**Universidad de San Carlos de Guatemala**

**Facultad de ingenieria**

**Escuela de estudios de postgrado**

**Maestría en Ingenieria 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.

1. **Control de versiones con GIT**

Dentro del control de versiones se puede ver los estados del proyecto, de las actulizaciones que se han realizado y los tiempos en los que se realizaron los cambios. Guarda el historial de edición de los archivos.

1. **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.

1. **¿Cómo se configura un repositorio?**

Luego de crear un repositorio se pueden aplicar configuraciones básicas como visbilidad, administracipón 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.

1. **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* *cheeckout*.

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