



CALIDAD EN EL SOFTWARE LIBRE TÍTULO

César Eduardo Barrón Bautista ^a, Emmanuel López Fernandez ^a, José Antonio González Lopez ^a,
Víctor Manuel Bucio Vargas ^a y Alejandra Martínez Salcido ^a.

^aInstituto Tecnológico de la Laguna., cesar.barron@gmail.com

RESUMEN:

La búsqueda de la Calidad ha existido desde hace mucho tiempo en ámbitos de la ingeniería, pero en la ingeniería de software este concepto se pudiera considerar relativamente nuevo. En este artículo se describe brevemente el estado actual del Aseguramiento de la Calidad de Software, y se plantea la pregunta, ¿Es posible aplicar estándares de calidad al Software Libre? El software de gestión de filosofía Libre ha tomado fuerza en los últimos años y proyectos como OpenERP, un software de gestión de recursos empresariales se han posicionado como opción real para implementar, sobre todo en PYMES. Pero surge la duda si en estos proyectos de código abierto, en el que la comunidad que lo desarrolla cuenta con numerosos miembros de distintas culturas y países, se pudiera aplicar alguna técnica de Aseguramiento de la Calidad de Software. En este artículo se pretende abordar esta interrogante y trataremos de dar una respuesta concreta.

1. INTRODUCCIÓN:

¿Qué es el Software Libre?

Aunque se pudiera dar una amplia explicación de lo que el Software Libre es, trataré de resumirlo describiendo sus principales características.

Un software Libre presenta, principalmente, las siguientes libertades:

1. Quien lo adquiere tiene la libertad de usar el programa, con cualquier propósito.
2. Quien adquiere el software tiene la libertad de estudiar cómo es que el software funciona y de modificarlo adaptándolo a sus necesidades.
3. Le da a quien adquiere este software la libertad de distribuir copias del programa.
4. Se tiene además la libertad de mejorar el programa y hacer públicas esas mejoras hacia el resto de la comunidad que lo desarrolla o que se dedica a mejorarlo, de este modo toda la comunidad se beneficia.

Otra de sus características es que se puede lucrar u obtener alguna remuneración derivada del desarrollo, soporte, implementación o puesta a punto siempre y cuando se entregue el código fuente.



Por otro lado ¿Qué es La calidad de Software?:

Hay miles de definiciones de lo que es la calidad de Software, por decirlo de alguna manera más sencilla la calidad de software es “La capacidad de una aplicación Software de ajustarse a los requerimientos”, además de solucionar cabalmente los requerimientos de los usuarios, un software de calidad debe ser fácilmente extensible, debe cumplir con ciertos lineamientos de seguridad, debe ser fácil de utilizar para los usuarios y debe tener un funcionamiento confiable y estable.

El proceso de aseguramiento de la calidad del software, según Pressman, en su libro Ingeniería de Software involucra:

1. Un enfoque de Gestión de Calidad.
2. Métodos y Herramientas de ingeniería de software efectiva.
3. Revisiones técnicas formales que se aplican durante todo el ciclo de vida del Software (Análisis, Diseño, Codificación, etc.).
4. Modelos y una estrategia de pruebas multiescalada.
5. El control de la documentación del software y de los cambios realizados.
6. Un procedimiento que asegure un ajuste a los estándares de desarrollo del software.
7. Mecanismos de medición y de generación de informes.

Por último, como bien es sabido, “Lo que no se puede medir, no se puede controlar” algo imprescindible para controlar y evaluar la calidad del software son las métricas, las más reconocidas están incluidas en los estándares de calidad de Software como la norma ISO-9126.

2. TEORÍA:

La manera clásica de desarrollar proyectos de Software Libre, es que un grupo de desarrolladores, al inicio de un número moderado de miembros, genera un producto de software que considera que será de utilidad para la comunidad. Posteriormente pueden subir su código fuente a un repositorio público, e invitar a la comunidad de programadores con determinado perfil a revisar y extender dicho código.

Los proyectos de Software Libre más exitosos llegan a tener una comunidad activa de cientos o miles de personas alrededor de todo el mundo colaborando en las revisiones y las mejoras al software. Esta característica en particular hace que sea difícil el aplicar procesos y métricas que permitan asegurar la calidad del software resultante de la mayoría de los proyectos de Software Libre que se consideren como interesantes.

Tal vez alguno de los que quisieran implementar dicho software aplicaría algo de ingeniería inversa a fin de asegurar hasta cierto punto la calidad del software libre, pero difícilmente lo compartiría con la comunidad desarrolladora del proyecto.



4. CONCLUSIONES:

En el ámbito del software existen dos corrientes que han madurado independientes una de la otra, paradójicamente éstas dos corrientes aunque son valiosas por si solas, una (el software libre) necesita de la otra (la calidad de software) para cumplir totalmente su propósito que es la ser útil a la comunidad a la que está destinado.

La siguiente etapa que debería de seguir el software libre es la de crear una estructura y una metodología que facilite a su comunidad aplicar métricas de calidad de software desde sus etapas más tempranas que son el análisis y diseño atreves de todo el ciclo de vida del software hasta llegar a la implementación y mantenimiento, solo de ésta manera el software libre daría el siguiente paso, para convertirse totalmente en un competidor serio de los software de paga.

BIBLIOGRAFÍA:

1. Pressman R., (2005). Ingeniería de software Un enfoque práctico. 6ª Edición. Mc. Graw Hill.
2. Craig, Larman, (2004). UML y Patrones. 2ª Edicion. Prentice Hall.
3. Martin Fowler, Kendall Scott (1999). UML Gota a Gota. Pearson.
4. Erik Freeman, Elisabeth Freeman (2004). Head First Design Patterns. O'Reilly.