

## Experiment No. 4

**Aim:** To understand continuous integration, install and configure Jenkins with Maven/ ANT/ Gradle to set up a building job.

**LO No. & Statement:** (LO3): Illustrate the importance of Jenkins to build and deploy Software Applications on a server environment.

**Theory:**

Jenkins is a self-contained, open-source automation server that can be used to automate all sorts of tasks related to building, testing, and delivering or deploying software. Jenkins is a software that allows continuous integration. Jenkins will be installed on a server where the central build will take place. Continuous Integration is a development practice that requires developers to integrate code into a shared repository at regular intervals. This concept was meant to remove the problem of finding later occurrences of issues in the build lifecycle. Continuous integration requires the developers to have frequent builds. The common practice is that whenever a code commit occurs, a build should be triggered.

**Installation of Jenkins:**

**Jenkins Setup Wizard**



## Jenkins Setup Wizard

Jenkins 2.361.1 Setup

**Destination Folder**

Click Next to install to the default folder or click Change to choose another.

Install Jenkins 2.361.1 to:

C:\Program Files\Jenkins\

Change...

Back Next Cancel

## Jenkins Setup Wizard

Jenkins 2.361.1 Setup

**Service Logon Credentials**

Enter service credentials for the service.

Jenkins 2.361.1 installs and runs as an independent Windows service. To operate in this manner, you must supply the user account credentials for Jenkins 2.361.1 to run successfully.

**Logon Type:**

☒ Run service as LocalSystem (not recommended)

☐ Run service as local or domain user:

Account:

Password:

Test Credentials

Back Next Cancel

## Jenkins Setup Wizard

Jenkins 2.361.1 Setup


**Port Selection**

Choose a port for the service.

Please choose a port.

**Port Number (1-65535):**

8080

Test Port 

It is recommended that you accept the selected default port.

Back Next Cancel

## Jenkins Setup Wizard

Jenkins 2.361.1 Setup

Jenkins 2.361.1 Setup

Select Java home directory (JDK or JRE)

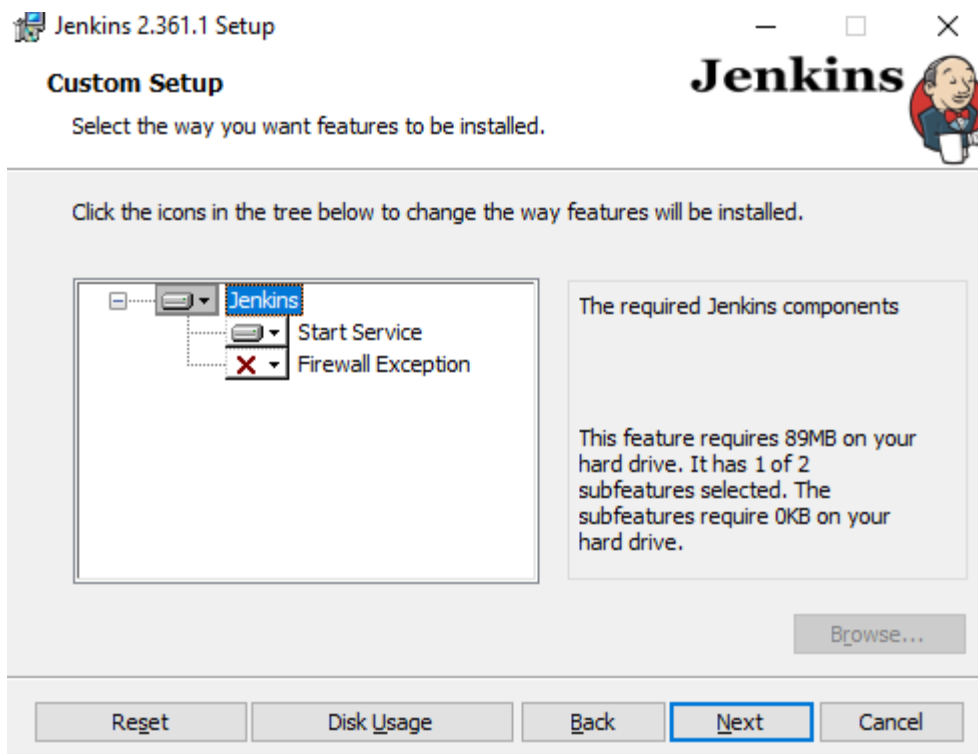
Please select the path of a Java Development Kit or Java Runtime Environment.  
Only Java 11 and 17 are supported by Jenkins.

C:\Program Files\Java\jdk-11.0.11\

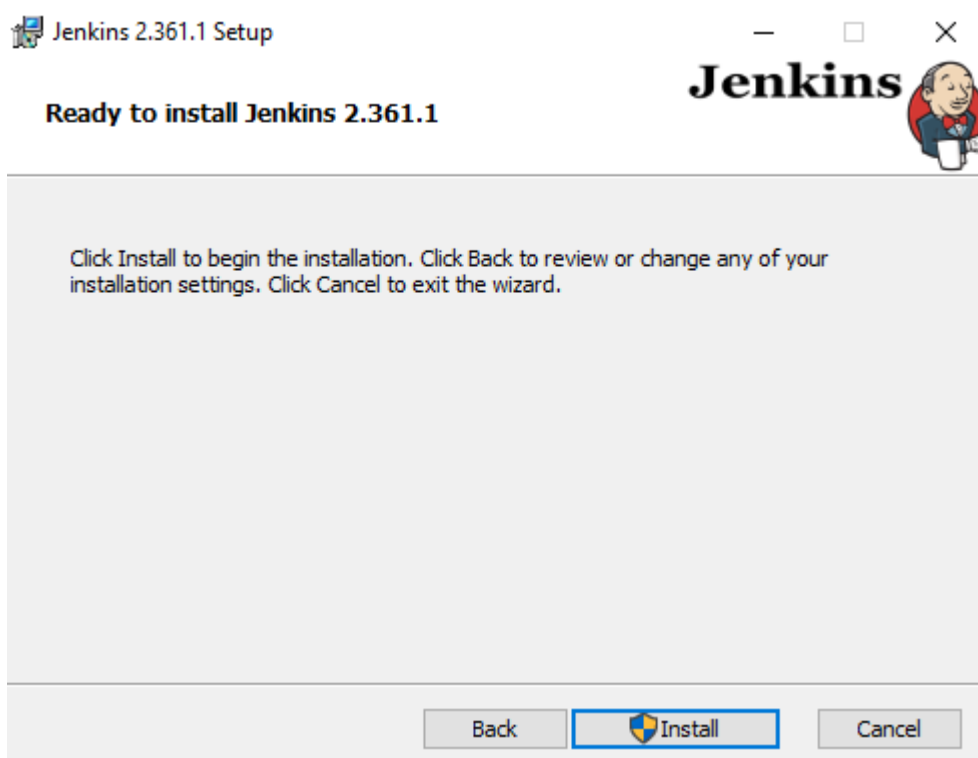
Change...

Back Next Cancel

## Jenkins Setup Wizard



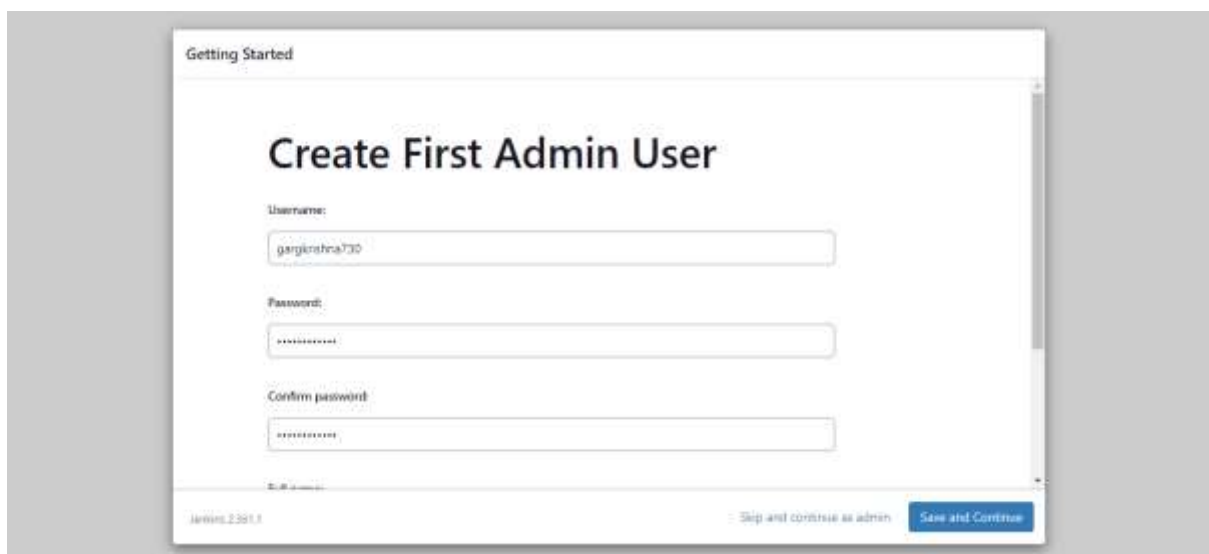
## Jenkins Setup Wizard



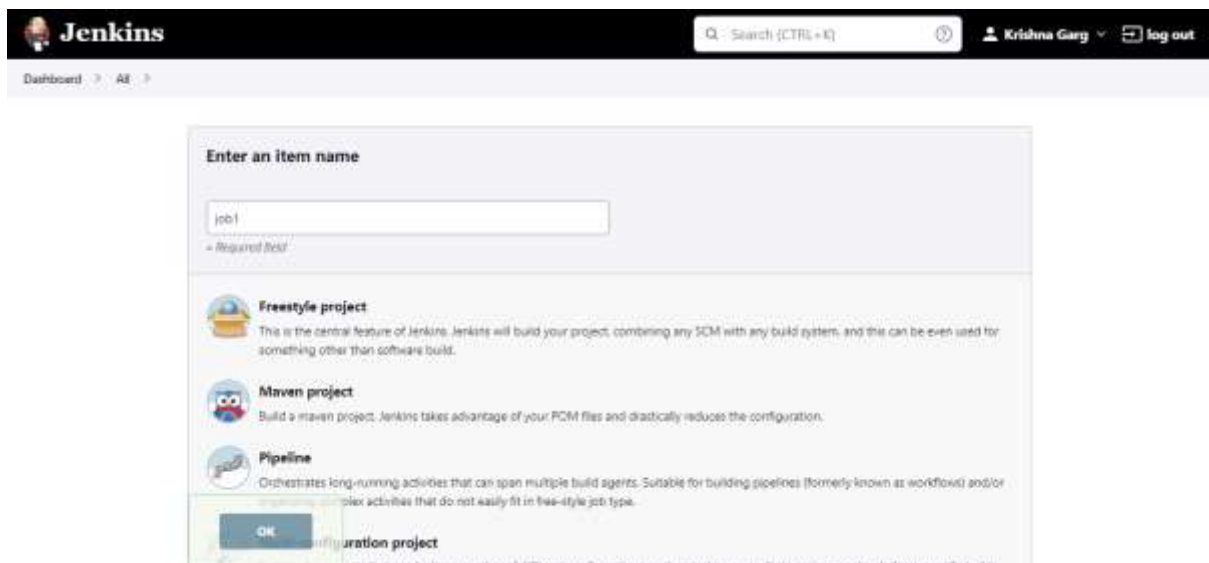
## Successful Installation of Jenkins



## Creating First User



## Creating a Job:



The screenshot shows the Jenkins 'Enter an item name' dialog. At the top, there's a search bar and user information. Below, a text input field contains 'job1'. Underneath, three project types are listed: 'Freestyle project' (described as the central feature), 'Maven project' (for Maven projects), and 'Pipeline' (for long-running activities). An 'OK' button is at the bottom left.

Enter an item name

job1

= Required field

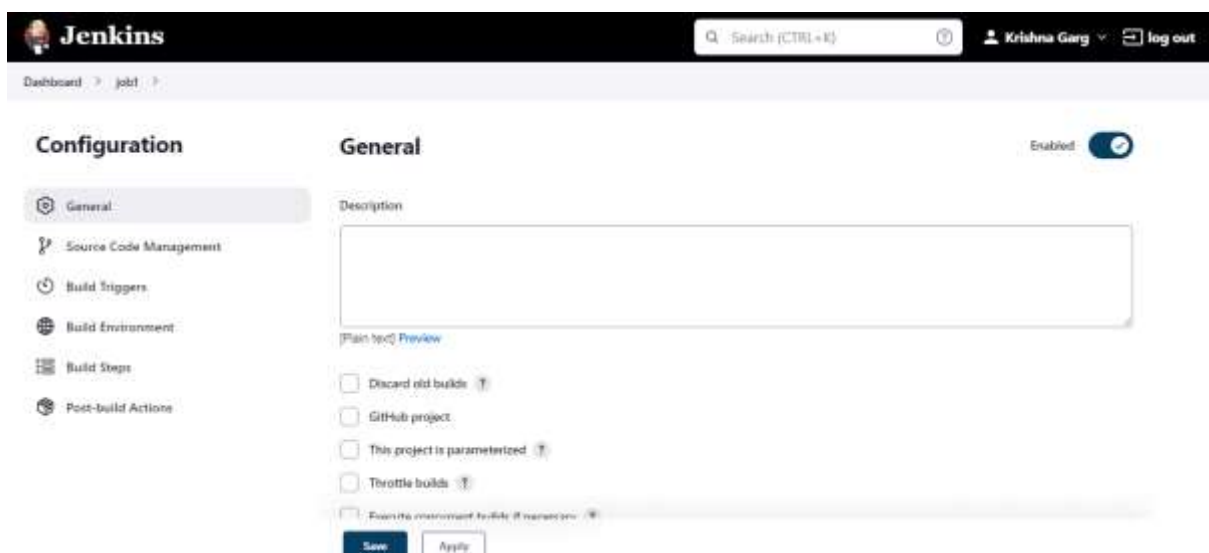
**Freestyle project**  
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

**Maven project**  
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

**Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or complex activities that do not easily fit in free-style job type.

OK

## Configuration:



The screenshot shows the Jenkins configuration page for 'job1'. The left sidebar lists configuration sections: General, Source Code Management, Build Triggers, Build Environment, Build Steps, and Post-build Actions. The 'General' section is active, showing a 'Description' text area, a 'Preview' link, and several checkboxes: 'Discard old builds', 'GitHub project', 'This project is parameterized', 'Throttle builds', and 'Erase the environment before if necessary'. 'Save' and 'Apply' buttons are at the bottom.

Configuration

General

Enabled

Description

[Plain text] Preview

☐ Discard old builds

☐ GitHub project

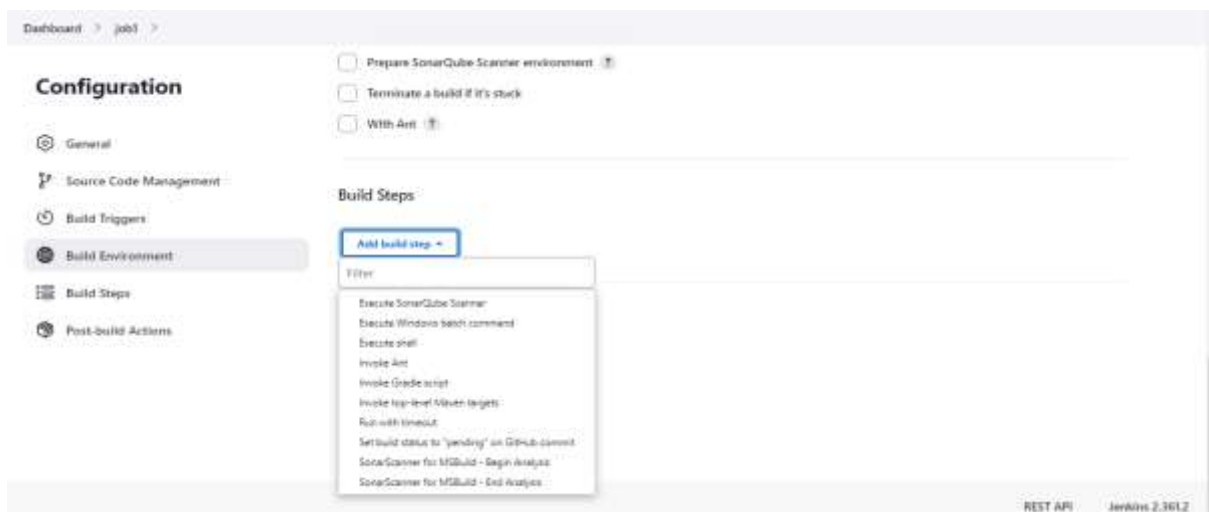
☐ This project is parameterized

☐ Throttle builds

☐ Erase the environment before if necessary

Save Apply

## Execute Windows Batch Command:



The screenshot shows the 'Build Steps' section of the Jenkins configuration page. A dropdown menu is open, showing a list of build steps. The 'Execute Windows batch command' option is highlighted. The bottom right corner shows 'REST API' and 'Jenkins 2.361.2'.

Configuration

General

Source Code Management

Build Triggers

Build Environment

Build Steps

Post-build Actions

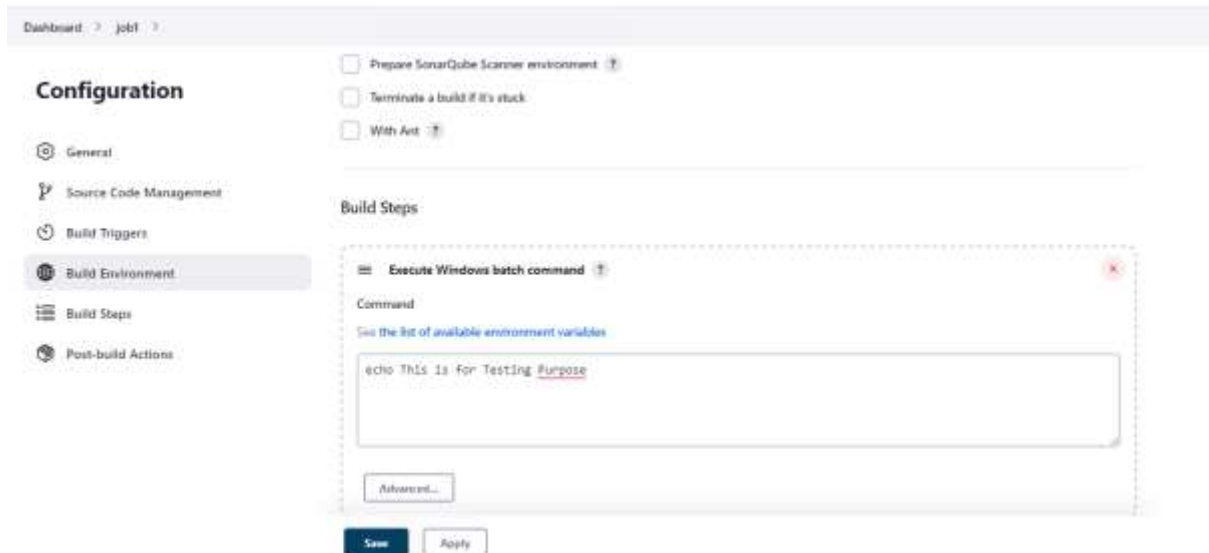
Build Steps

Add build step

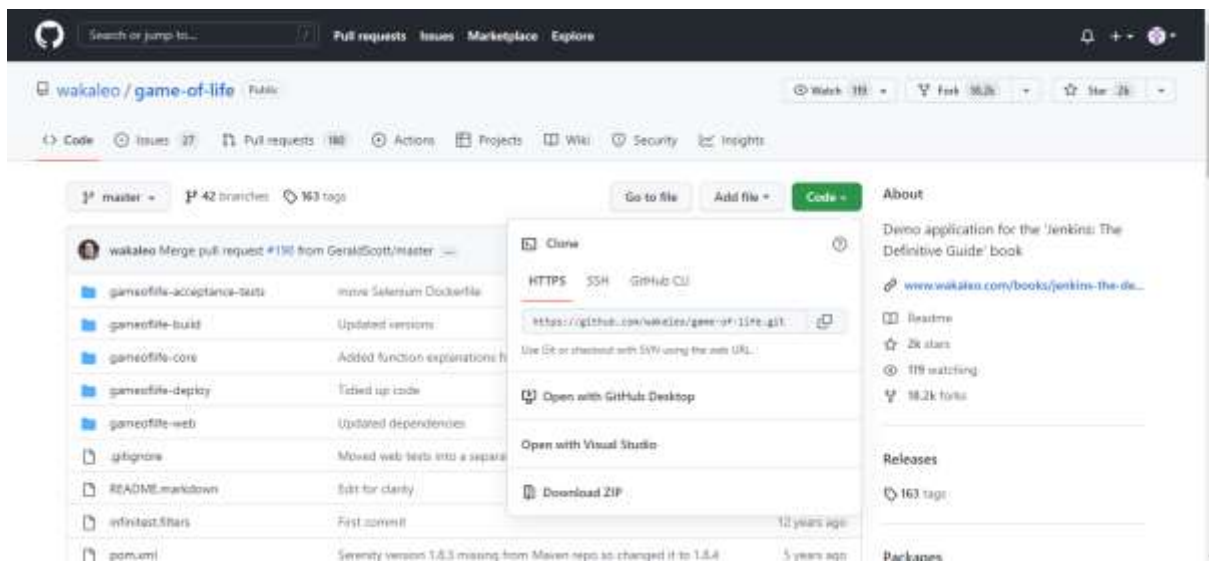
Filter

- Execute SonarQube Scanner
- Execute Windows batch command
- Execute shell
- Invoke Ant
- Invoke Gradle script
- Invoke top-level Maven targets
- Run with timeout
- Set build status to 'pending' on GitHub comment
- SonarScanner for MSBuild - Begin Analysis
- SonarScanner for MSBuild - End Analysis

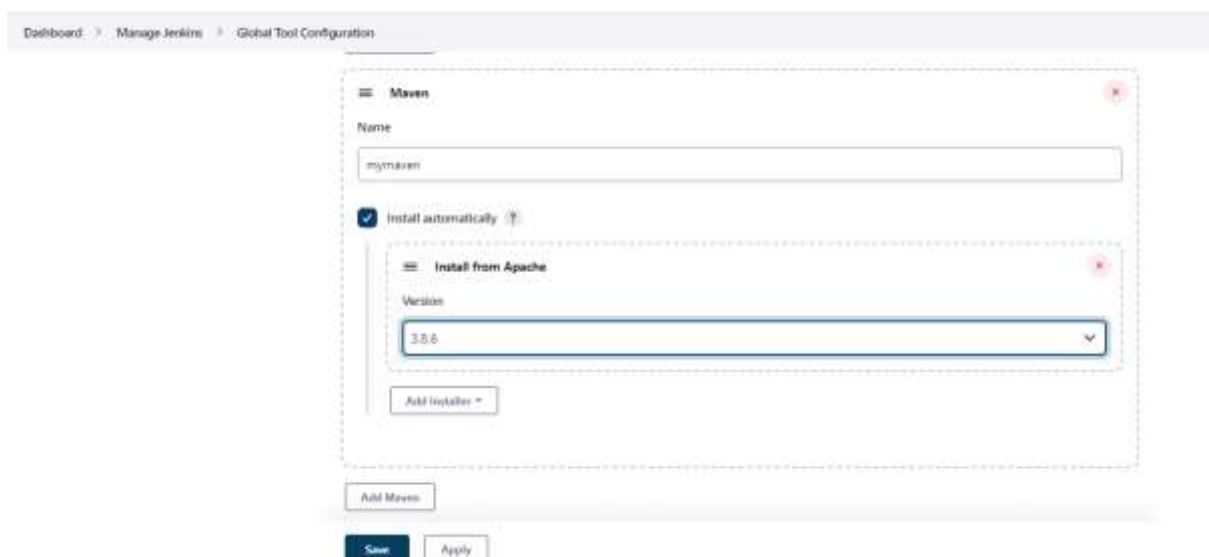
REST API Jenkins 2.361.2



Game of Life – GitHub Repository:



Configuration of Maven:



## Configuring Java JDK:

Dashboard > Manage Jenkins > Global Tool Configuration

JDK Name

☒ Install automatically

**Install Oracle Java SE Development Kit from the website**

Version

☒ I agree to the Java SE Development Kit License Agreement

**Installing JDK requires Oracle account. Please enter your username/password**

**Oracle Java SE 11+ is not available for business, commercial or production use without a commercial license. Public updates for Oracle Java SE 8 released after January 2019 will not be available for business, commercial or production use without a commercial license.**

[Oracle Java SE Licensing FAQ](#)

**Save** **Apply**

## Adding Source Code:

Dashboard > job1 >

**Configuration**

General

**Source Code Management**

Build Triggers

Build Environment

Build Steps

Post-build Actions

**Git**

Repositories

Repository URL

**Please enter Git repository.**

Credentials

**+ Add**

**Advanced...**

**Save** **Apply**

## Build Steps:

Dashboard > job1 >

**Configuration**

General

Source Code Management

Build Triggers

Build Environment

**Build Steps**

Post-build Actions

**Advanced...**

**Invoke top-level Maven targets**

Maven Version

Goals

**Advanced...**

**Add build step +**

**Post-build Actions**

**Save** **Apply**



## Successful Build:

```
Dashboard > job1 > #1

[INFO] --- jacoco-agent:0.7.2.201409121544-runtime:jar=destFile=C:\ProgramData\Jenkins\jenkins\workspace\job1\gameOfLife-web\target\jacoco-exe
[INFO]
[INFO] --- maven-compiler-plugin:3.1:compile (default-compile) @ gameOfLife-web ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 3 source files to C:\ProgramData\Jenkins\jenkins\workspace\job1\gameOfLife-web\target\classes
[INFO]
[INFO] -----
[INFO] Reactor Summary for gameOfLife 1.0-SNAPSHOT:
[INFO]
[INFO] gameOfLife ..... SUCCESS [01:30 min]
[INFO] gameOfLife-build ..... SUCCESS [ 41.333 s]
[INFO] gameOfLife-core ..... SUCCESS [ 1.853 s]
[INFO] gameOfLife-web ..... SUCCESS [ 11.481 s]
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 02:04 min
[INFO] Finished at: 2022-10-27T17:36:53+00:00
[INFO]
Finished: SUCCESS
```

REST API Jenkins 2.361.2

## Pom.xml file:

Code Issues 27 Pull requests 30 Actions Projects Wiki Security Insights

master 42 branches 163 tags Go to file Add file + Code + About

wakaleo Merge pull request #135 from GeraldScott/master 433 commits on 27 Jan 2018

gameOfLife-acceptance-tests	move Selenium Dockerfile	6 years ago
gameOfLife-build	Updated version	10 years ago
gameOfLife-core	Added function explanations for grid, cell, and related classes	6 years ago
gameOfLife-deploy	Tidied up code	11 years ago
gameOfLife-web	Updated dependencies	5 years ago
gitignore	Moved web tests into a separate module	11 years ago
README.markdown	Edit for clarity	5 years ago
infinitest.filters	First commit	12 years ago
pom.xml	Sevrenty version 1.8.3 missing from Maven repo so changed it to 1.8.4	5 years ago

README.markdown

2 contributors

400 lines (485 sloc) 21.8 KB Run Blame

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apa
3 <modelVersion>4.0.0</modelVersion>
4 <groupId>com.wakaleo.gameOfLife</groupId>
5 <artifactId>gameOfLife</artifactId>
6 <version>1.0-SNAPSHOT</version>
7 <packaging>war</packaging>
8 <name>gameOfLife</name>
9 <url>https://github.com/wakaleo/game-of-life</url>
10 <properties>
11 <build.number>SNAPSHOT</build.number>
12 <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
13 <sevnt.version>1.8.4</sevnt.version>
14 <cobertura.version>1.8.4</cobertura.version>
15 <!-- A workaround for a bug in PWD -->
16 <sourceId>1.0</sourceId>
17 <targetId>1.0</targetId>
18 <github.account>wakaleo</github.account>
19 <thucydides.version>1.8.4</thucydides.version>
20 <jelastic.context>gameOfLife</jelastic.context>
21 <jelastic.environment>wakaleo</jelastic.environment>
22 </properties>
23
24 <scm>
25 <connection>git@github.com:github.account:game-of-life.git</connection>
```

```

30     <jenkins:contextGameOfLife/>jenkins.context</jenkins:context>
31     <jenkins:environment/>jenkins.environment</jenkins:environment>
32   </properties>
33
34   <view>
35     <connection><git/>git@github.com:${github.account}/game-of-life.git/</connection>
36     <developerConnection><git/>git@github.com:${github.account}/game-of-life.git/</developerConnection>
37     <url><git/>git@github.com:${github.account}/game-of-life.git/</url>
38   </view>
39   <build>
40     <plugins>
41       <artifactDevelopment-compiler-plugin/>artifactDevelopment-compiler-plugin</artifactDevelopment-compiler-plugin>
42       <artifactDevelopment-compiler-plugin/>artifactDevelopment-compiler-plugin</artifactDevelopment-compiler-plugin>
43     </plugins>
44     <configuration>
45       <source/>1.7/</source>
46       <target/>1.7/</target>
47     </configuration>
48   </build>
49   <plugins>
50     <artifactDevelopment-compiler-plugin/>artifactDevelopment-compiler-plugin</artifactDevelopment-compiler-plugin>
51     <artifactDevelopment-compiler-plugin/>artifactDevelopment-compiler-plugin</artifactDevelopment-compiler-plugin>
52     <configuration>
53       <source/>1.7/</source>
54       <target/>1.7/</target>
55     </configuration>
56   </plugins>
57   <configuration>
58     <source/>1.7/</source>
59     <target/>1.7/</target>
60   </configuration>
61   <configuration>
62     <source/>1.7/</source>
63     <target/>1.7/</target>
64   </configuration>
65   <configuration>
66     <source/>1.7/</source>
67     <target/>1.7/</target>
68   </configuration>
69   <configuration>
70     <source/>1.7/</source>
71     <target/>1.7/</target>
72   </configuration>
73   <configuration>
74     <source/>1.7/</source>
75     <target/>1.7/</target>
76   </configuration>
77   <configuration>
78     <source/>1.7/</source>
79     <target/>1.7/</target>
80   </configuration>
81   <configuration>
82     <source/>1.7/</source>
83     <target/>1.7/</target>
84   </configuration>
85   <configuration>
86     <source/>1.7/</source>
87     <target/>1.7/</target>
88   </configuration>
89   <configuration>
90     <source/>1.7/</source>
91     <target/>1.7/</target>
92   </configuration>
93   <configuration>
94     <source/>1.7/</source>
95     <target/>1.7/</target>
96   </configuration>
97   <configuration>
98     <source/>1.7/</source>
99     <target/>1.7/</target>
100  </configuration>

```

## Conclusion:

In this experiment, we have learned about the installation of Jenkins. Jenkins which is an automation server was installed, configured, and used to test simple jobs of freestyle type, one to execute a windows batch command and the other to use maven to integrate a project Game of Life.

We have achieved LO3 from this experiment.

We have also achieved Program Outcomes PO1, PO2, PO3, PO4, PO5, PO12.