

“IDENTIFICA HERRAMIENTAS DE VERSIONAMIENTO ”

PRESENTADO A:
CRISTIAN CAMILO ARIAS

PRESENTADO POR:
ANGIE VANESSA PALACIO VILLEGAS

SERVICIO NACIONAL DE APRENDIZAJE - SENA
TECNOLOGÍA EN ANÁLISIS Y DESARROLLO SOFTWARE
2024



INTRODUCCIÓN

A continuación veremos las diferencias de Git remoto y Git local, Git es un sistema de control de versiones que nació en el año 2005 producto de las necesidades que afrontaba la comunidad del Kernel de Linux encontrar un mecanismo que les permitiera soportar las actualizaciones y mantenimientos que requiere este gran proyecto.

Diferencias	
Git	GitHub
Git es un sistema de control de versiones distribuido que registra las distintas versiones de un archivo (o conjunto de archivos). Le permite a los usuarios acceder, comparar, actualizar, y distribuir cualquiera de las versiones registradas en cualquier momento.	GitHub principalmente es una plataforma de alojamiento para albergar tus repositorios Git en la web. Esto permite a los usuarios mantener sus repositorios remotos privados o abiertos para esfuerzos colaborativos.
Los usuarios instalan y ejecutan Git en sus equipos locales. Esto significa que la mayoría de las operaciones de Git se pueden lograr sin una conexión a internet.	Sin embargo GitHub es un servicio basado en la web que opera solamente en línea. Esto significa que necesitas estar conectado para hacer cualquier cosa en GitHub.
git es un software de VCS local que permite a los desarrolladores guardar instantáneas de sus proyectos a lo largo del tiempo. Generalmente es mejor para uso individual.	Bifurcación: El proceso de copiar el código de otra persona del repositorio para modificarlo.
	Pull: Cuando haya terminado de hacer cambios en el código de otra persona, puede compartirlos con el propietario original a través de una «solicitud pull».

	Fusión: Los propietarios pueden añadir nuevos cambios a sus proyectos a través de una fusión, y dar crédito a los contribuyentes que los han sugerido

Comando básicos	
GIT	Git Remoto
Para inicializar un repositorio desde un directorio existente se debe usar el comando:	Para ver los repositorios remotos configurados se puede ejecutar el comando:
\$ git init	\$ git remote
Este comando creará un nuevo subdirectorio llamado .git, el cual contendrá todos los archivos necesarios para el repositorio. Si se desea obtener una copia desde un repositorio existente se debe utilizar el comando de clonación.	Para definir un repositorio remoto y asociarlo a un nombre para su referenciación se utiliza el siguiente comando:
\$ git clone https://url_del_repositorio	git remote add [nombre-remoto] [url]
Para visualizar el estado actual de tus archivos indicando si están o no	Donde nombre-remoto corresponde al nombre con que se referencia el repositorio y URL es la ubicación lógica del mismo en un entorno de red o en una dirección de internet. En el

rastreados por Git se puede ejecutar el siguiente comando:

```
$ git status
```

Para todos los archivos nuevos que se desean ser rastreados por Git, se debe indicar con el siguiente comando:

```
$ git add Nombre_archivo
```

También es posible indicar a Git que haga rastreo de un directorio, lo cual implica que recursivamente se hace rastreo de todos los archivos en el interior del directorio.

```
$ git add Directorio
```

el comando add indica que un archivo o directorio debe ser añadido en la próxima confirmación. Para confirmar cambios se ejecuta el comando:

```
$ git commit
```

Al ejecutar la confirmación el sistema desplegará un mecanismo para recibir un mensaje de confirmación que será asociado a esta operación de commit.

siguiente ejemplo se define un repositorio ubicado en un servidor de Github y cuyo nombre de referencia es ref.

```
$ git remote add ref  
https://github.com/paulboone/ticgit
```

Una vez definido el remote se pueden extraer los datos utilizando el siguiente comando:

```
$ git fetch [nombre-remoto]
```

El anterior comando se conecta al repositorio remoto y trae todos los datos de los cuales aún no se tiene copia en tu repositorio local. Para enviar información desde el repositorio local hacia el servidor remoto, se utiliza el siguiente comando:

```
git push [nombre-remoto]  
[nombre-rama]
```

Recordemos que, si se ha clonado un repositorio desde alguna ubicación, Git asigna el nombre origin al servidor del que se ha realizado la clonación. Así, si por ejemplo queremos enviar nuestra

<p>También es posible agregar el comentario explícitamente en la ejecución de la confirmación usando la opción -m.</p>	<p>rama master al servidor origin se debe ejecutar el comando de la siguiente forma:</p>
<pre>\$ git commit -m "En esta version se arreglo el archive W"</pre>	<pre>\$ git push origin master</pre>
<p>También es posible ejecutar una operación de confirmación que salte el paso de preparación que se logra con la ejecución del comando add. Es decir, la operación de confirmación se encarga de preparar todos los archivos rastreados y luego confirmar. Esto es posible agregando la opción -a.</p>	
<pre>\$ git commit -a -m 'comentario de esta confirmación'</pre>	
<p>Para visualizar el histórico de las confirmaciones desde la más reciente hasta la más antigua realizadas sobre un repositorio se ejecuta el comando:</p>	
<pre>\$ git log</pre>	

