



TEC | Tecnológico
de Costa Rica

Área Académica de Ingeniería Mecatrónica

MT7003 - Microprocesadores y microcontroladores

Tarea 1: GitHub, Pytest y Flake 8.

Profesor: Rodolfo Piedra Camacho.

Estudiantes:

Jose Andrés Aguilar Salas.

Diego Mora.

I Semestre, 2021.

Preguntas teóricas:

1) ¿Diferencie la herramienta Git de Github?

La herramienta Git consiste en un software de control de versiones el cual está enfocado en la eficiencia y confiabilidad del mantenimiento de versiones. Tiene como objetivo llevar el registro de los cambios en archivos de computadora y coordinar el trabajo que se realiza en conjunto entre varias personas sobre archivos compartidos. Por su parte, Github es un servicio gratuito que consiste en un sitio web con la implementación de servicio en la nube, que ayuda a los usuarios permitiéndoles la posibilidad de almacenar y administrar su código. Además, permite llevar un registro de control en cualquier cambio sobre el código.

2) ¿Qué es un branch?

Desde el punto de vista de Git, un Branch es una rama dentro del sistema de control de versiones. En dicho sistema de control, se puede observar el histórico de cambios como si se tratase de un árbol. Así mismo, es posible abrir ramas que parten de la rama principal llamada master o de otra rama (Branch).

3) ¿Qué es un commit?

Es un estado en el que se puede encontrar un archivo. Suponiendo que el repositorio local utilizado se compone por tres “árboles”, el directorio de trabajo que contiene los archivos, el index el cual es un intermediario, y el head, este último apunta al último commit realizado.

4) ¿Qué es la operación cherry-pick?

La ejecución cherry-pick se refiere a elegir una confirmación de una rama y aplicarla a otra. Este comando permite que las confirmaciones arbitrarias de Git se elijan por referencia y sean añadidas al actual head de trabajo.

5) ¿Qué hace el comando git stash?

Este comando guarda los cambios en un directorio de trabajo sucio. Es útil usarlo cuando se desea registrar el estado actual del directorio de trabajo y el índice, pero además se desea volver a un directorio de trabajo limpio. El comando guarda las modificaciones locales y revierte el directorio de trabajo para que coincida con la confirmación del head.

6) ¿Compare las operaciones git fetch y git pull

Estas operaciones son usadas para descargar datos nuevos desde un repositorio. El comando git fetch descarga nuevos datos de un repositorio, sin embargo no integra ningún dato nuevo en sus archivos de trabajo, por lo que no es posible que elimine o borre algún dato. El comando git pull, por su parte, lo que realiza es una actualización de su rama head actual con los últimos cambios del servidor remoto. Por lo que además de descargar nuevos datos, los integra de forma directa en sus archivos actuales de copia de trabajos.

7) Asumiendo que usted está en un Branch llamado “secundario” y su Branch principal se llama “master” ¿Qué resultado espera de hacer git rebase master? ¿Qué resultado espera de hacer git rebase origin/master?

En Git, una manera de integrar cambios de una rama en otra es utilizando la herramienta rebase para la reorganización. Con el comando git rebase, es posible capturar todos los cambios confirmados en una rama y aplicarlos sobre otra. En lugar de enlazar las ramas con un commit de merge, el hacer rebase master a una rama mueve completamente la rama con la nueva característica hacia la punta de master. Git rebase origin/master se utiliza de forma muy útil en proyectos de colaboración, cuando no se lleva el mantenimiento propiamente, de forma que sea posible trabajar sobre una rama y luego reorganizar lo realizado en la rama para enviarlo al proyecto principal cuando esté completamente listo. Si lo que se desea es rebasar una rama basada en la rama maestra remota, git rebase origin / master no es suficiente.

8) ¿Qué es una Prueba Unitaria o Unittest en el contexto de desarrollo de software?

Una prueba unitaria, o “unit test”, es un método que prueba una unidad estructural de código. Por lo general son simples y rápidas de codificar. Facilitan la reestructuración del código, pues permiten realizar pruebas sobre los cambios y verificar que las modificaciones no han introducido errores, además, permiten llegar a la fase de integración asegurando que las partes individuales funcionan de forma correcta.

9) Bajo el contexto de pytest. ¿Qué es un “assert”?

Un assert es una herramienta de Pytest para verificar expectativas y valores en pruebas de Python. Por ejemplo, para corroborar que una función devuelve un cierto valor. Si esta afirmación falla, verá el valor de retorno de la llamada a la función. Es posible ver un ejemplo en el siguiente código:

```
def f():
    return 3
def test_function():
    assert f() == 4
```

```
test_assert1.py F [100%]
===== FAILURES =====
test_function
> assert f() == 4
E assert 3 == 4
E + where 3 = f()
test_assert1.py:5: AssertionError
===== 1 failed in 0.12 seconds =====
```

Obtenido de: <https://docs.pytest.org/en/latest/assert.html#assert>

10) ¿Qué es Flake 8?

Es una herramienta la cual permite revisar si existen variables declaradas pero que no están siendo utilizadas. Permite que el código resultante sea más limpio.

Bibliografía:

Documentación oficial de pytest: Obtenido de:

<https://docs.pytest.org/en/latest/gettingstarted.html>

Git Immersion: Obtenido de: <http://gitimmersion.com/>

Git: fast version control. Obtenido de: <https://git-scm.com/>

Git: la guía sencilla. Obtenido de: <http://rogerdudler.github.io/git-guide/index.es.html> Pytest: helps you write better programs. Obtenido de: <https://docs.pytest.org/en/latest/>

Linux System Errors. Obtenido de:

https://wwwnumi.fnal.gov/offline_software/srt_public_context/WebDocs/Errors/unix_system_errors.html

¿Qué es GitHub? Una Guía para Principiantes sobre GitHub. Obtenido de:

<https://kinsta.com/es/base-de-conocimiento/que-es-github/>

¿Qué es y para qué sirve Github? Obtenido de:

<https://www.deustoformacion.com/blog/programacion-diseno-web/que-es-para-que-sirve-github>

Pruebas unitarias. Obtenido de:

<http://www.microgestion.com/index.php/mg-developers/articulos/74-unit-test-part1-mock>

Ramificaciones en Git - Reorganizar el Trabajo Realizado. Obtenido de: [Git - Reorganizar el Trabajo Realizado \(git-scm.com\)](#)