

CÓDIGO LIMPIO

Consiste en aplicar técnicas simples que facilitan la escritura y lectura de un código, volviéndolo así más fácil de entender.

Un sistema nunca está totalmente finalizado porque siempre existe la necesidad de actualizaciones y nuevas funcionalidades, junto con su envejecimiento. La idea es convertir el desarrollo web y el mantenimiento del código más simple.

Un código con muchos ajustes durante mucho tiempo, se vuelve difícil de mantener, por lo que la mejor opción es crear un código nuevo.

El código limpio evita gastos innecesarios y prepara el software para las actualizaciones y mejoras.

REGLAS PRINCIPALES

Los nombres son importantes

La definición de nombre es esencial para el entendimiento de un código, por lo que son importantes los nombres de variables, funciones, parámetros, clases y métodos.

Para definir un nombre hay que tener en cuenta dos aspectos principales:

- Deber ser preciso e y entregar la idea central, debe ir directo al punto.
- Si la función o el parámetro necesita un nombre extenso, se le debe poner un nombre extenso.

Boy scout

Este principio afirma que si sales del área hay que dejarla más limpia que cuando la encontraste. Es decir, hay que dejar el código más limpio de lo que estaba antes de editarlo.

Debes ser el verdadero autor del código

Hay preocuparse por la manera en la que el código está presentado. Para estructurar el código limpio hay que crear funciones simples, claras y pequeñas. Existen dos reglas.

- Las funciones deben ser pequeñas.
- Estas deben ser aún más pequeñas.

DRY (Don't Repeat Yourself)

Se traduce como “no te repitas a ti mismo” y que se aplica a áreas de desarrollo como banco de datos, test, documentaciones y codificaciones.

Cada parte del conocimiento de un sistema debe tener representación única y ser totalmente libre de ambigüedades. Define que no pueden existir dos partes del programa que realicen la misma función.

Comentar solamente lo necesario

Los comentarios pueden hacerse, pero solo si son necesarios. Mientras los códigos son modificados, los comentarios no. Son olvidados y por tanto, no retratan la funcionalidad real de los códigos.

Tratamiento de errores

Hay que tratar las excepciones de forma correcta es un gran paso para el desarrollo. Las cosas pueden salir mal, pero cuando salen mal hay que responsabilizarse para que el código siga realizando lo que necesita.

Test limpios

Realizar test en el área de programación, es una etapa muy importante pues un código se considera limpio, después de ser válido a través de algunas pruebas que también deben ser limpias. Hay que seguir algunas reglas:

- Fast: el test debe ser rápido, permitiendo que sea realizado muchísimas veces y en cualquier momento.
- Independent: debe ser independiente, con el fin de evitar que cause efecto cascada cuando ocurra alguna falla, lo que dificulta el análisis de los problemas.
- Repeatable: debe permitir la repetición del test muchas veces y en diferentes ambientes.
- Self-validation: los test bien escritos retornan con las respuestas true o false para que el error sea subjetivo.
- Timely: los test deben seguir estrictamente el criterio de puntualidad. Además deben ser escritos antes del propio código, ya que evita que sea muy completo para realizar el test.

El código limpio es un concepto que llegó y se quedó. Sus principios solucionan con eficacia uno de los principales problemas que gran parte de los proyectos de sistemas enfrentan, la manutención.