

Tarea 1

Preguntas Teóricas

1) ¿Diferencie la herramienta Git de GitHub?

Git es un software de control de versiones local, que permite a los desarrolladores tener un control de versiones en sus proyectos, lo que permite acceder como lector o escritor a cualquier versión guardada hasta el momento; GitHub por su parte, lleva ese mismo principio a la web, lo cual permite a los usuarios crear repositorios privados o públicos en caso de requerir colaboración entre varios desarrolladores. [1]

2) ¿Qué es un Branch?

Un Branch es una rama o una derivación de la rama principal (el main) del repositorio, la cual permite trabajar de manera segura (para experimentar, corregir bugs...), ya que toma como base la rama principal, pero no modifica a esta a menos que se haga un merge o unión, comando con el cual, los cambios y modificaciones realizadas en dicha rama o Branch, pasan a ser parte de la rama principal del repositorio. [2]

3) ¿Como se crea un nuevo Branch?

En la parte superior de la aplicación de GitHub, primero se selecciona el Branch del cual se quiere partir para crear el nuevo Branch, luego, se escribe el nombre que se le quiere dar a la nueva Branch, y finalmente, se selecciona “crear Branch” y ya con esto, queda lista la nueva Branch para comenzar a trabajar. [3]

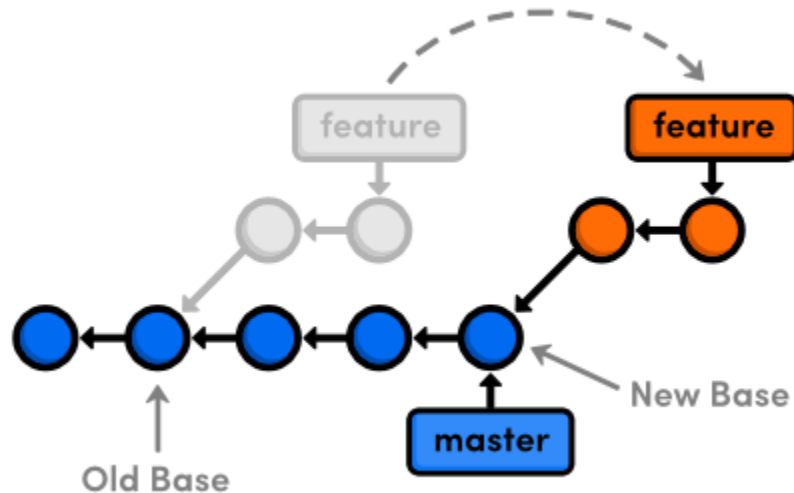
4) ¿Qué es un commit?

Un commit se utiliza cuando se quieren guardar los cambios de la sección del proyecto en la que se está trabajando. Realizar un commit permite al usuario, además de guardar los cambios realizados, poder acceder a dicho punto del proyecto a pesar de que haya algún otro commit realizado en un momento posterior a este. Esto es de gran ayuda, ya que permite recuperar versiones anteriores del proyecto en caso de que una versión futura contenga un error. [4]

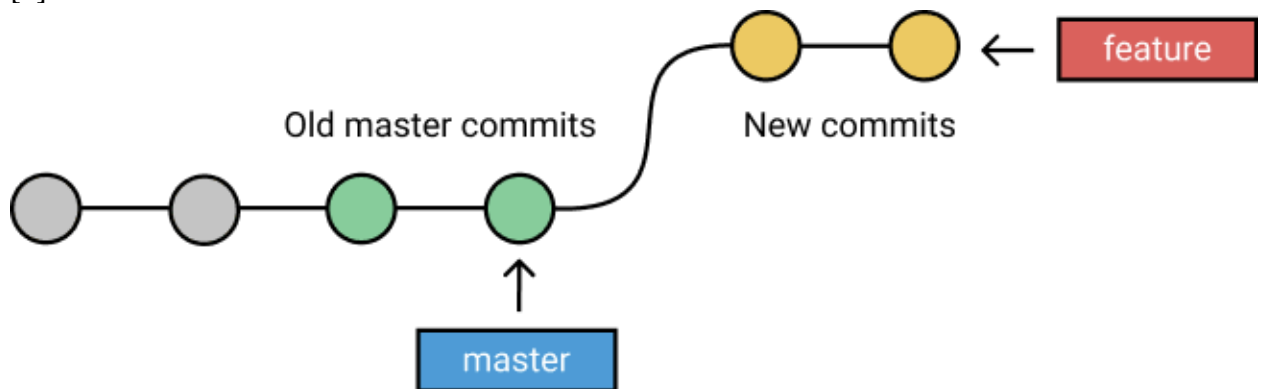
5) ¿Qué es la operación “git stash”?

Esta operación permite que se guarden cambios del código en la rama en que se está trabajando, y con esto se puede trabajar en otra rama, para luego volver y aplicar los cambios más adelante. [5][6]

- 6) Explique de forma gráfica: ¿Que sucede en mi Branch local cuando hago “git rebase master”?



[7]



[8]

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

Un Unittest permite que se haga una prueba a la parte más pequeña del código (la unidad), esto para asegurarse que ese segmento está funcionando de la forma correcta, así como generando los resultados esperados, con lo que se puede probar todo el código principal parte por parte y corroborar su correcto funcionamiento. [9]

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

Un assert es una comprobación que devuelve valores booleanos (Verdadero o Falso), si un assert falla en un método de prueba, entonces la ejecución del método se detiene ahí. El

código restante en ese método de prueba no se ejecuta y se continua con el siguiente método de prueba. [10]

9) ¿Qué es Flake 8?

Flake8 es una biblioteca de Python que envuelve varios métodos y scripts para verificar su base de código contra el estilo de codificación (PEP8), errores de programación (como "biblioteca importada, pero sin usar" y "Nombre indefinido") y para verificar la complejidad ciclomática, que es una métrica usada para medir el número de caminos independientes a través del código fuente. [11]

10) Explique la diferencia entre un “log de error” y un “valor de error de retorno”

Un log de error es un registro de errores críticos que encuentra la aplicación, el sistema operativo o el servidor mientras está en funcionamiento, mientras que el valor de error de retorno lo que hace es devolver un valor para determinar en qué punto de la aplicación es que se está encontrando el error. [12]

Referencias

- [1]"Git vs. GitHub: What's the Difference?", Dev Mountain Part of Strayer University. [Online]. Available: <https://blog.devmountain.com/git-vs-github-whats-the-difference/#:~:text=GitHub%E2%80%A6,what's%20the%20difference%3F,help%20you%20better%20manage%20them.>
- [2]GitHub, "About branches", GitHub Docs, 2021. [Online]. Available: [https://docs.github.com/es/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/about-branches.](https://docs.github.com/es/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/about-branches)
- [3]GitHub, "Creating and deleting branches within your repository", GitHub Docs, 2021. [Online]. Available: [https://docs.github.com/es/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/creating-and-deleting-branches-within-your-repository.](https://docs.github.com/es/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/creating-and-deleting-branches-within-your-repository)
- [4]"Git Guides - git commit", GitHub. [Online]. Available: [https://github.com/git-guides/git-commit.](https://github.com/git-guides/git-commit)
- [5]GitHub, "Stashing changes", GitHub Docs. [Online]. Available: [https://docs.github.com/en/desktop/contributing-and-collaborating-using-github-desktop/making-changes-in-a-branch/stashing-changes.](https://docs.github.com/en/desktop/contributing-and-collaborating-using-github-desktop/making-changes-in-a-branch/stashing-changes)
- [6]Canivore, "How to Git Stash", GitHub, 2011. [Online]. Available: [https://gist.github.com/carnivore/997001.](https://gist.github.com/carnivore/997001)
- [7]R. Hodson, "Reescribiendo la Historia con Git Rebase", Code Envato Tuts+. [Online]. Available: [https://code.tutsplus.com/es/tutorials/rewriting-history-with-git-rebase--cms-23191.](https://code.tutsplus.com/es/tutorials/rewriting-history-with-git-rebase--cms-23191)
- [8]Git Merge vs Git Rebase", Medium, 2021. [Online]. Available: [https://needoneapp.medium.com/git-merge-vs-git-rebase-17392c4d870d.](https://needoneapp.medium.com/git-merge-vs-git-rebase-17392c4d870d)
- [9]"Unit testing: la prueba de calidad para software", IONOS Digitalguide, 2019. [Online]. Available: [https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/el-papel-del-unit-test-en-el-desarrollo-de-software/.](https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/el-papel-del-unit-test-en-el-desarrollo-de-software/)
- [10]"Python Testing with pytest", O'Reilly Online Learning. [Online]. Available: [https://www.oreilly.com/library/view/python-testing-with/9781680502848/f_0020.xhtml.](https://www.oreilly.com/library/view/python-testing-with/9781680502848/f_0020.xhtml)

[11]V. Freitas, "How to Use Flake8", Simple is Better Than Complex, 2016. [Online]. Available:

<https://simpleisbetterthancomplex.com/packages/2016/08/05/flake8.html#:~:text=Flake8%20is%20a%20Python%20library,and%20to%20check%20cyclomatic%20complexity.>

[12]IBM, "IBM Docs", Ibm.com, 2020. [Online]. Available: <https://www.ibm.com/docs/es/icos/20.1.0?topic=errors-check-return-values>.