

17 de enero de 2017
Bogotá D.C.



William Sierra	william.sierra@exsis.com.co
Carlos Cardona	carlos.cardona@exsis.com.co
Ervid Molina	ervid.molina@exsis.com.co

The Clean Coder Capítulo 1: Profesionalismo

El capítulo inicia hablando sobre las características más importantes del desarrollador como ser humano integral, es decir aborda la parte profesional y la parte humana como una sola.

Recordemos que ser profesional es llevar una insignia de honor y orgullo, que conlleva unas responsabilidades y obligaciones únicas en cada persona.

Hablar de responsabilidad implica no afanarse por cumplir fechas estipuladas, si antes no se comprueba por completo el programa. Afanarse en programación puede acarrear inconvenientes más graves, debido a que un verdadero profesional nunca antepone su reputación por encima del bienestar de su empleador ni del cliente. Es cierto que todo desarrollador quiere que su software funcione, pero debemos tener en cuenta que no somos los únicos que quieren que eso suceda (Clientes y Empleador, que al final son los que nos contratan y pagan). La responsabilidad llega más profundo, ya que no somos seres humanos perfectos, por tanto no podemos esperar que nuestro software también lo sea; pero tener imperfecciones no nos exenta de nuestras responsabilidades.

Ahora, analicemos los riesgos de cometer un daño dentro de la compañía. Como desarrollador, se puede afectar las funcionalidades del software en construcción como su estructura. En un principio, se busca que el software funcione. Pero tengamos en cuenta que nuestros clientes y empleadores son los primeros interesados en que el software funcione justo como ellos lo necesitan.

Dicho software está condicionado por los errores que presente. Pero aunque el software por su complejidad va a tener errores, depende de nuestro profesionalismo que tales errores se acerquen a cero (madurez profesional). Además, cada desarrollador debe ser responsable de sus imperfecciones sin importar la frecuencia con la que ocurran. Aunque en una empresa pueden existir áreas especializadas en encontrar dichos errores, se deben realizar las acciones necesarias para que encuentren la menor cantidad, y esto se logra realizando múltiples pruebas. No olvidemos el control de calidad, ya que se deben realizar pruebas en el transcurso de la creación para disminuir los errores en lo posible. Esto se consigue con la construcción de código flexible. Dicha metodología se conoce como desarrollo impulsado por

pruebas (TDD). Otra forma de realizar este seguimiento de prueba, este se conoce como automatización de pruebas, con programas como Fitnesse.

El desarrollador profesional sabe que nunca debe sacrificar la estructura del código a expensas de incluir funciones por la presión de la entrega. Además el fundamento de un buen software es su flexibilidad, lo que implica que se le pueda realizar pequeños cambios sin costos exorbitantes; de esto se ha escrito en libros que tratan temas de principios y patrones de diseño de software (PPP 2001). Un truco de un buen desarrollador que busca que su código sea flexible, es que él debe serlo, esta técnica se conoce como "The boy scout rule". Cambiar código no es peligroso, lo peligroso es la rigidez (tratar el código como un escultor a la arcilla modelar continua y diariamente) y esta se refleja cuando se necesitan hacer cambios sencillos: si el desarrollador encuentra que dichos cambios no son tan sencillos de ejecutar, entonces está en la obligación de refinar el diseño para que el siguiente cambio ahora sí sea fácil.

Un profesional tiene una ética de trabajo la cual se basa en no responsabilizar su profesión a su empleador su pasión por aprender y por actualizarse debe ser dirigida por sí mismo, una ventaja adicional sucede si el empleador lo capacite, usted debe sacar el tiempo para direccionar su perfil. El deber de un desarrollador de software es conocer los aspectos relacionados a su campo, por lo cual diariamente debe ejecutar acciones que nos lleven a obtener este conocimiento, sin descuidar sus habilidades que ya tiene, las cuales debe seguir refinados y mejorándolas con ejercicios kata. Existen diferentes formas de aprender. Por medio de **colaboración** se trabaja junto con otro profesional, donde aprenden mutuamente, y complementan su trabajo. Por otro lado está la **mentoría**, que es una de las mejores formas de aprender, en este proceso, el profesional hace las veces de docente, asumiendo la responsabilidad de transmitir un conocimiento claro a su aprendiz, lo que obliga al profesional a tener un fuerte dominio del tema que desea instruir (La mejor forma de aprender es enseñar).

Cuando un profesional se enfrenta a un nuevo desarrollo de software, debe conocer del tema a que se dirige el programa. Con esto no se dice que sea un experto pero sí que conozca conceptos básicos, para que de esta manera pueda entender y trabajar en una mejor solución para las necesidades del cliente, debido a que la programación es un acto de creación; en otras palabras es un acto de orden sobre el caos.

La insignia de profesional también conlleva que seamos un poco arrogantes; pero debemos ser humildes, ya que ese profesional conoce y se enorgullece de su trabajo, confía en sus habilidades y toma riesgos audaces y calculados. Sabiendo que puede fracasar, cuando esto sucede es el primero en reírse, pero nunca menospreciando a otro por cometer un error ya que él puede ser el próximo.

Para concluir, estos son algunos aspectos que definirán y te harán acreedor de la insignia de honor y orgullo profesional.