



**Escuela de Ingeniería Mecatrónica**

**Microprocesadores y microcontroladores**

**Tarea 1: GitHub, Pytest y Flake 8**

**Profesor  
Ing. Rodolfo Piedra Camacho**

**Luis Diego Muñoz Marín | 2022035755  
José David Prendas Aguirre | 2022437787**

**27 de Febrero del 2026**

---

## Preguntas teóricas

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

R/ Poder llevar un registro de versiones previas de un código funcional, poder acceder a este y la colaboración simultánea entre distintos programadores.

### 2. ¿Qué es un commit?

R/ Es el equivalente a guardar en un documento cualquiera. Funciona como un respaldo o "zona segura" en el historial de un proyecto.

### 3. ¿Qué es un branch?

R/ Es una ramificación o universo paralelo en el cual se realizan arreglos, experimentos o modificaciones a un código principal sin alterar este.

### 4. En el contexto de github. ¿Qué es un Pull Request?

R/ Permite a los desarrolladores notificar a su equipo de trabajo sobre cambios realizados en una rama de código. Sirve para solicitar la revisión, discusión y aprobación de los cambios antes de fusionarlos en la rama principal.

### 5. ¿Si un usuario quiere “Actualizar su repositorio contra el Branch master” que debe de hacer?

R/ Debe sincronizar su copia local con la versión más reciente disponible en el repositorio remoto, con el comando git pull.

### 6. ¿Bajo qué condiciones una herramienta como Git necesita apoyo de un humano para saber como integrar cambios locales con cambios remotos?

R/ Git necesita intervención humana principalmente cuando ocurre un conflicto de fusión, ya que Git no puede decidir automáticamente cuál cambio es el correcto, por lo que el desarrollador debe revisar el conflicto y resolverlo manualmente. Algunas situaciones donde se requiere revisión del desarrollador son:

Dos personas modifican las mismas líneas de un mismo archivo.

Se elimina un archivo en una rama y se modifica en otra.

Se hacen cambios incompatibles en la estructura del código (por ejemplo, renombrar algo en una rama y modificarlo en otra).

El historial diverge y el merge automático no puede determinar qué versión debe prevalecer.

### 7. ¿Qué es una Prueba Unitaria o Unitest en el contexto de desarrollo de software?

R/ Las pruebas unitarias son un método de desarrollo basado en pruebas (TDD) para evaluar software que presta especial atención a un componente individual o unidad de código, generalmente una función o método, de manera aislada. Por lo tanto, esta prueba ayuda a detectar errores, comprobar el correcto funcionamiento de alguna parte específica del código y ayudar a la documentación del comportamiento del programa.

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

R/ Como tal un “assert” se utiliza para verificar que una condición es True. Si es False, la prueba falla. Por otro lado, Pytest mejora la función básica “assert” al proporcionar una salida detallada cuando la declaración falla, lo que facilita comprender el problema.

**9. ¿Qué es Flake 8?**

R/ Flake8 es una herramienta de análisis estático para Python que ayuda a verificar el cumplimiento del estilo de código (PEP 8), detectar errores de sintaxis e identifica posibles problemas en el código (variables no usadas, imports innecesarios, etc.). En otras palabras, Flake 8 no ejecuta el programa, sino que analiza el código fuente para mejorar su calidad, legibilidad y mantenibilidad.

**10. Comente sobre la utilidad de la aleatorización en pruebas de código.**

R/ La aleatorización en pruebas de código es de gran utilidad a la hora de testear un código, ya que permite evaluar el comportamiento del sistema ante múltiples combinaciones de datos, ayuda a detectar errores que no aparecen con valores fijos, reduce el sesgo del desarrollador al escoger solo casos deseables y mejora la cobertura de escenarios extremos o inesperados.

## Referencias

Fireship. (2018, octubre 26). Git It? How to use Git and Github [Video]. YouTube. <https://www.youtube.com/watch?v=HkdAHXoRtos>.

Arias, D. (2025, agosto 23). Git and Github Explained in 7 Minutes — Git and Github Tutorial (2025) [video]. YouTube. <https://www.youtube.com/watch?v=y-an0v208A0>.