

ESCUELA POLITECNICA NACIONAL

ESCUELA DE INGENIERIA

**APLICACIÓN DE LA PLATAFORMA BIZTALK 2006 EN EL
DESARROLLO DE SOFTWARE. CASO PRACTICO.**

**PROYECTO PREVIO A LA OBTENCION DEL TITULO DE INGENIERO
EN SISTEMAS INFORMATICOS Y DE COMPUTACION**

KARINA SALOME HIDALGO PALMA

DIRECTOR: ING. BOLIVAR PALAN

QUITO, JULIO DEL 2007

DECLARACION

Yo, Karina Salomé Hidalgo Palma, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Karina Salomé Hidalgo Palma

CERTIFICACION

Certifico que el presente proyecto ha sido realizado en su totalidad por la señorita Karina Salomé Hidalgo Palma, bajo mi supervisión.

Ing. Bolívar Palán

DIRECTOR DEL PROYECTO

DEDICATORIA

A mis padres por brindarme su apoyo incondicional durante toda mi vida. A mi Padre por enseñarme el valor de la palabra, por ser ejemplo de rectitud, honestidad y sacrificio incondicional. A mi Madre por su ejemplo, consejos, cuidados, amor y desvelos.

A mis hermanos, que con sus bromas han sabido apoyarme y darme ánimos para seguir luchando por mis sueños.

A mi novio, por brindarme su amor, apoyo y ayuda incondicional.

A mi abuelita por su ejemplo y todas sus oraciones.

A mis tíos y primos por estar siempre pendientes de mí.

A mis amigos y compañeros por todos los momentos compartidos.

Reciban este trabajo con todo mi cariño y eterno agradecimiento.

Karina

AGRADECIMIENTO

A Dios por todas las cosas buenas que ha puesto en mi vida, por permitirme educarme en una institución tan prestigiosa y reconocida.

A mis profesores, por compartir sus conocimientos y consejos, para formarme profesionalmente y crecer como persona.

Al Ingeniero Bolívar Palán, por su guía y apoyo incondicional durante la carrera y en este proyecto de titulación.

A todas las personas que colaboraron para el éxito de este proyecto.

Karina

INTRODUCCION

Un mercado altamente competitivo, no solo a nivel nacional ni regional sino a nivel mundial, requiere de agilidad en los procesos y el establecimiento de coaliciones con varios socios comerciales, no solamente para proveer productos sino servicios, siendo esta una realidad que cada vez más se palpa en nuestro país.

Este panorama afecta directamente en la vida de todos los que vivimos en esta sociedad de consumo. Es así que la calidad ya no se mide solo en qué tan bueno resultó un producto o servicio a lo largo del tiempo, sino en la velocidad y calidad de atención al cliente, en qué formas distintas al efectivo me presta para realizar pagos, que beneficios adicionales poseo, entre otras.

Es por esto que las alianzas con socios de negocios es trascendental. Para que un convenio de este tipo funcione eficientemente y preste el beneficio esperado, reflejado en la realización de más ventas en menos tiempo, es necesario primeramente tener la información de nuestra empresa integrada, ordena y funcionando eficazmente, para poder integrarla con la de nuestros socios en el menor tiempo posible y con los más altos resultados.

El presente proyecto nace de la necesidad de brindar a las empresas grandes y pequeñas la oportunidad de integrar las aplicaciones diversas que existen dentro de la misma, que no necesariamente están en un mismo lenguaje de programación, ni manejan los mismo tipos de datos, pero que tienen en determinado momento un propósito en común, o integrar determinados módulos de su empresa con socios comerciales, de la manera más rápida, con total fiabilidad y con el menor costo posible, permitiendo verificar los resultados de la alianza con prontitud y tecnológicamente actualizada.

Este proyecto se encuentra distribuido en tres capítulos cuyo contenido se describe a continuación:

El Primer Capítulo presenta estudios sobre las arquitecturas de datos, metodologías de desarrollo de software y un detallado análisis de la plataforma Microsoft BizTalk Server 2006 para el desarrollo e integración de aplicaciones.

El Segundo Capítulo nos brinda la más completa información sobre el proceso de desarrollo de software de cada una de las aplicaciones que se desarrollan a lo largo de este proyecto, siguiendo la metodología seleccionada en el primer capítulo, así como de la integración de las mismas. Al finalizar se encuentra desarrolladas las pruebas de los sistemas independientes y su orquestación, así como el caso de aplicación desarrollado para una evaluación objetiva del proyecto.

Finalmente, el Capítulo Tres contiene las conclusiones, recomendaciones y reflexión final, obtenidas una vez culminado el presente proyecto.

RESUMEN

En la actualidad, las personas que integran aplicaciones de distinto propósito, con diferentes protocolos de transmisión de datos y que utilizan o se desarrollaron en plataformas distintas, gastan tiempo valioso en la orquestación de las mismas, con el agravante de que el código de programación que utilizan no necesariamente es el más efectivo, no permite monitoreo de las aplicaciones ni de la integración, pierden muchos datos posiblemente importantes durante este proyecto y toma muchos meses el finalizar este proceso.

Una solución adecuada a los problemas anteriormente descritos es el empleo de nuevas arquitecturas orientadas a datos, modernos integradores de alta fiabilidad, potentes herramientas de monitoreo de orquestaciones, para satisfacer las necesidades del mercado y del usuario final.

En el presente proyecto se desarrolla una orquestación de aplicaciones multiplataforma, utilizando tecnología de punta, que es una clara demostración de los avances en este sentido el desarrollo de aplicaciones.

INDICE DE CAPITULOS

CAPITULO 1	13
SELECCIÓN DE ARQUITECTURAS DE DESARROLLO DE SOFTWARE	
MULTIPLATAFORMA.	18
1.1 DESCRIPCIÓN DE LA METODOLOGÍA DE DESARROLLO DE SOFTWARE A APLICAR.....	19
1.1.1 CARACTERIZACIÓN DEL PROBLEMA.....	19
1.1.2 SELECCIÓN DE LA METODOLOGÍA.....	22
1.1.3 Características.....	31
1.2 SELECCIÓN DE UNA ARQUITECTURA DE DESARROLLO DE SOFTWARE MULTIPLATAFORMA.....	37
1.2.1 ARQUITECTURAS DE COMUNICACIÓN.....	37
1.2.2 SELECCIÓN DE LA TECNOLOGÍA PARA IMPLEMENTAR LA ARQUITECTURA MULTIPLATAFORMA.	41
1.2.3 SELECCIÓN DE LA TECNOLOGÍA QUE SOPORTA LA ARQUITECTURA.....	45
Características.....	45
1.2.4 DESCRIPCIÓN DETALLADA DE LA ARQUITECTURA SELECCIONADA PARA EL PROYECTO.....	46
1.3 SELECCIÓN DE LAS HERRAMIENTAS DE DESARROLLO.	49
1.3.1 HERRAMIENTAS DE BASE	50
1.3.2 HERRAMIENTAS DE DESARROLLO DE APLICACIONES.....	59
1.3.3 SELECCIÓN DE LAS HERRAMIENTAS DE DESARROLLO	62
Características.....	63
1.3.4 DESCRIPCIÓN DETALLADA DE LAS HERRAMIENTAS SELECCIONADAS.	64
 CAPITULO 2	 73
INGENIERÍA DE SOFTWARE APLICANDO LA ARQUITECTURA MULTIPLATAFORMA BIZTALK 2006.	
2.1 INGENIERÍA DE REQUERIMIENTOS.....	73
2.1.1 ESPECIFICACION DE REQUERIMIENTOS.	73
2.1.1.1.1 Caso de uso del negocio del sistema de facturación en supermercados	74
2.1.1.1.2 Caso de uso del negocio del sistema de Tarjetas de crédito.....	75
2.1.1.1.3 Caso de uso del negocio del sistema Integrado.....	76
2.1.1.2 Diagramas de Casos de Uso de los Sistemas.....	76
2.1.1.2.1 Caso de uso del sistema del sistema de facturación en supermercados.	77
2.1.1.2.2 Caso de uso del sistema del sistema de Pago con tarjeta de Crédito...	81
2.1.1.2.3 Casos de uso del Sistema Integrado	85
2.1.1.3 Planificación de los Sitios Web de los sistemas	85
2.1.1.3.1 Propósito.....	85
2.1.1.3.2 Usuarios de los sitios Web	86
2.1.1.3.3 Almacenamiento de la información en el sitio Web.....	86
2.1.2 MODELO DE ANÁLISIS	86

2.1.2.1 Realización de los casos de uso del Sistema de Facturación de Supermercados.	87
2.1.2.1.1 Realización del caso de uso Generar Ventas.....	87
2.1.2.1.2 Realización del caso de uso Reportar Inventario.....	88
2.1.2.1.3 Realización del caso de uso Deshacer Venta.....	89
2.1.2.2 Realización de los casos de uso del Sistema de Tarjeta de Crédito.	89
2.1.2.2.1 Realización del caso de uso Solicitar Crédito	89
2.1.2.2.2 Realización del caso de uso Validar Tarjeta.....	90
2.1.2.2.3 Realización del caso de uso Crear Tarjeta.....	91
2.1.2.2.4 Realización del caso de uso Consultar Tarjeta.....	92
2.1.2.2.5 Realización del caso de uso Modificar Tarjeta.....	92
2.1.2.2.6 Realización del caso de uso Eliminar Tarjeta.....	93
2.1.2.3 Realización de los casos de uso del Sistema Integrado.....	94
2.1.2.3.1 Realización del caso de uso Integración.....	94
2.1.3 PAQUETES DE ANÁLISIS	95
2.1.3.1 Paquetes de análisis del Sistema de Facturación.....	95
2.1.3.1 Paquetes de análisis del Sistema de Tarjeta de Crédito.....	96
2.1.3.1 Paquetes de análisis del Sistema Integrado.	96
2.1.4 ANÁLISIS DEL SITIO WEB	98
2.1.4.1 Análisis del sitio Web para el sistema de facturación de supermercados. .	99
2.1.4.2 Análisis del sitio Web para el sistema de tarjeta de crédito.	100
2.2 DISEÑO	101
2.2.1 DISEÑO DE CASOS DE USO	103
2.2.1.1 Subsistema de Facturación de supermercados.....	104
2.2.1.2 Subsistema de Tarjeta de Crédito.	105
2.2.1.3 Subsistema de Integración.....	105
2.2.2 DIAGRAMA DE CLASES DE DISEÑO	106
Las clases de diseño se las realiza para cada uno de los sistemas de manera independiente por las características descritas en los puntos de análisis de los sistemas anteriores.....	106
2.2.2.1 Diagrama de clases de diseño para el sistema de facturación de supermercados.	107
2.2.2.2 Diagrama de clases de diseño para el sistema de tarjetas de crédito.	108
2.2.3 DISEÑO DE LOS SITIOS DE LOS SISTEMAS	109
2.2.3.1 Mapa de navegación del sitio Web.....	109
2.2.3.1.1 Mapa de navegación del sitio Web del sistema de facturación de supermercados.	109
2.2.3.1.2 Mapa de navegación del sitio Web del sistema de tarjeta de crédito.	
110	
2.2.3.2 Diseño de la estructura del sitio del Sistema de Facturación de Supermercados.	110
2.2.3.2.1 Index.	111
2.2.3.2.2 Menú Principal.	111
2.2.3.2.3 Crear y Modificar.	112
2.2.3.2.4 Consultar.....	113
2.2.3.2.5 Eliminar.	114
2.2.3.2.6 Generar Factura.	114
2.2.3.2.7 Deshacer Venta.....	115
2.2.3.3 Diseño de la estructura del sitio del Sistema de Tarjeta de Crédito.	116

2.2.3.3.1	Index.....	117
2.2.3.3.2	Menú Principal y Consultar.....	118
2.2.3.3.3	Crear Nueva y Modificar Tarjeta.....	119
2.2.3.3.4	Eliminar.....	120
2.3	IMPLEMENTACIÓN.....	121
2.3.1	DEFINICIÓN DE SUBSISTEMAS DE IMPLEMENTACIÓN.....	121
2.3.1.1	Subsistema de implementación del Sistema de Facturación de Supermercados.....	123
2.3.1.1.1	Subsistema de implementación de Gestión de Venta.....	123
2.3.1.2	Subsistema de implementación del Sistema Tarjetas De Crédito.....	124
2.3.1.2.1	Subsistema de implementación de Gestión de Tarjetas.....	124
2.3.1.2.2	Subsistema de implementación de Gestión de Créditos.....	125
2.3.2	DEPENDENCIAS ENTRE LOS SUBSISTEMAS DE IMPLEMENTACION.....	126
2.3.3	CLASES DE IMPLEMENTACIÓN.....	127
2.3.3.1	Subsistema de Implementación GestionDeProductos.....	128
2.3.3.2	Subsistema de Implementación GestionDeTarjetas.....	129
2.3.3.3	Subsistema de Implementación GestionDeVentas.....	130
2.3.3.4	Subsistema de Implementación GestionDeCreditos.....	130
2.3.4	DESARROLLO DE LA APLICACIÓN.....	131
2.3.4.1	Definición de herramientas para subsistemas pertenecientes al sistema de Facturación de Supermercados.....	131
2.3.4.1.1	Herramientas para el Front – End.....	131
2.3.4.1.2	Herramientas para el Middle – End.....	132
2.3.4.1.3	Herramientas para el Back – End.....	133
2.3.4.2	Definición de herramientas para subsistemas pertenecientes al sistema de Tarjetas de Crédito.....	134
2.3.4.2.1	Herramientas para el Front – End.....	134
2.3.4.2.2	Herramientas para el Middle – End.....	135
2.3.4.2.3	Herramientas para el Back – End.....	136
2.3.4.3	Alcance de la implementación el sistema.....	136
2.3.4.3.1	Sistema de Facturación de Supermercados.....	136
2.3.4.3.2	Sistema de Tarjeta de Crédito.....	137
2.3.4.4	Estándares de programación.....	137
2.3.4.5	Estructura de la Integración de las aplicaciones con Microsoft BizTalk Server 2006.....	139
2.3.4.5.1	Configuración del Orquestador.....	139
2.3.4.5.2	Mapas del Sitio.....	140
2.3.4.5.3	Pipelines.....	141
2.3.4.5.4	Esquemas.....	142
2.3.4.5.5	Referencias a esquemas creados.....	143
2.3.4.5.6	Creación y Habilitación de puertos para la orquestación.....	143
2.3.4.5.7	Orquestación.....	144
2.4	PRUEBAS Y CONTROL DE CALIDAD.....	146
2.4.1	CASOS Y PROCEDIMIENTOS DE PRUEBA PARA LOS CASOS DE USO DEL SISTEMA DE FACTURACIÓN DE SUPERMERCADOS.....	147
2.4.1.1	Casos y Procedimientos de prueba para el caso de uso Generar Ventas.....	147

2.4.1.2	Casos y Procedimientos de prueba para el caso de uso Crear Producto.	148
2.4.1.3	Casos y Procedimientos de prueba para el caso de uso Consultar Producto.	148
2.4.1.4	Casos y Procedimientos de prueba para el caso de uso Modificar Producto.	149
2.4.1.5	Casos y Procedimientos de prueba para el caso de uso Eliminar Producto.	150
2.4.1.6	Casos y Procedimientos de prueba para el caso de uso Reportar Inventario.....	151
2.4.1.7	Casos y Procedimientos de prueba para el caso de uso Deshacer Venta.	151
2.4.2	CASOS Y PROCEDIMIENTOS DE PRUEBA PARA LOS CASOS DE USO DEL SISTEMA DE TARJETA DE CREDITO.	152
2.4.3	RESULTADO DE LAS PRUEBAS PARA EL SISTEMA DE FACTURACIÓN DE SUPERMERCADOS.....	157
2.4.4	RESULTADO DE LAS PRUEBAS PARA EL SISTEMA DE TARJETA DE CRÉDITO.	157
2.4.5	EVALUACIÓN DEL SISTEMA EN UN CASO DE APLICACIÓN.....	158
2.4.5.1	Formulación del caso de aplicación.....	158
2.4.5.1.1	Objetivos.....	158
2.4.5.1.2	Alcance del caso de aplicación.....	158
2.4.5.2	Descripción del caso de aplicación.....	159
2.4.5.3	Construcción.....	159
2.4.5.4	Ejecución de las Pruebas	159
2.4.5.5	Análisis de los Resultados De las Pruebas del Caso de Aplicación.	163
CAPITULO 3		165
CONCLUSIONES Y RECOMENDACIONES		165
3.1	CONCLUSIONES.....	165
3.2	RECOMENDACIONES	167
3.3	REFLEXION FINAL	168
BIBLIOGRAFIA.....		170
ANEXO I.....		171
CONTENIDO DEL CD ADJUNTO		171

INDICE DE TABLAS

Tabla I - 1 Comparación de metodologías de Desarrollo de Software.....	31
Tabla I - 2 Comparación de Tecnologías.....	45
Tabla I - 3 Comparativa de herramientas de desarrollo de software.....	63
Tabla II - 1 Flujo de eventos del Caso de uso del Negocio Generar Ventas.....	74
Tabla II - 2 Flujo de Eventos del Caso de uso de negocio Generar Transacción.....	75
Tabla II - 3 Flujo de eventos del Caso de uso Generar Ventas.....	77
Tabla II - 4 Flujo de eventos del Caso de uso Crear Producto.....	78
Tabla II - 5 Flujo de eventos del Caso de uso Consultar.....	78
Tabla II - 6 Flujo de eventos del Caso de uso Eliminar Producto.....	79
Tabla II - 7 Flujo de eventos del Caso de uso Modificar Producto.....	79
Tabla II - 8 Flujo de eventos del Caso de uso Reportar Inventario.....	80
Tabla II - 9 Flujo de eventos del Caso de uso Deshacer Venta.....	80
Tabla II - 10 Flujo de eventos del Caso de uso Solicitar Crédito.....	82
Tabla II - 11 Flujo de eventos del Caso de uso Validar Tarjeta.....	82
Tabla II - 12 Flujo de eventos del Caso de uso Crear Tarjeta.....	83
Tabla II - 13 Flujo de eventos del Caso de uso Consultar.....	83
Tabla II - 14 Flujo de eventos del Caso de uso Eliminar Tarjeta.....	84
Tabla II - 15 Flujo de eventos del Caso de uso Modificar Tarjeta.....	84
Tabla II - 16 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Generar Venta.....	87
Tabla II - 17 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Reportar Inventario.....	88
Tabla II - 18 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Deshacer venta.....	89
Tabla II - 19 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Solicitar crédito.....	90
Tabla II - 20 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Validar tarjeta.....	90
Tabla II - 21 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Crear tarjeta.....	91
Tabla II - 22 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Consultar.....	92
Tabla II - 23 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Modificar tarjeta.....	93
Tabla II - 24 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Eliminar tarjeta.....	94
Tabla II - 25 Tipos de Botones de la página Web tipo Index.....	117
Tabla II - 26 Estándares de programación para los sistemas a desarrollarse.....	138
Tabla II - 27 Caso de Prueba para el caso de uso Generar Ventas.....	147
Tabla II - 28 Procedimiento de prueba para el Caso de prueba Generar Venta (001).....	147
Tabla II - 29 Caso de Prueba para el caso de uso Crear Producto.....	148
Tabla II - 30 Procedimiento de prueba para el Caso de prueba Crear Producto.....	148
Tabla II - 31 Caso de Prueba para el caso de uso Consultar Producto.....	148

Tabla II - 32 Procedimiento de prueba para el Caso de prueba Consultar Producto (009) Arroz.....	149
Tabla II - 33 Caso de Prueba para el caso de uso Modificar Producto.....	149
Tabla II - 34 Procedimiento de prueba para el Caso de prueba Modificar Producto (009) Arroz.....	150
Tabla II - 35 Caso de Prueba para el caso de uso Eliminar Producto.....	150
Tabla II - 36 Procedimiento de prueba para el Caso de prueba Eliminar Producto (009) Arroz.....	150
Tabla II - 37 Caso de Prueba para el caso de uso Reportar Inventario.....	151
Tabla II - 38 Procedimiento de prueba para el Caso de prueba Reportar Inventario.	151
Tabla II - 39 Caso de Prueba para el caso de uso Deshacer Venta.....	151
Tabla II - 40 Procedimiento de prueba para el Caso de prueba Deshacer Venta.	152
Tabla II - 41 Caso de Prueba para el caso de uso Validar Tarjeta.....	152
Tabla II - 42 Procedimiento de prueba para el Caso de prueba Validar Tarjeta (012) 3606 184800 3646.	153
Tabla II - 43 Caso de Prueba para el caso de uso Solicitar Crédito.	153
Tabla II - 44 Procedimiento de prueba para el Caso de prueba Solicitar Crédito de 150 dólares para la tarjeta 3606 184800 3646.	154
Tabla II - 45 Caso de Prueba para el caso de uso Crear Tarjeta.....	154
Tabla II - 46 Procedimiento de prueba para el Caso de prueba Crear Tarjeta.	154
Tabla II - 47 Caso de Prueba para el caso de uso Consultar Tarjeta.	155
Tabla II - 48 Procedimiento de prueba para el Caso de prueba Consultar Tarjeta (015) 3606 184800 3647.	155
Tabla II - 49 Caso de Prueba para el caso de uso Modificar Tarjeta.....	155
Tabla II - 50 Procedimiento de prueba para el Caso de prueba Modificar Tarjeta (015) 3606 184800 3647.	156
Tabla II - 51 Caso de Prueba para el caso de uso Eliminar Tarjeta.....	156
Tabla II - 52 Procedimiento de prueba para el Caso de prueba Eliminar Tarjeta (015) 3606 184800 3647.	157
Tabla II - 53 Matriz de resultados de las pruebas del sistema de facturación de supermercado.....	157
Tabla II - 54 Matriz de resultados de las pruebas del sistema de tarjeta de crédito.	157
Tabla II - 55 Caso de Prueba para Facturación de productos en stock y pago con tarjeta de crédito.	160
Tabla II - 56 Procedimiento de prueba para el Caso de Prueba para Facturación de productos en stock y pago con tarjeta de crédito.....	160
Tabla II - 57 Caso de Prueba para Facturación Estado de la aprobación del crédito una vez finalizada la factura.	161
Tabla II - 58 Procedimiento de prueba para el Caso de Prueba para Estado de la aprobación del crédito una vez finalizada la factura.	161
Tabla II - 59 Caso de Prueba para Estado del Stock de productos al finalizar la factura.	161
Tabla II - 60 Procedimiento de prueba para el Caso de Prueba para Estado del Stock de productos al finalizar la factura.	162
Tabla II - 61 Caso de Prueba para Cupo disponible en la tarjeta luego de realizada la transacción.....	162
Tabla II - 62 Procedimiento de prueba para el Caso de Prueba para Cupo disponible en la tarjeta luego de realizada la transacción.	163
Tabla II - 63 Matriz de resultados de las pruebas del caso de Aplicación.	163

INDICE DE FIGURAS

Figura I - 1 Fases e Interacciones de la Metodología RUP.	26
Figura I - 2 Modelo de Equipo de Trabajo. Metodología MSF.....	29
Figura I - 3 Etapas y procesos de MSF.....	30
Figura I - 4 Modelo de Integración con BizTalk Server 2006.....	51
Figura I - 5 Motor BizTalk Server 2006.....	53
Figura I - 6 Modelo Multicapa J2EE.....	66
Figura I - 7 Framework .NET.....	69
Figura I - 8 Entorno Común de Ejecución para Lenguajes (CLR).....	70
Figura I - 9 Biblioteca de Clases Base de .NET.....	71
Figura II - 1 Diagrama de caso de uso del negocio Generar Venta.....	74
Figura II - 2 Diagrama del caso de uso del negocio Generar Transacción.....	75
Figura II - 3 Diagrama del caso de uso del negocio de los Sistemas integrados.....	76
Figura II - 4 Diagrama de casos de uso del Sistema de Facturación de Supermercados...	77
Figura II - 5 Diagrama de casos de uso del Sistema de Tarjeta de Crédito.....	81
Figura II - 6 Diagrama de casos de uso del Sistema Integrado.	85
Figura II - 7 Diagrama de colaboración de una realización del caso de uso Generar Venta	87
Figura II - 8 Diagrama de colaboración de una realización del caso de uso Reportar Inventario.....	88
Figura II - 9 Diagrama de colaboración de una realización del caso de uso Deshacer venta	89
Figura II - 10 Diagrama de colaboración de una realización del caso de uso Solicitar credito.....	89
Figura II - 11 Diagrama de colaboración de una realización del caso de uso Validar Tarjeta.....	90
Figura II - 12 Diagrama de colaboración de una realización del caso de uso Crear tarjeta.	91
Figura II - 13 Diagrama de colaboración de una realización del caso de uso Consultar...	92
Figura II - 14 Diagrama de colaboración de una realización del caso de uso Modificar tarjeta.	92
Figura II - 15 Diagrama de colaboración de una realización del caso de uso Eliminar tarjeta.	93
Figura II - 16 Identificación de paquetes de análisis a partir de clases importantes del dominio del problema del sistema de Facturación.	95
Figura II - 17 Identificación de paquetes de análisis a partir de clases importantes del dominio del problema del sistema de Tarjeta de Crédito.	96
Figura II - 18 Identificación de paquetes de análisis a partir de clases importantes del dominio del problema del sistema de integración.	96
Figura II - 19 Diagrama de dependencia entre paquetes del sistema de Facturación.....	97
Figura II - 20 Diagrama de dependencia entre paquetes del sistema de Tarjeta de Crédito.	97
Figura II - 21 Diagrama de dependencia entre paquetes del Sistema Integrado.	98
Figura II - 22 Diagrama de Organización funcional del Sistema Integrado.....	101

Figura II - 23 Identificación de subsistemas de diseño a partir de paquetes de análisis del Sistema de Facturación de Supermercados.	102
Figura II - 24 Identificación de subsistemas de diseño a partir de paquetes de análisis del Sistema de Tarjeta de Crédito.....	102
Figura II - 25 Identificación de subsistemas de diseño a partir de paquetes de análisis del Sistema Integrado.	103
Figura II - 26 Diagrama de colaboración de una realización del caso de uso Generar Venta.....	104
Figura II - 27 Diagrama de colaboración de una realización del caso de uso Reportar Inventario.....	104
Figura II - 28 Diagrama de colaboración de una realización del caso de uso Deshacer Venta.....	104
Figura II - 29 Diagrama de colaboración de una realización del caso de uso Solicitar Crédito	105
Figura II - 30 Diagrama de colaboración de una realización del caso de uso Validar Tarjeta.....	105
Figura II - 31 Diagrama de colaboración de una realización de la integración de las aplicaciones.	106
Figura II - 32 Diagrama de clases de diseño del Sistema de Facturación de Supermercados.	107
Figura II - 33 Diagrama de clases de diseño del Sistema de Tarjetas de Crédito.	108
Figura II - 34 Mapa de navegación del sitio Web del sistema de facturación de supermercados.	109
Figura II - 35 Mapa de navegación del sitio Web del sistema de tarjeta de crédito.....	110
Figura II - 36 Diseño de la página Web tipo Index	111
Figura II - 37 Diseño de la página Web tipo Menú Principal.....	112
Figura II - 38 Diseño de la página Web de tipo Crear o Modificar.....	113
Figura II - 39 Diseño de la página Web tipo Consultar.....	113
Figura II - 40 Diseño de la página Web tipo Eliminar.	114
Figura II - 41 Diseño de la página Web tipo Generar Factura	115
Figura II - 42 Diseño de la página Web tipo Deshacer Venta.....	116
Figura II - 43 Diseño de la página Web de tipo Index.	118
Figura II - 44 Diseño de la página Web tipo Menú Principal y Consultar	119
Figura II - 45 Diseño de la página Web tipo Crear Nueva.	120
Figura II - 46 Diseño de la página Web tipo Eliminar.	121
Figura II - 47 Identificación de subsistemas de diseño a partir de paquetes de análisis ..	122
Figura II - 48 Dependencia de los componentes del subsistema de diseño de GestionDeVenta.	123
Figura II - 49 Dependencia de los paquetes del subsistema de diseño GestionDeProductos	124
Figura II - 50 Dependencia de los componentes del subsistema de diseño de GestionDeTarjetas.	125
Figura II - 51 Dependencias de componentes del subsistema de diseño GestionDeCreditos.	126
Figura II - 52 Dependencia de compilación de los componentes de los subsistemas de la integración.	127
Figura II - 53 Diagrama de Clases del subsistema de implementación GestionDeProductos.	128

Figura II - 54 Diagramas de Clases del subsistema de implementación GestionDeTarjetas.	129
Figura II - 55 Diagrama de Clases de implementación GestionDeVentas.....	130
Figura II - 56 Diagrama de Clases del subsistema de implementación GestionDeCreditos.	130
Figura II - 57 Intefaz de diseño con WebSphere Studio Application Developer.....	132
Figura II - 58 Interfaz de desarrollo con WebSphere Application Developer.....	133
Figura II - 59 Interfaz de trabajo de Macromedia Fireworks MX.....	134
Figura II - 60 Interfaz de trabajo Microsoft Visual C# .NET 2005.....	135
Figura II - 61 Interfaz de programación en Microsoft Visual C# .NET 2005.....	136
Figura II - 62 Pantalla de configuración de Microsoft BizTalk Server 2006.....	139
Figura II - 63 Programación del mapa de la orquestación.....	140
Figura II - 64 Creación de Pipelines en el esquema de integración.	141
Figura II - 65 Diseño de la estructura de los archivos XML.	142
Figura II - 66 Referencias a esquemas previamente diseñados.....	143
Figura II - 67 Creación de puerto de envío para la integración.....	144
Figura II - 68 Programación del proceso de orquestación.....	145
Figura II - 69 Explorador de la Solución de Integración	146

CAPITULO 1

SELECCIÓN DE ARQUITECTURAS DE DESARROLLO DE SOFTWARE MULTIPLATAFORMA.

Las aplicaciones de software son el resultado de un proceso de desarrollo ordenado y estructurado. Ninguna aplicación está aislada, y frente a esta realidad el conectar o enlazar aplicaciones con el avance de la tecnología se convierte en una norma.

La globalización impone la necesidad de integrar aplicaciones independientes al interior de la empresa y con socios estratégicos, estos programas pueden encontrarse en diferentes formatos e implementados en plataformas heterogéneas.

Las empresas se encaminan hacia la comercialización eficiente y masiva de servicios, integrando sistemas de propósito diferente dentro de un todo coherente e impulsando a que cada vez más arquitectos elaboren sus diseños orientados a servicios dentro de entornos corporativos.

Las organizaciones orientan sus sistemas a una arquitectura SOA¹, que integra eficientemente las capacidades de la empresa, agilitando tareas corporativas. Microsoft Biztalk Server 2006² desarrolla SOA dentro de una organización.

Los procesos de integración de aplicaciones y servicios requieren de un entorno computacional, que brinde la posibilidad de disponer de procesos de codificación, transformación, seguridad e integridad de la información. En estos aspectos BizTalk Server 2006 brinda soporte al equipo de desarrollo.

¹ WIKIPEDIA, Glosario, <http://es.wikipedia.org/wiki/SOA>, Noviembre del 2006, [SOA.- Concepto de Arquitectura de software que define la utilización de servicios para dar soporte a los requerimientos del usuario, proporciona la metodología y un marco de trabajo para documentar las capacidades del negocio y puede dar soporte a las actividades de integración y consolidación.]

² MICROSOFT CORPORATION, Hoja De Datos BT 2006.pdf, Diciembre del 2005, [Biztalk Server.- Poderosa herramienta que utiliza las últimas tecnologías de Microsoft y estándares del mercado para automatizar y gestionar procesos de negocio. Creación, gestión y monitorización de procesos de negocio distribuidos.]

En este capítulo se presenta el proceso de selección de la metodología que más se ajusta a las necesidades de este proyecto. Seguidamente la selección de las tecnologías que soportan las arquitecturas que facilitan la integración de aplicaciones con el orquestador Biztalk Server 2006 y finalmente se selecciona las herramientas de desarrollo a utilizar para la implementación de un caso de aplicación.

1.1 DESCRIPCIÓN DE LA METODOLOGÍA DE DESARROLLO DE SOFTWARE A APLICAR.

1.1.1 CARACTERIZACIÓN DEL PROBLEMA

La competencia en el mundo de los negocios, el crecimiento de los mercados y la globalización, exige que las empresas integren todos los sistemas independientes que poseen para cada uno de los procesos y departamentos que se encuentran dentro y fuera de la organización, como un todo, funcionando bajo la misma lógica de negocio, para elevar su rentabilidad e intercambiar información tanto dentro como fuera de la empresa, y no quedarse detrás de la competencia, alcanzando así el tan anhelado éxito en los negocios, que puede verse truncado por el manejo de una tecnología obsoleta dentro de la organización.

Las distintas aplicaciones que se manejan dentro de la empresa o los sistemas de comunicación con socios comerciales en su generalidad son elaboradas de tal manera que se ajusten a los requerimientos específicos del departamento, proceso o división para la que han sido implementados y tienen características diferentes, es decir que no necesariamente funcionan con la misma plataforma, son distintos los protocolos de comunicación o de red empleados para transportar los datos, no están elaborados con el mismo lenguaje de programación, sus datos se almacenan de hecho en distintas bases de datos, permitiendo que exista dentro de la misma empresa muchas de las veces información redundante que lejos de beneficiar a la organización puede ser causa de errores, pueden debilitar el poder de las búsquedas, dificultar el cruce de información, los sistemas no son

administrados ni actualizados de la misma forma, generalmente no están documentados y constituyen verdaderas cajas negras en las cuales solo los usuarios antiguos pueden dar solución a problemas, pero tienen una fortaleza que permite subsanar las deficiencias anteriormente mencionadas como que todos manejan una misma lógica de negocio.

Actualmente se integran los sistemas autónomos dentro de las empresas y los que permiten la comunicación con socios comerciales, mediante la utilización de código de programación en forma de switchs transaccionales. El conectar aplicaciones de software implica mucho más que intercambiar bytes, involucra la creación de procesos de negocio efectivos, que permitan incrementar la rentabilidad de la empresa dentro de un todo coherente, mediante la creación de entornos orientados a servicios, ya que el código simple de integración no tiene la eficiencia que presta una orquestación diseñada por arquitectos de software para entornos corporativos y soportada por herramientas de nueva generación.

Las diferentes arquitecturas tienden a integrarse, un ejemplo palpable de esta realidad es la cada vez más frecuente utilización del protocolo SOAP³, y de datos transportados y almacenados en formato XML⁴.

El orquestar aplicaciones requiere de codificación avanzada, formatos de datos adecuados y lenguajes de programación compatibles, gestión de mensajes, protocolos, soporte de procesos de transformación, además de la implementación de altas medidas de seguridad, no solo en las aplicaciones sino en el orquestados como tal, para que se encuentren al nivel que implica el intercambio de datos críticos para la empresa entre distintos sistemas.

³ WIKIPEDIA, Glosario, <http://es.wikipedia.org/wiki/SOAP>, Noviembre del 2006, [SOAP.- Simple Object Access Protocol, es un protocolo estándar creado por Microsoft, IBM entre otros, define como dos objetos en diferentes procesos que pueden comunicarse mediante el intercambio de datos XML. Es un protocolo utilizado en los servicios web.]

⁴ RÍO ÁLVAREZ Jesús, Servicios de Proceso de Imágenes, XML y publicación web.pdf, [XML.- Extensible Markup Lenguajes, es un metalenguaje que define una sintaxis utilizada para definir otros lenguajes de etiquetas estructurados, define una estructura de documento en un lenguaje de descripción de datos, permite la intercomunicación entre aplicaciones.]

Seguidamente se describe técnicamente el problema caracterizado anteriormente, esto se utilizará para la selección de la metodología y de las herramientas de desarrollo para el caso de aplicación.

1.1.1.1 Caracterización Técnica del Problema.

1.1.1.1.1 Auspiciante, cliente o interesado:

- Autora del proyecto de titulación.

1.1.1.1.2 Recursos:

- Humano: 1 persona.
- Tiempo de Duración: 5-6 meses.
- Económico: Limitado.
- Tecnológico:
 - Hardware: Se cuenta con 3 computadores para el desarrollo del caso de aplicación, 1 switch para conectar las máquinas, cables suficientes para la conexión.
 - Software: Microsoft Biztalk Server 2006 para orquestar las aplicaciones, herramientas necesarias para el desarrollo de aplicaciones e integración, además del software base requerido.
 - Comunicaciones: Se posee acceso ilimitado a Internet las 24 horas del día.
- Información: La bibliografía requerida la encontramos en el sitio Web oficial de Microsoft, en varios libros, manuales y más recursos del Internet.

1.1.1.1.3 Cobertura del Servicio:

La cobertura del servicio está dada por el acceso que posee cada una de las organizaciones a una red al menos interna, que permita la comunicación de cada una de las aplicaciones con el servidor y los clientes para poder integrarlas.

1.1.1.1.4 Estructura Tecnológica del proyecto:

- El caso de estudio de desarrollará sobre la plataforma Microsoft Biztalk Server 2006.
- Es necesaria la utilización de un servidor que orqueste las aplicaciones.
- Se requiere de al menos dos aplicaciones de prueba para ser orquestadas. Las aplicaciones deben tener un fin común, para el cuál sea necesario su integración.
- Los computadores estarán conectados dentro de una red LAN, para los cuales se utilizarán dispositivos de red como switch y cables.
- Se implantará un esquema de programación orientada a objetos, con una arquitectura orientada a servicios (SOA), aplicando el concepto de multicapa (aplicación, negocio, base de datos).
- La integración de datos y aplicaciones se la realizará mediante Web services.
- Es indispensable la entrega de la documentación completa del sistema y del proyecto de titulación.
- El caso de estudio debe demostrar las ventajas de Microsoft Biztalk Server 2006.

1.1.1.1.5 Mercado Objetivo:

- Medianas y grandes empresas que necesiten ser más competitivas, que requieran de una integración efectiva entre los aplicativos de su organización y de sus socios comerciales.
- A todas las personas interesadas en ahondar en esta plataforma de integración.

1.1.2 SELECCIÓN DE LA METODOLOGÍA

En esta sección se describe tres de las más importantes metodologías para desarrollo de aplicativos de software de manera formal, como son: Extreme

Programming (XP)⁵, Rational Unified Process (RUP)⁶ y Microsoft Solution Framework (MSF)⁷.

Luego se procederá a la selección de la metodología que se ajuste al problema descrito inicialmente. Cada metodología posee características, ventajas y desventajas que permiten realizar una evaluación objetiva de las mismas para este proyecto. La metodología seleccionada se describirá con mayor detalle.

1.1.2.1 Descripción de la Metodología Extreme Programming (XP).

Esta metodología se fundamenta en una serie de reglas y principios aplicables a desarrolladores, diseñadores y administradores, se pueden englobar dentro de las metodologías ágiles evolutivas, es decir se prioriza las tareas que dan resultados directos, mejorando la adaptabilidad y previsión. Casi todos los componentes de esta metodología son seleccionadas de las mejores prácticas utilizadas desde hace muchos años por especialistas en el desarrollo de proyectos informáticos, dando como resultado una metodología única y compacta.

1.1.2.1.1 Características básicas:

- **Pruebas Unitarias:** Son las pruebas realizadas en los procesos principales, que permiten un adelanto o una predicción a los posibles errores y fallas que puedan darse en el sistema una vez implantado, es una manera de vaticinar posibles errores.

⁵ WIKIPEDIA, Glosario, http://gl.wikipedia.org/wiki/Programaci%C3%B3n_Extrema, Noviembre del 2006, [XP.- Es una metodología que trata de un proceso ágil de desarrollo de software, es una aproximación a la ingeniería de software formulada por Kent Beck.]

⁶ WIKIPEDIA, Glosario, <http://es.wikipedia.org/wiki/RUP>, Noviembre del 2006, [RUP.- Proceso Racional Unificado, es un proceso de desarrollo de software que junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.]

⁷ MICROSOFT, Apéndice C: Guía de entrega, <https://www.microsoft.com/latam/technet/articulos/wireless/ap03.msp>, Noviembre del 2004, [MSF.- Microsoft Solution Framework, combina prácticas recomendadas del diseño y desarrollo de software con la creación e implementación de infraestructuras en un solo ciclo de vida de proyecto para dirigir soluciones de tecnología de todo tipo.]

- Refabricación: Facilita la reutilización de código, mediante la utilización de patrones, modelos, estándares, permitiendo flexibilidad al cambio.
- Programación en pares: Es una particularidad de esta metodología, alentando a que dos desarrolladores participen del proyecto en una misma estación de trabajo, cada miembro lleva a cabo la acción que el otro está haciendo ese momento.

1.1.2.1.2 Valores de la Programación Extrema.

- Comunicación: Permite que tanto los usuarios finales como los desarrolladores tengan una visión unísona de la aplicación, beneficiando la comunicación verbal frecuente, diseños fácilmente entendibles y simples.
- Simplicidad: Motiva la realización de diseños simples y la reconstrucción de aplicaciones.
- Realimentación: La retroalimentación se debe hacer a nivel de sistema, consumidor y equipo
- Coraje: Para no quedarse en el simple diseño y pasar a la ejecución.

1.1.2.1.3 Principios:

- Realimentación veloz: El más utilizado de los principios, con una buena comunicación con el cliente evita retrasos posteriores y cambios de requerimientos.
- Modificaciones incrementales: Asumiendo la solución de cada problema.
- Asunción de simplicidad: Los desarrolladores deben estar listos para cualquier cambio en los requerimientos.
- Trabajo de calidad: Asegura la calidad del software a entregar.

1.1.2.1.4 Actividades:

- Codificar: Plasmar las ideas en código base, esta programación expresa la solución al problema interpretado.

- Hacer Pruebas: Es la oportunidad de saber si el problema se encuentra bien interpretado mediante la codificación y detectar posibles errores, las pruebas deben ser exhaustivas y recreando los escenarios más remotos a los que se enfrentará el software.
- Escuchar: Los programadores no saben todo por lo tanto es indispensable escuchar las necesidades, opiniones y sugerencias de cambio realizadas por los clientes.
- Diseño: Crea una estructura que organiza la lógica del sistema, un buen diseño permite que el sistema crezca con cambios en un solo lugar, los diseños deben ser sencillos y si alguno es complejo se lo debe desglosar en más sencillos.

1.1.2.1.5 Ventajas y desventajas de la metodología:

Esta metodología presenta grandes ventajas para proyectos que no disponen de mucho tiempo para ser implementados, pero que no pueden por esto sacrificar la calidad del mismo, contempla la elaboración de una documentación robusta y el crecimiento futuro del tamaño del proyecto.

El permitir la programación en pares es una ventaja importante cuando se posee un grupo de desarrollo, en este proyecto de titulación solo se cuenta con una persona por lo que puede considerarse una desventaja.

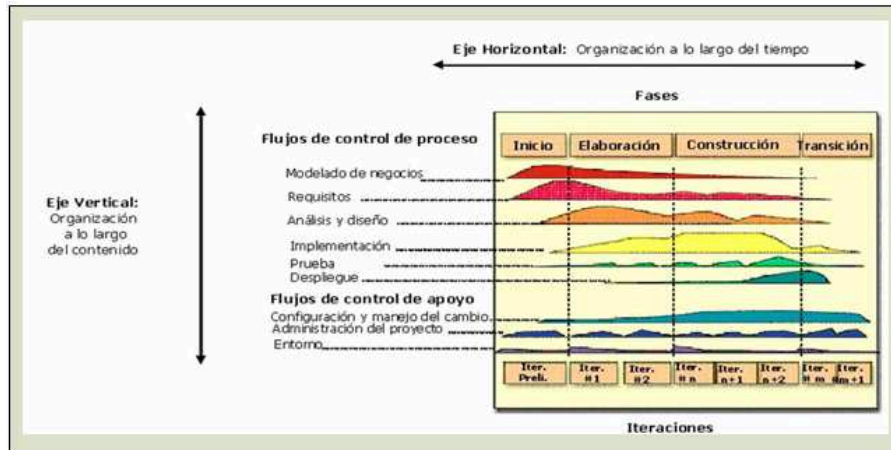
1.1.2.2 Descripción de la Metodología Rational Unified Process (RUP).

Es una metodología iterativa e incremental que designa tareas y responsabilidades coordinadamente, asegurando un producto de software de alta calidad, garantizando que el software esté de acuerdo con las necesidades del usuario final pues cada iteración soporta la realización de cambios en los requerimientos, con un costo y calendario predecibles, además todo el grupo de trabajo se involucra en las necesidades del cliente, de esta manera todos tienen claro el objetivo que debe alcanzar el producto final.

1.1.2.2.1 Estructura del Proceso:

RUP basa su metodología en un proceso iterativo e incremental que se puede apreciar en la Figura I - 1.

Figura I - 1 Fases e Interacciones de la Metodología RUP.



Fuente: WIKIPEDIA, Racional Unified Process, Noviembre del 2006.

1.1.2.2.2 Características Esenciales:

- Proceso dirigido por casos de uso: Son utilizados dentro de la metodología para capturar requerimientos, esto obliga al equipo de desarrollo en función del usuario final y no solamente en función de lo que sería bueno que contemple el sistema. Un caso de uso da un valor agregado ya que representan los requisitos funcionales del sistema. No solo inician el proceso de desarrollo del sistema sino que son un hilo conductor permitiendo establecer trazabilidad entre los artefactos que son generados en las diferentes actividades del proceso de desarrollo.
- Proceso centrado en la arquitectura: La arquitectura de la disposición u organización de las partes más relevantes, permitiendo una visión común de todos los involucrados y del sistema completo. Permite un mayor control del proceso de desarrollo ya que involucra los aspectos estáticos y dinámicos del proyecto, indica cómo debe ser construido el software, ayudando a la toma de decisiones.

- **Proceso Iterativo e Incremental:** Implica la división del proyecto en partes más pequeñas, cada una de estas partes puede ser visto como una iteración; una iteración es un recorrido más o menos completo a lo largo de todas las fases de desarrollo de software, mientras más incrementos tenga el sistema va tomando forma y ajustándose a las necesidades de el cliente final sin mayor inconveniente.

1.1.2.2.3 Mejores Prácticas:

- **Gestión de requisitos:** Es una guía para localizar, organizar, documentar y seguir los cambios de los requisitos funcionales del sistema y sus restricciones.
- **Desarrollo de software Iterativo:** Permite el desarrollo del proyecto, mediante la colocación de hitos claramente definidos en cada iteración, la repetición de actividades en cada iteración ayuda a poner atención especial en distintas fases del ciclo de vida.
- **Desarrollo basado en componentes:** Consiste en la división en componentes con interfases bien definidas que se ensamblarán posteriormente para obtener el todo del sistema.
- **Modelamiento visual del software:** Utiliza el lenguaje UML para especificar, construir y documentar los artefactos de un sistema de software.
- **Verificación continua de la calidad:** Se evalúa la calidad en algunos puntos durante el proceso de desarrollo, con mayor interés al final de cada iteración.
- **Gestión de cambios:** Los artefactos del software cambian no solo luego de la entrega en la etapa de mantenimiento, sino incluso durante el proceso de desarrollo impactando seriamente en los requerimientos, la gestión de los cambios se encarga de administrar las versiones, recursos del proyecto, herramientas, plataformas, entre otras.

1.1.2.2.4 Ventajas y desventajas para el proyecto.

RUP es una metodología fiable especialmente para proyectos medianos y grandes, el nivel de detalle, la cantidad de entregables y el tiempo que toma el desarrollar proyectos con esta metodología es una desventaja para un proyecto de este tipo.

El hecho de que sea iterativa e incremental es una ventaja porque nos permite depurar en cada una de las fases los requerimientos e irlos ajustando progresivamente a las necesidades y a los cambios que la aplicación sufre durante el proceso de desarrollo.

El proceso como tal requiere de experticia para aplicarlo correctamente, pero existe mucha información a disposición de todos con la que se puede trabajar paso a paso.

1.1.2.3 Descripción de la Metodología Microsoft Solution Framework (MSF).

Es una metodología que proporciona prácticas ensayadas para la planificación, creación e implementación de aplicaciones, centradas en los modelos de proceso y equipo, dejando de lado las elecciones tecnológicas. Provee a las organizaciones una serie de conceptos, modelos y mejores prácticas de uso interrelacionados de tal manera que controlan la planificación, desarrollo y administración de proyectos.

1.1.2.3.1 Componentes de MSF

- Principios:
 - Promover comunicaciones abiertas.
 - Trabajar una visión compartida.
 - Fortalecer los miembros del equipo.
 - Establecer responsabilidades claras y compartidas.
 - Focalizarse en agregar valor al negocio.

- Permanecer ágil y esperar los cambios.
 - Invertir en calidad.
 - Aprender de todas las experiencias.
- Disciplinas:
 - Gestión de proyectos: Describe el rol de la administración del proyecto dentro del rol de equipo, permite escalar proyectos pequeños a largos y complejos sin mayores inconvenientes.
 - Control de riesgos: Permite identificar prioridades y tomar decisiones estratégicas acertadas frente a emergencias que puedan surgir.
 - Control de Cambios: Se debe considerar a un cambio como un riesgo propio del proceso de desarrollo, donde los miembros del equipo deben ser preactivos y no reactivos.
 - Modelos:
 - Equipo de trabajo: Permite la reacción oportuna frente a cambios inesperados, asegurando que se involucre todo el equipo en la toma de decisiones, consiguiendo visualizar el problema desde todas las perspectivas.

Figura I - 2 Modelo de Equipo de Trabajo. Metodología MSF.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

- Procesos: La metodología por poseer una estrategia iterativa, proporciona una visión clara del estado del producto en cada etapa, identificando fácilmente el impacto de cualquier cambio en el

proyecto, administrándolo efectivamente y reduciendo al mínimo los efectos sobre el mismo.

Figura I - 3 Etapas y procesos de MSF



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

1.1.2.3.2 Ventajas y desventajas de usar la metodología MSF

- Permite un incremento en la productividad de las personas encargadas del proyecto.
- Presenta una estrategia integradora de herramientas, ayudas y contenidos de MSF, permitiendo un fácil mantenimiento.
- Usa estructuras ágiles incorporando un escenario básico para el desarrollo.

1.1.2.4 Comparación de las Metodologías para el proyecto.

En los puntos anteriores se ha descrito brevemente las principales metodologías con las que se pueden desarrollar proyectos software de toda magnitud, las principales ventajas y desventajas considerando la descripción inicial del problema.

A continuación se presenta una tabla comparativa de las metodologías frente a los requerimientos básicos del desarrollo de este proyecto. Se han asignado distintos pesos a los criterios con los que se procederá a la selección de la metodología según la importancia del criterio para las características antes descritas para el proyecto luego se evaluará cada uno de los criterios en las metodologías con los valores 1 - casi nada, 5 - bastante, 10- totalmente.

Tabla I - 1 Comparación de metodologías de Desarrollo de Software

<i>1.1.3 Características</i>	<i>1.1.3.1.1</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>
<i>1.1.3.1.5 Recomendada para proyectos de mediano a largo plazo</i>	<i>1.1.3.1.6</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>
<i>1.1.3.1.10 Calidad de software como uno de los objetivos principales</i>	<i>1.1.3.1.11</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>
<i>1.1.3.1.15 Cumplimiento con el ciclo de desarrollo completo</i>	<i>1.1.3.1.16</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>
<i>1.1.3.1.20 Documentación formal del proceso de desarrollo</i>	<i>1.1.3.1.21</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>
<i>1.1.3.1.25 Experiencia en el uso de la metodología</i>	<i>1.1.3.1.26</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>
<i>1.1.3.1.30 Permita cambios de requerimientos y configuraciones</i>	<i>1.1.3.1.31</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>
<i>1.1.3.1.35 Administración y previsión de riesgos</i>	<i>1.1.3.1.36</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>
<i>1.1.3.1.40 Arquitectura conducida</i>	<i>1.1.3.1.41</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>	<i>1.1.3.1.</i>

1.1.3.1.45 *Permita alcanzar un grado de certificación de software* 1.1.3.1.4 1.1.3.1. 1.1.3.1. 1.1.3.1.

1.1.3.1.50 *Agiliza la retroalimentación* 1.1.3.1.5 1.1.3.1. 1.1.3.1. 1.1.3.1.

1.1.3.1.55 *Iterativa e Incremental* 1.1.3.1.5 1.1.3.1. 1.1.3.1. 1.1.3.1.

1.1.3.1.60 *Permita la representación de sistemas complejos* 1.1.3.1.6 1.1.3.1. 1.1.3.1. 1.1.3.1.

1.1.3.1.65 *Totales* 1.1.3.1.6 1.1.3.1. 1.1.3.1. 1.1.3.1.

Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.*

Elaboración: Hidalgo, Karina.

Tomando en cuenta que la experiencia del desarrollador es uno de los factores determinantes para la selección de la metodología y luego de considerar los valores de la evaluación en la tabla comparativa de metodologías para la solución del problema descrito, tenemos que RUP es la metodología que más se ajusta a los requerimientos expuestos.

1.1.3.2 Descripción detallada de la metodología seleccionada para el proyecto.

Rational Unified Process (RUP) es una metodología para desarrollo de proyectos de software formales con alta calidad, creados a la medida de los usuarios finales del software con plazos y presupuestos predecibles.

Por ser desarrollado, mantenido y por ende respaldado por una empresa seria y de gran trayectoria como Rational, está constantemente actualizándose las mejores prácticas según las experiencias de los usuarios, garantizando el soporte, mejorando la rentabilidad del proyecto y aumentando la productividad de los desarrolladores por medio del acceso a la base del conocimiento, herramientas, plantillas, entre otros.

Es un proceso de software iterativo e incremental, dirigido por casos de uso y centrado en la arquitectura, que disciplinadamente asigna responsabilidades y tareas a los implicados en el desarrollo del proyecto.

El proceso unificado es una guía para los miembros del equipo de desarrollo de como administrar el desarrollo iterativo de manera controlada mientras se realimentan los requerimientos del negocio, es decir balancear los objetivos contrarios, gestionar riesgos mientras se produce software a satisfacción de los usuarios finales.

1.1.3.2.1 Características esenciales de RUP.

1. Proceso dirigido por casos de uso: Son además de una herramienta la guía para la fase de diseño, implementación y pruebas, es decir, son un elemento integrador.
2. Proceso centrado en la arquitectura: La arquitectura toma en cuenta elementos como la calidad del sistema, rendimiento, reutilización y capacidad de evolución. Se ve influenciada por la plataforma de software, sistema operativo, gestor de base de datos, protocolos de comunicación, todos estos son considerados requisitos no funcionales del sistema. El diseño de una buena arquitectura en fases tempranas permite que no se vea posteriormente afectada por cambios en otras etapas del proyecto,

mientras más avanzado esté el proyecto en el tiempo más robusta es la arquitectura.

3. Proceso Iterativo e Incremental: Cada iteración aborda una parte de la funcionalidad total del sistema, pasando por todos los flujos de trabajo relevantes, cada iteración es analizada y controlada la calidad al finalizarla. Se realiza cuantas iteraciones sean necesarias hasta lograr el producto a satisfacción del cliente en su versión actual.

1.1.3.2.2 Estructura Dinámica del proceso, fases e iteraciones.

RUP divide el proceso de desarrollo de software en ciclos, teniendo un producto al final de cada ciclo, cada ciclo se divide en cuatro fases que son: inicio, elaboración, construcción y transición, cada fase a su vez se subdivide en iteraciones que no es fijo en cada fase y concluye con un hito bien definido, un punto en el tiempo en el cual se deben tomar ciertas decisiones críticas. No todas las fases tienen la misma duración, esto depende básicamente de las características propias del proyecto.

Fase de Inicio

En esta fase se establece formalmente el modelo del negocio, oportunidades y alcance del proyecto. Se identifican las entidades externas como actores y casos de uso, de estos últimos se diseñan los más esenciales.

Entre los principales objetivos de esta fase está el establecer el ámbito, alcance y límites del proyecto, localizar casos de uso críticos y definir la funcionalidad básica del sistema, proponer la arquitectura básica, costear recursos y tiempo, además de estimar los riesgos.

Los entregables en esta fase son: Un documento con los requerimientos del sistema, alcance y restricciones, el modelo inicial de casos de uso, un glosario inicial de términos, el caso de negocio y un plan de contingencias conjuntamente con la lista de riesgos del proyecto.

Fase de Elaboración

En esta fase el objetivo es realizar un análisis minucioso del problema, desde varios puntos de vista, obteniendo una visión global, de tal manera que se logre la identificación y eliminación de la mayor cantidad de riesgos posibles, este conocimiento permite la elaboración del plan de proyecto y es la base establecer la arquitectura del sistema.

A partir de esta fase los requerimientos y la arquitectura son estables, es decir que se puede planificar el resto del proyecto sin temor a equivocaciones ya que la incertidumbre disminuye y las estimaciones son más certeras. Se debe tratar de abarcar todo el proyecto profundizando solamente en los puntos críticos de la arquitectura y riesgos importantes.

Los entregables de esta fase son: Un modelo casi completo de casos de uso, requerimientos funcionales adicionales que no necesariamente se asocian con los casos de uso, una clara descripción de la arquitectura del sistema, la planificación del proyecto, un prototipo de la arquitectura, lista de riesgos y casos del negocio, finalmente un manual preliminar de usuario.

Fase de Construcción

En esta fase el objetivo principal es desarrollar los componentes adicionales que se incorporarán al proyecto. Las pruebas del sistema se las realiza con mayor detalle. Permite la productividad más que el diseño intelectual del sistema, se puede desarrollar paralelamente, pero esto debe estar considerado dentro de la arquitectura y la planificación.

Se debe conseguir en esta etapa la calidad requerida en el menor tiempo posible, mediante la optimización de recursos, alcanzado versiones funcionales tan rápido como sea posible.

Los entregables de esta fase son: El producto de software completo y corriendo en la plataforma adecuada, manuales de usuario y la descripción de la versión actual del producto.

Fase de Transición

En esta fase el objetivo principal es entregar el producto al usuario final, de allí surgirán nuevos requerimientos empujando a la elaboración de nuevas versiones y ciclos de vida. Incluye completar la documentación, realizar las pruebas Beta para validar el producto con las expectativas del usuario final, funcionamiento en paralelo con sistemas antiguos, conversión de los datos existentes, capacitación a los usuarios en el manejo del nuevo sistema y la distribución de producto.

Mejores Prácticas

Desarrollo Iterativo

El desarrollo de software mediante la utilización de modelos lineales no es objetivo cuando se trata de sistemas novedosos y complejos, los modelos iterativos como RUP permiten incrementar los requerimientos mientras crece el sistema. Aborda primeramente las tareas de mayor riesgo, reduciendo los riesgos del sistema.

Administración de Requerimientos

La metodología proporciona información sobre cómo obtener los requerimientos del sistema, elaborar documentación, captar y comunicar los requerimientos del negocio. La elaboración de casos de uso son probados para el levantamiento de requerimientos y guiar el diseño, implementación y pruebas del sistema.

Arquitecturas basadas en componentes

Se basa en elaborar una arquitectura flexible, fácilmente modificable, comprensible de manera intuitiva, que permita la reutilización de componentes. La metodología promueve la reutilización de componentes recién creados y preexistentes.

Modelamiento Visual

Es la estructura y comportamiento tanto de la arquitectura como de los componentes. Se maneja mediante bloques de construcción que ocultan detalles, permiten la comunicación entre los miembros del equipo de desarrollo, analizan la consistencia. UML es la base para realizar los modelos que propone RUP.

Verificación de cualidades

Permite elaborar software de calidad, rendimiento y confiabilidad. La metodología permite administrar la calidad del producto que debe ser el resultado de todo el proceso de ingeniería y no de un determinado grupo de personas.

Control de cambios

Antes de realizar un cambio en el sistema se debe analizar si es indispensable y rastrear el impacto en la totalidad del sistema, es posible con esta metodología el realizar cambios por ser iterativo ya que hay cambios que son necesarios.

1.2 SELECCIÓN DE UNA ARQUITECTURA DE DESARROLLO DE SOFTWARE MULTIPLATAFORMA.

La arquitectura de software es un conjunto de patrones y abstracciones que permiten con seguridad y fiabilidad mediante un marco referencial conseguir el propósito de construir aplicaciones de software que administren información mediante parámetros comunes a los analistas, programadores, diseñadores, arquitectos, entre otros, que permitan alcanzar el objetivo planteado.

Las arquitecturas para el desarrollo de software se dividen básicamente en arquitecturas Monolíticas para sistemas centralizados y múltiples para sistemas distribuidos que se subdividen principalmente en arquitecturas cliente – servidor y n capas, que se describirán brevemente a continuación, luego se seleccionará las tecnologías con las que se hace posible el desarrollo de este tipo de arquitecturas.

La arquitectura seleccionada conjuntamente con la tecnología seleccionada deberá ser soportada por Microsoft BizTalk Server 2006 que es el integrador sobre el que se orquestrará.

1.2.1 ARQUITECTURAS DE COMUNICACIÓN

1.2.1.1 Arquitecturas Monolíticas o Centralizadas

Las arquitecturas de tipo monolíticas, permite el desarrollo de sistemas centralizados, es decir se coloca todo el aplicativo en un solo equipo, mainframe con sistema operativo multiusuario pero con un solo procesador central, a este último se le conectan terminales tontas que no poseen unidad central de proceso, cuando no se conectan terminales se ejecutan las aplicaciones en un solo computador sin necesidad de conexión red, son poco orientados a procesos de negocio y generalmente no son documentadas, de allí que mientras más grande es el sistema y un mayor grado de tecnificación requieren, son menos usadas y su tendencia es a desaparecer.

El hecho de que se encuentre en un solo computador todo el sistema con sus elementos lo hace altamente portable y fácilmente administrables, pero poco práctico cuando se trata de que varias personas tengan acceso al mismo, son poco re-usables para ampliaciones o cambios de requerimientos.

1.2.1.2 Arquitectura Cliente – Servidor

Un sistema de información se divide en un nivel de presentación y en un nivel de administración de la información como tal.

La arquitectura cliente – servidor, consiste en tener conectados un servidor que provee servicios mediante una interfaz de conexión en red y uno o varios clientes que se conectan con él ya sea para acceder a aplicativos que generalmente se publican en él, acceder a bases de datos, información contenida en archivos, entre otros.

El servidor provee de servicios y los clientes son consumidores de los mismos, si se tiene en cuenta que el servidor divide las responsabilidades de procesamiento de la información, encargando la tarea de administración de la información a los clientes y encargándose de la interfaz de usuario en el nivel de presentación, se puede decir que se optimiza tiempos de respuesta y el trabajo de las personas que mediante sus clientes procesan parte de los datos que se gestionan.

Un servidor además de soportar varios clientes concurrentes actuando como una sola entidad, puede actuar como cliente al conectarse con otro servidor, un cliente puede ser servidor de otros clientes conectados en la red, o pueden operar como entidades claramente separadas, un servidor y sus clientes pueden estar corriendo sobre distintos sistemas operativos, todo depende del diseño de la red y aplicación.

El servidor puede ser dedicado, es decir ejecuta una tarea o brinda un servicio definido, por ejemplo puede ser servidor de aplicaciones, de Internet, de correo electrónico, de bases de datos, etc. O puede brindar varios servicios a la vez, facilitando las actualizaciones ya que se realizan de forma transparente al usuario final.

Básicamente permite la distribución de sistemas, descentralizando y soportando una arquitectura informática heterogénea, que incluye múltiples plataformas, aplicaciones, sistemas operativos, estándares de comunicación proveedores, arquitecturas propietarias y no propietarias, todas funcionando de manera concurrente.

1.2.1.3 Arquitectura n Capas

La arquitectura de n capas es de las más usadas para crear aplicaciones en ambientes distribuidos multiplataforma, consiste en uno o varios servidores conectados con múltiples usuarios, brindando servicios varios, a diferencia de la arquitectura cliente – servidor se colocan varias capas de la aplicación en distintas máquinas lo que permite un desarrollo en paralelo de cada capa de las aplicaciones, son más robustas por el encapsulamiento, modulares, el mantenimiento y soporte son mucho más sencillos.

Son la siguiente etapa de las arquitecturas cliente – servidor, posee mayor flexibilidad para realizar cambios pues está basado en componentes y módulos, la escalabilidad se logra simplemente colocando más hardware sin necesidad de

colocar código para conseguir este objetivo, es decir puede atender muchas peticiones a la vez sin degradar los tiempos de respuesta.

Soporta varios estándares corriendo en las múltiples plataformas de distribución y trabajan con clientes ligeros, es decir no necesariamente son computadores los clientes sino que pueden ser equipos que puedan acceder a Internet como celulares, Personal Digital Assistants (PDAs), entre otros, se encuentran generalmente disponibles las 24 horas del día.

En una arquitectura en capas, sin importar de cuántas sean, los elementos contenidos dentro de ellas pueden comunicarse entre sí y con las demás con algunas variaciones que se describen a continuación:

1.2.1.3.1 Arquitectura Top – Down en capas.

Este modelo de arquitectura soporta una cascada de peticiones o solicitudes entre capas, ya que todos los elementos que componen la capa $n+1$ pueden enviar solicitudes a la capa inferior n . Es una arquitectura no estricta.

1.2.1.3.2 Arquitectura Bottom – Up en capas.

Este modelo de arquitectura soporta el manejo de eventos pues a cada uno de los elementos que compone la capa n le está permitido notificar a elementos de la capa $n+1$ la ocurrencia de algún evento de interés. Esto es muy útil especialmente cuando la aplicación soporta dispositivos móviles.

1.2.1.3.3 Arquitectura Bidireccional de capas.

En este modelo de arquitectura generalmente se involucran dos pilas de N capas con la capacidad de comunicarse entre si.

Típicamente cuando se realizan desarrollos en capas se lo hace en un número de tres, cuya estructura es la siguiente:

Capa de Presentación:

Es la capa que se encuentra en directa acción con el usuario final, esta capa soporta las interfaces de comunicación, es decir manejan ventanas, menús, gráficos, reportes, multimedia, entre otros.

Capa de Dominio de la Aplicación:

Es la capa en la que se automatiza los procesos de la lógica del negocio que ejecutarán los usuarios mediante la capa de presentación. Ejecuta código donde se implementan las reglas del negocio, restricciones, tareas, entre otras.

Capa de Repositorio:

Es la capa en la que se administran toda la información almacenada en el motor de base de datos independientemente de la herramienta que lo administre, con las seguridades que cada una de ellas posee para la fidelidad de los mismos, es decir, maneja persistencia, backups, entre otros.

El manejar arquitectura de capas independientemente de su modelo y número de capas implica seguir los lineamientos propuestos por la misma para un eficiente trabajo y conseguir las ventajas descritas anteriormente.

Por todo lo antes descrito y teniendo en cuenta que lo que en este proyecto de titulación se realizará es una orquestación e integración de aplicaciones multiplataforma, la arquitectura que además de ser la más moderna es la más apropiada a todas luces es la arquitectura de capas con todas las ventajas que esta brindará al desarrollo del proyecto, por lo que a continuación se seleccionará una tecnología que soporte esta arquitectura y que además sea soportada por Microsoft BizTalk Server 2006.

1.2.2 SELECCIÓN DE LA TECNOLOGÍA PARA IMPLEMENTAR LA ARQUITECTURA MULTIPLATAFORMA.

1.2.2.1 Web Services.

Un Servicio Web (web service en Inglés) es una colección de protocolos y estándares que permiten el intercambio de datos entre aplicaciones que no necesariamente fueron desarrolladas en el mismo lenguaje de programación ni sobre protocolos iguales, pero si se las ejecuta sobre una plataforma que use arquitectura con servicios web, que intermedien entre ellas para intercambiar datos entre redes de computadores se las puede integrar.

WS-I es el organismo encargado de desarrollar varios perfiles para definir de manera exhaustiva los estándares abiertos que utilizan los web services para conseguir interoperabilidad y los responsables por la arquitectura y reglamentación de los servicios web son las organizaciones OASIS⁸ y W3C⁹.

1.2.2.1.1 Estándares que poseen los Web Services:

- *WS-Security*: Garantiza la autenticación de las aplicaciones y la confidencialidad de los mensajes transmitidos, es un protocolo de seguridad y es aceptado como estándar.
- *XLM*: Archivo que contiene datos que pueden ser intercambiados.
- *Web Services Protocol Stack*: Conjunto de servicios y protocolos de los servicios web.
- *WSDL*: Es la interfaz pública basada en XML de los requisitos funcionales que permite establecer la comunicación entre servicios web.
- *UDDI*: Permite a los web services comprobar la disponibilidad de las aplicaciones, es un protocolo para la ubicación de los servicios web.
- *SOAP*: Protocolo sobre los que se establece el intercambio.
- Además se pueden transmitir datos en XML sobre protocolos comunes como HTTP, FTP o SMTP.

⁸ HOMEPAGES, Glosario, <http://homepages.mty.itesm.mx/al450951>, [OASIS: Organization of the Avancement Structures Information Standards, de los avances de la información estructurada, trabajan con tecnologías XML y sus mejoras, consiste en más de 400 miembros.]

⁹ DIAZ TOLEDANO Moisés, Web Services. Introducción y Escenario para su Uso, [W3C: Organismo encargado de desarrollar gran parte de los estándares para Internet.]

1.2.2.1.2 Ventajas de usar Web Services:

- Es una arquitectura que soporta la integración de aplicaciones multiplataforma, aportando interoperabilidad entre aplicaciones de software independientemente de sus propiedades.
- Es un estándar basado en XML, lo que permite mayor flexibilidad al momento de integrar sistemas autónomos o en red y de realizar cambios en los paquetes de intercambio.
- Permite la utilización de estándares abiertos y protocolos basados en texto, por lo que es fácil entender su funcionamiento.
- Soporta HTTP permitiendo el aprovechamiento de sistemas de seguridad sin necesidad de cambiar las reglas de filtrado.
- Facilita la integración de servicios de aplicaciones ubicadas geográficamente distantes y de diferentes compañías gracias a los protocolos estándar.

1.2.2.1.3 Desventajas de los Web Services

- El rendimiento es más bajo en comparación a modelos computacionales distribuidos como CORBA, RMI o DCOM por su formato basado en texto y por su grado de desarrollo menor a estándares abiertos como los antes mencionados.
- El apoyarse en HTTP puede evadir medidas de seguridad de los firewalls.

1.2.2.2 Tramas ISO 8583.

ISO 8583 es una estándar que permite hacer transparente la transmisión de información, utilizada especialmente en sistemas transaccionales. Es un estándar para transacciones financieras mediante mensajes, es decir da las especificaciones para el intercambio de mensajes, es regido por la Organización Internacional de Estandarización, creado para el intercambio de transacciones bancarias especialmente las realizadas con tarjetas.

Este intercambio de información se realiza sobre redes diseñadas para soportar este tipo de estándar, es una herramienta que pese a ser segura en su transmisión es poco flexible en el momento de realizar cambios en los códigos de programación ya sea por modificaciones o ampliaciones al software.

1.2.2.2.1 Ventajas de usar Tramas ISO8583

- Permite una comunicación estándar entre aplicativos independientemente de la plataforma sobre la que se ejecuten.
- Por la manera de transmisión la comunicación es segura permitiendo su utilización en sistemas transaccionales con tarjetas.
- Es usado frecuentemente en sistemas transaccionales.

1.2.2.2.2 Desventajas de usar Tramas ISO8583

- Es poco flexible a los cambios en los requerimientos de software.
- Implica muchas horas de trabajo el realizar pequeñas ampliaciones del sistema o modificar los campos en la base de datos.
- Se encuentra en sistemas transaccionales antiguos.

1.2.2.3 Archivos XML.

XML es el acrónimo de eXtensible Markup Lenguaje que significa lenguaje de marcas extensibles, se lo denomina así por ser un metalenguaje basado en etiquetas, no solo se lo puede usar en Internet sino como un estándar para el intercambio de información estructurada entre diferentes plataformas, permitiendo la compatibilidad de entre sistemas para compartir información de manera segura, fácil y fiable. Usando XML se pueden crear archivos que contienen este tipo de lenguaje permitiendo una programación flexible.

1.2.2.3.1 Ventajas de usar Archivos XML:

- Fácil de leer, editar, implementar y aplicar a distintos sistemas independientemente de la plataforma en la que esté implementado.
- Permite la transferencia de información ya que puede ser escrito en un archivo plano con datos requeridos por una aplicación para que estos sean utilizados por otra aplicación con la que está estableciendo comunicación de manera segura y fácil, flexibilizando el manejo de información.
- Es sencillo mover datos y realizar cambios en el código de programación al pasar como un parámetro un archivo XML, además facilita la migración de datos entre bases de datos por la comunicación estándar que maneja.
- Es frecuente la utilización de archivos de tipo XML en sistemas transaccionales modernos por la facilidad, seguridad, flexibilidad que ofrece.
- El archivo de tipo XML puede comprimirse para transferirlo.

1.2.2.3.2 Desventajas de usar Archivos XML:

- El tamaño de los ficheros requiere de un computador con buenas características en disco.

1.2.3 SELECCIÓN DE LA TECNOLOGÍA QUE SOPORTA LA ARQUITECTURA

Teniendo en cuenta la arquitectura de n capas a utilizar, la descripción breve realizada anteriormente de las tecnologías que la soportan y sin olvidar la caracterización del problema realizado en un inicio, a continuación se presenta una tabla comparativa de las tecnologías para su selección.

Tabla I - 2 Comparación de Tecnologías.

<i>Características</i>	<i>1.2.3.1.1</i>	<i>1.2.3.1.2</i>	<i>W</i>	<i>1.2.3.1.3</i>	<i>1.2.3.1</i>
			<i>eb</i>		
			<i>Se</i>		
			<i>rvi</i>		
			<i>ce</i>		

1.2.3.1.5	Soporta la integración de múltiples plataformas	1.2.3.1.6	1.2.3.1.7	10	1.2.3.1.8	1.2.3.1
1.2.3.1.10	Flexible al momento de integrar o realizar cambios	1.2.3.1.11	1.2.3.1.12	10	1.2.3.1.13	1.2.3.1
1.2.3.1.15	Soporta estándares abiertos y protocolos varios	1.2.3.1.16	1.2.3.1.17	10	1.2.3.1.18	1.2.3.1
1.2.3.1.20	Transmisión segura de información	1.2.3.1.21	1.2.3.1.22	10	1.2.3.1.23	1.2.3.1
1.2.3.1.25	Puede ser escrito en archivos planos	1.2.3.1.26	1.2.3.1.27	10	1.2.3.1.28	1.2.3.1
1.2.3.1.30	TOTAL	1.2.3.1.31	1.2.3.1.32	10	1.2.3.1.33	1.2.3.1

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tomando en cuenta la experiencia del desarrollador, la caracterización del problema, las distintas plataformas en las que se debe implementar para poder comprobar el poder de orquestación de BizTalk Server, se encuentra basado en XML y la valoración realizada en la tabla comparativa de tecnologías Web Services es la opción más indicada.

1.2.4 DESCRIPCIÓN DETALLADA DE LA ARQUITECTURA SELECCIONADA PARA EL PROYECTO

1.2.4.1 Web Services

Servicios Web o Web Services (en inglés) fueron creados para hacer accesibles los programas y aplicaciones que utilizan interfaces web a los clientes en tiempo real y trabajando sobre los datos originales de manera directa.

Un web service permite la comunicación entre aplicaciones o sus componentes de manera estándar, mediante la utilización de protocolos comunes independientemente del lenguaje de programación, plataforma de implantación o sistema operativo. Es una aplicación web que posee un identificador, las interfaces y los estándares que utiliza pueden ser definidos y configurados utilizando artefactos XML, soportando la interacción directa con aplicaciones Web, éstas son accesibles de manera remota mediante la utilización de mensajes XML transmitidos sobre protocolos estándar.

Utiliza un método de encapsulación de funciones específicas permitiéndoles ser utilizadas en otros servidores, teniendo en cuenta que los web services no son aplicaciones con interfaz gráfica con la que los usuarios pueden interactuar, son software accesible en Internet por otras aplicaciones web. Es así que permite el desarrollo de aplicaciones que hagan uso de la tecnología de web services, éstas pueden estar implantadas en cualquier lenguaje de programación, se las coloca para ser accedidas vía web para que se realice de esta manera la interacción con los web services. Mediante la interacción entonces las páginas web dejan de ser simples expositores de información, con diseños gráficos agradables al usuario, sino que permite interactuar con la misma brindando información, cálculos específicos, gestión de datos, entre otras tareas.

Los servicios web no son la solución a todos los problemas y menos a los de integración, en su estructura están diseñados básicamente para tres contextos:

- Brindar servicios públicos de manera simple.- Cuando se expone una funcionalidad sin necesidad de contraseñas al público en general, un claro

ejemplo de esto es el sistema de información sobre itinerarios y costos de boletos aéreos vía web, donde la interfaz gráfica presenta información como logotipos, textos, botones interactivos para una navegación ordenada, por secciones, para la consulta de disponibilidad de vuelos y rutas solicita la selección de la ruta a tomar y el horario la interfaz web, interactúa con el web service para saber la disponibilidad de viajar en el vuelo solicitado y los costos que tendría el hacerlo internamente (estas secuencias de tareas no se presentan al usuario) y finalmente la interfaz presenta estos datos al solicitante.

- Permite la integración de aplicaciones.- Un web service permite la integración, interacción, intercambio y extensión de sistemas previamente construidos, independientemente de la plataforma y lenguaje de programación, deben ser las aplicaciones, simplemente las que tengan la lógica de negocio similar y la integración por medio de web services se la puede realizar. Así se puede integrar software muy diverso y hasta se pueden realizar operaciones transaccionales al interior de la empresa y con socios comerciales.
- Soporta Sistemas de Grid Computing.- Para los problemas complejos cuya resolución requieren de multiprocesadores o computadores funcionando sincronizadamente, permitiendo la división en problemas más pequeños y repartiéndolos en nodos de computación distintos y uniéndolos nuevamente las soluciones.

1.2.4.2 Tecnologías en las que se basan los Web Services

- Formato.- Para que describa la interfaz del componente basado en XML.
- Protocolo de Aplicación.- Basado en mensajes, permite que la aplicación interactúe con el o los web services.
- Protocolo de Transporte.- Transporta los mensajes por Internet.

1.2.4.3 Razones básicas para usar Web Services.

- Son flexibles y soportan cambios de requerimientos futuros.

- Está basado en http sobre TCP en el puerto 80, dado que los firewalls cierran la mayoría de los puertos mas no el 80 para poder acceder al tráfico de Internet las aplicaciones que interactúan con web services no se ven bloqueadas.
- Las interfaces no eran buenas, con SOAP se ha logrado un avance para acceder a las funcionalidades de otros computadores en red.
- Aportan independencia entre las aplicaciones que usan web services.

1.2.4.4 Servidores de Aplicaciones para Web Services.

- Axis y Jakarta Tomcat de Apache
- ColdFusion MX de Macromedia
- Java Web Services Development Pack de Sun Microsystems
- JOnAS de Código Abierto
- Microsoft .NET
- Novell extend basado en J2EE
- Web Logic
- Web Sphere
- Zope
- Mono

1.3 SELECCIÓN DE LAS HERRAMIENTAS DE DESARROLLO.

En esta sección se describe detalladamente la herramienta de Base Microsoft Biztalk 2006 que se utilizará para la integración y orquestación de las aplicaciones antes mencionadas, luego se describirán las herramientas más utilizadas durante el proceso de desarrollo en la fase de implantación de software de manera formal como son: J2EE, .NET y su selección para la integración.

Como se trata de Microsoft BizTalk Server en su versión 2006, que es un integrador de multiplataformas se requiere de al menos dos herramientas distintas para demostrar su potencial, luego de la descripción se procederá a la selección de las herramientas que más se ajusten a la descripción de requerimientos realizada en pasos anteriores para cada uno de los sistemas a integrar y para la programación de componentes que soportará el integrador.

1.3.1 HERRAMIENTAS DE BASE

1.3.1.1 Descripción de Microsoft Biztalk Server 2006 como Plataforma de Integración.

Es poco frecuente encontrar una organización próspera que no tenga al menos un sistema para cada uno de los departamentos que la conforman, especialmente por la velocidad impuesta por el mercado que obliga a tener una mejor visión del negocio en un ambiente globalizado y altamente competitivo, llevando a las organizaciones a que sus sistemas integrados sean más eficientes, sometiéndolos no solamente a simples conexiones entre sistemas con código y protocolos, sino implementando herramientas de infraestructura que impongan una ventaja competitiva frente a la competencia.

Orquestar los sistemas en un lenguaje común para los analistas de negocios, el arquitecto de aplicaciones y los desarrolladores, con un enfoque totalmente distribuido es lo más complicado en proyectos de integración de mediana o alta complejidad.

BizTalk Server 2006 ayuda con estos procesos demorosos y confusos, permitiendo la conexión entre las distintas aplicaciones de software tanto a nivel

interno de la organización como con sus socios comerciales al exterior de la misma, creando y modificando los procesos lógicos que usan las aplicaciones. Permite la realización de monitoreos de procesos en tiempo de ejecución, tareas varias que se orientan a cumplir con los objetivos del negocio, entre otras que se describen a continuación.

1.3.1.2 Características de Biztalk 2006.

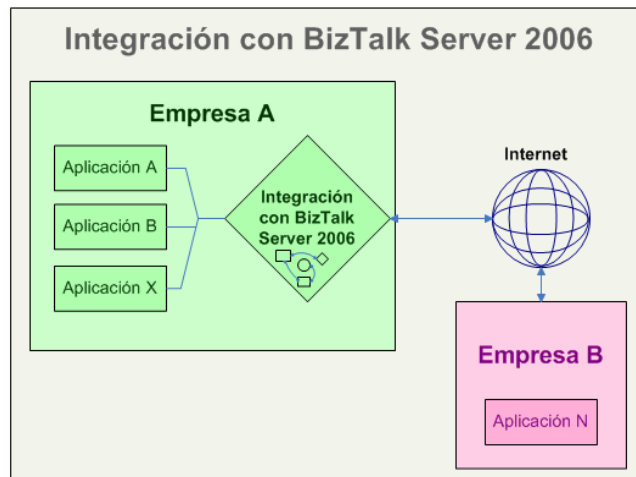
“BizTalk Server 2006 es el producto de servidor de referencia para la construcción de soluciones de proceso de negocio e integración”¹⁰. Es el producto estrella de Microsoft y es el corolario de las experiencias de los usuarios con sus 3 versiones predecesoras, sugerencias, enmiendas e innovación.

Es un producto que administra de manera eficiente los procesos de negocio independientemente de sus plataformas, lenguajes y protocolos. Suministra servicios de mensajería, orquestación de aplicaciones y poderosas herramientas de desarrollo. Proporciona un ambiente seguro y robusto para la ejecución de procesos de negocio.

Es una herramienta que utiliza los componentes más avanzadas de Microsoft y estándares de calidad que se encuentran vigentes en el mercado para la gestión, automatización e integración de procesos de negocio tanto al interior de la empresa entre departamentos como con el exterior facilitando la comunicación e integración con socios comerciales como se muestra en la siguiente figura.

Figura I - 4 Modelo de Integración con BizTalk Server 2006.

¹⁰ MICROSOFT, Novedades en BizTalk Server 2006, www.microsoft.com/spain/biztalk/evaluation/whatsnew.msp, junio del 2006.



Fuente: MICROSOFT Corporation, *Deploying and Managing Business Process and Integration Solutions Using Microsoft BizTalk Server 2006*, Junio, 2006.

Elaboración: Hidalgo, Karina.

1.3.1.3 Servicios y herramientas que ofrece Biztalk Server 2006

- Servicios de mensajería: Crea, modifica y enruta mensajes hacia y entre los procesos de negocio.
- Servicio de orquestación: El motor de orquestación es el responsable de procesar documentos y una automatización de las aplicaciones workflow¹¹.
- Herramientas de desarrollo de aplicaciones: Ofrece un número de herramientas gráficas que permiten los desarrolladores ejecutar tareas como la creación, pruebas y desarrollo de esquemas, mapas y orquestaciones.
- Motor de reglas del negocio: Las reglas del negocio se asocian en tiempo de ejecución, el motor habilita la dinámica de las políticas del negocio y la lógica de integración con los procesos sin la necesidad de re grabar y re deplorar las aplicaciones Biztalk.
- Servicio de integración web¹²: Biztalk habilita web services¹³ para la orquestación de los procesos de negocio, para los procesos de negocio que

¹¹ WIKIPEDIA, Glosario, <http://es.wikipedia.org/wiki/Workflow>, Noviembre del 2006, [Workflow.- Flujo de trabajo (en español) es el estudio de los aspectos operacionales de una actividad de trabajo: cómo se estructuran tareas, cómo se realizan, cuál es el orden correlativo, cómo se sincronizan, cómo fluye la información que soporta las tareas y cómo se hace el seguimiento al cumplimiento de las tareas]

se crearon basados en web para clientes, soportando así todos los estándares y protocolos incluyendo soporte para los mismos.

- Monitoreo de Actividades del negocio (BAM): Es usado para el monitoreo en tiempo real, permite archivar los datos y estadísticas de cada uno de los procesos de negocio.
- Servicios de Actividades del negocio (BAS): Habilita la integración entre socios de negocios, brindando una administración amigable.
- Actividad de intercambio de mensajes: Es usado como monitor y ejecutor de la actividad de mensajes y orquestación, mediante una consola de administración.

1.3.1.4 Descripción del Motor de Biztalk Server 2006

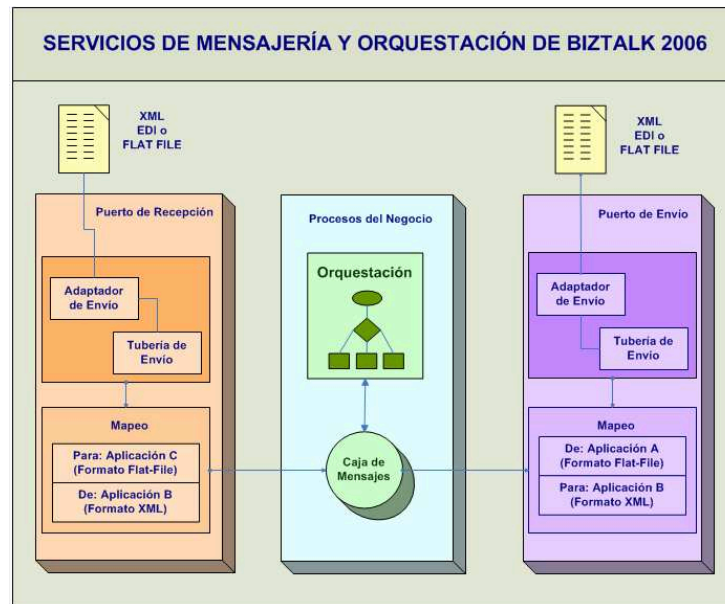
La arquitectura de BizTalk Server 2006 es un compendio de varias tecnologías ya que el armonizar varios sistemas con distintas plataformas en un unísono que administre de manera efectiva el proceso de negocio es extremadamente complicado.

Es así que podemos describir las partes del Motor de BizTalk Server 2006 mas no la arquitectura, la siguiente figura ilustra los componentes del motor:

Figura I - 5 Motor BizTalk Server 2006.

¹² WIKIPEDIA, Glosario, <http://es.wikipedia.org/wiki/Web>, Noviembre del 2006, [WEB.- La World Wide Web (www), es un sistema de navegador web para acceder a una página, sitio o conjunto de sitios que proveen información por los medios descritos, o la “Web”, que es la enorme red interconectada disponible prácticamente en todos los sitios de Internet.]

¹³ WIKIPEDIA, Glosario, http://es.wikipedia.org/wiki/Servicios_Web, Noviembre del 2006, [Servicio Web.- Un web service (en inglés) es una correlación de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes y ejecutadas sobre cualquier plataforma pueden utilizar web services para intercambiar datos en redes de ordenadores como Internet.]



Fuente: MICROSOFT Corporation, *Deploying and Managing Business Process and Integration Solutions Using Microsoft BizTalk Server 2006*, Junio, 2006.

Elaboración: Hidalgo, Karina.

El servicio de Mensajería y el Motor de orquestación de BizTalk Server 2006 son los principales servicios sobre los que se basa la arquitectura, para la integración e intercambio de mensajes con todos los tipos soportados de sistemas externos y aplicaciones existentes tanto en la organización como con los socios comerciales.

1.3.1.4.1 Motor de Mensajes

La mensajería es una de las principales operaciones que maneja BizTalk, ya que la manera de establecer la comunicación entre aplicaciones es mediante el intercambios de mensajes, para que este intercambio sea efectivo se realiza las tareas de recepción de los mensajes entrantes, identifica los formatos en los que llega la información, extrae las claves de identificación además de identificar las reglas de ruteo aplicadas, entrega los documentos a los respectivos destinatarios incluyendo los puertos u orquestación, finalmente documentos de pistas.

Este componente de mensajería facilita la comunicación con todos los aplicativos, bases de datos y software a orquestar, mediante los distintos adaptadores que posee BizTalk que soportan protocolos de comunicación, servicios web, entre otros.

1.3.1.4.2 Base de Datos de Mensajería

La Caja de Mensajes no es más que una base de datos que funciona sobre SQL Server, ésta es usada por BizTalk como un repositorio de datos y rutas de mensajes que se utilizan para la orquestación y puertos de envío. La base de datos de mensajería tiene metadatos asociados, cuando los mensajes llegan a este punto son asociados con la metadata y evaluados para determinar el tipo de servicio suscrito con este mensaje.

1.3.1.4.3 Modelo de Publicadores – Suscriptores

Es un modelo implementado por BizTalk Server 2006 para el enrutamiento de mensajes, los proveedores de los mismos denominados publicadores envían éstos a la caja de mensajes allí los suscriptores (envían los puertos y la orquestación) pudiendo suscribir mensajes específicos. Son recibidos en la caja de mensajes y reenviados a todos los suscriptores interesados.

1.3.1.4.4 Orquestación con BizTalk

La orquestación es utilizada para ejecutar y administrar los procesos de negocio, representa de manera detallada los procesos de negocio que luego son implementados programáticamente. La orquestación está designada para usar formas tales como enviar, recibir, transformar, construir y decidir que coordinadas entre los sistemas requieren para su comunicación, proceso de datos en el motor de mensajes.

Se puede decir que la orquestación permite la administración y ejecución de distintos procesos, éstos son construidos sobre los procesos de mensajería, poseen interfaz gráfica e implementan la lógica del negocio.

1.3.1.5 Lo Nuevo de Biztalk Server 2006

La versión 2006 de BizTalk Server ha tomado tiempo en desarrollarse para que sea compatible con las nuevas versiones de Microsoft SQL Server 2005 y de Microsoft Visual Studio 2005. Encontrándose construido sobre Microsoft .NET 2.0 Framework¹⁴ permite que BizTalk use todo el potencial del Common Language Runtime (CLR), Common Type Specification (CTS), Common Language Specification (CLS) y de las Librerías de Clase Bases (BCL), sacando el máximo provecho de cada una de las tecnologías, con una gestión mejorada de los servicios web, SOA, http, comunicaciones entre dispositivos especialmente móviles, utilizando todo el potencial de XML como lenguaje universal.

Puede correr sobre Microsoft Windows para 64 bits, beneficiándose así de una capacidad en memoria mayor.

Las nuevas características que Microsoft Biztalk 2006 frente a sus versiones anteriores son:

- Fácil instalación: Tiene varias formas de hacerlo incluyendo la forma de fácil instalación. Ahora consta dentro del mismo instalador todos los prerequisites para las distintas plataformas y adaptadores. La utilización del wizard para la configuración permite la administración de Biztalk en uno o varios servidores sin la necesidad de mayores cambios.
- Mejoras en el intercambio de mensajes y orquestación en tiempo de ejecución: Procesa los siguientes mensajes:
 - Mensajes de fallos y reporte de errores.
 - Resumen de mensajes.
 - Mejora el soporte para la entrega de mensajes en orden.
 - Transformación de mensajes largos.
 - Mejoras en el proceso de intercambio de mensajes.
 - Conductores parametrizados.

¹⁴ WIKIPEDIA, Glosario, <http://es.wikipedia.org/wiki/Framework>, Noviembre del 2006, [Framework.- Es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado, puede incluir soporte de programas, bibliotecas y un lenguaje scripting entre otros softwares para ayudar a desarrollar y unir los diferentes componentes de un proyecto.]

- Mejoramiento en las herramientas para desarrollo: El new Flat-File wizard (Asistente de creación de esquema de archivo planos) permite la simplificación de desarrollo de soluciones, brinda una interfase gráfica para la creación de esquemas en archivos planos. En la orquestación de aplicaciones tiene varias nuevas herramientas como la posibilidad de acercarse a detalles de esquemas de orquestación complejos y facilitar una gran ventana para programar directamente sobre C#¹⁵ desde Biztalk.
- Mejoras en el despliegue y la administración: Provee una organización de artefactos como aplicaciones, las mismas que pueden ser administradas individualmente y pueden ser exportadas como paquetes MSI. Presenta una nueva consola de administración, posee concentrador de grupos y el administrador por línea de comandos BTSTask.
- Nuevos adaptadores de integración y actualización
- Mejoras en el monitoreo de actividades del negocio: El Business Activity Monitoring (BAM) ofrece servicios como el portal BAM, alerta y sistema de notificaciones, servicios web, además de mejoras en la administración.
- Aplicaciones actualizadas para Biztalk Server 2004: Actualizaciones especialmente en la actualización de proyectos a Visual Studio 2005 y SQL Server 2005.

1.3.1.6 Adaptadores Integrados a BizTalk Server 2006

Posee nuevos adaptadores como:

- POP3 (Post Office Protocol): Adaptador habilitado para recibir mensajes de correo electrónico y adjuntos para monitoreo de la bandeja de entrada, puede ser habilitado para usarlo como recuperación mensajes de correo electrónico de cualquier sistema POP3 con el que esté configurado o al cual obedezca.
- Windows Shared Point Services: Adaptador que permite un acceso rápido a documentos almacenados en las librerías Shared Point, es similar al acceso a un servidor de archivos.

¹⁵ WIKIPEDIA, Glosario, http://es.wikipedia.org/wiki/C_Sharp, Noviembre del 2006, [C#.- Es un lenguaje de programación orientado a objetos desarrollado por Microsoft y estandarizado como parte de su plataforma .NET]

- MSMQ (Microsoft Message Queue): Adaptador que hace posible mayor seguridad en la integración de aplicaciones dispares, soporta la distribución ordenada de mensajes.
- MQSeries: Adaptador que hace posible seguridad y la integraciones de aplicaciones multiplataformas habilitando XML en las aplicaciones MQSeries existentes.
- SMTP (Simple Mail Protocol): Adaptador que provee la funcionalidad de transmitir mensajes de correo electrónico con un contenido en el cuerpo, puede ser habilitado para no solamente multipurpose Internet Mail Extensions (MIME) sino también componentes con texto plano y HTML, además de correos electrónicos que contengan archivos adjuntos.
- File: Este adaptador permite que BizTalk suba y elimine archivos desde una ubicación ya sea dentro del servidor de archivos o localmente, posee la habilidad de buscar una carpeta específica cuando la configuración de ambas partes permita enviar puertos y recibir ubicaciones, siempre y cuando tenga una manera de autenticarse con las credenciales para el acceso remoto a los archivos.
- HTTP (Hypertext Transfer Protocol): Este adaptador posee una nueva configuración que controla el estado con las solicitudes HTTP en caso de que el proceso falle.
- SOAP: Adaptador que en la versión 2006 puede ser usado para llamar a servicios web sin el uso de orquestaciones.

1.3.1.7 Beneficios de Integrar con Biztalk Server 2006.

Microsoft Biztalk 2006 es un integrador de agilidad conocida en el mundo entero, es así que más de 5000 empresas ya lo tienen dando un valor adicional a sus procesos de negocio al integrar sus aplicaciones. Los principales beneficios que presenta son:

- **Gestión completa:** El hecho de poseer nuevas herramientas de administración permite que los administradores efectúen su trabajo de una manera más rápida y eficiente.
- **Entorno de implantación de redes corporativas:** Los administradores de tecnologías de la información realizan tareas de gestión y operación de implantaciones de redes corporativas, con capacidades para modificar la configuración gracias a la nueva consola de administración, que incluye funciones de monitoreo y control de instalaciones.
- **Actualización directa desde Biztalk Server 2004:** La actualización desde la versión 2004 es directa, permitiendo mantener la configuración, arquitectura desde la versión anterior.
- **Experiencia del usuario final más placentera:** Las características incorporadas en BAM posee soporte en tiempo real para monitorizar las actividades del negocio y alertas, estas funcionalidades son personalizables en el portar BAM por basarse en ASP .NET 2.0.
- **Soporte a través de Windows Server System:** Ofrece soporte para varias herramientas además de Common Engineering Criteria que es un producto de Microsoft Server System.
- **Experiencia del desarrollador más satisfactoria:** Las herramientas de desarrollo mejoradas que soporta la versión 2006 de Microsoft Biztalk Server permiten una productividad nunca antes lograda al momento de construir y optimizar procesos de negocio, especialmente con la utilización del asistente de creación de esquema de archivo plano además de otros nuevos adaptadores conjuntamente con el nuevo framework 2.0 y Visual Studio 2005.
- **Integración de aplicaciones más rápida:** Aplicaciones ya conocidas en el mercado como Siebel, PeopleSoft entre otras, se integran rápidamente mediante los nuevos adaptadores para aplicaciones empresariales como las mencionadas.

1.3.2 HERRAMIENTAS DE DESARROLLO DE APLICACIONES

1.3.2.1 Descripción de la Herramienta J2EE

J2EE son las siglas de Java 2 Platform Enterprise Edition, es parte de la plataforma de programación Java que soporta aplicaciones creadas en ambientes con arquitecturas multicapas distribuidas con lenguaje de implementación Java. De cierta manera se puede decir que es el heredero de C++ por su manera de programar. Está diseñado de tal manera que simplifica el desarrollo pues el modelo de aplicación está basada en componentes de software modulares, los mismo que se ejecutan en servidores de aplicaciones, puede ser utilizado tanto en computadores de escritorio como en servidores, permitiendo mayor escalabilidad. A decir de Sun Microsystems que es la empresa creadora de Java, J2EE es la agrupación de estándares y especificaciones para el desarrollo de aplicaciones personales o empresariales sobre tecnología Java.

J2EE es una especificación acordada entre proveedores para crear aplicaciones de distinto nivel de complejidad ya que fue creada con la participación de cientos de empresas de diverso propósito lográndose una plataforma conjunta, por ser considerado informalmente como un estándar asegurando la portabilidad de las aplicaciones.

Facilita la integración de las aplicaciones de distintos clientes independientemente de los sistemas operativos o aplicaciones, por estar basado en un lenguaje neutral, robusto, estable, independiente de plataforma, hardware y sistemas operativos como JAVA su crecimiento entre los desarrolladores de aplicaciones crece de manera vertiginosa, alcanzando éxito en la implantación de grandes proyectos con resultados satisfactorios.

1.3.2.1.1 Ventajas de usar J2EE

- Soporta múltiples sistemas operativos, solamente requiere que se pueda ejecutar la máquina virtual.
- Permite crear soluciones basadas totalmente en software libre.
- Facilita la creación de software con mejor rendimiento, precio, entre otros, Mejorando la competitividad de los productos desarrollados sobre J2EE.

- Está controlado por el JCP¹⁶ garantizando la evolución de la plataforma.
- Soportado por más de 30 proveedores certificados en el mercado.
- Es una plataforma con mucho tiempo en el mercado por lo que posee soporte en todas las líneas.
- Soporta mejor los cambios de hardware.
- Posee una excelente portabilidad a otros servidores de aplicaciones.
- Amplia variedad de herramientas de desarrollo.

1.3.2.1.2 Desventajas de usar J2EE

- Depende exclusivamente del lenguaje de programación Java.
- Requiere de desarrolladores experimentados pese a ser una plataforma relativamente fácil.
- La gran cantidad de herramientas disponibles para su programación puede causar confusión o dependencia con una empresa entre los desarrolladores.
- Posee herramientas poco integradas entre proveedores.
- Las aplicaciones de otros lenguajes se consideran fuera de plataforma.

1.3.2.2 Descripción de la herramienta .NET.

Es una plataforma creada por Microsoft que se compone de un conjunto de aplicaciones como Visual Studio .Net, los servicios de .Net, entre otros, comprende servidores, clientes y servicios.

Pone énfasis en la transparencia de redes, permite un desarrollo acelerado de aplicaciones. Pretende integrar todos los productos Microsoft, desde el sistema operativo hasta las herramientas de mercado, esta plataforma es una propuesta para el creciente mercado de entornos web.

.NET ofrece una solución rápida y con costos bajos para el desarrollo de aplicaciones de software robustas y seguras, facilita la integración ágil entre

¹⁶ JAVAHISPANO, <http://www.javahispano.org/articles.article.action?id=70>, 2005, [JCP.- Organismo formado por más de 500 empresa como: SUN, IBM, Oracle, SAP, HP, AOL, entre otros. <http://www.JCP.org>]

empresas y un acceso simple y universal a todo tipo de información independientemente del dispositivo usado.

1.3.2.2.1 Ventajas de usar .NET

- Ideal para entornos Microsoft ya que posee los mejores Kits de herramientas.
- Permite manejar múltiples sistemas de lenguajes en una aplicación.
- Modelo simple de programación, permite desarrollar rápidamente pero con menos control.
- Alta integración con el Sistema Operativo Windows.
- Visual Studio .NET brinda de una variedad de herramientas potenciando un entorno homogéneo de desarrollo.
- Ampliamente usado gracias al eficiente departamento de marketing de Microsoft.
- Permite su utilización a programadores poco experimentados facilitándoles crear aplicaciones.

1.3.2.3 Desventajas de usar .NET

- No soporta múltiples sistemas operativos.
- La plataforma está dominada por su único dueño Microsoft, lo que significa que es la única empresa que lo puede mejorar o retirar del mercado, con poca estimulación a su evolución. Dependencia de un solo proveedor.
- Es una tecnología relativamente nueva y tiene pocos proyectos importantes a su haber.
- Permite poco desarrollo de aplicaciones en software libre.
- Al combinar varias herramientas puede dar lugar a que el código sea mantenible por ciertas personas.
- Monopolización de las herramientas de desarrollo.

1.3.3 SELECCIÓN DE LAS HERRAMIENTAS DE DESARROLLO

En las secciones anteriores se encuentran descritas brevemente las principales herramientas para implementar web services en BizTalk Server 2006. Para comprobar que en verdad la integración multiplataforma es posible se ha seleccionado las dos herramientas que se aplicarán con sus ventajas y desventajas en dos aplicativos distintos para este proyecto.

A continuación se presenta una tabla comparativa y valorativa de las dos herramientas a utilizarse, frente a la caracterización del problema realizada en un inicio, los valores usados son 1 – no cumple con esta característica, 5 – cumple parcialmente y 10 – cumple totalmente.

Tabla I - 3 Comparativa de herramientas de desarrollo de software.

<i>Características</i>	1.3.3.1.1	1.3.3.1.	1.3.3.1.
1.3.3.1.4 Soporta múltiples sistemas operativos	1.3.3.1.5	1.3.3.1.	1.3.3.1.
1.3.3.1.8 Permite trabajar con Web Services	1.3.3.1.9	1.3.3.1.	1.3.3.1.
1.3.3.1.12 Herramienta fácil de usar	1.3.3.1.13	1.3.3.1.	1.3.3.1.
1.3.3.1.16 Soporta múltiples lenguajes de programación	1.3.3.1.17	1.3.3.1.	1.3.3.1.
1.3.3.1.20 Soporte en cambios de hardware	1.3.3.1.21	1.3.3.1.	1.3.3.1.

1.3.3.1.24 Portabilidad a otros servidores de aplicaciones 1.3.3.1.25 1.3.3.1. 1.3.3.1.

1.3.3.1.28 Totales 1.3.3.1.29 1.3.3.1. 1.3.3.1.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

1.3.4 DESCRIPCIÓN DETALLADA DE LAS HERRAMIENTAS SELECCIONADAS.

1.3.4.1 J2EE

Java 2 Enterprise Edition es una plataforma creada para redes, intranets, Internet y afines, soportando una gran cantidad de usuarios conectados simultáneamente. Contiene bibliotecas desarrolladas específicamente para el acceso a servidores, sistemas de correo electrónico, bases de datos, entre otros.

J2EE no es un programa que se los puede descargar como el JDK que distribuye Sun Microsystems, ya que J2EE es una especificación, es concretamente un JRS¹⁷ que define una plataforma empresarial integrada de los siguientes componentes:

- Una agrupación de especificaciones

¹⁷ MOLPECERES Alberto, Arquitectura empresarial y software libre, <http://www.javahispano.org/articles.article.action?id=70>, Agosto del 2002, [JRS: Java Specification Request, Cuando una persona o entidad cree necesario la presencia de una determinada tecnología dentro de la plataforma basada en JAVA, crea un JRS y lo presenta para su aprobación. Dentro de este documento se relata porqué es necesaria la tecnología y cómo soluciona los problemas que se presentan con las tecnologías existentes. Cuando se aprueba se crea una especificación.]

- Una prueba de compatibilidad, J2EE Compatibility Test Suite (CTS)
- Una implementación de referencia de J2EE
- Un conjunto de guías de desarrollo y de prácticas aconsejables denominadas J2EE BluePrints.

Cada especificación está dada por varios JRS que lo sustentan, luego el test de compatibilidad está dado por más de 15000 pruebas en el caso de la versión de la especificación 1.4, los mismos que ayudan al fabricante a implementar la especificación y probar con todos los servidores de aplicaciones posibles la compatibilidad del producto, asegurando así un mínimo de calidad para evitar el surgimiento de aplicaciones que desvirtúen la plataforma.

Seguidamente de las pruebas se realizan las implementaciones de referencia de la nueva tecnología, que permite a los desarrolladores tener un producto con el cual empezar a desarrollar de manera compatible con la especificación.

Finalmente se encuentran los J2EE BluePrints que son un conjunto de documentos que se los puede descargar vía Web, allí se definen patrones de diseño y prácticas recomendadas para la creación de aplicaciones J2EE.

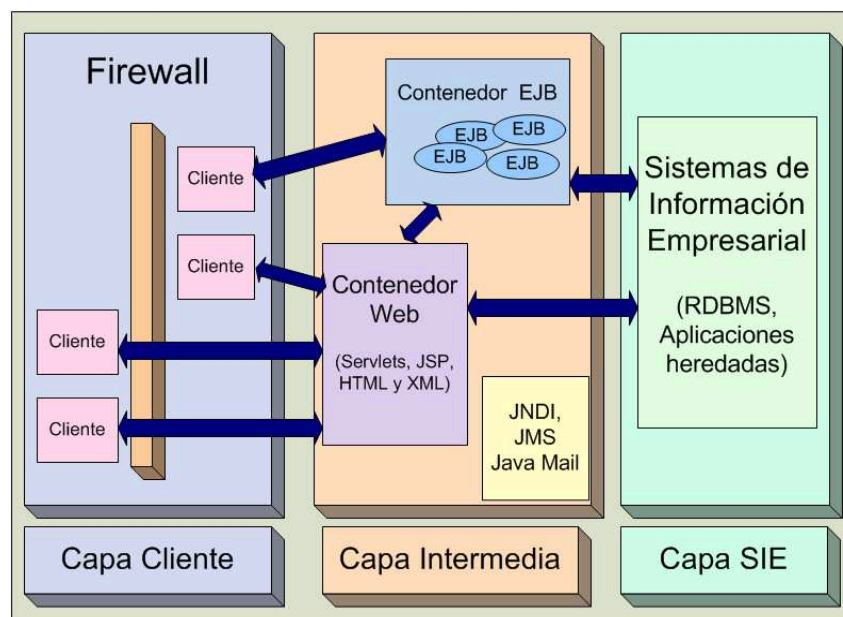
Es por esto que define un conjunto de lineamientos para desarrolladores de aplicaciones y otra para los proveedores de servicios de aplicaciones.

Esquemáticamente podemos decir que J2EE es el lenguaje de programación Java agregado componentes orientados a la lógica del negocio, es decir contiene una serie de especificaciones cada una con funcionalidades distintas y se encuentran a nuestra disposición como una gran caja de herramientas que las podemos utilizar para elaborar los aplicativos más variados, además de estar comprobados con todo el proceso descrito anteriormente.

La plataforma J2EE define un modelo de programación encaminado a la creación de aplicaciones basadas en n capas, generalmente se tienen cinco capas diferentes y se describen a continuación:

- Capa del cliente: Representa la interfaz del usuario, generalmente es gráfica y fácil de manejar por el cliente.
- Capa de presentación: Conjunto de componentes que generan la información que se presenta al cliente, generalmente se crean a través de componentes basados en Servlets y JSP.
- Capa de lógica del negocio: Contiene los componentes de negocio que hemos creado y que son reutilizables. Se forman generalmente a través de EJB.
- Capa de integración: Componentes que hacen transparente el acceso a la capa de sistemas de información.
- Capa de sistemas de información: Engloba el sistema de información, ya sea bases de datos relacionales, orientadas a objetos, entre otros.

Figura I - 6 Modelo Multicapa J2EE



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Todas estas capas con la arquitectura robusta que se describe por detrás deja notar las grandes ventajas mencionadas anteriormente que J2EE brinda a los desarrolladores.

No se puede dejar de destacar que J2EE es soportado por grandes sistemas empresariales y en su versión más básica por pequeños dispositivos.

Cabe destacar el nivel de seguridad que por estar implementado sobre Java nos brinda la plataforma, además de evitar las medidas más comunes de seguridad que tienen la mayoría de lenguajes y especificaciones como soportar comportamientos hostiles mediante hardware o software, protección contra pérdida de datos, mecanismos de autenticación, entre otros, tiene medidas de seguridad de otro nivel como la capacidad de minimizar el impacto de errores de programación, seguridad en la capacidad de manera fehaciente comunicarse, todo esto con la velocidad de desarrollo tanto de aplicaciones como servicios, elaborados con componentes reutilizables.

1.3.4.1.1 Servicios de Plataforma

Los principales servicios que la plataforma pone a disposición son:

- Servicio de Nombres: Mantiene un ambiente de nombre de tal manera que se pueden personalizar componentes.
- Servicios de Despliegue: Posibilita la configuración personalizada de componentes y aplicaciones en tiempo de empaquetado y despliegue.
- Servicios Transaccionales: Apoya a las transacciones distribuidas.
- Servicios de Seguridad: Permite el control de accesos mediante autenticación y autorización.

1.3.4.1.2 Tecnología de Servicios

Las principales tecnologías de servicios que ofrece J2EE son:

- Java Naming and Directory Interface (JNDI): Permite asociar atributos con objetos y búsqueda de los mismos.
- Java Transaction API (JTA) y Java Transaction Service (JTS): Soportan transacciones distribuidas.

- J2EE Connector Architecture (JCA): Facilita la conexión de la plataforma J2EE con SIE¹⁸.
- Java API para procesamiento XML (JAXP): Procedimiento para transformación de documentos XML.

1.3.4.1.3 Tecnologías de Comunicación

- Protocolos de Internet: Protocolos que permiten establecer la comunicación vía web como TCP/IP, http, SSL.
- RMI: Interfases para objetos remotos y serialización.
- Protocolos de OMG (Corba): Java IDL y RMI-IIOP
- Web Services.

1.3.4.1.4 Tecnologías de Mensajería

- Java Message Service API (JMS): Permite seguir con el proceso de tareas, además soporta la mensajería punto a punto (PTP) y de publicación – suscripción.
- Java Mail API: Presenta el API para el sistema de correo electrónico, soporta varias formas de almacenamiento, formatos y transporte.

1.3.4.2 .NET

.NET es una plataforma creada por Microsoft para el proceso de desarrollo de aplicaciones y servicios Web, la intención de lograr la transparencia en redes usando protocolos estándares, independencia de plataforma y desarrollo rápido de aplicaciones, entre las más destacables. La intención es que con la creación del framework de .NET se integren todos los productos de Microsoft desde el Sistema Operativo hasta las herramientas de mercado.

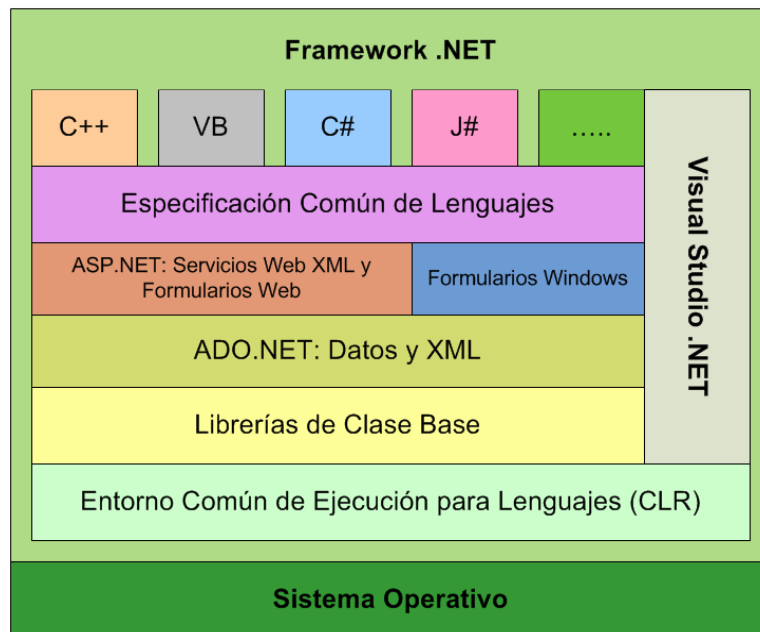
¹⁸ MONGE Raúl, Tecnología J2EE, Valparaíso, Julio 2005 [SIE: e.g. ERP, procesamiento transaccional en Mainframes, entre otros.]

El Framework cuyo significado en Español es marco de trabajo, es la base de la plataforma .NET, es la manera en cómo se comunican y logra su objetivo el conjunto de lenguajes de programación, herramientas y servicios de la plataforma, es aquí donde se reúnen las normativas impulsadas no solamente por Microsoft, garantizando así la funcionalidad en los lenguajes de desarrollo, independencia de Sistema Operativo aunque recién se realizan investigaciones en este aspecto.

1.3.4.2.1 Componentes del Framework de .NET.

Posee varios componentes que por su complejidad se los a agrupado para mostrar claramente la interacción entre ellos el momento del desarrollo y ejecución de las aplicaciones, se compone principalmente por un conjunto de lenguajes de programación, la biblioteca la clases base BCL y por el entorno común de ejecución para Lenguajes CLR.

Figura I - 7 Framework .NET



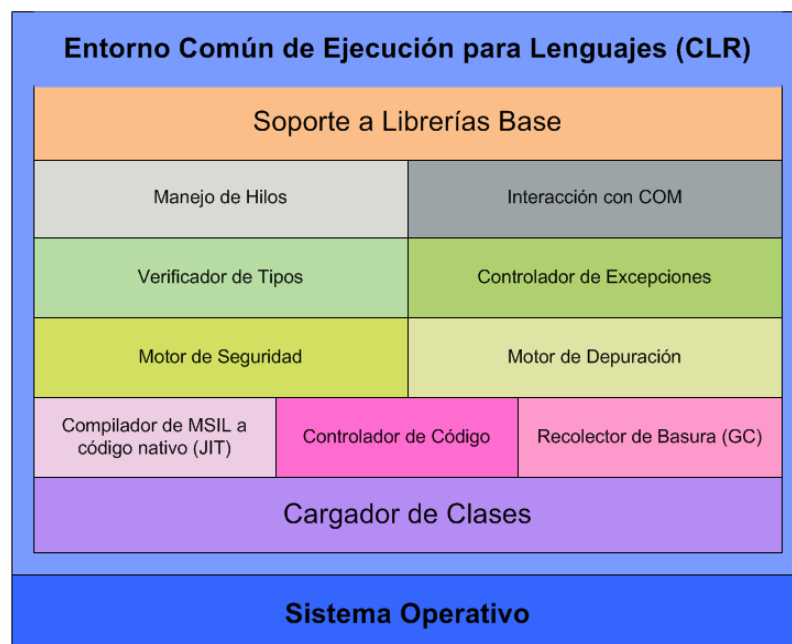
Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

- *Common Lenguaje Runtime (CLR)*: Es el núcleo en sí del Framework de .NET, ya que se encarga de la ejecución de las aplicaciones desarrolladas con esta plataforma, para esto aplica un conjunto de servicios del sistema

operativo. La ejecución del código se lo realiza en dos pasos. El CLR tiene un modelo de capas y directamente se encuentra sobre el sistema operativo, para su ejecución primeramente las herramientas de desarrollo compilan código fuente de cualquiera de los lenguajes de programación que soporta .NET, a éstos les transforma a código intermedio MSIL (Microsoft Intermediate Lenguaje), basándose en el CLS (Common Lenguaje Specification) que determina las reglas necesarias para crear el código MSIL. El segundo paso antes de la ejecución el compilador JIT (Just – In – Time) genera el código de máquina logrando así independencia de hardware.

Figura I - 8 Entorno Común de Ejecución para Lenguajes (CLR).



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

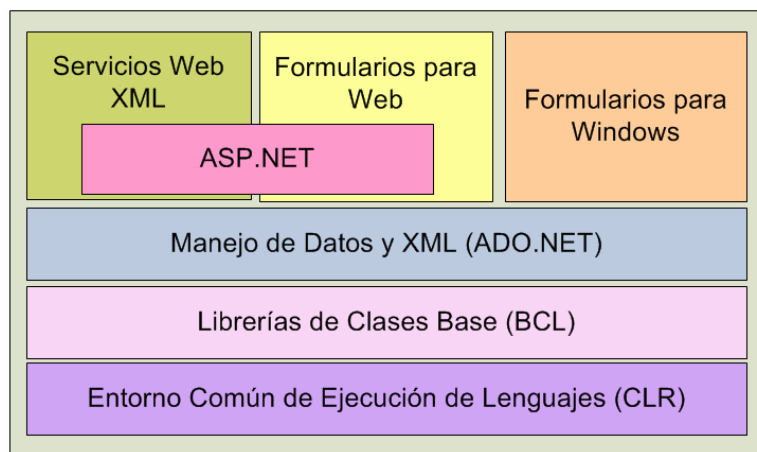
- Biblioteca de Clases Base de .NET: Aquí se maneja la mayoría de las operaciones básicas que se encuentran involucradas en el desarrollo de aplicaciones como la administración de la memoria, interacción con dispositivos, gestión de excepciones, operaciones aritméticas, cifrado de

datos, entre otras. La funcionalidad se maneja de manera ordenada por medio de espacios de nombre jerárquicos.

La biblioteca se clasifica en:

- ASP.NET y servicios Web XML
- Windows Forms
- ADO.NET

Figura I - 9 Biblioteca de Clases Base de .NET.



Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico*, Quito, 2007.

Elaboración: Hidalgo, Karina.

- *Ensamblados*: Son ficheros del tipo ejecutable (EXE) o DLL que contiene la funcionalidad de la aplicaciones de manera encapsulada evitando el registro de los componentes de la aplicación. Se debe tener en cuenta que los ensamblados son unidades lógicas y no físicas, puede consistir de uno o más módulos.

Un ensamblado contiene:

- *Manifiesto*: Contiene la descripción del ensamble.
- *Metadatos de Tipo*: Contiene la descripción de los tipos incluidos en el ensamble.
- *MSIL*: Contiene el código en Lenguaje Intermedio de Microsoft.
- *Recursos*: Audio, video, imágenes, tipos de letra u otro recurso usado por el ensamble.

CAPITULO 2

INGENIERÍA DE SOFTWARE APLICANDO LA ARQUITECTURA MULTIPLATAFORMA BIZTALK 2006.

En este capítulo se encuentra detalladamente todo el proceso de desarrollo de las aplicaciones a ser implementadas, la fase de análisis, diseño, implementación y pruebas del sistema, además se encuentra el diseño del proceso de integración de las aplicaciones.

Los resultados de la orquestación, alcances y restricciones se encuentran definidos en la especificación de requerimientos elaborada con estándares de calidad y diagramada usando la metodología RUP propuesta en el capítulo anterior.

2.1 INGENIERÍA DE REQUERIMIENTOS.

2.1.1 ESPECIFICACION DE REQUERIMIENTOS.

El proyecto se compone de dos sistemas independientes totalmente el uno de el otro pero que en determinado momento comparten una lógica del negocio en común, es así que en la especificación de requerimientos se hablará de un sistema A y un sistema B que finalmente al integrarlos permite la transferencia de información integrando los sistemas y permitiendo transacciones comerciales en beneficio de ambas empresas.

El sistema A es un sistema que realiza la facturación en un supermercado, es decir realiza el proceso de venta de productos mediante un cajero y posee su propio módulo de administración.

El sistema B es un sistema de Tarjetas de Crédito que lo que permite es procesar solicitudes de pago con Tarjeta ya sea con crédito corriente o diferido. También posee su propio módulo de administración.

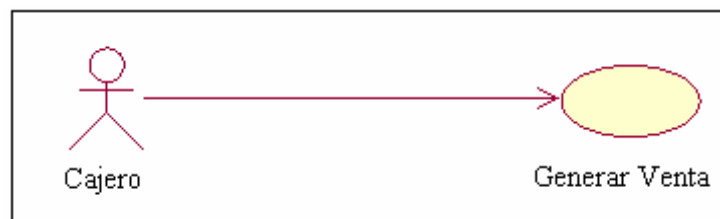
La integración de los dos sistemas permite que desde cualquier socio comercial, en este caso desde el supermercado se puedan cancelar las facturas de los clientes con tarjeta de crédito ya sea en corriente o diferido, facilitando el realizar compras tanto para los clientes como para las empresas que trabajan con esta modalidad.

Según los requerimientos funcionales que se describen detalladamente en el anexo titulado “Especificación de requerimientos”, se ha diseñado cada uno de los sistema independientes y luego la orquestación, dándonos como resultado los siguientes Diagramas de Casos de uso el Negocio.

2.1.1.1 Diagramas de Casos de uso del Negocio

2.1.1.1.1 Caso de uso del negocio del sistema de facturación en supermercados

Figura II - 1 Diagrama de caso de uso del negocio Generar Venta.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 1 Flujo de eventos del Caso de uso del Negocio Generar Ventas.

Nombre: Generar venta
Flujo de Eventos Principal
<ol style="list-style-type: none"> 1. El cajero ingresa un login, password y rol para el ingreso al sistema. 2. Selecciona la opción para realizar una venta 3. Factura los productos entregados por el cliente. 3. Solicita autorización del pago con tarjeta de crédito. 4. Ingresa el número de meses a diferir y los datos de la tarjeta. 5. Finaliza la factura y entrega los productos.

En 1, Si el login y password es incorrecto el sistema despliega un mensaje de error.

En 3, si no es autorizado el pago deshace la factura con la ayuda del administrador del sistema.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.1.1.2 Caso de uso del negocio del sistema de Tarjetas de crédito.

Figura II - 2 Diagrama del caso de uso del negocio Generar Transacción.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 2 Flujo de Eventos del Caso de uso de negocio Generar Transacción.

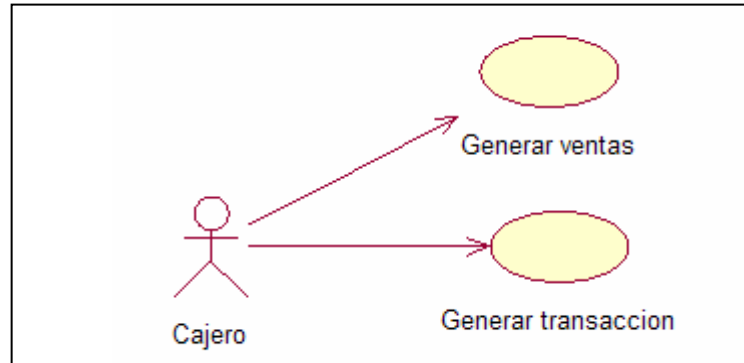
Nombre: Generar transacción
Flujo de Eventos Principal
1. El cajero ingresa datos de la tarjeta de crédito. 2. Solicita autorización de crédito por el valor de la factura. 3. Ingresa el número de meses a diferir. 4. Imprime el boucher.
En 1, en caso de que los datos sean incorrectos o que la tarjeta esté caducada despliega un mensaje de error. En 2, en caso de no autorizar el sistema el crédito se despliega un mensaje de error.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.1.1.3 Caso de uso del negocio del sistema Integrado.

Figura II - 3 Diagrama del caso de uso del negocio de los Sistemas integrados.



Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico*, Quito, 2007.

Elaboración: Hidalgo, Karina.

Los flujos de datos de los casos de uso del sistema integrado están descritos en los sistemas independientes por lo que no se describen en esta sección.

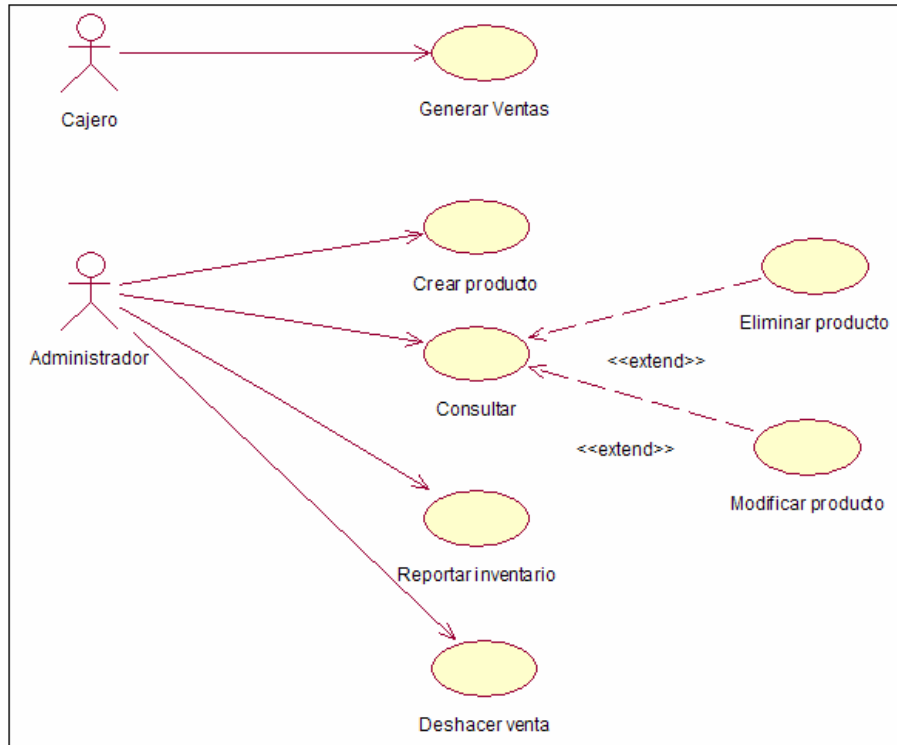
2.1.1.2 Diagramas de Casos de Uso de los Sistemas.

En los casos de uso del negocio se muestra de manera general la interacción de los usuarios finales con los sistemas y con la integración, es decir muestran el manejo usual de la lógica del negocio y las prácticas comunes de los usuarios.

Para detallar la funcionalidad de cada una de las aplicaciones se presenta a continuación los diagramas de casos de uso que se pueden observar con mayor detalle en el anexo titulado "Especificación de requerimientos" que se encuentra en el disco adjunto.

2.1.1.2.1 Caso de uso del sistema del sistema de facturación en supermercados.

Figura II - 4 Diagrama de casos de uso del Sistema de Facturación de Supermercados.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

A continuación se presenta el flujo de eventos de los casos de uso del Sistema de Facturación de Supermercados.

Tabla II - 3 Flujo de eventos del Caso de uso Generar Ventas.

Nombre: Generar ventas	
Descripción	El sistema debe ingresar los productos, a los precios de cada uno los debe sumar y calcular el total del valor a cancelar.
Entradas	El Cajero mediante el ingreso del identificador o código de barras de cada uno de los productos los ingresa, en caso de no poseer código de barras se ingresará una cadena de caracteres correspondiente al producto por medio del teclado.
Proceso	Compara la cadena de caracteres o el código de barras

	ingresado con la almacenada en la base de datos.
Salidas	Proyecta en el monitor y los imprime en la factura de venta a los productos con su respectivo precio.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 4 Flujo de eventos del Caso de uso Crear Producto.

Nombre: Crear Producto	
Descripción	El sistema permite crear nuevos productos en la base de datos.
Entradas	El administrador del sistema realiza el ingreso o creación de nuevos productos en la base de datos del supermercado.
Proceso	Ingresa todos los datos solicitados por la base de datos del nuevo producto, incluyendo el nombre, precio, stock, entre otros.
Salidas	Despliega un mensaje en el sistema indicando que el producto ha sido creado satisfactoriamente.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 5 Flujo de eventos del Caso de uso Consultar.

Nombre: Consultar	
Descripción	El sistema permite búsquedas de datos de productos previamente ingresados en la base de datos del sistema.
Entradas	El administrador del sistema coloca el o los criterios de búsqueda que considere necesarios para hallar el producto requerido.
Proceso	Se ingresa el o los términos que se desean buscar en la base de datos en la interfase del sistema y se presiona la opción buscar.

Salidas	Despliega los datos del producto en caso de encontrarlo, caso contrario se despliega un mensaje indicando que el producto no ha sido encontrado.
----------------	--

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 6 Flujo de eventos del Caso de uso Eliminar Producto.

Nombre: Eliminar producto	
Descripción	El sistema permite eliminar productos previamente creados en la base de datos.
Entradas	El administrador del sistema elimina todos los datos del producto existente en la base de datos del supermercado.
Proceso	Luego de realizar la búsqueda de todos los datos del producto que previamente debe estar creado en la base de datos, el administrador procede a la eliminación del producto con todos los datos que le corresponden de la base de datos.
Salidas	Despliega un mensaje en el sistema indicando que el producto ha sido eliminado satisfactoriamente.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 7 Flujo de eventos del Caso de uso Modificar Producto

Nombre: Modificar producto	
Descripción	El sistema permite realizar cambios en los datos de un producto previamente creado en la base de datos.
Entradas	El administrador del sistema modifica uno o varios datos del producto existente en la base de datos del supermercado.
Proceso	Luego de realizar la búsqueda de todos los datos del producto que previamente debe estar creado en la base de datos, el administrador procede a la modificación de uno o varios datos del producto en la base de datos.

Salidas	Despliega un mensaje en el sistema indicando que el producto ha sido modificado satisfactoriamente.
----------------	---

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 8 Flujo de eventos del Caso de uso Reportar Inventario.

Nombre: Reportar inventario	
Descripción	El sistema permite realizar reportes del inventario actualizado de productos existentes en el supermercado.
Entradas	El administrador del sistema solicita la realización del inventario actual de los productos existentes en el supermercado.
Proceso	Luego de realizar la solicitud al sistema de la elaboración del inventario actual de existencia de productos en el supermercado, el sistema realiza la búsqueda de todos los productos con su respectivo stock y elabora el reporte.
Salidas	Despliega el reporte solicitado con los datos que contiene la base en momento de la búsqueda.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 9 Flujo de eventos del Caso de uso Deshacer Venta.

Nombre: Deshacer venta	
Descripción	El sistema permite realizar devoluciones de productos, siempre y cuando el boucher haya sido el último en emitirse por el POS.
Entradas	El administrador del sistema solicita al POS deshacer el último boucher realizado.
Proceso	El cajero emite una factura y un boucher por el pago con tarjeta de crédito, para realizar una devolución el administrador del sistema solicita al sistema de autorizaciones

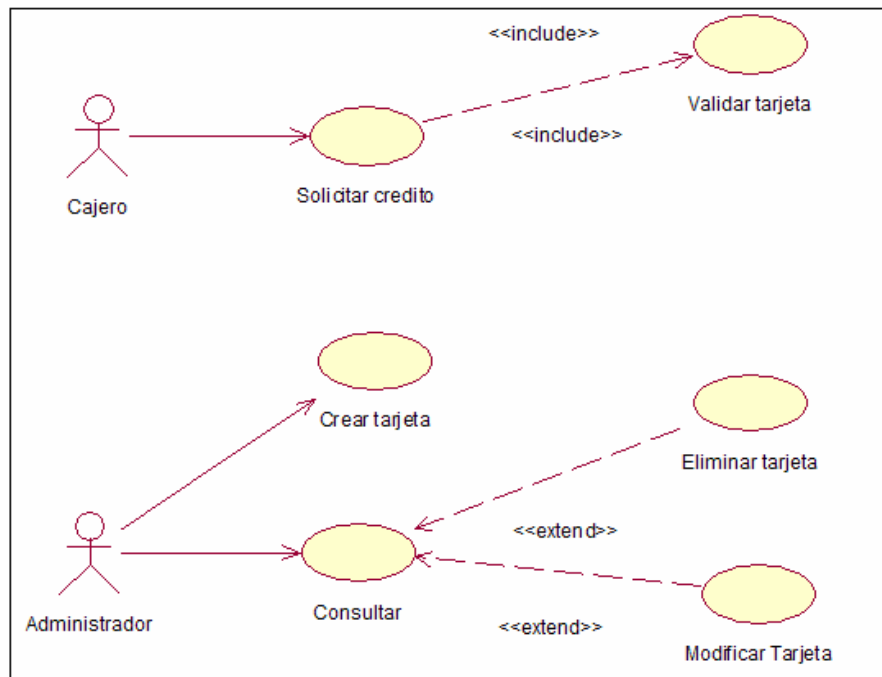
	de pago con tarjeta anular el boucher, esto se puede realizar siempre y cuando el boucher haya sido el último en emitirse por el POS.
Salidas	Despliega un mensaje en el sistema indicando que el boucher ha sido anulado mediante un código de autorización para deshacer el mismo. Caso contrario despliega un mensaje indicando la imposibilidad de realizar la anulación.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.1.2.2 Caso de uso del sistema del sistema de Pago con tarjeta de Crédito.

Figura II - 5 Diagrama de casos de uso del Sistema de Tarjeta de Crédito.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

A continuación se presenta el flujo de eventos de los casos de uso del Sistema de Tarjetas de Crédito.

Tabla II - 10 Flujo de eventos del Caso de uso Solicitar Crédito

Nombre: Solicitar crédito	
Descripción	Una vez validada la tarjeta en el sistema, se revisa el monto disponible que tiene la tarjeta y según el valor de la factura se autoriza o niega el pago.
Entradas	El valor de la factura a ser cancelada y el número de meses si desea diferir el pago.
Proceso	El sistema verifica el monto disponible que tiene la tarjeta de crédito, lo compara con el valor de la factura a cancelar, según el resultado de la comparación autoriza o niega el pago. Si desea diferir el pago se deberá ingresar este dato.
Salidas	En caso de que se autorice el pago se emitirá el boucher para ser firmado por el cliente, en caso de ser el cupo menor que el valor de la factura se emitirá un mensaje indicándolo.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 11 Flujo de eventos del Caso de uso Validar Tarjeta.

Nombre: Validar tarjeta	
Descripción	El sistema debe verificar la identidad y validez de los datos ingresados por el cajero de la tarjeta de crédito entregada por el cliente.
Entradas	El cajero ingresa los datos solicitados por el sistema de la tarjeta de crédito.
Proceso	Luego de conectarse el cajero con el sistema de pago con tarjeta de crédito, el cajero ingresa los datos de la tarjeta del cliente, el sistema valida los datos de la tarjeta de crédito.
Salidas	Despliega un mensaje en caso de ser aceptada o rechazada la tarjeta de crédito.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 12 Flujo de eventos del Caso de uso Crear Tarjeta.

Nombre: Crear Tarjeta	
Descripción	El sistema permite crear nuevas tarjetas de crédito en la base de datos.
Entradas	El administrador del sistema realiza el ingreso o creación de nuevas tarjetas en la base de datos del sistema.
Proceso	Ingresa todos los datos solicitados por la base de datos de la nueva tarjeta, incluyendo el nombre del tarjeta habiente, el cupo designado, fecha de caducidad, entre otros.
Salidas	Despliega un mensaje en el sistema indicando que la tarjeta ha sido creado satisfactoriamente.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 13 Flujo de eventos del Caso de uso Consultar.

Nombre: Consultar	
Descripción	El sistema permite consultas de información de tarjeta habientes previamente ingresados en la base de datos del sistema.
Entradas	El administrador del sistema coloca el o los criterios de búsqueda que considere necesarios para hallar la tarjeta de crédito requerida.
Proceso	Se ingresa el o los términos que se desean buscar en la base de datos en la interfase del sistema y se presiona la opción consultar.
Salidas	Despliega los datos de la tarjeta de crédito en caso de encontrarlo, caso contrario se despliega un mensaje indicando que la tarjeta no ha sido encontrada.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 14 Flujo de eventos del Caso de uso Eliminar Tarjeta.

Nombre: Eliminar Tarjeta	
Descripción	El sistema permite eliminar tarjetas de crédito previamente creadas en la base de datos.
Entradas	El administrador del sistema elimina todos los datos de la tarjeta de crédito existentes en la base de datos del sistema de tarjetas de crédito.
Proceso	Luego de realizar la búsqueda de los datos de la tarjeta que previamente deben estar creado en la base de datos, el administrador procede a la eliminación de la tarjeta de crédito con todos los datos que le corresponden de la base de datos.
Salidas	Despliega un mensaje en el sistema indicando que la tarjeta ha sido eliminada satisfactoriamente.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 15 Flujo de eventos del Caso de uso Modificar Tarjeta.

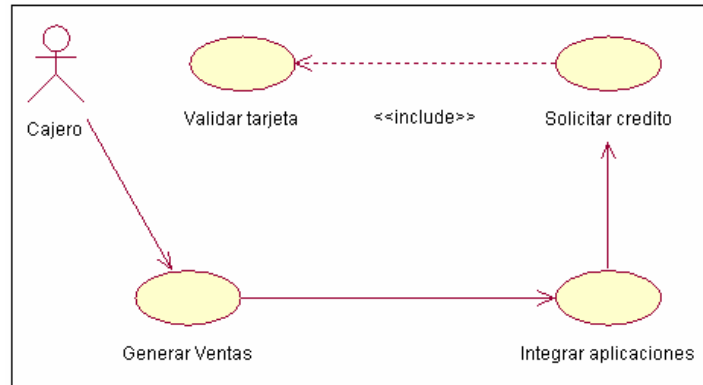
Nombre: Modificar Tarjeta	
Descripción	El sistema permite realizar cambios en los datos de una tarjeta de crédito previamente creada en la base de datos.
Entradas	El administrador del sistema modifica uno o varios datos de la tarjeta de crédito existente en la base de datos del sistema.
Proceso	Luego de realizar la consulta de los datos de la tarjeta que previamente debe estar creada en la base de datos, el administrador procede a la modificación de uno o varios datos de la misma en la base de datos.
Salidas	Despliega un mensaje en el sistema indicando que la tarjeta ha sido modificada satisfactoriamente.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.1.2.3 Casos de uso del Sistema Integrado

Figura II - 6 Diagrama de casos de uso del Sistema Integrado.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Los flujos de datos correspondientes a los casos de uso del sistema integrado se encuentran descritos anteriormente en cada uno de los sistemas individualmente. Los requerimientos funcionales del sistema integrado se especifican para cada uno de los diagramas de casos de uso, en tablas que contienen información adicional que permite un mejor entendimiento de la funcionalidad del negocio y del sistema.

Además de los diagramas de casos de uso del negocio y de casos de uso, el análisis de requerimientos consta también de los diagramas de secuencia y de actividad, que detallan con precisión los procesos que se llevarán a cabo dentro de cada sistema como con el integrador, éstos están ubicados dentro de el anexo que titula “Diagramas de Secuencia y Actividad” en el CD adjunto.

2.1.1.3 Planificación de los Sitios Web de los sistemas

2.1.1.3.1 Propósito

Los sitios Web de cada uno de los sistemas a implementarse y que posteriormente serán integrados tienen como propósito el presentar al usuario

final una interfase organizada, amigable y fácil de manejar par que pueda hacer uso tanto como usuario normal o como administrador de los sistemas, este último con todas las implicaciones de seguridad que conocemos.

El sitio Web en su interior almacenará los componentes programados para cada sistema que le permitirá a través de Internet el ejecutar los servicios y acciones requeridas.

2.1.1.3.2 Usuarios de los sitios Web

Los sitios están orientados para los clientes finales de los sistemas y para los usuarios que tengan el rol de administrador.

2.1.1.3.3 Almacenamiento de la información en el sitio Web.

La información por tener un sistema totalmente distribuido se almacenará en las respectivas bases de datos que los sistemas poseen y que se detallarán en páginas posteriores.

2.1.2 MODELO DE ANÁLISIS

Para la elaboración del modelo de análisis de los sistemas y su integración, se tomarán en cuenta los casos de uso de la especificación de requerimientos del sistema principales, ya que no es necesario realizar todos los casos de usos existentes, sino únicamente aquellos que tienen relación directa con la aplicación de integración y el modelo del negocio. La realización de todos los casos de uso se encuentran en el anexo titulado “Diagramas de Colaboración de Análisis” ubicado en el CD adjunto.

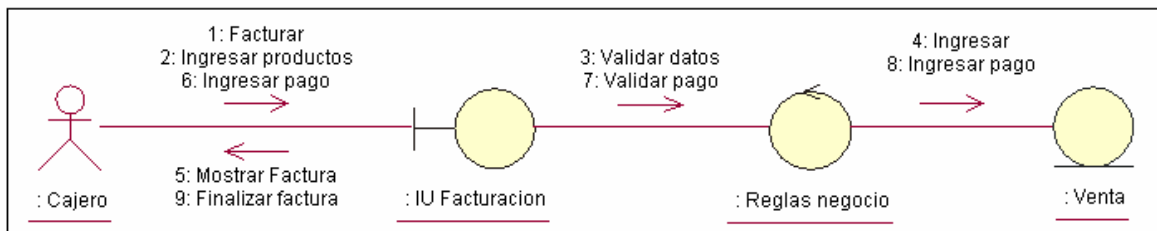
En este punto del proceso de desarrollo se presentan los sistemas y su integración, se analizan cada uno de ellos para determinar los flujos de trabajo fundamentales para un posterior análisis con los modelos de integración.

A continuación se presentan los diagramas de colaboración de análisis y los flujos de sucesos que corresponden a cada uno de los casos de uso analizados en pasos anteriores para cada uno de los sistemas y para la integración.

2.1.2.1 Realización de los casos de uso del Sistema de Facturación de Supermercados.

2.1.2.1.1 Realización del caso de uso Generar Ventas

Figura II - 7 Diagrama de colaboración de una realización del caso de uso Generar Venta



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 16 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Generar Venta

Flujo de sucesos – Diagrama de colaboración de la realización del caso de uso: Generar Venta

El cajero solicita iniciar con una factura nueva al objeto IU Facturación (1), el Administrador inicia con el ingreso de los productos uno a uno al sistema mediante el objeto IU facturación (2), este objeto solicita al objeto Reglas Negocio la validación de cada uno de los productos que están siendo ingresados en la factura (3) obteniendo así el nombre y precio del producto, el objeto Reglas Negocio solicita al objeto Procesos Sistema Facturación el ingreso de los productos verificados en la factura (4), los suma y desglosa la factura para el cajero, el sistema muestra el valor total de la factura a cancelar (5), El Cajero solicita al objeto IU Facturación la realización del pago, éste solicita al objeto Reglas Negocio la verificación de los valores ingresados para el pago (7) y el objeto Reglas Negocio solicita al objeto Procesos Sistema Facturación el ingreso

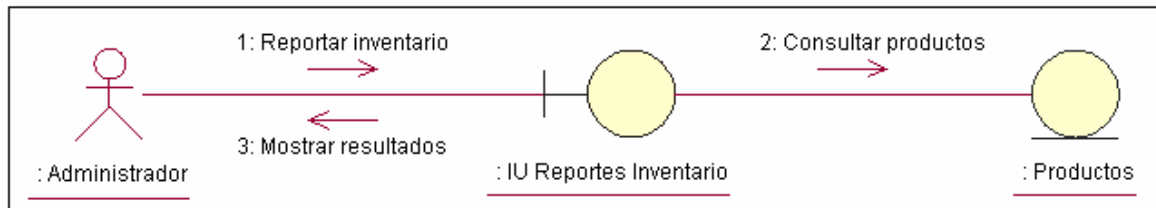
del pago de la factura (8), si el proceso de pago se realiza satisfactoriamente el sistema realiza la finalización de la factura (9).

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.2.1.2 Realización del caso de uso Reportar Inventario

Figura II - 8 Diagrama de colaboración de una realización del caso de uso Reportar Inventario



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 17 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Reportar Inventario

Flujo de sucesos – Diagrama de colaboración de la realización del caso de uso: Reportar Inventario

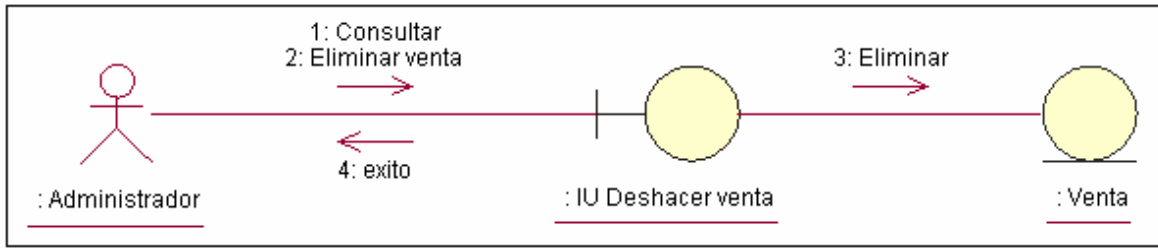
El Administrador solicita al objeto IU Reportes Inventario el realizar un reporte actualizado del stock de productos existentes en la base de datos del sistema (1), el objeto IU Reportes Inventario solicita al objeto Procesos Sistema Facturación realizar la consulta en la base de datos de la existencia de productos (2) y el objeto IU Reportes Inventario muestra los resultados de la búsqueda al Administrador (3).

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.2.1.3 Realización del caso de uso *Deshacer Venta*.

Figura II - 9 Diagrama de colaboración de una realización del caso de uso *Deshacer venta*



Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico*, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 18 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: *Deshacer venta*

Flujo de sucesos – Diagrama de colaboración de la realización del caso de uso: <i>Deshacer venta</i>
El Administrador solicita al objeto IU Deshacer venta desplegar la última factura realizada (1) por el sistema, luego solicita la eliminación de esa venta (2), el objeto IU Deshacer venta solicita al objeto Procesos Sistema Facturación la eliminación de la factura seleccionada (3), si este proceso ha sido realizado satisfactoriamente el sistema emite un mensaje de éxito al Administrador (4).

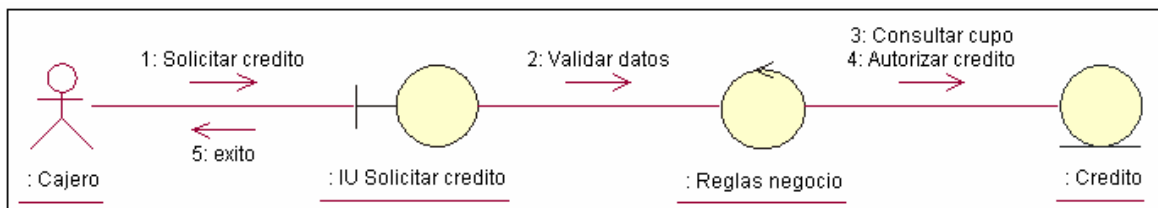
Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico*, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.2.2 Realización de los casos de uso del Sistema de Tarjeta de Crédito.

2.1.2.2.1 Realización del caso de uso *Solicitar Crédito*

Figura II - 10 Diagrama de colaboración de una realización del caso de uso *Solicitar credito*



Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico*, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 19 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Solicitar crédito

Flujo de sucesos – Diagrama de colaboración de la realización del caso de uso: Solicitar credito

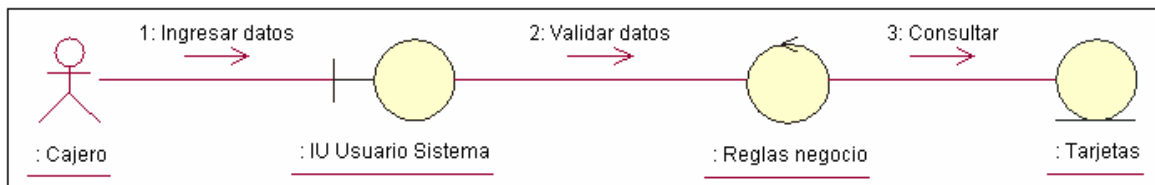
El Cajero solicita al objeto IU Solicitar crédito el pago con tarjeta de crédito (1), el objeto IU Solicitar credito envía los datos ingresados por el cajero al objeto Reglas Negocio para que sean verificados (2), el objeto Reglas Negocio solicita el consultar cupo disponible al objeto Procesos Sistema Tarjeta Credito para la tarjeta ingresada (3) y en caso de que el cupo sea suficiente para el valor del pago solicitado solicita la autorización de pago (4), si el proceso ha finalizado satisfactoriamente el sistema emite un mensaje de éxito al cajero (5).

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.2.2 Realización del caso de uso Validar Tarjeta

Figura II - 11 Diagrama de colaboración de una realización del caso de uso Validar Tarjeta



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 20 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Validar tarjeta

Flujo de sucesos – Diagrama de colaboración de la realización del caso de uso: Validar tarjeta

El cajero solicita al objeto IU Usuario Sistema el ingreso de los datos de la tarjeta de crédito al sistema para verificación de existencia y autenticidad (1), el objeto IU Usuario Sistema solicita al objeto Reglas Negocio la validación de los datos ingresados por el cajero (2) con lo existentes en la base de datos del sistema, el objeto Reglas Negocio realiza la consulta de datos al objeto Procesos Sistema

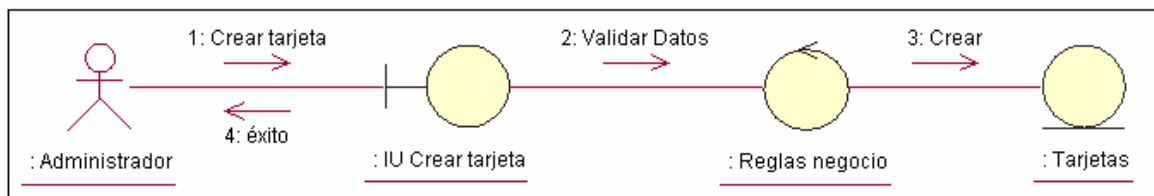
Tarjeta Credito (3) para dar a la tarjeta como válida o no.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.2.2.3 Realización del caso de uso Crear Tarjeta

Figura II - 12 Diagrama de colaboración de una realización del caso de uso Crear tarjeta.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 21 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Crear tarjeta.

Flujo de sucesos – Diagrama de colaboración de la realización del caso de uso: Crear tarjeta

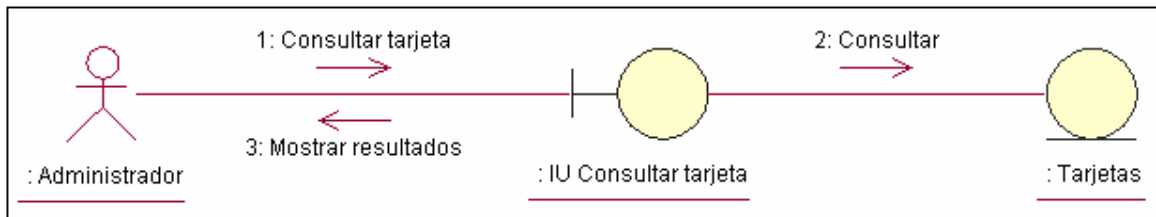
El usuario Administrador interactúa con el objeto IU Crear tarjeta para Ingresar los datos correspondientes a una nueva tarjeta (1) el objeto IU Crear tarjeta solicita al objeto Reglas Negocio validar los datos ingresados para la nueva tarjeta (2), el objeto Reglas negocio envía los datos para la creación de la tarjeta al objeto Procesos Sistema Tarjeta Crédito (3), en caso de que el proceso resulte satisfactorio el sistema envía un mensaje indicando el de éxito del proceso al Administrador (4).

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.2.2.4 Realización del caso de uso Consultar Tarjeta

Figura II - 13 Diagrama de colaboración de una realización del caso de uso Consultar.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 22 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Consultar.

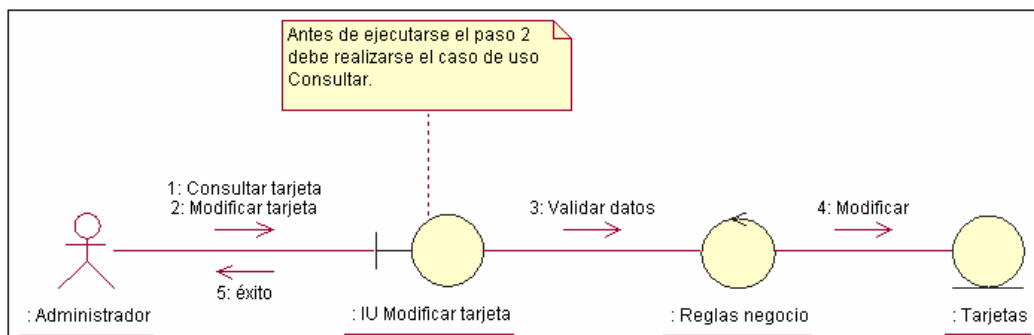
Flujo de sucesos – Diagrama de colaboración de la realización del caso de uso: Consultar
El usuario Administrador interactúa con el objeto IU Consultar tarjeta para consultar la información de una tarjeta de crédito en particular (1), El objeto IU Consultar tarjeta pide la información solicitada al objeto Procesos Sistema Tarjeta Crédito (2) y despliega los resultados al Administrador (3) en caso de haberlos.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.2.2.5 Realización del caso de uso Modificar Tarjeta

Figura II - 14 Diagrama de colaboración de una realización del caso de uso Modificar tarjeta.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 23 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Modificar tarjeta.

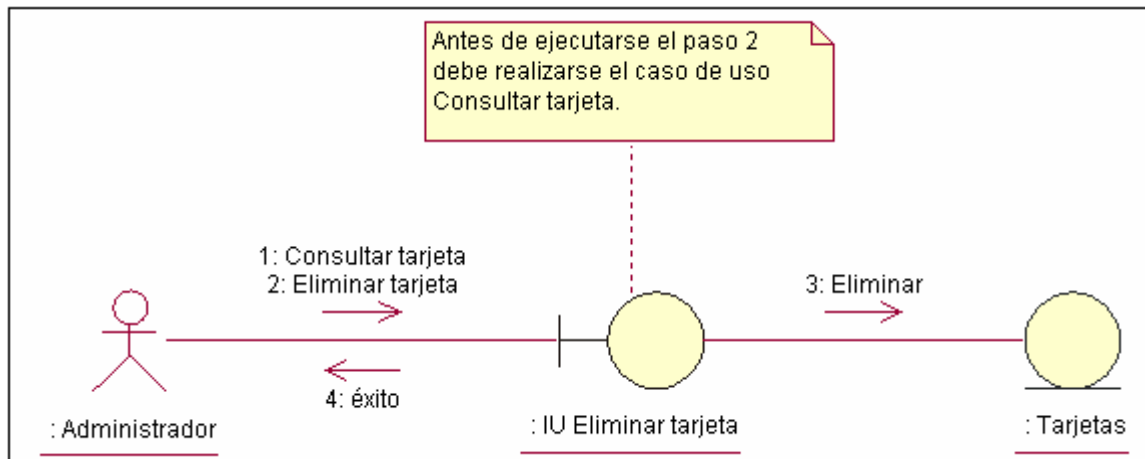
Flujo de sucesos – Diagrama de colaboración de la realización del caso de uso: Modificar tarjeta
El usuario Administrador solicita al objeto IU Modificar tarjeta la modificación de la tarjeta seleccionada previamente en el caso de uso Consultar tarjeta (1). A continuación el objeto IU Modificar tarjeta pide al objeto Reglas negocio la validación de los nuevos datos ingresados para la tarjeta (3). El Objeto Reglas Negocio solicita al objeto Procesos Sistema Tarjeta Crédito que cambie la información de la tarjeta seleccionada (4). En caso de que sea realizada la modificación satisfactoriamente el sistema emite un mensaje de éxito al Administrador (5).

Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico*, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.2.2.6 Realización del caso de uso Eliminar Tarjeta

Figura II - 15 Diagrama de colaboración de una realización del caso de uso Eliminar tarjeta.



Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico*, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 24 Flujo de sucesos del diagrama de colaboración de una realización del caso de uso: Eliminar tarjeta.

Flujo de sucesos – Diagrama de colaboración de la realización del caso de uso: Eliminar tarjeta

El usuario Administrador solicita al objeto IU Eliminar tarjeta la eliminación de la tarjeta seleccionada previamente en el caso de uso Consultar tarjeta (1). A continuación el objeto IU Eliminar tarjeta pide al objeto Procesos Sistema Tarjeta Crédito que elimine la información de la tarjeta seleccionada (3), si el proceso es realizado satisfactoriamente el sistema emite un mensaje de éxito al Administrador (4).

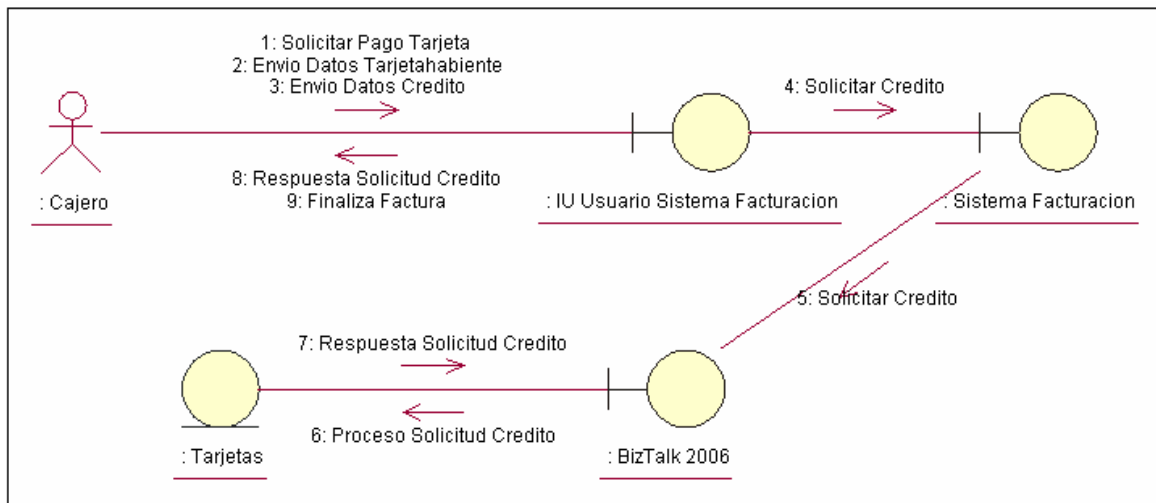
Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.2.3 Realización de los casos de uso del Sistema Integrado.

2.1.2.3.1 Realización del caso de uso Integración

Figura II - 37 Diagrama de colaboración de una realización del caso de uso de Integración.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.3 PAQUETES DE ANÁLISIS

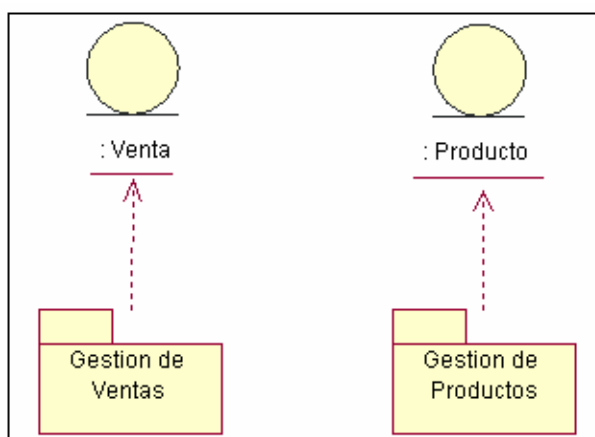
Durante el proceso de desarrollo de los sistemas y su integración se ha manejado una cantidad considerable de elementos como clases, componentes y diagramas, creando la necesidad de agrupar en bloques que contengan paquetes en su interior y que puedan ser manipulados en grupos, estos bloques son llamados paquetes.

En las siguientes figuras se muestra los paquetes de análisis obtenidos luego de un detenido análisis de las relaciones entre los casos de uso de cada uno de los sistemas e integración encontrados desde la fase de análisis de requerimientos y con las clases principales del dominio del problema.

2.1.3.1 Paquetes de análisis del Sistema de Facturación.

A continuación se presentan los paquetes correspondientes al sistema de Facturación de Supermercados.

Figura II - 16 Identificación de paquetes de análisis a partir de clases importantes del dominio del problema del sistema de Facturación.

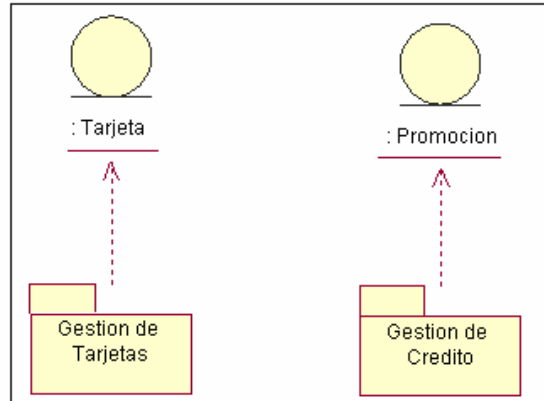


Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.3.1 Paquetes de análisis del Sistema de Tarjeta de Crédito.

Figura II - 17 Identificación de paquetes de análisis a partir de clases importantes del dominio del problema del sistema de Tarjeta de Crédito.

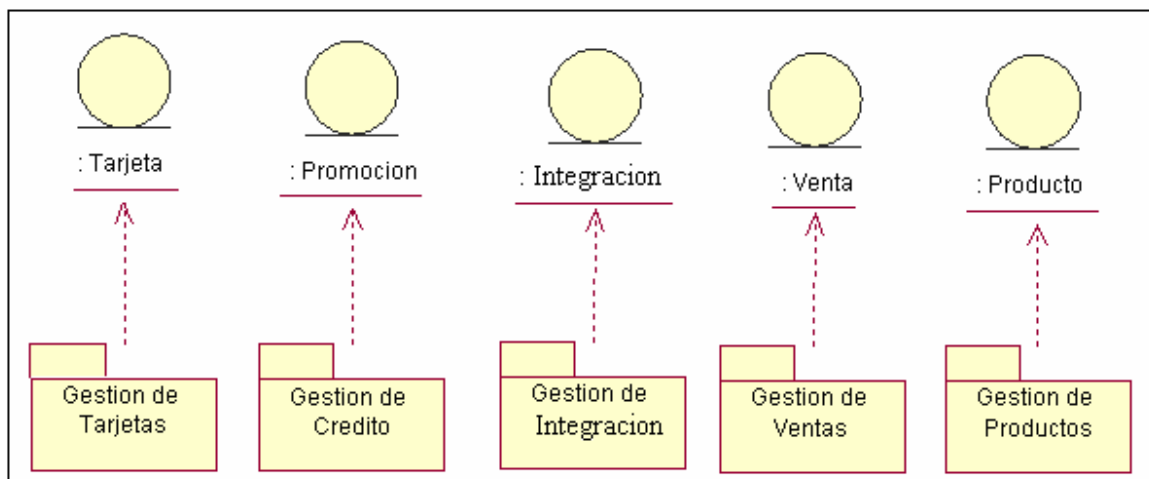


Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.1.3.1 Paquetes de análisis del Sistema Integrado.

Figura II - 18 Identificación de paquetes de análisis a partir de clases importantes del dominio del problema del sistema de integración.

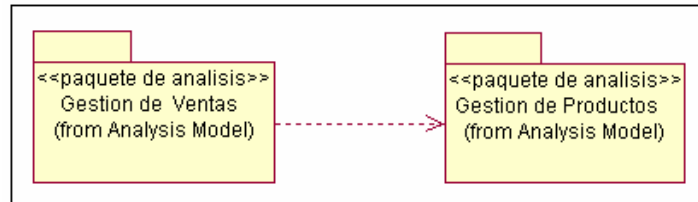


Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Además de los paquetes que se muestran en la figura anterior se ha determinado las relaciones de dependencia entre los paquetes que se presentan en la figura a continuación.

Figura II - 19 Diagrama de dependencia entre paquetes del sistema de Facturación.



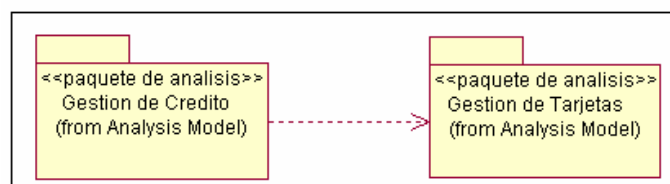
Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

El paquete Gestión de Ventas contiene las clases y elementos necesarios para administrar las ventas que pueden realizar mediante el sistema de Facturación, estas ventas son realizadas por el cajero mediante un Terminal del sistema, la parte netamente administrativa la realiza el Administrador del sistema desde cualquier Terminal con su respectiva clave de acceso.

El paquete Gestión de Productos contiene las clases y elementos que se requieren para administrar los productos del supermercado, es decir, crear, consultar, modificar, eliminar y reportar el inventario. Este paquete lo maneja el administrador con su clave desde cualquier Terminal activo.

Figura II - 20 Diagrama de dependencia entre paquetes del sistema de Tarjeta de Crédito.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

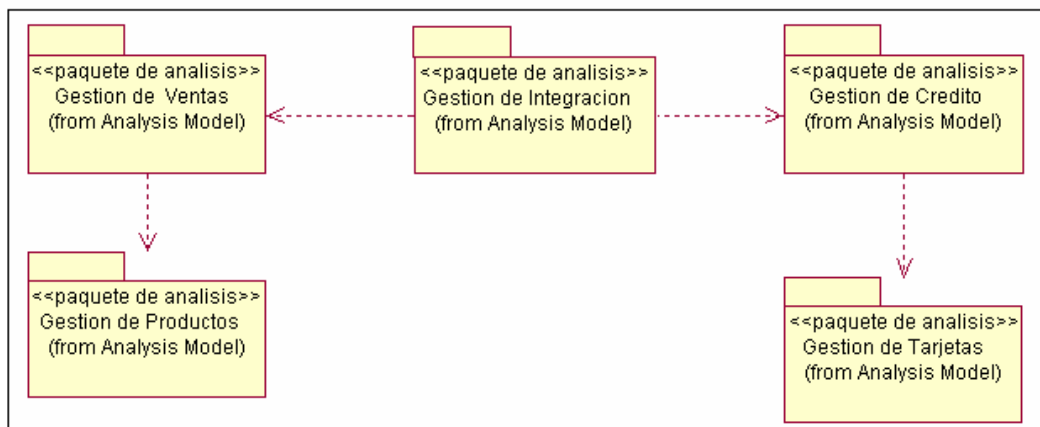
El paquete Gestión de Tarjetas permite administrar las tarjetas en el sistema, esto lo realiza el administrador del sistema con su clave de acceso desde cualquier

terminar activo. Este paquete contiene las clases y elementos necesarios para crear, consultar, modificar y eliminar tarjetas de crédito del sistema.

El paquete Gestión de créditos contiene las clases y elementos necesarios para activar y administrar el servicio web que permite manejar este tipo de transacciones desde terminales activos.

El paquete de Gestión de Promociones le permite al administrador realizar la administración de promociones mediante un Terminal activo. Este paquete contiene las clases y elementos necesarios para realizar este proceso.

Figura II - 21 Diagrama de dependencia entre paquetes del Sistema Integrado.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Los paquetes del sistema integrado realizan lo mismo que cuando los sistemas se encuentran aislados, simplemente el momento de integrarles se pueden comunicar entre ellos mediante el orquestador.

2.1.4 ANÁLISIS DEL SITIO WEB

El análisis del sitio web se lo debe hacer por separado para cada uno de los sistemas a programarse ya que son de propósito diferente pese a que la lógica del negocio en determinada etapa sea la misma.

2.1.4.1 Análisis del sitio Web para el sistema de facturación de supermercados.

Este sitio contiene tanto el módulo de facturación de los productos comercializados en el supermercado como el módulo administrativo del sistema. Los datos con los que trabaja el sistema se encontrarán almacenados en una base de datos y será recuperada por medio de las interfases del sistema que se conectan directamente. La información debe cumplir con las características necesarias para contribuir con el correcto funcionamiento del sistema.

La información de las interfases del sistema deberá tener como característica básica la exactitud en los datos, ya que esto es parte de la fidelidad del sistema. Las tareas administrativas se llevarán a cabo de manera ágil, brindando un manejo fácil y óptimo del sistema permitiendo un mantenimiento cómodo de la información. Para una administración certera se el sitio de administración contará con las siguientes operaciones básicas.

- Crear. Permite la creación de productos con sus respectivos datos en el sistema.
- Consultar. Permite realizar búsquedas de determinada información del sistema utilizando criterios de búsqueda varios.
- Modificar. Permite actualizar la información de productos previamente ingresados al sistema.
- Eliminar. Permite borrar información de productos previamente creados del sistema.

Con estas operaciones básicas el administrador puede garantizar un mantenimiento rápido de la información del sistema, todo esto mediante la utilización de las interfases amigables del sistema.

2.1.4.2 Análisis del sitio Web para el sistema de tarjeta de crédito.

Este sitio contiene tanto el módulo de administración de tarjetas de crédito como los servicios activos que posee el sistema. Los datos con los que trabaja el sistema se encontrarán almacenados en una base de datos y será recuperada por medio de las interfases del sistema que se conectan directamente, también se encuentran disponibles mediante el consumo de los web services activos. La información debe cumplir con las características necesarias para contribuir con el correcto funcionamiento del sistema.

La exactitud en la información que presenten las interfases del sistema y los servicios disponibles deberá ser una característica esencial en el sistema para satisfacer las necesidades del cliente.

La administración del sistema se llevarán a cabo de manera ágil, brindando un manejo fácil y óptimo del sistema permitiendo un mantenimiento cómodo de la información. El sistema poseerá las siguientes interfases para su administración y mantenimiento.

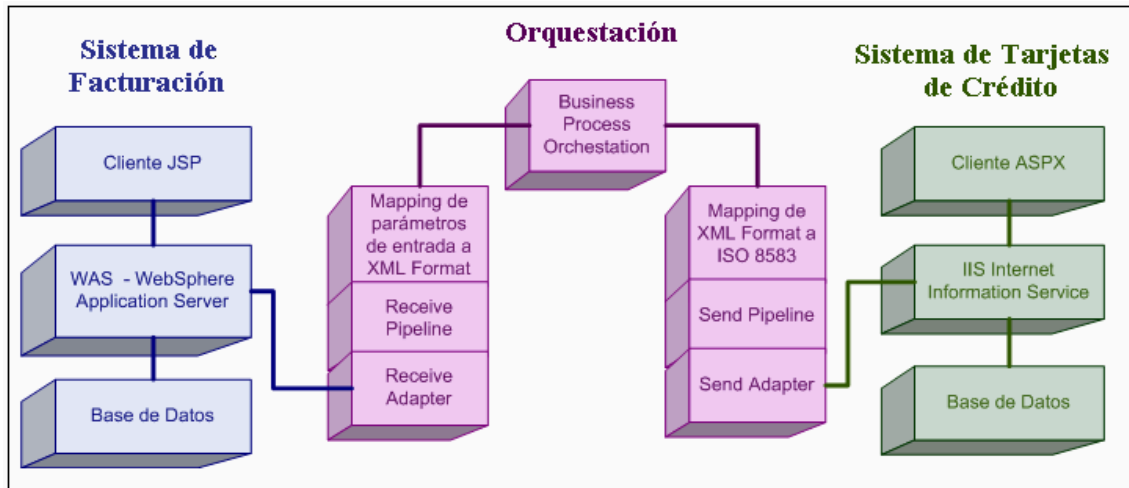
- Crear. Permite la creación de tarjetas de créditos y tarjeta habientes con todos los datos requeridos por el sistema.
- Consultar. Permite realizar búsquedas de determinada información del sistema.
- Modificar. Permite actualizar la información de tarjetas de crédito y tarjeta habientes previamente ingresados al sistema.
- Eliminar. Permite borrar información de tarjetas de crédito y tarjeta habientes previamente creados del sistema.

Con estas operaciones básicas el administrador puede garantizar un mantenimiento rápido y de calidad de la información del sistema, todo esto mediante la utilización de las interfases amigables del sistema con la seguridad requerida en un proceso de este tipo.

2.2 DISEÑO

El Sistema integrado para facturación de supermercados y pago con tarjeta de crédito se ejecutará al menos en cinco nodos como se muestra en la siguiente figura.

Figura II - 22 Diagrama de Organización funcional del Sistema Integrado.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

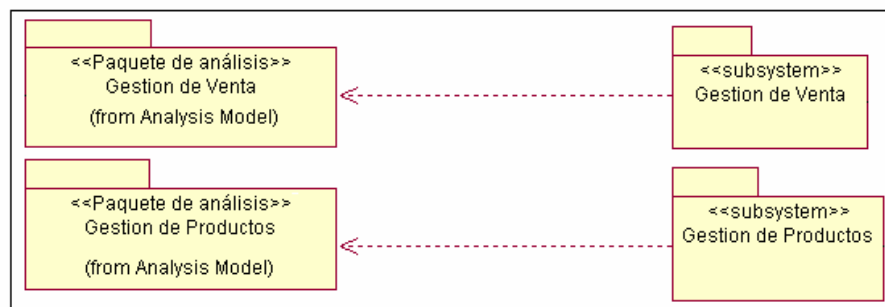
El sistema de Facturación de supermercados por ser un sistema distribuido en capas posee tres nodos básicos que son: La capa del cliente de tipo JSP que puede ser accedida por uno o varios clientes concurrentemente. La capa del servidor de la aplicación que se encuentra gestionada en WAS – WebSphere Application Server, que atenderá las peticiones de los clientes y finalmente la capa donde se coloca el motor de base de datos.

En el sistema de Tarjetas de crédito la distribución es básicamente la misma pero implementado con la Plataforma Microsoft .Net, donde el cliente es de tipo aspx. La capa del servidor que se encuentra subida en Internet Information Server y finalmente el repositorio de datos.

En el orquestador se tiene varios elementos, primeramente los datos del sistema de facturación tienen un formato de datos genérico por consumir un web service

mientras que el sistema de tarjeta de crédito recibe tramas de tipo ISO 8583 por lo que la orquestación además de soportar la múltiple plataforma de programación de las aplicaciones debe cambiar el formato de los datos a un tipo genérico que en este caso es XML. En la orquestación los datos se maneja por adaptadores de entrada y salida, se arman conductos para enrutar la información de entrada y salida y se mapean los datos, los de entrada del sistema de facturación se los pasa de formato genérico a XML y luego se los convierte al otro lado en ISO 8583 para enviarlos al sistema de tarjeta de crédito.

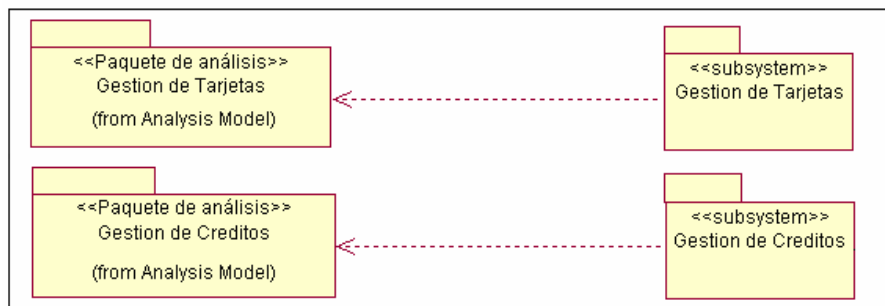
Figura II - 23 Identificación de subsistemas de diseño a partir de paquetes de análisis del Sistema de Facturación de Supermercados.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

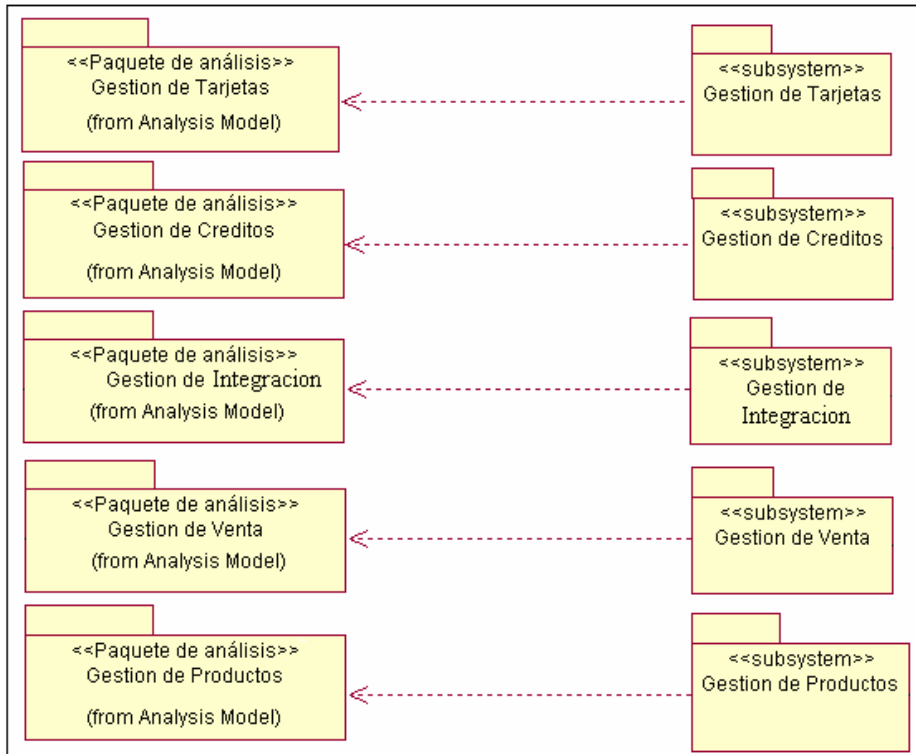
Figura II - 24 Identificación de subsistemas de diseño a partir de paquetes de análisis del Sistema de Tarjeta de Crédito.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Figura II - 25 Identificación de subsistemas de diseño a partir de paquetes de análisis del Sistema Integrado.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.2.1 DISEÑO DE CASOS DE USO

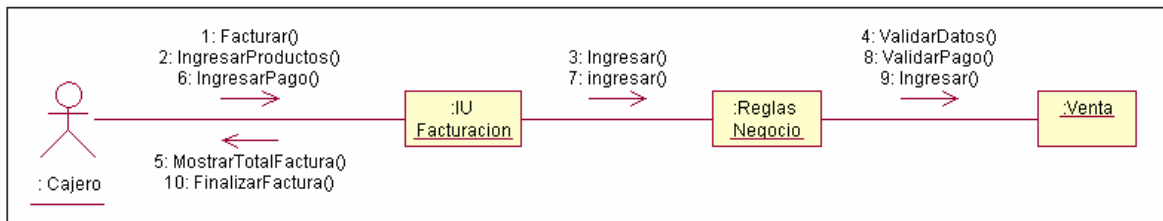
Los principales diagramas de colaboración que pertenecen al diseño de los casos de uso se presentan a continuación. Se encuentran diseñados de acuerdo a los subsistemas identificados, sin perder de vista que cada uno de los subsistemas pueden tener uno o más casos de uso que interactúan en su interior.

Los diagramas de colaboración de diseño completos se encuentra en el anexo titulado “Diagramas de colaboración de diseño” ubicado en el CD adjunto.

2.2.1.1 Subsistema de Facturación de supermercados.

A continuación se presentan los principales diagramas de colaboración del subsistema, los casos de uso del módulo administrativo se encuentran en el anexo mencionado anteriormente.

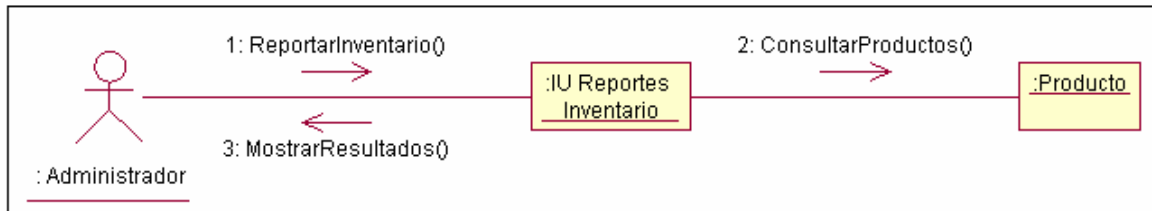
Figura II - 26 Diagrama de colaboración de una realización del caso de uso Generar Venta



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

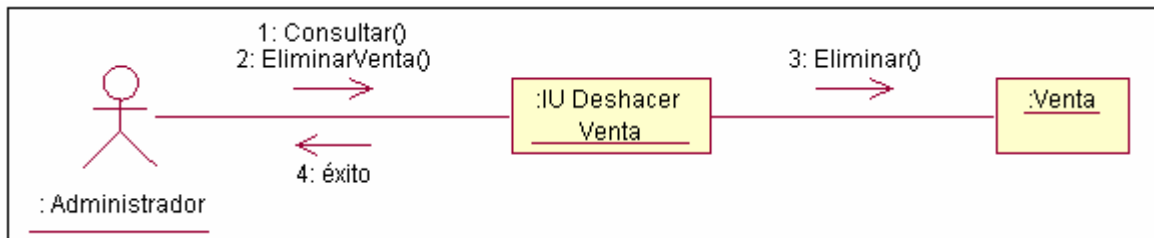
Figura II - 27 Diagrama de colaboración de una realización del caso de uso Reportar Inventario



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Figura II - 28 Diagrama de colaboración de una realización del caso de uso Deshacer Venta



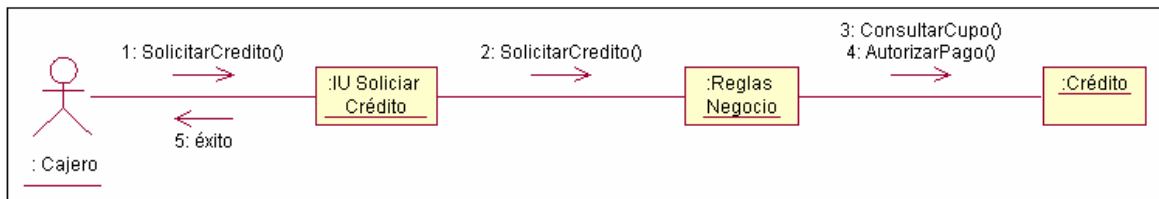
Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.2.1.2 Subsistema de Tarjeta de Crédito.

A continuación se presentan los principales diagramas de colaboración del subsistema, los casos de uso del módulo administrativos se encuentran en el anexo titulado “Diagramas de colaboración diseño” ubicado en el CD adjunto.

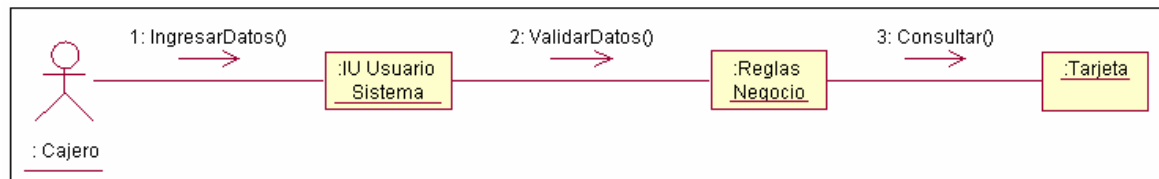
Figura II - 29 Diagrama de colaboración de una realización del caso de uso Solicitar Crédito



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Figura II - 30 Diagrama de colaboración de una realización del caso de uso Validar Tarjeta



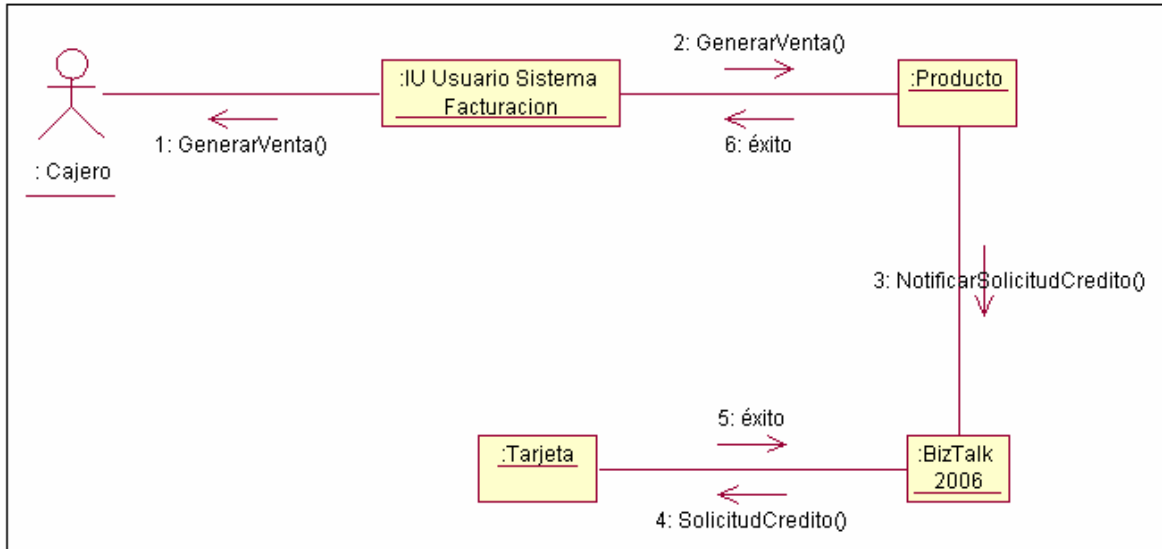
Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.2.1.3 Subsistema de Integración.

A continuación se presenta el diagrama de colaboración para el sistema integrado. Con un mayor detalle se encuentra la información en el anexo titulado “Diagramas de colaboración diseño” ubicado en el CD adjunto.

Figura II - 31 Diagrama de colaboración de una realización de la integración de las aplicaciones.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

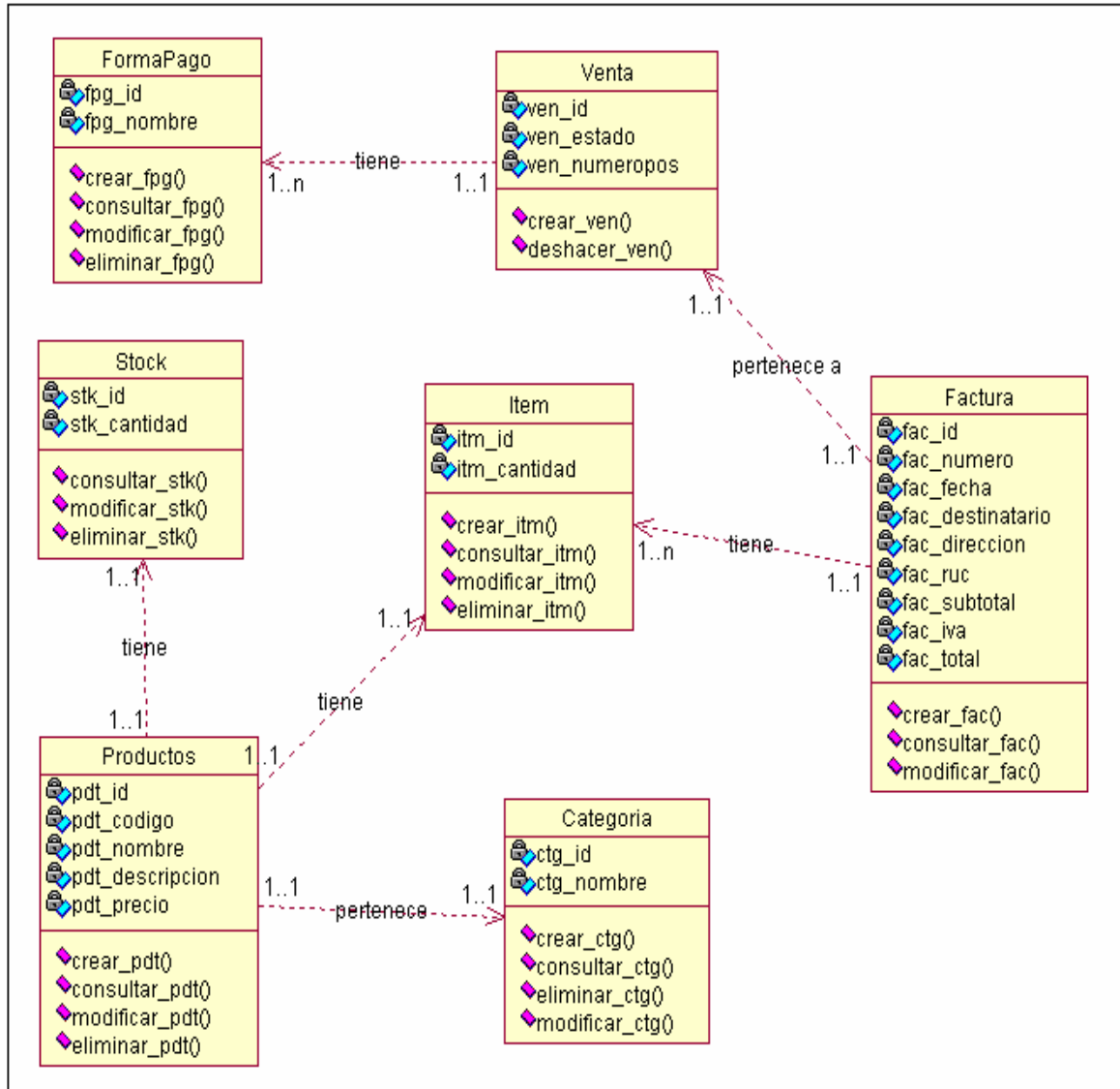
2.2.2 DIAGRAMA DE CLASES DE DISEÑO

Las clases de diseño se las realiza para cada uno de los sistemas de manera independiente por las características descritas en los puntos de análisis de los sistemas anteriores.

A continuación se presentan los diagramas de clases de diseño tanto para el sistema de tarjeta de crédito como para el sistema de facturación de supermercados, teniendo en cuenta que en este punto la orquestación no posee una base de datos ni un diseño de clases como tal pues muchos de los adaptadores con utilizados de manera transparente para el usuario.

2.2.2.1 Diagrama de clases de diseño para el sistema de facturación de supermercados.

Figura II - 32 Diagrama de clases de diseño del Sistema de Facturación de Supermercados.

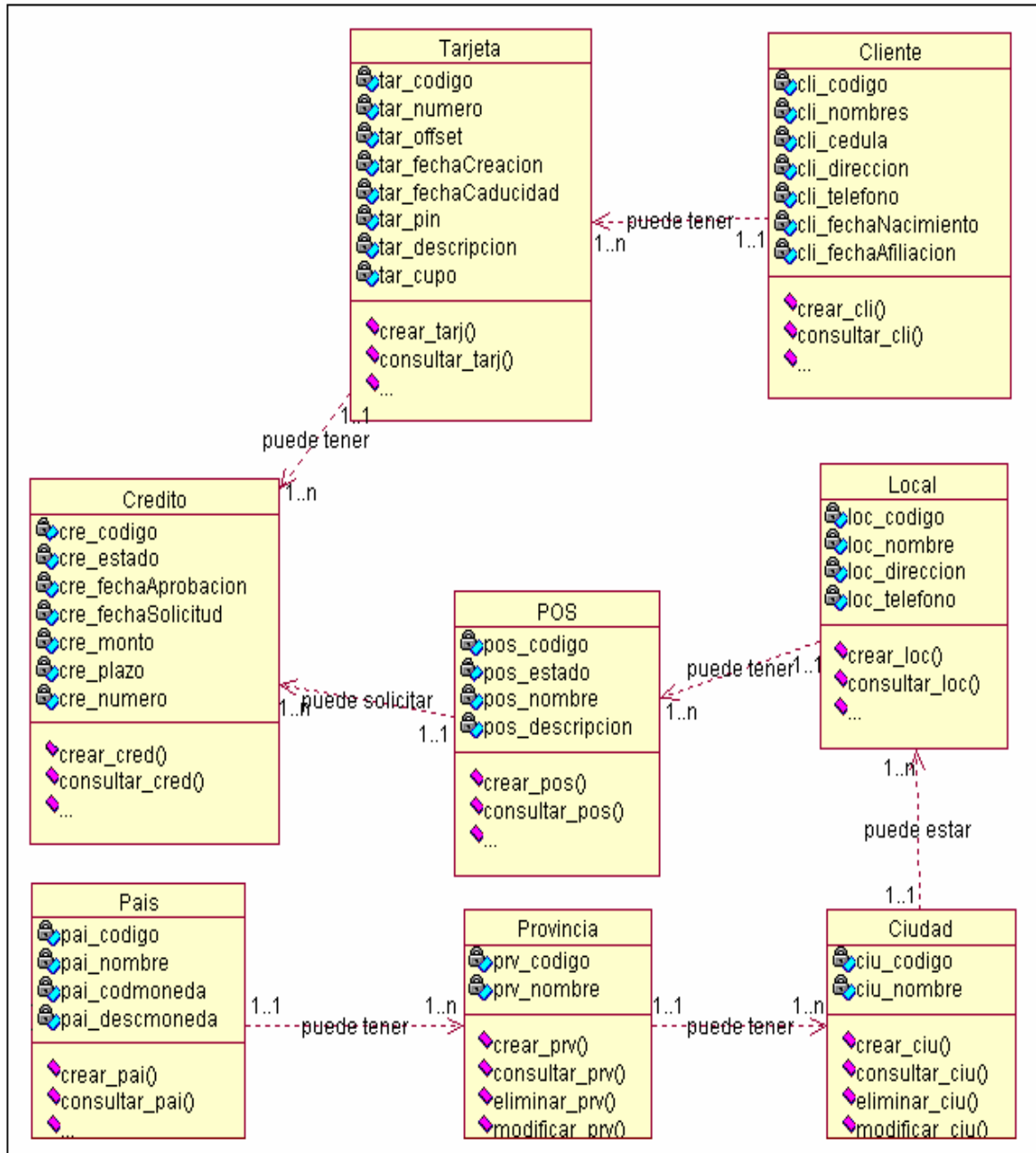


Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.2.2.2 Diagrama de clases de diseño para el sistema de tarjetas de crédito.

Figura II - 33 Diagrama de clases de diseño del Sistema de Tarjetas de Crédito.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.2.3 DISEÑO DE LOS SITIOS DE LOS SISTEMAS

Dentro de la etapa de diseño de las interfases tanto administrativas como las del servicio que brinda el sistema como tal, se deben considerar aspectos fundamentales del diseño como el mapa de navegación, la estandarización de las pantallas a lo largo del sistema y el diseño estructural de los sitios.

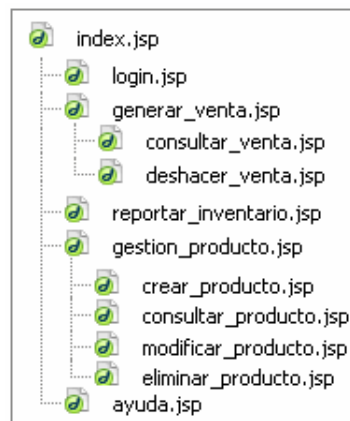
Se debe tener en cuenta que cada tipo de página tiene un propósito diferente, la que tengan una utilidad similar se las englobará dentro de un mismo esquema para mejor navegabilidad, todas mantienen un diseño básico estándar que preserva a lo largo del sistema, trabajando con interfases amigables y fáciles de manejar.

2.2.3.1 Mapa de navegación del sitio Web.

2.2.3.1.1 Mapa de navegación del sitio Web del sistema de facturación de supermercados.

A continuación se presenta el mapa de navegación del sistema de facturación de supermercados, el mismo que fue diseñado según las actividades que realizarán tanto el Administrador del sistema como el cajero.

Figura II - 34 Mapa de navegación del sitio Web del sistema de facturación de supermercados.



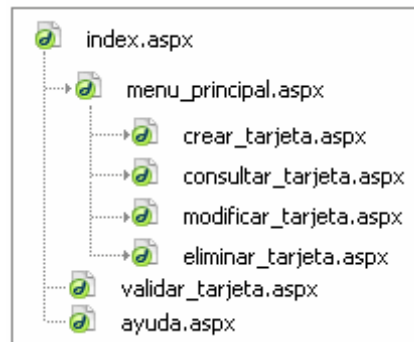
Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.2.3.1.2 Mapa de navegación del sitio Web del sistema de tarjeta de crédito.

A continuación se presenta el mapa de navegación del sistema de tarjetas de crédito, el mismo que fue diseñado según las actividades que realizarán los usuarios del sistema.

Figura II - 35 Mapa de navegación del sitio Web del sistema de tarjeta de crédito.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.2.3.2 Diseño de la estructura del sitio del Sistema de Facturación de Supermercados.

De acuerdo con lo planteado en la sección de análisis del sistema y el mapa de navegación del mismo, podemos plantear el diseño de las siguientes pantallas.

Los tipos de pantallas dentro del sistema de facturación son las siguientes:

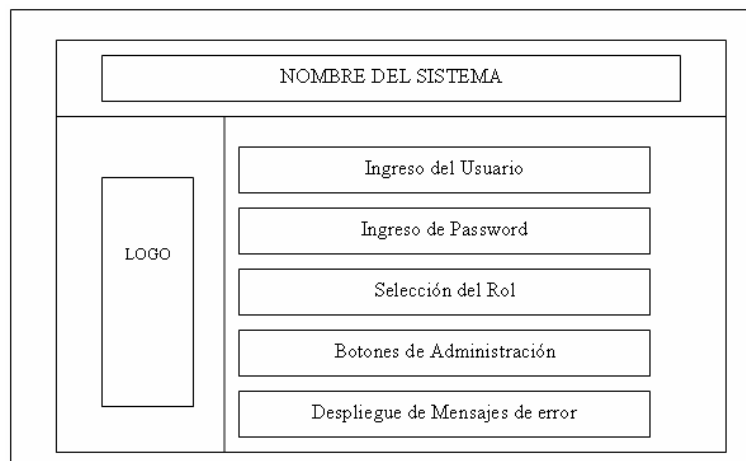
- Index
- Menú Principal
- Crear y Modificar
- Consultar
- Eliminar
- Generar Factura
- Deshacer Venta

2.2.3.2.1 *Index.*

Este tipo de página contendrá la presentación del sistema, es decir el nombre con su respectivo logo. El objetivo principal de esta página es que los usuarios puedan ingresar al sistema con sus respectivos permisos, esto se logra mediante el ingreso del usuario, password y selección del rol para su posterior validación. El texto que se ingresa en el campo de password se encuentra debidamente protegido para evitar que personas ajenas lo descubran.

En este tipo de ventana se tendrán 2 tipos de botones. El botón Aceptar que permite ingresar los campos llenos al sistema para su validación y el botón Cancelar que permite borrar las cadenas ingresadas.

Figura II - 36 Diseño de la página Web tipo Index



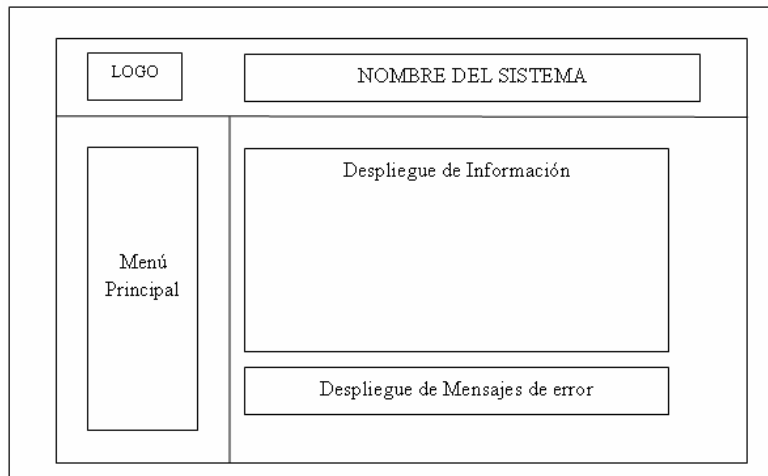
Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.*

Elaboración: Hidalgo, Karina.

2.2.3.2.2 *Menú Principal.*

Este tipo de página contiene el Nombre del sistema con su respectivo logo, en la parte izquierda se encuentra de manera permanente a lo largo de todo el sistema el menú que permite una administración rápida de la información que contiene el sistema. Posee además una sección donde se despliegan mensajes de satisfacción o error en caso de existirlos.

Figura II - 37 Diseño de la página Web tipo Menú Principal.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

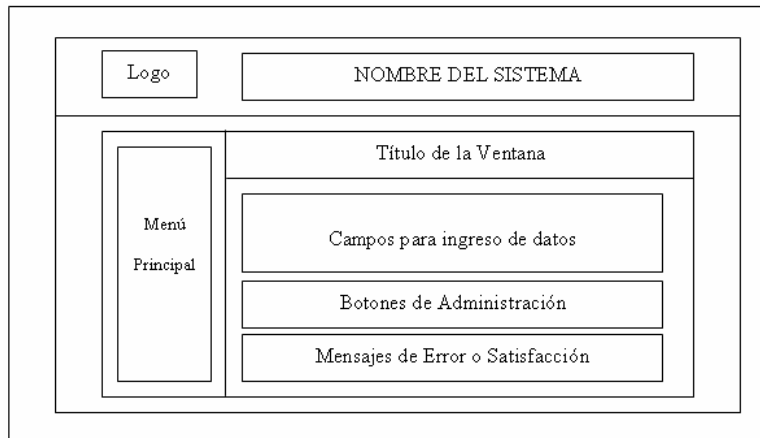
Elaboración: Hidalgo, Karina.

2.2.3.2.3 *Crear y Modificar.*

Las pantallas de tipo Crear y Modificar contienen el título y logo del sistema, contienen el menú principal en la parte izquierda. Posee además el título de la ventana y los campos para el ingreso de los datos requeridos por el sistema. En la sección de Botones de administración encontramos el botón Crear que permite el ingreso de los datos al sistema luego de ser validados, y el botón Limpiar que permite limpiar los datos de los campos existentes.

Tanto la pantalla Crear como la Pantalla Modificar poseen básicamente la misma estructura de diseño, cambia solamente en vez del botón Crear se encuentra el botón modificar, es por esto que en este punto del diseño se le ha colocado como una sola.

Figura II - 38 Diseño de la página Web de tipo Crear o Modificar.



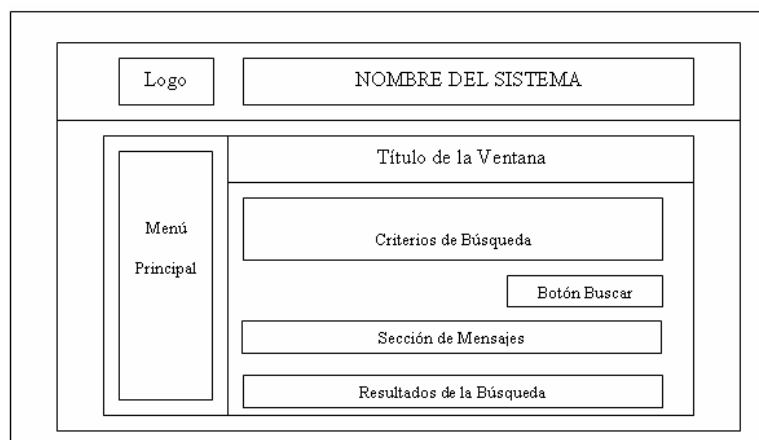
Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.2.3.2.4 Consultar.

Las pantallas del tipo Consultar contiene el nombre del sistema y su logo, el menú principal en la parte izquierda, el título de la ventana y los campos para ingresar los criterios con los que se va a realizar la consulta. Posee un botón Buscar para iniciar la consulta, una sección de mensajes en caso de existirlos y la sección donde se desplegarán los resultados de la consulta.

Figura II - 39 Diseño de la página Web tipo Consultar.



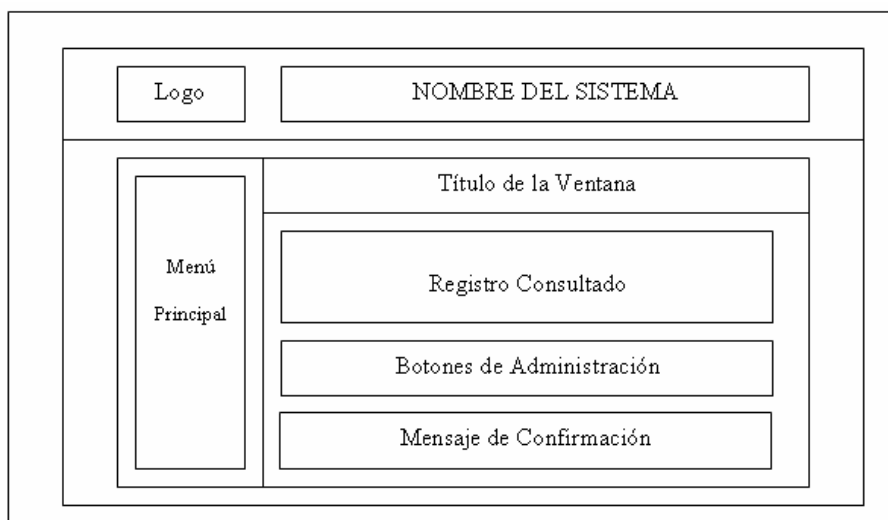
Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.2.3.2.5 *Eliminar.*

La pantalla de Eliminar permite luego de realizar el proceso de búsqueda de resultados el borrar de la base de datos el registro seleccionado. La pantalla de Eliminar contiene el nombre del sistema, el logo, el menú principal, el título de la ventana y el registro previamente consultado cargado en la ventana. Tiene la sección de botones para administración y la sección de mensajes.

Figura II - 40 Diseño de la página Web tipo Eliminar.



Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.*

Elaboración: Hidalgo, Karina.

2.2.3.2.6 *Generar Factura.*

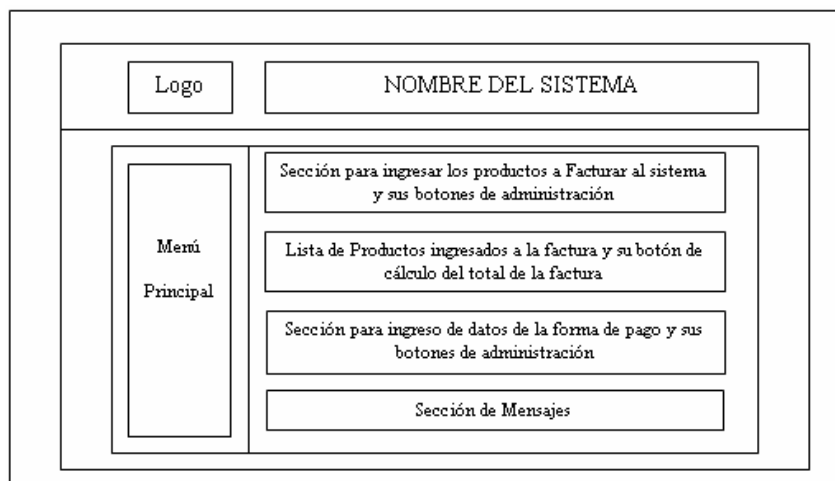
La interfase que permite la generación de facturas consta del nombre del sistema con su respectivo logo, el menú principal y las siguientes secciones:

En la Sección de Ingreso de productos a la factura se encuentran los campos para que con el ingreso del código y dando clic sobre el botón Ingresar se carguen los datos de este producto automáticamente, permite adjuntar la cantidad de estos productos a ingresar en el sistema. Posee los botones Ingresar para ingresar esos datos a la factura y Cancelar para eliminar los datos ingresados en los campos requeridos.

Luego se tiene la sección que lista los productos ingresados, en esta sección se cargan los datos antes ingresados en una grilla permitiendo el eliminarlos de la factura desde este punto. Muestra además los totales parciales y totales a cancelar, éstos se calculan al dar clic sobre el botón Calcular Total.

Finalmente en la sección Datos de Pago se presentan los campos a llenar con los datos del tarjeta habiente en caso de que el pago sea realizado con tarjeta de crédito. Contiene los botones Aprobación que permite enviar los datos al sistema de tarjetas de crédito para solicitar un crédito y el botón Cancelar que permite eliminar los datos ingresados en esta sección.

Figura II - 41 Diseño de la página Web tipo Generar Factura



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

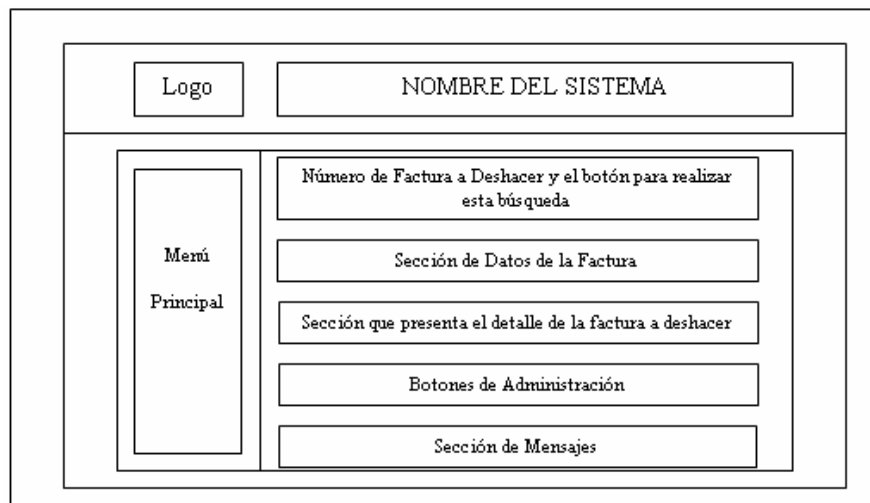
Elaboración: Hidalgo, Karina.

2.2.3.2.7 *Deshacer Venta.*

La interfase que permite deshacer la última venta según las políticas del supermercado contiene el nombre del sistema con su respectivo logo, el menú principal y el área para el ingreso del número de la factura con su respectivo botón Buscar para cargar en la interfase los datos de misma.

Seguidamente se tiene la sección de datos de la factura, aquí se cargarán los datos en caso de que el número de la factura buscado exista. Luego se tiene la sección del detalle de correspondiente a esa factura y los botones de Deshacer Venta para eliminarla del sistema y el botón Cancelar para regresar al menú principal. Finalmente se tiene la sección para la exposición de mensajes satisfactorios o de error en caso de existirlos.

Figura II - 42 Diseño de la página Web tipo Deshacer Venta.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

El diseño de la página web tipo Reportar inventario no consta descrito pues es una consulta del total de los productos y su stock, esto se presenta dentro de una tabla por lo que no amerita una descripción de la misma.

2.2.3.3 Diseño de la estructura del sitio del Sistema de Tarjeta de Crédito.

Para el sistema de Tarjetas de crédito se tiene las siguientes páginas, cada una de ellas tiene un propósito diferente, pero todas mantienen un mismo estándar de diseño, facilitando la administración, mantenimiento y utilización de las mismas.

Los tipos de páginas pertenecientes al sistema de Tarjeta de crédito son las siguientes:

- Index
- Menú principal y consultar
- Crear Nueva y Modificar Tarjeta
- Eliminar Tarjeta.

2.2.3.3.1 *Index.*

Este tipo de página contendrá el nombre del sistema y el logo del mismo. El objetivo principal de esta página es que el usuario Administrador pueda ingresar al sistema mediante su usuario y clave, esta página permite el ingreso de esta información al sistema para su validación. Los datos ingresados a esta página son debidamente protegidos para evitar usurpaciones de identidad.

En esta ventana existen cuatro distintos tipos de botones que se describen a continuación:

Tabla II - 25 Tipos de Botones de la página Web tipo Index.

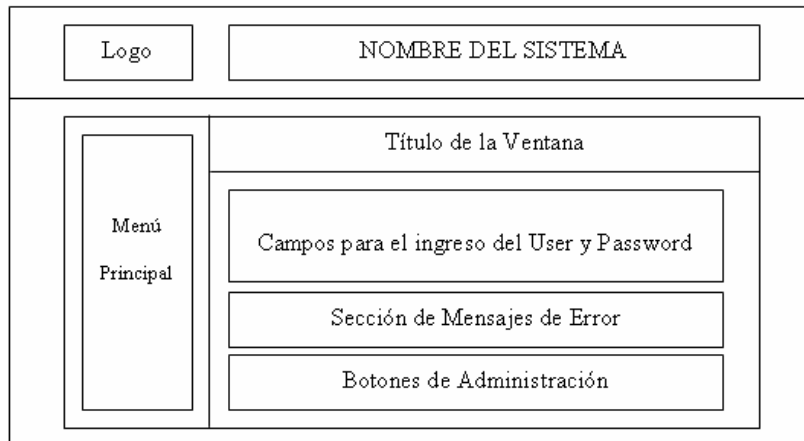
Tipos de Botones	Utilidad
Aceptar	Permite ingresar al sistema con los datos colocados en los campos libres.
Cancelar	Permite borrar la cadena de caracteres ingresada por el Administrador para intentar la autenticación nuevamente.
Salir	Permite abandonar la aplicación.
Cambiar Password	Permite que el usuario cambie su password antiguo por uno nuevo.

Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.*

Elaboración: Hidalgo, Karina.

El diseño de la página de login es el siguiente:

Figura II - 43 Diseño de la página Web de tipo Index.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

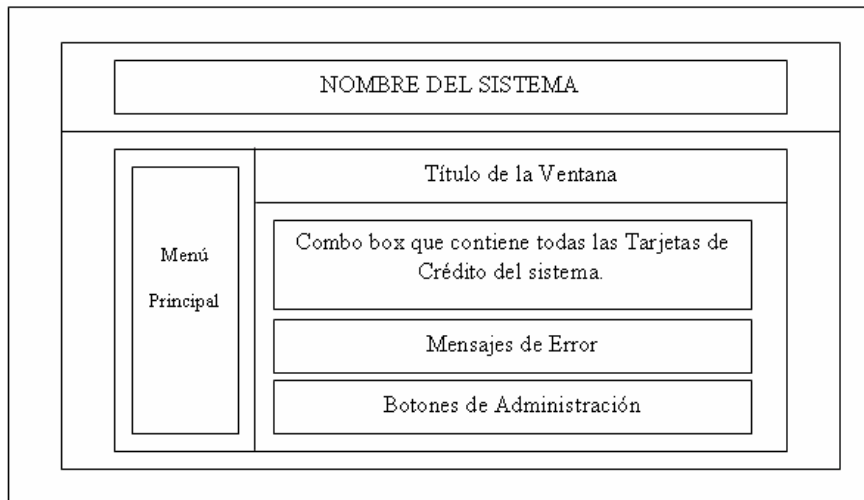
2.2.3.3.2 *Menú Principal y Consultar.*

En esta interfase se encuentra el nombre del sistema, y todas las tarjetas existentes en la base de datos hasta el momento presentadas mediante un combo box.

Luego se tiene la sección de botones que permiten la administración de las tarjetas. Los botones que se presentan son: Crear que permite acceder a la página para crear tarjetas. Modificar que luego de señalar la tarjeta a modificar en el combo box permite llevar los datos a la página de edición. El botón Eliminar que permite luego de señalada la tarjeta a eliminar en el combo box eliminarla del sistema. Finalmente el botón Cancelar que permite regresar al menú principal.

Todo el tiempo tenemos una ventana a la izquierda con todos los datos necesarios para la administración y mantenimiento del sistema.

Figura II - 44 Diseño de la página Web tipo Menú Principal y Consultar.



Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.*

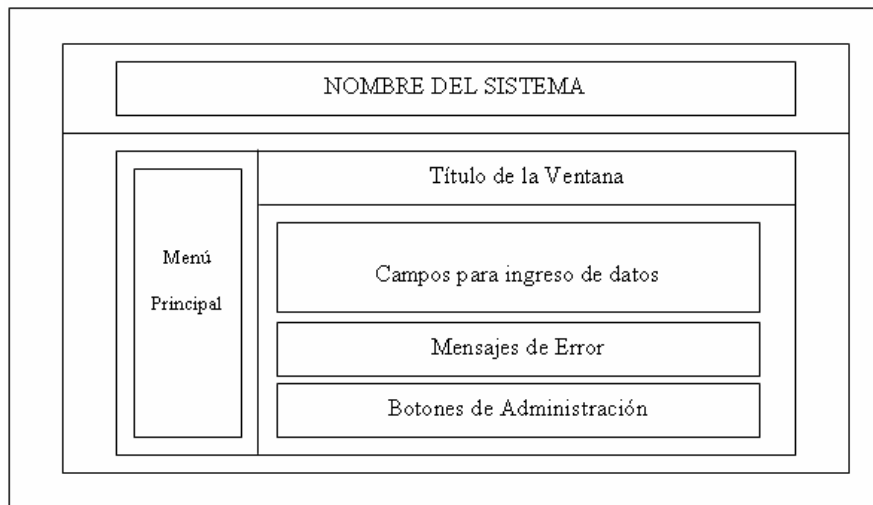
Elaboración: Hidalgo, Karina.

2.2.3.3.3 *Crear Nueva y Modificar Tarjeta*

Este tipo de página contendrá el nombre del sistema, el título de la ventana que nos permite identificar el tipo de acciones que se van a realizar en la misma. Tendrá una sección para el ingreso de los datos de la nueva tarjeta o modificaciones al sistema además de una sección donde se desplegarán los mensajes de satisfacción o de error en caso de existir. En la sección de botones se encontrarán los botones de Crear o Modificar que permite ingresar los datos llenados en los campos al sistema en caso de que sean válidos, Cancelar que permite borrar los datos llenados en los campos y regresar a la página de administración, Salir que permite salir del sistema.

En la parte izquierda de la pantalla se encuentra el Menú principal que nos indica las opciones de administración que posee el sistema.

Figura II - 45 Diseño de la página Web tipo Crear Nueva.



Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.*

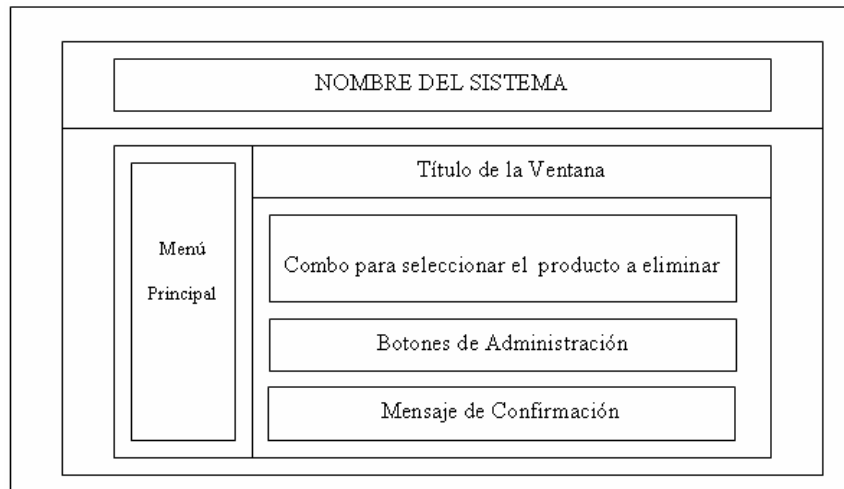
Elaboración: Hidalgo, Karina.

2.2.3.3.4 *Eliminar*

Este tipo de página contendrá el nombre del sistema, el título de la ventana que nos permite identificar el tipo de acciones que se van a realizar en la misma. Este proceso no se lo realiza en una interfase aislada, sino dentro de la interfase principal, ya que mediante el combo con los datos existentes podemos seleccionar fácilmente la Tarjeta a eliminar el sistema.

En todo caso en esta interfase se encuentra la sección con el nombre del sistema, el título de la ventana, el combo para la selección de la tarjeta a eliminar, los botones de administración detallados en la pantalla principal y el menú principal presente durante todo el módulo administrativo.

Figura II - 46 Diseño de la página Web tipo Eliminar.



Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.*

Elaboración: Hidalgo, Karina.

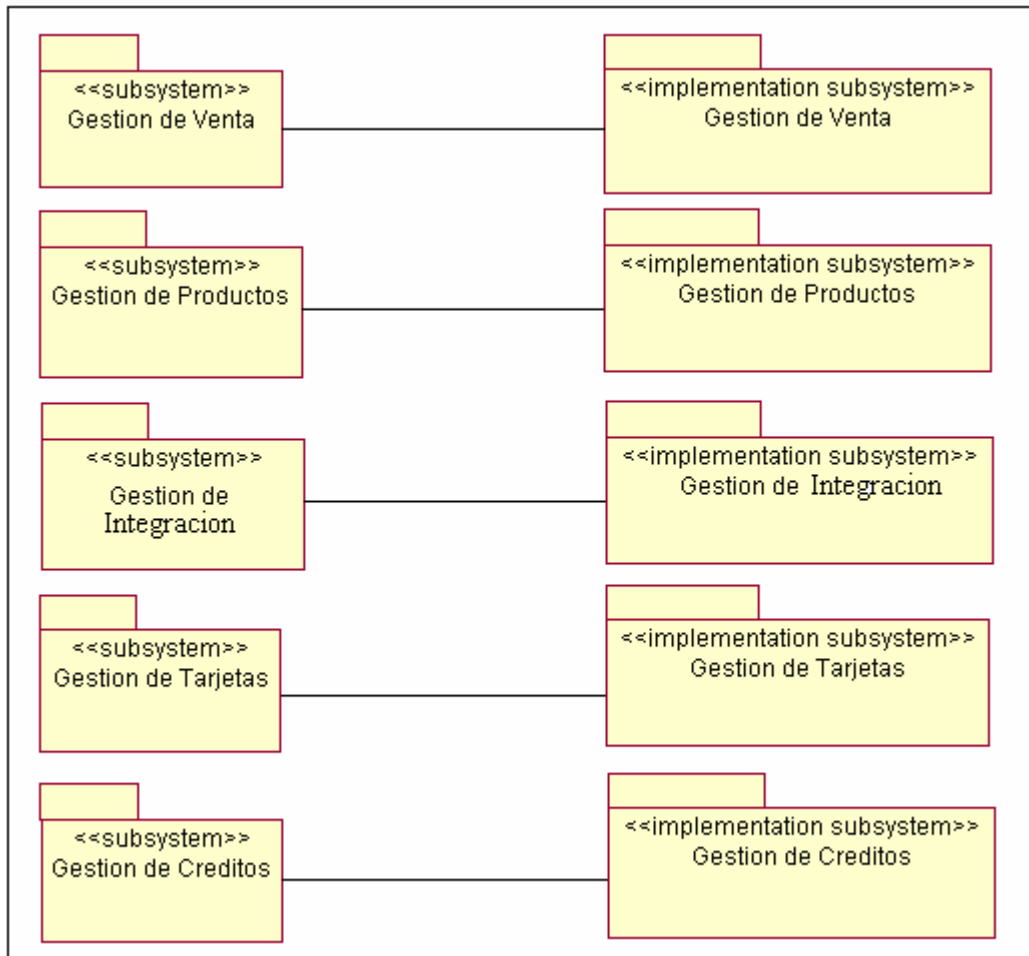
2.3 IMPLEMENTACIÓN

2.3.1 DEFINICIÓN DE SUBSISTEMAS DE IMPLEMENTACIÓN.

Para la definición de los subsistemas de implementación de cada uno de los sistemas y de la integración se ha tomado en cuenta los subsistemas de diseño identificados en la fase de diseño de los sistemas y la orquestación ya que tienen una relación directa en la etapa de implementación, es por esto que la siguiente figura pertenece al sistema integrado.

Los sistemas por separado no son necesarios de colocarlos por su relación con la fase anterior donde se encuentran claramente especificados, en todo caso si desea observarlos se encuentran detallados en el anexo titulado "Diagramas de paquetes y subsistemas" que se encuentra en el CD adjunto.

Figura II - 47 Identificación de subsistemas de diseño a partir de paquetes de análisis



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

En los gráficos subsiguientes se describe cada uno de los subsistemas de implementación identificados a lo largo de este proceso formal de desarrollo, además de las dependencias entre dichos paquetes.

Se debe tener en cuenta que los subsistemas identificados son los que directamente se implementarán con las herramientas que se decidirá utilizar posteriormente.

2.3.1.1 Subsistema de implementación del Sistema de Facturación de Supermercados.

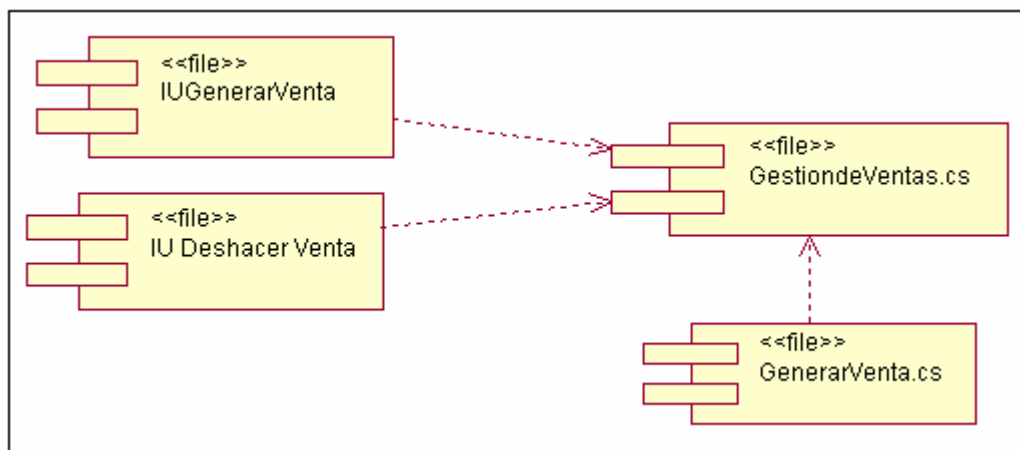
2.3.1.1.1 Subsistema de implementación de Gestión de Venta.

Estos subsistemas están en directa relación con el subsistema de diseño Gestión de Venta, está conformado de varios componentes que a su vez mantienen relación con el subsistema general Generar Venta.

Los componentes de este subsistema tienen como propósito el gestionar las ventas en el supermercado que serán ejecutadas por el cajero como usuario final y por el administrador del sistema.

A continuación se muestra el diagrama de dependencia de los componentes que forman el subsistema de implementación de Gestión de Ventas.

Figura II - 48 Dependencia de los componentes del subsistema de diseño de GestionDeVenta.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

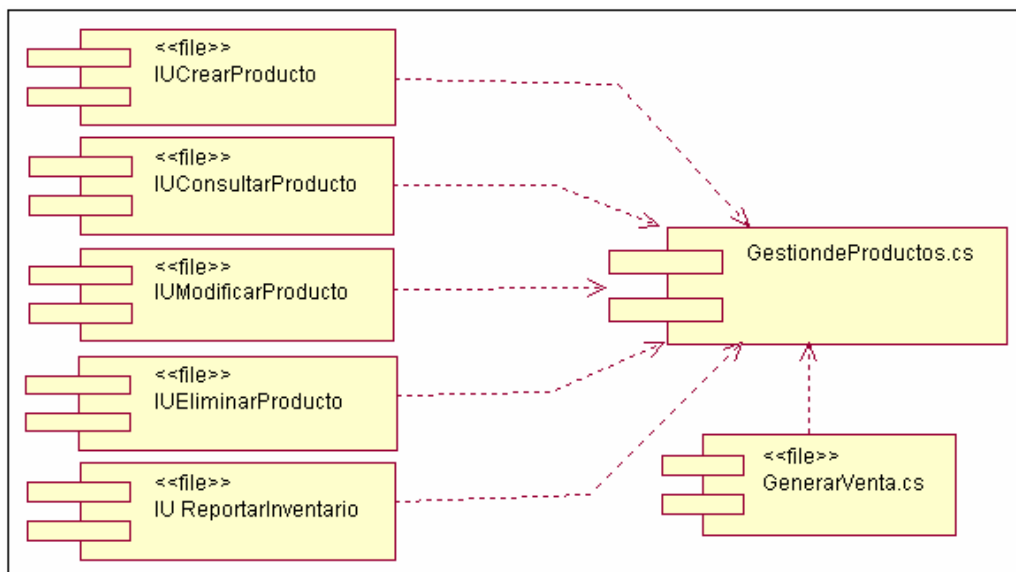
2.3.1.1.2 Subsistema de implementación de Gestión de Productos.

Estos subsistemas están en directa relación con el subsistema de diseño Gestión de Productos, está conformado de varios componentes que a su vez mantienen relación con el subsistema general Generar Venta.

Los componentes de este subsistema tienen como propósito el administrar los productos del sistema, la misma que será ejecutada por el Administrador del sistema.

A continuación se muestra el diagrama de dependencia de los componentes que forman el subsistema de implementación de GestionDeProductos.

Figura II - 49 Dependencia de los paquetes del subsistema de diseño GestionDeProductos



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.3.1.2 Subsistema de implementación del Sistema Tarjetas De Crédito.

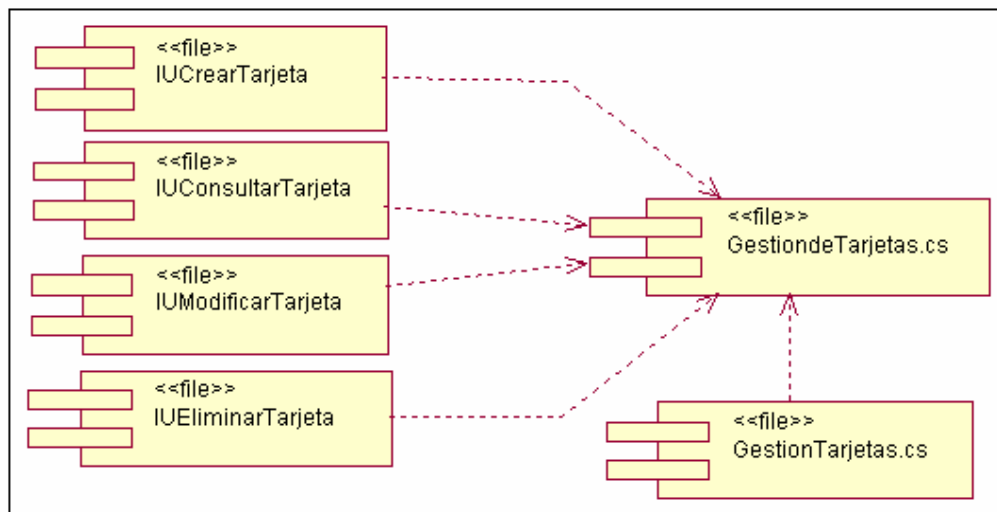
2.3.1.2.1 Subsistema de implementación de Gestión de Tarjetas.

Estos subsistemas están en directa relación con el subsistema de diseño Gestión de Tarjetas, está conformado de varios componentes que a su vez mantienen relación con el subsistema general Gestión Tarjetas.

Los componentes de este subsistema tienen como propósito el administrar las tarjetas y tarjeta habientes del sistema de manejo de tarjetas de crédito, esta tarea será realizada por el administrador.

A continuación se muestra el diagrama de dependencia de los componentes que forman el subsistema de implementación de GestionDeTarjetas.

Figura II - 50 Dependencia de los componentes del subsistema de diseño de GestionDeTarjetas.



Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.*

Elaboración: Hidalgo, Karina.

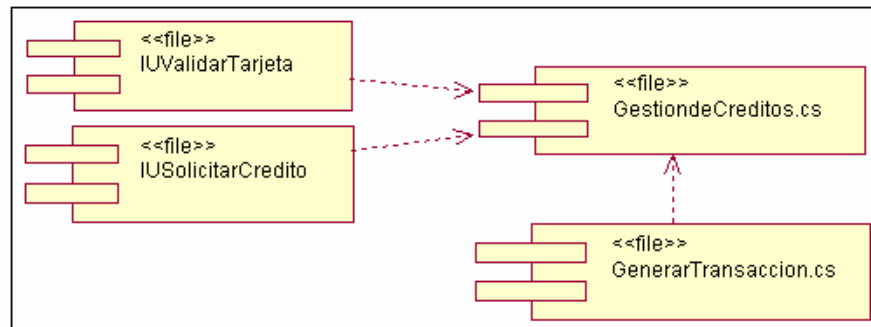
2.3.1.2.2 Subsistema de implementación de Gestión de Créditos.

Estos subsistemas están en directa relación con el subsistema de diseño Gestión de Créditos, está conformado de varios componentes que a su vez mantienen relación con el subsistema general Generar Transacción.

Los componentes de este subsistema tienen como propósito el gestionar los créditos, mediante atención a solicitudes, validación de tarjetas, entre otros servicios que se pondrán a disposición del cliente.

A continuación se muestra el diagrama de dependencia de los componentes que forman el subsistema de implementación de GestionDeCreditos.

Figura II - 51 Dependencias de componentes del subsistema de diseño GestionDeCreditos.



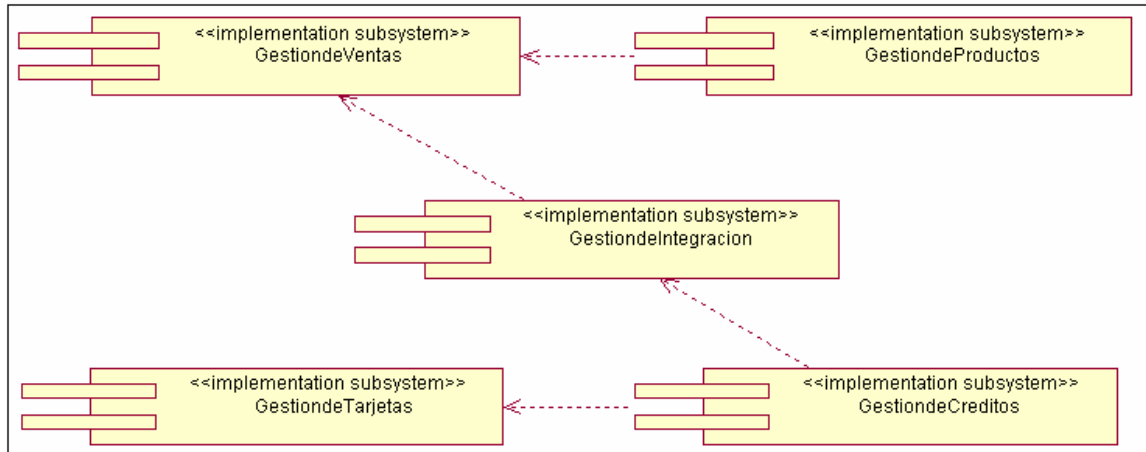
Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.3.2 DEPENDENCIAS ENTRE LOS SUBSISTEMAS DE IMPLEMENTACION.

El proceso llevado a cabo de la definición de subsistemas y sus dependencias hace evidente el hecho de las dependencias o relaciones que existe entre ellos, el hecho de que todos los subsistemas luego de la integración vayan a trabajar en un unísono, nos permite establecer el siguiente diagrama que contiene las dependencias de compilación entre los componentes de los subsistemas de integración.

Figura II - 52 Dependencia de compilación de los componentes de los subsistemas de la integración.



Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.*

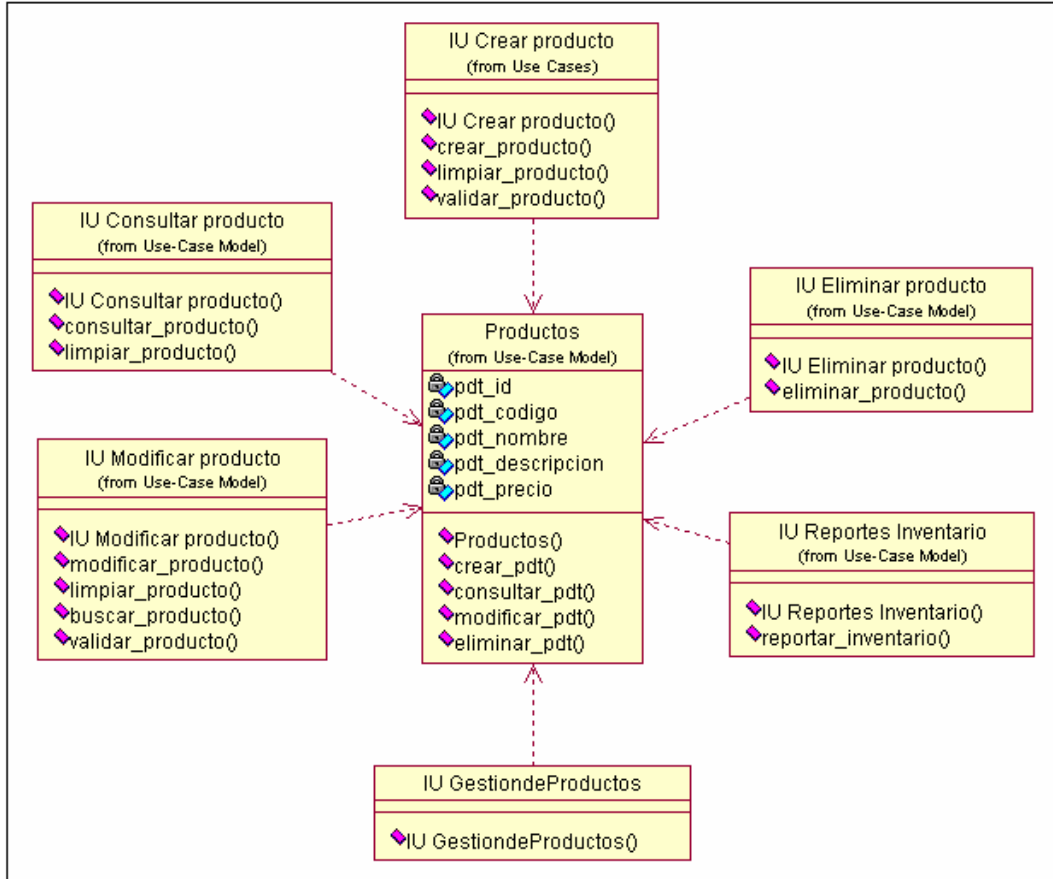
Elaboración: Hidalgo, Karina.

2.3.3 CLASES DE IMPLEMENTACIÓN.

A continuación se presenta las clases de implementación de cada uno de los subsistemas de las aplicaciones desarrolladas a largo de este proyecto e identificados en pasos anteriores.

2.3.3.1 Subsistema de Implementación GestionDeProductos.

Figura II - 53 Diagrama de Clases del subsistema de implementación GestionDeProductos.

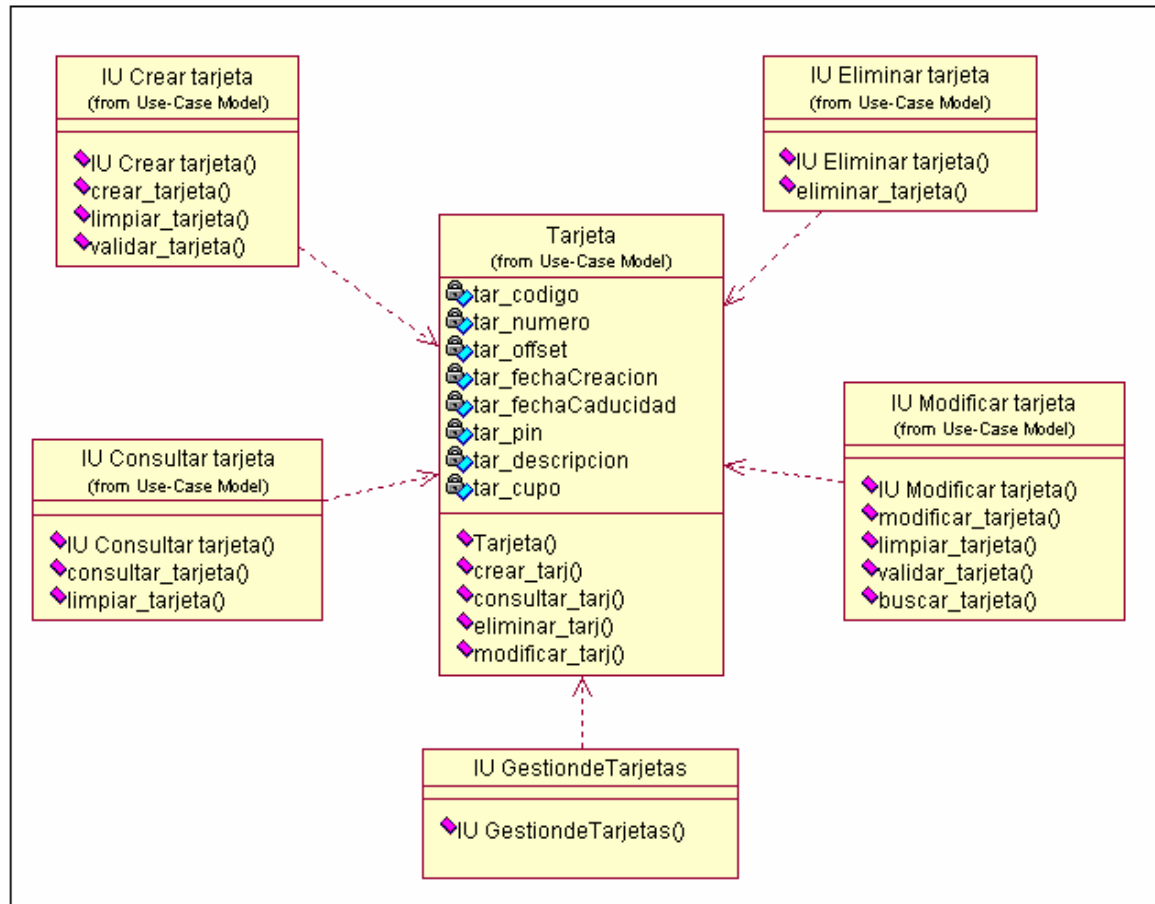


Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.3.3.2 Subsistema de Implementación GestionDeTarjetas.

Figura II - 54 Diagramas de Clases del subsistema de implementación GestionDeTarjetas.

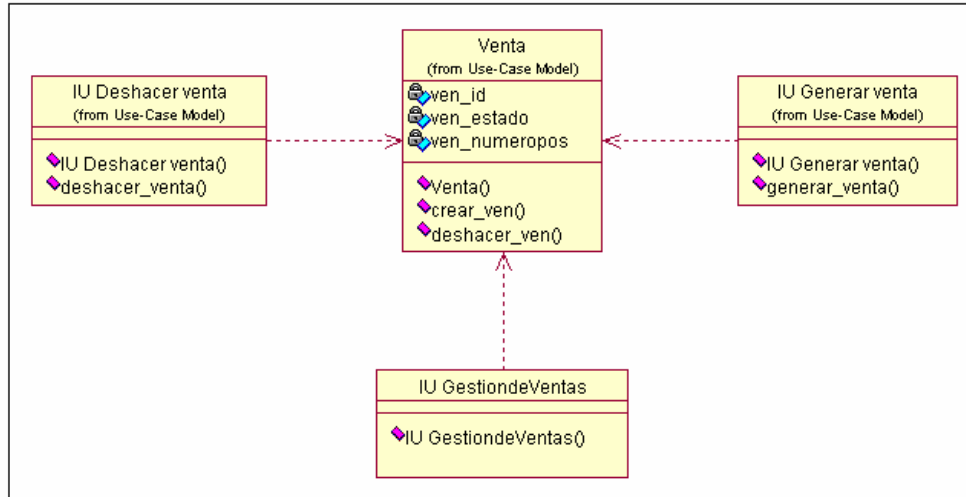


Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.3.3.3 Subsistema de Implementación GestionDeVentas.

Figura II - 55 Diagrama de Clases de implementación GestionDeVentas.

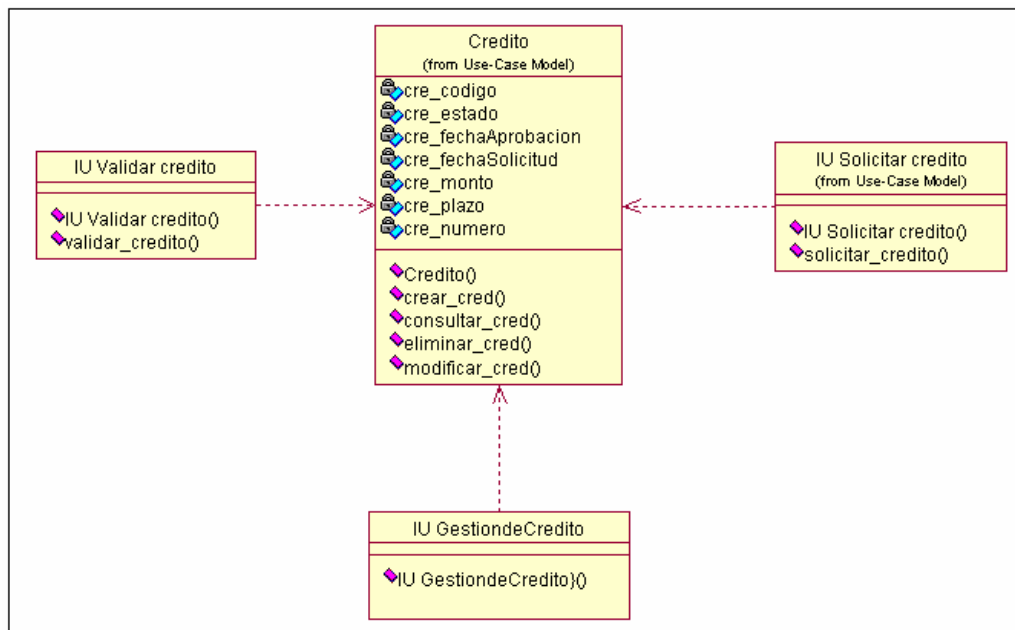


Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.3.3.4 Subsistema de Implementación GestionDeCreditos.

Figura II - 56 Diagrama de Clases del subsistema de implementación GestionDeCreditos.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.3.4 DESARROLLO DE LA APLICACIÓN.

La integración de los sistemas de facturación de supermercados y el sistema de tarjetas de crédito, han sido definidos en subsistemas de implementación a ser accedidos tanto de los clientes de cada uno de los sistemas como en el orquestador de las aplicaciones tanto para brindar servicios como para el mantenimiento desde los módulos administrativos de cada uno de los sistemas.

Las diferentes consideraciones de implementación se realizan por separado tanto para cada uno de los sistemas independientes como para el orquestador. A continuación se definirán para cada uno de los sistemas, subsistemas y orquestador las herramientas a utilizarse, los estándares de programación y desarrollo gráfico resultante.

2.3.4.1 Definición de herramientas para subsistemas pertenecientes al sistema de Facturación de Supermercados.

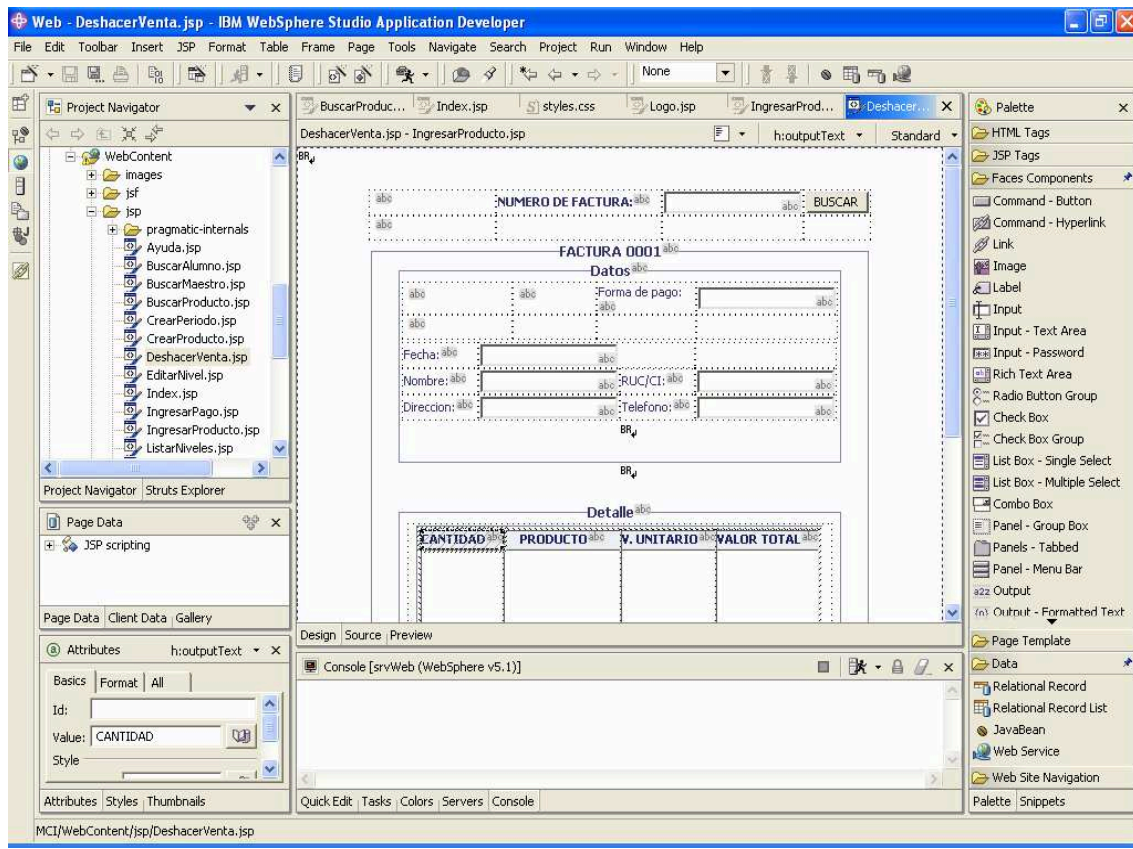
Se definen las herramientas para diseño de las interfases gráficas, manejo de la lógica del negocio y manejo de las estructuras y almacenamiento de los datos. Es decir las herramientas de Front – End, Middle - End y Back – End de la aplicación de Facturación de Supermercados.

2.3.4.1.1 Herramientas para el Front – End

El diseño de las pantallas del sistema, inserción de imágenes, botones y menús del sistema se empleará la herramienta de programación Rational Application Developer que permite el diseño de las interfases además de su programación.

A continuación se presenta la interfaz de trabajo de Rational Application Developer.

Figura II - 57 Intefaz de diseño con WebSphere Studio Application Developer.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

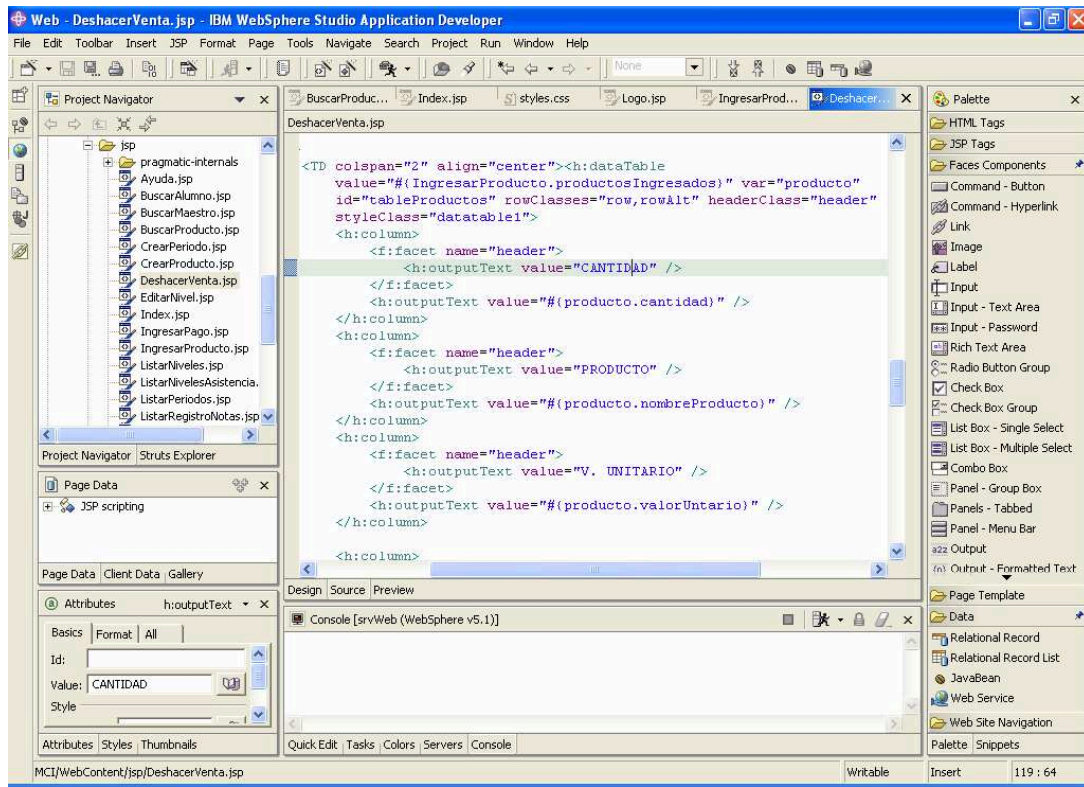
2.3.4.1.2 Herramientas para el Middle – End

El sistema de facturación de supermercados se implementará en J2EE, el código de programación será elaborado con Rational Application Developer con todas las ventajas que esta herramienta brinda a los programadores como son:

- Potencia la programación en J2EE de sistemas transaccionales.
- Entorno de desarrollo amigable.
- Permite el desarrollo de aplicaciones orientadas hacia SOA.
- Integra funcionalidades en la herramienta que permite la realización de pruebas del sistema facilitando el desarrollo.

El servidor donde se ejecuta la aplicación es WebSphere Application Developer.

Figura II - 58 Interfaz de desarrollo con WebSphere Application Developer.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.3.4.1.3 Herramientas para el Back – End

Al ser un sistema de facturación se necesita de una total consistencia en los datos, agilidad en el proceso y fiabilidad, todo esto y con el agregado de la compatibilidad de la herramienta de Middle End escogida el motor de base de datos para este sistema de Facturación de Supermercados será MySQL que además de ser libre es totalmente compatible con J2EE para un mejor rendimiento de la aplicación.

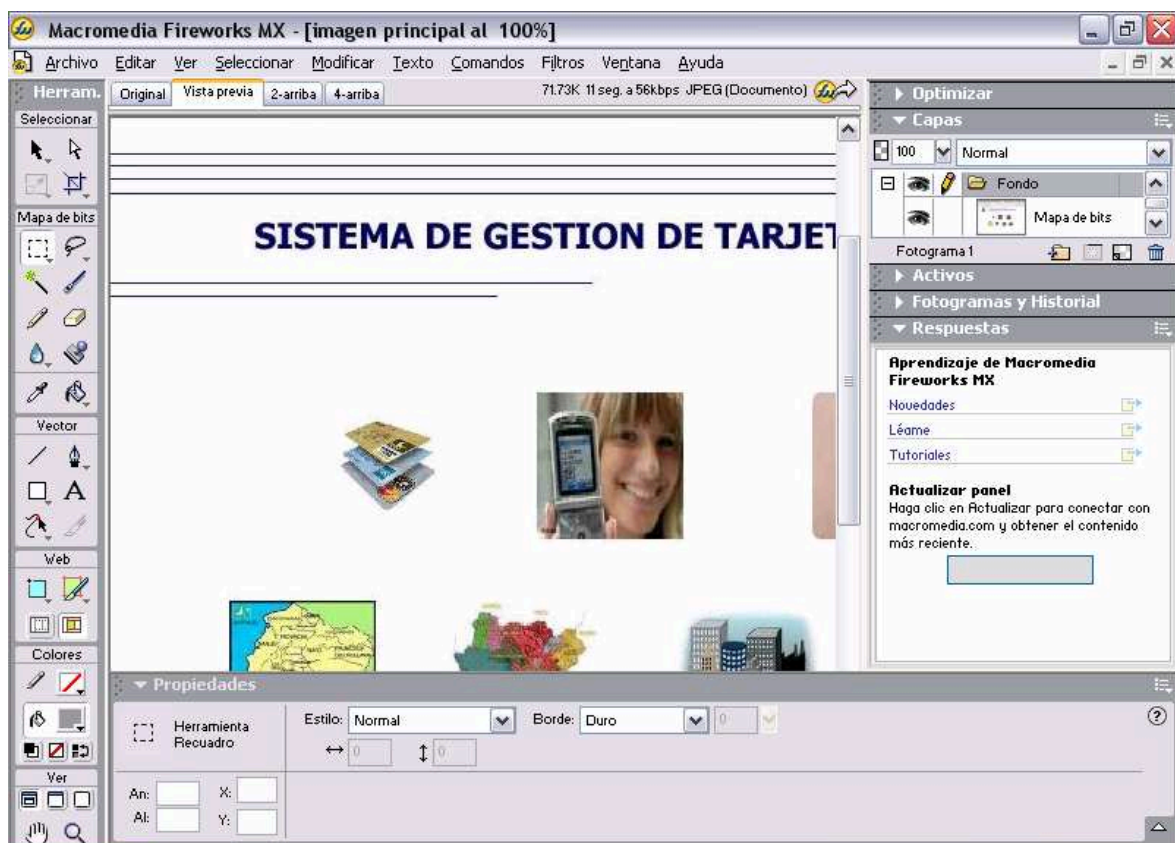
2.3.4.2 Definición de herramientas para subsistemas pertenecientes al sistema de Tarjetas de Crédito.

2.3.4.2.1 Herramientas para el Front – End

Para el diseño de las páginas Web, imágenes y botones del sistema se empleará la herramienta Macromedia Fireworks MX, permitirá la generación de las imágenes de alta calidad requeridas para el sistema, facilitando que las interfases creadas sean amigables para el usuario final, de fácil utilización y siendo totalmente portables a .Net para la programación que tendrán por detrás.

A continuación se presenta la interfaz de trabajo de Macromedia Fireworks.

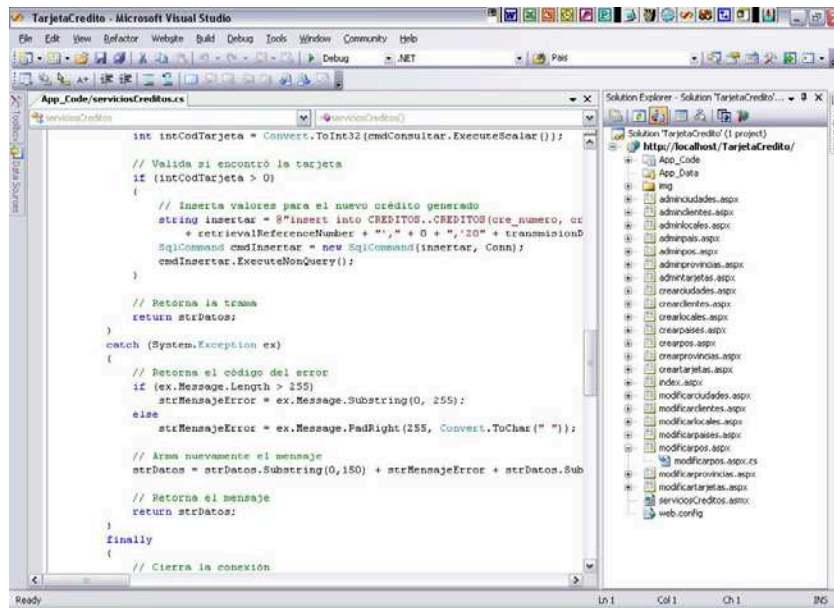
Figura II - 59 Interfaz de trabajo de Macromedia Fireworks MX.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Figura II - 61 Interfaz de programación en Microsoft Visual C# .NET 2005.



Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico*, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.3.4.2.3 Herramientas para el Back – End

Por ser un sistema transaccional que necesita de alta fidelidad y disponibilidad de los datos para atender solicitudes de los clientes, además de por su alta compatibilidad con los productos Microsoft se ha seleccionado como motor de la base de datos a Microsoft SQL Server 2005 para repositorio de la base de datos del sistema de Tarjetas de Créditos.

2.3.4.3 Alcance de la implementación el sistema.

2.3.4.3.1 Sistema de Facturación de Supermercados.

El sistema está implementado en dos módulos. Un módulo administrativo que permite la gestión de productos y usuarios. El segundo módulo que permite la facturación y gestión de ventas.

A continuación se presenta el alcance del módulo administrativo:

- El sitio no dispone de módulo de seguridad.
- El módulo no se encarga de la aprobación de créditos ni de la conexión con el sistema de Tarjetas de crédito.
- El reporte de inventario es sencillo, se limita a presentar el stock de productos en el sistema en una tabla fácil de entender, no se encarga de realizar estadísticas, ni de gestión de históricos.

En el módulo de facturación presenta el siguiente alcance:

- El sistema permite facturar los productos previamente ingresados al sistema, permitiendo como política del negocio el deshacer solo la última venta realizada por el cajero.
- El sistema no emite los comprobantes físicos de la venta.

2.3.4.3.2 Sistema de Tarjeta de Crédito.

El sistema se encuentra implementado en dos partes, el módulo administrativo que permite la gestión de tarjetas de crédito y tarjeta habientes y la sección de servicios web expuestos que atienden solicitudes de crédito.

- Existirá servicios web expuestos para atender solicitudes de aprobación de créditos.
- El sistema no dispondrá de módulo de seguridad.
- La información transmitida entre los sistemas no será encriptada.

2.3.4.4 Estándares de programación.

Para facilitar las tareas de mantenimiento y programación del sistema se hace indispensable el seguir ciertos estándares y normas de calidad que se deben cumplir a los largo del desarrollo del proyecto.

A continuación se definen los estándares de programación tanto para el sistema de Tarjetas de Crédito como para el de Facturación de supermercados.

Tabla II - 26 Estándares de programación para los sistemas a desarrollarse.

ELEMENTO	DESCRIPCION
VARIABLES	<p>Se nombran separando las palabras con un signo “_” y antes del nombre de la variable van las tres primeras letras de la clase, tabla o segmento al que pertenecen. Todas las letras son minúsculas. Por ejemplo.</p> <ul style="list-style-type: none">• pos_codigo• tar_numero• cli_nombre
METODOS	<p>Los métodos se nombran separando las palabras con un signo “_”. Todas las letras son minúsculas. Por ejemplo:</p> <ul style="list-style-type: none">• crear_loc()• eliminar_tar()• modificar_cre()

Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.*

Elaboración: Hidalgo, Karina.

Para nombrar las páginas desarrolladas en los sistemas se considerará la acción que realiza cada una de ellas, es decir el nombre debe tener relación con el servicio que presta y seguidamente se colocará la extensión de la misma, esto dependerá del lenguaje en el que se está programando, es así que para el sistema de tarjeta de crédito las páginas tendrán la extensión .aspx y para el sistema de facturación las páginas tendrán la extensión .jsp como se muestra en los ejemplos a continuación:

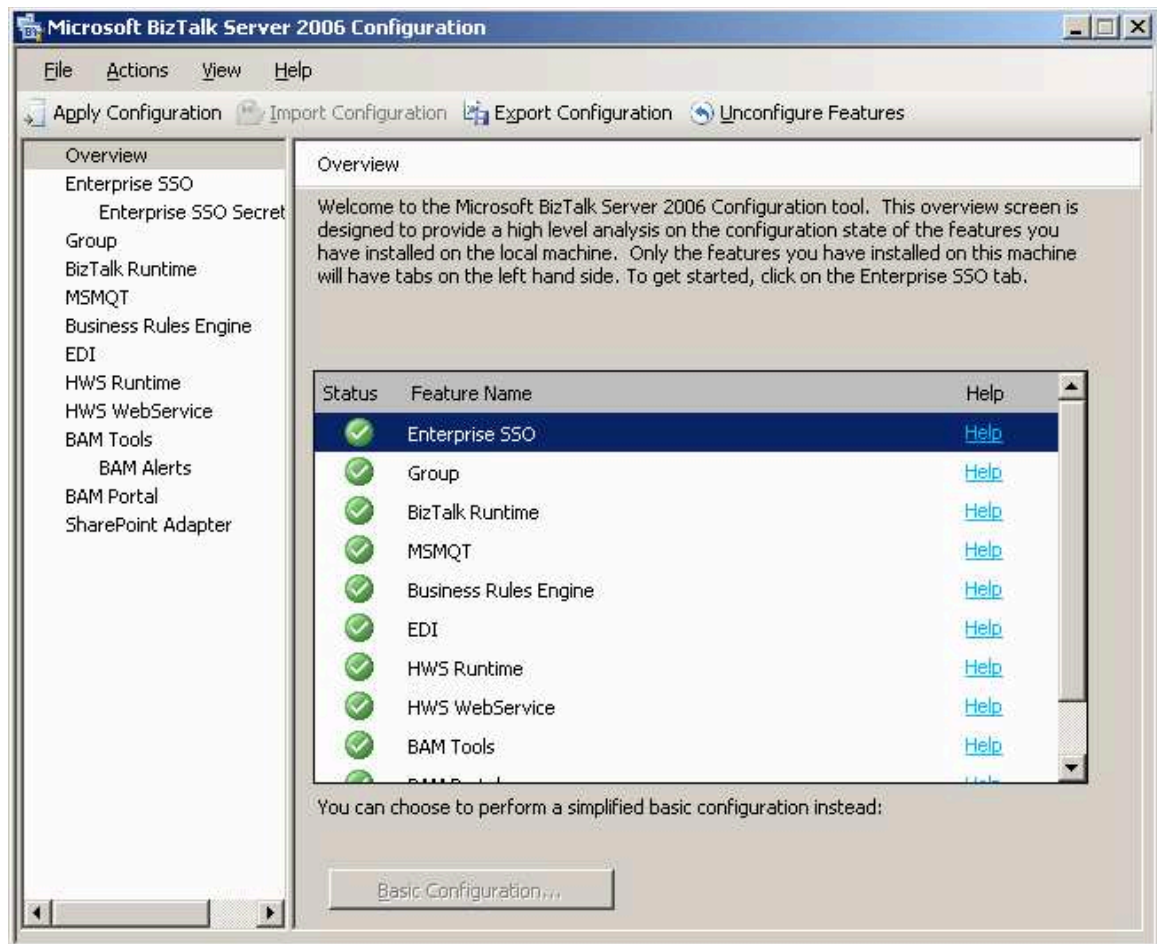
- crear_producto.jsp
- modificar_tarjeta.aspx

2.3.4.5 Estructura de la Integración de las aplicaciones con Microsoft BizTalk Server 2006.

2.3.4.5.1 Configuración del Orquestador.

Luego de la instalación de Microsoft BizTalk Server 2006 y todos los programas y componentes adicionales que requiere se procede a la configuración de el Enterprise SSO, los grupos requeridos, BizTalk Runtime, la construcción de la ingeniería de las reglas del negocio, HWS Runtime, HWS Web Services, Herramientas de monitoreo, alertas, adaptadores, entre otros hasta que la configuración sea satisfactoria y se presente como en la pantalla a continuación.

Figura II - 62 Pantalla de configuración de Microsoft BizTalk Server 2006.



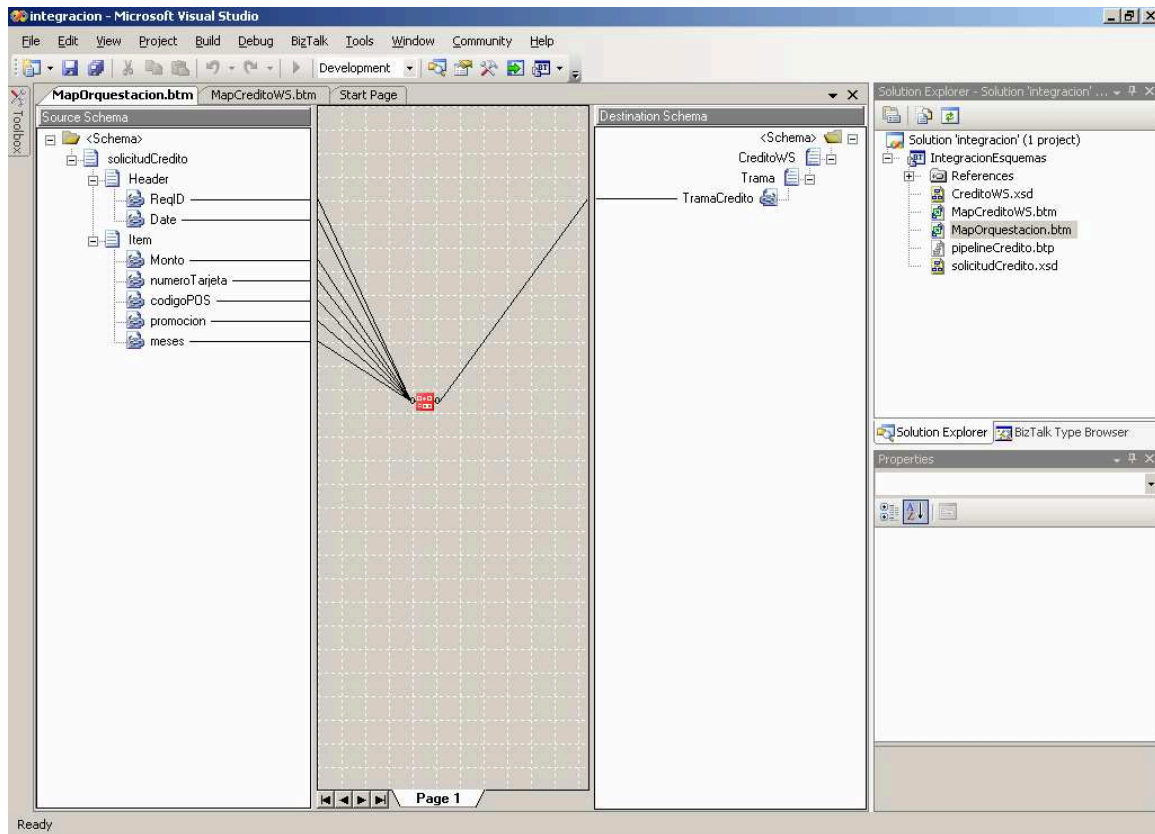
Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.3.4.5.2 Mapas del Sitio.

Para iniciar con la orquestación se debe crear un proyecto de integración de esquemas con la ayuda de Microsoft Visual Studio .Net 2005, se programan los mapas con las características requeridas, estos permiten cambiar el contenido del mensaje entrante a un formato que puede ser procesado por BizTalk. El mapeador de BizTalk programa los formatos tanto de origen como de destino y una vez procesados por el orquestador los transforma y los envía de salida a la otra aplicación. Es decir gracias al mapa del sitio se pueden manejar transar con distintos tipos de datos entre los sistemas. El mapa del sitio de nuestra orquestación es el que se muestra a continuación.

Figura II - 63 Programación del mapa de la orquestación.



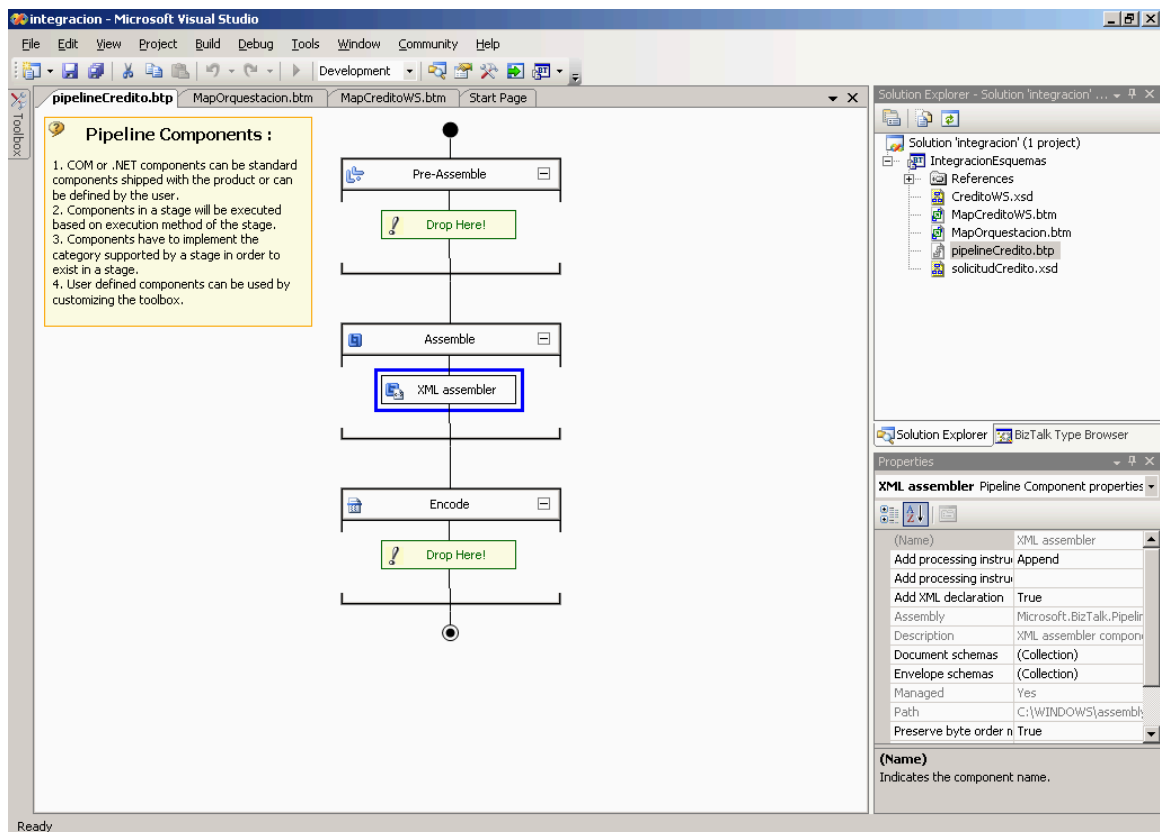
Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.3.4.5.3 Pipelines

Los conductos o pipelines es un software de componentes usados por el proceso de mensajes, este permite enlutar los mensajes, solicitudes, claves, entre otros, por distintas vías dependiendo del tipo al que pertenezcan los datos. Entre las principales utilidades que brindan los conductos es el normalizar los datos tanto de los puertos de entrada como de salida a un formato XML que se lo programa según las necesidades de la integración, permite la denormalización de los formatos XML a varios formatos, ensamblaje y desensamblaje de documentos, codificación y decodificación de archivos, encriptación y desencriptación, asignación y verificación de seguridades y proceso de configuración de los mensajes.

Figura II - 64 Creación de Pipelines en el esquema de integración.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

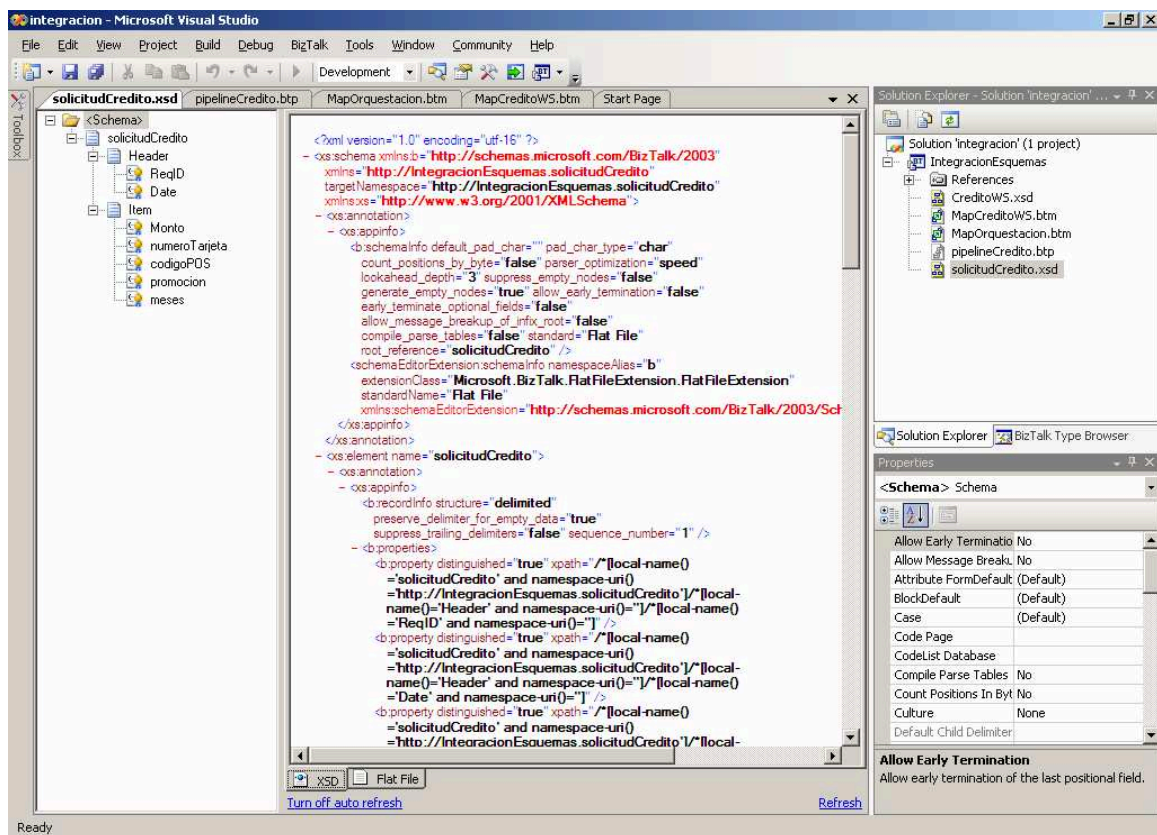
Elaboración: Hidalgo, Karina.

2.3.4.5.4 Esquemas

Los esquemas más conocidos como XSD por sus siglas en inglés Schema Definition Language es una parte importante dentro del proceso de integración pues permite definir la estructura de los mensajes, permite definir el formato de los datos y su procesamiento, los XSD definen los elementos, atributos y tipos de datos que aparecerán en el documento, el orden de las etiquetas en el documento, los campos mandatorios o que se pueden presentar en múltiples ocasiones dentro del mismo archivo, entre otros.

Una vez definidos el mapa y los conductos, podemos diseñar el formato que tendrá los archivos de tipo XML y crear el xsd para el esquema de integración como se muestra en la siguiente pantalla.

Figura II - 65 Diseño de la estructura de los archivos XML.



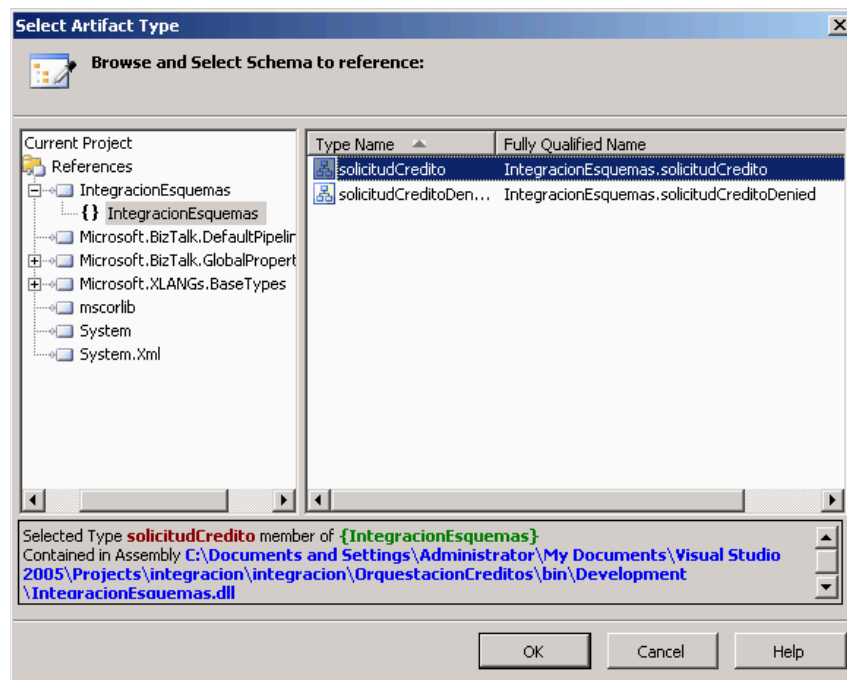
Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.3.4.5.5 Referencias a esquemas creados.

Luego de creados los esquemas se los hace referencia para la orquestación en el proyecto que se deberá elaborar para la lógica del proyecto de integración. Una muestra de estas referencias se aprecia en la siguiente pantalla.

Figura II - 66 Referencias a esquemas previamente diseñados.



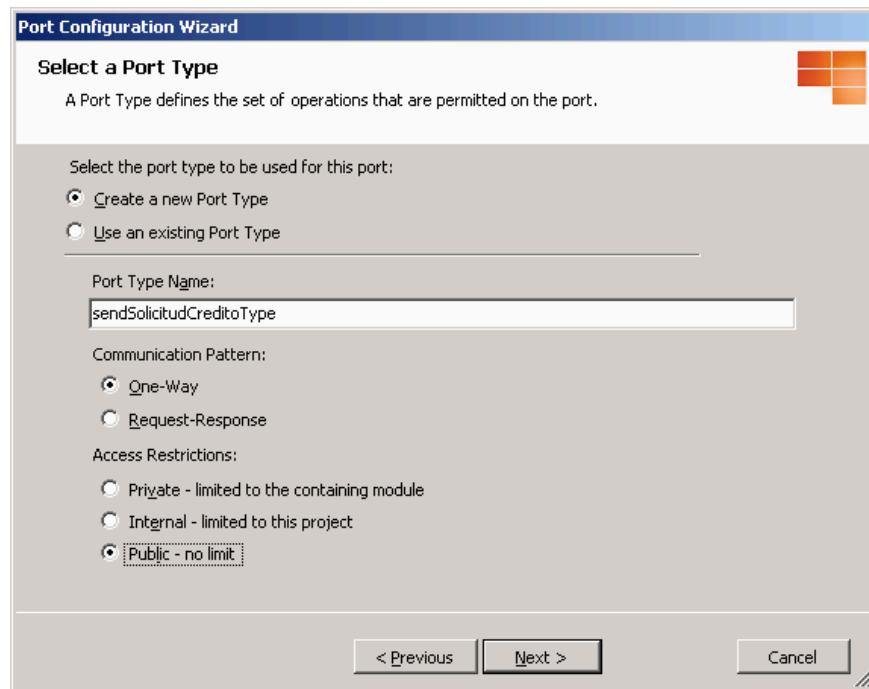
Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.3.4.5.6 Creación y Habilitación de puertos para la orquestación.

La orquestación requiere de puertos para el envío y recepción de la información, estos puertos se los crea y habilita para cada uno de los sistemas y depende del diseño de la integración el número de ellos que se deba utilizar. A continuación se muestra una parte del proceso de creación de puertos.

Figura II - 67 Creación de puerto de envío para la integración.



Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

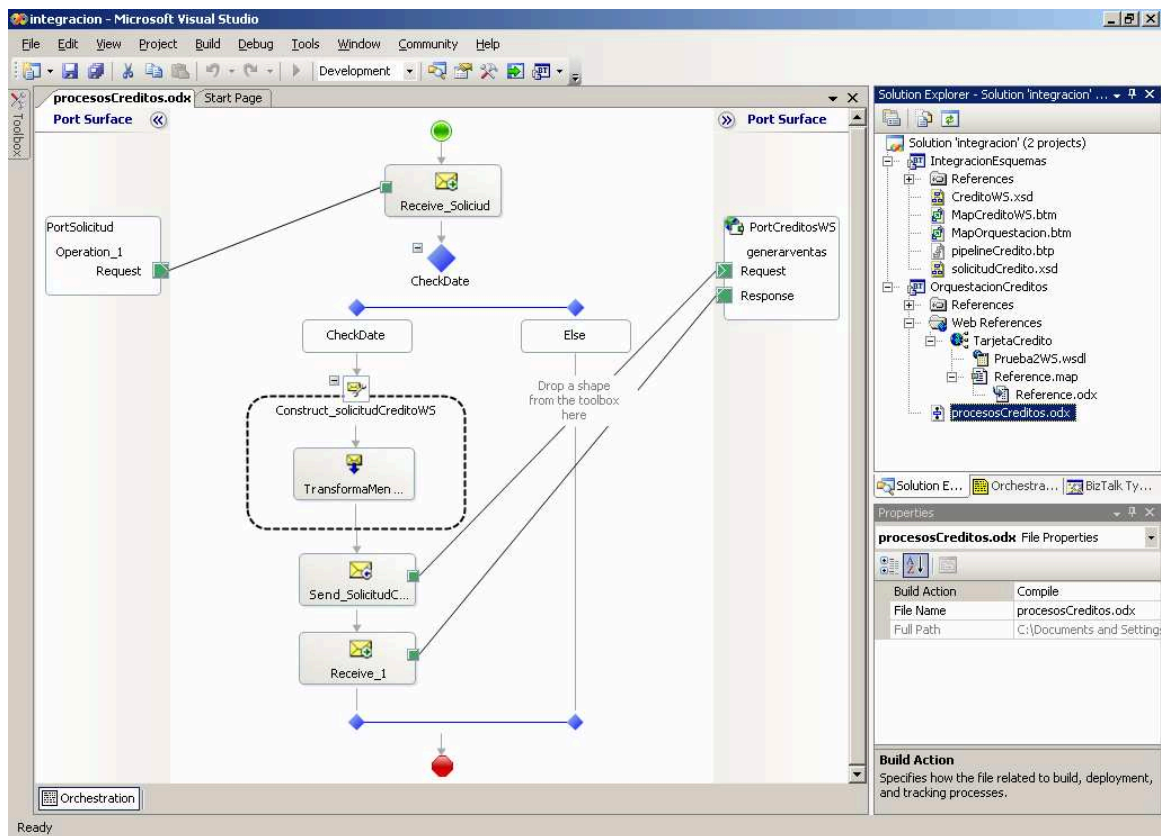
Elaboración: Hidalgo, Karina.

2.3.4.5.7 Orquestación

El orquestar la aplicación es un proceso que se realiza al final de programar los elementos antes mencionados, ya que permite la construcción y ejecución de los procesos de negocio mediante una herramienta llamada diseñador de orquestaciones, ésta permite programar en el lenguaje seleccionado por el desarrollador paso a paso la lógica del negocio y la forma en la que se definirán los procesos de integración.

Para esto se crea un proyecto adicional en Microsoft Visual Studio .Net 2005 para la orquestación, en este proyecto se definen las referencias, los wsdl y los odx como paso final para indicar la lógica de los mensajes y el procesamiento de la información que se llevará a cabo en el integrador, en la siguiente pantalla se muestra el odx de la integración finalizado.

Figura II - 68 Programación del proceso de orquestación.

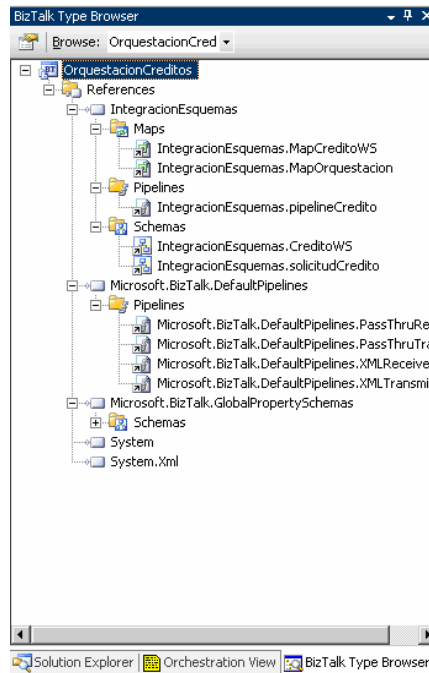


Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.*

Elaboración: Hidalgo, Karina.

Un detalle de los archivos elaborados para la integración de las aplicaciones multiplataforma se puede ver en el explorador de la solución, no se despliegan los esquemas generados por ser solo una muestra del trabajo desempeñado en este proyecto, ya que los esquemas diseñados son numerosos, en la pantalla a continuación se muestra el explorador.

Figura II - 69 Explorador de la Solución de Integración .



Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico*, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4 PRUEBAS Y CONTROL DE CALIDAD

Las pruebas de cada uno de los sistemas desarrollados a lo largo de este proyecto así como la orquestación se realizan empleando casos y procedimientos de prueba, para cada uno de los casos de uso detallados en la fase de Análisis de los sistemas y la integración descritos anteriormente.

A continuación se presentan los casos y procedimientos de prueba para cada uno de los casos de uso definidos, se muestra además el procedimiento seguido para las pruebas de los casos.

2.4.1 CASOS Y PROCEDIMIENTOS DE PRUEBA PARA LOS CASOS DE USO DEL SISTEMA DE FACTURACIÓN DE SUPERMERCADOS.

2.4.1.1 Casos y Procedimientos de prueba para el caso de uso Generar Ventas.

Tabla II - 27 Caso de Prueba para el caso de uso Generar Ventas.

Caso de Prueba	Generar Venta (001)
Rol del Usuario	Cajero
Entrada	<ul style="list-style-type: none"> • El Cajero debe ingresar todos los productos a ser facturados. • El Cajero puede realizar pagos en efectivo o con tarjeta de crédito. • El usuario debe tener el rol de Cajero.
Resultado	<ul style="list-style-type: none"> • Los datos de la venta realizada permanecerán almacenados en la base de datos. • Si el pago con Tarjeta de Crédito es exitoso el cajero deberá aceptar el mensaje para finalizar la transacción.
Condiciones	<ul style="list-style-type: none"> • En caso de que el pago sea realizado con Tarjeta de Crédito, el Cajero debe tener todos los datos de la tarjeta para ingresarlos cuando sea necesario. • El Cajero debe estar autenticado previamente en el sistema.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 28 Procedimiento de prueba para el Caso de prueba Generar Venta (001)

Caso de Prueba: Generar Venta (001)
<ol style="list-style-type: none"> 1. El Cajero selecciona en la interfase Generar Venta. 2. El Cajero ingresa uno a uno los productos a facturar en la interfase. 3. Una vez finalizado el ingreso de productos a facturar. Seleccionar la forma de pago Tarjeta de Crédito. 4. Ingresar los datos de la tarjeta solicitado en la interfase. 5. Dar clic sobre el botón Realizar Pago. 6. Dar clic sobre el botón Aceptar en el mensaje de Pago exitoso enviado por el sistema.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.1.2 Casos y Procedimientos de prueba para el caso de uso Crear Producto.

Tabla II - 29 Caso de Prueba para el caso de uso Crear Producto.

Caso de Prueba	Crear Producto (009) Arroz
Rol del Usuario	Administrador
Entrada	<ul style="list-style-type: none">• El usuario debe ingresar todos los datos solicitados en la interfase.• No debe existir un producto con el mismo código o nombre.• El usuario debe tener el rol de Administrador.
Resultado	<ul style="list-style-type: none">• Los datos del producto permanecen almacenados en la base de datos del sistema.• Se presenta un mensaje de creación exitosa.
Condiciones	<ul style="list-style-type: none">• Debe encontrarse el Administrador validado en el sistema de facturación.• Se debe ingresar todos los datos solicitados por el sistema.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 30 Procedimiento de prueba para el Caso de prueba Crear Producto.

Caso de Prueba: Crear Producto (009) Arroz
<ol style="list-style-type: none">1. Dar clic en la interfase sobre el botón Crear Producto.2. Ingresar los datos del Producto solicitados por la interfase.3. Dar clic sobre el botón Crear.4. Se presenta el mensaje de creación exitosa.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.1.3 Casos y Procedimientos de prueba para el caso de uso Consultar Producto.

Tabla II - 31 Caso de Prueba para el caso de uso Consultar Producto.

Caso de Prueba	Consultar Producto (009) Arroz
Rol del Usuario	Administrador
Entrada	<ul style="list-style-type: none">• El usuario debe ingresar al módulo de administración de Productos.• Debe existir previamente el producto almacenado en la base de datos del sistema.• El usuario debe tener el rol de Administrador.

Resultado	<ul style="list-style-type: none"> • Se presentan los datos del producto consultado.
Condiciones	<ul style="list-style-type: none"> • Debe encontrarse el Administrador validado en el sistema de facturación. • Se debe ingresar el código correcto del producto a consultar.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 32 Procedimiento de prueba para el Caso de prueba Consultar Producto (009) Arroz.

Caso de Prueba: Consultar Producto (009) Arroz
<ol style="list-style-type: none"> 1. Dar clic en la interfase de Administración de producto. 2. Se presenta en la interfase los productos existentes en el sistema. 3. Seleccionar la Producto.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.1.4 Casos y Procedimientos de prueba para el caso de uso Modificar Producto.

Tabla II - 33 Caso de Prueba para el caso de uso Modificar Producto.

Caso de Prueba	Modificar Producto (009) Arroz
Rol del Usuario	Administrador
Entrada	<ul style="list-style-type: none"> • El usuario debe ingresar al Administrador de Tarjetas. • Debe existir previamente la tarjeta en la base de datos del sistema. • El usuario debe tener el rol de Administrador. • Se debe ingresar todos los datos a modificar de la tarjeta seleccionada.
Resultado	<ul style="list-style-type: none"> • Se presentan los datos antiguos de la tarjeta a modificar y listos para editar. • Los nuevos datos de la tarjeta modificada se mantienen en la base de datos.
Condiciones	<ul style="list-style-type: none"> • Debe encontrarse el Administrador validado en el sistema de facturación. • Se debe ingresar los nuevos datos de la tarjeta a modificar.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 34 Procedimiento de prueba para el Caso de prueba Modificar Producto (009) Arroz.

Caso de Prueba: Modificar Producto (009)	
	<ol style="list-style-type: none"> 1. Seleccionar la Tarjeta a Modificar 2. Ingresar los nuevos datos de la Tarjeta. 3. Dar clic sobre el botón Modificar. 4. Aceptar el mensaje de modificación satisfactoria.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.1.5 Casos y Procedimientos de prueba para el caso de uso Eliminar Producto.

Tabla II - 35 Caso de Prueba para el caso de uso Eliminar Producto.

Caso de Prueba	Eliminar Producto (009) Arroz
Rol del Usuario	Administrador
Entrada	<ul style="list-style-type: none"> • El usuario debe ingresar al Administrador de Productos. • Debe existir previamente el producto en la base de datos del sistema. • El usuario debe tener el rol de Administrador. • Se debe seleccionar el producto a eliminar.
Resultado	<ul style="list-style-type: none"> • Se presentan los datos del producto a eliminar. • Se presenta un mensaje con la eliminación exitosa de los datos de la base de datos. • Los datos se eliminan permanentemente de la base de datos.
Condiciones	<ul style="list-style-type: none"> • Debe encontrarse el Administrador validado en el sistema de facturación.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 36 Procedimiento de prueba para el Caso de prueba Eliminar Producto (009) Arroz.

Caso de Prueba: Eliminar Producto (009) Arroz	
	<ol style="list-style-type: none"> 1. Seleccionar el producto a Eliminar. 2. Se debe dar clic sobre el botón Eliminar. 3. Aceptar el mensaje de Eliminación Exitosa de los datos de la base de datos.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.1.6 Casos y Procedimientos de prueba para el caso de uso Reportar Inventario.

Tabla II - 37 Caso de Prueba para el caso de uso Reportar Inventario.

Caso de Prueba	Reportar Inventario
Rol del Usuario	Administrador
Entrada	<ul style="list-style-type: none"> • El usuario debe ingresar al Administrador de Productos. • Deben existir previamente productos en la base de datos del sistema para que se presente en el reporte. • El usuario debe tener el rol de Administrador.
Resultado	<ul style="list-style-type: none"> • Se presentan los resultados de los productos ingresados en el sistema y el stock de los mismos. • Se muestra una pantalla con los resultados.
Condiciones	<ul style="list-style-type: none"> • Debe encontrarse el Administrador validado en el sistema de facturación.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 38 Procedimiento de prueba para el Caso de prueba Reportar Inventario.

Caso de Prueba: Reportar Inventario
<ol style="list-style-type: none"> 1. Seleccionar la opción Reportar Inventario. 2. Observar el reporte del sistema.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.1.7 Casos y Procedimientos de prueba para el caso de uso Deshacer Venta.

Tabla II - 39 Caso de Prueba para el caso de uso Deshacer Venta.

Caso de Prueba	Deshacer Venta (001)
Rol del Usuario	Administrador
Entrada	<ul style="list-style-type: none"> • El usuario debe ingresar al Administrador. • Deben existir previamente la factura creada en el sistema y además debe ser la última factura en haber generado el punto de venta para poder deshacerla (política de la empresa). • El usuario debe tener el rol de Administrador.
Resultado	<ul style="list-style-type: none"> • Se elimina la factura del sistema y los productos ingresan nuevamente al inventario. • Se muestra un mensaje satisfactorio en la

	pantalla.
Condiciones	<ul style="list-style-type: none"> • Debe encontrarse el Administrador validado en el sistema de facturación.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 40 Procedimiento de prueba para el Caso de prueba Deshacer Venta.

Caso de Prueba: Deshacer Venta (001)
<ol style="list-style-type: none"> 1. Buscar la factura en el sistema. 2. Una vez cargados los datos y si cumple con la condición de ser la última factura generada por el punto de venta. 3. Dar clic sobre el botón Eliminar Factura. 4. Se emite un mensaje de satisfacción luego de la eliminación del sistema.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.2 CASOS Y PROCEDIMIENTOS DE PRUEBA PARA LOS CASOS DE USO DEL SISTEMA DE TARJETA DE CREDITO.

2.4.2.1 Casos y Procedimientos de prueba para el caso de uso Validar Tarjeta.

Tabla II - 41 Caso de Prueba para el caso de uso Validar Tarjeta.

Caso de Prueba	Validar la Tarjeta (012) 3606 184800 3646
Rol del Usuario	Cajero
Entrada	<ul style="list-style-type: none"> • Debe existir en la base de datos del sistema la tarjeta de crédito (012) 3606 184800 3646. • El usuario debe tener el rol de Cajero. • El usuario debe haber ingresado en la interfase del sistema en número de la tarjeta.
Resultado	<ul style="list-style-type: none"> • La interfase del cajero envía mediante el integrador los datos de la tarjeta al sistema de Tarjeta de Crédito los datos, allí son validados con la información existente en la base de datos. • Se envía al usuario solicitante el resultado de la verificación de la tarjeta.
Condiciones	<ul style="list-style-type: none"> • Debe encontrarse el cajero autenticado en el sistema de facturación.

	<ul style="list-style-type: none"> • Debe estar iniciado el equipo integrador.
--	---

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 42 Procedimiento de prueba para el Caso de prueba Validar Tarjeta (012) 3606 184800 3646.

Caso de Prueba: Validar Tarjeta (012) 3606 184800 3646
<ol style="list-style-type: none"> 1. Ingresar en la interfase los datos de la tarjeta de crédito. 2. Seleccione el botón Realizar Pago en la interfase. 3. El sistema desplegará en la pantalla un mensaje de aceptación o negación del pago.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.2.2 Casos y Procedimientos de prueba para el caso de uso Solicitar Crédito.

Tabla II - 43 Caso de Prueba para el caso de uso Solicitar Crédito.

Caso de Prueba	Solicitar Crédito de 150 dólares para la tarjeta 3606 184800 3646
Rol del Usuario	Cajero
Entrada	<ul style="list-style-type: none"> • Debe existir la tarjeta en la base de datos del sistema de Tarjeta de Crédito. • El usuario debe tener el rol de Cajero en el sistema. • Se debe colocar los datos de la tarjeta de crédito en el sistema conjuntamente con el monto de la factura a cancelar y el número de meses a diferir.
Resultado	<ul style="list-style-type: none"> • La interfase del cajero envía mediante el integrador los datos de la solicitud de crédito al sistema de Tarjeta de Crédito, allí son validados con la información existente en la base de datos y con el monto disponible para la tarjeta enviada. • Se envía al usuario solicitante el resultado de la transacción con un mensaje de factura cancelada o negado el pago.
Condiciones	<ul style="list-style-type: none"> • Debe encontrarse el cajero autenticado en el sistema de facturación. • Debe estar iniciado el equipo integrador. • Se debe ingresar todos los datos solicitados por el sistema.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 44 Procedimiento de prueba para el Caso de prueba Solicitar Crédito de 150 dólares para la tarjeta 3606 184800 3646.

Caso de Prueba: Solicitar Crédito de 150 dólares para la tarjeta 3606 184800 3646	
	<ol style="list-style-type: none"> 1. Dar clic en la interfase de facturación sobre el botón Pago con Tarjeta de Crédito. 2. Ingresar todos los datos de la tarjeta de crédito y el crédito solicitados en la interfase. 3. Dar clic sobre el botón Realizar Pago. 4. Se presenta un mensaje Pago realizado satisfactoriamente. Dar clic sobre aceptar.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.2.3 Casos y Procedimientos de prueba para el caso de uso Crear Tarjeta.

Tabla II - 45 Caso de Prueba para el caso de uso Crear Tarjeta.

Caso de Prueba	Crear Tarjeta (015) 3606 184800 3647
Rol del Usuario	Administrador
Entrada	<ul style="list-style-type: none"> • El usuario debe ingresar todos los datos solicitados en la interfase. • No debe existir una tarjeta con el mismo número previamente creada. • El usuario debe tener el rol de Administrador.
Resultado	<ul style="list-style-type: none"> • Los datos de la tarjeta permanecen ingresados en la base de datos del sistema. • Se presenta un mensaje de creación exitosa.
Condiciones	<ul style="list-style-type: none"> • Debe encontrarse el Administrador validado en el sistema de facturación. • Se debe ingresar todos los datos solicitados por el sistema.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 46 Procedimiento de prueba para el Caso de prueba Crear Tarjeta.

Caso de Prueba: Crear Tarjeta (015) 3606 184800 3647	
	<ol style="list-style-type: none"> 1. Dar clic en la interfase sobre el botón Crear Tarjeta. 2. Ingresar los datos de la Tarjeta solicitada por la interfase.

3. Dar clic sobre el botón Crear.
4. Se presenta el mensaje de creación exitosa.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.2.4 Casos y Procedimientos de prueba para el caso de uso Consultar Tarjeta.

Tabla II - 47 Caso de Prueba para el caso de uso Consultar Tarjeta.

Caso de Prueba	Consultar Tarjeta (015) 3606 184800 3647
Rol del Usuario	Administrador
Entrada	<ul style="list-style-type: none"> • El usuario debe ingresar al Administrador de Tarjetas. • Debe existir previamente la tarjeta en la base de datos del sistema. • El usuario debe tener el rol de Administrador.
Resultado	<ul style="list-style-type: none"> • Se presentan los datos de la tarjeta consultada.
Condiciones	<ul style="list-style-type: none"> • Debe encontrarse el Administrador validado en el sistema de facturación. • Se debe ingresar el número correcto de la tarjeta a consultar.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 48 Procedimiento de prueba para el Caso de prueba Consultar Tarjeta (015) 3606 184800 3647.

Caso de Prueba: Consultar Tarjeta (015) 3606 184800 3647
<ol style="list-style-type: none"> 1. Dar clic en la interfase de Administración de tarjetas. 2. Se presenta en la interfase las tarjetas existentes en el sistema. 3. Seleccionar la tarjeta.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.2.5 Casos y Procedimientos de prueba para el caso de uso Modificar Tarjeta.

Tabla II - 49 Caso de Prueba para el caso de uso Modificar Tarjeta.

Caso de Prueba	Modificar Tarjeta (015) 3606 184800 3647
Rol del Usuario	Administrador
Entrada	<ul style="list-style-type: none"> • El usuario debe ingresar al Administrador de Tarjetas.

	<ul style="list-style-type: none"> • Debe existir previamente la tarjeta en la base de datos del sistema. • El usuario debe tener el rol de Administrador. • Se debe ingresar todos los datos a modificar de la tarjeta seleccionada.
Resultado	<ul style="list-style-type: none"> • Se presentan los datos antiguos de la tarjeta a modificar y listos para editar. • Los nuevos datos de la tarjeta modificada se mantienen en la base de datos.
Condiciones	<ul style="list-style-type: none"> • Debe encontrarse el Administrador validado en el sistema de facturación. • Se debe ingresar los nuevos datos de la tarjeta a modificar.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 50 Procedimiento de prueba para el Caso de prueba Modificar Tarjeta (015) 3606 184800 3647.

Caso de Prueba: Modificar Tarjeta (015) 3606 184800 3647
<ol style="list-style-type: none"> 1. Seleccionar la Tarjeta a Modificar 2. Ingresar los nuevos datos de la Tarjeta. 3. Dar clic sobre el botón Modificar. 4. Aceptar el mensaje de modificación satisfactoria.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.2.6 Casos y Procedimientos de prueba para el caso de uso Eliminar Tarjeta.

Tabla II - 51 Caso de Prueba para el caso de uso Eliminar Tarjeta.

Caso de Prueba	Eliminar Tarjeta (015) 3606 184800 3647
Rol del Usuario	Administrador
Entrada	<ul style="list-style-type: none"> • El usuario debe ingresar al Administrador de Tarjetas. • Debe existir previamente la tarjeta en la base de datos del sistema. • El usuario debe tener el rol de Administrador. • Se seleccionar la tarjeta a eliminar.
Resultado	<ul style="list-style-type: none"> • Se presentan los datos de la tarjeta a eliminar. • Se presenta un mensaje con la eliminación exitosa de los datos de la base de datos. • Los datos se eliminan permanentemente de la base de datos.

Condiciones	<ul style="list-style-type: none"> • Debe encontrarse el Administrador validado en el sistema de facturación.
--------------------	--

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 52 Procedimiento de prueba para el Caso de prueba Eliminar Tarjeta (015) 3606 184800 3647.

Caso de Prueba: Eliminar Tarjeta (015) 3606 184800 3647
<ol style="list-style-type: none"> 1. Seleccionar la tarjeta a Eliminar. 2. Se debe dar clic sobre el botón Eliminar. 3. Aceptar el mensaje de Eliminación Exitosa de los datos de la base de datos.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.3 RESULTADO DE LAS PRUEBAS PARA EL SISTEMA DE FACTURACIÓN DE SUPERMERCADOS.

Tabla II - 53 Matriz de resultados de las pruebas del sistema de facturación de supermercado.

Proceso Probado	Respuesta del Sistema
Generar venta	EXITOSA
Crear producto	EXITOSA
Consultar producto	EXITOSA
Modificar producto	EXITOSA
Eliminar producto	EXITOSA
Reportar inventario	EXITOSA
Deshacer Venta	EXITOSA

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.4 RESULTADO DE LAS PRUEBAS PARA EL SISTEMA DE TARJETA DE CRÉDITO.

Tabla II - 54 Matriz de resultados de las pruebas del sistema de tarjeta de crédito.

Proceso Probado	Respuesta del Sistema
Validar tarjeta	EXITOSA
Solicitar crédito	EXITOSA

Crear tarjeta	EXITOSA
Consultar tarjeta	EXITOSA
Modificar tarjeta	EXITOSA
Eliminar tarjeta	EXITOSA

Fuente: HIDALGO, Karina, *Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.*

Elaboración: Hidalgo, Karina.

2.4.5 EVALUACIÓN DEL SISTEMA EN UN CASO DE APLICACIÓN.

2.4.5.1 Formulación del caso de aplicación.

2.4.5.1.1 *Objetivos.*

El caso de prueba de la integración de aplicaciones utilizando Microsoft BizTalk Server 2006 debe cumplir con los siguientes objetivos:

- Permitir el desempeño de la integración, es decir del sistema de facturación de supermercados realizando pagos con tarjeta de crédito, en un ambiente lo más acercado a la realidad posible.
- Permitir el desempeño de cada uno de los sistemas implementados por separado, probándolo en un ambiente real.
- Probar tanto los módulos administrativos de cada sistema como los servicios activos que permiten atender solicitudes entrantes.
- Realizar el monitoreo del desempeño de la orquestación y analizar los resultados.

2.4.5.1.2 *Alcance del caso de aplicación.*

El alcance del caso de aplicación del sistema presenta los siguientes puntos.

- El alcance de las pruebas no contempla la instalación del software en ningún supermercado y su funcionamiento en un ambiente real.

- No contempla pruebas con tarjetas de crédito reales.

2.4.5.2 Descripción del caso de aplicación.

El caso de aplicación se constituye de un ejemplo de facturación en un supermercado, es decir, utilizando una factura real y realizando el pago de la misma con una tarjeta de crédito.

Para esto se deberán llenar las bases de datos de los dos sistemas con la información requerida y realizar el ejercicio de facturación.

El caso de aplicación tendrá la simulación de una facturación real, contendrá todos los datos que una factura requiere y su pago mediante tarjeta de crédito, el mismo que deberá ser aprobado por el sistema del socio comercial del supermercado, al número de meses que el cliente solicita, siempre y cuando el monto de crédito solicitado se encuentre dentro del cupo disponible del tarjeta habiente.

2.4.5.3 Construcción

La construcción del caso de aplicación tiene varias etapas. La primera que es la etapa de análisis y diseño de los aplicativos de manera separada, la segunda etapa se culmina con la implementación de los aplicativos incluyendo la integración de los mismos, en la siguiente etapa se realizan todo tipo de pruebas para verificar la fidelidad de los datos y los tiempos de respuesta de los sistemas y la integración, finalmente con el caso de aplicación planteado se procede a llevarlo a cabo con los sistemas en óptimas condiciones.

2.4.5.4 Ejecución de las Pruebas

En la ejecución de las pruebas se ha determinado la necesidad de probar el sistema en siguientes acciones en particular:

- Facturación de productos en stock y pago con tarjeta de crédito.
- Estado de la aprobación del crédito una vez finalizada la factura.
- Estado del Stock de productos al finalizar la factura.
- Cupo disponible en la tarjeta luego de realizada la transacción.

Tabla II - 55 Caso de Prueba para Facturación de productos en stock y pago con tarjeta de crédito.

Caso de Prueba	Facturación de productos en stock y pago con tarjeta de crédito.
Rol del Usuario	Cajero
Entrada	<ul style="list-style-type: none"> • El usuario cajero debe existir en el sistema de facturación. • Debe estar funcionando correctamente el integrador. • El cajero debe ingresar solo productos existentes en la base de datos. • Debe cajero debe enviar los datos correctos y completos de la tarjeta de crédito al integrador.
Resultado	<ul style="list-style-type: none"> • Se realiza una factura con todos los datos requeridos por el sistema. • Se realiza el pago de la factura con tarjeta de crédito. • Se difiere el monto a los meses que desee. • Se eliminan del stock los productos facturados. • Se finaliza exitosamente la venta con el cierre de la factura.
Condiciones	<ul style="list-style-type: none"> • Debe estar autenticado el cajero en el sistema. • Deben existir los productos en stock. • La tarjeta de crédito debe tener un cupo que abarque el monto de la factura.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 56 Procedimiento de prueba para el Caso de Prueba para Facturación de productos en stock y pago con tarjeta de crédito.

Caso de Prueba: Facturación de productos en stock y pago con tarjeta de crédito.
<ol style="list-style-type: none"> 1. El usuario con su cuenta de cajero ingresa al sistema. 2. Selecciona la opción generar Factura. 3. Ingresa los códigos de cada uno de los productos a facturar y su cantidad. 4. Selecciona la opción Calcular total de factura.

5. Selecciona la opción pago con Tarjeta de Crédito.
6. Ingresa los datos de la tarjeta solicitados por el sistema.
7. Una vez aprobado el crédito finaliza la factura y entrega los productos al cliente.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 57 Caso de Prueba para Facturación Estado de la aprobación del crédito una vez finalizada la factura.

Caso de Prueba	Estado de la aprobación del crédito una vez finalizada la factura
Rol del Usuario	Administrador
Entrada	<ul style="list-style-type: none"> • El usuario administrador del sistema de tarjetas de crédito debe conocer el modelo entidad relación de la base de datos. • El crédito debe haberse solicitado y aprobado, ya que si fue negado simplemente no se registra. • El administrador debe hacer la consulta a la base de datos.
Resultado	<ul style="list-style-type: none"> • Si el crédito fue aprobado se muestra en los resultados de la consulta, con todos los datos del mismo.
Condiciones	<ul style="list-style-type: none"> • Debe estar autenticado el administrador. • Debe haberse solicitado el crédito y aprobado para que se registre.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 58 Procedimiento de prueba para el Caso de Prueba para Estado de la aprobación del crédito una vez finalizada la factura.

Caso de Prueba: Estado de la aprobación del crédito una vez finalizada la factura.

1. El usuario administrador se loguea en el sistema de tarjetas de crédito.
2. Realiza la consulta del cliente con sus créditos aprobados.
3. Visualiza el resultado del estado del crédito solicitado.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 59 Caso de Prueba para Estado del Stock de productos al finalizar la factura.

Caso de Prueba	Estado del Stock de productos al finalizar la factura.
Rol del Usuario	Administrador
Entrada	<ul style="list-style-type: none"> • El usuario administrador del sistema de facturación de supermercados debe solicitar un reporte de inventario. • El usuario administrador debe tener ligado los permisos para realizar este reporte.
Resultado	<ul style="list-style-type: none"> • Se muestra el reporte del inventario, consta el stock de cada uno de los productos existentes de en la base de datos. • Se comparará manualmente que el resultado del débito del stock de productos sea igual al número de productos facturados.
Condiciones	<ul style="list-style-type: none"> • Debe estar autenticado el administrador.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 60 Procedimiento de prueba para el Caso de Prueba para Estado del Stock de productos al finalizar la factura.

Caso de Prueba: Estado del Stock de productos al finalizar la factura.
<ol style="list-style-type: none"> 1. El usuario administrador se loguea en el sistema de tarjetas de crédito. 2. Solicita al sistema Reporte de Inventario. 3. Realiza el proceso de facturación completamente. 4. Solicita al sistema Reporte de Inventario. 5. Compara que el débito de productos del inventario sea el correcto.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 61 Caso de Prueba para Cupo disponible en la tarjeta luego de realizada la transacción.

Caso de Prueba	Cupo disponible en la tarjeta luego de realizada la transacción.
Rol del Usuario	Administrador
Entrada	<ul style="list-style-type: none"> • El usuario administrador del sistema de tarjetas de crédito conocer el diagrama entidad relación de la base de datos. • El usuario debe verificar manualmente el estado de una cuenta antes y luego de solicitado un crédito.
Resultado	<ul style="list-style-type: none"> • Se presenta el resultado de la consulta.

	<ul style="list-style-type: none"> • Se verifica manualmente que el cupo de la cuenta es el correcto.
Condiciones	<ul style="list-style-type: none"> • Debe estar autenticado el administrador.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

Tabla II - 62 Procedimiento de prueba para el Caso de Prueba para Cupo disponible en la tarjeta luego de realizada la transacción.

Caso de Prueba: Cupo disponible en la tarjeta luego de realizada la transacción.
<ol style="list-style-type: none"> 1. El usuario administrador se loguea en el sistema de tarjetas de crédito. 2. Solicita al sistema el cupo de un determinado cliente. 3. Realiza el proceso de solicitud de un crédito. 4. Se aprueba el crédito. 5. Solicita al sistema el cupo de un determinado cliente. 6. Compara que el cupo disponible luego de la aprobación del crédito es el correcto.

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

2.4.5.5 Análisis de los Resultados De las Pruebas del Caso de Aplicación.

Tabla II - 63 Matriz de resultados de las pruebas del caso de Aplicación.

Proceso Probado	Respuesta del Sistema
Facturación de productos en stock y pago con tarjeta de crédito	EXITOSA
Estado de la aprobación del crédito una vez finalizada la factura	EXITOSA
Estado del Stock de productos al Cupo disponible en la tarjeta luego de realizada la transacción	EXITOSA

Fuente: HIDALGO, Karina, Aplicación de la plataforma BizTalk 2006 en el desarrollo de software. Caso práctico, Quito, 2007.

Elaboración: Hidalgo, Karina.

De acuerdo con la matriz de los resultados de la interacción del sistema con los clientes y de realizado el caso de aplicación ha sido exitosa.

Por el éxito de las pruebas se puede asegurar una correcta interacción de cada uno de los sistemas con los usuarios finales, así con la exitosa integración de los sistemas totalmente independientes multiplataforma a un unísono que permite mejorar la calidad de la atención al cliente y mejores lazos con socios comerciales, o al interior de la misma empresa.

De acuerdo con los resultados favorables que se ha obtenido a lo largo de este proceso, se puede afirmar que tanto la integración de los sistemas como los sistemas independientes, es óptimo y satisface los requerimientos y alcances plateados al inicio de este proyecto.

Ha sido una misión cumplida exitosamente, pues mediante la orquestación de aplicaciones creadas de manera aislada y multiplataformas se ha podido unir socios comerciales que realizan sus facturas y permiten pagos con tarjetas de crédito, mejorando la atención al cliente, con técnicas modernas de arquitecturas de datos e integración de sistemas.

CAPITULO 3

CONCLUSIONES Y RECOMENDACIONES

Al finalizar el presente proyecto es posible la realización de las siguientes conclusiones y recomendaciones.

3.1 CONCLUSIONES

- El estudio de la plataforma BizTalk Server 2006 y la aplicación en el desarrollo de software ha concluido de manera exitosa, cumpliendo a cabalidad los objetivos planteados inicialmente y de acuerdo con el alcance establecido.
- Durante el proyecto se ha realizado un estudio completo de la arquitectura de las soluciones BizTalk Server 2006, lo cual permitió que la implementación del caso de estudio se realice empleando las funcionalidades más sobresalientes de la plataforma mencionada.
- Como metodología de desarrollo de software se empleó Rational Unified Process, la misma que debido a sus características de un manejo iterativo e incremental del proceso de desarrollo, permitió que el proyecto sea completado satisfactoriamente.
- La aplicación de un caso de estudio constituye una verdadera tarea de investigación, en la cual han podido ser plasmadas en sistemas de software todas las ventajas de la plataforma BizTalk Server 2006.
- Se puede concluir luego de la realización exitosa del proyecto que el estudio de la arquitectura del sistema multiplataforma BizTalk Server 2006 es una plataforma de gran alcance, que efectiviza la integración de

sistemas independientemente de su lenguaje de programación o plataforma de origen, poniendo a disposición grandes mejoras respecto a sus versiones predecesoras.

- Para la integración de aplicaciones multiplataforma a lo largo de este proyecto se han implementado mediante un proceso formal de desarrollo dos aplicaciones distintas. Este proceso de análisis, diseño e implementación de cada uno de los sistemas en distintas plataformas facilitó el análisis detallado de los procesos de integración que se deben llevar a cabo.
- Dentro de las empresas de nuestro medio, la utilización de BizTalk Server puede ser de gran utilidad, para integrar sistemas nuevos, programados en tecnologías actuales, con los sistemas antiguos de la organización, lo cual permite ahorrar los gastos concernientes al reemplazo de los sistemas tradicionales de una compañía.
- La implementación de Biztalk Server 2006 también puede constituir para una empresa de nuestro medio, un paso muy importante en su camino hacia una organización de Sistemas basada en SOA. La integración de software, sobre todo mediante WebServices, es una alternativa altamente útil para encaminar a una organización hacia una arquitectura orientada a servicios.
- El producto resultante del presente proyecto es la muestra de un estudio que claramente deja notar las ventajas y desventajas de usar Microsoft BizTalk Server 2006 como orquestador de aplicaciones multiplataforma, así como la mejora en la productividad de las empresas permitiendo monitorear constantemente la integración.
- El desarrollo de este tipo de integraciones en el Ecuador es muy limitado por el tiempo que toma el implementarlo, el no hacerlo resta productividad

a las empresas, con este proyecto se ha demostrado que el integrador además de ser eficiente, facilita las tareas de administración y adhesión de una mayor cantidad de sistemas a la orquestación, esto significa lasos con nuevos socios comerciarles, elevando el nivel de ganancias.

- Este proyecto por ser totalmente distribuido no tiene limitaciones para su implementación en empresas que manejan un flujo notable de datos. La integración y monitoreo de aplicaciones realizada a lo largo de este proyecto permite destacar las ventajas de la distribución de los sistemas y del integrador, manejando los datos de manera segura.

3.2 RECOMENDACIONES

- Para el desarrollo de una integración de sistemas empleando BizTalk Server 2006 es recomendable un estudio previo de la plataforma BizTalk, lo cual permitirá que se aprovechen posteriormente todas las ventajas de éstas herramientas a favor del proyecto.
- Antes de realizar una integración de sistemas es necesario definir claramente la visión del sistema resultante, la cual debe estar enmarcada dentro de los planes estratégicos de la organización.
- Hoy en día la competencia entre las diferentes empresas es cada vez más intensa, por lo cual se hace necesario aprovechar al máximo los recursos disponibles. Dentro de ese contexto, es primordial que una herramienta como BizTalk Server 2006 sea estudiada a fondo para emplear eficientemente sus potencialidades a favor del negocio.
- Es recomendable para llevar a cabo la implementación de sistemas independientemente de su lenguaje de programación o plataforma, el elaborarlo mediante un proceso formar de desarrollo de software, esto permitirá además de tener una documentación detallada y estandarizada

de los sistemas, obtener diseños de la arquitectura del sistema expansibles, que soporten grandes flujos de datos y permitan la integración con sistemas que coinciden en un determinado momento en su lógica del negocio.

- Además de tener las herramientas adecuadas para la orquestación de aplicaciones disponibles, es necesario tener información de calidad sobre el manejo, novedades y administración eficiente de las mismas, esto permitirá además de una gestión eficiente un ahorro notable en tiempo y esfuerzo para el desarrollo sobre el integrador.
- En el caso de las dos aplicaciones multiplataformas desarrolladas a lo largo de este proyecto, por ser distribuidas y basadas en componentes es posible el ampliarlas, no solo integrándoles a con otras aplicaciones, sino que se pueden construir módulos adicionales para cada una de ellas, por lo que es recomendable la utilización de servicios web además de arquitecturas SOA.

3.3 REFLEXION FINAL

El desarrollo de este proyecto de software y su exitosa culminación me han ayudado a aprender que la perseverancia es un factor importante para lograr las metas que uno se propone, he aprendido que la entereza y esfuerzo que se dedica a un proyecto son directamente proporcionales a los resultados que se obtienen al final del camino.

Aunque el estudio de productos nuevos con poco auge en el mercado son difíciles de realizar, por la escasa información que especialmente en nuestro medio se maneja sobre tecnologías de punta, vale la pena incursionar en campos que se abren mercados de a poco en países como el Ecuador, es verdad que el esfuerzo y tiempo dedicados es mayor, pero gracias al impulso que la Escuela Politécnica Nacional brinda a proyectos de este tipo.

Para finalizar, este proyecto no hubiese culminado exitosamente de no ser por la ayuda de Dios y de todas las personas que me han apoyado de distintas maneras hasta llegar a este punto.

BIBLIOGRAFIA

- BOOCH / RUMBAUGH / JACOBSON, “El lenguaje unificado de modelado – Guía de usuario”, Addison Wesley, 2000.
- BOOCH / RUMBAUGH / JACOBSON, “El proceso unificado de desarrollo de software”, Addison Wesley, 2000.
- MICROSOFT, “Deploying and Managing Business Process and Integration Solutions Using Microsoft BizTalk Server 2006”, 2006.
- MICROSOFT, “Developing Business Process and Integration Solutions Using Microsoft BizTalk Server 2006”, 2006.
- MICROSOFT, “Windows Server System, BizTalk Server 2006”, 2006.
- MICROSOFT Corporation, Microsoft Visual Studio 2005, <http://msdn.microsoft.com/vstudio>, 2006.
- MICROSOFT Corporation, Microsoft Visual Studio, <http://msdn.microsoft.com/sqlserver>, 2006.
- WIKIPEDIA, Glosario, <http://es.wikipedia.org/wiki/>, 2006.
- RÍO ÁLVAREZ Jesús, “Servicios de Proceso de Imágenes, XML y publicación web.pdf”, 2005.
- MICROSOFT Corporation, Apéndice C: Guía de entrega, <https://www.microsoft.com/latam/technet/articulos/wireless/ap03.aspx>, 2004.
- DIAZ TOLEDANO Moisés, “Web Services. Introducción y Escenario para su Uso”, 2005.
- MICROSOFT Corporation, Arquitectura, <http://download.microsoft.com/download/a/9/5/a95b8caa-c2c7-439f-948d-2c6c14f5845c/ArquitecturaBTS.ppt>, 2006.
- INFORMATIZATE, Arquitecturas, http://www.informatizate.net/articulo/metodologias_de_desarrollo_de_software_07062004.html, 2005.
- MICROSOFT Corporation, “what is”, <http://www.microsoft.com/spain/biztalk/2006/prodinfo/what-is-biztalk-server.aspx>, 2006.

ANEXO I

CONTENIDO DEL CD ADJUNTO

El CD adjunto contiene toda la documentación que respalda este proyecto en forma de anexos, el código fuente de los programas, instaladores. Estructurados de la siguiente manera:

CD ADJUNTO

Anexos

- Anexo II. Características de BizTalk.
- Anexo III. Especificación de Requerimientos.
- Anexo IV. Diagramas de Secuencia.
- Anexo V. Diagramas de Colaboración de análisis.
- Anexo VI. Diagramas de Colaboración de diseño.
- Anexo VII. Diagramas de Actividad.
- Anexo VIII. Diagramas de Paquetes y Sub paquetes.
- Anexo IX. Diseño de Interfases.
- Anexo X. Casos de Prueba.

Tesis

- Capítulo I.
- Capítulo II.
- Capítulo III.

Instaladores y Códigos Fuente

- Sistema de Facturación de Supermercados.
- Sistema de Tarjetas de Crédito.
- Archivo de orquestación.