**Terraform Automation for Frontend Services**

**Frontend components are**

1. Auth
2. Common
3. Dashboard
4. Header
5. Webapp

**Step1:** Here we created 2 Jenkins pipelines

**Jenkins Pipeline-1:**

1. By using this Pipeline, we can create S3 and CloudFront, adding the bucket polices.
2. And store the terraform output files in S3.

**Jenkins Pipeline-2:**

1. By using this pipeline, we can build the frontend components and store the artifacts to S3 bucket (Which is created by pipeline-1) and invalidate the CloudFront.

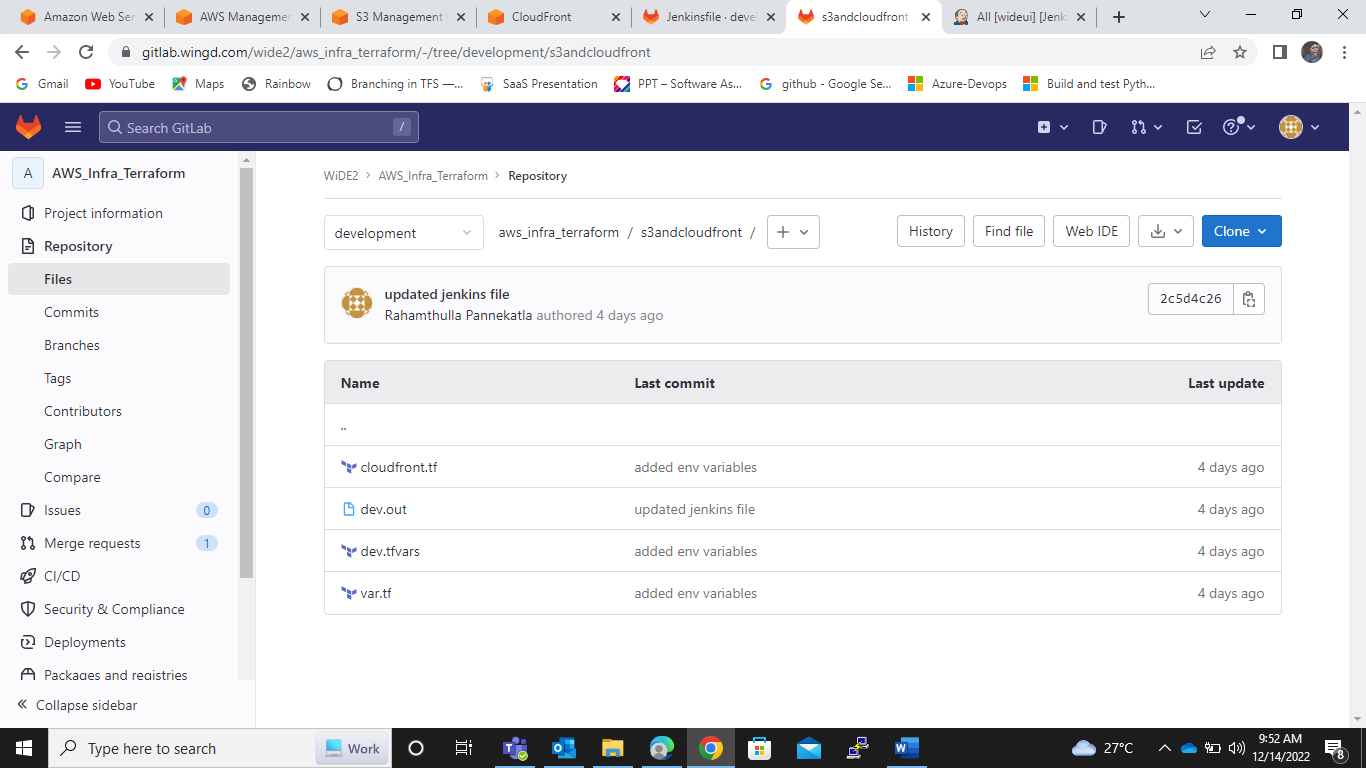
**Step2: Terraform script**

* Here we have the 3 files

1. Main.tf (cloudfront.tf)
2. Var.tf (Variables)
3. dev.tfvars (Note: you can change the as per your environment)

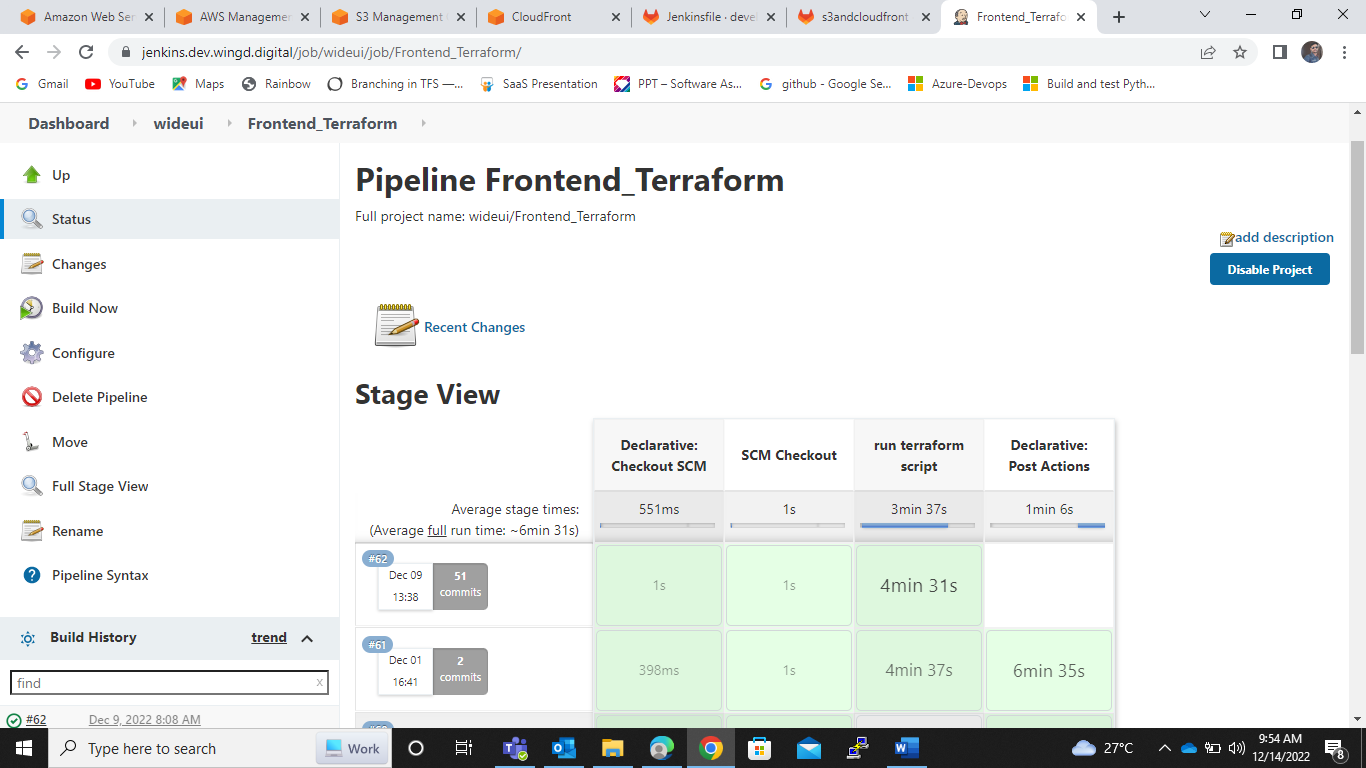
The scripts are in below repo

<https://gitlab.wingd.com/wide2/aws_infra_terraform/-/tree/development/s3andcloudfront>

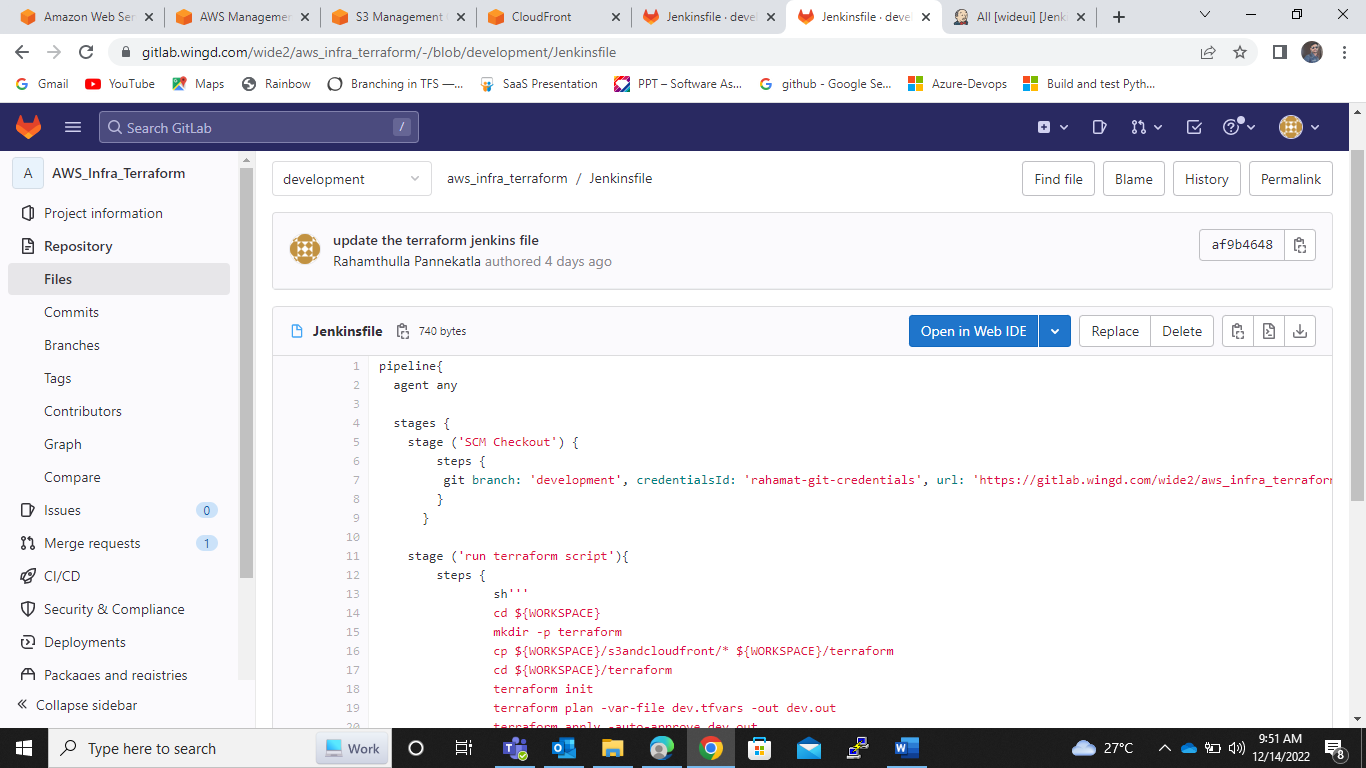


**Step3:** Trigger this terraform script through pipeline-1

<https://jenkins.dev.wingd.digital/job/wideui/job/Frontend_Terraform/>



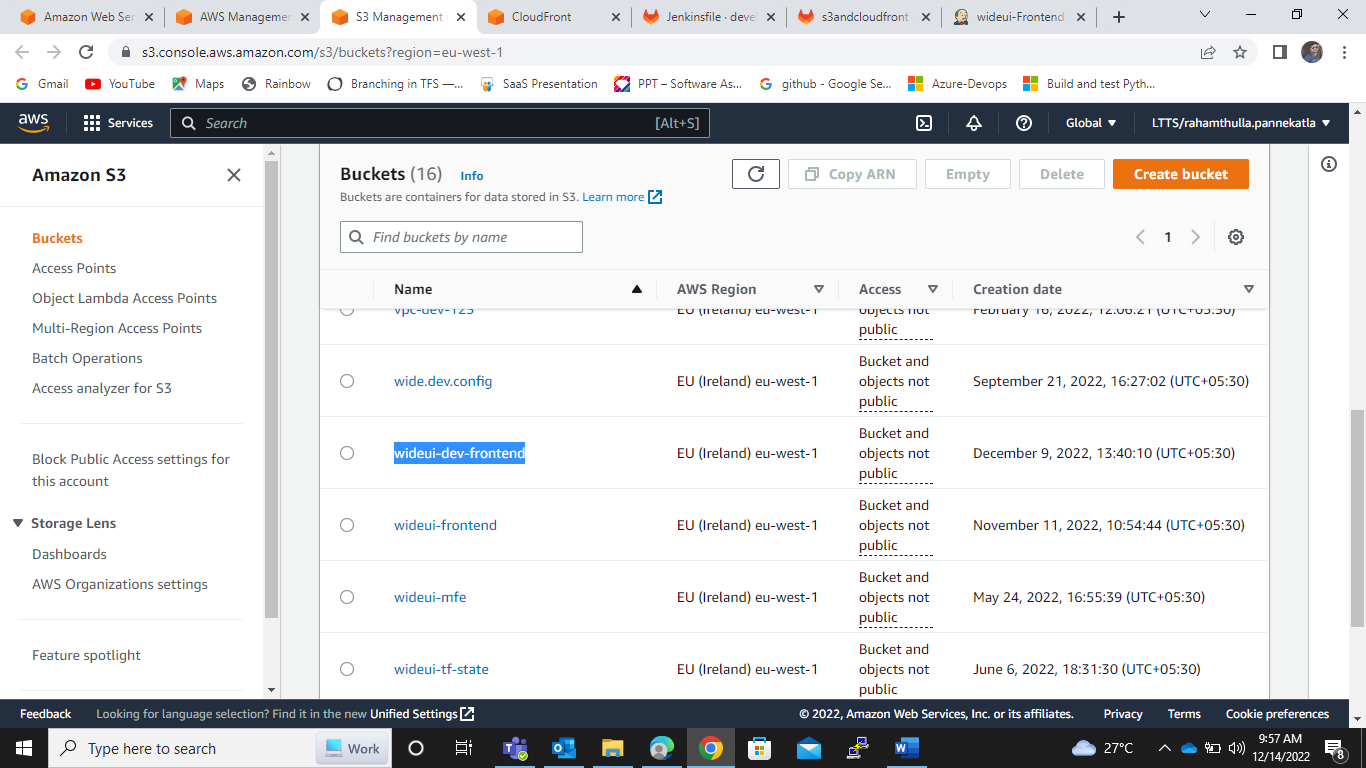
Pipeline is available in below repo



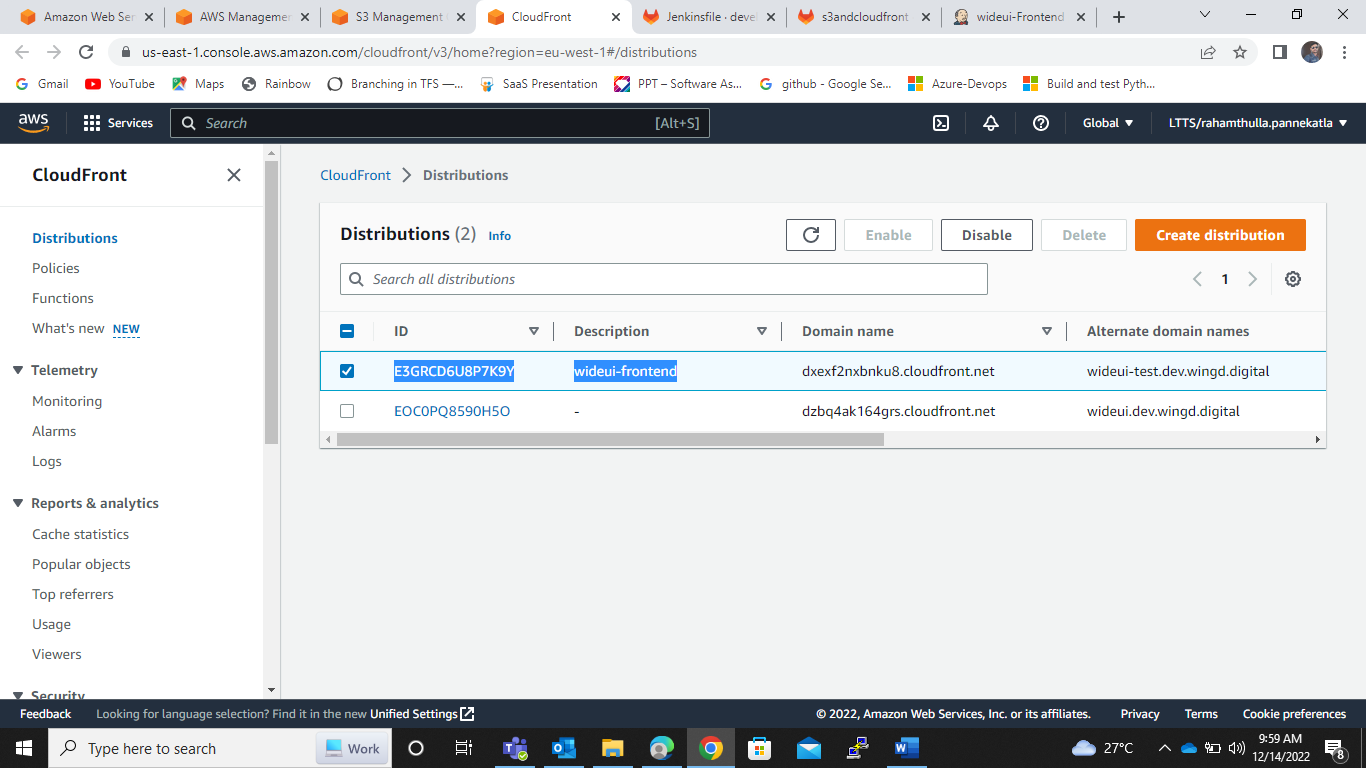
<https://gitlab.wingd.com/wide2/aws_infra_terraform/-/blob/development/Jenkinsfile>

**Step4:** After pipeline is build. The S3 and cloud front was created

S3 bucket name: wideui-dev-frontend



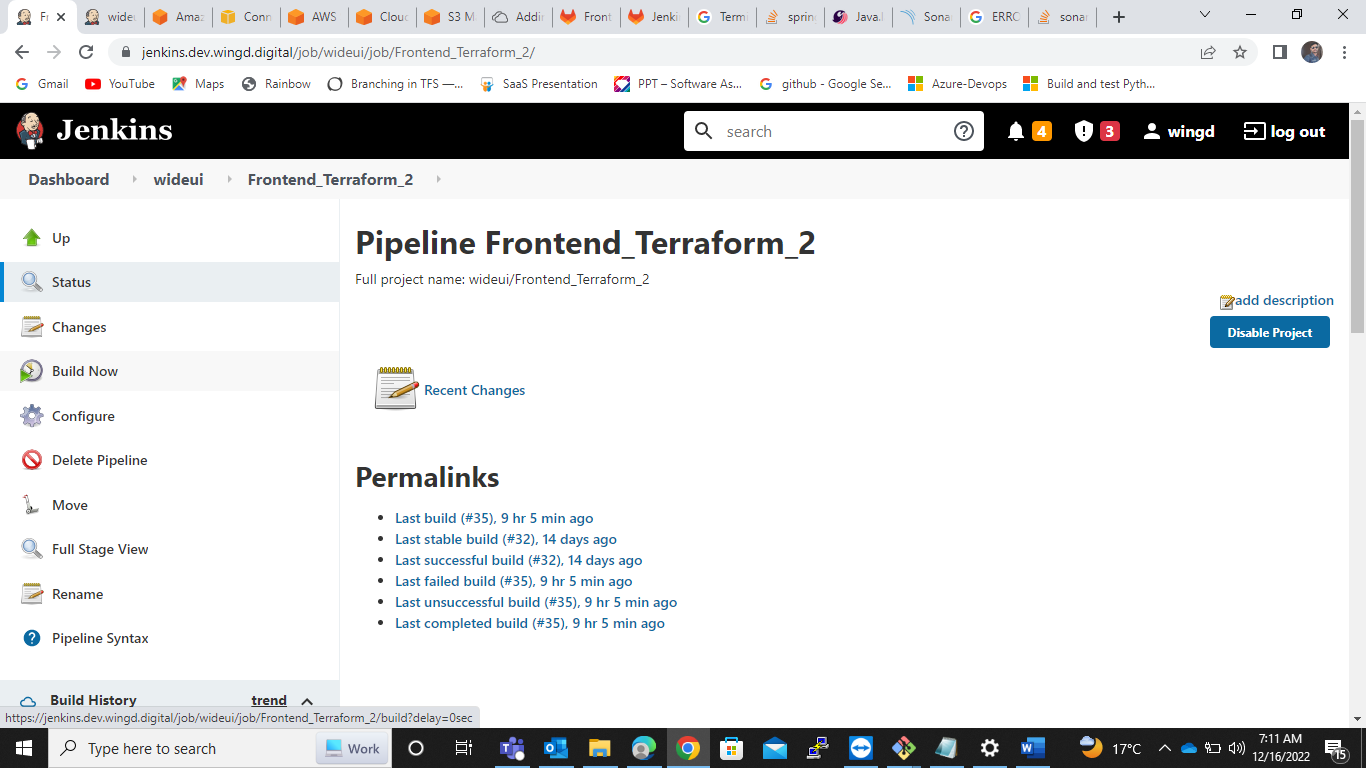
Cloudfront created through pipeline-1



Manual step: After created the domain name in cloudfront through terraform script. We don’t have SSL certificate in Dev environment to create a DNS record for new domain. So for this new domain we create a DNS record in AWS-master

**Step5:** Trigger the pipeline-2

<https://jenkins.dev.wingd.digital/job/wideui/job/Frontend_Terraform_2/>



By using this pipeline2 we will build the frontend components and store the artifacts to new S3 bucket that is created through pipeline-1

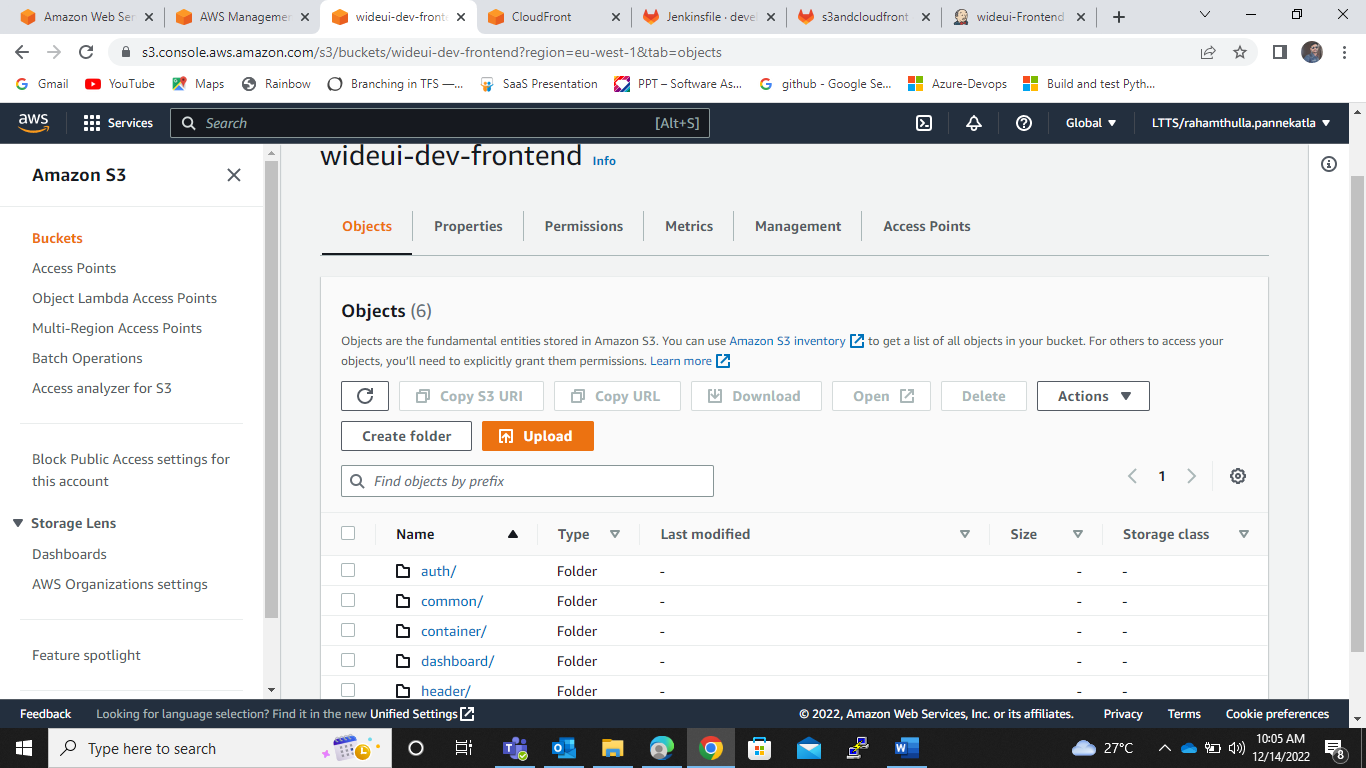
Pipeline is available in below repo

Graphical user interface, text, application

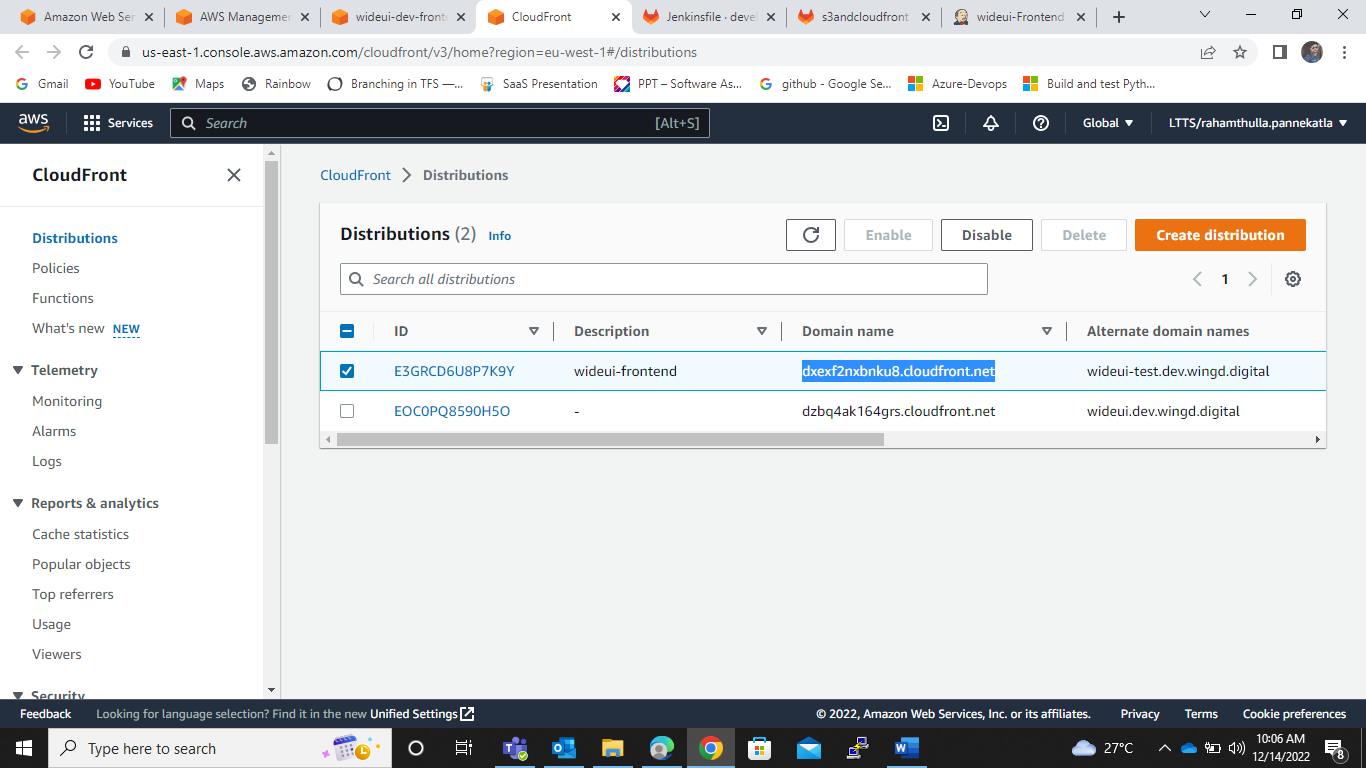
Description automatically generated

<https://gitlab.wingd.com/wide2/wideui/-/blob/development/Jenkinsfile>

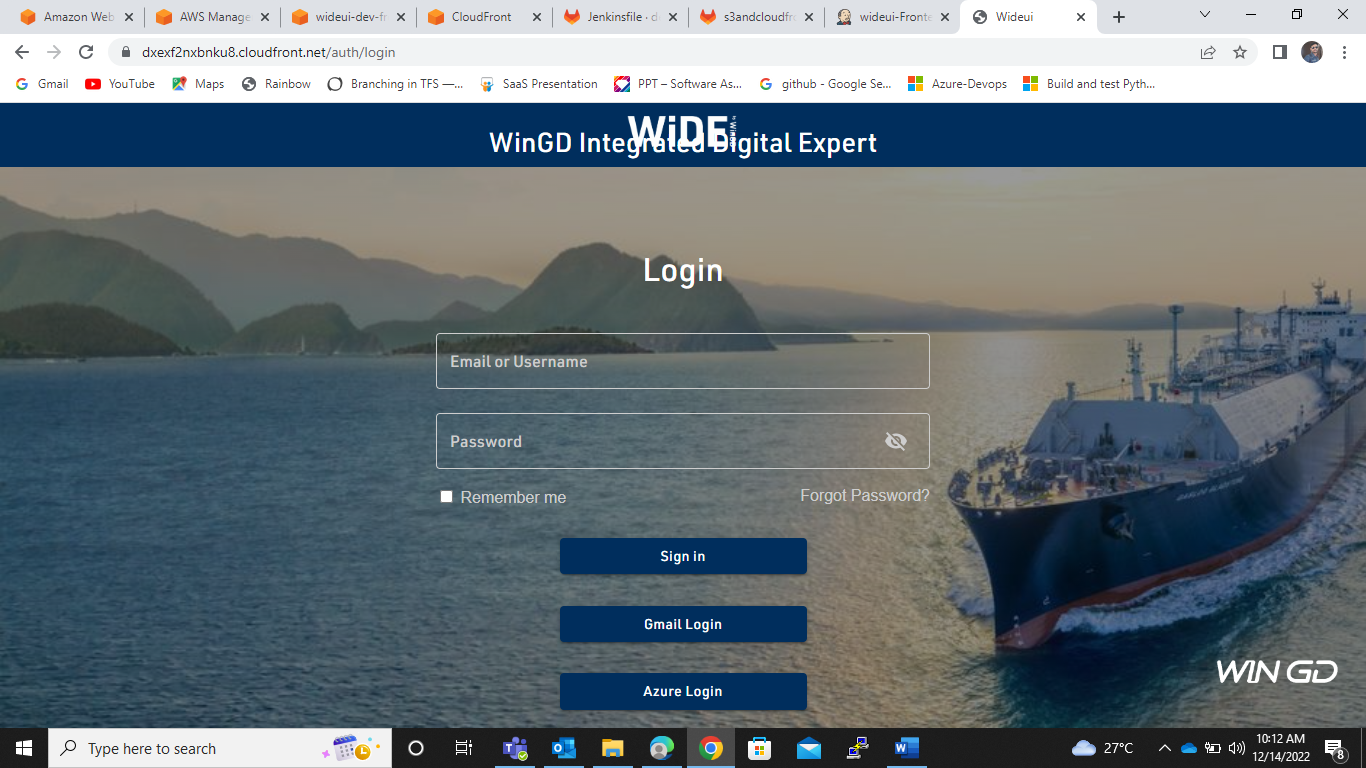
**Step6:** After build the pipeline 2 the artifacts are stored into the S3 bucket and invalidate the CloudFront



**Step7:** Access the new cloudfront distribution URL. Which is created by pipeline1



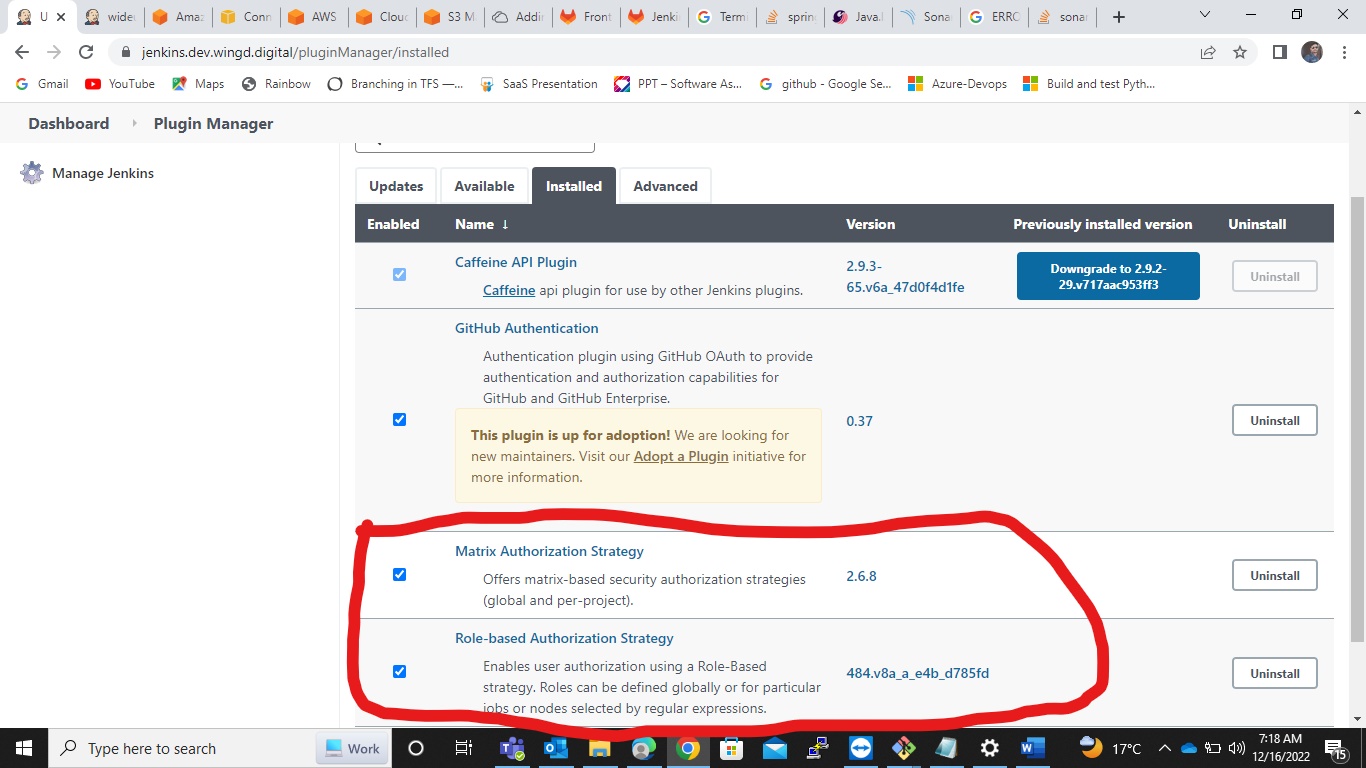
Access the CloudFront distribution URL



**Step8:** Restrict the two pipelines by using Jenkins

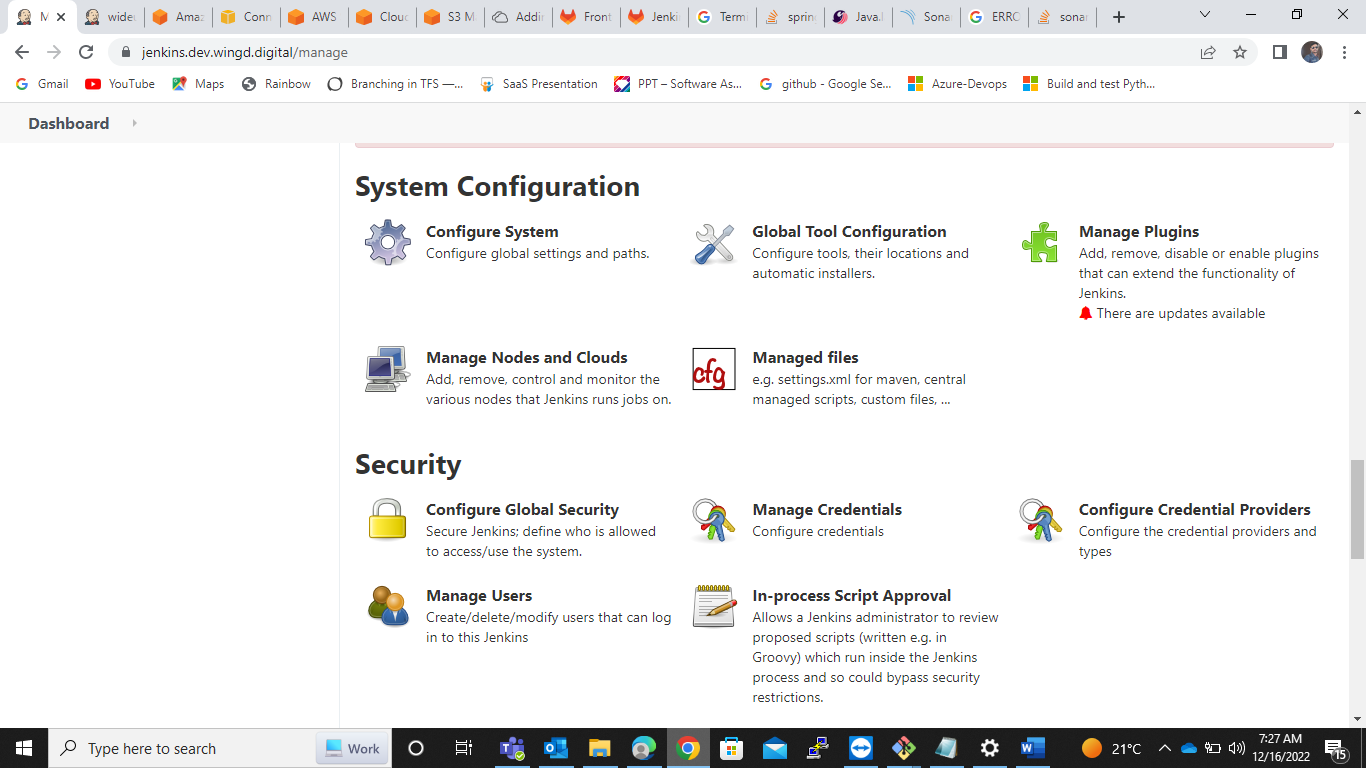
1. Install project based authorization plugin in Jenkins

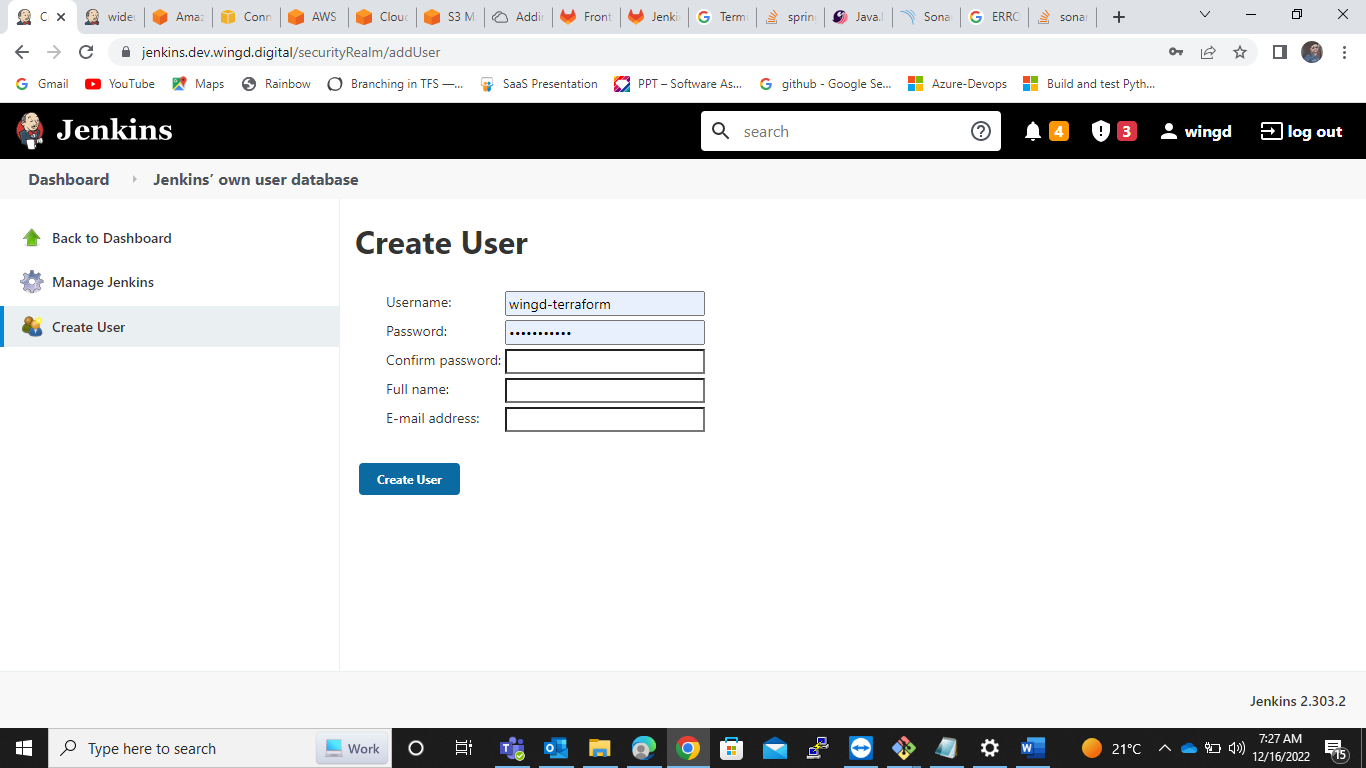
* Goto the manage jebkins and install project based authorization plugin



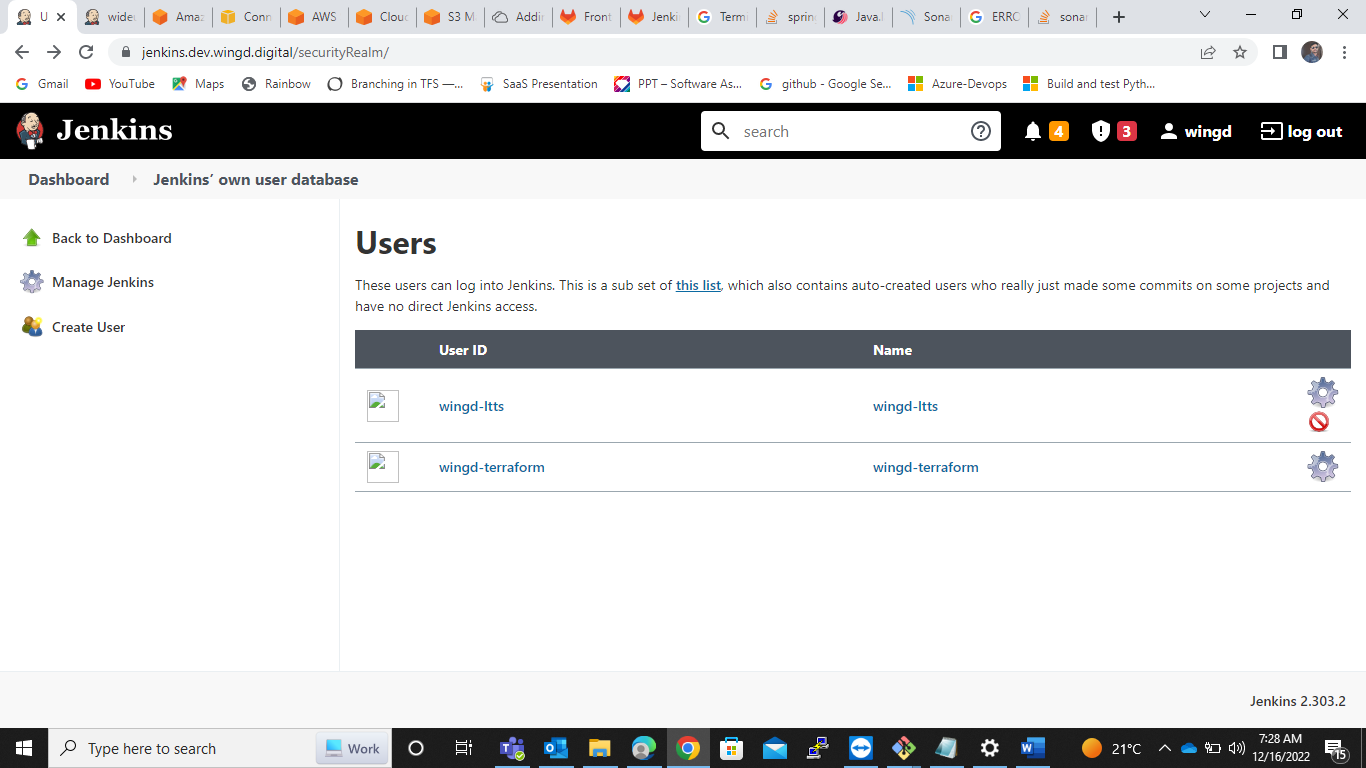
1. create a new user in Jenkins

Go to the **manage Jenkins** in that select the **manage users**





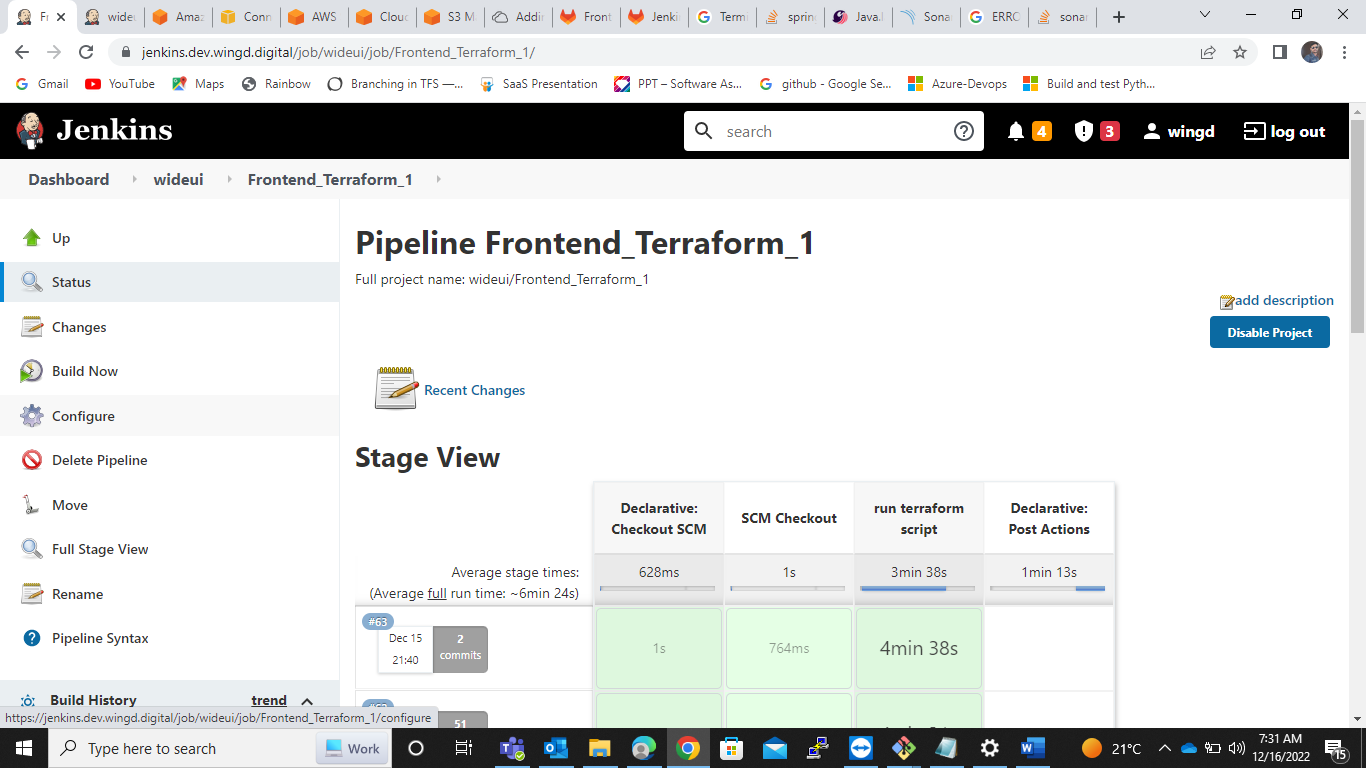
After creating the user



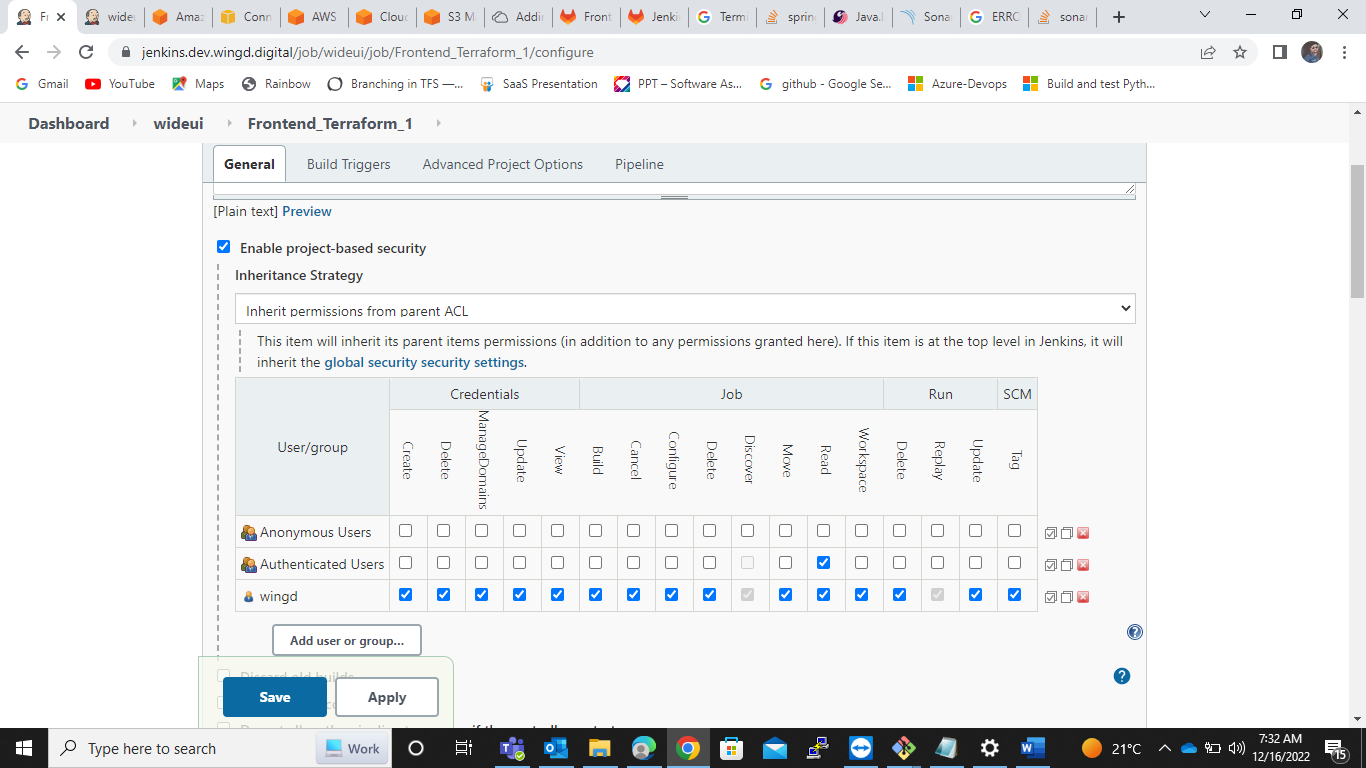
1. Add permission to the Pipeline 1&2 for particular user only

* Go to the Our first Jenkins job( Frontend\_Terraform\_1). This is our pipeline-1

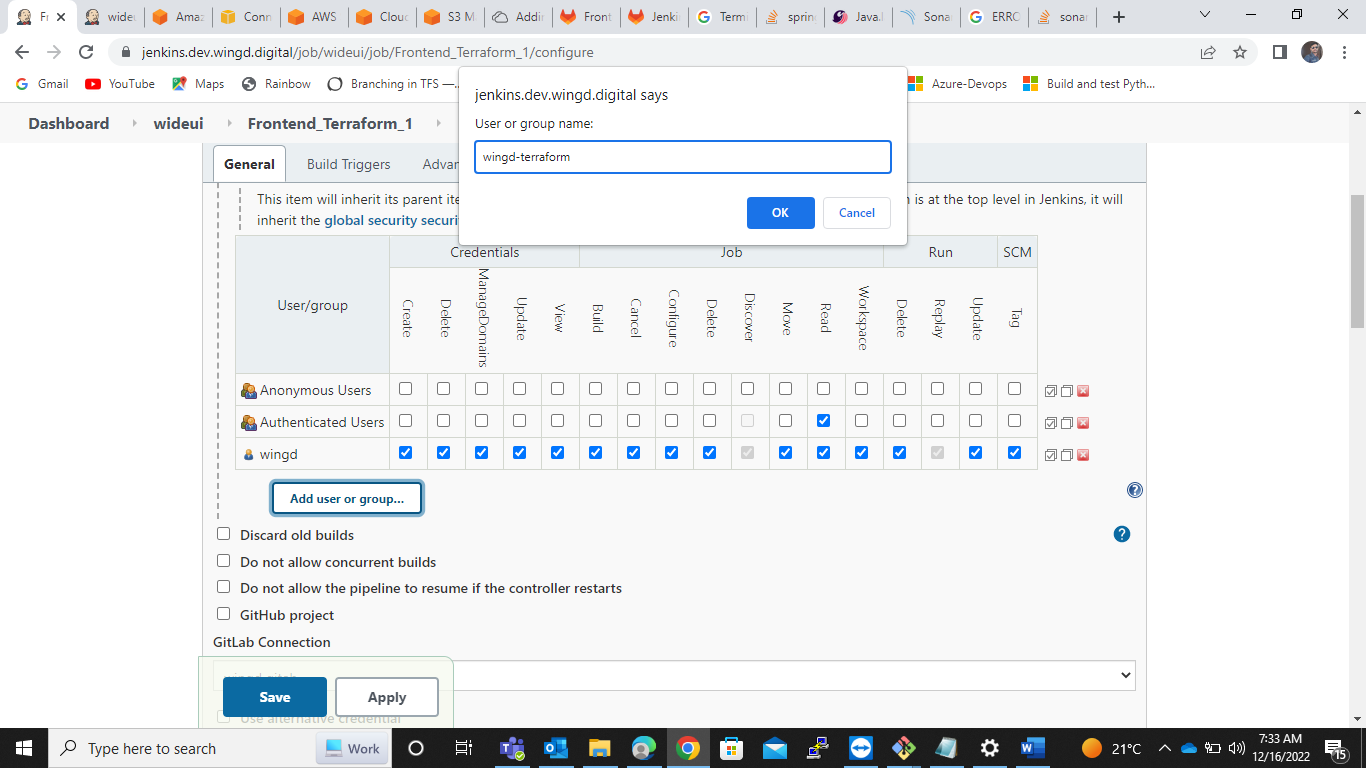
Select configure



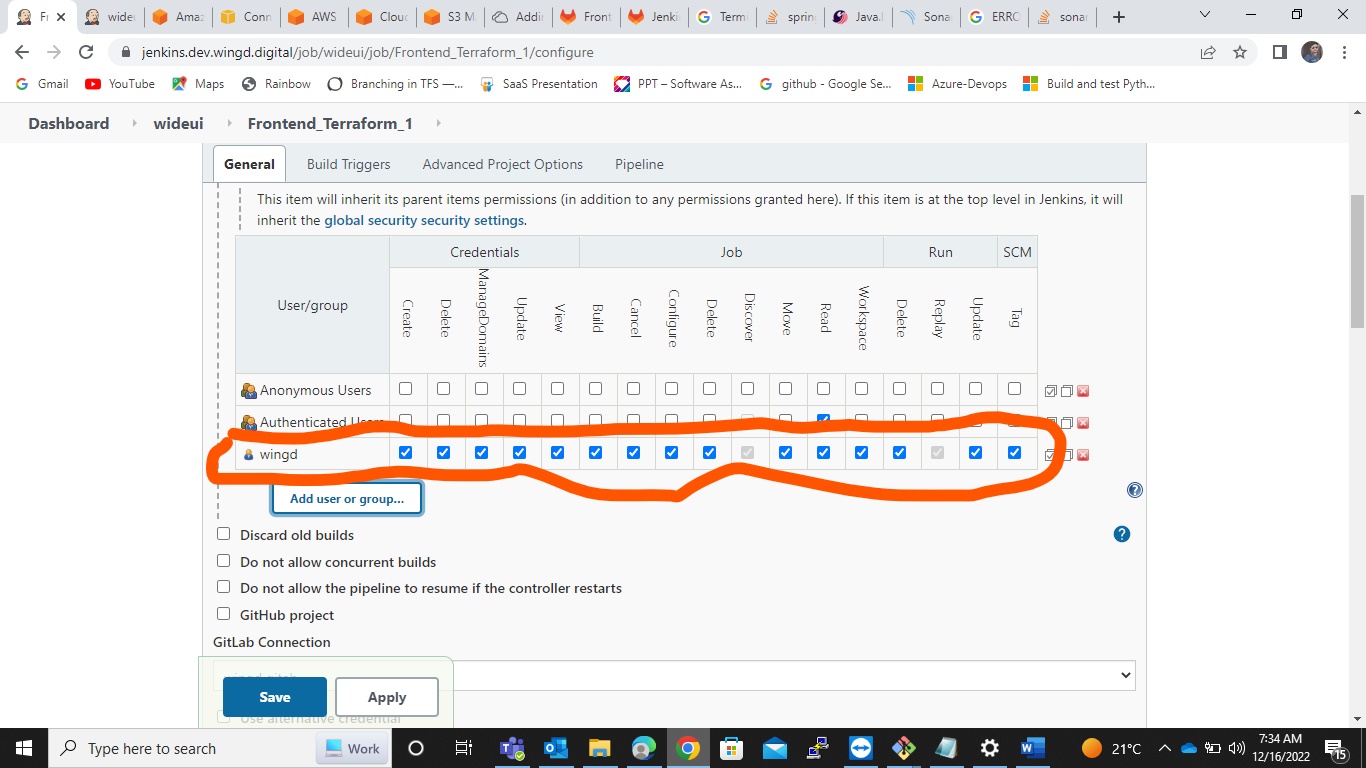
Enable the **Project-based-security**



In that we can select a **Add a user group** to give the restrict the permissions



After adding the user group given the permissions as below



1. Same process to pipeline-2 (Frontend\_Terrafrom\_2) also, Add user and permissions.