

Revisión Sistemática Sobre la Certificación del Producto Software

Moisés Rodríguez^{1,*}, Mario Piattini²

¹Alarcos Quality Center S.L. University of Castilla–La Mancha Ciudad Real, Spain

²Instituto de Tecnologías y Sistemas de Información University of Castilla–La Mancha Ciudad Real, Spain

Abstract La calidad del software está adquiriendo durante los últimos años una gran importancia por la necesidad creciente de asegurar el correcto funcionamiento de los sistemas informáticos que cada vez están presentes en más ámbitos de nuestra vida. Este interés por la calidad ha estado centrado principalmente en los procesos que se siguen para el desarrollo del software, surgiendo certificaciones bajo normas como CMMI o ISO/IEC 15504. Sin embargo, cada día es mayor la preocupación por la calidad del propio producto software y no solo por la de los procesos que se utilizan para su desarrollo. De esta manera, han surgido normas como la familia ISO/IEC 25000, que definen un modelo de calidad y un proceso de evaluación para el producto. A pesar de ello, son menos extendidas las certificaciones que tomen como objeto el propio producto software y permitan asegurar el cumplimiento por parte de este de un conjunto de requisitos. En este sentido, la presente revisión sistemática tiene por objetivo identificar las iniciativas existentes sobre la certificación de los productos software, las características que presentan y el estado de madurez en su aplicación.

Keywords Certificación del Producto Software, Calidad del Producto Software, ISO/IEC 25000, Revisión Sistemática

1. Introducción

La actividad económica relacionada con la Industria del Software está tomando cada vez mayor importancia a nivel mundial. La cantidad de empresas dedicadas al desarrollo software está experimentando un fuerte crecimiento, en línea con el incremento de la demanda de productos del sector. Y en este tipo de empresas la calidad del software tiene un papel fundamental, por su repercusión en los costes finales, como elemento diferenciador de la competencia y de imagen frente a sus clientes[1]. Más aún siendo conscientes de las pérdidas que los problemas de la calidad en el software pueden llegar a causar. En este sentido hay que destacar el último informe “CHAOS” del *Standish Group* en 2009 (Tabla 1), que señala que sólo el 32% de los proyectos informáticos finalizan en el tiempo estimado, con los recursos planificados y con una calidad aceptable, mientras que casi una cuarta parte no llegan a finalizar nunca. Ejemplos como el reciente caos aéreo producido en EEUU, o desastres como los del Ariane 5, las sondas de la misión a Marte, aviones comerciales, Ambulancias de Londres, Aeropuerto de Denver, etc.[2] hacen reflexionar sobre la importancia que la calidad de los sistemas de información y por tanto su correcta evaluación, tienen para el

funcionamiento de la sociedad actual[3].

Se puede considerar que hoy en día las actividades relacionadas con la calidad de software están cobrando cada vez más importancia debido principalmente a dos factores:

1. El crecimiento experimentado por la externalización (*outsourcing*) de software, cuyo mercado a nivel internacional, según estudios de Gartner Group, alcanzó los 1,3 billones de dólares en 2008. Hay que señalar además que España se está convirtiendo en uno de los centros de *nearshoring*[4] y *offshoring*[5], preferidos a nivel mundial con una gran cantidad de factorías de software instaladas[6]. Esto hace que, por un lado, las organizaciones que trabajan en “modo factoría” deban invertir recursos para “asegurar” la calidad del software que fabrican durante todo el ciclo de vida; mientras que por otro lado, los clientes deban “evaluar/controlar” la calidad del software que reciben de las factorías[7].

2. La importancia creciente de las certificaciones basadas en modelos de madurez como CMMI, ISO/IEC 15504, etc. que destacan las actividades de aseguramiento de calidad como clave para la madurez de una organización que desarrolle o mantenga software. En España, la importancia que está adquiriendo para las organizaciones los modelos de este tipo[8] se refleja en el elevado número de certificaciones respecto a otros países europeos. Tal es así, que hoy en día España ocupa el primer puesto europeo en número de empresas certificadas (220) bajo el modelo CMMI[9] y está avanzando considerablemente en la adopción de la Norma ISO 15504[10].

* Corresponding author:

moises.rodriguez@alarcosqualitycenter.com (Moisés Rodríguez)

Published online at <http://journal.sapub.org/computer>

Copyright © 2012 Scientific & Academic Publishing. All Rights Reserved

Tabla 1. Resultados del informe “CHAOS” del Standish Group hasta 2009

Resultado/ Año	1994	1996	1998	2000	2002	2004	2006	2009
Éxito	16%	27%	26%	28%	34%	29%	35%	32%
Deficiente	53%	33%	46%	49%	51%	53%	46%	44%
Fracaso	31%	40%	28%	23%	15%	18%	19%	24%

Aunque la evaluación de la calidad del software es un campo de gran actividad investigadora desde hace bastantes años, la mayor parte del esfuerzo realizado se ha centrado en la calidad de los procesos, habiéndose desarrollado una plétora de modelos y estándares de referencia, evaluación y mejora de procesos software: ISO 90003, ISO 12207, ISO 15504, CMM, CMMI, IDEAL, SCAMPI, Competisoft, etc. Con la premisa de que utilizando un proceso de calidad se obtiene un producto de calidad, las organizaciones y empresas se han preocupado de exigir a sus proveedores (y los proveedores de obtener) una certificación sobre la capacidad y madurez de sus procesos, principalmente con normas como ISO 9000 (aplicando la guía específica para software ISO 90003), CMMI-DEV o ISO 15504 (SPICE).

Sin embargo, hay poca evidencia de que cumplir un modelo de procesos asegure la calidad del producto software desarrollado, y aunque la estandarización de los procesos garantiza la uniformidad en la salida de los mismos, lo que puede incluso hacer es institucionalizar la creación de malos productos[11]. En este sentido, se está completamente de acuerdo con que las evaluaciones deberían basarse en evidencias directas del producto, y no solo en evidencias circunstanciales del proceso[12]. Por ello, es cada día mayor el número de organizaciones y empresas que se interesan, no solo por la calidad de los procesos que se siguen en el desarrollo de software, sino también por la calidad de los productos que desarrollan y/o adquieren, ya que una vez que el producto ha sido implantado en sus instalaciones se encuentran con graves problemas de calidad y complicaciones a la hora de corregirlo, adaptarlo o ampliarlo.

Dentro del ámbito de la calidad del producto software cabe destacar la nueva familia de normas ISO/IEC 25000[13], conocida como SQuARE (*Software Product Quality Requirements and Evaluation*). Esta familia de normas, que todavía no ha terminado de publicarse en su totalidad, tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del producto software, sustituyendo a las anteriores normas ISO 9126[14] e ISO 14598[15] y convirtiéndose así en la piedra angular de esta área de la Ingeniería del Software. La norma ISO/IEC 25000 define entre otros un modelo de calidad con las características del producto software que se pueden evaluar[16], así como un proceso de evaluación con las actividades y tareas que se deben llevar a cabo[17]. Sin embargo, esta familia de normas no describe un modelo de certificación para el producto software que permita identificar el nivel de calidad que presenta, compararlo con

otros productos semejantes y obtener un certificado de cumplimiento de requisitos de calidad.

Como se ha comentado, la certificación del software es una actividad extendida desde hace años a nivel de procesos con normas como ISO/IEC 15504 o CMMI. Sin embargo, la certificación es menos conocida a nivel del producto software, puesto que el interés en este ámbito se encuentra en estado incipiente (comparándolo con el de los procesos).

Por otro lado, para poder llevar a cabo una certificación del producto software serían necesarios un conjunto de elementos, entre los que hay que destacar:

- Un modelo de calidad con las características y métricas que se van a evaluar.
- Un proceso de evaluación, con las actividades que se realizan para evaluar el producto.
- Una entidad evaluadora, que se encuentre acreditada para realizar tales actividades.
- Un proceso de certificación, con las actividades de auditoría para certificar los productos software.
- Una entidad certificadora, acreditada para realizar la auditoría del producto.

Teniendo en cuenta todo lo anterior, en este trabajo se presenta una revisión sistemática sobre el estado del arte referente a la certificación del producto software. El objetivo es comprobar cuales de los elementos del conjunto anterior han sido ya investigados, en qué estado se encuentra su desarrollo y detectar que aspectos quedan por completar.

El resto de este trabajo se encuentra estructurado como sigue: el apartado 2 presenta los principales aspectos de la revisión sistemática que se ha realizado. El apartado 3 contiene el análisis de los resultados obtenidos. Por último, en el apartado 4 se exponen las conclusiones y las líneas de trabajo futuro.

2. Principales Aspectos de la Revisión Sistemática

Las revisiones sistemáticas de la literatura permiten identificar, evaluar, interpretar y sintetizar todas las investigaciones existentes y relevantes en un tema de interés particular. Este tipo de revisiones se ejecutan de manera rigurosa e imparcial para que tengan un alto valor científico. La principal motivación de realizar una revisión sistemática es aumentar la posibilidad de detectar más resultados reales del tema de interés que los que se pueden detectar con revisiones de menor dimensión.

Para la ejecución de la presente revisión sistemática de la

literatura se ha seguido la guía propuesta por Kitchenham en[18], junto con la plantilla descrita por Biolchini en[19]. Según estas bases, existen dos actividades principales a realizar previas al análisis de resultados que son: la planificación y la ejecución de la revisión.

2.1. Planificación de la Revisión

Para comenzar con la revisión, se plantearon las siguientes preguntas de investigación a resolver con el estudio:

- ¿Qué iniciativas se han llevado a cabo para la certificación del producto software?

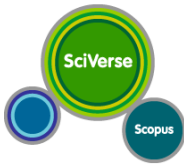

- ¿Cuáles de estas iniciativas cuentan con casos prácticos?

Tras el análisis de estas preguntas se extrajeron los términos clave y sus sinónimos, presentes en la Tabla 2. Hay que destacar que los términos de búsqueda han sido traducidos al inglés, debido a que las fuentes en las que se realizará la búsqueda se encuentran en este idioma. Por otro lado, se puede observar que en la Tabla II no se han incluido sinónimos para “*certification*”. Esto se debe a que el interés de esta revisión es buscar aquellas iniciativas que traten solo sobre certificación (entendida como se describió en la introducción del presente documento), y no incluir aquellas propuestas que hablen solo de evaluaciones, revisiones, auditorías, inspecciones, etc.

Tabla 2. Terminos de Búsqueda

Id	Término Clave	Sinónimos
1	certification	-
2	software	-
3	product	system component code

Tabla 3. Fuentes de Búsqueda

Id	Fuente de búsqueda
1	SciVerse Scopus  http://www.scopus.com/home.url
2	Springer  http://www.springerlink.com/

En base a estos términos se construyó la cadena de

búsqueda, formada por la siguiente expresión:

(“software product” OR “software system” OR “software component” OR “software code”) AND certification.

El siguiente paso en la planificación fue seleccionar las fuentes de búsqueda. El criterio para seleccionarlas se basó en considerar aquellas que estuvieran relacionadas con el tema de la revisión, pudieran ser accedidas mediante un navegador Web. De un listado inicial propuesto, formado por 8 fuentes, se realizó una revisión con expertos en la materia y se acordó seleccionar como un conjunto completo las fuentes de la Tabla 3, para cada una de las cuales se adaptó la cadena expuesta anteriormente de acuerdo al motor de búsqueda que presentaban.

La manera de llevar a cabo la selección de estudios queda definida por los criterios de inclusión y de exclusión, junto con el procedimiento de selección. Los criterios de inclusión que han hecho que un artículo sea considerado o no para la revisión sistemática han sido los siguientes:

- Responde a una de las preguntas de investigación.
- Expone características del modelo de certificación del producto propuesto.

Tras realizar una primera iteración, se identificó que los estudios relevantes trataban de manera diferente la certificación del producto software. En este sentido se clasificaron los estudios relevantes en cuatro niveles:

- Estudios que describen una propuesta para certificar el producto software y exponen un caso de aplicación.
- Estudios que describen un modelo o proceso para llevar a cabo la certificación del producto software.
- Estudios que hablan de la calidad del producto software pero se centran más en aspectos relacionados con la evaluación, métricas, o herramientas.
- Estudios que hablan de certificar el software, pero desde el punto de vista de los procesos y solo comentan la certificación del producto.

Para determinar que artículos relevantes debían ser considerados estudios primarios para la revisión, se definió como criterio de exclusión el eliminar aquellos que formaran parte de los niveles 3 y 4 anteriormente descritos.

Por último en esta fase, se definió un procedimiento de selección y análisis de trabajos, iterativo e incremental descrito por las siguientes actividades:

1. Para cada motor de búsqueda se introducía la cadena de búsqueda adaptada.

2. Para cada uno de los resultados se analizaba el título, abstract y palabras clave del artículo y se miraba si cumplía el criterio de inclusión.

3. En caso de cumplir el criterio de inclusión se analizaba en profundidad el artículo. En caso contrario se descartaba y se pasaba al siguiente.

4. Al analizarlo en profundidad se comprobaba si cumplía el criterio de exclusión.

5. En caso de cumplir el criterio de exclusión se añadía al conjunto de estudios primarios de la revisión sistemática, para los cuales se realizaría la extracción de información. En caso contrario se descartaba y se pasaba al siguiente.

Tabla 4. Resumen de Resultados por Fuente de Búsqueda

Fuente	Estudios					
	Fecha	Totales	No repetidos	Relevantes	Primarios	%
Scopus	Nov 2011	254	246	76	9	90
Springer	Nov 2011	14	3	1	1	10
Total		268	249	77	10	100

Tabla 5. Listado de Estudios Primarios Seleccionados

Id	Título	Año	Referencia
1	A software product certification model	2010	[20]
2	Embedded software component quality and certification	2009	[21]
3	Performance certification of software components	2011	[22]
4	Requirements certification for offshoring using LSPCM	2010	[23]
5	SCfM_PROD: A software product certification model	2008	[24]
6	Continuously ensuring quality through software product certification: A case study	2010	[25]
7	A Software Certification Consortium and its Top 9 Hurdles	2009	[26]
8	Software component certification	2001	[27]
9	Standardized code quality benchmarking for improving software maintainability	2011	[28]
10	Towards a software component certification framework	2007	[29]

2.2. Ejecución de la Revisión

A partir de la planificación realizada, para cada una de las fuentes de búsqueda seleccionadas y aplicando el protocolo definido, se llevó a cabo la ejecución de la revisión sistemática, actividad dividida principalmente en la selección de los estudios primarios y en su posterior extracción de información.

Los resultados obtenidos en la selección de estudios se reflejan en la Tabla 4. Como se puede observar, de los 77 estudios relevantes con los que se contaba inicialmente, solo 10 han cumplido con los criterios para formar parte de la presente revisión sistemática. Esto nos da una primera idea de la situación y es que, si bien existe bastante relacionado con la certificación del producto, son pocos los estudios que presentan propuestas de cómo llevarla a cabo. El detalle de estos estudios primarios seleccionados se presenta en la Tabla 5.

Una vez seleccionados los estudios, se realizó la extracción de la información para cada uno de los mismos, utilizando para ello un formulario de extracción con los siguientes apartados:

- Título y lugar de publicación (Revista/Conferencia).
- Fecha y Autores.
- Resultados objetivos del estudio:
 - País donde se ha realizado el estudio.
 - Modelo de certificación propuesto.
 - Tipos de producto software.
 - Características del producto certificadas.
 - Normas de referencia.
 - Entidades involucradas.
 - Resultados obtenidos.
- Otros aspectos relevantes del estudio.

Siguiendo estos apartados, se construyó una tabla con la

información extraída para cada uno de los estudios primarios. Debido a las restricciones de espacio, en este artículo no se presentan las tablas de extracción de datos elaboradas para todos y cada uno de los artículos primarios seleccionados, sin embargo sirva de ejemplo la Tabla 4 donde se presentan los resultados obtenidos para el primero de los estudios reflejado previamente en la Tabla 5.

3. Análisis de los Resultados

En este apartado se presenta un análisis de los resultados obtenidos, realizado a partir de la información conseguida mediante los formularios de extracción del apartado anterior. En primer lugar se analiza cómo se distribuyen los estudios relacionados con la certificación del producto software a lo largo de los últimos años. Aunque se sabe que mucho antes ya existían muestras de preocupación por la calidad de los productos software como se tiene constancia en la “*NATO Software Engineering Conference*” de 1968, según se puede observar en la Fig.1, el primer estudio recuperado que trata los temas de certificación data de 1983. A partir de ese año y durante la década de los 90 la situación se mantuvo estable, pero es a partir de la década del 2000, sobre todo en la segunda mitad, cuando han comenzado a surgir una importante cantidad de propuestas relacionadas con este tema. También se puede observar que los estudios considerados como primarios son todos de la década del 2000, y además la mayoría de ellos de los últimos tres años. Esto nos da una muestra de la relativa reciente importancia que está adquiriendo la certificación del producto software y que, a pesar de que hace varias décadas que existen trabajos relacionados con el tema, todavía hoy sigue siendo una tarea pendiente en la Ingeniería del Software.

Tabla 6. Ejemplo de Tabla de Extracción de Información

Formulario de extracción de datos	
Título del estudio	<i>A software product certification model</i> [20]
Revista/Conferencia	Software Quality Journal
Fecha	2009
Autores	Heck, P., Klabbers, M. y van Eekelen, M
País	Suiza, Holanda
Modelo de certificación propuesto	LaQuSo <i>Software Product Certification Model</i> (LSPCM). Compuesto por: <ul style="list-style-type: none"> • Elementos de entrada a certificar (clasificados por <i>Product Areas</i>) • Propiedades de Conformidad a certificar. • Criterios de certificación (completitud, uniformidad y cumplimiento) Niveles de certificación (5 niveles similares a los de madurez de CMMI)
Tipos de producto software	Cualquier producto software (dejando fuera aspectos como el hardware, las redes y los procesos de desarrollo)
Elementos a certificar	Clasificados en seis Áreas de Producto: <ul style="list-style-type: none"> • Descripción del contexto • Requisitos de usuario • Diseño de alto nivel • Diseño detallado • Implementación • Pruebas
Características del producto certificadas	Las características a certificar las llaman Propiedades de Conformidad y son las siguientes: <ul style="list-style-type: none"> • Consistencia entre los componentes del producto software. • Funcionalidad del producto software. • Comportamiento respecto a las restricciones que debe cumplir. • Calidad para alcanzar los requisitos no funcionales (rendimiento, seguridad, usabilidad). • Conformidad con estándares y legislación vigente.
Normas de referencia	<ul style="list-style-type: none"> • ISO 9001:2000, ISO 9126:2001 • ESA software engineering standards • IEEE Std. 610.12-1990. • IEEE Std. 829-1998. • IEEE Std. 830-1998. • IEEE Std. 1016-1998. • IEEE Std. 1063-2001. • IEEE Std. 1233-1998. • IEEE/EIA 12207.0-1996 • CMMI
Entidades involucradas	<ul style="list-style-type: none"> • Software Quality Systems • LaQuSo: <i>Laboratory for Software Quality</i>
Resultados obtenidos	El estudio muestra la aplicación de los conceptos para dos tipos de certificación: <ul style="list-style-type: none"> • Evaluación de la consistencia de requisitos de usuario en un proyecto con 15 casos de uso. • Evaluación del comportamiento de la implementación de un sistema software para balanceo de carga. • En ambos casos se elaboró una checklist de comprobaciones y en función de las respuestas se le asignó un nivel para emitir el certificado.

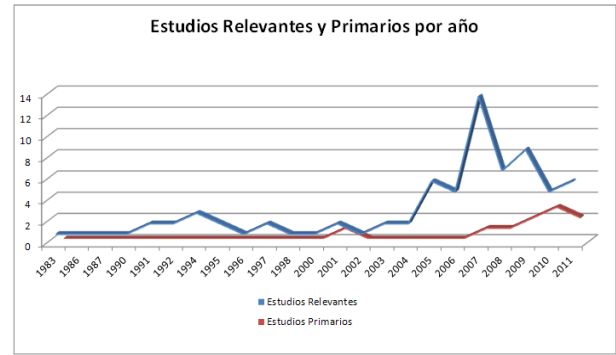


Figura 1. Estudios relevantes y primarios por año de publicación

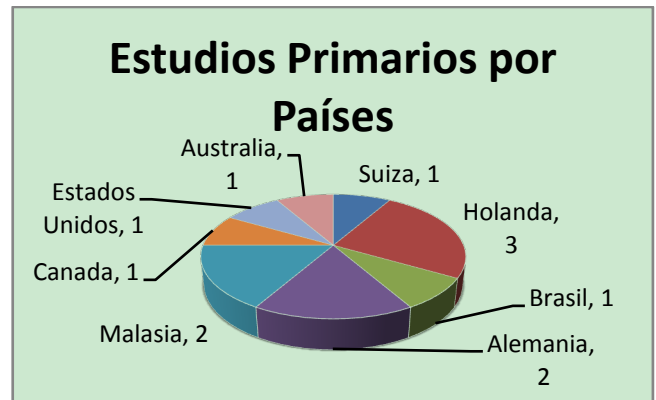


Figura 2. Estudios primarios por país

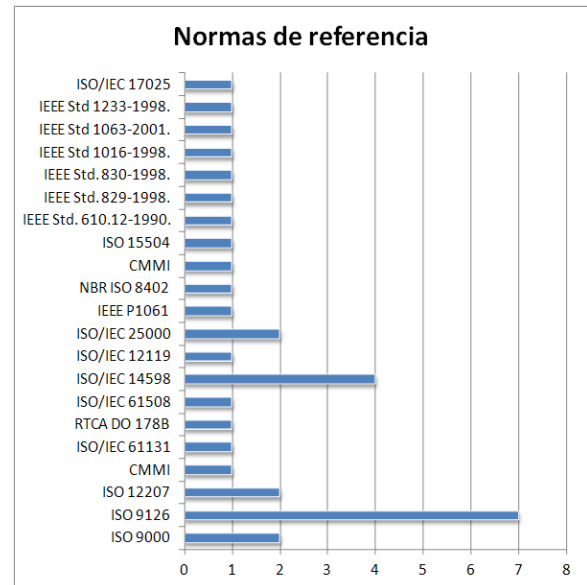


Figura 3. Normas de referencia utilizadas en los estudios de certificación

En segundo lugar, en la Fig.2 se muestra la distribución de estudios primarios por países donde han sido elaborados. Es de destacar que a nivel europeo hay un gran interés por este tema con países como Holanda y Alemania a la cabeza, donde se han realizado las principales aportaciones a la certificación del producto software, con laboratorios especializados en la materia. En este sentido llama también la atención Malasia, que en los últimos tres años ha elaborado un modelo de evaluación y certificación que está siendo utilizado a nivel nacional para ser validado. En la

Fig.3 se han recogido las normas de referencia que según los estudios primarios han sido consideradas a la hora de elaborar los modelos de certificación propuestos. Como se puede observar, hay más de 20 normas y modelos de referencia distintos, sin embargo destacan sobre las demás el uso de la ISO/IEC 9126 como modelo de calidad del producto software y la ISO/IEC 14598 como modelo de procesos para la evaluación del producto. Por otro lado, aunque hay dos estudios primarios que nombran la nueva familia de normas ISO/IEC 25000, hay que aclarar que realmente no la han adoptado completamente y solo la han considerado en su revisión antes de hacer su propuesta.

Por otro lado, en la Fig.4 se analizan las características de calidad que se proponen en los estudios primarios para ser certificadas. Hay que destacar que en 5 de los estudios primarios se identifica la funcionalidad (del producto o componente software) como la característica a ser certificada, para comprobar que los productos software cumplen con los requisitos funcionales que se identificaron al principio del ciclo de vida. Por otro lado, la mantenibilidad, fiabilidad y usabilidad del producto software son también características propuestas para su certificación en 4 de los estudios primarios, lo que refleja también la importancia que se les asigna a estas propiedades del producto software. Finalmente respecto a la Fig.4, llama la atención que la seguridad, una característica tan importante hoy en día en los productos software, solo sea considerada en uno de los estudios para ser certificada en los productos software. Esto hace pensar en la necesidad de una búsqueda especializada para este ámbito de la Ingeniería del Software.

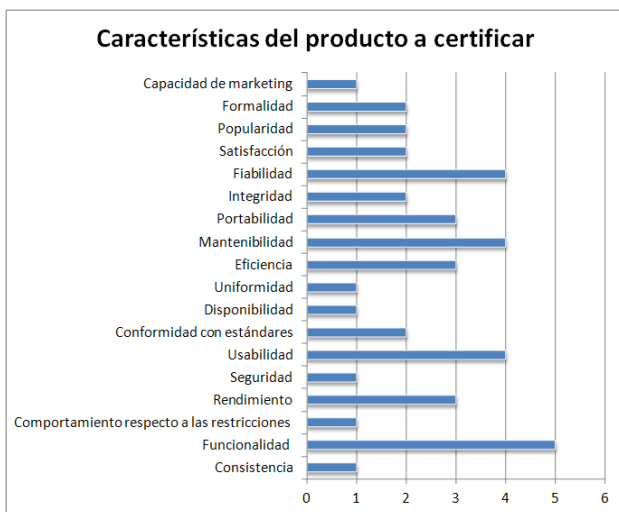


Figura 4. Características del producto que se certifican

En la Fig.5 se analiza el número de productos software en los que han sido aplicados los modelos de certificación propuestos en los estudios primarios analizados. Como se puede observar, en un 60% de los estudios no se habla de una aplicación práctica del modelo de certificación ni de resultados obtenidos en empresas que hayan certificado sus productos. Solo un 30% de los estudios han aplicado su modelo de certificación a un conjunto de entre 1 y 10 productos. Y tal solo un estudio habla de haber aplicado la

certificación a más de 10 productos software. Estos datos revelan de nuevo la novedad de estos modelos de certificación y la necesidad que existe de que sean aplicados a un conjunto mucho mayor de productos, para poder de esta manera establecer las bases de la certificación, identificar las características comunes y las particulares de los distintos tipos de productos software y definir un modelo estándar de certificación que sea aplicable a todos los productos software independientemente de la metodología de desarrollo utilizada (ágil, cascada, etc.), el ámbito de negocio del producto software desarrollado (gestión, sanidad, ocio, etc.) o las tecnologías empleadas (Java, .NET, C, etc.).



Figura 5. Número de productos software certificados por cada estudio

En la Fig.6 se han querido analizar los elementos de entrada al proceso de certificación que se han identificado en los distintos estudios analizados. Hay que destacar que el 40% de los estudios identifica como posible entrada cualquier tipo de producto software, independientemente de la tecnología o el ámbito de negocio del mismo. De igual manera un 30% de los estudios identifican como entrada un componente software que después pueda ser integrado en un sistema mayor. En estos dos casos (equivalentes al 70%), los estudios parecen indicar que lo que realmente analizan y certifican es el código fuente de dichos productos, sin embargo tampoco lo especifican explícitamente ni identifican si son exclusivos para ciertos lenguajes de programación, lo que técnicamente es por experiencia propia un aspecto muy importante, puesto que no es igual analizar un código Java orientado a objetos, que un código en C y la lógica de punteros. Por otro lado, en dos de los estudios primarios se identifica como elemento de entrada también a los requisitos, de manera que son estos los que se analizan y se certifican para mostrar que el análisis de requisitos representa exactamente lo que el cliente solicitó.

Por último, en la Fig.7 se presentan los elementos que se han identificado en los estudios primarios, del listado de elementos que se consideró al final de la introducción del presente artículo. Como se puede observar, en 8 de los 10 estudios se propone un proceso de certificación y en los otros dos estudios restantes del conjunto se ha propuesto, por un lado un "Consortio de Certificación" formado por empresas y universidades que tienen pendiente sacar su modelo de certificación, y por otro lado un estudio del estado de la certificación de los componentes de software que ha servido

para conocer la situación en este campo.

También en la Fig.7 se aprecia que 6 de los 10 estudios identifican el modelo de calidad con las características del producto software a evaluar para poder ser certificado, así como el proceso a seguir para poder llevar a cabo su evaluación. Sin embargo, llama la atención que mientras en 7 de los estudios se identifica el rol de la entidad evaluadora, como aquella encargada de ejecutar la evaluación, solo en 3 de los estudios aparece el rol de la entidad certificadora, como aquella entidad independiente que puede emitir el certificado de cumplimiento. Esto se debe a que en la mayoría de los estudios la entidad que realiza la evaluación es la misma que emite el certificado y hablan de entidad evaluadora y certificadora refiriéndose a la misma entidad. Tan solo en uno de los estudios (el noveno de la Tabla V), se diferencia entre estos dos tipos de entidades, lo que desde el punto de vista de los autores de la presente revisión sistemática se considera como una necesidad, ya que en los otros casos no se asegura la independencia en el proceso “evaluación-certificación”. Por esta misma razón, hoy en día existen normas para acreditar a entidades (laboratorios) que realizan procesos de ensayo y evaluación de productos (ISO/IEC 17025). Y por otro lado, normas que acreditan a entidades para poder certificar productos a partir de los informes emitidos por los anteriores laboratorios (UNE-EN 45011).

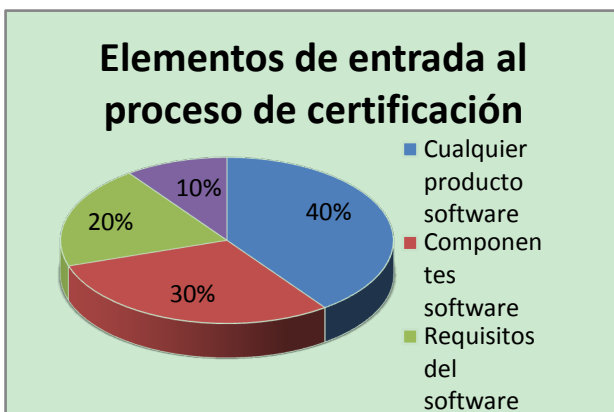


Figura 6. Elementos de entrada al proceso de certificación

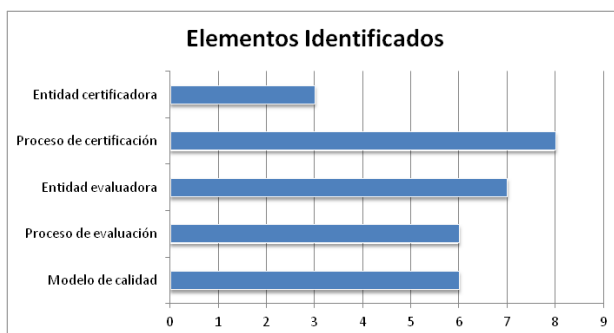


Figura 7. Recuento de elementos identificados en los estudios primarios

4. Conclusiones y Trabajo Futuro

En el presente artículo se ha presentado una revisión

sistemática de la literatura sobre la certificación de los productos software y las propuestas que existen para llevarla a cabo. La revisión se ha realizado siguiendo la metodología propuesta por Kitchenham y las plantillas desarrolladas por Biolchini.

De los 77 estudios relevantes identificados inicialmente, tan solo 10 se consideraron primarios para ser analizados. De estos estudios se puede extraer que a pesar del interés que existe por la calidad del producto software, el proceso para llevar a cabo su certificación es un campo todavía bastante incipiente en la Ingeniería del Software. A partir de los análisis realizados sobre la información extraída de los estudios se pueden presentar las siguientes conclusiones:

- La mayoría de los estudios destacan el trabajo que ya existe en la certificación de procesos de desarrollo software y la necesidad de que esta certificación sea también extendida a las características del producto para poder asegurar que el resultado final cumple con los requisitos establecidos.

- La mayoría de los estudios se basan en certificar características de calidad extraídas de normas internacionales como la ISO/IEC 9126, utilizando métodos de evaluación también estándares como los de la ISO/IEC 14598. Sin embargo, ninguna de las propuestas ha adoptado el nuevo modelo y proceso propuestos por la familia ISO/IEC 25000, probablemente debido a la novedad de este.

- Aunque ya existen varias propuestas para certificar el producto software, la mayoría de ellas carece o tiene un número reducido de aplicaciones reales en la industria del software, lo que se considera vital a la hora de poder extender su aceptación a nivel internacional y poder contrastar los niveles obtenidos por diferentes productos software.

- La mayoría de los estudios utiliza indistintamente los conceptos de evaluación y certificación, lo que se considera un error. Es necesario diferenciar entre el proceso de evaluación, realizado frente a un modelo de calidad. Y el posterior proceso de certificación realizado por un organismo acreditado e independiente que asegure la validez del certificado emitido. En este sentido se echa en falta que los estudios no hagan referencia a que el proceso de certificación se haga siguiendo los requisitos de un estándar internacional.

- Un denominador común de los estudios que presentan casos de aplicación práctica es la importancia que atribuyen a disponer de una herramienta que automatice las actividades. La mayoría de estos estudios carecen de dicha herramienta y se plantean como trabajo futuro abordar su construcción para que les permita medir, procesar los datos y analizar los resultados de una manera más rápida y sencilla.

Por otro lado, se considera que a pesar de que el procedimiento de la revisión sistemática realizado ha servido para identificar las principales aportaciones en la literatura científica relacionadas con la certificación del producto software, existen también algunas iniciativas que no se encuadran en este ámbito científico y sí están presentes en el ámbito de la industria del software. Por ello, resultaría interesante ampliar esta revisión sistemática, analizando la literatura gris que se pueda encontrar relacionada con este

tipo de iniciativas menos científicas. Algunos ejemplos de estas iniciativas son:

- el CISQ “*Consortium for IT Software Quality*”, grupo fundado por el SEI (autores del CMMI) y el OMG, y fundado por un conjunto de empresas, cuyo objetivo es definir un estándar para evaluar la calidad de los productos software.
- el IRAM “Instituto Argentino de Normalización y Certificación”, que cuenta ya entre sus servicios con un proceso para la certificación del producto software basado en la ISO/IEC 9126 y la ISO/IEC 14598.

Partiendo de todo lo anterior, se está trabajando junto con la empresa Alarcos Quality Center en una propuesta para la certificación del producto software que, basándose en la nueva familia de normas ISO/IEC 25000, identifique el modelo de calidad del producto software a utilizar, el proceso para poder llevar a cabo la evaluación de las características de calidad, las entidades independientes que participan en el proceso (laboratorio de evaluación y organismo certificador) y los requisitos de acreditación de cada una, el proceso de auditoría de certificación y las herramientas necesarias para facilitar su aplicación. De manera que después, sea todo aplicado mediante casos prácticos a la industria del desarrollo de software, con el objetivo de validar los resultados obtenidos.

AGRADECIMIENTOS

Este trabajo ha sido financiado por los siguiente proyectos de investigación: Proyecto MEDUSAS (CDTI-MICINN y FEDER IDI-20090557), Proyecto ORIGIN (CDTI-MICINN y FEDER IDI-2010043(1-5) y Proyecto PEGASO/MAGO (MICINN y FEDER, TIN2009-13718-C02-01).

REFERENCIAS

- [1] Piattini, M., et al., eds. *Medición y estimación del software: técnicas y métodos para mejorar la calidad y productividad del software*. 2008, Ra-ma.
- [2] Lee, M., *¡Qué mala suerte! Cincuenta formas seguras de fracasar en sus proyectos*. 2010: Ra-Ma.
- [3] Piattini, M., et al., *Calidad de Sistemas de Información*. 2 ed. 2011: Ra-Ma.
- [4] Carmel, E. and P. Abbott, Why "nearshore" mean that distance matter. *Communications of the ACM*, 2007. 50(10): p. 40-46.
- [5] Gartner, *Analysis of Spain as an Offshore Services Location*. 2007: Garnet Research, noviembre 2007 ID Number: G00152674.
- [6] INTECO, *Estudio sobre el modelo de factorías de software con un enfoque nearshore*. 2008: Laboratorio Nacional de Calidad de Software, Instituto Nacional de Tecnologías de Comunicación.
- [7] Piattini, M.G. and J. Garzás, *Fábricas de software: experiencias, tecnologías y organización (2ª edición actualizada)*. 2010, Paracuellos del Jarama (Madrid): RA-MA.
- [8] INTECO, *Estudio sobre la certificación de la calidad como medio para impulsar la industria de desarrollo del software en España*. 2008.
- [9] SEI, *CMMI® for Development SCAMPI Class A Appraisal Results 2010 End-Year Update*. 2011.
- [10] Garzás, J., C. Fernández, and M. Piattini, Una aplicación de la Norma ISO/IEC 15504 para la evaluación por niveles de madurez de Pymes y pequeños equipos de desarrollo. *Revista Española de Innovación Calidad e Ingeniería del Software (REICIS)*, 2009. 5(2): p. 88-98.
- [11] Kitchenham, B. and S.L. Pfleeger, *Software Quality: The Elusive Target*. *IEEE Software*, 1996. 20(1): p. 12-21.
- [12] Maibaum, T. and A. Wassysng, *A Product-Focused Approach to Software Certification*. *Computer*, 2008. 41(2): p. 91-93.
- [13] ISO, *ISO/IEC 25001 Software and system engineering – Software product Quality Requirements and Evaluation (SQuaRE) –Evaluation planning and management*, in *International Organization for Standardization*. 2005: Ginebra, Suiza.
- [14] ISO, *Software Product Evaluation–Quality Characteristics and Guidelines for their Use*. *ISO/IEC Standard 9126*. 2001, *International Organization for Standardization*.
- [15] ISO, *ISO/IEC 14598: 1999-2001. Information Technology - Software Product Evaluation - Parts 1-6*. 1999, *International Organization for Standardization*.
- [16] ISO, *ISO/IEC 25010, Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models*. 2011, *International Organization for Standardization: Ginebra, Suiza*.
- [17] ISO, *ISO/IEC 25040 Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - Evaluation process*. 2011, *International Organization for Standardization: Ginebra, Suiza*.
- [18] Kitchenham, B., *Guideline for performing Systematic Literature Reviews in Software Engineering*. Version 2.3. 2007, University of Keele (Software Engineering Group, School of Computer Science and Mathematics) and Durham (Department of Computer Science).
- [19] Biolchini, J., et al., *Systematic Review in Software Engineering*. 2005, *Systems Engineering and Computer Science Department, UFRJ: Rio de Janeiro, Brazil*.
- [20] Heck, P., M. Klabbers, and M. van Eekelen, *A software product certification model*. *Software Quality Journal*, 2009. 18(1): p. 37-55.
- [21] Carvalho, F., et al., *Embedded software component quality and certification*. *Conference Proceedings of the EUROMICRO*, 2009: p. 420-427.
- [22] Burger, E. and R. Reussner, *Performance certification of software components*. *Electronic Notes in Theoretical Computer Science*, 2011. 279(2): p. 33-41.

- [23] Serebrenik, A., et al., Requirements certification for offshoring using LSPCM. Proceedings - 7th International Conference on the Quality of Information and Communications Technology, QUATIC 2010, 2010: p. 177-182.
- [24] Yahaya, J.H., A. Deraman, and A.R. Hamdan, SCfM_PROD: A software product certification model. 2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications, ICTTA, 2008.
- [25] Yahaya, J., A. Deraman, and A.R. Hamdan, Continuously ensuring quality through software product certification: A case study. 2010 International Conference on Information Society, i-Society 2010, 2010: p. 183-188.
- [26] Hatcliff, J., et al., A Software Certification Consortium and its Top 9 Hurdles. Electronic Notes in Theoretical Computer Science, 2009. 238(4): p. 11-17.
- [27] Morris, J., et al., Software component certification. Computer, 2001. 34(9): p. 30-36.
- [28] Baggen, R., et al., Standardized code quality benchmarking for improving software maintainability. Software Quality Journal, 2011: p. 1-21.
- [29] Alvaro, A., E.S. De Almeida, and S.L. Meira, Towards a software component certification framework. Proceedings - International Conference on Quality Software, 2007: p. 298-303.