



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

SAAS PARA ASEGURADORAS

Julio Antonio Marroquin Villatoro

Asesorado por el Ing. César Rolando Batz Saquimux

Guatemala, noviembre de 2010

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

SAAS PARA ASEGURADORAS

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

JULIO ANTONIO MARROQUIN VILLATORO

ASESORADO POR EL ING. CÉSAR ROLANDO BATZ SAQUIMUX

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, NOVIEMBRE DE 2010

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympo Paiz Recinos
VOCAL I	Inga. Glenda Patricia García Soria
VOCAL II	Inga. Alba Maritza Guererro Spínola de López
VOCAL III	Ing. Miguel Ángel Dávila Calderón
VOCAL IV	Br. Luis Pedro Ortíz de León
VOCAL V	P.A. José Alfredo Ortíz Herincx
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Murphy Olympo Paiz Recinos
EXAMINADOR	Ing. Marlon Antonio Pérez Türk
EXAMINADORA	Inga. Sonia Castañeda de De Paz
EXAMINADORA	Inga. Floriza Ávila Pesquera de Medinilla
SECRETARIA	Inga. Marcia Ivonne Véliz Vargas

HONORABLE TRIBUNAL EXAMINADOR

Cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

SAAS PARA ASEGURADORAS,

tema que me fuera asignado por la Coordinación de la Carrera de Ciencias y Sistemas en marzo de 2010.

Julio Antonio Marroquin Villatoro

Guatemala, 15 de Agosto de 2010

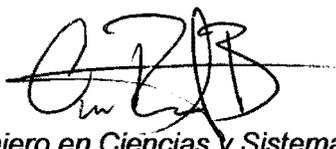
Ing. Carlos Azurdia
Seminario de Investigación
Escuela de Ciencias y Sistemas
Facultad de Ingeniería
Universidad de San Carlos de Guatemala

Estimado Ing. Azurdia

Por medio de la presente me dirijo a usted para hacer de su conocimiento que el trabajo de graduación de la Escuela de Ingeniería en Ciencias y Sistemas titulado: **"SaaS para Aseguradoras"** realizado por el estudiante JULIO ANTONIO MARROQUIN VILLATORO, carnet 9615830, ha sido considerado como finalizado de forma exitosa cumpliendo con los requisitos planteados.

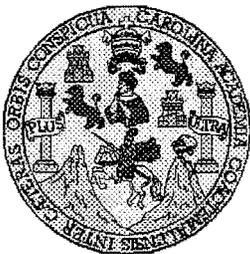
Agradeciendo de antemano su atención a la presente,

Atentamente,



Ingeniero en Ciencias y Sistemas
César Rolando Batz Saquimux
Colegiado. No. 8549

César Rolando Batz Saquimux
Ingeniero en Ciencias y Sistemas
Colegiado No. 8549



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 18 de Agosto de 2010

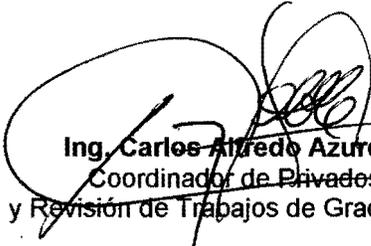
Ingeniero
Marlon Antonio Pérez Turk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación del estudiante **JULIO ANTONIO MARROQUIN VILLATORO** carné **1996-15830**, titulado: **“SAAS PARA ASEGURADORAS”**, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



E
S
C
U
E
L
A

D
E

C
I
E
N
C
I
A
S

Y

S
I
S
T
E
M
A
S

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
TEL: 24767644

El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, de trabajo de graduación titulado "SAAS PARA ASEGURADORAS", presentado por el estudiante JULIO ANTONIO MARROQUIN VILLATORO, aprueba el presente trabajo y solicita la autorización del mismo.

"ID Y ENSEÑAD A TODOS"



*Ing. Marlon Antonio Pérez Turk
Director, Escuela de Ingeniería Ciencias y Sistemas*

Guatemala, 09 de noviembre 2010



DTG. 363.2010

El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **SAAS PARA ASEGURADORAS**, presentado por el estudiante universitario **Julio Antonio Marroquin Villatoro**, autoriza la impresión del mismo.

IMPRÍMASE:

Ing. Murphy Olympo Paiz Recinos
Decano

Guatemala, 9 de noviembre de 2010.



/gdech

ACTO QUE DEDICO A:

Dios: por su amor, guía y fortaleza para terminar mi carrera.

Mi hijo Julio Andrés Marroquin Contreras, por ser mi gran amor y la razón que me hace ser mejor día a día.

Mis padres: Luvia Magdalena Villatoro Morales y Julio Edelfonso Marroquin Aldana, por su amor, por creer siempre en mí y ser mi apoyo a lo largo de mi vida.

Mis hermanos: Mónica Lucía Marroquin Villatoro y Juan Pablo Marroquin Villatoro, por su apoyo y cariño. Que este acto sea un ejemplo a seguir en su futuro para alcanzar sus metas.

Mi bisabuela: Dolores Antonia Montes Maldonado (Mamaloli) (q.e.p.d.), por su amor, consejos y apoyo.

Mis abuelos: Leonila Aldana (q.e.p.d.), Amelia Morales (q.e.p.d.), Julio Marroquin (q.e.p.d.) y Hector Villatoro (q.e.p.d.) por su cariño.

Mis padrinos: Fluvia Marroquin y Armando Umaña, por su amor y confianza.

Mi familia: que de una u otra forma me ayudaron y me apoyaron.

Agradecimientos a:

Dios:

Por permitirme cumplir con esta meta e iluminar mi camino día a día.

Mis padres:

Por su apoyo, amor y orientación incondicional.

Mis hermanos:

Por brindarme su amor y motivación para seguir adelante.

Mi bisabuela:

Por aconsejarme y apoyarme a lo largo de mi carrera.

Mi asesor:

Ingeniero César Rolando Batz Saquimux, por tomarse el tiempo para revisar mi trabajo de graduación y orientarme con las correcciones pertinentes durante el desarrollo del mismo.

Mis centros de estudios:

Desde mi educación primaria a diversificado, Colegio Liceo Montessori, Instituto Tecnológico de Computación, los cuales me permitieron prepararme para llegar a la Universidad de San Carlos de Guatemala y cumplir esta meta.

Mis amigos:

Por su apoyo y confianza.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES	V
GLOSARIO	VII
RESUMEN	XVII
OBJETIVOS	XIX
INTRODUCCIÓN	XXI
1. CONCEPTOS GENERALES	1
1.1 ¿Qué es <i>Cloud Computing</i> ?	1
1.2 ¿Qué es el modelo <i>SaaS</i> ?	4
1.3 Características del modelo <i>SaaS</i>	6
1.4 Ventajas y desventajas del modelo <i>SaaS</i>	8
1.4.1 Ventajas del modelo <i>SaaS</i>	8
1.4.2 Desventajas del modelo <i>SaaS</i>	9
1.5 Otras clases de <i>Cloud Computing</i>	11
1.5.1 <i>Infrastructure as a Service (IaaS)</i>	11
1.5.2 <i>Plataform as a Service (PaaS)</i>	12
1.6 ¿Qué es el modelo tradicional del <i>software</i> ?	15
1.7 Ventajas y desventajas del modelo tradicional del <i>software</i>	15
1.7.1 Ventajas del modelo tradicional del <i>software</i>	15
1.7.2 Desventajas del modelo tradicional del <i>software</i>	16
2. DESCRIPCIÓN DEL MODELO	17
2.1 Entorno tecnológico de proveedores del modelo <i>SaaS</i>	17
2.2 Características del modelo	18
2.3 Módulos del sistema	23
2.3.1 Módulo del ajustador de seguros	23
2.3.1.1 Registro de casos	23

2.3.1.2	Consulta de datos del asegurado y de la póliza vehicular	25
2.3.1.3	Seguimiento a casos	26
2.3.1.4	Estadísticas	28
2.3.2	Módulo administrativo	28
2.3.2.1	Pólizas	28
2.3.2.1.1	Registro y emisión de pólizas	28
2.3.2.1.2	Informe general de pólizas	30
2.3.2.1.3	Cobro de pólizas	30
2.3.2.2	Personal administrativo	31
2.3.2.3	Talleres	31
2.3.2.4	Control de siniestros	32
2.3.2.4.1	Registro de siniestros (percances o casos)	32
2.3.2.4.2	Seguimiento a casos	34
2.3.2.4.3	Ingreso de presupuestos de reparaciones	34
2.3.2.4.4	Aprobaciones de presupuestos de reparaciones de vehículos	35
2.3.2.4.5	Pago de deducible	35
2.3.2.4.6	Pago por pérdida total	35
2.3.2.5	Estadísticas	36
2.4	Diagrama entidad-relación	37
2.4.1	Diagrama entidad-relación de la aplicación	37
2.4.2	Diagrama entidad-relación de seguridad	44
2.5	Requerimientos de <i>hardware</i> , <i>software</i> y comunicaciones en los clientes del sistema	46
2.5.1	Requerimientos de <i>hardware</i>	46
2.5.2	Requerimientos de <i>software</i>	47
2.5.3	Requerimientos de comunicaciones	47
2.6	Esquema de soporte y niveles de servicio	48
3.	TECNOLOGÍA	51

3.1	Arquitectura de infraestructura	51
3.2	Arquitectura de componentes	56
3.2.1	Base de datos	56
3.2.2	Sistema operativo	58
3.2.3	Servidor <i>Web</i>	60
3.2.4	Plataforma de desarrollo	61
3.3	Arquitectura del <i>software</i>	63
3.3.1	Arquitectura de tres capas	64
3.3.2	Capa de presentación	66
3.3.3	Capa de negocio	68
3.3.4	Capa de datos	71
3.4	Seguridad	72
	CONCLUSIONES	77
	RECOMENDACIONES	79
	BIBLIOGRAFÍA	81

ÍNDICE DE ILUSTRACIONES

FIGURAS

1	Ejemplo de <i>Cloud Computing</i>	2
2	Modelo de consulta del <i>software</i>	4
3	<i>SaaS, PaaS y IaaS</i>	14
4	Diagrama de casos de uso para un ajustador de seguros	19
5	Diagrama de casos de uso para personal administrativo	22
6	Diagrama de modelo <i>SaaS</i> para aseguradoras	37
7	Diagrama entidad-relación de la aplicación	38
8	Diagrama entidad-relación de seguridad	44
9	Arquitectura de infraestructura	52
10	<i>SQL Server 2008 Cluster</i>	54
11	Arquitectura de <i>software</i> de tres capas	66

GLOSARIO

2G	La telefonía móvil 2G es la segunda generación de telefonía móvil. Es un término utilizado para describir las redes celulares digitales como la <i>GSM</i> que sustituyeron a las redes celulares analógicas.
3G	Es la abreviatura de tercera generación de la telefonía móvil. Tiene la posibilidad de transmitir tanto voz como datos (una llamada telefónica o una videollamada), y datos no-voz (como la descarga de programas, <i>email</i> , mensajería instantánea).
Agente ajustador	Funcionario de la aseguradora o designado por ella, que tiene a su cargo la investigación del evento ocurrido, el análisis del caso y el ajuste de los reclamos.
Agente de seguros	Funcionario de la aseguradora o designado por ella, cuya actividad comercial consiste en intervenir en la contratación de seguros mediante el intercambio de propuestas, así como en el asesoramiento, según la mejor conveniencia de los contratantes.

Aseguradora	Entidad que presta el servicio de aseguramiento y que asume la obligación del pago de la indemnización cuando se produzca algún siniestro sobre lo asegurado.
Base de datos	Es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.
<i>Bits</i>	Es el acrónimo de <i>Binary digit</i> (dígito binario). El <i>bit</i> es la unidad mínima de información empleada en informática.
<i>Centro de datos</i>	Es aquella ubicación donde se concentran todos los recursos necesarios para el procesamiento de la información de una organización. Suelen ser creados y mantenidos por empresas con el objeto de tener información necesaria para sus operaciones.
<i>Cluster</i>	Es un conjunto de computadoras construidas mediante la utilización de componentes de <i>hardware</i> comunes, y que se comportan como si fueran una única computadora, más potente que las comunes de escritorio. Son usualmente empleados para mejorar el rendimiento y/o disponibilidad por encima de la que es provista por un solo computador,

siendo más económico que computadores individuales de rapidez y disponibilidad comparables.

Código fuente Es un conjunto de líneas de texto escritas en un lenguaje de programación específico, y que contienen las instrucciones que debe seguir la computadora para ejecutar un programa. En el código fuente de un programa está descrito por completo su funcionamiento.

DMZ (Desmilitarized Zone) Zona desmilitarizada o red perimetral, es una red local que se ubica entre la red interna de una organización y una red externa, generalmente *Internet*. Su objetivo es que las conexiones desde la red interna y la externa a la *DMZ* estén permitidas, mientras que las conexiones desde la *DMZ* sólo se permitan a la red externa.

Escalabilidad Es la capacidad de un sistema informático de cambiar su tamaño o configuración para adaptarse a las circunstancias cambiantes. Como característica adicional es una propiedad deseable de un sistema para que esté preparado a hacerse más grande sin perder calidad en los servicios ofrecidos.

Firewall

Corta fuegos por su equivalente en español, es una parte de un sistema o de una red que está diseñado para bloquear el acceso no autorizado y permitiendo a la vez comunicaciones autorizadas. Pueden ser implementados en *hardware* o *software* o ambos. Se utilizan con frecuencia para evitar que los usuarios de *Internet* no autorizados tengan acceso a redes privadas conectadas a la *Internet*, especialmente *Intranets*.

Framework

Desde el punto de vista del desarrollo del *software*, un *Framework* es una estructura de soporte definida, en la cual otro proyecto de *software* puede ser organizado y desarrollado. El *Framework* suele incluir: soporte de programas, bibliotecas, lenguajes de *scripting* y *software* para desarrollar y unir diferentes componentes de un proyecto de desarrollo de programas.

GPRS

General Packet Radio Services o servicio general de paquetes vía radio es una extensión del Sistema Global para Comunicaciones Móviles (*Global System for Mobile Communications* o *GSM*) para la transmisión de datos no conmutada (o por paquetes). Permite una conexión de alta velocidad y capacidad de datos, la cual está

disponible para navegar en páginas en cualquier momento.

GSM

Global System for Mobile Communications o Sistema Global para las Comunicaciones Móviles, el cual es un sistema estándar, completamente definido para la comunicación por medio de teléfonos móviles que incorporan tecnología digital.

HTTP

Hypertext Transfer Protocol, en español “Protocolo de Transferencia de Hipertexto”, el cual es un protocolo de comunicaciones que se emplea para mover y visualizar documentos en hipertexto en la *WEB*. Es el protocolo usado en cada transacción de la *WEB*.

HTTPS

Hypertext Transfer Protocol Secure, en español “Protocolo Seguro de Transferencia de Hipertexto”, es un protocolo de red basado en el protocolo *HTTP*, destinado a la transferencia segura de datos de hipertexto, es decir, la versión segura del *HTTPS*.

IP

Internet Protocol (Protocolo de *Internet*), utilizado para que los equipos, regularmente computadores, se comuniquen a través de *Internet*. Este protocolo utiliza direcciones

numéricas denominadas direcciones *IP*.

IT Tecnología de la información o *Information Technology* por sus siglas en inglés (*IT*), es un término que toma a los sistemas informáticos, como un medio para potenciar la capacidad de uso de la información en una organización. Contempla el estudio, diseño, desarrollo, implementación, soporte o dirección de los sistemas de información computarizados en particular de *software* y *hardware* de computadoras.

Mainframe Es una computadora central, la cual tiene características de ser grande, potente y costosa, regularmente utilizada por grandes compañías para el procesamiento de una gran cantidad de datos, por ejemplo: procesar transacciones bancarias.

MB Un *Megabyte* es una medida de cantidad de datos informáticos que equivale a 1,024 kilobytes.

Mbps Un *Megabits* por segundo equivale a un millón de *bits* transmitidos por segundo. Se utiliza comúnmente para medir la velocidad de una conexión como la *Internet*.

Metalinguaje	Es un lenguaje que se usa para hablar acerca de otro lenguaje.
<i>Mhz</i>	Un <i>Megahercio</i> se utiliza frecuentemente como unidad de medida de la frecuencia de trabajo de un dispositivo de <i>hardware</i> y equivale a un millón de hercios.
Póliza de seguro	Es el documento o contrato por medio del cual la empresa aseguradora se obliga mediante el cobro de una prima a resarcir un daño o a pagar una suma de dinero al verificarse la eventualidad prevista en el contrato.
Proxy	Es un programa o dispositivo que realiza una tarea de acceso a <i>Internet</i> en lugar de otro ordenador. El proxy es el punto intermedio entre un ordenador conectado a <i>Internet</i> y el servidor que está accediendo. Con un proxy controlará el acceso y navegación al <i>Internet</i> por los usuarios.
<i>RAM</i>	<i>Random Access Memory</i> , por sus siglas en inglés, es la memoria de acceso aleatorio desde donde el procesador recibe las instrucciones y guarda los resultados. Es el área de trabajo para la mayor parte del <i>software</i> de un computador.

<i>Router</i>	Es un dispositivo para la interconexión de redes informáticas que permite asegurar el enrutamiento de paquetes entre redes o determinar la ruta que debe tomar el paquete de datos.
<i>SGML</i>	Son las siglas de <i>Standard Generalized Markup Language</i> (Lenguaje de Marcado Generalizado), el cual consiste en un sistema para la organización y etiquetado de documentos. Sirve para especificar las reglas de etiquetado de documentos y no impone en sí ningún conjunto de etiquetas en especial.
Sistema operativo	Es un <i>software</i> que sirve de interfaz entre los dispositivos de <i>hardware</i> y los programas usados por el usuario para poder manejar un computador. Es el responsable de gestionar, coordinar las actividades y llevar a cabo un intercambio de los recursos y actúa como estación para las aplicaciones que se ejecutan en la máquina. Ejemplos de sistemas operativos: <i>Windows, Linux, Mac</i> .
<i>Software</i>	Es el conjunto intangible de datos y programas de la computadora, es decir, el soporte lógico e inmaterial que permite que la computadora pueda desempeñar tareas inteligentes, dirigiendo a los componentes físicos o

hardware con instrucciones y datos por medio de diferentes tipos de programas.

WAP

Wireless Application Protocol o Protocolo de aplicaciones inalámbricas, es un estándar abierto internacional para aplicaciones que utilizan las comunicaciones inalámbricas, por ejemplo: acceso al servicio de *Internet* desde un teléfono móvil.

WEB

Nombre coloquial con que se nombra a *World Wide Web*, es un sistema de comunicación y publicación que fue diseñado para distribuir información a través de redes de computadoras en una modalidad llamada hipertexto.

Web Server

Es un dispositivo informático o programa permanentemente conectado a la red, en el cual se almacenan las distintas páginas que forman un sitio *Web* disponibles para ser visitadas por los usuarios. Está diseñado para transferir hipertextos, páginas *web* o páginas *HTML (HyperText Markup Language)*: textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música.

XML

Extensible Markup Language (lenguaje de marcas extensible) es un metalenguaje extensible de etiquetas desarrollado por la *World Wide Web Consortium*, el cual es una simplificación y adaptación del *SGML*, y permite definir la gramática de lenguajes específicos. *XML* no es un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades.

RESUMEN

Con el incremento del uso de *Internet*, los avances tecnológicos y la necesidad de las empresas de contar con sistemas que les permitan automatizar sus tareas y poder acceder a la información en cualquier momento y en cualquier lugar, se ha abierto la puerta para que muchas empresas desarrolladoras de *software* empiecen a brindar a sus clientes una solución completa que les permita maximizar sus recursos y aumentar su productividad.

En el primer capítulo se darán los conceptos generales del modelo *SaaS* (*Service as a Software*) o *Software* como servicio, sus ventajas y desventajas comparado con el modelo tradicional de *software*. Se explicará ampliamente el concepto de *Cloud Computing* y que otros tipos de *Cloud Computing* existen.

En el capítulo dos se definirá un modelo *SaaS* para una aseguradora. Los puntos a definir son:

- Características del modelo.
- Opciones del sistema.
- Diagrama entidad-relación de la base de datos.
- Requerimientos de *hardware*, *software* y comunicación para clientes.
- Esquema y nivel de soporte

En el último capítulo se explicará todos los requerimientos técnicos y tecnológicos requeridos para la implementación de un sistema de este tipo, los cuales son:

- Infraestructura física a utilizar y la forma en que se implementará.
- Tipo de componentes a utilizar: Base de datos, Servidor *Web*, Sistema operativo.
- Tipo de arquitectura de *software* a utilizar.
- Medidas de seguridad a implementar para que esete servicio e información estén siempre protegidos.

OBJETIVOS

General

Proponer una solución de automatización para Aseguradoras basado en el modelo *SaaS*, que permita automatizar las funciones que realiza un agente ajustador de seguros en el momento de ocurrir un percance y las tareas posteriores al mismo.

Específicos

1. Conocer los aspectos generales del modelo *SaaS* identificando sus ventajas y desventajas.
2. Definir un modelo *SaaS* orientado a un área específica.
3. Conocer todos los aspectos tecnológicos necesarios para desarrollar e implementar un modelo *SaaS*.

INTRODUCCIÓN

En un mundo donde la tecnología avanza a gran escala, muchas empresas de desarrollo de *software* han buscado nuevas alternativas de brindar servicios a sus clientes.

Un modelo bastante utilizado últimamente es el denominado SaaS (del inglés *Software as a Service*), o *Software* como servicio, en el cual las compañías que se dedican a este tipo de servicio proveen el *software*, el mantenimiento, la infraestructura, la operación diaria y todo el soporte para el funcionamiento del sistema. Toda la información, procesos, insumos, *software*, estarán hospedados en la compañía de *IT*.

Existen muchas áreas en las que se puede aplicar el modelo SaaS. El presente trabajo de investigación se enfocará en un modelo SaaS para Aseguradoras, en el cual en el momento de ocurrir un accidente automovilístico el agente del seguro o ajustador de seguros, contará con un dispositivo móvil (*PPC* o *Smartphone*) en el que tendrá instalada una aplicación que permitirá registrar los datos generales y específicos del percance, así como la toma de fotografías del automotor o automotores implicados en el mismo. Luego que se registre la información, ésta será transmitida vía *Internet* a un servidor central en el que se generarán los informes respectivos a quien corresponda, para acelerar de esta forma los trámites a realizar tanto por la aseguradora como por el asegurado.

Se llevará un seguimiento adecuado de todos los casos, por ejemplo:

- Aprobación de presupuestos en el taller de reparación.
- Seguimiento de los vehículos que estén en reparación, indicando la etapa o estatus en que se encuentra el vehículo, mostrando fotografías en los avances de la reparación.

En la aseguradora se contará con módulos que permitan llevar la administración de cada caso, desde el momento en que se recibe el informe del ajustador de seguros hasta que el caso se da por cerrado.

Todas las consultas y registro de la información se realizarán a través de *Internet*, lo que permitirá poder acceder al sistema y a la información en cualquier momento y desde cualquier lugar.

Con un *software* de este tipo, las metodologías utilizadas por los ajustadores de seguros se aplicarán de una forma más adecuada, elevando así el rendimiento de los mismos. SaaS para Aseguradoras, automatiza el registro de los datos en el lugar del percance, agiliza la aprobación de presupuestos para reparación de vehículos, da un mejor seguimiento de casos, además de ofrecer a todos una amplia perspectiva de los casos y de las partes involucradas en cada uno de ellos, en cualquier momento y desde cualquier lugar.

1 CONCEPTOS GENERALES

1.1 ¿Qué es *Cloud Computing*?

El *Cloud Computing* o Computación en la nube, es un concepto que se puede definir como una tecnología que ofrece servicios a través de una plataforma de *Internet*, de tal forma que todo lo que puede ofrecer un sistema informático se ofrece como servicio, al cual los usuarios acceden sin que tengan conocimientos sobre la gestión de los diversos recursos que utilizan. Los programas están en los servidores en línea y se pueden acceder a los servicios a través de *Internet*.

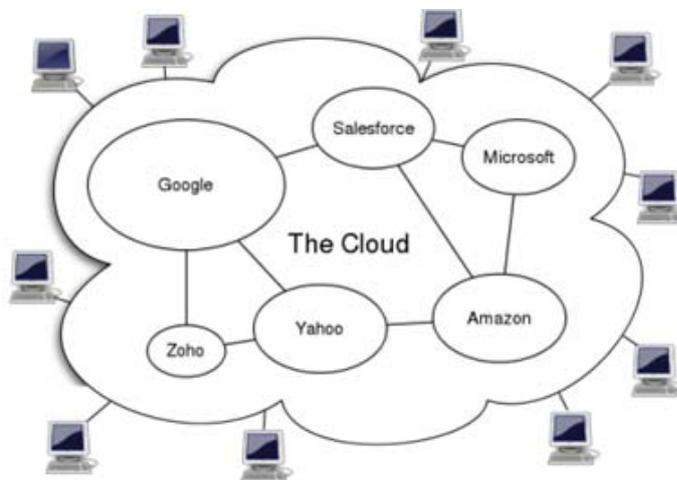
Lo que se conoce hoy en día como *Cloud Computing* ha tenido una evolución desde hace varios años:

- En la década de los sesenta, existían los *Mainframes*, los cuales eran computadoras centrales, grandes, muy costosas y pesadas, usadas principalmente por una gran compañía para el procesamiento de una gran cantidad de datos.
- En la década de los ochenta, se inició la fabricación de computadoras personales a gran escala, empezando con esto la descentralización, ya que anteriormente con los *Mainframe* se tenía todo centralizado.
- En la década de los noventa, se inició con el esquema de Cliente-servidor, en el cual algunas funciones están del lado del cliente y otras del lado del servidor; en este esquema los equipos clientes, es decir, los equipos que forman parte de la red, envían una solicitud al servidor mediante su dirección IP y el puerto que está reservado para un servicio

en particular que se ejecuta en el servidor, el cual recibe la solicitud y responde con la dirección IP de equipo cliente y su respectivo puerto.

- En los años dos mil, llega la era de *Internet*, el cual tiene un esquema similar a Cliente-servidor, pero otorga mayor inteligencia al servidor.
- Alrededor del año 2005, gracias a nuevas tecnologías e infraestructuras, el concepto de computación en la nube se empezó a nombrar a gran escala por proveedores de *Internet*, como *Google*, *Amazon AWS* y otros que construyeron su propia infraestructura de la cual surge una nueva arquitectura. Otras empresas que ofrecen estos servicios son: *Azure* de *Microsoft*, *Rackspace*, entre otras (Figura 1).

Figura 1. Ejemplo de *Cloud Computing*



Una de las principales diferencias del *Cloud Computing* consiste en que no es necesario conocer la infraestructura y funcionamiento que está detrás, ya que todo esto pasa a ser parte de la nube en la cual las aplicaciones y servicios pueden crecer, funcionar rápido y casi nunca fallan.

Entre las ventajas del *Cloud Computing* se tienen:

- Acceso a la información y a los servicios desde cualquier lugar.
- Es escalable, por ejemplo, una empresa pequeña con un costo accesible puede comprar un pequeño grupo de licencias y una multinacional comprará un grupo mayor de licencias. Ambas podrán compartir los mismos servicios.
- Toda la capacidad de procesamiento y respaldo de la información, estará a cargo de la empresa proveedora del servicio; con esto, el cliente no tendrá que tener equipos instalados en forma local para que se ocupen de dichas tareas.
- Ahorro en *hardware*, ya que no se tendrá que preocupar por la compra de equipo ni en cambios por equipos obsoletos, ya que para esto se encargará la empresa proveedora del servicio.
- Algunos servicios son gratuitos y otros con costo, según las necesidades del usuario.
- Implementación rápida de *software* el cual ya ha sido probado por varios usuarios.

Entre las desventajas tenemos:

- La privacidad de la información, ya que todos estos datos están en manos de terceras personas.
- Uno depende de que el proveedor de servicios tenga una buena política de seguridad y de la realización de copias de seguridad en forma periódica.
- Dependencia de una conexión a *Internet*.

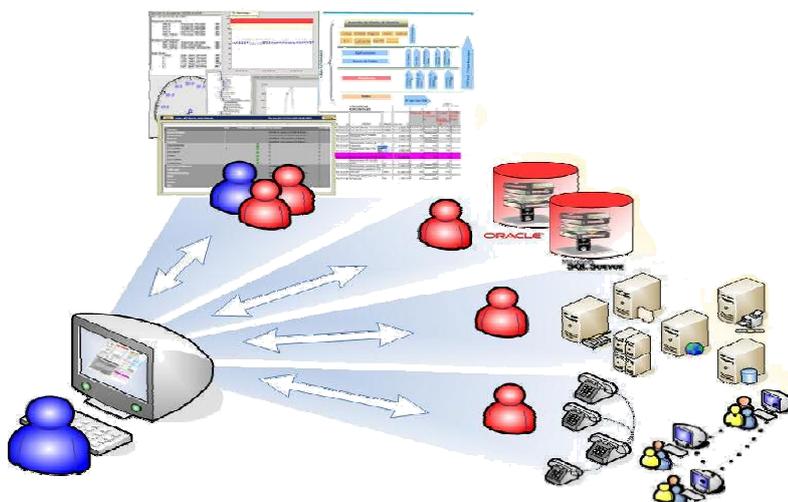
Se observará que son más las ventajas que las desventajas del *Cloud Computing*.

Una de las clases fundamentales a conocer de *Cloud Computing* es el *Software as a Service (SaaS)*.

1.2 ¿Qué es el modelo SaaS?

Software as a Service (SaaS) por sus siglas en inglés, es un modelo de distribución de *software* a través de la red, generalmente utilizando *Internet*, en el cual las aplicaciones residen en los servidores de la compañía que provee el servicio, la cual se encarga del mantenimiento del *hardware* y el *software*, operación diaria, soporte a clientes, etc. El *software* puede ser consultado en cualquier momento y en cualquier lugar a través de una conexión a *Internet* (Figura 2).

Figura 2. Modelo de consulta del *software*



Las empresas clientes de este tipo de servicio no tienen que preocuparse por mantener el *software*, realizar mejoras o actualizaciones ni dar el soporte a los usuarios, ya que todo esto está incluido en el costo por el servicio. Esto es un total cambio de enfoque, ya que habitualmente las empresas compraban o desarrollaban un *software* y se encargaban de su implementación, mantenimiento y soporte. Ahora con una cuota fija, regularmente mensual, toda empresa puede contar con este tipo de herramientas que anteriormente sólo podían ser costeadas por empresas grandes debido a su alto costo de desarrollo, implementación y mantenimiento. Con el modelo SaaS, las empresas mejorarán su competitividad, reduciendo los costos operativos y permitiendo realizar una gestión de una forma más adecuada.

SaaS no sólo es un modelo de tecnología sino también un modelo de negocios en el cual, con un *software* arrendado a un proveedor de servicios de tecnología, permitirá controlar los procesos del negocio como lo es la gestión de clientes, gestión de proyectos, puntos de venta, hasta la totalidad de todos los procesos de la organización. Aunque la infraestructura tecnológica es muy importante, los cambios frecuentes que se dan hoy en día hacen que se necesite más frecuentemente el cambio a un modelo SaaS que implicarán cambios en la gestión del negocio. Con el pasar de los tiempos y las nuevas necesidades de los clientes, SaaS se convertirá en un modelo de negocios maduro en el que las funciones básicas darán paso a funciones más avanzadas.

El modelo SaaS permite a los clientes eliminar altas sumas en compras de licencias, elimina los costos y riesgos de instalación, dar soporte y mantenimiento de *hardware* de la empresa y de mantener personal necesario para la operación adecuada del sistema.

Este modelo también da la apertura a muchos mercados para las compañías desarrolladoras de *software* al ofrecer soluciones *SaaS* a pequeñas y medianas empresas, ya que muchas de ellas no pueden invertir grandes sumas iniciales, pero sí pueden pagar una cuota fija mensual.

Este tipo de modelo, está teniendo mucha aceptación en los últimos tiempos, debido a:

- La mayoría de aplicaciones están diseñadas y adecuadas para el uso de *Internet*.
- Reducción de costos.
- Actualmente se utilizan mayores estándares de comunicación, programación y servicios de proveedores.
- El cliente se concentra más en el negocio y deja del lado del proveedor la administración y mantenimiento del *software* y el *hardware*, la seguridad de los sistemas y el soporte a los usuarios.

En resumen, la tendencia es que el modelo *SaaS* sea un estándar para las empresas que deseen adquirir o contratar un *software* por los bajos costos de adquisición, gestión y mantenimiento, siendo las empresas pequeñas y medianas las más potenciales para tener este tipo de servicio, ya que para una empresa con más de 1,000 usuarios sería conveniente hacer una evaluación de costo-beneficio.

1.3 Características del modelo *SaaS*

Las características principales del modelo *SaaS* son:

- Administración centralizada de los datos y aplicaciones permitiendo ser accedidos en forma remota.
- Regularmente este tipo de *software* se accede a través de *Internet*, permitiendo el uso de la aplicación desde diferentes ubicaciones, con lo cual se le permite al usuario una mayor flexibilidad para hacer uso de la herramienta.
- Actualización centralizada, con lo cual se evita que cada usuario descargue parches o tengan que realizar actualizaciones uno a uno. La actualización se ejecuta una sola vez en el servidor y aplica para todos los usuarios que acceden a la aplicación.
- No se paga por licencias, se paga por el uso del *software* ya sea por cantidad de usuarios y en algunos casos por las funciones implementadas en el sistema. Muchas empresas usan el esquema de fijar los precios por rangos de usuarios, es decir, aquellas que tienen más usuarios pagarán una cuota fija mensual menor por usuario que otras que tengan una menor cantidad de usuarios.
- La distribución de la aplicación tiene el modelo de uno a muchos, es decir, se desarrolla un producto y el mismo es utilizado en varios clientes.
- Flexibilidad y escalabilidad. Si el negocio crece, se contratan más licencias de uso, permitiendo una gran flexibilidad a los clientes, ya que podrían ampliar o reducir usuarios sin ningún problema. Esto permite al cliente adaptarse a las necesidades del presupuesto o del negocio en cualquier momento. La ventaja de que las licencias sean en alquiler y no en propiedad reduce significativamente los costos por el uso de la aplicación.
- Parametrizable, lo cual permite a cada usuario o tipo de usuario ver y usar lo que desee o necesite.

1.4 Ventajas y desventajas del modelo SaaS

1.4.1 Ventajas del modelo SaaS

- Costos más bajos. El cliente no necesita tener la infraestructura de *hardware* para implementar el *software* que soportará el sistema, por lo que bajará los costos y el riesgo de la inversión. Ya no se necesita tener servidores de bases de datos ni servidores de aplicaciones, ya que todo este tipo de *hardware* lo proporcionará el proveedor del servicio.
- Se paga una cuota fija mensual, la cual represente un costo menor que hacer todo el gasto de desarrollar un *software*. La inversión de *hardware* y *software* se refleja directamente en los balances financieros de la empresa, mientras que hacer un pago por un servicio se convierte en un gasto mensual menor.
- Las actualizaciones a nuevas versiones son transparentes para el cliente, ya que dicha función la realiza el proveedor de servicios quien optimiza costos al hacerlo para todos los clientes a la vez. En muchos casos, realizar actualizaciones de versiones significa altos costos de presupuestos y un tiempo considerable en el cual el departamento de IT tiene que dedicar recursos o pagar a una empresa de tecnología para realizar dicha tarea.
- Simplifica la gestión del *software* dentro de la empresa.
- La compañía de tecnología que ofrece el servicio dará todo el soporte al cliente aún después de la puesta a andar del *software*.
- La responsabilidad de la operación recae en la empresa de tecnología, quien se tiene que asegurar que el *software* esté disponible en cualquier momento. Con esto se evitará tener personal que administre el buen funcionamiento de los servidores y del sistema.

- El cliente únicamente requiere de una conexión a *Internet* para poder acceder a la aplicación con lo cual facilita su uso al poder acceder desde cualquier ubicación.
- Muchas aplicaciones también funcionan fuera de línea en computadoras y dispositivos móviles, los cuales tienen una aplicación instalada con la cual pueden realizar el registro de las actividades que luego con una sincronización a través de *Internet*, transmiten la información a un servidor central.
- Se le permite al cliente completa flexibilidad en el uso del sistema.
- Se pueden hacer actualizaciones o incluir mejoras a todos los clientes en poco tiempo, evitando estar comprando licencias frecuentemente. Con esto el cliente se asegurará de contar siempre con la última versión del *software*.
- No existe necesidad de estar renovando los equipos de computación constantemente, debido a que la mayor parte del procesamiento se realiza del lado del proveedor del servicio quien es el encargado de mantener un adecuado funcionamiento del sistema.

1.4.2 Desventajas del modelo SaaS

- El cliente no tiene acceso directo a los contenidos, ya que están almacenados en algún lugar remoto y si no se cuenta con mecanismos de cifrado y de control, disminuye la privacidad de los mismos, ya que la compañía de informática proveedora del servicio, podría consultarlos.
- Si el servicio de comunicaciones y/o *Internet* no está disponible, el usuario no podrá acceder al sistema y todas las funciones del mismo no estarán disponibles hasta que el servicio esté restablecido. Algo importante, es que el servicio de comunicaciones y/o *Internet* tiene que ser de buena calidad, ya que existen aplicaciones muy pesadas que

requieren de una buena conexión a dicho servicio, para que el funcionamiento del programa sea el adecuado.

- Integración con el resto de las aplicaciones propias del cliente. Las aplicaciones del modelo SaaS, poseen características estándares para ser utilizadas en varios clientes y muy pocas veces se pueden adaptar a las aplicaciones propias del cliente, debido a: estructura de base de datos, lenguaje de programación, procesos internos y características propias de las aplicaciones para lo que fueron desarrolladas.
- El cliente no tiene acceso directo al código fuente, por lo que modificaciones o nuevos desarrollos en la mayoría de los casos tendrán un costo adicional y una planificación para la entrega.
- La seguridad de la información es fundamental cuando una empresa tiene sus datos almacenados fuera de sus instalaciones. Esta es una de las principales desventajas ya que la información quedará almacenada con un tercero, en este caso el proveedor de servicios, quien tiene que garantizar la integridad y confidencialidad de la información. Para minimizar este riesgo, las empresas proveedoras de este tipo de servicio, tienen que:
 - Tener una política de seguridad adecuada a las necesidades del negocio.
 - Realizar periódicamente copias de seguridad de la información con respaldos en un centro de datos externo.
 - Firmar contratos de confidencialidad entre ambas partes. Un contrato de confidencialidad es un contrato cuyo objetivo principal es el de preservar el secreto o la confidencialidad de la información y/o datos del negocio de una empresa, cuando, por los motivos que sean, un tercero pueda acceder a los mismos. En el contrato se estipula que la información estará bien protegida, que podrá ser accedida únicamente por el cliente y que el

proveedor de servicios, no podrá hacer ningún uso de la información sin la debida autorización del cliente.

- Las aplicaciones con el modelo *SaaS*, deben tener forma de poder extraer toda la información y relaciones almacenadas en las bases de datos si en algún momento se decide cancelar el servicio. Para minimizar este riesgo, en el contrato se tiene que estipular la obligación que tiene el proveedor de servicios de proporcionarle al cliente, en un formato definido y aceptado por ambas partes, la información y/o datos y relaciones almacenadas en la base de datos. Se tiene que definir la estructura de la información que se entregará al cliente, para que se pueda hacer uso e interpretación de la misma con facilidad.

1.5 Otras clases de *Cloud Computing*

1.5.1 *Infrastructure as a Service (IaaS)*

En español, infraestructura como servicio, es la piedra angular del *Cloud Computing*. *IaaS*, por sus siglas en inglés, es un modelo de distribución de infraestructura de computación como un servicio, normalmente por una plataforma de virtualización, la cual permite el uso simultáneo de un mismo servidor físico entre varios sistemas operativos independientes.

En lugar de comprar servidores, espacio en un centro de datos o equipar la red, los clientes pagan por consumo de recursos: espacio en disco utilizado, tiempo en *CPU*, memoria, espacio en base de datos, transferencia de datos. Con *IaaS*, es posible escoger las cantidades de cada recurso que se necesita, permitiendo una escalabilidad automática o semiautomática, de forma que se pueda contratar más recursos según se necesiten.

Las empresas que proveen este tipo de servicio proveen y mantienen toda la infraestructura y conectividad necesaria. En la actualidad hay muchas empresas que ofrecen *IaaS*, por ejemplo: *Amazon*, *Joyent*, *Gogrid* y *Rackspace* entre otros.

Algo que hace la diferencia con el *hosting* virtual es que el provisionamiento de estos servicios se hacen de manera integral a través de la *Web*.

1.5.2 *Plataform as a Service (PaaS)*

En español, plataforma como servicio, regularmente se le conoce como la evolución de *SaaS*, porque en vez de acceder a un solo programa o una sola aplicación, se podrá acceder a todo un ambiente y trabajar en él.

PaaS, por sus siglas en inglés, es un modelo en el que se ofrece todo lo necesario para soportar el ciclo de vida completo de construcción y puesta en marcha de aplicaciones y servicios *web* completamente disponibles en *Internet*. Pone a disposición servidor de aplicaciones, bases de datos, *frameworks*, entre otros; además provee todas las facilidades al programador para hacer prototipos, analizar, desarrollar, realizar pruebas, documentar y poner en marcha aplicaciones, todo en un solo proceso. No más costos de actualización ya que el proveedor de *PaaS*, es el que asumirá dichos costos.

Implícitamente, debe tener un rendimiento óptimo, debe ser escalable y ofrecer seguridad y privacidad de la información que almacena. Otra característica importante es que no hay que descargar *software* para instalar en los equipos de los desarrolladores. *PaaS* ofrece varios servicios, pero todos como una solución integral en la *Web*.

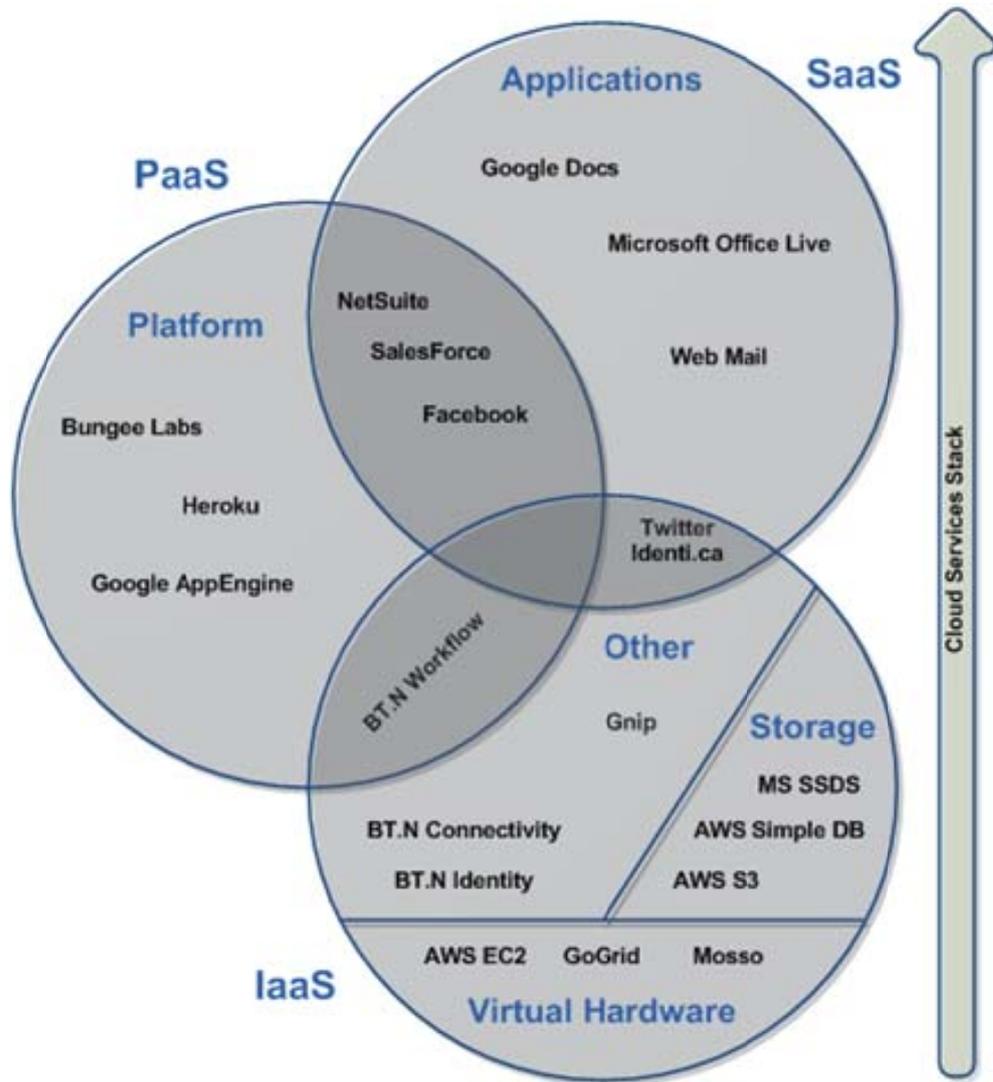
PaaS mejora la productividad de los equipos de desarrollo, ya que brinda la capacidad para desarrollar y compartir código fuente con diferentes desarrolladores, los cuales pueden estar ubicados en diferentes puntos geográficos

Algunos de los factores en contra de *PaaS* es que muchos desarrolladores y clientes tienen miedo de no tener la información en sus propios servidores, depender de la infraestructura de *hardware* del proveedor de *PaaS* y de tener una dependencia a las conexiones a *Internet*.

Al igual que en los anteriores modelos, son más las ventajas que las desventajas que ofrece *PaaS*, existiendo en el mercado varias compañías que ofrecen este tipo de servicio, siendo una de las más importantes *Force.com*.

En la Figura 3 se puede observar interactuando las tres clases de *Cloud Computing*. En la capa superior esta *SaaS*, el cual se caracteriza por ser una aplicación completa ofrecida como servicio bajo demanda; aquí se encuentra *Google Docs*, *Microsoft Office Live* y *Web Mail*. Algunas de estas aplicaciones han ido evolucionando operando en más de una capa en la nube como lo son: *NetSuite*, *SalesForce* (que opera en más de una capa con *Force.com*) y *Facebook* que interactúan en *PaaS* la cual ofrece toda una plataforma interactuando con un ambiente de desarrollo y una carga de servicios. Los ejemplos comerciales que tenemos de *PaaS* son: *Heroku*, *Bungee Labs* y *Google AppEngine* que sirve aplicaciones de la infraestructura *Google*. En la capa inferior encontramos a *IaaS* en la cual encontramos todos los recursos necesarios para formar la infraestructura, regularmente utilizamos la tecnología de virtualización. Un ejemplo comercial es *Amazon Web Services (AWS)*, cuyos servicios *AWS EC2* y *AWS S3* ofrecen cómputo y servicios de almacenamiento respectivamente.

Figura 3. SaaS, PaaS y IaaS



1.6 ¿Qué es el modelo tradicional del *software*?

El modelo tradicional del *software* es un modelo en el cual el cliente compra la licencia del producto para su uso, siendo el cliente el dueño del *software*, esto a diferencia del modelo SaaS donde el cliente paga una cuota mensual por el uso del *software*, siendo el propietario del mismo la empresa proveedora del servicio.

Regularmente, los costos de este tipo de *software* son muy elevados ya que se tiene que invertir en el desarrollo de la aplicación, infraestructura de *hardware*, seguridad del sistema, proveedores de *Internet* si es que el *software* se accederá por esta vía además de contar con el personal capacitado para dar soporte a los usuarios.

1.7 Ventajas y desventajas del modelo tradicional del *software*

1.7.1 Ventajas del modelo tradicional del *software*

- El cliente compra el *software* por lo que él es el único dueño de la aplicación.
- Acceso directo a la información ya que es almacenada directamente por el cliente.
- Integración con el resto de las aplicaciones, ya que muchas veces las aplicaciones están hechas a la medida.
- Acceso al código fuente. En algunas ocasiones se puede obtener el código fuente y con esto poder hacer modificaciones al *software* en cualquier momento

1.7.2 Desventajas del modelo tradicional del *software*

- La inversión inicial normalmente es muy alta.
- El desarrollo de sistemas a la medida del cliente lleva mucho tiempo.
- El programa se tiene que instalar en cada computadora, necesitando en algunas ocasiones de algún técnico especializado a un determinado costo.
- Constantemente surgen nuevas versiones del *software* las cuales tienen que pagarse por aparte.
- El usuario es el encargado de realizar las copias de seguridad de la información.
- Algunas compañías ofrecen el licenciamiento por máquina instalada, por lo que se tiene que hacer un cobro por cada máquina en la cual esté instalada la aplicación.
- El soporte tiene que ser dado por algún técnico para poder solucionar algún problema, teniendo que depender de la disponibilidad del técnico.
- A pesar de ser un *software* propio de la compañía, hay que pagar un costo de personal calificado interno o externo el cual se tiene que encargar de la implementación y soporte de los sistemas.
- Se tiene que hacer una inversión fuerte en la seguridad de los sistemas: *Firewall*, *Proxy*, *Antivirus* y tener personal calificado en seguridad de sistemas.

2 DESCRIPCIÓN DEL MODELO

2.1 Entorno tecnológico de proveedores del modelo SaaS

Las empresas que ofrecen este tipo de servicio tienen que tener estándares de calidad adecuados para ofrecer un buen producto y servicio. Es importante la integración de todos los procesos, la tecnología y las personas que están involucradas en el desarrollo, implementación, mantenimiento y soporte de la aplicación para brindar un servicio de calidad y en tiempo.

Las características principales que deberían tener las empresas que se dedican a este tipo de servicio son:

- Robustez y capacidad de infraestructura donde reside la aplicación.
- Capacidad de contar con la información en tiempo real. Todos los procesos y funciones tienen que estar disponibles cuando se requieran.
- No depender de un solo proveedor de servicio de *Internet*, ya que es el único canal por el cual accederán los usuarios a la aplicación.
- Personal capacitado y disponible para dar soporte a los usuarios cuando se requiera 24 horas por siete días.
- Capacidad de recuperación de la información y del sistema de forma inmediata ante cualquier eventualidad.
- Alto esquema de seguridad, el cual tiene que revisarse periódicamente para que esté preparado para funcionar siempre.
- Fácil adaptabilidad sobre nuevos requerimientos o necesidades de los clientes.

- La empresa tiene que ser estable, preferiblemente con presencia en varios países.
- La escalabilidad es un factor importante, por lo que el proveedor tiene que garantizar toda la infraestructura necesaria si el cliente llega a crecer.
- Es importante que este tipo de empresas tenga una visión clara y definida hacia dónde quieren llegar en el desarrollo de mejoras para la aplicación y de productos futuros a desarrollar.
- Como una forma de mejora continua, la empresa debería evaluar con sus clientes el nivel de satisfacción obtenido en los servicios que ofrece. Esto se podría realizar periódicamente con encuestas, en donde se evaluaría:
 - Calidad del servicio.
 - Calidad de la aplicación.
 - Procesos de comunicación.
 - Labor del ejecutivo que atiende la cuenta.
 - Imagen percibida de la empresa.
 - Capacitación a usuarios.
 - Satisfacción general.

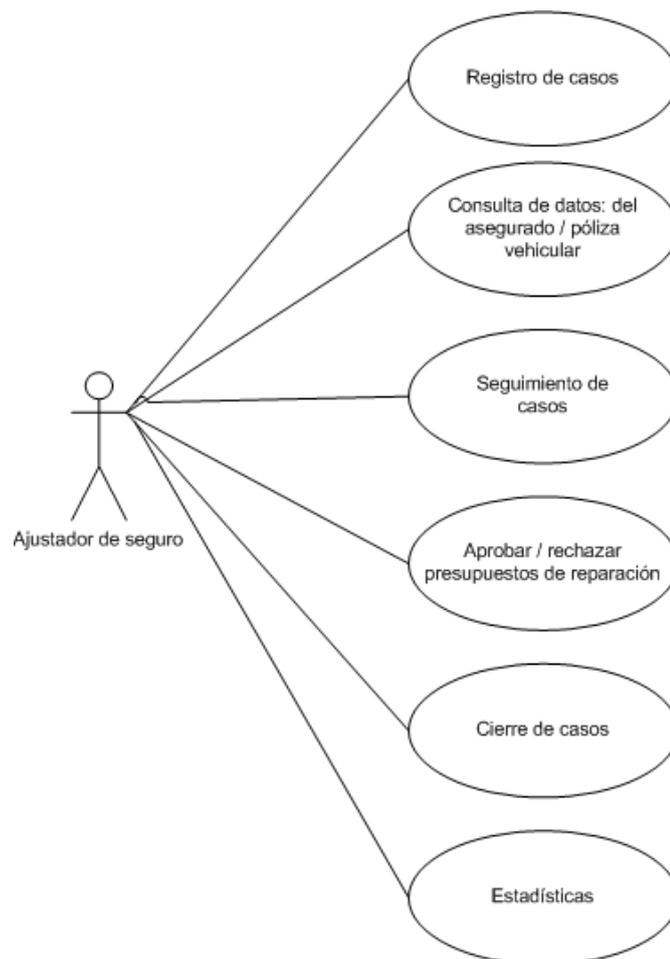
Todo esto con el objetivo de estar preparado para dar soporte a las necesidades de los clientes, convirtiéndose en un modelo de eficiencia operativa total.

2.2 Características del modelo

El modelo que se propone es un *software* utilizado por las compañías aseguradoras de vehículos, el cual proporcionará la solución definitiva para el

registro de los datos en el lugar donde ocurre un percance automovilístico, además que permitirá llevar un seguimiento de los casos en cualquier momento y en cualquier lugar. Con un *software* de este tipo, las metodologías utilizadas por los ajustadores de seguros se aplicarán de una forma más adecuada, elevando así el rendimiento de los mismos. SaaS para Aseguradoras, automatiza el registro de los datos en el lugar del percance, agiliza la aprobación de presupuestos para la reparación de vehículos, dando un mejor seguimiento a todos los casos (Figura 4).

Figura 4. Diagrama de casos de uso para un ajustador de seguros



Las características principales son:

- Registro de datos en tiempo real en el lugar del percance: el agente del seguro o ajustador de seguros contará con un dispositivo móvil en el cual tendrá instalada la aplicación con la cual podrá registrar todos los datos relacionados con el percance, los cuales pueden ser:
 - Número de póliza del asegurado.
 - Nombre y apellido del asegurado.
 - Datos de las licencias de los conductores que intervinieron en el percance: número de licencia, nombre de quien está la licencia, tipo de licencia, fecha de vencimiento.
 - Datos de los vehículos implicados en el percance.
 - Lugar del percance.
 - Declaraciones de los conductores que intervinieron en el percance.
 - Registrar las observaciones luego de las inspecciones ocular y física del lugar del accidente para determinar cómo fue el accidente, quién llevaba la vía, la posición final de cómo quedaron los vehículos implicados, si existen daños a terceros, etc.
 - Toma de fotografías.
 - Observaciones generales.

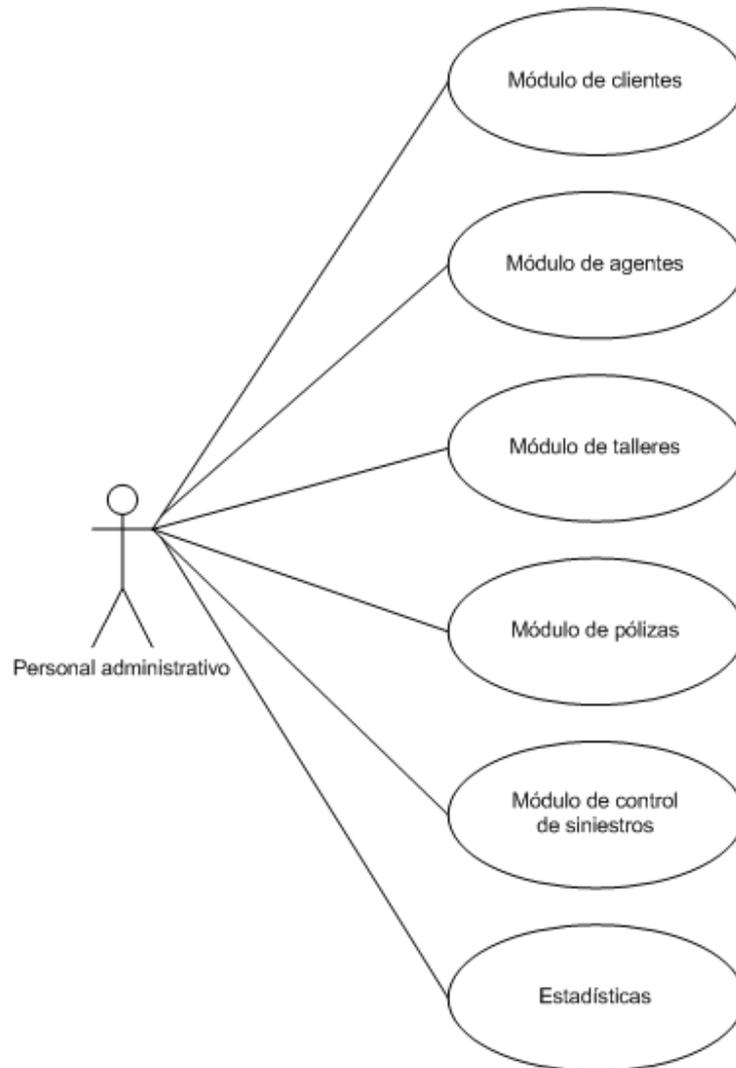
Con el registro y envío de los datos a la central, se abrirá un nuevo expediente con el cual se podrá continuar con la etapa de aprobación de presupuesto de reparación de vehículos.

- Aprobación de presupuestos en el taller de reparación: con la apertura del caso en el lugar del percance, el agente asegurador puede dar seguimiento a los casos abiertos en el taller donde el asegurado llevó el vehículo para la reparación. Se evaluará los costos en la reparación del

vehículo o vehículos implicados en el percance y si el presupuesto de reparación se considera adecuado se procederá en el sistema con la aprobación del mismo.

- Sincronización de la información en tiempo real a un servidor central: la información se transmitirá por *Internet* a un servidor en el cual se procesará la información recibida generando así los respectivos informes a las personas que correspondan para poder abrir los casos, aprobaciones o rechazos de presupuestos, consulta de casos, consulta de información del asegurado, etc.
- Fácil de usar. *SaaS* para Aseguradoras tendrá una interfaz sencilla y fácil de usar por los usuarios. Es indispensable tener acceso a *Internet* para que de esta forma permita conectar al sistema desde cualquier lugar y en cualquier momento y así registrar casos, consultar y actualizar datos de casos abiertos, visualizar datos relacionados con el asegurado, etc.
- Información centralizada. *SaaS* para Aseguradoras termina con hojas desordenadas de datos y con la posibilidad de no poder dar un seguimiento adecuado a los casos, ya que ahora se podrá contar con información actualizada en tiempo real. Únicamente necesitamos de una conexión a *Internet* para poder consultar la información en cualquier momento y lugar. Esta es una ventaja para el personal administrativo, el cual podrá acceder a los diferentes módulos del sistema (Figura 5).

Figura 5. Diagrama de casos de uso para personal administrativo



- Seguridad. Todos los datos están protegidos con algunas medidas como:
 - Seguridad física, con replica de información en un centro de datos externo
 - Cifrado de datos
 - Autenticación de usuarios
 - Seguridad de aplicaciones

- Sistema de detección de intrusos

Con todas estas medidas de seguridad se pretende dar confianza y tranquilidad a los usuarios (esta información se ampliará en el capítulo tres).

2.3 Módulos del sistema

SaaS para Aseguradoras cuenta con un *software* para el ajustador de seguros y otro para el personal administrativo; ambos tipos de *software* únicamente son para el control y cobertura de incidentes vehiculares. Las diferentes opciones se describirán a continuación.

2.3.1 Módulo del ajustador de seguros

2.3.1.1 Registro de casos

Se podrá registrar en el lugar del percance, todos los datos relacionados con el incidente o percance. Se registrarán tres tipos de percances:

- Accidente
- Robo parcial
- Robo total

Para los accidentes, los datos a registrar son:

- Registro de número de póliza del asegurado.
- Nombre y apellido del asegurado.
- Datos de licencias de los implicados.

- Datos de los vehículos implicados en el percance.
- Datos generales del lugar del percance.
- Toma de fotografías.
- Daños a terceros.
- Representación legal.
- Ajustador de seguros
- Observaciones generales del percance.

Con la anterior información se abrirá un caso que podrá ser supervisado por el ajustador de seguros o por el personal administrativo de la aseguradora.

Para el robo parcial los datos a registrar son:

- Registro de número de póliza del asegurado.
- Nombre y apellido del asegurado.
- Dirección del robo
- Fecha y hora del robo
- Equipo robado.
- Otros daños
 - Rotura de cristales
 - Cerraduras forzadas
 - Interior/exterior dañado
- Toma de fotografías.
- Ajustador de seguros
- Observaciones generales del robo.

En el caso de que sea un robo total, éste tiene que ser reportado directamente a la aseguradora, en la cual registrarán los datos del robo para abrir un caso o expediente.

Si en el momento de reportar un incidente (accidente o robo parcial), no se tiene acceso a la información en línea por medio del servicio de *Internet*, el incidente se podrá registrar en el *software* del dispositivo móvil y, cuando se restablezca el servicio de *Internet* la información será transmitida a los servidores centrales.

2.3.1.2 Consulta de datos del asegurado y de la póliza vehicular

Esta opción permitirá consultar los datos generales del asegurado y de la o las pólizas vehiculares que tenga asignadas. Esta consulta se podrá hacer ingresando: el número de póliza o nombre/apellido del asegurado o el número de la placa del vehículo. Los datos generales a mostrar son:

- De la póliza:
 - Número de póliza
 - Fecha de emisión de la póliza
 - Fecha de vencimiento de la póliza
 - Agente asegurador
- Del propietario:
 - Nombre y apellido
 - Número de Identificación Tributaria (NIT)
 - Dirección
- Del vehículo
 - Número de placa
 - Marca
 - Línea
 - Modelo
 - Número de chasis

- Número de pasajeros
- Tipo de vehículo
- Color
- Número de registro de motor
- Coberturas de:
 - Todo riesgo excluyendo robo
 - Robo y/o hurto total
 - Responsabilidad civil
 - Reembolso de gasto médicos
 - Indemnizaciones por accidentes personales
 - Muerte accidental de piloto del vehículo asegurado
 - Rotura de cristales
 - Asistencia en el camino
 - Asistencia jurídica
 - Territorio cubierto: nacional o regional

Si el asegurado cuenta con más de un vehículo asegurado, se mostrará la información anterior de cada vehículo.

2.3.1.3 Seguimiento a casos

Permitirá al ajustador de seguros, dar un seguimiento a todos aquellos casos registrados; para los casos aún abiertos indicará el estatus en que se encuentra el vehículo o vehículos implicados en el percance. El agente ajustador tendrá la posibilidad de cambiar el estatus actual de cada caso, lo cual es de gran beneficio para agilizar la aprobación de presupuestos, actualizar estatus de vehículos, brindando así un mejor servicio al cliente al contar con información actualizada, la cual puede ser consultada en cualquier momento y

en cualquier lugar y de esta forma el cliente conocerá el estatus actual de su vehículo.

Los diferentes estatus de casos son:

- Apertura de caso: cuando un caso es abierto por el agente ajustador en el lugar del percance, se registran los datos generales descritos en la sección 2.3.1.1.
- En taller (presupuesto pendiente de aprobación): luego de evaluar los costos de reparación del vehículo o vehículos, el agente ajustador tomará la decisión de aprobar o no el presupuesto de reparación. Este informe será transmitido por *Internet* a un servidor central al cual podrá acceder el personal encargado en la central de la aseguradora, quienes evaluarán si procede el presupuesto de reparación o es rechazado.
- En taller (presupuesto aprobado): al recibir la confirmación de la aprobación del presupuesto por parte de la central de la aseguradora, el agente ajustador autorizará al taller realizar la reparación del vehículo o vehículos implicados en el accidente.
- En reparación: indicará que el vehículo se encuentra en reparación.
- Trabajo finalizado: cuando la reparación del vehículo ha sido finalizada, el vehículo permanece en el taller hasta que se dé el visto bueno por el asegurado y sea cancelado el valor del deducible.
- Revisión del vehículo y cancelación de deducible: cuando la reparación del vehículo tiene el visto bueno del asegurado y éste cancela el valor del deducible.
- Salida del vehículo: cierre del caso si existe únicamente un vehículo en reparación, de lo contrario el caso continúa abierto hasta que se finalicen las reparaciones de los demás vehículos.

2.3.1.4 Estadísticas

Las estadísticas del módulo del agente ajustador es la mejor manera de conocer de una forma rápida y sencilla los registros estadísticos o informes de los diferentes registros de incidentes reportados por el agente ajustador. Las estadísticas con que contará el sistema son:

- Número de siniestros por asegurado.
- Número de siniestros por vehículo.
- Reporte de casos de accidentes en un período determinado: con información de número de póliza, nombre del asegurado, datos del vehículo, datos generales de los percances, estatus del caso, etc.
- Reporte de casos de robos parciales.

2.3.2 Módulo administrativo

2.3.2.1 Pólizas

2.3.2.1.1 Registro y emisión de pólizas

Permitirá registrar en el sistema los datos del propietario y de su vehículo, ingresando el vehículo a la cobertura del seguro y de esta forma se creará una nueva póliza. Los datos que debe llevar el registro de la póliza son:

- Datos generales de la póliza
 - Número de póliza
 - Nombre y apellido del propietario del vehículo
 - Dirección
 - Número de Identificación Tributaria (NIT)

- Agente asegurador
- Vigencia desde
- Vigencia hasta
- Prima neta
- Gastos de emisión
- Gastos por fraccionamiento
- IVA
- Descripción del vehículo
 - Marca
 - Línea
 - Modelo
 - Número de chasis
 - Pasajeros
 - Tipo de vehículo
 - Número de placa
 - Color
 - Número de registro de motor
- Coberturas y deducibles de:
 - Todo riesgo excluyendo robo
 - Robo y/o hurto total
 - Responsabilidad civil
 - Reembolso de gastos médicos
 - Indemnizaciones por accidentes personales
 - Muerte accidental de piloto del vehículo asegurado
 - Rotura de cristales
 - Asistencia en el camino
 - Asistencia jurídica
 - Territorio cubierto: Guatemala, Centro América, etc.

2.3.2.1.2 Informe general de pólizas

Teniendo en cuenta la importancia del acceso a la información, en esta opción se podrá generar diferentes informes de pólizas en pantalla, los cuales se podrán imprimir y exportar a distintos formatos (Excel, PDF, etc.), dependiendo del perfil del usuario. Entre los informes o reportes a generar están:

- Reporte de pólizas por clientes
- Reporte de pólizas por vencimiento
- Reporte de pólizas emitidas en un rango de fechas
- Reporte de pólizas por agente de seguros

2.3.2.1.3 Cobro de pólizas

Permitirá realizar el cobro de las primas pendientes a cancelar de cada póliza de vehículo asegurado. Los pagos podrán realizarse en efectivo, tarjeta de crédito y/o débito permitiendo la interacción con los bancos. Las principales opciones son:

- Cobro de primas
- Emisión de facturas
- Reporte de pagos de primas en efectivo, tarjeta de crédito y débito.
- Reporte de pólizas con primas atrasadas
- Reporte de pólizas con primas canceladas

2.3.2.2 Personal administrativo

Permitirá el alta/baja/modificación del personal administrativo contratado por la aseguradora. Los datos a registrar son:

- Nombre
- Apellido
- Dirección
- Teléfonos
- Número de Identificación Tributaria (NIT)
- Puesto
- Área dentro de la organización
- Fecha de ingreso
- Referencias personales
- Referencias laborales

2.3.2.3 Talleres

Permitirá el alta/baja/modificación de los talleres asociados a la red de talleres de la aseguradora. A los talleres se les permitirá ingresar al sistema en el cual por medio de un número de caso, podrán registrar los datos de los avances de reparación de vehículos con fotografías. Los datos para dar de alta un taller son:

- Nombre del taller
- Razón social
- Número de Identificación Tributaria (NIT)
- Nombre del propietario
- Dirección

- Teléfonos
- Contactos (personal)
- Número de fax

2.3.2.4 Control de siniestros

2.3.2.4.1 Registro de siniestros (percances o casos)

Si el equipo móvil del ajustador de seguros presenta un fallo por el cual no pueda registrar el percance, se contará con esta opción para que sea registrado el incidente a través del sistema del personal administrativo en la aseguradora. Existen tres diferentes tipos de percances:

- Accidente
- Robo parcial
- Robo total

Los datos a ingresar que proporcionará el ajustador de seguros, cuando se trata de un accidente son:

- Número de póliza
- Nombre y apellido del asegurado.
- Datos de licencias de los implicados.
- Datos de los vehículos accidentados.
- Datos generales del lugar del accidente.
- Toma de fotografías.
- Daños a terceros.
- Representación legal.
- Observaciones generales del percance.

- Ajustador de seguros
- Motivo por el cual no se registró el percance en el lugar.

En el caso de un robo parcial los datos a ingresar son:

- Registro de número de póliza del asegurado.
- Nombre y apellido del asegurado.
- Dirección del robo
- Fecha y hora del robo
- Equipo robado.
- Otros daños
 - Rotura de cristales
 - Cerraduras forzadas
 - Interior/exterior dañado
- Toma de fotografías.
- Ajustador de seguros
- Observaciones generales del robo.

Para el caso de un robo total, los datos a registrar son:

- Registro de número de póliza del asegurado.
- Nombre y apellido del asegurado.
- Dirección del robo
- Fecha y hora del robo
- Daños al asegurado
- Daños a terceros
- Agente asegurador

2.3.2.4.2 Seguimiento a casos

Todos los casos podrán ser supervisados por el personal asignado por la aseguradora, permitiendo verificar el estatus de los mismos.

Los talleres también podrán acceder a esta opción para que puedan actualizar los avances en la reparación de cada vehículo con sus respectivas fotografías para una mejor visualización. El usuario propietario del vehículo en reparación, podrá acceder al sistema para ver los avances generales de la reparación de su vehículo, permitiéndole hacer consultas en línea sobre la reparación del mismo.

Con esta opción se beneficiarán varias partes en el seguimiento a casos de reparaciones de vehículos, ya que estarán interactuando la aseguradora, los diferentes talleres y los múltiples usuarios asegurados con vehículos en reparación.

2.3.2.4.3 Ingreso de presupuestos de reparaciones

Con esta opción se permitirá al personal que labora en el taller de reparaciones de vehículos poder ingresar los presupuestos de reparación en línea con lo que se agilizará el envío del mismo a la central de la aseguradora para su revisión y aprobación. Esto será de mucho beneficio ya que ahorrará bastante trámite administrativo para iniciar el trabajo de reparación de un vehículo, y elevará la satisfacción del cliente al tener rápidamente su vehículo en reparación.

2.3.2.4.4 Aprobaciones de presupuestos de reparaciones de vehículos

Una vez recibido el informe del presupuesto de reparación de vehículos, el cual previamente ha sido ingresado en el sistema por parte del taller en donde se encuentra un vehículo para reparación, el Departamento de Ajustes evaluará si el presupuesto recibido está en el rango de costos para la reparación; de ser aceptado el presupuesto se autorizará al taller la reparación del vehículo para que su estatus cambie a “En taller (presupuesto aprobado)”, y se procederá a la reparación del vehículo.

2.3.2.4.5 Pago de deducible

Cuando el trabajo de reparación esté finalizado, el propietario del vehículo podrá realizar el pago en una agencia de seguro o en el taller en donde fue reparado el vehículo. Los pagos podrán realizarse en efectivo, tarjeta de crédito y/o débito permitiendo la interacción con los bancos. Luego de efectuado el pago, se emitirá una factura y una especificación de lo que se reparó en el vehículo, la cual servirá como garantía del trabajo realizado.

2.3.2.4.6 Pago por pérdida total

En los casos de robo total o pérdida total del vehículo, luego de un percance, el seguro pagará el monto estipulado en la póliza de seguro del carro. Para recibir el dinero de parte de la aseguradora, el asegurado primero tendrá que hacer efectivo el pago del deducible más otros gastos definidos en la póliza.

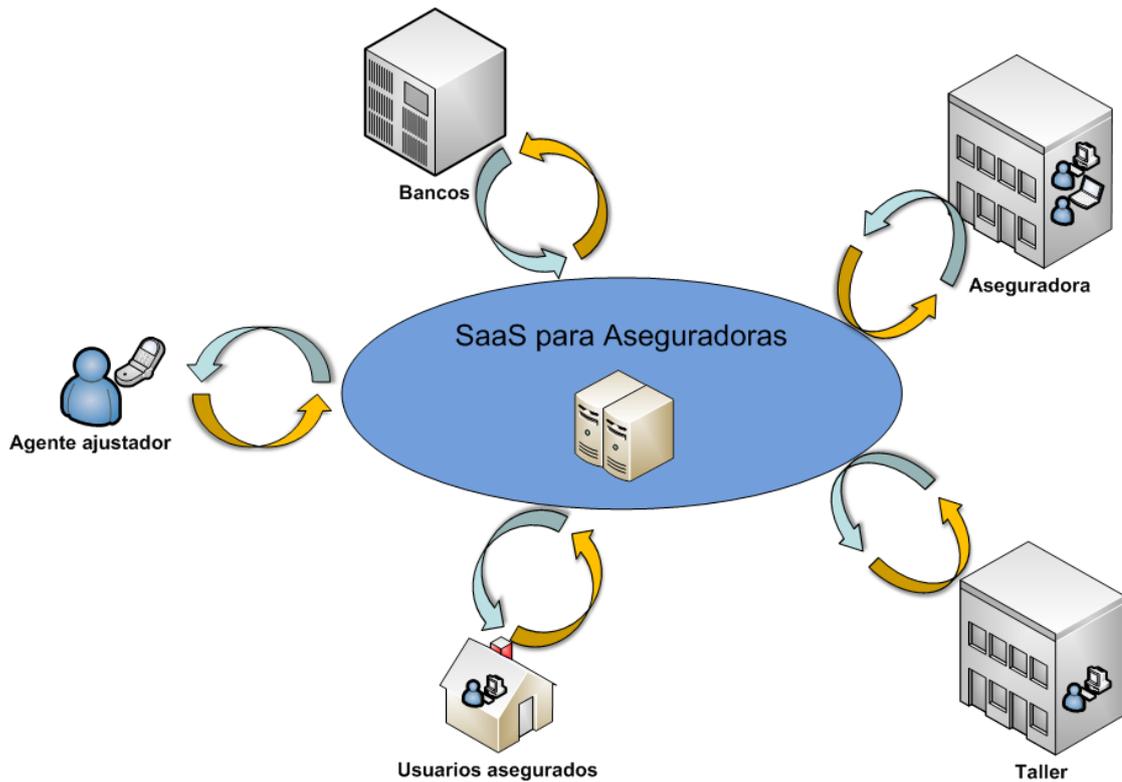
2.3.2.5 Estadísticas

Además de los informes que se mencionaron anteriormente, se contará con otros reportes estadísticos, los cuales son:

- Casos abiertos, mostrando los datos generales con su respectivo estatus.
- Casos cerrados
- Casos asignados por ajustador de seguros
- Informes de pagos de deducibles
- Lista de personal administrativo
- Lista de talleres afiliados a la red del seguro
- Reporte de casos con robos parciales
- Reporte de casos con robos totales

En la figura 6 se muestra un diagrama en el cual están interactuando los diferentes usuarios del sistema SaaS para Aseguradoras.

Figura 6. Diagrama de modelo SaaS para aseguradoras

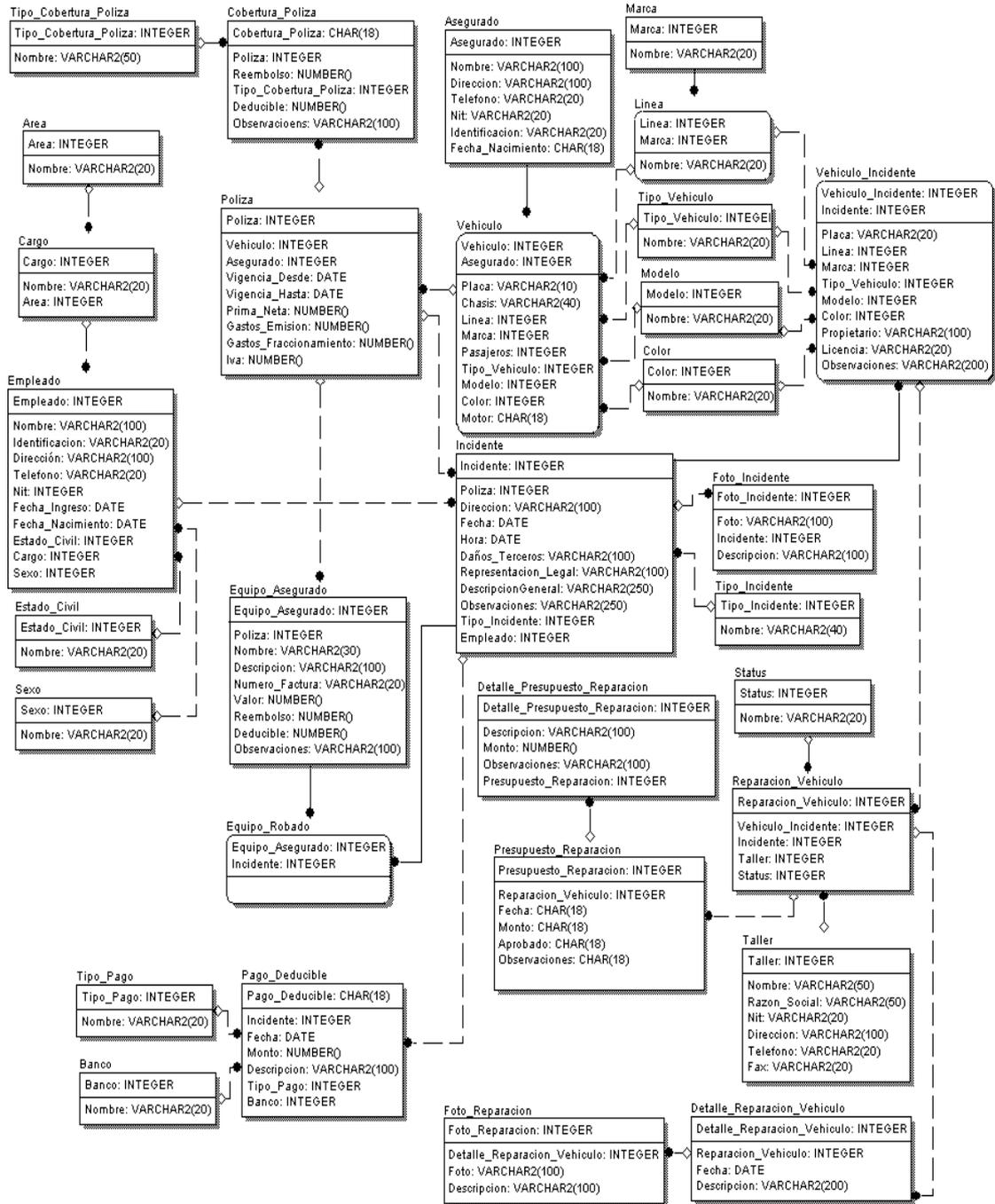


2.4 Diagrama entidad-relación

2.4.1 Diagrama entidad-relación de la aplicación

El diagrama entidad-relación de la aplicación del sistema SaaS para Aseguradoras (Figura 7) muestra las diferentes entidades y relaciones que existen entre ellas. Es importante indicar que por cada cliente (aseguradora) tendrá aplicado el diagrama en una base de datos independiente de las bases de los otros clientes.

Figura 7. Diagrama entidad-relación de la aplicación



A continuación se describirán todas las entidades del modelo entidad-relación de la aplicación:

ENTIDAD	DESCRIPCIÓN
Área	Representa la información de las diferentes áreas o departamentos dentro de la aseguradora: Gerencia, Informática, Mercadeo, etc.
Asegurado	Representa los datos generales de la persona propietaria del vehículo asegurado: nombre, dirección, teléfono, etc.
Banco	Representa la información de los bancos para referencia y verificación en el momento de realizar un pago con tarjeta de crédito y/o débito.
Cargo	Representa la información de los diferentes cargos que puede tener un empleado en la aseguradora.
Cobertura_Poliza	Representa la información de las diferentes coberturas que se tienen contratadas para un vehículo asegurado así como los deducibles y reembolsos. Algunos de los tipos de cobertura son: todo riesgo, robo y/o hurto total, responsabilidad civil, reembolso de gastos médicos, etc.
Color	Representa la información de los diferentes colores de vehículos.

Detalle_Presupuesto_Reparacion	Representa la información del detalle del presupuesto de reparación de un vehículo. Se ingresarán registros de repuestos, valor de mano de obra, etc.
Detalle_Reparacion_Vehículo	Representa la información del detalle de la reparación de un vehículo. Se ingresará registro de los avances en la reparación de cada vehículo.
Empleado	Representa la información de los datos generales de los empleados de la aseguradora, los cuales pueden ser: nombre, dirección, teléfono, fecha de ingreso, fecha de cumpleaños, etc.
Estado_Civil	Representa la información del estado civil que puede tener una persona: soltero, casado, viudo y divorciado.
Equipo_Asegurado	Representa la información del equipo asegurado de un vehículo, el cual ha sido previamente asegurado. Contiene datos como el número de póliza, número de factura de la compra del equipo, valor asegurado, valor de deducible, cantidad de reembolso, etc.
Equipo_Robado	Representa la información del equipo robado. Está formada por una relación de muchos a muchos entre las entidades Incidente y Equipo_Asegurado.
Foto_Incidente	Representa la información de las

	fotografías tomadas en el lugar del incidente: vehículos involucrados, partes de vehículos dañadas, otros daños, etc.
Foto_Reparacion	Representa la información de las fotografías de la reparación del vehículo con lo cual se podrá conocer la evolución de la reparación del vehículo.
Incidente	Representa la información de los datos generales de percances o incidentes vehiculares. La información a registrar es: dirección del percance, fecha, póliza de vehículo asegurado, datos de vehículos involucrados, tipo de incidente, descripción general, etc.
Linea	Representa la información de las diferentes líneas que puede tener una marca de vehículo, por ejemplo: Tercel, Civic, Corolla, etc.
Marca	Representa la información de las diferentes marcas de vehículos, por ejemplo: Toyota, Honda, Volvo, etc.
Modelo	Representa la información del modelo (año) que puede tener un vehículo, por ejemplo: modelo 1990, 2010, etc.
Pago_Deducible	Representa la información de los pagos de deducibles de acuerdo al tipo de cobertura cubierta.

Poliza	Representa la información de los datos generales de los vehículos asegurados por propietario así como las fechas de vigencia del seguro vehicular y los registros de gastos de las pólizas.
Presupuesto_Reparacion	Representa la información general de los datos del presupuesto de reparación, los cuales son: fecha de realización de presupuesto, monto total del presupuesto, si el presupuesto ha sido aprobado, etc.
Reparacion_Vehiculo	Representa la información general de la reparación de un vehículo. La información a registrar es: datos del vehículo en reparación, taller que está realizando la reparación, el estatus de la reparación del vehículo, etc.
Sexo	Representa la información del sexo de una persona: masculino y femenino.
Status	Representa la información de los diferentes estatus que un caso o incidente registrado puede tener. Por ejemplo: apertura de caso, en taller (presupuesto pendiente de aprobación), en taller (presupuesto aprobado), en reparación, trabajo finalizado, pago de deducible, salida del vehículo, etc.

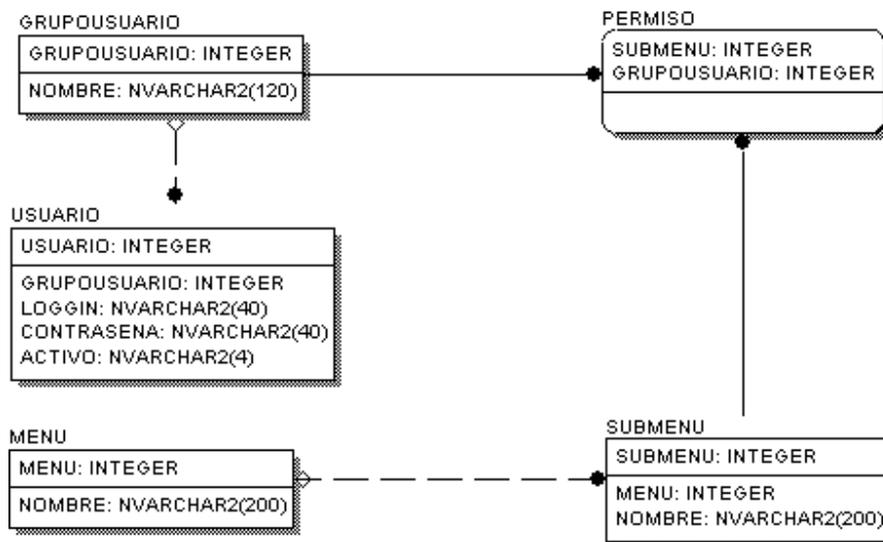
Taller	Representa la información de los talleres de reparación de vehículos a los que una persona con vehículo asegurado puede llevar para su reparación.
Tipo_Cobertura_Poliza	Representa la información de los diferentes tipos de cobertura vehicular, por ejemplo: todo riesgo excluyendo robo, robo y/o hurto total, responsabilidad civil, reembolso de gastos médicos, rotura de cristales, etc.
Tipo_Incidente	Representa la información de los tipos de incidentes a reportar, los cuales pueden ser: accidente, robo parcial y robo total.
Tipo_Pago	Representa la información de los tipos de pago que se pueden hacer en el sistema, por ejemplo: efectivo, tarjeta de crédito y/o débito, etc.
Tipo_Vehiculo	Representa la información de los diferentes tipos de vehículos, por ejemplo: particular, pesado, etc.
Vehiculo	Representa la información de los datos de los vehículos asegurados. Los datos a registrar son: propietario, número de placa, número de chasis, número de registro de motor, marca, línea, modelo, tipo de vehículo, color,

	número de pasajeros.
Vehiculo_Incidente	Representa la información de los vehículos que participaron en un percance o incidente.

2.4.2 Diagrama entidad-relación de seguridad

El diagrama entidad-relación de la seguridad del sistema SaaS para Aseguradoras (Figura 8) muestra las diferentes entidades y relaciones que existen entre ellas.

Figura 8. Diagrama entidad-relación de seguridad



A continuación se describirán todas las entidades del modelo Entidad-relación de seguridad del sistema:

ENTIDAD	DESCRIPCIÓN
GrupoUsuario	Representa la información de los diferentes grupos o tipos de usuarios que tendrán acceso a las diferentes opciones del programa.
Menu	Representa la información del nombre de cada módulo/opción del menú principal que posee el sistema, por ejemplo: módulo de pólizas, estadísticas, seguimiento a casos, etc.
Permiso	Representa la información del tipo de usuario y de las diferentes opciones que tiene acceso en el programa.
SubMenu	Representa la información de las opciones del programa, que estén contenidas dentro de una opción del menú principal.
Usuario	Representa la información de los diferentes usuarios con acceso al sistema.

2.5 Requerimientos de *hardware*, *software* y comunicaciones en los clientes del sistema

2.5.1 Requerimientos de *hardware*

Los requerimientos de *hardware* para *Laptops/Desktops* y dispositivos móviles, para la adecuada utilización del sistema *SaaS* para Aseguradoras son los siguientes:

- Para *Laptops/Desktops*:
 - Procesador Intel Core 2 Duo de 2.93 o superior.
 - Memoria DDR2 de 4 GB de RAM
 - Disco duro de 500 GB
 - Monitor, mouse y teclado
 - DVD-RW

- Para dispositivos móviles *PPC*:
 - Procesador Intel PXA270 520 Mhz o procesador Marvell PXA310 624 Mhz.
 - 128 de SDRAM de memoria principal para *software* y aplicaciones, 256 MB flash ROM SDRAM.
 - Pantalla táctil TFT a color con una resolución de 480 x 640 pixeles con 260,000 colores.
 - Conexión wireless 802.11b, Bluetooth, y puerto infrarrojo (Serial IR)
 - Batería recargable de Ion de Litio.
 - Tarjeta de expansión SD Card

- Para dispositivos móviles *SmartPhones*:
 - Procesador Qualcomm MSM7201A 528 MHz
 - Pantalla táctil TFT a color con una resolución 480 x 640 pixeles.
 - Memoria interna de: 128 *RAM* y 128 *ROM*.
 - Memoria expandible vía tarjeta microSD
 - Modem integrado
 - Puerto miniUSB 2.0, Bluetooth 2.0
 - WiFi 802.11b/g
 - Cámara integrada con una resolución mínima de cinco megapíxeles.

2.5.2 Requerimientos de *software*

- Para dispositivos móviles: *Windows Mobile* 6.0 o superior.
- Para *notebooks* y *desktops*:
 - Sistema operativo: *Windows* 2000, *Windows* XP, *Windows* Vista, *Windows* 7, Linux.
 - Navegador *Web*: *Internet Explorer* 7 o superior, *Firefox*, *Google Chrome*, etc.

2.5.3 Requerimientos de comunicaciones

Para los usuarios administrativos, de talleres y usuarios asegurados, se requiere tener conexión a *Internet*, la cual debe tener un ancho de banda sugerido de 1 mbps. Hay que tener en cuenta que, si existe gran cantidad de usuarios conectados al sistema desde un mismo punto, por ejemplo la central de la aseguradora, la navegación y el acceso a la información podría ser un poco lento, por lo que se recomienda un ancho de banda mayor.

Para los usuarios con dispositivos móviles se recomienda una navegación WAP, 2G o superior.

2.6 Esquema de soporte y niveles de servicio

Dentro del modelo de SaaS para Aseguradoras se contempla un área de soporte o *Help Desk* la cual solucionará todas las incidencias reportadas por los usuarios. Dependiendo de la incidencia presentada, se definirán varios niveles de soporte; en algunos casos una incidencia requiere de la intervención de más de un nivel de soporte. También se definirán tiempos máximos de resolución de cada incidencia según la gravedad de la misma.

Los niveles de soporte con que contará el modelo SaaS para Aseguradoras son:

- a) Soporte nivel A: el cual será proporcionado por el administrador del sistema. Regularmente esta persona formará parte del departamento de tecnología del cliente. El se encargará de:
 - Soporte de las funcionalidades del sistema
 - Generación de informes y reportes a los diferentes usuarios
 - Administración del sistema: creación de usuarios, ingreso de informes, seguimiento a casos, etc.
 - Seguimiento a nuevos requerimientos o modificaciones a los existentes.
 - Configuración e instalación de la aplicación en dispositivos móviles.

- b) Soporte nivel B: este soporte será brindado por un *Help Desk* de la empresa proveedora del servicio. Los casos a atender en este nivel son:

- Atender y dar seguimiento a todos los casos técnicos que el administrador no pueda resolver y los cuales pueden ser resueltos sin modificar las bases de datos o el código fuente.
- Verificación de información faltante o corrupción de datos.
- Analizar el problema y determinar si el mismo se puede resolver o si se tiene que escalar al área de programación y desarrollo.
- Analizar solicitudes de nuevos requerimientos por parte del cliente.
- Verificar que las comunicaciones con el servidor estén siempre disponibles.
- Verificar que el sistema esté funcionando correctamente.
- Seguimiento y aplicación de nuevas versiones y actualizaciones.

c) Soporte nivel C: este tipo de soporte será brindado por parte del área de programación y desarrollo de la empresa proveedora del servicio. Los casos a atender son:

- Analizar y realizar las modificaciones necesarias en el código fuente del sistema o en la base de datos con el objetivo de solucionar los problemas presentados o detectados.
- Soporte experimentado en casos que no han podido ser resueltos en los niveles de soporte 1 y 2.

Los anteriores son los niveles de soporte estándares que se deberían definir para una adecuada atención y seguimiento de los problemas presentados hasta que sean resueltos. También se tiene que considerar:

- Horario de atención de los niveles de soporte, los cuales pueden ser distintos entre cada nivel. El horario de soporte debe estar bien definido para poder brindar el servicio adecuado a los usuarios. Otra

consideración a tener en cuenta es que si el sistema funciona para usuarios ubicados en varios países, es probable que los usuarios se encuentren en diferentes zonas horarias.

- Nombre, correo electrónico y teléfono de las personas que brindan el soporte. Es importante que exista una persona de turno para atender todos aquellos casos fuera del horario establecido en los diferentes niveles de soporte.
- Definición de tiempos de resolución de problemas dependiendo la gravedad del problema. Se definirá tres tipos de prioridades según la magnitud del caso presentado:
 - Prioridad máxima.
 - El sistema está fuera de funcionamiento.
 - El problema afecta a todos o a varios usuarios, no permitiéndoles realizar tareas importantes en el sistema.
 - No existe conectividad de *Internet* en los servidores.
 - Prioridad media.
 - El problema se presenta en uno o varios usuarios, pero el sistema sigue funcionando. En la mayoría de los casos las personas pueden seguir con el uso de otras funcionalidades.
 - Prioridad baja:
 - El problema reportado afecta únicamente a una persona y no le permite continuar con el uso del sistema. Para los demás usuarios, el sistema continúa funcionando en forma normal. Se busca la solución en forma específica para resolverlo a la brevedad.
 - Alguna funcionalidad específica no anda bien, pero el usuario puede hacer uso de otras funcionalidades del sistema.

3 TECNOLOGÍA

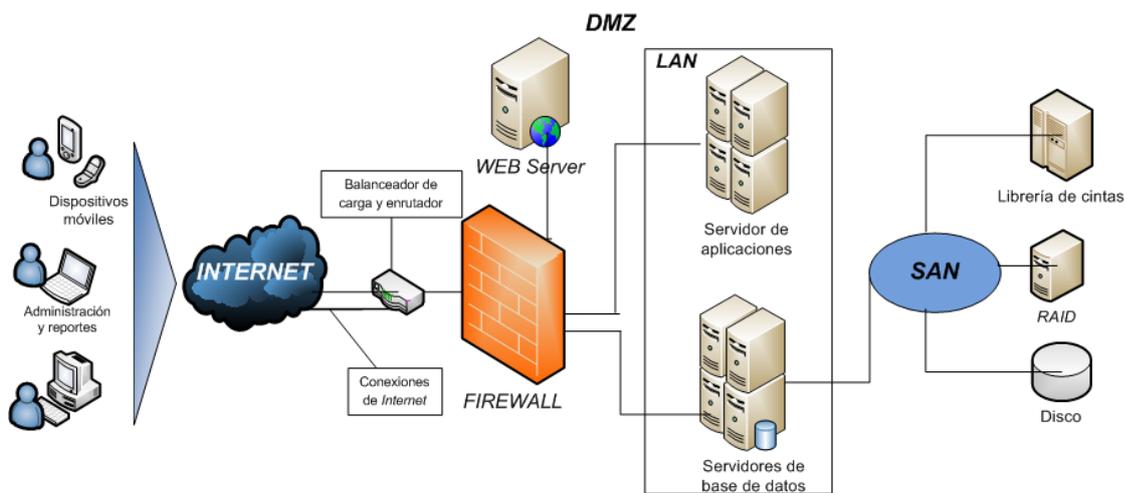
3.1 Arquitectura de infraestructura

Es importante contar con una muy buena infraestructura que garantice la disponibilidad de la información y servicios en cualquier momento. El sistema SaaS para Aseguradoras es un sistema crítico que debe estar disponible siempre que los usuarios quieran hacer uso del mismo. De esto surge la importancia de contar con una potente y segura arquitectura que garantice:

- Disponibilidad de la información. El sistema tiene que estar disponible en todo momento debido a que los usuarios podrán acceder al mismo a través de *Internet* en cualquier momento y desde cualquier lugar. Contar con una infraestructura y los mecanismos que garanticen el acceso a la información y a los servicios, permitirá asegurar que si por alguna razón exista un fallo, el acceso al sistema y a la información siempre estará disponible.
- Seguridad de la información. Tiene como fin la protección de la información y de los sistemas de información que garanticen la confiabilidad, integridad y disponibilidad de la información y datos. La infraestructura a utilizar tiene que garantizar una adecuada seguridad física y seguridad lógica.
 - La seguridad física se refiere a la protección del *hardware* y de los soportes de datos, así como de los edificios e instalaciones que los albergan. Contemplan las situaciones de incendios, sabotaje, robos, catástrofes naturales, etc.

- La seguridad lógica se refiere a la seguridad de uso del *software*, a la protección de los datos, procesos y programas, así como la del acceso autorizado de los usuarios a la información.
- Alto rendimiento. La infraestructura debe brindar una muy buena respuesta a múltiples solicitudes de clientes en el menor tiempo posible, permitiendo que si las necesidades aumentan se pueda tener una escalabilidad en los recursos fácilmente.

Figura 8. Arquitectura de infraestructura



La figura 8 muestra el diagrama de la arquitectura de infraestructura propuesta, el cual se describe a continuación.

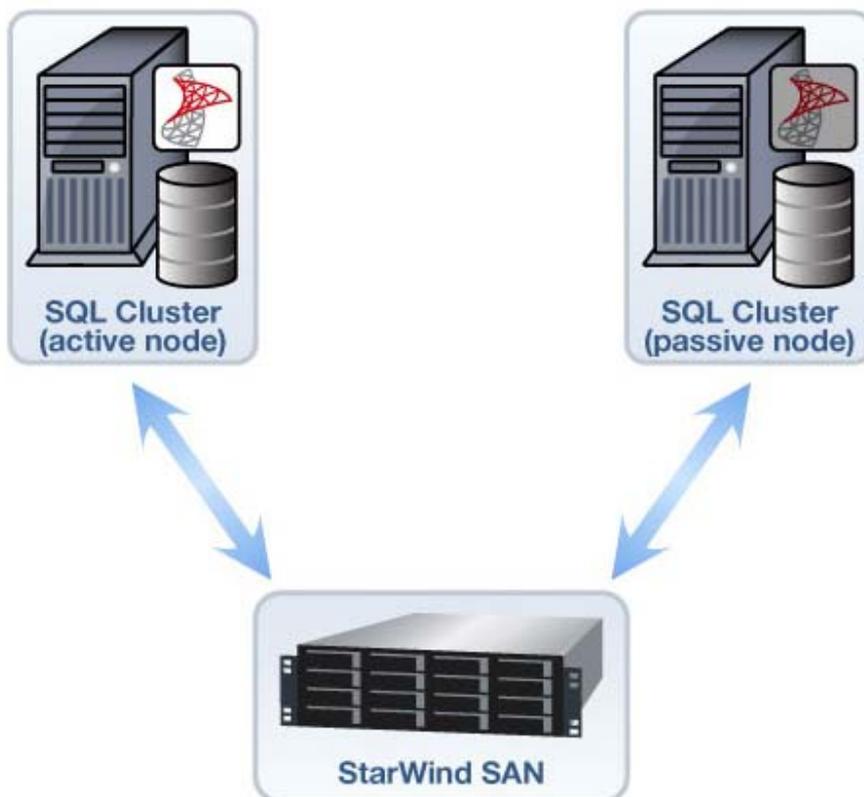
- Se debe contar con al menos dos conexiones a *Internet*, una principal y una de respaldo proporcionadas por proveedores diferentes; esto

garantizará que si una conexión falla la otra entrará a funcionar en forma automática. Aquí estarán los *routers* que direccionarán ambas conexiones.

- Se contará con un balanceador de carga y enrutador de tráfico a través de *Internet*, por ejemplo *Barracuda Link Balancer*, el cual es una muy buena solución y de gran alcance para el enrutamiento y gestión de tráfico a través de varias conexiones de *Internet*. Está diseñado para escalar a los altos requisitos de ancho de banda y proporcionar la continuidad del negocio. Capaz de realizar una conmutación por error automática en caso de falla de algún enlace, lo cual garantizará que el acceso al sistema estará siempre disponible.
- *Firewall*, el cual garantizará el filtrado de la información no permitiendo el acceso de usuarios no autorizados. Con un *Firewall* se podrá supervisar y restringir la información que viaja entre la red interna y la red exterior o *Internet*. Funciona básicamente como un filtro que controla todas las comunicaciones que existen entre una red y otra, permitiendo configurarlo para permitir o denegar el acceso a través de la red.
- *Web Server*, el cual es un programa que sirve para atender y responder las diferentes peticiones de los navegadores, proporcionando los recursos que soliciten usando el protocolo *HTTP* o *HTTPS*. El Servidor *Web* que se utilizará es el *Internet Information Server 7 (IIS7)* explicado a detalle en la sección 3.3.3. Este servidor estará ubicado en la *DMZ (Demilitarized Zone)*; tenerlo en la *DMZ* es importante como medida de seguridad (explicado en la sección 3.5), ya que permitirá conectar la red interior con el exterior pero no permitirá el acceso del exterior hacia la red interna.
- Los servidores de aplicaciones y bases de datos estarán instalados en *clúster*, lo cual garantizará que si algún servidor presenta alguna falla el otro lo reemplazará de forma automática, siendo esto transparente para

el usuario. La base de datos a utilizar es *SQL Server 2008*, explicada en la sección 3.3.1, la cual estará implementada en *clúster* de la siguiente manera: se tendrá un esquema de *Failover* o conmutación por error en el cual todo el trabajo es realizado por un solo nodo llamado el nodo activo, además, se tendrá un nodo pasivo el cual es el encargado de gestionar todo el trabajo del *clúster* si el nodo activo llega a fallar. Desde el punto de vista de rendimiento, esta es la mejor solución, pero esto significa que se hará menor uso del *hardware*, lo que significará ser una solución más costosa, pero mucho más efectiva (Figura 9).

Figura 9. *SQL Server 2008 Cluster*



- Los servidores de aplicaciones y bases de datos estarán en forma de granja de servidores, los cuales servirán para brindar una gran capacidad mucho más económica que contar con un súper computador.
- Para el almacenamiento utilizaremos una *SAN (Storage Area Network)*. Una *SAN*, por sus siglas en inglés, es una red propia de almacenamiento que combina los beneficios de la conectividad con la potencia de los servidores; esto ha sido posible con la tecnología de *Fibre Channel* para crear una red propia dedicada exclusivamente a dispositivos de almacenamiento. Con una *SAN* se podrá administrar los dispositivos de almacenamiento de forma dedicada y especializada ya sea en plataformas homogéneas o heterogéneas, de forma escalable y segura.
- Se utilizará la virtualización de servidores con lo cual se podrá agrupar varios servidores en uno solo, manteniendo cada ambiente individual y a la vez optimizando el uso de los recursos, como lo son: procesadores, memoria, almacenamiento, etc. El objetivo es maximizar el uso de los recursos balanceando las cargas de trabajo en una administración centralizada.
- Es importante contar con un centro de datos externo (ubicado geográficamente en un lugar diferente al centro de datos principal), el cual tiene que tener la misma infraestructura del centro de datos principal. Esto se hace con el objetivo de que si se presentará algún fallo o desastre en el centro de datos principal la infraestructura de respaldo entraría a funcionar en forma inmediata y transparente para el usuario. Tiene que existir un canal de comunicación entre ambos sitios para que se pueda lograr hacer en el sitio alterno las réplicas de los servidores de aplicaciones y bases de datos. Las bases de datos tendrán un reflejado sincrónico, el cual se explicará más adelante.

3.2 Arquitectura de componentes

3.2.1 Base de datos

La base de datos a utilizar es *SQL Server 2008* ya que provee un motor de base de datos escalable con muy buen desempeño para el funcionamiento de aplicaciones de misión crítica, ya que tiene la capacidad de estar funcionando siempre, reduciendo los gastos indirectos de dirección y desempeño, además de brindar los más altos niveles de seguridad y fiabilidad para obtener los mejores resultados en aplicaciones empresariales.

Otras características que posee *SQL Server 2008* son:

- Alto rendimiento en ejecución de transacciones con un gran manejo de consultas complejas y bases de datos grandes.
- Fácil de utilizar para construir, administrar e implementar aplicaciones de negocios.
- Gestión. Posee un completo interfaz gráfico que reduce la complejidad innecesaria de las tareas de administración y gestión de base de datos.
- Orientada al Desarrollo. *Visual Basic.NET*, *Visual j#.NET*, *Visual C++* y muchas otras aplicaciones son compatibles con *SQL Server 2008*.
- Soporte para información sobre *XML*, *Filestream*, información espacial.
- Proporciona una plataforma de datos segura para la información en misión crítica. Además permite encriptar toda la base de datos, archivos de datos o archivos log, sin necesidad de hacer cambios en las aplicaciones.
- Espejado de base de datos. Permite tener versiones sincronizadas de bases de datos, las cuales pueden estar ubicadas en dos sitios distintos

con el objetivo de brindar alta disponibilidad por la redundancia del espejo. Permite manejar dos tipos de espejado de base de datos:

- Espejado sincrónico: cuando se realiza el proceso de reflejo de la transacción del servidor principal al servidor espejo para realizar la ejecución de la transacción, se necesita un mensaje de recibido del servidor espejo por lo que debe existir una buena comunicación entre ambos servidores, ya que por esta operación, si no se tiene buena comunicación se puede disminuir el rendimiento aunque se aumente la alta disponibilidad. En caso de que el servidor quede fuera de servicio y se realice la conmutación por error o *failover*, se tiene la certeza de que ambos servidores están sincronizados.
- Espejado asincrónico: en este tipo de espejado, el servidor principal no espera el mensaje de recibido del servidor espejo, así que hace la ejecución de la transacción directamente, con lo cual se mejorará el rendimiento del sistema, pero esto no asegura en el caso de una conmutación por error que ambos servidores se encuentren sincronizados.
- Administración de políticas que permiten manejar de una mejor forma una o más instancias de *SQL Server 2008*.
- Compresión de datos, permitiendo el almacenamiento de una forma más eficiente, reduciendo los requisitos de almacenamiento para los datos.
- *SQL Server Clustering*. Esta es una de las principales características de *SQL Server 2008* que se tiene que considerar en la implementación de la base de datos, ya que la aplicación es de misión crítica y si se tiene una implementación de *clúster* de servidores de *SQL* cuando uno de los servidores o nodos falla, las responsabilidades son asumidas por el otro servidor *clúster* en forma inmediata.

3.2.2 Sistema operativo

El sistema operativo que residirá en los servidores es *Microsoft Windows Server 2008*, el cual representa la nueva generación de sistemas operativos de *Microsoft*. Este sistema operativo ofrece a los profesionales de *IT* un mejor control sobre la infraestructura y red, permitiendo que ellos se dediquen específicamente a las necesidades prioritarias del negocio. *Windows Server 2008* está diseñado específicamente para ser una de las plataformas más productivas para la virtualización y administración de cargas de trabajo, la creación y alojamientos de aplicaciones y servicios *Web*, con un alto esquema de seguridad.

Las características fundamentales por las que se utilizará el sistema operativo *Windows Server 2008* en los servidores para la aplicación *SaaS* para *Aseguradoras* son las siguientes:

- Alta disponibilidad. Es una de las características principales en *Windows Server 2008*, garantizando de esta forma que las aplicaciones fundamentales, servicios e información estén siempre disponibles cuando los usuarios las requieran.
- Administración del servidor. En *Windows Server 2008* esta tarea está bastante simplificada, utilizando herramientas de administración centralizada e interfaces sencillas, permitiendo entre otras tareas:
 - Configuración sencilla de servidores nuevos.
 - Administrar desde una consola la configuración del servidor y la información del sistema, y de esta forma administrar todas las funciones instaladas en el servidor, además de mostrar el estado del servidor, y de existir algún problema con la configuración del mismo, lo indicará.

- Administrar con facilidad servidores de la red central como aquellos distribuidos en otras ubicaciones, por ejemplo en sucursales.
 - Administración de impresoras.
- Virtualización. Contiene *Windows Server Virtualization (WSv)* por sus siglas en inglés, la cual es una tecnología de virtualización que tiene muy buenas características de administración y virtualización, ayudando a que las organizaciones aprovechen la familiaridad y flexibilidad de uso con la familia de servidores *Microsoft*. Teniendo en cuenta que la seguridad es uno de los temas fundamentales en la implementación de un servidor, *WSv* proporciona muchos beneficios de seguridad de virtualización sin necesidad de adquirir *software* de terceros. *WSv* aumenta la seguridad de servidores consolidados, los cuales son servidores que alojan varias máquinas virtuales (*VM*), evitando la exposición de aquellas *VM* que contienen información confidencial, protegiendo el sistema operativo *host* de todo riesgo que puede tener un sistema operativo invitado. *WSv* posee también un alto rendimiento en la administración de cargas de trabajo proporcionando una mayor flexibilidad de recursos.
- Plataforma *Web* y de aplicaciones. Contiene una plataforma segura y de fácil administración para desarrollar y alojar aplicaciones y servicios *web* que se entregan en el servidor o a través de la *web*. Muchas de las nuevas características son:
 - Administración simplificada.
 - Mayor seguridad
 - Plataforma *Web* personalizada y simple, que integra para publicación en la *web* a *Internet Information Server 7.0 (IIS7)*, *ASP.NET*, *Windows Communication Foundation* y *Microsoft Windows SharePoint*.

- Mayor rendimiento y escalabilidad.
- Administración a mayor detalle de cómo y en qué momento las aplicaciones y servicios *web* utilizan los recursos del sistema operativo.
- Seguridad. *Windows Server 2008* posee muchas características que mejoran el nivel de la seguridad, simplificando la administración e implementación de opciones y configuraciones relacionadas con la seguridad. Las características principales en seguridad, son las siguientes:
 - Mejoras en el *Firewall*.
 - Protección de acceso a redes, permitiendo que se configure la seguridad antes de permitir a los clientes acceso a la red.
 - Cifrado y protección de datos mediante cifrado de la unidad de disco.
 - Herramientas de criptografía.
 - Aislamiento del servidor y el dominio, y de esta forma separar los recursos del servidor y del dominio para permitir únicamente el acceso a equipos autenticados y autorizados.
 - Controlador de dominio de solo lectura (*RODC*), por sus siglas en inglés, es un controlador de dominio que se puede instalar en sitios remotos con niveles bajos de seguridad física.

3.2.3 Servidor Web

El servidor *Web* a utilizar es el *Internet Information Server 7.0 (IIS7)* por sus siglas en inglés, ya que posee las herramientas y funciones necesarias para administrar de forma sencilla y segura un servidor *web*. Algo muy importante es que viene integrado con *Windows Server 2008*.

Las ventajas principales para utilizarlo son:

- Características más eficientes y sencillas de administración. El Administrador de IIS7 es una herramienta bastante eficiente para la administración del servidor *web*, permitiendo compatibilidades para ejecuciones de *IIS* y *ASP.NET*, datos de usuario e información de diagnósticos en tiempo real; además, permite una administración local, remota o por *Internet*.
- Mejor seguridad.
 - Autenticación básica.
 - Autenticación de *Windows*.
 - Autenticación de asignaciones de certificado de clientes.
 - Autorización para URL.
 - Restricciones de *IP* y dominio.
- Diagnósticos y solución de problemas más eficiente. Los componentes de seguimiento y solución de problemas, ofrecen información detallada y útil para que los administradores y desarrolladores puedan aislar las páginas y código que presente problemas.
- Bastante soporte técnico a un costo reducido.
- Procesamiento de páginas *Active Server Pages (ASP)* y *ASP.NET*, además, soporta características de desarrollo de aplicaciones: *CGI*, Extensiones *ISAPI*, Filtros *ISAPI*, inclusiones del lado del servidor.

3.2.4 Plataforma de desarrollo

La plataforma a utilizar para el desarrollo del sistema *SaaS* para Aseguradoras es *Microsoft .Net*, la cual posee un conjunto de nuevas y potentes tecnologías para el desarrollo de aplicaciones, haciendo posible la

integridad con algún desarrollo existente y una inter-operatividad entre los diferentes lenguajes de programación. Las características para su selección en el desarrollo del sistema son:

- Modelo de programación consistente y sencillo completamente orientado a objetos.
- Multilenguaje, permitiendo de esta forma contar con programadores dedicados a desarrollar en el lenguaje de programación en que más experiencia posean.
- Es una solución rápida, segura y potente para desarrollar aplicaciones y permitir un acceso simple a todo tipo de información desde cualquier tipo de dispositivo.
- Soporte en la utilización de *XML* como medio para la interacción entre aplicaciones y servicios *Web*, utilizando el protocolo *SOAP (Simple Object Access Protocol)* para facilitar el intercambio de datos codificados en *XML* y la realización de llamadas a procedimientos remotos.
- Soporte para desarrollo de aplicaciones en dispositivos móviles. Los dispositivos móviles soportados y a utilizar en el sistema *SaaS* para Aseguradoras para el agente ajustador son: *Smartphone* y *Pocket PC*. Debido a que las aplicaciones móviles necesitaran estar conectadas a *Internet* para transmitir o recibir información, se podrá optar por trabajar con *Web Services* y acceso remoto a la base de datos, el cual puede ser por vía *WiFi*, *GPRS*, *3G*, etc.
- Utilizando el desarrollo del *software* basado en una arquitectura de capas (se explicará en el inciso 3.3.1), permitirá la reutilización de componentes y si se desea cambiar de un dispositivo a otro, se podrá hacer simplemente cambiando la capa de presentación.

3.3 Arquitectura del *Software*

La arquitectura del *software* o también llamada arquitectura lógica, es la que define la estructura de un sistema, el cual incluye componentes de *software*, propiedades de los componentes y las relaciones de estos; la arquitectura del *software* es el diseño de más alto nivel de la estructura de un sistema.

Con el paso del tiempo la forma en que se desarrollaba el *software* se ha ido perfeccionando, creando guías, estructuras, formas de programar con el objetivo de resolver algún problema o desarrollar alguna solución para un nuevo requerimiento.

Con lo descrito anteriormente, ha surgido la importancia de la arquitectura del *software*. A partir de la década del sesenta, se empezó a hablar sobre dicho término, pero fue hasta la década del noventa cuando toma mayor popularidad tras la denominada crisis del *software*, la cual se fundamentó en que luego de desarrollar algún *software*, este no generaba los resultados deseados, además de tener alto costo y poca flexibilidad.

La arquitectura del *software* ha ido evolucionando, existiendo diferentes tipos de arquitecturas, entre ellas:

- Monolítica.
- Cliente-servidor.
- Arquitectura de n-capas.

Para el modelo SaaS para Aseguradoras se utilizará una arquitectura multinivel, específicamente una arquitectura de tres capas, en la cual a cada

nivel se le asignan funciones específicas, lo que permite diseños de arquitecturas que sean escalables, las cuales pueden ampliarse con facilidad si las necesidades aumentan. La arquitectura de tres capas se describirá a continuación.

3.3.1 Arquitectura de tres capas

Esta arquitectura se basa en que la carga y funciones se divide en tres capas, y cuyo objetivo primordial es la separación de la lógica del negocio de la lógica del diseño.

Las capas en que se divide son las siguientes:

- Capa de presentación (Interfaz de usuario).
- Capa de negocio (Cálculo y procesos).
- Capa de datos (Almacenamiento).

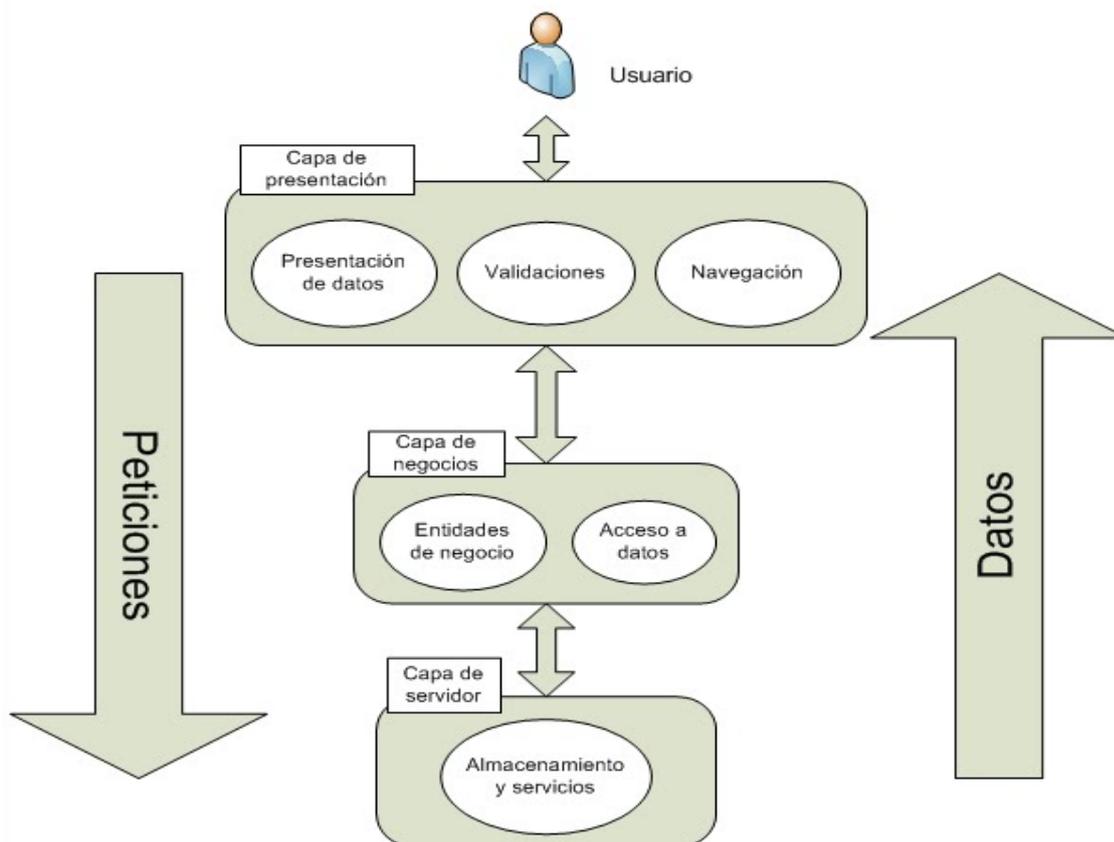
Una de las ventajas principales de este tipo de arquitectura es que cualquier desarrollo se puede llevar a cabo en varios niveles y si existiese algún cambio, en la mayoría de los casos, sólo se hace sobre el nivel requerido sin llegar a revisar los demás niveles.

La arquitectura de tres capas es bastante orientada al desarrollo de aplicaciones *Web*; para el modelo *SaaS* para Aseguradoras, en los usuarios administrativos la capa de presentación se accederá a la página de la aplicación *web* por medio del navegador y en los dispositivos móviles se tendrá la aplicación instalada en los dispositivos; en ambos casos se comunicarán con el servidor *web* y al servidor de aplicaciones en el que estará la lógica del negocio

(capa de negocio) y posteriormente se accederá a la base de datos (capa de datos) para almacenar información o extraerla a solicitud de la capa de negocio.

Al tener agrupadas estas funcionalidades en capas lógicas, se podrán relacionar unas con otras, por ejemplo, el agente ajustador interactúa en la capa de presentación solicitando información sobre personas aseguradas o de casos y a la vez registra incidentes. Todas sus solicitudes serán procesadas en la capa de negocio, la cual se encargará de la gestión de las solicitudes o de la traducción de las mismas para que éstas sean procesadas en la capa de datos; si esta última capa requiere devolver información, esta información será transmitida a la capa de negocio, la cual administrará la información recibida y de ser necesario la transmitirá a la capa de presentación. La Figura 10 muestra el esquema de arquitectura del *software* en tres capas.

Figura 10. Arquitectura del *software* de tres capas



3.3.2 Capa de presentación

Es la capa que ve el usuario, es la que le muestra el sistema, le comunica información y captura la información que él ha ingresado. Esta capa se comunica directamente con la capa del negocio y es conocida como interfaz gráfica. La capa de presentación se compone de:

- Interfaz de usuario: ofrece al usuario información, sugerencias, acciones y captura los datos de entrada ingresados por el usuario.
- Lógica de presentación: hace referencia a todo el procesamiento requerido para mostrar los datos y transformar los datos de entrada en acciones que se puedan ejecutar en la capa de negocio.

La capa de presentación cumple tres funciones principales, que son:

- Formateo de datos. Si se tienen sistemas diferentes que utilizan códigos distintos para representar los caracteres en la pantalla, la capa de presentación sirve como traductor entre los diferentes tipos de códigos.
- Cifrado de datos. Protege la información durante la transmisión. Por ejemplo, las transacciones financieras utilizan el cifrado para la transmisión confidencial de la información enviada por *Internet*. Se utiliza una clave de cifrado para cifrar los datos desde el origen y luego descifrarlos en el lugar de destino.
- Comprensión de datos. La comprensión funciona por medio de algoritmos para reducir el tamaño de los archivos. El algoritmo busca patrones de *bits* repetidos en el archivo y los reemplaza con un *token*. El *token* es un patrón de *bit* mucho más corto que representa un patrón largo.

El modelo SaaS para Aseguradoras proveerá una Interfaz sencilla para el usuario contando con las siguientes características:

- Mostrará las pantallas del sistema para poder hacer uso del mismo en sus diferentes opciones: ingreso al sistema, registro de incidencias, consulta de datos de incidentes reportados, consulta de datos de pólizas y personas aseguradas, seguimiento a casos, estadísticas, etc.

- Debe facilitar al usuario el ingreso de la información contando con ayuda visual para el ingreso de la misma, por ejemplo controles de ayuda visual o *tool tips*.
- Validar y/o restringir la información a ingresar en cada campo. Por ejemplo si se va a ingresar el nombre de una persona a la cual se asegurará su vehículo, el campo debería únicamente permitir letras. Otro tipo de validación es comprobar si el valor ingresado está en un rango de valores permitidos, por ejemplo, la edad de un propietario de vehículo a asegurar debería estar contemplada en un rango entre 1 a 100 años.
- Facilitar al usuario el ingreso de información utilizando controles como: *Combo Box*, *Check List*, Controles de Selección múltiple entre, otros.
- Mostrar mensajes en pantalla sobre el estado de una transacción. Por ejemplo, si se está consultando los datos de un asegurado, mostrar mensajes de “Ejecutando consulta”.
- Mostrar la información al usuario en un formato adecuado para su fácil visualización y comprensión.

3.3.3 Capa de negocio

Llamada también de lógica del negocio, porque es aquí donde se establecen todas las reglas (algoritmos) que deben cumplirse para el intercambio de información entre la base de datos y la interfaz del usuario. Esta capa se comunica con la capa de presentación recibiendo todas las solicitudes para luego presentar los resultados. También se comunica con la capa de datos para solicitar al gestor de base de datos almacenar o traer datos de él.

La capa de negocio ocupa un lugar muy importante en la construcción de una infraestructura de *software* que permitirá el crecimiento y la extensibilidad

de servicios para todas las aplicaciones existentes y todas aquellas nuevas a desarrollar, haciendo de esta forma aplicaciones más poderosas.

Para una adecuada creación de la capa de negocio se tiene que tener en cuenta:

- Modelo de objetos del dominio. Contiene una relación entre las entidades basándose en los requerimientos del usuario. Se utilizan para hacerlos diagramas de casos de uso y diagramas de clases en *UML*.
- Seguridad. Si bien este es un tema que se tiene que ver en todas las capas, en el modelo *SaaS* para Aseguradoras se pondrá especial énfasis en la capa de negocio al ser el enlace directo con la capa de datos; la seguridad podría implementarse por medio de un sistema basado en roles para permitir acceso únicamente a los usuarios autorizados.
- Entidades. Una entidad es la representación de un objeto o concepto del mundo real que se describe en una base de datos. Las entidades están conformadas por atributos. Un ejemplo de entidad puede ser una Persona asegurada y sus atributos serán: nombre, apellido, dirección, etc. Muchas veces se confunde el concepto “Objetos de negocio” con entidades, siendo diferentes, ya que un Objeto de negocio está conformado por propiedades y métodos y las entidades sólo se componen de propiedades o atributos. El acceso a los objetos del negocio siempre debería realizarse a través de entidades, a menudo denominadas *DTO (Data Transfer Object)*, ya que permiten mover datos entre capas. En el modelo de *SaaS* para Aseguradoras se necesitará poder hacer transferencia de información entre componentes, por ejemplo, en el momento de registrar un percance automovilístico será necesario tener acceso a la capa de de datos para desplegar toda la información referente a la persona asegurada con los respectivos datos

de su póliza vehicular. Este tipo de consultas serán realizadas a través de entidades del negocio.

- Componentes. Los componentes de *software* son recursos desarrollados para un fin concreto y que pueden formar solos o conjuntamente con otros un entorno funcional que se requiera por cualquier proceso predefinido. Los componentes en el sistema SaaS para Aseguradoras juegan un papel importante al ser una aplicación distribuida, ya que el cliente y el componente pueden conectarse sin la necesidad de un sistema intermedio, debido a que el cliente llama a los métodos del componente sin tener que preocuparse de niveles más complejos. Otra de las características importantes es que cualquier componente que sea desarrollado como parte de una aplicación distribuida puede ser reutilizado.
- Clases. Una clase es la representación abstracta de un objeto el cual a su vez es la consolidación de una clase. Un objeto puede ser cualquier cosa, por ejemplo una casa, y la clase sería los planos de la casa, en los cuales la casa se ve representada por completo, ya que en el plano se podrá ver qué tendrá la casa, dónde lo tendrá, cómo estará hecho. La clase es sólo el plano del que después se podrá construir el objeto. La clase por sí sola no es de mucha utilidad hasta que se convierta en un objeto. Es muy importante utilizar las clases ya que a nivel de lenguaje de programación una clase agrupa variables y funciones, lo que permite una organización del código de programación, haciendo el sistema más potente y flexible permitiendo separar la aplicación en objetos, subsistemas y componentes, siendo estos reutilizables de forma completa o parcial en más de un proyecto con poco o ningún cambio en el código, con lo que se ahorrará tiempo al no tener que hacer algo dos veces

- Reglas del negocio. Son la principal herramienta que tienen las empresas para implementar sus estrategias. Los motores de las reglas del negocio consisten en módulos de *software* que aceptan las reglas en un determinado formato aplicándolas en los objetos del negocio. Se utilizan muchas veces sentencias *IF, THEN, ELSE*.
- Reglas de validación. Son un conjunto de reglas a aplicar sobre los objetos del negocio para validar su estado.

Teniendo en cuenta las características anteriores se podrá construir una adecuada capa de negocio.

3.3.4 Capa de datos

Es la capa donde se encuentra los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos, los cuales se encargan del almacenamiento de la información o de la recepción de peticiones para devolver los datos a la capa de negocio.

Es importante tener en cuenta que debe existir una consistencia en el manejo de la información, es decir que tanto los datos que ingresan como los que se extraen de la base de datos deben ser consistentes y precisos.

Es en esta capa donde se definen las consultas a la base de datos. En el modelo de *SaaS* para Aseguradoras se definirá:

- Procedimientos almacenados, los cuales se ejecutarán con alguna llamada directa del cliente para realizar alguna tarea específica.
- Disparadores, los cuales son procedimientos que se ejecutan en forma indirecta a consecuencia de una solicitud de un cliente. Están asociados directamente a sentencias *INSERT, UPDATE* y *DELETE*.

- Procedimientos programados, los cuales son procedimientos a ejecutarse periódicamente según una calendarización definida; por ejemplo, la generación mensual de un cierre de pólizas de vehículos.

3.4 Seguridad

La seguridad de los sistemas es un tema muy importante, especialmente cuando se está conectado a *Internet*. Para contar con un nivel de protección adecuado es necesario tener una adecuada seguridad física y seguridad lógica que garanticen el acceso a la información y a los servicios.

Para la seguridad física se contará:

- Cuartos fríos. El equipo de cómputo debe estar almacenado en cuartos fríos que garanticen una temperatura adecuada para los mismos.
- Revisión de controles ambientales. Verificar si se cuenta con deshumidificadores, aire acondicionado, fuentes de energía continua, extintores de incendios, etc.
- Plan de mantenimiento. Los equipos principales deben tener un plan de mantenimiento adecuado que garantice su funcionamiento continuo.
- Acceso restringido a los equipos.
- Instalaciones adecuadas para la infraestructura.
- Instalaciones antisísmicas.

Para la seguridad lógica, es importante contar con una política de seguridad para prevenir el acceso no autorizado de usuarios a los recursos de la red y proteger contra la manipulación de la información contenida en los servidores. La política de seguridad contempla:

- Utilizar un *Firewall*. Esto es algo fundamental a utilizar ya que garantizará que si la red tiene algún tipo de conexión hacia el exterior, o a otras redes, esta sea segura y así evitar cualquier tipo de violaciones, permitiendo únicamente el acceso a los usuarios autorizados. El *Firewall* puede ser monitorizado, y si aparece algo sospechoso éste debe generar alguna alarma ante cualquier ataque o problema en el tránsito de datos.
- Codificación de datos. Utilizar la criptología, criptografía y criptociencia, las cuales tienen propósitos similares que es codificar y decodificar información para que distintos usuarios puedan enviar información a su destino sin que un tercero tenga acceso a la misma. El objetivo es designar una clave o algoritmo que codifique la información, es decir, encriptándola, y de esta forma se protegerá de otros usuarios que quieran tener acceso a la información y así se asegurará que ésta llegue de forma segura a su destino en donde se decodifica la información. Es muy importante que las claves o algoritmos para codificar la información sean muy difíciles de descifrar para evitar que ésta sea decodificada por algún intruso que logre acceder a la misma.
- Utilizar certificados de seguridad *SSL (Secure Socket Layer)*.
- Utilizar y Monitorear la Zona Desmilitarizada o *DMZ (Demilitarized Zone)* por sus siglas en inglés, tiene como objetivo principal que las conexiones desde la red interna y externa de la *DMZ* estén permitidas, pero que las conexiones desde la *DMZ* sólo sean permitidas a la red externa, es decir, que los equipos de la *DMZ* no pueda conectar a la red interna. Con esto se asegurará que los equipos (*host*) de la *DMZ* puedan prestar servicios a la red externa y a su vez proteger la red interna de intrusos que quieran acceder a ella a través de los equipos ubicados en la *DMZ*.
- *Software* protector de intrusos. Se deben poseer Antivirus, *Anti-spyware* y *Anti-Malware* y mantenerlos actualizados, con esto se garantizará tener protegidos los sistemas e información.

- Sistema de respaldo con réplica en algún centro de almacenamiento externo. Esto tiene las siguientes ventajas:
 - Mantener copias de seguridad en una ubicación física diferente a donde se encuentran los datos originales.
 - La copia de seguridad remota no requiere la intervención del usuario para cambio de cintas, procesos manuales, etiquetar cds, etc.
 - Muchos sistemas de copia de seguridad remota se ejecutan de modo continuo, almacenando la información en el instante en que ésta cambie, de esta forma se podrá contar con la última copia de seguridad si el sistema falla.
 - Varios sistemas de copias de seguridad remota envían los datos encapsulados en tramas seguras SSL.
- Autenticación de usuarios al sistema. Todos los usuarios con acceso al sistema tendrán roles con permisos definidos para la utilización del mismo.
- Política de contraseñas. Las características de la política de contraseñas son:
 - Longitud mínima de ocho caracteres.
 - El sistema debe solicitar cambio de contraseña en su primera entrada.
 - La contraseña no puede ser igual al número de usuario.
 - Libre de caracteres idénticos consecutivos numéricos o grupos alfabéticos, es decir, no son permitidos, por ejemplo: 123, abc. Sí son permitidos secuencia de caracteres: 146, adc.
 - El sistema debe solicitar cambio de contraseña periódicamente.
 - Cada vez que se haga cambio de contraseña, no se podrá volver a ingresar alguna contraseña utilizada anteriormente.

- Tener instalado en los equipos únicamente el *software* necesario para las aplicaciones. Tener un control del *software* da la seguridad de procedencia del mismo con lo que se reducen riesgos y así se evitan copias ilegales, teniendo de esta manera un método correcto en reinstalaciones de *software* a causa de un desastre.
- Plan de respuesta a incidentes. Es importante tener un plan de respuesta de incidentes, el cual permitirá minimizar los efectos ocurridos luego de una violación, además de permitir al equipo de seguridad desarrollar una guía de acciones a seguir para minimizar los daños potenciales. La respuesta a los incidentes debe ser apropiada y ejecutarse rápidamente. Un plan de respuesta a incidentes debería dividirse en:
 - Acción inmediata para detener o minimizar el incidente.
 - Investigación del incidente.
 - Restauración de los recursos afectados.
 - Reporte de incidente a los canales apropiados.
- Bitácora. Servirá para registrar todas las actividades realizadas en el sistema y en la base de datos. Esto se realizará por medio de:
 - *Triggers* y *Log* de Transacciones en la base de datos.
 - Código de programación en la capa de negocio.
 Con el registro de la bitácora, se podrá conocer:
 - Las operaciones de inserción, modificación y borrado (*INSERT*, *UPDATE* y *DELETE*) realizadas por cada usuario en las diferentes tablas de la base de datos.
 - Registro de los datos nuevos, borrados y modificados en la base de datos por tabla y por usuario.
 - Las diferentes pantallas a las que ingrese el usuario en el sistema.
 - Los datos consultados en el sistema.
 - Los usuarios que han ingresado en el sistema y el tiempo que han permanecido en él hasta que salen del sistema.

- o Las transacciones ejecutadas satisfactoriamente y las fallidas en la base de datos.

CONCLUSIONES

1. El enfoque del modelo SaaS es proporcionar a los usuarios todos los beneficios de contar con un *software* desarrollado para alguna área específica, sin la necesidad de que el usuario se preocupe por el desarrollo, puesta en marcha, mantenimiento, actualización de versiones y soporte que implica un *software* de este tipo, ya que la empresa desarrolladora del *software* asume todas estas tareas.
2. La mayoría de las aplicaciones bajo el modelo SaaS son desarrolladas para ser accedidas a través de *Internet*, representa una gran ventaja porque se puede acceder al sistema y a la información, desde cualquier momento y en cualquier lugar, únicamente contando con un canal de comunicación a *Internet* en la computadora o dispositivo móvil (*PPC*, *Smartphone*).
3. Las empresas desarrolladoras de *software* que brindan este tipo de servicio, tienen que tener toda la infraestructura física y lógica para que el acceso al sistema y a la información esté disponible siempre; además de contar con los mecanismos de seguridad que garanticen la integridad y confidencialidad de la información, la cual tiene que ser accedida únicamente por las personas autorizadas para el uso de la misma.
4. Con el modelo SaaS para Aseguradoras, las metodologías de trabajo utilizadas por los ajustadores de seguros se aplicarán de una forma más adecuada, elevando así el rendimiento de los mismos, ya que

automatizará el registro de los datos de un percance o incidente automovilístico en el lugar en que ocurre.

5. Uno de los problemas actuales cuando una persona lleva su vehículo a reparación, luego de haber tenido un incidente vehicular, es el tiempo que transcurre desde el momento en que el vehículo ingresa al taller hasta el momento en que inicia su reparación. SaaS para Aseguradoras permitirá agilizar el inicio de la reparación de un vehículo, ya que al momento de su ingreso al taller, se realizará el presupuesto de reparación para ser enviado y aprobado por la aseguradora.

RECOMENDACIONES

1. Para las empresas desarrolladoras de *software* que deseen desarrollar aplicaciones bajo el modelo *SaaS*, tienen que identificar áreas o segmentos de mercado con un potencial número de clientes que puedan contratar este tipo de aplicaciones.
2. El acceso a la mayoría de aplicaciones bajo el modelo *SaaS* es a través de *Internet*, lo cual significa que se tienen que tener buenas medidas de seguridad tanto lógicas como físicas para que personas no autorizadas, no puedan hacer uso de este sistema e información.
3. Debido a que toda la información se encontrará en los servidores de la empresa que provee el servicio, se tienen que firmar acuerdos de confidencialidad entre ambas partes en los cuales se tiene que especificar que: la información estará bien protegida, será accedida únicamente por el cliente y que la empresa proveedora de servicios, por ningún motivo, podrá hacer uso de la misma.

BIBLIOGRAFÍA

1. <http://arantxa.ii.uam.es/~ortigosa/poo/09-J2EE.pdf>. Fecha de consulta: agosto de 2010.
2. http://ciscoschool.net/wp-content/uploads/2010/03/iaas-paas-saas_2.png<http://www.alegsa.com.ar/Dic/arquitectura%20de%20software.php>. Fecha de consulta: mayo de 2010.
3. <http://elementalsql.blogspot.com/2010/02/creating-sql-server-2008-cluster-in.html>. Fecha de consulta: agosto de 2010.
4. http://es.wikipedia.org/wiki/Arquitectura_de_software. Fecha de consulta: mayo de 2010.
5. http://es.wikipedia.org/wiki/Capa_de_presentación. Fecha de consulta: agosto de 2010.
6. http://en.wikipedia.org/wiki/Cloud_computing. Fecha de consulta: abril de 2010.
7. http://es.wikipedia.org/wiki/Crisis_del_software. Fecha de consulta: julio de 2010.
8. http://es.wikipedia.org/wiki/Internet_Information_Services. Fecha de consulta: julio de 2010.

9. http://es.wikipedia.org/wiki/Microsoft_Visual_Studio_.NET. Fecha de consulta: agosto de 2010.
10. http://es.wikipedia.org/wiki/Programaci%C3%B3n_por_capas. Fecha de consulta: julio de 2010
11. http://es.wikipedia.org/wiki/Seguridad_inform%C3%A1tica. Fecha de consulta: agosto de 2010.
12. http://es.wikipedia.org/wiki/Software_como_servicio. Fecha de consulta: abril de 2010.
13. http://es.wikipedia.org/wiki/Windows_Server_2008. Fecha de consulta: agosto de 2010.
14. [http://es.wikipedia.org/wiki/Zona_desmilitarizada_\(inform%C3%A1tica\)](http://es.wikipedia.org/wiki/Zona_desmilitarizada_(inform%C3%A1tica)). Fecha de consulta: agosto de 2010.
15. <http://faustol.wordpress.com/2007/06/03/aplicaciones-moviles-con-net/>. Fecha de consulta: agosto de 2010.
16. <http://geeks.ms/blogs/eliasmereb/archive/2007/07/09/modelos-de-virtualizaci-243-n.aspx>. Fecha de consulta: julio de 2010.
17. <http://globaliza.blogia.com/temas/tecnologia.net.php>. Fecha de consulta: julio de 2010.
18. <http://hosting.solingest.com/cluster-de-servidores.html>. Fecha de consulta: julio de 2010.

19. <http://jmhogua.blogspot.com/2007/02/capa-lgica-de-negocios.html>. Fecha de consulta: julio de 2010.
20. <http://kartones.net/blogs/coco/archive/2009/11/27/la-capa-de-negocio-i-conceptos-generales.aspx>. Fecha de consulta: julio de 2010.
21. <http://kartones.net/blogs/coco/archive/2009/12/14/la-capa-de-presentaci-243-n-conceptos-b-225-sicos.aspx>. Fecha de consulta: julio de 2010.
22. <http://knol.google.com/k/aplicaciones-distribuidas#>. Fecha de consulta: abril de 2010.
23. http://lanrouter.com/index.php?option=com_content&task=view&id=38&Itemid=71. Fecha de consulta: mayo de 2010.
24. <http://pymecrunch.com/software-como-servicios-saas-concepto>. Fecha de consulta: mayo de 2010.
25. <http://sacares.com/informatica/hardware/DAS,SAN,NAS.htm>. Fecha de consulta: junio de 2010.
26. <http://techlosofy.com/cloud-computing-o-computacion-en-nube/>. Fecha de consulta: abril de 2010.
27. [http://technet.microsoft.com/es-es/library/cc753198\(WS.10\).aspx](http://technet.microsoft.com/es-es/library/cc753198(WS.10).aspx). Fecha de consulta: julio de 2010.
28. <http://technet.microsoft.com/en-us/magazine/2007.03.sqlclusters.aspx>. Fecha de consulta: julio de 2010.

29. http://users.dsic.upv.es/asignaturas/facultad/lsi/ejemplorup/Model_Negocio.html. Fecha de consulta: julio de 2010.
30. <http://www.act.cl/web/Productos/Infraestructura/Storage/ArquitecturadeAlmacenamientoSAN/tabid/228/language/es-CL/Default.aspx>. Fecha de consulta: julio de 2010.
31. <http://www.anexom.es/servicios-en-la-red/web-20/que-es-software-como-servicio-saas/>. Fecha de consulta: abril de 2010.
32. http://www.barracudanetworks.com/ns/products/link_overview.php. Fecha de consulta: agosto de 2010.
33. <http://www.desarrolloweb.com/articulos/1681.php>. Fecha de consulta: julio de 2010.
34. <http://www.di-mare.com/adolfo/cursos/2007-2/pp-3capas.pdf>. Fecha de consulta: julio de 2010.
35. http://www.docirs.cl/arquitectura_tres_capas.htm. Fecha de consulta: julio de 2010.
36. <http://www.editum.org/Que-Es-Un-Servidor-Web-p-401.html>. Fecha de consulta: julio de 2010.
37. <http://www.error500.net/software/infraestructura-como-servicio-iaas-cloud-computing>. Fecha de consulta: mayo de 2010.

38. http://www.es.masterbase.com/img/MasterBase_cloud_computing.pdf.
Fecha de consulta: abril de 2010.
39. http://www.financiatech-mag.com/_docum/138_DocumentoC_2.pdf.
Fecha de consulta: julio de 2010.
40. <http://www.infor.es/servicios/managed-services/saas/>. Fecha de consulta:
abril de 2010.
41. <http://www.informatica-hoy.com.ar/seguridad-informatica/Criptografia.php>
Fecha de consulta: agosto de 2010.
42. <http://www.ingenierosoftware.com/analisisydiseno/casosdeuso.php>.
Fecha de consulta: junio de 2010.
43. <http://www.mailxmail.com/curso-c-programacion/clases-objetos-encapsulacion>. Fecha de consulta: julio de 2010.
44. http://www.mercadeo.com/63_saas.htm. Fecha de consulta: mayo de 2010.
45. <http://www.microsoft.com/latam/technet/windowsserver/longhorn/evaluate/whitepaper.mspx>. Fecha de consulta: julio de 2010.
46. <http://www.mpsnet.com.mx/servicios/equipamiento/firewalls/ventajas>.
Fecha de consulta: julio de 2010.
47. <http://www.nubeblog.com/2008/10/15/saas-iaas-y-paas-las-tres-clases-de-cloud-computing>. Fecha de consulta: mayo de 2010.

48. <http://www.pc-actual.com/Actualidad/Noticias/Software-como-servicio-SaaS>. Fecha de consulta: mayo de 2010.
49. <http://www.sahw.com/wp/archivos/2008/10/12/seguridad-en-redes-de-almacenamiento-san-basadas-en-canal-de-fibra/>. Fecha de consulta: julio de 2010.
50. <http://www.slideshare.net/Decimo/arquitectura-3-capas>. Fecha de consulta: agosto de 2010.
51. <http://www.smdata.com/definicionsan.htm>. Fecha de consulta: agosto de 2010.
52. <http://www.starwindsoftware.de/san-for-windows-server-clusters>. Fecha de consulta: agosto de 2010.
53. <http://www.starwindsoftware.com/microsoft-sql-server>. Fecha de consulta: julio de 2010.
54. http://www.taringa.net/posts/downloads/3583964/SQL-Server-2008-Enterprise-*Full-Multilenguaje*.html. Fecha de consulta: julio de 2010.
55. <http://www.taringa.net/posts/info/2688390/Apache-vs-IIs-%C2%BFcual-elegir.html>. Fecha de consulta: julio de 2010.
56. http://www.uazuay.edu.ec/estudios/sistemas/teleproceso/apuntes_1/capa_presentacion.htm. Fecha de consulta: agosto de 2010.

57. <http://www.webtaller.com/maletin/articulos/que-es-certificado-seguridad-ssl.php>. Fecha de consulta: agosto de 2010.
58. <http://www.willydev.net/descargas/prev/IntroArq.pdf>. Fecha de consulta: julio de 2010.
59. <http://es.wikipedia.org/wiki/Cliente-servidor>. Fecha de consulta: julio de 2010.
60. http://es.wikipedia.org/wiki/Microsoft_SQL_Server. Fecha de consulta: julio de 2010.