

Instituto Tecnológico de Costa Rica

Área Académica de Ingeniería Mecatrónica

MT-7003 Microprocesadores y Microcontroladores

Tarea 1: GitHub, Pytest y Flake8

MSc. Felipe Meza Obando

Por: Aharon Moya Chavarría

Javier A. Hernández Castro

I semestre, 2021

Preguntas Teóricas (20 pts, 2pts c/u)

1. ¿Diferencie la herramienta Git de Github?

El software Git permite el control de versiones de un código fuente de manera distribuida. Los programadores pueden instalar el programa localmente y de esta manera poder modificar el código y observar los cambios realizados por los otros colaboradores. En lugar de tener un único espacio para todo el historial de versiones del software, que es el caso de los sistemas de control de versiones utilizadas antiguamente, al usar Git, la copia del código de cada programador también consiste en un repositorio que puede tener el historial de todos los cambios.

La herramienta de Github consiste en una plataforma de software libre de desarrollo colaborativo, derivada de Git, la cual se puede entender como una red social para programadores. Esta permite trabajar en colaboración con otras personas de todo el mundo, planificar proyectos y dar seguimiento. De esta manera, los usuarios pueden compartir sus proyectos y las personas de cualquier parte del mundo pueden trabajar en paralelo en ellos.

2. ¿Qué es un branch?

Un branch es una copia temporal del código o de otro branch; este permite realizar tareas de desarrollo sin afectar otros branches en el repositorio. Cada repositorio tiene un branch por defecto o branch principal y puede tener muchos branches. Finalmente, cuando se desee incluir los cambios realizados en el branch auxiliar en el branch principal se utiliza el comando merge, que permite combinarlos.

3. ¿Qué es un commit?

Es una acción que permite que se realicen cambios en la rama en la que se está trabajando. Estos cambios incluyen una breve descripción de lo que se está modificando, autor y fecha. Además, pueden ser locales o remotos. Se pueden entender como una captura instantánea de los cambios realizados en ese momento. Esas capturas confirmadas son consideradas como versiones seguras del proyecto. Antes de utilizar git commit, se usa el comando git add para preparar o pasar los cambios en el proyecto que se van a almacenar al confirmar.

4. ¿Qué es la operación cherry-pick?

Este comando permite tomar uno o varios commits de otra rama sin tener que hacer un merge completo. El comando cherry pick toma los cambios de un commit seleccionado y los

coloca en el head de la última rama a la cual se le aplicó el comando checkout. Para usar este comando lo único que se requiere es el commit específico que se quiere aplicar a la rama actual. Se debe tener en cuenta que este comando crea un nuevo commit, por lo que antes de usar cherry pick no se debe tener ningún archivo modificado que no haya sido incluido en un commit.

5. ¿Qué hace el comando git stash?

Git stash consiste en un comando que almacena temporalmente los cambios efectuados en el código en el cual se está trabajando, de esta manera se puede trabajar en otro código y después regresar y aplicar los cambios más tarde. Poder guardar los cambios en stashes es efectivo cuando se tiene que cambiar rápidamente de tarea o trabajo, pero se está trabajando en un cambio en el código y este aún no se encuentra listo para confirmar los cambios.

6. ¿Compare las operaciones git fetch y git pull

El comando “git fetch” sirve para descargar commits, archivos y referencias de un repositorio externo con el objetivo de incluirles en el repositorio local o actual. Se utiliza para ver el trabajo de los demás colaboradores y permite una visualización del progreso del historial central. Además, este comando no fusiona automáticamente los cambios en el repositorio actual. [3]

Ese último detalle hace de “git fetch” una operación más segura y menos destructiva que “git pull”. Según [3], el comando “git pull” se utiliza “para extraer y descargar contenido de un repositorio remoto y actualizar al instante el repertorio local para reflejar ese contenido.” Se puede entender el comando como un “git fetch” seguido de un “git merge”. El problema de cambiar el contenido del repositorio local de forma automática es que se pueden dar conflictos de contenido, ejecución, etc. [3]

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?

Rebase es un comando similar a merge, por el hecho de que integra los cambios de otra rama en la rama local. La diferencia es que “rebase” comprime los commits en un patch para luego aplicarlos a la rama en la que se está trabajando. Además, borra la parte trivial del historial.

Con “rebase master” los cambios se hacen en el branch original. Un “rebase origin/master” opera sobre una rama remota cuya rama local se llama master. [4]

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

Una Unittest es una manera de comprobar el correcto funcionamiento de una parte o extracto de un código. Constituye uno de los tantos procedimientos de una metodología de trabajo ágil. Estas ayudan a la mejora del producto final. Consisten en aislar parte del código para comprobar si opera como debería. Estos validan el comportamiento del objeto y la lógica, y permiten detectar errores que serían imposibles de identificar en etapas más avanzadas del flujo de diseño. Algunas de las ventajas que ofrecen son la comprobación del estado y funcionalidad de la lógica implementada, el aumento de la legibilidad del código y la documentación del proyecto. Además, se pueden ejecutar en intervalos de tiempo sumamente cortos. Estas también ayudan a refactorizar el código, mejorar la calidad del resultado final y probar partes que no funcionan por falta de etapas posteriores en el flujo del diseño. [5]

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

Pytest es una herramienta para realizar un desarrollo dirigido por pruebas o TDD. La función “assert” sirve como colchón para posibles errores que el programador cometa. Cuando existe inseguridad respecto al comportamiento de una función, la función assert permite asegurar el comportamiento deseado y si se incumple la condición, interrumpe el programa. Se le pueden agregar mensajes que indiquen en la consola la raíz de la interrupción del programa. Esta función se utiliza para condiciones que no deben suceder bajo ninguna circunstancia. Si se es consciente de un comportamiento no deseado, se recomienda usar otro tipo de enfoque (ej. un condicional if). En el entorno de pytest, la función assert se utiliza mayoritariamente para tests unitarios. De esta forma, es posible comprobar que el retorno de una función es el esperado. Asimismo, esta función ayuda a cuantificar la cantidad de resultados erróneos que se obtengan de una unittest. [6]

10. ¿Qué es Flake8?

Flake8 permite revisar que se cumpla PEP8, que es una guía sobre como escribir código en Python y que obedece una serie de reglas. Estas reglas mantienen cierto orden y etiqueta dentro de un proyecto colaborativo. Flake8 también ayuda a revisar las variables e imports que no se usan. A manera de resumen, ayuda a que el código sea más limpio. Adicionalmente, puede suprimir advertencias y es extensible. [7]

Bibliografía

- [1] Git-scm.com. n.d. *Git - Account Setup and Configuration*. [online] Available at: <<https://git-scm.com/book/en/v2/GitHub-Account-Setup-and-Configuration>> [Accessed 5 March 2021].
- [2] Git-scm.com. n.d. *Git - gitglossary Documentation*. [online] Available at: <<https://git-scm.com/docs/gitglossary>> [Accessed 5 March 2021].
- [3] Atlassian. n.d. *git pull / Atlassian Git Tutorial*. [online] Available at: <<https://www.atlassian.com/es/git/tutorials/ syncing/git-pull>> [Accessed 4 March 2021].
- [4] Git-scm.com. n.d. *Git - Reorganizar el Trabajo Realizado*. [online] Available at: <<https://git-scm.com/book/es/v2/Ramificaciones-en-Git-Reorganizar-el-Trabajo-Realizado>> [Accessed 4 March 2021].
- [5] YeePLY. 2020. *¿Qué son las pruebas unitarias y cómo llevar una a cabo?*. [online] Available at: <<https://www.yeeply.com/blog/que-son-pruebas-unitarias/>> [Accessed 4 March 2021].
- [6] Mushr00m, T., 2017. *Desarrollo dirigido por pruebas - Python y Pytest / La Espora Del Hongo*;*. [online] La Espora Del Hongo;*. Available at: <<https://laesporadelhongo.com/desarrollo-dirigido-por-pruebas-python-pytest/>> [Accessed 4 March 2021].
- [7] Navarro, E., 2016. *Tips para desarrollo en python* . [online] Codeandoando.com. Available at: <<https://codeandoando.com/tips-para-desarrollo-en-python/#:~:text=Otra%20herramienta%20bastante%20%C3%BAtil%20es,c%C3%B3digo%20resultante%20sea%20m%C3%A1s%20limpio>> [Accessed 4 March 2021].