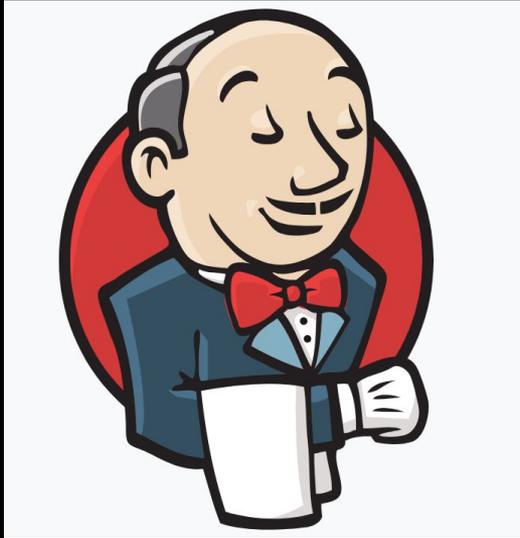


# Software Project Management - Laboratory

Lecture n° 13  
A.Y. 2021-2022

Prof. Fabrizio Fornari

# Jenkins

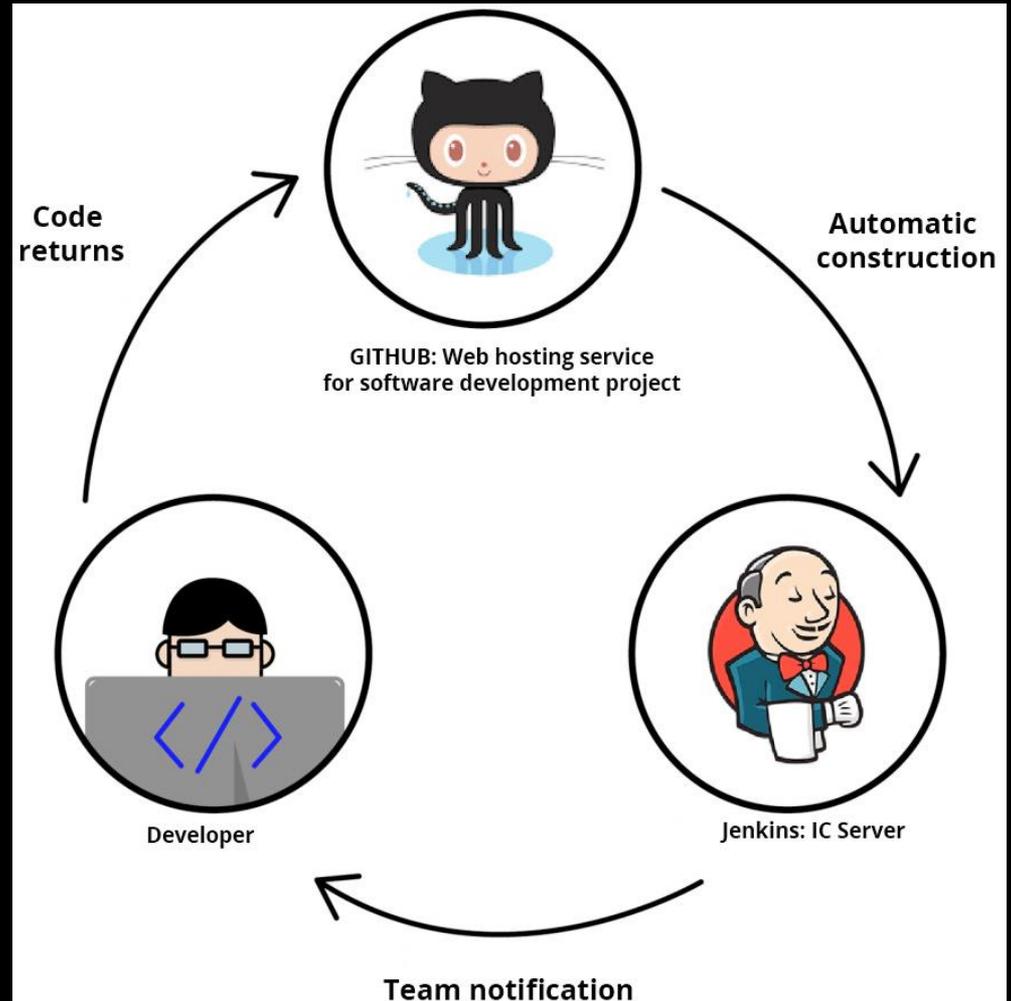


Jenkins is used to build and test your product continuously, so developers can continuously integrate changes into the build.

<https://jenkins.io/>

# Continuous Integration with Jenkins

Jenkins triggers a build upon every commit to the source code repository, typically to a development branch.



# Jenkins on Tomcat

Download the latest .war file

<https://www.jenkins.io/download/>

Deploy the .war on Tomcat

<https://www.jenkins.io/doc/book/installing/#war-file>



# Create Job

## 1 Create a new Job

**Enter an item name**

*» 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.

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

 **Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

## 2 Link it with your GitHub Repository

**Source Code Management**

None  
 Git

Repositories

Repository URL

Credentials

# Run a Job

Run a Job

Jenkins > SPM2020 >

- Back to Dashboard
- Status
- Changes
- Workspace
- Build Now**
- Configure
- Delete Project
- Rename

Build History [trend](#) ^

find

### Project SPM2020

This job has been created for the SPM2020 course

- Workspace
- Recent Changes

### Permalinks

Run a Job

S	W	Name ↓	Last Success	Last Failure	Last Duration	
		SPM2020	2 min 7 sec - #5	4 min 36 sec - #3	2.4 sec	

Icon: S M L

Legend

Atom feed for all

Atom feed for failures

Atom feed for just latest builds

# Configure JDK

Manage Jenkins > Global Tool Configuration

**JDK**

JDK installations Add JDK

JDK

Name

JAVA\_HOME

Install automatically ?

Delete JDK

Add JDK

List of JDK installations on this system

# Configure JDK & Maven

Manage Jenkins --> Global Tool Configuration

Maven

Maven installations

**Add Maven**

Maven

Name

MAVEN\_HOME

Install automatically

**Delete Maven**

Or

Tell Jenkins to  
install a version

Maven

Maven installations

**Add Maven**

Maven

Name

Install automatically

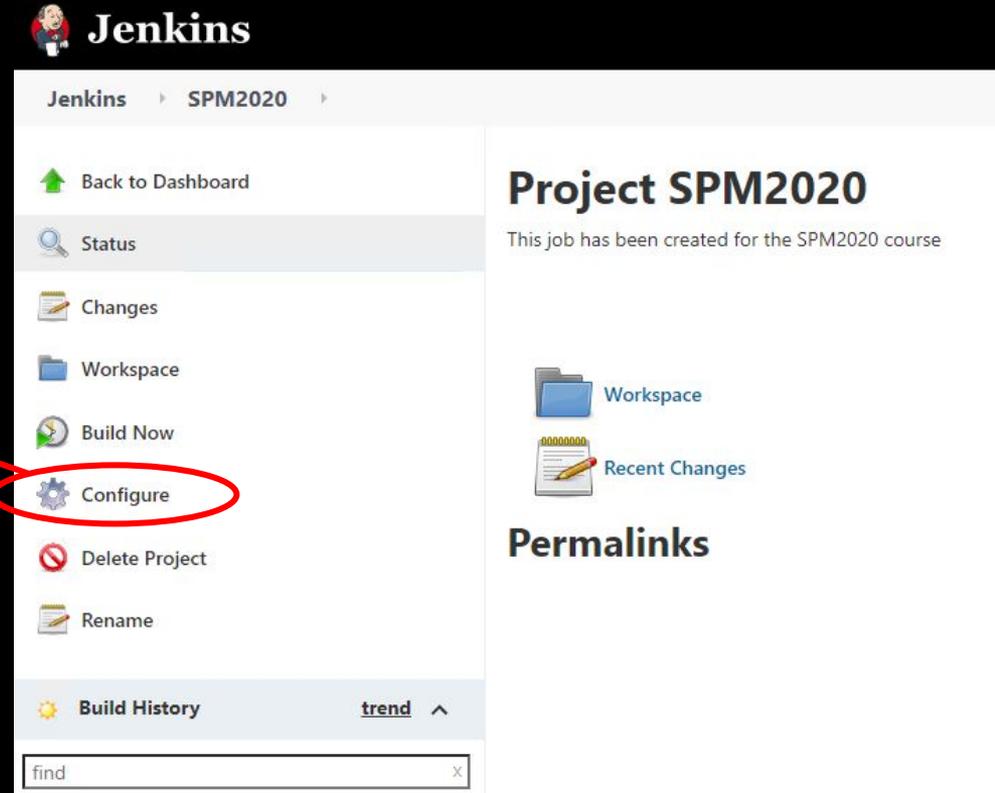
**Install from Apache**

Version

**Delete Installer**

# Configure a Job

Configure for  
changing settings



Jenkins

Jenkins SPM2020

- Back to Dashboard
- Status
- Changes
- Workspace
- Build Now
- Configure**
- Delete Project
- Rename

Build History trend ^

find x

## Project SPM2020

This job has been created for the SPM2020 course

- Workspace
- Recent Changes

## Permalinks

# Configure Build Environment

Build

**Invoke top-level Maven targets** ✖ ?

Maven Version

Goals

# Schedule a build

(Build Triggers → Build periodically)

Jenkins uses a cron expression, and the different fields are:

1. MINUTES Minutes in one hour (0-59)
2. HOURS Hours in one day (0-23)
3. DAYMONTH Day in a month (1-31)
4. MONTH Month in a year (1-12)
5. DAYWEEK Day of the week (0-7) where 0 and 7 are sunday

MINUTES	HOURS	DAYMONTH	MONTH	DAYWEEK
<code>*/5</code>	<code>*</code>	<code>*</code>	<code>*</code>	<code>*</code>

To schedule a build every 5 minutes : `*/5 * * * *`

To schedule a build every day at 8h00, this will do the job : `0 8 * * *`

# Email Notifications

Manage Jenkins → Configure System

**E-mail Notification**

SMTP server

Default user e-mail suffix

Use SMTP Authentication

User Name

Password

Use SSL

Use TLS

SMTP Port

Reply-To Address

Charset

Job Name --> Configure

**Post-build Actions**

**E-mail Notification** ✕ ?

Recipients

Whitespace-separated list of recipient addresses. May reference build parameters like \$PARAM. E-mail will be sent when a build fails, becomes unstable or returns to stable.

Send e-mail for every unstable build

Send separate e-mails to individuals who broke the build

Add post-build action ▾

# Email Notifications

If you are using gmail, allow third party application to login from <https://myaccount.google.com/security>

## Less secure app access

Your account is vulnerable because you allow apps and devices that use less secure sign-in technology to access your account. To keep your account secure, Google will automatically turn this setting OFF if it's not being used.

 On

[Turn off access \(recommended\)](#)



## Test the configuration

Test configuration by sending test e-mail

Test e-mail recipient

fabrizio.fornari@unicam.it

Email was successfully sent

Test configuration

# Email Notifications

## Post-build Actions

### E-mail Notification X ?

Recipients

Whitespace-separated list of recipient addresses. May reference build parameters like \$PARAM. E-mail will be sent when a build fails, becomes unstable or returns to stable.

Send e-mail for every unstable build

Send separate e-mails to individuals who broke the build ?

Test the configuration

# Email Notifications

<input type="checkbox"/>	☆	»	Indirizzo non ancor.	Build failed in Jenkins: SPM2020 #29 - See < <a href="http://localhost:8080/jenkins/job/SPM2020/29/display/redirect?page=changes">http://localhost:8080/jenkins/job/SPM2020/29/display/redirect?page=changes</a> > Chang
<input type="checkbox"/>	☆	»	<b>Indirizzo non ancor.</b>	<b>Build failed in Jenkins: SPM2020 #28</b> - See < <a href="http://localhost:8080/jenkins/job/SPM2020/28/display/redirect">http://localhost:8080/jenkins/job/SPM2020/28/display/redirect</a> > Changes: -----
<input type="checkbox"/>	☆	»	Indirizzo non ancor.	Build failed in Jenkins: SPM2020 #27 - See < <a href="http://localhost:8080/jenkins/job/SPM2020/27/display/redirect?page=changes">http://localhost:8080/jenkins/job/SPM2020/27/display/redirect?page=changes</a> > Chan
<input type="checkbox"/>	☆	»	Indirizzo non ancor.	Build failed in Jenkins: SPM2020 #26 - See < <a href="http://localhost:8080/jenkins/job/SPM2020/26/display/redirect">http://localhost:8080/jenkins/job/SPM2020/26/display/redirect</a> > Changes: -----
<input type="checkbox"/>	☆	»	Indirizzo non ancor.	Build failed in Jenkins: SPM2020 #25 - See < <a href="http://localhost:8080/jenkins/job/SPM2020/25/display/redirect">http://localhost:8080/jenkins/job/SPM2020/25/display/redirect</a> > Changes: -----
<input type="checkbox"/>	☆	»	Indirizzo non ancor.	Build failed in Jenkins: SPM2020 #24 - See < <a href="http://localhost:8080/jenkins/job/SPM2020/24/display/redirect">http://localhost:8080/jenkins/job/SPM2020/24/display/redirect</a> > Changes: -----
<input type="checkbox"/>	☆	»	Indirizzo non ancor.	Build failed in Jenkins: SPM2020 #23 - See < <a href="http://localhost:8080/jenkins/job/SPM2020/23/display/redirect">http://localhost:8080/jenkins/job/SPM2020/23/display/redirect</a> > Changes: -----
<input type="checkbox"/>	☆	»	Indirizzo non ancor.	Build failed in Jenkins: SPM2020 #22 - See < <a href="http://localhost:8080/jenkins/job/SPM2020/22/display/redirect?page=changes">http://localhost:8080/jenkins/job/SPM2020/22/display/redirect?page=changes</a> > Chan
<input type="checkbox"/>	☆	»	<b>Indirizzo non ancor.</b>	<b>Build failed in Jenkins: SPM2020 #21</b> - See < <a href="http://localhost:8080/jenkins/job/SPM2020/21/display/redirect?page=changes">http://localhost:8080/jenkins/job/SPM2020/21/display/redirect?page=changes</a> > Chan
<input type="checkbox"/>	☆	»	Indirizzo non ancor.	Build failed in Jenkins: SPM2020 #20 - See < <a href="http://localhost:8080/jenkins/job/SPM2020/20/display/redirect">http://localhost:8080/jenkins/job/SPM2020/20/display/redirect</a> > Changes: -----
<input type="checkbox"/>	☆	»	Indirizzo non ancor.	Test email #3 - This is test email #3 sent from Jenkins

To be removed: <https://mail.google.com/mail/u/0/#inbox/FMfcgxwKjdtDKSXnZkfdzMGGxxBRtLKj>

# Build failed in Jenkins: SPM2021

See

<http://proslabtest:8080/jenkins/job/SPM2021/4/display/redirect?page=changes>

Changes:

[fabrizio.fornari] added a failing test

Started by user Fabrizio Fornari

Running as SYSTEM

Building in workspace <http://proslabtest:8080/jenkins/job/SPM2021/ws/>

...

[INFO] -----< pros.unicam.it:spm2021 >-----

[INFO] Building webapp Maven Webapp 0.0.1-SNAPSHOT

[INFO] -----[ war ]-----

[INFO] --- maven-clean-plugin:3.1.0:clean (default-clean) @ spm2021 ---

[INFO] Deleting <http://proslabtest:8080/jenkins/job/SPM2021/ws/target>

...

[INFO] --- maven-compiler-plugin:3.8.0:testCompile (default-testCompile) @ spm2021 ---

[INFO] Changes detected - recompiling the module!

[INFO] Compiling 7 source files to

<http://proslabtest:8080/jenkins/job/SPM2021/ws/target/test-classes>

...

...

[INFO] -----

[INFO] T E S T S

[INFO] -----

[INFO] Running pros.unicam.spm2021.practice.JUnit.EmptyTest

[WARNING] Tests run: 1, Failures: 0, Errors: 0, Skipped: 1, Time elapsed: 0.05 s - in pros.unicam.spm2021.practice.JUnit.EmptyTest

[INFO] Running pros.unicam.spm2021.practice.JUnit.HelloWorldTest  
Nov 22, 2021 11:54:31 AM

pros.unicam.spm2021.practice.JUnit.HelloWorldTest setUpBeforeClass

[ERROR] Tests run: 12, Failures: 1, Errors: 0, Skipped: 3, Time elapsed: 0.185 s <<< FAILURE! - in

pros.unicam.spm2021.practice.JUnit.HelloWorldTest

[ERROR] testMain Time elapsed: 0.019 s <<< FAILURE!  
org.opentest4j.AssertionFailedError: Not yet implemented

at

pros.unicam.spm2021.practice.JUnit.HelloWorldTest.testMain(HelloWorldTest.java:111)

[INFO]

[INFO] -----

[INFO] BUILD FAILURE

[INFO] -----

[INFO] Total time: 9.285 s

[INFO] Finished at: 2021-11-22T11:54:32+01:00

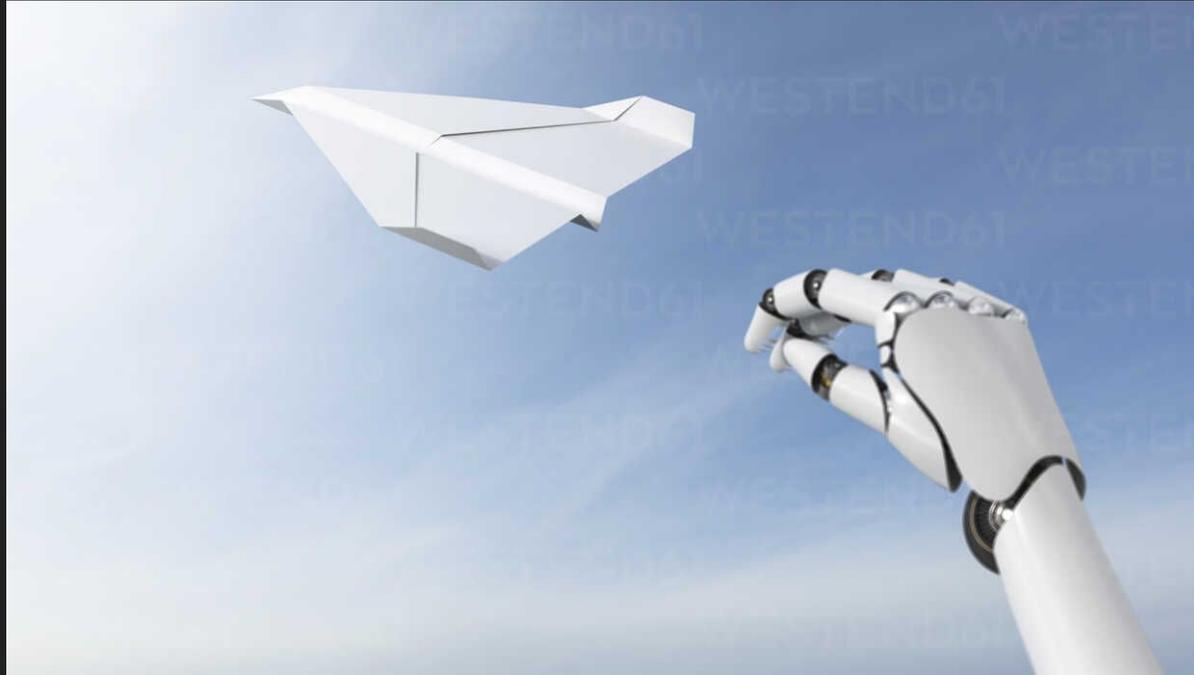
[INFO] -----

# Fix the error

Email: Jenkins build is back to normal : SPM2021 #5

See <<http://proslabtest:8080/jenkins/job/SPM2021/5/display/redirect?page=changes>>

# Automatic Deploy



# Apache Maven

Apache Maven is an open source, standards-based project management framework that simplifies the building, testing, reporting, and packaging of projects.



<http://maven.apache.org/>

# Apache Tomcat

The Apache Tomcat® software is an open source implementation of the Java Servlet, JavaServer Pages, Java Expression Language and Java WebSocket technologies.

**Download Tomcat**

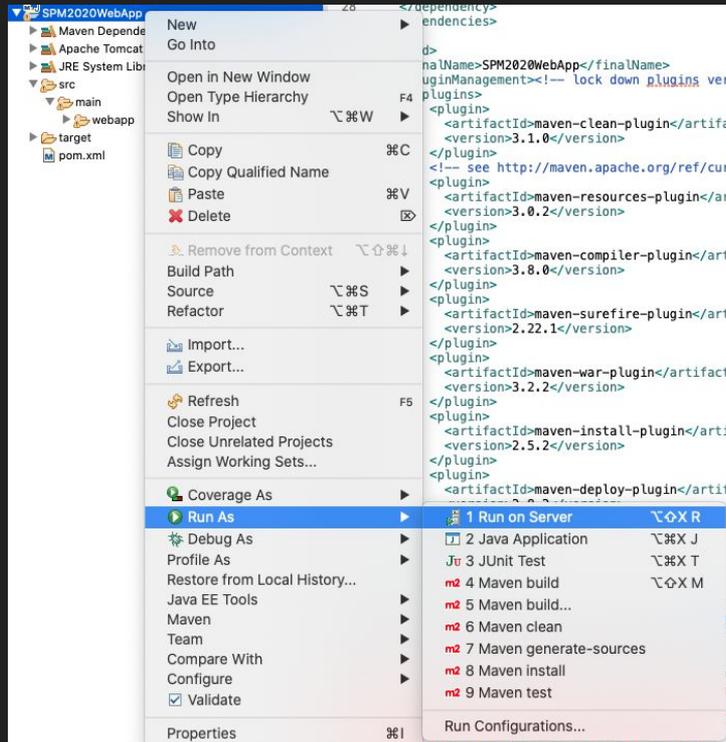
<https://tomcat.apache.org/download-90.cgi>



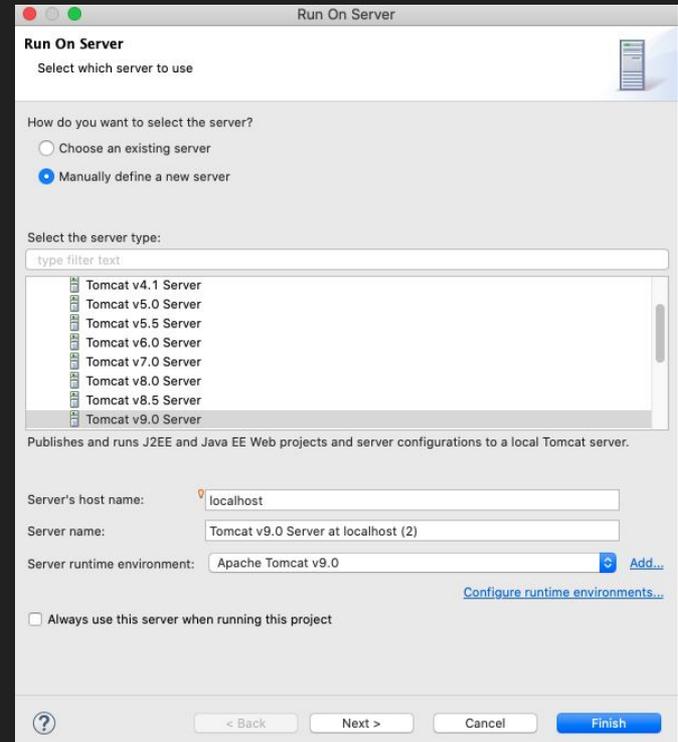
<http://tomcat.apache.org/>

# Run Your Application

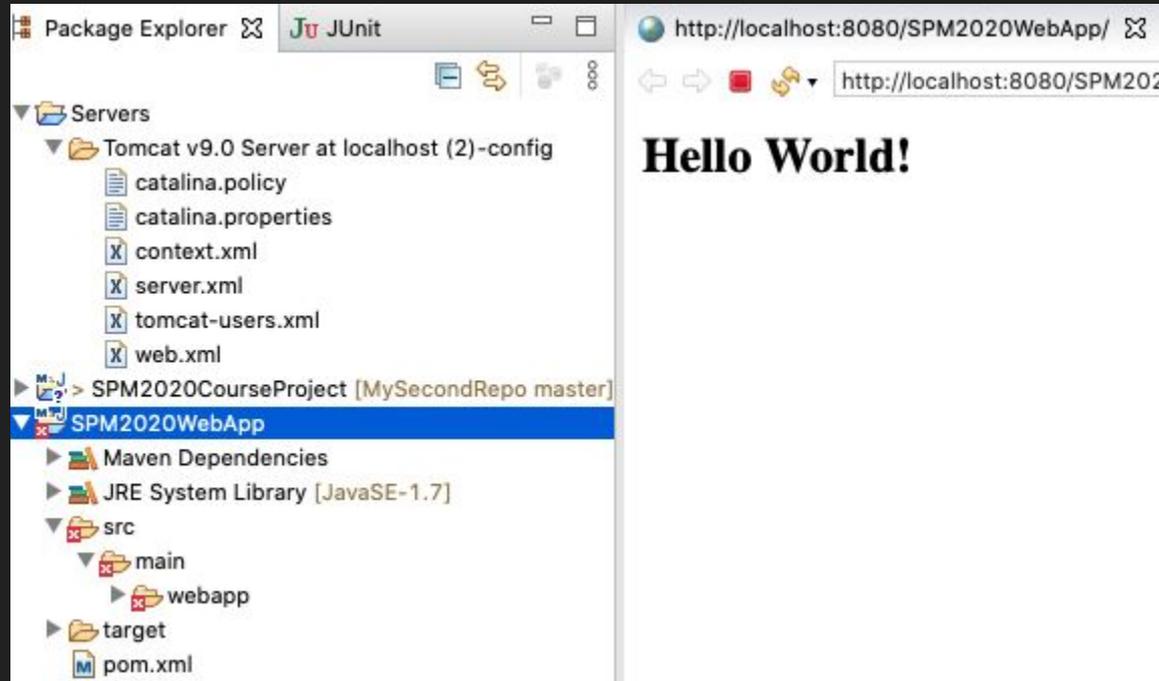
## Run On Server



## Pick the version you installed



# Run Your Application





# Automatically deploy a WAR on Tomcat with Maven

To perform a Maven Tomcat deploy of a WAR file you must first set up a user in Tomcat with the appropriate rights. You can do this with an edit of the tomcat-users.xml file, which can be found in Tomcat's conf sub-directory. Add the following entry inside the tomcat-users tag:

```
<!-- User to deploy WAR file to Tomcat from Maven -->  
<user username="war-deployer" password="maven-tomcat-plugin"  
      roles="manager-gui, manager-script, manager-jmx" />
```

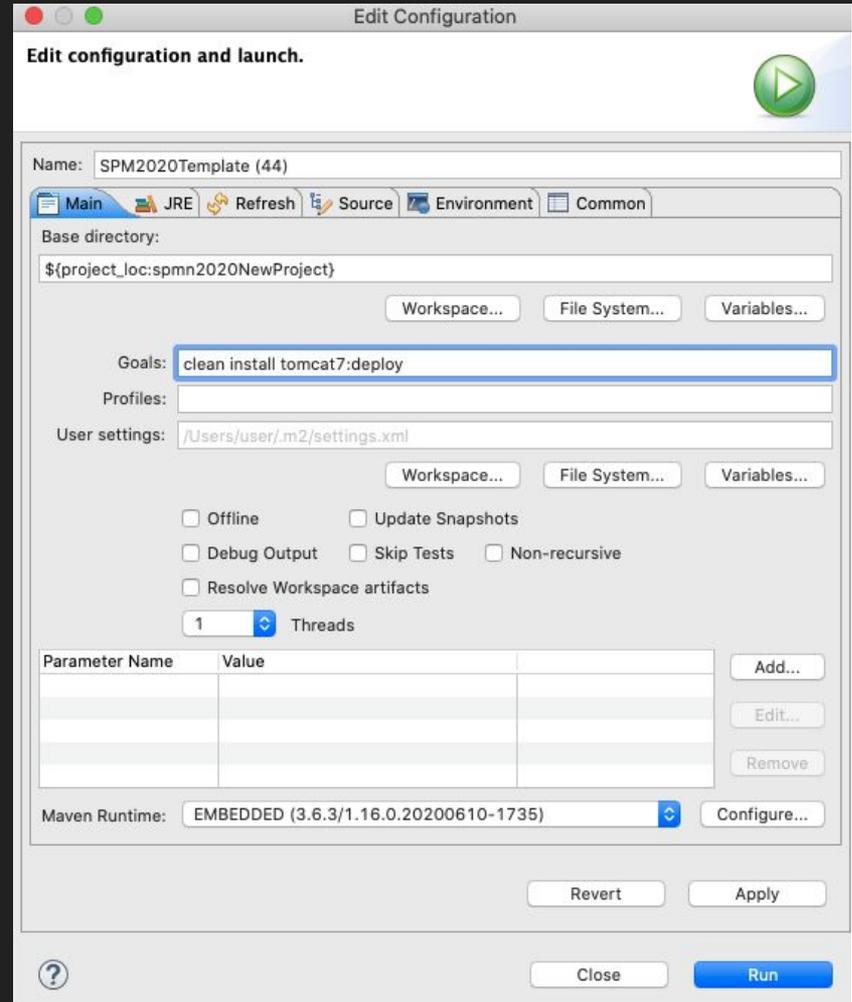
Save the tomcat-users.xml file and restart the server to have the changes take effect.

# Automatically deploy a WAR on Tomcat with Maven

```
<!-- Configure the Tomcat Maven plugin -->
  <plugin>
    <groupId>org.apache.tomcat.maven</groupId>
    <artifactId>tomcat7-maven-plugin</artifactId>
    <version>2.2</version>
    <configuration>
      <!-- Use tomcat9 user defined credentials.
      Usually you would place them under the Maven
      folder .settings.xml telling eclipse to read them from there
      by just using adding <server>tomcat9</server>-->
      <username>war-deployer</username>
      <password>maven-tomcat-plugin</password>
      <update>true</update>
      <!-- <url>http://localhost:8080/manager/text</url>-->
    </configuration>
  </plugin>
```

# Maven Goals

Goals: `clean install tomcat7:deploy`



# Automatically Deployed WebApp



## Tomcat Web Application Manager

Message: OK

### Manager

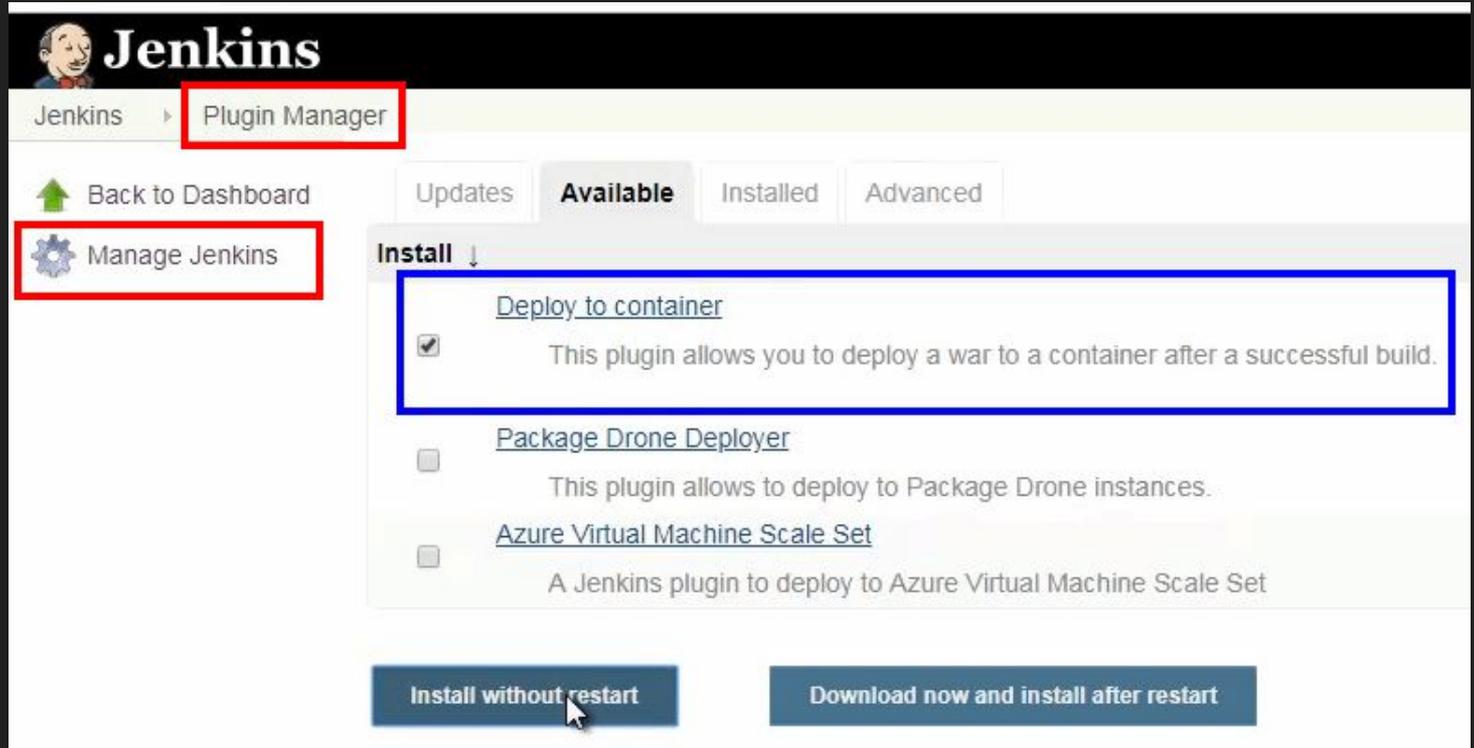
List Applications      HTML Manager Help      Manager Help      Server Status

### Applications

Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes
/docs	None specified	Tomcat Documentation	true	0	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes
/examples	None specified	Servlet and JSP Examples	true	0	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes
/host-manager	None specified	Tomcat Host Manager Application	true	0	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes
/jenkins	None specified	Jenkins v2.321	true	0	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes
/manager	None specified	Tomcat Manager Application	true	2	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes
/spm2021	None specified	Archetype Created Web Application	true	0	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes

# Automatically deploy a WAR on Tomcat with Jenkins

Install the Deploy to container plugin



The screenshot displays the Jenkins Plugin Manager interface. At the top, the Jenkins logo and name are visible. Below the logo, the breadcrumb navigation shows 'Jenkins' and 'Plugin Manager', with 'Plugin Manager' highlighted in a red box. On the left sidebar, there are two links: 'Back to Dashboard' with a green arrow icon and 'Manage Jenkins' with a gear icon, the latter being highlighted in a red box. The main content area has three tabs: 'Updates', 'Available', and 'Installed', with 'Available' selected. Below the tabs, there is an 'Install' dropdown menu with a downward arrow. The dropdown is open, showing three options: 'Deploy to container' (checked), 'Package Drone Deployer', and 'Azure Virtual Machine Scale Set'. The 'Deploy to container' option is highlighted with a blue box. Below the dropdown, there are two buttons: 'Install without restart' and 'Download now and install after restart'.

Jenkins

Jenkins > Plugin Manager

Back to Dashboard

Manage Jenkins

Updates Available Installed Advanced

Install ↓

- [Deploy to container](#)  
This plugin allows you to deploy a war to a container after a successful build.
- [Package Drone Deployer](#)  
This plugin allows to deploy to Package Drone instances.
- [Azure Virtual Machine Scale Set](#)  
A Jenkins plugin to deploy to Azure Virtual Machine Scale Set

Install without restart

Download now and install after restart

# Automatically deploy a WAR on Tomcat with Jenkins

### Build

- Invoke top-level Maven targets
- Aggregate downstream test results
- Archive the artifacts
- Build other projects
- Publish JUnit test result report
- Publish Javadoc
- Record fingerprints of files to track usage
- Use publishers from another project
- Git Publisher
- Deploy war/ear to a container**
- E-mail Notification
- Trigger parameterized build on other projects

**Add post-build action** ▾

### Post-build Actions

#### Deploy war/ear to a container

WAR/EAR files:

Context path:

#### Containers

##### Tomcat 9.x Remote

Credentials:  **Add** ▾

Tomcat URL:

**Add Container** ▾

Deploy on failure

### Add Credentials

Domain:

Kind:

Scope:

Username:

Password:

ID:

Description:

**Add** **Cancel**

# Automatically deploy a WAR on Tomcat with Jenkins



## Tomcat Web Application Manager

Message: OK

### Manager

List Applications      HTML Manager Help      Manager Help      Server Status

### Applications

Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	None specified	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	None specified	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	None specified	Tomcat Host Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/jenkins	None specified	Jenkins v2.321	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/manager	None specified	Tomcat Manager Application	true	2	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/spm2021	None specified	Archetype Created Web Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

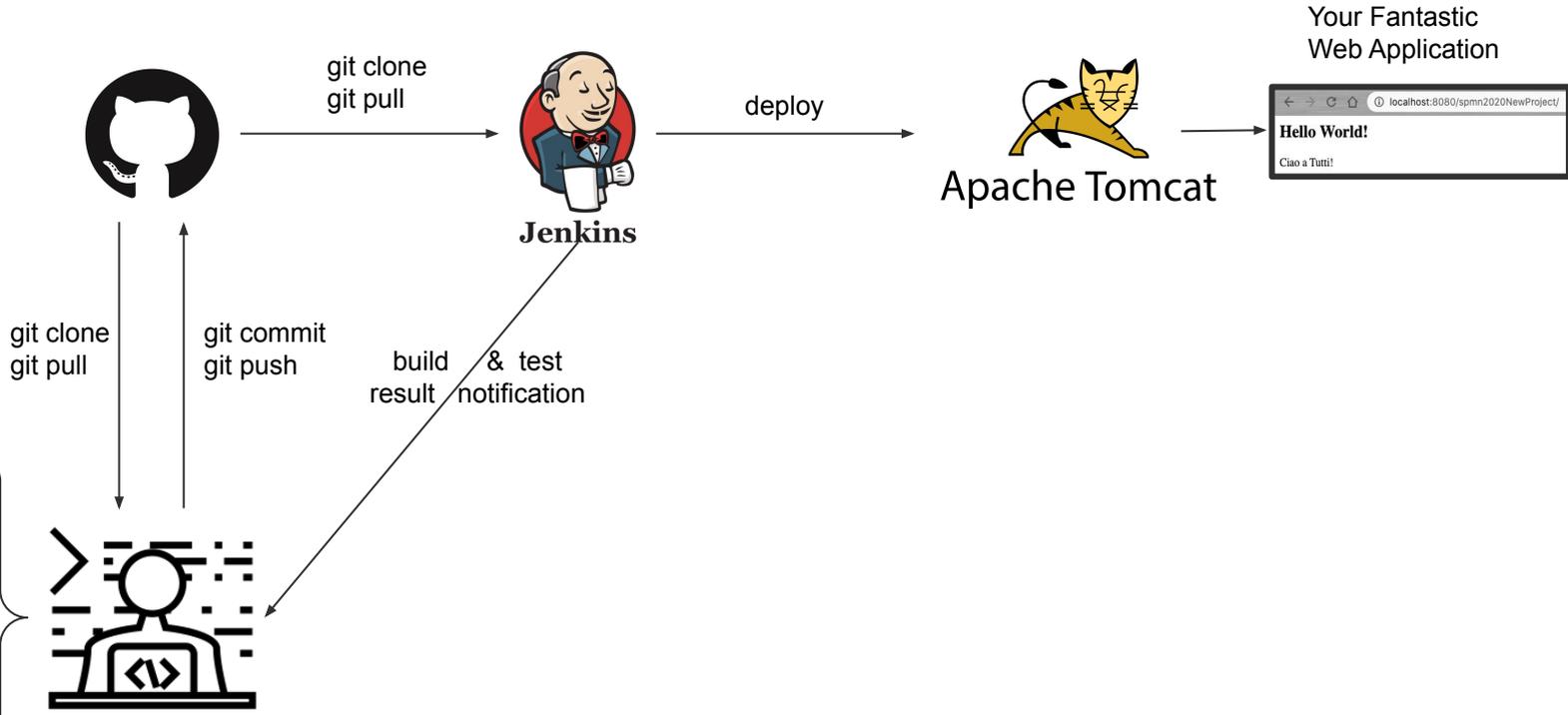
localhost:8080/spm2021/

Hello World!

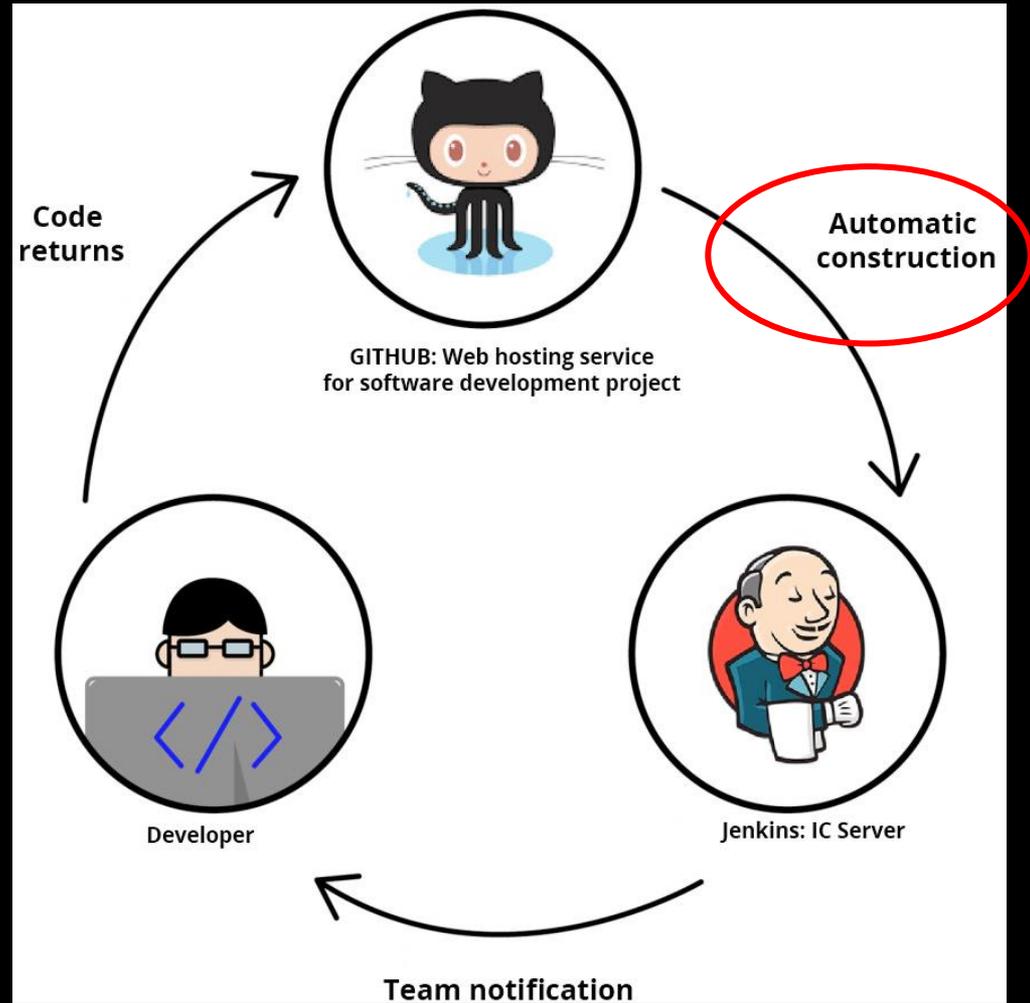
Today is the 25th of November

To Do: Change index.jsp

# What is Missing?



# How to connect Github and Jenkins?



# Webhook

- Webhooks allow you to build or set up GitHub Apps which subscribe to certain events on GitHub.com.
- When one of those events is triggered, we'll send a HTTP POST payload to the webhook's configured URL.
- Webhooks can be used to update an external issue tracker, trigger CI builds, update a backup mirror, or even deploy to your production server.

<https://developer.github.com/webhooks/>

[Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)[FabrizioFornari / spm2021Template](#) Public[Unwatch](#)

1

[Star](#)

0

[Code](#)[Issues](#)[Pull requests](#)[Actions](#)[Projects](#)[Wiki](#)[Security](#)[Insights](#)[Settings](#)

Options

Manage access

Security &amp; analysis

Branches

Webhooks

Notifications

Integrations

Deploy keys

Autolink references

Actions

## Webhooks / Manage webhook

Settings

Recent Deliveries

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in [our developer documentation](#).

**Payload URL \*****Content type****Secret**

<http://proslabtest.unicam.it:80/jenkins/github-webhook/>

# Jenkins GitHub hook trigger

## Build Triggers

- Trigger builds remotely (e.g., from scripts)
- Build periodically
- Build after other projects are built
- Poll SCM
- GitHub hook trigger for GITScm polling



If Jenkins will receive PUSH GitHub hook from repo defined in Git SCM section it will trigger Git SCM polling logic. So polling logic in fact belongs to Git SCM.

(from [GitHub plugin](#))

# Enable Selenium Tests

Is everything ok?

# Environments

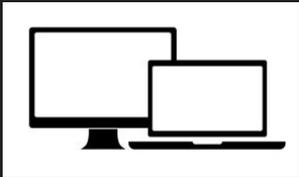
NOTE: Referred also as Development, Testing, Acceptance and Production (DTAP)

## Development

Development and Unit testing for the developed feature are done on the individual developer's laptop or desktop system with a proper version control system in place.

For web based applications, at a minimum, it requires:

- The same web server used in production.
- The same database used in production.
- The same language being used in production.



## Build/Test

The build/test server should automatically check out all the code, refresh the database and then execute tests.

All unit tests are run, then integration and regression testing are performed to make sure that all the pieces fit together and nothing previously working was broken.



## Staging

The staging site is used to assemble, test and review new versions of a web app before it goes into production.

It is often used to present the client with the final project for them to perform **Acceptance testing**



## Production

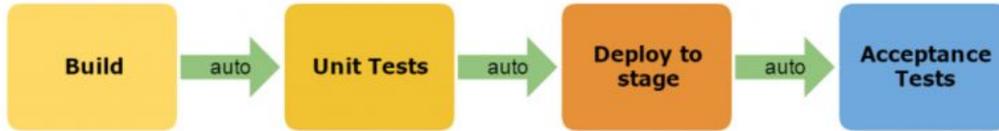
The accepted product, is deployed to a Production environment, making it available to all users of the system.



# The Product Pipeline

© Ankesh K,  
www.linuxnix.com

## Continuous Integration



## Continuous Delivery



## Continuous Deployment

