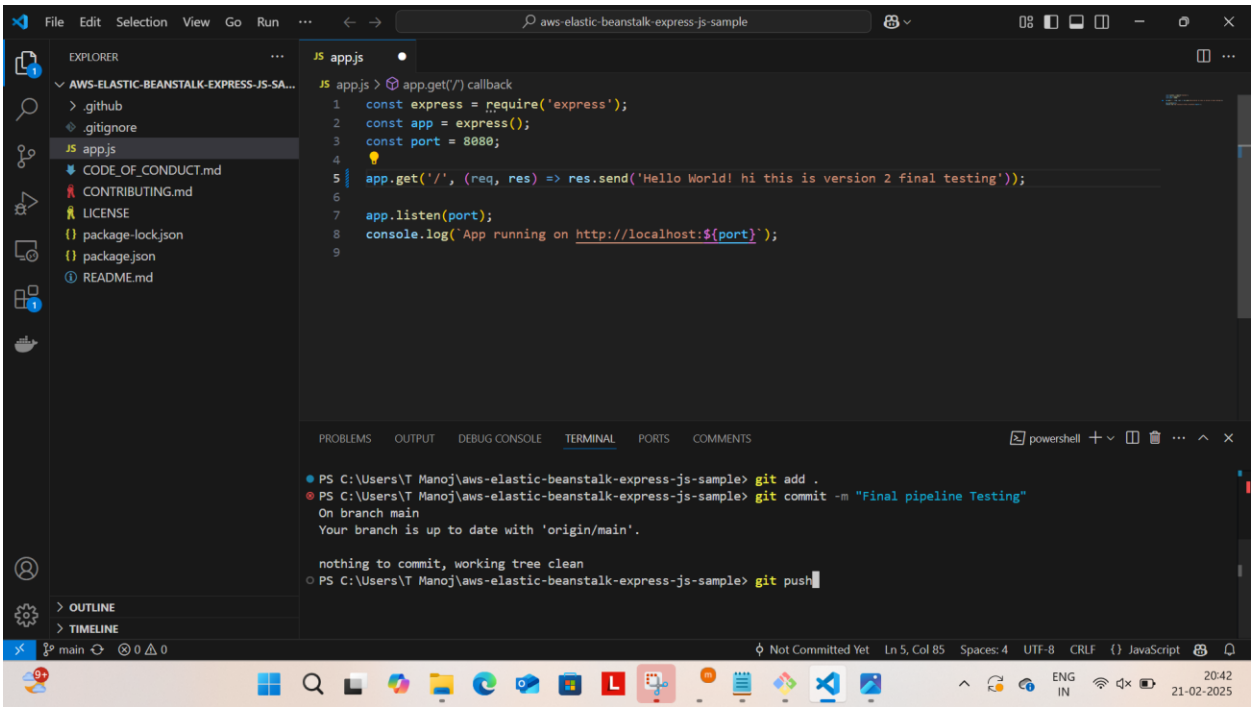


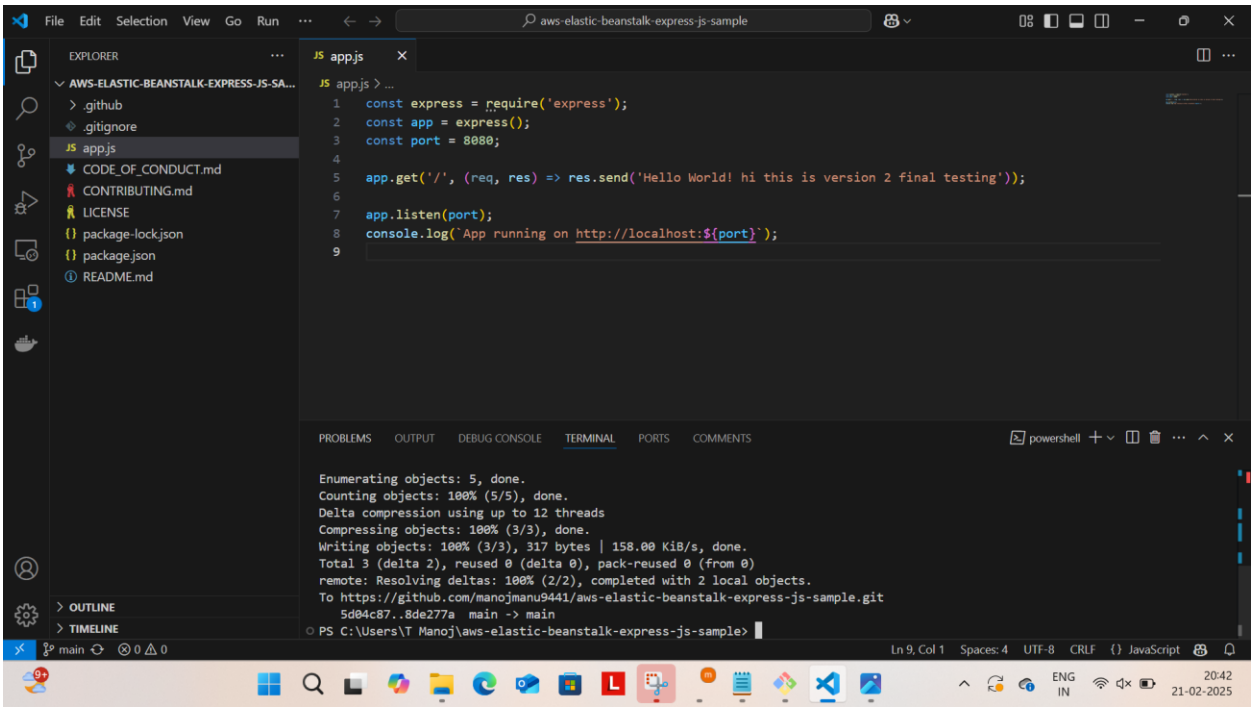


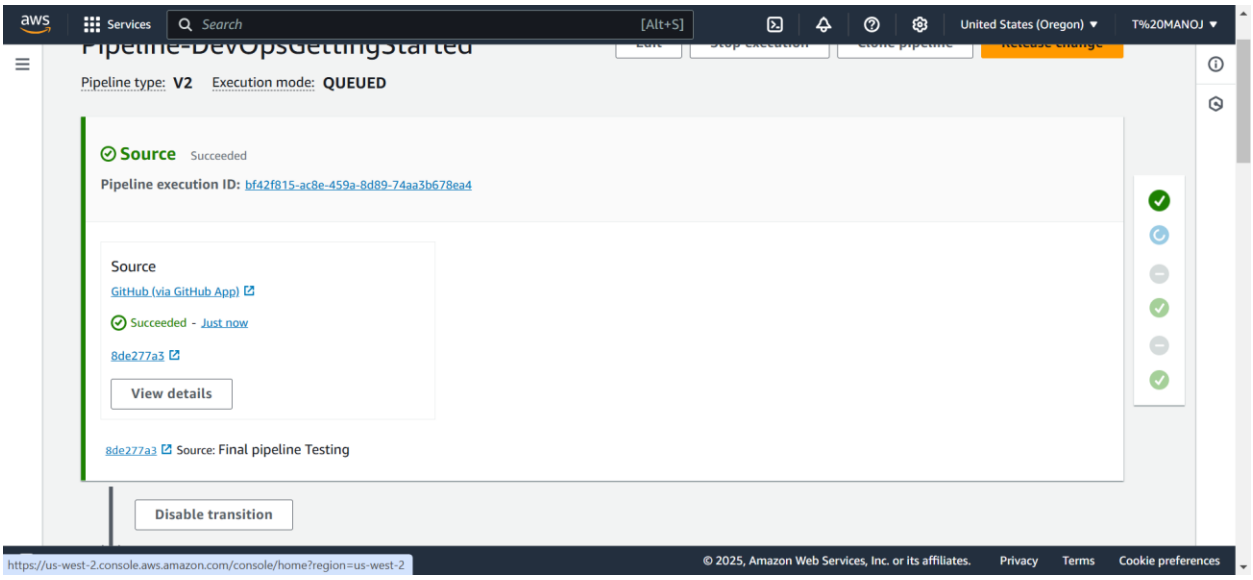



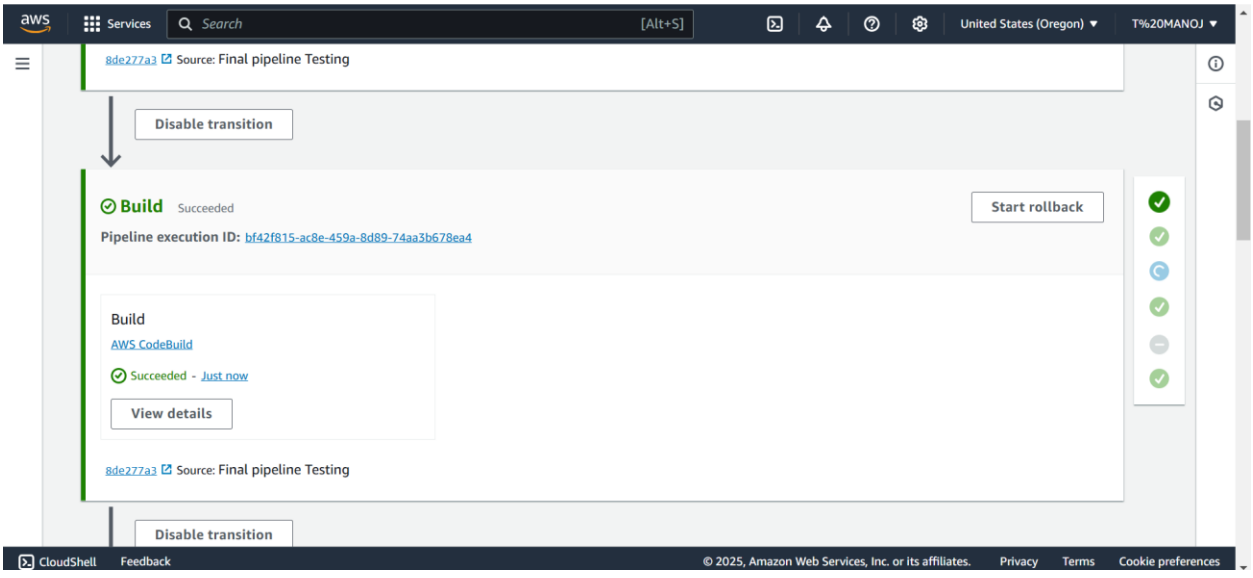

### *Step 4: Test Manual Approval Process*

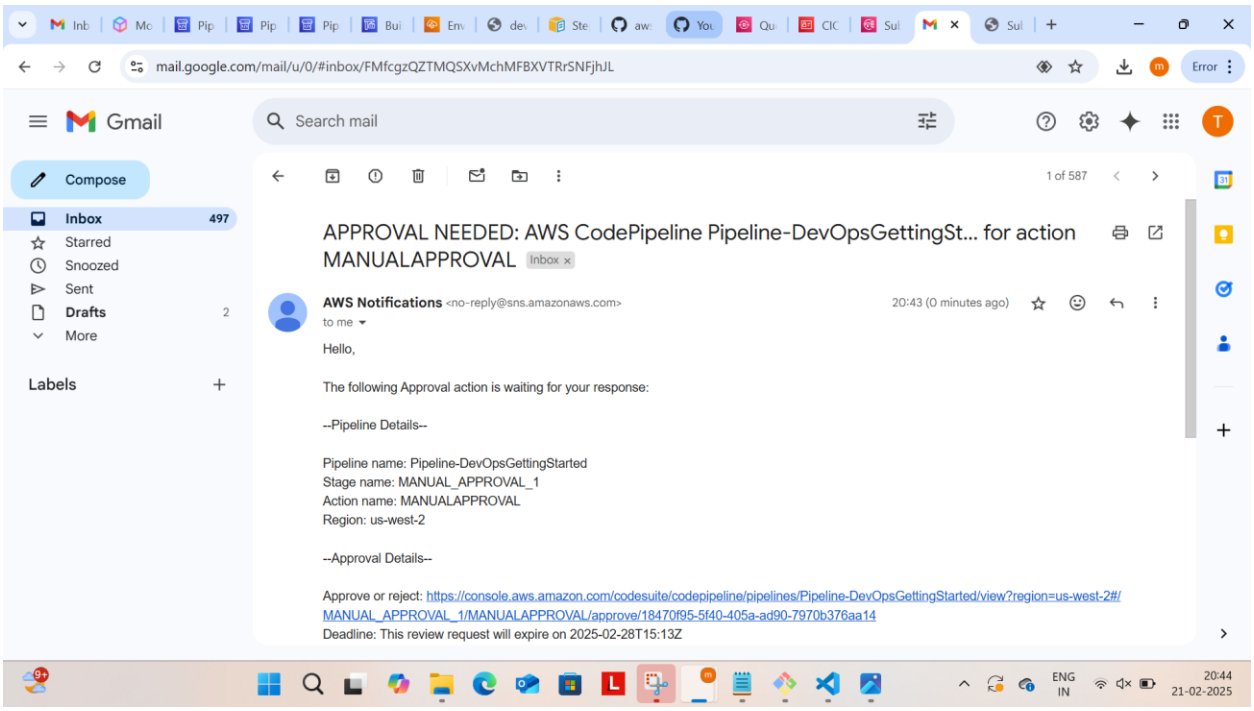
1. Make a code change and push to GitHub.
2. The pipeline stops at the Approval Stage.
3. Approve/Reject via the AWS Console or SNS email link.
4. Approve: Pipeline deploys.
5. Reject: Deployment rolls back automatically.

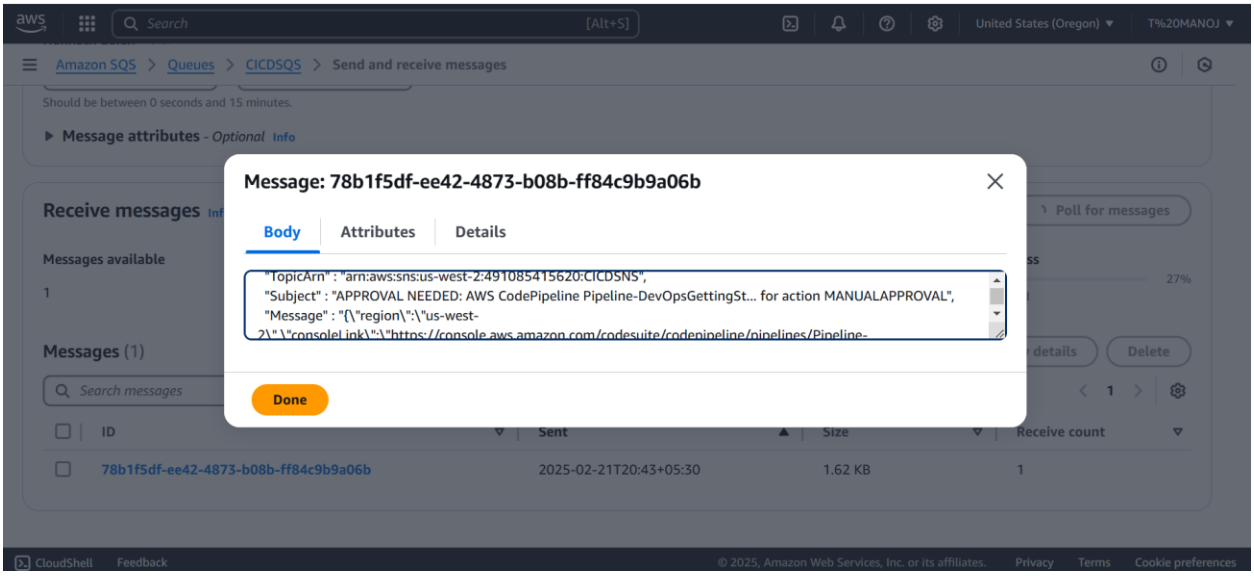

File Edit Selection View Go Run
aws-elastic-beanstalk-express-js-sample
EXPLORER
AWS-ELASTIC-BEANSTALK-EXPRESS-JS-SA...
.github
.gitignore
app.js
CODE_OF_CONDUCT.md
CONTRIBUTING.md
LICENSE
package-lock.json
package.json
README.md
app.js
app.js > app.get('/') callback
1 const express = require('express');
2 const app = express();
3 const port = 8080;
4
5 app.get('/', (req, res) => res.send('Hello World! hi this is version 2 final testing'));
6
7 app.listen(port);
8 console.log(`App running on http://localhost:${port}`);
9
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
powershell
PS C:\Users\T Manoj\aws-elastic-beanstalk-express-js-sample> git add .
PS C:\Users\T Manoj\aws-elastic-beanstalk-express-js-sample> git commit -m "Final pipeline Testing"
On branch main
Your branch is up to date with 'origin/main'.
nothing to commit, working tree clean
PS C:\Users\T Manoj\aws-elastic-beanstalk-express-js-sample> git push
OUTLINE
TIMELINE
main
Not Committed Yet
Ln 5, Col 85
Spaces: 4
UTF-8
CRLF
JavaScript
ENG
IN
20:42
21-02-2025

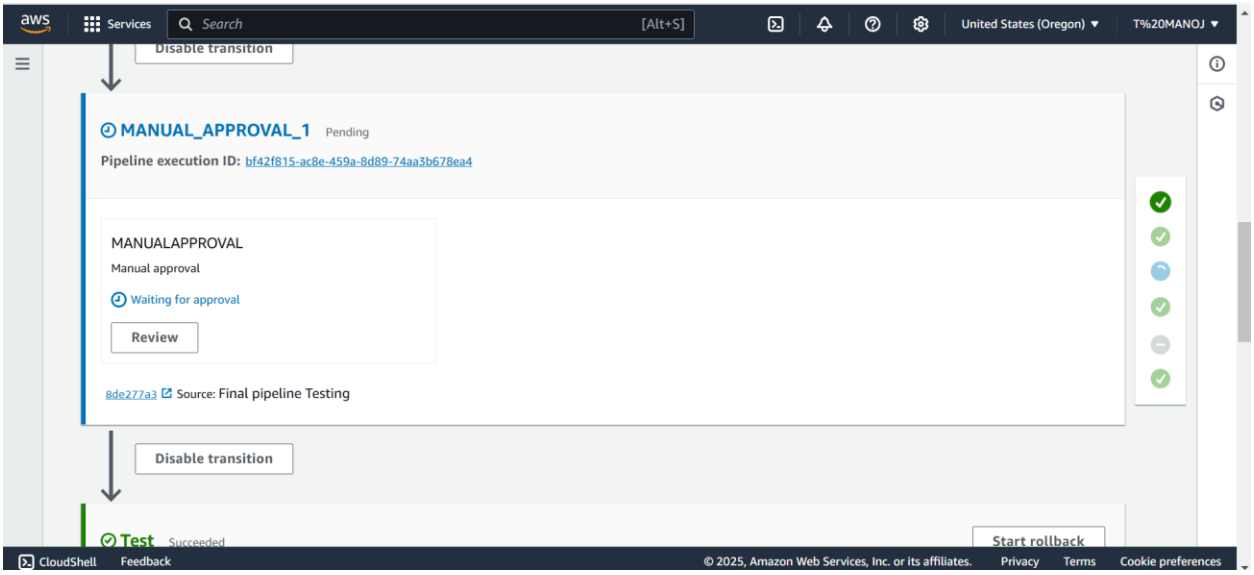

File Edit Selection View Go Run
aws-elastic-beanstalk-express-js-sample
EXPLORER
AWS-ELASTIC-BEANSTALK-EXPRESS-JS-SA...
.github
.gitignore
app.js
CODE_OF_CONDUCT.md
CONTRIBUTING.md
LICENSE
package-lock.json
package.json
README.md
app.js
app.js > ...
1 const express = require('express');
2 const app = express();
3 const port = 8080;
4
5 app.get('/', (req, res) => res.send('Hello World! hi this is version 2 final testing'));
6
7 app.listen(port);
8 console.log(`App running on http://localhost:${port}`);
9
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
powershell
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 317 bytes | 158.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/manojmanu9441/aws-elastic-beanstalk-express-js-sample.git
5d04c87..8de277a main -> main
PS C:\Users\T Manoj\aws-elastic-beanstalk-express-js-sample>
OUTLINE
TIMELINE
main
Ln 9, Col 1
Spaces: 4
UTF-8
CRLF
JavaScript
ENG
IN
20:42
21-02-2025

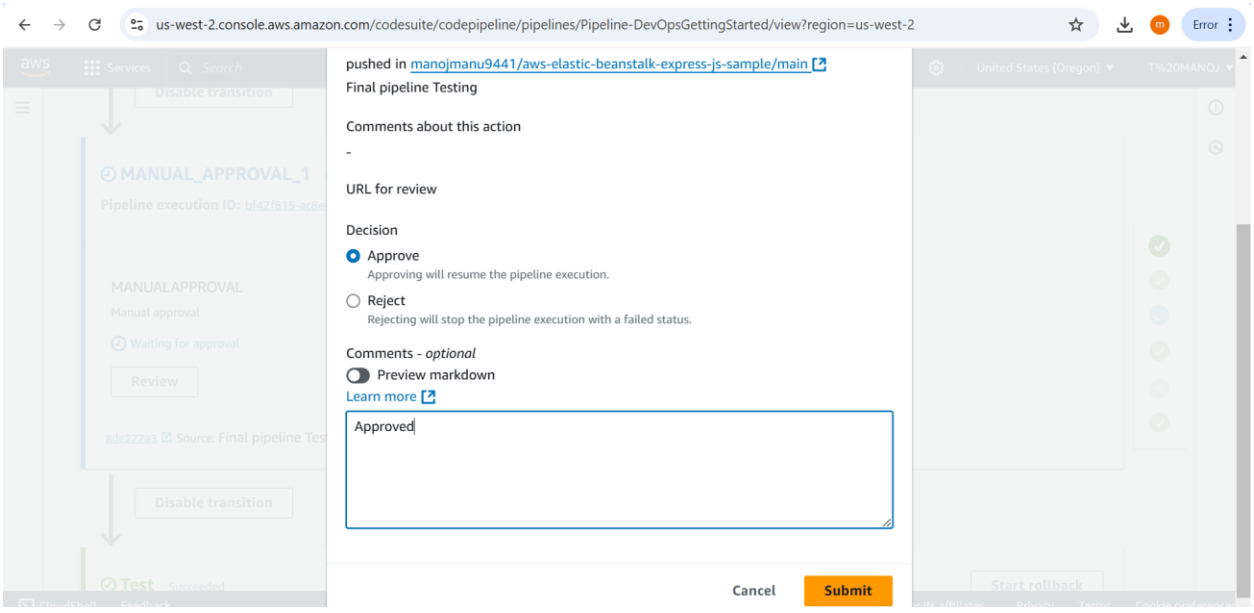

aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Pipeline type: V2
Execution mode: QUEUED
Source
Succeeded
Pipeline execution ID: bf42f815-ac8e-459a-8d89-74aa3b678ea4
Source
GitHub (via GitHub App)
Succeeded - Just now
8de277a3
View details
8de277a3 Source: Final pipeline Testing
Disable transition
https://us-west-2.console.aws.amazon.com/console/home?region=us-west-2
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

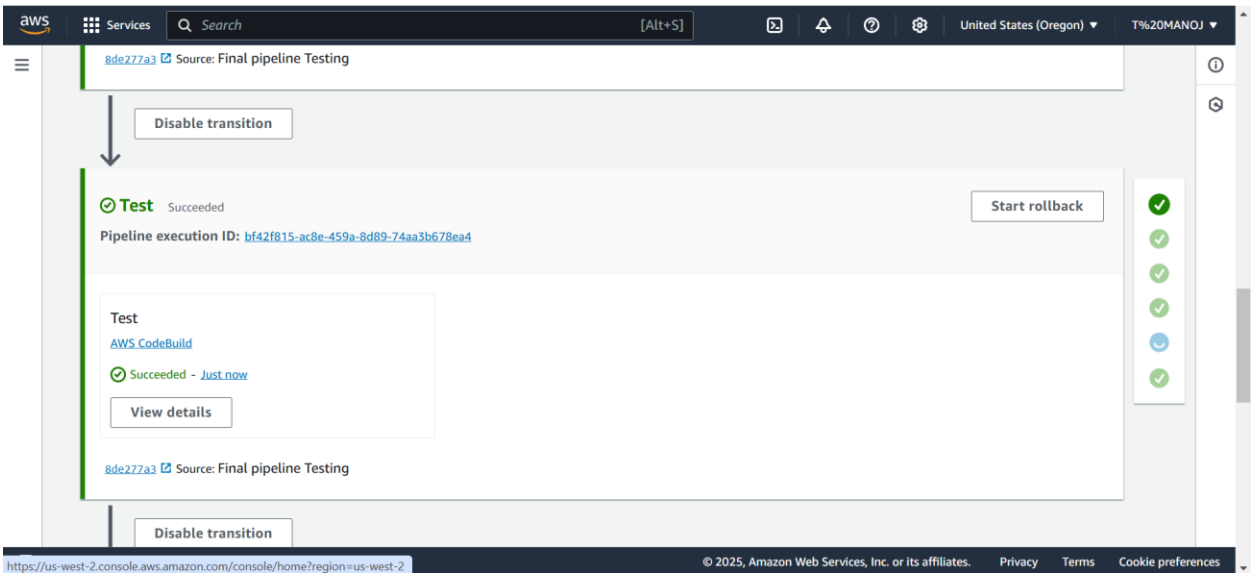

aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
8de277a3 Source: Final pipeline Testing
Disable transition
Build
Succeeded
Start rollback
Pipeline execution ID: bf42f815-ac8e-459a-8d89-74aa3b678ea4
Build
AWS CodeBuild
Succeeded - Just now
View details
8de277a3 Source: Final pipeline Testing
Disable transition
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

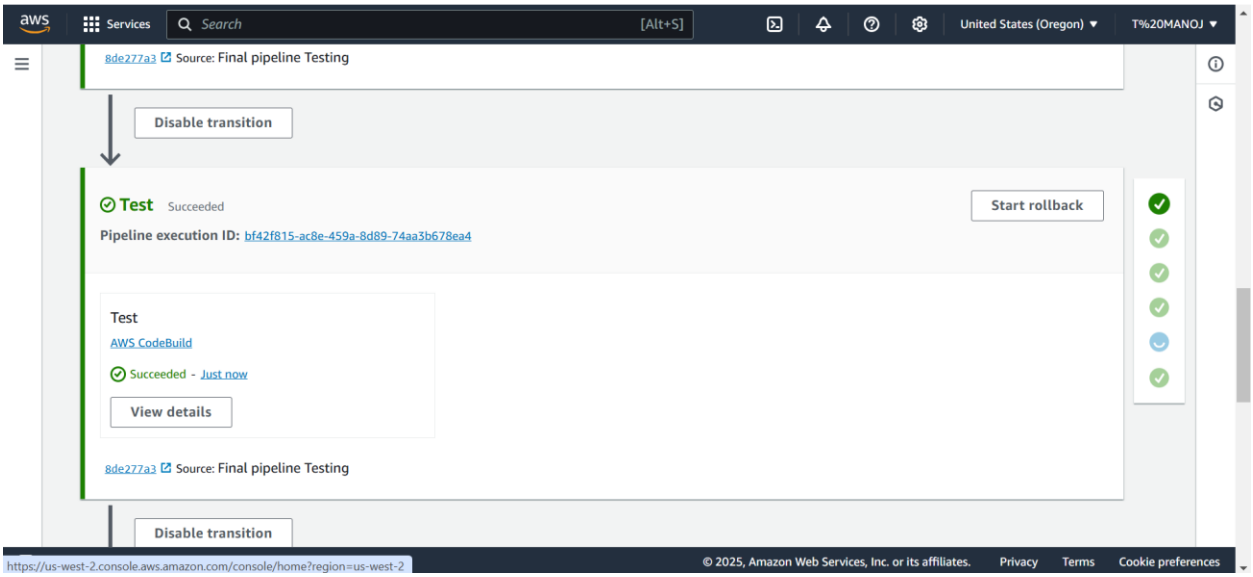

mail.google.com/mail/u/0/#inbox/FMfcgzQZTMQSXvMchMFBXVTRrSNFjhJL
Gmail
Search mail
Compose
Inbox 497
Starred
Snoozed
Sent
Drafts 2
More
Labels
1 of 587
APPROVAL NEEDED: AWS CodePipeline Pipeline-DevOpsGettingSt... for action MANUALAPPROVAL
Inbox
AWS Notifications <no-reply@sns.amazonaws.com>
to me
20:43 (0 minutes ago)
Hello,
The following Approval action is waiting for your response:
--Pipeline Details--
Pipeline name: Pipeline-DevOpsGettingStarted
Stage name: MANUAL_APPROVAL_1
Action name: MANUALAPPROVAL
Region: us-west-2
--Approval Details--
Approve or reject: https://console.aws.amazon.com/codesuite/codepipeline/pipelines/Pipeline-DevOpsGettingStarted/view?region=us-west-2#/MANUAL_APPROVAL_1/MANUALAPPROVAL/approve/18470f95-5f40-405a-ad90-7970b376aa14
Deadline: This review request will expire on 2025-02-28T15:13Z
ENG IN
20:44
21-02-2025

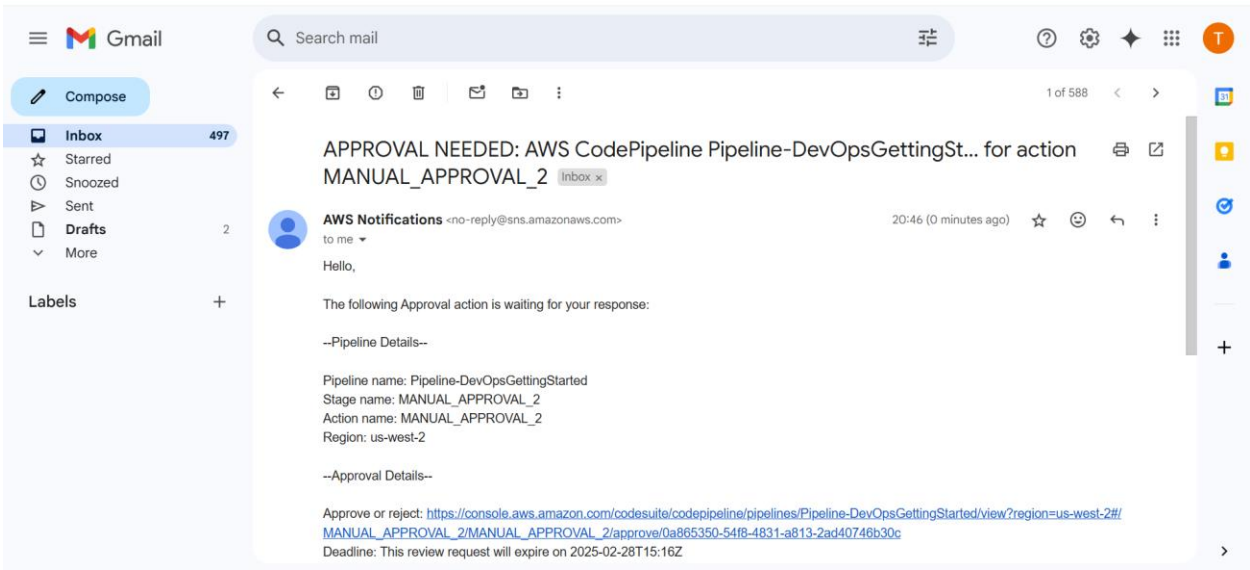

Amazon SQS > Queues > CICDSQS > Send and receive messages
Should be between 0 seconds and 15 minutes.
Message attributes - Optional Info
Receive messages
Messages available
1
Poll for messages
Messages (1)
Search messages
ID
Sent
Size
Receive count
78b1f5df-ee42-4873-b08b-ff84c9b9a06b
2025-02-21T20:43+05:30
1.62 KB
1
Message: 78b1f5df-ee42-4873-b08b-ff84c9b9a06b
Body
Attributes
Details
"TopicArn" : "arn:aws:sns:us-west-2:491085415620:CICDSNS",
"Subject" : "APPROVAL NEEDED: AWS CodePipeline Pipeline-DevOpsGettingSt... for action MANUALAPPROVAL",
"Message" : "{\"region\":\"us-west-
Done
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

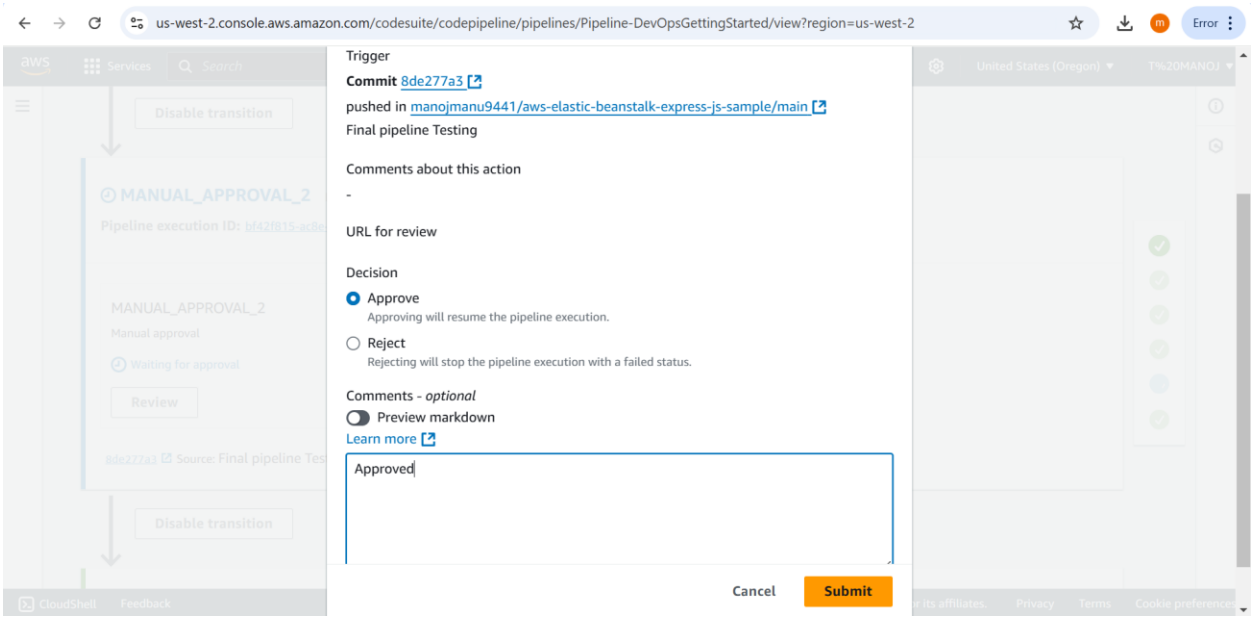

aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Disable transition
MANUAL_APPROVAL_1
Pending
Pipeline execution ID: bf42f815-ac8e-459a-8d89-74aa3b678ea4
MANUALAPPROVAL
Manual approval
Waiting for approval
Review
8de277a3
Source: Final pipeline Testing
Disable transition
Test
Succeeded
Start rollback
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

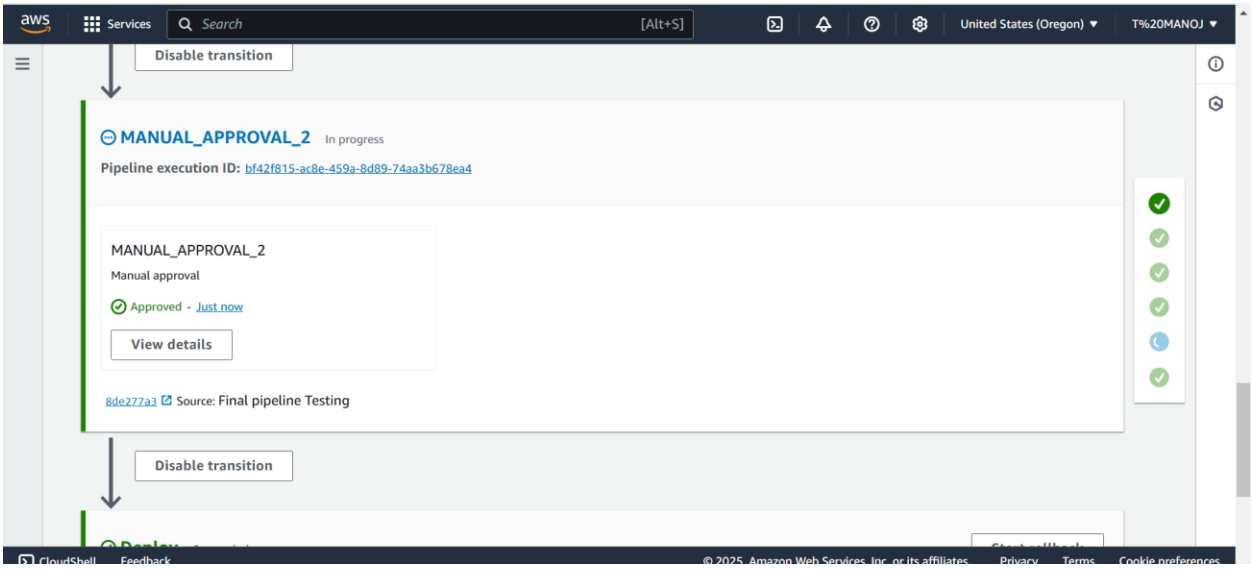

us-west-2.console.aws.amazon.com/codesuite/codepipeline/pipelines/Pipeline-DevOpsGettingStarted/view?region=us-west-2
Error
pushed in manojmanu9441/aws-elastic-beanstalk-express-js-sample/main
Final pipeline Testing
Comments about this action
-
URL for review
Decision
Approve
Approving will resume the pipeline execution.
Reject
Rejecting will stop the pipeline execution with a failed status.
Comments - optional
Preview markdown
Learn more
Approved
Cancel
Submit

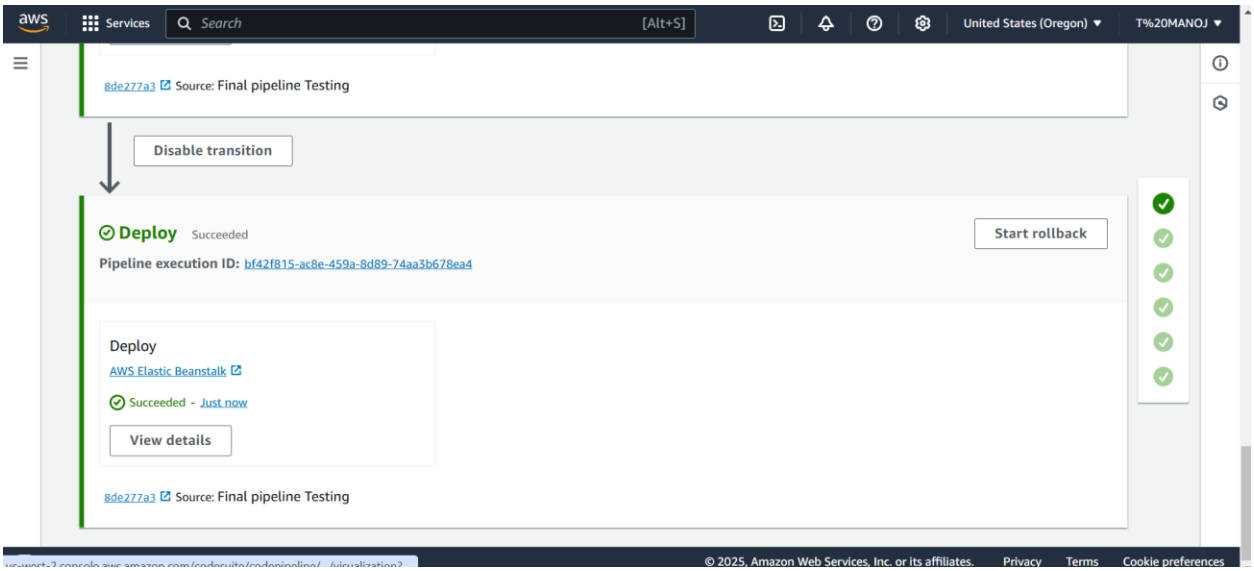

aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
8de277a3 Source: Final pipeline Testing
Disable transition
Test Succeeded
Start rollback
Pipeline execution ID: bf42f815-ac8e-459a-8d89-74aa3b678ea4
Test
AWS CodeBuild
Succeeded - Just now
View details
8de277a3 Source: Final pipeline Testing
Disable transition
https://us-west-2.console.aws.amazon.com/console/home?region=us-west-2
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

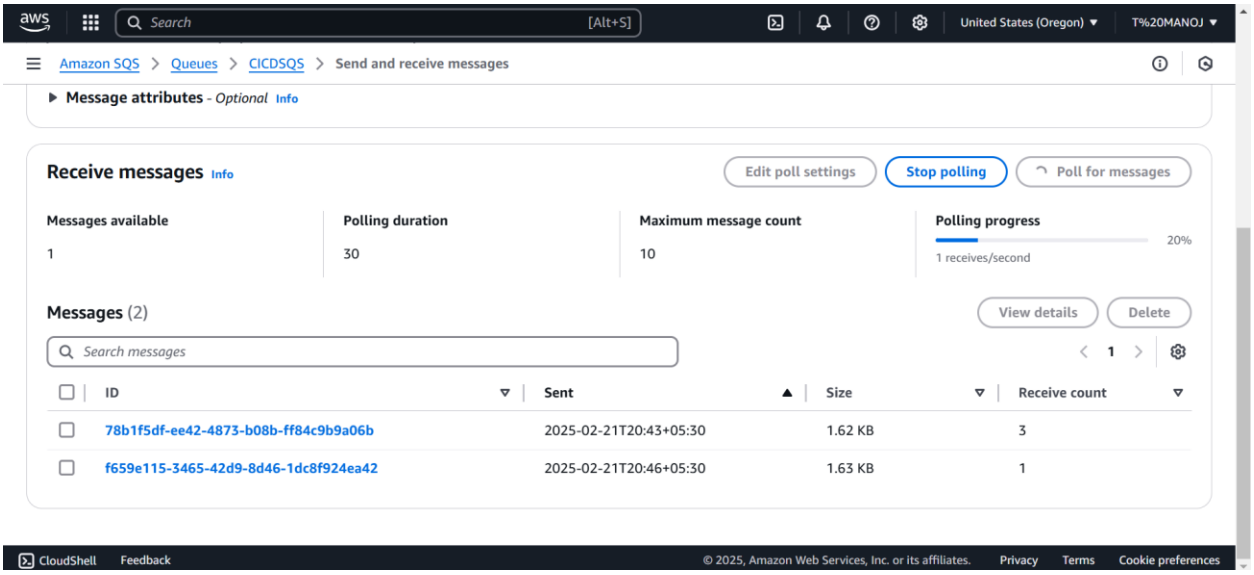

aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
8de277a3 Source: Final pipeline Testing
Disable transition
Test Succeeded
Start rollback
Pipeline execution ID: bf42f815-ac8e-459a-8d89-74aa3b678ea4
Test
AWS CodeBuild
Succeeded - Just now
View details
8de277a3 Source: Final pipeline Testing
Disable transition
https://us-west-2.console.aws.amazon.com/console/home?region=us-west-2
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

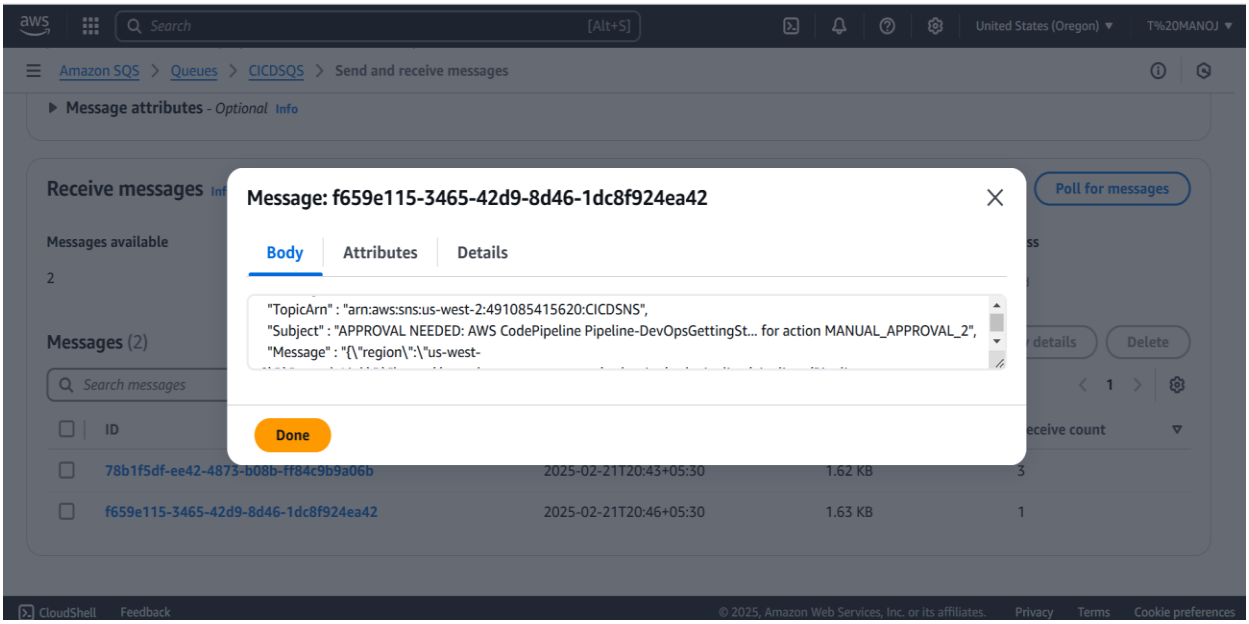

Gmail
Search mail
Compose
Inbox 497
Starred
Snoozed
Sent
Drafts 2
More
Labels
1 of 588
APPROVAL NEEDED: AWS CodePipeline Pipeline-DevOpsGettingSt... for action MANUAL_APPROVAL_2
Inbox
AWS Notifications <no-reply@sns.amazonaws.com>
to me
20:46 (0 minutes ago)
Hello,
The following Approval action is waiting for your response:
--Pipeline Details--
Pipeline name: Pipeline-DevOpsGettingStarted
Stage name: MANUAL_APPROVAL_2
Action name: MANUAL_APPROVAL_2
Region: us-west-2
--Approval Details--
Approve or reject: https://console.aws.amazon.com/codesuite/codepipeline/pipelines/Pipeline-DevOpsGettingStarted/view?region=us-west-2#/MANUAL_APPROVAL_2/MANUAL_APPROVAL_2/approve/0a865350-54f8-4831-a813-2ad40746b30c
Deadline: This review request will expire on 2025-02-28T15:16Z


us-west-2.console.aws.amazon.com/codesuite/codepipeline/pipelines/Pipeline-DevOpsGettingStarted/view?region=us-west-2
Trigger
Commit 8de277a3
pushed in manojmanu9441/aws-elastic-beanstalk-express-js-sample/main
Final pipeline Testing
Comments about this action
-
URL for review
Decision
Approve
Approving will resume the pipeline execution.
Reject
Rejecting will stop the pipeline execution with a failed status.
Comments - optional
Preview markdown
Learn more
Approved
Cancel
Submit


Services
Search
[Alt+S]
United States (Oregon)
Disable transition
MANUAL_APPROVAL_2
In progress
Pipeline execution ID: bf42f815-ac8e-459a-8d89-74aa3b678ea4
MANUAL_APPROVAL_2
Manual approval
Approved - Just now
View details
8de277a3 Source: Final pipeline Testing
Disable transition
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

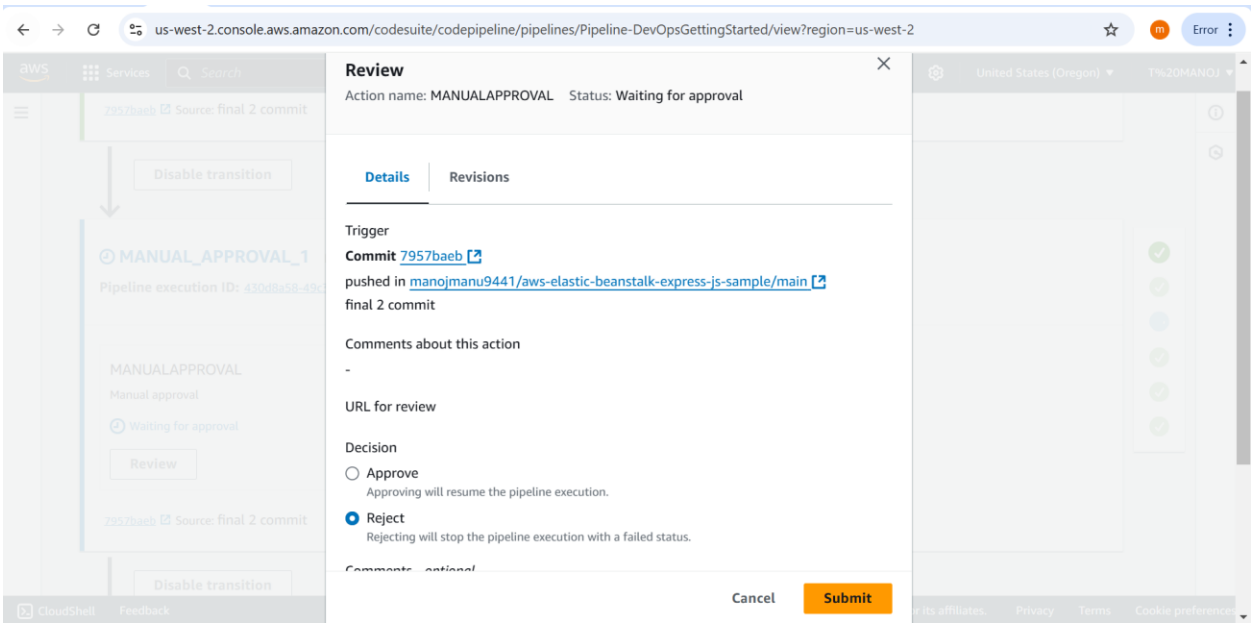

aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
8de277a3 Source: Final pipeline Testing
Disable transition
Deploy Succeeded
Start rollback
Pipeline execution ID: bf42f815-ac8e-459a-8d89-74aa3b678ea4
Deploy
AWS Elastic Beanstalk
Succeeded - Just now
View details
8de277a3 Source: Final pipeline Testing
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

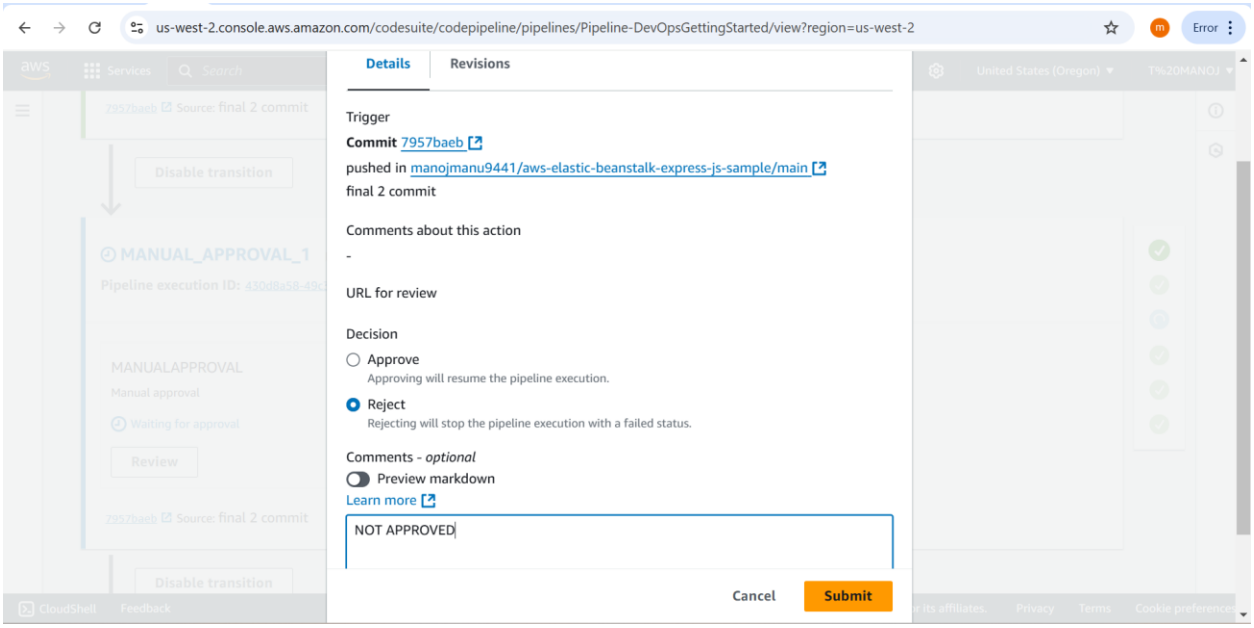

aws
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Amazon SQS > Queues > CICDSQS > Send and receive messages
Message attributes - Optional Info
Receive messages Info
Edit poll settings
Stop polling
Poll for messages
Messages available
1
Polling duration
30
Maximum message count
10
Polling progress
20%
1 receives/second
Messages (2)
View details
Delete
Search messages
1
ID
Sent
Size
Receive count
78b1f5df-ee42-4873-b08b-ff84c9b9a06b
2025-02-21T20:43+05:30
1.62 KB
3
f659e115-3465-42d9-8d46-1dc8f924ea42
2025-02-21T20:46+05:30
1.63 KB
1
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

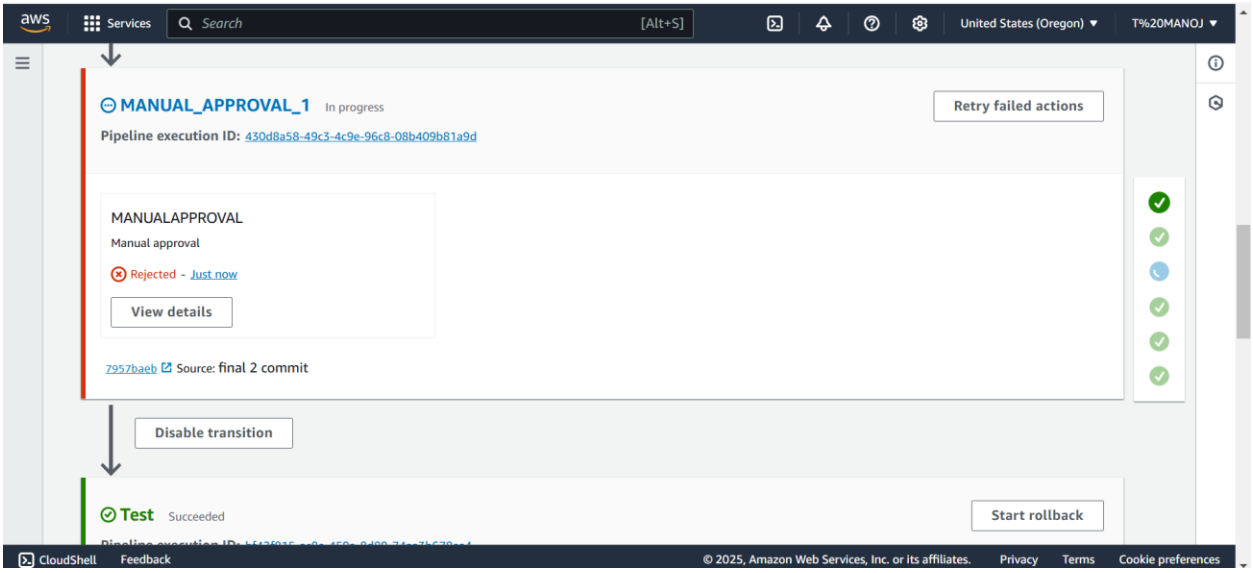


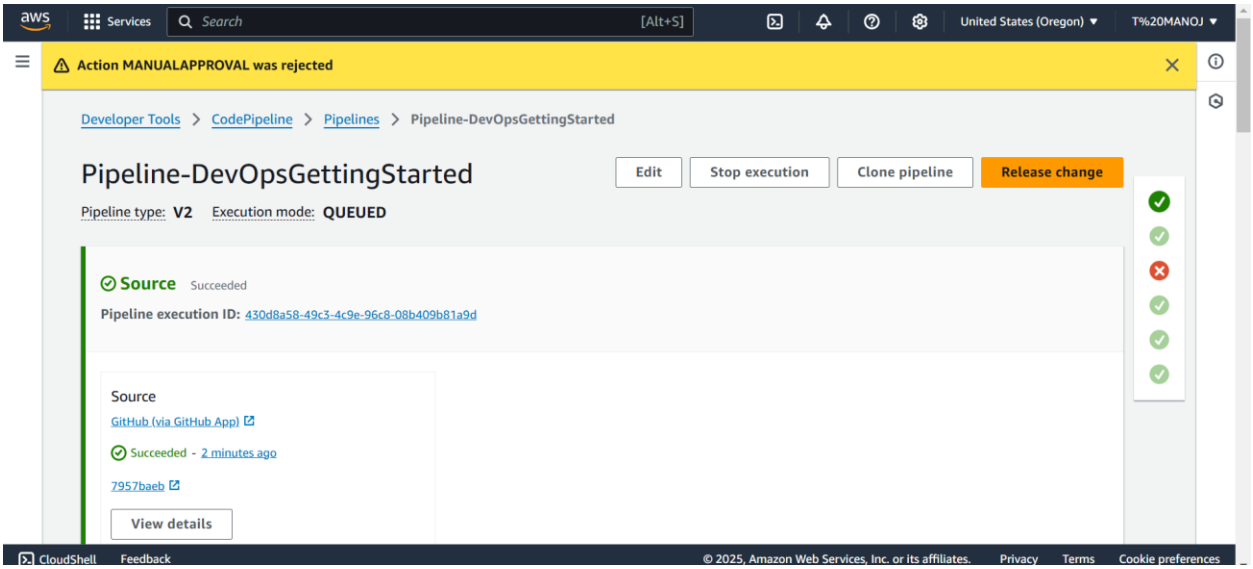


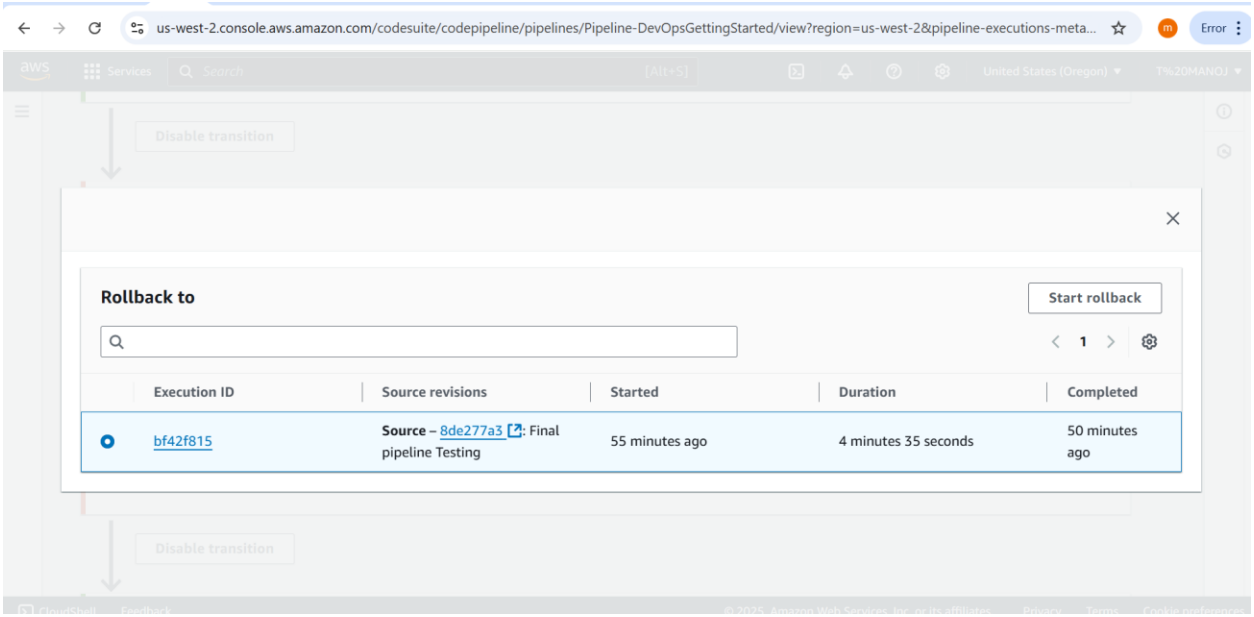
Hello World! hi this is version 2 final testing

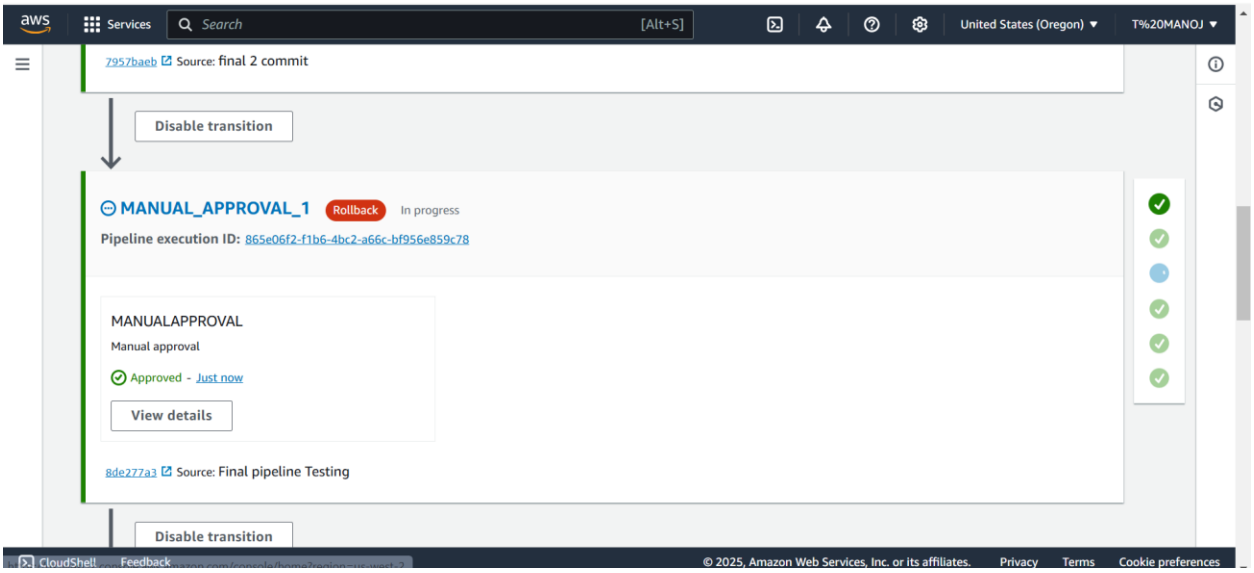

us-west-2.console.aws.amazon.com/codesuite/codepipeline/pipelines/Pipeline-DevOpsGettingStarted/view?region=us-west-2
Review
Action name: MANUALAPPROVAL Status: Waiting for approval
Details
Revisions
Trigger
Commit 7957baeb
pushed in manojmanu9441/aws-elastic-beanstalk-express-js-sample/main
final 2 commit
Comments about this action
-
URL for review
Decision
Approve
Approving will resume the pipeline execution.
Reject
Rejecting will stop the pipeline execution with a failed status.
Cancel
Submit

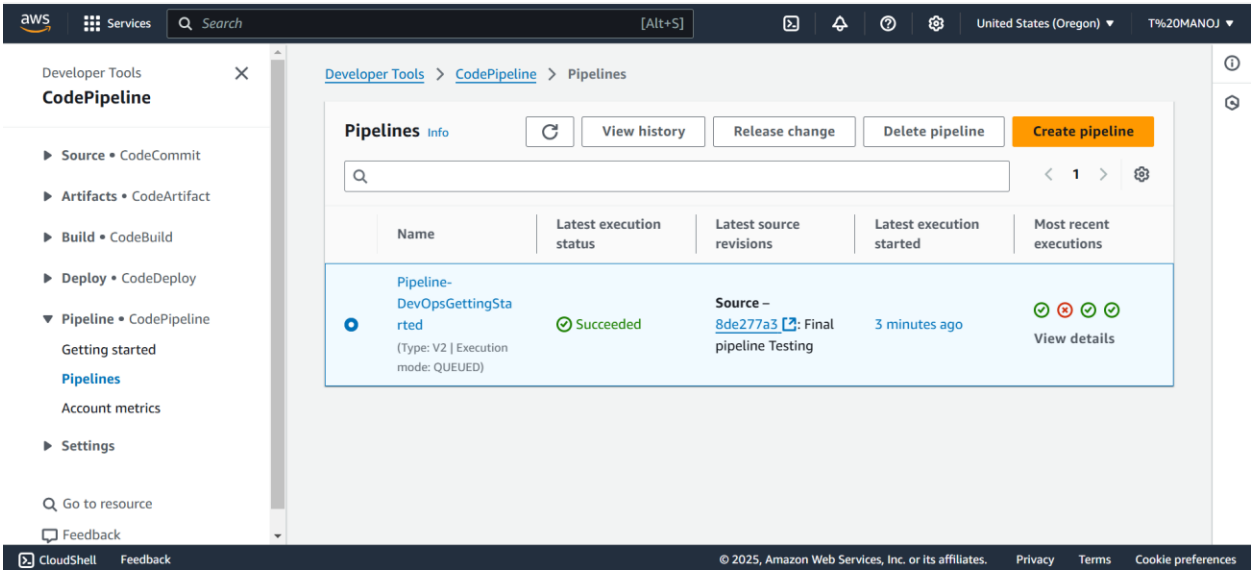

us-west-2.console.aws.amazon.com/codesuite/codepipeline/pipelines/Pipeline-DevOpsGettingStarted/view?region=us-west-2
Details
Revisions
Trigger
Commit 7957baeb
pushed in manojmanu9441/aws-elastic-beanstalk-express-js-sample/main
final 2 commit
Comments about this action
-
URL for review
Decision
Approve
Approving will resume the pipeline execution.
Reject
Rejecting will stop the pipeline execution with a failed status.
Comments - optional
Preview markdown
Learn more
NOT APPROVED
Cancel
Submit


aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
MANUAL_APPROVAL_1
In progress
Retry failed actions
Pipeline execution ID: 430d8a58-49c3-4c9e-96c8-08b409b81a9d
MANUALAPPROVAL
Manual approval
Rejected - Just now
View details
7957baeb
Source: final 2 commit
Disable transition
Test
Succeeded
Start rollback
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
Action MANUALAPPROVAL was rejected
Developer Tools
CodePipeline
Pipelines
Pipeline-DevOpsGettingStarted
Pipeline-DevOpsGettingStarted
Edit
Stop execution
Clone pipeline
Release change
Pipeline type: V2
Execution mode: QUEUED
Source
Succeeded
Pipeline execution ID: 430d8a58-49c3-4c9e-96c8-08b409b81a9d
Source
GitHub (via GitHub App)
Succeeded - 2 minutes ago
7957baeb
View details
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


us-west-2.console.aws.amazon.com/codesuite/codepipeline/pipelines/Pipeline-DevOpsGettingStarted/view?region=us-west-2&pipeline-executions-meta...
Error
Disable transition
Rollback to
Start rollback
1
Execution ID
Source revisions
Started
Duration
Completed
bf42f815
Source – 8de277a3: Final pipeline Testing
55 minutes ago
4 minutes 35 seconds
50 minutes ago
Disable transition


aws
Services
Search
[Alt+S]
United States (Oregon)
T%20MANOJ
7957baeb Source: final 2 commit
Disable transition
MANUAL_APPROVAL_1
Rollback
In progress
Pipeline execution ID: 865e06f2-f1b6-4bc2-a66c-bf956e859c78
MANUALAPPROVAL
Manual approval
Approved - Just now
View details
8de277a3 Source: Final pipeline Testing
Disable transition
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences




## Final CI/CD Architecture Overview

```
GitHub (Source Repository)
         ↓
 AWS CodePipeline (Orchestrates entire flow)
   ├── Source Stage (GitHub webhook trigger)
   ├── Build Stage (AWS CodeBuild compiles & tests code)
   ├── Approval Stage (Manual approval with SNS & SQS)
   └── Deploy Stage (AWS Elastic Beanstalk deployment)
```

### Advantages

- **Automation of the Entire Workflow**: Reduces manual intervention by automating build, test, and deployment processes.
- **Faster Delivery**: Accelerates the release cycle by enabling rapid deployment of new features and bug fixes.

- **Improved Code Quality**: Automated builds and tests catch errors early, reducing production bugs.
- **Scalability and Flexibility**: AWS Elastic Beanstalk automatically handles capacity provisioning, load balancing, and scaling.
- **Cost-Effective**: Leveraging AWS free-tier services (like single-instance environments) makes it budget-friendly for small projects.
- **Enhanced Security and Control**: Manual approval stages with SNS and SQS ensure critical changes undergo human review before deployment.
- **Seamless Integration**: Native integration between GitHub and AWS services ensures smooth operation without complex configurations.

## Results and Discussion

The project successfully established a fully automated CI/CD pipeline using GitHub, AWS Elastic Beanstalk, AWS CodeBuild, AWS CodePipeline, SNS, and SQS. The GitHub repository was correctly forked, cloned, and modified (`app.js`), with changes committed and pushed to the remote repository. The web application was deployed on AWS Elastic Beanstalk, running smoothly on an EC2 instance with the generated URL accessible. AWS CodeBuild compiled the code with the `buildspec.yml` file, completing the build process without errors. The AWS CodePipeline orchestrated the entire CI/CD flow, automatically triggering builds and deployments on new GitHub commits. The manual approval stage, integrated using SNS and SQS, worked as expected—notifications were sent via email, and approval actions managed through the console or email link. The rollback mechanism was validated successfully when a deployment was rejected during manual approval.

The discussion highlights the pipeline's efficiency in automating the build, test, and deployment process, significantly reducing manual intervention and deployment time. Elastic Beanstalk ensured a scalable and reliable infrastructure for seamless application updates. The robust rollback mechanism protected the production environment from unstable deployments. SNS and SQS integrations provided real-time notifications and

reliable message queuing, enhancing the manual approval workflow. The secure communication between AWS services, achieved through OAuth and appropriate IAM roles, ensured a safe deployment process. Overall, the continuous integration and deployment approach streamlined development workflows, improving productivity and ensuring faster, more reliable software releases

## Conclusion

By implementing this CI/CD pipeline, you have established an automated, scalable, and reliable deployment process. The pipeline detects changes in the GitHub repository, builds the application using AWS CodeBuild, and deploys it to AWS Elastic Beanstalk through AWS CodePipeline.The addition of a manual approval step with SNS and SQS ensures that critical updates are reviewed, providing a balance between automation and control.This solution not only speeds up development cycles but also ensures that deployments are secure, high quality, and highly available, embodying the best practices of DevOps .