Universidad de la República
Facultad de Ingeniería
InCo

Proyecto de Taller 5

PROTOCOLO BASADO EN XML PARA ENVÍOS DE COMERCIO ELECTRÓNICO EN FORMA SEGURA

A/c Anselmo Martínez *2.764.413-8*
A/c Pablo Nardone *1.661.180-9*
A/c Salvador Tercia *1.993.747-2*

Indice

Indice de Anexos

Presentación de las partes involucradas

Empresa

El presente trabajo es desarrollado para la empresa pública Administración Nacional de Correos (ANC)

Responsables

Por parte de la ANC

Tutor Inicial:
Ing.Marcelo Bagnulo – Gerente de División Servicios Electrónicos – ANC (Ingeniero Eléctrico perfil Telecomunicaciones – Docente grado 2 del Instituto de Ingeniería Eléctrica – Cátedra Redes de Datos)

Tutor Final:
A/C Fernanda Burgeño - Gerente de División Servicios Electrónicos – ANC

Otros Responsables:
Ing.Federico Piedrabuena
Ing.Julio Pérez

Por parte de la Facultad de Ingeniería

Tutor:
Prof. Alfredo Viola

Integrantes del grupo

A/C Anselmo Martinez
A/C Pablo Nardone
A/C Salvador Tercia

Prefacio

El e-commerce ha revolucionado el comercio a nivel mundial, impactando en el desarrollo económico e imponiendo una nueva forma de hacer negocios, pasando a ser una necesidad indispensable para la competitividad de las empresas. Muchas empresas han comprendido la importancia del acceso a este mercado global y pretenden vender sus productos en Internet, para ello necesitan de una infraestructura de distribución para poder realizar la entrega de dichos productos a sus compradores. En este contexto es que surge la oportunidad de la Administración Nacional de Correos (ANC), al brindar el servicio de distribución, permitiendo a las empresas la tercerización del mismo.

Para lograr su inserción en Internet como proveedor de servicios de entrega (delivery) de mercadería, la ANC necesita contar con un sistema de solicitud de envíos de paquetes a través de Internet. El sistema debe permitir la comunicación y la negociación de la entrega entre un Proveedor que vende sus productos a través de Internet y la ANC que ofrece el mencionado servicio. Para lograr dicha comunicación y negociación es necesario un protocolo de capa de aplicación que defina la lógica del intercambio de información entre las partes para realizar transacciones de solicitud de servicio de delivery.

Organización del Informe

Este punto tiene por objeto brindarle al lector una descripción de la estructura general del informe para que adquiera una visión global, previo al desarrollo del mismo. El informe se estructura en diferentes capítulos, al comienzo de cada uno de ellos se presenta una breve introducción de su contenido para que el lector conozca de antemano que aspectos serán tratados en dicho capítulo. A continuación se describen los ocho capítulos que lo componen:

El Capítulo1 Introducción tiene por objeto iniciar al lector en la temática del e-commerce (comercio electrónico), destacando el rol que cumple la ANC en el mismo, y la consecuente motivación de proponer e iniciar el presente proyecto. El resto del capítulo es de lectura opcional, dedicado a destacar la importancia de la seguridad en este tipo de emprendimientos, dando una introducción a las posibles amenazas, su impacto y la aplicación de la criptografía como solución.

En el Capítulo 2 Objetivos del Proyecto se describe – desde la perspectiva de la ANC - la problemática que se pretende abordar y los resultados esperados.

El Capítulo 3 Descripción General de la Solución describe en forma general la solución alcanzada, mostrando sus características y arquitectura, así como los requerimientos cumplidos. La lectura de este capítulo es indispensable para lograr una comprensión global del problema y la solución propuesta.

Los siguientes dos capítulos son de lectura opcional, ellos extienden la comprensión lograda en el Capítulo 3, profundizando en los detalles de la solución.

El Capítulos 4, Aspectos Relativos a la Especificación del Protocolo, describe el tipo de protocolo necesario a la problemática planteada, los estudios realizados sobre la aplicación de los protocolos existentes, así como los desafíos encontrados en la realización del protocolo finalmente diseñado, y las características alcanzadas en el mismo. Se destacan algunos aspectos de su diseño y el flujo de información definida.

En el Capitulo 5 Aspectos relativos al análisis y diseño de la Aplicación, se comenta como surgió la necesidad de una aplicación y que metodología se utilizó para el proceso de desarrollo. Se comentan también los desafíos surgidos durante el análisis y diseño de la aplicación y algunas características importantes de la solución obtenida.

El Capitulo 6, Aspectos Relativos a la Implementación, presenta una reseña de tecnologías y herramientas utilizadas, el criterio usado para su elección, así como también algunos aspectos importantes de la implementación realizada.

El Capítulo 7 Planificación y Administración comprende aspectos relativos a la planificación, la elaboración de cronogramas, plan de trabajo, su evolución, la adopción de una metodología de trabajo y otros aspectos relacionados con la gestión del proyecto. Asimismo se destaca la evolución del proyecto y el esfuerzo realizado en el mismo por intermedio de diversas gráficas.

Por último en el Capítulo 8 Conclusiones, se destacan las dificultades encontradas y desafios surgidos en el transcurso del proyecto, aspectos positivos y negativos del mismo, así como las metas alcanzadas, detallando el producto entregado y el trabajo futuro propuesto.

CAPÍTULO 1

1. Introducción

En este primer capitulo se pretende introducir al lector al comercio electrónico, su importancia y destacar la motivación de la ANC (Administración Nacional de Correos) en esta nueva manera de hacer negocios y su papel a desempeñar. También se quieren presentar algunos conceptos básicos de seguridad y destacar su importancia en el e-commerce y como lograrla, así como también la importancia y uso de estándares como SSL y XML en la solución a obtener. Por ultimo se expone la estructura del informe con una reseña de cada uno de los capítulos y del CD adjunto.

1.1 Internet y el Comercio Electrónico

El comercio electrónico (e-commerce) se refiere a transacciones comerciales realizadas a través de cualquier medio electrónico en contraposición con el intercambio físico directo, e involucra distintas tecnologías, procesos y procedimientos utilizados para automatizar dichas transacciones comerciales.

Dicha actividad comercial puede ser tanto de empresa a consumidor (B2C, Business to Consumer); en el cual los clientes finales compran productos o servicios a empresas; asi como de empresa a empresa (B2B, Business to Business o e-bussiness); en el cual las empresas compran y venden entre ellas.

La Internet se ha convertido en el principal canal para el comercio electrónico, estableciéndose sitios en ella a través de la World Wide Web. Este nuevo rol de Internet en el e-commerce es relativamente reciente y ha tenido un gran impacto, mucho mayor que el de otros canales de e-commerce que existen desde hace décadas (redes privadas de datos o intranets de corporaciones, etc), permitiendo así la implementación de negocios on-line de una manera masiva generando una verdadera revolución global.

Las actividades comerciales - en este mundo cada vez mas globalizado - han ido cambiando al aumentar el volumen de transacciones, la capacidad de los proveedores y la competitividad global, así como también las expectativas de los clientes. El comercio electrónico ha servido como medio para acompasarse a dicho cambio, impulsándolo y exponensiándolo. El e-commerce no solo es una forma de aumentar los ingresos y llegar a nuevos potenciales clientes, sino que también permite la reducción de costos de la venta, aumentar la eficiencia del servicio y la entrega del producto, acercarse más a clientes y proveedores, reducir el papeleo, los errores manuales y mejorar la calidad del servicio.

Esta nueva manera de hacer negocios revoluciona a nivel mundial el comercio, cambiando su infraestructura, necesidades logísticas, inversiones, velocidad de crecimiento, etc; creando un mercado global a través de

Internet. Estamos siendo testigos del cambio producido por esta nueva tecnología, su crecimiento masivo y su impacto en el desarrollo económico. Muchas empresas cambian totalmente dejando atrás las tradicionales estructuras y muchas desaparecen al no realizar dicha conversión. Desaparecen roles como el de ciertas empresas intermediarias tradicionales, dando paso a nuevos conceptos de intermediación; así como desaparecen también – en el proceso de conversión - ciertos papeles desempeñados en las empresas. Como consecuencia, cambian también los clientes, su mentalidad, sus costumbres y por tanto sus necesidades, la manera de comprar y hacer negocios y la respuesta esperada. Se eliminan también trabas internacionales, creando un gran mercado y permitiendo fácil participación a las distintas empresas; la mayoria de los negocios que no entren en esta modalidad, corren el riesgo de desaparecer.

Es importante destacar que el comercio electrónico es una realidad actual, no de futuro y es global, se lleva a cabo en todo el mundo liderado por USA, Japón y Europa. Actualmente se puede ver en Internet, una avalancha de ofertas de productos y servicios así como también un enorme incremento en la demanda.

El e-commerce pasa a ser entonces una herramienta estratégica para empresas de fuerte competencia, obligándolos a realizar una reingeniería en los procesos del negocio, redefinir sus objetivos de mercado e infraestructura, así como sus inversiones y estrategias para explotar las oportunidades ofrecidas por el e-commerce y evitar no desaparecer al no integrarse al mismo; permitiéndoles además ser más flexibles y eficientes en sus operaciones internas, acercarse más a sus proveedores, dar mejores y más rápidas respuestas a las necesidades y expectativas de los clientes, reduciendo costos e incrementando su competitividad, logrando también su incorporación a dicho mercado global y enfrentarse a una gran demanda de consumidores

1.1.1 El marco regional, el papel de la ANC, sus oportunidades y beneficios

Un negocio típicamente depende de otros negocios los cuales le proveen varias entradas directas o indirectas (insumos, bienes o servicios intermedios) para lograr un producto final; por ejemplo otro producto o un servicio de otra empresa que se utiliza para la confección del producto final. De aquí surge la importancia del B2B, pues automatiza estas transacciones entre las empresas en toda la cadena de valor del producto. No solamente en el proceso de fabricación de un producto, sino también en todas las demás etapas del mismo, en particular en su distribución y comercialización.

Se espera que el éxito de B2B mejore los procesos de negocio y se obtenga un mayor rendimiento de las inversiones. Investigadores de mercado predicen que las transacciones de B2B alcanzaran en unos años el trillón de dólares comparado con los 100 billones producidos por las transacciones B2C.

Destacamos entonces el inminente futuro del B2B y en particular la importancia del rol de delivery en el mismo, es decir la distribución de productos. Las empresas que venden sus productos en Internet (B2C e-commerce) necesitan hacer llegar dichos productos a sus compradores, y aquí es donde surgen las empresas intermediarias que brindan el servicio de distribución, coordinando con el vendedor (y a través de este con el comprador, o directamente con el comprador) la entrega del producto y sus condiciones, tercerizando así la distribución.

Es claro que las condiciones están dadas para la implantación del negocio de “delivery” en nuestro país, varias empresas ya figuran en Internet y Uruguaynet; muchas de ellas ya venden sus productos a través de sus Web Sites y muchas todavía no. Estas empresas comprendieron la necesidad de figurar en Internet para llegar a un mercado al que no accedían o promocionarse a través de la gran red; pero quizás aún no posean la infraestructura necesaria para afrontar la venta de productos on-line o la cantidad de demanda. La distribución (así como el mantenimiento de stock) forma parte de esa infraestructura necesaria y través de una empresa que brinde este servicio muchas micro, pequeñas y medianas empresas tendrán la posibilidad de entrar en el e-commerce y a otras les facilitará esta tarea aportándoles una significativa reducción de costos. En este contexto es que surge la oportunidad del negocio de delivery en el Uruguay.

La ANC (Administración Nacional de Correos) es la institución estatal que brinda el servicio de correo tradicional en el Uruguay y por lo tanto tiene una situación de privilegio en plaza, tanto por su renombre, como por su participación en el mercado y su amplia cobertura geográfica. Estas características le otorgan la oportunidad de incorporar el mencionado servicio de delivery convirtiéndolo en una ventaja competitiva. Su modernización con la inserción en esta modalidad de e-Business, será percibido por toda la sociedad, fomentando así el desarrollo del

e-commerce en nuestro país al brindar este servicio que permite a las empresas su inserción en esta nueva tecnología. Es importante destacar además que esta no es la primer experiencia de la ANC en servicios por Internet, actualmente ofrece servicios de Autoridad de Certificación, Track & Trace (seguimiento de paquetes) y Correonet (básicamente es un servicio en el cual se envían de manera electrónica cartas a imprimir y entregar por la ANC)

Entre los beneficios a obtener por la incorporación del servicio de delivery podemos mencionar:

- Liderar el negocio de delivery en el Uruguay
- Mayor productividad, disponibilidad 24x7
- Fomentar y permitir la adopción del comercio electrónico por parte de empresas en el Uruguay
- Lograr automatización interna, reducción de papeleo y costos operativos
- Incrementar su participación en el mercado y acceder a un mercado global
- Aumentar el volumen de transacciones e ingresos
- Aumentar la eficiencia
- Mejorar su imagen
- Posibilitar el futuro análisis estadístico del comportamiento de los clientes

1.2 La importancia de la seguridad en el Comercio Electrónico

Internet se construyo utilizando el stack de protocolos TCP/IP, y estos fueron diseñados para el intercambio de información en un ambiente de mutua confianza entre usuarios. Estas debilidades de TCP/IP, hacen a Internet vulnerable a los ataques de los hackers quienes las explotan tomando ventajas de ellas.

Cada vez que se conecta una red interna a Internet, se la expone a un peligro potencial, al ser vulnerable a un ataque. Los hackers (pirátas informáticos) atacan Internet y los dispositivos conectados a la misma, tratando de entrar en sistemas y bases de datos de información, husmear información que viaja por las rutas de la gran red, etc. Pueden dañar de varias formas, robar o dañar datos importantes, ocasionar daños en computadoras, en redes, utilizar recursos corporativos, hacerse pasar por un usuario, etc. También pueden usar los sistemas de computadoras para robar productos, dinero, software, ideas o información confidencial o realizar transacciones falsas o no autorizadas. Estos delitos pueden dañar la reputación, afectando así la percepción y confiabilidad del cliente y peligrar la existencia misma de una organización.

Por lo tanto, al realizar negocios en una red pública como Internet es crucial contar con una solución segura, confiable y transparente, de manera que la única preocupación sea el negocio en sí. El objetivo primordial es proteger la confidencialidad de la información intercambiada, así como proporcionar un ambiente seguro para la realización de transacciones a través de Internet, evitando cualquier intento de ataque.
Pero la seguridad de un sistema es inversamente proporcional a su operatividad, pues el establecer seguridad implica imponer algún tipo de restricción, y para lograrlo se utilizan mecanismos que afectan la operatividad en distintos grados o maneras. La solución óptima se obtiene al lograr el balance correcto de protección versus funcionalidad con baja complejidad y costo. Es importante a la vez, que sea compatible con todas las aplicaciones y plataformas, y sea fácil de integrar y administrar, transformando la seguridad en una ventaja competitiva

1.2.1 Posibles amenazas y su repercusión

Es importante destacar que las amenazas a una organización no siempre provienen del exterior (hackers, espías, crackers –o hackers contratados-, phrackers –o hackers a través de sistemas de comunicaciones-, competencia), en realidad la mayoría de las veces provienen de la propia organización o relacionados con la misma (empleados, ex empleados, vendedores, proveedores, etc)

1.2.1.1 Tipos de amenazas

Las siguientes son algunos tipos de amenazas más comunes:

Sniffing
Consiste en observar el contenido del tráfico que tiene lugar en la red. Los mensajes y datos (lo cual incluye e-mails, claves, etc) que viajan en Internet lo hacen a través de otras máquinas y pueden ser observados. Puede, por ejemplo, conducir al robo de información sensible o a la obtención de datos para el acceso a un sistema; atentando contra la confidencialidad de la información.

Spoofing
Consiste en hacer pasar una dirección de red (IP) por otra, es decir "disfrazar" la dirección IP. Dicho de otra manera, es burlar (spoof) un sistema haciendo creer que un paquete viene de una dirección distinta a la real. De esta manera es posible hacerse pasar por otra máquina para ingresar a un sistema (pasar por alto el filtrado de direcciones) y/o no ser fácilmente rastreado. También es usado para simular una página de un negocio legítimo y obtener por ejemplo tarjetas de crédito de clientes de manera ilegal.

Acceso no autorizado
Muchos paquetes de software de Internet, ya sea por contener vulnerabilidades o ser difíciles de configurar dejan a los sistemas sujetos a ataques, produciendo un alto porcentaje de incidentes de acceso no autorizado. Los daños que puedan causarse dependerán del tipo de acceso y el sistema al cual se accede, la mayoría de las veces los hackers acceden a un sistema solo para demostrar su logro, o la vulnerabilidad del sistema; otras veces también lo hacen para instalar algún programa con el fin de controlar luego dicho host o por ejemplo para poder realizar desde el mismo un DDoS (Distributed Denied of Service), o simplemente como escala para acceder a otro sistema y dificultar más su rastreo.

Pérdida de Integridad
No solo es fácil hacer sniffing, sino que también es posible interceptar y cambiar el contenido de un mensaje e incluso repetirlo. Sus consecuencias pueden ser desastrosas si se realiza con algún mensaje perteneciente a una transacción comercial.

Denegación de Servicio
Este es un ataque muy común hoy en día, y consiste en inundar a un host conectado a la red con solicitudes que deben ser atendidas. De esta manera el host se satura al atender estos mensajes, haciéndose imposible atender las verdaderas solicitudes. Puede afectar al negocio por quedar inactivo por horas o días, afectando tanto su imagen como sus ventas.

Utilización no autorizada de servicios y/o recursos
Aquí se hace referencia al uso no autorizado de un recurso o servicio que se brinda a través de Internet, lo cual es efectivamente un robo.

1.2.1.2 Repercusión
Las amenazas citadas en el punto anterior, pueden tener consecuencias de gran impacto en el negocio, pues pueden causar:

1. Daños a la reputación
    a. Afectar la percepción del negocio por parte del cliente (pérdida de confianza y como consecuencia de prestigio y margen competitivo)
    b. Impulsar al cliente a trasladarse a un sitio de un competidor, al no estar el propio disponible o considerársele inseguro.

2. Pérdidas Financieras
    a. Pérdida de ventas y por lo tanto ingresos causadas por la discontinuidad del servicio
    b. Pérdida por el costo de recuperación (corregir información y restablecer el servicio)
    c. Pérdida por fuga de fondos al verse afectado por fraude

3. Perdida de información confidencial o de valor (datos críticos, información de propiedad, contratos, etc que son revelados a partes no autorizadas)

4. Pérdida de secretos comerciales

5. Faltar al cumplimiento de compromisos legales o contractuales

Además pueden verse afectados por:

1. Espionaje extorsivo (al obtener acceso a información confidencial y amenazar extorsionando con el uso de la misma)

2. Divulgación de Información confidencial (lo que afectaría la credibilidad de una empresa, además de las consecuencias jurídicas.)

3. Sabotaje (al afectar una organización por propia satisfacción, sin obtener ganancias económicas)

1.2.1.3 Propiedades deseables en los mensajes intercambiados

Vistos los distintos tipos de amenazas y sus consecuencias, el objetivo es contar con mecanismos de seguridad que permitan asegurar las siguientes propiedades para los mensajes intercambiados que forman parte de las transacciones de comercio electrónico:

Confidencialidad: Solo las partes autorizadas deben poder acceder a la información transmitida (interpretar el contenido del mensaje). Esta propiedad se obtiene gracias a la encripción de los mensajes que forman parte de la transacción.

Autenticación: La identidad de las partes involucradas debe poder ser validada, es decir asegurarse mutuamente que son quien dicen ser. Esto se logra mediante el uso de certificados digitales y la generación de firmas digitales.

Integridad: Los datos que son transmitidos deben ser completos, exactos, estar protegidos contra modificaciones no autorizadas, es decir garantizar que el contenido del mensaje no fue alterado durante su transmisión entre el emisor y su receptor. Esto también es conseguido gracias al uso de firmas digitales.

No repudio: Quien origina el mensaje no puede negar luego su emisión, y por lo tanto su participación en la transacción en cuestión. Para lograrlo, se firman digitalmente los mensajes.

Control de Acceso: Los datos solo pueden ser accedidos por las partes involucradas. Esto nuevamente es logrado con la encripción de los mensajes.

Los mecanismos de seguridad que se adopten deberán evitar la exposición a las distintas amenazas. No existe un único mecanismo de seguridad que abarque todas las amenazas, pero la mayoría son cubiertas al asegurar las propiedades mencionadas, lo cual se logra con el uso de encripción.
La arquitectura de red definida (la correcta disposición y elección de firewalls y routers), la correcta configuración de los sistemas, así como la adopción de adecuadas prácticas por parte de la empresa, por ejemplo contar con un plan de contingencias y seguridad física pueden contribuir a disminuir la exposición a muchas de las amenazas.

1.2.2 La aplicación de la Criptografía

La Criptografía ha sido usada por siglos con la finalidad de proteger información sensitiva que es transmitida por un canal inseguro, de manera tal que si es interceptada por un intruso, este no pueda comprender el mensaje que se esta transmitiendo. Para ello, el mensaje (llamado texto plano) es encriptado utilizando una clave predeterminada, obteniéndose así un texto cifrado (o encriptado) el cual es enviado por el canal de comunicación. El receptor recibirá dicho mensaje y lo desencriptará (o descifrará) utilizando una clave, de manera de obtener el mensaje original.
Someramente se podría decir que “la criptografía es el arte de diseñar cifradores, y el criptoanálisis es el arte de descifrar”; y la ciencia que se ocupa de ambos se conoce como criptología.

1.2.2.1 Encripción

Básicamente, la encripción es el proceso mediante el cual se transforma (a través de una serie de pasos y técnicas) cierto texto plano en un mensaje codificado o cifrado , con la finalidad de ser ininteligible para todos excepto para su receptor. Un algoritmo criptográfico es la serie de pasos o técnicas usada para la encripción y/o desencripción. En la criptografía moderna, estos algoritmos se corresponden con funciones matemáticas que pueden ser distintas para encripción y desencripción, y la habilidad de mantener la información secreta encriptada esta dada no por el desconocimiento del algoritmo (el cual en general es ampliamente conocido), sino por la clave (usualmente un número randómico) utilizada junto con este para lograr la encripción o desencripción. Desencriptar conociendo la clave es simple; intentarlo sin conocerla es muy difícil y en algunos casos imposible para todos los propósitos prácticos.

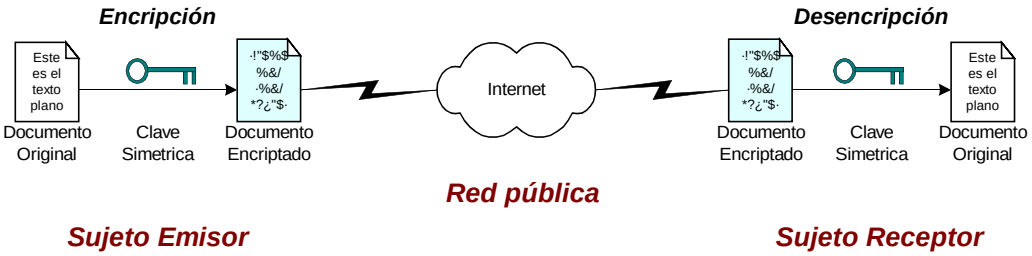
El uso de encripción permite que los datos sean transportados con un mínimo de exposición, impidiendo que un intruso interprete los datos de la red si no posee la llave de desencripción apropiada. Puesto que actualmente en Internet es imposible evitar la posibilidad de que un mensaje sea interceptado, para lograr que la información obtenida del mensaje interceptado no sea de utilidad para el intruso, debe encriptarse dicho mensaje.

Existen básicamente dos métodos de encripción actualmente en uso:

- Criptografía de clave privada o Criptografía Simétrica y
- Criptografía de clave pública o Criptografïa Asimétrica

1.2.2.1.1 Criptografía de clave privada o Simétrica

Su particularidad es que la clave de encripción puede ser calculada desde la clave de desencripción y viceversa. En la mayoría de los algoritmos simétricos, la misma clave es usada en ambos procesos.



Para la efectividad de este tipo de criptografía es crucial que la clave simétrica sea conocida únicamente por ambas partes involucradas. Si dicha clave es descubierta por una tercera persona afecta la confidencialidad y la autenticación; no sólo puede desencriptar los mensajes con dicha clave, sino que también podrá encriptar nuevos mensajes y enviarlos como si fuera alguna de las partes involucradas.

La criptografía de clave privada tiene las siguientes fortalezas:

Velocidad: Es muy rápida, permitiendo la encripción de gran cantidad de datos en poco tiempo.

Fortaleza: Tiene mucha fortaleza, haciendo difícil el "quebrar" los datos codificados. A menudo, dicha fortaleza es caracterizada por 128 bits, lo cual hace referencia al tamaño de la clave utilizada. Tamaños más pequeños son considerados débiles y son más vulnerables a ataques.

Disponibilidad: Muchas funciones matemáticas o algoritmos usados para codificar datos usando clave simétrica están disponibles gratuitamente.

A pesar de sus fortalezas, la criptografía de clave privada tiene debilidades significativas que limitan su aplicabilidad:

Distribución: Debido al uso de la misma clave para encriptar y desencriptar, es crucial disponer de un mecanismo muy seguro de distribución de claves para entornos no seguros como Internet. Por cada mensaje encriptado, debe distribuirse la clave utilizada, y debe hacerse separadamente pues si el mensaje es interceptado se compromete la seguridad de los datos contenidos. Esto hace muy costosa la distribución. Además, como las claves pueden ser hurtadas, no puede utilizarse la misma clave simétrica indefinidamente, es conveniente cambiar regularmente de claves aumentando así la sobrecarga en la distribución.

Escalabilidad: Puesto cada la clave simétrica utilizada entre un emisor y sus receptores debe ser única, el número de claves crece geométricamente con el número de usuarios. 45 claves son requeridas para una comunicación segura entre 10 usuarios, pero el numero crece a 499500 para 1000 usuarios.

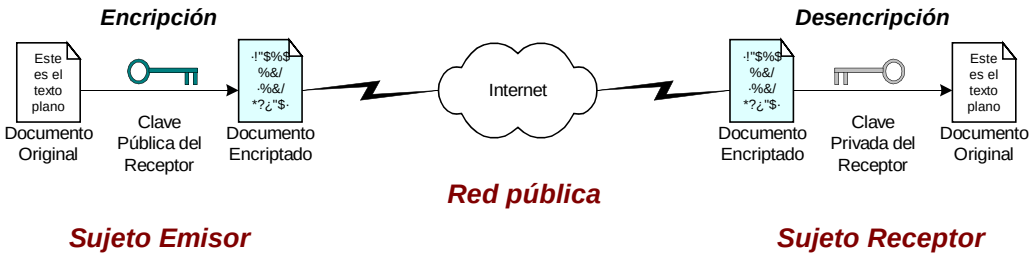
Seguridad Limitada: Las claves simétricas solo encriptan los datos y restringen su acceso, pero no proveen otros elementos de seguridad como autenticación, integridad y no repudio

Algunos algoritmos utilizados son:

- DES (Data Encryption Standard)
- Triple DES
- IDEA (International Data Encryption Algorythm)
- Blowfish
- Twofish
- RC4

1.2.2.1.2 Criptografía de clave pública o Asimétrica

En criptografía de clave pública se utilizan dos claves, una se distribuye públicamente (clave pública) y otra se mantiene privada (clave privada). Las dos claves están matemáticamente relacionadas, de manera tal que si se encriptan los datos con una de las claves, sólo pueden ser desencriptados usando la otra.



A partir de la mencionada relación entre claves, todo aquel que encripte los datos con la clave pública del destinatario y envíe los datos cifrados resultantes a dicho destinatario (dueño de la clave privada asociada), puede estar seguro que sólo este último podrá desencriptar el mensaje; siempre y cuando la clave privada en cuestión no sea conocida por nadie más. Se resuelve así el problema de distribución de claves sobre canales no seguros, que ocurría en criptografía de clave privada, surgiendo otros como el de suplantación de identidad.

Suplantación de Identidad: El problema surge al obtener la clave pública del sujeto al cual deseamos enviar el mensaje. Como estar seguros de que la clave obtenida es verdaderamente de dicho sujeto? Por ejemplo: Si un Sujeto A intenta enviar un mensaje a un Sujeto B y obtiene de su página Web su clave pública. Otro Sujeto C pudo haber sustituido la clave pública del Sujeto B con la suya propia. De esta manera, cada vez que el Sujeto A envíe un mensaje para B, C lo puede capturar y decodificar pues fue encriptado con su clave pública. Es más también puede volver a encriptar el mensaje con la clave pública de B y enviárselo para de esta manera no levantar sospechas.
Este problema se resuelve con la figura de la Autoridad Certificadora. Ellas emiten certificados digitales, a través de los cuales garantizan la autenticidad de las claves públicas de los usuarios, tanto para empresas como particulares. Su funcionamiento se describirá más adelante, en el punto relativo a Certificados Digitales.

En cuanto a su fortaleza es destacable, pero su velocidad, en comparación con los algoritmos de clave privada es menor debido a que requiere más calculo y puede no ser apropiado su uso en grandes cantidades de datos. Sin embargo, es posible el uso de encripción de clave pública para encriptar la clave simétrica a enviar, la cual luego puede usarse para encriptar el resto de los datos, aprovechando así las ventajas de ambos métodos. Este es el enfoque utilizado por el protocolo SSL (Secure Sockets Layer).

Como se verá en el punto de Firmas Digitales, basándose en la ya descripta propiedad de sus claves y aplicando el esquema inverso (encriptar con la clave privada propia el emisor y desencripte el receptor con la pública del emisor), la criptografía de clave pública puede tener otra utilidad, firmar digitalmente un documento.

El algoritmo más conocido de clave pública es el RSA (Rivest, Shamir, Adleman), pero también existen otros como PGP (Pretty Good Privacy)

1.2.2.1.3 La dependencia entre el largo de la clave y la fortaleza de la encripción

En general, la fortaleza de la encripción se refiere a la dificultad de descubrir la clave, lo cual depende del largo de la misma y del algoritmo de encripción utilizado (también podríamos decir que depende de otros factores: cómo se intercambiaron las claves, dónde y cómo se almacenan, como se obtuvieron, la confiabilidad de la procedencia del código utilizado en su elaboración y su manipulación, la confiabilidad del hardware y software del PC utilizado, etc)

La fortaleza de la encripción es normalmente descripta en términos del tamaño de la clave utilizada (medida en bits), normalmente cuanto más grande es, más fortaleza tiene. Por ejemplo, claves de 128 bits usadas con el cifrado simétrico RC4 proveen una significante protección criptográfica adicional que claves de 40 bits con el mismo cifrado.

Diferentes cifrados pueden requerir distintos largos de clave para lograr el mismo nivel de fortaleza de encripción. El cifrado RSA usado para encripción de clave pública, puede utilizar solo un subconjunto de todos los valores posibles para una clave de un largo dado (debido a la base matemática que utiliza); en cambio, otros cifrados como el de encripción de clave simétrica pueden utilizar todos los valores posibles para una clave de un largo dado. Por esta razón, el largo de 128 bits en una clave utilizada con cifrados de encripción de clave simétrica resulta proveer una encripción más fuerte que si usáramos dicho largo en una clave usada con RSA. Por lo anterior, es que el cifrado de encripción de clave pública RSA, necesita 512 bits o más para ser considerado fuerte, mientras que los cifrados de clave simétrica pueden lograr aproximadamente el mismo nivel de fortaleza con una clave de 64 bits.

1.2.2.1.4 Costo de obtener la clave

DES (56 bits)

Costo del Equipo	*Año de la Tecnología Aplicada*	*Tiempo de procesamiento*
U$S 1.000.000	1992	8 días
U$S 500.000	2000	5 días
U$S 1.000.000	2000 (con chips especiales)	1 a 2hs

RSA

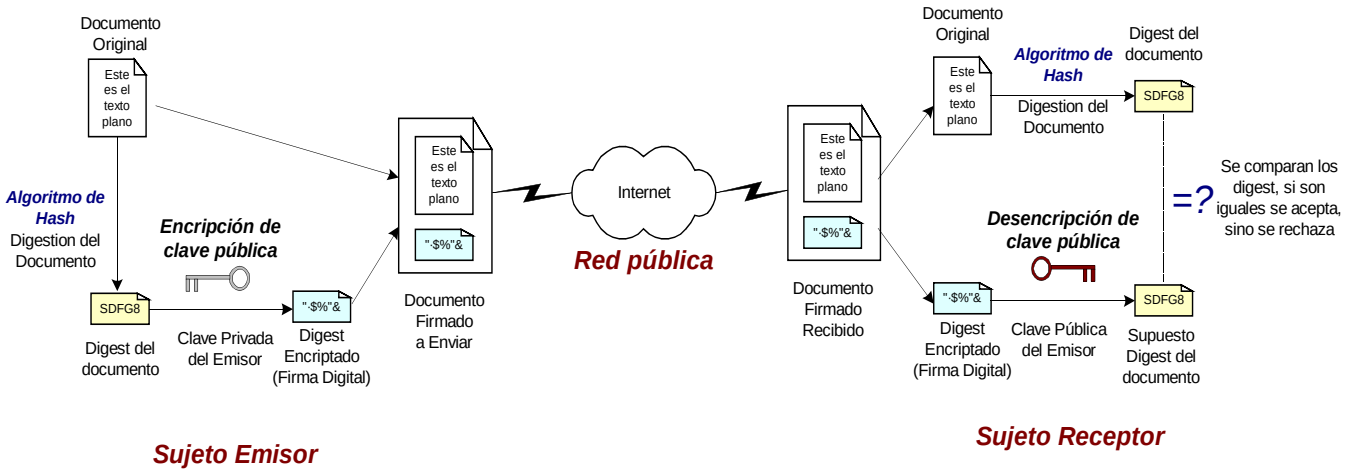
Largo de la clave (bits)	*Atacante individual*	*Red académica*	*Gran compañía*	*Agencia de Inteligencia*
274	Semanas/Días	Horas	Milisegundos	Microsegundos
384	Siglos/Décadas	Años	Horas	Segundos
512	Milenios/Siglos	Décadas	Días	Minutos
768	Inviable	Inviable	Siglos	Siglos
2048	Inviable	Inviable	Inviable	Milenios

Estas tablas nos permiten concluir - entre otras cosas – que si se usara un cifrado DES (que es uno de los más vulnerables de los que hoy en día se usan en criptografía de clave privada) para una transacción comercial,

siendo la clave elegida aleatoreamente en cada oportunidad, es suficientemente seguro dado el tiempo que puede demorar dicha transacción. El algoritmo que si debe ser fuerte es el que es empleado para el intercambio de las claves aleatorias antes mencionadas, pues es el que se utiliza siempre e identifica a las partes. De aquí surge la importancia en la elección del largo de clave utilizado en el cifrado RSA.
Es importante destacar que el avance en la capacidad de cómputo del hardware es exponencial y por lo tanto estos tiempos también se reducen con los años

1.2.2.2 Firmas Digitales

En las comunicaciones en Internet, la capacidad de anonimato y de presentarse bajo una identidad falsa es un verdadero problema. Cómo garantizar que cada una de las partes en una intercambio de mensajes es quien dice ser? Qué validez puede tener un documento correspondiente a una transacción comercial en estas condiciones?



El problema se podría intentarse resolver, por ejemplo, digitalizando la firma autógrafa real utilizada en los documentos tradicionales, pero tanto esta solución como utilizar un elemento biométrico y anexarlo al documento no soluciona el problema pues es posible copiar dicha firma de un documento y pegarla en otro. Si se realizase a través de una conexión y utilizase una password para la conformidad, esto expondría al usuario –por ejemplo - a la intercepción de su clave. La propuesta actual es el uso de Firmas Digitales para dicha autentificación, a través del uso de Certificados Digitales que identifican a cada parte, los cuales son expedidos por Autoridades Certificadoras y firmados por éstas.

La firma digital de un documento digital es un mensaje digital que se anexa como apéndice a dicho documento. La idea es que cualquiera pueda verificar que en efecto fue el emisor y no otro, quien firmó dicho documento, y que dicho documento no fue alterado de forma fraudulenta.

La firma digital surge de las relaciones matemáticas entre las claves pública y privada en la criptografía de clave pública: los datos encriptados con una de las claves, solo podrán ser desencriptados con la otra. Entonces, si el emisor (remitente) de un mensaje lo encripta con su clave privada, cualquiera que lo reciba podrá desencriptarlo con la clave pública del emisor, y de esta manera determinar la autenticidad del mismo. Además, para garantizar la integridad del contenido del mensaje y a la vez mejorar la performance en el tratamiento del mismo, se somete al mensaje un proceso adicional que consiste en generar un valor único y representativo de los datos. Este proceso es conocido como message digest (digestión del mensaje) y radica en pasar el mensaje a través de una función criptográfica irreversible (one-way hash function) –por lo cual a partir de dicho resultado no es posible, a los efectos prácticos, obtener el mensaje original – como MD5 (que produce un digest de 128 bits) o SHA-1 (cuyo digest es de 160 bits) para obtener un resultado único para cada mensaje de entrada. Es prácticamente imposible producir el mismo digest a partir de distintos mensajes de entrada, pues la probabilidad de que dos documentos tengan el mismo digest es casi cero (en el caso de MD5, $1/2^{128}$)

Entonces, luego de someter al documento original a la función de hash determinada y obtener así el digest asociado al mismo y encriptarlo con la clave privada del emisor, se obtiene la firma digital que será adjuntada a dicho documento. El destinatario del mensaje obtiene el digest del mismo al someterlo al mismo algoritmo de

hash; por otro lado, la firma digital que acompaña al documento original, se desencripta con la clave pública del emisor, obteniéndose así el digest original. El resultado de la comparación de ambos digest, indicará la integridad y autenticidad del mensaje. Cualquier modificación en algún carácter del mensaje resultará en una discrepancia en la comparación de los digest, pues al recalcular el mismo, cambiará de una manera impredecible. Por lo tanto, si coinciden, se puede estar "seguro" que el documento proviene de quien se esperaba y que no fue alterado durante su transmisión; en caso contrario, o hubo una intercepción o simplemente un error de transmisión que modificó el documento.

Como se fue indicando, a través del uso de las firmas digitales, se logran algunas de las propiedades en los mensajes intercambiados, mencionadas en el punto *Propiedades deseables en los mensajes intercambiados*; la integridad por el uso del digest obtenido a partir de la función de hash, la autenticación al encriptar el digest con algoritmos de clave pública y el no repudio debido a las propiedades de la criptografía de clave pública aplicadas al encriptar el digest.

1.2.2.2.1 Validez legal

En principio, la confiabilidad dada por una firma digital es superior a la obtenida con la firma tradicional, pues un buen falsificador puede falsificar la firma tradicional y no ser detectado, y esto no sucede con la firma digital, pues cualquier modificación en el documento es detectada.

En teoría lo anterior es cierto, pero independientemente de la base matemática de la criptografía, la firma digital es generada en una computadora y por lo tanto la confiabilidad de la generación de dicha firma depende fuertemente de la confiabilidad en dicha computadora. Por ejemplo depende del software contenido en ella y utilizado para firmar digitalmente; no se tiene certeza de lo que el software realmente este haciendo, simplemente se confía que dicho software calcule una firma digital válida y no envíe una copia de la clave privada utilizada a alguien, que luego pueda firmar mensajes en nombre del verdadero dueño de dicha clave. También dependerá de la seguridad de dicha computadora, que no haya sido hackeada e instalado una modificación del programa anterior, o un caballo de Troya para obtener la clave privada (por lo tanto depende también de dónde y como se almacenan dichas claves). Lo mismo se aplica al modo de obtención o generación de las claves públicas y privadas utilizadas.

En definitiva, según expertos, la validez jurídica de un documento firmado digitalmente puede ser cuestionada. Las firmas digitales prueban matemáticamente que la clave privada fue utilizada para generar la firma, pero no prueba que fue realizado por su dueño o que el mismo tuvo la intención de firmar el documento en concreto.

A nivel internacional

La UNCITRAL (Comisión de las Naciones Unidas para el Derecho Mercantil Internacional) tiene por objeto el fomento de reglas uniformes, tendientes a facilitar y armonizar las legislaciones de los países miembros. Esta comisión viene trabajando sobre el e-commerce desde 1985, emitiendo recomendaciones sobre el valor jurídico de los documentos digitales. También en distintos foros, organismos gubernamentales y privados, se esta discutiendo este tema con la finalidad de obtener un marco jurídico de referencia.
Más información internacional detallada y relativa a cada país en particular, puede encontrarse en http://rechten.kub.nl/simone/DS-LAWSU.HTM

Avances en el Uruguay

- Marzo de 1998: Decreto del poder ejecutivo regulando el "expediente electrónico" en el área pública.
- Setiembre de 1998: Primer autorización de Conadi a la ANC para operar como Autoridad de Certificación

1.2.2.3 Certificados Digitales

La firma digital garantiza la autenticidad del emisor y la integridad de los datos contenidos en el mensaje; pero como ya mencionamos en el punto *Criptografía de clave asimétrica (Suplantación de Identidad)*, pudo producirse una suplantación de identidad si un tercero le ha alterado la clave pública.
Para evitar este problema, existen dos modelos de confianza de claves públicas: confianza directa y confianza de tercera parte. En el primero ambas partes confían entre sí e intercambiaron las claves de una manera segura (por ejemplo personalmente). Es utilizado normalmente en compañías chicas, en las cuales puede haber un

contacto directo entre quienes intercambiarán documentos firmados. El modelo de confianza de tercera parte se da cuando los participantes no tienen una relación previa, y no confían entre sí, pero si confían ambos en una tercera parte, el cual es un intermediario para el intercambio de claves públicas. El papel de esta tercera parte es representado por las llamadas Autoridades Certificadoras, las cuales son entidades de confianza para las partes involucradas, y expiden certificados digitales que garantizan la autenticidad de las claves públicas de dichas partes.

Los certificados digitales contienen:

- el número de versión soportado del estándar X.509,
- la clave pública de la entidad o persona para la cual fue emitido,
- el algoritmo usado al generar dicha clave,
- la identificación (DN, Distinguished Name) de dicha entidad o persona (Subject),
- la identificación (DN, Distinguished Name) de la Autoridad Certificadora (CA) que garantiza su validez (Issuer),
- un identificador o número de serie del certificado (único para cada CA),
- una fecha de inicio y otra de expiración (Validity),
- otra información,
- la firma de dicha Autoridad Certificadora, y
- el algoritmo de cifrado usado para generar la firma digital
- opcionalmente puede contener extensiones (Extensions) con información adicional para el usuario, como el tipo de certificado (certificado de cliente, de servidor, para firmar email, etc)

Las claves públicas de dichas Autoridades son ampliamente conocidas (a través de la distribución de su propio certificado digital) y por esa razón no existe la posibilidad de suplantación. El estándar internacionalmente aceptado para certificados digitales es la especificación X.509 v3 de CCITT de fecha 1988 y contiene los campos mencionados anteriormente, en particular hay que destacar que como los certificados son utilizados internacionalmente, los campos de fecha estarán expresadas en notación UTC (Tiempo Universal Coordinado o Tiempo del Meridiano de Greenwich) y las identificaciones a través de DN o Distinguished Name cuyos campos se mencionan en el siguiente cuadro:

Campo del DN	**Abrev.**	**Descripción**	**Ejemplo**
Common Name	CN	*Nombre* que se esta certificando	CN=Joe Average
Organization or Company	O	El *Nombre* esta asociado con esta organización	O=Snake Oil, Ltd.
Organizational Unit	OU	El *Nombre* esta asociado con esta unidad de organización, tal como un departamento.	OU=Research Institute
City/Locality	L	Dicho *Nombre* esta ubicado en esta Ciudad o Localidad	L=Snake City
State/Province	ST	Dicho *Nombre* esta ubicado en este Estado o Provincia	ST=Desert
Country	C	Dicho *Nombre* esta ubicado en este País (codificación ISO)	C=XZ

El formato binario del certificado es definido usando notación ASN.1. La codificación binaria del certificado es definida usando Distinguished Encoding Rules (DER), la cual es basada en Basic Encoding Rules (BER). Para las transmisiones que no pueden manejar binario, puede convertirse dicho formato en un formato ASCII usando codificación Base24. Esta codificación es llamada PEM (Privacy Enhanced Mail) cuando esta entre delimitadores de línea de inicio y fin (BEGIN y END CERTIFICATE)

Un usuario que recibe un certificado digital, además de verificar la firma digital de la Autoridad Certificadora que lo expidió, deberá examinar la fecha de expiración del certificado y consultar además las listas de revocación (CRL Certificate Revocation List) publicadas por la Autoridad Certificadora correspondiente, para verificar la validez del mismo. Las listas de revocación son usadas para publicar los certificados que están comprometidos, ya sea porque expiraron, o porque el usuario comprometió su clave privada, o porque su propietario tiene una relación de dependencia con su empleador y se desvinculó de este, simplemente porque el propietario quiere desvincularse del certificado, etc.

A un documento firmado se le puede anexar un sello temporal (timestamp) para que pueda ser validado una vez que se venció el certificado.

Ejemplo de certificado digital:

```
Certificate:
    Data:
        Version: v3 (0x2)
        Serial Number: 3 (0x3)
        Signature Algorithm: PKCS #1 MD5 With RSA Encryption
        Issuer: OU=Ace Certificate Authority, O=Ace Industry, C=US
        Validity:
            Not Before: Fri Oct 17 18:36:25 1997
            Not  After: Sun Oct 17 18:36:25 1999
        Subject: CN=Jane Doe, OU=Finance, O=Ace Industry, C=US
        Subject Public Key Info:
            Algorithm: PKCS #1 RSA Encryption
            Public Key:
                Modulus:
                00:ca:fa:79:98:8f:19:f8:d7:de:e4:49:80:48:e6:2a:2a:
                86:ed:27:40:4d:86:b3:05:c0:01:bb:50:15:c9:de:dc:85:
                19:22:43:7d:45:6d:71:4e:17:3d:f0:36:4b:5b:7f:a8:51:
                a3:a1:00:98:ce:7f:47:50:2c:93:36:7c:01:6e:cb:89:06:
                41:72:b5:e9:73:49:38:76:ef:b6:8f:ac:49:bb:63:0f:9b:
                ff:16:2a:e3:0e:9d:3b:af:ce:9a:3e:48:65:de:96:61:d5:
                0a:11:2a:a2:80:b0:7d:d8:99:cb:0c:99:34:c9:ab:25:06:
                a8:31:ad:8c:4b:aa:54:91:f4:15
                Public Exponent: 65537 (0x10001)
        Extensions:
            Identifier: Certificate Type
                Critical: no
                Certified Usage:
                    SSL Client
            Identifier: Authority Key Identifier
                Critical: no
                Key Identifier:
                   f2:f2:06:59:90:18:47:51:f5:89:33:5a:31:7a:e6:5c:fb:
                   36:26:c9
    Signature:
        Algorithm: PKCS #1 MD5 With RSA Encryption
        Signature:
               6d:23:af:f3:d3:b6:7a:df:90:df:cd:7e:18:6c:01:69:8e:54:
               65:fc:06:30:43:34:d1:63:1f:06:7d:c3:40:a8:2a:82:c1:a4:
               83:2a:fb:2e:8f:fb:f0:6d:ff:75:a3:78:f7:52:47:46:62:97:
               1d:d9:c6:11:0a:02:a2:e0:cc:2a:75:6c:8b:b6:9b:87:00:7d:
               7c:84:76:79:ba:f8:b4:d2:62:58:c3:c5:b6:c1:43:ac:63:44:
               42:fd:af:c8:0f:2f:38:85:6d:d6:59:e8:41:42:a5:4a:e5:26:
               38:ff:32:78:a1:38:f1:ed:dc:0d:31:d1:b0:6d:67:e9:46:a8:
               dd:c4
```

Codificado en Base 24 (en realidad, codificación PEM) quedaría:

```
-----BEGIN CERTIFICATE-----
MIICKzCCAZSgAwIBAgIBAzANBgkqhkiG9w0BAQQFADA3MQswCQYDVQQGEwJVUzER
MA8GA1UEChMITmV0c2NhcGUxFTATBgNVBAsTDFN1cHJpeWEncyBDQTAeFw05NzEw
MTgwMTM2MjVaFw05OTEwMTgwMTM2MjVaMEgxCzAJBgNVBAYTAlVTMREwDwYDVQQK
```

```
EwhOZXRzY2FwZTENMAsGA1UECxMEUHViczEXMBUGA1UEAxMOU3Vwcml5YSBTaGV0
dHkwgZ8wDQYJKoZIhvcNAQEFBQADgY0AMIGJAoGBAMr6eZiPGfjX3uRJgEjmKiqG
7SdATYazBcABu1AVyd7chRkiQ31FbXFOGD3wNktbf6hRo6EAmM5/R1AskzZ8AW7L
iQZBcrXpc0k4du+2Q6xJu2MPm/8WKuMOnTuvzpo+SGXelmHVChEqooCwfdiZywyZ
NMmrJgaoMa2MS6pUkfQVAgMBAAGjNjA0MBEGCWCGSAGG+EIBAQQEAwIAgDAfBgNV
HSMEGDAWgBTy8gZZkBhHUfWJM1oxeuZc+zYmyTANBgkqhkiG9w0BAQQFAAOBgQBt
I6/z07Z635DfzX4XbAFpjlRl/AYwQzTSYx8GfcNAqCqCwaSDKvsuj/vwbf91o3j3
UkdGYpcd2cYRCgKi4MwqdWyLtpuHAH18hHZ5uvi00mJYw8W2wUOsY0RC/a/IDy84
hW3WWehBUqVK5SY4/zJ4oTjx7dwNMdGwbWfpRqjd1A==
-----END CERTIFICATE-----
```

Normalmente un software que maneja certificados tiene una lista de Autoridades de Certificación en las cuales confía (pues posee sus certificados). Esto determina que otros certificados se pueden validar directamente (los firmados por dichas Autoridades de Certificación, o sea expedidos por estas). Las Autoridades de Certificación pueden también expedir certificados para otras Autoridades de Certificación, entonces cuando se examina un certificado y no se confía en el directamente, puede examinarse a la Autoridad que expidió dicho certificado, y así sucesivamente continuando en dicha cadena de certificaciones hasta alcanzar una Autoridad en la cual si se confíe. Se puede limitar el “largo” de la cadena que se recorre y por lo tanto en la cual se confía.

1.2.3 Arquitecturas y aplicaciones seguras en Internet

Como ya se ha observado, la seguridad de un sistema no es completamente obtenida por el uso de la criptografía. Ya se ha destacado - por ejemplo en el punto de Firmas Digitales – que depende de la confiabilidad en el hardware y software subyacente. Entonces, además del uso de la criptografía, es necesaria una buena programación del sistema, una cuidadosa elección de las herramientas utilizadas (su procedencia y confiabilidad), tanto en el desarrollo (parsers, librerías varias, etc) como el software de base (elección del Sistema Operativo, Base de Datos, Web Server) y sus respectivas configuraciones (cuidadosa configuración e incorporación de patches); también es crucial la arquitectura de red elegida, pues disminuirá muchas de las amenazas mencionadas en puntos anteriores. Debe obtenerse una buena arquitectura combinando routers y firewalls protegiendo así la red interna de la externa; normalmente se crea una doble “muralla”, creando dos niveles de seguridad, el primero (DMZ, Zona DeMilitarizada) contiene los servidores a los que se accederá via Internet , y el segundo el resto de la red interna. Aquí es clave la asignación de IPs para la red interna y las direcciones IPs que serán vistas externamente, y la configuración de filtrado de las distintas IPs, así como el tipo de servicios habilitados. (Por ejemplo seguir RFC 1918 para asignación de IPs, no usar en red interna IPs validas en Internet, no mantener habilitados servicios no usados, etc). El uso de firewall y routers permite configurarlos para incluir filtros de ingreso (de determinadas direcciones IP) , limitación de frecuencia (de determinados tipos de paquetes p/ej SYN), búsqueda revertida de direcciones (para asegurarse de que la IP del paquete es la que dice ser) y monitoreo de trafico de red. La configuración de la red, sus routers y firewalls es un tema aparte y debe ser muy cuidadosa para que sea efectiva. Básicamente el firewall permite:

- Bloquear el acceso a sitios particulares de Internet
- Prevenir el acceso a ciertos servidores o servicios
- Monitorear las comunicaciones entre la red interna y externa
- Escuchar y registrar comunicaciones para investigar penetración a la red o detectar ataques internos
- Encriptar paquetes que han sido enviados a diferentes localidades físicas dentro de una organización.

También es conveniente contar con software de alertas contra ataques, así como revisiones periódicas de los logs de auditoría. Es recomendable poseer controles de detección de intrusión que permitan detectar archivos de sistemas cambiados (Tripwire), encontrar huellas de hackers (last, netstat, etc), detectar sniffers (check promiscuous mode), desplegar usuarios y procesos activos (ps, w, finger, who,ps, etc) y monitorear performance del sistema (COPS, SATAN, Scan Detector, etc)

Por otro lado, el contar con Alta Disponibilidad y Tolerancia a Fallas (HeFT, High e-vailability-Fault Tolerance) puede disminuir el impacto de ataques DoS al reducir el tiempo de no disponibilidad del sistema, así como servir en casos de recuperación de desastres. Esto se logra implementando espejado de archivos críticos, puntos de entrada alternativos para recibir y procesar transacciones, mecanismos para activar computador alterno inmediata y automáticamente, etc. El nivel de redundancia es una elección empresarial y dependerá de la criticidad del sistema y la importancia que tenga para el negocio, dicha redundancia puede ir desde contar con

discos RAID y mecanismos de fail-over, hasta contar con una arquitectura redundante en el caso de redes y routers, aunque consideramos que no se justifica en este caso, pues pueden reemplazarse fácilmente.

1.3 Referencias bibliográficas

[1] http://www.computer.org Computer Magazine – October 2000
(B2B e-commerce frameworks)

[2] http://www.comsoc.org IEEE Network Magazine – July/August 2000 Vol.14 No.4
(Securing Electronic Commerce)

[3] http://www.isaca.org Revista Percepciones – Octubre 2000
(HeFT, Derrotando al Ciber Criminal, Potencial del e-commerce y sus ramificaciones en la auditoria de TI, Tecnología de Firewalls, Firma Digital)

[4] http://www.isaca.org Revista Percepciones – Enero 1999

[5] http://www.isaca.org Information Systems Control Journal Volume 6, 2000
(e-Commerce, The Next Permutation)

[6] http://www.isaca.org Manual de revisión para el examen CISA

[7] http://www.modssl.org/mod_ssl-2.8.0-1.3.17/pkg.ssldoc/ssl_intro.html

[8] http://developer.netscape.com/docs/manuals/security/pkin/contents.htm

[9] http://www.entrust.com/products/pki/cryptography.htm

[10] http://www.verisign.com/server/trial/index.html

[11] http://www.verisign.com/cgi-bin/go.cgi?a=e008611090013000 (Web Security Guide)

[12] SUN@America.Latina año 2 número 2
(Comercio electrónico, Asegurando su red para negocios)

[13] SUN@America.Latina año 1999 número 3
(El comercio electrónico como objetivo)

[14] Redes de Computadoras – A.Tanenbaum

[15] Seguridad Informática – Juan José Nombela

CAPITULO 2

2. Objetivos del proyecto

La Administración Nacional de Correos (ANC), desea su inserción en Internet como proveedor de servicios de entrega (delivery) de mercadería (ver *Capítulo 1 Introducción, punto El marco regional, el papel de la ANC sus oportunidades y beneficios*). Para ello necesita contar con un sistema de solicitud de envíos de correspondencia en forma remota a través de Internet. El sistema debe permitir la comunicación y la negociación de la entrega entre un Proveedor que vende sus productos a través de Internet y la ANC que ofrece el mencionado servicio. Dicha comunicación y negociación se logrará a través de un protocolo de capa de aplicación que defina de forma correcta los mensajes de solicitud, confirmación, status y otras propiedades de los envíos. El sistema debe procesar las transacciones de solicitud de servicio de delivery, registrando las mismas en una base de datos para tal fin.

Entre otros aspectos destacados a ser cumplidos se encuentran:

- consideraciones de seguridad en lo referente al uso de firmas digitales;
- especificación de protocolo aplicable a una variedad de hardware y software (por ejemplo Windows NT y AIX),
- utilización de una base de datos SQLServer u Oracle.

En resumen, los resultados esperados por parte de la ANC son los siguientes:

- Especificación de un protocolo de solicitud de envíos estándar.
- Herramienta que implemente dicho protocolo.

Se puede destacar también como objetivo secundario, pero no menos importante, la formación ofrecida, lo cual fue una de las principales motivaciones en la elección del presente proyecto:

- Proceso de especificación de un protocolo
- Formación en protocolos de comunicación
- Diseño de esquemas en XML y signed XML
- Formación en Criptografía
- Formación en manejo de Certificados Digitales
- Conocimientos prácticos en Bases de Datos

CAPITULO 3

3. Descripción General de la Solución

En este capítulo se presentará la solución alcanzada, mencionando generalidades de la misma, sus características y arquitectura general y los requerimientos cumplidos. Un conocimiento más detallado se puede obtener en los capítulos *Capítulo 4: Aspectos relativos a la Especificación del Protocolo* y *Capítulo 5: Aspectos relativos al Análisis y Diseño de la Aplicación*, o bien en los anexos [1] y [2] si se pretende profundizar aún más.

3.1 Arquitectura General y Descripción

En el marco del comercio electrónico en Internet, el protocolo diseñado brinda la posibilidad a proveedores, de potenciar su negocio de venta de productos on-line, contratando el servicio de entrega de mercadería (servicio de delivery) a un tercero (ANC).

El ***servicio de delivery*** incluye opcionalmente subservicios de cobro y manejo de stock:

- El ***subservicio de cobro*** consiste en cobrarle al momento de la entrega al cliente el valor de la compra en nombre del proveedor. Dicho subservicio identifica el método de cobro utilizado (tarjetas de crédito, cheques, efectivo, etc) y la forma de pago por parte de la ANC al proveedor.

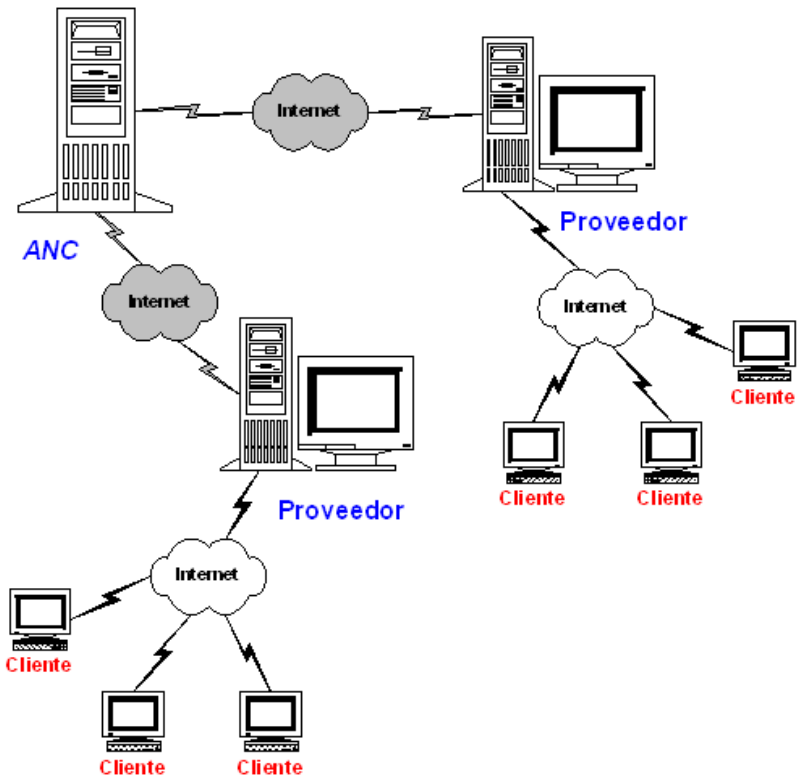
- El ***subservicio de stock*** consiste en manejar el stock de mercaderías del proveedor almacenándolo en locales propios de la ANC. La finalidad de este subservicio es tanto evitar la necesidad de retirar la mercadería en los locales del proveedor en cada solicitud de entrega, como evitar la necesidad de que el proveedor mantenga un stock almacenado en locales propios. La gestión del stock almacenado en los locales de la ANC escapa al alcance del presente protocolo.

En la figura siguiente se muestra el modelo de arquitectura contemplado por el presente protocolo. En este modelo, se distinguen tres entidades participantes en las transacciones comerciales:

- ***ANC***: Proveedor de servicios y subservicios on-line, en particular el servicio de delivery (entrega de productos) y opcionalmente los subservicios asociados de cobro y/o manejo de stock de productos de los proveedores.

- ***Proveedores***: Los cuales realizan ventas de productos on–line y negocian con la ANC la entrega de los mismos, eventualmente podrían contratar cualquier otro servicio y subservicio que brinde la ANC a través del presente protocolo.

- ***Cliente***: Es el comprador de productos de los proveedores; indirectamente negocian con la ANC – a través de los proveedores – la entrega de los productos y el método de pago de los mismos.



En dicha figura podemos observar que los Clientes solo se comunican con los proveedores y los proveedores son los únicos que se comunican con la ANC.

El protocolo de delivery especifica las interacciones entre los distintos participantes de una transacción de delivery. La misma modela los intercambios de información ocurridos en dicha transacción y el comportamiento asociado a cada una de las partes involucradas. Los intercambios definidos por el protocolo únicamente ocurren entre los proveedores y la ANC, interviniendo el cliente solo a través del proveedor con el cual interactúa. Toda la información intercambiada entre ambas partes esta constituida por documentos XML, los cuales son firmados digitalmente y encriptados proveyendo así un comercio seguro entre las mismas.

3.2 Generalidades

La solución obtenida se divide en dos grandes partes. el protocolo de delivery y la aplicación cuya responsabilidad principal es dar soporte al mismo. La principal razón por la cual dicha división tuvo lugar es lograr independencia entre ambos y de esta manera poder especificar el protocolo en un lenguaje estándar apropiado, logrando su independencia del resto del sistema de la ANC, facilitando así su distribución y haciéndola de domino público.
Podemos destacar las siguientes características de dicha solución:

- El protocolo de delivery diseñado es basado en el protocolo Two Phase Commit, adaptado para que la ANC asuma el riesgo de las transacciones inconclusas por fallas en las comunicaciones

  - Las Transacciones “inconclusas” son pasadas a estado pendiente.
  - Se define una operación de reintento para culminarlas.

- El sistema contempla un rol de Administrador, brindándole un amplio conjunto de funcionalidades para la completa gestión del sistema.
  - *Manipulación de Solicitudes*: Su cambio de estado (inicial, pendiente, finalizada, eliminada, en progreso) y sus atributos (cobro, entrega)
  - *Consultas varias*: Dichas consultas se refieren a Solicitudes, Proveedores, Contactos, y pueden ser individuales o por criterio.
  - *Configuración del Sistema*: Se pueden modificar parámetros tales como Monedas habilitadas, importe máximo permitido, así como máximo de fallas permitidas (por retiro, entrega y cobro).
  - *Manipulación de Faltas por Retiro, entrega o cobro*: Es posible consultar y modificar los contadores correspondientes a dichas faltas.
  - *Habilitaciones / Deshabilitaciónes*: Se permite la habilitación y deshabilitación de proveedores, contratos, certificados, servicios, subservicios, formas de cobro aceptadas, etc.
  - *Topes*: El sistema admite la configuración de los importes máximos por moneda aceptados en las solicitudes para cada proveedor.

- Se utiliza un Controlador cuyo diseño también fue basado en el protocolo Two-Phase Commit. El mismo es el encargado de la interacción de la aplicación con el resto de los subsistemas de la ANC en las transacciones atómicas que los involucran. Los subsistemas con los cuales se interactúa son:
  - Subsistema de Cobro
  - Subsistema Track & Trace
  - Subsistema de Proveedores
  - Subsistema de Facturación
  - Subsistema de distribución
  - Subsistema de Stock
  - Subsistema Financiero Contable
  - Subsistema de Autorización de Tarjetas de Crédito

- Recuperación ante caídas del sistema, logrando así que las transacciones inconclusas al momento de la caída, queden en un estado consistente y puedan ser finalizadas (ya sea automáticamente o a través de la operación de reintento por parte del proveedor, o por parte del Administrador) o canceladas según corresponda, asegurando siempre el conocimiento del estado de la misma por parte del proveedor.

- El sistema envía sugerencias de entrega al proveedor, en caso de falta de disponibilidad de distribución para la fecha solicitada.

- Integración con el sistema de Track & Trace, permitiendo el seguimiento de las solicitudes realizadas a través del sistema diseñado.

- Otorga flexibilidad a la lógica de negocio del Proveedor, permitiendo postergar la confirmación o cancelación de la transacción de solicitud. De esta manera es posible integrar otros procesamientos intermedios propios del proveedor en los que se necesite saber que la solicitud fue aceptada.

3.3 Atributos del Sistema

El sistema cuenta con importantes atributos, de los cuales se pueden destacar los siguientes:

- Interoperabilidad
- Protocolo abierto
- Facilidad de uso
- Tiempo de respuesta
- Confiable
- Control de acceso

- Seguro (Integridad, Autenticación, Confidencialidad, No Repudio)
- Tolerante a fallas
- Respuesta Rápida
- Multitransaccional
- Extensible
- Multipropósito

Para una descripción de los atributos mencionados, referirse al punto *Aspectos relativos a la Especificación del Protocolo* y el punto *Aspectos relativos al Análisis y Diseño de la Aplicación*

3.4 Funcionalidades del sistema

Las funcionalidades del sistema son las definidas en el documento de *Especificación de Requerimientos* [3] y detalladas a continuación:

- ***F1*** - Encriptación y desencriptación de todos los mensajes enviados / recibidos. Se deben emplear los Certificados digitales brindados por los servicios de certificación de la ANC.
- ***F2*** - Firma digital y su validación de todos los mensajes enviados / recibidos. Se deben emplear los Certificados digitales brindados por los servicios de certificación de la ANC.
- ***F3*** - Procesar las solicitudes recibidas efectuando las funcionalidades – que involucren a los distintos subsistemas - determinadas por la misma.
- ***F4*** – Validar el formato del documento de solicitud, sintáctica y semánticamente.
- ***F5*** - Registrar las bajas de existencias de stock cuando la solicitud incluye el manejo de stock
- ***F6*** - Verificar las existencias en deposito de la mercadería solicitada , cuando la solicitud incluye el manejo de stock
- ***F7*** - Validar el cliente final como cobrable cuando la solicitud incluye el manejo de cobro. Verificar que el cliente no haya realizado compras anteriores que hayan sido devueltas.
- ***F8*** - Ingresar (como pendiente de cobro) el cobro de mercadería en el sistema cuando la solicitud incluye el manejo de cobro, ingresándose al sistema financiero contable.
- ***F9*** - Ingresar el cobro de mercadería en el sistema de facturación, a los efectos de obtener las facturas a ser llevadas por el cartero para realizar el cobro.
- ***F10*** - Evaluar la existencia de recursos de distribución disponibles para cumplir con una solicitud.
- ***F11*** - Generar la sugerencia a partir de los datos de disponibilidad obtenidos del sistema de distribución, cuando no puede ser cumplida la solicitud por falta de capacidad.
- ***F12*** - Generar un numero de trackeo interno para cada solicitud aceptada y su mapeo con el sistema de track & trace.
- ***F13*** - Validar servicios / subservicios solicitados contra los contratados y habilitados para dicho proveedor.
- ***F14*** - Validar el cliente final como aceptable o no en función de los registros que se tengan de haber obtenido problemas en envíos anteriores.

- ***F15*** - Validar que el proveedor este habilitado
- ***F16*** - Validar que la forma de cobro esta habilitada para el cliente final y el proveedor
- ***F17*** - Validar la tarjeta de crédito contra una base de datos de tarjetas rechazadas
- ***F18*** - Asegurar que toda solicitud que la ANC determine que esta en condiciones de ser aceptada, se pueda efectivamente realizar.

Dichas funcionalidades serán requerimientos a cumplir por el protocolo y la aplicación de la ANC.

3.4.1 Requerimientos del protocolo

El protocolo oficia de intermediario entre las partes, coordinando el flujo de los intercambios para asegurar la correcta realización de las transacciones, asegurando las propiedades ACID (Atomicidad, Consistencia, Autonomía y Durabilidad) de las mismas.
Es su responsabilidad asegurar las propiedades de la transacción distribuida, contemplando fallas tanto en las partes involucradas como en las comunicaciones, y garantizar que el riesgo de la no realización de una transacción por alguna falla sea asumido por la ANC, brindándole al proveedor los medios para completar la transacción sabiendo que será aceptada.
El protocolo es el encargado de asegurar la confidencialidad e integridad de los datos intercambiados, así como también la autenticación y el no repudio.
Además, en el protocolo se especifica la estructura de los documentos intercambiados, así como también la semántica de los mismos; por lo tanto en la implementación del mismo se deberá manejar la validación el formato de los mencionados documentos.
Por lo antedicho el protocolo es el encargado de brindar las siguientes funcionalidades ya descritas: F1, F2, F4, F18

3.4.2 Requerimientos de la aplicación

La aplicación utiliza el protocolo de delivery para interactuar con los proveedores y aceptar sus solicitudes. Por lo tanto, entre los requerimientos funcionales definidos para la misma, están aquellas funcionalidades que dan soporte al funcionamiento del protocolo de delivery permitiendo realizar las transacciones solicitadas interactuando con los diversos subsistemas de la ANC.
Las funcionalidades F3 a F18 son requerimientos a ser cumplidos por la aplicación, pues están asociados al procesamiento de una solicitud (F3), lo cual es interno a dicha aplicación.
La funcionalidad F2 a su vez es un requerimiento que debe satisfacer la aplicación pues la misma brinda al protocolo la funcionalidad de firmado digital y validación de documentos.

3.5 Referencias bibliográficas

[1] Anexo E: Especificación del Protocolo
[2] Anexo F: Análisis y Diseño de la Aplicación
[3] Anexo D: Especificación de Requerimientos

CAPITULO 4

4.Aspectos relativos a la Especificación del Protocolo

En este punto se presentan los principales aspectos relacionados con la especificación del protocolo de delivery realizada por el grupo de Taller 5. Primeramente se introduce brevemente al lector en conceptos básicos de protocolos y arquitecturas de redes, para destacar el tipo de protocolo particular requerido por la problemática de la ANC. Seguidamente se expone el estudio realizado y las conclusiones obtenidas, con respecto a los protocolos y tecnologías estándares utilizados en el marco del comercio electrónico en Internet. Como resultado del análisis anterior se concluye la necesidad de la especificación de un protocolo el cual se describe en los puntos siguientes. En los mismos se destacan aspectos importantes del proceso de especificación como de la especificación en si. Se describe el lenguaje de especificación utilizado y las razones del mismo; los principales desafíos enfrentados y las soluciones alcanzadas para los mismos. Seguidamente se expone la arquitectura del protocolo diseñado y las funcionalidades brindadas por el mismo, así también como un flujo de información típico de una transacción definida por el protocolo. Por último se destacan las importantes características alcanzadas por el protocolo diseñado, y se presentan posibles extensiones al mismo.

4.1 Protocolos

La finalidad de esta sección es presentar al lector algunos conceptos básicos referentes a protocolos y arquitecturas de redes, de forma de poder comprender en profundidad los aspectos relativos a la especificación del Protocolo de Delivery expuesto en una sección posterior.

4.1.1 Que es un protocolo?

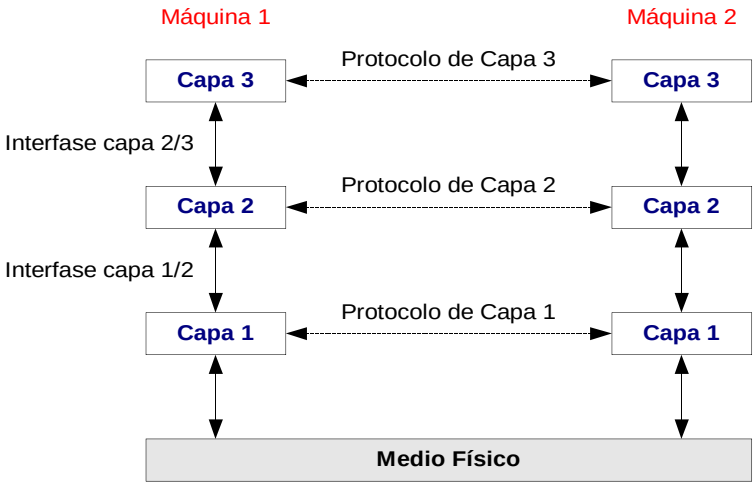
Un protocolo es una descripción formal de un conjunto de reglas y convenciones que gobiernan la forma en que distintas entidades intercambian información. Básicamente un protocolo es un acuerdo entre las partes comunicantes de cómo debe realizarse la comunicación.

4.1.2 Arquitectura de redes

Para reducir la complejidad de diseño, la mayoría de las redes se organizan como una serie de capas o niveles, donde cada capa está construida sobre su capa inmediatamente inferior. El objetivo de cada capa es ofrecer servicios a las capas superiores, ocultándoles los detalles de cómo dichos servicios están realmente implementados.

La capa n en una máquina conversa con la capa n de otra máquina. Las reglas y convenciones utilizadas en esta conversación constituyen el protocolo de la capa n. A las entidades que conforman las capas correspondientes en diferentes máquinas se le llaman pares (peers). En un modelo de capas, son las entidades pares que que se comunican utilizando protocolos. En realidad no existe una transferencia directa de datos entre las entidades pares, sino que cada capa pasa la información a transmitir a la capa inmediatamente inferior, y así sucesivamente hasta que se alcanza la capa mas baja (ver figura). Debajo de la misma se encuentra el medio físico, a través del cual se realiza la comunicación real. Entre cada par de capas adyacentes existe una interfase, la cual define los servicios que la capa inferior ofrece a la superior.

Al conjunto de capas, servicios ofrecidos y protocolos utilizados por las mismas se le denomina ***arquitectura de red***. En la figura se muestra una arquitectura de red de tres capas, en la cual se destacan mediante líneas punteadas la comunicación virtual entre cada capa, y con línea continua la comunicación real.



4.1.3 Modelos de referencia

Los modelos de referencia permiten un diseño más sencillo de las redes, dividiendo la complejidad de las mismas en subconjuntos de funcionalidades más manejables. A continuación se presentan dos modelos importantes: el modelo de referencia OSI y el modelo de referencia TCP/IP.

4.1.3.1 Modelo de referencia OSI

El modelo de referencia OSI (Open System Interconnection), esta basado en una propuesta desarrollada por la ISO (International Standars Organization) como el primer paso hacia la normalización internacional de los protocolos utilizados para la conexión de sistemas heterogéneos. El modelo OSI define siete capas:

Capa de Aplicación: Esta capa representa el propósito de la comunicación en sí, y contiene los programas del usuario que utilizan los servicios que ofrece la capa de presentación para sus necesidades de comunicación. Un protocolo de capa de aplicación define la lógica que deben seguir las aplicaciones para realizar la funcionalidad deseada.

Capa de Presentación: Esta capa a diferencia de las capas inferiores, se ocupa de aspectos de sintaxis y semántica de la información transmitida. Para posibilitar la comunicación entre sistemas heterogéneos, esta capa se encarga de negociar la representación y la codificación de la información intercambiada. Por ejemplo permite la comunicación entre máquinas con diferentes codificaciones de caracteres como ser ASCII y EBCDIC, o diferentes representaciones de números flotantes y enteros. Otros aspectos referentes a la representación de la información tratados en esta capa suelen ser la compresión y encriptación de la información transmitida.

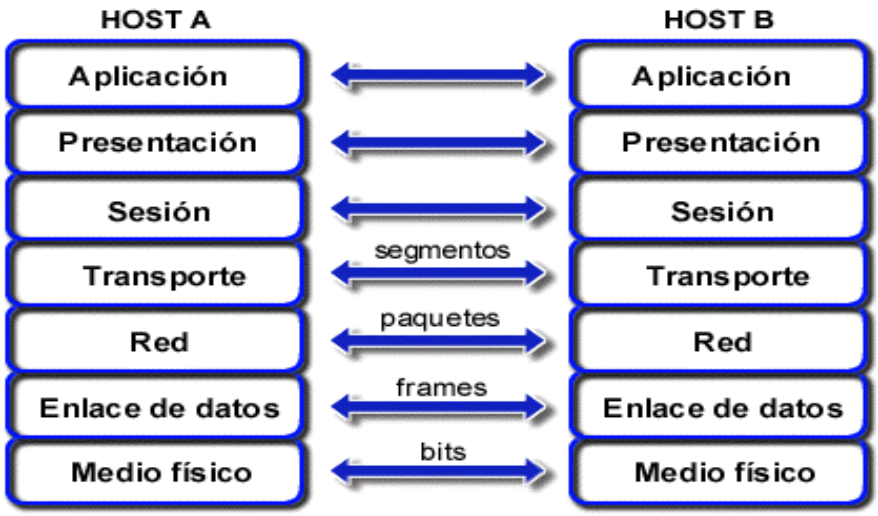
Capa de Sesión: Permite establecer sesiones entre usuarios de distintas máquinas. La idea de una sesión es proporcionar funcionalidad adicional a los servicios brindados por la capa de transporte, proveyendo una coordinación de la comunicación entre las partes.

Capa de Transporte: Se encarga de recibir datos de la capa de sesión, dividirlos en unidades más pequeñas si es necesario, pasarlos a la capa de red y asegurarse que todos ellos lleguen correctamente al otro extremo. El protocolo de esta capa establece una comunicación entre las máquinas origen y destino, mientras que los protocolos de las capas inferiores son entre cada máquina y su vecino inmediato.

Capa de Red: Se encarga de cómo rutear los paquetes del origen al destino. Extiende la funcionalidad de la capa de enlace – comunicación nodo a nodo - a toda la red.

Capa de Enlace: Se encarga de asegurar la integridad de los datos transmitidos entre dos nodos de comunicación, proporcionando una línea sin errores a la capa de red. Los datos transmitidos son divididos en frames.

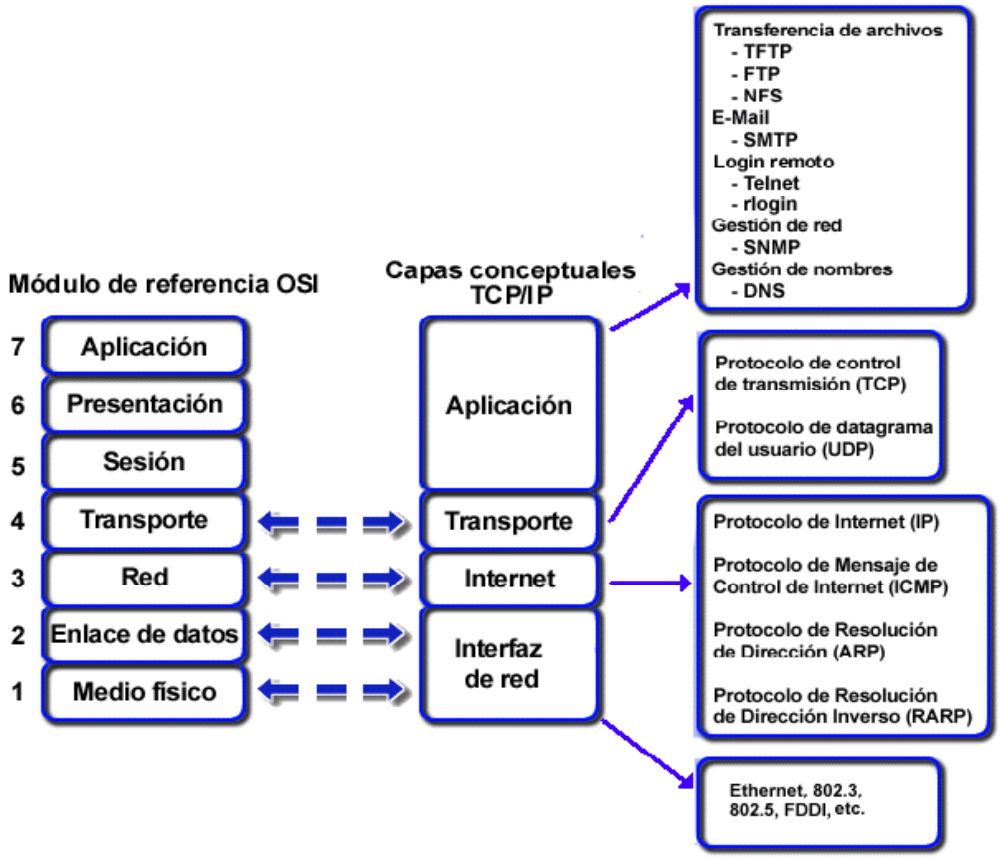
Capa Física: Se ocupa de la transmisión de bits por el canal de comunicación. Esta capa no tiene conocimiento del significado de los bits que transmite. Se ocupa de las características eléctricas, mecánicas y métodos de comunicación sobre el medio de transmisión física.



El modelo de referencia OSI no es en sí una arquitectura de red, sino mas bien un marco conceptual, pues no especifica los protocolos y servicios exactos que se han de usar en cada capa; sólo especifica las funcionalidades de cada capa. Sin embargo, la ISO también ha elaborado estándares para todas las capas, aunque no sean parte del modelo de referencia mismo.

4.1.3.2 Modelo de referencia TCP/IP

El modelo de referencia TCP/IP es el modelo utilizado en Internet. El siguiente diagrama muestra como el modelo TCP/IP puede mapearse según el modelo OSI.



4.1.4 Protocolo orientado a la problemática de la ANC

Resulta claro, según lo expuesto en los puntos precedentes, que un protocolo que cumpla las funcionalidades requeridas por la ANC, debe ser un protocolo de capa de aplicación, el cual defina la lógica necesaria que posibilite a los Proveedores realizar solicitudes de envío de paquetes, y otros servicios a la ANC.

4.2 Estudio de protocolos y frameworks de e-commerce existentes

La finalidad de esta etapa del proyecto fue realizar una investigación sobre la existencia de protocolos, marcos de trabajo (frameworks) y tecnologías estándares utilizadas para el comercio electrónico en Internet, y analizar la factibilidad de su aplicación a la problemática particular de la ANC.

Es de destacar que por ser el comercio electrónico uno de los temas principales que actualmente está concentrando mayor atención en la comunidad de Internet, una de las primeras dificultades enfrentadas en esta etapa fue el manejo del enorme volumen de información referente al tema. Además el procesamiento de dicha información requirió un gran esfuerzo de análisis y filtrado, dado que en general los temas eran tratados en forma poco precisa, y resultaba claro que la incertidumbre era una situación general dentro de la comunidad de Internet.

Del estudio realizado surge la existencia de dos tipos de comercio electrónico en Internet: ***B2C*** (Business To Consumer, negocio-consumidor) en donde los consumidores compran productos y servicios a los negocios, y ***B2B*** (Business To Business, negocio-negocio) en el cual los negocios realizan transacciones comerciales entre ellos. Dentro de la categoría B2B, para alcanzar la interoperabilidad necesaria entre los distintos negocios intrínsicamente heterogéneos, muchas compañías y organizaciones han formado iniciativas para desarrollar marcos de trabajo que posibiliten una comunicación eficiente entre negocios en Internet (***B2B frameworks***). La

meta de dichas iniciativas es proveer un estándar mundial fácilmente adoptable por las diferentes compañías. El resultado de dichos trabajos es un conjunto de estándares cada uno en diferentes etapas de desarrollo, algunos dedicados a áreas de comercio específica y otros mas genéricos.

4.2.1 Iniciativas de comercio electrónico

Entre las iniciativas de comercio electrónico más difundidas, orientadas al B2B podemos destacar Open Buying on the Internet (OBI), eCo, RosettaNet, commerce XML (cXML), BizTalk y IOTP. A continuación se presenta, en carácter informativo, una breve descripción de cada una de ellas, para una descripción mas detallada referirse al Anexo [1]

4.2.1.1 Open Buying on the Internet (OBI)

OBI se centra en la automatización de transacciones de alto volumen y bajo costo entre empresas. Esta orientado a las transacciones diarias de compra de materiales secundarios (como ser útiles de oficina, etc) que no son parte del proceso de producción de la organización y por lo tanto no estratégicos para la misma. Este tipo de transacciones son aproximadamente el 80% de las actividades de compra de la mayoría de las organizaciones y la principal finalidad de OBI es reducir el costo de dichas transacciones.

OBI esta orientado a una actividad comercial muy específica, no contemplando el amplio espectro del comercio electrónico relacionado con la compra, en particular no involucra el rol de delivery ni aparece mencionado de ninguna manera, se transfiere dicho rol a la organización vendedora, no atribuyéndole una funcionalidad importante y no estando definidos flujos de información asociado a la actividad de delivery.

4.2.1.2 eCo

El objetivo de eCo es crear una arquitectura para la interoperación entre los diferentes sistemas de comercio electrónico existentes en Internet. Utilizando una interfaz común, a través de la Web, los potenciales socios comerciales publican qué servicios online ofrece una compañía y cuales son los requerimientos necesarios para interactuar con dichos servicios, de manera que cualquier otra empresa pueda descubrirlos y determinar si sus sistemas de comercio electrónico son compatibles. Para lograr todo ello, se necesita al menos un protocolo común a través del cual los sistemas de comercio electrónico puedan describirse ellos mismos y sus requerimientos de interoperabilidad. Para ello eCo define una arquitectura en siete capas en la cual cada capa exporta una interfase que puede ser consultada y un conjunto de propiedades que utiliza para describirse ella misma.

Consultando la información publicada en cada capa otro comerciante puede: localizar el sistema, entender para que sirve, reconocer en que mercados participa, identificar los protocolos que el sistema utiliza para comunicarse, descubrir que documentos el sistema utiiza para realizar negocios, y aprender como interoperar con el sistema.

De esta forma una vez que dos empresas deciden negociar entre si, aún se requiere de un protocolo común para realizar las transacciones, dicho protocolo puede ser totalmente independiente de eCo, o ser implementado siguiendo la propia especificación de eCo.

4.2.1.3 RosettaNet

RosettaNet es un consorcium que desarrolla estándares de negocios basados en XML para la administración de la cadena de suministros dentro de las industrias de componentes electrónicos y tecnología de información. Define los procesos de negocios y proporciona las especificaciones técnicas para el intercambio de información.

4.2.1.4 Commerce XML (cXML)

Es un estándar diseñado para facilitar el intercambio de catálogos de compra entre el cliente y el vendedor. cXML esta orientado a una actividad comercial muy especifica, no contempla el amplio espectro del comercio electrónico relacionado con la compra.. cXML tiene sentido en organizaciones con varios usuarios asignados a la compra de artículos para dicha organización, en la cual es necesario centralizar de alguna manera dichas compras y su autorización. Este estándar es muy similar a OBI, ya que abarca la misma problemática y ofrece la misma funcionalidad que este último.

4.2.1.5 BizTalk

BizTalk Framework es básicamente un protocolo de capa de aplicación para el intercambio de documentos XML, basado en pasaje de mensajes BizTalk incluidos como tags en dichos documentos. Visto desde la perspectiva de la aplicación final, la misma aún debe implementar el protocolo que implemente su lógica de negocio

BizTalk define una arquitectura de tres capas, la capa de la aplicación (Application layer), la capa del Servidor BizTalk (BizTalk server layer) y la capa de comunicación (Data Communication layer). Las aplicaciones se comunican entre ellas mediante el envío y la recepción de documentos de negocios a través de Servidores BizTalk, los cuales proveen un conjunto definido de servicios a las aplicaciones. Los Servidores BizTalk se comunican entre ellos utilizando varios protocolos, como por ejemplo HTTP, SMTP o MSMQ. BizTalk no especifica que protocolos utilizar y es independiente de los detalles de implementación de los mismos.

4.2.1.6 IOTP

IOTP modela la realidad del comercio electrónico identificando los roles que juegan las entidades involucradas y definiendo un conjunto de transacciones entre ellas. En este modelo el cliente juega un papel principal, pues es el centro y coordinador de todas las transacciones. Todos los intercambios ocurren exclusivamente entre el cliente y las demás partes.

A pesar de contemplar al deliverer como un rol independiente, no se ha prestado demasiada atención al mismo. La actividad de delivery no esta demasiado integrada dentro del protocolo, su única interacción es con el cliente y se limita a validar, registrar y llevar a cabo la entrega. No existe una negociación entre el deliverer y el comerciante. La interacción con el comerciante se asume resuelta de alguna manera, ya sea por estar integrada al mismo o por estar implementada en forma externa al IOTP. En definitiva el rol de deliverer no ha sido explotado en toda su potencialidad, resultando un protocolo pobre desde dicha perspectiva, no ofreciendo ningún beneficio, ventaja o aporte para una organización orientada exclusivamente al delivery.

4.2.2 Protocolos específicos

Además de las iniciativas de B2B anteriores existen también protocolos específicos orientados a fomentar y facilitar la incorporación del comercio electrónico. Entre ellos podemos destacar SET y SOAP. A continuación se presenta una breve descripción de cada una de ellos, para una descripción mas detallada referirse al Anexo [1].

4.2.2.1 Secure Electronic Transaction (SET) protocol

SET es un protocolo desarrollado conjuntamente por Visa y MasterCard como un método para realizar en forma segura transacciones de pago con tarjetas de crédito en Internet. La especificación del protocolo esta disponible en forma pública para ser aplicada a cualquier servicio de cobro con tarjetas, y puede ser utilizada por cualquier vendedor para desarrollar aplicaciones. SET provee las siguientes características:

- Confidencialidad de la información
- Integridad de los datos
- Autenticación de los titulares de las tarjetas
- Autenticación de los comerciantes
- Interoperabilidad.

4.2.2.2 Simple Object Access Protocol (SOAP)

SOAP es un protocolo de capa de aplicación que utiliza HTTP y XML con el fin de implementar un mecanismo de RPC por Internet. La principal meta de SOAP es facilitar la interoperabilidad entre componentes de software heterogéneos sobre Internet.

SOAP define un mecanismo estándar para invocar servicios remotos sobre HTTP. Define un pequeño número de headers HTTP para indicar al servicio destinatario y al firewall que se esta intentando hacer una llamada SOAP a un procedimiento remoto a través de un paquete HTTP. Además especifica el vocabulario XML utilizado para representar la llamada, los parámetros y la respuesta.

4.2.3 Conclusiones

Como resultado de esta etapa se obtuvieron varias conclusiones, las cuales sirvieron de base para todo el trabajo posterior. Las conclusiones obtenidas son las siguientes:

- El comercio electrónico orientado al B2B, es actualmente uno de los temas mas difundidos dentro de la comunidad de Internet.
  - Existe una proliferación de iniciativas en pos de un estándar de comercio electrónico orientado al B2B. Algunas orientadas a segmentos específicos de la industria como ser OBI, RosettaNet y cXML, mientras que otras son más genéricas como eCo, BizTalk y IOTP.
  - La propia proliferación de dichas iniciativas, ha provocado un problema de incompatibilidad entre las mismas. Además considerando el hecho de que la mayoría de ellas no han alcanzado el grado de madurez necesario ni de aceptación general, hace difícil para una organización, la elección del estándar a seguir.
  - Se puede afirmar sin lugar a dudas que XML (Extensible Markup Language) es actualmente el lenguaje estándar para la definición de los datos intercambiados en el comercio electrónico en Internet. XML permite una comunicación entre las aplicaciones, independiente de las plataformas y lenguajes utilizados.
  - La amplia utilización de firmas y certificados digitales como mecanismo de autenticación, confidencialidad e integridad de la información intercambiada en las transacciones comerciales en formato electrónico.
  - La utilización de protocolos estándares como ser HTTP y SSL.
  - *No existe un protocolo estándar de delivery que contemple las funcionalidades requeridas, y aún utilizando los estándares anteriormente expuestos sigue siendo necesario la implementación de dicho protocolo.*

4.3 Especificación del Protocolo de Delivery

4.3.1 Lenguaje de especificación

La especificación del protocolo de delivery se realizó utilizando un lenguaje estándar de especificación de protocolos, de forma de lograr una especificación formal que asegurase la claridad y consistencia de la misma. Además como la especificación del protocolo debía ser de dominio público, se buscó un lenguaje que fuera a su ves fácil de entender y ampliamente utilizado. El lenguaje elegido para tal fin fue ***SDL (Specification and Description Language)*** [2].

SDL cumplía los requisitos buscados, pues entre sus características se pueden destacar las siguientes:

- ***estándar***: SDL es un lenguaje estándar definido por la ITU-T (recomendaciones Z.100 y Z.105), muy utilizado internacionalmente.
- ***formal***: SDL es un lenguaje formal, lo cual asegura una especificación precisa, consistente y clara.
- ***gráfico***: SDL es un lenguaje gráfico basado en símbolos, por lo cual es fácil de entender y fácil de usar.
- ***ampliamente difundido***: SDL es un lenguaje ampliamente difundido tanto a nivel profesional como educativo (SDL es el lenguaje de especificación de protocolos utilizado en el curso de Comunicación de Datos de la Facultad de Ingeniería).

4.3.2 Desafíos enfrentados

En este punto se presentan los principales desafíos enfrentados por el grupo de Taller 5, en el diseño del protocolo de delivery.

4.3.2.1 Tolerancia a fallas

El protocolo diseñado debía ser confiable, por lo cual debía garantizar el correcto funcionamiento de las transacciones comerciales entre los distintos Proveedores y la ANC, incluso frente a la posibilidad de fallas.

Las fallas contempladas por el protocolo diseñado corresponden a las posibles fallas de un sistema distribuido. Un sistema distribuido esta constituido por dos tipos de componentes: ***sitios***, los cuales procesan información, y ***canales de comunicaciones***, los cuales transmiten información entre los sitios. En un sistema de tales características se presentan dos tipos de fallas: fallas de los sitios y fallas de los canales de comunicaciones (pérdida de mensajes).

La especificación del presente protocolo debía ser resistente a ambos tipos de fallas

4.3.2.2 Operaciones atómicas

Desde el punto de vista de la ANC, la lógica requerida para el procesamiento de una solicitud de un proveedor, además de realizar transacciones en una base de datos interna, debía poder interactuar y utilizar recursos de varios subsistemas dentro de la ANC. Además toda solicitud realizada por un proveedor y aceptada por la ANC debía tener asociado un número de trackeo, el cual debía ser devuelto al proveedor como resultado de la transacción.

Por lo tanto el protocolo ***debía garantizar*** - teniendo siempre en cuenta la posibilidad de fallas anteriormente mencionadas - que para toda solicitud realizada por un proveedor y aceptada por la ANC, el proveedor no solo debía conocer el resultado de la transacción, sino que también debía poseer el número de trackeo asociado a la misma.

Por lo anterior resultaba claro que toda solicitud realizada por un proveedor mediante el protocolo, debía comportarse como una operación atómica - consecuentemente cumpliendo las características normales de una transacción - tanto para los Proveedores como para la ANC.

4.3.2.3 Imposibilidad teórica

La realización de una operación atómica implicaba que el protocolo debía garantizar que ambas partes pudieran llegar a una misma decisión asegurando a su ves que cada una supiera la decisión de la otra.

Esto nos llevaba a una situación similar a la planteada en ***el problema de los dos ejércitos*** [3]. En el mismo se plantea mediante una analogía con dos ejércitos, una situación en la cual dos partes que se comunican mediante un protocolo - a través de un medio no confiable, es decir con posibilidad de pérdida de mensajes – quieren asegurarse en llegar a una decisión en común.

Se demuestra que no existe protocolo capaz de resolver el problema. La demostración se realiza mediante la técnica del absurdo. Supongamos que tal protocolo existe. Entonces existen dos posibilidades, o bien el último mensaje del protocolo es esencial para el funcionamiento del mismo, o no lo es. Si no es esencial entonces elimínelo (y también todos los demás mensajes no esenciales) hasta que obtengamos un protocolo en el cual todos los mensajes son esenciales. ¿Entonces que pasa si el último mensaje se pierde.? Como acabamos de decir que era un mensaje esencial, entonces el protocolo deja de funcionar por lo cual las dos partes nunca llegarán a la decisión en común.

4.3.2.4 Transacción distribuida y protocolo 2PC (Two Phase Commit)

Con el fin de alcanzar los desafíos planteados en los tres puntos anteriores se modeló el sistema como un sistema distribuido. El mismo involucraba dos localidades diferentes (ambas partes participantes: Proveedor y ANC) que debían ejecutar operaciones sobre sus bases de datos para que la transacción pudiera concluir, garantizando la consistencia de las mismas. La ANC, debía operar sobre los recursos necesarios para garantizar la realización del servicio, y por parte del Proveedor, dentro de los recursos involucrados en la operación estaban el numero de trackeo obtenido de la ANC y el propio cliente el cual es mantenido en línea (como un recurso bloqueado) hasta la finalización de la transacción.

Por lo tanto todas las solicitudes de servicio de delivery debían ser modeladas como ***transacciones distribuidas*** y por lo tanto cumplir con las características ACID de toda transacción.

Las características ACID que debe cumplir toda transacción son las siguientes:

- ***Atomicidad (Atomicity):*** Para el mundo exterior, toda transacción ocurre de manera indivisible. Esta propiedad garantiza que cada transacción o bien no ocurre o bien se realiza en su totalidad, en cuyo caso se presenta como una acción instantánea e indivisible.

- ***Consistencia (Consistency):*** Transforma el sistema de un estado consistente en otro.

- ***Aislamiento (Isolation):*** Los resultados intermedios de una transacción no son visibles a otras transacciones, o expresado en otra forma las transacciones concurrentes no interfieren entre si. Esta propiedad garantiza que si dos o más transacciones se ejecutan al mismo tiempo, el resultado final aparece como si todas las transacciones se ejecutasen de manera secuencial en cierto orden (dependiente del sistema).

- ***Persistencia (Durability):*** Una vez comprometida una transacción, los cambios son permanentes. Esta propiedad garantiza que ningún fallo después del compromiso de una transacción puede deshacer los resultados o provocar la pérdida de los mismos.

Entonces con el fin de asegurar las propiedades ACID de toda transacción distribuida de solicitud de servicio de delivery, el protocolo diseñado se modeló como un protocolo Two Phase Commit (2PC). Para una descripción del protocolo 2PC ver el anexo [5], y para una descripción detallada de la aplicación del protocolo 2PC en el protocolo diseñado ver anexo [4].

Una de las ventajas de modelar el protocolo diseñado según el protocolo 2PC, es que en el mismo se contemplan tanto las fallas en las localidades (partes involucradas) como las fallas en las comunicaciones entre ellas .

4.3.2.4.1 Protocolo 2PC

El protocolo 2PC es un protocolo ampliamente utilizado en los entornos de bases de datos, con el fin de asegurar que todos los participantes en una transacción distribuida realizan un commit o un rollback, quedando todos así en un estado consistente.

En el protocolo 2PC existen dos roles diferentes: Participante y Coordinador; en el protocolo de delivery especificado, el ***Proveedor*** cumple el rol de un participante y la ***ANC*** cumple el doble rol de participante y coordinador. De esta manera, es la ANC (como coordinador) quien tiene en todo momento la capacidad de decidir si realizar commit o abort de la transacción.

Modificaciones al protocolo 2PC

El protocolo de delivery especificado realiza dos modificaciones sustanciales al protocolo 2PC, las cuales se describen a continuación:

- ***Riesgo asumido por la ANC***: En la etapa de preparación para el commit del protocolo 2PC, en el caso de que la ANC no recibiese respuesta alguna del Proveedor en el tiempo estipulado, sigiendo el protocolo 2PC, la ANC (actuando como coordinador) debería ***abortar*** la transacción. Sin embargo, en el protocolo de delivery diseñado el número de trackeo enviado por la ANC al Proveedor implica un ***compromiso de la realización de la transacción*** por parte de la ANC. Si el Proveedor ya tiene el mismo la transacción no puede ser abortada; pero como tampoco se tiene la certeza de que realmente haya recibido el número de trackeo entonces tampoco se puede realizar el commit de la misma, por lo cual la transacción es pasada a un ***estado pendiente*** liberando todos los recursos que la misma haya adquirido concluyendo así la transacción.

  Con esta modificación al protocolo 2PC se asegura que el riesgo de la no realización de una transacción por alguna falla, lo asuma la ANC, brindándole al Proveedor la seguridad que al recibir el numero de trackeo tiene garantizada la aceptación de su solicitud y por lo tanto la misma se realizará exitosamente o se le permitirá el reintento al conservarla en un estado pendiente en caso de alguna falla.

- ***Eliminación del estado inseguro***: En la etapa de commit del protocolo 2PC, en el caso del que el Proveedor (actuando como participante) no recibiese el mensaje de confirmación de realización de la

transacción por parte de la ANC (actuando como coordinador) en el tiempo estipulado, siguiendo el protocolo 2PC, el Proveedor pasaría a un ***estado inseguro manteniendo sus recursos bloqueados*** sin poder decidir unilateralmente si realizar o no el commit de la transacción. En el protocolo 2PC, la única manera de alcanzar una decisión y así salir del estado inseguro, es que el participante pueda consultar al coordinador y así concluir la transacción según la decisión tomada por este último. En el protocolo diseñado esta opción no sería aceptable, dado que uno de los recursos bloqueados mantenidos por el Proveedor es su propio Cliente (quien esta esperando por una respuesta del Proveedor) por lo que el Proveedor necesariamente debe llegar a una decisión en un breve lapso de tiempo. Sin embargo, en el ***protocolo diseñado***, el proveedor ***nunca se encuentra en un estado inseguro***, pues aunque no tiene certeza de que la transacción haya culminado exitosamente, desde el momento que posee el numero de trackeo, ***tiene la seguridad*** de que en el caso de no haberse realizado, podrá reintentar la transacción y la misma ***será aceptada*** por El Correo; de esta manera siempre puede responderle afirmativamente al cliente.

4.3.2.4.2 Transacción en estado pendiente

Toda transacción que fue aceptada por la ANC y para la cual nunca fue recibida una confirmación del Proveedor, es mantenida en el sistema en un estado especial denominado estado pendiente; una transacción en estado pendiente no mantiene recursos bloqueados.

Una transacción en ***estado pendiente***, permanece en dicho estado hasta que se pueda tomar la decisión de commit o abort. La decisión de commit puede ser alcanzada de dos formas diferentes:

- una forma es el caso en el cual el proveedor realiza ***la transacción de reintento de solicitud***, definida por el protocolo de delivery, con el número de trackeo obtenido en la transacción original.

- la otra forma es aquella en la cual el Administrador de la ANC decide unilateralmente realizar dicho commit (por fuera del protocolo).

Asimismo una transacción en estado pendiente puede abortar únicamente si el Administrador de la ANC decide unilateralmente hacerlo (por fuera del protocolo)

Es de destacar que en el protocolo diseñado existe la posibilidad que una transacción en estado pendiente nunca sea reintentada por el Proveedor, puesto que el mismo nunca recibió el numero de trackeo asociado a la misma. En este caso la transacción pasa a ser una transacción ***zombie***, pues nunca será reintentada y la única forma de sacarla del sistema es a través de un mecanismo manual con la intervención del Administrador.

Con el estado pendiente, se asegura que una transacción que fue aceptada por la ANC y para la cual nunca fue recibida una confirmación del Proveedor, siempre este en condiciones de ser realizada. La transacción pendiente representa la promesa de la ANC (asumida al enviar el numero de trackeo al Proveedor) de realización de una solicitud de servicio que nunca fue confirmada por el Proveedor. La finalidad de esta modificación al comportamiento del protocolo 2PC es que la ANC asuma el riesgo de que una transacción no se lleve a cabo debido a fallas en alguna de las partes participantes o en las comunicaciones.

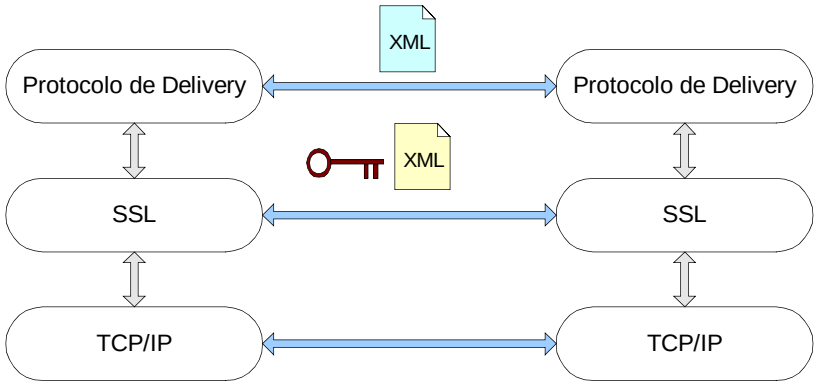
4.3.2.5 Seguridad

El protocolo diseñado debía asegurar la confidencialidad e integridad de los datos intercambiados, así como también la autenticación y el no repudio de las partes intervinientes en una transacción.

4.3.3 Arquitectura

El ***Protocolo de Delivery*** especificado es un protocolo de ***capa de aplicación*** donde toda la información intercambiada esta constituida por documentos XML. Los aspectos de seguridad tratados por el protocolo son los relativos a la integridad, la autenticación y el no repudio de los documentos transmitidos. Esto se logra mediante la utilización de firmas digitales. La arquitectura propuesta se basa en la existencia de una capa subyacente que resuelva el transporte seguro de la información intercambiada, y sea la responsable de

mantener la confidencialidad de dicha información así como también la autenticación de las partes que intervienen en la comunicación.



4.3.4 Lógica del protocolo

El protocolo especificado define una única transacción, la ***transacción de solicitud de servicios***. Actualmente los únicos servicios soportados son el servicio de ***delivery*** (que puede incluir la solicitud de subservicios adicionales: ***cobro*** y/o manejo de ***stock***) y el servicio de ***reintento de solicitud***, aunque la arquitectura del protocolo esta abierta a futuras extensiones.

El servicio de delivery es el servicio principal ofrecido a través del presente protocolo. Mediante la ***transacción de solicitud de servicio de delivery*** un Proveedor solicita a la ANC la entrega de paquetes a un Cliente, en un lugar estipulado y en un plazo de tiempo estipulado. El servicio de delivery incluye opcionalmente subservicios de cobro y manejo de stock:

- El ***subservicio de cobro*** consiste en cobrarle al momento de la entrega al cliente el valor de la compra en nombre del proveedor. Dicho subservicio identifica el método de cobro utilizado (tarjetas de crédito, cheques, efectivo, etc) y la forma de pago por parte de la ANC al proveedor.

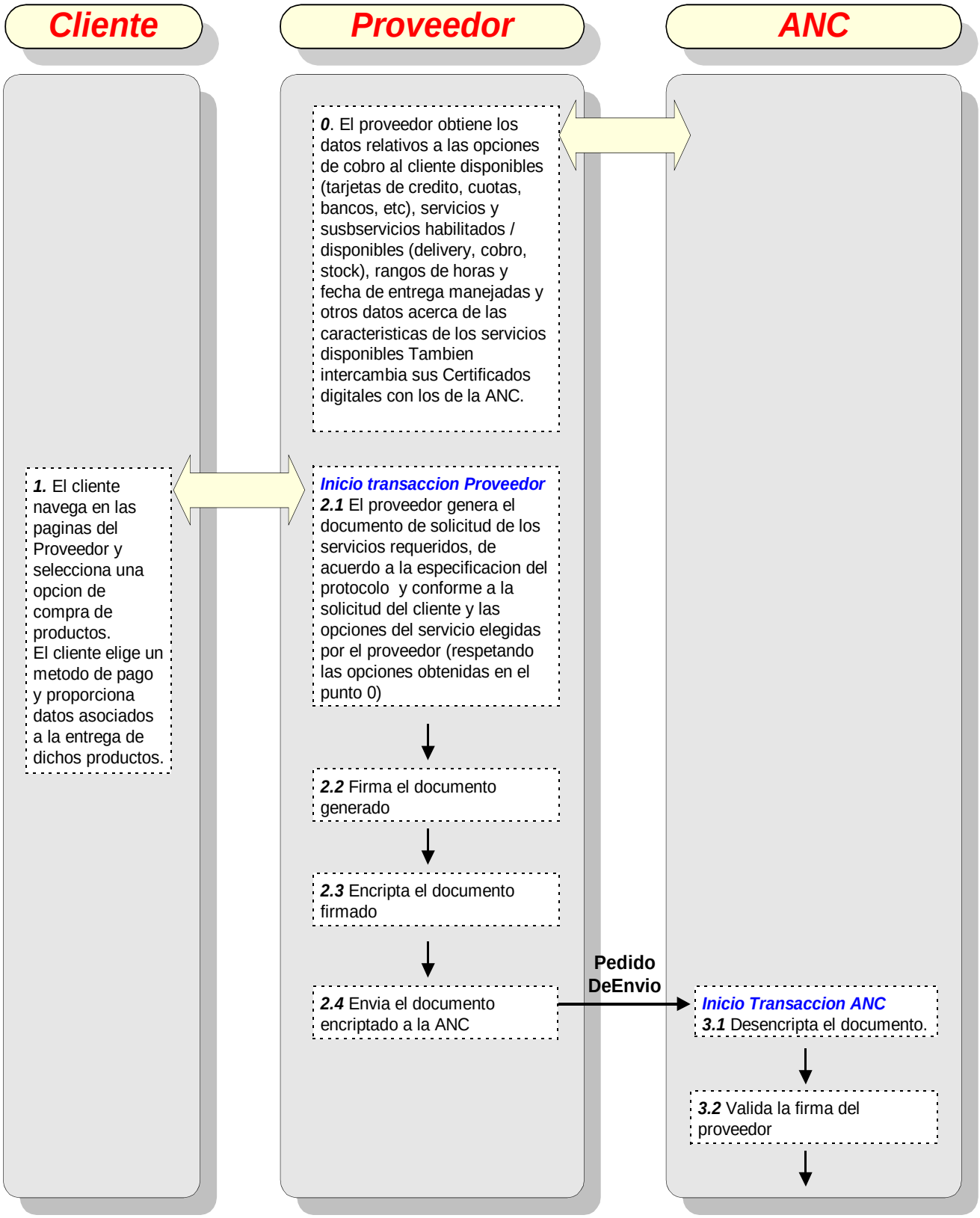
- El ***subservicio de stock*** indica que los paquetes a entregar sean retirados del stock del Proveedor mantenido en los locales de la ANC.

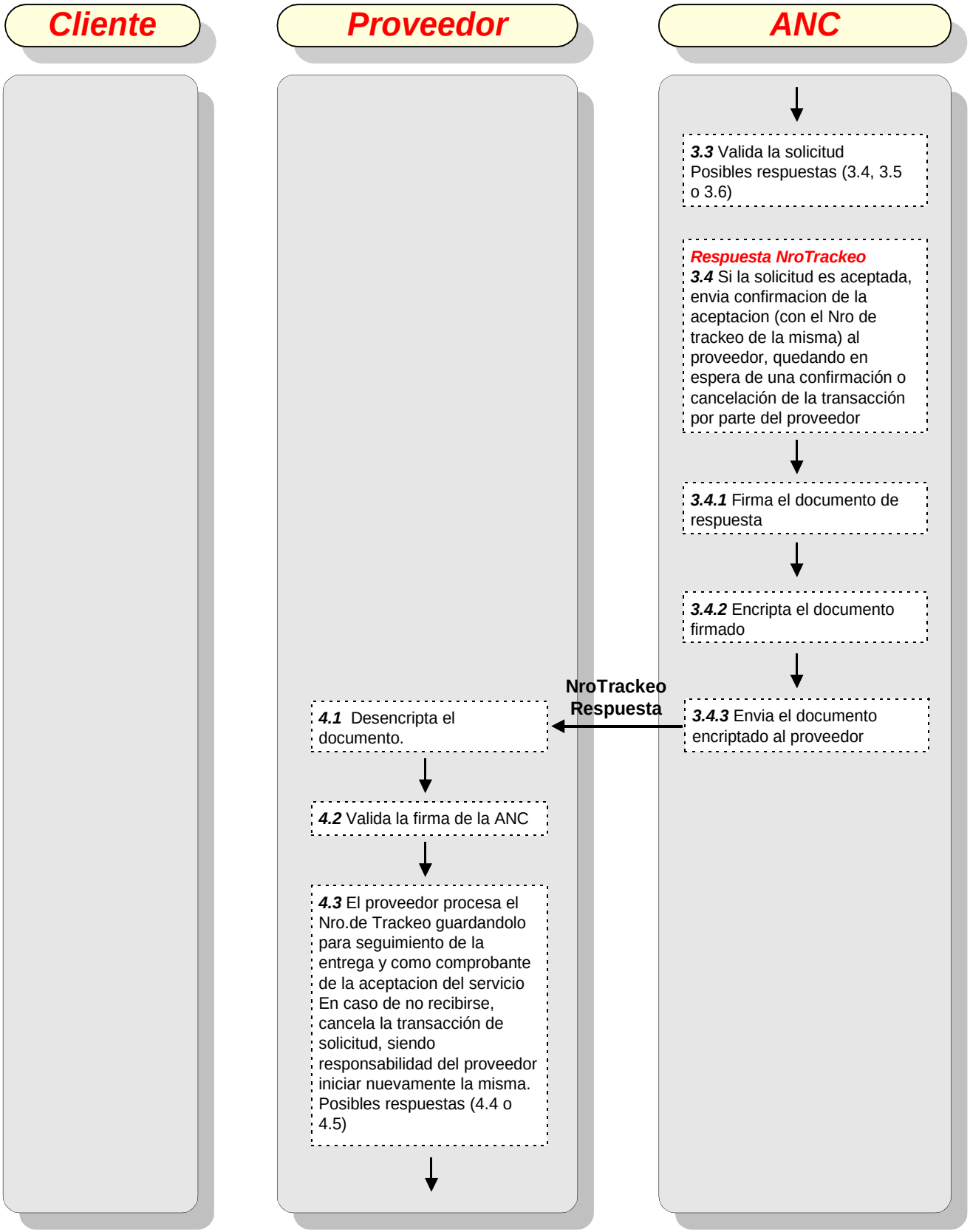
Para toda transacción de solicitud de servicio de delivery aceptada por la ANC, se genera un identificador (el número de trackeo) el cual es devuelto al Proveedor como resultado de la transacción, posibilitando un posterior seguimiento de la misma.

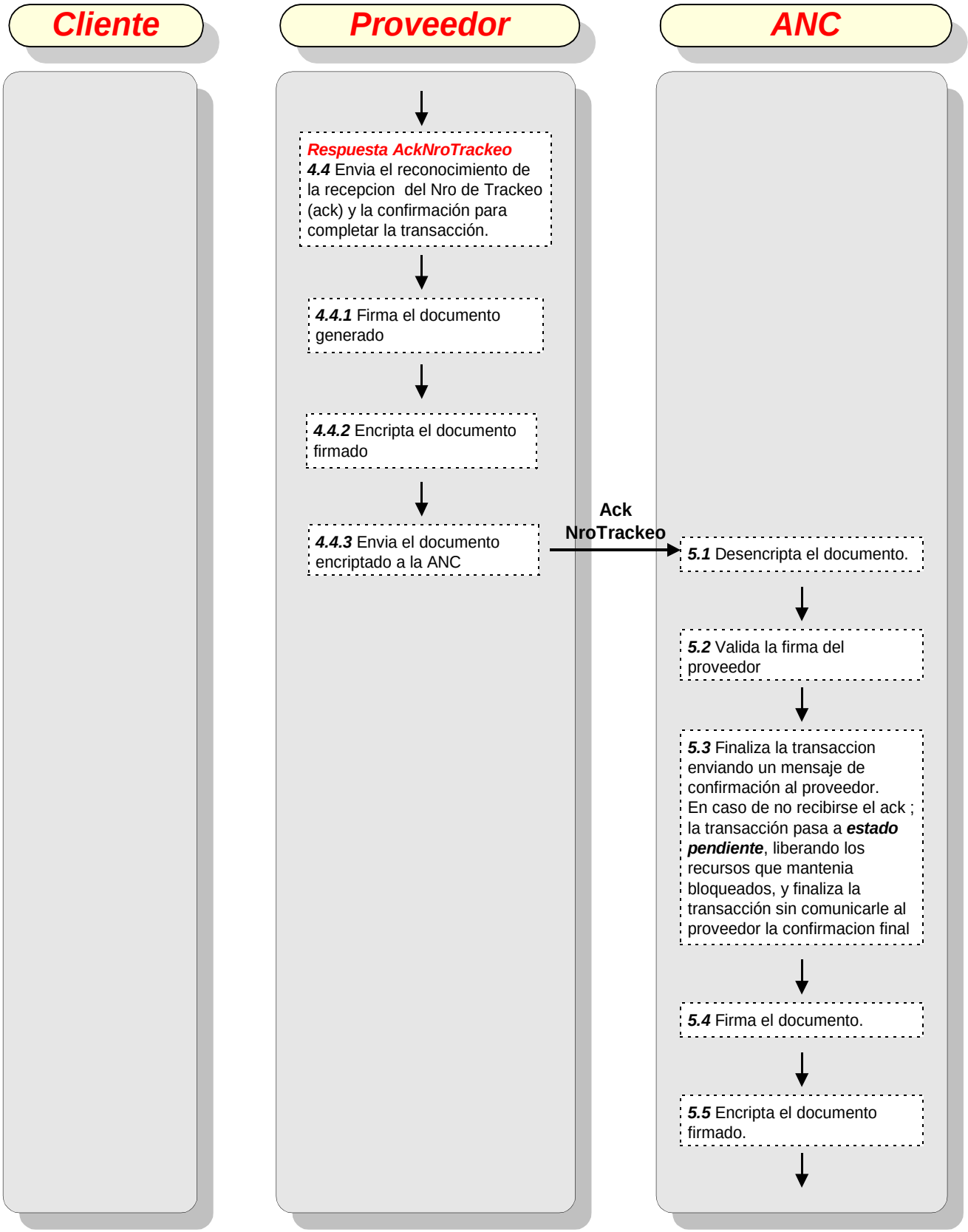
La ***transacción de reintento de solicitud*** es un complemento de la transacción de solicitud de servicio de delivery, pues a través de ésta – aún en casos de fallas en las comunicaciones y los sitios participantes- se logra garantizar la correcta terminación de la transacción original. Una transacción aceptada por la ANC para la cual no se ha recibido una confirmación por parte del Proveedor, queda en la ANC con estado pendiente, en espera de ser reintentada por el Proveedor y así concluir exitosamente. De esta manera, una vez que la ANC se compromete a realizar una solicitud (a través del envío del número de trackeo al Proveedor), siempre que la transacción esté en estado pendiente, aceptará el reintento del Proveedor y cumplirá con la misma. Esta promesa de cumplimiento es un riesgo asumido por la ANC que brinda confiabilidad en el servicio al Proveedor.

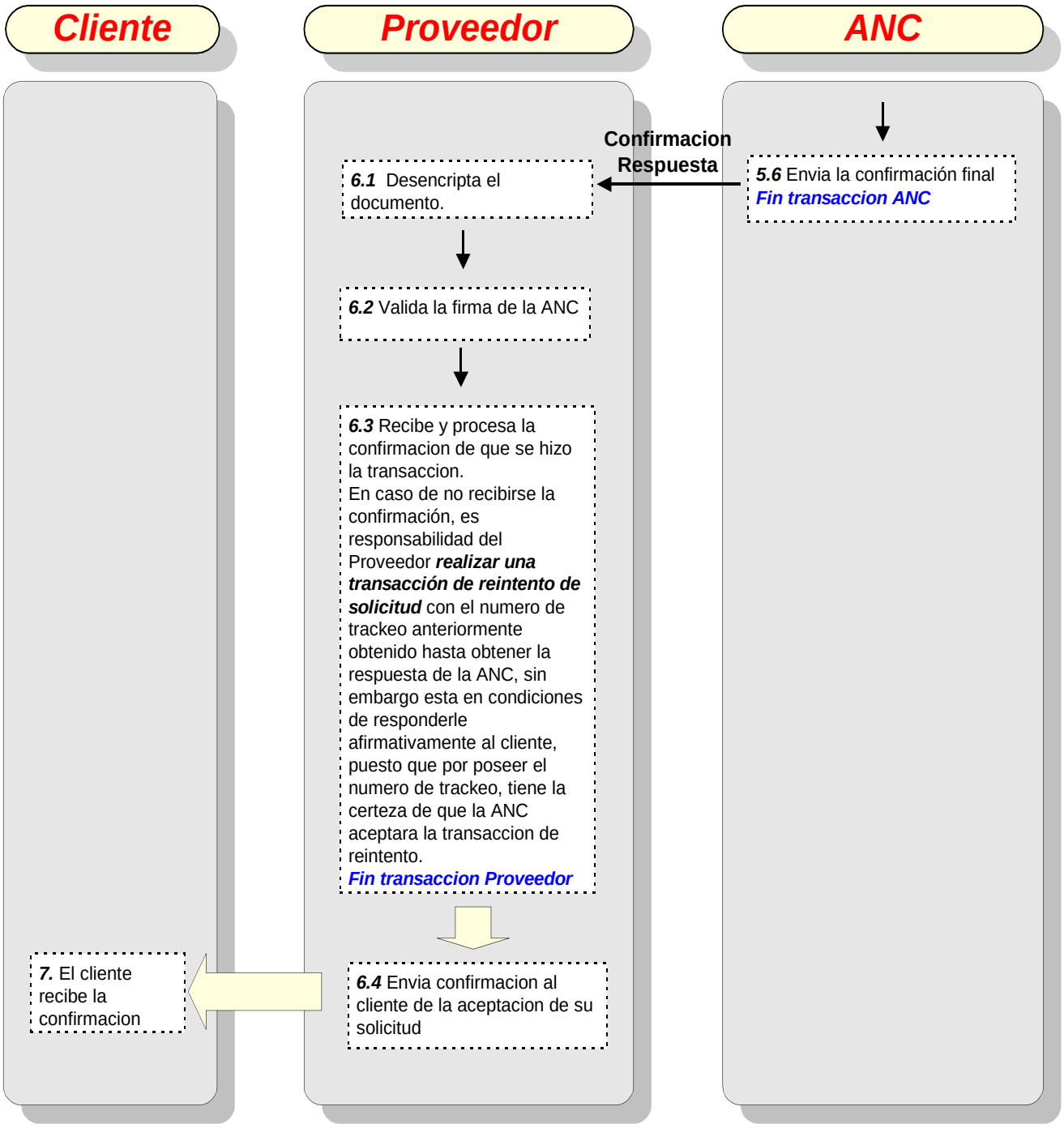
4.3.4.1 Flujo de información

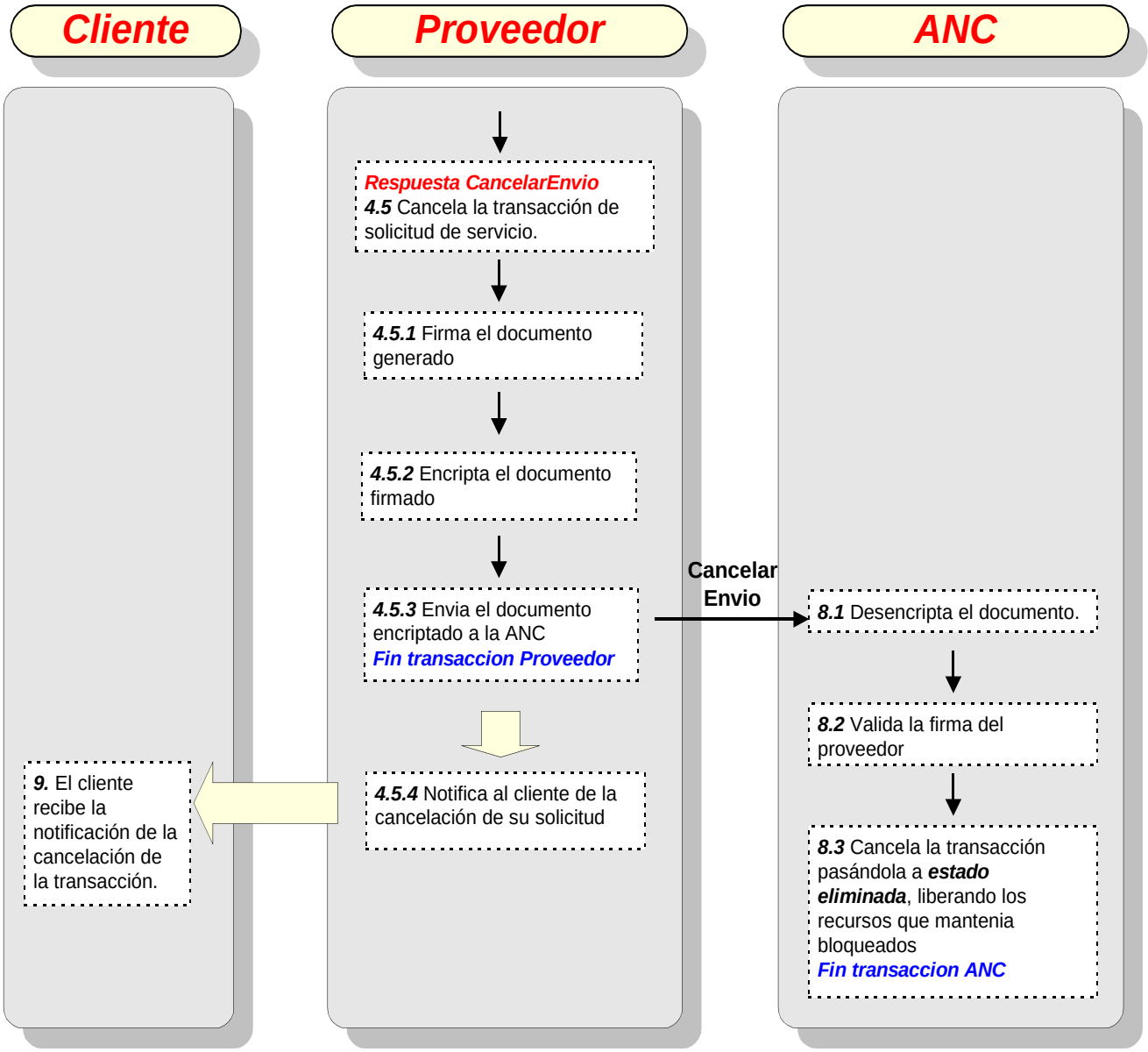
A continuación se muestra el flujo normal de información que tiene lugar en una transacción de solicitud de servicio de delivery, de acuerdo a la especificación del protocolo.

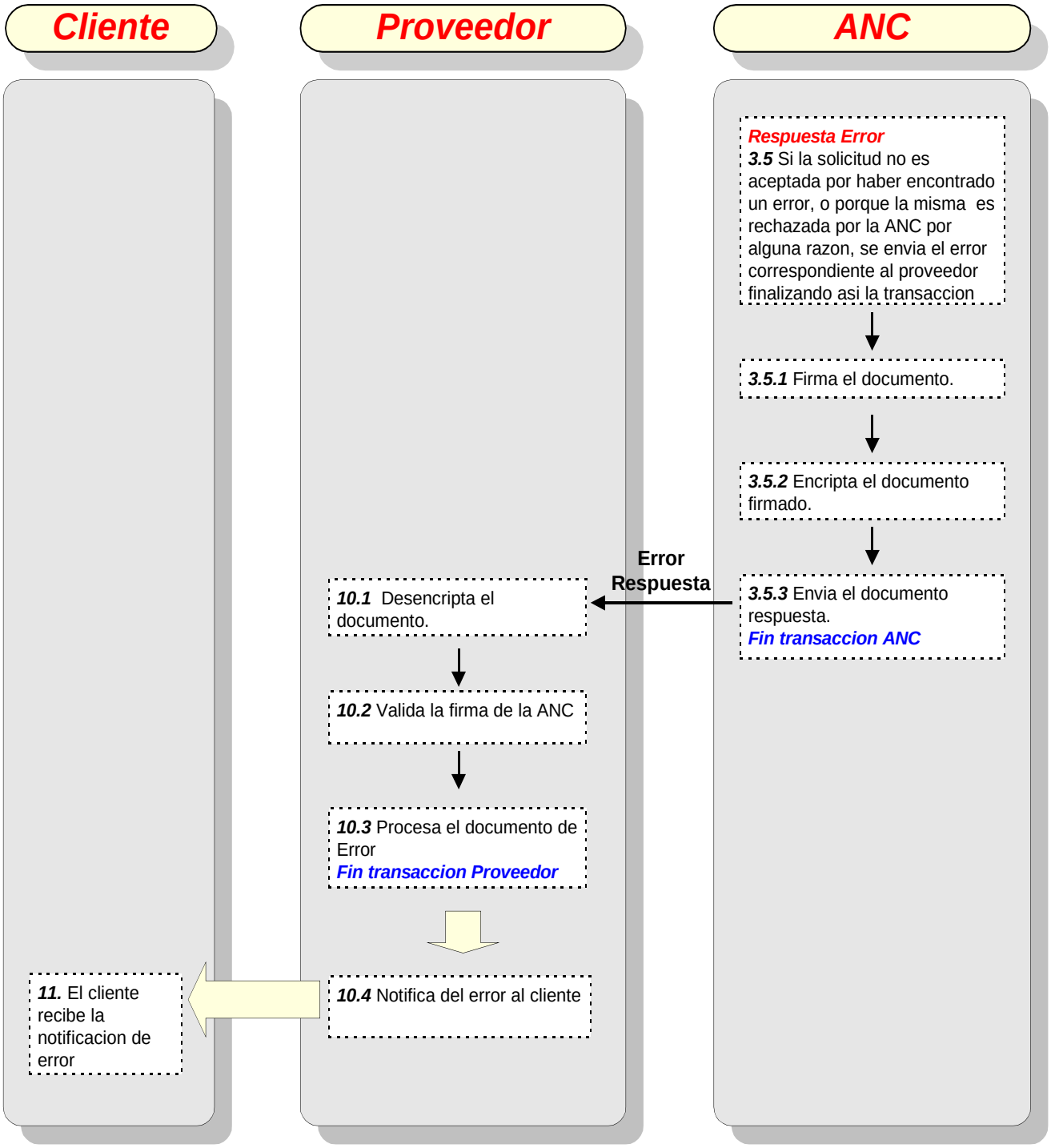

Cliente
Proveedor
ANC
0. El proveedor obtiene los datos relativos a las opciones de cobro al cliente disponibles (tarjetas de credito, cuotas, bancos, etc), servicios y susbservicios habilitados / disponibles (delivery, cobro, stock), rangos de horas y fecha de entrega manejadas y otros datos acerca de las caracteristicas de los servicios disponibles Tambien intercambia sus Certificados digitales con los de la ANC.
1. El cliente navega en las paginas del Proveedor y selecciona una opcion de compra de productos.
El cliente elige un metodo de pago y proporciona datos asociados a la entrega de dichos productos.
Inicio transaccion Proveedor
2.1 El proveedor genera el documento de solicitud de los servicios requeridos, de acuerdo a la especificacion del protocolo y conforme a la solicitud del cliente y las opciones del servicio elegidas por el proveedor (respetando las opciones obtenidas en el punto 0)
2.2 Firma el documento generado
2.3 Encripta el documento firmado
2.4 Envia el documento encriptado a la ANC
Pedido DeEnvio
Inicio Transaccion ANC
3.1 Desencripta el documento.
3.2 Valida la firma del proveedor

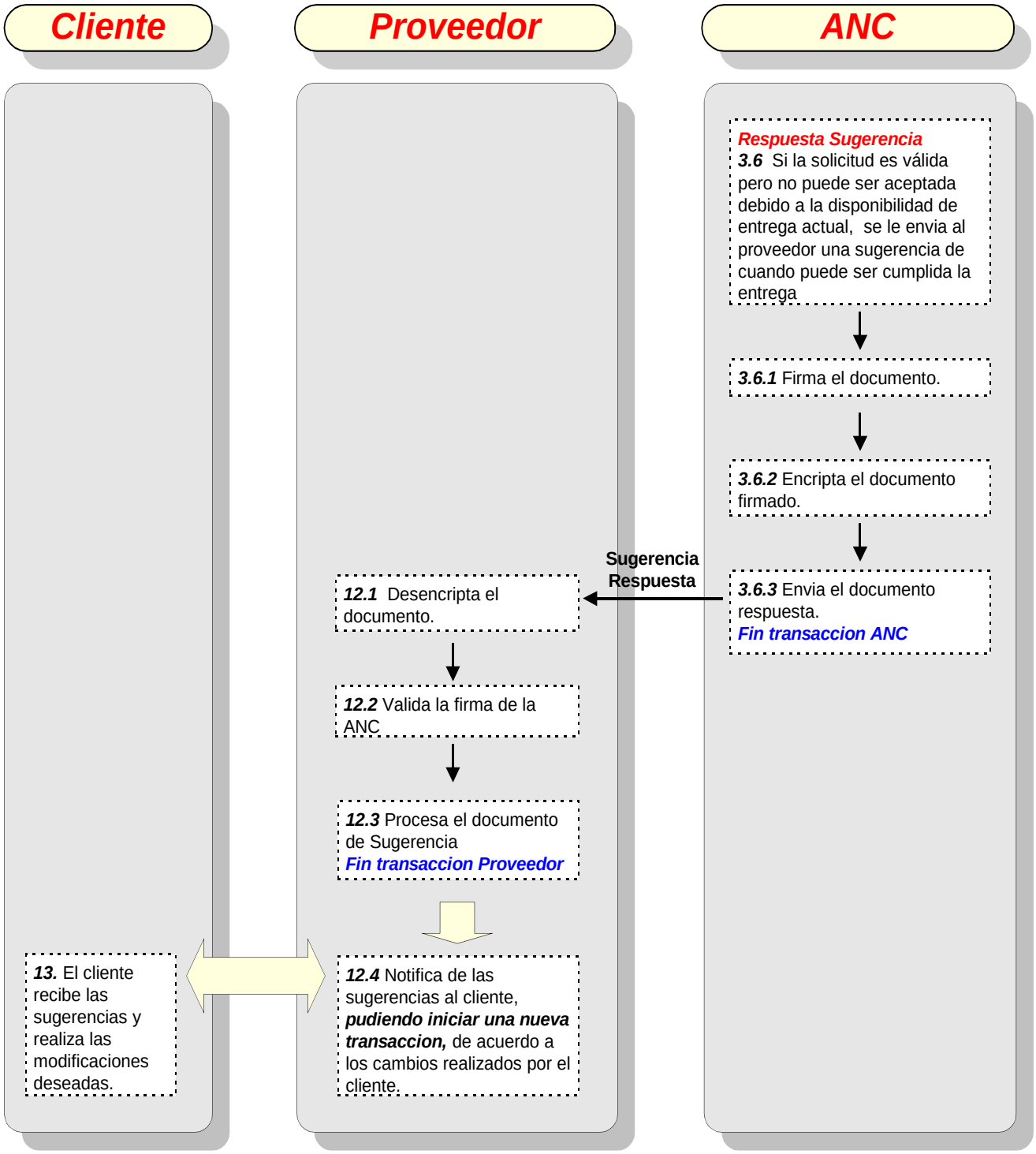

Cliente
Proveedor
ANC
3.3 Valida la solicitud Posibles respuestas (3.4, 3.5 o 3.6)
Respuesta NroTrackeo
3.4 Si la solicitud es aceptada, envia confirmacion de la aceptacion (con el Nro de trackeo de la misma) al proveedor, quedando en espera de una confirmación o cancelación de la transacción por parte del proveedor
3.4.1 Firma el documento de respuesta
3.4.2 Encripta el documento firmado
3.4.3 Envia el documento encriptado al proveedor
NroTrackeo Respuesta
4.1 Desencripta el documento.
4.2 Valida la firma de la ANC
4.3 El proveedor procesa el Nro.de Trackeo guardandolo para seguimiento de la entrega y como comprobante de la aceptacion del servicio En caso de no recibirse, cancela la transacción de solicitud, siendo responsabilidad del proveedor iniciar nuevamente la misma. Posibles respuestas (4.4 o 4.5)


Cliente
Proveedor
ANC
Respuesta AckNroTrackeo
4.4 Envia el reconocimiento de la recepcion del Nro de Trackeo (ack) y la confirmación para completar la transacción.
4.4.1 Firma el documento generado
4.4.2 Encripta el documento firmado
4.4.3 Envia el documento encriptado a la ANC
Ack NroTrackeo
5.1 Desencripta el documento.
5.2 Valida la firma del proveedor
5.3 Finaliza la transaccion enviando un mensaje de confirmación al proveedor. En caso de no recibirse el ack ; la transacción pasa a estado pendiente, liberando los recursos que mantenia bloqueados, y finaliza la transacción sin comunicarle al proveedor la confirmacion final
5.4 Firma el documento.
5.5 Encripta el documento firmado.


Cliente
Proveedor
ANC
Confirmacion Respuesta
5.6 Envia la confirmación final
Fin transaccion ANC
6.1 Desencripta el documento.
6.2 Valida la firma de la ANC
6.3 Recibe y procesa la confirmacion de que se hizo la transaccion.
En caso de no recibirse la confirmación, es responsabilidad del Proveedor realizar una transacción de reintento de solicitud con el numero de trackeo anteriormente obtenido hasta obtener la respuesta de la ANC, sin embargo esta en condiciones de responderle afirmativamente al cliente, puesto que por poseer el numero de trackeo, tiene la certeza de que la ANC aceptara la transaccion de reintento.
Fin transaccion Proveedor
6.4 Envia confirmacion al cliente de la aceptacion de su solicitud
7. El cliente recibe la confirmacion


Cliente
Proveedor
ANC
Respuesta CancelarEnvio
4.5 Cancela la transacción de solicitud de servicio.
4.5.1 Firma el documento generado
4.5.2 Encripta el documento firmado
4.5.3 Envia el documento encriptado a la ANC
Fin transaccion Proveedor
Cancelar Envio
8.1 Desencripta el documento.
8.2 Valida la firma del proveedor
8.3 Cancela la transacción pasándola a estado eliminada, liberando los recursos que mantenia bloqueados
Fin transaccion ANC
4.5.4 Notifica al cliente de la cancelación de su solicitud
9. El cliente recibe la notificación de la cancelación de la transacción.


Cliente
Proveedor
ANC
Respuesta Error
3.5 Si la solicitud no es aceptada por haber encontrado un error, o porque la misma es rechazada por la ANC por alguna razon, se envia el error correspondiente al proveedor finalizando asi la transaccion
3.5.1 Firma el documento.
3.5.2 Encripta el documento firmado.
3.5.3 Envia el documento respuesta.
Fin transaccion ANC
Error Respuesta
10.1 Desencripta el documento.
10.2 Valida la firma de la ANC
10.3 Procesa el documento de Error
Fin transaccion Proveedor
10.4 Notifica del error al cliente
11. El cliente recibe la notificacion de error



4.3.5 Características

A continuación se resaltan las principales características del protocolo especificado.

4.3.5.1 Facilidad de uso
La sencillez del flujo de información definida por el protocolo permite una lógica de procesamiento simple por parte de los Proveedores que quisieran utilizarlo, posibilitando así un rápida aceptación y difusión del mismo en el mercado.

4.3.5.2 Protocolo abierto
La especificación del protocolo debía ser de dominio publico, permitiendo su implementación y utilización a todo proveedor que desease utilizar los servicios brindados por la ANC. Para ello la especificación del protocolo se realizó utilizando un lenguaje estándar de especificación de protocolos, de forma de lograr una especificación formal que asegurase la claridad y consistencia de la misma. El lenguaje utilizado fue SDL (Specification and Description Language).

4.3.5.3 Extensible
La incorporación de nuevos servicios al protocolo, no invalida las versiones previas que posean los proveedores, permitiendo la coexistencia entre las distintas versiones. La incorporación de nuevos subservicios, no tiene impacto en el protocolo, debido al uso de documentos XML.

4.3.5.4 Interoperable
El protocolo diseñado permite la comunicación entre sistemas heterogéneos, es decir sistemas con diferentes plataformas tanto de hardware como de software. Para ello toda la información intercambiada por el protocolo se define en un formato estándar de Internet. El formato utilizado es el lenguaje XML.

4.3.5.5 Multitransaccional
Soporta múltiples transacciones concurrentemente.

4.3.5.6 Confiable
El protocolo especificado garantiza el correcto funcionamiento de las transacciones definidas, aún en casos de fallas tanto en las localidades (partes involucradas) como en las comunicaciones entre ellas. Una vez que el proveedor recibe el NroTrackeo (identificador de la solicitud), tiene la tranquilidad que la transacción se completará con éxito, a lo sumo tendrá que reintentarla mas tarde.

4.3.5.7 Tiempos de respuesta razonables
El protocolo se diseño teniendo en cuenta que las solicitudes de servicio involucraban transacciones on-line, en las cuales hay clientes en espera de su finalización, por lo cual los tiempos de respuesta debían ser razonables. El proveedor esta en condiciones de responder al cliente apenas obtenga el NroTrackeo.

4.3.5.8 Confidencialidad
El protocolo garantiza que sólo las partes intervenientes en una transacción pueden acceder a la información transmitida en la misma. Esto se logra mediante la utilización de algoritmos de encriptación asimétricos, con los cuales se encripta toda la información transmitida por el protocolo.

4.3.5.9 Autenticación
El protocolo permite validar las identidades de las partes intervenientes en toda transacción. Esto se logra mediante el uso de certificados digitales.

4.3.5.10 Integridad
El protocolo garantiza la integridad de toda la información transmitida, es decir que la misma no ha sido modificada en forma alguna.

4.3.5.11 No repudio
El protocolo garantiza que ninguna de las partes intervenientes en una transacción, puede negar su participación en la misma. Esto se logra mediante la utilización de firmas digitales.

4.3.5.12 Control de acceso
El protocolo garantiza que sólo quienes estén debidamente autorizados podrán acceder a los servicios brindados por la ANC. Para ello el protocolo especifica que toda solicitud de servicio debe contener el identificador del Proveedor que la solicita; dicho identificador es suministrado por la ANC.

4.3.5.13 Multipropósito
El protocolo puede ser utilizado para cualquier tipo de transacciones que tengan las características del servicio de delivery, es decir transacciones que necesiten ser modeladas con 2PC y que se asuma el riesgo de fallas en las comunicaciones, permitiendo el reintento de la transacción.

4.3.5.14 Seguimiento de las transacciones
El protocolo garantiza que toda transacción aceptada por la ANC tiene asociado un número de trackeo, el cual es devuelto al Proveedor como resultado de la transacción, posibilitando un posterior seguimiento de la misma.

4.3.5.15 Procesamiento de sugerencias
Si una solicitud de envío no puede ser aceptada por la ANC, debido a falta de disponibilidad, se le envía al proveedor un documento de respuesta en el cual se presentan distintas sugerencias de entrega de paquetes, agrupados en conjuntos de paquetes con distintos plazos de entrega entre ellos, para los cuales la ANC sí está en condiciones de cumplir en ese momento.

4.3.5.16 Protocolo de compromiso en dos fases (2PC)

El protocolo diseñado, al estar modelado como un protocolo de compromiso en dos fases, permite garantizar las propiedades ACID (Atomicidad, Consistencia, Autonomía y Durabilidad) de todas las transacciones realizadas. La especificación del protocolo garantiza además que el riesgo de la no realización de una transacción por alguna falla, sea asumido por la ANC, brindándole al Proveedor los medios para completar la transacción sabiendo que será aceptada.

Otra característica como protocolo 2PC es que otorga flexibilidad a la lógica de negocio del Proveedor, al permitirle determinar el inicio de la segunda fase de la transacción (confirmación o cancelación) para el momento que él considere apropiado. Los Proveedores, una vez sabida la aceptación de la solicitud de delivery por parte de la ANC y antes de la confirmación o cancelación de la misma, tienen la oportunidad de realizar cualquier lógica de negocio que consideren pertinentes para el procesamiento de las solicitudes de sus clientes, independientemente de la interacción con la ANC.

4.3.5.16.1 Ejemplos
A continuación se presentan dos ejemplos que muestran claramente las ventajas del protocolo diseñado como un protocolo de compromiso en dos fases.

4.3.5.16.1.1 Utilización del protocolo SET
Un caso típico podría ser la utilización del protocolo SET [8] entre el Proveedor y el Cliente, como protocolo de autorización y cobro de tarjetas de crédito on-line. En este caso al Proveedor le interesaría saber antes de corarle al cliente mediante SET, si la solicitud de envío es aceptada o no por la ANC, puesto que es muy probable que los precios presentados al Cliente, dependan de los plazos de entrega seleccionados por el mismo. Para ello el Proveedor primero enviaría la solicitud de delivery a la ANC, y si la misma es aceptada recién entonces le cobraría al cliente en forma on-line mediante SET; entonces después de asegurarse que se realizó el cobro correctamente, enviaría la confirmación de la solicitud a la ANC.

4.3.5.16.1.2 Transacciones compuestas
Un Proveedor que así lo quisiera, podría realizar varias transacciones de solicitud de envió, y que todas ellas en conjunto se comportasen como una única transacción. Para ello primero debería iniciar cada una de las transacciones deseadas y recién luego de recibir la aceptación de cada una de ellas por parte de la ANC, confirmaría cada una de las transacciones. En el caso de que alguna de las solicitudes no halla sido aceptada por la ANC, entonces cancelaría todas las transacciones.

4.3.6 Extensiones

A continuación se presentan un conjunto de características que consideramos deseables que una futura extensión al protocolo actual soporte.

4.3.6.1 Firmas digitales según las recomendaciones de la W3C,

Actualmente existe un grupo de trabajo, XML-Signature Working Group [6] conformado por las organizaciones IETF y W3C, con el fin de desarrollar una sintaxis XML para representar firmas digitales y procedimientos para calcular y verificar las mismas. Las firmas XML definidas pueden ser aplicadas a cualquier tipo de datos en formato digital manejado por las aplicaciones, y en particular a documentos XML. Es de esperar que las recomendaciones realizadas por dicho grupo de trabajo se conviertan en un futuro, en el estándar para el procesamiento de firmas digitales de documentos XML. Sin embargo debido a que las herramientas actuales aún no incorporan dichas recomendaciones, no fue viable su utilización en la implementación actual del protocolo de delivery. Para una descripción del mecanismo implementado en el protocolo para la firma digital de los documentos XML intercambiados ver el documento *Detalles de Implementación* [7].

4.3.6.2 Uso de Schemas XML

Actualmente la definición y validación de los documentos XML intercambiados por el protocolo se realiza mediante DTDs. Se recomienda que una futura versión del mismo substituya el uso de los DTDs por Schemas XML, los cuales son más sencillos – pues se especifican utilizando el propio lenguaje XML, en contraposición de los DTDs – y además permiten una mayor potencialidad en cuanto a la definición de los tipos de datos manejados en los documentos y las validaciones de los mismos.

El uso de Schemas XML no fue incorporado en la versión actual del protocolo debido a que todavía no han sido adoptados como un estándar, pues sigue siendo un trabajo en proceso de la W3C. Debido a ello los mismos no son actualmente soportados por las herramientas disponibles, y las que sí lo hacen lo que incorporan son implementaciones particulares de la recomendación de la W3C, por lo que no son 100% compatibles.

4.4 Referencias bibliográficas

[1] Anexo C: Estudio de la aplicación de los protocolos de e-commerce

[2] SDL (Specification and Description Language) http://www.sdl-forum.org/
El sitio web dispone de tutoriales de SDL, tanto para acceso en línea como para bajar.

[3] Andrew S. Tanenbaum “Computer Networks” Third Edition

[4] Anexo E: Especificación del Protocolo

[5] Anexo B: Algunos conocimientos previos requeridos, documento Protocolo de 2PC (Two Phase Commit)

[6] XML-Signature Working Group http://www.w3.org/Signature/
XML-Signature Syntax and Processing
http://www.w3.org/TR/2000/CR-xmldsig-core-20001031/

[7] Anexo G: Detalles de Implementación

[8] Anexo C: Estudio de la aplicación de protocolos de e-commerce, documento SET

CAPITULO 5

5.Aspectos relativos al Análisis y Diseño de la aplicación

En el desarrollo de los siguientes puntos se pretende explicar el cambio en la concepción inicial de la aplicación, cambiando así la relevancia de la misma (punto *Necesidad de una aplicación*) e impactando en la elección de la metodología a utilizar para el *Análisis y Diseño*. Se pretende también exponer el *proceso de desarrollo* realizado, comentando brevemente la documentación obtenida, su contenido y finalidad, haciendo referencia a los anexos que presentan cada tema en particular de manera extensa.
Luego en el punto *Algunos elementos del Análisis y Diseño*, se presentan en forma muy resumida los puntos del Análisis y Diseño que permiten un conocimiento del resultado obtenido. Por ultimo en el punto *Desafíos*, se destacan los desafíos y algunas dificultades puntuales encontradas.

5.1 Necesidad de una aplicación

Se podría resumir someramente que el problema inicial consistía en la especificación e implementación del ya mencionado protocolo de delivery. La herramienta que implementase dicho protocolo debía almacenar los documentos de solicitud del servicio y poder generar los documentos de respuesta. Pero lo que inicialmente no estaba claro era el tipo de procesamiento que se realizaría sobre estas solicitudes, así como no existía una idea clara de las características del servicio que se pretendía brindar. A medida que se profundizó en estos temas, y se definían algunas consideraciones relativas al negocio – al especificarse el antedicho protocolo – fue surgiendo la relevancia de la aplicación, incorporándole muchas responsabilidades que inicialmente no eran visibles (requerimientos ocultos del sistema). Estas responsabilidades la fueron convirtiendo en una pieza clave, tornándose una concepción más global del sistema original.

Entre las mencionadas responsabilidades de la aplicación, podemos citar:

- ***Soporte al protocolo:*** (ya considerada en la concepción original de la aplicación) Contiene todas las funcionalidades requeridas por el protocolo para la realización de las transacciones estipuladas (inicio, cancelación, pasaje a estado pendiente, confirmación, etc). Se encarga del procesamiento interno de las transacciones y su registro en la base de datos del sistema.

- ***Interacción con el resto de los sistemas de la ANC:*** Durante el procesamiento de las transacciónes, se interactúa con los sistemas de la ANC ya sea directamente o a través del uso del ***módulo Controlador2PC*** en el caso de ser necesaria una operación atómica que involucre varios sistemas. Dicho módulo, se encarga de implementar el protocolo Two Phase Commit en las operaciones que

involucren los distintos sistemas, garantizando así la atomicidad de las mismas. También se da el caso inverso, en el cual el resto de los sistemas interactúan con el sistema de delivery.

- ***Administración:*** Permite un monitoreo del sistema y por lo tanto del servicio brindado; admitiendo además la realización de consultas claves para poder mantener un mayor control del sistema. Posibilita la realización de cambios en su configuración, cambios relativos al estado de las solicitudes, relativos a los proveedores, sus contactos, etc. Es importante destacar uno de los puntos que más influencia tuvo en la definición de una administración, este es la funcionalidad de gestión manual de transacciones pendientes.

- ***Recuperación de caídas del sistema:*** La misma surge de la necesidad de mantener la consistencia de las transacciones realizadas, en caso de una caída del sistema. El proceso de recuperación se encarga de completar las transacciones que hayan quedado inconclusas debido a la falla mencionada, ya sea realizando el rollback o el commit de las mismas, según sea posible.

- ***Independencia del protocolo:*** La idea es tratar de mantener una independencia entre el protocolo y la aplicación, siendo el primero encargado de la lógica del intercambio de mensajes entre las partes, la validación del formato de los documentos XML intercambiados y las cuestiones relativas a seguridad (encriptado, firma digital y su validación.); y la segunda encargada de todos los puntos anteriormente mencionados: Soporte al protocolo, Interacción con los sistemas de la ANC, Administración y Recuperación de Fallas.
  Uno de los propósitos de dicha independencia es poder especificar el protocolo en un lenguaje estándar apropiado e independiente de las distintas arquitecturas; quedando la misma independiente del resto del sistema, facilitando así su distribución y haciéndola de dominio público

5.2 Que metodología utilizar?

Al ser conscientes de la envergadura del sistema a desarrollar, tomó gran relevancia en particular la metodología de desarrollo a seguir y el paradigma de programación a adoptar. Puesto que se pretendía crear un gran sistema correcto, confiable y de fácil mantenimiento y evolucionabilidad, cuyo diseño sea de fácil comprensión a pesar de las características complejas del sistema en sí; y visto el auge del paradigma de orientación a objetos y su contribución para lograr los objetivos antedichos; se optó por este paradigma. Pero no alcanza con la adopción del mencionado paradigma, es necesario una notación estándar para el modelado y una metodología de desarrollo que colaboren para alcanzar estos objetivos.

La notación estándar utilizada es UML (Unified Modeling Language) que es la primer propuesta en este sentido que logra estandarizarse y por lo cual puede servir para una comunicación eficaz y uniforme. Esta notación junta las ventajas de las distintas notaciones utilizadas hasta el momento para el modelado orientado a objetos, permitiendo la visualización, especificación, construcción y documentación de los elementos del sistema de software a través de todo su ciclo de vida, y es considerado elegante, expresivo y flexible.

Para el proceso de desarrollo adoptamos la propuesta por Craig Larman [1] por ser una simple y efectiva guía y estar orientada a la tecnología de objetos y el uso de la notación UML .

5.3 Proceso de desarrollo

A continuación se presenta una reseña del proceso de desarrollo seguido, indicándose la documentación obtenida, su contenido y finalidad.

El objeto del análisis orientado a objetos es el estudio del dominio y ámbito del problema, con la finalidad de comprender los términos empleados en el dominio y definir una especificación y requerimientos desde la perspectiva de la clasificación por objetos. La documentación obtenida de dicho análisis puede resumirse en el siguiente cuadro:

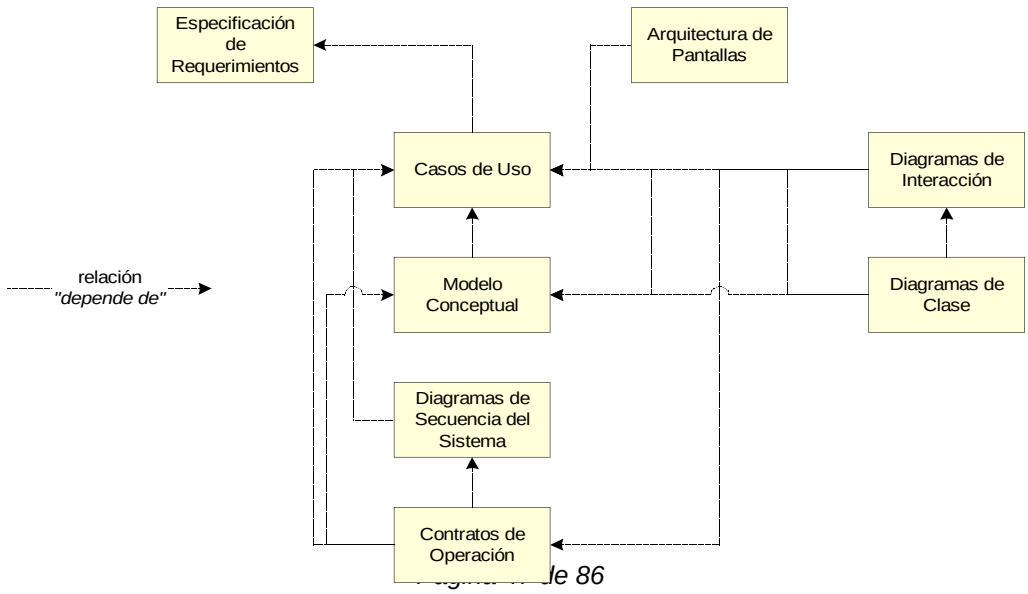
Documento	Propósito
especificación de Requerimientos	Identificar y definir las necesidades y características que se desean del sistema a desarrollar, así como servir para la comunicación con el cliente
Casos de Uso	Definir los procesos del dominio
Modelo Conceptual	Definir los conceptos y términos
Diagramas de secuencia del sistema	Determinar los eventos y operaciones del sistema.
Contratos	Definir que hacen las operaciones del sistema (determinar los cambios de estado que producen sobre los objetos definidos en el modelo conceptual)

La finalidad del diseño orientado a objetos, es definir las especificaciones lógicas del software de manera que cumplan los requisitos funcionales, basándose para ellos en la descomposición por clases de objetos. [1] Se resume la documentación de esta fase en el siguiente cuadro:

Documento	Propósito
Diagramas de Interacción	Definir las responsabilidades de los objetos y el flujo de mensajes con otros para completar una operación asociada a cada caso de uso.
Diagrama de Clase	Definir concretamente las clases de software a implementar, sus métodos, atributos con sus tipos, visibilidad y la navegación entre objetos
Diagrama Entidad - Relación	Modelar los elementos del diagrama de clases desde un enfoque relacional, mostrando gráficamente las entidades y sus relaciones, de cara al diseño físico del esquema de Base de Datos

Es de destacar que al finalizar cada documento, se enviaba el mismo a los responsables de la ANC, y que mediante reuniones periódicas [15] se obtenía su fundamental aporte, y consentimiento de las decisiones tomadas.

La dependencia entre la distinta documentación generada se expresa gráficamente en el siguiente cuadro:



5.3.1 Especificación de Requerimientos

Luego de haber adquirido un conocimiento general de la problemática, especificado el protocolo de delivery y a través de este proceso, haber definido las características del servicio a brindar (principalmente por la definición de los documentos intercambiados, los cuales deben contener todos los datos necesarios asociados); y por supuesto luego de reponderar la importancia de la aplicación de acuerdo a lo comentado en el punto *Necesidad de la aplicación*, se realizó un Análisis de Requerimientos más detallado, incluyendo en él las funcionalidades del protocolo ya definidas y extendiéndolo con las que surgían como nuevas funcionalidades de la aplicación. Dicho análisis se formalizó a través de la generación del documento *Especificación de Requerimientos* [2] el cual incluía además el objetivo inicial del sistema, los atributos deseables del mismo, una clasificación de los requerimientos y otras puntualizaciones acerca de la interacción con el resto de los sistemas de la ANC.

5.3.2 Casos de uso

A continuación se procedió a la definición de manera más precisa y concreta de las funcionalidades que debían ser provistas por el sistema a través de la definición de distintos casos de uso; identificando además la frontera del sistema y por consiguiente los actores que interactúan con el mismo iniciando los mencionados casos de uso. Este estudio fue registrado en el documento de *Casos de Uso* [3]. La definición de los mismos llevó a una mayor comprensión del sistema y en particular de los estados definidos para las solicitudes de delivery y sus transiciones; esto último se describe en el punto *Diagrama de Estados de una Solicitud de Servicio de Delivery*.

5.3.3 Modelo Conceptual

Luego de comprender más a fondo los procesos asociados al dominio del problema, es necesario descomponer éste identificando los conceptos y atributos significativos del dominio, así como las relaciones entre ellos. El resultado se expresa gráficamente con la generación de un *Diagrama Conceptual,* y su objeto es facilitar la comprensión de los principales componentes conceptuales del problema (incluidos en los requerimientos) y como se relacionan, sirviendo de base para la comunicación con los usuarios y como elemento previo fundamental para el diseño. A partir de dicho diagrama se genera el documento *Modelo Conceptual* [4]

5.3.4 Comportamiento del Sistema

Continuando con el análisis del sistema, se investiga y define el comportamiento del mismo como una “caja negra”. Los casos de uso se componen de eventos que son hechos externos de entrada al sistema (estímulos), que dan origen a operaciones de respuesta por parte de éste, o sea la acción que se ejecuta en respuesta. A través de los *Diagramas de Secuencia*, se expresan gráficamente para cada caso de uso, los eventos que fluyen entre los actores externos y el sistema. Por otro lado, mediante los *Contratos* (un contrato para cada operación definida), se describe el efecto que tiene cada operación sobre el sistema, describiendo como cambia el estado del mismo con la ejecución de dicha operación. Cada contrato describe aquello que se propone lograr la operación, definiendo el efecto de la misma a través de cambios de estado en los objetos del Modelo Conceptual (poscondiciones). Tanto los Diagramas de Secuencia como los Contratos, forman parte del documento de *Comportamiento del Sistema* [5]

5.3.5 Arquitectura de Pantallas

A esta altura, a partir de la documentación ya generada, parece apropiado la construcción de un prototipo con la finalidad de verificar con el cliente las funcionalidades - surgidas en el análisis - y su presentación en el producto final, obteniendo de éste una retroalimentación. Por las características del sistema, el único componente con interfaz gráfica y por lo tanto interacción con un usuario final, es el que corresponde a las funcionalidades de administración, por lo tanto dichas funcionalidades serían las únicas para las cuales se realizaría un prototipo. Teniendo en cuenta el perfil técnico del cliente a los cuales está dirigido, y para no orientar el prototipo a una implementación en particular al no estar definido todavía el lenguaje de programación a adoptar, se decidió crear un documento de *Arquitectura de Pantallas* [6] en el cual se define la secuencia de pantallas, describiendo exactamente el contenido de las mismas y las distintas opciones (ver punto *Arquitectura de Pantallas*)

El mencionado documento presenta las distintas funcionalidades de administración y su accesibilidad, dando una idea de la disposición de los datos proporcionados.

5.3.6 Diagramas de Interacción

La fase de Diseño está caracterizada por la elaboración de los *Diagramas de Interacción* [7], en los cuales se muestra gráficamente como los objetos interactúan entre ellos, para cumplir con los requerimientos. La interacción esta dada a través del flujo de mensajes entre instancias de objetos y la invocación de métodos. Para su realización fue necesario el estudio de los principios de asignación de responsabilidades y el uso de patrones de diseño. Estos diagramas son realizados a partir del *Modelo Conceptual*, del cual se obtienen los conceptos que se transformarán en clases; de los *Contratos* definidos para cada operación del sistema, de los cuales se identifican las responsabilidades y poscondiciones que determinan el flujo de mensajes y los objetos que los originan; y los *Casos de Uso*, a través de los cuales se obtiene más información acerca de las tareas que realizan los diagramas de interacción.

El Diseño final obtenido es un diseño detallado, y se modeló mediante diagramas de colaboración – es decir, expresado en formato de grafo o red - y en vista de la complejidad de los mismos, para una mayor claridad se extendió la notación de numeración utilizada [7]. Esta etapa normalmente exige mucho tiempo y esfuerzo, y debe ser realizada cuidadosamente, teniendo especial atención en la asignación de responsabilidades. Las responsabilidades se refieren a las obligaciones de un objeto respecto a su comportamiento - es decir algo que debe hacer consigo mismo o el iniciar una acción en otros objetos - las cuales son cumplidas a través de los métodos.
Los patrones de diseño son un conjunto de problemas tipo y su solución sugerida. Su objetivo es orientar en la asignación de responsabilidades, y su uso permite la obtención de un buen diseño. [1] Entre ellos se pueden destacar los más básicos:

- Experto
- Creador
- Alta Cohesión
- Bajo Acoplamiento
- Controlador

Es de destacar que estos diagramas fueron obtenidos luego de varios refinamientos y su realización llevó a una comprensión de la solución a un nivel muy cercano a su implementación. Los diagramas fueron agrupados en aquellos relativos a la aplicación en general, los relativos a la Recuperación y los asociados al Controlador2PC y su recuperación.

Una de las decisiones más importantes de esta etapa fue la relacionada al acceso a la Base de Datos, la cual es descrita en el punto *Manejo de los objetos y su impacto en una base de datos relacional*

5.3.7 Diagramas de clases

A partir de los diagramas de interacción, se elaboraron los diagramas de clases, que definen claramente las clases e interfaces a implementar. Los diagramas de interacción proporcionan las clases y métodos que intervienen en la solución; y a partir del modelo conceptual se obtienen algunos detalles para la definición de las mismas. Estos diagramas contiene además de las clases y sus métodos, los atributos e información del tipo de los mismos, su visibilidad y la navegación entre objetos.

En el documento de *Diagramas de Clases* [8] se presentan primero los métodos y atributos comunes a todas las clases, dividiéndolas en dos grupos: las clases comunes y las BDRClases (clases que manejan la interacción con la base de datos). Esta primera presentación es para no incluir dichos métodos y atributos en el resto de las clases, evitando así redundancia y simplificando el resto de las clases para poder concentrarse fácilmente en el resto de los atributos y métodos. Luego, dividimos el diagrama en dos grandes partes: lo relativo a la aplicación en sí y lo relativo al Controlador2PC.

La incorporación de los OID en los objetos para su identificación fue una importante decisión tomada en esta etapa, lo cual es tratado más en profundidad en el punto *Incorporación de OID para identificación de objetos*

Tanto por la elaboración del documento de Arquitectura de Pantallas, como por el pasaje por esta fase de Diseño, surge una retroalimentación para el análisis, lo cual obligó a la realización de una nueva iteración del ciclo de desarrollo. En este nuevo ciclo, se incorporaron algunos contratos y casos de uso administrativos, así como también relacionados con la recuperación de fallas y la interacción con los subsistemas.

5.3.8 Diagrama Entidad – Relación

Como paso previo al diseño del esquema físico de Base de Datos a utilizar, a partir del Diagrama de Clases, se elaboró un *Diagrama Entidad – Relación* [9] para modelar los elementos del diagrama de clases desde un enfoque relacional. Con el fin de poder automatizar la generación del diseño físico, dar más flexibilidad a la hora de realizar cambios e impactarlos en la Base de Datos, generar reportes asociados, y poder realizar Ingeniería Inversa, se utilizó la herramienta Designer/2000.

5.3.9 Estrategia de Implementación

Por ultimo, una vez culminado el diseño detallado, debido a limitantes de tiempo y recursos disponibles, se acoto la implementación de la aplicación de manera de obtener el producto de funcionalidad más básica a nuestro juicio. La intención fue implementar la parte más básica y difícil de la aplicación (exceptuando la Recuperación de fallas y el manejo del Controlador2PC), dejando las pautas para todo el resto de la implementación.

5.4 Algunos elementos del Análisis y Diseño

5.4.1 Casos de Uso

Al definir la frontera del sistema, es decir determinar que es interno al sistema y que externo, quedan determinados los actores o entidades externas al sistema que interactúan con el mismo iniciando secuencias de eventos correspondientes a distintos casos de uso. A continuación se exponen los actores de nuestro sistema, así como los casos de uso en los que intervienen los mismos.

5.4.1.1 Actor Protocolo

Este actor representa el rol que desempeña el protocolo de delivery y esta relacionado con todas las interacciones del mismo con la aplicación, que surgen de la interfaz del mismo

Casos de uso asociados a la firma y validación de documentos

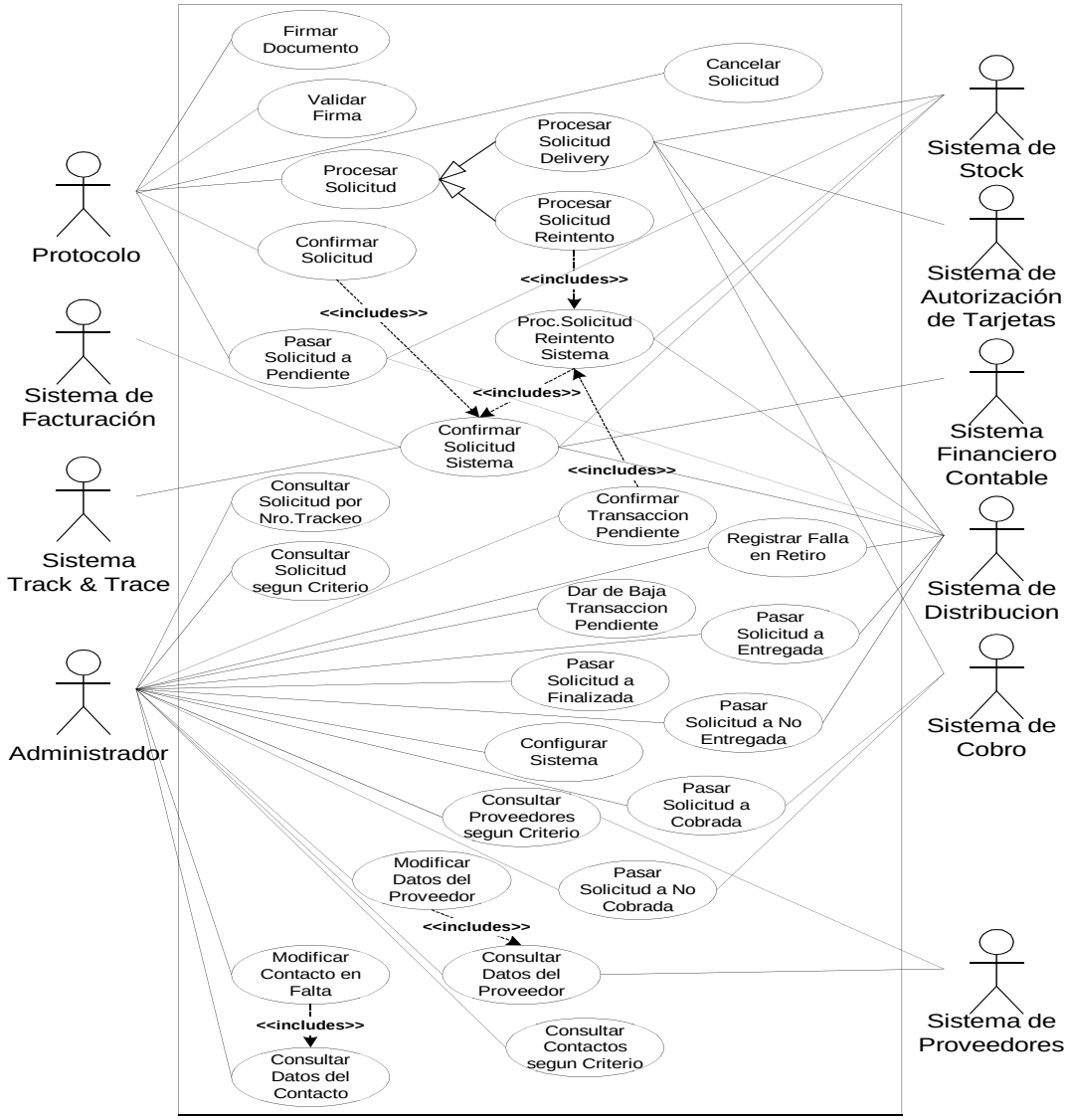
- Firmar Documento
- Validar Firma

Casos de uso asociados a solicitud de servicios

- Procesar Solicitud de Delivery
- Procesar Solicitud de Reintento
- Procesar Solicitud de Reintento Sistema
- Confirmar Solicitud
- Cancelar Solicitud
- Confirmar Solicitud Sistema
- Pasar Solicitud a Pendiente

5.4.1.2 Actor Administrador

Este actor representa al administrador del sistema cuyas interacciones con el sistema están determinadas por su responsabilidad de mantener el sistema funcional: configuración del sistema, mantenimiento del sistema, consultas básicas, etc.



Casos de uso relacionados con la administración del sistema

Funcionalidades básicas de administración del sistema

- Confirmar Transacción Pendiente
- Procesar Solicitud de Reintento Sistema
- Confirmar Solicitud Sistema
- Dar de Baja Transacción Pendiente
- Pasar Solicitud a Finalizada
- Pasar Solicitud a Cobrada
- Pasar Solicitud a No Cobrada
- Pasar Solicitud a Entregada
- Pasar Solicitud a No Entregada
- Registrar Falla en Retiro
- Modificar Contacto en Falta

Funcionalidades básicas de configuración del sistema

- Configurar Sistema
- Modificar Datos del Proveedor

- Modificar Servicios y Subservicios

Consultas básicas para la administración del sistema

- Consultar Solicitud por Nro.Trackeo
- Consultar Datos del Proveedor
- Consultar Proveedores según Criterio
- Consultar Datos del Contacto
- Consultar Contactos según Criterio

5.4.1.3 Actor Sistema de Distribución

Este actor representa el rol que desempeña el sistema de la ANC con el mismo nombre. El mismo es el encargado de todo lo relativo a la asignación de los recursos de distribución, la aceptación de los paquetes a entregar, las fechas de retiro y entrega, y las direcciones relacionadas. Sus interacciones con nuestro sistema definen la interfaz que dicho sistema proporcionara para el Sistema de Distribución.

Casos de uso relacionados con el registro de la entrega de los paquetes

- Pasar Solicitud a Entregada
- Pasar Solicitud a No Entregada
- Registrar Falla en Retiro

5.4.1.4 Actor Sistema de Cobro

Este actor representa el rol que desempeña el sistema de la ANC con el mismo nombre. El mismo es el encargado de todo lo relativo al ingreso de cobro, registro del mismo una vez realizado, validación del clientes como cobrables, etc. Sus interacciones con nuestro sistema definen la interfaz que dicho sistema proporcionara para el Sistema de Cobro.

Casos de uso relacionados con el registro del cobro de los paquetes

- Pasar Solicitud a Cobrada
- Pasar Solicitud a No Cobrada

5.4.1.5 Actor Sistema de Facturación

Este actor representa el rol que desempeña el sistema de la ANC con el mismo nombre. El mismo es el encargado de todo lo relativo a la facturación, generación de facturas, etc. La interacción con el mismo es iniciada únicamente por nuestro sistema (en los casos de uso que dicho Sistema de Facturación interviene) haciéndolo a través de la interfaz proporcionada por el Sistema de Facturación.

Casos de uso relacionados con la generación de facturas

- Confirmar Solicitud Sistema

5.4.1.6 Actor Sistema de Track & Trace

Este actor representa el rol que desempeña el sistema de la ANC con el mismo nombre. El mismo es el encargado de todo lo relativo al seguimiento de los envíos a través de un número de trackeo asociado a los mismos. La interacción con el mismo es iniciada únicamente por nuestro sistema (en los casos de uso que dicho Sistema de Track & Trace interviene) haciéndolo a través de la interfaz proporcionada por el Sistema de Track & Trace.

Casos de uso relacionados con la asignación de un Nro.Trackeo a una solicitud

- Confirmar Solicitud Sistema

5.4.1.7 Actor Sistema de Stock

Este actor representa el rol que desempeña el sistema de la ANC con el mismo nombre. El mismo es el encargado de todo lo relativo al stock de mercaderías de los proveedores en los locales de la ANC, registro y bajas de existencias, etc. La interacción con el mismo es iniciada únicamente por nuestro sistema (en los casos

de uso que dicho Sistema de Stock interviene) haciéndolo a través de la interfaz proporcionada por el Sistema de Stock.

Casos de uso relacionados con reservar, liberar y confirmar recursos de stock

- Procesar Solicitud de Delivery
- Pasar Solicitud a Pendiente
- Procesar Solicitud de Reintento Sistema
- Confirmar Solicitud Sistema

5.4.1.8 Actor Sistema de Autorización de Tarjetas de Crédito

Este actor representa el rol que desempeña el sistema de la ANC con el mismo nombre. El mismo es el encargado de todo lo relativo a la validación de las Tarjetas de Crédito, tarjetas aceptadas por la ANC, validación de tarjetas, mantenimiento de tarjetas fallidas, etc. La interacción con el mismo es iniciada únicamente por nuestro sistema (en los casos de uso que dicho Sistema de Autorización de Tarjetas de Crédito interviene) haciéndolo a través de la interfaz proporcionada por el Sistema de Autorización de Tarjetas de Crédito.

Casos de uso relacionados con la validación de una tarjeta de crédito

- Procesar Solicitud de Delivery

5.4.1.9 Actor Sistema Financiero Contable

Este actor representa el rol que desempeña el sistema de la ANC con el mismo nombre. En el mismo se registra la contabilidad de la ANC. La interacción con el mismo es iniciada únicamente por nuestro sistema (en los casos de uso que dicho Sistema Financiero Contable interviene) haciéndolo a través de la interfaz proporcionada por el Sistema Financiero Contable.

Casos de uso relacionados con la contabilización del cobro asociado a una solicitud

- Confirmar Solicitud Sistema

5.4.1.10 Actor Sistema de Proveedores

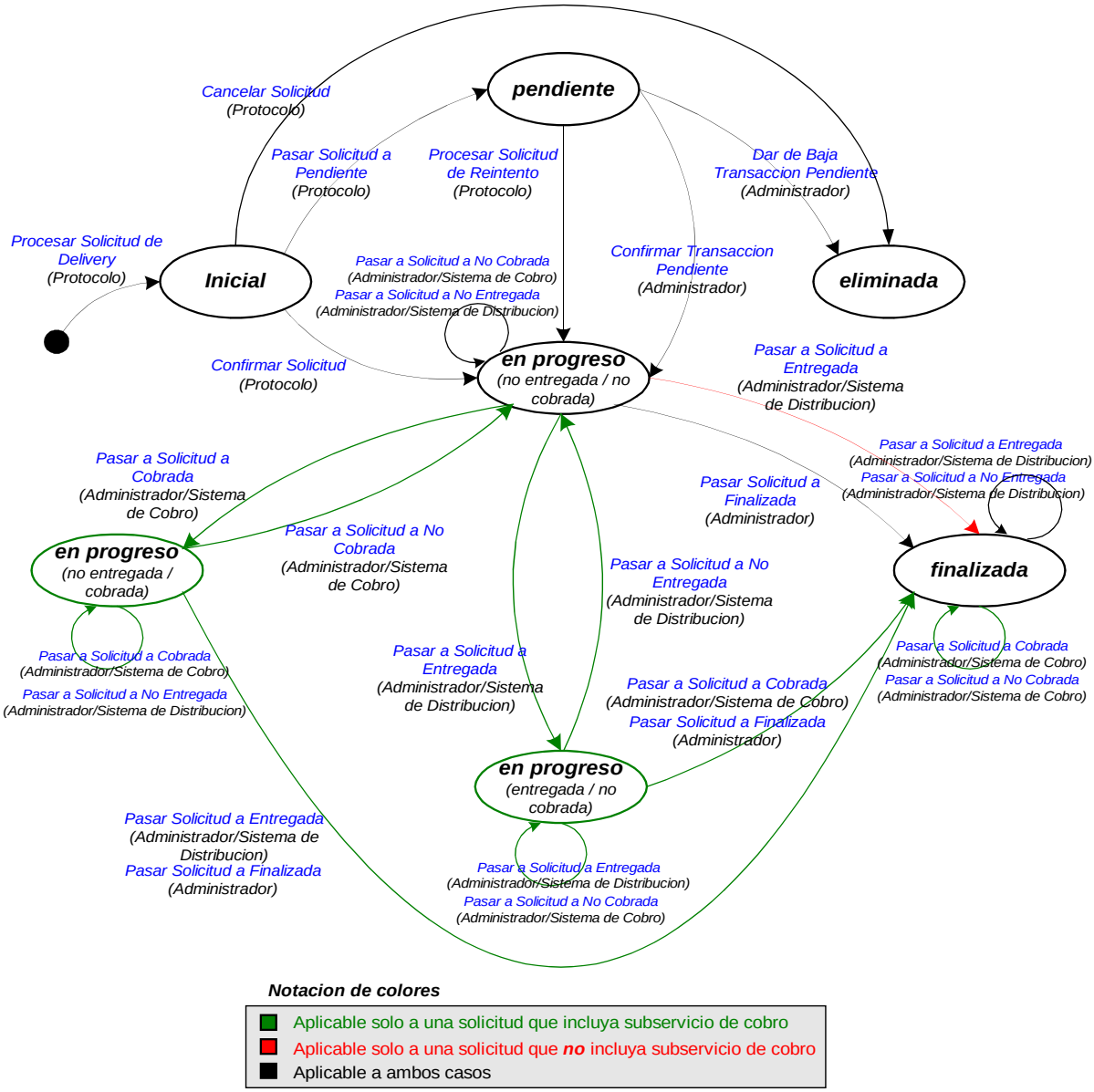
Este actor representa el rol que desempeña el sistema de la ANC con el mismo nombre. En el mismo se mantiene los datos básicos asociados a los proveedores que tienen contratos con la ANC. La interacción con el mismo es iniciada únicamente por nuestro sistema (en los casos de uso que dicho Sistema de Proveedores interviene) haciéndolo a través de la interfaz proporcionada por el Sistema de Proveedores.

Casos de uso relacionados con la obtención de datos básicos de los Proveedores

- Consultar Proveedores según Criterio
- Consultar Datos del Proveedor

5.4.2 Diagrama de estados de una Solicitud de Servicio de Delivery

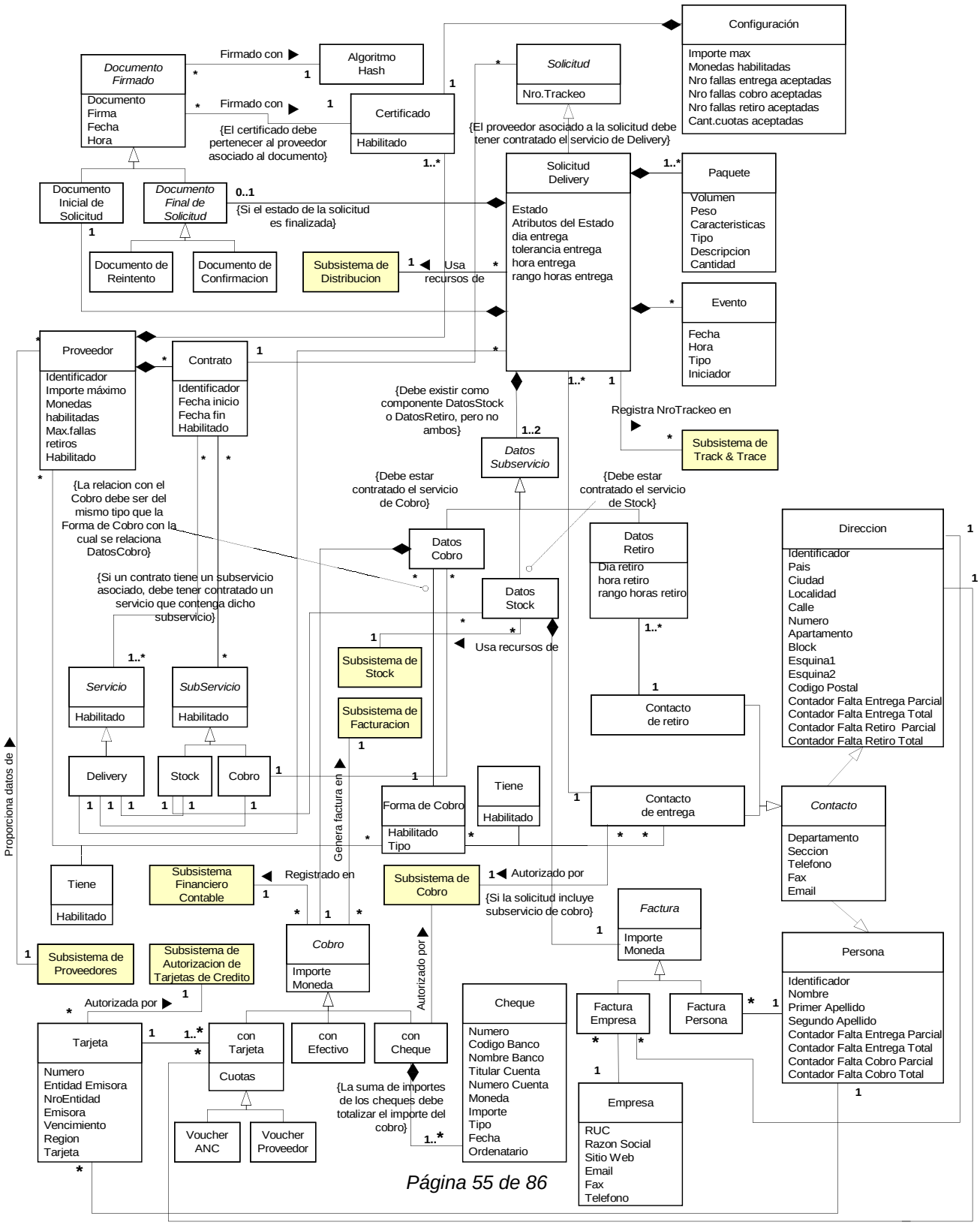
El siguiente diagrama representa los distintos estados en los cuales puede estar una solicitud de servicio de delivery, y sus transiciones provocadas por los distintos casos de uso, representando también quien puede iniciarlos.



Las etiquetas de las transiciones están compuestas por el caso de uso que la genera (en color azul) y quien la puede iniciar (en color negro y entre paréntesis).
El estado de una Solicitud de Servicio de Delivery, se compone del estado en sí más los atributos que pueden ser *cobrado* o *no cobrado* y *entregado* o *no entregado*.

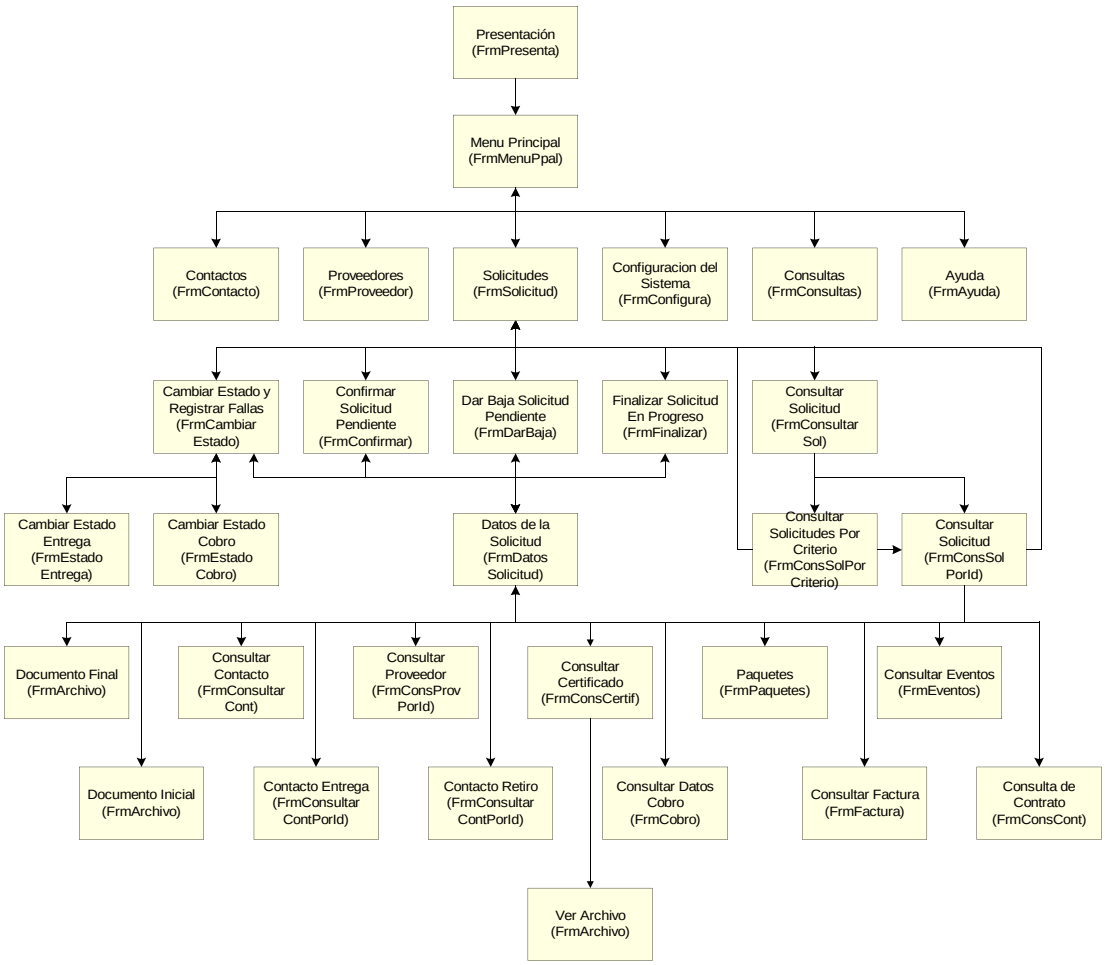
5.4.3 Modelo Conceptual

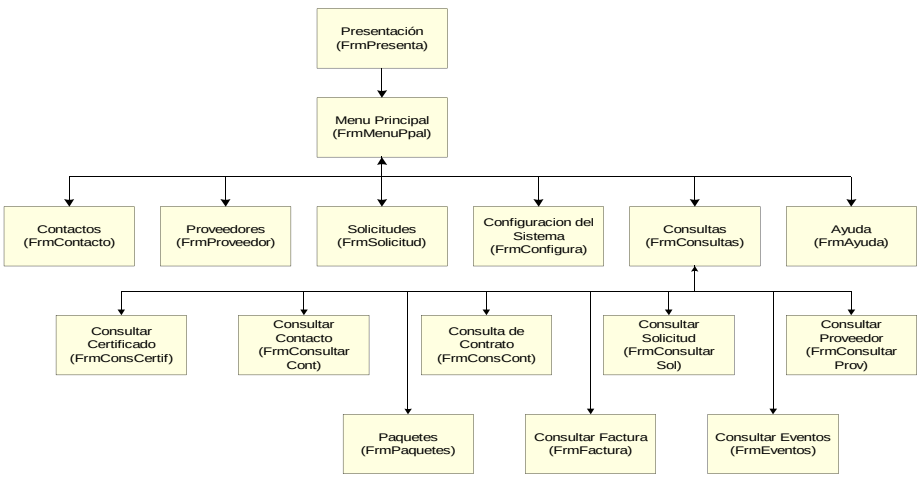
Aquí se presentan los principales conceptos del dominio del problema, sus atributos y como se relacionan entre sí. La cantidad de conceptos y sus variadas relaciones muestra el grado de complejidad.

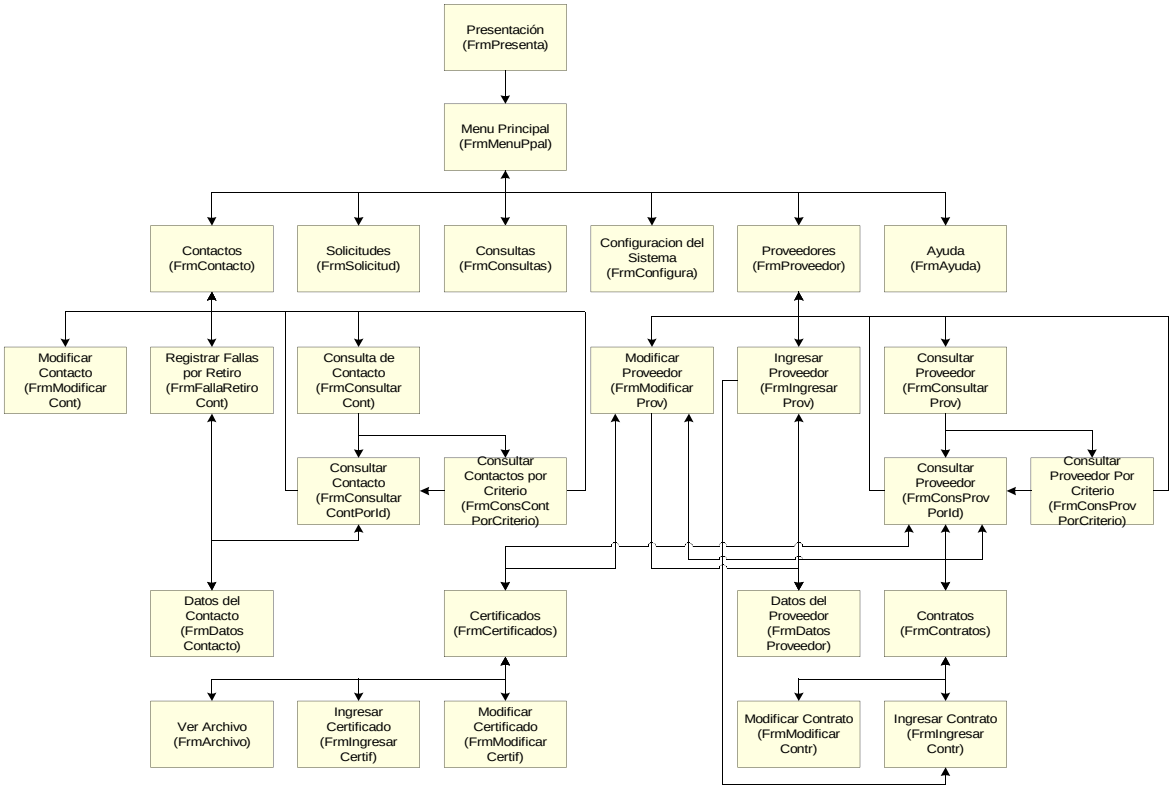


5.4.4 Arquitectura de Pantallas

Los siguientes cuadros exponen las posibilidades de navegación entre las pantallas definidas para las funcionalidades relativas a la administración:



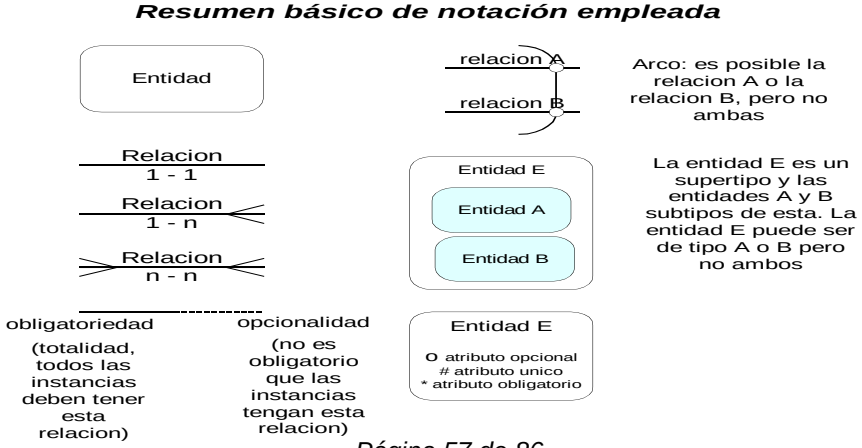


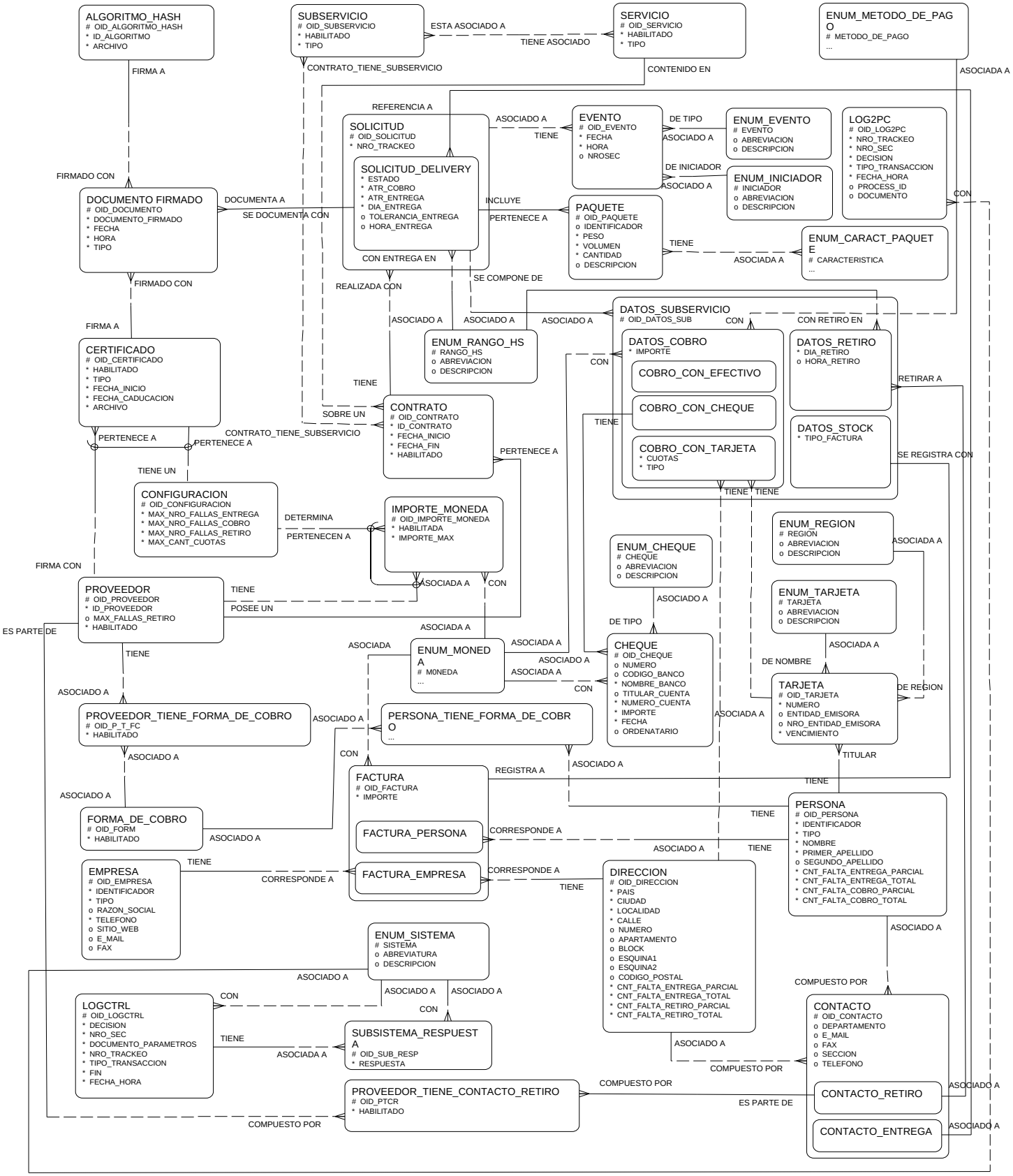


5.4.5 Diagrama Entidad – Relación

El Diagrama Entidad – Relación se obtuvo como mapeo del Diagrama de Clases. Normalmente se mapea una clase de objeto persistente a una entidad, y cada uno de sus atributos de tipos primitivos (número, string, booleano, etc) se convertirá en un atributo del Diagrama Entidad - Relación. Pero el mapeo no es tan directo cuando los atributos contenidos no son de tipos primitivos. Adicionalmente fue necesario considerar algunas complejidades del diseño de clases como la herencia múltiple, así como las limitantes de la notación del diagrama utilizado.
Desde la elaboración del Modelo Conceptual y durante todo el diseño, hasta el mapeo realizado al Diagrama Entidad – Relación fue considerada la futura extensión del sistema, sobre todo en los servicios y subservicios integrados.

Dado que la notación empleada en este diagrama difiere del clásico MER, se expondrán muy brevemente algunos detalles sobre la misma.




ALGORITMO_HASH
OID_ALGORITMO_HASH
* ID_ALGORITMO
* ARCHIVO
SUBSERVICIO
OID_SUBSERVICIO
* HABILITADO
* TIPO
ESTA ASOCIADO A
TIENE ASOCIADO
SERVICIO
OID_SERVICIO
* HABILITADO
* TIPO
ENUM_METODO_DE_PAGO
METODO_DE_PAGO
...
FIRMA A
CONTRATO_TIENE_SUBSERVICIO
CONTENIDO EN
ASOCIADA A
REFERENCIA A
SOLICITUD
OID_SOLICITUD
* NRO_TRACKEO
ASOCIADO A
TIENE
EVENTO
OID_EVENTO
* FECHA
* HORA
o NROSEC
DE TIPO
ASOCIADO A
ENUM_EVENTO
EVENTO
o ABREVIACION
o DESCRIPCION
LOG2PC
OID_LOG2PC
* NRO_TRACKEO
* NRO_SEC
* DECISION
* TIPO_TRANSACCION
* FECHA_HORA
o PROCESS_ID
o DOCUMENTO
CON
DE INICIADOR
ASOCIADO A
ENUM_INICIADOR
INICIADOR
o ABREVIACION
o DESCRIPCION
FIRMADO CON
SOLICITUD_DELIVERY
* ESTADO
* ATR_COBRO
* ATR_ENTREGA
* DIA_ENTREGA
o TOLERANCIA_ENTREGA
o HORA_ENTREGA
DOCUMENTO FIRMADO
OID_DOCUMENTO
* DOCUMENTO_FIRMADO
* FECHA
* HORA
* TIPO
DOCUMENTA A
SE DOCUMENTA CON
INCLUYE
PERTENECE A
PAQUETE
OID_PAQUETE
o IDENTIFICADOR
* PESO
* VOLUMEN
* CANTIDAD
o DESCRIPCION
TIENE
ASOCIADA A
ENUM_CARACT_PAQUETE
CARACTERISTICA
...
CON ENTREGA EN
SE COMPONE DE
REALIZADA CON
FIRMADO CON
FIRMA A
DATOS_SUBSERVICIO
OID_DATOS_SUB
CON
CON RETIRO EN
ASOCIADO A
ASOCIADO A
ASOCIADO A
ENUM_RANGO_HS
RANGO_HS
o ABREVIACION
o DESCRIPCION
DATOS_COBRO
* IMPORTE
DATOS_RETIRO
* DIA_RETIRO
o HORA_RETIRO
CON
COBRO_CON_EFECTIVO
COBRO_CON_CHEQUE
COBRO_CON_TARJETA
* CUOTAS
* TIPO
DATOS_STOCK
* TIPO_FACTURA
RETIRAR A
TIENE
CERTIFICADO
OID_CERTIFICADO
* HABILITADO
* TIPO
* FECHA_INICIO
* FECHA_CADUCACION
* ARCHIVO
TIENE
SOBRE UN
CONTRATO
OID_CONTRATO
* ID_CONTRATO
* FECHA_INICIO
* FECHA_FIN
* HABILITADO
PERTENECE A
CONTRATO_TIENE_SUBSERVICIO
PERTENECE A
PERTENECE A
SE REGISTRA CON
TIENE UN
CONFIGURACION
OID_CONFIGURACION
* MAX_NRO_FALLAS_ENTREGA
* MAX_NRO_FALLAS_COBRO
* MAX_NRO_FALLAS_RETIRO
* MAX_CANT_CUOTAS
DETERMINA
PERTENECEN A
IMPORTE_MONEDA
OID_IMPORTE_MONEDA
* HABILITADA
* IMPORTE_MAX
TIENE
TIENE
ENUM_REGION
REGION
o ABREVIACION
o DESCRIPCION
ASOCIADA A
ENUM_CHEQUE
CHEQUE
o ABREVIACION
o DESCRIPCION
ASOCIADO A
ENUM_TARJETA
TARJETA
o ABREVIACION
o DESCRIPCION
FIRMA CON
PROVEEDOR
OID_PROVEEDOR
* ID_PROVEEDOR
o MAX_FALLAS_RETIRO
* HABILITADO
TIENE
POSEE UN
ASOCIADA A
CON
ASOCIADA A
ASOCIADA A
DE TIPO
ASOCIADA A
ES PARTE DE
ASOCIADA
ENUM_MONEDA
MONEDA
...
ASOCIADA A
ASOCIADO A
CHEQUE
OID_CHEQUE
o NUMERO
o CODIGO_BANCO
* NOMBRE_BANCO
o TITULAR_CUENTA
* NUMERO_CUENTA
* IMPORTE
* FECHA
o ORDENATARIO
DE NOMBRE
ASOCIADA A
TARJETA
OID_TARJETA
* NUMERO
o ENTIDAD_EMISORA
o NRO_ENTIDAD_EMISORA
* VENCIMIENTO
DE REGION
TIENE
ASOCIADO A
PROVEEDOR_TIENE_FORMA_DE_COBRO
OID_P_T_FC
* HABILITADO
ASOCIADO A
PERSONA_TIENE_FORMA_DE_COBRO
...
CON
ASOCIADO A
TITULAR
FACTURA
OID_FACTURA
* IMPORTE
REGISTRA A
TIENE
ASOCIADO A
FORMA_DE_COBRO
OID_FORM
* HABILITADO
ASOCIADO A
FACTURA_PERSONA
CORRESPONDE A
TIENE
PERSONA
OID_PERSONA
* IDENTIFICADOR
* TIPO
* NOMBRE
* PRIMER_APELLIDO
o SEGUNDO_APELLIDO
* CNT_FALTA_ENTREGA_PARCIAL
* CNT_FALTA_ENTREGA_TOTAL
* CNT_FALTA_COBRO_PARCIAL
* CNT_FALTA_COBRO_TOTAL
ASOCIADO A
EMPRESA
OID_EMPRESA
* IDENTIFICADOR
* TIPO
o RAZON_SOCIAL
* TELEFONO
o SITIO_WEB
o E_MAIL
o FAX
TIENE
CORRESPONDE A
FACTURA_EMPRESA
CORRESPONDE A
TIENE
DIRECCION
OID_DIRECCION
* PAIS
* CIUDAD
* LOCALIDAD
* CALLE
o NUMERO
o APARTAMENTO
o BLOCK
o ESQUINA1
o ESQUINA2
o CODIGO_POSTAL
* CNT_FALTA_ENTREGA_PARCIAL
* CNT_FALTA_ENTREGA_TOTAL
* CNT_FALTA_RETIRO_PARCIAL
* CNT_FALTA_RETIRO_TOTAL
COMPUESTO POR
ENUM_SISTEMA
SISTEMA
o ABREVIATURA
o DESCRIPCION
ASOCIADO A
ASOCIADO A
ASOCIADO A
CON
CON
LOGCTRL
OID_LOGCTRL
* DECISION
* NRO_SEC
* DOCUMENTO_PARAMETROS
* NRO_TRACKEO
* TIPO_TRANSACCION
* FIN
* FECHA_HORA
TIENE
ASOCIADA A
SUBSISTEMA_RESPUESTA
OID_SUB_RESP
* RESPUESTA
ASOCIADO A
COMPUESTO POR
CONTACTO
OID_CONTACTO
o DEPARTAMENTO
o E_MAIL
o FAX
o SECCION
o TELEFONO
PROVEEDOR_TIENE_CONTACTO_RETIRO
OID_PTCR
* HABILITADO
COMPUESTO POR
ES PARTE DE
CONTACTO_RETIRO
ASOCIADO A
CONTACTO_ENTREGA
ASOCIADO A
COMPUESTO POR

A partir de este diagrama se generó el diseño físico del esquema de la Base de Datos a utilizar. El mismo puede consultarse en el *Anexo G: Detalles de implementación* [10] y un detalle mayor puede obtenerse a través del uso de la herramienta Designer/2000, incluso pueden generarse distintos reportes.

5.5 Desafíos

En cuanto a los desafíos al elaborar la aplicación, se puede incluir entre los mismos el cumplir con las responsabilidades ya mencionadas en el punto *Necesidad de una aplicación* . Algunos de los cuales se explicarán en más detalle en los puntos que continúan (*Recuperación de caídas del sistema* y *Controlador2PC*). Estos puntos también incluyen la solución encontrada para otros problemas más puntuales como ser el problema de la persistencia de los objetos en una base de datos relacional (*Manejo de los objetos y su impacto en una base de datos relacional*) y la gestión de las conexiones a dicha base (*Interacción de las clases para la gestión de las conexiones a la Base de Datos*) con el consiguiente logro de eficiencia e independencia de la base de datos seleccionada, así como el problema del mapeo entre los objetos y sus tablas asociadas y la flexibilidad lograda con el uso de OIDs (*Incorporación de OID para identificación de objetos*)

También puede considerarse como desafío más general el encontrar la mejor solución para la aplicación, lo cual fue en parte logrado a través del uso de la metodología de desarrollo adoptada y un estudio cuidadoso del negocio y su importancia.

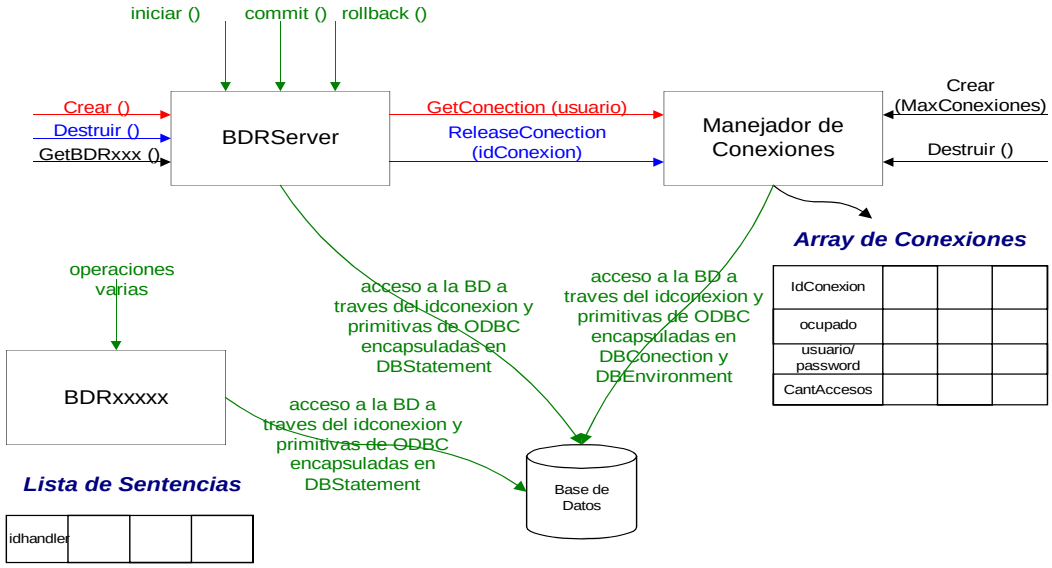
5.5.1 Manejo de los objetos y su impacto en una base de datos relacional

Al crear una instancia es creado el objeto en memoria y no en la base de datos. El objeto no es persistente, no esta mantenido todavía en algún almacenamiento no volátil. Cualquier cambio a una instancia no modifica los datos respectivos en la base de datos.
Una instancia de un objeto, es impactada (se hace persistente) a la base de datos a través de los métodos agregar o modificar (según si se esta agregando una instancia nueva en la BD o si se esta modificando una existente) de las instancias "BDR" correspondientes (similar al enfoque de Broker mencionado en [1]), y confirmados en la base luego de un commit al final de la transacción.
Es importante destacar que las clases BDR que estamos manejando representan una interfaz a la colección de objetos mantenidos en la base de datos relacional por medio de tablas (la tabla esta asociada a una clase, y contienen un registro por cada objeto).
Por otra parte, el método buscar de cada BDR, se encarga de obtener el registro correspondiente de la base de datos y materializarlo en una instancia del objeto que es devuelta.
A continuación se verán los detalles acerca de el manejo de las conexiones a la base y los objetos que surgen a partir de este.

5.5.2 Interacción de las clases para la gestión de las conexiones a la Base de Datos

El acceso a la base es manejado a través de los BDRServer, que inician una conexión virtual a través de la clase Manejador de Conexiones que administra un pool de conexiones asignando temporalmente y a medida de la necesidad de una conexión a cada BDRServer. A través del BDRServer, se obtiene primero una conexión y luego se obtienen los objetos "BDR*clase*" (por ejemplo BDREvento si dicho BDR esta asociado a la clase Evento) que participarán en las transacciones; además de iniciarse las transacciones y commitearse o rollbackearse. A su vez los "BDR*clase*" se utilizan para todas las operaciones sobre la base de datos que estén relacionadas a la clase *clase*, y de esta manera se encapsulan los accesos a las tablas relacionadas a dicha clase.

La idea general del Manejador de Conexiones, es que mantenga una cantidad de conexiones simultáneas (parametrizable). Cuando se solicita una conexión, *reutilizará* una existente *que no este siendo utilizada*, y en caso de pertenecer a otro usuario, y no haber conexiones no inicializadas, desconectará la menos accedida. En el caso de que las conexiones estén siendo todas utilizadas quedará esperando a través de la utilización de semáforos.



Cada operación esta definida en una sección crítica, pues el manejador de conexiones es una variable global. Por otro lado, en cada BDR*clase*, se mantiene una lista de sentencias que contiene los punteros al handler de cada sentencia para la cual se necesita un cursor.
El acceso final a la base de datos se logra a través de tres objetos que encapsulan las primitivas ODBC, ellos son:

- ***DBStatment*** (Manejado desde los BDR*clase*, tiene métodos como crear, ejecutar, fetch, etc)
- ***DBConection*** (Manejado desde el Manejador de Conexiones, posee métodos como conectar, desconectar, conectado?, etc)
- ***DBEnvironment*** (Manejado desde DBConection)

5.5.3 Incorporación de OID para identificación de objetos

Normalmente los objetos persistentes cuentan con un identificador único universal para poder individualizarlos. Este identificador es conocido como OID (Object IDentity), el cual es normalmente un valor alfanumérico.

En este caso, toma una importancia particular por el uso de una Base de Datos Relacional. Es conveniente contar con un identificador que relacione los objetos con los registros correspondientes en la base de datos, y a la vez se asegure que no existan objetos duplicados.

Las tablas en una base de datos relacional poseen una clave primaria para identificar los registros contenidos en las mismas, pero ésta se diferencia de un OID de una Base de datos Orientada a Objetos pues dicho OID es universal, único en toda la base, mientras que la clave primaria es necesario que sea única solo en la tabla en cuestión. La idea entonces es que todo objeto se identifique con el OID y a su vez se corresponda con el registro de la tabla correspondiente, es decir exista un mapeo entre ellos.

En este proyecto, se utilizaron OIDs únicos en toda la base relacional como claves primarias de las tablas correspondientes, de manera que la identificación de los objetos en la base no dependa de atributos particulares. Esto otorga flexibilidad pues si en un futuro cambian los atributos de un objeto - en particular los atributos que lo identifican - no es necesario repercutir estos cambios en todas las tablas que hagan referencia a la tabla del objeto en cuestión (cambiar su clave foránea)

Como aspecto negativo del OID se puede destacar las complicaciones que trae por ejemplo en la herencia múltiple: se hereda el OID? que OID se hereda?; o en el caso de las búsquedas o consultas, que normalmente no se busca un objeto por su OID. Se resolvió el primer problema heredando los OID y creando a su vez un nuevo OID que asocia ambas herencias. Para minimizar el impacto del segundo problema, se utilizarán índices por los campos normalmente consultados y se ofrecen métodos que a partir de los atributos que normalmente identifican al objeto se obtiene el OID y viceversa.

5.5.4 Controlador2PC

Para que las operaciones que involucran diversos Sistemas de la ANC sean transacciones atómicas, fue necesario la utilización de un protocolo Two Phase Commit (2PC) a través de un Controlador que centraliza dichas transacciones. [11] Todas las transacciones iniciadas por el sistema y que involucren distintos subsistemas son ejecutadas a través del Controlador, el cual debe cumplir el ***rol controlador*** del protocolo 2PC [12]. De esta manera se asegura el cumplimiento de las propiedades ACID (ver *Capitulo 4: Aspectos relativos a la Especificación del Protocolo*) de toda transacción [13] [14]

Básicamente consiste en realizar la transacción en dos etapas, reservando primero los recursos de otros subsistemas y luego procediendo a su confirmación o liberación. Estas acciones son realizadas a través de una interfaz ofrecida por el Controlador para tales efectos. Adicionalmente el Controlador provee una interfaz para la consulta del estado de una transacción PreCommiteada.

La reserva de recursos se realiza a través de la interfaz ***ReservarRecursos*** del Controlador, y consiste en realizar un ***PreCommit*** con cada uno de los subsistemas involucrados (pasados como parámetro). Al obtener un resultado positivo de esta operación, se tiene la certeza de que los subsistemas participantes han aceptado todos la transacción, estando en condiciones de confirmarla. Luego, a través de la interfaz ***ConfirmarRecursos***, es posible confirmar dichos recursos, realizando un ***Commit*** de una transacción previamente PreCommiteada con *ReservarRecursos*. En caso de no poder comunicar el Commit a un subsistema por fallas de dicho subsistema, es responsabilidad del mismo consultar al Controlador por el estado. Por otro lado, en caso de necesidad de abortar una transacción anteriormente Precommiteada con *ReservarRecursos*, es posible a través de la interfaz ***LiberarRecursos***, con la cual se realiza un ***Abort (Rollback)*** de la misma. En caso de no poder comunicar el Abort a un subsistema por fallas de dicho subsistema, es responsabilidad del mismo consultar al Controlador por el estado de dicha transacción.
Por otro lado la interfaz ***ConsultarEstadoTransaccion*** es proporcionada para ser invocada por un participante cuando dicho participante se recupera de una caída y no puede decidir en forma autónoma el resultado de una transacción, por lo cual debe consultar el resultado de la misma al Controlador.

La finalidad de permitir al sistema que inicia una transacción distribuida, realizar él mismo cada etapa (precommit y luego commit o abort) es para otorgar flexibilidad, por ejemplo para el caso del protocolo de delivery definido, una transacción reserva los recursos (precommit) y otra los confirma (commit) o libera (abort).

5.5.5 Recuperación de caídas del sistema

La necesidad de una lógica de recuperación de caídas del sistema surge a partir de la existencia de las ya mencionadas transacciones distribuidas – aquellas que involucran a diversos sistemas de la ANC - realizadas a través del Controlador2PC. Cuando se realiza una transacción directamente en una base de datos y el sistema cae, dicha transacción si no estaba culminada (commit) es deshecha (rollback). Pero al realizar una transacción entre diversos sistemas a través del Controlador2PC, esta transacción no es automáticamente deshecha (“rollbackeada”). Si esto sucede, podrían quedar - por ejemplo - recursos reservados en los distintos sistemas, pero la transacción haber sido deshecha en el sistema original y por lo tanto quedaría el estado de la transacción inconsistente en el sistema original con respecto al resto y no se liberarían estos recursos.

Para solucionar lo anterior, toda transacción del sistema que utilice el Controlador2PC, debe guardar en un log una referencia a la operación realizada [11], de manera que en caso de caída del sistema, al levantarse pueda consultar dicho log y determinar las transacciones no culminadas y su estado asociado, y entonces culminarlas o deshacerlas, consultando eventualmente al Controlador2PC el estado de dichas transacciones. Una vez recuperada una transacción, es eliminado el registro del log.

Surgen entonces una serie de escenarios de casos de recuperación que son considerados (por más detalles consultar el documento *Transacciones del Sistema* [11] y el diseño detallado en [7]), en los cuales se describe como realizar la recuperación.
En principio la recuperación se realiza al levantar el sistema pero podría realizarse periódicamente con el sistema funcionando, por ejemplo para el caso de procesos del sistema que no respondan.

5.6 Características

Además de cumplir con todos los Desafíos planteados anteriormente, es importante destacar las siguientes características de la aplicación:

- ***Control de faltas:*** El sistema provee controles relativos a la deshabilitación automática de proveedores y contactos implementando un mecanismo de registro de faltas. Los tipos de faltas considerados son las faltas por cobro y entrega no realizada a los contactos de entrega (cliente final), y falta por retiro no realizado a los contactos de retiro (normalmente Proveedores). Esto se realiza manteniendo contadores parciales y totales, los primeros pueden ser reseteados por el administrador para volver a habilitar al proveedor o contacto, y los segundos no se resetean y se mantienen como datos estadísticos.

- ***Control de topes:*** Para proveer mas flexibilidad al sistema, se implementó un mecanismo de topes en los montos de las solicitudes a cobrar. Dichos topes son establecidos tanto a nivel del sistema, como a nivel de proveedor y los mismos son para los distintos tipos de moneda configurados.

- ***Habilitaciones y Deshabilitaciones:*** El sistema permite al administrador habilitar o deshabilitar en forma manual Proveedores, Servicios, Subservicios, Certificados de un proveedor, Contratos de un proveedor, Formas de Cobro en general o asociadas a un Proveedor o a un Contacto de Entrega.

- ***Manejo de Certificados:*** Para la validación de la Firma Digital de los distintos documentos recibidos, se comprueba que los Certificados utilizados estén almacenados y habilitados en la base de datos del Sistema.

- ***Manejo de Eventos:*** Se registra un evento por cada suceso que afecta de alguna manera a una solicitud realizada y registrada en la Base de Datos del Sistema. Dichos eventos son generados especialmente para su uso por el sistema de Track & Trace en el seguimiento de las solicitudes realizadas.

- ***Controles varios:*** Además de los controles relativos a la habilitación y deshabilitación ya mencionada, en particular se controla la Fecha de Caducación de los Contratos utilizados, Tarjetas de Crédito permitidas y su Región, Tipos de Cheque permitidos, así como Monedas definidas. Además de los controles anteriormente mencionados, se realizan controles a través de la interacción con los subsistemas de la ANC (por ejemplo control de validación de dirección de entrega, autorización de cobro al cliente, etc).

5.7 Referencias bibliográficas

[1] *Applying UML and Patterns. An Introduction to Object-Oriented Analysis and Design* – Craig Larman
[2] Anexo D: Especificación de Requerimientos
[3] Anexo F: Análisis y Diseño de la aplicación y su Recuperación, documento Casos de Uso
[4] Anexo F: Análisis y Diseño de la aplicación y su Recuperación, documento Modelo Conceptual
[5] Anexo F: Análisis y Diseño de la aplicación y su Recuperación, documento Comportamiento del Sistema
[6] Anexo F: Análisis y Diseño de la aplicación y su Recuperación, documento Arquitectura de Pantallas
[7] Anexo F: Análisis y Diseño de la aplicación y su Recuperación, documento Diagramas de Interacción
[8] Anexo F: Análisis y Diseño de la aplicación y su Recuperación, documento Diagrama de Clases
[9] Anexo F: Análisis y Diseño de la aplicación y su Recuperación, documento Diagrama Entidad – Relación
[10] Anexo G: Detalles de implementación
[11] Anexo F: Análisis y Diseño de la aplicación y su Recuperación, documento Transacciones con Subsistemas
[12] Anexo B: Algunos conocimientos previos requeridos, documento Protocolo 2PC
[13] *Concurrency Control and Recovery in Database Systems* – Bernstein, Hadzilacos, Goodman
[14] *Oracle8 Distributed Database Systems, Release 8.0*
[15] Anexo H: Administración, documentos de Actas y Registros de Actividad

CAPITULO 6

6. Aspectos relativos a la Implementación

En el presente capítulo se presenta una reseña de las tecnologías y herramientas utiilzadas, destacando las cualidades de las mismas y el criterio general utilizado para su selección. Luego se resaltan algunos puntos importantes relativos a la implementación realizada (por más detalles consultar [1]), en particular la arquitectura desde esta perspectiva, destacando los diversos componentes, así como otras consideraciones.

6.1 Reseña de tecnologías y herramientas utilizadas

Durante el transcurso del proyecto se utilizaron varias tecnologías entre las que destacamos XML como el lenguaje estándar para la definición de los documentos intercambiados, SSL como protocolo de seguridad para asegurar la confidencialidad, integridad, autenticación y no repudio de los documentos intercambiados, mediante la utilización de Criptografía, Firmas Digitales y Certificados Digitales.

Otras tecnologías y herramientas, así como el software de base utilizado, se describen a continuación.

6.1.1 Elección de las herramientas a utilizar

Dada las características del sistema diseñado, la implementación del mismo implicaba una amplia utilización de herramientas estándares del mercado. Resultaba claro que la elección de las herramientas a utilizar era un aspecto crucial para una exitosa implementación del sistema. A tal fin se definió un conjunto de cualidades que serían deseables que las herramientas proporcionasen, y de esta forma poder realizar una selección objetiva de las mismas.

Las cualidades deseadas eran las siguientes:

- ***Confiable, Robusto y Eficiente***: Por las características del sistema diseñado el mismo debía ser lo mas confiable, robusto y eficiente posible; por lo cual las herramientas a utilizar debían a la ves cumplir y posibilitar dichas características.
- ***Portable o Multiplataforma***: Que esté disponible en la mayor cantidad de plataformas posibles, en particular las mas usadas y las encontradas en la ANC.
- ***Actual***: Que sea una versión actual, de forma de asegurar que incorporase las últimas ventajas.
- ***Funcionalidades***: Que posea las características o funcionalidades requeridas por el sistema a implementar y que posea la mayor cantidad de características.
- ***Estándar***: Que siga los estándares actualmente utilizados y las recomendaciones de futuros estándares prometiendo un desarrollo futuro acorde a las tendencias.

- ***Amplia aceptación***: Que esté disponible y sea aceptado y utilizado ampliamente.
- ***Respaldo***: Que esté respaldado por empresas u organizaciones importantes dentro del mercado.
- ***Open Source***: Que en lo posible sea una distribución gratuita y Open Source.

6.1.2 Elección de la plataforma

La elección de la plataforma a utilizar estaba condicionada por la ANC, la cual podía ser AIX o Windows NT. Al ser la ANC indiferente a la plataforma a utilizar el grupo de Taller 5 decidió adoptar ***Windows NT*** por ser más amigable, y estar disponible para los integrantes del grupo del Taller 5.

6.1.3 Elección del lenguaje de especificación del protocolo

La especificación del protocolo de delivery se realizó utilizando un lenguaje estándar de especificación de protocolos, de forma de lograr una especificación formal que asegurase la claridad y consistencia de la misma. Además como la especificación del protocolo debía ser de dominio público, se buscó un lenguaje que fuera a su vez fácil de entender y ampliamente utilizado. El lenguaje elegido para tal fin fue SDL (Specification and Description Language) [13], para más información consultar el *Capítulo 7: Planificación y Administración, punto Metodología utilizada para la especificación del protocolo* y [14]

6.1.4 Elección del lenguaje de modelado para el Análisis y Diseño

Se utilizó la notación estándar ***UML*** (Unified Modeling Language) [15] la cual permite la construcción de modelos de sistemas utilizando conceptos orientados a objetos. Más específicamente, UML permite la visualización, especificación, construcción y documentación de los elementos del sistema de software a través de todo su ciclo de vida, y es considerado elegante, expresivo y flexible. Se ha convertido en un estándar *de facto* de la industria ampliamente adoptado y por lo tanto sirve para una comunicación eficaz y uniforme. Por más información referirse al *Capítulo 5: Aspectos relativos al Análisis y Diseño de la aplicación, punto Que metodología utilizar?* y el *Capítulo 7: Planificación y Administración, punto Metodología de Análisis y Diseño de la aplicación* y [16]

6.1.5 Elección del lenguaje de programación y el ambiente de desarrollo.

En la elección del lenguaje de programación se consideró que el mismo estuviera soportado por un ambiente de desarrollo amigable y eficiente, en el cual se integrasen todas las funcionalidades necesarias para el proceso de implementación. Además como el diseño del sistema a implementar fue un diseño orientado a objetos, se buscaba un lenguaje también orientado a objetos.

En una primer instancia se obtuvieron como lenguajes posibles de ser utilizados Java y C++, los cuales comparten muchas de sus características. Las diferencias cruciales radican en la eficiencia, robustez y recursos utilizados. Java al ser un lenguaje interpretado es notoriamente menos eficiente que C++, también requiere mucho mas recursos, tanto de memoria como de CPU que C++, y por último hay que destacar que actualmente Java no es una plataforma confiable. Por otro lado la diferencia a destacar a favor de Java es que es un lenguaje multiplataforma pues un mismo código puede ser ejecutado en plataformas diferentes sin necesidad de modificarlo. Sin embargo a pesar que C++ no es multiplataforma, como C++ es un lenguaje estándar mundialmente aceptado y utilizado en todas las plataformas existentes, el código obtenido es fácilmente portable a cualquier plataforma.

Por lo anterior se decidió adoptar como lenguaje de programación, el lenguaje C++ y como herramienta de desarrollo ***Visual C++ 6.0***; por ser esta última un excelente ambiente de desarrollo en el cual se integran todas las funcionalidades necesarias. Entre ellas se pueden resaltar: editor de texto con sintaxis de colores, editores de componentes gráficos y un potente depurador.

6.1.6 Elección de la herramienta de seguridad

Según lo especificado en el documento de *Detalles de Implementación* [1], la capa de seguridad se implementa mediante el protocolo ***SSL***. Para ello se adoptó la herramienta ***OpenSSL*** [2].

OpenSSL es una herramienta Open Source que implementa los protocolos ***Secure Socket Layer*** (SSL v2/v3) y ***Transport Layer Security*** (TLS v1) así como también una librería de criptografía fuerte (strong cryptography) de propósito general. OpenSSL es la continuación de la excelente librería ***SSLeay*** [3] desarrollada por Eric A. Young y Tim J. Hudson. OpenSSL se distribuye bajo una licencia similar a la de Apache, lo cual básicamente significa que puede ser utilizada libremente tanto para fines comerciales como no comerciales. La versión utilizada correspondió a la versión OpenSSL 0.9.6.

OpenSSL es ampliamente utilizada a nivel mundial por todo tipo de organizaciones, lo que da una prueba de su confiabilidad. Un ejemplo de ello es el servidor web ***Apache*** [4], uno de los servidores web más utilizados a nivel mundial, y el más utilizado en entornos UNIX. El módulo ***mod_ssl*** [5] proporciona un módulo de criptografía fuerte al Apache a través de los protocolos SSL y TLS, utilizando OpenSSL. Otro ejemplo es el Apache-SSL [6], un servidor web basado en Apache y OpenSSL.

6.1.7 Elección del parser de XML

Según lo especificado en el documento *Detalles de Implementación* [1], toda la información intercambiada por el protocolo de delivery está constituida por documentos XML, los cuales se definen mediante DTDs. Dichos documentos deben ser parseados y validados, para lo cual se adoptó la herramienta ***Xerces-C*** [8] de Apache [4]. La versión utilizada correspondió a la versión Xerces-C 1.4.0.

Xerces-C es un parser XML con validación escrito en un subconjunto portable de C++. Xerces-C es Open Source y cumple con la especificación XML 1.0 y los estándares asociados (DOM 1.0, DOM 2.0, SAX 1.0, SAX 2.0, Namespaces).

Para la prueba de la validación de documentos XML contra los DTDs definidos se utilizaron además otros parsers como el ***MSXML*** versión 3.0 [12] y los parsers incluidos en las herramientas mencionadas en el siguiente punto.

6.1.8 Elección de generadores de documentos XML y DTDs

Se evaluaron distintas herramientas para la generación y pruebas de DTDs y documentos XML , entre las más destacadas se encuentran Near&Far, XmlAuthority, XmlViewer, XMLPro2 y Xeena. [11] Se comenta a continuación algunas características de las herramientas elegidas:

Xml Authority: Espectacular herramienta gráfica de generacion de DTDs, distintos tipos de XML Schema y conversión entre ellos. Además, maneja distintos tipos de datos y para el caso de DTD los simula a través del uso de una notación estándar (pero la aplicación deberá ser la responsable de chequear su validez), permitiendo luego la conversión a schema para automatizar así los tipos extendidos. Se utilizó una versión demo, válida por un tiempo limitado, la cual permite la generación de DTDs y su prueba, así como la generación de documentos de ejemplo randómicos, permitiendo además la validación de la construcción del DTD.

Xeena: Permite la creación de documentos XML partiendo de un DTD definido. Permite no solo generar el documento en cuestión, sino también verificar los distintos posibles documentos que se pueden generar con el DTD definido. Es una herramienta gráfica muy simple e intuitiva.

La única desventaja de ambas herramientas es su pobre performance y su alto grado de errores posiblemente atribuibles a su implementación en Java.

6.1.9 Elección de la Base de Datos y su mecanismo de acceso

La elección de la Base de Datos a utilizar estaba condicionada por la ANC, la cual podía ser ***SQLServer*** u ***Oracle***. En pos de la portabilidad del sistema el grupo de Taller 5 decidió dar soporte a ambas. Para ello se utilizó ***ODBC*** como mecanismo estándar de acceso a bases de datos. De todas maneras, se recomienda el uso de Oracle8i puesto que es la base de datos de Internet por excelencia, y es el motor detrás de los más grandes sitios de Internet: Amazon, Ebay, Etrade, Yahoo, Excite, Cisco, etc. El 70% de los mayores sitios de Internet usa Oracle. Además esta disponible en múltiples plataformas, y tiene extensiones para incrementar la seguridad (Oracle Advanced Security), los cuales son objetivos del presente proyecto.

6.1.10 Elección de herramienta para el modelado del esquema de la Base de Datos

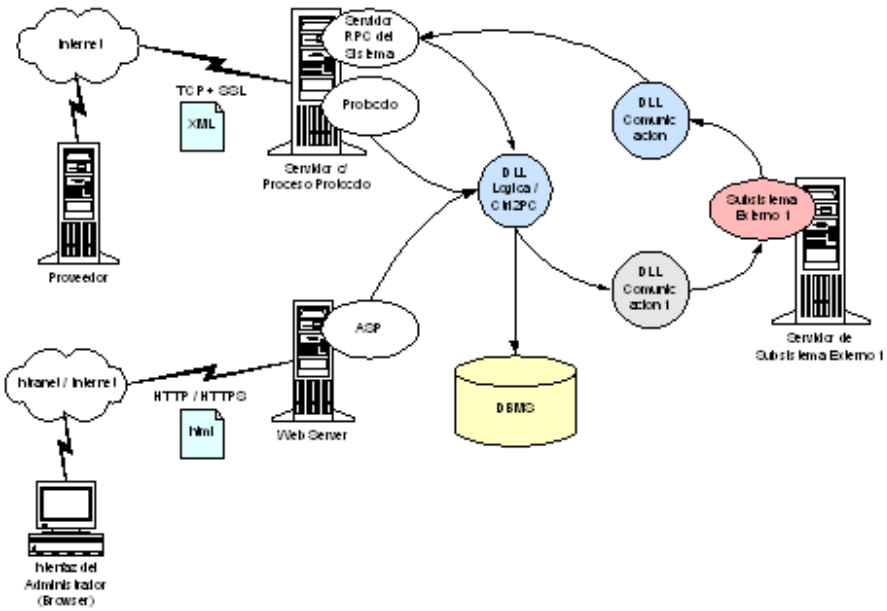
Se utilizó la herramienta Designer/2000 versión 6.0 de Oracle para la elaboración del Diagrama Entidad – Relación, con la finalidad de poder automatizar la generación del diseño físico, dar más flexibilidad a la hora de realizar cambios e impactarlos en la Base de Datos, generar reportes asociados, y poder realizar Ingeniería Inversa.

6.2 Consideraciones de Implementación

En cuanto al producto logrado, puede encontrarse una descripción detallada en el *Capítulo 8: Conclusiones, punto Producto entregado*. Aquí se describirán aquellos aspectos que se consideren relevantes referentes a la implementación de la solución.

6.2.1 Arquitectura de la solución

La siguiente figura muestra la arquitectura de la solución desde una perspectiva de la ANC y los componentes que intervienen en la misma.



La arquitectura propuesta permite la interacción con los distintos subsistemas heterogéneos de la ANC. Esto se logra mediante la utilización de un ***Módulo de comunicaciones*** (en la figura corresponde a *DLL Comunicación 1*, color gris) el cual define la interfaz con cada uno de los subsistemas, de forma tal que el sistema diseñado tiene siempre una interfaz homogénea con el resto de los subsistemas. Es responsabilidad de cada uno de los subsistemas proporcionar un módulo de comunicaciones que cumpla la interfaz especificada. De esta forma, el

sistema es independiente de la implementación de la comunicación con cada subsistema, siendo transparente al mismo.

De forma análoga, el sistema implementa un ***Módulo de comunicaciones*** (en la figura corresponde a DLL Comunicación, color celeste), el cual ofrece la interfaz al resto de los subsistemas para su interacción. En este caso la comunicación se realiza vía un ***Servidor RPC***, (en la figura corresponde a Servidor RPC del Sistema, color blanco) el cual atiende dicha solicitud y la ejecuta a través de la funcionalidad exportada por el ***Módulo principal*** del sistema (en la figura corresponde a DLL Lógica / Ctrl2PC, color celeste). Toda la comunicación entre los sistemas es realizada a través de intercambios de documentos XML.

Además del servidor RPC, tanto el ***Módulo protocolo*** (en la figura corresponde a Protocolo, color blanco) como el ***Módulo de Administración*** (en la figura corresponde a ASP, color blanco), acceden al módulo principal para ejecutar las funcionalidades brindadas por éste. De esta manera se logra centralizar la lógica de la aplicación en un único componente.

El módulo principal realiza el procesamiento relativo a sus funcionalidades exportadas, impactando los datos en su base de datos.

Para mejorar la eficiencia, por ser un componente crítico dada la cantidad de accesos que se preveen y el tiempo de respuesta esperado, el módulo protocolo esta implementado utilizando C++ y *multi-thread* (creando un thread por conexión). La comunicación con la contraparte Proveedor se realiza a través de SSL/TCP mediante la utilización de OpenSSL y sockets, intercambiando los documentos XML correspondientes.

Por otro lado, el módulo de Administración simplemente proporciona una interfaz gráfica a través de un servidor Web a las funcionalidades de administración brindadas por el módulo principal. Para esto se previó la utilización de páginas ASP y la posibilidad de configurar el Web Server para la utilización de Certificados Digitales para el acceso seguro a través de HTTPS, posibilitando así una administración vía Internet.

6.2.2 Algoritmos de encripción utilizados

Para la generación de firmas digitales es utilizado como algoritmo de hash SHA-1 y los Certificados Digitales utilizados serán los suministrados por la ANC, sin embargo en las pruebas realizadas se usaron los generados por la herramienta OpenSSL con criptografía RSA de 2048 bits.

En lo referente a la encripción en la sesión establecida por SSL, el algoritmo de clave simétrica utilizado es negociado con el cliente en función del los algoritmos soportados por ambas partes. En principio no hay restricción por parte de la ANC en cuanto a la lista de algoritmos a utilizar, esto queda pendiente hasta obtener una definición por parte de la ANC.

6.2.3 DTDs y documentos XML utilizados

Se generó un único DTD que define la estructura de todos los documentos XML intercambiados entre los Proveedores y la ANC, y es utilizado en su validación.

6.2.3.1 Criterios utilizados en la definición del DTD

Los siguientes son los principales criterios utilizados en la creación de DTDs, para conocer una lista más detallada de los mismos se debe consultar el *Anexo G: Detalles de implementación* [1]

- Mantener la claridad y simpleza del “código”.
- Si un objeto es de tipo enumerado, entonces es un atributo.
- Si un objeto contiene otros elementos o puede en un futuro contenerlos, entonces es un elemento.
- Si un objeto contiene gran cantidad de texto, entonces es un elemento.
- Reutilizar definiciones, si algún objeto estaba definido como atributo o elemento antes, entonces lo utilizo de la misma manera en futuros objetos
- Si varios atributos o varios elementos se utilizan mas de una vez juntos en distintos elementos, entonces los defino como entidades paramétricas.

6.2.3.2 Procedimiento utilizado en la creación del DTD y documentos XML

Para la creación del DTD y validar la estructura del mismo se utilizó la ya mencionada herramienta XMLAuthority, también generándose algunas partes de manera manual para sortear algunos "bugs" que posee la misma. Para la creación de los documentos XML y su validación contra el DTD, se utilizó principalmente la herramienta Xeena de IBM y también de manera manual. Esta herramienta permite verificar que la estructura que permite el DTD es la deseada, pues solo permite ir creando documentos según la estructura especificada en el mismo y permite visualizar la estructura posible. Asimismo, para la validación de los documentos y búsqueda de errores en los mismos y en el DTD, también se utilizó el parser MSXML porque especifica y ubica muy bien los errores, aunque tiene algunos problemas con documentos grandes.

6.2.4 Manejo de logs del protocolo

El módulo protocolo al correr como servicio, permite mediante la generación de logs, el seguimiento de los sucesos más importantes durante el funcionamiento del mismo. La generación de logs es diaria y configurable, así como también la cantidad de días que se mantienen dichos logs. Por más detalles consultar el *Manual de Instalación y Configuración del Protocolo* [17]

6.2.5 Consideraciones relativas a la seguridad

Además de validar la Firma Digital de los documentos recibidos, los Certificados Digitales utilizados en su validación son a su vez comparados contra los ***previamente almacenados en la Base de Datos del Sistema*** como nivel adicional de seguridad explícitamente solicitado por la ANC.
Como mecanismo de encripción de los mensajes intercambiados, se utiliza el protocolo SSL. Para una mayor seguridad y control de las transacciones, a nivel de aplicación, es posible la configuración de topes en los importes máximos permitidos tanto por proveedor como por cliente final. Adicional a lo anterior, la aplicación automáticamente registra las faltas por cobro, entrega y retiro y en caso de exceder el máximo configurable, deshabilita al cliente o proveedor en cuestión. (ver *Capítulo 5: Aspectos relativos al Análisis y Diseño de la aplicación, punto Características*)

6.2.6 Consideraciones relativas a la BD

Como aspectos interesantes de destacar de la implementación asociada a la base de datos, podemos mencionar:

- Para impactar en la base de datos las modificaciones de los objetos, se actualizan (update) solo aquellos campos que han cambiado y no todo el registro, lo cual tiene una significativa mejora en la performance.
- Se mantiene un pool de conexiones a la base de datos, con lo cual se reutilizan las conexiones en la medida de lo posible, mejorando así la performance al no tener que iniciar continuamente conexiones a la base.
- Se mantiene un registro de logs de eventos asociados a una solicitud, manteniendo así una trazabilidad sobre los cambios de la misma.

6.3 Referencias bibliográficas

[1] Anexo G: Detalles de Implementación
[2] OpenSSL es una herramienta Open Source que implementa los protocolos Secure Socket Layer (SSL v2/v3) y Transport Layer Security (TLS v1) y una librería de criptografía fuerte de propósito general. El sitio web es http://www.openssl.org
[3] SSLeay Documentation Archive http://www.columbia.edu/~ariel/ssleay/
[4] The Apache Software Foundation http://www.apache.org
[5] mod_ssl proporciona un módulo de criptografía fuerte al Apache a través de los protocolos SSL y TLS, utilizando OpenSSL. El sitio web es http://www.modssl.org
[6] Apache-SSL Servidor web seguro basado en Apache y OpenSSL. El sitio web es http://www.apache-ssl.org
[7] Apache XML Project http://xml.apache.org

[8] Xerces-C Es un parser XML con validación, escrito en un subconjunto portable de C++. El sitio web es http://xml.apache.org/xerces-c/index.html
[9] XML Authority Herramienta de generación de DTDs y XML Schemas. http://www.extensibility.com
[10] Xeena Es una herramienta para la generación de documentos XML a partir de DTDs. http://www.alphaworks.ibm.com/tech/xeena
[11] Una completa reseña del software disponible asociado a la tecnología XML se puede consultar en http://www.xmlsofware.com
[12] MSXML Es el parser XML de Microsoft y puede obtenerse en http://msdn.microsoft.com/downloads/tools/xmlparser/xmlparser.asp
[13] SDL (Specification and Description Language) http://www.sdl-forum.org/
El sitio web dispone de tutoriales de SDL, tanto para acceso en línea como para bajar.
[14] Anexo B: Algunos conocimientos previos requeridos, documento SDL
[15] UML: http://www.rational.com
Applying UML and Patterns. An Introduction to Object-Oriented Analysis and Design – Craig Larman
[16] Anexo B: Algunos conocimientos previos requeridos, documento UML
[17] Anexo I: Manuales, documento *Manual de Instalación y Configuración del Protocolo*

CAPITULO 7

7. Planificación y Administración

En este capítulo se destacan aspectos de planificación del proyecto, como ser la necesidad de adoptar una metodología de trabajo tanto para la especificación del protocolo como para el proceso de desarrollo de la aplicación, aspectos administrativos y análisis de factores críticos de éxito del proyecto, cuándo fue necesario replanificar y cuál fue la evolución de los diferentes cronogramas y planes de trabajo definidos durantes el transcurso del proyecto.

7.1 Necesidad de un proceso de desarrollo

El ***Proceso de Desarrollo de Software*** es el método de organizar las actividades relacionadas con la creación, presentación y mantenimiento de los sistemas de software. Hay varios modelos a seguir para dicho proceso, entre ellos destacamos : Modelo Codificar-Testear, Modelo en Cascada, Modelo Incremental y Modelo Iterativo.
Estos pueden ser aplicados en forma separada o combinada y la elección depende de las características del proyecto - grado en que los requerimientos del producto son documentados y entendidos, participación del cliente en el desarrollo, herramientas disponibles para el manejo del proyecto y el software, experiencia y habilidad de los miembros del proyecto, entre otras - y del producto a obtener como ser su complejidad, tamaño, nivel de calidad, temprana disponibilidad, madurez tecnológica, etc.
Una vez elegido el modelo a seguir, el próximo paso es identificar las actividades necesarias para llevarlo a cabo, cómo están relacionadas y cuál es su orden de precedencia. Bajo estas condiciones y dadas las características del presente proyecto, - Especificación de un protocolo y una Aplicación que le diese soporte –, la metodología adoptada se subdividió en dos metodologías diferentes en consecuencia de obtenerse productos de naturaleza diferente, la primera de ellas Metodología utilizada para la especificación del protocolo y la segunda Metodología de Análisis y Diseño, las mismas se describen a continuación.

7.1.1 Metodología utilizada para la especificación del protocolo

Para la especificación del protocolo utilizamos el lenguaje SDL - Specification and Description Language – el cual es un lenguaje formal y estándar de especificación que asegura principalmente consistencia y claridad en el diseño de aplicaciones criticas o complejas. Se orienta principalmente a la especificación de aplicaciones complejas orientadas a eventos, de tiempo real y que involucran interacciones entre varias actividades concurrentes utilizando señales discretas con el objeto de modelar la comunicación entre éstas. Sus principales áreas de aplicación son las telecomunicaciones, sistemas distribuidos y de tiempo real y su fuerte es la habilidad de representar estructuradamente un sistema describiendo estructura, comportamiento y datos del mismo. SDL

considera varios conceptos para describir un sistema, entre estos destacamos : ambiente, sistema, bloques, procesos, procedimientos, canales y rutas.

La metodología utilizada es la propia de SDL, la cual se caracteriza en el particionamiento de un sistema dividiéndolo en sistema, bloques y jerarquías de bloques donde los bloques pueden estar anidados unos en otros recursivamente, contener un conjunto de procesos lógicamente agrupados y donde un proceso pueda además contener uno o varios procedimientos. El objetivo de la misma es ocultar los detalles no importantes en una visión general del sistema desplazándolos a niveles mas bajos en la jerarquía de bloques de forma tal de crear módulos intelectualmente manejables, reutilizar especificaciones existentes y poder realizar subdivisiones funcionales naturales.

7.1.2 Metodología de análisis y diseño de la aplicación

Utilizamos UML - estándar plenamente adoptado para la construcción de modelos orientados a objetos independiente del proceso de desarrollo utilizado - para el modelado de la aplicación como un sistema de software orientado a objetos. Como método para organizar las actividades relacionadas con el proceso de desarrollo de software utilizamos el Desarrollo Iterativo sugerido por Craig Larman [1], el cual describimos a continuación mencionando además las diferentes actividades asociadas a cada fase del desarrollo y cómo se llevan a cabo estas últimas utilizando UML. Para ver cómo fue utilizado el mismo en el presente proyecto consultar *Capítulo 5: Aspectos relativos al análisis y diseño de la aplicación*

7.1.2.1 Descripción del Modelo iterativo

Se basa en la extensión y perfeccionamiento secuencial de un sistema a través de múltiples ciclos de desarrollo de análisis, diseño, implementación y pruebas considerando una fase de Planeación y Elaboración previa a la iteración y una fase de Aplicación posterior a la iteración.

Fase de Planeación y Elaboración.

En esta fase preliminar se generan documentos asociados a la concepción inicial del proyecto, la investigación de alternativas, la planificación, la especificación de requerimientos y otras actividades. Los documentos que se generan en esta etapa son :

- *Plan* : El cual contiene cronogramas, recursos con que se dispone y recursos necesarios, presupuesto, etc.
- *Informe preliminar de investigación* : Contiene necesidades de la empresa, motivos de la investigación, alternativas.
- *Especificación de requerimientos* : Declaración de los requerimientos.
- *Glosario* : Diccionario de términos y conceptos.
- *Prototipo* : Sistema elemental (ventanas y/o funcionalidades básicas) que facilita la comprensión de los problemas y requerimientos planteados.
- *Casos de usos* :Descripciones narrativas de los procesos de dominio.
- *Diagrama inicial de los casos de uso* : Descripción gráfica de todos los casos de uso y sus relaciones.
- *Modelo conceptual preliminar* : Su finalidad es facilitar el conocimiento del vocabulario del dominio, especialmente aquellos que intervienen en los casos de uso, su relación y la especificación de los requerimientos.

Fase de Construcción

En esta fase se producen los diferentes ciclos iterativos de análisis, diseño, implementación y prueba. Las diferentes acciones y documentos generados en estas actividades de cada ciclo de iteración son :

Análisis
El modelo de análisis sugerido por Larman, el cual se basa en UML, requiere llevar a cabo las siguientes actividades y documentos representativos de las mismas :

- Definir los casos de uso esenciales
- Perfeccionar el diagrama de casos de uso
- Perfeccionar el modelo conceptual
- Perfeccionar el glosario
- Definir los diagramas de secuencia de los sistemas
- Definir los Contratos de operaciones

Diseño
El modelo de Diseño también sugerido por Larman y basado en UML, requiere llevar a cabo las otras actividades y documentos representativos de las mismas :

- Definir los casos de uso reales
- Definir la interfaz con el usuario
- Perfeccionar la arquitectura del sistema
- Definir los diagramas de interacción
- Definir los diagramas de Clases
- Definir el esquema de la base de datos

Implementación y pruebas

Por último, estas son las últimas etapa de un ciclo de desarrollo y en general se realizan las siguientes actividades :

- Implementar las definiciones de clases
- Implementar los métodos
- Implementar Ventanas
- Implementar reportes
- Implementar esquema de base de datos

Fase de Aplicación

En esta fase se realizan las actividades necesarias para la transición de la implementación del sistema a su uso.

7.2 Administración del Proyecto

7.2.1 Utilización de Registros de Actividad

Se utilizaron registros de actividades para gestionarlas, llevar el control de las mismas y ver el cumplimiento del cronograma y el plan de trabajo propuesto.
En base a estos registros se elaboraron gráficas representativas del esfuerzo realizado a lo largo del proyecto, por etapa, por semana y la comparación con lo planificado. Dichas gráficas se encuentran en el punto *Gráficas*.

7.2.2 Elaboración de Actas de Reunión

Las reuniones con la ANC fueron debidamente registradas; por cada una de ellas generamos actas con documentos adjuntos probatorios del avance, definiciones realizadas, soluciones propuestas y acuerdos obtenidos conjuntamente.

7.2.3 Definición de Plan de Trabajo y Cronogramas

Inicialmente se definió un Plan de Trabajo Inicial para determinar la orientación del proyecto – evaluación e investigación previa a la especificación del protocolo - y poder luego definir el cronograma 1.Cuando fue necesario considerar una nueva planificación se definió un plan de trabajo asociado a la misma. Para consultar qué se realizó de cada cronograma y/o plan de trabajo, cuándo fue requerida una nueva planificación y cuál fue la evolución entre los diferentes planes de trabajo utilizados, ver el punto Evolución y Cumplimiento del Cronograma; el cual contiene además una gráfica representativa de dicha evolución.

Este punto tiene por objeto describir la evolución de los diferentes cronogramas y planes de trabajo definidos en el transcurso del proyecto, como también presentar un gráfico representativo de dicha evolución.

7.2.4 Utilización de Estándares de Documentación y Codificación

Al inicio del proyecto se definió un estándar de documentación para todos los documentos que se pudiesen generar en el mismo y previo a la implementación se generó un documento Estándar de Codificación asociado a Visual C++ conteniendo reglas de codificación como también recomendaciones y buenas practicas de desarrollo.

7.2.5 Gestión de los Cambios en los Documentos Generados

Se definió un documento adicional Gestión de Cambios, con el fin de registrar las cambios que sufran los documentos generados en el transcurso del proyecto al pasar de una versión a otra.

7.3 Análisis de factores críticos de éxito

7.3.1 Involucramiento de la ANC en decisiones y definiciones del negocio.

El grado de participación e involucramiento por parte de los entendidos del negocio contribuye a que la toma de decisiones referentes al mismo sea más o menos ágil no siendo necesario que el equipo asuma el rol de la empresa, minimizando así posibles errores por asunciones incorrectas y tiempos dedicados en la discusión de problemas y definiciones del negocio.

7.3.2 Tiempo requerido por el proyecto y su adecuación a los tiempos del taller5

Determinar a priori el tiempo que insumirá un proyecto es una tarea compleja, requiere fundamentalmente de experiencia, antecedentes y conocimiento entorno a la problemática inherente al tema planteado y la cuán accesibles son las fuentes de información – documental o Directa de Técnicos – y de investigación a ser utilizadas.
Más aún cuando el tiempo que debe insumir un proyecto debe ajustarse a un tiempo preestablecido; aunque el tiempo siempre es un elemento ineludible en todo proyecto.

7.3.3 Recursos humanos reales para emprender el proyecto

El número de integrantes de un equipo es un factor importante; pocos integrantes facilita las reuniones plenarias de trabajo y la comunicación entre estos como también la integración de las actividades realizadas por cada uno de ellos. Por el contrario, un número elevado contribuye a una menor frecuencia de las reuniones plenarias y a enlentecer la toma de decisiones, aunque pueda resultar muy rica en aportes. Es difícil determinar el número de integrantes adecuados al proyecto y en consecuencia puede llegar a influir negativamente en el mismo.

7.4 Evolución y Cumplimiento del Cronograma

El Cronograma de un proyecto es el aspecto medular y fundamental del Proceso de desarrollo de software ya que mediante este se define el camino que las actividades deben seguir; afectando a todos sus miembros y sobre el cual se centra el seguimiento y las comunicaciones a lo largo de todo el proyecto. Quizás la tarea más difícil del desarrollo del cronograma sea estimar la duración de una actividad; en particular porque muchas veces es necesario que estimar algo que nunca antes se hizo, o que se hizo no tuvo exactamente las mismas características. Desglosar cada actividad en piezas más pequeñas, utilizar datos históricos así como también incluir períodos de contingencia en cada actividad con el fin de absorber atrasos inesperados, son aspectos que contribuyen a una mejor estimación de las actividades y en consecuencia mejor elaboración del cronograma. La elaboración del cronograma e identificación de metas alcanzadas durante el desarrollo del proyecto, facilita el seguimiento de este ya que permite mantener una visión global del mismo; garantiza además que los plazos y compromisos fueron debidamente difundidos. Los plazos de un proyecto pueden cambiar a lo largo del desarrollo del mismo; no es conveniente cambiarlos si se tiene alguna chance de recuperar el tiempo, como tampoco hacerlo más de una vez pues cada cambio tiene asociado sus costos en la productividad.

Este punto tiene por objeto describir la evolución de los diferentes cronogramas y planes de trabajo definidos en el transcurso del proyecto, como también presentar un gráfico representativo de dicha evolución. Al comienzo del mismo, en las dos primeras reuniones conjuntas con la ANC - Actas 1 y 2 respectivamente [2] – se estableció un plan inicial para abordar el proyecto, en el cual su primer etapa consistía en comprender la necesidad de la ANC y el objetivo del proyecto, introducirse en el e-commerce y sus características, estudiar XML como estándar a utilizar para el intercambio de información e investigar la existencia de un protocolo o marco de trabajo aplicable a la problemática de la ANC. Se desconocía el tipo de información a la que se pretendía acceder en la investigación, su claridad y cuán compleja pudiese ser la evaluación de protocolos existentes, por lo cual no se definió un cronograma inicial y tampoco se estableció la duración de la misma ya que esta etapa era la base para determinar la orientación del proyecto y en consecuencia sus próximas etapas.

Al finalizar la antedicha etapa, se definió un primer cronograma [3], en el cual su primer etapa coincide con la etapa cumplida al momento de su realización y las siguientes son las etapas tentativas a ser consideradas para obtener el objetivo propuesto inicialmente, respetando el período estipulado de 8 meses para la duración del taller5 – inicio 8/5/2000 y fin 9/1/2001 -, siendo además revisado y ajustado conjuntamente con la ANC.
Como consecuencia de no existir un protocolo y ser necesario especificarlo en un lenguaje adecuado para tal fin, su segunda etapa comienza con la definición de las características del mencionado protocolo, búsqueda del lenguaje de especificación a utilizar, su aprendizaje y la especificación en sí. Es de destacar que al no existir una idea clara del negocio en si, y mucho menos un proceso que lo defina detalladamente, sus características fueron surgiendo progresivamente definiendo el servicio en sí mismo y no solo el protocolo, involucrando además importantes decisiones del negocio; decisiones que en su gran mayoría fueron tomadas por el grupo y comunicadas a la ANC para su evaluación y aceptación.

Al finalizar esta etapa – 04/9/00 [5] -, se entregó el documento de Arquitectura del Protocolo.doc versión 2.0, con el cual se da por finalizada la especificación del protocolo. A medida que se logró una mayor interiorización, entendimiento del negocio e ir definiéndose conceptualmente el servicio, fue surgiendo la necesidad de una aplicación que implementase el protocolo, interactuase con los subsistemas y que permitiese la administración de dicho protocolo, definiéndose así un solución global que determina toda una infraestructura asociada al servicio de Delivery.

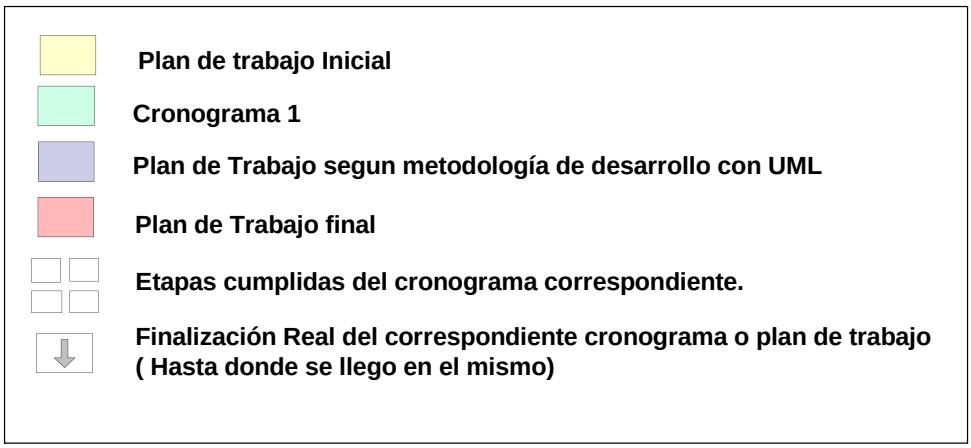
En virtud de la amplitud que fue adquiriendo la aplicación, se vio necesario incorporar el Análisis y Diseño de la misma – actividad sólo considerada para la implementación del protocolo por intermedio de la especificación del mismo - y se definió utilizar UML como herramienta para tal fin, impactando considerablemente en los tiempos y el trabajo a realizar, dando origen a una nueva planificación. En la misma, se estableció un plan de trabajo el cual incluye las dos primeras etapas ya cumplidas y agrega dos nuevas etapas, la tercera referida al estudio y aprendizaje de UML y una cuarta de Análisis y Diseño de la aplicación la cual se rige por la Metodología de desarrollo con UML detallada en el punto *Metodología de Análisis y Diseño de la aplicación*. Se postergó la definición de la fecha de finalización de esta última y en consecuencia el resto del cronograma, hasta lograr un

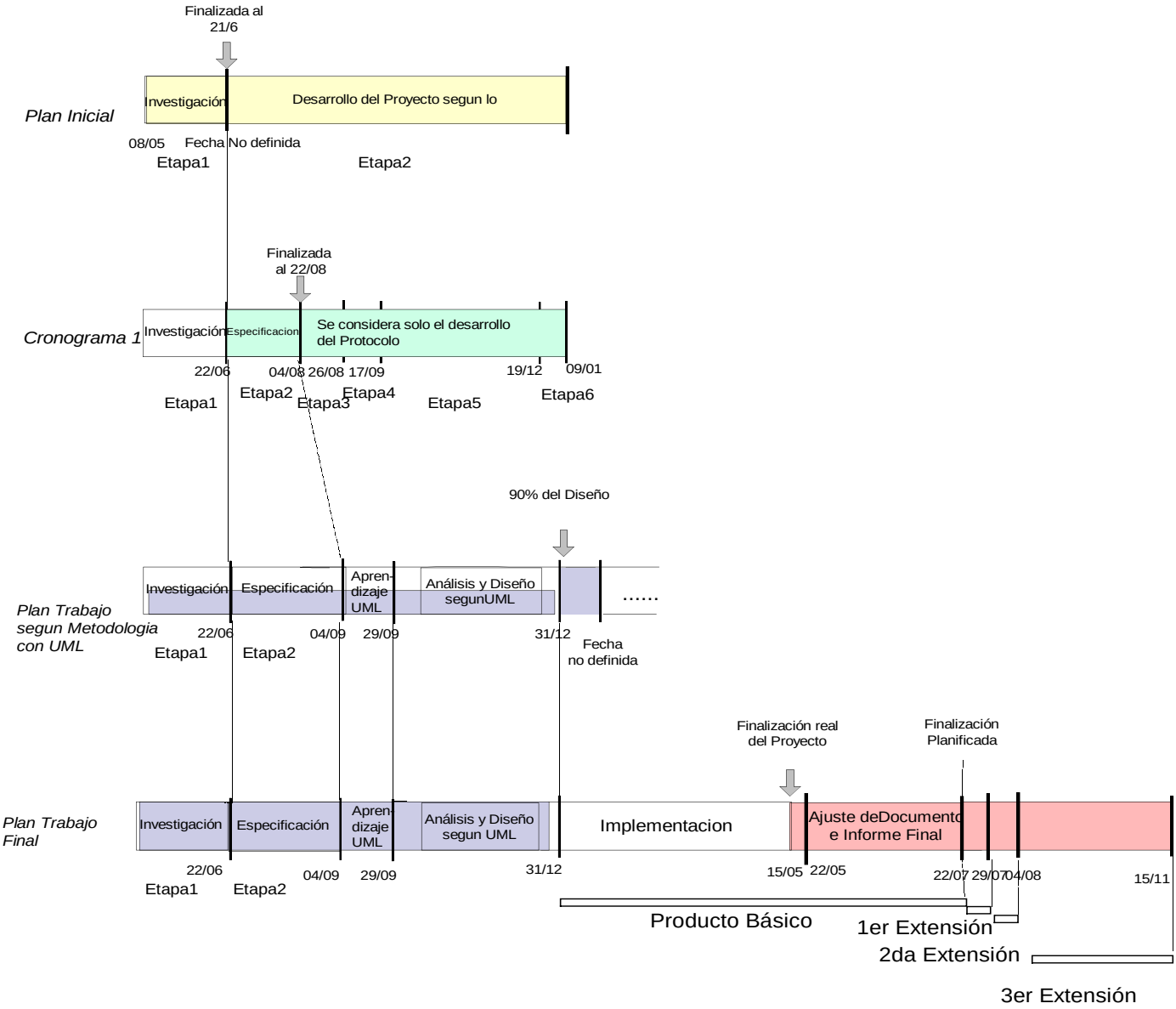
mayor avance de las etapas establecidas en la metodología utilizada, por desconocer el tiempo requerido por éstas.

A pesar de no haber definido fecha de finalización para la cuarta etapa, a medida que se fue avanzando en la misma, fueron surgiendo otros factores que ampliaron el producto a ser diseñado, como ser la creación de los módulo Controlador2PC y Recuperación de fallas [5], factores que – al requerirse el análisis y diseño de los mismos – incidieron notoriamente en la extensión del diseño, los cuales conjuntamente con el surgimiento de la aplicación - luego de finalizada la especificación del protocolo – contribuyeron a que el proyecto fuese tomando mayor amplitud a la establecida inicialmente, requiriéndose una replanificación del mismo. Por tal razón y debido también al cambio de responsables del proyecto por parte de la ANC, en diciembre del 2000 se realizó una presentación del avance del proyecto a todas las partes involucradas. Luego de ésta, se trataron algunos puntos y surgió la necesidades de elaborar un plan de trabajo detallado estimando la duración de cada actividad y prioridades de las mismas, el cual fue elaborado por el grupo y distribuido al resto de las partes involucradas a fines de enero de 2001. En el mismo se estudio la importancia de las actividades, su cohesión y los tiempos que insumía cada una, surgiendo que el mayor impacto era causado por la implementación de los diferentes contratos determinando la no factibilidad de alcanzar la implementación completa del producto totalmente diseñado en los plazos establecidos inicialmente para el presente taller. Según lo antedicho, se priorizaron los contratos asociados a las funcionalidades del protocolo, determinando de este modo el producto básico a implementarse, el cual requería que nuestro taller se extendiese hasta agosto de 2001. Aún así, se definieron además futuras extensiones a ser realizadas luego de finalizado el taller, de forma tal de acercarse gradualmente al producto totalmente diseñado.

A pesar de cumplir en un 90% con el plan establecido en dicho plan a abril de 2001 y en virtud que el plazo para la finalización de nuestro taller debía ser mediados de mayo del 2001, no pudiendo extenderse el mismo por razones externas al grupo de taller, se realizó una reunión conjunta de las partes involucradas con el objetivo de establecer cuál es el producto a entregar, qué queda por hacer y analizar la factibilidad y validez de plantear un nuevo taller V para el presente año con el fin de continuar y extender el presente taller.

A continuación presentamos la notación de la figura de integración de cronogramas y planes de trabajo que presentamos a continuación

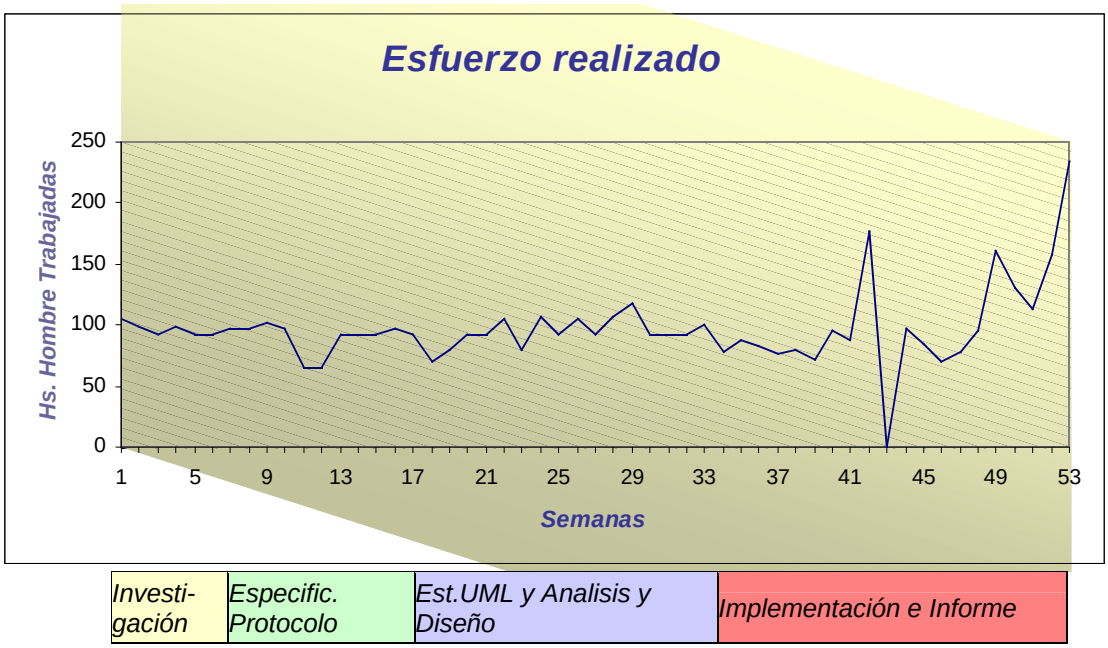




7.5 Gráficas

En este punto se presentan diferentes gráficas asociadas a la administración del proyecto, se discuten aspectos relevantes de las mismas y – si corresponde - se hace referencia al gráfico asociado al punto Evolución y Cumplimiento del Cronograma con el cometido de situar dichas gráficas en los períodos reales asociados a las diferentes etapas de evolución del proyecto. El esfuerzo total realizado en el transcurso del proyecto fue de 5146 hs con lo cual el promedio de horas diarias de trabajo por cada integrante fue de 4,6 Hs.

7.5.1 Esfuerzo realizado por Semana



Como puntos destacables de esta gráfica podemos mencionar :

- En la semana 42 (19/2/01 - 26/2/01), los integrantes de grupo se tomaron licencia en sus respectivos trabajos con el fin de invertir una mayor dedicación en la etapa de implementación, lo cual explica el "pico" producido en la misma.

- En la semana 43 (27/2/01 - 4/3/01), se realizó un descanso, por lo cual no hubo dedicación al proyecto durante la misma.

- La semana 49 (9/4/01 - 15/4/01) correspondió a Turismo invirtiéndose mayor dedicación y obteniéndose en consecuencia un "pico" asociado en la gráfica.

- Dado que la finalización del proyecto fue determinada un mes antes de su culminación, es notorio el pronunciado y continuo aumento en el esfuerzo semanal invertido a partir de la semana 49, alcanzándose un máximo absoluto en la semana previa a su finalización.

7.5.2 Esfuerzo realizado por Semana según Actividad General

La siguiente gráfica tiene como cometido la comparación de los esfuerzos invertidos en cada semana en las diferentes actividades generales realizadas a lo largo del proyecto.
En la misma destacamos :

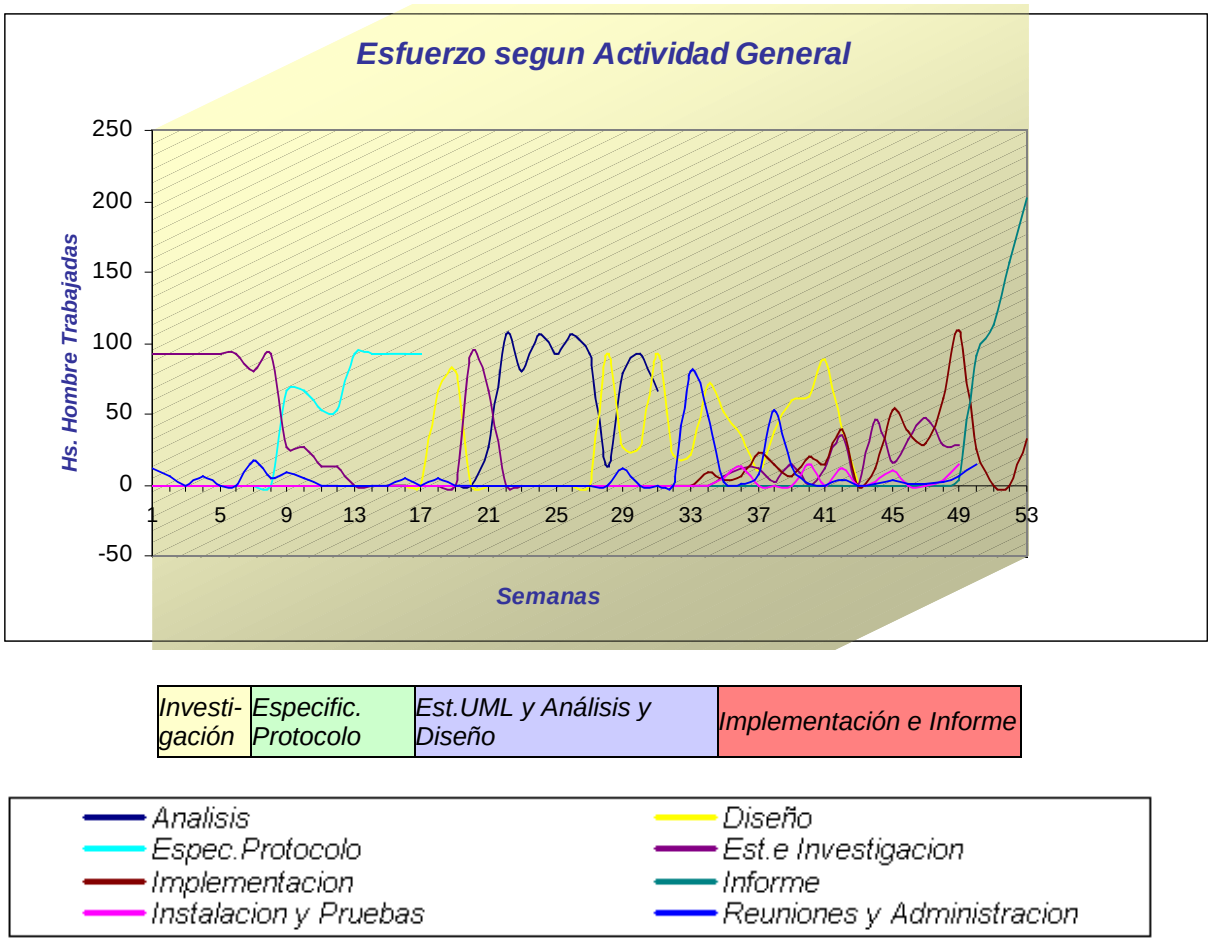
- En la actividad Reuniones y Administración se destacan los máximos asociados a las semanas 33 y 38; los cuales corresponden al período de elaboración de la presentación de Avance del proyecto a diciembre de 2000 y al período de elaboración de la propuesta detallada de plan de trabajo – la cual surgió de la mencionada reunión – respectivamente.
  El resto de los "picos" corresponden a las reuniones realizadas con ANC durante el desarrollo del proyecto y la consecuente generación de actas.
  Se destaca además el período de coexistencia entre la implementación y el diseño, el cual ascendió a tres meses.

- La actividad de Est. e Investigación tuvo una mayor dedicación en las etapas de investigación, estudio de UML e Implementación, destacándose por un lado el máximo obtenido en las dos primeras y por otro, el defasaje entre las actividades implementación y Estudio e Investigación dado que la primera se basa y es impulsada por el estudio previo asociado a la segunda

- La actividad de Especificación del protocolo tuvo un creciente aumento a medida que se fue avanzando en el estudio de SDL como lenguaje a utilizar para dicha especificación, alcanzando su punto máximo al finalizar la misma.

- Actividades Análisis, Diseño y estudio de UML .
  Previo al estudio de UML para el análisis y diseño de la aplicación, se realizó un "Bosquejo de MER" cuyo cometido fue lograr una mayor comprensión de la interrelación de las diferentes entidades involucradas en la problemática planteada. El esfuerzo invertido en el Análisis fue incrementándose a medida que se adquirió mayor conocimiento en UML hasta llegar a un punto máximo cuando el estudio fue mínimo. Destacándose además los ciclos iterativos entre las actividades Análisis y Diseño , los cuales son consecuencia directa de la aplicación del Modelo de Desarrollo Iterativo sugerido por Craig Larman en [1],

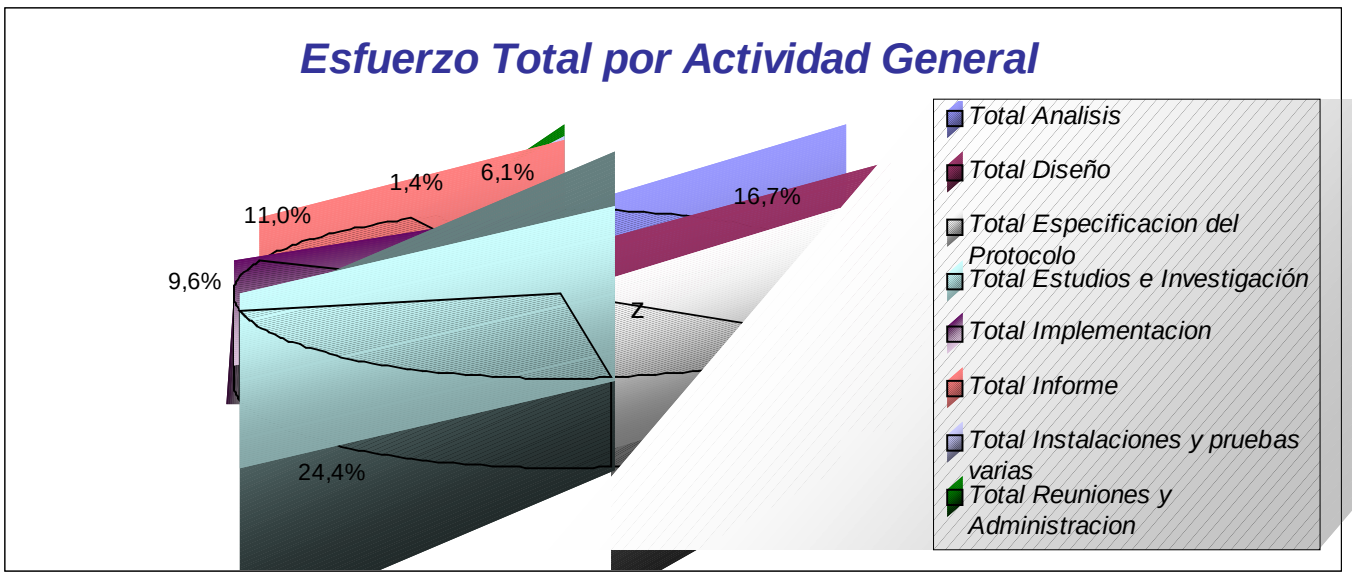
- Al igual que en la gráfica anterior se destaca el pronunciado y continuo incremento en el esfuerzo invertido en la generación del informe final.



7.5.3 Esfuerzo Total por Actividad General

El cometido de esta gráfica es visualizar el esfuerzo invertido en cada actividad general medido en horas hombre trabajadas.

7.5.4 Esfuerzo Total por Actividad General en Porcentajes



El cometido de esta gráfica es visualizar el esfuerzo invertido en cada actividad general medido en porcentaje con respecto al total invertido.

7.6 Referencias bibliográficas

[1] *UML and Patterns. An Introduction to Object-Oriented Analysis and Design* - Craig Larman
[2] Anexo H: Administración, documentos de Actas 1 y 2
[3] Anexo H: Administración, documento Cronograma
[4] Anexo H: Administración, documento de Acta 9.
[5] Anexo F: Análisis y Diseño de la Aplicación y su Recuperación, documento Transacciones con subsistemas.

CAPITULO 8

8. Conclusiones

Como modo de exponer las conclusiones, se presentará primero las *Dificultades encontradas y Desafíos*, destacando a continuación los aspectos positivos del proyecto (*Aspectos positivos a destacar*), así como los negativos (*Aspectos negativos a destacar*). Luego se expondrán las *Metas Alcanzadas* confrontándolas con los objetivos inicialmente definidos (ver *Capitulo 2: Objetivos*), destacando el producto obtenido (*Producto entregado*), sus beneficios destacables (*Beneficios de la solución*) y el impacto que se espera cause en el mercado regional (*Impacto en el Mercado*).

8.1 Dificultades encontradas y Desafíos

Las dificultades encontradas y desafíos fueron de distinta índole. Por un lado a nivel general se puede mencionar el desafío de enfrentar temas inicialmente desconocidos por los integrantes del grupo como ser las tecnologías y estándares de e-commerce o los aspectos relativos a la seguridad en Internet. El administrar y realizar un proyecto de gran envergadura, así como también la necesidad de analizar y elaborar un proceso de negocio - en particular el servicio en cuestión - fueron importantes desafíos.

Por otro lado, relacionado con el proceso de desarrollo, podemos destacar el desafío de enfrentar la realización de un sistema amplio, complejo y con partes críticas particulares. Para ello, fue necesario el aprendizaje de varias herramientas y tecnologías, como ser SDL para la especificación formal del protocolo y UML para el modelado en el análisis y diseño. El desafío de lograr las características deseables del sistema consistía en afrontar las dificultades técnicas especificas surgidas a partir de dichas características , las cuales son detalladas en el punto *Beneficios de la solución*

8.2 Aspectos positivos a destacar

- ***El Proyecto en sí mismo***

Es destacable la realización de un proyecto de esta envergadura e importancia para la ANC, así como su importancia a nivel nacional por ser la primer experiencia en el desarrollo de un protocolo de estas características.

- ***Conocimientos y Experiencia obtenida***

Es importante resaltar los conocimientos y experiencia adquirida como resultado de la realización de este trabajo:

- Capacidad de enfrentar e investigar temas desconocidos
- Capacidad de investigar en Internet, e interactuar con profesionales a nivel mundial
- e-commerce y protocolos/frameworks existentes
- Seguridad en Internet
- Criptografía y manejo de Certificados Digitales
- Administración y realización de un Proyecto de gran envergadura
- Analizar y elaborar un proceso de negocio, en particular un Servicio
- Especificación e Implementación de un protocolo
- Protocolos de comunicación
- Conocimientos prácticos en Bases de Datos
- Conocimientos prácticos referidos a implementación, tales como manejo de threads, sockets, etc
- Uso de nuevas herramientas y tecnologías (ver *Capítulo 6: Aspectos relativos a la Implementación, punto Reseña de Tecnologías y herramientas utilizadas*)

- ***Realización de un proyecto para la ANC***
  La importancia curricular de la realización de un proyecto para una de las empresas estatales pioneras en brindar sus servicios a través de Internet

- ***Apoyo al B2B en el Uruguay y ayuda a pequeñas empresas***
  Es importante la implementación del servicio de delivery en el Uruguay, pues brinda la posibilidad a proveedores, de potenciar su negocio de venta de productos *on-line*, contratando el servicio de entrega de mercadería a un tercero. De esta manera se fomenta la inserción en Internet de aquellas empresas que no poseen la infraestructura necesaria para encargarse ellos mismos de la entrega y mantenimiento de stock de productos. Posibilita también el desarrollo de servicios de intermediación *on-line*, permitiendo que una empresa ofrezca por ejemplo un servicio de reventa.

- ***Desarrollo con Costo Cero en tecnologías utilizadas.***
  Se tuvo especial cuidado en la elección de las herramientas utilizadas, considerando en dicha elección las siguientes características: multiplataforma, estándares, que sean actuales, ampliamente utilizadas, confiables y robustas, gratuitas, Open Source, y con posibilidades de desarrollo futuro (ver *Capítulo 6: Aspectos relativos a la Implementación, punto Reseña de tecnologías y herramientas utilizadas*)

- ***Amplio uso de estándares del mercado***
  La realización del proyecto involucró una amplia gama de estándares de mercado, entre ellas: XML, SSL, UML, SDL, ODBC, X.509

- ***Aprendizaje de herramientas varias***
  Utilización de una variedad de herramientas actuales, entre ellas: Visual C++, OpenSSL, Xerces (parser XML), Designer/2000, XMLAuthority, XMLPro2, Xeena(Herramientas para la generación de DTDs y documentos XML)

8.3 Aspectos negativos a destacar

- ***Poco involucramiento***
  Poco involucramiento por parte de la ANC en lo que se refiere a definiciones del negocio, debiendo el grupo de Taller 5 asumir, en ciertas decisiones, el rol de empresa.

- ***Discontinuidad en la tutoría de la ANC***
  El cambio de tutor en la ANC en una etapa avanzada del proyecto afectó negativamente al mismo, impactando de las siguientes maneras:

- Provocó un cuestionamiento de la definición inicial del proyecto, ocasionando un replanteo del mismo.
- Surgieron nuevos planteamientos, los cuales debieron ser considerados y analizados.
- Provocó un atraso en el proyecto, debido al tiempo dedicado en interiorizar a los nuevos integrantes, en las características y el avance del proyecto. Se realizaron reuniones para tal fin, se elaboró una presentación formal, se discutieron las distintas actividades a realizar y las características del producto a entregar. Como resultado y con la finalidad de aclarar y formalizar lo anterior, se elaboró un plan de trabajo describiendo las actividades al detalle, su duración y el producto a entregar incluyendo extensiones futuras.

- ***Insuficiencia de tiempo***
  El tiempo disponible no permitió la implementación total del producto básico definido, pero se dejaron las pautas claras para su realización. (ver punto Producto Entregado)

- ***Generación y mantenimiento de gran volumen de documentación***
  El problema del mantenimiento de un gran volumen de documentación proveniente principalmente de la utilización de UML debido al proceso iterativo y el no disponer de una herramienta para la gestión de dicha documentación. Rational Rose es una herramienta específica para tal fin pero solo se logró conseguir una versión de evaluación de 30 días.

- ***Jornadas extensas de trabajo durante tiempos prolongados y continuos.***
  Se realizó la máxima dedicación diaria posible de manera continua durante todo el transcurso del proyecto, resultando desgastante y estresante.

8.4 Metas Alcanzadas

La obtención de los conocimientos y experiencias propuestos como formación ofrecida inicial, fueron alcanzados satisfactoriamente y también superados como se expuso en el punto *Aspectos positivos a destacar*, quedando muy conformes al respecto.

En lo referente a la administración y realización del proyecto, cabe destacar que inicialmente no se conocía la verdadera envergadura del mismo y por lo tanto, no se fue muy meticuloso en su planeamiento y seguimiento. A medida que se avanzó en el proyecto se hizo más hincapié en estos aspectos, por ejemplo detallando más los Cronogramas y planes de trabajo elaborados, así como la generación de Registros de Actividad (como se menciona en el *Capítulo 7: Planificación y Administración*).

En cuanto al análisis y elaboración del servicio, también se llego a un gran nivel de conformidad, siendo un reto no solo técnico sino empresarial, pues debió pensarse desde la perspectiva del negocio en todo momento, prestando especial atención a quien iba orientado el servicio y tratando de que el producto elaborado (en particular el protocolo de delivery) fuese suficientemente flexible, con posibilidades de extensión futura y a la vez de adopción inmediata por parte de los proveedores.

Hay que mencionar la conformidad y satisfacción en la solución alcanzada, tanto por el cuidadoso diseño, como por las características técnicas del resultado final obtenido, en la cual se combinó la aplicación de tecnologías y conceptos técnicos existentes con la innovación para lograr su adaptación a la problemática en particular, y obtener así las características o atributos del sistema deseados (ver *Capítulo 3: Descripción General de la Solución*). En conclusión, la solución obtenida cumple los objetivos iniciales y los supera ampliamente. Se supo percibir, comprender y enfrentar los puntos críticos del sistema. Se logró la especificación del protocolo esperada, con las relevantes características comentadas en el *Capítulo 4: Aspectos relativos a la Especificación del Protocolo*, así como también se logró una aplicación robusta, en cuyo diseño fueron considerados importantes aspectos descritos en el *Capítulo 5: Aspectos relativos al Análisis y Diseño de la Aplicación*, logrando así los *atributos del sistema* especificados en el documento *Especificación de Requerimientos* [1]
En cuanto a la implementación de dicha solución, fue realizada parcialmente obteniéndose como resultado la implementación del protocolo en sí, pero no pudiendo lograr gran avance en la aplicación que le da soporte.

Todavía falta avanzar en la implementación, pero están dadas aquí las bases para continuar en ello. A continuación se detalla exactamente en que consiste el producto entregado y en particular la implementación realizada.

8.4.1 Resultado Obtenido

8.4.1.1 Producto entregado

Documentos entregados

- Especificación del protocolo de Delivery.
  - Anexo E: Especificación del Protocolo
  - Anexo G: Detalles de Implementación
- Análisis y Diseño al detalle de la aplicación, incluyendo Controlador2PC, Recuperación de fallas e Interfaz grafica .
  - Anexo D: especificación de Requerimientos
  - Anexo F: Análisis y Diseño de la Aplicación y su Recuperación
- Otra Documentación relativa al Proyecto
  - Informe Final
  - Anexo A: Documentos iniciales presentados por la ANC
  - Anexo B: Conocimientos previos requeridos
    - Protocolo 2PC
    - SDL
    - UML
  - Anexo C: Estudio de la aplicación de los protocolos de e-commerce
  - Anexo G: Detalles de la Implementación
  - Anexo I: Manuales
    - Manual de Instalación y Configuración
  - Anexo H: Administración
    - estándar de Codificación
    - Plantilla de Documentación
    - Actas de Reunión
    - Registros de Actividad
    - Cronograma y Plan de Trabajo
  - Anexo J: Referencias
    - Bibliografía y Cursos
    - Expertos Consultados
  - Anexo K: Presentación del Avance del Proyecto - Diciembre 2000

Implementación de Base de Datos

- Archivos Scripts de creación de la base de datos Oracle
- Archivos Scripts de creación del esquema de base de datos Oracle/SQLServer que genera todas las tablas, índices, restricciones (constraints), vistas y secuencias asociadas al diseño detallado. También ingresa los datos iniciales para algunas tablas básicas.
- Archivo dmp conteniendo las estructuras en Designer/2000 v.6.0 del modelado de la base de datos.

Módulos implementados

- Módulo del protocolo de Delivery
- Módulo de los contratos relativos a la firma digital de los documentos intercambiados y su validación.
- Módulo de acceso a la Base de Datos y manipulación del pool de conexiones.(Manejador de Conexiones)
- Módulo Dll de simulación de los diferentes subsistemas
- Módulo de prueba del protocolo

Otras implementaciones

- DTDs utilizados en la definición y validación de los documentos XML intercambiados por el protocolo

- Modelos de ejemplo de los documentos XML intercambiados.

Mediante los documentos de Análisis y Diseño se entrega una solución completa diseñada al detalle, que contempla las características de robustez y confiabilidad del sistema ,atributos esenciales en proyectos de esta importancia. En estos documentos se dejan las pautas claras para el resto de la implementación.

En cuanto a la implementación entregada, podemos aclarar que se cuenta con la implementación del lado de la ANC del protocolo especificado, así como también una implementación de prueba que simula al proveedor. El módulo del protocolo cumple con la especificación del mismo, pero falta la lógica del procesamiento de las solicitudes (DLL Logica/Ctrl2PC) ver punto *Arquitectura de la Solución*

8.4.1.2 Beneficios de la solución

La fácil incorporación al sistema de cada Proveedor, la confiabilidad, la eficiencia, las consideraciones relativas a la seguridad (confidencialidad, autenticación, integridad y no repudio de los mensajes intercambiados), el estar diseñado para poder ser de dominio público, independiente de la plataforma y la posibilidad de seguimiento de las transacciones son beneficios importantes de la solución alcanzada. También es destacable la tolerancia a fallas, eliminando el estado inseguro del Proveedor a través de la creación de un estado pendiente de la transacción, asumiendo así la ANC el riesgo de la falla, así como también la potencialidad dada por la generación de sugerencias y el uso de XML para los documentos intercambiados.

Además de los controles obtenidos por la interacción con los subsistemas de la ANC (por ejemplo control de dirección de entrega válida, autorización de cobro al cliente, etc) el sistema provee controles relativos a la habilitación o deshabilitación de proveedores y contactos implementando un mecanismo de registro de faltas. Los tipos de faltas considerados son las faltas por cobro y entrega no realizada a los contactos de entrega (cliente final), y falta por retiro no realizado a los contactos de retiro (normalmente Proveedores). Como control adicional y para proveer mas flexibilidad al sistema, se implementó un mecanismo de topes en los montos de las solicitudes a cobrar. Dichos topes son establecidos tanto a nivel del sistema, como a nivel de proveedor. Para más detalles puede consultarse el *Capítulo 5: Aspectos relativos al Análisis y Diseño de la aplicación, punto Características*

8.4.1.2.1 Complementos a las funcionalidades requeridas

Es de destacar que la solución obtenida se orientó a la *Mejor Calidad del Producto Final* a través de consideraciones como:

- Diseño de un Controlador Two-Phase Commit (2PC) para la realización de transacciones distribuidas que involucran los distintos subsistemas de la ANC.
- Diseño de una lógica de recuperación ante caídas del sistema
- Diseño de un Manejador de Conexiones como mecanismo eficiente de acceso a la Base de Datos
- Diseño de las funcionalidades de Administración para la completa gestión del servicio

Cabe destacar que la necesidad de estas funcionalidades fue reconocida y luego requerida por la ANC. El diseñar las funcionalidades antes descritas, por un lado incrementó el volumen de implementación total, y por otro redujo el tiempo disponible para la misma

8.4.1.2.2 Impacto en el Mercado

En virtud del inminente futuro del B2B y dado el volumen de empresas que actualmente venden sus productos a través de Internet, es de destacar la importancia del rol de delivery en el mismo, es decir la distribución de productos. Más destacable aún es para el caso de aquellas empresas locales que pretenden realizar sus ventas a través de Internet, y no poseen la infraestructura necesaria tanto para la distribución como para el mantenimiento de stock de productos.

Por lo tanto, es de esperar que el servicio a brindar por la ANC tenga gran aceptación, fomentando y permitiendo la adopción del comercio electrónico por parte de empresas en la región. (ver *Capítulo 1: Introducción, punto El marco regional, el papel de la ANC, sus oportunidades y beneficios*)

8.5 Trabajo Futuro

Resta todavía avanzar en la implementación para alcanzar por completo la solución diseñada. Es de esperar que a su vez la puesta en producción del sistema produzca una retroalimentación a la definición del servicio en sí, definiendo nuevos requerimientos y modificaciones, pues es la primer experiencia acerca del servicio definido y como es usual, tanto la empresa como los clientes (en este caso Proveedores) no tienen un conocimiento claro de lo que se espera de él.

Además de lo anteriormente mencionado, se puede citar otras extensiones sugeridas:

El ***Sistema de Distribución*** el cual es una necesidad inminente para la ANC, y aunque no esta claramente definido, es de suponer que el mismo lleve un control de las flotas de vehículos disponibles de la ANC y su capacidad, detallando todo lo relativo a la distribución de los paquetes a entregar por la ANC, inclusive direcciones o zonas no alcanzadas. El desarrollo de este sistema involucra temas como Ruteo de Vehículos y GIS (Sistemas de Información Geográfica).
Esta previsto que dicho sistema interactúe con el Sistema de Delivery para la comunicación de fallas al entregar o retirar mercadería de una dirección. También esta considerada la interacción inversa, en la cual el sistema de Distribución acepta una dirección y fecha de entrega o en su defecto genera una sugerencia.
Es una propuesta de desarrollo interesante, factible de ser considerada como proyecto de Taller V.

El ***Sistema de Stock*** también es una necesidad de la ANC que servirá de soporte al Sistema de Delivery, en particular para el subservicio de Stock, permitiendo mantener un registro de los paquetes almacenados en los locales de la ANC y que conforman el stock de los distintos proveedores que tengan contratado dicho servicio. Este sistema, además de llevar dicho registro, permitirá el seguimiento de cada paquete al asociarlo a una solicitud.

También es aconsejable la creación de un ***Datawarehouse*** para explotar toda la información registrada por el Sistema, permitiendo el análisis de dicha información para poder comprenderla y utilizarla en la toma de decisiones como instrumento para lograr una mayor eficiencia y competitividad al comprender más el negocio, el segmento de mercado al cual esta dirigido, sus características, etc.

8.6 Referencias bibliográficas

[1] Anexo D: Especificación de Requerimientos