Universidad de Alcalá

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

TESIS DOCTORAL

Metodología de desarrollo de modelos de calidad orientados a dominio y su aplicación al dominio de los productos finales de seguridad de tecnologías de la información

Autora:

MARÍA TERESA VILLALBA DE BENITO

Director:

Dr. Luis Fernández Sanz

Co-Director:

Dr. José Javier Martínez Herráiz

Alcalá de Henares, junio de 2009

Agradecimientos

Quiero agradecer a todas las personas que de una forma u otra han hecho posible esta tesis. La lista es larga y son muchos los que con sus consejos, escuchándome o dándome ánimos han contribuido a que este trabajo siguiera adelante. De manera especial me gustaría dar las gracias a:

Mis directores de tesis, por su confianza, guia y apoyo. Sin su empeño y conocimientos este trabajo no hubiese sido posible. Gracias por vuestros sabios consejos y correcciones.

A César, mi compañero, por soportar con paciencia mis vaivenes, por entender mis ausencias, por su apoyo y amor constante e incondicional, por estar siempre ahí. Pero también por sus inteligentes observaciones, por nuestras enriquecedoras discusiones matemáticas y sobre todo porque sin él no hubiese tenido una herramienta web para publicar los cuestionarios de recogida de datos de este trabajo. Sencillamente por todo, no me cansaré nunca de darte las gracias.

Al resto de mi familia por comprender mis ausencias, por su soporte y aliento durante todo este tiempo y por todos los momentos extraordinarios que pasamos juntos. Y en especial a mi padre que siempre nos inculcó la importancia de la superación constante tanto intelectual como personal.

A mis amigos por todos esos momentos de aliento y por estar siempre ahí para compartir un rato juntos. A mis compañeros de despacho que siempre están dispuestos a prestarme su ayuda en este u otros trabajos y también por esos cafes y ratos de distensión tan necesarios.

A todos, gracias por estar ahí.

Resumen

Los modelos de calidad son una parte fundamental en los procesos de desarrollo y evaluación de la calidad del software. El uso de estos modelos se ha generalizado sobre todo desde la aparición de modelos de calidad estándar. Estos modelos, de acuerdo a su naturaleza de estándares, constituyen modelos genéricos y no directamente aplicables a la práctica diaria, por lo que requieren de un esfuerzo adicional para adaptarlos a cada dominio de aplicación específico. De ahí que existan multitud de trabajos en los que el objetivo es la definición de modelos de calidad reutilizables para dominios de aplicación específicos que, al no tener que ser definidos para cada proyecto desde cero, ahorren tiempo. Por otra parte, este tipo de modelos pueden ofrecer una evaluación más exacta ya que sus propiedades se pueden definir de forma más precisa.

Tras llevar a cabo una revisión sistemática de los modelos de calidad orientados a dominio existentes en la actualidad, se deduce que dichos modelos están basados en el conocimiento y/o experiencia de los investigadores y, en consecuencia, no pueden generalizarse a otros entornos o proyectos, la cual, es la principal ventaja de estos modelos. Por otra parte, estos modelos de calidad definen únicamente características y atributos del software sin determinar ni validar la importancia relativa o peso de los mismos, ni tampoco, las relaciones de influencia existentes entre las diferentes características del modelo, ambas propiedades fundamentales en las evaluaciones cuantitativas, sobre todo, en situaciones en las que no se dispone de requisitos de usuario.

Nuestra propuesta define un proceso sistemático de desarrollo de modelos de calidad que, basándose en la experiencia y conocimiento de un amplio número de expertos, genera modelos de calidad orientados a dominio obteniendo, no sólo los factores sino también la importancia relativa o peso de los mismos sobre cada característica de calidad y las relaciones de influencia entre las diferentes características.

Para apoyar la propuesta de esta investigación se incluye un extenso trabajo de validación. Para llevar a cabo dicho proceso, tras obtener los atributos a través de estándares y otras publicaciones relacionadas, se realiza una validación preliminar que confirma la aplicabilidad de los mismos mediante su aplicación en 3 casos de estudio reales para 2 empresas de desarrollo de productos software finales. Esta validación preliminar permite obtener una serie de lecciones aprendidas que proporcionan la base para la definición del proceso aquí presentado. Además, se lleva a cabo una validación a posteriori tanto de la utilidad y necesidad de modelos de calidad como los aquí generados como de su aplicabilidad, a través de la presentación de un caso de estudio en el ámbito de la seguridad informática.

Abstract

Quality models are an essential part of today´s development and quality evaluation processes of software. Their use has been generalized above all since the appearance of standard quality models. These models, according to their nature of standards, are generics and not directly applicables in the daily practice so it is required an effort to fit them to each specific application domain and evaluation approach. Due to this, there are many research works with the main objective of the definition of reusable quality models oriented to specific application domains which save time because you do not have to define them from scratch each time. Moreover, this type of models may offer a more accurate evaluation because of their properties can be more precisely defined.

Later to carry out a systematic review of the today domain-oriented quality models, we concluded that these models are based on the knowledge and/or experience of the researchers and, therefore, they can not be generalized to others environment or projects, which is the main advantage of them. On the other hand, these quality models define only software characteristics and attributes not obtaining or validating their relative importance or weitht, neither the influence relations between the different model´s characteristics, both of them main properties of quantitative evaluations, above all when there is not user requirements or they are not reliable.

Our approach define a systematic process for quality models development which generate domain-oriented quality models based on the knowledge and experience of a wide number of experts, obtaining not only the factors, but also the relative importance or weigth and the influence relationships between them.

In order to support this approach, we include an extensive validation. To fullfil this validation process, next to obtain the attributes from standards and other related literature, we carry out an early validation to confirm their applicability through their

use in three real world study cases for two software development companies. This early validation allowed us to obtain multiple lessons learned which provide the basis of the process here presented. Moreover, we conducted a later validation on both the utility and necessity of these quality models and their applicability by means of a case study on Information Technology Security field.

ÍNDICE

1. Índice de tablas

2. Índice de figuras

Capítulo 1. Introducción

1.1.Presentación

La demanda, por parte de las organizaciones, de software con un tiempo rápido de puesta en producción y reducidos costes de desarrollo, ha llevado en los últimos años al uso masivo de productos comerciales genéricos en contra de largos proyectos de desarrollo de soluciones específicas para cada organización [Voas 1998, Ochs, Pfahl*, et al.* 2001]. El uso de software genérico comercial promete un tiempo rápido de puesta en producción, reducidos costes de desarrollo, incremento en la productividad y la posibilidad de que las compañías se centren en su propio negocio en lugar de en el desarrollo del software o su subcontratación [Maiden y Ncube 1998, Voas 1998]. Además, este tipo de software tiene como ventaja añadida, que el soporte del mismo es dado por el proveedor con experiencia en múltiples proyectos, o bien, por expertos con experiencia en el producto, permitiendo al cliente obtener un mejor servicio al reutilizar el conocimiento adquirido por el equipo de soporte en otros proyectos ("Saber hacer" o *"Know-how")*.

La enorme expansión de este tipo de productos y los altos costes asociados a la compra e implantación de los mismos ha impulsado la búsqueda de procesos de selección antes de su adquisición. La selección de productos software que se adapten a las necesidades de los usuarios y que, a su vez, cumplan ciertas características de calidad que aseguren su correcto funcionamiento, es un proceso clave sobre todo cuando una mala selección puede, cuanto menos, suponer pérdidas en el negocio. Una elección incorrecta no reconocida a tiempo puede ser extremadamente costosa para la organización. Entre las consecuencias de una deficiente evaluación de software o la

ausencia de la misma podemos destacar dificultades en la planificación, presupuestos excesivos, mala o desconocida calidad de productos adquiridos, proyectos interrumpidos o interminables u objetivos conseguidos con meses o años de retraso.

Según [ISO 1999b], los productos software deben ser evaluados por terceros (por ejemplo, laboratorios independientes de evaluación) según los requisitos de las normas ISO/IEC 14598-5 [ISO 1998b] y ISO/IEC 12119 [ISO 1994] (en la actualidad sustituida por la norma ISO/IEC 25051 [ISO 2005g]). El proceso de evaluación de la calidad del software en productos finales permite:

1. Aprender en profundidad y de modo más sistemático acerca de las características del producto proporcionando un proceso documentado de la experiencia que pueda ser utilizado en el proyecto actual o en futuros proyectos.
2. Aprender sobre las carencias de un producto para poder solventarlas antes de su comercialización.
3. Elaborar informes técnicos y clasificar productos según su calidad (por ejemplo, para revistas especializadas).
4. Evaluación final de la calidad del producto (por ejemplo para la selección del producto que más apropiado antes de su adquisición).
5. Comparación de productos (por ejemplo, compararlos con otros productos de la competencia, o para evaluar si cambiar el producto en uso por otro).

Para poder seleccionar un producto debemos compararlo con otros con el fin de elegir aquél de mayor calidad. Como la calidad está compuesta de muchas características, la noción de calidad es normalmente capturada en un modelo que describe las características compuestas y sus relaciones [Fenton y Pfleeger 1997]. Por otra parte, según el estándar ISO/IEC 9126 [ISO 2001a], la calidad de un producto software se debería evaluar usando un modelo de calidad. Aunque existen diferentes modelos de calidad para la evaluación del software, como se explicará en detalle en el siguiente capítulo, el estándar comúnmente aceptado por la comunidad internacional es la norma ISO/IEC 9126 [ISO 2001a]. En el modelo de calidad ISO/IEC 9126 se define un modelo básico de seis características de calidad como base para la evaluación de la calidad de productos software cualesquiera. Dicha norma, dada su propia naturaleza de

estándar, cubre un espectro tan genérico de productos que no es posible aplicarla directamente. Esto hace que sea necesario adaptarla a dominios de productos específicos, de forma que, sea posible concretar los factores que deberán evaluarse y qué peso tendrán unos con respecto a otros. El modelo así obtenido será entonces un modelo práctico para su aplicación por parte de evaluadores. Por otra parte, el estándar ISO/IEC 14598-5 [ISO 1996] proporciona una guía para la implementación práctica de evaluaciones de productos software, esto es, el proceso que hay que seguir para evaluar un producto a partir de su modelo de calidad. Por tanto, puede ser usado para aplicar los conceptos descritos en ISO/IEC 9126 es más, uno de los objetivos principales de la evaluación de la calidad de productos software es medir un conjunto de características respecto a unos requisitos de calidad especificados para un dominio de aplicación y un perfil de usuario dados.

Por las características intrínsecas de los productos finales, podemos deducir que estos productos se caracterizan porque:

- No existe ningún tipo de intervención por parte del comprador del producto en el proceso de desarrollo del mismo;
- No es posible acceder al código fuente del producto, es decir, el producto es lo que suele denominarse una "caja negra", lo cual, hace que la evaluación sea más complicada [Dean 1999].
- El soporte del producto lo proporciona el proveedor en forma de actualizaciones, parches o nuevas versiones. Los rápidos cambios estos productos dificultan su evaluación [Oberndorf 1997]
- En la evaluación de productos COTS (del inglés: *Commercial off-the-shelf*) o productos software comerciales no es suficiente con tener en cuenta los factores técnicos como funcionalidad, usabilidad o eficiencia, sino que hay otros criterios que deben tenerse en cuenta relacionados con el negocio, tales como el coste del producto, el tipo de licencia o el soporte ofrecido por el fabricante [Powell, Vickers, *et al.* 1997].

- Los fabricantes de productos COTS dan prioridad a la funcionalidad de sus productos por ser las que permiten una mayor competitividad en el mercado[Hissam, Carney, *et al.* 1998]. Otras propiedades de los productos que no son importantes desde el punto de vista del fabricante, probablemente se incluyan en posteriores versiones del producto. Por tanto, características tan importantes como la fiabilidad, la usabilidad o la eficiencia pueden pasar a un segundo plano en relación con la funcionalidad del producto final.

- Los productos COTS son más susceptibles a las vulnerabilidades que los productos hechos a medida. El motivo de esto es que los posibles intrusos tienen acceso al mismo conocimiento sobre los productos que los propios integradores incluidos los documentos sobre problemas conocidos y cómo solucionarlos. Este conocimiento permite a los atacantes instalar y probar los productos hasta encontrar vulnerabilidades que puedan comprometer los sistemas.

Todo esto hace que el proceso de evaluación de los productos finales difiera de los procesos de evaluación de productos intermedios, tanto en lo que a características de calidad a evaluar se refiere y, por tanto, al modelo de calidad definido, como al proceso de evaluación en sí. Al no participar el comprador en el proceso de desarrollo del producto final, sus requisitos no quedarán reflejados en el mismo. Esto afectará directamente al proceso de evaluación, en el que deberá contemplarse que la especificación de requisitos no podrá ser tan estricta como la definida en un producto desarrollado a medida, pues corremos el riesgo de que ningún producto se adapte a los requisitos y la evaluación fracase teniendo que asumir los costes que ello conlleva. Por otra parte, el no disponer de acceso al código fuente, ni a los productos intermedios generados durante el desarrollo del producto, implica que sólo podrá evaluarse el producto a través de pruebas del mismo en ejecución. Por último, destacar la importancia que puede tener en el proceso de selección del producto características no técnicas como puedan ser el coste del producto, el prestigio del proveedor de software, el soporte que éste proporcione del producto o la posibilidad de dar formación a los usuarios entre otros. Todo ello afecta a las características del producto que se evaluarán y, por tanto, al modelo de calidad que se defina.

En este sentido existen diferentes trabajos en la actualidad para la adaptación de los estándares de calidad ISO/IEC 9126 e ISO/IEC 14598 (actualmente integrados como ISO/IEC 25000 [ISO 2005f] a la selección de productos finales comúnmente denominados COTS como se explicará en detalle en el Capítulo 2. Sin embargo, aún particularizando los genéricos modelos de calidad existentes al dominio de productos finales COTS, el modelo resultante seguirá siendo demasiado genérico para que resulte útil en su aplicación práctica. Esto es debido a la enorme variedad de productos finales que existen en la actualidad y en lo diferentes que son unos de otros, lo que hace que las características de calidad que los caractericen en el modelo puedan ser muy distintas de unos a otros. Debemos, entonces, seguir refinando el modelo para cada subdominio de productos que tengan características comunes entre sí, de forma que permitan, aun manteniendo ciertos grados de libertad en el modelo como para que éste sea reutilizable para todos los productos de ese subdominio, que el evaluador (o el comprador del producto con conocimientos de evaluación) pueda aplicar el modelo de la forma más directa posible, y que el modelo sea adaptable a las características concretas de cada proceso de selección. En este sentido, con respecto a las características de calidad será necesario, por tanto, particularizar el modelo de calidad para que éste sea aplicable en la práctica tanto al dominio de los productos finales, como a cada subdominio de aplicación específico (por ejemplo, el subdominio de productos par la protección de la seguridad de los sistemas). Todo ello, con el fin de minimizar los tiempos de evaluación al máximo para que las organizaciones los incluyan en sus apretadas planificaciones de proyecto, pero sin perder por ello la fundamental característica de reutilización del mismo ni el rigor necesario en toda evaluación.

Dada la complejidad de la evaluación de las características que definen la calidad de un producto software, existen multitud de trabajos en los que se proponen métodos para medir las diferentes características definidas en ISO/IEC 9126 como la usabilidad, eficiencia o la seguridad. Y es la seguridad la que en los últimos tiempos ha tomado una mayor importancia como puso de manifiesto la adopción como estándar ISO/IEC 15408

de la versión 2.1 de los Criterios Comunes (*Common Criteria*[1]) para la evaluación de la seguridad de los productos software. En un principio la seguridad era sólo una preocupación de los gobiernos, y en concreto del ámbito militar, que debían proteger la información confidencial y sus sistemas con potentes productos de seguridad de red. Hoy día, con el aumento de las transacciones comerciales a través de la red y la presencia de las organizaciones en internet cada vez es mayor la preocupación y concienciación de la importancia de la seguridad. Por otra parte, en los últimos años han aumentado de forma considerable el número de vulnerabilidades en los Sistemas de Información tal como puede observarse en la Figura 1. Esto provoca a su vez un aumento en la dependencia y necesidad de utilizar software para la protección para nuestras redes y sistemas. En este sentido, los proveedores de software han reaccionado rápidamente desarrollando una gran cantidad de productos que incluyen soluciones a los más diversos problemas de seguridad. También se ha dotado a los sistemas tradicionales que están generalmente expuestos a las amenazas de la red, como sistemas operativos o bases de datos, de funciones de seguridad aumentando así su robustez en relación a las funciones de seguridad.

Existe, por tanto, una gran cantidad de productos para los que existe una demanda creciente sobre el nivel de seguridad que deben cumplir a la hora de ser seleccionados. Como consecuencia de ello, surge la necesidad de disponer de mecanismos para la evaluación del cumplimiento de unas características básicas de seguridad según el producto y el entorno en el que el mismo va a ser utilizado. Con el fin de certificar la seguridad de los productos software se crearon los ya mencionados Criterios Comunes (*Common Criteria)* que actualmente conforman el estándar ISO/IEC 15408 [ISO 2005b].

[1] La información relativa al *Common Criteria* puede encontrarse en http://www.commoncriteriaportal.org/

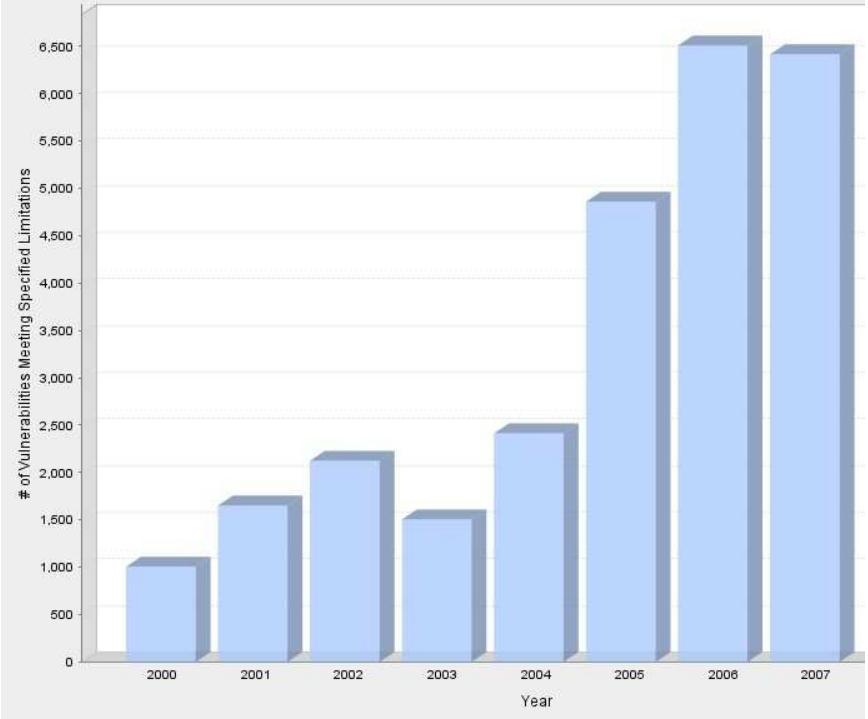


Figura 1. Estadísticas de vulnerabilidades por año registradas en la base de datos del NIST [NIST 2009]

El estándar ISO/IEC 15408 define los criterios de seguridad para de las Tecnologías de la información (en adelante TI) garantizando que las funciones de seguridad de tales productos y sistemas reúnen los requisitos declarados para ese tipo de productos según su perfiles de protección (PP, *Protection Profile*). Un perfil de protección define un conjunto de objetivos y requisitos de seguridad, independientes de la implantación, para una categoría de productos que cubre las necesidades de seguridad comunes a varios usuarios. Los perfiles de protección son reutilizables. Los perfiles de protección están compuestos de requisitos funcionales (características funcionales relacionadas con la seguridad de los productos que se deben verificar según la robustez requerida, por ejemplo: básica, media, alta) y de confianza en la evaluación. Tras la evaluación de la seguridad de un producto software final a través de la norma ISO/IEC 15408, el producto recibe un nivel de confianza o seguridad (EAL, *Evaluation Assurance Level*)

que indica el nivel de seguridad del producto (de 1 a 7). Las listas de productos junto con su nivel de seguridad son públicas[2].

A menudo cuando se evalúa una posible solución software, las organizaciones realizan una búsqueda de productos que satisfagan los requisitos funcionales requeridos. Además, con especial importancia en los casos en los que la seguridad de los productos es un factor esencial para la selección de los mismos, éstos deben satisfacer las características de seguridad o el nivel EAL requerido. En dominios software en los que la seguridad es una característica fundamental, esto puede llevar a los compradores a adquirir productos teniendo en cuenta sólo las funciones del producto y su nivel de seguridad EAL, o lo que es lo mismo, no teniendo en cuenta propiedades como, por ejemplo, la fiabilidad, el rendimiento o la usabilidad. Por ello, es importante disponer de un modelo que combine los estándares comúnmente aceptados sobre evaluación de calidad del software (ISO/IEC 9126 e ISO/IEC 14598) y sobre evaluación de la seguridad (ISO/IEC 15408), de forma que los evaluadores dispongan de un método sistemático y de aplicación práctica para evaluar estos productos finales antes de adquirirlos.

Por otra parte, el disponer de un proceso de selección sistemático y práctico es más importante si cabe cuando los productos a seleccionar tienen funciones críticas en relación a la seguridad de los datos y de los sistemas de la organización como es el caso de los productos de protección o seguridad de red[3]. Entre otras razones que motivan el interés acerca de la evaluación de los productos relacionados con la seguridad se encuentran los siguientes:

- Los necesitan un gran número de organizaciones. Hoy en día, cualquier organización por pequeña que sea necesita tener instalado un software mínimo contra amenazas en

[2] Los productos evaluados junto con su nivel de seguridad se encuentran disponibles en la web de los Criterios Comunes (http://www.commoncriteriaportal.org/products.html).

[3] Se consideran productos de seguridad de red aquellos cuya función principal es proteger las redes y sistemas contra intrusiones maliciosas.

la red. Es más, cualquier usuario doméstico conectado a internet también necesita de este tipo de soluciones.

- Tienen un gran impacto en las organizaciones debido a su criticidad. Estas aplicaciones protegen los sistemas y datos de la organización.

- Es necesario volver a evaluar cada cierto tiempo los productos por las nuevas amenazas que surgen en la red. En un área tan variable como ésta, los proveedores deben suministrar soporte, en forma de nuevas versiones, actualizaciones o parches, para que los usuarios puedan enfrentarse a las nuevas amenazas que tan a menudo surgen. Estos cambios en el panorama de la seguridad, hacen que los usuarios deban evaluar sus productos periódicamente para verificar que cumplen las políticas de seguridad definidas por la organización.

- Tienen características de calidad específicas para todos los productos. Esto convierte este dominio de productos en un buen candidato para la definición de un modelo de calidad con características comunes a todos los productos. Por otra parte, tendremos características particulares para cada subdominio.

- Existen una gran variedad de productos en el mercado. El aumento de estas amenazas en los últimos años, se ha convertido en un nicho de negocio para las compañías proveedoras de software que, en poco tiempo, han desarrollado una gran diversidad de soluciones software a los problemas y necesidades que han ido surgiendo. Dichas soluciones, además han ido evolucionando hacia paquetes que comprenden varias de ellas, aumentando así su funcionalidad.

- Cambian muy rápidamente. Muchos proveedores de software liberan nuevas versiones de productos al menos una vez al año, lo cual, en muchos casos (avances tecnológicos, por ejemplo) invalida las evaluaciones. En concreto, en el caso de la evaluación de seguridad, los Criterios Comunes establecen que la certificación de seguridad otorgada a un producto perderá su validez en cuanto se realice un cambio de versión del producto certificado. La razón está en que, aunque los cambios de

versiones se desarrollan para mejorar errores en el software y vulnerabilidades del producto, también puede añadir otros o modificar algo que estaba bien.

Por último, destacar que, en lo que a la evaluación y selección de productos de seguridad de red se refiere, éste es un proceso crítico que no debería realizarse a través de un proceso ad-hoc. Al ser estos productos los que salvaguardan los valiosos datos de las organizaciones, seleccionar un producto que no cumpla correctamente sus funciones puede resultar en grandes pérdidas no sólo económicas sino de reputación de la organización y, en consecuencia, en pérdida de la confianza de los clientes o, en el peor de los casos, de los clientes mismos.

Por tanto, podemos afirmar que en el campo de la evaluación de productos finales de seguridad en red, se observa la necesidad de un proceso sistemático que permita una evaluación y selección de este tipo de productos perfectamente documentada, reproducible y repetible con el fin de obtener un método práctico, basado en los estándares internacionalmente reconocidos, que facilite la evaluación de productos finales de seguridad, así como, de herramientas efectivas que permitan tomar una decisión corroborada del producto que mejor se adapta a las necesidades propias antes de la selección y compra del mismo.

1.2. Objetivos y aportaciones originales

1.1.1 Objetivos

Este trabajo de tesis se centra en la mejora de los procesos de selección de productos de protección y seguridad informática proporcionando una solución para tratar algunos de los problemas abiertos relacionados con la evaluación y mejora de dichos productos. Esta tesis propone un modelo orientado a dominio reutilizable tanto en los procesos de evaluación o selección con el objetivo de obtener la solución de seguridad de mayor calidad, como en los procesos de desarrollo del software para la mejora de soluciones específica. Para la obtención de dicho modelo ha sido necesario desarrollar una metodología de construcción de modelos de calidad a través de la ampliación de las ya

existentes. Son, por tanto, dos los objetivos principales de este trabajo de tesis doctoral para el área de calidad del software que podemos enunciar de la siguiente manera:

1. *Proporcionar soporte a la construcción de modelos de calidad orientados a dominio a través de una metodología formal y sistemática que permita desarrollar modelos reutilizables y validados.*
2. *Obtención de un modelo de calidad que permita la evaluación y selección de productos para la protección de sistemas informáticos a través de la aplicación de la metodología definida.*

El modelo de calidad se particularizará para el dominio de los productos finales de seguridad y protección de sistemas informáticos, con el fin de que se convierta en un modelo de uso práctico, evitando así las generalidades necesarias del estándar. Es importante destacar que este proceso no implicará una pérdida de la "reutilización" del modelo de calidad ya que es éste uno de los principales objetivos del modelo. Dado que la definición de un modelo de calidad para un producto específico es una tarea compleja [Carvallo, Franch*, et al.* 2003], [Dromey 1996] y [Kitchenham y Pfleeger 1996] que requiere mucho tiempo, conocimiento de los productos que se están evaluando y experiencia con los mismos, se trata de proporcionar herramientas que ahorren tiempo en las evaluaciones de la calidad de productos incluidos en el dominio en cuestión, dejando a su vez el número suficiente de grados de libertad en el proceso para que este sea aplicable a todos los productos del dominio definido. De esta forma, se evitará que haya que construir el modelo de calidad desde el inicio cada vez que se desee realizar una evaluación de un producto incluido en el dominio particular.

Por otra parte, cabe destacar que se ha generalizado el proceso seguido para la adaptación del modelo de calidad al dominio de productos finales de seguridad de sistemas informáticos, con el fin de que este proceso sirva de guía para futuros ajustes del modelo de calidad utilizado (ISO/IEC 9126), ya sea para extender el propio dominio de productos de seguridad y protección de sistemas informáticos o para otro dominio cualquiera de productos software. Para la definición del proceso de construcción utilizado para la obtención del modelo de calidad se ha tenido en cuenta que debe ser un proceso formal y sistemático y permitir la validación del modelo obtenido. Además,

deberá basarse en los estándares internacionalmente reconocidos, analizar la documentación existente sobre el dominio y tener en cuenta los distintos puntos de vista de los participantes en el proceso, así como, permitir el consenso entre expertos en el dominio de aplicación para así tener en cuenta las distintas experiencias.

Para la consecución de estos objetivos fue necesario resolver los siguientes aspectos:

- Estudio de los procesos actuales de evaluación de calidad de productos finales para validar la posibilidad de integrar en ellos un modelo de calidad predefinido.
- Análisis de los modelos de calidad orientados a dominio existentes y de su método de construcción con el fin de encontrar uno que cumpla los requisitos necesarios para cumplir nuestro propósito, o bien en caso contrario, integrar todo lo aprendido en ellos en un nuevo proceso propio sistemático, formal y que permita obtener el modelo utilizando el criterio de expertos en las diferentes áreas involucradas.
- Validación empírica del modelo utilizando la opinión de expertos en el área y a través del tratamiento estadístico de los datos recogidos.

1.1.2 Principales aportaciones

Las aportaciones fundamentales de este trabajo de tesis son dos:

1. Creación de un proceso sistemático y formal de generación de modelos de calidad orientados a dominio que pueda ser aplicado a cualquier dominio de aplicación para la obtención de un modelo de calidad jerarquizado, priorizado y que incluya las interrelaciones entre los diferentes criterios. El proceso debe incluir la validación del modelo de calidad obtenido.
2. Modelo de calidad orientado al dominio de aplicación de los productos para la protección de sistemas informáticos. En dicho modelo se proporcionarán no sólo las características esenciales jerarquizadas de estos productos, sino también, se incluirán prioridades para cada una de ellas que podrán utilizarse, al menos, como base para la negociación de la importancia de cada una de las

características, sub-características y atributos del modelo aplicado en cada proyecto concreto. También se proporcionarán las relaciones entre los criterios de evaluación.

Como aportaciones significativas, además de las principales anteriormente definidas, cabe destacar:

- Un catálogo de atributos para el dominio de productos de seguridad que facilite al usuario la definición de los requisitos técnicos y organizativos (coste, soporte ofrecido por el proveedor, reputación, etc.) para el producto a evaluar obtenido a través de la aplicación de estándares y otras publicaciones relacionadas. En el caso de los atributos organizativos, éstos serán válidos para ser usados para otros dominios de aplicación como entrada para posteriores refinamientos con el fin de adaptarlos al dominio específico.
- Una aplicación web para la generación de cuestionarios que faciliten la recogida y posterior tratamiento de los datos de los expertos. Dicha aplicación será reutilizable con el fin de facilitar la definición de futuros modelos.
- Un estudio sobre una muestra de más de 200 expertos acerca de las características más relevantes de los productos de seguridad informática, así como, su punto de vista en relación con la calidad del software y la utilidad de modelos de calidad para la evaluación y selección de productos de seguridad.

Cabe destacar que estas aportaciones pueden servir como punto de partida para otros estudios y aplicaciones que requieran de la generación de un modelo de calidad orientado a dominio de aplicación

1.1.3 Trabajos previos a la redacción de este documento

Distintas motivaciones originaron el interés por la evaluación de productos COTS de seguridad informática. Por una parte, el interés por medir y evaluar distintos aspectos de los productos software se inició en el año 1999, cuando el Dr. Luis Fernández Sanz formó un grupo de investigación con este fin en la Universidad Europea de Madrid en el

que la autora colabora desde Octubre del 2002. Por otra parte la experiencia profesional de la autora en el área de arquitectura de sistemas evaluando la calidad de productos software finales para emitir informes con recomendaciones a la dirección, así como, el proyecto de consultoría con la empresa OVEC Publications para emitir informes de evaluación de productos de seguridad informática [Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2008] iniciado en enero de 2007, promovieron el surgimiento de esta línea de investigación.

De las primeras investigaciones relacionadas con esta área, surgió la primera publicación en 2004 titulada *"Use cases for enhancing IS requirements management"* [Fernández-Sanz, Lara*, et al.* 2004] relacionada con la recogida de requisitos funcionales de los usuarios. Le siguieron otras publicaciones relacionadas con distintos aspectos de la calidad del software como el rendimiento en "*Un estudio sobre rendimiento web*" [Villalba de Benito, Fernández Sanz*, et al.* 2004]. Posteriormente, se aplicaron diversos análisis estadísticos siguiendo el procedimiento desarrollado en este trabajo de tesis, fruto de los cuales, surgieron varias publicaciones. La primera de ellas como resultado del trabajo desarrollado en RePRIS (Red para la promoción y mejora de las Pruebas en ingeniería del software) durante el año 2008 titulada "*Factores que afectan negativamente a la aplicación práctica de las pruebas de software"* [Fernández-Sanz, Lara*, et al.* 2008]. Este trabajo fue posteriormente mejorado extendiendo, entre otras mejoras, el análisis estadístico. Como resultado de ello surgió la publicación *"Factors with negative influence on software testing practice in Spain: a survey"* [Fernández-Sanz, Villalba*, et al.* 2009].

Otras publicaciones relacionadas con seguridad informática [Villalba, Fernández-Sanz*, et al.* 2008].

1.3.Planteamiento del problema

1.1.4 Metodología

Las actividades llevadas a cabo para lograr los objetivos anteriormente descritos han sido las siguientes:

1. Localización y análisis de normas específicas sobre la calidad del software.
2. Estudio y evaluación del estado de la cuestión para determinar, por una parte, los procesos de evaluación y selección de productos COTS existentes, y por otro, los modelos de calidad orientados a dominio y los métodos de construcción de tales modelos.
3. Definición de un proceso propio de generación de modelos de calidad orientados a dominio según el análisis del estado de la cuestión.
4. Adaptación del modelo de calidad estándar al dominio de productos de seguridad informática.
5. Recopilación de un catálogo de criterios para productos COTS, revisión interna y adaptación al dominio de los productos de seguridad informática y al modelo de calidad previamente obtenido.
6. Construcción de una herramienta web para facilitar la recogida de datos de expertos y su posterior tratamiento estadístico.
7. Realización de pruebas piloto con alumnos de último curso y de máster. Adaptación de la herramienta según las conclusiones obtenidas en las pruebas piloto.
8. Validación del catálogo obtenido a través de la recogida de datos de expertos que seleccionan la importancia relativa de cada uno de los criterios recogidos y revisados y aportan nuevos criterios según su experiencia y conocimiento.
9. Tratamiento estadístico de los datos utilizando análisis multivariante y, en concreto, análisis de componentes principales, análisis factorial exploratorio y análisis factorial confirmatorio del modelo a través del uso de ecuaciones estructurales. Así a través de la aplicación de los diferentes análisis el modelo de calidad obtenido a través de la aplicación de estándares y publicaciones

relacionadas, así como, la propia experiencia en proyectos de evaluación software se va trasformando en un modelo de calidad preliminar y, por último, en el modelo final empíricamente validado.

10. Formulación de conclusiones y planteamiento de posibles trabajos futuros que permitirán ampliar el alcance del uso del proceso a otros dominios de aplicación.

1.1.5 Método de evaluación

A partir de la actividad número 5 de entre las descritas en la sección anterior (1.1.4) para conseguir el objetivo principal, se obtiene un modelo de calidad específico para el dominio de aplicación de los productos finales para la protección de la seguridad informática de los sistemas. Dicho modelo se obtiene a través del análisis de las normas y otros documentos publicados relacionados tanto con modelos de calidad, como con productos COTS y productos de seguridad informática. Para validar que dicho modelo se ajusta a la realidad práctica de las organizaciones actuales y que, por tanto, para verificar su aplicabilidad práctica, se llevan a cabo dos acciones:

1. Como primer paso el modelo se ha aplicado a tres casos reales de la industria [Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008] para verificar que los criterios obtenidos se pueden aplicar en la práctica. Las lecciones aprendidas de estas evaluaciones se resumen en la sección 3.2.1.
2. Por otra parte, a pesar de que otros investigadores no incluyen ningún tipo de validación de los modelos obtenidos salvo la aplicación de estos a casos de estudio (ver sección 2.4 para un análisis completo), se consideró adecuado validar el modelo obtenido tras la aplicación del análisis de normas y otra literatura relacionada y la propia experiencia de los investigadores. Para ello, se valida y consensua el modelo obtenido a través de la consulta a expertos en las diferentes áreas de conocimiento relacionadas tanto con la calidad del software como con la seguridad informática. Las lecciones aprendidas en los casos de estudio nos mostraban que el modelo obtenido tenía demasiados

criterios para ser eficiente. Por ello, los datos recogidos se utilizan no sólo validar el modelo sino también reducirlo eliminando así información redundante para obtener un modelo riguroso, práctico y eficiente. Para la eliminación de información redundante se utiliza análisis estadístico multivariante. En concreto se utilizan las técnicas de análisis factorial exploratorio para validar y corregir en caso necesario el modelo teórico, análisis de ecuaciones estructurales y análisis factorial confirmatorio para confirmar el modelo teórico y obtener y cuantificar, además, las relaciones entre las características, sub-características y atributos de dicho modelo.

1.4. Estructura de la tesis

Esta tesis se estructura en 5 capítulos cuyo contenido se describe brevemente a continuación.

- Capítulo 1. Introducción

  En este capítulo se realiza una introducción del problema, se presentan los objetivos y las principales aportaciones de la tesis, se plantea el problema y se explica la metodología utilizada para resolverlo y evaluar la solución propuesta, además de explicar el contenido y estructura de todo el documento.

- Capítulo 2. Estado del arte

  Se presenta un resumen del estado del arte en relación a los modelos de calidad orientados a dominios de aplicación específicos a través de una revisión sistemática de los trabajos existentes en el momento de realizar este trabajo. Además, se incluye una discusión sobre los métodos utilizados en la construcción de dichos modelos que servirá de base posteriormente para la construcción del proceso desarrollado en este trabajo. También se muestra el estado de la práctica en los procesos de selección de productos COTS.

- Capítulo 3. Metodología

  Se presentan las fases del método de investigación usado para la obtención de la metodología de desarrollo de modelos de calidad orientados a dominio presentada en esta tesis.

- Capítulo 4. Evaluación del modelo y validación

  En este capítulo se muestra tanto la validación de la aplicabilidad del proceso definido en el capítulo 3 a través de su aplicación a al dominio de productos COTS de seguridad informática, como la corrección y validación del modelo teórico.

- Capítulo 5. Conclusiones y líneas futuras de investigación

  En él se resumen las contribuciones de la tesis y se proponen los trabajos futuros para la mejora de los procesos de evaluación de productos de seguridad informática.

Capítulo 2. Estado del Arte

2.1.Definición de términos

2.1.1. Términos relacionados con la calidad del software

A lo largo del presente trabajo se utilizarán términos de las áreas de ingeniería del software y seguridad informática. Por tanto, empezaremos por definirlos para poder fijar el marco de trabajo.

En el ámbito de la ingeniería del software nos interesa, primero de todo, definir la calidad del software. En este sentido, el objetivo de la norma internacional *Quality management systems -- Fundamentals and vocabulary* [ISO 2005a] (o en su versión en español: *Sistemas de gestión de la calidad. Fundamentos y vocabulario* [AENOR 2005]) es fijar y tipificar el vocabulario relativo a la calidad. Así, define calidad de la siguiente manera:

"Grado en el que un conjunto de características inherentes[4] cumple con los requisitos"

donde *requisito* se define como la necesidad o expectativa establecida, generalmente implícita u obligatoria.

[4] "Inherente" en contraposición a "asignado", significa que existe en algo, especialmente como una característica permanente.

También en el estándar IEEE Std 1061-1998: *Standard for a software quality metrics methodology* [IEEE 1998], se define calidad del software como el grado en el que el software posee una combinación deseada de características previamente definidas.

La definición de característica y característica de calidad, también en la norma ISO 9000, son las siguientes:

- **Característica:** rasgo diferenciador. Una característica puede ser inherente o asignada. Una característica puede ser cualitativa o cuantitativa. Existen varias clases de características, tales como: físicas, sensoriales, de comportamiento, de tiempo, ergonómicas y funcionales.
- **Característica de calidad:** característica inherente de un producto, proceso o sistema relacionada con un requisito.

Para este trabajo nos interesan las características de los productos software. Un producto se define, según de nuevo la norma ISO 9000, como el resultado de un proceso, donde proceso es el conjunto de actividades mutuamente relacionadas o que interactúan, las cuales trasforman elementos de entrada en resultados. En nuestro caso, el proceso es el de desarrollo del software. Es importante, entonces, destacar que consideraremos los productos ya desarrollados (en la introducción denominados COTS) y, por tanto, no forma parte de este trabajo ningún resultado relacionado con el propio proceso de desarrollo.

En conclusión, dada la dificultad de la evaluación formal de la calidad del software, ésta se suele descomponer en características, sub-características y, así sucesivamente, hasta obtener atributos[5] que se puedan medir directamente o a través de métricas[6]. El grado en el que dichas características cumplan con los objetivos previamente definidos nos dará la calidad del producto. Dicha descomposición recibe el nombre de **modelo de calidad** (los modelos de calidad se tratarán en la siguiente sección). De esta forma, en [Fenton y Pfleeger

[5] En la Norma ISO/IEC 14598 [ISO 1999a] se define atributo como una propiedad física o abstracta medible de una entidad.

[6] En la Norma ISO/IEC 14598 se define métrica como el método y escala de medición definidos.

1997] se define la medición del Software como "*the process by which numbers or categories are assigned to attributes of (software) entities*".

Por último, en la norma 14598-1 [ISO 1999a] se define la **evaluación de calidad** como:

"El examen sistemático de hasta que punto una entidad es capaz de cumplir con requisitos especificados. Los requisitos pueden especificarse formalmente, caso de que un producto se desarrolla específicamente para un usuario bajo contrato, o especificados por la organización desarrolladora, caso de un producto desarrollado para usuarios no especificados, como el software de consumo, o los requisitos pueden ser más generales, como el caso de un usuario que evalúa productos con fines de comparación y selección."

En el caso que nos ocupa, comparación y selección de productos software dirigidos a usuarios no especificados, los requisitos se especificarán de forma general para un mismo tipo de producto, tal como sugiere la definición anterior, al ser las necesidades o requisitos de los usuarios similares para un mismo tipo de producto y las características generales de los propios productos comunes.

2.1.2. Definición de COTS

Entre las definiciones relacionadas con este tipo de productos encontramos la de paquete software que, en ISO/IEC 2382 [ISO 1993], se define como:

"Un juego de programas completo y documentado, suministrado a varios usuarios para una aplicación o función genérica".

Por otra parte, en ISO/IEC 12207 [ISO 1995] se define producto *"off-the-shelf"* (OTS) como:

"Product that is already developer and available; usable either "as is" or with modification".

Según las definiciones anteriores un producto genérico (paquete software o producto *"off-the-shelf"* también denominado "producto final" en el estándar ISO/IEC 14598-1 [ISO 1996]), es un conjunto de programas que se comercializan ya desarrollados y listos para ser

utilizados. Como estos productos son desarrollados en base a una serie de requisitos que los proveedores de software creen que encontrarán en un amplio número de clientes potenciales, muchos de ellos permiten modificaciones para adaptarse a las necesidades particulares del cliente. Otros, en cambio, no disponen de dicha funcionalidad con lo que el cliente dispondrá de una serie de características genéricas.

En los últimos años se ha extendido el uso de *productos COTS (Commercial Off-The-Shelf)*. En [Basili y Boehm 2001], por ejemplo, se caracterizan los productos COTS de la siguiente forma:

> *"COTS software has the following characteristics: (1) the buyer has no access to the source code, (2) the vendor controls its development, and (3) it has a nontrivial installed base (that is, more than one customer; more than a few copies). This definition does not include some kind of products like special purpose software, special version of commercial software, and open source software."*

Otros autores [Comella-Dorda, Dean*, et al.* 2002] definen productos COTS como:

> *"A COTS product is a product that is sold, leased, or licensed to the general public; offered by vendor trying to profit from it; supported and evolved by the vendor, who retains the intellectual property rights; available in multiple, identical copies; and used without modification of the internals"*

Según estas definiciones el término *"commercial"* hace referencia a que el proveedor de software vende, alquila o autoriza el uso de este software para obtener un beneficio. En la primera definición se menciona de forma explícita que el software de código abierto, comúnmente conocido como "software libre" u *"open source"* no estaría incluido en este tipo de productos. También en la última definición se excluye este tipo de productos de forma indirecta pues generalmente los modelos de negocio considerados en los mismos no están basados en la obtención de beneficio a través de la venta de licencias del software. Tan sólo en [Morisio y Torchiano 2002], se incluye el software libre como una nueva característica en el mercado de COTS y, por tanto, dentro de la definición de producto COTS.

Dado que no existe una definición consensuada de productos COTS, en este trabajo nos referiremos a producto empaquetado, final o COTS acogiéndonos a las definiciones de los citados estándares ISO por ser éstos internacionalmente aceptados. Dado que las definiciones dadas en los estándares son muy generales, las completaremos con la definición dada por Basili [Basili y Boehm 2001]. Por tanto, no consideraremos el software de código abierto, como producto COTS.

2.2.El proceso de evaluación de la calidad del software

En las secciones anteriores se ha definido la calidad del software como el grado en que el software posee un conjunto de características previamente definidas y, por lo tanto, el proceso de evaluación de la calidad del software estará basado en la medición de dichas características. Pero el proceso de evaluación es un proceso complejo en el que intervienen más actividades aparte de la ya citada definición del modelo de calidad y verificación del grado de cumplimiento de dichas características con relación a los requisitos previamente impuestos. En esta sección se resumirán los aspectos más importantes de los procesos de evaluación más utilizados en la actualidad, así como, su capacidad para integrar un modelo de calidad previamente definido.

2.2.1. Estandarización del proceso de evaluación del software: ISO/IEC 14598

Como se explica más en detalle en la sección 2.3.1, los primeros pasos hacia la evaluación del software fueron dados por McCall [McCall, Richards*, et al.* 1977] y Boehm [Boehm, Brown*, et al.* 1978] a través de la definición de sus respectivos modelos de calidad. Posteriormente y con ánimo de obtener un modelo de calidad estandarizado, surgió la norma ISO/IEC 9126-1 [ISO 2001a] que describe un modelo de calidad consensuado e internacionalmente aceptado.

Por otra parte, como ya se mencionó en la sección anterior, la serie de normas ISO/IEC 14598 proporcionan métodos para valoración y evaluación del producto software. Las

mismas normas se pueden se puede usar para: 1) evaluar productos existentes, o 2) para evaluar productos en desarrollo (en este caso, el proceso de evaluación debe sincronizarse con el proceso de desarrollo).

La serie de normas ISO/IEC 14598 consta de seis partes. Las normas ISO/IEC 14598-2 [ISO 2000a] y 14598-6 [ISO 2001b] hacen referencia a la gestión y soporte de la evaluación mientras que las normas ISO/IEC 14598-3 [ISO 2000b], ISO/IEC 14598-4[ISO 1999b], ISO/IEC 14598-5[ISO 1998b] proporcionan requisitos y directrices para la evaluación desde el punto de vista del desarrollador, comprador y evaluador independiente respectivamente.

En la Figura 2 pueden verse las partes de las que consta esta serie y su relación con el proceso de evaluación.

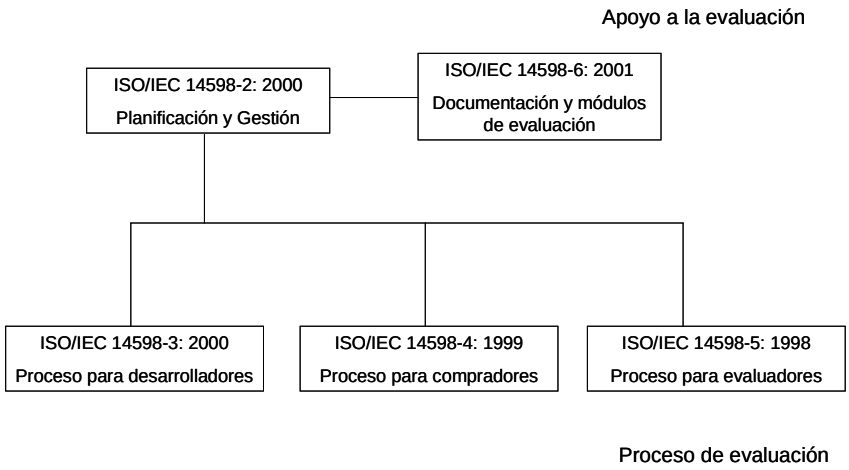


Figura 2. Relación entre las normas del ISO/IEC 14598 [ISO 1999a]

De esta forma, queda establecida la base para la definición de los requisitos de calidad para el desarrollo del software en la norma ISO/IEC 9126 a la que completa la serie ISO/IEC 14598 centrada en la evaluación de los productos software. En resumen, podemos decir que la serie de normas ISO/IEC 9126 [ISO 2001c] define un modelo de calidad genérico y proporciona ejemplos de métricas, mientras que, por su parte, la serie de normas

ISO/IEC 14598 presentan una visión general de los procesos de evaluación del producto software y proporcionan directrices y requisitos para la evaluación.

La norma ISO/IEC 14598-5 [ISO 1998b] describe el proceso a seguir para la obtención de un informe de evaluación por parte de evaluadores que lleven a cabo evaluaciones independientes. Además, puede usarse para verificar que se cumplen los requisitos de calidad para productos COTS enunciados en la norma ISO/IEC 25051 [ISO 2005g].

El proceso de evaluación definido en esta norma puede usarse para evaluar tanto productos en desarrollo como productos finales, de ahí que sea demasiado general para servir de forma práctica a nuestro propósito.

Según la norma, la evaluación comienza cuando el solicitante de la evaluación pide al evaluador la realización de una evaluación de un producto software. El solicitante debe expresar al evaluador los requisitos de la evaluación y ambos deben acordar a partir de éstos la especificación de la evaluación. Como datos de entrada al proceso de evaluación el evaluador dispone de la descripción y los componentes del producto. Tras el proceso de evaluación, el solicitante de la evaluación obtendrá junto con el informe de evaluación, los registros de evaluación incluyendo planificación y registros de las acciones de evaluación realizadas. En la Figura 3 puede observarse un resumen del proceso de evaluación definido en el estándar en el que además se identifica la información de flujo entre las distintas actividades. Como muestra figura, el proceso se divide en 5 fases que van desde la definición de los requisitos y objetivos de la evaluación, la especificación de las medidas que se realizarán o el diseño de la evaluación para documentar y planificar las acciones a llevar a cabo para la evaluación, hasta la ejecución de la misma y obtención de las conclusiones asociadas. En cuanto a las salidas al proceso, como se observa en la Figura 3, hay productos intermedios y productos finales. Entre los primeros se encuentran los documentos de requisitos, especificación y plan de la evaluación; entre los segundos los registros e informes de evaluación.

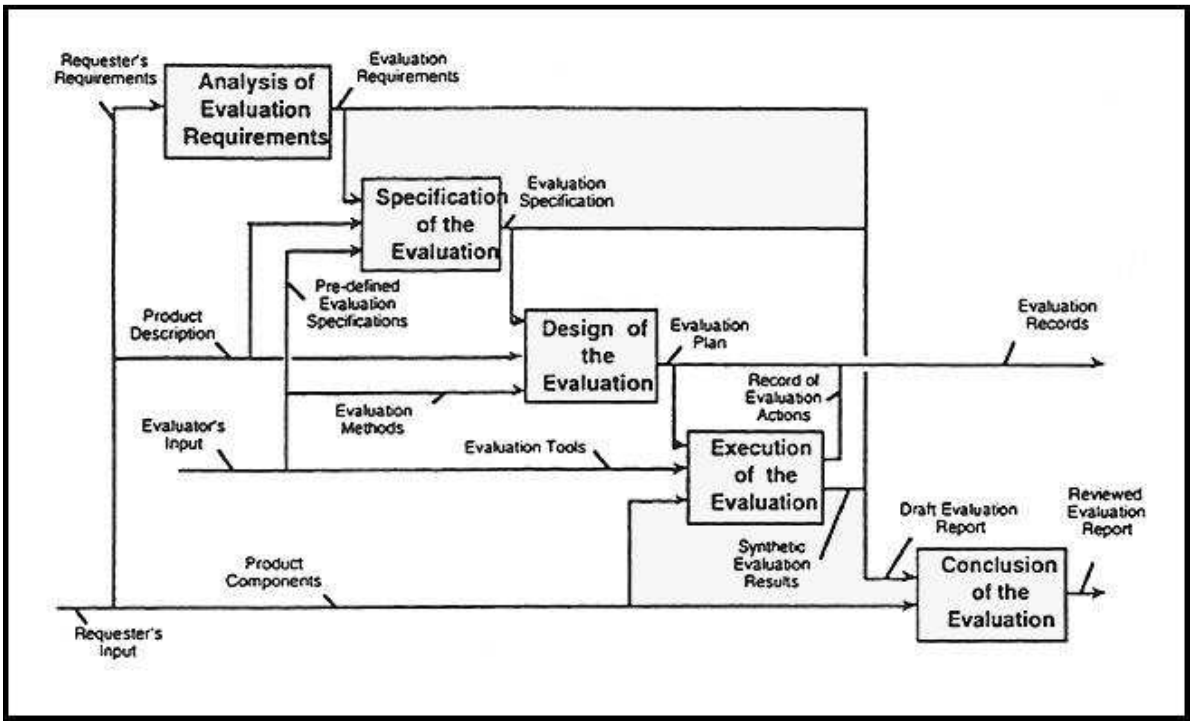


Figura 3. El proceso de evaluación para evaluadores [ISO 1998b]

El principal problema del ISO 14598-5 es la falta de soporte necesario para su aplicación práctica. Sin duda la norma es una valiosa base teórica para la definición de un proceso de evaluación de la calidad del software pero, como todo estándar, su necesaria abstracción y generalidad para poder ser aplicado en cualquier situación, hace que haya que adaptarlo al objetivo particular para obtener un proceso operativo. Este marco genérico permite la integración de modelos de calidad predefinidos (en la fase de establecimiento de los requisitos de evaluación). Sin embargo, no prescriben ni recomiendan metodologías, métodos ni procedimientos específicos para realizar las actividades sino que representan un marco conceptual (y normativo) genérico, esto es, un modelo en donde distintos métodos, técnicas, procedimientos y herramientas se puedan aplicar. Por ejemplo, el estándar está definido para cualquier tipo de evaluación tanto de productos en desarrollo como de productos finales. Por tanto, será necesario personalizarlo para cada caso concreto de evaluación.

Además existen otros problemas en relación a este estándar:

1. El estándar no relaciona adecuadamente las fases de establecimiento de requisitos y especificación de la evaluación. Más concretamente, el solicitante de la evaluación define durante el establecimiento de los requisitos de evaluación el grado de cobertura o rigor de la evaluación. De éste dependerá directamente la técnica de medición que se defina durante la especificación de la evaluación. Sin embargo, el estándar ni siquiera menciona esta relación y mucho menos la necesidad de revisar los requisitos de evaluación si el solicitante no aprobase las técnicas de medición definidas en relación al grado de cobertura de los requisitos.

2. Cada fase requiere la aprobación por parte del solicitante lo que ralentiza el proceso.

3. Aunque establece que los requisitos definidos en ISO/IEC 25051 [ISO 2005g] pueden verificarse utilizando el proceso de evaluación definido en la norma, no especifica cómo integrar la información de ambas normas en un mismo proceso. Es decir, si además de querer verificar la adecuación del sistema a los requisitos y/o evaluar la calidad de la documentación del producto, se pretende evaluar la calidad del producto con respecto a las características del modelo de calidad establecido, en la norma no se especifica cómo integrar todo el proceso en uno único.

4. El acuerdo entre solicitante de la evaluación y el evaluador se lleva a cabo en la primera fase. Por tanto, aún no se ha analizado si se dispone de toda la documentación necesaria. Esto implica que podría ser necesario tener que modificar los requisitos de la evaluación, por ejemplo el nivel de profundidad requerido por el solicitante, por falta de la documentación necesaria para llevar a cabo la evaluación acordada anteriormente. Tampoco se han especificado ni aprobado las medidas que se van a llevar a cabo sobre el producto, por lo que en el acuerdo con el cliente no se podrían especificar éstas. Por tanto, puede ocurrir que al definir dichas medidas sea necesario modificar el acuerdo o, en el peor de los casos, surja un conflicto a la entrega de la evaluación por falta de acuerdo entre lo solicitado y lo entregado.

5. La norma hace referencia al nivel de profundidad en la evaluación pero sólo define en el anexo B a nivel informativo cuatro niveles por orden de exhaustividad y rigor en la evaluación. A mayor nivel de profundidad mayor será el rigor de las técnicas de evaluación utilizadas teniendo en cuenta el tiempo y esfuerzo requerido para aplicarlas. Estos niveles de profundidad sólo se asocian al riesgo (económico, de seguridad, daño físico, etc.) del producto software objeto de evaluación. Sin embargo, aunque sin duda el riesgo del producto es una importante razón para realizar una evaluación más exhaustiva, ésta no es la única. Esta definición de nivel de evaluación según el nivel de riesgo al que va dirigido el producto, está muy enfocada a la selección de un producto para un entorno concreto. Sin embargo, existen evaluaciones de productos en las que el entorno de destino es desconocido. Por ejemplo, las evaluaciones que se realizan en revistas técnicas especializadas, las que solicita un fabricante para verificar su nuevo producto antes de que éste salga al mercado o para mejorar uno existente o las evaluaciones independientes que solicita con motivos de marketing.

2.2.2. Metodologías de evaluación de la calidad del software

En la actualidad existen numerosas metodologías de evaluación de la calidad del software. En esta sección nos centraremos en las metodologías de evaluación de la calidad del software basadas en el ISO 14598 y que, aun no siendo específicas de productos COTS, son aplicables a estos productos software. Dejaremos, por tanto, a un lado las metodologías que se centran en el proceso de desarrollo del software por no estar relacionadas con este trabajo.

2.2.2.1 El método SPACE (*Software Product Advanced Certification and Evaluation*)

El método SPACE fue desarrollado como parte del proyecto europeo SPACE-UFO (*Software Product Advanced Certification and Evaluation - User Focus*)[7] y amplia la norma ISO 14598 fundamentalmente a través del desarrollo del concepto de perfil de calidad. El perfil de calidad presenta las características de calidad relevantes y los niveles de evaluación para el producto software. De esa forma, el perfil refleja la noción de calidad para cierto producto software y hace la calidad tangible para usuarios y desarrolladores [Punter, Solingen*, et al.* 1997]. La información del perfil de calidad se basa en la información sobre el cliente o usuario, el proceso de negocio y el propio producto software [Trienekens, Veenendaal*, et al.* 1997]. En la Figura 4 puede verse un ejemplo de perfil de calidad según niveles de evaluación nombrados A, B, C y D.

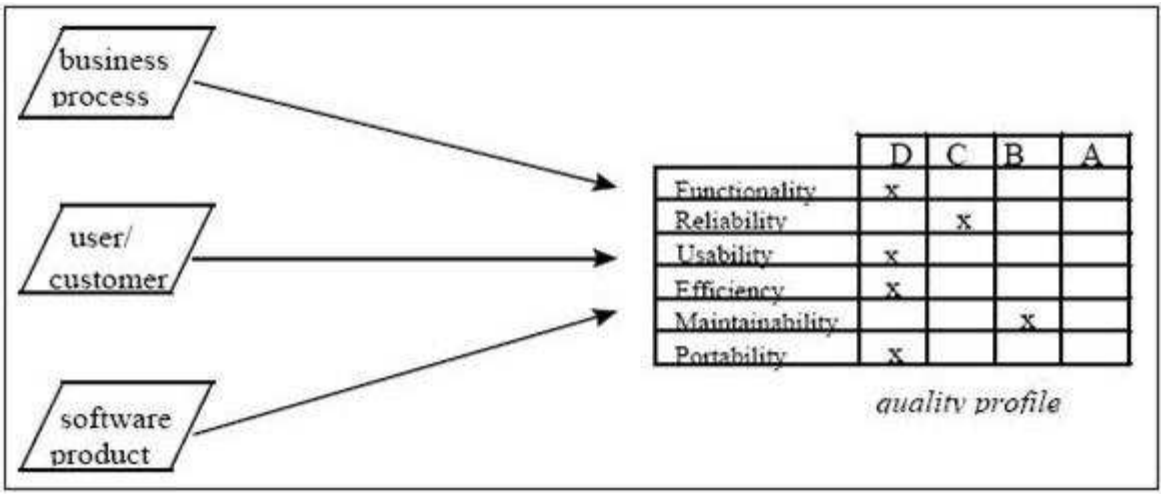


Figura 4. Proceso de obtención de un perfil de calidad [Punter, Solingen 1997]

Además, el método SPACE también desarrolla el concepto de nivel de evaluación ya mencionado (anexo informativo) en el ISO 14598 [ISO 1998b]. El nivel de evaluación hace

[7] Información del proyecto SPACE-UFO puede encontrarse en http://www.cse.dcu.ie/essiscope/sm2/atwork/spaceufo.html

referencia a la profundidad o rigor con que se llevará a cabo la evaluación y, por tanto, la técnica de evaluación utilizada dependerá de del nivel definido en la fase de establecimiento de requisitos. Diferentes niveles de evaluación proporcionan diferentes niveles de confianza en la evaluación de la calidad de un producto software. La metodología SPACE utiliza los cuatro niveles definidos en el proyecto Scope [Robert 1994] clasificados como D, C, B, y A por orden creciente de riesgo. Por tanto, productos con diferentes niveles de riesgo no serán evaluados con el mismo rigor. El nivel de riesgo puede estar relacionado con diferentes consecuencias derivadas de un fallo del software como pueden ser el riesgo económico de la organización, riesgo de daño físico al entorno o a las personas o riesgos relacionados con la falta de protección de los sistemas informáticos. Como puede apreciarse en la Tabla 1, para cada uno de estos niveles define una técnica de evaluación a utilizar dependiente de las características de calidad establecidas en la norma ISO/IEC 9126 [ISO 2001a]. De esta forma, una vez definido el modelo de calidad se asignarán, de acuerdo con el solicitante de la evaluación, niveles de evaluación a cada característica obteniendo como resultado el perfil de calidad.

	Level D	Level C	Level B	Level A
Functionality	functional testing	review (checklists)	component testing	formal proof
Reliability	programming language facilities	fault tolerance analysis	reliability growth model	formal proof
Usability	user interface inspection	conformity to interface standards	laboratory testing	user mental model
Efficiency	execution time measurement	benchmark testing	algorithmic complexity	performance profiling analysis
Maintainability	inspection of documents (checklists)	static analysis	analysis of development process	traceability evaluation
Portability	analysis of installation	conformity to programming rules	environment constraints evaluation	program design evaluation

Tabla 1. Técnicas de evaluación para varios niveles y características de calidad [Robert 1994]

La metodología SPACE introduce dos actividades o trasformaciones intermedias al proceso definido en el ISO 14598 en relación con la construcción del perfil de calidad:

1. Obtención del perfil de calidad a partir de los procesos de negocio, requisitos de usuario y el producto software (trasformación 1 o T1 en la Figura 5).

2. Definición de la especificación de las medidas de calidad y el plan de evaluación a partir del perfil de calidad(trasformación 2 o T2 en la Figura 5)

En la Figura 5 puede verse el proceso completo. Obsérvese que la metodología es válida tanto para el desarrollo de productos como para la evaluación de un producto ya desarrollado.

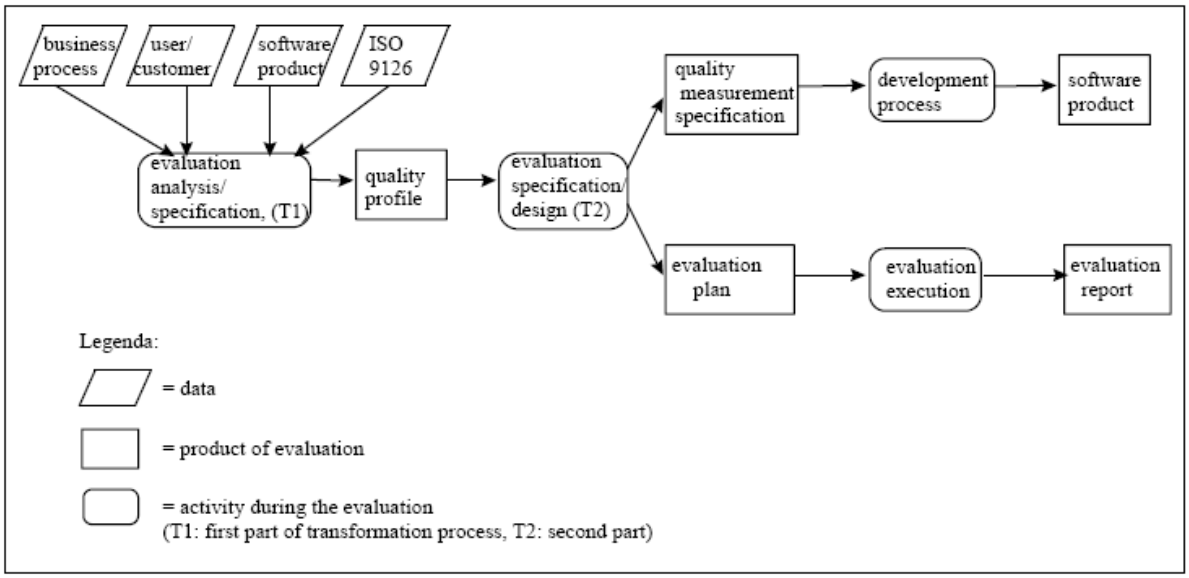


Figura 5. Metodología SPACE [Punter, Solingen 1997]

Esta propuesta aporta una idea muy importante para esta tesis: los perfiles de calidad. Como se mencionó anteriormente los perfiles de calidad el perfil reflejan la noción de calidad para cierto producto software. Con los perfiles de calidad se adaptan las características de calidad a cierto producto o tipo de productos (en el caso de este trabajo de tesis los productos de seguridad informática) de forma que puedan definirse de forma más exacta y no sea necesario en el momento de la evaluaicón definir las características de calidad a evaluar desde cero. La existencia de estos perfiles de calidad predefinidos puede hacer la evaluación más eficiente al ahorrar mucho tiempo en la obtención del modelo de calidad y, además, eficaz pues si el perfil está correctamente definido y validado los atributos serán medibles. Además, en este trabajo ya se menciona la necesidad de que el

perfil de calidad se base en la información sobre el cliente o usuario, el proceso de negocio y el propio producto software. Sin embargo, el método SPACE requiere de la existencia de una técnica de evaluación que pueda combinarse con las características de calidad que se han definido en el perfil. Esto no siempre es posible en la práctica ya que generalmente es necesario personalizar al caso concreto la técnica de evaluación utilizada. Desde luego, sí es necesario conocer la importancia relativa de cada criterio para poder asignar la técnica de evaluación adecuada y este es uno de los objetivos de esta tesis.

Por último, este proceso permite la integración con modelos de calidad predefinidos como el desarrollado en este trabajo de tesis, a través del concepto de perfil de calidad.

2.2.2.2 El Proceso W (*W-Process*)

El Proceso W [Punter, Kusters*, et al.* 2004] describe un método para dirigir el proceso de evaluación basado en el principio GQM [Basili y Weiss 1984] descrito en la sección 2.3.1 para reestructurar el proceso de evaluación de la norma ISO 14598. Obtendríamos así 3 niveles: el nivel de objetivo (*Goal)*, en el que se formulan los requisitos, el nivel Pregunta (*Question)*, en el que los requisitos se convierten en las preguntas que se necesitan contestar para obtener el objetivo de la evaluación; y el nivel Métrica (*Metric)*, en el que se refinan las preguntas en un conjunto de métricas asociadas. En Figura 6 se describe el proceso resultado.

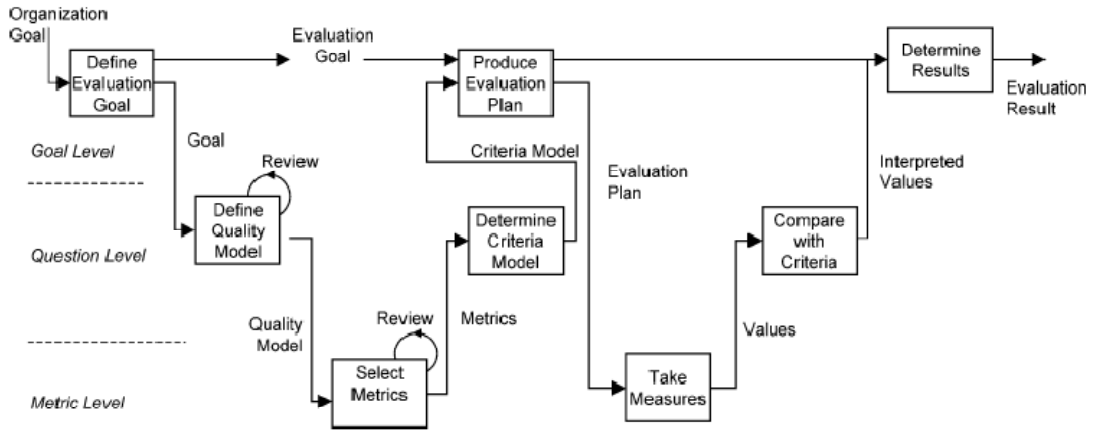


Figura 6. El Proceso W [Punter, Kusters 2004]

La forma en W que se observa en la Figura 6 da nombre al proceso y proviene de la unión de dos movimientos llamados: Definición V (*Definition-V)*, que comienza con la actividad "Definir el objetivo de la evaluación" y termina con "Producir el plan de evaluación"; y Ejecución V (*Execution-V)* que comienza con la actividad "Producir el plan de evaluación" y termina con "Determinar los resultados".

Además de implementar el proceso de evaluación definido en ISO 14598 a través de GQM, [Punter, Kusters 2004] detectan una serie de carencias en la definición del estándar. A continuación se enumeran dichas carencias y las soluciones propuestas integradas en el Proceso W:

1. Insuficiente información en la definición de requisitos. El estándar no proporciona información sobre cómo involucrar a las partes interesadas en la formulación de requisitos. A menudo en el proceso de evaluación existen distintas partes interesadas. Un ejemplo sería el de una organización que evalúa uno o varios productos para la adquisición de uno nuevo. En este caso, el usuario final, el director de proyecto y el director de Tecnologías de la información muy probablemente tendrán diferentes requisitos que incluso pueden entrar en conflicto unos con otros y, por tanto, es necesario llegar a un acuerdo entre las partes para obtener la definición de los requisitos finales. Para tratar este problema, [Punter, Kusters 2004] añade cuatro actividades más a la fase de "Establecimiento del propósito de la evaluación" de la norma ISO 14598:

    - Identificar los objetivos de negocio. Los objetivos de la evaluación deberían obtenerse a partir de los objetivos de negocio (aquellos que definen lo que la organización quiere conseguir) para asegurarse de que la evaluación tiene sentido y que se utilizará su resultado.

    - Identificar e involucrar a las partes interesadas. Es importante asegurarse que en la formulación de los objetivos se consulta a todas las partes involucradas y que haya tanto acuerdo entre las partes como sea posible.

    - Definir el objetivo de evaluación de acuerdo a una plantilla de objetivos de medida. De esta forma, se asegura que la evaluación sea tan explícita como

sea posible y se facilita la definición de los objetivos de evaluación. La plantilla utilizada por el Proceso W es la del paradigma GQM.

- Priorizar los objetivos de evaluación. Es importante cuando hay muchos objetivos definidos en la evaluación y/o cuando el tiempo y esfuerzo disponible para llevar a cabo la misma es reducido.

2. Falta de soporte para resolver la relación explícita entre los recursos (tanto personas como técnicas) y los objetivos de la evaluación relacionados con el coste, el tiempo y el esfuerzo del proceso. Por ejemplo, si uno de los principales objetivos de la evaluación es minimizar el esfuerzo dedicado a la misma, entonces será necesario aplicar la técnica de evaluación que requiera menos esfuerzo, por ejemplo, listas de verificación (*checklist)* en lugar de métricas. Para resolverlo, tal como se muestra en la Figura 7, el Proceso W propone realizar un equilibrado (*balancing)* entre las actividades de definición de los objetivos de evaluación y el modelo de calidad y, también, entre este último y la definición de métricas.

3. Insuficiente información para gestionar la retroalimentación. Aunque el ISO 14598 requiere que cada fase del proyecto sea aprobada por el solicitante del mismo, no da soporte sobre qué hacer en caso de que no haya aprobación. El Proceso W añade al modelo relaciones entre las actividades que permitan retroceder en las fases para solucionar la falta de satisfacción del solicitante. Así, como puede observarse en la Figura 7, una vez definido el modelo de calidad, si no hay acuerdo será necesario volver a los requisitos para redefinirlos según las discrepancias halladas en el modelo de calidad. También al definir las métricas y al obtener el plan de evaluación es necesario verificar con el solicitante de la evaluación que los resultados cubren sus expectativas para evitar que sea al final del proceso con la consecuente pérdida de tiempo.

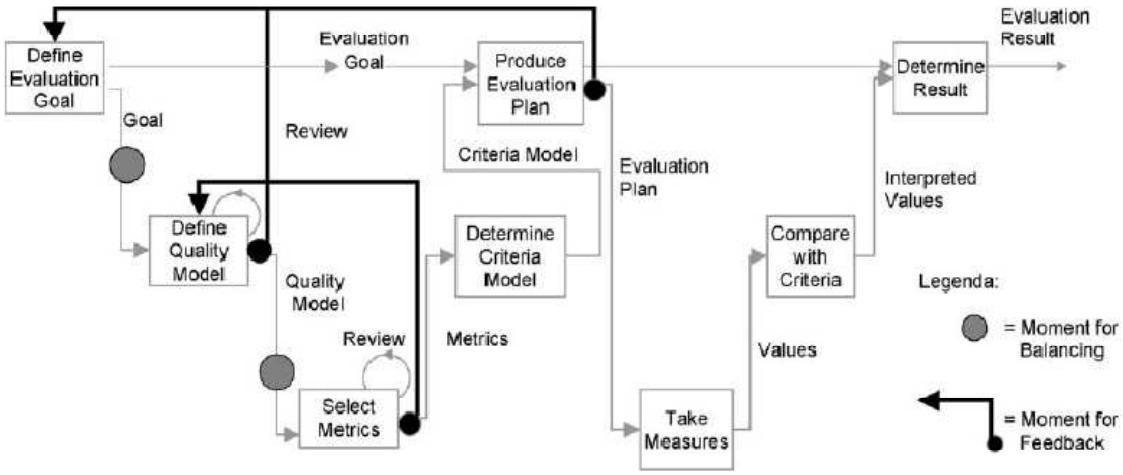


Figura 7. Procesos de retroalimentación y equilibrado de fases en el Proceso W [Punter, Kusters 2004]

Este proceso introduce cuatro ideas muy importantes para este trabajo de tesis:

1. La necesidad de equilibrar objetivos y recursos. Para ello, introduce la necesidad de involucrar a todas las partes interesadas. En nuestro caso, resolveremos esta carencia del estándar ISO 14598 teniendo en cuenta la opinión de expertos en los diferentes ámbitos relacionados tanto con el proceso de evaluación del software (ingenieros del software, desarrolladores) como de la toma de decisiones (dirección tecnológica) y, por supuesto, en el dominio de aplicación (expertos en seguridad).

2. La necesidad de priorizar los criterios de evaluación a medir con el fin de obtener un modelo eficiente y práctico en el entorno industrial actual. Existen muchos catálogos de atributos sin priorizar. Por tanto, uno de los objetivos de este trabajo es obtener la importancia relativa de unos criterios frente a otros.

3. De nuevo, y al igual que en el trabajo anteriormente citado, se insiste en que la técnica de evaluación o métrica deberá estar de acuerdo con los requisitos de tiempo y coste de la evaluación. Para resolver esta carencia, en nuestro caso las

métricas no se predefinirán, es decir, la técnica de evaluación para medir los criterios definidos y priorizados en el modelo se seleccionará en el momento de la evaluación. Para facilitar esta labor, sería recomendable almacenar las métricas utilizadas en cada proyecto de evaluación con el fin de poder ser reutilizadas.

4. En relación a la definición del modelo de calidad cuando existen requisitos de usuario, este trabajo define la obtención de retroalimentación del cliente de la evaluación tras la definición de métricas, de modo que si no hubiese acuerdo entre las partes, se volvería hacia atrás para modificar el modelo de calidad de acuerdo a la información de retroalimentación recogida. De esta forma, se evita descubrir el desacuerdo una vez ejecutada la medición de los criterios, permitiendo así corregir errores con antelación.

Por último, destacar que el proceso W permite el uso de modelos de calidad definidos a priori a través de la fase de "Definición del modelo de calidad".

2.2.2.3 El método MEDE-PROS

El método MEDE-PROS [Colombo y Cervigni 2002] tiene como objetivo proveer a los evaluadores de mecanismos y recomendaciones para la evaluación de productos COTS desde el punto de vista del usuario final. El proceso de evaluación está basado en la norma ISO/IEC 14598-5 [ISO 1998b] y el modelo de calidad se apoya es ISO/IEC 9126 [ISO 2001a] e ISO/IEC 12119 [ISO 1994]. En la Figura 8 puede observarse el modelo de calidad definido por MEDE-PROS.

El proceso de evaluación consiste en simular la operación normal del software, empezando por el análisis de la documentación, instalación del software tal como se indica en la documentación y, por último, el uso del producto. A lo largo del proceso el evaluador asigna calificaciones al producto en función de las preguntas de listas de verificación (*Checklist*). Además, registra el tiempo consumido en la evaluación y comentarios relevantes sobre la misma. Finalmente se elabora un informe de la evaluación que contiene los principales aspectos positivos del software y también sugerencias de mejora.

Las herramientas para la aplicación de MEDE-PROS se almacenan en una base de datos denominada "Base de datos de Evaluación" (*Evaluation DataBase*)[Martinez, Azevedo*, et al.* 1999]

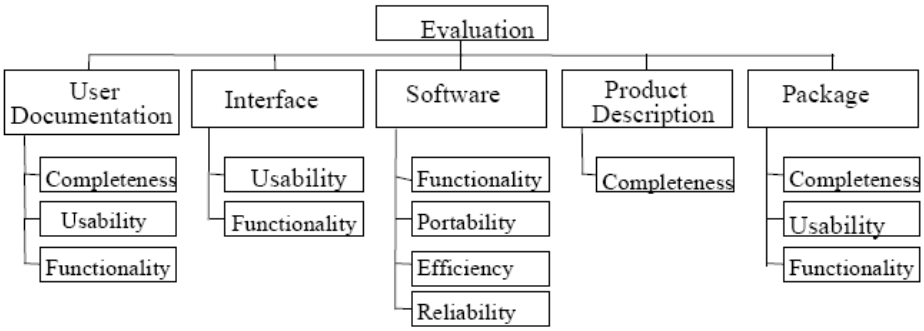


Figura 8. Modelo de calidad propuesto en el método MEDE-PROS

La ventaja de este método es que utiliza todos los estándares actuales sobre evaluación de calidad del software en un solo proceso integrándolos a través del modelo de calidad definido. Sin embargo:

1. No optimiza el proceso de evaluación sino que lo aplica directamente. Esto implica que, entre otras cosas, el proceso resultante sea sólo aplicable en la práctica debido al uso de listas de verificación (540 preguntas en la última versión informada [Colombo y Cervigni 2002]) ya que, en otro caso, sería inviable. Las listas de verificación utilizadas son de tipo nominal con respuestas del tipo "Sí", "No" o "No aplica". El resultado de evaluar el software utilizando sólo este mecanismo es que los resultados obtenidos dan muy poca confianza en la evaluación, lo cual, es válido sólo para ciertas circunstancias no permitiendo adaptar el nivel de profundidad de la evaluación a los requisitos de la misma.

2. El modelo de calidad utilizado (Figura 8) no contempla la evaluación de características no técnicas (por ejemplo, coste de adquisición, tipo de licencia o soporte proporcionado) necesarias en la evaluación de productos COTS como se muestra en la siguiente sección (2.2.3.1).

2.2.3. Metodologías de evaluación de la calidad de productos COTS

En esta sección nos centramos en las metodologías específicas para evaluación o selección de productos COTS. El objetivo es doble: por una parte, conocer las características específicas de estos procesos para poder aplicarlas a nuestros procesos de evaluación; por otra parte, averiguar si los procesos actuales permiten la integración de un modelo de calidad predefinido para conocer así la aplicabilidad del modelo.

2.2.3.1 El método OTSO (*Off-The-Shelf Option*)

El método OTSO (Off-The-Shelf Option) [Kontio, Caldiera*, et al.* 1996] establece un método de búsqueda, evaluación y selección de productos OTS (Off-The-Shelf)[8] dirigido por requisitos. El método proporciona técnicas específicas para la definición de criterios de evaluación, análisis coste-beneficio para productos alternativos y toma de decisiones a partir de los resultados obtenidos en la evaluación.

El proceso de definición de criterios de OTSO descompone los requisitos en un conjunto jerárquico categorizado en cuatro grupos:

- Requisitos funcionales;
- Características de calidad;
- Características relacionadas con el negocio tales como coste, estabilidad del fabricante, etc.;
- Arquitectura del producto.

Las principales características del método OTSO son:

- Proceso sistemático para la obtención de criterios de evaluación a partir de los requisitos del sistema;

[8] "OTS" hace referencia tanto a productos comerciales (COTS) como productos desarrollados a medida.

- Modelo para estimar el esfuerzo relativo o coste-beneficio de diferentes alternativas;

- Método para comparar los aspectos no financieros de diferentes alternativas, incluyendo situaciones que involucren criterios múltiples.

A continuación se resumen las fases del método:

1. Definición de criterios de evaluación. Se obtienen los requisitos funcionales y las características técnicas de calidad, de negocio y de arquitectura a través de un proceso de descomposición jerárquica basado en GQM [Basili y Weiss 1984].

2. Búsqueda (*Search)*. Durante esta fase se intentan identificar y encontrar todos los candidatos potenciales. Para ello, es necesario haber definido en la fase anterior los criterios principales de búsqueda que, generalmente, están basados en los principales requisitos funcionales y algunas restricciones clave (por ejemplo, la tecnología a utilizar o el coste del producto). Por tanto, para poder comenzar esta fase es suficiente con que las características de alto nivel, es decir, sin refinamientos, hayan sido obtenidas en la fase anterior.

3. Reconocimiento o rastreo (*Screening)*. Dado que la evaluación y análisis de todas las alternativas halladas en la fase anterior no sería práctico por el tiempo necesario para ello, es necesario decidir cuales de los productos COTS serán seleccionados para un análisis más detallado.

4. Evaluación (*Evaluation)*. El objetivo de esta fase es evaluar las alternativas según los criterios de evaluación y documentar los resultados. Por tanto, antes de comenzar esta fase, es necesario que se hayan refinado y formalizado los criterios de evaluación.

5. Análisis (*Analysis)*. El objetivo de esta fase es decidir qué alternativa es la mejor. Para ello, deberán ponderarse los criterios utilizados en la fase, de forma que pueda realizarse la toma de decisiones basándose en los resultados de la

evaluación. El método OTSO se apoya en el uso del método Analytic Hierarchy Process (AHP) [Saaty 1990] para este propósito.

6. Implementación (*Deployment)*. El producto seleccionado es implementado.

7. Valoración (*assessment)*. Finalmente, se valora el grado de éxito del producto seleccionado con el objetivo de mejorar el proceso de selección y proporcionar retroalimentación.

En la Figura 9 se pueden observar las seis fases de las que consta el método. Tal como indica el gráfico en la fase de búsqueda el número de alternativas puede crecer rápidamente y, a través de la fase de reconocimiento, disminuirá dicho número. De esta forma, se incrementa la eficiencia del proceso de selección al evaluar y analizar tan sólo aquellos productos que pasaron el filtro inicial (menos costoso).

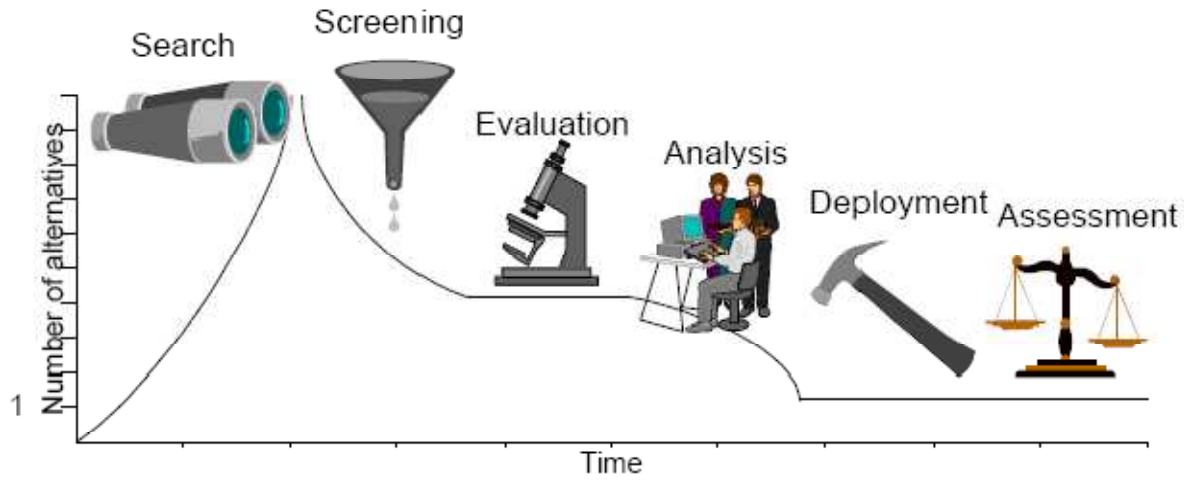


Figura 9. Fases principales del método OTSO [Kontio 1995]

Este método incluye una aportación muy valiosa para este trabajo de tesis: introduce la utilización de características no técnicas en la evaluación de productos COTS tales como el coste del producto, el soporte del proveedor, el tipo de licencia, etc. Tal como hemos comprobado a posterior, a través de numerosos trabajos, como por ejemplo [Kontio 1996], [Kontio, Caldiera 1996], [Burgués, Franch*, et al.* 2000], [Kunda y Brooks 2000], [Lawlis,

Mark*, et al.* 2001], [Ochs, Pfahl 2001], [Morisio y Torchiano 2002], [Comella-Dorda, Dean*, et al.* 2004] ,[Carvallo, Franch*, et al.* 2007], [Carvallo y Franch 2006], [Carvallo, Franch*, et al.* 2006], se ha demostrado la necesidad de incluir este tipo de características en la evaluación de productos COTS por la importancia que éstas tienen en la selección de un producto frente a otros.

Sin embargo, esta metodología mantiene una serie de puntos débiles que dificultan su implantación en el entorno actual:

1. Aunque el proceso de reconocimiento disminuye el número de alternativas a evaluar, es un proceso de tipo "fuerza bruta". A pesar de afirmar que las características técnicas se repiten para productos del mismo dominio, el método OTSO define desde cero el modelo de calidad para cada proyecto de evaluación. El tiempo consumido en la definición de requisitos es alto (en el caso de estudio reportado en [Kontio 1996] se informa de que un 28% del tiempo total se utilizó en la definición de requisitos). Por tanto, el esfuerzo requerido para ser aplicado en entornos industriales es muy alto, tal como otros autores han informado anteriormente [Ochs, Pfahl 2001]. Este proceso podría, por tanto, beneficiarse del uso de modelos de calidad orientados a dominio como el definido en este trabajo de tesis.

2. El proceso no tiene en cuenta la experiencia y conocimiento de expertos tanto en el dominio de aplicación en cuestión del producto a evaluar como en ingeniería del software, dirección, etc. Por tanto, la selección del producto en este tipo de procesos, se halla limitada por las experiencias acumuladas de los participantes en el proceso.

3. El proceso de evaluación está totalmente enfocado a la selección de productos dirigida por requisitos, con lo cual, sólo es válido cuando:

    a. el objetivo es seleccionar un producto,

    b. tenemos varios productos que comparar y

    c. tenemos requisitos de usuario como dato de entrada.

Esto claramente no se ajusta a todos los posibles objetivos de evaluación descritos en la sección 1.1 lo que limita su aplicabilidad.

4. El proceso no se apoya en ninguno de los estándares internacionalmente reconocidos. El modelo de calidad definido se modifica con respecto al del ISO 9126 sin justificación y el proceso de selección tampoco se apoya en el definido en el ISO 14598.

2.2.3.2 El método STACE (*Social Technical Approach to COTS software Evaluation*)

El método STACE [Kunda 2003] es un proceso de selección de productos COTS basado en los siguientes principios:

1. Alcance sistemático para la evaluación y selección de productos COTS.

2. Soporte a la evaluación de productos COTS y a la tecnología subyacente.

3. Uso de métodos "socio-técnicos" para mejorar el proceso de selección de productos COTS. Los criterios socio-técnicos incluyen:

   - Factores tecnológicos. Se refieren a las tecnologías que el cliente quiere utilizar.

   - Características funcionales. Tal como se definen en el estándar ISO/IC 9126. Estas características deberían ayudar en la selección inicial de alternativas.

   - Características de calidad del producto. Al igual que las características funcionales, estas hacen referencia a las definidas en el estándar ISO/IEC 9126. Estas características no es necesario cambiarlas de un proyecto de evaluación a otro sino tan sólo ajustarlas a los requisitos del cliente.

   - Factores socio-económicos. Factores no técnicos que deberían incluirse en la evaluación y selección de COTS tales como el coste del producto, la eficiencia del proveedor, el gasto en formación, etc. Los factores socio-económicos los clasifica en:

- Cuestiones relacionadas con el negocio. Por ejemplo, coste de adaptación e integración, tipo de licencias o coste de mantenimiento, soporte o formación.
- Capacidad del cliente. Por ejemplo, expectativas del cliente, experiencia del cliente o política organizacional.
- Variables del mercado. Por ejemplo, tendencias de mercado o reputación o restricciones del producto o la tecnología.
- Capacidad del fabricante. Por ejemplo, disponibilidad de formación o soporte, certificaciones, reputación o estabilidad del fabricante.

Como puede apreciarse en la Figura 10, el método STACE comprende cuatro procesos relacionados entre sí. A continuación se resumen cada una de estas fases:

1. Selección de la tecnología subyacente y otras cuestiones clave. Se deduce la tecnología en la que se basará el producto a seleccionar a partir de los requisitos de alto nivel definidos por el cliente. El proceso de selección incluye:
    a. Definición de criterios de evaluación
    b. Búsqueda y reconocimiento de alternativas disponibles
    c. Revisión de criterios y requisitos basados en las tecnologías disponibles
    d. Evaluación y selección de la mejor tecnología.
2. Definición de criterios de evaluación socio-técnicos. Obtener los criterios socio-técnicos a partir de los requisitos de alto nivel y la tecnología seleccionada.
3. Búsqueda y reconocimiento de los productos COTS disponibles. Buscar los productos candidatos basándose en las características de funcionalidad y realizar un reconocimiento para reducir el número de candidatos final. Esto puede hacerse a través de una rápida revisión de la documentación de los productos.

4. Revisión de los requisitos y criterios socio-técnicos basada en los productos COTS disponibles. Examinar los productos alternativos en función de los criterios de evaluación utilizando la técnica AHP [Saaty 1990]

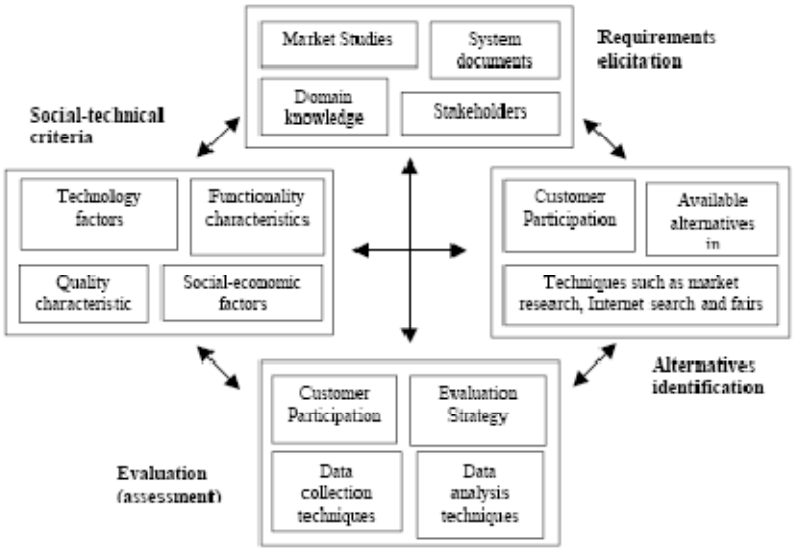


Figura 10. Método STACE [Kunda y Brooks 1999]

Como importante aportación para este trabajo la metodología STACE proporciona una clasificación de los factores no técnicos obtenida a partir de un estudio realizado sobre siete organizaciones del Reino Unido [Kunda y Brooks 2000]. Sin embargo, no proporciona ninguna forma de obtener los atributos que no sea a través de los requisitos de usuario, lo que supone, definirlos desde el principio para cada proyecto de evaluación. Del mismo modo, aunque establece que las características de calidad no cambian de un proyecto a otro, no las facilita un catálogo de éstas ni proporciona ningún proceso para obtenerlas.

Otra importante aportación es el uso de factores clave (*keystone)* que permiten filtrar productos que no los verifican. El uso de este tipo de factores puede aumentar la eficiencia del proceso. En la metodología STACE el factor clave definido es la tecnología. Por tanto, se filtra el número de productos COTS candidatos basándose en la tecnología que el usuario define en los requisitos iniciales. Sin embargo, el proceso aquí utilizado es costoso ya que

hay involucrados dos procesos de evaluación: selección de la tecnología idónea y selección del producto COTS. En nuestro caso, recomendamos una selección de criterios inicial a usar en el filtrado como son: la tecnología, el coste del producto y otros que el cliente considere clave (el producto no se selecciona si no lo verifican) o que sean necesarias por el tipo de proyecto de evaluación. De esta forma, el modelo de calidad definido no incluirá dichas características ya que éstas se evaluarán en una fase previa dependiendo del proyecto de evaluación específico.

En el caso de los productos de seguridad informática tratado en este trabajo de tesis, el factor clave propuesto es el nivel de seguridad del producto según los Criterios Comunes (EAL). Tal como se comentó en la sección 1.1, dada la importancia de la sub-característica seguridad para este tipo de productos, es fácil caer en la tentación de seleccionarlos según los requisitos funcionales y de seguridad (nivel EAL). Por ello, los productos candidatos se filtrarán según su nivel de seguridad y los requisitos funcionales, para luego, aplicar el resto de criterios a los productos restantes.

Por otra parte, la metodología presupone que se han adquirido los requisitos del cliente, por tanto, no es válida cuando no existe este dato de entrada. En este caso, los modelos de evaluación predefinidos actuarían como requisitos de usuario cuando éstos no existen solucionando así esta carencia.

2.2.3.3 El método PORE (*Procurement-Oriented Requirements Engineering*)

El método PORE [Maiden, Ncube*, et al.* 1997, Maiden y Ncube 1998] es un método basado en plantillas para adquisición de requisitos, evaluación y selección de productos COTS. PORE soporta adquisición de requisitos y selección de producto de forma iterativa hasta que uno o más satisfacen un número suficiente de requisitos de usuario. Para ello, divide el proceso en etapas que estructura a través de plantillas. En la Figura 11 se resume el proceso PORE. Como puede observase en la figura, a lo largo del tiempo el número de productos considerados es inversamente proporcional a la cantidad de requisitos aplicados. Así, en la primera etapa (*template 1)* hay pocos requisitos de usuario pero muchos productos candidatos. Según los productos se van rechazando, el número de requisitos de usuario aumenta y el de los productos candidatos disminuye.

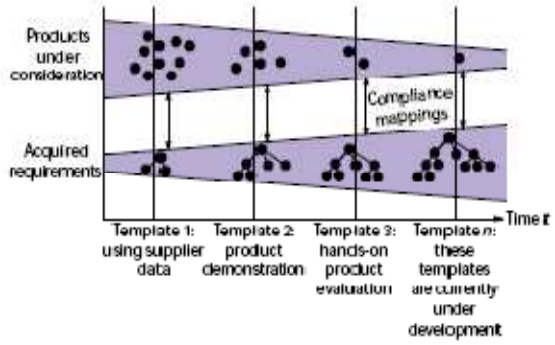


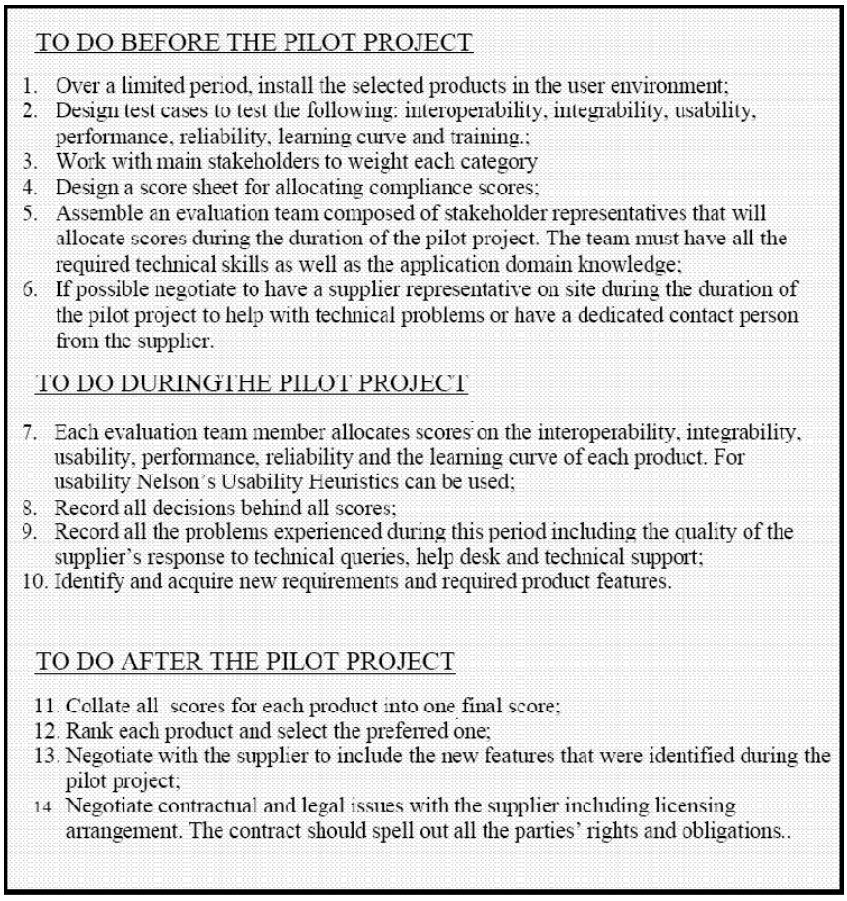
Figura 11. Resumen del proceso de selección de productos PORE.

En cada una de las etapas se aplica una plantilla. Cada plantilla proporciona las técnicas e indicaciones a seguir para obtener los requisitos del cliente y la información necesaria del producto, así como, para seleccionar y rechazar productos en base a:

- Plantilla 1. La información dada por el fabricante: propaganda comercial, documentos técnicos, conversaciones telefónicas, respuestas a cuestionarios, información del sitio web y análisis de mercado público o interno.

- Plantilla 2. Demostraciones dirigidas por el proveedor o fabricante usando casos de uso para requisitos individuales. Es común tener demostraciones de productos durante el proceso de selección. Dichas demostraciones se utilizarán para llevar a cabo verificaciones más complejas de los productos candidatos.

- Plantilla 3. Análisis de los productos dirigido por el evaluador en el entorno del cliente. El evaluador puede recomendar al cliente que implemente uno o dos productos en el entorno de trabajo para uso de evaluación durante un período limitado. De esta forma, se verifica la compatibilidad del producto e interoperabilidad y también que se ajusta a la arquitectura existente en la organización. En la Figura 12 puede observarse parte de esta plantilla.



TO DO BEFORE THE PILOT PROJECT

1. Over a limited period, install the selected products in the user environment;
2. Design test cases to test the following: interoperability, integrability, usability, performance, reliability, learning curve and training.;
3. Work with main stakeholders to weight each category
4. Design a score sheet for allocating compliance scores;
5. Assemble an evaluation team composed of stakeholder representatives that will allocate scores during the duration of the pilot project. The team must have all the required technical skills as well as the application domain knowledge;
6. If possible negotiate to have a supplier representative on site during the duration of the pilot project to help with technical problems or have a dedicated contact person from the supplier.

TO DO DURINGTHE PILOT PROJECT

7. Each evaluation team member allocates scores on the interoperability, integrability, usability, performance, reliability and the learning curve of each product. For usability Nelson's Usability Heuristics can be used;
8. Record all decisions behind all scores;
9. Record all the problems experienced during this period including the quality of the supplier's response to technical queries, help desk and technical support;
10. Identify and acquire new requirements and required product features.

TO DO AFTER THE PILOT PROJECT

11. Collate all scores for each product into one final score;
12. Rank each product and select the preferred one;
13. Negotiate with the supplier to include the new features that were identified during the pilot project;
14. Negotiate contractual and legal issues with the supplier including licensing arrangement. The contract should spell out all the parties' rights and obligations..

Figura 12. Parte de la plantilla 3 definida en el proceso PORE.

Este método ha sido posteriormente adaptado al entorno de la banca para la selección de componentes COTS dando lugar a un proceso denominado SCARLET (inicialmente conocido como BANCKSEC) [Maiden, Kim*, et al.* 2002]

En relación con este trabajo de tesis, la aportación principal de este método es el uso de una técnica de filtrado iterativo a través de plantillas predefinidas que permite hacer la evaluación más eficiente al aplicar al conjunto inicial de productos COTS un número reducido de requisitos y, a medida que el número de productos candidatos disminuye, aplicar técnicas de evaluación más complejas para verificar el resto de requisitos. La técnica de filtrado en nuestro caso se utilizará tal como se especificó en la sección 2.2.3.2. Por otra parte, el método ha sido obtenido a través de entrevistas a organizaciones y de la aplicación práctica en un caso real de evaluación [Maiden, Ncube 1997], lo cual aumenta las

posibilidades de que sea aplicable en el entorno empresarial al ajustarse a las prácticas actuales de selección de productos y no cambiar radicalmente la forma de trabajar de los profesionales del entorno. Por último, el proceso es altamente guiado a través de las plantillas (*templates)* que en realidad, no son formatos de documentos pre-configurados para completar en cada etapa, como su nombre podría dar a entender, sino guías detalladas con los pasos a seguir en las mismas.

Sin embargo, el método sólo es aplicable a la selección de productos COTS para la adquisición por parte de una organización y, en este caso, tiene una fuerte orientación a la adquisición de requisitos de usuario, por otra parte, objetivo principal de la metodología PORE. De nuevo, en este caso también la incorporación de modelos de calidad predefinidos cuando no existen requisitos de usuario podría solucionar esta carencia de la metodología. Además, existen otros problemas para la aplicación de este método:

- No da soporte a la búsqueda inicial de productos.
- La definición de características técnicas de calidad se debe realizar desde el principio para cada proyecto.

2.2.3.4 CAP (*COTS Acquisition Process*)

EL método CAP [Ochs, Pfahl 2001] está basado en el método OTSO pero está estrictamente orientado a medidas con el objetivo de incrementar la rentabilidad de la evaluación. En la Figura 13 se presenta un resumen del proceso completo.

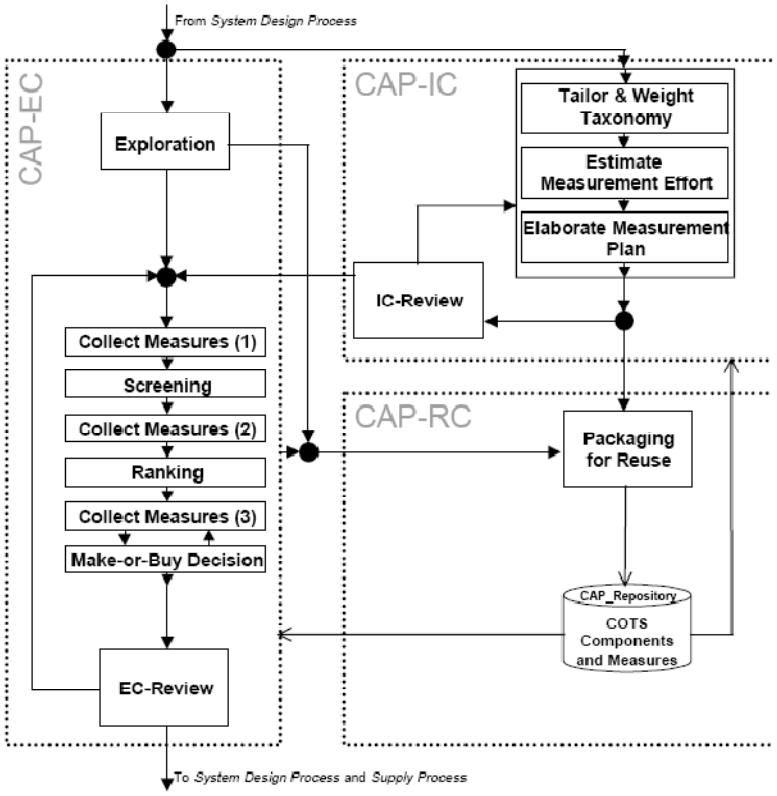


Figura 13. Resumen del modelo CAP a través de un diagrama de control de flujo [Ochs, Pfahl 2001]

Los principios fundamentales en los que se basa el método CAP son:

- Utiliza un modelo de calidad previamente construido basado en ISO/IEC 9126-1 [ISO 2001a], junto con otros criterios extraídos de otras investigaciones [Kontio 1996, Kontio, Caldiera 1996, Maiden y Ncube 1998] y la propia experiencia. Los requisitos de usuario son adaptados a los criterios definidos en el modelo.

- Estima el esfuerzo requerido para aplicar todos los criterios de evaluación a todos los productos COTS candidatos para así obtener el coste estimado de aplicar CAP, el presupuesto y otras restricciones relacionadas con los recursos disponibles.

- La selección del producto se basa en una fase de exploración (*Exploration*) que consiste en la evaluación a través del filtrado iterativo de productos COTS y la selección final a través de la aplicación del método AHP [Saaty 1990].

- Después de cada etapa se ejecuta una actividad de revisión en la que se verifica que todas las actividades se dirigieron correctamente para llevar a cabo retroalimentación del proceso y poder volver hacia atrás en caso necesario.

- Almacena el conocimiento obtenido para reutilización en futuros proyectos de evaluación.

Aunque el método CAP utiliza un modelo de calidad adaptado del estándar para productos COTS y, en ese sentido, es fácilmente integrable con modelos de calidad predefinidos, es un método largo y muy complejo. La estimación del esfuerzo necesario para la toma de medidas es una tarea que consume mucho tiempo y que sólo se justifica en grandes proyectos de evaluación. Por otra parte, aunque realiza dos iteraciones en el proceso de filtrado de productos no puntualiza en base a qué criterios realiza el filtrado. Por último, al igual que la mayoría de los métodos vistos hasta ahora, no incluye soporte de evaluación de un solo producto, por lo que sólo podría aplicarse cuando el objetivo es la selección de un producto frente a otros.

2.2.3.5 RCPEP (*Requirements-driven COTS Product Evaluations Process*)

RCPEP [Lawlis, Mark 2001] utiliza como entrada los requisitos de usuario en lugar de usar una lista de criterios de evaluación para determinar la idoneidad o calidad de productos COTS. Para ello, identifica cada producto que posiblemente cumple dichos requisitos. Para cada uno de estos productos, los evaluadores determinan cuántos, de los requisitos definidos por los usuarios, satisfacen. Como proceso de toma de decisiones, RCPEP utiliza MCDM (Multiple Criteria Decision Making) [Zeleny 1982] sobre matrices de requisitos (ver Figura 14 (a)). Una vez filtrados los productos que cumplen un mínimo establecido de los requisitos definidos por los usuarios, los productos restantes se someten a evaluación implementándolos en escenarios prácticos de uso. Para ello, se refinan los requisitos y se evalúan sobre los escenarios configurados. De nuevo, se utiliza MCDM para obtener la puntuación final en base a los pesos obtenidos a través de los usuarios y las puntuaciones de cada requisito asignadas por los evaluadores. Finalmente, se realiza un proceso de análisis entre todos los evaluadores de los datos recogidos durante el proceso y se hacen las recomendaciones finales sobre los productos (ver Figura 14 (b)).

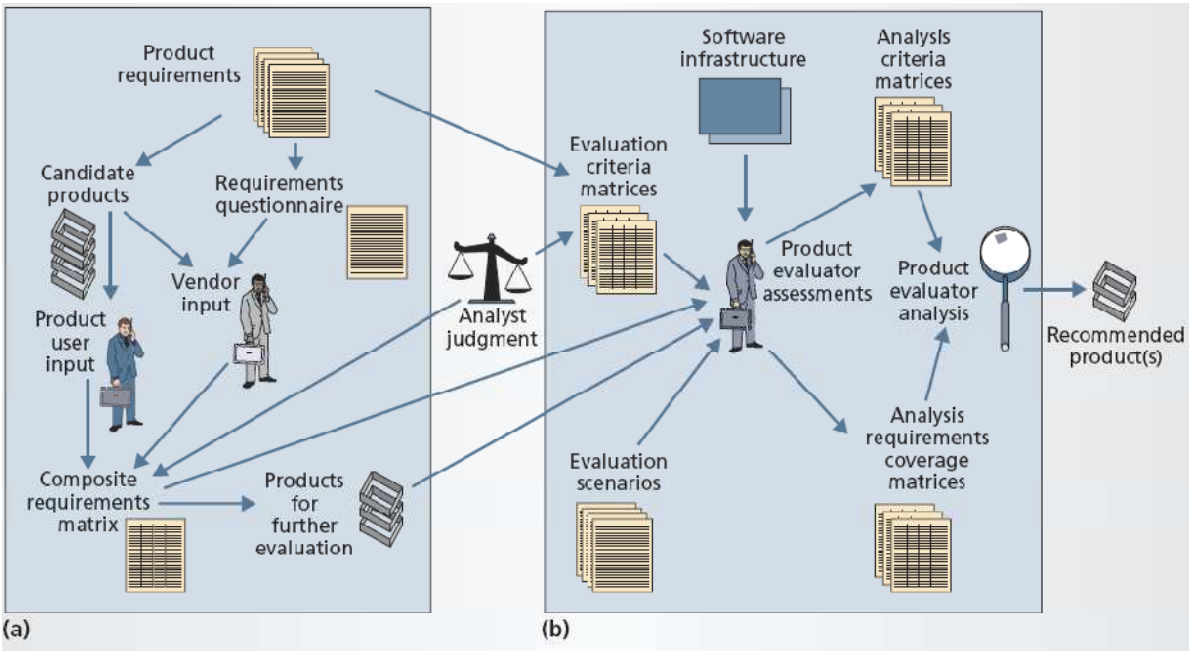


Figura 14. Resumen del proceso RCPEP.

El proceso RCPEP es un ejemplo de proceso de evaluación realizada por profesionales de la industria (no académicos). Esto tiene como ventaja que se adapta a las técnicas y recursos utilizados en el entorno empresarial y, como desventaja, que el proceso definido es poco formal y metódico. Así, en el método no se especifica cómo realizar la búsqueda inicial de productos, ni en base a qué criterios de los definidos por el usuario realizar el filtrado posterior. Además, no distingue los requisitos esenciales o factores clave (aquellos que debe cumplir cualquier producto seleccionado) del resto, con lo que puede ocurrir que en la primera etapa de filtrado se descarten productos que verifiquen estos requisitos y a cambio se seleccionen otros que, aún cumpliendo un mayor número de los requisitos no esenciales, no verifican éstos. Además utiliza MCDM que, si bien es mucho más simple que AHP, diversos estudios han corroborado que AHP es más eficiente [Chu y Kalaba 1979, Shoemaker y Waid 1982, Saaty 1992, Forman 1993, Kontio 1996]. Por último, utiliza la técnica de consenso entre evaluadores para la toma de decisiones final. Para utilizar esta técnica es necesario disponer de un mínimo de evaluadores (por ejemplo, entre 5 y 30 en la técnica Delphi [Landeta 1999]). Por tanto, este método sólo es práctico en grandes proyectos de evaluación.

2.2.3.6 El método PECA (*Planning, Establishing, Collecting, Analyzing*)

El método PECA [Comella-Dorda, Dean 2002] adapta el ISO/IEC 14598 [ISO 1999a] a la selección de COTS en organizaciones utilizando la experiencia adquirida. En la Figura 15 pueden verse las fases utilizadas y la relación entre las mismas. Las principales características del método son las siguientes:

1. Con respecto a la norma 14598-4 [ISO 1999b], modifica el orden de las fases situando la fase de diseño de la evaluación antes que el establecimiento de los criterios de evaluación.

2. Utiliza los requisitos de los usuarios pero también las características de calidad (basadas en el modelo de calidad de ISO/IEC 9216 [ISO 2001c]) y organizacionales de los productos para obtener los criterios. Aunque no proporciona un modelo de calidad concreto para este tipo de productos y, por tanto, es necesario definirlo desde cero para cada proyecto, sí proporciona un catálogo de atributos de calidad no técnicos.

3. Proporciona soporte sobre cómo involucrar a las partes interesadas en la formulación de objetivos y también claves para la selección del equipo de evaluación.

4. Describe los métodos y técnicas disponibles en la actualidad para llevar a cabo cada una de las fases del proceso de evaluación. También recomendaciones basadas en su experiencia.

5. Utiliza el filtrado iterativo para la reducción óptima inicial de productos COTS. No especifica cuantas fases de iteración utilizar ni qué criterios usar en el filtrado, tan sólo, recomienda usar los menos costosos en las primeras fases de iteración.

Aunque el método PECA es una buena base para la introducción en la evaluación metódica de productos COTS, el proceso es demasiado genérico para poder ser aplicado directamente. Precisamente esta generalidad parte del punto de ser válido para cualquier situación por lo que, al igual que el estándar, requiere de adaptación a la situación particular antes de aplicarlo.

Por otra parte, aunque no proporciona soporte a la elaboración de informes técnicos de evaluación cuando se evalúa un único producto, es un buen punto de partida por la contribución realizada en relación a la adaptación del método a la evaluación de productos COTS. En relación a este punto, la metodología PECA cambia el orden de las fases situando la planificación de la evaluación antes que el establecimiento de los criterios a evaluar. Si bien es cierto que incorpora en la fase de "Planificación de la evaluación" las actividades que en la norma ISO/IEC 14598-5 aparecen en la fase de "Especificación de la evaluación", al menos para el propósito de evaluación tratado en esta tesis y según nuestra experiencia [Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2007a], no es posible hacer una planificación coherente sin conocer antes qué criterios se van a evaluar. En nuestro caso, al proporcionar un modelo de calidad predefinido sólo será necesario adaptarlo a las especificaciones del proyecto concreto y a los requisitos de usuario. Pero cuando en el proceso de evaluación se utilizan sólo los requisitos de usuario o se define el modelo de calidad partiendo de cero, es todavía más difícil conocer las necesidades de costes y recursos (humanos y materiales) a emplear cuando aún se desconoce qué propiedades se van a medir en la evaluación.

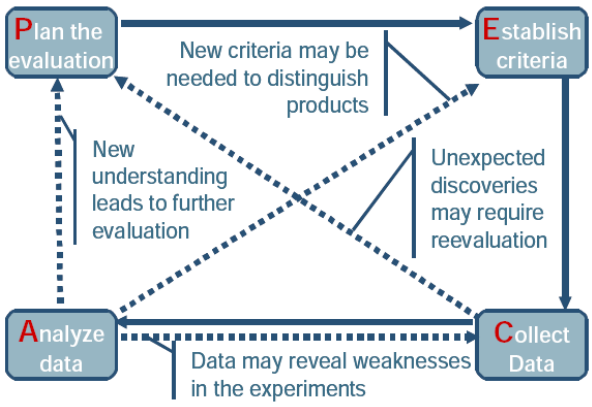


Figura 15. El método PECA [Comella-Dorda, Dean 2002]

2.2.4. Evaluación de la seguridad del software

Existen diferentes métodos y procesos que se especializan en la evaluación de una o varias características de calidad en lugar de evaluar la calidad del software como una cualidad global. Por la aplicación de este trabajo al dominio de los productos COTS de seguridad informática, nos interesan especialmente aquellos procesos enfocados a la medición de la sub-característica de calidad seguridad. Los productos COTS de seguridad informática como dominio o área de funcionalidad se relacionan con la seguridad como propiedad de calidad de dos formas:

1. Por una parte, su funcionalidad consiste en proporcionar algún servicio o servicios de seguridad[9]. Por tanto, las metodologías de evaluación de seguridad actuales son de aplicación para la evaluación de la funcionalidad de dichos productos COTS.

2. Por otra parte, la seguridad de dichos productos COTS es una importante característica de los mismos (el software que proporcione protección de seguridad debe ser seguro). Prueba de ello es el gran número de productos de seguridad que han sido certificados como seguros según los Criterios Comunes[10].

Por tanto, dada la importante influencia de la evaluación de la seguridad en este trabajo de tesis, en esta sección trataremos el estándar actual de evaluación de seguridad: ISO/IEC 15408 [ISO 2005e].

[9] La norma ISO/IEC 7498-2 [ISO 1982] define "servicio de seguridad" como un servicio que garantizan la seguridad adecuada de los sistemas y de la transferencia de datos. Los servicios de seguridad definidos por la norma son: autenticación, control de acceso, confidencialidad de los datos, integridad de los datos y no repudio.

[10] Los productos certificados pueden verse en http://www.commoncriteriaportal.org/products.html.

2.2.4.1 Los Criterios Comunes. ISO/IEC 15408: Criterios de evaluación de la seguridad de las Tecnologías de la información

Los Criterios Comunes tienen su origen en 1990 y surgen como resultado de la armonización de los criterios utilizados por el Departamento de Defensa de EE.UU (TCSEC, *Trusted Computer System Evaluation Criteria*, frecuentemente llamado "Libro Naranja") y los estándares europeo (ITSEC, *Information Technology Security Evaluation Criteria*, comúnmente conocido como "Libro Blanco") y canadiense (CTCPEC, *Canadian Trusted Computer Product Evaluation Criteria*) para la seguridad de productos con el fin de que el resultado del proceso de evaluación pudiese ser aceptado en múltiples países. En 1999 los Criterios Comunes en su versión 2.0 fueron adoptados por la International Organisation for Standards (ISO) como estándar internacional en 1999. En la actualidad la versión 2.1 está vigente como norma ISO/IEC 15408 [ISO 2005e].

Para certificar un producto según los Criterios Comunes se deben comprobar, por parte de uno de los laboratorios independientes aprobados, numerosos parámetros de seguridad que han sido consensuados y aceptados por 22 países de todo el mundo, hasta el punto de que algunos países como EEUU exigen esta certificación para el software que se use en sus sistemas. En el proceso de evaluación se certifican para un producto específico los siguientes aspectos:

1. Si los requisitos del producto están definidos correctamente.
2. Si los requisitos están implementados correctamente.
3. Si el proceso de desarrollo y documentación del producto cumple con ciertos requisitos.

Los Criterios Comunes establecen entonces un conjunto de requisitos para definir las funciones de seguridad de los productos y sistemas de Tecnologías de la información y de los criterios para evaluar su seguridad. El proceso de evaluación, realizado según lo prescrito en los Criterios Comunes, garantiza que las funciones de seguridad de tales productos y sistemas reúnen los requisitos declarados. Así, los clientes pueden especificar la funcionalidad de seguridad de un producto en términos de perfiles de protección estándares y de forma independiente seleccionar el nivel de confianza en la evaluación de un conjunto

definido desde el EAL1 al EAL7. Un perfil de protección (*Protection Profile*) define un conjunto de objetivos y requisitos de seguridad, independiente de la implantación, para un dominio o categoría de productos que cubre las necesidades de seguridad comunes a varios usuarios. Los perfiles de protección son reutilizables y normalmente públicos y están compuestos de:

- Requisitos funcionales (SFR, *Security Funcional Requirement*) proporcionan mecanismos para hacer cumplir la política de seguridad. Como ejemplos de requisitos funcionales mencionar la protección de datos de usuario, el soporte criptográfico, la autenticación, la privacidad o el control de acceso.
- Requisitos de confianza o aseguramiento (SAR, *Security Assurance Requirement*) proporcionan la base para la confianza en que un producto verifica sus objetivos de seguridad.

Los requisitos de confianza se han agrupado en niveles de confianza en la evaluación (EAL, Evaluation Assurance Levels) que contienen requisitos de confianza construidos específicamente en cada nivel. Los EALs proporcionan una escala incremental que equilibra el nivel de confianza obtenido con el coste y la viabilidad de adquisición de ese grado de confianza. El incremento de confianza de un EAL a otro se obtiene incrementando rigor, alcance y/o profundidad en el componente y añadiendo componentes de confianza de otras familias de confianza (por ejemplo, añadiendo nuevos requisitos funcionales). Los niveles de confianza en la evaluación definidos en el ISO/IEC 15408-3 [ISO 2005d] van desde EAL1 (el menor) a EAL 7 (el mayor) y se definen de forma acumulativa (verificaciones de nivel n+1 implican realizar las de nivel n):

1. EAL1 (funcionalidad probada): es aplicable donde se requiere tener cierta confianza de la operación correcta, y donde además, las amenazas a la seguridad no son vistas como serias. Una evaluación en este nivel debe proporcionar evidencia de que las funciones del objeto de evaluación son consistentes con su documentación, y que proporcionan protección útil contra amenazas identificadas.

2. EAL2 (estructuralmente probado): requiere la cooperación del desarrollador en términos de la distribución de la información del diseño y de los resultados de las

pruebas. Proporciona confianza a través de un análisis de las funciones de seguridad, usando una especificación funcional y de interfaz, manuales y diseño de alto nivel del producto para entender el comportamiento de seguridad. Además, en este nivel se verifica que el desarrollador realizó un análisis de vulnerabilidades a través de la ejecución de pruebas de caja negra[11] (*black-box).*

3. EAL3 (probado y verificado metódicamente): permite a un desarrollador alcanzar una máxima garantía de ingeniería de seguridad positiva en el estado de diseño sin la alteración substancial de prácticas de desarrollo válidas existentes. El análisis en este nivel se apoya en las pruebas de caja gris[12] (*grey box*), la confirmación selectiva independiente de los resultados de las pruebas del desarrollador, y la evidencia de búsqueda de vulnerabilidades obvias del desarrollador. Además, se realizan controles del entorno de desarrollo y de gestión de configuración del producto.

4. EAL4 (diseñado, probado y revisado metódicamente): este nivel le permite a un desarrollador alcanzar máxima garantía de ingeniería de seguridad positiva basada en buenas prácticas de desarrollo comercial, las cuales, aunque rigurosas, no requieren del conocimiento especializado substancial, destreza, ni otros recursos. En este caso, el análisis se apoya en el diseño de bajo nivel de los módulos del producto y se realiza búsqueda de vulnerabilidades independiente de las pruebas realizadas por el desarrollador. Los controles de desarrollo se apoyan en un modelo de ciclo de vida de desarrollo, identificación de las herramientas utilizadas y gestión de configuración automatizada.

5. EAL5 (diseñado y probado semiformalmente): permite a un desarrollador alcanzar máxima garantía de ingeniería de seguridad positiva mediante la aplicación

[11] En las pruebas de caja negra (también denominadas funcionales) las pruebas se derivan de la especificación externa del comportamiento del software sin tener en cuenta la organización interna, ni la lógica, control o flujo de datos [Adrion, Branstad*, et al.* 1982] y [IEEE 2003].

[12] Pruebas que emulan el comportamiento del atacante cuando éste tiene información parcial del sistema (por ejemplo, diseño de alto nivel, parte del código, etc.) [ISO 2005d].

moderada de técnicas de ingeniería de seguridad. La confianza se apoya, en este caso, en un modelo formal y una presentación semiformal de la especificación funcional y el diseño de alto nivel. La búsqueda de vulnerabilidades debe asegurar la resistencia relativa a los ataques de penetración.

6. EAL6 (diseño verificado y probado semiformalmente): permite a los desarrolladores alcanzar una alta garantía en la aplicación de técnicas de ingeniería de seguridad para un entorno de desarrollo riguroso y donde el objeto de evaluación es considerado de gran valor para la protección del alto costo o estimación de esos bienes contra riesgos significativos. Además, es aplicable para el desarrollo de objetos de evaluación, destinados a salvaguardar la seguridad informática en situaciones de alto riesgo donde el valor de los bienes protegidos justifica los costos adicionales. El análisis en este nivel se apoya en un diseño modular y en una presentación estructurada de la implementación del producto COTS. La búsqueda de vulnerabilidades debe mostrar una alta resistencia a los ataques de penetración.

7. EAL7 (diseño verificado y probado formalmente): es aplicable al desarrollo de objetos de evaluación de seguridad, para su aplicación en situaciones de muy alto riesgo o donde el alto valor de los bienes justifica los más altos costos. La aplicación práctica del nivel EAL7 está limitada actualmente a objetos de evaluación con seguridad estrechamente enfocada a la funcionalidad, y que es sensible al análisis formal y extenso. Este EAL representa un incremento significativo respecto a la garantía de nivel EAL6 a través del requisito de análisis de gran amplitud, mediante representaciones formales y correspondencia formal y pruebas de gran amplitud. Además, el evaluador confirmará de forma independiente y compelta los resultados de las pruebas de caja blanca[13] (*White-box*) realizadas por el desarrollador.

[13] Pruebas utilizadas para verificar que la salida de un programa, para ciertas entradas, es conforme al diseño interno y a la implementación del programa. Este tipo de pruebas depende totalmente de la organización lógica interna del software ([IEEE 2003], [Adrion, Branstad 1982]).

Los niveles EAL 5 al 7 incluyen modelos y demostraciones semi-formales y formales por lo que suponen un coste alto para los productos comerciales COTS. Por tanto, se aplican a productos con objetivos de seguridad muy específicos (entorno militar, por ejemplo). Por otra parte, estos niveles requieren de la generación de una gran cantidad de documentación durante el proceso de desarrollo (en el nivel EAL7 el evaluador debe tener el código completo del producto) que debe entregarse al evaluador para que éste pueda confirmar la información. Esta situación en la mayor parte de los casos no es posible para productos COTS. Por este motivo, los niveles EAL 5 a 7 no forman parte de este trabajo.

Además, para la aplicación de los Criterios Comunes se proporciona una metodología con los criterios a evaluar para cada uno de los niveles de confianza [Common Criteria 2004] estandarizada por la norma ISO/IEC 18045 [ISO 2004b].

El objetivo principal de este trabajo de tesis doctoral es obtener un modelo de calidad integral que aúne los criterios de evaluación definidos en los distintos estándares y trabajos de investigación relacionados. Los Criterios Comunes definen a través de los perfiles de protección los criterios funcionales de adecuación y de seguridad según el tipo de producto. Por tanto, consideraremos dentro de la evaluación la medición de dichas sub-características a través de la norma estándar o su nivel de evaluación EAL cuando el producto haya sido evaluado.

2.3.Modelos de evaluación de calidad del software

2.3.1. Los primeros modelos de calidad del software

Como ya hemos visto en la sección anterior, los modelos de evaluación de la calidad del software describen características generales o comunes a la mayoría de los productos software. Esta descomposición en características se realiza con el fin de facilitar la verificación del grado de cumplimiento de las mismas a los requisitos previamente definidos obteniendo así la calidad del software del producto. Los primeros en utilizar este enfoque de descomposición descendente para obtener modelos de calidad fueron McCall [McCall, Richards 1977] y Boehm [Boehm, Brown 1978]. En el modelo de McCall, comúnmente

llamado FCM (Factor Criteria Metric), se definen once atributos o características llamados *factores* que se clasifican según su uso:

- Capacidad de operación del software.
- Capacidad de revisión o mantenimiento del software
- Capacidad de transición o adaptación a otros entornos

Son factores de operación del software, por ejemplo, la usabilidad, la integridad o la eficiencia (la lista completa puede verse en la columna 2 de la Figura 16). Pero es difícil medir o cuantificar dichos factores sin antes de descomponerlos en otros menos abstractos y que puedan medirse directamente. Por ello, cada factor está compuesto de criterios de nivel inferior que son más fáciles de entender y de medir que los factores. Por ejemplo, son criterios del factor eficiencia la eficiencia de almacenamiento y la eficiencia de ejecución. Por último, algunas veces se requiere un nivel de descomposición mayor, en el que se asocia el criterio de calidad con un conjunto de métricas que directamente se pueden medir. El modelo FCM de McCall fue utilizado posteriormente como base para el desarrollo de normas estándares.

El modelo de Boehm (Figura 16) es similar al de McCall aunque reduce el número de atributos de once, que se utilizan en el modelo de McCall, a siete. En la Figura 17 puede verse el modelo completo. En ambos modelos se asume que todos los factores de calidad importantes para cualquier proyecto están publicados en el modelo. Por tanto, se consideran modelos de calidad fijos o estáticos.

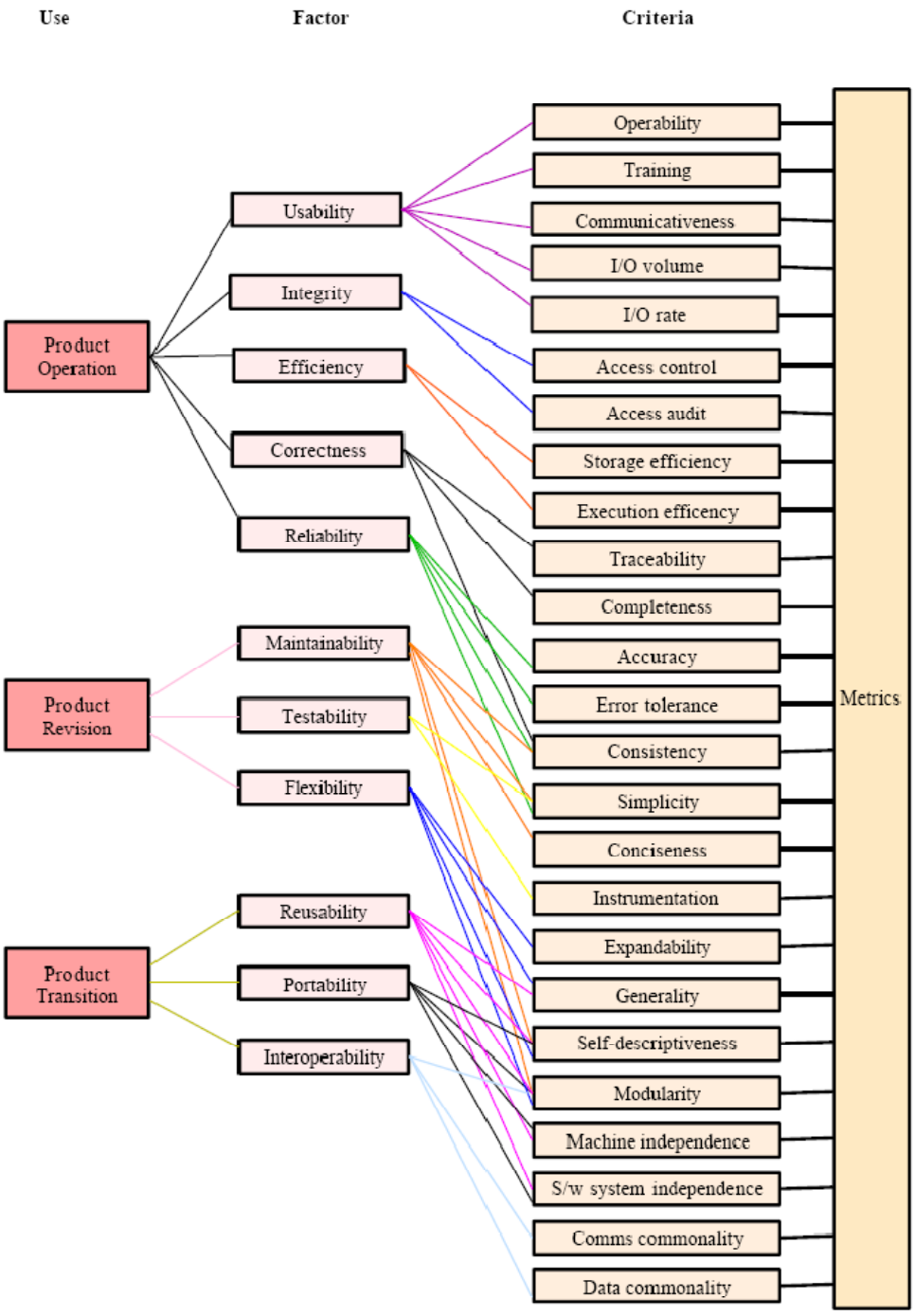


Figura 16. Modelo de calidad del software de McCall [McCall, Richards 1977]

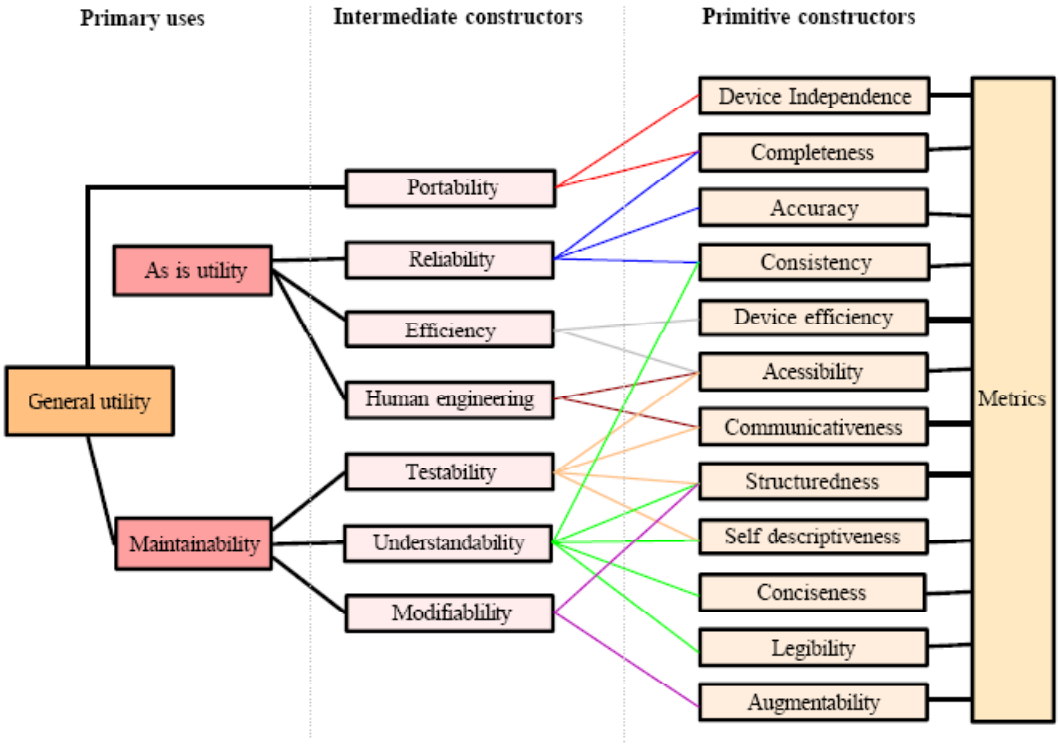


Figura 17. Modelo de calidad del software de Boehm [Boehm, Brown 1978]

En contraposición a los modelos de calidad estáticos, surge el alcance "define-tu-propio-modelo" (*define-your-own models*) [Fenton y Pfleeger 1997] en el que se acepta el planteamiento de la calidad como descomposición en propiedades del software, pero no se acepta un modelo preestablecido y común para todos los productos software. Por el contrario, el modelo debe definirse en cada proyecto y producto dado por acuerdo de las partes involucradas. Ejemplos de este enfoque son los métodos de Gilb [Gilb 1988] y COQUAMO (COnstructive QUAlity MOdel) de Kitchenham y Walker [Kitchenham y Walker 1989] que extiende el primero. El método de Gilb está relacionado con su filosofía de desarrollo evolutivo en el que el producto se proporciona al usuario incrementalmente según el orden de prioridad de funciones. Para identificar dicha prioridad, se pide al usuario que identifique los atributos software clave durante la fase de especificación del proyecto. Por último, cabe destacar el paradigma GQM (*Goal-Question-Metric*: Objetivo-Pregunta-Métrica) propuesto por Basili y Rombach [Basili y Weiss 1984] y ampliamente utilizado para la obtención de modelos de calidad. En realidad, GQM es un enfoque general de la

medición, pero se trata de un método muy apropiado para evaluar la calidad del software en cada proyecto. Este método comienza con una definición de un objetivo (en nuestro caso, evaluar la calidad de un producto software). Dicho objetivo es descompuesto en diferentes preguntas dividiéndose así en sus principales atributos (por ejemplo, ¿qué funcionalidad debe tener el producto software?, ¿cómo influye la eficiencia en la calidad del producto?, etc.). Luego se refina cada pregunta obteniendo métricas que proporcionan información para responder a estas preguntas (por ejemplo, tiempo medio de funcionamiento entre fallos de la aplicación). Por tanto, cada pregunta puede ser vista como una característica del modelo de calidad.

Dado que el modelo está fuertemente ligado con el desarrollo del software, su aplicación a productos finales adquiridos o COTS es poco útil, al menos en los términos citados, ya que no puede aprovecharse la toma de requisitos para la obtención de los atributos de calidad y la importancia o peso de los mismos con respecto al total.

2.3.2. Estandarización de modelos de calidad del software: ISO 9126 e IEEE 1061

La idea de tener un único modelo válido para expresar la calidad de cualquier producto software resulta atractiva por varias razones. Por una parte, no hay duda de que disponer de un modelo ya definido ahorra tiempo al no tener que definir tu propio modelo. Pero, además, tiene otra ventaja fundamental: facilita la comparación entre productos software. Por ello, en 1991, tomando como base el modelo de McCall, se propuso el estándar internacional llamado *Software product evaluation- Quality characteristics and guidelines for their use* [ISO 2001c] comúnmente referenciado como ISO 9126. Con esta norma se establecía, tanto un proceso de evaluación, como un modelo de calidad del software estándar. El proceso de evaluación fue posteriormente revisado y, como fruto de dicha revisión, se publicó la norma *Software product evaluation -- Part 1: General overview* [ISO 1999a]. En un esfuerzo por coordinar el contenido de la norma ISO 9126 con el de ISO/IEC 14598-1, la norma ISO/IEC 9126-1:2001 sustituyó a su predecesora. Posteriormente, tanto ISO/IEC 9126-1:2001, como ISO/IEC 14598-1 fueron completadas con otras normas,

obteniendo las series ISO/IEC 9126 e ISO/IEC 14598. La relación entre las series de normas ISO/IEC 9126 e ISO/IEC 14598 se muestran en la Figura 18.

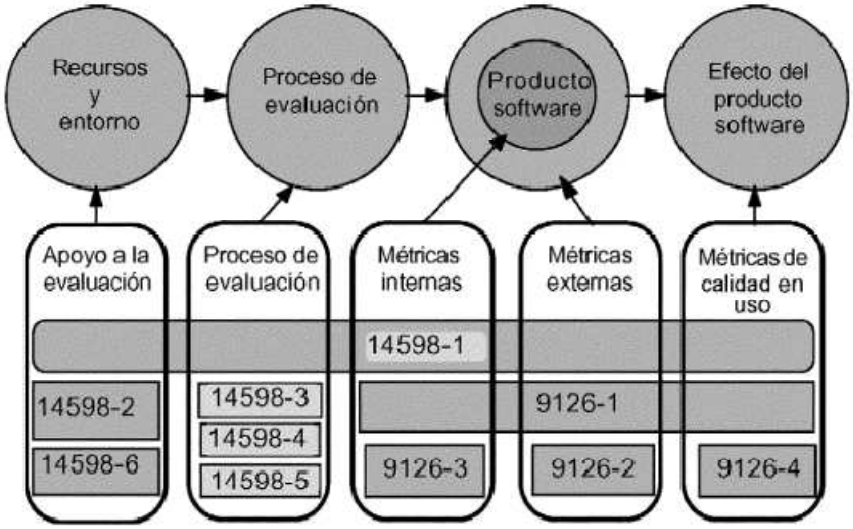


Figura 18. Relación entre las normas ISO 9126 e ISO 14598 [ISO 2001a]

Este estándar describe la calidad del software, con mínimo solapamiento, a partir de seis características generales que se dividen a su vez en sub-características y proporciona recomendaciones y requisitos para métricas. Se define, por tanto, un modelo de calidad del software estático a través de características y sub-características lo suficientemente generales para que sean válidas para cualquier producto software.

La serie de normas ISO/IEC 9126 consta de cuatro partes:

- Parte 1: Modelo de calidad [ISO 2001a]
- Parte 2: Métricas externas [ISO 2003a]
- Parte 3: Métricas internas [ISO 2003b]
- Parte 4: Métricas de calidad en uso [ISO 2004a]

ISO 9126 describe un modelo de calidad dividido en dos partes: calidad interna y externa y calidad en uso. La calidad interna y externa hace referencia a la calidad de los productos software en sí y se puede evaluar, bien a través de las características de los productos

intermedios resultantes del desarrollo del producto (entonces hablamos de calidad interna), o bien, a través del comportamiento del propio producto en ejecución (calidad externa). Por otra parte, la calidad en uso surge con el fin de reflejar la visión que tiene el usuario del producto cuando lo usa. Todos ellos están relacionados ya que, un nivel adecuado de calidad interna, es prerrequisito de un comportamiento externo requerido y, éste a su vez, es necesario para conseguir una apropiada calidad en uso. En este trabajo nos interesa especialmente la calidad externa.

Dado que los productos que se pretenden evaluar son productos ya desarrollados, no se intervine en modo alguno el proceso de desarrollo de los mismos. Tampoco se dispone de la información necesaria para poder evaluar sus características internas: en la mayoría de los casos el código fuente no se encuentra disponible y la documentación proporcionada no es suficiente para poder evaluar de forma rigurosa las características internas del producto. En relación a la calidad en uso no nos centramos tampoco en ella por dos razones. En primer lugar, en la evaluación de la calidad en uso, por su propia definición, es necesaria la intervención del usuario. Sin embargo, no en todos los posibles escenarios de selección y comparación de productos software vamos a disponer de usuarios que nos proporcionen su visión del producto. Por ejemplo, cuando se realizan evaluaciones de calidad de un producto o comparaciones de varios para obtener un informe de la calidad ya sea a petición de un cliente o para publicar en una revista de difusión no se dispone de usuarios a los que consultar. Por otra parte, la evaluación de la calidad en uso es un proceso muy costoso tanto en tiempo como en dinero. El hecho de tener que obtener la percepción del usuario a través de cuestionarios a los mismos o de experimentos de uso del propio software para su posterior análisis suele requerir, no sólo tiempo de los usuarios dedicados a esta labor en lugar de a su trabajo diario, sino además la necesidad de un elevado grupo de evaluadores y un alto tiempo de los mismos dedicados a la evaluación.

Para guiar el proceso de desarrollo y selección de productos software, la norma define características a evaluar (relacionadas tanto con la calidad interna y externa, como con la calidad en uso) como relevantes para todo producto software. Dichas características se dividen a su vez en sub-características que no son más que refinamientos de las primeras. Sin embargo, ni características ni sub-características pueden ser medidas directamente, sino

que hay que definir, lo que la ISO denomina atributos. Los atributos son cualidades del software que sí pueden medirse y la combinación ponderada de atributos relacionados con cierta característica nos dará como resultado el valor asociado al mismo. Dichos atributos no son definidos en la norma sino que es necesario determinarlos en el momento en el que se va a llevar a cabo la evaluación convirtiendo el proceso en largo y costoso. Además, el estándar sugiere que algunos atributos podrían contribuir a más de una sub-característica pero nada dice sobre cómo tratarlos en relación al modelo global en ese caso. Se trata, por tanto, de un modelo mixto, en el que están definidas la totalidad de las posibles características y sub-características que pueden darse en cualquier producto software pero deben definirse los atributos específicos del producto y proyecto particular. Pero, además de esto, el ISO 9126-1 dice:

> "En la práctica, no es posible medir todas las sub-características internas y externas de todos 1os componentes de un producto software grande. De igual manera, normalmente no es práctico medir la calidad en uso para todos 1os posibles escenarios usuario-tarea. Los recursos para la evaluación necesitan asignarse a 1os distintos tipos de mediciones dependiendo de 1os objetivos de negocio y de la naturaleza del producto o del proceso de diseño"[ISO 2001a].

Por tanto, también habrá de decidir, entre la totalidad de características y sub-características proporcionadas por el modelo de calidad definido en ISO 9126-1, cuales de ellas afectan en mayor medida a la calidad del producto software concreto que se va a evaluar, obteniendo así, un modelo de calidad específico para dicho producto. Dicho modelo de calidad se denomina **perfil de calidad**. En numerosos trabajos [Kontio 1996, Punter, Solingen 1997, Botella, Burgués*, et al.* 2002, Carvallo, Franch*, et al.* 2004a] se advierte sobre la dificultad de encontrar las características de calidad más relevantes y que, por tanto, deberían ser evaluadas. En este sentido, una técnica ampliamente extendida trata de encontrar perfiles de calidad reutilizables para cierto tipo de producto también llamados modelos de calidad orientados a dominio. De esa forma, al disponer de un catálogo de atributos válido para cierto dominio de productos, éste puede ser utilizado en cada evaluación de productos del dominio en cuestión ahorrando así tiempo en el proceso de evaluación. Algunos ejemplos de dominios para los que se han construido modelos de

calidad específicos son: sitios web [Olsina y Rossi 2002b], herramientas de desarrollo de aplicaciones en entornos visuales [Pérez y Tornés 2005], ERP [Burgués, Franch 2000], Servidores de Correo [Carvallo, Franch 2003], Sistemas de Información de Comercio electrónico [Antonia y Michalis 2008], sistemas críticos de prevención de la seguridad (*safety*) [Ye y Kelly 2004] o sistemas de información bancarios (proyecto ESSISCOPE[14]). Dado que el objetivo principal de este trabajo de tesis es la obtención de un modelo de calidad para el dominio de los productos COTS de seguridad informática, este tipo de modelos nos interesan especialmente y, por ello, los modelos de calidad orientados a dominio existentes se analizan en detalle en la siguiente sección (2.4).

Figura 19. Modelo de calidad para calidad en uso [ISO 2001a].

Las partes 2, 3 y 4 de la serie 9126, proporcionan ejemplos de métricas para calidad interna, externa y en uso. Relacionada con estas normas, ponemos encontrar la norma ISO/IEC 15939 [ISO 2002] que define el proceso de medición del software. Para ello, describe las tareas y actividades, así como, el modelo y la terminología asociada a dicho proceso. El ISO/IEC 15939 en concreto cubre las actividades de medición, información

[14] Información sobre el proyecto ESSISCOPE puede encontrarse en http://www.cse.dcu.ie/essiscope/sm4/**ibisco**.doc

requerida, aplicación de resultados del análisis de medición y determinación de la validez de los resultados.

Aunque la norma no define atributos, sí extiende algunas de las sub-características haciendo referencia a otras normas ISO (ISO/IEC 9241-10 e ISO/IEC 9142-11). La serie de normas ISO/IEC 9241 trata sobre la interacción hombre-computador. La ISO/IEC 9241 consta en la actualidad de 17 partes que proporcionan requisitos y recomendaciones relacionadas con atributos de hardware, software y de entorno que contribuyen a la usabilidad[15]. La parte 11 (Guía sobre usabilidad) [ISO 1998a] explica cómo identificar la información a tener en cuenta cuando se evalúa la usabilidad en términos de satisfacción y eficiencia del usuario. Además, proporciona soporte sobre el modo en el que describir el contexto de uso del producto a evaluar y las medidas de usabilidad a utilizar. Las partes especialmente relacionadas con atributos del software son la 10 y de la 12 a la 17. La norma ISO/IEC 9241-110 [ISO 2006] sustituye a ISO/IEC 9241-10 y trata sobre los principios ergonómicos generales que aplican al diseño de los diálogos persona-computador. En cuatro de las sub-características de la ISO/IEC 9126-1 (adecuación, capacidad para ser aprendido, capacidad para ser operado y adaptabilidad), dicha norma redirige a la norma ISO/IEC 9241-10 como extensión de la misma. La norma presenta como principios importantes para el diseño y evaluación de los principios de diálogo: adaptación a la tarea, carácter auto-descriptivo, control por el usuario, conformidad con las expectativas del usuario, tolerancia a errores, aptitud para la individualización y facilidad de aprendizaje. Las partes 12 a 17 de la serie 9241 proporcionan guías para el diseño de interfaces de usuario y se centran fundamentalmente en los diseñadores de aplicaciones.

Los estándares ISO sobre calidad y evaluación del software están siendo revisados en la actualidad como proceso de organización lógica y unificada que cubra, tanto la especificación de los requisitos de calidad, como la evaluación de calidad del software. Esta revisión está dando lugar a la segunda generación de estándares sobre calidad de productos software. La serie ISO 250nn se ha denominado SQuaRE (Software product Quality

[15] La usabilidad es una de las características de primer nivel definidas en ISO/IEC 9126.

Requirements and Evaluation) y pretende coordinar y enriquecer los estándares actuales sobre especificación de requisitos de calidad del software, su medida y evaluación. En la Figura 20 puede verse el proceso de transición y sus relaciones entre las normas actuales y la serie ISO 250nn. Como puede verse en la Figura 20, SQuaRE se ha dividido en las siguientes divisiones o secciones:

- ISO/IEC 2500n – División de Gestión de Calidad,
- ISO/IEC 2501n - División de Modelo de Calidad,
- ISO/IEC 2502n - División de Medidas de Calidad,
- ISO/IEC 2503n - División de Requisitos de Calidad, y
- ISO/IEC 2504n - División de Evaluación de Calidad.

ISO/IEC 25050 a 25099 están reservados para extensiones de SQuaRE.

En la actualidad sólo existen cuatro normas de la serie 250nn publicadas como estándares. Una de ellas (ISO/IEC 25051:2006) está especialmente vinculada a este trabajo de investigación al describir los requisitos de calidad para productos COTS y, por tanto, lo trataremos en la sección 2.2.

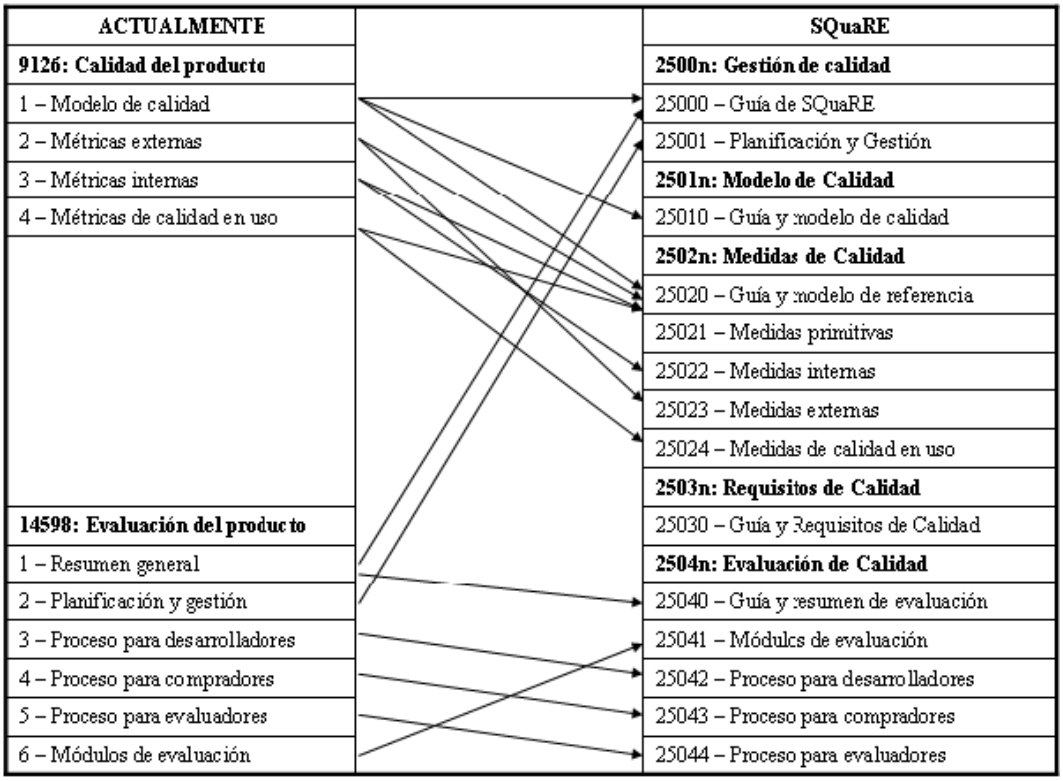


Figura 20. Relaciones y proceso de transición entre las series ISO/IEC 9126 e ISO/IEC 14598 a la serie de normas SQuaRE

Desde su publicación el ISO 9126 ha recibido diversas críticas: [Botella, Burgués 2002], [Veenendaal y Trienekens 1997] y [Punter, Solingen 1997] entre otros; que han propuesto, al mismo tiempo, mejoras del modelo con el fin de solventarlas. Todos ellos coinciden en la falta de precisión de la citada norma. El que uno de los objetivos principales de la norma haya sido proporcionar las características y subcaracterísicas válidas para todo tipo de software, le ha conferido a la misma un carácter abstracto debido a lo general que pretende ser. Así, en su afán por abarcar todas las posibilidades, la norma ISO 9126, no especifica si es posible descomponer, por ejemplo, sub-características en otras sub-características o alternativamente en atributos. Por otra parte, aunque sí menciona que ciertos atributos pueden influir en varias sub-características en el mismo modelo, no dice cómo tratarlo con respecto al total. Además, el modelo definido en ISO 9126 sólo considera factores técnicos. Sin embargo, en el caso de los paquetes software finales, por ejemplo, los factores de tipo organizatico tienen una importancia muy alta en el momento de la selección de uno de ellos.

Ejemplos de este tipo de factores son el coste, el tipo de licencia, los servicios ofrecidos por el proveedor de software (mantenimiento, formación, etc.), el directorio de clientes o la cuota de mercado del producto. Existen numerosos alcances, como por ejemplo [Kontio 1996], [Kontio, Caldiera 1996], [Maiden, Ncube 1997], [Burgués, Franch 2000], [Kunda y Brooks 2000], [Lawlis, Mark 2001], [Ochs, Pfahl 2001], [Morisio y Torchiano 2002], [Comella-Dorda, Dean 2004], [Carvallo y Franch 2006], que destacan la importancia de incluir estas características no técnicas u organizativas en la evaluación de los productos COTS.

Por su parte, el estándar IEEE 1061 [IEEE 1998] no fija características o factores, ni sub-características, ni métricas, sólo en su Anexo A (informativo), a modo de ejemplo, define características y subcaracterísicas. Las características o factores definidos ahí coinciden significativamente en su contenido con la dada en el ISO 9126, manifestándose sólo algunas diferencias en las sub-características informadas (o subfactores), para alguna de las características. El ISO 9126 ha sido el que se ha establecido como modelo de calidad base en la comunidad científica y, por ello, se ha considerado más apropiado seguir las indicaciones y terminología de dicha norma.

2.3.3. Requisitos para la calidad de productos COTS e instrucciones de verificación

Tal como hemos visto en la sección anterior, en la norma ISO/IEC 9126 se especifican los requisitos de calidad para productos software, tanto para identificar los requisitos del software o los objetivos de diseño y pruebas software de productos en desarrollo, como para la evaluación para la aceptación de un producto software completado. Los productos COTS tienen sus propias características por lo que es necesario especificar sus propios requisitos de calidad. La norma ISO/IEC 25051 [ISO 2005g] (antes ISO/IEC 12119 [ISO 1994]) establece los requisitos para este tipo de productos, así como las instrucciones para verificar la adecuación del sistema a dichos requisitos. Más concretamente, la norma ISO/IEC 25051 establece:

- Requisitos de calidad para productos COTS. La norma establece requisitos a cumplir para:
    - la descripción del producto: es decir, qué información debe estar contenida en la misma;
    - la documentación de usuario: describe los requisitos de calidad que debe cumplir;
    - el software: se verifica que se cumple todo lo expuesto en la descripción del producto y en la documentación de usuario.
- Requisitos para la documentación de productos COTS. Se establecen los requisitos que debe cumplir toda la documentación suministrada con el producto software para que sea posible verificar la calidad del mismo.
- Instrucciones de cómo comprobar el cumplimiento de los requisitos para productos COTS de acuerdo a los dos puntos anteriores.

En relación con esta tesis, la norma es útil para verificar que lo expuesto en la documentación del producto es cierto y también proporciona las instrucciones necesarias para comprobar la calidad de la documentación proporcionada con el software. Sin embargo, el cumplimiento de esta norma con respecto a toda la información que establece como necesaria para realizar la evaluación del producto con respecto a la misma, según nuestra experiencia en evaluaciones de productos de seguridad informática [Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b] es realmente baja, es decir, muy pocos fabricantes de software proporcionan tanta información sobre sus productos.

Por tanto, para la evaluación de productos COTS habrá que tener en cuenta, además de las características de calidad definidas en ISO/IEC 9126, los requisitos de calidad definidos para la documentación de usuario en ISO/IC 25051. También será necesario adaptar la característica “funcionalidad” definida en ISO/IEC 9126 de acuerdo con lo especificado en esta norma en relación a los requisitos de calidad para productos COTS.

2.4. Revisión sistemática de los modelos de calidad orientados a dominio

Tal como comentamos en la sección 2.3.2 dado que nuestro objetivo en la obtención y validación de un modelo de calidad orientado al dominio de los productos COTS de seguridad informática, nos interesa especialmente cómo han sido obtenidos los modelos de este tipo existentes hasta el momento. Si ya existe una metodología para su obtención y validación, ésta podrá ser aplicada a nuestro dominio objetivo. En otro caso, el conocimiento y análisis de lo que otros autores han hecho, nos proporcionará la base para poder definir una metodología sistemática que nos permita definir nuestro modelo.

Con el fin de realizar un análisis exhaustivo del estado del arte en relación a los modelos de calidad orientados a dominio, se ha llevado a cabo una revisión sistemática de los trabajos existentes con el objetivo de identificar, analizar y comparar las propuestas más relevantes. En esta revisión sistemática hemos seguido las directrices proporcionadas por Kitchenham [Kitchenham 2004] que adapta el proceso al área de la ingeniería del software. Además, hemos utilizado la plantilla desarrollada por Biolchini [Biolchini y Gomes 2005] que facilita la planificación y ejecución de las revisiones sistemáticas realizadas en el campo de la ingeniería del software. Kitchenham [Kitchenham 2007] define el concepto de revisión sistemática de la siguiente forma:

> *A systematic literature review (often referred to as a systematic review) is a means of identifying, evaluating and interpreting all available research relevant to a particular research question, or topic area, or phenomenon of interest. Individual studies contributing to a systematic review are called primary studies; a systematic review is a form of secondary study.*

Este tipo de metodología se caracteriza por ser rigurosa, confiable y auditable lo que le confiere un alto valor científico en las revisiones del estado de la cuestión. Las razones para llevar a cabo una revisión sistemática más comunes son la exploración de las publicaciones existentes acerca de un área concreta con el fin de resumir los resultados existentes y encontrar sus fortalezas y debilidades, la identificación de áreas que requieren más investigación o la capacidad para proporcionar un marco para nuevas investigaciones.

Para llevar a cabo la revisión sistemática en modelos de calidad orientados a dominio se siguió el protocolo definido en [Biolchini y Gomes 2005] que pasamos a explicar y aplicar a nuestro caso en las siguientes sub-secciones.

1. Planificación de la revisión

En esta fase se definen los objetivos de investigación y el modo en el que se ejecutará la revisión incluyendo la formulación de las preguntas de investigación y de cómo se llevará a cabo la selección de fuentes y estudios relevantes o primarios.

1.1.Formulación de la pregunta

La finalidad de la formulación de las preguntas de investigación es la definición clara de los objetivos. Se define el foco de la pregunta como la identificación de los trabajos relevantes relacionados con la adaptación del estándar ISO/IEC 9126 a dominios específicos de aplicación. La identificación de estos trabajos nos permitirá analizar los métodos utilizados hasta el momento para el desarrollo de modelos de calidad orientados a dominio. El propósito final es que esta información nos permita averiguar si ya existe algún método sistemático para obtener de forma eficiente dichos modelos o si, por el contrario, es necesario desarrollar alguna metodología con el objetivo de mejorar los procesos actuales.

Formulamos la cuestión inicial de investigación de la siguiente forma:

> *¿Cuáles son los métodos utilizados para la adaptación del modelo de calidad definido en ISO/IEC 9126-1 a dominios de aplicación específicos para la evaluación de la calidad del software de productos finales?*

El resultado esperado al final del proceso de revisión sistemática es la identificación de los trabajos relevantes en los que se ha llevado a cabo una adaptación del estándar a un dominio de aplicación específico. Por tanto, las propuestas obtenidas serán entonces analizadas y comparadas.

Para contestar a esta pregunta se utilizarán las siguientes palabras clave y conceptos:

- Evaluation, assessment, selection, measuring

- Quality Model, ISO 9126
- Development, building, construction, generating

1.2.Selección de fuentes

El objetivo de esta fase es seleccionar las fuentes entre las que se ejecutarán las búsquedas de los estudios relevantes o primarios. El criterio de selección utilizado para evaluar las fuentes se basó en cubrir al máximo las publicaciones en revistas y actas de congresos relevantes en el área de la calidad del software. Desde luego fue requisito indispensable que las fuentes pudieran ser accedidas forma electrónica y que se proporcionase un motor de búsquedas a través de palabras clave. Finalmente, se seleccionaron las siguientes fuentes:

- IEEExplore
- ACM Digital library:
- Citeseer library
- ScienceDirect (Elsevier)
- SpringerLink

1.3. Identificación y selección de los estudios primarios

Una vez definidas las fuentes que se utilizarán en el proceso, es necesario definir los criterios para la selección y evaluación de estudios primarios.

La cadena de búsqueda debe adaptarse al motor de búsqueda usado en cada una de las fuentes seleccionadas antes de llevar a cabo la ejecución de la búsqueda, para luego obtener los resultados iniciales a los que se les aplican los criterios de inclusión y exclusión para, finalmente, obtener los estudios relevantes o primarios. Los criterios de inclusión y exclusión se basarán en la pregunta de investigación enunciada en la sección 1.1. Los criterios de inclusión se aplican primero sobre el título, resumen y palabras clave de cada uno de los trabajos obtenidos con la cadena de búsqueda. Al total de los trabajos obtenidos tras aplicar el criterio de inclusión, se les aplica el de exclusión esta vez sobre el total del texto.

Por tanto, como primer paso, a partir de las palabras clave y la pregunta de investigación se construye la cadena de búsqueda que se utilizarán para la identificación de los trabajos

relevantes. En nuestro caso y tras probar con diferentes combinaciones y criterios de búsqueda, la cadena de búsqueda con la que más resultados relacionados hemos obtenido ha sido la siguiente:

"Quality model" and 9126 and (evaluation OR assessment OR selection OR measuring)

Como criterio de inclusión, seleccionaremos sólo aquellos trabajos que adaptan el estándar ISO/IEC 9126 a un dominio de aplicación específico. Nos interesa conocer la forma en la que dichos modelos se han desarrollados. Como se comentó anteriormente, el criterio de inclusión se aplica al título, resumen y palabras clave de los trabajos obtenidos en cada fuente tras la ejecución de la búsqueda. Por su parte, el criterio de exclusión se aplica a todo el texto. Se han definido como criterios de exclusión que no formarán parte del conjunto de trabajos primarios aquellos trabajos relacionados con:

- El proceso de desarrollo del software puesto que la revisión se centra en la evaluación de productos finales;
- La ingeniería de requisitos, es decir, modelos de calidad definidos a través de la recogida de requisitos de usuarios, ya que nos interesa definir modelos reutilizables por dominio de aplicación;
- Con modelos constructivos [Dromey 1995] o aquellos en los que el modelo se construye desde cero para cada proyecto.

2. Ejecución de la revisión y extracción de la información

Durante la fase de ejecución de la revisión se ejecuta la búsqueda sobre las fuentes seleccionadas y los estudios obtenidos se evalúan de acuerdo a los criterios de inclusión y exclusión. El número de estudios primarios analizados fue de 374, una vez aplicado el criterio de inclusión quedaron 61 trabajos y el número final de estudios primarios, tras la aplicación de los criterios de exclusión y la eliminación y/o agrupación de trabajos redundantes, fueron 25. En la se desglosan los resultados clasificados por fuente. En la Tabla 2 se muestra un resumen del número de resultados obtenidos en cada fase según la fuente.

Fuente	Resultados	Criterio de inclusión [a]	Criterio de exclusión [b]
IEEE Xplore	109	22	14
ACM Digital Library	160	14	3
ScienceDirect	16	5	5
SpringerLink	64	15	9
Citeseer library	25	5	2
TOTAL	374	61	25

Tabla 2. Resultados obtenidos en la fase de ejecución.

Nota: (a) Tras aplicar el criterio de inclusión y eliminar repeticiones
(b) Tras aplicar el criterio de exclusión y eliminar propuestas repetidas.

3. Extracción de la información y resumen de resultados

Una vez obtenidos los estudios primarios se pasa a la fase de extracción de la información relevante. Con el fin de apoyar dicho proceso, nos formulamos las siguientes preguntas:

1. ¿Se sigue algún proceso sistemático para el desarrollo del modelo de calidad?
2. ¿El modelo desarrollado se basa en estándares internacionalmente aceptados (aparte del ISO/IEC 9126), trabajos de investigación relacionados con el dominio aparte de la experiencia propia?
3. ¿El modelo incluye pesos y relaciones?
4. ¿Se ha llevado a cabo algún tipo de validación? En caso afirmativo:
   - ¿tiene rigor científico?
   - ¿en qué ha consistido?

En la Tabla 3 se muestra una tabla comparativa con los estudios primarios obtenidos y las respuestas a las preguntas de investigación más arriba enunciadas.

Referencia	¿Sigue un proceso sistemático de adaptación?	¿Está basado en otros estándares y trabajos?	¿Proporciona pesos?	¿Proporciona relaciones?	¿Ha sido aplicado a la industria?	¿Muestra modelo final?	Nº de factores	¿Sigue algún proceso de validación?
[J. Hall, Hall, *et al.* 2003]		x					>1000	Retroalimentación tras evaluaciones
[Stefan y Florian 2007]	x					x	-	
[Losavio, Matteo, *et al.* 2008]		x	x			x	-	
[Helmut, Benjamin, *et al.* 2008]		x					-	
[Malak, Badri, *et al.* 2004]	x	x					>300	
[Kilsup Lee y Lee 2006]			x		x			
[Stefani y Xenos 2001]			x				32	
[Andreou y Tziakouris 2007]						x	46	Aplicación de la metodología por terceros y recogida de retroalimentación
[Moraga, Calero, *et al.* 2009]		x				x	93	
[Behkamal, Kahani, *et al.* 2009]	x		x				-	
[Losavio, Chirinos, *et al.* 2004]	x						-	
[Olsina y Rossi 2002a]	x		x			x	96	
[Won Jun, Ji Hyeok, *et al.* 2007]			x			x	-	
[Gi oug, Doo yeon, *et al.* 2006]			x			x	-	
[Rodriguez, Harrison, *et al.* 2002]						x	-	
[Perez, Tornes, *et al.* 2008]		x	x			x	44	
[Strahonja 2007]		x					-	
[Alexandre Alvaro, Almeida, *et al.* 2006]							-	
[YeongSeok, JungHyun, *et al.* 2005]		x	x			x	-	
[Jin, Yin, *et al.* 2008]			x				-	

[Yoonjung, Sungwook, *et al.* 2008]				x			34	
[Sangeeta y Hausi 2007]		x					-	
[Carvallo, Franch 2003]	x	x		x	x		102-1832	
[Bertoa y Vallecillo 2002b]	x					x		
[Spriestersbach y Springer 2002]							-	

Tabla 3. Resumen comparativo de estudios primarios extraídos en la revisión sistemática.

Con el fin de unificar el modo de extracción utilizado en todos los trabajos, se utilizaron plantillas para recoger los datos de cada uno de ellos. Las plantillas constaban de una cabecera con información de control (título, fuente, autores y referencia) e información general sobre el contenido del artículo (resumen de la propuesta y evaluación subjetiva de la misma). En esta sección presentamos un breve resumen de cada uno de los estudios primarios y, en la siguiente sección, se hace una breve discusión de acuerdo a la información extraída en las plantillas

- A process for evaluating legal knowledge-based systems based upon the Context Criteria Contingency-guidelines Framework [J. Hall, Hall 2003].

Los autores adaptan el proceso de evaluación definido en ISO/IEC 14598 [ISO 1999a] [ISO 1999a] al dominio de los sistemas basados en el conocimiento en el ámbito legal. Parte de este proceso incluye la definición de un modelo de calidad que denominado CCCF (Criteria, Context, Contingency-guidelines Framework). El modelo propuesto es un catálogo compuesto por más de 1000 criterios que debe adaptarse al producto y características de cada evaluación. Los autores proponen que la adaptación se realice aplicando técnicas de negociación como, por ejemplo, la técnica DELPHI con el fin de tener en cuenta diferentes perspectivas: desarrollador, evaluador, directores, autoridades reguladoras, etc.)

- An Integrated Approach to Quality Modeling [Stefan y Florian 2007].

Los autores proponen un meta-modelo como marco de referencia para la definición de modelos de calidad. Como ejemplo de aplicación presentan un modelo para la característica

ISO 9126 “facilidad de mantenimiento” basado en su experiencia. De nuevo resaltan la necesidad de tener en cuenta los diferentes puntos de vista de los usuarios.

- Web Services Domain Analysis Based on Quality Standards [Losavio, Matteo 2008].

Aunque no siguen un proceso sistemático para el desarrollo del modelo de calidad, los autores lo adaptan de acuerdo a otros estándares y normas relativas a web services de forma justificada. También asignan pesos cualitativos (importancia baja, media o alta) a los criterios a través del consenso de expertos aunque no especifican ningún otro detalle sobre cómo se lleva a cabo el proceso.

- An approach to quality engineering of TTCN-3 test specifications [Helmut, Benjamin 2008]

Los autores adaptan el modelo ISO 9126 al dominio de las especificaciones de pruebas para la evaluación de las características internas de facilidad de mantenimiento. Para ello, según su propia experiencia, eliminan los criterios que no aplican y añaden nuevos relacionados con el dominio. Finalmente, aplican el modelo a un caso de estudio.

- Towards a Multidimensional Model for web-Based Applications Quality Assessment [Malak, Badri 2004].

En esta propuesta los autores desarrollan un catálogo de atributos para las aplicaciones basadas en web a través de un proceso sistemático de adaptación del ISO 9126, los trabajos de otros autores y su propia experiencia, obteniendo como resultado una lista de más de 300 factores. Se presenta un caso de estudio en el que se utiliza GQM (Goals, Questions, Metrics) [Kontio, Caldiera 1996] para desarrollar cuestionarios basados en este catálogo y obtener a través de él los atributos a utilizar.

- A Quantitative Evaluation Model Using the ISO/IEC 9126 Quality Model in the Component Based Development Process [Kilsup Lee y Lee 2006].

Esta propuesta presenta un modelo de calidad adaptado, según la experiencia de los autores, al dominio de los procesos de desarrollo basado en componentes (CBD, Component

Based Development). Además, calculan los pesos o importancia relativa de cada criterio a través de cuestionarios a expertos en el proceso que, posteriormente tratan utilizando el método de agregación AHP (Analytic Hierarchy Process).

- A model for assessing the quality of e-commerce systems [Stefani y Xenos 2001]

Los autores desarrollan un catálogo de criterios partiendo de los definidos en el modelo de calidad ISO 9126 clasificados en 3 niveles según la importancia relativa (alta, media, baja) de los criterios en el dominio de e-commerce. Todo ello basado en su experiencia.

- A quality framework for developing and evaluating original software components [Andreou y Tziakouris 2007]

En este trabajo los autores proponen un modelo obtenido a través de la modificación y refinamiento del modelo ISO 9126 para el dominio de los componentes COTS. El modelo definido tiene un alto grado de detalle, además de que se proporciona el modelo completo lo que supone una gran aportación para el dominio de componentes COTS. Además los autores resaltan la necesidad de reducir al mínimo aceptable los criterios a utilizar en la aplicación práctica del mismo. Para ello, utilizan a un grupo de 20 expertos en desarrollo basado en componentes COTS a los que entregan cuestionarios preguntándoles por la importancia relativa de cada uno de los criterios y utilizan las medias de los resultados obtenidos para la selección de los criterios finales.

- Assessment of portlet quality: Collecting real experience [Moraga, Calero 2009].

En este caso los autores también proporcionan el modelo de calidad completo basado en otros trabajos y su propia experiencia en el dominio del desarrollo basado en componentes y, más concretamente, en portlets. Además, asignan pesos a cada uno de los factores aunque lo hacer según su propio criterio.

- Customizing ISO 9126 quality model for evaluation of B2B [Behkamal, Kahani 2009]

Los autores proponen un modelo para el dominio de las aplicaciones B2B. Una vez adaptado el modelo ISO 9126 al dominio, se utiliza la opinión de 15 expertos para ponderar

la importancia de cada uno de los factores de calidad utilizando (según en la media aritmética de los datos recogidos) y utilizando, para ello, dos puntos de vista: el de los expertos en desarrollo de aplicaciones B2B y el de los usuarios. Aunque el número de expertos utilizado es bajo el estudio está diseñado de forma correcta y se muestran todos los resultados finales obtenidos, así como, la metodología utilizada.

- ISO quality standards for measuring architectures [Losavio, Chirinos 2004]

En este caso se adapta el modelo ISO 9126 al dominio del software de diseño de arquitecturas software. Aunque especifican de forma razonada los cambios realizados a nivel de características, no definen niveles inferiores de la jerarquía y, por tanto, no proporcionan tampoco el modelo de calidad. Tampoco se asignan pesos a los criterios.

- Measuring web Application Quality with webQEM [Olsina y Rossi 2002a]

Los autores definen un proceso de evaluación que incluye un modelo de calidad adaptado a partir de ISO 9126 al dominio de las aplicaciones web. El modelo se obtiene a través de la experiencia y conocimiento de los autores del dominio, al igual que los pesos que se asignan a cada uno de los criterios.

- A Quality Model for Open Source Software Selection [Won Jun, Ji Hyeok 2007].

Los autores definen un modelo de calidad para la selección de software de código abierto. Para ello, seleccionan 23 compañías con experiencia en desarrollo de este tipo de software para que, a través de una encuesta, clasifiquen por orden de importancia (de 1 a 5) cada una de las sub-características quedándose finalmente con aquellas que se habían clasificado con importancia 3 o superior (no especifica el estadístico ni ningún otro detalle sobre la metodología y el análisis usado).

- A Quality Evaluation Technique of RFID Middleware in Ubiquitous Computing [Gi oug, Doo yeon 2006].

En esta propuesta los autores adaptan a partir de su experiencia el modelo ISO 9126 al dominio de RFID (Radio Frequency Identification). También obtienen pesos para cada

criterio utilizando AHP (Analytical Hierarchy Process) aunque no proporcionan ningún dato sobre los participantes en el estudio ni ningún otro detalle del mismo.

- A Generic Model and Tool Support for Assessing and Improving web Processes [Rodriguez, Harrison 2002].

Los autores adaptan el modelo ISO 9126 al dominio de los procesos web basándose en su experiencia mostrando las características y sub-características obtenidas. No se proporcionan pesos.

- MECRAD: Model and Tool for the Technical Quality Evaluation of Software Products in Visual Environment [Perez, Tornes 2008]

Los autores utilizan, además del modelo ISO 9126, otros estándares relacionados con el dominio de las herramientas de entorno visual. Se muestran las características y sub-características finales pero no los atributos. Los pesos se asignan según el criterio propio de los autores.

- The Evaluation Criteria of Workflow Metamodels [Strahonja 2007]

En esta propuesta se define un modelo de calidad para "workflow metamodels" de nuevo basada en la experiencia de los autores. No se proporcionan atributos ni pesos para los criterios.

- A Software Component Quality Model: A Preliminary Evaluation [Alexandre Alvaro, Almeida 2006]

En esta propuesta los autores presentan el modelo CQM (Component Quality Model) compuesto por 46 atributos y obtenido a través de la adaptación del modelo de calidad ISO 9126 al dominio de los componentes COTS usando para ello su experiencia. Además, se lleva a cabo un estudio para averiguar si es posible encontrar, en los componentes que en ese momento existían en el mercado, la información requerida por las métricas propuestas.

- Development of Quality Evaluation Metrics for BPM (Business Process Management) System [YeongSeok, JungHyun 2005].

Los autores en este trabajo se basan en diferentes informes de Gartner[16] acerca de las características principales de los productos de gestión de procesos de negocio para construir el modelo de calidad específico. Para asignar pesos al modelo se basan en su propia experiencia.

- Fuzzy Integrated Evaluation for Measuring Quality of Feature Space-Based Component Model [Jin, Yin 2008]

En esta propuesta los autores toman 3 de las características ISO 9126 y añaden otras 4 relacionadas con el dominio de modelos de componentes orientado al dominio espacial basándose exclusivamente en su propia experiencia. De la misma forma, definen sub-características y las ponderan según su importancia.

- Practical S/W Component Quality Evaluation Model [Yoonjung, Sungwook 2008]

En este trabajo se muestra la adaptación del modelo ISO 9126 al dominio de componentes software. Para ello, adaptan el estándar al dominio según su propia experiencia y utilizan la técnica Delphi con expertos para obtener las relaciones entre características y sub-características de calidad aunque no especifican ningún detalle sobre el método utilizado.

- Quality Criteria and an Analysis Framework for Self-Healing Systems [Sangeeta y Hausi 2007].

En esta propuesta se define un modelo de calidad para el dominio de sistemas que pueden adaptarse de manera automática a su entorno (SHS, Self-Healing Systems). No se definen pesos ni relaciones entre los criterios.

- Using quality models in software package selection [Carvallo, Franch 2003]

Los autores proponen un alcance sistemático para el desarrollo de modelos de calidad orientados a dominio y basados en ISO 9126. En este trabajo se formaliza a través de una metodología la obtención del modelo de calidad. Dicha metodología consiste en 6 pasos: la

[16] Gartnet Group: http://www.gartnet.com

definición del dominio, identificación de sub-características aplicables al mismo, su descomposición en atributos, la descomposición de éstos en atributos atómicos, determinación de relaciones entre criterios y, por último, obtención de métricas. El proceso ha sido aplicado a diferentes dominios como son el de herramientas para workflow [Carvallo, Franch*, et al.* 2004b], servidores de correo [Carvallo, Franch 2003], ERP (Enterprise Resource Planning) [Botella, Burgués*, et al.* 2003], componentes COTS [Carvallo y Franch 2006] o software de gestión de contenidos [Franch, Quer*, et al.* 2008]. Además a partir de este proceso han desarrollado una herramienta para la selección de componentes COTS [Grau, Carvallo*, et al.* 2004, Carvallo Vega, Franch*, et al.* 2007] y un modelo UML para la formalización de un modelo de calidad genérico que representa los conceptos fundamentales de la calidad del software y que facilita el refinamiento en modelos orientados a dominios específicos [Burgués y Franch 2004]. Los modelos orientados a dominio publicados por los autores comprenden entre 102 y 1832 criterios [Carvallo, Franch 2006], por lo que, debido al alto número de criterios, deben luego adaptarse a cada proyecto particular tras la recogida de requisitos. Por tanto, tiene una baja aplicabilidad práctica en procesos de evaluación en los que no se dispone de requisitos de usuario como es el caso, por ejemplo, de las evaluaciones independientes para fabricantes. Además, no proporciona pesos o importancias relativas de cada uno de los criterios ni dispone de ningún tipo de validación externa de los criterios.

- Quality Attributes for COTS Components [Bertoa y Vallecillo 2002b]

En este trabajo los autores proponen un modelo de calidad para componentes COTS. Para ello, se identifican las características y sub-características ISO 9126 que aplican al dominio y luego definen atributos y métricas para medirlos basándose para ello en su experiencia.

- Quality Attributes in Mobile web Application Development [Spriestersbach y Springer 2002]

Los autores adaptan el modelo ISO 9126 al dominio de las aplicaciones web en móviles. Para ello, aplican su propia experiencia en el dominio de aplicación. No definen pesos ni relaciones entre los criterios.

4. Análisis de resultados

Los modelos de calidad orientados a dominio existentes representan una importante aportación en el conocimiento de dominios de aplicación específicos. Aunque este es un punto de inicio fundamental en el desarrollo de este tipo de modelos, a través de la realización de esta revisión sistemática de la literatura relacionada, se han encontrado una serie de deficiencias en el desarrollo de los mismos que resumimos a continuación:

- A pesar de que todos ellos son una adaptación del comúnmente aceptado estándar internacional ISO/IEC 9126 [ISO 2001a], en muchos casos no se tienen en cuenta otros estándares relacionados con el propio dominio, por ejemplo el ISO/IEC 15408 [ISO 2005e] para la sub-característica de seguridad [Andreou y Tziakouris 2007], o incluso con ciertas características de modelo ni tampoco otros trabajos relacionados.
- En muchos casos la adaptación del modelo se realiza sin incluir razonamiento alguno y de forma subjetiva [Morisio, Stamelos*, et al.* 2002], [Olsina y Rossi 2002a], [Rodriguez, Harrison 2002], [Carvallo, Franch 2004b], [Losavio, Chirinos 2004], [Dae-Woo, Hyun-Min*, et al.* 2006], [Gi oug, Doo yeon 2006], [Losavio, Matteo 2008], [Jin, Yin 2008] y [Yoonjung, Sungwook 2008], en otros, se basan en la propia experiencia del investigador. La validación realizada se basa en la aplicación a casos industrial pero sin obtener posteriormente retroalimentación por parte del cliente para conocer, al menos, el grado de satisfacción con la evaluación realizada [Abrahao y Insfran 2006], [Carvallo, Franch 2003] , [Kilsup Lee y Lee 2006] ,[Moraga, Calero 2009]. Los modelos basados en la propia experiencia del investigador, aunque son un interesante punto de partida y tienen un gran valor, tienden a ser subjetivos y difíciles de reutilizar al ser generados para una situación o proyecto concreto.
- En muy pocos casos se proporcionan datos técnicos relativos a los casos de estudio tales como el número total de criterios a medir o el tiempo empleado en la evaluación. En los casos en que se proporciona el número de criterios, estos son elevados, como es el caso, por ejemplo, de los 410 criterios en el modelo de calidad para servidores de correo [Carvallo, Franch 2003] o los más de 300 criterios para el modelo de aplicaciones basadas en web [Malak, Badri 2004]. Este elevado número de criterios hace que, aunque sean muy útiles como punto de partida para la

generación de modelos de calidad, requieran de una optimización para poder ser aplicables en la industria.

- Además, en la mayoría de los casos no se proporciona información sobre la importancia relativa de cada uno de los factores o las relaciones de influencia de unos sobre otros y, cuando se hace, o se obtiene de forma subjetiva, como en [Olsina y Rossi 2002a] o [Moraga, Calero 2009], o sólo se presentan los resultados finales proporcionando pocos datos sobre la técnica empleada, el número y conocimiento o experiencia de los expertos que participaron o cómo se llevo el análisis sobre los datos recogidos. Es el caso de [Yoonjung, Sungwook 2008] donde se utiliza la técnica Delphi con expertos o de [Losavio, Matteo 2008] donde se utiliza el consenso también con expertos; o en [Gi oug, Doo yeon 2006], [Kilsup Lee y Lee 2006] y [Won Jun, Ji Hyeok 2007] donde utilizan encuestas para obtener la importancia relativa de las características pero no proporcionan los detalles sobre el número y conocimiento de los expertos que intervienen en la misma. Tan sólo en [Behkamal, Kahani 2009] proporcionan datos sobre el perfil de los expertos seleccionados e incluyen la necesidad de tener en cuenta diferentes perspectivas de la calidad como la visión del director, desarrollador, usuario, etc. Además, detallan la técnica utilizada para tratar los datos: la comparación por pares a través de AHP (Analytical hierarchy process) para las relaciones. El principal problema de este proceso es que sólo utilizan las encuestas a expertos para obtener los pesos y las relaciones (los criterios no se validan). Al no validarse los criterios tampoco es posible eliminar posible información redundante proporcionada con diferentes criterios y, por supuesto, no se reduce la lista de factores final.
- La obtención del catálogo de atributos no se realiza de forma metódica en la mayoría de los casos. Tan sólo en [Olsina y Rossi 2002a], [Carvallo, Franch 2003, Franch y Carvallo 2003], [Malak, Badri 2004], [Grau, Carvallo 2004], [Losavio, Chirinos 2004], [Stefan y Florian 2007] y [Behkamal, Kahani 2009] se definen procesos sistemáticos para construir modelos de calidad para la obtención de los modelos de calidad. Sin embargo, aunque estos procesos permiten la adaptación razonada y a través del conocimiento del dominio, no incluyen ningún método ni técnica que permita la consolidación ni validación del mismo. De todos estos procesos

adaptaremos a nuestra metodología para la obtención de modelos de calidad orientados a dominio, el definido en [Carvallo, Franch 2003] por haber sido definido de manera formal y haberse aplicado en diferentes proyectos en la industria.

2.5. Conclusiones

En la Tabla 4 se muestra un resumen de cada uno de los trabajos analizados en este capítulo y las aportaciones que se han tenido en cuenta para la definición de una metodología propia que englobe todas las aportaciones de estos trabajos y proponga soluciones a los problemas no resueltos que se han ido enunciando a lo largo de este capítulo.

Método	Aportaciones utilizadas en este trabajo
[Punter, Solingen 1997], [Trienekens, Veenendaal 1997]	• Perfiles de calidad: calidad para cierto producto software. • El perfil de calidad debe incluir las características sobre el propio negocio (organizacionales)
[Punter, Kusters 2004]	• Involucra a todas las partes interesadas. • Importancia relativa de los criterios.
[Colombo y Cervigni 2002]	• Utiliza todos los estándares actuales sobre evaluación de calidad del software en un solo proceso integrándolos a través del modelo de calidad definido.
[Kontio, Caldiera 1996]	• Uso de criterios no técnicos u organizacionales.
[Kunda y Brooks 2000], [Kunda 2003]	• Obtiene una clasificación de criterios no técnicos u organizacionales. • Uso de factores clave (keystone) que permiten filtrar productos que no los verifican.
[Maiden, Ncube 1997, Maiden y Ncube 1998]	• Técnica de filtrado iterativo
[Comella-Dorda, Dean 2002]	• Propuesta de criterios no técnicos u organizacionales. • Filtrado iterativo
[J. Hall, Hall 2003]	• Validación por retroalimentación tras las evaluaciones. • Propone adaptación del modelo mediante técnica DELPHI teniendo en cuenta el criterio de expertos en las diferentes áreas relacionadas.
[Stefan y Florian 2007]	• Proceso sistemático de adaptación del modelo de calidad. • Tienen en cuenta los diferentes puntos de vista de los usuarios.

[Losavio, Matteo 2008]	• Asignan importancia relativa de los criterios a través del consenso de expertos.
[Helmut, Benjamin 2008]	-
[Malak, Badri 2004]	• Proceso sistemático de adaptación del modelo de calidad usando ISO 9126, los trabajos de otros autores y su propia experiencia.
[Kilsup Lee y Lee 2006]	• Importancia relativa de los criterios a través de cuestionarios a expertos. Los datos recogidos se tratan utilizando el método AHP (Analytic Hierarchy Process).
[Stefani y Xenos 2001]	• Importancia relativa de los criterios.
[Andreou y Tziakouris 2007]	• Obtención de un modelo de calidad para componentes COTS adaptado a partir de ISO 9126. • Reducción de factores a través de cuestionarios a expertos y aplicación de media aritmética para obtener importancia relativa. • Aplicación de la metodología por terceros y recogida de retroalimentación
[Moraga, Calero 2009]	-
[Behkamal, Kahani 2009]	• Proceso sistemático de adaptación del modelo de calidad. • Importancia relativa de los criterios a través de cuestionarios a expertos y media aritmética de los datos recogidos. • Uso de diferentes puntos de vista de expertos.
[Losavio, Chirinos 2004]	-
[Olsina y Rossi 2002a]	• Proceso sistemático de adaptación del modelo de calidad. • Importancia relativa de los criterios.
[Won Jun, Ji Hyeok 2007]	• Reducción de los factores a través de encuestas a empresas para conocer la importancia relativa de los criterios. Los datos se analizan usando la media aritmética.
[Gi oug, Doo yeon 2006]	• Importancia relativa de los criterios a través de consulta a expertos.
[Rodriguez, Harrison 2002]	-
[Perez, Tornes 2008]	• Se utilizan los estándares relativos al dominio de aplicación además de los de calidad. • Importancia relativa de los criterios.
[Strahonja 2007]	-

[Alexandre Alvaro, Almeida 2006]	-
[YeongSeok, JungHyun 2005]	• Importancia relativa de los criterios. • Relaciones entre criterios.
[Jin, Yin 2008]	• Importancia relativa de los criterios.
[Yoonjung, Sungwook 2008]	• Técnica Delphi con expertos para obtener las relaciones entre características
[Sangeeta y Hausi 2007]	-
[Carvallo, Franch 2003]	• Proceso sistemático de adaptación del modelo de calidad. • Relaciones entre criterios.
[Bertoa y Vallecillo 2002b]	• Proceso sistemático de adaptación del modelo de calidad.
[Spriestersbach y Springer 2002]	-

Tabla 4. Resumen de las aportaciones de los trabajos analizados

En relación a la evaluación sistemática de los métodos utilizados para obtener modelos de calidad, en resumen, se han examinado 374 artículos encontrados en 2 bases de datos y 3 bibliotecas digitales de artículos en el área de la ingeniería del software. De ellos, tras aplicar los criterios de inclusión y exclusión quedaron un total de 25 artículos relacionados con las preguntas planteadas, de los que sólo el 24% desarrollan el modelo de forma rigurosa y tan sólo el 8% declaran haber llevado a cabo algún tipo de validación sobre los resultados finales obtenidos distinta de aplicar el modelo obtenido por los evaluadores por ellos mismos a un caso de estudio (ya sea un caso real en la industria o no). El tipo de validación llevada a cabo en ese 8% de trabajos, consistió en la recogida de información de retroalimentación por parte del cliente sobre la evaluación llevada a cabo y en uno de ellos el modelo lo aplican terceros (un equipo de evaluadores distinto de los autores del trabajo). Por otra parte, de todos los estudios un 24% de ellos (6 trabajos) utilizan la opinión de expertos en el área para validar alguna parte del proceso (criterios, pesos, relaciones, etc.) y de estos en tan sólo 2 de ellos (el 8% del total) el estudio ha sido correctamente diseñado, detallando el perfil demográfico de los expertos que participan en el estudio, enunciando hipótesis, definiendo la técnica estadística utilizada para tratar los datos recogidos detalladamente, etc.

Dado que el objetivo inicial de esta revisión sistemática era averiguar si existía alguna metodología sistemática que incluyese algún proceso de validación, debemos deducir según los resultados obtenidos que, aunque existen distintos trabajos e ideas que suponen una importante aportación, es necesario profundizar en la definición de una metodología que permita, no sólo obtener a través de estándares y otros trabajos relacionados y utilizando la propia experiencia y conocimiento de los investigadores los criterios a aplicar en el dominio específico, sino también validar de alguna forma que esos criterios son adecuados, conseguir reducirlos obteniendo aquellos que son esenciales, y obtener pesos y relaciones de forma consensuada con otros expertos en el área abarcando las diferentes áreas de conocimiento involucradas.

Para la primera parte del proceso consistente en la obtención de un modelo de calidad inicial (no validado) usaremos los pasos descritos en [Carvallo, Franch 2003] pero, una vez obtenido el catálogo de atributos jerarquizado según características y sub-características, necesitamos priorizarlos y obtener las relaciones existentes entre ellos. Además, si el número de atributos obtenido es muy grande, con el fin de obtener un proceso aplicable a la industria, necesitaremos reducirlos. Esta reducción también puede ayudar a evitar la redundancia a través de la eliminación de atributos altamente correlacionados. Por último, necesitamos también validar que ese catálogo obtenido tiene todos los criterios importantes, es decir, no falta ni sobra ninguno. Desde luego, la decisión de qué criterios son más importantes requiere de un juicio experto con experiencia en una gran variedad de proyectos y diferentes perfiles profesionales. También la validación de la lista de criterios requiere de este tipo de conocimiento. Por tanto, parece claro que tener en cuenta el juicio de un gran número de expertos (tantos como nos sea posible y de diferentes perfiles profesionales) es necesario en este caso. Pero, además, necesitamos encontrar la metodología estadística apropiada para tratar posteriormente estos datos recogidos de expertos. En los trabajos analizados en la revisión sistemática el análisis estadístico se basa en todos los casos en la media aritmética. Sin embargo, el uso de un solo estadístico no proporciona información suficiente para poder seleccionar unos factores a favor de otros, como se verá en el Capítulo 4. Además, para poder aplicar la media es necesario que los datos sigan una distribución normal, lo cual no siempre ocurre.

En la búsqueda de la metodología estadística que nos permitiese validar los criterios definidos y obtener los pesos y las relaciones entre ellos, el análisis que más se ajustaba a nuestros requisitos era el análisis estadístico multivariante y, en particular, el análisis confirmatorio a través de ecuaciones estructurales. Este tipo de análisis se utilizan en el campo de las Ciencias Sociales y ha sido ampliamente utilizado para verificar el nivel de satisfacción en estudios sociológicos, de marketing y también del área de la economía. Es el caso, por ejemplo, del modelo SERVQUAL [Parasuraman, Zeithaml*, et al.*, Parasuraman, Zeithaml*, et al.* 1985]. Este alcance está basado en catálogos de características recogidas y agrupadas por los investigadores o adaptadas desde otros modelos ya validados al dominio en cuestión. Con estas características se construyen cuestionarios y se recogen datos de los usuarios del servicio para, más tarde, aplicar Análisis factorial para reducir el número de variables manteniendo las más importantes. Estos estudios sólo utilizan la reducción de factores para quedarse un número de variables manejable, sin embargo, el análisis confirmatorio permite además obtener la importancia numérica de cada una de las variables definidas, así como, la influencia, también numérica, de unas sobre otras. Por tanto, es posible extender este alcance para, utilizando los estadísticos obtenidos en el análisis confirmatorio, obtener los pesos de los criterios y las relaciones entre ellos.

Capítulo 3. Metodología de desarrollo de modelos de calidad

3.1. Introducción

En el Capítulo 2 se discutió la problemática en el desarrollo de modelos de calidad y en la selección de productos COTS llegando a la conclusión de que, al no disponer de un proceso que permita validar los criterios y obtener la importancia relativa de cada uno de ellos, así como, las relaciones de influencia entre los mismos, era necesario desarrollar un proceso propio para la posterior obtención del modelo de calidad para productos COTS de seguridad. En este capítulo se define el proceso que hemos denominado DuMoD del inglés *Domain-oriented qUality MOdels Development*. Dicho proceso está basado, por una parte en un enfoque teórico a través de la aplicación de normas, estándares y otros trabajos de investigación relacionados tanto con la ingeniería del software como con el dominio de aplicación al que se aplica, y por otra, en un enfoque empírico, a través de la consulta a expertos en las distintas áreas involucradas en la evaluación de productos. Los objetivos del proceso son los siguientes:

1. Obtener un catálogo completo de criterios adaptados al dominio de aplicación específico a través del análisis de las normas y otros trabajos de investigación relacionados.
2. Consensuar, validar y reducir el catálogo utilizando el criterio de expertos en las diferentes áreas de conocimiento involucradas en el proceso de evaluación validando así el modelo teórico y reduciéndolo con el fin de que sea eficiente.
3. Cuantificar la importancia relativa de cada criterio para poder usarlos en las evaluaciones cuantitativas, ya sea para tener definido el peso de cada factor en las evaluaciones en las que no se dispone de requisitos de usuario para poder

obtenerlo (por ejemplo, las evaluaciones de productos para fabricantes), o ya sea para que sirva de base para un posterior refinamiento en su adaptación a cada proyecto específico.

La principal aportación de este proceso es la participación de expertos en el mismo y la aplicación de análisis multivariable a los datos obtenidos para obtener así el modelo final validado por expertos. De esta forma, los criterios obtenidos de forma teórica a través de la aplicación de estándares y otros trabajos de investigación pueden ser consensuados y validados por expertos con múltiples experiencias en las áreas de conocimiento involucradas. Por tanto, la aplicación del proceso proporciona como principal beneficio la obtención de un modelo de calidad validado por expertos de diferentes áreas de conocimiento o perfiles profesionales que puede ser aplicado tanto en el proceso de desarrollo o mejora de productos software de calidad por parte de fabricantes, como en el proceso de evaluación o selección de productos por parte de empresas usuarias del software. Para facilitar una mejor adaptación del modelo según los requisitos de cada proyecto particular, se proporcionan los modelos obtenidos de cada grupo de expertos por área de conocimiento por separado. Este modelo incluye tanto los criterios considerados esenciales por los expertos como su importancia relativa dentro del modelo, lo que permite como mínimo guiar el proceso de adaptación del modelo al proyecto particular.

Este capítulo se estructura de la siguiente forma. En la sección 3.2 se describe el esquema de trabajo seguido para obtener la metodología. Dado que para definir el proceso se ha utilizado información ya descrita en el Capítulo 2, en esta sección se detallará el resto de información utilizada en su desarrollo. Por último, en la sección 3.3 se describen los objetivos, principios fundamentales y requisitos del proceso DuMoD para pasar posteriormente a definir paso por paso cada una de las fases que lo forman.

3.2.Esquema de trabajo

Para obtener la metodología de evaluación de productos COTS de seguridad informática que cumpla con todos las propiedades requeridas, se llevó a cabo un estudio centrado en

encontrar y validar con expertos en el área (investigadores y profesionales) las características más relevantes para este tipo de productos.

El proceso DuMoD que se desarrolla en este capítulo se ha obtenido a partir de:

- la aplicación del análisis realizado del estado del arte visto en el capítulo 3; y
- las lecciones aprendidas de dos estudios en los que se evaluaron productos software finales de seguridad informática.

A continuación se resumen los dos estudios llevados a cabo para la obtención del modelo de calidad desarrollado en este trabajo de tesis doctoral. Puede consultarse la información completa de evaluación en [Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b].

3.2.1. Trabajo previo al desarrollo del modelo. Casos de estudio sobre evaluación de productos COTS de seguridad

Dentro del proyecto de investigación UEM OTRI 2007/02 se llevaron a cabo dos estudios de evaluación, los cuales, motivaron nuestro interés en definir un nuevo método de evaluación más práctico y preciso para productos software finales de seguridad informática y nos ayudaron a obtener y evaluar dicho proceso. Los productos evaluados fueron Cryptosec 2048 de Realsec [Realsec 2008] e ISA Server 2006 de Microsoft [Microsoft 2009]. Un extracto de los informes de evaluación de los productos puede leerse en [Realsec 2007, Villalba y Fernández-Sanz 2007a] y [Villalba y Fernández-Sanz 2007b].

3.2.1.1 Descripción de los casos de estudio

La primera evaluación que se llevó a cabo fue solicitada por Realsec (http://www.realsec.com) quien, tras obtener el nivel EAL3 de seguridad (FIPS 140-2 [NIST 2001][17]) en la evaluación de la seguridad de los módulos criptográficos desarrollados

[17] FIPS 140-2 equivale a EAL3 según el estándar ISO/IEC 15408.

para su producto Cryptosec 2048, tenía el objetivo de mejorar la calidad de su módulo de seguridad hardware (HSM, *Hardware Security Module*) denominado Crytosec 2048. Dicho módulo proporciona soporte a operaciones criptográficas de manera segura. El sistema consiste un módulo criptográfico con las funciones principales de generación, almacenamiento y protección de cifrado criptográfico. Además, también proporciona otras funciones como aceleración hardware para operaciones criptográficas tales como cifrado/descifrado. Una vez certificada la seguridad de Crytosec 2048, el objetivo de Realsec era que se evaluasen otras características como la eficiencia, dado que incluye aceleración hardware de las operaciones criptográficas, o la facilidad de uso, con el fin de mejorar el producto y obtener una valoración independiente del mismo. Un equipo de tres miembros con experiencia en calidad del software, productos de seguridad y procesos de evaluación, llevaron a cabo un proceso de evaluación que duró 8 semanas. Dado que existían métodos estándar de evaluación del software [ISO 1999a] y también diferentes métodos de evaluación para productos COTS), éstos se aplicaron en la medida de lo posible. Para llevar a cabo el proceso fue necesario un contacto continuo con el cliente: el proceso completo requirió 5 reuniones y un total de 202 criterios para medir. Los criterios de evaluación se obtuvieron de la adaptación de una lista mayor de 302 criterios que se adaptó al dominio específico. Dicha lista consistía en una colección de propiedades atómicas técnicas y estratégicas tal como se indica en la sección 4.1.2 . La evaluación solicitada por Realsec fue cualitativa por lo que, tras determinar los evaluadores cuántos de esos criterios eran satisfacía el producto, se entregó al cliente un informe con recomendaciones y los datos y registros de evaluación obtenidos.

En el segundo caso, se evaluaron dos componentes del producto *Microsoft Internet Security and Acceleration (ISA) Server 2006*: *Application Firewall* y *web publication*. El producto acababa de obtener el nivel EAL4+ del ISO/IEC 15408 [ISO 2005e] y el objetivo de la evaluación solicitada era conseguir un informe independiente para poder publicarlo en revistas de difusión. En este caso, el equipo de evaluación tuvo 4 semanas para llevar a cabo la evaluación. Al tener un espacio de tiempo tan corto para llevar conducir la evaluación, se llevó a cabo, una vez consensuado con el cliente, una evaluación basada en listas de comprobación (*checklist)* funcionales. Los criterios de evaluación utilizados en este caso fueron 122 y las reuniones se redujeron a 4. En la primera reunión el cliente fue informado

de la profundidad o exhaustividad de las pruebas de evaluación que se podían aplicar, dado el tiempo disponible para hacer la evaluación, y las implicaciones de las mismas. Además, en el informe cualitativo entregado al cliente, se dejó constancia de todo ello.

Junto con el informe de evaluación se entregó en ambos casos un sencillo cuestionario de satisfacción a los clientes de ambos estudios con el fin obtener retroalimentación para mejorar el proceso. Los resultados de estas encuestas, así como, las lecciones aprendidas de los principales problemas encontrados y las soluciones propuestas a cada uno de estos problemas se resumen a continuación.

3.2.1.2 Análisis de resultados

Aunque los clientes mostraron su satisfacción con nuestras recomendaciones tras la entrega del informe de evaluación, el proceso resultó en algunos aspectos problemático. A continuación se presentan las lecciones aprendidas tras la aplicación de la norma estándar de evaluación de la calidad del software y el análisis del estado del arte realizado en el Capítulo 2.

Lección 1: ISO/IEC 14598-5 [ISO 1998b] trata el proceso sistemático de evaluación software para evaluadores. Aunque este estándar supone un importante soporte como base teórica al proceso de evaluación, hay varios problemas cuando se intenta poner en práctica. Además, según el análisis del estado del arte mostrado en el Capítulo 2, los procesos de evaluación existentes para productos COTS están únicamente centrados en la selección de productos y no en la evaluación de cada producto en particular. Sin embargo, distintas revistas técnicas de difusión informática publican evaluaciones de productos de ambos tipos (productos individuales y comparación entre productos) y tienen una gran influencia sobre su audiencia. Además, existe una demanda por parte de los proveedores software en la evaluación de sus productos para mejorarlos o por motivos de marketing.

Lección 2: Tal como se muestra en el capítulo 2, diferentes experiencias en evaluación de productos COTS (Capítulo 2, sección 2.2.3.1) han mostrado que existen otra clase de requisitos que se deben tener en cuenta. Dichos requisitos son los relacionados con aspectos no técnicos de los productos software tales como el coste, licencia, soporte del fabricante o fiabilidad de los mismos. Existen catálogos genéricos de estas características no técnicas

para productos COTS pero no existe ningún tipo de validación de estas características ni resultados sobre cuales de ellas son relevantes para el dominio de productos finales de seguridad informática y evaluar todas las características existentes no es práctico por la cantidad de esfuerzo y trabajo requerido. Entre las características no técnicas se encuentran las relacionadas con la calidad del servicio proporcionado por el fabricante en relación, por ejemplo, con el mantenimiento o el soporte ofrecido para sus productos. En este sentido, el Ministerio de Industria, Comercio y Turismo de España aprobó en Marzo del 2006 la marca de garantía CAYSER (CAlidad Y SERvicio) presentada por AEDI (Asociación Española de Directores de Informática) [AEDI 2006]. Esta marca de garantía certifica oficialmente la calidad del servicio en general, y el mantenimiento y soporte de calidad del proveedor software en particular. Por tanto, se considera necesario tener en cuenta en el modelo de calidad, tanto los criterios utilizados en CAYSER, como que el proveedor de software disponga de esta u otra certificación oficial sobre la calidad del servicio proporcionada. Por tanto, es necesario incluir las características no técnicas e integrar en la evaluación global del producto las certificaciones y marcas de garantía relacionadas con la calidad del servicio del proveedor.

Lección 3: En la actualidad un gran número de productos software de seguridad han sido certificados de acuerdo al estándar ISO/IEC 15408 [ISO 2005e] obteniendo de esta forma un nivel de confianza en la evaluación (EAL, *Evaluation Assurance Level*). Sin embargo, el estándar solo evalúa la sub-característica "seguridad". No se evalúan otras propiedades críticas del software como, por ejemplo, la usabilidad, la fiabilidad o la eficiencia. Por otra parte, las metodologías existentes no proporcionan un método para integrar el nivel de seguridad EAL en el resto de la evaluación de calidad. Aunque en nuestros casos de estudio no tuvimos que enfrentarnos a este problema dado que ambas evaluaciones eran cualitativas, en el caso de tener que llevar a cabo una evaluación cuantitativa sería necesario disponer de algún modelo de calidad que incluyera la integración de la evaluación de seguridad a través de la certificación si el producto ya dispone de ella. Esto permitiría no tener que evaluar dos veces la sub-característica seguridad mediante dos procesos diferentes facilitando así la reutilización de los resultados y aumentando la eficiencia en los procesos de evaluación. Por tanto, concluimos que el proceso de evaluación debe proporcionar un método para integrar

la evaluación de la seguridad del producto software en la evaluación global de la calidad del software.

Lección 4: Los modelos existentes están especialmente centrados en las propiedades generales de los productos en lugar de en propiedades orientadas al dominio de aplicación. Un modelo de calidad orientado al dominio de aplicación (por ejemplo, al dominio de los productos de seguridad) puede ofrecer una evaluación más exacta dado que las propiedades pueden definirse de una forma más precisa. Además, el uso de modelos predefinidos ahorra tiempo al evitar que se tenga que definir el modelo desde cero cada vez que se realiza una evaluación. Además, como puede observarse en la Tabla 5, el mayor porcentaje de esfuerzo para ambos casos de estudio fue el dedicado a las actividades de definición del modelo de calidad y ejecución de la evaluación. En relación a la definición del modelo de calidad, ésta se llevó a cabo partiendo de un catálogo de atributos previamente obtenido, es decir, no se partió de cero para su definición. Sin embargo el número de atributos del catálogo era muy grande, lo cual, no sólo aumentó el tiempo requerido para adaptarlo al tipo de producto particular, sino también el tiempo necesario para ejecutar la evaluación. Por ello, en nuestro caso, concluimos la necesidad de un modelo de calidad específico y validado para los productos finales software de seguridad informática adaptado a partir de los modelos de calidad estándar y que recoja características no técnicas. Además es necesario aplicar en la validación alguna técnica que permita la reducción del catálogo final de atributos a través de la eliminación de información redundante. En definitiva, para obtener una evaluación más eficiente y exacta, es necesario definir un modelo de calidad específico para productos de seguridad dado que el uso de modelos de calidad genéricos es una actividad que consume mucho tiempo.

Actividad	Crytosec 2048			ISA Server 2006		
	Esfuerzo (hrs)		%	Esfuerzo (hrs)		%
Análisis de los requisitos de evaluación		64			40	
- Requisitos del cliente, reunión inicial, …	4		4%	2		3%
- Análisis de documentación	8		7%	12		19%
- Definición del modelo de calidad	46		40%	22		34%
- Reunión para acordar criterios finales y elaboración del acuerdo final			5%	4		6%
Diseño de la evaluación		6			4	
- Planificación de la evaluación	6		5%	4		6%
Ejecución de la evaluación		32	28%		12	19%
Conclusiones de la evaluación		12			8	
- Informe con resultados finales y recomendaciones	8		7%	3		5%
- Retroalimentación del proceso (aprendizaje de lecciones aprendidas durante el proceso, almacenamiento de criterios y métricas para su reutilización, etc.)	4		4%	5		8%
TOTAL		114			64	

Tabla 5. Distribución del esfuerzo en los casos de estudio previos.

Lección 5: Los clientes de las evaluaciones que llevamos a cabo estuvieron satisfechos con el resultado del informe final de la evaluación pero ¿qué ocurriría si el resultado no fuera el esperado por el cliente? Según la norma ISO/IEC 14598-5 [ISO 1998b], la evaluación de la calidad del software comienza con la solicitud de la evaluación del mismo por parte del solicitante de la evaluación al evaluador. En dicha solicitud el cliente expresa unos requisitos que deberán ser analizados por el evaluador, de forma que finalmente lleguen a un acuerdo sobre la especificación de la evaluación. Sin embargo, con el fin de dejar abiertas todas las posibilidades, la norma no establece cómo debe realizarse este acuerdo. Por otra parte, tal como se discutió en el Capítulo 2 (sección 2.2.2.2), el Proceso W hace alusión a la necesidad de acuerdo entre las partes interesadas en el proceso para evitar que los resultados sean rechazados al final de la misma por no cumplir las expectativas del solicitante de la evaluación pero tampoco proporciona mecanismos para resolver el problema. Por todo ello, estimamos que es importante establecer un acuerdo formal entre el evaluador y el cliente de la evaluación que explique los términos del proceso de evaluación, así como, el nivel de profundidad (o técnica de evaluación) que se utilizará en la misma y las implicaciones derivadas del mismo. Dicho acuerdo formal formaría parte del contrato de evaluación.

Lección 6: El proceso estándar de evaluación requiere un número elevado de reuniones entre solicitante y evaluador. En el caso de que el objetivo de la evaluación sea la selección de un producto COTS para una organización para las que generalmente se trabaja en la propia organización en contacto continuo con el cliente, esto puede ser beneficioso para ambas partes a la vez que no perjudica a la eficiencia del proceso. Sin embargo, cuando se realiza una evaluación en un laboratorio independiente las reuniones pueden suponer un problema con respecto al tiempo disponible para realizar la evaluación y una molestia para el cliente. En los casos de estudio anteriormente descritos, tras entregar el informe de evaluación se entrevistó al solicitante de la evaluación con el fin de obtener retroalimentación que pudiese mejorar el proceso. En ambas evaluaciones, el cliente destacó que el número de reuniones requeridas era demasiado alto. Por su parte, el equipo de evaluación también consideró que las reuniones consumieron demasiado tiempo. Por tanto, reorganizar el proceso con el fin de reducir las reuniones entre solicitante y evaluador para aumentar la eficiencia de los procesos de evaluación.

3.2.1.3 Conclusiones

A pesar de los problemas reportados en la sección anterior los procesos de evaluación se llevaron a cabo con éxito y ambos clientes mostraron su satisfacción con los resultados. Sin embargo el proceso, tal como se muestra en la sección anterior, se puede optimizar. En ambos procesos de evaluación los evaluadores estuvieron de acuerdo en que la cantidad de esfuerzo requerido en el proceso de evaluación fue elevado. Por otra parte, la velocidad con la que cambia la tecnología hace que los proveedores de software requieran cada vez más procesos de evaluación eficientes. Para ello, de nuevo, el contar con un modelo de calidad orientado a productos COTS de seguridad informática sin información redundante y validado nos permitiría obtener una importante reducción en el tiempo total de evaluación.

Como se mostró en la Tabla 5, el mayor problema encontrado fue la gran cantidad de criterios a adaptar y posteriormente a medir y analizar, es por ello que en este trabajo nos centramos en la obtención de un modelo de calidad eficiente dejando para trabajo futuro la mejora del proceso de evaluación. Los problemas aquí reportados junto con otros analizados en el estado del arte (Capítulo 2) se han utilizado, no sólo para la generación de un modelo de calidad para productos COTS de seguridad informática, sino también para desarrollar un

método para la obtención de modelos de calidad orientados a dominio validados presentado en este capítulo.

3.3. El proceso DuMoD (Domain-oriented qUality MOdels Development)

La parte fundamental del proceso de evaluación es el modelo de calidad que se utilizará durante el proceso; en definitiva, qué características del software se van a medir. En la mayor parte de la literatura consultada y tal como se muestra en la sección 2.4 de este trabajo, las características se definen basándose en la propia experiencia del equipo de evaluadores. Dada la importancia del modelo en el resultado final de evaluación, el proceso definido en esta sección, se caracteriza por la validación de las características obtenidas utilizando para ello el conocimiento de expertos de diferentes áreas de conocimiento relacionadas tanto con el proceso de evaluación como con dominio de seguridad informática. Para ello, se siguió un proceso de 4 fases que se resume en la Figura 21.

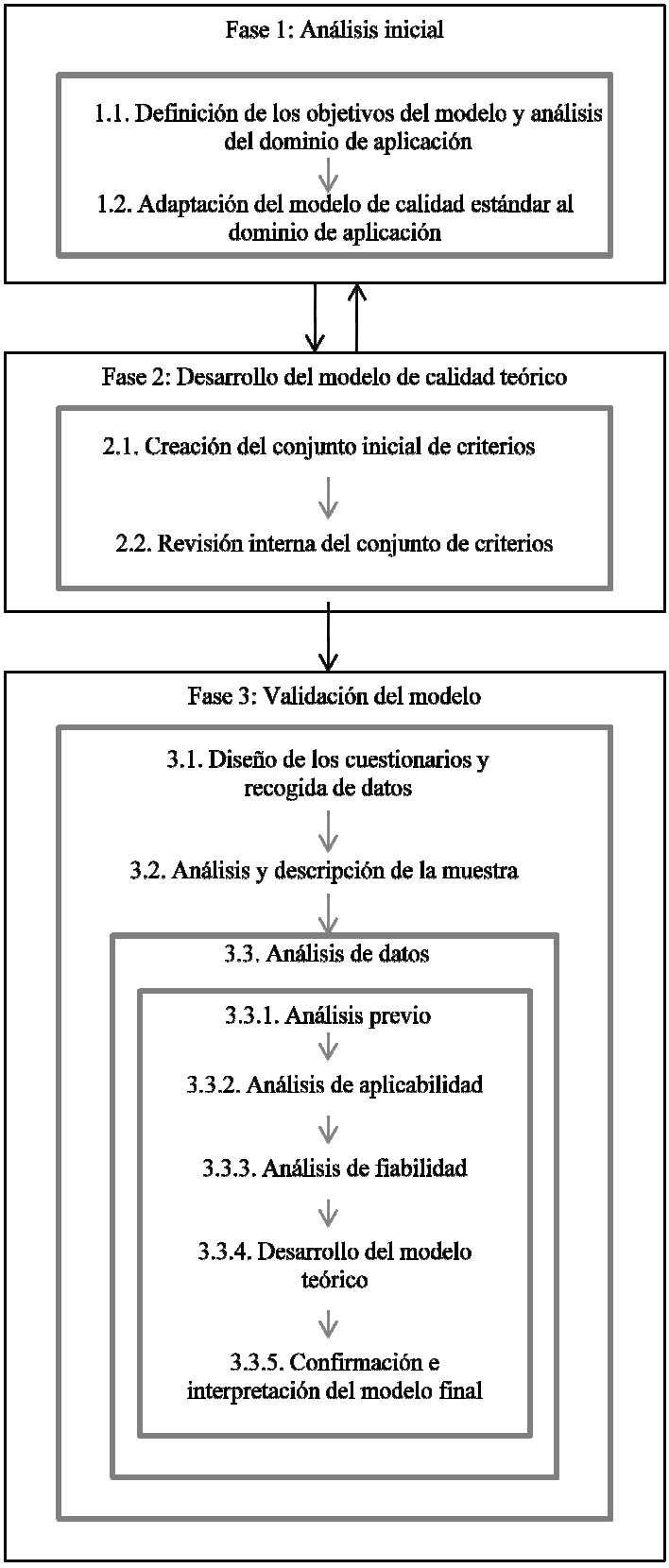


Figura 21. Fases seguidas en el proceso DuMoD

3.3.1. Objetivos perseguidos con el proceso

Basándonos en el estado del arte, el estado de la práctica, las lecciones aprendidas en los casos de estudio presentados en la sección 3.2.1 y los objetivos de esta tesis, las contribuciones perseguidas con el desarrollo del proceso DuMoD son las siguientes:

- Mejora en la calidad de los productos software COTS. El disponer de un modelo de calidad, obtenido de forma sistemática, y validado por expertos en las áreas de conocimiento relacionadas, puede facilitar el desarrollo de productos de calidad a los fabricantes de los mismos al poder centrarse en las características que los expertos consideran más importantes en el dominio de aplicación específico.
- Mejora en la efectividad y eficiencia de los procesos de selección de productos COTS:
    - al permitir a los evaluadores centrarse en las características más importantes en la evaluación de los productos COTS de un dominio de aplicación específico, y
    - al proporcionar los pesos y relaciones entre los criterios como base para un posterior refinamiento.
- Disponer de un proceso que permita la evaluación de productos software por parte de laboratorios independientes cuando el usuario del producto(o la organización cliente) no está definido a priori. En este caso, no existen requisitos de usuario que permitan definir cuales son las características que debe verificar el producto. Tampoco se dispone de la importancia relativa de cada una de ellas lo que dificulta los procesos de evaluación cuantitativos.

3.3.2. Principios fundamentales del proceso

A continuación se describen los principios fundamentales en los que se basa el proceso DuMoD derivados del análisis del estado del arte (Capítulo 2) y de la experiencia en evaluación de productos COTS (sección 3.2.1):

- Tiene en cuenta dos enfoques:

1. el académico que pretende dar una base científica al proceso a través del uso de estándares y trabajos científicos relacionados;
2. el profesional o aplicado a la industria a través de la consulta a profesionales expertos.

- Además, se tienen en cuenta los distintos puntos de vista en la evaluación del dominio, consultando no sólo a expertos en el dominio de aplicación específico para el que se construye el modelo (por ejemplo, servicios de correo electrónico, productos de protección de la seguridad en red, etc.) sino que también se tienen en cuenta otros perfiles por su relación con el objetivo para el que esté destinado el modelo de calidad. Por ejemplo, los directivos de tecnologías de la información o los ingenieros de calidad del software como parte fundamental el los procesos de selección y adquisición de productos cuando éste sea el objetivo del modelo de calidad a desarrollar.
- Por otra parte, existen procesos de evaluación en los que no se dispone de un usuario final definido para el uso del producto. Es el caso, por ejemplo, de las evaluaciones que se realizan para fabricantes de productos ya sea con la intención de mejorar los mismos antes de su comercialización o por motivos de marketing. También es el caso de las revistas especializadas en las que se realizan informes de evaluación sobre productos dirigidos a un público que puede tener unos requisitos muy diferentes. En ambos casos, no se dispone de requisitos funcionales ni de calidad: sin embargo, sigue siendo necesario saber cuales son las características más importantes a evaluar, cual es su peso en la evaluación final del producto y cuales son las relaciones de influencia entre los distintos criterios a evaluar.

3.3.3. Requisitos de aplicabilidad

A pesar de que el uso en la evaluación de productos de un modelo de calidad reutilizable en un dominio de aplicación específico construido a través de la aplicación del proceso DuMoD es menos arriesgado y más práctico (una vez se tiene el modelo) que llevar a cabo una evaluación ad-hoc, es decir, sin un proceso que seguir, la obtención de dicho modelo supone un esfuerzo extra. Por ello, el uso de dicho proceso se justifica en situaciones de

reutilización repetida por la evaluación de múltiples productos en el dominio específico de aplicación y/o en los casos en los que se requiera un alto rigor en el resultado de la evaluación.

En concreto, el proceso DuMoD requiere de las siguientes características para ser aplicable:

- El método debería aplicarse a un dominio de aplicación de interés general. De esta forma el número de procesos de selección o evaluación que se llevan a cabo será alto, asegurando así la reutilización del modelo. Algunos ejemplos de dominios de aplicación de este tipo son: sistemas ERP, CRM, sistemas de protección de seguridad, etc.
- Es necesaria la colaboración con empresas o expertos independientes que participen en la validación del modelo obtenido a través de la aplicación de estándares, trabajos científicos relacionados y la propia experiencia de los evaluadores en el dominio de aplicación.

3.3.4. Definición del proceso DuMoD

El proceso DuMoD consta de 4 fases que se encuentran esquematizadas para una mejor comprensión en la Figura 21. Además, en la Tabla 6 se muestra un resumen de las fases y tareas clave de cada una de ellas, así como, las entradas y salidas que en las siguientes secciones se describen más en detalle.

ENTRADA	FASE	TAREAS CLAVE	SALIDA
Documentación técnica relacionada con el dominio de aplicación	Definición de los objetivos del modelo y análisis del dominio de aplicación (sección 3.3.4.1)	Especificar objetivos para los que el modelo de calidad obtenido será aplicado. Identificar el dominio de aplicación para el que se desarrollará el modelo.	Definición de objetivos y de dominio de aplicación.
Dominio de aplicación definido, ISO 9126-1, publicaciones y experiencia en el dominio de aplicación.	Adaptación del modelo de calidad estándar al dominio de aplicación (sección 3.3.4.1)	A partir del modelo de calidad estándar ISO 9126-1, definir el modelo de alto nivel (características y sub-características) aplicables al dominio de aplicación anteriormente definido. Identificar también razonadamente posibles características o sub-características no definidas en el estándar pero necesarias en el dominio.	Modelo de calidad de alto nivel (características y sub-características) adaptado al dominio.
Estándares de calidad del software, estándares y otras publicaciones relacionadas con el dominio de aplicación, experiencia en el dominio.	Creación del conjunto inicial de criterios (sección 3.3.4.2)	Utilizando toda la documentación relacionada generar un catálogo con todos los factores aplicables al modelo.	Catálogo inicial de factores adaptados al dominio.
Experiencia en el dominio y en evaluación del software.	Revisión interna del conjunto inicial de criterios (sección 3.3.4.2)	Realizar una revisión del catálogo de factores obtenido en la fase anterior para eliminar redundancias, factores que no aplican al dominio, reescribir factores mal expresados, etc.	Tablas con catálogos de factores revisados y adaptados al dominio de aplicación
Tablas con catálogos de atributos	Diseño de cuestionarios y recogida de datos (sección 3.3.4.3)	Utilizando el catálogo final de factores obtenido en la fase anterior diseñar los cuestionarios para la recogida de datos de expertos y validarlos a través de pruebas piloto	Cuestionarios
Datos recogidos con los cuestionarios	Análisis y descripción de la muestra y Análisis de datos (sección 3.3.4.3)	Con los datos obtenidos en la fase anterior, aplicar el análisis estadístico de los datos para obtener el modelo o modelos de calidad finales.	Modelos de calidad finales con los factores, sus pesos y las relaciones de influencia cuantificadas.

Tabla 6. Resumen del proceso DuMoD.

3.3.4.1 Fase 1. Análisis inicial

Definición de los objetivos del modelo y análisis del dominio de aplicación

Antes de iniciar el desarrollo del modelo es necesario definir los objetivos para los que el modelo se va a desarrollar, así como, el dominio de aplicación para el que se aplicará el proceso. Entre los objetivos más comunes se encuentran la evaluación y el desarrollo de productos COTS. La definición de objetivos determinará posteriormente el área de conocimiento de los expertos que deberán intervenir en el proceso de recogida de datos. Por otra parte, el dominio de aplicación debería ser lo suficientemente general para que el modelo pueda ser reutilizado para diferentes productos pero lo suficientemente concreto para que los productos incluidos en él tengan características de calidad comunes.

La definición del dominio puede complicarse debido a la falta de terminología estándar en el área de los productos software COTS y a la ausencia de una taxonomía de clasificación común entre los fabricantes de productos software. Al no existir acuerdo entre los fabricantes de software, debe analizarse cuidadosamente el dominio para evitar posibles conflictos posteriores.

Son ejemplos de dominios de aplicación, por ejemplo: el servicio de correo, los sistemas ERPs, los sistemas datawarehouse, los sistemas para la protección de la seguridad física, etc.

Adaptación del modelo de calidad al dominio de aplicación

Una vez definido el dominio de aplicación, el modelo de calidad internacionalmente reconocido ISO 9126-1 [ISO 2001a] debe adaptarse a dicho dominio. El estándar ISO 9126-1 debería utilizarse a no ser que, durante el análisis del dominio, se encuentre una buena razón para no hacerlo.

Para adaptar el modelo de calidad estándar al dominio de aplicación antes definido, es necesario aplicar la experiencia de los evaluadores y realizar un análisis preliminar del dominio de aplicación a través, tanto de la revisión de los trabajos publicados relativos al dominio, como de especificaciones y pruebas de productos. Posteriormente las características de primer y segundo nivel definidas en ISO 9126-1 son adaptadas al dominio de forma justificada por si fuera necesario modificar posteriormente el modelo. En este paso

podrían definirse nuevas características, refinar la definición de las existentes o incluso eliminar alguna de las existentes.

Es requisito imprescindible que los evaluadores tengan conocimientos y experiencia en el uso de productos en el área o dominio de aplicación específico para el que se va a construir el modelo. En caso de que no sea así, deberán incorporarse al equipo expertos en el área.

3.3.4.2 Fase 2. Desarrollo del modelo de calidad teórico

Creación del conjunto inicial de criterios

Las características y sub-características definidas en la fase anterior proporcionan una visión abstracta global del dominio de aplicación. El paso siguiente consiste en descomponer los conceptos abstractos en otros más concretos, los atributos de calidad. Un atributo de calidad es una propiedad del software que puede medirse a través de una métrica. La norma ISO 9126-1 diferencia entre atributos internos y externos. Dado que nos centramos en los productos COTS y, dado que en ellos el código no está disponible, para el proceso DuMoD únicamente consideraremos los atributos externos. Tal como menciona el estándar, no es posible desde un punto de vista práctico medir todos los atributos relativos a un producto software, pero sí es posible crear una lista con los más importantes. Por otra parte, un atributo puede aparecer en más de una sub-característica del modelo. Por ejemplo, el tiempo de respuesta de un sistema formaría parte de la característica eficiencia. Sin embargo, un mal tiempo de respuesta afecta directamente al tiempo de operación del usuario que forma parte de la característica usabilidad. Por ello, es importante considerar las relaciones entre los diferentes atributos, sub-características y características del modelo de forma que, en evaluaciones cuantitativas, no se considere una mayor influencia de un criterio por su relación de influencia con otro u otros criterios.

Para obtener el conjunto inicial de criterios es necesario tener en cuenta no sólo el estándar ISO 9126-1 sino también:

- el resto de estándares relativos a la evaluación de la calidad del software como ISO 9126-2 [ISO 2003a]. ISO 9241-110 [ISO 2006], ISO 9241-11 [ISO 1998a], ISO 25051[ISO 2005g];
- estándares relativos al dominio de aplicación si los hubiera;

- trabajos de investigación relacionados como publicaciones científicas, libros, informes técnicos e incluso publicaciones profesionales específicas;
- la propia experiencia en el dominio de aplicación o la colaboración directa con expertos en el dominio que puedan aportar ese conocimiento.

Por otra parte, como ya se mencionó en el Capítulo 2 en la evaluación de productos COTS hay otro tipo de criterios no técnicos que influyen directamente en la selección de dichos productos, por lo que deben tenerse en cuenta en la definición del modelo. Se trata de los factores de tipo organizativo o factores no técnicos como el coste del producto, el tipo de licencia, la formación o el soporte ofrecido por el proveedor. Por tanto, debe extenderse el modelo ISO 9126 añadiendo dichos factores. En el Capítulo 3 se muestra un caso de estudio con la aplicación del proceso DuMoD y en él se proporciona un catálogo de factores no técnicos obtenido a través del análisis de los trabajos relacionados existentes que puede servir de base para su uso en otros dominios de aplicación.

Finalmente, como resultado de esta revisión puede ocurrir que sea necesario revisar también el modelo inicial retrocediendo para ello a la fase 1.

Revisión interna del modelo preliminar

El conjunto de criterios obtenidos en la fase anterior debe ser reducido y optimizado con el fin de obtener un modelo práctico. Para ello, el primer paso a seguir es revisar todos los criterios con el fin de:

- identificar y combinar criterios duplicados;
- identificar y eliminar criterios que no aplican al dominio de estudio; y
- reescribir criterios ambiguos.

La revisión se llevará a cabo por parte del grupo de evaluadores y, opcionalmente, expertos en el dominio de aplicación para el que se está desarrollando el modelo de calidad, Ingenieros de calidad del software, analistas y desarrolladores con experiencia en el área de aplicación, directivos y gestores en el área de las Tecnologías de la información.

Es requisito imprescindible la verificación de que los criterios son medibles, aplicables y prácticos por medio de su aplicación a casos reales de evaluación de productos. También es recomendable recoger información de retroalimentación del cliente a la finalización del proceso de evaluación con el fin de comprobar la satisfacción con los resultados de evaluación y, por tanto, la validez de los criterios utilizados.

La salida obtenida en esta fase consistirá en tablas que contengan al menos: el nombre del atributo, su descripción y la fuente o fuentes de las que se obtuvo.

3.3.4.3 Fase 3: Validación del modelo

Diseño de los cuestionarios y recogida de datos

La revisión externa de los criterios obtenidos en las fases anteriores se lleva a cabo con dos objetivos:

1. Reducir el catálogo de atributos obtenido. Como se comentó anteriormente y, según la norma ISO 9126, no es práctico medir todos las propiedades del software, por lo que, es necesario reducir los criterios obtenidos con el fin de obtener un modelo eficiente.
2. Validar los factores obtenidos a través de estándares y documentación científica por los evaluadores con profesionales expertos en distintas áreas mostrando así la aplicabilidad empírica de los factores teóricamente extraídos y su utlidad.

Con el fin de lograr estos objetivos, durante esta fase se lleva a cabo una revisión externa a través de un estudio con expertos (investigadores y profesionales en activo) en las distintas áreas de conocimiento involucradas. Para determinar dichas áreas es fundamental tener en cuenta los objetivos del modelo de calidad definidos en la fase 1.1. Por ejemplo, no es lo mismo que el modelo de calidad vaya orientado al desarrollo de productos COTS, en cuyo caso habría que tener en cuenta el criterio de los usuarios finales, que su objetivo sea el de evaluación de productos para su adquisición, ya que en este caso sería importante tener en cuenta el criterio de los directores TIC.

Para llevar a cabo dicha revisión se desarrollan cuestionarios basados en el modelo preliminar obtenido en la fase anterior con el fin de llegar al mayor número de expertos

posible. Para el diseño de los cuestionarios en esta fase se recomienda aplicar las directrices dadas en [García M 1993] y en [Koyanl, Balley*, et al.* 2006]. Basándonos en dichas normas y en el análisis del estado del arte realizado en el Capítulo 2 se han obtenido una serie de guías para apoyar el proceso de construcción de los cuestionarios que se detallan a continuación:

1. Definir una hipótesis general. Dicha hipótesis se verificará una vez recogidos los datos a través del tratamiento estadístico de los mismos. Definir la hipótesis a priori es un requisito necesario dado que la herramienta se construirá en base a la hipótesis definida.

2. Diferentes trabajos relacionados con el desarrollo de modelos de calidad [Behkamal, Kahani 2009], [Buglione y Abran 1999], [J. Hall, Hall 2003] y [Stefan y Florian 2007] destacan la importancia de tener en cuenta diferentes perspectivas de los expertos relacionados con procesos de evaluación, así como, con el dominio de aplicación. Para incluir estas distintas perspectivas se crean cuestionarios diferentes según las áreas de conocimiento a contemplar18.

3. Debe contemplarse el tamaño muestral de forma que quede adecuadamente representada la variedad de componentes en la muestra.

4. Debe incluirse una sección inicial en la que se explique el estudio y se soliciten datos demográficos de control que permitan antes de comenzar el análisis conocer la población muestral y descartar datos que no cumplan el perfil mínimo determinado (por ejemplo, que tengan una experiencia mínima o sean de un perfil profesional determinado).

[18] Estas área dependerán del dominio específico y el alcance del modelo de calidad (por ejemplo, puede interesar que intervengan programadores y analistas en el caso de que el modelo de calidad vaya destinado a su uso en el desarrollo del productos en el dominio de aplicación específico y, sin embargo, no interesar si el objetivo del modelo es sólo la evaluación de productos).

5. Se deben realizar pruebas para determinar el tipo de pregunta que se utilizará en los cuestionarios. En nuestro caso, como se explicará en el Capítulo 4, seleccionamos preguntas de respuesta múltiple con opciones de estimación (las alternativas se encuentran graduadas en intensidad) y con escala tipo Likert dado que son fácilmente procesables, suponen menos esfuerzo para los encuestados que otros tipos de escalas y porque son las más extendidas. También se recomienda utilizar una pregunta final abierta con el fin de recoger posibles características que los expertos pudieran considerar como no incluidas en los cuestionarios u otros comentarios.

6. Las variables en el estudio serán las características de 2º y 3er nivel. Las preguntas deberán formularse lo suficientemente concretas para poder discriminar unas de otras pero a su vez generales para no obtener un número demasiado grande de preguntas por el riesgo de no ser contestadas.

7. Con la intención de poder llegar a una mayor población muestral se recomienda utilizar preguntas claras, concisas y cortas (no más de 25 palabras), al contrario que los cuestionarios tipo SUMI [Kirakowski y Corbett 1993] en los que la intención en que los completen sólo unos pocos usuarios a los que se les proporciona mucha información. En este caso, como los cuestionarios van dirigidos a expertos en el área de conocimiento, se pueden reducir las frases (poniéndolas a nivel técnico) sin perder por ello información relevante para el experto. Además, dado que las preguntas se dividen por grupos de conocimiento, éstas se pueden personalizar al lenguaje técnico de cada uno de ellos. También es importante poner especial cuidado en la formulación de las mismas evitando, por ejemplo, usar negaciones o no obligando a recurrir a la memoria (por ejemplo, relacionando una pregunta con otra que apareció anteriormente).

8. Antes de publicar los cuestionarios, debe realizarse una prueba piloto para la validación de los mismos en la que, además de las preguntas del cuestionario, se pida retroalimentación sobre el proceso.

9. Por último, dado que el objetivo es llegar al mayor número de expertos posible, es importante disponer de un método automatizado de recogida de datos y posterior

tratamiento estadístico de los mismos. Además, es importante que los expertos dispongan de un método sencillo para introducir los datos (por ejemplo, que el cuestionario pueda ser accedido a través de una dirección en internet que puede llegarles por correo electrónico de forma que sólo tienen que hacer click en el enlace y tienen acceso al estudio).

Para la recogida de datos deben seleccionarse los expertos a los que se dará acceso a los cuestionarios. Para su divulgación pueden utilizarse asociaciones especializadas, contactos de profesionales, congresos, *workshops*, etc. relacionados con el dominio de aplicación.

Análisis y descripción de la muestra

Antes de cerrar el estudio y comenzar el análisis de los datos recogidos, se debe asegurar que los datos almacenados suponen una muestra poblacional fiable y confirmar que los perfiles de los encuestados que participaron en el estudio son los requeridos. Para ello, es necesario calcular el error muestral y hacer una análisis de los datos demográficos recogidos para descartar aquellos cuyo perfil no coincide con el esperado (por ejemplo, no tiene suficientes años de experiencia en el dominio de aplicación).

Análisis de datos

Para llevar a cabo la validación del modelo conceptual obtenido en la fase II se debe aplicar análisis multivariante sobre los datos recogidos en el estudio. Concretamente se utilizarán las técnicas de análisis factorial exploratorio, análisis de ecuaciones estructurales y análisis factorial confirmatorio. Dado que se han demostrado los beneficios del uso del análisis factorial como un complemento a la teoría en la cuantificación de las relaciones en el modelo de medida [Gerbin y Hamilton 1996], se utilizará el análisis factorial exploratorio para validar y corregir en caso necesario el modelo teórico. Una vez obtenido el modelo teórico final se construye el diagrama de relaciones causales para, posteriormente, transformarlo en un conjunto de ecuaciones estructurales. La modelización de ecuaciones estructurales nos permite obtener la matriz de correlaciones que se utiliza como entrada para la estimación del modelo. Una vez obtenido dicho modelo, se evalúan los criterios de calidad del ajuste para, finalmente, si estos son satisfactorios, se pasa a la interpretación del modelo final.

Como notación, las variables del modelo serán los criterios o atributos obtenidos en la fase II y las dimensiones serán las sub-características.

Antes de comenzar la aplicación del análisis multivariante deben formularse las hipótesis que se contrastarán. Las hipótesis deben describir qué variables concretas se agrupan bajo cada dimensión. Para ello, nos basamos en el modelo teórico obtenido en la fase II. Por tanto, obtendremos una hipótesis por dimensión o sub-característica definida para las que contrastaremos su estructura con el análisis multivariante. Para contrastar este tipo de hipótesis, es decir, demostrar hasta qué punto los datos recogidos se ajustan a la estructura teórica esperada, se requiere de una prueba de hipótesis formalizada que se obtiene a través de un análisis confirmatorio [Hair, Tatham*, et al.* 1998].

La Figura 22 resume las etapas del proceso de investigación seguido en la fase 3 del proceso DuMoD (esta figura representa una extensión de la fase 3 de la Figura 21 que mostraba el proceso completo). Así mismo, también se indican las técnicas estadísticas aplicadas en la fase de validación y las aplicaciones informáticas utilizadas en el proceso de análisis: SPSS 16.0.1 [SPSS Inc. 1990] y LISREL 8.80 [Joreskog y Sorbom 1988, Joreskog y Sorbom 1993], ambos en su versión para Windows.

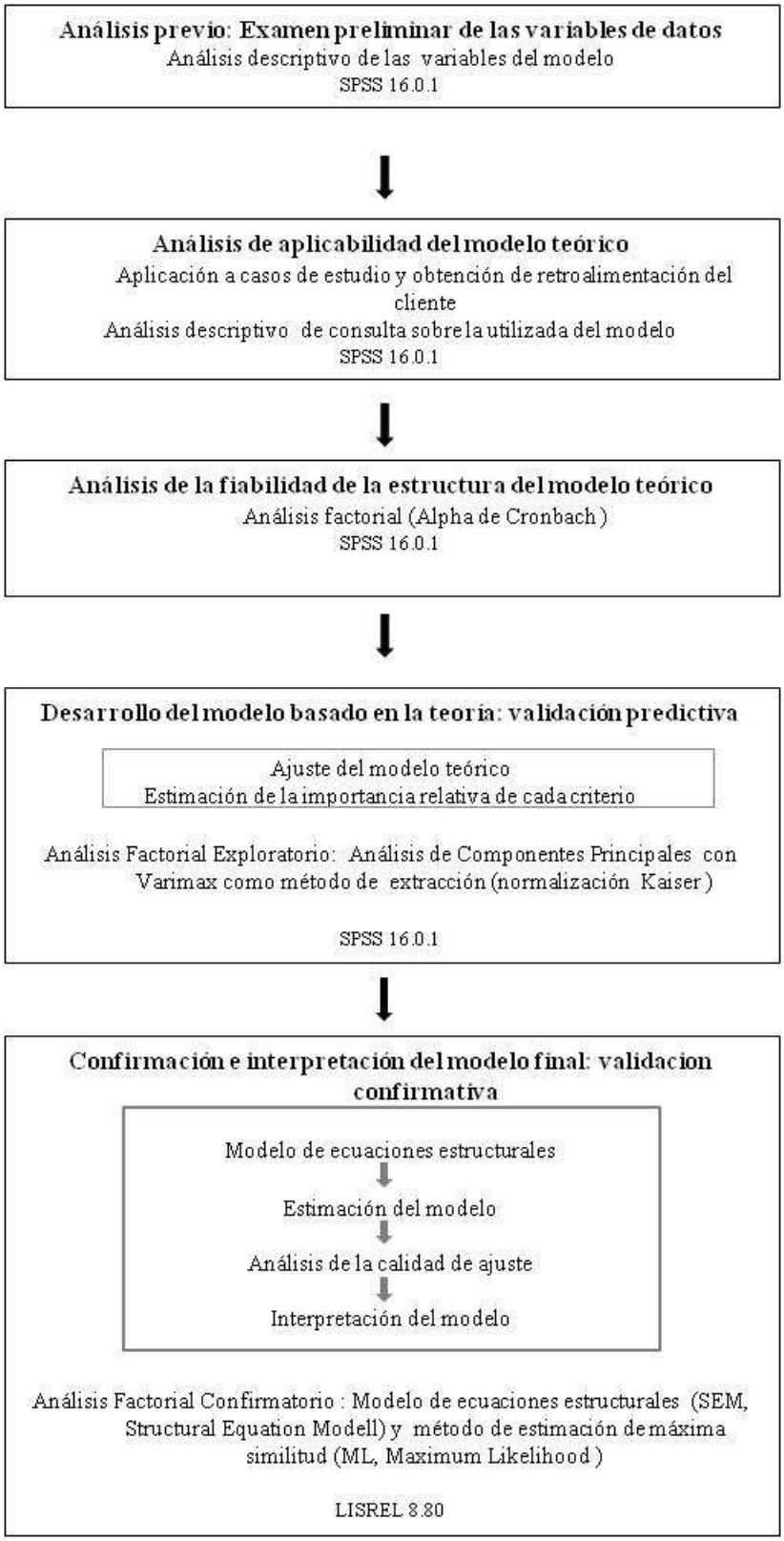


Figura 22. Etapas en el desarrollo de la investigación empírica

La primera etapa es preliminar y consiste en un análisis exploratorio de los datos que supone un primer acercamiento a los mismos con el objeto de conocer su naturaleza, además de permitirnos eliminar posibles errores producidos durante la recogida de datos. Las siguientes etapas constituyen el análisis empírico propiamente dicho. El objetivo principal de estas etapas es la reducción del error de medida del modelo planteado. Esta reducción se alcanza a través del aumento de la fiabilidad y la validez de la medida [Hair, Tatham 1998]. Además, se desarrollan mediciones multivariantes (también llamadas medidas compuestas): se unen diversas variables en una medida compuesta para representar un concepto. Esto se ajusta a la perfección a nuestro modelo teórico basado en la norma ISO 9126. Por ejemplo, para medir la eficiencia se utilizan las siguientes variables: tiempo de respuesta, capacidad de procesamiento, consumo de recursos y escalabilidad. Las mediciones multivariantes tienen la ventaja de reducir el error de medida al reducir la desconfianza sobre una única respuesta. Por otra parte, permite representar los múltiples aspectos de un concepto (variables) en una única medida (dimensión o variable latente[19]) con el mínimo solapamiento. Es decir, reduce la redundancia de la información asociada a ese concepto. De ese modo, podemos representar un concepto con múltiples facetas para reducir posteriormente la complejidad del modelo y la redundancia introducida por las múltiples variables con la mínima pérdida de información.

Para la evaluación del modelo teórico obtenido en la fase II se confirmarán la aplicabilidad, la fiabilidad y la validez del modelo:

1. La aplicabilidad del modelo es la capacidad del mismo para ser aplicado a casos reales en la industria. Para verificar la aplicabilidad del modelo se deben aplicar los criterios obtenidos a casos de estudio para verificar que son medibles y, de ser posible, obtener información de retroalimentación al cliente de la evaluación para conocer su grado de satisfacción con los resultados obtenidos; y se debe consultar a

[19] En el análisis factorial exploratorio las características bajo las que se agrupan las variables se denominan factores pero, dado que este término se utiliza en los modelos de calidad también, para no llevar a equívoco se utilizará la terminología del análisis factorial confirmatorio (dimensión o variable latente).

profesionales expertos en el dominio de aplicación específico para verificar que los criterios obtenidos de forma teórica son las propiedades principales del producto.

2. La validez es el grado en el que un conjunto de medidas representa con precisión el concepto de interés [Hair, Tatham 1998]. Existen diferentes formas de validación[Peter 1981]:
    a. Validación de contenido. En primer lugar, se ha de tener en cuenta la definición conceptual que es el punto de partida para construir un modelo de medida. La definición conceptual especifica las bases teóricas del modelo definiendo el concepto que se está representando en términos aplicables al contexto de la investigación. En este sentido, la validación de contenido (también llamada aparente) es la evaluación de la correspondencia de las variables seleccionadas con su definición conceptual. Para llevar a cabo la validación de contenido los ítems que forman el modelo deben ser seleccionados a través de la investigación académica del concepto global (en este caso la calidad del software) y de cada dimensión o variables latente (características ISO 9126 en este caso), además del uso de evaluaciones de expertos y contrastes previos con subpoblaciones. Según las fases I y II del proceso DuMoD, el modelo teórico se obtiene a través de investigación académica de los conceptos estudiados, así como, en los contrastes previos con subpoblaciones descritos en la fase III (pruebas piloto del estudio). Por otra parte, las evaluaciones de expertos forman parte de esta fase, garantizando así la validez de contenido del modelo propuesto.
    b. Validación convergente o fiabilidad. La fiabilidad es el grado en el que la variable observada (atributo) mide el valor verdadero y está libre de error. Es, por tanto, lo contrario del error de medida y permite obtener el grado de consistencia de la escala. Para medir la fiabilidad se valora el grado en el cual dos medidas del mismo concepto están correlacionadas. Se trata, por tanto, de verificar que existen altas correlaciones entre las variables que miden una misma dimensión. Este análisis se realiza a través del estudio del coeficiente

α de Cronbach[20] y las correlaciones ítem-total (correlación de la variable con la puntuación de la suma total de todas las variables de la dimensión).

c. Validación discriminante. Es el grado en el cual dos conceptos conceptualmente parecidos difieren. De nuevo, el contraste empírico es la correlación pero esta vez entre las conceptualmente diferentes dimensiones. En este caso el análisis que se realiza es un análisis factorial exploratorio.
d. Validación nomológica. Confirma si el modelo final demuestra las relaciones cuya existencia se deriva de la teoría y/o investigación previa. El análisis que se lleva a cabo para verificar la validez nomológica es un análisis factorial confirmatorio basado en ecuaciones estructurales que cuantifica tanto las relaciones entre dimensiones como los pesos de cada variable sobre la dimensión a la que pertenece. El análisis factorial proporciona, por tanto, la base empírica para la valoración de la estructura subyacente de los datos y permite obtener el modelo potencial para crear las medidas compuestas y seleccionar un subconjunto de variables para hacer estimaciones de los factores mismos (puntuaciones de factores comúnmente denominadas cargas factoriales).

Para una mejor comprensión del proceso de evaluación del modelo de calidad llevado a cabo durante el proceso DuMoD, en la Tabla 7 se muestra un resumen de los tipos de evaluación utilizados junto con las técnicas estadísticas aplicadas en cada evaluación y la fase del proceso en la que se aplican.

En relación a las etapas del desarrollo de la investigación empírica descritas al inicio del capítulo en la Figura 22, la etapa de "Análisis de la fiabilidad de la estructura del modelo" se corresponde con la prueba de validez convergente; mientras que en la etapa de "Desarrollo del modelo" se trata la validación discriminante y, por último, la etapa de

[20] El α de Cronbach es la medida más ampliamente usada pero tiene como inconveniente su alta dependencia lineal con el número de variables utilizados para medir un concepto. Por ello, es necesario imponer requisitos más restrictivos para este tipo de modelos [Peter 1981], de ahí que se tengan en cuenta también las correlaciones ítem-total.

"Confirmación e interpretación del modelo final" correspondería con la prueba de validez nomológica.

Tipo de evaluación	Descripción	Técnica usada	Fase
Aplicabilidad del modelo	Capacidad de ser aplicado y útil en entornos reales	Aplicación a casos de estudio reales, encuesta a profesionales	2.2
Validación de contenido	Grado en el que las variables representan el concepto deseado	Modelo teórico basado en normas y estudios de investigación, evaluaciones de expertos y contrastes previos con subpoblaciones	1.2, 2.1.
Validación convergente o fiabilidad	Grado en el que la variable mide el valor verdadero	Coeficiente α de Cronbach y las correlaciones ítem-total	3.3.2
Validación discriminante	Grado en el que dos conceptos conceptualmente parecidos difieren	Análisis factorial exploratorio	3.3.3
Validación nomológica	Grado en el que el modelo final demuestra las relaciones cuya existencia se deriva de la teoría y/o investigación previa	Análisis factorial confirmatorio	3.3.4

Tabla 7. Resumen de las técnicas de evaluación del modelo llevadas a cabo en el proceso DuMoD

A continuación pasamos a describir en detalle cada una de las actividades de las que consta la fase 3.3 del proceso.

Análisis previo: examen preliminar de las variables de datos

En primer lugar se resume el análisis descriptivo efectuado para cada una de las escalas de medida del modelo teórico. Para ello, en primer lugar se calculan los diagramas de caja con el fin de descubrir datos atípicos, permitiéndonos así, eliminar posibles errores producidos durante la recogida de datos. Después, para cada variable se calculan los siguientes estadísticos descriptivos: media, mediana, moda y desviación típica. Por último, con el fin de obtener información sobre la distribución de los datos, se examinan también los histogramas y se verifica la normalidad o no de los datos.

Análisis de la aplicabilidad del modelo teórico

Para verificar que el modelo es aplicable al entorno profesional, se llevan a cabo dos acciones:

1. Los criterios obtenidos en la fase 2 son aplicados en casos de estudio reales para verificar que son medibles y que son prácticos en el entorno empresarial. Además, si es posible, sería deseable obtener información de retroalimentación del cliente de la evaluación para conocer su satisfacción con los resultados de la misma.
2. Se consulta a profesionales expertos en activo en el dominio de aplicación específico para determinar si un modelo así les resultaría de utilidad en su empresa. De esta forma es posible determinar si los atributos obtenidos de forma teórica son adecuados en el entorno profesional.

Análisis de la fiabilidad de la estructura del modelo teórico

Con el fin de determinar la fiabilidad de las escalas de medida utilizadas, se realiza un primer análisis individual de las mismas. Como ya se mencionó anteriormente, la fiabilidad es el grado en el que la variable observada (atributo o sub-característica) mide el valor verdadero y está libre de error. Por tanto, el objetivo de esta actividad será maximizar la fiabilidad de las escalas de medida utilizadas, o lo que es lo mismo minimizar el error de medida, permitiéndonos así obtener al final del proceso de validación un modelo fiable. Para ello, las variables se agrupan en sus características (comúnmente denominadas dimensiones) y, para cada una de éstas, se examinan el coeficiente α de Cronbach y las correlaciones

ítem-total (correlación de la variable con la puntuación de la suma total de todas las variables de la dimensión), descartando las variables cuya eliminación mejora la fiabilidad de la escala de medida. Las variables eliminadas son, por tanto, aquellas que reducen la consistencia de la escala (coeficiente α de Cronbach), o tienen una insuficiente relación con el concepto que se está midiendo (correlaciones ítem-total). Todo ello determinado por el hecho de no superar el nivel mínimo definido para los coeficientes utilizados. Estos mínimos son el 0,70 para el α de Cronbach [J. Nunnally 1978, Flynn, Curran*, et al.* 2002, Thomsen 2002] y el 0,50 para la correlación ítem-total [Hair, Tatham 1998].

Desarrollo del modelo basado en la teoría: validación predictiva

Antes de llevar a cabo el análisis factorial es necesario verificar si se cumplen los supuestos básicos subyacentes del análisis factorial exploratorio. Para ello, se deben considerar los siguientes indicadores:

1. Contraste de esfericidad de Barlett. Prueba estadística para saber si existen correlaciones entre variables
2. Índice KMO (Kaiser-Meyer-Olkin). Mide la correlación existente entre las variables, una vez eliminada la influencia que las restantes variables ejercen sobre ellas. Un KMO próximo a 0 indica una baja o nula correlación entre las variables y, por tanto, desaconseja el análisis factorial. Los valores de corte comúnmente aceptados son [Kaiser 1974]:
    a. KMO > 0.5 debería ser aceptado;
    b. KMO entre 0.7 y 0.8 se considera un buen valor;
    c. KMO > 0.8 es meritorio.
3. Determinante de la matriz de correlación. Determinantes de la matriz de correlación cercanos a cero sin llegar a ser cero (matriz no singular)[21] confirman que las variables están correlacionadas.

[21] Cuando la matriz es no singular no pueden hacerse ciertos cálculos del análisis factorial que requieren de la matriz inversa.

El análisis factorial exploratorio permite identificar la estructura subyacente de las relaciones. Para realizar esta operación es necesario decidir el método de extracción de las dimensiones que se utilizará. En este caso nos interesa utilizar análisis de componentes principales frente al análisis factorial común, dado que uno de los objetivos principales es la reducción de las dimensiones originales, y precisamente el análisis de componentes principales permite resumir la mayoría de la información original (varianza) en una cantidad mínima de dimensiones [Hair, Tatham 1998]. Por otra parte, los criterios utilizados para decidir cuántas dimensiones extraer en cada caso son el criterio de porcentaje de la varianza y el criterio de los valores propios (más conocido como *"eigenvalues"*). El criterio de porcentaje de la varianza se basa en obtener un porcentaje acumulado especificado de la varianza total extraída tal que se asegure que las componentes o dimensiones derivadas explican al menos una cantidad especificada de la varianza que suele fijarse como satisfactoria cuando alcanza, al menos, el 60% de la varianza total [Hair, Tatham 1998]. El criterio de los valores propios consiste en incluir tantas dimensiones como sean necesarias para representar de forma adecuada cada una de las variables originales [Tabachnick y Fidell 2006]. De esta forma, nos aseguramos de eliminar la redundancia conservando toda la información original. La matriz de factores obtenida tras la ejecución del análisis factorial exploratorio, contiene cargas factoriales[22] para cada variable sobre cada dimensión ordenadas según su importancia (varianza) sobre dicha dimensión. Se considera que las cargas superiores a ± 0,32 están en el nivel mínimo mientras que las cargas mayores a ± 0,50 son significativas [Tabachnick y Fidell 2006].

Tras obtener los componentes a conservar, se deben interpretar con el fin de obtener la solución factorial definitiva. Para ello, se rotan con el fin de obtener un patrón más simple y teóricamente más significativo que facilite la interpretación conceptual de las dimensiones extraídas. En el caso del proceso DuMoD se ha seleccionado como método de rotación ortogonal: VARIMAX. El motivo de seleccionar un método ortogonal frente a los oblicuos es que éstos son los recomendados cuando el objetivo principal es la reducción del número

[22] Las cargas factoriales son las correlaciones de cada variable y el componente o dimensión e indican el grado de correspondencia entre ellos haciendo a una variable con mayor carga representativa del componente.

de variables originales a un conjunto de variables incorrelacionadas para un uso posterior en el análisis factorial confirmatorio [Hair, Tatham 1998]. Entre los métodos ortogonales se ha seleccionado VARIMAX por ser el más robusto [Hair, Tatham 1998].

Confirmación e interpretación del modelo final: validacion confirmativa

El análisis factorial exploratorio permite explorar las relaciones entre variables agrupándolas en componentes o dimensiones subyacentes que hemos identificado conceptualmente según las ponderaciones de dichas variables sobre la componente en cuestión. El siguiente paso sería la confirmación de los modelos obtenidos. El modelo de medida de ecuaciones estructurales (a partir de ahora SEM del inglés *Structural Equation Modeling*) proporciona la transición entre el análisis factorial exploratorio y confirmatorio a través de una serie de ecuaciones de regresión múltiple distintas pero interrelacionadas mediante la especificación del modelo estructural. SEM, a diferencia del análisis factorial exploratorio, permite controlar el error de medida y realizar un test estadístico de calidad del ajuste para las solución confirmatoria propuesta, lo cual, permite una validación nomológica del modelo. SEM es la única técnica que permite examinar simultáneamente una serie de relaciones de dependencia múltiples y cruzadas que constituyen un modelo a gran escala, capaz de representar conceptos no observados y tener en cuenta el error de medida en el proceso de estimación [Hair, Tatham 1998]. Para llevar a cabo este análisis se han definido una serie de etapas que pasamos a describir a continuación:

Etapa 1. Obtención del modelo de ecuaciones estructurales. Después de desarrollar el modelo teórico, éste debe especificarse en términos formales. Para ello, a continuación se plantean las ecuaciones estructurales correspondientes a cada uno de los modelos a validar. Al menos se obtendrán ecuaciones estructurales para los modelos de factores técnicos y no técnicos.

Etapa 2. Estimación del modelo. Para la estimación del modelo se utilizará:

- el método de estimación de Máxima Verosimilitud Robusto cuando se haya detectado no normalidad de los datos en la fase 3.3.1[Byrne 1994];
- el método de Máxima Probabilidad puede utilizarse cuando los datos son normales [Hu y Bentler 1999].

Etapa 3. Análisis de la calidad del ajuste. Para verificar que los modelos obtenidos son representaciones adecuadas del conjunto completo de relaciones causales, se analiza el ajuste global del modelo. Para ello, de acuerdo a diferentes estudios [Bollen 1989], [Bentler 1990], [Hair, Tatham 1998], [Hu y Bentler 1999] y [Schermelleh-Engel y Moosbrugger 2003] se han utilizado los índices que se muestran a continuación pertenecientes a los 3 tipos de medidas existentes de calidad de ajuste[23]:

1. Las medidas de ajuste absoluto utilizadas son:
    a. El estadístico chi-cuadrado (χ^2 de Satorra-Bentler) cuyo valor se verifica según las tablas estadísticas.
    b. El error de aproximación cuadrático medio o RMSEA (Root Mean Square Error Of Approximation). Se considera un valor bueno de ajuste cuando es menor que 0,08 [Hu y Bentler 1995].
2. Las medidas de ajuste incremental miden el modelo en comparación al modelo nulo. Los índices utilizados son:
    a. Índice de ajuste normado NFI (Normed Fit Index) o índice de Bentler Bonett;
    b. Índice de ajuste no normado NNFI (Non-Normed Fit Index) o TLI (Tucker-Lewis index)
3. Las medidas de ajuste de parsimonia evalúan la parsimonia del modelo propuesto mediante la evaluación del ajuste del mismo frente al número de coeficientes estimados necesarios para conseguir ese nivel de ajuste. Los índices utilizados en este caso son:
    a. χ^2 normada (ratio entre el valor de χ^2 y los grados de libertad) cuyo valor recomendado es entre 1 y 2 [Bentler 1990] y [Hu y Bentler 1999].
    b. Índice de ajuste relativo RFI (Relative Fit Index)
    c. Índice de ajuste comparativo CFI (Comparative Fit Index);

[23] Sólo se han utilizado las medidas que no se ven afectadas por el tamaño de la muestra.

Cuanto más cercano sea el valor de CFI, NFI y NNFI a 1, mejor es el ajuste. Los valores superiores a 0,90 se consideran muy buenos [Hu y Bentler 1999].

Medida de ajuste	Valores de corte
χ^2 (df)	-
S- χ^2	>1, <2
RMSEA	< 0,08
CFI	> 0,9
NFI	> 0,9
NNFI	> 0,9

Tabla 8. Valores de corte para las medidas de ajuste principales

Etapa 4. Interpretación final del modelo. Para finalizar, el modelo o modelos obtenidos deben interpretarse y obtenerse las conclusiones a partir de los resultados obtenidos.

Capítulo 4. Evaluación del proceso y validación

En este capítulo se detallan los resultados obtenidos en la fase de investigación cuantitativa. El método ha sido evaluado a través de su aplicación al caso de estudio del dominio de aplicación de productos COTS de seguridad informática. El caso de estudio, así como, los datos recogidos a partir de un amplio número de expertos en las diferentes área relacionadas tanto con la ingeniería del software como con el dominio de aplicación en cuestión, proporcionan datos sobre la aplicabilidad del proceso y la utilidad de disponer de modelos de calidad como los obtenidos con el proceso DuMoD.

Este capítulo está organizado como sigue. La sección 4.1 describe la aplicación de la metodología DuMoD al dominio de aplicación de los productos de seguridad de TI con el fin de obtener los modelos de calidad adaptados a dicho dominio. Para ello, se han recogido datos de expertos con el fin de obtener un modelo consensuado y aplicable en distintas situaciones de proyectos de evaluación. También se muestra en esta sección la utilidad, aplicabilidad e importancia del proceso a través de nuevo de la consulta a expertos. Por último, la sección 4.2 muestra el resumen de los resultados obtenidos y las conclusiones del capítulo.

4.1.Caso de estudio: productos de seguridad de TI

A lo largo de esta sección se aplicarán cada una de las fases descritas en el capítulo anterior al dominio de productos de Sesguridad TI mostrando así la aplicabilidad del proceso a un dominio de aplicación concreto.

4.1.1. Fase 1. Análisis inicial

4.1.1.1 Definición de los objetivos y análisis del dominio de aplicación

El objetivo del modelo o modelos de calidad desarrollado a partir de este proceso, es su utilización en la evaluación de productos COTS de seguridad informática. Los motivos de esta evaluación podrán ser diversos, incluyendo entre los mismos, la evaluación cualitativa o cuantitativa de productos ya sea para fabricantes que desean una evaluación independiente de sus productos con el fin de mejorarlos, ya sea para la selección de un producto entre varios para su adquisición.

Dentro de la categoría de productos COTS en este trabajo se considerará el sub-dominio de aplicación de los productos de protección o seguridad informática en red. Según Daniel E. Geer [Geer 2005]:

> *"A product is a security product when it has sentient opponents"*

Es decir, si un producto no tiene oponentes que deliberadamente quieran causar un daño en el software, entonces el producto no es un producto de seguridad. Según esta definición, un producto de seguridad es cualquier producto para el cual ganar acceso a él pueda proporcionar algún beneficio al intruso o bien algún daño a su oponente. Por tanto, esta definición es demasiado amplia o poco concreta para permitirnos conceptualizar apropiadamente el dominio de aplicación.

Por otra parte, según Brian Snow [Snow 2005]:

> *"Security products and services should stop malice in the environment from damaging their users"*

Y en la misma línea Wang Lingyu [Lingyu, Anoop*, et al.* 2007]:

> *"In protecting the networks against malicious intrusions, a standard way form measuring network security will bring together users, vendors and labs in specifying, implementing and evaluating network security products"*

Por otra parte, revisando los documentos web de fabricantes de software, se observa que los productos clasificados por los fabricantes como productos de seguridad son aquellos que proporcionan protección a sistemas y datos.

Por tanto, definimos el dominio de productos de seguridad como los productos cuya funcionalidad principal consiste en proteger el entorno de red de amenazas intencionadas[24]. Dado que se trata de productos finales COTS, no se tendrán en consideración dentro de este trabajo de tesis, las funciones de seguridad que se incluyen en muchos productos software tales como bases de datos, sistemas operativos, etc. con el fin de proteger de amenazas externas el propio producto software.

4.1.1.2 Adaptación del modelo de calidad estándar al dominio de aplicación

En la primera fase del estudio se realizó un análisis de las características de primer y segundo nivel definidas en el modelo de calidad internacionalmente reconocido ISO 9126-1 [ISO 2001a]. El objetivo de este análisis fue obtener un modelo adaptado al dominio de seguridad informática a partir del estándar. Tal como se mostró en el capítulo 2, en diferentes publicaciones se han realizado análisis similares con el fin de adaptar el modelo definido en el ISO 9126-1 mostrado en la Figura 23 al dominio de productos COTS. Sin embargo, no existe ningún trabajo en la actualidad para el área específica de la seguridad informática. Por ello, hubo que adaptar el modelo resultante de aplicar los trabajos referentes a productos COTS, al dominio de la seguridad informática. A continuación se resumen las modificaciones realizadas sobre el modelo de calidad definido en el estándar ISO 9126 para adaptarlo al dominio de productos COTS de seguridad:

[24] Una amenaza no intencionada es, por ejemplo, el borrado de uno o varios datos por error humano.

- Las sub-características *adecuación* y *seguridad* de la característica *funcionalidad* no se incluyen dado que ISO/IEC 15408-2 [ISO 2005c] define los atributos genéricos de seguridad, así como, las propiedades funcionales de los productos y proporciona perfiles de protección adaptados a los diferentes tipos de productos de seguridad. Por tanto, sólo tendremos en cuenta que el producto disponga o no de una certificación de seguridad.
- Se ha añadido la sub-característica *Escalabilidad* dentro de la característica eficiencia. Esta sub-característica se encuentra normalmente en las especificaciones técnicas de los productos de seguridad, además, se ha utilizado con anterioridad en otros modelos de calidad orientados a productos COTS como son [Grance, Stevens*, et al.* 2003, Torchiano y Jaccheri 2003, Comella-Dorda, Dean 2004, Pérez y Tornés 2005, Alvaro, Santana de Almeida*, et al.* 2006, Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008].
- La *capacidad para ser analizado y probado* (ambas sub-características de *facilidad de mantenimiento*) se han eliminado. La razón para no tenerlas en cuenta es que los productos COTS tienen propiedades particulares debido a que no se dispone de información sobre su estructura interna, no se dispone del código y, además, los clientes normalmente no están involucrados en el proceso de desarrollo de los mismos. Por tanto, al igual que otros autores [Marco, Letizia*, et al.* 2002, Alvaro, Santana de Almeida 2006], no tendremos en cuenta las sub-características de *capacidad para ser analizado y probado* dado que requieren acceso a la estructura interna del producto.
- Por otra parte, basándonos en nuestra experiencia en la evaluación de productos de seguridad, la sub-característica de *mantenimiento, capacidad para ser cambiado,* se ha sustituido por *capacidad para ser actualizado*, dado que las modificaciones sobre los productos COTS dependen del fabricante que normalmente los libera para que el cliente tenga acceso a la eliminación de defectos del producto y/o mejoras del mismo (añadiendo funcionalidad o mejorando la existente) a través de nuevas versiones o parches. Definimos la

capacidad para ser actualizado como "atributos del software relacionados con su capacidad para mantener el producto sin fallos o vulnerabilidades de seguridad ".

- Por último, la *capacidad para ser reemplazado,* así como, la coexistencia, son propiedades intrínsecas de los productos COTS por lo que, al igual que otros autores en modelos de calidad para productos COTS [Kontio, Caldiera 1996, Pérez y Tornés 2005], no los tendremos en cuenta.

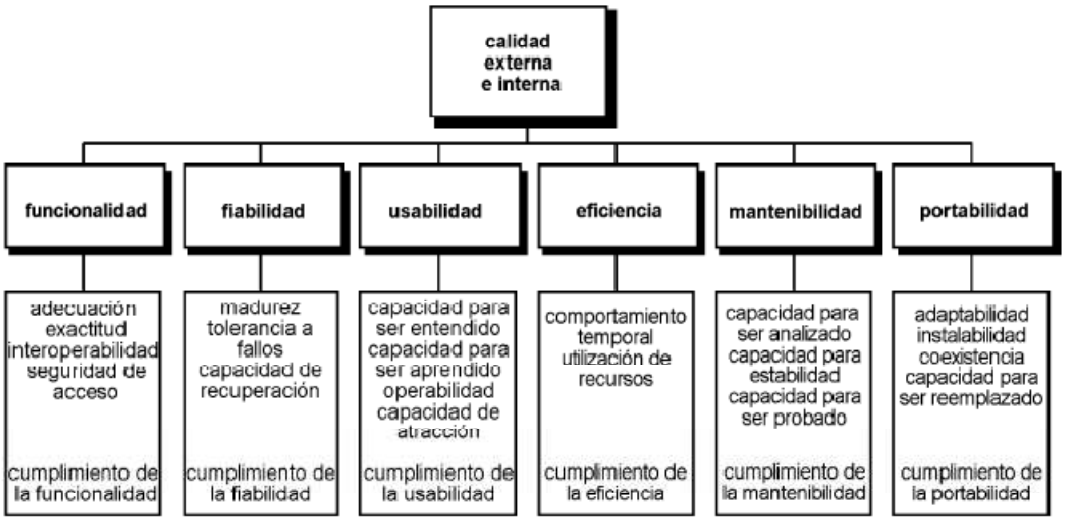


Figura 23. Modelo de calidad para calidad interna y externa [ISO 2001a].

En la Figura 24 se muestra el modelo de calidad a nivel de características y sub-características que se utilizará como modelo preliminar para las siguientes fases del proceso.

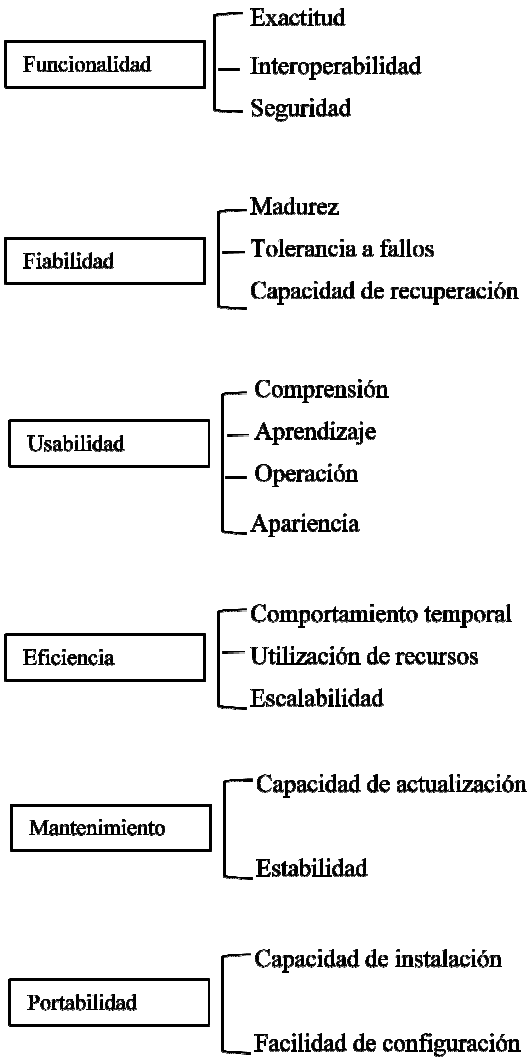


Figura 24. Modelo de calidad preliminar adaptado a productos COTS.

4.1.2. Fase 2. Desarrollo del modelo de calidad teórico

4.1.2.1 Creación del conjunto inicial de criterios

Durante esta fase se recopilaron los criterios de calidad aplicables al dominio de productos finales de seguridad informática. Para ello, se utilizaron los siguientes recursos:

- estándares internacionales de calidad del software y de seguridad: ISO 9126-1[ISO 2001a], ISO 9126-2 [ISO 2003a]. ISO 9241-110 [ISO 2006], ISO 9241-11

[ISO 1998a], ISO 25051 [ISO 2005g], IEEE 1061 [IEEE 1998], ISO 15408-2 [ISO 2005c];

- publicaciones de investigación: [Obeso 2008], [Pérez y Tornés 2005], [Avizienis, Laprie*, et al.* 2004], [Bertoa y Vallecillo 2002a], [Kontio, Caldiera 1996], [Kunda y Brooks 2000], [Ochs, Pfahl 2001], [Morisio y Torchiano 2002], [Comella-Dorda, Dean 2002], [Carvallo y Franch 2006], [Carvallo, Franch 2006];

- libros e informes técnicos: [Fenton y Pfleeger 1997], [Oberndorf, Brownsword*, et al.* 1997], [Grance, Stevens 2003], [Comella-Dorda, Dean 2004], [Koyanl, Balley 2006],

- cuestionarios de usabilidad (*usability test)*: Software Usability Measurement Inventory (SUMI) [Kirakowski y Corbett 1993], IsoMetrics [Gediga, Hamborg*, et al.* 1999], y, por último;

- las lecciones aprendidas por la experiencia acumulada en evaluaciones de calidad de productos de seguridad [Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008].

Como resultado de este esfuerzo se recogieron un total de 302 criterios. Estos criterios fueron los que se utilizaron en los casos de estudio descritos en la sección 3.2.1 del Capítulo 3. Tal como se ha descrito en dicha sección, después de cada evaluación se entrevistó al cliente para recoger la información de retroalimentación en relación a la satisfacción tanto con el proceso de evaluación seguido como con los resultados obtenidos recibiendo en ambos casos resultados satisfactorios. Con lo cual, se considera como satisfactoria la prueba de aplicabilidad de los atributos obtenidos en esta fase.

4.1.2.2 Revisión interna del conjunto de criterios

El conjunto inicial de criterios obtenido en la fase anterior era demasiado grande para ser utilizado de forma eficiente en la evaluación de calidad de productos de seguridad. Por otra parte, es objetivo de este trabajo que el modelo sea validado de forma externa. Por ello, durante esta fase se llevó a cabo un proceso previo a la validación externa de revisión

interna con el fin de identificar y combinar criterios duplicados identificar y reescribir criterios ambiguos.

El resultado final de esta revisión interna fue la reducción del conjunto inicial de criterios a 111. Posteriormente se llevó a cabo una clasificación de los criterios restantes según el modelo de calidad obtenido en la primera fase del estudio. A continuación se presentan los criterios obtenidos una vez adaptados al dominio, revisados y clasificados según el modelo de calidad previamente obtenido (Figura 24).

Sub-característica	Criterio	Descripción	Fuente
Exactitud	Funcionalidades requeridas ejecutada con exactitud	La lista de funcionalidades se obtiene según perfil de protección	[ISO 2001a, ISO 2005e]
Interoperabilidad	Adecuación a las especificaciones Hardware	Las especificaciones hardware (memoria, procesador, disco) requeridas por el producto son adecuadas	[Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008]
	Adecuación a las especificaciones Software	Las especificaciones software requeridas por el producto son adecuadas	[Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008]
	Intercambio de datos	Compatibilidad con otros programas relacionados para intercambiar datos entre sí	[ISO 2003a]
	Uso de interfaces estándares		[Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008]
Seguridad	Certificación de seguridad	Disponer de una certificación de seguridad reconocida que haya sido otorgada por un laboratorio independiente (evaluado según ISO 15408)	[Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008]

Tabla 9. Sub-características y criterios para la característica funcionalidad.

Sub-característica	Criterio	Descripción	Fuente
Madurez	Estabilidad	Fallos en el software	[ISO 2003a]
		Fallos del software no afectan a otros programas	[ISO 2003a]
	Depuración	Tiempo de espera para la corrección de fallos del software (parches, *updates*, *upgrades*)	Adaptada desde fallos solucionados en [ISO 2003a]
Tolerancia a fallos	Disponibilidad de mecanismos de tolerancia fallos	Capacidad para evitar caídas totales del sistema	[ISO 2003a]
	Eficacia del mecanismo de tolerancia a fallos	Un producto cuyos fallos leves no afecten a la disponibilidad del resto de funciones	[ISO 2005b]
		Un producto cuyos fallos graves no afecten a la disponibilidad de las funciones críticas	[ISO 2005b]
Capacidad de recuperación	Disponibilidad o capacidad para estar disponible durante un período especificado de tiempo	Reparación automática (excluido el mantenimiento humano)	[ISO 2003a]
		Capacidad para evitar fallos (producto software robusto), es decir, reacciona adecuadamente ante situaciones anormales	[ISO 2003a]
		Un tiempo adecuado de no disponibilidad del producto tras un fallo	[ISO 2003a]
	Mecanismos de recuperación	Capacidad del producto de restaurarse (volver a un estado previo) automáticamente después de un evento anormal	[ISO 2003a]
		Capacidad del producto de recuperarse después de un fallo	[ISO 2003a]
		Proporciona documentación sobre las acciones a realizar en caso de desastre para la recuperación del sistema	[ISO 2005g]

Tabla 10. Sub-características y criterios para la característica fiabilidad.

En el caso de usabilidad (Tabla 11) se han definido a nivel de sub-característica. No se han definido criterios por considerar que las aplicaciones de seguridad no tienen diferentes

requisitos que el resto para la evaluación de la comprensión del interfaz o el diálogo, el aprendizaje de la aplicación, su facilidad de operación o en relación a qué atributos utilizar para medir su apariencia. Sin embargo, sí puede ser diferente el grado de importancia otorgado a cada una de las sub-características definidas para usabilidad, por lo tanto, sí tendremos en cuenta la validación de las sub-características.

Sub-característica	Descripción	Fuente
Comprensión	Una ayuda, retroalimentación e información proporcionada en los diálogos de la aplicación fáciles de entender	[ISO 2001a]
Aprendizaje	Un producto con un rápido aprendizaje inicial de los expertos para el uso de la aplicación	[ISO 2001a]
Operación	Un producto fácil de operar y controlar por el administrador	[ISO 2001a]
Apariencia	Disponer de un interfaz atractivo de la aplicación	[ISO 2001a]

Tabla 11. Sub-características para la característica Usabilidad.

Sub-característica	Criterio	Descripción	Fuente
Comportamiento termporal	Tiempo de respuesta	Tiempo de respuesta adecuado a las expectativas del usuario	[ISO 2001a]
	Capacidad de procesamiento	Capacidad de procesamiento en un período de tiempo dado (número de tareas ejecutadas por unidad de tiempo o *throughput*)	[ISO 2001a]
Utilización de recursos	Utilización de recursos hardware	El consumo de recursos hardware (RAM, procesador, disco) del producto es adecuado a las expectativas de los usuarios.	Adaptado desde [ISO 2001a], [Comella-Dorda, Dean 2004], [Alvaro, Santana de Almeida 2006]
Escalabilidad	Capacidad del producto para ser escalable	Capacidad del producto para conserva su efectividad cuando ocurre un incremento significativo en el número de recursos y de usuarios	[Alvaro, Santana de Almeida 2006], [Grance, Stevens 2003], [Torchiano y Jaccheri 2003], [Comella-Dorda, Dean 2004], [Pérez y Tornés 2005], [Villalba y Fernández-Sanz 2007a], [Villalba y Fernández-Sanz 2007b], [Villalba y Fernández-Sanz 2008]

Tabla 12. Sub-características y criterios para la característica eficiencia

Sub-característica	Criterio	Descripción	Fuente
Capacidad para ser actualizado[25]	Disponibilidad de las actualizaciones y parches del producto	Las actualizaciones y parches del producto están disponibles para su instalación por parte de los clientes en un tiempo aceptable	Adaptado desde [ISO 2003a, Avizienis, Laprie 2004] [Grance, Stevens 2003, Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008]
	Facilidad de instalación	Facilidad de instalación de parches, actualizaciones o mejoras proporcionadas por el proveedor de software.	Adaptado desde [ISO 2003a], [Grance, Stevens 2003]
	Esfuerzo requerido para actualizar el producto	Cantidad de esfuerzo requerido para una modificación o borrado de un defecto a través de la instalación de un parche proporcionado por el fabricante	Adaptado desde [Fenton y Pfleeger 1997, Comella-Dorda, Dean 2004]
	Soporte a los cambios de versión	Capacidad de mantener las funcionalidades del producto anteriores a la instalación del parche, así como, las configuraciones realizadas sobre el mismo	Adaptado desde [ISO 2003a] [Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008]
Estabilidad	Estabilidad de los parches o actualizaciones	Los parches o actualizaciones no provocan errores (errores producidos por la instalación de una actualización)	Adaptado desde [Fenton y Pfleeger 1997] e [ISO 2003a]
	Existencia de mecanismos de recuperación de fallos	Capacidad de volver a un estado previo estable tras la aplicación de una actualización o cambio de versión	Adaptado desde [ISO 2003a] [Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008]

Tabla 13. Sub-características y criterios para la característica mantenimiento

[25] La sub-característica ISO 9126 "Capacidad para ser modificado" se ha adaptado a los productos COTS de seguridad como "Capacidad para ser actualizado" ya que las modificaciones en el producto las realiza el fabricante y las proporciona en forma de actualizaciones, parches o *upgrades.*

Sub-característica	Criterio	Descripción	Fuente
Capacidad de instalación	Facilidad en la instalación	El producto proporciona facilidades para su instalación (por ejemplo, asistente guiado)	[ISO 2003a]
	Esfuerzo de instalación	Nivel de esfuerzo requerido por el usuario para instalar el producto.	[ISO 2003a]
	Documentación	Proporciona ayudas o documentación para el proceso de instalación del producto	[ISO 2003a]
	Soporte a la lengua nativa	Permite realizar la instalación del producto en la lengua nativa del operador	[Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008]
	Retroalimentación del proceso de instalación	Proporciona retroalimentación al usuario sobre las acciones que realiza el proceso de instalación y sobre el progreso de la misma.	Adaptado desde [ISO 2006]
	Retroalimentación en los procesos de espera	Proporciona retroalimentación al usuario sobre los procesos en los que hay que esperar.	Adaptado desde [ISO 2006]
Facilidad de configuración[26]	Facilidad de configuración	Proporciona facilidades para su adaptación al entorno operacional (por ejemplo, asistente guiado)	[ISO 2003a]
	Documentación	Proporciona ayudas o documentación para el proceso de configuración del producto	[Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008]
	Personalización de la información visualizada en el registro según conocimiento del	Personalización de la información visualizada en el registro según conocimiento del usuario	[Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008]

[26] Adaptado desde ISO 9126 “Facilidad de adaptación” para productos COTS de protección o seguridad informática.

	usuario		
	Disponibilidad de información sobre la criticidad de los eventos y sucesos notificado	Aporta información sobre la criticidad de los eventos y sucesos notificado (por ejemplo, divide las notificaciones en críticas, advertencias e informativas)	[Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008]
	Capacidad de adaptación del registro de eventos	Permite configuraciones del registro para adaptarlo al entorno del usuario (por ejemplo, ajustar el tamaño de los ficheros o la ubicación de los mismos.)	Adaptado desde [ISO 2003a]

Tabla 14. Sub-características y criterios para la característica Portabilidad

Al igual que con las características, sub-características y criterios técnicos se ha intentado en todo momento mantener los nombres y acepciones originales antes de la validación de las mismas, para el caso de los factores no técnicos se ha procedido de la misma forma. En este caso, se ha tomado la clasificación más general con el fin de que sea el propio ajuste del modelo a través de la aplicación del análisis factorial exploratorio el que se encargue de ajustar el número final de sub-características a tener en cuenta. Dicha clasificación consiste en sólo dos sub-características: producto y proveedor. La sub-característica *producto* se define como cualquier aspecto no técnico relacionado con el producto. Por su parte, la sub-característica *proveedor* se define como cualquier aspecto relacionado con el proveedor que pueda afectar a la selección o evaluación del producto.

Sub-característica	Criterio	Descripción	Fuente
Proveedor	Posicionamiento o cuota de mercado del fabricante	El fabricante del producto tiene una alta participación en el mercado (market share)	[Oberndorf, Brownsword 1997, Torchiano y Jaccheri 2003, Botella, Burgués*, et al.* 2004, Comella-Dorda, Dean 2004, Jaccheri y Torchiano 2004, Carvallo y Franch 2006]
	Reputación del fabricante	El fabricante del producto tiene una buena reputación (por ejemplo, publica un historial de los defectos y	[Oberndorf, Brownsword 1997, Kunda y Brooks 1999, Kunda y Brooks 2000, Alves y Finkelstein 2002, Grance, Stevens 2003, Botella,

		soluciones)	Burgués 2004, Comella-Dorda, Dean 2004, Carvallo y Franch 2006]
	Solvencia del fabricante	El fabricante del producto tiene una adecuada solvencia	[Kontio 1996, Oberndorf, Brownsword 1997, Ochs, Pfahl 2001, Grance, Stevens 2003]
	Experiencia	El fabricante del producto tiene una amplia experiencia en el mercado	[Oberndorf, Brownsword 1997, Kunda y Brooks 2000, Grance, Stevens 2003]
	Fácil acceso	El fabricante del producto tiene una alta accesibilidad con sus clientes	[Comella-Dorda, Dean 2004]
	Autonomía	El fabricante del producto tiene una alta autonomía o independencia con respecto a otros fabricantes	[Comella-Dorda, Dean 2004]
	Calidad del servicio	El fabricante del producto posee algún estándar o certificación sobre calidad del servicio proporcionado	[Oberndorf, Brownsword 1997, Kunda y Brooks 1999, Comella-Dorda, Dean 2004]
Producto	Certificación de seguridad	El producto posee una certificación de seguridad reconocida	[Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008]
	Certificación de calidad del software	El producto conforme con los estándares de calidad del software	[Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008]
	Tecnología	Producto desarrollado con las últimas tecnologías o tendencias	[Oberndorf, Brownsword 1997, Kunda y Brooks 1999, Kunda y Brooks 2000, Comella-Dorda, Dean 2004]
	Compatibilidad con la Arquitectura	Capacidad del producto de adaptarse a la arquitectura en la que se implantará	[Kunda y Brooks 1999, Ochs, Pfahl 2001, Comella-Dorda, Dean 2004]
	Compatibilidad corporativa	Capacidad del producto de adaptarse a la política corporativa de uso de las TIC de la organización	[Kunda y Brooks 1999, Ochs, Pfahl 2001, Comella-Dorda, Dean 2004]
	Estabilidad en el mercado	Estabilidad del producto en el mercado (tiempo que lleva en el mercado, versiones)	[Oberndorf, Brownsword 1997, Kunda y Brooks 1999, Alves y Finkelstein 2002, Botella, Burgués 2004,

			Alvaro, Santana de Almeida 2006, Carvallo y Franch 2006]
	Liderazgo	Producto líder en el mercado	[Kunda y Brooks 2000]
	Licencia	El tipo de licencia se ajusta a las necesidades de la organización	[Oberndorf, Brownsword 1997, Kunda y Brooks 1999, Kunda y Brooks 2000, Botella, Burgués 2004, Comella-Dorda, Dean 2004, Carvallo y Franch 2006]
	Coste	El coste total del producto (licencia, adaptación, integración, formación, soporte, etc.) se ajusta a los requisitos de la organización	[Kontio 1996, Oberndorf, Brownsword 1997, Kunda y Brooks 1999, Kunda y Brooks 2000, Ochs, Pfahl 2001, Alvaro, Santana de Almeida 2006, Carvallo y Franch 2006]
	Soporte	El soporte del producto proporcionado por el proveedor se ajusta a los requisitos de la organización	[Kunda y Brooks 1999, Kunda y Brooks 2000, Ochs, Pfahl 2001, Alves y Finkelstein 2002, Grance, Stevens 2003, Botella, Burgués 2004, Carvallo y Franch 2006]
	Formación	La oferta de formación proporcionada para el producto se ajusta a las necesidades de la organización	[Oberndorf, Brownsword 1997, Kunda y Brooks 1999, Kunda y Brooks 2000, Grance, Stevens 2003, Comella-Dorda, Dean 2004]
	Forma de pago	La forma de pago se ajusta a las necesidades de la organización	[Comella-Dorda, Dean 2004]
	Referencias del cliente	Un producto que ha sido recomendado	[Comella-Dorda, Dean 2004]

Tabla 15. Sub-características y criterios para el modelo de factores no técnicos (NTF)

En el caso de los criterios de usabilidad, se han tomado como sub-características las definidas en el ISO/IEC 9126-1. Tal como puede verse en la Tabla 11, a los expertos en seguridad se les consulta sobre las sub-características más importantes para la usabilidad de los productos software de seguridad de TI. Con el fin de obtener los criterios más importantes para cada una de estas sub-características, se consulta a Ingenieros del Software. Los criterios obtenidos tras la revisión interna suman un total de 53 factores y se

muestran en la Tabla 16 y se han adaptado al dominio de productos de seguridad de TI a partir de los estándares y otra literatura relacionada.

Sub-característica	Criterio	Descripción	Fuente
Comprensión	Compresión de la ayuda	La ayuda proporcionada por el programa se entiende adecuadamente	[Koyanl, Balley 2006]
	Comprensión de la retroalimentación	La información de retroalimentación proporcionada por el programa se entiende adecuadamente	[Koyanl, Balley 2006]
	Comprensión de los diálogos del programa.	La información de los diálogos se entiende de forma adecuada.	[Koyanl, Balley 2006]
	Localización de información.	Es posible localizar la información de manera eficiente.	[Koyanl, Balley 2006]
	Uso de elementos de distracción	Los elementos de distracción (imágenes, multimedia) se utilizan de forma moderada para no interferir en la comprensión del programa.	[Koyanl, Balley 2006]
	Agrupación de la información	La información se agrupa de forma adecuada para mejorar la comprensión.	[Koyanl, Balley 2006]
	Formulación de mensajes	Los mensajes son formulados de forma constructiva, objetiva y comprensible.	[ISO 2006]
	Comprensión global del diálogo mediante la retroalimentación.	La aplicación devuelve información sobre las acciones que realiza permitiendo al usuario una comprensión global del diálogo.	[ISO 2006]
	Información adaptada al nivel de conocimiento de los expertos en seguridad.	La información de retroalimentación, así como, el resto de información dada por el programa se encuentra adaptada al nivel de conocimiento esperado de la población usuaria	[ISO 2006]
	Ayuda sobre el tipo de los datos a introducir.	El sistema indica al usuario la naturaleza de los datos a introducir (formato de los datos)	[ISO 2006]
	Uso de voz activa.	Se mejora la comprensión a través del uso de un lenguaje simple y directo.	[Gediga, Hamborg 1999, Koyanl, Balley 2006]
	Soporte a la lengua nativa en los diálogos	Los diálogos de la aplicación están disponibles en la lengua nativa del usuario.	[Gediga, Hamborg 1999, ISO 2006]
	Soporte a la lengua nativa en la ayuda.	La ayuda de la aplicación está disponible en la lengua nativa del usuario.	[Gediga, Hamborg 1999, ISO 2006]
Aprendizaje	Interfaz único de acceso	Para facilitar el aprendizaje de la aplicación se dispone de un interfaz único de acceso a todas las funcionalidades de la misma.	[Koyanl, Balley 2006]
	Distinción clara de los accesos a las distintas funcionalidades.	Se distingue claramente el acceso a las distintas funcionalidades de la aplicación.	[Koyanl, Balley 2006]

	Claridad del acceso a los mecanismos de apoyo al uso de la aplicación	Se distingue de forma clara el acceso a las utilidades de ayuda proporcionadas por el programa (ayuda, tutoriales, etc.).	[Koyanl, Balley 2006]
	Aspectos conceptuales del programa.	Se proporciona en la ayuda información básica sobre los aspectos conceptuales del programa.	[ISO 2006]
	Ayuda global	Se proporciona acceso a ayuda global de la aplicación	[Koyanl, Balley 2006]
	Ayuda asociada al diálogo	Se proporciona ayuda de cada función particular asociada al diálogo de ejecución correspondiente.	[ISO 2006]
	Ejemplos de aplicación.	Se proporcionan ejemplos de aplicación para facilitar el aprendizaje en la ayuda.	[ISO 2006]
	Tutoriales.	Se proporcionan tutoriales para facilitar el aprendizaje inicial de la aplicación.	[ISO 2006]
	Funcionalidades más usadas	Se tienen en cuenta los comandos más usados para mejorar la experiencia del usuario.	[ISO 2006]
	Soporte a usuarios avanzados	Se proporcionan atajos y soluciones por defecto para los comandos más usados	[Gediga, Hamborg 1999, ISO 2006]
	Soporte a funcionalidades poco usadas	Las funcionalidades poco usadas se acompañan de información más completa de uso.	[ISO 2006]
	Entorno familiar. Mensajes.	Ubicación similar para el mismo tipo de mensajes.	[ISO 2006]
	Entorno familiar. Tareas.	Disposición de pantalla similar para tareas similares.	[Kirakowski y Corbett 1993, ISO 2006]
Operabilidad	Velocidad de interacción.	La velocidad de interacción con el programa no viene impuesta por la aplicación.	[ISO 2006, Koyanl, Balley 2006]
	Control de la aplicación. Modificación.	Las opciones auto-configurables pueden modificarse.	[ISO 2006, Koyanl, Balley 2006]
	Control de la aplicación. Finalización de procesos.	Es posible finalizar cualquier proceso del producto en ejecución.	[Gediga, Hamborg 1999]
	Soporte a la accesibilidad.	El programa permite al usuario elegir entre formas alternativas de representación cuando sea apropiado para las necesidades individuales de los diferentes usuarios.	[Gediga, Hamborg 1999, ISO 2006, Koyanl, Balley 2006]
	Control de la información proporcionada	El programa permite configurar la información proporcionada por la aplicación (por ejemplo, permite ocultar ciertos mensajes de notificación).	[ISO 2006]
	Interrupción de los diálogos	Posibilidad de interrumpir al menos la última parte del diálogo(funciones, acciones) en cualquier momento	[Gediga, Hamborg 1999, ISO 2006]
	Métodos de interacción según experiencia del usuario	Permite la selección del modo de hacer las tareas según perfil y preferencias de usuario.	[ISO 2006]

	Valor recomendado	Se muestra el valor recomendado o diferentes perfiles a elegir cuando existen diferentes opciones	[ISO 2006]
	Control sobre los datos presentados.	Si es útil ejercer control de los datos presentados, el usuario podrá ejercer tal control	[ISO 2006]
	El usuario ejerce el control sobre la entrada de datos	Ejecución de las acciones con teclas de método abreviado (selección entre ratón y teclado)	[ISO 2006]
	El usuario ejerce el control sobre la salida de datos	Almacenamiento de los datos de salida en formato estándar para su uso posterior.	[ISO 2006]
	Diálogos coherentes	El diálogo muestra un aspecto coherente.	[ISO 2006, Koyanl, Balley 2006]
	Retroalimentación en las esperas.	Aviso al usuario si el tiempo de espera va a ser superior al esperado	[ISO 2006, Koyanl, Balley 2006]
	Soporte a la prevención de errores de entrada.	En las entradas de datos se explica cómo deben ser los datos a introducir	[Kirakowski y Corbett 1993, Gediga, Hamborg 1999, ISO 2006]
	Retroalimentación en entradas de datos erróneas.	Se proporciona información sobre cómo deben ser los datos de entrada cuando éstos se introducen de forma errónea	[Kirakowski y Corbett 1993, ISO 2006]
	Validación de los datos de entrada	Se proporciona validación de los datos de entrada introducidos	[Gediga, Hamborg 1999, ISO 2006, Koyanl, Balley 2006]
	Datos de entrada erróneos.	Los datos de entrada erróneos no provocan fallos en la aplicación.	[ISO 2006]
	Facilidades para la corrección de errores. Valores predeterminados.	Posibilidad de volver a los valores predeterminados en cualquier momento.	[Gediga, Hamborg 1999, Villalba y Fernández-Sanz 2007b]
	Facilidades para la corrección de errores. Pérdida de datos.	No se pierde la información que se acaba de introducir tras un error.	[Gediga, Hamborg 1999, ISO 2006]
	Facilidades para la corrección de errores. Recomendaciones.	En los mensajes de error se proporciona información sobre la acción a tomar (por ejemplo, se coloca el cursor donde se encuentra el error de entrada).	[ISO 2006]
	Notificación al usuario de los errores corregidos.	Notificación al usuario de los errores corregidos de forma automática con posibilidad de cancelarlo.	[ISO 2006]
	Obtención de información adicional sobre un error.	Posibilidad de obtener información adicional sobre un error (página web del proveedor, por ejemplo).	[ISO 2006]
	Acciones destructivas	Cuando se lleva a cabo una acción destructiva como borrado de datos, avisa y pide confirmación	[Gediga, Hamborg 1999, ISO 2006]
	Corrección de errores sin cambiar a otro diálogo.	Posibilidad de corregir un error sin cambiar de un diálogo a otro siempre que sea posible.	[ISO 2006]
	Documentación	La documentación se proporciona en formatos y forma operativa.	[ISO 2005g]
Apariencia	Personalización de la apariencia	Personalización de la apariencia del programa para ajustarla al gusto del	[ISO 2006]

		usuario.	
	Combinación de colores y fondo	La combinación de colores y fondo es estéticamente agradable.	[Kirakowski y Corbett 1993]

Tabla 16. Criterios para el modelo de factores de usabilidad (UF).

Por último, las métricas dependen del producto que se vaya a evaluar, así como, de los objetivos y circunstancias del proyecto de evaluación. Por ejemplo, métricas posibles para el criterio "métodos de interacción según experiencia del usuario" cuando el producto a evaluar es un firewall personal podrían ser "permite elegir entre la configuración de reglas en modo experto o mediante asistente", "protección pre-configurada para distintos niveles de seguridad (por ejemplo: bajo, medio, alto)"o "se proporcionan recomendaciones sobre las acciones a realizar tras una alerta". Por ello, las métricas deberán definirse para cada caso concreto de evaluación.

4.1.3. Fase 3. Validación del modelo

4.1.3.1 Diseño de los cuestionarios y recogida de datos

El número de criterios obtenido en la fase anterior era todavía demasiado grande para ser utilizado en la práctica de forma eficiente. Además, dicho conjunto de criterios, aunque se había obtenido de fuentes fidedignas creímos relevante que debía ser validado para el dominio de productos de seguridad informática. Por ello, durante esta fase se llevó a cabo una revisión externa a través de un estudio con un total de 251 expertos (investigadores y profesionales en activo) en las áreas de Ingeniería del software, dirección TIC y seguridad. El proceso de revisión externa se basó en cuestionarios que se entregaron a expertos en distintas áreas.

La hipótesis general que se utilizó fue:

> *Es posible identificar y consensuar las características más relevantes para la evaluación de los productos software finales de seguridad informática, así como su importancia relativa y las relaciones entre los diferentes factores*

Por otra parte, con el fin de tener en cuenta diferentes perspectivas de los expertos relacionados con procesos de evaluación, así como, con el dominio de aplicación, se crearon

3 cuestionarios diferentes según las áreas de conocimiento requeridas[27]: seguridad informática, ingeniería del software y dirección de TI. A partir de los resultados de los cuestionarios dirigidos a expertos en seguridad informática obtuvimos las características técnicas más importantes para los productos COTS de seguridad informática que denominamos TF (del inglés *Technical Factors*). Por su parte, los cuestionarios dirigidos a expertos en ingeniería del software nos proporcionaron las propiedades de facilidad de uso más relevantes para el dominio de seguridad informática que denominamos UF (del inglés *Usability Factors)*. Por último, a partir de los cuestionarios dirigidos a directores y gerentes de TI obtendríamos las características no técnicas (u organizacionales) a evaluar en el dominio objeto del estudio que denominamos NTF (del inglés *Non-Technical Factors*). Esta división además nos proporcionaba dos ventajas adicionales:

1. Por una parte, nos permitía hacer cuestionarios más cortos, reduciendo así el riesgo por abandono del cuestionario;
2. Además, dado que cada uno de los cuestionarios iba enfocado a una población muy concreta la varianza poblacional de la muestra sería potencialmente pequeña y, por tanto, el tamaño muestral requerido para que quedase adecuadamente representada la variedad de componentes en la muestra no debería ser a priori elevado.

Los tres cuestionarios compartían una sección inicial en la que se explicaba el estudio y se requerían datos demográficos de control. El tipo de pregunta utilizada en los cuestionarios fue el de respuesta múltiple con preguntas de estimación (las alternativas se encuentran graduadas en intensidad) y con escala tipo Likert aunque también se utilizó una pregunta final abierta con el fin de recoger posibles características que los expertos pudieran considerar como no incluidas en los cuestionarios. Además de datos de control como edad o país se recogieron datos relacionados con el cargo en la empresa, años de experiencia y conocimiento y experiencia con productos de seguridad. El diagrama de flujo de ejecución

[27] Entre los datos demográficos figuraba una pregunta sobre el área de conocimiento del encuestado a seleccionar entre Ingeniería del software, seguridad informática y Dirección. De esta forma, según la respuesta dada, el encuestado era redirigido al cuestionario correspondiente.

se muestra en la Figura 25 y otros datos técnicos y de diseño en el Anexo 7.1. También se muestra el cuestionario completo en el Anexo 7.3.

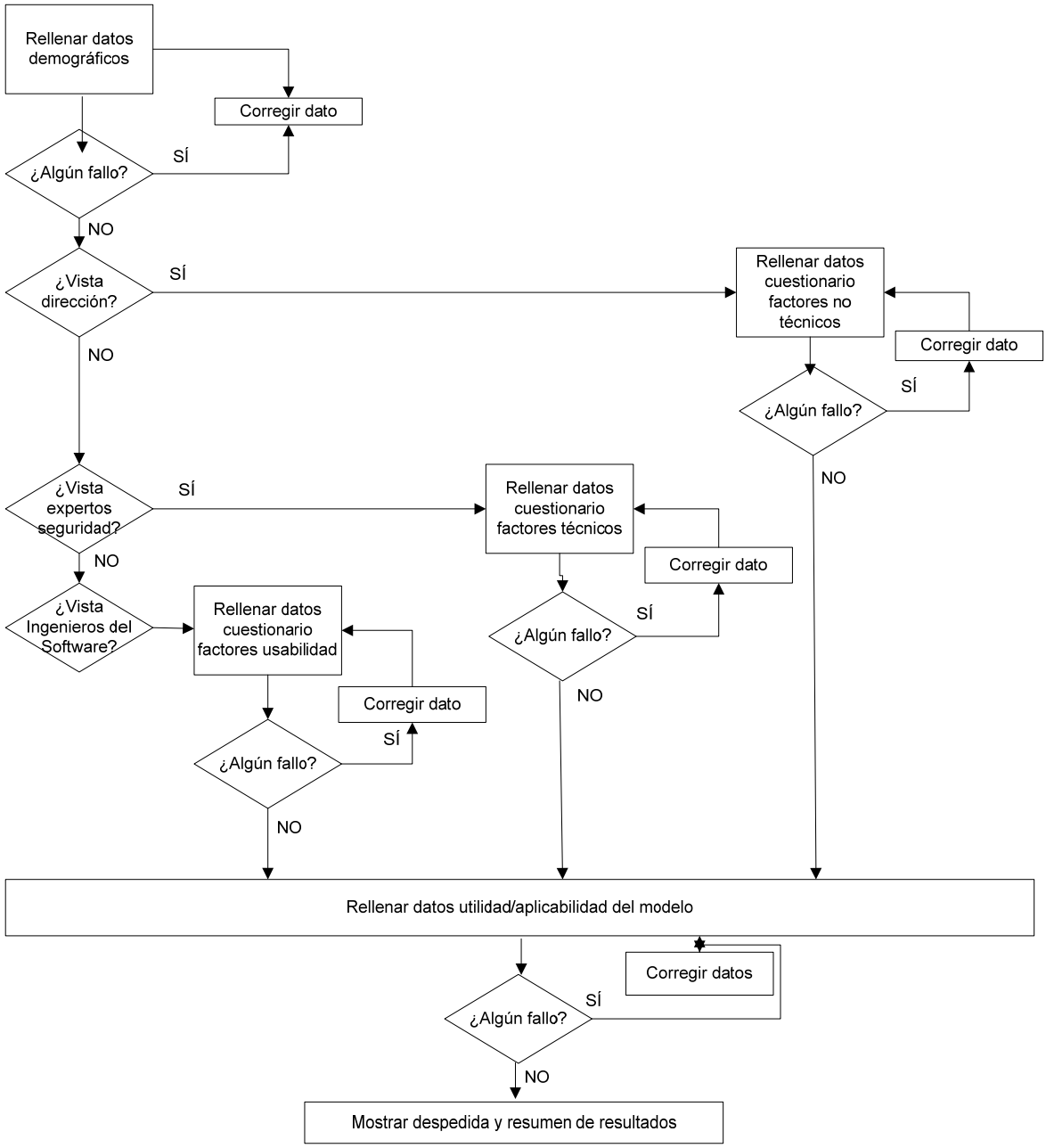


Figura 25. Flujo de ejecución de la aplicación web de recogida de datos.

Con la intención de poder llegar a una mayor población muestral se utilizaron preguntas claras, concisas y cortas. Además, al ir dirigidos los cuestionarios a expertos en el área de conocimiento se pudieron reducir las frases poniéndolas a nivel técnico. Además, las

preguntas se personalizaron dividiéndolas por grupos de conocimiento (seguridad informática, ingeniería del software y dirección) y se puso especial cuidado en la formulación de las mismas evitando, por ejemplo, usar negaciones o no obligando a recurrir a la memoria.

Antes de publicar los cuestionarios, se realizó una validación de los mismos en la que, además de las preguntas del cuestionario, se pedía retroalimentación sobre el proceso. La encuesta piloto se realizó entre:

- alumnos de 5º curso de Ingeniería Informática que habían cursado la asignatura de Seguridad y protección de sistemas informáticos (asignatura optativa de 5º curso) durante dos cursos consecutivos (06/07 y 07/08) para los cuestionarios dirigidos a expertos en el área de la seguridad;
- alumnos de 5º curso de Ingeniería Informática cursando la asignatura de Comunicación hombre-máquina (asignatura optativa) para los cuestionarios dirigidos a expertos en el área de usabilidad;
- alumnos del Máster Oficial de Gestión de Tecnologías de la información (principalmente profesionales en el área de gestión y dirección de TI) para los cuestionarios dirigidos a expertos en el área de dirección de TI.

La retroalimentación recibida en las pruebas iniciales de los cuestionarios en los que los encuestados se enfrentaban a la pregunta:" ¿qué importancia tiene cada criterio para la adquisición de un producto de seguridad informática?" en una escala de 1 (nada importante) a 5 (muy importante), nos indicó que resultaba difícil discriminar dentro de la escala pues en realidad todas las propiedades dadas eran realmente importantes. Por ello, se decidió que para obtener respuestas fiables y válidas cada revisor evaluase cada criterio asignándole un valor basado en la respuesta a la pregunta "¿en qué medida estaría dispuesto a asumir un coste extra en el proceso de compra por cada una de las siguientes características?". En este caso la escala definida fue: "nunca", "casi nunca", "a veces", "casi siempre" y "siempre".

Por último, resaltar el hecho de que se automatizaron las encuestas y se publicaron en internet para un fácil acceso y posterior tratamiento estadístico de los datos. La información de acceso se proporcionaba a los expertos a través de correo electrónico con un enlace al sitio web en el que se encontraban los cuestionarios, de forma que, con un simple click se pudiera acceder a los mismos. Además, para asegurar que las personas que participaban en el estudio tenían el perfil requerido, se invitó tan sólo a expertos de empresas colaboradoras de los másteres de seguridad TI y de Gestión de Tecnologías de la información de la Universidad Europea de Madrid. También se pidió la colaboración de asociaciones, como ATI (Asociación de Técnicos de Informática), AEDI (Asociación Española de Directores de Informática) o ASIMELEC (Asociación Multisectorial de Empresas Españolas de Electrónica) y revistas especializadas como Red Seguridad y e.Security para la difusión del estudio.

4.1.3.2 Análisis y descripción de la muestra

De los 253 cuestionarios recogidos, el número de completados y válidos fue de 203 por lo que el margen de error fue del 6,9% con un intervalo de confianza del 95%. En la Tabla 17 se muestran los resultados completos obtenidos. Como puede observarse en dicha tabla, la mayoría de los participantes en el estudio tenían entre 31 y 40 años (36.46%) o entre 41 y 50 años (34.25%). En relación a la distribución de los encuestados según la actividad económica de la empresa los principales sectores fueron el sector de Tecnologías de la información y Comunicaciones (38.12 %), Industria (23.20%), Administraciones públicas y Educación y Formación (9.94%). El perfil técnico de los participantes estaba bastante igualado entre gestores o directores de Tecnologías de la información (30.54%) e Ingenieros de seguridad (25.82%), seguido por Ingenieros del Software (19.34), consultores de Tecnologías de la información (7.73%) y profesores e investigadores (6.63%). La mayoría de ellos tenía más de 5 años de experiencia en su puesto (64.64%). Con respecto a su cualificación profesional en relación a los productos software de seguridad, tal como puede observarse en la Figura 26, la mayoría de ellos tenía un nivel muy alto de experiencia en el uso de este tipo de productos con un 30,94% para el nivel experto (nivel muy alto), un 16,57 % para el nivel de evaluador/consultor (nivel alto), un 23,93% para el nivel administrador (nivel medio) y sólo un 15,04% con nivel usuario (nivel bajo). Por último, la mayoría de ellos tenían una formación básica (49.72%) o avanzada (32.04%) en productos de seguridad.

EDAD		PERFIL PROFESIONAL	
21 años o menos	1,66%	Desarrollador (programador, analista, etc.)	19,34%
Entre 22 y 30 años	19,89%	dirección de Informática	14,36%
Entre 31 y 40 años	36,46%	Administración de sistemas informáticos	14,36%
Entre 41 y 50 años	34,25%	Jefe de proyecto	9,94%
51 años o más	7,73%	Consultor Informático	7,73%
SECTOR EMPRESARIAL		Personal docente	6,63%
Informática y Telecomunicaciones	38,12%	Operador	3,87%
Administraciones públicas	23,20%	Propietario o gerencia	3,31%
Educación y Formación	9,94%	dirección de seguridad TI	3,31%
Consultoría y/o auditoria de negocio	7,18%	Especialista en seguridad informática	2,76%
Industria	5,52%	dirección de Redes y Comunicaciones	2,76%
Organismos oficiales	2,76%	Otra dirección no informática	2,76%
Entidades financieras	2,76%	Administrador de Bases de Datos	2,21%
Administración de empresas	1,66%	Especialista en Redes informáticas	1,66%
Energía y transportes	1,66%	Auditor informático	1,66%
Construcción y Obras públicas	1,66%	dirección de documentación e información	1,10%
I+D	1,66%	Arquitectura informática	1,10%
Comercio y distribución	1,10%	Ingeniero de Calidad	1,10%
Salud y Farmacia	1,10%	**AÑOS DE EXPERIENCIA EN EL PUESTO**	
Marketing y comunicación	1,10%	Ninguna	1,66%
Recursos Humanos	0,55%	Menos de 1 año	3,31%
Seguros	0,00%	Entre 1 y 3 años	13,81%
Hostelería y Turismo	0,00%	Entre 3 y 5 años	16,57%
Comercial y ventas	0,00%	Más de 5 años	64,64%
FORMACIÓN EN PRODUCTOS SOFTWARE DE SEGURIDAD		**EXPERIENCIA CON PRODUCTOS SOFTWARE DE SEGURIDAD**	
Sólo tengo información	18,23%	Los he visto funcionar alguna vez	1,88%
Tengo formación específica básica	49,72%	Nivel de usuario	15,04%
Tengo formación específica avanzada	32,04%	Nivel de administrador	23,93%
		Evaluador	16,57%
		Asesores, expertos	30,94%

Tabla 17. Resultados datos demográficos

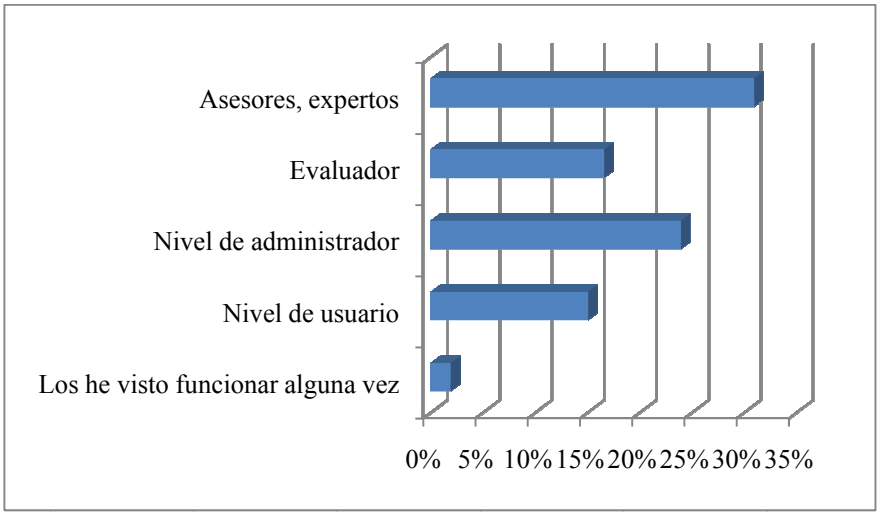


Figura 26. Experiencia en el uso de productos software de seguridad informática.

4.1.3.3 Análisis de datos

Análisis previo: examen preliminar de las variables de datos

En primer lugar se resume el análisis descriptivo efectuado para cada una de las escalas de medida del modelo teórico obtenido en el capítulo 3. Para cada variable se muestra de forma comparativa la valoración realizada a través de un resumen los principales estadísticos descriptivos (media, mediana, moda y desviación típica), además, con el fin de obtener información sobre la distribución de los datos, se examinan también los histogramas. Previo al análisis exploratorio de los datos se han observado los diagramas de caja con el fin de descubrir datos atípicos, permitiéndonos así, eliminar posibles errores producidos durante la recogida de datos.

Examen preliminar del modelo de factores técnicos

En primer lugar examinamos las variables correspondientes a las características obtenidas tras la adaptación del modelo estándar (ISO/IEC 9126) al dominio específico tal como se especificó en el capítulo anterior.

a) Funcionalidad

		Media	Mediana	Moda	Desviación estándar	Kolmogorov-Smirnov[a] Estad.	gl	Sig.
FUNC1	Exactitud de los resultados obtenidos tras la ejecución de las funciones.	3,792	4	4	0,821	0,254	61	0,000
FUNC2	Interactúa adecuadamente con otros programas.	3,681	4	4	0,869	0,263	61	0,000
FUNC3	Especificaciones hardware.	3,5	3,5	3	1,035	0,189	61	0,000
FUNC4	Especificaciones y requisitos software.	3,722	4	3	0,843	0,252	61	0,000
FUNC5	Compatibilidad con otros programas del mismo tipo o similar.	3,639	4	4	0,793	0,311	61	0,000
FUNC6	Dispone de certificación de seguridad	3,639	4	4	0,827	0,31	61	0,000
	TOTAL MUESTRA	61						

a. Corrección de Lilliefors

Tabla 18. Análisis descriptivo para las variables correspondientes a la característica de Funcionalidad.

En relación a la normalidad sólo la variable FUNC3 sigue una distribución normal (nivel de significación α=0,01).

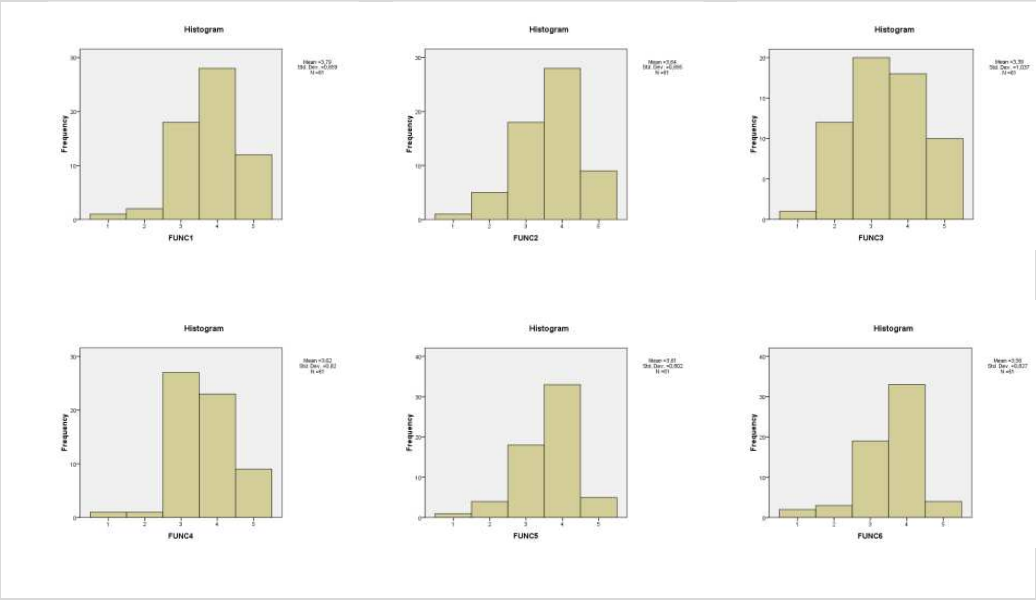


Figura 27. Histogramas para las variables correspondientes a la característica de Funcionalidad.

b) Fiabilidad

		Media	Mediana	Moda	Desviación estándar	Kolmogorov-Smirnoff Estad.	gl	Sig.
FIAB8	Bajo porcentaje de fallos (madurez del producto)	4,236	4	4	0,661	0,373	61	0,000
FIAB9	Tiempo de espera para la corrección de fallos (disponibilidad de parches).	3,931	4	4	0,828	0,272	61	0,000
FIAB10	Los fallos provocados por errores del software no afectan a las funciones críticas.	4,056	4	5	0,963	0,218	61	0,000
FIAB11	El número de fallos críticos del software es mínimo o inexistente.	4,181	4	4	0,845	0,326	61	0,000
FIAB12	Los fallos leves en el software no afectan a la disponibilidad del resto de funciones.	3,903	4	4	0,922	0,249	61	0,000
FIAB13	Los fallos graves del software no afectan a la disponibilidad de las funciones críticas.	4,069	4	5	0,954	0,23	61	0,000
FIAB14	Disponibilidad de funciones para recuperarse de forma automática tras un fallo.	3,903	4	4	0,891	0,219	61	0,000
FIAB15	Tiempo en el que el software no está disponible tras un fallo.	3,903	4	4	0,875	0,335	61	0,000
FIAB16	Capacidad de volver a un estado previo automáticamente después de un evento anormal *(restore).*	3,875	4	4	0,838	0,273	61	0,000
FIAB17	Capacidad de volver a un estado de funcionamiento normal después de un fallo *(recovery)*.	3,944	4	4	0,820	0,297	61	0,000
FIAB18	Robustez del producto software.	4,097	4	4	0,754	0,223	61	0,000
FIAB19	Documentación sobre las acciones a realizar para la recuperación del sistema	3,639	4	3	1,066	0,207	61	0,000
	TOTAL MUESTRA	61						

a. Corrección de Lilliefors

Tabla 19. Análisis descriptivo para las variables correspondientes a la característica de Fiabilidad.

En relación a la normalidad ninguna de las variables sigue una distribución normal.

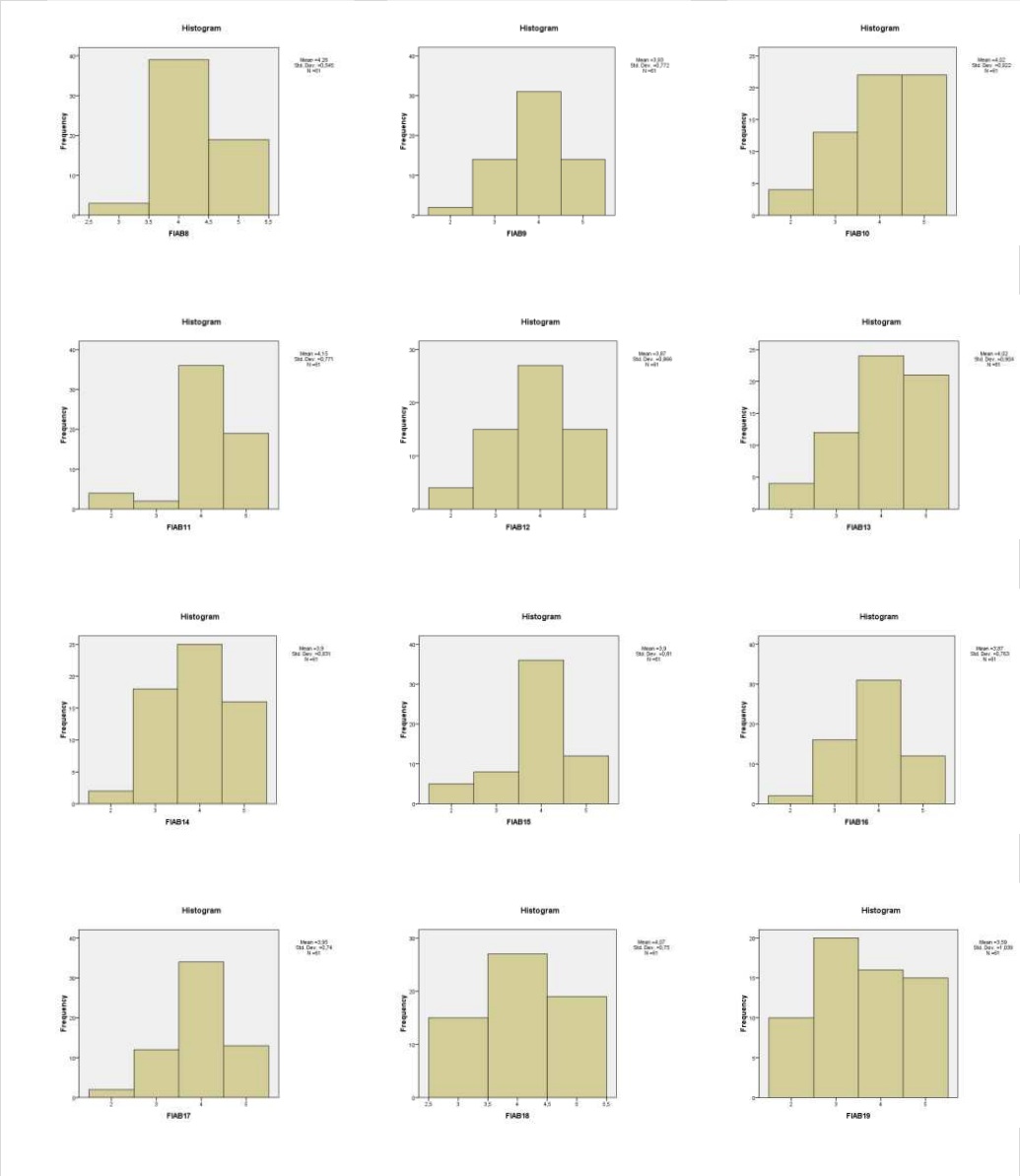


Figura 28. Histogramas para las variables correspondientes a la característica de Fiabilidad.

c) Usabilidad

		Media	Mediana	Moda	Desviación estándar	Kolmogorov-Smirnoff Estad.	gl	Sig.
USAB21	Facilidad de comprensión de la ayuda, información y los diálogos de la aplicación.	3,458	3,5	4	0,768	0,27	61	0,000
USAB22	Facilidad de aprendizaje inicial de los expertos para el uso de la aplicación.	3,514	4	4	0,888	0,234	61	0,000
USAB23	Facilidad de operación y control por el administrador.	3,819	4	4	0,828	0,255	61	0,000
USAB24	Interfaz atractivo.	3,125	3	3	1,087	0,242	61	0,000
	TOTAL MUESTRA	61						

a. Corrección de Lilliefors

Tabla 20. Análisis descriptivo para las variables correspondientes a la característica de Usabilidad.

En relación a la normalidad ninguna de las variables sigue una distribución normal.

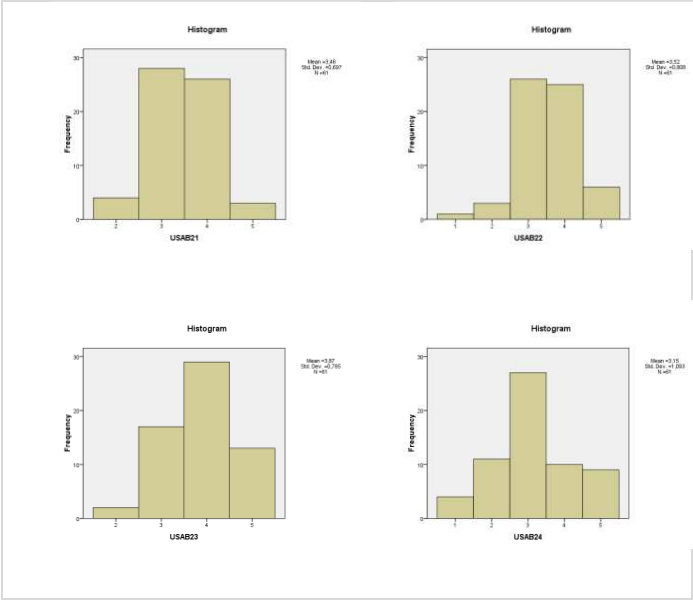


Figura 29. Histogramas para las variables correspondientes a la característica de Usabilidad.

d) Eficiencia

						Kolmogorov-Smirnoff		
		Media	Mediana	Moda	Desviación estándar	Estad.	gl	Sig.
EFIC26	Tiempo de respuesta.	3,944	4	4	0,710	0,301	61	0,000
EFIC27	Número de tareas ejecutadas por unidad de tiempo.	3,778	4	4	0,809	0,245	61	0,000
EFIC28	Consumo de memoria.	3,500	3	3	0,751	0,275	61	0,000
EFIC29	Consumo de procesador.	3,319	3	3	0,901	0,26	61	0,000
EFIC30	Espacio en disco necesario para la ejecución del programa tras la instalación.	3,264	3	3	0,872	0,247	61	0,000
EFIC31	Escalabilidad del producto software.	4,056	4	4	0,837	0,223	61	0,000
	TOTAL MUESTRA	61						

a. Corrección de Lilliefors

Tabla 21. Análisis descriptivo para las variables correspondientes a la característica de Eficiencia.

En relación a la normalidad ninguna de las variables sigue una distribución normal.

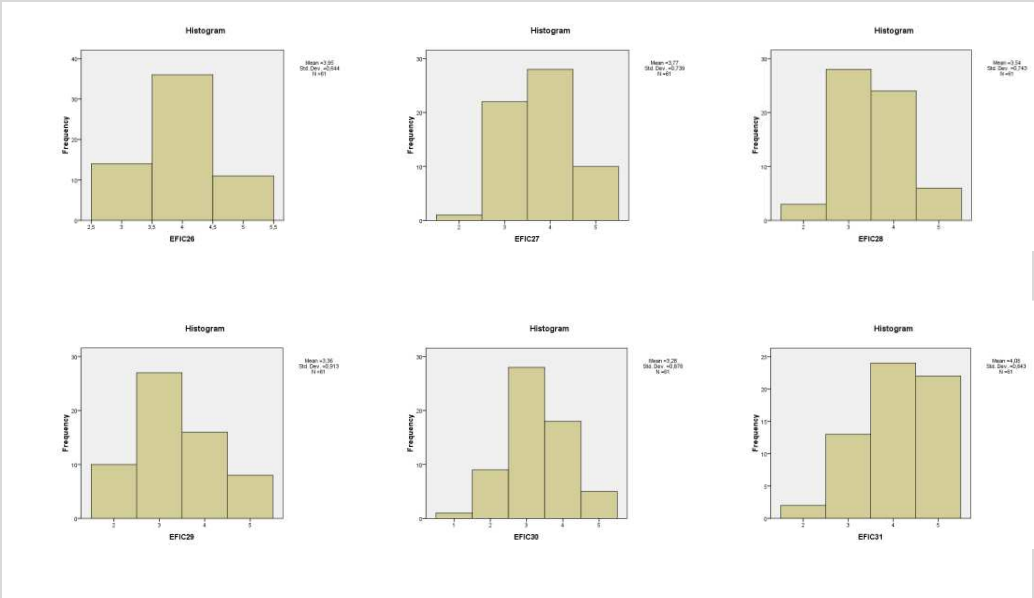


Figura 30. Histogramas para las variables correspondientes a la característica de Eficiencia.

e) Mantenimiento

		Media	Mediana	Moda	Desviación estándar	Kolmogorov-Smirnoff Estad.	gl	Sig.
MANT33	Tiempo de disponibilidad de las actualizaciones del producto.	3,625	4	4	0,985	0,207	61	0,000
MANT34	Facilidad de instalación de actualizaciones.	3,625	4	4	0,846	0,276	61	0,000
MANT35	Esfuerzo requerido por el usuario para actualizar el producto.	3,639	4	4	0,827	0,29	61	0,000
MANT36	Estabilidad de los parches (actualizaciones del sistema para resolver errores).	3,917	4	4	0,900	0,209	61	0,000
MANT37	Capacidad de volver a un estado previo tras la aplicación de un parche o de un cambio de versión.	3,792	4	4	0,871	0,223	61	0,000
MANT38	Capacidad de mantener las configuraciones en los cambios versión del producto.	4,083	4	4	0,835	0,243	61	0,000
	TOTAL MUESTRA	61						

a. Corrección de Lilliefors

Tabla 22. Análisis descriptivo para las variables correspondientes a la característica de Mantenimiento.

En relación a la normalidad ninguna de las variables sigue una distribución normal.

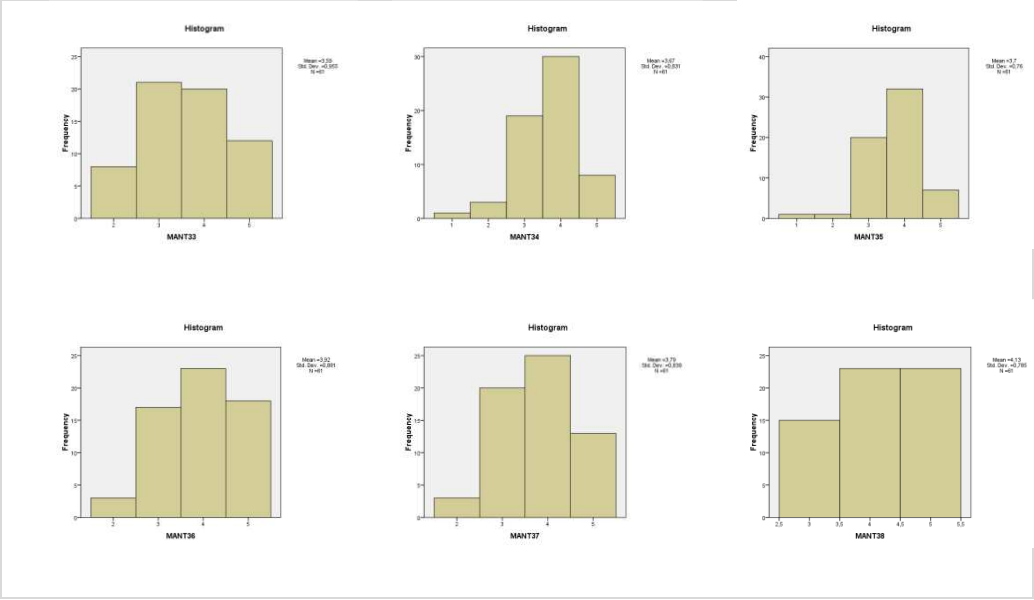


Figura 31. Histogramas para las variables correspondientes a la característica de Mantenimiento.

f) Portabilidad

						Kolmogorov-Smirnoff		
		Media	Mediana	Moda	Desviación estándar	Estad.	gl	Sig.
PORT40	Facilidad de instalación.	3,222	3	3	0,982	0,229	61	0,000
PORT41	Esfuerzo requerido por el usuario para instalar el producto.	3,236	3	3	1,000	0,201	61	0,000
PORT42	Ayuda al proceso de instalación del producto.	3,556	4	4	1,005	0,246	61	0,000
PORT43	Soporte a la lengua nativa en el proceso de instalación.	3,000	3	2	1,256	0,181	61	0,000
PORT44	Retroalimentación al usuario durante el proceso de instalación.	3,250	3	3	0,884	0,243	61	0,000
PORT45	Retroalimentación al usuario sobre los procesos en los que hay que esperar.	3,083	3	3	0,915	0,235	61	0,000
PORT46	Asistente de configuración guiada tras la instalación del producto.	3,125	3	3	1,020	0,189	61	0,000
PORT47	Ayuda al proceso de configuración del producto.	3,417	3	3	0,900	0,226	61	0,000
PORT48	Personalización de la información visualizada en el registro de eventos (logs).	3,333	3	3	0,919	0,25	61	0,000
PORT49	Información sobre la criticidad de los eventos y sucesos notificados.	3,861	4	4	0,793	0,286	61	0,000
PORT50	Posibilidad de realizar personalizaciones del registro de eventos.	3,472	4	4	0,978	0,232	61	0,000
	TOTAL MUESTRA	61						

a. Corrección de Lilliefors

Tabla 23. Análisis descriptivo para las variables correspondientes a la característica de Portabilidad.

En relación a la normalidad sólo las variables PORT41 y PORT46 siguen una distribución normal (nivel de significación α=0,01).

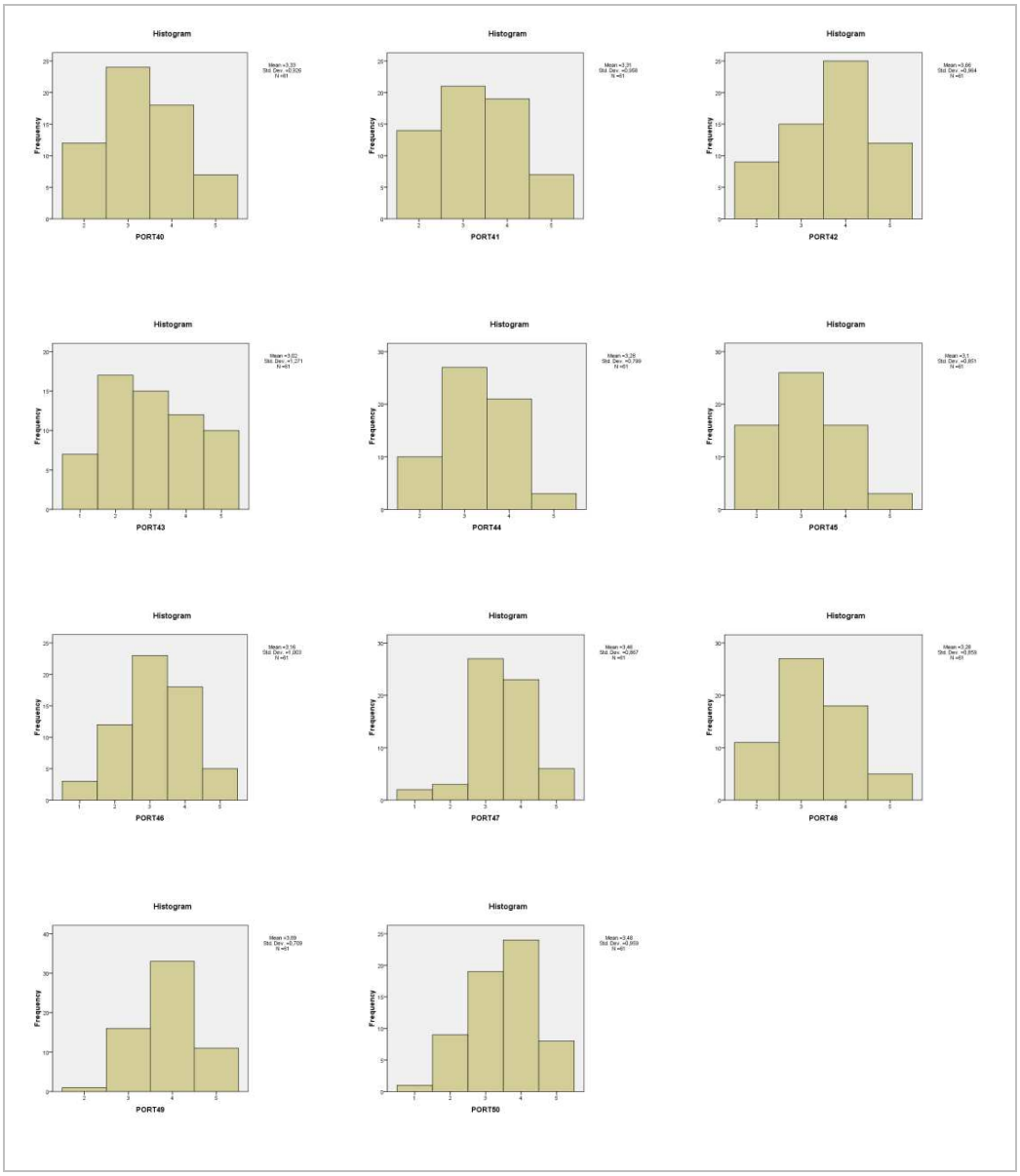


Figura 32. Histogramas para las variables correspondientes a la característica de Portabilidad.

Examen preliminar del modelo de factores no técnicos

En esta sección haremos un análisis análogo al anterior con los datos recogidos para los factores no técnicos.

a) Características no técnicas relacionadas con el proveedor o fabricante del producto software:

Test de Kolmogorov-Smirnoff

		Media	Mediana	Moda	Desv. estándar	Estad	gl	Sig.
PROV12	Participación en el mercado (market share) del proveedor.	2,984	3	3	0,806	0,230	61	0,000
PROV13	Reputación del proveedor de software.	3,197	3	3	0,749	0,243	61	0,000
PROV14	Estabilidad o solvencia del proveedor.	3,377	3	3	0,734	0,254	61	0,000
PROV15	Experiencia del proveedor.	3,656	4	4	0,704	0,392	61	0,000
PROV16	Accesibilidad del proveedor.	3,574	4	4	0,590	0,355	61	0,000
PROV17	Autonomía e independencia del proveedor con respecto a otros fabricantes.	3,197	3	3	0,726	0,243	61	0,000
PROV18	Posee algún estándar o certificación que valoren la calidad del servicio del proveedor	3,459	3	3	0,743	0,258	61	0,000
	TOTAL MUESTRA	61						

a. Corrección de Lilliefors

Tabla 24. Análisis descriptivo para las variables relacionadas con el Proveedor.

En relación a la normalidad ninguna de las variables sigue una distribución normal.

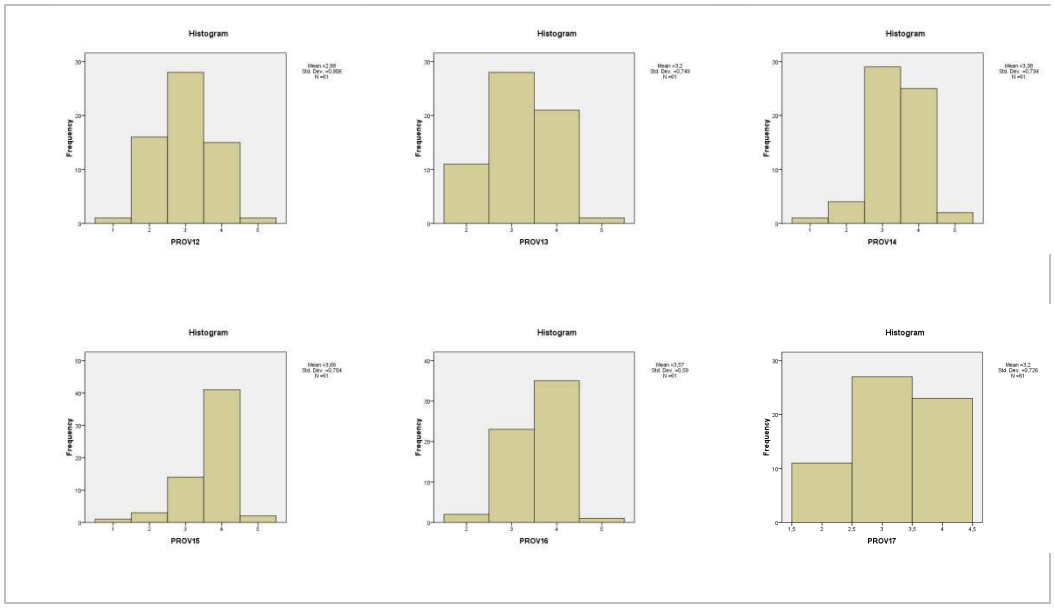


Figura 33. Histogramas para las variables relacionadas con el proveedor.

b) Características no técnicas relacionadas con el producto software:

						Test de Kolmogorov-Smirnoff		
		Media	Mediana	Moda	Desv. estándar	Estad	gl	Sig.
PROD20	Uso de las últimas tecnologías.	3,295	3	3	0,760	0,241	61	0,000
PROD21	Compatible con la arquitectura de la organización en la que se integrará.	4,246	4	4	0,745	0,254	61	0,000
PROD22	Compatible con la política corporativa de uso de las TIC de la organización.	4,098	4	4	0,676	0,279	61	0,000
PROD23	Estabilidad del producto.	3,803	4	4	0,726	0,312	61	0,000
PROD24	Liderazgo del producto en el mercado.	3,230	3	3	0,761	0,307	61	0,000
PROD25	Tipo de licencia (por puesto, por CPU, etc.).	4,066	4	4	0,772	0,237	61	0,000
PROD26	Coste total del producto (licencia, adaptación, integración, formación, soporte, etc)	3,934	4	4	0,750	0,223	61	0,000
PROD27	Tipo de soporte.	3,918	4	4	0,822	0,261	61	0,000
PROD28	Oferta de formación para el uso y administración del producto.	3,525	4	4	0,887	0,261	61	0,000
PROD29	Forma de pago del producto software.	3,607	4	3	0,988	0,222	61	0,000
PROD30	Recomendado por otros.	3,000	3	3	0,816	0,270	61	0,000
TECN9	Posee certificación de seguridad.	3,541	3	3	0,828	0,251	61	0,000
TECN10	Conformidad con los estándares existentes de calidad del software.	3,689	4	4	0,765	0,281	61	0,000
	TOTAL MUESTRA	61						

Tabla 25. Análisis descriptivo para las variables relacionadas con el Producto.

En relación a la normalidad ninguna de las variables sigue una distribución normal.

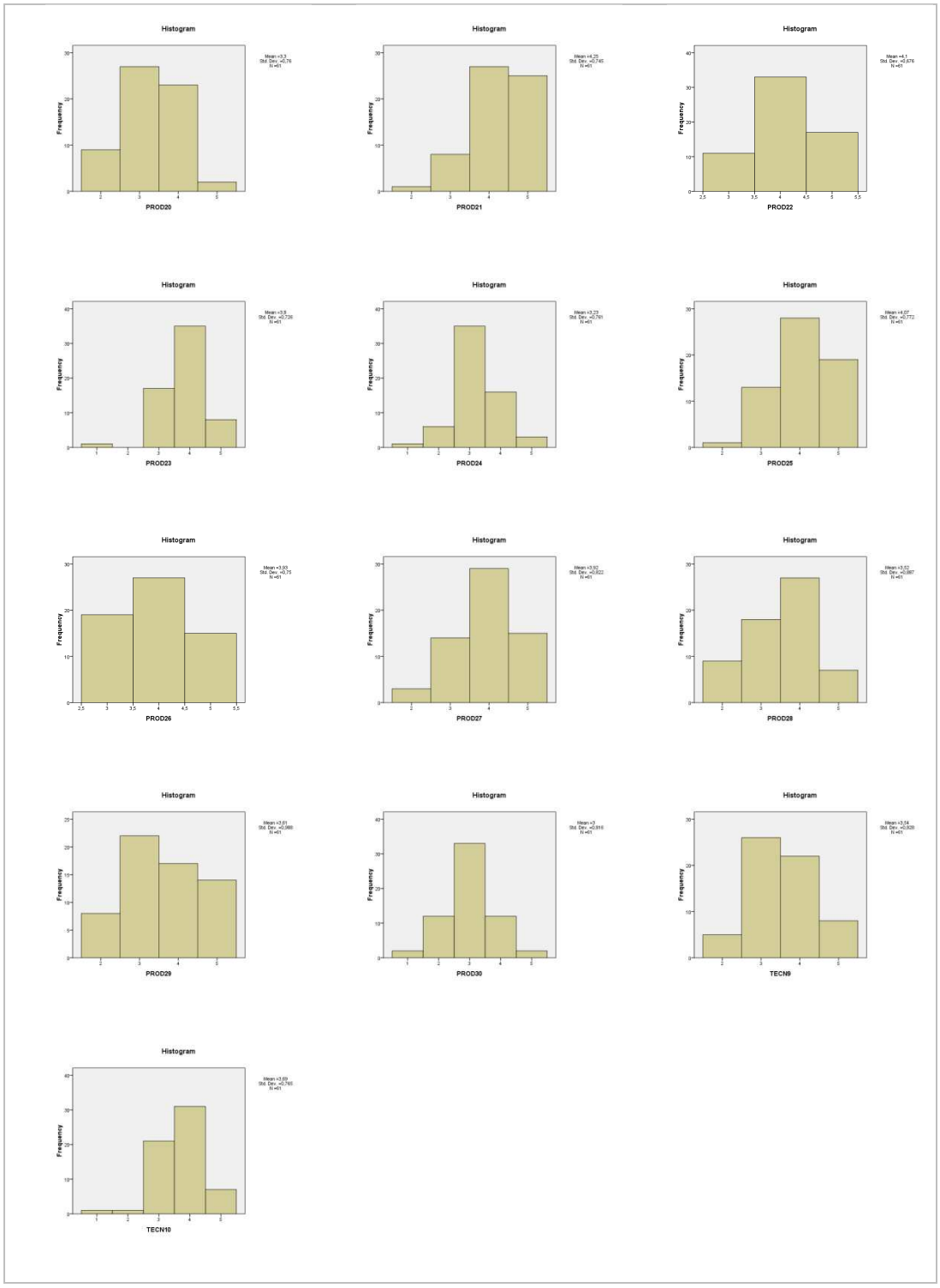


Figura 34. Histogramas para las variables relacionadas con el producto.

Examen preliminar del modelo de criterios de usabilidad

Por último, analizamos los datos de usabilidad recogidos.

a) Compresión:

						Test de Kolmogorov-Smirnoff		
		Media	Mediana	Moda	Desv. estándar	Estad.	gl	Sig.
COMPR1	Compresión de la ayuda proporcionada por el programa.	3,419	3	3	0,922	0,229	74	0,000
COMPR2	Comprensión de la retroalimentación proporcionada por el programa.	3,297	3	3	0,903	0,224	74	0,000
COMPR3	Comprensión de la información proporcionada en los diálogos del programa.	3,608	4	3	0,934	0,229	74	0,000
COMPR4	Localización eficiente de información.	3,919	4	4	0,856	0,294	74	0,000
COMPR5	Uso de elementos de distracción (imágenes, multimedia) moderado.	3,122	3	3	1,204	0,162	74	0,000
COMPR6	Agrupación de la información adecuada para mejorar la comprensión.	3,770	4	4	0,853	0,295	74	0,000
COMPR7	Mensajes de formulados de forma constructiva, objetiva y comprensible.	4,027	4	4	0,793	0,311	74	0,000
COMPR8	Comprensión global del diálogo mediante la retroalimentación.	3,432	3	3	0,877	0,228	74	0,000
COMPR9	Información adaptada al nivel de conocimiento de los expertos en seguridad.	3,419	4	4	1,110	0,254	74	0,000
COMPR10	Ayuda y/o retroalimentación sobre la naturaleza de los datos a introducir.	3,568	4	4	0,893	0,240	74	0,000
COMPR11	Lenguaje simple y directo.	4,149	4	4	0,806	0,233	74	0,000
COMPR12	Soporte a la lengua nativa en los diálogos de la aplicación.	3,770	4	5	1,177	0,217	74	0,000
COMPR13	Soporte a la lengua nativa en la ayuda.	3,797	4	5	1,170	0,217	74	0,000
	TOTAL MUESTRA	74						

Tabla 26. Análisis descriptivo para las variables correspondientes a la sub-característica de Comprensión.

En relación a la normalidad sólo la variable COMPR5 sigue una distribución normal (nivel de significación α=0,01).

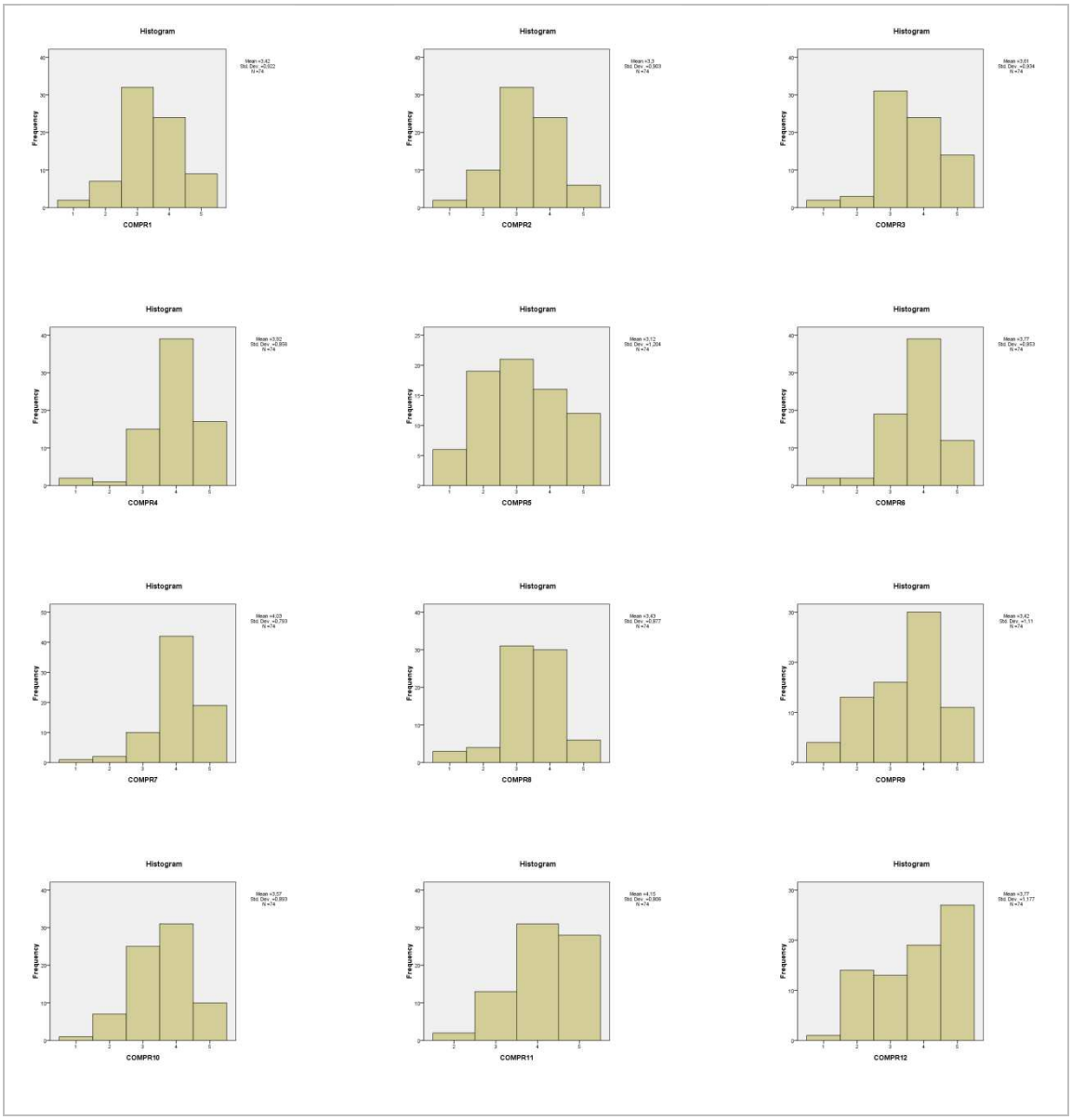


Figura 35. Histogramas para las variables correspondientes a la sub-característica de Comprensión

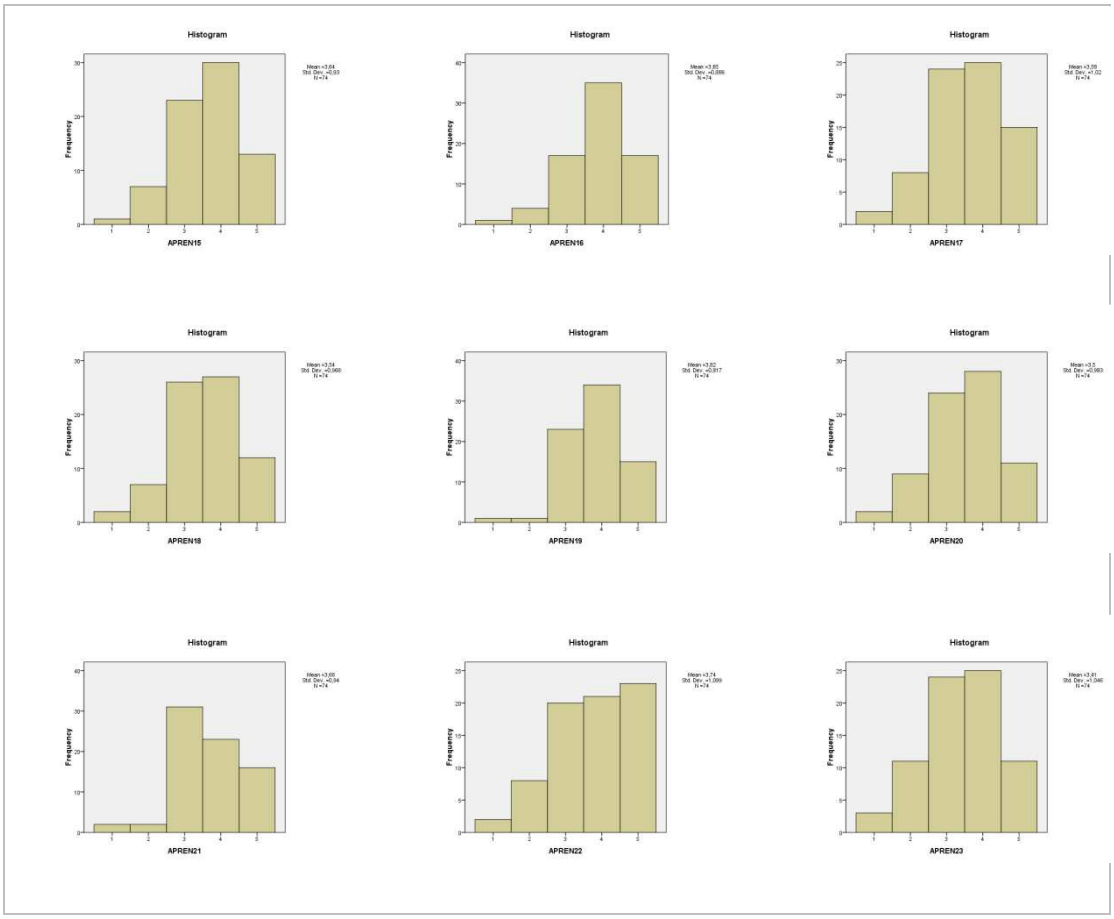
b) Aprendizaje:

						Test de Kolmogorov-Smirnoff		
		Media	Mediana	Moda	Desv. estándar	Estad.	gl	Sig.
APREN14	Interfaz único de acceso a la aplicación para facilitar el Aprendizaje.	3,635	4	4	0,930	0,234	74	0,000
APREN15	Distinción clara de los accesos a las distintas funcionalidades de la aplicación.	3,851	4	4	0,886	0,269	74	0,000
APREN16	Claridad del acceso a los mecanismos de apoyo al uso de la aplicación (ayuda, tutoriales, etc.).	3,581	4	4	1,020	0,200	74	0,000
APREN17	Información básica sobre aspectos conceptuales del programa en la ayuda.	3,541	4	4	0,968	0,210	74	0,000
APREN18	Ayuda global del programa.	3,824	4	4	0,817	0,247	74	0,000
APREN19	Ayuda de cada función particular asociada al diálogo de ejecución correspondiente.	3,500	4	4	0,983	0,222	74	0,000
APREN20	Ejemplos de aplicación para facilitar el APREN en la ayuda.	3,662	4	3	0,940	0,232	74	0,000
APREN21	Tutoriales para facilitar el APREN inicial de la aplicación.	3,743	4	5	1,099	0,187	74	0,000
APREN22	Uso de las funcionalidades más utilizadas para mejorar la experiencia del usuario.	3,405	3	4	1,046	0,202	74	0,000

APREN23	Atajos para usuarios avanzados y soluciones por defecto para las funciones más usadas.	3,432	3	3	0,994	0,236	74	0,000
APREN24	Funcionalidades poco usadas se acompañan de información más completa de uso.	3,014	3	3	0,972	0,235	74	0,000
APREN25	Ubicación similar para el mismo tipo de mensajes.	3,311	3	3	1,006	0,216	74	0,000
APREN26	Disposición de pantalla similar para tareas similares.	3,649	4	4	0,943	0,253	74	0,000
	TOTAL MUESTRA	74						

Tabla 27. Análisis descriptivo para las variables de la sub-característica de Aprendizaje.

En relación a la normalidad sólo la variable APREN21 sigue una distribución normal (nivel de significación α=0,01).



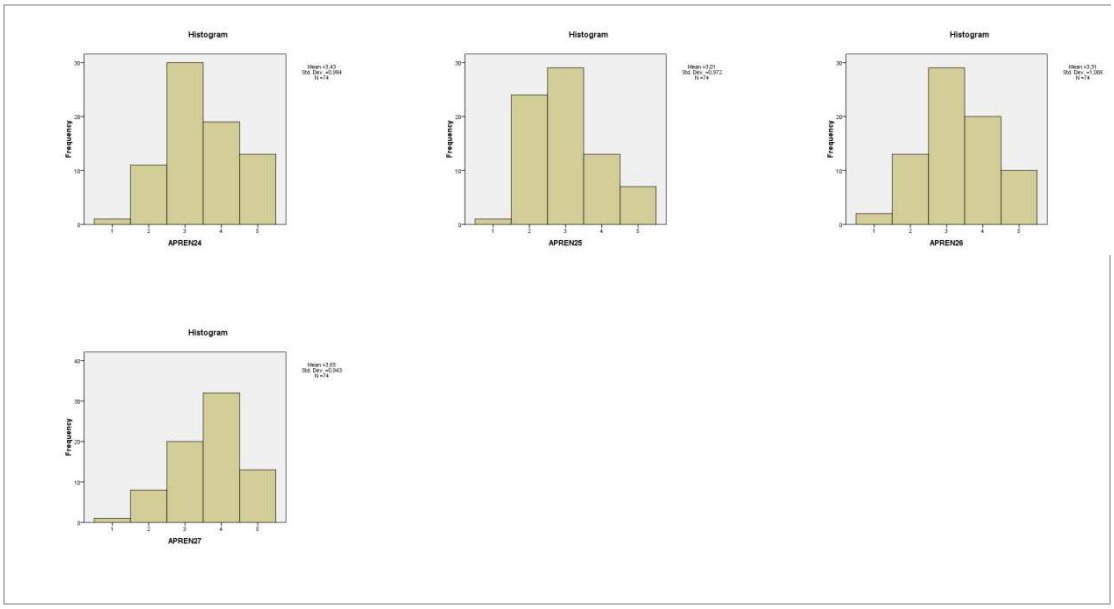


Figura 36. Histogramas para las variables correspondientes a la sub-característica de Aprendizaje.

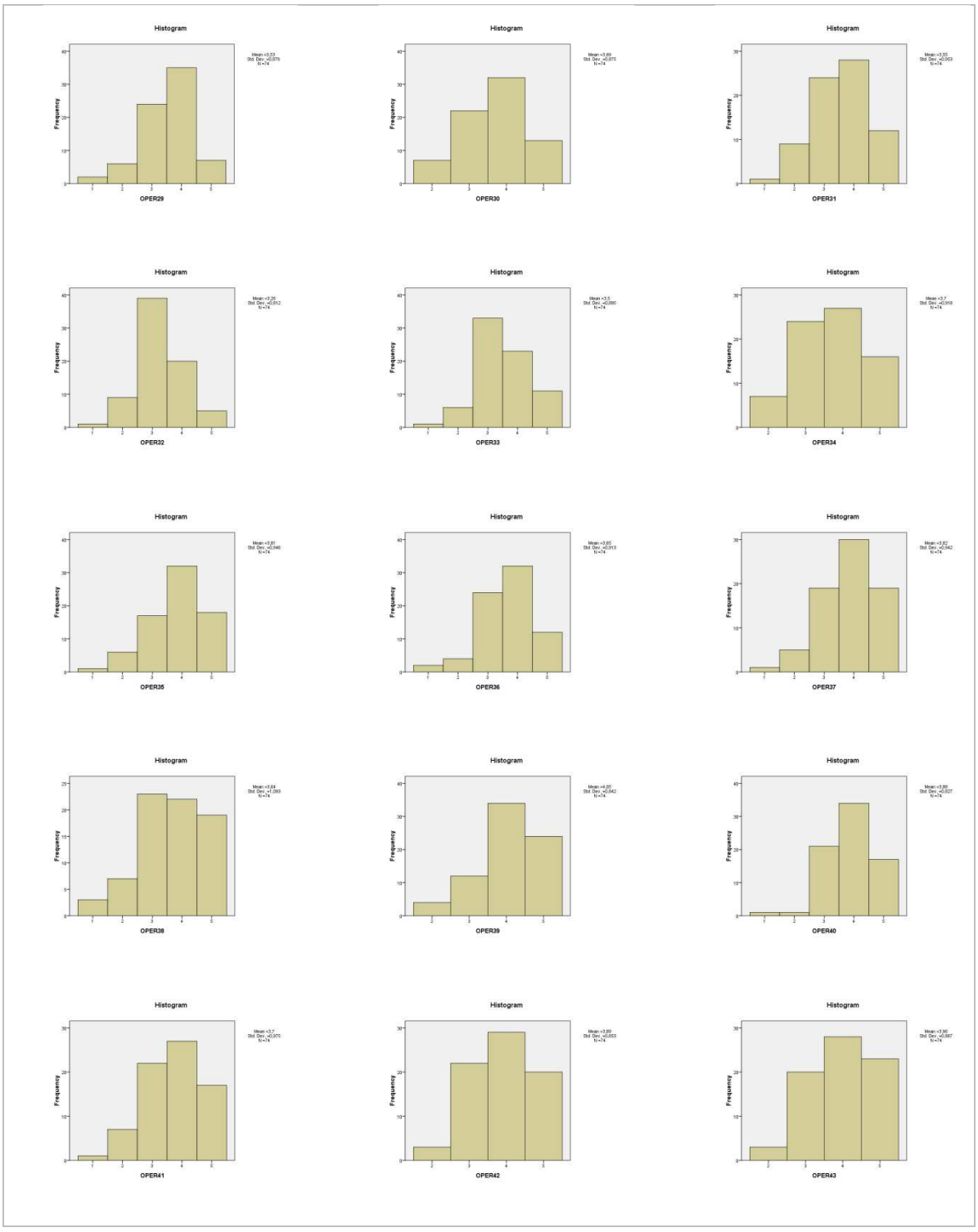
c) Operación:

						Test de Kolmogorov-Smirnoff		
		Media	Mediana	Moda	Desv. estándar	Estad.	gl	Sig.
OPER27	Velocidad de interacción no impuesta por la aplicación.	3,527	4	4	0,879	0,272	74	0,000
OPER28	Opciones auto-configurables pueden modificarse.	3,689	4	4	0,875	0,247	74	0,000
OPER29	Finalización de cualquier proceso del producto en ejecución.	3,554	4	4	0,953	0,221	74	0,000
OPER30	Soporte a la accesibilidad.	3,257	3	3	0,812	0,286	74	0,000
OPER31	Control de la información proporcionada por la aplicación.	3,500	3	3	0,895	0,252	74	0,000
OPER32	Interrupción de los diálogos en cualquier momento.	3,703	4	4	0,918	0,208	74	0,000
OPER33	Selección del modo de hacer las tareas según perfil y preferencias de usuario.	3,811	4	4	0,946	0,255	74	0,000
OPER34	Valor recomendado cuando existen diferentes opciones a elegir.	3,649	4	4	0,913	0,244	74	0,000
OPER35	Control sobre los datos presentados.	3,824	4	4	0,942	0,236	74	0,000

OPER36	Ejecución de las acciones con ratón y con teclado	3,635	4	3	1,093	0,185	74	0,000
OPER37	Almacenamiento de los datos de salida en formato estándar para su uso posterior.	4,054	4	4	0,842	0,258	74	0,000
OPER38	El diálogo muestra un aspecto coherente.	3,878	4	4	0,827	0,248	74	0,000
OPER39	Aviso al usuario si el tiempo de espera va a ser superior al esperado	3,703	4	4	0,975	0,214	74	0,000
OPER40	Soporte a la prevención de errores de entrada.	3,892	4	4	0,853	0,213	74	0,000
OPER41	Ayuda a la corrección de entradas de datos erróneas.	3,959	4	4	0,867	0,208	74	0,000
OPER42	Validación de los datos de entrada introducidos.	4,081	4	5	0,962	0,236	74	0,000
OPER43	Datos de entrada erróneos no provocan fallos en la aplicación.	4,068	4	5	1,139	0,253	74	0,000
OPER44	Posibilidad de volver a los valores predeterminados en cualquier momento.	3,878	4	4	0,979	0,198	74	0,000
OPER45	No se pierde la información que se acaba de introducir tras un error.	4,311	4	5	0,720	0,290	74	0,000
OPER46	En los mensajes de error se proporciona información sobre la acción a tomar.	4,284	4	5	0,785	0,265	74	0,000
OPER47	Notificación al usuario de los errores corregidos de forma automática con posibilidad de cancelarlo.	3,514	3	3	1,010	0,221	74	0,000
OPER48	Obtención de información adicional sobre un error.	3,959	4	4	0,784	0,250	74	0,000
OPER49	Aviso y necesaria confirmación para acciones destructivas (como borrado de datos).	4,338	5	5	0,848	0,336	74	0,000
OPER50	Corrección de errores sin cambiar a otro diálogo.	3,932	4	4	0,833	0,235	74	0,000
OPER51	Documentación en formatos y forma operativa.	3,838	4	4	0,937	0,258	74	0,000
	TOTAL MUESTRA	74						

Tabla 28. Análisis descriptivo para las variables correspondientes a la sub-característica de Operación.

En relación a la normalidad sólo la variable OPER36 sigue una distribución normal (nivel de significación α=0,01).


Histogram
Frequency
OPER29
Histogram
Frequency
OPER30
Histogram
Frequency
OPER31
Histogram
Frequency
OPER32
Histogram
Frequency
OPER33
Histogram
Frequency
OPER34
Histogram
Frequency
OPER35
Histogram
Frequency
OPER36
Histogram
Frequency
OPER37
Histogram
Frequency
OPER38
Histogram
Frequency
OPER39
Histogram
Frequency
OPER40
Histogram
Frequency
OPER41
Histogram
Frequency
OPER42
Histogram
Frequency
OPER43

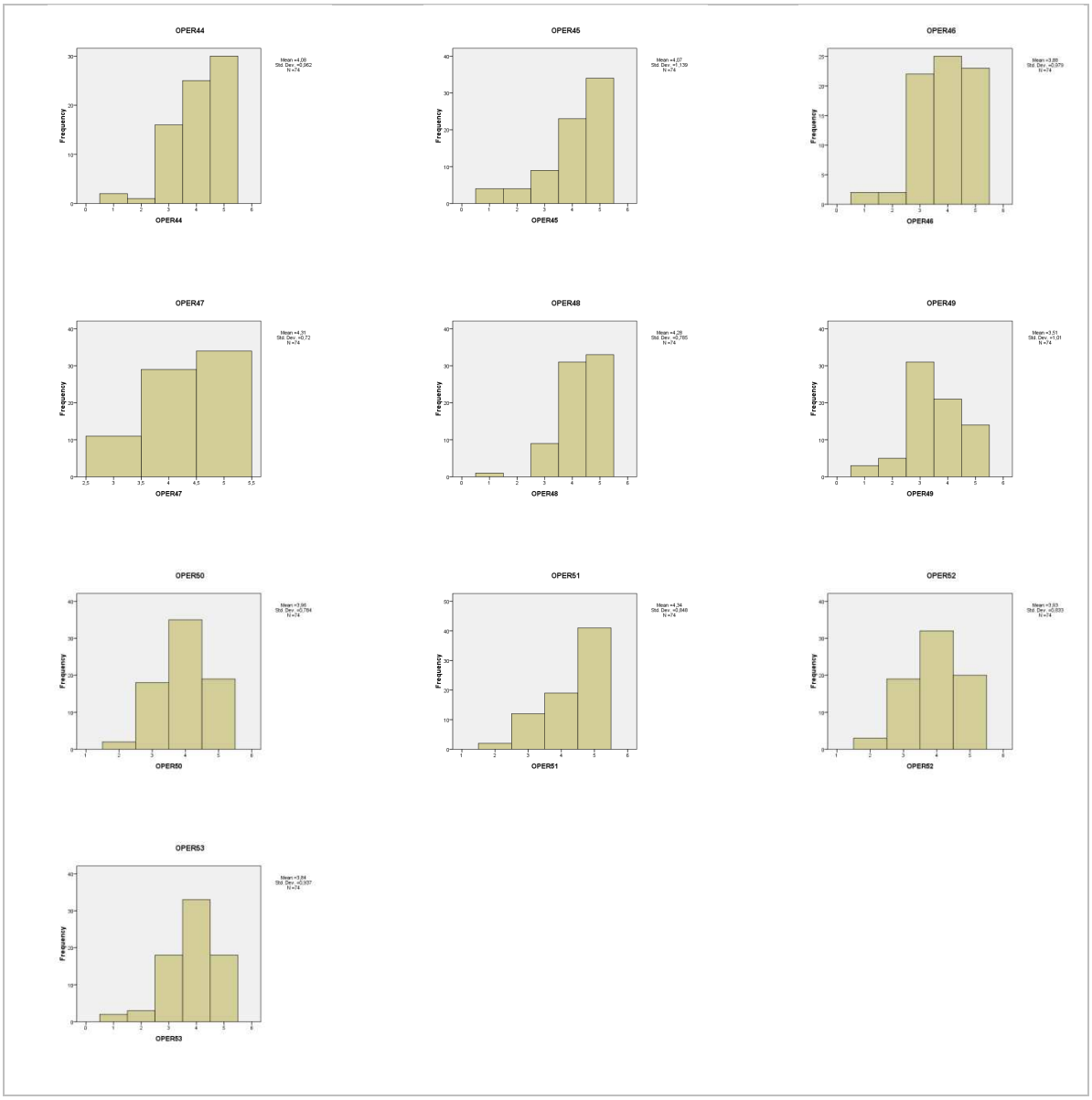


Figura 37. Histogramas para las variables correspondientes a la sub-característica de Operación.

d) Apariencia:

						Test de Kolmogorov-Smirnoff		
		Media	Mediana	Moda	Desv. estándar	Estad.	gl	Sig.
APAR52	Personalización de la apariencia del programa para ajustarla al gusto del usuario.	2,878	3	3	1,020	0,209	74	0,000
APAR53	Combinación de colores y fondo estéticamente agradable.	2,824	3	3	0,970	0,234	74	0,000
	TOTAL MUESTRA	74						

Tabla 29. Análisis descriptivo para las variables correspondientes a la sub-característica de Apariencia.

En relación a la normalidad ninguna de las variables sigue una distribución normal.

Figura 38. Histogramas para las variables correspondientes a la sub-característica de Apariencia.

Conclusiones obtenidas tras el análisis preliminar de las variables de datos

El coeficiente de variación obtenido fue menor que 0,5 en todos los casos lo que demuestra que los datos obtenidos son muy homogéneos. Por otra parte, el análisis descriptivo preliminar no permitía obtener importantes conclusiones ya que no era posible discriminar la importancia de cada criterio con los datos obtenidos mediante los estadísticos descriptivos. Tan sólo había 2 variables cuyos estadísticos destacan ligeramente entre los demás (media y mediana menores o iguales que 3, desviación típica menor que 1 y asimetría negativa o sesgada a la izquierda). Las variables eran las siguientes:

- APAR55: Posibilidad de personalizar la apariencia del software para poder adaptarla a los requisitos del usuario.

- APAR56: Posibilidad de parametrizar la apariencia del software para mejorar su apariencia.

Para el resto de factores: las medias son mayores de 3.5, las medianas que 4 y la desviación típica baja. Tampoco los histogramas dan mucha información sobre la importancia de los factores.

Por otra parte, parece razonable pensar que haya relaciones entre las variables. Por ejemplo, la velocidad de interacción (criterio de usabilidad) podría estar relacionada con el tiempo de respuesta (criterio de eficiencia). Para verificar estas relaciones, analizamos la matriz de correlaciones. Una primera inspección visual mostró que había un alto número de correlaciones superior a 0,30, lo que demuestra que las variables están correlacionadas [Hair, Tatham 1998]. Además, la mayoría de los niveles de significación son cercanos a cero por lo que se rechaza la hipótesis nula y se concluye que hay relación lineal entre las variables. Por último, los determinantes son cercanos a cero también (4,97E-010 para el cuestionario de NTF, 6,81E-024 para el de TF y 4,11E-025 para el UF) lo que confirma que las variables están altamente correlacionadas unas con otras.

4.1.3.4 Análisis de aplicabilidad del modelo teórico

Para verificar la aplicabilidad del modelo se realizaron dos acciones. Por una parte, se aplicaron los factores obtenidos en la fase 2.2 a tres casos de estudio de evaluaciones de productos COTS de seguridad para fabricantes de software y se les solicitó retroalimentación sobre su nivel de satisfacción tanto con el proceso como con los resultados obtenidos. El análisis de dichas evaluaciones y los resultados obtenidos se describieron en la sección 3.2.1. Por otra parte, se consultó a los expertos, una vez mostrado el catálogo de factores a través de los cuestionarios, sobre la utilidad de un modelo de calidad como el expuesto. Para ello, la parte final del estudio incluía la solicitud de información relacionada con la utilidad práctica de un modelo de calidad con el objetivo de validar la aplicabilidad práctica del mismo. Las preguntas incluidas fueron las siguientes:

1. Asumiría un coste extra en la adquisición de productos software de seguridad TIa cambio de una mayor calidad del software en un rango entre 1 (nunca) y 5 siempre);

2. Cree que sería útil disponer de un modelo práctico para la selección de productos de seguridad informática que tenga en cuenta todas las características que expertos en calidad del software, seguridad TI y dirección consideran importantes? donde las respuestas posibles eran “sí” o “no".

Los resultados obtenidos mostraron que un 94.48% de los participantes consideraron que es útil tener un modelo de calidad para los productos software de seguridad TI (mediana=1, media=0.94, moda=1, desviación típica=0.22). Con respecto a la importancia de la calidad del software, tal como se muestra en la Figura 39, la mayoría de los expertos consideraron que la calidad del software es muy importante en el proceso de selección de productos de seguridad TI (media=3.85, mediana=4, moda=4, desviación típica =0.75).

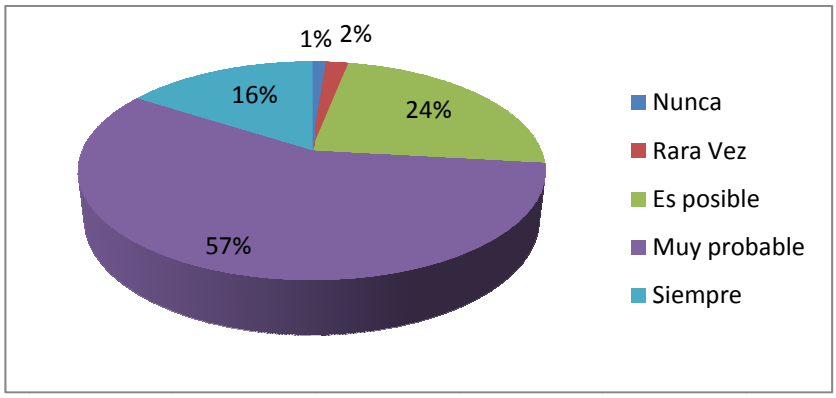


Figura 39. Distribución de respuestas de los expertos a la pregunta sobre la importancia de la calidad del software en el proceso de selección de productos de seguridad TIC.

4.1.3.5 Análisis de la fiabilidad de la estructura del modelo propuesto

Con el fin de determinar la fiabilidad de las escalas de medida utilizadas, se realiza un primer análisis individual de las mismas. Como ya se mencionó al inicio de este capítulo, la fiabilidad es el grado en el que la variable observada (atributo o sub-característica) mide el valor verdadero y está libre de error. Por tanto, el objetivo de esta fase será maximizar la fiabilidad de las escalas de medida utilizadas, o lo que es lo mismo minimizar el error de medida, permitiéndonos así obtener al final del proceso de validación un modelo fiable. Para ello, las variables se agrupan en sus características (comúnmente denominadas dimensiones) y, para cada una de éstas, se examinan el coeficiente α de Cronbach y las correlaciones ítem-total (correlación de la variable con la puntuación de la suma total de todas las

variables de la dimensión). Las variables eliminadas son aquellas que reducen la consistencia de la escala (coeficiente α de Cronbach) o tienen una insuficiente relación con el concepto que se está midiendo (correlaciones ítem-total). Todo ello determinado por el hecho de no superar el nivel mínimo definido para los coeficientes utilizados. Estos mínimos son el 0,70 para el α de Cronbach [J. Nunnally 1978, Flynn, Curran 2002, Thomsen 2002] y el 0,50 para la correlación ítem-total [Hair, Tatham 1998]. En las siguientes secciones se muestran los resultados obtenidos para los tres modelos de medida teóricamente obtenidos en las fases anteriores.

Análisis de la fiabilidad para el modelo de factores técnicos

El análisis de la fiabilidad se ha llevado a cabo individualmente para cada una de las dimensiones (funcionalidad, fiabilidad, usabilidad, eficiencia, mantenimiento y portabilidad) a través de un proceso iterativo en el que se eliminan aquellas variables que no cumplen los niveles mínimos de los coeficientes establecidos. El resultado final se muestra en la Tabla 30. En dicha tabla puede observarse que las variables FIAB14, FIAB18, EFIC26 se han mantenido a pesar de tener una correlación ítem-total inferior a 0,5. La variable FIAB14 se ha mantenido porque el valor de correlación ítem-total es muy cercano a 0,5 (0,495). Además, el análisis descriptivo mostrado en la sección 4.1.3.3 arrojó resultados favorables para estas tres variables con medias altas (3,93, 4,097 y 3,944 respectivamente), medianas y modas de 4 para todas ellas e histogramas con ausencia del valor 1 (importancia relativa nula) y asimetría negativa (mayor concentración a la derecha). Es decir, la mayor parte de los expertos consideró importante estas características. Por otra parte, el coeficiente α de Cronbach no mejora si se eliminan ninguna de estas variables y, dado que el coeficiente de correlación ítem-total mide la relación de las variables con respecto al total, es posible que el análisis factorial exploratorio las relacione con otra de las características o dimensiones contempladas (es el caso, por ejemplo, de la variable FIAB18 relacionada también con usabilidad al tratarse de documentación y con el mantenimiento o administración del software). Por tanto, dado que los valores de correlación no están muy por debajo del valor de corte, se ha considerado importante mantenerlas de momento para decidir según los resultados del análisis factorial exploratorio (y/o confirmatorio) sobre la conveniencia de eliminarlas o no.

	Factor	Correlación item-total	α de Cron bach
	FUNCIONALIDAD		
FUNC1	Exactitud de los resultados obtenidos tras la ejecución de las funciones	0,515	
FUNC2	Uso de interfaces estándares	0,592	0,736
FUNC3	Requisitos HW	0,583	
FUNC4	Requisitos SW	0,598	
	FIABILIDAD		
FIAB8	Porcentaje de fallos (madurez del producto)	0,541	
FIAB9	Tiempo necesario para obtener un parche (tiempo trascurrido entre que el fabricante investiga el fallo y desarrolla y prueba el parche)	0,691	
FIAB10	Los fallos software no interfieren con otros programas o sistema operativos.	0,518	
FIAB11	Número critico de fallos.	0,654	
FIAB12	Fallos leves no interfieren con otras funciones.	0,627	0,902
FIAB13	Fallos graves no interfieren con funciones críticas.	0,524	
FIAB14	Capacidad de volver a un estado normal después de un fallo.	0,495	
FIAB15	Tiempo de no disponibilidad después de un fallo.	0,569	
FIAB16	Capacidad de volver a un estado previo después de un evente anormal (*restore*)	0,600	
FIAB17	Capacidad de volver a un estado normal después de un fallo (*recovery*)	0,598	
FIAB18	Documentación de apoyo a la recuperación (*recovery*)	0,435	
	USABILIDAD		
USAB21	Comprensión	0,697	
USAB22	Aprendizaje	0,771	0,853
USAB23	Operabilidad	0,755	
	EFICIENCIA		
EFIC26	Tiempo de respuesta	0,450	
EFIC27	Capacidad de procesamiento por unidad de tiempo (*throughput*)	0,511	
EFIC28	Uso de memoria	0,831	0,819
EFIC29	Uso de procesador	0,738	
EFIC30	Uso de espacio en disco	0,642	
EFIC31	EScalabilidad	0,527	
	MANTENIMIENTO		
MANT33	Tiempo para obtener una actualización (desde que se descubre la vulnerabilidad hasta que los usuarios disponene de la actualización)	0,749	
MANT34	Facilidad de instalación de actualizaciones	0,812	
MANT35	Esfuerzo requerido por el usuario para actualizar el producto.	0,677	0,896
MANT36	Estabilidad de los parches	0,790	
MANT37	Capacidad de volver a un estado previo después de la instalación de un parche o cambio de version (*upgrade*)	0,745	
MANT38	Cambios de version no eliminan configuración previa.	0,668	
	PORTABILIDAD		
PORT40	Facilidad de instalación	0,720	
PORT41	Esfuerzo de instalación	0,743	
PORT42	Documentación de instalación	0,578	
PORT45	Retroalimentación al usuario sobre los procesos en los que hay que esperar	0,819	0,930
PORT46	Asistencia a la configuración	0,749	
PORT47	Documentación y ayuda para la configuración	0,716	

Tabla 30. Resultado del análisis de fiabilidad para el modelo TF.

Destacar también que la variable relacionada con la apariencia del software (USAB24) ha sido eliminada por reducir la fiabilidad del análisis. Esto es coherente con los resultados obtenidos en el análisis descriptivo.

Análisis de la fiabilidad para el modelo de factores no técnicos

Al igual que en el caso anterior el análisis de la fiabilidad se ha llevado a cabo individualmente para cada una de las dimensiones con el fin de obtener escalas fiables para cada una de las dimensiones consideradas. El resultado final se muestra en la Tabla 31.

	Factor	Correlación item-total	α de Cronbach
	PROVEEDOR	0,570	
PROV12	Cuota de mercado del fabricante	0,611	
PROV13	Reputación	0,678	
PROV14	Solvencia	0,628	0,846
PROV15	Experiencia	0,464	
PROV17	Autonomía	0,530	
PROV18	Certificación de calidad del servicio	0,550	
	PRODUCTO	0,621	
PROD21	Compatibilidad con la arquitectura del cliente	0,611	
PROD22	Compatibilidad con la política de seguridad corporativa	0,527	
PROD23	Estabilidad del producto en el mercado	0,560	
PROD25	Tipo de licencia	0,567	0,823
PROD27	Tipo de soporte	0,547	
PROD28	Oferta de formación	0,569	
TECN9	Certificación de seguridad del producto	0,570	
TECN10	Conformidad con estándares de calidad	0,611	

Tabla 31. Resultado del análisis de fiabilidad para el modelo NTF.

Un resultado a destacar de este análisis es la eliminación de la variable relacionada con el coste total del producto. Dado que el análisis descriptivo en este caso arrojaba valores significativos para dicha variable (mediana=4, media=3.94, moda=4, desviación estándar=0.74), tras el cálculo de la fiabilidad de la dimensión que contiene dicha variable (producto), se calculó también la fiabilidad de la escala global donde se comprobó también la necesaria eliminación de la variable. Se deduce, por tanto, que dicha variable no está

relacionada con el resto y, en consecuencia, debe considerarse de forma separada del proceso de evaluación de la calidad del software de productos COTS de seguridad (quizás después de la evaluación de la calidad del software cuando se dispone de los resultados del resto de variables o antes para reducir el número de productos a evaluar).

Análisis de la fiabilidad para el modelo de atributos de usabilidad

En la Tabla 32 se muestran los resultados obtenidos en el caso del modelo para los factores de usabilidad.

	Factor	Correlación item-total	α de Cronbach
	COMPRENSIÓN		
COMPR1	Compresión de la ayuda proporcionada por el programa.	0,663	
COMPR2	Comprensión de la retroalimentación proporcionada por el programa.	0,572	
COMPR3	Comprensión de la información proporcionada en los diálogos del programa.	0,647	
COMPR4	Localización eficiente de información.	0,679	0,883
COMPR6	Agrupación de la información adecuada para mejorar la comprensión.	0,694	
COMPR7	Mensajes de formulados de forma constructiva, objetiva y comprensible.	0,631	
COMPR8	Comprensión global del diálogo mediante la retroalimentación.	0,664	
COMPR10	Ayuda y/o retroalimentación sobre la naturaleza de los datos a introducir.	0,512	
COMPR11	Lenguaje simple y directo.	0,581	
	APRENDIZAJE		
APREN15	Interfaz único de acceso a la aplicación para facilitar el aprendizaje.	0,645	
APREN16	Distinción clara de los accesos a las distintas funcionalidades de la aplicación.	0,713	
APREN17	Claridad del acceso a los mecanismos de apoyo al uso de la aplicación (ayuda, tutoriales, etc.).	0,767	
APREN18	Información básica sobre aspectos conceptuales del programa en la ayuda.	0,626	0,909
APREN19	Ayuda global del programa.	0,610	
APREN20	Ayuda de cada función particular asociada al diálogo de ejecución correspondiente.	0,597	
APREN21	Ejemplos de aplicación para facilitar el aprendizaje en la ayuda.	0,615	
APREN23	Uso de las funcionalidades más utilizadas para mejorar la experiencia del usuario.	0,624	
APREN25	Funcionalidades poco usadas se acompañan de información	0,600	

	más completa de uso.		
APREN26	Ubicación similar para el mismo tipo de mensajes.	0,707	
APREN27	Disposición de pantalla similar para tareas similares.	0,715	
	OPERACIÓN		
OPER30	Opciones auto-configurables pueden modificarse.	0,544	
OPER31	Finalización de cualquier proceso del producto en ejecución.	0,538	
OPER32	Soporte a la accesibilidad.	0,555	
OPER34	Interrupción de los diálogos en cualquier momento.	0,586	
OPER35	Selección del modo de hacer las tareas según perfil y preferencias de usuario.	0,619	
OPER36	Valor recomendado cuando existen diferentes opciones a elegir.	0,630	
OPER37	Control sobre los datos presentados.	0,671	
OPER38	Ejecución de las acciones con ratón y con teclado	0,676	
OPER39	Almacenamiento de los datos de salida en formato estándar para su uso posterior.	0,621	0,940
OPER40	El diálogo muestra un aspecto coherente.	0,603	
OPER41	Aviso al usuario si el tiempo de espera va a ser superior al esperado	0,539	
OPER42	Soporte a la prevención de errores de entrada.	0,739	
OPER43	Ayuda a la corrección de entradas de datos erróneas.	0,761	
OPER44	Validación de los datos de entrada introducidos.	0,683	
OPER45	Datos de entrada erróneos no provocan fallos en la aplicación.	0,525	
OPER46	Posibilidad de volver a los valores predeterminados en cualquier momento.	0,622	
OPER47	No se pierde la información que se acaba de introducir tras un error.	0,691	
OPER48	En los mensajes de error se proporciona información sobre la acción a tomar.	0,664	

Tabla 32. Resultado del análisis de fiabilidad para el modelo UF.

Como puede observarse en la Tabla 32, las variables relacionadas con la sub-característica Apariencia se han eliminado ya que aumentaban el error de medida. Esto es coherente con los resultados obtenidos en el análisis descriptivo, tal como se indicaba en la sección 4.1.3.3.

Por otra parte puede observarse que también se han eliminado las variables relacionadas con la adaptación de la información al perfil del usuario (APREN24 y COMPR9), lo cual tiene sentido al estar dirigidos a un grupo específico de usuarios (ingenieros, consultores y administradorse de seguridad) y, además, con altos conocimientos en el dominio de aplicación.

4.1.3.6 Desarrollo del modelo basado en la teoría: validación predictiva

Antes de llevar a cabo el análisis factorial es necesario verificar si se cumplen los supuestos básicos subyacentes del análisis factorial exploratorio.

Desarrollo del modelo de factores técnicos

En relación a los índices que muestran la viabilidad para aplicar el análisis factorial exploratorio todos ellos arrojaron valores positivos. Los valores obtenidos pueden verse en la Tabla 33. El método de rotación convergió tras 9 iteraciones.

En esta tabla se muestran también las variables finales junto con su descripción y las cargas factoriales para cada una de las 8 dimensiones obtenidas. Dichas dimensiones se han etiquetado atendiendo al criterio de variables con cargas superiores a 0,5 mencionado en la sección 4.1.3.6 sobre cada dimensión del siguiente modo:

1. Interoperabilidad. Facilidad de interacción con otros sistemas especificados.
2. Estabilidad (madurez y tolerancia a fallos). Frecuencia de fallos (debida a fallos en el software) y capacidad de mantener un nivel especificado de funcionamiento en caso de fallo.
3. Recuperación. Capacidad de restablecer el nivel de funcionamiento normal en caso de fallo.
4. Eficiencia con respecto al tiempo. Capacidad del producto de proporcionar tiempos de respuesta y proceso adecuados bajo condiciones determinadas [ISO 2001a].
5. Utilización de recursos. La capacidad del producto software para usar las cantidades y tipos de recursos adecuados cuando el software lleva a cabo su función bajo condiciones determinadas [ISO 2001a].
6. Usabilidad. Conjunto de atributos relacionados con el esfuerzo requerido para el uso del software, y la valoración individual de tal uso por el conjunto de usuarios establecido o implicado [ISO 2001a].
7. Capacidad de actualización. Atributos relacionados con la capacidad para mantener el software sin fallos o vulnerabilidades de seguridad y con las últimas tecnologías.

8. Soporte. Facilidad de gestión del software (instalación, configuración, actualización, cambios de versión, etc.).

		Cargas del análisis factorial exploratorio (tras la rotación varimax)							
	Factor	Soporte	Capacidad de actualización	Estabilidad	Recuperación	Utilización de recursos	Eficiencia temporal	Usabilidad	Interoperabilidad
	FUNCIONALIDAD								
FUNC2	Uso de interfaces estándares		0,36						0,56
FUNC3	Requisitos HW								0,88
FUNC4	Requisitos SW			0,35					0,83
	FIABILIDAD								
FIAB8	Porcentaje de fallos (madurez del producto)		0,45	0,61					
FIAB9	Tiempo necesario para obtener un parche (tiempo trascurrido entre que el fabricante investiga el fallo y desarrolla y prueba el parche)		0,38	0,69					
FIAB10	Los fallos software no interfieren con otros programas o sistema operativos.			0,78					
FIAB11	Número critico de fallos.			0,79					
FIAB12	Fallos leves no interfieren con otras funciones.			0,78					
FIAB13	Fallos graves no interfieren con funciones críticas.			0,87					
FIAB14	Capacidad de volver a un estado normal después de un fallo.				0,72		0,36		
FIAB15	Tiempo de no disponibilidad después de un fallo.			0,42	0,65				
FIAB16	Capacidad de volver a un estado previo después de un evente anormal (*restore*)		0,36		0,66				
FIAB17	Capacidad de volver a un estado normal después de un fallo (*recovery*)			0,32	0,72				
FIAB18	Documentación de apoyo a la recuperación (*recovery*)		0,45	0,61					
USABILIDAD									
USAB21	Comprensión	0,55						0,64	
USAB22	Aprendizaje							0,86	
USAB23	Operabilidad							0,78	
EFICIENCIA									
EFIC26	Tiempo de respuesta		0,32	0,36			0,59		
EFIC27	Capacidad de procesamiento por unidad de tiempo (*throughput*)			0,49			0,56		
EFIC28	Uso de memoria					0,92			
EFIC29	Uso de procesador					0,84			
EFIC30	Uso de espacio en disco	0,38				0,74			
EFIC31	EScalabilidad		0,45		0,34	0,56			
MANTENIMIENTO									
MANT33	Tiempo para obtener una actualización		0,57	0,37					

	(desde que se descubre la vulnerabilidad hasta que los usuarios disponene de la actualización)		
MANT36	Estabilidad de los parches		0,74
MANT37	Capacidad de volver a un estado previo después de la instalación de un parche o cambio de version (*upgrade*)	0,34	0,73
MANT38	Cambios de version no eliminan configuración previa.		0,70
	PORTABILIDAD		
PORT40	Facilidad de instalación	0,84	
PORT41	Esfuerzo de instalación	0,87	
PORT42	Documentación de instalación	0,74	
PORT46	Asistencia a la configuración	0,82	
PORT47	Documentación y ayuda para la configuración	0,72	
Medida KMO (Kaiser-Meyer-Olkin) de adecuación de la muestra			0,79
Test de Esfericidad de Bartlett (Approx. Chi-Square)			2050,612 (p<0.001)
Determinante de la matriz de correlación			5,98E-016
Varianza total explicada			76,921%

Tabla 33. Resultados del análisis factorial exploratorio para el modelo TF[28].

Los resultados de este análisis permiten predecir las relaciones de influencia entre las variables que pertenecen a diferentes dimensiones. Como puede observarse en la Tabla 33, hay variables con cargas factoriales superiores a 0,32 con influencia en varias dimensiones. Por ejemplo, las variables MANT33 (relativa al tiempo necesario para obtener una actualización) y MANT38 (relativa a la posibilidad de mantener las configuraciones previas tras los cambios de versión) se encuentran bajo la dimensión "capacidad de actualización" pero ambas tienen influencia sobre la dimensión de "soporte" la cual consiste en compresión, facilidad y esfuerzo de instalación (tanto del producto como de las actualizaciones), documentación de instalación y proceso de configuración.

Desarrollo del modelo de factores no técnicos

En este caso también se cumplieron los supuestos para la aplicación del análisis factorial exploratorio tal como se muestra en la Tabla 34 convergiendo el método de rotación tras 5 iteraciones. Los componentes extraídos para el caso del modelo de factores no técnicos se etiquetaron de la siguiente forma:

[28] En la tabla no se muestran las cargas factoriales inferiores a 0,32 por no considerarse significativas.

1. Criterios relacionados con el fabricante. Criterios tales como experiencia, reputación, cuota de mercado del fabricante, etc.
2. Criterios no técnicos relacionados con el producto. Criterios relacionados con cuestiones organizacionales tales como el tipo de licencia o el tipo de soporte ofertado para ese producto concreto.
3. Conformidad con los estándares. Que se disponga de certificaciones tanto de producto (de seguridad o de calidad) como del servicio ofrecido por el fabricante

Como puede observarse en este caso las dimensiones prácticamente se confirmaron a excepción de que se extrajo una componente adicional relacionada con las certificaciones de calidad (tanto de producto como de fabricante). Además, en este caso las influencias son más reducidas que en el caso de los factores técnicos. Tan sólo se observa carga factorial de la variable PROV15 (relacionada con la experiencia del proveedor) en la dimensión referente a los criterios de producto. También se observan influencias de las variables PROD22 (relacionada con la capacidad del producto para facilitar el cumplimiento de la política corporativa) y PROD23 (relacionada con la estabilidad del producto en el mercado) sobre la dimensión que aglutina los criterios relacionados con el fabricante.

	Factor	Cargas del análisis factorial exploratorio (tras la rotación varimax)		
		Criterios relacionados con el producto	Criterios relacionados con el fabricante	Conformidad con estándares
	PROVEEDOR			
PROV12	Cuota de mercado del fabricante		0.87	
PROV13	Reputación		0.79	
PROV14	Solvencia		0.75	
PROV15	Experiencia	0.40	0.58	
PROV17	Autonomía		0.62	
PROV18	Certificación de calidad del servicio			0.63
	PRODUCTO			
PROD21	Compatibilidad con la arquitectura del cliente	0.76		
PROD22	Compatibilidad con la política de seguridad corporativa	0.69	0.35	
PROD23	Estabilidad del producto en el mercado	0.54	0.45	
PROD25	Tipo de licencia	0.76		
PROD27	Tipo de soporte	0.71		
PROD28	Oferta de formación	0.69		
TECN9	Certificación de seguridad del producto			0.83
TECN10	Conformidad con estándares de calidad			0.81
	Medida KMO (Kaiser-Meyer-Olkin) de adecuación de la muestra	0.842		
	Test de Esfericidad de Bartlett (Approx. Chi-Square)			363,014 (p<0.001)
	Determinante de la matriz de correlación			0.001
	Varianza total explicada			62,319%

Tabla 34. Resultados del análisis factorial exploratorio para el modelo NTF.

Desarrollo del modelo de usabilidad

Los valores obtenidos para el caso del modelo de los factores de usabilidad se muestran en la Tabla 35. El método de rotación convergió tras 28 iteraciones. En este caso se obtuvieron 7 dimensiones con un total del 70,59% de la varianza explicada que fueron etiquetadas del siguiente modo:

1. Tolerancia al error. Un diálogo es tolerante a errores cuando a pesar de un error evidente de entrada, puede alcanzarse un resultado sin acción del usuario o con una mínima corrección. La tolerancia a error se logra por medio de alguno de los siguientes medios:
    a. control de errores (control de daños);
    b. corrección de errores;
    c. gestión de errores para hacer frente a los errores que aparezca [ISO 2006].

2. Ayuda y adecuación al aprendizaje. Un diálogo es adecuado al aprendizaje cuando ayuda y guía al usuario en el aprendizaje de la utilización del sistema [ISO 2006].
3. Capacidad para ser entendido (comprensión). La capacidad del producto software que permite al usuario entender si el software es adecuado y cómo puede ser usado para unas tareas o condiciones de uso particulares [ISO 2001a].
4. Soporte a la entrada/salida. La capacidad del programa para proporcionar ayuda y utilidades al usuario que faciliten la entrada de datos y posterior la manipulación de los datos de salida.
5. Facilidad de operación. Capacidad del programa para facilitar la operación del mismo a través de mecanismos de retroalimentación durante su operación, consideración de las funcionalidades más usadas y facilidades de accesibilidad para usuarios con discapacidades.
6. Soporte a la administración. Facilidades que proporciona el programa para el uso avanzado del mismo.
7. Diálogos. Facilidades que proporciona el programa para una mejor comprensión, aprendizaje y operación de los diálogos.

		Cargas del análisis factorial exploratorio (tras la rotación varimax)						
	Factor	Tolerancia al error	Ayuda y adecuación al aprendizaje	Capacidad para ser entendido	Soporte a la entrada/salida	Facilidad de operación	Soporte a la administración	Diálogos
	COMPRENSIÓN							
COMPR1	Compresión de la ayuda proporcionada por el programa.			0,80				
COMPR2	Comprensión de la retroalimentación proporcionada por el programa.			0,45		0,49		
COMPR3	Comprensión de la información proporcionada en los diálogos del programa.			0,79				
COMPR4	Localización eficiente de información.			0,75				
COMPR6	Agrupación de la información adecuada para mejorar la comprensión.			0,66		0,34		
COMPR7	Mensajes de formulados de forma constructiva, objetiva y comprensible.			0,54		0,51		
COMPR8	Comprensión global del diálogo mediante la retroalimentación.			0,44		0,58		
COMPR10	Ayuda y/o retroalimentación sobre la naturaleza de los datos a introducir.					0,70		
COMPR11	Lenguaje simple y directo.		0,56	0,33				
	APRENDIZAJE							

APREN15	Interfaz único de acceso a la aplicación para facilitar el aprendizaje.		0,77				
APREN16	Distinción clara de los accesos a las distintas funcionalidades de la aplicación.		0,86				
APREN17	Claridad del acceso a los mecanismos de apoyo al uso de la aplicación (ayuda, tutoriales, etc.).		0,79				
APREN19	Ayuda global del programa.		0,58				
APREN20	Ayuda de cada función particular asociada al diálogo de ejecución correspondiente.		0,32				0,58
APREN21	Ejemplos de aplicación para facilitar el aprendizaje en la ayuda.		0,34		0,57	0,32	0,38
APREN23	Uso de las funcionalidades más utilizadas para mejorar la experiencia del usuario.					0,56	0,34
APREN26	Ubicación similar para el mismo tipo de mensajes.		0,54				0,60
APREN27	Disposición de pantalla similar para tareas similares.		0,59				0,54
	OPERABILIDAD						
OPER32	Soporte a la accesibilidad.					0,55	0,35
OPER34	Interrupción de los diálogos en cualquier momento.	0,36	0,32			0,42	0,38
OPER35	Selección del modo de hacer las tareas según perfil y preferencias de usuario.	0,42					0,55
OPER36	Valor recomendado cuando existen diferentes opciones a elegir.				0,56	0,47	
OPER37	Control sobre los datos presentados.				0,68		0,42
OPER38	Ejecución de las acciones con ratón y con teclado				0,56		0,53
OPER39	Almacenamiento de los datos de salida en formato estándar para su uso posterior.	0,39					0,70
OPER40	El diálogo muestra un aspecto coherente.	0,62					0,49
OPER42	Soporte a la prevención de errores de entrada.	0,77			0,33		
OPER43	Ayuda a la corrección de entradas de datos erróneas.	0,78			0,36		
OPER44	Validación de los datos de entrada introducidos.	0,43			0,75		
OPER46	Posibilidad de volver a los valores predeterminados en cualquier momento.	0,40			0,63	0,35	
OPER47	No se pierde la información que se acaba de introducir tras un error.	0,71					
OPER48	En los mensajes de error se proporciona información sobre la acción a tomar.	0,36		0,38	0,66		
OPER50	Obtención de información adicional sobre un error.	0,66					0,41
OPER51	Aviso y necesaria confirmación para acciones destructivas (como borrado de datos).	0,67					
OPER52	Corrección de errores sin cambiar a otro diálogo.	0,78					
OPER53	Documentación en formatos y forma operativa.	0,48					0,59
Medida KMO (Kaiser-Meyer-Olkin) de adecuación de la muestra							0,953
Test de Esfericidad de Bartlett (Approx. Chi-Square)							1999,835 (p<0.001)
Determinante de la matriz de correlación							3,67E-015
Varianza total explicada							70,59%

Tabla 35. Resultados del análisis factorial exploratorio para el modelo UF.

La dimensión "Diálogos" cuenta tan sólo con 2 variables (APREN20 y APREN26), cuya carga sea la máxima sobre dicha dimensión. Además, estas 2 variables tienen cargas significativas sobre otras dimensiones. Dado que es recomendable tener al menos 3 variables significativas en cada una de las dimensiones [Hair, Tatham 1998], se verificará el modelo confirmatorio con la dimensión "Diálogos" y sin ella comparando los valores obtenidos y eligiendo el modelo que mejor se ajuste.

4.1.3.7 Confirmación e interpretación del modelo final: validación confirmativa

El siguiente paso en el proceso DuMoD es la confirmación de los modelos obtenidos. El modelo de medida de ecuaciones estructurales o SEM proporciona la transición entre el análisis factorial exploratorio y confirmatorio a través de una serie de ecuaciones de regresión múltiple distintas pero interrelacionadas mediante la especificación del modelo estructural. SEM, a diferencia del análisis factorial exploratorio, permite controlar el error de medida y realizar un test estadístico de calidad del ajuste para las solución confirmatoria propuesta, lo cual, permite una validación nomológica del modelo. SEM es la única técnica que permite examinar simultáneamente una serie de relaciones de dependencia múltiples y cruzadas que constituyen un modelo a gran escala, capaz de representar conceptos no observados y tener en cuenta el error de medida en el proceso de estimación [Hair, Tatham 1998]. Para llevar a cabo este análisis se han definido una serie de etapas que pasamos a describir a continuación.

Etapa 1: Modelo de ecuaciones estructurales

Después de desarrollar el modelo teórico, éste debe especificarse en términos formales. Para ello, a continuación se plantean las ecuaciones estructurales correspondientes a cada uno de los modelos a validar:

1. Modelo de ecuaciones estructurales para los factores técnicos:

Interoper. $= \lambda_{11}FUNC2 + \lambda_{12}FUNC3 + \lambda_{13}FUNC3 + \varepsilon_1$

Estabilidad $= \lambda_{21}FIAB8 + \lambda_{22}FIAB9 + \lambda_{23}FIAB10 + \lambda_{24}FIAB11 + \lambda_{25}FIAB12 + \lambda_{26}FIAB13 + \lambda_{27}FIAB18 + \varepsilon_2$

Recuperación $= \lambda_{31}FIAB14 + \lambda_{32}FIAB15 + \lambda_{33}FIAB16 + \lambda_{34}FIAB17 + \varepsilon_3$

Usabilidad $= \lambda_{41}USAB21 + \lambda_{42}USAB22 + \lambda_{43}USAB23 + \varepsilon_4$

UsoRecursos $= \lambda_{51}EFIC28 + \lambda_{52}EFIC29 + \lambda_{53}EFIC30 + \lambda_{54}EFIC31 + \varepsilon_5$

EficTemporal $= \lambda_{61}EFIC26 + \lambda_{62}EFIC27 + \varepsilon_6$

Actualización $= \lambda_{71}MANT33 + \lambda_{72}MANT36 + \lambda_{73}MANT37 + \lambda_{74}MANT38 + \varepsilon_7$

Soporte $= \lambda_{81}PORT40 + \lambda_{82}PORT41 + \lambda_{83}PORT42 + \lambda_{84}PORT46 + \lambda_{85}PORT47 + \varepsilon_8$

Siendo:

- λ_{ij} los coeficientes de regresión que relacionan las variables latentes (dimensiones) con los indicadores (variables observadas),
- ε_i los errores de medida para las variables observadas.

2. Modelo de ecuaciones estructurales para los factores no técnicos:

Proveedor $= \gamma_{11}PROV12 + \gamma_{12}PROV13 + \gamma_{13}PROV14 + \gamma_{14}PROV15 + \gamma_{15}PROV17 + \xi_1$

Producto $= \gamma_{21}PROD21 + \gamma_{22}PROD22 + \gamma_{23}PROD23 + \gamma_{24}PROD25 + \gamma_{25}PROD27 + \gamma_{26}PROD28 + \xi_2$

Estándares $= \gamma_{31}PROV18 + \gamma_{32}TECN9 + \gamma_{33}TECN10 + \xi_3$

Siendo:

- γ_{ij} los coeficientes de regresión que relacionan las variables latentes (dimensiones) con los indicadores (variables observadas),
- ξ_i los errores de medida para las variables observadas.

3. Modelo de ecuaciones estructurales para usabilidad:

   En el caso del modelo de usabilidad definimos las ecuaciones para los dos casos que vamos a estimar:

   - El modelo con 7 dimensiones:

     Tol_error $= \alpha_{11}OPER40 + \alpha_{12}OPER42 + \alpha_{13}OPER43 + \alpha_{14}OPER47 + \alpha_{15}OPER50 + \alpha_{16}OPER51 + \alpha_{17}OPER52 + \eta_1$

     Aprendizaje $= \alpha_{21}COMPR11 + \alpha_{22}APREN15 + \alpha_{23}APREN16 + \alpha_{24}APREN17 + \alpha_{25}APREN19 + \alpha_{26}APREN27 + \eta_2$

     Comprensión $= \alpha_{31}COMPR1 + \alpha_{32}COMPR3 + \alpha_{33}COMPR4 + \alpha_{34}COMPR6 + \alpha_{35}COMPR7 + \eta_3$

     SoporteE_S $= \alpha_{41}OPER36 + \alpha_{42}OPER37 + \alpha_{43}OPER38 + \alpha_{44}OPER44 + \alpha_{45}OPER46 + \alpha_{46}OPER48 + \alpha_{47}APREN21 + \eta_4$

     FacilidadOper $= \alpha_{51}COMPR2 + \alpha_{52}COMPR8 + \alpha_{53}COMPR10 + \alpha_{54}APREN23 + \alpha_{55}OPER32 + \eta_5$

$SoporteAdmin = \alpha_{61}OPER35 + \alpha_{62}OPER39 + \alpha_{63}OPER53 + \eta_6$

$Diálogos = \alpha_{71}APREN20 + \alpha_{72}APREN26 + \eta_7$

Siendo:

- α_{ij} los coeficientes de regresión que relacionan las variables latentes (dimensiones) con los indicadores (variables observadas),
- η_i los errores de medida para las variables observadas.

- El modelo con 6 dimensiones:

$Tol\_error = \alpha_{11}OPER40 + \alpha_{12}OPER42 + \alpha_{13}OPER43 + \alpha_{14}OPER47 + \alpha_{15}OPER50 + \alpha_{16}OPER51 + \alpha_{17}OPER52 + \eta_1$

$Aprendizaje = \alpha_{21}COMPR11 + \alpha_{22}APREN15 + \alpha_{23}APREN16 + \alpha_{24}APREN17 + \alpha_{25}APREN19 + \alpha_{26}APREN27 + \alpha_{27}APREN20 + \alpha_{28}APREN26 + \eta_2$

$Comprensión = \alpha_{31}COMPR1 + \alpha_{32}COMPR3 + \alpha_{33}COMPR4 + \alpha_{34}COMPR6 + \alpha_{35}COMPR7 + \eta_3$

$SoporteE\_S = \alpha_{41}OPER36 + \alpha_{42}OPER37 + \alpha_{43}OPER38 + \alpha_{44}OPER44 + \alpha_{45}OPER46 + \alpha_{46}OPER48 + \alpha_{47}APREN21 + \eta_4$

$FacilidadOper = \alpha_{51}COMPR2 + \alpha_{52}COMPR8 + \alpha_{53}COMPR10 + \alpha_{54}APREN23 + \alpha_{55}OPER32 + \eta_5$

$SoporteAdmin = \alpha_{61}OPER35 + \alpha_{62}OPER39 + \alpha_{63}OPER53 + \eta_6$

Siendo:

- α_{ij} los coeficientes de regresión que relacionan las variables latentes (dimensiones) con los indicadores (variables observadas),
- η_i los errores de medida para las variables observadas.

Etapa 2: Estimación del modelo

Para la estimación del modelo se utilizará el método de estimación máximo verosimilitud robusto con el fin de solucionar los problemas de no normalidad de los datos [Byrne 1994] cuestión detectada en la mayoría de las variables utilizadas tal como se mostró en la sección 4.1.3.3. Para ello, se han introducido las ecuaciones estructurales en LISREL 8.8 obteniendo los siguientes modelos:

1. Modelo final de factores técnicos

En la Figura 40 se muestra el modelo estimado. Los números que aparecen en la columna más a la izquierda son los pesos de cada factor sobre la dimensión correspondiente, mientras que los que aparecen a la derecha son los errores estimación.

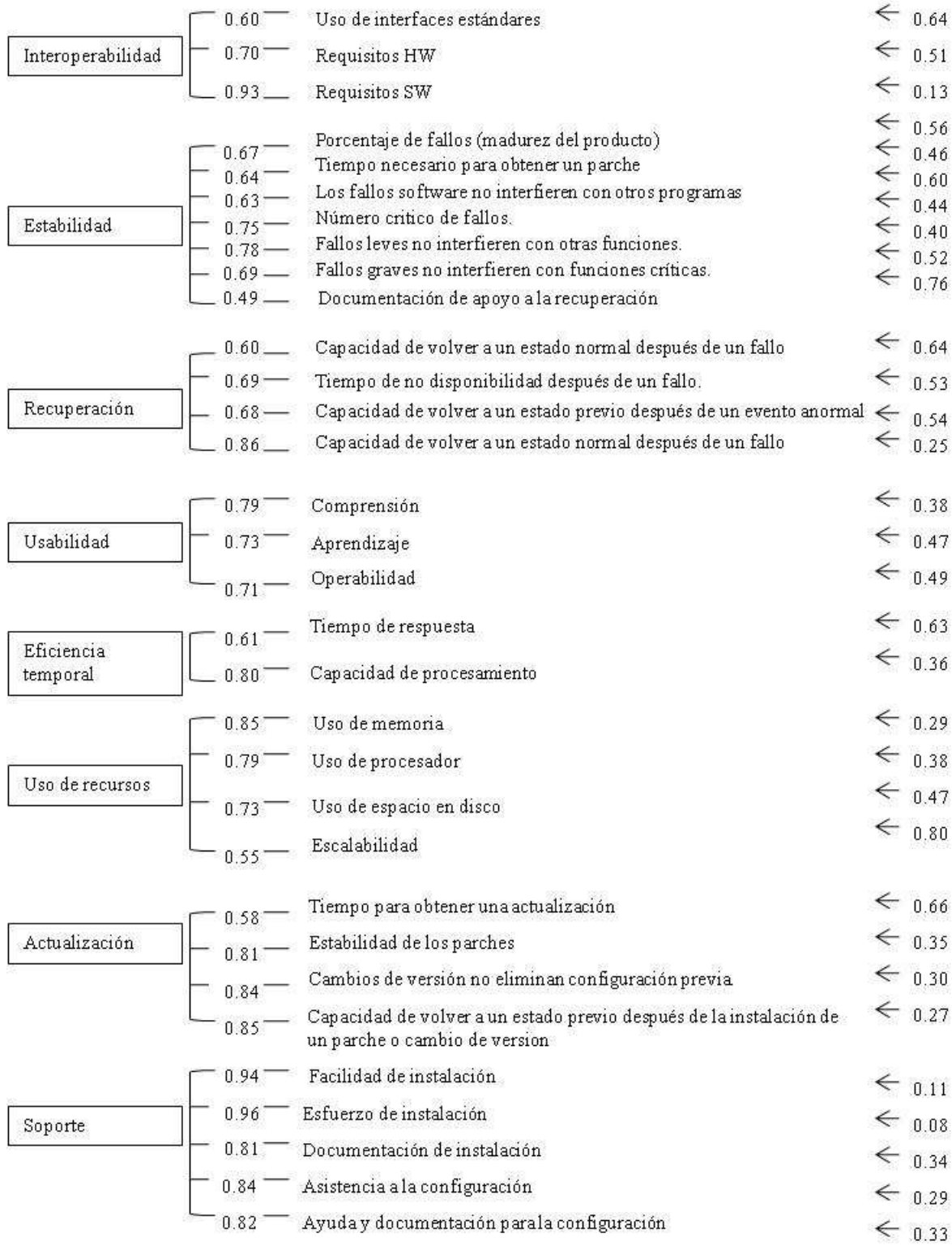


Figura 40. Modelo estimado para factores técnicos.

	Interoperabilidad	Estabilidad	Recuperación	Usabilidad	EficTemporal	UsoRecursos	Actualización	Soporte
Interoperabilidad	1,000							
Estabilidad	0,510	1,000						
Recuperación	0,410	0,720	1,000					
Usabilidad	0,180	0,340	0,480	1,000				
EficTemporal	0,550	0,710	0,490	0,440	1,000			
UsoRecursos	0,320	0,420	0,370	0,410	0,560	1,000		
Actualización	0,490	0,720	0,690	0,470	0,710	0,530	1,000	
Soporte	0,180	0,270	0,480	0,650	0,500	0,520	0,680	1,000

Tabla 36. Correlaciones entre dimensiones para el modelo de factores técnicos

2. Modelo final de factores no técnicos

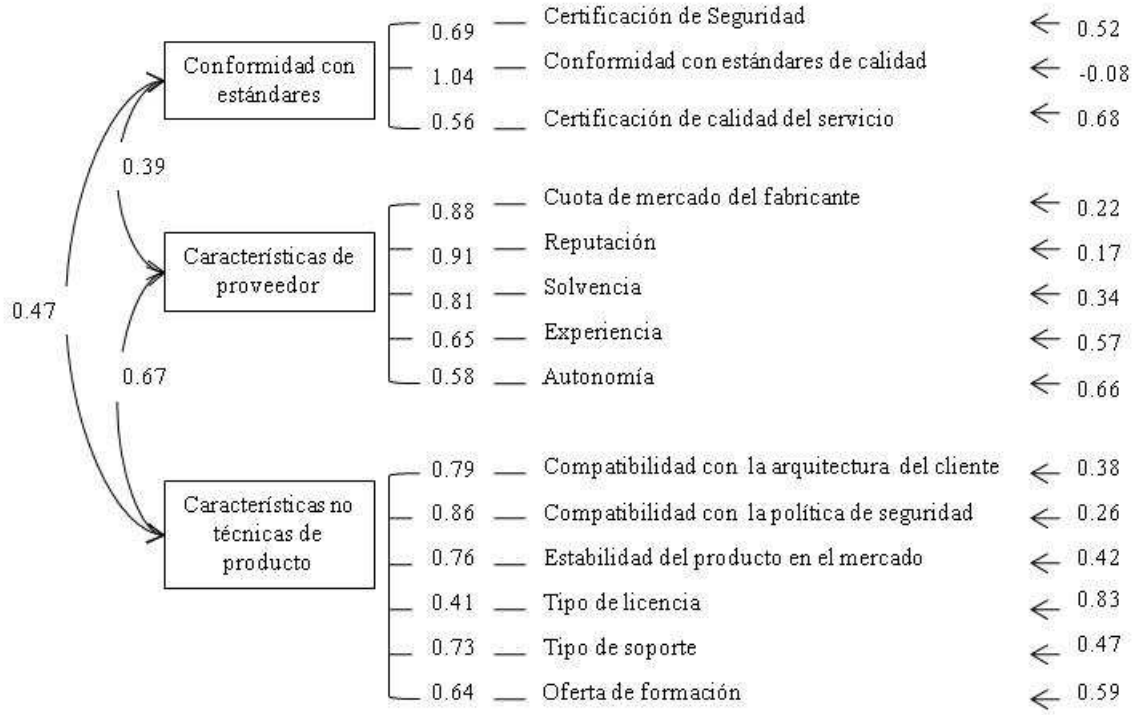


Figura 41. Modelo estimado para factores no técnicos.

En el modelo NTF la ponderación de la variable TECN10 (Conformidad con estándares de calidad) es mayor que 1. Además, el error de medida tiene un valor negativo (conocido

como caso Heywood) [Hair, Tatham 1998] para la misma variable. Se trata de una estimación infractora. Tales estimaciones son teóricamente inapropiadas y deben corregirse antes de la interpretación del modelo. Para ello, existen 2 opciones [Bentler y Chou 1987, Dillon, Kumar*, et al.* 1987]:

- eliminar la variable con estimación infractora obteniendo estadísticos de ajuste apropiados para la aceptación del modelo,
- retener la variable y fijar la varianza del error de dicha variable a un valor reducido (0,01).

Con el fin de obtener el modelo más óptimo, en este caso se ha optado por reestimar el modelo con ambas opciones y seleccionar el que mejor ajuste estadístico muestra. En la siguiente sección (fase 3.3.5, etapa 3) se muestra la discusión sobre los estadísticos de bondad de ajuste en ambos casos (Tabla 39). El modelo final obtenido puede verse en la Figura 42.

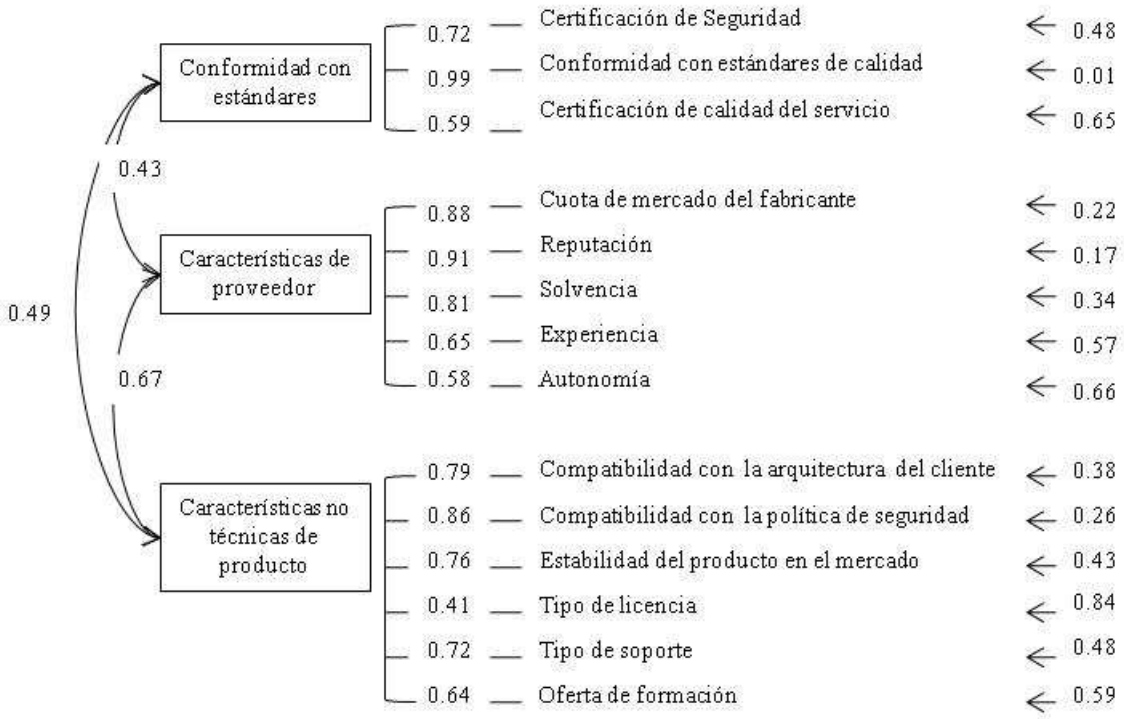


Figura 42. Modelo revisado para factores no técnicos.

3. Modelo final de Usabilidad

Las figuras siguientes muestran los modelos estructurales obtenidos tras aplicar el análisis confirmatorio a los dos casos contemplados: Figura 43 para el caso en el que consideramos 7 dimensiones y Figura 44 para el caso de las 6 dimensiones. Tal como reflejan ambas figuras no hay ninguna variable con estimación infractora, por lo que, se consideran válidos ambos modelos.

Figura 43. Modelo UF para el caso de 7 dimensiones.

Las correlaciones para este caso se muestran en la Tabla 37.

	Tolerancia a errores	Ayuda y adecuación al aprendizaje	Capacidad para ser entendido	Soporte E/S	Facilidad de operación	Soporte a la administración	Diálogos
Tolerancia a errores	1,000						
Ayuda y adecuación al aprendizaje	0,543	1,000					
Capacidad para ser entendido	0,462	0,653	1,000				
Soporte E/S	0,773	0,458	0,634	1,000			
Facilidad de operación	0,562	0,621	0,784	0,763	1,000		
Soporte a la administración	0,832	0,509	0,405	0,701	0,593	1,000	
Diálogos	0,541	0,894	0,552	0,515	0,735	0,551	1,000

Tabla 37. Correlaciones entre dimensiones para el modelo UF de 7 dimensiones

Figura 44. Modelo UF para el caso de 6 dimensiones.

Las correlaciones para este caso se muestran en la Tabla 38.

	Tolerancia a errores	Ayuda y adecuación al aprendizaje	Capacidad para ser entendido	Soporte E/S	Facilidad de operación	Soporte a la administración
Tolerancia a errores	1,000					
Ayuda y adecuación al aprendizaje	0,555	1,000				
Capacidad para ser entendido	0,462	0,646	1,000			
Soporte E/S	0,773	0,477	0,632	1,000		
Facilidad de operación	0,562	0,652	0,782	0,761	1,000	
Soporte a la administración	0,833	0,529	0,406	0,702	0,590	1,000

Tabla 38. Correlaciones entre dimensiones para el modelo UF de 6 dimensiones

Etapa 3: Análisis de la calidad de ajuste

En la Tabla 39 se muestran los valores de corte utilizados, así como, los valores obtenidos para la bondad de ajuste de los modelos NTF revisados para eliminar la estimación infractora obtenida en el paso anterior. Aunque los valores son muy buenos en ambos casos (todos los estadísticos de calidad del ajuste superan los umbrales recomendados), el modelo obtenido para el caso de no eliminación de la variable fijando la varianza del error a 0,01, presenta el mejor ajuste de los datos. Además, según el análisis descriptivo mostrado en la sección 4.1.3.3, los expertos consultados consideran importante tener en cuenta las certificaciones de calidad del producto en el proceso de evaluación del mismo: moda=mediana=4, desviación típica=0,765 e histograma con asimetría negativa y con ausencia casi total de valores 1 y 2 (un único experto en cada caso). Por todo ello, se decide retener la variable infractora.

Estadísticos de bondad de ajuste

	Valores de corte	Modelo NTF1 (eliminando variable)	Modelo NTF2 (fijar la varianza del error a 0,01)
χ^2 (df)[1]		99,802 (62)	116.837(75)
S- χ^2	>1, <2	1,610	1,558
RMSEA	< 0,08	0,0710	0,068
CFI	> 0,9	0,978	0,979
NFI	> 0,9	0,945	0,944
NNFI	> 0,9	0,973	0,975

1. p=0,001

Tabla 39. Comparación de estadísticos de bondad de ajuste para los dos posibles modelos NTF revisados

En el caso del modelo de calidad para los factores de usabilidad, en la Tabla 40 se muestran los valores obtenidos para los dos modelos contemplados junto con los valores sugeridos de corte. En ambos casos los valores obtenidos superan los valores de ajuste recomendados con valores similares en ambos casos. Por tanto, dado que el modelo con 7 dimensiones infringe la recomendación de tener, al menos, 3 variables por dimensión, obtamos por seleccionar como definitivo el modelo de 6 dimensiones (correspondiente a la Figura 44).

Estadísticos de calidad de ajuste

	Valores sugeridos de corte	Modelo UF con 7 dimensiones	Modelo UF con 6 dimensiones
χ^2 (df)		644.95 (539) [1]	662.07 (545) [2]
S- χ^2	>1, <2	1.20	1.21
RMSEA	< 0.08	0.052	0.054
CFI	> 0.9	0.984	0.982
NFI	> 0.9	0.910	0.908
NNFI	> 0.9	0.982	0.981

1. p=0,001, 2. p=0,00042

Tabla 40. Comparación de estadísticos de bondad de ajuste para los dos posibles modelos UF revisados

Por último, en la Tabla 41 puede verse el ajuste obtenido para los 3 modelos finales. En todos los casos se superan los umbrales recomendados, lo cual demuestra la calidad de los modelos obtenidos.

Estadísticos de bondad de ajuste

	Valores de corte	Modelo TF	Modelo NTF	Modelo UF
χ^2 (df)		725.54 (436)	116.923(74)	662.07 (545)
S- χ^2	>1, <2	1,66	1,58	1.21
RMSEA	< 0,08	0,0789	0,0692	0.054
CFI	> 0,9	0,900	0,979	0.982
NFI	> 0,9	0,907	0,944	0.908
NNFI	> 0,9	0,955	0,974	0.981

Tabla 41. Estadísticos de bondad de ajuste para los tres modelos.

Etapa 4: Interpretación de los modelos finales

En el caso del modelo TF, en la Tabla 42 se muestran los coeficientes de correlación para las variables latentes, es decir, el grado de dependencia entre las dimensiones obtenidas en el análisis factorial exploratorio ahora confirmadas a través del análisis factorial confirmatorio. Tal como muestra la tabla, las dependencias más altas son aquellas entre estabilidad y recuperación y también entre estabilidad y actualización. Conceptualmente teniendo en cuenta la definición de estas dimensiones, el resultado obtenido es coherente. Por una parte, la estabilidad del programa favorece una mayor capacidad de recuperación frente a fallos producidos en el mismo y viceversa, a mayor capacidad de recuperación más estable será el programa. En lo que respecta a la relación entre estabilidad y actualización, dentro de la dimensión o sub-característica actualización, se contemplan factores relacionados con la estabilidad de los parches obtenidos, por tanto, se justifica también la relación obtenida. En relación a los pesos de los factores sobre su correspondiente dimensión, según los datos mostrados en la Figura 40, en el caso de la dimensión "estabilidad", el factor "documentación de apoyo a la recuperación" tiene un peso más bajo que el resto de factores, lo cual, parece lógico dado que el resto de factores está relacionado con la ausencia o no de fallos en el sistema que conceptualmente tiene mayor importancia

sobre la estabilidad que la existencia o no de documentación. En cuanto a la dimensión "uso de recursos", el factor "escalabilidad" tiene un peso bajo con respecto al resto. Nótese que dicho factor no proviene del modelo de calidad definido en ISO/IEC 9126 si no que fue introducido según publicaciones relacionadas y la propia experiencia. El modelo, por tanto, confirma la necesidad de este factor aunque su importancia sea menor que la de otros factores de la misma sub-característica. Por último, en la dimensión "actualización" el factor "tiempo para obtener una actualización" tiene un peso más bajo que el resto de factores relacionados con la estabilidad de los parches o su capacidad para no provocar errores, por lo que se deduce que es más importante que las actualizaciones sean estables que el tiempo en sí que se tarde en obtenerlas, o lo que es lo mismo,los expertos prefieren que los fabricantes tarden más tiempo en desarrollar las actualizaciones a cambio de que éstas sean estables.

	Interoperabilidad	Estabilidad	Recuperación	Usabilidad	EficTemporal	UsoRecursos	Actualización	Soporte
Interoperabilidad	1,000							
Estabilidad	0,510	1,000						
Recuperación	0,410	0,720	1,000					
Usabilidad	0,180	0,340	0,480	1,000				
EficTemporal	0,550	0,710	0,490	0,440	1,000			
UsoRecursos	0,320	0,420	0,370	0,410	0,560	1,000		
Actualización	0,490	0,720	0,690	0,470	0,710	0,530	1,000	
Soporte	0,180	0,270	0,480	0,650	0,500	0,520	0,680	1,000

Tabla 42. Coeficientes de correlación para las variables latentes del modelo TF

Con respecto al modelo NTF, tal como se muestra en la Figura 42, la correlación más alta es aquella entre los factores relacionados con el producto y los relacionados con el proveedor del producto. Esto justifica la alta dependencia ya mostrada anteriormente por otros autores [Kontio, Caldiera 1996, Ochs, Pfahl 2001, Comella-Dorda, Dean 2002] como un único grupo de criterios llamado "no técnico", "organizacional" o "estratégico". Un

resultado a destacar en relación a este modelo, tal como se comentó anteriormente, es la eliminación de la variable relacionada con el coste de adquisición del producto por tener una baja correlacción con respecto al resto de características. Una posibilidad es que el coste deba tenerse en cuenta a parte de las medidas, tal como se propone en [Comella-Dorda, Dean 2004]. Por otra parte, según los datos de la Figura 42, el peso de los factores "experiencia" y "autonomía" sobre la dimensión "características relacionadas con el proveedor" es menor que el del resto de factores, lo mismo ocurre con "tipo de licencia" sobre la dimensión "características relacionadas con el producto". Otro resultado a destacar es el de los pesos obtenidos para la dimensión "conformidad con estándares". El peso más alto es el correspondiente al factor de "conformidad con estándares de calidad", seguido de "certificaciones de seguridad" y, por último, de "certificaciones de calidad del servicio".

Por último, en el caso del modelo UF la Tabla 43 muestra unos valores muy altos de correlación. Esto indica una alta influencia entre los factores obtenidos. Aunque son altas también, las relaciones más bajas se observan entre los factores: "Capacidad para ser entendido" y "Soporte a la administración" (0,406); "Capacidad para ser entendido" y "Tolerancia a errores" (0,462), y "Ayuda y adecuación al aprendizaje" y "Soporte a la entrada/salida" (0,477). El resto de las relaciones superan el 0,5 de correlación. Además, en la Figura 44 no se observan diferencias significativas en los pesos de los factores sobre la dimensión correspondiente, por lo que se deduce que la importancia de cada uno de los factores sobre su dimensión es similar.

	Tolerancia a errores	Ayuda y adecuación al aprendizaje	Capacidad para ser entendido	Soporte E/S	Facilidad de operación	Soporte a la administración
Tolerancia a errores	1,000					
Ayuda y adecuación al aprendizaje	0,555	1,000				
Capacidad para ser entendido	0,462	0,646	1,000			
Soporte E/S	0,773	0,477	0,632	1,000		
Facilidad de operación	0,562	0,652	0,782	0,761	1,000	
Soporte a la administración	0,833	0,529	0,406	0,702	0,590	1,000

Tabla 43. Coeficientes de correlación para las variables latentes para el modelo UF

4.2. Resumen de los resultados obtenidos y conclusiones

A través de la aplicación del proceso DuMoD se han obtenido 3 modelos de calidad aplicables al dominio de los productos COTS de seguridad informática. Entre las principales características de dichos modelos cabe destacar las siguietnes:

1. Los modelos han sido validados a través de la opinión de expertos en diferentes áreas involucradas en los procesos de calidad del software de los productos de seguridad y, en concreto, en la evaluación de la misma.
2. En la fase 1 del proceso se ha definido de forma teórica un catálogo de factores a través de la aplicación de estándares, publicaciones y otra documentación técnica relacionada, así como, la experiencia propia, obteniendo finalmente tras diferentes revisiones internas un catálogo compuesto por un total de 111 factores. Dichos factores se han ido reduciendo en la fase 3 del proceso a través de la aplicación de análisis multivariante a los datos recogidos de expertos, obteniendo un catálogo final de 78 factores lo que supone una reducción del 29,73% con respecto al catálogo inicial. Además, se ha probado la aplicabilidad, fiabilidad y validez (de contenido, discriminante y nomológica) de estos factores.

3. Además, a partir del catálogo y de los datos recogidos de expertos se ha obtenido un modelo que establece no sólo el peso o importancia de cada uno de los factores dentro de su sub-característica, sino que además, cuantifica la influencia entre las distintas características, lo cual, puede resultar de gran interés para la aplicación de los métodos de agregación en las evaluaciones cuantitativas. Dichos pesos sólo tienen que normalizarse para poder ser aplicados directamente en los métodos de agregación o, al menos, para servir de apoyo en el proceso de toma de decisiones sobre los pesos finales a considerar teniendo en cuenta la importancia otorgada por expertos en la materia.

Capítulo 5. Conclusiones y líneas futuras de investigación

Este capítulo describe las conclusiones obtenidas en la investigación llevada a cabo en esta tesis, así como, las aportaciones principales en relación con los objetivos planteados al inicio de este trabajo. Además, se identifican las posibles líneas de trabajo futuro relacionadas con esta tesis.

El capítulo está organizado de la siguiente forma. En la sección 5.1 se enumeran las principales conclusiones obtenidas en este trabajo, mientras que en la sección 5.2 se describen las posibles ampliaciones de la misma, así como, las nuevas líneas de investigación que ha dado lugar esta tesis.

5.1.Conclusiones

Antes de enunciar las conclusiones obtenidas durante la investigación llevada a cabo en este trabajo, pasamos a presentar en la Tabla 44, a modo de resumen, los objetivos enunciados en la sección 1.2 junto con los resultados que apoyan la consecución de los mismos.

OBJETIVO	RESULTADO	CAPÍTULO/SECCIÓN
Metodología formal y sistemática de apoyo a la construcción de modelos de calidad orientados a dominio reutilizables y validados.	Proceso DuMoD	Capítulo 3
Obtención de un modelo de calidad que permita la evaluación y selección de productos para la protección de sistemas informáticos a través de la aplicación de la metodología definida.	Figura 40, Figura 42, Figura 44	Capítulo 4
Estudio de los procesos actuales de evaluación de calidad de productos finales para validar la posibilidad de integrar en ellos un modelo de calidad predefinido.	Síntesis de cada uno de los procesos y razonamiento fundamentado sobre su capacidad de adaptación o no.	Capítulo 2/ sección 2.2
Análisis de los modelos de calidad orientados a dominio existentes y de su método de construcción con el fin de encontrar uno que cumpla los requisitos necesarios para cumplir nuestro propósito o bien, en caso contrario, integrar todo lo aprendido en ellos en un nuevo proceso.	Revisión sistemática de los procesos de construcción utilizados en los modelos de calidad existentes y comparación de los mismos con respecto a los requisitos predefinidos (Tabla 3).	Capítulo 2/ secciones 2.3 y 2.4
Validación empírica del modelo utilizando la opinión de expertos en el área y a través del tratamiento estadístico de los datos recogidos.	Análisis estadístico de los datos recogidos.	Capítulo 4./ sección 4.1.3

Tabla 44. Objetivos planteados y resultados que apoyan su consecución.

La necesidad de disponer de un modelo de calidad para evaluar productos finales de seguridad, nos llevó a buscar un proceso de construcción sistemático que permitiera la obtención no sólo de un catálogo de atributos sino también de las relaciones entre las características del modelo y, sobre todo, de los pesos de los atributos sobre cada una de las características. En la investigación llevada a cabo, llegamos a la conclusión de que no existía ningún método que englobase todos los requisitos necesarios para el desarrollo de nuestro modelo, por lo que era necesario definir un nuevo proceso que comprendiera estas características. Además, en nuestra búsqueda de información, llegamos también a la

conclusión de que la única forma viable de evaluar el modelo era teniendo en cuenta la opinión de expertos en el área con experiencia en diferentes proyectos y conocimiento en las diferentes áreas involucradas en los procesos de adquisición de software. Como resultado de todo ello y, aplicando técnicas de análisis multivariante muy utilizadas en la investigación médica y también en las ciencias sociales, se ha definido un proceso de desarrollo de modelos de calidad que, posteriormente, se ha aplicado a un caso de estudio. Si nos centramos en los resultados obtenidos para los objetivos principales, podemos afirmar que:

1. La propuesta define formalmente un proceso sistemático de construcción de modelos de calidad orientados a dominios de aplicación. Los modelos de calidad obtenidos con el proceso DuMoD son reutilizables dentro del dominio de aplicación definido.
2. De los datos recogidos por los cuestionarios realizados a expertos, se deduce que la propuesta es válida y útil en el entorno industrial. La Tabla 45 muestra los resultados en los que se apoya esta afirmación.

CRITERIO A EVALUAR	**PREGUNTA**	**RESPUESTA**
Importancia de la calidad del software en los productos de seguridad informática	Asumiría un coste extra en la adquisición de productos software de seguridad TI a cambio de una mayor calidad del software en un rango entre 1 y 5.	Media=3,85 Mediana=4 Moda=4
Utilidad de un modelo de calidad como el mostrado en el estudio	Cree que sería útil disponer de un modelo práctico para la selección de productos de seguridad informática que tenga en cuenta todas las características que expertos en calidad del software, seguridad TI y dirección consideran importantes? donde las respuestas posibles eran “sí” o “no".	94,48% contestaron que SÍ.

Tabla 45. Resultados sobre la utilidad e importancia del modelo.

3. El proceso es aplicable como se demuestra en el Capítulo 4 en el que se aplica al caso de estudio de los productos de seguridad informática. Dado que la validez de los modelos y metodologías es un proceso difícil de validar ya que sería necesaria su aplicación en el entorno empresarial en diferentes y variados proyectos y, para ello, se requiere de años de pruebas, por lo que ese tipo de validación se escapa de los objetivos de este trabajo de tesis y, por tanto, se ha optado por la validación a través de un caso de estudio como parte de la misma. El catálogo de atributos obtenido teóricamente durante la primera fase del proceso DuMoD, ha sido utilizado en diferentes proyectos de evaluación [Villalba y Fernández-Sanz 2007a, Villalba y Fernández-Sanz 2007b, Villalba y Fernández-Sanz 2008], además, se recogió información de retroalimentación de los clientes de la evaluación recibiendo resultados satisfactorios en todos los casos, por lo que se considera también probada la aplicabilidad de los factores obtenidos. Además de la aplicabilidad de los factores, se ha probado a través del análisis de los datos recogidos de expertos, la fiabilidad y validez (de contenido, discriminante y nomológica) de los mismos. Por tanto, el modelo de calidad orientado al dominio de productos COTS de seguridad informática obtenido en el Capítulo 4, supone una mejora con respecto al utilizado anteriormente:
    a. al llevarse a cabo una reducción del 29,73% sobre el catálogo de factores inicial, dicha reducción aumentará la eficiencia de los futuros procesos de evaluación;
    b. al obtener los pesos de importancia de cada uno de los factores y, por tanto, permitir realizar también evaluaciones cuantitativas;
    c. por último, al haber sido validado por expertos, se aumentará también la eficacia de los futuros procesos de evaluación.

Además, cabe destacar los siguientes importantes resultados obtenidos en paralelo a esta investigación:

- Se ha probado la existencia de relaciones de influencia entre los distintos criterios del modelo antes citada por otros autores [Punter, Solingen 1997], [Veenendaal y Trienekens 1997] y [Carvallo, Franch 2003] pero todavía no empíricamente confirmada.
- Se han obtenido importantes conclusiones sobre la importancia de ciertas propiedades sobre otras y sus influencias entre sí, tal como se muestra en las conclusiones obtenidas en el caso de estudio (ver sección 4.2). Estas conclusiones pueden ser de utilidad, tanto en los procesos de evaluación de productos de seguridad para compradores y evaluadores, como en los procesos de desarrollo de estos productos para los fabricantes de los mismos.

5.2. Trabajo futuro

Las líneas de investigación abiertas con este trabajo de tesis son fundamentalmente las relacionadas con las dos aportaciones principales realizadas:

1. El proceso de generación de modelos de calidad orientados a dominios de aplicación. En relación al proceso DuMoD, entre las principales líneas de investigación podrían citarse las siguientes:
    a. Validación más exhaustiva del proceso. Para ello, tal como se comentó anteriormente, es necesario aplicar dicho proceso a un gran número de proyectos de diferentes tipos y objetivos en el entorno industrial que permitan obtener conclusiones sobre la validez completa del proceso y, también, sobre las posibles mejoras del mismo.
    b. Aplicación del proceso a otros dominios de aplicación para poder recoger datos sobre atributos de diferentes dominios y poder comparlos entre sí.
    c. Definición de una taxonomía de productos COTS que apoye la actividad de definición del dominio de aplicación (fase 1.1) para el que se obtendrá el modelo de calidad. El disponer de una taxonomía de este tipo no sólo facilitaría la actividad de definición del modelo de calidad sino que, además, permitiría obtener una visión clara y estructurada de los diferentes dominios, así como, deducir factores válidos para diferentes productos según la herencia

de características por pertenencia a un mismo grupo (por ejemplo, todo el software de servidor podría tener en común algún factor relacionado con la importancia de la administración del producto software).

d. Desarrollo de herramientas software que faciliten el proceso DuMoD. Una de las principales restricciones del proceso DuMoD es el tiempo requerido para obtener un modelo de calidad orientado a un dominio de aplicación particular. Por ello, el proceso sólo es aplicable en la práctica en aquellos casos en los que vaya a ser reutilizado de forma reiterada. Sin embargo, el conocimiento del dominio de aplicación que proporciona es muy amplio, por lo que puede ser interesante aplicarlo en otros casos, como el desarrollo de productos software. Por todo ello, el desarrollo de herramientas que optimicen el proceso y faciliten su aplicación supone una interesante línea de investigación.

2. El modelo de calidad para productos de seguridad informática. En relación al modelo de calidad obtenido, las principales líneas de investigación identificadas durante su desarrollo son las siguientes:

   a. Extensión del estudio realizado a entornos del extranjero. Aunque el estudio recoge la opinión de muchos profesionales en activo de multinacionales, éstos estaban trabajando en su mayoría, en España. Por lo tanto, uno de los trabajos futuros a realizar es que el estudio se lleve a cabo por expertos trabajando en países diferentes de España. Los resultados obtenidos se podrían integrar con los ya existentes para volver a calcular los modelos de calidad y comparar los resultados obtenidos, o bien, tratar por separado para obtener conclusiones sobre las diferencias y similitudse de los modelos.

   b. Aplicación del modelo de calidad en futuras evaluaciones y comparación de tiempos de evaluación con las ya realizadas con el fin de cuantificar el tiempo de mejora.

   c. Adaptación del proceso de evaluación estándar definido en ISO/IEC 14598-5 [ISO 1998b] con el fin de mejorarlo en base en las lecciones aprendidas de los casos de estudio analizados en la sección 3.2.1.2. De las conclusiones obtenidas en las evaluaciones realizadas, decidimos comenzar con la obtención del modelo de calidad, dada la importancia del mismo en el

resultado final obtenido en el proceso de evaluación y dado que el mayor tiempo consumido en las evaluaciones presentadas era el dedicado tanto a la adaptación del modelo al producto específico como en la medición de los factores. Esto era así principalmente debido al gran número de factores. Una vez obtenido el modelo, el siguiente paso en la mejora de los procesos de evaluación llevados a cabo sería la mejora del proceso de evaluación con el fin de solucionar el resto de problemas encontrados en las evaluaciones realizadas.

d. Por último, dado que en la defición del modelo de calidad no se han tenido en cuenta los criterios relacionados con la sub-característica "seguridad" por existir Perfiles de Protección que definen dichas características, sería interesante definir un proceso que permita pasar de forma sistemática desde los criterios definidos en los Criterios Comunes a los atributos del modelo de calidad para los casos en los que el producto no está certificado. Esto ocurre sobre todo en productos no relacionados con la seguridad informática, donde la concienciación de la seguridad de los expertos que intervienen en los procesos de adquisición de los productos no es tan grande. Por ello, esta línea de investigación podría aplicarse también en relación al proceso DuMoD, es decir, podría extenderse a cualquier dominio de aplicación.

Capítulo 6. Bibliografía y referencias

Abrahao S. y Insfran E. Early Usability Evaluation in Model Driven Architecture Environments. Proceedings of the Sixth International Conference on Quality Software; IEEE Computer Society; 2006. p. 287-94.

Adrion W.R., Branstad M.A. y Cherniavsky J.C. Validation, Verification, and Testing of Computer Software. ACM Computing Surveys (CSUR). 1982;14(2):159-92.

AEDI, (2006). Marca de garantía CAYSER. Accedido en: Mayo 2009; disponible en: http://www.aedi.es/cayser/CAYSER.asp

AENOR, (2005). UNE-EN ISO 9000 Sistemas de gestión de la calidad. Fundamentos y vocabulario. AENOR.

Alexandre Alvaro, Almeida E.S.d. y Meira S.L. A Software Component Quality Model: A Preliminary Evaluation. Proceedings of the 32nd EUROMICRO Conference on Software Engineering and Advanced Applications; IEEE Computer Society; 2006. p. 28-37.

Alvaro A., Santana de Almeida E. y Lemos Meira S. A Software Component Quality Model: A Preliminary Evaluation. Proceedings of the 32nd EUROMICRO Conference on Software Engineering and Advanced Applications; IEEE Computer Society; 2006. p. 28-37.

Alves C. y Finkelstein A. Challenges in COTS decision-making: a goal-driven requirements engineering perspective. Proceedings of the 14th international conference on Software engineering and knowledge engineering; ACM; 2002. p. 789-94.

Andreou A.S. y Tziakouris M. A quality framework for developing and evaluating original software components. Information and Software Technology. 2007;49(2):122-41.

Antonia S. y Michalis X. E-commerce system quality assessment using a model based on ISO 9126 and Belief Networks. Software Quality Control. 2008;16(1):107-29.

Avizienis A., Laprie J.-C., Randell B. y Landwehr C. Basic Concepts and Taxonomy of Dependable and Secure Computing. IEEE Transactions on Dependable and Secure Computing. 2004;01 (1):11-33.

Basili V.R. y Boehm B.W. COTS-Based Systems Top 10 List. IEEE Computer. 2001;34(5):91-3.

Basili V.R. y Weiss D.M. A Methodology for Collecting Valid Software Engineering Data. IEEE Trans Software Eng. 1984;10(6):728-38.

Behkamal B., Kahani M. y Akbari M.K. Customizing ISO 9126 quality model for evaluation of B2B applications. Elsevier Information and Software Technology Journal. 2009;51(3):599-609.

Bentler P.M. Comparative fit indexes in structural models. Psychological Bulletin. 1990;107:238-46.

Bentler P.M. y Chou C.P. Practical issues in structural modeling. Sociological Methods and Research 1987;16(1):78-117.

Bertoa M. y Vallecillo A. Atributos de calidad para componentes COTS. Proceedings of IDEAS'02; 2002a. p. 352-63.

Bertoa M.F. y Vallecillo A. Quality Attributes for COTS Components. 6th ECOOP Workshop on Quantitative Approaches in Object-Oriented Software Engineering (QAOOSE 2002); 2002b. p. 128-44.

Biolchini J. y Gomes P. Systematic Review in Software Engineering. Río de Janeiro, Brazil: Systems Engineering and Computer Science Department, UFRJ; 2005. Report No.: PESC - COPPE/UFRJ.

Boehm B.W., Brown J.R., Kaspar H., Lipow M., McLeod G. y Merritt M. Characteristics of Software Quality. American Elsevier, 1978.

Bollen K.A. Structural Equation with Latent Variables . Wiley, 1989.

Botella P., Burgués X., Carvallo J.P., Franch X., Pastor J.A. y Quer C. Towards a quality model for the selection of ERP systems. En: Heidelberg S.B., ed. *Component-Based Software Quality*. LNCS, 2693 2003:225--45

Botella P., Burgués X., Carvallo J.P., Franch X. y Quer C. Using Quality Models for Assessing COTS Selection. Workshop em Engenharia de Requisitos (WER 02); 2002; 2002. p. 263-77.

Botella P., Burgués X., Carvallo J.P., Franch X. y Quer C. ISO/IEC 9126 in practice: what do we need to know?. Software Measurement European Forum 2004; 2004; 2004. p. 297-306.

Buglione L. y Abran A. A quality factor for software. Proceedings of QUALITA99, Third International Conference on Quality and Reliability; 1999. p. 335-44.

Burgués X. y Franch X. Formalising Software Quality Using a Hierarchy of Quality Models. En: Springer Berlin / Heidelberg, ed. *Lecture Notes in Computer Science, Database and Expert Systems Applications* 2004:741-50.

Burgués X., Franch X. y Pastor J.A. Formalising ERP Selection Criteria. Proceedings of the 10th International Workshop on Software Specification and Design; IEEE Computer Society; 2000. p. 115 - 22.

Byrne B.M. Structural Equation Modelling with EQS and EQS/Windows. Basic Concepts, Applications, and Programming. Sage Publications, 1994.

Carvallo J.P. y Franch X. Extending the ISO/IEC 9126-1 quality model with non-technical factors for COTS components selection. Proceedings of the 2006 international workshop on Software quality; ACM; 2006. p. 9-14.

Carvallo J.P., Franch X. y Carme Q. Towards a Unified Catalogue of Non-Technical Quality Attributes to Support COTS-Based Systems Lifecycle Activities. Proceedings of the Sixth International IEEE Conference on Commercial-off-the-Shelf (COTS)-Based Software Systems; IEEE Computer Society; 2007. p. 21-32.

Carvallo J.P., Franch X., Grau G. y Quer C. On the Use of Quality Models for COTS Evaluation. MPEC Workshop In Proceedings of the 25rd International Conference on Software Engineering (ICSE'2004); IEEE Computer Society; 2004a. p. 31-6.

Carvallo J.P., Franch X. y Quer C. Defining a Quality Model for Mail Servers. Proceedings of the Second International Conference on COTS-Based Software Systems; 2003. p. 51-61.

Carvallo J.P., Franch X. y Quer C. Managing Non-Technical Requirements in COTS Components Selection. Proceedings of the 14th IEEE International Requirements Engineering Conference (RE'06); 2006. p. 316-21.

Carvallo J.P., Franch X., Quer C. y Rodríguez N. A Framework for Selecting Workflow Tools in the Context of Composite Information Systems. Proceedings of the 15th International Workshop on Database and Expert Systems Applications (DEXA'04); Springer-Verlag; 2004b. p. 109-19.

Carvallo Vega J.P., Franch X. y Carme Q. Towards a Unified Catalogue of Non-Technical Quality Attributes to Support COTS-Based Systems Lifecycle Activities. Proceedings of the Sixth International IEEE Conference on Commercial-off-the-Shelf (COTS)-Based Software Systems; IEEE Computer Society; 2007. p. 21-32.

Colombo R. y Cervigni A. The evaluation method for software product. Proceedings of the 15th International Conference Software & Systems Engineering and their Applications (ICSSEA '2002); 2002.

Comella-Dorda S., Dean J., Lewis G., Morris E., Oberndorf P. y Harper E. A Process for COTS Software Product Evaluation. Pittsburgh: Carnegie Mellon, Software Engineering Institute; 2004. Report No.: CMU/SEI-2003-TR-017 ESC-TR-2003-017.

Comella-Dorda S., Dean J.C., Morris E. y Oberndorf P. A Process for COTS Software Product Evaluation. Proceedings of the First International Conference on COTS-Based Software Systems (ICCBSS '02), LNCS 2255; Springer-Verlag; 2002. p. 86-96.

Common Criteria, (2004). Common Methodology for Information Technology Security Evaluation. Version 2.2, revision 311 ed.

Chu A.T.W. y Kalaba R.E. A Comparison of Two Methods for Determining the Weights Belonging to Fuzzy Sets. Journal of Optimization Theory and Applications. 1979;27(4):531-8.

Dae-Woo K., Hyun-Min L. y Sang-Kon L. A Case Study on Testing and Evaluation in the KT-OSS Development. IEEE Tenth International Symposium on Consumer Electronics (ISCE '06); 2006. p. 1-6.

Dean C.J. Timing the Testing of COTS Software Products. Proceedings of the 1st Workshop on Testing Component Based Systems; 17 May 1999; 1999. p. 5-8.

Dillon W.R., Kumar A. y Mulani N. Offending estimates in covariance structure analysis: Comments on the causes and solutions to Heywood cases. Psychological Bulletin. 1987;101:126-35.

Dromey R.G. A Model for Software Product Quality. IEEE Trans Softw Eng. 1995;21(2):146-62.

Dromey R.G. Cornering the Chimera. IEEE Softw. 1996;13(1):33-43.

Fenton N. y Pfleeger S.L. Software metrics: a rigorous and practical approach. 2nd ed. PWS Publishing Co., 1997.

Fernández-Sanz L., Lara P., Escribano J.J. y Villalba M.T. Use cases for enhancing IS requirements management. International Association for Development of the Information Society e-Society 2004; 2004. p. 541-8.

Fernández-Sanz L., Lara P., Villalba M.T. y Vos T. Factores que afectan negativamente a la aplicación práctica de las pruebas de software. III Taller sobre Pruebas en Ingeniería del Software (PRIS 2008); 2008. p. 19-26.

Fernández-Sanz L., Villalba M.T. y Hilera J.R. Factors with negative influence on software testing practice in Spain: a survey. European Systems & Software Process Improvement and Innovation (EUROSPI´09); Springer-Verlag; 2009.

Flynn P., Curran K. y Lunney T. A decision support system for telecommunications. Int J Netw Manag. 2002;12(2):69-80.

Forman E.H. Facts and Fictions about the Analytic Hierarchy Process. Mathematical and Computer Modelling 1993;17(4-5):19-26.

Franch X. y Carvallo J.P. Using quality models in software package selection. IEEE Software. 2003;20(1):34-41.

Franch X., Quer C., A. Canton J. y Salietti R. Experience Report on the Construction of Quality Models for Some Content Management Software Domains. Proceedings of the Seventh International Conference on Composition-Based Software Systems (ICCBSS 2008); IEEE Computer Society; 2008. p. 63-71

García M I.J., Alvira F. . El análisis de la realidad social. Métodos y técnicas de investigación. Alianza Universidad Textos, 1993.

Gediga G., Hamborg K.-C. y Düntsch I. The IsoMetrics Usability Inventory. An operationalisation of ISO 9241-10 supporting summative and formative evaluation of software systems. Behaviour and Information Technology. 1999;18(3):151-64.

Geer D.E. When Is a Product a Security Product? IEEE Security and Privacy. 2005;3(5):80.

Gerbin D.W. y Hamilton J.G. Viability of Exploratory Factor Analysis as a Precursor to Confirmatory Factor Analysis. Structural Equation Modeling. 1996; 3:2-72.

Gi oug O., Doo yeon K., Sang il K. y Sung yul R. A Quality Evaluation Technique of RFID Middleware in Ubiquitous Computing. Proceedings of the 2006 International Conference on Hybrid Information Technology; IEEE Computer Society; 2006. p. 730-5.

Gilb T. Principles of Software Engineering Management. AdissonWesley, 1988.

Grance T., Stevens M. y Myers M. Guide to Selecting Information Technology Security Products. Recommendations of the National Institute of Standards and Technology. Gaithersburg.: U.S. Department of Commerce; 2003. Report No.: NIST SP 800-36.

Grau G., Carvallo J.P., Franch X. y Quer C. DesCOTS: A Software System for Selecting COTS Components. Proceedings of the 30th EUROMICRO Conference; IEEE Computer Society; 2004. p. 118-26.

Hair J.F., Tatham R.L., Anderson R.E. y Black W. Multivariate Data Analysis. fifth ed. Prentice Hall, 1998.

Helmut N., Benjamin Z. y Jens G. An approach to quality engineering of TTCN-3 test specifications. Int J Softw Tools Technol Transf. 2008;10(4):309-26.

Hissam S.A., Carney D. y Plakosh D. DoD Security Needs and COTS-Based Systems. SEI Monographs on the Use of Commercial Software in Government Systems. Final Report. Pittsburgh, PA: Carnegie Mellon Software Engineering Institute; 1998.

Hu L. y Bentler P.M. Evaluating model fit. En: Hoyle R.H., ed. *Structural equation modeling: Concepts, issues and applications*. Sage 1995:76-99.

Hu L. y Bentler P.M. Cutoff Criteria for Fit Indexes in Covariance Structure Analysis: Conventional Criteria versus New Alternatives. Structural Equation Modeling. 1999;6(1):1-55.

IEEE, (1998). IEEE Std. 1061-1998. Standard for a software quality metrics methodology. IEEE Computer Society.

IEEE, (2003). IEEE Std. 610.12-1990: IEEE Standard Glossary of Software Engineering Terminology. IEEE Computer Society.

ISO, (1982). ISO 7498-2 Information processing systems - Open Systems Interconnection - Basic Reference Model -- Part 2: Security Architecture International Standards Organization.

ISO, (1993). ISO/IEC 2382 - Information technology - Vocabulary. ed: International Standards Organization.

ISO, (1994). ISO/IEC 12119 Information technology - Software packages - Quality requirements and testing. International Standards Organization.

ISO, (1995). ISO/IEC 25000:2005 - Software Engineering -- Software product Quality Requirements and Evaluation (SQuaRE) -- Guide to SQuaRE. First edition ed. Ginebra: International Standards Organization.

ISO, (1996). ISO/IEC 14598-1. Information Technology — Software Product Evaluation— Part 1: General Overview. Ginebra: International Standards Organization.

ISO, (1998a). ISO 9241-11 Ergonomic requirements for office work with visual display terminals (VDTs) - Part 11: Guidance on usability. International Standards Organization.

ISO, (1998b). ISO/IEC 14598-5 Information technology - Software product evaluation - Part 5: Process for evaluators. International Standards Organization.

ISO, (1999a). ISO/IEC 14598-1 Information Technology - Software Product Evaluation - Part 1: General Overview. International Standards Organization.

ISO, (1999b). ISO/IEC 14598-4 Software engineering - Product evaluation - Part 4: Process for acquirers. International Standards Organization.

ISO, (2000a). ISO/IEC 14598-2 Software engineering - Product evaluation - Part 2: Planning and management International Standards Organization.

ISO, (2000b). ISO/IEC 14598-3 Software engineering - Product evaluation - Part 3: Process for developers. International Standards Organization.

ISO, (2001a). ISO/IEC 9126-1 Software engineering - Product quality - Part 1: Quality model. International Standards Organization.

ISO, (2001b). ISO/IEC 14598-6 Software engineering - Product evaluation - Part 6: Documentation of evaluation modules. International Standards Organization.

ISO, (2001c). ISO/IEC IS 9126 Information technology -- Software product evaluation -- Quality characteristics and guidelines for their use. International Standards Organization.

ISO, (2002). ISO/IEC 15939 Software engineering - Software measurement process.: International Standards Organization.

ISO, (2003a). ISO/IEC 9126-2 Software engineering - Product quality - Part 2: External metrics International Standards Organization.

ISO, (2003b). ISO/IEC 9126-3 Software engineering - Product quality - Part 3: Internal metrics. International Standards Organization.

ISO, (2004a). ISO/IEC 9126-4 Software engineering - Product quality - Part 4: Quality in use metrics International Standards Organization.

ISO, (2004b). ISO/IEC 18045 Information technology - Security techniques - Methodology for Information Technology Security Evaluation. International Standards Organization.

ISO, (2005a). ISO 9000 Quality management systems - Fundamentals and vocabulary. International Standards Organization.

ISO, (2005b). ISO/IEC 15408-1. Information technology - Security techniques - Evaluation criteria for IT security - Part 1: Introduction and general model. International Standards Organization.

ISO, (2005c). ISO/IEC 15408-2. Information technology - Security techniques - Evaluation criteria for IT security - Part 2: Security functional requirements. International Standards Organization.

ISO, (2005d). ISO/IEC 15408-3. Information technology - Security techniques - Evaluation criteria for IT security -- Part 3: Security assurance requirements. International Standards Organization.

ISO, (2005e). ISO/IEC 15408. Information technology - Security techniques - Evaluation criteria for IT security. International Standards Organization.

ISO, (2005f). ISO/IEC 25000 - Software Engineering - Software product Quality Requirements and Evaluation (SQuaRE) - Guide to SQuaRE. ed: International Standards Organization.

ISO, (2005g). ISO/IEC 25051 Software engineering - Software product Quality Requirements and Evaluation (SQuaRE) - Requirements for quality of Commercial Off-The-Shelf (COTS) software product and instructions for testing. ed: International Standards Organization.

ISO, (2006). ISO/IEC 9241-110. Ergonomics of human-system interaction - Part 110: Dialogue principles. International Standards Organization.

J. Hall M.J., Hall R. y Zeleznikow J. A process for evaluating legal knowledge-based systems based upon the context criteria contingency-guidelines framework. Proceedings of the 9th international conference on Artificial intelligence and law; ACM; 2003. p. 274-83.

J. Nunnally. Psychometric Theory. McGraw-Hill, 1978.

Jaccheri M.L. y Torchiano M. Classifying COTS Products. European Conference on Software Quality, LNCS 2349; Springer-Verlag; 2004. p. 246-55.

Jin L., Yin G. y Yang D. Fuzzy Integrated Evaluation for Measuring Quality of Feature Space-Based Component Model. International Conference on Internet Computing in Science and Engineering; 2008. p. 349-54.

Joreskog K.G. y Sorbom D. LISREL VII: Analysis of Linear Structure Relationships by the Methods of Maximum Likelihood. Scientific Software, 1988.

Joreskog K.G. y Sorbom D. LISREL 8: Structural Equation Modeling with SIMPLIS Command Language. Scientific Software, 1993.

Kaiser H.F. An index of factorial simplicity. Psychometrika. 1974;39(1):31-6.

Kilsup Lee y Lee S.J. A Quantitative Evaluation Model Using the ISO/IEC 9126 Quality Model in the Component Based Development Process. Proceedings of the 2006 International Conference on Computational Science and its Applications (ICCSA'06); Springer-Verlag; 2006. p. 917-26.

Kirakowski J. y Corbett M. SUMI: The Software Usability Measurement Inventory. . British Journal of Educational Technology. 1993;24:210-2.

Kitchenham B. Procedures for Performing Systematic Reviews; 2004. Report No.: TR/SE0401, Keele University, and 0400011T.1, National ICT Australia.

Kitchenham B. y Pfleeger S.L. Software Quality: The Elusive Target. IEEE Softw. 1996;13(1):12-21.

Kitchenham B.A. Guidelines for performing Systematic Literature Reviews in Software Engineering Version 2.3: Keele University and University of Durham,; 2007. Report No.: EBSE-2007-01.

Kitchenham B.A. y Walker J.G. A Quantitative Approach to Monitoring Software Development. Software Engineering Journal. 1989;4(1):2-13.

Kontio J. OTSO: a systematic process for reusable software component selection: University of Maryland, Technical report; 1995. Report No.: CS-TR-3478.

Kontio J. A case study in applying a systematic method for COTS selection. Software Engineering, 1996, Proceedings of the 18th International Conference on Software engineering March 25 - 29, 1996; IEEE Computer Society.; 1996. p. 201-9.

Kontio J., Caldiera G. y Basili V.R. Defining factors, goals and criteria for reusable component evaluation. Proceedings of the 1996 conference of the Centre for Advanced Studies on Collaborative research; IBM Press; 1996. p. 21- 32.

Koyanl S.J., Balley R.W., Nall J.R., Allison S., Mulligan C., Bailey K. y Tolson M. Research-Based Web Design & Usability Guidelines. U.S. Dept. of Health and Human Services. Disponible en: http://www.usability.gov/pdfs/guidelines.html, 2006.

Kunda D. STACE: Social Technical Approach to COTS Software Evaluation. En: Cechich A., Piattini M. y Vallecillo A., eds. *Component-Based Software Quality, LNCS 2693*. Springer-Verlag 2003:64-84.

Kunda D. y Brooks L. Applying social-technical approach for COTS selection. Proceedings of the 4th UKAIS Conference; McGraw Hill; 1999. p. 552-65.

Kunda D. y Brooks L. Identifying and classifying processes (traditional and soft factors)that suuport COTS component selection: a case study. European Journal of Information Systems. 2000;9(4):226-34.

Landeta J. El método Delphi: Una técnica de previsión para la incertidumbre. Ariel Practicum, 1999.

Lawlis P.K., Mark K.E., Thomas D.A. y Courtheyn T. A Formal Process for Evaluating COTS Software Products. IEEE Computer. 2001;34(5):58-63.

Lingyu W., Anoop S. y Sushil J. Toward measuring network security using attack graphs. Proceedings of the 2007 ACM workshop on Quality of protection; ACM; 2007. p. 49-54.

Losavio F., Chirinos L., Matteo A., Lévy N. y Ramdane-Cherif A. ISO quality standards for measuring architectures. Journal of Systems and Software. 2004;72(2):209-23.

Losavio F., Matteo A. y Rahamut R. Web Services Domain Analysis based on Quality Standards. Proceedings of the 2nd European conference on Software Architecture (ECSA '08); Springer-Verlag; 2008. p. 354-8

Maiden N.A. y Ncube C. Acquiring COTS software selection requirements. IEEE Software. 1998;15(2):46-56.

Maiden N.A., Ncube C. y Moore A. Lessons learned during requirements acquisition for COTS systems. Communications of ACM. 1997;40(12):21-5.

Maiden N.A.M., Kim H. y Ncube C. Rethinking Process Guidance for Selecting Software Components. First International Conference COTS-Based Software Systems, ICCBSS 2002 Springer-Verlag; 2002. p. 151-64.

Malak G., Badri L., Badri M. y Sahraoui H. Towards a Multidimensional Model for Web-Based Applications Quality Assessment. En: Bauknecht K., Bichler M. y Pröll B., eds. *E-Commerce and Web Technologies, LNCS 3182*. Springer-Verlag 2004:316-27.

Marco T., Letizia J., Carl-Fredrik S., rensen y Alf Inge W. COTS products characterization. Proceedings of the 14th international conference on Software engineering and knowledge engineering; ACM; 2002. p. 335 - 8.

Martinez M., Azevedo G., Lopes S., Pagliuso P., Colombo R., Rodrigues M. y Jino M. The Software Product Evaluation Data Base - Supporting MEDE-PROS. Proceedings of the 4th IEEE International Symposium and Forum on Software Engineering Standards; IEEE Computer Society 1999. p. 182-91.

McCall J., Richards P. y Walters G. Factors in software quality. Volume I. Concepts and Definitions of Software Quality. Roma (Italia): US Rome Air Development Center; 1977. Report No.: CDRL A003.

Microsoft, (2009). ISA Server 2006. Accedido en: Mayo 2009; disponible en: http://www.microsoft.com/spain/isaserver/default.mspx

Moraga M.Á., Calero C., Garzás J. y Piattini M. Assessment of portlet quality: Collecting real experience. Computer Standards & Interfaces. 2009;31(2):336-47.

Morisio M., Stamelos I. y Tsoukias A. A new method to evaluate software artifacts against predefined profiles. Proceedings of the 14th international conference on Software engineering and knowledge engineering (SEKE' 02); 2002. p. 811 - 8

Morisio M. y Torchiano M. Definition and Classification of COTS: A Proposal. ICCBSS '02: Proceedings of the First International Conference on COTS-Based Software Systems; Springer-Verlag; 2002. p. 165-75.

NIST, (2001). Security Requirements For Cryptographic Modules (FIPS PUB 140-2). In: National Institute of Standards and Technology (NIST). Computer Security Division, ed.

NIST, (2009). National vulnerability database. Accedido en: Mayo 2009; disponible en: http://nvd.nist.gov/.

Oberndorf P.A. Facilitating Component-Based Software Engineering: COTS and Open Systems. Proceedings of the 5th International Symposium on Assessment of Software Tools (SAST '97); IEEE Computer Society; 1997. p. 143-8.

Oberndorf P.A., Brownsword L., Morris E. y Sledge C. Workshop on COTS-Based Systems: Software Engineering Institute, Carnegie Mellon University; 1997 Noviembre 1997. Report No.: CMU/SEI-97-SR-019.

Obeso M.E.A. Proposal of a Tool of Support to the Evaluation of User in Educative Web Sites. Proceedings of the 1st world summit on The Knowledge Society: Emerging Technologies and Information Systems for the Knowledge Society; LNCS 5288; 2008. p. 149-57

Ochs M., Pfahl D., Chrobok-Diening G. y Nothhelfer-Kolb B. A Method for Efficient Measurement-based COTS Assessment and Selection -Method Description and

Evaluation Results. Proceedings of the 7th IEEE International Software Metrics Symposium (METRICS 2001); IEEE Computer Society; 2001. p. 285-96.

Olsina L. y Rossi G. Measuring Web Application Quality with WebQEM. IEEE MultiMedia. 2002a;9(4):20-9.

Olsina L. y Rossi G.A. Quantitative Method for Quality Evaluation of Web Sites and Applications. IEEE Multimedia 2002b;9(4):20-9

Parasuraman A., Zeithaml V.A. y Berry L.L. SERVQUAL: A Multiple-Item Scale for Measuring Consumer for Perceptions of Service Quality. Journal of Retailing. 64:12-40.

Parasuraman A., Zeithaml V.A. y Berry L.L. A conceptual model of service quality and its implications for future research. Journal of Marketing. 1985;70(3):201-30.

Pérez L.S.V. y Tornés A.F.G. MECHDAV- A quality model for the technical evaluation of applications development tools in visual enviroments. Software measurement European forum, SMEF 2005; 2005. p. 155-63.

Perez L.S.V., Tornes A.F.G. y Riveron E.M.F. MECRAD: Model and Tool for the Technical Quality Evaluation of Software Products in Visual Environment. Computing in the Global Information Technology, 2008 ICCGI '08 The Third International Multi-Conference on; 2008. p. 107-12.

Peter J.P. Construct validity: A review of basic issues and marketing practices. Journal of Marketing Research. 1981;18:133-45.

Powell A., Vickers A., Wing L., Williams E. y Cooke B. Evaluating Tools to Support Component Based Software Engineering. Proceedings of the 5th International Symposium on Assessment of Software Tools (SAST '97); IEEE Computer Society; 1997. p. 80-9.

Punter T., Kusters R., Trienekens J., Bemelmans T. y Brombacher A. The W-Process for Software Product Evaluation: A Method for Goal-Oriented Implementation of the ISO 14598 Standard. Software Quality Control. 2004;12(2):137-58.

Punter T., Solingen R.V. y Trienekens J. Software Product Evaluation - Current status and future needs for customers and industry. 4th IT Evaluation Conference (EVIT-97); Kluwer Academic Publishers; 1997. p. 137-58.

Realsec, (2007). CRYPTOSEC LAB TEST Accedido en: 2008; disponible en: http://www.realsec.com/fotos/noticia_EN_20071113_104152_32.pdf

Realsec, (2008). Accedido en: Mayo 2009; disponible en: http://www.realsec.com/en/index.php

Robert P. SCOPE: Final report Scope Consortium; 1994.

Rodriguez D., Harrison R. y Satpathy M. A Generic Model and Tool Support for Assessing and Improving Web Processes. Proceedings of the Eighth IEEE Symposium on Software Metrics (METRICS'02); IEEE Computer Society; 2002. p. 141-51.

Saaty T.L. The Analytic Hierarchy Process. McGraw-Hill, 1990.

Saaty T.L. Analytic Hierarchy. McGraw-Hill, 1992.

Sangeeta N. y Hausi A.M. Quality Criteria and an Analysis Framework for Self-Healing Systems. Proceedings of the 29th International Conference on Software Engineering Workshops; IEEE Computer Society; 2007. p. 6-16.

Schermelleh-Engel K. y Moosbrugger H.y.M., H. . Evaluating the Fit of Structural Equation Models: Tests of Significance and Descriptive Goodness of Fit Measures. Methods of Psychological Research Online. 2003;8(2):23-74.

Shoemaker P.J. y Waid C.C. An Experimental Comparison of Different Approaches to Determining Weights in Additive Utility Models. Management Science. 1982;28(2):182-96.

Snow B. We Need Assurance! Proceedings of the 21st Annual Computer Security Applications Conference (ACSAC'05); IEEE Computer Society; 2005. p. 3-10.

Spriestersbach A. y Springer T. Quality Attributes in Mobile Web Application Development 4th International Conference Product-Focused Software Process Improvement (PROFES'02); Springer-Verlag; 2002. p. 120-30.

SPSS Inc. SPSS advanced statistics guide. 4th ed. Chicago Press, 1990.

Stefan W. y Florian D. An Integrated Approach to Quality Modelling. Proceedings of the 5th International Workshop on Software Quality; IEEE Computer Society; 2007. p. 1-6.

Stefani A. y Xenos M. A model for assessing the quality of e-commerce systems. Proceedings of the PC-HCI 2001 Conference on Human Computer Interaction; 2001. p. 105-9.

Strahonja V. The Evaluation Criteria of Workflow Metamodels. Information Technology Interfaces, 2007 ITI 2007 29th International Conference on; 2007. p. 553-8.

Tabachnick B.G. y Fidell L.S. Using Multivariate Statistics 5th ed. Pearson Education, 2006.

Thomsen E. Olap Solutions: Building Multidimensional Information Systems. Wiley, 2002.

Torchiano M. y Jaccheri M.L. Assessment of Reusable COTS Attributes. Proceedings of the Second International Conference on COTS-Based Software Systems; Springer-Verlag; 2003. p. 219-28.

Trienekens J., Veenendaal E. y Veenendaal J. Software Quality from a business perspective: Directions and Advanced Approaches. Kluwer bedrijfsinformatie, 1997.

Veenendaal E.P.W.M.V. y Trienekens J.J.M. Testing based on users' quality needs. IFIP TC5 WG54 3rd International Conference on reliability, quality and safety of software-intensive systems; Chapman & Hall, Ltd.; 1997. p. 242 - 55.

Villalba de Benito M.T., Fernández Sanz L., Escribano J.J. y Lara P. Un estudio sobre rendimiento web. International Association for Development of the Information Society WWW/Internet 2004; 2004. p. 227-34.

Villalba M.T. y Fernández-Sanz L. Technical report -- Evaluation report of Cryptosec 2048. eSecurity, European Security. 2007a;13:78-81.

Villalba M.T. y Fernández-Sanz L. Technical report -- Evaluation report of Internet Security and Acceleration (ISA) Server 2006. eSecurity, European Security. 2007b November;14:78-81.

Villalba M.T. y Fernández-Sanz L. Technical report -- Evaluation report of Internet Application Gateway (IAG) 2007 eSecurity, European Security. 2008 June;18:53-7.

Villalba M.T., Fernández-Sanz L. y Martínez Herraiz J.J. Un nuevo marco de convergencia y calidad para la gestión de la seguridad en el servicio de TI. Revista de Procesos y Métricas - Asociación Española de Métricas del Software (AEMES). 2008;5(2):35-42.

Voas J. COTS Software: The Economical Choice? IEEE Software. 1998;15(2):16-9.

Won Jun S., Ji Hyeok K. y Sung Yul R. A Quality Model for Open Source Software Selection. Proceedings of the Sixth International Conference on Advanced Language Processing and Web Information Technology (ALPIT 2007) - Volume 00; 2007. p. 515-9.

Ye F. y Kelly T. COTS Product Selection for Safety-Critical Systems. COTS-Based Software Systems; Springer-Verlag; 2004. p. 53-62.

YeongSeok L., JungHyun B. y Seokkoo S. Development of Quality Evaluation Metrics for BPM (Business Process Management) System. Proceedings of the Fourth Annual ACIS International Conference on Computer and Information Science; IEEE Computer Society; 2005. p. 424- 9.

Yoonjung C., Sungwook L., Houp S., Jingoo P. y SunHee K. Practical S/W Component Quality Evaluation Model. 10th International Conference on Advanced Communication Technology, ICACT 2008; 2008. p. 259-64.

Zeleny M. Multiple Criteria Decision Making. McGraw-hill, 1982.

Capítulo 7. Anexos

7.1. Acrónimos

En la Figura 45 se muestra la lista de los acrónimos utilizados a lo largo de este trabajo para una mayor comprensión.

Acrónimo	Significado
AEDI	Asociación Española de Directores de Informática
AHP	Analytic Hierarchy Process
ASIMELEC	Asociación Multisectorial de Empresas Españolas de Electrónica
ATI	Asociación de Técnicos de Informática
CAYSER	CAlidad Y SERvicio
CBD	Component Based Development
CFI	Comparative Fit Index
COQUAMO	COnstructive QUAlity MOdel
COTS	Commercial off-the-shelf
CTCPEC	Canadian Trusted Computer Product Evaluation Criteria
DuMoD	Domain-oriented qUality MOdels Development
EAL	Evaluation Assurance Level
EAL	Evaluation Assurance Levels
GQM	Goal-Question-Metric
HSM	Hardware Security Module
ISO	International Organisation for Standards
ITSEC	Information Technology Security Evaluation Criteria
KMO	Kaiser-Meyer-Olkin
MCDM	Multiple Criteria Decision Making
NFI	Normed Fit Index

NNFI	Non-Normed Fit Index
NTF	Non-Technical Factors
OTS	Off-The-Shelf
PP	Protection Profile
RePRIS	Red para la promoción y mejora de las Pruebas en ingeniería del software
RFI	Relative Fit Index
RMSEA	Root Mean Square Error Of Approximation
SAR	Security Assurance Requirement
SEM	Structural Equation Modeling
SFR	Security Funcional Requirement
SQuaRE	Software product Quality Requirements and Evaluation
SUMI	Software Usability Measurement Inventory
TCSEC	Trusted Computer System Evaluation Criteria,
TI	Tecnologías de la Información
TF	Technical Factors
TLI	Tucker-Lewis index
UF	Usability Factors

Figura 45. Lista de acrónimos utilizados.

7.2. Descripción técnica de la plataforma web de soporte a la recogida de datos

7.2.1. Arquitectura tecnológica

Con el fin de llegar al máximo número de profesionales posible y facilitarles la tarea de completar los cuestionarios, se decidió proporcionar el acceso a los mismos a través de una plataforma web. La arquitectura final de la plataforma se muestra en la Figura 46. Tal como se muestra en la figura, el acceso a los cuestionarios se realizaba a través de un navegador desde el cual se completaban los datos requeridos. Dichos datos se iban almacenando en la base de datos para su posterior análisis.

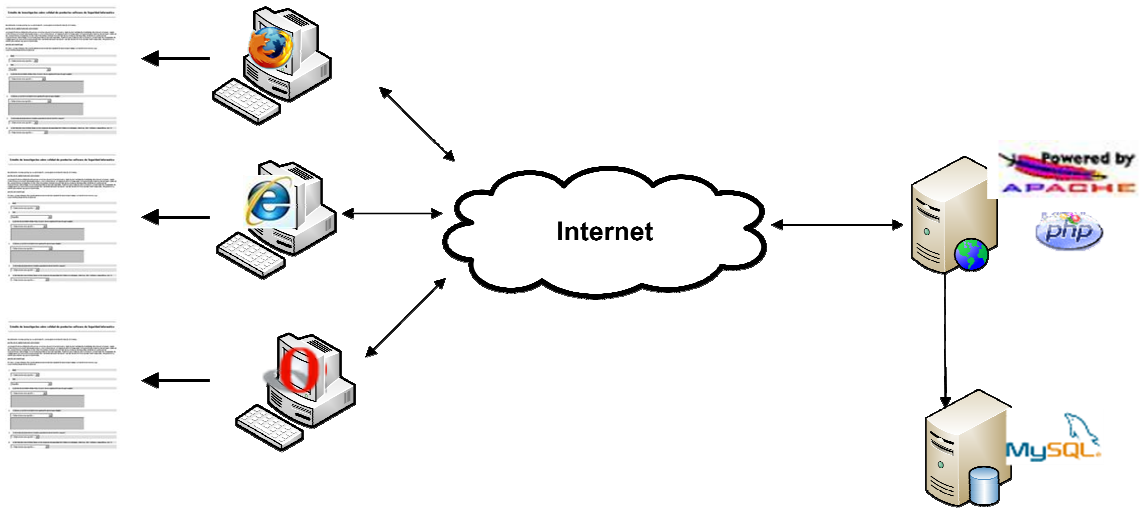


Figura 46. Arquitectura de la plataforma web de recogida de datos.

7.2.2. Estructura de la base de datos

En la Figura 47 se muestra el diagrama relacional de la base de datos utilizada por la aplicación web para la recogida de datos y su posterior extracción para llevar a cabo el tratamiento estadístico de los mismos.

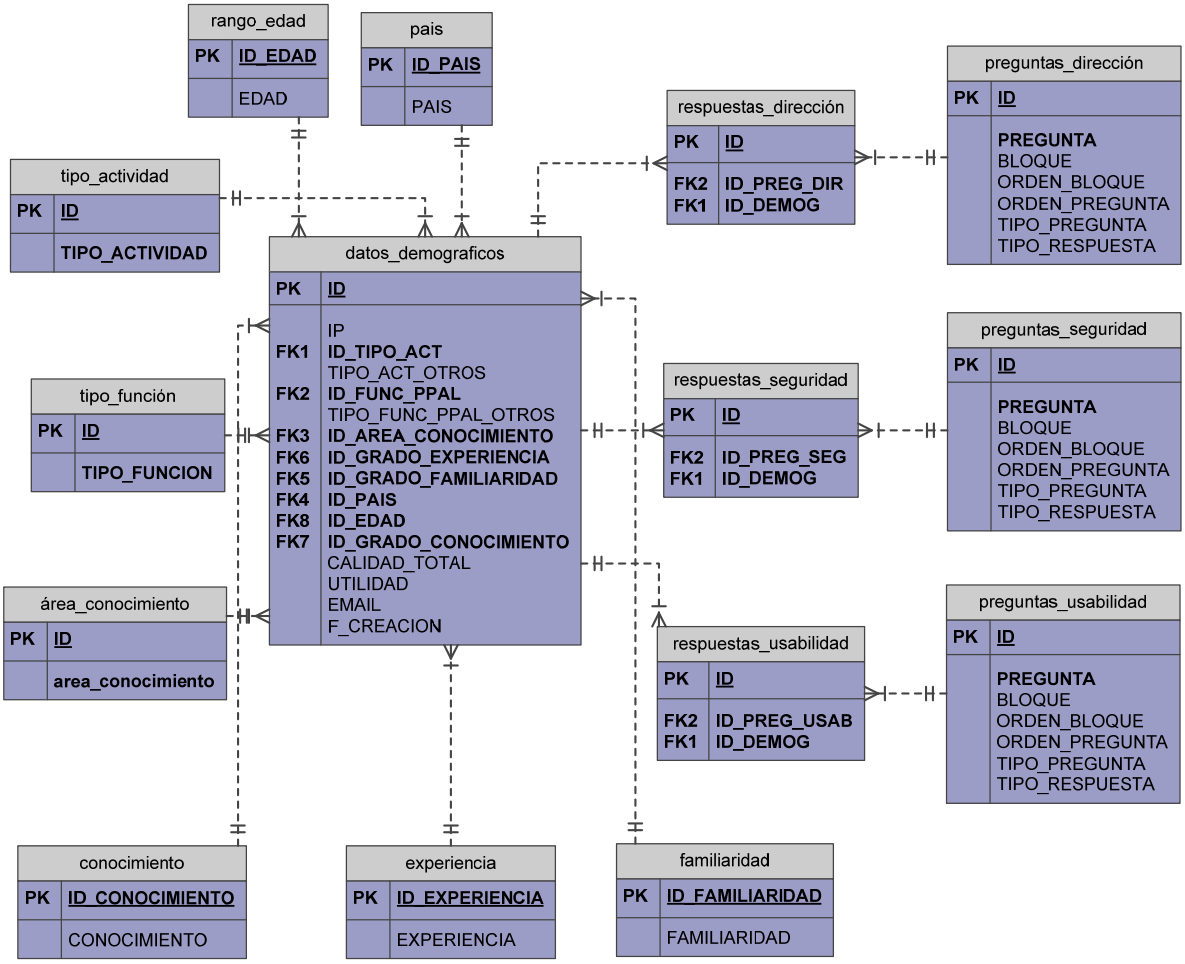


Figura 47. Modelo relacional de la aplicación web de recogida de datos.

7.3.Cuestionarios

El acceso inicial (ver sección 7.3.1) era común a todos los perfiles de usuario. En dicho acceso se mostraba una breve introducción al estudio y se explicaba el objetivo del mismo. Además, se solicitaba a los participantes una serie de datos demográficos que sirviesen más tarde para verificar que el perfil se ajustaba a los requisitos del estudio. Por último, se pedía que el usuario seleccionase el área de conocimiento bajo la que deseaba completar el estudio: seguridad de tecnologías de la información y comunicaciones, ingeniería del software o dirección y gestión. Según el área de conocimiento seleccionada, el usuario era dirigido al cuestionario con los atributos referentes al área en cuestión. Una vez completado el cuestionario y enviadas las respuestas, los usuarios eran dirigidos a una página con dos preguntas finales acerca de la importancia de la calidad del software y de disponer de un modelo de calidad para la selección de productos COTS de seguridad informática (ver sección

7.3.1. Inicio y recogida de datos demográficos

-

Estudio de investigación sobre calidad de productos software de seguridad informática

---

De antemano muchas gracias por su participación. La encuesta no le llevará más de 15 minutos.

¿CUÁL ES EL OBJETIVO DEL ESTUDIO?

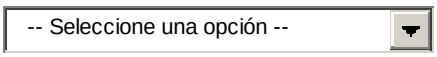
La evaluación de la calidad del software es un proceso de gran importancia pero, dada la gran cantidad de propiedades del software a evaluar, puede convertirse en un proceso demasiado largo y, como consecuencia, no realizarse de forma adecuada. Con este estudio tratamos de averiguar cuales de las características propuestas son las más importantes a evaluar en el ámbito de los productos de seguridad, teniendo en cuenta que las características relacionadas con la propia seguridad ya han sido obtenidas.

Sabemos que lo ideal es que un producto cumpla todas las propiedades de calidad que se le mostrarán en este estudio pero, partiendo del punto de que por razones de tiempo no es posible medir todas ellas, necesitamos su opinión para obtener las que son esenciales.

ANTES DE EMPEZAR,

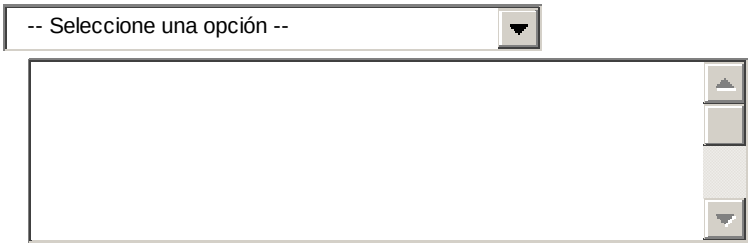
Por favor, proporciónenos información general sobre el tipo de organización para la que trabaja, su función en la misma y sus conocimientos/experiencia profesional.

1.Edad:

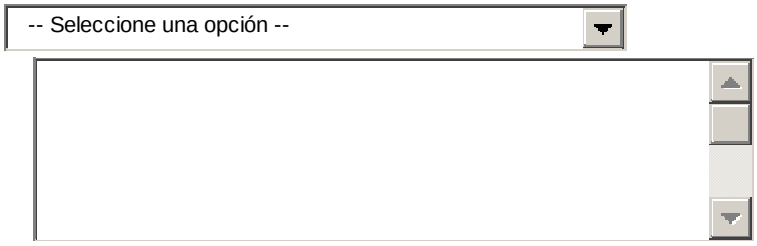


2.País:

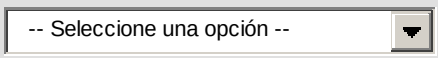
3.¿Qué tipo de actividad refleja mejor el sector de la organización para la que trabaja?:



4.¿Cuál es su función principal en la organización para la que trabaja?:



5.¿Qué grado de experiencia considera que tiene en dicha función o puesto?



6.¿Qué nivel de conocimiento tiene con los productos de seguridad informática (cortafuegos, antivirus,

IDS, módulos criptográficos, etc.)?

-- Seleccione una opción --

7.¿Cuál el su grado de familiaridad con dichos productos?

-- Seleccione una opción --

En función de su experiencia o cualificación, por favor, elija el área de conocimiento bajo la cual prefiere contestar a la encuesta sobre características de calidad de productos de seguridad:

- dirección o gestión (empresas o departamentos relacionados con las Tecnologías de la información)
- Ingeniería del software (programación, calidad)
- Seguridad de Tecnologías de la información y Comunicaciones

Por favor, introduzca una cuenta de correo electrónico a la que desea que le remitamos los resultados del estudio, le **aseguramos** que **no** usaremos dicha dirección electrónica para ninguna otra finalidad:

SOBRE LA INFORMACIÓN PROPORCIONADA…

Durante el proceso no se pedirá ninguna información personal, tan solo la relacionada con sus conocimientos y experiencia profesional. Al finalizar el estudio, si así lo desea, le enviaremos un detallado informe de los resultados obtenidos a la dirección de correo electrónico que nos facilite.

La información aquí proporcionada será estrictamente confidencial, sólo se utilizará para este estudio y no se tratará individualmente.

7.3.2. Cuestionario para el modelo de calidad de factores técnicos

CUESTIONARIO SOBRE CALIDAD DE PRODUCTOS SOFTWARE DE SEGURIDAD INFORMÁTICA

Por favor, complete el siguiente cuestionario indicando por cuales de las siguientes características de productos de seguridad informática estaría dispuesto a asumir un coste o esfuerzo extra en el proceso de compra.

De antemano, muchas gracias por su participación.

FUNCIONALIDAD

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre
Obtener unos resultados exactos tras la ejecución de las funciones software					
Uso de interfaces estándares que le permita interaccionar con otros productos					
Disponer de unas especificaciones hardware adecuadas					
Unas especificaciones y requisitos software adecuados					
Un producto compatible con otros programas del mismo tipo o similar					
Una certificación de seguridad reconocida que haya sido otorgada por un laboratorio independiente					

En general, asumiría un coste extra por tener una mejor funcionalidad en mis productos de seguridad					

FIABILIDAD

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre

Un producto con un bajo porcentaje de fallos (madurez del producto)					
Un tiempo de espera adecuado para la corrección de fallos del software(parches)					
Un producto en el que los fallos causados por errores del software no afecten al sistema operativo o a otros programas					
Un número de fallos críticos del software mínimo o inexistente					
Un producto cuyos fallos leves no afecten a la disponibilidad del resto de funciones					
Un producto cuyos fallos graves no afecten a la disponibilidad de las funciones críticas					

Un producto capaz de recuperarse de forma automática tras un fallo					
Un tiempo adecuado de no disponibilidad del producto tras un fallo					
Un producto capaz de volver a un estado previo después de un evento anormal (*restore*)					
Una adecuada capacidad de volver a un estado normal después de un fallo (*recovery*)					
Un producto software robusto (reacciona adecuadamente ante situaciones anormales)					
Una adecuada documentación sobre las acciones a realizar para la recuperación del sistema					

En general, asumiría un coste extra por tener una mayor fiabilidad en mis productos de seguridad					

USABILIDAD

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre
Una ayuda, retroalimentación e información proporcionada en los diálogos de la aplicación fáciles de entender					
Un producto con un rápido aprendizaje inicial de los expertos para el uso de la aplicación					
Un producto fácil de operar y controlar por el administrador					

Disponer de un interfaz atractivo de la aplicación					
En general, asumiría un coste extra por tener una mayor facilidad de uso en mis productos de seguridad					

EFICIENCIA

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre
Un tiempo de respuesta adecuado de la aplicación					
Una adecuada capacidad de procesamiento (número de tareas ejecutadas por unidad de tiempo)					
Un producto con un bajo consumo de memoria					
Un producto con un bajo consumo de procesador					
Un producto que requiera un adecuado espacio en disco para su ejecución tras la instalación					
Un producto escalable (que permite manejar el incremento de trabajo)					
En general, asumiría un coste extra por tener una mayor eficiencia en mis productos de seguridad					

MANTENIMIENTO

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre
Una disponibilidad de las actualizaciones del producto adecuada					
Actualizaciones fáciles de instalar					
Un bajo esfuerzo requerido por el usuario para actualizar el producto					
Unos parches (actualizaciones del sistema para resolver errores) estables					
Un producto capaz de volver a un estado previo tras la aplicación de un parche (update) o de un cambio de versión					

(*upgrade*)					
Un producto capaz de mantener las configuraciones tras un cambios versión					
En general, asumiría un coste extra por un producto de seguridad con un mejor mantenimiento					

PORTABILIDAD

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre
Un producto fácil de instalar					
Un producto que requiera un bajo esfuerzo de instalación					
Unas adecuadas ayudas o documentación para el proceso de instalación del producto					
La posibilidad de realizar la instalación del producto en mi lengua nativa					
Disponer de retroalimentación durante el proceso de instalación					
Disponer de un asistente de configuración guiada tras la instalación del producto					
Disponer de ayuda al proceso de configuración del producto					
La posibilidad de personalizar la información visualizada en el registro de eventos (logs)					
Disponer de información sobre la criticidad de los eventos y sucesos notificados					
La posibilidad de realizar personalizaciones del registro de eventos					
En general, asumiría un coste extra por tener una mayor portabilidad en mis productos de seguridad					

CARACTERÍSTICAS ORGANIZATIVAS

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre
Un proveedor que proporcione un mejor servicio (de mayor reputación, estabilidad, experiencia, etc.)					

(tipo de licencia, soporte, liderazgo en el mercado, etc.)					
En general, asumiría un coste extra por tener un producto de un proveedor de calidad y con unas características no técnicas mejores en mis productos de seguridad					

Si cree que hay alguna característica de productos de seguridad informática que sería importante incluir que no aparece aquí, por favor, indíquela a continuación (opcional):

7.3.3. Cuestionario para el modelo de calidad de factores no técnicos

CUESTIONARIO SOBRE CALIDAD DE PRODUCTOS SOFTWARE DE SEGURIDAD INFORMÁTICA

Por favor, complete el siguiente cuestionario indicando por cuales de las siguientes características de productos de seguridad informática estaría dispuesto a asumir un coste o esfuerzo extra en el proceso de compra.

De antemano, muchas gracias por su participación.

CARACTERÍSTICAS TÉCNICAS

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre
Una funcionalidad que cumpla con las expectativas de los usuarios					
Una mayor seguridad del producto software					
Una mayor fiabilidad (madurez, estabilidad, robustez, etc) del					

producto software de seguridad					
Una mayor usabilidad (facilidad de uso, ayuda, etc)					
Un mejor rendimiento del producto software					

Una mayor facilidad de mantenimiento					
Un producto software de seguridad con una frecuencia adecuada de las actualizaciones					
Una mayor facilidad de instalación y configuración del software					
Un producto que posea una certificación de seguridad reconocida					
Un producto conforme con los estándares existentes de calidad del software					

En general, asumiría un coste extra por un producto con unas buenas características técnicas					

CARACTERÍSTICAS ORGANIZATIVAS: PROVEEDOR DE SOFTWARE

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre
Un producto cuyo proveedor tenga una mayor participación en el mercado (market share)					
Un proveedor con una reputación mayor					
Un proveedor con mayor estabilidad o solvencia					
Un proveedor con más experiencia					
Un proveedor tenga una mejor accesibilidad para el cliente					

Un proveedor con una autonomía e independencia mayor con respecto a otros fabricantes					
Un producto cuyo proveedor posea algún estándar o certificación sobre calidad del servicio proporcionado					
En general, asumiría un coste extra por un producto de un proveedor de calidad					

CARACTERÍSTICAS ORGANIZATIVAS: PRODUCTO SOFTWARE

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre
Un producto desarrollado con las últimas tecnologías					
Un producto compatible con la arquitectura en la que se integrará					
Compatibilidad con la política corporativa de uso de las TIC de la organización					
Una mayor estabilidad del producto en el mercado					
Un producto líder en el mercado					
Un tipo de licencia que se ajuste a los requisitos de la organización					
Un menor coste total del producto (licencia, adaptación, integración, formación, soporte, etc.)					
Un tipo de soporte adecuado a las necesidades de mi organización					
Una mejor oferta de formación para el uso y administración del producto					
Una forma de pago ajustada a las necesidades de la organización					
Un producto que me han recomendado					
En general, asumiría un coste extra por tener un producto con unas características no técnicas mejores en mis productos de seguridad					

Si cree que hay alguna característica de productos de seguridad informática que sería importante incluir y que no aparece aquí, por favor, indíquela a continuación:

7.3.4. Cuestionario para el modelo de calidad de factores de usabilidad

CUESTIONARIO SOBRE CALIDAD DE PRODUCTOS SOFTWARE DE SEGURIDAD INFORMÁTICA

Por favor, complete el siguiente cuestionario indicando por cuales de las siguientes características de productos de seguridad informática estaría dispuesto a asumir un coste o esfuerzo extra en el proceso de compra.

De antemano, muchas gracias por su participación.

COMPRENSIÓN

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre
Una mejor comprensión de la ayuda proporcionada por el programa					
Un mejor comprensión de la retroalimentación proporcionada por el programa					
Una mejor comprensión de la información proporcionada en los diálogos del programa					
Una localización eficiente de información					
Un uso moderado de elementos de distracción (imágenes, multimedia)					
La agrupación de la información adecuada para mejorar la comprensión					
Un interfaz en el que los mensajes sean formulados de forma					

constructiva, objetiva y comprensible					
Una mejor comprensión global del diálogo mediante la retroalimentación					
Disponer de información adaptada al nivel de conocimiento de los expertos en seguridad					
Disponer de ayuda y/o retroalimentación sobre la naturaleza de los datos a introducir					

Una aplicación que utilice un lenguaje simple y directo					
Disponer de soporte a la lengua nativa en los diálogos de la aplicación					
Disponer de soporte a la lengua nativa en la ayuda					
En general, asumiría un coste extra por un buen soporte a la comprensión del programa					

APRENDIZAJE

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre
Disponer de un interfaz único de acceso a la aplicación para facilitar el aprendizaje					
Una distinción clara de los accesos a las distintas funcionalidades de la aplicación					
Disponer de claridad en el acceso a los mecanismos de apoyo al uso de la aplicación (ayuda, por ejemplo)					
Disponer de información básica sobre aspectos conceptuales del programa en la ayuda					
Disponer de ayuda global del programa					

Disponer de ayuda de cada función particular asociada al diálogo de ejecución correspondiente					
Disponer de ejemplos de aplicación para facilitar el aprendizaje en la ayuda					
Tener tutoriales que faciliten el aprendizaje inicial de la aplicación					
El uso de las funcionalidades más utilizadas para mejorar la experiencia del usuario					
Disponer de atajos para usuarios avanzados y soluciones por defecto para las funciones más usadas					
Disponer de información más completa de uso para las funcionalidades poco usadas					
Una ubicación similar para el mismo tipo de mensajes en los diálogos de la aplicación					
Una disposición de pantalla similar para tareas similares					
En general, asumiría un coste extra por un buen soporte al aprendizaje del programa					

OPERABILIDAD

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre
Una aplicación en la que la velocidad de interacción no venga impuesta por la misma					
Disponer de opciones auto-configurables que puedan modificarse					
La posibilidad de finalización de cualquier tarea o proceso en ejecución del programa					
Disponer de soporte a la accesibilidad independiente de la capacidades técnicas o físicas					

Disponer de control de la información proporcionada por la aplicación					

La posibilidad de Interrumpir los diálogos de la aplicación en cualquier momento					
La posibilidad de seleccionar el modo de hacer las tareas según perfil y preferencias de usuario (por ejemplo: usuario, administrador)					
Disponer de un valor recomendado cuando existen diferentes opciones a elegir					
Disponer de control sobre los datos presentados (modificación, cancelación, etc)					
Posibilidad de ejecución de las acciones con ratón o con teclado					
Posibilidad de almacenar los datos de salida en formato estándar para su uso posterior					
Un aspecto coherente de los diálogos					
Disponer de avisos al usuario si el tiempo de espera va a ser superior al esperado					
Disponer de soporte a la prevención de errores de entrada					
Disponer de ayudas a la corrección de entradas de datos erróneas					
Disponer de validación de los datos de entrada introducidos					
Una aplicación en la que los datos de entrada erróneos no provoquen fallos					
La posibilidad de volver a los valores predeterminados en cualquier momento					
Una aplicación en la que no se pierda la información que se acaba de introducir tras un error					
Mensajes de error en los que se proporcione información sobre la					

acción a tomar					

La posibilidad de cancelar las notificaciones al usuario de los errores corregidos					
La posibilidad de disponer de información adicional sobre un error					
Disponer de avisos y confirmaciones para acciones destructivas (como borrado de datos)					
Posibilidad de corregir datos de entrada erróneos sin cambiar a otro diálogo					
Disponer de documentación en formatos y forma operativos					

En general, asumiría un coste extra por tener una operabilidad del programa aceptable					

ATRACTIVO

Estaría dispuesto a asumir un coste o esfuerzo extra por:	Nunca	Rara vez	Es posible	Muy probable	Siempre
Posibilidad de personalizar la apariencia del programa para ajustarla al gusto del usuario					
Disponer de la posibilidad de personalizar la aplicación para que resulte estéticamente agradable					
En general, asumiría un coste extra por tener un interfaz atractivo de la aplicación					

Si cree que hay alguna característica de productos de seguridad informática que sería importante incluir y que no aparece aquí, por favor, indíquela a continuación:

7.3.5. Cuestionario para datos finales

CUESTIONARIO SOBRE CALIDAD DE PRODUCTOS SOFTWARE DE SEGURIDAD INFORMÁTICA

Para terminar:

	Nunca	Rara vez	Es posible	Muy probable	Siempre
Asumiría un coste extra en la adquisición de productos software de seguridad TIa cambio de una mayor calidad del software:					

	Sí	No
Cree que sería útil disponer de un modelo práctico para la selección de productos de seguridad informática que tenga en cuenta todas las características que expertos en calidad del software, seguridad TI y dirección consideran importantes?:		