



**EVIDENCIA DE CONOCIMIENTO: GA7-220501096-AA1-EV03 IDENTIFICA HERRAMIENTAS DE VERSIONAMIENTO**

**ELABORADO POR LOS APRENDICES:**

**JHON FREDY ARBOLEDA MUÑOZ**

**DAVID GIRALDO PINEDA**

**JOHAN SEBASTIÁN SANDOVAL GARZÓN**

**FECHA:**

**21/04/2025**

**FICHA: 2834907**

**INSTRUCTOR:**

**JHON HADER RODRIGUEZ PERDOMO**

**CENTRO DE DESARROLLO AGROINDUSTRIAL, TURÍSTICO Y TECNOLÓGICO DEL GUAVIARE**



## Contenido

INTRODUCCION .....	3
OBJETIVO .....	4
TABLA CON LAS DIFERENCIAS ENTRE EL SISTEMA DE CONTROL DE VERSIONAMIENTO GIT LOCAL Y GIT REMOTO.....	5
CONCLUSIONES .....	7



## INTRODUCCION

Como aprendiz del Sena del programa adso 2834907 con la elaboración de las siguientes tablas con las diferencias entre git local y remoto podemos profundizar y así saber cómo es su funcionamiento, de igual forma al utilizar estas herramientas tenemos que tener muy claro cuáles son su beneficio y sus contras y cuál es la mejor herramienta que nos favorece.



## OBJETIVO

El principal objetivo es saber diferenciar entre git remoto y git local para así saberlos utilizar y a la hora de alojar nuestros proyectos tengamos la plena seguridad de que no se nos va a perder algún cambio que realicemos.



## TABLA CON LAS DIFERENCIAS ENTRE EL SISTEMA DE CONTROL DE VERSIONAMIENTO GIT LOCAL Y GIT REMOTO

<b>GIT LOCAL</b>	<b>GIT REMOTO</b>
Los repositorios locales residen en nuestras máquinas	Está alojado en la web y se puede acceder a él desde cualquier lugar
permiten aprovechar al máximo el control de versiones de Git	es difícil profundizar en sus confirmaciones pasadas y ver realmente el historial de su aplicación para poder usar el control de versiones
Las operaciones son muy rápidas porque todo está en tu máquina.	tiene una interfaz más agradable para que podamos navegar por la aplicación en diferentes puntos del historial
Puedes trabajar sin conexión a internet.	nos permite publicar nuestro código al mundo para que puedan leerlo o escribirlo
Tú decides cuándo y qué compartir.	Sirve para guardar y sincronizar cambios entre varios desarrolladores
Sirve como copia de seguridad externa.	Está alojado en nuestra computadora y si esta falla se perderán los datos
Permite trabajar de forma personal	Es más un trabajo colaborativo entre los integrantes de un grupo de trabajo
Usa commando como: git add, git commit	Usa commando como: git push, git pull, git fetch
Puedes hacer commits y ver el historial completo (git log) sin depender de nadie.	Necesitas sincronizar con el repositorio remoto para ver los últimos cambios de otros.



Tus cambios solo son visibles en tu máquina, hasta que los envíes.	Una vez haces git push, los demás pueden ver y usar tus cambios.
Puedes crear, modificar y eliminar ramas localmente (git branch, git checkout, etc.).	Puedes subir (push) una rama nueva, ver ramas remotas (git branch -r) y colaborar en ellas.
Solo resuelves conflictos con tus propios cambios.	Puedes enfrentar conflictos al hacer pull si tu código choca con el de otros.
Solo tú puedes acceder (a menos que compartas tu máquina).	Cualquiera con permisos puede acceder desde cualquier parte del mundo.
Puedes iniciar un proyecto desde cero con git init.	Puedes clonar (git clone) un proyecto existente y contribuir.



## CONCLUSIONES

- 1- Aprender las diferencias entre git local y git remoto
- 2- Podemos conocer cuál es la mejor herramienta que nos sirve para alojar nuestros proyectos sin tener el riesgo de perderlos.
- 3- Con la elaboración de este trabajo podemos entender que estas herramientas son muy importantes para nuestros proyectos y tanto el git local sirve para almacenar nuestras copias de seguridad, pero el git remoto nos da más seguridad y accesibilidad al igual que podemos trabajar de forma grupal.



## BIBLIOGRAFIA

- 1- <https://frontend.turing.edu/lessons/module-2/git-forking-vs-cloning.html#:~:text=our%20local%20repo%20%2D%20which%20lives,fork%20their%20own%20local%20copy>.