

# Desarrollando herramientas para la gestión efectiva y eficiente de la calidad

J.M. Sánchez-Begines, F.J. Domínguez-Mayo, M.J. Escalona, M. Mejías

juan.sanchez@iwt2.org, fjdominguez@us.es, mjescalona@us.es, risoto@us.es

Grupo de investigación IWT2 (Ingeniería Web y Testing Temprano), Universidad de Sevilla, España  
Escuela Técnica Superior de Ingeniería Informática. Avda. Reina Mercedes s/n. 41012  
Sevilla – España

**Resumen:** *La gestión de la calidad del Software es una tarea que indica el grado de concordancia existente entre el software producido con respecto a los requisitos especificados y las necesidades del cliente. En este artículo, hablaremos sobre QuEF (Quality Evaluation Framework) y sus herramientas. QuEF es un framework que permite realizar las distintas tareas de la gestión de la calidad en base a un modelo, de forma que los elementos software usados para la misma puedan ser generados de forma automática o semiautomática en base a la definición del modelo. Inicialmente se desarrolló para analizar, evaluar y mejorar la calidad de metodologías de ingeniería Web guiadas por modelos, aunque actualmente se ha orientado a un framework genérico capaz de gestionar la calidad de productos, procesos o servicios. QuEF define cinco fases las cuales requieren de unas herramientas para poder llevarlas a cabo de forma rápida y eficiente. Las primeras herramientas de soporte fueron desarrolladas como un plugin para la aplicación Enterprise Architect, una herramienta CASE que limitaba su uso a aquellos que tuvieran una licencia. Actualmente se están desarrollando una serie de herramientas que conforman un entorno capaz de llevar a cabo las distintas fases del framework QuEF. Con estas nuevas herramientas, QuEF comienza su independencia en el mercado laboral como una herramienta menos costosa y más asequible técnicamente para los usuarios.*

**Palabras claves:** Calidad del Software, Metodologías, Ingeniería Guiada por Modelos

**Abstract:** *Software quality management is a task which indicates the degree of agreement between the produced software regarding the specified requirements and customer needs. In this paper, we will discuss about QuEF (Quality Evaluation Framework) and its tools. QuEF is a framework that performs different quality management tasks based on a model so that the software elements used for quality management can be generated automatically or semi-automatically according to the model. Initially, this framework was developed to analyze, evaluate and improve the quality of Model-Driven Web Engineering methodologies, but now it has been oriented towards a generic framework to manage the quality of products, processes or services. QuEF defines five phases which require some tools to implement them quickly and efficiently. The first support tools were developed as a plugin for Enterprise Architect application a CASE tool which limited their use to those who have a license. Nowadays a series of tools that make an environment capable of carrying out the various phases of the framework QuEF are in development. With these new tools, QuEF begins its independence in the labor market as a cheaper and attainable tool to users.*

**Palabras claves:** Software Quality, Methodologies, Model-Driven Engineering

## 1. Introducción

La gestión de la calidad del software es una de las preocupaciones de la industria actual ya que una buena gestión de ésta garantiza que un determinado proyecto, producto o servicio cumpla con las expectativas e intereses de todos los interesados. Esta tendencia motiva que actualmente existan múltiples líneas de investigación que se centran en la mejora de la eficiencia y eficacia de los diferentes procesos de gestión de la calidad y que permite obtener un proyecto, producto o servicio con las máximas garantías de calidad. Para una buena gestión de la calidad es necesario el análisis y la evaluación constante del estado actual del proyecto, producto o servicio, así como la definición de políticas, objetivos y planes para su mejora y de esta forma acrecentar la satisfacción del cliente o usuario.

Las entidades que apuestan por la implantación de técnicas y métodos para la gestión de la calidad aporta un gran beneficio tanto en coste al verse reducidos significativamente como en la satisfacción con los clientes de la entidad lo cual mejora los ingresos.

La ingeniería guiada por modelos o, Model-Driven Engineering (MDE) [Butler02], es un paradigma de

desarrollo de software que consiste en la creación de modelos o abstracciones cercanas a un dominio particular en lugar de conceptos complejos de computación.

MDE es un paradigma que ofrece unos beneficios para los procesos de gestión ya que permiten que éstos puedan definirse en base a modelos y realizarse de forma sistemática, a un bajo coste y esfuerzo, y con la mejor calidad.

Por tanto, este paradigma y las metodologías que se basan en él permiten descomponer el producto, proceso o servicio en distintas características y subcaracterísticas de forma jerárquica que permitan ser evaluadas y analizadas para obtener los resultados solicitados por los clientes o usuarios.

Todas estas ventajas pueden ser utilizadas para los procesos de gestión de la calidad.

Por todo esto, se presenta el framework QuEF [Dominguez-Mayo12a] [Dominguez-Mayo12b], cuyo objetivo es el de establecer un marco de referencia que obtenga un entorno de gestión de calidad eficiente y eficaz para una determinada entidad y que permita analizar y evaluar la calidad desde el punto de vista del

propietario de la entidad así como desde el punto de vista de las organizaciones interesadas en la misma.

Por otra parte, solo con la idea de QuEF no se arregla nada. Como en toda gran idea, se necesita de unas herramientas que den soporte a la misma para poderla llevar a cabo y ver su funcionamiento. Para esto, se han desarrollado dos herramientas hasta ahora que permiten llevar a cabo dos de las cinco fases que conforman el QuEF framework. Éstas reciben el nombre de QuEF-D (QuEF-Design) y QuEF-O (QuEF-Operation).

La idea de estas herramientas es que formen un entorno capaz de realizar cada una de las fases del framework de forma automática o semiautomática para los distintos roles que podemos identificar en este proceso y de esta forma poder aplicar la funcionalidad de QuEF a los distintos productos, procesos o servicios de las entidades interesadas así como mostrar a las entidades interesadas su funcionamiento de una forma práctica y visible.

El resto de este artículo está organizado de la siguiente manera. En la Sección 2, se muestran los trabajos previamente realizados y relacionados con la temática tratada en el presente artículo. La Sección 3 describe los resultados obtenidos tras los experimentos realizados. Las conclusiones se encuentra en la Sección 4, y, finalmente, los trabajos futuros en los que está desembocando este trabajo está en la Sección 5.

## 2. Trabajos relacionados

En una primera toma de contacto con el framework QuEF, se comenzó a desarrollar una herramienta para las fases de diseño (llamada QuEF-D) y de operación (llamada QuEF-O) a partir del programa Enterprise Architect.

La herramienta Enterprise Architect [EA14] es una herramienta CASE que permite a los desarrolladores crear diseños y construcciones de sistemas software y que nos proporciona un entorno de modelado que abarca por completo el ciclo de vida del desarrollo de proyectos software, desde la captura de requisitos o especificaciones, pasando por análisis, diseño, implementación y pruebas, hasta el despliegue y el mantenimiento. En nuestro caso Enterprise Architect nos proporcionará un entorno de modelado Unified Modeling Language 2.2 [UML14], mediante el cual se facilita la construcción de modelos.

Con esta herramienta se pretendía elaborar o modelar un sistema con los requisitos que define el framework QuEF y, a través de un plugin, exportar dichos datos para poder construir una web de forma automática para la fase de Operación que permitiera realizar sus tareas tales como el análisis, la evaluación y los planes de mejora continua sobre el modelo generado. Con ello se dio un primer paso para poder visualizar en qué consistía este framework y ver todo su potencial a la hora de realizar los modelos para la gestión de la calidad.

A pesar de la potencia que aportaba esta herramienta, había una serie de problemas los cuales podrían ser solventados con facilidad. Entre ellos destacamos los siguientes:

- El envío de datos se realizaba directamente de la herramienta Enterprise Architect lo cual hacía que su rendimiento se viera reducido.
- Además, su dependencia de dicha herramienta lo convertía en un software más limitado y exclusivo para aquellos usuarios que dispusieran de la herramienta Enterprise Architect.

Finalmente, destacar que dicha herramienta ha sido utilizada en varios proyectos reales en los cuales demostró su eficacia y gracias a ella permitió al QuEF exponer sus cualidades y ventajas de cara al público.

## 3. Experimentación, Desarrollo de herramientas y Resultados

Una vez puesto en contexto se comenzará a explicar qué es y en qué consiste el framework QuEF y, posteriormente, se verá el funcionamiento de las herramientas QuEF-Design y QuEF-Operation implementadas para realizar las tareas que se llevan a cabo de dos de sus fases.

### 3.1. ¿Qué es QuEF?

QuEF (Quality Evaluation Framework) es un framework que permite definir las múltiples tareas que conforman la gestión de la calidad en base a un modelo, de forma que todos los artefactos software utilizados puedan ser generados de forma automática o semiautomática en base a la definición de dicho modelo. Está basado en el conjunto de buenas prácticas de ITILv3 (Information Technology Infrastructure Library) [ITIL,2011]

QuEF se desarrolló inicialmente para analizar, evaluar y mejorar la calidad de propuestas Model-Driven Web Engineering [Escalona04] [Escalona08] [Dominguez-Mayo14] [Vallecillo07] [Schwinger08], aunque actualmente ha evolucionado a un framework genérico que permite gestionar la calidad de productos, procesos o servicios, es decir, planear, controlar y mejorar aquellos elementos de una organización que afectan a la satisfacción del cliente y en la consecución de los resultados deseados por la organización.

QuEF [Dominguez-Mayo12a], como ya se ha dicho anteriormente, se compone de cinco fases que detallamos a continuación:

- **Fase de Estrategia del Modelo de Calidad (Quality Model Strategy Phase):** Esta fase es central al concepto de la gestión de la calidad basada en el ciclo de vida del modelo de calidad, ya que su principal objetivo es que la gestión de la calidad se convierta en un activo estratégico.
- **Fase de Diseño del Modelo de Calidad (Quality Model Design Phase):** Esta fase abarca cada uno de los elementos necesarios para diseñar el modelo de calidad.
- **Fase de Transición del Modelo de Calidad (Quality Model Transition Phase):** El modelo de calidad puede cambiar debido a la identificación de nuevas tendencias o cambios en la tecnología del producto, servicio o proceso. Esta fase se centra en todos

aquellos elementos que son necesarios para que los cambios en el modelo de calidad no influyan en la fase de operación.

- **Fase de Operación del Modelo de Calidad (Quality Model Operation Phase):** En esta fase, el modelo de calidad se utiliza para gestionar la calidad de los productos, procesos o servicios. Se trata de la realización de cada una de las actividades que usan el modelo de calidad para la propia gestión de la calidad como es el análisis, evaluación y planes de mejora continua de la calidad.
- **Fase de Mejora Continua (Quality Continual Improvement Phase):** Esta fase tiene como objetivo alinear el modelo de calidad con las propiedades y características de calidad que deben cubrirse. Debe ser la base para asegurar la estabilidad del proceso y la posibilidad de mejora.

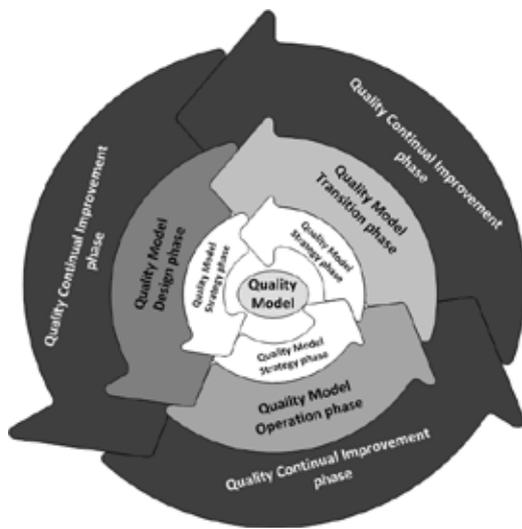


Figura 1. Esquema del QuEF framework al que da soporte el conjunto de factorías de QuEF [Dominguez-Mayo12b],

Los beneficios que se obtienen al aplicar la gestión de la calidad centrandose toda la atención en un único modelo están muy relacionados con los que se obtienen en el paradigma de la ingeniería guiada por modelos.

El propósito del framework QuEF es converger hacia una mejora continua de la calidad automática a través de la producción de checklist y documentación, evaluaciones automatizadas y planes para controlar y mejorar la calidad de forma automática y así conseguir una reducción del esfuerzo, coste y tiempo. Además, obtenemos un aumento del nivel de calidad en el propio proceso de gestión de la calidad trayendo consigo un mayor grado de satisfacción del cliente, así como una mayor facilidad para los usuarios.

QuEF se centra en la gestión de la calidad a medio-largo plazo, aunque también es posible usarlo para la gestión de la calidad a corto plazo. En particular, en una gestión de la calidad a corto plazo, las checklist son usadas para controlar la calidad y poder garantizarla más que para conocer el análisis del estado actual del dominio bajo estudio. Ya se realice una gestión de calidad a medio, largo plazo o a corto plazo, la implantación de QuEF siempre requiere un esfuerzo inicial mayúsculo que se ve

recompensado con los beneficios anteriormente mencionados.

Todo esto favorece a un buen diseño de la calidad y conduce a una gestión de la misma basada en la mejora continua.

Además, esto ha permitido sentar las bases para definir un entorno que establece una estrategia para la realización de la gestión de calidad. Por otro lado, la gestión de la calidad necesita de un modelo de calidad donde se indiquen las expectativas y objetivos que han de ser alcanzados y especificar características que son la base para indicar la calidad requerida y evaluarla.

Con este objetivo, se va a hablar de la plataforma web que se está desarrollando para llevar a cabo las fases anteriormente comentadas, en concreto las fases de diseño y de operación. Con la herramienta para la fase de diseño (QuEF-D) lo que se quiere conseguir es una web sencilla, fácil y usable de cara al usuario que permita crear los modelos y poder visualizarlos de una forma jerarquizada obteniendo así una perspectiva que le permita comprenderlo mejor.

Por otra parte, con la herramienta para la fase de operación (QuEF-O), lo que se quiere obtener es una integración con la herramienta anterior, generándola de forma automática a partir del modelo y de esta forma permita cargarla en una web para poder realizar el análisis y la evaluación de las checklist generadas.

### 3.2. QuEF-Design

Como ya hemos comentado, QuEF-Design es la herramienta encargada de elaborar el modelo de calidad sobre el cual trabajemos a lo largo de todo el proceso que se lleva a cabo en el framework QuEF.

La tecnología utilizada ha sido .NET, ya que es una de las herramientas más utilizadas y demandas en la actualidad.

Además, el lenguaje utilizado (C#) y el framework (.NET) han facilitado el desarrollo de unas herramientas que cumplen con los requisitos y objetivos que se habían marcado desde un principio en el proyecto.

Además, ofrece una integración muy sencilla con la base de datos Microsoft SQL Server lo cual facilitaba mucho la tarea de la base de datos.

Por otra parte, el interés por crear una herramienta independiente que permita llevar a cabo las ventajas que ofrece este framework, ha sido de vital importancia para llevar a cabo el desarrollo de estas herramientas las cuales se han diseñado con mucha simplicidad para evitar las confusiones de los usuarios y puedan cumplir exclusivamente con los objetivos que requiere dicha fase.

Los elementos a tener en cuenta en esta aplicación son los siguientes:

- **Modelo (Quality Model):** Es el modelo que se desea desarrollar para realizar las distintas operaciones que se llevan a cabo en el QuEF y sobre el que gira esta herramienta
- **Características (Features):** Se trata de un conjunto de propiedades referentes al modelo anterior.

- **Subcaracterísticas (Subfeatures):** Es una ramificación dentro de las características anteriores para ampliar el nivel de detalles de éstas.
- **Propiedades (Properties):** En el modelo de calidad, deben indicar la descripción de las características así como las necesidades que debe cubrir.

A estas propiedades se les pueden asignar unas métricas previamente definidas que indican el tipo de medidas a las que se van a someter las propiedades las cuales disponen de un rango de valores definidos por el diseñador del modelo que permitirán realizar unas valoraciones de las mismas en función de la necesidad de los usuarios.

Por otra parte, se encuentran las características de calidad (Quality Characteristics) también referidas al modelo de calidad, que indican las características derivadas de los productos software. Las Quality Characteristics guardan una relación con las Features ya que para la consecución de una de éstas características es necesario cumplir una serie de requisitos definidos por las Features. Las Quality Characteristics, como las características anteriores, pueden ramificarse en subcaracterísticas de calidad (Quality Subcharacteristics) para poder exponer un mayor detalle. Todos estos conceptos han sido adaptado de la norma ISO/IEC 15939 [ISO/IEC11] la cual define un proceso de medida aplicable a las disciplinas de sistema e ingeniería software y de gestión.

Con esto, se procede a presentar fragmentos de la aplicación desarrollada donde se puede ver cómo construir dichos modelos.



Figura 2. Creación de Quality Model

Tal y como se ha comentado, la estructura y visualización de la página está desarrollada de forma que no se mezclen ninguno de los conceptos que hemos definido anteriormente así pues, en primer lugar, aparecen los cuatro elementos comentados en la primera parte de la descripción anterior. Aquí es donde se desarrolla el modelo sobre el cual trabajaremos y a los que se le podrá aplicar las métricas y las características de calidad una vez creados. Cada uno de ellos tiene una serie de atributos que permiten describir un modelo de forma detallada y finalmente, en la parte de las propiedades nos permitirá asociarla con una métrica que esté definida.

Las métricas se definen en un apartado aislado en la misma vista, ya que es un concepto relacionado pero no forma parte de lo que es el modelo en sí. Estas métricas tienen un rango de valores a los cuales se le asignará un valor que será utilizado en la fase de operación para que un usuario pueda evaluar las distintas características del modelo.



Figura 3. Creación de las métricas

Para acabar de introducir los conceptos de esta aplicación, se puede también asignar a un modelo ya desarrollado una serie de características de calidad adicionales y opcionales. Estas son características que los propios autores tienen que asegurar a los equipos de desarrollo. Del mismo modo que hemos hecho con los anteriores, se ha colocado en una parte de la vista de forma que se vea su relación con el modelo pero que aisle los conceptos.

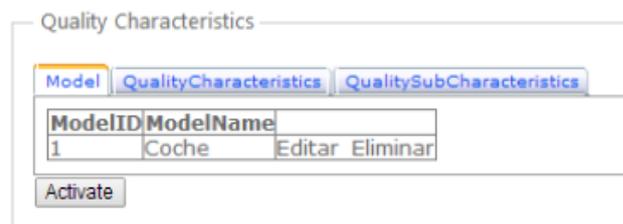


Figura 4. Creación de las características de calidad

El conjunto de cada uno de estos conceptos conforman lo que se llama un modelo de calidad que nos permite obtener una abstracción de un dominio particular. Pero tantos conceptos pueden ser confuso si no se tiene una visualización jerárquica que nos clarifique las partes de dicho modelo.

Es por esto que se ha desarrollado una segunda sección de visualización que nos permite ver de forma jerárquica como queda la estructura de nuestro modelo así como los distintos valores que tienen las propiedades definidas para ellos. Esta sección se ha jerarquizado en forma de árbol o estructura de carpetas que pensamos es la forma más clara que puede ser mostrada este tipo de objeto.

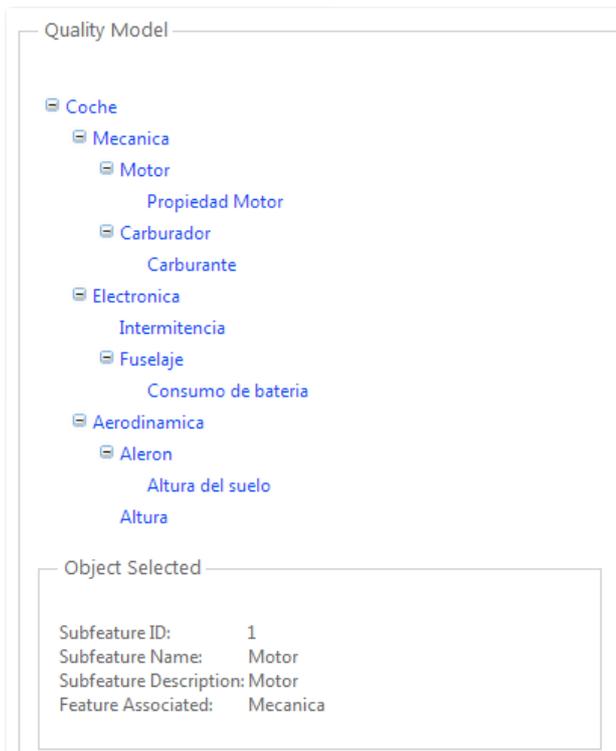


Figura 5. Visualización de un modelo de calidad

Con esta forma de visualización se puede ver claramente las distintas partes que tiene el modelo, de una forma clara y visible. Su comodidad para visualizar cada uno de los elementos que conforman los modelos desarrollados así como de las distintas propiedades ofrecen al usuario una visión del modelo diseñado que le permite saber si realmente lo ha definido conforme a sus necesidades.

Una vez desarrollado el modelo, se puede pasar a la fase de operación. En esta fase, se puede realizar un análisis sobre el modelo desarrollado y poder evaluarlo conforme a las expectativas de los usuarios.

### 3.3. QuEF-Operation

Como bien ya se ha introducido anteriormente, una vez se tenga desarrollado un modelo, es hora de pasar a su evaluación y análisis, tareas que se llevan a cabo en la fase de operación de este framework. Para ello, se ha desarrollado una herramienta capaz de generar de forma automática un sitio web en el que se puede realizar las distintas tareas de análisis y evaluación del modelo que se haya elegido.

Para que esto haya sido posible, se ha integrado la base de datos de la herramienta anterior con esta para obtener todos los datos necesarios para la construcción del sitio. Por otra parte, se han generado otras dos bases de datos que permiten recoger los datos almacenados por los usuarios que quieren analizar y evaluar las distintas checklist que se hayan generado del modelo para, así, realizar el conjunto de fórmulas que se llevan a cabo en la representación gráfica de los valores obtenidos tras su evaluación y de esta forma poder visualizar dichos datos de una forma más visible y clara.

En primer lugar, se muestra la zona en la cual se va a evaluar cada uno de los checklist generados a partir del

modelo generado en la herramienta anterior. Estos checklist se corresponden con cada una de las características (Feature) de las que esté compuesto el modelo generado. En cada uno de ellos hay una subsección en los que se muestran las subcaracterísticas (Subfeature) asociadas a ellas. A su vez, podemos ver las propiedades que definen estas subcaracterísticas justo debajo.

Estas propiedades serán evaluadas a partir de las métricas asignadas a ellas según los rangos que contengan dichas métricas. Estos rangos, tienen un valor predefinido en el modelo el cual será recogido en las bases de datos que comentamos anteriormente. Así, se podrá asignar un valor a cada una de las propiedades de éstas características que permitan evaluarlas.

Figura 7. Asignación de valores

Posteriormente a la asignación de éstos valores, se realizará la asignación del peso a las distintas subcaracterísticas de las checklist anteriores de forma que podamos marcar unas prioridades las cuales nos indiquen la importancia de cada una de ellas dentro del modelo y según convenga al usuario que realiza dicha evaluación. Estos pesos determinarán el resultado de las distintas operaciones matemáticas que se llevan a cabo en la evaluación de estos modelos.

Simplemente, se ha añadido un apartado en el que, a partir de la elección de las características aparezca una tabla con las distintas subcaracterísticas de éstas junto con el valor que se quiera asignar. Por defecto, dicho valor será uno para que no altere la formula si no se desea aplicar pesos.

Name	Field	Value
<a href="#">Editar</a>	Carburador	5
<a href="#">Editar</a>	Motor	5

Figura 8. Asignación del peso

Finalmente, llegamos a la última parte de nuestra web generada para el análisis y la evaluación de los modelos. Todo este proceso se ha llevado a cabo para poder visualizar una serie de gráficas que nos indican las preferencias de nuestros usuarios sobre las características que definen nuestro modelo. Para ello, se ha realizado una media ponderada de los distintos valores asignados a cada una de las propiedades que definen las características del

modelo junto con el peso registrado para las características en cuestión, dando lugar a los valores de importancia que han adquirido cada una de las checklist generadas en nuestra web.

La representación de estos valores se lleva a cabo mediante unas gráficas donde mostramos tanto los valores de las características como de las subcaracterísticas asociadas.



Figura 9. Representación gráfica de valores

Como se puede observar en la Figura 9, se obtienen dos tipos de gráficas según el número de características que queramos mostrar. Para ello se despliega un conjunto de botones que permiten seleccionar que características se desean mostrar y sobre las cuales se van a realizar las operaciones para su evaluación. Los valores de las checklist dependen en gran medida de los pesos asignados a cada una de las subcaracterísticas puesto que, el nivel de importancia asignado a uno de ellos hace que la variación de la media ponderada se desequilibre por lo que es muy importante elegir unos pesos adecuados para ello. En la parte superior, destacando los valores más importantes, se encuentran las gráficas referentes a los checklist mientras que la gráfica de más abajo, indican los valores asignados a las subcaracterísticas.

Con todo esto, se puede realizar un análisis del modelo generado y poder evaluarlo conforme a las necesidades del usuario según sus prioridades.

#### 4. Conclusiones

Una vez realizado el proyecto planteado en un principio consistente en el desarrollo de las herramientas QuEF-Design y QuEF-Operation correspondientes a dos de las fases que se llevan a cabo en el QuEF, ha permitido ver la necesidad de tener unas herramientas que den soporte a una idea.

La idea en sí no es suficiente para poder llevar a la realidad lo que se quiere expresar o decir a través de esa idea. Es necesaria complementarla mediante unas herramientas que hagan posible las funcionalidades que otorga la idea planteada.

Con estas herramientas, se puede llevar a cabo ambas fases con mucha facilidad y rapidez además de ser muy intuitivas para los usuarios. Además de la construcción sencilla, permite una visualización jerárquica que posibilita tener una visión clara y sencilla del resultado generado.

Los puntos fuertes que se pueden obtener de estas herramientas es, en primer lugar, hacer realidad la idea del framework desarrollado de una forma rápida, fácil e intuitiva.

Otro punto a tener en cuenta es la independencia obtenida en esta herramienta a partir del plugin desarrollado en otro proyecto anterior para Enterprise Architect de forma que este abierto a otros usuarios que no dispongan de recursos para adquirir la herramienta anteriormente mencionada y además también a los propietarios de la idea les otorga una independencia que les permite desvincularse de la misma.

Durante el desarrollo de la herramienta, también se han podido aclarar y mejorar conceptos de tal manera que se van asentando y va tomando forma de cara a los que se quiere expresar. Esto, junto con la aplicación de las herramientas en diferentes proyectos reales, ha permitido realizar una validación de las mismas y realizar mejoras que la conviertan en unas herramientas usables y simples.

En definitiva, gracias a esta herramienta, QuEF ha podido dar un paso más en su continua evolución que se espera desemboque en un marco de referencia que obtenga un entorno de gestión de calidad eficiente y eficaz.

#### 5. Trabajos futuros

Durante el desarrollo del proyecto, se ha descubierto nuevas formas de entender y comprender la idea que quiere transmitir QuEF esclareciendo y mejorando los distintos procesos que lleva a cabo así como la aplicación de nuevos conceptos en la construcción del modelo. Con esto, se ha ido convirtiendo en una idea de mayor envergadura y menos abstracta llevando a nuevas formas de llevar a cabo los procesos de este framework.

Una de estas formas es a través de la gestión de etiquetas. Es decir, durante la construcción del modelo, generamos una serie de etiquetas que, simplemente asignándolas a los distintos elementos del modelo, podemos después evaluarlos y analizarlos según la prioridad que otorgue el usuario a dichas etiquetas. Es una forma mucho más simple y mejor integrada que la que se lleva a cabo con las herramientas actuales. Además, el concepto de etiqueta se está aplicando en muchos sitios hoy día lo que nos ahorra mucho trabajo de explicación y entendimiento por parte del usuario.

Como resultado de la realización de este proyecto y a las ideas generadas a partir del mismo, se ha conseguido esclarecer las ideas del QuEF desembocando en un nuevo proyecto, VALORTIA, actualmente en producción, que permite realizar la medición cuantificada y certificación de la productividad personal de los profesionales desarrolladores de software y cuya visión final es implantar un estándar de mercado que contribuya a la maduración del sector de la Tecnología de la Información y la Comunicación, ayudando así a mejorar los procesos de selección y planificación de recursos, contratación asignación del precio y de las tarifas, el seguimiento y control de proyectos, calidad en la entrega y control de desviaciones entre otras muchas actividades.

El proyecto VALORTIA, con el fin de convertirse en una plataforma interactiva en Cloud Computing (denominada VALORTIA: plataforma cloud de VALORación y certificación de competencias TIC Adquiridas), que permita realizar lo anteriormente comentado aplicando el concepto de etiquetas, se está desarrollando como un proyecto de gran envergadura que dé un paso más en la continua mejora en la que está sumergido actualmente QuEF.

## Referencias bibliográficas

- [Butler02] Butler, M., Petre, L., Sere, K., Model Driven Engineering, Integrated Formal Methods, Lecture Notes in Computer Science, Springer Berlin Heidelberg, Volume: 2335, pp. 286-298, 2002, ISBN: 978-3-540-43703-1
- [Dominguez-Mayo14] Domínguez-Mayo, F. J., Escalona, M. J., Mejías, M., Ross, M., & Staples, G. Towards a Homogeneous Characterization of the Model-Driven Web Development Methodologies. *Journal of web engineering*, 13(1-2), 129-159, 2014.
- [Dominguez-Mayo12a] F.J. Domínguez-Mayo, M.J. Escalona, M. Mejías, M. Ross, G. Staples, A quality management based on the Quality Model life cycle, *Computer Standards & Interfaces*, Volume 34, Issue 4, pp. 396-412, ISSN 0920-5489, 2012.
- [Dominguez-Mayo12b] F.J. Domínguez-Mayo, M.J. Escalona, M. Mejías, M. Ross, G. Staples, Quality evaluation for Model-Driven Web Engineering methodologies, *Information and Software Technology*, Volume 54, Issue 11, pp. 1265-1282, ISSN 0950-5849, 2012.
- [EA14] Enterprise Architect, Sparx Systems site, 2014 <http://www.sparxsystems.com.au/>
- [Escalona08] Escalona, M.J., Aragón, G., “NDT. A Model-Driven Approach for Web Requirements”. *IEEE Transactions on software engineering*, Vol. 34, No. 3, pp. 377-390, 2008.
- [Escalona04] Escalona, M.J., Koch, N., “Requirements Engineering for Web Applications – A comparative study”. *Journal of Web Engineering*. Vol. 2, No. 3, pp. 193-212, 2004.
- [ITIL11] ITIL Official site, 2011 <http://www.itil-officialsite.com/>
- [ISO/IEC11] ISO/IEC 15939:2007, Systems and software engineering -- Measurement process, 2011, [http://www.iso.org/iso/catalogue\\_detail.htm?csnumber=44344](http://www.iso.org/iso/catalogue_detail.htm?csnumber=44344)
- [UML14] Object Management Group site, 2014, <http://www.omg.org/spec/UML/2.2/>
- [Vallecillo07] Vallecillo, A., Koch, N., Cachero, C., Comai, S., Fraternali, P., Garrigós, I., Gómez, J., Kappel, G., Knapp, A., Matera, M., Meliá, S., Moreno, N., Pröll, B., Reiter, T., Retschitzegger, W., Rivera, J. E., Schwinger, W., Wimmer, M., and Zhang, G., “MDWenet: A Practical Approach to Achieving Interoperability of Model-Driven Web Engineering Methods”, *Proc. Third Int’l Workshop Model-Driven Web Eng.*, pp. 246-254, 2007.
- [Schwinger08] Schwinger, W., Retschitzegger, W., Schauerhuber, A., Kappel, G., Wimmer, M., Pröll, B., Cachero C., Castro, Casteleyn, S., De Troyer, O., Fraternali, P., Garrigos, I., Garzotto, F., Ginige, A., Houben, G-J., Koch, N., Moreno, N., Pastor, O., Paolini, P., Pelechano V., Ferragud, Rossi, G., Schwabe, D., Tisi, M., Vallecillo, A., van der Sluijs and Zhang, G., “A survey on Web modeling approaches for ubiquitous Web applications”. *International Journal of Web Information Systems* Vol. 4 No. 3, pp. 234-305, 2008.