



FUNDAMENTOS DE PROGRAMACIÓN Y SCRIPTING Sección A

Héctor Armando Arévalo Moreno 999007797

HOMEWORK 1

¿Qué es GIT?: Es un sistema utilizado principalmente por los desarrolladores de Software, para el control de las distintas versiones que surjan del programa que se esté desarrollando. También se puede utilizar con cualquier otro tipo de archivos en donde se requiera llevar un control de los cambios realizados en el tiempo y una copia de seguridad de las versiones anteriores.

Control de Versiones con GIT:

Permite registrar cada cambio realizado en el código o archivos, incluyendo cuándo se realizó, quién lo hizo y por qué. Se pueden crear ramas auxiliares **Branching** para trabajar sobre nuevas funcionalidades sin afectar el código principal y posteriormente utilizar **Merging** para fusionar los cambios en la rama principal.

Estados de un Archivo en GIT: En GIT, un archivo puede tener los siguientes estados:

1. **Sin formato:** cuando un archivo es creado o subido al sistema, se encuentra en estado "sin formato" y no puede ser procesado por la plataforma.
2. **Formateando:** este estado indica que el archivo está siendo procesado por la plataforma para poder ser utilizado.
3. **Activo:** cuando el archivo ha sido procesado y está listo para su uso, se encuentra en estado "activo". En este estado, el archivo puede ser utilizado por las aplicaciones que lo requieran.
4. **Archivado:** cuando un archivo no se utiliza con frecuencia, puede ser movido a estado "archivado". En este estado, el archivo no está disponible para su uso directo, pero se puede restaurar a estado "activo" en cualquier momento.
5. **Eliminado:** cuando un archivo ya no es necesario, puede ser eliminado y moverse a estado "eliminado". En este estado, el archivo ya no está disponible y no puede ser restaurado.

Como se configura un Repositorio: Para configurar un repositorio en GIT estos son los pasos por seguir:

1. Ingresar a la página de GIT y hacer clic en el botón "Configuración" en la barra de navegación superior.
2. En el menú de configuración, seleccionar la opción "Repositorios".
3. Seleccionar el tipo de repositorio que se desea configurar (por ejemplo, GIT, SVN, Mercurial).
4. Ingresar las credenciales de acceso al repositorio (nombre de usuario y contraseña o clave de acceso, dependiendo del tipo de repositorio).
5. Configurar los parámetros específicos del repositorio (URL, rama por defecto, subdirectorios, etc.).
6. Hacer clic en el botón "Guardar" para guardar la configuración.

Una vez configurado el repositorio, se pueden obtener los datos necesarios para realizar operaciones con él, como por ejemplo, obtener los archivos del repositorio, hacer una revisión (pull request) o para fusionar cambios (merge) entre ramas.

COMANDOS EN GIT: algunos de los comandos más comunes utilizados en GitHub:

1. **git init:** Inicializa un nuevo repositorio de git vacío.
2. **git add:** Agrega cambios al área de preparación o "staging area".
3. **git commit:** Guarda los cambios realizados en la rama.
4. **git clone:** Crea una copia de un repositorio remoto en una ubicación local.
5. **git pull:** Descarga los cambios en un repositorio local.
6. **git push:** Envía los cambios realizados en un repositorio local a uno ya existente en un servidor remoto.
7. **git branch:** Lista todas las ramas en un repositorio.
8. **git checkout:** Cambia entre diferentes ramas en un repositorio.
9. **git merge:** Combina dos ramas de un repositorio.
10. **git status:** Identifica el estado actual de un repositorio.

Esos son algunos de los comandos básicos, pero hay muchos más que se pueden aprender en el uso diario del repositorio. Cada comando tiene diferentes opciones y argumentos que se pueden utilizar para ajustar su funcionalidad y hacer tareas más específicas.