



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Post Grados

Alex Ronaldo Mendoza Chonay 3009941330101

Git

Git es un sistema de control de versiones utilizado para guardar diferentes versiones de un archivo o conjunto de archivos para que cualquier versión sea recuperable cuando se desee.

Git también facilita el registro y comparación de diferentes versiones de un archivo. Esto significa que los detalles sobre qué cambio, quien cambio que, o quien ha iniciado una propuesta, se pueden revisar en cualquier momento.

Sistema de control de versiones

Un control de versiones se refiere al método utilizado para guardar las versiones de un archivo para referencia futura.

De manera intuitiva muchas personas ya utilizan control de versiones en sus proyectos al renombrar las distintas versiones de un mismo archivo de varias formas. Pero esta forma de abordarlo es propensa a errores e inefectivo para proyectos grupales.

Además, con esta forma de abordarlo, rastrear qué cambió, quién lo cambió y porqué se cambió, es un esfuerzo tedioso. Esto resalta la importancia de un sistema de control de versiones confiable y colaborativo como Git.

Estado de los archivos en Git

En Git hay tres etapas primarias en las cuales un archivo puede estar: estado modificado, estado preparado y estado confirmado.

Estado modificado

Un archivo en el estado modificado es un archivo revisado, pero sin registrar.

En otras palabras, archivos en el estado modificado son archivos que has modificado, pero no le has instruido explícitamente a Git que controle.

Estado preparado

Archivos en la etapa preparado son archivos modificados que han sido seleccionados en su versión actual y están siendo preparados para ser guardados al repositorio. git durante la próxima instantánea de confirmación.

Una vez que el archivo está preparado implica que has explícitamente autorizado a Git que controle la versión de ese archivo.

Estado confirmado

Archivos en el estado confirmado son archivos que se guardaron en el repositorio. git exitosamente.

Por lo tanto, un archivo confirmado es un archivo en el cual has registrado su versión preparada en el directorio (carpeta) Git.

Configuración de un repositorio y comandos en Git

Un repositorio de Git es un almacenamiento virtual de un proyecto. Que permite guardar versiones del código a las que se puede acceder cuando se necesite.

Inicio de un nuevo repositorio: git init

Para crear un nuevo repositorio, se usa el comando git init. git init es un comando que se utiliza una sola vez durante la configuración inicial de un repositorio nuevo. Al ejecutar este comando, se creará un nuevo subdirectorio. git en el directorio de trabajo actual. También se creará una nueva rama principal.

Clonación de un repositorio existente: git clone

Si un repositorio ya se ha configurado en un repositorio central, el comando de clonación es la manera más común de obtener una copia de desarrollo local. Igual que git init, la clonación suele ser una operación única. Una vez que un desarrollador ha obtenido una copia de trabajo, todas las operaciones de control de versiones se administran por medio de su repositorio local.

El comando git clone se usa para crear una copia o clonar un repositorio remoto. Se utiliza git clone con la URL de un repositorio. Git es compatible con varios protocolos de red y sus formatos de URL correspondientes.

Guardar cambios en el repositorio: git add y git commit

Cuando se ha iniciado o clonado un repositorio, se puede realizar commits en la versión del archivo.

Cuando se crea un commit nuevo con un mensaje que describa qué trabajo se ha hecho en el commit.

Colaboración entre repositorios: git push

Es importante comprender que la idea de "copia de trabajo" de Git es muy distinta a la copia de trabajo que se obtiene al extraer código fuente de un repositorio SVN. A diferencia de SVN, Git no distingue entre las copias de trabajo y el repositorio central: todos son repositorios Git completos.

Por tanto, colaborar con Git es intrínsecamente distinto que con SVN. Mientras que SVN depende de la relación entre el repositorio central y la copia de trabajo, el modelo de colaboración de Git se basa en la interacción entre repositorios. En lugar de insertar una copia de trabajo en el repositorio central de SVN, se extraen o envían commits de un repositorio a otro.

Ajustes y configuración: git config

Una vez configurado el repositorio remoto, se debe añadir una URL de repositorio remoto al git config local y configurar una rama de nivel superior para tus ramas locales. El comando git remote ofrece tal utilidad

Este comando asignará el repositorio remoto en una referencia en el repositorio local bajo. Una vez asignado el repositorio remoto se puedes enviar ramas locales ahí.