

**Servicio Nacional de Aprendizaje**

**(SENA)**

**Técnica**

**GA7-220501096-AA1-EV03** identifica herramientas de versionamiento

**Santiago Rodríguez Vallejo**

Ficha N° 2721431

**Medellín, Antioquia**

**Introducción**

En el desarrollo de software moderno, el control de versiones es una práctica fundamental que permite a los equipos gestionar y encontrar los cambios en el código fuente de una aplicación a lo largo del proyecto. Las herramientas de versionamiento no solo facilitan la colaboración para el desarrollo, sino también aseguran la integridad y recuperación del código en la línea del tiempo del proyecto. En este trabajo se expondrá la herramienta de versionamiento Git y se expondrán las características del versionamiento local y el versionamiento remoto.

**Herramienta de versionamiento**

Una herramienta de versionamiento es un sistema que registra los cambios realizados en un conjunto de archivos en la línea de tiempo de un proyecto. Esto permite a los involucrados revertir a versiones anteriores, comparar cambios y colaborar de manera más efectiva en proyectos de software. Ejemplos populares de herramientas de versionamiento incluyen Git, Subversion, Mercurial y CVS. Aquí hablaremos de las características de Git, la herramienta de versionamiento más popular y usada en el desarrollo de software.

**Diferencias entre Git local (versionamiento local) y Git remoto (versionamiento remoto)**

|  |  |  |
| --- | --- | --- |
| **Característica** | **Git local** | **Git remoto** |
| **Almacenamiento** | Los datos se almacenan en el repositorio local del usuario, es decir, se usa el almacenamiento local del sistema. | Los datos se almacenan en un servidos remoto, es decir, en un servidor de internet mediante el uso de una plataforma. (Ej. GitHub o GitLab). |
| **Acceso** | Accesible solo desde el computador del usuario. | Accesible desde cualquier computador o dispositivo con acceso a internet. |
| **Colaboración** | Limitada, solo se trabaja individual. | Facilita la colaboración entre varios desarrolladores. |
| **Seguridad** | Mas segura física y menos vulnerabilidades externas al no estar expuesto. | Depende de la seguridad del servidor remoto: la plataforma que usa. |
| **Copia de seguridad** | Responsabilidad del usuario. | Servicio ofrecido por la plataforma. |
| **Sincronización** | No requiere porque todos los cambios son localmente | Requiere sincronización mediante comandos para la actualización del proyecto. |
| **Visibilidad** | Solo el usuario puede ver los cambios. | Cambios visibles para todos los colaboradores con acceso al respositorio. |