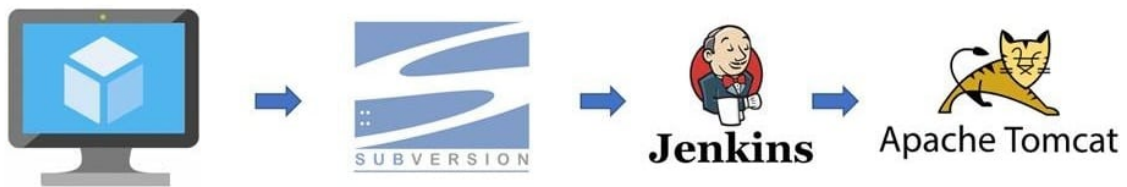


Instalación de Jenkins y Tomcat



José Antonio Castillejo Lobato



1. Lo primero que debemos de hacer es entrar en Jenkins irnos a la pestaña Manage Jenkins

Manage Jenkins

It appears that your reverse proxy set up is broken. [More Info](#) [Dismiss](#)

New version of Jenkins (2.390) is available for [download](#) ([changelog](#)).

Building on the built-in node can be a security issue. You should set up distributed builds. See [the documentation](#). [Set up agent](#) [Set up cloud](#) [Dismiss](#)

System Configuration

- [Configure System](#)
Configure global settings and paths.
- [Global Tool Configuration](#)
Configure tools, their locations and automatic installers.

2. Una vez dentro de Manage Jenkins entramos en el apartado Manage Plugins

Manage Plugins

Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

3. Se nos mostrará la siguiente página y debemos de entrar en Available plugins

Dashboard > Manage Jenkins > Plugin Manager

Download progress

Preparation

- Checking internet connectivity
- Checking update center connectivity
- Success

Javadoc ✓ Success

JSch dependency ✓ Success

Maven Integration ✓ Success

Loading plugin extensions ✓ Success

→ [Go back to the top page](#)
(you can start using the installed plugins right away)

→ ☐ Restart Jenkins when installation is complete and no jobs are running



4. Buscamos en el buscador el plugin Deploy to container que nos va a servir a la hora de unir el servidor Tomcat con Jenkins y Maven Integration plugin para crear el proyecto, lo descargamos y reiniciamos Jenkins. Si no sabemos cómo reiniciarlo es tan fácil como añadir un restart al final de la url de Jenkins (localhost:8082/manage/restart)

Install	Name ↓	Released
<input checked="" type="checkbox"/>	Deploy to container 1.16 Artifact Uploaders This plugin allows you to deploy a war to a container after a successful build. Glassfish 3.x remote deployment	2 yr 3 mo ago

Install without restart

Download now and install after restart

Update information obtained: 22 hr ago

Check now

Name ↓	Enabled
Maven Integration plugin 3.20 This plugin provides a deep integration between Jenkins and Maven. It adds support for automatic triggers between projects depending on SNAPSHOTS as well as the automated configuration of various Jenkins publishers such as Junit. Report an issue with this plugin	<input checked="" type="checkbox"/>

5. Nos dirigimos al inicio y le damos a New Item para crear el proyecto

+ New Item

People


Build History




6. Creamos un proyecto Maven (si no tenemos el plugin Maven Integration plugin no nos aparecerá esta opción) y de damos un nombre y descripción

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, a 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.

7. Ahora vamos a configurar nuestro proyecto, lo primero que tenemos que configurar es añadir el repositorio git de nuestro proyecto, tenemos que activar la opción de Git, añadir la URL del repositorio y especificar la rama principal

Git ?

Repositories ?

Repository URL ?

Credentials ?

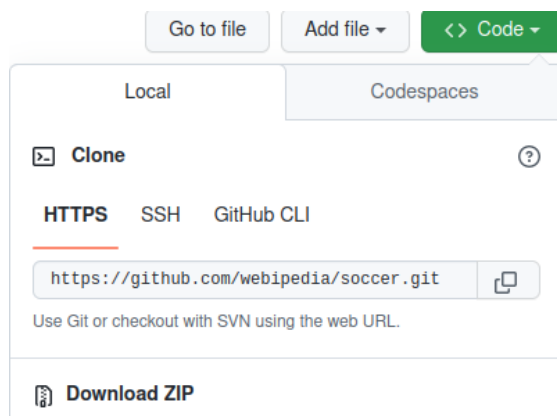
Advanced ▾

Branch Specifier (blank for 'any') ?

*/main

8. Si no sabemos obtener la URL de nuestro repositorio Git es tan fácil como dirigirnos al repositorio darle a Code y nos mostrará la URL

<> Code ▾



9. Una vez añadido el repositorio Git debemos de configurarlo de tal manera para que cuando se haga algún cambio lo compile y despliegue automáticamente, para ello tenemos que en el apartado **Build Triggers** activar el **Build whenever a SNAPSHOT dependency is built** y **Poll SCM** para que indicarle el intervalo de que Jenkins observe si hay modificaciones le añadimos en Schedule el código que queramos



Build Triggers

- ☒ Build whenever a SNAPSHOT dependency is built ?
- ☐ Schedule build when some upstream has no successful builds ?
- ☐ Trigger builds remotely (e.g., from scripts) ?
- ☐ Build after other projects are built ?
- ☐ Build periodically ?
- ☐ GitHub hook trigger for GITScm polling ?

☒ Poll SCM ?

Schedule ?

H/15 ****

10. Tenemos que ponerle un nombre al Maven e indicarle la versión y guardar los cambios

≡ Maven

Name

Mav

❗ Required

☒ Install automatically ?

≡ Install from Apache

Version

3.9.0

Add Installer ▾

11. Finalmente tenemos que dirigirnos a nuestro proyecto y darle a Build Now para que compile el proyecto



Status



Changes



Workspace



Build Now



Configure



Delete Maven project



Build #1 (Feb 24, 2023, 7:14:01 AM)



No changes.



Started by user [jose ant cast lob](#)



Revision: 9ddffa6e2029c506f256d9b2e69d60acf3d08392

Repository: <https://github.com/webipedia/soccer.git>

• refs/remotes/origin/master

Module Builds




soccer


10 sec


12. Para unir Tomcat con Jenkins nos hace falta el plugin anteriormente mencionado y nos dirigimos a la configuración del proyecto **Manage Jenkins/Configure**




 **Manage Jenkins**


☐ My Views

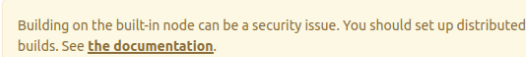
Build Queue 
10 builds in the queue.


Build Executor Status 

System Configuration

 **Configure System**
Configure global settings and paths.

 **Global Tool Configuration**
Configure tools, their locations and automatic installers.

 [Set up agent](#) [Set up cloud](#) [Dismiss](#)

 **Configure**

13. En Post-build Actions debemos de indicarle la siguiente opción

Post-build Actions


[Add post-build action](#) 

Save

Apply


Filter


- Aggregate downstream test results
- Archive the artifacts
- Build other projects
- Deploy artifacts to Maven repository
- Record fingerprints of files to track usage
- Git Publisher
- Deploy war/ear to a container**
- Editable Email Notification
- Set GitHub commit status (universal)
- Set build status on GitHub commit [deprecated]
- Delete workspace when build is done


[Add post-build action](#) 

14. Tenemos que en **WAR/EAR files** indicar la dirección de nuestro proyecto .war y en Context path el nombre del proyecto en Jenkins e indicamos la versión que utilizamos


Post-build Actions

 **Deploy war/ear to a container**

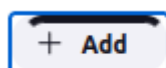
WAR/EAR files 

Context path 

Containers

 **Tomcat 8.x Remote**

15. Tenemos que añadir las credenciales pulsando en Add y añadiendo el usuario y la contraseña que tengamos en Tomcat y su URL





Username ?

Tomcat

☐ Treat username as secret ?

Password ?

.....

≡ **Tomcat 9.x Remote**

Credentials

Tomcat/*****

+ Add

Tomcat URL ?

https://localhost:8888/