

GitHub contra GitLab



Escrito por **Juan Carlos Rubio** el 02 de Marzo de 2019

2 min de lectura

Metodologías y Herramientas



Principales herramientas para albergar repositorios de Git en la nube

Las herramientas principales para albergar repositorios de código usando la herramienta Git en la nube son prácticamente dos, GitHub y GitLab, y ambos tienen una funcionalidad similar.

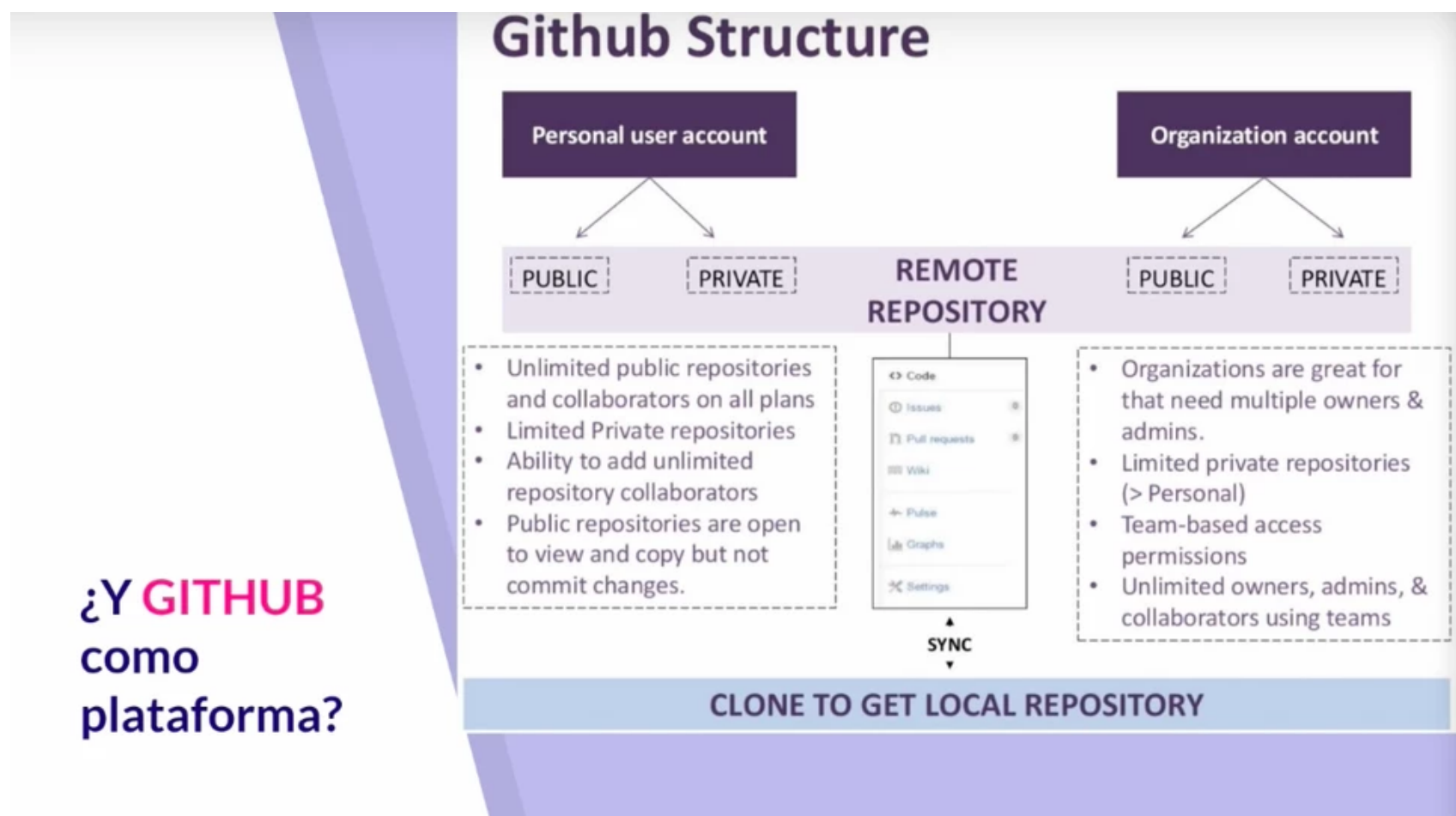
Aparte de ser servicios de hosting de Git en la nube, es decir, prestan el servicio y depositamos nuestra confianza en que nuestro código está en la nube, ambos proporcionan además una interfaz web con la que nos ofrecen algunas facilidades a la hora de trabajar con proyectos que están siendo gestionados a través de la herramienta de Git.

Ambos funcionan como una plataforma en la que se le suman funcionalidades de corte social o de trabajo grupal, que posibilitan la compartición de conocimiento. Para ello ponen a disposición del usuario herramientas como un wiki e incluso herramientas que permiten hacer algún tipo de gestión sobre las tareas o los trabajos.

Cómo funciona GitHub

La plataforma de GitHub es la plataforma que tiene más penetración para proyectos de código y de software libre que están puestos a disposición del público, y con los cuales podemos vincularnos y clonar el repositorio. Incluso podemos llegar a contribuir como un desarrollador que forme parte de la comunidad que colabora o enriquece determinado software.





Tiene dos formas de trabajo con los que nos podemos conectar a la herramienta:

- Usando una cuenta personal, con la que podemos aprender, tener una cuenta pública sin coste.
- También podría ser una organización que quiere utilizar el servicio de hosting, es decir, incluir su código en esa plataforma, y todo lo que la plataforma le ofrece de forma adicional aparte del almacenaje del código fuente.

Como herramienta similar también tenemos GitLab, que es una herramienta que también posibilita el albergar nuestro código, y en la que también podemos tener cuentas de corte personal y confiar el hosting a la misma.

Comparativa GitHub vs GitLab

Si vamos la página web de Github podemos ver en esta vista, en la que se muestran los principales proyectos o de repositorios que son tendencia.

En el caso de la página web de GitLab en la vista de proyectos podemos ver algunos ejemplos de código que está almacenado en GitLab.

La principal ventaja de trabajar con GitLab es que podemos montarnos una réplica de un producto on premise, es decir, en nuestra instalación. Si formamos parte de una estructura empresarial que es celosa de albergar su código fuente en un tercero, por problemas o necesidades varias, podemos montarnos en nuestra instalación este mismo servicio tal cual, y lo podemos hacer de varias formas:

- Podemos hacerlo en una máquina virtual.
- Si estamos usando en nuestra organización Docker Containers, podemos arrancarlo con una línea de Docker.
- Si trabajamos como desarrollador y queremos aprender, podemos levantarlo con un Docker Compose.
- Si tenemos un clúster de Kubernetes, podemos lanzarlo con una única línea y desplegarlo después en dicho clúster.

De todas estas formas tendríamos un servicio on premise, porque no deseamos confiar en un hosting de un tercero el almacenaje de un código fuente, que para nosotros puede tener una especial sensibilidad, o un software privativo que nuestra empresa u organización desarrolla.



Si nuestra meta es que esa plataforma nunca caiga y que nunca tengamos problemas para utilizar ese sistema de control de versiones, podemos tenerlo en la nube.

Consejos

Nuestro consejo es que si vas a dedicarte a un gran proyecto de software libre, con una enorme comunidad detrás, con desarrolladores de todo el mundo y quieres tener un entorno accesible y siempre va a ser un proyecto de software libre, GitHub puede ser una plataforma excelente.

Si quieres empezar a usar GitLab y llega un momento en el que tu código ya no conviene tenerlo en la nube, siempre te lo puedes montar un on premise, con lo cual es un plus de la herramienta GitLab respecto a GitHub en proyectos que en algún momento pueden llegar a convertirse en un proyecto que no convenga tener almacenado en la nube.

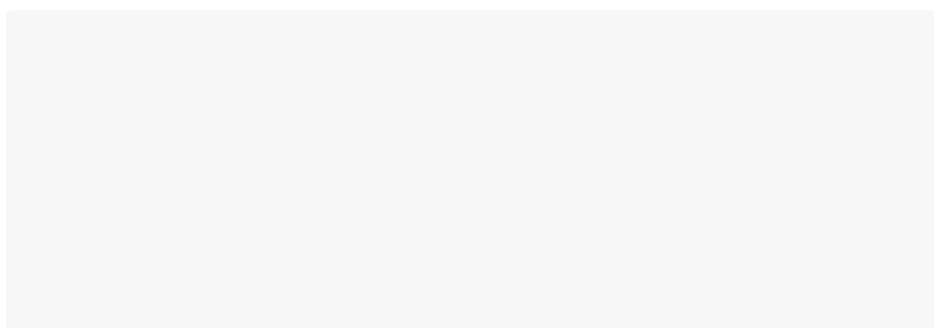
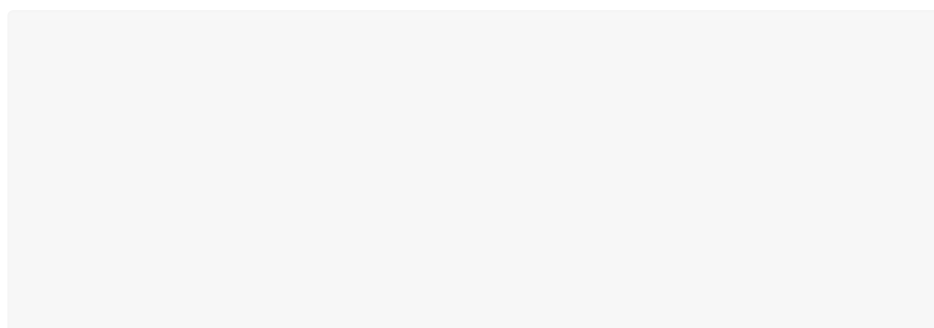
¿Necesitas utilizar un sistema de control de versiones para desarrollar tu software? **La mejor opción es trabajar con Git**, un VCS muy potente y lleno de ventajas, y para ello te proponemos este [curso de Git](#).



Recuerda que puedes comenzar este curso con tu [suscripción de OpenWebinars](#). Si todavía no estás suscrito, aprovecha para hacerlo ahora.

Relacionado

Te dejamos una selección de cursos, carreras y artículos





Git en el flujo DevOps tradicional

Metodologías y Herramientas

01 de Marzo de 2019

Conoce el papel que juega Git tanto en el ciclo de desarrollo de software como en un modelo de DevOps tradicional.



Especialista en Automatización DevOps

carrera

Incluye **6 cursos**:

- Curso de Ansible
- Curso Online de Vagrant
- Curso de Terraform Online

y 3 cursos más!

Duración: **22 horas y 32 minutos**



Curso de Git

curso

Con este curso **aprenderás**:

- Aprenderás qué es un repositorio, una rama o una etiqueta
- Conocerás los principales parámetros de trabajo con GIT
- Aprenderás cómo trabajar con un servidor GIT básico y usaremos Gitlab

Duración: **2 horas y 52 minutos**

Inicia sesión para comentar

Más de 300 empresas confían en nosotros



INFORMACIÓN PARA EMPRESAS

Soluciones

- [Particulares](#)
- [Empresas](#)
- [Centros Educativos](#)

Productos

- [Cursos](#)
- [Carreras](#)
- [Talleres](#)
- [BecasOW](#)

Acerca de

- [Nosotros](#)
- [Contacta](#)
- [Blog](#)
- [Recursos](#)