

Herramientas para el trabajo con Requerimientos No Funcionales que afectan la Arquitectura de Software en la Cátedra de Diseño de Sistemas

María Cecilia Massano, Elizabeth Jeinson

Departamento de Ingeniería en Sistemas

Facultad Regional Córdoba, Universidad Tecnológica Nacional

Maestro Marcelo López s/n, Córdoba, ceciliamassano@gmail.com, ejinson@gmail.com

Resumen

En este artículo se presentan herramientas para trabajar los requerimientos no funcionales como entrada para el diseño arquitectónico de productos de software, en el contexto de la primera etapa del proceso de diseño arquitectónico. Basándonos en las definiciones de requerimientos no funcionales de Ian Sommerville (Sommerville, 2011) y la ISO 25000 (ISO/IEC, 2014), y el concepto de diseño arquitectónico propuesto por Ian Gorton (Gorton, 2006) se presentan dos herramientas, a saber: el modelo de calidad de producto propuesto por la ISO 25010:2011 (ISO/IEC, 2011) y una plantilla de especificación de requerimientos no funcionales. El objetivo del artículo es demostrar que los estudiantes abordan el tema con elementos que faciliten la identificación y especificación de los requerimientos no funcionales, así como la vinculación con la arquitectura de un producto de software.

Palabras clave: *diseño de sistemas, requerimientos no funcionales, diseño arquitectónico.*

1. Introducción

La especificación de los requerimientos es una tarea con múltiples dificultades. Aun suponiendo que las actividades anteriores a la especificación (elicitación y análisis de requerimientos) (Sommerville, 2011) hayan superado las barreras de la comunicación entre analistas y usuarios, y se haya logrado interpretar acertadamente los requisitos, la correcta especificación sigue siendo un reto. Existen múltiples formas de especificar requerimientos, tanto mediante métodos

formales como en lenguaje natural más o menos estructurado (casos de uso y otras técnicas). En este sentido, es necesario que se definan de manera específica y consistente, sin ambigüedades posibles (Mannion & Keepence, 1995). En esa misma línea, los requerimientos no funcionales deben ser verificables (IEEE, 1998), lo que implica que sean específicos, medibles, accesibles, realizables y de tiempo limitado (Mannion & Keepence, 1995), estableciendo los criterios de aceptación. Además, es necesario negociar con el cliente requerimientos realizables que su inversión de infraestructura sea capaz de atender (Mannion & Keepence, 1995). A partir de esta necesidad, hacemos foco en esta temática en pos de lograr buenas especificaciones en el ámbito profesional. Por ello, a continuación, se presenta una propuesta basada en el modelo de calidad de producto de la ISO 25010:2011 (ISO/IEC, 2011), y bajo el marco teórico que define requerimientos no funcionales (Sommerville, 2011) y diseño arquitectónico (Gorton, 2006). Dicha propuesta consta de dos herramientas a utilizar en el aula, durante el cursado de Diseño de Sistemas (de Información) o materias afines, para la enseñanza de la identificación y especificación de requerimientos no funcionales, y el posterior vínculo con el diseño arquitectónico de productos de software.

2. Marco teórico

El proceso del diseño de la arquitectura de un producto de software descrito por Gorton inicia con la determinación de los requerimientos arquitectónicos significativos

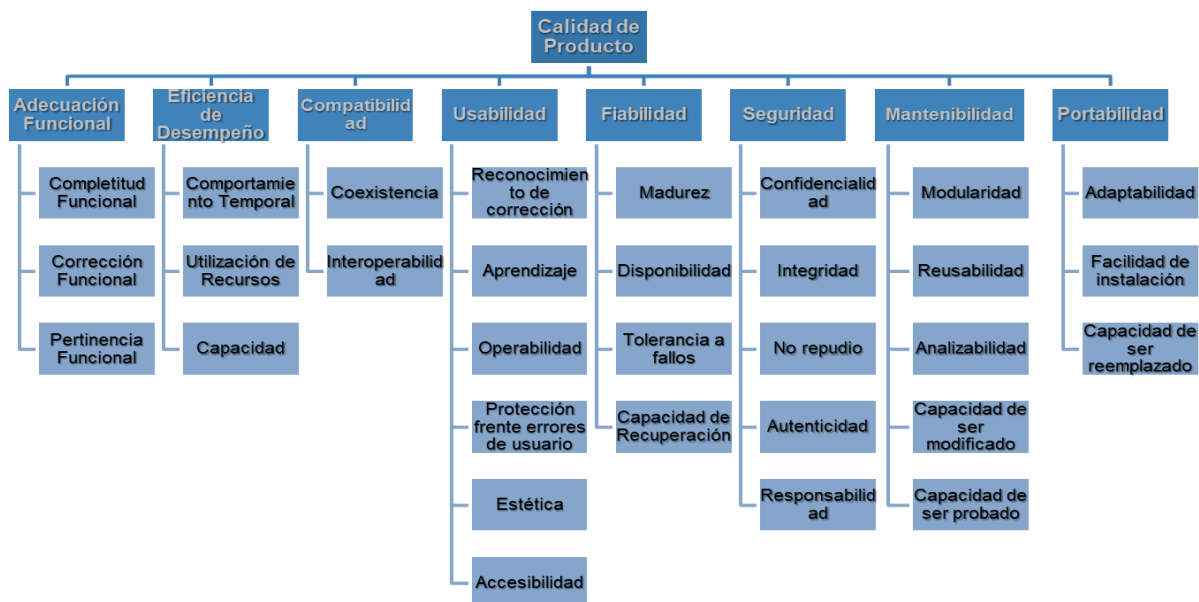


Fig. 1 - Modelo de Calidad de Producto - ISO 25010:2011

para la arquitectura. La segunda etapa es el diseño de la arquitectura y la etapa final es la validación de la arquitectura (Gorton, 2006). Estos requerimientos arquitectónicos están compuestos por los requerimientos funcionales y los “requerimientos de los involucrados” o necesidades de los involucrados (Gorton, 2006) (T. de A.). Estos últimos conforman las restricciones al sistema o los atributos de calidad. La funcionalidad es la habilidad del sistema para hacer el trabajo que se espera de él. Debe ser independiente de las estructuras que componen el sistema. Sin embargo, esas estructuras son necesarias para implementar los atributos de calidad que se imponen como necesidades de los involucrados tales como la usabilidad, modificabilidad, desempeño (Bass, Clemens, & Kazman, 2003). Los requerimientos no funcionales se definen por la negativa ya que describen las propiedades emergentes del sistema, y no los servicios que ofrece. También definen restricciones sobre el sistema tales como capacidad de dispositivos o las interfaces con otros sistemas (Sommerville, 2011).

La ISO/IEC 25000:2014 (ISO/IEC, 2014) es una serie denominada SQuaRE (Systems and software Quality Requirements and Evaluation) que presenta un conjunto de modelos de calidad con el objetivo de conformar un marco para la especificación, medición y evaluación de la calidad del software. La serie que se enfoca en los modelos propone los siguientes: Calidad de

Producto (ISO/IEC 25010:2011), Calidad en Uso (ISO/IEC 25010:2011), Calidad de Datos (ISO/IEC 25012:2008). La estructura de estos modelos de calidad se basa en categorizar la calidad de un producto en características. Estas características pueden a su vez desglosarse en sub-características. Se genera así una descomposición jerárquica que proporciona una visión completa respecto a la calidad del producto. El modelo aclara que, si bien el agrupamiento pretende ser representativo de una idea de calidad, puede no ser exhaustivo ni único. Es decir, en ocasiones, distintas características, pueden superponerse al concepto de calidad que están interpretando. Cada característica o sub-característica debe tener una o más propiedades de calidad que sean mensurables. Luego, identifica las propiedades de calidad en uso, que son aquellas que capturan la percepción del usuario cuando experimenta el uso real o simulado del producto o sistema en cuestión. En el modelo de Calidad de Producto (ISO/IEC, 2011) basamos este trabajo, como base para identificar requerimientos no funcionales.

3. Objetivos y Metodología

Presentamos una propuesta vinculada a conceptos de Diseño de Arquitectura aplicados a la práctica. Este tema es contenido mínimo de la cátedra de Diseño de Sistemas en el programa aprobado 2007 de la carrera de Ingeniería en Sistema de

Información, en la Universidad Tecnológica Nacional (Consejo Superior Universidad Tecnológica Nacional, 2007).

El objetivo de la propuesta es encontrar una forma simple para que los estudiantes encuentren y especifiquen requerimientos no funcionales de manera correcta.

Con el propósito de lograr la identificación de los requerimientos no funcionales, los ejercicios prácticos constan de una descripción de un dominio particular, en donde se incluyen dentro del texto requerimientos no funcionales. Para facilitar dicha identificación previa presentación de estos casos prácticos, se le presenta a los estudiantes el modelo de calidad de producto propuesto por la ISO 25010: 2011 (ISO/IEC, 2011), como muestra la Fig. 1. Este modelo incluye la descripción de cada una de las características y sub-características del modelo. Este material puede ser utilizado tanto en las clases como en las instancias de evaluación práctica. El objetivo de explicar y entregar este material es que los estudiantes se familiaricen con el concepto de requerimiento no funcional o atributo de calidad, y que cuenten con una herramienta que los ayude a encontrar estos requerimientos. Conociendo el modelo los estudiantes pueden plantearse ante ejercicios prácticos preguntas vinculadas a la identificación de los requerimientos. Cabe aclarar que la característica *Adecuación Funcional* no se desarrolla en este tema ya que abarca los requerimientos funcionales, que no son foco de la unidad de Diseño de Software.

Sumando al modelo de calidad de producto también se le presenta a los alumnos una tabla tipo plantilla para especificar los requerimientos no funcionales identificados. La plantilla ordena la forma en que se identifican los requerimientos no funcionales, de tal forma de asegurar su correcta especificación. La tabla contiene:

1. **Número:** identificación unívoca de los requerimientos no funcionales.
2. **Nombre:** Frase para identificar al requerimiento no funcional.
3. **Descripción:** En este apartado el requerimiento debe quedar claramente especificado, cumpliendo los principios

SMART (Mannion & Keepence, 1995) para asegurar que el requerimiento sea verificable en una etapa de pruebas del software (IEEE, 1998).

4. **Significativo para la arquitectura:** En esta columna se determina si el requerimiento afecta la arquitectura de software indicando SI ó NO. El objetivo de esta columna es que los estudiantes logren identificar los requerimientos que condicionan la arquitectura del software, como primer paso del proceso de diseño arquitectónico.

5. **Justificación:** La última columna requiere que los estudiantes expliquen porqué indicaron que un requerimiento no funcional afecta o no la arquitectura del software. Para esto pueden utilizar la descripción de las características y sub-características del modelo de calidad de producto propuesto por la ISO 25010:2011 (Sommerville, 2011).

Con este formato de plantilla los estudiantes pueden realizar prácticas de identificación y especificación de requerimientos no funcionales, así como identificar los requerimientos que determinan la arquitectura del software, tema que atañe a los contenidos mínimos de la cátedra de Diseño de Sistemas.

El propósito de esta propuesta es lograr afianzar los conocimientos obtenidos en las materias de años anteriores vinculados a la temática de requerimientos no funcionales, y generar nuevos conocimientos vinculados al diseño arquitectónico. Esto se debe lograr en no más de dos clases prácticas, ya que la extensión de los contenidos no permite abarcar más horas cátedra. Por esta razón, se buscó implementar una forma simple de enseñar y realizar prácticas de identificación y especificación de requerimientos no funcionales que permitan para avanzar con la disciplina de diseño arquitectónico.

A continuación, se presenta un caso de aplicación donde se muestra la implementación de las herramientas propuestas. El caso de estudio es utilizado en 4 divisiones la cátedra de Diseño de Sistemas, de la Universidad Tecnológica Nacional, Facultad Regional Córdoba.

El ejercicio está basado en la gestión de una empresa concesionaria de peaje. Para caracterizar el caso práctico se presentan sus objetivos y alcances:

Objetivo: gestionar pasadas de usuarios de un peaje y el cobro correspondiente, según la definición de tarifas y las formas de pago vigentes.

Alcances:

- Gestión de pasadas de vehículos por las casillas el peaje.
- Administración de tarifas en función de categorías de vehículos y rutas.
- Administración de clientes y de sus vehículos.
- Administración de casillas de peaje.
- Gestión del cobro en efectivo y con tarjeta de crédito.
- Administración de usuarios, roles, empleados y turnos de atención.

Para la práctica y evaluación de la temática *Diseño Arquitectónico*, se plantea una descripción del dominio planteado, donde se explica el caso, incluyendo en el texto requerimientos no funcionales. Al momento de trabajar con el Proceso de Diseño Arquitectónico, comenzamos con la etapa de Determinar los Requerimientos Arquitectónicos (Gorton, 2006). Los estudiantes deben identificar los requerimientos no funcionales en la descripción del dominio, y determinar cuáles son los significativos para la arquitectura. Para esto se les entregan las dos herramientas mencionadas. El modelo de calidad de producto los ayuda a encontrar requerimientos no funcionales en el dominio planteado. Por ejemplo, ante la siguiente descripción: *Para las fechas pico el sistema debe soportar la pasada de vehículos en simultáneo por las 10 casillas de cada una de las rutas donde se encuentra un peaje de la empresa (son 9 rutas en total). Además, debe trabajar eficientemente con este tráfico considerando que en un minuto pueden pasar hasta 10 vehículos. Se considera que el sistema es eficiente si procesa una pasada en hasta 4 segundos.*

Utilizando el modelo de calidad de producto, si buscamos requerimientos asociados a la característica de *Eficiencia*

de Desempeño, podemos identificar uno de ellos en la descripción anterior.

A partir de la plantilla presentada anteriormente, se especifica el requerimiento eligiendo un nombre para representativo con una frase breve. Se selecciona la característica correspondiente del modelo de calidad de producto. Esto tiene por objetivo que el estudiante tenga que hacer un análisis respecto a establecer si están cubiertas las características que garantizan la calidad de producto. Aunque no es necesario indicar la sub-característica, los estudiantes cuentan con la descripción de la jerarquía completa, de tal manera que les facilite la identificación. En el ejemplo que venimos desarrollando, el estudiante se puede ayudar con las siguientes definiciones de la ISO 25010:2011 (ISO/IEC, 2011). La característica *eficiencia de desempeño* se define como el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones establecidas.

Las sub-características asociadas son:

1. **Comportamiento temporal:** el grado con el cual la respuesta, los tiempos de procesamiento, tasas de rendimiento de un producto o sistema cuando ejecuta sus funciones cumple con los requerimientos.
2. **Utilización de recursos:** el grado con el cual las cantidades y los tipos de recursos utilizados por un producto o sistema cuando ejecuta sus funciones cumplen con los requerimientos.
3. **Capacidad:** el grado con el cual los límites máximos de los parámetros de un producto o sistema cumplen con los requerimientos. Los parámetros pueden ser ítems almacenados, usuarios concurrentes, ancho de banda, rendimiento de las transacciones y tamaño de base de datos.

Puede observarse que la descripción provista responde a las sub-características de capacidad: *para las fechas pico el sistema debe soportar la pasada de vehículos en simultáneo por las 10 casillas de cada una de las rutas*, y de comportamiento temporal: *se considera que el sistema es eficiente si procesa una pasada en hasta 4 segundos*. De esta forma,

el estudiante podrá identificar fácilmente un requerimiento no funcional a partir de la descripción provista.

Tabla 1 – Extracto de Ejemplo de Requerimiento de Calidad/No Funcional en Ejercicio de Peaje

Nombre	Característica	Descripción
Performance de procesamiento en horario pico	Eficiencia de desempeño	Para las horas pico el sistema debe poder soportar el procesamiento en simultáneo de 90 transacciones, procesando cada transacción en hasta 4 segundos considerando que en un minuto pueden pasar hasta 10 vehículos.

Como segundo paso el estudiante deberá determinar si este requerimiento afecta la arquitectura de software. Para esto también utiliza el modelo de referencia de calidad de producto de la ISO 25010:2011 (ISO/IEC, 2011). A partir de las definiciones de la ISO/IEC los estudiantes deben ser capaces de identificar si el requerimiento afecta la arquitectura del software, y debe justificar su elección.

En el ejemplo dado el requerimiento identificado afecta la arquitectura de software, por lo que se requiere una justificación. Cabe aclarar que la justificación enunciada por los estudiantes evoluciona con el tiempo. En las primeras justificaciones que realizan, cuando aun no tienen vastos conocimientos para expresar justificaciones más completas, se espera que indiquen una aproximación de una decisión arquitectónica.

A modo ejemplificador, se presenta la solución:

Tabla 2 – Extracto de Ejemplo de Justificación de Requerimiento de Calidad/No Funcional que afecta la Arquitectura de Software

Nombre	Justificación
Performance de procesamiento en horario pico	Requerimiento de concurrencia y tiempo de respuesta que <i>afecta la arquitectura ya que</i> requiere el desarrollo de un módulo de lectura y procesamiento de pasadas que debe cumplir con este requerimiento de performance.

4. Resultados

La aplicación de las dos herramientas propuestas se ha llevado a cabo en el aula desde hace varios años. Se mostrarán resultados de los últimos 5 años. El modelo de calidad de producto ISO/IEC comenzó a utilizarse en 2018, por lo que aún está probándose su aplicación. Sin embargo, los resultados son comparables ya que esta herramienta se entregaba a los estudiantes en años anteriores. El modelo anterior era un cuadro similar que plasmaba modelos de otros autores: Ian Sommerville (Sommerville, 2011), Ian Gorton (Gorton, 2006) y la ISO/IEC 9126-1:2001 (ISO/IEC, 2001) que evolucionó a la ISO/IEC 25000:2014.

A continuación, se presenta una tabla con la cantidad de horas cátedra consumidas para ver el tema de requerimientos no funcionales, como parte de la primera etapa del proceso de diseño arquitectónico, desde la clase práctica.

Tabla 3 - Hs Cátedra de Trabajos Prácticos Consumidas para el tema Requerimientos No Funcionales

		Curso			
		1	2	3	4
Año	2018	5	5	5	4
	2017	5	5	5	5
	2016	4	3	5	3
	2015	3	3	3	3
	2014	3	3	3	S/D

Durante 2018, a partir de la utilización del modelo de calidad de producto propuesto por la ISO 25010:2011 (ISO/IEC, 2011) notamos que es posible avanzar en la misma

clase de requerimientos no funcionales con la aplicación de decisiones arquitectónicas relacionadas a su identificación. Esto justifica un leve aumento en la cantidad de horas cátedras dedicadas al tema.

Respecto a los resultados de las evaluaciones en donde se incluye este tema, se presenta la siguiente tabla:

Tabla 4 - Porcentaje de Aprobación de Parcial Donde se Incluye el Tema de Requerimientos No Funcionales

Año	Curso				Tipo de evaluación
	1	2	3	4	
2018	No evaluado	No evaluado	No evaluado	No evaluado	2do parcial práctico
2017	100%	92%	100%	94%	3er parcial práctico
2016	92%	96%	55%	62%	3er parcial teórico/práctico
2015	71%	61%	58%	83%	2do parcial teórico/práctico
2014	71%	54%	S/D	S/D	2do parcial teórico/práctico

Se puede observar la mejora en los porcentajes de aprobación a medida que se afianza la utilización de las herramientas propuestas.

Respecto a un análisis cualitativo de los resultados de la aplicación de las herramientas propuestas, se percibe en el aula que los estudiantes logran comprender los conceptos, basados en los conocimientos previos y valiéndose de las herramientas mencionadas para identificar los requerimientos no funcionales.

5. Conclusiones

Podemos concluir que los estudiantes utilizan ambas herramientas simplificando el trabajo de identificación de los requerimientos no funcionales, a partir de un modelo de calidad de producto avalado por una organización internacional de estandarización: la ISO/IEC. El contar con estas herramientas también en las instancias de evaluación logra que los estudiantes se enfoquen en aprender a utilizarlas y no en memorizar su contenido.

De esta manera se logra tanto una buena identificación y descripción de requerimientos no funcionales, como así también una correcta justificación de la implicancia en la arquitectura del software.

En función de los resultados cuantitativos, podemos aseverar que la dedicación a este tema en el aula se puede minimizar a no más de 5 horas cátedra. Esto permite optimizar los tiempos y reforzar el tema de diseño arquitectónico, sin afectar los resultados en las evaluaciones. Estas herramientas abordan la temática en forma simple y didáctica, y los estudiantes pueden utilizarla en ambientes educativos y profesionales.

Referencias

Bass, L., Clemens, P., & Kazman, R. (2003). Software Architecture in Practice - Segunda Edición. Boston: Addison Wesley.

Consejo Superior Universidad Tecnológica Nacional. (2007). Ordenanza 1150 - Diseño Curricular Carrera Sistemas de Información. Buenos Aires.

Gorton, I. (2006). Essential Software Architecture. Editorial Springer.

IEEE. (1998). Recommended Practice for Software Requirements Specifications. En IEEE, IEEE Std 830.06-1998.

ISO/IEC. (2001). ISO/IEC 9126-1. Ginebra: ISO/IEC©.

ISO/IEC. (2011). ISO/IEC 25010:2011(en). Ginebra: ISO/IEC©. Obtenido de www.iso.org

ISO/IEC. (2014). ISO/IEC 25000:2014. Ginebra: ISO/IEC©.

Mannion, M., & Keepence, B. (1995). Software Engineering Notes Volumen 2 Número 2. Edimburgo: ACM SIGSOFT.

Sommerville, I. (2011). Ingeniería de Software. Mexico: Addison Wesley.