

1. anotar ideas clave relacionadas con:
 - a. ¿Qué es arquitectura de software?

Arquitectura y diseño van de la mano, así que se puede mencionar como el diseño detallado de, en este caso, el sistema.

El objetivo de la arquitectura de software es minimizar los recursos humanos necesarios para construir y mantener el sistema requerido.

- b. ¿Por qué define el éxito o fracaso de un producto?

Éxito: Cuando el sistema maneja una estructura y un diseño previo, así como también tiene la facilidad de ser manipulado y puede llegar a ser escalable.

Fracaso: Cuando los sistemas se ensamblan rápidamente, cuando el gran número de programadores es el único factor que impulsa el resultado y cuando se presta poca o ninguna atención a la limpieza del código o a la estructura del diseño, entonces puedes confiar en aprovechar esta curva hasta su feo final.

- c. Principales atributos de calidad: escalabilidad, seguridad, disponibilidad, mantenibilidad, flexibilidad

Escalabilidad: Debe ser un proyecto limpio y con un buen orden para que sea fácil de manipular.

Seguridad: Un código limpio y ordenado ayuda reducir riesgos.

Disponibilidad: El desorden hace que el sistema sea frágil e inestable; una buena arquitectura asegura que siga funcionando de manera confiable.

Mantenibilidad: La calidad se mide por el esfuerzo de satisfacer al cliente. Si el esfuerzo se mantiene bajo a lo largo del tiempo, el diseño es bueno.

Flexibilidad: El software puede ser capaz de adaptarse a cambios sin convertirse en caos. El desorden, en cambio, rigidiza el sistema y bloquea la innovación.

Arquitectura de Software

La arquitectura de software puede llegar a referirse como la suma de decisiones de diseño de alto y bajo nivel que buscan mantener un sistema sostenible a lo largo del tiempo, de igual forma busca organizar un sistema con el propósito de minimizar el esfuerzo humano necesario para construirlo, extenderlo y mantenerlo durante toda su vida útil.

Un mal diseño puede llegar a terminar en caos, pérdida de productividad y costos que pueden amenazar incluso la supervivencia de una empresa. En cambio, una buena arquitectura es aquella que hace que cada nueva versión del software requiera igual o menor esfuerzo que la anterior.

Ejemplos de atributos de calidad y su importancia:

Escalabilidad: Es la capacidad del sistema para crecer en tamaño, usuarios o funcionalidades sin perder rendimiento ni aumentar los costos de desarrollo. Todo software está destinado a evolucionar, sin esto el sistema simplemente llega a quedar obsoleto y para nada rentable a futuro lo cual igual se traduce a pérdidas.

Mantenibilidad: la mantenibilidad está muy de la mano con base a calidad de un diseño el cual se mide por el esfuerzo requerido para satisfacer al cliente en cada versión. Cuando el código está limpio, ordenado y bien estructurado, las modificaciones se realizan de forma ágil y con menor riesgo de errores, si no, el desorden conlleva a una situación donde incluso los cambios más pequeños requieren un gran esfuerzo y tiempo. Por ende, debe tener desde el inicio un buen orden para que sea fácil de sostener.

Flexibilidad: La flexibilidad es la capacidad del software para adaptarse a cambios en los requerimientos sin que esto conlleve a rehacer gran parte del sistema. Una mala arquitectura provoca rigidez, lo que bloquea la innovación y vuelve al sistema insostenible. La importancia de la flexibilidad es que asegura la vigencia del sistema ante cambios inevitables en el negocio, la tecnología o los usuarios.

Pregunta para debate:

¿Es la arquitectura de software una responsabilidad compartida entre todos los desarrolladores o debe recaer en un rol específico?

