Identifica herramientas de Versionamiento

Heidy Dayana Otagrí Rodríguez

Tecnología en Análisis y Desarrollo de Software

Servicio Nacional de Aprendizaje (SENA)

Ficha 2627060

Ing. Yerman Augusto Hernández Sáenz



Bogotá D.C., marzo del 2023

En este trabajo, exploraremos el uso de Git, un sistema de control de versiones ampliamente utilizado en programación. Nos centraremos en las diferencias entre el uso local y remoto de Git, destacando cómo estas dos dimensiones son fundamentales en la gestión de proyectos y la colaboración entre desarrolladores. Este análisis nos permitirá comprender mejor la importancia de Git en el desarrollo de software moderno.

|  |  |  |
| --- | --- | --- |
| **Característica** | **Git Local** | **Git Remoto** |
| Acceso | Acceso sin necesidad de conexión a Internet. | Acceso a través de Internet a través de servidores remotos o plataformas de alojamiento (GitHub, GitLab, etc.). |
| Actualizaciones | Actualiza el repositorio local mediante comandos como `git pull` y `git fetch`. | Los cambios se actualizan automáticamente en el repositorio remoto con `git push` y `git pull`. |
| Almacenamiento | Los archivos y la historia de cambios se almacenan en tu máquina local. | Los archivos se almacenan en el servidor remoto, actuando como una copia de seguridad principal. |
| Comandos Clave | Utiliza comandos como `git commit`, `git branch`, `git merge`, `git checkout`, `git log`, etc. | Emplea comandos como `git push`, `git pull`, `git clone`, `git fork`, `git remote`, etc. |
| Copias del Repositorio | Puedes tener múltiples copias de tu repositorio local en diferentes carpetas de tu máquina. | Un único repositorio remoto accesible por múltiples colaboradores. |
| Copias de Seguridad | Responsabilidad tuya crear copias de seguridad si pierdes tu máquina local | El repositorio remoto actúa como copia de seguridad principal. |
| Propósito | Ideal para trabajo individual, experimentación y registro personal de cambios. | Diseñado para colaboración, compartición de proyectos con otros desarrolladores y trabajo en equipo. |
| Ramas | Puedes crear y gestionar múltiples ramas localmente para separar y organizar el trabajo. | Las ramas son compartidas y colaborativas, facilitando el trabajo en equipo. |
| Sincronización | Necesitas sincronizar manualmente tu repositorio con el remoto usando `git push` y `git pull`. | Los cambios se sincronizan automáticamente con el repositorio remoto. |
| Ubicación | El repositorio local reside en tu máquina, lo que permite trabajar sin conexión a Internet | El repositorio remoto se encuentra en servidores remotos o plataformas de alojamiento accesibles en línea. |