



Un Caso de Estudio para la Adopción de un Modelo de Trazabilidad de Requisitos en el Sector Energético

A Case Study for the Adoption of a Requirement Tracking Model for Energy Sectors

Lina María Montoya-Suárez, MSc

*Facultad de Ingenierías y Arquitectura
Fundación Universitaria Luis Amigó
Medellín, Colombia.
lina.montoyasu@amigo.edu.co*

Juan Carlos Monsalve-Gómez, MSc

*Facultad de Ingenierías y Arquitectura
Fundación Universitaria Luis Amigó.
Medellín-Colombia
juan.monsalvego@amigo.edu.co*

Jorge Mauricio Sepúlveda-Castaño, MSc

*Facultad de Ciencias Básicas e Ingeniería
Corporación Universitaria Remington
Medellín, Colombia,
jsepulveda@remington.edu.co*

(Recibido el 09-09-2014. Aprobado el 12-12-2014)

Resumen. Es fundamental para un desarrollo de software, mejorar el proceso relacionado con los requisitos, los aspectos que facilitan una mejora, es contar con la posibilidad de realizar trazados a los requisitos a lo largo del ciclo de vida de un sistema. Esta trazabilidad permite evaluar las implicaciones que puede tener el cambio y posibilita su detección en fases tempranas o incluso en la fase de gestión de la configuración, siguiendo las huellas de trazado.

A través de este artículo de investigación, se presenta un caso de estudio aplicando la herramienta Enterprise Architect para el sector energético, en el que se evidencia cómo se puede dar seguimiento a un requisito, comenzando en la primera fase de análisis y diseño (requisitos funcionales, requisitos no funcionales, regla de negocio, casos de uso, diagrama de clase, diagrama de componente, etc.), por medio del cual es posible generar una matriz de trazado dentro de un proceso de desarrollo de software.

Palabras claves: Trazabilidad, trazabilidad de requisitos, arquitectura empresarial.

Abstract. In order to improve software development process relating to requirements for improvement, it is necessary to have a possibility of requirements drawn along the life cycle of a system. For this reason, traceability allows evaluating the potential impacts of change so we can detect changes at an early stage or even at the stage of configuration management in the footsteps of software development.

Through this paper a case study is presented using the Enterprise Architect tool applied to the energy sector while evidencing how to track a requirement beginning in the first phase of analysis and design (functional requirements, non-functional requirements, business rules, use cases, class diagram, component diagram etc.) generating array layout within a software development process.

Keywords: Traceability, requirements traceability, enterprise architecture.

Citación de artículo, estilo IEEE:

L.M. Montoya-Suárez, J.C. Monsalve-Gómez, J.M. Sepúlveda-Castaño "Un Caso de Estudio para la Adopción de un Modelo de Trazabilidad de Requisitos en el Sector Energético", *Lámpsakos*, N° 13, pp. 88-100, 2015.

DOI: <http://dx.doi.org/10.21501/21454086.1494>

1. INTRODUCCIÓN

En la Ingeniería de Software, hacer trazabilidad en los requisitos permite obtener en el producto soluciones más exactas y fiables, con base en las necesidades solicitadas por el cliente. La trazabilidad adquiere gran importancia debido a que es la “aptitud para rastrear la historia, la aplicación o la localización de una entidad mediante indicaciones registradas” [1][2].

Es fundamental hacer el seguimiento a los requisitos cuando se desea construir un software de calidad, lo cual se constituye en un proceso de análisis y estudio detallado. Cuando ocurren cambios en el software, la trazabilidad hace que sea relativamente más fácil evaluar el impacto que los cambios podrían tener en otras partes del proceso de desarrollo; esto permite seguir las huellas en los requisitos [3].

Es primordial para la fase de desarrollo de software poder fijar líneas bases desde el comienzo hasta el final de las etapas, logrando una perspectiva horizontal; una de las mejores prácticas consiste en rastrear las actividades mismas, las historias que estas presentan, pues permiten a los desarrolladores no caer en los mismos errores, buscando soluciones en menos tiempo y mayor rendimiento. Para describir y seguir los requisitos, se genera en dos direcciones: hacia delante (forward) y hacia atrás (backward), como lo expresa Ramesh y Jarke [4].

Los errores en la gestión de requisitos pueden ser muy costosos para las empresas; contar con un mecanismo integral de trazabilidad facilita la finalización de las tareas relacionadas con pruebas y documentación, para esto la trazabilidad garantiza que todos los requisitos sean diseñados y que todos los diseños se codifiquen y se prueben, aunque no solo resalta las pruebas que se deben actualizar o repetir, sino que también señala los documentos (análisis de riesgos, especificaciones y manuales del usuario) [5].

El objetivo fundamental de este artículo de investigación es, mediante el proyecto IDICA, presentar un caso de estudio para el sector energético, con el propósito de hacer matrices de trazado de los requisitos y así rastrear las historias, haciendo seguimiento en diferentes dimensiones de análisis y diseño como lo permite la herramienta Enterprise Architect. Esta contribución técnica se estructura de la siguiente manera: En la sección 2, se presentan las generalidades de trazabilidad de requisitos. En la sección 3, se

estructuran los elementos y metodología empleada en la investigación. En la sección 4, los resultados obtenidos de las matrices generadas y, finalmente, se presentan las conclusiones.

2. GENERALIDADES DE TRAZABILIDAD DE REQUISITOS

Cuando un requisito pasa por un proceso de calidad es necesario tener la confianza acerca de la corrección y viabilidad del requisito por el cual permite hacer preguntas como [6]: ¿Qué ocurre con la especificación de los requisitos en conjunto?

Para la especificación de los requisitos, se hace referencia a la colección de requisitos especificados y definidos previamente. La especificación no tiene que estar en un determinado formato, puede ser una especificación sobre papel, o un blog, o algo similar [6][3][7].

Cuando la especificación de los requisitos está completa, se tendrá un conocimiento preciso del alcance, el impacto, la funcionalidad del producto y el costo asociado. Este es el momento de llevar a cabo la revisión de la especificación, en la cual se validan todos los requisitos.

Es importante tener en cuenta y asegurarse de que los requisitos tengan consistencia y completitud; en caso contrario, cualquier conflicto entre los requisitos ha sido resuelto [6]. Dos requisitos están en conflicto si no pueden implementarse juntos, es decir, si la solución a un requisito impide la implementación de otro.

Hay que observar cuando las expectativas del cliente son altas, es elemental tener en cuenta la fecha de entrega y la disponibilidad de los recursos; además, asegurarse de que las funciones más importantes del producto sean entregadas tan pronto como sea posible. Otro problema que puede suceder es que haya demasiados requisitos. La solución a los problemas anteriores es la priorización de los requisitos funcionales, no funcionales y la de regla de negocio [6] [8].

A continuación se puede observar cómo se clasifican los requisitos:

Tabla 1. Clasificación de los requisitos de software [9].

Factor de Calidad	Atributos
Funcionalidad	Característica y capacidades del Programa
	Generalidad de las funciones
	Seguridad del sistema
Facilidad de uso	Factores humanos
	Factores estéticos
	Consistencia de la interfaz
	Documentación
Confiabilidad	Frecuencia y severidad de los fallos
	Exactitud de las salidas
	Tiempo medio de fallos
	Capacidad de recuperación ante fallos
	Capacidad de predicción
Rendimiento	Velocidad de procesamiento
	Tiempo de respuesta
	Consumo de recursos
	Rendimiento
	Eficacia
Capacidad de Soporte	Extensibilidad
	Adaptabilidad
	Capacidad de prueba
	Capacidad de configuración
	Compatibilidad
	Requisitos de instalación

Matriz de Trazabilidad

Un concepto clave en el proceso de gestión de cambios es la trazabilidad. Los requisitos deben ser trazables, es decir, “rastreadables”. Se podría decir que un requisito es trazable si se pueden identificar todas las partes del producto existente relacionadas con ese requisito. Todos los requisitos deberían ser trazables para mantener consistencia entre los distintos documentos de un proyecto [7].

Es importante conocer aspectos de los requisitos tales como [8]:

- Su origen (Quién los propuso).
- Necesidad (Por qué existe).
- Relación con otros requisitos (Dependencias).
- Relación con otros elementos (Dependencias).

Una de las claves para la gestión de cambio es la matriz de trazabilidad, la cual permite establecer las relaciones entre los diferentes requisitos desde un

punto de vista jerárquico y la documentación generada durante todo este proceso es la base sobre la que debe construirse todo el plan de pruebas. El uso de matrices de trazabilidad es una buena técnica para llevar a cabo esta actividad de forma eficiente [9] [10].

En la matriz se irán registrando los requisitos de negocio. Por cada requisito de negocio, se identificarán los requisitos de usuario correspondientes. De cada requisito de usuario se identificarán cuáles son los requisitos de sistema asociados a cada uno de ellos. Y así sucesivamente, se irá llenando toda la matriz de requisitos con el fin de gestionar [11].

La siguiente matriz se utiliza para relacionar requisitos. Es una matriz de dependencias, como se muestra en la Figura 1.

En este caso, los requisitos X representan los requisitos que originan las dependencias y los requisitos Y serían los requisitos que dependen de otros requisitos, de los requisitos X.

Por ejemplo, se puede ver cómo los requisitos Y 1,4 y 7 dependen del requisito X 2, o también depende del requisito X 5. También se observa cómo los requisitos Y 2 y 5 dependen del requisito 1. De esta forma, se puede ver de qué manera se relacionan los requisitos, para analizar mejor el impacto de los cambios.

Otro tipo de matriz de dependencias muy usada es la *matriz hacia atrás / hacia adelante*

- Requisitos vs especificación
- Especificación vs caso de prueba
- Especificación vs diseño
- Diseño vs código

La construcción de estas matrices trae beneficios más allá de un simple registro de la correlación o dependencia entre los elementos de los modelos [11]. A partir de ellas, es posible analizar o inferir información acerca del nivel de especificación de los requisitos, el nivel de participación de los usuarios, el costo asociado a cada fase de desarrollo, la arquitectura requerida, el plan de las pruebas, etc. Si las matrices son actualizadas continuamente, pueden usarse como herramienta de soporte para el personal encargado del mantenimiento y las pruebas del Sistema [12].

		REQUISITOS (X)					
		R1	R2	R3	R4	R5	Rn
REQUISITOS (Y)	R1		X				X
	R2					X	
	R3						
	R4		X	X			X
	R5					X	
	R6						
	R7		X	X			X
	Rn			X		X	

Fig. 1. Matriz de dependencias requisitos X vs requisitos Y [6] [1].

3. ELEMENTOS Y METODOLOGÍA

Para el caso de adopción de un modelo de trazado, para el grupo de desarrollo INDICA (Índices de Calidad), se generó una matriz de trazabilidad para las necesidades, con el fin de garantizar la consistencia y la completitud de los modelos de desarrollo. Mediante estudio de casos, según Fong, enmarcada dentro de cinco fases, se definió lo siguiente [13]:

A. El fenómeno que se aborda en el estudio

El objetivo es aplicar un caso de estudio por medio de la herramienta Enterprise Architect, haciendo matrices de trazado para el proyecto INDICA, en el que se evidencia cómo se puede dar seguimiento al levantamiento de los requisitos.

B. Proposición

La investigación que se expone pretende responder la siguiente pregunta: ¿Es posible generar seguimiento de los requisitos por medio de modelo de trazado para el proyecto INDICA?

C. Unidades de análisis de la investigación

Estas unidades de análisis son:

- 1) Identificar los modelos de trazado y su aplicabilidad.
- 2) Aplicación del proyecto INDICA por medio de un modelo de trazado para el levantamiento de los requisitos.
- 3) Generación de matrices de trazabilidad.

D. Definición de lógica que relaciona los datos con las hipótesis del estudio

- 1) Trazabilidad de requisitos, desarrollo de software, calidad en el desarrollo de software.
- 2) A partir de la teoría de la trazabilidad de requisito, identificar cuál es la evidencia necesaria para verificar y garantizar la objetividad.
- 3) De un caso de estudio, evidenciar el modelo de trazado con base al levantamiento de requisitos.

E. Criterios para interpretación de datos obtenidos

- 1) Matriz de trazabilidad
- 2) Necesidades vs stakeholders
- 3) Necesidades vs requisitos
- 4) Necesidades vs proceso
- 5) Casos de uso vs validaciones y mensajes del sistema
- 6) Casos de uso vs regla de negocio
- 7) Casos de uso vs requisitos funcionales generales
- 8) Casos de uso vs requisitos no funcionales
- 9) Casos de uso vs diagrama de actividades
- 10) Requisito no funcional vs requisitos funcionales generales
- 11) Requisitos vs interfaces.

El proyecto se trata de un sistema INDICA que permite cargar datos de los operadores de red del sistema eléctrico nacional, acerca de las indisponibilidades de sus circuitos y transformadores que generar índices, con el fin de realizar auditoría y control de cali-

dad, lo cual redundará en beneficio de los operadores de distribución local.; este sistema genera proceso de calidad para los operadores de distribución local.

El modelo de trazabilidad propuesto tiene el objetivo de orientar el uso de la matriz de trazabilidad de los requisitos, casos de usos y necesidades del sistema. Está compuesto por un conjunto de pasos que se explican por medio del desarrollo del caso de estudio INDICA para calcular índices del sistema de distribución local.

Se usó para el caso de estudio la herramienta Enterprise Architect. Es una plataforma de modelado, diseño y administración, colaborativa, basada en UML 2.3 y estándares relacionados. Ágil, intuitiva y extensible, permite visualizar, analizar, modelar, probar y mantener un amplio rango de sistemas, software, procesos y arquitecturas [14]

Esta herramienta provee trazabilidad completa desde los modelos de requisitos, análisis y diseño, hasta la implementación y despliegue. La verificación y validación efectiva y el análisis del impacto inmediato son posibles a través del ciclo de vida completo, usando capacidades tales como la matriz de relaciones y la vista de jerarquía de Enterprise Architect [15].

4. RESULTADOS

Actualmente, se mide la calidad del servicio que presta el Sistema de Distribución Local (SDL) en Medellín-Colombia, para que el usuario final perciba los beneficios o perjuicios relacionados con este y el Operador de Red (OR) tenga un incentivo para prestar el mismo con ciertos estándares de eficiencia.

A raíz del inicio de un nuevo periodo tarifario, cambia la metodología para medir la calidad del SDL (resolución CREG (Comisión Reguladora de Energía y Gas 097/2008), por lo que es necesario definir un sistema de información para respaldar este hecho. Expertos en Mercados (XM), en su calidad de LAC (área de liquidación de cuentas en XM), se ve impactado debido a que se le asignan nuevas responsabilidades, de tal manera que estará encargado de gestionar las actividades relacionadas con el cálculo de índices de calidad en el SDL. Adicionalmente los OR se ven afectados, ya que dependiendo del comportamiento

de los índices, podrán tener posibles variaciones en sus ingresos (a favor o en contra), vía cargos por uso del SDL.

La Figura 2 muestra los *stakeholders* directos e indirectos. Para los directos, presenta 9 actores: La entidad reguladora tiene asociados los actores SSPD (Administrador del sistema SUI, a través del cual se obtiene la información relacionada con las interrupciones declaradas por los OR y la información del PAR Programación de activo en reparación) y CREG (Encargado de definir las reglas de negocio que deben ser usadas para llevar a cabo el cálculo de índices de calidad del SD). Para los indirectos, 11 actores, de los cuales el líder técnico tiene asociados a 4.

4.1. Matriz de trazabilidad

Al tener toda la información recopilada en artefactos, en Enterprise Architect se obtienen las matrices de trazabilidad que se relacionen a cualquier elemento del sistema INDICA. Las matrices de trazabilidad y las restricciones ayudan al proyecto a verificar que la metodología se cumpla correctamente [15].

Las matrices de trazabilidad tienen una doble funcionalidad:

- Obtener las plantillas por cada tipo de artefacto que hacen referencia a otros elementos relacionados.
- Permitir la trazabilidad de los elementos entre las fases.

Para el proyecto INDICA se han definido las siguientes matrices de trazabilidad:

- 1) Necesidades vs. *stakeholders*
- 2) Necesidades vs. requisitos
- 3) Necesidades vs. procesos
- 4) Casos de uso vs. validaciones y mensajes del sistema
- 5) Casos de uso vs. regla de negocio
- 6) Casos de uso vs. requisitos funcionales
- 7) Casos de uso vs. requisitos no funcionales
- 8) Casos de uso vs. diagrama de actividades
- 9) Requisito no funcional vs. requisitos funcionales generales

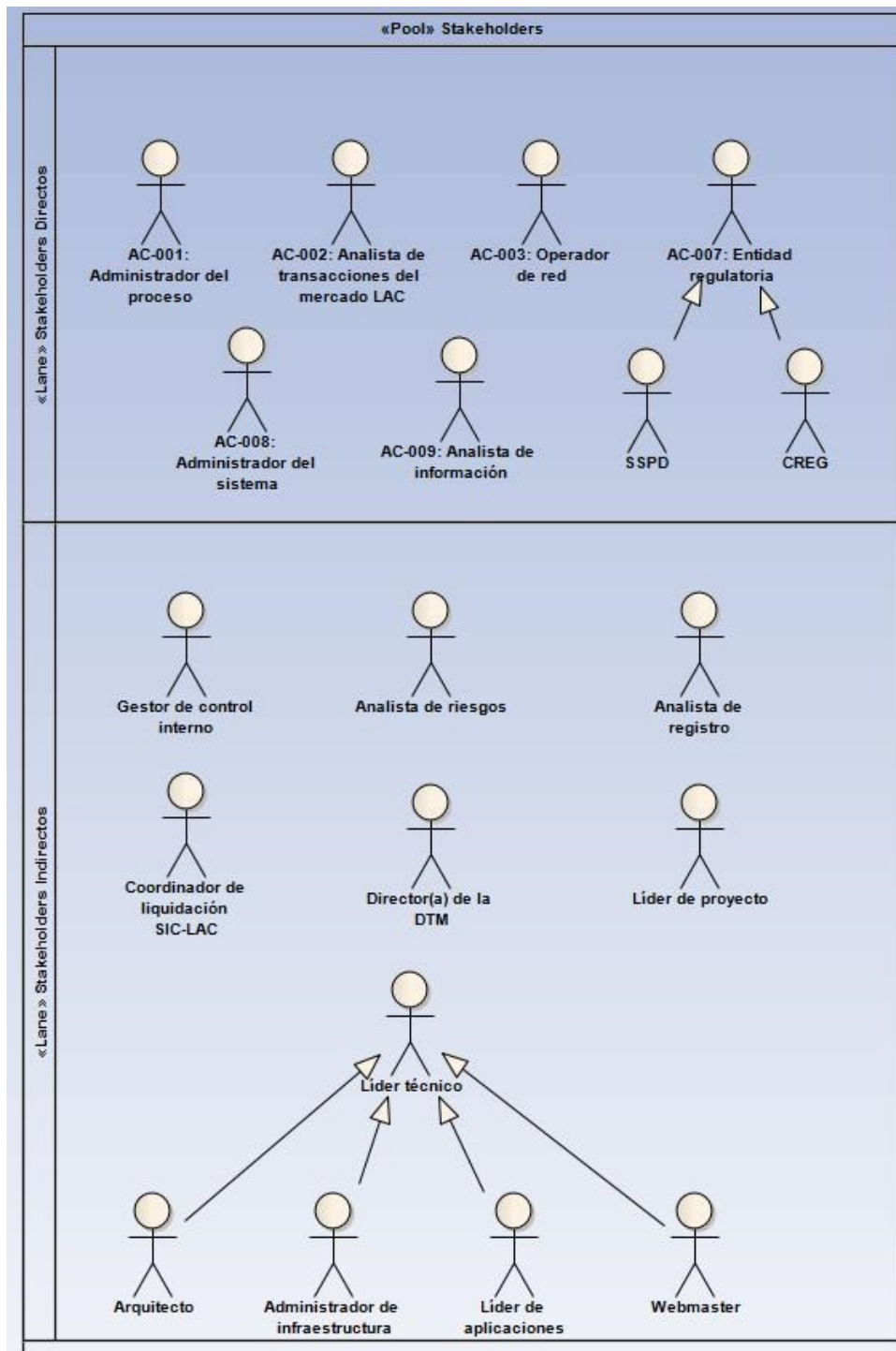


Fig. 2. Stakeholders directos e indirectos [7].

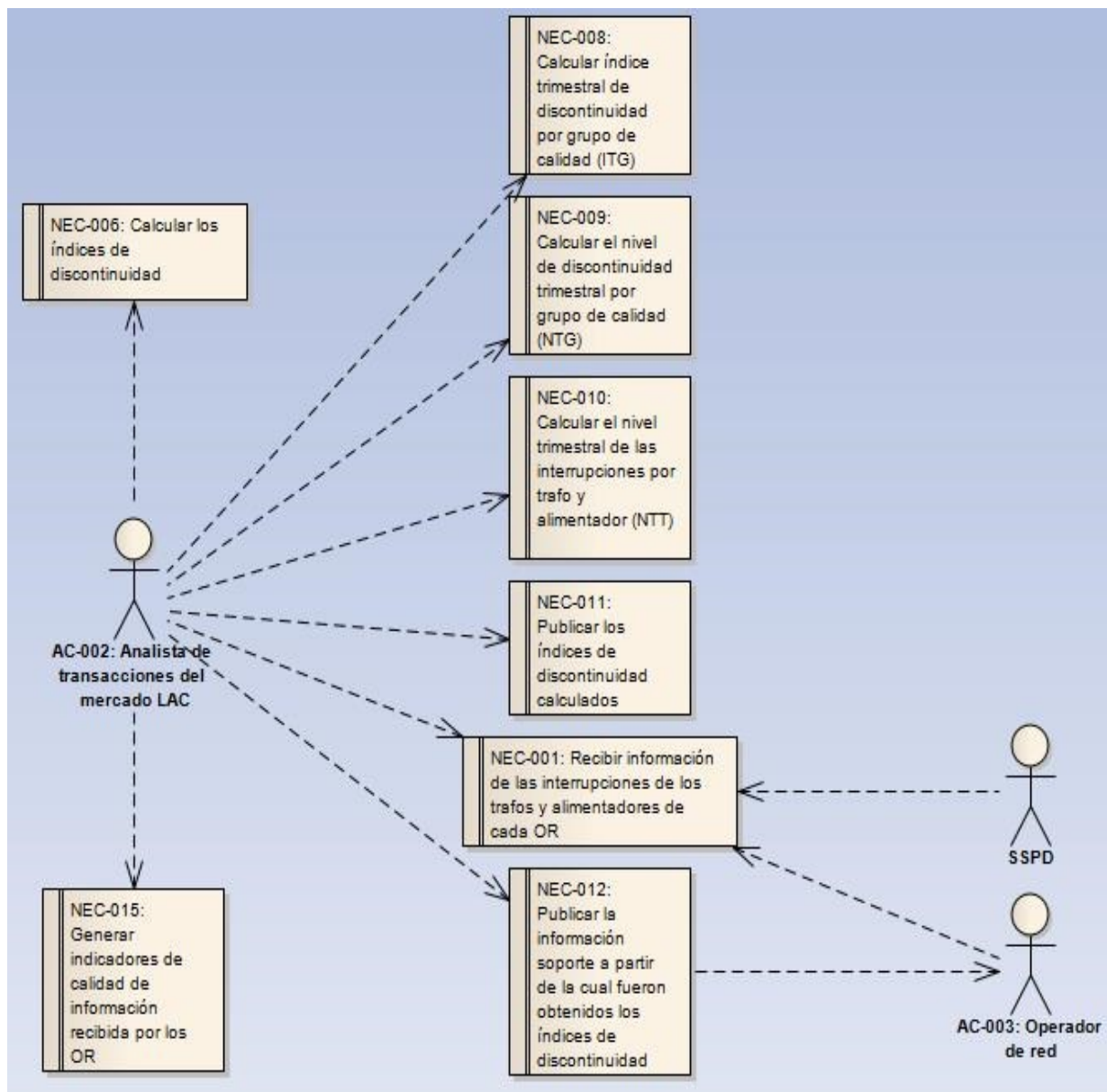


Fig. 3. Necesidades vs. stakeholders [7].

10) Requisitos vs. interfaces generales

Las matrices de trazabilidad se generan y se consultan accediendo a la vista de recursos (resources) del Enterprise Architect.

4.2. Selección de matriz de trazabilidad

Una vez que el proceso de desarrollo de software es seleccionado por el equipo de desarrollo, ellos crean o seleccionan matrices de trazabilidad después de atender las sugerencias definidas.

Aunque el patrón facilite la creación de diferentes tipos de modelos de trazado para soportar las actividades de gestión de la configuración, el método de trazabilidad es probado con dos definiciones genéricas:

- 1) El de matriz de trazabilidad basado en casos de uso vs. clase. (Anexo 1)
- 2) El de matriz de trazabilidad basado en casos de uso vs. requisitos. (Anexo 2)

- 3) El de matriz de trazabilidad basado en casos de uso vs. sistema. (Anexo 3)

Para el método de matriz de trazabilidad basado en necesidades, el vínculo de trazado se define de la siguiente forma:

- 4) Necesidades asociado con los stakeholders. (Figura 3)

La selección de la matriz de trazabilidad para el alcance del sistema permite dar solución al flujo de información requerido en empresas del sector energético, pues se pueden gestionar los cálculos de los índices de calidad del SDL y hacerlos públicos a los stakeholders involucrados directamente, es decir que se facilita el análisis y comportamiento de los índices.

El lector puede referirse a los siguientes anexos:

Anexo 1: Matriz de trazabilidad casos de uso vs. requisitos funcionales: De esta forma, se puede ver de qué manera se relacionan los casos de usos con los requisitos funcionales y así analizar mejor el impacto de los cambios.

Anexo 2: Matriz de trazabilidad casos de uso vs clase: De esta forma, se puede ver de qué manera se relacionan los casos de usos con la clase y así analizar mejor el impacto de los cambios.

Anexo 3: La matriz de trazabilidad casos de uso vs sistema: De esta forma, se puede ver de qué manera se relacionan los casos de usos con los sistemas.

5. CONCLUSIONES

La presentación de este tema mediante un caso de estudio para el Proyecto de Software INDICA permitió generar matrices de trazabilidad, con la finalidad de que los desarrolladores, así como los ingenieros de requisitos, lograran tener un conocimiento pleno de los artefactos del sistema y su relación entre ellos, tanto dentro de la aplicación como con otras aplicaciones de software enlazados al proyecto, para la práctica de la trazabilidad y la transformación de modelos en cualquier entorno de desarrollo de software.

El marco conceptual, en primera instancia, estandariza el uso de los vínculos de trazado definidos en la herramienta Enterprise Architect [15], con el fin de controlar la transformación de modelos. Esto ayuda a eliminar una de las grandes incertidumbres de los desarrolladores, al usar una relación de dependencia para la trazabilidad, ya que sus definiciones son tan similares que no es fácil identificar la situación real de uso, además de la utilidad de usar los diferentes tipos ofrecidos por la herramienta.

La matriz de trazabilidad generada de manera horizontal y vertical permite un mejor cubrimiento de los artefactos involucrados en un sistema, de forma que se pueda hacer un seguimiento de trazado hacia adelante y hacia atrás dentro del desarrollo de software.

La práctica de la trazabilidad aplicada por la herramienta Enterprise Architect permite hacer una administración de los requisitos a la hora de generar un desarrollo de software. Es de suma importancia considerar el personal de desarrollo o calidad dedicado a la validación y verificación de la trazabilidad de requisitos, para que establezcan un modelo de trazado con el cual se pueda medir el nivel de exactitud de los modelos de desarrollo con respecto a los requisitos planteados por los usuarios y los clientes.

REFERENCIAS

- [1] ISO for Standardization, ISO 8402: 1994: *Quality Management and Quality Assurance Vocabulary*. International Organization for Standardization, 1994.
- [2] B. G. Dale, T. Van Der Wiele, and J. Van Iwaarden, *Managing quality*, John Wiley & Sons, 640p., 2013.
- [3] V. Anaya and P. Letelier, "SmartTrace: Una herramienta para trazabilidad de requisitos en proyectos basados en UML", en *WER*, pp. 210-224, 2002.
- [4] B. Ramesh and M. Jarke, "Toward reference models for requirements traceability," *Softw. Eng. IEEE Trans.*, vol. 27, no. 1, pp. 58-93, 2001.

- [5] P. Letelier and V. Anaya, "Integrando especificaciones textuales y elementos de modelado UML en un marco de trabajo para trazabilidad de requisitos," en *JISBD*, pp. 2002, 151-162.
- [6] INTECO, Instituto Nacional de Tecnologías de la Comunicación, "Guía práctica de gestión de requisitos", España, 2008.
- [7] L. M. Montoya, "Levantamiento de un estado del arte para la trazabilidad de requisitos," Tesis de maestría, Universidad de Medellín, 2011.
- [8] J. A. Aguilar, I. Garrigós, and J. N. Mazón, "Modelos de weaving para trazabilidad de requisitos web en A-OOH," Actas del VII Taller sobre Desarrollo de Software *Dirigido por Modelos*, *JISBD*, Congreso Español de Informática (CEDI), Valencia, España, SISTEDES, 2010, pp. 146-155.
- [9] M. A. Carvajal, M.M. Sandoval, *La trazabilidad en el proceso de requerimientos de software*, 2008, 6p. Disponible en: <http://www.iiis.org/cds2008/cd2008csc/cisci2008/paperspdf/c601uz.pdf>
- [10] D. G. Firesmith, *Object-oriented requirements analysis and logical design: A software engineering approach*, John Wiley, New York, NY, USA: John Wiley & Sons, Inc., 1993, pp. 1-250.
- [11] J. Mendoza Castelán, *Unified Modeling Language*, Universidad Autónoma del Estado de Hidalgo, México, 2011, pp. 1-19.
- [12] B. W. Boehm, "Verifying and validating software requirements and design specifications," in *IEEE Software*, 1984, pp. 1-20.
- [13] C. F. Reynoso, "El estudio de casos en la investigación relacionada con la Pyme en México," *Revista Internacional de Investigación y Aplicación del Método de Casos*, vol. 21, p. 4, 2009.
- [14] J. W. Ross, P. Weill, and D. C. Robertson, *Enterprise architecture as strategy: Creating a foundation for business execution*, USA: Harvard Business Press, 2006, pp. 1-241.
- [15] E. Architect, *Sparx Systems*, 2012. Disponible en: <http://www.sparxsystems.com.au/>

Anexo 2: Matriz de Trazabilidad Casos de Uso vs Clase.

	Domain::AdminstracionPlazoReporte	Domain::AgenteCalidad	Domain::Bitacora	Domain::BitacoraDetalle	Domain::Carga	Domain::CargaDiarria	Domain::CargaTrimestral	Domain::CausaEvento	Domain::Documento	Domain::GrupopPlazo	Domain::GrupopPlazoAgente	Domain::Hoja	Domain::ProcesosBitacora	Exception::Reso097SDLException	Negocio::AdministracionService	Negocio::CargaEventosService
Administración::CU-011: Administrar perfiles del sistema		X												X		
Administración::CU-012: Asignar perfiles a usuarios del sistema					X			X				X				
Administración::CU-025: Autorizar OR para SDL												X				
Administración::CU-026: Definir plazos máximos						X									X	
Administración::CU-027: Definir plazos máximos de carga de información							X								X	
Administración::CU-028: Definir plazos máximos de cálculos y publicación	X					X	X	X								
Administración::CU-029: Administrar los tipos y códigos de los tipos de interrupción			X								X			X		
Administración::CU-030: Definir rangos de validez de datos								X								
Administración::CU-031: Administrar documentos					X						X				X	
Administración::CU-032: Administrar hojas	X				X										X	
Administración::CU-033: Autorizar recarga de información a los OR						X		X							X	
Administración::CU-042: Cambiar contraseña (Eliminado)								X								
Carga de Información::Administrador mapeo de OR y comercializador			X						X							
Carga de Información::CU-001: Descargar información del PAR			X													
Carga de Información::CU-004: Cargar información mensual OR desde un archivo de datos					X											X
Consultas y Reportes::CU-040: Consultar indicadores	X							X								X
Consultas y Reportes::CU-041: Consultar bitácora del sistema																
Consultas y Reportes::CU-056: Generar reporte consolidado por elemento												X				
Procesar Información::CU-024: Publicar índices e insumos									X							
Procesar Información::CU-050: Verificar factibilidad de ejecución de los cálculos		X												X		X

Anexo 3: Matriz de Trazabilidad Casos de Uso vs Sistema.

Administración::CU-011: Administrar perfiles del sistema	MA::MA-001	MA::MA-002	MA::MA-003	MA::MA-004	MA::MA-005	MA::MA-006	MA::MA-007	MA::MA-008	MA::MA-009	MA::MA-010	MA::MA-011	MA::MA-012	MA::MA-013	MA::MA-014	MC::MC-001	MC::MC-002	MC::MC-003	MC::MC-004	MC::MC-005	MC::MC-006	MC::MC-007
Administración::CU-012: Asignar perfiles a usuarios del sistema																					
Administración::CU-025: Autorizar OR para SDL													X								
Administración::CU-026: Definir plazos máximos																					
Carga de Información::CU-004: Cargar información mensual OR desde un archivo de datos					X															X	
Carga de Información::CU-005: Cargar información diaria OR a través de un servicio B2B		X								X											X
Carga de Información::CU-006: Cargar información mensual OR a través de un servicio B2B																					
Carga de Información::CU-007: Validar información cargada por el OR			X							X											
Carga de Información::CU-008: Generar indicador de calidad de la información del OR - Eliminado																					
Carga de Información::CU-010: Eliminar descargas de información del SUI		X																			
Carga de Información::CU-046: Cargar información mensual del SUI desde un archivo de datos										X											X
Carga de Información::CU-047: Cargar información del PAR desde un archivo de datos																					
Carga de Información::CU-048: Validar información del PAR																					
Carga de Información::CU-049: Validar la información mensual del SUI																					
Carga de Información::CU-058: Interfaz agentes comercial			X																		
Carga de Información::CU-059: Autenticar identidad usuario																					X
Consultas y Reportes::CU-056: Generar reporte consolidado por elemento		X																			
Consultas y Reportes::CU-057: Generar reporte oficial de eventos																					X
Procesar Información::CU-009: Verificar completitud de la información SUI																					
Procesar Información::CU-054: Calcular duración diaria de interrupciones																					
Procesar Información::CU-055: Calcular acumulado de interrupciones		X																			
Procesar Información::CU-060 Calcular EPD																					