



Instituto Tecnológico de Costa Rica

MT7003 - Microprocesadores y microcontroladores

Tarea 1

Integrante:

Kendall Aaron Guerrero González 2019244692

Profesor

Rodolfo Jose Piedra Camacho

II semestre 2025

1- ¿Explique que es git y su relación con github?

Git es un sistema gratuito y de código abierto para el control de versiones, que se instala localmente en tu computadora. Permite llevar un historial completo de cambios en los archivos de un proyecto, facilitando su seguimiento [1].

Git es el mecanismo técnico local para el control de versiones; GitHub, en cambio, es una plataforma online que se apoya en Git para facilitar la colaboración, el hosting de código y la gestión de proyectos [1].

2- ¿Qué es un branch? ¿Qué es un fork?

Un branch (o rama) en Git es como un espacio de trabajo paralelo, que te permite desarrollar nuevas funcionalidades, corregir errores o experimentar sin afectar la versión principal del proyecto [2].

Un fork es una copia completa e independiente de un repositorio existente, alojada en tu propia cuenta de GitHub. Incluye todos los archivos, historial de commits y ramas del proyecto original [2].

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

Un Pull Request es una propuesta que busca fusionar un conjunto de cambios (commits) desde una rama origen (source branch) hacia otra rama destino (target/base branch), habitualmente hacia la rama principal (main o master). Es la forma principal de colaborar en GitHub [3].

4- ¿Qué es un commit?

Un commit en Git es un punto de guardado que registra una instantánea del proyecto en un momento específico, incluyendo los cambios realizados, un mensaje descriptivo, el autor, la fecha y un identificador único (hash), sirve para rastrear el historial, revertir a versiones anteriores y organizar el trabajo en equipo [4].

5- 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.

En Git, un merge conflict o rebase conflict ocurre cuando, al intentar fusionar (merge) o reaplicar (rebase) cambios entre ramas, dos o más modificaciones afectan la misma línea o sección del mismo archivo de forma incompatible. Git no puede decidir cuál versión conservar, por lo que detiene la operación y marca el archivo en conflicto para que el usuario lo resuelva manualmente [5].

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

Una prueba unitaria (unit test) es un tipo de prueba de software que verifica de forma aislada el funcionamiento de la unidad más pequeña de código ejecutable

generalmente una función, método o clase para asegurar que produce el resultado esperado con entradas específicas [6].

Objetivos principales:

- Detectar errores de manera temprana.
- Facilitar el mantenimiento y la refactorización del código.
- Garantizar que cambios futuros no rompan funcionalidades existentes (regression testing).

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

En pytest, un assert se utiliza para verificar que una condición sea verdadera durante una prueba. Si la condición es verdadera, la prueba pasa; si es falsa, la prueba falla y pytest muestra automáticamente, el valor esperado y el obtenido, el código exacto que produjo el fallo [7].

8- ¿Explique que son github-actions y su utilidad para el desarrollo continuo de código?

GitHub Actions es una funcionalidad integrada en GitHub que permite automatizar flujos de trabajo (workflows) directamente desde el repositorio. Está basada en eventos: cada vez que ocurre una acción (por ejemplo, un push, un pull request o una publicación de release), se pueden ejecutar automáticamente tareas definidas mediante archivos YAML en la carpeta `.github/workflows/` [8].

9- ¿Qué es Flake 8?

Flake8 es una herramienta de análisis estático para Python que combina pycodestyle, PyFlakes y McCabe para revisar automáticamente el código, asegurando que cumpla con la guía de estilo PEP 8, detectando errores comunes como variables sin uso o importaciones innecesarias, y evaluando la complejidad ciclomática. Se utiliza para mantener un código limpio, consistente y legible, prevenir errores antes de la ejecución y facilitar la colaboración en equipos, además de integrarse fácilmente en flujos de integración continua (CI/CD) [9].

10-Explique la funcionalidad de parametrización de pytest.

La parametrización en pytest permite ejecutar una misma prueba varias veces con distintos valores de entrada, usando el decorador `@pytest.mark.parametrize`. De esta forma, en lugar de escribir múltiples funciones de prueba repetitivas, se define una sola prueba genérica y pytest la corre automáticamente con cada conjunto de parámetros, esto mejora la cobertura de casos, evita duplicación de código y genera reportes claros en los que cada combinación de valores aparece como un test independiente [10].

Bibliografía

- [1] S. Sharma, "Git vs GitHub – What is Version Control and How Does it Work?," FreeCodeCamp, Aug. 19, 2020. Disponible en: <https://www.freecodecamp.org/news/git-and-github-overview/>.
- [2] "What is a fork in Git?," GeeksforGeeks, Jul. 23, 2025. Disponible en: <https://www.geeksforgeeks.org/git/git-fork/>.
- [3] GitHub Docs, "About pull requests," GitHub, Disponible: <https://docs.github.com/articles/about-pull-requests>.
- [4] GeeksforGeeks, "What is Git Commit?," Disponible: <https://www.geeksforgeeks.org/git/what-is-git-commit/>.
- [5] Atlassian, "Git merge conflicts," Atlassian Git Tutorial. Disponible: <https://www.atlassian.com/git/tutorials/using-branches/merge-conflicts>.
- [6] Amazon Web Services, "What is Unit Testing?,". Disponible: <https://aws.amazon.com/what-is/unit-testing/>.
- [7] pytest Documentation, "How to write and report assertions in tests,". Disponible: <https://docs.pytest.org/en/stable/how-to/assert.html>.
- [8] GitHub Docs, "Quickstart for GitHub Actions," GitHub. Disponible en: <https://docs.github.com/en/actions/get-started/quickstart>.
- [9] flake8 Development Team, "Flake8: Your Tool For Style Guide Enforcement," flake8 Documentation. Disponible en: <https://flake8.pycqa.org/en/latest/>.
- [10] pytest Documentation, "How to parametrize fixtures and test functions," pytest. Disponible en: <https://docs.pytest.org/en/stable/how-to/parametrize.html>.