



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Especificación de requisitos para servicios cloud dirigido por valor

Trabajo Fin de Grado

Grado en Ingeniería Informática

Autor: Mario Guzmán Carretero

Tutor: César Emilio Insfrán Pelozo

2014/2015

Resumen

Uno de los principales problemas a la hora de desarrollar sistemas *cloud* es la brecha que existe entre la vista del negocio y la vista del sistema de información, ya que se suelen producir malentendidos durante la elicitación de requisitos. La utilización de un enfoque dirigido por modelos, donde los modelos iniciales son modelos de la vista del negocio, reducen esta brecha. En este trabajo proponemos un método para derivar, a partir de los modelos de proceso y de valor, los modelos necesarios para derivar la arquitectura automática o semi-automáticamente.

Palabras clave: ingeniería de requisitos, proceso de negocio, desarrollo dirigido por modelos, cloud computing.

Abstract

One of the main problems in cloud system development is the gap between the business and the information system view. That gap brings misconceptions that usually happen during requirements elicitation. Using a model-driven approach, where the initial models are part of business view, reduce that gap. We propose a method for deriving automatically the system architecture, using as onset process and value models.

Keywords: requirements engineering, business process, model-driven development, cloud computing.



Tabla de contenido

1.	Introducción.....	7
1.1.	Objetivo.....	7
1.2.	Contexto	8
1.3.	Alcance.....	9
1.4.	Estructura del trabajo	10
2.	Trabajo previo	11
2.1.	Procesos de negocio	11
2.2.	Modelos de requisitos basados en escenarios.....	11
2.3.	Metodologías para desarrollo orientado a servicios	12
2.4.	Modelado del valor	13
3.	Método para la especificación de requisitos de servicios <i>cloud</i>	14
3.1.	Introducción y conceptos fundamentales	14
3.2.	Vista del negocio.....	17
3.2.1	Modelo de proceso de negocio	18
3.2.2	Modelo del valor	20
3.3.	Vista del sistema de información	22
3.3.1	Modelo de casos de uso.....	23
3.3.2	Modelo casos de uso extendido.....	24
3.3.3	Modelo de arquitectura de servicios.....	25
3.3.4	Modelo de proceso de servicio	26
4.	Caso de estudio.....	28
4.1.	Definición.....	28
4.2.	Aplicación del método	28
4.2.1	Modelando la vista del negocio.....	28
4.2.2	Modelando la vista del sistema de información	30
4.3.	Resultado.....	35



5.	Conclusiones y trabajo futuro.....	36
5.1.	Conclusiones alcanzadas con este trabajo	36
5.2.	Trabajo futuro.....	36
6.	Bibliografía.....	38

1. Introducción

Hoy en día, las organizaciones se enfrentan a la presión de crear y prestar servicios innovadores que satisfagan las necesidades de clientes internos y externos, y por encima de todo, ayuden a la empresa a impulsar una ventaja competitiva en el mercado. Sin embargo, la mayoría de las aproximaciones para el desarrollo de servicios en el *cloud* no están preparadas para ser ágiles y flexibles por lo que el proceso de desarrollo y prestación de nuevos servicios no está lo suficientemente adaptado al medio empresarial como debería.

Debido a la poca integración de los procesos de negocio y las actividades de desarrollo la forma en que los servicios *cloud* son definidos, modelados, diseñados, implementados y monitorizados está fundamentalmente guiada por decisiones técnicas antes que por el valor que aporta a las organizaciones. El resultado puede ser que pasen semanas o incluso meses antes que un determinado servicio, que representa un proceso crítico para la organización, llegue a su fase de producción. Además, cuando no se dispone de mecanismos necesarios para medir el índice de consumo y de valor proporcionado por los servicios *cloud*, no se puede prever con precisión los aspectos prioritarios, la demanda requerida, ni aprovisionar en consecuencia.

Cloud computing ofrece un nuevo modelo dinámico y flexible de prestación de servicios que enfatiza la necesidad de responder con agilidad y rapidez ante las cambiantes demandas empresariales aunque como todo cambio tecnológico implica nuevos niveles de complejidad que también deben abordarse. En este sentido, los procesos de desarrollo de software deben adaptarse a este nuevo modelo de prestación de servicios para dar cuenta de dicha flexibilidad respondiendo a las necesidades cambiantes de las organizaciones y comprendiendo el valor que se debe brindar a las organizaciones de una forma ágil e incremental garantizando un nivel apropiado de calidad y coste.

1.1. Objetivo

El objetivo de este Trabajo Fin de Grado consiste el análisis y estudio de diferentes propuestas de ingeniería de requisitos que den soporte al diseño de servicios *cloud*. Con ello se quiere llegar a la definición de modelos de requisitos para servicios *cloud* con trazabilidad al valor proporcionado por los objetivos y los procesos de negocio, y usar esta información para dirigir el análisis y la priorización de los servicios *cloud* organizados por incrementos.

Para la especificación de servicios *cloud* es necesario modelar los escenarios de uso del proceso de negocio que se desea modelar. Los modelos basados en escenarios [1] pueden ser usados para este propósito, ya que son comúnmente utilizados por la industria para la representación de la interacción entre los usuarios y un sistema software. Estos modelos pueden ser el punto de partida para la derivación de modelos de arquitectura para servicios *cloud*. Para llevar a cabo esta derivación, adoptaremos técnicas de Desarrollo de Software Dirigido por Modelos [2], aprovechándose las múltiples ventajas que este enfoque de desarrollo de software lleva implícitas.

La ingeniería de requisitos para servicios en el *cloud* difiere de la ingeniería de requisitos tradicional en el sentido que aparecen nuevos *stakeholders* y necesidades específicas derivadas de la flexibilidad, escalabilidad, elasticidad y reconfiguración dinámica necesarias en servicios *cloud*. Además, en este trabajo se pretende dar soporte al desarrollo incremental dirigido por valor ya que las aproximaciones de desarrollo ágil son las más utilizadas para el desarrollo de servicios *cloud*. Con este enfoque se favorece la competitividad de las organizaciones que lo adopten, pudiendo desarrollar nuevos servicios demandados por la sociedad, o migrando funcionalidades de sistemas legados a una nueva plataforma *cloud*, en un período razonablemente corto de tiempo.

1.2. Contexto

El presente Trabajo Fin de Grado se enmarca dentro del proyecto “*Desarrollo Incremental de Servicios Cloud Dirigido por Modelos y Orientado al Valor del Cliente*” (Value@Cloud), un proyecto del Ministerio de Economía y Competitividad en el Programa Estatal de I+D+i Orientada a los Retos de la Sociedad. El proyecto está dirigido por la Dra. Silvia Mara Abrahao Gonzales.



Figura 1: logo del proyecto Value@Cloud

Value@Cloud describe una aproximación para abordar el reto Economía y Sociedad Digital. Más concretamente, la prioridad relacionada al desarrollo, innovación y adopción de soluciones y tecnologías ligadas a *cloud computing*.

Dentro del proyecto, este trabajo se encuentra situado dentro del *Paquete de trabajo 2: Definir modelos de requisitos para servicios cloud*. En todo momento se ha tenido que tener en cuenta dónde se enmarcaba el trabajo a realizar, ya que se partía de la base que la entrada del paquete de trabajo 2 era el resultado del *Paquete de trabajo 1: Definir un marco de gestión para la obtención de valor por medio de la mejora incremental de los procesos críticos de negocio*, y el resultado del trabajo del paquete 2 debía ser la entrada para el *Paquete de trabajo 3: Definir una arquitectura de referencia para el despliegue de servicios cloud*. Por lo tanto, se partía con la dependencia de lo que se desarrollase en una fase anterior del proceso, cuyo trabajo no necesariamente se tendría previamente a este trabajo. Así mismo, se debería tener en todo momento presente que nuestro trabajo debería derivar en una arquitectura *cloud* con ciertas necesidades específicas como son la flexibilidad, la escalabilidad, la elasticidad y la reconfiguración dinámica.

1.3. Alcance

Dado que el trabajo necesario para llevar a cabo el paquete de trabajo en el que está enmarcado el presente trabajo requiere un esfuerzo muy superior a lo que se espera de un Trabajo Fin de Grado, se ha acotado el trabajo a realizar a las siguientes tareas:

- a. Estudio y análisis de propuestas de ingeniería de requisitos para dar soporte al diseño de servicios y lenguajes de representación de requisitos basados en escenarios
- b. Estudio y análisis de los mecanismos de priorización y determinación del valor para la gestión de incrementos, a nivel de requisitos, para servicios *cloud*
- c. Definición una propuesta de método para la especificación de requisitos de servicios *cloud* que tenga en cuenta la priorización de los procesos de negocio que aporten valor al cliente y que gestione los incrementos
- d. Validación la propuesta por medio de la realización de un caso de estudio

El enfoque que se ha seguido para el estudio y análisis de documentación, así como el método para encontrar ésta, no ha seguido un procedimiento científico estricto, aunque se ha intentado hacer de la forma más rigurosa y metódica posible con los escasos conocimientos de métodos de investigación que se disponen en el nivel de estudios en que se enmarca el presente trabajo.

Se deja para futuros estudios abordar y ampliar, mediante métodos más adecuados y científicos, el trabajo desarrollado, con el fin de generar un conocimiento más amplio y útil sobre el tema abordado.



Por último, hay que destacar que el trabajo se centra en todo momento en requisitos funcionales, dejando los requisitos no funcionales para futuros trabajos.

1.4. Estructura del trabajo

Este trabajo consta, principalmente, de tres partes que vertebran su estructura. En primer lugar, en el punto 2 se va a hacer una exposición del trabajo previo que se ha obtenido como resultado de un estudio informal del estado del arte. Han sido abordados diferentes campos, ya que la temática del proyecto incluye campos como requisitos basados en escenarios, orientación a servicios, procesos de negocio y modelado del valor.

En segundo lugar, en el punto 3, se va a definir un método para la especificación de requisitos de servicios *cloud*, haciendo una diferenciación entre la vista del negocio y la vista del sistema de información, y explicando para cada vista los modelos necesarios a aplicar.

A continuación, en el punto 4, se va a realizar un pequeño caso de estudio en el que se aplica el modelo definido en el punto anterior, expresando el resultado obtenido.

Por último, el trabajo se cerrará con una exposición de conclusiones, trabajo futuro y bibliografía sobre la que se apoya el estudio realizado.

2. Trabajo previo

2.1. Procesos de negocio

En el contexto de los procesos de negocio en [3] se define un vocabulario para definir requisitos con un alto nivel de abstracción, separándose del habitual punto de vista de los desarrolladores y, con este vocabulario como base, se define una metodología para el desarrollo automático de aplicaciones de negocio basadas en *cloud*. Siguiendo con la línea de los procesos de negocio, se puede encontrar una propuesta para la visualización casos de uso como procesos de negocio definiéndose mediante BPMN en [4]. Este es el proceso inverso que estábamos buscando, puesto que nuestro objetivo inicial del trabajo es partir de procesos de negocio bien definidos y obtener de ellos los requisitos para, posteriormente, poder derivar la arquitectura de éstos. A pesar de tratarse del proceso inverso, se plantea la trazabilidad entre los dos modelos, de forma que es posible realizar el camino contrario sin dificultad.

En [5] se sigue la misma línea, planteándose el hecho de que los casos de uso sirven para encontrar requisitos del sistema software, mientras que los modelos de proceso de negocio ayudan a encontrar requisitos del propio proceso. Se plantea la coexistencia de ambos modelos, y se define una transformación semiautomática entre ambos modelos, definiendo la correlación entre ambos.

2.2. Modelos de requisitos basados en escenarios

Con respecto a los modelos de requisitos basados en escenarios se puede encontrar una descripción general de su uso en la ingeniería de requisitos en [1]. En esa dirección es posible leer en [6] una aproximación basada en escenarios para la elicitación de requisitos para sistemas ubicuos, donde el objetivo de esta elicitación es el descubrimiento de los requisitos funcionales del sistema ubicuo, con las particularidades que este tipo de sistema conlleva. El modelado del sistema se lleva a cabo con la técnica descrita en [4].

Al respecto del modelo de requisitos más adecuado es posible encontrarse con múltiples documentos defendiendo el modelo basado en objetivos. En [7] se exponen los pasos para la adaptación al *cloud computing* utilizando un modelo basado en objetivos. Los autores argumentan



que se trata de un paradigma prometedor, donde los objetivos son genéricos y flexibles a los requisitos de los usuarios, los cuales pueden ser refinados, elaborados, negociados y analizadas sus repercusiones económicas y el valor que aportan al cliente. Estos dos modelos de representación de requisitos son analizados en [8], donde se hace un estudio de la literatura actual para obtener una comparativa entre ambas técnicas. Más aun, se ofrecen sugerencias sobre en qué tipo de proyectos es mejor utilizar modelos basados en escenarios y modelos basados en objetivos.

2.3. Metodologías para desarrollo orientado a servicios

La orientación a servicios se nutre de servicios como elemento principal en el desarrollo de sistemas distribuidos. En las metodologías que dan soporte a esta orientación en el desarrollo de software se presentan modelos, buenas prácticas, estándares y arquitecturas de referencia que son necesarias para construir sistemas de información de una forma correcta.

En [9] se presenta SOMA, una metodología respaldada por IBM para el desarrollo de soluciones SOA. SOMA define un ciclo de vida en siete fases: modelado del proceso de negocio y transformación, gestión de la solución, identificación, especificación, realización, implementación y monitorización, y gestión del desarrollo. En cada fase, las diferentes tareas realizadas se definen, al contrario que la propuesta presentada en este trabajo, una orientación dirigida por modelos.

Otra contribución muy interesante en esta área es la propuesta que encontramos en [10], donde se define una metodología para el desarrollo de servicios de negocio orientada por servicios, donde se pone el foco en la definición de las distintas fases del ciclo de vida en el desarrollo de procesos de negocio. Estas fases son seis: planificación, análisis y diseño, construcción y pruebas, aprovisionamiento, desarrollo, y ejecución y monitorización. Al contrario que nuestra propuesta, se centra en las fases del ciclo de vida, dejando las decisiones de modelado en manos de analistas y desarrolladores de servicios.

Finalmente encontramos la metodología en la que se ha basado este trabajo, SOD-M [11]. Es la única propuesta en la que se toma un enfoque dirigido por modelos que se ha encontrado en la revisión de la literatura. Las principales diferencias de esta metodología con el método propuesto en este trabajo son: 1) en SOD-M se hace uso de modelos propuestos por los propios autores, mientras que en nuestra propuesta son todo modelos estándar con una amplia base de conocimiento y herramientas de soporte, y 2) en nuestro método el objetivo es definir el modelos de requisitos, entendido como el conjunto de modelos con los conceptos clave necesarios para llevar a cabo el proceso de negocio, mientras que SOD-M cubre todo el proceso de desarrollo.

2.4. Modelado del valor

En [12] se define e³value, una metodología en la que se trabaja de forma similar a la de la ingeniería de requisitos, pero utilizando términos del área de los negocios, el márketing y la axiología. Toda la metodología gira alrededor del intercambio de objetos de valor entre los diferentes actores implicados en el proceso de negocio.

Los esfuerzos por encontrar otros modelos con los que poder describir el valor en la búsqueda bibliográfica que se ha llevado a cabo no han tenido éxito.

3. Método para la especificación de requisitos de servicios *cloud*

3.1. Introducción y conceptos fundamentales

En esta sección se va a definir un método mediante el cual se pueda definir un modelo de requisitos, teniendo en cuenta en todo momento que se desea desarrollar un sistema de información basado en servicios *cloud*.

El proceso que se va a definir tiene como entrada la especificación del proceso de negocio para el cual se desea desarrollar el sistema de información en la nube, y el valor que los diferentes componentes del proceso aportan a la organización interesada, permitiendo derivar de forma automática/semi-automática la arquitectura del sistema de información.

Se van a presentar los diferentes modelos que van a intervenir en el proceso de derivación de la arquitectura y, con este objetivo, se va a exponer, además de los modelos relacionados estrechamente con la especificación de los propios requisitos, el siguiente paso de derivación y transformación de la arquitectura resultante que se tiene como objetivo. Se ha tomado esta decisión por la estrecha relación que hay entre los requisitos y la propia arquitectura siguiendo el paradigma de desarrollo dirigido por modelos que se plantea, ya que se debe establecer una correspondencia entre los diferentes metamodelos que intervienen en el proceso.

Respecto a los modelos que van a intervenir podemos hacer una clasificación en función de la cercanía a la plataforma en la que se va a llevar a cabo la implementación. Se pueden distinguir tres tipos de modelos, según los principios de la ingeniería del software dirigida por modelos [13]:

- Modelos independientes de la computación (*Computation Independent Models*, CIM): hacen referencia a modelos de negocio o de dominio, ya que hacen uso de un vocabulario muy próximo y familiar para los expertos en el dominio del problema. Con estos modelos se presenta lo que se espera del sistema, pero se deja a un lado todos los aspectos tecnológicos, desentendiéndose en todo momento de cómo será el sistema.

La principal ventaja que aporta este tipo de modelos es el salto en la brecha que normalmente existe entre los expertos en el dominio del problema y los expertos en la tecnología que se utilizará para dar solución a las necesidades

- Modelos independientes de la plataforma (*Platform Independent Models*, PIM): este tipo de modelos tiene la suficiente independencia del sistema como para poder ser implementados en diferentes plataformas, pero ya se refieren al sistema de información en sí.
- Modelos específicos de plataforma (*Platform Specific Models*, PSM): estos modelos, por último, son los más cercanos a la plataforma, lo que se refleja en el hecho de que contienen datos específicos para la plataforma para la que se va a implementar la solución, haciendo estos rasgos específicos que sea imposible utilizar estos modelos para plataformas que no sean la que se ha tenido en mente a la hora de desarrollarlos.

En este trabajo se va a hacer uso principalmente de modelos CIM y PIM, ya que el objetivo es definir modelos de requisitos para aplicaciones *cloud* dirigidos por valor. Quedaría propuesto para una siguiente fase del proceso de desarrollo el refinamiento de los modelos PIM para transformarlos en PSM, y a partir de ahí poder derivarlos en código de forma automática mediante transformación de modelos, pero este paso queda fuera del alcance y los objetivos del presente trabajo

Vamos a hacer una adaptación del método para el desarrollo orientado a servicios (*Service-Oriented Development Method, SOD-M*) [11]. En este método para el desarrollo de software se definen una serie de conceptos que todos los sistemas software *cloud* deben contemplar:

Concepto	Vista del negocio	Vista del sistema de información
Proceso de negocio (<i>business process</i>)	Conjunto de tareas relacionadas que se realizan para alcanzar unos resultados deseados	
Consumidor de servicios (<i>end consumer</i>)	Entidad que necesita y consume servicios de negocio. Son aquellas entidades que pagan (con dinero o cualquier otro objeto de valor) para obtener un servicio que es utilizado	Usuario del sistema de información que consume servicios provistos por éste
Servicio de negocio (<i>business service</i>)	Resultado de un proceso de negocio (o parte de él) ofreciendo valor al consumidor, creado porque la interacción consumidor/proveedor cubre alguna necesidad del consumidor de servicios	Funcionalidad compleja ofrecida por el sistema de información para cubrir una necesidad específica del consumidor de servicios



Especificación de requisitos para servicios cloud dirigido por valor

Tarea de negocio (<i>business task</i>)	Función del negocio realizada por un colaborador como parte de un proceso de negocio	Funcionalidad del negocio efectuada por el sistema para realizar una parte del servicio de negocio
Colaborador de negocio (<i>business collaborator</i>)	Entidad que colabora en el proceso de negocio, realizando ciertas tareas necesarias para realizar un servicio de negocio	Entidad responsable de la ejecución de algunas de las acciones necesarias para llevar a cabo el servicio de negocio (puede ser una entidad interna o externa)
Caso de uso básico (<i>basic use case</i>)		Conjunto de acciones efectuadas por el sistema para realizar un servicio de negocio
Caso de uso compuesto (<i>composite use case</i>)		Conjunto de acciones efectuadas por el sistema con el fin de efectuar parte del servicio de negocio. Puede ser dividido en diferentes casos de uso, básicos o compuestos
Proceso de composición de servicios (<i>service composition process</i>)		Conjunto de tareas necesarias para realizar un servicio de negocio
Actividad de servicio (<i>service activity</i>)		Conjunto de acciones que forman parte de la ejecución de un servicio de negocio
Acción (<i>action</i>)		Unidad fundamental de comportamiento del sistema, donde se describe una transformación o procesamiento en el sistema

Tabla 1: Conceptos desarrollo orientado por servicios

A continuación se muestra un diagrama de dominio mostrando como los conceptos fundamentales de SOD-M están relacionados:

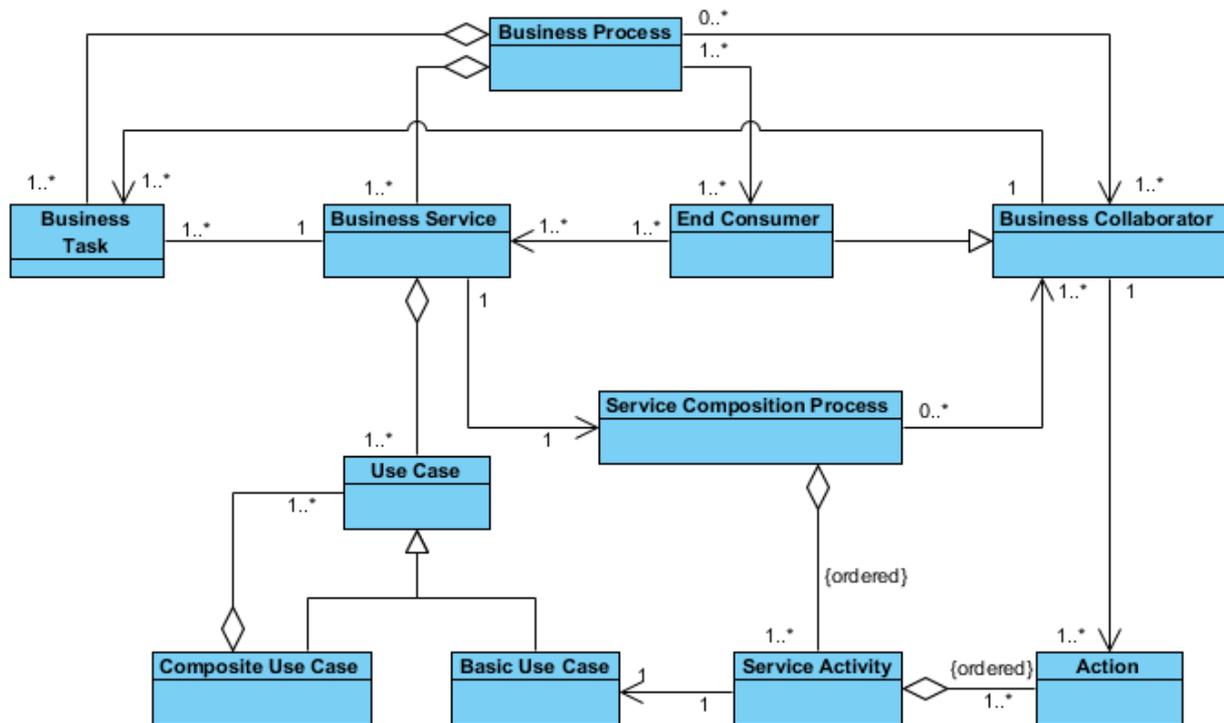


Figura 2: Conceptos fundamentales SOD-M

3.2. Vista del negocio

Desde este punto de vista del negocio se tienen dos aspectos fundamentales que deben ser tenidos en cuenta a la hora de abordar la problemática de la especificación de requisitos para servicios *cloud* dirigido por valor: el proceso de negocio y el valor que éste aporta. Alineados a estos dos aspectos de la vista del negocio se encuentra un modelo para cada uno de ellos: el modelo de proceso de negocio (Business Process Model, BPM) y el modelo de valor (Value Model, VM). Ambos modelos se centran en describir el entorno en el cual el sistema de información resultante del proceso de desarrollo será implantado, siempre sin ofrecer detalles de cómo este sistema será implementado.

Como se comentó en los objetivos de este trabajo, se quiere dar soporte al desarrollo incremental por la agilidad y los beneficios que esta modalidad de desarrollo aporta a las organizaciones que lo adoptan. Es muy importante tener esto en mente a la hora de entender cómo se debe especificar cada uno de los dos modelos implicados dentro de la vista del negocio, ya que para poder dividir el desarrollo del sistema de información en diferentes iteraciones será necesario hacer un particionado de estos modelos. Para este particionado se deberá tener en cuenta una serie de KPIs, que deberán ser definidos con anterioridad en una fase previa de estudio, y las diferentes



dependencias entre servicios, ya que no se puede incluir en una iteración un servicio dependiente de otro que vaya a ser incluido en una iteración posterior, porque el resultado de la iteración no sería funcional. Esto queda fuera del alcance del este trabajo, como se explicará en el apartado dedicado al trabajo futuro en el punto 5.2.

A continuación se van a exponer ambos modelos, tanto el modelo de proceso de negocio como el modelo de valor, y sus correspondientes metamodelos, así como las propiedades que deben cumplir y las propiedades de los mismos. Ambos modelos pertenecen a la categoría de los modelos CIM, ya que ponen el foco en las necesidades de las organizaciones implicadas, y no en el producto software que se tiene como objetivo desarrollar o las tecnologías involucradas en el desarrollo.

3.2.1 Modelo de proceso de negocio

El modelo de proceso de negocio va a ser utilizado para describir, en primer término, y entender el proceso de negocio, así como el entorno en el que se va a integrar el sistema de información que se tiene como objetivo del proceso completo de desarrollo. En un proceso de negocio se describen un conjunto de tareas que deben ser llevadas a término para alcanzar los objetivos definidos por el negocio y que son necesarios para el correcto funcionamiento de la organización que lleva a cabo el proceso.

Con el fin de modelar este proceso se ha adoptado la notación para el modelado de procesos de negocio (*Business Process Modeling Notation*, BPMN) [14] [15]. La principal razón para tomar esta decisión ha sido que BPMN se ha convertido en el estándar en la industria para la definición de procesos de negocio, y su utilización está muy extendida entre los profesionales implicados en este ámbito, con lo que será posible aprovechar la mayoría del conocimiento sobre definición de procesos de negocio que ya tienen las organizaciones. Además, el *Object Management Group* (OMG) respalda BPMN y se encarga de su mantenimiento y actualización.

BPMN tiene como objetivo la definición de modelos en los cuales se pueda ver, de una forma gráfica, la especificación de un proceso de negocio, de forma que se establezca un marco estándar que cubra la brecha que hay entre la definición y la implementación de los procesos de negocio.

En este primer paso del método para la especificación de requisitos se pretende construir modelos de un alto nivel de abstracción, con los cuales los analistas y arquitectos de servicios puedan identificar los servicios que deben ser ofrecidos al usuario final del sistema de información. Éstos pueden identificar servicios a partir de los diferentes subprocesos que se

definan dentro del propio proceso de negocio. Entrando más en detalle, a partir de los modelos BPMN podemos identificar casos de uso a partir del conjunto de tareas identificadas, lo cual será detallado más a fondo en el punto del método dedicado a los casos de uso y los casos de uso extendidos.

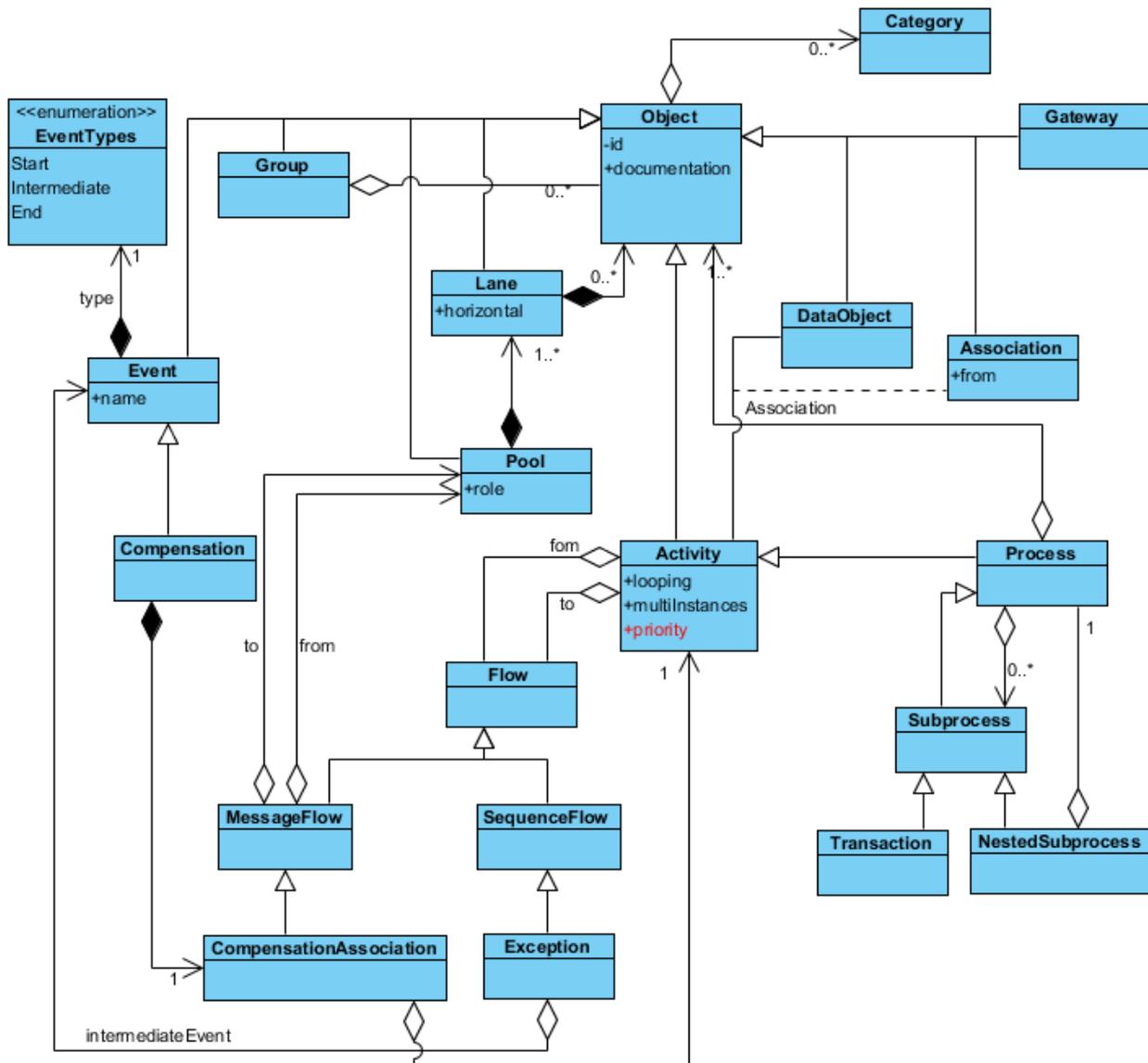


Figura 3: Metamodelo BPMN

Con el fin de simplificar el metamodelo se han omitido algunas metaclases hijas de *Gateway*, *SequenceFlow* y *Event*.

Como se ha comentado en los objetivos y alcance del presente trabajo, el enfoque que se ha tomado es desde el punto de vista incremental en el desarrollo de software, de forma que sea posible dividir los modelos en incrementos más pequeños y, mediante transformación de modelos, derivar la arquitectura. Yendo un paso más allá, el objetivo final es que sea posible desplegar los

servicios en la nube y que la propia arquitectura se reconfigure dinámicamente. Con el fin de poder hacer esta fragmentación de los modelos para su división en incrementos se ha extendido el metamodelo de BPMN añadiendo un atributo a la metaclass *Activity*. Con esto se puede asignar a cada tarea una prioridad, de forma que sea posible la ordenación entre tareas a la hora de decidir qué partes del proceso de negocio se desea desplegar en la nube.

Por otro lado se encuentra la problemática de la dependencia entre tareas. BPMN ofrece un mecanismo para saber las dependencias que tiene una tarea a través de la metaclass *Flow*. Con el conocimiento de estas dependencias hay que modificar la lista priorizada de tareas a ser desplegadas, teniendo que ponderar la prioridad de las tareas dependientes a la hora de tomar la decisión sobre qué procesos desplegar y cuáles no.

3.2.2 Modelo del valor

El modelo del valor es un modelo que representa un caso de negocio como un conjunto de intercambios y actividades que aportan valor, llevadas a cabo por actores de negocio, permitiendo entender el entorno de negocio en el cual es sistema de información actúa.

Para el modelado del valor que aportan los diferentes elementos del proceso de negocio a las organizaciones hemos elegido el método de modelado de negocio e³value [12], cuya expresividad nos ayuda a definir un conjunto de conceptos del área de la administración empresarial, el márketing y la axiología, y propone su propia notación específica para la representación del intercambio de valor. Esta notación provee de herramientas para denotar el intercambio entre los actores de elementos con un valor económico, objetos de valor, que no necesariamente tiene que ser dinero, sino que se contemplan cosas como la reputación, bienes, posibles pérdidas derivadas de problemas, así como cualquier intercambio posible entre actores que provea algún tipo de valor.

También se trabaja en este modelo con segmentos de mercado, encontrándonos con un conjunto de actores que atribuyen valor económico a una serie de elementos o actividades, siendo éstas llevadas a término por estos actores, para los cuales estas acciones les reportan un valor. Una actividad que aporta valor puede ser, por ejemplo, un servicio ofrecido por un actor.

El metamodelo que se puede apreciar en la figura 3 ofrece un conjunto de conceptos que representan valor en un modelo de valor.

Las razones que han llevado a la elección de la ontología e³value para el modelado del valor han sido:

- En la investigación previa que se ha llevado a cabo no se ha encontrado ninguna alternativa factible del nivel, calidad y difusión similar a e³value.
- Se está transformando en el estándar principal a la hora de representar el intercambio de valor en el ámbito empresarial.

E³value se desarrolló en la Universidad de Ámsterdan y su difusión dista mucho de otro tipo de modelos, como los que tienen detrás al *Object Management Group*, que cuentan con entidades internacionales y grupos empresariales detrás.

Al igual que en el caso de los modelos BPMN con los que describen los procesos de negocio, los modelos e³value tienen un alto grado de abstracción, y gracias a ellos podemos identificar conceptos como: los actores implicados en la actividad del sistema (tanto los consumidores de servicios como los colaboradores) y los servicios de negocio que debe ofrecer el sistema de información para satisfacer las necesidades de los diferentes colaboradores. Además, la necesidad de un intercambio de información identifica una tarea de negocio que debe ser contemplada.



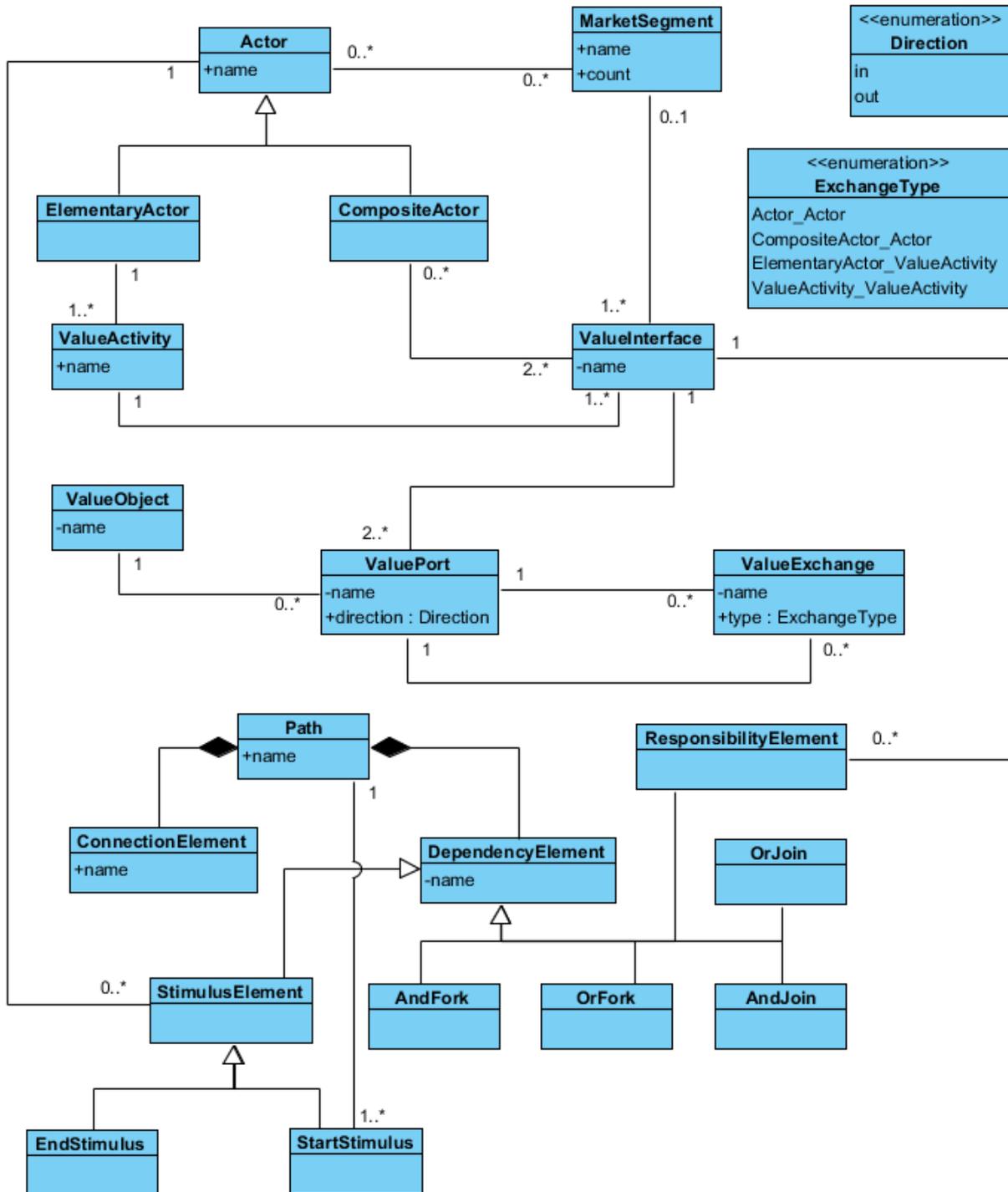


Figura 4: Metamodelo e3value

3.3. Vista del sistema de información

Los modelos que van a ser utilizados en este área, al contrario que los utilizados en la vista del proceso de negocio, están en el nivel independiente de plataforma, PIM, ya que en este

nivel el interés está puesto en los requisitos que se deben cumplir por el propio sistema de información.

El objetivo de la definición de estos modelos es la identificación de procesos de negocio que debe soportar el sistema de información, así como las funcionalidades y procesos necesarios para llevar éstos a cabo.

3.3.1 Modelo de casos de uso

Los modelos de casos de uso son los primeros modelos del método que presentamos que expresar de forma abstracta el propio sistema de información. Esto se debe a que estos modelos solo representan funcionalidades específicas del sistema, siendo modelos que representan el comportamiento del sistema de información, dejando fuera aspectos del negocio que sí que se deben haber tenido en cuenta en los modelos de valor (e³value) o en los modelos de proceso de negocio (BPMN), como podrían ser tareas manuales realizadas por operarios o toma de decisiones que tengan que ser tomadas por entes externos al sistema.

Los modelos de casos de uso forman parte del lenguaje unificado de modelado (*Unified Modeling Language*, UML [16]), el cual es el lenguaje de modelado de sistemas software más conocido y utilizado en la actualidad, tanto en el ámbito empresarial como en el académico y también está respaldado por el OMG. Adicionalmente, mediante la utilización de casos de uso se pueden modelar requisitos basados en escenarios, que es el enfoque adoptado en este trabajo y en el proyecto en el que se encuentra enmarcado.

Se propone la utilización de diagramas de caso de uso, pero con un enfoque diferente a la hora de realizar los modelos: lo que se desea es identificar servicios de negocio en lugar de las típicas funcionalidades descritas mediante los casos de uso.

Mediante la utilización de este tipo de modelos se puede llegar a identificar conceptos tan fundamentales para elicitación de requisitos para sistemas *cloud* como son el consumidor de servicios, que es representado como un actor, y los propios servicios de negocio *cloud*, que son representados como casos de uso.



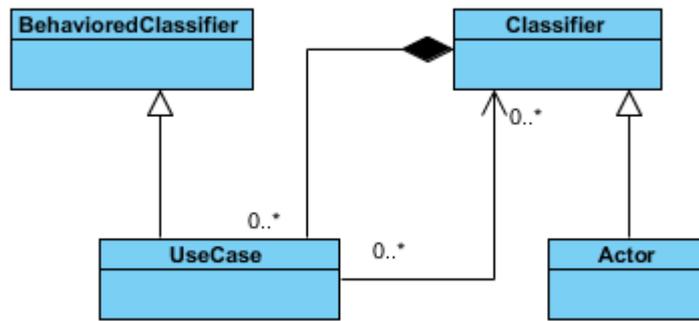


Figura 5: Metamodelo Casos de Uso

3.3.2 Modelo casos de uso extendido

Los modelos de casos de uso extendido son, al igual que los casos de uso expuestos en el punto anterior, modelos que intentan modelar el comportamiento del sistema, definiendo las funcionalidades del mismo que son necesarias para llevar a término los objetivos del sistema que tenemos como objetivo. Como es lógico siguiendo un paradigma orientado a servicios, estas funcionalidades se traducen en servicios.

Mediante la utilización de este tipo de modelos podrán ser identificados los servicios necesarios que incluye el proceso de negocio. Así mismo, también podrá ser identificado el proceso en sí, en forma de *workflow*, de forma que puedan ser identificadas las dependencias entre servicios y cómo éstos son compuestos entre sí.

En los modelos de casos de uso extendido son representados los servicios como casos de uso. Además, se pueden identificar conceptos como consumidor de los servicios, el cual será representado como un actor; así como acciones y conjuntos de acciones que el sistema debe llevar a cabo para realizar partes del proceso de negocio, que serán representados como casos de uso básicos y casos de uso compuestos, respectivamente. Éstos últimos, los casos de uso compuestos, representan un conjunto de casos de uso básico, y pueden ser representados en términos de éstos.

Este tipo de modelo hace uso de las relaciones *include* y *extend* entre los casos de uso identificados para definir la relación y dependencia entre ellos. La semántica que utiliza es la habitual en UML: “Una relación *include* especifica que un caso de uso contiene el comportamiento definido en otro caso de uso” y “una relación *extend* especifica que el comportamiento de un caso de uso base puede ampliarse opcionalmente por el comportamiento de otro caso de uso” [16].

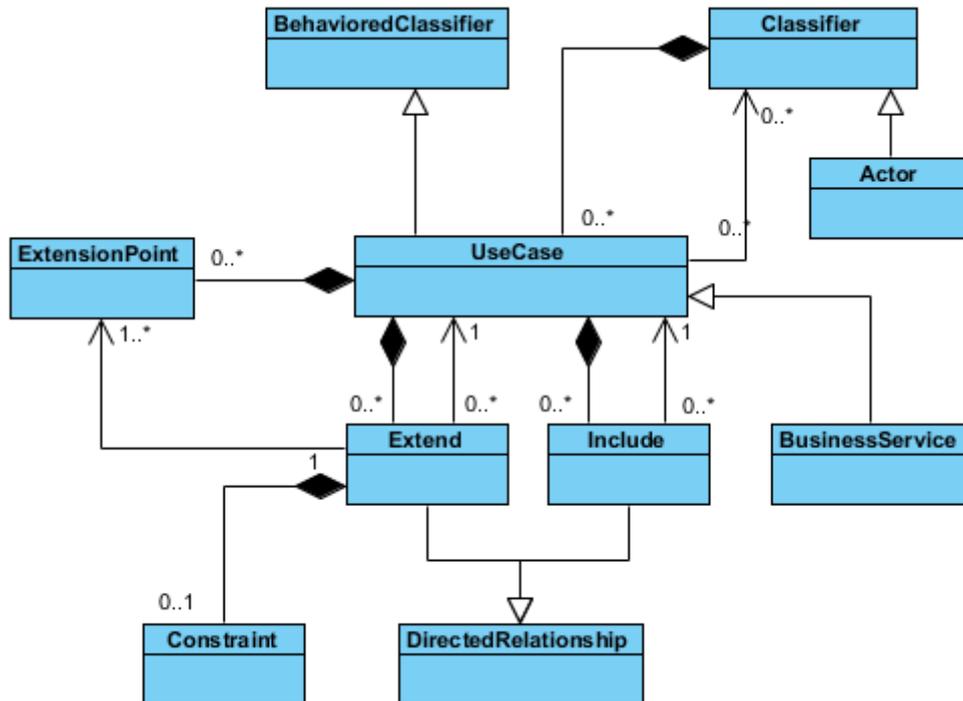


Figura 6: Metamodelo Casos de Uso extendido

3.3.3 Modelo de arquitectura de servicios

Para la definición de los modelos de arquitectura se va a hacer uso del Diagrama de Arquitectura de Servicios. Este diagrama se define dentro del lenguaje de modelado de arquitecturas orientadas a servicios (*Service-Oriented Architecture Modeling Language*, SoaML) [17].

En SoaML se describe un perfil UML y un metamodelo para el modelado y diseño de servicios enmarcados en una arquitectura orientada a servicios. Al ser parte de UML también cuenta, como el resto de modelos (a excepción del modelo e³value), con el respaldo del OMG.

La elección de este perfil de UML no ha sido arbitraria, ya que se han encontrado otras opciones viables. En el estudio previo realizado se ha encontrado que otros autores han hecho uso de otros framework para el modelado de arquitecturas orientadas a servicios. SoaML, a pesar de contar con poco recorrido desde su adopción y estandarización por el OMG, se está erigiendo en un estándar en la industria, con multitud de herramientas que le dan soporte y una gran base de conocimiento. La decisión de su utilización se sustenta en este razonamiento: el aprovechamiento de esa base de conocimiento y herramientas de soporte ya existente.

Por otro lado, y complementariamente a los criterios que se han tenido en cuenta para la elección de UML hay que destacar que en el marco del proyecto en el que se desarrolla el presente trabajo, en el tercer paquete de trabajo “*Definir una arquitectura de referencia para el despliegue de servicios cloud*” se define la arquitectura con la que se trabaja en el proyecto. Esta arquitectura está definida en términos de SoaML, así pues, se ha decidido utilizar la misma arquitectura por criterios de compatibilidad con el proceso DIARy [18], en el cual se utiliza un perfil UML en el cual se extiende la funcionalidad de SoaML [19].

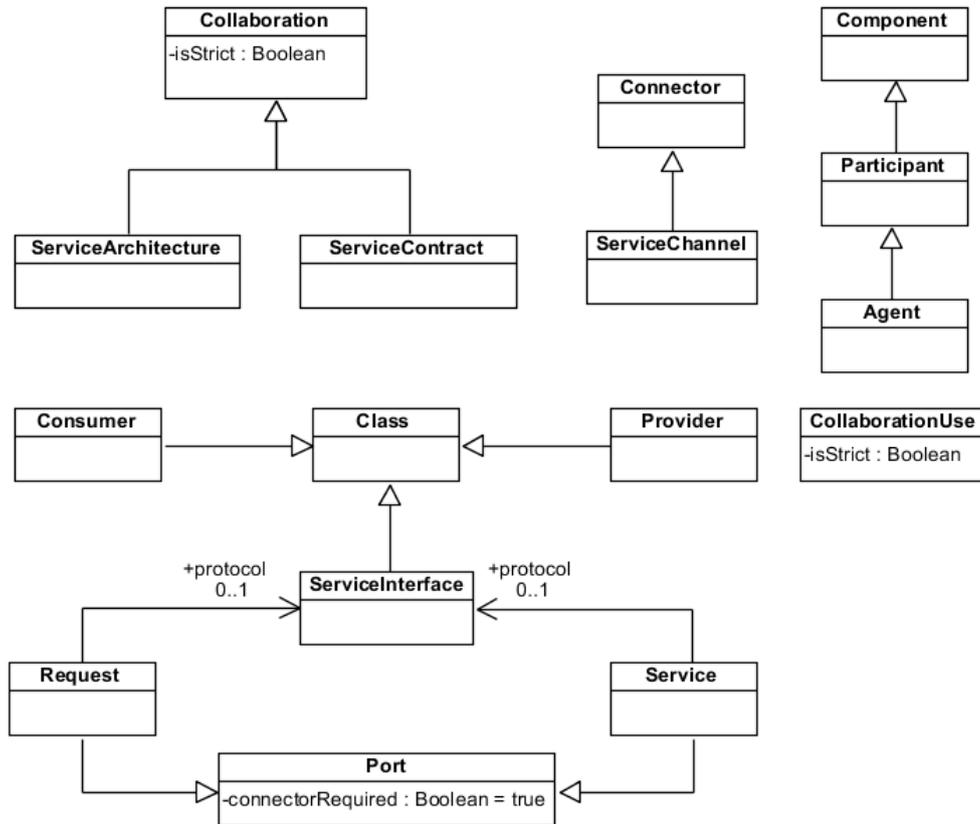


Figura 7: Metamodelo SoaML

Con este paso se pretende la identificación de las entidades que colaboran con el objetivo de llevar a buen término un proceso de negocio, que serán representadas como participantes (*Participant*). Adicionalmente, también se desea identificar las diferentes actividades que serán desarrolladas entre los diferentes participantes, que se representarán como contratos de servicio (*ServiceContract*).

3.3.4 Modelo de proceso de servicio

El modelo de proceso de servicio se define como un conjunto de actividades relacionadas que deben ser llevadas a término para realizar un servicio de negocio. Es por esto que, las

actividades en este modelo representan las acciones incluidas en el flujo de trabajo que son necesarias para desempeñar un servicio de negocio.

El método que se ha seleccionado para modelar los procesos de servicio es la utilización de diagramas de secuencia para definir la interacción entre el consumidor y el prestador del servicio. Mediante la utilización de estos diagramas se puede identificar un concepto clave en el desarrollo orientado por servicios: las acciones de servicios, las cuales son representadas por cada uno de mensajes que intercambian los diferentes actores en el diagrama de secuencia, los cuales representan a los colaboradores de negocio.

Este paso del método sirve para especificar el funcionamiento interno de cada uno de los servicios de contrato (*ServiceContract*) definidos en el modelo de arquitectura de servicio en el paso anterior, de forma que al finalizar el proceso se tenga la arquitectura completa de la aplicación. Con esto no se quiere decir que se tenga el resultado final pues, como se expresa en los objetivos de este trabajo, uno de los objetivos es la definición del presente método para la definición de requisitos *cloud*. Por esto, hay que tomar la arquitectura resultante, y el resto de modelos anteriores, como modelos de requisitos, los cuales pueden ser refinados y mejorados en sucesivas iteraciones.



4. Caso de estudio

Con el fin de validar la propuesta de método para la especificación de requisitos para servicios *cloud* se va a plantear un sencillo caso de estudio para ilustrar su funcionamiento. En primer lugar se va a proceder a la descripción del mismo, a continuación se va a aplicar el método sobre el caso de estudio descrito, y para finalizar se va a explicar el resultado obtenido.

4.1. Definición

El caso de estudio que se desarrolla en el presente trabajo consiste en una tienda online la cual ofrece sus productos a sus clientes, tiene acuerdos con una empresa de mensajería para hacer llegar a sus clientes los productos que éstos compran y con una entidad financiera para la gestión de los pagos.

En primer lugar el cliente selecciona los artículos que desea comprar y, cuando desea finalizar la compra paga los productos. Una entidad bancaria externa es la encargada de gestionar el pago, y una vez recibido el pago avisa de esto a tienda online, que procede a solicitar a la empresa de mensajería el envío de los productos y, una vez enviados, la tienda online se desentiende del envío.

4.2. Aplicación del método

4.2.1 Modelando la vista del negocio

4.2.1.1 Modelado del proceso de negocio

El primer paso en el modelado de la vista del negocio es el modelado del propio proceso de negocio mediante BPMN.

En el modelado del caso de estudio han sido identificados cuatro carriles, uno por cada uno de los actores que están implicados en el proceso de negocio, así como dos subprocesos, el proceso de compra y el proceso de envío. Se pueden apreciar las diferentes tareas y los colaboradores del negocio: la entidad financiera y la empresa de mensajería, los cuales se han

representado como un carril. Por último se ha modelado el consumidor de servicios como el cliente, también en un carril independiente.

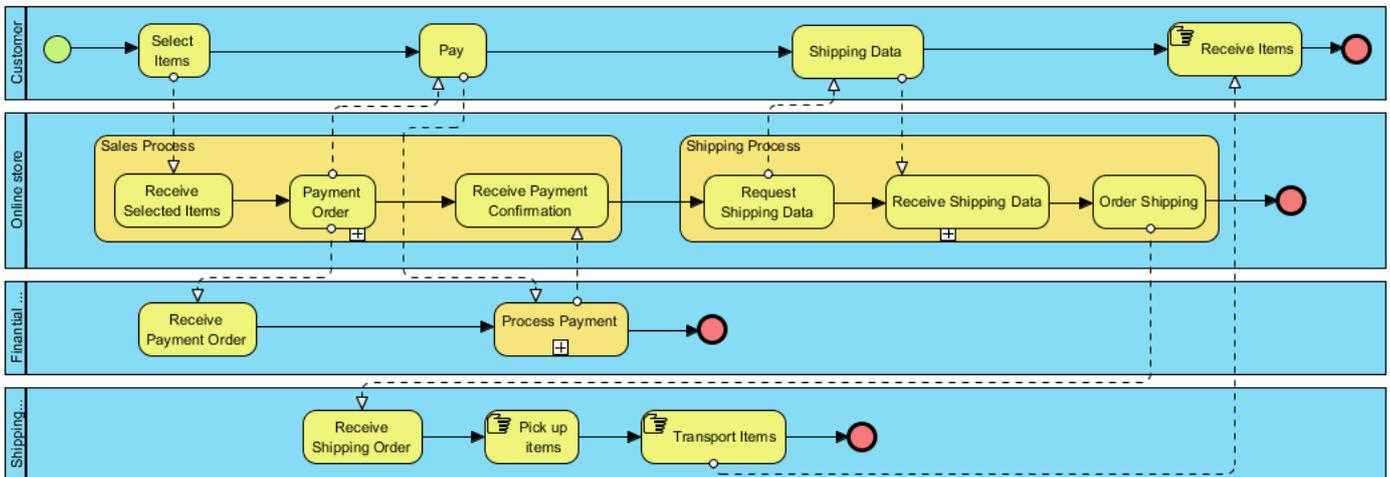


Figura 8: Modelo de proceso de negocio para la tienda online del caso de estudio

Podemos apreciar los siguientes conceptos en el diagrama BPMN:

- Servicio de negocio: los diferentes subprocesos que encontramos en el sistema, en este caso el proceso de compra y el proceso de envío. Los subprocesos de otras entidades no se deben tener en cuenta al ser servicios externos que no deben ser implementados.
- Tarea de negocio: las diferentes tareas que encontramos en el carril referente al sistema objetivo, en este caso recibir los artículos seleccionados, recibir confirmación del pago y el resto de tareas.
- Consumidor de servicios: el carril que actúa como consumidor de los servicios, en este caso el cliente.
- Colaborador de negocio: los carriles que actúan como prestadores de servicios, en este caso la entidad financiera y la empresa de mensajería.

Como se puede apreciar, sólo se tienen en cuenta aquellas tareas del carril del sistema que se va a desarrollar, pero se puede ver la necesidad de interacción entre diferentes sistemas, hecho que deberá tenerse en cuenta a la hora de hacer compatible esa interacción.

4.2.1.2 Modelado del valor

Se ha utilizado el método definido en e³value para explorar el modelo de valor de la tienda online, representando los actores que intercambian objetos de valor con otros actores. En la figura

9 se puede apreciar el modelo de valor que describe el intercambio de valor en el caso de estudio presentado.

La tienda online se muestra como un actor elemental, mientras que el cliente, la entidad financiera y la compañía de mensajería se han modelado como segmentos de mercado. Los servicios que ofrece la tienda online han sido especificados como actividades de valor.

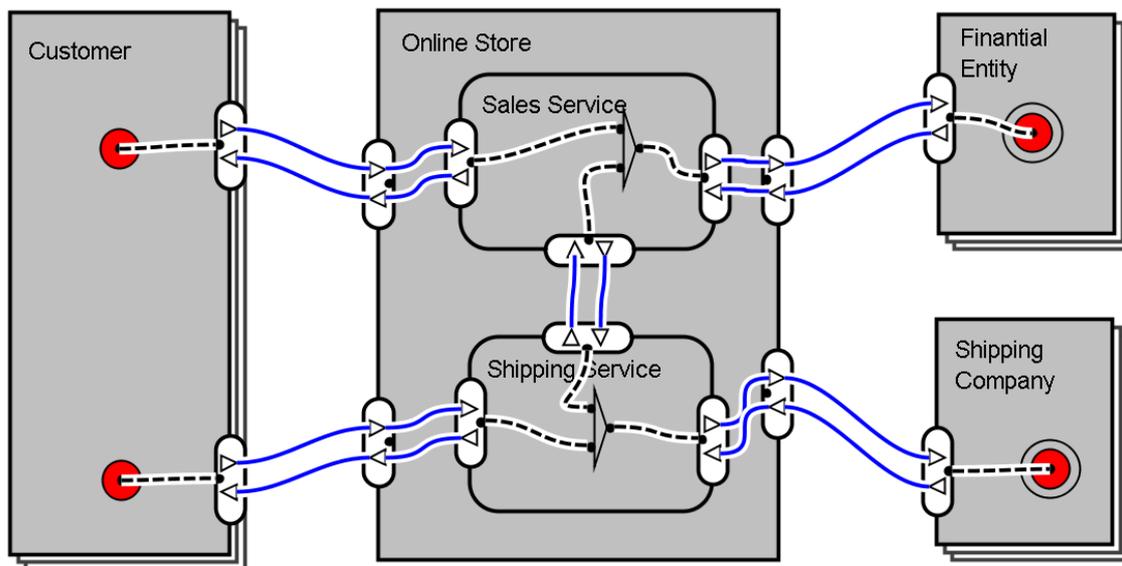


Figura 9: Modelo de valor para la tienda online del caso de estudio

Podemos apreciar los siguientes conceptos en el diagrama e³value:

- Servicio de negocio: las necesidades de intercambio de objetos de valor por parte del cliente, en este caso el servicio de compra y el servicio de envío.
- Tarea de negocio: las necesidades de intercambio de valor entre los colaboradores del negocio, en este caso la comprobación de pago, la confirmación del mismo y la orden de envío.
- Consumidor de servicios: el segmento de mercado que actúa como cliente.
- Colaborador de negocio: los segmentos de mercado que actúan como colaboradores.

4.2.2 Modelando la vista del sistema de información

4.2.2.1 Modelando los casos de uso

Para aplicar el método propuesto el primer modelo en el cual se tiene en cuenta el sistema de información es el modelo de casos de uso. Con este modelo solamente se representan los

servicios del negocio que deben ser implementados por el sistema, quedando fuera aquellos procesos que, de alguna forma, estén fuera del sistema, ya sea porque los implementan sistemas externos o porque sean procesos manuales que no requieran de informatización.

Como se puede apreciar en la figura 10, en el caso de estudio se han detectado dos servicios, el servicio de venta y el de envío, y un actor: el cliente.

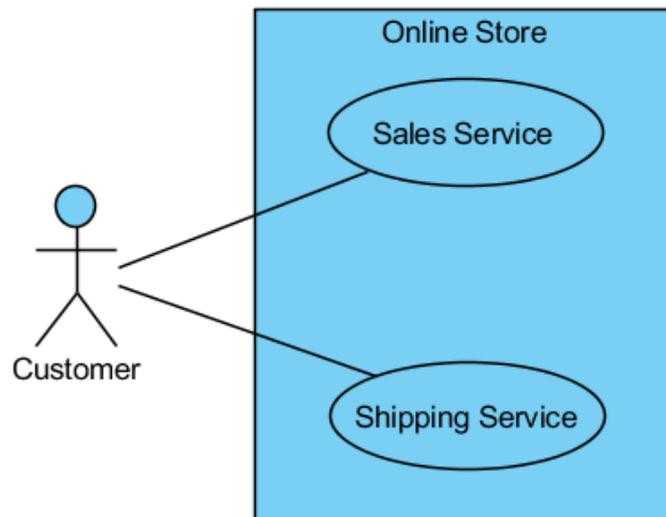


Figura 10: Modelo de casos de uso para la tienda online del caso de estudio

Podemos apreciar los siguientes conceptos en el diagrama de casos de uso:

- Servicio de negocio: los casos de uso, en este caso el servicio de compra y el servicio de envío.
- Consumidor de servicios: los actores que actúan como consumidores de los servicios del sistema, en este caso el cliente.

4.2.2.2 Modelando los casos de uso extendido

Después de identificar los servicios de negocio el método propuesto hace uso de los casos de uso extendido para modelar las funcionalidades del sistema que son necesarias para llevar a cabo dichos servicios. Este paso es de especial interés siguiendo un desarrollo orientado por servicios, ya que estas funcionalidades serán traducidas posteriormente en servicios del sistema, y es muy importante el orden en que estos servicios serán invocados para realizar el servicio de negocio en su conjunto.

En este paso los servicios de negocio son representados como un conjunto de casos de uso básicos o compuestos, y los actores permanecen del mismo modo que el en paso anterior.

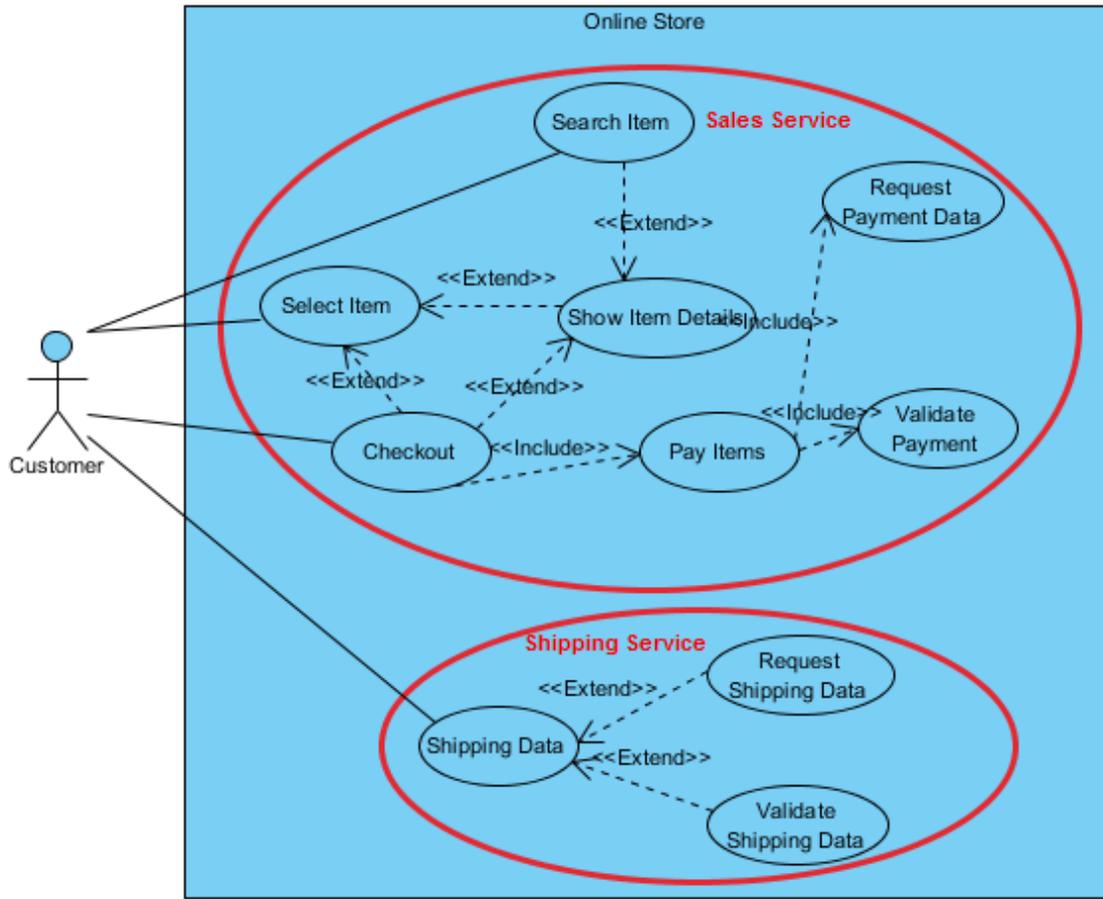


Figura 11: Modelo de casos de uso extendido para la tienda online del caso de estudio

Como se puede apreciar en la figura, en el modelo de casos de uso extendido se ha entrado en profundidad en el funcionamiento de cada uno de los servicios de negocio, en este caso el servicio de ventas y el servicio de envío.

Podemos apreciar los siguientes conceptos en el diagrama de casos de uso extendido:

- Servicio de negocio: los conjuntos de casos de uso que desempeñan un servicio común en conjunción, en este caso el servicio de compra y el servicio de envío.
- Tarea de negocio: cada uno de los casos de uso especificados en el modelo.
- Consumidor de servicios: los actores implicados, en este caso el cliente.

4.2.2.3 Modelado de la arquitectura de servicios

Para modelar la arquitectura de servicios mediante el método propuesto se sigue un enfoque *top-down*, conectando los servicios con los consumidores y proveedores del mismo, y en el siguiente paso serán especificados los propios servicios.

En este paso se va a hacer uso del Diagrama de Arquitectura de Servicios, obteniendo un diagrama en el cual se puede ver cómo estos servicios se componen entre sí para realizar el proceso de negocio.

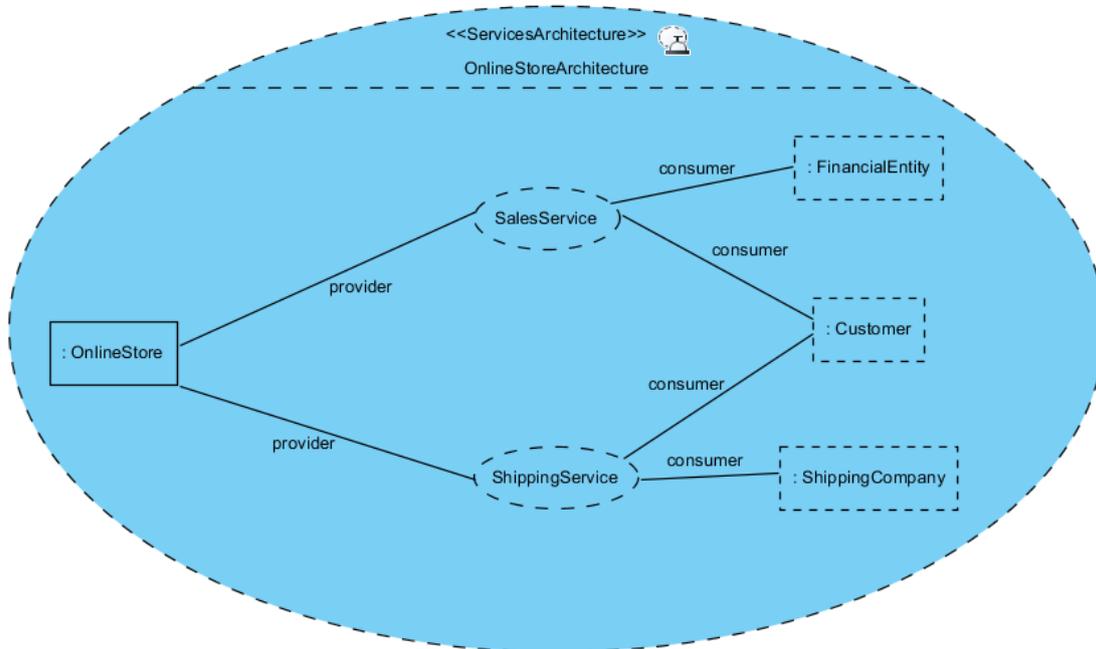


Figura 12: Modelo de arquitectura de servicios para la tienda online del caso de estudio

Se puede ver en la figura como los servicios de negocio de ventas y de envío han sido traducidos por contratos de servicio. Complementariamente, los participantes reflejan los consumidores de servicios, en este caso el cliente, y los colaboradores, en este caso la entidad financiera y la empresa de mensajería. De nuevo, hay que aplicar conocimiento del negocio a la hora de transformar este modelo para poder discriminar los colaboradores de los consumidores de servicios.

Podemos apreciar los siguientes conceptos en el diagrama de arquitectura de servicios:

- Servicio de negocio: los diferentes contratos de servicio, en este caso los servicios de ventas y de envío.
- Consumidor de servicios: los participantes que actúan como consumidor de servicios, en este caso el cliente.
- Colaborador de negocio: los participantes que actúan como colaboradores, en este caso la entidad financiera y la empresa de mensajería.

4.2.2.4 Modelado del proceso de servicios

Siguiendo la estrategia *top-down* que se comentaba en el punto anterior, en este último paso del método propuesto el objetivo es la especificación de la funcionalidad de cada uno de los servicios de negocio que han sido identificados en el paso anterior. Por el hecho de mantener la sencillez del caso de estudio sólo se va a modelar uno de los servicios de negocio, concretamente el servicio de ventas.

Este paso ayuda con la identificación de las diferentes tareas que intervienen en cada uno de los servicios de negocio, así como el orden de éstas. Este orden es de vital importancia ya que se está siguiendo una orientación a servicios y la invocación de estos servicios es muy importante para el correcto funcionamiento del sistema.

Para el modelado del proceso de servicios se van a utilizar, como explicó en la propuesta del método, diagramas de secuencia. Con ello es posible expresar la secuencia de pasos y tareas que deben ser realizados para la consecución de un servicio de negocio.

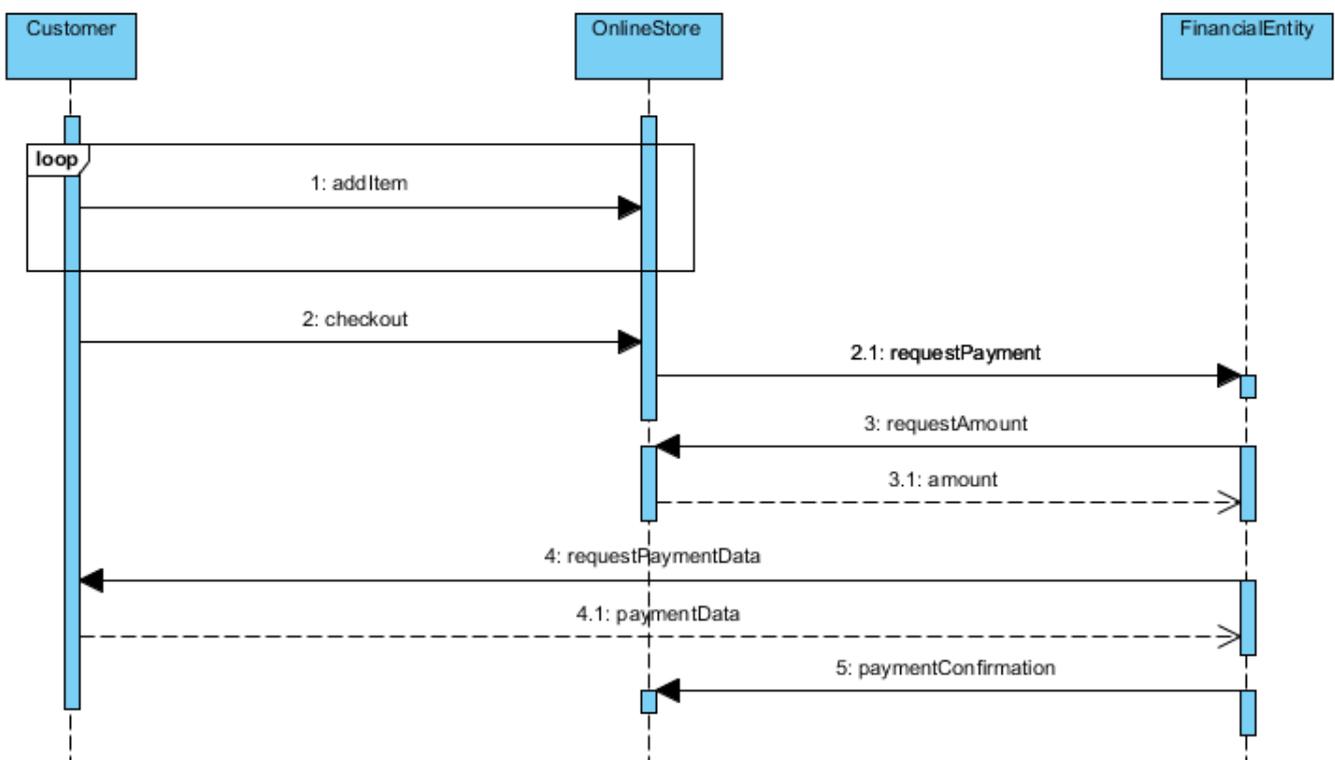


Figura 13: Modelo de proceso de servicios para la tienda online del caso de estudio

En el diagrama de la figura es posible apreciar como las diferentes tareas, expresadas en forma de mensajes entre los diversos participantes, están ordenadas temporalmente, dotando de lógica a la ejecución del servicio de negocio que es especificado.

Podemos apreciar los siguientes conceptos en el diagrama de proceso de servicios:

- Servicio de negocio: el conjunto de actividades define un servicio de negocio.
- Actividad de servicio: cada uno de los mensajes que son intercambiados entre los diferentes participantes
- Consumidor de servicios: los participantes que actúan como consumidor de servicios, en este caso el cliente.
- Colaborador de negocio: los participantes que actúan como colaboradores, en este caso la entidad financiera.

4.3. Resultado

Como resultado de este proceso se obtienen dos puntos de vista. En primer lugar se obtiene la vista del negocio, donde se ha modelado el proceso de negocio en sí, utilizando BPMN, y por otro lado el modelado del valor, el cual modela el intercambio de objetos de valor entre los participantes en el modelo de valor. Con esta vista se obtiene una visión general de todo el proceso y son identificados todos los conceptos fundamentales necesarios para abordar el desarrollo del sistema de información.

Al modelo de proceso de negocio se le aplicarán las KPIs definidos en un paso previo de estudio del negocio y será definida la prioridad de cada tarea. Esta priorización, junto a la dependencia de las tareas, resultará en incrementos a ser desarrollados.

A partir de los modelos de esta vista, y mediante transformación de modelos (ya sea de forma automática o asistida) se derivan los casos de uso y casos de uso extendido, donde se apreciarán las funcionalidades o, en este enfoque orientado a servicios, servicios que serán necesarios para llevar a cabo el proceso de negocio.

En este punto ya se dispondría de un modelo de requisitos, pero se ha querido ir un paso más allá por el contexto del trabajo, ya que se está siguiendo una orientación a servicios, y mediante el modelado de la arquitectura mediante SoaML se especifican los requisitos de los propios servicios. Además, en el siguiente paquete de trabajo del proyecto Value@Cloud son necesarios los citados incrementos de la arquitectura para ser desplegados y reconfigurados dinámicamente. De nuevo, mediante transformación de modelos la arquitectura es derivada a partir de los casos de uso, obteniéndose tanto la especificación de cada uno de los servicios de negocio esperados, así como la ordenación temporal de éstos para realizar el proceso de negocio de forma lógica.



5. Conclusiones y trabajo futuro

A continuación se van a exponer las conclusiones del presente trabajo y el posible trabajo futuro que sería la continuación lógica del mismo.

5.1. Conclusiones alcanzadas con este trabajo

En el actual panorama empresarial, donde la competitividad entre las diferentes empresas está en un punto límite, el aprovechamiento de las nuevas tecnologías es un factor diferencial para el éxito o el fracaso en el mercado global. Las actuales aproximaciones para el desarrollo de servicios en el cloud no están preparadas para el entorno cambiante en el que trabajan las empresas, así que es necesario un método ágil que permita a estas empresas competir estando a la última.

Complementariamente, el principal escollo a la hora de desarrollar sistemas basados en servicios es la brecha existente entre el mundo del negocio y el mundo del desarrollo de sistemas de información. Utilizando el método propuesto en este trabajo esta brecha se hace mucho más pequeña, ya que el punto de partida es la vista del negocio y, a partir de ésta, se derivan las funcionalidades del sistema de información, siguiendo siempre las pautas que el mismo proceso de negocio dicta.

Finalmente, la aproximación de desarrollo dirigido por modelos, en adición a todo lo anterior, sirve como refuerzo al factor de agilidad que aporta este trabajo, ya que al poder derivar las arquitecturas derivando modelos sería posible desplegar nuevos servicios o modificar los actuales simplemente modificando los modelos. Concretamente, desde el punto de vista de la ingeniería de requisitos dirigida por modelos, cualquier cambio que surgiera en los requisitos sería fácilmente reflejado en los modelos y, por ende, fácil y rápidamente desplegado en la nube.

5.2. Trabajo futuro

La magnitud y los objetivos del proyecto en el cual se enmarca este trabajo, Value@Cloud, hace que quede mucho trabajo por desarrollar en el área de los requisitos.

En primer lugar sería necesario contemplar los requisitos no funcionales, dado el especial impacto que este tipo de requisitos tiene en los entornos cloud, donde encontramos conceptos como flexibilidad, escalabilidad, elasticidad y reconfiguración dinámica, todos ellos de vital importancia para este tipo de entornos y las aplicaciones que en ellos se ejecutan.

Además, es necesario trabajar en la línea de la ayuda a la toma de decisiones a la hora de decidir qué partes del proceso de negocio tienen que desarrollarse en las diferentes iteraciones. Para ello, además de la priorización de las tareas a partir de los KPIs que determinan su valor para el negocio, hay que tener en cuenta las dependencias entre las diferentes tareas.

Por último, hay que poner el foco en la derivación de los modelos en cada uno de los pasos del método. Para que esta transformación entre modelos se pueda generar automática o semi-automáticamente, se hace necesaria la definición de estas transformaciones con lenguajes como QVT o ATL.

Este trabajo se espera llevar a cabo en el trabajo fin de máster del Máster Universitario en Ingeniería y Tecnología de Sistemas Software durante el curso académico 2015/16.

6. Bibliografía

- [1] K. Cox, «Fitting Scenarios to the Requirements Process,» de *11th International Workshop on Database and Expert Systems Applications*, London, 2000.
- [2] Object Management Group, «MDA Guide Version 1.0.1,» 12 06 2003. [En línea]. Available: http://www.omg.org/news/meetings/workshops/UML_2003_Manual/00-2_MDA_Guide_v1.0.1.pdf. [Último acceso: 27 07 2015].
- [3] H. Benfenatki, C. Ferreira Da Silva, A.-N. Benharkat y P. Ghodous, «Cloud-Based Business Applications Development Methodology,» de *WETICE Conference (WETICE), 2014 IEEE 23rd International*, Parma, 2014.
- [4] D. Lubke, K. Schneider y M. Weidlich, «Visualizing Use Case Sets as BPMN Processes,» de *Requirements Engineering Visualization (REV'08)*, Barcelona, 2008.
- [5] A. Sinha y A. Paradkar, «Use Cases to Process Specifications in Business Process Modeling Notation,» de *IEEE International Conference on Web Services (ICWS)*, Miami, 2010.
- [6] M. Alawairdhi y E. Aleisa, «A Scenario-Based Approach for Requirements Elicitation for Software Systems Complying with the Utilization of Ubiquitous Computing Technologies,» de *Computer Software and Applications Conference Workshops (COMPSACW), 2011 IEEE 35th Annual*, Munich, 2011.
- [7] S. Zardari y R. Bahsoon, «Cloud adoption: a goal-oriented requirements engineering approach,» de *2nd International Workshop on Software Engineering for Cloud Computing (SECLOUD'11)*, New York, 2011.
- [8] S. Misra, V. Kumar y U. Kumar, «Goal-oriented or scenario-based requirements engineering technique - what should a practitioner select?,» de *Canadian Conference on Electrical and Computer Engineering, 2005*, Saskatoon, Sask., 2005.

- [9] A. Arsanjani, “SOMA: A method for developing service-oriented solutions,” *IBM Systems Journal*, vol. 47, no. 3, pp. 377-396, 2008.
- [10] M. P. Papazoglou y W.-J. Papazoglou, «Business process development life cycle methodology,» *Communications of the ACM*, vol. 50, n° 10, pp. 79-85, 2007.
- [11] V. De Castro, E. Marcos y R. Wieringa., «Towards a service-oriented mda-based approach to the alignment of business processes with it systems: from the business model to a web service composition model,» *International Journal on*, n° 18, pp. 225-260, 2009.
- [12] J. Gordijn y J. Akkermans, «Value-based requirements engineering: exploring innovative e-commerce idea,» *Requirements Engineering Journal*, vol. 8, n° 4, pp. 114-134, 2003.
- [13] F. Truyen, «The Fast Guide to Model Driven,» junio 2006. [En línea]. Available: http://www.omg.org/mda/mda_files/Cephas_MDA_Fast_Guide.pdf. [Último acceso: 12 agosto 2015].
- [14] Object Management Group, «Business Process Model and Notation,» 03 enero 2011. [En línea]. Available: <http://www.omg.org/spec/BPMN/1.1/PDF>. [Último acceso: 21 julio 2015].
- [15] Berliner BPM-Offensive, «BPMN 2.0 - Business Process Model and Notation,» 23 septiembre 2014. [En línea]. Available: http://www.bpmb.de/images/BPMN2_0_Poster_ES.pdf. [Último acceso: 17 julio 2015].
- [16] Object Management Group, «OMG Unified Modeling Language TM (OMG UML),» 01 marzo 2015. [En línea]. Available: <http://www.omg.org/spec/UML/2.5/PDF/>. [Último acceso: 2013 agosto 17].
- [17] Object Management Group, “Service Oriented Architecture Modeling Language (SoaML),” 10 mayo 2010. [Online]. Available: <http://www.omg.org/spec/SoaML/1.0.1/PDF>. [Accessed 20 julio 2015].
- [18] M. Zúñiga-Prieto, S. Abrahao y E. Insfran, «An Incremental and Model Driven Approach for the Dynamic Reconfiguration of Cloud Application Architectures,» de *24th International Conference on Information Systems Development*, Harbin, China, 2015.



- [19] M. A. Zúñiga Prieto, S. Abrahao y E. Insfrán Pelozo , «Perfil UML para el Modelado de la Integración de Servicios Cloud en Procesos de Desarrollo Incremental,» de *XI Jornadas de Ciencia e Ingeniería de los Servicios (JCIS 2015)*, Santander, 2015.
- [20] A. Tariq, S. Khan y S. Iftikhar, «Requirements Engineering process for Software-as-a-Service (SaaS) cloud environment,» de *2014 International Conference on Emerging Technologies (ICET)*, Islamabad, 2014.
- [21] A. Tariq, S. Khan y S. Iftikhar, «Remapping of CMMI level-2 KPA's for development process improvement of Software-as-a-Service (SaaS) cloud environment,» de *2014 International Conference on Open Source Systems and Technologies (ICOSST)*, Lahore, 2014.
- [22] J. J. Gutiérrez, C. Nebut, M. J. Escalona, M. Mejías y I. M. Ramos, «Visualization of Use Cases through Automatically Generated Activity Diagrams,» de *MoDELS '08: Proceedings of the 11th international conference on Model Driven Engineering Languages and Systems*, Berlin, Heidelberg, 2008.
- [23] A. Fazziki, H. Lakhri, K. Yetognon y M. Sadgal, «A Service Oriented Information System: A Model Driven Approach,» de *Signal Image Technology and Internet Based Systems (SITIS), 2012 Eighth International Conference on*, Naples, 2012.
- [24] J. M. Carroll, *Making Use: Scenario-Based Design of Human-Computer Interactions*, The MIT Press, 2003.