

Modelos de calidad de software: Una revisión sistemática de la literatura

Ana Villalta¹, Juan Pablo Carvallo²

¹ Fundación Consorcio Ecuatoriano para el desarrollo de Internet Avanzado (CEDIA), La Condamine 12-109, Cuenca, Ecuador, 010150.

² Departamento de Ciencias de la Computación, Universidad de Cuenca, Av. 12 de Abril s/n y Agustín Cueva, Ciudadela Universitaria, Cuenca, Ecuador, 010150.

Autores para correspondencia: ana.villalta@cedia.org.ec, pablo.carvallo@ucuenca.edu.ec

Fecha de recepción: 28 de septiembre 2015 - Fecha de aceptación: 12 de octubre 2015

RESUMEN

La evaluación de la calidad de un producto de software juega un papel fundamental tanto en la selección de componentes de software, como en la valoración de un producto resultante de un proyecto de desarrollo. Los Modelos de Calidad del Software (QMs), son artefactos específicamente diseñados y contruidos para apoyar en estos procesos. En este trabajo resume el proceso de aplicación de la técnica de Revisión Sistemática de la Literatura, utilizada para actualizar el estado del arte en relación a QM, levantado en un proceso previo realizado en 2007, con el objeto de incluir los trabajos relacionados que han sido publicados entre el 2007 y 2014. El estudio se enfoca los elementos estructurales, propiedades, métodos de construcción y aplicaciones para las cuales han propuestos los QM en este periodo.

Palabras clave: Modelos de calidad de software, QMs, revisión sistemática de la literatura, SLR.

ABSTRACT

Quality evaluation of a software product plays an important role both, in the selection of software components and quality assessment of the deliverables resulting from a software development project. Software Quality Models (QMs) are artifacts specifically designed and constructed to support these activities. This paper summarizes the process of application of a Systematic Literature Review, leaded to update the State of the art in relation to QM resulting from a previous process carried out in 2007, in order to identify and include related works that have been published between 2007 and 2014. The study focuses on the structural elements, properties, construction methods and applications for which QM have been used in this period.

Keywords: Software quality models, QMs, systematic literature review, SLR.

1. INTRODUCCIÓN

La evaluación de la calidad de un producto de software juega un papel fundamental tanto en la selección de un componente de software (ej., un sistema legado o un componente a ser adquirido, ya sea de índole comercial -COTS- o libre y de código abierto -FOSS-), como la validación de un producto resultante de un proceso de construcción a la medida (sea este construido por una unidad interna de la organización o por una empresa externa subcontratada para este propósito). Lejos de ser un problema sencillo, esta tarea suele ser muy compleja de realizar. Los Modelos de Calidad del Software (QM), son artefactos específicamente diseñados y contruidos para apoyar en estos procesos. Un QM se define como “el conjunto de factores de calidad, y de relaciones entre ellos, que proporciona una base para la especificación de requisitos de calidad y para la evaluación de la calidad” (ISO 8402). Los QM se estructuran generalmente como una jerarquía multinivel (ya sea un árbol o un

grafo dirigido), en donde los factores de calidad más genéricos se descomponen en otros más particulares, y eventualmente en métricas que permiten evaluar las distintas propiedades de un producto de software. Con el objeto de entender diversos aspectos en relación a su estructura, construcción y aplicación práctica, en el 2007 se realizó un estudio preliminar (Díaz & Ortega, 2014), que permitió determinar el estado del arte en relación a los QM con corte al 2007. Este estudio presenta los resultados de aplicar la técnica de Revisión Sistemática de la Literatura con el objeto de actualizar el estado del arte al año 2014.

2. EL ESTUDIO

En el año 2007, se efectuó un trabajo inicial con el objeto de generar un catálogo de *Atributos de Calidad* utilizados para la Evaluación de Sistemas Basados en Componentes (Díaz & Ortega, 2014). La revisión de los estudios base se llevó a cabo con un mapeo de la literatura y como resultado se obtuvo un catálogo de atributos de calidad, técnicos y no - técnicos, estructurados a modo de un QM genérico. A más de las relaciones jerárquicas existentes entre los atributos identificados, el resultado permitió identificar casos de sinonimia existentes entre ellos.

Este estudio pretende extender el trabajo inicial, incorporando una técnica formal de investigación empírica conocida como Revisión Sistemática de la Literatura (SLR). Para llevar a cabo esta investigación se utilizaron las guías propuestas en (Kitchenham & Charters, 2007) en donde se describe las fases requeridas para conducir este tipo de estudio. Como punto de partida, se propusieron seis preguntas de investigación (Tabla 1) a ser respondidas en el estudio. Las preguntas fueron diseñadas para atender a cuatro dimensiones fundamentales, los elementos estructurales, propiedades, métodos de construcción y las aplicaciones para las que han sido utilizados los QM.

Tabla 1. Preguntas de Investigación propuestas para el estudio.

PREGUNTAS DE INVESTIGACIÓN

- P1. ¿Cuáles son los elementos que estructuran los modelos de calidad de software?
 - P2. ¿Cómo se relacionan los distintos elementos entre sí?
 - P3. ¿Cuáles son las principales propiedades de los modelos de calidad del software que han sido descritas en la literatura?
 - P4. ¿Cuáles son las aplicaciones para las cuales han sido propuestos los modelos de calidad de software?
 - P5. ¿Qué métodos se han utilizado para su construcción?
 - P6. ¿Qué tipo de calidad evalúan?
-

3. BÚSQUEDA Y SELECCION DE ESTUDIOS PRIMARIOS

Para la búsqueda de estudios primarios se elaboró una lista de palabras clave, "Models", "Quality", "Software", "Product", "Elements", "Structure", "Properties", "Applications", "Metrics", que se obtuvieron de la separación de las palabras incluidas en cada pregunta de investigación. Se crearon 21 cadenas de búsqueda utilizando varias combinaciones de estas palabras. Las cadenas resultantes fueron utilizadas en una prueba piloto efectuada sobre tres bases de datos digitales, ACM Digital, IEEE Xplore y Springer. Debido al elevado número de trabajos identificados en esta prueba (2'330.259), se realizaron dos refinamientos sobre las cadenas de búsqueda; en el primero se incorporó operadores lógicos que permitieron formar cadenas de búsqueda estructuradas y se realizó una lista de términos alternativos a las palabras clave, con el objeto de obtener mejores resultados en las bases de datos que apliquen sinonimia. Como resultado se obtuvo un total de 30.768 artículos; en el segundo refinamiento se utilizaron las herramientas de búsqueda avanzada de las bases de datos digitales. De esta manera se obtuvo un total de 2048 artículos relacionados.

Una vez eliminados los trabajos redundantes, se obtuvieron 283 artículos, que fueron sometidos a un proceso de evaluación de calidad utilizando criterios de inclusión y exclusión tales como:

- En SPRINGER se seleccionarán los artículos que respondan a las preguntas de investigación en el título del documento.
- En IEEE se seleccionarán los artículos que respondan a las preguntas de investigación en su resumen y conclusiones, aun si su título no llevase las palabras clave.
- En ACM se dará lectura a todos los resúmenes de los artículos tomando en cuenta aquellos que también respondan a la selección en el título del documento.

Estos criterios permitieron clasificar a los estudios como “Seleccionado”, “Dudoso” o “Falso”. Como resultado se obtuvo 164 artículos clasificados como “Seleccionados” y se dio lectura completa al resumen y conclusiones de cada uno de ellos. De los artículos seleccionados, se extrajeron los que respondían al menos a una pregunta de investigación en su resumen o conclusiones. Como resultado, 66 artículos respondieron a por lo menos una pregunta de investigación. Se dio lectura completa los 66 artículos extraídos, de estos, 20 respondieron empíricamente a todas las preguntas planteadas en el protocolo.

4. EXTRACCIÓN DE DATOS

Para sintetizar la información de los estudios se utilizó una tabla estructurada con los campos que se describen en la Tabla 2. La Tabla 3 presenta los datos tabulados en el estudio.

5. ANÁLISIS DE DATOS

Una vez concluida la tabulación de datos se procedió a analizar la información estructurada con el objeto de responder a las preguntas de investigación propuestas.

P1.- ¿Cuáles son los elementos que estructuran los modelos de calidad de software?

De la lectura de los 20 artículos, fue posible obtener hasta 5 niveles de jerarquía con un total de 650 atributos de calidad (ver Tabla 3), siendo el “Nivel 1” el nivel más abstracto de la estructura. 77 atributos pertenecieron al nivel 1; 489 al nivel 2, siendo el nivel con mayor cantidad de atributos; 72 al nivel 3; 7 al nivel 4 y finalmente 5 al nivel 5. De los atributos mostrados, el 75% fueron del tipo métricas, mientras que un 11% fueron de tipo “Técnico funcionales”; un 13% de tipo “Técnico no funcionales” y un 1% correspondiente a otros tipos. Todos los artículos describen factores de calidad en un primer nivel; dieciocho artículos presentaron factores de calidad en un segundo nivel; siete artículos presentaron factores de calidad en un tercer nivel; dos artículos presentaron factores de calidad en un cuarto nivel; y un artículo presentó factores de calidad en un quinto nivel.

P2. ¿Cómo se relacionan los distintos elementos entre sí?

De los 20 artículos, 17 de ellos identificaron a la relación entre los elementos como jerárquica; 1 como modular jerárquica, 1 como capas y 1 no específica. La relación jerárquica entre factores de calidad organiza a los elementos en una estructura de árbol en donde cada factor posee una relación padre-hijo. En Chotjaratwanich & Arpnikanondt (2012) los autores proponen una técnica de visualización para las métricas basada en este modelo (ver Figura 1). El número de capas puede ser utilizado como una medida para determinar el nivel de detalle con el que describe el dominio de software para el cual ha sido construido el QM: a más niveles, mayor descomposición (Carvalho, *et al.*, 2010). En Correia, *et al.* (2009) se ha desarrollado un modelo en capas para la medición y la calificación de la calidad técnica de un sistema de software en términos de las características de la norma ISO / IEC 9126 (Figura 2).

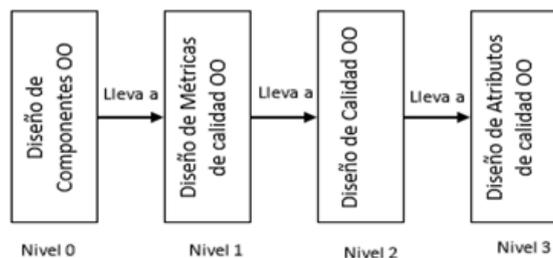


Figura 1. Ejemplo de aplicación de la relación jerárquica entre elementos.

ISO 9126	PROPIEDADES DEL SISTEMA	Volumen	Duplicación	Tamaño de la unidad	Complejidad de la unidad	Unidad de interconexión	Calidad de tests	Acoplamiento hacia adentro	Acoplamiento hacia afuera	Manejo de excepción
	MANTENIBILIDAD	Analizabilidad	X		X					
AD	Mutabilidad		X		X					
	Estabilidad		X		X		X	X		X
	Capacidad de prueba				X		X			

Figura 2. Ejemplo de aplicación de la relación en capas entre elementos.

Tabla 2. Campos para síntesis de datos de los artículos extraídos.

Campo	Descripción
Autores	Nombre asignado de manera nemotécnica compuesto por las iniciales de los apellidos de los autores, más el año de publicación del trabajo analizado.
Número de niveles	Niveles que corresponden a las tablas formato individual.
Número de atributos por nivel	Sumatoria del número de atributos.
Total de atributos	Sumatoria de todos los atributos, por niveles y por artículo.
Tipos de atributos identificados	Atributos Técnicos Funcionales (TF)
	Atributos Técnicos No Funcionales (TNF)
	Atributos No Técnicos (NT)
	Atributos Categorizados como Otros (O)
	Atributos que son Métricas (M)
	Atributos Técnicos No Definidos (T-ND)
Solapamiento en QF	Número de atributos que tienen solapamiento categorizándolos por tipo y por artículo
	Sumatoria de todos los atributos de calidad incluyendo los que tienen solapamiento.
Total de atributos	Sumatoria de todos los atributos de calidad incluyendo los que tienen solapamiento.
Total de atributos por cada nivel	Sumatoria de los atributos por cada nivel.
Total de atributos por tipo	Sumatoria de los atributos por cada tipo.
Total de atributos con solapamiento	Sumatoria de los atributos que incorporan la propiedad de solapamiento.
Incluye métricas (Si/No)	En este campo se incorpora la palabra “Si”, en el caso de que en el artículo leído se encuentren métricas de calidad, caso contrario, la palabra “No”.
Métricas aplicadas a	Se define a que factor de calidad se han aplicado las métricas encontradas, (características, sub características), dentro de la jerarquía establecida por los autores de los artículos
Incluye método de construcción (Si/No)	En este campo, se incorpora la palabra “Si”, en el caso de que en el artículo leído se encuentre un método de construcción, caso contrario, la palabra “No”.
Pasos del método de construcción	Se listan los pasos que los autores de los artículos han propuesto para la construcción del método.

Basado en qué modelo	Se detalla en qué modelo se han basado los autores de los artículos para describir su modelo actual.
Tipos de relaciones entre elementos	Se lista el tipo de relación entre los elementos, ya sea, jerárquica, de soporte, en capas, etc.
Tipo de modelo	Se describe si el modelo es fijo, a la medida o mixto.
Aplicación para la que se usa el modelo	Se describe en qué etapa de construcción del software se puede aplicar el modelo propuesto.

P3. ¿Cuáles son las principales propiedades de los modelos de calidad del software que han sido descritas en la literatura?

A más de la relación jerárquica entre elementos de distintos niveles, en (Carvallo *et al.*, 2010), se han propuesto varias maneras en que los factores de calidad pueden relacionarse, tales como:

Solapamiento.- Un factor de calidad participa en la descomposición jerárquica de varios niveles superiores, dicho factor puede evaluarse con métricas diferentes para cada uno de los factores que descompone.

Transversalidad.- Se basa en el concepto de solapamiento, pero en este caso, el factor de calidad que descompone niveles superiores, cambia de métrica y definición para cada uno de los factores que descompone.

Dependencia.- Es un factor de calidad que se relaciona con otros factores, generalmente del mismo nivel, contribuyendo de distinta manera a su consecución.

De los artículos analizados solamente dos mencionan la propiedad de solapamiento (ver Tabla 3), fila “Overlapping of QF”). 18 factores identificados en estos trabajos son de índole No-funcional y 4 de índole Funcional.

P4. ¿Cuáles son las aplicaciones para las cuales han sido propuestos los modelos de calidad de software?

Los autores han propuesto diversas aplicaciones para los modelos de calidad, algunas de las cuales se alinean con lo propuesto en (Carvallo *et al.*, 2010). La Tabla 4 presenta una relación entre los trabajos estudiados y las aplicaciones para las cuales han sido utilizados en cada una de ellas.

P5. ¿Qué métodos se han utilizado para su construcción?

Dieciséis de los artículos estudiados no presentan métodos de construcción de QM, mientras que, 4 de ellos sí lo hacen. La Tabla 5 incluye una descripción de los pasos propuestos para la construcción de QM en cada uno de estos estudios.

P6. ¿Qué tipo de calidad evalúan?

En (Carvallo *et al.*, 2010), se describen las siguientes propiedades para los modelos de calidad:

Número de Capas.- Es el nivel de detalle para describir el dominio de software.

Tipos de elementos del modelo.- Se distinguen los elementos de alto nivel, para el propósito de clasificación; y los de bajo nivel, para la descripción detallada y evaluación de características observables de los componentes

Propósito del modelo.- Al construir un modelo es necesario considerar al menos dos dimensiones: la “específico/general” y la “reutilizable/descartable”.

Tabla 3. Síntesis de los datos de los artículos

Nombre Nemotécnico	CAV09	OBL14	FKVH12	BO09	SLHF09	VSH10	MTK09	AVO9	JMS07	BF09	ASAM12	P09	ALCM309	ORCDW08	BLLO9	CK09	CA12	KKMS07	MSI11	KJYQ13	TOTAL DE ATRIBUTOS ENCONTRADOS
Título de Artículo	A Survey-based Study of the Mapping of System Properties to ISO/IEC 9126 Maintainability Characteristics	Object-Oriented Software Maintainability Measurement in the past Decade	A Mobile Software Quality Model	Analysis of Software Evolvability in Quality Models Software Engineering - Product Quality Assessment	User-oriented Measurement of Software Flexibility	An Approach based on DEVS for Evaluating Quality Attributes	An Economical Approach to Usability Testing	Requirements of Software Quality Assurance Model	Guidelines for Eliciting Usability Functionalities	Software aging assessment through a specialization of the SQuARE quality model	Using Security Metrics in Software Quality	Enabling the evolution of J2EE Applications through reverse engineering and quality assurance	Towards a set of Measures for Evaluating Software Agent Autonomy	Software quality metrics and their impact on embedded software	Quantitative Evaluation of Software Quality Metrics in Open-Source Projects	Software Quality Model for Consumer Electronics Product	A Visualization Technique for Metrics-Based Hierarchical Quality Models	Evaluating the Impact of Adaptive Maintenance Process on Open Source Software Quality	Towards Data Warehouse Quality through Integrated Requirements Analysis	Analysis of the Key Factors for Software Quality in Crowdsourcing Development: An Empirical Study on TopCoder.com	
Número de Niveles	3	3	2	2	2	2	2	5	3	2	3	1	3	2	1	2	3	2	4	2	
Nombre de Cada Nivel																					
Nivel 1	Características	Características	Cualidades	Características	Cualidades	Elementos	Factores	Factores	Atributos	Características	Fases - Criterios	Métricas	Características	Atributos	Métricas	Factores	Atributos	Atributos	Dimensiones	Dimensiones	
Nivel 2	Subcaracterísticas	Subcaracterísticas/Métricas	Atributos	Subcaracterísticas	Elementos Medibles	Métricas	Métricas	Factores	Características de usabilidad	Subcaracterísticas	Parametros		Atributos	Métricas		Métricas	Propiedades	Métricas	Atributos	Factores	
Nivel 3	Propiedades							Factores	Mecanismos de usabilidad		Métricas		Métricas			Métricas			Atributos		
Nivel 4								Factores											Atributos		
Nivel 5								Factores													
Número de Atributos por nivel																					
Nivel 1	1	1	4	1	1	6	3	1	1	1	6	9	6	5	16	9	6	6	3	5	
Nivel 2	4	332	3	6	6	27	5	4	9	5	5	3	21		48	11	27	16	23		
Nivel 3	7							12	15		21		6			11		4			
Nivel 4								7										16			
Nivel 5								5													
Total de Atributos	12	333	7	7	7	33	8	29	25	6	32	9	15	26	16	57	28	33	39	28	
Tipos de Atributos Identificados																					
Atributos Técnicos Funcionales (TF)																					
Atributos Técnicos No Funcionales (TNF)	12	1	5	7	6			29	25	6			15	5		9	16	6	4	10	
Atributos No Técnicos (NT)							3												26		
Atributos Categorizados como Otros (O)											4								3		
Atributos que son Métricas (M)		332				27	5				21	9		21	16	48	11	27			
Atributos Técnicos No Definidos (T-ND)			1																	13	
Atributos de Calidad en el Uso (CU)																					
Overlapping of QF																					
Atributos T-NT & No Funcionales																	18				
Atributos T-NT & Funcionales	4																				
Atributos Técnicos F&NF																					
Atributos T-NT F&NF																					
Total de Atributos	16	333	7	7	7	33	8	29	25	6	32	9	15	26	16	57	28	33	39	28	
Incluye métricas	No	Si	No	No	Si	Si	Si	No	No	Si	Si	Si	Si	Si	Si	Si	Si	Si	No	No	
Métricas aplicadas a		Características			Características	Características	Factores			Subcaracterísticas	Fases - Criterio	Atributos		Atributos	N/E	Factores	Propiedades	Atributos			
Incluye metodo de construcción	Si	No	No	No	No	No	No	No	Si	No	No	No	No	No	No	No	Si	No	Si	No	
Basado en que modelo	ISO / IEC 9126	Varios	Boehm, McCall, ISO 9126	McCall, Boehm, FURPS, ISO9126, Dromey	Ninguno	Ninguno	Ninguno	ISO 9126	Ninguno	Square	Ninguno	Ninguno	Ninguno	Ninguno	Ninguno	ISO 9126	QM00D	ISO/IEC 15201 [10] ISO / IEC 14764 IEEE1219-1220	No	No	
Tipos de relaciones entre	En capas	Jerarquico	Jerarquico	Jerarquico	Jerarquico	Modular/Jerarquico	Jerarquico	Jerarquico	Jerarquico	Jerarquico	Jerarquico	Jerarquico	Jerarquico	Jerarquico	N/E	Jerarquico	Jerarquico	Jerarquico	Jerarquico	Jerarquico	
Tipo de modelo	Mixto	Varios	A la medida	Mixto	A la medida	A la medida	A la medida	Mixto	A la medida	Mixto	A la medida	A la medida	A la medida	A la medida	A la medida	Mixto	Mixto	Mixto	A la medida	A la medida	
Aplicación para la que se usa el modelo	Mantenimiento	Varios	Desarrollo	Mantenimiento - Evolución del SW	Implementación	Desarrollo	Desarrollo	Desarrollo	Especificación - Requisitos	Implementacion	Desarrollo	Desarrollo	Evaluación de Autonomia	Diseño	Validación	Especificación	No Define	Mantenimiento	Diseño	Desarrollo	
Total																					
Total 1																					
Total 2																					
Total 3																					
Total 4																					
Total 5																					
Total																					
Total (TF)																				69	
Total (TNF)																				83	
Total (NT)																				3	
Total (O)																				4	
Total (M)																				490	
Total (T-ND)																				1	
Total (CU)																				0	
Total T-NT & NF																				18	
Total T-NT & F																				4	
Total T F&NF																				0	
Total T-NT F&NF																				0	
Total																				672	
Total																				0	
Total																				0	

Tabla 4. Modelos de calidad y sus aplicaciones

Nombre nemotécnico	Aplicación	Descripción de la aplicación
CKV09	Mantenimiento	Medición de la capacidad de mantenimiento de productos de software.
OBL12	No Define	
FKW12	Desarrollo	Evaluación del proceso de desarrollo de aplicaciones móviles.
BC09	Evolución del SW	Análisis de varios modelos de calidad existentes con el fin de evaluar cómo se aborda la evolución del software en estos modelos.
SLHF09	Implementación - Planificación	Uso los puntos flexibles para implementar las funciones del sistema.
VSH10	Desarrollo	Enfoque para la simulación de software en una etapa temprana de su desarrollo, usando su arquitectura.
MTK09	Desarrollo	Enfoque en pruebas de usabilidad que permite las pruebas en el proceso de desarrollo más temprano.
AW09	Diseño - Desarrollo	Proporción de un producto seguro de software de ingeniería para el desarrollo, mediante la adopción de a la norma ISO 9126.
JMS07	Especificación - Requisitos	Determinación de características de usabilidad con implicaciones funcionales para lidiar con ellos en la etapa de requisitos.
BF09	Implementación - Mantenimiento	Enfoque de monitoreo de envejecimiento de software para proporcionar una especialización del modelo de calidad Squar.
ASAM12	Desarrollo	Desarrollo de métricas de seguridad en diferentes fases de desarrollo de software y validación en base a unos criterios estandarizados.
P09	Desarrollo	Definición de métricas para medir diferentes aspectos de aplicaciones y la identificación de patrón para el desarrollo.
AFMS09	Evaluación- Implementación	Definición de medidas para evaluar la autonomía de un agente.
ORCDW08	Diseño	Verificación y mejora de uso de métricas, con fines de exploración del espacio de diseño y en el diseño de sistemas embebidos.
BLL09	Validación	Definición y validación de los umbrales absolutos para métricas de calidad. Con el fin de mejorar la eficiencia de validación.
CK09	Reutilización	Determinación tres factores críticos de calidad en la norma ISO 9126, teniendo en cuenta la reutilización como objetivo de calidad.
CA12	No Define	
KKMS07	Mantenimiento	El concepto de aplicación de mantenimiento ha sido resuelto de acuerdo a la lectura del artículo completo.
MSI11	Análisis y Diseño	Definición de metodología para el análisis de requisitos en sistemas de almacenamiento de datos denotados.
KJYQ13	Desarrollo	Identificación de factores clave para la calidad del software en el crowdsourcing en su etapa de desarrollo.

Tabla 5. Métodos de construcción de los modelos de calidad.

Pasos del método de construcción				
AUTOR	Correia, J., Kanellopoulos, Y., Visser, J.	Juristo, N., Moreno, A., & Sanchez-Segura, M.I.	Chotjaratwanich, U., Arpnikanondt, C.	Munawar, M., Salim, N., Ibrahim, R.
1	ISO/IEC 9126 (sub) características, influencia y puede ser causa de	Identificación del mecanismo de usabilidad dirigida por el patrón	Componentes de diseño	Identificación de los factores, atributos y dimensiones de datos que se encuentran fuera de los requisitos
2	propiedades del sistema, indicadores y puede ser medido por	El problema dirigido por cada patrón	Métricas de calidad de diseño	Identificación de jerarquías de dimensión. Modelos de datos de fuentes de datos usadas para determinar las jerarquías de atributos de dimensión. Uno de los métodos impulsados de datos
3	medidas de código fuente	El contexto de usabilidad en el cual este patrón será útil	Calidad del diseño	
4		La solución al problema abordado por el patrón	Atributos de la calidad del diseño	
5		Los patrones relacionados se refieren a otros patrones de elicitación de usabilidad cuyos contenidos están relacionados con el objeto de estudio		

Separación entre los Elementos Internos y Externos.- Los factores externos son los que pueden ser directamente percibidos por los usuarios y afectan a su trabajo; los factores internos se refieren a las características constructivas de los componentes, que son accesibles por los fabricantes.

- **Relación de las métricas con los factores de calidad.-** Ciertas propuestas de modelos de calidad existentes, incluyen métricas asociadas a un nivel más detallado de descomposición, aunque en algunos casos, requieren que las métricas sean aplicadas a los niveles más altos o abstractos de la jerarquía.

En este estudio se han diferenciado 69 atributos funcionales que evalúan la calidad externa y 91 atributos no funcionales que están divididos entre evaluar la calidad interna. Además existió un hallazgo significativo en métricas en relación al trabajo previo descrito en (Díaz & Ortega, 2014). Un total de 490 métricas han sido identificadas en estos estudios, un extracto de ellas se presenta en la Tabla 6.

Tabla 4. Extracto de métricas encontradas en el estudio (Mueller *et al.*, 2009).

An economical approach to usability testing software engineering - Product Quality Assessment							
No	Nivel	[MTKF09]	Tipo	Referencias	Descripción	Descripción de la métrica	Interpretación de la métrica
Usabilidad basada en esfuerzo							
1	1	Esfuerzo	NT	Pag.1 (2. Effort-based Usability)			E = (Emental Ephysical)
2	1 1	E-mental	NT		Cantidad de esfuerzo mental para completar la tarea.		Emental = (Eyemental Eothermental)
3	1 1 1					Eojo-mental	Cantidad de esfuerzo mental necesario para mover y enfocar los ojos para completar la tarea.
4	1 1 2					Eotro-mental	Cantidad de esfuerzo mental no especificado necesarios para completar la tarea.
5	1 2	E-Físico	NT		Cantidad de esfuerzo físico para completar la tarea.		Ephysical = (Emanual_physical Eye_physical Eother_physical)
6	1 2 1					Emanual - físico	Cantidad de esfuerzo manual para completar la tarea. El esfuerzo manual incluye, pero no se limita a, el movimiento de los dedos, manos, brazos, etc.
7	1 2 2					E-ojo-físico	Cantidad de esfuerzo físico en el ojo para completar la tarea
8	1 2 3					E-otro-físico	Cantidad de esfuerzo físico no especificado para completar la tarea.

6. CONCLUSIONES

En este trabajo se presentan los resultados de un estudio realizado con el objeto de actualizar el estado del arte en relación a QM, considerando sus elementos estructurales, propiedades, métodos de construcción y aplicaciones. Para cumplir este objetivo se usó la técnica de Revisión Sistemática de la Literatura, siguiendo las guías propuestas en (Kitchenham & Charters, 2007). Como parte del protocolo se plantearon 6 preguntas de investigación; para responderlas se revisaron 3 bases de datos digitales que contenían publicaciones comprendidas entre el año 2007 y 2014. Sesenta y seis artículos seleccionados fueron revisados de manera íntegra, 20 de los cuales proporcionaron la evidencia empírica necesaria para responder a las preguntas de investigación.

Los resultados obtenidos permiten extender el estado del arte resultante de un estudio previo que comprendió artículos publicados hasta el año 2007. A futuro se pretende publicar el catálogo completo de elementos de calidad identificados en estos estudios, con el objeto que puedan ser accesibles por parte de la comunidad científica y técnica. También se pretende construir un repositorio semántico en el que se almacene los elementos identificados, conjuntamente con anotaciones, que permitan identificar relaciones de antonimia, sinonimia, jerarquía y dependencia a diversos niveles. Se espera que este repositorio pueda ser utilizado para optimizar el proceso de construcción de QM a futuro.

REFERENCIAS

- Abdi, A., A. Souzani, M. Amirfakhri, A. Moghadam, 2012. *Using security metrics in software quality assurance process*. Sixth International Symposium on Telecommunications, 1, 4.
- Alonso, F., J. Fuertes, L. Martinez, H. Soza, 2009. *Towards a set of measures for evaluating software agent autonomy*. Eight Mexican International Conference on Artificial Intelligence, 1, 6.
- Alsultanny, Y., A. Wohaiishi, 2009. *Requirements of software quality assurance model*. Second International Conference on Environmental and Computer Science, 1, 5.
- Barkmann, H., R. Lincke, W. Lowe, 2009. *Quantitative evaluation of software quality metrics in open-source projects*. International Conference on Advanced Information Networking and Applications Workshops, 1, 6.
- Bogado, V., S. Gonnet, H. Leone, 2010. *An approach based on DEVS for evaluating quality attributes*. XXIX International Conf. of the Chilean Computer Science Society (SCCC), 1, 9.
- Bombardieri, M., F. Fontana, 2009. *Software aging assessment through a specialization of the SQuARE quality model*. ICSE Workshop on Software Quality, 1, 6.
- Breivold, H., I. Crnkovic, 2009. *Analysis of software evolvability in quality models*. 35th Euromicro Conference on Software Engineering and Advanced Applications, 1, 4.
- Carvalho, J., X. Franch, C. Quer, 2010. *Calidad de componentes software*. Calidad del Producto y Proceso Software, 30, España: RA-MA Editorial.
- Chotjaratwanich, U., C. Arpnikanondt, 2012. *A visualization technique for metrics-based hierarchical quality models*. 19th Asia-Pacific Software Engineering Conference, 1, 4.
- Correia, J., Y. Kanellopoulos, J.A. Visser, 2009. *A survey-based study of the mapping of system properties to ISO/IEC 9126 maintainability characteristics*. IEEE International Conference on Software Maintenance, 1, 10.
- Diaz, A., M. Ortega, 2014. *Catálogo de atributos de calidad para la evaluación de sistemas basados en componentes*. Universidad del Azuay, Sitio web: dspace.uazuay.edu.ec.
- Franke, D., S. Kowalewski, C. Weise, 2012. *A mobile software quality model*. 12th International Conference on Quality Software, 1, 4.
- International Standards Organization, 1994. *ISO International Standard 8402: Quality Management and Quality Assurance Vocabulary* (2nd ed.).

- Juristo, N., A. Moreno, M-I. Sanchez-Segura, 2007. *Guidelines for eliciting usability functionalities*. IEEE Transactions on Software Engineering, 33, 15.
- Kim, C., K. Lee, 2009. *Software quality model for consumer electronics product*. International Conference on Quality Software, 1, 6.
- Kitchenham, B., S. Charters, 2007. *Guidelines for performing systematic literature reviews in software engineering*. EBSE Technical Report, 1, 65.
- Kozlov, D., J. Koskinen, J. Markkula, M. Sakkinen, 2007. *Evaluating the impact of adaptive maintenance process on open source software quality*. First International Symposium on Empirical Software Engineering and Measurement, 1, 10.
- Li., K., J. Xiao, Y. Wang, Q. Wang, 2013. *Analysis of the key factors for software quality in crowdsourcing development: An empirical study on TopCoder.com*. IEEE 37th Annual Computer Software and Applications Conference, 1, 6.
- Mueller, C., D. Tamir, O. Komogortsev, L. Feldman, 2009. *An economical approach to usability testing*. 33rd Annual IEEE International Conf. on Computer Software and Applications, 1, 6.
- Munawar, M., N. Salim, R. Ibrahim, 2011. *Towards data warehouse quality through integrated requirements analysis*. International Conference on Advanced Computer Science and Information System, 1, 6.
- Oliveira, M., R. Redin, L. Carro, L. da Cunha Lamb, F. Wagner, 2008. *Software quality metrics and their impact on embedded software*. 5th International Workshop on Model-based Methodologies for Pervasive and Embedded Software, 1, 10.
- Orenyi, B., S. Basri, J. Low Tan, 2012. *Object-Oriented software maintainability measurement in the past decade*. International Conference on Advanced Computer Science Applications and Technologies, 1, 6.
- Perin, F., 2009. *Enabling the evolution of J2EE applications through reverse engineering and quality assurance*. 16th Working Conference on Reverse Engineering, 1, 4.
- Siwei, P., S. Limin, L. Hui, L. Feng, 2009. *User-oriented measurement of software flexibility*. WRI World Congress on Computer Science and Information Engineering, 7, 5.