**JENKINS**

What is Jenkins?

Jenkins is an opensource automation server written process via Continuous integration and facilitates Continuous delivery.

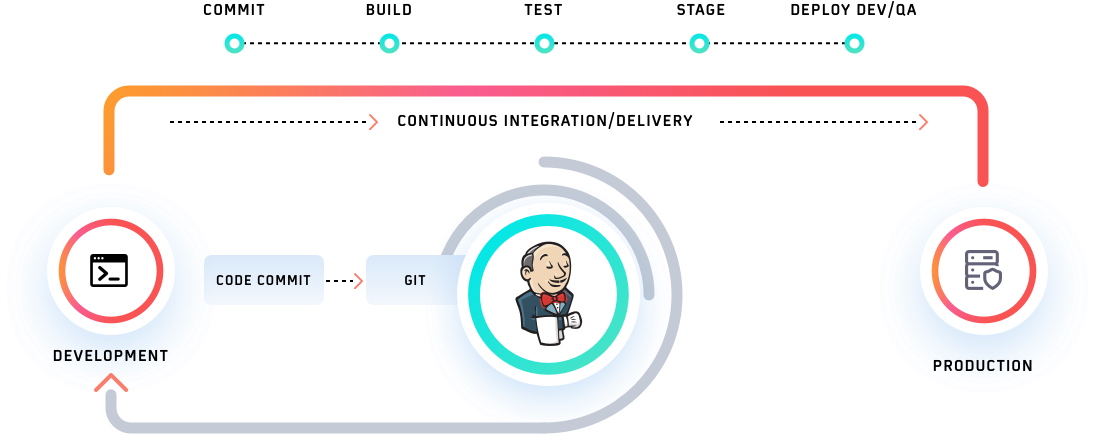
Jenkins is now become that open-source project it is one of the most popular and continuously contributed projects that is available as open source. So , we are always getting new features being added to it.Tool that really becomes the centre for the continuous integration environment.

Why Jenkins?

Jenkins really comprised around five:

* Easy Installation—Jenkins is a self contained Java program and that allows it to run on most popular operating system includes windows,mac os etc.
* Easy configuration--It can be easily set-up and configured via its web interface which includes error checks and built-in help.
* Plugins—It has hundreds of plugins in the update centre and integrates with every tool in the continuous integration and continuous deployment toolchain.
* Extensible-Jenkins can be extended via its plugin architecture nearly infinite possibilities for what it can do.
* Distributed—It can easily distribute work across multiple machine helping in faster builds, tests and deployments across multiple platforms

Standard Jenkins pipeline:



When we are doing development, we start off and we are coding away on our computer. The first thing we have to do when we are working in Jenkins pipeline is to actually commit our code.

As a developer, this is something that we are ready doing at least we are committing our code to a git server .In this instance we are using Jenkins as the place for us to commit our code.

Jenkins will create a build of our code and part of that build process is actually going through and running tests and again as a developer we are already comfortable with running a unit test and writing those test to validate our code but there may be additional tests that Jenkins is running, so, for a instance as a team we may have standard sets of code that’s been written and those tests can be also included in the testing process within the Jenkins environment. Assuming that everything passed that the tests can get everything placed in a stage and release ready environment within Jenkins and finally again ready to deploy and deliver our code to a production environment.

Jenkins is going to be tool that helps us with our server environment to be able to deploy our code to the production environment and the result is that we are able to move from a developer to production code and really quickly this whole process can be automated rather than having to wait for people to actually test our codes .

Pipeline syntax:--Declarative, scripted

Declarative: are designed to be simpler and more structured. They use a predefined set of syntax to define the pipeline, making it easier to read and understand.

We used declarative here,

Pipeline{ ------------------------------------------Top-level

Agent any------------------------------------Where to build

Stages{---------------------------------------where work happens

Stage(“build”){-----------------------what to do

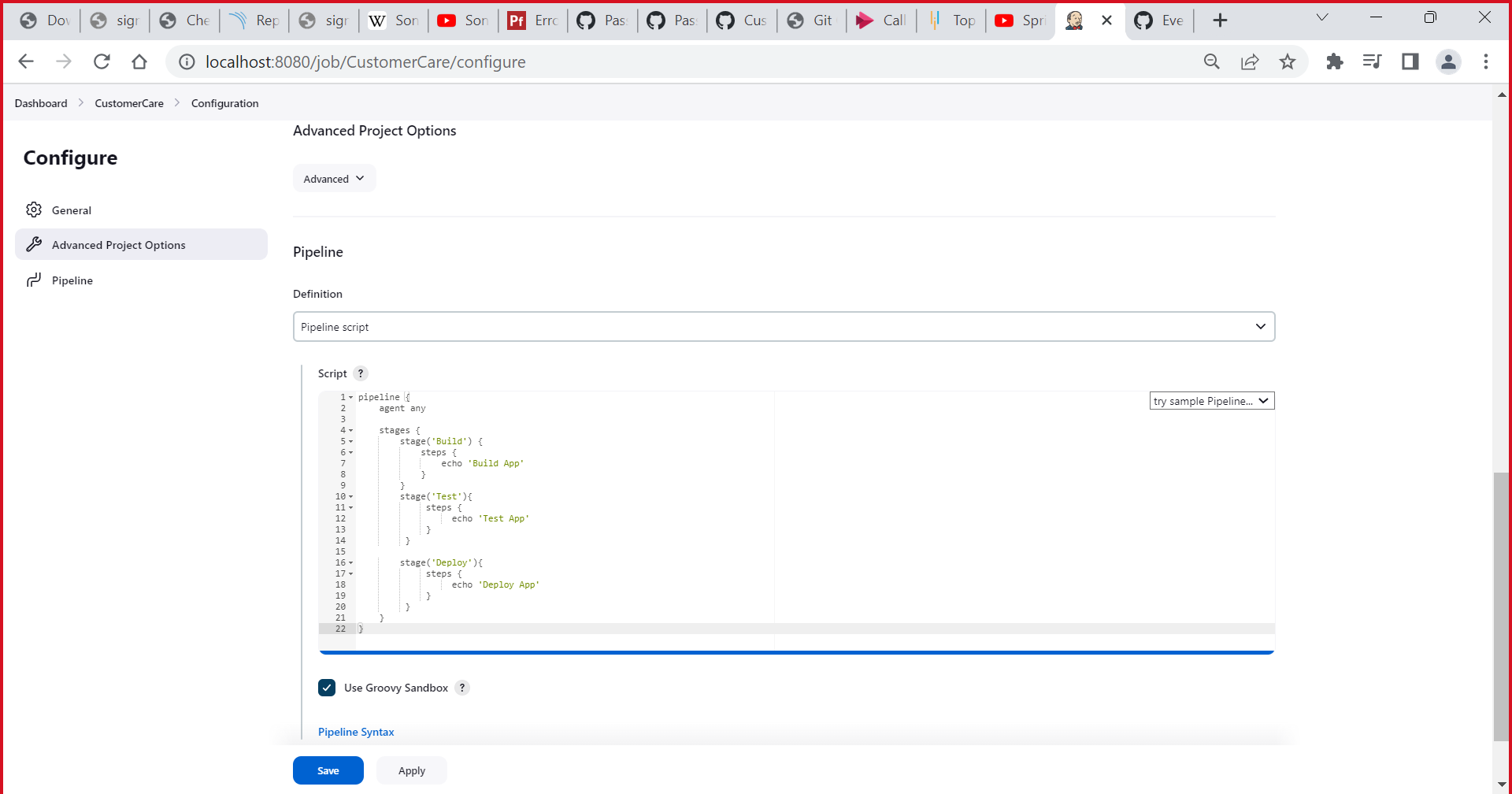
Steps {

}

}

}

}



As per the syntax we written a code in the pipeline.

Pipeline Architecture:

Jenkins pipeline architecture consists of several components, including:

1. Jenkins server: Jenkins is a server-based tool that manages your pipeline jobs, build agents, and other configuration details. It provides a web-based interface for users to configure and run pipeline jobs.
2. Pipeline job: A pipeline job is the basic building block of Jenkins pipeline. It defines the steps that Jenkins will take to build, test, and deploy your application.
3. Pipeline script: A pipeline script is a set of instructions that define the steps of the pipeline job. It can be written in Groovy or another scripting language.
4. Jenkinsfile: A Jenkinsfile is a text file that contains the pipeline script. It is typically stored alongside your application code in version control.
5. Pipeline stage: A pipeline stage represents a logical step in the pipeline. Each stage contains one or more steps that Jenkins will execute.
6. Pipeline step: A pipeline step is a specific task that Jenkins will execute during a stage. For example, a step might run a unit test, build the application, or deploy it to a server.
7. Agent: An agent is a worker node that performs the actual work of the pipeline job. It can be a physical machine or a virtual machine, and it can be located on-premise or in the cloud.
8. Pipeline view: A pipeline view is a visual representation of the pipeline job. It displays the current status of each stage and provides detailed information about any failures or issues.

Overall, the Jenkins pipeline architecture is designed to be flexible and scalable, allowing teams to customize their pipeline jobs to meet their specific needs.

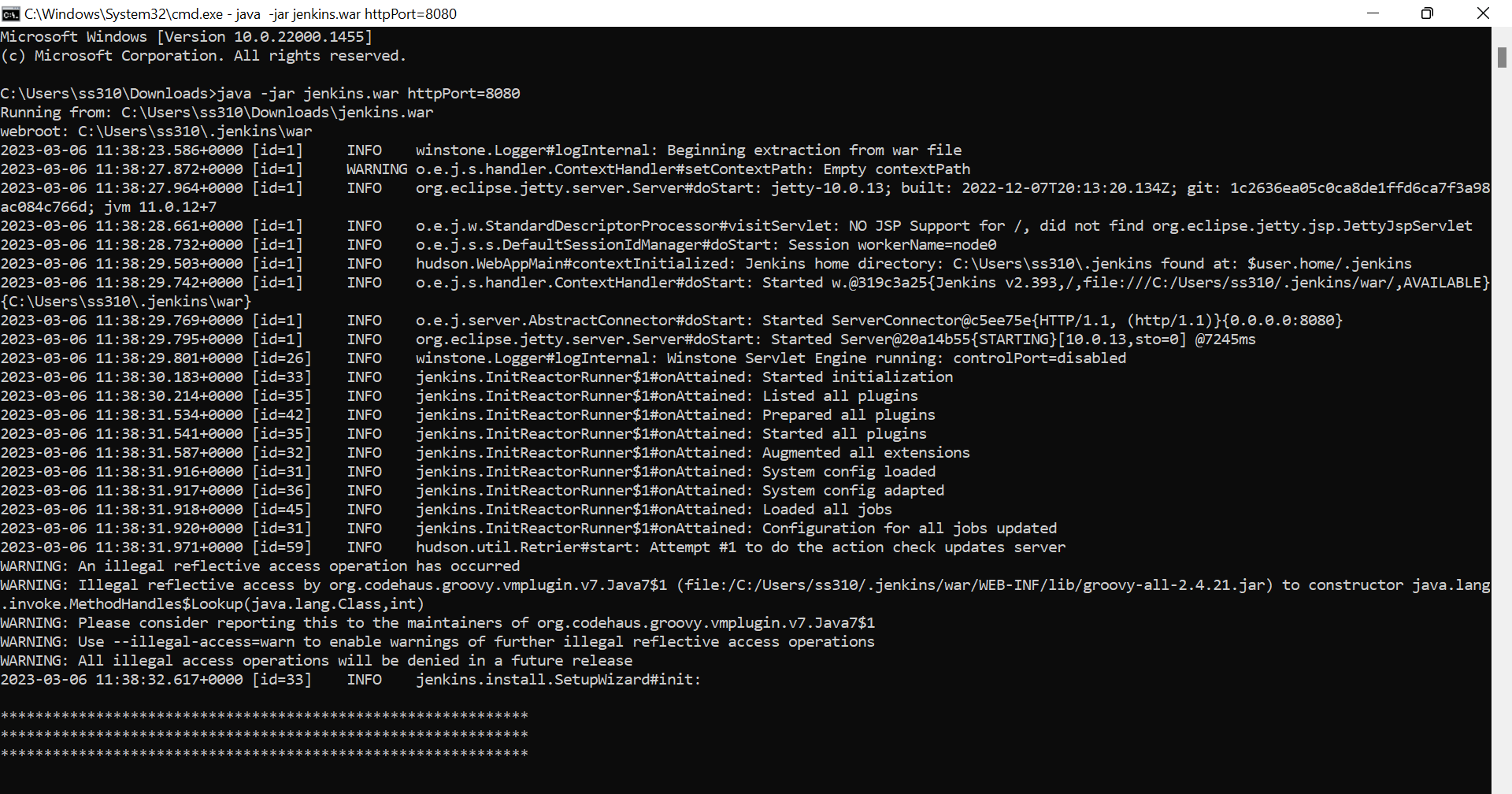
Step 1: java -jar jenkins.war --httpPort=8080

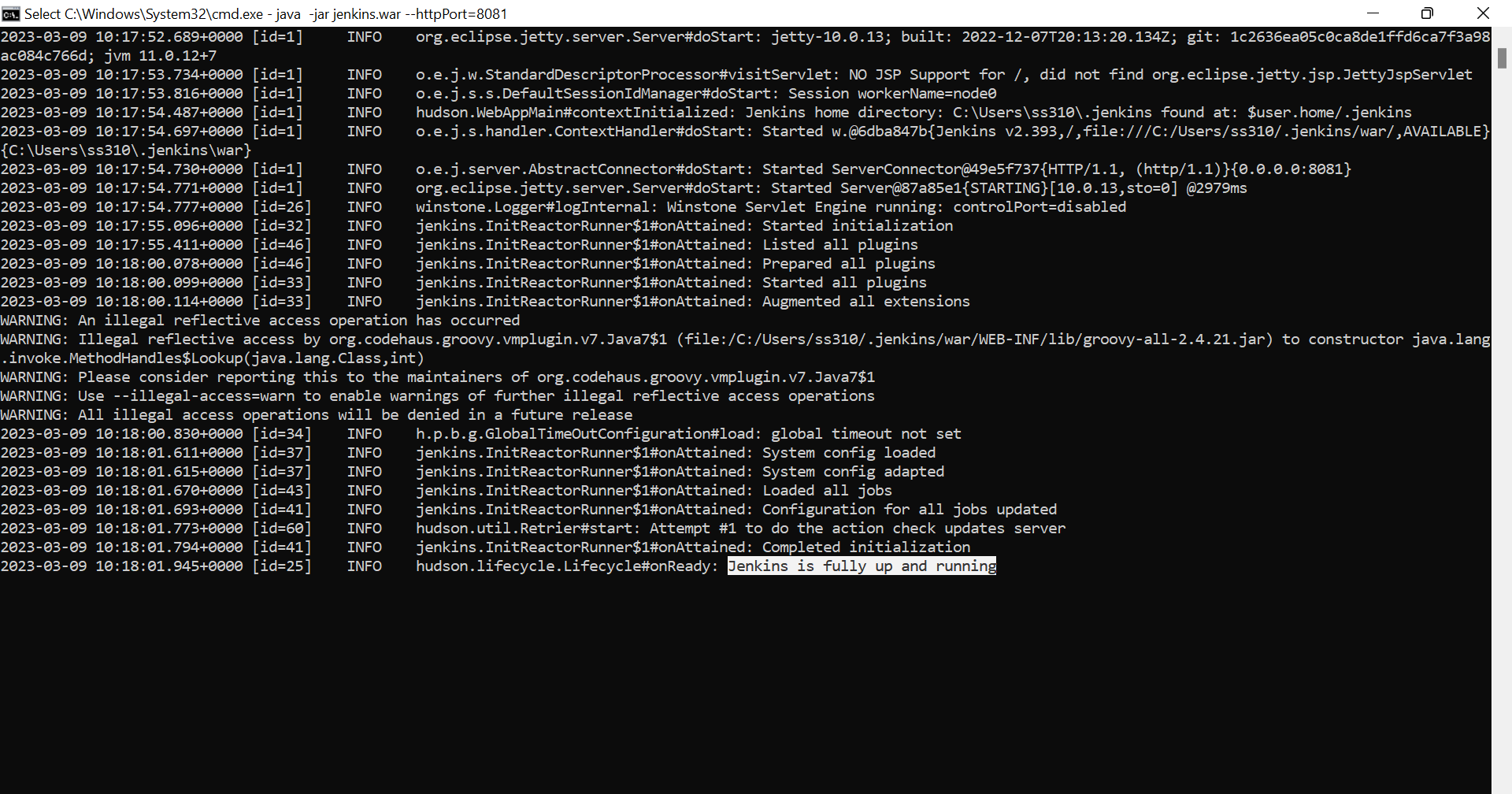
used to start Jenkins as a standalone application from the command line.

Here's what each part of the command does:

* "java" - This starts the Java Virtual Machine (JVM), which is required to run Jenkins.
* "-jar jenkins.war" - This tells the JVM to execute the Jenkins war file.
* "--httpPort=8080" - This sets the port number that Jenkins will use to listen for HTTP requests. In this case, it sets the port to 8080.

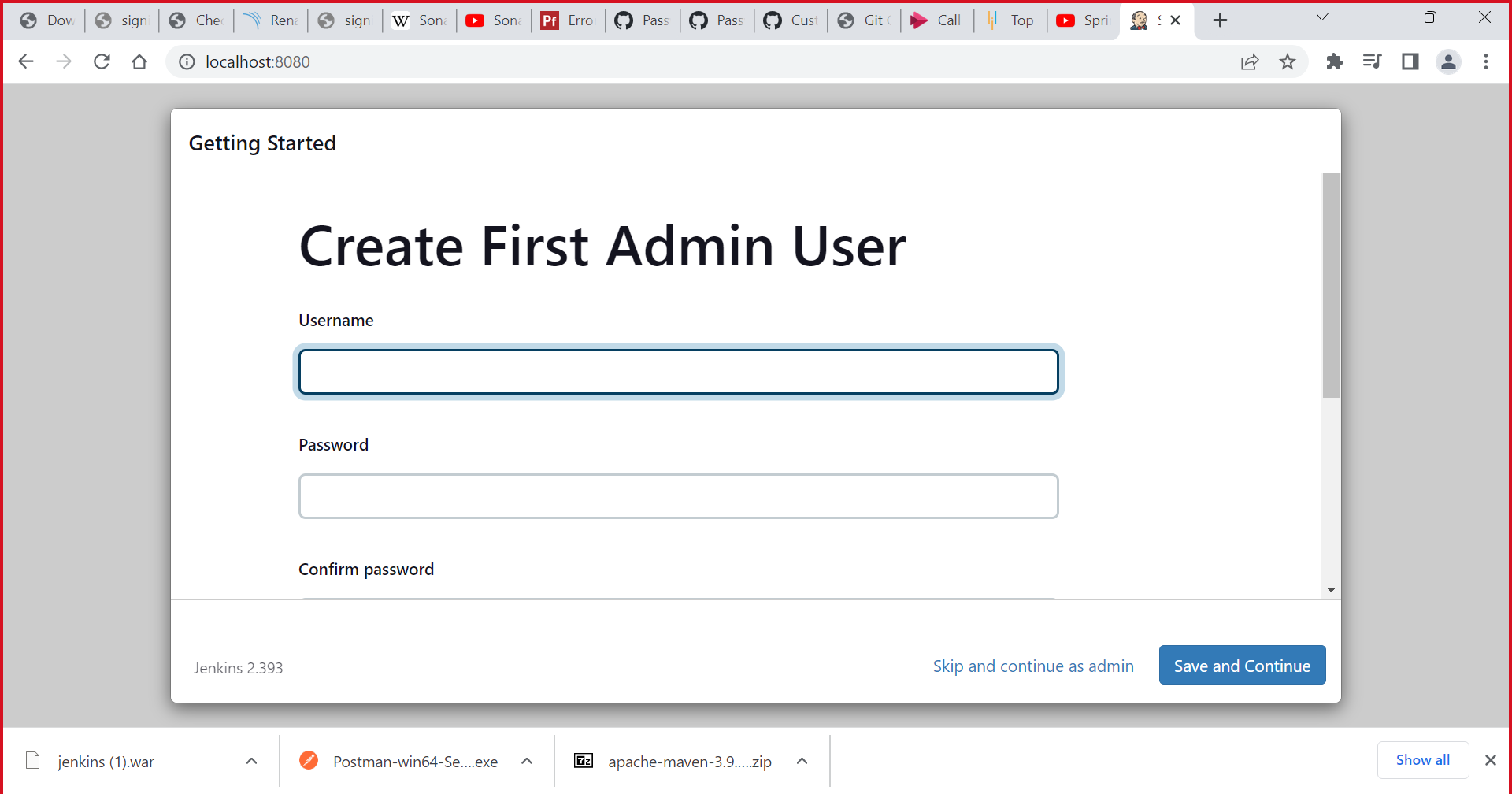
Once the command is executed, Jenkins will start and be available at the URL "http://localhost:8080". You can then access Jenkins through a web browser and begin configuring your Jenkins instance and creating pipelines.

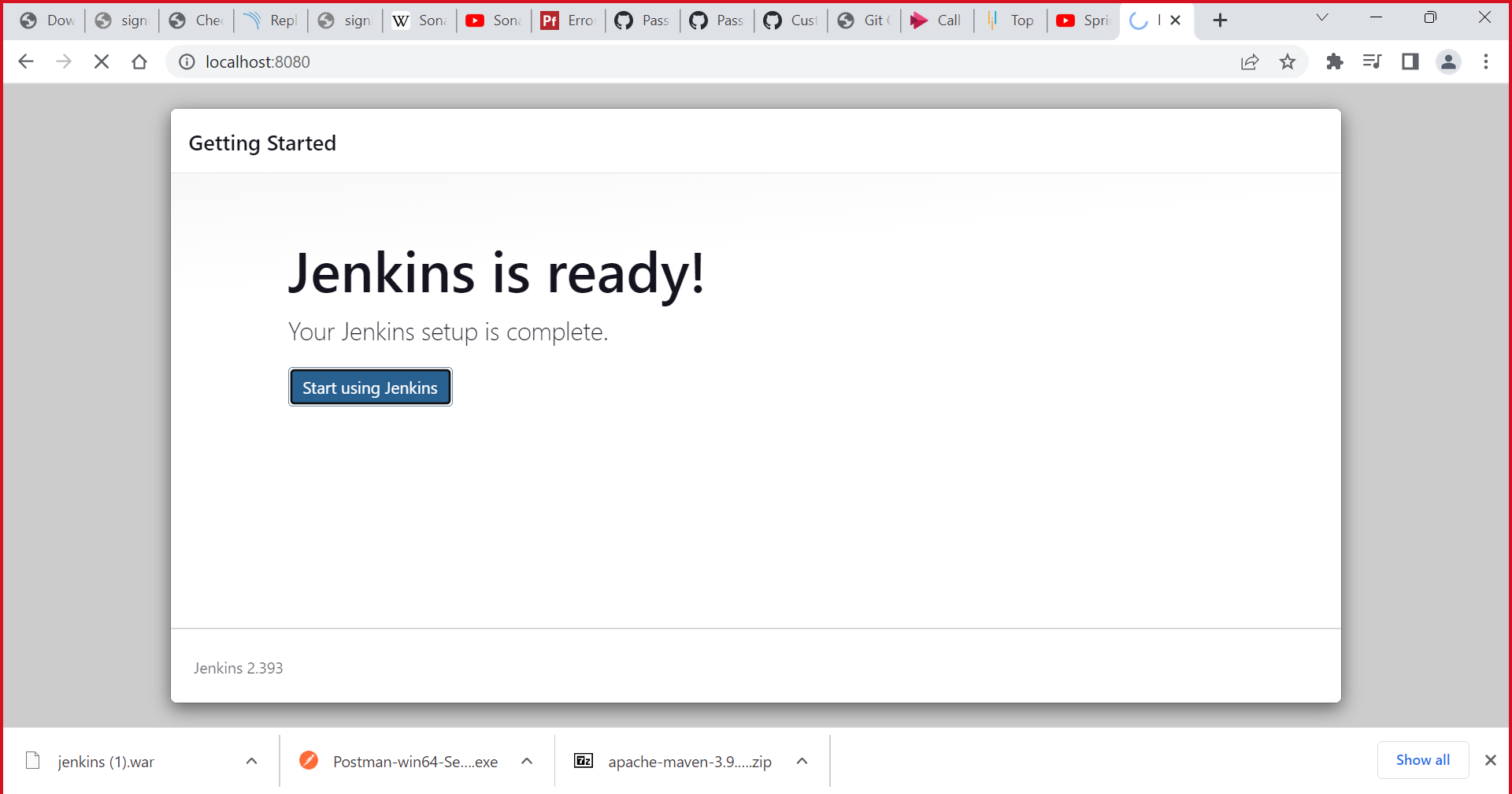




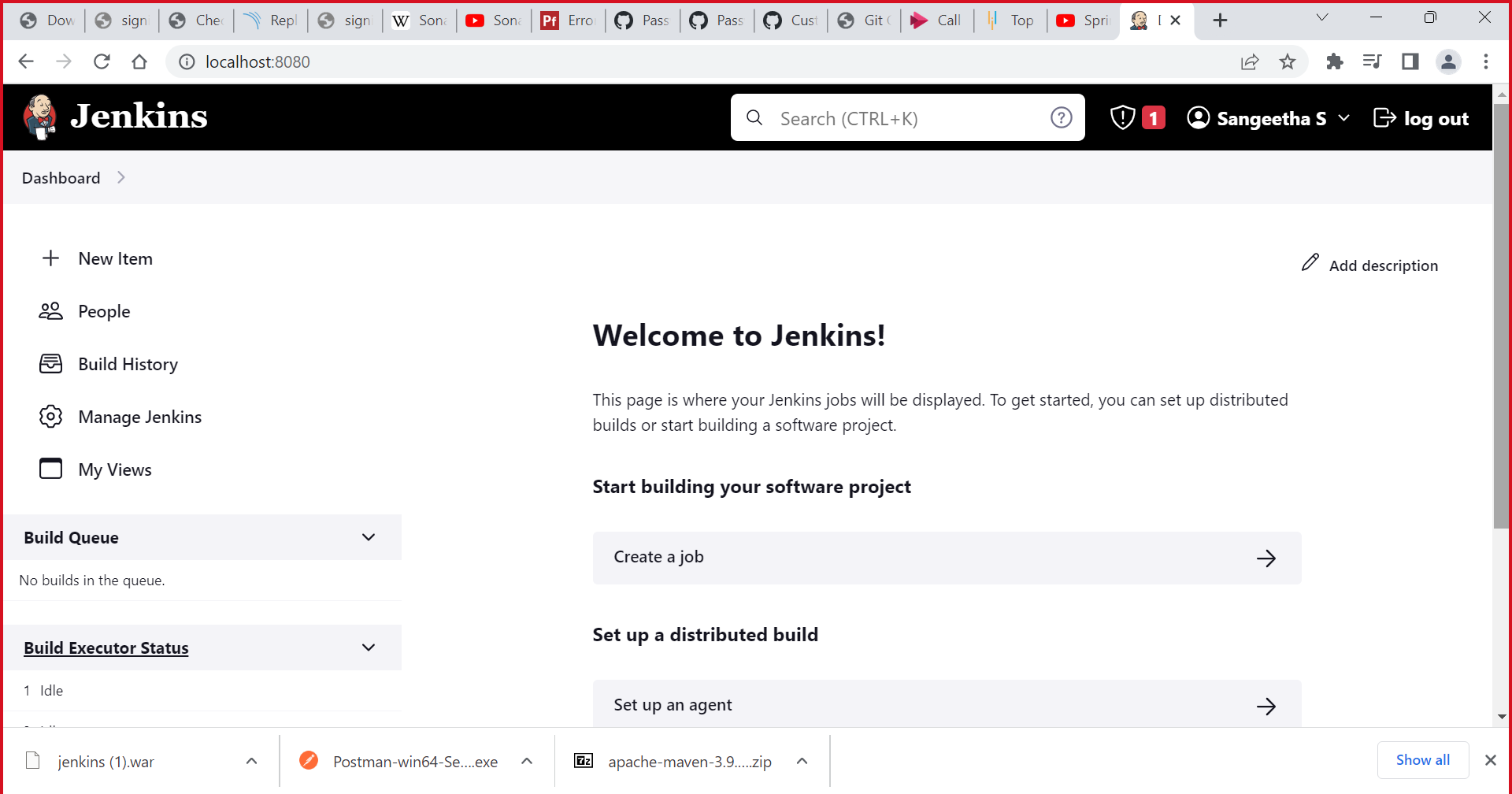
Step 2: after Jenkins is fully up and running search in chrome <http://localhost:8082>

Create user name and password afterthat our Jenkins will ready

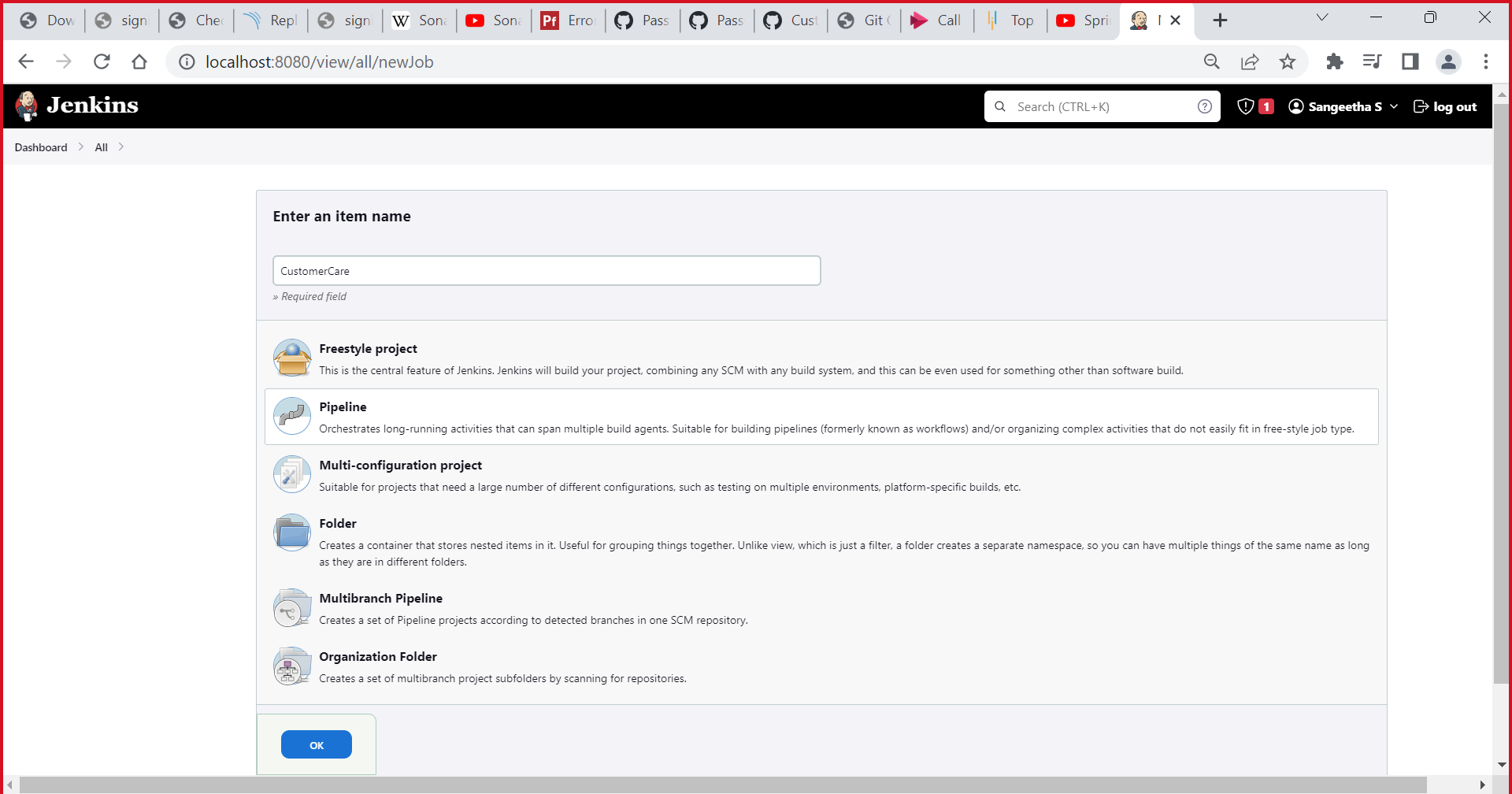




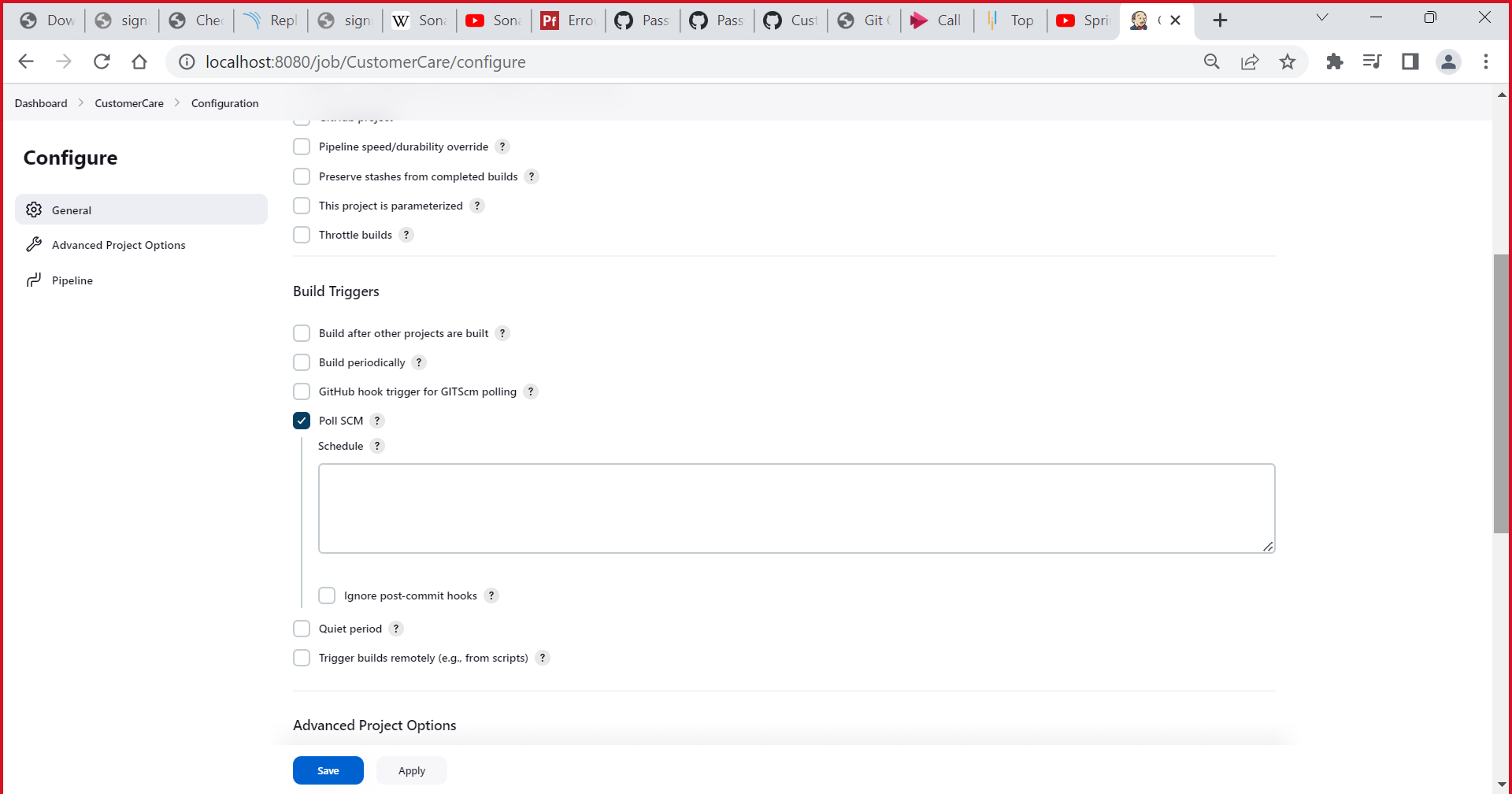
Step 3: Jenkins Dashboard



Step 4: To create a pipeline .We have to enter an item name should select pipeline



Step 5: To create a pipeline ,we should install plugins called maven integration and should configure the build triggers



Step 6: after the configuration part .we should build our code

