

PEDECIBA Informática
Instituto de Computación – Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay

Reporte Técnico RT 06-17

**Metodología para desarrollo de aplicaciones
con enfoque SOA (Service Oriented
Architecture)**

Andrea Delgado

2006

Metodología para desarrollo de aplicaciones con enfoque SOA (Service Oriented Architecture)

Delgado, Andrea

ISSN 0797-6410

Reporte Técnico RT 06-17

PEDECIBA

Instituto de Computación – Facultad de Ingeniería

Universidad de la República

Montevideo, Uruguay, 2006

Metodología para desarrollo de aplicaciones con enfoque SOA (Service Oriented Architecture) [‡]

Andrea Delgado
*Universidad de la República,
Instituto de Computación,
Grupo de Ingeniería de Software,
Montevideo, Uruguay
adelgado@fing.edu.uy*

Agosto 2006

Resumen

El área de TI en las Organizaciones actuales se puede caracterizar por tener diversidad de sistemas que tienen entre sí dependencias complejas, que han ido creciendo en forma separada y heterogénea a lo largo de los años. Un desafío que se plantea es poder integrarlos para reaccionar ágilmente a los cambios en los requerimientos del negocio, principalmente en dos aspectos: los procesos de la Organización y las tecnologías disponibles. La definición y disponibilidad de estos servicios para toda la Organización es la base del enfoque SOA. El Grupo de Ingeniería de Software (Gris) del Instituto de Computación tiene un programa de construcción y prueba de modelos de proceso basado en una adaptación del RUP, para el que se definió una metodología para el desarrollo de aplicaciones con enfoque SOA.

Palabras clave: Ingeniería de Software, Metodología y Arquitectura de Software

[‡] Artículo publicado en la “XXXII Conferencia Latinoamericana de Informática (CLEI’06)”, sesión 4, artículo nro. 265, Santiago de Chile, Chile, Agosto de 2006.

1 Introducción

El programa de construcción y prueba de modelos de proceso es una de las actividades prioritarias del trabajo del grupo de Ingeniería de Software (Gris) de la Facultad de Ingeniería de la Universidad de la República, Uruguay [1]. El programa inició en el año 2000 y utiliza como banco de pruebas de los procesos al curso "Proyecto de Ingeniería de Software" [2] que se dicta en cuarto año de la carrera de Ingeniería en Computación. Entre los objetivos principales del curso se plantea que los estudiantes contrasten la teoría con su aplicación práctica en proyectos sometidos a restricciones análogas a las de la industria, para lo que, entre otras características, se cuenta con clientes externos al curso. Estos clientes son en general grupos de investigación, proyectos o áreas de la misma facultad, o empresas del medio, quienes plantean el desarrollo de sistemas de mediano porte que en general presentan desafíos tecnológicos importantes. El objetivo central del programa es el desarrollo de modelos de proceso que puedan resultar adecuados para su transferencia a la industria, identificando buenas prácticas que puedan ser incorporadas por las distintas empresas.

El programa cuenta con tres hitos principales: obtener un proceso definido, obtener un proceso validado y obtener un proceso ajustado y mejorado. Para esto se definen los procesos en forma previa al curso "Proyecto de Ingeniería de Software", realizando la prueba de estos procesos en dicho curso por parte de grupos de estudiantes que desarrollan las aplicaciones planteadas por los distintos clientes. Luego se evalúa la aplicación de los procesos en los proyectos y los resultados obtenidos, identificando mejoras a los procesos definidos, obteniendo procesos ajustados y mejorados que pueden ser puestos en práctica al siguiente año. Desde su puesta en marcha y hasta la fecha se han estudiado y adaptado varios modelos de proceso de las distintas corrientes existentes, tanto procesos "pesados" como "ágiles", siendo el representante más importante de los primeros el Rational Unified Process (RUP) [3] y de los segundos el eXtreme Programming (XP) [4]. De estos dos procesos se han realizado adaptaciones que se han puesto a prueba en distintas ediciones del curso "Proyecto de Ingeniería de Software", definiéndose como proceso base del programa la adaptación realizada del RUP que es sobre la cual se agregan otros enfoques y metodologías de desarrollo [5].

La metodología para el desarrollo de aplicaciones con enfoque SOA [6] se inscribe en este programa de prueba de procesos y fue puesta en práctica por dos grupos del curso "Proyecto de Ingeniería de Software" en el curso del año 2005 para el desarrollo de una aplicación de Help-Desk para el proyecto Link-All financiado por la Unión Europea en el cual participa el Instituto de Computación [7]. En esta aplicación de la metodología se pudo realizar una primera evaluación de las actividades y entregables propuestos [8]. En el corriente año se utilizará la metodología por parte de dos proyectos de fin de carrera de quinto año de la carrera de Ingeniería en Computación, los que desarrollarán componentes que se integrarán en un proyecto con enfoque SOA para una aplicación LIMS (Laboratory Information Management System) en el cual participan el Instituto de Computación y el Instituto Pasteur de Montevideo [9]. En estos proyectos se investigará la integración de aplicaciones móviles y workflow con interacción de usuarios en una arquitectura SOA.

2 Service Oriented Architecture (SOA)

Según [10] SOA es un estilo de Arquitectura de Software basado en la definición de servicios reutilizables, con interfaces públicas bien definidas, donde los proveedores y consumidores de servicios interactúan en forma desacoplada para realizar los procesos de negocio. Se basa en cuatro abstracciones básicas: servicios, application frontend, repositorio de servicios y bus de servicios. Un servicio consiste en una implementación que provee lógica de negocio y datos, un contrato de servicio, las restricciones para el consumidor, y una interfaz que expone físicamente la funcionalidad. Las application frontend consumen los servicios formando procesos de negocios. Un repositorio de servicios almacena los contratos de servicios y el bus de servicios interconecta las application frontend y los servicios.

Los servicios representan grupos lógicos de operaciones relacionadas con algún concepto del negocio. Por su parte, los procesos del negocio se realizan en servicios orientados a procesos que se componen de secuencias definidas de invocaciones a servicios, mediante una orquestación de los mismos en lo que se conoce como orquestación o coreografías de servicios, dependiendo de la ubicación del control del flujo del proceso. Un Business Process Management System (BPMS) es el aliado ideal para definir esta orquestación desde donde invocar los servicios necesarios para realizar el proceso establecido. El termino Business Process Management (BPM) refiere al conjunto de actividades que las organizaciones pueden realizar para optimizar o adaptar sus procesos de negocios a las nuevas necesidades organizacionales. Como estas actividades en general están soportadas por herramientas de software, el termino BPM es utilizado indistintamente para referirse tanto a las herramientas de software para soportar estos procesos como a las actividades que se realizan [8].

Los problemas de heterogeneidad, interoperabilidad y requerimientos cambiantes, son solucionados por una SOA que provee una plataforma para la construcción de aplicaciones basadas en servicios con las características de bajo acoplamiento, ubicación transparente de servicios e independencia de protocolos. Un consumidor de servicios no debe preocuparse por un servicio particular con el que comunicarse debido a que la infraestructura por debajo, el bus de servicios, puede hacer la elección en representación del consumidor. La infraestructura esconde la mayor cantidad de detalles técnicos posible al consumidor, y diferentes tecnologías de implementación como J2EE o .NET no afectan a los usuarios de una SOA. También debe ser posible sustituir una implementación de un servicio por una mejor si la hay disponible, con mejores características de calidad de servicio, según la metadata que describe al servicio. El paradigma base del enfoque es el de descubrir-ligar-invocar, como se muestra en la siguiente figura;

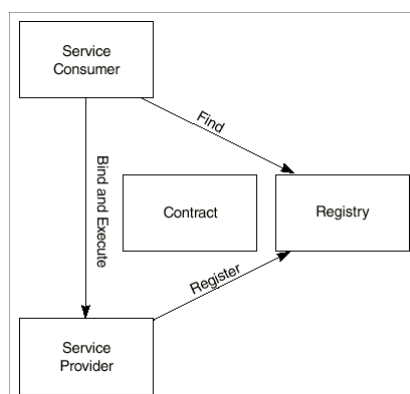


Figura 1: paradigma descubrir-ligar-invocar

El paradigma descubrir-ligar-invocar permite que el consumidor de un servicio pregunte en el registro por un servicio que cumpla su criterio de búsqueda, si en el registro se encuentra un servicio de ese tipo, el registro le envía al consumidor el contrato del servicio y un la dirección de un punto de entrada donde el servicio atiende invocaciones, luego el consumidor realiza la invocación al servicio correspondiente. El ligamiento de servicios se puede realizar tanto en forma estática (tiempo de desarrollo) como en forma dinámica (tiempo de ejecución) siendo ésta última más compleja pero la que provee la mayor flexibilidad en cuanto a la utilización de servicios según sus características.

0 Clasificación de servicios y tipos de SOA

Una primera clasificación de SOA es en dos niveles: application frontend y backend. Las applications frontend si bien no son servicios, son elementos activos que inician los procesos de negocio y reciben los resultados, los servicios del backend se clasifican [10] en: básicos, intermediarios, centrados en procesos y empresariales públicos. Comenzando por los servicios básicos, se puede decir que son el fundamento del enfoque SOA. Son proveedores

puros y no mantienen un estado conversacional. Se dividen en servicios centrados en los datos y centrados en la lógica, donde los primeros tienen como propósito manejar los datos persistentes, almacenamiento, recuperación y gestión transaccional, y los segundos proveen encapsulamiento para cálculos complejos o reglas de negocio, tradicionalmente encapsulados en bibliotecas o frameworks.

Los servicios intermediarios son servicios sin estado que hacen de puente entre las inconsistencias técnicas o discrepancias conceptuales en el diseño. Son tanto consumidores como proveedores, mediando entre los distintos elementos que deben funcionar juntos. Se dividen a su vez en *technology gateways*, *adapters*, *facades* y *functionality-adding*. Los servicios de *technology gateways* hacen de puente entre discrepancias tecnológicas, incorporando dos o más tecnologías para comunicación o codificación de datos. Hacen de *proxys* para sus servicios de Negocio y representan la funcionalidad de los servicios que están por debajo en un ambiente que es distinto tecnológicamente que el ambiente de ejecución del servicio de Negocio original. Los *Adapters* son un tipo especial de servicio intermediario que mapea las firmas y formatos de mensaje de un servicio a los requerimientos de un cliente. Los servicios de *Facade* tienen como propósito proveer una vista distinta (y posiblemente agregada) de uno o más servicios existentes, por lo que también pueden actuar como *technology gateways* y/o *adapters*. Como el patrón definido en [11], una *facade* puede utilizarse para proveer una vista específica de un conjunto de servicios que se encuentran por debajo, que en general son servicios básicos. Finalmente, los servicios intermediarios de tipo *functionality-adding* se usan cuando se quiere agregar funcionalidad a un servicio existente sin cambiar el servicio mismo. En este caso se crea un servicio que provea la misma funcionalidad del servicio original y agregue las nuevas características requeridas.

Los servicios centrados en procesos encapsulan el conocimiento de los procesos de negocio de la Organización y controlan y mantienen el estado del proceso en ejecución. Son tanto consumidores como proveedores y desde el punto de vista técnico son la clase de servicios más sofisticada. Una de las principales ventajas que presentan estos servicios es que separan la lógica de los procesos, definiendo los procesos de negocio y su control, en base a orquestación de servicios existentes. Es en este tipo de servicios donde aparecen los BPMS y los lenguajes de modelado de procesos como BPML [12] (Business Process Modeling Language) y BPEL4WS [13] (Business Process Execution Language for Web Services) para orquestar servicios. Por último los servicios empresariales públicos pueden o no existir dependiendo del negocio, ya que a diferencia de los anteriores son servicios que una empresa ofrece a socios y clientes externos. Estos servicios tienen requerimientos específicos de facturación de uso, interfaz, desacoplamiento y seguridad ya que las empresas deben acordar claramente como se realiza el uso de los mismos.

En base a la clasificación de servicios provista, al abordar un proyecto de construcción o transformación de un sistema basado en SOA, se plantea el desafío de cómo definir y estructurar los niveles de SOA. Estos niveles están relacionados directamente con los tipos de servicios que se incluirán, como se indica en [10]. Lo que determinan estos niveles es una progresión posible en la adopción del enfoque SOA, de forma de minimizar el impacto del mismo en la Organización. Comenzando por el nivel de "Fundamental SOA" que se compone únicamente de dos capas de abstracción: *application frontend* y servicios básicos, donde se identifican, definen e implementan servicios básicos del negocio que permitan que dos o más aplicaciones compartan la lógica de negocio. Aunque es un enfoque bastante simple provee las bases para ir avanzando hacia una SOA de mayor nivel. Es técnicamente fácil de implementar, aunque las *application frontend* son complejas ya que deben implementar el control de los procesos de negocio y la integración con el backend. El siguiente nivel de "Networked SOA" agrega la capa de abstracción de servicios intermediarios que pueden incluir servicios del tipo *facades*, *technology gateways*, *adapters* y *functionality-adding*, que como se mencionó encapsulan parte de la complejidad de los servicios básicos y permiten la integración de varias aplicaciones sin tener que cambiarlas completamente, mientras se van introduciendo los cambios.

Por último, el nivel de "Process-Enabled SOA" es el más completo, donde los procesos de la Organización se modelan en servicios orientados a procesos que mantienen además el

estado. En este nivel se conjuntan el enfoque SOA con el de BPM, constituyendo el ideal del modelado de procesos en la Organización y su informatización independiente de la tecnología. Las applications frontend son más simples, y la capa de servicios intermediarios puede o no existir. Una ventaja de este nivel es que permite que la lógica de los procesos quede claramente separada de otro tipo de código, como puede ser el control de diálogos y el núcleo de lógica de negocio localizado en los servicios básicos. Utilizando un BPMS para la definición y control de estos procesos, se evita la codificación de información y reglas de negocio directamente en la aplicación. Esto facilita la modificación, re-configuración y optimización de las definiciones de los procesos con herramientas gráficas que permiten definir los flujos de los procesos, por ejemplo en forma de Diagrama de Actividad de UML. Se puede modificar o agregar procesos con pocos ajustes, ya que los cambios están acotados a los servicios que representan el control de los procesos. La lógica de negocio será implementada una única vez en forma centralizada en un servicio, aumentando la reutilización del conocimiento y reduciendo inconsistencias y redundancias.

3 Metodología de desarrollo

La mayoría de los procesos de desarrollo de software en la actualidad están basados principalmente en la filosofía de la interacción y el cambio, teniendo en cuenta que los requerimientos de los sistemas son inestables y debe ser posible incorporarlos al desarrollo a medida que van surgiendo. Estos modelos, estén clasificados como “pesados” o “ágiles”, típicamente indican liberaciones frecuentes que son utilizadas para obtener feedback de los usuarios, y la construcción del sistema en forma iterativa incremental sobre estas liberaciones. Una metodología para el desarrollo orientado a servicios no es una nueva metodología, sino que se construye sobre el proceso o enfoque de desarrollo que se siga en la Organización, agregando actividades y artefactos específicos para el desarrollo de los servicios, y por lo tanto cualquier proceso podría servir de base. Sin embargo, dada la naturaleza compleja y cambiante de los requerimientos del negocio en el contexto de las Organizaciones que podrían incorporar el enfoque SOA, es recomendable que se siga un enfoque iterativo incremental como forma de lidiar con estos cambios. Dada también la importancia estratégica de asumir un proyecto de desarrollo con enfoque SOA se espera contar con un proceso base que sea más del tipo de los “pesados” de forma de ir obteniendo productos claves en el avance del mismo.

3.1 Proceso de desarrollo base

El principal proceso de desarrollo base con que cuenta la Organización, en este caso el Grupo de Ingeniería de Software (Gris) en el curso “Proyecto de Ingeniería de Software”, es una adaptación del Rational Unified Process (RUP) que ha ido evolucionando desde su primera versión que fuera definida por estudiantes en el año 2000 como proyecto de grado [14]. En esta edición se compone de un esqueleto de actividades que es común a todos los tipos de desarrollo que se emprendan, y dos extensiones principales para realizar desarrollos específicos en OO (Orientación a Objetos) y Genexus[15], herramienta de cuarta generación para el desarrollo de aplicaciones orientadas a bases de datos. En este contexto la metodología de desarrollo para aplicaciones en con enfoque SOA se incorpora al esqueleto base más la extensión OO, agregando actividades y entregables específicos para el desarrollo orientado a servicios.

El proceso base tiene como el RUP dos dimensiones, el tiempo y las disciplinas. En la dimensión del tiempo se definen cuatro fases: Inicial, Elaboración, Construcción y Transición, en las tres primeras fases se definen dos iteraciones de dos semanas cada una, y en la última fase una iteración de dos semanas, que completan junto con la semana de preparación previa y la semana de evaluación final, las dieciséis semanas con que cuenta el curso “Proyecto de Ingeniería de Software”. En la dimensión de las Disciplinas se definen las disciplinas tradicionales del desarrollo de software: requerimientos, diseño, implementación y verificación, más las disciplinas de soporte: gestión del proyecto, de la calidad, de la configuración e implantación. En cada disciplina se definen actividades, roles encargados y

participantes de las actividades, y entregables de entrada y salida de dichas actividades. Se cuenta además con una agenda definida que indica para cada semana que actividades se deben realizar y que entregables se deben obtener. Una descripción completa del proceso base está fuera del alcance de este artículo y puede consultarse su concepción y evolución en [5], pero se mencionará en lo que sigue, como se incluyen las disciplinas, actividades y entregables definidos en la metodología de enfoque SOA, con el proceso base adaptación del RUP. Vale aclarar también, que se sigue la notación definida para el proceso base, donde las actividades se codifican identificando la Disciplina a la que pertenecen y numerándolas consecutivamente, así las actividades de Requerimientos se notan R1, R2, etc. y las de Diseño D1, D2, etc.

3.2 Disciplinas, Actividades, Entregables y Roles para SOA

La Extensión SOA [6] como se mencionó previamente, requiere la realización de las actividades definidas en las Disciplinas del proceso base, agregando las Disciplinas y actividades específicas para SOA que define la metodología, así como entregables específicos del enfoque. Entre los roles definidos los más importantes para la Extensión SOA son: el Arquitecto, quién tiene responsabilidad en el área de Diseño pero participa también en el relevamiento de requerimientos y como coordinador del desarrollo; el Analista, quién principalmente participa en el relevamiento de requerimientos; el Especialista Técnico, quién tiene a su cargo el estudio e investigación de las tecnologías disponibles a utilizar y el Implementador, quién tiene a su cargo la codificación de la aplicación.

En primer lugar se agrega una disciplina que está en el RUP pero no en el proceso base, que es el Modelado del Negocio. Como se mencionó un aspecto principal en una SOA es el modelado de los procesos de la Organización como forma de conjuntar los aspectos del negocio con las soluciones informáticas provistas, teniendo en cuenta que los servicios orientados a procesos constituirán la traducción de los procesos del negocio a los procesos que se modelan en las aplicaciones de la Organización. Como parte central de la metodología definida se incorporan cinco actividades en la Disciplina de Diseño, que son las actividades claves del enfoque, en las que se identifican, categorizan, reutilizan, especifican, y definen los servicios de la aplicación, y la orquestación de los mismos para proveer los procesos del negocio modelado. Luego en la Disciplina de Implementación se agrega la actividad de implementación de servicios según fueron definidos en la etapa de diseño. Además para cada actividad se definen los roles responsables, participantes y los entregables entrada y salida de las mismas.

4 Modelado del Negocio (MN)

El propósito de la disciplina Modelado del Negocio es: asegurar que clientes, usuarios finales, desarrolladores y otros involucrados tienen un conocimiento común de la Organización objetivo, derivar los requerimientos del sistema de software que son necesarios para apoyar a la Organización y comprender como el sistema de software a implantar se inscribe en la misma. El esfuerzo de modelado del Negocio puede tener distintos alcances dependiendo del contexto y las necesidades. En este caso se está en el escenario de obtener un mapa de la Organización y sus procesos para ganar un mejor entendimiento de los requerimientos de la aplicación en desarrollo. El modelado del Negocio es parte entonces del proyecto de Ingeniería de Software y es realizado principalmente durante la Fase Inicial. En esta disciplina participan el rol de Analista y Arquitecto que son quienes realizan principalmente el relevamiento de requerimientos a partir de reuniones con el cliente. Los entregables que se obtienen de esta Disciplina son la Evaluación de la Organización Objetivo que contiene aspectos claves de la misma, y el Modelo de Casos de Uso del Negocio donde se especifican los procesos identificados en la Organización.

Evaluar la Organización Objetivo (MN1)

Objetivo: Esta actividad tiene como objetivo involucrar al equipo de proyecto con la Organización para la cual se está realizando el desarrollo, interiorizando a los participantes sobre aspectos como: el área, funcionamiento, empleados, etc. de la Organización.

Descripción: Se debe describir el estado actual de la Organización en la cual la aplicación será implantada, en términos de sus procesos actuales, herramientas, competencias de las personas, actitudes de las personas, clientes, competencia, desafíos tecnológicos, problemas y áreas de mejora, identificando claramente los involucrados (stakeholders) en el esfuerzo de modelado del negocio.

Roles: El rol responsable de esta actividad es el Analista y participa también el rol de Arquitecto, ya que ambos roles están involucrados en el modelado del Negocio a partir del cual se derivan los requerimientos para la aplicación conjuntamente con el relevamiento de requerimientos que se realiza en la Disciplina de Requerimientos posteriormente.

Entregables: Esta actividad tiene como entrada información recabada con los clientes del Negocio sobre diversos aspectos de la Organización. Como salida tiene el documento de Evaluación de la Organización Objetivo donde se detallan los aspectos claves relevados.

Identificar los procesos del Negocio (MN2)

Objetivo: Esta actividad tiene como objetivo describir y entender los procesos que se realizan en el negocio en lo que tiene que ver principalmente con la aplicación que se va a desarrollar. Para esto los procesos se describen como Casos de Uso del Negocio con los flujos correspondientes y se modelan con Diagramas de Actividad.

Descripción: Esta actividad está compuesta de varias sub-actividades que en conjunto permiten identificar los Procesos del Negocio, describiendo los actores participantes y el flujo de ejecución de los mismos en la Organización. Para esto se deben encontrar los actores y Casos de Uso del Negocio, las reglas del Negocio y los términos que se manejan en el Negocio. Es importante definir los límites del Negocio a ser modelado y quién y qué interactúa con el Negocio, describir los procesos del Negocio y plasmarlos en diagramas de actividad en el Modelo de Casos de Uso del Negocio.

Roles: El rol responsable de esta actividad es el Analista y participa también el rol de Arquitecto, igual que en la actividad anterior.

Entregables: Esta actividad tiene como entrada el documento de Evaluación de la Organización Objetivo y el/las Acta/s de Requerimientos realizada/s en las reuniones de requerimientos con el cliente. Como salida tiene el Modelo de Casos de Uso del Negocio donde se definen los flujos de los procesos identificados en la Organización Objetivo, así como el Glosario con la inclusión de los nuevos términos que vayan apareciendo.

Relación e inclusión de esta Disciplina con el Proceso Base adaptación del RUP

En el proceso base la primer Disciplina de trabajo es la de Requerimientos, en la cual se deben realizar actividades como R1 – Especificar los requerimientos y R2 – Definir el Modelo de Casos de Uso del Sistema. La Disciplina de Modelado del Negocio, como plantea el RUP, se realiza en forma previa a la de Requerimientos y constituye una entrada fundamental para la misma. Los Casos de Uso del Negocio definidos en base al estudio de los procedimientos de la Organización, se analizan para identificar los Casos de Uso del Sistema, que en general serán de menor granularidad dado que identifican interacciones puntuales de los actores con el Sistema, y éstas se encuentran inmersas en procesos de mayor duración y que generalmente cruzan las fronteras de Departamentos de la Organización o incluso de la Organización misma. En lo que sigue se presenta un ejemplo que relaciona los dos tipos de Casos de Uso definidos:

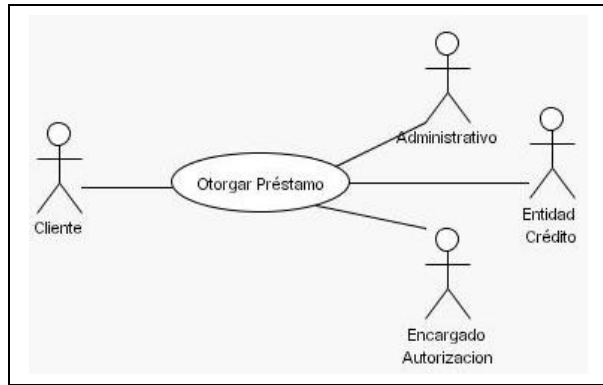


Figura 2: Caso de Uso del Negocio “Otorgar Préstamo”

En la figura se presenta el Caso de Uso del Negocio: Otorgar Préstamo que corresponde al procedimiento que tiene definido un banco para realizar los pasos correspondientes al otorgamiento de préstamos a sus clientes. Este procedimiento comienza cuando un cliente se acerca al banco a solicitar un préstamo, presentando la documentación asociada. Esta solicitud es entonces enviada a la sección de Autorización de préstamos, donde los encargados realizan el estudio de la documentación presentada, así como la evaluación de ingresos-egresos del solicitante, monto pedido, garantías presentadas, así como pedido de informes a alguna entidad de créditos como puede ser por ejemplo un Clearing de Morosos. La sección expide entonces su resolución aprobando o rechazando la solicitud, y ésta es enviada al área de atención al cliente la que se comunica con el solicitante para informarle de la resolución tomada. En caso de ser aprobada la solicitud, el solicitante debe volver al banco a retirar el efectivo y firmar los pagaré correspondientes. En ambos casos, otorgamiento o rechazo, se archiva la solicitud para tener la historia de solicitudes de la persona y resoluciones asociadas. Este Caso de Uso del Negocio describe todos los pasos y los distintos actores involucrados. Cuando debemos pasar a la identificación de los Casos de Uso del Sistema, podemos ver que en el procedimiento completo se ejecutarán más de un Caso de Uso del Sistema, que se pueden identificar como sigue:

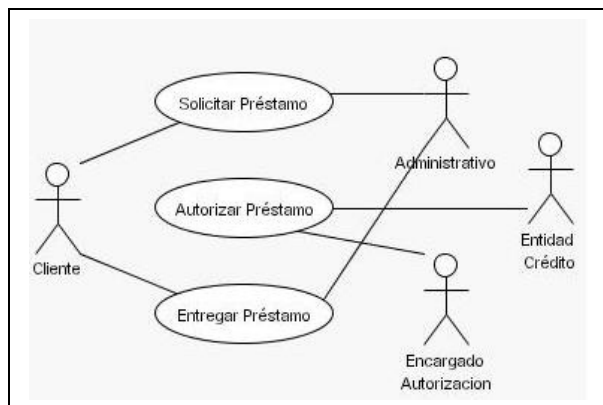


Figura 3: Casos de Uso del Sistema identificados a partir del Caso de Uso del Negocio “Otorgar Préstamo”

El Caso de Uso del Sistema “Solicitar Préstamo” corresponde a la primer parte de la ejecución del proceso, donde la persona se presenta en el banco a solicitar el préstamo y sus datos y comprobantes son ingresados al sistema, registrándose el pedido y otorgando un número de solicitud asociado. A continuación se ejecutará por parte de la Sección de Autorización de préstamos el Caso de Uso del Sistema “Autorizar Préstamo” donde intervienen no solo los encargados de la sección sino la interacción con la entidad de créditos que proveerá información sobre la morosidad del solicitante en créditos otorgados por otras instituciones previamente. Este Caso de Uso finalizará con la toma de la resolución correspondiente por parte de los encargados luego de evaluar toda la información provista. Finalmente, se ejecutará el Caso de Uso del Sistema “Entregar Préstamo”, donde el Área de Atención al cliente se comunicará con el mismo para informar de la resolución, si la resolución es sido favorable el cliente concurrirá al banco a retirar el efectivo solicitado y

firmar los pagaré correspondientes, si no únicamente se registrará el rechazo. Los Casos de Uso del Negocio serán los candidatos a convertirse en servicios centrados en procesos, donde el flujo del procedimiento se defina según el relevamiento realizado, y los distintos pasos de la ejecución del mismo sean realizados por operaciones correspondientes a servicios de menor granularidad que realizarán los Casos de Uso del Sistema identificados.

5 Diseño

El propósito de la disciplina de Diseño en una SOA agrega los siguientes objetivos a los definidos en el modelo base: identificar y catalogar los servicios necesarios para cumplir con los procesos identificados de la Organización en los Casos de Uso del Negocio definidos especificar estos servicios definiendo las interfaces y sus operaciones, así como los componentes que los implementarán y la reutilización de otros servicios y componentes dentro de la Organización definir la secuencia de invocaciones a servicios necesaria para la ejecución de los procesos del Negocio identificados como Casos de Uso del Negocio en la orquestación de servicios que define la coreografía de servicios propender a la utilización de un BPMS (Business Process Management System) que permita la definición del workflow del proceso, y BPML (Business Process Modeling Language) para la orquestación de servicios.

Identificar y categorizar servicios (D6)

Objetivo: Esta actividad tiene como objetivo identificar los servicios necesarios para realizar los procesos del negocio, clasificándolos según el tipo de servicio: básicos, intermediarios, centrados en procesos y públicos empresariales.

Descripción: Teniendo en cuenta la Descripción de la Arquitectura en sus vistas de Casos de Uso del Sistema y Lógica, se realiza en primer lugar un análisis de los subsistemas definidos, asegurándose que existe un mapeo de por lo menos un subsistema a cada área funcional del Negocio, por ejemplo clientes. De la misma forma se identifica claramente en que subsistema se realiza cada Caso de Uso del Sistema definido, comenzando con los Casos de Uso identificados como relevantes a la Arquitectura. A partir de esta correspondencia se identifican los servicios que deben ser provistos por cada subsistema para realizar los distintos Casos de Uso del Sistema, y seguidamente los Casos de Uso del Negocio correspondientes a los mismos. Los servicios identificados se clasifican según se estableció previamente.

Roles: El rol responsable de esta actividad es el Arquitecto y participan también los roles de Analista y Especialista Técnico, como forma de conjuntar el conocimiento del Negocio y el conocimiento técnico de las herramientas a utilizar.

Entregables: Esta actividad tiene como entrada los entregables Descripción de la Arquitectura (SAD) como está definido en el RUP, y el Modelo de Casos de Uso del Negocio, para contar con la información técnica del software en diseño y los procesos del negocio identificados para la Organización. Como salida tiene el Modelo de Servicios con la sección correspondiente a la identificación y categorización de servicios informada.

Especificar servicios (D7)

Objetivo: Esta actividad tiene como objetivo especificar los servicios identificados, definiendo los contratos de servicio para cada uno incluyendo las interfaces que brindará y sus operaciones, parámetros, etc.

Descripción: Para cada servicio identificado se realiza el contrato funcional especificando para cada interface definida los métodos que deben ser implementados para proveer el servicio acordado para la interface. Para cada operación en la interface se debe especificar la información siguiente: a) nombre del método, b) parámetros requeridos por el método, y para cada parámetro el nombre, tipo y descripción, c) valor retornado, indicando el nombre, tipo y descripción, d) lista de excepciones levantadas, e) breve descripción de la funcionalidad provista, f) pre-condiciones requeridas para la ejecución exitosa de la operación, g) post-condiciones que serán válidas luego de la ejecución del método.

Roles: El rol responsable de esta actividad es el Arquitecto y participa también el rol de Analista, ya que luego de definidos los servicios la especificación puede ser realizada por ambos.

Entregables: Esta actividad tiene como entrada y salida el Modelo de Servicios, como entrada tiene la sección de identificación y categorización de servicios informada, y como salida la sección correspondiente a la especificación informada.

Investigar servicios existentes (D8)

Objetivo: Esta actividad tiene como objetivo encontrar servicios que ya estén implementados en la plataforma SOA y puedan o deban ser reutilizados en la aplicación en desarrollo. En general los servicios que serán más reutilizados son los servicios básicos.

Descripción: Una vez que los servicios necesarios han sido identificados y especificados, se deben investigar los servicios con que se cuenta en la Organización, tanto básicos como de otros niveles, para reutilizar en la aplicación en desarrollo. Puede suceder que se deban implementar servicios intermediarios para reutilizar otros servicios que no se adapten totalmente a los requerimientos de la aplicación lo cual será preferible a implementar nuevamente el servicio desde cero. Los servicios básicos como los de acceso a datos, autenticación, seguridad, sesión, etc. deberán ser reutilizados por todos los servicios definidos en niveles superiores de la SOA.

Roles: El rol responsable de esta actividad es el Arquitecto y participa también el rol de Analista, ya que la investigación de servicios existentes tiene que ver con el modelado del negocio realizado.

Entregables: Esta actividad tiene como entrada la Evaluación de la Organización Objetivo realizada, donde están indicadas las capacidades existentes en la Organización. Como salida tiene el Modelo de Servicios con la sección correspondiente a los servicios existentes informada.

Asignar servicios a componentes (D9)

Objetivo: Esta actividad tiene como objetivo definir los componentes que deberán ser implementados para proveer los servicios especificados, esta correspondencia no tiene porque ser 1:1 con los subsistemas identificados previamente. Asimismo se debe especificar para cada componente los servicios que provee.

Descripción: Para asignar servicios a componentes se debe responder la pregunta de quién (cual componente) proveerá la implementación y gestión de los servicios definidos según las interfaces especificadas. Se debe tener en cuenta los servicios existentes identificados si existen componentes que los brinden o funcionalidades existentes para las cuales se pueda crear algún servicio intermediario que las provea. Para cada componente necesario se realiza la especificación del componente, indicando reglas del Negocio y servicios que implementa, así como otros componentes que utiliza.

Roles: El rol responsable de esta actividad es el Arquitecto y participa también el rol de Especialista Técnico, ya que involucra mapeos del diseño a la implementación.

Entregables: Esta actividad tiene como entrada el Modelo de Diseño y el Modelo de Servicios, con la información del software diseñado y los servicios identificados, categorizados, especificados y existentes. Como salida tiene el Modelo de Implementación y el Modelo de Servicios, con la asignación de componentes correspondiente en cada caso.

Definir orquestación de servicios (D10)

Objetivo: Esta actividad tiene como objetivo definir la secuencia de interacción entre servicios que es necesaria para realizar los procesos del Negocio identificados y descritos como Casos de Uso del Negocio.

Descripción: La orquestación de servicios para la implementación de procesos del Negocio se muestra para cada proceso en un diagrama de secuencia que describa la interacción entre los distintos servicios involucrados. Además se utilizará preferiblemente una herramienta BPM (Business Process Management) para definir, ejecutar y gestionar la orquestación del servicio centrado en procesos, mediante la utilización de BPML (Business Process Modeling Language) como BPELWS y el workflow de ejecución del proceso del Negocio descrito.

Roles: El rol responsable de esta actividad es el Arquitecto y participan también el rol de Analista y Especialista Técnico, ya los servicios centrados a procesos tienen un componente importante de visión del negocio y otro de visión técnica.

Entregables: Esta actividad tiene como entrada el Modelo de Casos de Uso del Negocio y el Modelo de Servicios, que tienen toda la información para definir los servicios centrados en procesos. Como salida tiene el Modelo de Implementación y el Modelo de Servicios con los mapeos correspondientes realizados y la orquestación de servicios definida para los servicios centrados en procesos.

Relación e inclusión de esta Disciplina con el Proceso Base adaptación del RUP

La Disciplina de Diseño en el proceso base cuenta con una importante cantidad de actividades que se deben realizar en coordinación con las que plantea la metodología para el enfoque SOA, en la cual se define la agrupación de actividades especificada como DA1 – Definir Servicios que corresponde a la realización conjunta de las actividades D6 – identificar y categorizar servicios, D7 – especificar servicios y D8 – investigar servicios existentes, ya que las tres se realizan en forma conjunta por la relación que tienen entre sí, y además se realiza iterando con la actividad del proceso base identificada como D2 – Describir la Arquitectura, ya que se retroalimentan entre sí. En este sentido, dicha actividad plantea la identificación de los Casos de Uso del Sistema relevantes a la Arquitectura, que son aquellos que por sus requerimientos tanto funcionales como no funcionales, definen el esqueleto de la aplicación a construir en cuanto a definición de subsistemas e interacción entre los mismos y con otras aplicaciones, definiendo por ejemplo la necesidad de aplicar patrones de diseño como la encapsulación del acceso a un dispositivo de hardware o a otro sistema sobre el cual no se tiene control, o el patrón de acceso a datos, entre otros. En base a dichos Casos de Uso se realiza la modularización del software en subsistemas con alta cohesión y bajo acoplamiento que permitan identificar todas las áreas que involucra el desarrollo, ejercitando distintas hipótesis sobre el comportamiento del mismo. Es teniendo en cuenta esta modularización, los Casos de Uso del Sistema identificados, los Casos de Uso del Negocio que describen los procesos de la Organización, y la clasificación de servicios presentada, que es posible pensar y definir los servicios en los distintos niveles, que seguirán luego el flujo de trabajo de la disciplina de Diseño en la metodología SOA planteada.

El proceso base tiene como entregables fundamentales de la Disciplina de Diseño, en primer lugar el Documento de Descripción de la Arquitectura o SAD (por sus siglas en inglés Software Architecture Document) en el cual se presentan las distintas vistas que tienen injerencia en la comprensión de la Arquitectura definida para la aplicación, siguiendo el modelo planteado en [16] que son: Vista del Modelo de Casos de Uso, del Modelo de Diseño (incluye vista lógica y de procesos), del Modelo de Implementación y del Modelo de Distribución (o Despliegue), y en segundo lugar el Modelo de Diseño donde se presenta el diseño para todos los Casos de Uso del Sistema. La metodología agrega como entregable fundamental de esta disciplina el Modelo de Servicios que acompaña al Modelo de Diseño del cual el SAD presenta únicamente la vista correspondiente a los Casos de Uso relevantes a la Arquitectura, y agrega también en el SAD la vista del Modelo de Servicios donde se muestran los servicios definidos para los Casos de Uso relevantes a la Arquitectura.

Flujo de actividades para la Disciplina de Diseño

A continuación se presenta a modo de ejemplo el flujo de actividades para la Disciplina de Diseño modelado como Diagrama de Actividad de UML. Este flujo se ejecuta cada vez que se entra a una iteración en una Fase del proceso al momento de realizar las actividades de diseño que corresponda. Según la Fase e iteración en la que se esté se realizarán todas, algunas o ninguna de las actividades definidas, según las transiciones permitidas en el flujo.

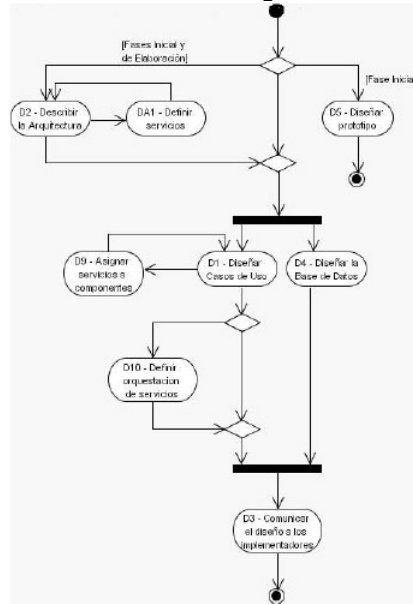


Figura 4: flujo de las actividades de Diseño en la metodología SOA

6 Implementación

El propósito de la disciplina de Implementación en una SOA agrega los siguientes objetivos a los definidos en el modelo base: implementar los componentes proveedores de servicios según lo establecido por la asignación de servicios a componentes implementar el ligamiento de servicios en las application frontend y en los servicios que llaman a otros servicios, utilizando la estrategia definida en la Organización.

Implementar servicios (I13)

Objetivo: Esta actividad tiene como objetivo implementar los servicios definidos, para lo cual se deben tener en cuenta el tipo de servicio, las interfaces diseñadas, la interacción con otros servicios (con o sin repositorio de servicios, con ligamiento en tiempo de desarrollo o de ejecución).

Descripción: En esta actividad se deben implementar los componentes definidos para los servicios identificados, de acuerdo a las interfaces y componentes especificados en la Disciplina de Diseño. Se debe proveer para cada interface los métodos definidos respetando los parámetros (nombre, tipo) especificados, valor retornado (nombre, tipo) y la utilización de otros servicios para la realización de las operaciones definidas, según lo establecido en la especificación de las interfaces y componentes definidos.

Roles: El rol responsable de esta actividad y único participante es el Implementador que programa los servicios definidos.

Entregables: Esta actividad tiene como entrada el Modelo de Implementación y el Modelo de Servicios, donde está definida la correspondencia entre el diseño y la implementación. Tiene como salida el servicio implementado.

7 Pruebas de la Metodología

El primer desarrollo en que se utilizó la metodología fue en la edición 2005 del curso “Proyecto de Ingeniería de Software” que dicta el Grupo de Ingeniería de Software (Gris), donde dos grupos desarrollaron una aplicación de Help-Desk para el Proyecto Link-All de la Unión Europea y el Instituto de computación. El Proyecto Link-all es financiado por la Unión Europea e integrado por diecisiete socios de Sudamérica y Europa, está dirigido a asistir en la apropiación de prácticas de tecnologías de la información y comunicaciones (TIC), apoyar la colaboración, promover el intercambio de experiencias, la transferencia de conocimiento y mejorar habilidades en tres sectores objetivo en Latinoamérica: artesanía, eco-agro turismo y cultura. El Sistema Link-all incluye herramientas diseñadas e integradas para proveer de una serie de facilidades de inclusión tecnológica, que fortalezcan la integración de actividades de desarrollo local que posibiliten y faciliten la inserción exitosa de comunidades remotas en el mercado global, a través del desarrollo de una red transnacional europeo-latinoamericana de cooperación. El sistema Link-all está desarrollado orientado a una arquitectura Process-Enabled SOA, en la que se identifica un primer nivel de servicios de infraestructura, sobre el cual se encuentra el segundo nivel con servicios orientados al dominio que son orquestados en un tercer nivel, en el que se construyen los procesos de negocio. Este caso de estudio puede verse en detalle en [8].

La metodología será probada nuevamente este año en dos proyectos de fin de carrera que desarrollarán componentes que serán integrados a una aplicación de tipo LIMS (Laboratory Information Management System) en un proyecto conjunto entre el Instituto de Computación y el Instituto Pasteur Montevideo. Los sistemas LIMS existentes en general se encuentran orientados al área farmacéutica y no proveen todas las funcionalidades que son necesarias en un ambiente biológico de investigación como es el Instituto Pasteur. Este tipo de sistemas permite compartir datos de los experimentos realizados por los distintos grupos así como conclusiones obtenidas en base a los mismos, lo que aporta al avance del área. Para ambos proyectos se plantea la investigación de los aspectos conceptuales del enfoque de desarrollo SOA y tecnologías asociadas, así como el comportamiento de las mismas en la integración de aplicaciones móviles para un grupo, y con utilización de workflow para el otro. Se quiere probar también la aplicabilidad de las herramientas WebSphere de IBM [17] al enfoque de desarrollo SOA y a los requerimientos del uso de aplicaciones móviles y workflows en este contexto. Se incluirá desarrollo de componentes en otras herramientas que no sean propietarias de IBM realizando interacción entre éstos y los desarrollados con herramientas IBM, con el objetivo de probar la interoperabilidad entre los mismos. En cuanto a las aplicaciones móviles se espera obtener una solución conceptual al problema de la integración de las mismas en un ambiente SOA, en lo que tiene que ver con el backend de la misma, no con el cliente móvil que sería el frontend. En cuanto a la integración de workflows se espera poder investigar el comportamiento de los mismos en una SOA como forma de definir los procedimientos para realizar los experimentos en la Organización, interactuando con los usuarios de los mismos en el momento en que sea necesario tomar alguna decisión por parte de los usuarios sobre el flujo definido.

8 Conclusiones y trabajo a futuro

En cuanto a la metodología para el desarrollo de aplicaciones en plataforma SOA definida, se piensa ajustar con los aportes realizados por el desarrollo de los proyectos del Help-Desk, así como incorporar actividades específicas en las áreas de Verificación, Gestión del Proyecto, Calidad y Configuración, que refuercen la orientación a servicios en el desarrollo, la verificación de los mismos, la planificación y gestión de configuración en base a estos, y el aseguramiento de la calidad de los servicios definidos y de su implementación. En la aplicación del año 2005 se vio que la mayor dificultad se presentaba en la Disciplina de Diseño donde se veía que no resultaba intuitivo iterar en la definición de subsistemas de la Arquitectura y la definición de servicios adecuados a estos. Si bien la dificultad en la definición de la granularidad correcta para los servicios identificados es reconocida en el enfoque SOA, se cree que en esta experiencia parte de la misma provenía de la falta de experiencia en diseño y desconocimiento de la conceptualidad del enfoque SOA, lo que se tradujo en dificultades en el diseño de los servicios y su orquestación. Con los proyectos de grado se espera evaluar su aplicabilidad identificando buenas prácticas y oportunidades de mejora, así como la inclusión

de patrones de comunicación e integración para la definición de arquitecturas de ese tipo, que sienten bases firmes para el diseño de este tipo de aplicaciones y la definición de los servicios en su granularidad correcta.

9 Referencias

- [1] Grupo de Ingeniería de Software (Gris), Instituto de computación, Facultad de Ingeniería, Universidad de la República, <<http://www.fing.edu.uy/inco/cursos/ingsoft/gris>> [Consulta: abril de 2006]
- [2] Proyecto de Ingeniería de Software, Instituto de computación, Facultad de Ingeniería, Universidad de la República, <<http://www.fing.edu.uy/inco/cursos/ingsoft/pis/>> [Consulta: abril de 2006]
- [3] IBM Rational Unified Process. <<http://www-130.ibm.com/developerworks/rational/products/rup/>> [Consulta: abril de 2006]
- [4] Beck, K. Extreme Programming Explained: Embrace Change, Addison Wesley Professional, 1999, ISBN 201-61641-6
- [5] Delgado, A. Pérez, B. “Modelo de Desarrollo de Software OO – Experimentación en un curso de Ingeniería de Software”, en V Jornadas Iberoamericanas de Ingeniería de Software e Ingeniería del Conocimiento (JIISIC’06), Puebla, México, Febrero de 2006, ISBN 970-94770-0-5
- [6] Delgado A. Metodología de desarrollo de aplicaciones con enfoque SOA (Service Oriented Architecture), Agosto 2005 <<http://www.fing.edu.uy/~adelgado/ExtensionSOA/index.htm>> [Consulta: abril de 2006]
- [7] Proyecto Link-All, Programa @lis, Unión Europea, en <<http://www.link-all.org/>> [Consulta: abril de 2006]
- [8] Delgado A., González L., Piedrabuena F. “Desarrollo de aplicaciones con enfoque SOA (Service Oriented Architecture)”, en V Jornadas Iberoamericanas de Ingeniería de Software e Ingeniería del Conocimiento (JIISIC’06), Puebla, México, Febrero de 2006, ISBN 970-94770-0-5
- [9] Proyecto LIMS, <<http://www.fing.edu.uy/inco/pm/Convenios/LIMS2005>> [Consulta: abril de 2006]
- [10] Krafzig, D. Banke, K. Slama, D. “Enterprise SOA, Service Oriented Architecture Best Practices”, Prentice Hall Pearson Education, 2005, ISBN 0-13-146575-9
- [11] Gamma E., Helm R., Johnson R., Vlissides J., “Design Patterns: Elements of Reusable Object-Oriented Software” Addison-Wesley, 1995, ISBN 0-201-63361-2
- [12] Business Process Management Initiative, <<http://www.bpmi.org/>> [Consulta: abril de 2006]
- [13] IBM et al, “Business Process Execution Language for Web Services, Version 1.1”, Mayo 2003.
- [14] Delgado, A. Pérez, B. Modelado del proceso de software - Informe final Proyecto de Taller V, Instituto de Computación, Facultad de Ingeniería, Universidad de la República, año 2000
- [15] Visión General, ARTech, Febrero 2003, <http://www.genexus.com/DOCUM/GeneXus_VG.pdf> [Consulta: abril de 2006]
- [16] Kruchten, Phillipe. “The 4+1 View Model of Architecture”. IEEE Software, Noviembre 1995, Volume: 12, pags: 42-50, ISSN: 0740-7459
- [17] IBM Websphere, <<http://www-306.ibm.com/software/websphere/>> [Consulta: abril de 2006]