



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

**FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA**

**INTEGRACIÓN DE SISTEMAS TECNOLÓGICOS
POR MEDIO DE LA ARQUITECTURA
ORIENTADA AL SERVICIO
DENTRO DEL GRUPO FINANCIERO.**

**PROYECTO DE TRABAJO
PROFESIONAL
QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN
PRESENTA:
Jonathan Adolfo Pérez Tiburcio**

Asesor

M.I. Aurelio Adolfo Millán Nájera



MÉXICO, D.F, CIUDAD UNIVERSITARIA, 2014

AGRADECIMIENTOS.

Quiero expresar mis más sinceros agradecimientos a la Universidad Nacional Autónoma de México por la oportunidad de llevar a cabo mi formación profesional, por brindarme una estancia inolvidable en una de las etapas más importantes de mi vida.

Agradezco a la Facultad de Ingeniería, así como a todos los profesores, que con esa dedicación y profesionalismo me brindaron parte de sus conocimientos, por permitirme tener una trayectoria académica de la carrera que tanto me apasiona.

Quiero agradecer por último a mis padres, quienes me brindaron el apoyo necesario con el cual he logrado culminar mi esfuerzo, terminando así mi carrera profesional, que es para mí la mejor de las herencias.

Contenido

INTRODUCCIÓN.	1
CAPÍTULO 1.....	3
La Empresa.....	3
Misión y Filosofía.....	3
Atención Personal.	3
Calidad Certificada.	3
Integridad.	3
Costo Beneficio.....	3
CAPÍTULO 2.....	5
Organigrama.	5
Descripción Organizacional.....	5
CAPÍTULO 3.....	9
Proyectos implementados en el sector financiero.	9
Implementación de PayTrue con el Core Bancario.....	9
Filtrado de Nombres para la prevención de lavado de dinero.	13
Declaración informativa mensual y anual de depósitos en efectivo.	15
Pago de servicios.	16
Implementación del producto tarjeta de débito.	17
CAPÍTULO 4.....	23
Integración de sistemas tecnológicos por medio de la arquitectura orientada al servicio dentro del grupo financiero.....	23
Objetivo.	23
Antecedentes	23
Marco Teórico.	25
Definición del problema o contexto de la participación profesional.....	35
Análisis y metodología empleada.	36
Implementación.	43
Participación profesional.....	44

CAPÍTULO 5.....	49
Resultados.....	49
Conclusiones.....	51
Glosario.....	52
Referencias.....	57
Bibliográficas.....	57
Digitales.....	57

INTRODUCCIÓN.

El presente documento se redacta con carácter de titulación por experiencia profesional, para la obtención del título de Ingeniero en Computación, dentro del cual se presentan los resultados obtenidos durante el desarrollo de distintos proyectos realizados en el ejercicio laboral, con el objetivo de demostrar la puesta en práctica de los conocimientos teóricos y prácticos adquiridos a lo largo de mi trayectoria académica en la Facultad de Ingeniería de la Universidad Nacional Autónoma de México.

A partir de este objetivo general, se seleccionó el proyecto de mayor robustez y con mayor impacto para la empresa, el cual se describe su implementación por medio de la metodología de software dictada por el área de sistemas.

Gracias a esta metodología interna se puede contar con un conjunto de procedimientos bien definidos, los cuales, facilitan las tareas durante todo el ciclo de vida del software, logrando así cumplir con los requerimientos funcionales definidos por el usuario.

En la primera parte del documento se describe los principales objetivos de la empresa, así como su misión y valores, cuidando de no mencionar el nombre de la empresa por cuestiones de privacidad de la información.

En el segundo apartado se describe por medio de un organigrama la estructura organizativa que conforma el departamento de sistemas como parte de la empresa, dentro de esta estructura se muestran las múltiples relaciones e interacciones que tienen las distintas partes o sub áreas.

Dentro de la descripción de las funciones de cada sub área se puede entender la forma en que la interacción de las distintas partes ayuda a cumplir los objetivos principales del área de sistemas dentro de una institución financiera, como lo son:

- Proveer y garantizar alta disponibilidad del hardware y las comunicaciones.
- Garantizar alta disponibilidad de los sistemas productivos con el fin de mantener en operación el negocio y prevenir pérdidas por inoperatividad de los sistemas.
- Atender los nuevos requerimientos de software dictados por las distintas áreas de negocio, con el fin de mantener los sistemas de acuerdo a las regulaciones de los distintos órganos, así como ofrecer desarrollos innovadores que permitan ofrecer mayores y mejores servicios para los clientes.

En la tercera parte del documento se describen los proyectos más completos y sobresalientes que desarrollé, los cuales fueron elegidos ya que hacen uso de conocimientos de distintos ámbitos como los son, programación, base de datos, redes, protocolos de comunicación y lógica del negocio en el sector financiero.

La cuarta parte del documento se centra en el proyecto principal titulado *“Integración de sistemas por medio de la arquitectura orientada al servicio dentro del grupo financiero”*, donde se inicia describiendo los objetivos principales de una arquitectura orientada al servicio, dentro de los cuales se hace notar principalmente como una solución al momento de reorientar los sistemas, gracias a la utilización de procesos de negocios en forma de servicios, lo cual ofrece grandes ventajas como la reducción en la complejidad de los desarrollos, al mismo tiempo que disminuyen los tiempos de implementación.

Adicionalmente se detalla dentro de un marco teórico, los conceptos fundamentales, así como las partes más importantes de una infraestructura que constituyen a una arquitectura orientada al servicio, para finalmente adentrarse a los antecedentes, análisis e implementación, para lo cual en esta última etapa se le da un seguimiento mediante la metodología interna, con la cual se hace notar mi participación profesional dentro del proyecto.

Finalmente en la última parte del documento se describe los resultados, así como las conclusiones, con las cuales se hace notar la experiencia adquirida durante este periodo, así como la demostración de la puesta en práctica de los conocimientos adquiridos durante mi formación académica, los cuales me han permitido desarrollar mis actividades propias del área de ingeniería en computación.

CAPÍTULO 1.

La empresa.

La organización en la que me encuentro laborando, es un grupo de empresas con una trayectoria exitosa en lo que corresponde al sector financiero, la misma que ofrece soluciones integrales basadas principalmente en el conocimiento de cada cliente, lo que destaca a la empresa ya que presta servicio a más de 50,000 mil clientes; por esta razón se encuentra comprometida de tal manera que todos los días se interactúa para poder asesorar en el manejo de las tesorerías, así como en las inversiones patrimoniales personales, con el fin de la satisfacción oportuna de las necesidades

Misión y filosofía.

La misión es proveer servicios financieros a corporaciones multinacionales, corporaciones mexicanas y a individuos, cumpliendo con la regulación local e internacional, comprometida con el desarrollo social y ofreciendo a sus empleados un excelente ambiente de trabajo, con una filosofía orientada al cliente y basada en cuatro valores fundamentales: atención al personal, calidad certificada, integridad, costo beneficio

Atención personal.

Sabemos que no existen necesidades iguales, y mucho menos clientes iguales, por ello, estamos convencidos de que cada uno requiere una respuesta oportuna y personalizada. Estamos preparados para escuchar, entender y diseñar una solución integral para cada uno de ellos.

Calidad certificada.

Diseñamos servicios con procesos simples y confiables, todos ellos certificados bajo la norma ISO 9001:2008, y con el compromiso de una mejora continua supervisada directamente por la alta dirección.

Integridad.

La congruencia entre el pensar, el decir y el actuar, son la base para establecer una relación de confianza con cada uno de nuestros clientes, ofreciendo la seguridad de que el manejo de sus inversiones se hará siempre bajo un marco de estricta confidencialidad.

Costo beneficio.

Proponemos a cada uno de nuestros clientes soluciones financieras que siempre busquen optimizar la relación costo – beneficio para así lograr la completa satisfacción de sus necesidades.

CAPÍTULO 2.

Organigrama.

Descripción organizacional.

La organización se encuentra conformada por distintas áreas dentro de las cuales se presenta el área de sistemas o tecnologías de la información (*véase figura 1*), dicha área se encuentra dividida en diferentes sub áreas, las cuales desempeñan tareas fundamentales que contribuyen al buen funcionamiento logrando así sus principales objetivos, entre los cuales se encuentran:

- Proveer y garantizar alta disponibilidad del hardware y las comunicaciones.
- Garantizar alta disponibilidad de los sistemas productivos con el fin de mantener en operación el negocio y prevenir pérdidas por inoperatividad de los sistemas.
- Atender los nuevos requerimientos de software dictados por las distintas áreas de negocio, con el fin de mantener los sistemas de acuerdo a las regulaciones de los distintos órganos, así como ofrecer desarrollos innovadores que permitan ofrecer mayores y mejores servicios para los clientes.

Dentro de la estructura del área se encuentra como figura principal el Director Corporativo de Sistemas, el cual tiene la función de dirigir al departamento creando estrategias y tomando las mejores decisiones, buscando cumplir con los objetivos primordiales del área en beneficio de la organización.

Debajo del director se encuentran gerentes y subdirectores los cuales fueron designados a dirigir a las distintas sub áreas que conforman el departamento.

Dentro de estas sub áreas se encuentran:

- Infraestructura.
- Operación de tecnologías de la información.
- Base de datos.
- Arquitectura.
- Desarrollo de sistemas.
- Aseguramiento de la calidad.
- Soporte de aplicaciones.

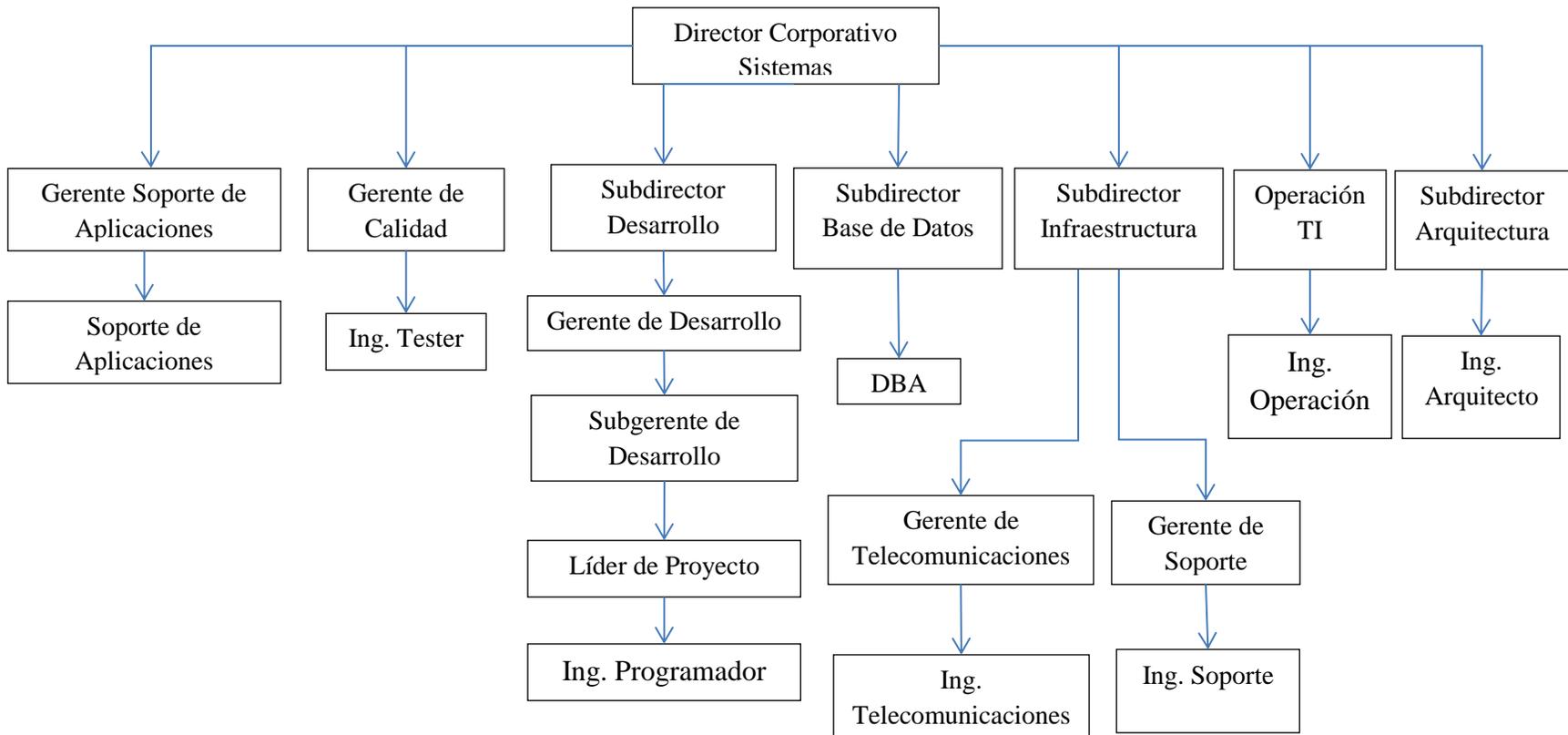


Figura 1 Organigrama del departamento de Sistemas.

Soporte de aplicaciones.

Es el área encargada de impartir cursos de capacitación sobre las nuevas funcionalidades de los distintos sistemas, comunicar y divulgar información sobre mejoras y soluciones de los sistemas de la empresa, así como atender a los usuarios en las problemáticas que presentan día a día ante el uso de los sistemas tecnológicos de la empresa, canalizando los problemas con las áreas correspondientes.

Aseguramiento de la calidad.

Es el área encargada de verificar y validar el software desarrollado, para lo cual debe realizar actividades primordiales como verificar el cumplimiento de los requerimientos funcionales del sistema, realizar el monitorear del proceso de desarrollo de software durante su ciclo de vida, realizar las pruebas a los distintos componentes desarrollados, todo esto con el fin de garantizar el menor número de errores en el sistema y asegurar la calidad en el producto final.

Desarrollo de sistemas.

Es el área encargada de la atención y solución de bugs y requerimientos de software, a los cuales se les da seguimiento a través de la metodología de desarrollo interna, con el fin de entregar un producto que cumpla con las especificaciones en tiempo y forma.

Base de datos.

Es el área encargada de mantener en alta disponibilidad las bases de datos, así como realizar constante monitoreo del performance de las distintas base de datos, generar respaldos, realizar liberaciones de desarrollos y dar mantenimiento y tuning a los distintos manejadores de bases de datos de la empresa.

Infraestructura.

Es el área encargada de monitorear y mantener en línea el hardware y las comunicaciones de la empresa con el fin de mantener los sistemas en línea y operables en todo momento.

Operación de tecnologías de la información o soporte a producción.

Es el área encargada de dar atención de incidentes y requerimientos sobre la infraestructura que soporta los servicios de TI, con el fin de mantener la disponibilidad y el correcto desempeño de la infraestructura de TI.

Arquitectura.

Es el área encargada de establecer políticas y estándares, distribuir las mejores prácticas de software entre las áreas de desarrollo, así como analizar, diseñar y proponer la arquitectura de los nuevos sistemas.

CAPÍTULO 3.

Proyectos implementados en el sector financiero.

Implementación de PayTrue con el Core Bancario.

Objetivo.

Implementar una herramienta para permitir el monitoreo de operaciones o transacciones originadas por los distintos medios de pago, con el fin de identificar patrones de comportamiento de los clientes y usuarios de acuerdo a las operaciones financieras que realicen.

Esta herramienta provee la capacidad de acotar pérdidas económicas, reduciendo el riesgo de fraude, con lo cual ayuda a reducir el riesgo de daño en la imagen de la empresa, al mismo tiempo que permite fortalecer la confianza hacia el cliente.

Descripción.

El sistema PayTrue está compuesto por distintos módulos que complementan la solución, los cuales son:

- **Detection Engine.**- Es el módulo que permite aplicar un conjunto de modelos y matrices paramétricas a las transacciones de autorización que se reciban desde diferentes canales, detectar aquellas que violen alguna regla predefinida y generar las alertas correspondientes, permitiendo modelar comportamientos, previniendo comportamientos fraudulentos de alta complejidad, a partir de su modelador de escenarios paramétricos.

Esta herramienta únicamente recibe una copia de las solicitudes de autorización o transacciones entrantes, para luego proceder a realizar la evaluación y analizar la información contenida en la transacción y de aplicar los controles establecidos por las reglas paramétricas que hayan sido definidas por la empresa. Estas reglas son totalmente configurables por el usuario y adaptables a cada organización.

- **Advanced Console.**- Es el módulo que permite a los analistas de riesgos completar exitosamente la tarea de investigación de las alertas de transacciones que reciben desde los diferentes sistemas de detección, logrando una óptima relación de falso/positivo en las alertas trabajadas.

Cualquiera que sea el origen de las alertas que la institución genere, *Advanced Console* es capaz de centralizar su procesamiento ofreciendo un conjunto de funcionalidades de administración de casos, que asegurarán que las mismas sean procesadas en tiempo y forma; pudiendo incluso bloquear cuentas o medios de pago, en modos preventivo o permanente, directamente desde la interfaz de trabajo.

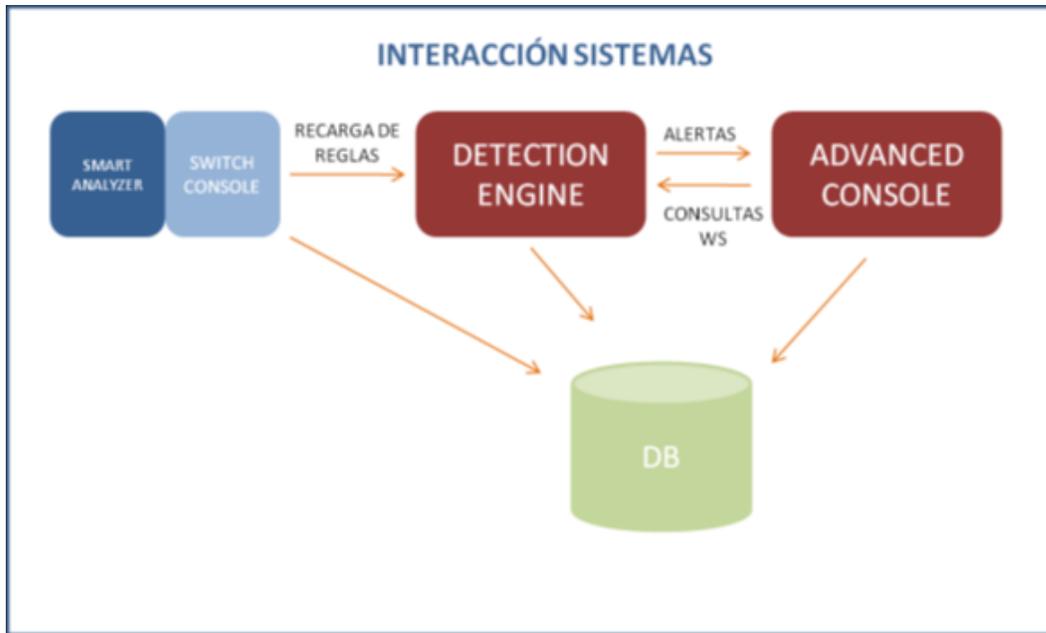


Figura 2 Interacción entre los módulos de la solución PayTrue.

Para la interacción entre los sistemas se planteó un mecanismo de intercambio entre el sistema PayTrue y el core bancario, el cual consistió en intercambio de tiras de bytes de texto plano. Cada tipo de mensaje consistió en una interfaz que define los datos requeridos para realizar una transacción financiera y no financiera.

Se estableció dos tipos de canales para la comunicación entre el core bancario y el motor de detecciones, el primer mecanismo consistió en ejecución *Real Time (RT)* y otro mecanismo en ejecución *Near Real Time (NRT)* para procesar aquellos mensajes con menor prioridad de tiempo de respuesta y mensajes recibidos en lote desde el core bancario para el monitoreo de forma POST.

Se estableció que el canal *RT* devolvería un código de respuesta mediante socket por cada transacción recibida. En caso que la respuesta del motor de detecciones sufriera una demora de más del periodo establecido, se definió que el flujo establecido continuaría de forma normal sin considerar la respuesta del motor de detecciones.

Los mensajes que se pactaron para el canal *Real Time* fueron los siguientes:

- Pago de servicios.
- Solicitud de cheques.
- Solicitud de transferencias por banca electrónica y sucursal.
- Inicio de sesión.
- Alta de cuenta destino.

El canal *Near Real Time* se estableció que no generaría una respuesta, sino que simplemente realizaría una marca si el mensaje fue procesado correctamente. Para el canal *Near Real Time* se estableció que se alimentaría de todas las transacciones a ser monitoreadas de forma POST y esta información sería utilizada por el motor de detecciones para evaluar las reglas de riesgo definidas generando las alertas necesarias.

Los mensajes que se pactaron para *Real Time* fueron los siguientes:

- Transacciones ATM/POS.
- Confirmación de cheques.
- Confirmación de transferencias.
- Confirmación de inicio de sesión.
- Confirmación de pagos.
- Confirmación de alta de cuentas destinos.
- Operación de banca electrónica.
- Información de clientes.
- Información de direcciones.
- Información de cuentas.
- Información de tarjetas.

Para el canal de *Near Real Time* se pactó utilizar como mecanismo de intercambio una tabla, en la cual se insertaría un registro por cada transacción, incluyendo un identificador, el mensaje intercambiado y la fecha y hora de inserción. Posterior al procesamiento del mensaje por el motor de detecciones, se procedería hacer el registro de la fecha y hora de procesamiento, así como el estado de éxito de la transacción.

Participación.

Desarrollé la comunicación RT y NRT para cada interfaz financiera y no financiera para lo cual analicé dentro del core bancario todos los puntos posibles donde se realizan transacciones.

Para la comunicación RT desarrollé sockets administrados por procesamiento de hilos, los cuales realizan la tarea de intercambiar la información transaccional definida por las interfaces en forma de tiras de bytes.

Para la comunicación NRT desarrollé un mecanismo de centralización de información, el cual en forma de proceso batch, toma la información generada de confirmaciones de transacciones, para el envío al sistema PayTrue mediante un proceso calendarizado mediante la herramienta crontab.

Infraestructura de la solución.

Para la instalación de PayTrue fueron necesarios dos servidores para aplicativos y un servidor de base de datos, los cuales alojan la solución de PayTrue; a su vez estos sistemas presentan conexión con el core bancario, el cual provee la información de las transacciones financieras y no financieras.

Dentro de la infraestructura se presentan las estaciones de trabajo, desde donde se da el monitoreo de las transacciones, así como la configuración de las reglas del motor de detecciones (como se muestra en la figura 3).

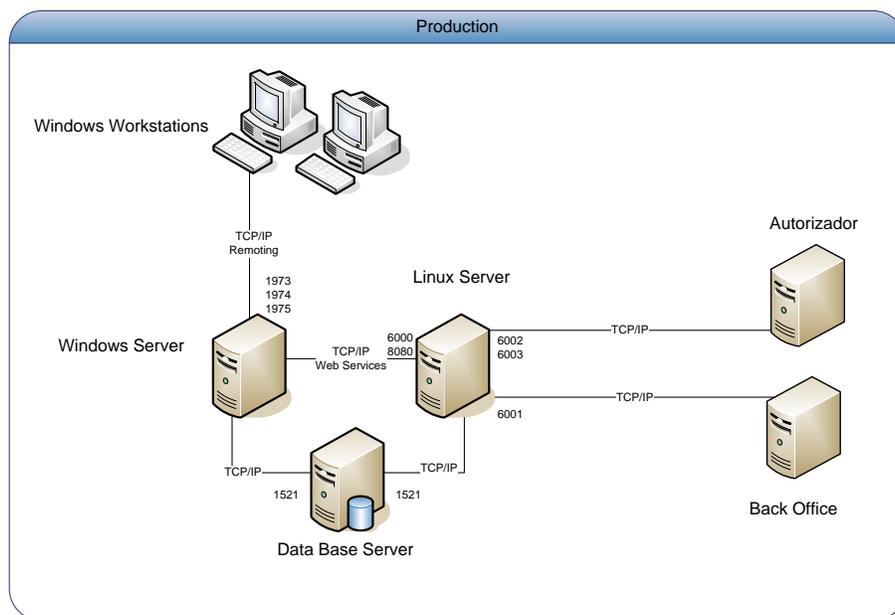


Figura 3 Diagrama de Infraestructura de PayTrue e interacción con el Core Bancario.

Filtrado de nombres para la prevención de lavado de dinero.

Objetivo.

Filtrar nombres dentro de los procesos operativos con el fin de contar con un monitoreo de coincidencias de nombres en los diferentes procesos operativos que involucran terceros, cumpliendo con las disposiciones oficiales para el filtrado de nombres.

Mitigar riesgos de realizar operaciones con personas que no mantienen un registro o relación contractual con la empresa y puedan estar relacionadas en listas restringidas y/o negras.

Alcance.

Filtrar todos los nombres tanto de personas físicas como morales que tengan participación o un rol dentro de una cuenta, contrato u operación comparándolo contra las listas negras, antes de establecer una relación de negocios o durante la vigencia de la misma así como en cada operación o servicio realizado por la institución.

Descripción.

Se planteó realizar el filtrado de información de nombres procedentes de contratos, operaciones y servicios del banco, por medio del sistema interno de filtrado de nombres, el cual tiene la facultad de realizar la comparación de nombres con listas actualizadas de PEPS (Personas Políticamente Expuestas) y listas negras, con el fin de generar alertas y así evitar la relación de negocios con personas no deseadas.

Para este proyecto se planteó realizar el filtrado de todos los nombres incluidos en los siguientes roles:

- Contratos:
 - Titular.
 - Cotitular.
 - Beneficiario.
 - Tercero autorizado.
 - Representante legal.
 - Accionista.
 - Apoderado.
 - Proveedor de recursos.
 - Propietario real.
 - Asesores.

- Operaciones y servicios:
 - Terceros
 - Destinatarios.
 - Ordenantes.
 - Beneficiarios.

Para el envío de información se planteó que la revisión considerara datos que permitieran descartar homonimias dentro de las comparaciones, por lo que se acordó considerar:

- Tipo de lista.
- Tipo de persona (físicas y morales).
- Fecha de nacimiento.
- Nacionalidad.
- RFC (Nacionalidades).

Participación.

Para enviar la información al sistema de filtrado de nombres, desarrollé procedimientos almacenados en SQL Transact encargados de enviar la información de nombres a una tabla central, para que posteriormente un sistema web el cual desarrollé en Java, procedería a tomar toda esta información, para el envío masivo al sistema de filtrado de nombres, el cual tras realizar su comparación permitiría advertir por medio de alertas a los distintos analistas.

Declaración informativa mensual y anual de depósitos en efectivo.

Objetivo.

Realizar las modificaciones en los módulos del core bancario relacionados con el proceso de recaudación del impuesto sobre depósitos en efectivo (IDE) para la generación de las declaraciones informativas tanto mensuales como anuales en formato XML de acuerdo al XSD versión 2.0 proporcionado por la entidad reguladora. En conformidad con las reglas de la resolución Miscelánea Fiscal.

Descripción.

Se planteó realizar la modificación a la generación del archivo XML que presenta la declaración informativa mensual y anual, con el fin de cumplir con la nueva estructura y restricciones de contenidos impuestas por el XSD versión 2.0 dictadas por el SAT.

Participación.

Modifiqué el procedimiento almacenado en SQL Transact, para la extracción de información, así como el proceso de generación del archivo en formato XML con ayuda del componente XML Outputter desarrollado en Java dentro del Front de la plataforma bancaria.

Pago de servicios.

Objetivo.

Contar con el módulo de pago de servicios, el cual deberá cumplir con lo marcado en la circular única de bancos respecto al alta de cuentas de terceros y traspasos a cuentas de terceros.

Descripción.

Para implementar el pago de servicios como lo son ADT, CFE, INFONAVIT, NOVABOX y TELMEX, fue necesario adquirir los servicios de dos compañías llamadas Diestel y DataLogic, las cuales son compañías especializadas en el pago de servicios, las cuales ofrecen métodos de intercambio de información para el pago de servicios.

Para el proyecto se eligió el método vía servicio web, para lo cual se planteo que por medio de la banca electrónica se realizaría el llamado o invocación del servicio web que proporciona DataLogic para el registro del pago del servicio, para posteriormente realizar la afectación de saldos e impresión del comprobante internamente.

En cuanto a la parte de la conciliación se estableció el intercambio diario de archivos de texto plano de acuerdo a un layout previamente definido, el cual se intercambiaría por medio de FTP explícito bajo TLS, a través de un enlace dedicado entre ambas instituciones.

Participación.

Para la parte de la conciliación, desarrollé un componente el cual extrae de la base de datos la información de los pagos realizados en el día, posteriormente mediante un proceso que desarrollé, se crea un archivo que cumple con las especificaciones del layout proporcionado por el proveedor, el cual es enviado por FTP y posterior al envío se realiza la notificación vía mail. Este proceso es ejecutado por medio de un proceso calendarizado de lunes a viernes con ayuda de la herramienta crontab.

Implementación del producto tarjeta de débito.

Objetivo.

Incorporar a la institución bancaria en el esquema emisor con autorizador Payware para operar el BIN VISA débito con las siguientes las siguientes características:

- BIN débito Internacional.
- Marca VISA.
- Comercios de autorización:
 - TPV.
 - Interredes.
 - ATM.
 - Comercio electrónico.
 - Sucursal.
 - CashBack.

Descripción.

Para llevar a cabo el proyecto tarjeta de débito, fue necesario la intervención de distintas instituciones como son:

- **Institución Bancaria.**- Institución financiera la cual funge como entidad emisora de tarjetas de débito, esta entidad es la encargada de realizar las validaciones correspondientes, así como las afectaciones de saldos de las transacciones entrantes.
- **PROSA.**- Institución que asegura que las transacciones que envían las diversas instituciones a través de ATM, POS y Host, cumplan con los estándares y reglas operativas de VISA, MasterCard, UnionPay y CARNET, así como marcas privadas con el fin de que todas las transacciones sean enviadas y procesadas correctamente.

Para recibir estas transacciones, PROSA cuenta con:

- *Switch para Cajeros Automáticos (ATM).*- Enlace, recepción, identificación de destino y enrutamiento de las transacciones generadas por productos aceptados en el Sistema RED de un cajero automático propiedad del adquirente hacia el emisor de la tarjeta con la que se realiza la transacción.

El Sistema RED interconecta a todas las instituciones financieras del país y a los procesadores TSYS, FDR, Visa / Plus, MasterCard / Cirrus, UnionPay, ATH y Servibanca.

- *Switch para Terminales Punto de Venta (TPV).*- Ruteo y registro de transacciones generadas con productos de un emisor a partir de terminales punto de venta (POS) de otras instituciones que participan en el sistema como adquirentes.

PROSA cuenta con una plataforma de acceso flexible para los comercios, mediante enlaces dedicados, dial up, GPRS-CDMA y Ethernet.

PROSA interconectan más de 200 mil dispositivos punto de venta de los distintos clientes en todo el país por lo que se encuentran enlazados directamente con las principales cadenas comerciales, a través del esquema “interredes”.

- *Veryfone System.*- Empresa que ofrece soluciones de pago globales y servicios gestionados, entre sus distintas soluciones se encuentra Payware CMS, la cual es una solución para adquirentes y emisores dirigida para bancos e instituciones financieras.
- *Giesecke & Decrient.*- G&D proveedor en el ámbito de pago electrónico, el cual está en disposición de ofrecer productos como lo son tarjetas clásicas de banda y tarjetas con chip incorporado para realizar transacciones más seguras durante el pago electrónico.

Para cumplir con el proyecto fue necesario crear un plan de trabajo entre las instituciones anteriormente mencionadas, dentro de este plan se definieron etapas o fases, las cuales establecían fechas de término muy estrictas debido al esfuerzo colectivo.

Las etapas definidas fueron:

Actividades previas.- Esta primer fase consistió en la solicitud de alta del BIN (Número de Identificación del Banco) dentro de Eglobal e Interredes, estas compañías proveen procesamiento transaccional y están registradas globalmente con las redes de debito VISA, MasterCard, Cirrus, Pulse y Star.

Adicionalmente se realizaron las configuraciones de los equipo de comunicaciones, equipos de seguridad como HSM's y firewalls, asignaciones de direcciones Ip's y puertos entre las aplicaciones para los distintos ambientes.

Análisis y diseño.- Dentro de esta fase tanto en prosa como en la institución bancaria, se comenzó con la definición de la arquitectura de los sistemas, medios de comunicación, e infraestructura. Internamente se realizó el diseño de entidad relación y diseño de pantallas.

Desarrollo.- Dentro de esta etapa se dio inicio al desarrollo de los módulos de administración de tarjetas, así como el módulo de procesador de Mensajes ISO8583, el cual es el estándar que se maneja para transacciones electrónicas financieras realizadas por procesadores de tarjetas.

Dentro de las tareas que realice se encuentra la administración de los procesos de solicitud de generación de plásticos, recepción de respuestas de generación y conciliación de operaciones de punto de venta; adicionalmente desarrollé los servicios web para el cambio de estado de tarjetas y los consumos de los servicios expuestos por parte de PROSA para cambio de NIP y cambio de estado.

Certificación.- Finalizada la etapa de desarrollo, se dio inicio a la fase de certificación en la cual se contemplaron las siguientes tareas:

- Preparación de ambiente de aplicaciones y carga de información.
- Preparación de ambientes de herramientas de E/S de información.
- Preparación de ambiente de monitoreo.
- Configuración de reglas del firewall.
- Pruebas de comunicación.
- Procesamiento en línea.
 - Ventas.
 - Devoluciones.
 - Disposiciones.
- Procesamiento batch.
 - Generación de archivo de solicitud de tarjetas.
 - Carga de archivos de éxito/error.
 - Carga de archivo de 510 posiciones (conciliación).
- Prueba de servicios web
 - Cambio de NIP.
 - Cambio de estado de tarjeta.

Implementación.- Una vez cumplida satisfactoriamente la fase de certificación se dio lugar a la puesta en producción en la cual se realizaron tareas sobre los equipos productivos como el intercambio de llaves para validación del criptograma en equipos HSM, configuración en el Switch B24 de Prosa, configuración de servidores y aplicativos, configuración de reglas en firewalls, así como la configuración y calendarización de procesos batch.

En la *figura 4* se ejemplifica la interacción entre los sistemas para punto de venta y en la figura 5 se muestra la interacción entre los sistemas para cajeros automáticos una vez terminada la implementación.

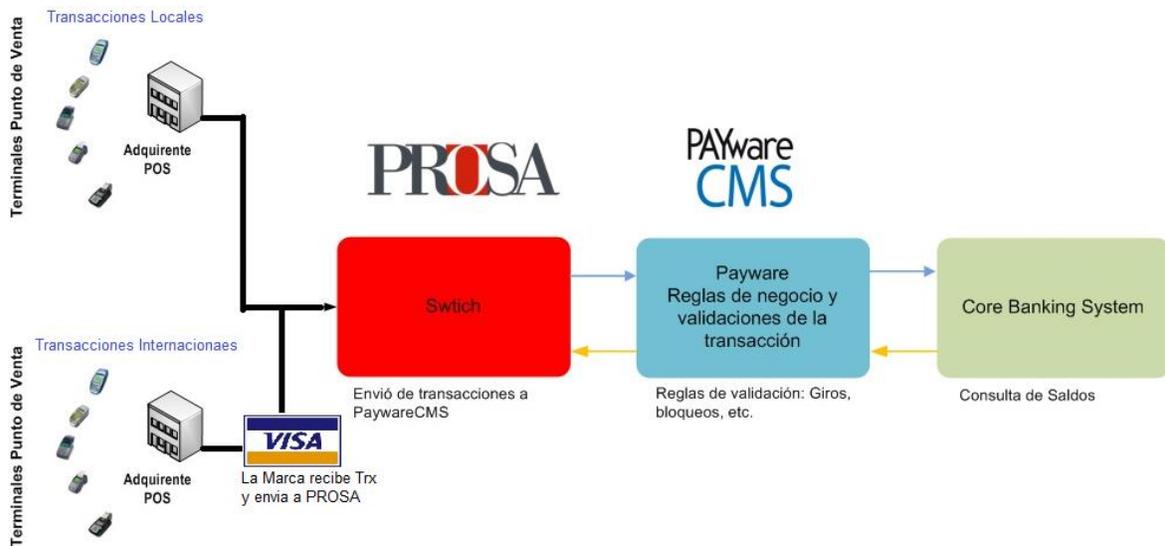


Figura 4 Esquema de conexión para Puntos de Venta.

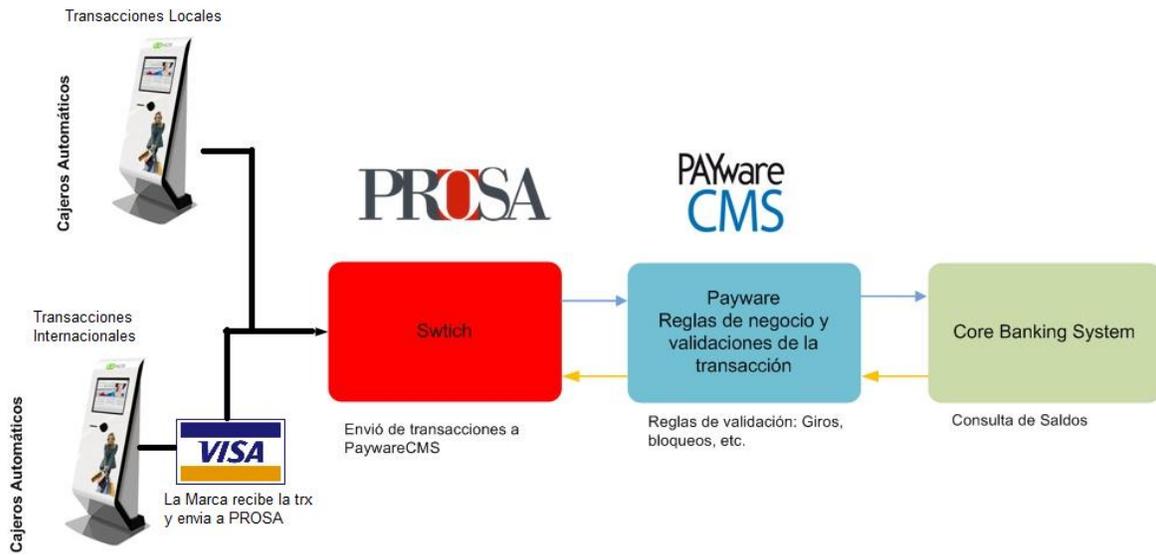


Figura 5 Esquema de conexión para ATM (Cajeros Automáticos).

CAPÍTULO 4.

Integración de sistemas tecnológicos por medio de la arquitectura orientada al servicio dentro del grupo financiero.

Objetivo.

Implementar una *Arquitectura Orientada al Servicio* SOA para reorientar los sistemas tecnológicos de la organización, logrando así una integración que facilite la creación de soluciones en menor tiempo y costo, permitiendo contar a futuro con plataformas escalables y de fácil mantenimiento.

Posibilitar el intercambio de datos en tiempo real con otros sistemas con el fin de fusionar procesos de negocio, permitiendo así la conexión de aplicaciones con distintas tecnologías, produciendo un aumento en la flexibilidad de las mismas.

Permitir la construcción de componentes de negocio intercambiables y de rápida adaptación a las necesidades específicas que se plantean a diario en una organización.

Inculcar una visión en la que se haga notar que los procesos de negocio deben ser vistos en forma de servicios y que dichos servicios forman la base para la construcción de SOA.

Antecedentes

Dentro de los antecedentes para la construcción de una SOA se pueden tomar distintas problemáticas, las cuales se pueden ver desde dos puntos de vista, las tecnológicas y las de negocio, dentro de ambos puntos de vista se describen las necesidades que generalmente se presentan antes de implementar una SOA, entre las cuales se encuentran:

- Problemáticas vistas desde el punto de vista tecnológico:
 - Mal uso de la infraestructura tecnológica.
 - Falta de estándares tecnológicos.
 - Necesidad por implementar plataformas de integración sólidas.
 - Modelos de integración mal diseñados.
 - Falta de reutilización de software.
 - Expansión descontrolada de las tecnologías de la información.

- Problemáticas vistas desde el punto de vista negocio:
 - Necesidad por modificar las aplicaciones existentes de forma rápida con el fin de soportar los cambios en los procesos y requerimientos del negocio.
 - Falta de reutilización de las aplicaciones existentes y maduras para soportar nuevos procesos sin partir de cero en la implementación.
 - Requerimientos que dictan construir nuevos proyectos más complejos que involucren más áreas y más aplicaciones.
 - Necesidad de conectar sistemas propios para el intercambio de información.

Expansión descontrolada de las tecnologías de la información como antecedente de una SOA.

Dentro de los antecedentes tecnológicos, los principales factores que son vistos para la construcción de una SOA, es el crecimiento descontrolado de los activos de TI el cual se da dentro de las organizaciones debido a factores como el crecimiento organizacional, expansión geográfica, fusiones y adquisiciones, lo cual propicia a tener una proliferación de sistemas y departamentos informáticos.

Este fenómeno repercute en problemas como:

- Conflictos de poder.
- Sucesión de proyectos fallidos.
- Retrasos en las entregas de software.
- Requisitos normativos sumamente elaborados.
- Introducción de cambios lentos y arriesgados.
- Falta de conocimiento de todos los sistemas.
- Falta de comunicación entre los distintos sistemas.

Una SOA principalmente propicia la solución a los problemas de integración tecnológica reduciendo la mayor parte de los problemas relacionados entre el negocio y los activos tecnológicos.

Marco Teórico.

Introducción a SOA.

En la actualidad la complejidad de las organizaciones es cada vez mayor. Las tecnologías avanzan hacia un modelo de organización sin barreras que se comunica a través de distintos canales de información y datos sobre un conjunto heterogéneo de sistemas de información operacionales que deben permanecer integrados para dar una respuesta conjunta a las demandas del negocio. No obstante, cada departamento o equipo de trabajo debe poder evolucionar tecnológicamente acorde a sus propias necesidades sin limitaciones impuestas por la visión integrada de la organización. Los responsables de sistemas de información deben asumir que es necesario mantener intacto el doble compromiso de evolución tecnológica para el negocio y la integración de los sistemas.

No es posible tener un único sistema operacional, pero igualmente, los sistemas no pueden permanecer aislados. Necesitan por un lado ser autónomos y por otro lado permanecer conectados. Ante este escenario, se demandan nuevas soluciones tecnológicas que permitan la continuidad del negocio sin comprometer cuestiones como la seguridad o la calidad del servicio.

La competitividad y los nuevos retos del negocio hacen indispensable la innovación tecnológica, la cual en la mayoría de los casos se traduce en la proliferación de sistemas de información que a menudo degenera en un estado de entropía tecnológica. La respuesta ante dicha entropía no puede ser la renuncia a la innovación sino que debe dar pie a abordar seriamente el problema de la integración de los nuevos sistemas con los sistemas legados, lo que se denomina comúnmente EAI o Enterprise Application Integration.

El EAI es el intercambio sin restricciones de datos y procesos de negocio entre cualquier aplicación y fuente de datos existente en la empresa. A un nivel básico, las arquitecturas de integración inicialmente implementadas en la empresa han sido:

- Punto a punto.- Cada par de sistemas se integran atendiendo únicamente a sus requerimientos particulares. FTP, RMI o Remoting han sido protocolos típicamente empleados en llevar a cabo este tipo de integración.
- Base de datos.- A un nivel muy simple, una base de datos puede establecerse como unidad central de intercambio de datos entre sistemas e implementar por tanto un modelo de integración básica.

Estas dos arquitecturas, probablemente por su simplicidad y bajo costo aparente, siguen siendo hoy día la respuesta mayoritaria al problema de la integración en la empresa. Sin embargo, a medida que la complejidad tecnológica aumenta, surge la necesidad de

mantener controladas, administradas y gestionadas dichas integraciones. Para atender a esta necesidad, han surgido nuevas respuestas al problema de la integración necesariamente más sofisticadas y ambiciosas.

Entre estas nuevas propuestas de integración se encuentra presenta la arquitectura SOA la cual es una opción robusta al momento de integrar y optimizar los activos de TI que soportan el proceso de negocio.

SOA puede transformar las organizaciones a través de un enfoque a la arquitectura orientada al servicio que abarquen los tres motores más importantes del éxito empresarial de TI: recursos humanos, procesos y tecnología.

Esta arquitectura soporta verdadero software y alineación de procesos de negocio para ayudar a darle forma a los propios procesos, en lugar de tener procesos dictados por las limitaciones de software.

Definición de SOA.

La Arquitectura Orientada al Servicio se trata de un paradigma de arquitectura de software, que define la utilización de servicios como un conjunto estructurado de tareas, las cuales contribuyen en soportar los requisitos de negocio.

SOA permite descomponer las funcionalidades que una organización requiere en partes que son reutilizables, haciendo más sencillo e independiente su manejo, lo cual permite la creación de sistemas altamente escalables que reflejan el negocio de la organización, y a su vez brinda una forma estándar de exposición e invocación de servicios, lo cual facilita la interacción entre diferentes sistemas.

Debido a que la arquitectura orientada al servicio se construye a base de servicios, es necesario definir que son los servicios.

Servicios dentro de una SOA.

Como su nombre lo indica una SOA se construye a base de servicios y dichos servicios se representan como funciones de negocios reutilizables claramente definida que pueden ser invocadas remotamente mediante protocolos de comunicación.

Los servicios cuentan con un método específico para poder usarlos, lo que se llama invocación y se caracterizan por contar con una interfaz bien definida. Un aspecto importante es la separación entre la interfaz del servicio y su implementación, es decir a la parte consumidora del servicio no necesita conocer la forma en que el servicio procesa las solicitudes.

Entre las características más importantes de un servicio se encuentran:

- Son componentes “*auto-contenidos*”, es decir realizan tareas predeterminadas.
- Son “*débilmente acoplados*”, lo que permite la reutilización, y la construcción de nuevos servicios usando los ya existentes.
- Son “*sin estado*”, es decir no dependen de la condición de ningún otro servicio ya que reciben en su llamada toda la información necesaria para regresar una respuesta.
- Son de “*granularidad gruesa*”, es decir los servicios tienden a usar un número reducido de operaciones con una complejidad relativamente grande en las operaciones que llevan a cabo.

Servicios Web.

Una vez definidos los servicios es importante hablar de la tecnología *servicios web*, la cual resulta una estrategia adecuada al momento de integrar servicios gracias a las características que presentan como la independencia de plataformas, bajo acoplamiento, auto descripción y descubrimiento. Estas características brindan grandes ventajas al momento de la integración, sin embargo cabe mencionar que no es una parte obligatoria dentro de una SOA.

Los servicios web son una tecnología que se comunica con los clientes a través de un conjunto estándar de protocolos y tecnologías. Estos estándares están implementados en las plataformas y productos de los principales proveedores de software, lo que hace de los servicios web la principal opción para la construcción de una SOA.

Entre las tecnologías y estándares que conforman los servicios web se encuentran:

- Protocolos de transporte:
 - *HTTP (Hypertext Transfer Protocol)/HTTPS (Hypertext Transfer Protocol Secure)*, es el protocolo de transferencia de hipertexto usado para el intercambio de información sobre la World Wide Web.
- Codificación de datos:
 - Protocolo *SOAP (Simple Access Protocol)*, es el protocolo de aplicación basado en mensajes que permite que una aplicación interactúe al servicio web. Estos mensajes son transmitidos en formato XML, montados sobre HTTP.
 - Esquemas *XML (DTD/XSD)*

- Descripción de interfaces o puntos de acceso a aplicaciones:
 - *WSDL (Web Services Description Language)*, es un lenguaje formato XML que se utiliza para describir la interfaz de un webservice como un conjunto de puntos finales de comunicación capaces de intercambiar mensajes.
- Descripción de servicio y descubrimiento:
 - *UDDI (Universal Description, Discovery and Integration)*, es un directorio de servicios web distribuido y basado en web que permite que se listen, busquen y descubran dichos servicios.
- Seguridad
 - *WS-Security*
 - *XML Signature*
 - *XML Encryption.*

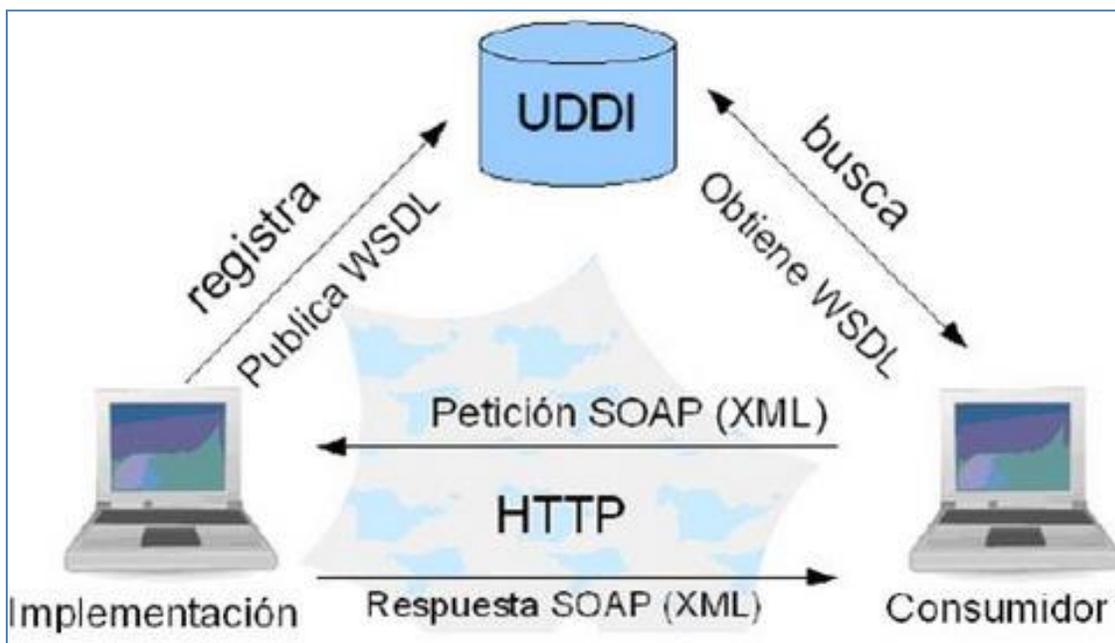


Figura 6 Servicios Web.

En la *figura 6* se muestra como la parte de implementación publica el WSDL en el registro o UDDI, para que posteriormente las partes consumidoras descubran el WSDL y por medio de peticiones y respuestas SOAP viajando por HTTP se de el consumo del servicio web.

Infraestructura de servicios (Enterprise Service Bus).

Una vez que se cuentan con los servicios detectados y la forma de integrarlos es necesario contar con un componente de infraestructura fundamental en la construcción de una SOA, el ESB *Enterprise Service Bus*, el cual es un componente que se sitúa entre los proveedores y consumidores de servicios, y tiene como funciones principales:

- Mediar mensaje entre el solicitante del servicio y el proveedor.
- Eliminar la lógica de la infraestructura, de la lógica de negocios.

En el momento de la adopción de SOA lo más probable es que ya se tengan dentro de la organización aplicaciones y sistemas los cuales son indispensables para la operación cotidiana de la organización por lo que no es posible deshacerse de dichos sistemas, sino que se puede desarrollar una capa de servicios que permita la reutilización del sistema y así obtener una enorme fuente de servicios; esto se puede lograr gracias al ESB que permite exponer las funcionalidades de los sistemas existentes a través de servicios.

Para lograr esta funcionalidad el ESB cuenta con:

- Compatibilidad con varios protocolos.
- Compatibilidad con varios patrones de respuesta.
 - *Request / Reply.*- Donde el ESB envía una consulta a la aplicación correspondiente mediante un protocolo compatible y la aplicación remite inmediatamente la respuesta.
 - *Publicación / suscripción.*- Los mensajes publicados se organizan en especies de clases, sin conocimiento de quienes puedan ser los suscriptores, estos suscriptores expresan el interés por una o más clases y sólo reciben estos mensajes, sin conocimiento de cuáles son todos los publicadores.
- Compatibilidad con diferentes formatos de mensaje.- Traducción y transformación de XML, MIME, texto plano, entre otros.
- Adaptadores.- Pueden acceder al interior de una aplicación en ejecución o acoplarse a los programas internos o llamadas a funciones y exponerlos como servicios, dichos adaptadores gestionan todos los detalles para conectar las aplicaciones existentes.

La capa de mediación de servicios permite la existencia de un débil acoplamiento entre los consumidores y proveedores de servicios, para que ambos tengan un cierto grado de libertad para cambiar y evolucionar.

Al ofrecer una infraestructura centralizada favorece que los desarrolladores se centren en la lógica del negocio, a la vez

que se reducen los costos de desarrollo de servicios y se incrementa la reutilización.

Estructura y funcionamiento del ESB.

Dentro de las principales funciones del ESB, se mencionó la mediación de servicios, la cual se consigue con la utilización de un servicio virtual.

Un servicio virtual no es un servicio real, se trata de un simple intermediario, el cual reside en la capa de mediación de servicios. Este servicio representa la interfaz deseada para los consumidores, los cuales invocan al servicio intermediario, que despacha y envía los mensajes al servicio real.



Figura 7 Virtualización de servicios.

Como resultado de la virtualización ejemplificada en la figura 7, la interfaz e implementación de servicios se ubica en dos capas diferentes. Los consumidores de servicios nunca se conectan directamente con los proveedores.

Estructura de un servicio ESB.

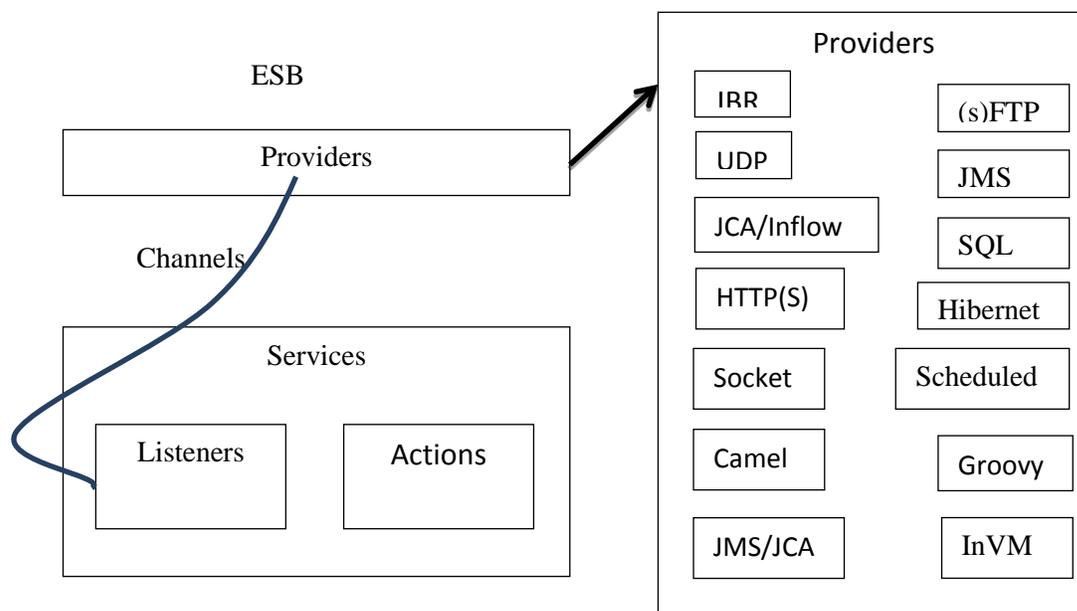


Figura 8 (Componentes internos del ESB).

Un servicio ESB está compuesto de actores que aceptan mensajes, escuchan por mensajes y actúan sobre mensajes. Estos actores son:

- **Providers.**- Es el punto de ingreso del mensaje. Estos exponen el ESB al exterior. Un *bus* o *canal*, es usado para conectar los *providers* a los *listeners*. El propio *provider* se encarga de manejar el protocolo, pasando lo que recibe a través del canal.
- **Services.**- Son una colección lógica de *listeners* y *actions* requeridas para realizar una tarea. El ESB administra estos servicios por nosotros, invocándolos como se requieran.
 - *Listeners.*- Son usados para conectar servicios a un proveedor vía el bus o el canal. Estos *listeners* pueden tener una variedad de configuraciones como escuchar un servicio en particular, una URL, o por ciertos tipos de mensajes.
 - *Actions.*- Son los componentes que realizan el trabajo final.

Providers

ESB Providers actúan como la puerta de enlace de un protocolo especificado dentro del servicio ESB. Los *providers* tienen, además de canales, propiedades asociadas a ellos, estas propiedades son pares simples de llave valor.

ESB Message Interface.

Los *providers* transforman los mensajes que están asociados con un protocolo en particular en formato XML genérico usado por el ESB, a estos mensajes se les conoce como *message interface*.

Los componentes del *message interface* son:

- *Header.*- Contiene toda la meta información acerca del mensaje.
- *Context.*- Contiene información del contexto acerca del mensaje.
- *Body.*- Contiene la carga útil del mensaje.
- *Fault.*- Contiene cualquier información relacionada con fallas.
- *Attachment.*- Actualmente en la versión del ESB JBoss no es utilizado.
- *Properties.*- Actualmente en la versión del ESB JBoss no es utilizado.

ESB Services Listeners.

Existen dos tipos básicos de *listeners*:

- Gateway Listeners.- Acepta mensajes de fuentes externas.
- ESB-aware Listeners.- Acepta mensajes sólo de otros participantes ESB-aware (nativos).

Después de aceptar mensajes, es la responsabilidad de un *listener* alimentar el mensaje en una tubería de *action* para ser procesada.

ESB Services Action.

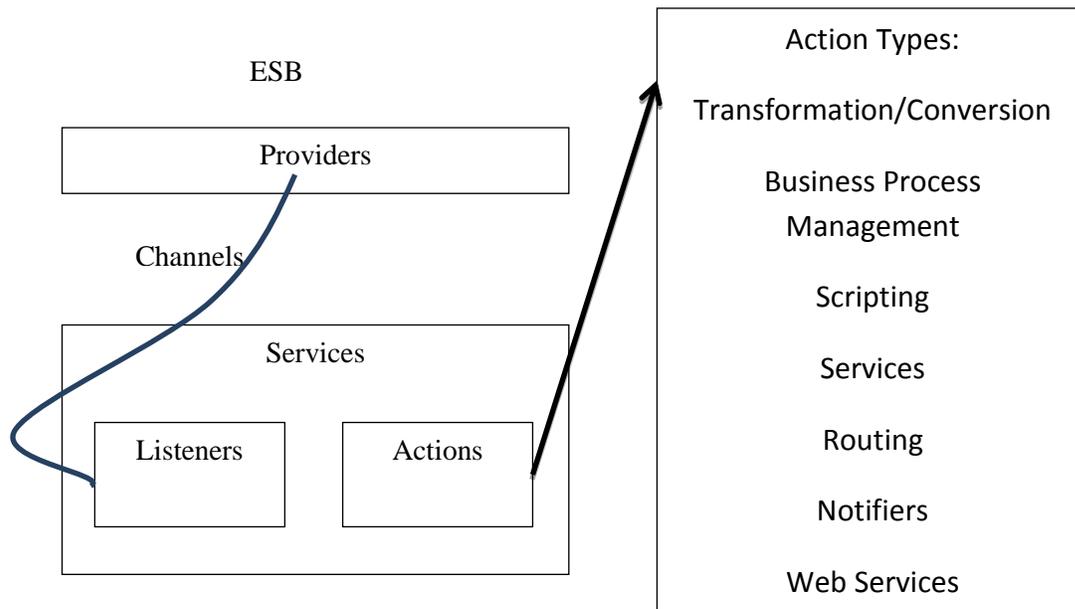


Figura 9 (Estructura del ESB y los componentes internos del Action).

El ESB cuenta con una variedad de *actions* para realizar la mayoría de las tareas comunes que se requieren para procesar un mensaje ESB.

Entre las *actions* que se encuentran en el ESB se tienen las siguientes categorías:

- *Transformation*
- *Business Process Management*
- *Scripting*
- *Services*
- *Routing*
- *Notifiers*
- *Web Services*

Gobierno de SOA.

El gobierno en una SOA es el conjunto de roles, políticas y procedimientos que sirven de guía para la adopción de la SOA. Al implementar los componentes tecnológicos de gobierno, está creando la infraestructura para soportar y aplicar estos roles, políticas y procedimientos en toda la SOA.

Una de las misiones fundamentales del gobierno SOA es garantizar que todos los participantes y sistemas se comporten del modo deseado. Por eso debe comunicar sus políticas con claridad. A partir de ahí es necesario que se apliquen estas políticas de forma consistente durante todo el ciclo de vida de su arquitectura.

Registro.

Para establecer un buen gobierno hablando en términos de la arquitectura SOA, se deberá crear un único catálogo maestro, a lo que se le conoce como registro, en el que estén visibles, para todas las partes interesadas, los elementos más importantes de una arquitectura orientada al servicio.

Cuando los servicios son publicados en el registro, el sistema puede comprobar automáticamente si cumplen los estándares de diseño especificados en el esquema global de la SOA, requisitos que facilitan el descubrimiento del servicio, documentación que describa el momento y la forma de utilizar el servicio, entre otras características.

Dentro de las mejores prácticas lo más común es que se tenga un ciclo de vida establecido en el desarrollo de software y combinar con este ciclo las políticas de SOA para crear un sistema de gobierno flexible que favorecerá cualquier tipo de colaboración, orientación y al mismo tiempo sirva para proporcionar puntos de control para verificar el cumplimiento de dichas políticas.

Vinculación entre consumidores y servicios.

Cuando un consumidor descubre en el registro un servicio que desea utilizar, inicia un proceso para vincularse con este servicio a través de un punto de acceso. Este proceso de vinculación es un lugar de control ideal para negociar políticas de ejecución entre el proveedor y el consumidor.

Antes de establecer un punto de acceso para un servicio, el consumidor, generalmente, negociará un acuerdo de nivel de servicio para establecer políticas de autenticación que aseguren que el proveedor le permitirá el uso de ese punto sólo a él.

Estos acuerdos se suelen negociar por medio del registro y se almacenan bajo la forma de un contrato, el cual define las políticas de ejecución exclusivas de las relaciones establecidas entre cada cliente con cada servicio vinculado.

Composición de servicios.

Con la utilización de SOA, las compañías pueden componer rápidamente nuevas aplicaciones y automatizar procesos de negocio que abarquen los actuales sistemas de TI. La utilización de una arquitectura SOA aumenta tanto la eficacia como la eficiencia de las operaciones de negocio sin sacrificar las inversiones que haya realizado ya la compañía para procesos y aplicaciones.

Una vez que se cuenta con los servicios reunidos en el registro, se puede comenzar con el proceso de composición de servicios, el cual consiste en conectar los servicios existentes para crear funcionalidades útiles de negocio. Una de las grandes ventajas, se presenta cuando las necesidades de negocio cambian, y del lado de TI sólo es necesario reconectar los servicios de acuerdo a las nuevas necesidades.

Procesos de negocios con BPM.

El objetivo de BPM (*Business Process Management*) es proporcionar, dentro de las TI, implementaciones automatizadas de procesos de negocio de la vida real, lo cual ofrece ventajas como las descritas a continuación:

- BPM ofrece un enfoque efectivo y dirigido por el negocio para identificar los servicios que TI debe desarrollar.
- Reduce los tiempos en la implementación de nuevos procesos de negocio.
- Permite la definición de procesos de negocio similar a un diagrama de flujo con poca o nula programación.
- Proporciona un lenguaje común que tanto el personal de TI como del negocio pueden comprender fácilmente, por lo que favorece el trabajo en conjunto.

En esta etapa es en donde se puede apreciar la reducción de tiempos y la reutilización de servicios gracias a la forma de orquestar los servicios con muy poca o nula programación, es por esto que resulta muy fácil ajustarse a las cambiantes necesidades del negocio.

Definición del problema o contexto de la participación profesional.

Dentro de la organización donde se realiza la adopción de SOA, se tienen sistemas con tecnología antigua que poseen una función crítica en relación al proceso funcional de la organización ya que dan soporte a la operación diaria del negocio, estos sistemas resultan difíciles de mantener debido a las restricciones que poseen, así como su antigua tecnológica, por lo que limita la comunicación con los nuevos sistemas adquiridos.

El intento por reescribir estos sistemas resultaría en un alto costo para la empresa ya que sería necesario asignar recursos y capital para realizar la migración de los sistemas legados por sistemas desarrollados con nueva tecnología como lenguajes de cuarta generación.

Otro de los problemas presentados surgió tras la adquisición de un banco, el cual incluía los sistemas informáticos que soportaban la operación de la institución, estos sistemas resultaron difíciles de mantener y modificar debido a factores como la complejidad de la plataforma, la dimensión de los sistemas y principalmente el desconocimiento del negocio.

Un factor que repercute gravemente en el mantenimiento o desarrollo de un sistema es el desconocimiento del negocio por parte del equipo de desarrollo, ya que existe una gran relación en el esfuerzo y tiempo que se debe dedicar en cuanto a conocimiento de las necesidades del cliente. Es decir, si el equipo de desarrollo no conoce el dominio del problema y por lo tanto no conoce sus procesos de negocio, será necesario dedicar un esfuerzo mayor en el estudio del dominio lo que repercute en un retraso en los tiempos de desarrollo de nuevos requerimientos y mantenimiento del sistema en cuestión.

Los sistemas anteriormente mencionados, tanto legados como recientemente adquiridos fueron desarrollados con lenguajes de programación distintos, características de rendimiento y diseños diferentes, por lo que resulta difícil la comunicación e interacción entre las múltiples plataformas.

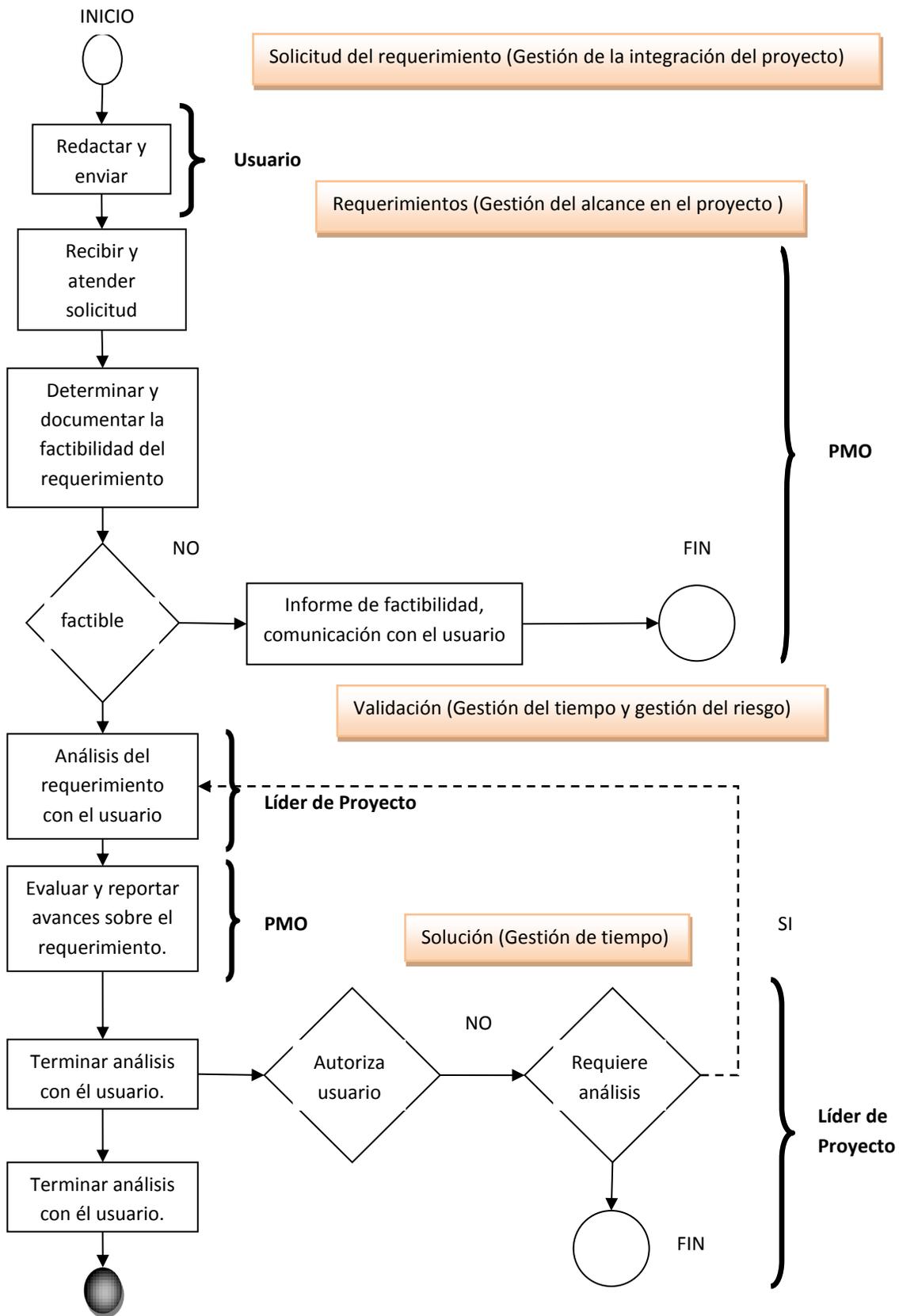
Debido a la necesidad de comunicar los sistemas internos de la empresa y la problemática anteriormente descrita se decidió en implementar un arquitectura SOA la cual ayudaría a presentar como servicios de negocio, las antiguas funciones que las capas existentes de los sistemas legados encierran, así como crear acuerdos de interoperabilidad para dejar claro cómo es que los sistemas deben comunicarse. Y por otra parte destruir las barreras organizativas con el fin de lograr una buena cooperación entre los distintos sistemas y sus equipos de desarrollo.

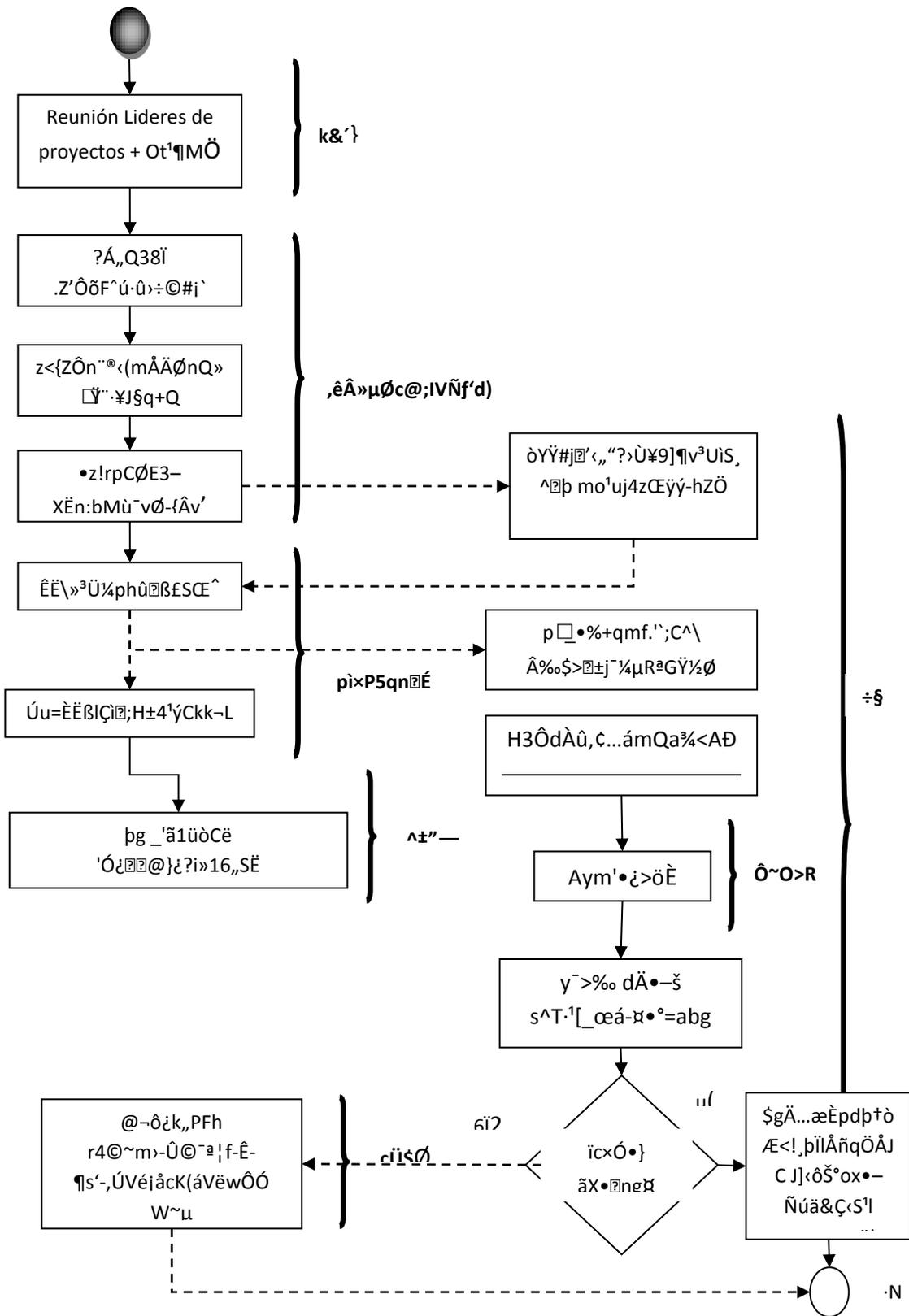
Análisis y metodología empleada.

La empresa cuenta con una oficina de gestión de proyectos también conocida como PMO (Project Management Office), la cual es la encargada del control y gestión de proyectos mediante la metodología ejemplificada en la figura 5. Para dar seguimiento a un requerimiento mediante esta metodología se inicia por la elaboración del requerimiento, en donde se realiza la redacción y envío por parte del usuario a la oficina de gestión de proyectos, una vez recibido dicho requerimiento en la oficina de gestión de proyectos, se le da el seguimiento correspondiente para determinar la factibilidad del requerimiento, en caso de ser factible, se identifica al líder de proyecto y se establece una fecha para análisis en donde participa el líder de proyecto y el usuario, con el fin de pactar una carta de entendimiento, en la cual, acuerdan ambas partes los puntos del requerimiento. Una vez terminada dicha carta, el líder de proyecto elabora la especificación de caso de uso, en donde se lleva a cabo el análisis, para lo cual es necesario la elaboración de los casos de uso, diagramas de entidad relación y diagramas de arquitectura del sistema. Cuando se tiene concluido el documento se discute con el departamento de aseguramiento de la calidad para llevar a cabo el monitoreo del desarrollo. Posteriormente el líder de proyecto elabora el plan de trabajo y notifica a su equipo de desarrollo con el fin de dar inicio al desarrollo, al mismo tiempo que se monitorea la calidad del desarrollo por parte del área de Quality Assurance.

Una vez concluida la etapa de desarrollo, se solicita el despliegue en el ambiente de pruebas para que los ingenieros tester apliquen las pruebas correspondientes, si las pruebas no presentan incidencias detectadas, se da comienzo a las pruebas de usuario. En caso de que el usuario se encuentre satisfecho y no requiera modificaciones, se formaliza con el usuario, la conclusión del requerimiento, para así finalizar mediante la metodología descrita la implementación de solución al requerimiento.

Esta metodología es utilizada para todo requerimiento solicitado por las distintas áreas de negocio, incluyendo las solicitudes provenientes del área de sistemas, por lo que el proyecto de integración de sistemas por medio de la arquitectura orientada al servicio siguió por cada etapa de la metodología.





Capitulo 4 figura 5 Metodología Interna

Una vez que se levantó el requerimiento y se determinó su factibilidad, la solicitud fue enviada al área de arquitectura ya que se identificó al arquitecto de sistemas como el líder de proyecto. Dentro del área de arquitectura el proyecto se comenzó a trabajar bajo un enfoque de desarrollo similar al enfoque en cascada en donde una vez formulados los requerimientos se inició con la etapa de planeación.

Planeación.- Durante esta fase se sostuvieron pláticas con los distintos proveedores de middleware para evaluar ventajas, desventajas y el comparativo costo beneficios con el fin de identificar al mejor candidato que satisficiera las necesidades de la empresa.

Una vez evaluados a los distintos proveedores se eligió a *Red Hat* como el proveedor de la suite de infraestructura de middleware con lo cual se comenzó un proceso de pláticas impartidas por un experto en el tema de SOA, con el fin de capacitar conceptualmente al personal acerca del cambio que implica la adopción de la arquitectura en una organización, así como dejar claro los fundamentos y principios básicos de SOA.

Posterior a la capacitación, se realizó una evaluación de las tareas necesarias para llevar a cabo la instalación y configuración del middleware. Por lo tanto se comenzó a realizar un plan de trabajo que detallara las actividades requeridas para concluir el proyecto, así como los responsables para cada actividad.

Análisis y diseño.- Para esta fase, el departamento de arquitectura en conjunto con un experto en implementaciones SOA, comenzaron por identificar que sistemas legados podrían ser expuestos de acuerdo a las capacidades del ESB, por lo que en esta etapa se determinó que sería necesario contar con un desarrollo que expusiera distintos sistemas legados en forma de servicios con el fin de poner a prueba la infraestructura a implementar.

Por otra parte se inició con el análisis para determinar la distribución del middleware y los equipos de monitoreo, así como la forma en que ocurriría la interacción entre los sistemas legados y las nuevas capas propias de la arquitectura orientada al servicio, definiendo así las características necesarias de cada servidor que alojaría a las distintas suites de middleware.

Entre las características de los servidores que alojarían a la suite de SOA se encuentran:

- Capacidad de procesamiento de 4 cores.
- 64 Gb en memoria RAM.
- 170 Gb en disco duro.
- Interfaces de red Ethernet 100/1000
- Sistema operativo Red Hat.

Estas características se definieron para cada servidor que albergaría el middleware del la suite de SOA.

Siguiendo con la parte de la distribución e interacción del middleware, dentro de la figura 11 se muestra en la parte inferior a los distintos manejadores de base de datos, de los cuales hacen uso los sistemas legados, un nivel arriba, se puede observar la forma en que los servidores de aplicaciones (middleware) despliegan a las distintas aplicaciones; dentro de esta capa se encuentra la lógica de negocio que puede ser reutilizada y expuesta como servicios por medio del bus de servicios, que es parte de la suite de SOA Platform, para posteriormente ser orquestada y crear procesos de negocio. A su vez se muestra de forma un tanto aislada el componente conocido como JON JBoss Operation Network que permite medir y monitorear la forma en que los servicios responden dentro de la SOA Platform con el fin de poder optimizar ciertas aéreas de la arquitectura.

En la figura 12 o diagrama de arquitectura, se presenta como primera capa a los sistemas transaccionales o legados que son los sistemas usados diariamente, una capa arriba se presentan las funcionalidades de la primer capa expuestas en forma de servicios, para que dentro de la capa superior se permita la orquestación de estos servicios y así un nivel más arriba se permita la construcción de procesos de negocio robustos, para que por ultimo sean expuestos dentro del portal o el sistema, desde donde el usuario podrá gestionar las funcionalidades que el sistema provee.

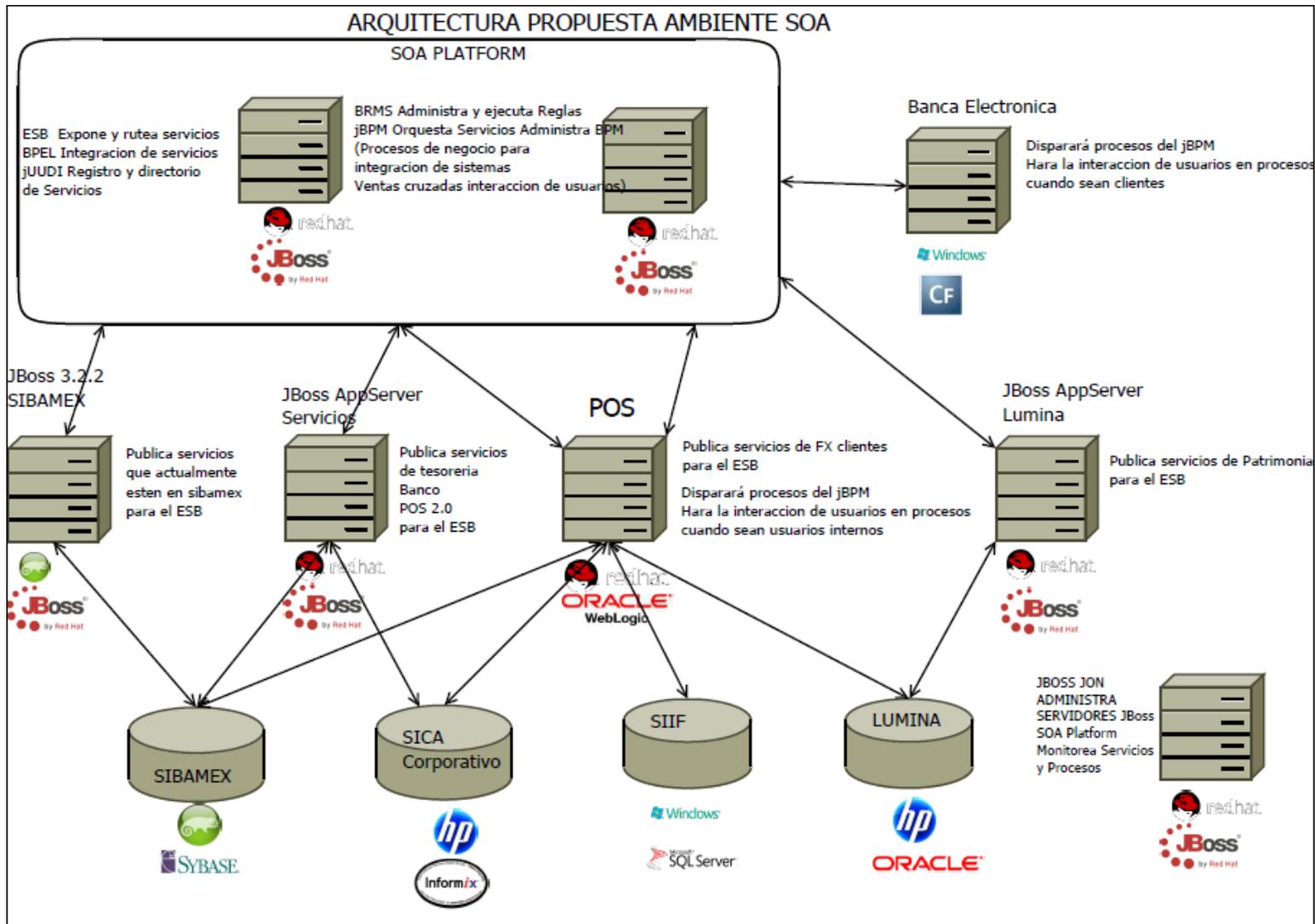


Figura 11 Diagrama de Arquitectura.

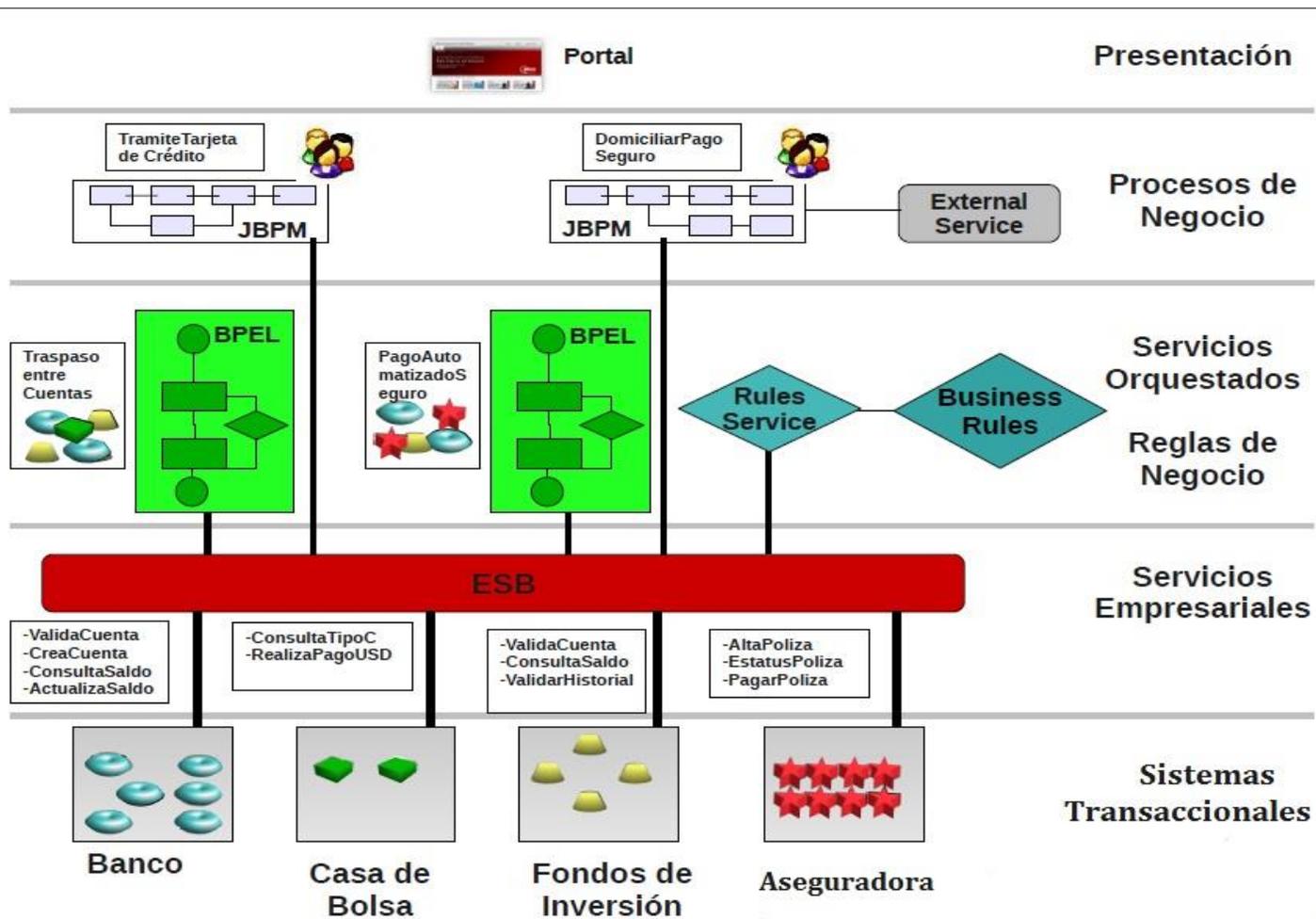


Figura 12 Diagrama de Arquitectura SOA.

Implementación.

Instalación del entorno y configuración.

Para la etapa de implementación se comenzó con la creación de los servidores virtuales mediante las herramientas de VM WARE, estos servidores se crearon de acuerdo a las especificaciones técnicas propuestas en etapas anteriores, estos servidores tendrían el fin de alojar al middleware de la suite SOA.

Una vez que se tenían creados los servidores virtuales, así como las configuraciones requeridas a nivel comunicaciones, se dio inicio con ayuda de los consultores de Red Hat a la instalación y configuración del middleware.

Implementación del primer proyecto software, bajo una arquitectura SOA.

Como se describió en la parte de análisis y diseño, se determinó un proyecto de software que pusiera a prueba la funcionalidad de la infraestructura montada, por lo que para iniciar en los primeros pasos de SOA se determinó un proyecto que uniría a los principales sistemas de la empresa. El proyecto tenía como objetivo principal la captura y registro de operaciones de liquidación bancaria por parte de promotores de casa de bolsa, dentro del sistema que es utilizado por el área de promoción, con el fin de que dichas operaciones se vieran reflejadas en el sistema bancario y así poder realizar la liquidación de estas operaciones desde las ventanillas bancarias o a través de los sistemas de mensajería y liquidación.

Una vez que se definió el proyecto, se comenzó con la etapa de análisis en la cual se realizó la definición de roles y responsabilidades de las aplicaciones, es decir, se hizo una definición para elegir qué servicios debían ser expuestos por cada aplicación o sistema.

Para el diseño del primer proyecto se comenzó con la definición de los detalles técnicos para los sistemas lógicos. Por lo cual se eligió como estrategia de integración de servicios SOA, la exposición mediante tecnología de servicios web, esto debido a que está considerado el uso de esta tecnología como una de las mejores prácticas, ya que están basados en un modelo universal de programación que facilita el intercambio de funcionalidad e información vía internet.

Para la parte del negocio se decidió utilizar tecnología Java EE, esto debido a que la mayoría de los sistemas de la empresa están construidos mediante dicha tecnología, por lo que se optó por programar las operaciones de negocio mediante a EJB'S (Componentes Java EE que encapsulan la lógica del negocio) expuestos como Servicios Web.

Para la parte de presentación se definió que el sistema de promoción sería el cual expondría las interfaces graficas de usuario mediante tecnología Flex, esto debido a que

toda la interfaz grafica se encuentra desarrollada mediante esta tecnología y una de las políticas principales de SOA es la reutilización del software.

Una vez concluido el desarrollo se comenzó la etapa de pruebas, en la cual se realizaron pruebas de unidad, pruebas de integración, así como el proceso de Quality Assurance con el fin de eliminar posibles errores.

Una vez que el proceso de calidad fue satisfactorio se dio lugar a la fase de liberación o puesta en producción del primer proyecto que pondría a prueba la infraestructura SOA.

Participación profesional.

La mayor parte de mi participación profesional se presenta en la implementación del primer proyecto de software bajo la arquitectura de SOA.

Una vez que se propuso el proyecto y se definió su objetivo, se comenzó a dar seguimiento de acuerdo a la metodología establecida dentro de la empresa, para lo cual se le dio seguimiento mediante las siguientes solicitudes:

1. Solicitud del Requerimiento (Gestión de la integración del proyecto).
2. Carta de Entendimiento (Documento en el que se establece un acuerdo mutuo sobre el alcance del proyecto).
3. Caso de Uso (Documento en el que se plasma el análisis y diseño del proyecto).
4. Carta de Aceptación (Documento que se emite para la aceptación de liberación del proyecto).
5. Carta de Liberación (Documento en el cual queda como evidencia la liberación del proyecto, así como se recaban las firmas necesarias las cuales permiten la liberación).

Dentro de los requerimientos de este primer proyecto que pondría en prueba la implementación de la infraestructura de SOA, se determinó contar con la participación de distintos equipos de desarrollo entre los cuales se encontraban los equipos de los sistemas:

- Banco, equipo en el que actualmente me encuentro.
- Casa de bolsa.
- Promoción o punto de venta.
- Liquidación y mensajería.

Este proyecto tenía como objetivo posibilitar a los promotores de casa de bolsa en la captura y consulta de operaciones bancarias dentro del sistema de promoción, con el fin

de ver reflejadas dichas operaciones dentro del sistema bancario para así permitir la liquidación de los puntos de ventanilla, mensajería o liquidación.

Entre las operaciones que se decidieron exponer como servicios se encuentran:

- Depósito en efectivo con abono a cuenta.
- Cargo a cuenta con salida de efectivo.
- Transferencia entre cuentas.
- Depósito SBC (Salvo Buen Cobro).
- Cobro de cheque propio.
- Depósito de cheque propio.
- Depósito de cheque tercero.

Para las operaciones listadas anteriormente, se pensó en el conjunto de estados por los cuales podrían pasar dichas operaciones durante su ciclo de vida en la aplicación, estos estados son ejemplificados en la figura 13.

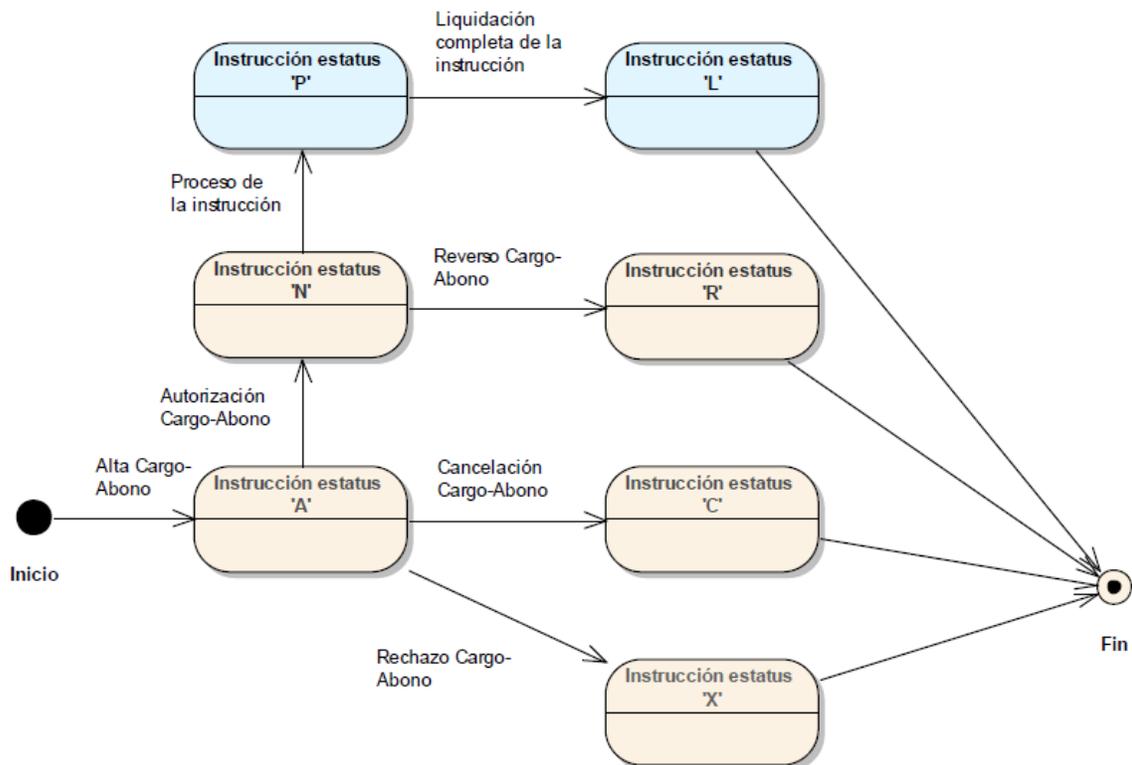


Figura 13 Diagrama de estados para las operaciones bancarias dentro del sistema de casa de bolsa.

Abono y Cargo a Intercuenta desde POS VAMOS

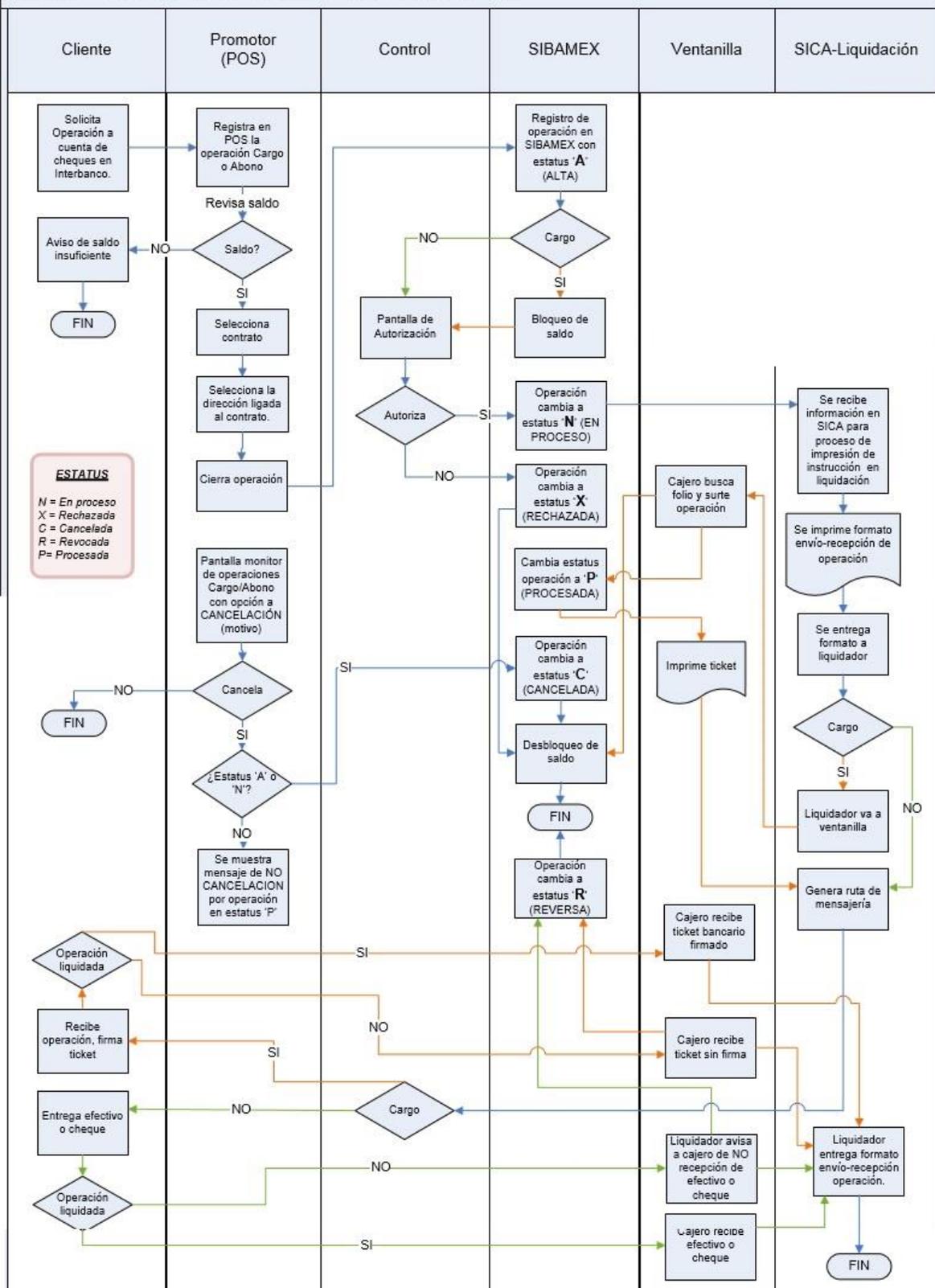


Figura 14 (Flujo de la operación)

Dentro de la figura 13 se puede observar los estados por los cuales puede pasar una operación, es decir desde el momento de su captura en un flujo ideal puede pasar de capturada o dada de alta, posteriormente autorizada y finalmente liquidada. Cuando se encuentra en un estado de alta puede pasar a dos estados más, cancelada o rechazada y cuando se encuentra en un estado de autorizada puede pasar a un estado de reversada.

Una vez definido el alcance del proyecto y aceptado, comencé a determinar en conjunto con otras aéreas, que operaciones debían ser expuestas como servicios, para lo cual realicé un análisis dentro de los procedimientos almacenados desarrollados en *SQL transact SyBase* del sistema bancario, esto debido a que la arquitectura del sistema tiene presente la lógica del negocio dentro de la capa de datos, determinando así, que las siguientes funciones existentes serian reutilizadas y convertidas a servicios web para posteriormente realizar su exposición:

- Validación de cliente.
- Validación de cuenta.
- Consulta de Saldo
- Cargo a cuenta de cheques.
- Abono a cuenta de cheques.
- Cargo contable.
- Abono contable.
- Validación de cheque.

Para estas funcionalidades detectadas, realicé el desarrollo de servicios web bajo las políticas y estándares establecidas por el área de arquitectura, para lo cual, desarrollé en lenguaje Java, componentes EJB 3.0 (Enterprise Java Beans especificación 3) que encapsulan la lógica del negocio dentro del servidor de aplicaciones, esta especificación de componentes incluye características como:

- Comunicación remota.
- Manejo de Transacciones.
- Control de concurrencia.
- Servicios de nombre y directorio.
- Seguridad.
- Ubicación de componentes en un servidor de aplicaciones.

Una vez que terminé de desarrollar los procedimientos almacenados como funciones de negocio contenidas en EJB's, debía desarrollar el código necesario para exponer estos componentes como servicios web para lo cual utilicé el framework Apache CXF, el cual me permitió crear los servicios web necesarios.

Una vez que finalicé los servicios, estos fueron expuestos en el middleware ESB o bus de servicios, para que el equipo del sistema de promoción, comenzara el desarrollo de la capa de presentación.

Se determinó que el sistema de promoción sería el sistema que proporcionaría la mayor presentación hacia el usuario debido a factores como:

- El sistema se encuentra desarrollado con tecnología Java versión 1.6 la más estable en el mercado.
- El sistema se encuentra en gran parte estandarizado.
- Es un sistema en el cual se conoce perfectamente la arquitectura, ya que fue desarrollado dentro de la empresa.
- Se cuenta con constantes capacitaciones hacia el usuario sobre la funcionalidad y utilización de sistema.
- Cuenta con una interfaz web desarrollada en Flex la cual resulta muy innovadora y fácil de utilizar.
- Utiliza en su mayoría el framework Spring.
- Está en constante innovación tecnológica.

La otra parte de presentación se presentó en las ventanillas bancarias, que se encuentran desarrolladas en tecnología Java 1.4 con interfaz swing que son aplicaciones de escritorio instaladas en cada computadora cliente que se encuentra en las ventanillas.

Dentro de estas ventanillas se llevaría a cabo la liquidación de operaciones para ciertas operaciones.

Cuando el equipo encargado de la presentación concluyó el desarrollo, cumpliendo con las fechas establecidas dentro del plan de trabajo, el desarrollo fue asignado al área de calidad o Q.A, dicha área realizó las pruebas pertinentes, con el fin determinar, que el sistema se encontraba libre de bugs, en caso de que dicha área detectara un error o fallo en el sistema que causara un resultado indeseado, se procedería a realizar un alta del bug al desarrollador implicado con el fin de repararlo lo antes posible.

Cuando el equipo de calidad determinó que el desarrollo podía pasar a producción, el área encargada de realizar la compilación tomó el desarrollo de acuerdo a la versión contralada, para realizar la liberación y puesta en producción.

CAPÍTULO 5.

Resultados.

En este capítulo mencionare los resultados obtenidos en el proyecto central de este informe, así como los obstáculos que se presentaron durante la implementación y los beneficios que se podrán alcanzar en un futuro.

Tras la implementación de SOA pude observar lo difícil que resulta adaptarse al cambio dentro de las distintas áreas de la organización, esto debido a que es necesario replantear las metodologías de trabajo, así como reorganizar los recursos; estos problemas pueden presentar un obstáculo en un comienzo, pero los resultados de la adopción serán notables con el paso del tiempo.

Durante los primeros proyectos la disminución en el tiempo de desarrollo no será tan notable ya que en un principio se tendrán registrados pocos servicios, por lo que será necesario el desarrollo de nuevos servicios, lo cual implica aumento en el tiempo de desarrollo ya que afecta en labores de búsqueda de funcionalidades que puedan ser expuestas como servicios dentro de los sistemas legados. Una vez que el registro comienza a crecer serán notables las ventajas de contar con una arquitectura SOA.

Tecnológicamente una arquitectura SOA nos permite contar con una infraestructura robusta, brindando sistemas escalables, de fácil mantenimiento, permitiendo la reutilización de software y a su vez reduciendo tiempos y costos de desarrollo. Estas características permiten cumplir con los constantes cambios que dictan las áreas de negocio, gracias a la forma de ver los procesos de negocio como servicios.

De manera generalizada podemos listar algunos de las principales ventajas y desventajas que nos ofrece una arquitectura orientada al servicio.

Ventajas:

- Aumento dramático en el retorno de la inversión reutilizando funciones comunes de negocio a través de la organización alojando estas funciones en una sola plataforma.
- Reducción de costos al minimizar el mantenimiento de sistemas y la complejidad.
- Mejor rango de visibilidad de la información a través de toda su organización.
- Óptimos procesos de negocio.
- Aumento en la reutilización de componentes de software.
- Mayor interoperabilidad entre los distintos sistemas.

Desventajas:

- Adoptar una arquitectura orientada al servicio requiere de un esfuerzo grande, lo que resulta una tarea difícil en la institución.
- La implementación de un bus de servicios empresarial se convierte en un factor crítico, lo que representaría un colapso del sistema en caso de una falla.
- La velocidad de las operaciones resulta ligeramente afectada en tiempo, a diferencia de los procesos que presentan una conexión directa.

Con estos resultados obtenidos es posible notar que SOA puede ser una arquitectura muy útil, pero siempre es necesario evaluar si el tamaño de la empresa realmente requiere una arquitectura tan robusta, ya que para implementar esta arquitectura es necesario realizar un fuerte esfuerzo y una inversión significativa. Entre otros factores que son necesarios evaluar antes de la implementación, se encuentra, la velocidad de respuesta en los sistemas, ya que debido a la adición del middleware los procesos requieren pasar por una capa extra dentro de la comunicación, lo que significa un aumento poco significativo en el tiempo de respuesta.

Conclusiones.

Laborar como desarrollador de software dentro de una empresa del sector financiero, me ha permitido adquirir experiencia y nuevos conocimientos sobre procesos de desarrollo de software más estrictos y regulados, los cuales son requeridos en sistemas de misión crítica, ya que éstos representan un impacto grande en el funcionamiento de la empresa, debido al manejo de flujos de efectivo.

Dentro de este periodo como desarrollador de software, me he percatado de la importancia de contar con sistemas que se encuentran implementados en multicapas, con arquitecturas que permiten la reutilización de componentes de software, ya que esto permite la reducción de tiempos y costos, estas características son de gran importancia en empresas del sector financiero, debido a la gran cantidad de proyectos que ingresan al área de desarrollo, los cuales son requeridos por las constantes necesidades cambiantes del negocio, así como las nuevas leyes y reglas dictadas por las distintas entidades reguladoras del sistema financiero en México.

Mi estancia dentro de esta empresa, me ha permitido comprender mejor a las instituciones financieras, ya que he estado involucrado en los distintos procesos back office que son requeridos para hacer funcionar partes fundamentales de un banco como lo son, medios de pago, tipos de productos, transferencias electrónicas, conciliaciones con proveedores e intercambio de información con instituciones diversas. Esto por mencionar algunos de los procesos con los cuales interactúo en mi labor diaria dentro del área de tecnología.

Dentro de este periodo he comprendido sobre lo importante de aplicar un enfoque sistemático y disciplinado al desarrollo, operación y mantenimiento de software, ya que esto permite contar con sistemas robustos, escalables, de fácil mantenimiento y documentados, lo cual facilita la divulgación del conocimiento a futuros desarrolladores permitiendo siempre disminuir la curva de aprendizaje entre los nuevos desarrolladores.

Todo esto me permitió aprender nuevos frameworks de trabajo para plataformas Java, los cuales presentan gran demanda en el mercado, así como distintas arquitecturas, metodologías de trabajo y procesos de negocio.

Glosario.

Adobe Flex.- Es un conjunto de tecnologías publicadas por Macromedia para dar soporte al despliegue y desarrollo de Aplicaciones Enriquecidas de Internet, basadas en su plataforma propietaria Flash.

Bug.- Es un error, defecto, error o fallo en un programa o sistema informático que hace que se produzca un resultado incorrecto o inesperado, o se comporte de forma no intencionada.

Bytes.- Unidad de información digital que consisten en ocho bits contiguos.

Caso de Uso.- Es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema, ejemplificando así la comunicación y el comportamiento de dicho sistema mediante su interacción con los usuarios y/u otros sistemas.

Core Bancario.- Plataforma donde se combinan la tecnología de la comunicación y la tecnología de la información, para satisfacer necesidades básicas de la banca.

Crontab.- Es una utilidad de software basada en planificación de tareas en sistemas operativos Linux.

DTD (Document Type Definition).- Es una definición de tipo de documento en la cual se describe la estructura de datos y sintaxis de un documento XML.

EJB (Enterprise JavaBeans).- Son una de las API que forman parte del estándar de construcción de aplicaciones empresariales J2EE actualmente JEE. Su especificación detalla cómo los servidores de aplicaciones proveen objetos desde el lado del servidor.

EAI (Enterprise Application Integration).- Se define como el uso de software y principios de arquitectura de sistemas para integrar un conjunto de aplicaciones dentro de una empresa.

ESB (Enterprise Service Bus).- Es un bus de servicios empresarial, que consiste en un combinado de arquitectura de software que proporciona servicios fundamentales para arquitecturas complejas a través de un sistema de mensajes basado en normas y que responde a eventos.

Framework.- Es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, que puede servir de base para la organización y desarrollo de software.

Hilo de procesamiento.- Tarea de procesamiento que puede ser ejecutada de forma concurrente con otras tareas y generalmente comparte recursos de forma directa.

Homonimias.- Es la cualidad de dos palabras, de distinto origen y significado por evolución histórica, que tienen la misma forma, es decir, la misma pronunciación o la misma escritura.

Impuesto IDE.- El IDE es un impuesto que se aplica al importe excedente de 15,000 pesos, en los depósitos en efectivo realizados en las instituciones del sistema financiero, ya sea por uno o por la suma de varios depósitos en el mes.

Java EE.- Java Platform, Enterprise Edition o Java EE (anteriormente conocido como Java 2 Platform, Enterprise Edition o J2EE hasta la versión 1.4; traducido informalmente como Java Empresarial), es una plataforma de programación parte de la Plataforma Java para desarrollar y ejecutar software de aplicaciones en el lenguaje de programación Java. Permite utilizar arquitecturas de N capas distribuidas y se apoya ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones.

Listas Negras.- Lista de personas, instituciones u objetos que deben ser discriminados en alguna forma con respecto a los que no están en la lista.

MIME (Multipurpose Internet Mail Extensions).- Extensiones multipropósito de correo de internet, son una serie de convenciones o especificaciones dirigidas al intercambio a través de Internet de todo tipo de archivos (texto, audio, vídeo, etc.) de forma transparente para el usuario.

Middleware.- Es un software que asiste a una aplicación para interactuar o comunicarse con otras aplicaciones, software, redes, hardware y/o sistemas operativos.

PEPS (Personas Políticamente Expuestas).- Son aquellas personas que desempeñan o han desempeñado funciones públicas destacadas en un determinado país, así como sus más allegados familiares.

PMO (Project Management Office).- Es una oficina de gestión de proyectos, departamento o grupo que define y mantiene estándares de procesos, generalmente relacionados a la gestión de proyectos, dentro de una organización.

Proceso Batch.- Ejecución de una serie de tareas en una computadora sin la intervención manual.

Protocolo de Comunicación.- Es un conjunto de reglas y normas que permiten que dos o más entidades de un sistema de comunicación se comuniquen entre ellos para transmitir información por medio de cualquier tipo de variación de una magnitud física.

Q.A (Quality Assurance).- El aseguramiento de la calidad es el conjunto de actividades planificadas y sistemáticas aplicadas en un sistema de calidad para que los requisitos de calidad de un producto o servicio sean satisfechos.

Red Hat.- Proveedor mundial en soluciones de código abierto, usando un enfoque impulsado por la comunidad para entregar soluciones confiables y de alto rendimiento en almacenamiento, virtualización, Sistemas Operativos Linux y tecnologías middleware.

RFC (Registro Federal de Contribuyentes).- Es una clave que requiere toda persona física o moral en México para realizar cualquier actividad económica lícita por la que esté obligada a pagar impuestos.

SAT (Servicio de Administración Tributaria).- Es un órgano desconcentrado de la Secretaría de Hacienda y Crédito Público, que tiene la responsabilidad de aplicar la legislación fiscal y aduanera, con el fin de que las personas físicas y morales contribuyan proporcional y equitativamente al gasto público; de fiscalizar a los contribuyentes para que cumplan con las disposiciones tributarias y aduaneras; de facilitar e incentivar el cumplimiento voluntario, y de generar y proporcionar la información necesaria para el diseño y la evaluación de la política tributaria.

Sistemas Legados.- Son los sistemas antiguos que poseen una función crítica en relación al proceso funcional de una organización que sufrirán diversas modificaciones de requisitos durante su tiempo de vida, ya que también operan en un hardware antiguo en específico, con diversas limitaciones. Los sistemas legados están caracterizados por una o más de las siguientes propiedades, que fueron implementados hace muchos años, su tecnología está obsoleta, su estructura está deteriorada, representan una gran inversión, contiene reglas de negocios ya necesarias o inservibles, no pueden ser remplazadas fácilmente.

Sistemas de misión Crítica.- Aquellos servidores que ejecutan aplicaciones esenciales que, si fallan, tienen un impacto significativo en el funcionamiento de cualquier empresa.

SOA (Service Oriented Architecture).- Arquitectura Orientada al Servicio, es un paradigma de arquitectura para diseñar y desarrollar sistemas distribuidos.

Socket.- Es una instancia de punto final, definido por una dirección IP y un puerto en el contexto de cualquiera conexión TCP.

Stored Procedure.- Un procedimiento almacenado es un programa el cual es almacenado físicamente en una base de datos. Su implementación varía de un gestor de bases de datos a otro. La ventaja de un procedimiento almacenado es que al ser ejecutado, en respuesta a una petición de usuario, es ejecutado directamente en el motor de bases de datos, el cual usualmente corre en un servidor separado. Como tal, posee acceso directo a los datos que necesita manipular y sólo necesita enviar sus resultados de regreso al usuario, deshaciéndose de la sobrecarga resultante de comunicar grandes cantidades de datos salientes y entrantes.

SyBase.- Sybase Inc. es una compañía de software principalmente conocida por su base de datos relacional Adaptive Server Enterprise Sybase.

TI.- Son aquellas herramientas y métodos empleados para recabar, retener, manipular o distribuir información. La tecnología de la información se encuentra generalmente asociada con las computadoras y las tecnologías afines aplicadas a la toma de decisiones.

XML (Extensible Markup Language).- Lenguaje de Marcas Extensible, es un lenguaje desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible.

XSD.- Es un lenguaje de esquema utilizado para describir la estructura y las restricciones de los contenidos de los documentos XML de una forma muy precisa, más allá de las normas sintácticas impuestas por el propio lenguaje XML.

Referencias.

Bibliográficas.

- I. Miko Matsumura, Bjoern Brauel y Jignesh Shah. (2009). Adopción de SOA para Dummies, edición especial de Software AG. Welly Publishing, Inc. Indianápolis Indiana.
- II. Red Hat, Inc. (2011). JBoss SOA: ESB Service Implementation Release 1-20110223
- III. Thomas Erl, SOA Principles of Services Design, Prentice Hall / PearsonPTR.
- IV. SOAction Arquitectura Orientada a Servicios, SIS.
- V. O'Brien Liam, Merson Paulo, Bass Len. Quality Attributes for Service-Oriented Architectures, International Workshop on Systems Development in SOA Environment 2007. IEEE Computer Society.

Digitales.

I. Portal.

<http://portal.prosa.com.mx/http://access.prosa.com.mx:7070/portal/dt>

Descripción.

Portal de la empresa PROSA, la cual ofrece productos y servicios sobre medios de pago.

Fecha de consulta.

15/03/2014

II. Portal.

<http://www.gi-de.com/mex/es/index.jsp>

Descripción.

Portal de la empresa Giesecke & Devrient, la cual ofrece productos y soluciones tecnológicas para industrias de distintos sectores.

Fecha de consulta.

15/03/2014

III. Portal.

<http://global.verifone.com/>

Descripción.

Portal de la empresa VeriFone, la cual ofrece soluciones y servicios sobre medios de pago, seguridad, procesamiento de pago y herramientas de negocio.

Fecha de consulta.

15/03/2014

IV. Portal.

<http://lac.visa.com/home.jsp?country=mx>

Descripción.

Portal de de la red de comercios VISA.

Fecha de consulta.

15/03/2014

V. Portal.

<http://www.tecsisa.com/>

Descripción.

Portal dedicado a temas sobre la arquitectura orientada al servicio.

Fecha de consulta.

15/03/2014

P