



Las Americas Institute of Technology

Nombre: Keury Ramirez

Matricula: 2023-1101

Carrera: Desarrollo de software

Materia: Programación III

Docente: Kelyn Tejada Belliard

Tema: Tarea 3

Fecha:

21/11/2024

Tabla de contenido

Introducción	3
1. ¿Qué es Git?	3
2. ¿Para qué funciona el comando git init?	3
3. ¿Qué es una rama?	3
4. ¿Cómo saber en cuál rama estoy?	3
5. ¿Quién creó Git?	3
6. ¿Cuáles son los comandos más esenciales de Git?	4
7. ¿Qué es Git Flow?	4
8. ¿Qué es Trunk-Based Development?	4



Introducción

Git es una herramienta indispensable para desarrolladores, ya que esta nos permite llevar un control super detallado de los cambios que hemos hecho en el proyecto de software. Su enfoque en la colaboración y el manejo de versiones facilita nuestro trabajo en equipo y asegura que los avances en un proyecto sean bien organizados y recuperables. A lo largo de este texto, podremos ver de una forma clara y precisa los conceptos básicos de **Git**, sus **comandos** más importantes, y dos de los **flujos** de trabajo más populares: **Git Flow** y **Trunk-Based Development**.

1. ¿Qué es Git?

Git es una herramienta que nos ayuda a llevar el control de versiones en nuestros proyectos. Esto significa que podemos **guardar**, **revisar** y **regresar** a **versiones** anteriores de nuestros archivos si algo sale mal. Es muy útil cuando trabajamos en equipo, ya que permite que varias personas trabajen en el mismo proyecto sin pisarse el trabajo.

2. ¿Para qué funciona el comando git init?

El comando **git init** sirve para iniciar un repositorio Git en una carpeta. Simplemente le decimos a Git que queremos empezar a seguir los cambios de los archivos que estén ahí.

3. ¿Qué es una rama?

Una **rama** es como una línea de trabajo paralela que está dentro de un proyecto. Imagina que estamos probando una nueva idea, podemos usar una rama para hacer cambios sin afectar el proyecto principal, como una rama llamada cambios, y si todo sale bien, luego se puede unir esa rama con la principal con **Git Merge**.

4. ¿Cómo saber en cuál rama estoy?

Para saber en qué **rama** estás **trabajando**, puedes usar el comando **git branch**. Esto mostrará una lista de **ramas** y **marcará** con un asterisco (*) la rama en la que estás actualmente, esto nos sirve para no perdernos entre ramas, mayormente si se usa **git** desde la consola.

5. ¿Quién creó Git?

Git fue creado por **Linus Torvalds**, la misma persona que desarrolló el sistema operativo **Linux**. Lo creó en **2005** para gestionar el código del **kernel** de **Linux**.

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

Algunos de los comandos más importantes de Git son:

- ❖ **git init:** Para iniciar un repositorio.
- ❖ **git add:** Para agregar archivos/cambios que no han sido agregados.
- ❖ **git commit:** Para guardar los cambios con un mensaje que explique que se hizo.
- ❖ **git status:** Para ver el estado de los archivos y ramas.
- ❖ **git branch:** Para ver y trabajar con ramas.
- ❖ **git merge:** Para combinar ramas.
- ❖ **git pull y git push:** Para subir y bajar cambios de un repositorio remoto, como GitHub, (Recomiendo utilizar **Pull** primero para descargar los cambios del repositorio y aplicar el **Push** después para asegurarnos de que está actualizado en su totalidad y de que todo permanece en su sitio).

7. ¿Qué es Git Flow?

Git Flow es una forma organizada de trabajar con ramas en Git. Divide el trabajo en ramas específicas: la rama principal, la de desarrollo, y ramas para funcionalidades, correcciones de errores, y lanzamientos. Es útil para equipos grandes que necesitan un flujo de trabajo claro.

8. ¿Qué es Trunk-Based Development?

Trunk-Based Development es otro estilo de trabajo con Git, pero más simple que Git Flow. Aquí, la mayoría de los cambios van directo a la rama principal (trunk), y las ramas suelen ser cortas y rápidas de fusionar. Es ideal para equipos que buscan entregar cambios rápido y con frecuencia.



9. Fuentes Bibliográficas

1. <https://git-scm.com/book/en/v2/Getting-Started-A-Short-History-of-Git>
2. <https://www.atlassian.com/es/git/tutorials/what-is-git>
3. <https://www.hostinger.es/tutoriales/comandos-de-git>
4. <https://es.wikipedia.org/wiki/Git>
5. <https://www.freecodecamp.org/espanol/news/hoja-de-referencia-de-git-50-comandos-de-git-que-debe-conocer-2/>

Este documento ha sido **completamente** realizado por **Keury Ramirez**, no hubo uso de **IAs** de **Ningún** tipo, el tema es bastante interesante y temas como este considero son importantes de profundizar al menos una vez para tener conocimientos de esto, espero le haya gustado esta edición, me aseguré de que cumple con todos los criterios, gracias de antemano,

feliz resto del día!

