

Universidad de San Carlos de Guatemala

Facultad de ingeniería

Escuela de estudios de postgrado

Maestría en Ingeniería para la industria

Con especialización en Ciencias de la computación

Fundamentos de programación y scripting

Diana Paola Gutiérrez Lima – 999012382

1. ¿Qué es GIT?

Es una herramienta basada en la nube que permite la gestión y actualización de versiones de softwares. Se suele utilizar cuando se tienen muchos documentos con código fuente, aumentando la eficiencia del control y actualizaciones. También se utiliza para mantener la compatibilidad de las nuevas versiones.

Permite que el trabajo en equipo o dentro de una comunidad sea más sencillo dado que se guarda en la nube y cualquier persona con acceso a internet y al proyecto tendrá la opción de editar los documentos.

Git puede ser categorizada dentro de DVCS el cual es un Sistema de Control de Versiones Distribuido, lo cual se explica al inicio, mantiene un historial de ediciones y actualizaciones.

2. Control de versiones con GIT

Dentro del control de versiones se puede ver los estados del proyecto, de las actualizaciones que se han realizado y los tiempos en los que se realizaron los cambios. Guarda el historial de edición de los archivos. Esta función es sencilla de ejecutar dentro de Git dado que se cuenta con asignación de permisos para edición, ramificación, la que permite tener mayor organización, fusión y comparación con versiones anteriores, lo que mantiene las versiones anteriores y recientes siendo compatibles.

3. Estados de un archivo en GIT

Como se mencionaba anteriormente se tiene un historial de edición de los archivos guardados en un proyecto o repositorio, se tendrá acceso a todos los cambios y las versiones anteriores de los documentos del proyecto.

4. ¿Cómo se configura un repositorio?

Luego de crear un repositorio se pueden aplicar configuraciones básicas como visibilidad, administración de equipos, notificaciones, configuraciones de vínculos, así como la

personalización del repositorios donde pueden configurarse temas, lenguajes, y la gestión de una licencia.

Luego de crear una cuenta, se crea un repositorio donde se solicita: nombre del repositorio, descripción, tipo de visibilidad (pública o privada), seleccionar archivos que no se desea tener historial y se elige una licencia.

5. Comandos en GIT

Los comandos se utilizan para el manejo de documentos, crear, borrar, mover, eliminar (*git clean*), o crear búsquedas. También obtener desde ahí el historial de un documento (*git log*). Además de organizar los documentos también puede organizar las ramas que se tengan creadas y la confirmación de nuevas versiones, como *git checkout*.

Uno de los principales comandos que pueden utilizarse después de instalar Git es *git config*, el cual permite personalizar las configuraciones de instalación. A continuación se mencionan algunos comandos:

- *git checkout*: extrae confirmaciones y sirve para navegar por ramas de repositorios.
- *git clone*: clona un repositorio existente dentro de Git.
- *git commit-amend*: permite la notificación de la última confirmación.
- *git config*: establece opciones para configuración en la instalación de Git.

Referencias

Anónimo, (Noviembre, 2022). Comandos. Lugar de publicación: Atlassian. <https://www.atlassian.com/es/git/glossary>.

Dasdo, (Marzo, 2020). Configuración básica. Lugar de publicación: Github. <https://gist.github.com/dasdo/9ff71c5c0efa037441b6>.

Anónimo, (Febrero, 2023). Administrar la configuración y características de tu repositorio. Lugar de publicación: Github. <https://docs.github.com/es/repositories/managing-your-repositorys-settings-and-features>.