

UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMATICA



PROYECTO DE GRADO

**“SISTEMA WEB DE COMERCIO ELECTRONICO Y CONTROL DE
INVENTARIO
CASO: FIBBY’S S.R.L.”**

PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMATICA
MENCION: INGENIERIA DE SISTEMAS INFORMATICOS

POSTULANTE: Israel Gabriel Bernabé Condorenz
TUTOR METODOLOGICO: Lic. Freddy Miguel Toledo Paz
ASESOR: Lic. Jhonny Roberto Felipez Andrade

LA PAZ – BOLIVIA
2015



**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA**



LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON EstrictAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

- a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.

Dedicatoria

*A mis padres Hilda y José, por haberme dado la oportunidad de estudiar, a Dios, por darme la sabiduría y fuerza necesaria para aprovechar esta oportunidad
A mi compañera de vida por apoyarme tanto...*

Gracias.

AGRADECIMIENTOS

A Dios por siempre guiarme en el camino de la vida, por las lecciones, oportunidades y todo lo grandioso de la vida.

A mi mamá Hilda Condorenz Sanchez por darme la vida, por estar siempre conmigo apoyándome en todo momento.

A mi papá José Bernabé por encaminarme en la vocación de mi vida.

A mi abuela Justina Sánchez, por enseñarme la fortaleza que debe tener una persona.

A mis hermanos, Abraham y Moisés por el apoyo que siempre me brindaron.

A mis Tías, por toda la guía y apoyo brindando en esta etapa.

A la empresa FIBBY'S S.R.L., por la oportunidad de realizar este proyecto.

A mi tutor Lic. Freddy Miguel Toledo Paz por la guía en esta última etapa de la carrera.

Al mi asesor Lic. Jhonny Roberto Felipez Andrade, por la guía, conocimiento y apoyo brindado en el transcurso de esta etapa de mi vida.

A los docentes de la carrera por la formación que nos brindan...

Muchas Gracias.

RESUMEN

El presente proyecto de grado titulado “SISTEMA WEB DE COMERCIO ELECTRONICO Y CONTROL DE INVENTARIO CASO: FIBBY’S S.R.L.” parte de la necesidad de ampliar los medios de venta de la empresa por medios más actuales.

Cuenta con un catalogo en línea, por el cual se puede realizar compras y pagos por TIGO MONEY y cuentas bancarias.

Para el análisis y diseño se emplearon las metodologías (AUP) Proceso Unificado Ágil que consta de cuatro fases, además se uso en el diseño la metodología UWE, un método de ingeniería basado en diagramas UML (Unified Modeling Language).

El desarrollo del Sistema de Comercio Electrónico se lo realizo en lenguaje de programación PHP y como gestor de Base de Datos MySQL.

La calidad del Sistema se lo realizó bajo el **WEB SITE QEM** este evalúa aspectos como la Usabilidad, Funcionalidad, Confiabilidad y Eficiencia, proporcionando una evaluación tras la implementación del Sistema.

Utiliza el framework Bootstrap por el cual se implementa el responsive web design.

El análisis sobre Costo se lo realizó tras una primera evaluación sobre COCOMO (Modelo Constructivo de Costes), que posteriormente se realizó el cálculo del VAN (Valor Actual neto), TIR (Tasa Interna de Retorno) para así conocer el Costo del Sistema.

Palabras Claves: Comercio Electrónico, Metodología AUP, Herramienta de modelado UWE, WEB SITE QEM, Cocomo.

Contenido

CAPITULO I

Marco Referencial

1.1	Introducción	11
1.2	Antecedentes	12
1.2.1	Trabajos realizados en otras universidades	12
1.2.2	Trabajos realizados en la carrera de informática.....	12
1.3	Problemas	13
1.3.1	Problemas Secundarios.....	13
1.3.2	Problema General	13
1.4	Objetivos	14
1.4.1	Objetivo General	14
1.4.2	Objetivos Específicos.....	14
1.5	Justificación.....	14
1.5.1	Justificación Económica	14
1.5.2	Justificación Social.....	14
1.5.3	Justificación Técnica.....	15
1.6	Alcances	15
1.7	Límites	16
1.8	Metodología.....	16
1.8.1	Metodología de Investigación.....	16

CAPITULO II

Marco Teórico

2.1	Marco Institucional	20
2.1.1	Misión	20
2.1.2	Visión.....	20
2.2	Metodología.....	20
2.2.1	El Proceso Unificado Agil – Aup	21
2.2.2	Modelado	32
2.2.3	Uwe	32
2.3	Comercio Electrónico	44
2.4	Inventarios	46
2.4.1	Sistemas de inventario.....	46
2.5	Tecnología de software.....	47
2.5.1	Php	47
2.5.2	JavaScript	49
2.5.3	Jquery.....	50
2.5.4	Html.....	50
2.5.5	Html5.....	52
2.5.6	Css	52
2.5.7	Responsive Web Design	53
2.5.8	MySQL	53
2.6	Calidad de software	54
2.6.1	Calidad de Software WEB SITE QEM	54
2.7	Estudio de costos y beneficios	56
2.7.1	Cocomo	56

CAPITULO III

Marco Aplicativo

3.1	Introducción	57
3.2	Situación actual de la empresa	58
3.3	Fase de Inicio.....	58
3.3.1	Disciplina de Modelado.....	58
3.3.1.1	Definición de Actores	59
3.3.2	Disciplina de Implementación.....	59
3.3.2.1	Modelo del negocio	60
3.3.3	Disciplina de Pruebas	60
3.3.4	Disciplina de Despliegue	60
3.3.5	Disciplina de Administración de la Configuración.....	60
3.3.6	Disciplina de Administración de Proyecto	60
3.3.7	Disciplina de Entorno	61
3.4	Fase de Elaboración	61
3.4.1	Disciplina de Modelado.....	61
3.4.1.1	Diagramas de Casos de Uso	61
3.4.1.2	Descripción de Los casos de Uso de Alto Nivel	61
3.4.1.2.1	Caso de Uso: Acceso al sistema	61
3.4.1.2.2	Caso de Uso: Registro de pedido	62
3.4.1.2.3	Caso de Uso: Registro de cliente minorista al sistema	63
3.4.1.2.4	Caso de Uso: Cotización de confección.....	63
3.4.1.2.5	Caso de Uso: Validación de pago por venta en pagina web	64
3.4.1.2.6	Caso de Uso: Venta en sucursal	64
3.4.1.2.7	Caso de Uso: Registro de productos	65
3.4.1.2.8	Caso de Uso: Registro de Ingresos de productos a Almacén	65
3.4.1.2.9	Caso de Uso: Registro usuarios del sistema.....	66
3.4.1.3	Diagramas de Casos de Uso Expandido	66
3.4.1.3.1	Diagrama de Caso de Uso Expandido: Registro de pedido	66
3.4.1.3.2	Diagrama de Caso de Uso Expandido: Registro de cliente minorista al sistema	67
3.4.1.3.3	Diagrama de Caso de Uso Expandido: Cotización de confección	68
3.4.1.3.4	Diagrama de Caso de Uso Expandido: Validación de pago por pedido en pagina web	68
3.4.1.3.5	Diagrama de Caso de Uso Expandido: Venta en sucursal	69
3.4.1.3.6	Diagrama de Caso de Uso Expandido: Registro de productos.....	70
3.4.1.3.7	Diagrama de Caso de Uso Expandido: Registro de ingresos y movimientos de mercadería	70
3.4.1.3.8	Diagrama de Caso de Uso Expandido: Registro usuarios del sistema	71
3.4.2	Disciplina de Implementación.....	72
3.4.2.1	Diagrama de Clases	72
3.4.2.2	Modelo de Procesos.....	73
3.4.2.2.1	Diagrama de estructura del proceso.....	73
3.4.2.2.2	Diagrama de Flujo de Procesos	73
3.5	Fase de Construcción	81
3.5.1	Disciplina de Modelado.....	81
3.5.1.1	Diagrama de Navegación	81
3.5.1.1.1	Diagramas de navegación cliente mayorista o minorista	81

3.5.1.1.2	Diagramas de navegación: Encargado de ventas.....	82
3.5.1.1.3	Diagramas de navegación: Encargado de almacén.....	82
3.5.1.1.4	Diagramas de navegación: Administrador	83
3.5.2	Disciplina de Implementación.....	84
3.5.2.1	Modelo Entidad Relación	84
3.5.2.2	Diccionario de Datos	84
3.5.3	Disciplina de Despliegue	89
3.5.3.1	Diagrama de Presentación	89
3.5.3.1.1	Diagrama de Presentación: Pantalla de Logueo	89
3.5.3.1.2	Diagrama de Presentación: Cliente Mayorista y minorista	91
3.5.3.1.3	Diagrama de Presentación: Encargado de Ventas	93
3.5.3.1.4	Diagrama de Presentación: Encargado de Almacén	95
3.5.3.1.5	Diagrama de Presentación: Administrador	97
3.6	Fase de Transición.....	99
3.6.1	Disciplina de Modelado.....	99
3.6.2	Disciplina de Implementación.....	99
3.6.3	Disciplina de Despliegue	99

CAPITULO IV

Calidad, Seguridad y Costos

4.1.	Calidad del software	100
4.1.1.	Evaluación elemental WEB SITE QEM.....	100
4.1.2.	Tipos de Criterios de Preferencia de Calidad Elemental	100
4.1.3.	Evaluación elemental de calidad	101
4.2.	Seguridad del software	107
4.2.1.	Seguridad del Sistema.....	107
4.2.1.1.	Inyeccion Sql	107
4.2.1.2.	Uso indebido: Subida de archivos.....	108
4.2.2.	Seguridad de Tigo Money	108
4.3.	Evaluación de costos.....	108
4.3.1.	Cocomo	109

CAPITULO V

Conclusiones y Recomendaciones

5.1	Conclusiones	117
5.2	Recomendaciones.....	118

Índice de Figuras

Figura 1. Fase y Disciplinas del AUP.....	23
Figura 2. Modelo de Liberación de Producción.....	24
Figura 3. Diagrama de caso de uso en UWE	36
Figura 4. Diagrama de actividad de UWE	38
Figura 5. Diagrama de contenido en UWE	38
Figura 6. Diagrama de navegación en UWE	40
Figura 7. Diagrama de presentación en UWE.....	41
Figura 8. Diagrama de clases de proceso	42
Figura 9. Diagrama de flujo de proceso de UWE	44
Figura 10. Registro de pedido.....	67
Figura 11. Registro de cliente minorista al sistema	67
Figura 12. Cotización de confección.....	68
Figura 13. Validación de pago por pedido en página web	69
Figura 14. Venta en sucursal	69
Figura 15. Registro de productos	70
Figura 16. Registro de ingresos y movimientos de mercadería	71
Figura 17. Registro usuarios del sistema.....	71
Figura 18. Diagrama de Clases del Sistema	72
Figura 19. Diagrama de estructura del proceso.....	73
Figura 20. Logueo de Usuario y Cliente.....	74
Figura 21. Registro de Pedido	75
Figura 22. Registro de cotización	76
Figura 23. Registro de venta en sucursal.....	77
Figura 24. Validación de pedido	78
Figura 25. Registro de producto	79
Figura 26. Registro de movimiento de mercadería.....	80
Figura 27. Registro de usuario.....	81
Figura 28. Diagramas de navegación cliente mayorista o minorista	82
Figura 29. Encargado de ventas.....	82
Figura 30. Encargado de almacén	83
Figura 31. Administrador.....	83
Figura 32. Modelo entidad relación.....	84
Figura 33. Pantalla de logueo	90
Figura 34. Acceso de usuario	90
Figura 35. Acceso y registro de clientes.....	91
Figura 36. Cliente mayorista y minorista	91
Figura 37. Catalogo del producto	92
Figura 38. Detalles del producto.....	92
Figura 39. Cesta de productos.....	93
Figura 40. Encargado de ventas.....	93

Figura 41. Menú encargado de ventas.....	94
Figura 42. Formulario de registro de venta.....	94
Figura 43. Encargado de almacén	95
Figura 44. Menú encargado de almacén.....	96
Figura 45. Formulario de registro de mercadería	96
Figura 46. Accesos del administrador.....	97
Figura 47. Menú del administrador.....	98
Figura 48. Reporte de productos vendidos.....	98
Figura 49. Tipos de criterios de preferencia de calidad elemental.....	100
Figura 50. Escala de preferencia.....	103
Figura 51. Tipo de dato de recolección	104
Figura 52. Resultados obtenidos de la evaluación	107
Figura 53. Resumen de los valores obtenidos	107

Índice de Tablas

Tabla 1. Disciplinas y Principales Actividades de la fase de comienzo.....	26
Tabla 2 Disciplinas y Principales Actividades de la fase de elaboración	27
Tabla 3. Disciplinas y Principales Actividades de la fase de construcción	29
Tabla 4. Disciplinas y Principales Actividades de la fase de Transición	30
Tabla 5. Fases del AUP	58
Tabla 6. Definición de actores.....	59
Tabla 7. Caso de uso de alto nivel	62
Tabla 8. Registro de pedido	62
Tabla 9. Registro de cliente minorista al sistema.....	63
Tabla 10. Cotización de Confección	64
Tabla 11. Validación de pago por venta en página web	64
Tabla 12. Venta en sucursal.....	65
Tabla 13. Registro de productos.....	65
Tabla 14. Registro de ingresos de productos a almacén	66
Tabla 15. Registro usuarios del sistema	66
Tabla 16. Diccionario de datos cliente.....	85
Tabla 17. Diccionario de datos color	85
Tabla 18. Diccionario de datos inventario	85
Tabla 19. Diccionario de datos marca	85
Tabla 20. Diccionario de datos movimiento de mercadería	86
Tabla 21. Diccionario de datos pedido.....	86
Tabla 22. Diccionario de datos pedido producto	86
Tabla 23. Diccionario de datos producto	87
Tabla 24. Diccionario de datos producto color	87
Tabla 25. Diccionario de datos producto imagen.....	87
Tabla 26. Diccionario de datos producto subtipo	87
Tabla 27. Diccionario de datos producto tipo	88
Tabla 28. Diccionario de datos producto tipo cliente.....	88
Tabla 29. Diccionario de datos rol	88
Tabla 30. Diccionario de datos Sucursal.....	88
Tabla 31. Diccionario de datos Tipo cliente	88
Tabla 32. Diccionario de datos usuario	89
Tabla 33. Diccionario de datos venta.....	89
Tabla 34. Diccionario venta producto.....	89
Tabla 35. Evaluación elemental de calidad.....	103
Tabla 36. Tabla de valores de proyectos de software.....	110
Tabla 37. Costo de desarrollo del software.....	111
Tabla 38. Costo de elaboración del proyecto	115
Tabla 39. Costo total del software	115

1.1 Introducción

Debido al constante avance de la tecnología el mundo está más interconectado; especialmente con las tecnologías de información y comunicación. Y se ve más potenciados con la aparición de nuevos dispositivos que apoyan a la portabilidad (laptops, smarphone's, tabletas) con los cuales se puede acceder a la web desde cualquier punto.

Pero con la aparición de estos nuevos dispositivos de diferentes dimensiones y resoluciones, llega el problema de mostrar el contenido diseñado para una sola resolución lo que hace incomoda la experiencia desde dispositivos de diferentes resoluciones (ya sean mucho mayores o menores). El responsive web design (diseño web adaptativo) es un nuevo método de diseño de páginas web; con el cual se puede mostrar la información con un diseño para cada dispositivo diferente sin la necesidad de utilizar lenguajes de programación, diferentes hojas de estilo para cada dispositivo o tener que diseñar varias veces el mismo contenido; trata de la reutilización de código css.

El comercio data desde los inicios de la humanidad, pero con esta se van desarrollando nuevos modos de hacer comercio; El comercio electrónico consiste en realizar electrónicamente transacciones comerciales de cualquier tipo con la ayuda de las tics; permitiendo adquirir bienes y servicio, inclusive contenidos digitales, actualmente el comercio electrónico va pasando a ser una necesidad para potenciar a las empresas de cualquier tamaño.

Las compañías o empresas pueden automatizar todo el proceso de compra; registrando todas estas transacciones; toda esta información puede ser utilizada y analizada para generar nuevas estrategias o decisiones, otro de los grandes aportes que conlleva el uso del comercio electrónico.

Actualmente la empresa “FIBBY’S S.R.L.” no cuenta con el medio para realizar el comercio electrónico o con un medio de promoción en internet, por esta razón se ve la necesidad de desarrollar e implementar una aplicación web de comercio electrónico que utilice el responsive web design dando una experiencia completa desde cualquier dispositivo que acceda a la misma.

1.2 Antecedentes

1.2.1 Trabajos realizados en otras universidades

- Tesis de Grado: “Sistema de comercio electrónico escalable” por Ing. Leonardo Oscar Montesano, 2014 en la ciudad de Buenos Aires Argentina, cuyo objetivo es implementar un modelo de sistema de ventas por internet mateniendo la concepto de las tiendas físicas.
- Tesis de Grado: “Comercio Electrónico: Implementación de una Tienda Virtual a través del estudio comparativo de una Tienda Física utilizando software libre” por Franklin Eduardo Barros Sarmiento, Universidad Tecnologica de Informatica, Quito, con el objetivo Por ello este trabajo investigativo tiene como objetivo la implementación de una tienda virtual, con el propósito de mejorar la forma de mostrar los productos de una tienda física; aumentando de alguna manera las ventas de dicho negocio.

1.2.2 Trabajos realizados en la carrera de informática

- Tesis de Grado: “Sistema de gestión de ventas por internet” por Luigi Antonio Antequera Tamari, 2004 en la carrera de informática, Cuyo objetivo es Implementar un sistema de ventas mediante internet en la empresa Quazar Systems S.R.L. que permita seleccionar los componentes que formaran el pedido, como también mostrar los servicios de ofertas de la institución.
- Tesis de Grado: “Tienda Virtual Eyetek” por Benjamín Marcelo Vargas Salazar, 2007 en la carrera de informática, con el objetivo de Aplicar una herramienta para el desarrollo de interfaces de usuario, que permita

mejorar la experiencia de navegación, utilizando contenido sofisticado, y mejorando la comunicación y transmisión de datos entre el cliente y el servidor.

- Tesis de Grado: “venta y promoción en línea: empresa multi acceso (La Paz)” por Macario Rocha Coria, del año 2010, la cual tiene como objetivo Desarrollar e implementar un sistema de información de venta y promoción en línea para la empresa Multi-Acceso, que permite mejorar y acercar la información al consumidor final sobre el amplio catalogo de productos ofertas y promociones planificadas.

1.3 Problemas

1.3.1 Problemas Secundarios

- Los productos no cuentan con ningún tipo de promoción en ningún medio.
- El ingreso económico en la empresa se mantiene estancado durante las dos últimas gestiones; esto ocasiona que el capital de la empresa no se incremente.
- El capital de la empresa no es suficiente para la apertura de nuevas sucursales; lo que causa que la empresa no pueda expandirse o abrir nuevos mercados.
- Los productos permanecen mucho tiempo en los depósitos; por tanto el capital no produce utilidades.
- No existe suficiente carga laboral en el área de producción; lo que incrementa el tiempo de ocio en los empleados.

Todo lo anterior repercute en pérdidas económicas en la empresa.

1.3.2 Problema General

¿Cómo se puede incrementar el número de ventas y contratos en la empresa “FIBBY’S S.R.L.”?

1.4 Objetivos

1.4.1 Objetivo General

Desarrollar un sistema web de comercio electrónico y control de inventario, que promueva los productos de la empresa "FIBBY'S S.R.L." permitiendo incrementar el número de ventas y cotizaciones para la empresa.

1.4.2 Objetivos Específicos

- Promocionar los productos por medio de una pagina web.
- Buscar un nuevo medio de venta para los productos, que tenga un costo bajo y sea sostenible.
- Al tener otro medio de promocion se incrementan las posibles ventas, lo que incrementan las utilidades que desemboca en mas capital para invertir en produccion e importacion.
- Generar reportes y utilizar la información para crear promociones, para los productos que permanecen demasiado tiempo en el almacen.
- Automatizar el control del inventario, utilizando la información generada para la planeación en el área de producción.

1.5 Justificación

1.5.1 Justificación Económica

Se implementará el sistema de comercio electrónico para potenciar el proceso de venta y promoción de los productos ofrecidos, facilitando el proceso de búsqueda, selección y adquisición de los productos; además llegando a más clientes potenciales, con lo que se pretende incrementar los ingresos en la empresa. Además controlar el inventario de la empresa pudiendo programar ofertas, planificando las actividades del área de producción y las importaciones.

1.5.2 Justificación Social

La implementación del sistema beneficiará a dos grupos de personas:

A los clientes permitiendo realizar compras y cotizaciones sin la necesidad de demorar un tiempo extenso en este proceso y teniendo toda la información sobre los productos y servicios de una manera ordenada y accesible.

Al crear promociones en base a la información generada por el sistema se reducen las pérdidas económicas en la empresa conserva el capital lo que repercute en mantener las fuentes laborales que brinda la empresa a su personal (administrador, personal de ventas, personal de producción); inclusive darse la posibilidad de incrementar las mismas.

1.5.3 Justificación Técnica

En la actualidad con el avance tecnológico existen nuevos dispositivos que permiten el acceso a internet (laptop, Smartphone, tabletas), medios de pago electrónicos y mas las herramientas para el desarrollo de software; la unión de estas técnicas hacen posible que los negocios o comercio de cualquier nivel puedan acceder al comercio electrónico a un costo accesible.

La cantidad de comercio llevada a cabo electrónicamente ha crecido de manera extraordinaria debido a Internet. Una gran variedad de comercio se realiza de esta manera, estimulando la creación y utilización de innovaciones como la transferencia de fondos electrónica, la administración de cadenas de suministro, el marketing en Internet, los sistemas de administración del inventario y los sistemas automatizados de recolección de datos. Siendo tan versátil el comercio electrónico es necesario aplicarlo en nuestra sociedad.

1.6 Alcances

Los alcances de este proyecto son:

- Se podra acceder al sistema desde cualquier dispositivo con internet.
- La informacion de productos y cotizaciones sera accesible para cualquier persona.

- Tendrá la opción de registrar un carrito de compras con diferentes productos.
- Se podrá realizar pagos electrónicos.
- Se podrá registrar nuevos productos, incrementar o reducir el stock de los mismos.
- Se relizaran lo siguientes modulos:
 - ✓ Modulo de ventas
 - ✓ Modulo de inventarios
 - ✓ Modulo de cotizaciones

1.7 Límites

El sistema tendrá las siguientes limitaciones:

- Será necesario registrarse como cliente para realizar la compra de los productos.
- El sistema no contempla la interacción con el sistema contable de la empresa.
- En el caso de cotizaciones para producción, solamente se generará un respaldo de la cotización, con el cual se continuará el proceso de persona a persona con un encargado de la empresa.

1.8 Metodología

1.8.1 Metodología de Investigación

1.8.1.1 Método Científico

El método científico es un método de investigación usado principalmente en la producción de conocimiento en las ciencias. Para ser llamado científico, un método de investigación debe basarse en la empírica y en la medición, sujeto a los principios específicos de las pruebas de razonamiento. Según el Oxford English Dictionary, el método científico es: “Un método o procedimiento que ha caracterizado a la ciencia natural desde el siglo XVII, que consiste en la

observación sistemática, medición, experimentación, la formulación, análisis y modificación de las hipótesis”.

El método científico está sustentado por dos pilares fundamentales. El primero de ellos es la reproducibilidad, es decir, la capacidad de repetir un determinado experimento, en cualquier lugar y por cualquier persona. Este pilar se basa, esencialmente, en la comunicación y publicidad de los resultados obtenidos (por ej. en forma de artículo científico). El segundo pilar es la refutabilidad. Es decir, que toda proposición científica tiene que ser susceptible de ser falsada o refutada (falsacionismo). Esto implica que se podrían diseñar experimentos, que en el caso de dar resultados distintos a los predichos, negarían la hipótesis puesta a prueba. La falsabilidad no es otra cosa que el *modus tollendo tollens* del método hipotético deductivo experimental. Según James B. Conant, no existe un método científico. El científico usa métodos definitorios, métodos clasificatorios, métodos estadísticos, métodos hipotético-deductivos, procedimientos de medición, entre otros. Y según esto, referirse al método científico es referirse a este conjunto de tácticas empleadas para constituir el conocimiento, sujetas al devenir histórico, y que eventualmente podrían ser otras en el futuro. Ello nos conduce tratar de sistematizar las distintas ramas dentro del campo del método científico.

1.8.1.2 Método Explicativo

El método exploratorio se aplica en el problema de investigación con el propósito de destacar los aspectos fundamentales de una problemática determinada y encontrar los procedimientos adecuados para elaborar una investigación posterior.

Los estudios exploratorios se efectúan, normalmente, cuando el objetivo es examinar un tema o problema de investigación poco estudiado o que no ha sido abordado antes.

Al utilizar este método se pretende aumentar el grado de familiaridad con fenómenos relativamente desconocidos, obtener información sobre la

posibilidad de llevar a cabo una investigación más completa sobre un contexto particular de la vida real, investigar problemas del comportamiento humano que consideren cruciales los profesionales de determinada área, identificar conceptos o variables promisorias, establecer prioridades para investigaciones posteriores o sugerir afirmaciones (postulados) verificables

Características:

- El objetivo es documentar el objeto de forma tan completa como sea posible, y no solo aquellos temas que fueron documentados en estudios anteriores.
- Búsqueda fenomenológica de una comprensión profunda y desconfianza en las anteriores descripciones y explicaciones.
- Recoger e identificar antecedentes generales, números y cuantificaciones, temas y tópicos respecto del problema investigado.
- Determinan tendencias, identifican relaciones potenciales entre variables y establecen el 'tono' de investigaciones posteriores más rigurosas.

1.8.1.3 Método Descriptivo

En las investigaciones de tipo descriptiva, llamadas también investigaciones diagnósticas, buena parte de lo que se escribe y estudia sobre lo social no va mucho más allá de este nivel. Consiste, fundamentalmente, en caracterizar un fenómeno o situación concreta indicando sus rasgos más peculiares o diferenciadores.

En la ciencia fáctica, la descripción consiste, según Bunge, en responder a las siguientes cuestiones:

- ¿Qué es? > Correlato.
- ¿Cómo es? > Propiedades.
- ¿Dónde está? > Lugar.
- ¿De qué está hecho? > Composición.
- ¿Cómo están sus partes, si las tiene, interrelacionadas? > Configuración.

- ¿Cuánto? > Cantidad

El objetivo de la investigación descriptiva consiste en llegar a conocer las situaciones, costumbres y actitudes predominantes a través de la descripción exacta de las actividades, objetos, procesos y personas. Su meta no se limita a la recolección de datos, sino a la predicción e identificación de las relaciones que existen entre dos o más variables. Los investigadores no son meros tabuladores, sino que recogen los datos sobre la base de una hipótesis o teoría, exponen y resumen la información de manera cuidadosa y luego analizan minuciosamente los resultados, a fin de extraer generalizaciones significativas que contribuyan al conocimiento.



2.1 Marco Institucional

La empresa “FIBBY’S S.R.L.” fue creada en 2008, cuenta con la oficina central en la ciudad de La Paz, ubicada en la Pasaje Ortega Nro. 102 donde se encuentra también el centro de producción, además de dos sucursales de ventas en la ciudad de La Paz y El Alto.

- ❖ La empresa realiza las siguientes actividades:
- ❖ Importación y venta de morrales y accesorios de vestir.
- ❖ La empresa produce morrales ya accesorios de vestir para instituciones públicas y privadas.

En la actualidad la empresa realiza un comercio tradicional en las dos sucursales con las que cuenta. La cotización y producción se realiza solamente en la oficina central. (FIBBY’S, 2008)

2.1.1 Misión

Ser la organización líder en el desarrollo, producción y comercialización de productos de excelente calidad, dentro de un mercado Boliviano, promoviendo el desarrollo integral de nuestro personal, clientes, proveedores y colaboradores, logrando crecimiento, rentabilidad y solidez, con responsabilidad social. (FIBBY’S, 2008)

2.1.2 Visión

La empresa aspira en convertirse en un referente en la producción nacional abriendo mercados en el exterior. (FIBBY’S, 2008)

2.2 Metodología

Una metodología es un conjunto integrado de técnicas y métodos que permite abordar de forma homogénea y abierta cada una de las actividades del ciclo de

vida de un proyecto de desarrollo. Es un proceso de software detallado y completo.

Las metodologías se basan en una combinación de los modelos de proceso genéricos (cascada, incremental...). Definen artefactos, roles y actividades, junto con prácticas y técnicas recomendadas.

La metodología para el desarrollo de software en un modo sistemático de realizar, gestionar y administrar un proyecto para llevarlo a cabo con altas posibilidades de éxito. Una metodología para el desarrollo de software comprende los procesos a seguir sistemáticamente para idear, implementar y mantener un producto software desde que surge la necesidad del producto hasta que cumplimos el objetivo por el cual fue creado.

Una metodología de desarrollo de software es un marco de trabajo que se usa para estructurar, planificar y controlar el proceso de desarrollo de sistemas de información. Una gran variedad de estos marcos de trabajo han evolucionado durante los años, cada uno con sus propias fortalezas y debilidades. Una metodología de desarrollo de sistemas no tiene que ser necesariamente adecuada para usarla en todos los proyectos. Cada una de las metodologías disponibles es más adecuada para tipos específicos de proyectos, basados en consideraciones técnicas, organizacionales, de proyecto y de equipo. (INTECO, 2009)

2.2.1 El Proceso Unificado Agil – Aup

La mayor parte de desarrollo de software es una actividad caótica, frecuentemente caracterizada por la frase "codifica y corrige". El software se escribe sin la mayor parte de un plan subyacente, y el diseño del sistema se construye de muchas decisiones a corto plazo. En realidad, esto funciona muy bien cuando el sistema es pequeño, pero a medida que crece el sistema se

vuelve cada vez más difícil de agregar nuevas funcionalidades. El movimiento original para tratar de cambiar esta introdujo la noción de la metodología. Estas metodologías imponen un proceso disciplinado sobre el desarrollo de software con el objetivo de que el desarrollo de software sea más predecible y más eficiente. Lo hacen mediante el desarrollo de un proceso detallado con un fuerte énfasis en la planificación inspirada en otras disciplinas de ingeniería y que las mismas no resultaron exitosas.

Las metodologías ágiles se desarrollaron como una reacción a estas metodologías que proporcionan procesos suficientes para obtener una rentabilidad razonable. El resultado de todo esto es que los métodos ágiles tienen algunos cambios significativos en el énfasis de los métodos de ingeniería. La diferencia más inmediata es que son menos orientadas a documentos, por lo general haciendo hincapié en una menor cantidad de documentación para una tarea dada.

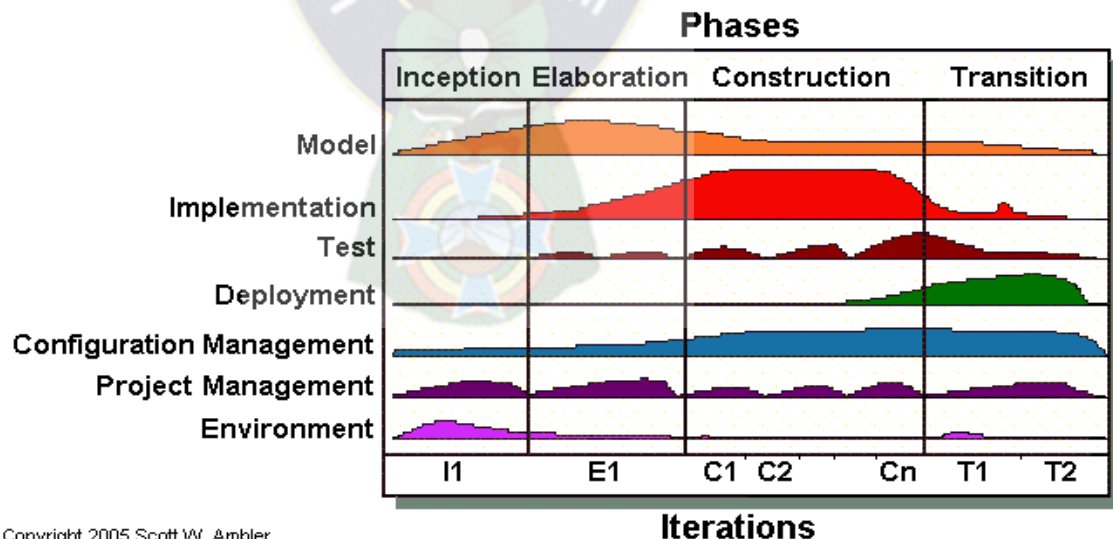
El Proceso Unificado Ágil (Agile Unified Process – AUP) de Scott Ambler es una versión simplificada del Rational Unified Process (RUP), el cual fue desarrollado por IBM. AUP describe una manera simple de entender el desarrollo de aplicaciones de negocio usando técnicas ágiles y conceptos heredados del RUP.

En los proyectos que usan AUP, normalmente se entregan versiones de desarrollo al final de cada iteración. Una versión de desarrollo de una aplicación es una versión que potencialmente puede ser lanzada en producción si pasa la garantía de calidad de pre-producción, supera la fase de pruebas y los procesos de despliegue. En un proceso de desarrollo AUP, normalmente la primera versión de producción tarda más que las demás, desarrollándose antes más versiones de desarrollo, pero en las siguientes iteraciones tardarán menos en desarrollarse una versión de producción. Un enfoque en las tareas de despliegue ayuda a evitar problemas, y permite aprender de la experiencia a lo largo del desarrollo.

Sus creadores han tratado de mantenerlo lo más simple posible. El enfoque usa técnicas de Desarrollo dirigido por pruebas (Test Driven Development), Desarrollo Ágil Dirigido por Modelo (Agile Model Driven Development), Administración de Cambios Ágil (Agile Change Management) y Refactorización de Base de Datos.

2.2.1.1 Ciclo de Vida de AUP

En el Proceso Unificado Ágil la disciplina de Modelado abarca las disciplinas de Modelado del Negocio, de Requerimientos, de Análisis y Diseño de RUP. El modelado es una parte importante en AUP como se puede ver en la Figura, pero no domina el proceso, puede ir creando modelos ágiles y documentos los cuales son apenas lo suficientemente bueno. Segundo, las disciplinas de la Administración de la Configuración y Cambios ahora es la disciplina de la Administración de la Configuración. En el desarrollo ágil las actividades de administración de cambios son típicamente parte del esfuerzo de la administración de requerimientos, la cual es parte de la disciplina de Modelado.



Copyright 2005 Scott W. Ambler

Figura 1. Fase y Disciplinas del AUP
Fuente: <http://www.cc.una.ac.cr/AUP/index.html>

En lugar del enfoque "bigbang" donde se entrega todo el software en una sola liberación en vez de hacerlo por partes (por ejemplo versión 1, versión 2, y así sucesivamente). Los equipos de AUP típicamente liberan versiones desarrolladas al final de cada iteración mediante demostraciones. Una versión de desarrollo de una aplicación es algo que podría ser liberado en la producción si se ponen a través de su pre-producción el aseguramiento de la calidad (QA), pruebas, y proceso de despliegue. En la siguiente figura usted verá que la primera versión de producción comúnmente toma más tiempo de entrega que otras versiones subsecuentes; en la primera liberación de un sistema probablemente necesita colocar mucha "cañería" en su lugar y su equipo probablemente no cuente con lo suficiente para convertirlo en algo colaborativamente eficiente.



Figura 2. Modelo de Liberación de Producción
Fuente: <http://www.cc.una.ac.cr/AUP/index.html>

La primera versión de producción puede tardar 12 meses en liberación, la segunda nueve meses para liberación, y las demás liberaciones son entregadas cada 6 meses. Un enfoque temprano en el caso del despliegue no solo le permite evitar problemas, le permite aventajarse en su experiencia durante el desarrollo, por ejemplo, cuando se está desplegando software en un ambiente deben tomarse notas de qué funciona y de qué no, estas pueden servir como columna vertebral para la instalación de scripts.

2.2.1.2 Disciplina del AUP

Las disciplinas se llevan a cabo de manera sistemática, a la definición de las actividades que realizan los miembros del equipo de desarrollo a fin de

desarrollar, validar, y entregar el software de trabajo que responda a las necesidades de sus interlocutores. Las disciplinas son:

- a. Modelo. El objetivo de esta disciplina es entender el negocio de la organización, el problema de dominio que se abordan en el proyecto, y determinar una solución viable para resolver el problema de dominio.
- b. Aplicación. El objetivo de esta disciplina es transformar su modelo (s) en código ejecutable y realizar un nivel básico de las pruebas, en particular, la unidad de pruebas.
- c. Prueba. El objetivo de esta disciplina consiste en realizar una evaluación objetiva para garantizar la calidad. Esto incluye la búsqueda de defectos, validar que el sistema funciona tal como está establecido, y verificando que se cumplan los requisitos.
- d. Despliegue. El objetivo de esta disciplina es la prestación y ejecución del sistema y que el mismo este a disposición de los usuarios finales.
- e. Gestión de configuración. El objetivo de esta disciplina es la gestión de acceso a herramientas de su proyecto. Esto incluye no sólo el seguimiento de las versiones con el tiempo, sino también el control y gestión del cambio para ellos.
- f. Gestión de proyectos. El objetivo de esta disciplina es dirigir las actividades que se lleva a cabo en el proyecto. Esto incluye la gestión de riesgos, la dirección de personas (la asignación de tareas, el seguimiento de los progresos, etc), coordinación con el personal y los sistemas fuera del alcance del proyecto para asegurarse de que es entregado a tiempo y dentro del presupuesto.
- g. Entorno. El objetivo de esta disciplina es apoyar el resto de los esfuerzos por garantizar que el proceso sea el adecuado, la orientación (normas y directrices), y herramientas (hardware, software, etc) estén disponibles para el equipo según sea necesario.

2.2.1.3 Fases de AUP

AUP está caracterizado por ser "serial en lo grande", algo que usted puede ver a través de estas cuatro fases las cuales se pueden mover en una forma serial.

a. FASE DE INICIO

El objetivo principal de la fase de inicio es registrar el consenso de los interesados del proyecto en relación a los objetivos del proyecto para obtener el financiamiento. Las principales actividades de esta fase incluyen:

Disciplina	Principales Actividades	
Modelado	<ul style="list-style-type: none">• Inicial, los requisitos de alto nivel de modelado• Modelado de arquitectura inicial y de alto nivel	
Implementación	<ul style="list-style-type: none">• Technical prototyping Prototipos Técnica• Interfaz de usuario de prototipos	
Prueba	<ul style="list-style-type: none">• Planificación inicial de ensayo• Revise los productos de trabajo de gestión de proyectos iniciales• Revisar los modelos iniciales	
Despliegue	<ul style="list-style-type: none">• Identificar la ventana de posible liberación• Comience alto nivel de planificación de despliegue	
Adm. de la Configuración	<ul style="list-style-type: none">• Configuración del entorno de configuración• Ponga todos los productos de trabajo bajo control CM	
Adm. de Proyectos	<ul style="list-style-type: none">• Empezar a construir el equipo• Construir relaciones con sus grupos de interés del proyecto• Determinar la viabilidad del proyecto• Desarrollar un programa de alto nivel para todo el proyecto• Desarrollar un plan de iteración detallado para la próxima iteración• Gestionar el riesgo• Obtener apoyo de los interesados y la financiación• Cierre de la fase	
Entorno	<ul style="list-style-type: none">• Configuración del entorno de trabajo• Identifique la categoría de proyecto	

Tabla 1. Disciplinas y Principales Actividades de la fase de comienzo
Fuente: [Ambysoft, 2006]

b. FASE DE ELABORACION

El principal objetivo de la fase de elaboración es probar la arquitectura del sistema a desarrollar. El punto es asegurar que el equipo puede desarrollar un sistema que satisfaga los requisitos, y la mejor manera de hacerlo que es la construcción de extremo a extremo del esqueleto de trabajo del sistema conocido como "prototipo de la arquitectura". Esto es en realidad un concepto pobre porque mucha gente piensa en deshacerse de los prototipos. En cambio, su significado es software funcional de alto nivel, el cual incluye varios casos de uso de alto riesgo (a partir de un punto de vista técnico) para demostrar que el sistema es técnicamente factible.

Disciplina	Principales Actividades	
Modelado	<ul style="list-style-type: none">• Identificar los riesgos técnicos• Architectural modeling Modelado arquitectónico• User interface prototyping Interfaz de usuario de prototipos	
Implementación	<ul style="list-style-type: none">• Demostrar la arquitectura	
Prueba	<ul style="list-style-type: none">• Validar la arquitectura• test model Desarrolle su modelo de prueba	
Despliegue	<ul style="list-style-type: none">• deployment plan Actualice su plan de despliegue	
Adm. de la Configuración	<ul style="list-style-type: none">• Ponga todos los productos de trabajo bajo control CM	
Adm. de Proyectos	<ul style="list-style-type: none">• Construir el equipo• Proteja el equipo• Obtener recursos• Gestionar el riesgo• Actualizar el plan del proyecto• Close out the phase Cierre de la fase	
Entorno	<ul style="list-style-type: none">• Evolucionar el entorno de trabajo• Adaptar los materiales de proceso	

Tabla 2 Disciplinas y Principales Actividades de la fase de elaboración
Fuente: [Ambysoft, 2006]

c. FASE DE CONSTRUCCION

El objetivo de la fase de Construcción consiste en desarrollar el sistema hasta el punto en que está listo para la pre-producción de pruebas. En las etapas anteriores, la mayoría de los requisitos han sido identificados y la arquitectura del sistema se ha establecido. El énfasis es priorizar y comprender los requerimientos, modelado que ataca una solución y, a luego, la codificación y las pruebas del software. Si es necesario, las primeras versiones del sistema se desarrollan, ya sea interna o externamente, para obtener los comentarios de los usuarios.

Disciplina	Principales Actividades	
Modelado	<ul style="list-style-type: none">• Análisis de modelo de asalto• Diseño modelo de asalto• Documento decisiones críticas de diseño	
Implementación	<ul style="list-style-type: none">• Pruebe primero• Construir continua• Evolucionar la lógica de dominio• Evolucionar la interfaz de usuario• Evoluciona el esquema de datos• Crear enlaces con los activos heredados• Escribir secuencias de comandos de conversión de datos	
Prueba	<ul style="list-style-type: none">• Pruebe el software• Desarrolle su modelo de prueba	
Despliegue	<ul style="list-style-type: none">• Desarrollar (de) Los scripts de instalación• Elaborar notas de la versión• Desarrollar documentación inicial• Implementar el sistema en entornos de pre-producción	
Adm. de la Configuración	<ul style="list-style-type: none">• Ponga todos los productos de trabajo bajo control CM	
Adm. de	<ul style="list-style-type: none">• Manejar el equipo• Gestionar el riesgo	

Proyectos	<ul style="list-style-type: none"> • Actualizar el plan del proyecto • Cierre de la fase
Entorno	<ul style="list-style-type: none"> • Apoyar al equipo • Evolucionar el entorno de trabajo • Entorno de formación de configuración

Tabla 3. Disciplinas y Principales Actividades de la fase de construcción
Fuente: [Ambysoft, 2006]

d. FASE DE TRANSICION

La fase de Transición se enfoca en liberar el sistema a producción. Deben hacerse pruebas extensivas a lo largo de esta fase, incluyendo las pruebas beta. Una buena afinación del proyecto tiene lugar aquí incluyendo el retrabajo dirigido a los defectos significantes (su usuario puede escoger aceptar la existencia de algunos defectos conocidos en la versión actual).

El tiempo y esfuerzo necesarios en la Transición varía de un proyecto a otro. Shrink-wrapped software supone la fabricación y distribución de software y documentación. Sistemas internos son generalmente más simples de desplegar que sistemas externos. Los sistemas de alta visibilidad pueden requerir pruebas beta extensiva por grupos pequeños antes de liberarse a la población más grande. La liberación de un nuevo sistema de marca puede traer consigo la compra y configuración de hardware mientras se actualiza un sistema existente que también puede traer una conversión de datos y una coordinación exhaustiva con la comunidad de usuarios. Cada proyecto es diferente.

Disciplina	Principales Actividades
Modelado	<ul style="list-style-type: none"> • Modelo storming • Finalizar la documentación visión general del sistema
Implementación	<ul style="list-style-type: none"> • Corregir los defectos
Prueba	<ul style="list-style-type: none"> • Validar el sistema • Validar la documentación • Finalizar el modelo de prueba
Despliegue	<ul style="list-style-type: none"> • Finalice el paquete de implementación • Finalizar la documentación • Anunciar la implementación

	<ul style="list-style-type: none"> • Capacitar a las personas • Implementar el sistema en producción
Adm. de la Configuración	<ul style="list-style-type: none"> • Ponga todos los productos de trabajo bajo control CM
Adm. de Proyectos	<ul style="list-style-type: none"> • Manejar el equipo • Cierre de la fase • Iniciar el próximo ciclo del proyecto
Entorno	<ul style="list-style-type: none"> • Ambientes de apoyo de operaciones de instalación y / o Recuperar licencias de software

Tabla 4. Disciplinas y Principales Actividades de la fase de Transición
Fuente: [Ambysoft, 2006]

2.2.1.4 Técnicas Ágiles Que Aplica Aup

a. Desarrollo dirigido por pruebas (Test Driven Development – TDD)

El Desarrollo Dirigido por Pruebas, es una práctica de programación que involucra otras dos prácticas: Escribir las pruebas primero (Test First Development) y Refactorización (Refactoring). Para escribir las pruebas generalmente se utilizan las pruebas unitarias (unit test). En primer lugar se escribe una prueba y se verifica que las pruebas fallen, luego se implementa el código que haga que la prueba pase satisfactoriamente y seguidamente se refactoriza el código escrito. El propósito del desarrollo guiado por pruebas es lograr un código limpio que funcione. La idea es que los requisitos sean traducidos a pruebas, de este modo, cuando las pruebas pasen se garantizará que los requisitos se hayan implementado correctamente.

b. Desarrollo Dirigido por el Modelado (Agile ModelDrivenDevelopment – AMDD)

El Desarrollo Dirigido por el Modelado Ágil, adopta un enfoque realista: su objetivo es describir cómo los desarrolladores y los interesados pueden colaborar entre sí para crear modelos que son apenas lo suficientemente bueno. Se supone que cada individuo tiene algunas habilidades de modelado, o

al menos algún conocimiento de dominio, que se aplicará junto con un equipo a fin de obtener el trabajo realizado. Es razonable suponer que los desarrolladores comprendan un puñado de las técnicas de modelado por ahí, pero no todos de ellos. También es razonable suponer que la gente está dispuesta a aprender nuevas técnicas a través del tiempo, a menudo trabajando con alguien que ya tiene esas habilidades. AMDD no requiere a todos a ser un experto en el modelado, sólo les obliga a estar dispuestos a probar. AMDD también permite a la gente a usar la herramienta de modelado más adecuado para el trabajo, las herramientas suelen ser muy simples, tales como pizarras digitales o de papel, porque quiere encontrar la manera de comunicarse de manera efectiva, no documentarán exhaustivamente. No hay nada malo con herramientas CASE sofisticadas en manos de personas que saben cómo usarlos, pero AMDD no depende de estas herramientas.

c. Gestión de Cambios Ágil (Agile Change Management – ACM)

El desarrollo de software ágil acepta la idea de cambios en los requisitos que se desarrollará en el transcurso de un proyecto. Entender que los requerimientos evolucionan con el tiempo que cualquier inversión temprana en la documentación detallada sólo será en vano. En lugar de esto se hará requisitos iniciales para imaginar e identificar lo suficiente para el ámbito de aplicación del proyecto y desarrollar un programa de alto nivel y presupuesto.

d. Refactorización de Base de Datos

Una refactorización de base de datos es un simple cambio a un esquema de base de datos que mejora su diseño, conservando ambos semántica de comportamiento y de información. En aras de esta discusión un esquema de base de datos incluye tanto los aspectos estructurales, como la tabla de vista y definiciones así como los aspectos funcionales, tales como procedimientos almacenados y disparadores. Una cosa interesante a destacar es que una refactorización de base de datos es conceptualmente más difícil que una refactorización de código, refactorización de código sólo es necesario para

mantener la semántica de comportamiento mientras refactorizaciones base de datos también debe mantener la semántica de información.

2.2.1.5 Principios de AUP

AUP, se basa en los siguientes principios:

- a. Simplicidad. Todo se describe concisamente utilizando poca documentación, no miles de ellas.
- b. Agilidad. AUP se ajusta a los valores y principios de desarrollo de software ágil y la Alianza Ágil.
- c. Centrarse en actividades de alto valor. La atención se centra en las actividades que en realidad lo requieren, no en todo el proyecto.
- d. Independencia de herramientas. Usted puede usar cualquier conjunto de herramientas que desea con el AUP. Se sugiere utilizar las herramientas más adecuadas para el trabajo, que a menudo son las herramientas simples o incluso herramientas de código abierto.
- e. Usted querrá adaptar este producto para satisfacer sus propias necesidades. La metodología AUP es un producto de fácil uso utilizando cualquier herramienta. No es necesario comprar una herramienta especial, o tomar un curso, para adaptar esta metodología.

2.2.2 Modelado

2.2.3 Uwe

UWE (UML-Based Web Engineering) es una propuesta basada en UML y en el proceso unificado para modelar aplicaciones web. Esta propuesta está formada por una notación para especificar el dominio (basada en UML) y un modelo para llevar a cabo el desarrollo del proceso de modelado. Los sistemas adaptativos y la sistematización son dos aspectos sobre los que se enfoca UWE.

UWE es un método de ingeniería del software para el desarrollo de aplicaciones web basado en UML. Cualquier tipo de diagrama UML puede ser usado, porque UWE es una extensión de UML.

Además de estar considerado como una extensión del estándar UML, también se basa en otros estándares como por ejemplo: XMI como modelo de intercambio de formato, MOF para el metamodelado, los principios de modelado de MDA, el modelo de transformación del lenguaje QVT y XML.

El modelo que propone UWE está compuesto por 6 etapas o sub-modelos:

- a. Modelo de Casos de Uso:* modelo para capturar los requisitos del sistema.
- b. Modelo de Contenido:* es un modelo conceptual para el desarrollo del contenido.
- c. Modelo de Usuario:* es modelo de navegación, en el cual se incluyen modelos estáticos y modelos dinámicos.
- d. Modelo de estructura:* en el cual se encuentra la presentación del sistema y el modelo de flujo.
- e. Modelo Abstracto:* incluye el modelo a de interfaz de usuario y el modelo de ciclo de vida del objeto.
- f. Modelo de Adaptación.*

En cuanto a los requisitos, UWE los clasifica dependiendo del carácter de cada uno. Además distingue entre las fases de captura, definición y validación de requisitos.

2.2.3.1 UWE y su Relación con UML

UWE define una extensión del Lenguaje Unificado de Modelado (UML). Ésta, es considerada como una extensión ligera de peso e incluye en su definición tipos, etiquetas de valores y restricciones para las características específicas del

diseño Web, las cuales, unidas a las definiciones de UML forman el conjunto de objetos de modelado que se usarán para el desarrollo del modelo utilizado en UWE.

Las funcionalidades que cubren UWE abarcan áreas relacionadas con la Web como la navegación, presentación, los procesos de negocio y los aspectos de adaptación.

Una de las ventajas de que UWE extienda el estándar UML es la flexibilidad de éste para la definición de un lenguaje de modelado específico para el dominio web y sobre todo la aceptación universal de dicho estándar en el campo de la ingeniería del software.

Otra gran ventaja es que actualmente existen múltiples herramientas CASE basadas en UML, con lo cual es relativamente sencillo su utilización y ampliación para utilizar los objetos de modelado definidos en UWE. Estas herramientas se verán en el siguiente punto.

2.2.3.2 Herramientas Software

Como se ha explicado en la sección anterior, para aplicar las funcionalidades que añade UWE, se utilizan las mismas herramientas software de modelado basadas en UML y se les añade una extensión a la aplicación para que permita estas nuevas funcionalidades.

Se ha desarrollado un plugin para una de las herramientas UML más conocidas, MagicDraw, que es una herramienta que soporta la versión 2.1.2 de este estándar para lenguajes de programación como por ejemplo Java, C++ o C#. Este plugin llamado MagicUWE, es de libre distribución pudiendo adquirirse gratuitamente y está desarrollado para la versión 16 de MagicDraw. En la página oficial de UWE se encuentra disponible así como un manual de instalación y uso de esta extensión.

También se ha desarrollado UWEet, que es un plugin para la herramienta UML de código abierto UMLet. Esta herramienta se caracteriza por una interfaz simple para el usuario y por su compatibilidad con Eclipse a la hora de compartir diagramas UML.

También puede exportar estos a distintos formatos como el archiconocido PDF. Dicho esto, este plugin proporciona a UWEet de una paleta en su interfaz con todos los elementos que son definidos por UWE, permitiendo así la extensión del lenguaje UML. UWEet también se encuentra disponible de manera gratuita en la página web oficial de UWE.

Para uno de los entornos de desarrollo más utilizados en todo el mundo, Eclipse, también se ha creado una extensión. Este plugin se denomina UWE4JSF y permite la generación automática de aplicaciones web para JavaServer Faces (JSF) platform.

Por último destacar que existe una herramienta software basada específicamente en la metodología UWE, esta herramienta fue desarrollada como una extensión de ArgoUML, herramienta de modelado basada en UML. Se trata de la aplicación ArgoUWE que permite la semiautomática generación de los modelos característicos de UWE como son el de navegación, el de presentación, el de procesos y el de adaptación. Está herramienta también se encuentra disponible en la página web oficial de UWE.

2.2.3.3 Modelos De UWE

En esta sección se explicarán los modelos para cada una de los aspectos web que cubre la metodología UWE, recordemos que estos aspectos eran navegación, presentación, los procesos de negocio y adaptación. Así procedemos a explicar con un breve ejemplo cada uno de estos modelos.

a. MODELO DE REQUISITOS

En UWE el modelado de requisitos consiste de dos partes:

1. Casos de uso

En UWE se distinguen casos de uso estereotipados con «browsing» y con «processing» para ilustrar si los datos persistentes de la aplicación son modificados o no. Como ejemplo para ilustrar de forma clara tomaremos el ejemplo de la página oficial de UWE sobre una agenda de direcciones.

Por ejemplo en la figura1, "SearchContact" modela la búsqueda de contactos y por ello lleva el estereotipo «browsing» pues los datos son solamente leídos y presentados al usuario. Los otros casos de uso por el contrario modelan cambios, lo que se especifica con el estereotipo «processing».

Nombres de estereotipos y sus iconos

- browsing
- ⌋ processing
- ☁ webUseCase

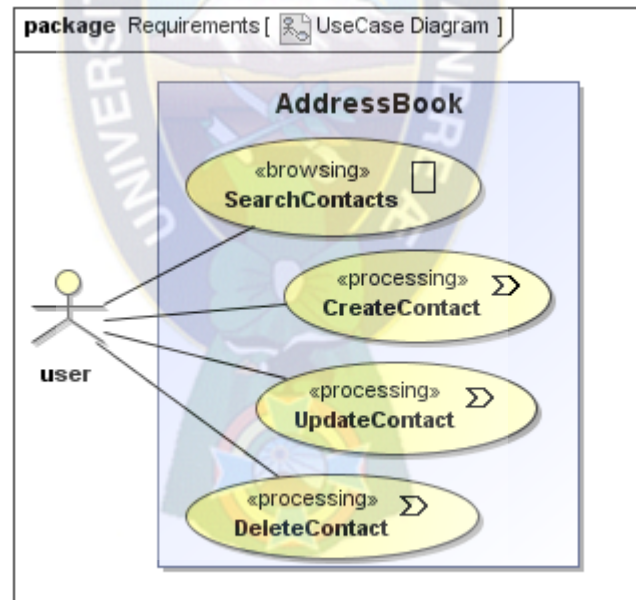








Figura 3. Diagrama de caso de uso en UWE

Fuente: <http://uwe.pst.ifi.lmu.de/teachingTutorialRequirementsSpanish.html>

//.Actividades

Como con casos de uso solamente es posible capturar poca información, cada caso de uso puede ser descrito más detalladamente mediante un proceso. Es decir, las acciones que son parte de un caso de uso así como los datos presentados al usuario y aquellos requeridos como entrada de datos pueden ser modelados con precisión como actividades.

nombrados de estereotipos y los iconos correspondientes

 »userAction	 »systemAction
 »displayAction	 »navigationAction
 »displayPin	 »interactionPin

Los dos estereotipos «userAction» y «systemAction» pueden ser usados análogamente al flujo de procesos. El estereotipo «userAction» es usado para indicar interacciones de usuario en la página web iniciando un proceso o respondiendo a un explícito requisito de información. Por lo contrario, «systemAction» describe acciones que son ejecutados por el sistema.

Durante ingeniería de requisitos es usual determinar que datos son representados donde y cuando. Para modelar grupos de presentación en UWE son usados el estereotipo «displayAction», mientras que los dos pines de acción estereotipados «interactionPin» y «displayPin» son usados para modelar la entrada y la salida de datos.

Finalmente el estereotipo «navigationAction», puede ser usado para modelar opciones de navegación y los elementos asociados de presentación.

Este modelo especifica cómo se encuentra relacionados los contenidos del sistema, es decir, define la estructura de los datos que se encuentran alojados en el sitio web. En la figura 2, se muestra un ejemplo de este modelo contenido en la página web de UWE.

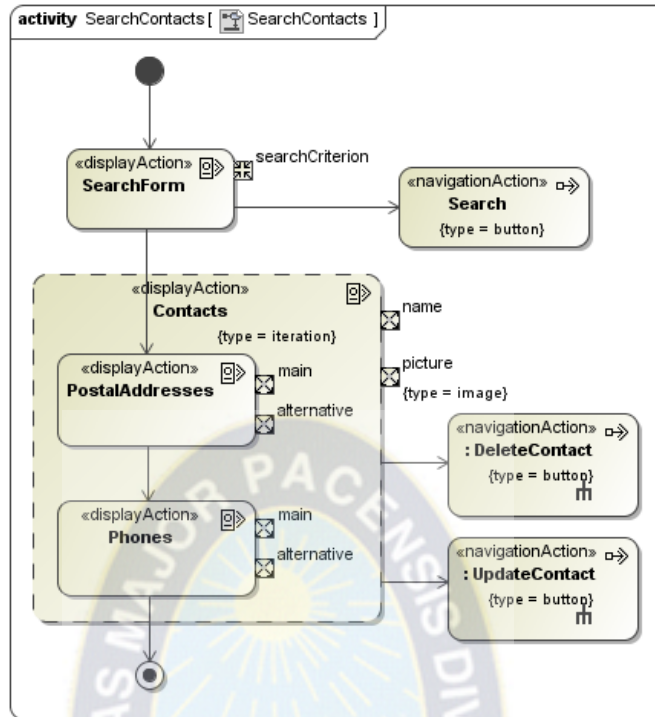


Figura 4. Diagrama de actividad de UWE
Fuente: <http://uwe.pst.ifi.lmu.de/teachingTutorialRequirementsSpanish.html>

b. MODELO DE CONTENIDO

Este modelo especifica cómo se encuentra relacionados los contenidos del sistema, es decir, define la estructura de los datos que se encuentran alojados en el sitio web. En la figura 3 se muestra un ejemplo de este modelo contenido en la página web de UWE.

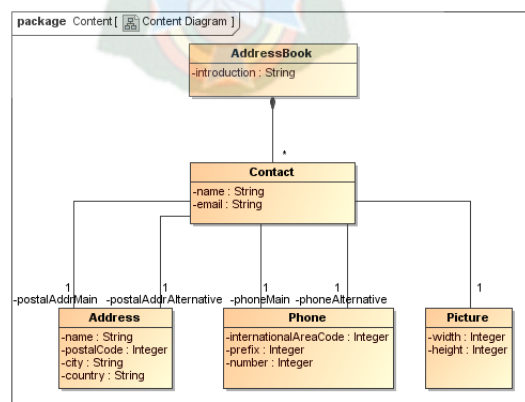


Figura 5. Diagrama de contenido en UWE
Fuente: <http://uwe.pst.ifi.lmu.de/teachingTutorialRequirementsSpanish.html>

Este es un diagrama UML normal de clases, por ello debemos pensar en las clases que son necesarias para el ejemplo. Primero queremos disponer de una clase agenda ("AddressBook") conteniendo un conjunto de contactos. Cada contacto debe contener un nombre, y puede contener una dirección de correo, dos números de teléfono y dos direcciones postales. El nombre y la dirección de correo son Strings, el teléfono y la dirección postal son clases que representan más información.

c. MODELO DE NAVEGACION

En un sistema para la web es útil saber cómo están enlazadas las páginas. Ello significa que necesitamos un diagrama conteniendo nodos (nodes) y enlaces (links).

Pero ¿qué es un nodo? Nodos son unidades de navegación y están conectados por medio de enlaces. Nodos pueden ser presentados en diferentes páginas o en una misma página.

Al mismo tiempo que explicamos este modelo con el ejemplo de la agenda de contactos, podemos ir viendo los distintos elementos que introduce la metodología UWE, los elementos introducidos son los siguientes:

Nombres de estereotipos y sus iconos

□ clase de navegación	☰ menú
≡ índice	? pregunta
→ visita guiada	➤ clase de proceso
➤ nodo externo	

En la figura 4 se puede observar el modelo de navegación reflejado en el ejemplo de la agenda de contactos.

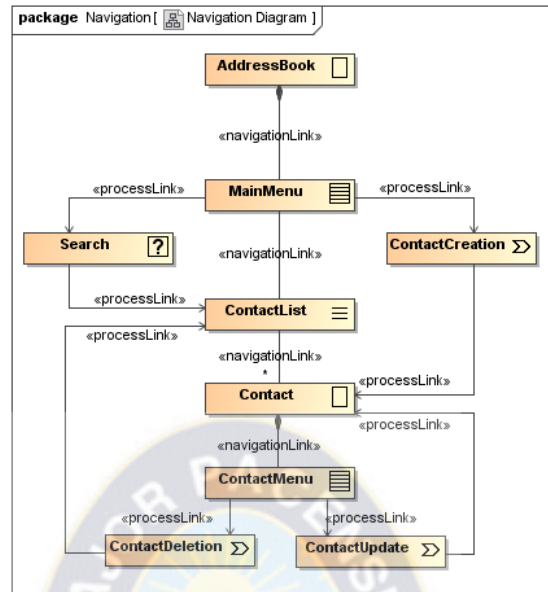


Figura 6. Diagrama de navegación en UWE
Fuente: <http://uwe.pst.ifi.lmu.de/teachingTutorialRequirementsSpanish.html>

Para empezar tenemos AddressBook como página de inicio, así que está etiquetada como {isHome} y como clase de navegación con el símbolo correspondiente (ver símbolos más abajo). La página de inicio enlaza con un menú, que sería nuestra página de índice, para ello la clase MainMenu esta etiquetada como pagina Menu.

Desde la clase MainMenu enlazamos con las clases Search (que implementará la función de buscar un contacto y es etiquetada con la etiqueta de query) que es un proceso predefinido, y con la clase ContactCreation (que creará un contacto), esta clase es un proceso no definido con lo cual llevará la etiqueta de processClass, así ambos enlaces serán del tipo process link.

Para finalizar vemos que la clase ContactCreation está enlazada con Contact ya que cuando se crea un nuevo contacto, este se debe mostrar. Como también cuando se realiza una búsqueda se debe mostrar la lista con los contactos del resultado, de ahí que exista otro processLink entre las clases Search y ContactList, esta ultima además etiquetada como index, al ser una lista.

d. MODELO DE PRESENTACION

En este modelo se representan las clases de navegación y de procesos que pertenecen a cada página web, estos son los elementos que introduce la metodología UWE en este modelo.

Nombres de estereotipos y sus iconos

 grupo de presentación	 página de presentación
 texto	 entrada de texto
 ancla	 SubirArchivo
 botón	 imagen
 formulario	 componente de cliente
 alternativas de presentación	 selección

En la figura 5 se muestra el diagrama de presentación del ejemplo de la agenda de contactos. Se puede observar que la clase contacto es presentada como presentation_class, cubriendo también diferentes textos y botones, esto significa que por cada contacto, tiene que ser mostrado un email, direcciones y los teléfonos. También se puede observar que la página de inicio AddressBook contiene un texto de introducción y un formulario de búsqueda con campo de texto y un botón para lanzar la búsqueda.

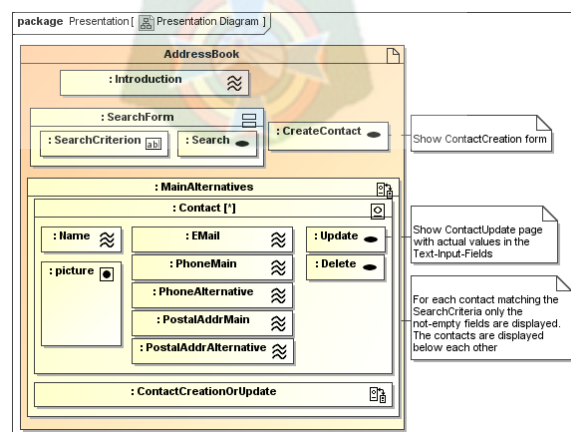


Figura 7. Diagrama de presentación en UWE
Fuente: <http://uwe.pst.ifi.lmu.de/teachingTutorialRequirementsSpanish.html>

e. MODELO DE PROCESOS

Hasta ahora podemos modelar muchos aspectos de nuestro sitio web. Pero no hemos hablado en ningún momento de que aspecto tienen las acciones de nuestras clases de proceso. El Modelo de Proceso comprende:

1. **Modelo de Estructura del Proceso**, que describe las relaciones entre las diferentes clases de proceso. Dado el ejemplo de la agenda de contactos en la figura 6 se puede observar el correspondiente diagrama de clases para este modelo.

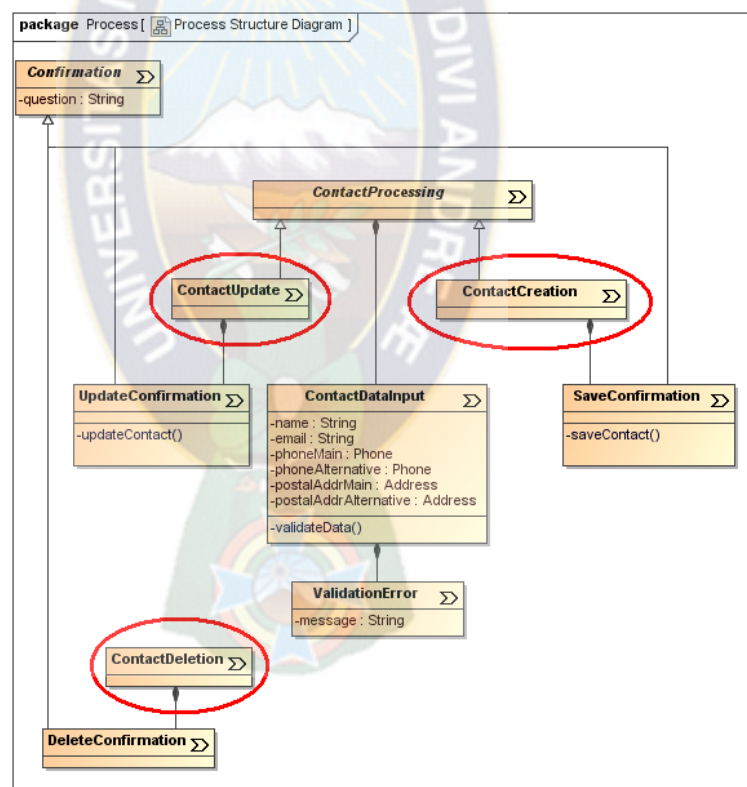


Figura 8. Diagrama de clases de proceso

Fuente: <http://uwe.pst.ifi.lmu.de/teachingTutorialRequirementsSpanish.html>

Como puede observarse, hemos agregado otras clases para expresar, que las tres operaciones requieren una confirmación (recuerda nuestro diagrama de

presentación) con una pregunta. Esto significa que si un usuario quiere borrar un contacto, un mensaje será mostrado, el cuál deberá ser confirmado con un ok para que el contacto sea borrado. ContactCreation and ContactUpdate funcionan en forma similar, ambos heredan de la clase abstracta ContactProcessing, asegurando que los campos de texto, que son atributos de ContactDataInput contienen valores válidos (por ejemplo podemos pensar en prohibir un nombre en blanco para prevenir entradas inservibles en la base de datos). No bien los datos han sido validados y no hay errores de validación (ValidationError) la página de confirmación es presentada al usuario. Para más detalles sobre las actividades, véase el próximo párrafo.

//. Modelo de Flujo del Proceso, que especifica las actividades conectadas con cada clase de proceso. Un flujo del proceso (flujo de trabajo) es representado como un diagrama de actividades, describiendo el comportamiento de una clase de proceso, por ejemplo que sucede en detalle, cuando el usuario navega a una clase de proceso. Los elementos que introduce UWE en este modelo son los siguientes:

Nombres de estereotipos y sus iconos

⌘»acción de usuario □»acción de sistema

El estereotipo «userAction» es usado para indicar interacciones de usuario con la página web iniciando un proceso o respondiendo a un requerimiento explícito de información. Por el contrario, «systemAction» describe acciones, que son ejecutadas por el sistema. Por ejemplo para la operación de borrado de contactos tenemos reflejado en la figura 7.

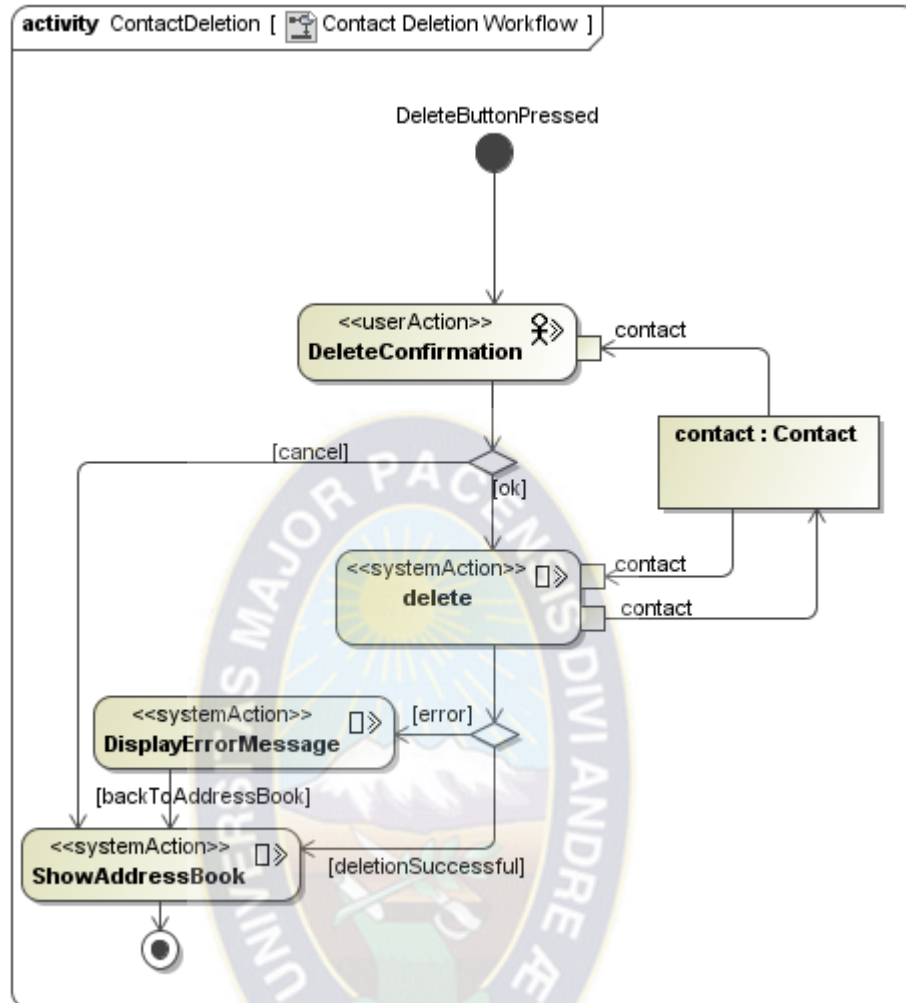


Figura 9. Diagrama de flujo de proceso de UWE
Fuente: <http://uwe.pst.ifi.lmu.de/teachingTutorialRequirementsSpanish.html>

En la figura 7 de arriba, podemos ver que la etiqueta <<userAction>> es usada para indicar las interacciones entre el usuario y la página web iniciando un proceso o respondiendo a una petición de información. Se puede ver el flujo que ocurre en cada operación con sus distintas rutas en caso de éxito en la operación o en caso de error. (München, 2015)

2.3 Comercio Electrónico

El comercio electrónico se puede definir como la compra-venta de productos y servicios a través de sistemas electrónicos, principalmente Internet. Sin

embargo, en este libro se intenta dar un enfoque mucho más amplio al término y definir aquellas herramientas electrónicas que apoyan la compraventa de productos y servicios, independientemente de dónde se realice la transacción del pago final.

La principal razón de este enfoque más amplio radica en la realidad actual de Internet, que demuestra la influencia de este medio sobre casi todos los sectores de actividad, independientemente del momento de la venta, en el cuál también nos enfocaremos de manera concreta, por supuesto. Así, en España, tres cuartas partes del PIB están influidas por Internet de manera directa o indirecta, y esta cifra es más elevada en los países más maduros en este medio como Estados Unidos, Reino Unido, Alemania o Francia.

En el mundo existen cerca de 2.300 millones de internautas, lo que representa un tercio de la población mundial. Todos y cada uno de estos internautas son potenciales clientes para el comercio electrónico, accesibles a través de diferentes herramientas de marketing online y cuyas acciones son medibles sobre una población universal, en contraposición al muestreo utilizado en otros canales masivos de comunicación para la medición. No existe, por tanto, ningún otro canal de venta con mayor potencial que el medio online.

En España, existen 28 millones de internautas conectados, lo que supone un 66% de la población. En 2011, casi el 64% de los hogares españoles tenía acceso a la Red, accediendo un 97% de ellos a través de banda ancha. Pero no sólo la banda ancha está impulsando el crecimiento de los negocios digitales, España es uno de los países de Europa con mayor penetración de la banda móvil, con casi 14 millones de líneas, superando en un 9% la media de la Unión Europea, y con un 46,5% de smartphones, liderando este mercado junto al Reino Unido. Por último destacar el crecimiento de las televisiones inteligentes como tercer dispositivo de acceso a internet (tras el PC y los smartphones/tablets). ((adigital), 2012)

2.4 Inventarios

Los Inventarios, son bienes tangibles que se tienen para la venta en el curso ordinario del negocio o para ser consumidos en la producción de bienes o servicios para su posterior comercialización. Los inventarios comprenden, además de las materias primas, productos en proceso y productos terminados o mercancías para la venta, los materiales, repuestos y accesorios para ser consumidos en la producción de bienes fabricados para la venta o en la prestación de servicios; empaques y envases y los inventarios en tránsito.

2.4.1 Sistemas de inventario

Sistema de Inventario Perpetuo, en este sistema el negocio mantiene un registro continuo para cada artículo del inventario. Los registros muestran por lo tanto el inventario disponible todo el tiempo. Los registros perpetuos son útiles para preparar los estados financieros mensuales, trimestral o provisionalmente. EL negocio puede determinar el costo del inventario final y el costo de las mercancías vendidas directamente de las cuentas sin tener que contabilizar el inventario.

El sistema perpetuo ofrece un alto grado de control, porque los registros de inventario están siempre actualizados. Anteriormente, los negocios utilizaban el sistema perpetuo principalmente para los inventarios de alto costo unitario, como las joyas y los automóviles; hoy día con este método los administradores pueden tomar mejores decisiones acerca de las cantidades a comprar, los precios a pagar por el inventario, la fijación de precios al cliente y los términos de venta a ofrecer. El conocimiento de la cantidad disponible ayuda a proteger el inventario.

Sistema de Inventario Periódico, en este sistema el negocio no mantiene un registro continuo del inventario disponible, más bien, al fin del periodo, el negocio hace un conteo físico del inventario disponible y aplica los costos unitarios para determinar el costo del inventario final. Se utiliza también para

calcular el costo de las mercancías vendidas. El sistema periódico es conocido también como sistema físico, porque se apoya en el conteo físico real del inventario. El sistema periódico es generalmente utilizado para contabilizar los artículos del inventario que tienen un costo unitario bajo. Los artículos de bajo costo pueden no ser lo suficientemente valiosos para garantizar el costo de llevar un registro al día del inventario disponible. Para usar el sistema periódico con efectividad, el propietario debe tener la capacidad de controlar el inventario mediante la inspección visual. (Ramírez, 2006)

2.5 Tecnología de software

2.5.1 Php

PHP tal y como se conoce hoy en día es en realidad el sucesor de un producto llamado PHP/FI. Creado en 1994 por Rasmus Lerdorf, la primera encarnación de PHP era un conjunto simple de ficheros binarios Common Gateway Interface (CGI) escritos en el lenguaje de programación C. Originalmente utilizado para rastrear visitas de su currículum online, llamó al conjunto de scripts "Personal Home Page Tools", más frecuentemente referenciado como "PHP Tools". Con el paso del tiempo se quiso más funcionalidad, y Rasmus reescribió PHP Tools, produciendo una implementación más grande y rica. Este nuevo modelo fue capaz de interaccionar con bases de datos, y mucho más, proporcionando un entorno de trabajo sobre cuyos usuarios podían desarrollar aplicaciones web dinámicas sencillas tales como libros de visitas. En junio de 1995. Rasmus publicó el código fuente de PHP Tools, lo que permitió a los desarrolladores usarlo como considerasen apropiado. Esto también permitió -y animó- a los usuarios a proporcionar soluciones a los errores del código, y generalmente a mejorarlo.

En septiembre de ese mismo año, Rasmus amplió PHP y -por un corto periodo de tiempo- abandonó el nombre de PHP. Ahora, refiriéndose a las herramientas como FI (abreviatura de "Forms Interpreter"), la nueva implementación incluía

algunas de las funciones básicas de PHP tal y como la conocemos hoy. Tenía variables como las de Perl, interpretación automática de variables de formulario y sintaxis incrustada HTML. La sintaxis por sí misma era similar a la de Perl, aunque mucho más limitada, simple y algo inconsistente. De hecho, para embeber el código en un fichero HTML, los desarrolladores tenían que usar comentarios de HTML. Aunque este método no era completamente bien recibido, FI continuó gozando de expansión y aceptación como una herramienta CGI, pero todavía no completamente como lenguaje. Sin embargo, esto comenzó a cambiar al mes siguiente; en octubre de 1995 Rasmus publicó una versión nueva del código. Recordando el nombre PHP, ahora era llamado (resumidamente) "Personal Home Page Construction Kit," y fue la primera versión que presumía de ser, en aquel momento, considerada como una interfaz de scripts avanzada. El lenguaje fue deliberadamente diseñado para asemejarse a C en estructura, haciéndolo una adopción sencilla para desarrolladores familiarizados con C, Perl, y lenguajes similares. Habiendo sido así bastante limitado a sistemas UNIX y compatibles con POSIX, el potencial para una implementación de Windows NT estaba siendo explorada.

El código fue completamente rehecho de nuevo, y en abril de 1996, combinando los nombres de versiones anteriores, Rasmus introdujo PHP/FI. Esta implementación de segunda generación comenzó realmente a desarrollar PHP desde un conjunto de herramientas dentro de un lenguaje de programación de derecho propio. Incluía soporte interno para DBM, mSQL, y bases de datos Postgres95, cookies, soporte para funciones definidas por el usuario, y mucho más. Ese mes de junio, PHP/FI brindó una versión 2.0. Sin embargo, un interesante hecho sobre esto, es que sólo había una única versión completa de PHP 2.0. Cuando finalmente pasó de la versión beta en noviembre de 1997, el motor de análisis subyacente ya estaba siendo reescrito por completo.

Aunque vivió una corta vida de desarrollo, continuó gozando de un crecimiento de popularidad en el aún joven mundo del desarrollo. En 1997 y 1998, PHP/FI

tenía un culto de varios miles de usuarios en todo el mundo. Una encuesta de Netcraft en mayo de 1998 indicó que cerca de 60,000 dominios reportaron que tenían cabeceras que contenían "PHP", indicando en efecto que el servidor host lo tenía instalado. Este número se correspondía con aproximadamente el 1% de todos los dominios de Internet del momento. A pesar de estas impresionantes cifras, la maduración de PHP/FI estaba condenada por limitaciones; mientras había varios contribuidores menores, aún era desarrollado principalmente por un individuo. (PHP, 2001 - 2015)

2.5.2 JavaScript

JavaScript (abreviado comúnmente "JS") es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos,³ basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente, sin acceso a funciones del servidor. JavaScript se interpreta en el agente de usuario, al

mismo tiempo que las sentencias van descargándose junto con el código HTML.

Una cuarta edición está en desarrollo e incluirá nuevas características tales como paquetes, espacio de nombres y definición explícita de clases. (W3SCHOOLS, 1999-2015)

2.5.3 JQuery

jQuery es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. Fue presentada el 14 de enero de 2006 en el BarCamp NYC. jQuery es la biblioteca de JavaScript más utilizada.

jQuery es software libre y de código abierto, posee un doble licenciamiento bajo la Licencia MIT y la Licencia Pública General de GNU v2, permitiendo su uso en proyectos libres y privativos. jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

Las empresas Microsoft y Nokia anunciaron que incluirán la biblioteca en sus plataformas. Microsoft la añadirá en su IDE Visual Studio y la usará junto con los frameworks ASP.NET AJAX y ASP.NET MVC, mientras que Nokia los integrará con su plataforma Web Run-Time. (The jQuery Foundation, 2015)

2.5.4 Html

HTML, siglas de **HyperText Markup Language** («lenguaje de marcas de hipertexto»), hace referencia al lenguaje de marcado para la elaboración de páginas web. Es un estándar que sirve de referencia para la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un

código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, etc. Es un estándar a cargo de la W3C, organización dedicada a la estandarización de casi todas las tecnologías ligadas a la web, sobre todo en lo referente a su escritura e interpretación. Es el lenguaje con el que se definen las páginas web.

El lenguaje HTML basa su filosofía de desarrollo en la referenciación. Para añadir un elemento externo a la página (imagen, vídeo, *script*, etc.), este no se incrusta directamente en el código de la página, sino que se hace una referencia a la ubicación de dicho elemento mediante texto. De este modo, la página web contiene sólo texto mientras que recae en el navegador web (interpretador del código) la tarea de unir todos los elementos y visualizar la página final. Al ser un estándar, HTML busca ser un lenguaje que permita que cualquier página web escrita en una determinada versión, pueda ser interpretada de la misma forma (estándar) por cualquier navegador web actualizado.

Sin embargo, a lo largo de sus diferentes versiones, se han incorporado y suprimido diversas características, con el fin de hacerlo más eficiente y facilitar el desarrollo de páginas web compatibles con distintos navegadores y plataformas (PC de escritorio, portátiles, teléfonos inteligentes, tabletas, etc.). Sin embargo, para interpretar correctamente una nueva versión de HTML, los desarrolladores de navegadores web deben incorporar estos cambios y el usuario debe ser capaz de usar la nueva versión del navegador con los cambios incorporados. Usualmente los cambios son aplicados mediante parches de actualización automática (Firefox, Chrome) u ofreciendo una nueva versión del navegador con todos los cambios incorporados, en un sitio web de descarga oficial (Internet Explorer). Un navegador no actualizado no será capaz de interpretar correctamente una página web escrita en una versión de HTML superior a la que pueda interpretar, lo que obliga muchas veces a los desarrolladores a aplicar técnicas y cambios que permitan corregir problemas de visualización e incluso de interpretación de código HTML. Así mismo, las

páginas escritas en una versión anterior de HTML deberían ser actualizadas o reescritas, lo que no siempre se cumple. Es por ello que ciertos navegadores aún mantienen la capacidad de interpretar páginas web de versiones HTML anteriores. Por estas razones, aún existen diferencias entre distintos navegadores y versiones al interpretar una misma página web.

2.5.5 Html5

HTML5 (*HyperText Markup Language*, versión 5) es la quinta revisión importante del lenguaje básico de la World Wide Web, HTML. HTML5 especifica dos variantes de sintaxis para HTML:(text/html), la variante conocida como *HTML5* y una variante XHTML conocida como sintaxis *XHTML5* que deberá ser servida como XML. Esta es la primera vez que HTML y XHTML se han desarrollado en paralelo.

Todavía se encuentra en modo experimental, lo cual indica la misma W3C, aunque ya es usado por múltiples desarrolladores web por sus avances, mejoras y ventajas.

Al no ser reconocido en viejas versiones de navegadores por sus nuevas etiquetas, se recomienda al usuario común actualizar a la versión más nueva, para poder disfrutar de todo el potencial que provee HTML5.

2.5.6 Css

Las hojas de estilo en cascada o (Cascading Style Sheets, o sus siglas CSS) hacen referencia a un lenguaje de hojas de estilos usado para describir la presentación semántica (el aspecto y formato) de un documento escrito en lenguaje de marcas. Su aplicación más común es dar estilo a páginas webs escritas en lenguaje HTML y XHTML, pero también puede ser aplicado a cualquier tipo de documentos XML, incluyendo SVG y XUL.

La información de estilo puede ser adjuntada como un documento separado o en el mismo documento HTML. En este último caso podrían definirse estilos

generales en la cabecera del documento o en cada etiqueta particular mediante el atributo "<style>". (Bos, Bert, 1994–2015)

2.5.7 Responsive Web Design

El **diseño web adaptable** o adaptativo (en inglés, *Responsive Web Design*) es una técnica de diseño y desarrollo web que mediante el uso de estructuras e imágenes fluidas, así como de media-queries en la hoja de estilo CSS, consigue adaptar el sitio web al entorno del usuario.

El diseñador y autor norteamericano Ethan Marcotte creó y difundió esta técnica a partir de una serie de artículos en *A List Apart*, una publicación en línea especializada en diseño y desarrollo web, idea que luego extendería en su libro *Responsive Web Design*. (Refsnes Data, 1999-2015)

2.5.8 MySQL

MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y los derechos de autor del código están en poder del autor individual, MySQL es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones

contratan trabajadores alrededor del mundo que colaboran vía Internet. MySQL AB fue fundado por David Axmark, Allan Larsson y Michael Widenius. (ShareAlike, 1999-2014)

2.6 Calidad de software

La calidad del software es una preocupación a la que se dedican muchos esfuerzos. Sin embargo, el software casi nunca es perfecto. Todo proyecto tiene como objetivo producir software de la mejor calidad posible, que cumpla, y si puede supere las expectativas de los usuarios.

- Es la aptitud de un producto o servicio para satisfacer las necesidades del usuario.
- Es la cualidad de todos los productos, no solamente de equipos sino también de programas.

En el desarrollo de software, la calidad de diseño acompaña a la calidad de los requisitos, especificaciones y diseño del sistema. La calidad de concordancia es un aspecto centrado principalmente en la implementación; Si la implementación sigue al diseño, y el sistema resultante cumple con los objetivos de requisitos y de rendimiento, la calidad de concordancia es alta.

2.6.1 Calidad de Software WEB SITE QEM

La estrategia propuesta, denominada Metodología de Evaluación de Calidad de Sitios Web (o, en inglés, Web-site Quality Evaluation Method, o, metodología Web-site QEM), pretende realizar un aporte ingenieril al proponer un enfoque sistemático, disciplinado y cuantitativo que se adecue a la evaluación, comparación y análisis de calidad de sistemas de información centrados en la Web (más o menos complejos).

Web-site QEM, incluye un conjunto de fases, actividades, productos, modelos y constructores de proceso que introduciremos seguidamente. Una de las metas principales de la evaluación y comparación de calidad de artefactos Web, radica

en comprender el grado de cumplimiento de un conjunto de características y sub características con respecto a los requerimientos de calidad establecidos.

Con respecto a la fase de Planificación y Programación de la Evaluación de Calidad, la misma contiene actividades y procedimientos de soporte, con el fin de determinar objetivos estratégicos, tácticos y operativos.

Considerando a la fase de Definición y Especificación de Requerimientos de Calidad, la misma trata con actividades y modelos para la elicitación, determinación, análisis y especificación de los requerimientos.

Con respecto a la fase de Definición e Implementación de la Evaluación Elemental la misma trata con actividades, modelos, técnicas y herramientas para determinar métricas y criterios de evaluación para cada atributo cuantificable.

Considerando a la fase Definición e Implementación de la Evaluación Global la misma trata con actividades, modelos, y herramientas para determinar los criterios de agregación de las preferencias de calidad elemental para producir la preferencia global, para cada sistema seleccionado.

Con respecto a la fase de Análisis de Resultados, Conclusiones y Documentación la misma trata con actividades de análisis y comparación de las preferencias de calidad elementales, parciales y globales, y, asimismo, la justificación de los resultados.

Por último, la Validación de las métricas es un proceso importante en la disciplina de evaluación de software. Podemos definirla como el proceso de asegurar que las medidas sean una caracterización numérica apropiada del atributo, mostrando que se satisfaga la condición de representación.

2.7 Estudio de costos y beneficios

2.7.1 Cocomo

El Modelo Constructivo de Costos (o COCOMO, por su acrónimo del inglés COConstructive COst MOdel) es un modelo matemático de base empírica utilizado para estimación de costos de software. Incluye tres submodelos, cada uno ofrece un nivel de detalle y aproximación, cada vez mayor, a medida que avanza el proceso de desarrollo del software: básico, intermedio y detallado.

Este modelo fue desarrollado por Barry W. Boehm a finales de los años 70 y comienzos de los 80, exponiéndolo detalladamente en su libro "Software Engineering Economics". (Nancy Merlo- Shett, 2002-2009)



3.1 Introducción

En este capítulo se realizara el análisis y diseño del sistema basándose en la metodología AUP (Proceso Unificado Ágil) contando con:

Sus diferentes fases: Inicio, Elaboración, Construcción y Transición.

Todo esto siendo modelado y representado por el Modelo UWE (Herramienta de modelado basado en UML), tomando los siguientes modelos:

- ✓ Diagrama de Casos de uso
- ✓ Diagrama de Clases
- ✓ Diagrama de Procesos
- ✓ Diagrama de Flujo de Procesos
- ✓ Diagrama de Navegación
- ✓ Diagrama de Presentación

Y basados en los elementos que recomienda el “Libro Blanco del Comercio electrónico”:

- ✓ Catálogo de productos.
- ✓ Carrito de Compras
- ✓ Proceso de registro
- ✓ Proceso de Venta
- ✓ Motor Interno de búsqueda
- ✓ Gestión de Stock

Tomando en cuenta estas tres herramientas y representándolas en el cuadro de fases del AUP se tiene:

	FASES			
DISCIPLINAS	Inicio	Elaboración	Construcción	Transición
Modelado	UWE Definición Actores	UWE (Casos de Uso, Casos de uso de alto Nivel y Casos de uso Extendidos)	UWE (Diagrama de Navegación)	
Implementación	Modelo del Negocio (UWE Casos de uso)	UWE(Diagrama de Clases, Diagrama de	UWE (Modelo Entidad Relación, Diccionario de	

		Procesos, Diagrama de Flujo de Procesos)	Datos)	
Pruebas				
Despliegue			UWE (Diagrama de Presentación)	Manual de usuario
Adm. de la Configuración				
Adm. de Proyecto				
Entorno				

Tabla 5. Fases del AUP
Fuente: Elaboración propia

3.2 Situación actual de la empresa

Actualmente la empresa “FIBBY’s S.R.L.” importa y produce productos del área textil, estos productos se comercializan de forma tradicional en las sucursales de la empresa. Los registros de ventas e inventario actualmente se llevan en documentos Excel.

Las entregas de los productos a domicilio solo se hace a los clientes mayoristas, también se realizan envíos al interior del país. Estos se coordinan vía telefónica y/o correo electrónico.

3.3 Fase de Inicio

3.3.1 Disciplina de Modelado

El objetivo de esta disciplina es entender el negocio de la organización, el problema de dominio que se abordan en el proyecto, para lo cual identificaremos los actores y el modelo del negocio.

3.3.1.1 Definición de Actores

Actor	Descripción
Cliente Minorista	Es el cliente común, este se registra al momento de realizar el primer pedido.
Cliente Mayorista	Este cliente hace pedidos constantes por cantidades altas, es necesario que sea registrado por un trabajador de la empresa.
Encargado de Ventas	Es el encargado de registrar a los clientes mayoristas, y de verificar los pagos para su posterior envío o entrega en su respectiva sucursal.
Encargado de Almacén	Es el encargado de registrar los productos al almacén de la empresa.
Administrador	Es el encargado del control de usuarios del sistema. Además de contar con acceso a los reportes que genera el sistema.

Tabla 6. Definición de actores
Fuente: Elaboración propia

3.3.2 Disciplina de Implementación

3.3.2.1 Modelo del negocio

En el modelo del negocio podemos identificar como se desarrolla las actividades en el área de ventas y almacén de la empresa “FIBBY’s S.R.L.”, todo el registro se desarrolla de forma manual en libros. Este proceso se desarrolla en un tiempo excesivo, además de perder información en el mismo.

3.3.3 Disciplina de Pruebas

La disciplina de pruebas consta de pruebas que se van haciendo constantemente al producto, verificando si cumple o no con los requerimientos, todo este proceso se encuentra en el apartado Calidad de Software, del **Capítulo IV**.

3.3.4 Disciplina de Despliegue

La disciplina de despliegue en la fase inicial estima el costo y tiempo de entrega del proyecto, por lo cual toda la información se encuentra en el apartado **Evaluación de costo** del **Capítulo IV**.

3.3.5 Disciplina de Administración de la Configuración

Esta disciplina establece los frameworks, lenguajes, bases de datos y demás herramientas a utilizarse en el proyecto. Además de definirse la estructura del proyecto y un reposito. Toda esta información ya se encuentra reflejada en el Capítulo I.

3.3.6 Disciplina de Administración de Proyecto

Esta disciplina se centra en la administración del equipo de desarrollo, en este caso el sistema fue realizado solo por una persona y sin contar con un pago

formal al ser un proyecto de grado, con lo cual el único documento generado es el cronograma que se encuentra en el apartado de anexos.

3.3.7 Disciplina de Entorno

Esta disciplina está relacionada con el entorno del proyecto (cantidad de personas necesarias, licencias de software), esto se encuentra definido en el apartado costo del proyecto en el capítulo V. En cuanto a las licencias de software se utilizó software libre para realización de este proyecto por lo cual no se tiene más inconvenientes.

3.4 Fase de Elaboración

El principal objetivo de la fase de elaboración es probar la arquitectura del sistema a desarrollar, por lo que describiremos las diferentes soluciones que da el sistema al proceso de ventas y almacenamiento de la empresa. Esta descripción será representada por diagramas de UWE.

3.4.1 Disciplina de Modelado

3.4.1.1 Diagramas de Casos de Uso

En este diagrama se representará la interacción de los diferentes actores con el sistema basado en el nuevo modelo de negocio propuesto.

3.4.1.2 Descripción de Los casos de Uso de Alto Nivel

3.4.1.2.1 Caso de Uso: Acceso al sistema

En el siguiente Caso de Uso de alto nivel se especifica el proceso de ingreso al sistema.

CASO DE USO	Logeo de usuario
ACTORES	Cliente Minorista, Cliente Mayorista, Encargado de ventas, Encargado de

	Almacén Y Administrador
TIPO	Primario, Esencial
DESCRIPCION	Este proceso de Logueo de usuario consiste en la validación de los códigos de usuario y sus respectivas contraseñas para el acceso a las demás opciones del sistema, estas dependen del tipo de usuario.

Tabla 7. Caso de uso de alto nivel
Fuente: Elaboración propia

3.4.1.2.2 Caso de Uso: Registro de pedido

En el siguiente Caso de Uso de alto nivel se especifica el registro del pedido o venta a un cliente, para su posterior pago y entrega.

CASO DE USO	Registro de pedido
ACTORES	Cliente Minorista, Cliente Mayorista
TIPO	Primario, Esencial
DESCRIPCION	Este proceso consiste en el registro de diferentes productos y cantidades en forma de pedido. Este proceso cuenta con una búsqueda, y un carrito de compras en el catalogo en línea donde se adicionan o eliminan los productos a ser solicitados. Además de identificar el medio de pago y lugar de entrega del pedido. La diferencia entre el cliente mayorista y minorista reside en el costo de cada producto y la cantidad mínima del pedido.

Tabla 8. Registro de pedido
Fuente: Elaboración propia

3.4.1.2.3 Caso de Uso: Registro de cliente minorista al sistema

En el siguiente Caso de Uso de alto nivel se especifica el registro de un cliente para acceder al sistema.

CASO DE USO	Registro de cliente minorista al sistema
ACTORES	Cliente Minorista
TIPO	Primario
DESCRIPCION	En este proceso el cliente minorista introduce sus datos personales para su validación y posterior registro. Este paso habilita al cliente para poder registrar pedidos.

Tabla 9. Registro de cliente minorista al sistema
Fuente: Elaboración propia

3.4.1.2.4 Caso de Uso: Cotización de confección

En el siguiente Caso de Uso de alto nivel se especifica el registro de una cotización de producción realizada por un cliente.

CASO DE USO	Cotización de confección
ACTORES	Cliente Minorista, Cliente Mayorista
TIPO	Secundario
DESCRIPCION	El proceso de Cotización de Confección, se realiza la cotización de los diferentes productos que puede confeccionar la empresa tipos, modelos, colores, materias primas y acabadas pueden ser elegidos. Una vez registrada la cotización se le asigna un código único con el que el

	cliente puede concluir el proceso de adquisición.
--	---

Tabla 10. Cotización de Confección
Fuente: Elaboración propia

3.4.1.2.5 Caso de Uso: Validación de pago por venta en pagina web

En el siguiente Caso de Uso de alto nivel se especifica la validación de pago de los pedidos de clientes realizados por el sistema, para su posterior entrega.

CASO DE USO	Validación de pago por venta online
ACTORES	Encargado de Ventas
TIPO	Primario
DESCRIPCION	El proceso de validación verifica los pagos hechos en la cuenta TIGO Money de la empresa, cuentas bancarias o en alguna sucursal, para su posterior entrega física en la dirección elegida por el cliente previa verificación con los pedidos registrados en la base de datos del sistema y actualización de estos.

Tabla 11. Validación de pago por venta en página web
Fuente: Elaboración propia

3.4.1.2.6 Caso de Uso: Venta en sucursal

En el siguiente Caso de Uso de alto nivel se especifica el registro de una venta realizada en instalaciones de la empresa, registrando cantidades y precios.

CASO DE USO	Venta en sucursal
ACTORES	Encargado de Ventas

TIPO	Primario
DESCRIPCION	El proceso de venta en sucursal registra la venta de productos y todos los datos del mismo, precios, cantidades, sucursal en la que se realizo la venta y la actualización del stock.

Tabla 12. Venta en sucursal
Fuente: Elaboración propia

3.4.1.2.7 Caso de Uso: Registro de productos

En el siguiente Caso de Uso de alto nivel se especifica el registro de nuevos productos y todas sus características en el sistema.

CASO DE USO	Registro de productos
ACTORES	Encargado de Almacén
TIPO	Primario
DESCRIPCION	El proceso de registro de productos registra productos y sus características, fotografías, precios de adquisición y venta, además de asignarle un código de barras único para cada producto.

Tabla 13. Registro de productos
Fuente: Elaboración propia

3.4.1.2.8 Caso de Uso: Registro de Ingresos de productos a Almacén

En el siguiente Caso de Uso de alto nivel se especifica el registro de ingreso de productos al almacén de la empresa.

CASO DE USO	Registro de Ingresos de productos a Almacén
ACTORES	Encargado de Almacén
TIPO	Primario
DESCRIPCIÓN	El proceso de registro de productos al

	almacén de la empresa, actualiza el stock del producto ingresado en la base de datos del sistema. Detallando fechas y ubicaciones.
--	--

Tabla 14. Registro de ingresos de productos a almacén
Fuente: Elaboración propia

3.4.1.2.9 Caso de Uso: Registro usuarios del sistema

En el siguiente Caso de Uso de alto nivel se especifica el registro de nuevos usuarios al sistema, además de asignarles un rol de usuario.

CASO DE USO	Registro usuarios del sistema
ACTORES	Administrador
TIPO	Primario, Esencial
DESCRIPCION	El proceso de registro de nuevos usuarios al sistema, registra la información, valida los datos, asigna un código de usuario, contraseña y un rol de usuario (Encargado de almacén, Encargado de Ventas y Cliente Mayorista).

Tabla 15. Registro usuarios del sistema
Fuente: Elaboración propia

3.4.1.3 Diagramas de Casos de Uso Expandido

3.4.1.3.1 Diagrama de Caso de Uso Expandido: Registro de pedido

Para el registro de un pedido los actores protagonistas de este proceso son el cliente mayorista y minorista, los cuales ingresan al sistema con su respectivo rol, seleccionando del catalogo productos al carrito de compras, en el cual pueden adicionar, sustraer o modificar cantidades y productos, una vez concluida la selección se procede al registro del pedido solicitando los datos de: forma de pago y dirección de entrega. Para registrar el pedido es necesario el registro del cliente.

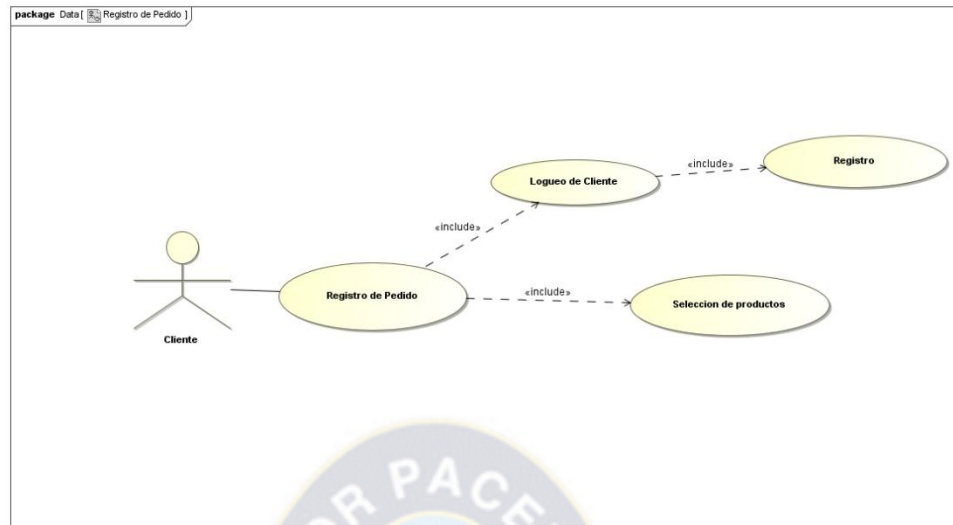


Figura 10. Registro de pedido
Fuente: Elaboración propia

3.4.1.3.2 Diagrama de Caso de Uso Expandido: Registro de cliente minorista al sistema

Para el registro de un cliente minorista es el mismo usuario el encargado de registrar sus datos como ser: nombre completo, número de cuenta TIGO Money, C.I. y dirección.

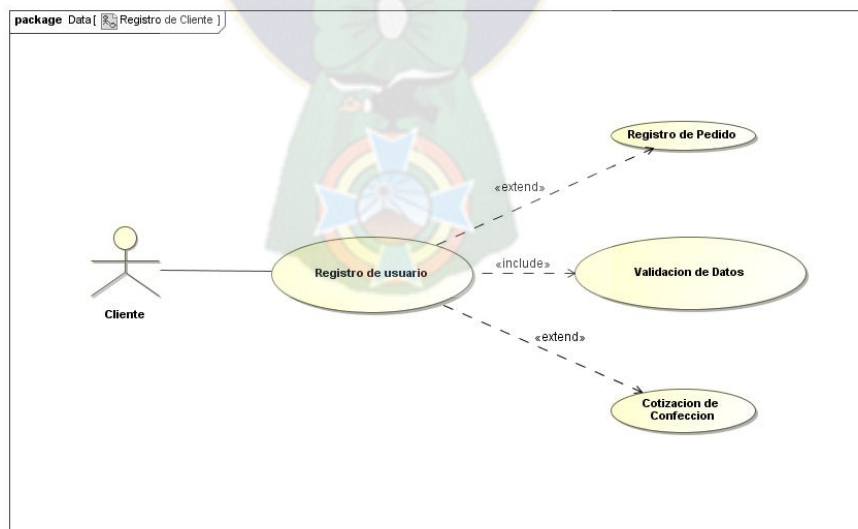


Figura 11. Registro de cliente minorista al sistema
Fuente: Elaboración propia

3.4.1.3.3 Diagrama de Caso de Uso Expandido: Cotización de confección

Para el registro de una cotización de confección los clientes debe seleccionar un producto del catalogo de productos, e identificar los detalles del pedido: cantidades, acabados, materias primas e insumos. Para realizar la cotización es necesario que el cliente se encuentre registrado.

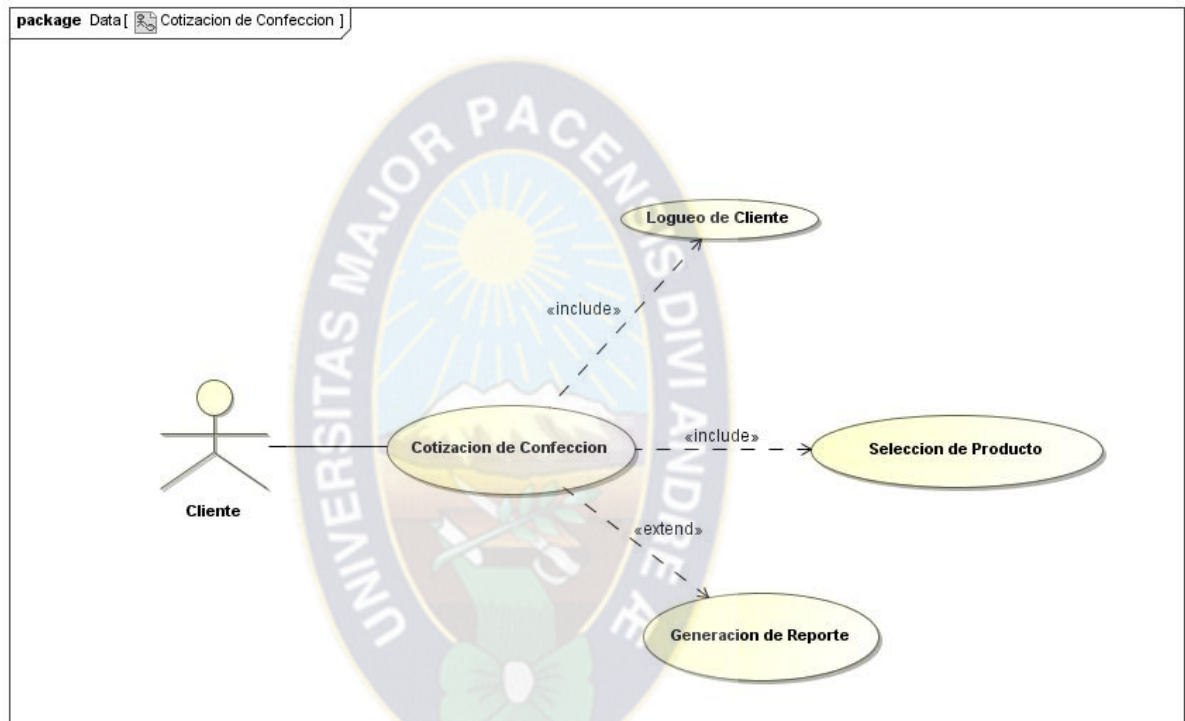


Figura 12. Cotización de confección
Fuente: Elaboración propia

3.4.1.3.4 Diagrama de Caso de Uso Expandido: Validación de pago por pedido en pagina web

Para la validación de un pago por pedido el actor principal es el encargado de ventas, este verifica los pagos hechos a la cuenta TIGO Money y cuenta bancaria de la empresa, posterior a esto acepta el pedido en el sistema y procede al envío de los productos a la dirección brindada por el cliente.

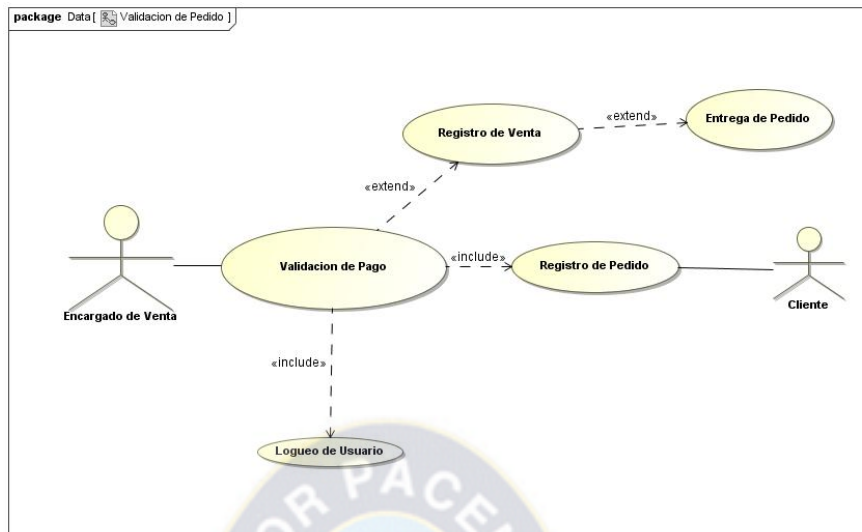


Figura 13. Validación de pago por pedido en página web
Fuente: Elaboración propia

3.4.1.3.5 Diagrama de Caso de Uso Expandido: Venta en sucursal

Para la venta en sucursal el actor principal es el encargado de ventas, el cual va registrando los productos seleccionados por el cliente en base al código único de los productos, todos estos datos se registran en la tabla **venta**, modificando automáticamente el stock de los productos.

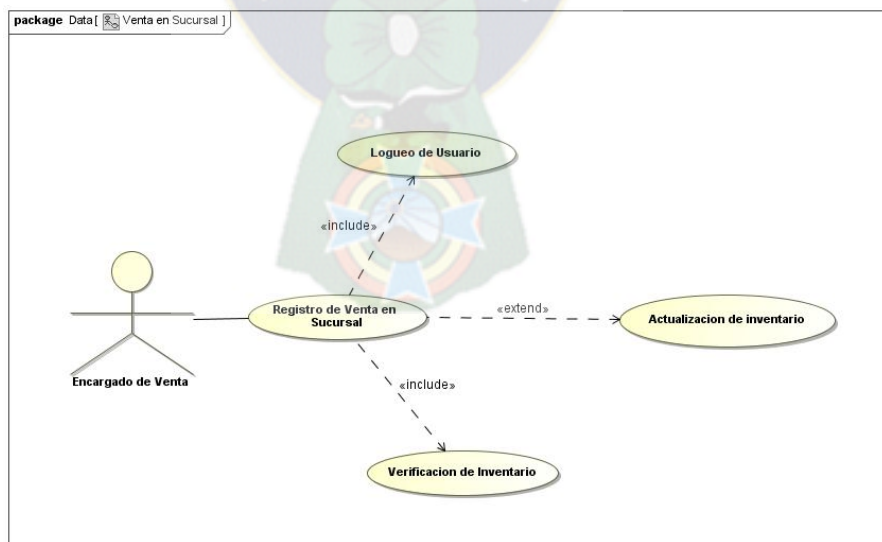


Figura 14. Venta en sucursal
Fuente: Elaboración propia

3.4.1.3.6 Diagrama de Caso de Uso Expandido: Registro de productos

Para el registro de productos nuevos el actor principal es el encargado de almacenes, el cual va registrando los datos del producto: código, marca, colores, precio por mayor y menor, y todos los datos necesarios para el catalogo de productos, estos datos se van registrando en la tabla de productos.

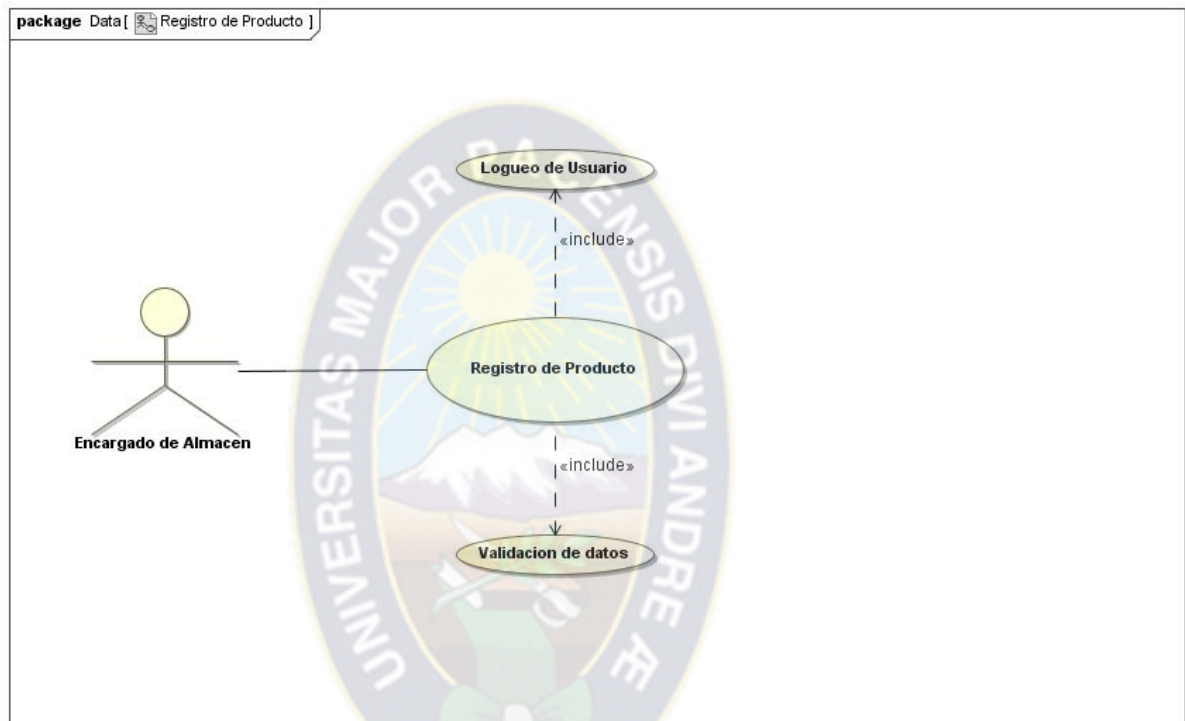


Figura 15. Registro de productos
Fuente: Elaboración propia

3.4.1.3.7 Diagrama de Caso de Uso Expandido: Registro de ingresos y movimientos de mercadería

Para el registro de ingreso de productos al Inventario el actor principal es el encargado de almacenes el cual se encarga de registrar la cantidad de productos que ingresan o se mueven de una sucursal a otra, para esto se utilizan las tablas inventario e historial de inventario.

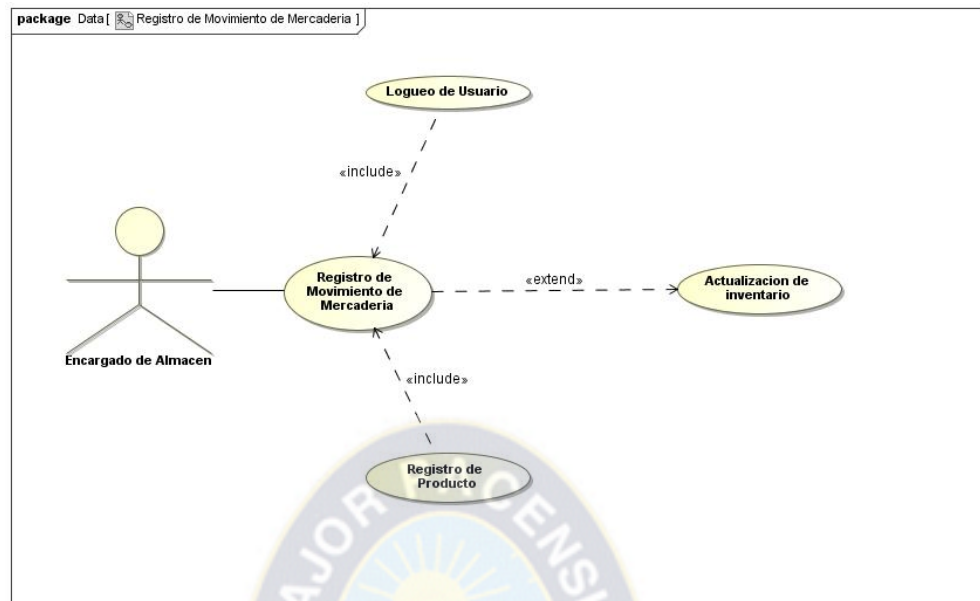


Figura 16. Registro de ingresos y movimientos de mercadería
Fuente: Elaboración propia

3.4.1.3.8 Diagrama de Caso de Uso Expandido: Registro usuarios del sistema

Para el registro de usuarios al sistema el actor principal es el Administrador, el se encarga de llenar un formulario con los datos del nuevo usuario, darle un rol y habilitar o en algún caso deshabilitar la cuenta del nuevo usuario.

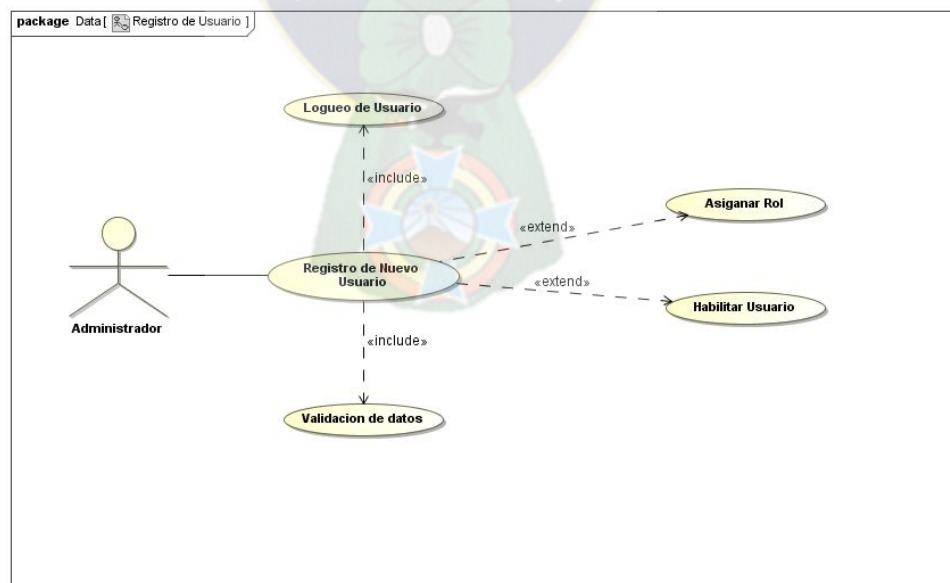


Figura 17. Registro usuarios del sistema
Fuente: Elaboración propia

3.4.2 Disciplina de Implementación

3.4.2.1 Diagrama de Clases

En el diagrama de clases se puede observar el modelo conceptual del sistema a implementar, en esta se encuentran las clases y las relaciones entre las mismas, obtenidas después del análisis del modelo del negocio.

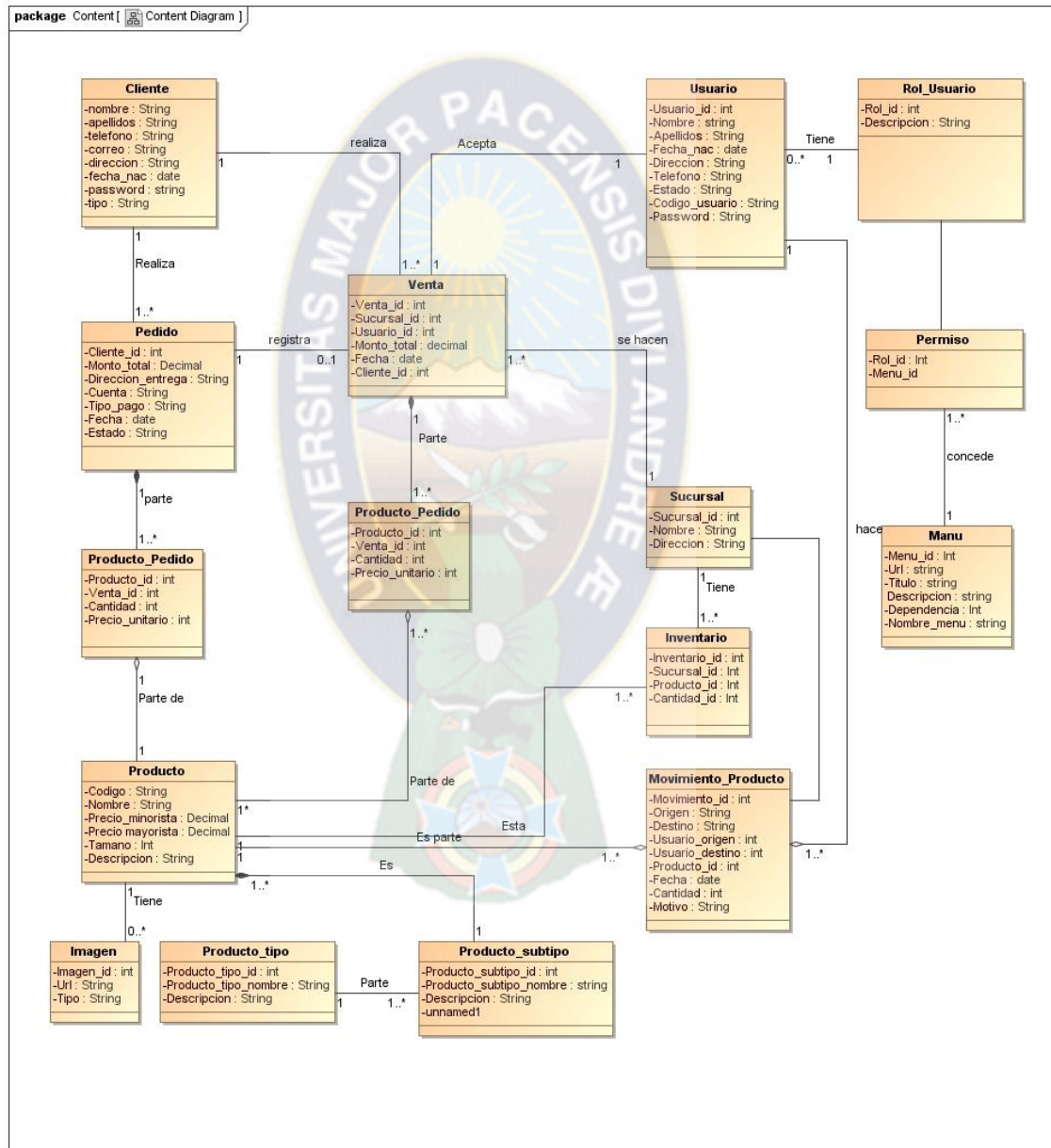


Figura 18. Diagrama de Clases del Sistema
Fuente: Elaboración propia

3.4.2.2 Modelo de Procesos

Este modelo muestra las acciones de nuestro proceso de clases y se divide en dos partes:

- ✓ Modelo de estructura del proceso.
- ✓ Modelo de flujo de procesos.

3.4.2.2.1 Diagrama de estructura del proceso

En el diagrama de Estructura de Procesos podemos ver el modelo conceptual del sistema a implementar, se puede observar la descripción jerárquica de todas las clases que se encontraron en el análisis del sistema.

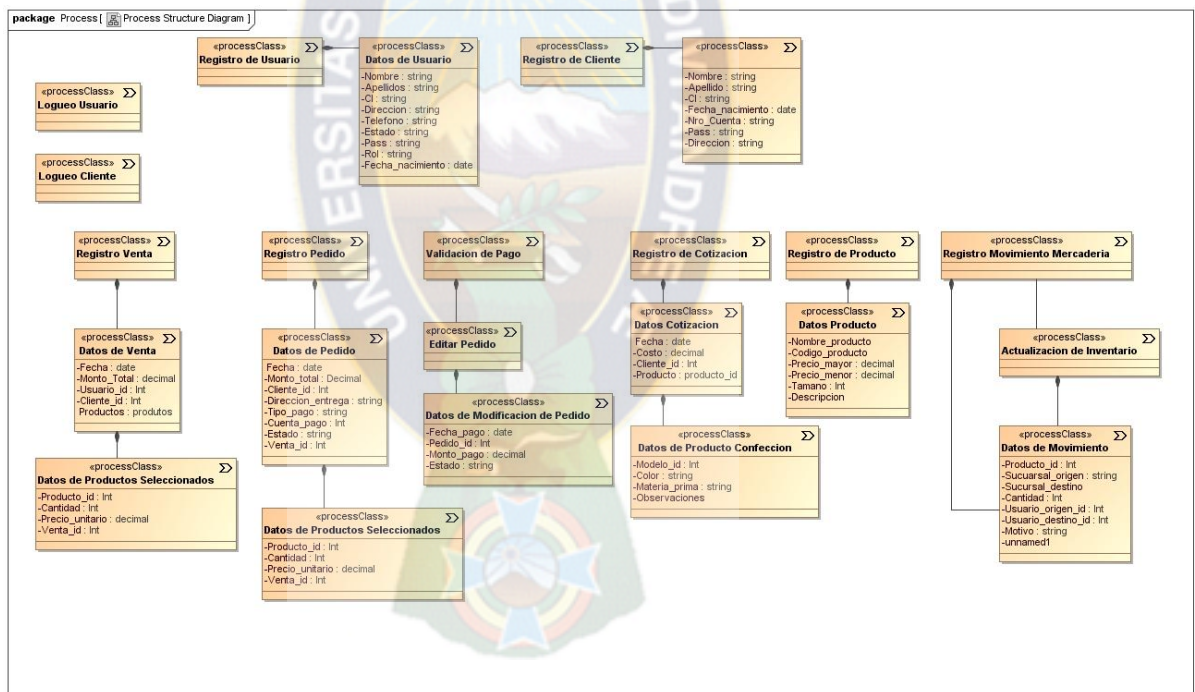


Figura 19. Diagrama de estructura del proceso
Fuente: Elaboración propia

3.4.2.2.2 Diagrama de Flujo de Procesos

a) Logueo de Usuario y Cliente

En el proceso de Logueo, es necesario contar con una cuenta de usuario o de cliente con su rol correspondiente. La contraseña y los datos se

encuentran en la base de datos, los cuales se verifican antes de permitir el acceso al sistema.

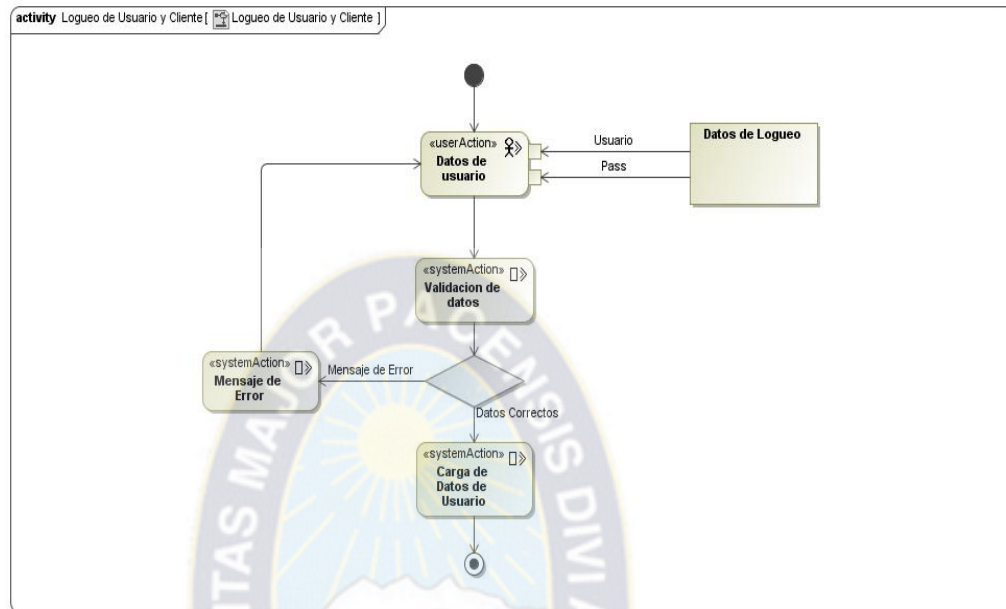


Figura 20. Logueo de Usuario y Cliente
Fuente: Elaboración propia

b) Registro de Pedido

En el proceso de pedido para la selección de productos no es necesario estar registrado o logueado. Sin embargo al momento de realizar el registro es necesario estar registrado o hacerlo en ese momento para posteriormente loguearse al sistema.

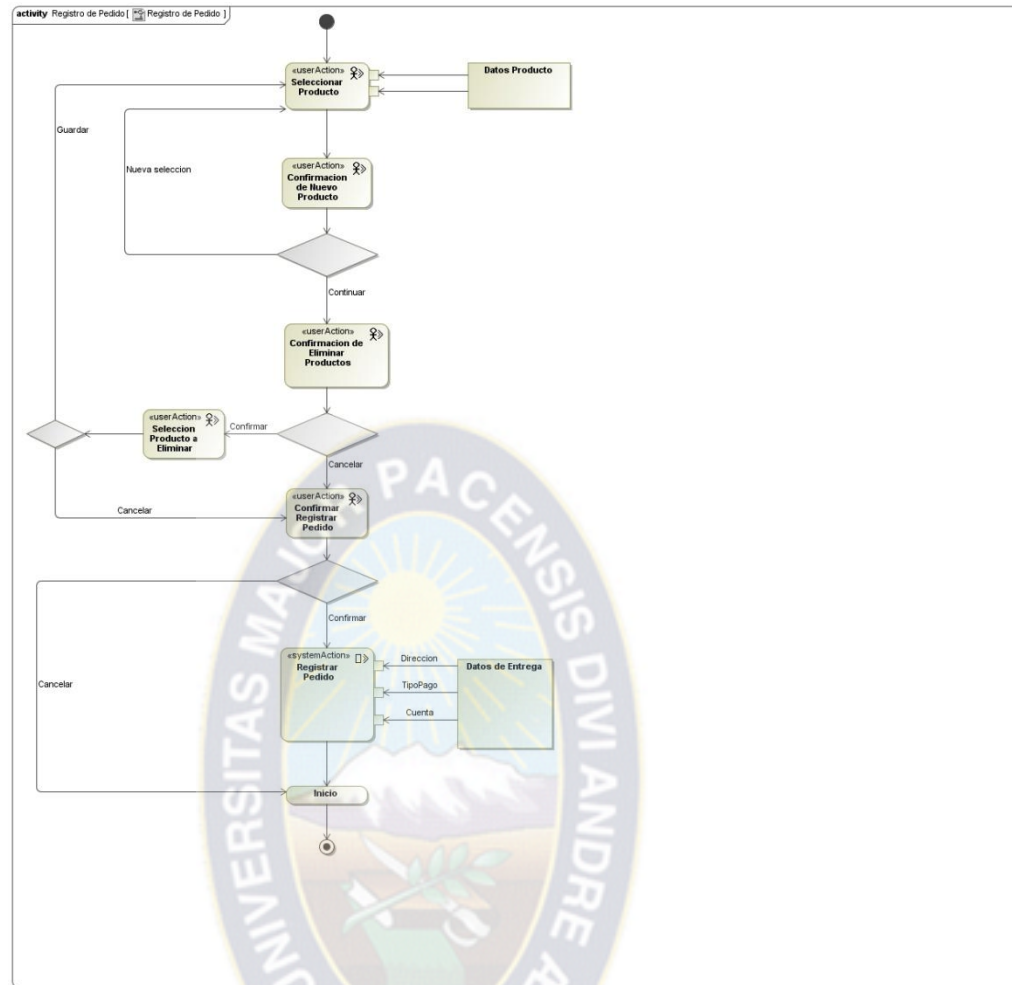


Figura 21. Registro de Pedido
Fuente: Elaboración propia

c) Registro de Cotización

En el procesos de Cotización para la selección de productos no es necesario estar registrado o logueado. Sin embargo al momento de realizar el registro es necesario estar registrado o hacerlo en ese momento para posteriormente loguearse al sistema.

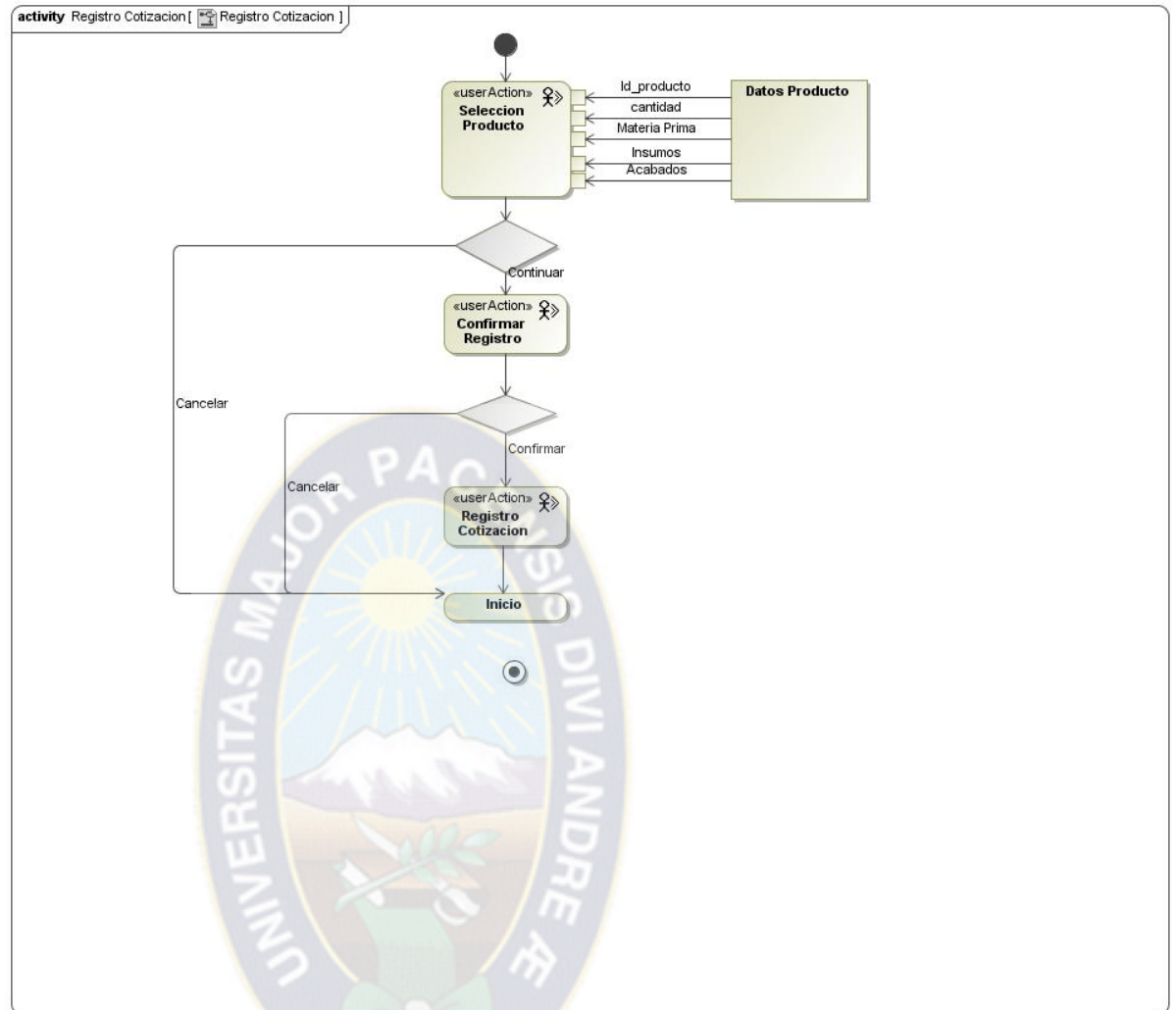


Figura 22. Registro de cotización
Fuente: Elaboración propia

d) Registro de Venta en Sucursal

En el proceso de venta en sucursal es necesario introducir los códigos de los productos de la venta, además del CI del cliente a ser atendido.

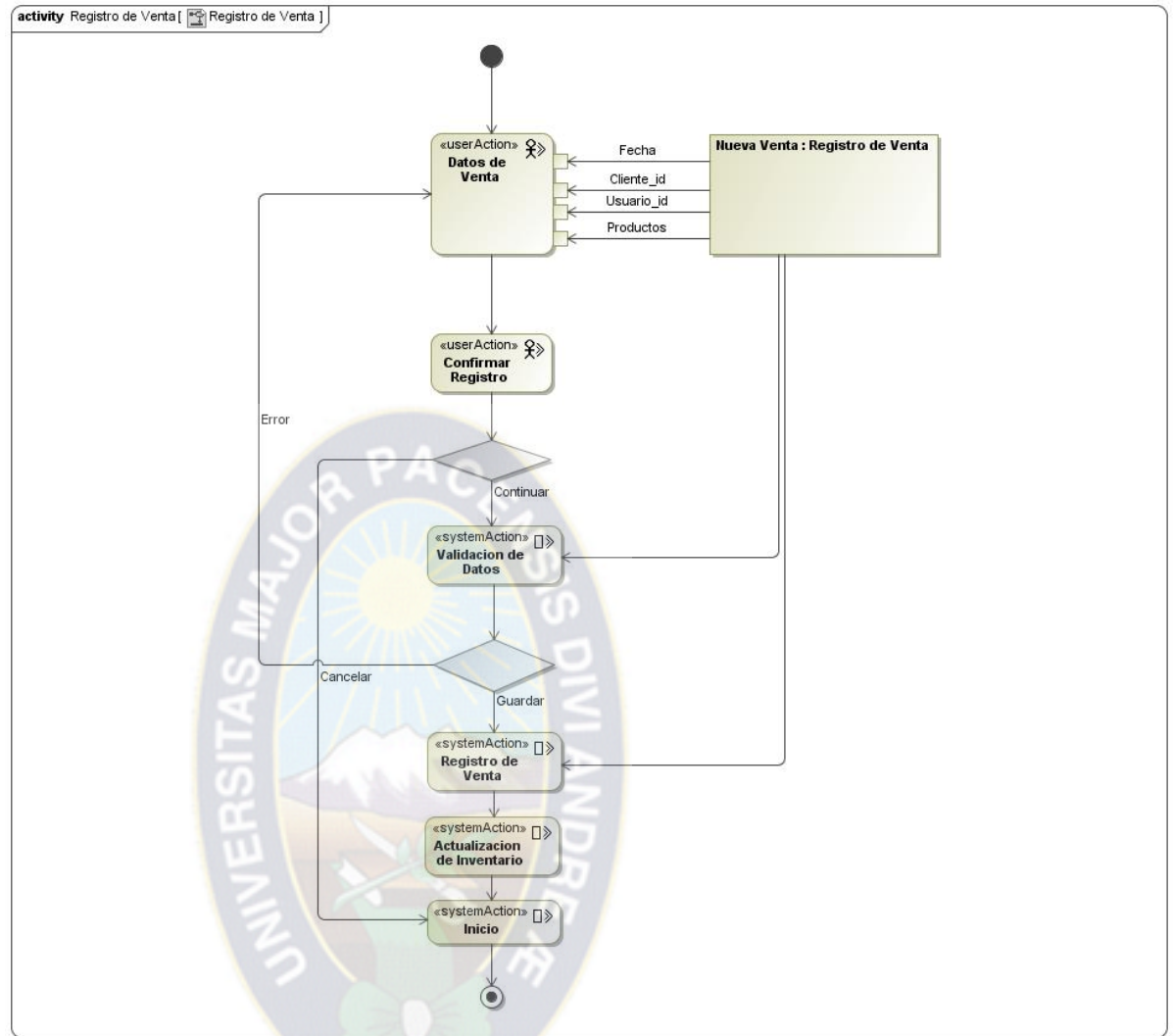


Figura 23. Registro de venta en sucursal
Fuente: Elaboración propia

e) Validación de Pedido

En el proceso de validación y entrega de Pedido el encargado de ventas verifica los pagos realizados a las cuentas de la empresa, posteriormente se realiza la búsqueda del pedido a ser atendido y hace efectiva la venta.

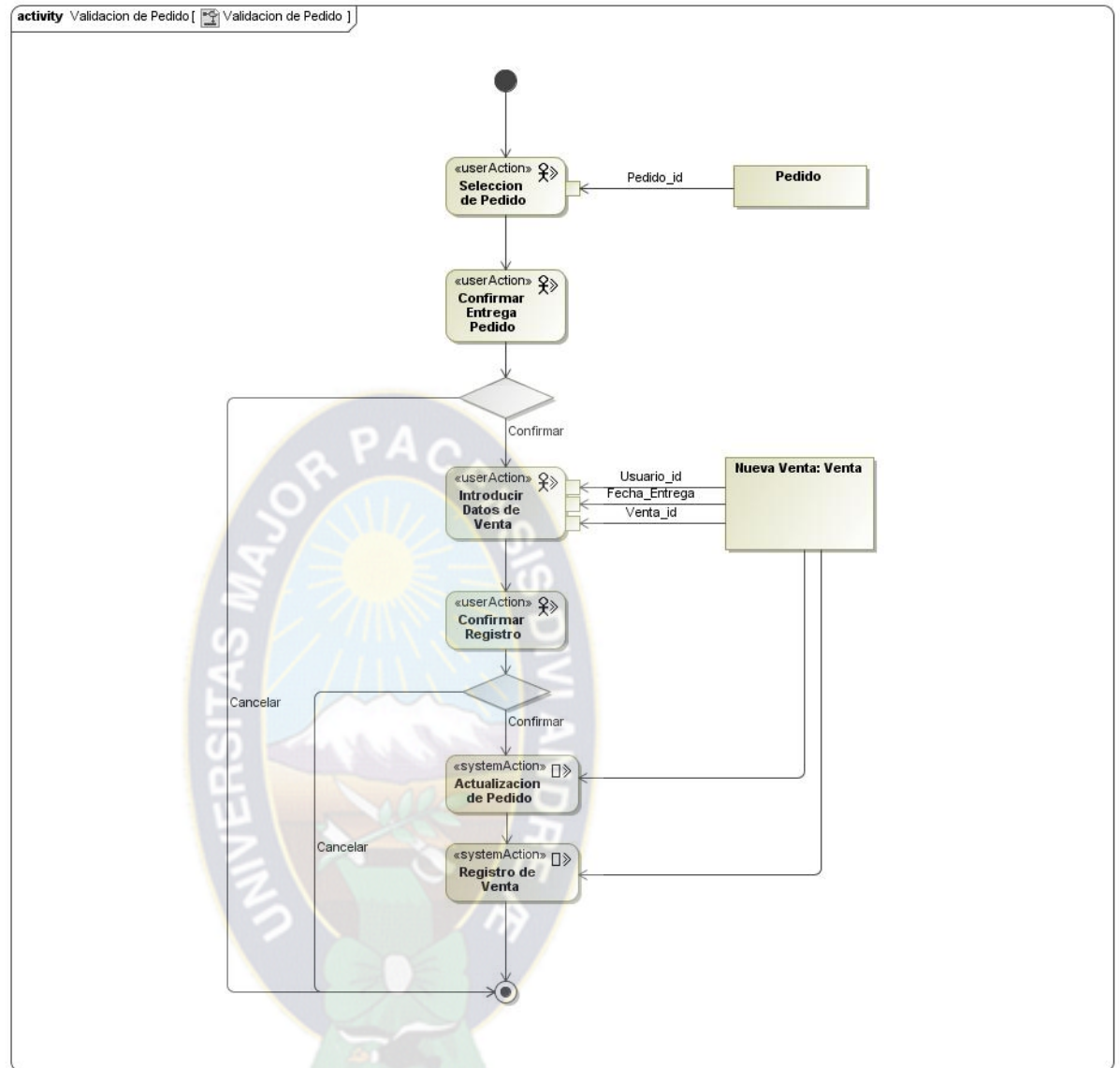


Figura 24. Validación de pedido
Fuente: Elaboración propia

f) Registro de Producto

En el Proceso de Registro de Producto, es necesario contar con todos los datos del nuevo producto.

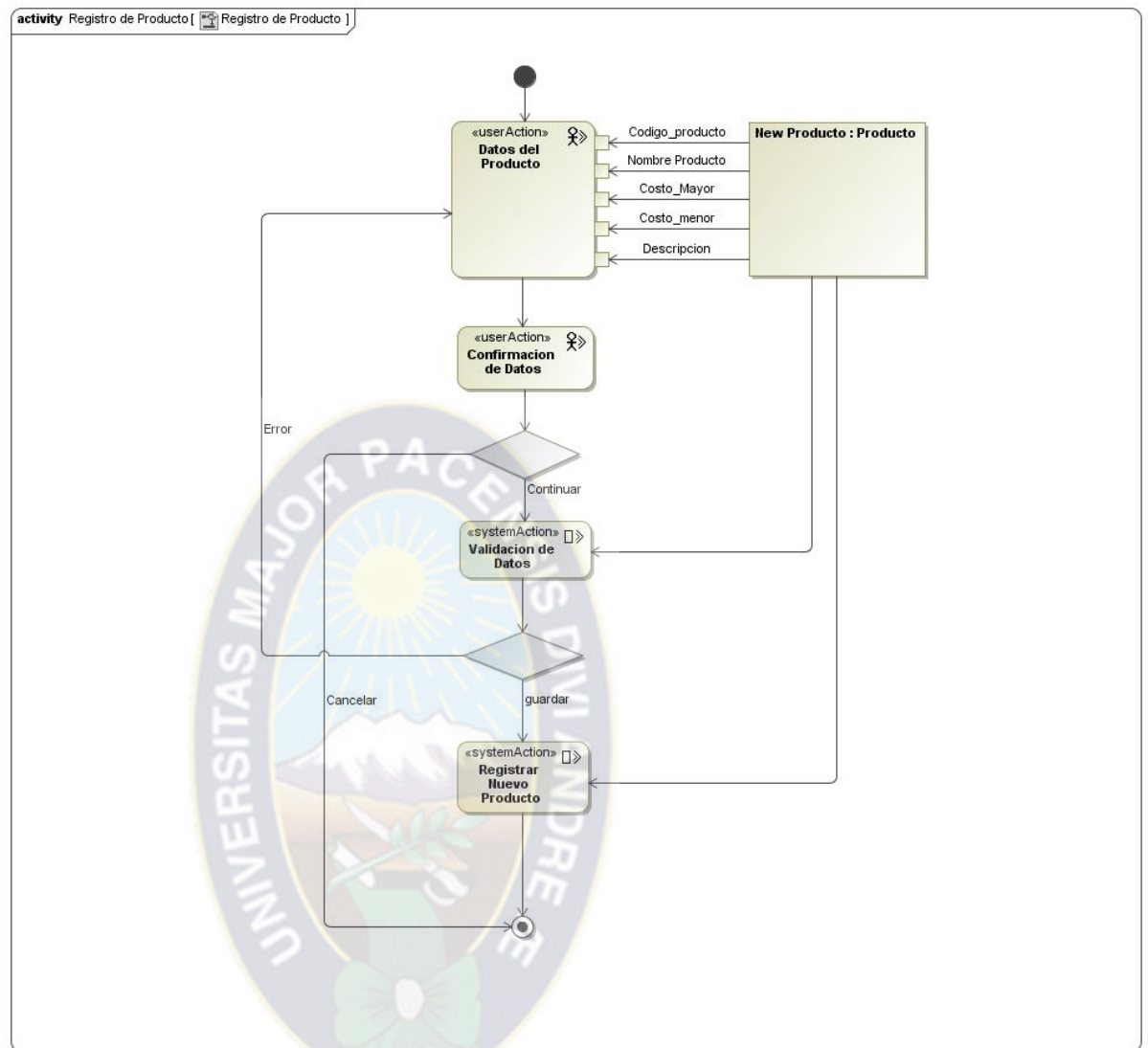


Figura 25. Registro de producto
Fuente: Elaboración propia

g) Registro de Movimiento de Mercadería

En el proceso de Registro de Mercadería, es necesario tener todos los datos del movimiento de la mercadería y las sucursales involucradas en el proceso.

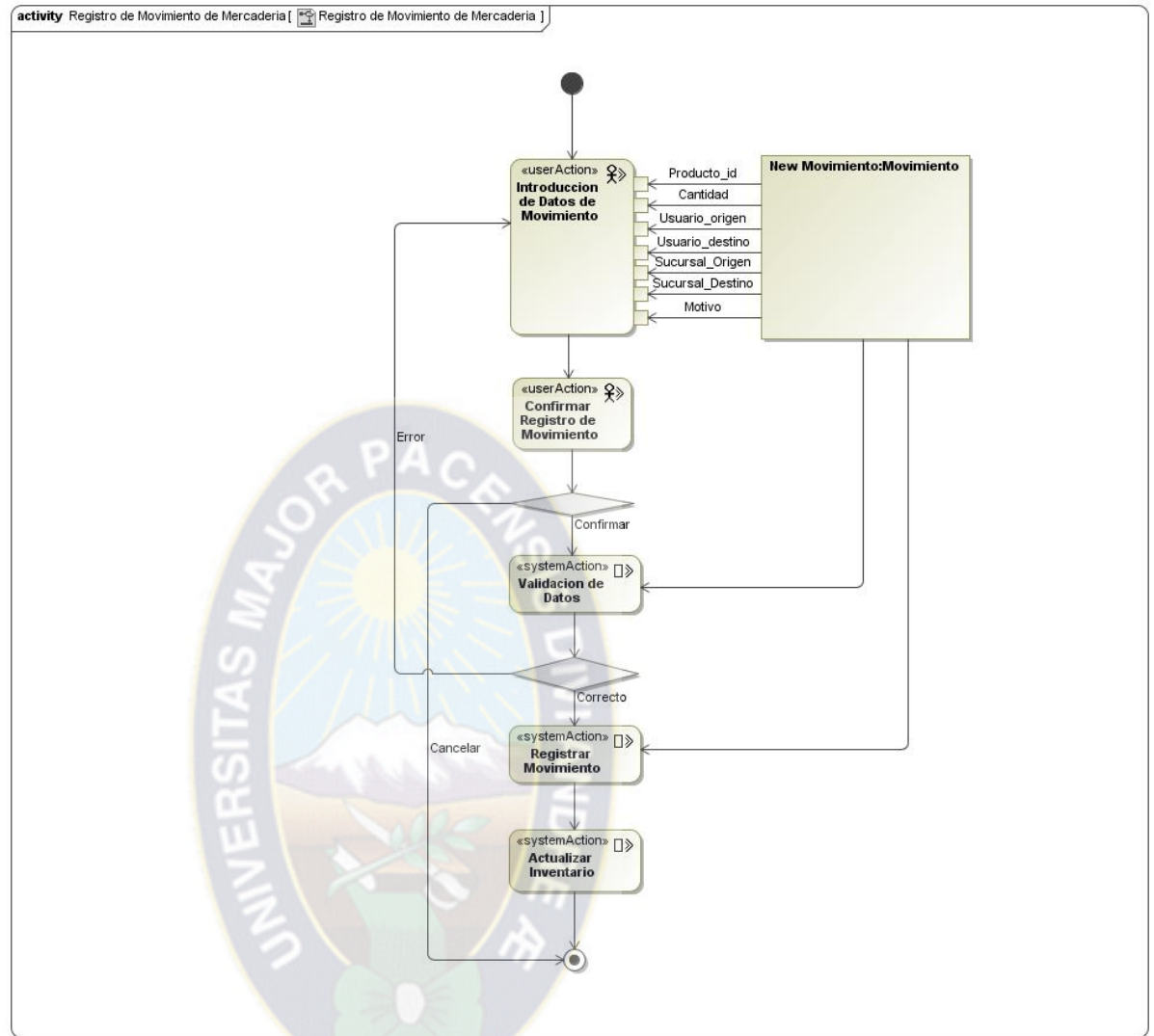


Figura 26. Registro de movimiento de mercadería
Fuente: Elaboración propia

h) Registro de Usuario

En el proceso de registro de usuario, es necesario contar con los datos del nuevo usuario además del rol a cumplir.

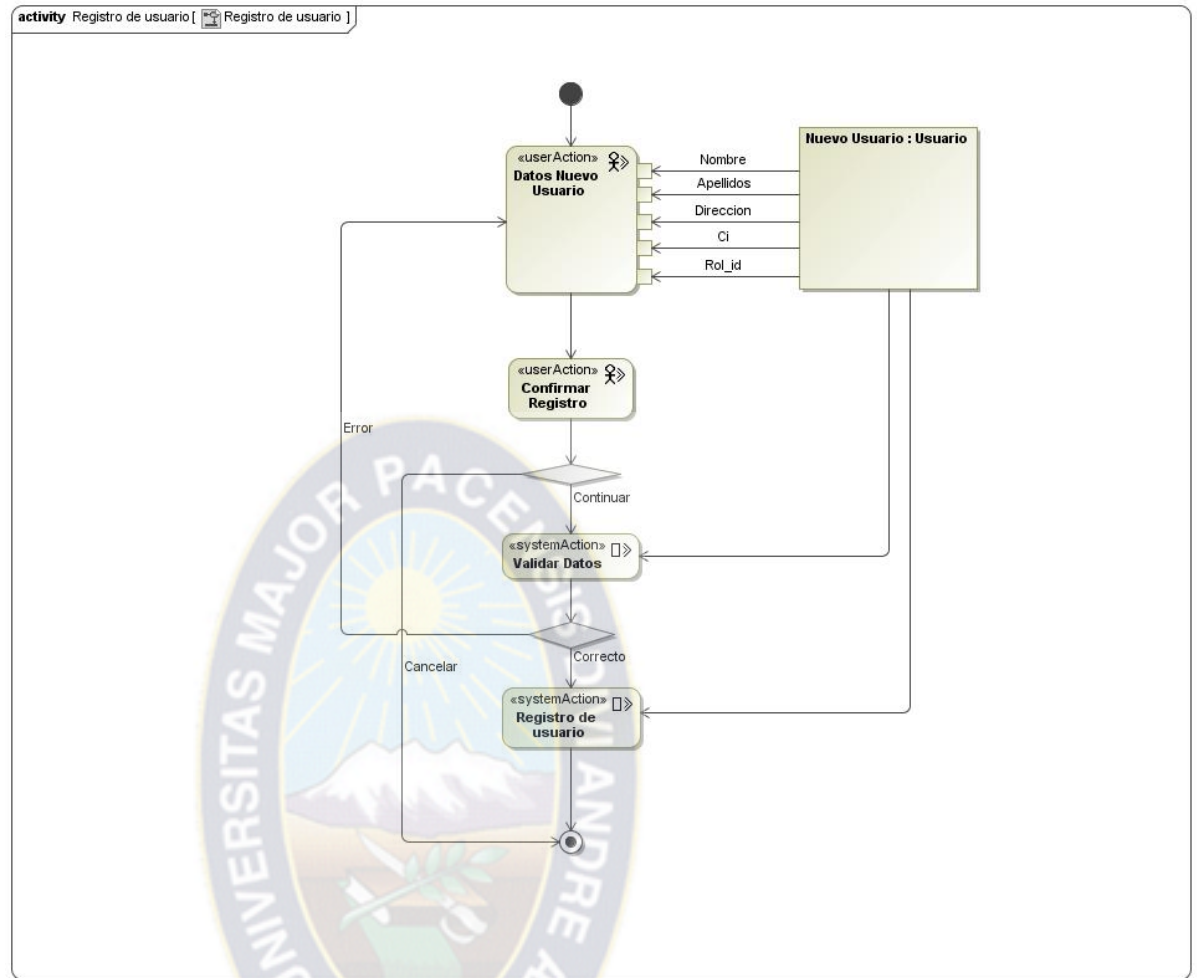


Figura 27. Registro de usuario
Fuente: Elaboración propia

3.5 Fase de Construcción

3.5.1 Disciplina de Modelado

3.5.1.1 Diagrama de Navegación

3.5.1.1.1 Diagramas de navegación cliente mayorista o minorista

En el siguiente diagrama de navegación se muestra detalladamente todos enlaces a los que podrá acceder el cliente mayorista y el cliente minorista.

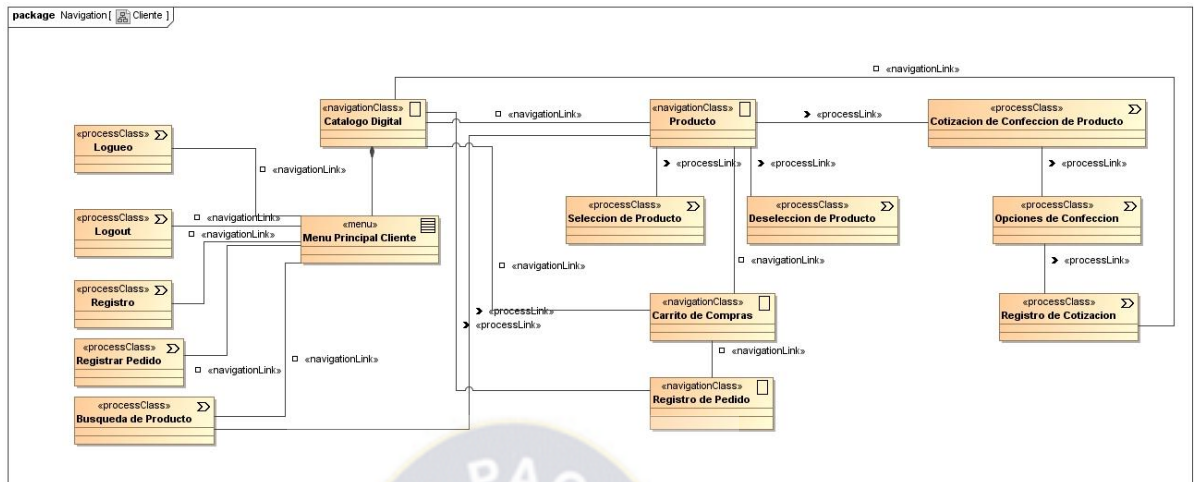


Figura 28. Diagramas de navegación cliente mayorista o minorista
Fuente: Elaboración propia

3.5.1.1.2 Diagramas de navegación: Encargado de ventas

En el siguiente diagrama de navegación se muestra detalladamente todos enlaces a los que podrá acceder el encargado de ventas.

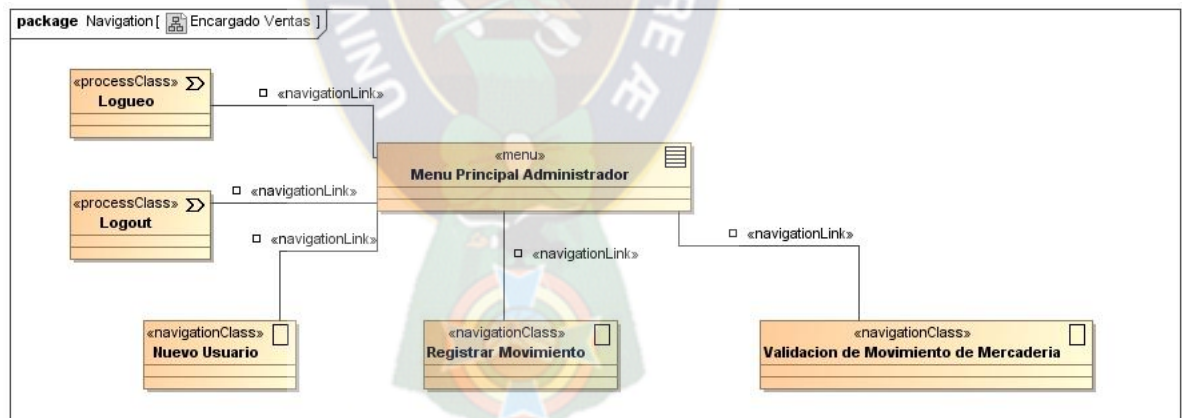


Figura 29. Encargado de ventas
Fuente: Elaboración propia

3.5.1.1.3 Diagramas de navegación: Encargado de almacén

En el siguiente diagrama de navegación se muestra detalladamente todos enlaces a los que podrá acceder el encargado de almacén.

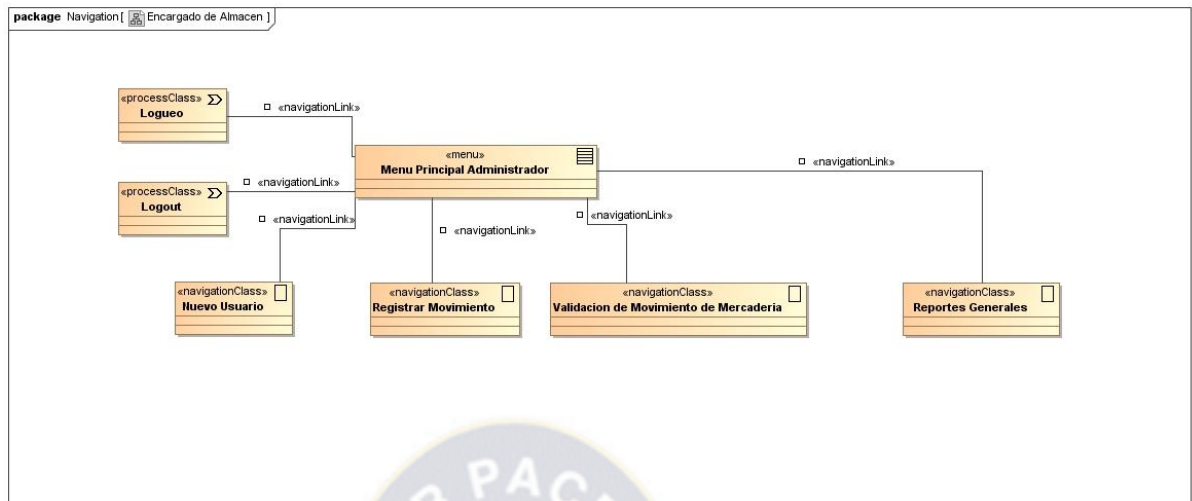


Figura 30. Encargado de almacén
Fuente: Elaboración propia

3.5.1.1.4 Diagramas de navegación: Administrador

En el siguiente diagrama de navegación se muestra detalladamente todos enlaces a los que podrá acceder el administrador.

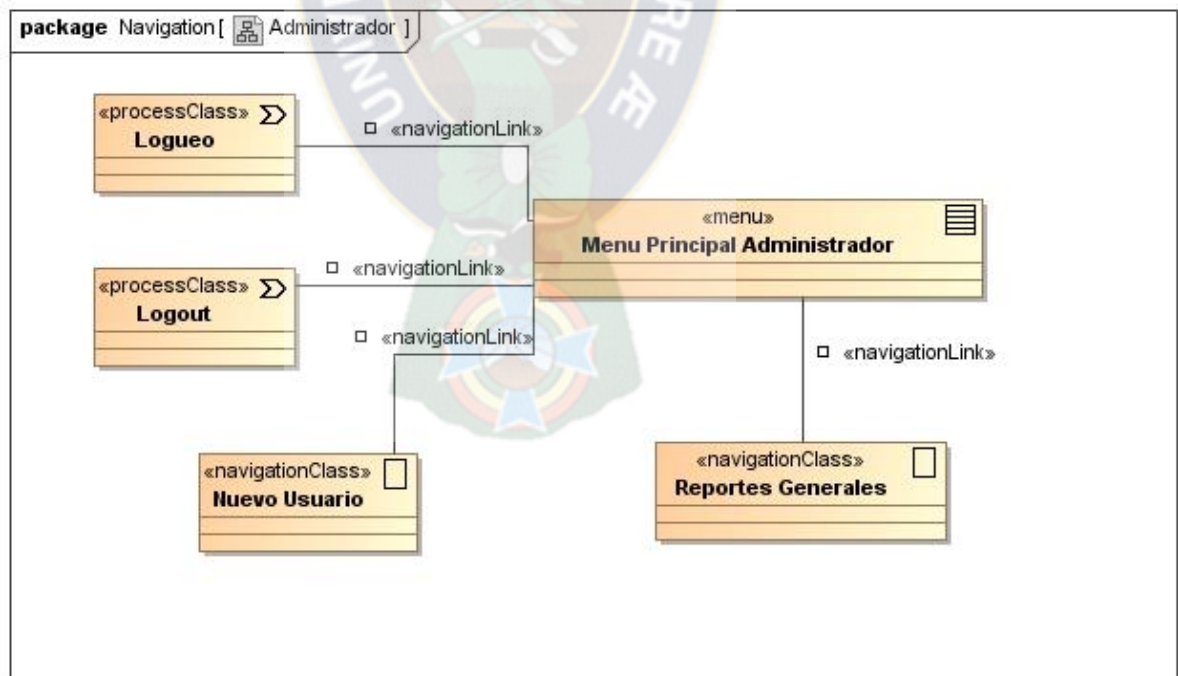


Figura 31. Administrador
Fuente: Elaboración propia

3.5.2 Disciplina de Implementación

3.5.2.1 Modelo Entidad Relación

A continuación se muestra el diagrama entidad relación del sistema, con todas las entidades y relaciones obtenidas a partir del modelo de negocio.

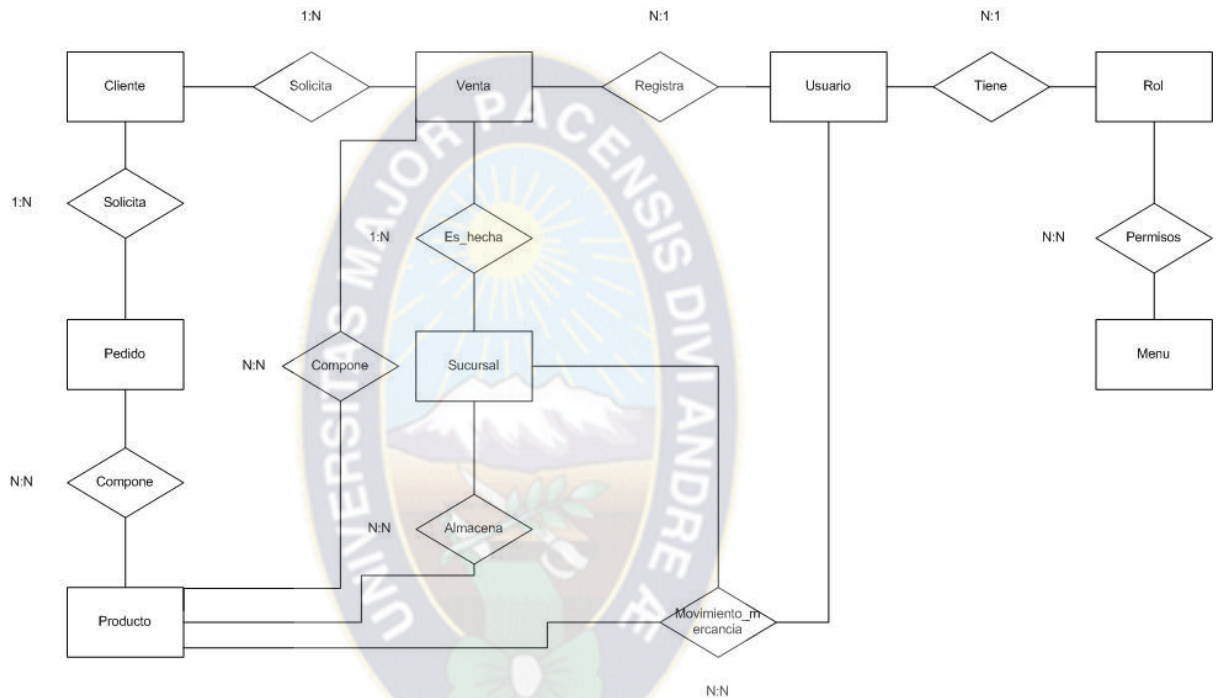


Figura 32. Modelo entidad relación
Fuente: Elaboración propia

3.5.2.2 Diccionario de Datos

Cliente

Columna	Tipo	Nulo	Predeterminado	Comentarios
cliente_id	int(11)	No		
nombre	varchar(30)	No		
apellidos	varchar(30)	No		
ci_cliente	varchar(15)	No		
telf_cuenta_tigo	int(11)	No		
correo_electronico	varchar(30)	Sí	NULL	
pass	varchar(50)	No		

tipo	int(11)	No
-------------	---------	----

Tabla 16. Diccionario de datos cliente
Fuente: Elaboración propia

Color

Columna	Tipo	Nulo	Predeterminado	Comentarios
color_id	int(11)	No		
nombre	varchar(30)	No		

Tabla 17. Diccionario de datos color
Fuente: Elaboración propia

Inventario

Columna	Tipo	Nulo	Predeterminado	Comentarios
inventario_id	int(11)	No		
sucursal_id	int(11)	No		
producto_id	int(11)	No		
cantidad	int(11)	No		

Tabla 18. Diccionario de datos inventario
Fuente: Elaboración propia

Marca

Columna	Tipo	Nulo	Predeterminado	Comentarios
marca_id	int(11)	No		
marca_nombre	varchar(30)	No		

Tabla 19. Diccionario de datos marca
Fuente: Elaboración propia

Movimiento_mercaderia

Columna	Tipo	Nulo	Predeterminado	Comentarios
movimiento_mercaderia_id	int(11)	No		
producto_id	int(11)	No		
origen	int(30)	No		
destino	int(30)	No		
usuario_origen_id	int(11)	No		
usuario_destino_id	int(11)	No		
cantidad	int(11)	No		
motivo	varchar(50)	No		
fecha_envio	datetime	No		
fecha_recepcion	datetime	No		

fecha_verificacion	datetime	No
estado	varchar(50)	No
observacion_origen	text	No
observacion_destino	text	No

Tabla 20. Diccionario de datos movimiento de mercadería
Fuente: Elaboración propia

Pedido

Columna	Tipo	Nulo	Predeterminado	Comentarios
pedido_id	int(11)	No		
cliente_id	int(11)	No		
monto_total	int(11)	No		
direccion_entrega	varchar(100)	No		
cuenta_asociada	varchar(30)	No		
tipo_pago	varchar(30)	No		
pedido_fecha	date	No		
venta_id	int(11)	No		
estado	varchar(30)	No		
nit_factura	varchar(20)	No		
nombre_factura	varchar(50)	No		
nota	varchar(300)	No		
coordenadas	varchar(70)	No		

Tabla 21. Diccionario de datos pedido
Fuente: Elaboración propia

Pedido_producto

Columna	Tipo	Nulo	Predeterminado	Comentarios
pedido_producto_id	int(11)	No		
pedido_id	int(11)	No		
producto_id	int(11)	No		
cantidad	int(11)	No		
precio_unitario	decimal(10,0)	No		

Tabla 22. Diccionario de datos pedido producto
Fuente: Elaboración propia

Producto

Columna	Tipo	Nulo	Predeterminado	Comentarios
producto_id	int(11)	No		
producto_codigo	varchar(50)	No		
nombre	varchar(50)	No		
precio_compra	decimal(10,0)	No		
precio_minorista	decimal(10,0)	No		

precio_mayorista	decimal(10,0)	No
tamano	int(11)	No
descripcion	text	No
marca_id	int(11)	No
producto_subtipo_id	int(11)	No

Tabla 23. Diccionario de datos producto
Fuente: Elaboración propia

Producto_color

Columna	Tipo	Nulo	Predeterminado	Comentarios
producto_id	int(11)	No		
color_id	int(11)	No		

Tabla 24. Diccionario de datos producto color
Fuente: Elaboración propia

Producto_imagen

Columna	Tipo	Nulo	Predeterminado	Comentarios
producto_imagen_id	int(11)	No		
url	varchar(50)	No		
url_miniatra	varchar(100)	No		
url_icono	varchar(100)	No		
tipo	tinyint(1)	No		
producto_id	int(11)	No		

Tabla 25. Diccionario de datos producto imagen
Fuente: Elaboración propia

Producto_subtipo

Columna	Tipo	Nul o	Predeterminad o	Comentario s
producto_subtipo_id	int(11)	No		
producto_subtipo_nombre	varchar(30)	No		
descripcion	varchar(50)	Sí	NULL	
producto_tipo_id	int(11)	No		

Tabla 26. Diccionario de datos producto subtipo
Fuente: Elaboración propia

Producto_tipo

Columna	Tipo	Nul o	Predeterminad o	Comentario s
producto_tipo_id	int(11)	No		
producto_tipo_nombre	varchar(30)	No		

e				
descripcion	varchar(100)	Sí	NULL	

Tabla 27. Diccionario de datos producto tipo
Fuente: Elaboración propia

Producto_tipo_cliente

Columna	Tipo	Nulo	Predeterminado	Comentarios
producto_id	int(11)	No		
tipo_cliente_id	int(11)	No		

Tabla 28. Diccionario de datos producto tipo cliente
Fuente: Elaboración propia

Rol

Columna	Tipo	Nulo	Predeterminado	Comentarios
rol_id	int(11)	No		
nombre	varchar(30)	No		

Tabla 29. Diccionario de datos rol
Fuente: Elaboración propia

Sucursal

Columna	Tipo	Nulo	Predeterminado	Comentarios
sucursal_id	int(11)	No		
sucursal_nombre	varchar(30)	No		
direccion	varchar(60)	No		
departamento	varchar(20)	No		

Tabla 30. Diccionario de datos Sucursal
Fuente: Elaboración propia

Tipo_cliente

Columna	Tipo	Nulo	Predeterminado	Comentarios
tipo_cliente_id	int(11)	No		
descripcion	varchar(20)	No		

Tabla 31. Diccionario de datos Tipo cliente
Fuente: Elaboración propia

Usuario

Columna	Tipo	Nulo	Predeterminado	Comentarios
usuario_id	int(11)	No		
nombre	varchar(50)	No		
apellidos	varchar(50)	No		
codigo_usuario	varchar(50)	No		

pass	varchar(50)	No
rol_id	int(11)	No
direccion	varchar(50)	No
ci	varchar(25)	No
telefono	varchar(50)	No
estado	tinyint(1)	No
fecha_nacimiento	date	No
sucursal_id	int(11)	No

Tabla 32. Diccionario de datos usuario
Fuente: Elaboración propia

Venta

Columna	Tipo	Nulo	Predeterminado	Comentarios
venta_id	int(11)	No		
cliente_id	int(11)	No		
usuario_id	int(11)	No		
fecha	datetime	No		
monto_total	int(11)	No		
sucursal_id	int(11)	No		
tipo_venta	int(11)	No		

Tabla 33. Diccionario de datos venta
Fuente: Elaboración propia

Venta_producto

Columna	Tipo	Nulo	Predeterminado	Comentarios
venta_producto_id	int(11)	No		
venta_id	int(11)	No		
producto_id	int(11)	No		
precio_unitario	int(11)	No		
cantidad	int(11)	No		

Tabla 34. Diccionario venta producto
Fuente: Elaboración propia

3.5.3 Disciplina de Despliegue

3.5.3.1 Diagrama de Presentación

3.5.3.1.1 Diagrama de Presentación: Pantalla de Logueo

En este diagrama es un diseño del acceso al sistema para los diferentes tipos de usuario y clientes, posterior a ser validado se permite el acceso con su determinado rol.

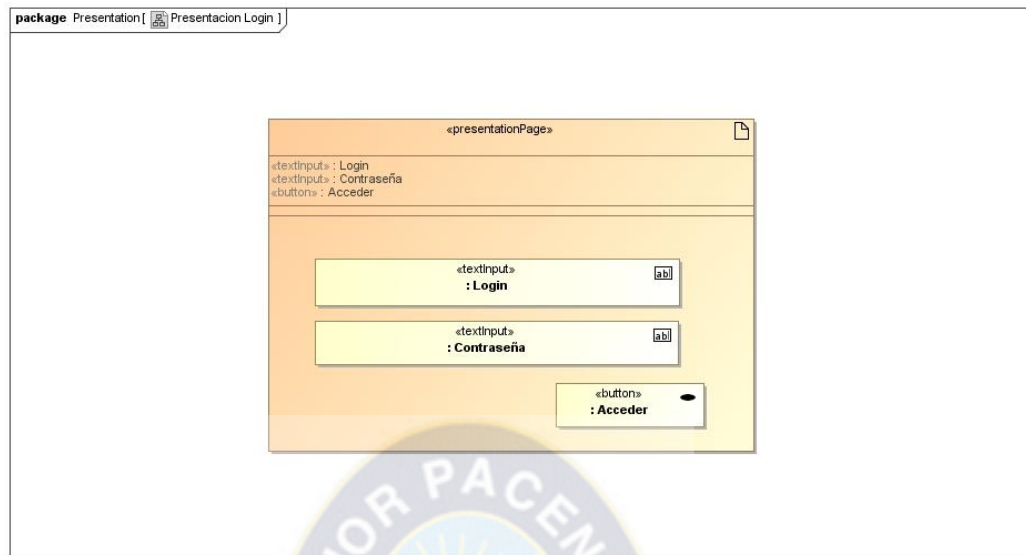


Figura 33. Pantalla de logueo
Fuente: Elaboración propia

A continuación capturas de pantalla del acceso a la aplicación , el siguiente acceso es para los usuarios (trabajadores de la empresa) de la aplicación.



Figura 34. Acceso de usuario
Fuente: Elaboración propia

A continuación una captura de pantalla de logueo y registro para clientes.

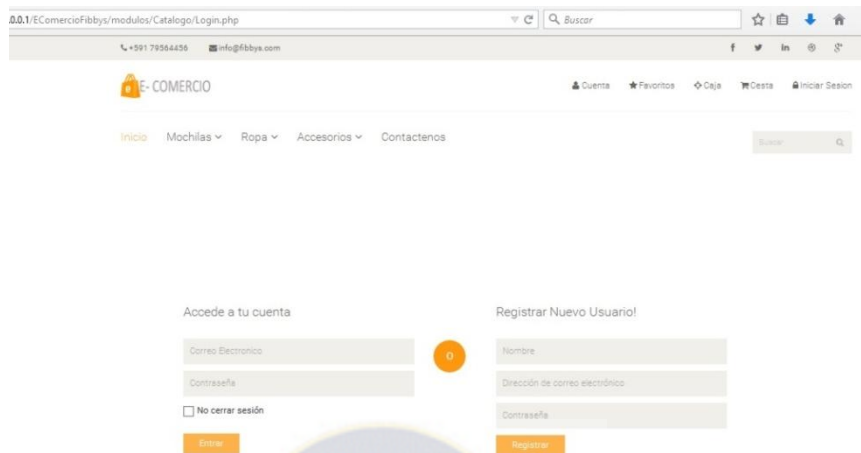


Figura 35. Acceso y registro de clientes
Fuente: Elaboración propia

3.5.3.1.2 Diagrama de Presentación: Cliente Mayorista y minorista

En el siguiente diagrama se muestra la estructura de la página para los clientes de la empresa. Ya sea que estén o no Logueados.

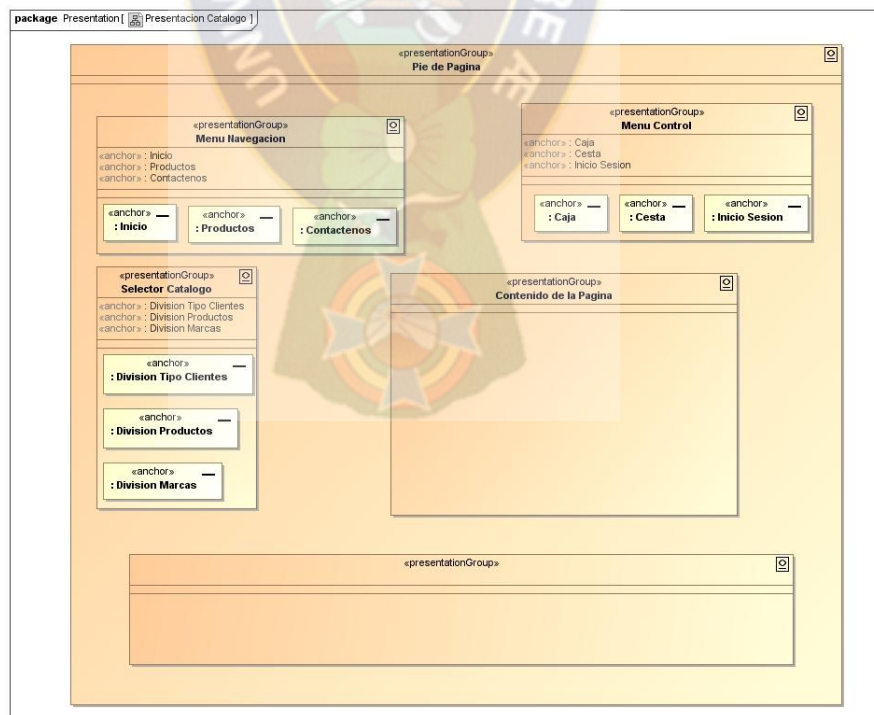


Figura 36. Cliente mayorista y minorista
Fuente: Elaboración propia

A continuación capturas de pantalla del catalogo de productos, descripción de un producto y Cesta de productos.

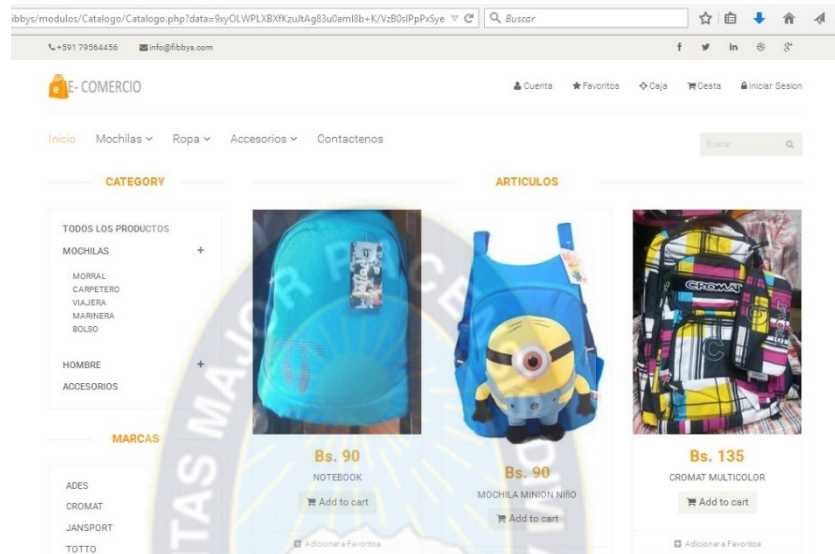


Figura 37. Catalogo del producto
Fuente: Elaboración propia

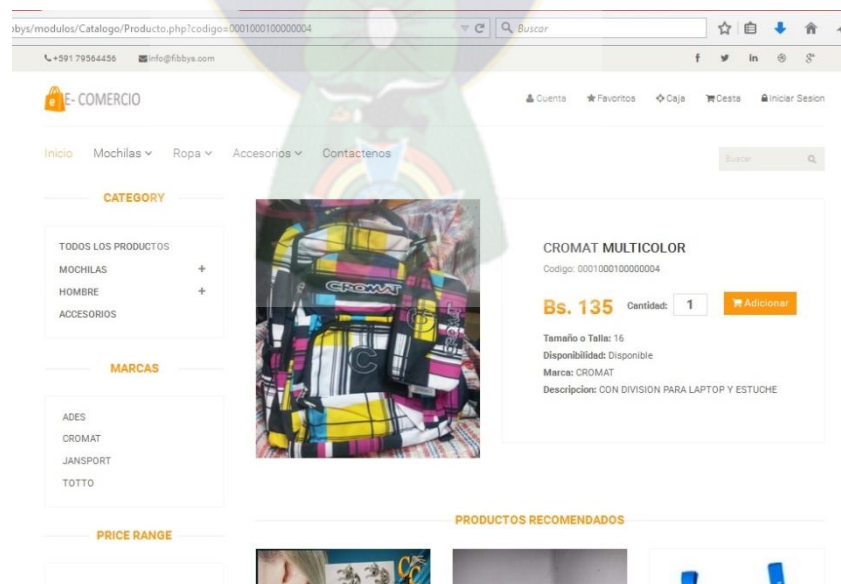


Figura 38. Detalles del producto
Fuente: Elaboración propia

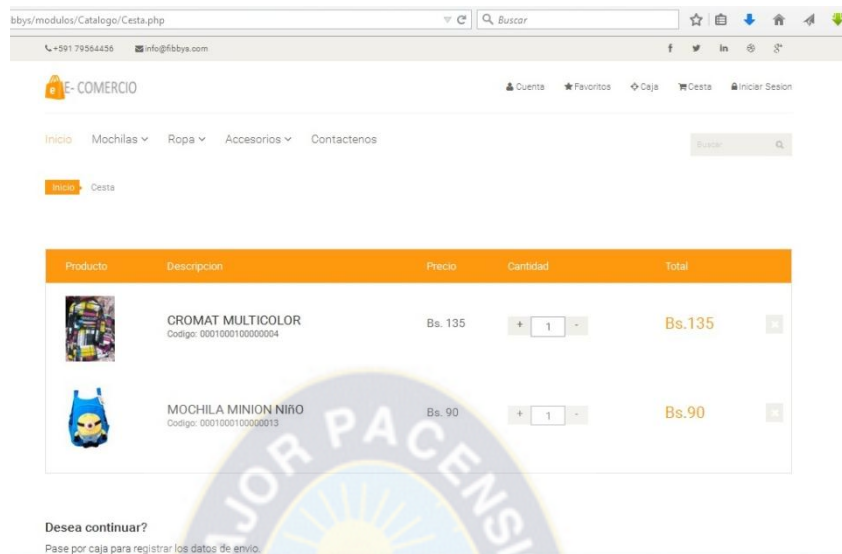


Figura 39. Cesta de productos
Fuente: Elaboración propia

3.5.3.1.3 Diagrama de Presentación: Encargado de Ventas

En el siguiente diagrama se muestra el diseño de todos los accesos que tendrá el Encargado de Ventas.

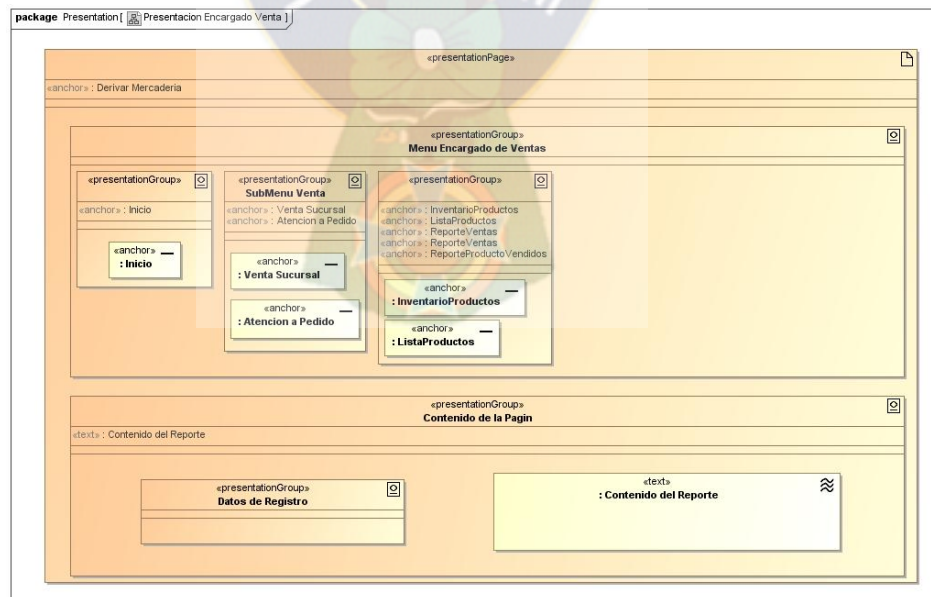


Figura 40. Encargado de ventas
Fuente: Elaboración propia

A continuación capturas de pantalla del menú del Encargado de Ventas, y el formulario de registro de una venta.

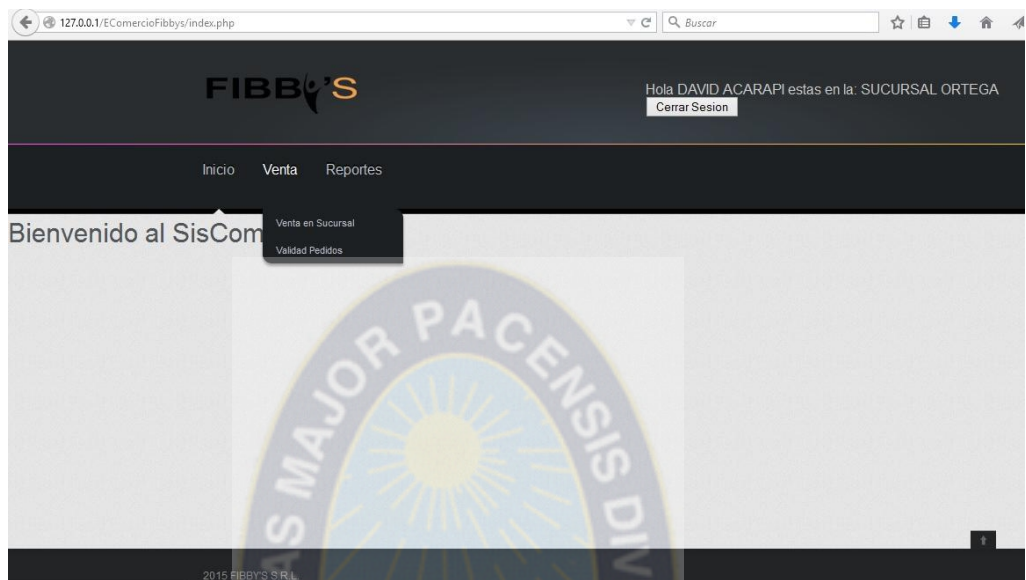


Figura 41. Menú encargado de ventas
Fuente: Elaboración propia

The screenshot shows the "Registro de venta" form in the FIBBY'S web application. The form includes a navigation bar with "Inicio", "Venta", and "Reportes". The main form area contains the following fields and controls:

- Buscar Producto:** A search bar with the text "buscar".
- Fecha:** A date field showing "Jueves, 28 de Mayo de 2015".
- Codigo Producto:** A text field with the value "0001000100000004".
- Tipo de Venta:** A dropdown menu with the selected option "MENOR".
- Marca:** A text field with the value "CROMAT".
- Nombre:** A text field with the value "CROMAT MULTICOLOR".
- CI Cliente:** A text field with the value "111111".
- Tamaño:** A text field with the value "16".
- Nombre Cliente:** A text field with the value "Perez".
- Descripción:** A text field with the value "CON DIVISION PARA LAPTOP Y ESTUCHE".
- Monto Total:** A text field with the value "300".
- Precio:** A text field with the value "135".
- Registrar Venta:** A button to submit the form.
- Disponibilidad:** A text field with the value "0".
- Cantidad:** A text field with the value "1" and increment/decrement buttons.
- Adicionar Canto:** A button to add a corner.

Below the form, there is a table with the following columns: Codigo, Nombre, Imagen, Precio, Cantidad, Total Parcial, and Opcion. The table contains one row:

Codigo	Nombre	Imagen	Precio	Cantidad	Total Parcial	Opcion
0001000200000009	CARPETERO OMEGA		150	1	300	Eliminar

Figura 42. Formulario de registro de venta
Fuente: Elaboración propia

3.5.3.1.4 Diagrama de Presentación: Encargado de Almacén

En el siguiente diagrama se muestra el diseño de todos los accesos que tendrá el Encargado de Almacén.

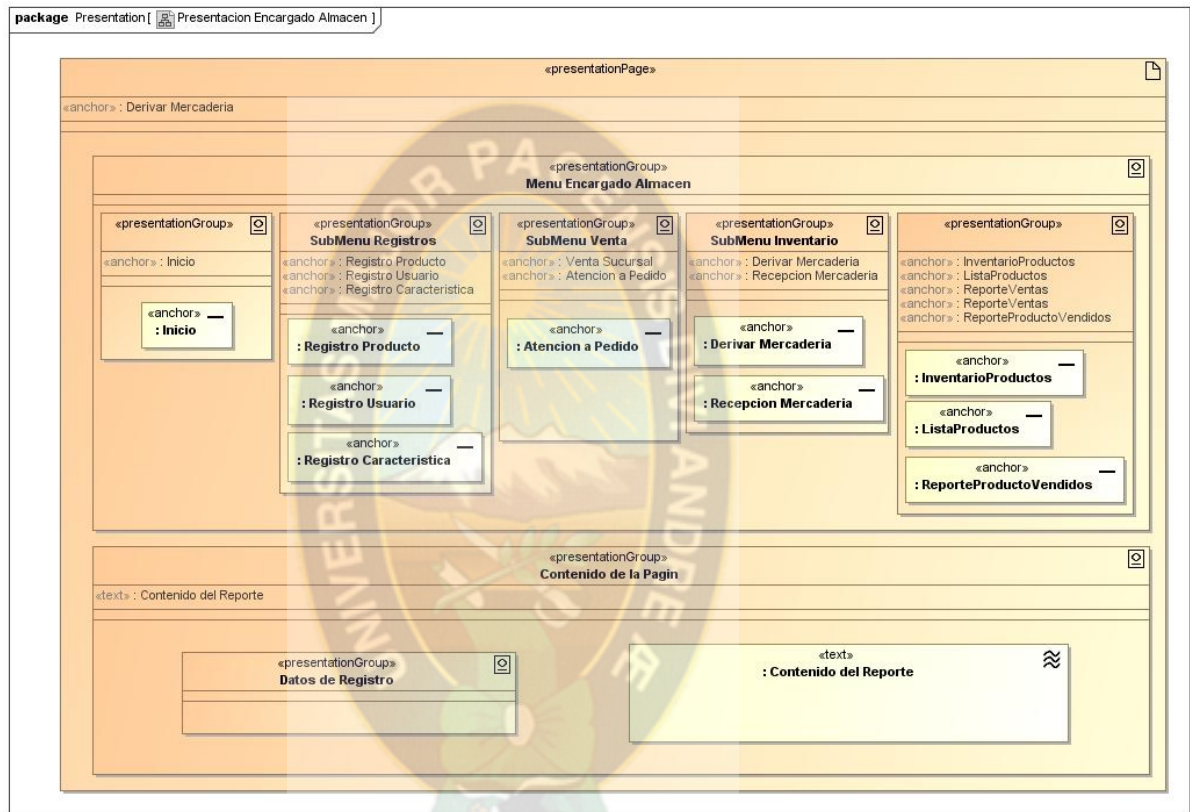


Figura 43. Encargado de almacén
Fuente: Elaboración propia

A continuación capturas de pantalla del menú del Encargado de Almacén, y el formulario de registro de recepción de mercadería.

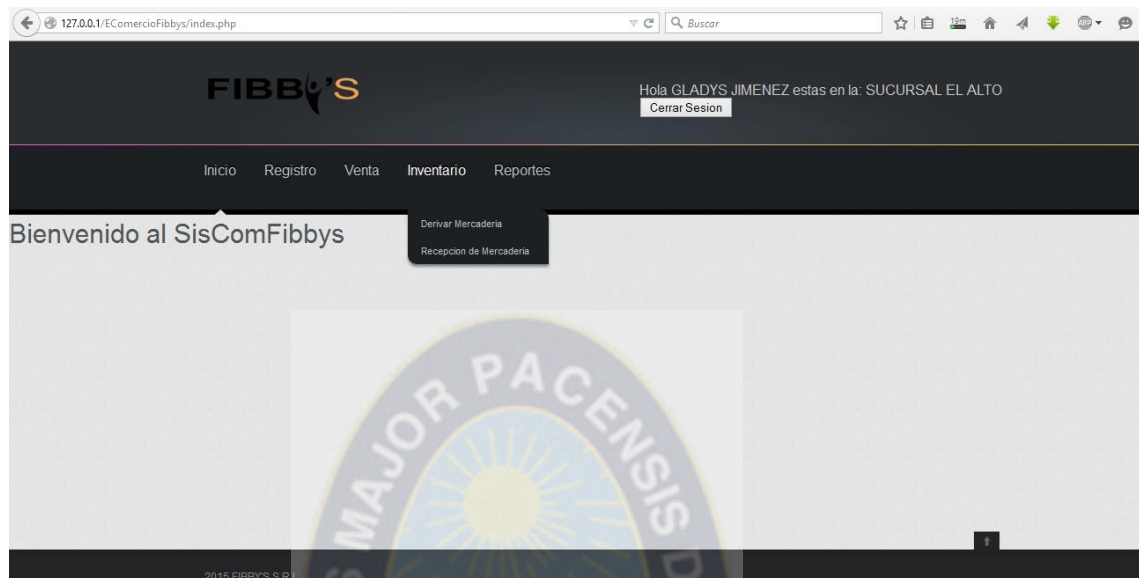


Figura 44. Menú encargado de almacén
Fuente: Elaboración propia

Figura 45. Formulario de registro de mercadería
Fuente: Elaboración propia

3.5.3.1.5 Diagrama de Presentación: Administrador

En el siguiente diagrama se muestra el diseño de todos los accesos que tendrá el Administrador.

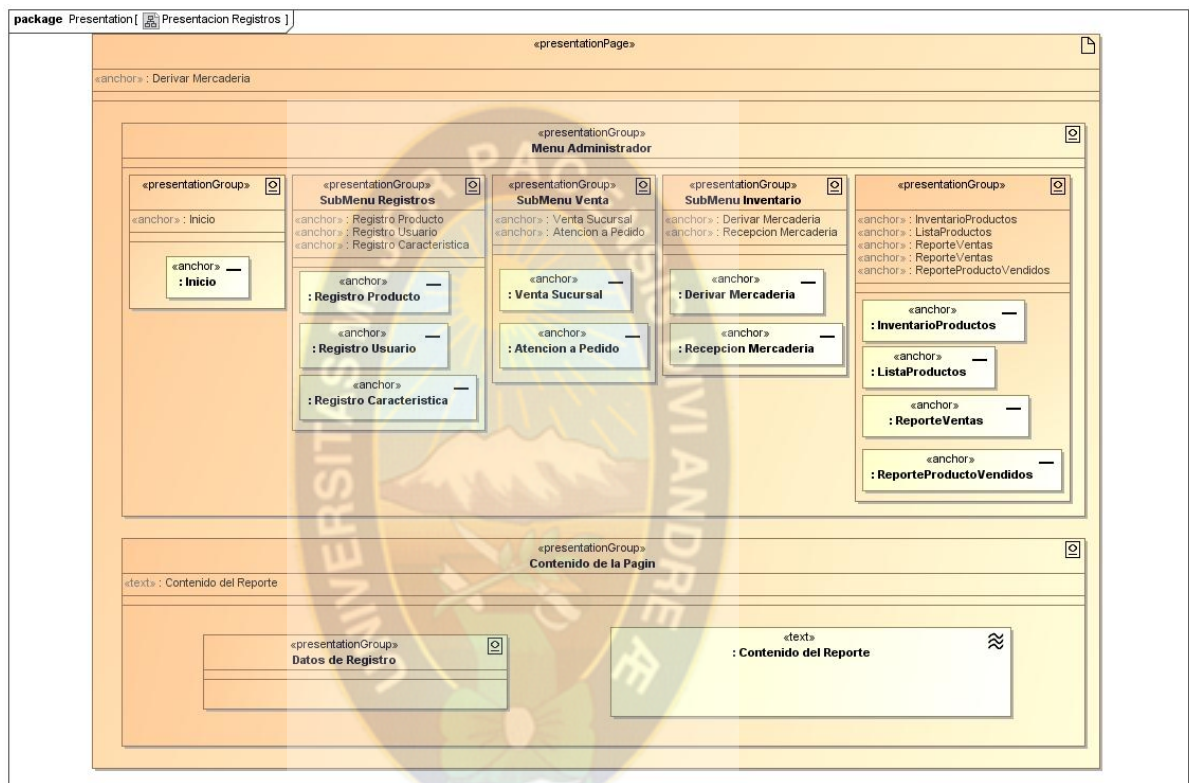


Figura 46. Accesos del administrador
Fuente: Elaboración propia

A continuación capturas de pantalla del menú del Administrador, y el reporte de productos vendidos.

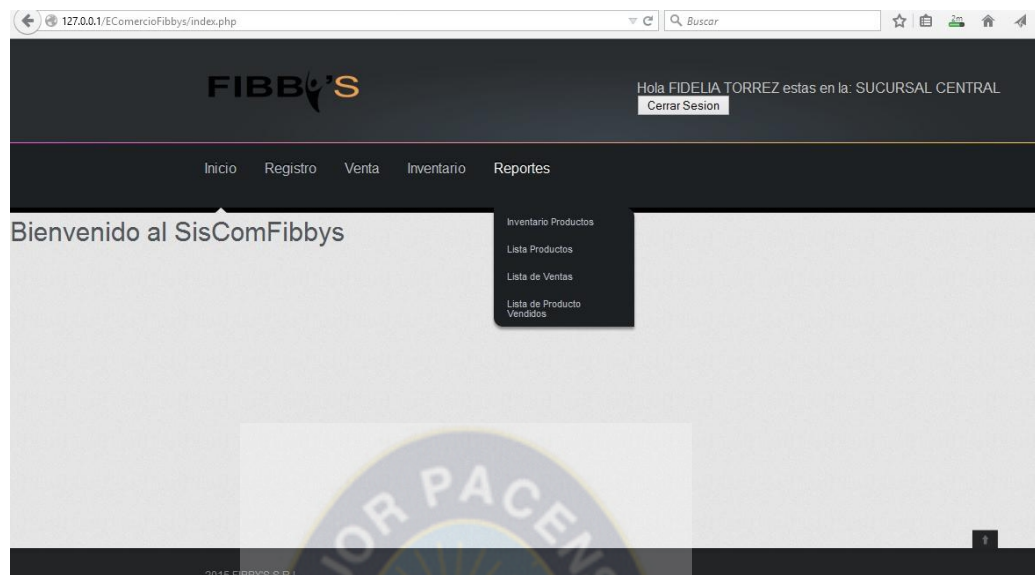


Figura 47. Menú del administrador
Fuente: Elaboración propia





Código Producto	Nombre	Imagen	Marca	Cantidad total	Ingreso Total Por Producto	Costo de Producción o Adquisición
0001000100000001	JANSPORT CLASICO		JANSPORT	6	330	168
0001000100000002	JANSPORT CLASICO		JANSPORT	4	220	112
0001000100000003	NOTEBOOK		TOTTO	10	775	400
0001000200000009	CARPETERO OMEGA		TOTTO	4	240	180

Figura 48. Reporte de productos vendidos
Fuente: Elaboración propia

3.6 Fase de Transición

3.6.1 Disciplina de Modelado

En la fase de transición en la disciplina de modelado se recopila toda la documentación generada en el proyecto, este documento esta toda esa información recopilada.

3.6.2 Disciplina de Implementación

Se verificaron todos los problemas en el sistema, y se subsanaron estos. Sin embargo el sistema debe completarse con otros módulos ajenos al presente trabajo, los cuales se mencionan en el **Capítulo V** de conclusiones y recomendaciones.

3.6.3 Disciplina de Despliegue

En este apartado se procedió a capacitar al personal de la empresa, el cual se realizo con ayuda del manual de usuario, este se encuentra en la sección de anexos del presente documento.

4.1. Calidad del software

En este capítulo se hará un análisis posterior al desarrollo e implementación del SISTEMA WEB DE COMERCIO ELECTRONICO Y CONTROL DE INVENTARIO

CASO: “FIBBY’S S.R.L.”, en este análisis se comprobará la calidad del Software mediante un análisis y haciendo uso de uno de los estándares.

4.1.1. Evaluación elemental WEB SITE QEM

Pretende realizar un aporte ingenieril al proponer un enfoque sistemático, disciplinado y cuantitativo que se adecue a la evaluación, comparación y análisis de calidad de sistemas de información centrados en la Web (OLSINA, 1999)

4.1.2. Tipos de Criterios de Preferencia de Calidad Elemental

Dos tipos básicos de criterios elementales son los absolutos y los relativos, y, dentro de los primeros se pueden descomponer en criterios con variables continuas, y criterios con variables discretas. La siguiente figura muestra una jerarquía detallada de los tipos de criterios.

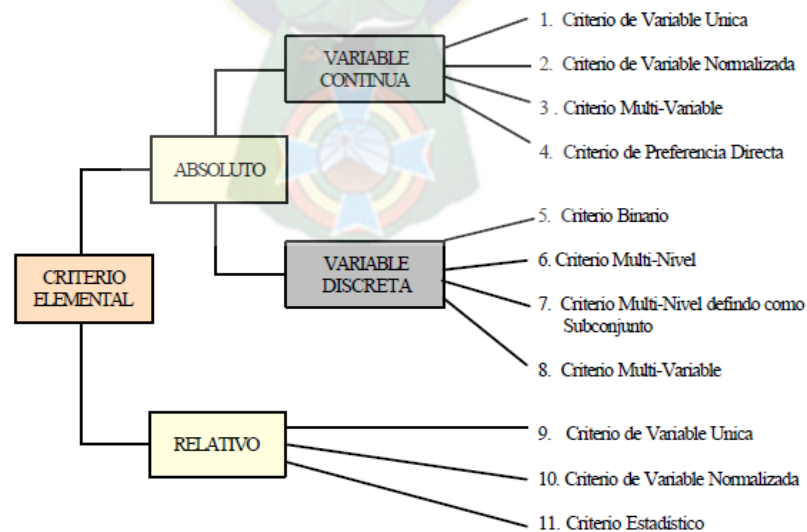


Figura 49. Tipos de criterios de preferencia de calidad elemental
Fuente: <http://myslide.es/documents/15-metodologia-web-qem.html>

4.1.3. Evaluación elemental de calidad

Se tomara en cuenta la especificación de una docena de atributos y sus criterios elementales, principalmente los que se encuentran en el árbol de requerimientos de calidad para el sistema de comercio electrónico e inventario.

1.	USABILIDAD
1.1.	Comprensibilidad Global del Sitio
1.1.1.	<i>Esquema de Organización Global</i>
1.1.1.1.	<i>Mapa del Sitio</i>
1.2.	Mecanismo de Ayuda y Retroalimentación en línea
1.2.1.	<i>Calidad de la ayuda</i>
1.2.1.1.	<i>Ayuda de la Búsqueda</i>
1.2.2.	<i>Indicador de la última Actualización</i>
1.2.2.1.	<i>Global</i>
1.2.2.2.	<i>Por producto</i>
1.2.3.	<i>Retroalimentación</i>
1.2.3.1	<i>Reportes</i>
1.2.3.2.	<i>Formularios de entrada</i>
1.3	Aspectos de Interfaces y Estéticos
1.3.1.	<i>Coherente al agrupar los objetos de controles principales</i>
1.3.2.	<i>Estabilidad y Permanencia en la presentación de los controles Principales</i>
1.3.2.1.	<i>Permanencia de Controles Directos</i>
1.3.2.2.	<i>Permanencia de Controles internos</i>
1.3.3.	<i>Preferencia estética</i>
1.3.3.1.	<i>Uniformidad en el color de enlaces</i>
2.	FUNCIONALIDAD
2.1	Aspectos de búsqueda
2.1.1.	<i>Mecanismo de Búsqueda en el Sitio</i>
2.1.1.1.	<i>Búsqueda Restringida</i>
2.1.1.1.1.	<i>Búsqueda de Productos</i>
2.1.1.1.2.	<i>Búsqueda de Clientes</i>
2.1.1.2.	<i>Búsqueda Global</i>
2.1.1.2.1.	<i>Búsqueda de Productos</i>

2.1.1.2.2.	<i>Catalogo digital</i>
2.2.	Aspectos de Navegación y Exploración
2.2.1.	<i>Navegabilidad Local</i>
2.2.1.1	<i>Orientación</i>
2.2.2.	<i>Navegabilidad Global</i>
2.2.3.	<i>Objetos de Control Navegacional</i>
2.2.3.1.	<i>Nivel de Desplazamiento</i>
2.3.	Funciones Especificas del Dominio
2.3.1.	<i>Información del Producto</i>
2.3.1.1.	<i>Descripción del producto</i>
2.3.1.1.1.	<i>Características del producto</i>
2.3.1.1.2.	<i>Imagen del producto</i>
2.3.1.1.2.1.	<i>Galería de Imágenes del Producto</i>
2.3.1.1.2.2.	<i>Zoom</i>
2.3.1.2.	<i>Disponibilidad del Producto</i>
2.3.2	<i>Características de la compra</i>
2.3.2.1.	<i>En línea</i>
2.3.2.1.1.	<i>Carrito de compras</i>
2.3.2.1.2.	<i>Registro de Pedido</i>
2.3.2.1.3.	<i>Registro de datos de Entrega</i>
2.3.2.1.4.	<i>Validación y entrega de Pedido</i>
2.3.2.2.	<i>En sucursal</i>
2.3.2.2.1.	<i>Información de Venta</i>
2.3.3.	<i>Inventario</i>
2.3.3.1.	<i>Movimientos de Mercadería</i>
2.3.3.1.1.	<i>Ingreso de productos a la empresa</i>
2.3.3.1.2.	<i>Entre Sucursales</i>
2.3.5.	<i>Políticas de Descuentos</i>
2.3.5.1.	<i>Liquidación de productos</i>
2.3.5.2.	<i>Descuentos por temporada</i>
3.	CONFIABILIDAD
3.1.	No Deficiencia
3.1.1.	<i>Errores de Enlaces</i>
3.1.1.1.	<i>Enlaces rotos</i>
3.1.1.2.	<i>Enlaces inválidos</i>
3.1.1.3.	<i>Enlaces no implementados</i>

3.1.2.	<i>Errores y Deficiencias Varias</i>
3.1.2.1.	<i>Deficiencias a diferentes navegadores</i>
3.1.2.2.	<i>Nodos muertos</i>
4.	EFICIENCIA
4.1.	Accesibilidad de Información
4.1.1.	<i>Legibilidad al desactivar la propiedad del Navegador</i>
4.1.1.1.	<i>Imagen con titulo</i>
4.1.1.2.	<i>Legibilidad global</i>
4.1.2.	<i>Accesibilidad de información</i>
4.1.3.	<i>Accesibilidad de ventanas</i>
4.2	Performance
4.2.1.	<i>Paginas de acceso rápido</i>

Tabla 35. Evaluación elemental de calidad
Fuente: Elaboración propia

Título: Carrito de compras; **Código:** 2.3.2.1.1.1; **Tipo:** Atributo

Característica de más Alto Nivel: Funcionalidad

Súper-característica: Modos de Compra

Definición/Comentarios: Esta principalmente disponible en la página principal y resalta parcialmente de la estructura jerárquica de la información de manera que el usuario se familiarice.

Tipo de Criterio Elemental: Es un criterio multi-nivel, discreto y absoluto definido como un subconjunto. Podemos decir que: 0 = No disponible el carro de compras; 1= Parcialmente disponible; 2 = Totalmente disponible

Escala de preferencia:

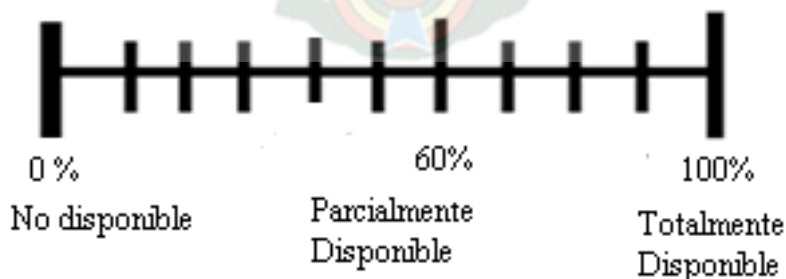


Figura 50. Escala de preferencia
Fuente: <http://myslide.es/documents/15-metodologia-web-qem.html>

Tipo de dato de Recolección: Observacional

Los criterios usados para la evaluación de la calidad serán y su evaluación mediremos mediante:

	Criterio de Preferencia Directa(CrPD) $E_i = (X_i / X_j) * 100$ Donde: X_i = Cantidad total de datos para la variable X_j = Cantidad total de datos
Criterio Binario(CrB) $E_i = 0 \approx 0$, no está disponible, $E_i = 1 \approx 100$, está disponible	Criterio Multi-Nivel(CrMN) $E_i = 0 \approx 0$ No Disponible, $E_i = 1 \approx 60$ Parcialmente Disponible, $E_i = 2 \approx 100$ Totalmente Disponible.

Figura 51. Tipo de dato de recolección
Fuente: Elaboración propia

En la siguiente se muestra los resultados obtenidos en una evaluación.

Código	Título	Tipo de Criterio	E _i (%)
1.	USABILIDAD	CrVN	92.5
1.1.	Comprensibilidad Global del Sitio	CrVN	100
1.1.1.	Esquema de Organización Global	CrVN	100

1.1.1.1.	Mapa del Sitio	CrB	100
1.2.	Mecanismo de Ayuda Retroalimentación en línea	CrVN	91.5
1.2.1.	Calidad de la ayuda	CrVN	80
1.2.1.1.	Ayuda de la Búsqueda	CrPD	80
1.2.2.	Indicador de la ultima Actualización	CrVN	100
1.2.2.1.	Global	CrMN	100
1.2.2.2.	Por producto	CrMN	100
1.2.3.	Retroalimentación	CrVN	91,5
1.2.3.1	Reportes	CrVN	100
1.2.3.2.	Formularios de entrada	CPN	83
1.3	Aspectos de Interfaces y Estéticos	CrVN	85
1.3.1.	Coherente al agrupar los objetos de controles principales	CrPD	70
1.3.2.	Estabilidad y Permanencia en la presentación de los controles principales	CrVN	80
1.3.2.1.	Permanencia de Controles Directos	CrPD	80
1.3.2.2.	Permanencia de Controles internos	CPN	80
1.3.3.	Preferencia estética	CPN	100
1.3.3.1.	Uniformidad de en el color de enlaces	CrMN	100
2.	FUNCIONALIDAD	CrVN	83.2
2.1	Aspectos de búsqueda	CrVN	94.15
2.1.1.	Mecanismo de Búsqueda en el Sitio	CrVN	95
2.1.1.1.	Búsqueda Restringida	CrVN	90
2.1.1.1.1.	Búsqueda de Productos	CrVN	90
2.1.1.2.	Búsqueda Global	CrMN	100
2.1.1.2.1.	Búsqueda de Productos	CrB	100
2.1.1.2.2.	Catalogo digital	CrB	90
2.2.	Aspectos de Navegación y Exploración	CrVN	75
2.2.1.	Navegabilidad Local	CrVN	80
2.2.1.1	Orientación	CrVN	80
2.2.2.	Navegabilidad Global	CrVN	80
2.2.3.	Objetos de Control Navegacional	CrVN	65
2.2.3.1.	Nivel de Desplazamiento	CrVN	65
2.3.	Funciones Especificas del Dominio	CrVN	81,5
2.3.1.	<i>Información del Producto</i>	CrVN	76,67
2.3.1.1.	<i>Descripción del producto</i>	CrVN	80

2.3.1.1.1.	Características del producto	CrMN	100
2.3.1.1.2.	Imagen del producto	CrVN	60
2.3.1.1.2.1	Galería de Imágenes del Producto	CrMN	60
2.3.1.1.2.2	Zoom	CrVN	80
2.3.1.2.	Disponibilidad del Producto	CrVN	80
2.3.2	Características de la compra	CrVN	80
2.3.2.1.	En línea	CrVN	70
2.3.2.1.1.	Carrito de compras	CrVN	92,5
2.3.2.1.2.	Registro de Pedido	CrVN	100
2.3.2.1.3.	Registro de datos de Entrega	CrMN	100
2.3.2.1.4.	Validación y entrega de Pedido	CrMN	100
2.3.2.2.	En sucursal	CrVN	85
2.3.2.2.1.	Información de Venta	CrVN	80
2.3.3.	Inventario	CrVN	90
2.3.3.1.	Movimientos de Mercadería	CrVN	100
2.3.3.1.1.	Ingreso de productos a la empresa	CrB	100
2.3.3.1.2.	Entre Sucursales	CrVN	80
2.3.5.	Políticas de Descuentos	CrPD	80
2.3.5.1.	Liquidación de productos	CrMN	60
2.3.5.2.	Descuentos por temporada	CrMN	60
2.3.1.	Información del Producto	CrMN	60
3.	CONFIABILIDAD	CrVN	80
3.1.	No Deficiencia	CrVN	80
3.1.1.	Errores de Enlaces	CrMN	100
3.1.1.1.	Enlaces rotos	CrMN	100
3.1.1.2.	Enlaces inválidos	CrMN	100
3.1.1.3.	Enlaces no implementados	CrMN	100
3.1.2.	Errores y Deficiencias Varias	CrMN	60
3.1.2.1.	Deficiencias a diferentes navegadores	CrMN	60
3.1.2.2.	Nodos muertos	CrMN	60
4.	EFICIENCIA	CrVN	85,8 3
4.1.	Accesibilidad de Información	CrVN	91,6 7
4.1.1.	Legibilidad al desactivar la propiedad del Navegador	CrB	100

4.1.1.1.	Imagen con titulo	CrB	100
4.1.1.2.	Legibilidad global	CrB	100
4.1.2.	Accesibilidad de información	CrVN	75
4.1.3.	Accesibilidad de ventanas	CrMN	100
4.2	Performance	CrVN	80
4.2.1.	Paginas de acceso rápido	CrPD	80

Figura 52. Resultados obtenidos de la evaluación
Fuente: Elaboración propia

En la siguiente tabla mostramos un resumen de los valores obtenidos de las cuatro características que nos ofrece la evaluación elemental de la calidad de software WEB SITE QEM, más la calidad total del software.

Características	E _i (%)
Usabilidad	84,71
Funcionalidad	83,94
Confiabilidad	80,00
Eficiencia	85,83
Calidad del proyecto	83,62

Figura 53. Resumen de los valores obtenidos
Fuente: Elaboración propia

Lo que se podría interpretar que 8 de cada 10 Consumidores consideran al Sistema de Comercio Electrónico un sistema de calidad.

4.2. Seguridad del software

4.2.1. Seguridad del Sistema

4.2.1.1. Inyeccion Sql

Construir una sentencia SQL dinámica utilizando para ello la entrada proporcionada por un usuario podría permitir a un atacante modificar el significado de la sentencia o incluso ejecutar comandos SQL arbitrarios.

Los errores de inyección SQL se presentan cuando se cumplen las siguientes condiciones:

- El programa recibe datos que proceden de una fuente no confiable.
- Dichos datos son utilizados para construir dinámicamente una sentencia SQL.

En esta aplicación se hace el uso de funciones de php en todos los datos ingresados por el usuario, para evitar la inyección SQL.

4.2.1.2. Uso indebido: Subida de archivos

Dejar a los usuarios que suban ficheros puede permitir a los atacantes inyectar contenido o código malicioso en el servidor.

En esta aplicación se restringe el tipo de archivos ingresados al sistema, solamente aceptando imágenes.

4.2.2. Seguridad de Tigo Money

Tigo Money es un servicio de pago móvil que te permite manejar tu dinero de forma fácil y segura en una billetera móvil activada en tu celular.

A través de la billetera móvil, puedes realizar cargas dinero electrónico, transferir dinero electrónico a otra billetera móvil y convertir el dinero electrónico almacenado, en dinero físico de curso legal.

Todo cliente cuenta con un PIN de seguridad, siendo necesario para realizar cualquier tipo de transacción.

4.3. Evaluación de costos

El objetivo de este apartado es demostrar que con la utilización de este sistema se obtendrá beneficios para el Comercio Electrónico de la Empresa “FIBBY’s S.R.L.”.

Para tal efecto se hace uso de ciertas herramientas y heurísticas que nos ayudarán a calcular el VAN (Valor Actual Neto) y el TIR (Taza Interna de Retorno).

Después de realizar los cálculos necesarios para la obtención de los resultados esperados estaremos en la capacidad de afirmar si el proyecto es viable, redituable, y comprobar que la mejor opción es invertir en el proyecto.

4.3.1. Cocomo

El Modelo Constructivo de Costes (COCOMO) es un modelo matemático de base empírica, utilizado para la estimación de costes de software. Incluye tres sub modelos, cada uno ofrece un nivel de detalle y aproximación, cada vez mayor, a medida que avanza el proceso de desarrollo del software: básico, intermedio y detallado.

Este modelo fue desarrollado por Barry W. Boehm a finales de los años 70 y comienzos de los 80, exponiéndolo detalladamente en su libro “Software Engineering Economics”, COCOMO consta con tres modelos de estimación. En los tres modelos de estimación se utilizan las tres siguientes ecuaciones:

$$E = a(KLDC)^b * m(X), \text{ en persona-mes}$$

$$D = c(E)^d, \text{ en meses}$$

$$P = E/D, \text{ en personas}$$

Dónde:

E: Esfuerzo requerido por el proyecto, en mes.

D: Tiempo requerido por el proyecto, en meses.

P: Número de personas requerido por el proyecto.

a, b, c y d: Constantes con valores definidos, según cada sub-modelo.

KLDC: Cantidad de líneas de código, en miles.

m(X): Multiplicador que depende de 15 atributos.

A la vez cada modelo se subdivide en modos. Estos modos son:

- **Modo orgánico:** Es un pequeño grupo de programadores experimentados desarrollando software en un entorno familiar. El tamaño del software varía desde unos pocos miles de líneas (tamaño pequeño) a unas docenas de miles (medio).
- **Modo semi – libre o semi – acoplado:** Corresponde a un esquema intermedio entre el orgánico y el rígido; el grupo de desarrollo puede incluir una mezcla de personas experimentadas y no experimentadas.
- **Modo rígido o empotrado:** El proyecto tiene fuertes restricciones, que pueden estar relacionadas con la funcionalidad y/o pueden ser técnicas. El problema a resolver es único, siendo difícil basarse en la experiencia puesto que puede no haberla. (kiumek, 2011)

Proyecto de Software	a	b	c	d
Orgánico	2.4	1.05	2.5	0.38
Semi-Acoplado	3.0	1.12	2.5	0.35
Empotrado	3.6	1.20	2.5	0.32

Tabla 36. Tabla de valores de proyectos de software
Fuente: Elaboración propia

4.3.1.1 Costo del Sistema

El Costo del sistema lo plantearemos en tres partes: Desarrollo del Software, Elaboración del Proyecto, Total del Software.

4.3.1.2 Costo de desarrollo del Software

A continuación haremos el cálculo del esfuerzo necesario para la programación del sistema, tomando en cuenta los valores de los atributos que se encuentran

detallados en la siguiente tabla, se encuentran remarcados los valores que se tomaran en cuenta.

Atributos	Valor					
	Muy Bajo	Bajo	Normal	Alto	Muy Alto	Extra Alto
Atributos de Software						
Fiabilidad	0,75	0,88	1,00	1,15	1,40	
Tamaño de la base de datos		0,94	1,00	1,08	1,16	
Complejidad	0,70	0,85	1,00	1,15	1,30	1,65
Atributos de Hardware						
Restricciones del tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de memoria virtual			1,00	1,06	1,21	1,56
Volatilidad de la maquina virtual		0,87	1,00	1,15	1,30	
Tiempo de respuesta		0,87	1,00	1,07	1,15	
Atributo de personal						
Capacidad de análisis	1,46	1,19	1,00	0,86	0,71	
Experiencia en la aplicación	1,29	1,13	1,00	0,91	0,82	
Capacidad de los programadores	1,42	1,17	1,00	0,86	0,70	
Experiencia en la maquina virtual	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje	1,14	1,07	1,00	0,95		
Atributos de proyecto						
Técnicas actualizadas de programación	1,24	1,10	1,00	0,91	0,82	
Utilización de herramientas de software	1,24	1,10	1,00	0,91	0,83	
Restricciones de tiempo de desarrollo	1,23	1,08	1,00	1,04	1,10	

Tabla 37. Costo de desarrollo del software
Fuente: Elaboración propia

La ecuación que nos ayudará a hallar el esfuerzo, viene dada de la siguiente manera:

$$E=a *(KLCD)^{b*} m(X), \text{ en persona-mes}$$

Dónde

E: es el esfuerzo expresado en personas por mes.

a, b : son constantes empíricas.

KLDC: es un número estimado de código fuente en miles distribuidas.

M(X): es un multiplicador que depende de 15 atributos

Para **KLDC** se requiere del número de líneas de código del lenguaje de programación con el que se desarrollara el sistema en base al punto de función obtenido.

Para el cálculo de $m(X)$ tomemos en cuenta los siguientes atributos

Fiabilidad: Si se produce un error en las transacciones o una venta puede causar grandes conflictos por lo tanto su valoración entra en el rango de alto.

Tamaño de la base de datos: La base de datos es estándar por lo tanto su valoración entra en el rango normal.

Complejidad: No existen ninguna complejidad por lo tanto su valoración entra en el rango de Muy bajo.

Restricciones del tiempo de ejecución: Al ser una base de datos pequeña el tiempo de ejecución no es una restricción por lo tanto la valoración entra en el rango normal.

Restricciones de memoria virtual: No existen ninguna restricción por lo tanto su valoración entra en el rango normal.

Volatilidad de la maquina virtual: No existe problema en este apartado puesto que se ejecuta en un servidor, y del lado del cliente todo el código ejecutado por el navegador es de una complejidad baja por lo tanto su valoración entra en el rango bajo.

Tiempo de respuesta: El uso de parte del cliente y de los usuarios no implica una interactividad continua, por lo tanto su valoración entra en el rango bajo.

Capacidad de análisis: El análisis no presenta una complejidad alta por lo tanto la valoración bajo.

Experiencia en la aplicación: Al ser una aplicación pequeña, la experiencia del programador no es muy exigente por lo tanto su valoración entra en el rango de Muy Bajo.

Capacidad de los programadores: Es muy importante que la capacidad de los programadores ante los proyectos similares sea valorado en un rango alto.

Experiencia en la maquina virtual: La experiencia es estándar por lo tanto su valoración entra en el rango muy bajo.

Experiencia en el lenguaje: Es muy importante que la experiencia en el lenguaje que los proyectos son realizados sea valorado en un rango normal.

Técnicas actualizadas de programación: Las técnicas utilizadas son de carácter estándar por lo tanto su valoración entra en el rango normal.

Utilización de herramientas de software: Las herramientas utilizadas son estándar por lo tanto su valoración entra en el rango normal.

Restricciones de tiempo de desarrollo: No existen restricciones por lo tanto su valoración entra en el rango muy bajo.

$$M(x) = 1,15 * 1,00 * 0,70 * 1,0 * 1,00 * 0,87 * 0,87 * 1,46 * 1,29 * 0,86 * 1,21 * 1,00 * 1,00 * 1,00 * 1,23 = 1,468810$$

La aplicación cuenta con un promedio de 20 líneas de código, con un total de 45 scripts entre html, css, javascript y php, con un total de 900 líneas de código

Como nuestro proyecto es del tipo Orgánico utilizaremos: $a=2.4$ y $b=1.05$, reemplazando estos valores en la ecuación:

$$E=a \cdot (KLCD)^b \cdot m(X), \text{ en persona-mes}$$

$$E=2,4 \cdot (0,9)^{1,05} \cdot (1,47) = 3.16$$

Calculando D con $c=1.05$ y $d=0.38$ tenemos:

$$D=c(E)^d, \text{ en meses}$$

$$D=1.05 \cdot (3,16)^{0,38}$$

$$D=1,62$$

El proyecto deberá tener un desarrollo de aproximadamente 2 meses.

Para calcular la cantidad de programadores utilizamos la siguiente fórmula:

$$P=E/D, \text{ en personas}$$

Reemplazando los datos ya conocidos tenemos:

$$P=4,28/1,82$$

$$P= 1,95= 2 \text{ (Programadores)}$$

El salario promedio de un programador oscila entre los 2500 y 4500 Bs., en nuestro caso tomaremos un promedio con un valor de 2500 Bs. Sueldo de un programador Junior, esto debido a los 15 puntos calculados anteriormente. A partir de este monto podemos calcular el costo total del desarrollo del software:

Costo mensual de desarrollo = Número de programadores * Salario promedio

$$\text{Costo mensual de desarrollo} = 2 \cdot 2500$$

$$\text{Costo mensual de desarrollo} = 5000 \text{ Bs.}$$

Como el desarrollo de software se lo estima en 2 meses, tendremos:

Costo total de desarrollo = Costo del software por mes* Número de meses

$$\text{Costo total de desarrollo} = 5000 * 1.62 = 8100 \text{ Bs.} = 1165 \text{ \$us}$$

4.3.1.3 Costo de Elaboración del Proyecto

Los costos de elaboración del proyecto se refieren principalmente a los gastos que se realizaron a lo largo de las diferentes fases del PROCESO UNIFICADO AGIL (AUP). Estas las podemos ver expresadas en la siguiente tabla

Detalle	Costo (\$us)
Análisis y diseño del proyecto	60
Material de escritorio	15
Internet	70
Otros	45
Total	190

Tabla 38. Costo de elaboración del proyecto
Fuente: Elaboración propia

4.3.1.4 Costo Total del Software

El costo total del software se lo obtiene de la sumatoria de los diferentes costos vistos hasta el momento, costo de desarrollo, implementación y elaboración del proyecto. Esto lo vemos expresado en la siguiente tabla:

Detalle	Costo(\$us)
Costo de desarrollo	1165
Costo de implementación	50
Costo de elaboración del proyecto	190
Total	1405

Tabla 39. Costo total del software
Fuente: Elaboración propia

El costo total del software es de 1405 \$us evaluados de los diferentes costos vistos hasta el momento, costo de desarrollo, implementación y elaboración del proyecto.



5.1 Conclusiones

A continuación se brinda las conclusiones del presente proyecto de grado:

- Actualmente la empresa se promociona en internet por medio de la pagina web desarrollada en este trabajo.
- Se desarrolló el modulo de comercio electronico del sistema brinda una nueva manera de realizar ventas para la empresa con un costo bajo y sostenible.
- Respecto a las ventas por internet el numero de estas se va incrementando con el tiempo puesto que si bien en nuestro contexto son conocidas los clientes todavia muestran desconfianza.
- El modulo de inventario fue desarrollado de tal manera que no se pierdan datos al momento de realizar movimientos o adquisiciones por parte de la empresa, además adecuarse en caso de errores por parte del personal.
- Los clientes asiduos están conformes con el medio de pago “TIGO MONEY”, en cuanto a los clientes nuevos van acostumbrándose a este medio de pago.
- Se logro desarrollar el modulo de los reportes con gran valor para la empresa para realizar un seguimiento.
- La interrelación de metodologías optimizó el trabajo a desarrollar e implementar el Sistema de comercio electrónico, considerando que la metodología (AUP) es práctica al momento de trabajar y la Metodología UWE es útil y concreto al momento de realizar el diseño del Sistema.

- Si bien es buena la metodología y la unión con el modelado, existen tareas que no son simples de realizarlas con una persona detrás de todo, puesto que todas las metodologías por lo general involucran por lo menos 8 personas.

5.2 Recomendaciones

- Se recomienda encargar a una persona de la empresa en promocionar la página web por las redes sociales.
- Toda la información de la página web debe ser constantemente actualizada registrando o deshabilitando productos como se vaya requiriendo.
- Se recomienda hacer un módulo donde exista una interacción con el sistema contable, y el área de producción.
- La información que genera el sistema es amplia, por lo que se sugiere a la empresa proponer reportes más específicos y detallados conforme realicen el uso de la aplicación.
- La información generada a partir de los usuarios puede ser usada para la implementación de un módulo de control de personal.
- Se recomienda la implementación de pago mediante transferencia bancaria, conforme se vayan haciendo más accesibles estas transferencias.
- En cuanto a la metodología AUP y el modelado UWE es bastante útil la unión para representar todo el sistema, pero este no contempla el diagrama entidad relación o relacional que podría ayudar a representar

las base de datos, por lo que es recomendable adicionar este o buscar otro diagrama para este proposito.



Bibliografía

The jQuery Foundation. (2015). *jQuery write less, do more*. Recuperado el 10 de Febrero de 2015, de jQuery write less, do more.: <https://jquery.com/>

(adigital), A. E. (2012). *Libro Blanco del Comercio Electronico*.

Ambyssoft, I. (2006). *The Agile Unified Process - CGI*. Recuperado el 2015, de <http://www.cc.una.ac.cr/AUP/>

Bos, Bert. (27 de Mayo de 1994–2015). *Cascading Style Sheets*. Recuperado el 24 de Marzo de 2015, de Cascading Style Sheets: <http://www.w3.org/Style/CSS/>

FIBBY'S. (2008). *Reglamento*. La Paz.

INTECO. (2009). *CURSO DE INTRODUCCIÓN A LA INGENIERÍA DEL SOFTWARE*.

kiumek. (2011). *angelfire*. Obtenido de angelfire:
<http://www.angelfire.com/creep/kiumek/Cocomo.htm>

Lazcano, S.d. A. (4 de junio de 2015). Sólo dos rubros explotan el comercio electrónico. (L. Tiempos, Ed.) *los Tiempos*, 1.

München, L.-M.-U. (2015). *UML-BASED WEB ENGINEERING*. Recuperado el 2015, de <http://uwe.pst.ifi.lmu.de>

Nancy Merlo- Shett. (2002-2009). *Cocomo*. University of Zurich, Switzerland: Department of Computer Science.

OLSINA, L. A. (Noviembre de 1999). Metodología Cuantitativa para la Evaluación y Comparación de la Calidad de Sitios Web. *Tesis Doctoral*. La Plata, Argentina.

PHP, G. (2001 - 2015). *Php*. Recuperado el 2015, de <http://php.net/>

Ramírez, J. (2006). *Inventarios*. MARACAIBO.

Refsnes Data. (1999-2015). *Bootstrap*. Recuperado el 2015, de Bootstrap:
<http://www.w3schools.com/bootstrap/>

ShareAlike. (1999-2014). *MySql*. Recuperado el 2015, de MySql:
<http://www.w3schools.com/sql/>

W3SCHOOLS. (1999-2015). *W3SCHOOLS*. Recuperado el 2015, de <http://www.w3schools.com/js/>