

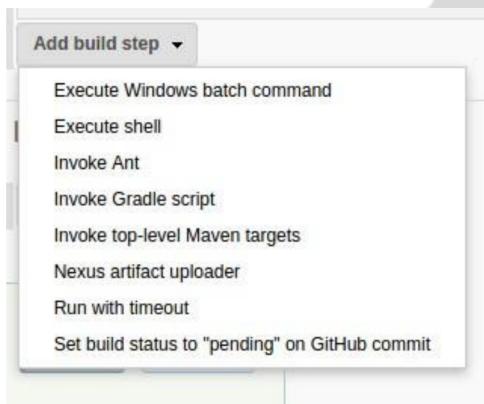
New Hosted Repository

Repository ID	gol-repo
Repository Name	gol-repo1
Repository Type	hosted
Provider	Maven2
Format	maven2
Repository Policy	Release
Default Local Storage Location	
Override Local Storage Location	
Access Settings	
Deployment Policy	Disable Redeploy
Allow File Browsing	True
Include in Search	True
Publish URL	True
Expiration Settings	
Not Found Cache TTL	1440 minutes

Save Cancel

Configure nexus plugin to push the artefact to nexus repository.

Open the Jenkins build job => Add build step => Nexus artefact uploader.



The screenshot shows the Jenkins configuration page for the 'GameOfLife-proj' job. The 'Build' tab is selected. Under the 'Nexus Details' section, the following settings are configured:

- Nexus Version: NEXUS2
- Protocol: HTTP
- Nexus URL: 192.168.1.11:8081/nexus
- Credentials: admin***** (Nexus)
- Groupid: Dev
- Version: \$BUILD_ID
- Repository: gameoflife-repo
- Artifacts:
 - Artifact: \$BUILD_TIMESTAMP
 - Type: war
 - Classifier:
 - File: gameoflife-web/target/gameoflife.war

At the bottom, there are 'Save' and 'Apply' buttons.

Now build the job multiple times and verify. Login to Nexus repository and check the hosted URL's

Console Output

```
file: gameoflife.war
Repository:gol-repo
Uploading: http://192.168.1.9:8081/nexus/content/repositories/gol-repo/Dev/$BUILD_TIMESTAMP/3/$BUILD_TIMESTAMP-3.war
10 % completed (319 kB / 3.2 MB).
20 % completed (639 kB / 3.2 MB).
30 % completed (958 kB / 3.2 MB).
40 % completed (1.3 MB / 3.2 MB).
50 % completed (1.6 MB / 3.2 MB).
60 % completed (1.9 MB / 3.2 MB).
70 % completed (2.2 MB / 3.2 MB).
80 % completed (2.6 MB / 3.2 MB).
90 % completed (2.9 MB / 3.2 MB).
100 % completed (3.2 MB / 3.2 MB).
Uploaded: http://192.168.1.9:8081/nexus/content/repositories/gol-repo/Dev/$BUILD_TIMESTAMP/3/$BUILD_TIMESTAMP-3.war (3.2 MB at 9.2 MB/s)
Uploading artifact gameoflife.war completed.
Finished: SUCCESS
```

Nexus output

Login to nexus and verify the repository data. You should see a versioned artifact their.

The screenshot shows the Nexus 'gol-repo' interface. The 'Browse Index' tab is selected. The repository structure is as follows:

- gol-repo
 - Dev
 - \$BUILD_TIMESTAMP
 - +201706061052
 - 201706061054
 - 5
 - 201706061054-5.war

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: - +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

Click on its hosted url to verify and download the artefact.

gol-repo hosted ANALYZE maven2 Release In Service http://192.168.1.9:8081/nexus/content/repositories/gol-repo



Index of /repositories/gol-repo/Dev/201706061054/5

Name	Last Modified	Size	Description
Parent Directory			
201706061054-5.war	Tue Jun 06 10:54:12 UTC 2017	3192600	
201706061054-5.war.md5	Tue Jun 06 10:54:12 UTC 2017	32	

If you run
this job
multiple
times you

will see every time we get an artefact with a new name.

In any point in time we can use older versions of the artefact if something breaks in newer version.

11. Static Code Analysis for Game of life Dev project.

What is Static Code Analysis?

Static Code Analysis (also known as Source Code Analysis) is usually performed as part of a Code Review (also known as white-box testing) and is carried out at the Implementation phase of a Security Development Lifecycle (SDL). Static Code Analysis commonly refers to the running of Static Code Analysis tools that attempt to **highlight possible vulnerabilities within 'static' (non-running) source code** by using techniques such as Taint Analysis and Data Flow Analysis.

Steps:

1. Install checkstyle plugin.
2. In Maven build step update the Goals as displayed below.



3. Click Post build action and select "Publish checkstyle analysis results".

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: - +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

Post-build Actions

The screenshot shows the Jenkins 'Post-build Actions' configuration page. A single step is selected: 'Publish Checkstyle analysis results'. This step has a 'Checkstyle results' field containing a placeholder message about fileset includes. There is an 'Advanced...' button at the bottom right.

4. Save the project and run it minimum two times.
5. Go to Dev Jobs dashboard and see the checkstyle trend graph.

The screenshot shows the Jenkins Project GOL-Dev dashboard. On the right, there is a large chart titled 'Checkstyle Trend' which is currently solid red, indicating a failing trend. The left sidebar shows build history with three recent builds (#18, #17, #16) all showing a failure icon.

6. Click on checkstyle warnings and see the analysis in detail

The screenshot shows the Jenkins CheckStyle Result page for build #18. It displays a summary table of warnings and a detailed 'Files' tab showing a bar chart of warning distribution across various Java files. The chart shows the following data:

File	Total Distribution
Cell.java	8
GameController.java	23
Grid.java	30
GridReader.java	7
GridWriter.java	6
HomePageController.java	5
Universe.java	26
system.properties	1
Total	106

12. Continuous Delivery with Jenkins

We are going to see now how deployment happens from Jenkins to tomcat server.

This is going to be a full-fledged Continuous delivery project by using Jenkins.

We will need to setup two tomcat servers where we will deploy our artefact.

Tomcat setup

Login as root user

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

```
sudo -i  
apt-get update  
apt-get install default-jre  
apt-get install zip  
wget https://www-eu.apache.org/dist/tomcat/tomcat-8/v8.5.15/bin/apache-tomcat-8.5.15.zip  
mv apache-tomcat-8.5.15.zip /opt  
cd /opt  
unzip apache-tomcat-8.5.15.zip  
cd apache-tomcat-8.5.15
```

Give tomcat start & stop script executable permission

```
cd /opt/apache-tomcat-8.5.15/bin  
chmod u+x startup.sh  
chmod u+x shutdown.sh  
chmod u+x catalina.sh
```

Open browser and verify tomcat service

<http://tomcatIP:8080>

Set user, password & role for tomcat server.

```
cd /opt/apache-tomcat-8.5.15/  
vi conf/tomcat-users.xml
```

Replace below mentioned content

```
-->  
<!--  
    <role rolename="tomcat"/>  
    <role rolename="role1"/>  
    <user username="tomcat" password="" roles="tomcat"/>  
    <user username="both" password="" roles="tomcat,role1"/>  
    <user username="role1" password="" roles="role1"/>  
-->  
</tomcat-users>
```

with

```
-->
```

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

```

<role rolename="manager-script"/>
<role rolename="manager-gui"/>
<user username="tomcat" password="tomcat" roles="manager-script,manager-gui"/>
</tomcat-users>

```

Comment antiResourceLocking values.

vi webapps/manager/META-INF/context.xml

Replace

```

<Context antiResourceLocking="false" privileged="true" >
    <Valve className="org.apache.catalina.valves.RemoteAddrValve"
        allow="127\\.\\d+\\.\\d+\\.\\d+|::1|0:0:0:0:0:0:0:1" />
</Context>

```

with

```

<Context antiResourceLocking="false" privileged="true" >
<!--
    <Valve className="org.apache.catalina.valves.RemoteAddrValve"
        allow="127\\.\\d+\\.\\d+\\.\\d+|::1|0:0:0:0:0:0:0:1" />
-->
</Context>

```

Stop and start tomcat.

bin/shutdown.sh
bin/startup.sh

Verify the settings

<http://tomcatIP:8080/manager/>

Username: tomcat
password: tomcat

You should see below mentioned screen.



Tomcat Web Application Manager

Message:		OK	
Manager			
List Applications		HTML Manager Help	
Manager Help		Server Status	
Applications			
Path	Version	Display Name	Running
	None specified	Welcome to Tomcat	true
docs	None specified	Tomcat Documentation	true
examples	None specified	Servlet and JSP Examples	true
host-manager	None specified	Tomcat Host Manager Application	true
manager	None specified	Tomcat Manager Application	true

Project Setup

This project is a complete build pipeline which will build java SC, create war package, run static code analysis, deploy package to staging tomcat server & deploy to prod tomcat server.

Pipeline consist of below mentioned Jenkins job working together to create entire continuous delivery pipeline.

1. **Package job** – This job will checkout source code from git repository, use Maven to build the code, archive the artefacts, trigger Static code analysis job & staging deploy job
2. **Static code analysis** – This job will checkout source code from git repository, use Maven to run checkstyle code analysis and publish graph for STA.
3. **Deploy to staging Tomcat server** – This job will copy the war artefact from package job to deploy job, Deploy the artefacts to Staging tomcat server & trigger Prod deploy job.
4. **Deploy to Prod Tomcat Server** – This job will copy the war artefact from package job to deploy job, Deploy the artefacts to Staging tomcat server & trigger Prod deploy job.



Prerequisites:

1. Jenkins server: - Jenkins server should have openjdk 1.7, Git, Maven (Follow Jenkins setup doc)
2. Plugins: - git, maven, copy artifacts, deploy to container, checkstyle, build pipeline
3. Staging tomcat server: - Create a vm or ec2 instance with ubuntu OS & Follow tomcat setup doc.
4. Prod tomcat server:- Create a vm or ec2 instance with ubuntu OS & Follow tomcat setup doc.

Create Jenkins jobs:

1. **CJP-package:** - Create a new job with CJP-package name, freestyle project.
Refer below mentioned screenshots to setup the job

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

Source Code Management

None
 Git

Repositories

Repository URL: <https://github.com/jleetutorial/maven-project.git>

Credentials: - none -

Branches to build

Branch Specifier (blank for 'any'): */master

Repository browser: (Auto)

Additional Behaviours:

Subversion

Build

Invoke top-level Maven targets

Goals: clean package

Post-build Actions

Archive the artifacts

Files to archive: **/*.war

Build other projects

Projects to build: CJP-code-analysis, CJP-Deploy-Stage

Trigger only if build is stable
 Trigger even if the build is unstable
 Trigger even if the build fails

2. CJP-code-analysis: - Create a new job with CJP-code-analysis name, freestyle project.
Refer below mentioned screenshots to setup the job

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

Source Code Management

- None
- Git

Repositories

Repository URL

Credentials

- none -



[Advanced...](#)

[Add Repository](#)

Branches to build

Branch Specifier (blank for 'any')

[Add Branch](#)

Repository browser

(Auto)

Additional Behaviours

[Add](#) ▾

- Subversion

Build

Invoke top-level Maven targets

Goals

[Advanced...](#)

[Add build step](#) ▾

Post-build Actions

Publish Checkstyle analysis results

Checkstyle results

Fileset **includes** setting that specifies the generated raw CheckStyle XML report files, such as `**/checkstyle-result.xml`. Basedir of the fileset is [the workspace root](#). If no value is set, then the default `**/checkstyle-result.xml` is used. Be sure not to include any non-report files into this pattern.

[Advanced...](#)

[Add post-build action](#) ▾



Visualpath Training & Consulting

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in

3. CJP-Deploy-Stage:- Create a new job with CJP-Deploy-Stage name, freestyle project.

Refer below mentioned screenshots to setup the job

Build

Copy artifacts from another project

Project name: CJP-Package

Which build: Latest successful build

Stable build only

Artifacts to copy: **/*.war

Artifacts not to copy:

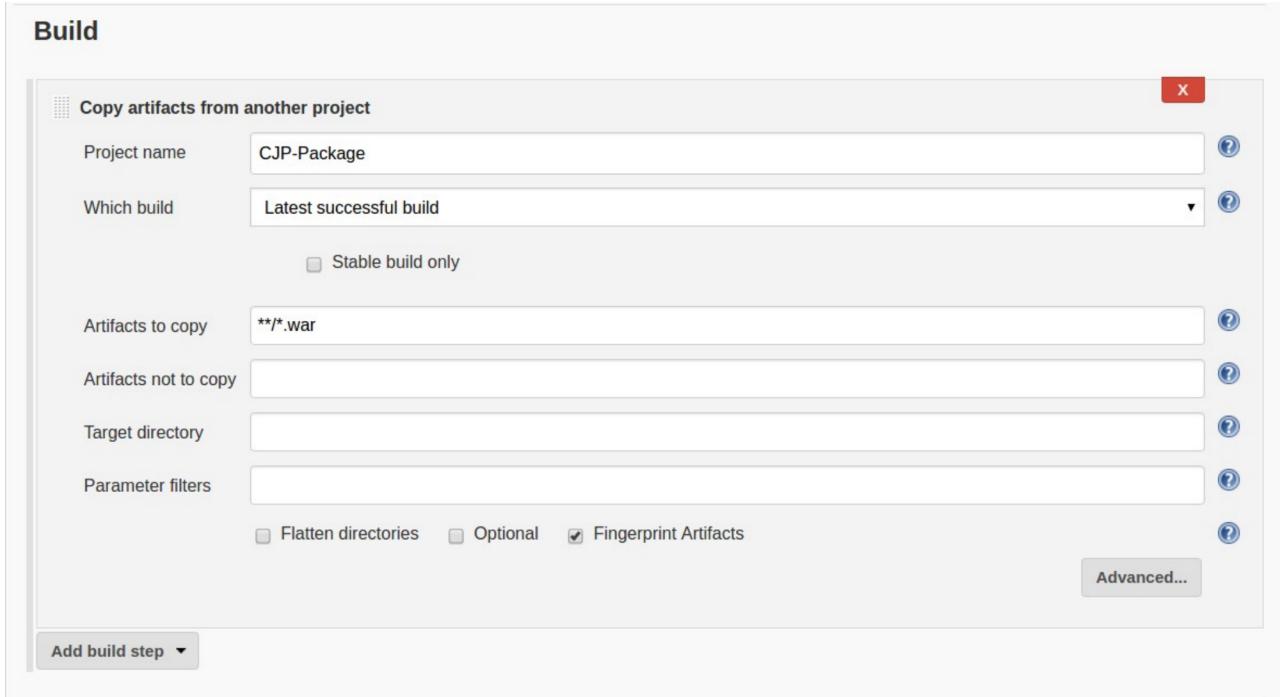
Target directory:

Parameter filters:

Flatten directories Optional Fingerprint Artifacts

[Advanced...](#)

[Add build step ▾](#)



Post-build Actions

Deploy war/ear to a container

WAR/EAR files: **/*.war

Context path:

Containers

Tomcat 7.x

Manager user name: tomcat

Manager password: *****

Tomcat URL: http://192.168.1.9:8080

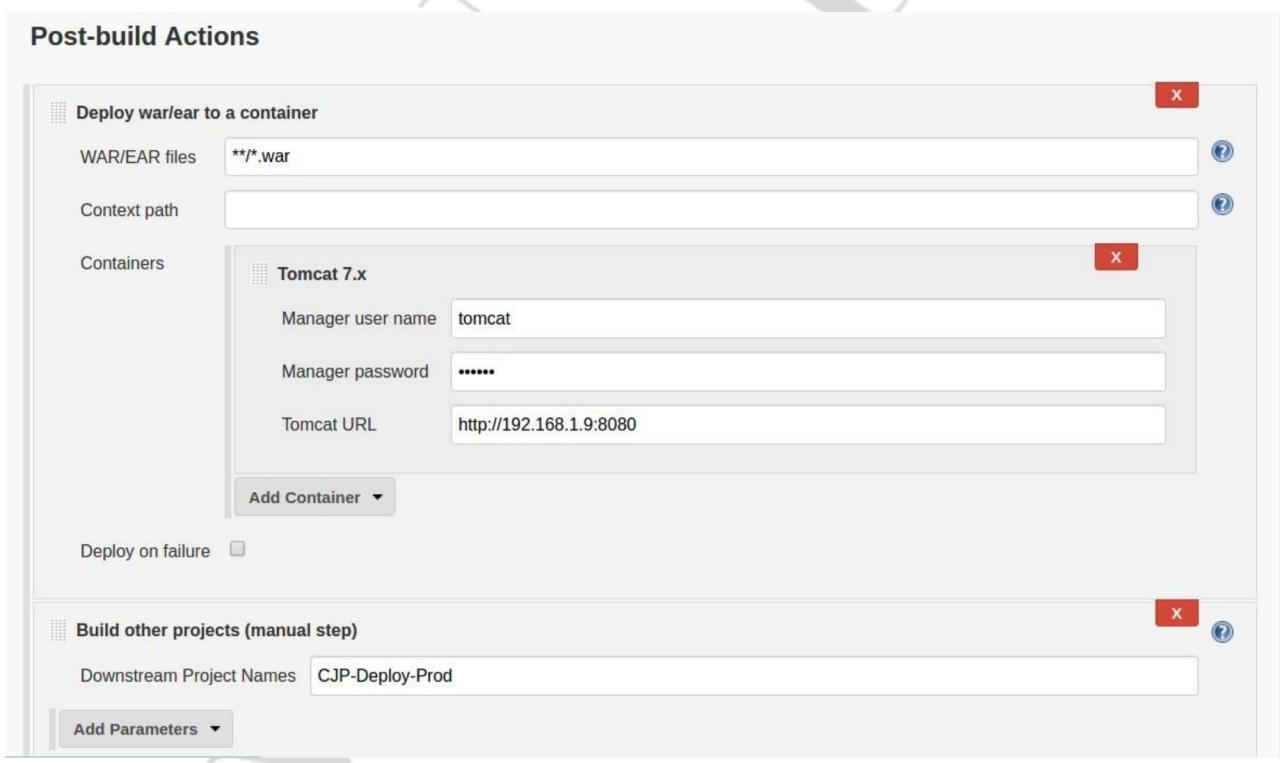
[Add Container ▾](#)

Deploy on failure:

Build other projects (manual step)

Downstream Project Names: CJP-Deploy-Prod

[Add Parameters ▾](#)



Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: - +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

4. CJP-Deploy-Prod: - Create a new job with CJP-Deploy-Prod name, freestyle project.

Refer below mentioned screenshots to setup the job

Build

Copy artifacts from another project

Project name: CJP-Package

Which build: Latest successful build

Stable build only

Artifacts to copy: **/*.war

Artifacts not to copy:

Target directory:

Parameter filters:

Flatten directories Optional Fingerprint Artifacts

Advanced...

Add build step ▾

Post-build Actions

Deploy war/ear to a container

WAR/EAR files: **/*.war

Context path:

Containers

Tomcat 7.x

Manager user name: tomcat

Manager password: *****

Tomcat URL: http://192.168.1.10:8080

Add Container ▾

Deploy on failure

Add post-build action ▾

Setup build pipeline view

Go to Jenkins main dashboard

Click on the plus symbol

All	CJP-place-pipeline	CJP-proj-pipeline	GOL-proj	+
S	W	Name ↓	Last Success	

Visualpa¹

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

Select Build Pipeline View, give pipeline view a name and click OK

View name

Build Pipeline View
Shows the jobs in a build pipeline view. The complete pipeline of jobs that a version propagates through are shown as a row in the view.

List View
Shows items in a simple list format. You can choose which jobs are to be displayed in which view.

My View
This view automatically displays all the jobs that the current user has an access to.

OK

Select initial Job – CJP-Package & click OK

Name: CJP-pipeline

Description:

[Plain text] Preview

Filter build queue

Filter build executors

Build Pipeline View Title

Pipeline Flow

Layout: Based on upstream/downstream relationship

Upstream / downstream config

Select Initial Job: CJP-Package

Trigger Options

Restrict triggers to most recent successful builds: Yes No

Always allow manual trigger on pipeline steps: Yes No

Display Options

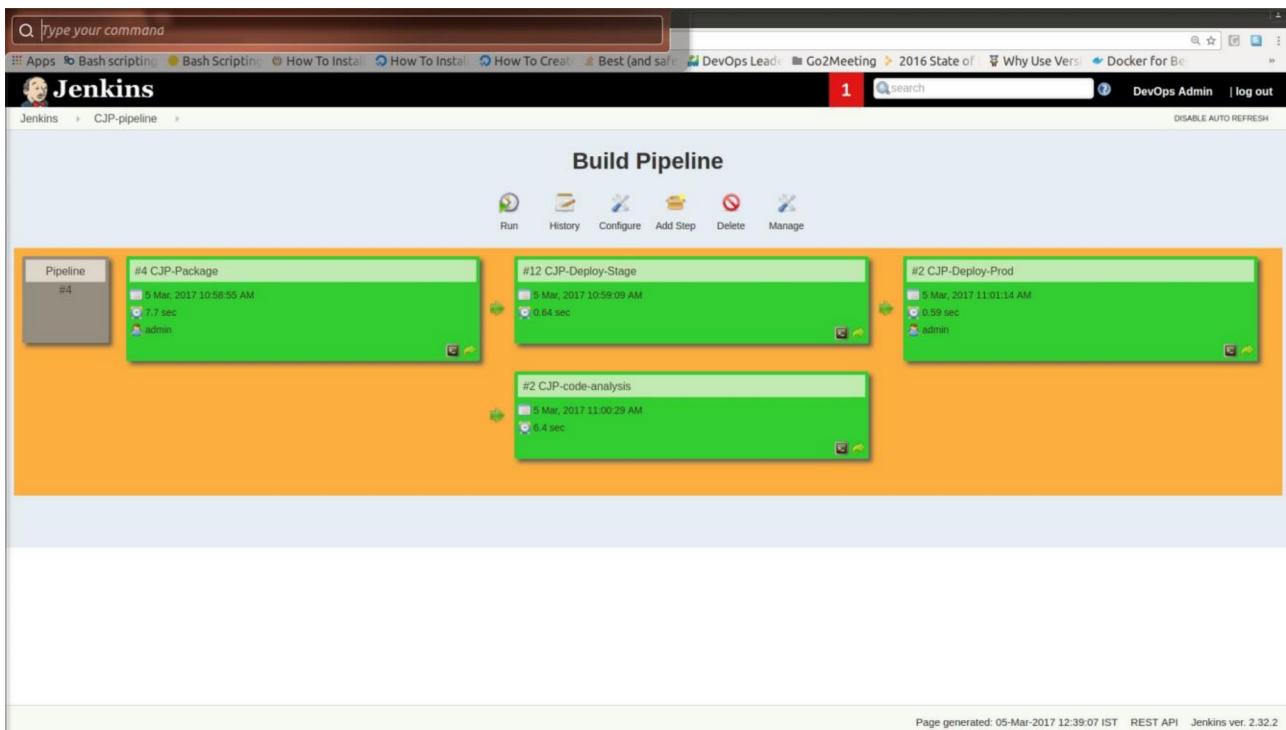
No Of Displayed Builds: 1

Row Headers

OK Apply

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: - +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.



Click run and verify if the entire pipeline works.

13. Jenkins Build Triggers

So far we have seen executing Jenkins jobs manually by clicking on the Build Now button.

But there are other ways by which the jobs can be executed those are called as Build Triggers.

Build Triggers

- Trigger builds remotely (e.g., from scripts)
- Build after other projects are built
- Build periodically
- Poll SCM



Trigger build remotely

We can use this trigger If we want to run the job from a remote location like from a script or from Github or any other tool that can hit the URL.

One typical example for this feature would be to trigger new build from the source control system's hook script, when somebody has just committed a change into the repository, or from a script that parses your source control email notifications.

You'll need to provide an authorization token in the form of a string so that only those who know it would be able to remotely trigger this project's builds.

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: - +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

Trigger builds remotely (e.g., from scripts)

Authentication Token

Use the following URL to trigger build remotely: JENKINS_URL/job/beandeploy/build?token=TOKEN_NAME or /buildWithParameters?token=TOKEN_NAME

Optionally append &cause=Cause+Text to provide text that will be included in the recorded build cause.

Now you can use the URL with the token

JENKINS_URL/job/beandeploy/build?token=TOKEN_NAME to call this job from

- CLI like curl command
 - curl http://username:password@192.168.1.9:8080/job/beandeploy/build?token=electric
- Through a script
- From a browser
- Github webhooks
 - In github repo go to Settings => Webhooks => Enter Jenkins URL and Token => Add Webhook.

Now whenever there is a commit in the git repo it will trigger the Jenkins job by hitting Jenkins remote URL.

Build after other projects are build.

This is to set the job as downstream for another job. Mention the upstream job name, once the upstream job gets completed successfully our job will get triggered.

Build after other projects are built

Projects to watch

Trigger only if build is stable
 Trigger even if the build is unstable
 Trigger even if the build fails

Build Periodically

If we want to schedule our job to get executed for example every night 8 pm then we can use a cronjob format to specify the time.

Build periodically

Schedule

⚠ Spread load evenly by using 'H 20 * * *' rather than '00 20 * * *'. Would last have run at Saturday, 17 June, 2017 8:00:10 PM IST; would next run at Sunday, 18 June, 2017 8:00:10 PM IST.

This field follows the syntax of cron (with minor differences). Specifically, each line consists of 5 fields separated by TAB or whitespace:

MINUTE HOUR DOM MONTH DOW

MINUTE Minutes within the hour (0–59)

HOUR The hour of the day (0–23)

DOM The day of the month (1–31)

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: - +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

MONTH The month (1–12)

DOW The day of the week (0–7) where 0 and 7 are Sunday.

Poll SCM

It's similar to cronjob but instead of running the job at the interval it will go and check if there is any new commit in the VCS repo like github repo's and then execute the job.

14. Sonarqube Integration Jenkins.



DOCUMENTED BY KARUNAKAR G.

SonarQube is an open source platform for continuous inspection of code quality. This will reports on duplicated code, coding standards, unit tests, code coverage and code complexity.

SonarQube can provides a central place to view and define the rules used during analysis of projects. These rulesets are organized in quality profiles. Every member of the organization can see which rules are applied to their project.

Every project administrator can choose which quality profile is used for the project.

Required Plugins

- ✓ SonarQube scanner
- ✓ Maven

SonarCode scanner process in Jenkins

Pull code From Git.

Build and Run Unit Tests

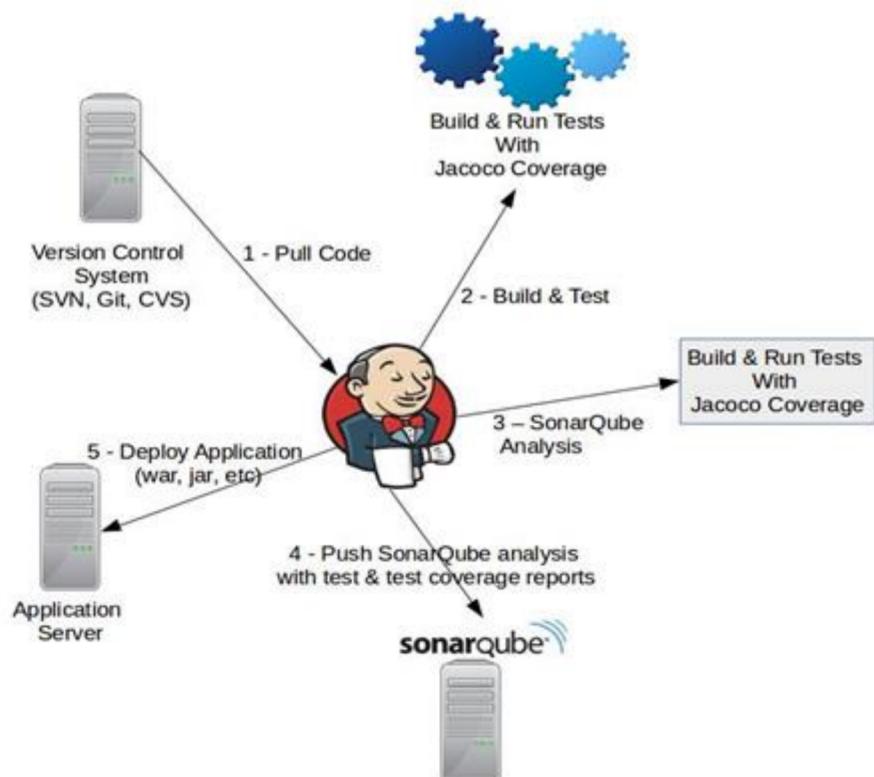
SonarQube analysis with sonarqube scanners.

Push SonarQube analysis coverage reports to SonarQube Dashboard.

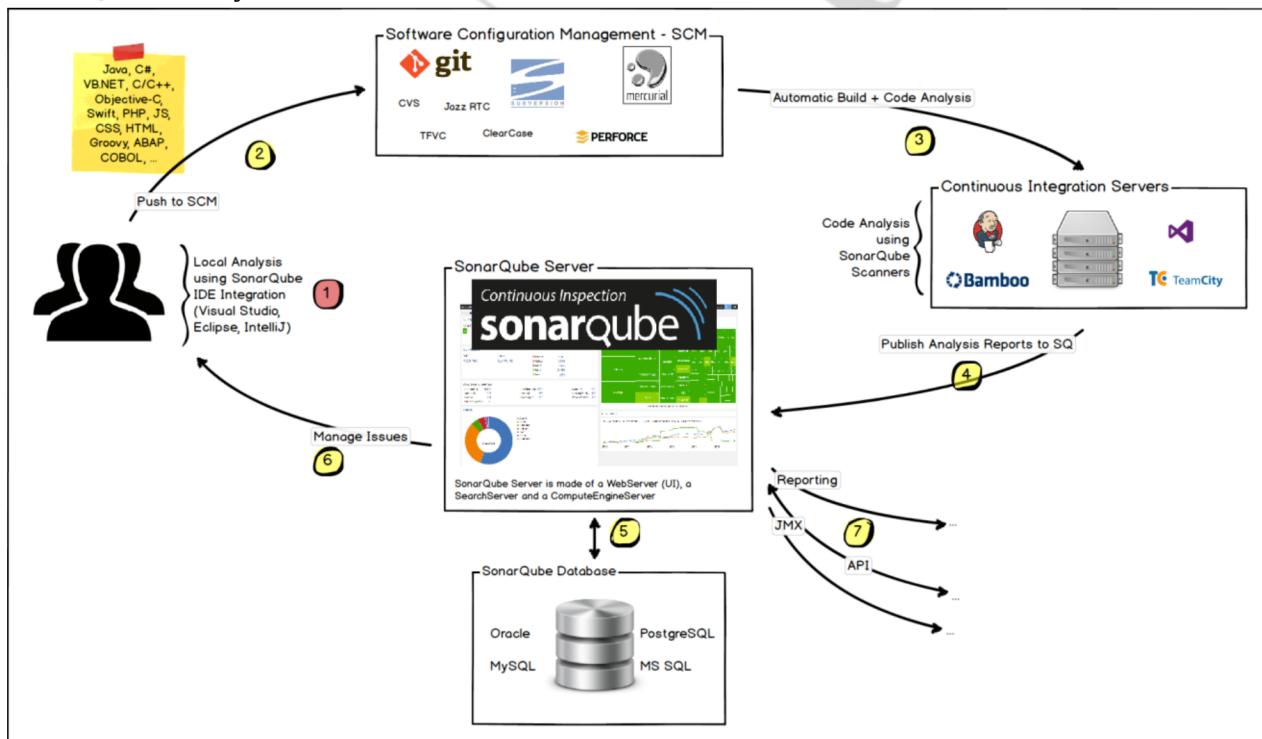
We can store the SonarQube data in DataBase (Mysql).

Deploy applications to server.

SonarQube Lifecycle



SonarQube LifeCycle



Prerequisite for exercise

For the exercise of SonarQube we need two ec2 instances with below details.

For Jenkins Instance

OS – ubuntu 16.10

Type – t2.micro

Securitykey for Jenkins (.pem)

Security Group for Jenkins

Type Custom TCP ->Port 8080->source MyIP

SSH -->Port 22 -> source MyIP

The screenshot shows the AWS EC2 Management Console with the 'Edit inbound rules' dialog box open. The dialog box has four columns: Type, Protocol, Port Range, and Source. There are two rows of rules:

Type	Protocol	Port Range	Source
Custom TCP	TCP	8080	My IP 183.82.113.195/32
SSH	TCP	22	My IP 183.82.113.195/32

Below the table, there is a note: "NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created." At the bottom right of the dialog box are 'Cancel' and 'Save' buttons.

For SonarQube Instance.

OS – ubuntu 16.10

Type – t2.medium

Securitykey for SonarQub (.pem)

Security Group for SonarQube

Type Custom TCP ->Port 9000->source MyIP

Type Custom TCP ->Port 8080->source JenkinsSg

Type Custom TCP ->Port 8080->source MyIP

SSH -->Port 22 -> source MyIP

Edit inbound rules

Type	Protocol	Port Range	Source
Custom TCP	TCP	9000	Custom 183.82.113.195/32
Custom TCP	TCP	9000	Custom sg-978bf5ec
Custom TCP	TCP	8080	Custom 183.82.113.195/32
SSH	TCP	22	Custom 183.82.113.195/32

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel **Save**

In Next Slides, we are showing that complete setup of SonarQube

Launch Instance

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS
tomcat	i-06fa2e6d18ee7a670	t2.micro	us-west-2c	running	Initializing	None	ec2-35-164-52-
sonarqube	i-083d59a6701c9812a	t2.micro	us-west-2b	running	2/2 checks ...	None	ec2-52-33-133-
jenkins ci 2	i-09f7ea6e55020949b	t2.micro	us-west-2c	running	2/2 checks ...	None	ec2-52-88-219-
Jenkins CI-old	i-0d5932cd1fb8f31d5	t2.micro	us-west-2c	stopped		None	

Instances: i-083d59a6701c9812a (sonarqube), i-09f7ea6e55020949b (jenkins ci 2)

Description **Status Checks** **Monitoring** **Tags**

- i-083d59a6701c9812a: ec2-52-33-133-37.us-west-2.compute.amazonaws.com
- i-09f7ea6e55020949b: ec2-52-88-219-149.us-west-2.compute.amazonaws.com

SonarQube Instalation Procedure in Ubuntu 16.10

Prerequisites to setup SonarQube:

EC2 Ubuntu Instance,

- Java-8-oracle,
- Sonarqube version 5.6.6,
- mysql-server 5.7.

Adding Java PPA to Our Ubuntu repository

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

```
add-apt-repository -y ppa:webupd8team/java
```

Updating apt repository

```
apt-get update
```

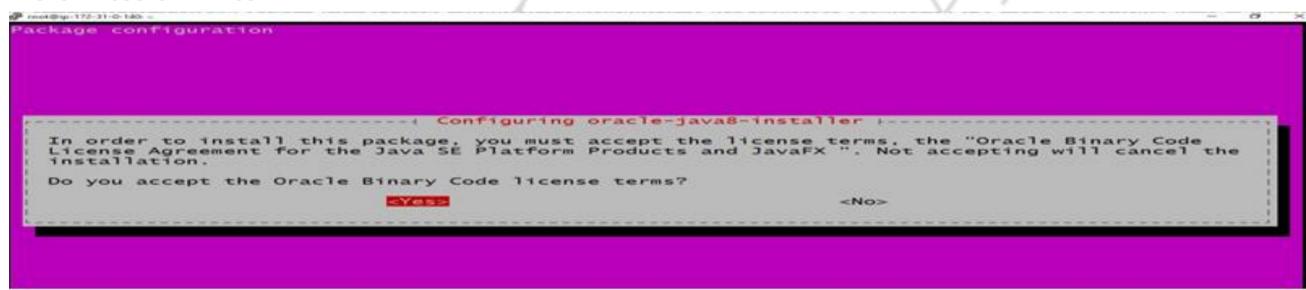
Installing java on EC2 Ubuntu 16.10 Instance

```
apt-get -y install oracle-java8-installer
```

Here Press on “OK”



Here Press on “Yes”



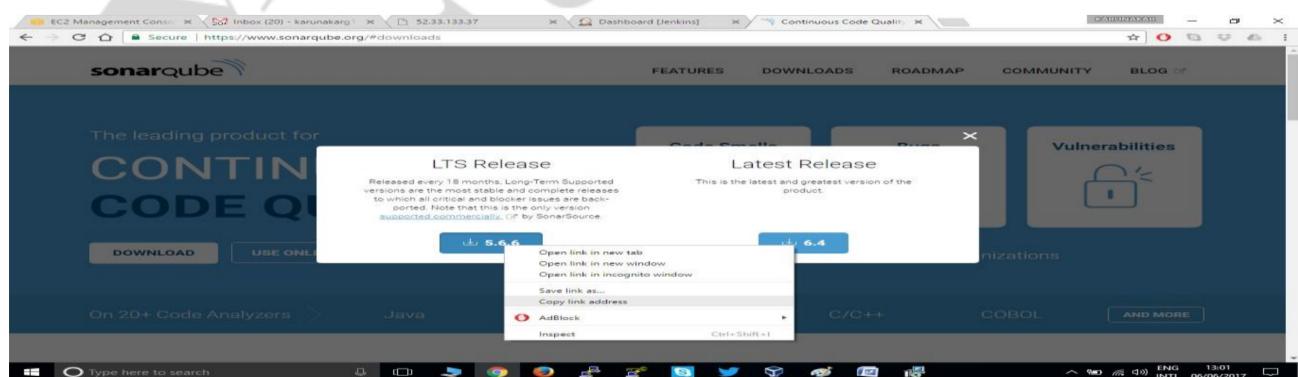
Here We are Creating soft link for java = default java

```
ln -s /usr/lib/jvm/java-8-oracle/ /usr/lib/jvm/default-java
```

SonarQube Installation

```
sudo apt-get install unzip
```

```
wget https://sonarsource.bintray.com/Distribution/sonarqube/sonarqube-5.6.6.zip
```



Here we are installing Unzip

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.

Unzipping the SonarQube Zip file

```
unzip sonarqube-5.6.6.zip  
cd sonarqube-5.6.6/
```

Mysql Installation Procedure in sonarqube server and configuration

Install MySQL Server.

```
$ sudo apt-get install mysql-server
```

Configure the mysql bind address with sonarqube ip address

Start the mysql server

```
$ mysql -u user -p  
Mysql>Create the MySQL database.  
Mysql>CREATE DATABASE sonar CHARACTER SET utf8 COLLATE utf8_general_ci;  
Mysql>CREATE USER 'sonar' IDENTIFIED BY 'sonar';  
Mysql>GRANT ALL ON sonar.* TO 'sonar'@'%' IDENTIFIED BY 'sonar';  
Mysql>GRANT ALL ON sonar.* TO 'sonar'@'localhost' IDENTIFIED BY 'sonar';  
FLUSH PRIVILEGES;
```

Edit the Sonar Configuration in sonarserver.

```
$ sudo vi /sonarqube5.6/conf/sonar.properties  
  
# User credentials.  
# Permissions to create tables, indices and triggers must be  
granted to JDBC user.  
# The schema must be created first.  
sonar.jdbc.username=sonar  
sonar.jdbc.password=sonar  
  
----- MySQL 5.x  
sonar.jdbc.url=jdbc:mysql://sonarip:3306/sonar?useUnicode=true&characterEncoding  
=utf8&rewriteBatchedStatements=true&useConfigs=maxPerformance
```

Starting the sonarqube server

```
bin/linux-x86-64/sonar.sh start
```

stop the firewalls

```
service ufw stop
```

SonarQube DashBoard Mainpage:

Visualpath Training & Consulting.

Flat no: 205, Nilgiri Block, Aditya Enclave, Ameerpet, Hyderabad, Phone No: +91-970 445 5959, 961 824 5689 E-Mail ID : online.visualpath@gmail.com, Website : www.visualpath.in.