



ESTUDIA EN EL
INSTITUTO
TECNOLÓGICO DE LAS
AMÉRICAS (ITLA)

Nombre: Ángel Miguel Ramírez Santos

Matricula: 2023-0291

Asignatura: Programación 3

Fecha: 22-11-2024

Profesor: Kelyn Tejada

1. ¿Qué es Git?

Git es un sistema de control de versiones distribuido que permite a múltiples desarrolladores trabajar en el mismo proyecto de software sin sobrescribir los cambios de los demás. Es una herramienta que facilita la gestión del historial de cambios en los archivos de un proyecto, permitiendo crear versiones, revertir a versiones anteriores y colaborar de manera eficiente en equipos.

2. ¿Para qué funciona el comando git init?

El comando git init se utiliza para inicializar un nuevo repositorio de Git en un directorio. Esto crea un repositorio local, que incluye una carpeta oculta llamada .git, donde se almacenan los datos necesarios para el seguimiento de versiones y cambios. Este comando se usa al comenzar un proyecto desde cero con Git.

3. ¿Qué es una rama?

Una rama en Git es una versión paralela del código en la que se pueden realizar modificaciones sin afectar la rama principal o main (también conocida como master). Las ramas permiten trabajar en características o arreglos específicos de manera independiente, y luego combinar esos cambios con la rama principal mediante un proceso llamado *merge*.

4. ¿Cómo saber en cuál rama estoy?

Para saber en cuál rama estoy trabajando, utilizo el comando `git branch`. Este comando lista todas las ramas del repositorio, y marca con un asterisco la rama en la que me encuentro actualmente. También puedo usar `git status` para ver el estado del repositorio, incluyendo la rama actual.

5. ¿Quién creó Git?

Git fue creado por **Linus Torvalds**, el mismo creador del núcleo de Linux, en 2005. Fue desarrollado para gestionar el código fuente del núcleo de Linux, y desde entonces ha crecido hasta convertirse en una herramienta estándar para el control de versiones en proyectos de software.

6. ¿Cuáles son los comandos más esenciales de Git?

Los comandos más importantes o esenciales de Git son:

- `git init`: Inicializa un repositorio Git.
- `git clone`: Clona un repositorio existente.
- `git add`: Agrega cambios al área de *staging*.
- `git commit`: Confirma los cambios agregados en el área de *staging*.
- `git status`: Muestra el estado actual del repositorio.
- `git push`: Sube cambios locales a un repositorio remoto.
- `git pull`: Trae cambios desde un repositorio remoto.
- `git branch`: Muestra, crea o elimina ramas.
- `git checkout`: Cambia de rama o revisa una versión específica.
- `git merge`: Fusiona ramas diferentes en una sola.

7. ¿Qué es Git Flow?

Git Flow es una metodología de trabajo que organiza el desarrollo de software mediante un conjunto de reglas sobre cómo utilizar las ramas en un proyecto con Git. Define un flujo de trabajo estándar que incluye ramas principales (master o main), una rama de desarrollo (develop), ramas de características (feature), corrección de errores (hotfix) y lanzamientos (release), facilitando la colaboración y el mantenimiento del código.

8. ¿Qué es Trunk-Based Development?

Trunk-Based Development es una estrategia de desarrollo que implica trabajar principalmente en una sola rama, llamada trunk o main, en lugar de tener múltiples ramas para cada característica. Los desarrolladores crean cambios pequeños y frecuentes que se integran rápidamente al trunk, evitando largas ramas separadas. Esta metodología promueve la integración continua y reduce los conflictos al fusionar cambios, lo que resulta en una entrega de software más ágil y eficiente.