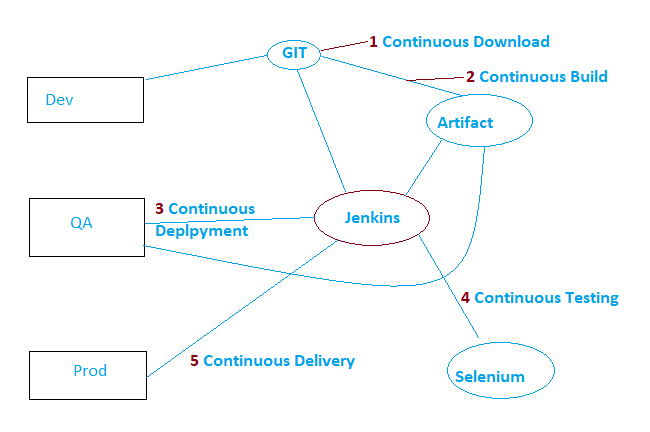
**Jenkins Configuration**

Jenkin is an Open-Source continuous integration and continuous delivery & deployment automation  
DevOps tool which is written in JAVA programming language. It is used to automate parts of the software deployment process such as build, test & deployment.



**How to install and configure Jenkins on cloud?**

1. Sign-in to AWS console
2. Choose EC2 compute
3. Choose Instances and Launch 3 Instances
4. Choose for the required OS flavor

A diagram of a server

Description automatically generated  
DEV Server - Install Java $sudo apt-get install openjdk-8-jdk  
 - Install Git & Maven $sudo apt-get install git maven  
 - Install Jenkin $wget https://www.jenkins.io/download  
 - Start Jenkins $java -jar Jenkins.war  
  
Access Jenkins <ip/hostname:8080>

1. Unlock Jenkins by entering password at the screen
2. Choose Install Suggested plugin
3. Create First User Account, Name-Password-Email  
    **Jenkins installation is completed.  
   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
   Configuration of QA Server**

1. Connect to QA Server using Git bash

2. update the apt repository  
 $sudo apt-get update  
3. Install Tomcat  
 $sudo apt-get install -y tomcat8  
4. Install Tomcat-admin  
 $sudo apt-get install -y tomcat8-admin  
  
  
5. Set the username & password for the tomcat login page  
 $sudo vim /etc/tomcat8/tomcat8-user.xml  
Copy the below command and paste

<tomcat-users>  
 <user username="tomcat" password="Tomcat@123" roles=”manager-script”/>  
</tomcat-users>

6. Restart tomcat server  
 $sudo service tomcat8 restart

1. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
   **Configuration of PROD Server**

1. Connect to PROD Server using Git bash

2. update the apt repository  
 $sudo apt-get update  
3. Install Tomcat  
 $sudo apt-get install -y tomcan8  
4. Install Tomcat-admin  
 $sudo apt-get install -y tomcan8-admin  
  
  
5. Set the username & password for the tomcat login page  
 $sudo vim /etc/tomcat8/tomcat8-user.xml  
Copy the below command and paste  
<tomcat-users>  
 <user username=”testuser” password=”Test@123” roles=”manager-script”/>  
</tomcat-users>  
  
6. Restart tomcat server  
 $sudo service tomcat8 restart

**Stage 1 (Continuous Download)**

The developer creates code and upload that into a remote version controlling system (GIT). Jenkins should be integrated with the remote VCS in such a way that whenever developer makes changes to the code Jenkins should download that code. This is called Continuous Download.

1. Open the dashboard of Jenkins
2. Click New Item ---- Enter the name “TestDev”
3. Select free style project ---OK
4. Goto the Source code management option
5. Select GIT
6. Enter the github url < https://github.com/intelliqittrainings/maven.git > where Developer updated the code
7. Click on Apply --- Save
8. Goto the dashboard of Jenkins and select the job TestDev and Click on build icon
9. This job will download all the codes from the remote github repository

**Stage 2 (Continuous Build)**

The downloaded code, previous stage has to be converted into to be Artifact, This artifact can be in the format of war, jar, ear, exe ect… To create this artifact Jenkins will take the help of build tolls like ant, Maven, MS build etc… This stage is called as continuous build.

**Stage 3 (Continuous Deployment)**

The Artifact created in the previous stage will be deployed into the testing environment. This testing environment will be running on same application servers like tomcat, jboss etc…   
Jenkins will deploy the artifact into these applications servers so that testers can start accessing it This stage is called continuous Deployment.1. Open the dashboard of Jenkins  
2. Click on manage Jenkins ---- Manage Plugins  
3. Goto Available section ---- Search for “Deploy to Container” Plugin – Install it  
4. Open the dashboard of Jenkins  
5. Goto the Development Job ---- click on configure  
6. Giti Post Build actions ----- click on Add port build action  
7. Click on Deploy war/ear to container  
8. Enter files to be deployed as: \*\*/\*.war  
 Context Path: testapp  
 Click on Add container ---- Select tomcat9  
 enter the credentials of tomcat9

Tomcat url: hostname / server ip:8080

9. Click on Apply --- Save  
10. Goto the dashboard of Jenkins and run the development job.  
 Now it will deploy the artifact into the Q Servers and we can access the application  
a) Launch any browser  
b) ipaddress\_hostname:8080/testapp