Comprensión Git/Github

Amaro Morales Cedrés Matias Ariel Zayas Vera

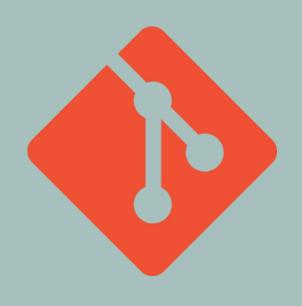
Índice

- Explicación sistema de control de versiones
- ♦ Diferencias entre GIT y GITHUB
- Beneficios del control de versiones
- Flujo básico de trabajo con GIT y GITHUB
- Conclusión final

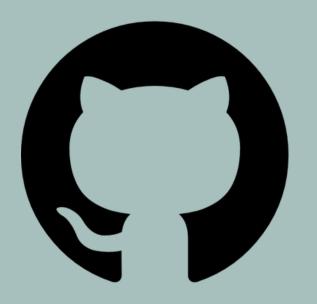


Explicación sistema de control de versiones

Es una herramienta que permite gestionar los cambios que se van realizando en archivos o proyectos. Facilita seguirlo con claridad o restaurar algunas versiones anteriores



En Git funciona para guardar y controlar cambios en el proyecto a lo largo del tiempo, permitiendo volver a lo anterior sin problema



Github funciona de manera muy similar a Git, esta esta caracterizada por ser una plataforma en línea.

Diferencias entre GIT y GITHUB

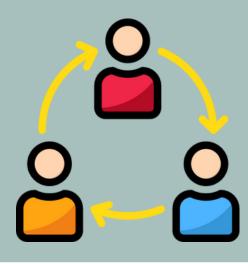
ASPECTOS	GIT	GITHUB
UBICACIÓN	Ordenador local	Plataforma en línea
USO	Manejar versiones	Colaborar, compartir código
DEPENDENCIA	No necesita conexión	Necesita internet
HERRAMIENTAS	Solo maneja versiones locales	Ofrece distintas opciones para su uso
¿QUÉ ES?	Sistema de control de versiones	Plataforma en linea para proyectos que usan Git

Beneficios del control de versiones

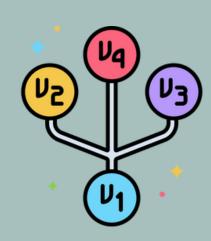
Registro histórico



Colaboración



Ramas



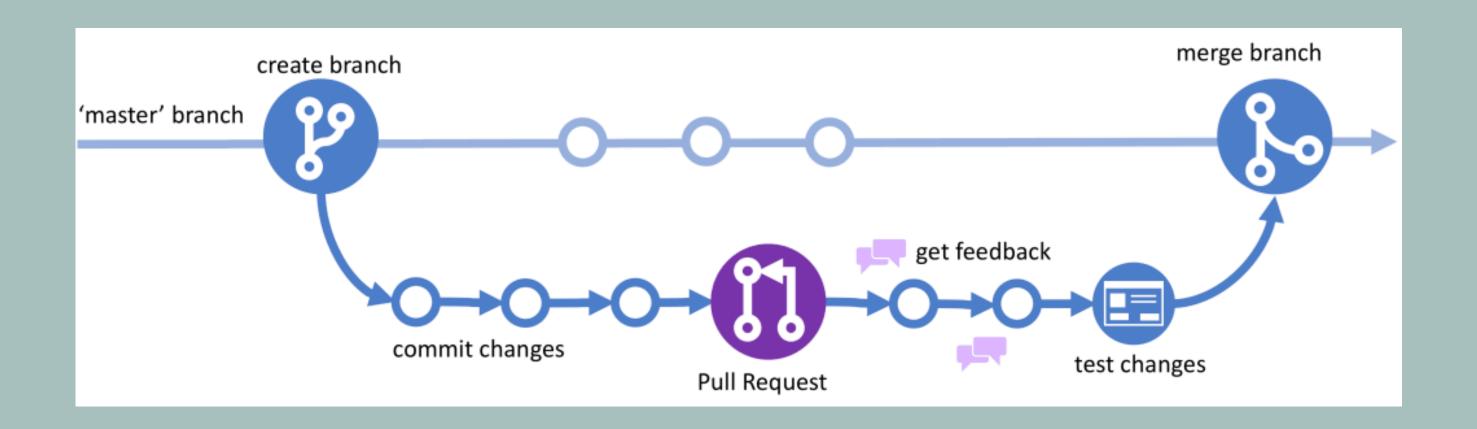
Registra los cambios realizados en el código lo que facilita la identificación y resolución de errores, en caso de problemas podemos volver a versiones anteriores y restaurar el proyecto, también nos facilita en la documentación.

Esto permite a varios desarrolladores trabajar simultáneamente en el mismo proyecto sin conflictos, ya que cada uno tiene una copia local del código y puede actualizarla sin afectar a los demás.

Los desarrolladores pueden trabajar en diferentes ramas del código sin afectar al resto del equipo.

Flujo básico de trabajo con GIT y GITHUB

- 1. El repositorio remoto inicial es la rama (Branch) main.
- 2. Creas una rama a partir de main.
- 3. Trabajas localmente en tu rama, guardando tus cambios (commit).
- 4. Publicas tu rama (Push).
- 5. Solicitas un "pull request" para que tu rama sea parte de la rama main (merge).
- 6. Eliminas tu rama.



Conclusión Final

Git es una herramienta de control de versiones que permite gestionar proyectos y rastrear cambios en el código de forma eficiente. GitHub, por su parte, es una plataforma que aprovecha Git para ofrecer funcionalidades colaborativas como repositorios remotos, control de acceso y herramientas de integración.

Git es esencial para el manejo local de versiones, mientras que GitHub es más útil para la colaboración en equipo y el acceso remoto. Uno no es "mejor" que el otro, ya que se complementan: Git para trabajar en local y GitHub para colaborar y compartir.

Muchas 91acias!

AMARO Y MATÍAS