



Instituto Tecnológico De Costa Rica
Área Académica de Ingeniería Mecatrónica
Microprocesadores y Microcontroladores
MT – 7003

Tarea 1

Estudiantes:

Nicole Jiménez Herrera – 2020035277

Johan Campos Cerdas – 2020

Profesor:

Rodolfo Piedra Camacho

II Semestre, 2023

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

Git es un sistema de control de versiones, este permite a los desarrolladores subir su código, lo que genera un repositorio de versiones del programa, acá los usuarios pueden comparar las diferencias entre versiones, restaurar a versiones antiguas y fusionar los cambios. Entre algunas ventajas que presenta se encuentran; tiene una copia local del código por lo que permite trabajar de manera simultánea, soporte técnico de la comunidad ya que es de código abierto y es fácil de administrar [1].

2) ¿Qué es un branch?

Una rama o **branch** es una versión del código del proyecto sobre el que estás trabajando. Estas ramas ayudan a mantener el orden en el control de versiones y manipular el código de forma segura [2].

Las ramas en Git figuran una línea independiente de desarrollo. Estas sirven como una abstracción en los procesos de edición, preparación y confirmación. Se entienden como una forma de solicitar un nuevo directorio, ambiente o historial de trabajo en el proyecto. Todas esas nuevas confirmaciones son registradas en el historial de la rama principal, creando una bifurcación en el historial del proyecto [3].

Según [2], existen 3 tipos de ramas base en un proyecto en Git:

- Rama Main (Master): Cuando se añade código y se guardan los cambios, lo que se realiza es un commit, que es añadir el nuevo código a una rama. Esto genera nuevas versiones de esta rama o **branch**, hasta llegar a la versión actual de la rama Main.
- Rama development: Esta rama se utiliza cuando se quiere experimentar se utiliza las ramas development que están basadas en alguna rama main, pero sobre las cuales se pueden hacer cambios sin necesidad de afectar directamente al código principal.
- Rama hotfix: En otros casos, si se encuentra un bug o error de código en la rama Main se debe crear una nueva rama (que usualmente se llaman bug fixing o hot fix) para hacer los arreglos necesarios. Cuando los cambios estén listos, se tendrán que fusionar con la rama Main para que los cambios sean

aplicados. Para esto, se usa un comando llamado Merge, que mezcla los cambios de la rama que originaste a la rama Main.

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

Es una función que permite solicitar la revisión y aprobación de los cambios realizados antes de unificarlos o fusionarlos en la rama principal de desarrollo del programa [4].

4) ¿Qué es un commit?

El comando **commit** se usa para confirmar los cambios en el área de preparación y guardarlos en el repositorio local en Git. Sin embargo, antes de guardar algún cambio, se tiene que especificar a Git cuáles cambios se quieren incluir de los muchos que se hicieron. Una excelente manera de hacerlo es agregar un mensaje en el **commit** para identificar los cambios [5].

Como usuario, se pueden recordar los cambios que fueron aplicados en cualquier versión con fechas anteriores, incluso logrando revertir el progreso del proyecto a esa última versión. Además, si se editan varios commits en diferentes partes del proyecto, estos cambios no se sobrescribirán entre sí, aunque los autores no tengan conexión alguna.

5) Describa lo que sucede al ejecutar las siguientes operaciones: “git fetch” “git rebase origin/master”

Git fetch, realiza la búsqueda de los archivos, referencias o el código del repositorio central para posteriormente descárgalo, permitiendo ver el progreso sin necesidad de funcionar con el repositorio local. Por otra parte, Git rebase origin/master, lo que se realiza es el movimiento de una rama particular en su totalidad de una ubicación a otra [6].

6) Explique que es un “merge conflict” o “rebase conflict” en el contexto de tratar de hacer merge a un Pull Request o de completar una operación git rebase.

“Merge conflict” o conflicto de fusión implica que dos confirmaciones realizadas modificaron la misma línea del mismo archivo, lo que puede suceder

cuando se realiza una operación “git rebase”, donde se mueven confirmaciones de un lado a otro [7].

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

Una prueba unitaria es cuando se aísla un fragmento del código de una aplicación; funciones, clases, entre otras para poder verificar el correcto funcionamiento de estas.

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

El uso de `assert` en Python permite realizar comprobaciones. Si la expresión contenida dentro del mismo es `False`, se lanzará una excepción, concretamente `AssertionError` [8].

9) ¿Qué es Flake 8?

Flake8 es una librería Python que se utiliza para comprobar el estilo de código base en Python, verifica la calidad del código y asegura que este se ajusta a las convenciones establecidas para Python [9].

10) Explique la funcionalidad de parametrización de pytest.

La parametrización de una prueba tiene la funcionalidad de ejecutar el prueba con varios conjuntos de entrada, de esta forma comprobar distintos casos sin necesidad de correr múltiples veces la prueba por cada caso.

Se puede usar la parametrización para separar los datos de la prueba del comportamiento de la prueba para que quede claro cuál es la prueba, y también para hacer que los diferentes casos de prueba sean más fáciles de leer y mantener [10].

Referencias

- [1] M. Jacobs, L. Casey y E. Kaim. "¿Qué es Git?" Microsoft Learn: Build skills that open doors in your career. <https://learn.microsoft.com/es-es/devops/develop/git/what-is-git> (accedido el 31 de julio de 2023).
- [2] F. Vega. "¿Qué es branch (rama) y cómo funciona un Merge en Git?". Platzi. <https://platzi.com/clases/1557-git-github/19947-que-es-un-branch-rama-y-como-funciona-un-merge-en-/>
- [3] KeepCoding Team. "¿Qué son las ramas en git y para qué sirven?". <https://keepcoding.io/blog/que-son-las-ramas-en-git/>
- [4] F. Vega. "Utilizando Pull Requests en GitHub". Platzi. <https://platzi.com/clases/1557-git-github/19957-utilizando-pull-requests-en-github/#:~:text=905-,Pull%20request:,a%20una%20rama,%20normalmente%20master>
- [5] KeepCoding Team. "¿Qué es un Commit en Git?". https://keepcoding.io/blog/que-es-un-commit-en-git/#Funcionamiento_de_un_commit
- [6] Equipo editorial de IONOS. "Git pull: descarga y fusión de contenidos". IONOS Digital Guide. <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/git-pull/> (accedido el 31 de julio de 2023).
- [7] "Resolver conflictos de fusión después de una rebase de Git". Documentación de GitHub. <https://docs.github.com/es/get-started/using-git/resolving-merge-conflicts-after-a-git-rebase>
- [8] El libro de Python. "Assert en Python". <https://ellibrodepython.com/assert-python>
- [9] "Flake8 - Herramienta de verificación de estilo en Python". DevOpspain - El foro para todo lo relacionado con DevOps / SRE / SysAdmin. <https://www.devopspain.com/Thread-Tutorial-Flake8-Herramienta-de-verificación-de-estilo-en-Python> (accedido el 31 de julio de 2023).
- [10] "Pytest - Pruebas de parametrización". iSolution. <https://isolution.pro/es/t/pytest/pytest-parameterizing-tests/pytest-pruebas-de-parametrizacion>