Informe de Git y GitHub

Cristian andres criollo tovar

Análisis y desarrollo de software ADSO

Servicio nacional de aprendizaje SENA

Bogotá DC

16/08/2025

**¿Qué es Git?**

**Git** es un sistema de control de versiones distribuidas, **gratuito y de código abierto** , diseñado para manejar de todo tipo y tamaño con **velocidad y eficiencia** (Chacon & Straub, 2014). Fue creado por Linus Torvalds en 2005 para gestionar el desarrollo del kernel de Linux. La característica principal de Git es que cada desarrollador tiene una copia completa del repositorio, incluyendo todo el historial de cambios, lo que permite trabajar sin conexión y fusionar los cambios con el repositorio central posteriormente. Esto lo diferencia de los sistemas centralizados, que dependen de un servidor único. Git permite **rastrear y gestionar los cambios en el código** , facilitar la colaboración entre múltiples desarrolladores y restaurar versiones anteriores del proyecto si es necesario (Chacon & Straub, 2014).

**¿Qué es GitHub?**

**GitHub** es una plataforma de **alojamiento de repositorios Git** basada en la web. Es una de las plataformas más populares para la colaboración en el desarrollo de software. Si bien Git es la herramienta de control de versiones, GitHub proporciona la **infraestructura y la interfaz gráfica** para que los equipos colaboren de manera efectiva (O'Reilly, 2022). Actúa como un **servidor remoto** donde los desarrolladores pueden almacenar sus repositorios, gestionar el control de acceso, colaborar en proyectos y **facilitar la comunicación** (O'Reilly, 2022).

**¿Qué es GitLab?**

**GitLab** es una plataforma integral de **DevOps** (Desarrollo y Operaciones) que permite a los equipos gestionar todo el ciclo de vida del desarrollo de software en una sola aplicación (Hostinger, 2025). A diferencia de otras herramientas que solo se centran en el control de versiones, GitLab ofrece una solución completa que abarca desde la planificación del proyecto hasta el monitoreo del despliegue, lo que minimiza la necesidad de utilizar múltiples herramientas de terceros (Kinsta, 2023).

**Git**

* **¿Para qué y cómo funciona?** Git es un sistema de **control de versiones distribuidas** de código abierto. Fue creado por Linus Torvalds en 2005 para gestionar el desarrollo del kernel de Linux (Webempresa, 2020). Su principal es rastrear los cambios en los archivos de un proyecto, permitiendo a los desarrolladores volver a versiones anteriores, comparar cambios y trabajar de manera colaborativa sin sobrescribir el trabajo de los demás.

La clave de su funcionamiento es que cada desarrollador tiene una copia completa del repositorio (el historial del proyecto) en su máquina local. Esto significa que pueden trabajar sin conexión y fusionar sus cambios posteriormente. Git gestiona un flujo de tres estados para los archivos: el directorio de trabajo (archivos modificados), el área de preparación ( *staging* area ) y el repositorio local, que contiene las versiones confirmadas ( *commits* ) (Apptec, sf).

**GitHub**

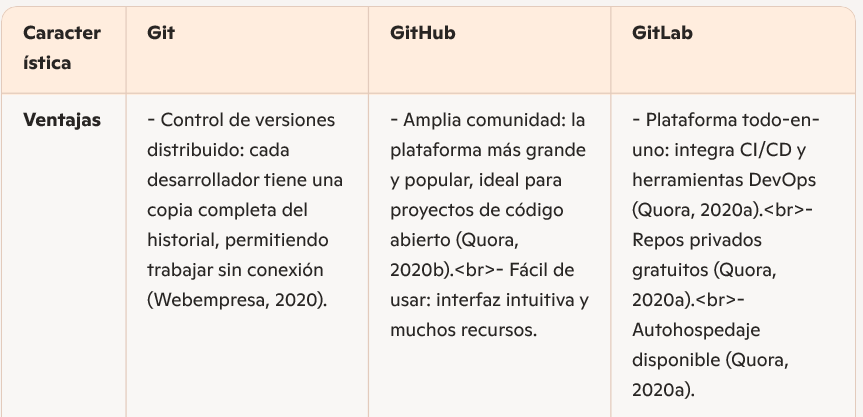
* **¿Para qué y cómo funciona?** GitHub es una plataforma en línea que proporciona servicios de alojamiento de repositorios Git (Webempresa, 2020). Su principal propósito es facilitar la colaboración en proyectos de software. Si bien Git es la herramienta de control de versiones que se usa localmente, GitHub actúa como un **servidor remoto** que permite a los equipos compartir, gestionar y acceder al código desde cualquier lugar.

Funciona como una "red social para programadores" (Quora, 2020a), donde los desarrolladores pueden almacenar sus proyectos, colaborar mediante la creación de ramas ( *branching* ), solicitar la fusión de código ( *pull request* ) y gestionar problemas ( *issues* ).

**GitLab**

* **¿Para qué y cómo funciona?** GitLab es una plataforma web que, al igual que GitHub, ofrece alojamiento para repositorios Git. Sin embargo, su enfoque principal es ser una **solución integral para el ciclo de vida de DevOps** (Quora, 2020b). Su propósito es unificar la planificación, el control de versiones, la integración continua y la entrega continua (CI/CD) en una sola aplicación.

Funciona al integrar estas funcionalidades directamente en la plataforma, lo que elimina la necesidad de usar herramientas de terceros. Por ejemplo, permite a los desarrolladores ejecutar pruebas y desplegar su código de forma automática después de un cambio, todo dentro del mismo entorno de GitLab (Quora, 2020a). A diferencia de GitHub en su momento, GitLab ofrece la posibilidad de crear repositorios privados de forma gratuita y la opción de autohospedar la plataforma en servidores propios (Quora, 2020a).

Tabla

El contenido generado por IA puede ser incorrecto.Tabla

El contenido generado por IA puede ser incorrecto.

**Referencias**

• Apptec. (s.f.). Conoce las diferencias claves entre GitLab, GitHub y Bitbucket. Recuperado de https://api.apptec.cl/blog/conoce-las-diferencias-claves-entre-gitlab-github-y-bitbucket

• Chacon, S., & Straub, B. (2014). Pro Git (2da ed.). Apress.

• Kinsta. (2023). GitLab vs. GitHub: Explora sus principales diferencias y similitudes. Recuperado de https://kinsta.com/blog/gitlab-vs-github/

• O'Reilly, S. (2022). Fundamentos de GitHub: Una guía completa. Wiley.

• Quora. (2020a, 4 de febrero). ¿Cuál es mejor, GitHub o GitLab? Recuperado de https://es.quora.com/Cual-es-mejor-GitHub-o-GitLab

• Quora. (2020b, 19 de enero). ¿Te sirve algo tener GitHub para encontrar trabajo? Recuperado de https://es.quora.com/Sirve-de-algo-tener-GitHub-para-encontrar-trabajo

• Webempresa. (2020, 22 de septiembre). ¿Qué es GitHub y para qué sirve? Recuperado de https://www.webempresa.com/hosting/que-es-github.html