

1.

Git: sistema de control de código abierto (en inglés: versión control system - VCSs). Los usuarios tienen la capacidad de guardar diferentes versiones de un proyecto gracias a esta herramienta. Esto es útil puesto que pueden regresar a cualquier versión anterior si así lo consideran necesario. Esta herramienta te permite también hacer “push” y “pull”, opciones que son realizables desde y hacia cualquier computadora, permitiendo así que varios usuarios trabajen en el proyecto desde su ordenador. Su única desventaja parece ser que no se pueden ver las ediciones de otras personas en tiempo real, así que Git es bueno para trabajar individualmente o grupalmente en tiempos diferentes.

Github: Github es un repositorio que guarda versiones de proyectos en la nube. Es muy similar a git, pero con la diferencia de que esta herramienta sí permite ver los cambios realizados por otros colaboradores en tiempo real. Los repositorios son públicos y se pueden realizar 3 acciones principales: Fork (copiar el código de otra persona para modificarlo), Pull (compartir tu código al creador original del código) y Merge (Los creadores pueden realizar cambios en su código y dar crédito a la persona que lo mejoró). [7]

2.

El término “branch” se refiere a cuando el programa recibe una instrucción que le indica que debe escoger entre dos o más opciones, fragmentando la linealidad del algoritmo y sometiéndolo a elegir uno de los caminos separados. A la hora de programar, uno de los comandos más utilizados es el “if”, cuando se cumple o no cierta condición, el camino que toma el programa será distinto.

En concreto, branch también se puede referir a descargar el código de otro programador y trabajar en él por separado. Al finalizar, puedes volver al “branch” original y contribuir activamente al código que has mejorado. Un branch separado es un camino que sale del camino original del código del programador para intentar mejorarlo [1]

3.

Un commit tiene mucha relación con el repositorio descrito anteriormente. Cuando un grupo de personas trabaja en un mismo material, cometer errores es muy sencillo. Gracias a esto, el programa permite tener varias versiones del archivo y si ocurre un error, se puede volver a una versión anterior del trabajo. Se utiliza software de control como git. Al realizar un commit, tu versión del código queda guardada en la nube y se puede acceder a este en caso de haber cometido errores luego. [10]

4.

El comando Cherry-pick es un comando en git que te permite integrar commits elegidos de diferentes ramas a la rama principal como gustes, sin embargo, no es el comando más útil porque puede crear duplicados de commits. [5]

5.

El comando git stash se usa cuando se desee registrar el estado actual del directorio de trabajo y el índice, pero se desea volver a un directorio de trabajo limpio. Este comando guarda las modificaciones y revierte el directorio de trabajo para que coincida con el commit principal. [4]

6.

Ambos comandos se utilizan para descargar información de un repositorio online, solo que fetch únicamente descarga los datos, pero no integra nada de eso a tus archivos ya existentes. Su principal uso es por si se necesita analizar lo que se decargó con más detenimiento antes de incorporarlo al código. Por otro lado, pull tiene una utilidad diferente, ya que este comando integra directamente los cambios del server remoto en tu programa. Se debe utilizar este comando con cuidado ya que puede ocasionar errores comunes. [6]

7.

El comando rebase se utiliza para mezclar los cambios de un branch a otro. Funciona parecido a Merge, solo que con esta herramienta puedes agarrar todos los cambios que se hicieron en un branch secundario y replicarlo. En el ejemplo, se copiarán todos los cambios del branch secundario al branch máster. Git rebase origin/master se refiere a hacer rebase desde la rama maestra al origen. [3]

8.

El Unit testing es utilizado a nivel de desarrollo de software y, como su nombre lo indica, es el proceso de revisar cada compenente y hacerle diversas pruebas con el propósito de saber si están funcionando correctamente o no. Realizar este paso es muy importante, debido a que una unidad, por más pequeña que sea puede malfuncionar y esto puede provocar consecuencias severas al software. Esto se traduce también a funciones y métodos dentro del programa. [9]

9. Un assert en pytest se refiere a una afirmación única dentro del método o función. Por ejemplo, para pruebas unitarias (unit tests), cada una de las funciones que se tienen deberían tener un único argumento assert, de manera que, si son llamadas desde el programa principal, nosotros deberíamos poder recibir una cantidad de argumentos assert coherente conforme lo pedido en las líneas de código. [8]

10. Flake8 es una librería en Python que recoge elementos de otros códigos. Su principal uso es que puedes utilizar esta librería para comparar tu trabajo actual con algun código estándar de programación, como lo es PEP8. Gracias a eso se puede identificar y arreglar errores, y se pueden identificar todos los caminos independientes que posee el código. Esto último es conocido como la complejidad cíclica y esta aumenta dependiendo de cuantos caminos diferentes tenga el código. Los IFs suelen abrir varios caminos, aumentando la complejidad. [2]

.

Bibliografía:

1. () Branching. Available: <https://www.cs.utah.edu/~germain/PPS/Topics/branching.html>
2. Freitas, V. () How to use Flake8. Available: <https://simpleisbetterthancomplex.com/packages/2016/08/05/flake8.html#:~:text=Flake8%20is%20a%20Python%20library,and%20to%20check%20cylomatic%20complexity>
3. Git () 3.6 Git Branching – Rebasing. Available: <https://git-scm.com/book/en/v2/Git-Branching-Rebasing>
4. Git () Git Stash. Available: <https://git-scm.com/docs/git-stash>
5. Git FAQ. () The Git cherry-pick command: what it is and how to use it. Available: <https://www.git-tower.com/learn/git/faq/cherry-pick/>
6. Git FAQ () What's the difference between git fetch and git pull. Available: <https://www.git-tower.com/learn/git/faq/difference-between-git-fetch-git-pull/>
7. Kinsta (2021) Git vs Github: What's the Difference and How to Get Started with Both. Available: <https://kinsta.com/knowledgebase/git-vs-github/>
8. Pytest (2015) Assert statements. Available: https://docs.pytest.org/en/reorganize-docs/newdocs/user/assert_statements.html
9. Software Testing Fundamentals (2021) Unit Testing. Available: <https://softwaretestingfundamentals.com/unit-testing/>
10. Su, K. (2014) R/What exactly does "committing code" mean and how is it done at different companies? Available: <https://www.quora.com/Whatexactly-does-committing-code-mean-and-how-is-it-done-at-differentcompanies>

Instituto Tecnológico de Costa Rica

Área Académica Ingeniería mecatrónica
MT-7003 Microprocesadores y Microcontroladores
Est.: Adriana Hernández
Alejandro Sancho

Tarea 1
I Semestre 2021