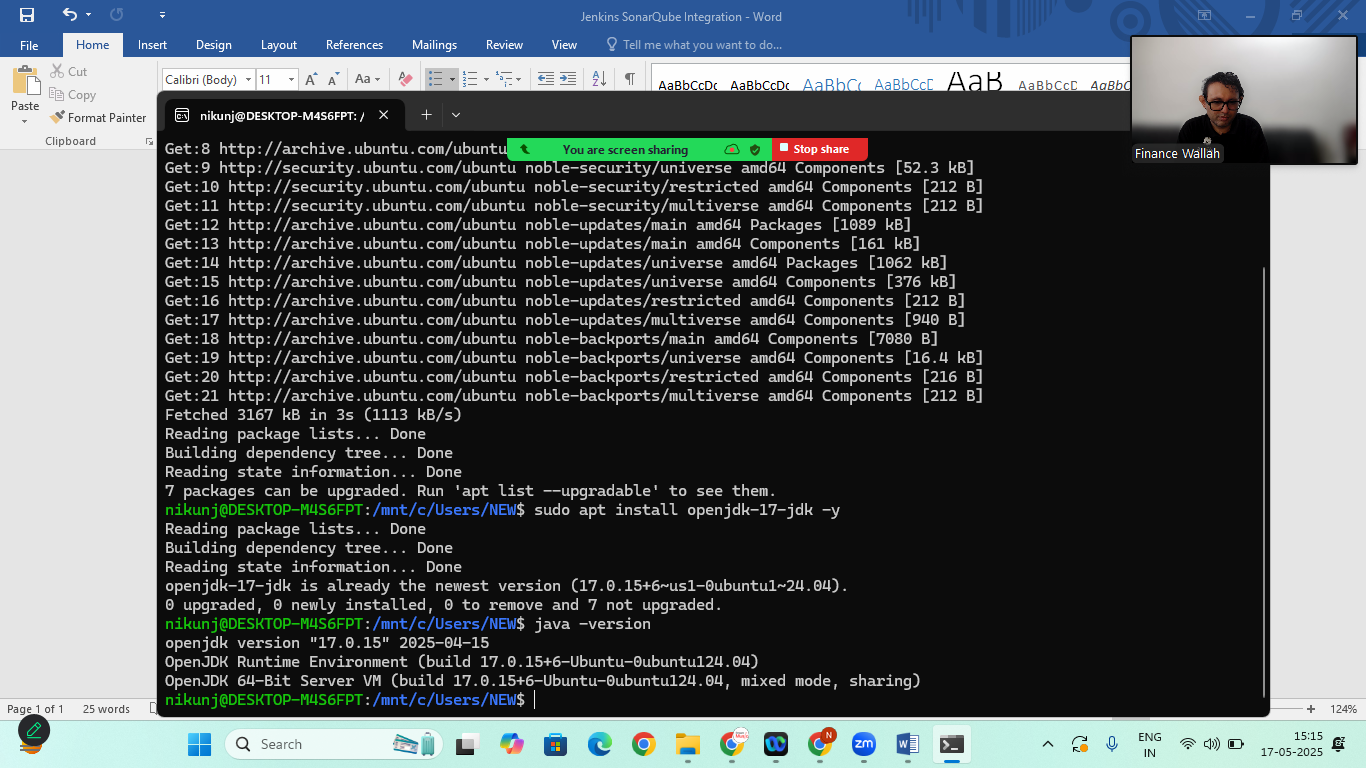
Step:1 Install SonarQube on WSL

Requrements for this is:

Machine must have jdk-17 in wsl

* sudo apt update
* sudo apt install openjdk-17-jdk –y
* java –version



Step:2 install PostgreSQl

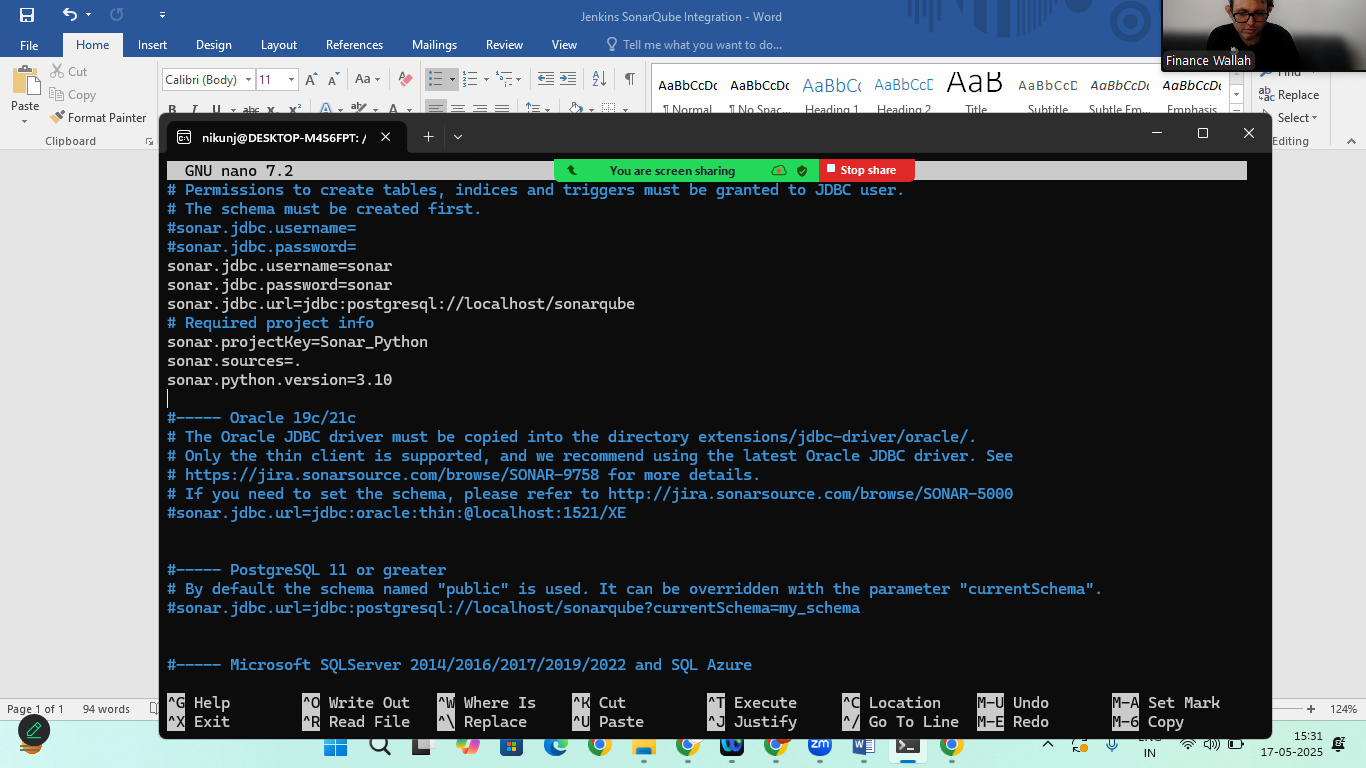
* sudo apt install postgresql postgresql-contrib –y

create sonarQube DB and User:

* sudo -u postgres psql
* CREATE USER sonar WITH PASSWORD 'sonar'
* CREATE DATABASE sonarqube OWNER sonar
* \q

Step:3 Install Required dependencies

* sudo apt install unzip wget gnupg2 –y
* cd /opt
* sudo wget <https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-10.5.1.90531.zip>
* sudo unzip sonarqube-10.5.1.90531.zip
* sudo mv sonarqube-10.5.1.90531 sonarqube
* sudo chown –R $USER:$USER sonarqube
* nano /opt/sonarqube/conf/sonar.properties



sonar.jdbc.username=sonar

sonar.jdbc.password=sonar

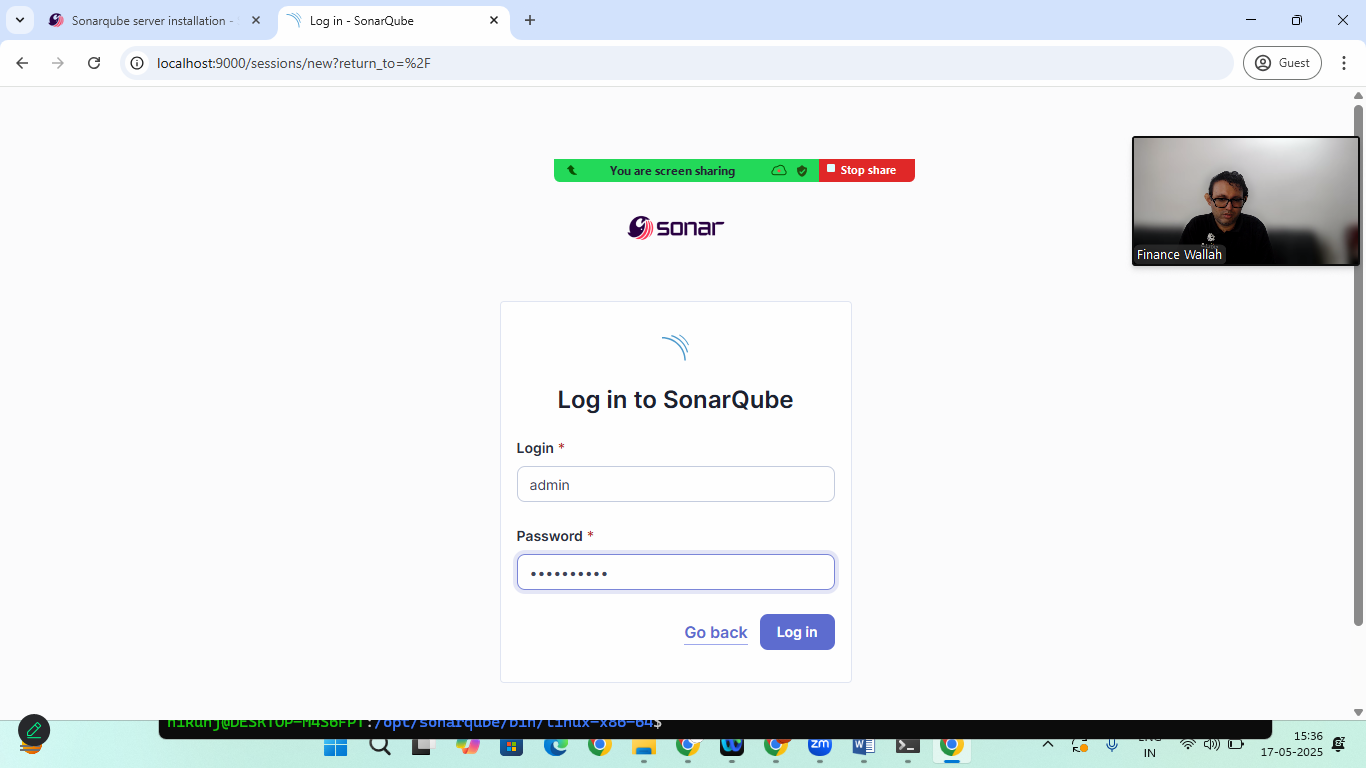
sonar.jdbc.url=jdbc:postgresql://localhost/sonarqube

Step:5 Run the SonarQube

* cd /opt/sonarqube/bin/linux-x86-64
* ./sonar.sh start

Check the status

* ./sonar.sh status

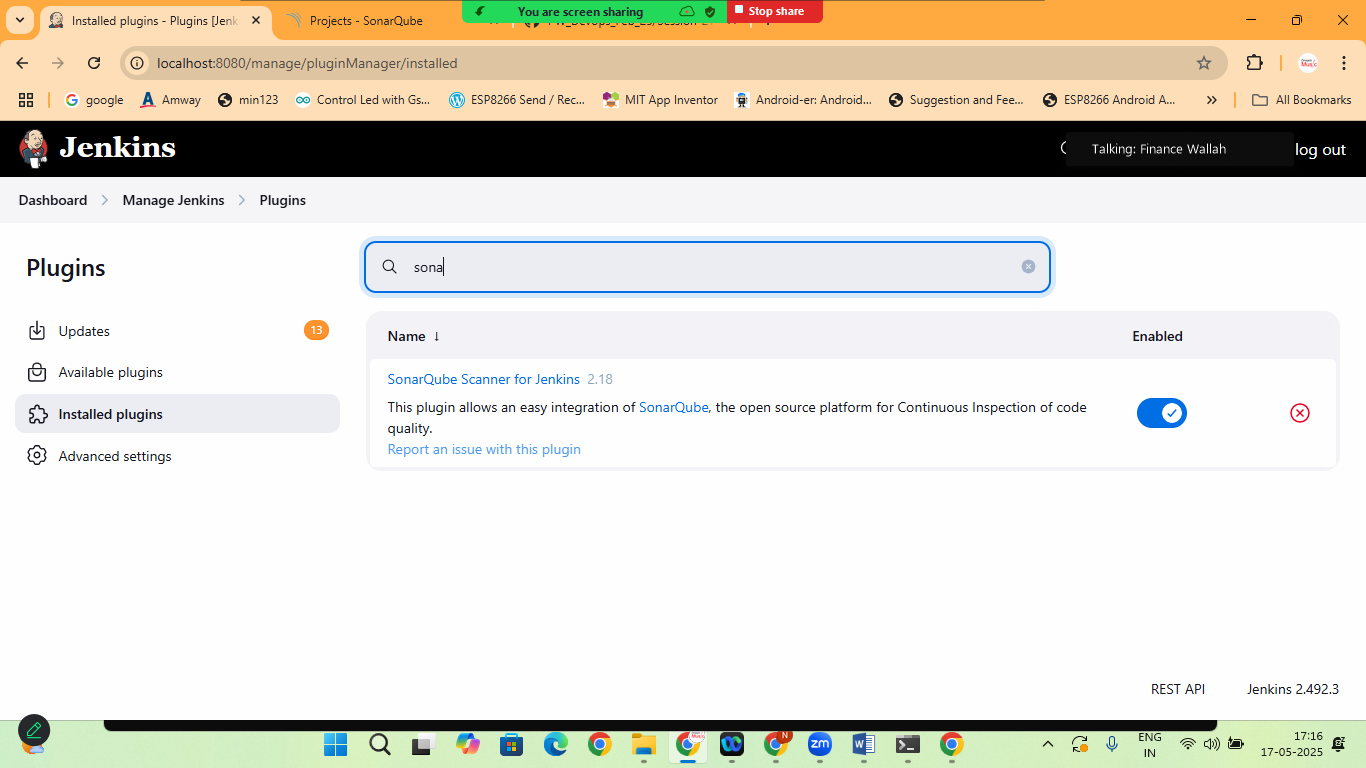


Login:admin

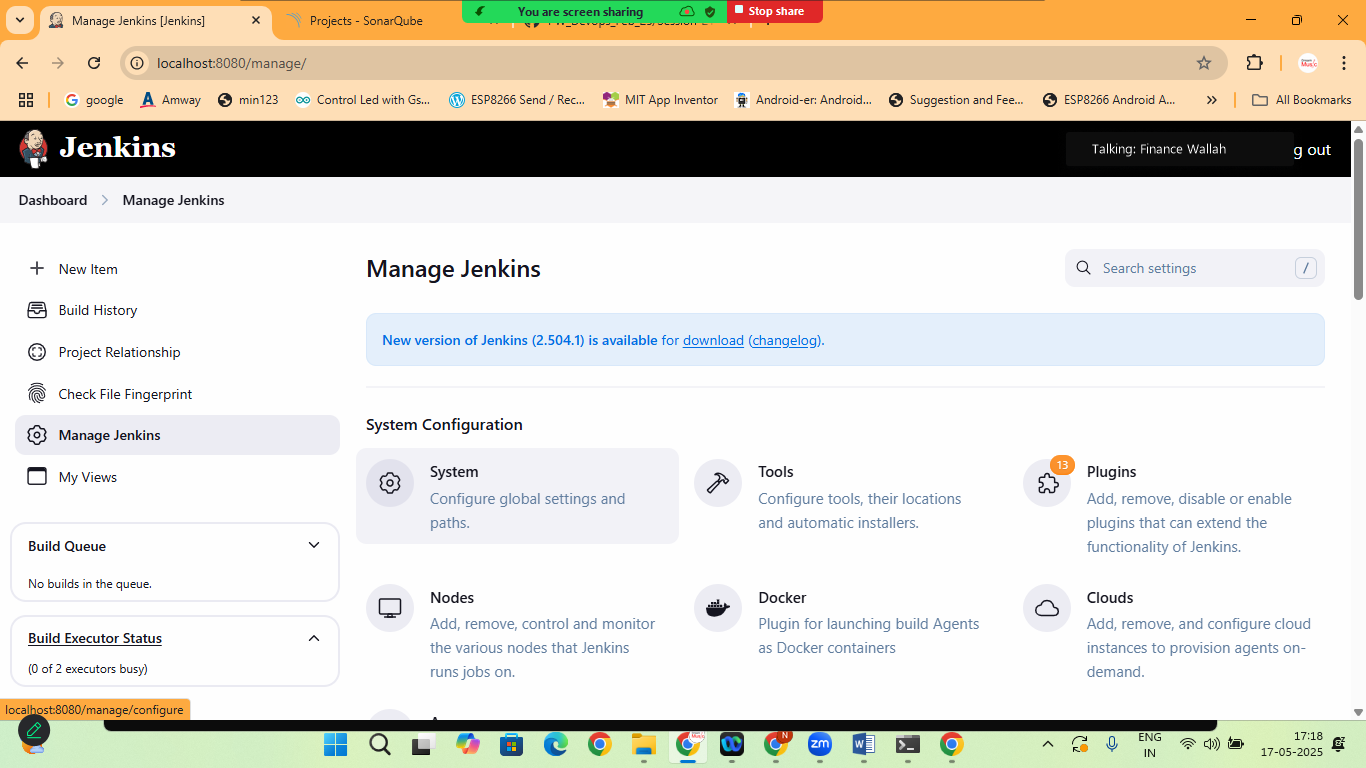
Password:admin

Starting with Jenkins Part

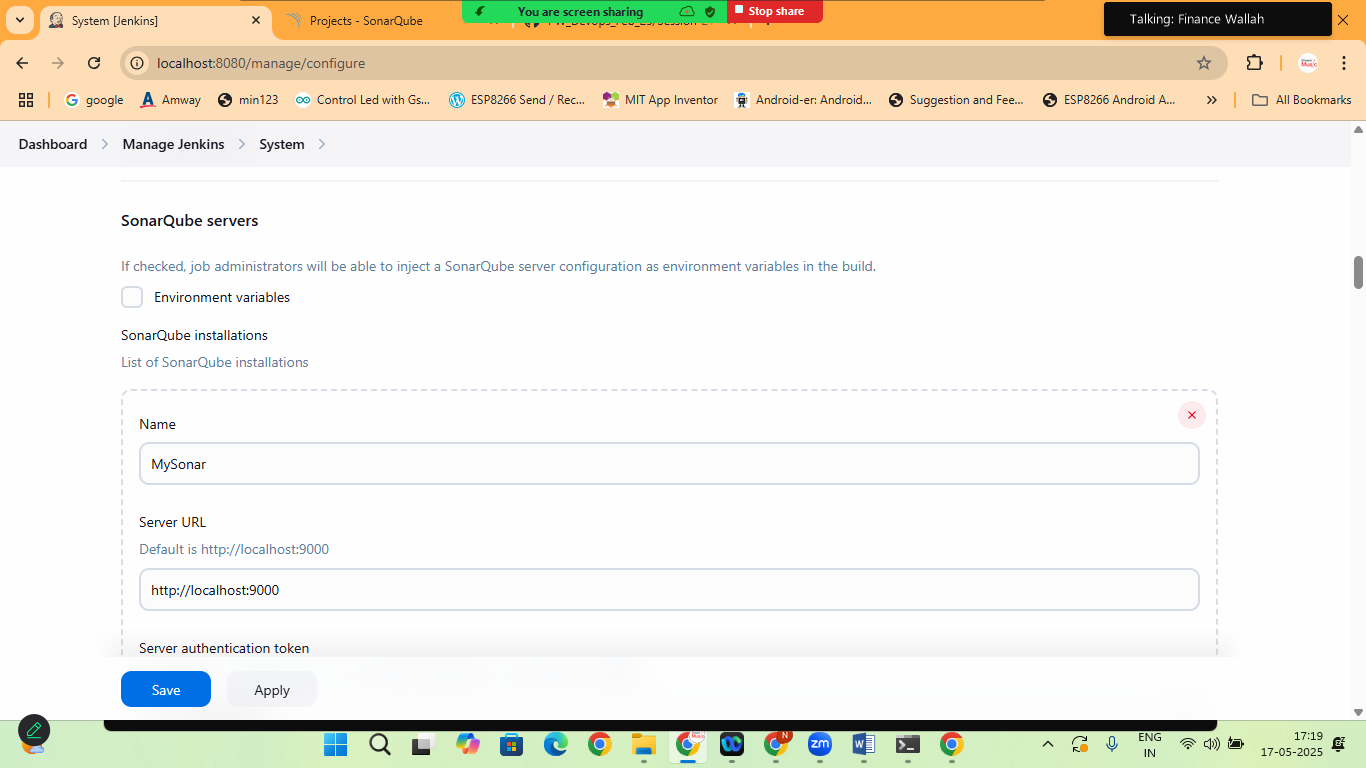
* Open wsl
* sudo syatemctl start Jenkins
* localhost:8080
* goto>manage Jenkins> plugins
* download SonarQubeScanner



Goto> manage Jenkins> system configuration



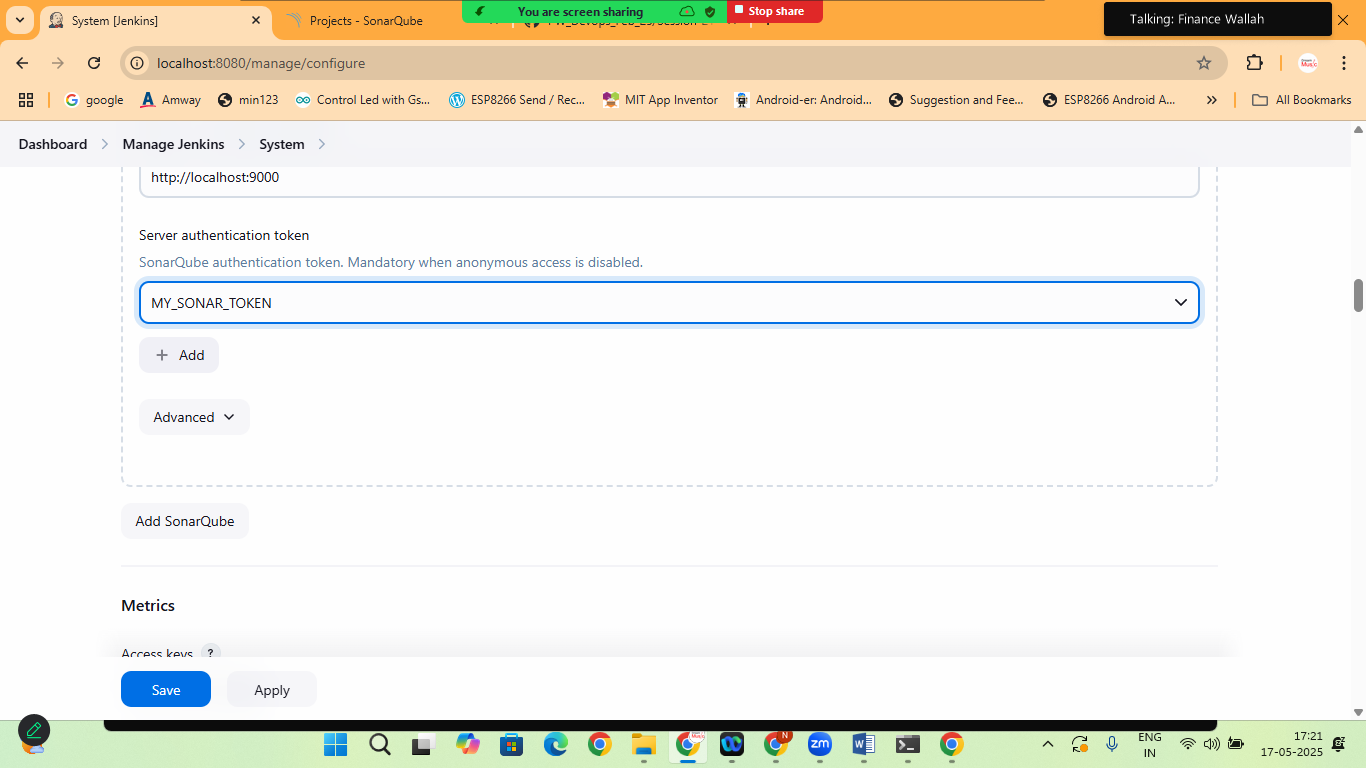
Goto SonarQube Servers>



Give the name : MySonar

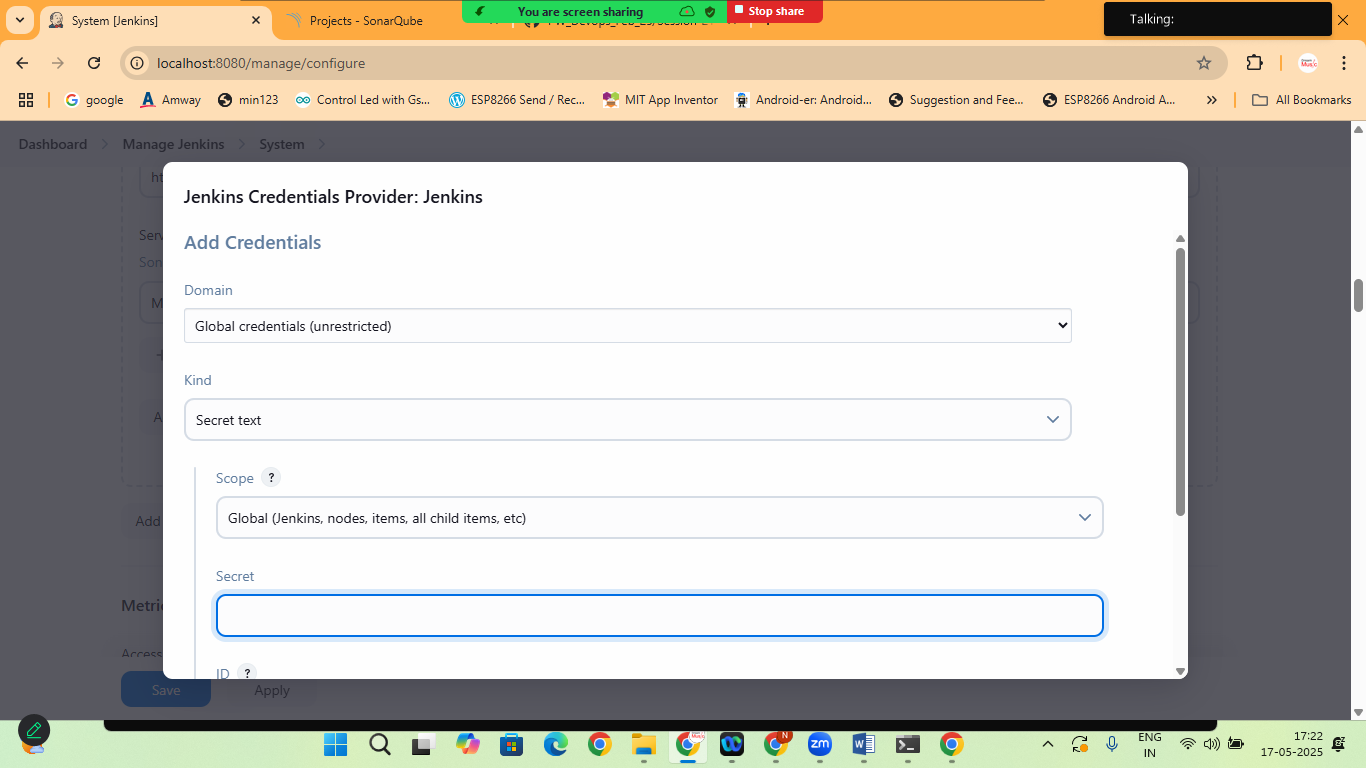
Server <URL:http://localhost:9000>

In server Authentication token > click on add

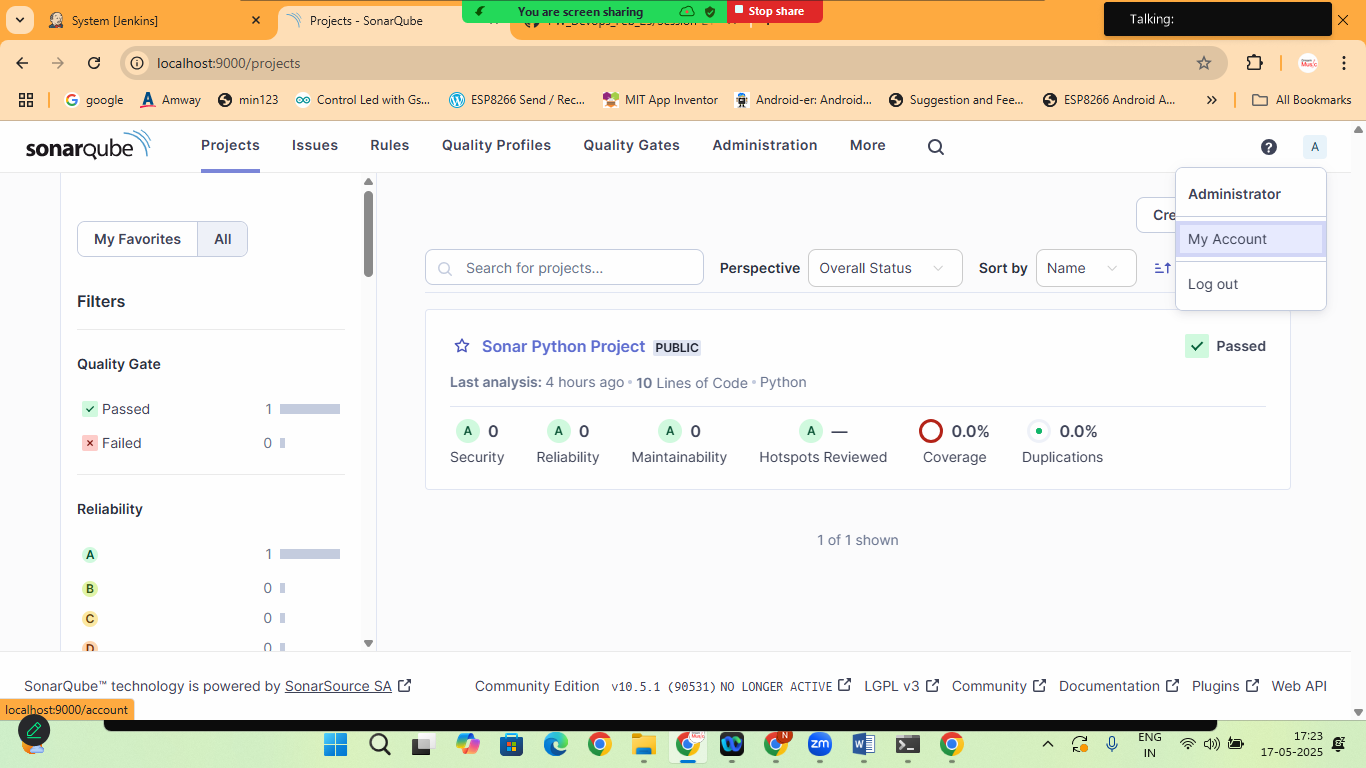


Choose secret text

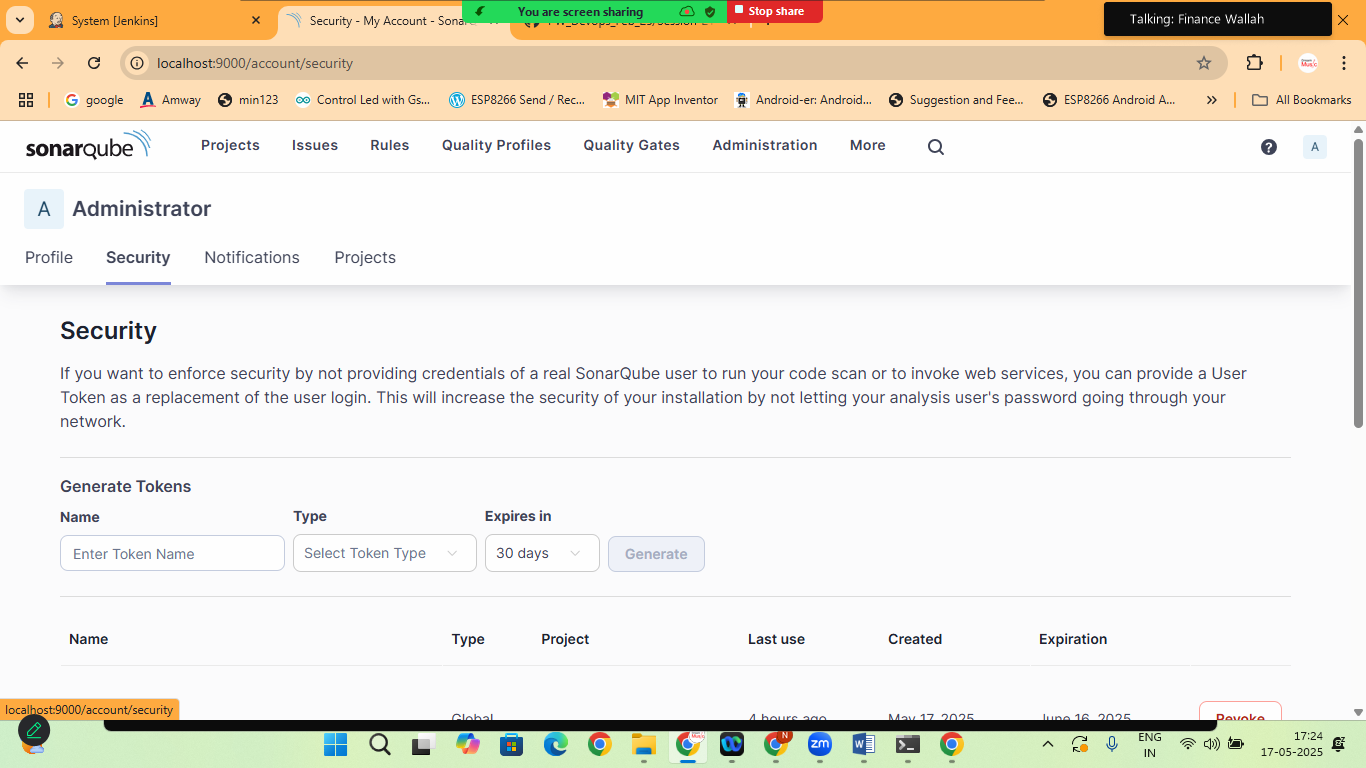
Add your authentication token in Secret filed:



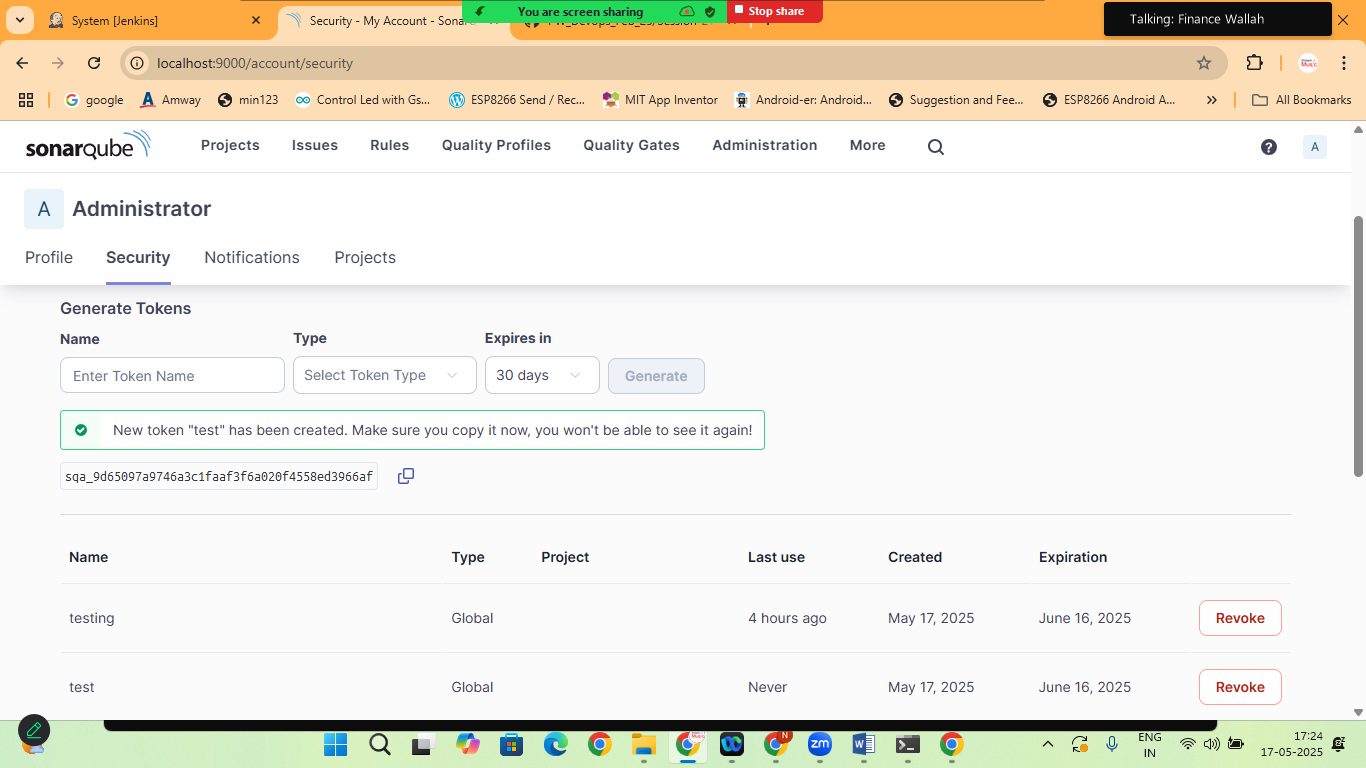
This token will be generated from sonarqube:> click on MyAccount



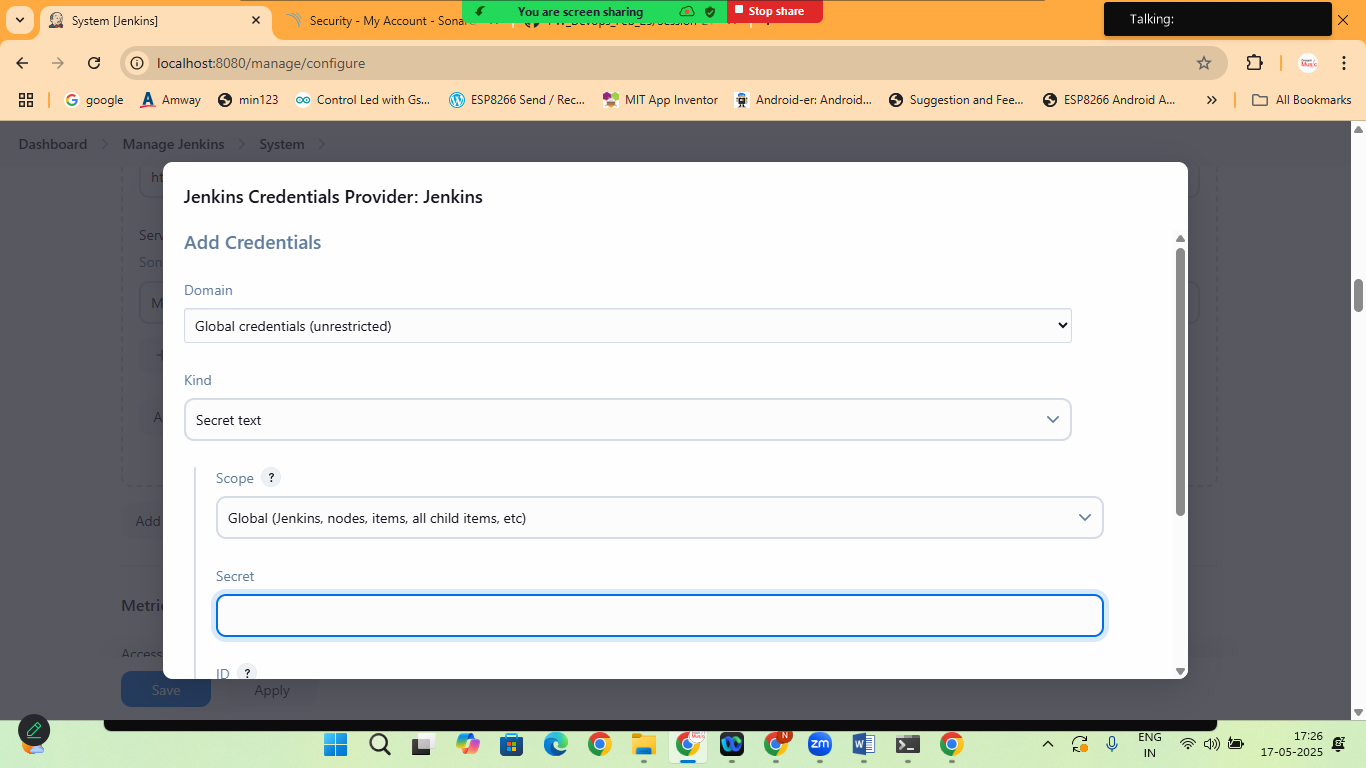
Click on security tab and create your token >



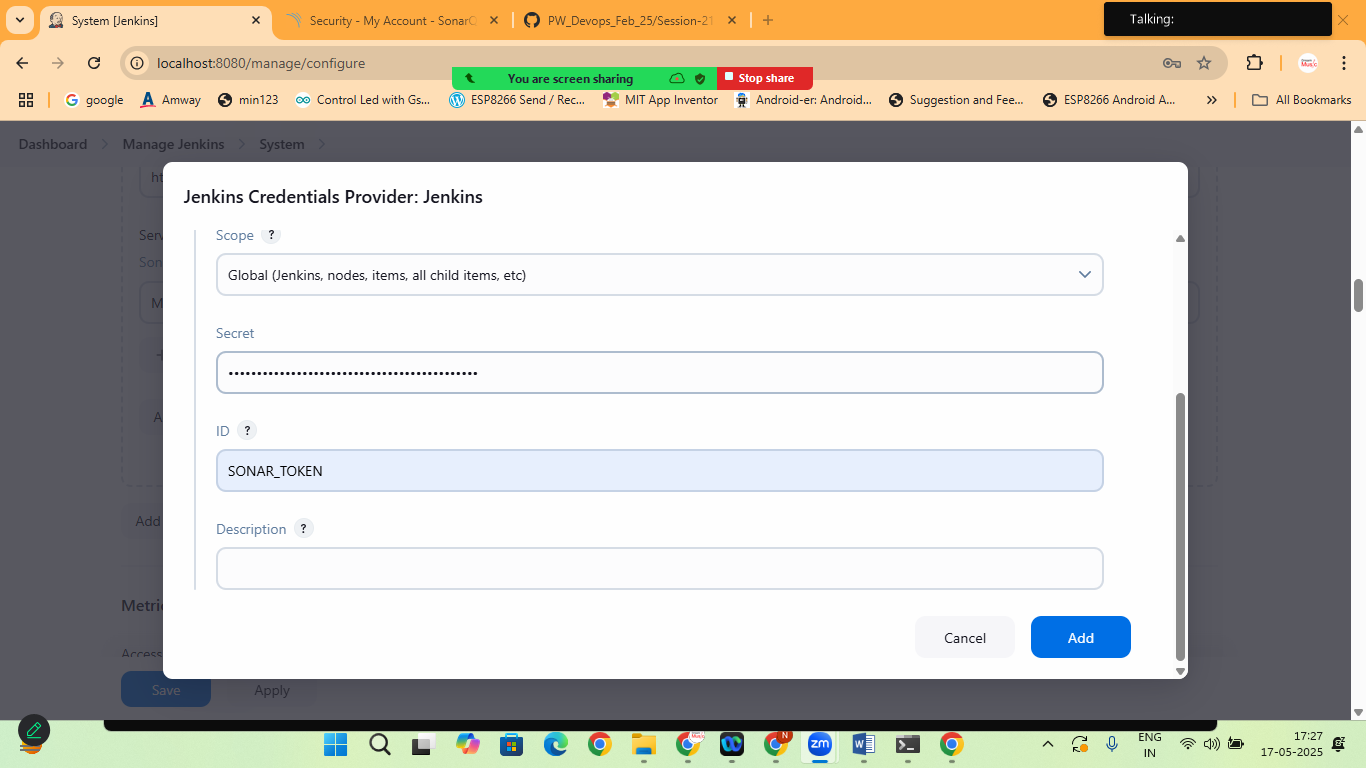
Genrate token >



Copy this token and past it to the Jenkins> secret text



Give the unique name: MY\_SONAR\_TOKEN



Click on Save and Apply

All set with integration

Create new pipeline> add the below pipeline

pipeline {

agent any

environment {

SONARQUBE\_ENV = 'MySonar' // Jenkins → Manage Jenkins → Configure System

SONAR\_TOKEN = credentials('MY\_SONAR\_TOKEN') // From Jenkins Credentials store

}

stages {

stage('Checkout') {

steps {

git url: 'https://github.com/Nikunj-Java/Sonar\_Python.git', branch: 'main'

}

}

stage('SonarQube Analysis') {

steps {

withSonarQubeEnv("${SONARQUBE\_ENV}") {

sh '''

/opt/sonar-scanner/bin/sonar-scanner \

-Dsonar.projectKey=Sonar\_Python \

-Dsonar.sources=app \

-Dsonat.tests=tests \

-Dsonar.python.version=3.10 \

-Dsonar.login=${MY\_SONAR\_TOKEN}

'''

}

}

}

stage('Quality Gate') {

steps {

timeout(time: 2, unit: 'MINUTES') {

waitForQualityGate abortPipeline: true

}

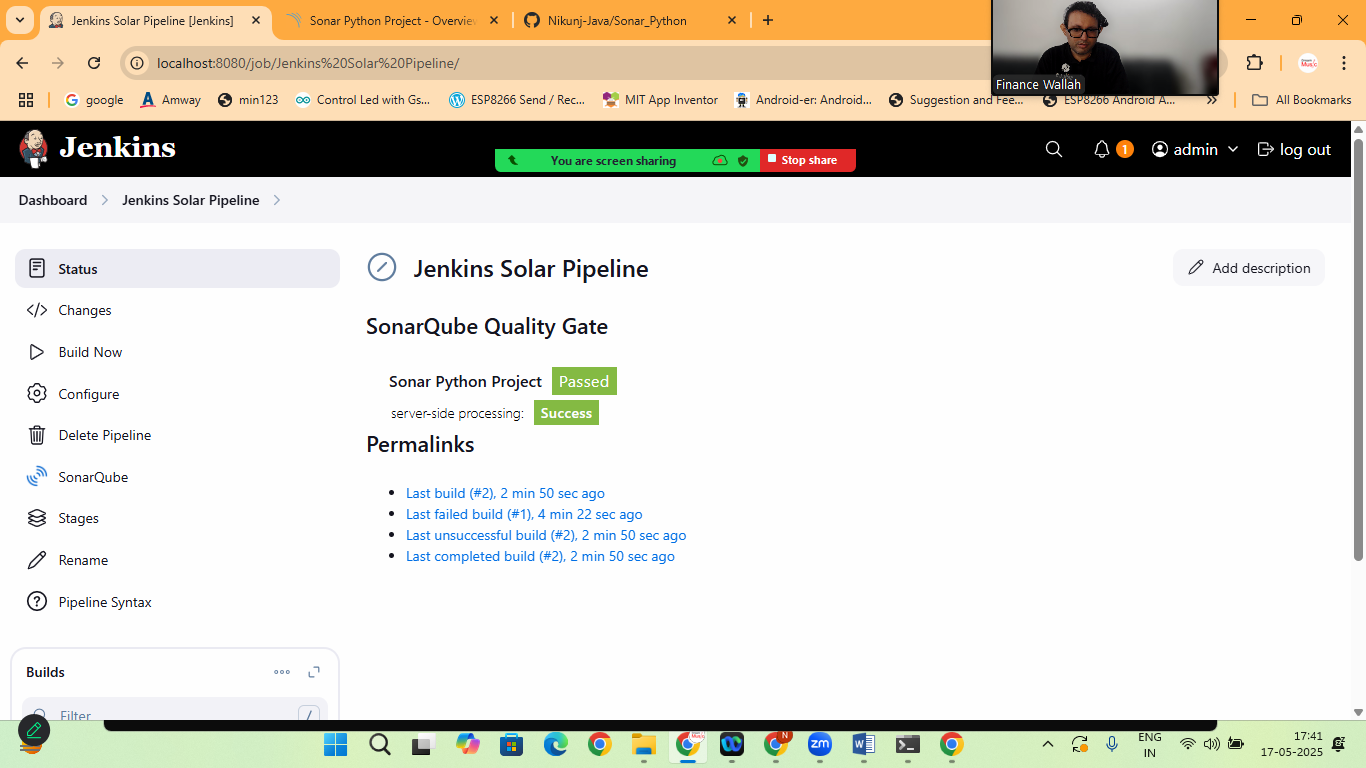
}

}

}

}

Build and configure



Check the output in sonar cube as well

