



Tecnológico de Costa Rica
Escuela de ingeniería mecatrónica

Microprocesadores y microcontroladores

Tarea 1

Estudiantes:

Sherman Harvey Rodríguez

Enmanuel Sandoval Solano

Profesor: Carlos Adrián Salazar García

Cartago, II Semestre 2024

Git como herramienta de desarrollo de código:

Pensado para habilitar la ramificación y el etiquetado como desarrollo esencial de un proyecto, Git es un software de control de versiones que permite rastrear y administrar cambios en el código fuente a lo largo del tiempo, concebido para la eficiencia, confiabilidad y compatibilidad del mantenimiento de versiones emergentes del trabajo colaborativo en proyectos de software.

Diferencia entre Git y GitHub:

Git: Es una herramienta de control de versiones, o Sistema de Control de Versiones Distribuido (DVCS), que se instala localmente en el equipo de cada desarrollador.

GitHub: Es una plataforma web que utiliza Git para alojar distribuciones de archivos, en otras palabras, habilita la gestión de repositorios de código, permitiendo compartir y colaborar en proyectos de código abierto o privados.

Branch:

Un branch es una línea independiente de desarrollo en un repositorio Git o una interpretación del código utilizada por cada trabajador aislada del código principal.

Pull Request en GitHub:

Un pull request es una solicitud para fusionar o hacer un merge de una rama(branch) en otra, usualmente la principal o "main branch", permite revisar cambios antes de integrarlos en la rama principal y así evitar errores.

Commit:

Un commit, o un comando git commit, es una acción en Git que captura un conjunto de cambios o instancia en el historial del repositorio. Cada commit es una captura del estado del código en un momento dado al que se puede recaer de ser necesario y no son alterados a menos que sea solicitado.

Operación "git rebase main":

Desde la perspectiva del contenido, se utilizan los commits desarrollados en el branch individual y se aplican en la versión más actualizada de la rama main o principal, manteniendo un historial de commits lineal.

Merge conflict:

Es una situación en la que el sistema intenta integrar dos ramas diferentes que han realizado cambios en las mismas líneas de un proyecto, por lo que los cambios se superponen superponen o afectan la misma parte del código de formas incompatibles. La manera recomendada para solucionarlo, una vez conscientes del problema es determinar cuales cambios deben mantenerse absolutamente y cuales pueden ser trasladados y reprocesados o añadidos al área de staging con git add para finalmente, completar la fusión total con git commit.

Prueba Unitaria o Unittest:

Una prueba unitaria es un bloque de código que verifica el comportamiento correcto de una unidad individual de código específica, como una función o método, diseñada para verificar que cada parte del código funciona según lo esperado de forma aislada.

En el contexto de pytest, ¿cuál es la utilidad de un assert?

Es una herramienta de pruebas para Python, utilizada para verificar que una condición específica es verdadera o establecer verdades fundamentales en tu código, permitiendo validar suposiciones en el código. Si la condición es falsa, pytest reporta la prueba como fallida y proporciona un mensaje de error detallado, incluyendo los valores que causaron el fallo.

Errores de formato detectables con Flake8

Funciones o Métodos con Demasiados Argumentos (C901): Determina funciones o métodos que tienen demasiados argumentos, promoviendo la simplicidad, comprensión y mantenimiento del código al reducir la complejidad de las funciones.

Nombre de Variables No Coincide con las Convenciones (N8xx): Verifica que los nombres de variables y funciones sigan las convenciones de nomenclatura de PEP 8, asegurando la claridad en los nombres, haciendo el código más comprensible.

Uso de Sentencias print y pdb en Código (T4xx): Detecta el uso de sentencias “print” y “pdb” (debugger) en el código, muchas veces utilizadas en el proceso de desarrollo, sin embargo deben ser eliminadas en el código de producción.