



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERA EN SISTEMAS COMPUTACIONALES**

TEMA:

**“DESARROLLO DE UNA APLICACIÓN WEB UTILIZANDO
HERRAMIENTAS RIA CON TECNOLOGÍA ASP.NET. PARA LA
GESTIÓN DE VENTAS Y COMISIONES EN LA EMPRESA
REDPRIMCELL CIA LTDA”**

AUTORA: VERÓNICA JIMENA USIÑA TIRIRA

DIRECTOR: ING. MARCO PUSDÁ

IBARRA – ECUADOR

2016



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN

A FAVOR DE LA UIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

La UNIVERSIDAD TÉCNICA DEL NORTE dentro del proyecto Repositorio Digital Institucional determina la necesidad de disponer los textos completos de forma digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO	
CÉDULA DE IDENTIDAD:	040134749-7
APELLIDOS Y NOMBRES:	USIÑA TIRIRA VERONICA JIMENA
DIRECCIÓN:	SAN JUAN BOSCO 2-80 Y JOSE ESPINOZA DE LOS MONTEROS – CIUDADELA MUNICIPAL YUYUCOCHA
EMAIL:	vjut_83@hotmail.com
TELÉFONO FIJO:	TELÉFONO MÓVIL: 0997223104
DATOS DE LA OBRA	
TÍTULO:	“DESARROLLO DE UNA APLICACIÓN WEB UTILIZANDO HERRAMIENTAS RIA CON TECNOLOGÍA ASP.NET. PARA LA GESTIÓN DE VENTAS Y COMISIONES EN LA EMPRESA REDPRIMCELL CIA LTDA”
AUTORA:	USIÑA TIRIRA VERONICA JIMENA
FECHA:	ENERO DEL 2016
PROGRAMA:	<input type="checkbox"/> PREGRADO <input type="checkbox"/> POSTGRADO
TÍTULO POR EL QUE OPTA:	INGENIERA EN SISTEMAS COMPUTACIONALES
DIRECTOR:	ING. MARCO PUSDÁ

2. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, USIÑA TIRIRA VERÓNICA JIMENA, con cédula de ciudadanía Nro. 040134749-7, en calidad de autor y titular de los derechos patrimoniales de la obra o trabajo de grado descrito anteriormente, hago entrega del ejemplar respectivo en forma digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y el uso del archivo digital en la biblioteca de la universidad con fines académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión, en concordancia con la Ley de Educación Superior Artículo 144.



Firma

Nombre: Verónica Jimena Usiña Tirira

Cédula: 040134749-7

Ibarra, Enero del 2016



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CONSTANCIA

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Firma

Nombre: Verónica Jimena Usiña Tirira

Cédula: 040134749-7

Ibarra, Enero del 2016



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

**CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A FAVOR
DE LA UNIVERSIDAD TÉCNICA DEL NORTE**

Yo, Usiña Tirira Verónica Jimena, con cédula de ciudadanía Nro. 040134749-7, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la ley de propiedad intelectual del Ecuador, artículo 4, 5 y 6, en calidad de autor del trabajo de grado denominado: **“DESARROLLO DE UNA APLICACIÓN WEB UTILIZANDO HERRAMIENTAS RIA CON TECNOLOGÍA ASP.NET. PARA LA GESTIÓN DE VENTAS Y COMISIONES EN LA EMPRESA REDPRIMCELL CIA LTDA”**, que ha sido desarrollada para optar por el título de Ingeniería en Sistemas Computacionales, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En mi condición de autor me reservo los derechos morales de la obra antes mencionada, aclarando que el trabajo aquí descrito es de mi autoría y que no ha sido previamente presentado para ningún grado o calificación profesional.

En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la biblioteca de la Universidad Técnica del Norte.

Firma

Nombre: Verónica Jimena Usiña Tirira

Cédula: 040134749-7

Ibarra, Enero del 2016



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN

Certifico que la Tesis "DESARROLLO DE UNA APLICACIÓN WEB UTILIZANDO HERRAMIENTAS RIA CON TECNOLOGÍA ASP.NET. PARA LA GESTIÓN DE VENTAS Y COMISIONES EN LA EMPRESA REDPRIMCELL CIA LTDA", ha sido realizada en su totalidad por la señorita: Usiña Tirira Verónica Jimena, portador de la cédula de Ciudadanía Nro. 040134749-7.

A handwritten signature in blue ink, appearing to read "Marco Pusdá", is written over a horizontal line.

Ing. Marco Pusdá

DIRECTOR DE TESIS



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

DEDICATORIA

A Dios, al Divino Niño Jesús y nuestra Madre Santísima, por las bendiciones y fortalezas para la culminación de este trabajo.

A mis Queridos PADRES quienes con su amor, ejemplo y comprensión han hecho de mí una persona con valores para poder alcanzar esta meta. Mi triunfo es el de ustedes, ¡los amo!

A mi adorada hija Franciny Nazarena, quien da sentido a mi vida y me presto el tiempo que le pertenecía durante mis estudios, siempre me motivaron sus palabras “Ya quiero que seas Ingeniera” ¡Gracias, mi Niña Bella!

A mis Hermanos porque siempre he contado con ellos para todo, gracias por el apoyo amistad y la confianza que siempre nos hemos tenido.

A Juan Carlos que nunca dejó de creer en mí e hizo que mis prioridades sean las suyas, gracias por tu apoyo incondicional.

A los que nunca dudaron que lograría este triunfo mis Abuelitos, soy tan bendecida que aun puedo disfrutar de su cariño.

¡Gracias!

VERONICA JIMENA



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

AGRADECIMIENTO

A Dios.

Por haberme permitido llegar hasta este punto y haberme dado salud para lograr mis objetivos, además de su infinita bondad y amor.

A mi madre Elsa Marina.

Por haberme apoyado en todo momento, por sus consejos, sus valores, por la motivación constante que me ha permitido ser una persona de bien, pero más que nada, por su amor y por cuidar de mí hija mientras realizaba mis estudios.

A mi padre Iván Guillermo.

Por los ejemplos de perseverancia y constancia que lo caracterizan y que me ha infundado siempre, por el valor mostrado para salir adelante y por su amor.

A la Universidad Técnica del Norte.

En la Facultad de Ciencias Aplicadas, y en ella a los distinguidos Docentes quienes con su profesionalismo y ética puesto manifiesto en las aulas enrumban a cada uno de los que acudimos con sus conocimientos que nos servirán para ser útiles a la sociedad.

Quisiera expresar mi agradecimiento al Director y tutor de esta tesis al Ing. Marco Pusdá e Ing. Jorge Caraguay deseo reconocer su trabajo y dedicación permanente y continua al presente trabajo de investigación, así como sus sugerencias y observaciones, siempre inteligentes y oportunas.

A la Empresa REDPRIMCEL Cía. Ltda.

Por haberme brindado la oportunidad para desarrollar mi trabajo de grado, a la Sra. Yesenia Azuero Gerente General, a los compañeros de trabajos que son excelentes personas con quienes comparto día a día, nuevas experiencias y conocimientos, gracias por la colaboración prestada en todo momento para el desarrollo de este trabajo.

A Todos aquellos familiares y amigos que no recordé al momento de escribir esto. Ustedes saben quiénes son.

VERONICA JIMENA

RESUMEN

La Cultura informática se ha hecho presente en todo ámbito como una herramienta que apoya la evolución de toda organización hacia un enfoque de practicidad y mejoras en el procesamiento de la información, a través de la Teoría General de Sistemas.

Es por esto que se requiere de adoptar una postura teórica que lo guíe en su práctica profesional, comprendiendo, aplicando y sintetizando los principios generales de sistemas así como una metodología de análisis que permita implementar procesos de mejora en las organizaciones. Todo ello contribuyendo a la formación integral promoviendo el desarrollo del intelecto y las diferentes alternativas donde pueda desarrollarse el futuro egresado.

Por eso es que se he decidido emprender este proyecto para la Empresa REDPRIMCEL CIA LTDA. Con el objetivo de sintetizar y automatizar procesos manuales para así poder brindar a la empresa información precisa y rápida, pudiendo así situarla en el innumerable lista de empresas que se encuentran a la vanguardia de brindar servicios de calidad, en este caso hacia el cliente interno, sus empleados, que podrán tener la información que requieran y así tener el control de su información y poder saber hacia dónde se dirige la empresa.

SUMMARY

The Culture computing has been present in all areas as a tool that supports the development of an organization to a practical approach and improvements in information processing, through the General Systems Theory.

This is why it is necessary to adopt a theoretical position to guide you in your practice, understanding, applying and synthesizing general principles and systems analysis methodology that allows implementing process improvement in organizations. All this contributes to promoting the integral development of intellect and the alternatives which can develop the future graduates.

That's why you've decided to take this project to the Company REDPRIMCEL CIA LTDA. In order to synthesize and automate manual processes in order to provide an organization with quick and accurate information, thus being able to place it at the innumerable list of companies that are at the forefront of providing quality services, in this case to the internal customer their employees, which may take the information they require and thus have control of your information and to find out where the company is headed.

INDICE DE CONTENIDOS

AUTORIZACIÓN DE USO Y PUBLICACIÓN	II
CONSTANCIA	IV
CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE	V
CERTIFICACIÓN.....	VI
DEDICATORIA	VII
AGRADECIMIENTO	VIII
RESUMEN.....	IX
SUMMARY	X
INDICE DE CONTENIDOS	XI
INDICE FIGURAS	XVI
INDICE DE TABLAS.....	XVIII
CAPÍTULO I	1
1 INTRODUCCIÓN.....	1
1.1 ESTRATEGIAS DE MERCADEO	4
1.1.1 ESTRATEGIA DE VENTA DIRECTA	4
1.1.2 ESTRATEGIA INDIRECTA (CON DISTRIBUIDORES)	5
1.2 SITUACION ACTUAL EMPRESARIAL	6
1.2.1 MISIÓN	6
1.2.2 VISIÓN.....	6
1.2.3 VALORES CENTRALES.....	6
1.2.4 VALORES GENERALES	6
1.2.5 FODA.....	7
1.3 ESTRUCTURA ORGANIZACIONAL	8
1.3.1 ORGANICO FUNCIONAL.....	8
1.3.2 PROCESO DE COMPRAS	9
1.3.3 PROCESO DE VENTAS.....	10

1.3.4 PROCESO DE COMPRA DE PINES Y TARJETAS	11
1.4 OBJETIVOS.....	12
1.4.1 OBJETIVO GENERAL	12
1.4.2 OBJETIVOS ESPECÍFICOS.....	12
1.5 PROBLEMÁTICA DE LA EMPRESA.....	12
1.5.1 SITUACIÓN ACTUAL.....	12
1.5.2 DEFINICIÓN DEL PROBLEMA	13
1.5.3 PROSPECTIVA DEL PROBLEMA	13
1.6 BENEFICIOS DE LA APLICACIÓN WEB.....	13
CAPÍTULO II	16
2 MARCO TEÓRICO.....	16
2.1 BASE DE DATOS SQL SERVER 2008.....	16
2.1.1 INTRODUCCIÓN A SQL SERVER 2008	17
2.1.2 HERRAMIENTAS DE SQL SERVER 2008.....	19
2.1.2.1 LIBROS EN PANTALLA.....	19
2.1.2.2 SQL SERVER CONFIGURATION MANAGER	19
2.1.2.3 EL SQL SERVER MANAGEMENT STUDIO	19
2.2 PLATAFORMA RIA	19
2.2.1 INTRODUCCIÓN	19
2.2.2 QUÉ SON LOS RIA SERVICES?	20
2.2.3 ARQUITECTURA	22
2.2.4 ¿CUÁL ES EL OBJETIVO DE RIA SERVICES?.....	23
2.2.5 VENTAJAS DE LAS RIA'S.....	24
2.3 MICROSOFT SILVERLIGHT	25
CAPÍTULO III	28
3 DISEÑO Y ARQUITECTURA DEL SISTEMA	28
3.1 INSTALACIÓN Y CONFIGURACIÓN DE MICROSOFT SILVERLIGHT	28
3.1.1 BLOQUES DE CONSTRUCCIÓN	28
3.1.1.1 DEPENDENCYOBJECT	29

3.1.1.2	THREADING Y LA INTERFAZ DE USUARIO.....	30
3.1.1.3	UIELEMENT	30
3.1.1.4	LA CLASE CONTROL	35
3.2	ACCESO A DATOS	36
3.2.1	ENLACE CON DATOS.....	36
3.3	PATRÓN MVVM.....	39
3.3.1	EL MODELO.....	39
3.3.2	LA VISTA.....	39
3.3.3	EL MODELO DE VISTA (NUESTRO CONTROLADOR O PRESENTADOR)	40
3.3.4	LA VISTA Y EL MODELO DE VISTA	41
3.3.5	EL MODELO DE VISTA Y EL MODELO	41
3.3.6	INFRAESTRUCTURA BÁSICA PARA MVVM	41
3.4	WCF RIA SERVICES	42
3.4.1	WCF INTEGRACIÓN	43
3.4.1.1	SERVICIOS DE DOMINIO.....	43
3.4.1.2	SERVICIOS DE DOMINIO Y SUS FUENTES DE DATOS.....	44
3.4.2	ASEGRAR UNA SOLUCIÓN DE RIA SERVICES	44
3.4.2.1	SEGURIDAD DE LOS SERVICIOS RIA WCF	44
3.4.2.2	LISTA DE VERIFICACIÓN DE SEGURIDAD.....	45
3.4.3	HERRAMIENTAS Y DOCUMENTACIÓN.....	47
3.5	PROGRAMACION EXTREMA XP	47
3.5.1	HISTORIA.....	47
3.5.2	INTRODUCCION	48
3.5.3	¿QUÉ ES PROGRAMACIÓN EXTREMA O XP?.....	48
3.5.4	OBJETIVOS.	48
3.5.5	CONTEXTO XP.....	49
3.5.6	CARACTERÍSTICAS XP.....	49
3.5.7	VALORES XP	49
3.5.8	EL ESTILO XP.....	50

3.5.9 PRÁCTICAS BÁSICAS DE LA PROGRAMACIÓN EXTREMA	50
3.5.10 VENTAJAS Y DESVENTAJAS DE EXTREME PROGRAMMING.....	52
3.5.11 LOS ROLES EN XP	52
3.5.11.1 CLIENTE	53
3.5.11.2 DESARROLLADOR (PROGRAMADOR)	53
3.5.11.3 GESTOR (MANAGER)	54
3.5.11.4 CONSULTOR.....	54
3.5.11.5 ENCARGADO DEL SEGUIMIENTO (TRACKER)	54
3.5.11.6 ENCARGADO DE PRUEBAS (TESTER).....	55
3.5.11.7 ENTRENADOR (COACH).....	55
CAPÍTULO IV	56
4 DESARROLLO DEL APLICATIVO	56
4.1 PLANIFICACIÓN.....	56
4.1.1 HISTORIAS DE USUARIO.....	56
4.1.1.1 IDENTIFICACIÓN DE HISTORIAS DE USUARIO.....	57
4.1.2 PLAN DE ENTREGAS	64
4.1.3 VELOCIDAD DEL EQUIPO DE DESARROLLO	66
4.1.4 ITERACIONES POR CADA ENTREGA	66
4.2 DISEÑO.....	66
4.2.1 METÁFORA.....	66
4.2.2 CRC. TARJETAS DE RESPONSABILIDAD COLABORACIÓN.....	67
4.3 DESARROLLO.....	68
4.3.1 DISPONIBILIDAD DEL CLIENTE	68
4.3.1.1. DIAGRAMA ENTIDAD-RELACIÓN	68
4.3.2 UNIDAD DE PRUEBAS	69
4.3.2.1 ESCENARIO DE PRUEBA DE “INGRESO DE PLANES TELEFÓNICOS”	69
4.3.2.2 ESCENARIO DE PRUEBA DE “INGRESO DE SOLICITUD DE PLAN”	70
4.3.2.3 ESCENARIO DE PRUEBA DE “APROBACIÓN SOLICITUD”.....	71

4.3.2.4 ESCENARIO DE PRUEBA DE “DESPLIEGUE DE INFORMACIÓN DEL SUScriptor”	72
4.3.2.5 ESCENARIO DE PRUEBA DE “DETERMINACIÓN DE VALORES DE VENTAS”	73
4.3.3 TRABAJO EN PAREJAS	73
4.3.4 PROPIEDAD COLECTIVA DEL CÓDIGO.....	73
4.4 ENTREGA	74
4.4.1 IMPLANTACIÓN DEL SISTEMA	74
4.4.1.1 PLAN DE IMPLANTACIÓN.....	74
4.4.1.2 EQUIPO DE TRABAJO.....	75
4.4.1.3 CAPACITACIÓN PARA LA IMPLANTACIÓN	75
4.4.2 REQUISITOS PARA LA IMPLANTACIÓN	75
CAPÍTULO V	77
5 CONCLUSIONES Y RECOMENDACIONES	77
5.1 CONCLUSIONES.....	77
5.2 RECOMENDACIONES	78
5.3 BIBLIOGRAFÍA.....	80
5.3.1 LIBROS.....	80
5.4 LINKOGRAFIA	81
5.4.1 ARTÍCULOS.....	82
5.5 ANEXOS.....	83
5.5.1 INTEPROYECTO – ANEXO A.....	83
5.5.2 METRICA V3 – ANEXO B.....	83

INDICE FIGURAS

FIGURA 1.1: Abonados telefonía móvil.....	1
FIGURA 1.2: Mercado Operadora Claro	2
FIGURA 1.3: Crecimiento Abonados Telefonía Celular.....	3
FIGURA 1.4: Orgánico Funcional.....	8
FIGURA 1.5: Proceso de Compras	9
FIGURA 1.6: Proceso Ventas	10
FIGURA 1.7: Proceso de compras y pines	11
FIGURA 1.8: Beneficios de la web	13
FIGURA 2.1: Aplicaciones de Sql Server 2008.....	16
FIGURA 2.2: Arquitectura Report Services.....	18
FIGURA 2.3: Aplicaciones RIA	21
FIGURA 2.4: Componentes Aplicaciones RIA	21
FIGURA 2.5: Arquitectura de una aplicación RIA.....	22
FIGURA 2.6: Procesamiento de datos en una aplicación RIA.....	23
FIGURA 2.7: Arquitectura Orígenes de Datos de Dominio	24
FIGURA 3. 1: Clase Jerárquica de la interfaz de usuario en Silverlight.....	28
FIGURA 3. 2: Una visión conceptual de enlace de datos. La fuente posee los datos, la función objetivo (exposiciones, ediciones, y así sucesivamente) de los datos	37
FIGURA 3. 3: Una visión conceptual de la otrora enlace a un origen de datos. El valor es un principio leer desde la fuente y nunca se volvió a actualizar.	37
FIGURA 3. 4: Una visión conceptual de OneWay enlace a un origen de datos. El valor se actualiza cada vez que cambia la fuente, pero los cambios en el valor en el control de destino no lo hacen de nuevo a la fuente.	38
FIGURA 3. 5: Una visión conceptual de TwoWay enlace a un origen de datos. El control de destino refleja cambios en el origen y la fuente se actualiza con los cambios realizados en el destino.....	38
FIGURA 3. 6: Detalle del modelo de vista.....	40
FIGURA 3. 7: Detalle del modelo de MVVM	42
FIGURA 3. 8.: Se muestra una versión simplificada de una aplicación de n niveles.....	43

FIGURA 3.9: Roles que existen en un proyecto XP	53
FIGURA 4.1: Tarjeta CRC para Solicitud de Plan.....	67
FIGURA 4.2: Tarjeta CRC para Solicitud de Plan.....	67
FIGURA 4.3: Diagrama Entidad - Relación.....	68
FIGURA 4.4: Diagrama de clases.....	68

INDICE DE TABLAS

TABLA 3. 1: Métodos de la clase System.Windows.DependencyObject	29
TABLA 3. 2: Métodos de la clase System.Windows.UIElement.....	31
TABLA 3. 3: Propiedades de la clase System.Windows.UIElement.....	32
TABLA 3. 4: Eventos de la clase System.Windows.UIElement.....	33
TABLA 3. 5: Propiedades de la clase System.Windows.Controls.Control.....	35
TABLA 4-1: Historia de Usuario "Ingreso del Usuario al Sistema"	58
TABLA 4-2: Historia de Usuario "Creación de Usuarios"	58
TABLA 4-3: Historia de Usuario "Eliminación de Usuarios"	58
TABLA 4-4: Historia de Usuario "Modificación de Usuarios"	58
TABLA 4-5: Historia de Usuario "Ingreso de Planes Telefónicos"	59
TABLA 4-6: Historia de Usuario "Ingreso de Solicitud de Plan"	59
TABLA 4-7: Historia de Usuario "Aprobación Solicitud"	59
TABLA 4-8: Historia de Usuario "Despliegue de Información del Suscriptor"	59
TABLA 4-9: Historia de Usuario "Consulta de Información del Plan Telefónico".....	60
TABLA 4-10: Historia de Usuario "Despliegue de la Información Obtenida de la Activación del Plan".....	60
TABLA 4-11: Historia de Usuario "Despliegue del Diagnóstico en un DBGRID".....	60
TABLA 4-12: Historia de Usuario "Determinación de Valores de Ventas"	61
TABLA 4-13: Historia de Usuario "Determinación de Comisiones".....	61
TABLA 4-14: Historia de Usuario "Creación de Equipos"	62
TABLA 4-15: Historia de Usuario "Ingreso de Equipos"	62
TABLA 4-16: Historia de Usuario "Control Stock Equipos"	62
TABLA 4-17: Historia de Usuario "Reporte de Ventas por Vendedor".....	63
TABLA 4-18: Historia de Usuario "Reportes por Fechas"	63
TABLA 4-19: Historia de Usuario " Reportes de Comisión de Ventas".....	63
TABLA 4-20: Historia de Usuario "Reporte Equipos Vendidos"	63
TABLA 4-21: Prioridad Historias de usuario	64
TABLA 4-22: Fechas de entrega historias de usuario	65

TABLA 4-23: Prueba de Aceptación para el escenario "Ingreso de Planes Telefónicos"	69
TABLA 4-24: Prueba de Aceptación para el escenario "Ingreso de Solicitud de Plan" .	70
TABLA 4-25: Prueba de Aceptación para el escenario "Ingreso de Planes Telefónicos"	71
TABLA 4-26: Prueba de Aceptación para el escenario "Despliegue de Información del Suscriptor"	72
TABLA 4-27: Prueba de Aceptación para el escenario "Determinación de valores de Ventas"	73
TABLA 4-28: Equipo de Trabajo para la Implantación del Sistema	75
TABLA 4-29: Requisitos del Servidor de Base de Datos	76
TABLA 4-30: Requisitos del Cliente	76

CAPÍTULO I

1 INTRODUCCIÓN

En un mundo globalizado, donde no existen fronteras y la competencia crece cada día más, tanto en número como en mejores servicios, procesos, capacidad y sobre todo calidad, es que se hace evidente, para una empresa sea pequeña, mediana o grande, el mejorar sus procesos y controles para ofrecer servicios y productos de calidad, que le permitirá no solo mejores rendimientos, sino el garantizar su permanencia en el mercado.

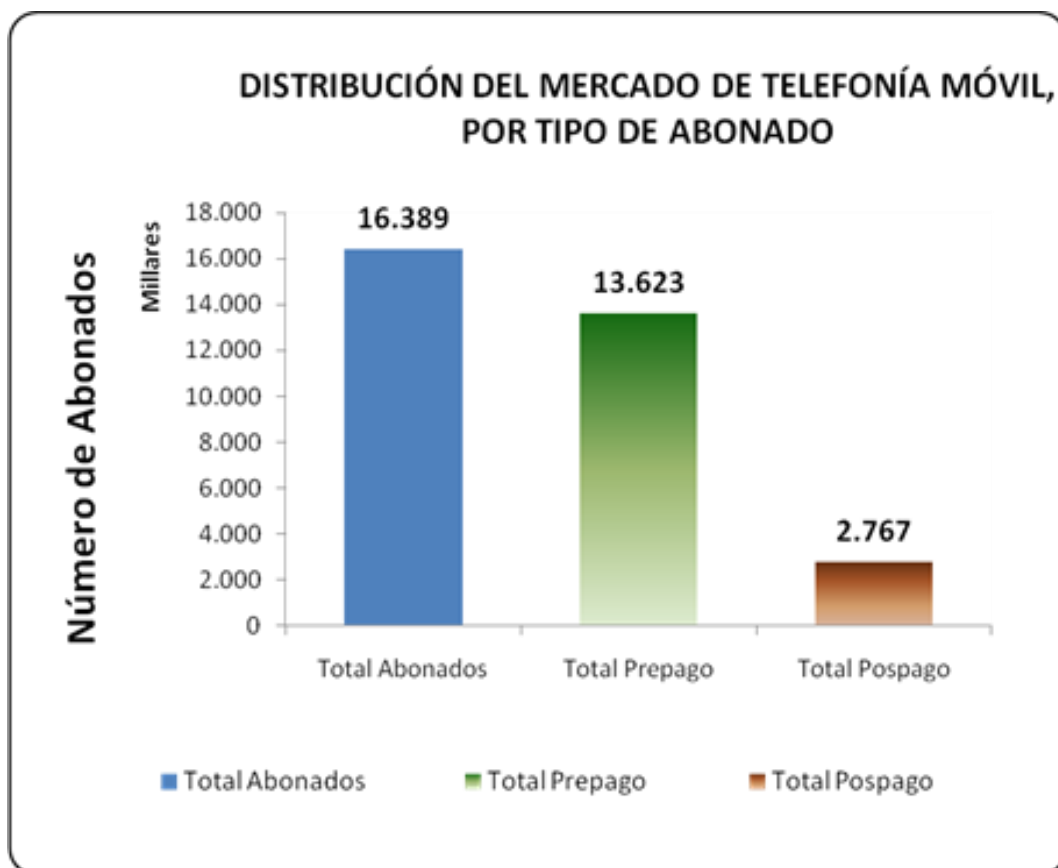


FIGURA. 1.1: Abonados telefonía móvil

Fuente: <http://www.supertel.gob.ec/index.php/20120326436/principales/estadisticas-de-telefonía-móvil-hasta-febrero-de-2012.supertel>

CLARO es la marca que utiliza América Móvil en la mayoría de los países donde está presente. América Móvil tiene operaciones en 18 países del continente americano y el Caribe. CLARO es una marca líder en 14 países de América Latina (Argentina, Brasil, Chile, El Salvador, Guatemala, Honduras, Jamaica, Nicaragua, Panamá, Paraguay, Puerto Rico, Perú, República Dominicana y Uruguay) a los que ahora se suma Ecuador con más de 10 millones de usuarios.

En Ecuador Claro está operando desde Abril del 2011, en reemplazo de PORTA que estaba operando en el Ecuador desde 1993, es la empresa de telefonía celular líder con más de 10 millones de usuarios y con cobertura en todas las provincias del país¹.

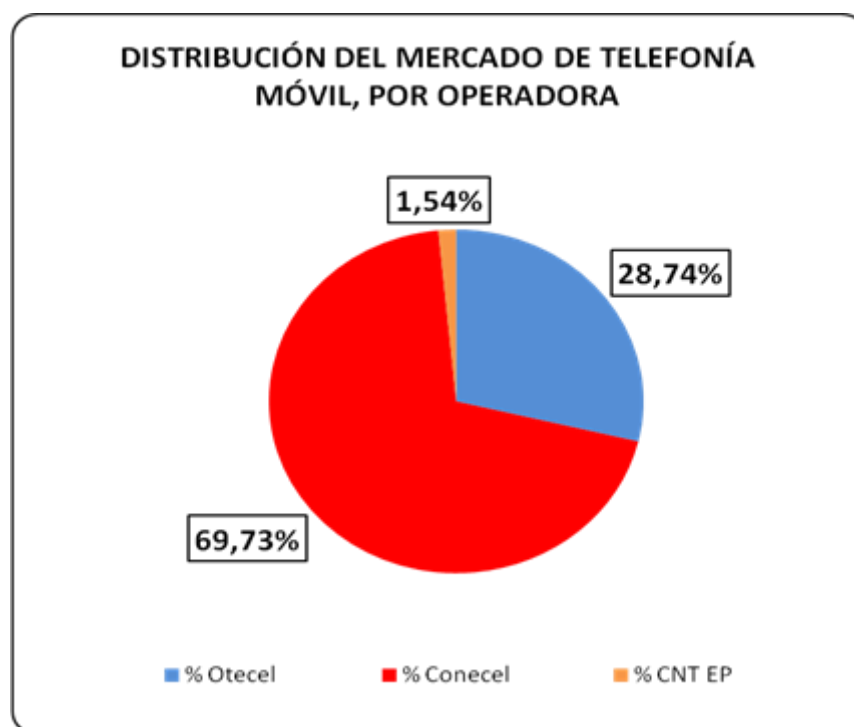


FIGURA 1.2: Mercado Operadora Claro

Fuente: [<http://www.supertel.gob.ec/index.php/20120326436/principales/estadisticas-de-telefonía-movil-hasta-febrero-de-2012.supertel>]

¹ <http://www.hoy.com.ec/noticias-ecuador/desde-abril-la-marca-porta-pasara-a-llamarse-claro-459266.html>

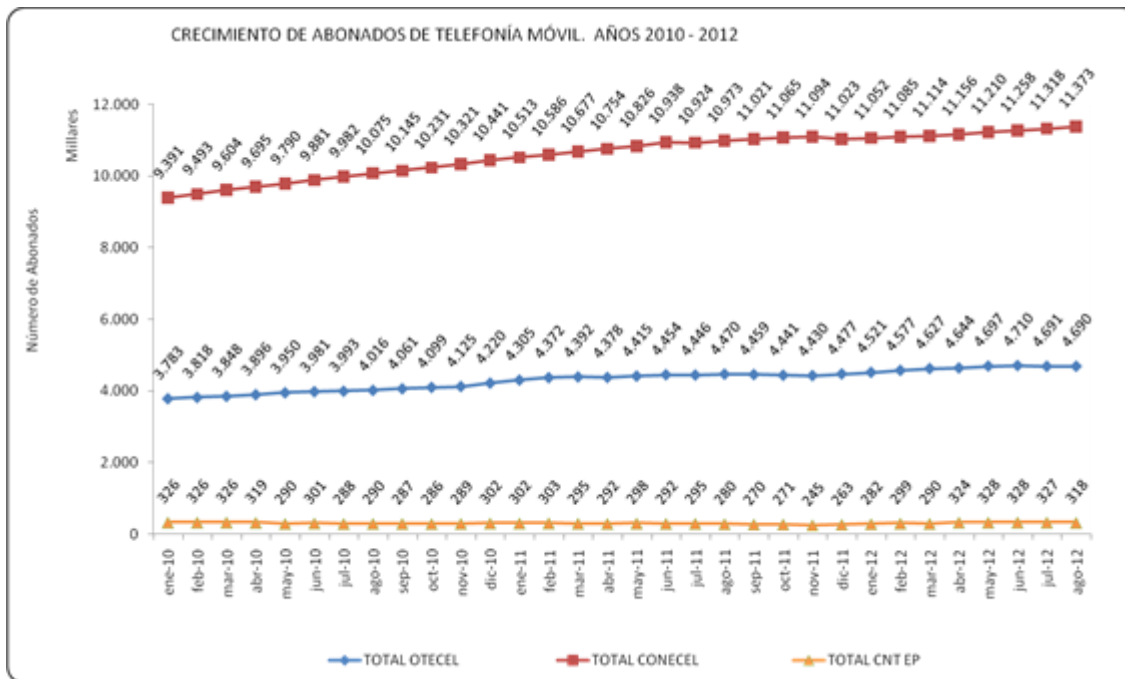


FIGURA 1.3: Crecimiento Abonados Telefonía Celular

Fuente: [<http://www.supertel.gob.ec/index.php/20120326436/principales/estadisticas-de-telefonía-movil-hasta-febrero-de-2012.supertel>]

Redprimcel Cía. Ltda. Distribuidor Máster de Claro, es una empresa de telefonía móvil que comercializa servicios de comunicación, fue creada el 1 de Junio del 2010, incluyendo un total de 15 Empleados, 5 Administrativos y 10 Asesores Comerciales (vendedores).

Cuentan con los siguientes departamentos:

- Operaciones.
- Contabilidad.
- Ventas
- Atención al Cliente.
- Recursos Humanos

Redprimcel Cía. Ltda., comercializa Planes individuales, Internet Banda Ancha Móvil, Bases de recargas electrónicas, tarjetas Prepago para las ciudades de Ibarra, Otavalo, Cayambe, Cotacachi, Atuntaqui, Tulcán y demás ciudades del norte del país.

1.1 ESTRATEGIAS DE MERCADEO

El mundo empresarial nos muestra lo variado de las estrategias comerciales. Algunas empresas eligen vender directamente a los consumidores mientras que otras lo hacen a través de una red de distribución.

Pero, ¿cuál es la racionalidad que debe guiar la elección? ¿Cuándo conviene cada estrategia?

La respuesta no es evidente ni existen fórmulas matemáticas que determinen, a través de algún índice, cuál es la mejor estrategia.

1.1.1 ESTRATEGIA DE VENTA DIRECTA

En general, la estrategia directa ofrece una “atención al cliente”² más dedicada y, principalmente, una mayor facilidad para hacer negocios "taylor made", es decir, adaptados a las necesidades del cliente.

No obstante, esta estrategia también implica una mayor utilización de recursos. La estructura comercial de la empresa debe incluir un plantel de vendedores, servicio de atención al cliente y personal de administración, finanzas y logística.

Así, una gran desventaja de esta estrategia radica en una mayor cantidad de costos directos asociados al proceso de venta.

Ahora bien, esta desventaja se agrava en mercados de productos masivos consumidos por una gran cantidad de clientes en un amplio rango geográfico.

En este caso, los desafíos administrativos y logísticos se multiplican y a la empresa se le complica atender adecuadamente a tantos clientes.

² <http://www.materiabiz.com/mbz/economiayfinanzas/nota.vsp?nid=35146>

1.1.2 ESTRATEGIA INDIRECTA (CON DISTRIBUIDORES)

La estrategia basada en un canal de distribución tiene un sinnúmero de variantes.

En el mercado actual se puede distinguir entre dos casos extremos: el modelo "dealers" (donde existe una menor relación entre fabricante y distribuidor) y el de franquicia (donde existe una fuerte dependencia del franquiciado respecto del fabricante). En el modelo dealers³, el vendedor tiene un compromiso de compra y obtiene descuentos en función de los volúmenes adquiridos.

La elección de una estrategia de venta indirecta se determina fundamentalmente por la diversidad de productos y la atomización del mercado (tanto por su cantidad de clientes como por su distribución geográfica).

En el libro "Canales de Distribución", Hugo Rodolfo Paz nos presenta las ventajas de este modelo:

- 1) Permite un rápido crecimiento con mínima inversión de capital
- 2) Permite acceder a nuevos mercados geográficos
- 3) Se asegura una clientela fija y estable
- 4) Aumenta la notoriedad de la marca a medida que se incorporan nuevos distribuidores
- 5) Potencia la imagen del producto o servicio
- 6) Permite distribuir mejor el presupuesto publicitario
- 7) Recibe ingresos por diversos conceptos: derechos de entrada, royalties, márgenes sobre productos, publicidad y otros.
- 8) Reduce los costos fijos.

³ Es una persona o entidad que actúa como principal, y no como agente, en la compra y venta de valores

1.2 SITUACION ACTUAL EMPRESARIAL

1.2.1 MISIÓN

Satisfacemos las necesidades de nuestros clientes, manteniendo relaciones duraderas, anticipando eficazmente las demandas del mercado, ofreciendo una variedad de productos de alta calidad, con excelencia en el servicio y responsabilidad social; laborando internamente con cualidades empresariales de profundo contenido humano y profesional.

1.2.2 VISIÓN

Convertirnos en empresa líder en el área comercial de las telecomunicaciones en los mercados local y regional, sustentados en competitividad funcionalidad y calidad

1.2.3 VALORES CENTRALES

- Equipo humano, capacitado para satisfacer eficientemente la demanda del mercado.
- Gestión administrativa eficiente.
- Comunicación interna efectiva.
- Control y manejo financiero, estricto constante y eficiente.

1.2.4 VALORES GENERALES

- Integridad
- Puntualidad
- Responsabilidad
- Esfuerzo
- Perseverancia
- Respeto a las normas
- Trabajo en equipo

1.2.5 FODA

FORTALEZAS	OPORTUNIDADES
<ul style="list-style-type: none"> ✓ Personal capacitado para el asesoramiento a los clientes ✓ Ubicación estratégica de la distribuidora ✓ Los servicios que presta son económicos ✓ Tener publicidad informativa y de conocimiento 	<ul style="list-style-type: none"> ✓ La demanda creciente de mercado en cuanto a telefonía móvil ✓ Aprovechar las fallas que deja la competencia en el mercado
DEBILIDADES	AMENAZAS
<ul style="list-style-type: none"> ✓ Ser distribuidora de una empresa nueva en el mercado ✓ La existencia en el mercado de otras distribuidoras de telefonía móvil 	<ul style="list-style-type: none"> ✓ El reconocimiento que tienen ciertas empresas reflejado por su acelerado crecimiento ✓ La continua oferta que tiene las empresas en busca de ganar demanda de los clientes ✓ Problemas externos no controlables: como la inflación, inestabilidad política y económica, entre otros ✓ Entrada de nuevas empresas al mercado

1.3 ESTRUCTURA ORGANIZACIONAL

1.3.1 ORGANICO FUNCIONAL

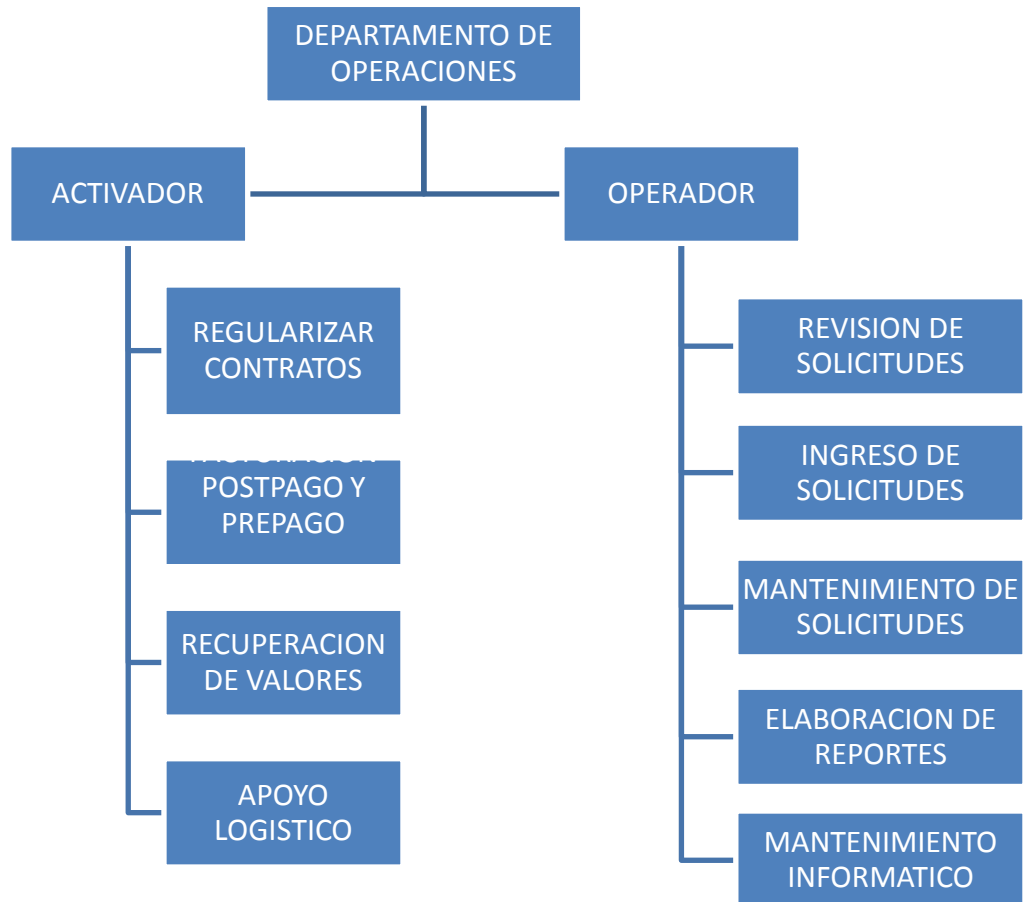


FIGURA 1.4: Orgánico Funcional

Fuente: Redprimcel Cía. Ltda.

1.3.2 PROCESO DE COMPRAS

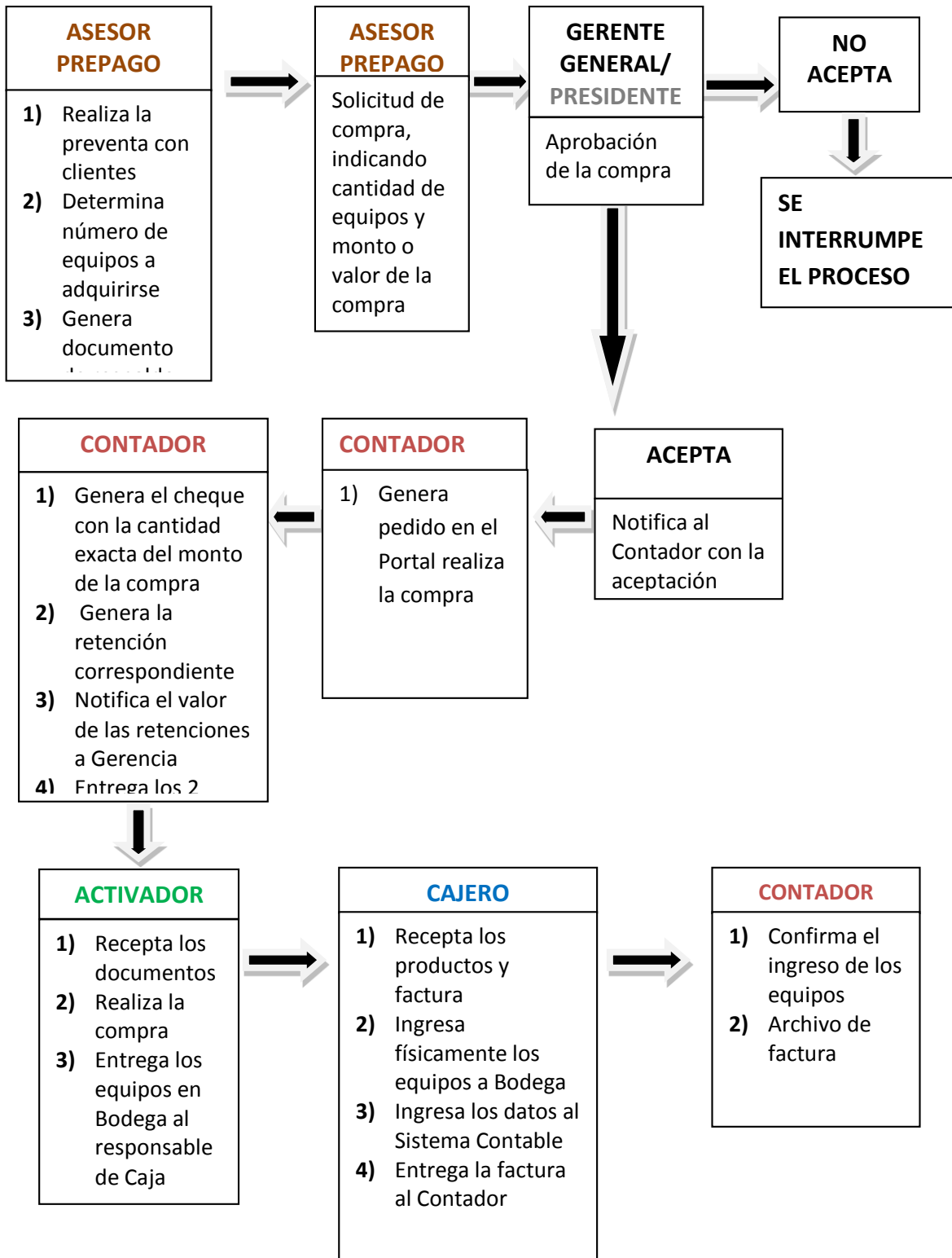


FIGURA 1.5: Proceso de Compras

Fuente: Redprimcel Cía. Ltda.

1.3.3 PROCESO DE VENTAS

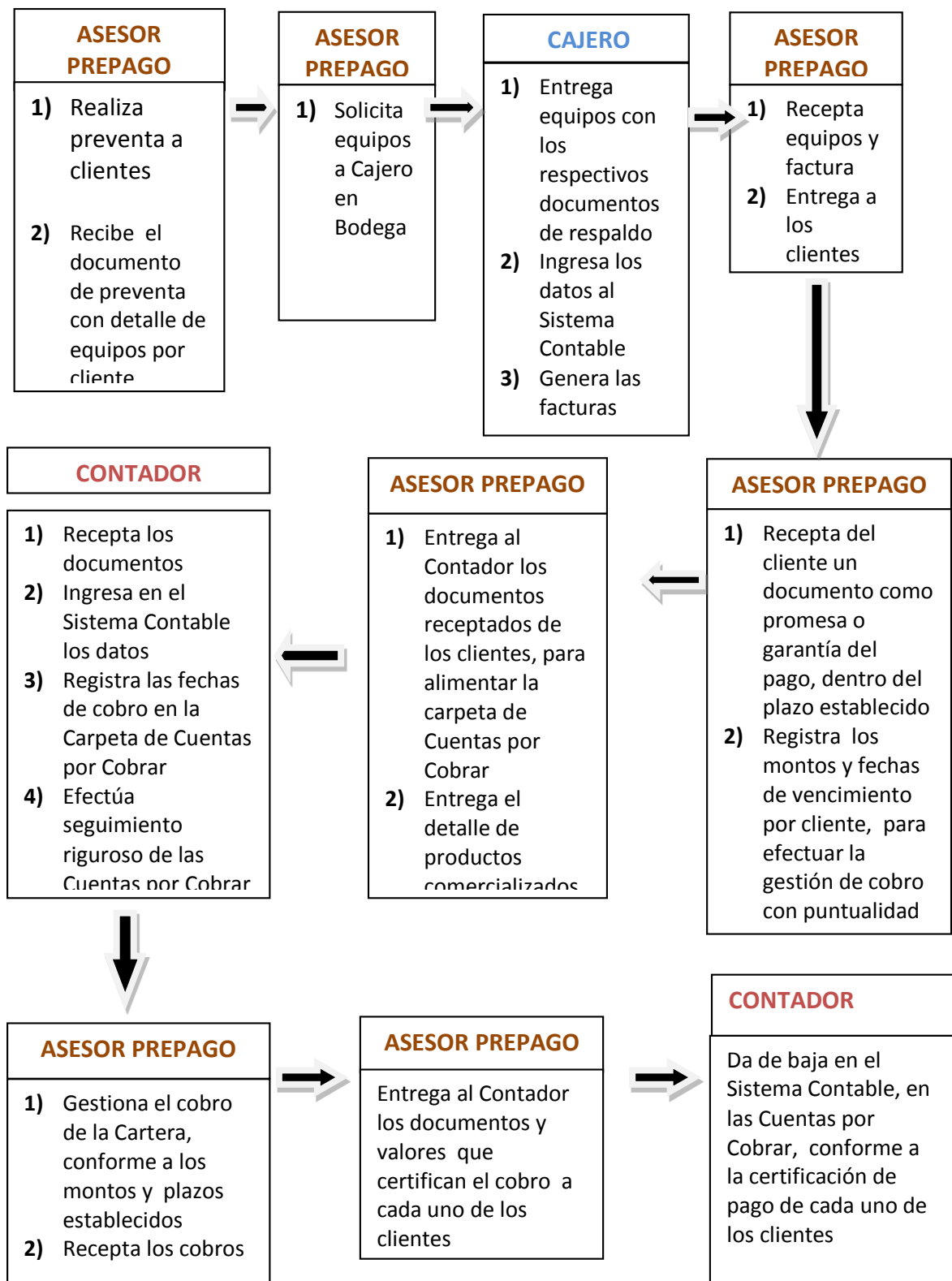


FIGURA 1.6: Proceso Ventas

Fuente: Redprimcel Cía. Ltda.

1.3.4 PROCESO DE COMPRA DE PINES Y TARJETAS

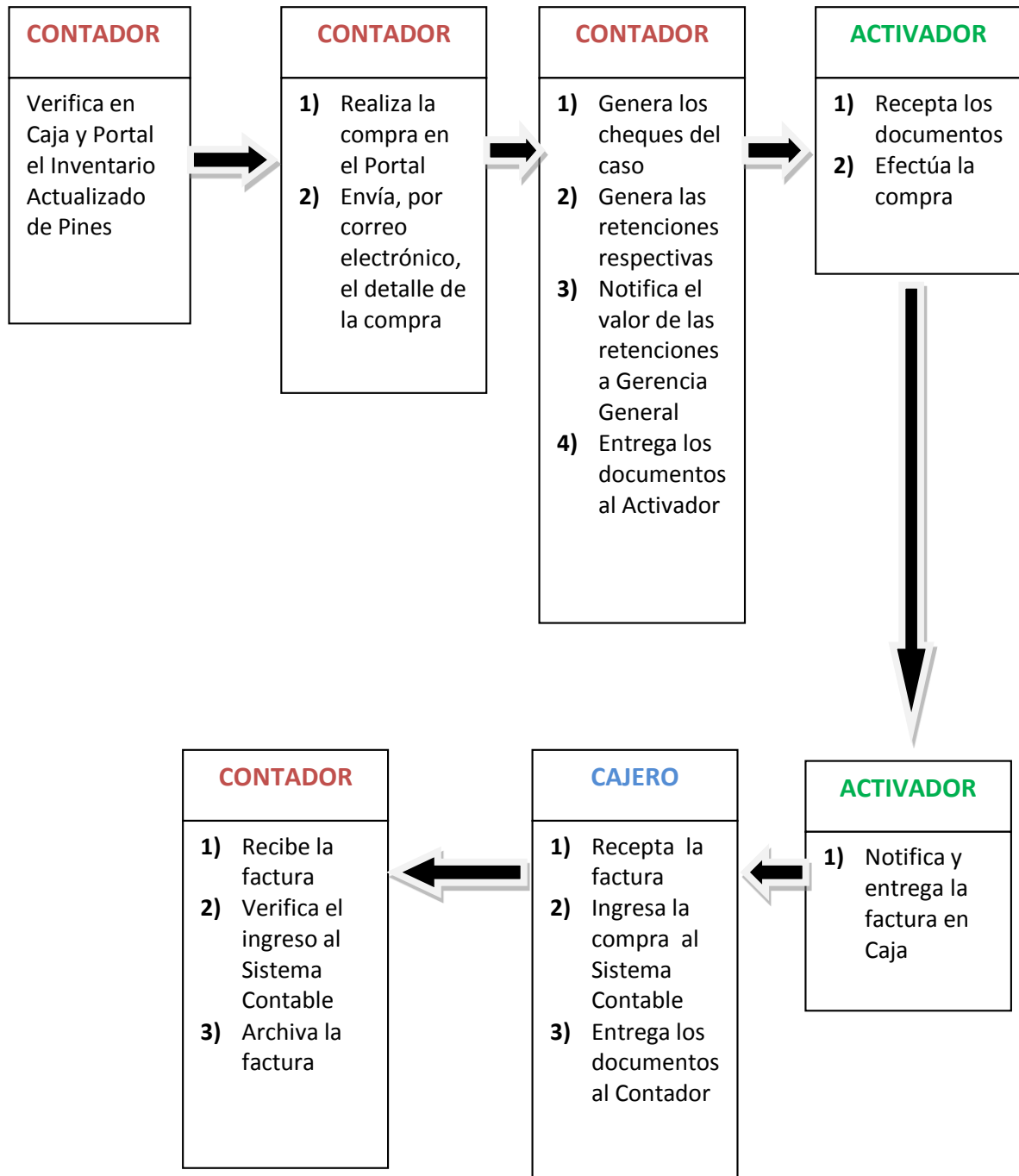


FIGURA 1.7: Proceso de compras y pines

Fuente: Redrimcel Cía. Ltda.

1.4 OBJETIVOS

1.4.1 OBJETIVO GENERAL

Implementar una Aplicación Web para la gestión de Ventas y Comisiones de la empresa Redprimcel Cía. Ltda., mediante el uso de herramientas RIA (Rich Internet Application).

1.4.2 OBJETIVOS ESPECÍFICOS.

- Documentar la información y los procesos generados al desarrollar el sistema mediante el uso de la metodología XP.
- Investigar el uso de la herramienta Microsoft Silverlight 5.0 para el desarrollo de la aplicación.
- Desarrollar la aplicación web para la gestión de ventas y comisiones de tal manera que sea amigable y fácil de utilizar.
- Realizar pruebas, validar e implementar la aplicación web para la empresa.

1.5 PROBLEMÁTICA DE LA EMPRESA

1.5.1 SITUACIÓN ACTUAL

Actualmente esta institución tiene la misión de alcanzar la satisfacción total de sus clientes, comerciando productos postpago, brindando atención personalizada , basada en principios de calidad humana y ética profesional, con el apoyo del Recurso Humano, por lo que ha incrementado su personal, hoy en día cuenta con 20 Empleados, 10 Administrativos y 20 Asesores comerciales, la Oficina Matriz se encuentra situada en la ciudad de Ibarra en las calles Bartolomé García 11-32 y Rafael Larrea, y cuatro sucursales en Otavalo, Lago Agrio, Tena y El Puyo. Para la gestión de ventas, la empresa cuenta con un sistema en Excel para registrar los clientes y ventas de la empresa y con esta información calcular las comisiones de los vendedores, causando inconformidad entre vendedores ya que este sistema es poco eficiente para satisfacer las necesidades de los mismos, además existe ineficiencia en la presentación de reportes.

1.5.2 DEFINICIÓN DEL PROBLEMA

La empresa Redprimcel Cía. Ltda., tiene dificultades en la sistematización de procesos tanto en la gestión de ventas como en las comisiones, ya que al no contar con un sistema informático, la información se encuentra dispersada en diferentes formatos de archivos y cada departamento maneja a su manera la información de los procesos mencionados.

1.5.3 PROSPECTIVA DEL PROBLEMA

Si la empresa prosigue con el mismo sistema de procesamiento de la información y no se realiza la aplicación web, surgirán los mismos inconvenientes inadecuados y podrían ocasionar pérdidas económicas si existieran errores en la información presentada.

1.6 BENEFICIOS DE LA APLICACIÓN WEB

En la actualidad con el avance de la tecnología, las aplicaciones Web RIA han tenido un auge acelerado en todo tipo de empresas, mejorando la calidad en los procesos y en el entorno para el usuario final. Este tipo de aplicaciones requieren una sola configuración en el servidor ya que los clientes solo necesitan de los navegadores web y una buena conexión de red.



FIGURA 1.8: Beneficios de la web

Fuente: http://www.amh2web.com/images/stories/ventaja_aplicacion_web.jpg

Las aplicaciones web para Internet e Intranet presentan una serie de ventajas y beneficios con respecto al software de escritorio, con lo cual logrará aprovechar y acoplar los recursos de su empresa de una forma mucho más práctica que el software tradicional. Entre los beneficios que las aplicaciones desarrolladas para la web tienen respecto a las aplicaciones de escritorio se encuentran:

- El trabajo a distancia se realiza con mayor facilidad

Para trabajar en la aplicación web solo se necesita un computador con un buen navegador Web y conexión a internet.

- Las aplicaciones Web no necesitan conocimientos previos de informática.

Con una aplicación Web se tiene total disponibilidad en cuanto a hora y lugar, trabajar en ella en cualquier momento y en cualquier lugar del mundo siempre que tenga conexión a internet.

- Compatibilidad multiplataforma.

Las aplicaciones web tienen un camino mucho más sencillo para la compatibilidad multiplataforma que las aplicaciones de software descargables.

- Actualización.

Las aplicaciones basadas en web están siempre actualizadas con el último lanzamiento.

- Inmediatez de acceso.

Las aplicaciones basadas en web no necesitan ser descargadas, instaladas y configuradas. Usted accede a su cuenta online trabajar sin importar cuál es su configuración o su hardware.

- Menos requerimientos de memoria.

Las aplicaciones basadas en web tienen muchas más razonables demandas de memoria RAM de parte del usuario final que los programas instalados localmente.

- Menos Bugs.

Las aplicaciones basadas en web deberán ser menos propensas a colgarse y crear problemas técnicos debido a software o conflictos de hardware con otras aplicaciones existentes, protocolos o software personal interno. Con aplicaciones basadas en web, todos utilizan la misma versión, y todos los bugs pueden ser corregidos tan pronto como son descubiertos.

- Múltiples usuarios concurrentes.

Las aplicaciones basadas en web pueden realmente ser utilizada por múltiples usuarios al mismo tiempo.

- Disponibilidad del servidor

Toda la aplicación puede alojarse en un servidor propio garantizando soporte e información oportuna de los servicios soportados.

- Diseño personalizado

Las aplicaciones propias permiten que el diseño de su interfaz de usuario pueda ser personalizado de acuerdo a la imagen corporativa de la empresa o modelo de negocio que se tenga. Aprovechando el posicionamiento de su marca.

- Mantenimiento escalable

Una aplicación desarrollada especialmente para una empresa tiene la ventaja de ser escalable. Esto quiere decir que puede crecer, actualizarse y mejorarse a medida que las tecnologías cambian y evolucionan para tener mejores prestaciones en funcionamiento y seguridad.

- Exclusividad empresarial

Uno de los beneficios de las aplicaciones web personalizadas, son los datos y esto es lo que puede diferenciar en gran medida una aplicación web propia, ya que los datos y servicios de uso pertenecen y son de acceso exclusivo de la empresa para la cual se desarrolla la aplicación

CAPÍTULO II

2 MARCO TEÓRICO

2.1 BASE DE DATOS SQL SERVER 2008

SQL Server 2008 es la versión del Sistema Gestor de Bases de Datos de Microsoft para el mercado empresarial. Esta versión muestra grandes avances con respecto a sus predecesora (sobre todo SQL Server 2000) y, aunque solo está disponible para la plataforma Windows, comienza a ser un serio competidor para sistemas como Oracle, orientados a bases de datos de gran tamaño.



FIGURA 2.1: Aplicaciones de Sql Server 2008

Fuente: <http://www.cepeu.edu.py/imagenes/sql2008.gif>

2.1.1 INTRODUCCIÓN A SQL SERVER 2008

SQL Server 2008 incluye una gran cantidad de características que permiten una gestión más racional y eficaz del mismo, aumentan el rendimiento, la escalabilidad y la estabilidad del servidor y, permiten una configuración avanzada a nivel de servicios, seguridad del servidor, etc.

Entre las características, quizás la más llamativa sea el soporte para compatibilidad con .NET. Esto permite la programación de ensamblados en C# o Visual Basic 2008 y su ejecución en SQL Server, lo que abre un gran abanico de posibilidades complementando la funcionalidad que proporciona T-SQL.

Además, nos encontramos con soporte completo para XML, incluyendo un nuevo tipo de dato XML, que permite el almacenamiento de documentos de este tipo en formato nativo, y posibilita el uso de funciones de búsqueda y manipulación en código XML, como XQuery o XPath directamente en el propio campo de la tabla. También permite la validación de la estructura del propio documento contra esquemas XSD.

Las herramientas de configuración y mantenimiento también han sufrido una renovación completa. El Administrador Corporativo, o el Analizador de Consultas de SQL Server 2000 dejan paso al SQL Server Management Studio (SSMS), que agrupa y amplía la funcionalidad de ambos. Esta herramienta permite gestión multiservidor tanto de servidores OLTP como de servidores OLAP o gestión de funciones adicionales. Una característica muy interesante es la posibilidad de crear una solución, lo que permite una gestión más eficaz de proyectos de desarrollo, así como la integración con Visual Source Safe, que posibilita el trabajo en grupo y el control de código fuente.

SQL Server incorpora una herramienta denominada Business Intelligence Development Studio, que no es más que una versión reducida de Visual Studio 2008. Esta nueva herramienta sirve para desarrollar proyectos de integración con .NET, o también otros como proyectos de Analysis Services, Integration Services o Reporting Services.

SQL Server amplia y mejora los Reporting Services (SSRS), que aunque inicialmente fueron concebidos para aparecer con esta versión del servidor, Microsoft decidió publicar una versión reducida para SQL Server 2000. También incluye un mejorado y ampliado entorno para los Servicios de Integración (SQL Server Integration Services, SSIS), anteriormente conocidos como Data Transformation Services.

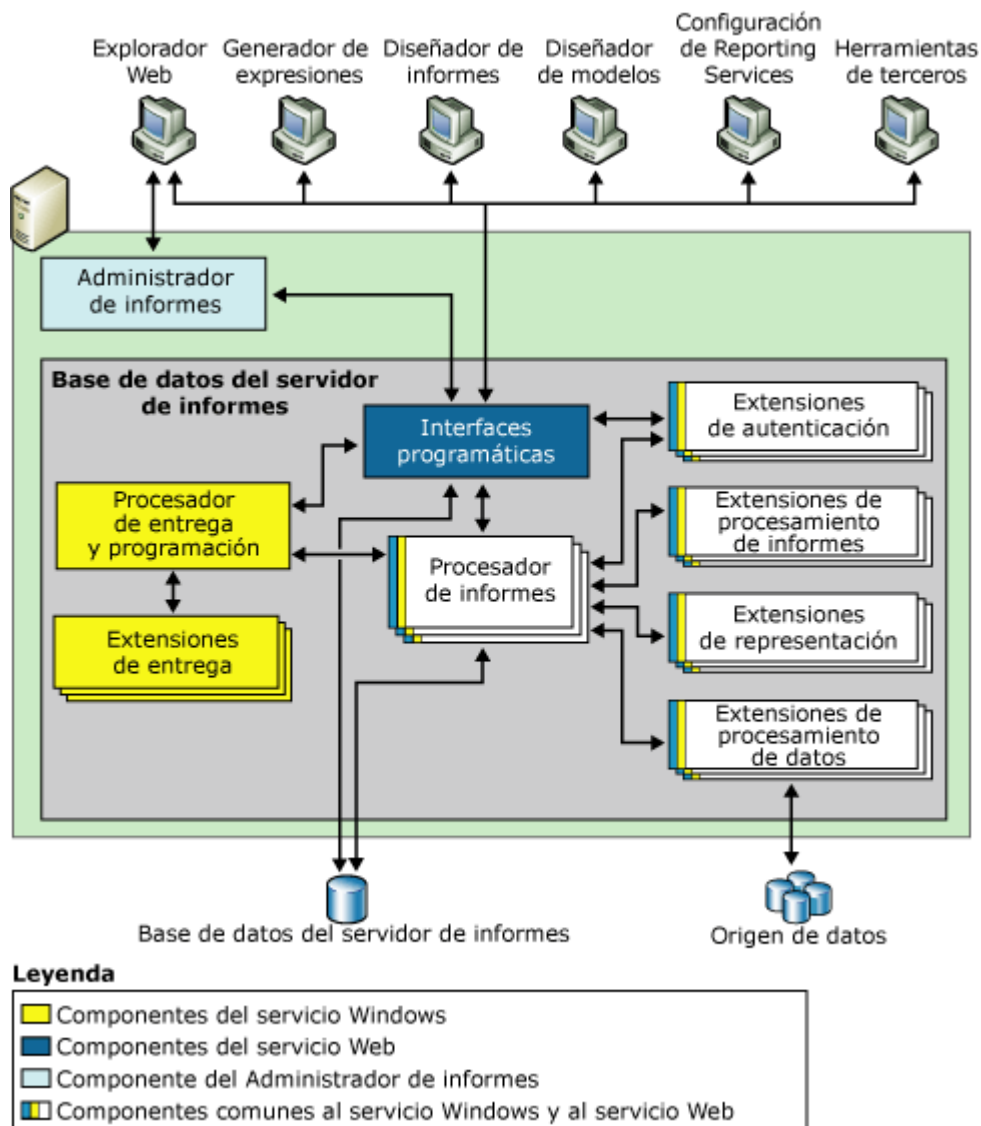


FIGURA 2.2: Arquitectura Report Services

Fuente: <http://i.technet.microsoft.com/dynimg/IC29933.gif>

Otra de las características la constituye el Service Broker, que es un entorno de trabajo diseñado para la distribución asíncrona de mensajería.

2.1.2 HERRAMIENTAS DE SQL SERVER 2008

La versión de SQL Server 2008, trae consigo herramientas, algunas nuevas, y otras simplemente una evolución de las anteriores.

2.1.2.1 LIBROS EN PANTALLA

Los libros en pantalla de SQL Server han sufrido una buena revisión, y utilizan la interfaz de ayuda de .NET 2008, que proporciona nuevos modelos de organización de datos, búsquedas por preferencias, online, en local, y muchas otras opciones.

2.1.2.2 SQL SERVER CONFIGURATION MANAGER

Esta herramienta agrupa la funcionalidad de varias en la anterior versión de SQL Server. Básicamente, su ámbito de aplicación se centra en la gestión avanzada de los servicios de soporte de SQL Server, para los cuales presenta un amplio abanico de opciones, y en la configuración de red del servidor.

2.1.2.3 EL SQL SERVER MANAGEMENT STUDIO

El SQL Server Management Studio es la nueva herramienta de administración de SQL Server, y representa una gran evolución frente a las versiones anteriores.

Hereda su aspecto y algo de funcionalidad de Visual Studio 2008, además de integrar un gran número de funciones que antes estaban disponibles como herramientas independientes.

2.2 PLATAFORMA RIA

2.2.1 INTRODUCCIÓN

Rich Internet Application (RIA), es una aplicación web que tiene muchas de las características de las aplicaciones de escritorio, por lo general entregados ya sea a través de un navegador de sitios específicos, a través de un navegador plug-in, o de forma independiente sandboxes o máquinas virtuales.

Adobe Flash, JavaFx y Silverlight de Microsoft son actualmente las tres plataformas más comunes, con tasas de penetración de alrededor del 99%, 80% y 54% respectivamente (en julio de 2010)⁴. A pesar de los nuevos estándares web han surgido, se siguen utilizando los principios detrás de RIAs.

El término “aplicación rica de Internet” se introdujo en un libro blanco de marzo de 2002 por Macromedia (ahora Adobe), aunque el concepto ha existido durante varios años antes bajo nombres tales como:

- Remote Scripting, por Microsoft, circa 1998
- X Internet, por Forrester Research en octubre de 2000
- Rich (web) clientes
- Aplicación web rica

Flash, Silverlight y JavaFx pretenden enriquecer las experiencias de usuario en parte debido a su menor dependencia de las comunicaciones red/servidor. Dado que el 80% del tiempo se dedica a descargar todos los componentes de la página, simplificando el diseño de una página, es también una manera de reducir el tiempo de respuesta. Otra forma de hacer frente a esto es reducir las peticiones HTTP mediante la combinación de todas las peticiones HTTP y hojas de estilo CSS.

Este enfoque permite que el sistema reduzca la cantidad y frecuencia del tráfico cliente-servidor.

2.2.2 QUÉ SON LOS RIA SERVICES?

Las aplicaciones RIA (Rich Internet Application) ⁵ están cada vez más desarrolladas por empresas de software, ya que entre otras permiten ofrecer una mejor experiencia al usuario.

⁴ <http://www.nohaylimites.com/?p=184>

⁵ Las *Rich Internet Applications*, o *RIA* (en español "aplicaciones de Internet enriquecidas"), son aplicaciones web que tienen la mayoría de las características de las aplicaciones de escritorio tradicionales

Arquitectura de una aplicación Web tradicional

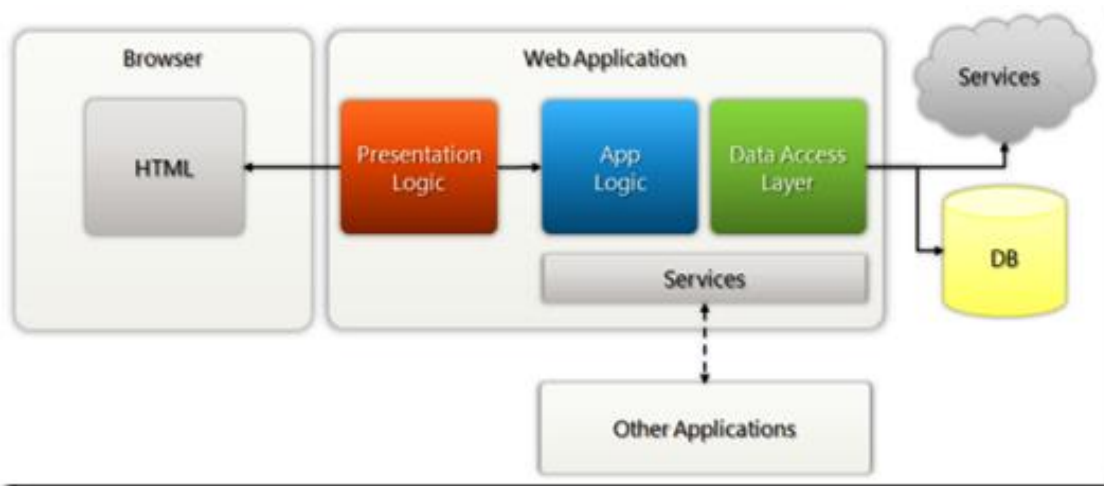


FIGURA 2.3: Aplicaciones RIA

Fuente: [http://es.wikipedia.org/wiki/Rich_Internet_Applications]

Arquitectura de una aplicación RIA

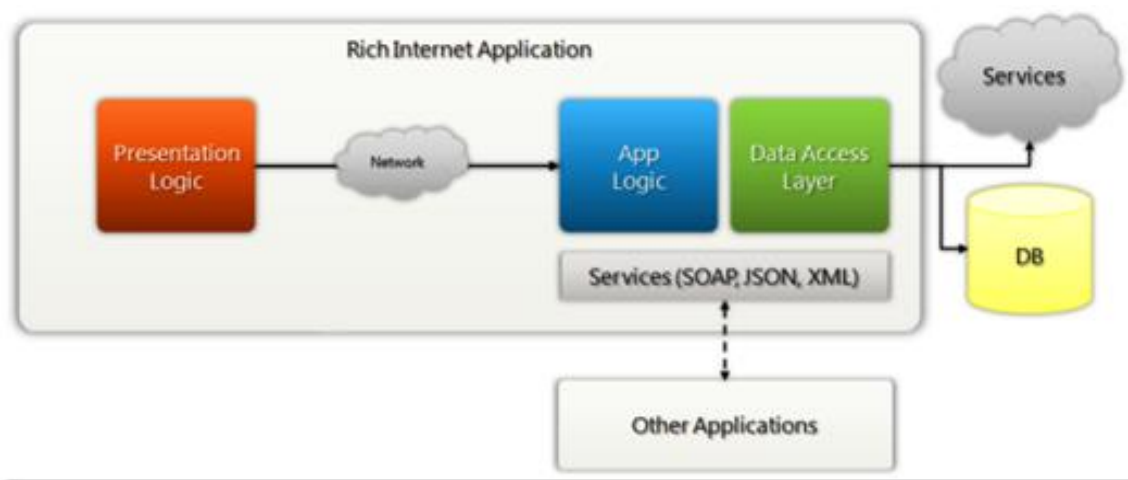


FIGURA 2.4: Componentes Aplicaciones RIA

Fuente: [http://es.wikipedia.org/wiki/Rich_Internet_Applications]

2.2.3 ARQUITECTURA

Hay tres aspectos muy importantes que influyen en forma directa en la performance de una aplicación RIA.

- El diseño de la aplicación y el contexto o ambiente de uso
- El comportamiento y expectativas del usuario
- El comportamiento de la aplicación cuando se la utiliza

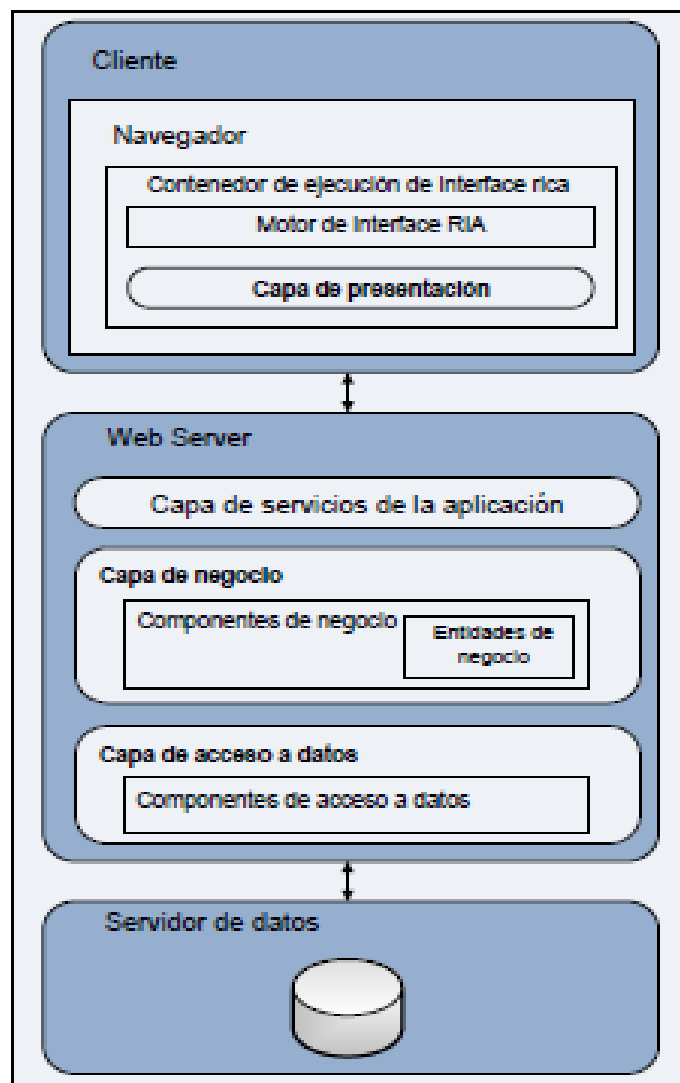


FIGURA 2.5: Arquitectura de una aplicación RIA

Fuente: [Autor]

2.2.4 ¿CUÁL ES EL OBJETIVO DE RIA SERVICES?

El objetivo de RIA Services es simplificar el desarrollo de aplicaciones RIA, diseñando aplicaciones como si fuesen aplicaciones web tradicionales, sin tener que preocuparnos de los aspectos básicos.

La arquitectura de las aplicaciones RIA, presenta dos componentes que aporta RIA Services, uno en la parte cliente (DomainContext) y otro en la parte servidora (DomainService).

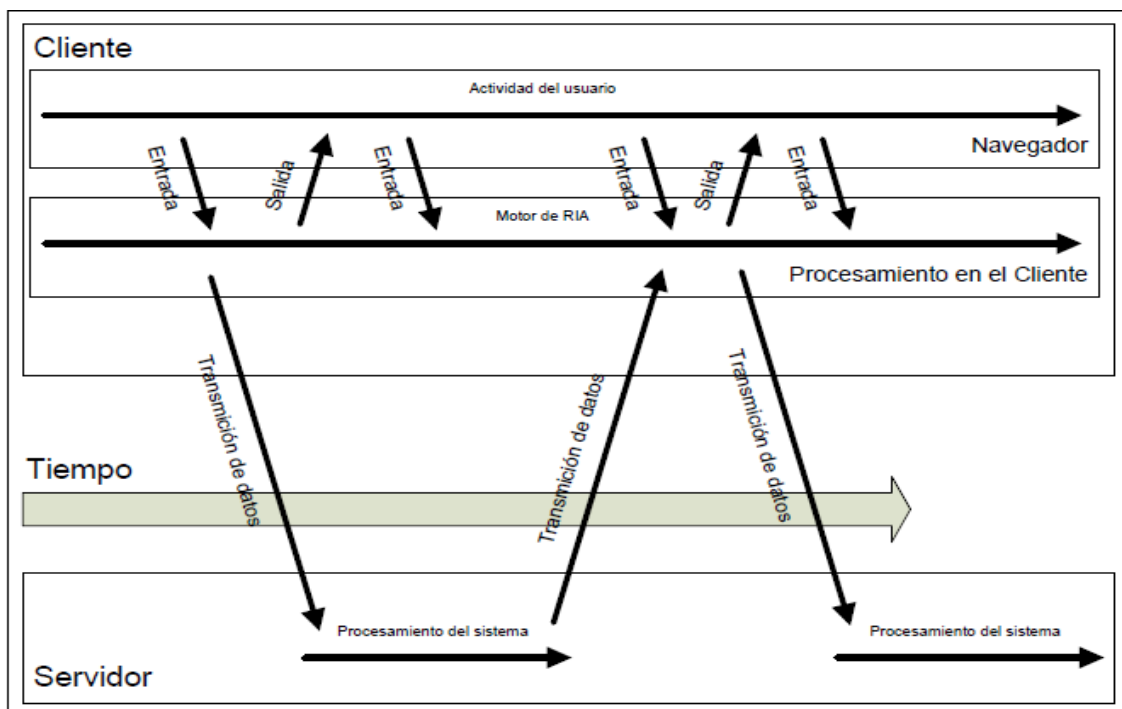


FIGURA 2.6: Procesamiento de datos en una aplicación RIA

Fuente: [http://es.wikipedia.org/wiki/Rich_Internet_Applications]

DomainDataSource. Este componente es muy similar al resto de DataSources, permite trabajar de una manera bastante cómoda y sencilla contra servicios de dominio que se exponen con RIA Services.

En una aplicación RIA hecha con Silverlight podemos tener los típicos formularios de lista/detalles, con las operaciones de selección, inserción, actualización, borrado, filtrado y ordenación como si tratase de una aplicación web tradicional.

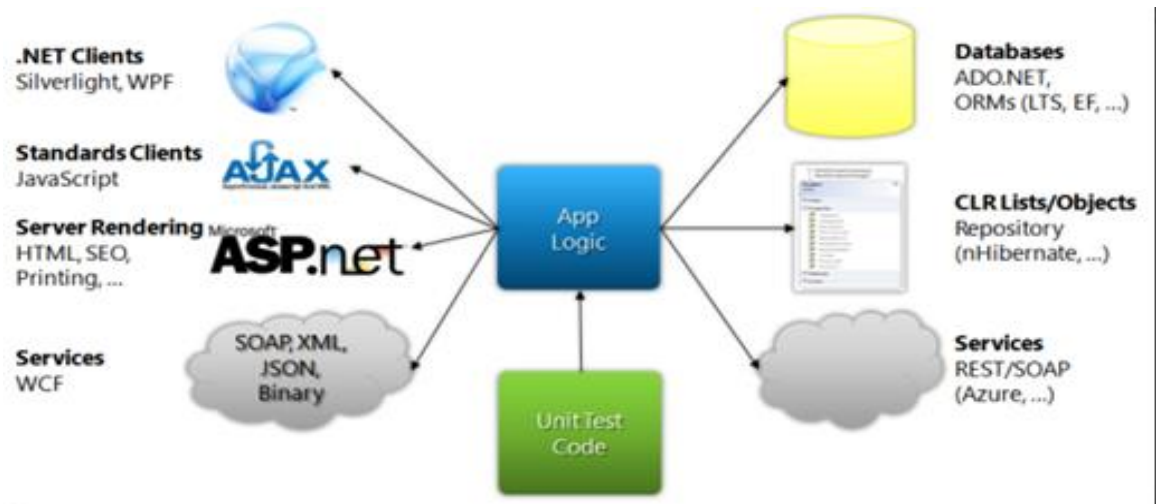


FIGURA 2.7: Arquitectura Orígenes de Datos de Dominio

Fuente: [[http://msdn.microsoft.com/es-es/library/ee707376\(v=vs.91\).aspx](http://msdn.microsoft.com/es-es/library/ee707376(v=vs.91).aspx)]

2.2.5 VENTAJAS DE LAS RIA'S

- ✓ La aplicación Web desarrollada permitirá que la ejecución de los procesos inherentes a la gestión de ventas y comisiones de la empresa sea más fácil, más rápido y se puede acceder desde cualquier lugar gracias al internet. Los controles relacionados a dicha gestión ejecutará el sistema de forma automática, la puesta en marcha de la aplicación reducirá notablemente el tiempo de entrega de reportes diarios y mensuales, además de eliminar totalmente la documentación que en la actualidad se genera (constancias, formatos de solicitudes, entre otros). La aplicación Web dará a conocer las aprobaciones de planes contratados de clientes por lo que se mantendrá informado a los usuarios a través de los reportes sobre las solicitudes ingresadas (Aprobación de Solicitudes, Denegación de Solicitudes, Anulación de Solicitudes, entre otros), para de esta manera llevar un control administrativo y llevar así a tomar decisiones para la mejora de las políticas de la empresa, así como también beneficiará a los asesores comerciales ya que podrán tener conocimiento de sus planes aprobados y el valor de su comisión a recibir que se calculara de acuerdo a la tabla de porcentajes establecida en la empresa.

- ✓ La arquitectura de la aplicación está basada en el modelo MVC, integrando interfaz web y modelo en un mismo servidor aunque conservando su independencia funcional.
- ✓ Actualmente, con la masificación de Ajax y el desarrollo de plataformas como Flex u Open Lazlo las RIA's ("Rich Internet Applications", o aplicaciones web de comportamiento similar a las de escritorio) se están tomando la web agresivamente.
- ✓ Los recursos son reutilizables y se pueden agregar a los favoritos del navegador. Pero lo más importante es que los recursos son indexables. Eso significa que los motores de búsqueda, vitales para organizar los millones de datos que circulan en la Red, pueden llegar a ellos, analizarlos y categorizarlos, facilitando la búsqueda para nosotros los usuarios y abriendo la posibilidad de generar nueva información a partir de la recombinação de recursos.

2.3 MICROSOFT SILVERLIGHT

Microsoft Silverlight es un complemento para navegadores de Internet basado en la plataforma Windows que agrega nuevas funciones multimedia como la reproducción de vídeos, gráficos vectoriales, animaciones y de entorno de desarrollo; en forma similar a lo que hace Adobe Flash.

Se define como "una tecnología que permite crear aplicaciones de vanguardia", destacando entre sus características más sobresalientes:

- XAML. El subconjunto de WPF que define los elementos en el navegador para crear la interfaz de usuario. Esto supone crear gráficos, animaciones y elementos multimedia, y otras características de cliente enriquecido, permitiendo ir más allá de lo que HTML permite.
- Extensiones de JavaScript. Las extensiones al lenguaje de scripting de explorador que permiten controlar la interfaz de usuario, incluida la posibilidad

de trabajar con elementos XAML. La comunicación entre los dos mundos (HTML/DOM y XAML) puede, además, realizarse en los dos sentidos.

- Compatibilidad con otros exploradores y plataformas. Su ejecución es idéntica en todos los exploradores conocidos (y en sus plataformas). Esto incluye varias versiones de Windows y Mac, y gracias a la plataforma MoonLight, Linux y FreeBSD.
- Integración con las aplicaciones existentes. Se integra perfectamente con el código JavaScript y ASP.NET AJAX existente, de modo que complementa la funcionalidad ya creada. De esta forma, Silverlight puede empotrarse como una isla dentro de una página o cubrir toda la interfaz de usuario.
- Modelo de programación similar y compatible con .NET Framework. Basado en la Capa de Adaptación de Plataforma (PAL), permite su ejecución en paralelo junto a otras versiones de .NET. También se pueden utilizar lenguajes dinámicos, como IronPython o IronRuby y enlazar con código de un lenguaje funcional como F#.
- Amplio conjunto de herramientas para desarrollo. Tanto Visual Studio .NET (2008/2010) como Expression Blend, permiten desarrollar para esta plataforma, si bien el soporte de la versión 4 de Silverlight se alinea con el mismo número de versión en Blend.
- Compatibilidad de red. Silverlight incluye compatibilidad con HTTP sobre TCP. Se puede conectar a los servicios WCF, SOAP, REST o ASP.NET AJAX y manejar formatos como Texto plano, XML, JSON o RSS.
- Soporte LINQ. Language Integrated Query, el lenguaje integrado de consultas que permite analizar y manejar orígenes de datos diversos con una sintaxis sencilla, intuitiva y muy potente.

Se mejora el rendimiento de la tecnología DeepZoom, que permite a los usuarios profundizar en el detalle de fotografías o collages de fotografías, permitiendo una transición suave entre ellas, para obtener la sensación de profundidad.

Las imágenes se pueden escalar desde resoluciones de 2 ó 3 megapíxeles hasta un gigapíxel, pero el usuario no tiene que esperar a la descarga completa, gracias a la tecnología subyacente que permite la descarga concreta de las zonas necesarias. Internamente, utiliza el formato de fichero XML.

Se denomina una potente API de Media Stream Source (o streaming adaptativo) que permite la descarga dinámica adaptable de contenidos multimedia. Esto permite a la aplicación que reproduce el contenido multimedia seleccionar la velocidad de descarga basándose en el ancho de banda del cliente y su potencia de CPU. Se implementa mediante una instalación en el servidor donde se aloja el contenido multimedia.

Inconvenientes

- La creación de gráficos vectoriales y animaciones debe realizarse con Expression Design, herramienta con licencia.
- No soportado en todos los navegadores. Soporte para Internet Explorer, Safari y Firefox

CAPÍTULO III

3 DISEÑO Y ARQUITECTURA DEL SISTEMA

3.1 INSTALACIÓN Y CONFIGURACIÓN DE MICROSOFT SILVERLIGHT

3.1.1 BLOQUES DE CONSTRUCCIÓN

Silverlight proporciona muchos controles útiles para mostrar la información y la manipulación de datos de entrada. Antes de llegar a las características específicas de cada control, es importante entender la funcionalidad básica de todos los controles disponibles para Silverlight.

La figura 3.1 muestra un diagrama de clase abreviado con un subconjunto de controles y de paneles de Silverlight 4.

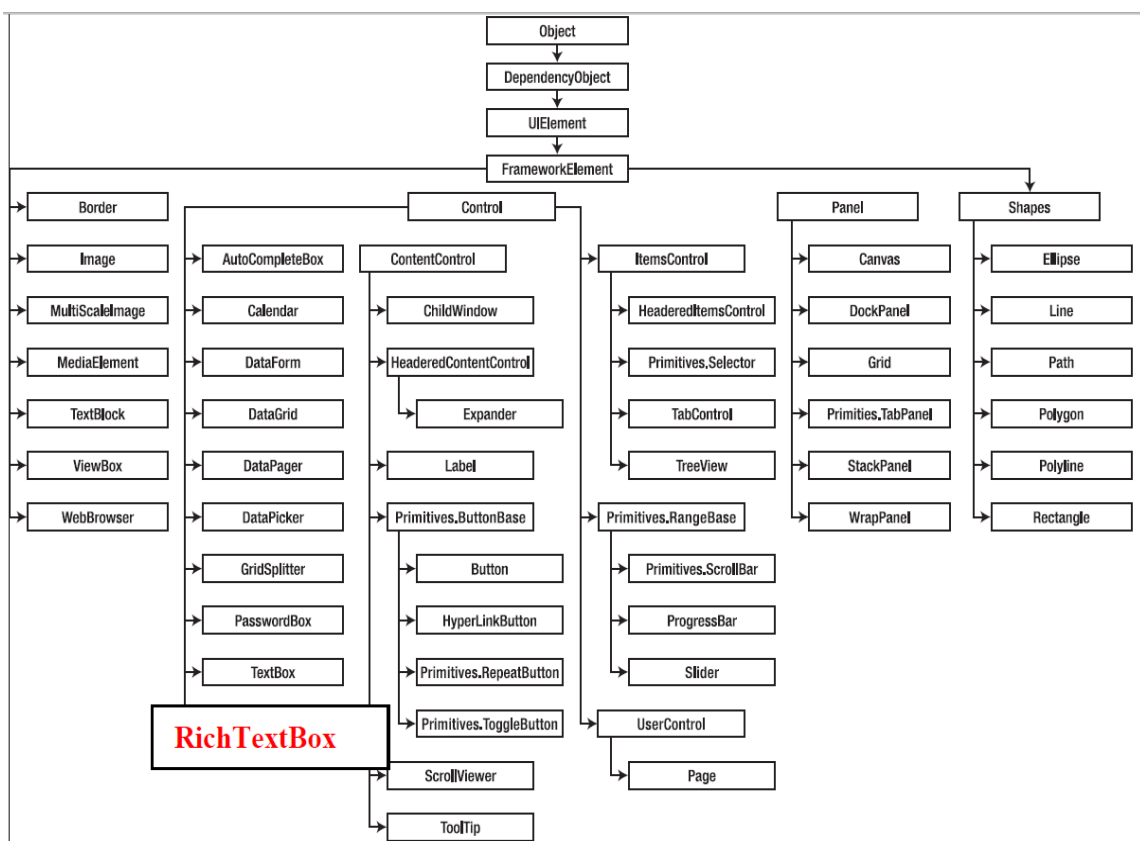


FIGURA 3. 1: Clase Jerárquica de la interfaz de usuario en Silverlight

Fuente: www.microsoft.com/la/silverlight

3.1.1.1 DEPENDENCYOBJECT

La clase DependencyObject es sin duda la clase más importante en Silverlight ya que habilita los servicios del sistema de propiedades de dependencia en sus numerosas clases derivadas.

Los servicios y características de DependencyObject incluyen lo siguiente:

- Compatibilidad con el host de propiedades de dependencia.
- Compatibilidad con el host de propiedades de dependencia personalizadas.
- Compatibilidad con el host de propiedades asociadas.
- Compatibilidad con el host de propiedades adjuntas personalizadas.
- Métodos para obtener y establecer valores de las propiedades de dependencia que existan en DependencyObject.
- API del distribuidor para escenarios de subprocesos avanzados.
- Enlace de datos básico y compatibilidad de estilo, permitiendo establecer las propiedades como expresiones que se van a evaluar en algún momento posterior de la duración de un objeto.

Sus características más importantes son los medios que proporciona, que se muestra en la Tabla 3.1.

TABLA 3. 1: Métodos de la clase System.Windows.DependencyObject

Método	Descripción
CheckAccess	Determina si el subproceso que realiza la llamada tiene acceso a este objeto.
ClearValue	Borra el valor local de una propiedad de dependencia.
GetAnimationBaseValue	Devuelve cualquier valor base establecido para una propiedad de dependencia de Silverlight, que se aplicaría en los casos en que una animación no esté activa.
ReadLocalValue	Devuelve el valor local de una propiedad de dependencia, si hay un valor local establecido.
SetValue	Establece el valor local de una propiedad de dependencia de un objeto DependencyObject.

Fuente: www.microsoft.com/la/silverlight

3.1.1.2 THREADING Y LA INTERFAZ DE USUARIO

Silverlight es un entorno multihilo. Usted no puede modificar los elementos de una interfaz de usuario desde un subproceso, ya que puede dar lugar a una serie de problemas. La forma correcta de modificar un interfaz de usuario desde un subproceso diferente es el uso de un despachador. La clase `DependencyObject` proporciona una sola propiedad, `Dispatcher`, que contiene una referencia a la operadora asociada. Si desea establecer el valor de un bloque de texto desde un subproceso diferente, debe utilizar `Dispatcher.BeginInvoke` a la cola de la modificación en el trabajo, de la cola del hilo principal de este tipo de elementos:

```
Dispatcher.BeginInvoke (delegado () {textBlock.Text = "ha cambiado";});
```

3.1.1.3 UIELEMENT

`UIElement` proporciona un punto inicial para las características de diseño de Silverlight.

Gran parte del comportamiento de los elementos visibles en una interfaz de usuario de Silverlight que se define en la clase `UIElement`. Esto incluye los eventos de teclado, mouse y entrada del lápiz óptico así como los eventos de foco.

El tipo UIElement expone los siguientes miembros en la tabla 3.2.

TABLA 3. 2: Métodos de la clase System.Windows.UIElement

Método	Descripción
AddHandler	Agrega un controlador de un evento enrutado especificado y agrega el controlador a la colección de controladores del elemento actual. Especifique handledEventsToo como true de modo que el controlador proporcionado se invoque para el evento enrutado ya marcado como controlado por otro elemento a lo largo de la ruta de evento.
Arrange	Coloca los objetos secundarios y determina el tamaño de UIElement. Los objetos primarios que implementan un diseño personalizado de sus elementos secundarios llaman a este método desde las implementaciones de invalidación del diseño para formar una actualización recursiva del diseño.
CaptureMouse	Establece la captura del mouse en UIElement.
InvalidateArrange	Invalida el estado de organización (diseño) de UIElement. Después de la invalidación, UIElement actualizará su diseño, lo que se producirá de forma asíncronica.
InvalidateMeasure	Invalida el estado de medida (diseño) de UIElement.
Measure	Actualiza la propiedad DesiredSize de UIElement. Normalmente, los objetos que implementan el diseño personalizado para sus elementos secundarios del diseño llaman a este método desde sus propias implementaciones de MeasureOverride para formar una actualización recursiva del diseño.
OnCreateAutomationPeer	Cuando se implementa en una clase derivada, devuelve implementaciones AutomationPeer específicas de la clase para la infraestructura de automatización de Silverlight.
ReleaseMouseCapture	Quita la captura del mouse de UIElement. Normalmente, después de esta llamada, ningún objeto tiene la captura del mouse.
RemoveHandler	Quita el controlador de eventos enrutados especificado de UIElement.
TransformToVisual	Devuelve un objeto de transformación que se puede usar para transformar las coordenadas desde el objeto UIElement en el objeto visual especificado.
UpdateLayout	Se asegura de que todas las posiciones de objetos secundarios de UIElement se actualizan correctamente para el diseño.

Fuente: www.microsoft.com/la/silverlight

Las propiedades del tipo `UIElement` se describen en la tabla 3.3

TABLA 3. 3: Propiedades de la clase `System.Windows.UIElement`

Propiedad	Descripción
<code>AllowDrop</code>	Obtiene o establece un valor que determina si <code>UIElement</code> puede ser un destino de colocación en las operaciones de arrastrar y colocar de <code>Silverlight</code> .
<code>CacheMode</code>	Obtiene o establece un valor que indica que el contenido representado debe almacenarse en la memoria caché cuando sea posible.
<code>Clip</code>	Obtiene o establece el objeto <code>Geometry</code> utilizado para definir el contorno del contenido de <code>UIElement</code> .
<code>DesiredSize</code>	Obtiene el tamaño que este <code>UIElement</code> calculó durante el paso de la medida del proceso de diseño.
<code>Dispatcher</code>	Obtiene el objeto <code>Dispatcher</code> al que está asociado este objeto. (Se hereda de <code>DependencyObject</code>).
<code>Effect</code>	Obtiene o establece el efecto de sombreador de píxeles que se va a utilizar para representar esta instancia de <code>UIElement</code> .
<code>IsHitTestVisible</code>	Obtiene o establece si el área contenida de <code>UIElement</code> puede devolver valores <code>true</code> para la prueba de posicionamiento.
<code>Opacity</code>	Obtiene o establece el grado de opacidad del objeto.
<code>OpacityMask</code>	Obtiene o establece el pincel utilizado para modificar la opacidad de las regiones de este objeto.
<code>Projection</code>	Obtiene o establece la proyección en perspectiva (efecto 3D) que se va a aplicar al representar esta instancia de <code>UIElement</code> .
<code>RenderSize</code>	Obtiene el tamaño de representación final de un <code>UIElement</code> .
<code>RenderTransform</code>	Obtiene o establece información de transformación que afecta a la posición de representación de <code>UIElement</code> .
<code>RenderTransformOrigin</code>	Obtiene o establece el punto de origen de cualquier posible transformación de representación declarada por <code>RenderTransform</code> , relativo a los límites de <code>UIElement</code> .
<code>UseLayoutRounding</code>	Obtiene o establece un valor que determina si la representación del objeto y su subárbol visual deben utilizar un comportamiento de redondeo que alinee la representación a píxeles enteros.
<code>Visibility</code>	Obtiene o establece la visibilidad de un <code>UIElement</code> . Un <code>UIElement</code> que no está visible no representa ni comunica su tamaño deseado al diseño.

Fuente: www.microsoft.com/la/silverlight

UIElement también define varios eventos importantes, que se muestra en la Tabla 3.4

TABLA 3. 4: Eventos de la clase System.Windows.UIElement

Evento	Descripción
DragEnter	Tiene lugar cuando el sistema de entrada comunica un evento de arrastre subyacente que tiene este elemento como destino.
DragLeave	Se produce cuando el sistema de entrada comunica un evento de arrastre subyacente que tiene este elemento como origen.
DragOver	Se produce cuando el sistema de entrada comunica un evento de arrastre subyacente que tiene este elemento como posible destino de colocación.
Drop	Se produce cuando el sistema de entrada comunica un evento de colocación subyacente que tiene este elemento como destino de la colocación.
GotFocus	Se produce cuando UIElement recibe el foco.
KeyDown	Se produce cuando se presiona una tecla del teclado mientras UIElement tiene el foco.
KeyUp	Se produce cuando se suelta una tecla del teclado mientras UIElement tiene el foco.
LostFocus	Se produce cuando UIElement pierde el foco.
LostMouseCapture	Aparece cuando el objeto UIElement pierde la captura del mouse.
MouseEnter	Se produce cuando el mouse (o un lápiz) entra en el área delimitadora de UIElement.
MouseLeave	Se produce cuando el mouse (o el lápiz) sale del área delimitadora de UIElement.
MouseLeftButtonDown	Se produce cuando se presiona el botón primario del mouse (o cuando la punta del lápiz toca el Tablet PC) mientras el puntero del mouse está sobre un objeto UIElement.

MouseButtonUp	Se produce cuando se suelta el botón primario del mouse (o la punta del lápiz se aparta del Tablet PC) mientras el mouse (o el lápiz) está sobre UIElement (o mientras UIElement tiene la captura del mouse).
MouseMove	Se produce cuando cambia la posición de las coordenadas del mouse (o lápiz) mientras está sobre UIElement (o mientras UIElement tiene la captura del mouse).
MouseButtonDown	Se produce cuando se presiona el botón secundario mientras el puntero del mouse está sobre un objeto UIElement.
MouseButtonUp	Se produce cuando se suelta el botón secundario mientras el puntero del mouse está sobre un objeto UIElement. Sin embargo, este evento solo se generará si un llamador marca el evento MouseButtonDown anterior como controlado; vea la sección Comentarios.
MouseWheel	Se produce cuando el usuario gira la rueda del mouse mientras el puntero del mouse está sobre UIElement o mientras UIElement tiene el foco.
TextInput	Se produce cuando un elemento de la interfaz de usuario obtiene el texto de forma independiente del dispositivo.
TextInputStart	Se produce cuando un elemento de la interfaz de usuario obtiene inicialmente el texto de forma independiente del dispositivo.
TextInputUpdate	Se produce cuando el texto continúa componiéndose por medio de un editor de métodos de entrada (IME).

Fuente: www.microsoft.com/la/silverlight

3.1.1.4 LA CLASE CONTROL

La clase Control es la clase base de muchos de los controles que se agregan a una aplicación. La clase Control define un comportamiento muy reducido y aunque es posible agregar una clase Control a la aplicación, es más habitual agregar un control que herede de Control, como Button o ListBox.

El tipo Control expone los siguientes miembros que se muestran en la tabla 3.5

TABLA 3. 5: Propiedades de la clase System.Windows.Controls.Control

Nombre	Descripción
Background	Obtiene o establece un pincel que proporciona el fondo del control.
BorderBrush	Obtiene o establece un pincel que describe el fondo del borde de un control.
BorderThickness	Obtiene o establece el grosor del borde de un control.
DefaultStyleKey	Obtiene o establece la clave que hace referencia al estilo predeterminado del control.
FontFamily	Obtiene o establece la fuente utilizada para mostrar texto en el control.
FontSize	Obtiene o establece el tamaño del texto de este control.
FontStretch	Obtiene o establece el grado en el que se comprime o se expande una fuente en la pantalla.
FontStyle	Obtiene o establece el estilo en que se representa el texto.
FontWeight	Obtiene o establece el grosor de la fuente especificada.
Foreground	Obtiene o establece un pincel que describe el color de primer plano.
HorizontalAlignment	Obtiene o establece la alineación horizontal del contenido del control.

IsEnabled	Obtiene o establece un valor que indica si el usuario puede interactuar con el control.
IsTabStop	Obtiene o establece un valor que indica si un control está incluido en la navegación por tabulación.
Padding	Obtiene o establece el relleno interior del control.
TabIndex	Obtiene o establece un valor que determina el orden en el que los elementos reciben el foco cuando el usuario navega por los controles utilizando la tecla TAB.
TabNavigation	Obtiene o establece un valor que modifica el modo en que la tabulación y el control TabIndex funcionan para este control.
Template	Obtiene o establece una plantilla de control.
UseLayoutRounding	Obtiene o establece un valor que determina si la representación del objeto y su subárbol visual deben utilizar un comportamiento de redondeo que alinee la representación a píxeles enteros. (Se hereda de UIElement).
VerticalContentAlignment	Obtiene o establece la alineación vertical del contenido del control.

Fuente: www.microsoft.com/la/silverlight

3.2 ACCESO A DATOS

3.2.1 ENLACE CON DATOS

BINDING

Binding es una poderosa manera de crear una conexión entre la interfaz de usuario y una fuente de datos. Esta sencilla técnica puede usarse para crear una clara separación entre la interfaz de usuario y sus datos subyacentes y es esencial para la arquitectura, independientemente de la razón, puede utilizar el enlace de datos en su aplicación mediante la creación de una instancia de la clase Binding.

La clase Binding se utiliza para definir una conexión entre un objeto CLR y un componente de interfaz de usuario. Esta conexión se define por tres elementos esenciales: la fuente de los datos (el objeto CLR), el modo de enlace, y el objetivo de los datos.

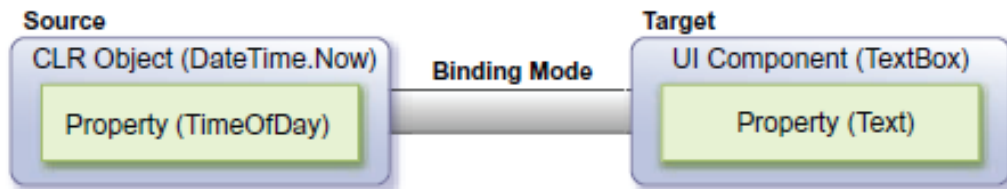


FIGURA 3. 2: Una visión conceptual de enlace de datos. La fuente posee los datos, la función objetivo (exposiciones, ediciones, y así sucesivamente) de los datos

Fuente: www.microsoft.com/la/silverlight

ONETIME

La opción ONETIME establece la propiedad de destino a la propiedad de origen cuando un enlace se inicia. Cuando se utiliza este BindingMode, cualquier cambio en el origen de datos no se envía automáticamente a la meta.

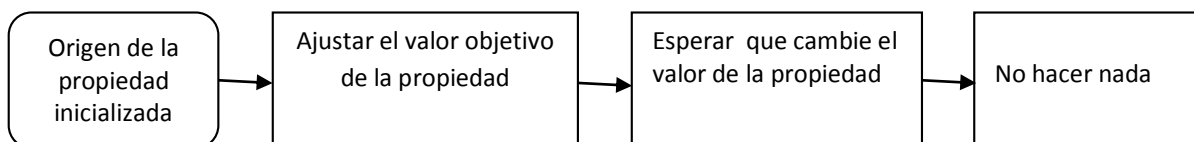


FIGURA 3. 3: Una visión conceptual de la otrora enlace a un origen de datos. El valor es un principio leer desde la fuente y nunca se volvió a actualizar.

Fuente: www.microsoft.com/la/silverlight

ONEWAY

El BindingMode OneWay es la opción predeterminada cuando se crea un enlace. Esta opción le da la capacidad para recibir automáticamente los cambios de una propiedad de origen, pero la propiedad de origen no va a cambiar si el destino se ve alterado.

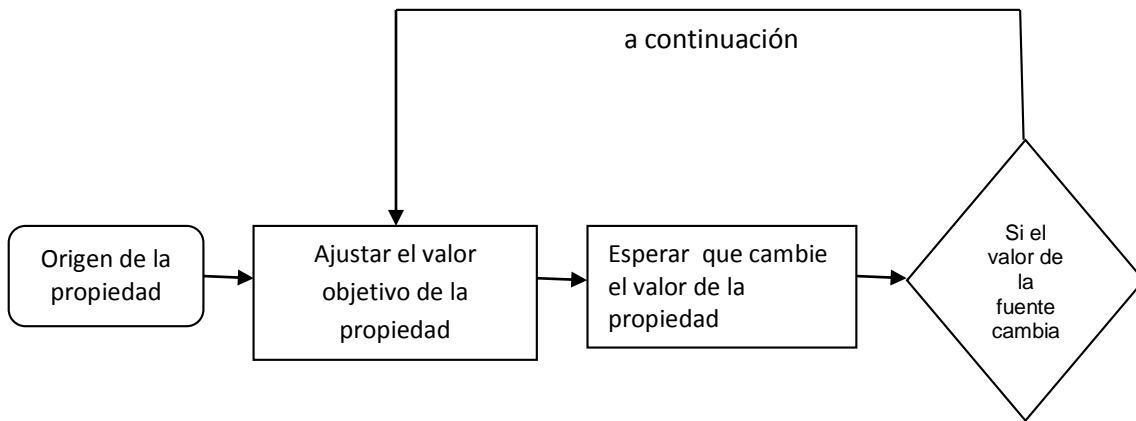


FIGURA 3. 4: Una visión conceptual de OneWay enlace a un origen de datos. El valor se actualiza cada vez que cambia la fuente, pero los cambios en el valor en el control de destino no lo hacen de nuevo a la fuente.

Fuente: www.microsoft.com/La/Silverlight

TWOWAY

Binding TwoWay permite que dos propiedades que están obligadas a cambiar al otro. Esto puede parecer recurrente, pero no lo es. Un cambio Binding TwoWay el objetivo cuando se cambia la fuente. Si los cambios de destino, la fuente se actualiza.

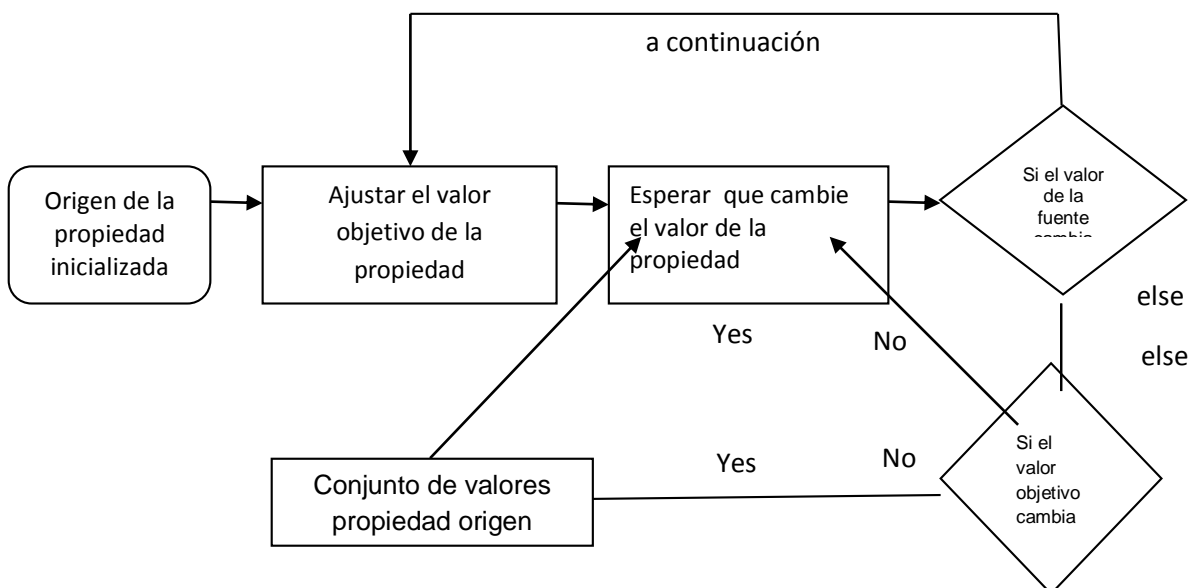


FIGURA 3. 5: Una visión conceptual de TwoWay enlace a un origen de datos. El control de destino refleja cambios en el origen y la fuente se actualiza con los cambios realizados en el destino.

Fuente: www.microsoft.com/la/silverlight

3.3 PATRÓN MVVM

3.3.1 EL MODELO

El modelo representa los datos o información con la que se trabaja, un ejemplo de modelo puede ser un contacto (como su nombre, número de teléfono, dirección y demás) o la descripción de un punto de publicación para un medio audiovisual transmitido en vivo.

La clave para recordar el modelo es que contiene la información, pero no las acciones o servicios que la manipulan. No es responsable de darle forma para que se vea bien en la pantalla, o de obtener una lista de elementos de un servidor remoto (de hecho, en tal lista cada elemento sería a su vez un modelo). La lógica de la aplicación o “reglas empresariales” son generalmente mantenidas en clases separadas del modelo y actúan en él.

3.3.2 LA VISTA

La vista es la parte con la que más se puede familiarizar y la que ve el usuario. Su papel es representar la información, tomándose a veces ciertas libertades con el fin de hacerla más clara o presentable. Una vista puede también contener ciertos comportamientos, como el aceptar la entrada de datos. La vista se encarga de esta faceta (teclas presionadas, movimientos del ratón, gestos en una pantalla táctil, y así por el estilo) que eventualmente ejerce influencia en las propiedades del modelo.

En MVVM la vista es activa. A diferencia de una vista pasiva sin conocimiento del modelo, y bajo el manejo total de un controlador o presentador, la vista en MVVM contiene comportamientos, eventos y enlaces a datos que, hasta cierto punto, necesitan saber sobre el modelo subyacente y el modelo de vista

Algo digno de recordar sobre la vista es que **no** es responsable de llevar cuenta de su estado. El modelo de vista se encarga de ello y mantiene la vista al tanto de los cambios.

3.3.3 EL MODELO DE VISTA (NUESTRO CONTROLADOR O PRESENTADOR)

El modelo de vista introduce el concepto de *separación de la presentación*, es decir, mantiene al modelo separado y protegido de los minuciosos detalles de la vista. Por eso es que el modelo de vista es la pieza clave del trío. Esta separación permite que el modelo se limite a contener los datos.

El modelo de vista también hace disponibles métodos, comandos y otros puntos de acceso que ayudan a mantener el estado de la vista, manipular el modelo en respuesta a acciones de la vista y disparar eventos en la misma.

Los ejemplos del patrón generalmente enfatizan el uso de XAML para definir la vista y enlaces de datos para los comandos y propiedades.

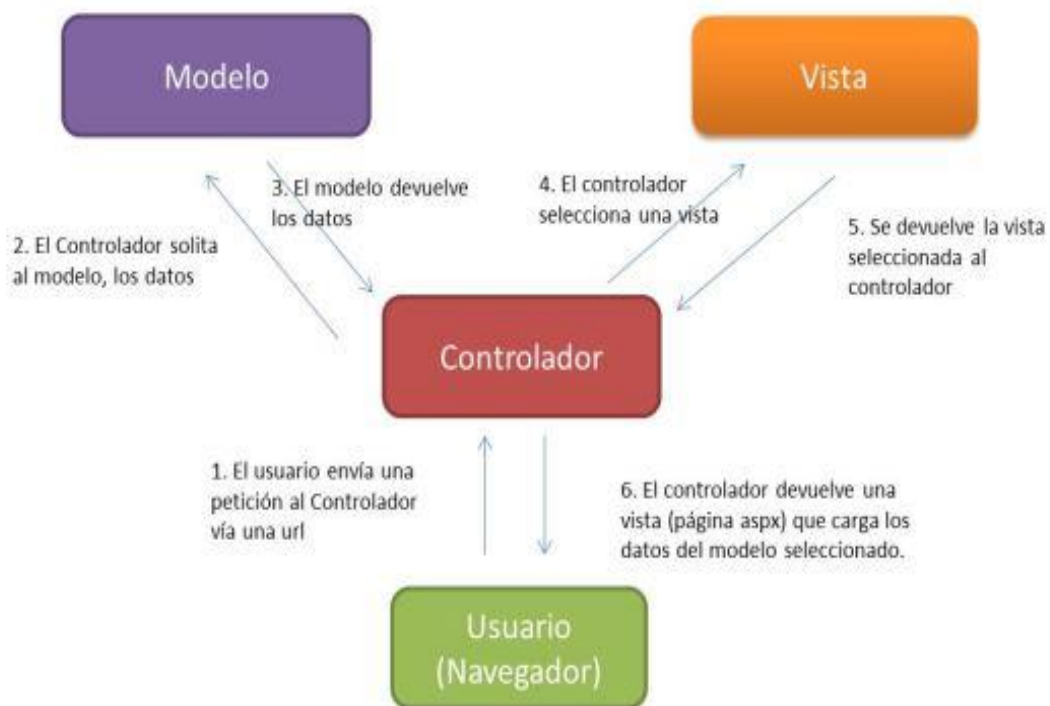


FIGURA 3. 6: Detalle del modelo de vista

3.3.4 LA VISTA Y EL MODELO DE VISTA

- La vista y el modelo de vista se comunican mediante enlaces de datos, métodos, propiedades, eventos y mensajes
- El modelo de vista expone propiedades (tal como información sobre el estado de la vista, p.ej. el indicador de “ocupado”) y comandos además de modelos
- La vista se encarga de sus propios eventos relacionados con la interfaz al usuario y los pasa al modelo de vista mediante comandos
- Los modelos y propiedades en el modelo de vista son actualizados desde la vista usando enlaces de datos bidireccionales

3.3.5 EL MODELO DE VISTA Y EL MODELO

El modelo de vista es completamente responsable del modelo en este escenario.

Por dicha no está solo:

- El modelo de vista puede que exponga el modelo directamente o mediante propiedades relacionadas para uso en enlaces de datos
- El modelo de vista puede contener interfaces a servicios, datos de configuración y similares, con el fin de obtener y manipular las propiedades que ofrece

3.3.6 INFRAESTRUCTURA BÁSICA PARA MVVM

La infraestructura para MVVM básica requiere sólo dos cosas:

1. Una clase basada en *DependencyObject* o que implemente *INotifyPropertyChanged* de manera que sea utilizable en el enlace de datos, y
2. Algún tipo de mecanismo para comandos

“Silverlight 3 cumple hasta cierto grado con el segundo requisito al proveer la interface *ICommand*. Sin embargo, ahora en Silverlight 4 se tiene algo más completo: los comandos permiten enlazar eventos en la vista al modelo de vista

Es un detalle de implementación, pero facilita el uso del patrón MVVM⁶.

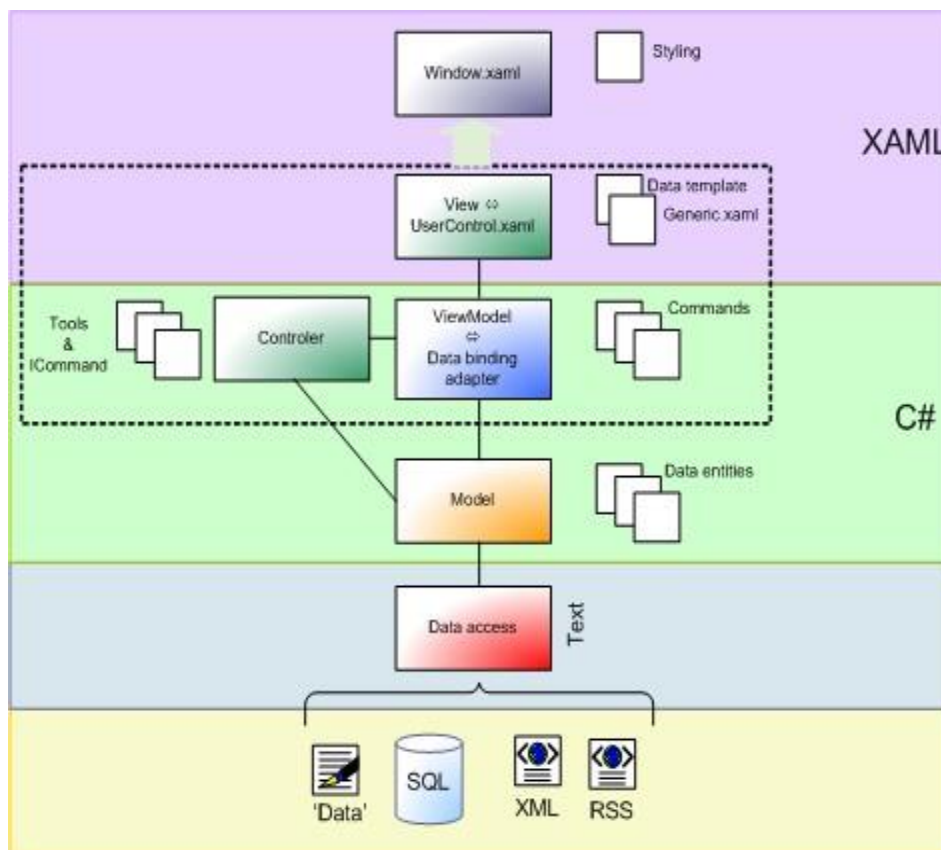


FIGURA 3. 7: Detalle del modelo de MVVM

3.4 WCF RIA SERVICES

“Servicios WCF RIA simplifica el desarrollo de soluciones de n niveles de Rich Internet Applications (RIA), como las aplicaciones de Silverlight⁷. Un problema común en el desarrollo de una solución RIA n niveles es la coordinación de la lógica de aplicación entre el nivel medio y nivel de la presentación.

Servicios RIA proporcionan componentes del marco, las herramientas y servicios que hacen que la lógica de aplicación en el servidor a disposición del cliente RIA Servicios sin necesidad de duplicar manualmente la lógica de programación.

⁶ Patrón Modelo-Vista-Modelo de Vista <http://maromasdigitales.net/2010/05/patron-mvvm-explicado/>

⁷ WINDOWS COMMUNICATION FOUNDATION RIA SERVICES;

<http://www.todoroms.com/windows-communication-foundation-ria-services-2>

En la figura 3.8 se muestra una versión simplificada de una aplicación de n niveles. Servicios RIA se centra en la caja entre el nivel de presentación y la capa de acceso a datos (DAL) para facilitar el desarrollo de n niveles con un cliente de Servicios de RIA.

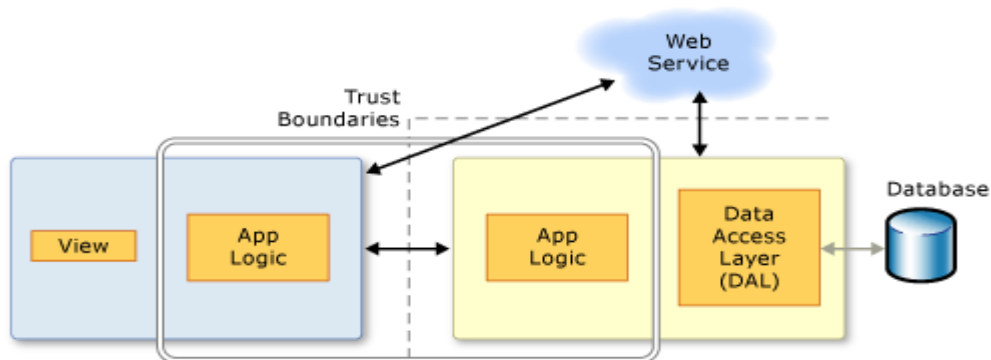


FIGURA 3. 8.: Se muestra una versión simplificada de una aplicación de n niveles

RIA Services agrega herramientas para Visual Studio 2010 que permiten vincular los proyectos de cliente y servidor en una única solución y la generación de código para el proyecto del cliente a partir del código de nivel medio. Los componentes del marco de apoyo a los patrones de prescripción para la escritura lógica de la aplicación para que pueda ser reutilizado en la capa de presentación.

3.4.1 WCF INTEGRACIÓN

En Servicios de RIA, se exponen los datos del proyecto de servidor de proyecto del cliente mediante la adición de los servicios de dominio. El marco RIA Services implementa cada servicio de dominio como un servicio Windows Communication Foundation (WCF).

3.4.1.1 SERVICIOS DE DOMINIO

Los servicios son de dominio de Windows Communication Foundation (WCF) que encapsulan la lógica de negocio de una aplicación RIA Servicios WCF. Un servicio de dominio expone un conjunto de operaciones relacionadas en forma de una capa de servicio.

Cuando se define un servicio de dominio, debe especificar los datos de las operaciones que se permiten a través del servicio de dominio.

Normalmente, estas tareas implican un pequeño grupo de entidades estrechamente relacionadas. Por ejemplo, en una solicitud de informes de gastos, que podría exponer a las entidades de informes de gastos, las partidas, y los detalles

3.4.1.2 SERVICIOS DE DOMINIO Y SUS FUENTES DE DATOS

El `DomainService` es la clase base para todas las clases que sirven de servicios de dominio. Para crear un servicio de dominio que se une a un objeto de datos personalizada, debe crear una clase que se deriva directamente de `DomainService`, hay clases especiales de lo abstracto se derivan de `DomainService` que utilice en su lugar.

- Para crear un servicio de dominio que se une a una entidad modelo de ADO.NET, se crea una clase que deriva de `LinqToEntitiesDomainService`. RIA Services proporciona la clase **`LinqToEntitiesDomainService`**.
- Si desea crear un servicio de dominio que expone clases de LINQ to SQL en su aplicación, se crea una clase que deriva de `LinqToSqlDomainService`. Si desea crear un servicio de dominio que expone clases de LINQ to SQL en la aplicación mediante esta clase, se debe descargar el RIA Services Toolkit

3.4.2 ASEGURAR UNA SOLUCIÓN DE RIA SERVICIOS

Para asegurarse de que su aplicación responde a las preocupaciones de seguridad asociadas con la exposición de un servicio de dominio, debe considerar cuidadosamente cómo se implementa el servicio de dominio.

3.4.2.1 SEGURIDAD DE LOS SERVICIOS RIA WCF

Al exponer un servicio de dominio mediante la aplicación de la `EnableClientAccessAttribute` atributo, el servicio de dominio está disponible para todos en la red donde se expone. No se puede suponer que la aplicación cliente es la única aplicación que tendrá acceso al servicio de dominio.

Esta consideración es particularmente importante en una red pública. Sin embargo, también es importante en una red restringida, como una red corporativa de datos sensibles cuando se está expuesto.

3.4.2.2 LISTA DE VERIFICACIÓN DE SEGURIDAD

Para garantizar el uso seguro de un servicio de dominio, se debe tener en cuenta las siguientes orientaciones.

1. Minimizar los datos y las operaciones expuestas por un servicio de dominio. Esta es la primera línea de defensa contra la divulgación de información y denegación de servicio.
 - a. Exponer sólo aquellas entidades que son necesarios por el cliente. Este enfoque puede requerir que la lógica de servidor independiente y validación de la lógica del cliente y de validación, si se le permite reducir el número de entidades expuestas.
 - b. Usted puede utilizar el ExcludeAttribute atributo o modelos de presentación para reducir los datos que está disponible a un cliente
 - c. Métodos requieren de consulta para tomar parámetros que son necesarios en la aplicación, en lugar de confiar en las capacidades de filtrado de datos en LINQ⁸.
 - d. Crear los métodos de consulta que sólo proporcionan los datos necesarios para los escenarios específicos de su aplicación. Este enfoque significa que se puede proporcionar múltiples métodos de consulta que las porciones retorno de los datos en lugar de un método de consulta que devuelve todos los datos.
 - e. Filtrar los datos para proporcionar sólo los datos normalmente se requiere para su aplicación.

⁸ LINQ: Lenguaje Integrado de Consulta que es un conjunto de extensiones de .NET Framework

- f. Restringir el número de resultados que pueden ser devueltos por una consulta para minimizar la sobrecarga accidental o deliberada del servidor. Se utiliza el ResultLimit propiedad en la QueryAttribute para estrangular el número de resultados que se pueden devolver.
 - g. Minimizar el número de operaciones de cada entidad expuestas.
 - h. Siempre que sea posible, se utiliza el nombre actualizar los métodos que restringen el cual los miembros se pueden actualizar.
2. Restringir el acceso de datos y operación para los usuarios autenticados y usuarios en roles específicos.
- a. Evitar el acceso anónimo siempre que sea posible mediante el RequiresAuthenticationAttribute atributo. Cuando se debe permitir el acceso anónimo, límite que hasta el más mínimo conjunto de servicios de dominio y el conjunto más pequeño de operaciones dentro de los servicios de dominio.
 - b. Añadir la operación específica de RequiresRoleAttribute atributo siempre que sea posible. Considerar la posibilidad de cada operación por separado en un servicio de dominio.
 - c. Considerar el uso de la AuthorizationContext propiedad para proporcionar un modelo de autorización personalizado.
 - d. El tratamiento de los datos enviados por un cliente como sospechoso. Un cliente malicioso (incluso uno que sea autenticado y autorizado) puede proporcionar valores alterados de los valores actuales y originales en un conjunto de cambios. Su lógica de la aplicación no debe asumir que estos valores son dignos de confianza.
3. Utilizar el protocolo https para la autenticación de formularios. El envío de contraseñas en texto claro es una vulnerabilidad significativa, pero puede ser mitigado mediante https.

4. Exponer el número mínimo de puntos finales. De forma predeterminada, RIA Services crear un extremo de binarios para un servicio de dominio. Agregar puntos finales adicionales sólo si tiene clientes que específicamente la necesidad de puntos finales. Deshabilitar todos los extremos que no están en uso

3.4.3 HERRAMIENTAS Y DOCUMENTACIÓN

La WCF RIA requeriré de varios programas de requisitos previos, tales como Visual Studio 2010 y el tiempo de ejecución de Silverlight para desarrolladores y SDK, ser instalado y configurado correctamente, además de servicios WCF RIA y el Fondo de Operaciones RIA Services Toolkit para trabajar a fondo los tutoriales y cómo temas. También requieren la instalación y configuración de SQL Server 2008 R2 Express con Advanced Services y la instalación de la base de datos OLTP AdventureWorks y LT.

3.5 PROGRAMACION EXTREMA XP

3.5.1 HISTORIA

La programación extrema o eXtreme Programming (XP) es un enfoque de la ingeniería de software formulado por Kent Beck, autor del primer libro sobre la materia, Extreme Programming Explained: Embrace Change (1999). Es el más destacado de los procesos ágiles de desarrollo de software. Al igual que éstos, la programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad.

Los defensores de XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. Creen que ser capaz de adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto es una aproximación mejor y más realista que intentar definir todos los requisitos al comienzo del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos.

3.5.2 INTRODUCCION

Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo.

XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

3.5.3 ¿QUÉ ES PROGRAMACIÓN EXTREMA O XP?

- ✓ Metodología liviana de desarrollo de software
- ✓ Conjunto de prácticas y reglas empleadas para desarrollar software
- ✓ Basada en diferentes ideas acerca de cómo enfrentar ambientes muy cambiantes
- ✓ Originada en el proyecto C3 para Chrysler
- ✓ En vez de planificar, analizar y diseñar para el futuro distante, hacer todo esto un poco cada vez, a través de todo el proceso de desarrollo

3.5.4 OBJETIVOS.

- ✓ Establecer las mejores prácticas de Ingeniería de Software en los desarrollo de proyectos.
- ✓ Mejorar la productividad de los proyectos.
- ✓ Garantizar la Calidad del Software desarrollando, haciendo que este supere las expectativas del cliente.

3.5.5 CONTEXTO XP

- ✓ Cliente bien definido
- ✓ Los requisitos pueden (y van a) cambiar
- ✓ Grupo pequeño y muy integrado (máximo 12 personas)
- ✓ Equipo con formación elevada y capacidad de aprender

3.5.6 CARACTERÍSTICAS XP

- ✓ Metodología basada en prueba y error
- ✓ Fundamentada en Valores y Prácticas
- ✓ Expresada en forma de 12 Prácticas (Conjunto completo). Se soportan unas a otras.

3.5.7 VALORES XP

- ✓ Simplicidad XP propone el principio de hacer la cosa más simple que pueda funcionar, en relación al proceso y la codificación. Es mejor hacer hoy algo simple, que hacerlo complicado y probablemente nunca usarlo mañana.
- ✓ Comunicación Algunos problemas en los proyectos tienen origen en que alguien no dijo algo importante en algún momento. XP hace casi imposible la falta de comunicación.
- ✓ Realimentación Retralimentación concreta y frecuente del cliente, del equipo y de los usuarios finales da una mayor oportunidad de dirigir el esfuerzo eficientemente.
- ✓ Coraje El coraje (valor) existe en el contexto de los otros 3 valores.(si funciona...mejóralo)

3.5.8 EL ESTILO XP

- ✓ Está orientada hacia quien produce y usa el software
- ✓ Reduce el costo del cambio en todas las etapas del ciclo de vida del sistema.
- ✓ Combina las que han demostrado ser las mejores prácticas para desarrollar software, y las lleva al extremo.

3.5.9 PRÁCTICAS BÁSICAS DE LA PROGRAMACIÓN EXTREMA

Para que todo esto funcione, la programación extrema se basa en doce "prácticas básicas" que deben seguirse al pie de la letra. Dichas prácticas están definidas para facilitar el desarrollo de aplicaciones⁹⁹

- ✓ **Equipo completo:** Forman parte del equipo todas las personas que tienen algo que ver con el proyecto, incluido el cliente y el responsable del proyecto.
- ✓ **Planificación:** Se hacen las historias de usuario y se planifica en qué orden se van a hacer y las mini-versiones. La planificación se revisa continuamente.
- ✓ **Test del cliente:** El cliente, con la ayuda de los desarrolladores, propone sus propias pruebas para validar las mini-versiones.
- ✓ **Versiones pequeñas:** Las mini-versiones deben ser lo suficientemente pequeñas como para poder hacer una cada pocas semanas. Deben ser versiones que ofrezcan algo útil al usuario final y no trozos de código que no pueda ver funcionando.
- ✓ **Diseño simple:** Hacer siempre lo mínimo imprescindible de la forma más sencilla posible. Mantener siempre sencillo el código.
- ✓ **Pareja de programadores:** Los programadores trabajan por parejas (dos delante del mismo ordenador) y se intercambian las parejas con frecuencia (un cambio diario).

⁹⁹ www.xprogramming.com/xpmag/whatisxp.htm

- ✓ **Desarrollo guiado por las pruebas automáticas:** Se deben realizar programas de prueba automática y deben ejecutarse con mucha frecuencia. Cuantas más pruebas se hagan, mejor.
- ✓ **Integración continua:** Deben tenerse siempre un ejecutable del proyecto que funcione y en cuanto se tenga una nueva pequeña funcionalidad, debe recompilarse y probarse. Es un error mantener una versión congelada dos meses mientras se hacen mejoras y luego integrarlas todas de golpe. Cuando falle algo, no se sabe qué es lo que falla de todo lo que hemos metido.
- ✓ **El código es de todos:** Cualquiera puede y debe tocar y conocer cualquier parte del código. Para eso se hacen las pruebas automáticas.
- ✓ **Normas de codificación:** Debe haber un estilo común de codificación (no importa cual), de forma que parezca que ha sido realizado por una única persona.
- ✓ **Metáforas:** Hay que buscar unas frases o nombres que definan cómo funcionan las distintas partes del programa, de forma que sólo con los nombres se pueda uno hacer una idea de qué es lo que hace cada parte del programa. Un ejemplo claro es el "recolector de basura" de java. Ayuda a que todos los programadores (y el cliente) sepan de qué estamos hablando y que no haya mal entendidos.
- ✓ **Ritmo sostenible:** Se debe trabajar a un ritmo que se pueda mantener indefinidamente. Esto quiere decir que no debe haber días muertos en que no se sabe qué hacer y que no se deben hacer un exceso de horas otros días. Al tener claro semana a semana lo que debe hacerse, hay que trabajar duro en ello para conseguir el objetivo cercano de terminar una historia de usuario o mini-versión.

3.5.10 VENTAJAS Y DESVENTAJAS DE EXTREME PROGRAMMING

Ventajas:

- ✓ Programación organizada.
- ✓ Menor tasa de errores.
- ✓ Satisfacción del programador.

Desventajas:

- ✓ Es recomendable emplearlo solo en proyectos a corto plazo.
- ✓ Altas comisiones en caso de fallar.

3.5.11 LOS ROLES EN XP

A diferencia de las metodologías tradicionales, en XP más que asignar roles que denotan estatus o importancia, esta metodología confía en la sinergia que pueda existir entre las personas que conforman el equipo XP, para crear éxito en el proyecto.

Es así, como los roles en XP no son estáticos; por ejemplo, en un momento determinado puedes ser un desarrollador, mientras que en otro momento serás el encargado del seguimiento del proyecto (*tracker*).

Alcanzar las metas y construir lo que el cliente necesita es el objetivo primordial del equipo XP, y para esto se vale de una colaboración efectiva que se asume como compromiso por parte de cada uno de los integrantes, resaltando que en XP, las personas en sí son más importantes para la consecución del proyecto que el mismo proceso.

Cabe recalcar que XP si sigue un proceso, pero que el empoderamiento a personas, guiados por buenas prácticas de desarrollo, maximizan los resultados.

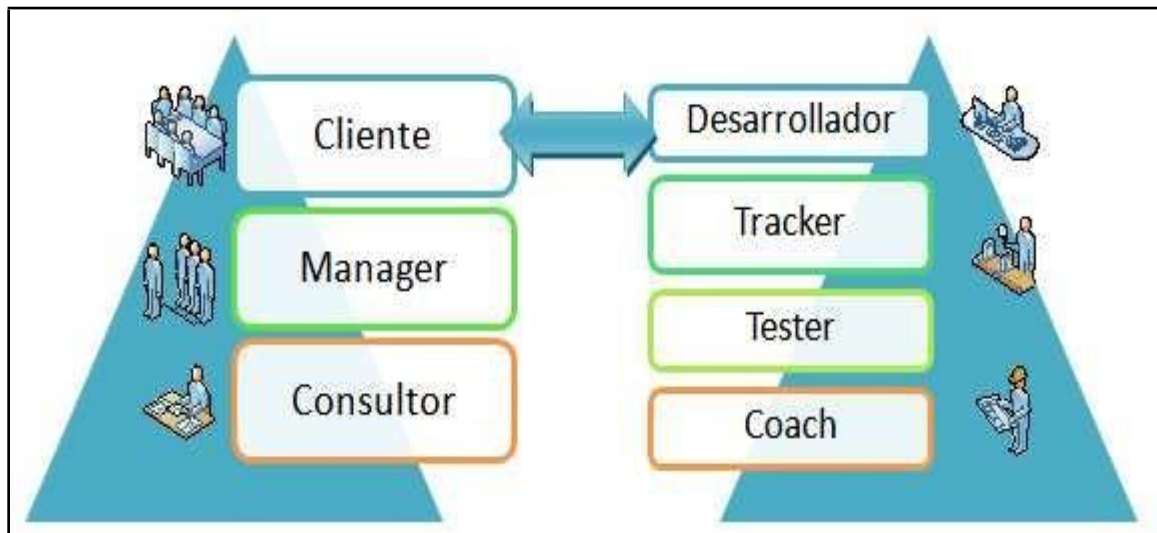


FIGURA 3.9: Roles que existen en un proyecto XP.

3.5.11.1 CLIENTE

El cliente define la visión y metáfora de nuestro proyecto, el valor del negocio, y la dirección que el equipo de desarrollo debe tomar a medida que progresa el proyecto. El cliente es una persona o grupo de personas que tienen un sólido conocimiento del proceso de negocio actual y clarificará los problemas que el nuevo sistema ha de resolver. Entre algunas tareas del cliente, están:

- ✓ Definir el valor del negocio.
- ✓ Escribir historias de usuario.
- ✓ Escribir o especificar las pruebas de aceptación.
- ✓ Realizar conjuntamente con el equipo de desarrollo la planificación de las entregas.

3.5.11.2 DESARROLLADOR (PROGRAMADOR)

Para el desarrollador, no debe existir ningún obstáculo que no le permita el desarrollo de código, pues con el código es como el desarrollador entrega las historias de usuario al cliente. En otras palabras, los desarrolladores en XP tienen que hacer lo siguiente:

- ✓ Estimar las historias de usuario.
- ✓ Realizar prototipos desechables cuando sea necesario.
- ✓ Escribir pruebas unitarias (usando herramientas para pruebas).
- ✓ Escribir código.
- ✓ Participar en reuniones para la planificación.
- ✓ Participar en reuniones diarias con el equipo.

3.5.11.3 GESTOR (MANAGER)

El gestor o “Big Boss”, es la persona encargada de la gestión del proyecto internamente y hacia los clientes. Un gestor se ocupa en aclarar obstáculos e interferencias que obstruya el trabajo del equipo de desarrollo (proveniente de los clientes o de la gestión del equipo). Es capaz de mantenerse en un nivel alto de entendimiento del negocio, pero a la vez está alineado con los valores del equipo.

3.5.11.4 CONSULTOR

Por lo general es un experto externo que solventa inquietudes sobre temas técnicos o de negocios.

3.5.11.5 ENCARGADO DEL SEGUIMIENTO (TRACKER)

El encargado del seguimiento es el responsable de reunir métricas tales como historias de usuario o tareas completadas y comunicar el progreso al equipo de desarrollo.

El trabajo del tracker es determinar justamente que métricas son significativas para mostrar el estado del proyecto. De manera general, el tracker debe utilizar el menor número de indicativos posibles, entre ellos está: el estado del último build, el número de historias de usuario completados, la velocidad de desarrollo, entre otros.

3.5.11.6 ENCARGADO DE PRUEBAS (TESTER)

Es el encargado de realizar las pruebas de aceptación conjuntamente con el cliente, asegurando que el sistema funcione como se esperaba. La ventaja del encargado de pruebas está en la probabilidad de no tener muchos inconvenientes con respecto a errores en el sistema, puesto que el equipo de desarrollo escribe código de calidad que anteriormente ya ha pasado todas sus pruebas unitarias exitosamente.

3.5.11.7 ENTRENADOR (COACH)

El concepto de entrenador, es primordialmente de ser la persona encargada de facilitar la comunicación dentro del equipo de desarrollo, asegurando que se estén siguiendo los lineamientos que son necesarios en un desarrollo XP.

CAPÍTULO IV

4 DESARROLLO DEL APLICATIVO

4.1 PLANIFICACIÓN

4.1.1 HISTORIAS DE USUARIO

Las historias de usuario, son la técnica utilizada en XP para especificar la funcionalidad del sistema desde el punto de vista del cliente. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales.

Cabe señalar que a diferencia de los casos de uso, las historias de usuario son descritas por el cliente en un lenguaje no técnico, obviando detalles de implementación y algoritmos, y aunque su propósito sea similar al de los casos de uso, ambos no deben ser confundidos.

El tratamiento de las historias de usuario es muy dinámico y flexible, en cualquier momento las historias de usuario pueden romperse, reemplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en semanas.

El cliente participa activamente en la redacción de las historias de usuario, de hecho lo normal es que sea el cliente quien las redacte, sin embargo y para evitarles excesivas molestias se llevaron a cabo reuniones con el cliente en donde fue el equipo de desarrollo quien redactó las historias de usuario, de acuerdo a lo que el cliente necesitaba.

Cabe recalcar que pueden existir historias de usuario que son conocidas como “de valor de negocio cero”, ya que a primera vista no generan ningún beneficio para el cliente pero que son indispensables para el desarrollo del sistema. Tales historias de usuario pueden consistir en la instalación del ambiente de desarrollo, creación del esquema inicial de la base de datos, creación de usuarios a aplicativos necesarios para el desarrollo, entre otros.

4.1.1.1 IDENTIFICACIÓN DE HISTORIAS DE USUARIO

Para nuestro proyecto se describen las historias de usuario recopiladas por el grupo de tesis y nuestro cliente, las mismas que luego de ser analizadas han sido agrupadas en las respectivas categorías que se describen a continuación:

- *Gestión de Usuarios.*- Es el componente del sistema que se encarga del ingreso y validación del usuario que usará la aplicación, parámetros para ventas y comisiones, como también de la creación, modificación y eliminación de usuarios.
- *Gestión de Planes Telefónicos.*- Este componente realiza el despliegue de la información del suscriptor al que pertenece el plan telefónico. Adicionalmente este componente es el encargado de realizar el diagnóstico de varios planes telefónicos. Es útil para los usuarios que desean obtener de forma masiva información del estado operacional de los planes contratados
- *Gestión de Ventas:* Permite obtener los valores de las comisiones que cada vendedor debe obtener por la venta de planes y equipos telefónicos.
- *Gestión de Bodega:* Permite obtener el control de equipos pertenecientes a REDPRIMCELL CÍA. LTDA y a equipos comprados fuera de la misma
- *Reportes.*- Permite obtener reportes que indican el estado operacional de las Planes Telefónicos de acuerdo a los diagnósticos que se han guardado en la base de datos.

Con esta premisa, en la siguiente sección describimos cada una de las historias de usuario que conforman a nuestro sistema.

4.1.1.1.1 Gestión de Usuarios

En las siguientes tablas, se detallan las historias de usuario correspondientes al componente del sistema "Gestión de Usuarios".

TABLA 4-1: Historia de Usuario "Ingreso del Usuario al Sistema"

Historia N°001	Ingreso del Usuario al Sistema
<p>Descripción: El ingreso al sistema debe realizarse por medio de una ventana de ingreso en la cual se especifique el nombre de usuario, la contraseña y el rol asignado.</p> <p>Debe existir algún mecanismo de control que no permita que el usuario pueda pegar caracteres en el campo de contraseña, como también no debe permitirse poder copiar la clave del usuario hacia otra fuente.</p>	

Fuente: [Propia]

TABLA 4-2: Historia de Usuario "Creación de Usuarios"

Historia N°002	Creación de Usuarios
<p>Descripción: Los usuarios de la aplicación deben estar asignados a uno de los siguientes roles: Administrador, Vendedor, Supervisor, Operador o Técnico. Dichos usuarios pueden ser creados únicamente por el usuario Administrador, siendo estos guardados en la base de datos.</p> <p>Cabe señalar que cada usuario tiene ciertos privilegios asignados.</p>	

Fuente: [Propia]

TABLA 4-3: Historia de Usuario "Eliminación de Usuarios"

Historia N°003	Eliminación de Usuarios
<p>Descripción: El usuario Administrador puede eliminar en cualquier momento a un usuario que sea del tipo Vendedor, Supervisor, Operador o Técnico.</p> <p>Una vez que se elimine al usuario, se debe actualizar la base de datos, para reflejar los cambios que se han realizado.</p>	

Fuente: [Propia]

TABLA 4-4: Historia de Usuario "Modificación de Usuarios"

Historia N°004	Modificación de Usuarios
<p>Descripción: El usuario Administrador puede modificar información del usuario, como por ejemplo los nombres, apellidos y teléfono del usuario.</p> <p>Una vez que se modifique al usuario, se debe actualizar la base de datos, para reflejar los cambios que se han realizado.</p>	

Fuente: [Propia]

4.1.1.1.2 Gestión de Planes Telefónicos

A continuación, se detallan las historias de usuario correspondientes al componente del sistema "Gestión Individual de Planes Telefónicos".

TABLA 4-5: Historia de Usuario "Ingreso de Planes Telefónicos"

Historia N°005	Ingreso de Planes Telefónicos
Descripción: Registrar los datos correspondientes a los planes telefónicos como son: código, descripción, categoría, precios, porcentajes, siendo estos guardados en la base de datos. Los datos de categorías deben estar registrados previamente	

Fuente: [Propia]

TABLA 4-6: Historia de Usuario "Ingreso de Solicitud de Plan"

Historia N°006	Ingreso de Solicitud de Plan
Descripción: Los datos del Plan solicitado deben registrarse por parte del Vendedor, los datos que deben incluir son: Datos del cliente, fecha de registro, tipo de plan y otros datos que faciliten el proceso para calcular comisiones, siendo estos guardados en la base de datos. Cabe señalar que los clientes deben estar registrados previamente.	

Fuente: [Propia]

TABLA 4-7: Historia de Usuario "Aprobación Solicitud"

Historia N°007	Aprobación Solicitud
Descripción: Los datos del Plan solicitado previamente ingresados, son revisados y aprobados por un Usuario tipo Supervisor, con esta información el plan solicitado por el cliente o suscriptor, es activado con las características solicitadas. Esta información queda registrada en la base de datos para que un Supervisor o Administrador, pueda revisar para su respectivo análisis	

Fuente: [Propia]

TABLA 4-8: Historia de Usuario "Despliegue de Información del Suscriptor"

Historia N°008	Despliegue de Información del Suscriptor
Descripción: La información del suscriptor es importante para los usuarios Operadores que necesitan identificar datos como nombres, dirección, ciudad, provincia y estado de plan telefónico. Para obtener dicha información, es necesario acceder a la base de datos de REDPRIMCELL CÍA. LTDA.	

Fuente: [Propia]

TABLA 4-9: Historia de Usuario "Consulta de Información del Plan Telefónico"

Historia N°009	Consulta de Información del Plan Telefónico Telefónica
<p>Descripción: Si el usuario desea, solamente se puede consultar información acerca de la línea telefónica es decir información del suscriptor.</p> <p>Para realizar otra consulta, el sistema deberá permitir esta opción, limpiando los eventos y elementos gráficos que ya fueron utilizados.</p>	

Fuente: [Propia]

TABLA 4-10: Historia de Usuario "Despliegue de la Información Obtenida de la Activación del Plan"

Historia N°010	Despliegue de la Información Obtenida del Plan de Línea
<p>Descripción: Cuando se realiza la activación, se obtiene información proveniente de la operadora correspondiente a la prueba de línea que se realizó. Esta información que luego será interpretada por el sistema para obtener el diagnóstico, es de utilidad para los técnicos quienes interpretan esta información para la toma de decisiones de acuerdo a su nivel de experiencia en el servicio técnico.</p>	

Fuente: [Propia]

TABLA 4-11: Historia de Usuario "Despliegue del Diagnóstico en un DBGRID"

Historia N°011	Despliegue de Diagnostico en un DBGRID
<p>Descripción: El componente de diagnósticos masivos deberá presentarle al usuario una tabla en donde se visualice el resultado luego de haber realizado el diagnóstico masivo. Esta tabla debe poder organizarse por sus campos, es decir ordenarse alfabéticamente por cualquier campo que el usuario escoja.</p>	

Fuente: [Propia]

4.1.1.1.3 Gestión de Ventas

A continuación, se detallan las historias de usuario correspondientes al componente del sistema "Gestión de Ventas".

TABLA 4-12: Historia de Usuario "Determinación de Valores de Ventas"

Historia N°012	Determinación de Valores de Ventas
Descripción: Todos los planes son vendidos a clientes, los mismos que tienen valores, fecha, los mismos que deben conocerse por vendedor o por tipo de plan. Los tipos de planes, están previamente registrados en la base de datos.	

Fuente: [Propia]

TABLA 4-13: Historia de Usuario "Determinación de Comisiones"

Historia N°013	Determinación de Comisiones
Descripción: A nivel operacional, un plan telefónico está activado, luego del diagnóstico cumplen con los valores que son permitidos en REDPRIMCELL CÍA. LTDA. Estos valores son las correspondientes a comisiones de los vendedores, los que conjuntamente con el equipo deben guardarse en la base de datos.	

Fuente: [Propia]

4.1.1.1.4 Gestión de Bodega

TABLA 4-14: Historia de Usuario "Creación de Equipos"

Historia N°014	Creación de Equipos
<p>Descripción: Los equipos deben ser creados, con la información necesaria como: código, nombre, fecha caducidad, tipo de equipo. Dichos equipos pueden ser creados únicamente por el usuario Administrador, siendo estos guardados en la base de datos.</p> <p>Los tipos de equipos están previamente registrados en la base de datos</p>	

Fuente: [Propia]

TABLA 4-15: Historia de Usuario "Ingreso de Equipos"

Historia N°015	Ingreso de Equipos
<p>Descripción: Los equipos son ingresados de acuerdo a las necesidades de REDPRIMCELL CÍA. LTDA, dependiendo del número de planes activados, los equipos deben ingresarse indicando el número de unidades y la fecha de compra, fecha caducidad, la misma que luego será considerada prioritaria para su venta</p>	

Fuente: [Propia]

TABLA 4-16: Historia de Usuario "Control Stock Equipos"

Historia N°016	Control Stock Equipos
<p>Descripción: Los equipos son ingresados con un plan de un cliente específico, previamente aprobado y activado. Los stocks de los equipos deben ser controlados en base a la fecha de caducidad, de esta manera los primeros equipos que deben ser vendidos son los de mayor fecha de compra.</p>	

Fuente: [Propia]

4.1.1.1.5 Reportes

A continuación, se detallan las historias de usuario correspondientes al componente del sistema "Reportes".

TABLA 4-17: Historia de Usuario "Reporte de Ventas por Vendedor"

Historia N°017	Reporte de Ventas por Vendedor
Descripción: Cuando un vendedor realiza ventas de planes y equipos, estas acciones son guardadas en la base de datos de modo que en cualquier instante se pueda auditar quien está haciendo uso de la aplicación. El cliente solicitó reporte por usuario, de modo que se muestre que usuario ha realizado ventas y en qué medida los realizó.	

Fuente: [Propia]

TABLA 4-18: Historia de Usuario "Reportes por Fechas"

Historia N°018	Reportes por Fechas
Descripción: El cliente necesita reportes que se puedan obtener por rangos de fechas. Es decir, el usuario especificará una fecha inicial y una fecha final, y el sistema obtendrá todos los reportes de ventas que fueron efectuados en dicho rango.	

Fuente: [Propia]

TABLA 4-19: Historia de Usuario " Reportes de Comisión de Ventas"

Historia N°019	Reportes de Comisión de Ventas
Descripción: Cuando el usuario haya escogido los parámetros para la generación de comisiones por ventas, el sistema deberá generar el reporte solicitado y presentarlo con los valores generados por las ventas realizadas por REDPRIMCELL CÍA. LTDA.	

Fuente: [Propia]

TABLA 4-20: Historia de Usuario "Reporte Equipos Vendidos"

Historia N°020	Reporte Equipos Vendidos
Descripción: Los supervisores necesitan conocer el número de equipos vendidos ya sea propios de REDPRIMCELL CÍA. LTDA o equipos vendidos por otros distribuidores. Esto facilitara el control interno de stock y determinar que productos son necesarios venderlos o adquirir	

Fuente: [Propia]

4.1.2 PLAN DE ENTREGAS

De acuerdo a las historias de Usuario consideradas en el sistema se ha realizado el siguiente plan de entregas, el cual tiene las iteraciones para cada tarea. Este plan se encuentra desarrollado tomando en cuenta el esfuerzo y la prioridad de cada Historia de Usuario.

TABLA 4-21: Prioridad Historias de usuario

Historia	Nombre	Prioridad	Esfuerzo
1	Ingreso del Usuario al Sistema	Alta	4
2	Creación de Usuarios	Media	2
3	Eliminación de Usuarios	Baja	1
4	Modificación de Usuarios	Media	2
5	Ingreso de Planes Telefónicos	Alta	4
6	Ingreso de Solicitud de Plan	Media	2
7	Aprobación Solicitud	Alta	3
8	Despliegue de Información del Suscriptor	Media	4
9	Consulta de Información del Plan Telefónico	Media	4
10	Despliegue de la Información Obtenida del Plan	Media	3
11	Despliegue del Diagnóstico en un DBGRID	Baja	2
12	Determinación de Valores de Ventas	Media	3
13	Determinación de Comisiones	Alta	3
14	Creación de Equipos	Media	2
15	Ingreso de Equipos	Baja	2
16	Control Stock Equipos	Alta	4
17	Reporte de Ventas por Vendedor	Media	2
18	Reportes por Fechas	Media	2
19	Reportes de Comisión de Ventas	Alta	4
20	Reporte Equipos Vendidos	Media	3

Fuente: [Propia]

Evaluando cada historia de usuario se puede determinar en qué iteración se desarrollará la misma, a partir de esto se determina las fechas de inicio y fin para las entregas. A continuación se muestra la tabla de entregas

TABLA 4-22: Fechas de entrega historias de usuario

Historia	Nombre	Prioridad	F. Inicio	F. Fin
1	Ingreso del Usuario al Sistema	Alta	22-04-2013	27-04-2013
2	Creación de Usuarios	Media	29-04-2013	04-05-2013
3	Eliminación de Usuarios	Baja	06-05-2013	11-05-2013
4	Modificación de Usuarios	Media	13-05-2013	18-05-2013
5	Ingreso de Planes Telefónicos	Alta	20-05-2013	25-05-2013
6	Ingreso de Solicitud de Plan	Media	27-05-2013	01-06-2013
7	Aprobación Solicitud	Alta	03-06-2013	05-06-2013
8	Despliegue de Información del Suscriptor	Media	06-06-2013	08-06-2013
9	Consulta de Información del Plan Telefónico	Media	10-06-2013	12-06-2013
10	Despliegue de la Información Obtenida del Plan	Media	13-06-2013	15-06-2013
11	Despliegue del Diagnóstico en un DBGRID	Baja	17-06-2013	19-06-2013
12	Determinación de Valores de Ventas	Media	20-06-2013	22-06-2013
13	Determinación de Comisiones	Alta	24-06-2013	26-06-2013
14	Creación de Equipos	Media	27-06-2013	29-06-2013
15	Ingreso de Equipos	Baja	01-07-2013	03-07-2013
16	Control Stock Equipos	Alta	04-07-2013	06-07-2013
17	Reporte de Ventas por Vendedor	Media	08-07-2013	10-07-2013
18	Reportes por Fechas	Media	11-07-2013	13-07-2013
19	Reportes de Comisión de Ventas	Alta	15-07-2013	17-07-2013
20	Reporte Equipos Vendidos	Media	18-07-2013	20-07-2013

Fuente: [Propia]

4.1.3 VELOCIDAD DEL EQUIPO DE DESARROLLO

En XP, la velocidad de desarrollo se refiere a la rapidez con la que el equipo de desarrollo puede trabajar implementando historias de usuario que han sido estimadas utilizando el tiempo ideal. Cuando se refiere a la velocidad de desarrollo, también se utiliza el término de esfuerzo ideal para denotar el tiempo que le es posible asignar a un desarrollador para implementar una funcionalidad del sistema en un tiempo determinado.

4.1.4 ITERACIONES POR CADA ENTREGA

Una vez establecida la velocidad del equipo de desarrollo, se determina el número de iteraciones que se llevarán a cabo para la entrega del software.

En la sección 2.1.2, se obtuvo el total de semanas ideales (25.6 semanas) que serían necesarias para implementar la solución propuesta por el equipo de desarrollo. Tomando en cuenta ese total, se procede a calcular el número de iteraciones de la siguiente manera:

Semanas ideales totales para implementar la solución: 25.6 semanas

Velocidad del equipo de desarrollo al mes: 8 semanas

4.2 DISEÑO

4.2.1 METÁFORA

REDPRIMCELL CÍA. LTDA es un distribuidor autorizado para la comercialización de servicios de telefonía móvil, estos servicios en su mayoría están dedicados a planes que generan comisiones para los vendedores.

Estas comisiones permiten que se realicen cálculos de acuerdo a parámetros asignados por la empresa, calculados y pagados cada mes.

Se debe tener un reporte de todas las ventas realizadas es decir los valores generados por comisiones cada cierto periodo, conociendo su fecha inicial y final.

Todas estas tareas deben ser realizadas por un administrador o supervisor.

4.2.2 CRC. TARJETAS DE RESPONSABILIDAD COLABORACIÓN.

Usaremos una tarjeta de Responsabilidad y Colaboración para una Solicitud de Plan que es la historia de usuario la cual está en estudio. En esta historia de usuario se necesita realizar 2 CRC, ya que se refiere a una solicitud de acuerdo al análisis realizado se trata de una tabla Maestro Detalle, su descripción a continuación:

NOMBRE DE CLASE: SOLICITUD_PLAN
RESPONSABILIDADES
CAMPOS SO_ID SO_FECHA SO_DESCRIPCION SO_TIPOLINEA SO_NUMSERVICIO
METODOS Ingresar_Datos() Ingresar_Solicitud(int id, date fecha, string id_v, string id_ta, string id_cl) Modificar_Solicitud(int id) Eliminar_Solicitud(int id)
COLABORACIÓN: setId(int id) setFecha(date fecha) setDescripcion(String descripción) setTipoLinea(String tipo) setNumeroServicio(String numero)

FIGURA 4.1: Tarjeta CRC para Solicitud de Plan

Fuente: [Propia]

NOMBRE DE CLASE: DETALLE_SOLICITUD
RESPONSABILIDADES
CAMPOS DS_ID DS_CANTIDAD DS_VALOR PR_ID SO_ID
METODOS Ingresar Detalle(int id, int cantidad, double valor)
COLABORACIÓN: setId(int id) setCantidad(int cantidad) setValor(double valor)

FIGURA 4.2: Tarjeta CRC para Solicitud de Plan

Fuente: [Propia]

4.3 DESARROLLO

4.3.1 DISPONIBILIDAD DEL CLIENTE

4.3.1.1. DIAGRAMA ENTIDAD-RELACIÓN

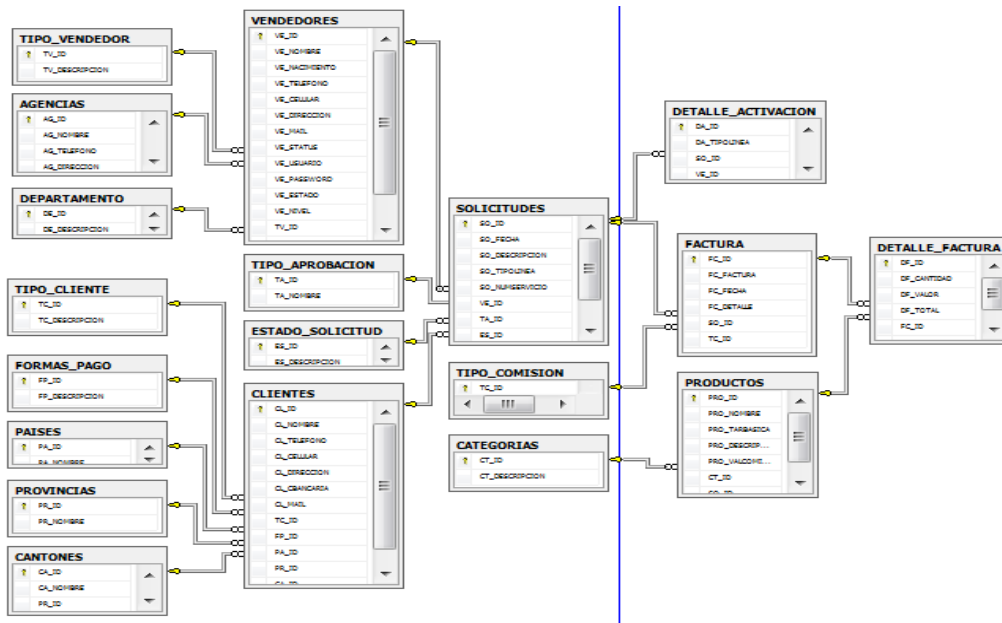


FIGURA 4.3: Diagrama Entidad - Relación

Fuente: [Propia]

4.3.1.2 DIAGRAMA DE CLASES

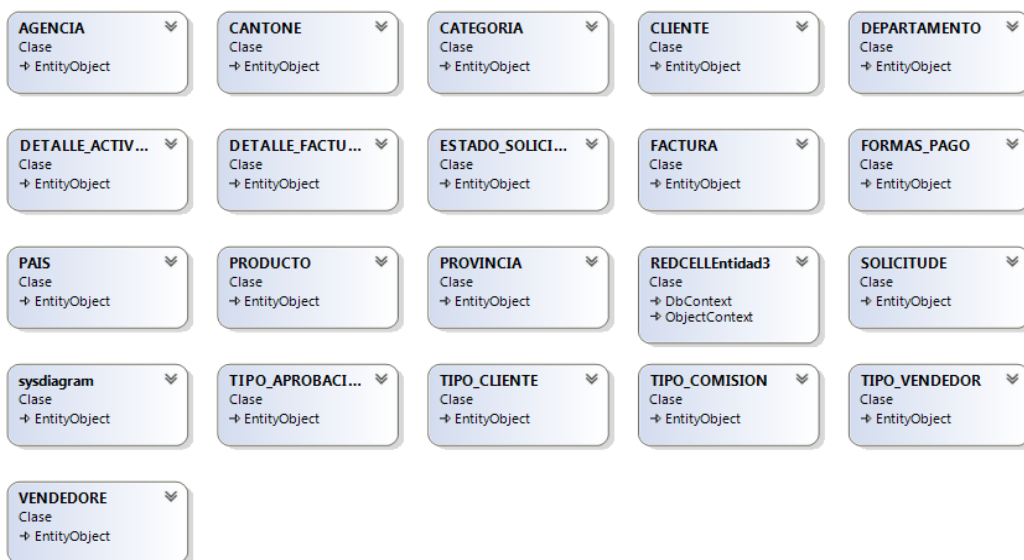


FIGURA 4.4: Diagrama de clases

Fuente: [Propia]

4.3.2 UNIDAD DE PRUEBAS

4.3.2.1 ESCENARIO DE PRUEBA DE “INGRESO DE PLANES TELEFÓNICOS”

En esta historia de usuario, se ingresa todos los datos correspondientes a un plan telefónico, tenemos los siguientes: código, descripción, tipo de plan, valor comisión. El sistema debe controlar que los datos estén correctamente ingresados y **validados**.

TABLA 4-23: Prueba de Aceptación para el escenario "Ingreso de Planes Telefónicos"

<p>Precondiciones:</p> <p>El usuario debe ingresar nuevos planes con el tipo correspondiente, además puede modificar los planes existentes, ciertos parámetros pueden variar de acuerdo a las políticas de la empresa</p>
<p>Entrada:</p> <p>El usuario accede a la aplicación con el nombre, usuario y rol asignado.</p> <p>El usuario escoge el componente “Planes Telefónicos” de la aplicación.</p> <p>El usuario ingresa todos los campos que identifiquen al plan</p> <p>El usuario presiona el botón <i>Guardar</i>.</p>
<p>Resultado esperado 1:</p> <p>El sistema verifica la existencia del plan telefónico ingresado</p> <p>El sistema consulta la información solicitada en la base de datos y presenta la información de los planes existentes.</p>
<p>Resultado esperado 2:</p> <p>El sistema selecciona un plan específico y puede modificar la información cambiando ciertos parámetros para las ventas.</p>

Fuente: [Propia]

4.3.2.2 ESCENARIO DE PRUEBA DE “INGRESO DE SOLICITUD DE PLAN”

En esta historia de usuario, se ingresa los campos para una solicitud de un plan telefónico: código, fecha, descripción, identificación del vendedor, identificación del cliente. El sistema debe controlar y validar todos los datos para estén correctamente ingresados.

TABLA 4-24: Prueba de Aceptación para el escenario "Ingreso de Solicitud de Plan"

<p>Precondiciones:</p> <p>El usuario debe ingresar nuevos planes con el tipo correspondiente, además puede modificar los planes existentes, ciertos parámetros pueden variar de acuerdo a las políticas de la empresa</p>
<p>Entrada:</p> <p>El usuario accede a la aplicación con el nombre, usuario y rol asignado.</p> <p>El usuario escoge el componente “Planes Telefónicos” de la aplicación.</p> <p>El usuario ingresa todos los campos que identifiquen al plan</p> <p>El usuario presiona el botón <i>Guardar</i>.</p>
<p>Resultado esperado 1:</p> <p>El sistema verifica la existencia del plan telefónico ingresado</p> <p>El sistema consulta la información solicitada en la base de datos y presenta la información de los planes existentes.</p>
<p>Resultado esperado 2:</p> <p>El sistema selecciona un plan específico y puede modificar la información cambiando ciertos parámetros para las ventas.</p>

Fuente: [Propia]

4.3.2.3 ESCENARIO DE PRUEBA DE “APROBACIÓN SOLICITUD”

En esta historia de usuario, permite la búsqueda y verificación de los datos ingresados en las solicitudes de planes telefónicos. Una vez ingresadas las solicitudes, el sistema permite buscar un plan con su identificador o con la identificación del cliente y procede a activar para su posterior venta.

TABLA 4-25: Prueba de Aceptación para el escenario "Ingreso de Planes Telefónicos"

<p>Precondiciones:</p> <p>El usuario debe ingresar nuevos planes con el tipo correspondiente, además puede modificar los planes existentes, ciertos parámetros pueden variar de acuerdo a las políticas de la empresa</p>
<p>Entrada:</p> <p>El usuario accede a la aplicación con el nombre, usuario y rol asignado.</p> <p>El usuario escoge el componente “Planes Telefónicos” de la aplicación.</p> <p>El usuario ingresa todos los campos que identifiquen al plan</p> <p>El usuario presiona el botón <i>Guardar</i>.</p>
<p>Resultado esperado 1:</p> <p>El sistema verifica la existencia del plan telefónico ingresado</p> <p>El sistema consulta la información solicitada en la base de datos y presenta la información de los planes existentes.</p>
<p>Resultado esperado 2:</p> <p>El sistema selecciona un plan específico y puede modificar la información cambiando ciertos parámetros para las ventas.</p>

Fuente: [Propia]

4.3.2.4 ESCENARIO DE PRUEBA DE “DESPLIEGUE DE INFORMACIÓN DEL SUSCRIPTOR”

En esta historia de usuario, se despliega información del suscriptor referente a sus datos personales (nombres completos, dirección de domicilio, ciudad, provincia) además del estado plan telefónico (suspendido, conectado, entre otros).

El sistema debe controlar que al momento de consultar la información requerida, no existan datos redundantes, es decir que para un plan telefónico solo debería existir un suscriptor asignado. Además el sistema deberá verificar que el / los **números ingresados pertenezcan al servicio de telefonía claro.**

TABLA 4-26: Prueba de Aceptación para el escenario "Despliegue de Información del Suscriptor"

<p>Precondiciones:</p> <p>El usuario debe haber ingresado un código que identifique al plan contratado o con la identificación registrada en el sistema, la identificación puede ser la cédula de identidad o RUC.</p>
<p>Entrada:</p> <p>El usuario accede a la aplicación con el nombre, usuario y rol asignado.</p> <p>El usuario escoge el componente “Despliegue de información del Suscriptor” de la aplicación.</p> <p>El usuario hace clic en el campo <i>CódigoPlan</i> para ingresar el plan a ser consultado o también puede hacer clic en Documento de Identificación e ingresar la cédula o RUC.</p> <p>El usuario presiona el botón <i>Consultar</i>.</p>
<p>Resultado esperado 1:</p> <p>El sistema verifica la existencia del plan telefónico ingresado</p> <p>El sistema consulta la información solicitada en la base de datos y presenta la información de nombres, dirección, ciudad, provincia, y estado de corte, números telefónicos, equipos del suscriptor.</p>
<p>Resultado esperado 2:</p> <p>El sistema no encuentra dicho número telefónico en la base de datos y a continuación presenta un mensaje de error indicando justamente que no existe el Suscriptor.</p>

Fuente: [Propia] Escenario de Prueba de “Determinación de Valores de Ventas”

4.3.2.5 ESCENARIO DE PRUEBA DE “DETERMINACIÓN DE VALORES DE VENTAS”

Este escenario es de suma importancia ya que presenta las ventas realizadas mostrando los valores de los planes definidos por la empresa.

La prueba de aceptación será satisfactoria siempre y cuando los valores se desplieguen de manera concisa

TABLA 4-27: Prueba de Aceptación para el escenario "Determinación de valores de Ventas"

Precondiciones: El usuario debe haber realizado la aprobación de la solicitud y asignando la información de cada plan telefónico.
Entrada: El usuario pulsar en la opción Ver Valores de Ventas.
Resultado esperado: El sistema deberá presentar los valores de los planes vendidos y aprobados por la empresa.

4.3.3 TRABAJO EN PAREJAS

No se aplica para el proyecto, ya que es una investigación personal de trabajo de tesis.

4.3.4 PROPIEDAD COLECTIVA DEL CÓDIGO.

Al trabajar individualmente el código no se ha compartido durante todo el diseño y desarrollo del sistema, pero si existen sugerencias o cambios en el código cualquier persona relacionada con el software le puede modificar el código de tal forma que no es prescindible la presencia de una sola persona. Para que cualquier persona trabaje en el código se utilizó: .Net 2012 (C#), utilizando Silverlight 5

4.4 ENTREGA

Llegado a este punto, se ha implementado en su totalidad las iteraciones que conforman la entrega del “SISTEMA PARA LA GESTIÓN DE VENTAS Y COMISIONES EN LA EMPRESA REDPRIMCEL Cía. Ltda.” y el producto se encuentra listo para su puesta en producción.

Es así como se procede a realizar las actividades finales para la entrega: la implantación y evaluación del sistema.

4.4.1 IMPLANTACIÓN DEL SISTEMA

Al ser una metodología abierta, XP permite integrar métricas que mejoren la calidad del producto final por lo que se adoptó el concepto de implantación que propone la Métrica V3 (Ver ANEXO D), reflejando su uso en las siguientes secciones.

4.4.1.1 PLAN DE IMPLANTACIÓN

Para la implantación del sistema se deben tomar en cuenta los siguientes aspectos:

- ✓ La capacitación necesaria para la implantación, tanto a usuarios finales como al equipo que se encarga de realizar las pruebas de implantación y del sistema.
- ✓ La preparación de la infraestructura necesaria para la incorporación del sistema al entorno de producción.
- ✓ La configuración de todos los componentes, procedimientos manuales y automáticos asociados en la implantación.
- ✓ La realización de las pruebas de implantación del sistema.

4.4.1.2 EQUIPO DE TRABAJO

La Gerencia de REDPRIMCEL Cía. Ltda., en coordinación con la Desarrolladora decidió trabajar de la siguiente manera:

TABLA 4-28: Equipo de Trabajo para la Implantación del Sistema

CARGO	DEPARTAMENTO	RESPONSABLE
Gerencia de Agencia	Administrativo	Sra. Yesenia Azuero
Gerencia de Supervisión	Sucursales	Ing. Cristian Reinoso
Desarrollador del Sistema:	Equipo de Desarrollo	Srta. Verónica Usiña

Fuente: [Autor]

4.4.1.3 CAPACITACIÓN PARA LA IMPLANTACIÓN

La capacitación para la implantación del sistema en el ambiente de producción se realizó conjuntamente con los Administrativos quienes conocen todo acerca del sistema.

De la misma manera capacitaron a los Supervisores, en el funcionamiento del aplicativo para que ellos sean las personas quienes capaciten a los Vendedores, además se les facilitaron los manuales de instalación y de usuario respectivamente, mismos que previamente han sido revisados y autorizados por los directivos de REDPRIMCEL Cía. Ltda.

Esta capacitación se realizó en las instalaciones de REDPRIMCEL Cía. Ltda., en coordinación con los directivos, para la cual se utilizó un computador y el proyector de la empresa.

4.4.2 REQUISITOS PARA LA IMPLANTACIÓN

En base a un análisis técnico ha determinado los requisitos que requiere el “SISTEMA PARA LA GESTIÓN DE VENTAS Y COMISIONES EN LA EMPRESA REDPRIMCEL Cía. Ltda.” tanto en el servidor como en los clientes.

4.4.2.1.1 Requisitos del Servidor

TABLA 4-29: Requisitos del Servidor de Base de Datos

CARACTERÍSTICA	VALOR
Procesador	Pentium Core Dúo o superior, mínimo a 750MHz, recomendable 64bits.
Memoria RAM	Mínimo 512 MB, recomendable 4 GB
Disco Duro	8 GB Libres
Sistema Operativo	Windows XP, Windows 2003 Server, Windows 7, Windows 8
Software Adicional	Servidor Web Internet Information Server, SQL Server 2008 ó superior

4.4.2.1.2 Requisitos del Cliente

TABLA 4-30: Requisitos del Cliente

CARACTERÍSTICA	VALOR
Memoria RAM	Mínimo 512 MB, recomendable 1 GB.
Silverlight	Versión 5
Resolución del Cliente	800x600 mínimo, recomendable 1024x768.
Software	Internet Explorer 8 o superior, Firefox 14 o superior
Conectividad	Directa hacia y desde el servidor (Intranet).
Sistema Operativo	Windows XP, Windows 2003 Server, Windows 7, Windows 8

CAPÍTULO V

5 CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- ✓ La implantación del sistema en el ambiente de producción demostró una optimización clara en tiempos de gestión con respecto al proceso ventas y comisiones. Es así, que gracias a un diagnóstico eficaz y obtención de reportes que apoyen a los ejecutivos en la toma de decisiones.
- ✓ Silverlight es una de las plataformas de desarrollo poderosa de Microsoft para aplicaciones tradicionales de escritorio y aplicaciones web, al tiempo que ofrece las capacidades de ejecución en diferentes navegadores y sistemas operativos diferentes.
- ✓ Las aplicaciones RIA (Rich Internet Application) integran facilidad de despliegue y mantenimiento, actualización centralizada, independencia de plataforma y una mejor experiencia para el usuario, tanto en lo visual multimedia, familiaridad de controles, entre otros.
- ✓ Los Servicios RIA simplifican el desarrollo de aplicaciones enriquecidas para todo tipo de clientes, ya sea aplicaciones de escritorio o aplicaciones web tradicionales.
- ✓ La metodología XP es una excelente alternativa para el desarrollo de software, ya que da la posibilidad de ir definiendo más requerimientos a medida que va avanzando el proyecto, permitiéndole que sea escalable.
- ✓ Las metodologías ágiles como XP permite, desarrollar software de escritorio, web, móvil de buena calidad en menor tiempo.
- ✓ Las historias de usuario son una muy buena estrategia para definir claramente que es lo que desea que realice el sistema.
- ✓ Las pruebas de aceptación permiten afianzar lo que especificó el cliente en las historias de usuario, ya que se escriben como debe funcionar el sistema ante el ingreso de datos erróneos o bien cuando el ingreso de los datos son exitosos.

- ✓ La herramienta para desarrollo .Net 2012, permitió estandarizar los módulos y organizar de una mejor manera el código fuente del aplicativo con la inclusión de Silverlight , además con la corrección de errores de código en tiempo de compilación y ejecución

5.2 RECOMENDACIONES

- ✓ Para sacar el máximo provecho de la plataforma, se debe tener conocimientos de lenguaje XAML, manteniendo un equilibrio entre lo que se escribe en el código y lo que se pone en el marcado.
- ✓ Para complementar el estudio realizado se debería considerar el análisis de frameworks adicionales para RIA, como es Java Fx de Sun Oracle, Adobe Flex o Java Rich Faces que día a día cambian constantemente.
- ✓ Se recomienda utilizar un framework de desarrollo para establecer estándares y patrones de diseño que aseguren una arquitectura sólida que saquen provecho a dicho framework.
- ✓ Para establecer una arquitectura sólida en aplicaciones RIA utilizando Microsoft Silverlight, se recomienda usar en la capa de persistencia de datos hacer uso de tecnologías base tipo O/RM (Object/Relational Mapping frameworks), para obtener entidades y contextos necesarios para el modelo que exige Microsoft Silverlight.
- ✓ Para mejorar las vistas dentro de la capa de presentación se recomienda usar Microsoft Expression Blend por la mejora en el desarrollo del diseño de aplicaciones enriquecidas.
- ✓ La metodología XP permite grupos de trabajo pequeños (2 personas), tiempos cortos de desarrollo, proyectos de desarrollo orientados a objetos; siendo actualmente una de las metodologías con mayor acogida para proyectos de desarrollo de software, mientras que otras metodologías requieren una documentación detallada en sus procesos generando tiempos de desarrollo más extensos.

- ✓ Es recomendable llevar un estándar de programación permitiendo de esta manera que el aplicativo sea más manejable, de preferencia usar clases globales cuyos métodos o procedimientos puedan ser reutilizados y de ser el caso facilite en la agregación de más funcionalidad al aplicativo en una próxima versión.
- ✓ Los cambios en la funcionalidad del sistema deben realizarlos el cliente con el equipo de desarrollo; se recomienda no asumir ningún requerimiento por parte del equipo de desarrollo, pues el cliente es la única persona que determina lo que debe implementarse.
- ✓ Se recomienda el uso de herramientas automatizadas en las diferentes fases de XP ya que permite una mejor planificación, seguimiento y ejecución del proyecto siendo esto una gran ayuda para el equipo de desarrollo.

5.3 BIBLIOGRAFÍA

5.3.1 LIBROS

WAKE, William, Extreme Programming Explored, Prentice Hall PTR, Quinta Edición, Addison-Wesley, USA, 2010.

BECK, Kent, FOWLER, Martin, Planning Extreme Programming, Tercera Edición, USA, 2008.

CRISPIN, Lisa, Testing Extreme Programming, Primera Edición, Addison-Wesley, USA, 2012.

IOANNIS, Stamelos, PANAGIOTIS Sfetsos, Agil Software Development Quality Assurance, Second Edition, British Library, London 2009.

Pressman, Roger, Ingeniería de software. Un enfoque práctico, Séptima Edición, México 2012

Ghoda Ashish; Springer, INTRODUCIONG SILVERLIGHT 4; Science+Business Media; United States of America; 2010

Lair Robert; BEGINNING SILVERLIGHT 4 IN C#; Springer Science+Business Media; United States of America; 2010

Brown Pete, SILVERLIGHT 4 IN ACTION; Printed in the United States of America, ©2010 by Manning Publications Co

Gaudio Victor; FUNDATION EXPRESSION BLEND 4 WITH SILVERLIGHT; Springer Science+Business Media; United States of America; 2010

Posadas Mario; PROGRAMACION EN SILVERLIGTH 5.0; Netalia; Netalia, S.L. 2011

5.4 LINKOGRAFIA

Conozca Silverlight, <http://msdn.microsoft.com/es-es/silverlight/bb187401>

Qué es Silverlight, <http://www.tucompu.com/2007/10/qu-es-silverlight.html>

Introducción a Silverlight, <http://web.ontuts.com/tutoriales/introduccion-a-microsoft-silverlight-parte-i/>

Introducción a Silverlight, <http://web.ontuts.com/tutoriales/introduccion-a-microsoft-silverlight-parte-ii/>

Empiece a disfrutar aún más navegando por la Web, <http://msdn.microsoft.com/es-es/magazine/cc163404.aspx>

Interfaz de usuario, <http://www.fismat.umich.mx/~crivera/tesis/node6.html>

Introduccion a RIA (Rich Internet Application), <http://www.nohaylimites.com/?p=184>

Rich Internet Applications (RIA): A Convergence of User Interface Paradigms of Web and Desktop, <http://www.flomedia.de/diploma/>

Fundamento a la Arquitectura RIA, <http://www.mailxmail.com/curso-mysql-php/fundamento-arquitectura-ria>

Rich Internet Application, <http://www.slideshare.net/lisepi09/rich-internet-applications-2615942>

El mundo de RIA, <http://www.alejandromicheloud.com.ar/blog/?p=33>-- revisar

Metodologías Ágiles, <http://www.cenitec.com.mx/Manifiesto.pdf>

Desarrollo de software, <http://seccperu.org/files/Metodologias%20Agiles.pdf>

Metodologías de desarrollo de software,

<http://www.willydev.net/descargas/prev/TodoAgil.Pdf>

Desarrollando software,

http://tratandodeentenderlo.blogspot.com/2010_06_01_archive.html

Metodologías Ágiles, <http://www.ambyssoft.com/unifiedprocess/agileUP.html>

Desarrollo de software, http://en.wikipedia.org/wiki/Agile_software_development

Metodologías, <http://procesosdesoftware.wikispaces.com/METODOLOGIA+XP>

Metodologías Ágiles, <http://oness.sourceforge.net/proyecto/html/ch05s02.html>

Metodología XP,

<http://www.info-ab.uclm.es/asignaturas/42551/trabajosAnteriores/Presentacion-XP.pdf>

Ejemplo Metodología XP,

<http://users.dsic.upv.es/asignaturas/facultad/lsi/ejemploxp/index.html>

5.4.1 ARTÍCULOS

Diseño de aplicaciones en Silverlight 4 usando Servicios RIA, MEF y MVVM Mora David; 2010

De la base de datos al cliente con WCF RIA Service; Mora David; 2010

Expression Blend (5-13); Sanchez Jaime; Febrero 2010

5.5 ANEXOS

5.5.1 ANTEPROYECTO – ANEXO A SE ENCUENTRA EN CD

5.5.2 METRICA V3 – ANEXO B SE ENCUENTRA EN CD