

26 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 Cap. 2 Diciendo NO.

Este capítulo nos dice los inconvenientes que se pueden presentar por ser desarrolladores pasivos y sin carácter al defender una posición, los desarrolladores profesionales tienen el coraje de decir no cuando sea necesario, pero de una forma donde las partes queden parcialmente satisfechos. De hecho, los buenos gerentes anhelan a alguien que tiene las agallas de decir que no. Es la única manera en que puedes hacer algo.

Si eres profesional, perseguirás y defenderás tus objetivos de la manera más agresiva posible; pero si esto lleva a una confrontación debes buscar el mejor resultado posible: el objetivo que usted y su empleador comparten. El truco es encontrar ese objetivo, encontrando un balance en lo que ambos requieren y pueden aceptar y cumplir, lo cual requiere una negociación.

El momento más importante en el cual un desarrollador debe decir no es cuando las apuestas son más altas. Cuanto más alto es el riesgo, más valioso se vuelve el no. Por ello debes ser un jugador de equipo y que es eso. Ser un jugador de equipo significa jugar su posición lo mejor posible, y ayudar a sus compañeros de equipo cuando se atascan. Un jugador de equipo comunica con frecuencia, está pendiente de sus compañeros de equipo, y ejecuta sus propias responsabilidades tan bien como sea posible. Este concepto de trabajo en equipo llega más profundo: si en un determinado momento tu jefe no está siendo sincero con el gerente sobre tu estimación, debes tener la capacidad de enfrentar a tu jefe. Si no llegas a una conciliación, tienes la posibilidad de disminuir el desastre o catástrofe dirigiendo tus argumentos directamente al gerente.

El verdadero profesional piensa y analiza profundamente el costo de decir sí, ya que pensar que una aplicación es simple; es una idea errónea. El gerente y los desarrolladores deben saber negociar ese sí y diseñar el mejor plan de acción en mutuo acuerdo. Es normal que diferentes roles tengan diferentes objetivos, y que de ahí surjan confrontaciones que son situaciones que se pueden aprovechar para hallar un objetivo en común. El pasar de la confrontación al objetivo en común requiere de **negociación**.

Pensemos en el código imposible. Todo alrededor del desarrollador se ve fuertemente afectado: su empleador, los clientes, su familia y los usuarios, todo por no actuar profesionalmente. Hay que tener presente que el cliente va a pedir nuevos requerimientos. Además los proyectos suelen ser más complejos a como suenan, ya que a nivel técnico traen más cosas de fondo.

Otro factor es el de querer ser héroe, pensando solo en el beneficio propio. Eso no es profesionalismo. Ser profesional implica lograr resultados concretos: lograr el objetivo en el tiempo estimado con el presupuesto destinado. Es correcto decir que no a obligaciones que requieran horribles horas extras y sacrificios, porque de lo contrario lo único que se va a conseguir es un gran problema. De esa manera, se observa que asumir dichas condiciones es un comportamiento poco profesional, y que en esa misma vía se verán las consecuencias.

Concluimos que la tentación de ser un héroe y "resolver el problema" siempre estará presentes en nuestro campo; pero el problema radica cuando decidimos dejar de lado nuestras prácticas profesionales, porque no conseguiremos el resultado deseado. Ser un verdadero héroe significa hacer bien el trabajo, a tiempo y cumpliendo el presupuesto.