**Lesson 02 Demo 01**

**Creating a Pipeline from the GitHub Repository**

**Objective:** To create an AWS CodePipeline for continuous integration and delivery (CI/CD) using a GitHub repository

**Tools required:** GitHub and CodePipeline

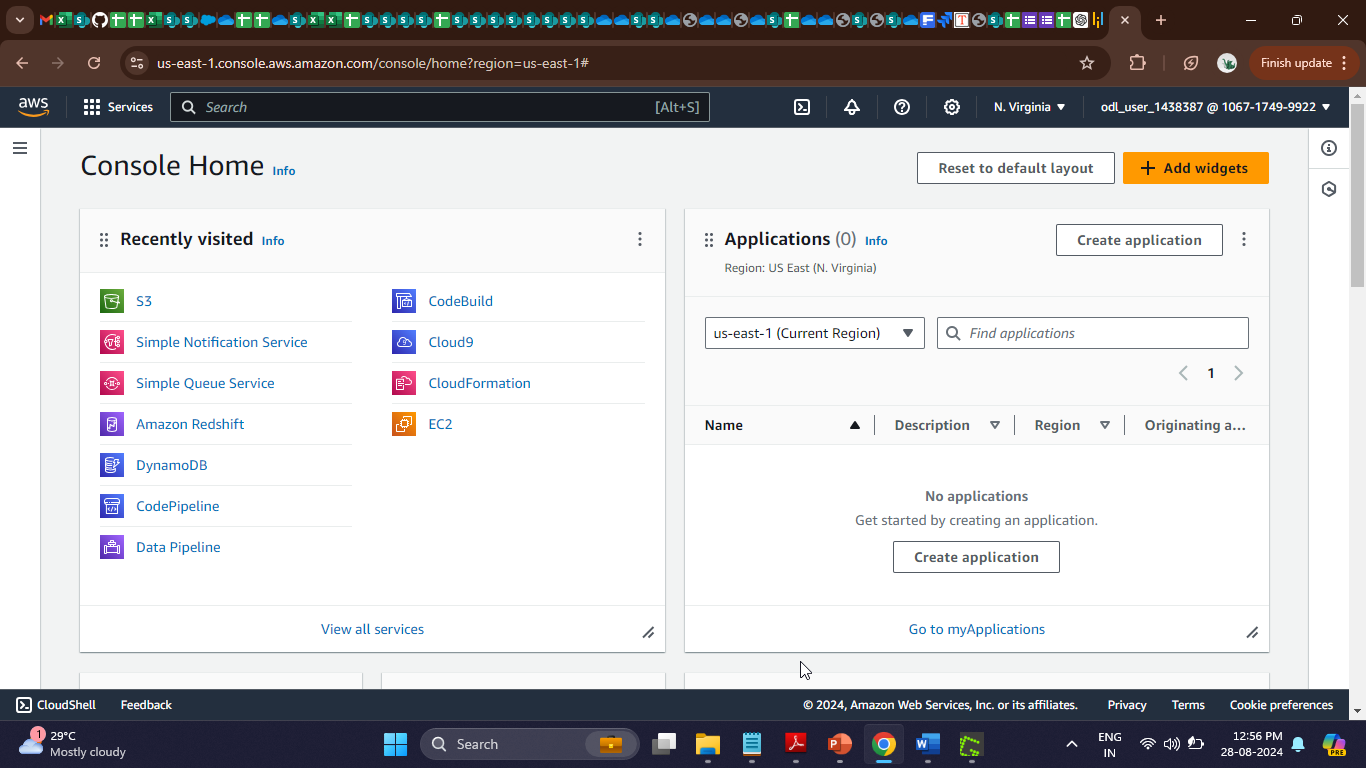
**Prerequisites:** A GitHub Git repository with the master branch should be set up and integrated with AWS CodePipeline for continuous integration and delivery (CI/CD)

Steps to be followed:

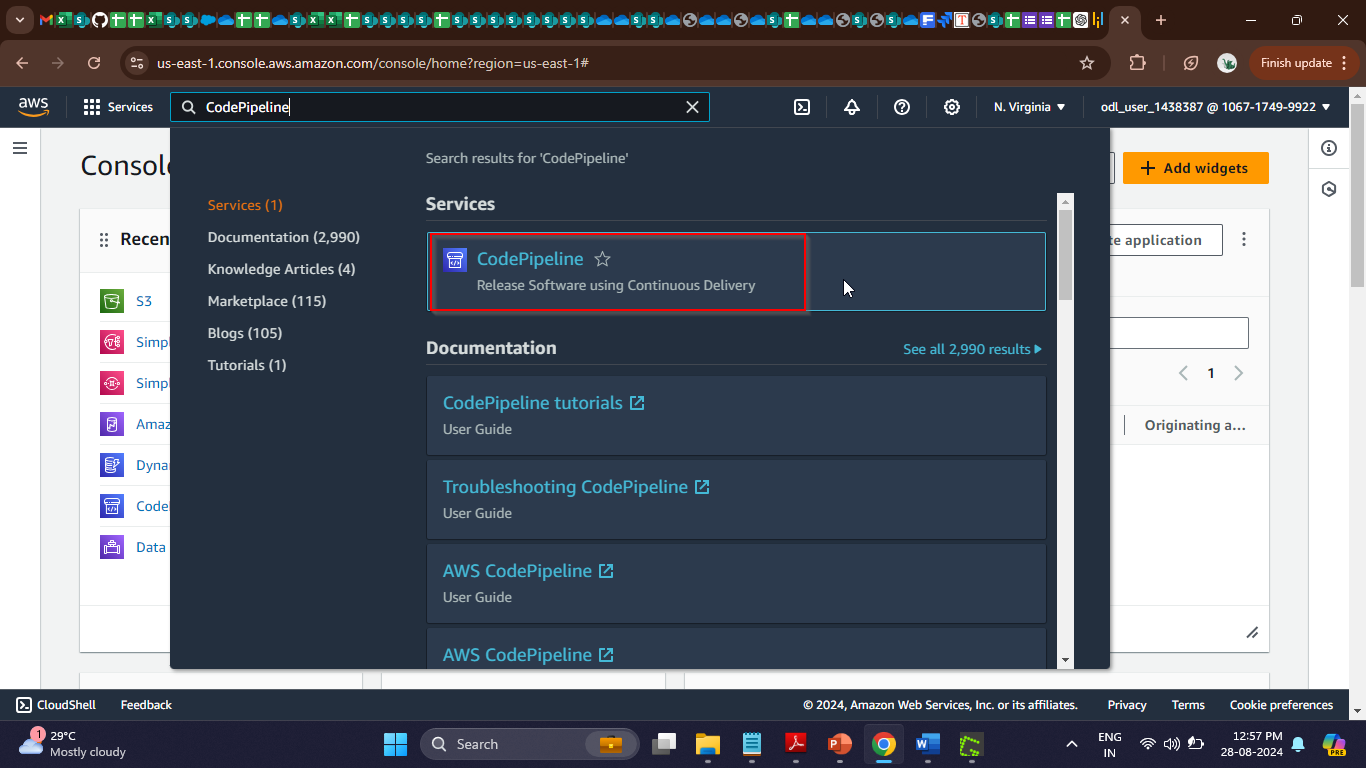
1. Create a new pipeline using the GitHub repository

**Step 1: Create a new pipeline using the GitHub repository**

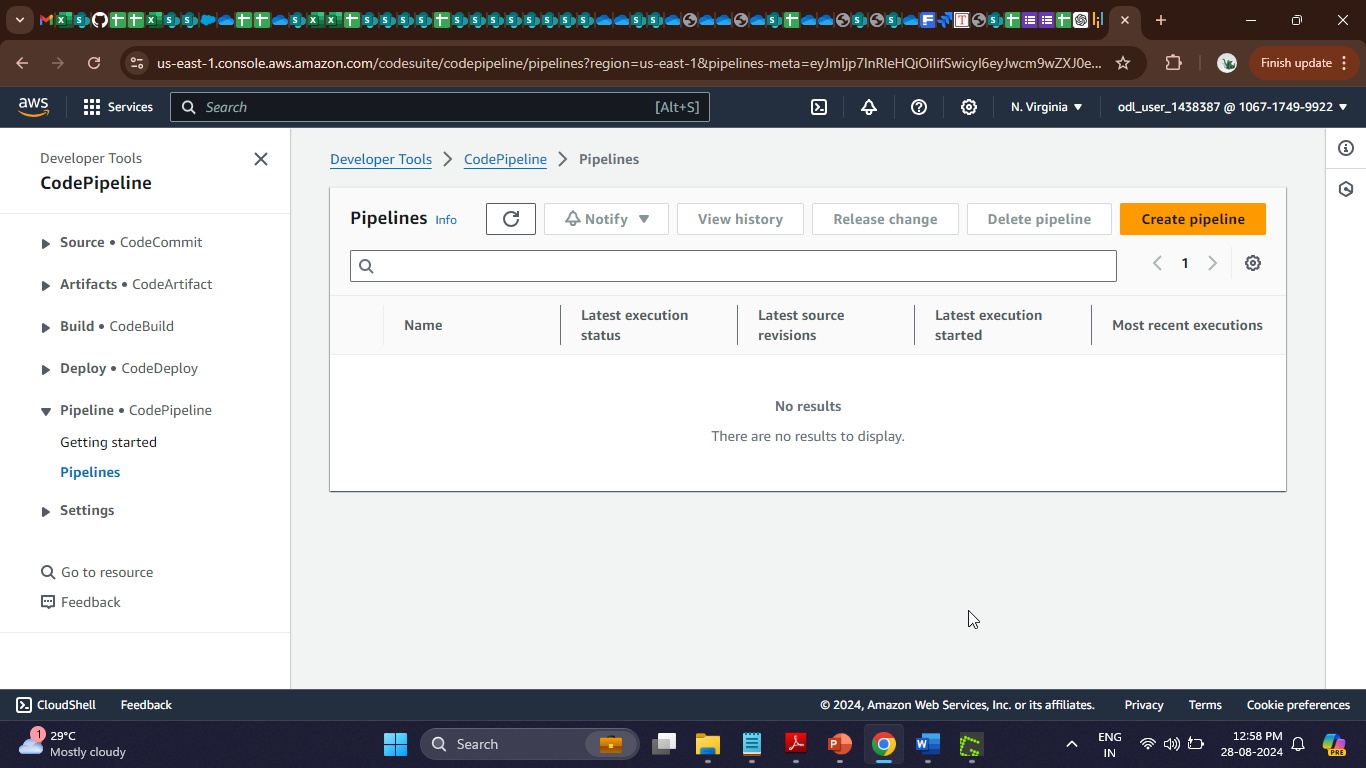
1. Sign in to the AWS Console



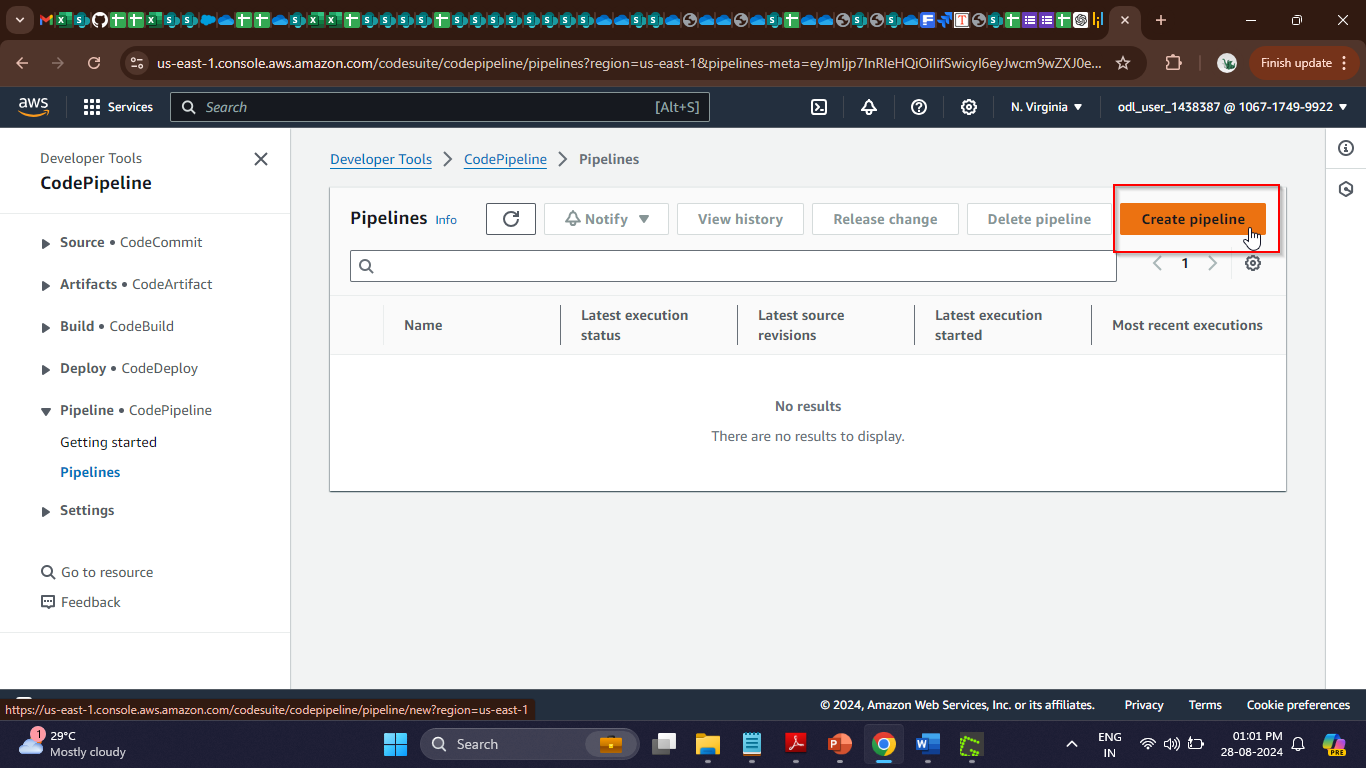
1. Search for and click on **CodePipeline**



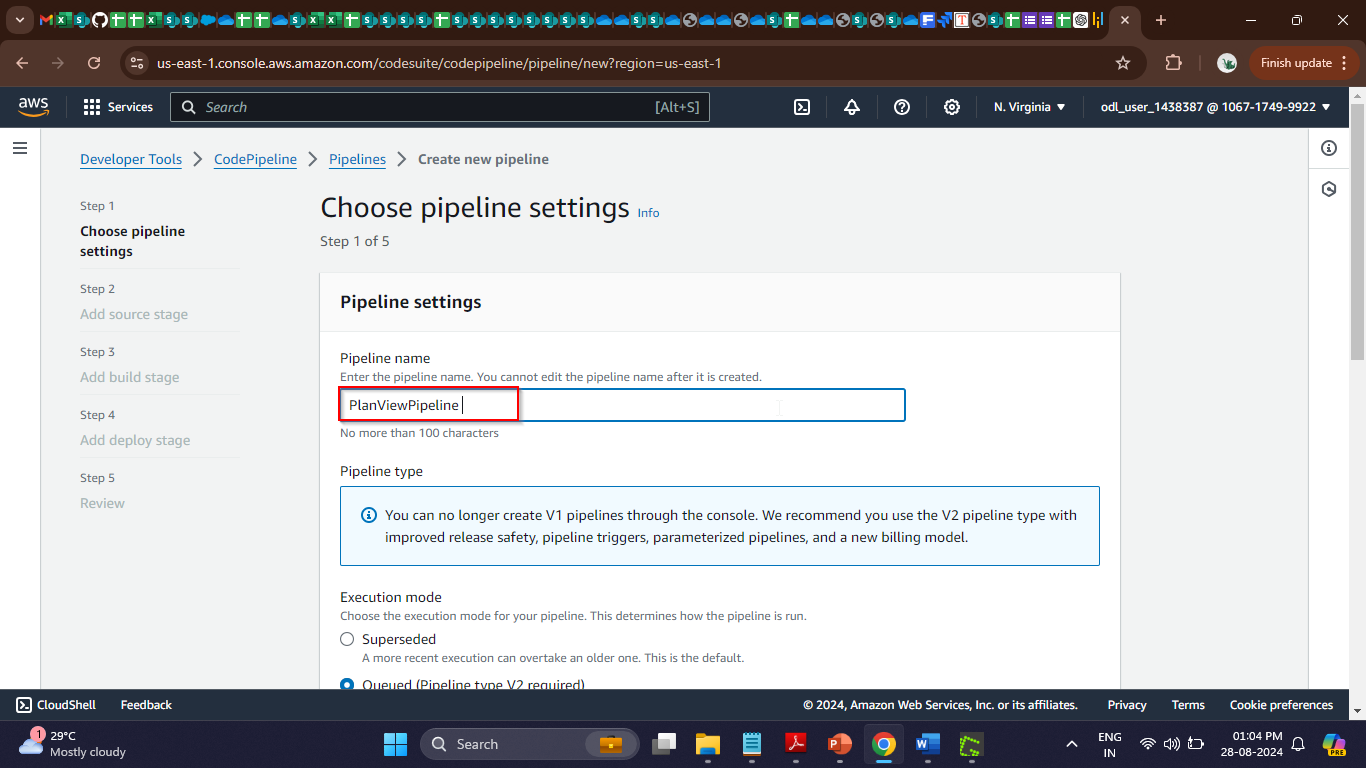
After clicking on the **CodePipeline** service, you will see the following interface:

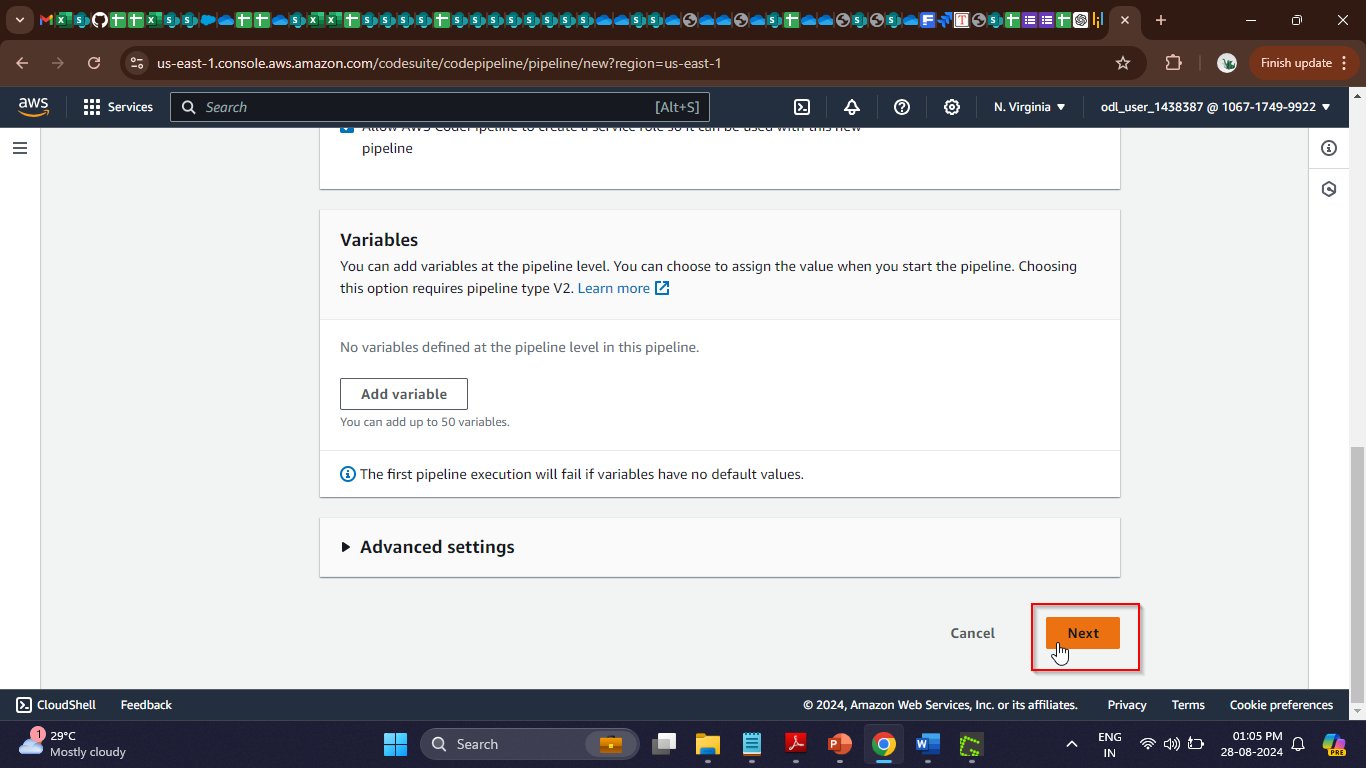


1. Click on the **Create pipeline** button

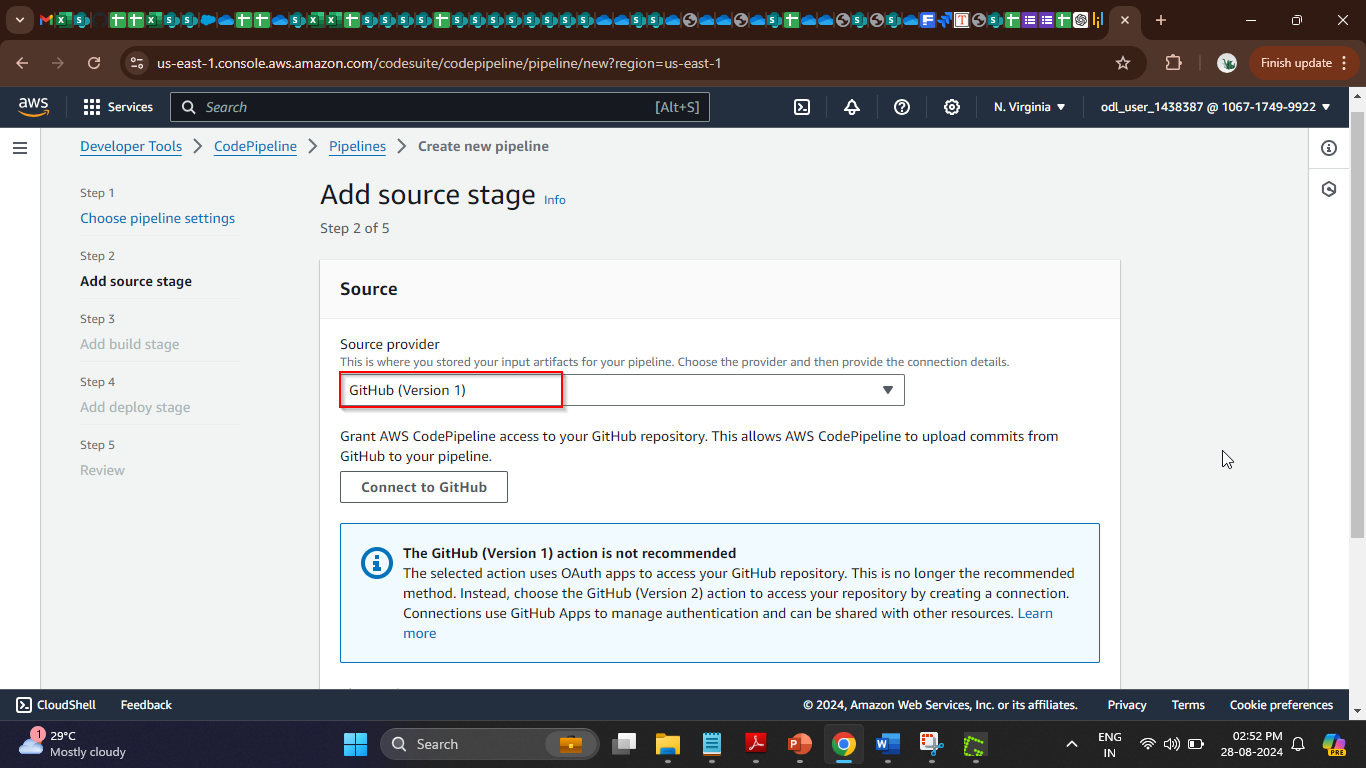
****

1. Enter a name for the pipeline and click the **Next** button

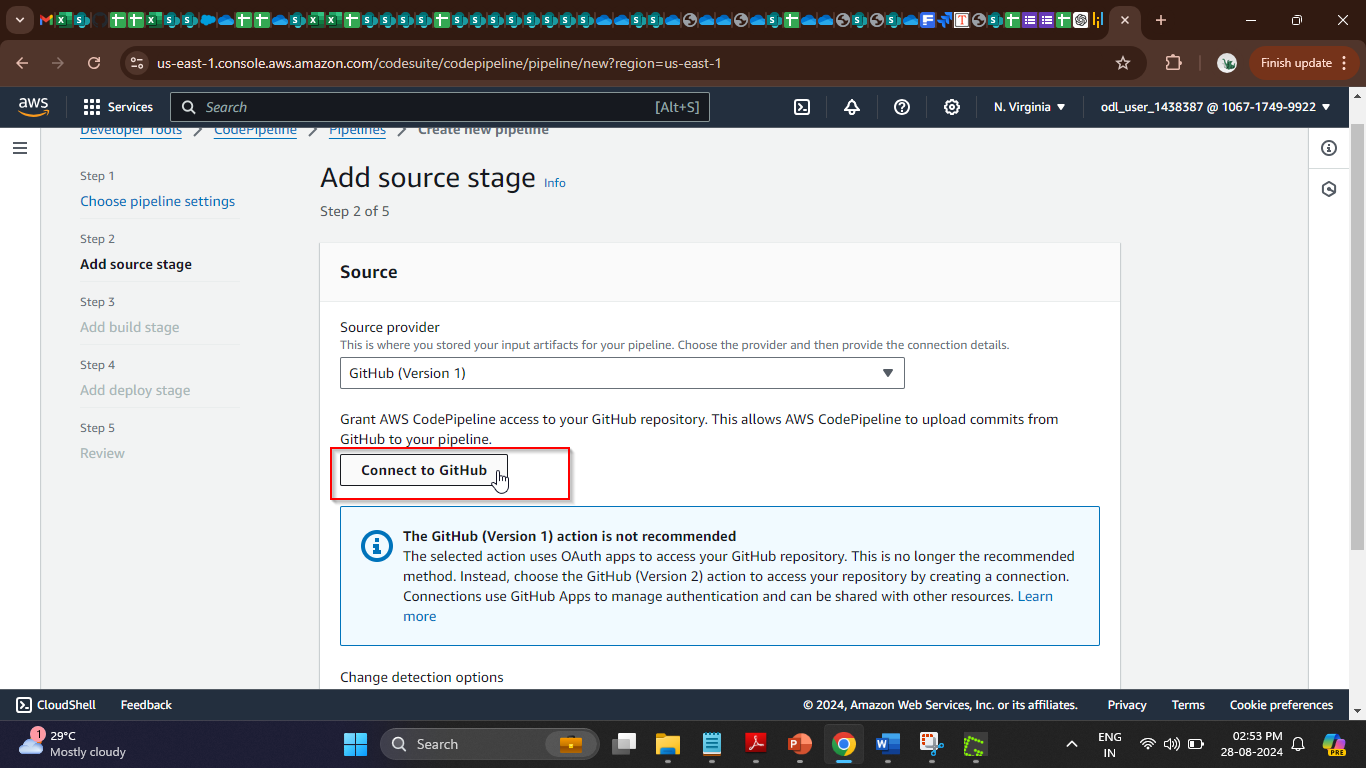




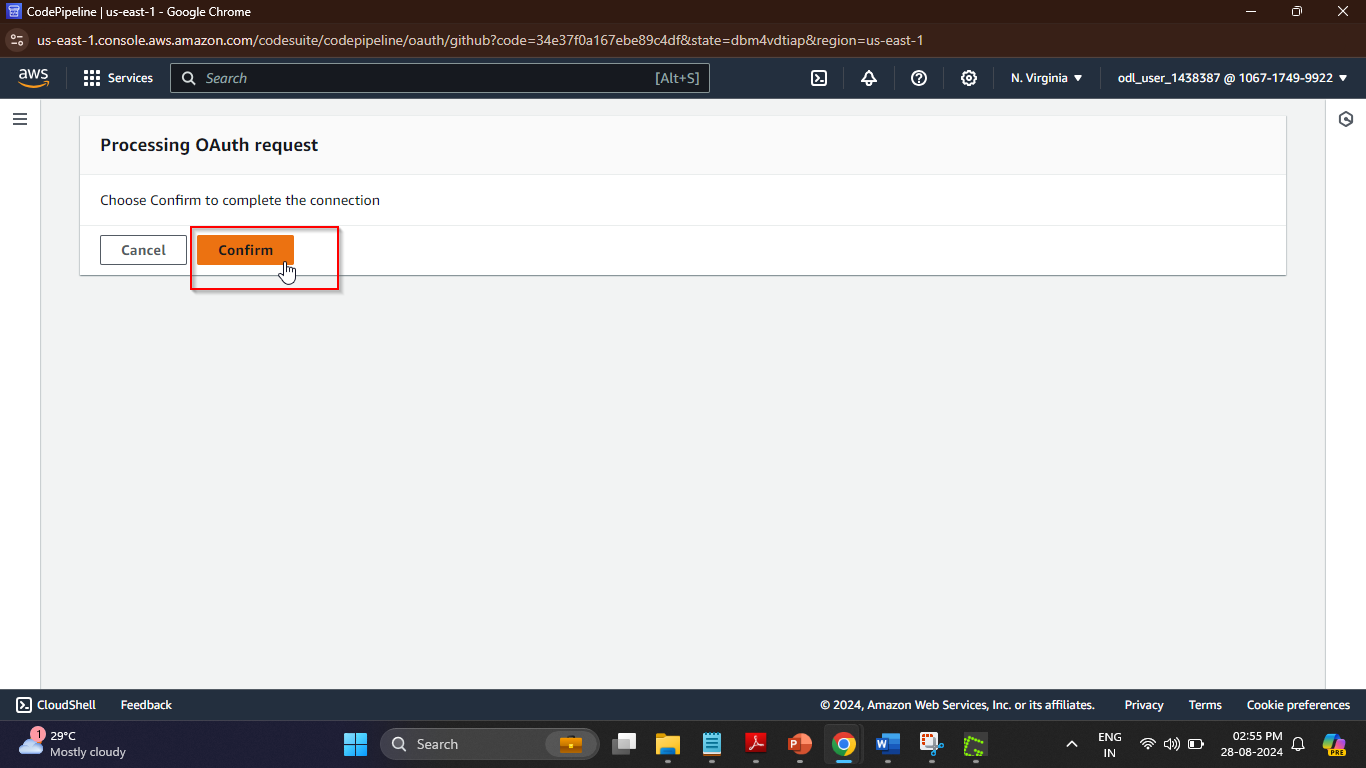
1. Select the desired **Source provider** as **GitHub (Version 1)**

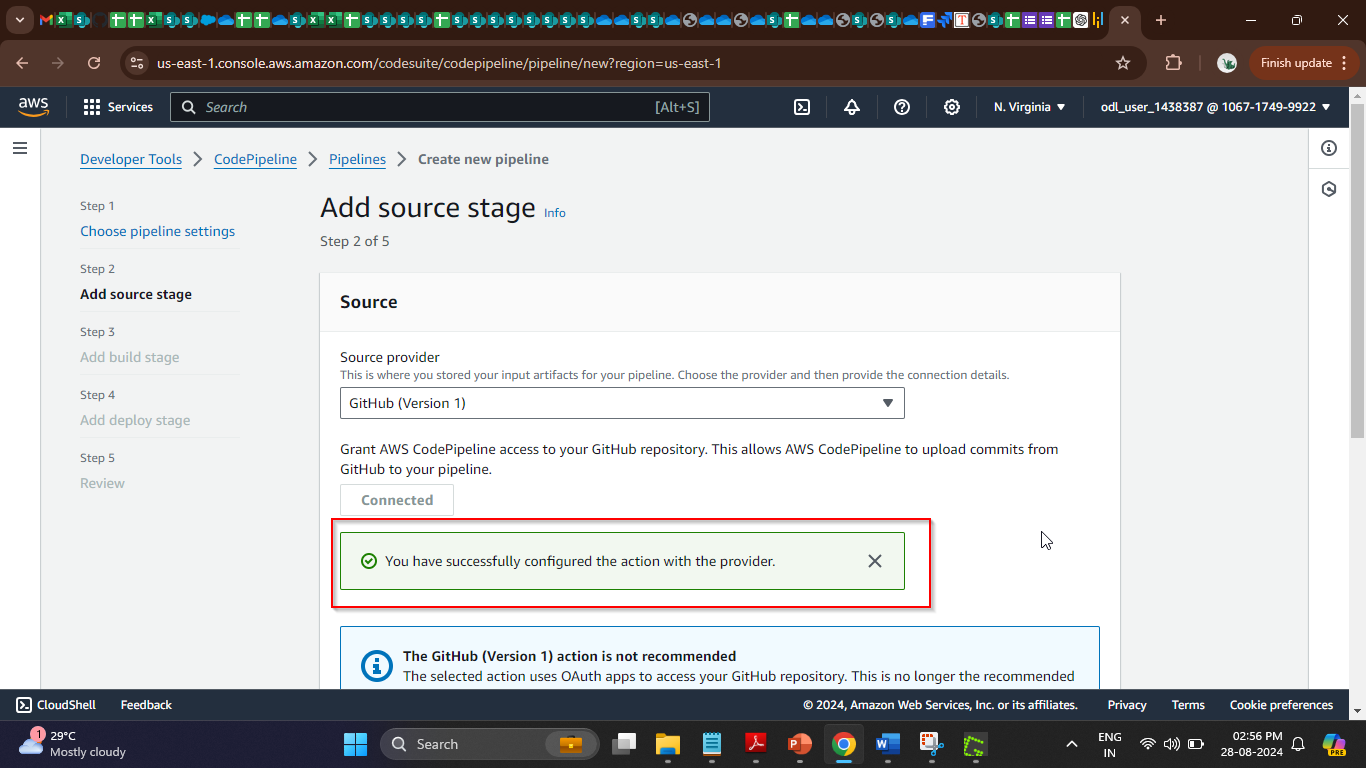
****

1. Click on **Connect to GitHub**

****

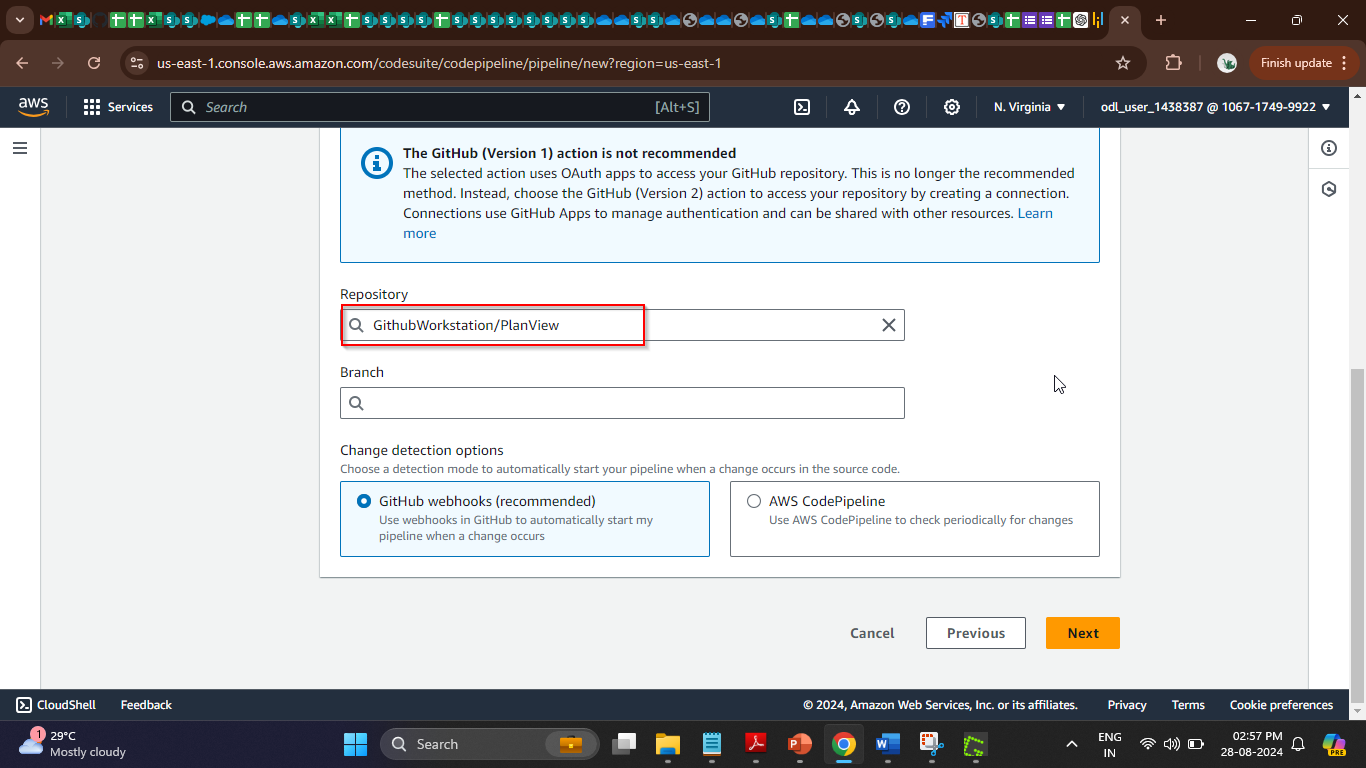
1. Click on **Confirm**

****

****

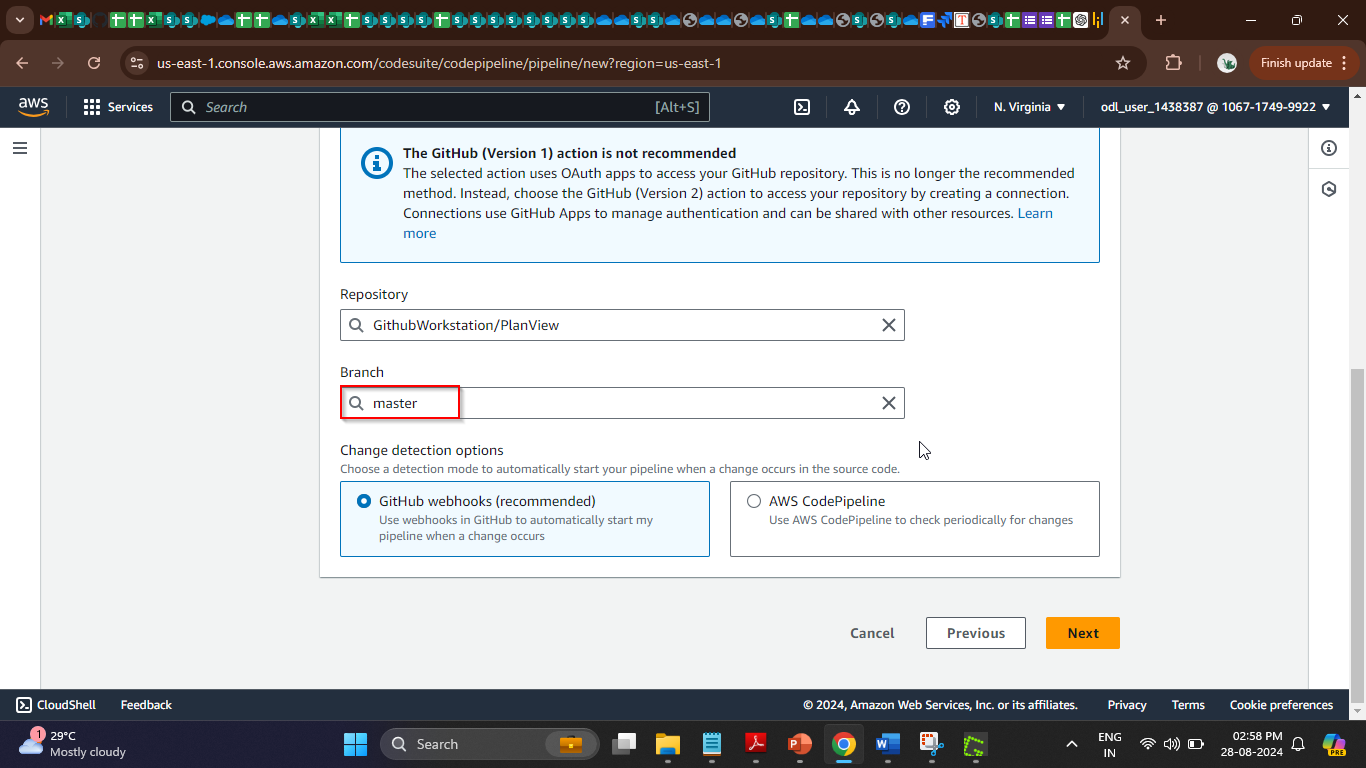
The GitHub connection is now ready for use.

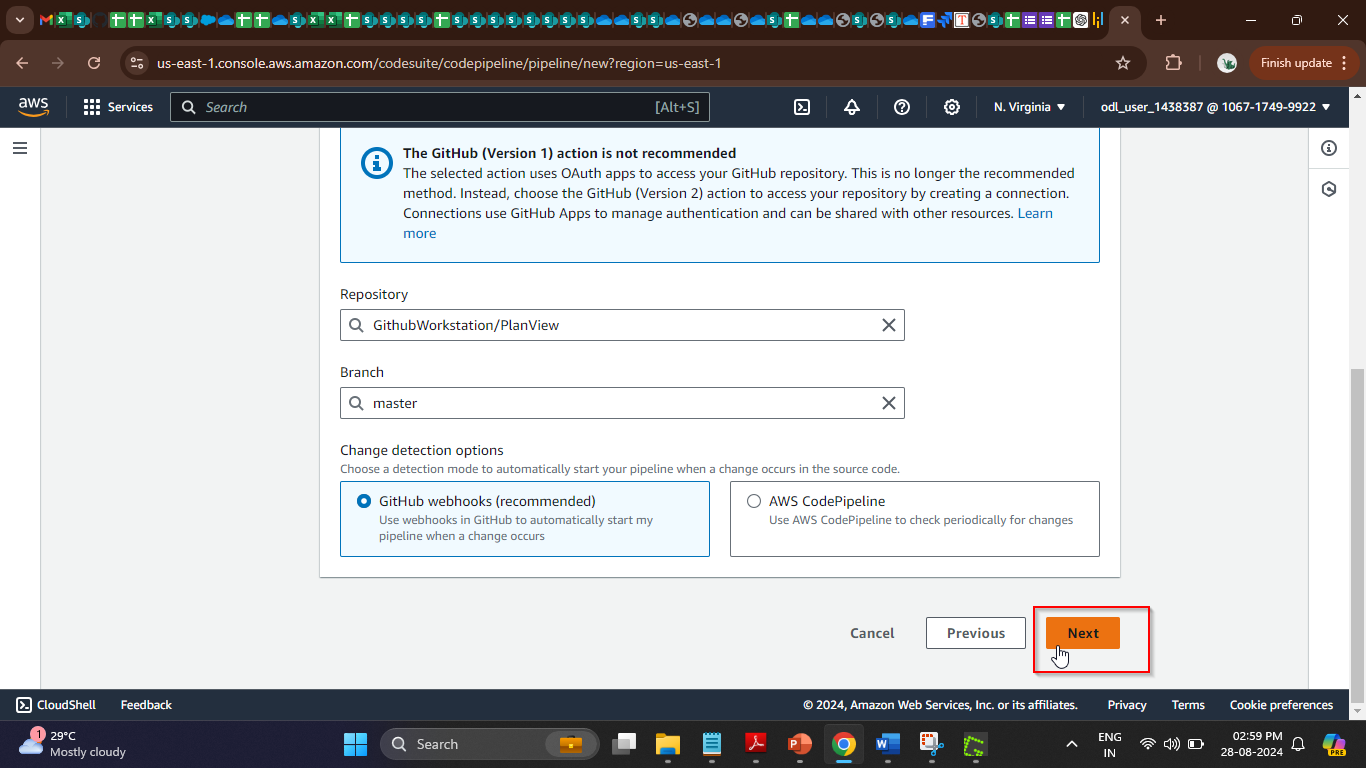
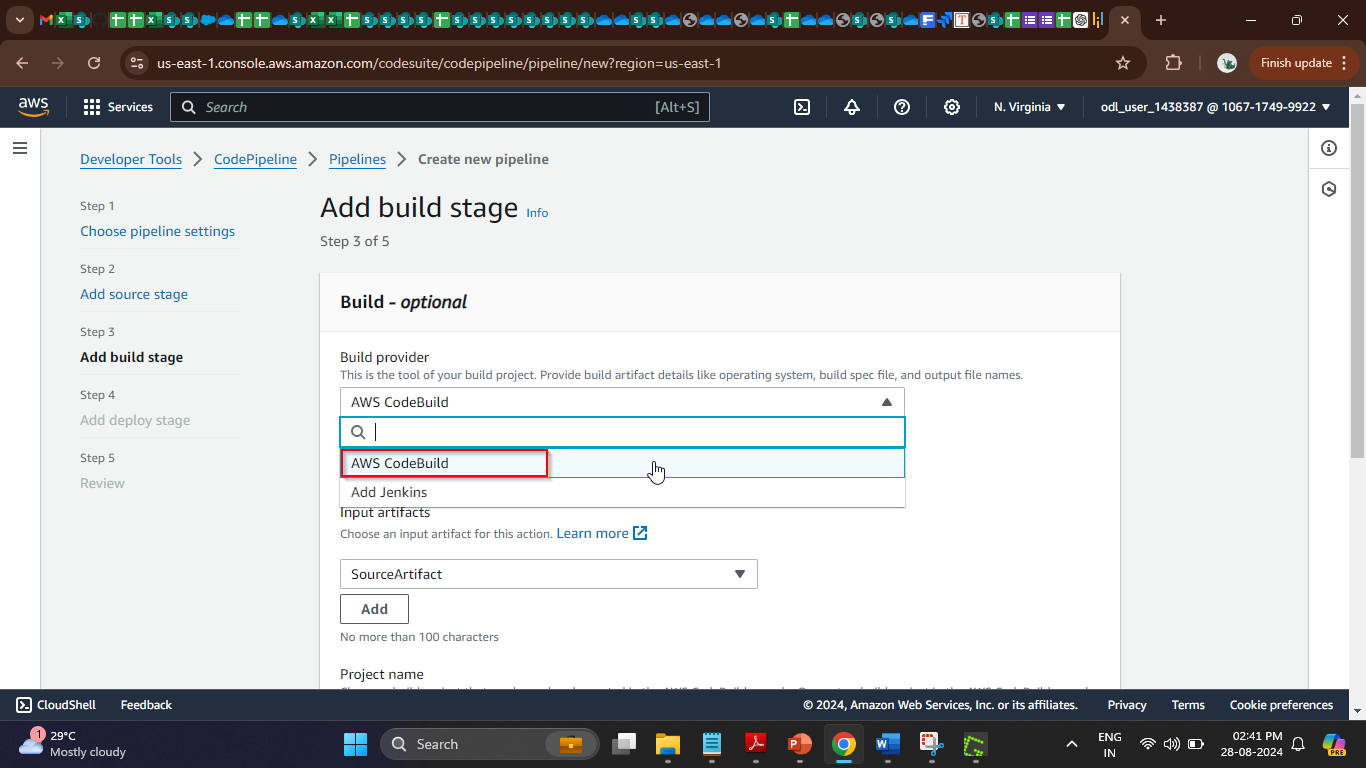
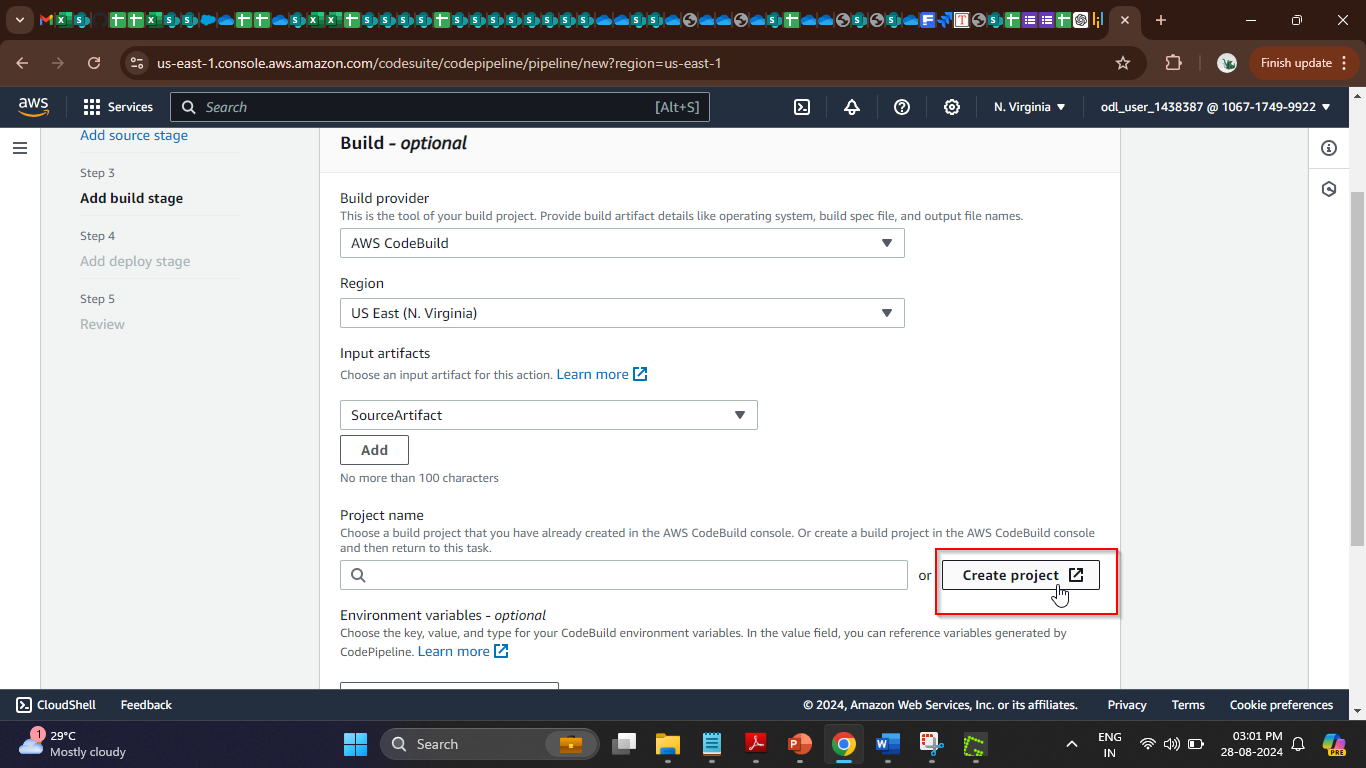
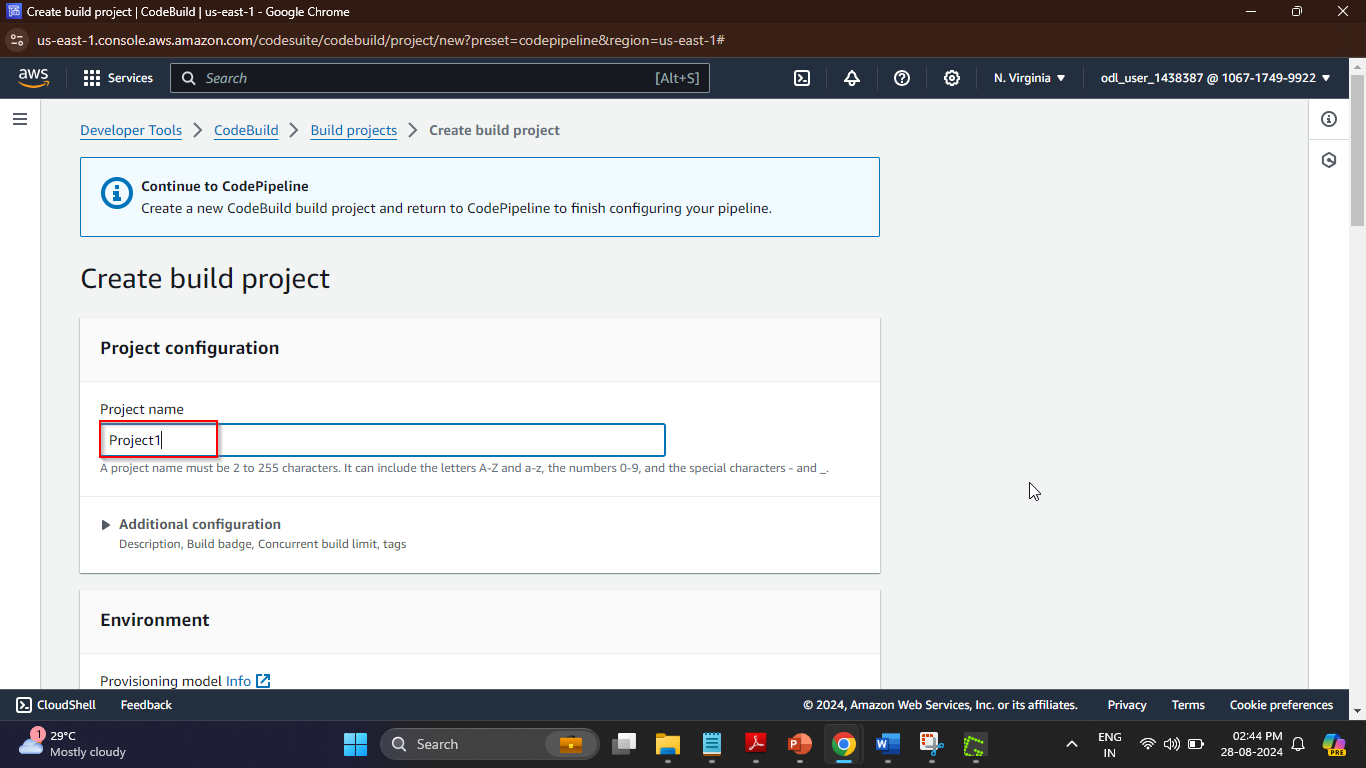
1. Select the **GithubWorkstation/PlanView** option under **Repository**

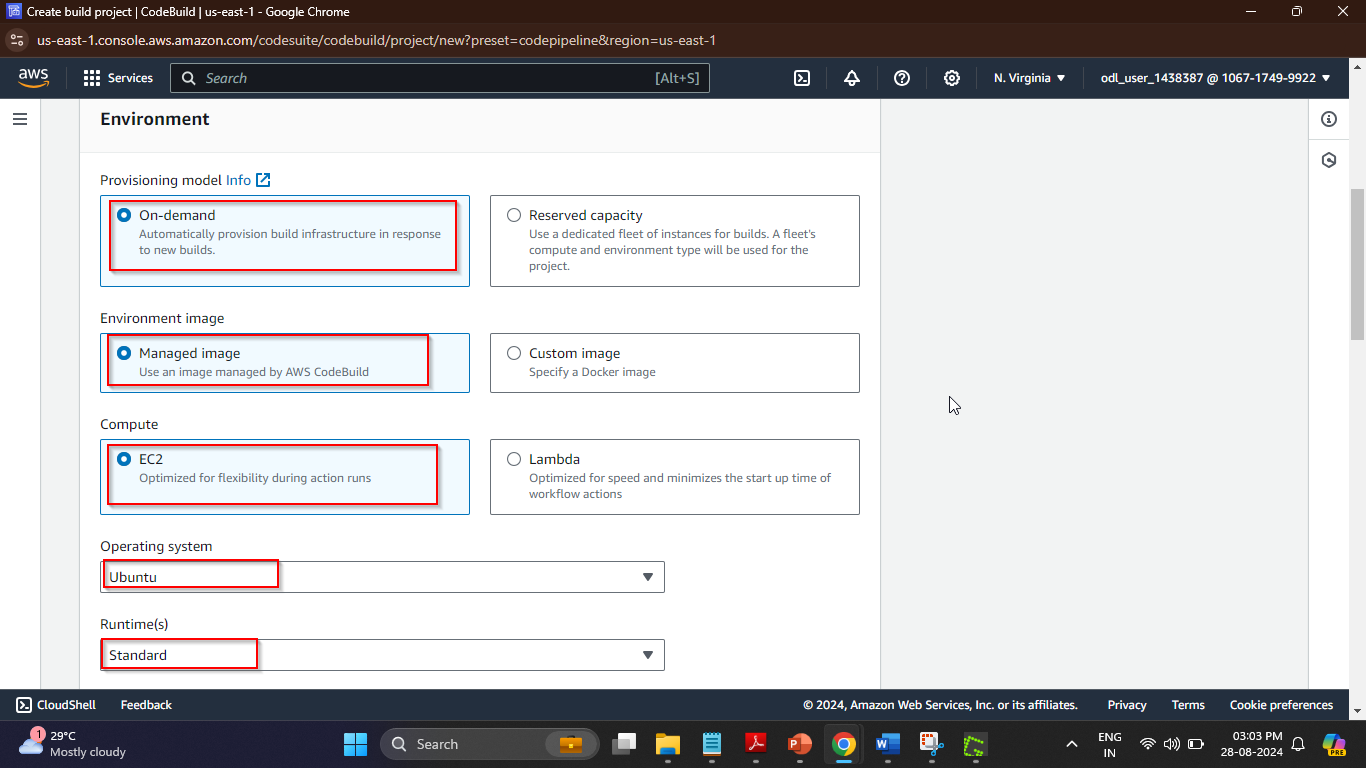
****

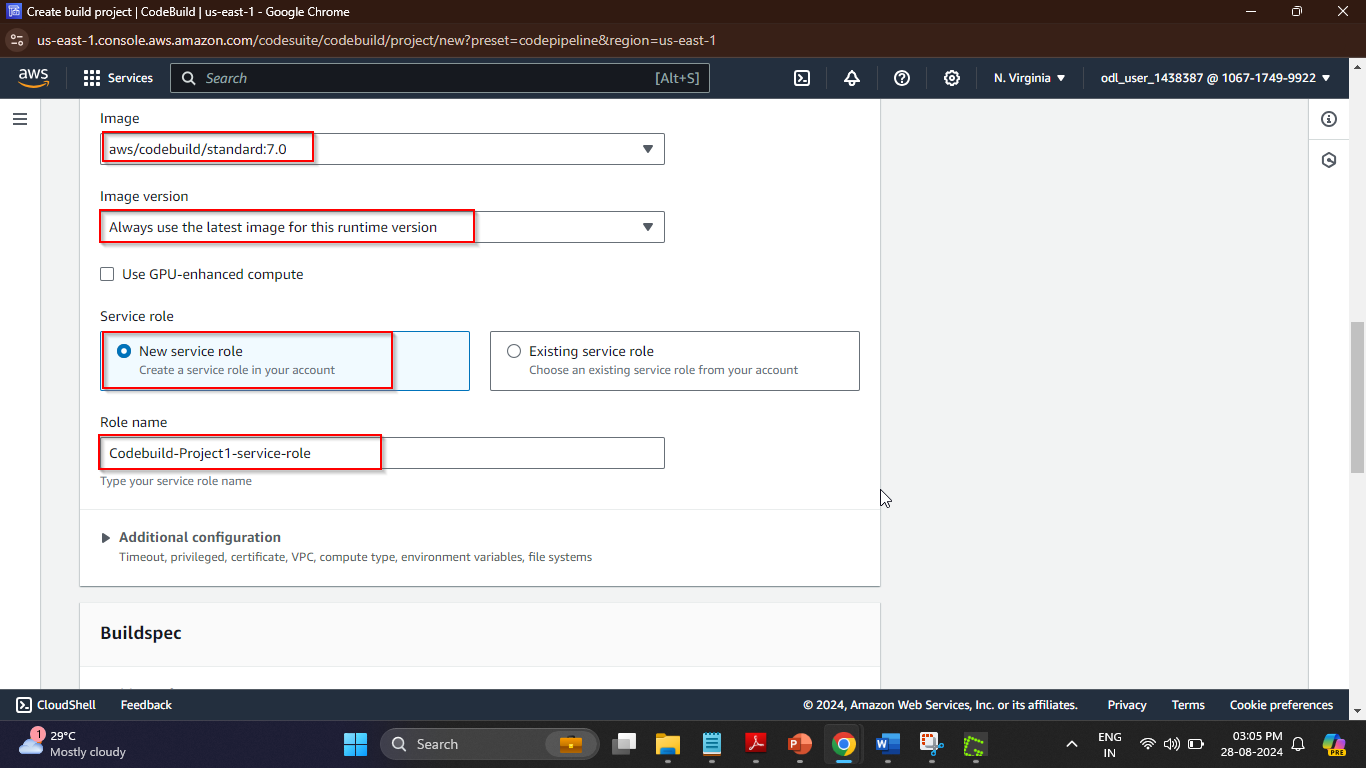
|  |
| --- |
| **Note:** Name the previously created repository as **PlanView** or specify any desired name of the repository you created |

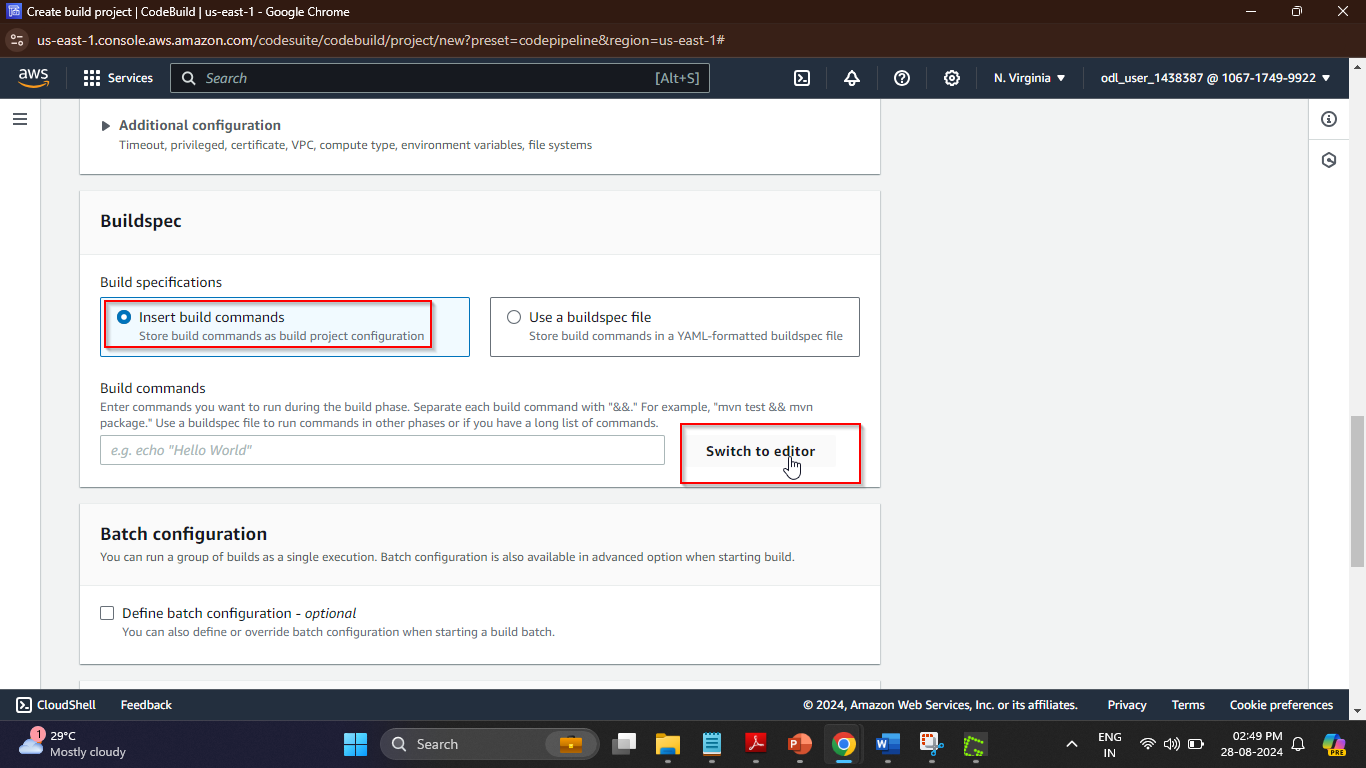
1. Select the **master** option under **Branch**

****

1. Click the **Next** button  
     
   
2. Select **AWS CodeBuild** under **Build provider** for building and automating tests  
     
   
3. Click on the **Create project** button; a pop-up window will appear  
     
   
4. In the new window, add the name of the project  
     
   
5. In the **Environment** section, add the details as shown in the screenshot below:

****

****

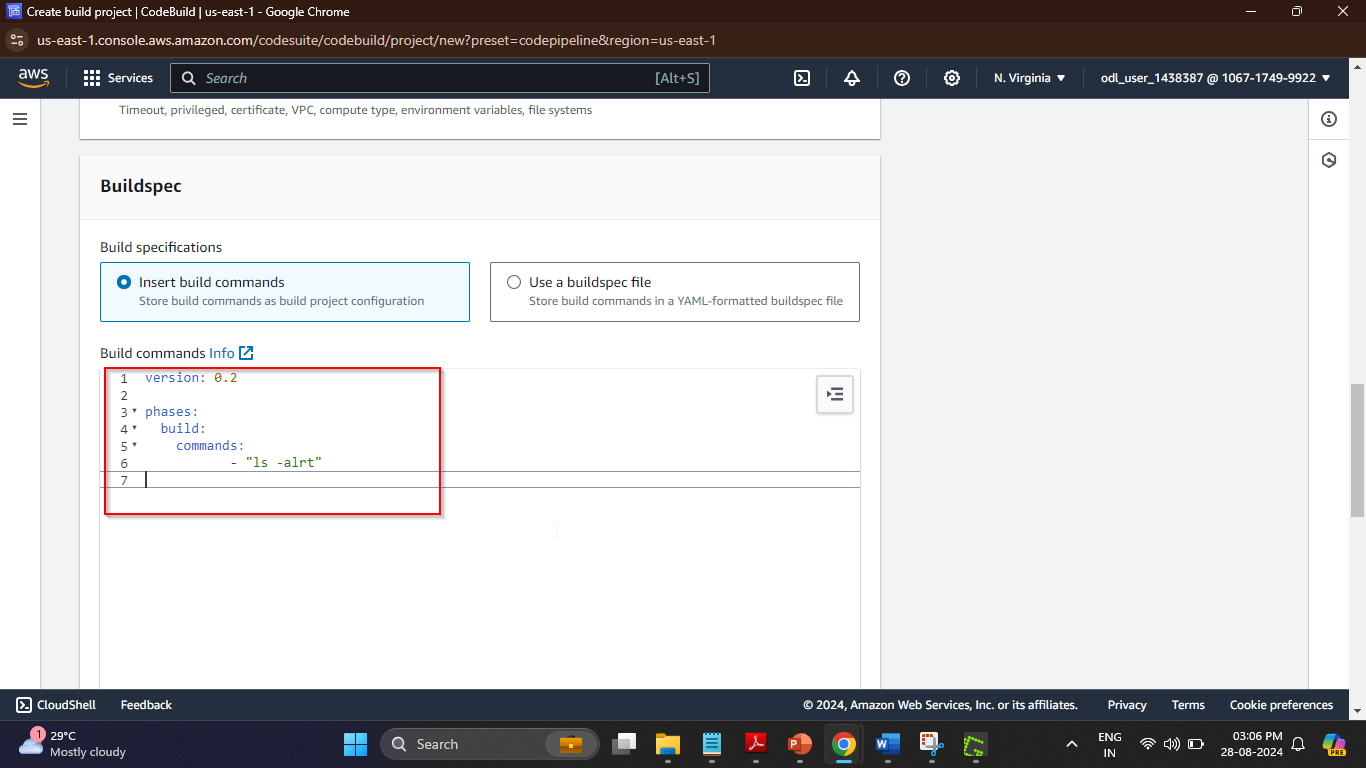
1. Under **Buildspec**, select the **Insert build commands** option and click on the **Switch to editor** button  
     
   
2. Remove the existing build commands from the editor and enter the following YAML code:

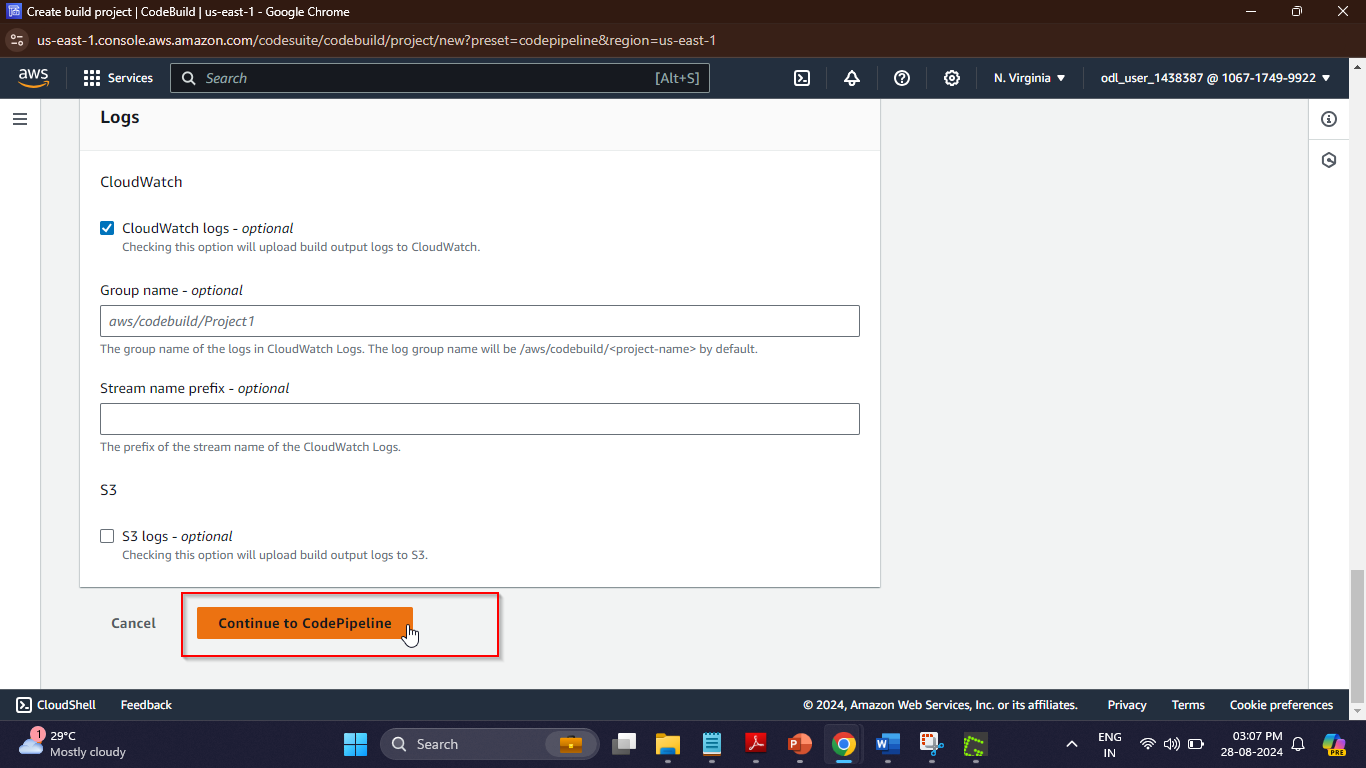
**version: 0.2**

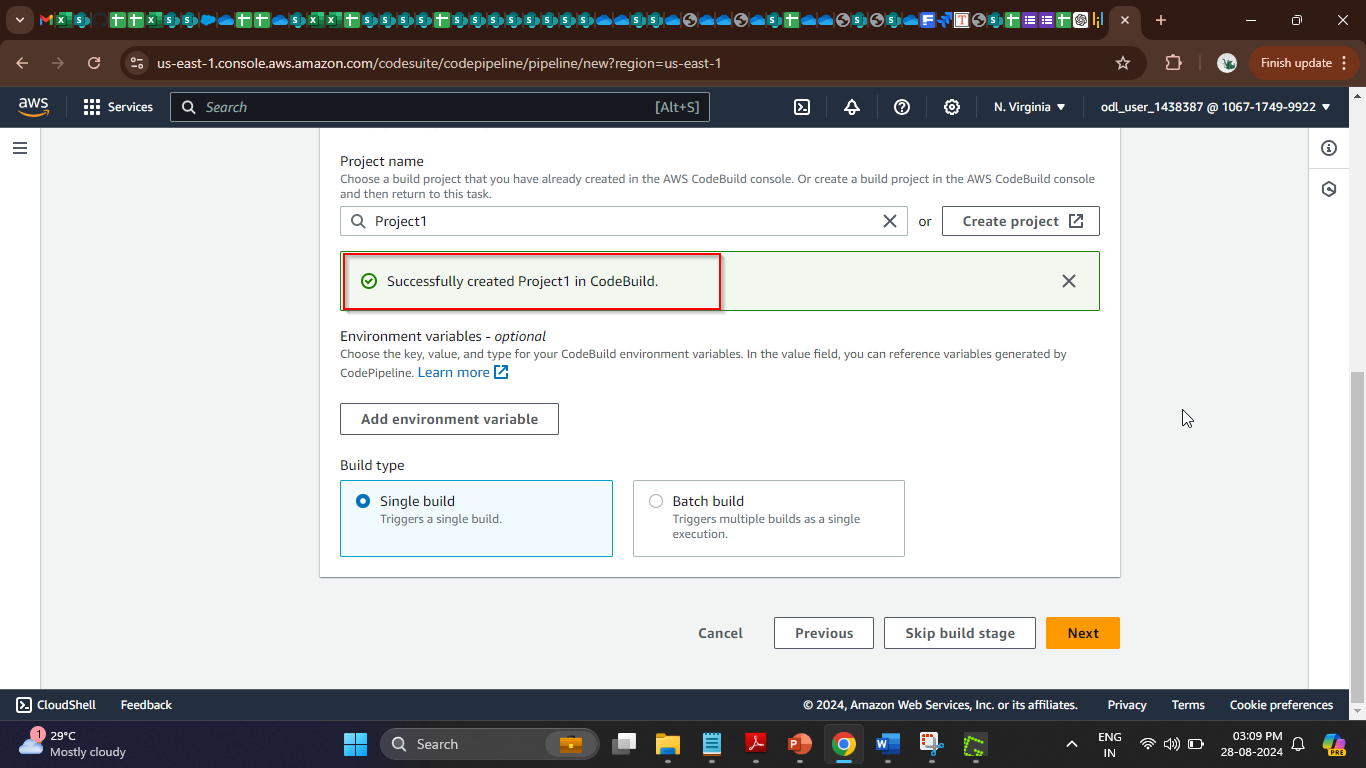
**phases:**

**build:**

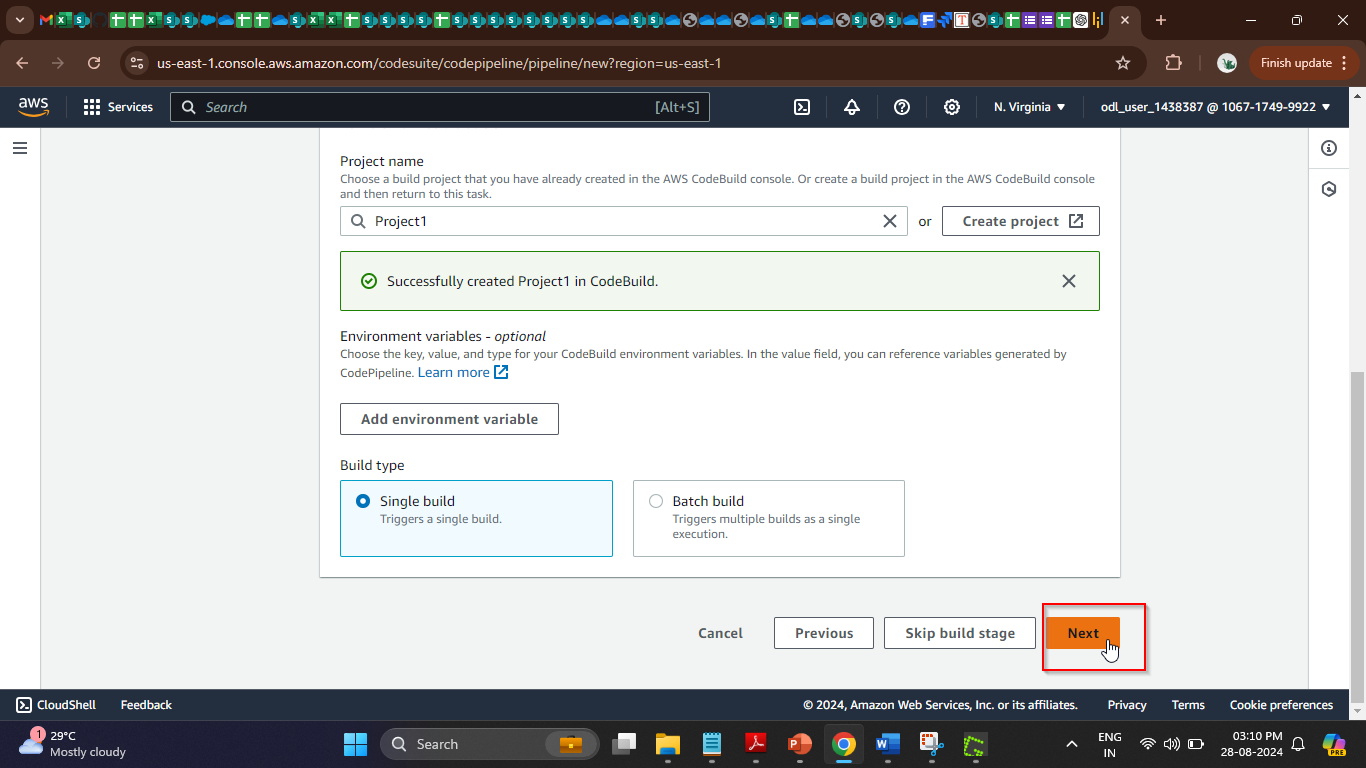
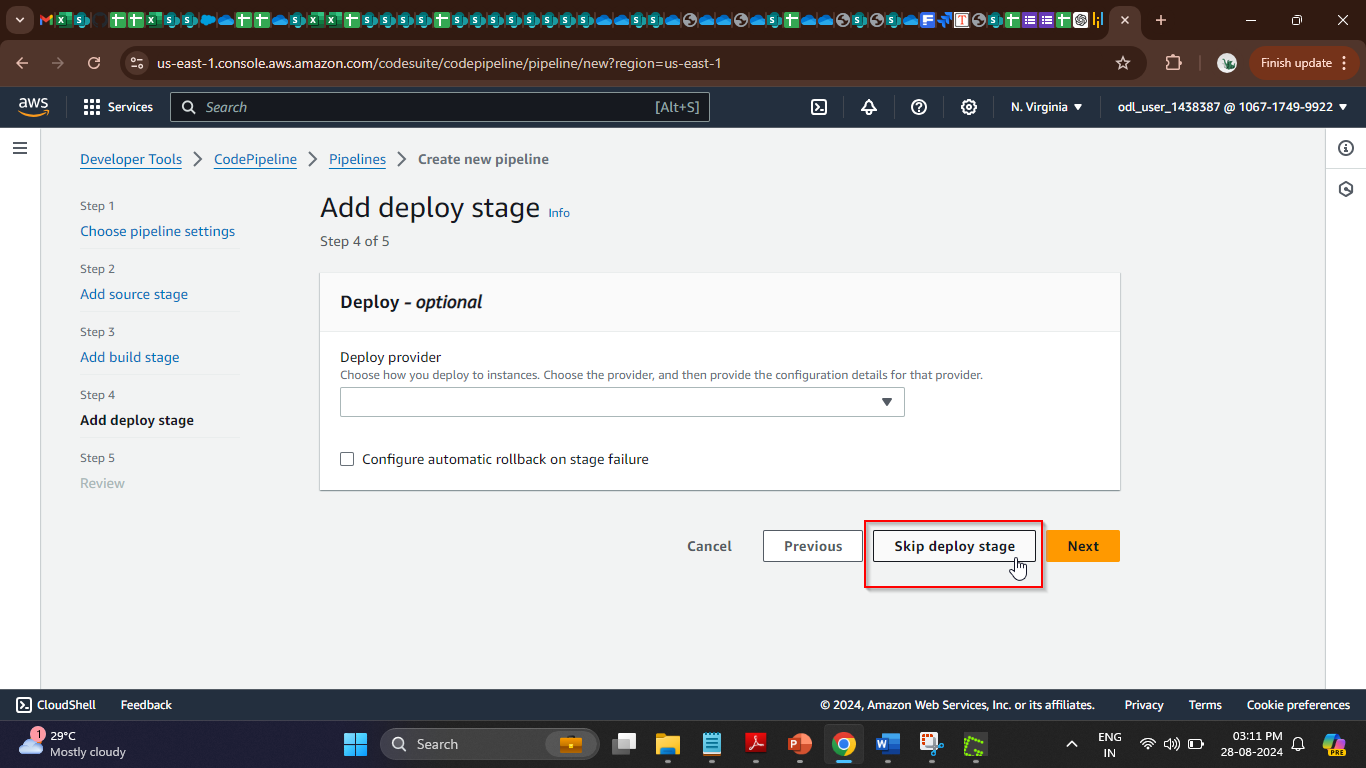
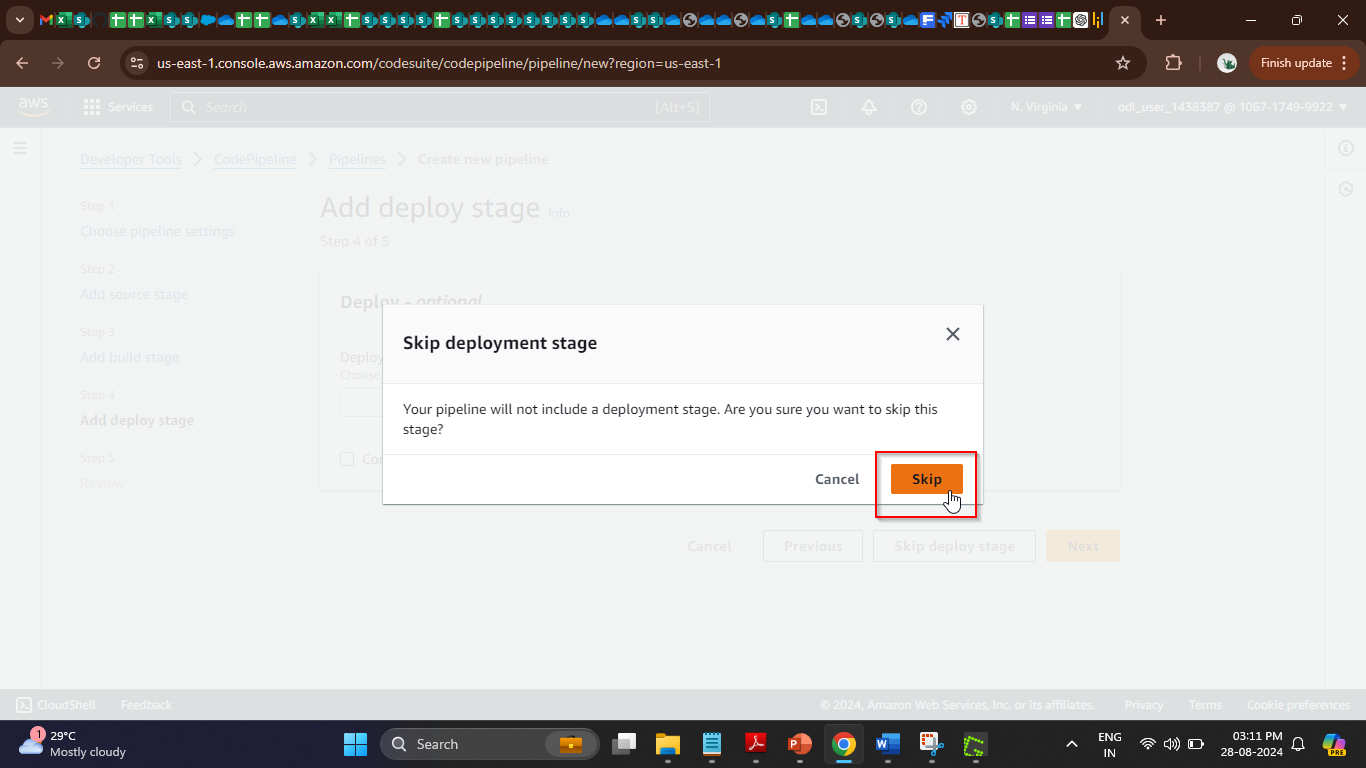
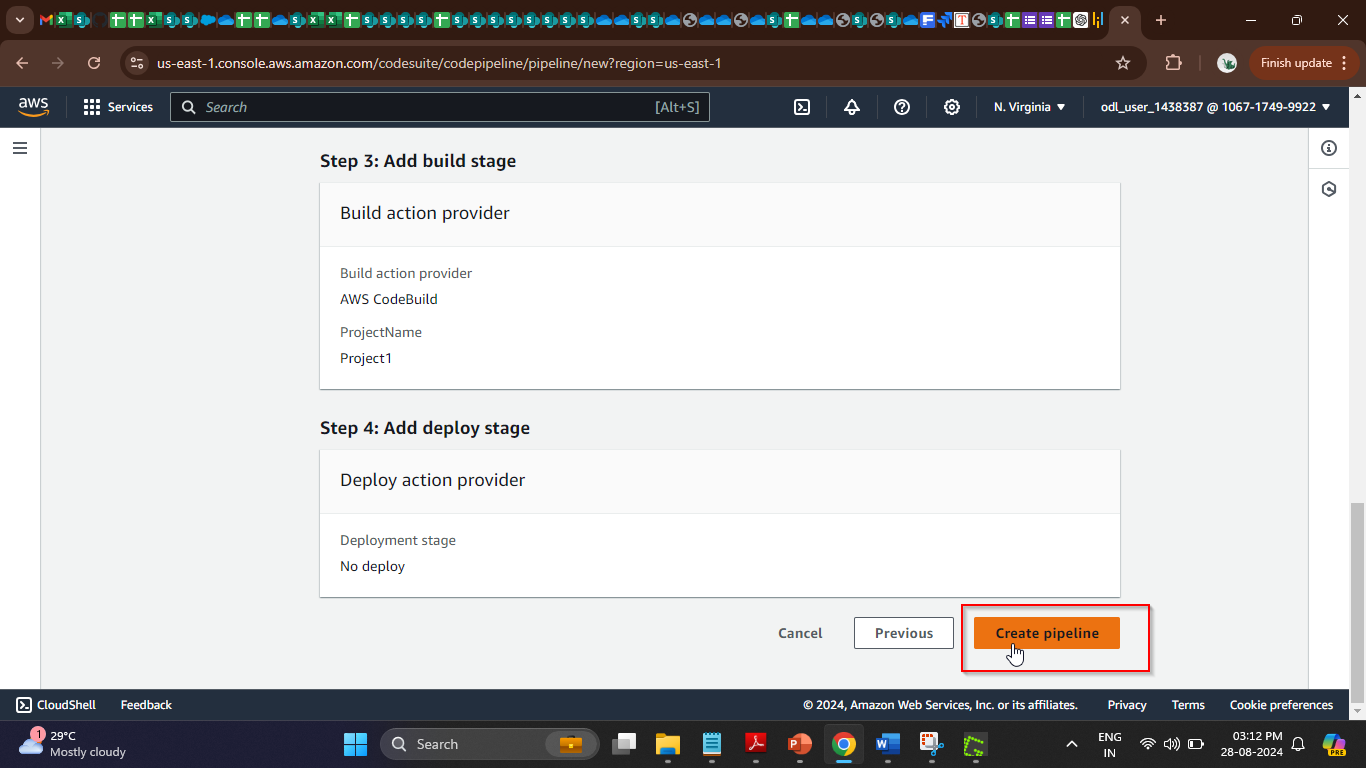
**commands:**

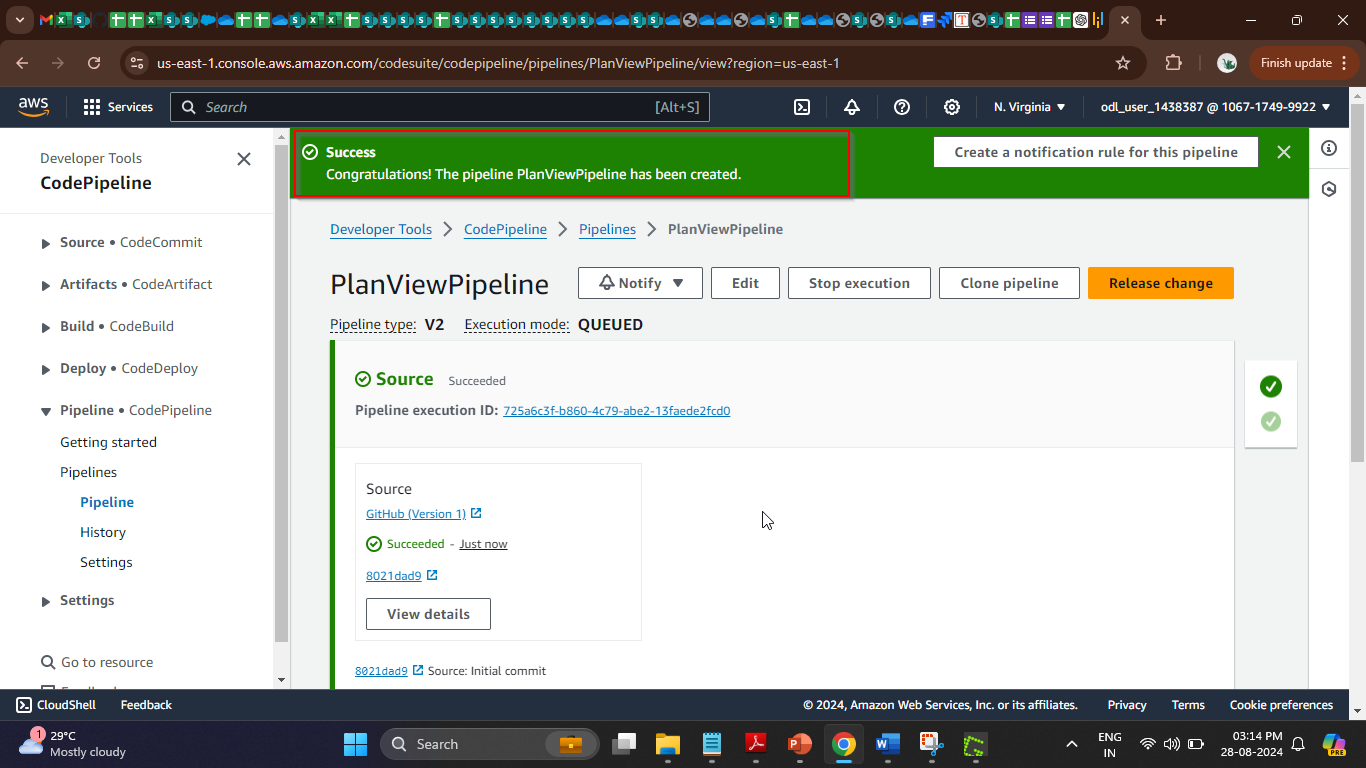
**- "ls -alrt"   
  
**

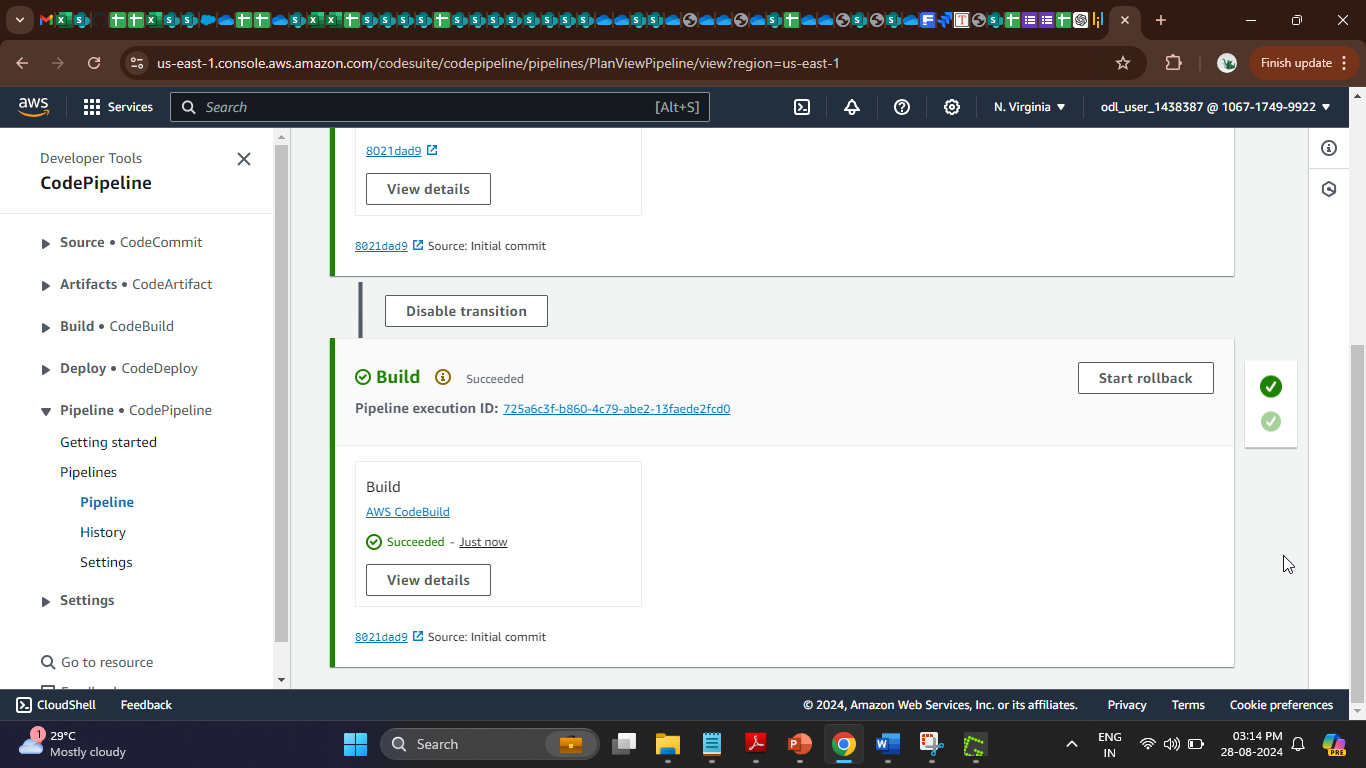
1. Scroll to the bottom of the page and click on the **Continue to CodePipeline** button ****

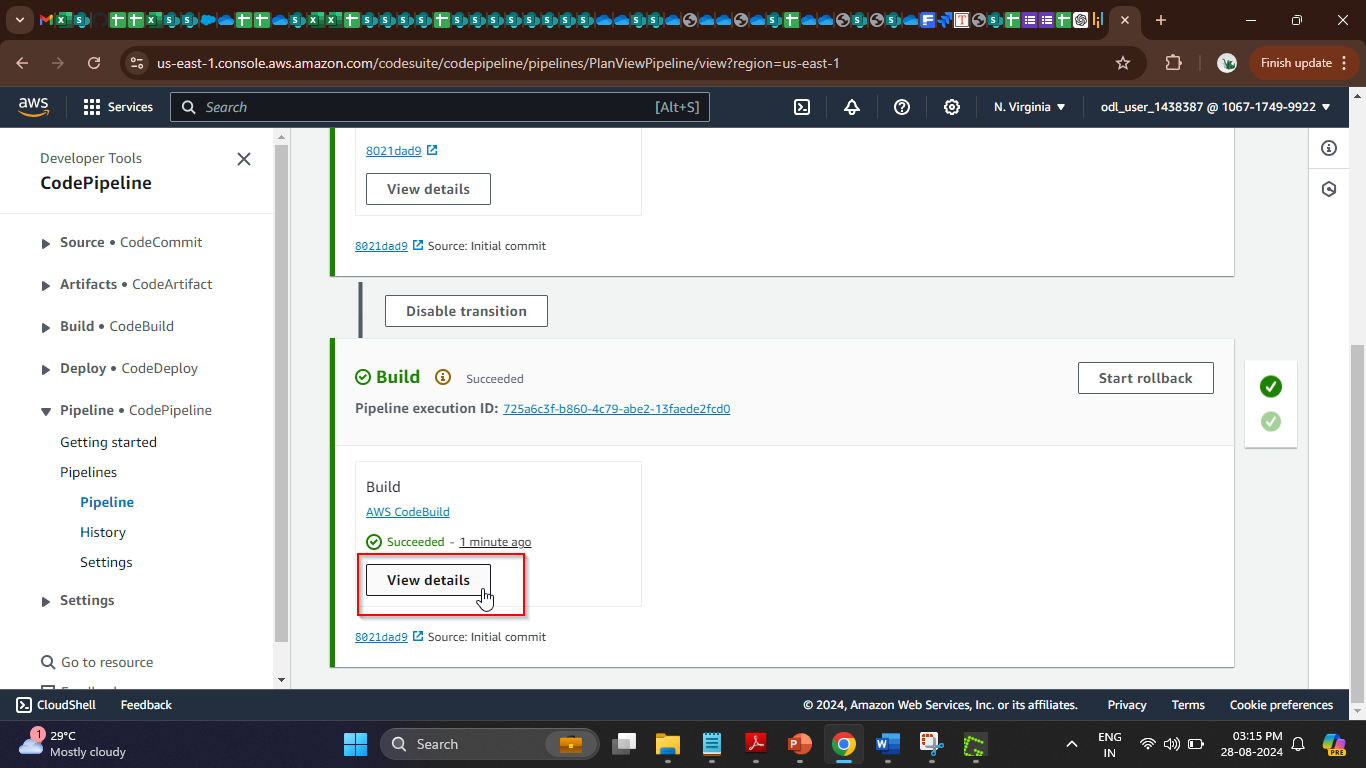
****

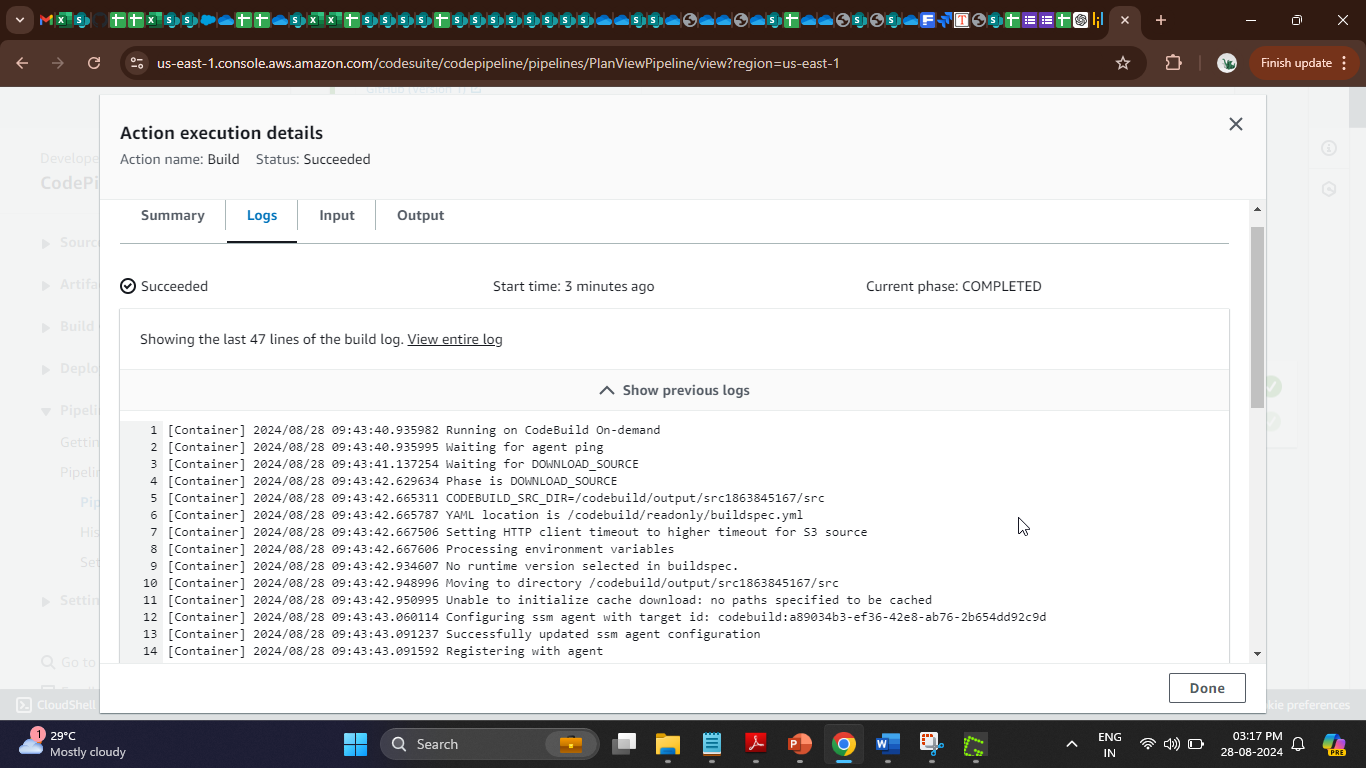
The project has been successfully created.

1. Now, click on the **Next** button  
     
   ****
2. Click on the **Skip deploy stage** button and then on the **Skip** button to finalize the   
    pipeline  
     
     
    ****
3. Scroll to the bottom of the page and click on the **Create pipeline** button ****

After creating the pipeline, the execution starts.  
  




1. Once the build stage is completed, click on the **View details** button in the **Build** section for validation ****

****

By following these steps, you have successfully created an AWS CodePipeline for continuous integration and delivery (CI/CD) using a GitHub repository.