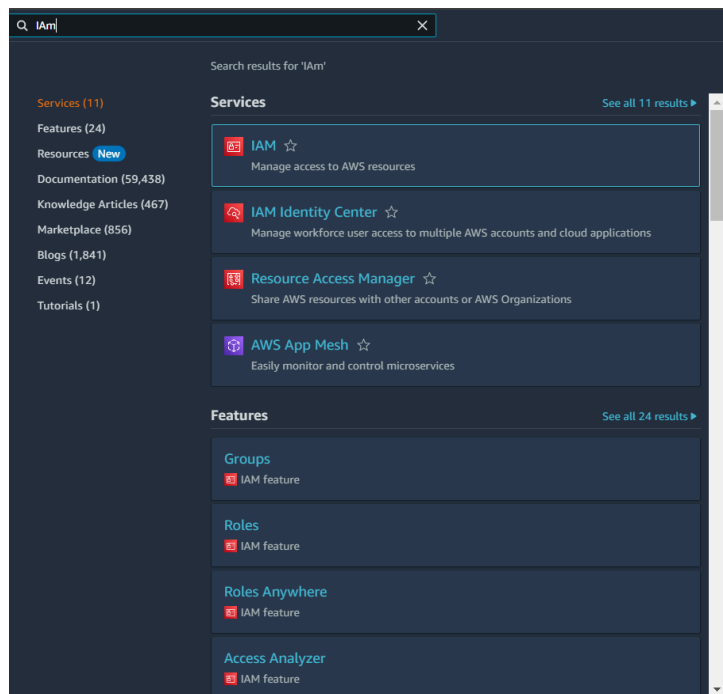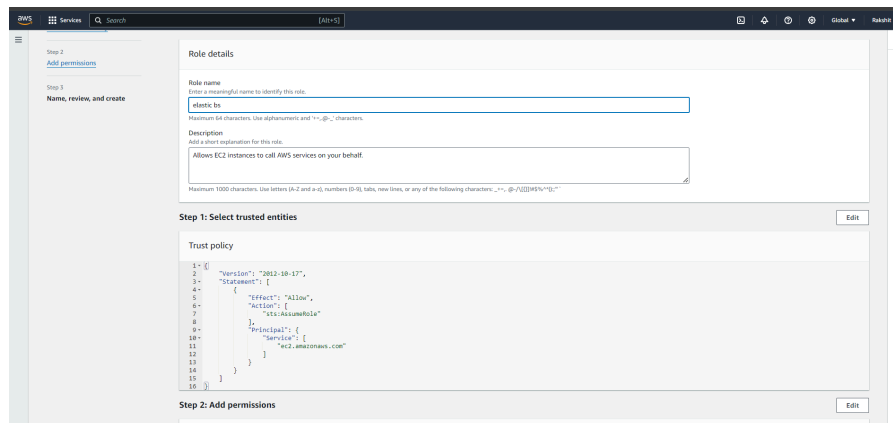## Experiment No: 2
**Aim: To Build Your Application using AWS CodeBuild and Deploy on S3 / SEBS using**
**AWS CodePipeline, deploy Sample Application on EC2 instance using AWS CodeDeploy.**

Step 1: Search for IAM ,  Go to Roles section and click on create a role.



Step 2: Fill the role details. Select AWS service and enable web tier and work tier permissions.

Step 3: Search elastic beanstalk . After loading the page click on Create application.
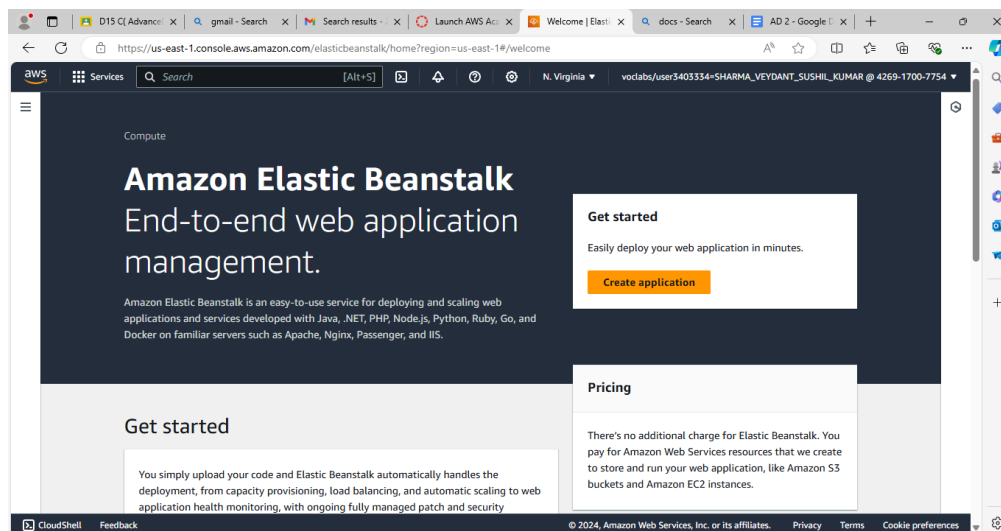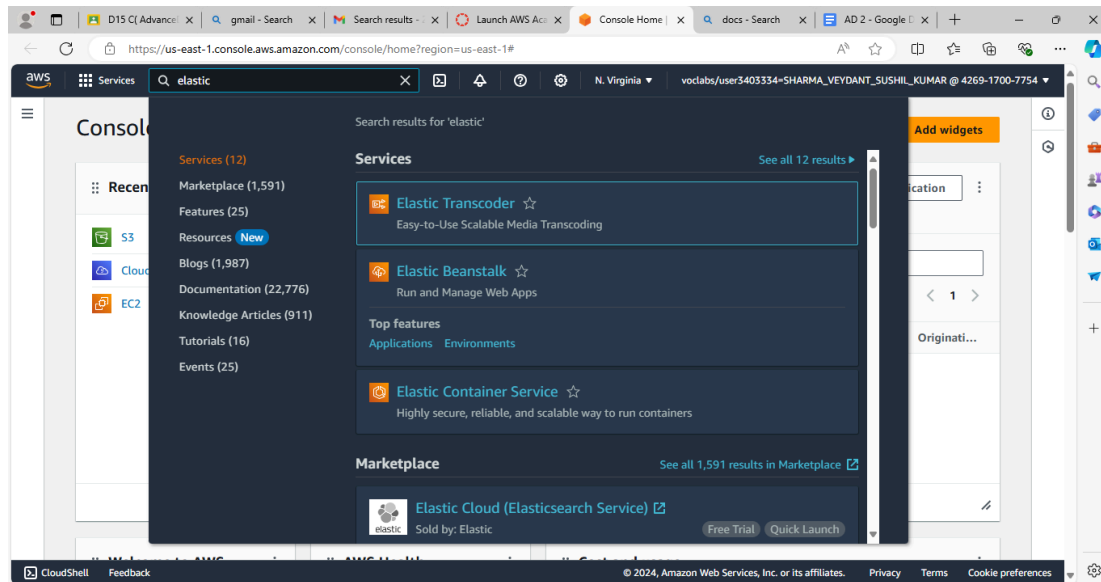
Step 4: Create application , provide application name and environment



Select the platform.



Click on 'Use existing service role' in the service menu . Assign the service role as the one create in IAM, also assign the key pair.
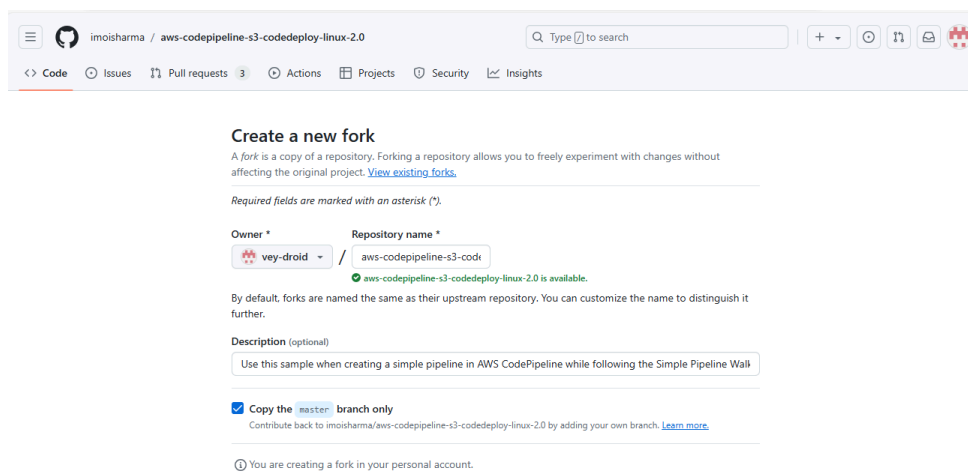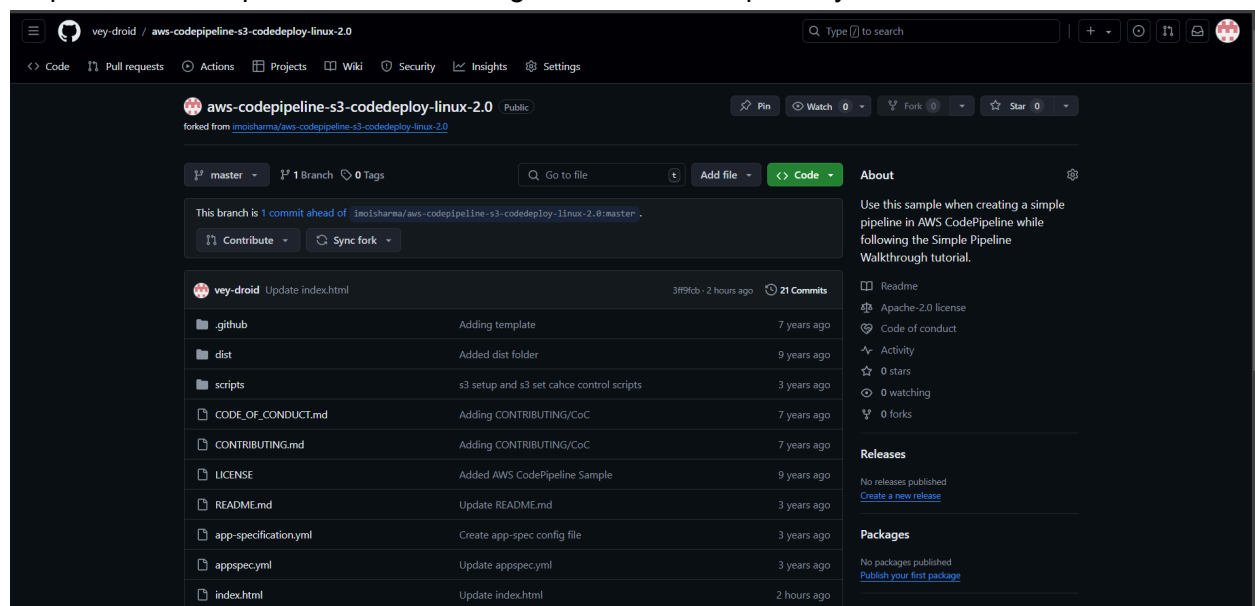
Keep the rest of the settings as default. Wait for the environment to be created/launched successfully.



Step 5: Get a sample of the code from github , fork the repository.

Step 6: Go to codepipeline, Create new pipeline.
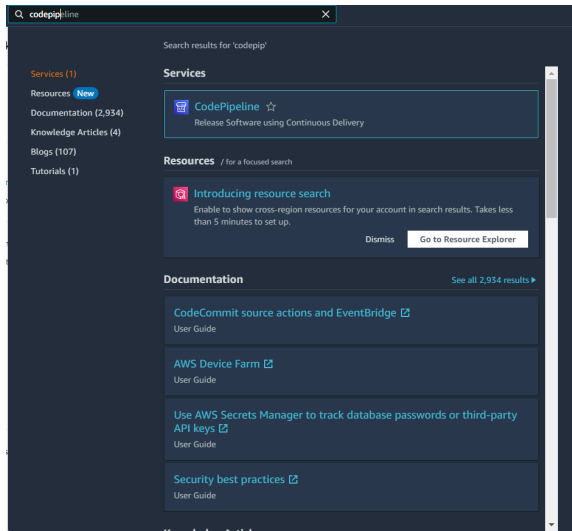


Step 7: Name the pipeline. Head on to the 'Source' Page.



On the source page, select source provider as Github V2 ,  connect github account, and select the repository and the branch.

**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 2)

**New GitHub version 2 (app-based) action**
To add a GitHub version 2 action in CodePipeline, you create a connection, which uses GitHub Apps to access your repository. Use the options below to choose an existing connection or create a new one. Learn more

Connection
Choose an existing connection that you have already configured, or create a new one and then return to this task.

arn:aws:codeconnections:us-east-1:975050293750:connection/eac0d1b2-a1:  ✕    or    **Connect to GitHub**

**Ready to connect**
Your GitHub connection is ready for use.

Repository name
Choose a repository in your GitHub account.

🔍 vey-droid/aws-codepipeline-s3-codedeploy-linux-2.0    ✕

You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

Default branch
Default branch will be used only when pipeline execution starts from a different source or manually started.

🔍 master    ✕

Output artifact format
Choose the output artifact format.

| ● CodePipeline default | ○ Full clone |
|---|---|
| AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include Git metadata about the repository. | AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full Git clone. Only supported for AWS CodeBuild actions. |

Select defaults and skip the build page.

On the deploy page , select provider as AWS Elastic Beanstalk. Select the Application and environment name, then proceed further.

**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

US East (N. Virginia)

Input artifacts
Choose an input artifact for this action. Learn more ⬈

No more than 100 characters

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

🔍 veydant-server    ✕

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.
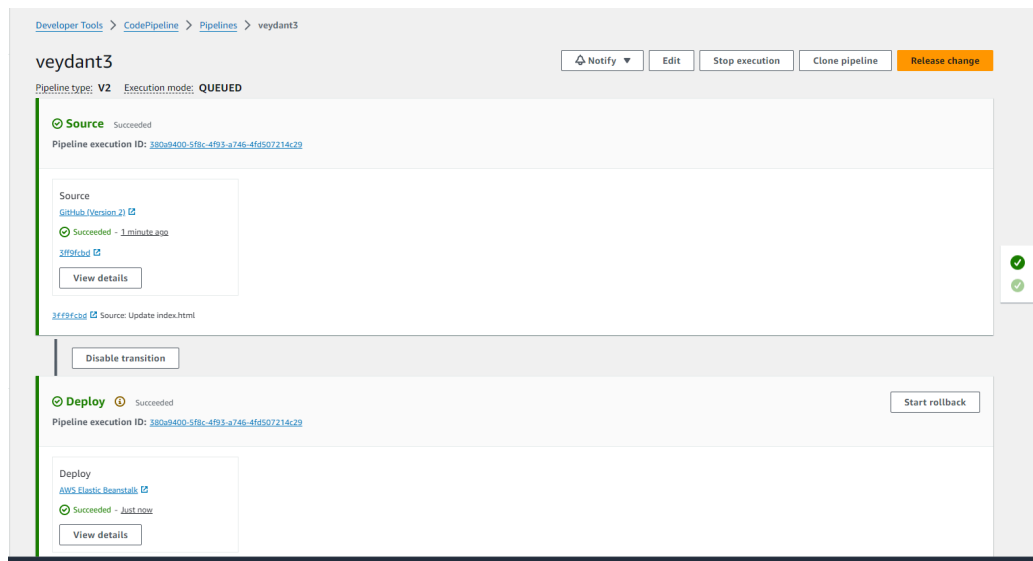
🔍 Veydant-server-env    ✕

☐ Configure automatic rollback on stage failure

Step 8: The pipeline will be launched in a few moments, once it is successfully created head over to elastic beanstalk environment.



Step 9: A URL will be created on the environment page, that is the hosted link, click on it.