

docker

INTEGRACIÓN CONTINUA Y JENKINS

PILAR GUZMÁN CABEZAS
2º DESARROLLO DE APLICACIONES WEB
MÓDULO: DESPLIEGUE DE APLICACIONES WEB

INTEGRACIÓN CONTINUA Y JENKINS

¿Qué es la integración continua?

La integración continua (CI) es la práctica de automatizar la integración de los cambios de código de varios contribuidores en un único proyecto de software. Es una de las principales prácticas recomendadas de DevOps, que permite a los desarrolladores fusionar con frecuencia los cambios de código en un repositorio central donde luego se ejecutan las compilaciones y pruebas. Las herramientas automatizadas sirven para verificar que el nuevo código es correcto antes de la integración.

Un sistema de control de versiones del código fuente es el punto clave del proceso de CI. El sistema de control de versiones también se complementa con otras comprobaciones como las pruebas automatizadas de calidad del código, las herramientas de revisión de estilo de sintaxis y mucho más.

Cómo se utiliza la CI

La CI se utiliza generalmente junto con un flujo de trabajo de desarrollo de software de metodología ágil. Una organización compilará una lista de tareas que constituyan una hoja de ruta de productos. A continuación, estas tareas se distribuyen entre los miembros del equipo de ingeniería de software para la entrega. Al utilizar la CI, estas tareas de desarrollo de software pueden desarrollarse independientemente y en paralelo entre los desarrolladores asignados. Una vez que una de estas tareas esté completa, un desarrollador introducirá el nuevo trabajo en el sistema de CI para integrarlo con el resto del proyecto.

Qué es Jenkins?

Jenkins es un servidor automatizado de integración continua de código abierto capaz de organizar una cadena de acciones que ayudan a lograr el proceso de integración continua (y mucho más) de manera automatizada.

Jenkins está completamente escrito en Java y es una aplicación conocida y reconocida por DevOps de todo el mundo, más de 300.000 instalaciones y + 15.500 estrellas en Github lo respaldan.

La razón por la que Jenkins se hizo tan popular es porque se encarga de supervisar las tareas repetitivas que surgen dentro del desarrollo de un proyecto.

Por ejemplo, si tu equipo está desarrollando un proyecto, Jenkins probará continuamente las compilaciones del mismo y será capaz de mostrarte los errores que aparezcan a lo largo de las primeras etapas del desarrollo.

Al usar Jenkins, las compañías de software pueden acelerar su proceso de desarrollo del código; ya que Jenkins puede automatizar, agilizar y aumentar el ritmo de toda la compilación y las pruebas de los proyectos.

Además, Jenkins puede ser implementado a lo largo de todo el ciclo de vida completo del desarrollo.

Desde la fase construcción inicial, la fase de pruebas, en la documentación del software, en su implementación y en todas las demás etapas existentes dentro del ciclo de vida que desees aplicar.

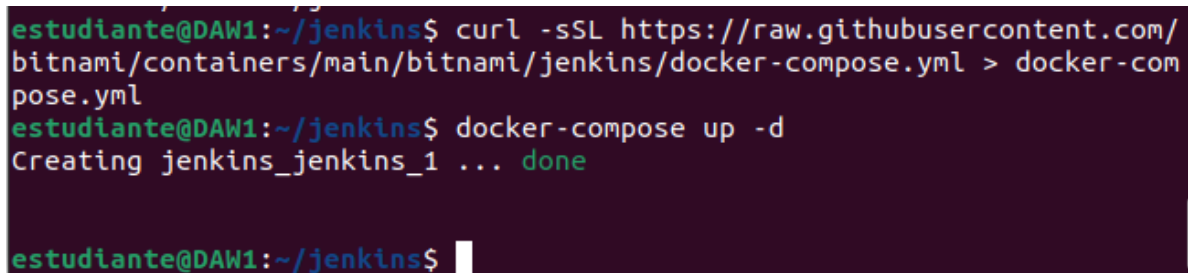
INSTALACIÓN EN UBUNTU

Creamos una carpeta Jenkins para almacenar al archivo YML que se va a crear.

Usamos el siguiente comando para crear el docker-compose:

```
curl -sSL
```

```
https://raw.githubusercontent.com/bitnami/containers/main/bitnami/jenkins/docker-compose.yml > docker-compose.yml
```


A terminal window with a dark background and light-colored text. The prompt is 'estudiante@DAW1:~/jenkins\$'. The first command is 'curl -sSL https://raw.githubusercontent.com/bitnami/containers/main/bitnami/jenkins/docker-compose.yml > docker-compose.yml'. The second command is 'docker-compose up -d', which outputs 'Creating jenkins_jenkins_1 ... done'. The prompt is now 'estudiante@DAW1:~/jenkins\$' with a cursor.

```
estudiante@DAW1:~/jenkins$ curl -sSL https://raw.githubusercontent.com/bitnami/containers/main/bitnami/jenkins/docker-compose.yml > docker-compose.yml
estudiante@DAW1:~/jenkins$ docker-compose up -d
Creating jenkins_jenkins_1 ... done
estudiante@DAW1:~/jenkins$
```

y docker-compose up para levantarlo.

Accedemos a la ruta localhost:8080 en nuestro navegador y nos saldra la pagina de login

← → ↻ ⓘ localhost/login?from=%2F



Welcome to Jenkins!

Username


Password

☐ Keep me signed in

Sign in

Introducimos user como username y bitnami como password, que son las que vienen por defecto para administradores

← → ↻ ⓘ localhost

 **Jenkins** Search (CTRL+K) 🔔 1 🛡️ 1 👤 user ✎ log out Update

Dashboard >

+ New Item

👤 People

📅 Build History

⚙️ Manage Jenkins

📄 My Views

Build Queue ▾

No builds in the queue.

Build Executor Status ▾

1 Idle

2 Idle

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a Job →

Set up a distributed build

Set up an agent →