

Tarea 1

GitHub, Pytest y Flake 8

A continuación, se presentan 2 asignaciones para entender y aplicar herramientas de desarrollo de proyectos de programación.

Preguntas Teóricas (20 pts, 2pts c/u)

- 1) ¿Explique la principal utilidad de git como herramienta de desarrollo de código?

Hubo un problema en el mundo de la programación que fue que en su momento no existía una forma de respaldo del progreso que se llevaba a cabo en cada programa, y si se hacía algún cambio negativo no había forma de volver al estado anterior a menos que se hubiera hecho una copia antigua por alguna parte de la computadora en uso. Git es la solución a ese problema, ya que no sólo obtenía “capturas” de diferentes versiones del documento a través del tiempo, sino que también lo hacía localmente, sin necesidad de conectarse a una nube para poder trabajar los archivos contenidos en el máster con una copia local del registro en cada computadora implicada en el proceso, a lo cual se llama un proceso distributivo.

- 2) Explique la diferencia entre git y github

Las mayores diferencias que se encuentran son con respecto a como un nuevo cambio es introducido y la forma en que los usuarios se conectan por un medio, GitHub es ese medio, pero el VCS (Versión Control System) es Git. Se podría decir que es una adhesión al programa existente que permite un uso masivo de la plataforma en donde diferentes personas pueden colaborar, revisar los cambios antes de introducirlos en el main, detección de errores en las nuevas ramas a implementar, entre otros.

- 3) ¿Qué es un branch?

Una rama es un trabajo de programación aparte que se desea modificar para añadir características, arreglar errores o probar qué haría un cambio en la rama principal, si estos cambios se desean implementar en el programa completo, entonces se fusionan las dos ramas.

- 4) En el contexto de github. ¿Qué es un Pull Request?

El pull request o PRs, es una forma en que los colaboradores dentro de un proyecto pueden anunciar los cambios en la fusión de dos ramas, dichos colaboradores pueden comentar, aprobar cambios o agregar commits antes de hacer el cambio. En resumen, es un tipo de anuncio para que todos los colaboradores estén de acuerdo con el cambio y puedan aportar ideas o más características.

- 5) ¿Qué es un commit?

Un commit es un “snapshot” del documento en ese momento, son como las versiones de un archivo, pero conteniendo como se veía el archivo en ese momento; posee una etiqueta la cual proporciona información sobre la fecha de modificación y una descripción de los cambios que se hicieron, así como un número hexadecimal que es único para identificar el cambio en la línea de tiempo.

- 6) Describa lo que sucede al ejecutar la siguiente operación: “git rebase main”.

Este comando funciona para reposicionar los commits (en realidad copiarlos) de una rama diferente a la rama principal, produciendo así un mejor orden de los snapshots para poder seguir los cambios en el tiempo. Varía del merge en que los cambios en los commits son copiados enteramente en la rama principal y en el merge son combinados en un solo snapshot cuando se requiera, lo que hace que se pierda un poco el seguimiento de los cambios.

- 7) Explique que es un “merge conflict” y como lo resolvería.

Un merge conflict sucede cuando hay modificaciones que se hicieron en diferentes ramas, pero de una misma característica del programa, lo que conlleva a que haya un problema si se quieren unir las dos, ya que se sobrelapan. Pero lo anterior no ocurre, ya que se impide por medio de git en la mayoría de los casos. La manera de resolverlo es cambiando la característica que se modificó en alguno de los dos archivos, dejando que una de ellas predomine.

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

Una prueba unitaria hace referencia a probar una única función dentro del desarrollo del programa, un ejemplo puede ser el desarrollo de una calculadora, se desea saber si la multiplicación, división y suma están bien implementadas antes de unirlo al programa principal, entonces se hace una prueba a cada función por aparte, las cuales deben cumplir ciertos requerimientos, si hay algún problema se aísla y se soluciona por aparte a la implementación completa. Es muy útil a lo largo de un proyecto, ya que se verifica que cada parte por separado debe hacer lo que fue destinada o intencionada a hacer.

- 9) Bajo el contexto de pytest. ¿Cuál es la utilidad de un “assert”?

Un assert es la manera de hacer pruebas unitarias, su función es retornar un valor de verdadero si la expectativa de la función se cumplió, o en su defecto un AssertionError en caso de ser falso y, por ende, no cumplido.

- 10) Mencione y explique tres errores de formato detectables con Flake8.}

F401: Se importó un módulo, más no se usó, esto sucede cuando hay módulos que se importan por pensar que son necesarios, pero no se usa ninguna función de los mismos, por lo que se consumen recursos, este error lo evita.

F701: Una condición de “break” fuera de un while o un for. Es permitido usarse, mas no tiene sentido usarlo fuera de esos dos bucles, por lo que ensucia el código, que es la razón por la cual flake8 fue creado, para hacer un código más limpio, optimizado y se podría decir que estandarizado también.

E117: Sobre indentado, hace referencia a dejar un espacio de más de múltiplos de 2, un ejemplificador es el de colocar 5 espacios antes de colocar un print. La mejor practica seria dejarlo en 4 para mantener un orden en cada línea.