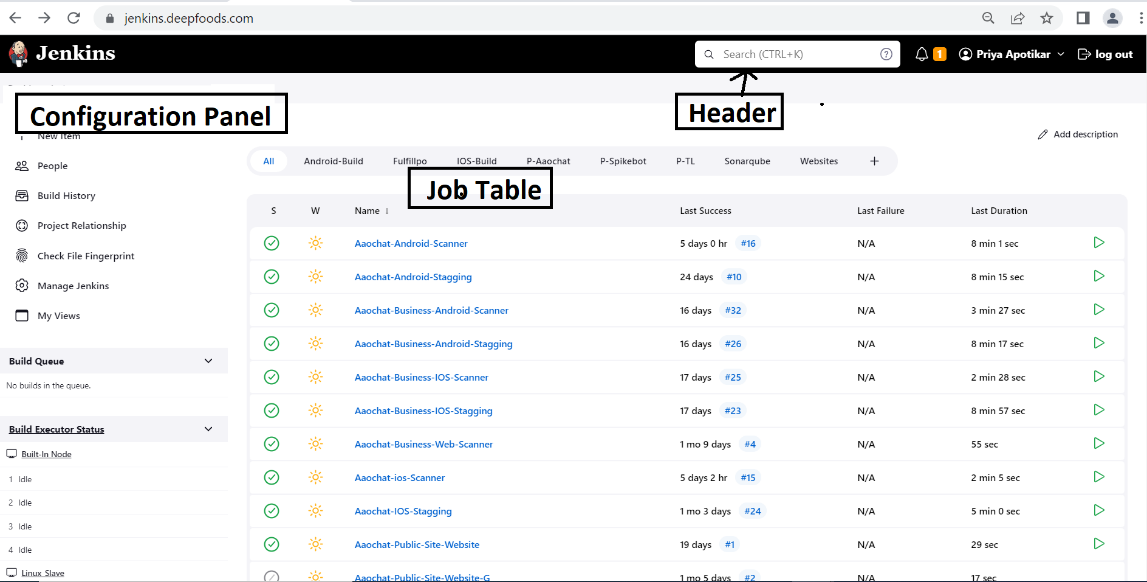
**Jenkins**

**What Is Jenkins**

Jenkins is an open-source solution comprising an automation server to enable continuous integration and continuous delivery (CI/CD), automating the various stages of software development such as build, test, and deployment.



**A Pictorial Representation of the Jenkins Interface**

**Jenkins Used For**

* Continuous Integration and Continuous Delivery
* Building and Testing Projects
* Automating Deployment & Execution of Scripts
* Pipeline as Code
* Supporting Various Plugins
* Reporting

Jenkins is aimed at enabling swift and safe delivery of software, from integrating and building to testing and deploying.

**How Does Jenkins Work?**

Jenkins may be operated as a server on various operating systems, including Windows, macOS, Unix versions, and, most notably, Linux. It also runs on the Oracle JRE or OpenJDK and requires a Java 8 virtual machine or higher. Jenkins is often executed as a Java servlet within a Jetty application server, and other Java application servers, such as Apache Tomcat, Nginx can be used to run it.

**What Is a Jenkins Pipeline**

Pipelines are needed to run Jenkins. A pipeline is a set of steps the Jenkins server will execute to complete the CI/CD process’s necessary tasks. In the context of Jenkins, a pipeline refers to a collection of jobs (or events) connected in a specific order.

Jenkins Pipelines comprise a powerful technology that includes a set of tools for hosting, monitoring, compiling, and testing code or code modifications across various tools such as:

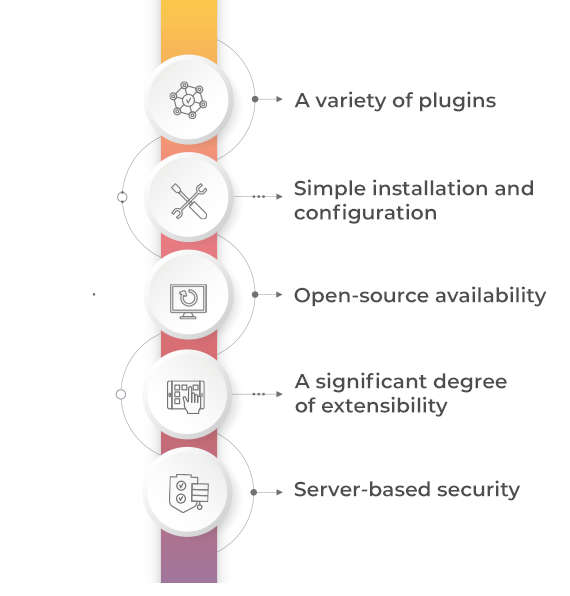
Continuous integration server: Jenkins

Source control software: GIT

Build tools: Gradle, Ant

Automation testing framework:

**Key Features of Jenkins**



### 1. A variety of plugins

Jenkins plugins are extensions to the Jenkins system. Providing integration points for [CI/CD tools,](https://www.spiceworks.com/tech/devops/articles/best-cicd-tools/)sources and destinations is among the most prevalent plugin applications.

### 2. Simple installation and configuration

Jenkins is a self-contained Java software that doesn’t care about the platform it’s running on. It runs on all standard operating systems, including Windows, Unix versions, and Mac OS. Jenkins’ online interface is simple to set up and configure, including error checks and a built-in help feature.

### 3. Open-source availability

Jenkins is an open-source project that is entirely free to use. Jenkins has a vibrant development community that meets both in-person and online regularly. This simplifies installation while restricting resources to a single computer, virtual machine, or container.

### 4. A significant degree of extensibility

Jenkins’ plugin design allows it to be expanded in almost any way, giving it practically limitless capabilities. Jenkins’ community is its backbone, and members have played a vital role in the development (and testing).

**5. Server-based security**

Jenkins security is concerned with both the server and the user. The server, whether a virtual machine or a bare metal server, is designed to enable the fewest number of processes to interface with it. This is achieved using a standard server operating system and networking security capabilities. Furthermore, using common mechanisms such as multifactor authentication, access to the server via the Jenkins UI is limited to the fewest number of people possible.

Jenkins’ internal user database has security features as well. The Jenkins web UI is used to access these capabilities. The “Security Realm” and the “Authorization Realm” are the two security realms that Jenkins supports.

**Jenkins Master**

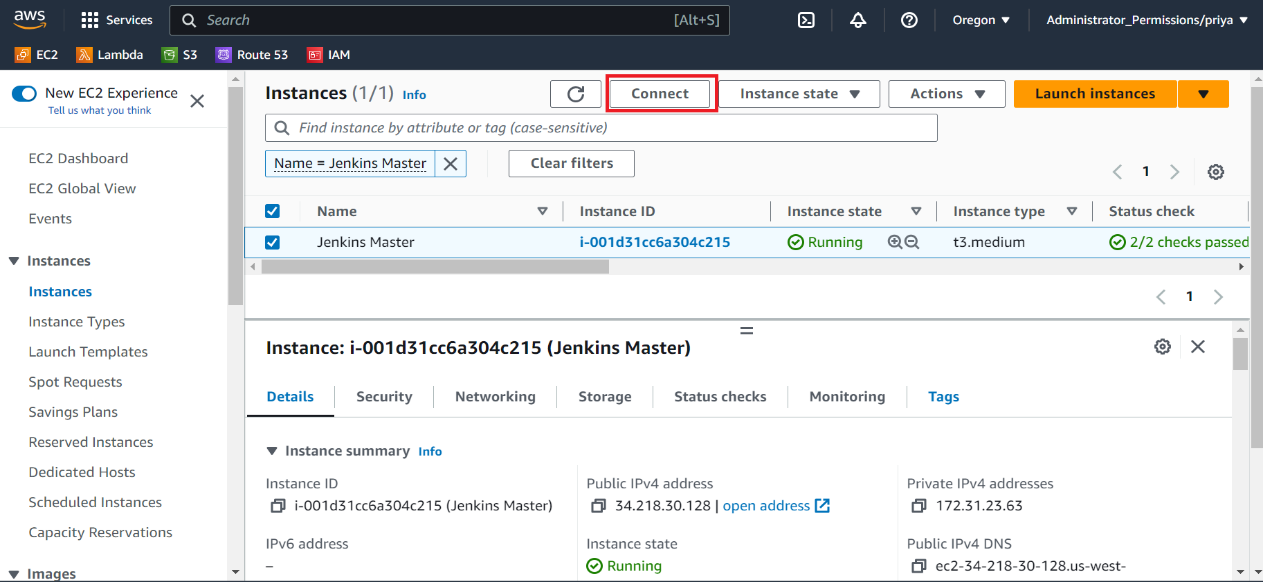
**Procedure:**

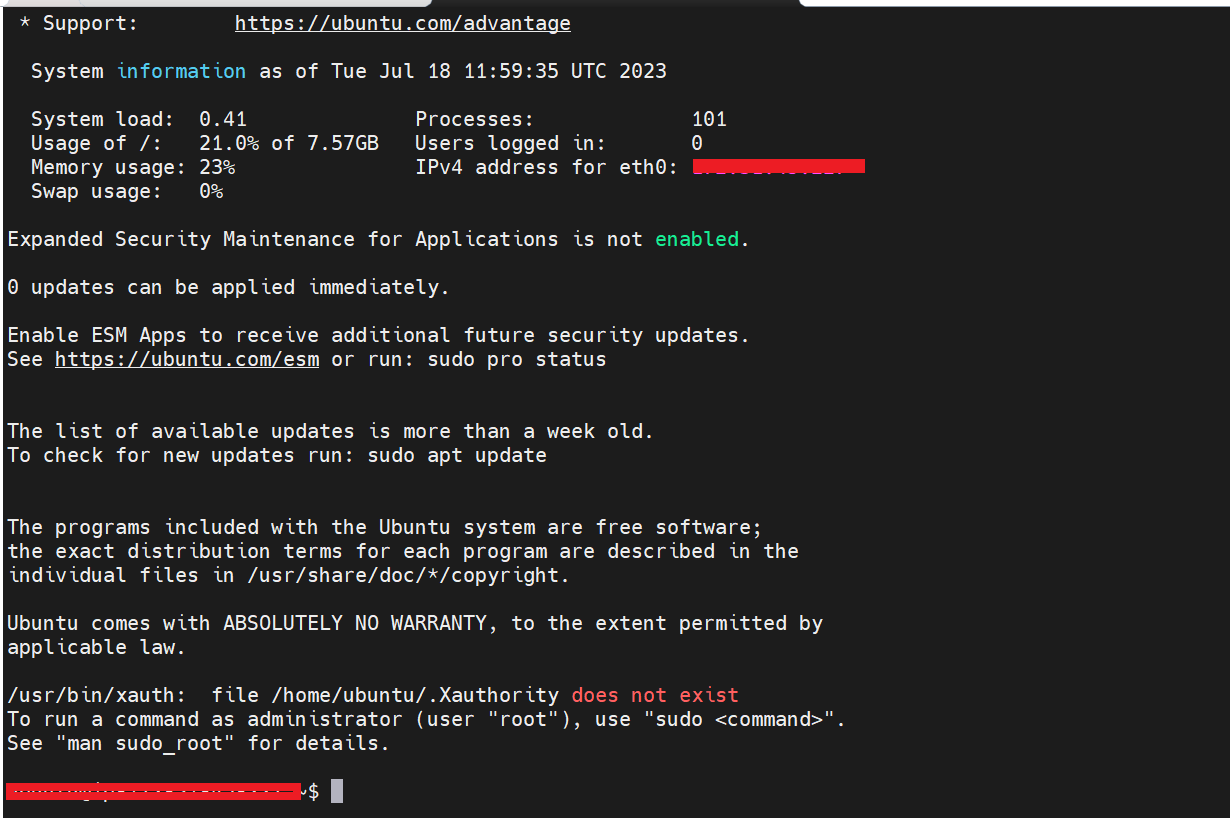
* **Login to Your Account, and Launch an EC2 Instance**

Login into your AWS account and type “**EC2**” in the search box. Click on “**EC2**” and select the “**Ubuntu**” image.

Give a name for your instance Jenkins Master Next, create a security group and make sure to have **port 22,** **port 80** and **port 8080** open so that the instance can “**send and receive**” traffic to the internet. Leave the rest of the default settings as is and **CLICK LAUNCH TO DEPLOY YOUR UBUNTU INSTANCE!**

You have the option to connect to your instance remotely via SSH or through “**instance connect**” within the instance console.

****



**Steps to Setup Jenkins Master serve**r:

* Update Your System

sudo apt update

* Install Java

Jenkins requires Java to run., you can install the OpenJDK package:

sudo apt install openjdk-11-jdk

* Check the Java Version:

java --version

* Add the repository key to the system:

wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add -

* Append the Debian package repository address to the server’s sources.list:

sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ > /etc/apt/sources.list.d/jenkins.list'

* Update the repository:

sudo apt update

curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee /usr/share/keyrings/jenkins-keyring.asc

echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] https://pkg.jenkins.io/debian-stable binary/ | sudo tee /etc/apt/sources.list.d/jenkins.list

* Install Jenkins:

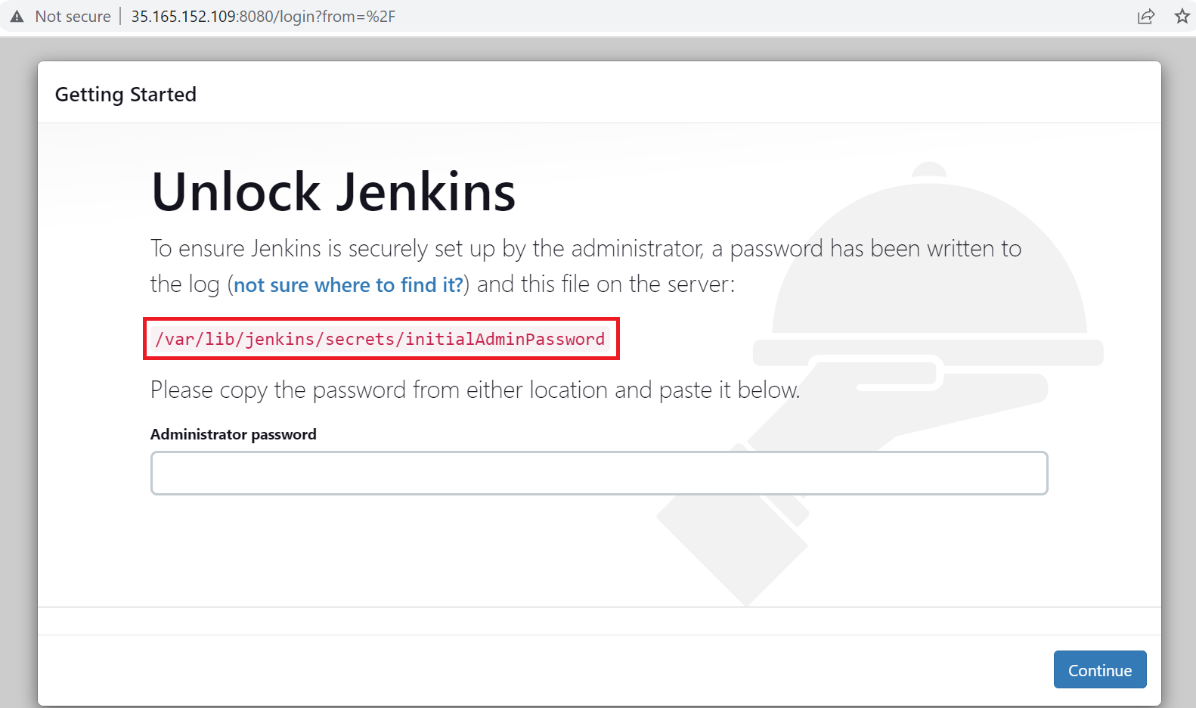
sudo apt install Jenkins

* Start Jenkins:

sudo systemctl start jenkins

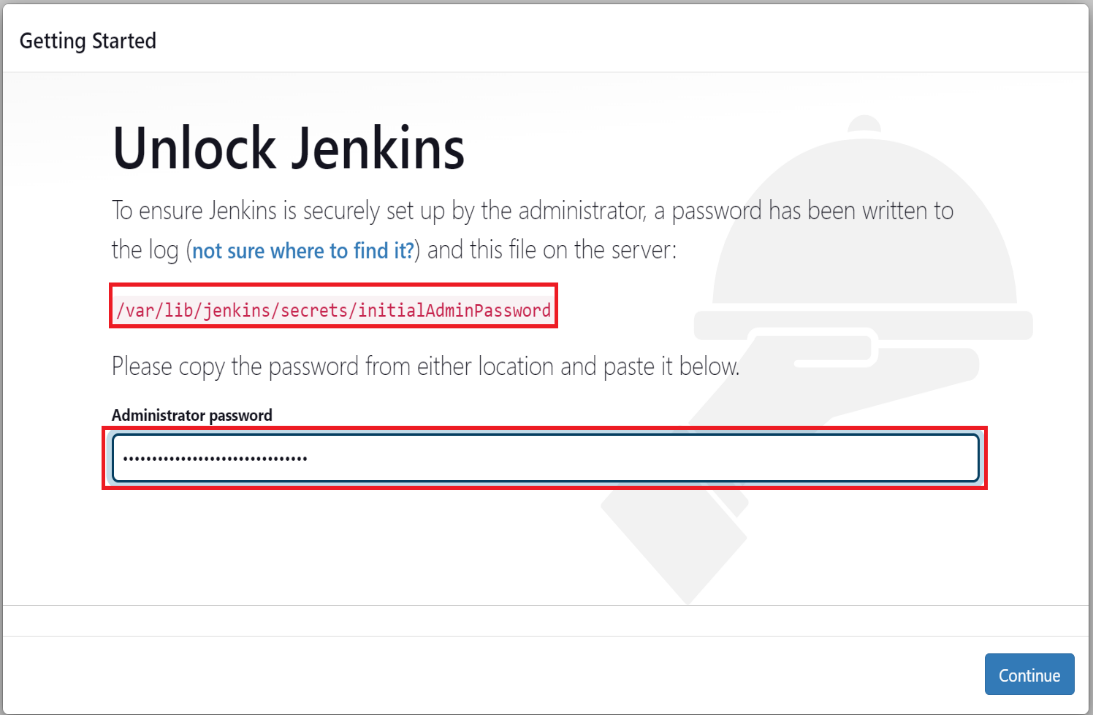
* Enable Jenkins:

sudo systemctl enable jenkins



* Setting up Jenkins

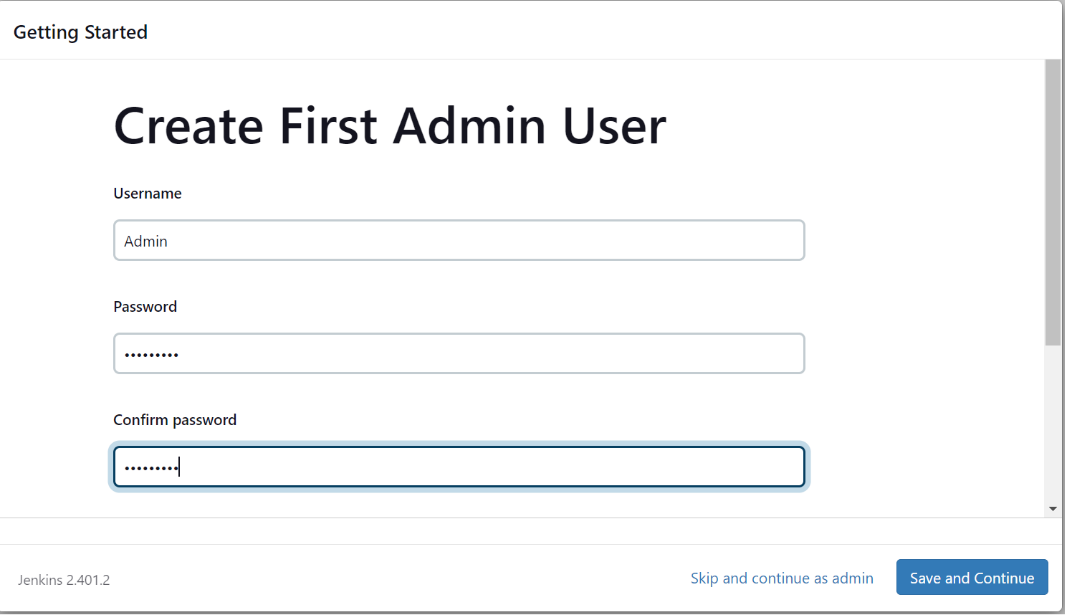
cat /var/lib/jenkins/secrets/initialAdminPassword

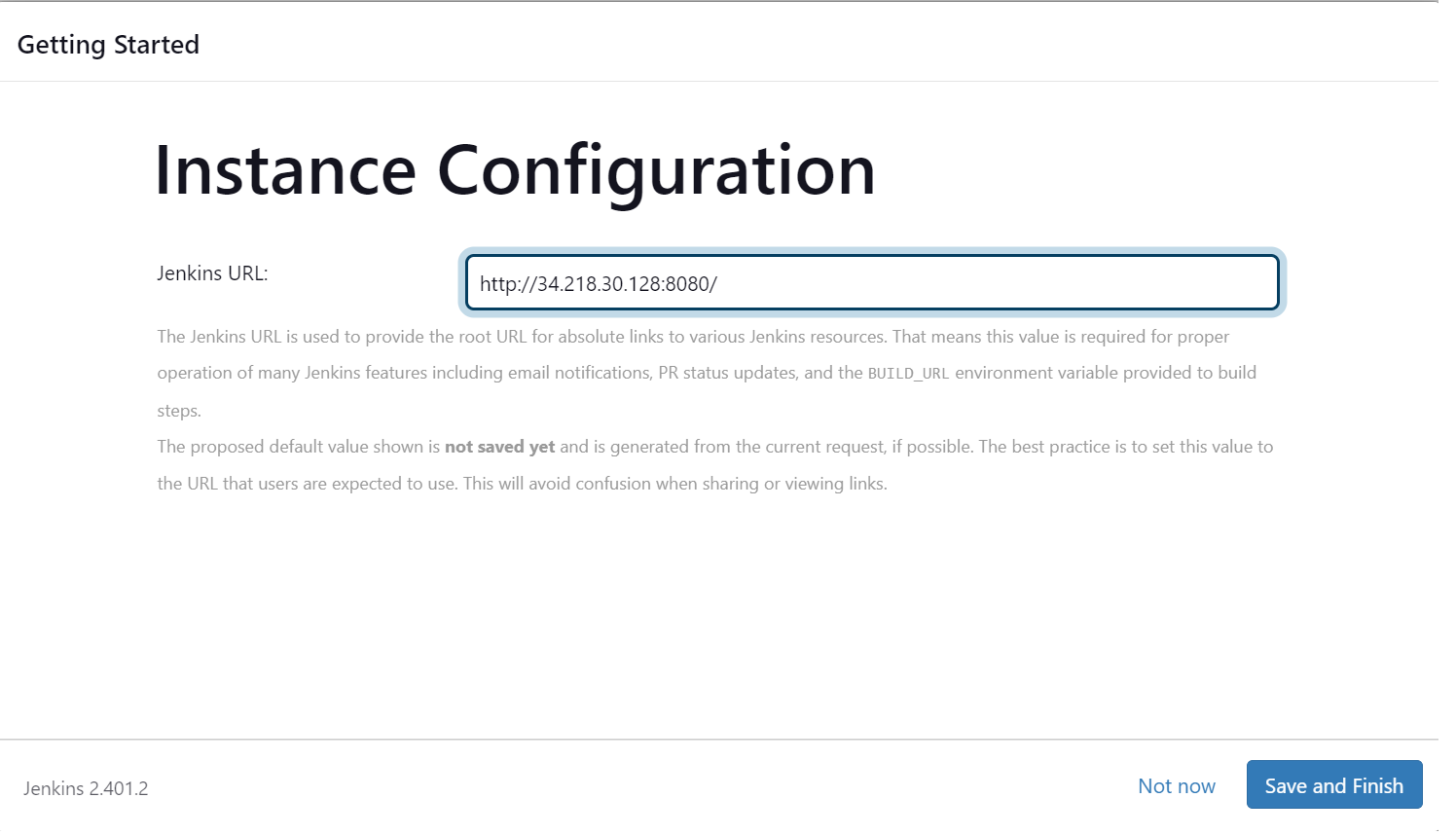


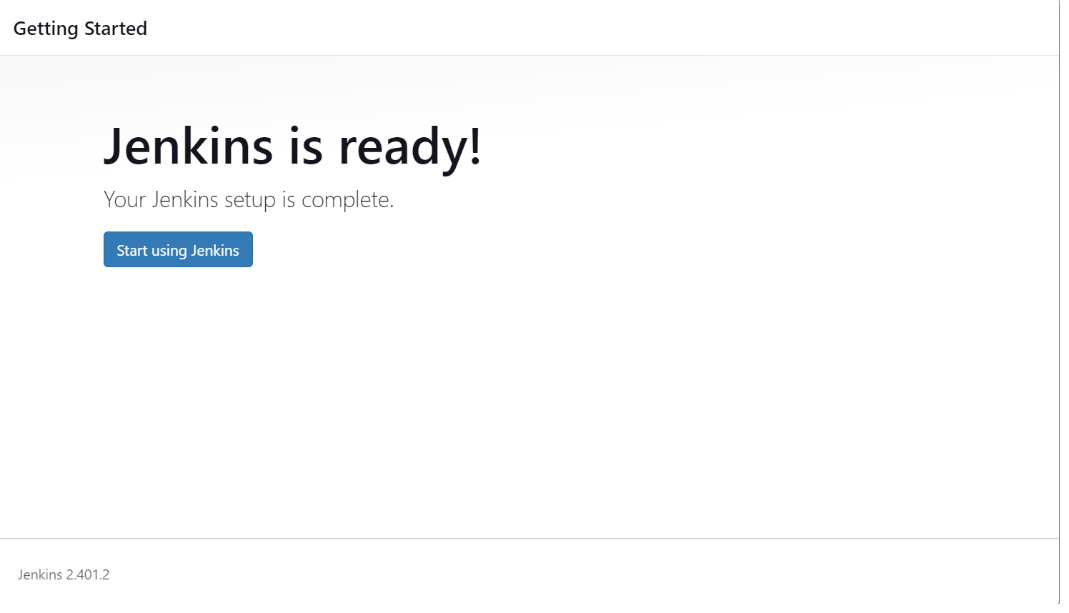
* Customize jenkins

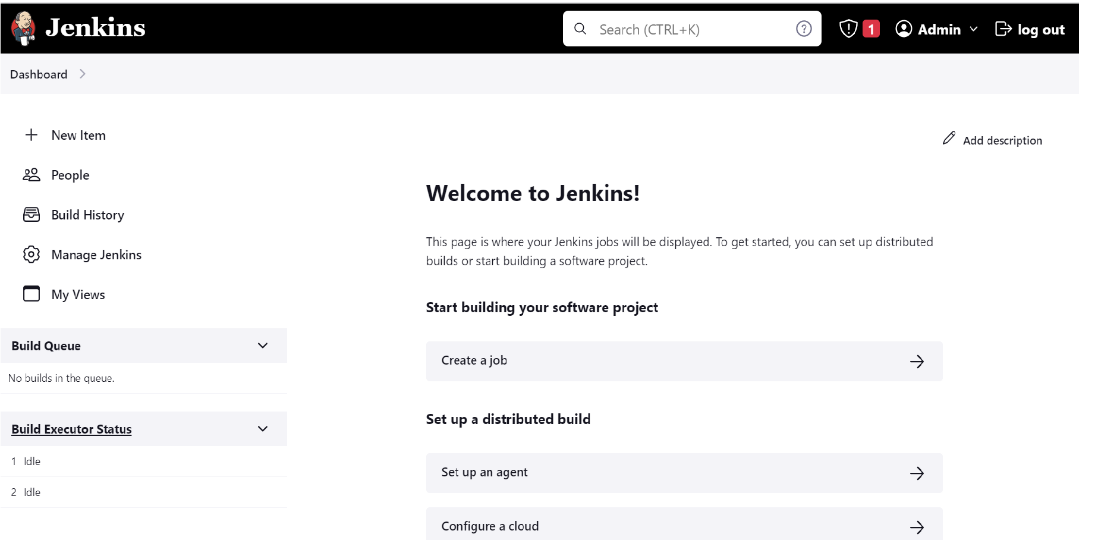
install suggested plugins



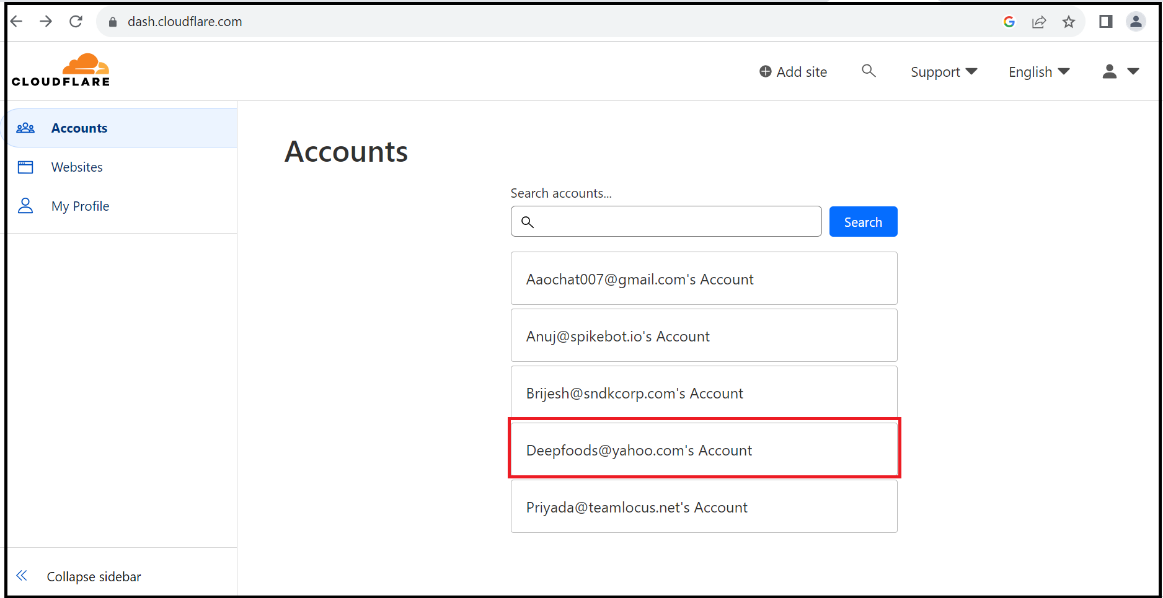




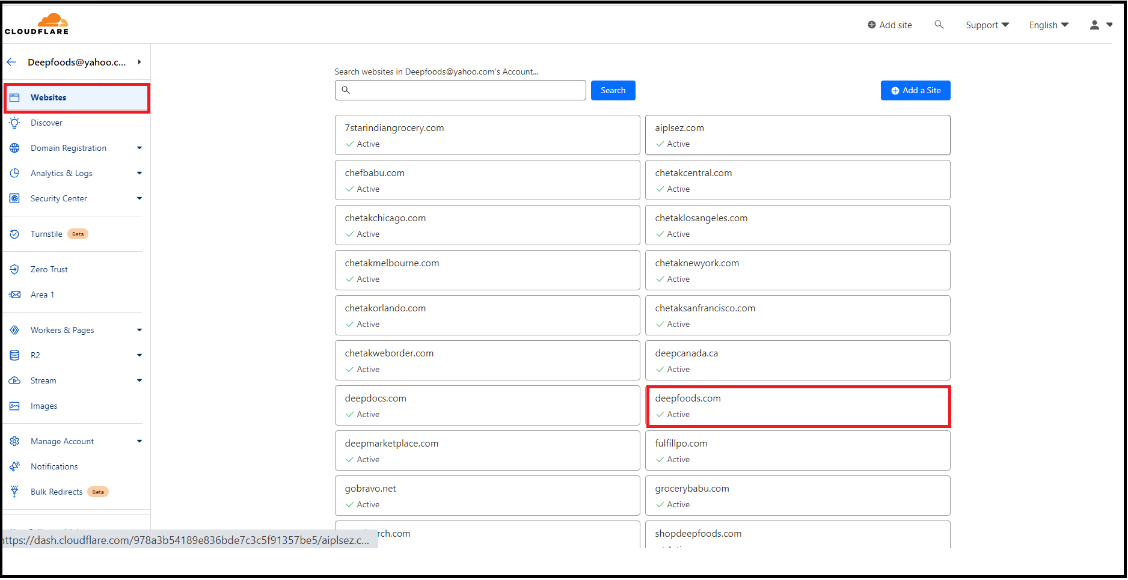




* Add domain in CLOUDFLARE
* Select [Deepfoods@yahoo.com’s](mailto:Deepfoods@yahoo.com’s) Account



* Navigate to the Cloudflare website and 'Log In' Deepfoods@yahoo.com account.
* Once you are logged in, click on the '+Add a Site' or ‘Select the site’ at the top of the page.

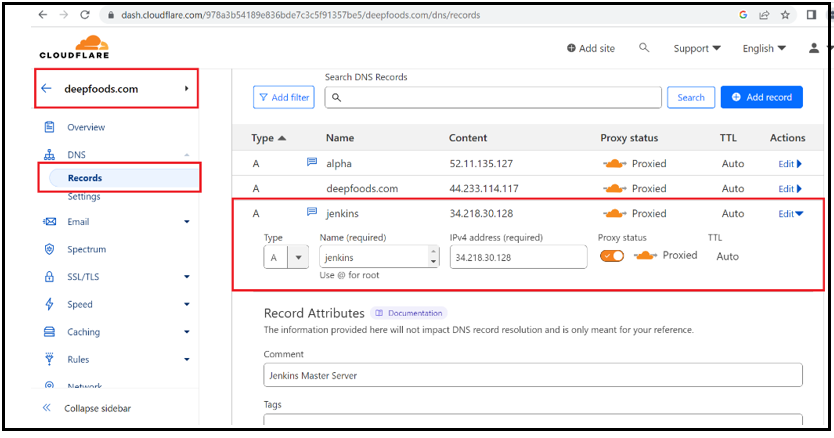


* Scan DNS Records

Cloudflare will automatically scan the DNS records of your domain. This process can take up to a minute.

After the scan, Cloudflare will display the DNS records they found. Check them carefully to ensure they match the records you have. If they don’t, manually add them. Click 'Continue' when you are done.

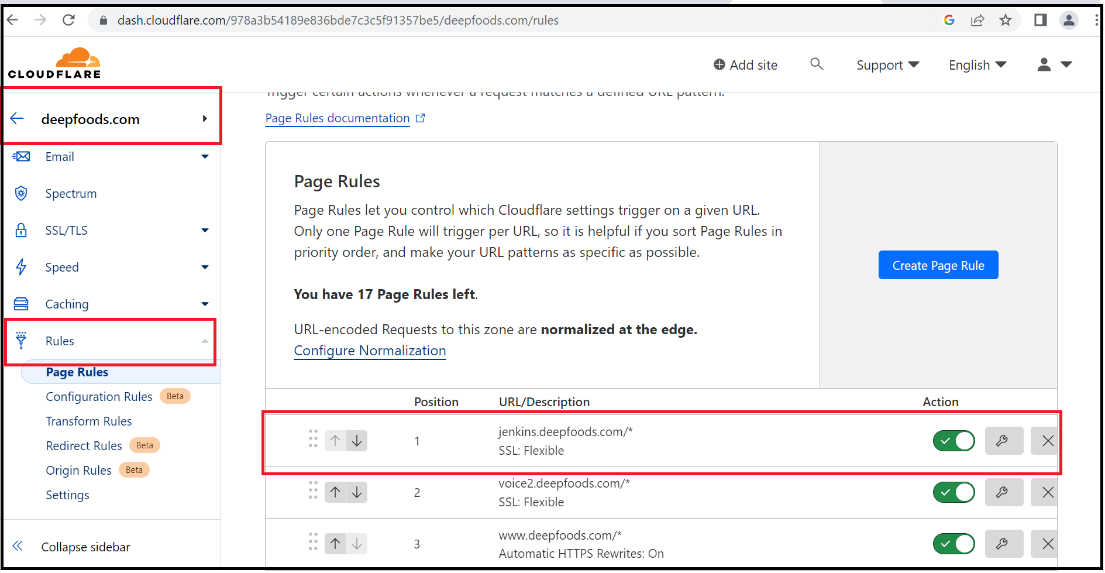
* Add records for Jenkins



* Add Rules

Jenkins.deepfoods.com/\*

SSL Flexible



**Create Jobs and Views**

Creating views help you to organize your jobs.

**Crons:**

All jobs will get automatically to default view, when you create number of jobs its hard to maintain.

**Best Practice**:

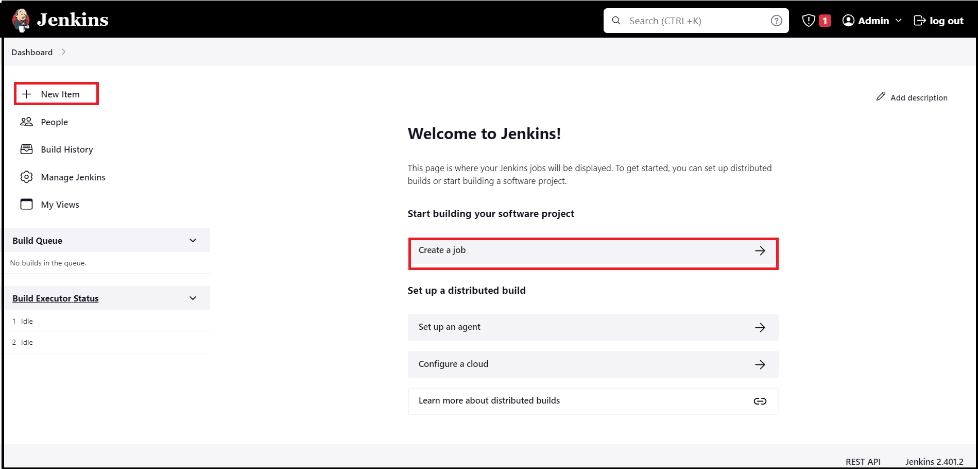
Create view per application and pull respective jobs to that view for better maintainability.

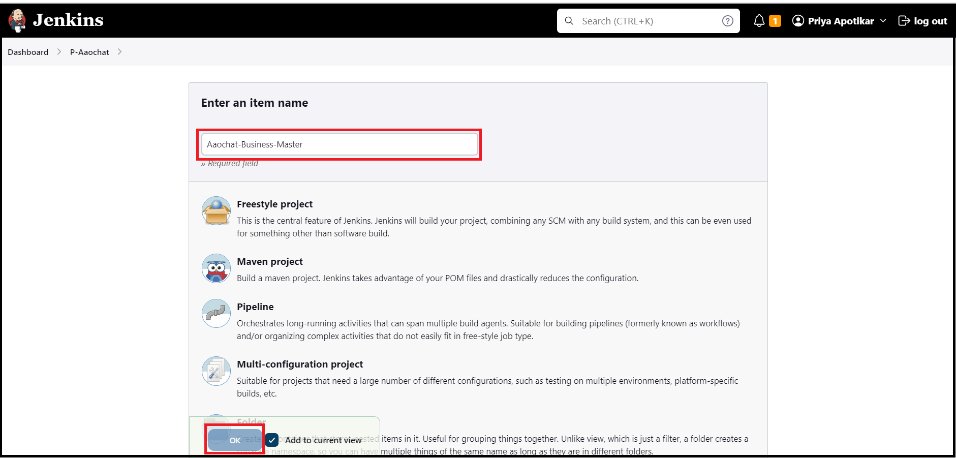
**Prerequisites:**

Jenkins Installed

**Steps:**

create view and add jobs

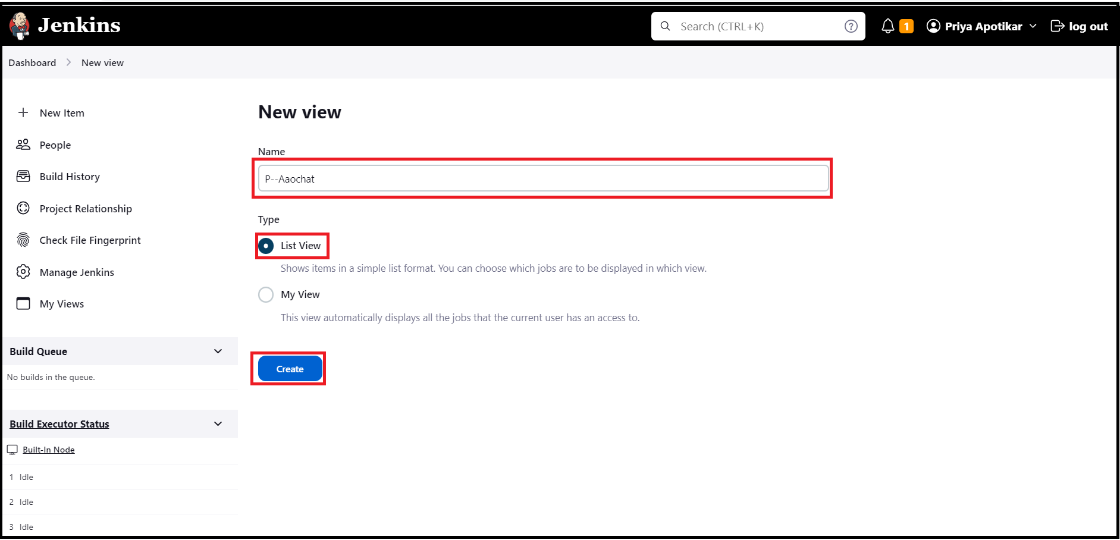




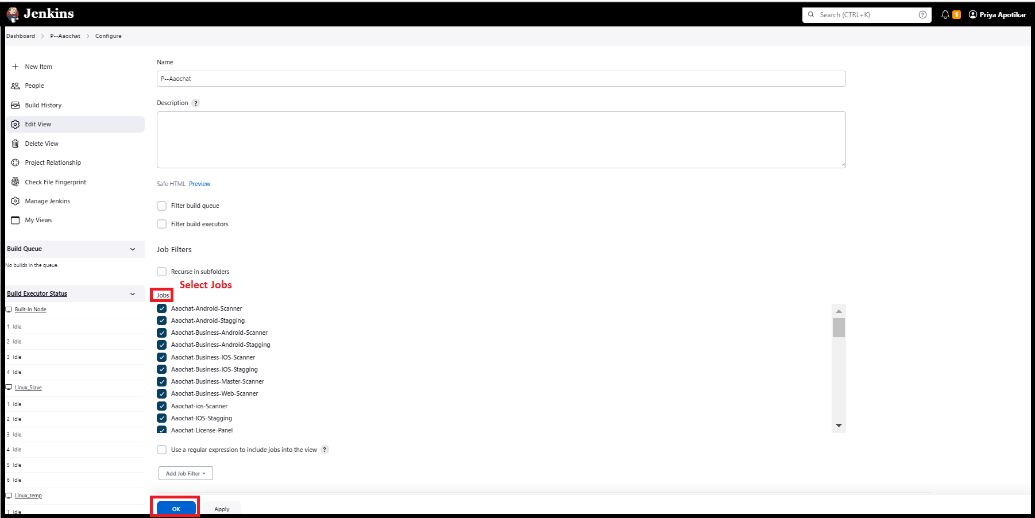
* Click on "New View
* Click this "+" tab.



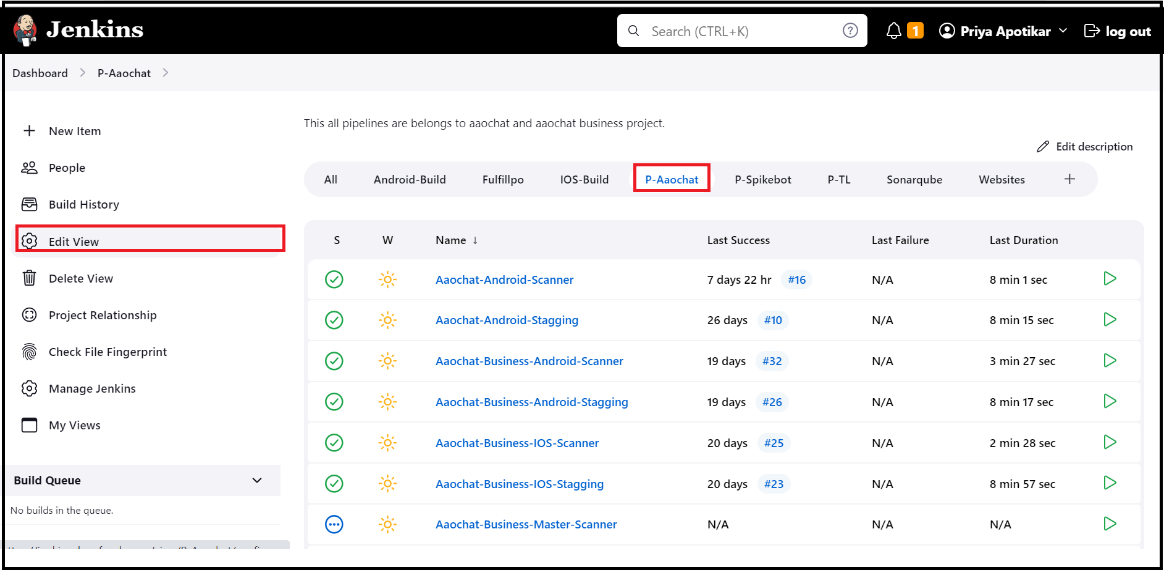
* Enter View Name and Select View Type
* You will be directed to a new page. where need to enter details for the new view.
* In the "Name" field, specify a unique name for the view.
* Below the "Name" field, you'll have to select the "View type". By default, Jenkins offers two types of views - "List View", "My View". Select the one you prefer.
* Click "create" to proceed.



* Configure the View
* Now you'll see a configuration page where you can design your view according to your preferences.
* Under the "Job" section, you can configure which jobs to include in the view.
* Save the View
* Once all the configurations are done, click "Apply" and then "OK" to save the view.
* Now, when you go back to your Jenkins dashboard, you'll see the view you just created in the top menu. Click on it to see the jobs you've included



* You add jobs to a Jenkins view during the view's creation or modifying an existing view.
* On the Jenkins dashboard, click on the view you want to edit, then click on "Edit View" on the left side of the page.



* On the view configuration page, look for the "Job" section.
* Once you've added all the jobs you want, scroll to the bottom of the page and click "Apply", then "OK".

