

Sistema de control de versiones

¿Qué es un sistema de control de versiones?

Un sistema de control de versiones (VCS) es una herramienta que permite gestionar y realizar un seguimiento de los cambios en archivos y proyectos a lo largo del tiempo. Facilita mucho colaboración entre múltiples desarrolladores, permitiendo que trabajen en el mismo proyecto sin conflictos. Sirven para el desarrollo de software, ya que permiten revertir cambios, comparar versiones y mantener un historial completo de modificaciones.

Justo en este curso lo que buscamos es entender como funcionan estos sistemas, pero, con git.

¿Qué es Git?

Git es un sistema de control de versiones distribuido, lo que significa que cada desarrollador tiene una copia completa del repositorio en su máquina local.

Esto permite trabajar sin conexión y realizar cambios de manera independiente antes de sincronizar con el repositorio central.

Git es ampliamente utilizado en el desarrollo de software debido a su eficiencia, flexibilidad y capacidad para manejar proyectos grandes y complejos.

Y lo más importante, es software libre y de código abierto.

Conceptos básicos de Git

Snapshot

Un snapshot en Git es una captura del estado actual de los archivos en un repositorio en un momento específico. Cada vez que realizamos un commit, Git guarda un snapshot de cómo lucen todos los archivos en ese momento.

Commit

Un commit es un registro de los cambios realizados en el repositorio. Cada commit tiene un identificador único (hash) y contiene información sobre los cambios realizados, el autor y la fecha. Los commits permiten rastrear el historial del proyecto y revertir a versiones anteriores si es necesario.

Branch

Una branch o rama es una línea de desarrollo independiente dentro de un repositorio. Permite trabajar en diferentes características o correcciones de errores sin afectar la rama principal. Este concepto es principalmente usado para desarrollo colaborativo.

Repositorio

Un repositorio es un espacio de almacenamiento donde se guardan todos los archivos y el historial de cambios de un proyecto. Puede ser local (en la máquina del desarrollador) o remoto (en un servidor, como gitlab o github). Git permite clonar, crear y gestionar repositorios de manera eficiente.

Cherry-pick

El cherry-pick es una operación en Git que permite aplicar un commit específico de una rama a otra. Esto es útil cuando se desea incorporar un cambio particular sin fusionar toda la rama. Al hacer cherry-pick, Git crea un nuevo commit en la rama de destino con los cambios del commit seleccionado.

Rebase

El rebase es una operación en Git que permite mover o combinar commits de una rama a otra. A diferencia del merge, que crea un nuevo commit de fusión, el rebase reescribe el historial de commits, aplicando los cambios de una rama sobre otra.

Rebase onto

`git rebase -onto` es una variante del rebase que permite especificar una nueva base para una serie de commits. Es similar a un cherry pick, pero en lugar de aplicar un solo commit, se pueden mover varios commits a una nueva.