

Especificación de Requerimientos para Sistemas que emplean Servicios Web en Cloud Computing

Zalazar, Ana Sofía¹; Gonnet, Silvio^{1,2}; Leone, Horacio^{1,2}

¹INGAR (UTN-CONICET) - Instituto de Desarrollo y Diseño

²Universidad Tecnológica Nacional, Facultad Regional Santa Fe

Abstract

Cloud computing, como nuevo paradigma de negocios de servicios informático no posee un procedimiento para la especificación de los requerimientos en el modelo de software como servicio. La primera actividad que un consumidor en la nube debe realizar es la captura de las necesidades que espera solucionar con la contratación de los servicios cloud y clarificar los requerimientos, principalmente aquellos no funcionales que ponen en manifiesto la calidad de servicio. Un buen análisis de requerimientos bajo el entorno de cloud computing permitirá a los consumidores de servicio conocer los aspectos necesarios en la adquisición de funcionalidades en los entornos de la nube y comparar los diferentes proveedores que se ajusten a las especificaciones del consumidor de servicio. Cuando los requerimientos son detectados, se sientan las bases de los acuerdos de nivel de servicio para poder contratar las prestaciones y gestionar los cambios en los requerimientos, que bajo este entorno son muy comunes. Si un consumidor desconoce los aspectos funcionales y de calidad del servicio que va utilizar, los acuerdos de servicios pueden no cubrir las expectativas de la implementación esperadas por los consumidores. Por lo tanto, en este trabajo se analizan propuestas existentes de ingeniería de requerimientos para estas actividades y se propone un enfoque para cloud computing con la finalidad de acompañar el proceso de captura y la especificación de requerimientos para la contratación de software en el contexto de la nube, la creación de los acuerdos de servicios y la adquisición de servicios Web.

Palabras Clave

Cloud Computing, Software como Servicio, Ingeniería de Requerimiento, Acuerdo de Nivel de Servicio.

1. Introducción

La computación en nube, también conocida como *cloud computing*, es un nuevo paradigma de negocio gestionado a través de Internet, donde los diferentes proveedores ofrecen sus servicios utilizando

mecanismo de virtualización escalable. Generalmente, se realiza un análisis de los objetivos y los requerimientos de un producto de software, para diseñarlo y programarlo en su totalidad. En cambio, la computación en la nube permite utilizar las rutinas y procedimientos desplegados en Internet, como si fuera un servicio que se paga por su uso.

Debido a la naturaleza estocástica y dinámica de los contextos de la nube, no existe un procedimiento ampliamente aceptado para la ingeniería de requerimientos, y la manera de relacionar los requerimientos con los servicios de software ofertados por los proveedores.

Ingeniería de Requerimiento (RE) consiste en cuatro actividades principales: licitar, especificar, gestionar y formalizar requerimientos. En la especificación de los requerimientos, que es un proceso incremental e iterativo, cada interacción puede indicar nuevas necesidades y más detalles de la implementación del servicio [1].

Sin embargo, la selección de servicios depende muchas veces del acuerdo de nivel de servicio (SLA), que es un tipo de contrato entre proveedores y consumidores, donde queda plasmado los requerimientos que serán cumplidos y las funcionalidades de los servicios adquiridos. Este acuerdo es la base para el control, validación, y adopción de medidas, cuando el nivel de servicio no es el esperado por el consumidor.

Dentro del entorno de computación en la nube, es necesario poder definir un marco de trabajo para dar soporte a los consumidores en la ingeniería de requerimientos, en especial en la etapa

temprana de un proyecto de migración o implementación de servicios de software en la nube.

Repschlaeger et al. [2] presentan un marco de trabajo para la adopción de soluciones en la nube a través de un análisis de indicadores de servicios, dejando de lado el análisis de conflictos entre estos parámetros. Y Hanna et al. [3] provee una lista de requerimientos de aplicación que espera que los servicios de cloud cumplan.

En el trabajo de Escalona y Koch [4] se presenta un análisis comparativo de los requerimientos manejados por las metodologías web, para capturar, validar y especificar requerimientos web.

En este trabajo, se realiza un análisis de dos enfoques (SoRE y GoRE) que han cobrado importancia los últimos años para la especificación de requerimientos de servicios [5][6]. Basado en estos métodos se elabora el enfoque de este trabajo, teniendo en cuenta que para entornos de computación en la nube, los requerimientos son dinámicos, escalables y temporalmente modificables.

A diferencia de otros sistemas, los servicios de software que necesita el consumidor usualmente no deben ser construidos porque se encuentran desplegados en la nube, y es el consumidor quien debe evaluar qué servicio es más acorde a sus necesidades específicas.

Por último, se propone un método para especificar requerimientos para aplicaciones que emplean servicios ofrecidos en la nube, cuya principal actividad inicial es captar las necesidades y objetivos que deben resolverse por los servicios ofrecidos en la nube. También permite al consumidor evaluar las ofertas de servicios de diferentes proveedores y generar contratos de servicio que contiene todos los requisitos previstos.

El aporte de este trabajo es presentar una estrategia alternativa para realizar ingeniería de requerimiento y ofrecer una guía para la adquisición de software como servicios en la nube, basándose en los pasos

propuestos por Zardari y Bahsoon [5] para la adopción de servicios en la nube.

A continuación se presenta una síntesis de Computación en la Nube, Software como Servicio e Ingeniería de Requerimiento. Luego, se analizan las diferentes propuestas de Ingeniería de Requerimiento que podrían aplicarse al modelo de software como servicio y se presenta la estrategia propuesta para llevar adelante las actividades prescriptas por la ingeniería requerimientos en entornos de computación en la nube. Para concluir, se presenta un caso de aplicación de la propuesta como una validación preliminar y se realizan las conclusiones finales, definiendo algunas líneas de investigación para continuar completando la estrategia planteada.

Computación en la Nube

La definición mayormente aceptada y que abarca las características más importantes de computación en la nube es la de *National Institute of Standards and Technology* (NIST) [7]:

“Cloud computing es un modelo que permite acceso a redes bajo demanda, para compartir un conjunto de recursos de computación configurable (es decir, redes, servidores, almacenamiento, aplicaciones y servicios) que pueden ser rápidamente provistos o liberados con un mínimo esfuerzo de administración o interacción con los proveedores de servicio.”

Bajo el marco de esta definición, Mell y Grance [8] puntualizan cinco características esenciales:

- *Autoservicio a demanda*: el consumidor se aprovisiona automáticamente de los recursos según sus necesidades.
- *Amplio acceso a redes*: el acceso a los recursos se realiza por medio de diferentes plataformas clientes.
- *Puesta común de recursos*: los recursos son compartidos por medio de múltiples clientes, usando mecanismos de virtualización y abstrayendo la ubicación exacta de los recursos.
- *Rápida elasticidad*: los recursos son adicionados o liberados de forma

rápida, de manera de aprovecharlos al máximo posible.

- *Medición de Servicios*: los recursos utilizados son medidos, controlados y monitorizados, de forma transparente, para poder ser facturados.

Los cuatro modelos de despliegue de servicios que especifica la NIST son:

- *Nube privada*: los recursos y accesos son de uso exclusivo de una organización con múltiples clientes internos.
- *Nube comunitaria*: los recursos de la nube son compartidos por una comunidad de organizaciones.
- *Nube pública*: la infraestructura de la nube es compartida por varios clientes independientes y existe una interfaz que asegura la independencia y gestiona los recursos de los clientes.
- *Nube híbrida*: es una combinación de una nube privada y una nube pública, y aprovecha rápido aprovisionamiento público cuando existen picos de trabajos.

Los tres modelos de servicios, los cuales abarca cualquier tipo de servicio son:

- *Software como un Servicio*: el servicio está formado por aplicaciones que los usuarios finales pueden acceder a través de interfaces, donde se abstraen los servidores que soportan la capa lógica del software.
- *Plataforma como un Servicio*: el servicio ofrecido es un contenedor que posee un entorno de programación, con bibliotecas y herramientas que dan soporte al desarrollo, y las tareas de administración y configuración del hardware son realizadas por el proveedor del servicio.
- *Infraestructura como un Servicio*: el servicio está dado por máquinas virtuales, memoria de almacenamiento, base de datos y componentes de redes, dejando de lado los detalles administrativos y de mantenimiento de estos recursos físicos.

La NIST también definen cinco roles de referencias para este nuevo paradigma de negocios:

- *Proveedor de Servicio*: responsable de poner un servicio a disposición de las partes interesadas y se encarga de la administración del hardware subyacente.
- *Consumidor del servicio*: es quien solicita un servicio del proveedor para completar su proceso de negocio.
- *Bróker del Servicio*: es el mediador encargado de que se lleve a cabo el contacto entre el consumidor y el proveedor de servicio.
- *Auditor del Servicio*: verifica que se lleve a cabo las evaluaciones independientes de los servicios de cloud, y de rendimiento de la nube de acuerdo a los contratos establecidos.
- *Portador del Servicio*: se encarga de proporcionar el transporte y la conectividad de los servicios desde los proveedores a los consumidores, y cumplen un papel importante para la migración de servicios en el contexto de la nube.

El motivo por el cual, la computación en la nube, como nuevo paradigma de computación distribuida, ha tenido tanto auge en los últimos años es porque de alguna manera ha incorporado el concepto de economía de escala en el empleo de los diversos tipos de recursos informáticos, tanto en calidad de servicio como en flexibilidad. Además los proveedores de servicios poseen mejores prácticas y experiencia en la gestión de servidores y dispositivos de red, lo que provoca mayor seguridad y confianza a los clientes inexperimentados.

La computación en la nube ha permitido que los accesos a los servicios sean llevados a campo por cualquier medio y dispositivo, de manera flexible y escalable, reutilizando módulos de códigos y conectando aplicaciones a través de Internet.

Para muchas organizaciones, la computación en la nube ha representado una manera de cubrir sus requerimientos de

alta disponibilidad, fiabilidad y tolerancia a fallos, consumiendo servicios de distribución geográfica muy dispersa. Sin embargo, la aceptación de la computación en la nube depende ampliamente de la manera que los servicios cumplan con los requerimientos funcionales y no funcionales planteados por los consumidores del servicio [9] [10].

Software como Servicio en la Nube

Las aplicaciones basadas en servicios son desarrolladas como software independiente que ofrecen una interfaz bien definida a los usuarios potenciales. Mientras que el servicio encapsula la funcionalidad del negocio, alguna forma de infraestructura de servicio es requerida para facilitar las interacciones y comunicación entre servicios. Un servicio web (“*webservice*”) es un servicio que se encuentra programado utilizando lenguajes y protocolos de Internet, también puede ser implementado a través de interfaces auto descriptivas basadas en estándares web [11].

Los webservices constan de componentes integrados basados en arquitectura orientada a servicio (SOA), que pueden ser adaptados, configurados, y hasta extendidos en tiempo de ejecución [12]. Cada webservice de los proveedores de servicios es identificado por *Uniform Resource Identifier* (URI) accesibles en Internet o una red privada, y puede ser publicado, localizado y accedido a través de mensajes codificados en estándares basados en *Extensible Markup Language* (XML) y transmitidos utilizando protocolos de internet.

Los consumidores o brókeres de servicios buscan los webservices que cumplan con la funcionalidad y calidad requerida, utilizando registros de *Simple Object Access Protocol* (SOAP), *Web Services Description Language* (WSDL) y *Universal Description, Discovery and Integration* (UDDI).

UDDI es un estándar de publicación y descubrimiento de los componentes de software basado en SOA, y ofrece un

mecanismo para clasificar, catalogar, y administrar los webservices, para que puedan ser descubiertos y consumidos por otras aplicaciones, ya sea en una red pública o en la infraestructura interna de una organización.

En resumen, el consumidor o bróker del servicio encuentra el servicio a través de los registros UDDI, que tienen una vinculación a la descripción del servicio, y a través del protocolo SOAP se comunica con este servicio (Fig. 1).

El mecanismo de software como servicio permite la obtención de procesos de negocios complejos y transacciones como webservices, y permite la composición y selección de servicios en tiempo de ejecución [11].

Los participantes más importantes de este mecanismo son el proveedor del servicio, el consumidor del servicio y el bróker del servicio.

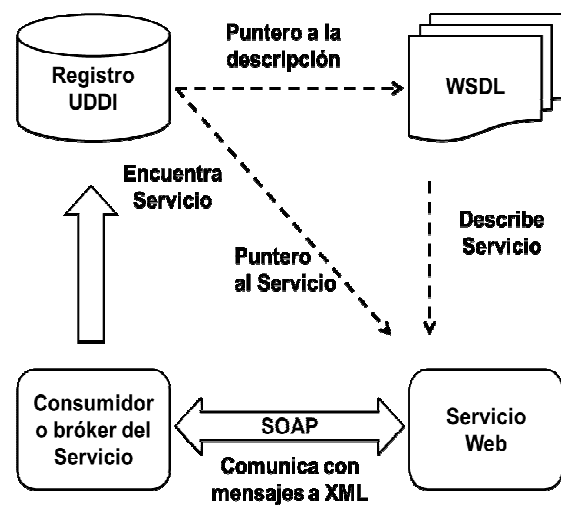


Figura 1. Modelo de Servicio Web¹

Las interacciones de estos participantes involucran la publicación del servicio por parte de proveedor, la búsqueda del servicio por parte del bróker y el enlace al servicio que realiza el consumidor. Por lo tanto, el proveedor deberá proporcionar la descripción del servicio y publicarlo para que el consumidor o el bróker puedan descubrirlo. El consumidor utiliza la

¹ Adaptación (<http://support.novell.com>)

búsqueda o contrata al bróker del servicio para descubrir el servicio que se corresponda a sus requerimientos, y a partir de la descripción del servicio, finalmente, el consumidor puede invocar los webservices o acceder a sus operaciones.

En los últimos años, este mecanismo de adquisición de operaciones a través de internet se ha vuelto de mucha importancia, especialmente por los bajos costes del paradigma de computación en la nube y por la alta disponibilidad de los servicios a través de toda la red. También, los servicios pueden interactuar y componerse con otros servicios, agregando nuevos desafíos al contexto de computación en la nube, y la necesidad creciente de monitoreo de los requerimientos dinámicos del servicio. Por lo tanto, es necesario comprender como refinar los requerimientos de alto nivel del consumidor en tiempo de ejecución y modificar los cambios dentro del modelo de software como servicio [13].

Ingeniería de Requerimiento

El área de conocimiento de ingeniería de software se dedica a la obtención, análisis, especificación y validación de los requerimientos de software [14]. Los requerimientos de software expresan las necesidades y restricciones en un producto de software o un servicio que contribuye a solucionar los problemas del mundo real.

Una alternativa de descomposición de requerimientos puede ser bajo la estructura del producto, esto es: requerimientos del sistema, requerimientos del software, los prototipos, casos de uso, escenarios, entre otros.

El área de conocimiento requerimiento de software está relacionada estrechamente al diseño de software, las pruebas de software, el mantenimiento de software, la gestión de configuración del software, el proceso de ingeniería de software y la calidad de software [14].

Wind y Schrödl [15] indican que para poder obtener los requerimientos de los servicios, es necesario evaluar el modelo de desarrollo de los servicios, y afirman que ninguno de

los modelos existentes (*V Modelo, Rational Unified Process, Volere y Extreme Programing*) es indicado para cubrir las necesidades de ingeniería de requerimiento dentro del computación en la nube, por lo tanto representa un desafío presentar un enfoque para el análisis de requerimientos en este entorno.

Básicamente, un proceso de ingeniería de requerimiento tradicional abarca las siguientes actividades [16]:

1. Captura de Requerimientos: los requerimientos del servicio son descubiertos.
2. Análisis de Requerimientos: los requerimientos son analizados en detalle, se evalúan los conflictos posibles, y se negocia con los interesados su viabilidad.
3. Validación de Requerimientos: los requerimientos deben pasar por evaluaciones de consistencia y completitud.
4. Gestión de Requerimiento: abarca todo el proceso y resuelve los problemas que pueden presentarse.

Wieger en su trabajo [17] clasifica los elementos que se obtienen durante el proceso de captura de requerimientos en los siguientes artefactos: requerimientos del negocio, escenario o casos de usos, reglas de negocio, requerimientos funcionales, atributos de calidad, requerimientos de interfaz externa, restricciones, definición de datos, ideas de soluciones. Por otro lado, Pohl [1] especifica las metas, los escenarios y requerimientos orientados a la solución, como parte de los artefactos de ingeniería de requerimientos. Es importante poder especificarlos para el análisis de los requerimientos de servicios y la negociación del SLA.

GoRE

GoRE (*“Goal-oriented Requirement Engineering”*) consiste en utilizar metas u objetivos del negocio para la captura, elaboración, estructuración, especificación, análisis, negociación, documentación y trazabilidad de requerimientos [5][18].

Estas metas son formuladas en diferentes niveles de abstracción y considerando la funcionalidad asociada a los servicios y su calidad.

En el trabajo de Van Lamsweerder [19], se analiza con mayor detalle el enfoque de sistemas diseñados utilizando metas. Se podría considerar a los objetivos de un sistema o servicio como una abstracción para la ingeniería de requerimientos.

Una meta es una afirmación prescriptiva de una intención que el sistema debe satisfacer para complacer a los interesados.

Las metas pueden tener diferente nivel de abstracción: de alto nivel que son las metas amplias establecidas con los objetivos de la estrategia relacionada al negocio o la organización y de bajo nivel que son las metas más específicas que se establecen a partir de los objetivos técnicos relacionados a las opciones del diseño.

Las ventajas que propone Van Lamsweeder [18] con respecto a este enfoque de requerimiento es que los modelos de objetivos y los requerimientos pueden ser derivados sistemáticamente a través de las metas. Estas metas proporcionan una manera de razonar los requerimientos, los gráficos de las metas permiten una trazabilidad vertical desde la estrategia de alto nivel hacia detalles técnicos de bajo nivel, permiten tener una consideración integrada de alternativas, y a través de la formalización de las metas se pueden realizar refinamientos para validación, verificación y corrección de conflictos entre requerimientos.

SoRE

oRE (“*Service-oriented Requirement Engineering*”) está enfocado en modelar, especificar, y analizar requerimientos y restricciones en servicios que se ejecutan bajo la arquitectura orientada a servicios (SOA).

Si bien SoRE posee actividades comunes con el análisis de requerimiento tradicional, las entidades identificadas dentro del proceso son diferentes. En vez de clases, SoRE se centra en servicios y composición

de servicios que pueden ser enlazados durante el modelado o en tiempo de ejecución, este último escenario es muy frecuente en entornos de computación en la nube.

Algunas actividades extras son el enlace, la composición y estructuración dinámica de los servicios, cambiando de esta manera el flujo de trabajo, las interfaces de usuarios, las políticas y los datos de configuración, según los requerimientos presentes [6].

El proceso sistemático de SoRE puede ser definido como [16]:

1. *Fase de modelado de procesos de negocios*: a través de los objetivos del negocio y los procesos del negocio se da soporte a las metas que son identificados en un modelo de alto nivel del proceso de negocio, realizando las siguientes tareas: comprensión de las metas del negocio, captura de los procesos de negocio, captura de la información estratégica.
2. *Fase Flow-Down*: esta fase se relaciona con cada proceso de negocio identificados en la fase anterior, para detectar, comprender y analizar cada actividad necesaria para ejecutar con éxito el proceso de negocio que fluye hacia abajo hasta descubrir la arquitectura de procesos de negocio.
3. *Fase de especificación de requerimiento formal*: se establecen los procesos de negocio a través de los acuerdos de nivel de servicio (SLA), capturan los requerimientos y su especificación.

Enfoque Propuesto

El enfoque propuesto da soporte a la naturaleza dinámica de los requerimientos de servicios en la nube y brinda importancia a la especificación y la gestión de cambios para que sean consistentes a lo largo de la vida de un contrato de servicio. Se requiere involucrar de forma automática a los interesados de los procesos, para minimizar los riesgos introducidos por el entorno y reducir los impactos potenciales de los

cambios realizados en las especificaciones (Fig. 2).



Figura 2. Especificación y gestión de requerimientos para aplicaciones en la nube

Durante la especificación de requerimientos, un consumidor de servicio debe analizar los objetivos que quiere conseguir con la utilización del servicio y esto está asociado con capturar los requerimientos de funcionalidad y de calidad del servicio. El consumidor o bróker del servicio deberá considerar las diferentes alternativas de servicios ofertados por los proveedores de la nube. Una vez seleccionado el servicio que más se ajusta a las metas y los requerimientos, el acuerdo comienza a regir.

El análisis de requerimientos se ha vuelto una actividad transversal en toda la historia del servicio y contratación de los servicios, debido al contexto dinámico que representa la computación en la nube.

Los requerimientos pueden obtenerse de forma inmediata, a través de las necesidades presentes en el contexto del consumidor. Sin embargo, se debe garantizar que los requerimientos dentro del servicio deben poder ser especificados, validados, verificados y trazables.

Generalmente los requerimientos de calidad de servicio generan conflictos y son difíciles de implementar.

Se propone las siguientes etapas o pasos para la formulación y gestión de los requerimientos de sistemas que hacen uso de servicios en la nube:

1. *Definición del Alcance y Metas:* provee los límites o el alcance en que serán válidas las metas y los requerimientos de la adquisición o la migración de procesos a los servicios de computación en la nube. Las metas dan soporte a todo el proceso de elaboración de requerimiento, y se mantendrán estables hasta el último paso de este proceso de ingeniería de requerimiento. Las metas de alto nivel son expresiones generales de la estrategia, las restricciones, los requerimientos funcionales y no funcionales. Las especificaciones de las metas se declaran en términos de tipo, criticidad, atributos y relaciones entre ellas.

2. *Especificación del Proceso de Negocio:* consiste en una colección de tareas interrelacionadas, las cuales son diseñadas y agrupadas según las metas del negocio. Los procesos pueden ser descompuestos en varios procesos más específicos y sirven para considerar las posibles operaciones y los recursos que serán parte del servicio de la nube. En este paso también se puede analizar el nivel de concurrencia de las operaciones, las configuraciones iniciales y los mensajes.

3. *Validación de Requerimientos de Alto Nivel:* una vez que las primeras operaciones y recursos de computación en la nube han sido especificados, se debe

evaluar los conflictos potenciales que deben ser resueltos por el consumidor antes de contactarse con los proveedores de servicios. Esta primera validación sirve para analizar coherencia, riesgos, consistencias, divergencia, e incertidumbre presentes en las metas, los procesos y las operaciones.

4. *Análisis de Servicios y Proveedores Potenciales*: se analiza el catálogo de servicios, las operaciones que cubrirían, y las alternativas posibles evaluando la concurrencia entre operaciones. Se evalúa los proveedores que podrían suplir los servicios que cubren los requerimientos de servicios.

5. *Refinamiento de los Requerimientos*: en este paso se formaliza y especifica los requerimientos que podrán ser factiblemente cubiertos por los proveedores existentes en los entornos de computación en la nube, y esto se realiza teniendo en cuenta las descripciones presentes en el catálogo de servicios de Internet.

6. *Evaluación y Negociación de los Acuerdos de Nivel de Servicios*: generalmente, los acuerdos de nivel de servicios son genéricos y especifican, a grandes rasgos, las operaciones que ejecutará el servicio y el nivel esperado. Por lo tanto, se debería recorrer los puntos tratados en los acuerdos de nivel de servicios, y en caso de existir, utilizar alguna política de negociación o ajuste de los niveles especificados en estos contratos.

7. *Documentación y Formalización de Requerimientos*: una vez que las operaciones principales han sido identificadas junto con los recursos necesarios, y se han especificado claramente en los acuerdos de los proveedores, el consumidor de servicio debería tener constancia de los requerimientos funcionales, no funcionales y de calidad, para asegurar la trazabilidad y el cumplimiento de sus metas. La constancia será la documentación de estos requerimientos utilizando algún mecanismo de formalización de requerimientos.

8. *Composición del Flujo de Ejecución*: Este paso se lleva a cabo para

asegurar que los servicios, las entradas y salidas de las operaciones, la configuración de los mensajes entre servicios, y los datos están planificados en forma coherente con los requerimientos del negocio.

9. *Verificación y Validación de Requerimientos*: evaluar los requerimientos, descubrir los errores, las inconsistencias, las contradicciones y los defectos en los requerimientos que puedan generar errores en la configuración de los servicios. Los requerimientos deben ser verificables en el flujo de ejecución de las operaciones de los servicios, y se debe asegurar que las actividades principales logran los requerimientos claves, evaluando consistencia y completitud.

10. *Gestión y Trazabilidad de Requerimientos*: la gestión se encarga de todo el proceso de ingeniería de requerimiento, considerando el análisis de cambios y la vinculación de los requerimientos con los servicios contratados. Los requerimientos deben ser considerados y administrado a lo largo de todo el acuerdo de servicio, convirtiendo este enfoque iterativo y trazable. Además, un cambio en un requerimiento implica modificar los parámetros y configuraciones con la que los componentes de proveedor se relacionan, por lo tanto es necesario llevar el control y monitoreo de estos cambios en las especificaciones de los consumidores.

Caso de Aplicación: Paquete Turístico

La meta principal de este proceso de negocio es organizar un paquete turístico completo que abarque transporte alojamiento y excursiones (*Paso 1: Definición de Alcance y Metas*).

En esta propuesta se consideró un proceso de negocio de una empresa de turismo (consumidor de servicio), donde el gestor de servicios contratado o centralizados de servicios turísticos (bróker de servicio) inicia la búsqueda de diferentes alternativas de servicios para la planificación de viajes de los clientes de la empresa de turismo. Inicialmente, un servicio deberá interactuar con los usuarios de la empresa, a través de

una interfaz web y capturar las características deseadas durante la validez del viaje. Se deberá chequear que el alojamiento y las excursiones deseadas estén disponibles durante la fecha del viaje (*Paso 2: Especificación de Procesos de Negocios* y *Paso 3: Validación de Requerimiento de Alto Nivel*).

En este ejemplo, se consideró la necesidad de un gestor de servicios que opere con diferentes proveedores de servicios, y a este se le delega la evaluación de los diferentes proveedores que podrían suplir los servicios que cubren los requerimientos especificados (*Paso 4: Análisis de Servicios y Proveedores Potenciales*), por lo tanto deberá ser capaz de comunicarse con el centro de información de ventas de pasajes de transportes (aéreos o terrestres) o empresa encargada alquiler de autos (particulares o con chofer incluido).

El gestor deberá intercambiar mensajes con los servicios de hospedajes, preferentemente un centralizador de reserva de hoteles, hostales, y otros tipos de alojamiento. Finalmente, el gestor deberá proveer de un servicio que se comunique con las empresas locales o regionales del sector donde se planifica el viaje, para poder buscar excursiones, travesías y visitas a lugares de interés de la zona.

Los proveedores de servicios poseen su propio sistema y bases de datos, que por su seguridad en la gestión de recursos sólo son accesibles por sus aplicaciones web.

Todas las especificaciones funcionales y no funcionales (*Paso 5: Refinamiento de los Requerimientos*) que se obtienen de los requerimientos de los primeros pasos, deberán estar presentes en el acuerdo de nivel de servicio (*Paso 6: Evaluación y Negociación de los Acuerdos de Nivel de Servicios*).

El gestor de servicios contratado deberá realizar la descomposición de las tareas que debe realizar los servicios web implicados y chequea el catálogo de servicios disponible para cumplir con las funciones de negocio de la empresa turística (*Paso 8: Composición del Flujo de Ejecución*) y

asegurar que la ejecución se realizará de forma consistente y cumpliendo los objetivos planteados (*Paso 9: Verificación y Validación de Requerimientos*).

Si bien, la empresa turística delega la mayoría de los pasos del enfoque propuesto al gestor de servicios contratado, dejando de lado el cobro de reservas y pago de comisiones que también son webservices, se encarga de los pasos de *Documentación y Formalización de Requerimientos (Paso 7)* y de *Gestión y Trazabilidad de Requerimientos (Paso 10)* para asegurar el cumplimiento de las metas y los requerimientos de su negocio.

Conclusiones

En el presente trabajo se propone un enfoque de ingeniería de requerimientos para servicios web en los entornos dinámicos de computación en la nube y ofrecer una guía para la adquisición de software como servicio.

Este trabajo está basado en la necesidad de lineamientos claros para el proceso de captura, validación y especificación de requerimientos, que son necesario para las especificaciones funcionales y de calidad de servicios, que generalmente se encuentran especificado en los acuerdo de nivel de servicio.

El caso de estudio permitió validar la aplicabilidad de los distintos pasos propuestos para la especificación y gestión de requerimientos.

Como trabajo futuro se propone refinar los pasos propuestos para la especificación de requerimientos en la nube proponiendo recomendaciones acerca de qué requerimientos se podrían tener en cuenta en este paradigma de negocio y la incorporación de métricas que asistan en el proceso de verificación del cumplimiento de los requerimientos por los servicios adquiridos.

Agradecimientos

Este trabajo ha sido financiado en forma conjunta por CONICET, la Universidad Tecnológica Nacional y la Agencia Nacional de Promoción Científica y Tecnológica (PAE-PICT-2007-02315).

Se agradece el apoyo brindado por estas instituciones.

Referencias

- [1] K. Pohl, "Requirements engineering: fundamentals, principles, and techniques". Springer Publishing Company, Incorporated, 2010.
- [2] J. Repschlaeger, R. Zarnekow, S. Wind, and T. Klaus, "Cloud Requirement Framework: Requirements and Evaluation Criteria to adopt Cloud Solutions". 2012.
- [3] E. M. Hanna, N. Mohamed, and J. Al-Jaroodi, "The Cloud: Requirements for a Better Service". In Cluster, Cloud and Grid Computing (CCGrid), 2012 12th IEEE/ACM International Symposium on (pp. 787-792), 2012.
- [4] M. J. Escalona, and N. Koch, "Requirements engineering for web applications-a comparative study". J. Web Eng., 2(3) (pp. 193-212), 2004.
- [5] S. Zardari, and R. Bahsoon, "Cloud adoption: a goal-oriented requirements engineering approach". In Proceedings of the 2nd International Workshop on Software Engineering for Cloud Computing (pp. 29-35). ACM, 2011.
- [6] W. T. Tsai, Z. Jin, P. Wang, and B. Wu, "Requirement engineering in service-oriented system engineering". In e-Business Engineering, 2007. ICEBE 2007. IEEE International Conference on (pp. 661-668). IEEE, 2007.
- [7] L. Bass, P. Clements, and R. Kazman, "Software architecture in practice". Addison-Wesley Professional, 2012.
- [8] P. Mell, and T. Grance, "The NIST Definition of Cloud Computing". National Institute of Standards and Technology. NIST Special Publication 800-145, 2011.
- [9] S. Khaddaj, "Cloud Computing: Service Provisioning and User Requirements". In Distributed Computing and Applications to Business, Engineering & Science (DCABES), 2012 11th International Symposium on (pp. 191-195). IEEE, 2012.
- [10] R. Schmidt, "Augmenting Cloud Requirements Engineering with Meta-Services". In Computer Software and Applications Conference Workshops (COMPSACW), 2011 IEEE 35th Annual (pp. 488-493). IEEE, 2011.
- [11] M. P. Papazoglou, "Service-oriented computing: Concepts, characteristics and directions". In Web Information Systems Engineering, 2003. WISE 2003. Proceedings of the Fourth International Conference on (pp. 3-12). IEEE, 2003.
- [12] P. Bianco, G. A. Lewis, and P. Merson, "Service Level Agreements in Service-Oriented Architecture Environments". Technical Report CMU/SEI-2008-TN-021. Carnegie Mellon, 2008.
- [13] Z. Ma, L. Liu, H. Yang, and J. Mylopoulos, "Adaptive Service Composition Based on Runtime Requirements Monitoring". In Web Services (ICWS), 2011 IEEE International Conference on (pp. 339-346). IEEE, 2011.
- [14] A. Abran, J. Moore, P. Bourque, R.L. Dupuis, and L. Tripp, "SWEBOK: Guide to the software engineering Body of Knowledge". IEEE, 2004.
- [15] S. Wind, and H. Schrödl, "Requirements Engineering for Cloud Computing: A Comparison Framework". In Web Information Systems Engineering – WISE 2010 Workshops, D. K. W. Chiu, L. Bellatreche, H. Sasaki, H. Leung, S.-C. Cheung, H. Hu, and J. Shao, (pp. 404–415). Springer, 2011.
- [16] F. Flores, M. Mora, F. Álvarez, L. Garza, and H. Durán, "Towards a Systematic Service-oriented Requirements Engineering Process (S-SoRE)". In ENTERprise Information Systems, J. E. Q. Varajão, M. M. Cruz-Cunha, G. D. Putnik, and A. Trigo, (pp. 111–120). Springer, 2010.
- [17] K. E. Wiegers, "Software Requirements", 2nd Ed. Redmond, Wash. Microsoft Press, 2003.
- [18] A. Van Lamsweerde, "Goal-oriented requirements engineering: A guided tour" Requirements Engineering, 2001. Proceedings. Fifth IEEE International Symposium on. IEEE, 2001.
- [19] A. Van Lamsweerde, "Requirements Engineering: From System Goals to UML Models to Software Specifications". Wiley, 2009.

Datos de Contacto:

Ana Sofía Zalazar. INGAR (UTN-CONICET) - Instituto de Desarrollo y Diseño, Avellaneda 3657, 3000, Santa Fe, Argentina.
azalazar@santafe-conicet.gov.ar

Silvio Gonnet. INGAR (UTN-CONICET) - Instituto de Desarrollo y Diseño, Avellaneda 3657, 3000, Santa Fe, Argentina. Universidad Tecnológica Nacional, Facultad Regional Santa Fe, Lavaise 610, 3000, Santa Fe, Argentina.
sgonnet@santafe-conicet.gov.ar

Horacio Leone. INGAR (UTN-CONICET) - Instituto de Desarrollo y Diseño, Avellaneda 3657, 3000, Santa Fe, Argentina. Universidad Tecnológica Nacional, Facultad Regional Santa Fe, Lavaise 610, 3000, Santa Fe, Argentina.
hleone@santafe-conicet.gov.ar