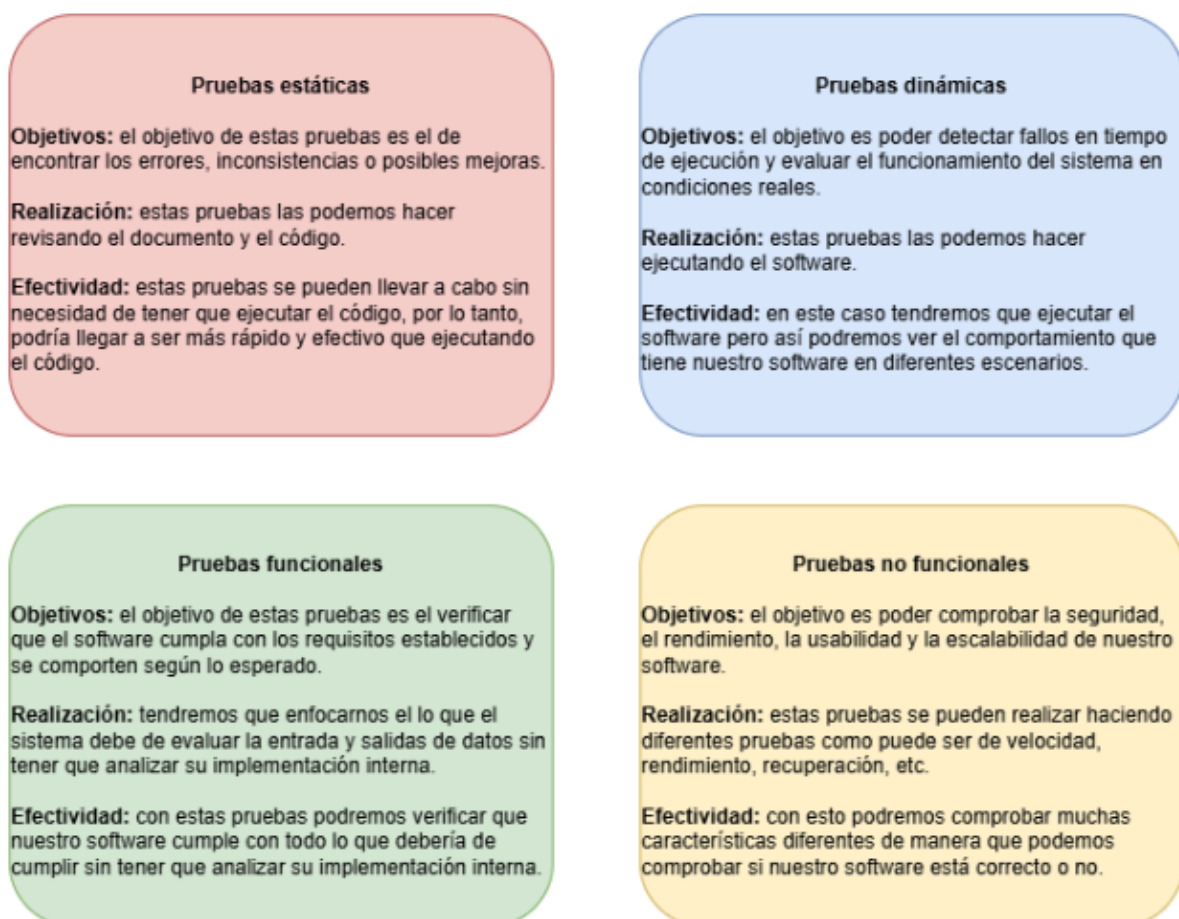


1. Identifica los objetivos principales de las pruebas de software y explícalos brevemente.

El objetivo principal es verificar que el software cumple con los requisitos funcionales y no funcionales establecidos, esto lo podemos comprobar gracias a:

- **Funcionalidad:** esto nos asegura que cada componente cumple con su debido propósito.
- **Rendimiento:** esto nos garantiza un tiempo de respuesta adecuado a la hora de cargar, además que se hacen pruebas con diferentes tipos de carga.
- **Seguridad:** esto hace que nuestro software no tenga vulnerabilidades ni que podamos tener problemas que comprometan la integridad de nuestro sistema.
- **Usabilidad:** esto hace que el software tenga un fácil manejo para el usuario final.
- **Compatibilidad:** esto se encarga de que nuestro software se pueda utilizar en diferentes sistemas, dispositivos y plataformas.

2. Elabora un diagrama comparativo de las estrategias de pruebas, evaluando Objetivos, Realización, Efectividad...



3. Juan y María están implementando la mayor parte de la aplicación. ¿Cómo se prueba los valores devueltos por una función o método? ¿Es posible seguir la ejecución de un programa, y ver si toma los caminos diseñados?

Para poder comprobar que los valores devueltos sean correctos podemos hacer pruebas unitarias, estas se basan en partir el código por métodos o bloques y así analizar y comprobar por partes si el código está correcta o no, con esto nos aseguramos encontrar algún tipo de error y además dónde está ese error ya que al estar partidos por métodos es más fácil de encontrar. Si es posible seguir la ejecución del programa y ver si toma los caminos diseñados ya que gracias a estas pruebas unitarias podemos revisar si la parte del código que hemos hecho está correcta o contiene algún tipo de error.

4. Todos en la empresa están inmersos en el desarrollo de la aplicación de gestión hotelera. Para garantizar la corrección del desarrollo, Ada propone establecer la planificación de las pruebas. ¿Por qué hay que probar el software? ¿Es necesario seguir un orden concreto en la realización de pruebas? ¿Qué pruebas se realizan?

Es necesario probar el software porque este puede llegar a contener errores, entonces si estos errores llegan al usuario final tendríamos un problema ya que al usuario final tendría que llegar una versión sin errores para su correcto funcionamiento. El orden que se debe seguir es:

1. Pruebas unitarias
2. Pruebas de integración
3. Pruebas de regresión
4. Prueba de aceptación

5. Juan y María prueban cada parte de código que están implementando. Algunos métodos requieren una comprobación de su estructura interna, en otros, valdría con probar los resultados que devuelven. Antonio se pregunta en que consiste cada prueba, y como se lleva a cabo en la práctica.

Las pruebas que realizan Juan y María son pruebas unitarias, pruebas dinámicas y pruebas funcionales. Las pruebas unitarias se basan en pruebas que encuentran errores, inconsistencias o posibles mejoras de nuestro software, esto se realiza de manera que partimos nuestro código en bloques y analizamos el software por partes separadas. En cuanto a las pruebas dinámicas, estas se basan en detectar fallos en tiempo de ejecución y evaluar el funcionamiento del sistema en condiciones reales, estas se realizan ejecutando el código. Finalmente las pruebas funcionales se basan en verificar que el software cumpla con los requisitos establecidos y se comporten según lo esperado, estas pruebas las podemos hacer evaluando la entrada y salida de datos sin tener que analizar su implementación interna.