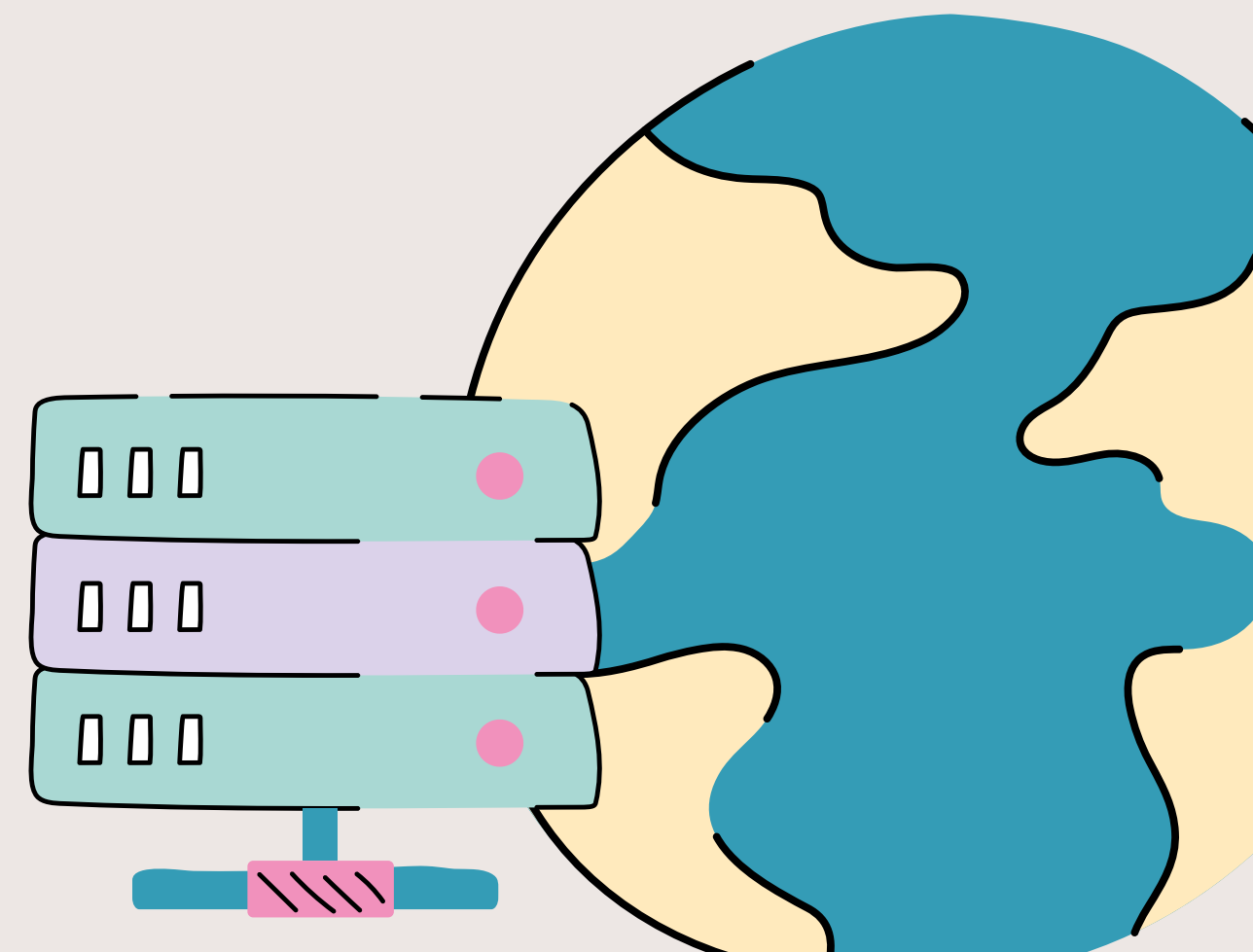
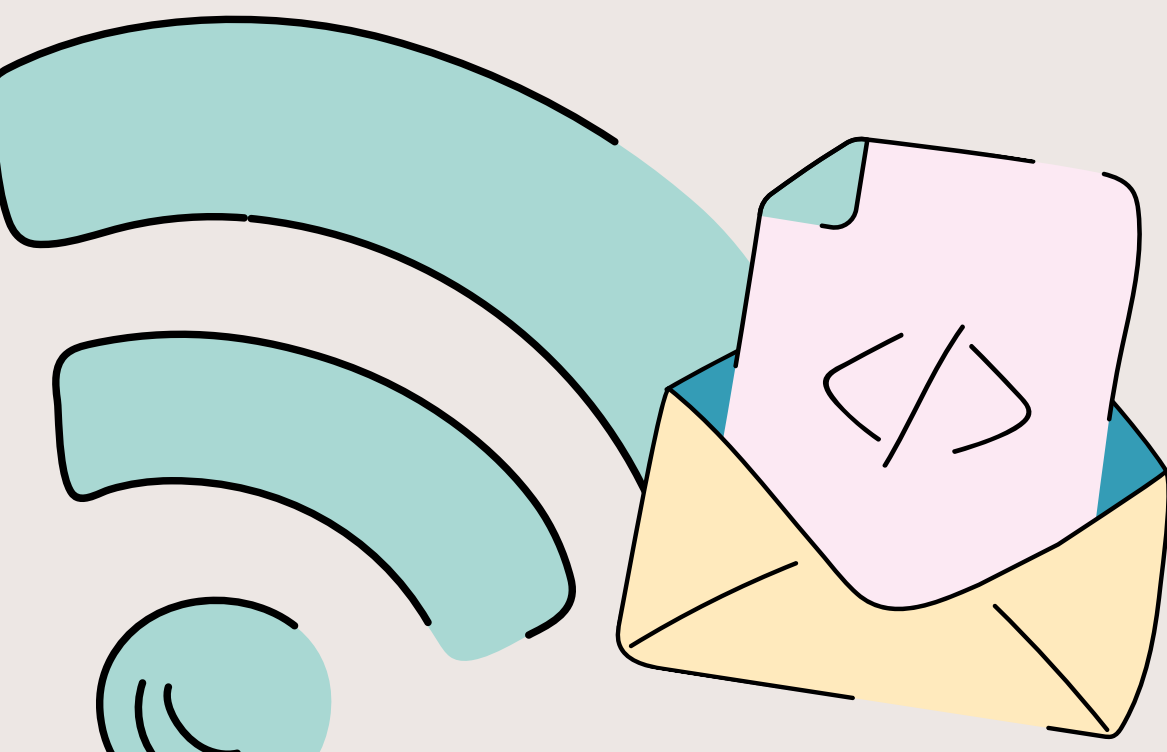


Introducción a Git y GitHub

**Control de versiones y colaboración
eficiente**



Por Juan Duran





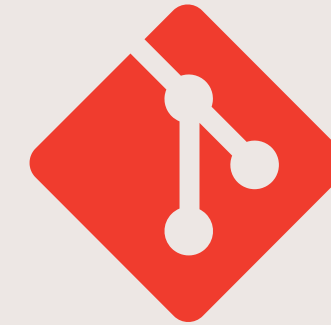
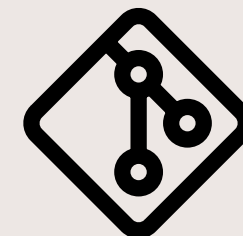
Contenidos

- Introducción a Git y GitHub 📖
- Comparativa ⚖️
- Puntos clave 🔑
- ¿Qué es Git? 🛠️
- ¿Qué es GitHub? 🌐
- Aplicaciones de Git y GitHub 💻
- Recursos para mejorar tu perfil de GitHub 🌟
- Conclusiones 📝

Introducción a Git y GitHub

¿Qué es Git? Git es un sistema de **control de versiones** distribuido que permite a múltiples desarrolladores trabajar en el mismo proyecto sin conflictos. Fue creado por Linus Torvalds en 2005 y se ha convertido en una herramienta esencial para el desarrollo de software. Git permite a los desarrolladores realizar cambios en el código, crear ramas para nuevas características y fusionar esos cambios de manera eficiente y segura.

¿Qué es GitHub? GitHub es una **plataforma de alojamiento de código** basada en Git que facilita la colaboración y el control de versiones. GitHub permite a los desarrolladores almacenar sus proyectos en repositorios, colaborar con otros desarrolladores a través de pull requests y issues, y automatizar flujos de trabajo mediante GitHub Actions. Además, GitHub ofrece características adicionales como wikis y GitHub Pages para la documentación y el hospedaje de sitios web estáticos.



git



Puntos clave de Git

- Sistema de **control de versiones** distribuido.
- Permite **trabajar sin conexión**.
- Soporte robusto para **ramas y fusión de código**.
- **Seguridad** mediante SHA-1.
- **Historial** completo y desempeño rápido.
- **Integridad** de datos y desarrollo paralelo.
- Posibilidad de **deshacer** cambios.



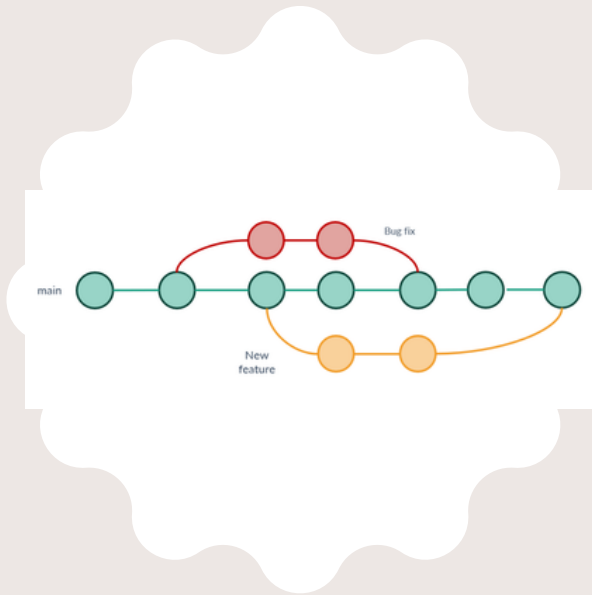
versus



Puntos clave de GitHub

- **Plataforma** de colaboración basada en Git.
- **Repositorios** públicos y privados.
- **Issues** para seguimiento de errores y tareas.
- **Pull requests** para revisión de código.
- **GitHub Actions** para automatización.
- **GitHub Pages** para sitios web estáticos.
- **Wikis** y herramientas de gestión de proyectos.
- **Integraciones** con otras herramientas y servicios.
- Facilita la **colaboración** en proyectos de código abierto.

Puntos clave



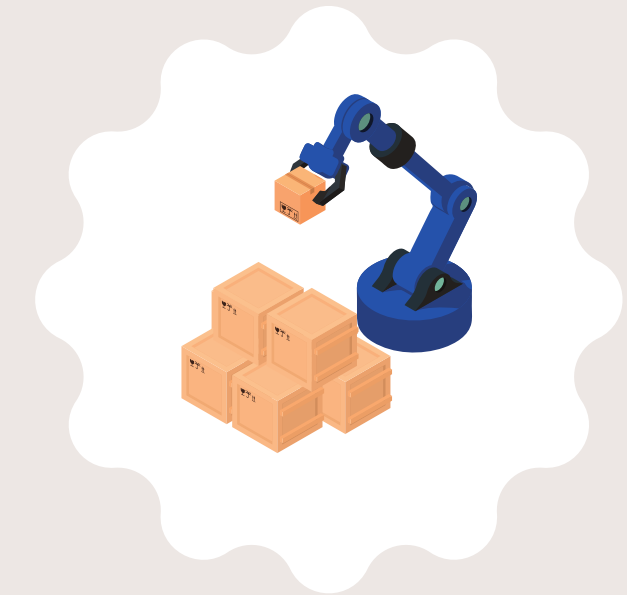
Control de versiones

Git permite **gestionar** y rastrear cambios en el **código** de manera eficiente. Además, facilita la recuperación de **versiones** anteriores del código en caso de errores.



Colaboración

GitHub facilita la **colaboración** entre desarrolladores mediante pull requests e issues. También permite la **revisión** de código y la discusión de cambios antes de fusionarlos.



Automatización

GitHub Actions permite **automatizar** flujos de trabajo como pruebas y despliegues. Esto mejora la **eficiencia** y reduce el riesgo de errores humanos en el proceso de desarrollo.

¿Qué es Git? 🛠️

Git es un sistema de **control de versiones** distribuido que permite a los desarrolladores trabajar en proyectos de manera **colaborativa** y eficiente. Cada desarrollador tiene una copia completa del **repositorio**, lo que permite trabajar sin conexión y realizar **operaciones** locales rápidas. Git ofrece un soporte robusto para la creación de **ramas**, lo que facilita el desarrollo de nuevas características de manera aislada. Además, Git utiliza SHA-1 para identificar versiones y cambios, asegurando la **integridad** del historial de versiones.



Control de Versiones

Historial

Repositorio

Clonación

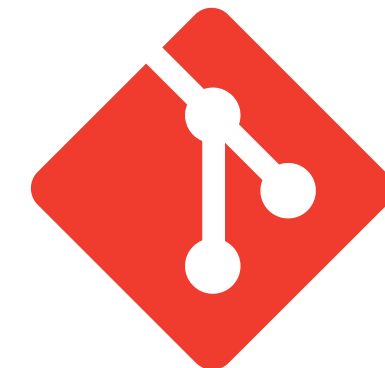
Branching

Ramas



Fusión

Merge Conflicts



¿Qué es GitHub?

GitHub es una **plataforma** de colaboración basada en Git que permite a los desarrolladores **almacenar** y **gestionar** proyectos de código. Los **repositorios** en GitHub pueden ser públicos o privados, lo que facilita la **colaboración** en proyectos de código abierto y privados. GitHub ofrece **herramientas** como issues para el seguimiento de errores y tareas, y pull requests para la revisión y discusión de cambios antes de fusionarlos en la rama principal. Además, GitHub Actions permite **automatizar** flujos de trabajo como CI/CD mediante la definición de tareas en archivos YAML.



Colaboración

Revisión de Código

Pull Requests

Comentarios

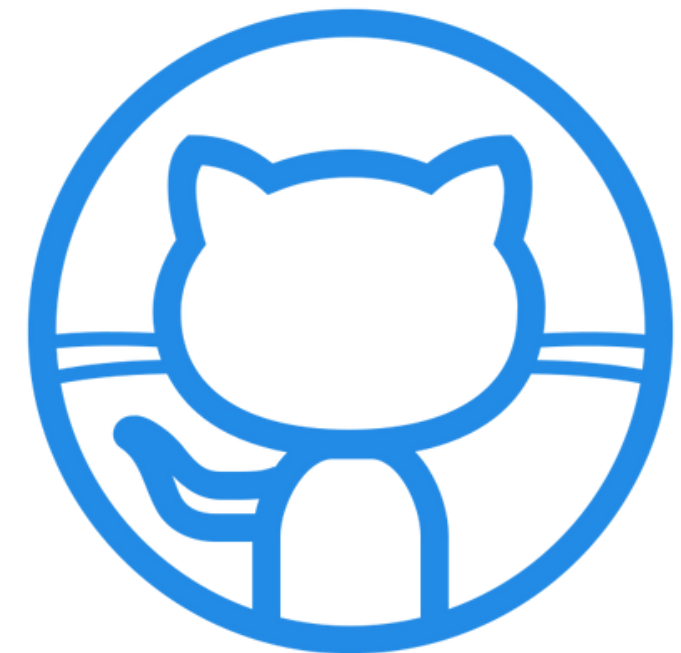


Issues

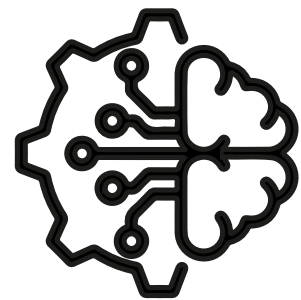
Seguimiento

Automatización

CI/CD

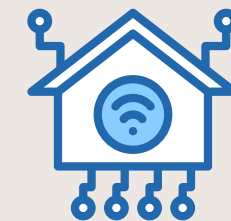


Aplicaciones de Git y GitHub



Aplicaciones de Git

- **Desarrollo de Software:** Git es esencial para el control de versiones y la colaboración en proyectos de software, permitiendo a los desarrolladores trabajar en paralelo y fusionar cambios de manera eficiente.
- **Documentación:** Git se utiliza para gestionar la documentación técnica y los wikis, asegurando que los cambios en la documentación se registren y puedan ser revertidos si es necesario.
- **Automatización:** Git se integra con herramientas de automatización como GitHub Actions, permitiendo la automatización de pruebas, despliegues y otras tareas repetitivas.

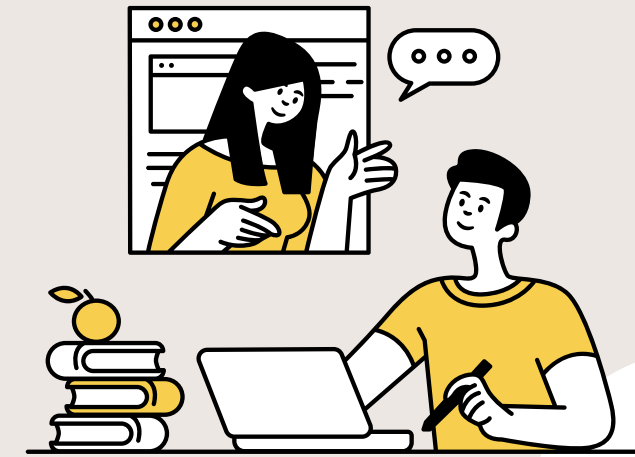


Aplicaciones de GitHub

GitHub facilita la **colaboración** en proyectos de código abierto y privados, proporcionando **herramientas** para el seguimiento de errores, la revisión de código y la gestión de proyectos. Además, GitHub permite a los desarrolladores **crear** y **hospedar** sitios **web** estáticos mediante GitHub Pages, y **automatizar** flujos de trabajo con GitHub Actions.

Recursos para mejorar tu perfil de GitHub

- GitHub Profile README Generator: Generador impulsado por IA que te ayuda a crear un perfil impresionante en minutos. Ofrece plantillas personalizables y vista previa en tiempo real.
- ReadMeMaker: Generador de README profesional con plantillas y vista previa en vivo. Perfecto para documentar proyectos de manera clara y efectiva.
- readme.so: Editor sencillo para personalizar y agregar rápidamente todas las secciones necesarias para tu README. Ideal para crear documentación clara y bien estructurada



Conclusiones

Esencial

Git y GitHub son herramientas fundamentales para el desarrollo de software colaborativo.

Control Eficiente

Git permite gestionar y rastrear cambios en el código de manera segura y eficiente.

Colaboración

GitHub proporciona herramientas para la colaboración y gestión de proyectos, como pull requests e issues.

Automatización

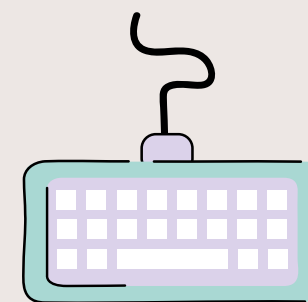
GitHub Actions permite automatizar tareas repetitivas, mejorando la eficiencia del desarrollo.

Adopción

Git y GitHub son ampliamente utilizados en la industria del software, lo que los convierte en habilidades esenciales para los desarrolladores.

Aprendizaje

Dominar Git y GitHub es crucial para cualquier desarrollador que desee trabajar en proyectos de software modernos y colaborativos.



Gracias



Por Juan Duran

“Coding, Gaming and Leveling Up”