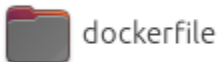


Este documento explica como configurar Jenkins para hacer un pipeline de CI del siguiente proyecto de prueba: https://github.com/panizolledotangel/c_devops_test.git. El repositorio es público y ya viene con un Jenkinsfile que compila el código, pasa los tests unitarios y se comunica con sonarcloud para medir la calidad del código.

Configurar jenkins

1. El entorno docker se encuentra en el repositorio: https://github.com/panizolledotangel/docker_jenkins_sonar_c y tiene esta estructura:



2. Compilar la imagen de docker del jenkins con los requisitos para compilar el proyecto C. (docker compose build jenkins)
3. Arrancar el contenedor docker (docker compose up)
4. Si es la primera vez que se ejecuta el contenedor es necesario configurar Jenkins.
5. Copiar la contraseña admin aleatoria del log del terminal donde de ejecutó el docker compose up.

```
Jenkins initial setup is required. An admin user has been created and a password generated.  
Please use the following password to proceed to installation:
```

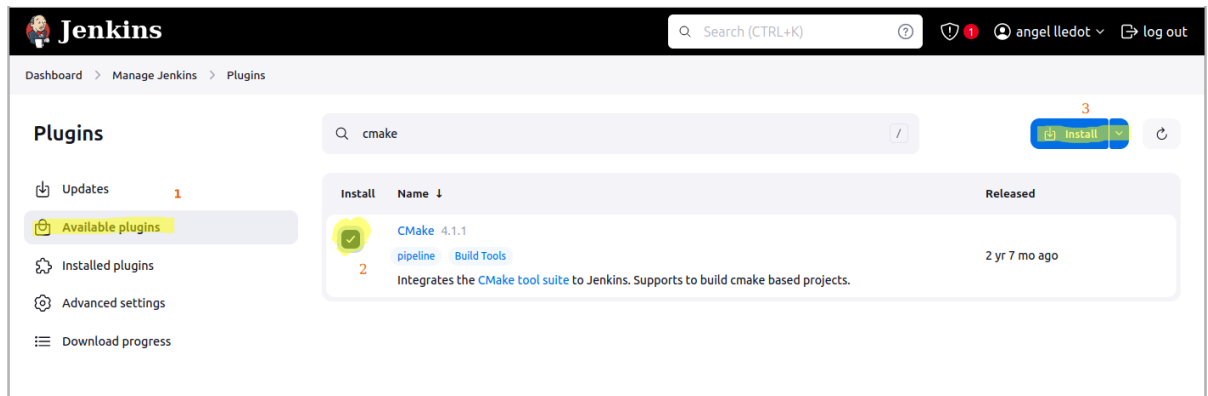
```
86c4015ab13743baa140ee7286db3295
```

```
This may also be found at: /var/jenkins_home/secrets/initialAdminPassword
```

6. Abrir un navegador en la url <http://localhost:8888> y continuar el proceso de configuración.
 - a. Poner el password del log.
 - b. Instalar los plugins recomendados.
 - c. Crear el usuario admin con su password.
 - d. Poner la url del servidor donde estará alojado jenkins. Si es en local, dejarlo por defecto

Configurar Cmake en jenkins para compilar el proyecto C y pasar las pruebas

1. Una vez hecha la configuración inicial, es necesario instalar el plugin de cmake.
(Manage Jenkins -> Plugins -> Available Plugins -> Cmake)



2. Reiniciar Jenkins.

CMake
Loading plugin extensions
Success
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

3. Configurar Cmake: Manage Jenkins -> Tools

CMake installations

CMake installations ^ Edited

Add CMake 1

CMake

Name

cmake 2

☒ Install automatically ?

Install from cmake.org 4

Version

3.30.0

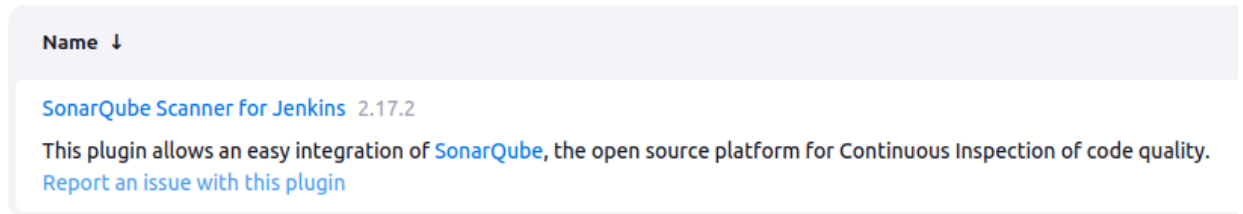
Add Installer

Add CMake

Save 5 Apply

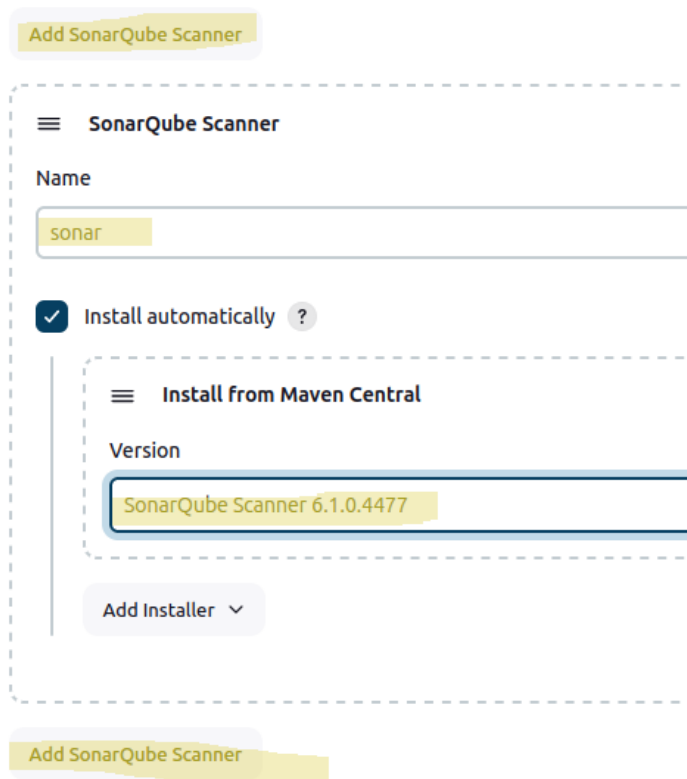
Configurar sonarcloud en jenkins

1. Instalar el plugin de sonarqube scanner (Manage Jenkins -> Plugins -> Available Plugins -> SonarQube Scanner for Jenkins)



2. Configurar la herramienta de sonarqube scanner (Manage Jenkins -> Tools -> SonarQube Scanner installations -> Add SonarQube Scanner).

SonarQube Scanner installations



3. [EN SONARCLOUD] Si no la tienes ya; créate una cuenta en <https://sonarcloud.io/>, puedes usar tu cuenta de github para mayor facilidad.

4. [EN SONARCLOUD] Genera un token nuevo para tu usuario y copiar el valor. (User > My Account > Security)

Profile **Security** Notifications Organizations Appearance

Security

If you want to enforce security by not providing credentials of a real SonarCloud user to run your code scan or to invoke web services, you can provide a User Token as a replacement of the user login. This will increase the security of your installation by not letting your analysis user's password going through your network.

Generate Tokens

Generate Token

Existing Tokens

5. Crear un secreto para el token en Jenkins (Manage Jenkins -> Credentials -> System -> Global Credentials -> Add credentials)

Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >

Global credentials (unrestricted) Add Credentials

Credentials that should be available irrespective of domain specification to requirements matching.

ID	Name	Kind	Description
This credential domain is empty. How about adding some credentials?			

6. Rellenar el formulario. El valor del token que copiamos de sonarcloud debe ir en la casilla de secret.

New credentials

Kind

Secret text

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

Secret

token de sonar

ID ?

SONAR_TOKEN

Description ?

token para logearse en sonar

Create

7. Configurar el servidor de sonar en Jenkins (Manage Jenkins -> System -> SonarQube servers -> add SonarQube)

SonarQube servers

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

☐ Environment variables

SonarQube installations

List of SonarQube installations

Add SonarQube

8. Rellenar el formulario.

SonarQube servers

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

☐ Environment variables

SonarQube installations

List of SonarQube installations

Name

SonarCloud

Server URL

Default is http://localhost:9000

https://sonarcloud.io

Server authentication token

SonarQube authentication token. Mandatory when anonymous access is disabled.

SONAR_TOKEN

+ Add

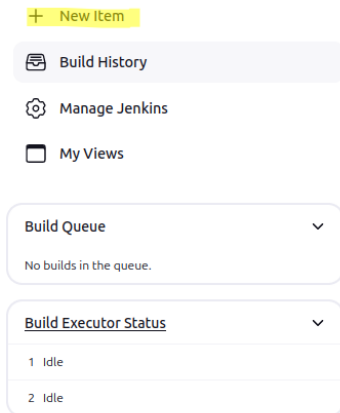
Advanced

Save

Apply

Crear el pipeline para el proyecto

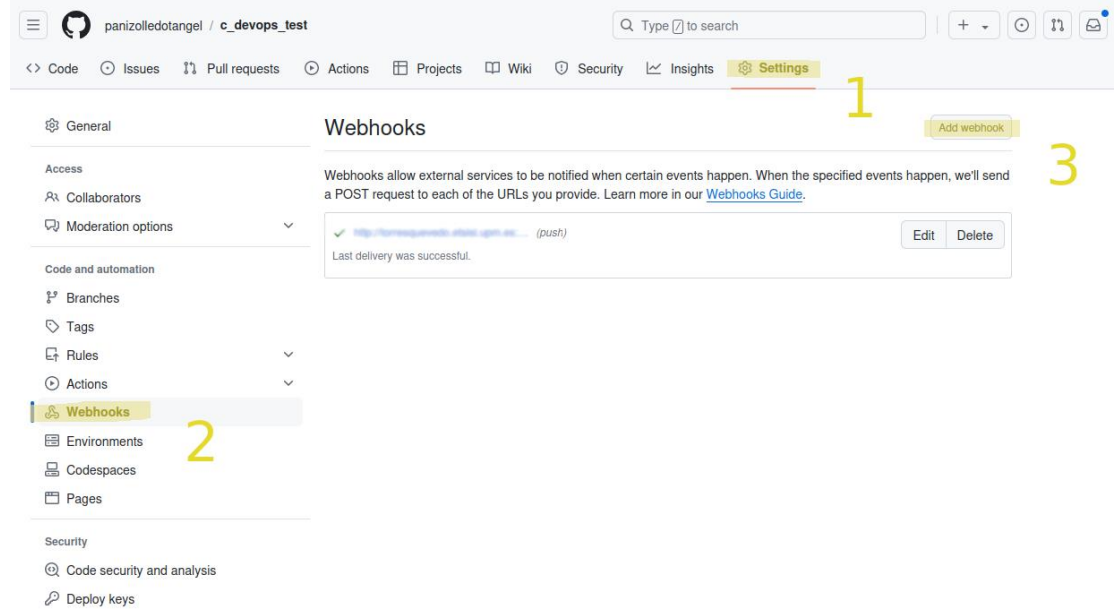
1. Crear un proyecto nuevo de tipo pipeline



7. Configurar el pipeline

- a. Build triggers -> GitHub hook trigger for GITScm polling
 - b. Pipeline -> pipeline script from SCM
 - c. SCM -> Git
 - d. Repositories -> repository URL -> github url
 - e. Branches to build -> */main
 - f. Script Path -> JenkinsFile.
 - g. Save
8. Probar el pipeline con “Build Now”. Deberia ejecutarse sin problema.
9. Configurar un webhook en github para disparar la ejecución de pipeline cada vez que se envíe un push.

a. En github: ir a Settings -> webhooks -> add webhook



b. Poner la “<url del servidor jenkins>/github-webhook/” (OJO! Muy importante terminar la URL con /)

c. Content type = application/json

Payload URL *


`http://localhost:3000/github-webhook/`

Content type *

`application/json`

Secret

SSL verification

 By default, we verify SSL certificates when delivering payloads.

☒ **Enable SSL verification** ☐ **Disable** (not recommended)

Which events would you like to trigger this webhook?

☒ Just the push event.

☐ Send me **everything**.

☐ Let me select individual events.

☒ **Active**

We will deliver event details when this hook is triggered.

Update webhook **Delete webhook**

10. Enviar un cambio al repositorio y comprobar que el pipeline se ejecuta automáticamente.