

# Tarea 1: GitHub, Pytest, Flake 8

Gabriel Devandas-Mata  
gabodeva@estudiantec.cr

Área Académica de Ingeniería Mecatrónica  
Instituto Tecnológico de Costa Rica  
MT7003 Microprocesadores y Microcontroladores

## I. ¿DIFERENCIE LA HERRAMIENTA GIT DE GITHUB?

### Git

- Para usar Git, cada desarrollador debe tener instalada la herramienta en su dispositivo local para colaborar.
- Git permite mantener un historial de commits, invertir cambios y compartir código.
- Usa los repositorios guardados localmente en cada computador.
- No posee una GUI propia.
- Sus comandos propios como commit y merge deben ser trabajados en formato de línea de código.
- Se concentra en tareas de SCM como commit, pull, fetch, merge, reset.

### GitHub

- GitHub es un servicio basado en web para el hosting de repositorios de Git.
- Además de funciones Git, permite funciones extra como administración de proyectos, tiquetes de soporte y rastreo de bugs.
- Se mantiene almacenado a la nube, por lo que requiere acceso a internet.
- Posee una GUI amigable para el usuario.
- Las herramientas como commit pueden ser accesadas con una interfaz gráfica.
- Funciona más como un host para almacenar código en un lugar centralizado.

## II. ¿QUÉ ES UN BRANCH?

Se refiere a una bifurcación de la línea de trabajo original, en ciertos casos es útil trabajar en una línea de desarrollo paralela que permite agregar nuevas funciones al código principal. En git, su comando crea un nuevo apuntador en el commit que se encuentre en ese momento.

## III. ¿QUÉ ES UN COMMIT?

Es una “captura instantánea” de los cambios realizados hasta el momento a un proyecto, git guarda este commit como un objeto que contiene un apuntador a la instantánea del contenido elegido, este también contiene el nombre y correo del autor, un mensaje agregado y apuntadores al commit y a otros directamente anteriores a este.

## IV. ¿QUÉ ES LA OPERACIÓN CHERRY-PICK?

Es una operación que permite elegir un commit de una rama y aplicarlo a otra, puede ser usado para invertir un cambio; se debe manejar con cuidado ya que puede duplicar commits no deseados.

## V. ¿QUÉ HACE EL COMANDO GIT STASH?

Guarda en otro lugar las modificaciones locales realizadas en un directorio y revierte el directorio para que coincida con el commit HEAD.

## VI. ¿COMPARE LAS OPERACIONES GIT FETCH Y GIT PULL?

El comando fetch hace que nuestro git local recupere la última información de la rama u objeto original que estemos buscando de uno o más repositorios, pero no transfiere estos datos al dispositivo local. Por otro lado, el comando pull realiza primero la acción fetch, recuperando los datos de rama y apuntadores relacionados de otro repositorio, y llama al comando merge para unir los datos de rama recuperados con la rama seleccionada localmente.

## VII. ASUMIENDO QUE USTED ESTÁ EN UN BRANCH LLAMADO “SECUNDARIO” Y SU BRANCH PRINCIPAL SE LLAMA “MASTER”

### VII-A. ¿Qué resultado espera de hacer git rebase master?

El branch secundario se va a mover y empezará a un extremo del master, y los commits de ambos se van a reorganizar y reescribir en el historial del branch principal.

### VII-B. ¿Qué resultado espera de hacer git rebase origin/master?

Realizará el reordenamiento de la rama seleccionada en la rama “master” de un repositorio remoto llamado en caso de encontrarlo.

## VIII. ¿QUÉ ES UNA PRUEBA UNITARIA O UNITTEST EN EL CONTEXTO DE DESARROLLO DE SOFTWARE?

Es un método que establece si uno o varios módulos de cierto código están funcionando correctamente, estas pruebas son utilizadas cuando un proyecto experimenta modificaciones para asegurar que este siga funcionando apropiadamente. Normalmente se implementan durante todo el desarrollo de un módulo o proyecto.

## IX. BAJO EL CONTEXTO DE PYTEST, ¿QUÉ ES UN “ASSERT”?

Es un módulo que nos permite verificar valores devueltos por funciones presentes en código Python que deseemos probar, al utilizar el módulo en un ambiente de prueba, si el valor retornado por la función no cumple con los parámetros del valor que esperábamos (ingresados anteriormente), el assert devolverá un error.

## X. ¿QUÉ ES FLAKE 8?

Es una utilidad para línea de comandos que busca hacer cumplir una consistencia de estilo entre proyectos de Python, que incluye una serie de programas de “linting” y verificación de código. Buscando que los programas escritos sigan ciertos estándares y evitando errores de sintaxis, formato.

## REFERENCIAS

- [1] K. Holger, “The writing and reporting of assertions in tests — pytest documentation”, Docs.pytest.org, 2021. [Online]. Available: <https://docs.pytest.org/en/stable/assert.html#assert>. [Accessed: 04- Mar- 2021].
- [2] “Git - Rebasing”, Git-scm.com, 2021. [Online]. Available: <https://git-scm.com/book/en/v2/Git-Branching-Rebasing>. [Accessed: 04- Mar- 2021].
- [3] I. Staplestone, “Flake8: Your Tool For Style Guide Enforcement — flake8 3.8.4 documentation”, Flake8.pycqa.org, 2021. [Online]. Available: <https://flake8.pycqa.org/en/latest/>. [Accessed: 05- Mar- 2021].
- [4] “Git - Branches in a Nutshell”, Git-scm.com, 2021. [Online]. Available: <https://git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell>. [Accessed: 05- Mar- 2021].