

Título: “Tarea 3 – Git y Git Flow”

**Nombre completo: Jennipher Johanny
Adames Carrasco**

Matrícula: 2022-0495

Asignatura: Programación III

Profesor: Kelyn Tejada

Fecha de entrega: 4/4/2024

Índice

Portada	PG1
Indice	PG2
¿Qué es Git?	PG3
¿Para qué sirve el comando git init?	PG3
Qué es una rama en Git	PG3
Cómo saber en cuál rama estoy trabajando	PG4
Quién creó Git?	PG4
Cuáles son los comandos esenciales de Git?	PG5
Qué es Git Flow?	PG6
Qué es el desarrollo basado en trunk (Trunk Based Development)	PG6
Bibliografía	PG7

1-¿Qué es Git?

es un sistema de control de versiones que permite a los desarrolladores administrar el código fuente de un proyecto.

Es una herramienta gratuita y de código abierto que se utiliza para proyectos de software comerciales y de código abierto.

Git es un sistema de control de versiones distribuido , gratuito y de código abierto, diseñado para gestionar todo, desde proyectos pequeños hasta muy grandes, con velocidad y eficiencia.

2-¿Para qué sirve el comando git init?

El comando git init sirve para crear un repositorio de Git. Se utiliza una sola vez al configurar un nuevo repositorio. Al ejecutarlo, se crea una carpeta oculta de llamada .git que contiene toda la información necesaria para el control de versiones del proyecto.

3-¿Qué es una rama en Git?

es una versión de un repositorio, es decir, una línea de desarrollo independiente. Las ramas son útiles para controlar el código, desarrollar características, corregir errores, o experimentar con nuevas ideas., ademas las ramas facilitan el trabajo en equipo y la experimentación sin comprometer el código estable

4-¿Cómo saber en cuál rama estoy trabajando?

Para saber en qué rama de Git estás trabajando, puedes usar el comando git branch. Este comando te mostrará todas las ramas locales y marcará con un asterisco (*) la rama activa,

además muchas personas utilizan `git status` que muestra información sobre la rama activa y sus cambios pendientes.

5-¿Quién creó Git?

Linus Torvalds creó Git en 2005. Torvalds es el famoso programador que creó el kernel del sistema operativo Linux.

6-¿Cuáles son los comandos esenciales de Git?

Algunos de los comandos esenciales de Git son:

`git init`: Crea un nuevo repositorio de Git

`git clone`: Descarga el código fuente de un repositorio remoto

`git branch`: Crea, lista, o elimina ramas

`git checkout`: Cambia de rama, chequea archivos y commits

`git status`: Muestra información sobre la rama actual

`git add`: Incluye los cambios de archivos en tu siguiente commit

git commit: Guarda tus cambios

git push: Envía tus commits al repositorio remoto

git pull: Recibe actualizaciones del repositorio remoto y las aplica inmediatamente en local

git merge: Fusiona la rama en la que estás trabajando con su rama padre

Otros comandos de Git son:

git archive, git bisect, git cherry-pick, git fsck, git gc, git rebase, git remote add, git stash, git tag, git revert

7- ¿Qué es Git Flow?

es un modelo de ramificación de Git que utiliza ramas de características y varias ramas principales. Git Flow es un modelo de desarrollo basado en ramas que organiza el trabajo con un flujo estructurado. Defina ramas principales (main y develop) y ramas auxiliares (feature, release y hotfix), facilitando el desarrollo, pruebas y despliegue de software.

8- ¿Qué es el desarrollo basado en trunk (Trunk Based Development)?

El desarrollo basado en troncos garantiza que los equipos publiquen el código de forma rápida y coherente. A continuación, se presenta una lista de ejercicios y prácticas que te ayudarán a perfeccionar el ritmo de trabajo de tu equipo y a desarrollar un calendario de publicaciones optimizado.

Bibliografías

- <https://git-scm.com/about/free-and-open-source>
- <https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow>
- <https://www.swhosting.com/es/blog/comandos-basicos-de-git-guia-para-principiantes>

