

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR  
FACULTAD DE INGENIERÍA  
ESCUELA DE SISTEMAS**



**DISERTACIÓN PREVIA A LA OBTENCIÓN DEL TÍTULO DE  
INGENIERO EN SISTEMAS Y COMPUTACIÓN**

**“ANÁLISIS, DESARROLLO E IMPLEMENTACIÓN DE UNA  
APLICACIÓN WEB ORIENTADA PARA EL REGISTRO DE  
ÓRDENES DE COMPRA, CLIENTES Y PRODUCTOS PARA  
LA EMPRESA FERRETERIA TOBAR NOLIVOS CIA. LTDA.”**

**DIEGO RODRIGO TOBAR CHICAIZA**

**DIRECTOR: ING. OSWALDO ESPINOSA**

**CIUDAD QUITO, AÑO 2014**

## **Dedicatoria**

Dedico mi tesis de grado a mi mami Lolita y a mi papi Humberto quienes me han apoyado en todo para poder llegar a esta instancia de mi vida, ya que siempre han estado en todo momento dándome sus consejos, su amor, su comprensión, su apoyo y por darme toda su ayuda para poder culminar mis estudios. Son un gran ejemplo a seguir por su constancia, esfuerzo, dedicación y humildad, que con su trabajo han hecho grandes cosas y les tengo una gran admiración por ser los ejes y pilares de mi familia, que sus valores y gran ejemplo siempre llevamos sus hijos y sus nietos.

Los quiero mucho mami y papi, muchas gracias por todo lo que me han dado, este es el comienzo de una nueva etapa de mi vida y quiero que sepan que siempre voy a estar a su lado porque les debo tanto porque llegaron a ser más que mis padres sino mis grandes amigos.

Diego

## **Agradecimientos**

Agradezco de todo corazón a mi Dios por darme la salud, la sabiduría y la fuerza para seguir adelante y no desmayar para conseguir este gran sueño de mi vida.

A mis hermanas Mónica, Adriana y a mi hermano José, por darme tantos consejos y su apoyo incondicional, los quiero mucho ñaños gracias por mantenernos juntos y nunca separarnos.

A mis sobrinos Cristian, Esteban, Andrés y a mi futuro sobrino o sobrina que está en camino los quiero mucho a todos, quiero ser para ustedes su amigo incondicional que siempre los va apoyar y velar por su bienestar siempre.

A mi novia Evelyn por acompañarme en todo momento, por su comprensión, su amor y su cariño que en los momentos más difíciles ha sido la que me ha dado el empuje para poder seguir adelante, le agradezco eternamente por todo lo que me ha dado.

A mi director de tesis Ing. Oswaldo Espinosa muchas gracias por su ayuda, consejos y motivación en todo momento para poder culminar mi tesis, gracias Inge por guiarme en el proceso de desarrollo de mi tesis.

Diego

## **Resumen**

En la presente disertación de grado se evidencia el análisis, desarrollo e implementación de una aplicación web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferretería Tobar Nolivos Cía. Ltda.

Con el presente proyecto se logró que los procesos de ingreso de información de la empresa se automaticen y ya no sea necesario el manejo de información en una forma manual sino ya con la utilización de la aplicación web.

Para el desarrollo del sistema se utilizó la metodología de trabajo SCRUM, lo cual nos permitió presentar iterativamente al cliente el progreso del proyecto y poder realizar en ciertos casos cambios en el sistema para que este cumpla con todos los requerimientos requeridos por el cliente.

Las herramientas utilizadas para la elaboración del proyecto son de software libre, por lo que se utilizó el framework CodeIgniter que usa el patrón de diseño Modelo - Vista -Controlador (MVC), el framework ayudo en gran medida para la organización del código y de los archivos para desarrollar la aplicación web. Se utilizó el lenguaje de programación PHP, JavaScript y como base de datos se utilizó MySQL.

El desarrollo del proyecto se ha dividido en tres etapas: El levantamiento de requerimientos, apoyado en la norma IEEE 830 que asegura la integración de los requerimientos desde la perspectiva del usuario, cliente y desarrollador, definiendo los límites y alcance del proyecto. El diseño que es el proceso en el que se realizan todos los diagramas que propone la metodología y el desarrollo cuyo principal objetivo es la codificación misma del producto basada en los requerimientos del cliente.

Por último a la aplicación se le genero ciertas pruebas para verificar que todos los componentes de la misma funcionen correctamente sin ningún error, estas pruebas fueron pasadas exitosamente. Así la aplicación web para la empresa Ferretería Tobar Nolivos Cía. Ltda. fue presentada al gerente general de la empresa y demás colaboradores siendo de gran aceptación y está en marcha para el trabajo diario de la empresa.

# Índice de Contenidos

CAPÍTULO 1.....	1
<b>1. Introducción</b> .....	1
<b>1.1. Problema</b> .....	2
<b>1.2. Justificación</b> .....	3
<b>1.3. Alcance</b> .....	4
<b>1.4. Objetivos</b> .....	5
<b>1.4.1. Objetivos Específicos</b> .....	5
<b>1.4.2. Objetivos Generales</b> .....	5
CAPÍTULO 2.....	6
<b>2. Marco Teórico</b> .....	6
<b>2.1. Metodologías Tradicionales de Desarrollo de Software</b> .....	7
<b>2.1.1. Principales Metodologías Tradicionales</b> .....	8
<b>2.1.1.1. Rational Unified Process (RUP)</b> .....	8
<b>2.1.1.2. Microsoft Solution Framework (MSF)</b> .....	9
<b>2.2. Metodologías Ágiles de Desarrollo de Software</b> .....	10
<b>2.2.1. Manifiesto Ágil</b> .....	10
<b>2.2.2. Principales Metodologías Ágiles</b> .....	12
<b>2.2.2.1. Extreme Programming</b> .....	12
<b>2.2.2.2. Lean Development</b> .....	12
<b>2.2.2.3. Crystal Methodologies</b> .....	13
<b>2.3. SCRUM</b> .....	14
<b>2.3.1. Introducción</b> .....	14
<b>2.3.2. La Esencia de SCRUM</b> .....	15
<b>2.3.3. Elementos del SCRUM</b> .....	15
<b>2.3.3.1. Roles</b> .....	16
<b>2.3.3.2. Poda de requerimientos</b> .....	18
<b>2.3.3.3. Product Backlog</b> .....	18
<b>2.3.3.4. Sprint</b> .....	18
<b>2.3.3.5. Planificación</b> .....	19

2.3.3.6.	Valores .....	22
2.4.	Definición General del Desarrollo Web.....	23
2.4.1.	Desarrollo Web .....	23
2.4.2.	Desarrollo de aplicaciones Web con metodologías ágiles.....	23
2.4.3.	Definición de una Aplicación Web.....	24
2.4.3.1.	Estructura de las Aplicaciones Web .....	24
2.4.3.2.	Uso de las Aplicaciones Web.....	25
2.5.	PHP .....	25
2.5.1.	Arquitectura y Servicios .....	26
2.6.	Framework Desarrollo Web .....	26
2.7.	CodeIgniter .....	27
2.7.1.	Funcionalidades de CodeIgniter .....	27
2.8.	Gestor Base de Datos .....	28
2.8.1.	MySQL.....	28
2.9.	Herramientas de Desarrollo .....	29
2.9.1.	Power Designer 15.1.....	29
2.9.2.	Zend Studio 9.0 .....	29
CAPÍTULO 3.....		¡Error! Marcador no definido.
3.	Análisis.....	30
3.1.	Identificación de Requerimientos.....	30
3.1.1.	Especificación de Requerimientos de Software ERS .....	30
3.2.	Análisis de Factibilidad.....	31
3.2.1.	Factibilidad Técnica.....	31
3.2.2.	Factibilidad Operativa.....	31
CAPÍTULO 4.....		32
4.	Diseño y Desarrollo del Software .....	32
4.1.	Requerimientos Funcionales.....	32
4.1.1.	Administraciones .....	32
4.1.2.	Perfiles de Usuario .....	32
4.2.	Diagrama General.....	33

<b>4.3.</b>	<b>Diagrama de Casos de Uso .....</b>	<b>34</b>
<b>4.3.1.</b>	<b>Ingreso al Sistema .....</b>	<b>34</b>
<b>4.3.2.</b>	<b>Administración de Usuarios.....</b>	<b>35</b>
<b>4.3.2.1.</b>	<b>Ingresar Usuarios .....</b>	<b>35</b>
<b>4.3.2.2.</b>	<b>Modificar Usuarios.....</b>	<b>36</b>
<b>4.3.2.3.</b>	<b>Eliminar Usuarios .....</b>	<b>37</b>
<b>4.3.2.4.</b>	<b>Buscar Usuarios .....</b>	<b>38</b>
<b>4.3.3.</b>	<b>Administración de Clientes.....</b>	<b>39</b>
<b>4.3.3.1.</b>	<b>Ingresar Clientes .....</b>	<b>39</b>
<b>4.3.3.2.</b>	<b>Modificar Clientes .....</b>	<b>40</b>
<b>4.3.3.3.</b>	<b>Eliminar Clientes.....</b>	<b>41</b>
<b>4.3.3.4.</b>	<b>Buscar Clientes.....</b>	<b>42</b>
<b>4.3.4.</b>	<b>Administración de Productos .....</b>	<b>43</b>
<b>4.3.4.1.</b>	<b>Ingresar Productos.....</b>	<b>44</b>
<b>4.3.4.2.</b>	<b>Modificar Productos .....</b>	<b>45</b>
<b>4.3.4.3.</b>	<b>Eliminar Productos .....</b>	<b>46</b>
<b>4.3.4.4.</b>	<b>Buscar Productos .....</b>	<b>47</b>
<b>4.3.5.</b>	<b>Administración de Órdenes de Compra.....</b>	<b>48</b>
<b>4.3.5.1.</b>	<b>Ingresar Órdenes de Compra .....</b>	<b>48</b>
<b>4.3.5.2.</b>	<b>Modificar Órdenes de Compra.....</b>	<b>49</b>
<b>4.3.5.3.</b>	<b>Eliminar Órdenes de Compra.....</b>	<b>50</b>
<b>4.3.5.4.</b>	<b>Buscar Órdenes de Compra.....</b>	<b>51</b>
<b>4.3.6.</b>	<b>Administración de Proveedores.....</b>	<b>52</b>
<b>4.3.6.1.</b>	<b>Ingresar Proveedores .....</b>	<b>52</b>
<b>4.3.6.2.</b>	<b>Modificar Proveedores.....</b>	<b>53</b>
<b>4.3.6.3.</b>	<b>Eliminar Proveedores.....</b>	<b>54</b>
<b>4.3.6.4.</b>	<b>Buscar Proveedores.....</b>	<b>55</b>
<b>4.4.</b>	<b>Diagrama de Clases.....</b>	<b>56</b>
<b>4.5.</b>	<b>Diagrama de Navegación.....</b>	<b>57</b>
<b>4.6.</b>	<b>Diagrama de Interfaz.....</b>	<b>57</b>

4.7.	Diseño del Software.....	58
4.7.1.	Patrones de Diseño Web .....	58
4.7.2.	Diseño Arquitectónico.....	58
4.7.2.1.	Patrón Modelo –Vista- Controlador (MVC) .....	58
4.7.3.	Diagramas de Secuencia .....	60
4.7.3.1.	Ingresar Clientes .....	60
4.7.3.2.	Modificar Clientes .....	61
4.7.3.3.	Eliminar Clientes.....	62
4.7.3.4.	Buscar Clientes.....	63
4.7.3.5.	Ingresar Productos.....	64
4.7.3.6.	Modificar Productos .....	65
4.7.3.7.	Eliminar Productos .....	66
4.7.3.8.	Buscar Productos .....	67
4.7.3.9.	Ingresar Usuarios .....	68
4.7.3.10.	Modificar Usuarios.....	69
4.7.3.11.	Eliminar Usuarios .....	70
4.7.3.12.	Buscar Usuarios .....	71
4.7.3.13.	Ingresar Órdenes de Compra .....	72
4.7.3.14.	Modificar Órdenes de Compra.....	73
4.7.3.15.	Eliminar Órdenes de Compra.....	74
4.7.3.16.	Buscar Órdenes de Compra.....	75
4.7.3.17.	Ingresar Proveedores .....	76
4.7.3.18.	Modificar Proveedores.....	77
4.7.3.19.	Eliminar Proveedores.....	78
4.7.3.20.	Buscar Proveedores.....	79
4.7.4.	Modelo Físico .....	80
CAPÍTULO 5.....		81
5.	Pruebas .....	81
5.1.	Pruebas de la aplicación con la Técnica FLOOT (testing) .....	81
5.1.1.	Pruebas de Caja Negra .....	81

<b>5.1.2. Pruebas de Interfaz de Usuario</b> .....	84
<b>CAPÍTULO 6</b> .....	88
<b>6. Conclusiones y Recomendaciones</b> .....	88
<b>6.1. Conclusiones</b> .....	88
<b>6.2. Recomendaciones</b> .....	89
<b>BIBLIOGRAFÍA</b> .....	91
<b>ANEXO A</b> .....	92

## Índice Tablas

Tabla 1: Roles SCRUM .....	16
Tabla 2: Flujo Ingreso al Sistema .....	34
Tabla 3: Flujo Ingresar Usuarios .....	36
Tabla 4: Flujo Modificar Usuarios .....	37
Tabla 5: Flujo Eliminar Usuarios.....	37
Tabla 6: Flujo Buscar Usuarios .....	38
Tabla 7: Flujo Ingresar Clientes .....	40
Tabla 8: Flujo Modificar Clientes.....	41
Tabla 9: Flujo Eliminar Usuarios.....	42
Tabla 10: Flujo Buscar Clientes .....	43
Tabla 11: Flujo Ingresar Productos .....	44
Tabla 12: Flujo Modificar Productos .....	45
Tabla 13: Flujo Eliminar Usuarios.....	46
Tabla 14: Flujo Buscar Productos.....	47
Tabla 15: Flujo Ingresar Órdenes de Compra.....	49
Tabla 16: Flujo Modificar Órdenes de Compra.....	50
Tabla 17: Flujo Eliminar Órdenes de Compra.....	50
Tabla 18: Flujo Buscar Órdenes de Compra .....	51
Tabla 19: Flujo Ingresar Proveedores .....	53
Tabla 20: Flujo Modificar Proveedores .....	54
Tabla 21: Flujo Eliminar Proveedores .....	54
Tabla 22: Flujo Buscar Proveedores .....	55
Tabla 23: Pruebas Caja Negra Administración de Usuarios .....	82
Tabla 24: Pruebas Caja Negra Administración de Clientes .....	82
Tabla 25: Pruebas Caja Negra Administración de Productos .....	83
Tabla 26: Pruebas Caja Negra Administración de Órdenes de Compra .....	84
Tabla 27: Pruebas Caja Negra Administración de Proveedores.....	84
Tabla 28: Pruebas Interfaz Administración de Usuarios .....	85
Tabla 29: Pruebas Interfaz Administración de Clientes .....	86
Tabla 30: Pruebas Interfaz Administración de Productos.....	86
Tabla 31: Pruebas Interfaz Administración de Órdenes de Compra .....	87
Tabla 32: Pruebas Interfaz Administración de Proveedores .....	87

## Índice Diagramas

Diagrama 1: Diagrama General .....	33
Diagrama 2: Ingreso al Sistema .....	34
Diagrama 3: Administración de Usuarios .....	35
Diagrama 4: Ingresar Usuarios .....	35
Diagrama 5: Modificar Usuarios .....	36
Diagrama 6: Eliminar Usuarios .....	37
Diagrama 7: Buscar Usuarios .....	38
Diagrama 8: Administración de Clientes .....	39
Diagrama 9: Ingresar Clientes .....	39
Diagrama 10: Modificar Clientes .....	40
Diagrama 11: Eliminar Usuarios .....	41
Diagrama 12: Buscar Clientes .....	42
Diagrama 13: Administración de Productos .....	43
Diagrama 14: Ingresar Productos .....	44
Diagrama 15: Modificar Productos .....	45
Diagrama 16: Eliminar Usuarios .....	46
Diagrama 17: Buscar Productos .....	47
Diagrama 18: Administración de Órdenes de Compra .....	48
Diagrama 19: Ingresar Órdenes de Compra.....	48
Diagrama 20: Modificar Órdenes de Compra.....	49
Diagrama 21: Eliminar Órdenes de Compra .....	50
Diagrama 22: Buscar Órdenes de Compra .....	51
Diagrama 23: Administración de Proveedores .....	52
Diagrama 24: Ingresar Proveedores .....	52
Diagrama 25: Modificar Proveedores .....	53
Diagrama 26: Eliminar Proveedores .....	54
Diagrama 27: Buscar Proveedores.....	55
Diagrama 28: Diagrama de Clases .....	56
Diagrama 29: Diagrama de Navegación .....	57
Diagrama 30: Diagrama de Interfaz .....	57
Diagrama 31: Diagrama Secuencia Ingresar Clientes .....	60
Diagrama 32: Diagrama Secuencia Modificar Clientes .....	61
Diagrama 33: Diagrama Secuencia Modificar Clientes .....	62
Diagrama 34: Diagrama Secuencia Buscar Clientes .....	63
Diagrama 35: Diagrama Secuencia Ingresar Productos .....	64
Diagrama 36: Diagrama Secuencia Modificar Productos.....	65
Diagrama 37: Diagrama Secuencia Eliminar Productos .....	66
Diagrama 38: Diagrama Secuencia Buscar Productos .....	67
Diagrama 39: Diagrama Secuencia Ingresar Usuarios .....	68

Diagrama 40: Diagrama Secuencia Modificar Usuarios .....	69
Diagrama 41: Diagrama Secuencia Eliminar Usuarios .....	70
Diagrama 42: Diagrama Secuencia Buscar Usuarios .....	71
Diagrama 43: Diagrama Secuencia Ingresar Órdenes de Compra .....	72
Diagrama 44: Diagrama Secuencia Modificar Órdenes de Compra .....	73
Diagrama 45: Diagrama Secuencia Eliminar Órdenes de Compra .....	74
Diagrama 46: Diagrama Secuencia Buscar Órdenes de Compra .....	75
Diagrama 47: Diagrama Secuencia Ingresar Proveedores.....	76
Diagrama 48: Diagrama Secuencia Modificar Proveedores .....	77
Diagrama 49: Diagrama Secuencia Eliminar Proveedores .....	78
Diagrama 50: Diagrama Secuencia Buscar Proveedores.....	79
Diagrama 51: Modelo Físico BDD .....	80

## Índice Ilustraciones

Ilustración 1: Rational Unified Process.....	8
Ilustración 2: Ciclo de vida del proyecto.....	9
Ilustración 3: Extreme Programming .....	12
Ilustración 4: Lean Development.....	13
Ilustración 5: Crystal Methodologies .....	13
Ilustración 6: Elementos SCRUM .....	16

## **CAPÍTULO 1**

### **1. Introducción**

El levantamiento de información es parte fundamental al momento de la ejecución de procesos operativos en una empresa, por ejemplo el manejo de productos, clientes y órdenes de compras; es indispensable la ejecución de estos procesos en una forma eficiente y oportuna. En ocasiones este proceso se lleva a cabo de una forma manual, utilizando formularios que el personal dedica tiempo tanto para la toma de datos como para el levantamiento de los mismos.

Al momento del procesamiento de la información sea esta de una forma manual o automática, puede presentar errores o resultados incorrectos, como errores de caligrafía o de interpretación de datos, errores de transcripción, pérdidas de formularios debido a la manipulación, entre otros.

Como respuesta a esta problemática y con los avances tecnológicos que existe en el mercado actualmente, los cuales disminuyen significativamente el riesgo de errores que presenta un proceso manual. Estos métodos agilitan el procesamiento de información siendo este proceso en tiempo real y sin depender de un sistema operativo o de un programa dedicado al mismo.

Siendo la Web una herramienta informática que utilizaremos en la elaboración de este proyecto, para no depender de un software específico para su correcto funcionamiento, de no depender exactamente de un computador fijo o portátil para la administración de los contenidos del mismo; la web una herramienta ágil, oportuna que se la puede usar en un teléfono móvil, Tablet, sin necesidad de cables o estar atado a un sistema.

En este contexto la empresa Ferretería Tobar Nolivos Cía. Ltda. busca una herramienta informática que permita: gestionar en forma ágil, oportuna y eficiente los procesos de recolección de información de sus productos, clientes y órdenes de compra, aprovechando al máximo la tecnología y que esta nueva herramienta se convierta en poco tiempo en una alternativa viable

## **1.1. Problema**

Ferretería Tobar Nolivos Cía. Ltda. empresa dedicada a la importación y comercialización de materiales para la construcción durante 39 años; en todo este tiempo la empresa es líder en la comercialización de sus productos en la ciudad de Quito, que tiene como objetivo a mediano plazo convertirse en una empresa que abarque el mercado de ventas al por mayor de materiales de construcción a nivel nacional, siendo este uno de los principales motivos para que la empresa decida implementar un sistema informático que permita la mejora de sus procesos en la administración de información.

Actualmente la empresa no cuenta con un sistema que le ayude a organizar sus procesos, en lo que tiene que ver con: registro de órdenes de compra, clientes y productos. La creciente demanda de clientes y de los servicios que presta la empresa hace que la empresa Ferretería Tobar Nolivos Cía. Ltda. decida auspiciar el presente proyecto, con la finalidad de obtener una herramienta informática propia que le permita gestionar en forma ágil, oportuna y eficiente sus procesos de registro de órdenes de compra, clientes y productos.

Aprovechando al máximo la tecnología disponible en el mercado tanto en el sector de la informática como de la tecnología móvil y hacer que la nueva herramienta se convierta en poco tiempo en una alternativa altamente viable para el control de sus procesos en tiempo real.

## **1.2. Justificación**

Se determina que el principal objetivo es brindar un sistema para optimizar los procesos y reducir los gastos generados en su ejecución. Además se determina que la principal deficiencia en la ejecución de los procesos se encuentra ligadas a problemas de tecnologías de información, por lo que es necesario enfocarse en resolver estos problemas para optimizar a la brevedad posible los procesos antes mencionados.

Se plantea la utilización del Método Ágil de Desarrollo “SCRUM”, aplicado para el Análisis, Desarrollo e Implementación de una aplicación web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferretería Tobar Nolivos Cía. Ltda.

Este proyecto permitirá a Ferretería Tobar Nolivos Cía. Ltda., desarrollar e implantar en corto tiempo una aplicación web propia, estableciendo un marco de trabajo eficiente para el desarrollo y posterior mantenimiento, optimizar la utilización de recursos y brindar un servicio más eficiente a sus clientes actuales y nuevos.

Los principales motivos que han llevado a la elección de SCRUM son:

- Es uno de los métodos de gestión de proyectos que permite un manejo apropiado de las expectativas del cliente, basadas en resultados tangibles.
- Emplea la estructura de desarrollo ágil, incremental basada en iteraciones y revisiones.
- Es un modo de desarrollo de carácter adaptable más que predictivo.

### **1.3. Alcance**

En este contexto, se establecen los aspectos que debe abarcar la ejecución del presente proyecto, así como los alcances del nuevo sistema informático para el Análisis, Desarrollo e Implementación de una aplicación web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferretería Tobar Nolivos Cía. Ltda., esto es:

- Estudio y aplicación del Método Ágil “SCRUM” aplicado a la implantación de una nueva aplicación web para Ferretería Tobar Nolivos Cía. Ltda., el mismo que le permitirá gestionar los procesos de registro de órdenes de compra, clientes y productos.
- Establecer SCRUM como el marco de trabajo oficial de Ferretería Tobar Nolivos Cía. Ltda. para el desarrollo de nuevas funcionalidades y/o mantenimiento de la aplicación, que no se ha establecido como alcance de este proyecto.
- Realizar las pruebas técnicas y funcionales de la aplicación, en base a las características del sistema.

La nueva aplicación brindara todas las funcionalidades descritas anteriormente, permitirá gestionar no solo el proceso en sí, sino también los recursos utilizados (equipos, usuarios, etc.), optimizara la utilización de recursos tecnológicos con que cuenta la empresa. Finalmente la nueva aplicación web será desarrollada en una arquitectura distribuida, que brinde escalabilidad y permita la implementación de nuevos procesos futuros que no han sido definidos como alcance de este proyecto.

## **1.4. Objetivos**

### **1.4.1. Objetivos Específicos**

- Revisar y analizar los conceptos relacionados a las metodologías de desarrollo de software.
- Analizar el método SCRUM y su aporte a la ejecución del presente proyecto.
- Diseñar la arquitectura a emplear en la Aplicación Web para la administración de los procesos.
- Establecer SCRUM como el marco de trabajo para el desarrollo de la aplicación web propuesta así como de futuras implementaciones de funcionalidades y/o mantenimiento de la misma.
- Desarrollar las pruebas e implementación de la aplicación web propuesta.

### **1.4.2. Objetivos Generales**

Implementar una aplicación web en la empresa Ferretería Tobar Nolivos Cía. Ltda. que permita mejorar y controlar sus procesos de registro de órdenes de compra, registro de clientes y registro de productos.

## **CAPÍTULO 2**

### **2. Marco Teórico**

Este capítulo nos brindara una descripción del marco teórico poniendo como referencia al método ágil de desarrollo de software SCRUM. En el ámbito de la ingeniería de software existe un debate entre los partidarios de las metodologías tradicionales y aquellos que apoyan las ideas emanadas del Manifiesto Ágil. (Agile Alliance, 2014)

Por un lado las metodologías tradicionales han estado presentes durante mucho tiempo, y precisamente no se han distinguido por ser muy exitosas; puesto que generan problemas en retrasos por falta de acuerdos con el cliente, el coste de los cambios es muy alto y genera mucho gasto a nuestro cliente, no ofrece una buena solución para proyectos donde el entorno es volátil y donde los requisitos no se conocen con exactitud.

Por otro lado, los proyectos que utilizan las metodologías ágiles han sido pensadas para aquellos proyectos en los cuales los equipos de desarrollo son pequeños, de corto plazo, con requisitos volátiles y basados en nuevas tecnologías.

Estas metodologías ágiles están especialmente orientadas a proyectos que necesitan soluciones a medida, con una elevada simplificación en términos de tiempo y recurso, sin dejar de lado el aseguramiento de la calidad del producto.

Estas metodologías se centran en el factor humano y el producto de software; es decir, le dan mayor importancia al desarrollo incremental del software con iteraciones cortas en tiempo.

## **2.1. Metodologías Tradicionales de Desarrollo de Software**

En sus inicios el desarrollo de software se llevaba a cabo de una forma totalmente rudimentaria. El crecimiento en la demanda de sistemas cada vez más complejos, asociado a la inmadurez del propio sector informático, así como también la falta de recursos y métodos provocó lo que se llamó una crisis del software (Bauer, 1992) en los años 1965 y 1985.

Durante esta época muchos proyectos importantes superaban ampliamente los presupuestos y fechas estimadas para su ejecución, por lo que las pérdidas iban mucho más allá del ámbito económico.

La importancia adquirida por los sistemas de computación dio lugar a una fuerte necesidad de mejorar el proceso para llevar los proyectos de desarrollo a la meta deseada.

Para resolver este problema fue necesario la concepción y fundamentos de metodologías existentes en otras áreas y adaptarlas al desarrollo de software. Esta adopción llevó a dividir el proceso de desarrollo en etapas generalmente secuenciales, lo que en algo ayudó en la latente necesidad de formalizar los procesos de desarrollo de software.

Esta etapa de crisis fue superada no únicamente por la mayoría en la gestión de los proyectos sino también por los progresos que se estaban dando en los procesos de diseño y metodologías de desarrollo.

La solución definitiva al problema de la planificación, previsión de costes y aseguramiento de la calidad en el proceso de desarrollo de software está ligada principalmente a la aparición de herramientas, metodologías y tecnologías.

Entre las principales metodologías de este tipo se encuentra RUP y MSF, que centran su atención en llevar una documentación exhaustiva de todo el proyecto y el cumplimiento estricto de un plan de proyecto, definido todo esto, en la fase inicial del proyecto.

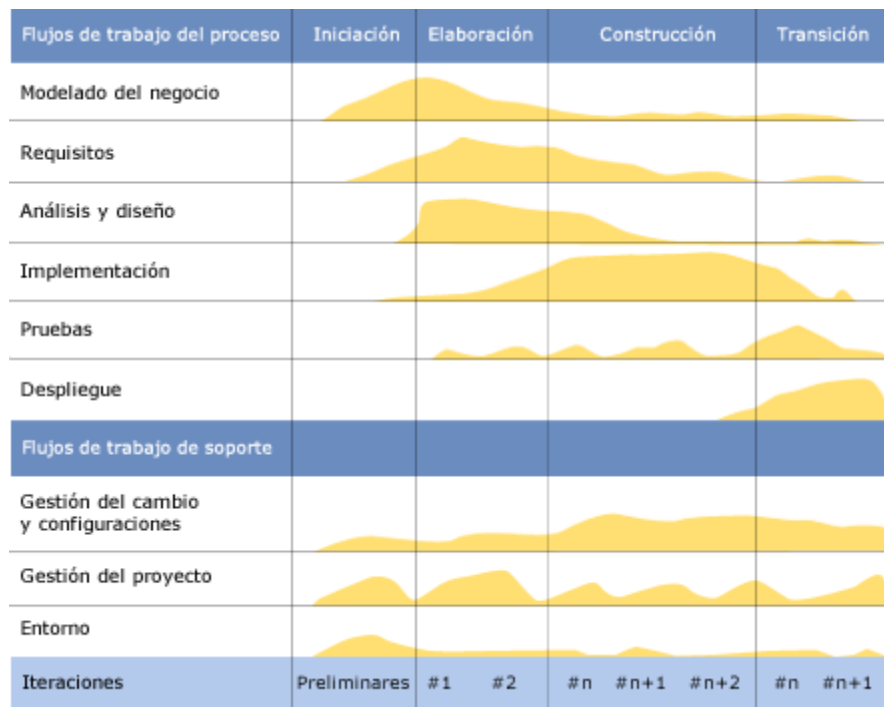
Otras características importantes dentro de este enfoque son los altos costos de implementar un cambio y las dificultades de aplicarlo en proyectos donde el entorno es cambiante.

## 2.1.1. Principales Metodologías Tradicionales

### 2.1.1.1. Rational Unified Process (RUP)

Rational Unified Process provee un acercamiento disciplinado para asignar tareas y responsabilidades dentro de una organización de desarrollo (Kruchten, 1999). Su objetivo es asegurar la producción de software de alta calidad que satisfaga los requerimientos de los usuarios finales respetando el cronograma y presupuesto. Fue desarrollado por Rational Software, y está integrado con toda la suite de herramientas Rational.

Puede ser adaptado y extendido para satisfacer las necesidades de la organización que lo adopte, es guiado por casos de uso y centrado en la arquitectura, utiliza UML como lenguaje de notación.



**Ilustración 1:** Rational Unified Process

**Fuente:** (Kruchten, 1999)

### 2.1.1.2. Microsoft Solution Framework (MSF)

MSF es un compendio de las mejores prácticas en cuanto a administración de proyectos. Más que una metodología rígida de administración de proyectos, MSF es una serie de modelos que puede adaptarse a cualquier proyecto de tecnología de información.

MSF se divide a todo proyecto separándolo en cinco fases principales:

- Visión y Alcances
- Planificación
- Desarrollo
- Estabilización
- Implantación



**Figura 2.2:**

**Ilustración 2:** Ciclo de vida del proyecto

**Fuente:** (Keeton, 2006)

## **2.2. Metodologías Ágiles de Desarrollo de Software**

### **2.2.1. Manifiesto Ágil**

El manifiesto ágil (Agile Alliance, 2014) está fundamentado en los siguientes puntos:

- Al individuo y las iteraciones del equipo de desarrollo sobre el proceso y las herramientas. La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo de trabajo antes que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que este configure su propio entorno de desarrollo en base a sus necesidades.
- Desarrollar software que funciona más que conseguir una buena documentación. La regla a seguir es “no producir documentos a menos que sean necesarios de forma inmediata para tomar en decisión importante”. Estos documentos deben ser cortos y centrarse en lo fundamental.
- La colaboración con el cliente más que en la negociación de un contrato. Se propone que exista una iteración constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.
- Responder a los cambios más que seguir estrictamente un plan. La habilidad de responder a los cambios que puedan surgir a los largo del proyecto como por ejemplo cambios en los requisitos, en la tecnología, en el equipo, etc. Determina también el éxito o fracaso del mismo; por lo tanto la planificación no debe ser estricta sino flexible y abierta

Los valores enumerados anteriormente inspiran los doce principios del manifiesto los cuales son:

- I. La prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor.
- II. Dar la bienvenida a los cambios. Se capturan los cambios para que el cliente tenga una ventaja competitiva.
- III. Entregar frecuentemente software que funcione desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas.
- IV. La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto
- V. Construir el proyecto en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir finalizar el trabajo.
- VI. El dialogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo.
- VII. El software que funciona es la medida principal de progreso.
- VIII. Los procesos agiles promueven un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener una paz constante.
- IX. La atención continua a la calidad técnica y al buen diseño mejora la agilidad.
- X. La simplicidad es esencial.
- XI. Las mejoras arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos.
- XII. En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto ajusta su comportamiento.

## **2.2.2. Principales Metodologías Ágiles**

### **2.2.2.1. Extreme Programming**

Extreme Programming o Programación Extrema es uno de los llamados procesos o metodologías ágiles de desarrollo de software (Beck, 2004) . Consiste en un conjunto de prácticas que a lo largo de los años han demostrado ser las mejores prácticas de desarrollo de software, llevadas al extremo, fundamentadas en los valores de las metodologías ágiles.

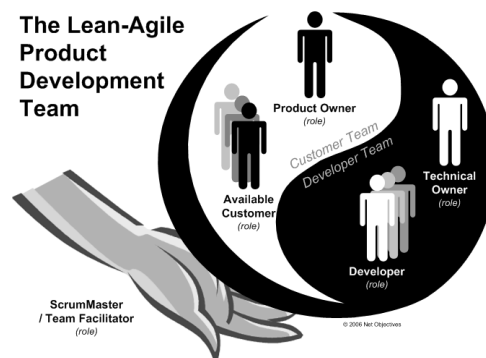


**Ilustración 3:** Extreme Programming

**Fuente:** (Programming, 2014)

### **2.2.2.2. Lean Development**

Definida por Bob Charette a partir de su experiencia en proyectos con la industria japonesa del automóvil en los años 80 y utilizada en numerosos proyectos de telecomunicaciones en Europa. En Lean Development, los cambios se consideran riesgos, pero si se manejan adecuadamente se pueden convertir en oportunidades que mejoren la productividad del cliente. Su principal característica es introducir un mecanismo para implementar dichos cambios.



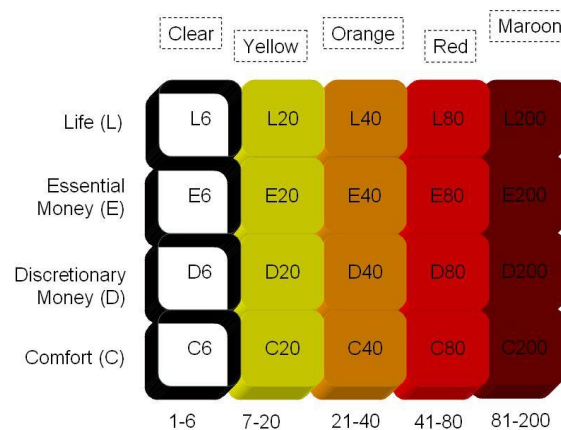
**Ilustración 4:** Lean Development

**Fuente:** (Poppendieck, 2003)

### 2.2.2.3. Crystal Methodologies

Se trata de un conjunto de un conjunto de metodologías para el desarrollo de software caracterizadas por estar centradas en las personas que componen el quipo y la reducción al máximo del número de artefactos producidos. Han sido desarrolladas por Alistair Cockburn.

El desarrollo de software se considera un juego cooperativo de invención y comunicación, limitado por los recursos a utilizar. El equipo de desarrollo es un factor clave, por lo que se deben invertir esfuerzos en mejorar sus habilidades y destrezas, así como tener políticas de trabajo en quipo bien definidos. Estas políticas dependerán del tamaño del equipo estableciéndose una clasificación por colores.



**Ilustración 5:** Cystal Methodologies

**Fuente:** (Agile Modeling Practices, 2009)

## **2.3. SCRUM**

### **2.3.1. Introducción**

SCRUM es un método ágil de gestión de proyectos cuyo objetivo principal es elevar al máximo la productividad en un equipo de desarrollo. Reduce al máximo las actividades no orientadas a producir software funcional y produce resultados en periodos cortos de tiempo. Como método en fática valores y prácticas de gestión, sin pronunciarse sobre requerimientos, prácticas de desarrollo, implementación y demás cuestiones técnicas. Más bien delega completamente al equipo la responsabilidad de decidir la mejor manera de trabajar para se lo mas productivos posibles (Palacio, 2008).

La palabra SCRUM procede de la terminología del juego de rugby, donde SCRUM se define como el acto de preparar el avance del equipo en unidad pasando la pelota entre uno y otro jugador.

SCRUM fue desarrollado por Jeff Sutherland y elaborado más formalmente por Ken Schwaber, ellos se dieron cuenta de que un proceso necesita ser adaptable al cambio, y que no sea un proceso algo predictivo puesto que algunos procesos son definidos y solo funciona para atacar problemas definidos con gente y ambientes ya establecidos. Tomar el cambio como una forma de entregar al final del desarrollo algo más cercano a la verdadera necesidad del cliente fue para ese entonces la principal solución.

SCRUM no se trata únicamente de un método para desarrollo de software, sino que puede ser aplicado teóricamente a cualquier contexto en donde un grupo de personas necesitan trabajar juntos para lograr una meta definida.

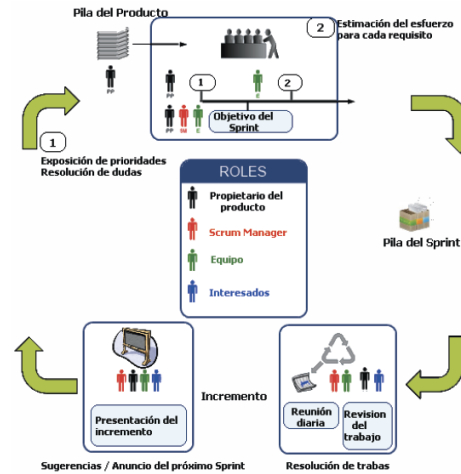
Como metodología para desarrollo de software, se basa en los principios ágiles mencionados anteriormente, pero es necesario complementar en el ámbito de desarrollo de software.

### **2.3.2. La Esencia de SCRUM**

- Más que una metodología de desarrollo, SCRUM es una herramienta para gestionar proyectos.
- Está basado en el hecho de que el trabajo es realizado por equipos auto organizados y auto dirigidos, logrando motivación, responsabilidad y compromiso.
- Es un proceso constructivo, iterativo e incremental donde las iteraciones tienen una duración fija.
- Contiene definición de roles, prácticas y productos de trabajo escritas de forma simple y soporta en un conjunto de valores y principios

### **2.3.3. Elementos del SCRUM**

- Roles
  - Product Owner (Propietario del producto)
  - SCRUM Master
  - Team (Equipo)
- Poda de requerimientos
- Product Backlog
- Sprint
  - Planificación
  - Sprint Backlog
  - SCRUM
  - Estimaciones
  - Builds continuos
  - Revisión del Sprint
  - Reunión retrospectiva
- Valores
  - Foco, comunicación, respeto y coraje



**Ilustración 6:** Elementos SCRUM

Fuente: (Palacio, 2008)

### 2.3.3.1. Roles

El nombre de los miembros del equipo y los extremos se deriva de una típica fabula agilista: un cerdo y una gallina discutían que nombre ponerle a un nuevo restaurante; la gallina propuso “Jamos y Huevos”. “No, gracias”, respondió el cerdo, “Yo estaré comprometido, mientras tu solo implicada”.

SCRUM diferencia claramente entre estos dos grupos para garantizar que quienes tienen la responsabilidad tienen también la autoridad necesaria para poder lograr el éxito, y que quienes no tienen la responsabilidad, los observadores externos, no produzcan interferencias innecesarias.

Comprometido en el Proyecto	Implicados en el Proyecto
- Dueño del producto	- Marketing
- Equipo	- Comercial
-SCRUM Master	- Otros

**Tabla 1:** Roles SCRUM

Fuente: Diego Tobar

**i. Product Owner (Dueño del producto)**

Representa a todos los interesados en el producto final. Es el responsable oficial del proyecto, gestión, control y visibilidad de la lista de acumulación o lista de retraso del producto (Product Backlog). Toma las decisiones finales de las tareas asignadas al registro y convierte sus elementos en rasgos a desarrollar. Sus áreas de responsabilidad son:

- Financiación del proyecto
- Requisitos del sistema
- Retorno de la inversión del proyecto
- Lanzamiento del proyecto

**ii. SCRUM Master (Líder del proyecto)**

Responsable del proceso SCRUM, de cumplir la meta y resolver los problemas. Así como también de asegurarse que el proyecto se lleve a cabo de acuerdo con las prácticas, valores y reglas de SCRUM para que progrese el proyecto según lo planificado

Interactúa con el cliente y el equipo, además de coordinar los encuentros diarios y es el encargado de eliminar eventuales obstáculos. Debe ser miembro del equipo y trabajar a la par.

**iii. Team (Equipo)**

Responsable de transformar el Backlog de la iteración en un incremento de la funcionalidad del software. Tiene autoridad para reorganizarse y definir las acciones necesarias o sugerir remoción de impedimentos.

La dimensión del equipo total de SCRUM no debería ser superior a veinte. El número ideal es diez, más o menos dos lo mínimo. Si hay más, lo más recomendable es formar varios equipos, no hay una técnica oficial para coordinar equipos, divididos en SCRUMs de SCRUMS, definiendo un equipo central que se encarga de la coordinación, las pruebas cruzadas y la rotación de los miembros (Palacio, 2008).

### **2.3.3.2. Poda de requerimientos**

La poda de requerimientos es una buena práctica implícita en modelos ágiles, y consiste en hacer lo que el cliente realmente desea.

La primera actividad en un proyecto manejado con SCRUM es armar una lista exhaustiva de los requerimientos originales del sistema. Posteriormente se realiza una revisión para evaluar que requerimientos son realmente necesarios, cuales pueden posponerse y cuales eliminarse. Para ello debe identificarse un representante con capacidad de decisión, priorizar los requerimientos en base a su importancia y acordar cuales son los prioritarios para la fecha de entrega.

### **2.3.3.3. Product Backlog**

Con los requerimientos priorizados y podados se arma el Backlog de producto, este es una forma de registrar y organizar el trabajo pendiente para el producto entre ellas están las actividades y requerimientos.

Es un documento dinámico que incorpora constantemente las necesidades del sistema. Por lo tanto, nunca llega a ser una lista completa y definitiva, se mantiene durante todo el ciclo de vida y es responsabilidad del Product Owner.

### **2.3.3.4. Sprint**

SCRUM está basado en el control empírico de procesos. Se utiliza cuando la capacidad de predicción es vaga, la incertidumbre es alta o el proceso es demasiado complejo para ser modelado y definido.

En el enfoque empírico de control de procesos se establecen reglas simples y se crea una disciplina de inspección frecuente para adaptarse rápidamente a situaciones no previstas y problemas.

Un Sprint es el periodo de tiempo durante el que se desarrolla un incremento de funcionalidad, constituye el núcleo de SCRUM que divide de esta forma el desarrollo de un proyecto en un

conjunto de pequeñas carreras.

### **2.3.3.5. Planificación**

Se planifica en detalle el trabajo al inicio de cada Sprint asumiendo que los objetivos no van a cambiar durante el mismo. De esta manera se atenúa el riesgo.

Aspectos a tener en cuenta sobre la planificación de un Sprint:

- Una determinación general de alcance, frecuentemente basada en una estructura de división del trabajo.
- Estimaciones de esfuerzo de alto nivel realizadas durante la etapa de concepción del proyecto.
- Esfuerzo dedicado a labores de soporte o de preparación de los ambientes requeridos por el proyecto.
- Esfuerzo asociados a las reuniones diarias, de planificación y de revisión.
- Requerimientos de recursos de infraestructura o logísticos.
- Habilidades presentes y necesarias en el equipo.
- Restricciones asociadas al conocimiento del negocio, la tecnología o externas (legales, reglamentarias, estándares, etc.).

El rol del SCRUM Master durante la planificación es:

- Dirigir la planificación.
- Facilitar acuerdos entre el equipo y el Product Owner del proyecto.
- Registrar problemas y riesgos detectados durante la planificación.
- Registrar las tareas, asignaciones y estimaciones.
- Iniciar el Backlog del Sprint.

### **i. Sprint Backlog**

Trabajo o tareas determinadas por el equipo para realizar en un Sprint:

- Tareas a convertir en producto funcional.
- Las tareas se estiman en una duración entre 1 y 20 horas de trabajo. Las de mayor duración deben intentar descomponerse en sub tareas en el rango de tiempo indicado.
- La estimación como el avance de las tareas se actualiza día a día.

### **ii. SCRUM Diario**

SCRUM asume que el proceso es complejo y que es necesario inspeccionarlo frecuentemente, por eso realiza una reunión diaria de seguimiento. El encuentro diario impide caer en el dilema señalado por Fred Brooks: “¿Cómo es que un proyecto puede atrasarse un año? Un día a la vez”

El foco de la reunión es determinar el avance en las tareas y detectar problemas que estén haciendo lento el progreso del equipo o que eventualmente impidan a un equipo cumplir con la meta del Sprint. La idea es que ningún problema quede sin resolver o por lo menos, sin iniciar alguna acción de respuesta dentro de las 24 horas después de su detección.

La reunión es además un espacio definido para que cada miembro del equipo comunique a los demás el estado de su trabajo.

El rol del SCRUM Master durante el SCRUM es:

- Dirigir la reunión y mantiene el foco.
- Hacer preguntas para aclarar dudas.
- Registrar los problemas para su resolución después de la reunión.
- Asegura que los miembros cuenten con el ambiente adecuado para la reunión.

### **iii. Estimaciones**

Las estimaciones se realizan por primera vez en la reunión de planificación al inicio del Sprint. Luego las tareas se estiman todos los días y se registran sus cambios en el Backlog del Sprint; esta actividad puede ser realizada inmediatamente antes o después del SCRUM diario.

### **iv. Builds Continuos y Pruebas Básicas**

La estrategia que generalmente se utiliza es la de Builds continuos y pruebas básicas para la funcionalidad del sistema se basa en:

- Los programadores desarrollan según el Backlog del Sprint, y al finalizar notifican al integrador.
- El integrador toma el código y lo integra con el resto del producto
- Se compila el software y se prueba superficialmente el producto, para verificar la estabilidad de la versión.
- Si se encuentran problemas se devuelve al desarrollador.
- Si no se han encontrado errores se notifica al equipo que hay una nueva versión estable del código para usar como base.

### **v. Revisión del Sprint**

El objetivo de la reunión de revisión es presentar el producto o porción del producto desarrollado por el equipo a los usuarios. La reunión se utiliza para detectar inconformidades mayores que se vuelven elementos del Back log de Producto y que eventualmente se resuelven en el siguiente Sprint.

A la reunión asisten el equipo, el SCRUM Master, el Product Owner y todas las personas implicadas en el proyecto. Esta revisión se convierte en un demo del producto con las nuevas funcionalidades desarrolladas durante el Sprint.

#### **vi. Reunión Retrospectiva**

SCRUM involucra el concepto de mejora continua a través de las reuniones de retrospectiva. Las reuniones buscan detectar los puntos positivos y negativos del Sprint para generar propuestas de mejora para futuros Sprints.

Las reuniones de retrospectiva son el concentrador del aprendizaje organizacional sobre el SCRUM donde los puntos positivos y negativos se registran y se definen ítems de acción para cada uno.

A la reunión asisten el equipo, el SCRUM Master y opcionalmente el Product Owner.

#### **2.3.3.6. Valores**

##### **i. Foco**

Los individuos y sus iteraciones son más importantes que el proceso o las herramientas. El recurso humano es el principal factor de éxito de un proyecto de software.

##### **ii. Comunicación**

SCRUM pone en comunicación directa y continua a clientes y desarrolladores. El cliente se integra en el equipo para establecer prioridades y resolver dudas, de esta forma ve el avance día a día, y es posible ajustar la agenda y las funcionalidades de forma consecuente.

##### **iii. Respeto**

SCRUM diferencia claramente entre dos grupos para garantizar que quienes tienen la responsabilidad tienen también la autoridad necesaria para poder lograr el éxito, y que quienes no tienen la responsabilidad, los observadores externos, no produzcan interferencias innecesarias.

##### **iv. Coraje**

El coraje implica saber tomar decisiones difíciles, reparar un error cuando se detecta y mejorar el código siempre que el feedback y las sucesivas iteraciones se manifiesten susceptibles de mejora.

## **2.4. Definición General del Desarrollo Web.**

### **2.4.1. Desarrollo Web**

La aparición de aplicaciones y sitios web conlleva un importante crecimiento en el desarrollo de software, que necesitan de mecanismos adecuados para que la realización de este tipo de aplicaciones satisfaga las necesidades tanto de los usuarios como de los clientes que contratan el desarrollo de este tipo de aplicaciones.

La web permite desarrollar otras capacidades y satisfacer otros servicios en el ámbito de los sistemas informáticos, de este tema se valen personas que no necesitan tener un aplicativo instalado en un computador, permitiendo mayor movilidad y acceso desde cualquier parte del mundo y no depender estrictamente de un sistema operativo en el que la aplicación se ejecute; en pocas palabras el punto decisivo es el acceso a la información independiente de la ubicación del usuario.

Por ejemplo, y según afirma (Jacobson, 2000), el modelo de proceso más adecuado para el desarrollo de software es un proceso iterativo e incremental, puesto que a diferencia de otros modelos de proceso, como por ejemplo el modelo de cascada, permite la obtención de diversas versiones del producto software antes de la entrega final del mismo y la depuración y validación progresiva del mismo, lo que sin duda redundará en un software más satisfactorio para usuarios y cliente.

### **2.4.2. Desarrollo de aplicaciones Web con metodologías ágiles**

Las metodologías ágiles son una buena elección cuando trabajamos con requisitos desconocidos o variables. Si no existen requisitos estables, no existe una gran posibilidad de tener un diseño estable y de seguir un proceso totalmente planificado, que no vaya a variar ni en tiempo ni en dinero. Un proceso adaptativo facilita la generación de prototipos y de versiones previas a la entrega final, lo cual agradara al cliente, lo que se garantiza es un menor riesgo ante la posibilidad de cambios en los requisitos (Cáceres, 2000).

Las aplicaciones web cumplen con la mayor parte de las características mencionadas, por lo que la utilización de procesos ágiles podría ser beneficioso para este tipo de desarrollo. La

necesidad del cliente al momento de contratar un desarrollo web es que su producto esté disponible en la red lo más pronto posible. Puesto que los procesos ágiles permiten obtener versiones de producto previas a la versión final, si se aplican adecuadamente estos procesos el cliente podrá disponer de forma rápida de alguna versión intermedia

### **2.4.3. Definición de una Aplicación Web**

Una aplicación Web es una interfaz o conjuntos de páginas Web que interactúan con el usuario final, de esta manera permiten el acceso a la información solicitada y se toma los datos propios del modelo de negocio, así cualquier persona puede interactuar con ella desde Internet por medio de un navegador Web (Mora, 2001).

Las características de las aplicaciones Web son:

- Fácil Acceso
- El usuario puede ingresar a la aplicación Web desde cualquier parte del mundo, donde tenga acceso a Internet.
- Pueden existir miles de clientes que acceden al mismo tiempo a dicha aplicación; por lo tanto se puede actualizar y mantener una única aplicación y todos sus clientes verán los resultados inmediatamente.
- Utilizan tecnologías como ASP.NET, AJAX, FLASH, JavaScript, HTML, JSP, PHP entre otras que brindan un gran interfaz de usuario a la aplicación Web.

#### **2.4.3.1. Estructura de las Aplicaciones Web**

Normalmente una aplicación Web en su interior es una estructura de tres capas, en la primera capa corresponde a lo que se puede visualizar en un navegador Web o más conocida como la capa de presentación, se encuentra también la capa conocida como la de negocio, la misma que puede usar en la actualidad tecnologías Web dinámicas como: PHP, JAVA, ASP.NET, por último se tiene la que corresponde a la Base de Datos o más conocida como la de accesos a datos determinada por un software de gestión de manejo de datos por ejemplo MySQL (Mora, 2001).

### **2.4.3.2. Uso de las Aplicaciones Web**

Instituciones y empresas en la actualidad brindan acceso vía Web a sus usuarios, lo cual les ayuda a lograr un desarrollo en un mercado más grande, mediante aplicaciones distribuidas creadas para una interfaz Web, lo cual permite que se interactúe con el usuario de una manera más oportuna y eficaz. De esta manera al usar este tipo de aplicaciones, no hay necesidad de que la aplicación Web sea instalada en el ordenador del usuario, la institución o empresa deberá tener instalada la aplicación en su servidor.

## **2.5. PHP**

Es un lenguaje de programación interpretado en el servidor, con una sintaxis similar a la de C++ o Java; creado originalmente en el año 1994 pero que actualmente es ampliamente usado en entornos de desarrollo web por su facilidad de uso, su integración perfecta con ficheros HTML y su versatilidad de uso en diferentes Sistemas Operativos (Instituto Celaya, 2009).

Tanto es su expansión que se calcula su uso en más de 20 millones de sitios web y un millón de servidores en todo el mundo. Aunque el lenguaje se puede usar para realizar cualquier tipo de programa, es en la generación dinámica de páginas web donde ha alcanzado su máxima popularidad. Algunas de sus características son:

- Es un lenguaje libre.
- Está disponible para muchos sistemas (GNU/Linux, Windows, UNIX)
- Tiene una extensa documentación oficial en varios idiomas.
- Al ejecutarse en el servidor, se puede usar todo tipo de máquinas con todo tipo de sistemas operativos.

Según el criterio de varios autores las siguientes bases de datos soportan programación PHP:

- Oracle
- MySQL
- ODBC

### **2.5.1. Arquitectura y Servicios**

PHP permite una arquitectura Cliente-Servidor de 3 y más capas; para el manejo y el intercambio de información; la ejecución de PHP inspecciona las consultas enviadas al servidor de Bases de Datos para administrar correctamente sus recursos y interpretar entre el Servidor Web que es el encargado de toda la interpretación de los archivos PHP y su generación a ficheros HTML.

Los servicios ofrecidos por PHP están relacionados con el objetivo del proyecto o sistema a desarrollar, su gran cantidad de bibliotecas y funciones propias desarrolladas en la web de otros autores por su tendencia Open Source; permiten desarrollar a través de la Web. Además el gran auge de sistemas manejadores de contenidos CMS y frameworks de trabajo basados en PHP, lo convierten en una gran alternativa en el desarrollo de software orientado a la Web.

## **2.6. Framework Desarrollo Web**

Los frameworks para PHP, son un conjunto de archivos PHP que vienen preparados con toda la estructura necesaria para desarrollar varios tipos de proyectos. Hay muchos frameworks de PHP; cada uno orientado a un tipo de proyecto, por lo que elegir el adecuado puede ahorrar una gran cantidad de trabajo, además se debe tomar en cuenta que framework se ajusta más a las necesidades de un proyecto de Desarrollo (Gutierrez, 2013).

Desarrollar un proyecto en PHP, muchas veces requiere escribir todo el código desde cero, lo que no es muy práctico. Para ello se puede de aplicar e implementar códigos ya desarrollados para funciones frecuentes (Insertar, Recupera Datos, Autenticarse en el sistema, etc) que se encuentran disponibles en los frameworks, que hará más óptimo el trabajo y centralizar el esfuerzo específicamente en la lógica de la aplicación (Pavón, 2008).

## **2.7. CodeIgniter**

Es un framework ligero que usa el Patrón de Diseño MVC y contiene una muy buena estructuración de sus componentes (Ellislab, 2013). Permite desarrollar proyectos mucho más rápido ya que cuenta con un rico conjunto de bibliotecas para tareas comunes, así como una interfaz sencilla y una estructura lógica para acceder a las mismas, permite enfocarse creativamente en el proyecto al minimizar la cantidad de código necesario para una tarea dada; otras aspectos importantes son:

- Está liberado bajo licencias Open Source del estilo Apache/BSD.
- Liviano puesto que el núcleo del sistema sólo requiere algunas bibliotecas muy pequeñas.
- Las bibliotecas adicionales se cargan dinámicamente.
- Las URLs generadas son claras y amigables ya que usa un enfoque basado en segmentos: ejemplo.com/noticias
- Compatibilidad con una amplia variedad de servidores y configuraciones.

### **2.7.1. Funcionalidades de CodeIgniter**

Dentro de los componentes disponibles del framework se encuentran varias funcionalidades para los diferentes escenarios posibles en el desarrollo de una aplicación Web a mediana o gran escala; por el momento, la lista de funcionalidades principales de CodeIgniter son:

- Sistema basado en Modelo-Vista-Controlador
- Amplia biblioteca de funciones "helper"
- Clases de base de datos con soporte para varias plataformas
- Soporte para base de datos con Active Record
- Validación de datos y formularios
- Seguridad y filtrado
- Administración de sesiones
- Clase para subir archivos

## **2.8. Gestor Base de Datos**

Los sistemas de gestión de bases de datos SGBD son un tipo de software específico dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.

El objetivo principal de las bases de datos es el de unificar los datos que se manejan y los programas o aplicaciones que los manejan. Anteriormente los programas se codificaban junto con los datos, es decir, se diseñaban para la aplicación concreta que los iba a manejar, lo que desembocaba en una dependencia de los programas respecto a los datos, ya que la estructura de los ficheros va incluida dentro del programa, y cualquier cambio en la estructura del fichero provocaba modificar y recompilar programas. Además, cada aplicación utiliza ficheros que pueden ser comunes a otras de la misma organización, por lo que se produce una redundancia de la información, que provoca mayor ocupación de memoria, laboriosos programas de actualización (unificar datos recogidos por las aplicaciones de los diferentes departamentos), e inconsistencia de datos (no son correctos) si los datos no fueron bien actualizados en todos los programas.

### **2.8.1. MySQL**

MySQL es un sistema gestor de bases de datos muy conocido y ampliamente usado por su simplicidad y notable rendimiento (Pavón, 2008). Aunque carece de algunas características avanzadas disponibles en otros Gestores de Bases de Datos del mercado, es una opción atractiva tanto para aplicaciones comerciales, como de entretenimiento precisamente por su facilidad de uso y tiempo reducido de puesta en marcha. Esto y su libre distribución en Internet bajo licencia GPL le otorgan como beneficios adicionales contar con un alto grado de estabilidad y un rápido desarrollo. Algunas características de MySQL son:

- Está desarrollado en C/C++.
- Se distribuyen ejecutables para cerca de diecinueve plataformas diferentes.
- La API se encuentra disponible en C, C++, Java, Perl, PHP, Python, Ruby.
- Está optimizado para equipos de múltiples procesadores.
- Es muy destacable su velocidad de respuesta.

## **2.9.Herramientas de Desarrollo**

Para el desarrollo del presente proyecto se utiliza varias herramientas relacionadas con cada una de las fases de desarrollo del sistema, tomando como inicio el Análisis y Modelado de la lógica de negocio y la estructura del Sistema (Casos de Uso), Diseño y Modelado de los Datos a utilizar (Modelo Físico y Lógico); hasta llegar a la implementación de la misma (Construcción de Consultas, Carga de Datos y Generación de código).

### **2.9.1. Power Designer 15.1**

Es una herramienta para el análisis, diseño y construcción sólida de diagramas UML además de bases de datos y un desarrollo orientado a modelos de datos a nivel físico y conceptual, que da a los desarrolladores Cliente/Servidor la más firme base para aplicaciones de alto rendimiento. Ofrece un acercamiento de diseño para optimizar las estructuras de las bases de datos, capturando el flujo de datos de la aplicación, permite crear un modelo conceptual y físico de la base de datos.

### **2.9.2. Zend Studio 9.0**

Editor web orientado a la programación de páginas PHP, con ayudas en la gestión de proyectos y depuración de código muy utilizado en proyectos bajo tecnología de servidor PHP, orientada a desarrollar aplicaciones web. El programa, además de servir de editor de texto PHP, proporciona una serie de ayudas que pasan desde la creación y gestión de proyectos hasta la depuración de código.

Zend Studio consta de dos partes en las que se dividen las funcionalidades de parte del cliente y las del servidor; permite depuraciones simples de scripts, aunque para la herramienta de depuración es necesario la parte del servidor, que instala Apache y el módulo PHP para trabajar juntos en depuración.

## **CAPÍTULO 3**

### **3. Análisis**

#### **3.1. Identificación de Requerimientos**

##### **3.1.1. Especificación de Requerimientos de Software ERS**

###### **Propósito:**

El documento presenta la especificación de requerimientos de software del sistema en la empresa Ferretería Tobar Nolivos Cía. Ltda., para el control de los procesos de órdenes de compra, clientes y productos.

Esta especificación se basa en el estándar IEEE 830 (ERS Especificación de Requerimientos de Software). Pretende abstraer principalmente los conceptos funcionales del producto que se espera realizar. En base a los requerimientos se desarrollara una aplicación Web para automatizar los procesos. A fin de que se realicen las funciones de desarrollo a partir del contenido marcado en este documento.

El sistema Web busca solucionar una necesidad muy importante de la empresa Ferretería Tobar Nolivos Cía. Ltda., con respecto a la organización de su información y procesos que se llevan a cabo manualmente.

El documento ERS está constituido por elementos esenciales, entre los cuales se pueden mencionar los siguientes:

- El personal involucrado dentro del desarrollo del proyecto.
- La perspectiva del producto mediante la visión general de las características de software.
- La funcionalidad del producto con las características más importantes.
- Los requisitos no funcionales que determinan aspectos importantes del sistema.
- Los requerimientos funcionales que describen los requerimientos del cliente y el alcance del proyecto.

El documento de Especificación de Requerimientos de Software se encuentra adjunto en los Anexos del presente documento, revisar el Anexo A.

## **3.2. Análisis de Factibilidad**

### **3.2.1. Factibilidad Técnica**

La empresa Ferretería Tobar Nolivos Cía. Ltda. es el auspiciante del desarrollo del proyecto de tesis, esta entidad ayudará y facilitará el proceso de entrega de información necesaria para la satisfactoria culminación del mismo. La empresa colaborará para el desarrollo del sistema con los siguientes puntos:

- Entrevistas con el personal de la empresa que manejaran el sistema para saber todos los problemas que les aquejan y las necesidades que tienen en cuanto a la automatización de las fases.
- Acceso a la información que maneja la empresa, tal como: clientes, productos, órdenes de compra, características de cada uno de los documentos y procesos que manejan.

### **3.2.2. Factibilidad Operativa**

- El desarrollador tiene el conocimiento necesario y suficiente en la creación de aplicaciones web, conocimientos principalmente relacionados con las herramientas de desarrollo a utilizar y en el motor de Base de Datos.
- En la fase de implementación del sistema, la empresa Ferretería Tobar Nolivos Cía. Ltda. necesitará un computador, el cual hará el papel de servidor de aplicaciones y de servidor de Base de Datos

## **CAPÍTULO 4**

### **4. Diseño y Desarrollo del Software**

#### **4.1. Requerimientos Funcionales**

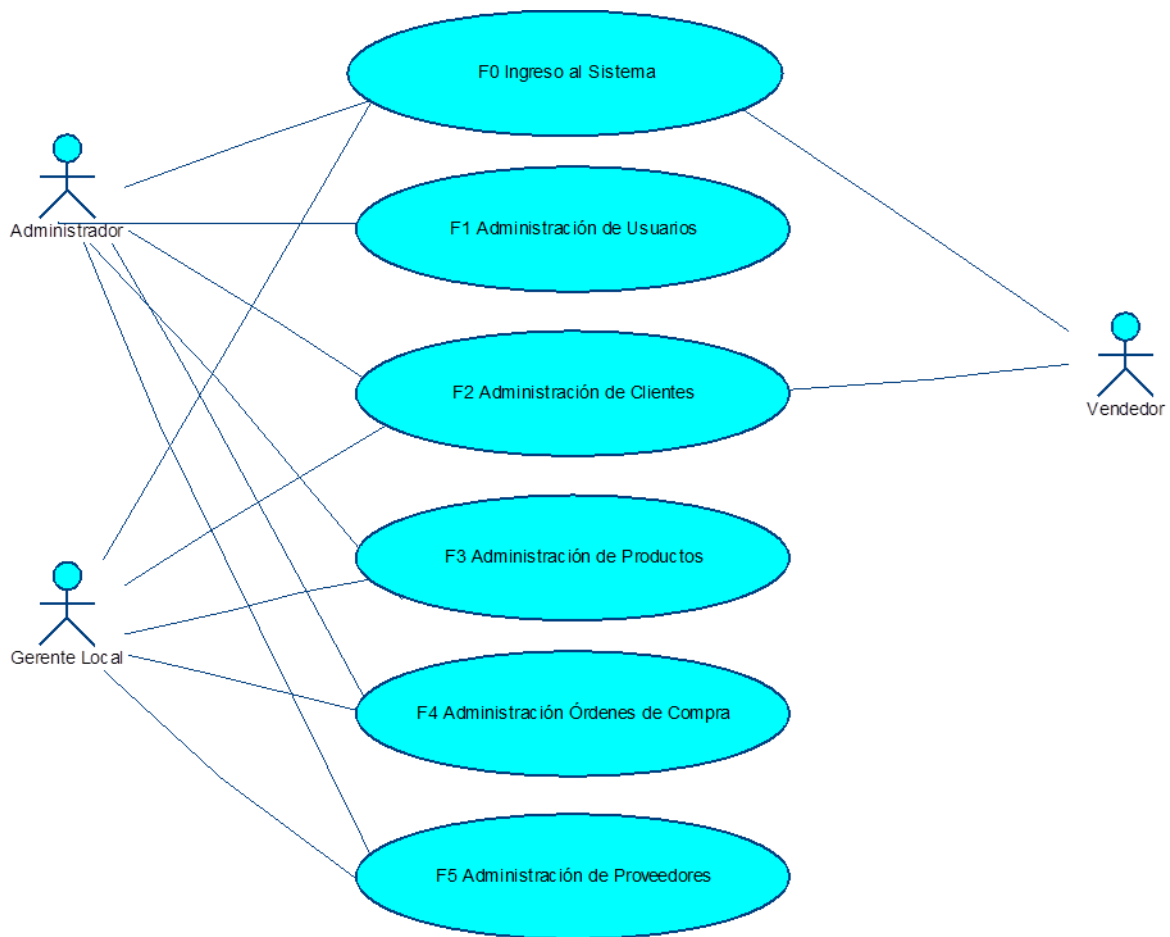
##### **4.1.1. Administraciones**

- Clientes
- Productos
- Usuarios
- Órdenes de compra
- Proveedores

##### **4.1.2. Perfiles de Usuario**

- Administrador del Sistema
- Vendedor
- Gerente Local

## 4.2. Diagrama General



**Diagrama 1:** Diagrama General

**Fuente:** Diego Tobar

### 4.3. Diagrama de Casos de Uso

#### 4.3.1. Ingreso al Sistema

Caso de Uso: Ingreso al Sistema

Identificador: F0

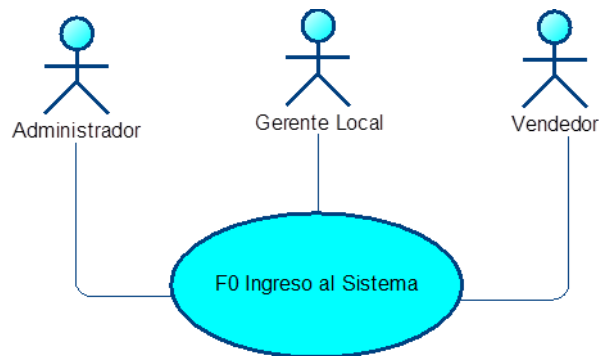


Diagrama 2: Ingreso al Sistema

Fuente: Diego Tobar

**Descripción:** Se realiza esta acción por todos los usuarios, permitirá el ingreso al sistema mediante validación de usuario y clave.

Flujo de Ingreso al Sistema				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de Ingreso	2	Valida usuario y clave	EX1 – EX2
		3	Permite ingreso	
Excepciones de Ingreso al Sistema				
ID	Descripción	Acción		
EX1	Usuario no encontrado	Mensaje de error		
EX2	Clave incorrecta	Mensaje de error		

Tabla 2: Flujo Ingreso al Sistema

Fuente: Diego Tobar

### 4.3.2. Administración de Usuarios

Caso de Uso: Administración de Usuarios

Identificador: F1

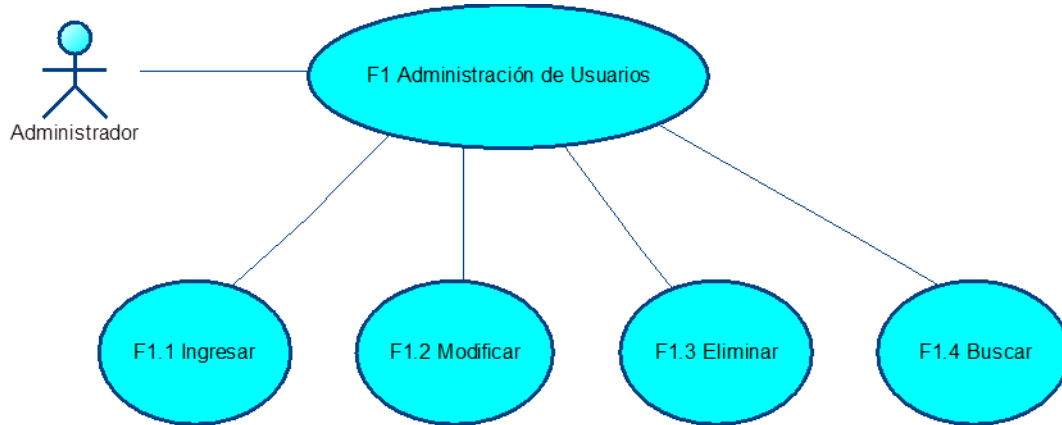


Diagrama 3: Administración de Usuarios

Fuente: Diego Tobar

**Descripción:** Permite el manejo de usuarios que tienen acceso al sistema.

#### 4.3.2.1. Ingresar Usuarios

Caso de Uso: Ingresar Usuarios

Identificador: F1.1



Diagrama 4: Ingresar Usuarios

Fuente: Diego Tobar

**Descripción:** Permite el ingreso de usuarios al sistema.

Flujo de Ingresar Usuarios				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de Ingreso	2	Presenta plantilla	
3	Ingresar datos	4	Permite ingreso	

5	Evento de ingreso de datos	6	Valida datos	EX1
		7	Ingresa datos	
<b>Excepciones de Ingresar Usuarios</b>				
<b>ID</b>	<b>Descripción</b>	<b>Acción</b>		
EX1	Datos incorrectos	Mensaje de error		

**Tabla 3:** Flujo Ingresar Usuarios

**Fuente:** Diego Tobar

#### 4.3.2.2. Modificar Usuarios

**Caso de Uso:** Modificar Usuarios

**Identificador:** F1.2



**Diagrama 5:** Modificar Usuarios

**Fuente:** Diego Tobar

**Descripción:** Permite la modificación de usuarios al sistema.

<b>Flujo de Modificar Usuarios</b>				
<b>ID</b>	<b>Actor</b>	<b>ID</b>	<b>Sistema</b>	<b>Excepción</b>
1	Genera un evento de modificación	2	Presenta plantilla	
3	Modifica datos	4	Permite modificar	
5	Evento de modificación de datos	6	Valida datos	EX1
		7	Modifica datos	

<b>Excepciones de Modificar Usuarios</b>		
<b>ID</b>	<b>Descripción</b>	<b>Acción</b>
EX1	Datos incorrectos	Mensaje de error

**Tabla 4:** Flujo Modificar Usuarios

**Fuente:** Diego Tobar

### 4.3.2.3. Eliminar Usuarios

**Caso de Uso:** Eliminar Usuarios

**Identificador:** F1.3



**Diagrama 6:** Eliminar Usuarios

**Fuente:** Diego Tobar

**Descripción:** Permite eliminar usuarios existentes del sistema.

<b>Flujo de Eliminar Usuarios</b>				
<b>ID</b>	<b>Actor</b>	<b>ID</b>	<b>Sistema</b>	<b>Excepción</b>
1	Genera un evento de eliminación	2	Presenta confirmación	EX1
3	Evento de eliminación de datos	4	Elimina datos	
<b>Excepciones de Eliminar Usuarios</b>				
<b>ID</b>	<b>Descripción</b>	<b>Acción</b>		
EX1	Datos incorrectos	Mensaje de error		

**Tabla 5:** Flujo Eliminar Usuarios

**Fuente:** Diego Tobar

#### 4.3.2.4. Buscar Usuarios

Caso de Uso: Buscar Usuarios

Identificador: F1.4



Diagrama 7: Buscar Usuarios

Fuente: Diego Tobar

**Descripción:** Permite buscar usuarios existentes del sistema.

Flujo de Buscar Usuarios				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de búsqueda	2	Presenta datos	EX1
Excepciones de Buscar Usuarios				
ID	Descripción	Acción		
EX1	Ausencia de Datos	Mensaje de no existencia		

Tabla 6: Flujo Buscar Usuarios

Fuente: Diego Tobar

### 4.3.3. Administración de Clientes

Caso de Uso: Administración de Clientes

Identificador: F2

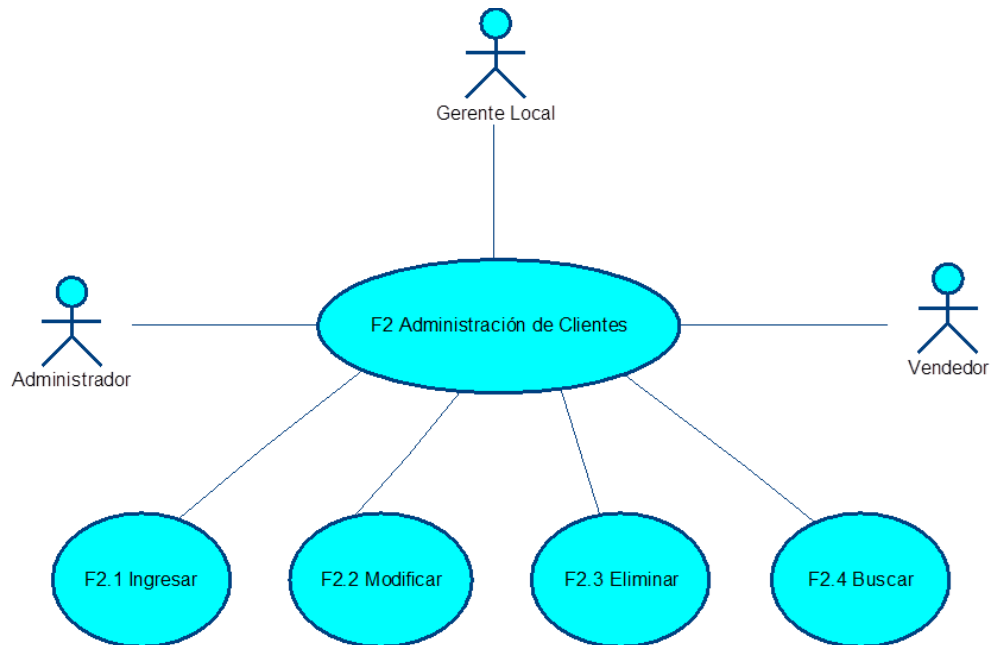


Diagrama 8: Administración de Clientes

Fuente: Diego Tobar

Descripción: Permite el manejo de clientes de la empresa.

#### 4.3.3.1. Ingresar Clientes

Caso de Uso: Ingresar Clientes

Identificador: F2.1

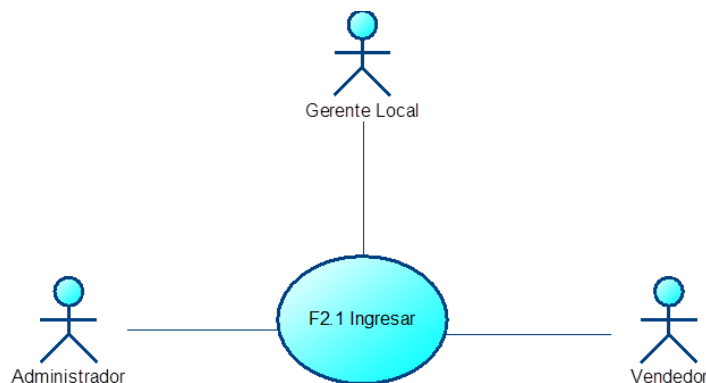


Diagrama 9: Ingresar Clientes

Fuente: Diego Tobar

**Descripción:** Permite el ingreso de clientes al sistema.

Flujo de Ingresar Clientes				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de Ingreso	2	Presenta plantilla	
3	Ingresar datos	4	Permite ingreso	
5	Evento de ingreso de datos	6	Valida datos	EX1
		7	Ingresar datos	
Excepciones de Ingresar Clientes				
ID	Descripción	Acción		
EX1	Datos incorrectos	Mensaje de error		

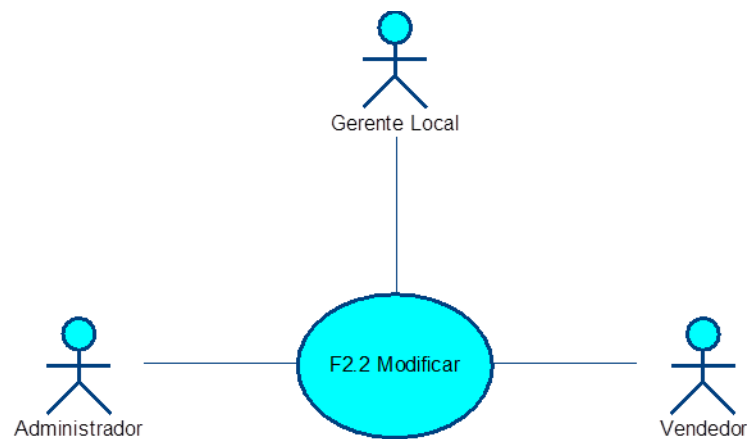
**Tabla 7:** Flujo Ingresar Clientes

**Fuente:** Diego Tobar

### 4.3.3.2. Modificar Clientes

**Caso de Uso:** Modificar Clientes

**Identificador:** F2.2



**Diagrama 10:** Modificar Clientes

**Fuente:** Diego Tobar

**Descripción:** Permite la modificación de clientes al sistema.

Flujo de Modificar Clientes				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de modificación	2	Presenta plantilla	
3	Modifica datos	4	Permite modificar	
5	Evento de modificación de datos	6	Valida datos	EX1
		7	Modifica datos	
Excepciones de Modificar Clientes				
ID	Descripción	Acción		
EX1	Datos incorrectos	Mensaje de error		

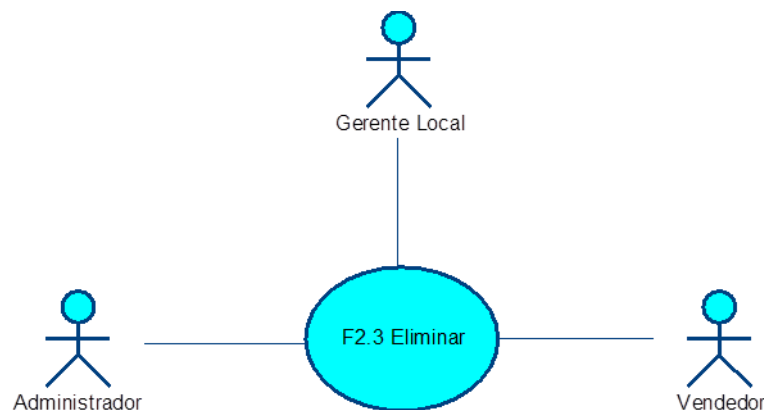
**Tabla 8:** Flujo Modificar Clientes

**Fuente:** Diego Tobar

### 4.3.3.3. Eliminar Clientes

**Caso de Uso:** Eliminar Clientes

**Identificador:** F2.3



**Diagrama 11:** Eliminar Usuarios

**Fuente:** Diego Tobar

**Descripción:** Permite eliminar clientes existentes del sistema.

Flujo de Eliminar Clientes				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de eliminación	2	Presenta confirmación	EX1
3	Evento de eliminación de datos	4	Elimina datos	
Excepciones de Eliminar Clientes				
ID	Descripción	Acción		
EX1	Datos incorrectos	Mensaje de error		

Tabla 9: Flujo Eliminar Usuarios

Fuente: Diego Tobar

#### 4.3.3.4. Buscar Clientes

**Caso de Uso:** Buscar Clientes

**Identificador:** F2.4

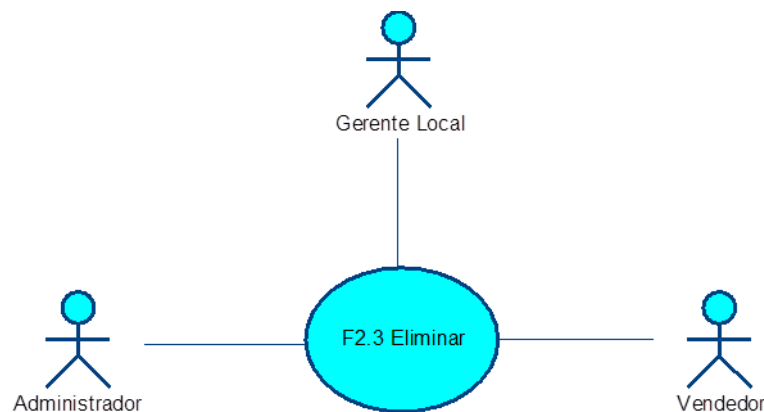


Diagrama 12: Buscar Clientes

Fuente: Diego Tobar

**Descripción:** Permite buscar clientes existentes del sistema.

Flujo de Buscar Clientes				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de búsqueda	2	Presenta datos	EX1
Excepciones de Buscar Clientes				
ID	Descripción		Acción	
EX1	Ausencia de Datos		Mensaje de no existencia	

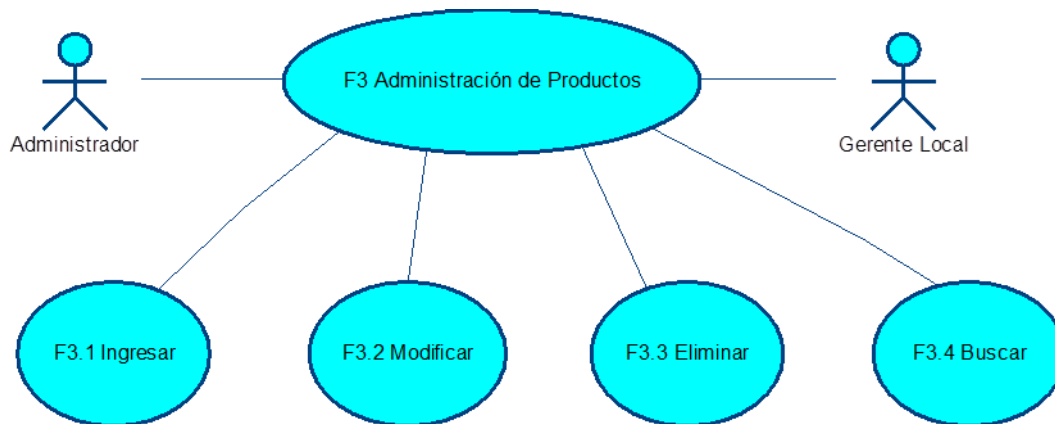
**Tabla 10:** Flujo Buscar Clientes

**Fuente:** Diego Tobar

#### 4.3.4. Administración de Productos

**Caso de Uso:** Administración de Productos

**Identificador:** F3



**Diagrama 13:** Administración de Productos

**Fuente:** Diego Tobar

**Descripción:** Permite el manejo de productos de la empresa.

#### 4.3.4.1. Ingresar Productos

Caso de Uso: Ingresar Productos

Identificador: F3.1



Diagrama 14: Ingresar Productos

Fuente: Diego Tobar

Descripción: Permite el ingreso de productos al sistema.

Flujo de Ingresar Productos				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de Ingreso	2	Presenta plantilla	
3	Ingresar datos	4	Permite ingreso	
5	Evento de ingreso de datos	6	Valida datos	EX1
		7	Ingresar datos	
Excepciones de Ingresar Productos				
ID	Descripción	Acción		
EX1	Datos incorrectos	Mensaje de error		

Tabla 11: Flujo Ingresar Productos

Fuente: Diego Tobar

#### 4.3.4.2. Modificar Productos

Caso de Uso: Modificar Productos

Identificador: F3.2

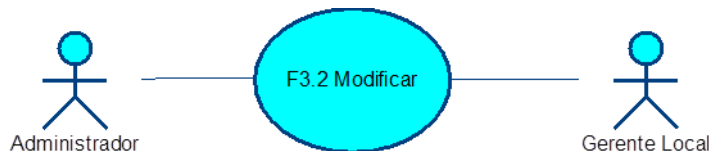


Diagrama 15: Modificar Productos

Fuente: Diego Tobar

**Descripción:** Permite la modificación de productos al sistema.

Flujo de Modificar Productos				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de modificación	2	Presenta plantilla	
3	Modifica datos	4	Permite modificar	
5	Evento de modificación de datos	6	Valida datos	EX1
		7	Modifica datos	
Excepciones de Modificar Productos				
ID	Descripción	Acción		
EX1	Datos incorrectos	Mensaje de error		

Tabla 12: Flujo Modificar Productos

Fuente: Diego Tobar

#### 4.3.4.3. Eliminar Productos

Caso de Uso: Eliminar Productos

Identificador: F3.3

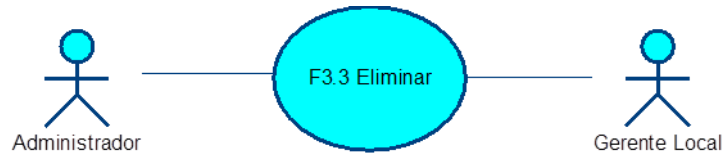


Diagrama 16: Eliminar Usuarios

Fuente: Diego Tobar

**Descripción:** Permite eliminar productos existentes del sistema.

Flujo de Eliminar Productos				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de eliminación	2	Presenta confirmación	EX1
3	Evento de eliminación de datos	4	Elimina datos	
Excepciones de Eliminar Productos				
ID	Descripción	Acción		
EX1	Datos incorrectos	Mensaje de error		

Tabla 13: Flujo Eliminar Usuarios

Fuente: Diego Tobar

#### 4.3.4.4. Buscar Productos

Caso de Uso: Buscar Productos

Identificador: F3.4

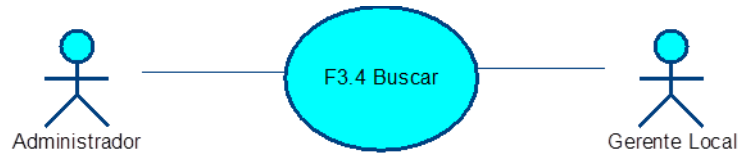


Diagrama 17: Buscar Productos

Fuente: Diego Tobar

**Descripción:** Permite buscar productos existentes del sistema.

Flujo de Buscar Productos				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de búsqueda	2	Presenta datos	EX1
Excepciones de Buscar Productos				
ID	Descripción	Acción		
EX1	Ausencia de Datos	Mensaje de no existencia		

Tabla 14: Flujo Buscar Productos

Fuente: Diego Tobar

### 4.3.5. Administración de Órdenes de Compra

Caso de Uso: Administración de Órdenes de Compra

Identificador: F4

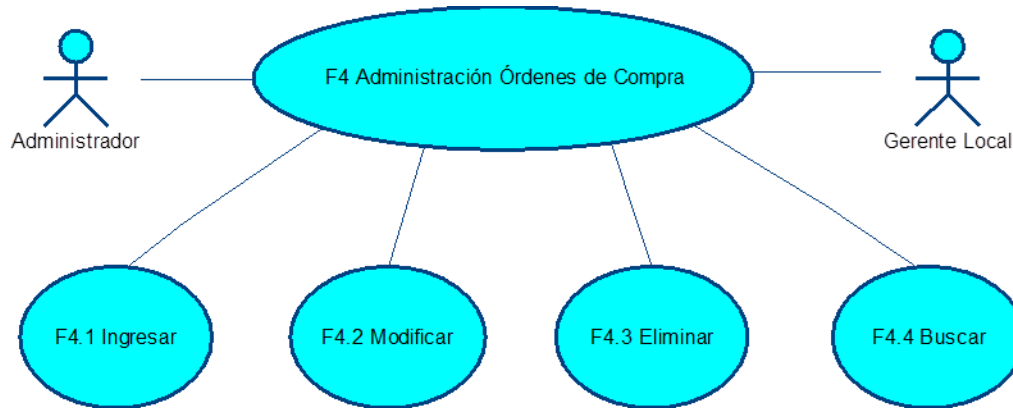


Diagrama 18: Administración de Órdenes de Compra

Fuente: Diego Tobar

Descripción: Permite el manejo de órdenes de compra de la empresa.

#### 4.3.5.1. Ingresar Órdenes de Compra

Caso de Uso: Ingresar Órdenes de Compra

Identificador: F4.1

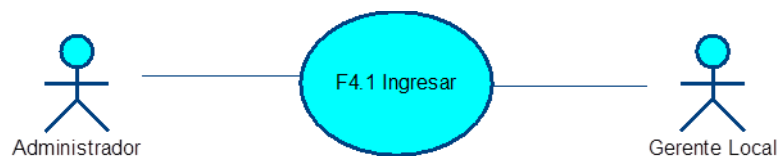


Diagrama 19: Ingresar Órdenes de Compra

Fuente: Diego Tobar

Descripción: Permite el ingreso de órdenes de compra al sistema.

Flujo de Ingresar Órdenes de Compra				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de Ingreso	2	Presenta plantilla	
3	Ingresar datos	4	Permite ingreso	

5	Evento de ingreso de datos	6	Valida datos	EX1
		7	Ingresa datos	
<b>Excepciones de Ingresar Órdenes de Compra</b>				
ID	Descripción	Acción		
EX1	Datos incorrectos	Mensaje de error		

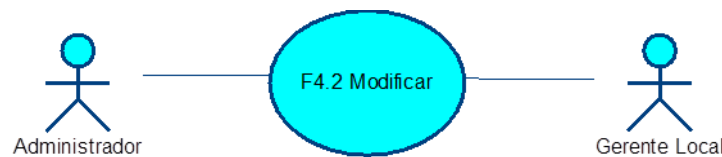
**Tabla 15:** Flujo Ingresar Órdenes de Compra

**Fuente:** Diego Tobar

#### 4.3.5.2. Modificar Órdenes de Compra

**Caso de Uso:** Modificar Órdenes de Compra

**Identificador:** F4.2



**Diagrama 20:** Modificar Órdenes de Compra

**Fuente:** Diego Tobar

**Descripción:** Permite la modificación de órdenes de compra al sistema.

<b>Flujo de Modificar Órdenes de Compra</b>				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de modificación	2	Presenta plantilla	
3	Modifica datos	4	Permite modificar	
5	Evento de modificación de datos	6	Valida datos	EX1

		7	Modifica datos	
Excepciones de Modificar Órdenes de Compra				
ID	Descripción	Acción		
EX1	Datos incorrectos	Mensaje de error		

Tabla 16: Flujo Modificar Órdenes de Compra

Fuente: Diego Tobar

### 4.3.5.3. Eliminar Órdenes de Compra

Caso de Uso: Eliminar Órdenes de Compra

Identificador: F4.3

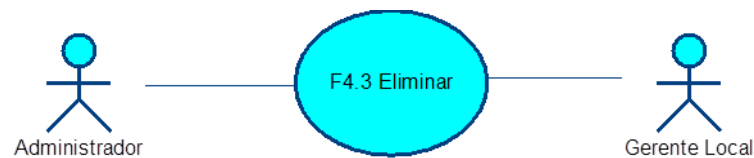


Diagrama 21: Eliminar Órdenes de Compra

Fuente: Diego Tobar

**Descripción:** Permite eliminar órdenes de compra existentes del sistema.

Flujo de Eliminar Órdenes de Compra				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de eliminación	2	Presenta confirmación	EX1
3	Evento de eliminación de datos	4	Elimina datos	
Excepciones de Eliminar Órdenes de Compra				
ID	Descripción	Acción		
EX1	Datos incorrectos	Mensaje de error		

Tabla 17: Flujo Eliminar Órdenes de Compra

Fuente: Diego Tobar

#### 4.3.5.4. Buscar Órdenes de Compra

Caso de Uso: Buscar Órdenes de Compra

Identificador: F4.4

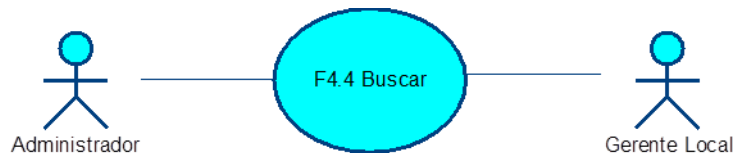


Diagrama 22: Buscar Órdenes de Compra

Fuente: Diego Tobar

**Descripción:** Permite buscar órdenes de compra existentes del sistema.

Flujo de Buscar Órdenes de Compra				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de búsqueda	2	Presenta datos	EX1
Excepciones de Buscar Órdenes de Compra				
ID	Descripción	Acción		
EX1	Ausencia de Datos	Mensaje de no existencia		

Tabla 18: Flujo Buscar Órdenes de Compra

Fuente: Diego Tobar

### 4.3.6. Administración de Proveedores

Caso de Uso: Administración de Proveedores

Identificador: F5

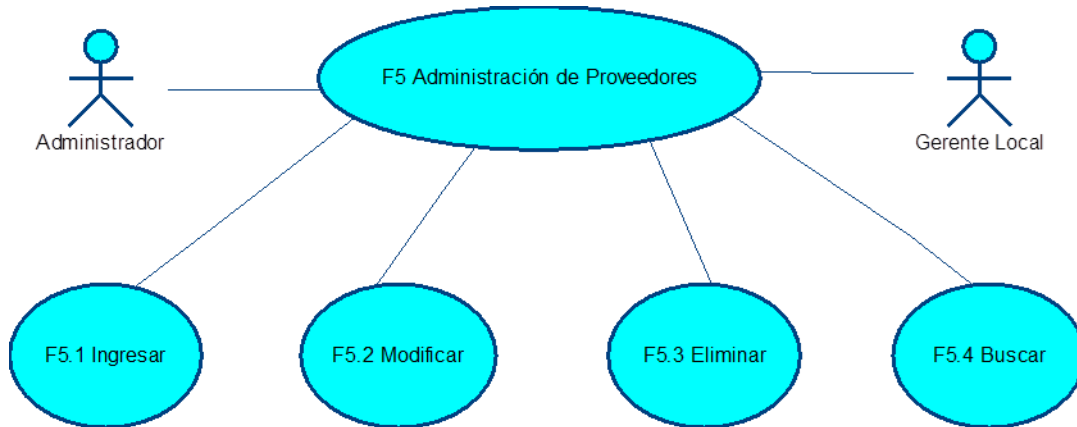


Diagrama 23: Administración de Proveedores

Fuente: Diego Tobar

**Descripción:** Permite el manejo de proveedores de la empresa.

#### 4.3.6.1. Ingresar Proveedores

Caso de Uso: Ingresar Proveedores

Identificador: F5.1



Diagrama 24: Ingresar Proveedores

Fuente: Diego Tobar

**Descripción:** Permite el ingreso de proveedores al sistema.

Flujo de Ingresar Proveedores				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de Ingreso	2	Presenta plantilla	
3	Ingresar datos	4	Permite ingreso	

5	Evento de ingreso de datos	6	Valida datos	EX1
		7	Ingresa datos	
<b>Excepciones de Ingresar Proveedores</b>				
ID	Descripción	Acción		
EX1	Datos incorrectos	Mensaje de error		

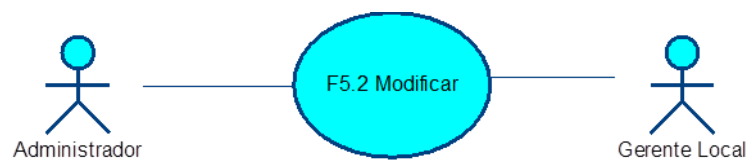
**Tabla 19:** Flujo Ingresar Proveedores

**Fuente:** Diego Tobar

#### 4.3.6.2. Modificar Proveedores

**Caso de Uso:** Modificar Proveedores

**Identificador:** F5.2



**Diagrama 25:** Modificar Proveedores

**Fuente:** Diego Tobar

**Descripción:** Permite la modificación de proveedores al sistema.

<b>Flujo de Modificar Proveedores</b>				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de modificación	2	Presenta plantilla	
3	Modifica datos	4	Permite modificar	
5	Evento de modificación de datos	6	Valida datos	EX1
		7	Modifica datos	
<b>Excepciones de Modificar Proveedores</b>				

ID	Descripción	Acción
EX1	Datos incorrectos	Mensaje de error

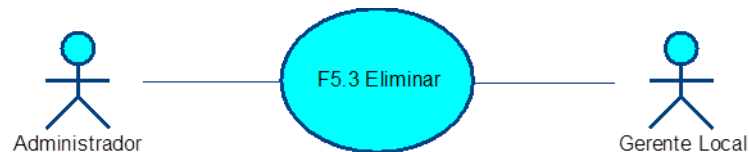
**Tabla 20:** Flujo Modificar Proveedores

Fuente: Diego Tobar

### 4.3.6.3. Eliminar Proveedores

**Caso de Uso:** Eliminar Proveedores

**Identificador:** F5.3



**Diagrama 26:** Eliminar Proveedores

Fuente: Diego Tobar

**Descripción:** Permite eliminar proveedores existentes del sistema.

Flujo de Eliminar Proveedores				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de eliminación	2	Presenta confirmación	EX1
3	Evento de eliminación de datos	4	Elimina datos	
Excepciones de Eliminar Proveedores				
ID	Descripción	Acción		
EX1	Datos incorrectos	Mensaje de error		

**Tabla 21:** Flujo Eliminar Proveedores

Fuente: Diego Tobar

#### 4.3.6.4. Buscar Proveedores

Caso de Uso: Buscar Proveedores

Identificador: F5.4



Diagrama 27: Buscar Proveedores

Fuente: Diego Tobar

**Descripción:** Permite buscar proveedores existentes del sistema.

Flujo de Buscar Proveedores				
ID	Actor	ID	Sistema	Excepción
1	Genera un evento de búsqueda	2	Presenta datos	EX1
Excepciones de Buscar Proveedores				
ID	Descripción	Acción		
EX1	Ausencia de Datos	Mensaje de no existencia		

Tabla 22: Flujo Buscar Proveedores

Fuente: Diego Tobar

#### 4.4. Diagrama de Clases

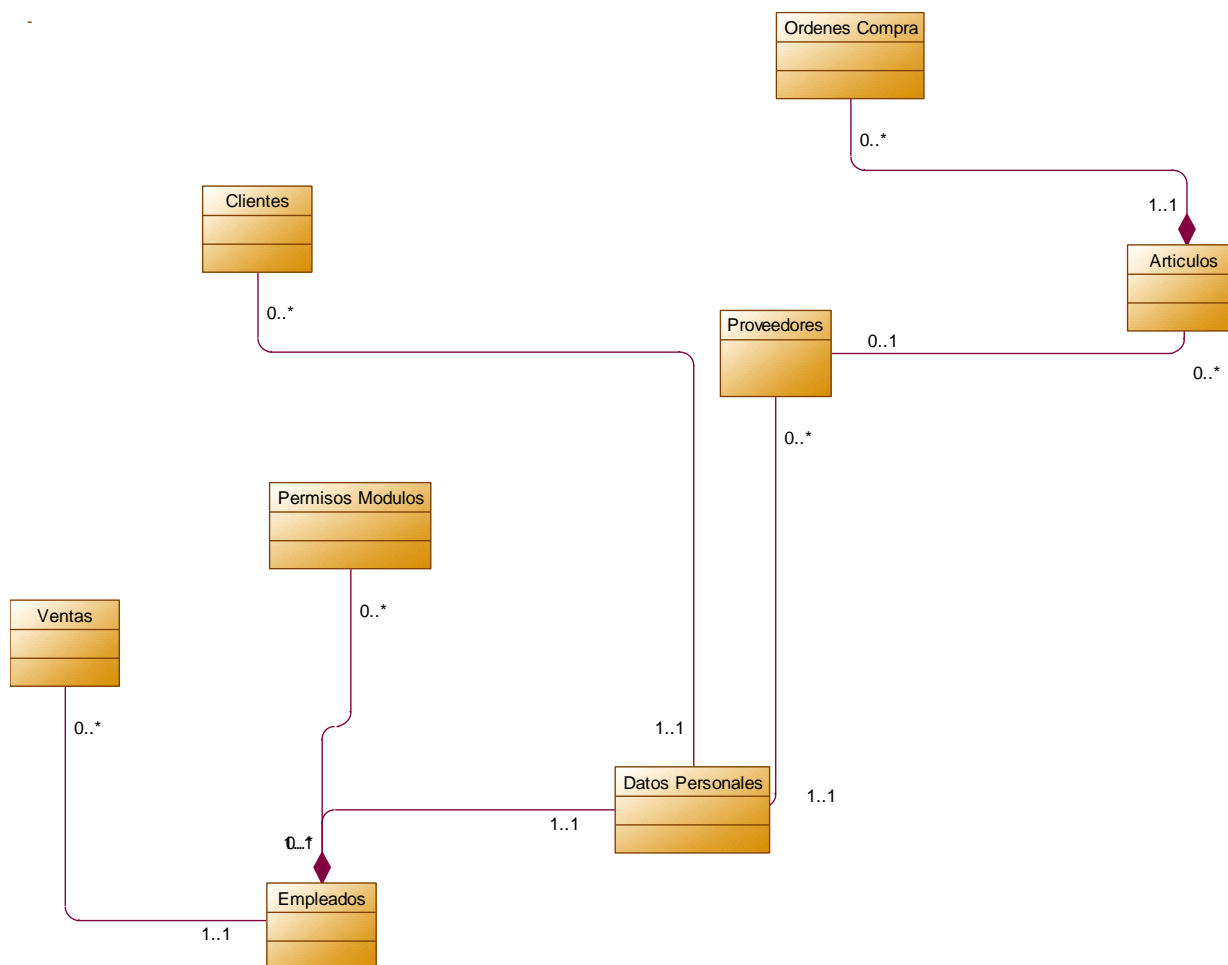


Diagrama 28: Diagrama de Clases

Fuente: Diego Tobar

## 4.5. Diagrama de Navegación

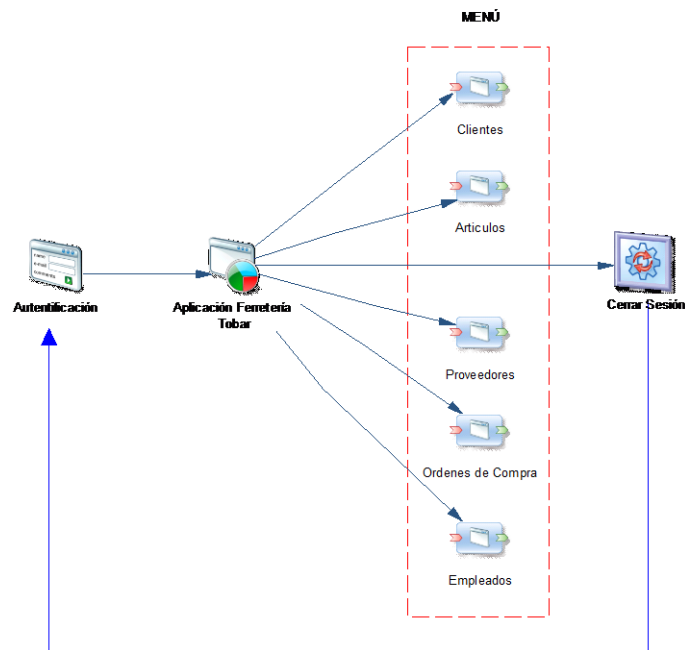


Diagrama 29: Diagrama de Navegación

Fuente: Diego Tobar

## 4.6. Diagrama de Interfaz

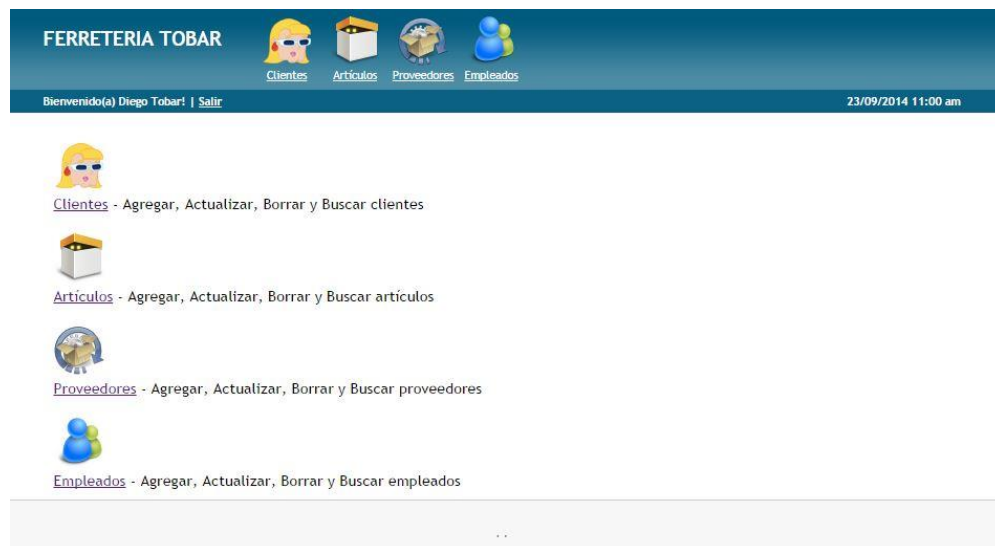


Diagrama 30: Diagrama de Interfaz

Fuente: Diego Tobar

## **4.7. Diseño del Software**

### **4.7.1. Patrones de Diseño Web**

Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces.

Un patrón de diseño resulta ser una solución a un problema de diseño. Para que una solución sea considerada un patrón debe poseer ciertas características. Una de ellas es que debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores. Otra es que debe ser reutilizable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias.

El creciente auge en el desarrollo de aplicaciones Web que se vive actualmente nos orienta a buscar un Patrón de Desarrollo altamente extensible, que permita futuras modificaciones posteriores, cambios o adaptaciones con el menor esfuerzo posible; esto nos lleva a plantear dentro del presente trabajo uno de los más utilizados como lo es el modelo MVC, que sirve de base para muchas aplicaciones disponibles hoy en día; y que se complementa y adapta fácilmente con Codeigniter.

### **4.7.2. Diseño Arquitectónico**

#### **4.7.2.1. Patrón Modelo –Vista- Controlador (MVC)**

El patrón de Arquitectura Modelo-Vista-Controlador. MVC es un enfoque de software que separa la lógica de la aplicación de la presentación. En la práctica, permite a las páginas web contener el mínimo código ya que la presentación está separada del código PHP y de la recuperación e inserción de información en la Base de Datos (Wikipedia, 2013).

Se originó en la comunidad Smalltalk para implementar interfaces de usuario en los que las responsabilidades están bien distribuidas entre distintas partes del diseño (componentes); dividiendo la aplicación de acuerdo a las responsabilidades dentro de la misma; entre la lógica de negocio, la gestión de eventos y la presentación.

- **Modelo**

El Modelo representa las estructuras de datos; es decir toda la información con la que opera la aplicación; Gestionar el comportamiento y los datos del dominio; típicamente las clases del modelo contendrán funciones que ayudan a devolver, insertar y actualizar información de su base de datos, independientemente de la Base de Datos a utilizar o posibles cambios.

Encapsular el modelo de una aplicación en componentes facilita la depuración, mejora la calidad y favorece la reutilización de código, y los cambios que el sistema pudiera sufrir a futuro.

- **Vista**

La Vista es la información que se presenta al usuario. La vista gestiona la presentación de la información de la aplicación; todo lo relativo a la interfaz de usuario, los datos necesarios para que el usuario pueda seguir interactuando con la aplicación. Una vista será normalmente una página web, la vista representa la lógica de presentación de la aplicación. Los componentes de la vista extraen el estado actual del sistema del modelo y proporcionan la interfaz de usuario para el protocolo que se está usando.

Como parte de la generación la vista debe presentar al usuario el conjunto de eventos que puede generar en ese momento en CodeIgniter, una vista también puede ser un fragmento de página como el encabezado o pie de página.

- **Controlador**

Es el indicado a responder a los eventos invocados desde la vista; y a su vez llamar a los métodos correspondientes del modelo para procesar y producir las respuestas necesarias con los datos adecuados para la interacción de la aplicación. El Controlador sirve como un intermediario entre el Modelo, la Vista y cualquier otro recurso necesario para procesar la solicitud HTTP y generar una página web.

### 4.7.3. Diagramas de Secuencia

#### 4.7.3.1. Ingresar Clientes

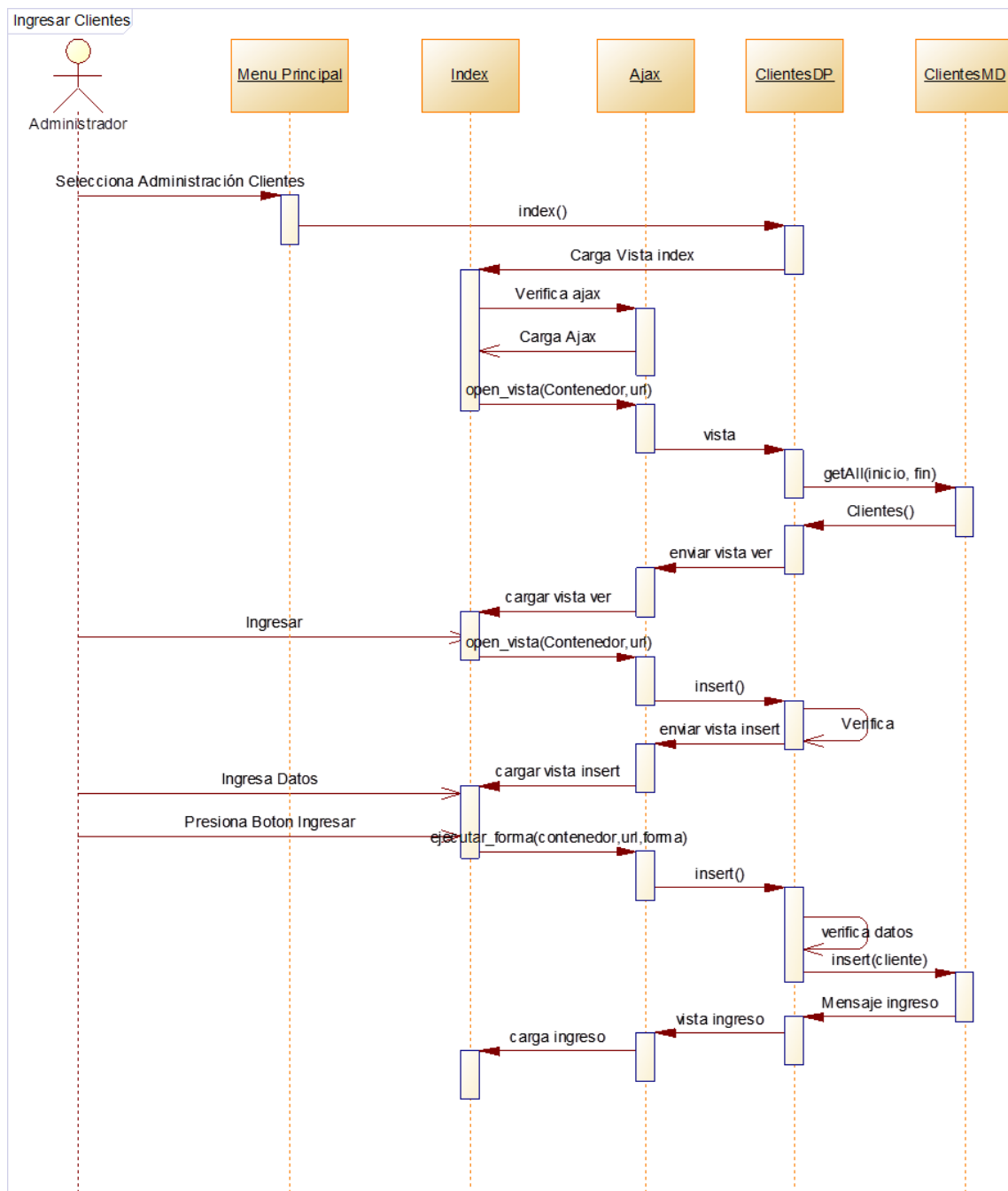


Diagrama 31: Diagrama Secuencia Ingresar Clientes

Fuente: Diego Tobar

### 4.7.3.2. Modificar Clientes

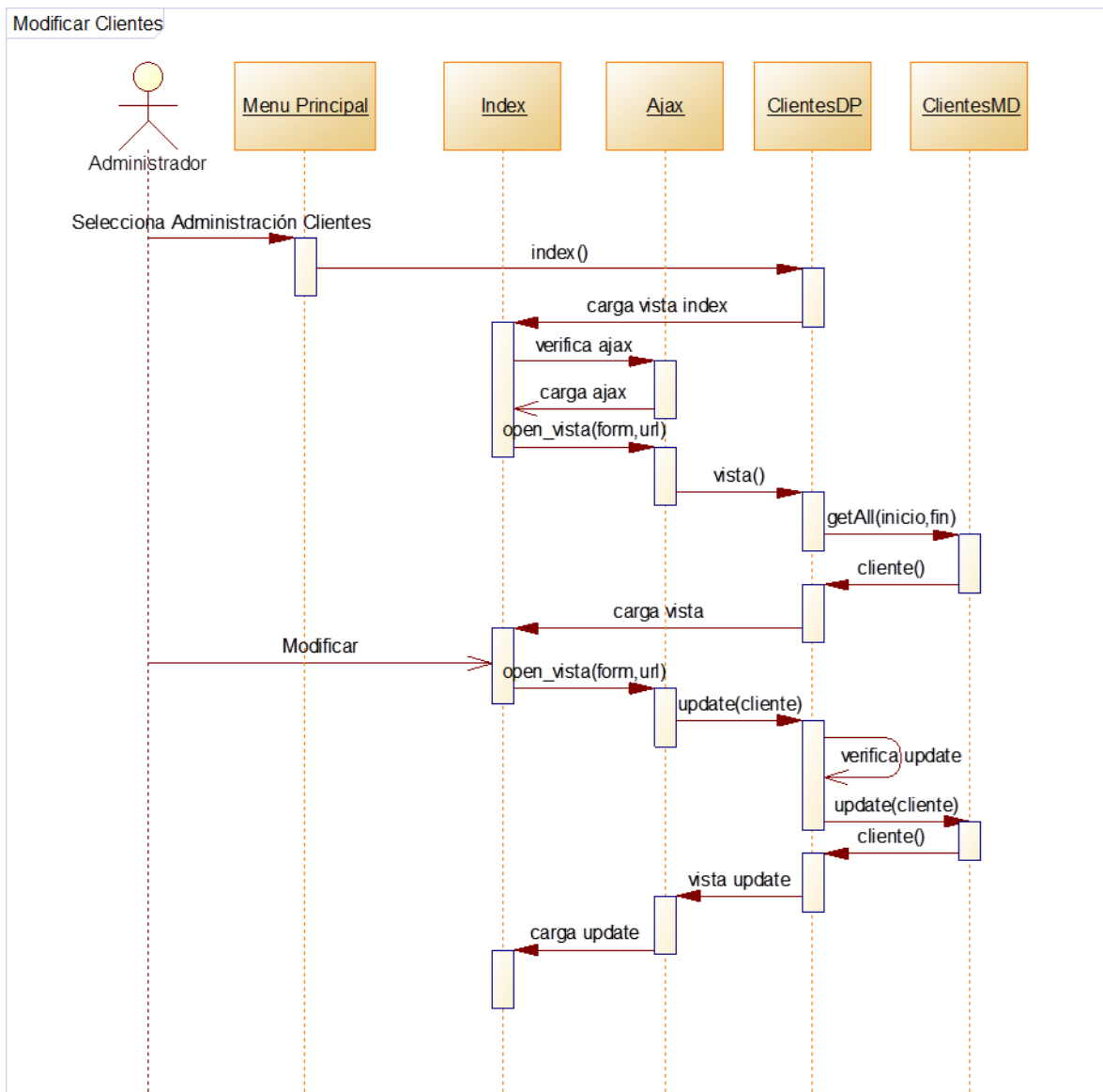


Diagrama 32: Diagrama Secuencia Modificar Clientes

Fuente: Diego Tobar

### 4.7.3.3. Eliminar Clientes

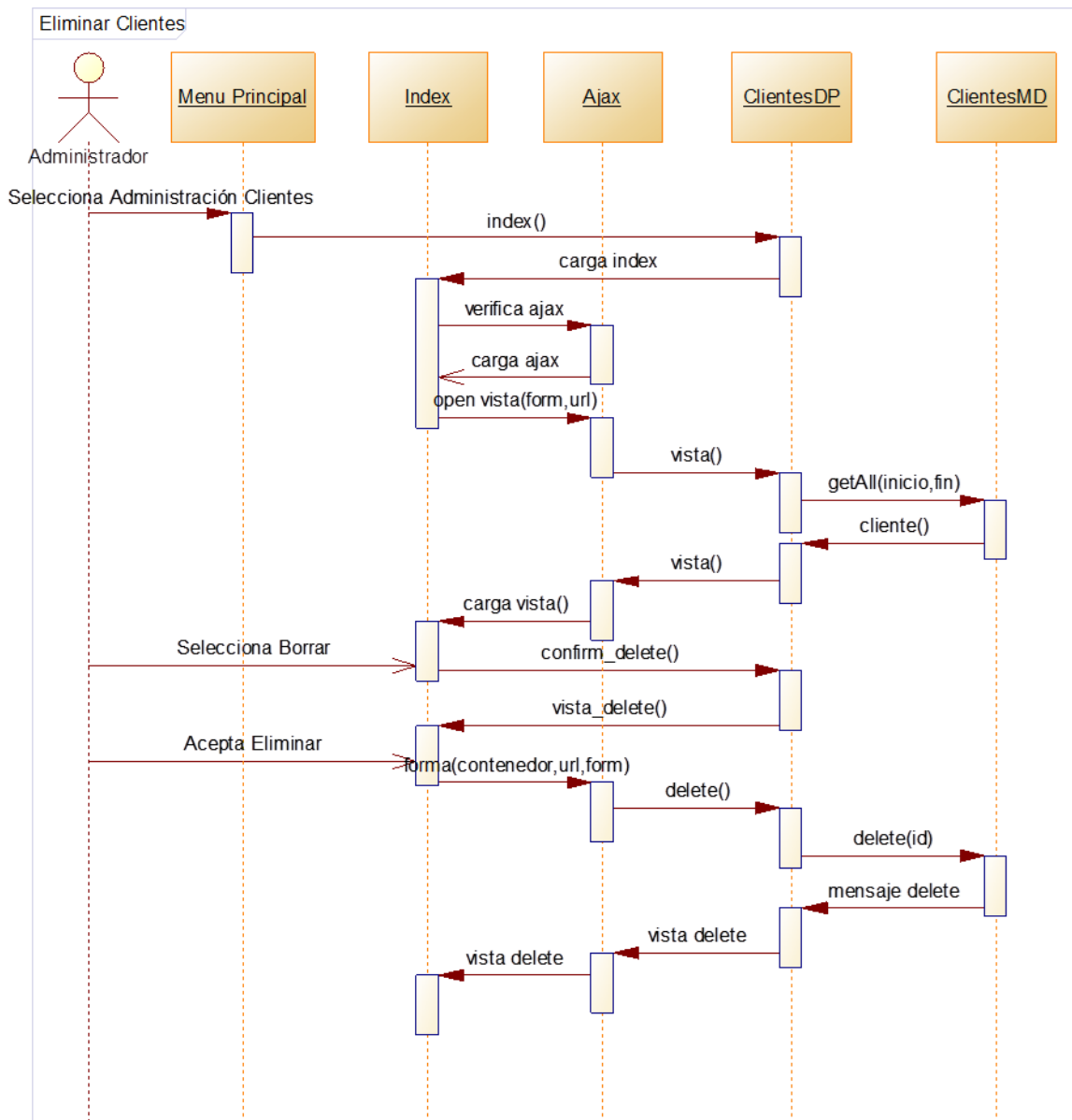


Diagrama 33: Diagrama Secuencia Modificar Clientes

Fuente: Diego Tobar

#### 4.7.3.4. Buscar Clientes

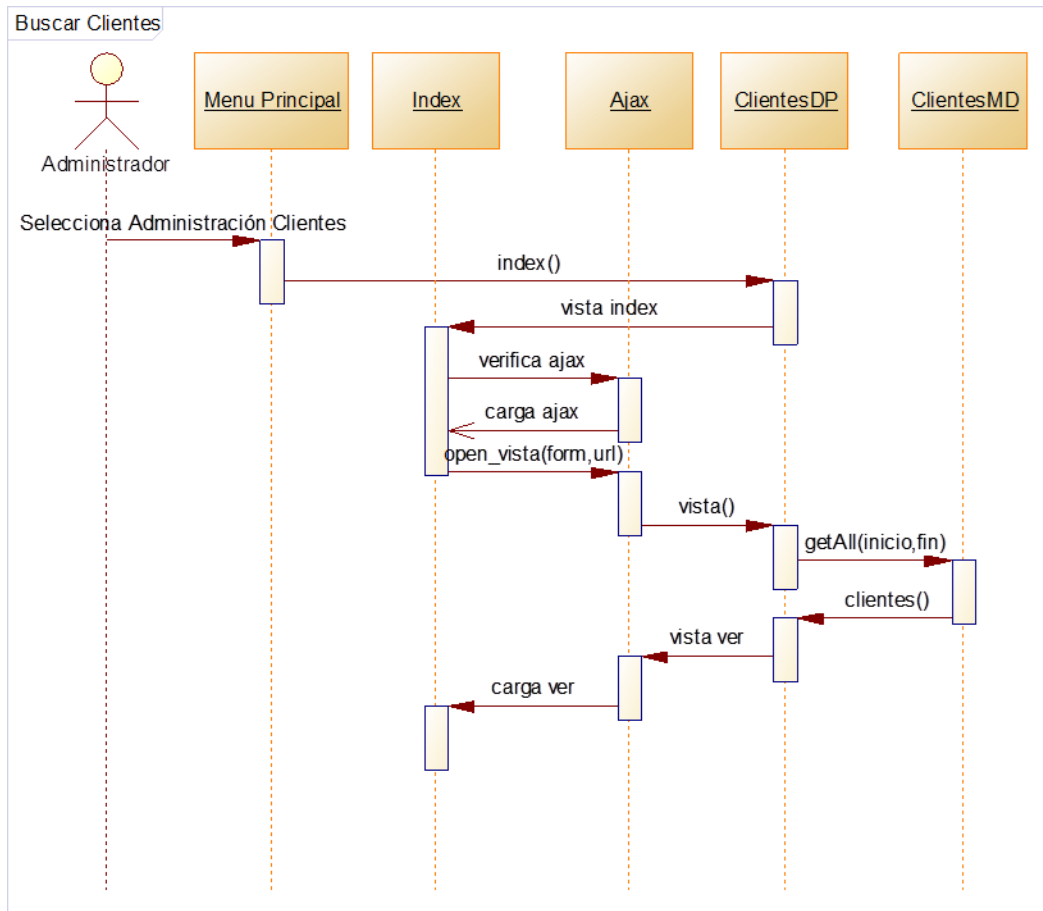
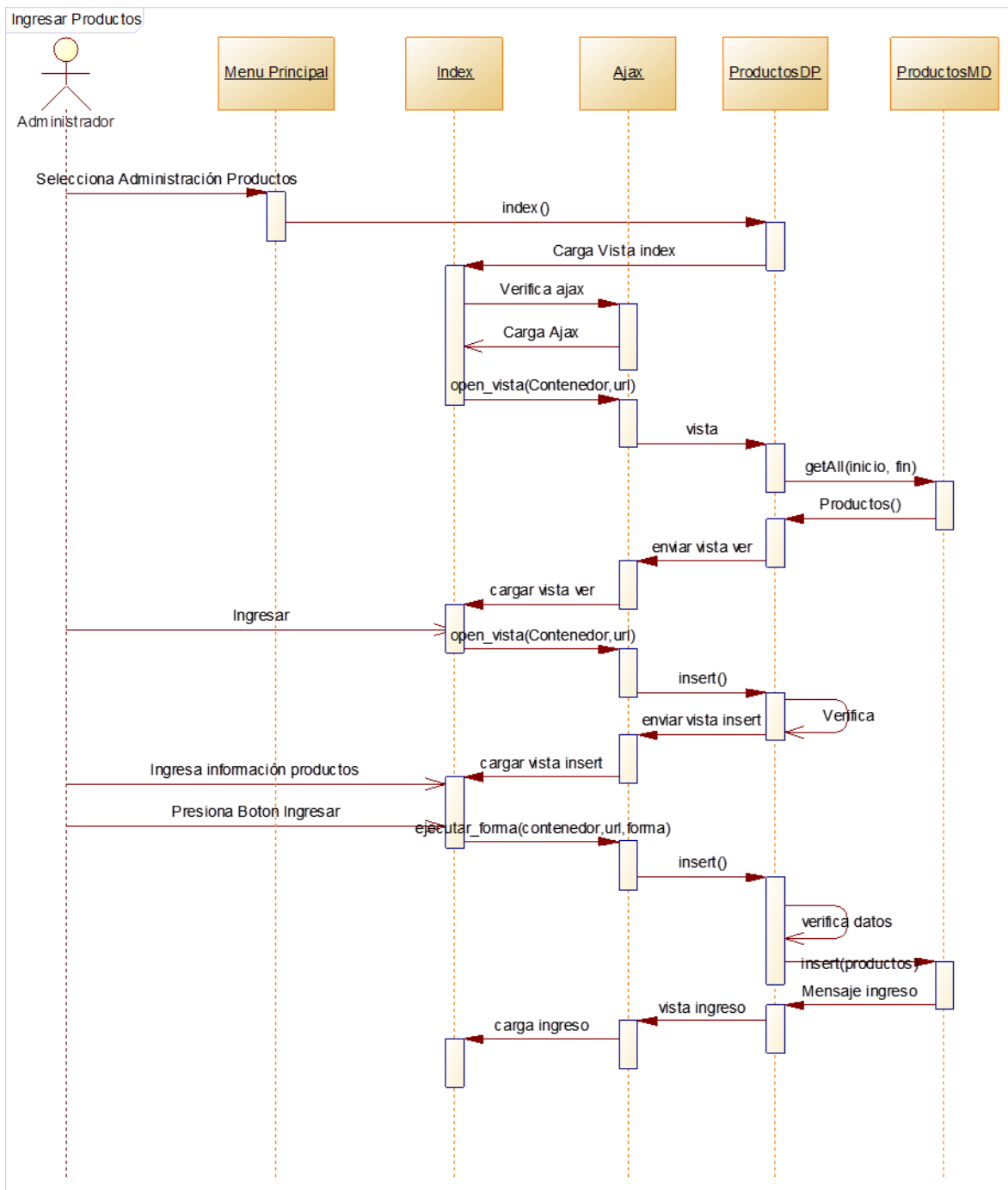


Diagrama 34: Diagrama Secuencia Buscar Clientes

Fuente: Diego Tobar

### 4.7.3.5. Ingresar Productos



**Diagrama 35:** Diagrama Secuencia Ingresar Productos

**Fuente:** Diego Tobar

### 4.7.3.6. Modificar Productos

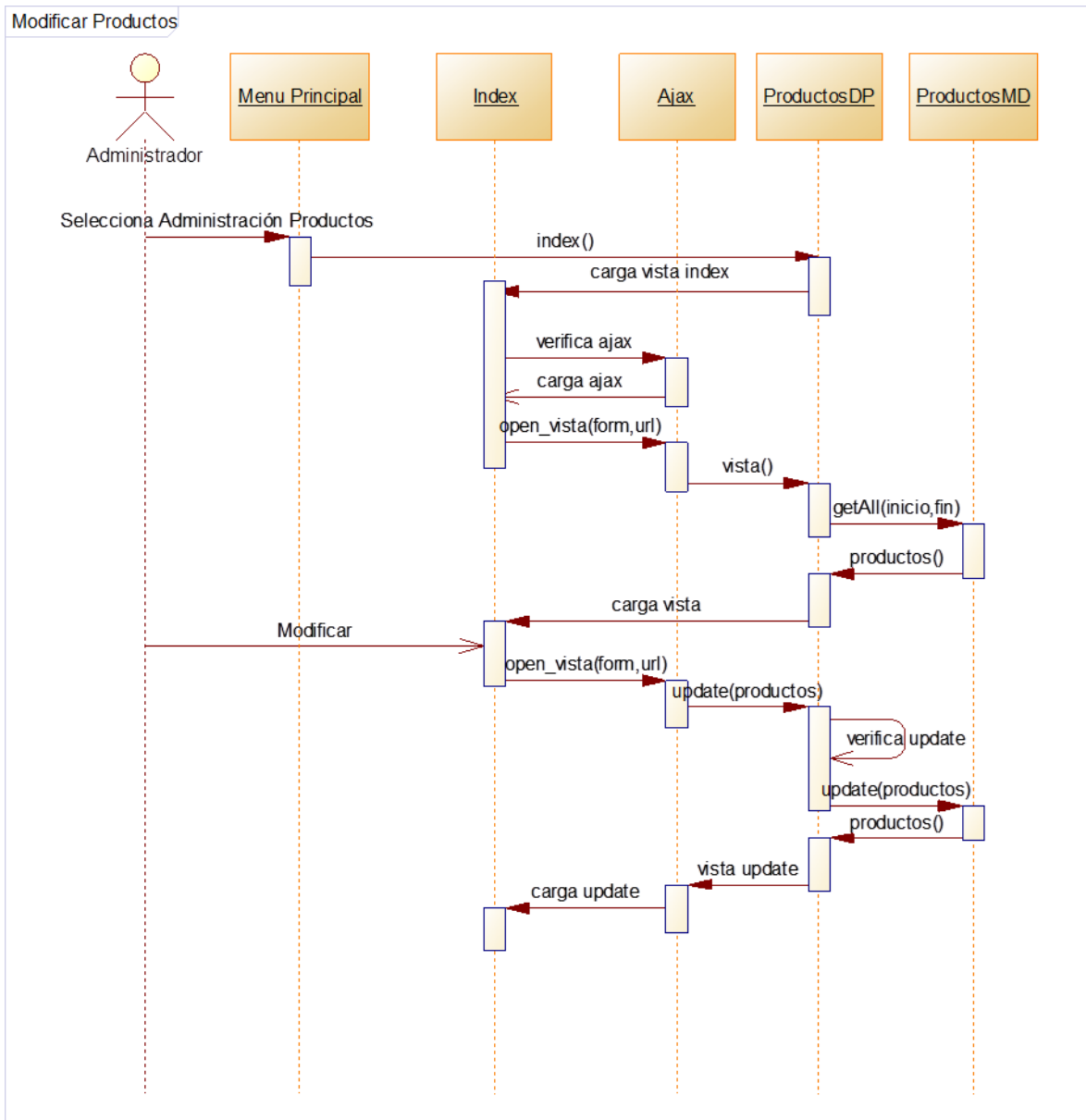


Diagrama 36: Diagrama Secuencia Modificar Productos

Fuente: Diego Tobar

### 4.7.3.7. Eliminar Productos

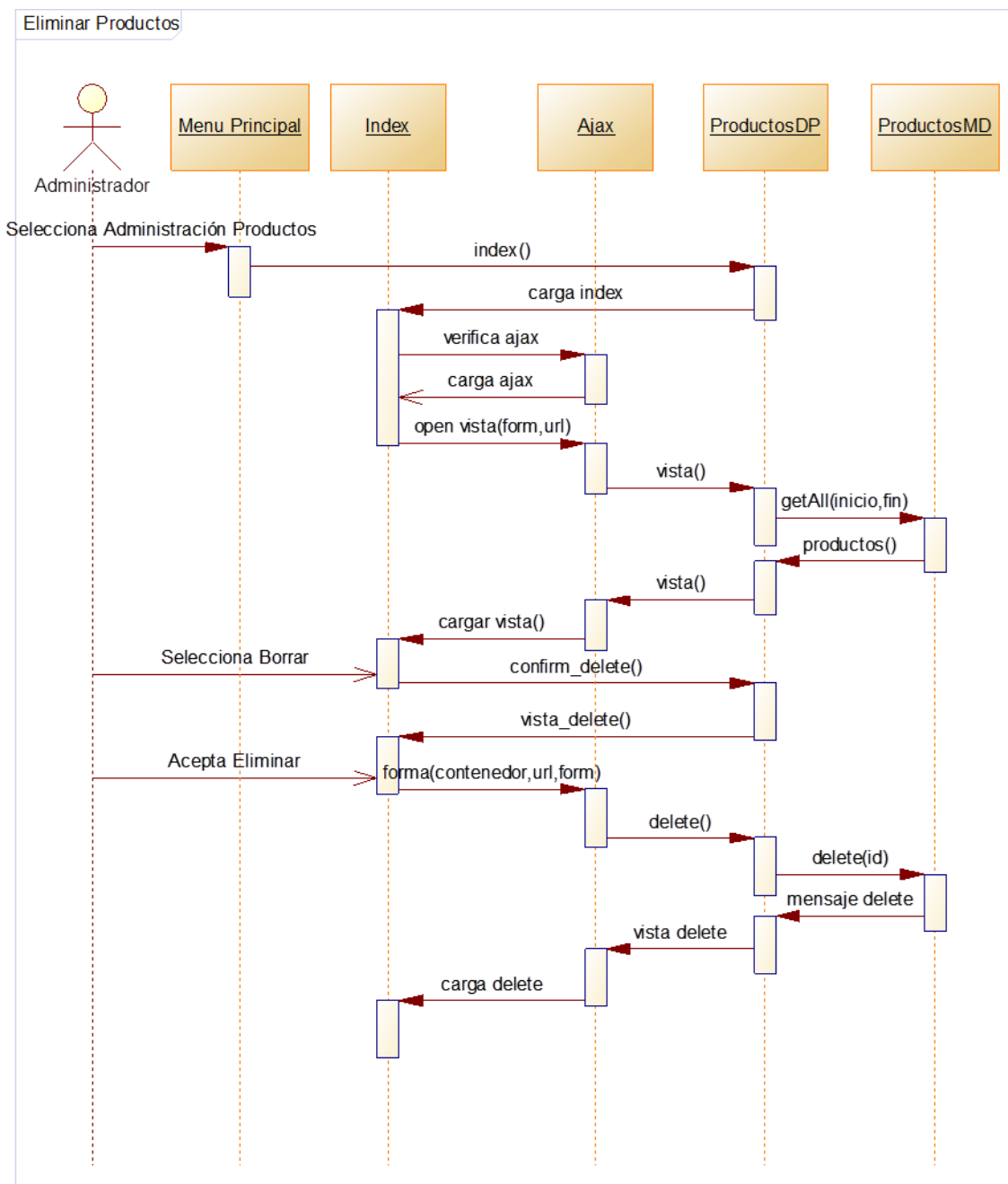


Diagrama 37: Diagrama Secuencia Eliminar Productos

Fuente: Diego Tobar

#### 4.7.3.8. Buscar Productos

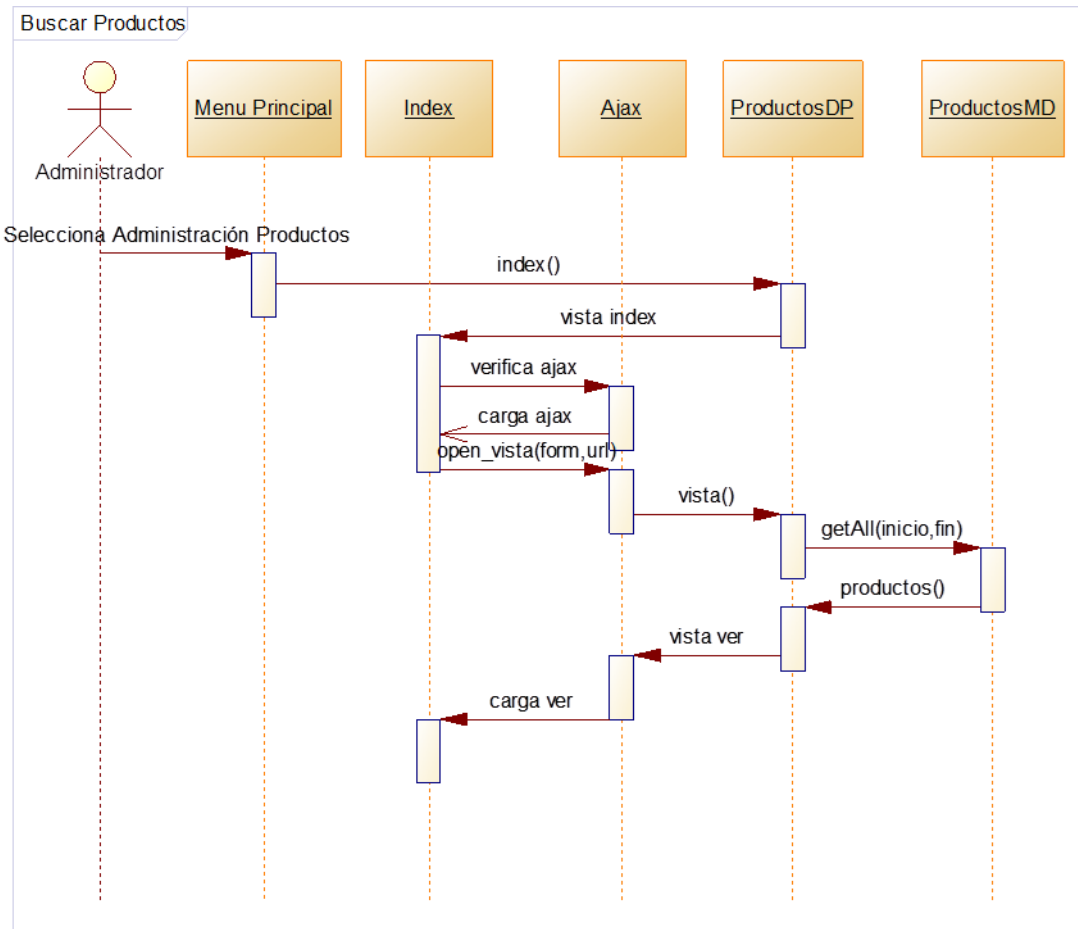


Diagrama 38: Diagrama Secuencia Buscar Productos

Fuente: Diego Tobar

### 4.7.3.9. Ingresar Usuarios

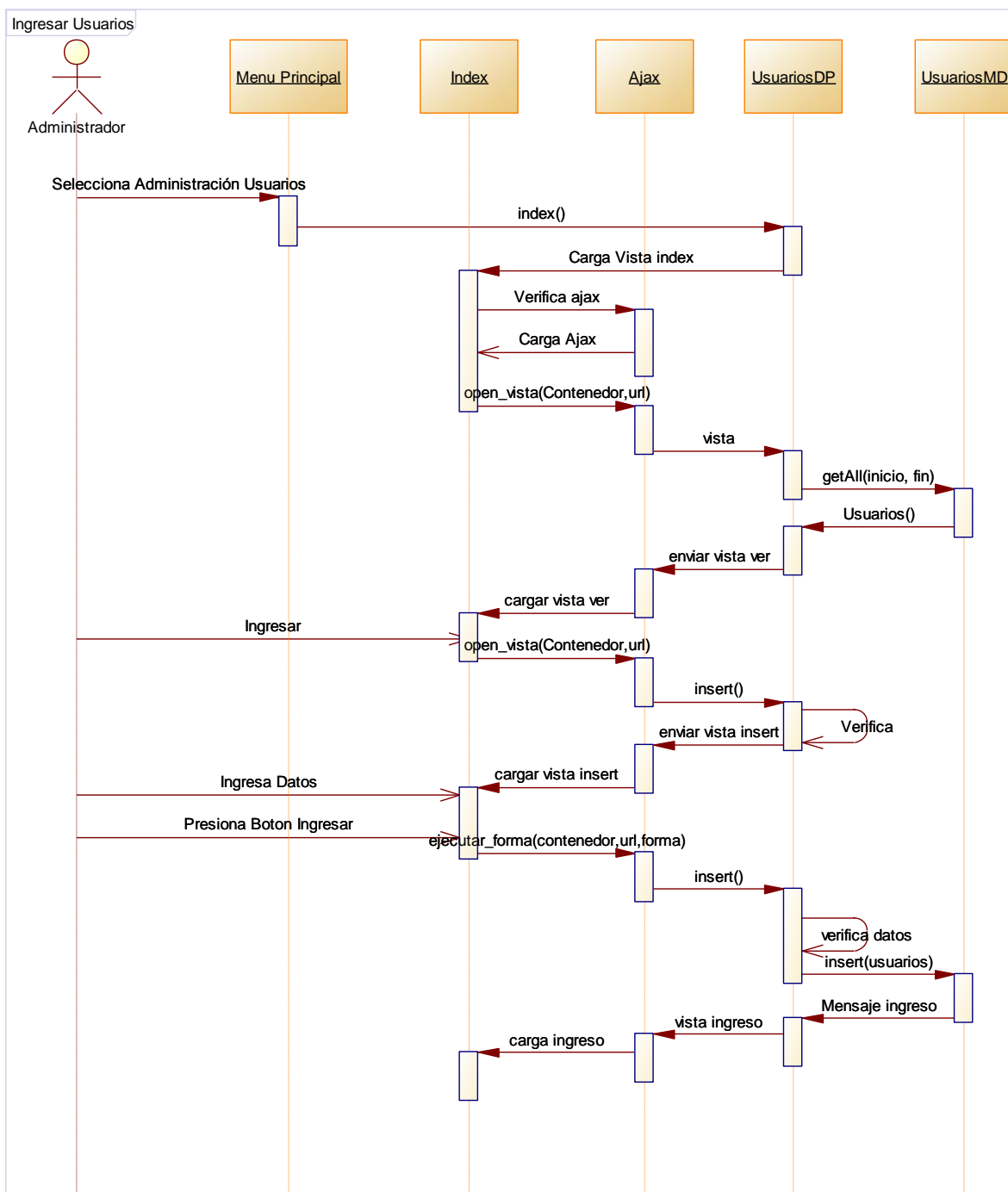


Diagrama 39: Diagrama Secuencia Ingresar Usuarios

Fuente: Diego Tobar

#### 4.7.3.10. Modificar Usuarios

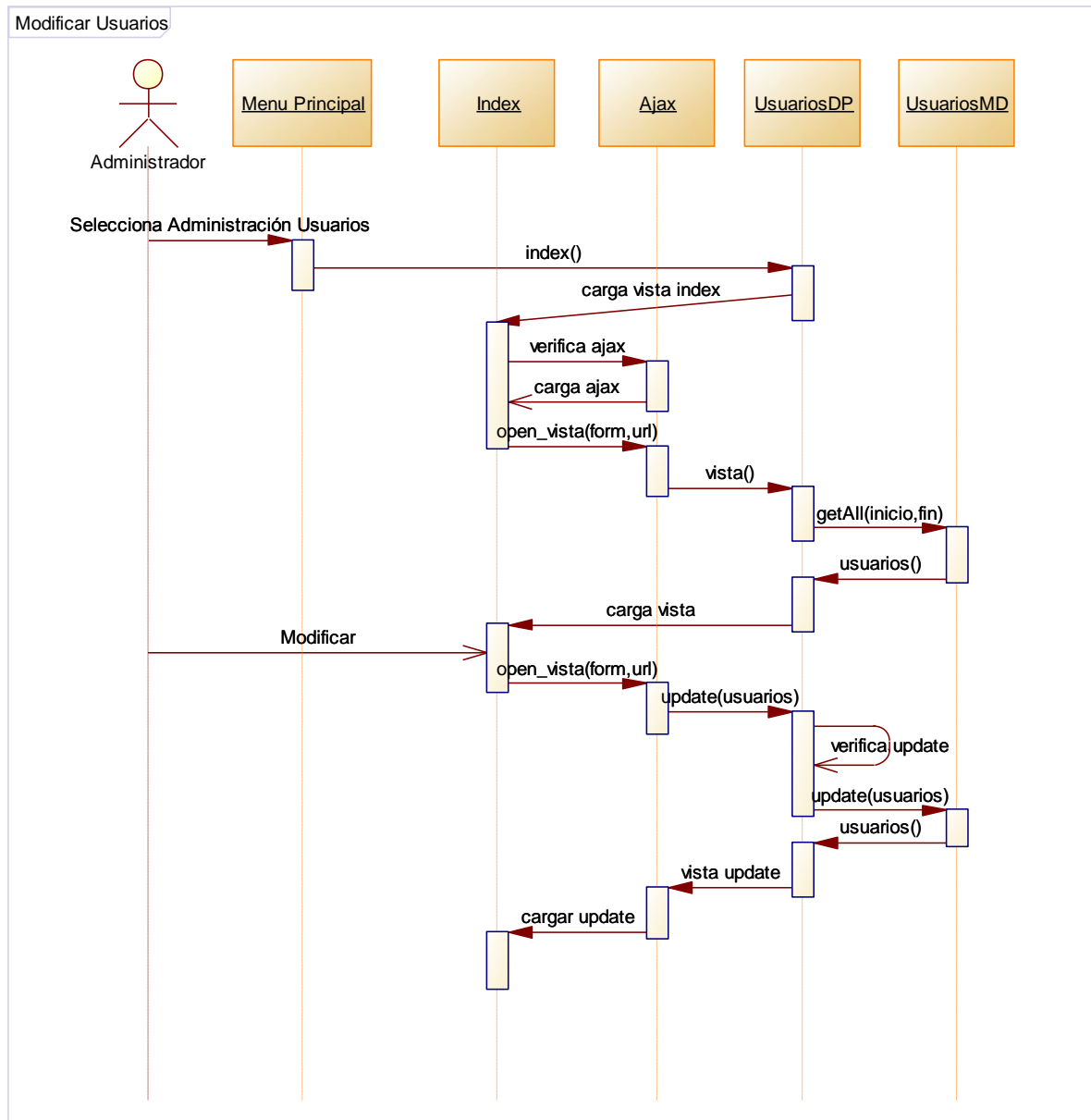


Diagrama 40: Diagrama Secuencia Modificar Usuarios

Fuente: Diego Tobar

### 4.7.3.11. Eliminar Usuarios

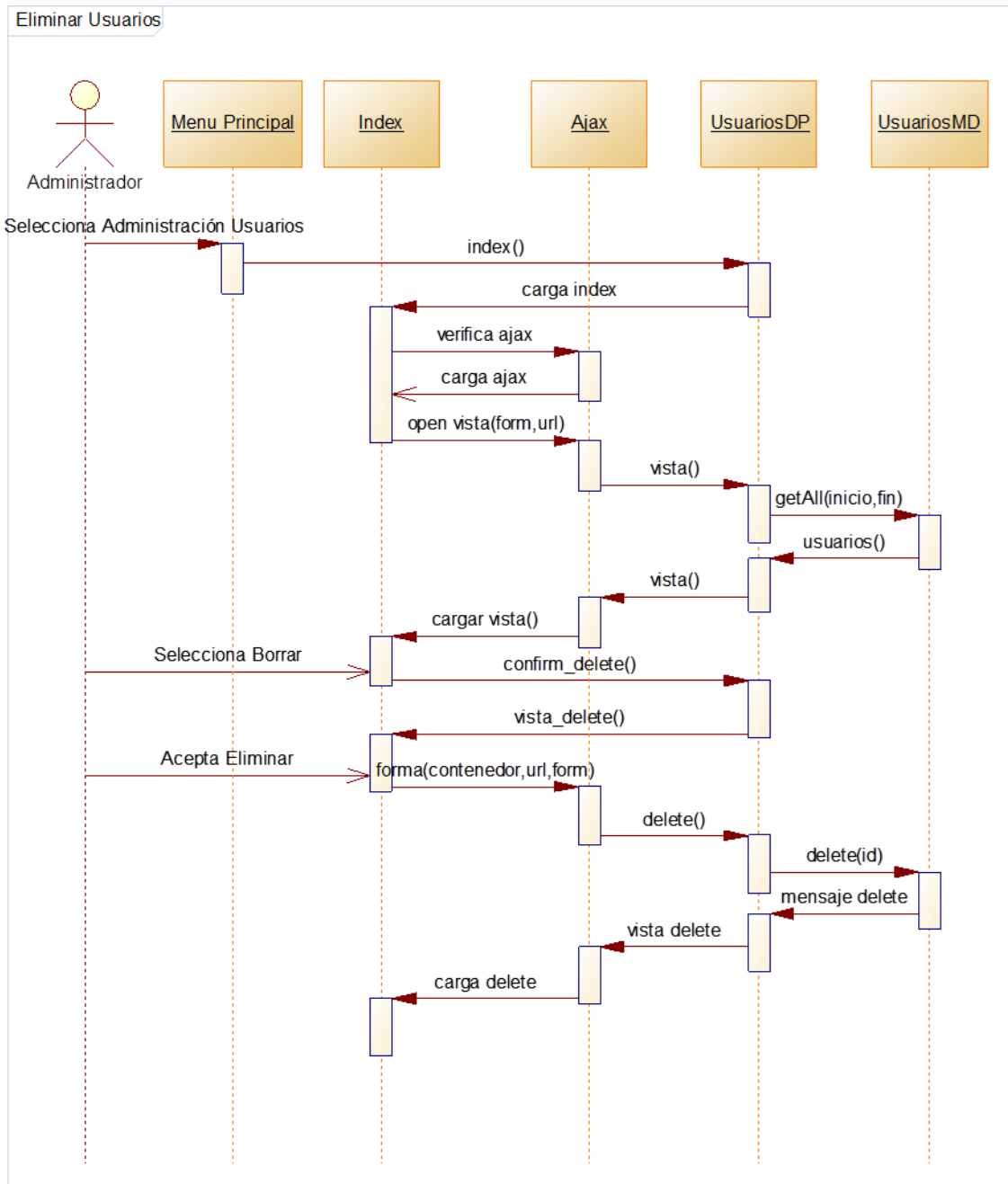


Diagrama 41: Diagrama Secuencia Eliminar Usuarios

Fuente: Diego Tobar

#### 4.7.3.12. Buscar Usuarios

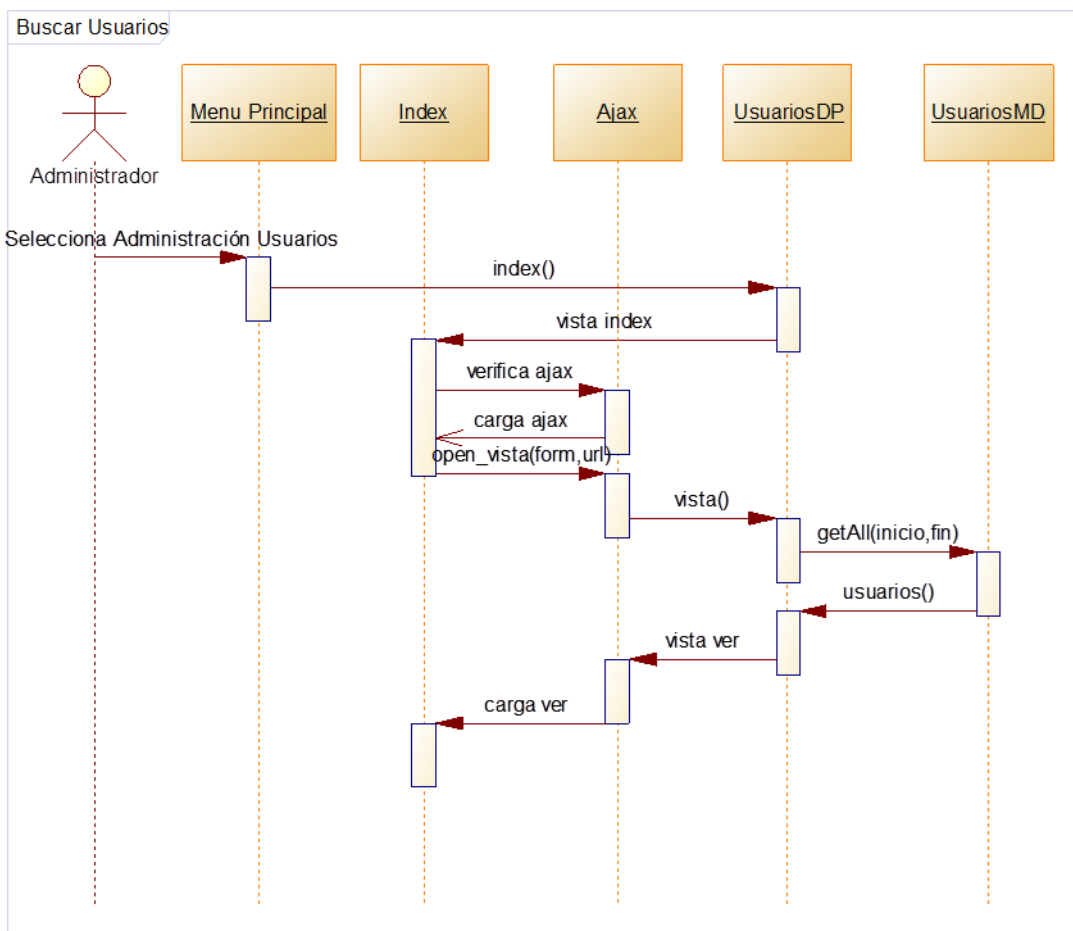


Diagrama 42: Diagrama Secuencia Buscar Usuarios

Fuente: Diego Tobar

### 4.7.3.13. Ingresar Órdenes de Compra

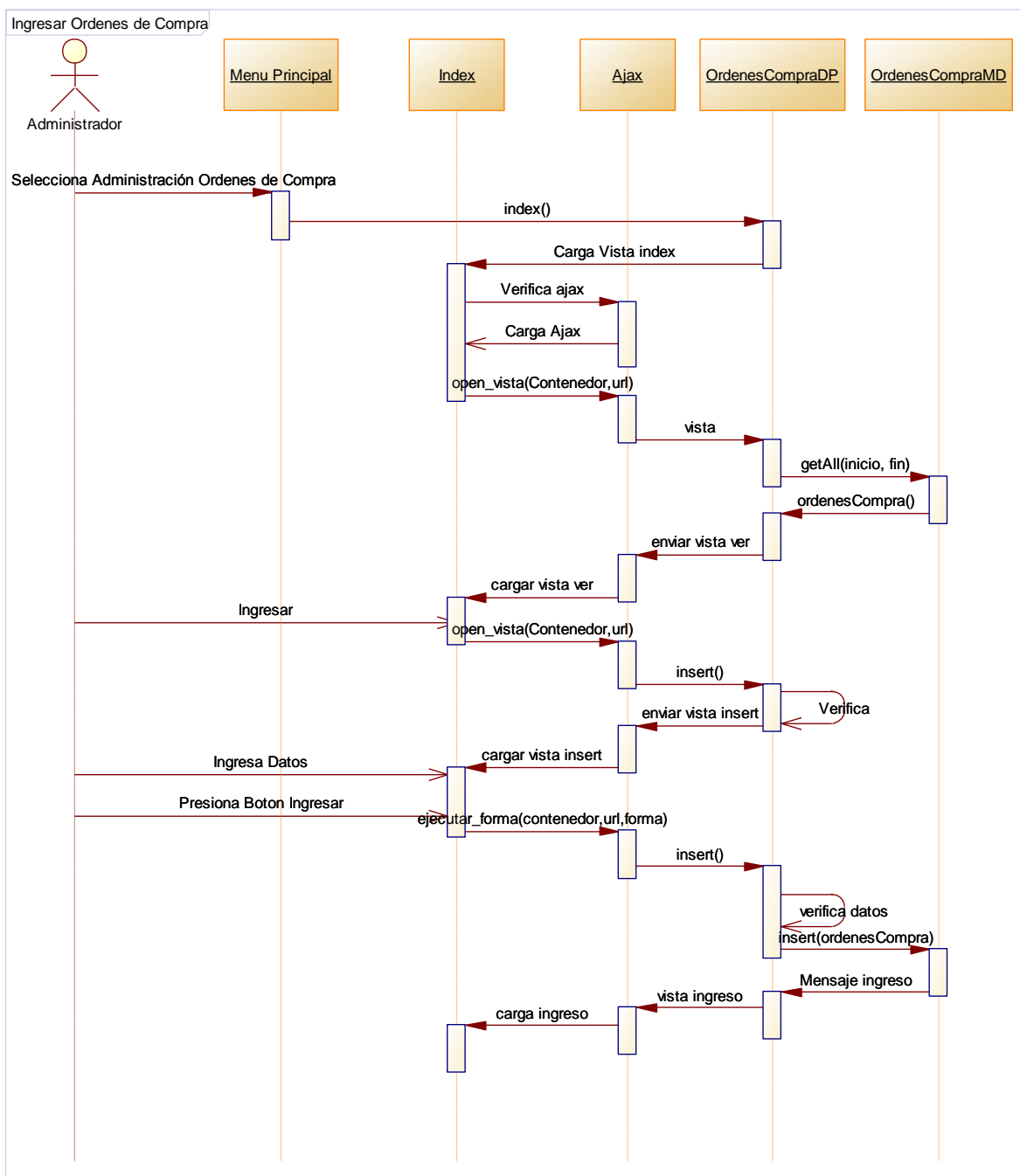


Diagrama 43: Diagrama Secuencia Ingresar Órdenes de Compra

Fuente: Diego Tobar

#### 4.7.3.14. Modificar Órdenes de Compra

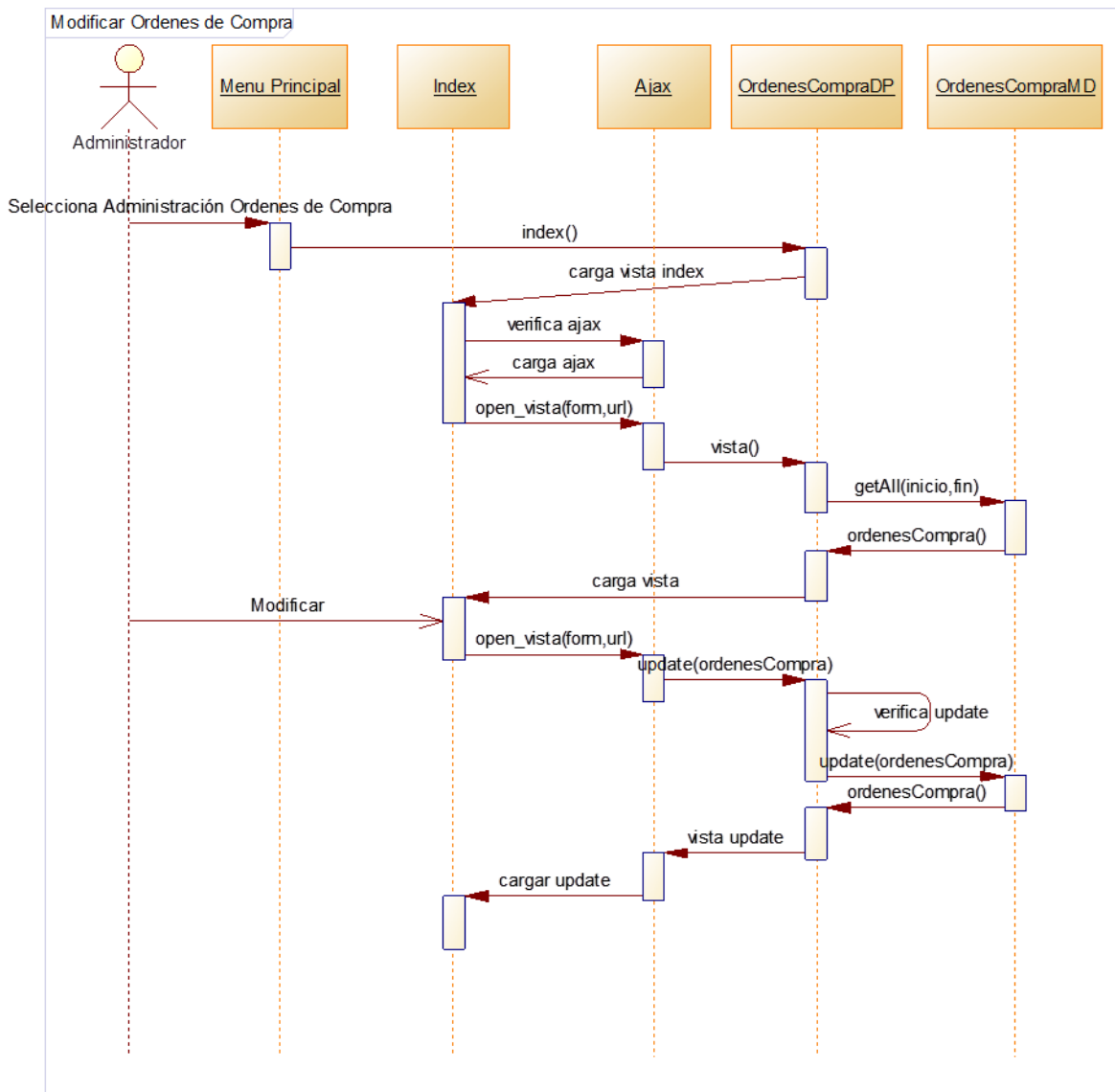


Diagrama 44: Diagrama Secuencia Modificar Órdenes de Compra

Fuente: Diego Tobar

### 4.7.3.15. Eliminar Órdenes de Compra

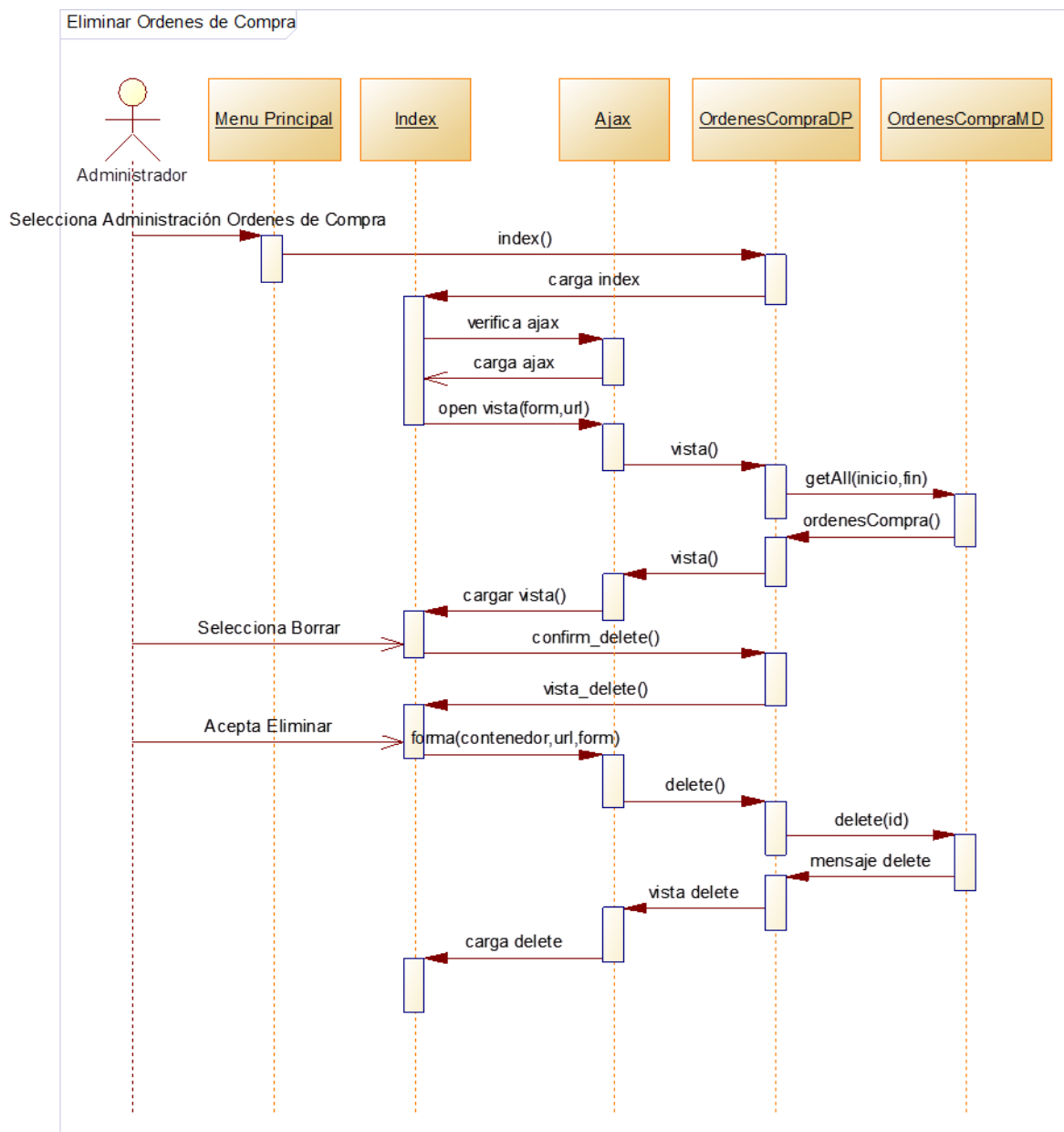


Diagrama 45: Diagrama Secuencia Eliminar Órdenes de Compra

Fuente: Diego Tobar

#### 4.7.3.16. Buscar Órdenes de Compra

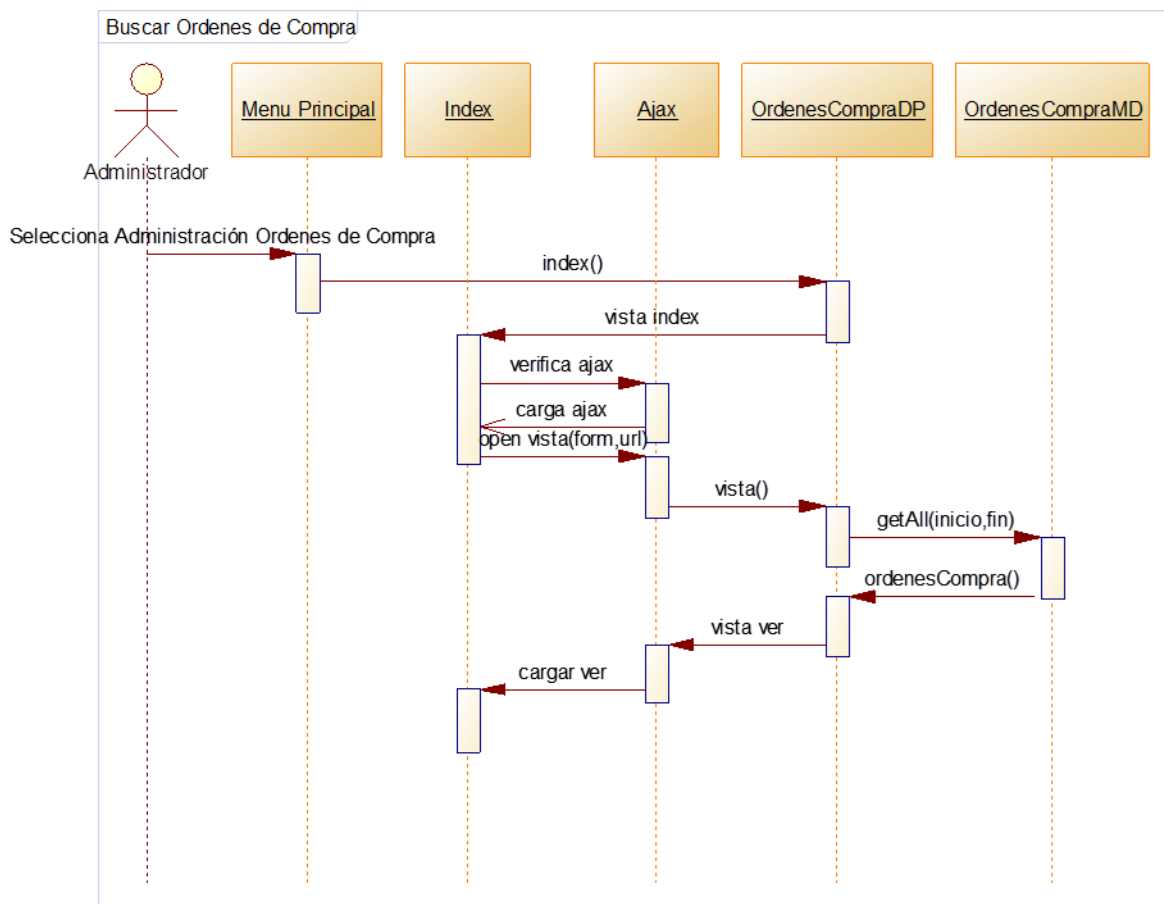


Diagrama 46: Diagrama Secuencia Buscar Órdenes de Compra

Fuente: Diego Tobar

### 4.7.3.17. Ingresar Proveedores

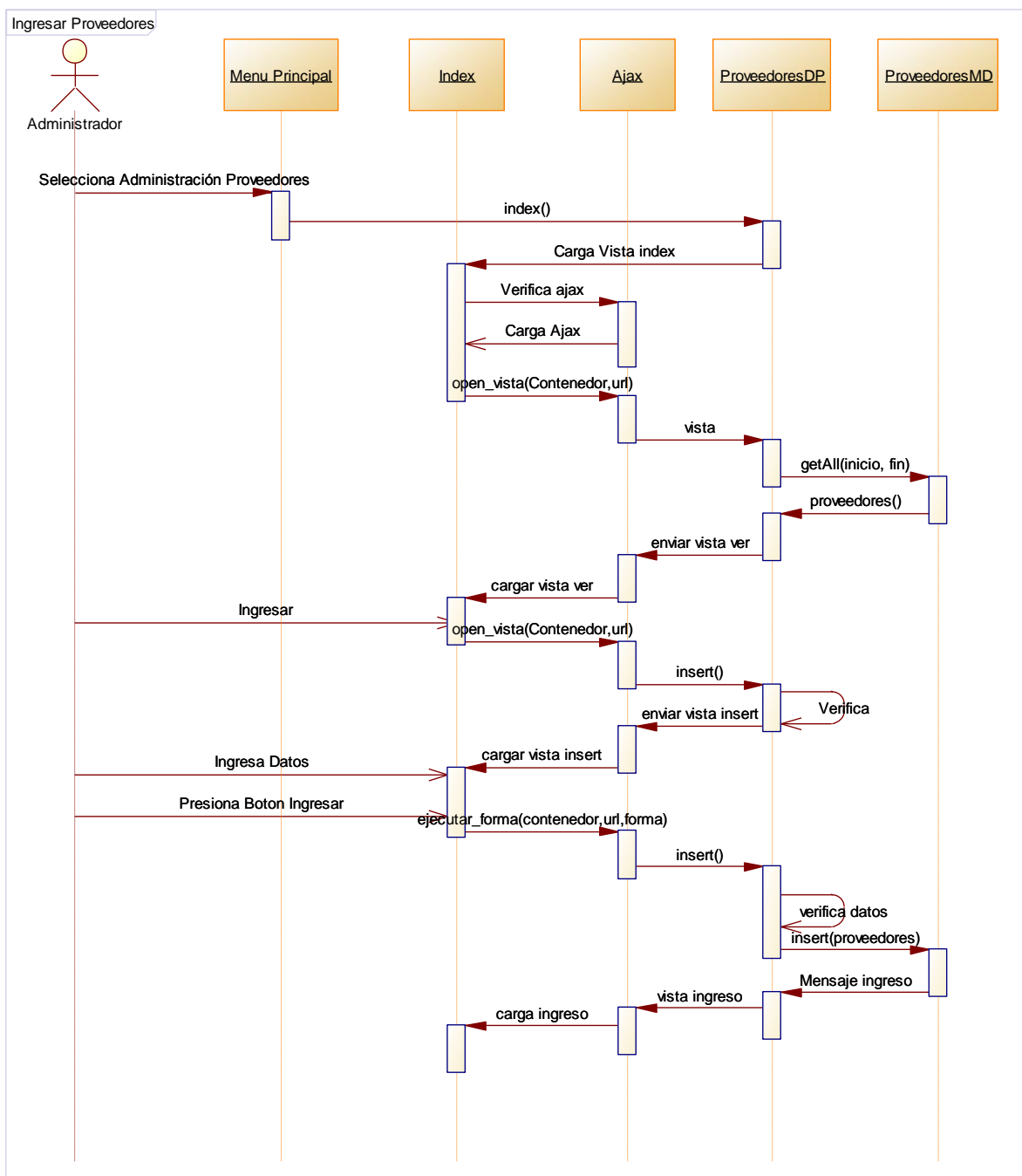


Diagrama 47: Diagrama Secuencia Ingresar Proveedores

Fuente: Diego Tobar

### 4.7.3.18. Modificar Proveedores

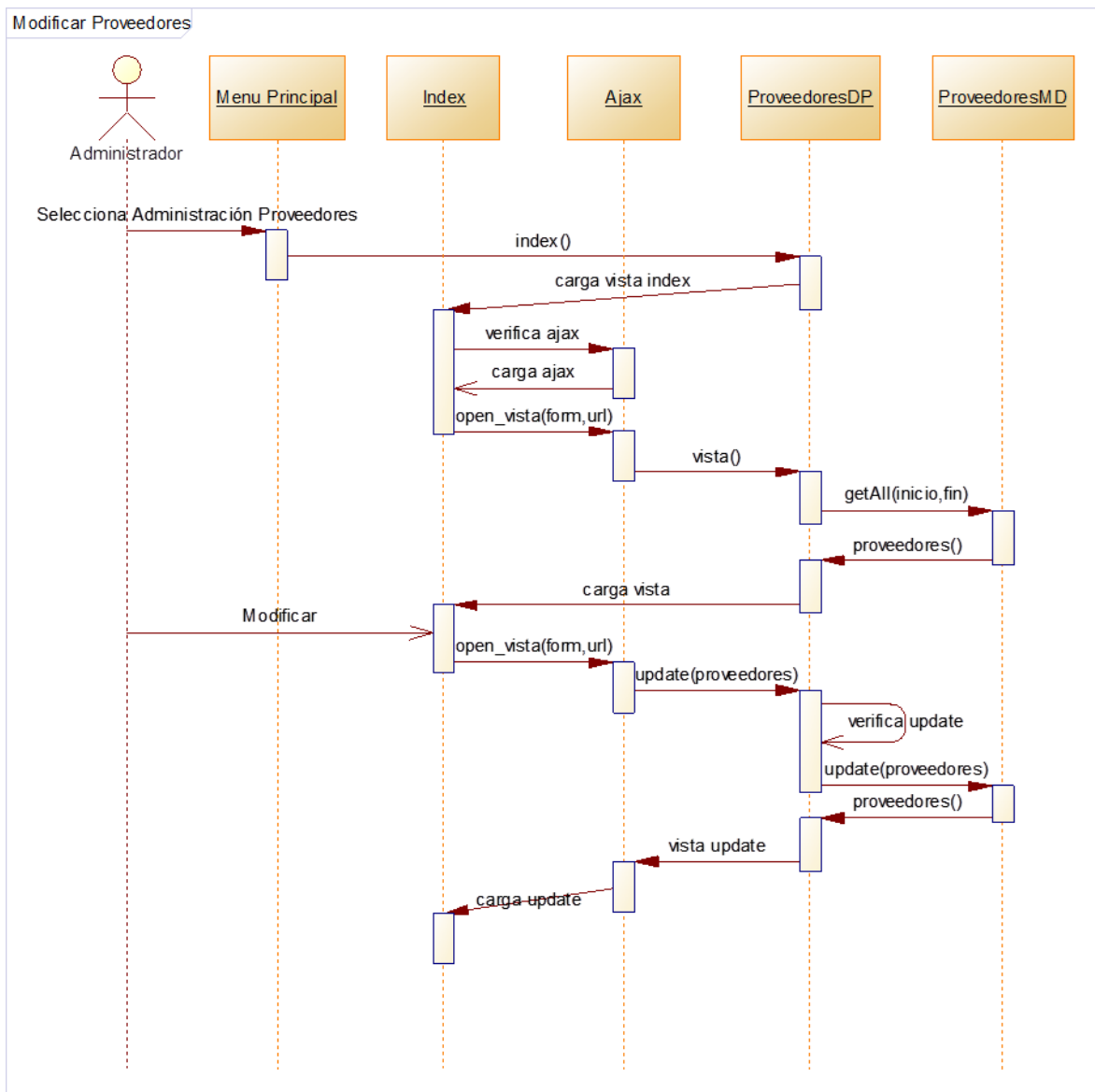


Diagrama 48: Diagrama Secuencia Modificar Proveedores

Fuente: Diego Tobar

### 4.7.3.19. Eliminar Proveedores

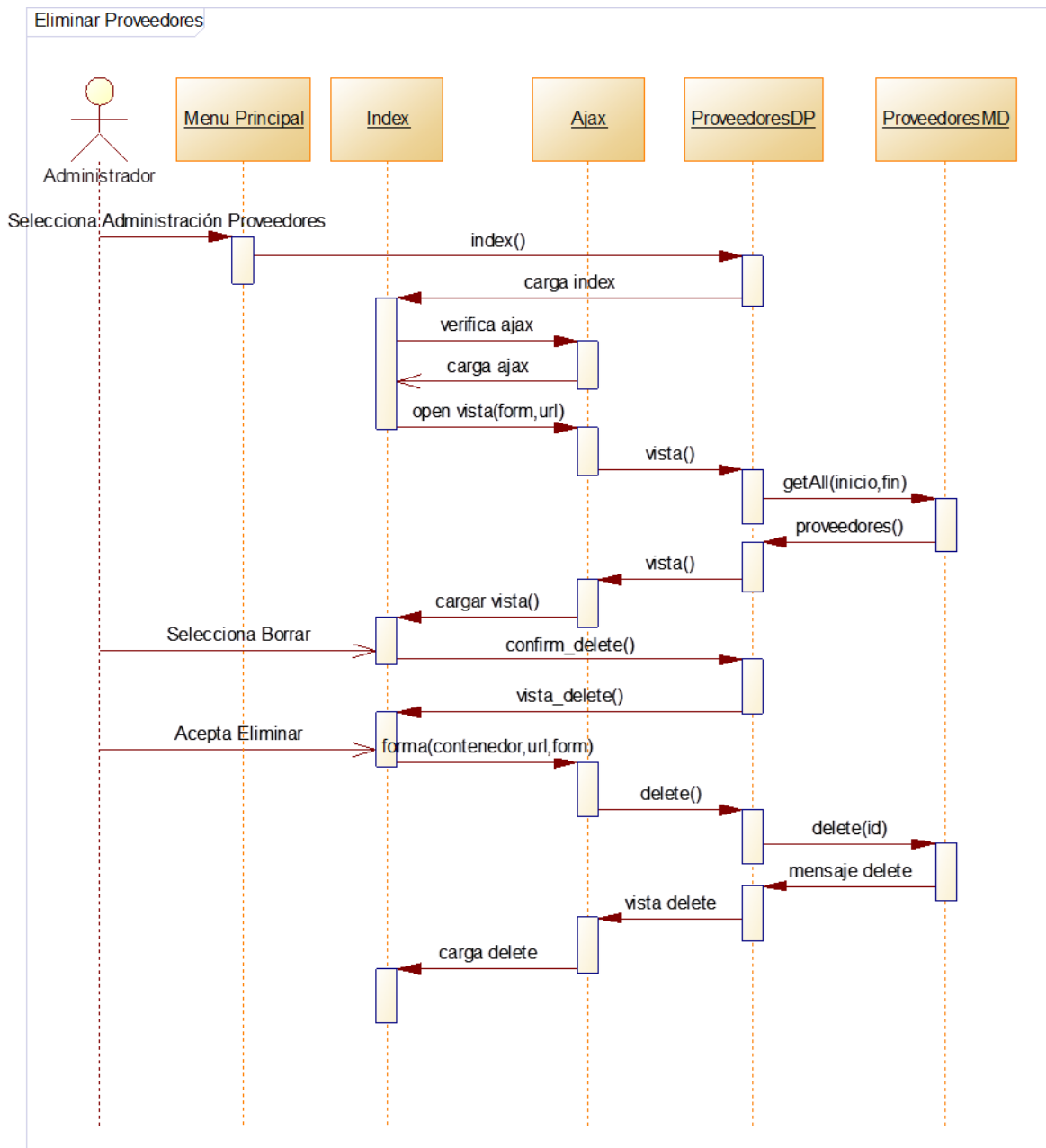


Diagrama 49: Diagrama Secuencia Eliminar Proveedores

Fuente: Diego Tobar

#### 4.7.3.20. Buscar Proveedores

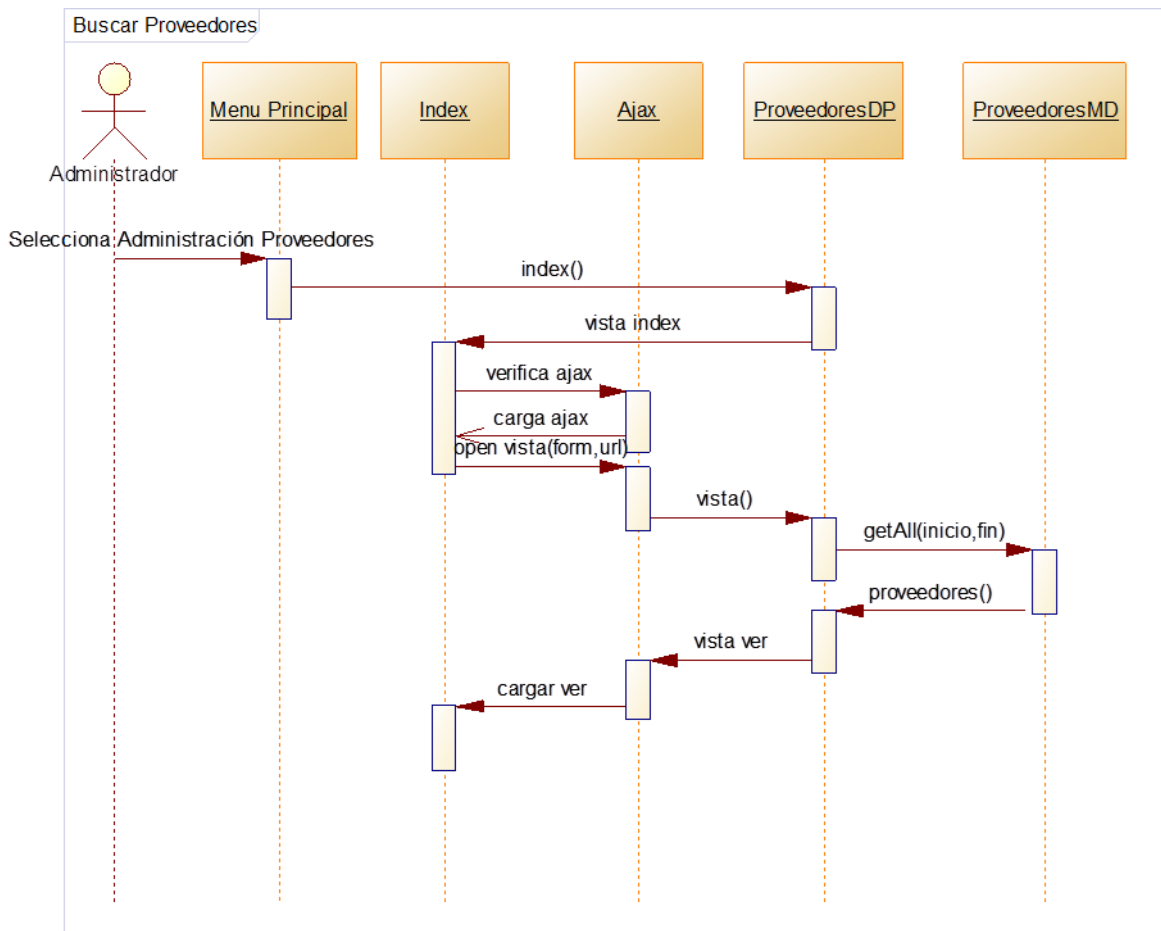


Diagrama 50: Diagrama Secuencia Buscar Proveedores

Fuente: Diego Tobar



## **CAPÍTULO 5**

### **5. Pruebas**

#### **5.1. Pruebas de la aplicación con la Técnica FLOOT (testing)**

La técnica FLOOT (Full Life-Cycle Object-Oriented Testing) es una colección de técnicas para validar y verificar software orientado a objetos que maneja un ciclo de vida de pruebas para un sistema que pueden ser aplicadas según la necesidad del mismo.

El plan de pruebas para la aplicación está dado por dos pruebas importantes que plantea la técnica FLOOT y que han sido consideradas a ejecutarse, estas son:

- Pruebas de caja negra
- Pruebas de interfaz de usuario

##### **5.1.1. Pruebas de Caja Negra**

Las pruebas de caja negra son aquellas que examinan que las entradas apropiadas produzcan los resultados esperados, sin importar el funcionamiento interno del sistema sino el resultado final. Los resultados presentados a continuación son el producto de una etapa de pruebas realizadas frente a las transacciones más importantes de cada módulo del sistema y de acciones correctivas ante las pruebas que resultaron fallidas en el proceso de test de la aplicación.

Las acciones correctivas o ajustes en la aplicación se llevaron a cabo de forma inmediata antes las pruebas no superadas, lo cual quiere decir que se manejó versiones de pruebas hasta obtener el resultado definitivo y exitoso presentado a continuación:

**Análisis, Desarrollo e Implementación de una Aplicación Web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferretería Tobar Nolivos Cía. Ltda.**

<b>F1. Administración de Usuarios</b>			
<b>Caso Uso</b>	<b>Nombre Transacción</b>	<b>Caso Prueba</b>	<b>Resultado</b>
F1.1	Ingresar un usuario nuevo	Se crea el usuario DTOBAR con el perfil de administrador	OK
F1.1	Validar campos obligatorios	Se valida que no se pueda crear un usuario sin llenar los campos obligatorios	OK
F1.2	Modificar un usuario existente	Se modifica el usuario DTOBAR por DTOBAR88, cambiando privilegios de entrada al sistema.	OK
F1.3	Eliminar un usuario existente	Se elimina al usuario DTOBAR88	OK
F1.4	Mostrar usuarios existentes	Se muestra los usuarios registrados en la administración de usuarios	OK

**Tabla 23:** Pruebas Caja Negra Administración de Usuarios

**Fuente:** Diego Tobar

<b>F2. Administración de Clientes</b>			
<b>Caso Uso</b>	<b>Nombre Transacción</b>	<b>Caso Prueba</b>	<b>Resultado</b>
F2.1	Ingresar un cliente nuevo	Se crea el cliente Humberto Tobar con el HTOBAR con número cuenta HTOBAR	OK
F2.1	Validar campos obligatorios	Se valida que no se pueda crear un cliente sin llenar los campos obligatorios	OK
F2.2	Modificar un cliente existente	Se modifica el mail del cliente HTOBAR.	OK
F2.3	Eliminar un cliente existente	Se elimina al usuario HTOBAR	OK
F2.4	Mostrar clientes existentes	Se muestra los clientes registrados en la administración de clientes.	OK

**Tabla 24:** Pruebas Caja Negra Administración de Clientes

**Fuente:** Diego Tobar

<b>F3. Administración de Productos</b>			
<b>Caso Uso</b>	<b>Nombre Transacción</b>	<b>Caso Prueba</b>	<b>Resultado</b>
F3.1	Ingresar un producto nuevo	Se crea el producto Cemento Chimborazo con el código CC0001	OK
F3.1	Validar campos obligatorios	Se valida que no se pueda crear un producto sin llenar los campos obligatorios	OK
F3.2	Modificar un producto existente	Se modifica el producto con código CC0001, por el nuevo código CC0002	OK
F3.3	Eliminar un producto existente	Se elimina el producto con código CC0002.	OK
F3.4	Mostrar productos existentes	Se muestra los productos registrados en la administración de productos.	OK

**Tabla 25:** Pruebas Caja Negra Administración de Productos

**Fuente:** Diego Tobar

<b>F4. Administración de Órdenes de Compra</b>			
<b>Caso Uso</b>	<b>Nombre Transacción</b>	<b>Caso Prueba</b>	<b>Resultado</b>
F4.1	Ingresar una orden de compra	Se ingresa el código del proveedor CCH0001	OK
F4.1	Buscar productos por filtro de nombre/código	Se ingresa el producto por su nombre o por su código.	OK
F4.1	Validar campos obligatorios	Se valida que no se pueda ingresar un producto sin llenar los campos obligatorios.	OK
F4.2	Modificar productos de la orden de compra	Se modifica la cantidad de ingreso del producto CC0002 y se edita su cantidad.	OK

**Análisis, Desarrollo e Implementación de una Aplicación Web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferretería Tobar Nolivos Cía. Ltda.**

F4.4	Reportar por pantalla o por listado las órdenes de compra	Se muestra las órdenes de compra registradas en la administración.	OK
------	---	--	----

**Tabla 26:** Pruebas Caja Negra Administración de Órdenes de Compra

**Fuente:** Diego Tobar

<b>F5. Administración de Proveedores</b>			
<b>Caso Uso</b>	<b>Nombre Transacción</b>	<b>Caso Prueba</b>	<b>Resultado</b>
F5.1	Ingresar un proveedor nuevo	Se crea el proveedor Cemento Chimborazo con el número cuenta CCH0001.	OK
F5.1	Validar campos obligatorios	Se valida que no se pueda crear un proveedor sin llenar los campos obligatorios.	OK
F5.2	Modificar un proveedor existente	Se modifica el teléfono del proveedor CCH0001.	OK
F5.3	Eliminar un proveedor existente	Se elimina al usuario CCH0001.	OK
F5.4	Mostrar proveedores existentes	Se muestra los proveedores registrados en la administración de proveedores.	OK

**Tabla 27:** Pruebas Caja Negra Administración de Proveedores

**Fuente:** Diego Tobar

### **5.1.2. Pruebas de Interfaz de Usuario**

Consiste en probar la interfaz de usuario para garantizar que cumple los requerimientos definidos. Usualmente se refiere a la prueba de interfaz de usuario grafica de la mayoría de componentes que el usuario pueda observar y apreciar frente a un monitor.

Para las pruebas antes mencionadas se siguió la misma logística que se aplicó en las pruebas de caja negra, por lo tanto se presenta a continuación el resultado definitivo y exitoso:

<b>F1. Administración de Usuarios</b>			
<b>Caso Uso</b>	<b>Nombre Transacción</b>	<b>Caso Prueba</b>	<b>Resultado</b>
F1.1	Validar la correcta ortografía.	Se valida que no existan errores ortográficos.	OK
F1.1	Validar textos legibles y entendibles.	Se valida que se pueda leer bien los textos y que sean comprensibles.	OK
F1.2	Validar el correcto despliegue de la información de usuarios.	Se valida que la información de los usuarios se despliegue de forma correcta.	OK
F1.3	Validar la correcta posición de los componentes (cajas de texto, etiquetas, botones)	Se valida el posicionamiento de los componentes.	OK
F1.4	Validar el correcto despliegue de las listas de información de usuarios.	Se valida la lista de usuarios con sus respectivos datos.	OK

**Tabla 28:** Pruebas Interfaz Administración de Usuarios

**Fuente:** Diego Tobar

<b>F2. Administración de Clientes</b>			
<b>Caso Uso</b>	<b>Nombre Transacción</b>	<b>Caso Prueba</b>	<b>Resultado</b>
F2.1	Validar la correcta ortografía.	Se valida que no existan errores ortográficos.	OK
F2.1	Validar textos legibles y entendibles.	Se valida que se pueda leer bien los textos y que sean comprensibles.	OK
F2.2	Validar el correcto despliegue de la información de clientes.	Se valida que la información de los clientes se despliegue de forma correcta.	OK
F2.3	Validar la correcta posición de los componentes (cajas de texto, etiquetas, botones)	Se valida el posicionamiento de los componentes.	OK

**Análisis, Desarrollo e Implementación de una Aplicación Web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferrería Tobar Nolivos Cía. Ltda.**

F2.4	Validar el correcto despliegue de las listas de información de clientes	Se valida la lista de clientes con sus respectivos datos.	OK
------	---	---	----

**Tabla 29:** Pruebas Interfaz Administración de Clientes

**Fuente:** Diego Tobar

<b>F3. Administración de Productos</b>			
<b>Caso Uso</b>	<b>Nombre Transacción</b>	<b>Caso Prueba</b>	<b>Resultado</b>
F3.1	Validar la correcta ortografía.	Se valida que no existan errores ortográficos.	OK
F3.1	Validar textos legibles y entendibles.	Se valida que se pueda leer bien los textos y que sean comprensibles.	OK
F3.2	Validar el correcto despliegue de la información de productos.	Se valida que la información de los productos se despliegue de forma correcta.	OK
F3.3	Validar la correcta posición de los componentes (cajas de texto, etiquetas, botones)	Se valida el posicionamiento de los componentes.	OK
F3.4	Validar el correcto despliegue de las listas de información de productos.	Se valida la lista de productos con sus respectivos datos.	OK

**Tabla 30:** Pruebas Interfaz Administración de Productos

**Fuente:** Diego Tobar

<b>F4. Administración de Órdenes de Compra</b>			
<b>Caso Uso</b>	<b>Nombre Transacción</b>	<b>Caso Prueba</b>	<b>Resultado</b>
F4.1	Validar la correcta ortografía.	Se valida que no existan errores ortográficos.	OK
F4.1	Validar textos legibles y entendibles.	Se valida que se pueda leer bien los textos y que sean comprensibles.	OK

**Análisis, Desarrollo e Implementación de una Aplicación Web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferrería Tobar Nolivos Cía. Ltda.**

F4.1	Validar el correcto despliegue de la información de las órdenes de compra.	Se valida que la información de las órdenes de compra estén correctas.	OK
F4.2	Validar la correcta posición de los componentes (cajas de texto, etiquetas, botones)	Se valida el posicionamiento de los componentes.	OK

**Tabla 31:** Pruebas Interfaz Administración de Órdenes de Compra

**Fuente:** Diego Tobar

<b>F5. Administración de Proveedores</b>			
<b>Caso Uso</b>	<b>Nombre Transacción</b>	<b>Caso Prueba</b>	<b>Resultado</b>
F5.1	Validar la correcta ortografía.	Se valida que no existan errores ortográficos.	OK
F5.1	Validar textos legibles y entendibles.	Se valida que se pueda leer bien los textos y que sean comprensibles.	OK
F5.2	Validar el correcto despliegue de la información de proveedores.	Se valida que la información de los proveedores se despliegue de forma correcta.	OK
F5.3	Validar la correcta posición de los componentes (cajas de texto, etiquetas, botones)	Se valida el posicionamiento de los componentes.	OK
F5.4	Validar el correcto despliegue de las listas de información los proveedores.	Se valida la lista de proveedores con sus respectivos datos.	OK

**Tabla 32:** Pruebas Interfaz Administración de Proveedores

**Fuente:** Diego Tobar

## **CAPÍTULO 6**

### **6. Conclusiones y Recomendaciones**

#### **6.1. Conclusiones**

1. La etapa más importante del proyecto es la identificación de los requerimientos del sistema puesto que ayudo a entender el giro del negocio y poder automatizar los principales procesos que requería la empresa, la ayuda del cliente y su conocimiento del negocio permitió que el proceso de levantamiento de requerimientos se realice en el menor tiempo posible.
2. La utilización de la metodología SCRUM fue muy importante en el desarrollo de la aplicación puesto que su principal característica es incorporar cambios sobre el desarrollo del sistema y a la vez permite mantener un control sobre el avance del mismo, esto agilitó en gran medida el avance del proyecto.
3. El seguir paso a paso las características de la metodología de programación SCRUM, permitió crear un software completo y de calidad, debido a que el enfoque de trabajo entre el desarrollador y el cliente es parte fundamental, se simplificó errores, mejoró las funcionalidades y se obtuvo un resultado positivo con la completa elaboración del producto.
4. El cliente participo en la toma de decisiones en el proceso de desarrollo del producto, realimentando con su visión para el mejoramiento del producto en cada momento, esto permite una mejor estrategia de satisfacción del cliente por lo que al momento de terminar el producto le agrega el valor a lo que necesita el cliente como parte de la solución.
5. La automatización del sistema ha permitido llevar de mejor manera la información de los productos, clientes y órdenes de compra, proporcionando alertas sobre el estado de los productos y la escasez de los mismos, la información detallada en tiempo real de los órdenes de compra así también como el correcto funcionamiento de la información de los clientes.

6. El uso de un framework PHP como Codeigniter en lugar de PHP puro mejoró los tiempos de desarrollo y la reutilización del código sin comprometer el desempeño del mismo.
7. La arquitectura implementada para el sistema, hizo del desarrollo una tarea más fácil y flexible al momento de aplicar soluciones a los problemas presentados por los diferentes módulos del sistema, y al mismo tiempo permitió que las transacciones del software se ejecuten a mayor velocidad.
8. Este sistema en un corto plazo se ampliará con nuevas funcionalidades puesto que el cliente nos comentó que quisiera automatizar más sus procesos de negocio como son por ejemplo: el área de despacho, transportes y puntos de venta; debido a que sistema tiene la capacidad de ampliarse por el modelamiento que se realizó desde un comienzo, esperamos realizarlo cuando el cliente nos autorice hacer la ampliación del sistema.

## **6.2. Recomendaciones**

1. Se debe conocer los requerimientos reales del usuario antes del desarrollo de la aplicación, de no hacerlo se presentarán dificultades en fases posteriores del desarrollo y si se realizan cambios muy severos complica y alarga los tiempos de implementación del sistema.
2. Es importante que siempre se tenga un respaldo actualizado de la Base de Datos, porque los fallos que tienen los sistemas informáticos puede ocasionar una pérdida total de la información, por lo tanto se sugiere que dicho respaldo sea realizado al menos una vez por semana.
3. Se recomienda utilizar la presente documentación como base para aplicar cambios o incrementos de funcionalidades al sistema, ya que así se mantendría el estándar de la metodología y de la programación utilizada para el desarrollo del software.
4. Se recomienda que los usuarios al operar el sistema tengan conocimientos básicos de computación y manejo de navegadores web, así se podrá utilizar el sistema de un forma fácil y adecuada.

5. Se debe poner especial atención al administrador de la aplicación en la asignación de accesos a los módulos del sistema a los usuarios, puesto que nunca se sabe los verdaderos usos que los usuarios tengan con la aplicación y con la información que tiene, con esto evitaremos fuga de información así como también posibles fraudes por personas no autorizadas al sistema.
6. A la Pontificia Universidad Católica del Ecuador en especial a la Facultad de Ingeniería Escuela de Sistemas recomendaría eliminar del pensum ciertas materias de la carrera que no influyen tanto en el aprendizaje continuo del estudiante, y enfocar con nuevas materias que ayuden al estudiante a tener un aprendizaje práctico.
7. Al haber puesto en marcha este proyecto, me he fijado en que sería interesante ampliar la aplicación a una nueva versión 2.0 con más funcionalidades como por ejemplo: punto de venta, envíos de mercadería, recursos humanos y reportes gráficos.
8. Finalmente la aplicación me ofreció una apertura de ideas y conocimientos dando como resultado a corto plazo una idea de modelo de negocio en donde podemos ofrecer la aplicación como un servicio y que cada funcionalidad adicional podemos cobrar un rédito por la ampliación del sistema base, este proyecto puede ser difundido a manera local como también a nivel internacional.

## **Bibliografía**

- Agile Alliance. (2014). <http://www.agilemanifesto.org>. Acceso el 11 de febrero de 2014
- Agile Modeling Practices. (17 de Marzo de 2009). [www.agilemodeling.com](http://www.agilemodeling.com) . Acceso el 14 de Febrero de 2014
- Bauer, F. (1992). *Ingeniería Software*. Munich: Information Processing.
- Beck, K. (2004). *Programación Extrema* . Addison-Wesley.
- Cáceres, M. (2000). *El proceso de Desarrollo en las Aplicaciones Web*. Valladolid: Ed. Delgado.
- Ellislab. (2013). <https://ellislab.com/codeigniter>. Acceso el 24 de Marzo de 2014
- Gutierrez, J. (2013). [http://www.lsi.us.es/~javierj/investigacion\\_ficheros/Framework.pdf](http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf). Acceso el 5 de Marzo de 2014
- Instituto Celaya. (2009). [http://sisinfo.itc.mx/ITC-APIRGG/Fundamentos\\_PHP/Introduccion.htm](http://sisinfo.itc.mx/ITC-APIRGG/Fundamentos_PHP/Introduccion.htm). Acceso el 5 de Marzo de 2014
- Jacobson, I. B. (2000). *El proceso unificado de desarrollo de software*. Addison-Wesley Professional.
- Keeton, M. (2006). *Microsoft Solutions Framework (MSF)*. Van Haren Publishing.
- Kruchten, P. (1999). *Rational Unified Process - An Introduction*. Addison-Wesley.
- Mora, S. (2001). *Programación de aplicaciones web: historia, principios básicos y clientes web*. Editorial Club Universitario.
- Palacio, J. (2008). *Flexibilidad con Scrum*. Safe Creative.
- Pavón, J. (2008). *Creación de un portal PHP y MySQL*. Mexico: Grupo Alfa Omega.
- Poppendieck, M. (2003). *Lean Software Development an Agile Toolkit*. Addison-Wesley .
- Programming, E. (2014). [www.extremeprogramming.org](http://www.extremeprogramming.org). Acceso el 14 de Febrero de 2014
- Wikipedia. (2013). <http://es.wikipedia.org/wiki/Modelo%20%80%93vista%20%80%93controlador>. Acceso el 24 de Marzo de 2014

# **ANEXO A**

## **Documento de Especificación de Requerimientos de Software**

### **Introducción**

#### **1. Propósito**

Este documento de especificaciones requerimientos de software tiene como propósito definir las especificaciones funcionales, no funcionales y del sistema para la implementación de una aplicación web que permitirá administrar y consultar los procesos de la empresa Ferretería Tobar Nolivos Cía. Ltda.

#### **2. Alcance**

Implementar una aplicación web en la empresa Ferretería Tobar Nolivos Cía. Ltda. que permita mejorar y controlar sus procesos de registro de órdenes de compra, registro de clientes y registro de productos. La aplicación permitirá una flexibilidad de acceso a la información, igualmente una actualización dinámica de la misma.

#### **3. Definiciones, siglas y abreviaciones**

##### **3.1. Del negocio**

- a. Cliente:** Es la persona o empresa receptora de un bien, servicio o producto.
- b. Producto:** Es un objeto que se ofrece en una mercado para satisfacer necesidad del consumidor
- c. Órdenes de compra:** Es un documento que emite el comprador para pedir mercadería al vendedor indicando cantidad, detalle, precio y pago.
- d. Vendedor:** Es la persona que tiene encomendada la acción de vender o comercializar productos de una empresa

## **3.2. Del Sistema**

- a. Administrar:** Acción de agregar, modificar, eliminar y consultar la información de un determinado objeto o persona.
- b. Visitante:** Persona que navega o utiliza la aplicación.
- c. Usuario:** Persona que puede ingresar a la zona privada de la aplicación, a través de un proceso de autenticación utilizando nombre de usuario y contraseña.
- d. Permiso:** Parámetro que especifica si su poseedor dispone de acceso a una determinada función del sistema o a una parte de la interfaz de usuario del sistema.
- e. Rol:** Es un conjunto de permisos que puede asignarse a un usuario.
- f. Administrador del Sistema:** Persona encargada de ofrecer el soporte técnico.
- g. Pruebas:** Proceso mediante el cual se realizan actividades para verificar la óptima función del sistema.

### **3.2.1. De Tecnología**

- a. Dominio:** Nombre base que agrupa a un conjunto de equipos o dispositivos y que permite proporcionar nombres de equipo más fácilmente recordables en lugar de una dirección IP numérica
- b. Servidor Web:** Es un programa que implementa el protocolo HTTP (hypertext transfer protocolo).
- c. URL:** Uniform Resource Locator, es una secuencia de caracteres, de acuerdo a un formato estándar que se usa para nombre recursos, documentos e imágenes en internet.
- d. Base de datos:** Es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso.
- e. MySQL:** Sistema de gestión de base de datos relacional.
- f. Apache:** Servidor Web de código abierto.
- g. Protocolo IP:** Es un protocolo no orientado a la conexión, usado tanto por el origen como por el destino para la comunicación a través de una red (Internet).
- h. Dirección IP:** Número que identifica de manera lógica y jerárquica a una interfaz de un dispositivo dentro de una red.

### 3.3.Descripción

En el presente documento se encontrará la información acerca de las características del producto de software, interfaces del usuario, interfaces del sistema, características de los usuarios, descripción de los requerimientos funcionales, no funcionales y del sistema, los cuales se representarán en el siguiente formato:

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
Referencia de requerimiento	Nombre de requerimiento		Fecha de Especificación	Importancia del requerimiento
<b>Descripción</b>	Descripción del requerimiento			
Entradas	Fuente	Salida	Destino	Restricciones
Entradas del requerimiento	Fuentes de entrada	Salidas Requerimiento	Donde se lleva la salida	Restricciones a tener en cuenta
<b>Proceso</b>	Descripción detallada de las actividades que realiza el requerimiento			
<b>Efecto Colateral</b>	Efectos generados a otros procesos o sistemas, si es el caso			

**Código:**

**RF:** Requerimiento Funcional

**RFN:** Requerimiento No Funcional

**RI:** Requerimiento de Interfaz

## **4. Descripción General**

### **4.1. Perspectivas del Producto**

La aplicación Web será un producto diseñado para trabajar en entornos Web, lo que permitirá su utilización de forma descentralizada, además trabajará de manera independiente por lo tanto no interactúa con otros sistemas.

### **4.2. Funciones del Producto**

La aplicación web permitirá realizar las siguientes funciones:

- a. Administración de Usuarios:** El administrador del sistema podrá gestionar los usuarios (agregar, modificar, eliminar, buscar)
- b. Administración de Clientes:** El administrador del sistema podrá gestionar los clientes (agregar, modificar, eliminar, buscar)
- c. Administración de Productos:** El administrador del sistema podrá gestionar los productos (agregar, modificar, eliminar, buscar)
- d. Administración de Órdenes de Compra:** El administrador del sistema podrá gestionar las órdenes de compra (agregar, modificar, eliminar, buscar)
- e. Administración de Proveedores:** El administrador del sistema podrá gestionar los proveedores (agregar, modificar, eliminar, buscar)

### **4.3. Características del Usuario**

La aplicación web contendrá 3 tipos de usuarios que interactúan y lo administran:

- Administrador del Sistema
- Vendedor
- Gerente Local

### **4.3.1. Perfil del usuario**

Cada usuario tendrá un perfil específico para que su interacción con el sistema sea correcto y no conlleve a fallos:

**Administrador del Sistema:** Usuario con gran conocimiento en el manejo del sistema con una previa capacitación por parte de la entidad. Encargado de manejar el sistema con gran responsabilidad sobre los criterios de permisos sobre los usuarios.

**Vendedor:** Persona que interactúa continuamente con el sistema, debe tener el conocimiento de la aplicación para su correcta utilización.

**Gerente Local:** Usuario con gran conocimiento y administración de información de los productos, clientes, órdenes de compra y proveedores, el cual tendrá una capacitación del sistema para una correcta administración del mismo.

## **4.4. Restricciones**

### **4.4.1. Políticas Reguladoras**

La aplicación se desarrollará mediante software de licencia abierta por lo tanto no se deberá pagar por el uso de:

- Servidor Web
- Sistema de Gestión de Base de Datos
- Lenguaje de Programación

Por lo tanto, la utilización de estos programas se hará mediante las políticas establecidas por este tipo de licenciamiento.

### **4.4.2. Limitaciones de Hardware**

Para esta aplicación será necesario un computador servidor en el cual se instalara el servidor WEB apache, MySQL, PHP y la aplicación web.

### **4.4.3. Interfaces con otras aplicaciones**

Debido a que el sistema no interactúa con otros sistemas y es autónomo no se desarrollaran interfaces con otras aplicaciones. Las conexiones necesarias para la utilización del servidor Web, MySQL, PHP, se harán por medio de la configuración de los programas.

### **4.4.4. Funciones de control**

El sistema debe controlar los permisos que tiene cada usuario para su accesibilidad de una manera correcta, de tal forma que pueda acceder a la información que le corresponde de acuerdo a su rol.

### **4.4.5. Requisitos de lenguaje**

Todo el material que se realice para el usuario y la aplicación debe estar en lenguaje español.

### **4.4.6. Protocolos Señalados**

Se usará protocolos de comunicación TCP/IP, HTTP.

## **5. Requisitos Especificaos**

### **5.1. Interfaces Externas**

La aplicación Web no tendrá interconexión con otros sistemas de información, por lo tanto no es necesaria la utilización de interfaz alguna. La relación con un servidor Web, DNS y Gestor de Base de Datos se hará a través de los archivos de configuración de estos.

## 5.2. Requisitos Funcionales

### 5.2.1. Ingreso a la aplicación

SRS – Especificación de Requerimientos				
Código	Nombre	Fecha	Grado Necesidad	
RF_1.1.1	Apertura de la aplicación	12/05/2014	Esencial	
Descripción	El sistema debe permitir ingresar por medio de la cabecera HTTP			
Entradas	Fuente	Salida	Destino	Restricciones
El IP del computador donde este conectado	Red	Pantalla Registro Usuario	Pantallas	Ingreso solo para personal autorizado
Proceso	El sistema deberá tener un nombre por medio del cual permita su ingreso digitándolo en la barra del navegador, todo esto debidamente configurado en el servidor web.			
Efecto Colateral	Link aplicación web.			

### 5.2.2. Administración de usuarios

SRS – Especificación de Requerimientos				
Código	Nombre	Fecha	Grado Necesidad	
RF_2.2.1	Ingreso o eliminación de roles.	12/05/2014	Esencial	
Descripción	El sistema debe permitir el ingreso de uno o más roles.			
Entradas	Fuente	Salida	Destino	Restricciones
Usuarios del sistema	Formulario de ingreso datos	Usuarios con su rol asignado	Base de Datos	Cada usuario un rol como mínimo.
Proceso	En la administración del sistema tendrá la opción de administrar usuarios, al ingresar a esta opción se desplegara un listado de los usuarios, se les asignará roles. Una vez que el usuario administrador del sistema asigne los roles a los usuarios esto se guardará y se pedirá confirmación para almacenar los cambios,			
Efecto Colateral	No aplica			

**Análisis, Desarrollo e Implementación de una Aplicación Web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferretería Tobar Nolivos Cía. Ltda.**

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RF_2.2.2	Creación de usuarios		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la creación de diferentes usuarios.			
Entradas	Fuente	Salida	Destino	Restricciones
Datos Usuario: Password, Nombre, Tipo Usuario, Cedula, Mail, Cargo	Formulario de ingreso datos	Usuarios con acceso al sistema	Base de Datos	Los campos son obligatorios y no puede haber usuarios repetidos.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá administrar los usuarios definiendo su rol. Se verificara la información para la creación y luego dará la opción guardar, el sistema creara el usuario en la BD.			
<b>Efecto Colateral</b>	No aplica			

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RF_2.2.3	Actualización password		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la actualización de los password de los usuarios.			
Entradas	Fuente	Salida	Destino	Restricciones
Identificación y password	Formulario de ingreso datos	Confirmación por pantalla	Base de Datos	Máximo 10 caracteres.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá Administrar los usuarios para la modificación de su password, por medio de un formulario ya definido en el cual se solicitara el número de identificación y el password actual y el nuevo password. Al confirmar la operación su nuevo password se almacenara en la base de datos y el usuario lo verificara en su próximo ingreso			
<b>Efecto Colateral</b>	No aplica			

**Análisis, Desarrollo e Implementación de una Aplicación Web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferretería Tobar Nolivos Cía. Ltda.**

<b>SRS – Especificación de Requerimientos</b>				
<b>Código</b>	<b>Nombre</b>		<b>Fecha</b>	<b>Grado Necesidad</b>
RF_2.2.4	Modificar o eliminar usuarios		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la modificación o eliminación de los usuarios.			
<b>Entradas</b>	<b>Fuente</b>	<b>Salida</b>	<b>Destino</b>	<b>Restricciones</b>
Identificación usuario	Formulario de ingreso datos	Confirmación por pantalla	Base de Datos	El administrador del sistema no podrá ser eliminado.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá listar los usuarios para modificar o eliminar.			
<b>Efecto Colateral</b>	Usuario eliminado no puede ingresar al sistema.			

<b>SRS – Especificación de Requerimientos</b>				
<b>Código</b>	<b>Nombre</b>		<b>Fecha</b>	<b>Grado Necesidad</b>
RF_2.2.5	Consultar usuarios		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la consulta de los usuarios.			
<b>Entradas</b>	<b>Fuente</b>	<b>Salida</b>	<b>Destino</b>	<b>Restricciones</b>
Identificación usuario	Formulario de ingreso datos	Confirmación por pantalla	Base de Datos	Solo se consulta con usuarios creados en el sistema.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá consultar individualmente o en lista a los usuarios del sistema.			
<b>Efecto Colateral</b>	Usuario eliminado no puede consultar al sistema.			

### 5.2.3. Administración de clientes

SRS – Especificación de Requerimientos				
Código	Nombre	Fecha	Grado Necesidad	
RF_2.3.1	Ingresar clientes.	12/05/2014	Esencial	
<b>Descripción</b>	El sistema debe permitir la creación de diferentes clientes.			
Entradas	Fuente	Salida	Destino	Restricciones
Datos Cliente: Código, Nombre, Tipo Cliente, Cedula, RUC, Mail, Dirección	Formulario de ingreso datos	Confirmación por pantalla de clientes.	Base de Datos	Los campos son obligatorios y no puede haber datos repetidos.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá administrar los clientes. Se verificara la información para la creación y luego dará la opción guardar, el sistema creara el cliente en la BD.			
<b>Efecto Colateral</b>	No aplica			

SRS – Especificación de Requerimientos				
Código	Nombre	Fecha	Grado Necesidad	
RF_2.3.2	Modificar o eliminar clientes	12/05/2014	Esencial	
<b>Descripción</b>	El sistema debe permitir la modificación o eliminación de clientes.			
Entradas	Fuente	Salida	Destino	Restricciones
Código cliente, Descripción, Ruc/Cedula	Formulario de ingreso datos	Confirmación por pantalla	Base de Datos	El cliente será eliminado solo cuando se realice la operación.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá listar los usuarios para modificar o eliminar.			
<b>Efecto Colateral</b>	Clientes eliminados deberán ser creados nuevamente.			

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RF_2.3.3	Consultar clientes		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la consulta de los usuarios.			
Entradas	Fuente	Salida	Destino	Restricciones
Identificación cliente	Formulario de ingreso datos	Confirmación por pantalla	Base de Datos	Solo se consulta los clientes creados en el sistema.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá consultar individualmente o en lista a los clientes del sistema.			
<b>Efecto Colateral</b>	Cliente eliminado no puede consultar al sistema.			

#### 5.2.4. Administración de productos

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RF_2.4.1	Ingresar productos.		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la creación de diferentes productos.			
Entradas	Fuente	Salida	Destino	Restricciones
Datos Producto: Código, Descripción, Cantidad, Costo, PVP, Descuento	Formulario de ingreso datos	Confirmación por pantalla de productos.	Base de Datos	Los campos son obligatorios y no puede haber datos repetidos.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá administrar los productos. Se verificara la información para el ingreso y luego dará la opción guardar, el sistema creara el producto en la BD.			
<b>Efecto Colateral</b>	No aplica			

**Análisis, Desarrollo e Implementación de una Aplicación Web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferretería Tobar Nolivos Cía. Ltda.**

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RF_2.4.2	Modificar o eliminar clientes		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la modificación o eliminación de productos.			
Entradas	Fuente	Salida	Destino	Restricciones
Código producto, Descripción, Cantidad	Formulario de ingreso datos	Confirmación por pantalla	Base de Datos	El producto será eliminado solo cuando se realice la operación.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá listar los productos para modificar o eliminar.			
<b>Efecto Colateral</b>	Productos eliminados deberán ser creados nuevamente.			

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RF_2.4.3	Consultar productos		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la consulta de los productos.			
Entradas	Fuente	Salida	Destino	Restricciones
Identificación producto	Formulario de ingreso datos	Confirmación por pantalla	Base de Datos	Solo se consulta con productos creados en el sistema.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá consultar individualmente o en lista a los productos del sistema.			
<b>Efecto Colateral</b>	Producto eliminado no puede consultar al sistema.			

### 5.2.5. Administración Órdenes de Compra

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RF_2.5.1	Creación de órdenes de compra		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la creación de órdenes de compra.			
Entradas	Fuente	Salida	Destino	Restricciones
Datos: Numero Orden, Cliente, Productos, Cantidad, Total	Formulario de ingreso datos	Usuarios con acceso al sistema	Base de Datos	Los campos son obligatorios y no puede haber usuarios repetidos.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá administrar las órdenes de compra. Se verificara la información para la creación y luego dará la opción guardar, el sistema creara el usuario en la BD.			
<b>Efecto Colateral</b>	No aplica			

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RF_2.5.2	Modificar o eliminar órdenes de compra		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la modificación o eliminación de las órdenes de compra.			
Entradas	Fuente	Salida	Destino	Restricciones
Identificación orden de compra	Formulario de ingreso datos	Confirmación por pantalla	Base de Datos	No se podrá modificar órdenes de compra eliminadas.
<b>Proceso</b>	El administrador del sistema o un usuario con acceso a la administración de órdenes de compra tendrá una opción que le permitirá listar las órdenes de compra para modificar o eliminar.			
<b>Efecto Colateral</b>	Órdenes de compra eliminadas no se pueden ingresar al sistema.			

**Análisis, Desarrollo e Implementación de una Aplicación Web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferretería Tobar Nolivos Cía. Ltda.**

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RF_2.5.3	Consultar órdenes de compra		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la consulta de órdenes de compra.			
Entradas	Fuente	Salida	Destino	Restricciones
Identificación órdenes de compra	Formulario de ingreso datos	Confirmación por pantalla	Base de Datos	Solo se consulta con órdenes de compra creadas en el sistema.
<b>Proceso</b>	El administrador del sistema o un usuario con acceso a la administración de órdenes de compra tendrá una opción que le permitirá consultar individualmente o en lista las órdenes de compra.			
<b>Efecto Colateral</b>	Ordenes de compra eliminadas no se pueden consultar al sistema.			

### 5.2.6. Administración Proveedores

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RF_2.6.1	Ingresar Proveedores.		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la creación de diferentes proveedores.			
Entradas	Fuente	Salida	Destino	Restricciones
Datos Proveedor : Código, Nombre, Tipo Cliente, Cedula, RUC, Mail, Dirección	Formulario de ingreso datos	Confirmación por pantalla de proveedores.	Base de Datos	Los campos son obligatorios y no puede haber datos repetidos.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá administrar los proveedores. Se verificara la información para la creación y luego dará la opción guardar, el sistema creara el cliente en la BD.			
<b>Efecto Colateral</b>	No aplica			

**Análisis, Desarrollo e Implementación de una Aplicación Web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferrería Tobar Nolivos Cía. Ltda.**

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RF_2.6.2	Modificar o eliminar proveedores		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la modificación o eliminación de proveedores.			
Entradas	Fuente	Salida	Destino	Restricciones
Código proveedor, Descripción, Ruc/Cedula	Formulario de ingreso datos	Confirmación por pantalla	Base de Datos	El proveedor será eliminado solo cuando se realice la operación.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá listar los proveedores para modificar o eliminar.			
<b>Efecto Colateral</b>	Proveedores eliminados deberán ser creados nuevamente.			

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RF_2.6.3	Consultar proveedores		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la consulta de los proveedores.			
Entradas	Fuente	Salida	Destino	Restricciones
Identificación proveedor	Formulario de ingreso datos	Confirmación por pantalla	Base de Datos	Solo se consulta los clientes creados en el sistema.
<b>Proceso</b>	El administrador del sistema tendrá una opción que le permitirá consultar individualmente o en lista a los proveedores del sistema.			
<b>Efecto Colateral</b>	Proveedor eliminado no puede consultar al sistema.			

### 5.2.7. Navegación

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RI_2.7.1	Vínculos de navegación		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe presentar o contener vínculos de navegación entendibles.			
Entradas	Fuente	Salida	Destino	Restricciones
Textos o iconos	Hojas de estilo en cascada	Vinculo entendible	Usuario	Si los vínculos son imágenes deben tener un comentario tipo texto
<b>Proceso</b>	No aplica			
<b>Efecto Colateral</b>	El usuario puede navegar de una manera más fácil y dirigida.			

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RI_2.7.2	Imagen empresarial		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe en cada una de las paginas contener la imagen empresarial			
Entradas	Fuente	Salida	Destino	Restricciones
Logo	Formularios de inscripción datos	Despliegue de la imagen en pantallas	Usuario	El formato de la imagen debe ser tipo: jpg - gif
<b>Proceso</b>	La imagen será vinculada desde el archivo principal de estilos de la aplicación			
<b>Efecto Colateral</b>	El usuario interactúa continuamente con el logo de la empresa.			

SRS – Especificación de Requerimientos				
Código	Nombre	Fecha	Grado Necesidad	
RI_2.7.3	Ayudas tipo texto	12/05/2014	Esencial	
<b>Descripción</b>	El sistema debe indicar ayudas tipo texto en el ingreso de datos en los formatos			
Entradas	Fuente	Salida	Destino	Restricciones
Formularios	Diagrama de navegación	Ayudas tipo texto	Usuario	Idioma Español
<b>Proceso</b>	Los campos de todos los formatos o formularios tendrán ayuda tipo texto que permita al usuario guiarlo en su ingreso. Si existen restricciones deben indicarse por medio de esta herramienta			
<b>Efecto Colateral</b>	Menor posibilidad de error en el ingreso de los datos.			

### 5.3. Requisitos de Desarrollo

SRS – Especificación de Requerimientos				
Código	Nombre	Fecha	Grado Necesidad	
RNF_2.8.1	Cantidad de usuarios concurrentes	12/05/2014	Esencial	
<b>Descripción</b>	El número máximo dependerá de la capacidad del servidor. El sistema debe soportar un número de usuarios de acuerdo a los recursos de infraestructura.			
Entradas	Fuente	Salida	Destino	Restricciones
Infraestructura	Arquitectura del sistema	Número de usuarios soportados	Sistema	Si se desea incrementar los usuarios debe revisarse la infraestructura
<b>Proceso</b>	El usuario podrá ingresar tantas veces al sistema como le sea posible.			
<b>Efecto Colateral</b>	Si el administrador no ha planificado bien la estructura el sistema puede colapsar.			

**Análisis, Desarrollo e Implementación de una Aplicación Web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferretería Tobar Nolivos Cía. Ltda.**

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RNF_2.8.2	Cantidad de información almacenada		12/05/2014	Esencial
<b>Descripción</b>	El número máximo de información a registrar dependerá de la capacidad del servidor.			
Entradas	Fuente	Salida	Destino	Restricciones
Infraestructura	Arquitectura del sistema	Número de usuarios soportados	Sistema	Si se desea incrementar la información debe revisarse la infraestructura
<b>Proceso</b>	El usuario podrá ingresar tantas veces la información al sistema como le sea posible.			
<b>Efecto Colateral</b>	Si el administrador no ha planificado bien la estructura el sistema puede colapsar.			

SRS – Especificación de Requerimientos				
Código	Nombre		Fecha	Grado Necesidad
RNF_2.8.3	Base de Datos		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe permitir la manipulación de la información por medio de un motor de base de datos.			
Entradas	Fuente	Salida	Destino	Restricciones
Esquema de Base de Datos	Documentación	Ejecución de consultas	Scripts de la aplicación	La carga de la aplicación de la base de datos debe estar distribuida
<b>Proceso</b>	Las consultas que permiten la interacción de los scripts con la base de datos debe permitir interactuar con el motor de base de datos.			
<b>Efecto Colateral</b>	No aplica.			

**Análisis, Desarrollo e Implementación de una Aplicación Web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferrería Tobar Nolivos Cía. Ltda.**

SRS – Especificación de Requerimientos				
Código	Nombre	Fecha	Grado Necesidad	
RNF_2.8.4	Sistema Operativo	12/05/2014	Esencial	
<b>Descripción</b>	El sistema debe permitir instalar en un sistema operativo Windows y/o Linux, y los clientes pueden correrlo en los mismo sistemas operativos.			
Entradas	Fuente	Salida	Destino	Restricciones
Características de la arquitectura	Arquitectura del sistema	No aplica	No aplica	Se instalara en sistema
<b>Proceso</b>	La aplicación es independiente del sistema operativo.			
<b>Efecto Colateral</b>	No aplica.			

SRS – Especificación de Requerimientos				
Código	Nombre	Fecha	Grado Necesidad	
RNF_2.8.5	Servidor Web	12/05/2014	Esencial	
<b>Descripción</b>	El sistema debe ejecutarse bajo el servidor web apache			
Entradas	Fuente	Salida	Destino	Restricciones
Características de la arquitectura	No aplica.	No aplica.	No aplica.	No aplica.
<b>Proceso</b>	Se entregara toda la información necesaria para configurar el servidor de la aplicación.			
<b>Efecto Colateral</b>	No aplica.			

**Análisis, Desarrollo e Implementación de una Aplicación Web orientada para el registro de órdenes de compra, clientes y productos para la empresa Ferretería Tobar Nolivos Cía. Ltda.**

---

<b>SRS – Especificación de Requerimientos</b>				
<b>Código</b>	<b>Nombre</b>		<b>Fecha</b>	<b>Grado Necesidad</b>
RNF_2.8.6	Disponibilidad del sistema		12/05/2014	Esencial
<b>Descripción</b>	El sistema debe ofrecer una disponibilidad completa.			
<b>Entradas</b>	<b>Fuente</b>	<b>Salida</b>	<b>Destino</b>	<b>Restricciones</b>
Arquitectura de diseño	No aplica.	No aplica.	No aplica.	Depende de controladores y factores externos
<b>Proceso</b>	El sistema deberá estar disponible a menos que sucedan causas externas como: pérdida de fluido eléctrico y que el administrador este actualizando información.			
<b>Efecto Colateral</b>	No aplica.			