

**PONTIFICIA UNIVERSIDAD CATOLICA DEL ECUADOR**

**FACULTAD DE INGENIERIA**

**ESCUELA DE SISTEMAS**

**DESARROLLO E IMPLANTACIÓN DEL SISTEMA DE  
ADMINISTRACIÓN DE CLIENTES Y PRODUCTOS EN LA  
EMPRESA LUBRIMOTOR AUTOMOTRIZ**

**MARIA GABRIELA BEDON MELO**

**JULIO ALBERTO LARRAGA MENA**

**DISERTACION DE GRADO PREVIA LA OBTENCION DEL TITULO DE INGENIERO  
EN SISTEMAS Y COMPUTACION**

**Quito, 2013**

## INDICE

<b>CAPITULO 1 Introducción</b>	<b>3</b>
1.1. Antecedentes	3
1.2. Objetivo	4
1.2.1. Objetivo General	4
1.2.2. Objetivo Especifico	4
1.3. Alcance	5
<b>CAPITULO 2 Fundamento Teórico</b>	<b>6</b>
2.1. Metodología	6
2.1.1. Programación Orientada a Objetos	6
2.1.2. Ciclo de Vida	10
2.2. Arquitectura de Aplicaciones	15
2.2.1. Basadas en Capas	18
<b>CAPITULO 3 Caso de Estudio Empresa Lubrimotor Automotriz</b>	<b>23</b>
3.1. Análisis de la situación actual de la empresa	23
3.1.1. Levantamiento de información productos y clientes	25
3.2. Estándares del sistema	30
3.3. Plan de configuración	36
3.4. Requerimientos	42
3.4.1. Funcionales	49
3.4.2. No Funcionales	52
<b>CAPITULO 4 DESARROLLO DEL CICLO DE VIDA</b>	<b>56</b>
4.1. Ciclo 1	57
4.1.1. Análisis de Requerimientos	59
4.1.2. Diseño del Sistema	75
4.1.3. Implementación y Pruebas	89
4.1.4. Postmortem	110
4.2. Ciclo 2	120
4.1.1. Análisis de Requerimientos	121
4.1.2. Diseño del Sistema	140
4.1.3. Implementación y Pruebas	159
4.1.4. Postmortem	162
<b>CAPITULO 5</b>	<b>167</b>
5.1. Conclusiones	168
5.2. Recomendaciones	169
<b>ANEXOS</b>	<b>170</b>

## **CAPITULO 1**

### **INTRODUCCIÓN**

En el presente capítulo se tratará sobre la exposición de puntos preliminares para la disertación de grado, iniciando por una introducción que brinde una clara idea del propósito, importancia y conocimientos que fueron necesarios para dar forma a éste proyecto, se definirá también el objetivo general con sus respectivos objetivos específicos y finalmente el alcance del proyecto.

#### **1.1. INTRODUCCIÓN**

Esta disertación de grado tiene como propósito principal mostrar cómo se desarrollo el sitio web empezando desde un análisis de los requerimientos del cliente, pasando por diseños hasta llegar al producto final que se encuentra actualmente on-line en [www.lubrimotorautomotriz.com](http://www.lubrimotorautomotriz.com); todo esto para la empresa ecuatoriana Lubrimotor Automotriz comercializadora de maquinaria agrícola. Esta ha acordado la creación de un sistema por sus necesidades y las ventajas que presenta el tener un sitio que automatice ciertos procesos de la empresa. Las diferentes necesidades y requerimientos junto con el deseo de la misma de crecer ha sido el punto de inicio para comenzar la realización de este proyecto, para proveer de un sistema que administre productos, clientes, proveedores y usuarios para al final automatizar el proceso de facturación y brindar a los clientes la posibilidad de realizar cotizaciones de los productos que dispone la empresa.

Una de los principales puntos que los autores desean resaltar como importancia de este proyecto, es la metodología usada, la programación orientada a objetos pues permite acortar la distancia entre el software que se ha desarrollado y la realidad, además gracias a la posibilidad de heredar atributos y métodos de clases, se logra reutilizar código fuente, por lo que los tiempos de desarrollo se optimizan eficientemente. Por otro lado, la metodología para el manejo de las fases del proyecto, fue seleccionada de forma específica y que mantenga una correlación con la naturaleza del proyecto, de generar un aplicativo web, para lo cual se decidió utilizar un ciclo de vida de minicascada, en dos ciclos de vida, cuyas fases son Análisis de Requerimientos, Diseño del Sistema, Construcción y Pruebas, Implantación y Postmortem.

Otra estrategia importante que incluye la presente disertación es el uso del Modelo Vista Controlador (MVC) como arquitectura de software basado en capas que fue utilizado para este proyecto, por su gran facilidad de manejo en el mantenimiento que presenta en el desarrollo web.

Cabe resaltar también que en el levantamiento de requerimientos funcionales y no funcionales se usaron diagramas UML para facilitar la presentación de requerimientos y el diseño. Entre los diagramas más importantes, se tiene a los casos de uso que son el punto de partida para representar de manera objetiva las necesidades del cliente y obtener una próxima fase de diseño clara, real y detallada del sistema.

Con todo esto en mente, el detalle de esta información está distribuida de la siguiente forma: el capítulo de fundamento teórico se encargara de explicar la metodología y la arquitectura de software, la sección del caso de estudio de la empresa dará una amplia idea de la situación actual de Lubrimotor Automotriz y los requerimientos funcionales y no funcionales levantados. En el capítulo de desarrollo del Ciclo Vida veremos la ejecución del proyecto en cada una de las fases establecidas. Y finalmente se obtendrá el aprendizaje de la presente disertación.

## 1.2. OBJETIVOS

### 1.2.1. OBJETIVO GENERAL

Desarrollar e Implementar un Sistema de Administración de clientes, productos, usuarios y proveedores además la automatización de los procesos de facturación y cotización para la empresa Lubrimotor Automotriz en un entorno Web.

### 1.2.2. OBJETIVOS ESPECÍFICOS

- Realizar un levantamiento de información completo que permita una definición correcta de los requerimientos, documentando las necesidades en base a un proceso de desarrollo de software definido.
- Documentar el progreso del proyecto mediante una definición específica de los estándares de calidad y desarrollo para el sistema.
- Hacer uso de la programación orientada a objetos como metodología para el desarrollo del sistema.
- Aplicar el ciclo de vida en cascada para optimizar el desarrollo de este software y recibir oportuna retroalimentación por parte del cliente.
- Generar la documentación adecuada para el correcto mantenimiento del sistema que permita las futuras mejoras del mismo, utilizando documentación detallada y diagramas de UML.

- Establecer un plan de pruebas adecuado para el sistema, que permita detectar errores y realizar la actualización del sistema.
- Implementar un software de alta calidad que satisfaga las exigencias del cliente junto con sus necesidades.

### 1.3. ALCANCE

El proyecto culminará con la entrega del Sistema de Administración de de clientes, productos, usuarios y proveedores además la automatización de los procesos de facturación y cotización para la empresa Lubrimotor Automotriz en un entorno Web, que se ajuste a los requerimientos levantados con el cliente, tras el trabajo de implantación y pruebas realizadas; y conjuntamente con la generación y entrega de la documentación.

## **CAPITULO 2**

### **FUNDAMENTO TEÓRICO**

Se considera indispensable un análisis detallado de la Metodología y Arquitectura de Aplicaciones que se utilizará para este proyecto. Para lo cual, el presente capítulo comenzará rompiendo el paradigma de una estructura de software orientada a objetos; seguido de una amplia explicación sobre cada una de las técnicas, herramientas y procesos que rigen el ciclo de vida que se ha escogido para desarrollar el sistema.

#### **2.1. METODOLOGÍA**

##### **2.1.1. PROGRAMACIÓN ORIENTADA A OBJETOS**

El ser humano al estar rodeado de diversos objetos, mismos que pueden ser categorizados, organizados, combinados, manipulados y hasta creados, ha permitido que estos principios evolucionen y que hoy se apliquen en el momento de crear un sistema o modelar un software. Uno de los trascendentales principios que se debe recalcar es la reusabilidad que una tecnología basada en objetos proporciona, reusar ciertos componentes del programa que facilita al ingeniero de software realizar rápidamente cambios y asegurarse que la calidad del producto que está desarrollando es la mejor.

##### **2.1.1.1. CONCEPTOS BÁSICOS**

Para entender el punto de vista orientado a objetos se pensará en un automóvil que es un objeto que pertenece a la clase vehículo, los vehículos tienen atributos como nombre, color, precio, descripción; mismos que se aplican para cualquier tipo de vehículos autobuses, automóviles, etc. Se definirá entonces que son clases, objetos y atributos.

##### **2.1.1.1.1. CLASES Y OBJETOS**

Los objetos son entidades utilizadas para representar o modelar algo particular del sistema, y son las unidades más básicas utilizadas para crear modelos abstractos. Un objeto se caracteriza únicamente por la conducta en el momento de interactuar con su “mundo”.

Una clase es un concepto orientado a objetos que encapsula los datos y la abstracción de procedimientos necesarios para describir el contenido y el comportamiento de alguna entidad

(objeto) del mundo real. La abstracción de procedimientos se refiere a las operaciones y métodos capaces de usar y manipular los datos.

Un objeto encapsula datos (atributos) y funciones (operaciones, métodos) que manipulan los datos.

#### 2.1.1.1.2. ATRIBUTOS

Las características que se pueden detallar de un objeto son los atributos, y ellos pueden tomar un valor definido por un dominio, la mayoría de veces un dominio está formado por valores específicos. La importancia de asignar estos valores a los atributos radica en que pueden hacer únicos a los objetos.

#### 2.1.1.1.3. OPERACIONES Y MÉTODOS

Como se dijo, un objeto encapsula datos que son los atributos, y a través de las sentencias que forman un algoritmo se procesa estos datos. Estos algoritmos son las operaciones y métodos que hacen que cada módulo tenga sentido.

Los métodos están asociados a un objeto o a una clase de objetos, cuya ejecución se desencadena tras la recepción de un “mensaje”. Desde el punto de vista del comportamiento, es lo que el objeto puede hacer. Un método puede producir un cambio en las propiedades del objeto, o la generación de un “evento” con un nuevo mensaje para otro objeto del sistema.

#### 2.1.1.1.4. MENSAJES

El medio por el cual cada objeto interactúa entre sí son los mensajes, mismos que pueden controlar el comportamiento en el objeto que lo reciba. Un objeto puede ser requerido para realizar una operación mediante el envío de un mensaje que dirá que es lo que debe hacer con los datos que maneja.

#### 2.1.1.2. ROMPIENDO EL PARADIGMA

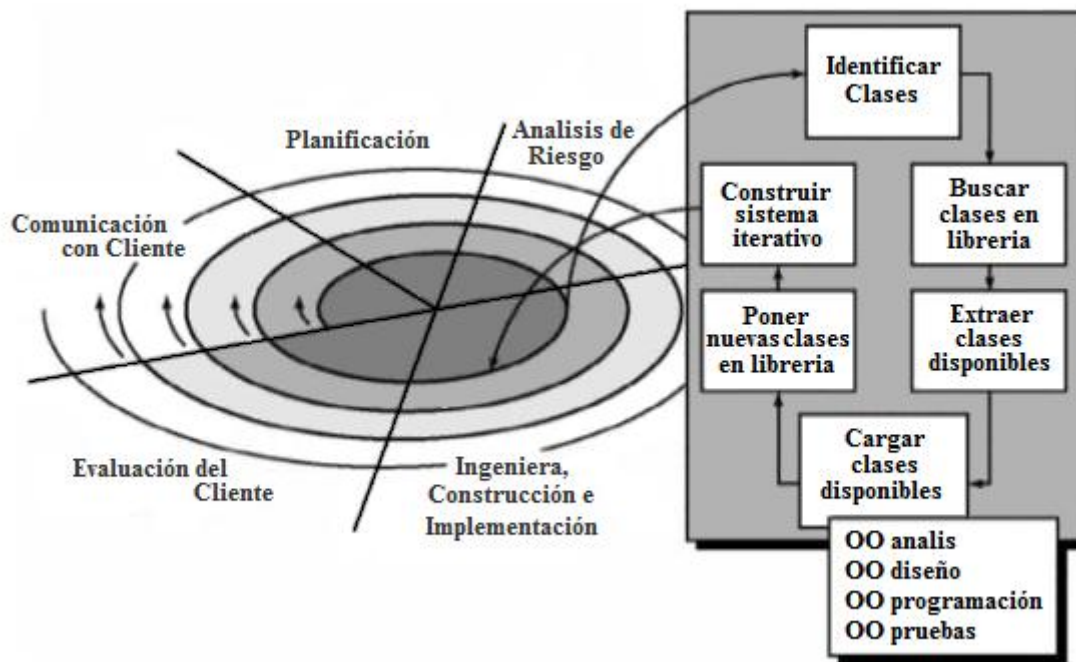


Figura II - 1: Proceso de enfoque Orientado a Objetos

Fuente: PRESSMAN S, (2010) “Ingeniería de Software: Un enfoque practico” Ed.

MacGraw-Hill, Barcelona-España, 6ta Edición.

Traducido por: María Gabriela Bedón, Julio Alberto Lárraga

Marcar la diferencia entre el uso de una tecnología clásica para modelar un sistema de ingeniería de software y usar una estructura orientada a objetos radica en su capacidad de evolucionar con el tiempo. Por lo tanto, un modelo que maneje un proceso evolucionario, junto con un enfoque que favorezca el enlace de componentes (reutilización), es el mejor paradigma para la ingeniería de software orientado a objetos.

El proceso orientado a objetos se mueve alrededor de un espiral que inicia con la comunicación con el cliente. Aquí se define claramente el problema, se planifica y se procede a analizar el riesgo que puede tener el plan. Como se ve en el cuadro de la figura II - 1 se enfatiza en la reutilización, donde las clases se buscan en una librería antes de ser construidas. Cuando no se logra encontrar una clase, esta ingeniería del software aplica Análisis Orientado a Objetos (OOA), Diseño Orientado a Objetos (OOD), Programación Orientada a Objetos (OOP) y Pruebas Orientadas a Objetos (OOT). La nueva clase es puesta dentro de la librería y usada cuando se la necesite. El modelo termina con una evaluación final del cliente.

Por otro lado el paradigma orientado a objetos se centra en el comportamiento y en la estructura de las entidades como unidades completas. El paradigma comprende y apoya los siguientes pilares básicos: abstracción, encapsulación, herencia y polimorfismo.

#### 2.1.1.2.1. ABSTRACCIÓN

El concepto de abstracción se relaciona con la idea de esconder los datos que no son necesarios. La abstracción de datos logra una clara separación entre las propiedades de tipo de datos y los detalles de implementación asociados. Esta separación se obtiene con el fin de que las propiedades de los tipos de datos abstractos sean visibles para la interfaz del usuario y los detalles de implementación están ocultos. De este modo, la abstracción constituye la plataforma básica para la creación de tipos definidos por los objetos. Algunas de las ventajas que el concepto de abstracción integra son:

- Da acceso a los datos que necesita el usuario y oculta los detalles de implementación, dando una mayor seguridad de la aplicación.
- Es posible reemplazar el código sin recopilación, esto hace el proceso más fácil y ahorra tiempo al usuario.
- Ayuda al usuario a dividir la aplicación del proyecto en módulos y probar cada uno de ellos por separado, este enfoque hace que el desarrollo de aplicaciones sea más fácil.

Cuando se considera dar una solución modular a un problema, hay muchos niveles de abstracción. Un alto nivel de abstracción es usar un lenguaje adecuado al ambiente del problema, mientras que un bajo nivel de abstracción se refiere a una orientación basada en procedimientos. Esta abstracción apunta a la secuencia de instrucciones que tiene unas específicas y limitadas funciones.

#### 2.1.1.2.2. ENCAPSULACIÓN

Como ya se ha señalado, las clases orientadas a objetos y los objetos generados por clases, encapsulan datos y las operaciones que manejan los datos en un solo paquete. Esta idea puede traernos innumerables beneficios, como:

- La descripción interna de los datos y procesos están ocultos en el momento de la implementación, esto permite reducir la propagación de efectos secundarios cuando se realizan cambios.
- Consolidar bajo un mismo nombre una entidad de la clase que maneja una estructura de datos y de operaciones que manipulan datos facilita la reutilización de componentes.
- Se simplifican las interfaces entre objetos encapsulados, un objeto que envía un mensaje no tiene que preocuparse de los detalles de las estructuras internas. Por lo tanto, es simple la

conexión y el acoplamiento del sistema tiende a reducirse.

#### 2.1.1.2.3. HERENCIA

Esta definición es bastante intuitiva, y es una principal diferencia entre un sistema convencional y uno orientado a objetos. Se trata entonces de que una subclase (instancia de clases) hereda todos los atributos y operaciones asociadas a una superclase (conjunto de clases). Es decir, cualquier cambio que se realice en una superclase inmediatamente las subclases visualizarán esas modificaciones y el ingeniero de software puede hacer uso de las mismas en cualquier momento. Usar herencia para desarrollar un sistema de software trae los siguientes beneficios:

- Ayuda a que el código pueda ser reutilizado en muchas situaciones, y que el programador pueda crear clases derivadas de la clase base, según sea necesario.
- El concepto anterior de reutilización alcanzado por herencia ahorra el tiempo y el esfuerzo del programador. Dado que el código principal puede ser usado en cualquier momento.
- Aumenta la estructura del programa que se traduce en una mayor fiabilidad.

Este concepto de la herencia nos lleva al concepto de polimorfismo.

#### 2.1.1.2.4. POLIMORFISMO

El polimorfismo es la capacidad de utilizar códigos, las operaciones o los objetos que se comportan de manera diferente en diferentes contextos. El polimorfismo se refiere a la posibilidad de llamar a las diferentes funciones con un solo tipo de llamada, es decir dar diferentes significados a un mismo concepto.

Una vez que una aplicación está escrita utilizando el concepto de polimorfismo presenta ventajas, tales como:

- Puede ampliarse fácilmente, reducir tiempo y esfuerzo en el trabajo y además de hacer más rápido el mantenimiento futuro.
- Apoya en la reutilización del código.
- Ayuda en la consecución de la robustez de las aplicaciones.

#### 2.1.1.3. ANALISIS Y DISEÑO ORIENTADO A OBJETOS

La primera actividad que se debe aplicar dentro de un sistema de ingeniería de software orientado a objetos es el análisis, que tiene como objetivo desarrollar un modelo que describa cada uno de los requerimientos que el cliente definió con anterioridad. Para asegurar una solución efectiva al problema que presenta el cliente, es indispensable contar con herramientas de análisis y diseño que permitan establecer un acuerdo entre los analistas, desarrolladores y el usuario final, para asegurar el entendimiento del problema se ha logrado combinar el análisis y el diseño en un método unificado llamado UML (Unified Modeling Language)

#### 2.1.1.3.1. UNIFIED MODELING LANGUAGE (UML)

Es una herramienta que ayuda a capturar la idea de un sistema y comunicarla posteriormente a quien esté involucrado en su proceso de desarrollo; esto se lleva a cabo mediante un conjunto de símbolos y diagramas que tienen fines distintos en el proceso de desarrollo. En UML, un sistema es representado a través de diferentes vistas, que van describiendo el sistema desde diferentes perspectivas:

- Vista del modelo del usuario (análisis)
- Vista del modelo estructural (análisis)
- Vista del modelo de comportamiento (diseño)
- Vista del modelo de implementación (diseño)
- Vista del modelo de ambiente (diseño)

#### 2.1.1.3.2. PROCESO DEL ANALISIS ORIENTADO A OBJETOS

##### 2.1.1.3.2.1. CASOS DE USO

Los casos de uso modelan el sistema desde el punto de vista del usuario y deben cumplir con los siguientes objetivos:

- Definir la funcionalidad y los requerimientos operacionales del sistema, definiendo un escenario equilibrado para el usuario final y para el ingeniero de software.
- Clarificar como el usuario final y el sistema interactuarían entre sí.
- Preparar las pruebas de validación.

Usando la notación de UML podemos obtener el diagrama caso de uso con algunos niveles de abstracción, se centra en los actores que intervienen en el sistema y los roles que realizan.

#### 2.1.1.3.2.2. MODELO CLASES-RESPONSABILIDAD-COLABORADOR

Nos permite organizar las clases que son relevantes en el sistema, donde las responsabilidades son los atributos y operaciones importantes para cada clase; y los colaboradores son aquellos que proporcionan una clase con la información o acciones necesarias para completar las responsabilidades.

Una vez que los objetos y clases hayan sido identificados mediante el modelo CRC el análisis comienza en focalizar la estructura de los modelos de clases y la jerarquía resultante que genera clases y subclases.

Hay otras definiciones que en este punto es indispensable mencionar, se trata de diferenciar entre realizar un programa que funcione a generar un programa que funcione bien. Para conseguirlo usaremos conceptos de diseño como la Abstracción ya mencionada, Refinamiento, Modularidad, Cohesión y Acoplamiento

##### 2.1.1.3.2.2.1. REFINAMIENTO

Es una estrategia de diseño que tiene como principio que un trabajo de desarrollo de software pase por algunos niveles de mejora para que alcance un nivel satisfactorio y hasta un mayor nivel que ese. Se sustenta en descomponer la estructura a nivel macro e ir puliéndolo en su elaboración con muchos más detalles.

La abstracción y el refinamiento son definiciones complementarias, la abstracción permite al diseñador especificar procedimientos y datos y el refinamiento ayuda a elevar los niveles bajos de detalles como progreso en el diseño.

##### 2.1.1.3.2.2.2. MODULARIDAD

El aporte más importante que hizo el diseño estructurado fue la idea de que, para resolver un problema complejo de desarrollo de software, conviene separarlo en partes más pequeñas, que se puedan diseñar, desarrollar, probar y modificar, de manera sencilla y lo más independientemente posible del resto de la aplicación.

En el diseño orientado a objetos, la modularización esencial se da a nivel de clases, que no son

funciones del sistema, sino entidades del dominio del problema. Por lo que en el análisis y diseño orientados a objetos, no se modulariza la solución, sino primero el problema (en el análisis) y luego, partiendo de esas clases conceptuales, del dominio del problema, se modulariza la solución (diseño). Dentro de una orientación a objetos también se tiene módulos funcionales, que serían los métodos u operaciones de las clases, pero estos tienen una importancia menor respecto del módulo por excelencia, que es la clase. Finalmente, en el diseño orientado a objetos, suele aparecer otro tipo de módulo más, el paquete, importante para agrupar clases en el diseño de aplicaciones.

#### 2.1.1.3.2.2.3. COHESION

La cohesión tiene que ver con que cada módulo del sistema se refiera a un único proceso o entidad. A mayor cohesión, mejor: el módulo en cuestión será más sencillo de diseñar, programar, probar y mantener.

En el diseño orientado a objetos hay tres tipos de módulos: clases, métodos y paquetes. Se puede garantizar una fuerte cohesión disminuyendo al mínimo las responsabilidades de una clase: si una clase tiene muchas responsabilidades probablemente haya que dividirla en dos o más. Y el test a aplicar sería ver si se puede describir a la clase con una oración simple que tenga un único sustantivo en el sujeto. En los paquetes no es usual analizar cohesión, lo crucial es el acoplamiento, que se le vaya a aplicar.

#### 2.1.1.3.2.2.4. ACOPLAMIENTO

El acoplamiento mide el grado de relacionamiento de un módulo con los demás. A menor acoplamiento, mejor: el módulo en cuestión será más sencillo de diseñar, programar, probar y mantener. En el diseño orientado a objetos se complican las cosas con sus tres tipos de módulos.

A los métodos, como pasó con la cohesión, podemos analizarlos con los mismos criterios que a los módulos del diseño estructurado. Una clase, en cambio, tendrá bajo acoplamiento cuando tenga la menor dependencia posible de otras clases. Esta dependencia significa que, si bien puede haber muchas clases que dependen de una, debería haber pocas dependencias hacia otras clases desde una sola. Las dependencias que importan son, de mayor a menor: generalización/herencia, composición, asociación y dependencia débil. Para visualizar estas cuestiones, los diagramas de clases son herramientas fundamentales.

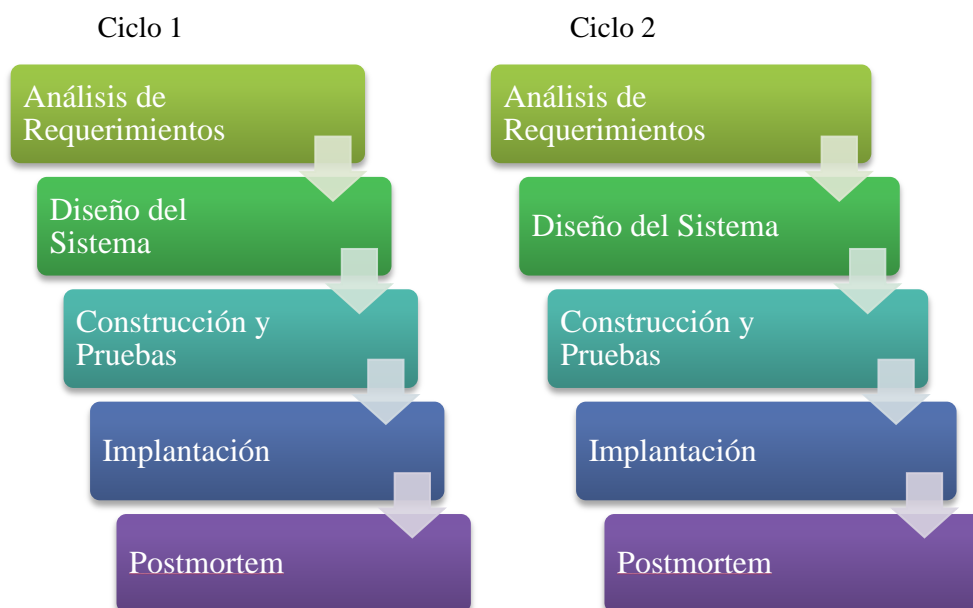
Un paquete debe cumplir con estos mismos requisitos, en el sentido de que debe tener vinculaciones mínimas con otros paquetes. Hay dependencia entre paquetes cuando hay clases de un paquete que dependen de clases de otro paquete, sea por herencia, asociación o simple dependencia débil. En este caso, se puede ayudar del diagrama de paquetes, que debido a que nos muestra dependencias entre conjuntos de clases, sirven para eliminar problemas de acoplamiento.

## 2.1.2. CICLO DE VIDA

La definición de un ciclo de vida para el desarrollo de una página web debe ser realizada de forma específica que mantenga una correlación con la naturaleza del proyecto. Los proyectos de desarrollo de sitios o aplicaciones web se realizan de manera diferente a los desarrollos de aplicaciones de escritorio, puesto que sus características, uso y usuarios son diferentes los unos de los otros. Si bien es posible adaptar uno de los ciclos de vida tradicionales, en este proyecto se ha decidido escoger un ciclo de vida de acuerdo a la naturaleza del proyecto.

### 2.1.2.1. DEFINICIÓN

El ciclo de vida para el desarrollo web según el cual se planificó la realización del proyecto web es un ciclo de vida en minicascada que maneja un flujo secuencial entre las fases, cada etapa tiene un input y un output, y para iniciar una etapa debe haber finalizado la anterior, después de un ciclo inicia otro que contendrá mejoras del ciclo anterior, véase la Figura II - 2.





**Figura II - 2:** Ciclo de Vida para Aplicación Web

**Fuente:** [http://es.wikipedia.org/wiki/Metodolog%C3%ADa\\_de\\_desarrollo\\_de\\_software](http://es.wikipedia.org/wiki/Metodolog%C3%ADa_de_desarrollo_de_software)

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

Este gráfico nos permite definir completamente las cinco fases de las cuales se conforma este ciclo de vida, y las cuales permitirán una correcta codificación y documentación sobre la creación sitio Web.

El proyecto se desarrollará en diferentes fases que, basadas en este ciclo de vida, se definirán con el fin de recopilar toda la información necesaria para la correcta consecución del proyecto. En base a cada una de las fases y a la planificación del proyecto por parte del equipo se hará de la siguiente manera:

#### 2.1.2.1.1. ANALISIS DE REQUERIMIENTOS

Una vez que el cliente ha empezado con la discusión de sus exigencias, es necesario el realizar un análisis preliminar de requerimientos. Los portales web son aplicaciones que, como todo software, deben conservar la documentación de todos los requerimientos de manera clara, a través de los casos de uso, tomando en referencia el diagrama general.

La importancia de esta fase está dada en base a los flujos que la empresa realiza, y que en base a ellos se dará inicio a transformar estos requerimientos en parte tangible para el diseño y construcción del sitio web.

#### 2.1.2.1.2. DISEÑO DEL SISTEMA

Las especificaciones preliminares permiten la definición de cada uno de los elementos que el cliente requiere en el producto. Al ser un desarrollo de un sitio web, la especificación deberá contar con todos los diseños de cada requerimiento o caso de uso, en lenguaje claro, para que los desarrolladores puedan iniciar con la fase de construcción rápidamente. Cabe recalcar que la fase de diseño es pieza clave en cualquier desarrollo de software, debido a que un buen diseño permite que con el tiempo no cambie de fondo, únicamente de forma; con el paso del tiempo se puede

actualizar la manera de visualizar el sistema, pero bajo un prototipo definido y aprobado desde el inicio.

En esta fase es también necesario el diseñar planes de prueba y procedimientos para asegurar la calidad del producto. Estos planes también deben ser puestos a consideración del cliente para su aprobación.

#### 2.1.2.1.3. CONSTRUCCIÓN Y PRUEBAS

Tras la construcción de la especificación, el desarrollo es aceptado tras toda transacción legal que sea necesaria para el comienzo del proyecto. En esto se especifica los diseños que deberán servir como prototipos de la aplicación.

Estos prototipos deberán ser definidos en relación a los requerimientos de los clientes. Algunos desearán ver prototipos funcionales en los cuales puedan manejar la interactividad del sitio, mientras otros aceptarán prototipos gráficos del sitio sin necesidad de probar la navegación.

En este paso pueden existir gran cantidad de cambios y consejos por parte del cliente, y éstos deben, según su importancia, tomarse en cuenta y posteriormente congelarse antes de moverse a la siguiente fase. Las revisiones del trabajo deberán ser realizadas mediante la web para el cliente. Los cambios y comentarios hechos en esta fase deberán ser realizados y tratados con registros de comunicación por e-mail, fax, teléfono o entrevistas.

#### 2.1.2.1.5. IMPLANTACIÓN

La fase de implantación, no es más que poner en ejecución el sistema previamente desarrollado, lo cual implica que se realice un entrenamiento a los usuarios, así como también se realicen un conjunto de actividades desde generar un informe que resuma los recursos de hardware y software que se necesitarán para el correcta cumplimiento del sistema, hasta la entrega oficial de todo el desarrollo del aplicativo.

#### 2.1.2.1.6. POSTMORTEM

La etapa postmortem se refiere a las actividades de reflexión a realizarse cuando el proyecto está acabado. Es importante evaluar el propio desempeño y también la productividad que tuvieron integrantes del equipo de trabajo y su interacción con el cliente.

Es el último paso a realizarse en el ciclo. En esta fase se revisará y evaluará el cumplimiento adecuado de cada uno de los roles en las diferentes etapas.

## 2.2. ARQUITECTURA DE APLICACIONES

La arquitectura que se usará en el proyecto es una de tres capas. En donde cada capa es un proceso separado, bien definido y corre en plataformas separadas. La arquitectura basada en Web transforma la interfaz de búsqueda existente (el explorador de Web), en la interfaz del usuario final. La interfaz del cliente no es requerida para comunicarse con el receptor de los datos. Por lo tanto, esa estructura de los datos puede ser modificada sin cambiar la interfaz del usuario. Por otro lado el código de la capa intermedia puede ser reutilizado por múltiples aplicaciones y el manejo de esta arquitectura sin dudas hace más fácil reemplazar o modificar una capa sin afectar a los módulos restantes.

### 2.2.1. BASADAS EN CAPAS

#### 2.2.1.1. ARQUITECTURA MODELO-VISTA-CONTROLADOR (MVC)

El patrón de diseño MVC es un patrón utilizado generalmente por aplicaciones que necesitan poseer múltiples vistas de un mismo elemento y con una misma funcionalidad. Este modelo se maneja de manera que permite una correcta abstracción de los objetos, en diferentes categorías dependiendo de su funcionalidad:

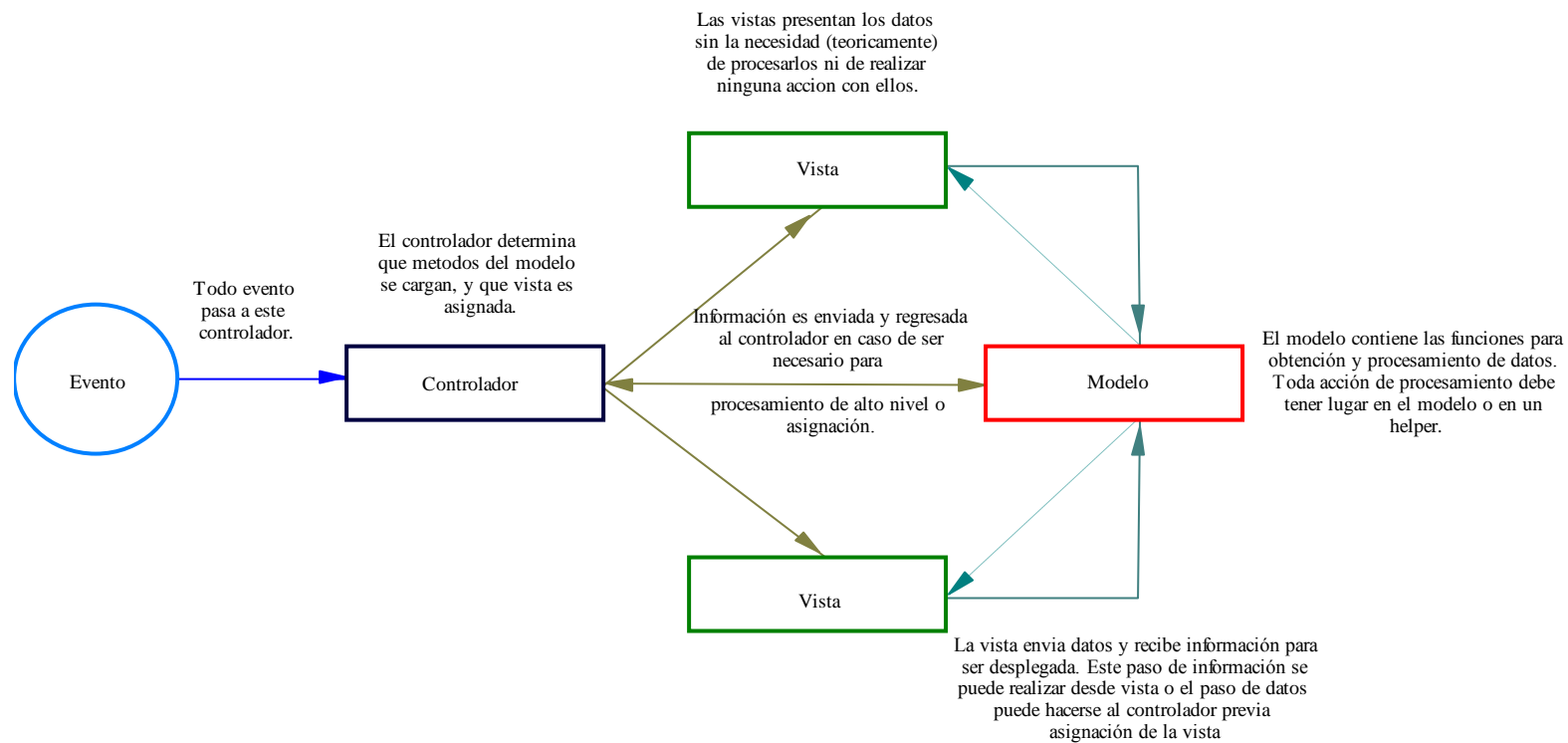
**Modelo:** permiten el manejo y mantenimiento de la data. Poseen funciones utilizadas dentro de los elementos para el correcto manejo de los mismos. Maneja el comportamiento y la data de la aplicación, respondiendo a requerimientos de información sobre su estado (generalmente hechas por la vista) y a instrucciones para cambiar este estado (generalmente hechas por el controlador).

**Vista:** permiten desplegar porciones de datos gracias a su manejo del modelo.

**Controlador:** maneja los eventos que afectan al modelo y/o a la vista.

Esta separación entre las diferentes instancias de trabajo permite la separación de código y que múltiples vistas actúen con un mismo controlador, permitiendo la creación y modificación de vistas para usuarios específicos, sin que signifique modificar toda la funcionalidad de un área del sitio.

En la figura II - 3 se explica de manera clara el proceso de funcionamiento del patrón modelo – vista – controlador.



**Figura II - 3:** Modelo- Vista – Controlador

**Fuente:** <http://www.enode.com/x/markup/tutorial/mvc.html>

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

### 2.2.1.2. VENTAJAS Y DESVENTAJAS

La utilización de un patrón MVC posee varias ventajas y desventajas, las cuales se detallarán a continuación:

#### 2.2.1.2.1. VENTAJAS

**Soporta múltiples vistas:** la principal características de un patrón MVC es que soporta varias vistas para un mismo componente. Al no existir una dependencia rígida entre el modelo y la vista, la interfaz puede mostrar diferentes vistas de la misma data al mismo tiempo.

**Mantenimiento y actualización:** las interfaces de usuario son aquellas que generalmente cambian (a diferencia de la funcionalidad, que se mantiene igual). Los cambios en colores, posiciones, etc., son muy requeridos, por lo que, al no depender el modelo de la vista, el añadir nuevas vistas se puede hacer sin afectar el modelo y el funcionamiento del sitio.

**Pruebas:** el patrón MVC mejora las capacidades de prueba, al no tener el código de presentación (display) y su funcionalidad en el mismo lugar. Además reduce el número de pruebas de interfaz, al tenerlas completamente separadas unas de otras.

#### 2.2.1.2.1. DESVENTAJAS

**Complejidad:** si bien mejora la abstracción del código, el modelo MVC crea nuevos niveles de abstracción y requiere de experiencia para direccionarse dentro de su código. También aumenta la complejidad del código utilizado para interfaz, puesto que utiliza llamados para obtención de data del modelo.

**Costo de cambios frecuentes:** si bien el aumento o cambios en vistas es relativamente simple, el realizar frecuentes cambios a los modelos puede afectar gravemente los sitios puesto que el cambio en un modelo afecta a todas las vistas relacionadas a éste. Además, los cambios a modelo pueden crear problemas de interfaz, causando cambios en la información y como se va a desplegar.

### 2.3. CUADRO DE ENTREGABLES

No.	Etapas	Tipo	Nombre	Ubicación	Ciclo Iteración	Descripción
1	Levantamiento de Información	Documento	Análisis de la situación actual de la empresa	Capitulo 3 (3.1.)	NA	Este documento contiene la información actual de la empresa y los primeros requerimientos de la empresa
2	Línea Base	Formulario	Formulario de cambios en Línea base	Capitulo 3 (3.3.1.)	NA	Este formulario será completado cuando se genere algún cambio en la línea base.
3	Análisis de Requerimientos	Documento	Análisis de requerimientos	Capitulo 4 (4.1.1.)	1	Este documento muestra los casos de uso que representan las necesidades del cliente
4	Diseño del Sistema	Documento	Diseño del Sistema	Capitulo 4 (4.1.2.)	1	Contiene todos los diseños generados para este software.
5	Diseño del Sistema	Documento	Plan de Pruebas	Capitulo 4 (4.1.2.3.)	1	Contiene el plan de pruebas que se realizará en el sistema.
6	Construcción y Pruebas	Documento	Construcción y Pruebas	Capitulo 4 (4.1.3.)	1	Contiene la parte técnica del Sistema y la aplicación de las pruebas
7	Implantación	Documento	Plan de Implantación	Capitulo (4.1.4.1.)	1	Documento que contiene las responsabilidades y cronogramas de entrenamiento de la implantación del sistema
8	Implantación	Formulario	Formato de Reporte de Fallos	Capitulo 4(4.1.4.1.3.3)	1	Formulario para reportar fallas del sistema
9	Implantación	Documento	Resultados de las Pruebas	Capitulo 4(4.1.4.2.1.)	1	Contiene el resultado de las pruebas junto con la versión final de las mismas
10	Implantación	Documento	Acta de Entrega y Recepción	Anexos	1	Documento de aceptación del software entre las partes
11	Implantación	Documento	Manual Técnico	Anexos	1	El manual de configuración y de las piezas de software para futuras cambios, para administrador seleccionado previamente
12	Implantación	Documento	Manual de Usuarios	Anexos	1	EL manual de entrenamiento para los usuarios generales
13	Análisis de Requerimientos	Documento	Análisis de requerimientos	Capitulo 4 (4.2.1.)	2	Este documento muestra los casos de uso que representan las necesidades del cliente
14	Diseño del Sistema	Documento	Diseño del Sistema	Capitulo 4 (4.2.2.)	2	Contiene todos los diseños generados para este

						software.
15	Diseño del Sistema	Documento	Plan de Pruebas	Capitulo 4 (4.2.2.3.)	2	Contiene el plan de pruebas que se realizará en el sistema.
16	Construcción y Pruebas	Documento	Construcción y Pruebas	Capitulo 4 (4.2.3.)	2	Contiene la parte técnica del Sistema y la aplicación de las pruebas
17	Implantación	Documento	Plan de Implantación	Capitulo (4.2.4.1.)	2	Documento que contiene las responsabilidades y cronogramas de entrenamiento de la implantación del sistema
18	Implantación	Formulario	Formato de Reporte de Fallos	Capitulo 4(4.2.4.1.3.3)	2	Formulario para reportar fallas del sistema
19	Implantación	Documento	Resultados de las Pruebas	Capitulo 4(4.2.4.2.1.)	2	Contiene el resultado de las pruebas junto con la versión final de las mismas
20	Implantación	Documento	Manual Técnico	Anexos	2	El manual de configuración y de las piezas de software para futuras cambios, para administrador seleccionado previamente
21	Implantación	Documento	Manual de Usuarios	Anexos	2	EL manual de entrenamiento para los usuarios generales

**Tabla II - 3:** Cuadro de Entregables

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

## **CAPITULO 3**

### **CASO DE ESTUDIO EMPRESA LUBRIMOTOR AUTOMOTRIZ**

El presente capítulo trata de exponer el caso de estudio de la empresa Lubrimotor Automotriz, además de levantar toda la información del cliente para analizarla y buscar la mejor solución para obtener un producto que se ajuste a sus necesidades. A través de historias de usuarios empezaremos a mapear la situación actual de la empresa y servirán para plantear los requerimientos funcionales y no funcionales del sistema. Definiremos también una línea base para cambios posteriores en el producto.

#### **3.1. ANALISIS DE LA SITUACIÓN ACTUAL DE LA EMPRESA**

El presente proyecto de desarrollo e implementación del sistema de administración de clientes y productos se construirá para la empresa Lubrimotor Automotriz .

La empresa Lubrimotor Automotriz, es la comercializadora de maquinaria agrícola más importante de la provincia de Imbabura. Tiene más de 30 líneas de productos con más de 100 ítems para el sector industrial, ferretero, automotriz y principalmente al sector agrario de la provincia. Algunos productos van desde maquinaria forestal, agrícola y de jardinería; hasta equipamientos individuales y accesorios en general; todos estos productos amparados bajo las marcas internacionales más importantes como Stihl, Husqvarna, TimberTech, Toyama, Fema, entre otras.

Su menú de productos incluye el mantenimiento predictivo de maquinarias, servicios de protección y monitoreo, herramientas de corte, compresores, motosierras, motoguadañas, cortadores de césped, generadores, motobombas, etc. En todas estas líneas, el manejo del servicio al cliente post venta es crucial. Así también esta organización interviene en lo referente a la responsabilidad social y ambiental. Además, su meta no es vender productos sino entregar soluciones eficientes. Su carta de presentación son sus productos y su atención combinada con una acertada asesoría.

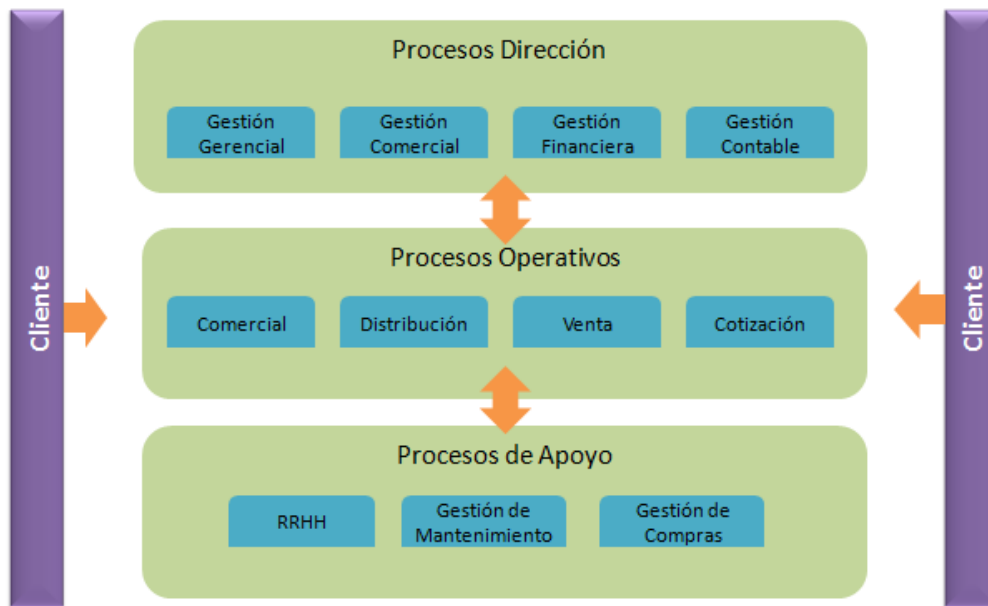
Lubrimotor Automotriz se diferencia de su competencia por ser una empresa que busca entregar paquetes completos. Es decir, el cliente no solo puede encontrar el producto principal sino también sus complementos, Además, 25 años de historia avalan a la organización para afrontar las necesidades del mercado y las garantías correspondientes; de forma que el cliente sienta seguridad en los productos que adquiere. Su filosofía de negocio consiste en tratar a sus clientes como a socios. Por ello, el interés primordial de la empresa es que los individuos crezcan en cada uno de sus espacios con los mejores costos y los niveles de calidad acordes a ese desarrollo, permitiendo así a su gente y a sus clientes el desarrollo de su máximo potencial.

Actualmente la empresa cuenta con su matriz y una sucursal en la ciudad de Ibarra, además de otra sucursal en la ciudad de Otavalo. Uno de los objetivos de Lubrimotor Automotriz es expandirse en un futuro en todo el norte del país. Para lograrlo, el Gerente General de Lubrimotor Automotriz, Germán Bedón manifiesta la necesidad de organizar sus productos y proveedores a través de una administración web, el cual permita a los clientes internos de la empresa, encontrar un producto y su proveedor en cualquier momento y en cualquier sucursal.

Esta necesidad se focaliza en la promoción del trabajo, los productos y sus proveedores, como una forma de permitir al recurso humano de la empresa, la búsqueda de los diferentes productos acorde a las necesidades de los clientes, y la interacción con los proveedores de los mismos. Esto permitirá proveer una mejor información a los clientes sobre los diferentes productos, mejorar la promoción de los mismos, puesto que el vendedor podrá con propiedad manejar la venta, y personalizar la atención a su cliente.

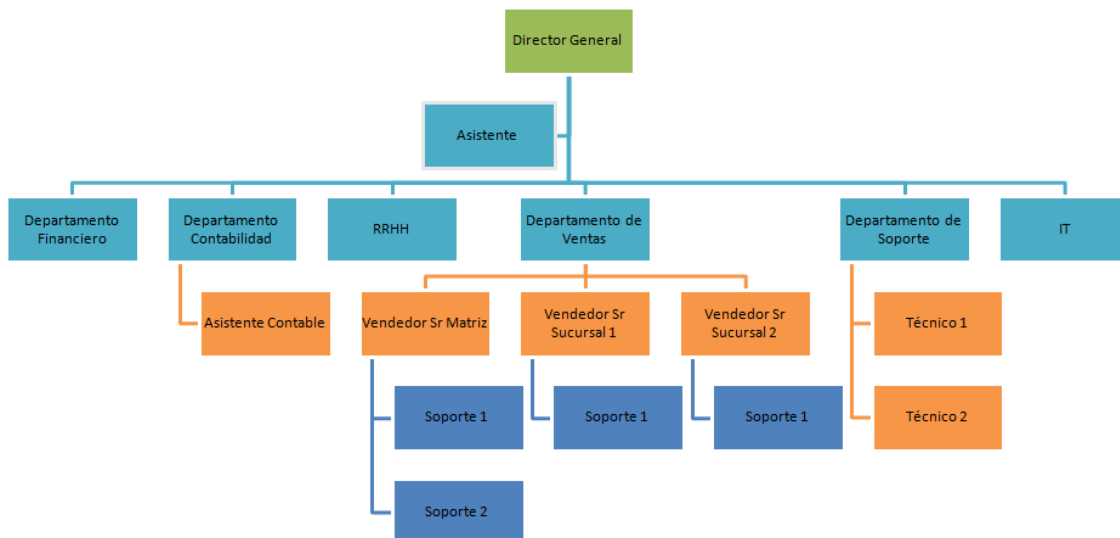
La empresa necesita, en base a la información del usuario, manejar un registro de compras del mismo, siendo indispensable mantener un historial de compras del cliente para, con este fin, poder proveer una calificación dependiendo de sus compras. Así mismo, es necesaria una administración de contenidos y de publicidad para el manejo del sitio, permitiéndole al usuario el publicitar sus productos y gestionar la información de su sitio de manera simple y organizada.

MAPA DE PROCESOS



**Figura III - 1.** Descripción de la Información recopilada del Cliente  
**Desarrollado por:** María Gabriela Bedón, Julio Alberto Lárraga

ORGANIGRAMA



**Figura III - 2.** Descripción de la Información recopilada del Cliente  
**Desarrollado por:** María Gabriela Bedón, Julio Alberto Lárraga

### 3.1.1. LEVANTAMIENTO DE INFORMACIÓN

Para poder definir un modelo correcto en la implementación de las diferentes administraciones para el manejo de usuarios, clientes, productos y proveedores se realizaron en base a entrevistas con el cliente, en este caso con el Sr. Germán Bedón, quien en varias conversaciones ha manifestado los requerimientos de información necesarios.

La información para poder definir el modelo conceptual, el modelo de base de datos, de los diagramas y la funcionalidad del sistema se determinará en los primeros acercamientos con el cliente donde se pueda delinear las principales necesidades de la empresa. Si bien es cierto, el proceso de desarrollo del proyecto está sujeto a variaciones directas o indirectas, ya sean por tiempo o ajustes en las necesidades de los diferentes actores que afecten el sistema, la información inicial es pieza clave para comenzar una correcta implementación.

La información a recopilarse se ha definido en tres partes: productos, proveedores y clientes. Estos tres campos de acción están en base a que son las tres áreas en que la implementación del sistema se llevará a cabo.

El manejo de productos es la base en la implementación, puesto que el portal web se enfatizará en el buscador de productos, mediante el cual el usuario del sistema pueda obtener información sobre el estado del producto, hasta realizar la facturación. En segunda instancia, se manejará a los clientes, se mantendrá un registro detallado y constante de los clientes y de las compras que cada uno está realizando. Finalmente, la información de los proveedores y los productos que provee cada uno, es también de vital importancia, y está ligada a los clientes.

Existe también información sobre el proceso de facturación, el cual integra clientes productos y proveedores. Esta implementación se realizará al final, pues es un proceso que utiliza las tres administraciones ya mencionadas.

La información recopilada para el manejo de los diferentes productos puede resumirse en la Tabla III - 1. separada con la información necesaria en base a su campo de acción.

	Dato	Tipo	Descripción
<b>Productos</b>	Nombre	Texto	Nombre del producto.
	Proveedor	Texto	Nombre del proveedor del producto.
	Imagen del producto	Foto	Foto del producto (no es necesario pero es recomendado tenerla).
	Tipo	Texto	Tipo del producto, puede pertenecer a una de varias características definidas por el usuario.
	Marca	Texto	Marca del producto, definido por el proveedor del mismo.
	Modelo	Texto o Número	Modelo del producto, definido por el proveedor del mismo.
	Características	Texto	Texto con las características del producto.
	Precio	Número	Precio, o referencial del producto. El precio real lo dará el proveedor del producto al solicitar cotización.
	Creador	Texto	Nombre del creador del producto.
	Fecha de creación	Fecha	Fecha de creación del producto.
	Fecha de modificación	Fecha	Fecha de la última modificación del producto.
	Calificación	Valor	Calificación del producto dada por el usuario y/o administrador del sistema.
<b>Proveedores</b>	Nombre	Texto	Nombre de la empresa proveedora de productos o servicios.
	Descripción	Texto	Descripción de la empresa proveedora.
	Correo electrónico	Email	Dirección de correo electrónico del proveedor a la que se enviarán las cotizaciones.
	Teléfonos	Número (s)	Número telefónico del proveedor.

		telefónico	
	RUC	RUC	Identificador legal que poseen las empresas legalmente constituidas.
	Dirección	Texto	Dirección de ubicación de la empresa (o matriz de la misma).
	Calificación	Identificador	Campo de definición para que el administrador del sistema califique a los proveedores en base a criterios definidos como su cumplimiento o respuestas a los usuarios, y en las relaciones con los clientes.
	Logo del Proveedor	Foto	Logo del proveedor (No es necesario pero es recomendado tenerlo).
<b>Clientes</b>	Nombres	Texto	Nombres y Apellidos del cliente.
	Tipo de Cliente	Teto	Campo que contendrá el tipo de cliente si es pequeño, grande.
	Número de Cédula o RUC	Número	El número único de cédula de cada cliente.
	Correo Electrónico	Email	Dirección de correo electrónico. Se utilizará esta para las cotizaciones, y para envío de cualquier información al cliente.
	Teléfonos	Número (s) telefónico	Número telefónico del cliente. Es necesario para poder realizar seguimientos de las acciones tomadas por el mismo en sus cotizaciones o compras.
	Dirección	Texto	Dirección del cliente, para poder contactarlo o enviarle información de ser necesario.
	Calificación	Identificador	Campo de definición para que el

			administrador del sistema califique a los clientes en base a criterios definidos en las relaciones a los proveedores.
<b>Facturación</b>	Número de Cedula o RUC del Cliente	Texto	Para llamar los datos del cliente para la factura
	Producto	Texto	Nombre del producto que el cliente adquirirá desea.
	Fecha de Creación	Fecha	Fecha en que se realizó la compra por parte del cliente.
	Estado	Identificador	Campo de definición si la compra se encuentra activa o inactiva.

**Tabla III - 1.** Descripción de la Información recopilada del Cliente

**Fuente:** Reuniones con el cliente.

**Desarrollado por:** María Gabriela Bedón, Julio Alberto Lárraga

Cabe mencionar que la Tabla III - 1, no representa un modelo de base de datos ni la forma en que la información será guardada, es simplemente una recopilación de las necesidades de información requeridas por el cliente. La estructura de la base de datos y la manera de ser almacenada la información la determinarán los desarrolladores.

### 3.1.2. ANALISIS DEL PROYECTO

Después de tener claro las necesidades del usuario, se ve la necesidad de delinear cuál es el propósito y descripción del mismo. Se exhibirá también las políticas bajo las cuales el desarrollo del sistema dará lugar, preparando el territorio para el levantamiento de la información.

Vale mencionar que este análisis se lo ha basado en los requerimientos del cliente, para lo cual fue muy necesario documentar las necesidades actuales de la empresa, y de esta forma enfocar este proyecto a un excelente producto terminado.

### 3.1.2.1. PROPÓSITO

El propósito del proyecto es proveer a la empresa Lubrimotor Automotriz de una administración de clientes, productos y proveedores en un entorno web, además de los procesos de facturación y cotización

Este sitio debe en su finalización, permitir que los empleados de la empresa que diariamente manejan productos, puedan realizar búsquedas inmediatas de lo que el cliente solicita o desea. Sin embargo, debe existir restricción para la información que se despliegue, siendo el total de la información desplegada solamente para los usuarios registrados.

Gracias a este registro el portal debe proveer de una base de usuarios los cuales contribuirán con el mercadeo de la empresa, y tras la respectiva confirmación de tanto usuarios como empresas, permitirán la interacción entre éstas para la compra/venta de productos.

El sitio web a desarrollarse debe mantener características de manejo en un nivel simple de tal manera que los usuarios puedan interactuar con el sistema sin dificultad. Todo el sitio debe ser fácilmente administrable de manera que requiera del menor conocimiento técnico del lenguaje y herramientas posibles. Esto incluye los productos y proveedores, el manejo de usuarios, las cotizaciones y el proceso de facturación.

Haciendo una síntesis del propósito, el proyecto proporcionara un aplicativo web que se ajuste a los estándares definidos en este documento, abarcando con los requerimientos del cliente de manera eficiente y segura, cumpliendo con las capacidades de administración y actualización necesarias para el mantenimiento de un sitio en óptimas condiciones.

### 3.1.2.2 DESCRIPCIÓN

El sitio web a ser desarrollado es un sistema de búsqueda de los diferentes productos que la organización ofrece a sus clientes, al igual que los proveedores de los mismos. El fin de la página web es proveer a los usuarios con un lugar centralizado en el cual puedan buscar sus productos y ponerse en contacto fácilmente con los proveedores de los productos que requieran.

El proyecto se desarrollará en base a los diferentes requerimientos presentados por el cliente para cumplir con estos fines. De esta manera requerirá tanto un diseño y manejo del sitio en front-end al igual que el manejo de administrador (back-end).

Se desarrollará el sistema en fases, de tal manera que el proyecto contará con una descripción de los requerimientos presentados por el usuario, así como con las restricciones y limitantes que el proyecto presentará. A partir de éstos, se determinarán también diseños y gráficos para determinar el avance y directrices del proyecto.

Es necesario, de igual manera tomar en cuenta que todo el trabajo será realizado guiándose en base al ciclo de vida y su aplicación definidas en capítulos anteriores, siendo esta definición la guía de la documentación presente.

### 3.1.2.3. POLÍTICAS DE DESARROLLO

Poniendo en consideración el factor cambio en la estructura del sistema, se debe tener clara la idea de establecer los cimientos del proyecto, con información que sea constante, que no se modifique y sobre la cual se pueda proceder con la implantación. De igual manera, todos los cambios en requerimientos de información a recopilar deberán ser discutidos y correctamente documentados, puesto que pueden requerir modificación de diseños, de base de datos y, en casos extremos, llevar a cambios mayores en el funcionamiento del sistema.

Para mantener una armoniosa organización en el desarrollo de software, es necesario definir correctamente las políticas bajo las cuales el sistema tendrá lugar. Se debe aclarar que estas políticas no se manejan como requerimientos o restricciones, pero si a la forma de trabajo, entrega de los ciclos y fases del proyecto y mejorar el desempeño de los ingenieros de software.

Se debe tener presente algunos inconvenientes que ocurren en el desarrollo de proyectos como pérdida de versiones o errores imprevistos, donde el trabajo en equipo es fundamental y para ello todos los integrantes deben hablar el mismo idioma. Por lo tanto, se han planteado un grupo de políticas para el desarrollo, las cuales se encuentran detalladas a continuación:

### **Para documentos**

- Al generar un documento debe tener relación con el temario definido, además se agregará en él, cada nuevo ítem en caso de ser creado.
- Todos los archivos, codificación, diseños, etc., en cualquier extensión, se almacenarán en una carpeta creada en el servidor online Dropbox. Esto con el fin de que no existan problemas de versiones entre los integrantes del grupo. De igual manera se evita la pérdida de documentos o código en caso de que algo ocurra con una de las computadoras de desarrollo.
- Archivos esenciales y con información sobre el proyecto se encuentran como documentos online en googleDocs. Los miembros del equipo tienen acceso a estos documentos, por lo que pueden ingresar y modificarlos cuando sea pertinente.
- Existe un archivo dentro de googleDocs para las fuentes bibliográficas y electrónicas, que deberá ser actualizado cada vez que una fuente es utilizada.
- Los miembros del equipo deberán tener una copia de todos los documentos actualizados, y deberán siempre trabajar sobre las últimas versiones de los mismos.
- Todo el equipo debe tener en cuenta toda la información redactada y, de ser necesario, presentar cambios se debe informar previamente.

### Para márgenes y espacios

- Márgenes superior e izquierdo: 4 cm.
- Márgenes inferior y derecho: 2.5 cm.
- Texto a doble espacio, escrito a una sola cara de la hoja.
- Pies de página y citas textuales a espacio simple.

### Para fuentes

- Las fuentes que se manejarán son de tipo Times New Roman, con un tamaño 11, color negro. Para citas importantes, se utilizarán negritas, cursivas o subrayado dependiendo del caso.

### Para títulos

- Para los títulos, se ha decidido usar la fuente Times New Roman en tamaño 14, en negrilla, color negro, para los subniveles de títulos se usara Times New Roman tamaño 11. Los títulos menores a partir de este nivel se sucederán como subtítulos, en negrilla o subrayados, con el mismo tamaño que el texto.

#### Tamaño de la Hoja

- Los estándares definidos para el tamaño de la hoja determinan que este tamaño debe regirse a la norma 75 definida por el Instituto Ecuatoriano de Normalización (INEN), siendo el papel escogido el de tamaño A4, con dimensiones de 29.7 x 21 cm.

#### **Para la comunicación**

- Es imprescindible una comunicación constante entre los miembros del equipo, es necesario enviar una notificación por correo electrónico cada vez que se realice un cambio en algún archivo.
- Toda comunicación debe ser registrada, para no perder de ninguna manera los requerimientos o necesidades dentro de la codificación.
- Los integrantes del equipo deben estar presentes en todas las reuniones que se mantengan con el cliente para la entrega de información en relación a los requerimientos.

#### **Para la codificación**

- Se deberá mantener los estándares definidos de acuerdo a los patrones de trabajo establecidos. De igual manera, deberán ajustarse al lenguaje de los diseños para evitar conflictos.
- La documentación deberá ir correctamente comentada y documentada, para evitar problemas dentro del código. Se incluirá el autor de dicha versión en el código, siendo este cualquiera de los dos miembros del equipo.

#### **Para el diseño**

Los diseños que se generarán en el transcurso de este proyecto tienen que ver con todos los diagramas y las representaciones gráficas que se irán agregando con la final de ampliar el contenido o serán mencionados como anexos. Cuando se habla de diagramas se refiere a todos los requerimientos, restricciones o funcionalidades del cliente. Se debe entonces, tener en cuenta lo siguiente:

- Fondo general de los diagramas: blanco.
- Borde de los diagramas: verde oscuro
- Fuente de las letras pueden variar manteniéndose similares en todos los diseños de un mismo tipo (Todos los diagramas de casos de uso deben poseer la misma fuente y tamaño de letra). La letra debe ser clara y poderse ver bien. De preferencia se utilizará Times New Roman tamaño 10 px color negro.
- Fondo de los elementos en los diagramas: de preferencia blanco, con borde naranja. Es posible utilizar los fondos predeterminados por la herramienta. Todos los elementos similares de los diseños deben manejarse de igual manera.

### 3.2. ESPECIFICACIÓN DE REQUERIMENTOS

La especificación de requerimientos es la fase que permitirá definir de manera correcta las necesidades que el cliente necesita sobre el sistema. Esta fase iniciará con algunas historias de usuarios para luego establecer las administraciones e implementaciones necesarias para el sistema, que se determinarán como requerimientos funcionales, requerimientos no funcionales otros requerimientos y restricciones de la implementación con el fin de proveer una especificación de requerimientos completa.

#### 3.2.1. HISTORIAS DE USUARIOS

##### Cliente y Empleado

Un cliente acude a Lubrimotor Automotriz a adquirir una máquina, el empleado orienta al cliente en cuanto a marca, modelos, precios y beneficios de cada uno. Si el producto está en stock se lo indica y hace una demostración de ser necesario. En caso de que el cliente desee

cerrar la compra, el empleado manualmente revisa inventario y hace el despacho de la máquina, caso contrario debe contactar a un proveedor para el abastecimiento y agendar con el cliente el día en que la máquina será despachada a su domicilio.

Para la facturación el cliente brinda toda su información y el empleado la digita en una hoja de cálculo de Excel y la imprime.

### Proveedor

El proveedor recibe las órdenes de compra y libera la maquinaria a Lubrimotor Automotriz en un tiempo determinado dependiendo del stock o realiza una importación de ser necesario.

Lubrimotor Automotriz ingresa estas órdenes de compra en una hoja de cálculo de Excel, para facturarlas a fin de mes.

### 3.2.2. REQUERIMIENTOS FUNCIONALES

Los requerimientos funcionales son aquellos que engloban las administraciones a ser generadas para el correcto funcionamiento del sistema. Estas administraciones son las necesidades que posee el cliente sobre el sitio, definidas como requerimientos de productos, proveedores, usuarios del sistema y el proceso de facturación. Cuando se hable de una administración se referirá a realizar un CRUD (acrónimo de Crear, Obtener, Actualizar y Borrar) que ayudará a generar los siguientes requerimientos funcionales:

F1: El sistema se encargará de la administración de usuarios

- Cada usuario tendrá su propio perfil dependiendo del alcance del mismo.
- Cada usuario accederá al sistema con un nombre y una contraseña.

F2: El sistema se encargará de la administración de clientes

- El sistema será capaz de manejar clientes tanto locales como a nivel nacional.
- El sistema proporcionará un historial de compras en cada cliente para ser categorizados.

F3: El sistema se encargará de la administración de productos

- El sistema manejará el estado de disponibilidad de cada producto.

- El sistema mantendrá informado al usuario.

F4: El sistema se encargará de la administración de proveedores

- El sistema podrá enviar la orden de compra al proveedor automáticamente.
- El sistema manejará una bitácora de estados de un pedido.

F5: El sistema se encargará del proceso de facturación

- El sistema será manejado en múltiples sucursales, esto implica que al final del día este toda la información disponible y consolidada.
- El sistema generará un reporte de la facturación realizada.

### 3.2.3. REQUERIMIENTOS NO FUNCIONALES

Los requerimientos no funcionales hacen relación a las características del sistema que aplican de manera general como un todo, más que a rasgos particulares del mismo. Estos requerimientos son adicionales a los requerimientos funcionales que debe cumplir el sistema, como es definir las herramientas que delimitarán el alcance del sistema además éstos requerimientos corresponden a aspectos tales como la disponibilidad, mantenibilidad, flexibilidad, seguridad, facilidad de uso, etc., los cuales se describen en las siguientes líneas.

#### Desempeño

- Garantizar la confiabilidad, la seguridad y el desempeño del sistema informático a los diferentes usuarios a nivel local y nacional. En este sentido la información almacenada podrá ser consultada y actualizada permanente y simultáneamente, sin que se afecte el tiempo de respuesta.
- El sistema debe estar en capacidad de dar respuesta al acceso de todos los usuarios con tiempo de respuesta aceptable y uniforme, en períodos de alta, media y baja demanda de uso del sistema.

#### Disponibilidad:

- Estar disponible 100% o muy cercano a esta disponibilidad durante el horario hábil laboral de la empresa Lubrimotor Automotriz a nivel local y nacional (Ejemplo: de lunes a viernes de 9:00 a.m. a 7:00 p.m., con excepción de los días festivos).
- Operar de la misma manera para todos los niveles de la estructura jerárquica de la empresa.

#### Escalabilidad:

- El sistema debe ser construido sobre la base de un desarrollo evolutivo e incremental, de manera tal que nuevas funcionalidades y requerimientos relacionados puedan ser incorporados afectando el código existente de la menor manera posible; para ello deben incorporarse aspectos de reutilización de componentes.
- El sistema debe estar en capacidad de permitir en el futuro el desarrollo de nuevas funcionalidades, modificar o eliminar funcionalidades después de su construcción y puesta en marcha inicial.

#### Facilidad de Uso e Ingreso de Información:

- El sistema no debe permitir el cierre de una operación hasta que todos sus procesos, subprocesos y tareas relacionados, hayan sido terminados y cerrados satisfactoriamente.
- El ingreso de información al sistema debe diseñarse con transacciones que permitan el ingreso de los datos de forma parcial; es decir, que el tamaño de las páginas de registro (o formularios) de información sean adecuadas de acuerdo con la estabilidad de la red.
- El sistema debe presentar mensajes de error que permitan al usuario identificar el tipo de error y comunicarse con el administrador del sistema.

#### Facilidad para las Pruebas:

- El sistema debe contar con facilidades para la identificación de la localización de los errores durante la etapa de pruebas y de operación posterior.

#### Flexibilidad:

- El sistema debe ser diseñado y construido con los mayores niveles de flexibilidad en cuanto a la parametrización de los tipos de datos, de tal manera que la administración del sistema sea realizada por un administrador funcional del sistema.

#### Mantenibilidad:

- Toda el sistema deberá estar complemente documentado, cada uno de los componentes de software que forman parte de la solución propuesta deberán estar debidamente documentados tanto en el código fuente como en los manuales de administración y de usuario.
- El sistema debe estar en capacidad de permitir en el futuro su fácil mantenimiento con respecto a los posibles errores que se puedan presentar durante la operación del sistema.

#### Operatividad:

- El sistema debe ser de fácil operación por los empleados de Lubrimotor Automotriz, que demande un bajo nivel de soporte de los usuarios del sistema.

#### Seguridad:

- El acceso al Sistema debe estar restringido por el uso de claves asignadas a cada uno de los usuarios. Sólo podrán ingresar al Sistema las personas que estén registradas, estos usuarios serán clasificados en varios tipos con acceso a las opciones de trabajo definidas para cada uno.
- El control de acceso implementado debe permitir asignar los perfiles para cada uno de los roles identificados.

- Respecto a la confidencialidad, el sistema debe estar en capacidad de rechazar accesos o modificaciones indebidos (no autorizados) a la información y proveer los servicios requeridos por los usuarios legítimos del sistema.
- El sistema deberá contar con mecanismos que permitan el registro de actividades con identificación de los usuarios que los realizaron.
- El sistema debe contar con pistas de auditoría de las actividades que se realizan sobre el sistema con niveles razonables para su reconstrucción e identificación de los hechos.

### Validación de Información

- El sistema debe validar automáticamente la información contenida en los formularios de ingreso. En el proceso de validación de la información, se deben tener en cuenta aspectos tales como obligatoriedad de campos, longitud de caracteres permitida por campo, manejo de tipos de datos, etc.

### Definición de Herramientas

La definición de herramientas es un paso importante que es necesario tomar previa la correcta definición de los diseños. Es necesario, por parte del desarrollador de software, el tener conocimiento de la utilización de las diferentes herramientas, con el fin de sacar el mayor provecho de éstas.

Las herramientas básicas a utilizarse dentro de este proyecto como ya lo vimos, se pueden diferenciar en tres categorías: de diseño, de codificación, y de pruebas.

### **Herramientas de Diseño**

Las herramientas de diseño se tratarán de dos especialmente dedicadas a los dos tipos de diseños utilizados dentro del proceso de desarrollo de software. Estas herramientas son Sybase Power Designer y Adobe Illustrator CS5, utilizadas para los diseños y diagramas UML, y para la construcción de los diseños gráficos del sitio.

### Sybase Power Designer

Power Designer es una herramienta desarrollada por la empresa Sybase, siendo la herramienta más utilizada en el mundo para el modelado de datos. Estos modelados se realizan con el fin de permitir a las organizaciones el visualizar, analizar y manipular meta-data para poseer una efectiva información de la arquitectura de la empresa.

Esta herramienta provee, además, un acercamiento basado en el modelo para relacionar a los negocios con las tecnologías de la información, lo cual facilita la implementación de arquitecturas para las empresas.

Las diferentes implementaciones que permite realizar Power Designer se realizan en base a diversas técnicas de modelado, entre las cuales se pueden destacar Unified Modeling Language (Lenguaje Unificado de Modelado), Business Process Modeling (Modelado de Procesos del Negocio) y Market-Leading Data Modeling (Modelado de Datos de Liderazgo de Mercado). Estas técnicas de modelado se integran a diferentes plataformas de trabajo, como .NET, Power Builder, JAVA y Eclipse, lo cual permite al negocio un análisis correcto así como desarrollar soluciones de bases de datos para el negocio.

La utilización de esta herramienta se ha planteado en el desarrollo del proyecto en base a la utilización de diferentes diagramas relacionados con UML para el desarrollo de las especificaciones y diseños del producto de software.

La utilización de diagramas UML se realizarán en relación a la capacidad que poseen para explicar la realidad y métodos de trabajo del negocio. De esta manera, es necesario determinar la naturaleza de dichos diagramas para el desarrollo del proyecto.

Los diagramas UML son diagramas para visualizar, especificar, construir y documentar los sistemas. Estos permiten describir el funcionamiento de los sistemas, incluyendo aspectos conceptuales, de proceso, del negocio y las funcionalidades del sistema. Este lenguaje de modelado permite especificar los métodos y procesos en base a un lenguaje gráfico estandarizado, y con ésto indicar el proceso correcto para el desarrollo de software.

Existe un gran número de diagramas UML, los cuales deben ser tomados en cuenta al momento de modelar un sistema. Para el proyecto actual, se planea realizar algunos diagramas, entre los cuales se encuentran: de clases, de casos de uso, de secuencia.

Además de los diagramas UML, se planea la utilización de Sybase Power Designer para el modelado de la base de datos, al generar el modelo conceptual de la base de datos a ser utilizada.

### Adobe Illustrator CS6

Illustrator, en su versión CS6, es un producto de la familia Adobe que permite el realizar diseños de manera profesional. Este producto es ampliamente utilizado alrededor del mundo para el diseño de productos de impresión y de web, debido a la sencillez que presenta para la creación y la edición, y a las grandes capacidades y herramientas que presenta para los desarrolladores. Permite visualizar colores y definir escalas utilizadas para trabajos tanto de impresión como de web, definidos por sus escalas y necesidades. Estas se han definido debido a las necesidades y naturalezas de los trabajos, y a la necesidad de definir diferencias entre las escalas lumínicas y de tinta utilizadas para el trabajo de diseño.

Posee una amplia gama de opciones para facilitar el desarrollo web, mejorando así la experiencia de realizar diseños gráficos que se ajusten a las necesidades de los clientes. Entre sus mejoras más notables se encuentran:

- Aplicación de escala basada en el contenido, una funcionalidad que permite mantener la naturaleza de las imágenes al redimensionarlas.
- Fusión automática de imágenes, una funcionalidad que permite observar y fusionar diferentes capas de manera simple, y observar los resultados sin la necesidad de manejar un solo lienzo.
- Opciones de visualización de archivos, mejora sustancial al sistema de manejo de archivos que permite al diseñador poseer un mejor manejo y control sobre su trabajo.
- Panel de ajustes, una nueva implementación que permite al diseñador el manejar transiciones y ajustes de manera rápida y organizada para aplicarlas a su trabajo.

- Conjunto de herramientas más sofisticadas de dibujo y pintura, una mejora sustancial a los métodos de trabajo de ediciones anteriores que permiten la creación y edición de imágenes de manera más simple.

La utilización de Illustrator CS6 para el desarrollo de los diseños gráficos se ha realizado teniendo en cuenta diferentes importantes puntos que los desarrolladores del sistema han deseado tener presentes. Entre las principales razones podemos determinar las siguientes:

- Determinación simple de los colores a ser utilizada en Web.
- Utilización de capas específicas para fuentes, lo que permitirá al desarrollador investigar la fuente, color y tamaño directamente en el diseño en cualquier momento.
- Familiaridad con la herramienta.

### Adobe Dreamweaver CS5

Dreamweaver, en su versión CS5, es otro producto de la familia Adobe, el cual permite de manera rápida y sencilla la creación de sitios web regidos por estándares.

La creación de sitios web mediante esta herramienta se puede realizar de manera simple y ordenada, y permite tanto a desarrolladores con poco conocimiento como a desarrolladores expertos el crear sitios acordes a sus necesidades. Esto por la capacidad de extensión y la forma de trabajo que posee, además de la gran cantidad de lenguajes que permite manejar, incluyendo el estándar para el manejo de páginas Web HTML, asp, php, javascript, entre otros, y el manejo de archivos xml, css, etc.

Una de las principales ventajas de esta aplicación es su naturaleza. Dreamweaver nació como un editor WYSIWYG (What You See Is What You Get), un editor que permite el desarrollo visual de los elementos, de manera que no es necesario ingresar directamente en el código para realizar una aplicación sencilla.

La utilización de Dreamweaver como herramienta principal para el desarrollo del sitio obedece a varias razones, entre las cuales podemos especificar las siguientes:

- El editor permite se maneja en base a los estándares definidos para la creación y edición de sitios web.

- Permite el manejo de tecnología PHP, lo que es necesario para la programación de módulos y componentes.
- Permite la utilización de diferentes tipos de archivos y herramientas para la creación de los sitios, en especial el manejo sencillo de estilos para los templates del sitio.
- El conocimiento previo de la herramienta es fundamental, y el amplio trabajo de los desarrolladores en esta permitirá un trabajo más rápido y utilizado de mejor manera las capacidades del sitio, sin la necesidad de una curva de aprendizaje para la utilización correcta de la herramienta.

## Herramientas de Desarrollo

### Framework CodeIgniter

CodeIgniter es un framework que permite desarrollar proyectos rápidamente que si se tuviera que comenzar desde cero, ya que pone a tu disposición **librerías** para las tareas comunes, así como una interfaz simple y una estructura lógica para acceder a dichas librerías. Así pues, los desarrolladores pueden centrarse exclusivamente en el proyecto **reduciendo la cantidad de código** que necesitas para cierta tarea. Además **CodeIgniter** es open source, trabaja bajo licencia **Apache/BSD-style**, es ligero, rápido, y genera **URLs amigables**. Por otra parte, facilita la vida a los programadores y diseñadores, ya que utiliza la arquitectura denominada **MVC**, **esto es muy bueno para tener separado la lógica los datos y la presentación**, donde los programadores se encargarán de la parte de código y los diseñadores exclusivamente del lenguaje de maquetado, como HTML y CSS.

### Eclipse Helios – IDE

Eclipse es un entorno de desarrollo integrado, de Código abierto y Multiplataforma. Mayoritariamente se utiliza para desarrollar lo que se conoce como "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores. Es una potente y completa plataforma de Programación, desarrollo y compilación de elementos tan variados como sitios web, programas en C++ o aplicaciones Java. No es más que un entorno de desarrollo integrado (IDE) en el que encontrarás todas las herramientas y funciones necesarias para tu trabajo, recogidas además en una atractiva interfaz que lo hace fácil y agradable de usar.

Eclipse Helios es una versión específica para desarrolladores PHP.

## PHP

PHP es un lenguaje de programación diseñado especialmente para desarrollo web y puede ser embebido dentro de código HTML. Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. PHP se encuentra instalado en más de 20 millones de sitios web y en un millón de servidores, aunque el número de sitios en PHP ha declinado desde agosto de 2005. El uso de este lenguaje se debe a que contiene muchas ventajas como son:

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados extensiones).
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos.
- Biblioteca nativa de funciones sumamente amplia e incluida.
- No requiere definición de tipos de variables.
- Tiene manejo de excepciones (desde php5).

## Apache

El servidor HTTP Apache es un servidor web HTTP de código abierto, para plataformas Unix, Microsoft Windows, Macintosh y otras. Su nombre se debe a que Behelendorf quería que tuviese la connotación de algo que es firme y enérgico pero no agresivo, y la tribu Apache fue la última en rendirse al que pronto se convertiría en gobierno de EEUU, y en esos momentos la preocupación de su grupo era que llegasen las empresas y "civilizasen" el paisaje que habían creado los primeros ingenieros de internet. Además Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA. El

servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation.

Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Las ventajas de usar apache se definen a continuación:

- Altamente configurable, casi cualquier cosa que se necesite hacer con un servidor web se puede hacer con apache.
- Está disponible en una amplia gama de plataformas.
- Al ser una tecnología gratuita de código fuente abierto le da una transparencia inmediata, de forma que si quisiéramos ver qué es lo que estamos instalando como servidor, lo podemos saber, sin ningún secreto.
- Trabaja con gran cantidad de Perl, PHP y otros lenguajes de script, Java y páginas JSP, teniendo todo el soporte que se necesita para tener páginas dinámicas.
- Por su diseño modular es muy sencillo ampliar las capacidades del servidor web Apache. En la actualidad hay casi infinitos módulos para Apache que los vamos instalando a medida que se necesitan.
- Otra de sus ventajas es que se puede personalizar la respuesta ante los posibles errores que se den en el servidor, por supuesto se puede configurar para que ejecute un determinado script cuando ocurra un error marcado.

### MySQL Workbench

MySQL Workbench es una herramienta visual de diseño de bases de datos que integra desarrollo de software, Administración de bases de datos, diseño de bases de datos, creación y mantenimiento para el sistema de base de datos MySQL.

Con esta herramienta se puede elaborar una representación visual de las tablas, vistas, procedimientos almacenados y claves foráneas de la base de datos. Además, es capaz de sincronizar el modelo en desarrollo con la base de datos real. Se puede realizar una ingeniería directa e ingeniería inversa para exportare e importar el esquema de una base de datos ya existente el cual haya sido guardado o hecho copia de seguridad con MySQL Administrador.

MySQL Workbench puede generar también el guión necesario para crear la base de datos que se ha dibujado en el esquema; es compatible con los modelos de base de datos de DBDesigner 4 y soporta las novedades incorporadas en MySQL 5.x

## HTML5

Esta nueva versión del lenguaje básico del Web proporciona mecanismos para simplificar el trabajo y facilitar la inclusión de elementos multimedia. El principal criterio de diseño de html5 ha sido el de resolver problemas prácticos, y con este objetivo adopta soluciones dirigidas a facilitar el trabajo en situaciones reales.

Por otra parte, existe un medio para que el usuario pueda hacer llegar datos de entrada a un servidor: los formularios, que recogen información que después remiten a aplicaciones que se ejecutan en el servidor (CGI, API, JSP, *servlets* u otras interfaces). Html5 define más de una docena de nuevos controles (*email, range, date, time, placeholder, autofocus*, etc.) que actúan, por fin, sin necesidad de utilizar *JavaScript*, un lenguaje de programación que no todos los usuarios tienen habilitado. Html5 propone, pues, que sean los navegadores, y no los creadores de contenido, quienes faciliten la entrada y la validación de datos que tienen un patrón regular o están sometidos a restricciones. Así, el dolor de cabeza que supone para los diseñadores de páginas web el verificar el formato de direcciones electrónicas, intervalos de valores, términos de búsqueda, colores, fechas y horas, entre otros tipos de datos, se descarga ahora sobre los navegadores.

Html5 se presenta como un marco estable para el desarrollo de páginas web. Ha sido definido para durar muchos años gracias a diversos mecanismos que permiten extender el lenguaje con facilidad. La filosofía de html5 aboga además por los estándares abiertos, que son fundamentales para impulsar la innovación e introducir en la sociedad los beneficios de las nuevas tecnologías. Bien sabemos que los estándares abiertos tienen la capacidad de mejorar nuestras vidas, y de ello son ejemplo las nuevas funciones multimedia de html5.

## **Tecnologías de desarrollo**

### Ajax

El término AJAX es el acrónimo de Asynchronous JavaScript And XML, Javascript Asíncrono y XML. Se trata de una técnica para usar el protocolo HTTP mediante Javascript. Aunque en el nombre pone XML, puede usarse simplemente para solicitar páginas al servidor con Javascript, documentos que al fin y al cabo son contenidos de texto. Aunque también pueden manejarse documentos XML e insertarlos en el DOM del navegador para su posterior acceso.

Lo de asíncrono se debe a la característica para descargar contenidos desde el servidor en segundo plano. Así mientras se van recuperando esos datos, la página puede seguir su curso de ejecución. AJAX no es un lenguaje de programación, sino una técnica basada en el objeto XMLHttpRequest y combinada con JavaScript, HTML y CSS.

Realmente el componente que permitía este uso es bastante antiguo, pues los elementos iframe de Internet Explorer (1996) ya permitían la descarga de documentos en un marco separado y de forma asíncrona. Más adelante (1999) Microsoft creó el objeto ActiveX llamado XMLHttpRequest, primer precursor del actual XMLHttpRequest y que luego sería también incluido en otros navegadores. En el 2006 el World Wide Web Consortium (W3C) emite la primera especificación XMLHttpRequest, definiéndola como una API que provee funcionalidad de programación en el cliente mediante transferencia de datos entre cliente y servidor.

### jQuery

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

jQuery es software libre y de código abierto, posee un doble licenciamiento bajo la licencia MIT y de la GNU General Public License, Versión 2. jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en Javascript que de otra manera requerirían de mucho más código. Es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

jQuery consiste en un único fichero JavaScript que contiene las funcionalidades comunes de DOM, eventos, efectos y AJAX. La característica principal de la biblioteca es que permite

cambiar el contenido de una página web sin necesidad de recargarla, mediante la manipulación del árbol DOM y peticiones AJAX. Para ello utiliza las funciones `$( )` o `jQuery()`.

### **Editores de Texto**

Existen, sin embargo, un número de archivos que no pueden ser abiertos por la herramienta Dreamweaver, y cuya edición puede ser necesaria para el correcto funcionamiento del sitio. Esto es provocado debido a su formato y a las seguridades necesarias de los mismos. Por lo tanto, es recomendable, dependiendo de la naturaleza de los archivos, el utilizar una herramienta para la edición de texto mucho más simple o dedicada específicamente para dicho tipo de archivos.

Por lo tanto, se ha decidido la utilización de un editor mucho más simple para estos casos especiales, el cual se trata de notepad y wordpad, ambas versiones muy simples que vienen con el sistema operativo Windows, y que son herramientas que pueden ser utilizadas sin ningún problema por los desarrolladores, y que permitirá el manejo de estos archivos conflictivos de manera fácil.

### **Herramientas de Pruebas**

La creación de páginas web es un desarrollo que debe ser puesto a prueba en diferentes navegadores para poder obtener un producto óptimo que se vea de manera similar para cada usuario. Por lo tanto, se ha decidido realizar pruebas en los tres tipos de navegadores más utilizados en la actualidad: Internet Explorer, Firefox y Google Chrome.

#### Internet Explorer Versión 9

Desde la aparición de la compañía Windows, ha evolucionado hasta convertirse en la proveedora del sistema operativo más utilizado en el mundo. Su promoción que incluía a Internet Explorer junto al sistema operativo permitió una masificación del acceso y uso de internet, y provocó que Internet Explorer sea el navegador más utilizado en el mundo. Según W3Counter, hasta septiembre de 2011 cerca del 86% de la población utiliza un sistema

operativo Windows, mientras que alrededor del 50% de la población (51,73 %) utilizan un navegador Internet Explorer, en sus versiones desde la 6 hasta la 8.

Por esta razón, las pruebas deben realizarse en estas versiones de navegador, si bien la tendencia de utilización de estos navegadores se encuentra a la baja. De estos tres, el más utilizado es todavía la versión 7, seguida por la versión 8 y 9. Estas dos versiones son aquellas que se acoplan de mejor manera al trabajo web, debido a que se encuentran mucho más apegadas a los estándares definidos para los sitios web.

La versión del navegador de IE6 es, a diferencia de las otras, un problema menos sencillo de resolver. Al poseer todavía un 13,28% de uso entre la población mundial, es necesario tenerlo en cuenta. Su programación diverge de los estándares, por lo que muchos estándares y trabajo (especialmente en la utilización de estilos por medio de css) no se aplican a este navegador, y deben ser generados específicamente para estos casos.

#### Mozilla Firefox Versión 16.

Mozilla Firefox es probablemente el segundo navegador más popular en el mundo, contando con alrededor de 32% de usuarios. Este es un navegador multiplataforma desarrollado por la Corporación Mozilla, la Fundación Mozilla y varios voluntarios de manera libre, lo que ha ayudado a su gran popularidad. Además, posee grandes niveles de seguridad, de velocidad y de extensión, lo que mejora de gran manera la experiencia con el usuario. Por sus capacidades y similitudes entre versiones, se ha escogido como el navegador de prueba a su última versión, la 16.

Es además uno de los navegadores más populares para el desarrollo web debido a sus capacidades, gran cantidad de extensiones para desarrolladores y además por su buen manejo en relación a estándares. Estas funcionalidades hacen que este navegador sea la principal herramienta para pruebas del equipo de trabajo, y aquel sobre el cual se basará el trabajo para ajustarla a los diseños gráficos existentes.

#### Google Chrome

Google Chrome, desarrollado por Google, es el navegador que más ha crecido en la actualidad. Su interfaz amigable, así como la popularidad adjunta del buscador y las aplicaciones Google (Google Maps, Google Earth, Google Docs, Gmail, etc.) ha provocado un crecimiento paulatino de este navegador, si bien se encuentra todavía en una utilización de alrededor del 4%.

Este crecimiento ha provocado que este navegador sea escogido para la realización de pruebas. Esto no debería generar muchos cambios ni diferencias con Firefox, puesto que Chrome se ajusta a los estándares generados para el desarrollo Web y el manejo de estilos, si bien todavía carece de funcionalidades para la lectura de cierto tipo de contenido, como xml. Además, este navegador ha sido escogido debido a que se ajusta a otros navegadores y permite, debido a que utiliza un mismo motor de navegación, que los cambios realizados sean también aplicables a otros navegadores como Safari y Opera, lo que hace innecesaria las pruebas en dichos navegadores.

#### 3.2.4. OTROS REQUERIMIENTOS

Después de definir los requerimientos del sistema es momento de delimitar algunos requisitos indispensables para el buen funcionamiento del sitio web. Se habla de los requerimientos de software y hardware, los cuales irán de mano con las exigencias del cliente y del proyecto.

##### 3.2.4.1. REQUERIMIENTOS DE HARDWARE

Los usuarios del sistema, para poder acceder al sitio, necesitan un computador (fijo, portátil o dispositivo móvil), el requisito indispensable es poseer acceso a internet y un navegador instalado.

Por parte del servidor, los requerimientos a tomar en cuenta para el óptimo funcionamiento del sistema son los siguientes:

- Equipo/procesador: PC con procesador a 1 GHz o superior
- RAM: 1GB o superior

- Tarjeta de red
- Periféricos entrada: monitor, teclado y mouse
- Disco duro: IDE (7.500 rpm y 2 Mb de caché) o superior

#### 3.2.4.2. REQUERIMIENTOS DE SOFTWARE

Para los usuarios los requerimientos son mínimos, de manera que debe poseer un ordenador funcional y un browser, al igual que posibilidad para navegar en el internet. Por lo mismo, los requerimientos de software indicados son:

- Sistema operativo: Windows XP o superior, Linux, Mac OSX
- Navegador de internet: IE6 en adelante, Firefox 2.0 en adelante, Opera, Safari.

Por parte del servidor, los requerimientos son más avanzados, puesto que necesita soportar las tecnologías indicadas para el funcionamiento del sitio. Además de un sistema operativo funcional, siendo recomendado Red Hat 6.2 en adelante o superior, los requerimientos de software son indicados en la Tabla III - 2.

Software	Min Requisitos	Recomendado
<b>PHP</b>	5.5.0	5.3.8
<b>MySQL</b>	5.0.0.x o superior	5.5.20
<b>Apache</b>	1.3 o superior	2.0

**Tabla III - 2.:** Tabla de Requerimientos de Software.

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

### 3.3. GESTIÓN DE CONFIGURACIÓN

Una vez que se ha realizado todo el levantamiento de la información y se procedió a transformarlos en requerimientos funcionales y no funcionales, los cuales quedaron expuestos al cliente para su respectiva aprobación. Con ésta aprobación todos estos requerimientos pasaron a formar parte de la línea base.

#### 3.3.1. DEFINICIÓN DE LÍNEA BASE

Una línea base es una especificación o producto revisado y aprobado formalmente, que sirve como base para el desarrollo posterior, y puede ser modificado solo a través de procedimientos formales de control de cambios.

Una línea base también se usa para referirse a una versión particular de un elemento de software que ha sido aprobado. En cualquier caso, la línea base solo se puede modificar a través de procedimientos formales de control de cambios. Una línea base, junto con todos los cambios aprobados a la línea base, representa la Configuración aprobada actual.

Con esta idea clara, el presente proyecto establecerá la siguiente línea base para cualquier cambio, y establece:

- Al finalizar cada fase del proyecto se revisará el plan del proyecto y de ser necesario se podrá definir una nueva línea base.
- En las reuniones periódicas con el cliente, el mismo puede presentar un cambio o un nuevo requerimiento el cual será documentado.
- Se analizará la factibilidad del cambio en una reunión posterior con los ingenieros desarrolladores del sistema.
- El análisis de factibilidad será documentado y contendrá una estimación de costos, plazos, alcance e impacto sobre las desviaciones que se produzcan.
- Se presentará al cliente el análisis realizado y de ser aprobado se procederá a ejecutar los cambios solicitados.
- Se actualizará la línea base.
- Se utilizará el siguiente formato:

<b>Cambio de Línea Base</b>
<b>Fecha Inicia:</b>
<b>Fecha Termina:</b>
<b>Solicitante:</b>
<b>Tipo de Cambio:</b>

<b>Descripción del cambio:</b>
<b>Análisis de Factibilidad</b>
<b>Descripción:</b>
<b>Costo:</b>
<b>Plazo:</b>
<b>Observaciones Generales:</b>

**Tabla III - 3.:** Formulario de Cambios en Línea Base  
**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

### 3.3.2. RESTRICCIONES

Es necesario definir ciertos puntos a tener en cuenta al momento de realizar los diseños sobre el sitio, puntos que afectarán de igual manera la futura implementación del sitio. Estos puntos sobre el diseño y la implementación restringirán el trabajo futuro, enfocando aspectos específicos que deberán ser tomados en cuenta y puntos que no serán implementados.

Es importante, por lo tanto, el presentar las diferentes restricciones en base a sus diferentes características, de manera que permitan un desarrollo óptimo de la aplicación. Estas

características se definirán en base a tres puntos esenciales: funcionalidad, requerimientos y el diseño.

### Restricciones sobre la Funcionalidad

La funcionalidad poseerá varias restricciones necesarias previas la implementación el sitio. Estas restricciones deberán ser tomadas en cuenta pues determinarán las diferentes herramientas a ser generadas. En base a esto, las restricciones sobre la funcionalidad son las siguientes:

- Las herramientas de administrador se ajustarán a las definidas por el administrador del contenido en cuanto a su funcionalidad (creación, actualización, borrado y consulta).
- Las consultas generales se realizarán de manera automática en el administrador, y no se generará otro tipo de consulta general.
- El sitio se ajustará a las características del administrador de contenidos, definiendo componentes y módulos adicionales sin modificar la codificación original del sitio para permitir actualizaciones futuras del código original.

### Restricciones sobre los Requerimientos

Los requerimientos presentados por el cliente a los desarrolladores serán congelados tras la presentación de los diferentes requerimientos. Por lo mismo, es necesario el definir restricciones que limitarán al desarrollo, siendo todo requerimiento futuro implementado en una actualización posterior.

Estas restricciones sobre los requerimientos quedan definidas como:

- El cliente no cambiará los requerimientos aceptados en medio de la codificación.
- Los requerimientos no contendrán acciones extras sobre las estipuladas y la utilización de componentes adicionales de terceros será realizadas y documentadas en situaciones previamente estipuladas.

### Restricciones sobre el Diseño

Las restricciones sobre los diseños se referirán a las diferentes acciones que, si bien podrían ser realizadas por los desarrolladores, su complejidad genera que no sean recomendadas para el diseño.

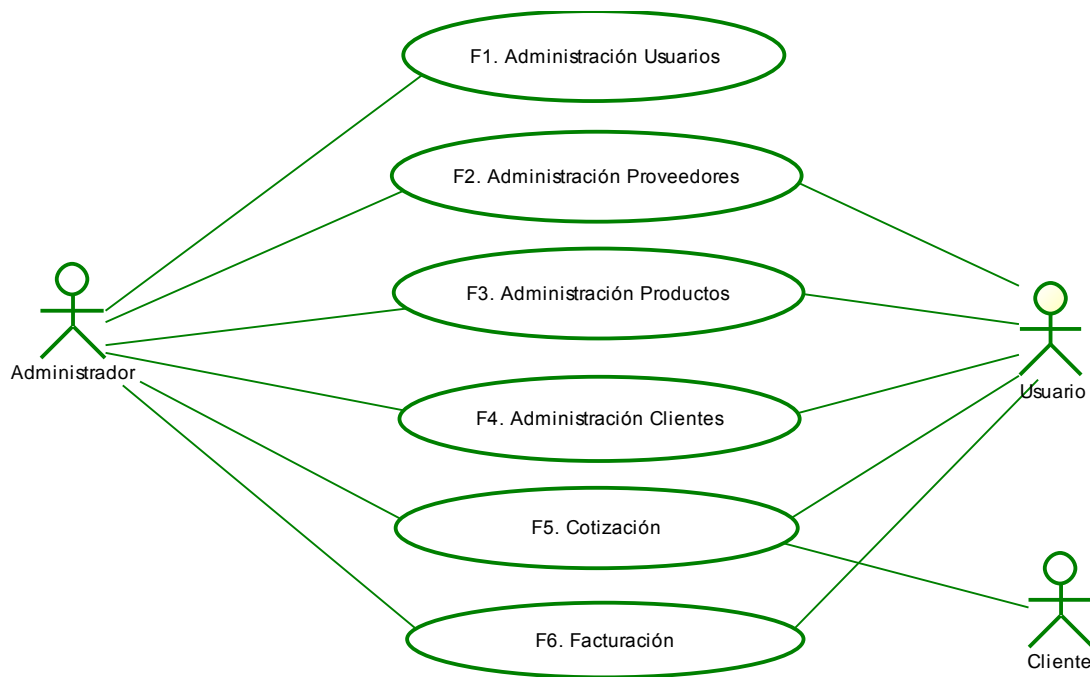
Las restricciones de diseño a ser tomadas en cuenta en el desarrollo del sistema son las siguientes:

- Las secciones del sitio serán reutilizadas, por lo que los diseños de estas no pueden diferir.
- El esquema general del sitio (layout) no cambiará en ninguna de las páginas del sitio, conservando elementos como menú principal y secundario en todas las páginas.
- El diseño de los contenidos no poseerá elementos como tablas curvas, círculos o variaciones de color fuera del estándar general del sitio.

## CAPITULO 4

### DESARROLLO DEL CICLO DE VIDA

Como se ha detallado en capítulos anteriores, el desarrollo del ciclo de vida del sistema se realizara en dos ciclos, en el primer ciclo nos focalizaremos en todas las administraciones del sistema, es decir Administración de Usuarios, Administración de Proveedores, Administración de Productos y Administración de Clientes, en el segundo ciclo nuestro foco será solucionar los procesos que el sistema automatizara es decir el de Facturación y el de Proforma. Con todo este en mente, se quiere presentar a continuación el diagrama general que contiene Administraciones y procesos para tener una idea clara de a donde se quiere llegar.



**Figura IV - 1:** Diagrama General

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

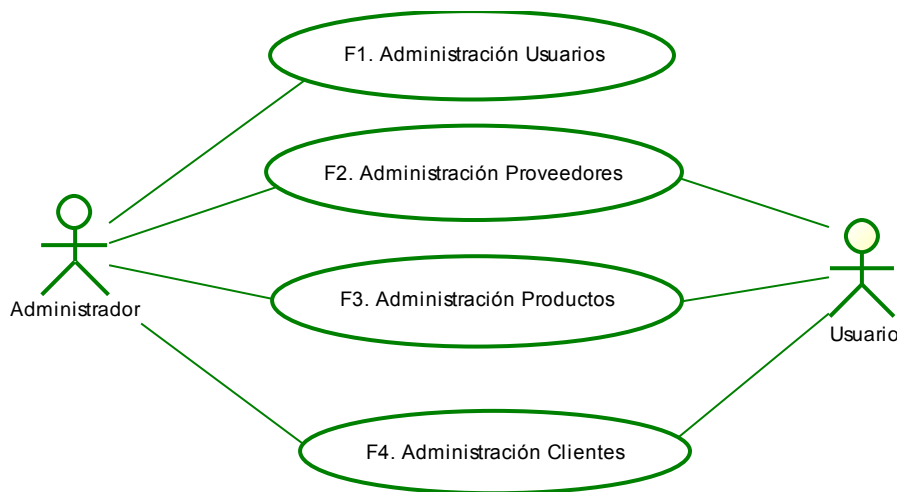
#### 4.1. CICLO 1

##### 4.1.1. ANALISIS DE REQUERIMIENTOS

En esta parte del desarrollo del proyecto, se da rienda al primer acercamiento con el cliente, en donde se tratará de recolectar todos los requerimientos del mismo, para transformarlos en funcionalidades eficientes. Y plasmar cada necesidad para el desarrollo de la empresa y por supuesto para este desarrollo.

#### 4.1.1.1. Diagrama General

La definición de un diagrama general para el proyecto en curso se debe hacer diferenciando las funcionalidades del sistema. El diagrama general que se observa en la Figura IV - 2 permite visualizar las principales funcionalidades a ser desarrolladas para la presente disertación, y las relaciones de éstas con los diferentes tipos de usuarios del sistema.



**Figura IV - 2:** Diagrama Ciclo 1

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.1.2. Diagrama de Casos de Uso

Los diagramas de casos de uso que se definirán a continuación son una representación de los requerimientos de los usuarios. Estos reflejan las necesidades del cliente de manera concreta para el futuro desarrollo. Estos se han dividido en base a su funcionalidad y naturaleza, en las diferentes administraciones que se mostraron en el diagrama general. Las funcionalidades se definirán a continuación.

#### 4.1.1.2.1 Administración de Usuarios

**Caso de Uso:** Administración de Usuarios

**Identificador:** F1

La administración a tratarse a continuación trata sobre el manejo de los usuarios del sistema. Por lo tanto, ajustándose a opciones de administración tradicionales, se ha definido los procesos de ingreso, actualización y consulta.



**Figura IV - 3:** Diagrama de Administración de Usuarios

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

Las tareas de administración se deberán realizar por parte de los actores que interactúan con el aplicativo para el manejo de usuarios. Por lo tanto, este proceso debe regirse a un flujo realizado por el usuario el cual se puede determinar de manera correcta en la Tabla IV - 1.

**Caso de Uso:** Administración de Usuarios

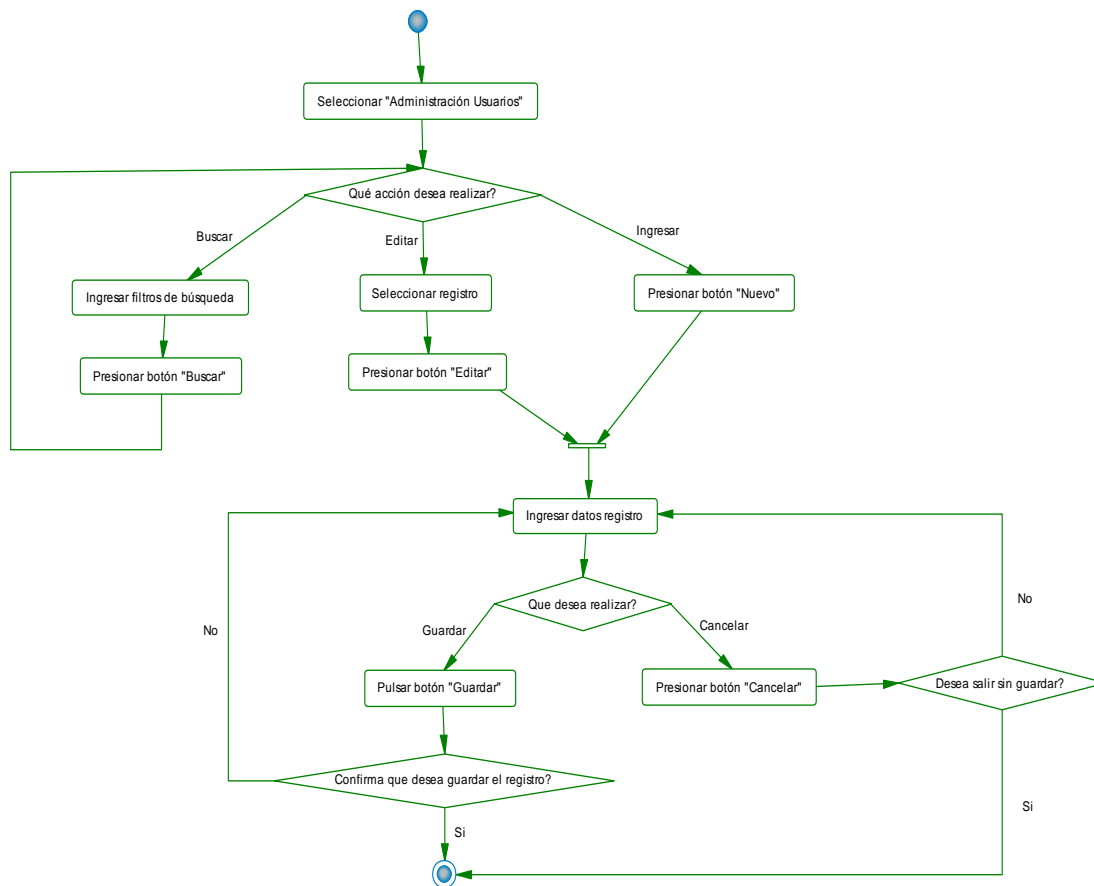
**Identificador:** F1

**Actores:** Administrador

**Pre-condición:**

Nuevas contrataciones de personal en la empresa implican crear nuevos usuarios.  
Se necesitan actualizar los datos de un usuario.

**Flujo:**



**Pos-condición:**

La base queda actualizada  
Los nuevos usuarios pueden hacer uso del sistema.

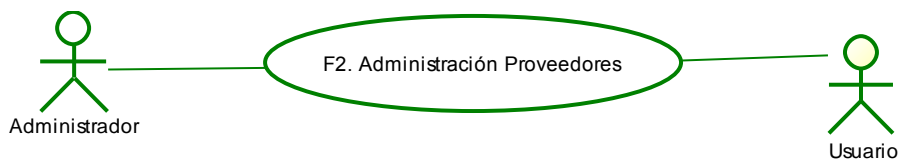
**Tabla IV – 1:** Tabla de Flujo para la Administración de Usuarios  
**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.1.2.3. Administración de Proveedores

**Caso de Uso:** Administración de Proveedores

**Identificador:** F2

La administración a tratarse a continuación trata sobre el manejo de los diferentes proveedores de los productos. Por lo tanto, ajustándose a aspectos de administración tradicionales, se ha definido los procesos de ingreso, actualización y consulta presentes en esta administración y que se pueden observar en la Figura IV - 4.



**Figura IV - 4:** Diagrama de Administración de Proveedores

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

Los casos de uso determinados para esta administración se definen en base a cada uno de estos procesos. Al igual que el manejo de usuarios, tanto el administrador como el usuario poseerán diferentes actividades en un mismo proceso, desarrollándose éstos en diferentes secciones del sitio.

La Tabla IV – 2 explicará claramente el flujo que el autor debe hacer en esta administración.

**Caso de Uso:** Administración de Proveedores

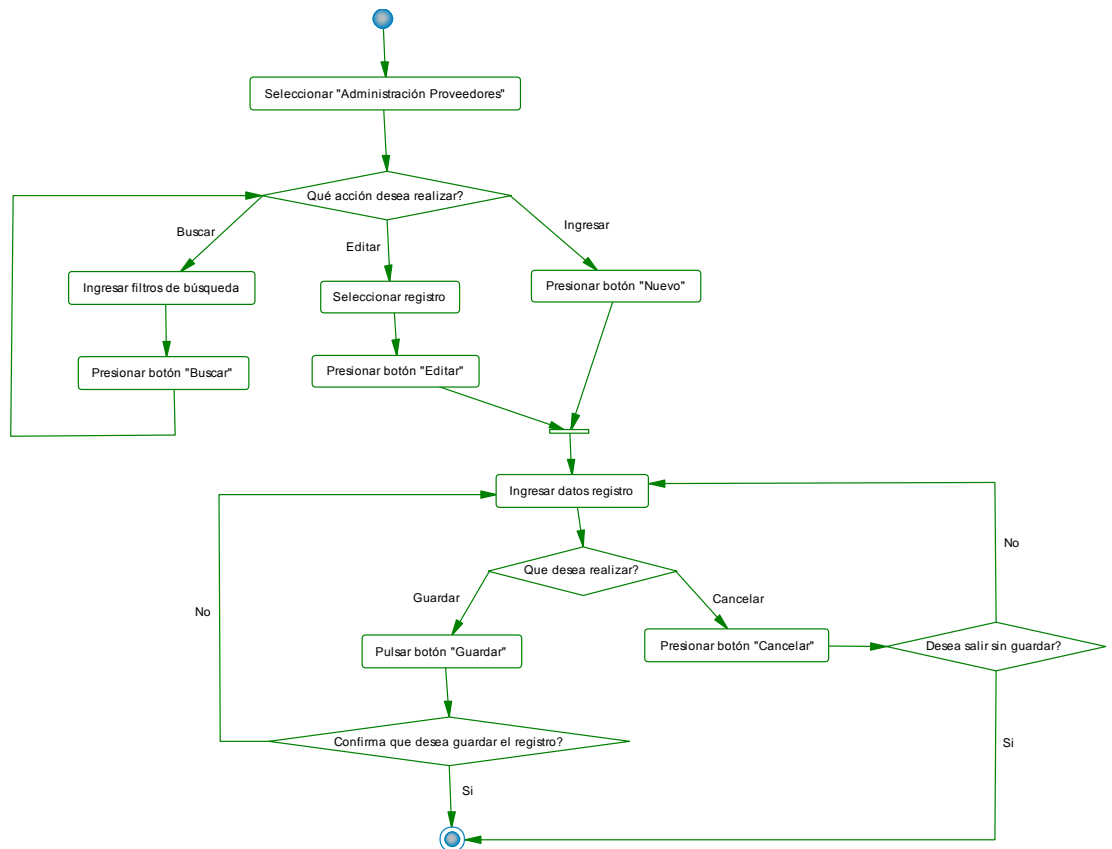
**Identificador:** F2

**Actores:** Administrador/ Usuario

**Pre-condición:**

La empresa necesita adquirir los productos de una nueva marca, y esto implica registrar un nuevo proveedor.

**Flujo:**



**Pos-condición:**

La base queda actualizada  
Se podrán ingresar los productos del nuevo proveedor.

**Tabla IV – 2:** Tabla de Flujo para la Administración de Proveedores

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.1.2.3. Administración de Productos

##### **Caso de Uso:** Administración de Productos

##### **Identificador:** F3

El manejo de los diferentes productos registrados en el portal se realizará por parte del Administrador del sistema.

Los métodos de ingreso, actualización, inhabilitación de los mismos son tradicionales y no serán definidos a detalle, puesto que se consideran parte del administrador propio de los productos y su manejo debe definirse como ya definido para la administración de los productos.

La Figura IV – 5 presenta esta administración de los productos, incluyendo sus actores.



**Figura IV – 5:** Diagrama de Administración de Productos

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

La administración de los productos es la acción esencial en el desarrollo de la página web, debido a que es el principal requerimiento del usuario. De esta manera, el manejo de los diferentes productos se realizará por parte del administrador mediante tareas de búsqueda, ingreso y actualización de los mismos.

Las tareas de administración se deberán realizar por parte de los autores del aplicativo. Por lo tanto, este proceso debe regirse a un flujo realizado por el usuario el cual se puede determinar de manera correcta en la Tabla IV - 3.

**Caso de Uso:** Administración de Productos

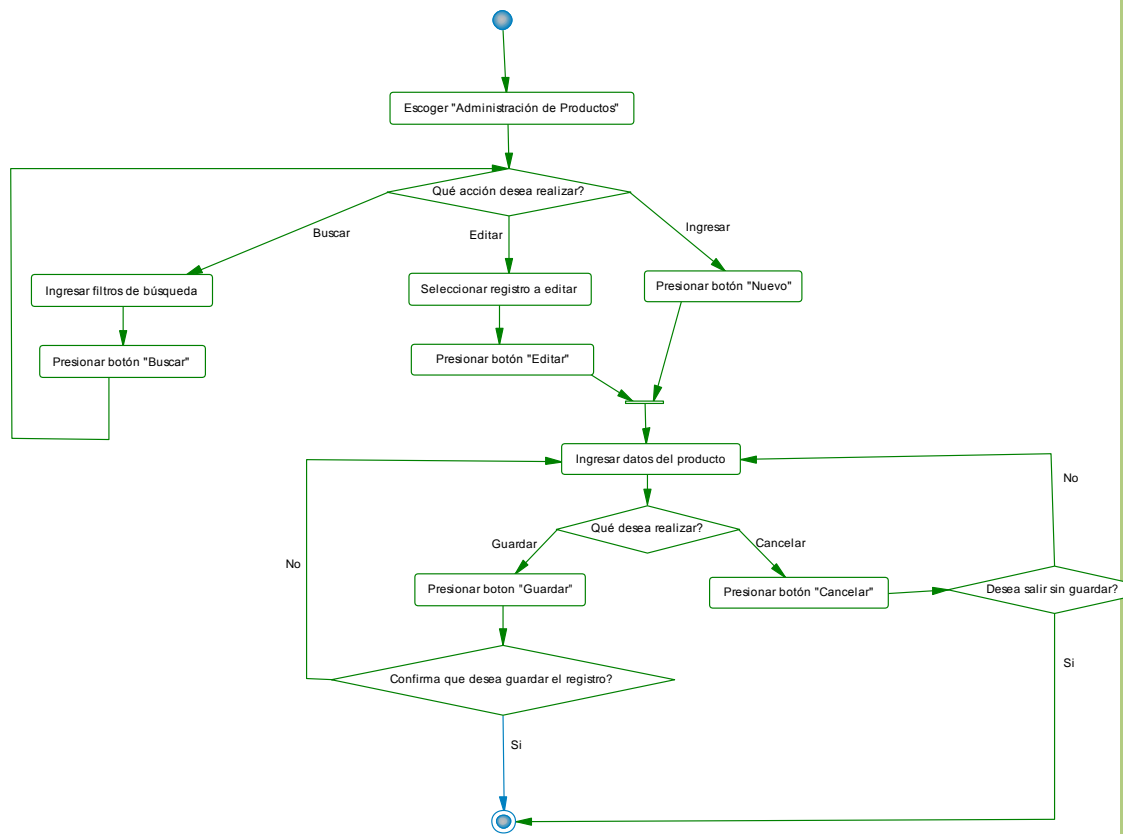
**Identificador:** F3

**Actores:** Administrador/ Usuario

**Pre-condición:**

El negocio adquiere nuevos productos, y el administrador debe registrarlo.  
 Hay cambios en el precio, descripción o detalle de los productos.  
 Hay cambios en la disponibilidad de los productos

**Flujo:**



**Pos-condición:**

La base queda actualizada  
 Los productos quedan actualizados y disponibles para ser usados en cotizaciones o en el proceso de facturación.

**Tabla IV – 3:** Tabla de Flujo para la Administración de Productos

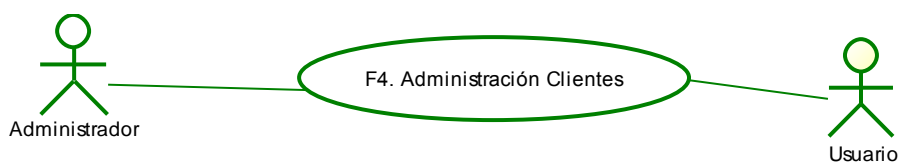
**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.1.2.4. Administración de Clientes

**Caso de Uso:** Administración de Clientes

**Identificador:** F4

Es muy importante para la empresa, administrar la cartera de sus clientes, por esta razón es crucial definir las opciones de administración. Estas opciones serán tratadas en la definición de cada una de las secciones de esta administración. La Figura IV- 6 muestra la administración de clientes junto con los actores involucrados.



**Figura IV - 6:** Diagrama de Administración de Clientes

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

Las tareas de administración se deberán realizar por parte del administrador en la herramienta. Por lo tanto, este proceso debe regirse a un flujo realizado por el usuario el cual se puede determinar de manera correcta en la Tabla IV - 4.

**Caso de Uso:** Administración de Clientes

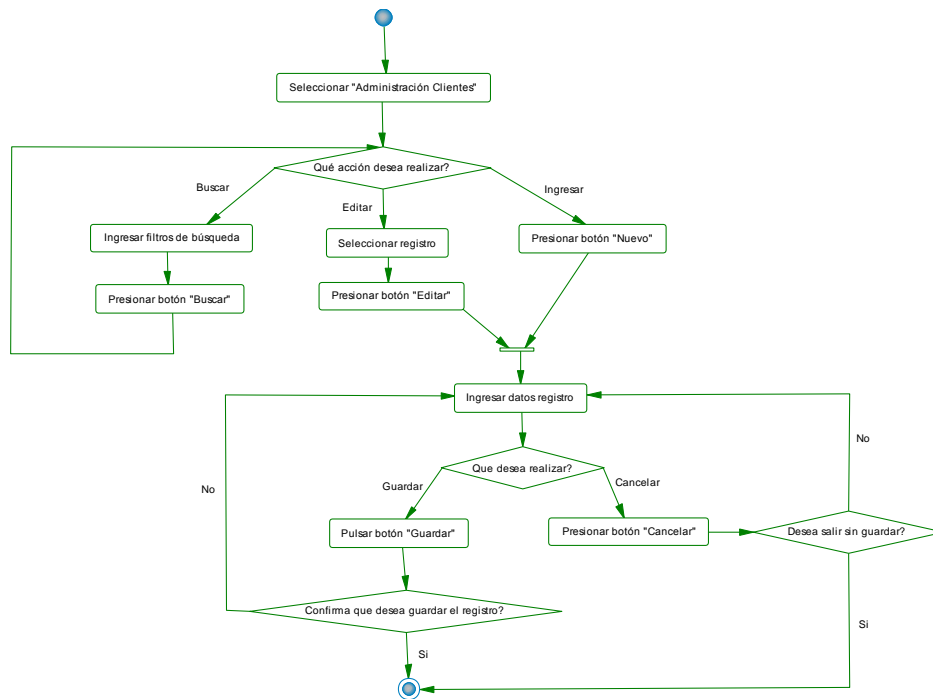
**Identificador:** F4

**Actores:** Administrador/Usuario

**Pre-condición:**

Para los procesos de facturación y cotización la empresa necesita registrar el cliente, con el propósito de mantener un historial y una cartera de clientes.

**Flujo:**



**Pos-condición:**

La base queda actualizada  
 Los clientes quedan actualizados y disponibles para ser usados en cotizaciones o en el proceso de facturación.

**Tabla IV – 4:** Tabla de Flujo para la Administración de Clientes

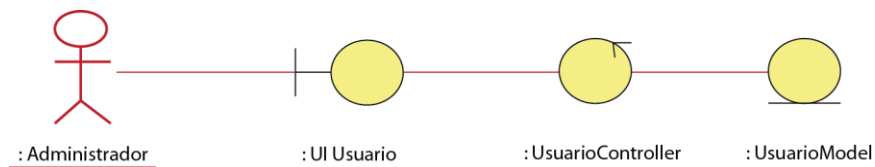
**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.1.3. Diagrama de Casos de Uso Realizables

Una vez que diagramados los requerimientos con nuestro cliente mediante los casos de uso, es preciso realizar una descripción abstracta de los mismos a través de casos de uso realizables, éste análisis luego evolucionará hacia otras clases más detalladas en el modelo de diseño. A continuación las Figuras mostrarán cada uno de los diagramas realizables por Caso de Uso.

**Caso de Uso:** Administración de Usuarios

**Identificador:** F1

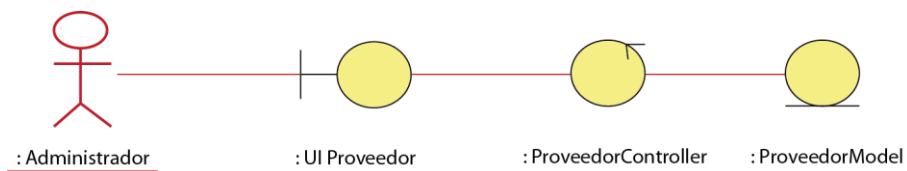


**Figura IV – 7:** Diagrama de Administración de Usuarios

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

**Caso de Uso:** Administración de Proveedores

**Identificador:** F2

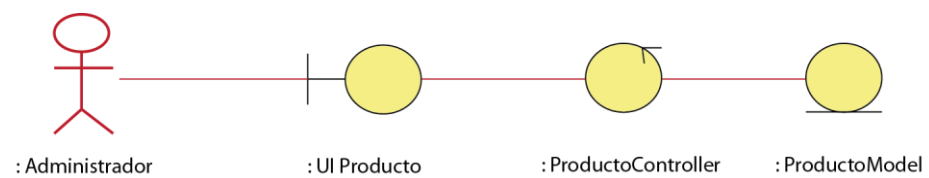


**Figura IV – 8:** Diagrama de Administración de Proveedores

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

**Caso de Uso:** Administración de Productos

**Identificador:** F3

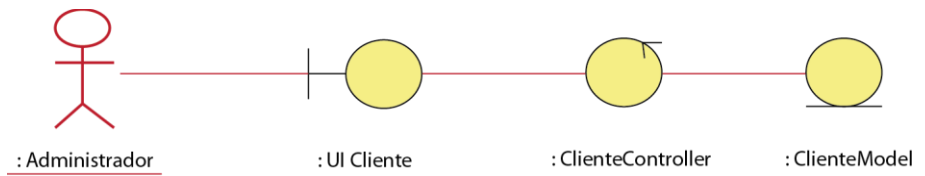


**Figura IV – 9:** Diagrama de Administración de Productos

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

**Caso de Uso:** Administración de Clientes

**Identificador:** F4



**Figura IV – 10:** Diagrama de Administración de Clientes

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.2. DISEÑO DEL SISTEMA

Lo que comprende la fase de diseño es la generación de diagramas y modelos que se ajusten a los requerimientos que ya fueron previamente definidos. En base a estos requerimientos es necesario definir clases definitivas representadas en diagramas de secuencias y especificaciones de cada clase con el fin de que el programador pueda generar el código correspondiente.

##### 4.1.2.1. Diagramas de Secuencia

Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada caso de uso. A continuación se mostrará cada diagrama de secuencia:

Caso de Uso: Administración de Usuarios

Identificador: F1

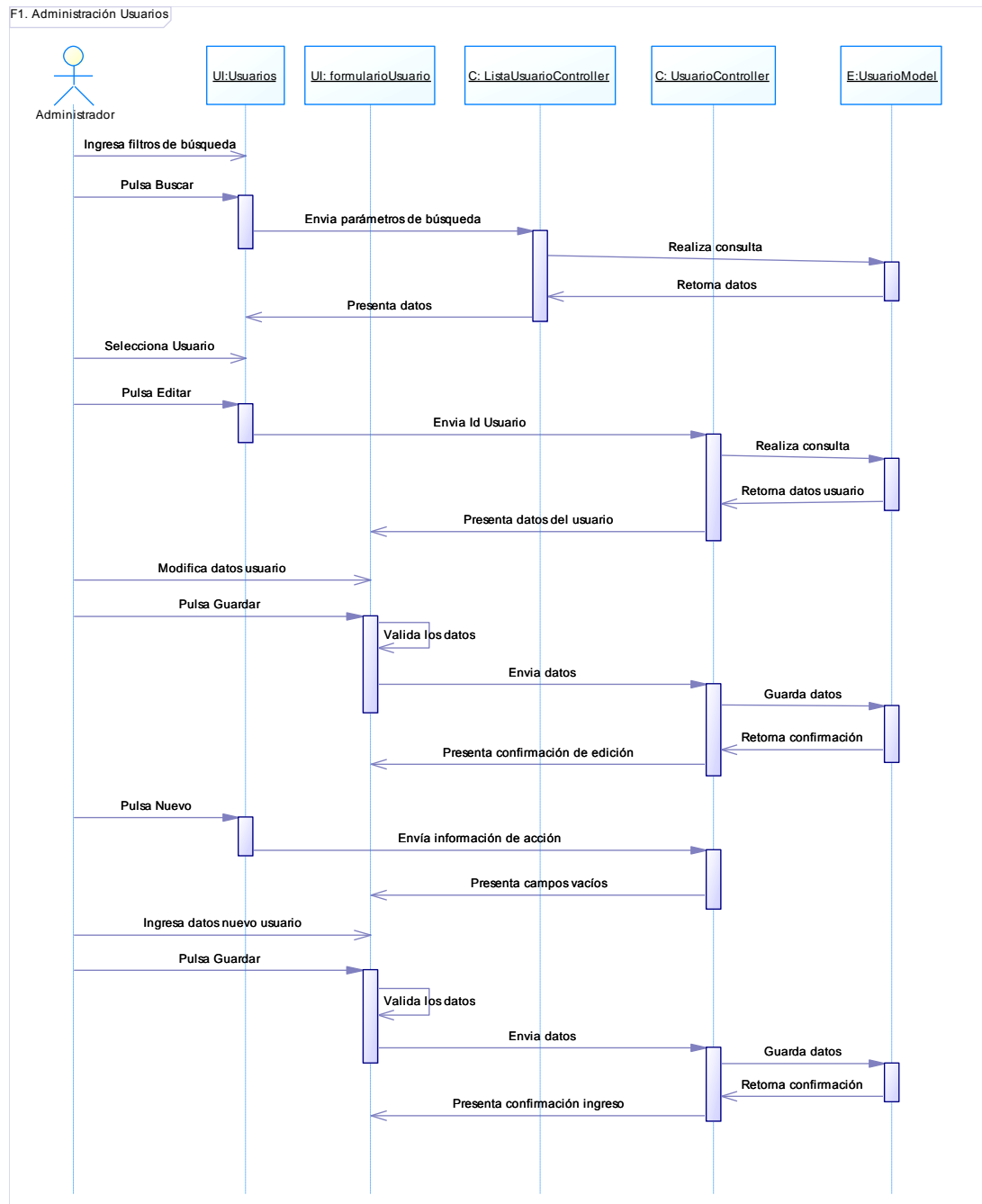


Figura IV – 11: Diagrama de Secuencia para la Administración de Usuarios

Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

Caso de Uso: Administración de Proveedores

Identificador: F2

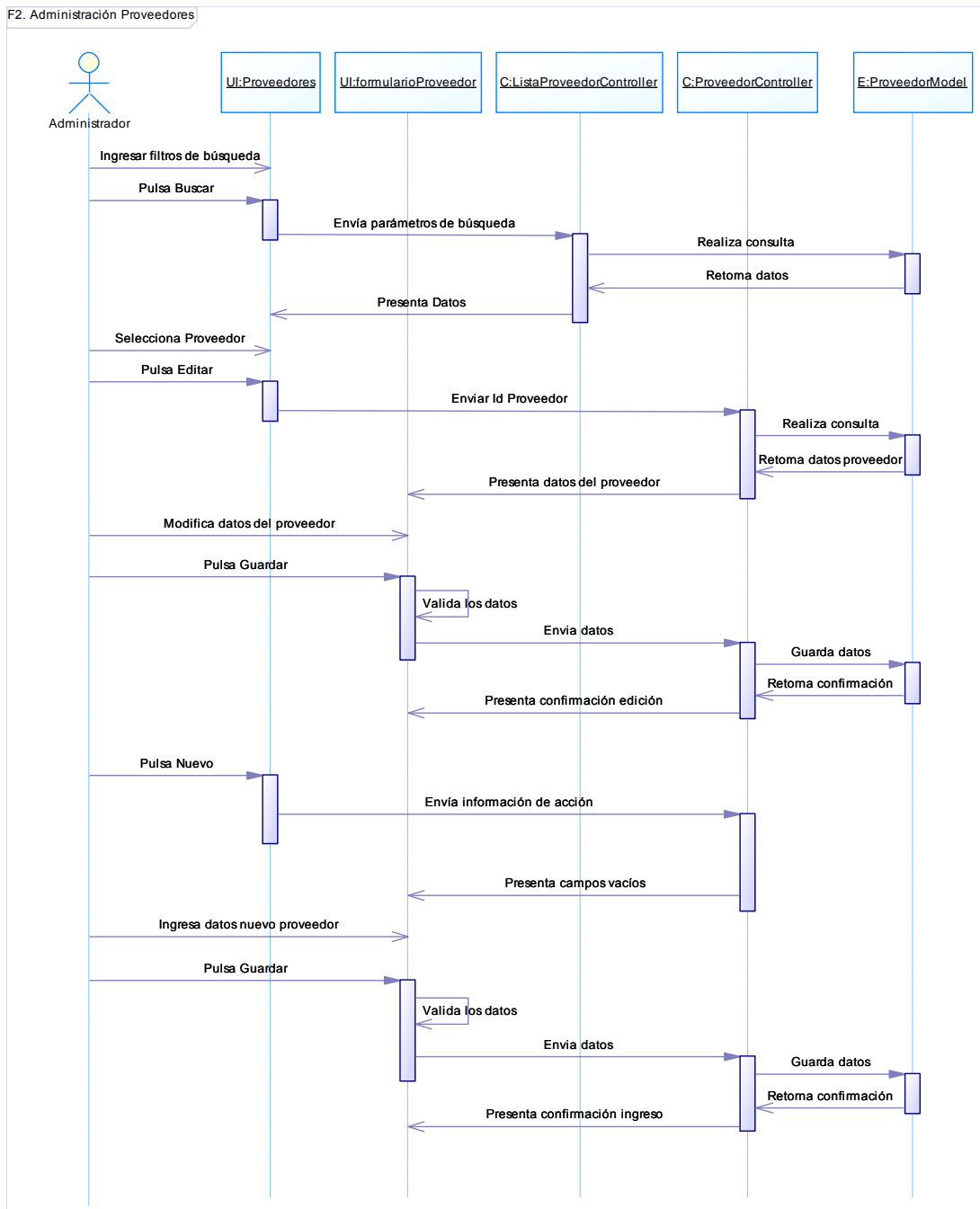


Figura IV – 12: Diagrama de Secuencia para la Administración de Proveedores

Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

Caso de Uso: Administración de Productos

Identificador: F3

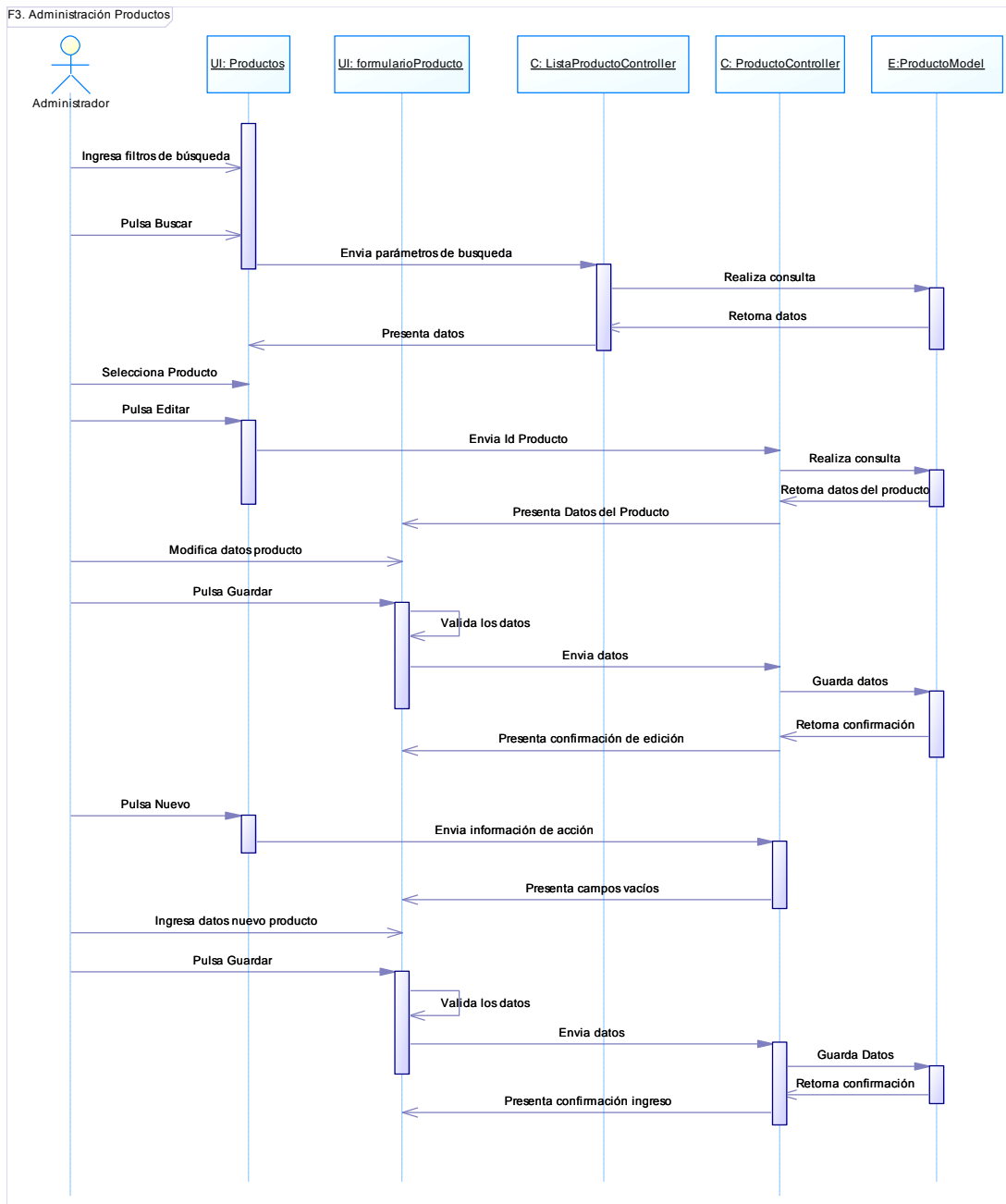


Figura IV – 13: Diagrama de Secuencia para la Administración de Productos

Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

Caso de Uso: Administración de Clientes

Identificador: F4

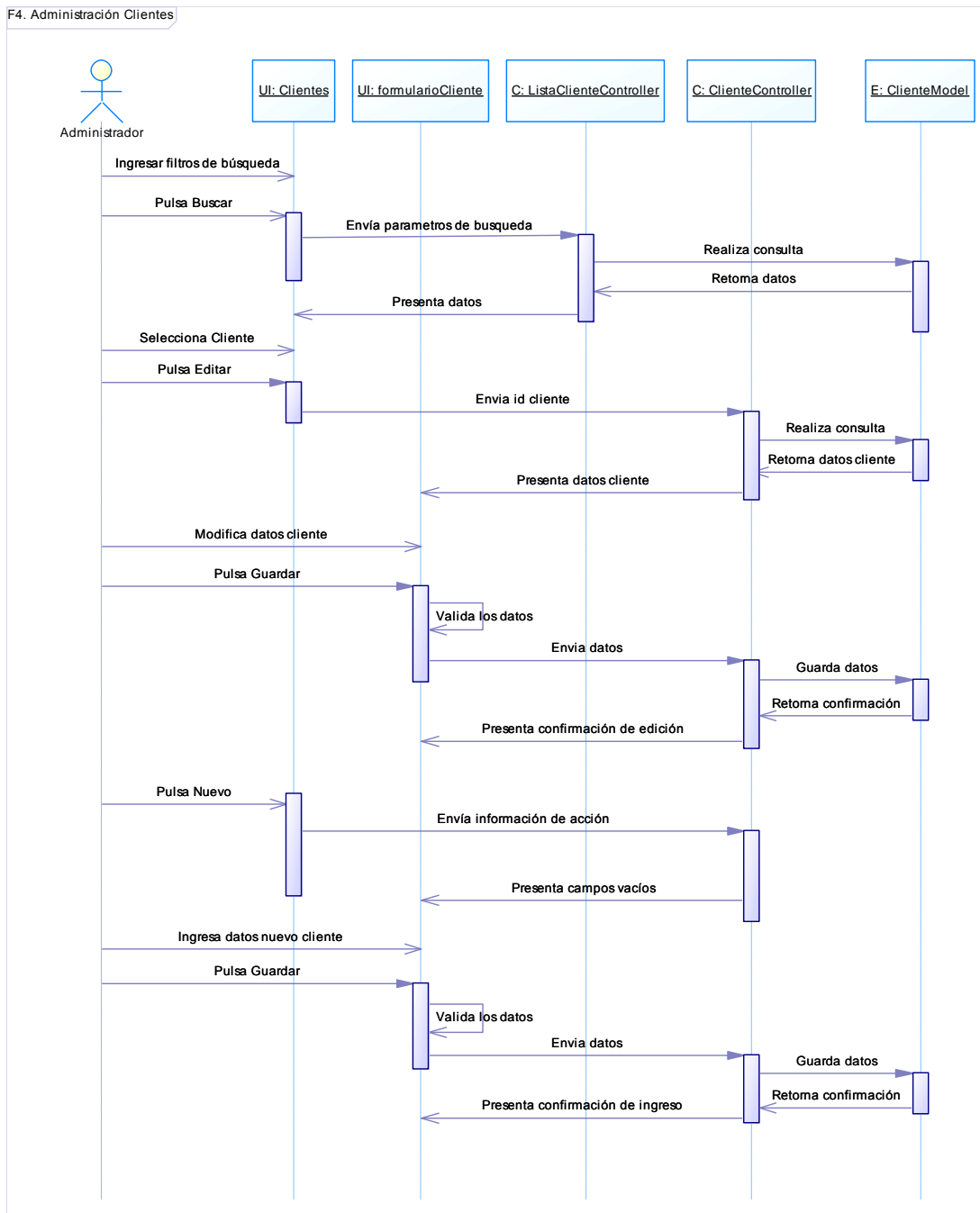


Figura IV – 14: Diagrama de Secuencia para la Administración de Clientes

Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.2.2. Diseño de Clases

El Diagrama de Clases es el diagrama principal para el análisis y diseño. Un diagrama de clases presenta las clases del sistema con sus relaciones estructurales y de herencia. La definición de clase incluye definiciones para atributos y operaciones. El diseño de las clases deben ser representadas en los tres modelos de interfaz, de control y de entidades.

##### 4.1.2.2.1. Diseño UI

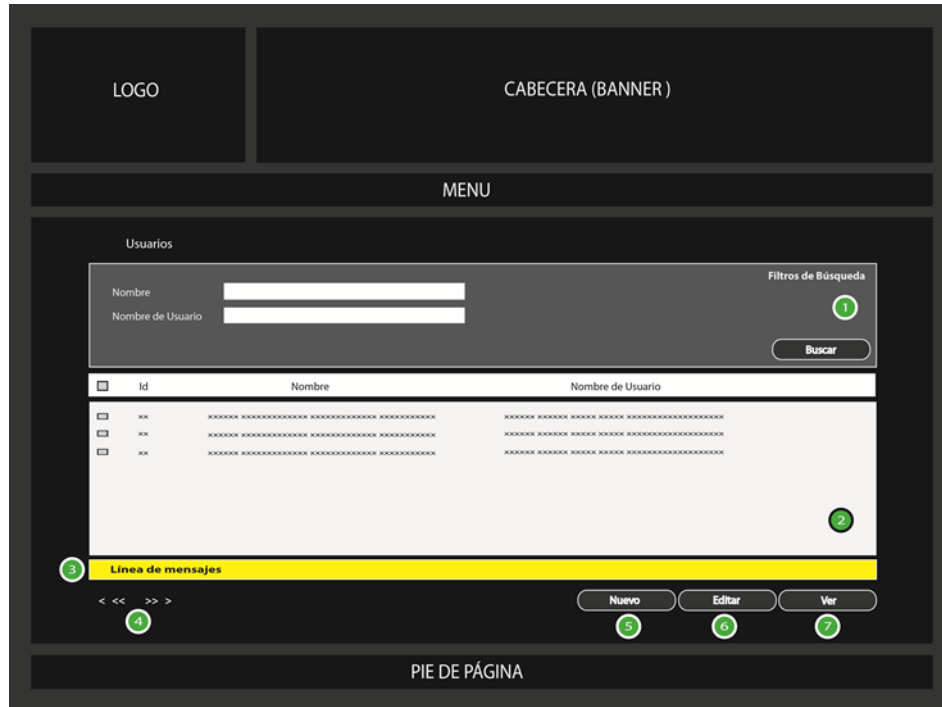
El diseño de las aplicaciones web es esencial antes de iniciar con la codificación del sitio. La generación de diferentes modelos según los cuales los programadores se basan para la codificación ayuda a la abstracción del código, permitiendo observar claramente las secciones del sitio y que secciones de código pueden llegar a ser reutilizadas.

Esta generación de los diferentes diseños permitirá a los clientes aprobar la vista gráfica previa su codificación, y, de ser necesario, solicitar cambios. Una vez aprobados los diseños no pueden (o no deben) ser modificados, pues los programadores, podrán codificar el sitio sin la necesidad de realizar modificaciones instantáneas y no documentadas. Cualquier cambio tras la aprobación de los diseños deberá ser realizado en una etapa posterior de retroalimentación.

**Caso de Uso:** Administración de Usuarios

**Identificador:** F1

**Interfaz:** UI\_F1\_01\_Usuarios

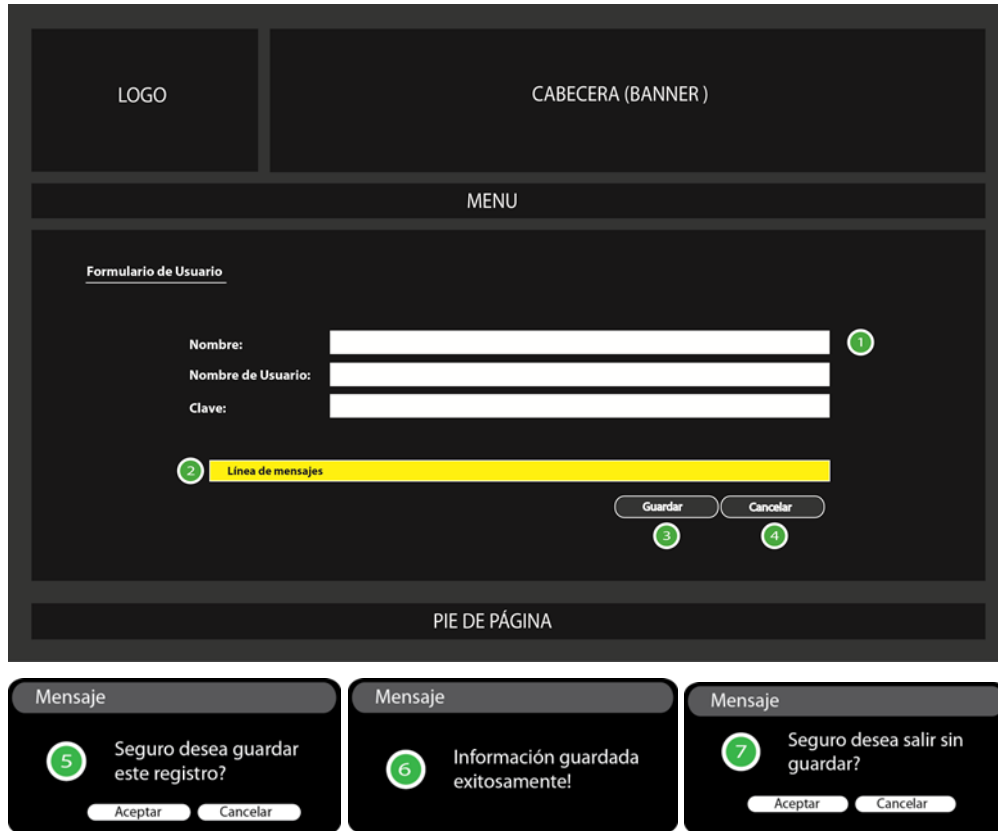


**Figura IV – 15:** Diseño de interfaz para la Administración de Usuarios

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

1. Filtros de Búsqueda. Área de filtros donde se ingresan los parámetros de búsqueda para luego presionar el botón “Buscar”, y el sistema presentara los resultados en la grilla.
2. Grilla. Componente que presenta el listado general o resultado de una búsqueda.
3. Línea de mensajes. Área de la interfaz donde se presentan los mensajes del sistema
4. Paginado. Componente que pagina los resultados evitando los scrolls.
5. Botón “Nuevo”. Al presionar el sistema mostrará el formulario de usuario con los campos vacíos, con permisos de edición.
6. Botón “Editar”. Habiendo seleccionado un registro, al presionar este botón el sistema mostrará el formulario de usuario con los datos del mismo, con permisos de edición.
7. Botón “Ver”. Habiendo seleccionado un registro al presionar este botón el sistema mostrará el formulario de usuario con los datos del mismo sin permisos de edición.

**Interfaz:** UI\_F1\_02\_FormularioUsuario



**Figura IV – 16:** Diseño de interfaz para la Administración de Usuarios

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

1. Datos del registro. Los campos están validados para que no se ingrese información incorrecta.
2. Línea de mensajes. Área de la interfaz donde se presentan los mensajes del sistema
3. Botón “Guardar”. El sistema mostrara el mensaje de confirmación de persistencia.
4. Botón “Cancelar”. El sistema mostrara el mensaje de confirmación de cancelación.
5. Caja de mensaje de confirmación de persistencia de un registro. Nos dará la opción de aceptar y guardar el registro o cancelar en caso de que sea necesario.
6. Caja de mensaje de confirmación de información guardada. Nos informará que la información se guardó exitosamente.
7. Caja de mensaje de confirmación de cancelación.  
Nos dará la opción de salir del ingreso del usuario o bien continuar con el registro

**Interfaz:** UI\_F2\_01\_Proveedores

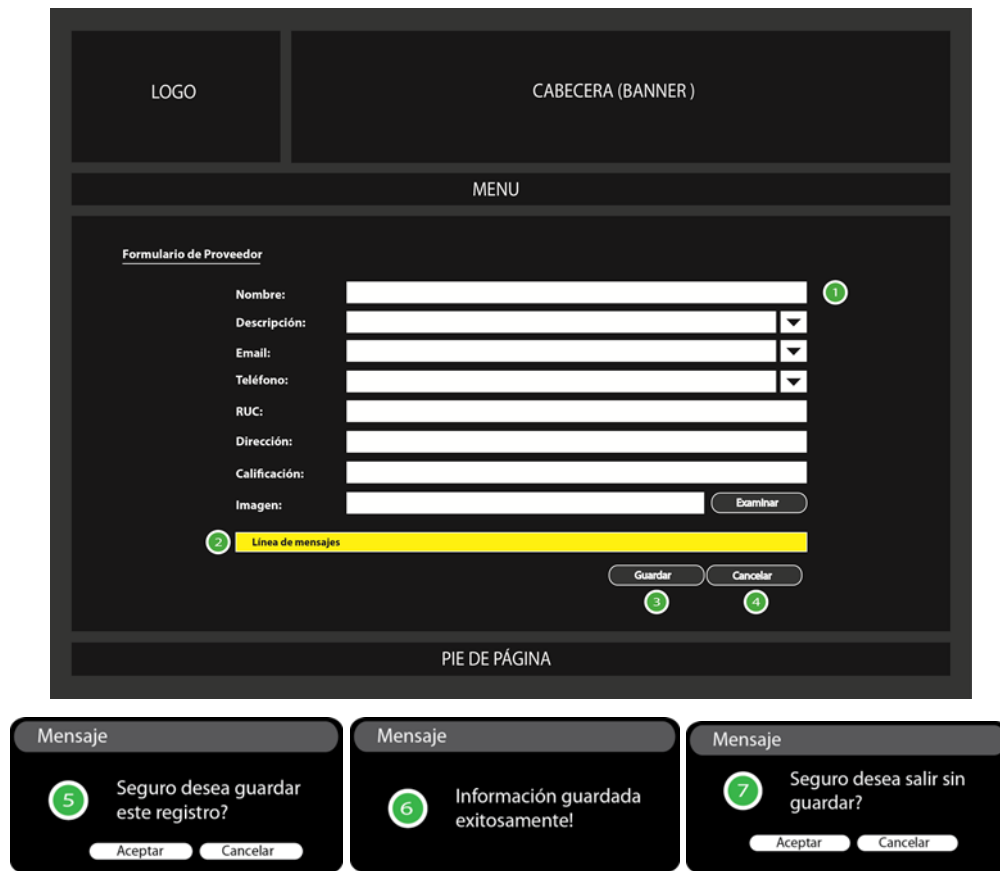


**Figura IV – 17:** Diseño de interfaz para la Administración de Proveedores

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

1. Filtros de Búsqueda. Área de filtros donde se ingresan los parámetros de búsqueda para luego presionar el botón “Buscar”, y el sistema presentara los resultados en la grilla.
2. Grilla. Componente que presenta el listado general o resultado de una búsqueda.
3. Línea de mensajes. Área de la interfaz donde se presentan los mensajes del sistema
4. Paginado. Componente que página los resultados evitando los scrolls.
5. Botón “Nuevo”. Al presionar el sistema mostrará el formulario de proveedor con los campos vacíos, con permisos de edición.
6. Botón “Editar”. Habiendo seleccionado un registro, al presionar este botón el sistema mostrará el formulario de proveedor con los datos del mismo, con permisos de edición.
7. Botón “Ver”. Habiendo seleccionado un registro al presionar este botón el sistema mostrará el formulario de proveedor con los datos del mismo sin permisos de edición.

**Interfaz:** UI\_F2\_02\_FormularioProveedor

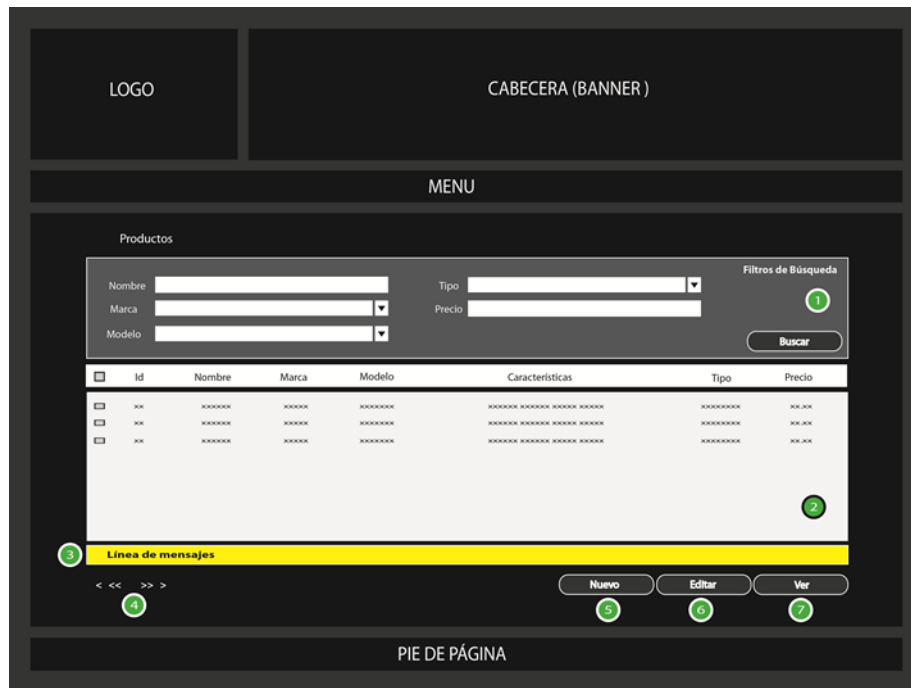


**Figura IV – 18:** Diseño de interfaz para la Administración de Proveedores

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

1. Datos del registro. Los campos están validados para que no se ingrese información incorrecta.
2. Línea de mensajes. Área de la interfaz donde se presentan los mensajes del sistema
3. Botón “Guardar”. El sistema mostrara el mensaje de confirmación de persistencia.
4. Botón “Cancelar”. El sistema mostrara el mensaje de confirmación de cancelación.
5. Caja de mensaje de confirmación de persistencia de un registro. Nos dará la opción de aceptar y guardar el registro o cancelar en caso de que sea necesario.
6. Caja de mensaje de confirmación de información guardada. Nos informará que la información se guardó exitosamente.
7. Caja de mensaje de confirmación de cancelación.  
Nos dará la opción de salir del ingreso del proveedor o bien continuar con el registro

**Interfaz:** UI\_F3\_01\_Productos

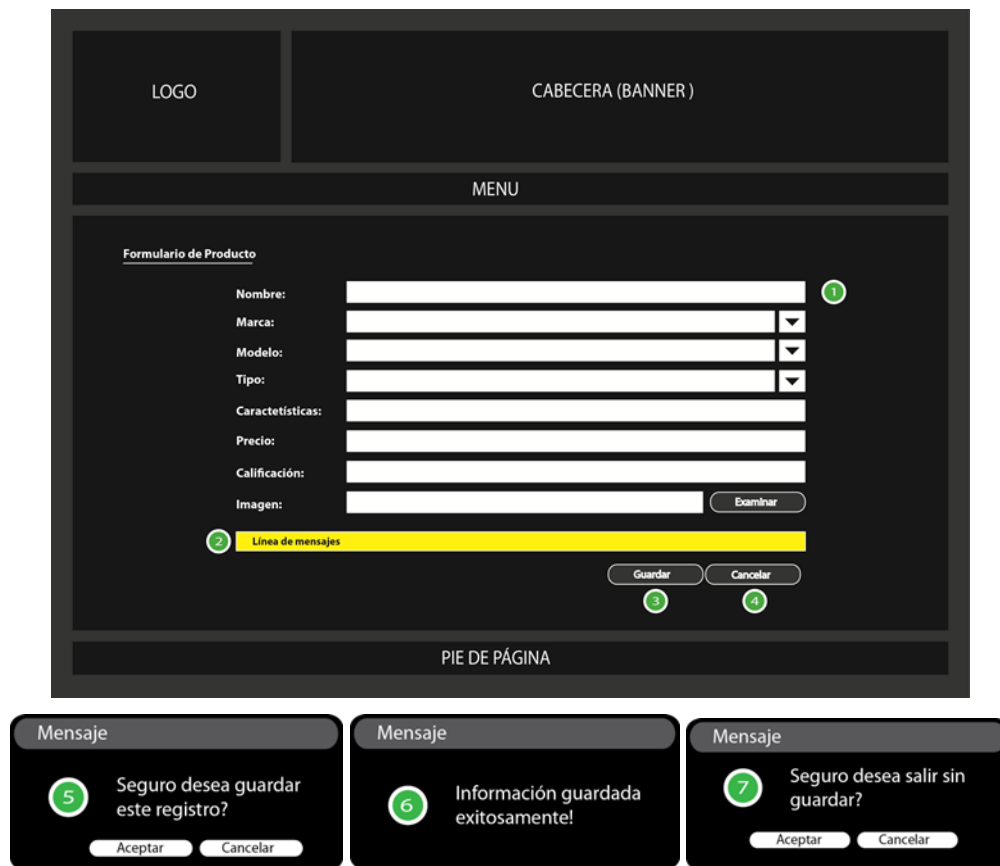


**Figura IV – 19:** Diseño de interfaz para la Administración de Productos

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

1. Filtros de Búsqueda. Área de filtros donde se ingresan los parámetros de búsqueda para luego presionar el botón “Buscar”, y el sistema presentara los resultados en la grilla.
2. Grilla. Componente que presenta el listado general o resultado de una búsqueda.
3. Línea de mensajes. Área de la interfaz donde se presentan los mensajes del sistema
4. Paginado. Componente que pagina los resultados evitando los scrolls.
5. Botón “Nuevo”. Al presionar el sistema mostrará el formulario de producto con los campos vacíos, con permisos de edición.
6. Botón “Editar”. Habiendo seleccionado un registro, al presionar este botón el sistema mostrará el formulario de producto con los datos del mismo, con permisos de edición.
7. Botón “Ver”. Habiendo seleccionado un registro al presionar este botón el sistema mostrará el formulario de producto con los datos del mismo sin permisos de edición.

**Interfaz:** UI\_F3\_02\_FormularioProducto



**Figura IV – 20:** Diseño de interfaz para la Administración de Productos

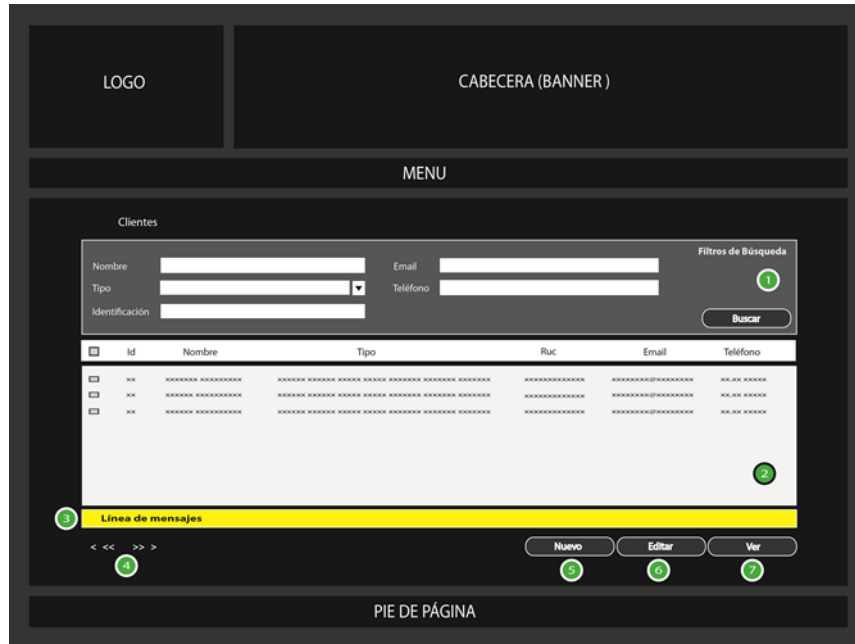
**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

1. Datos del registro. Los campos están validados para que no se ingrese información incorrecta.
2. Línea de mensajes. Área de la interfaz donde se presentan los mensajes del sistema
3. Botón “Guardar”. El sistema mostrara el mensaje de confirmación de persistencia.
4. Botón “Cancelar”. El sistema mostrara el mensaje de confirmación de cancelación.
5. Caja de mensaje de confirmación de persistencia de un registro. Nos dará la opción de aceptar y guardar el registro o cancelar en caso de que sea necesario.
6. Caja de mensaje de confirmación de información guardada. Nos informará que la información se guardó exitosamente.
7. Caja de mensaje de confirmación de cancelación.  
Nos dará la opción de salir del ingreso del producto o bien continuar con el registro

**Caso de Uso:** Administración de Clientes

**Identificador:** F4

**Interfaz:** UI\_F4\_01\_Clientes

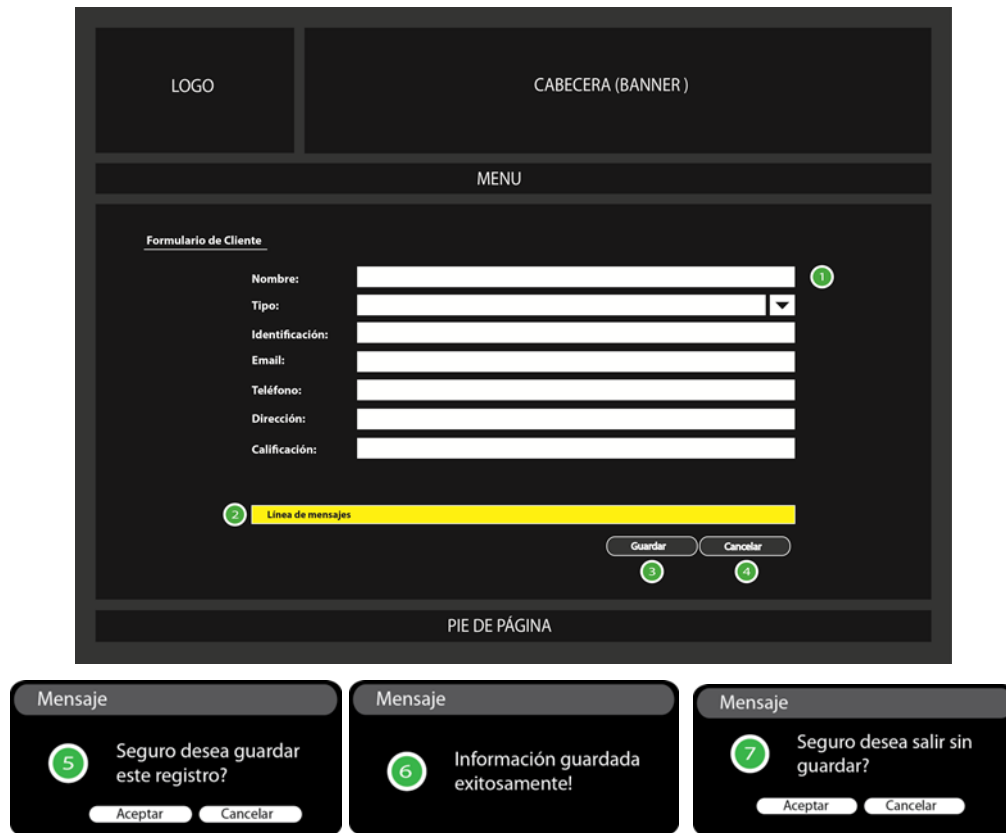


**Figura IV – 21:** Diseño de interfaz para la Administración de Clientes

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

1. Filtros de Búsqueda. Área de filtros donde se ingresan los parámetros de búsqueda para luego presionar el botón “Buscar”, y el sistema presentara los resultados en la grilla.
2. Grilla. Componente que presenta el listado general o resultado de una búsqueda.
3. Línea de mensajes. Área de la interfaz donde se presentan los mensajes del sistema
4. Paginado. Componente que pagina los resultados evitando los scrolls.
5. Botón “Nuevo”. Al presionar el sistema mostrará el formulario de cliente con los campos vacíos, con permisos de edición.
6. Botón “Editar”. Habiendo seleccionado un registro, al presionar este botón el sistema mostrará el formulario de cliente con los datos del mismo, con permisos de edición.
7. Botón “Ver”. Habiendo seleccionado un registro al presionar este botón el sistema mostrará el formulario de cliente con los datos del mismo sin permisos de edición.

**Interfaz:** UI\_F4\_02\_FormularioCliente



**Figura IV – 22:** Diseño de interfaz para la Administración de Clientes

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

1. Datos del registro. Los campos están validados para que no se ingrese información incorrecta.
2. Línea de mensajes. Área de la interfaz donde se presentan los mensajes del sistema
3. Botón “Guardar”. El sistema mostrara el mensaje de confirmación de persistencia.
4. Botón “Cancelar”. El sistema mostrara el mensaje de confirmación de cancelación.
5. Caja de mensaje de confirmación de persistencia de un registro. Nos dará la opción de aceptar y guardar el registro o cancelar en caso de que sea necesario.
6. Caja de mensaje de confirmación de información guardada. Nos informará que la información se guardó exitosamente.
7. Caja de mensaje de confirmación de cancelación.  
Nos dará la opción de salir del ingreso del cliente o bien continuar con el registro

4.1.2.2.2. Diseño de Clases de Control

El diseño de las clases de control son aquellas que tienen que ver directamente con el negocio y como el mismo interactúa con los procesos cotidianos de la empresa. Estas clases dependiendo de la complejidad se los pueden definir a través de algoritmos o arboles de decisión. En esta oportunidad, y considerando que los procesos especificados por el cliente son sencillos se tratara de expresarlo en la Tabla IV – 5.

<b>Caso de Uso</b>	<b>Control</b>	<b>Nombre</b>	<b>Descripción</b>
<b>F1</b>	C_F1_01_UsuarioController	UsuarioController	Realiza las funciones de ingreso y modificación de Usuarios
<b>F1</b>	C_F1_02_ListaUsuarioController	ListaUsuarioController	Realiza búsquedas de Usuarios por parámetros y devuelve resultados en listados.
<b>F2</b>	C_F2_01_ProveedorController	ProveedorController	Realiza las funciones de ingreso y modificación de Proveedores
<b>F2</b>	C_F2_02_ListaProveedorController	ListaProveedorController	Realiza búsquedas de Proveedores por parámetros y devuelve resultados en listados.
<b>F3</b>	C_F3_01_ProductoController	ProductoController	Realiza las funciones de ingreso y modificación de Productos
<b>F3</b>	C_F3_02_ListaProductoController	ListaProductoController	Realiza búsquedas de Productos por parámetros y devuelve resultados en listados.
<b>F4</b>	C_F4_01_ClienteController	ClienteController	Realiza las funciones de ingreso y modificación de Clientes
<b>F4</b>	C_F4_02_ListaClienteController	ListaClienteController	Realiza búsquedas de

Clientes por parámetros y devuelve resultados en listados.

Tabla IV – 5: Diseño de Clases de Control

Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

4.1.2.2.3. Diseño de Clases Entidad

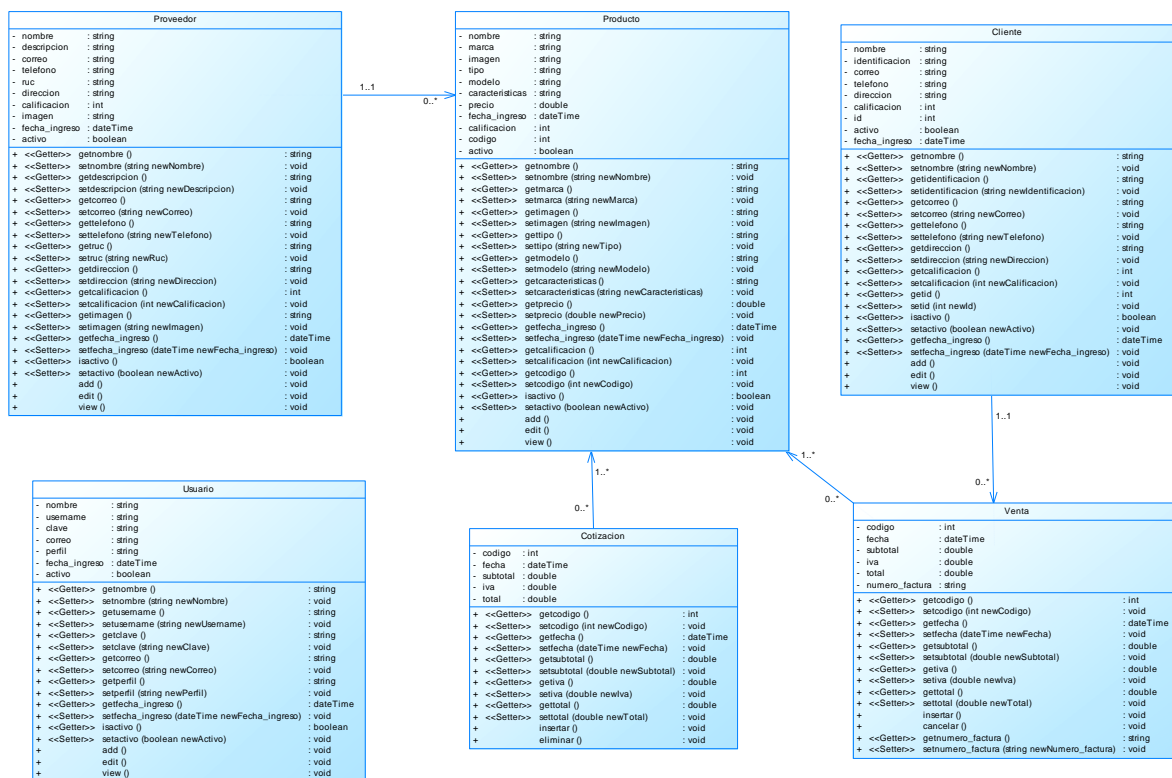


Figura IV – 6: Diseño de Clases Entidad

Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

4.1.2.3. Plan de Pruebas

4.1.2.3.1 Introducción

Acorde con el enfoque del desarrollo de la solución, el plan de pruebas está basado en la metodología de Rational Unified Process (RUP), lo que hace que este plan de pruebas tenga como propósito establecer las técnicas, herramientas y actividades relacionadas con la ejecución y

validación de cada una de las pruebas, incluyendo responsabilidades de cada una de las actividades, los recursos y los prerrequisitos que deben ser considerados en el esfuerzo de cada una de las pruebas; lo anterior permite garantizar el cumplimiento de los requerimientos planteados en el marco del desarrollo del proyecto.

Este documento de plan de pruebas, se convierte en una guía para desarrollar de una forma organizada las diferentes actividades que se realizarán en el proceso del plan de pruebas en el desarrollo del proyecto.

A continuación se describen las diferentes pruebas a ser aplicadas:

Tipo de Prueba	Definición	Ambiente
<b>Sistema</b>	<p><b>Sistema:</b> Estas pruebas buscan diferencias entre la solución desarrollada y los requerimientos, con el fin de identificar errores que se puedan generar entre la especificación funcional y el diseño del sistema.</p> <p><b>Interfaz de usuario:</b> Permite verificar que la navegación a través de los elementos que se están probando, reflejen las funciones del negocio y los requerimientos funcionales.</p> <p><b>Rendimiento:</b> Permite validar si la aplicación cumple los criterios de tiempos de respuesta establecidos.</p>	<b>De Desarrollo</b>
<b>Funcionales</b>	<p><b>Usabilidad:</b> Esta prueba permite encontrar problemas de factores humanos, o usabilidad.</p>	<b>De Usuario</b>

**Tabla IV – 6:** Tipos de Prueba

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

Definiciones, acrónimos y abreviaturas

- **El plan de prueba:** describe todos los métodos que se utilizarán para verificar que el software satisface la especificación del producto y las necesidades del cliente.
- **Reporte de pruebas:** describen los problemas encontrados al ejecutar los casos de prueba.
- **Herramientas de pruebas y automatización:** documentación de las herramientas empleadas en el proceso de pruebas.
- **Métricas, estadísticas y resúmenes:** indican cómo ha sido el progreso del proceso de prueba.

### Vista General

Descripción resumida de contenido de cada una de las secciones que siguen, y explicación de la forma en que está organizado el presente documento.

#### **Estrategia de Pruebas:**

Se presenta una perspectiva general de la estrategia que se va a seguir para analizar, diseñar, implementar y ejecutar las pruebas del proyecto. Así mismo se definirá qué tipos de pruebas se van a realizar y cómo se ejecutarán.

#### **Recursos del Plan de Pruebas:**

Se identifica los recursos humanos y no humanos (hardware, software, herramientas de soporte, configuración de entorno de pruebas, entre otros), necesarios para desarrollar el proceso del plan de pruebas.

#### **Evaluación de Pruebas Ejecutadas:**

Se describe de los métodos de evaluación de las pruebas ejecutadas, de tal forma que permitirá evaluar los grados de aceptación de las pruebas.

#### **Formularios de las Pruebas:**

Se describen los formularios que se utilizarán para la especificación y la documentación de la ejecución de las pruebas.

### 4.1.2.3.2 Estrategia de Pruebas

#### Ciclo de pruebas

El ciclo de pruebas comprende seis actividades las cuales deberán ser desarrolladas de la siguiente manera:

#### **Planificación**

Para el desarrollo del proyecto, se considera de gran importancia la ejecución del plan de pruebas, haciéndose necesario la planificación de las mismas, lo que en consecuencia hace necesario tener claro los siguientes planteamientos:

- Se planifican pruebas personalizando los estándares.
- Se establece el tiempo para la ejecución de cada una de las pruebas.
- Uso de herramientas.
- Criterios de aceptación.
- Recursos involucrados.

En la definición del plan de pruebas, se valorará:

- El alcance de la aplicación.
- La complejidad de sus procesos.
- Conocimientos y formación de quienes ejecutarán las pruebas.

Se tendrá en cuenta que:

- Las pruebas estarán presentes a lo largo de todo el ciclo de vida del desarrollo, de la solución.
- Siempre hay errores.
- Probar exhaustivamente el software es imposible.
- No es recomendable que el programador pruebe sus propios programas.
- Se puede disponer de herramientas.

Resultado de la planificación:

- Cronograma detallado de la ejecución de las pruebas.
- Formatos a utilizar para el diseño de las pruebas.
- Formatos a utilizar para el registro y análisis de los resultados de las pruebas.

### **Diseño de las pruebas:**

Para el diseño de las pruebas, se tendrán en cuenta aspectos que permitirán encontrar defectos en el periodo de desarrollo del software, la realización de pruebas propias de verificación y validación de datos, según se aclara en los siguientes ítems:

- A. Alcance: El alcance de las pruebas estará dado por el marco del sistema que se encuentra en desarrollo.
- B. Inventario de las pruebas: En esta sección se especifica el inventario de las pruebas, el cual permitirá:
  - Definir y asignar prioridades como; alta, media o baja.
  - Establecer un orden de trabajo.

- Se estima el tiempo en probar cada funcionalidad.
  - Evaluar aspectos técnicos del sistema.
- C. Resultado de la ejecución de las pruebas: En este punto se resaltan las entradas fundamentales que son la partida para la ejecución del plan de pruebas.
- D. Ciclo de la prueba: Las actividades de la prueba se realizarán para una determinada versión del producto, sobre la cual se ejecutan las pruebas y se reportan los incidentes encontrados.

#### **Configuración:**

Se establece el mantenimiento e integridad del software a través del ciclo de vida del proyecto y se proveen contextos de trabajo estables para los posibles cambios antes de ser entregado formalmente en producción.

#### **Evaluación y cierre:**

Para la evaluación y cierre de las pruebas se presentará el informe de pruebas donde se documentará el resultado de cada una de las diferentes pruebas ejecutadas.

#### **Seguimiento y control:**

Para el seguimiento y control de las pruebas se evaluarán los siguientes temas:

- Avance de las pruebas según cronograma
- Estado o resultado de las pruebas ejecutadas
- Seguimiento a las incidencias reportadas según la ejecución de pruebas.

#### 4.1.2.3.3 Recursos del plan de pruebas

- Recurso Humano
- Recurso del sistema

#### 4.1.2.3.4 Evaluación de pruebas ejecutadas

A continuación se mostrarán los criterios de ejecución, evaluación, terminación y suspensión de las pruebas.

##### ➤ Criterios de inicio de ejecución

A continuación se señalan las condiciones mínimas que se deben presentar para iniciar la ejecución de las pruebas:

- Se poseen los set de pruebas aprobados con escenarios claros.

- El entorno de pruebas es el adecuado para el tipo de pruebas a iniciar.
- Todos los artefactos requeridos se encuentran disponibles.
- Se recibió la Versión del Software para pruebas.
- Todos los recursos humanos y técnicos necesarios se encuentran disponibles.

➤ Criterios de evaluación

Los criterios de evaluación estarán dados de forma independiente para cada tipo de pruebas; el siguiente cuadro muestra los criterios de evaluación generales de las pruebas ejecutadas.

TIPO DE PRUEBAS	CRITERIOS DE EVALUACION
<b>Pruebas de Sistema</b>	<ul style="list-style-type: none"> <li>• Detectar errores en la ejecución de las pruebas</li> <li>• Tener en cuenta todos los escenarios posibles.</li> <li>• El 90% de las pruebas realizadas deben ser exitosas.</li> </ul>
<b>Pruebas Funcionales</b>	<ul style="list-style-type: none"> <li>• El resultado de cada caso de prueba debe ser igual al resultado de salida esperado.</li> <li>• Encontrar fallas al ejecutar los diferentes casos de uso.</li> <li>• La aplicación cumple con los requerimientos funcionales especificados en la fase de análisis</li> <li>• La aplicación cumple con los requerimientos mínimos para el funcionamiento</li> </ul>
<b>Pruebas de Usabilidad</b>	<ul style="list-style-type: none"> <li>• El resultado de cada caso de prueba debe ser igual al resultado de salida esperado.</li> <li>• Se deben incluir todos los resultados válidos y esperados, aun cuando no hayan sido exitoso este resultado.</li> <li>• La aplicación debe cumplir con los requerimientos funcionales especificados en la fase de análisis.</li> </ul>

**Tabla IV – 7:** Tipos de Prueba y Criterios de Evaluación

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

➤ Criterios de terminación

A continuación se señalan los criterios de terminación de las pruebas a ejecutar.

- Se ejecutaron todas las pruebas del sistema.

- Todas las pruebas se ejecutaron de acuerdo a los criterios de evaluación.
- Las pruebas de carga demuestran que se posee un grado satisfactorio de capacidad operativa y funcional.
- Los incidentes encontrados en las pruebas fueron corregidos y probados.

➤ Criterios de suspensión

Los criterios de suspensión impiden la iniciación y/o continuación de las pruebas ante cualquier situación de imprevisto que hace que la ejecución de las pruebas no logre grados satisfactorios de probabilidad de éxito.

- Después de la instalación y configuración del sistema, se evidencia problemas o situaciones anormales en cualquiera de sus componentes.
- Después de la instalación y configuración del sistema, se evidencia que el ambiente de pruebas no es lo suficientemente estable para la ejecución de las pruebas.
- Discrepancia entre la documentación (Set de Pruebas, Casos de Uso) y el sistema.

#### 4.1.2.3.5 Formularios de las Pruebas

A continuación se listarán los anexos al plan de pruebas, que básicamente corresponden a todos los documentos, formatos o plantillas que se utilizarán en la especificación, ejecución y documentación de resultados de las pruebas.

#### Formato Pruebas Funcionales

A continuación se presenta el formato que se utilizará como Set de Pruebas funcionales.

INFORMACIÓN GLOBAL DE LA PRUEBA			
<b>CASO DE USO:</b>	<b>&lt;Número del caso de prueba constituido por SP-[numero del caso de uso]-[Numero del caso de prueba]&gt;</b>	<b>VERSIÓN DE EJECUCIÓN N</b>	<b>&lt;Versión diligenciado por el analista de pruebas en el momento de ejecutarla. Este número se incrementa de 1 en 1&gt;</b>

	<Identificación del caso de uso objeto de la prueba>	<b>FECHA EJECUCIÓN</b>	<Fecha de ejecución diligenciado por el analista de pruebas>	
		<b>MODULO DEL SISTEMA</b>	<Nombre del modulo al que corresponde el caso de uso objeto de la prueba>	
<b>Descripción del caso de prueba:</b>	<Descripción de lo que se pretende probar en el caso de prueba>			
<b>1. CASO DE USO</b>				
<b>a. Precondiciones</b>				
<Lista de precondiciones que deben cumplirse para realizar la prueba>				
<b>b. Pasos de la prueba</b>				
PASOS	RESPUESTA ESPERADA DE LA APLICACIÓN	COINCIDE		RESPUESTA DEL SISTEMA
		SI	NO	
<Pasos secuenciales que deben ser ejecutados por el analista de pruebas o usuario, ante el sistema para ejecutar la prueba>	<Respuesta que se espera de la aplicación>			<Respuesta que se obtuvo de la aplicación en el momento de la ejecución de la prueba>
<b>c. Post condiciones</b>				
<Lista de poscondiciones que deben cumplirse después de realizar la prueba>				
<b>2. RESULTADOS DE LA PRUEBA</b>				
<b>Defectos y desviaciones</b>			<b>Veredicto</b>	
<Lista de defectos o desviaciones encontrados por el analista o usuario al ejecutar la prueba>			<input type="checkbox"/> Paso <input checked="" type="checkbox"/> Falló	

Observaciones / Modificaciones Sugeridas	Probador
<Observaciones generales del analista o usuario sobre la ejecución de la prueba>	
	<b>Firma:</b> <b>Nombre:</b> <b>Fecha:</b>

**Tabla IV – 7:** Formato Caso de Prueba Funcionales  
**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

Encuesta para pruebas de usabilidad

Las pruebas de usabilidad se guiarán por la siguiente estructura de encuesta:

INFORMACIÓN GLOBAL DE LA PRUEBA		
<b>VERSIÓN DE EJECUCIÓN</b>	<Versión diligenciado por el analista de pruebas en el momento de ejecutarla. Este número se incrementa de 1 en 1>	
<b>FECHA EJECUCIÓN</b>	<Fecha de ejecución diligenciado por el analista de pruebas>	
<b>MODULO DEL SISTEMA</b>	<Nombre del módulo al que corresponde el caso de uso objeto de la prueba>	
1. PRUEBA DE USABILIDAD		
PREGUNTA	CRITERIOS DE EVALUACIÓN	RESPUESTA
1. ¿Es simple el vocabulario utilizado?	1 = El vocabulario es demasiado técnico. 2 = El vocabulario presenta algunas dificultades de comprensión. 3 = El vocabulario es completamente comprensible.	
2. ¿Hay algún tipo de asistencia para los usuarios que hacen uso del sistema por primera vez?	1 = No existe ninguna ayuda. 2 = Se encuentra ayuda en algunas partes. 3 = Existen ayudas en todo el sistema.	
3. ¿El sistema es fácil de operar para alguien que no recibió capacitación en su operación?	1 = El sistema es de difícil comprensión. 2 = El sistema es fácil de operar en algunas de sus funcionalidades. 3 = El sistema es completamente fácil de operar.	
4. ¿Se entienden la interfaz y su contenido?	1 = No se entiende su interfaz. 2 = La interfaz se entiende en algunas partes. 3 = La interfaz es completamente entendible.	

<p>5. ¿Resulta fácil identificar un objeto o una acción?</p>	<p>1 = Es difícil identificar los objetos o acciones.                  2 = Se pueden identificar los objetos y acciones en algunas partes del sistema.                  3 = Todos los objetos y acciones son fácilmente identificables.</p>	
<p>6. ¿Resulta fácil entender el resultado de una acción?</p>	<p>1 = Los resultados de las acciones no son entendibles.                  2 = Los resultados de las acciones son entendibles en algunas partes o la mayor parte del sistema.                  3 = Todos los resultados de las acciones son entendibles.</p>	
<p>7. ¿Está diseñada la interfaz para facilitar la realización eficiente de las tareas de la mejor forma posible?</p>	<p>1 = La interfaz es difícil de usar.                  2 = La interfaz es difícil de usar en algunas partes del sistema.                  3 = La interfaz es completamente sencilla de usar.</p>	
<p>8. ¿Son apropiados los mensajes presentado por el sistema?</p>	<p>1 = Los mensajes non son apropiados.                  2 = Los mensajes son apropiados en algunas partes del sistema.                  3 = Todos los mensajes son apropiados y fáciles de comprender.</p>	
<p>9. ¿Actúa el sistema en la prevención de errores?</p>	<p>1 = El sistema no previene errores del usuario.                  2 = El sistema previene algunos o la mayoría de los errores del usuario.                  3 = El sistema previene cualquier error que pueda cometer el usuario.</p>	
<p>10. ¿El sistema informa claramente sobre los errores presentados?</p>	<p>1 = El sistema no informa de manera adecuada sobre los errores cometidos.                  2 = El sistema informa de manera adecuada algunos o la mayoría de los errores cometidos por el usuario.</p>	

	3 = El sistema informa de forma adecuada todos los errores cometidos por el usuario.	
11. ¿Se utiliza mensajes y textos descriptivos?	1 = Los mensajes de texto no son descriptivos. 2 = La mayoría de los textos son descriptivos o fáciles de interpretar 3 = Todos los textos son descriptivos o fáciles de interpretar.	
12. ¿Se proporciona información visual de dónde está el usuario, qué está haciendo y qué puede hacer a continuación?	1 = No se presenta ninguna información visual ni otro tipo de ayuda. 2 = Presenta ayudas en algunas partes del sistema. 3 = Las ayudas son apropiadas y están distribuidas a lo largo del sistema.	
<b>Observaciones / Modificaciones Sugeridas</b>		<b>Probador</b>
<a href="#">&lt;Observaciones generales del analista o usuario sobre la ejecución de la prueba&gt;</a>		
		<b>Firma:</b>
		<b>Nombre:</b>
		<b>Fecha:</b>

**Tabla IV – 8:** Encuesta para pruebas de usabilidad

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

Lista de Chequeo

A continuación se presenta el formato que se utilizará para lista de chequeo de las pruebas ejecutadas.

TIPO DE PRUEBA	Versión de Ejecución	Fecha de Ejecución	EJECUTADA	CUMPLE	NO CUMPLE	Observaciones


**Tabla IV – 9:** Lista de Chequeo de Pruebas

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

### 4.1.3. CONSTRUCCIÓN Y PRUEBAS

En la fase de construcción y pruebas, se desea comentar la experiencia que los desarrolladores tuvieron con las herramientas de construcción, puesto que en el Capítulo 3 (sección 3.2.3. Requerimientos no funcionales) se definieron todas las herramientas para cada fase, en esta etapa se verá además las transformaciones que se realizaron para ir del Modelo E/R al Modelo Relacional y con esto generar el Modelo Conceptual y el Modelo físico del sistema.

Por otro lado se creará el diccionario de datos con cada uno de los atributos del programa, así como de un catálogo de las piezas del sistema, de tal forma que sea muy sencillo mirar la estructura de la herramienta. Es necesario definir las estrategias y los estándares de programación con el fin de normalizar el código fuente y también facilitar la generación del mismo para los programadores.

Al final de la fase, se pondrá en ejecución las pruebas en ambiente de desarrollo, puesto que las pruebas de usabilidad se realizarán en la siguiente fase de implantación.

#### 4.1.3.1. Descripción de Herramientas

##### **Framework Codeigniter**

- El uso de este framework ha sido muy satisfactorio, primero porque de una manera muy sencilla utiliza el patrón Modelo/Vista/Controlador, y de esta forma el código es más compacto.
- Incentiva a trabajar con programación orientada a objetos, y se pudo con mucha facilidad reutilizar el código en múltiples módulos del sistema y ahorrar tiempo.
- Codeigniter permitió generar fácilmente páginas con URL entendibles y amigables para los buscadores.

##### **Eclipse Helios – IDE**

- Este IDE, ha facilitado el desarrollo del aplicativo, pues de una manera muy sencilla identifica errores, como variables no declaradas, o llamadas a funciones incorrectas.
- Eclipse al proporcionar una vista de jerarquía de paquetes de clase, campos de clase y los métodos vista de esquema hace que sea más fácil de navegar.

##### **Mysql Workbench**

- El panel de conexiones que proporciona Mysql Workbench es muy útil para los desarrolladores, pues permite mirar todos los conectores de la base de datos de una manera rápida y efectiva.
- El Examinador de objetos permite el acceso instantáneo a esquemas de bases de datos y objetos.
- MySQL Workbench proporciona una consola visual para administrar fácilmente entornos MySQL y obtener una mejor visibilidad en bases de datos. Los desarrolladores pueden utilizar las herramientas visuales para la configuración de los servidores, la administración de usuarios, bases de datos de forma muy recursiva.

#### 4.1.3.2. Transformación Modelo E/R al Modelo Relacional

A continuación las reglas que se usaron para transformas el Modelo Entidad/ Relación en Modelo Relacional.

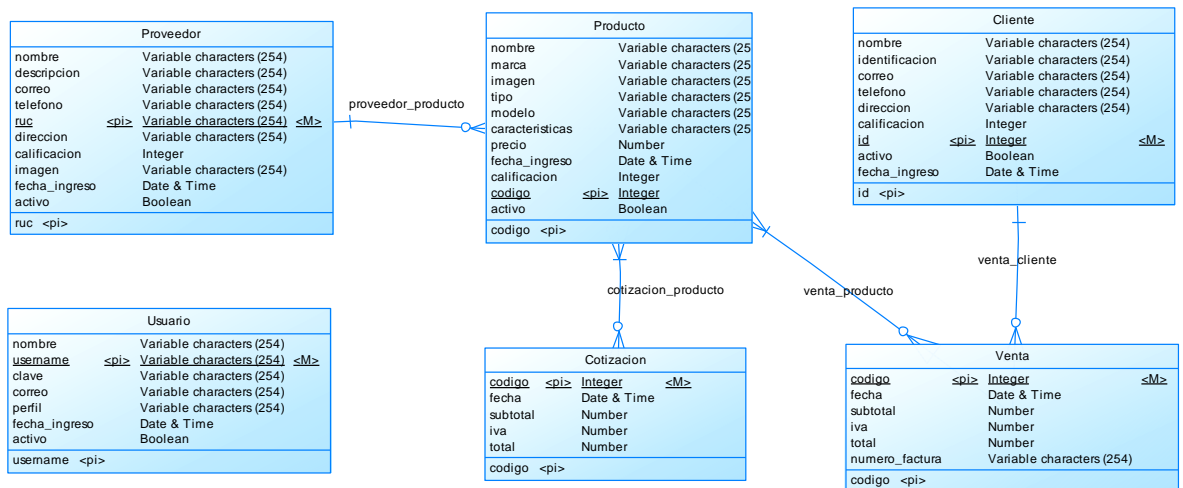
Modelo E/R	Modelo Relacional
<b>Entidades</b>	Tablas
<b>Atributo</b>	Campo (manteniendo claves primarias)
<b>Relaciones N:M</b>	Tabla (clave primaria la concatenación de los atributos clave de las entidades que relaciona)
<b>Relación 1:N</b>	<ul style="list-style-type: none"> <li>• Si la entidad que participa con cardinalidad máxima uno lo hace también con cardinalidad mínima uno, entonces se propaga el atributo de la entidad que tiene cardinalidad máxima n a la que tiene cardinalidad máxima 1, desapareciendo el nombre de la relación. Si existen atributos en la relación éstos también se propagarán.</li> <li>• Si la entidad que participa con cardinalidad máxima uno lo hace también cardinalidad mínima cero, entonces se crea una nueva tabla formada por las claves de cada entidad y los atributos de la relación. La clave primaria de la nueva tabla será el identificador de la entidad que participa con cardinalidad máxima 1.</li> </ul>

<p><b>Relación 1:1</b></p>	<ul style="list-style-type: none"> <li>• Si las entidades poseen cardinalidades (0,1), la relación se convierte en una tabla.</li> <li>• Si una de las entidades posee cardinalidad (0,1) y la otra (1,1), conviene propagar la clave a la entidad con cardinalidad (1,1) de la tabla resultante de la entidad con cardinalidad (0,1). Si ambas entidades poseen cardinalidades (1,1) se puede propagar la clave de cualquiera de ellas a la tabla resultante de la otra.</li> </ul>
----------------------------	--

**Figura IV – 10:** Reglas de Transformación

Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

4.1.3.3. Modelo Conceptual



**Figura IV – 23:** Modelo Conceptual

Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

4.1.3.4. Modelo Físico

Consideraciones tomadas para la transformación del modelo conceptual al modelo físico:

- Colocar las multiplicidades entre las clases
- Identificar los atributos de Enlace o Clase de Enlace de las asociaciones de muchos a muchos
- Identificar las clases que tendrán sus propios atributos

- NO INCLUIR los Atributos identificadores de la clase (Se agregan en el modelo Físico)
- Incluir los atributos de las clases que se necesitan para satisfacer los requerimientos del sistema
- Verificar que las Reglas del Negocio se sigan cumpliendo

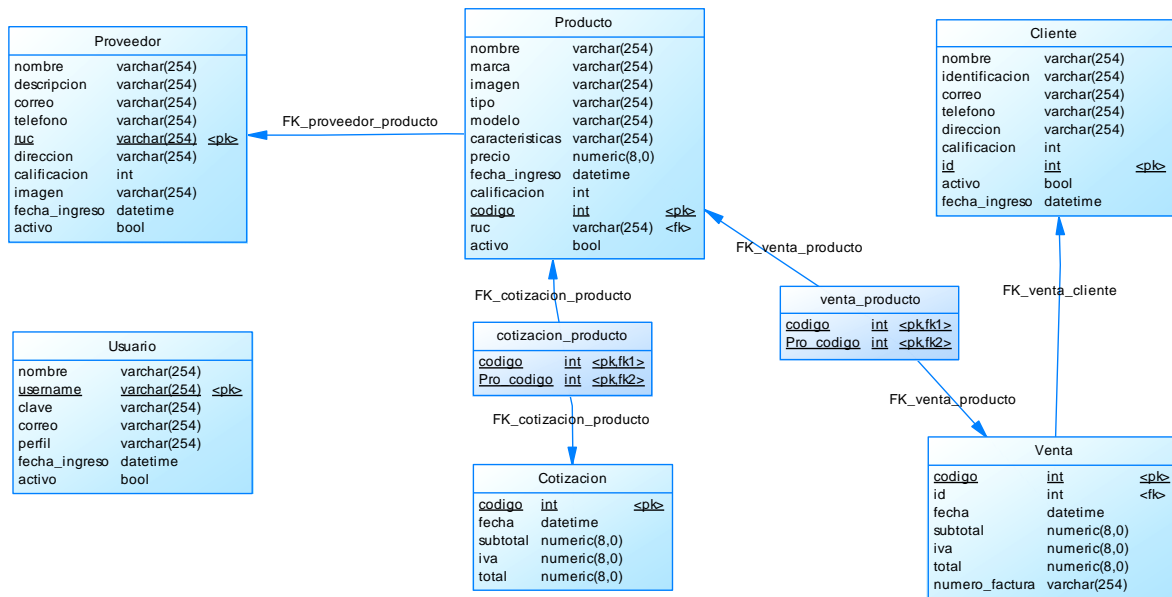


Figura IV – 24: Modelo Físico

Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.3.5. Diccionario datos

Entidades	Atributos	Tipo de Dato	Descripción
<b>Usuario</b>	username	varchar(50)	<b>Clave</b> única de registro de usuarios
	nombre	varchar(254)	Nombre del Usuario
	clave	varchar(254)	Contraseña para ingreso al sistema
	correo	varchar(254)	Correo electrónico del usuario
	activo	tinyint(4)	Estado del Usuario
	fecha_ingreso	datetime	Fecha en la que el Usuario fue creado
	perfil	varchar(50)	Nombre del perfil del usuario
<b>Producto</b>	codigo	int	<b>Clave</b> única de registro de productos
	nombre	varchar(254)	Nombre del Producto
	marca	varchar(254)	Marca del Producto

	imagen	varchar(254)	Imagen del Producto
	tipo	varchar(254)	Tipo de Producto
	modelo	varchar(254)	Modelo del Producto
	caracteristicas	varchar(254)	Características que tiene el Producto
	precio	numeric(8,0)	Precio del Producto
	fecha_ingreso	datetime	Fecha de Creación del Producto
	calificacion	varchar(254)	La calificación del Producto
	activo	tinyint(1)	Estado del Producto
	proveedor_ruc	varchar(13)	Clave foránea de la tabla Proveedor
<b>Cliente</b>	id	int	<b>Clave</b> única de registro de clientes
	nombre	varchar(254)	Nombre del Cliente
	identificacion	varchar(254)	Cedula o pasaporte del Cliente
	correo	varchar(254)	Correo electrónico del Cliente
	telefono	varchar(254)	Número de teléfono del Cliente
	direccion	varchar(254)	Domicilio del Cliente
	calificacion	int	La calificación del Cliente
	activo	tinyint(1)	Estado del Cliente
	fecha_ingreso	datetime	Fecha de creación del Cliente
<b>Proveedor</b>	ruc	varchar(13)	<b>Clave</b> única de registro de proveedores
	nombre	varchar(254)	Nombre del Proveedor
	descripcion	varchar(254)	Descripción del Proveedor
	correo	varchar(254)	Correo electrónico del Proveedor
	telefono	varchar(254)	Número de teléfono del Proveedor
	direccion	varchar(254)	Dirección del Proveedor
	calificacion	int	Calificación del Proveedor
	imagen	varchar(254)	Imagen del Proveedor
	fecha_ingreso	datetime	Fecha de creación del Proveedor
	activo	tinyint(1)	Estado del Proveedor
<b>Venta</b>	codigo	int(11)	<b>Clave</b> única de registro de facturas
	cliente_id	int(11)	Clave foránea
	fecha	datetime	Fecha que se realiza la factura
	subtotal	numeric(8,0)	Suma de todos los productos adquiridos en la factura

	iva	numeric(8,0)	Porcentaje de IVA
	total	numeric(8,0)	Suma de subtotal más IVA
<b>Cotización</b>	codigo	int	<b>Clave</b> única de registro de Cotizaciones
	fecha	datetime	Fecha que se realiza la Cotización
	subtotal	numeric(8,0)	Suma de todos los productos seleccionados en la Cotización
	iva	numeric(8,0)	Porcentaje de IVA
	total	numeric(8,0)	Suma de subtotal más IVA
<b>cotizacion_producto</b>	codigo	int(11)	<b>Clave</b> primaria. Clave foránea de la tabla cotización.
	Pro_codigo	int(11)	<b>Clave</b> primaria Clave foránea de la tabla producto.
<b>venta_producto</b>	codigo	int(11)	<b>Clave</b> primaria. Clave foránea de la tabla venta.
	Pro_codigo	int(11)	<b>Clave</b> primaria Clave foránea de la tabla producto.

**Tabla IV – 11:** Diccionario de Datos

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.3.6. Estándares y estrategias

La codificación del sitio debe estandarizarse tomando en cuenta las reglas definidas en la Tabla IV - 7, y deberán ser cumplidos al pie de la letra pues al haber dos responsables del código fuente, se desea evitar conflictos de estándares de codificación.

Tipo	Regla	Excepción / Observación
<b>Nombres de variables</b>	La declaración de variables, de ser necesarias, se definirá de manera clara, seguida por un comentario que describa el por qué de su declaración.	
	Los nombres deben estar en minúsculas, excepto la primera letra de cada palabra.	Únicamente las variables temporales pueden llamarse usando caracteres especiales, por

			ejemplo: \$aa o \$a1
<b>Nombres de las funciones</b>	El nombre de las funciones o métodos empezarán con minúsculas y se intercalará con mayúsculas en el inicio de la siguiente palabra.	Por	ejemplo buscarClienteLubrimotor();
<b>Nombres de archivos</b>	Todo en minúscula excepto la primera letra de cada palabra		
<b>Comentarios</b>	El código creado por los desarrolladores deberá redactarse de manera ordenada, entendible y con los comentarios necesarios para su correcto entendimiento.	Los comentarios serán puestos bajo tags HTML (<!-- .... -->) o PHP (/* ..... */).	
<b>Constantes y variables globales</b>	Todos los caracteres deben estar en mayúsculas y las palabras separadas por "_".	Se deben evitar constantes numéricas sin mucho significado.	
			Se debe evitar el uso de variables globales ya que pueden ser modificadas erróneamente causando problemas difíciles de detectar.
<b>Corchetes e indentación</b>	La indentación es algo que ayuda a darle claridad a un programa y es indispensable que se realice usando "tabs" y no con espacios en blanco.		
			Los corchetes de un bloque if, o switch, o for, deben ir en la misma línea de la cláusula.
<b>Claridad en la programación</b>	Es importante que los programas y rutinas que se escriban sean claros y fáciles de entender. Por eso, además de dar la explicación de que hace cada programa o función al principio, hay que tratar que las funciones		

	quepan en una sola página y que antes de cada sección se explique qué es lo que se está realizando
<b>HTML</b>	Los tags tienen que estar escritos en minúsculas. Por ejemplo <td> en vez de <TD>
	Los objetos html deben tener id y name, y ambos deben ser iguales. Cuando solo se usa el name no se visualiza correctamente en todos los Browsers.
	Cuando hay que escribir variables de PHP dentro de un código HTML se debe tener el HTML dentro de PHP y no al revés.
	El código HTML generado por un programa también debe ser legible, bien estructurado e indentado. Por lo que se recomienda el uso de "<n" y "<t".
<b>CSS</b>	No usar una línea para cada atributo, ya que hace los archivos muy grandes.
<b>Generales</b>	Es necesario acoplarse al modelo MVC y al trabajo con componentes, plugins, módulos y extensiones de los mismos.
<b>Framework</b>	Se respetara los estándares que propone el Codeigniter. Esto es: nombre de archivos, código comentado, uso de variables e idioma de los mismos

**Tabla IV – 12:** Estándares y estrategias de Programación  
**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

4.1.3.7. Catálogo de piezas de software construidos

Nombre	Tipo	Descripción
<b>Application</b>	directorio	Contiene la aplicación web
<b>config</b>	directorio	Contiene todos los archivos de configuración de la aplicación
autoload.php	archivo PHP	Carga automáticamente los recursos necesarios.
config.php	archivo PHP	Contiene las configuraciones de la aplicación como zona horaria y url base.
database.php	archivo PHP	Archivo de configuración de la base de datos.
<b>controllers</b>	directorio	Capa de controladores de la aplicación.
home.php	archivo PHP	Controlador principal, que maneja la pantalla de inicio y de ingreso.
usuario.php	archivo PHP	Controlador que maneja la edición e ingreso de usuarios.
usuarioList.php	archivo PHP	Controlador que maneja la lista de usuarios.
proveedor.php	archivo PHP	Controlador que maneja la edición e ingreso de proveedores.
proveedorList.php	archivo PHP	Controlador que maneja la lista de proveedores.
producto.php	archivo PHP	Controlador que maneja la edición e ingreso de productos.
productoList.php	archivo PHP	Controlador que maneja la lista de productos.
cliente.php	archivo PHP	Controlador que maneja la edición e ingreso de clientes.
clienteList.php	archivo PHP	Controlador que maneja la lista de clientes.
<b>models</b>	directorio	Capa de modelos de la aplicación.
usuario_model.php	archivo PHP	Modelo que maneja los datos de los usuarios.
proveedor_model.php	archivo PHP	Modelo que maneja los datos de los proveedores.

producto_model.php	archivo PHP	Modelo que maneja los datos de los productos.
cliente_model.php	archivo PHP	Modelo que maneja los datos de los clientes.
<b>views</b>	directorio	Capa de vistas de la aplicación.
index_view.php	archivo PHP	Plantilla del sistema web
loginView.php	archivo PHP	Pantalla de ingreso al sistema
principal_view.php	archivo PHP	Pantalla principal
usuarioList_view.php	archivo PHP	Archivo que muestra la lista de usuarios
usuarioForm_view.php	archivo PHP	Archivo que muestra el formulario de ingreso o edición de un usuario
proveedorList_view.php	archivo PHP	Archivo que muestra la lista de proveedores
proveedorForm_view.php	archivo PHP	Archivo que muestra el formulario de ingreso o edición de un proveedor
productoList_view.php	archivo PHP	Archivo que muestra la lista de productos
productoForm_view.php	archivo PHP	Archivo que muestra el formulario de ingreso o edición de un producto
clienteList_view.php	archivo PHP	Archivo que muestra la lista de clientes
clienteForm_view.php	archivo PHP	Archivo que muestra el formulario de ingreso o edición de un cliente

**Tabla IV – 13:** Catalogo de Piezas de Software

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.3.8. Aplicación Pruebas

##### 4.1.3.8.1. Pruebas en Ambiente de Desarrollo

Las pruebas funcionales en ambiente de desarrollo fueron ejecutadas dos por cada caso de uso, sin embargo a continuación se muestran una por cada caso de uso y las demás se encontraran en el Anexo IV.

INFORMACIÓN GLOBAL DE LA PRUEBA			
<b>CASO DE USO:</b>	F1	<b>VERSIÓN DE EJECUCIÓN</b>	1.0

	Administración de Usuarios	<b>FECHA EJECUCIÓN</b>		15/04/2013
		<b>MODULO DEL SISTEMA</b>		Administración
<b>Descripción del caso de prueba:</b>	Consulta, ingreso y edición de usuarios.			
<b>1. CASO DE USO</b>				
<b>a. Precondiciones</b>				
Ninguna.				
<b>b. Pasos de la prueba</b>				
PASOS	RESPUESTA ESPERADA DE LA APLICACIÓN	COINCIDE		RESPUESTA DEL SISTEMA
		SI	NO	
1. Seleccionar opción de administración de Usuarios	Desplegar plantilla de usuarios	X		
2. Consulta de datos de Usuarios.				
Datos Correctos	Se presenta una tabla con los datos de los usuarios existentes.		X	No presenta ningún dato y hay un error de programación no estandarizado
Datos Incorrectos	Se presenta un mensaje de error.	X		
3. Edición de datos de Usuario.				
Datos Correctos	Se actualizan los datos del Usuario.		X	No edita al usuario, sino que lo crea como uno nuevo
Datos Incorrectos	Se presentan mensajes de error.		X	No presenta ningún error pues no hay validaciones.
4. Registro de datos de Usuario.				
Datos Correctos	Se crea el registro del Usuario.	X		
Datos Incorrectos	Se presentan mensajes de		X	No presenta ningún error pues no

	error.		hay validaciones.
<b>c. Post condiciones</b>			
La base de datos debe quedar actualizada. El usuario ingresado puede acceder al sistema.			
<b>2. RESULTADOS DE LA PRUEBA</b>			
<b>Defectos y desviaciones</b>			<b>Veredicto</b>
En la consulta de Usuarios se presentó un error no estandarizado y no entendible.			<b>Falló</b>
<b>Observaciones / Modificaciones Sugeridas</b>			<b>Probador</b>
Revisar el código de edición de usuario pues no está realizando su función y está creando un nuevo Usuario.  Verificar las validaciones necesarias para el formulario de creación y edición de Usuario.			<b>Firma:</b> <b>Nombre: Julio Alberto Lárraga</b> <b>Fecha: 2013/04/15</b>

**Tabla IV – 14:** Prueba de Caso de Uso (Administración de Usuarios)

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

<b>INFORMACIÓN GLOBAL DE LA PRUEBA</b>			
<b>CASO DE USO:</b>	F2  Administración de Proveedores	<b>VERSIÓN DE EJECUCIÓN</b>	1.0
		<b>FECHA EJECUCIÓN</b>	15/04/2013
		<b>MODULO DEL SISTEMA</b>	Administración
<b>Descripción del caso de prueba:</b>	Consulta, ingreso y edición de proveedores		
<b>1. CASO DE USO</b>			
<b>a. Precondiciones</b>			
Información de proveedores.			

<b>b. Pasos de la prueba</b>				
<b>PASOS</b>	<b>RESPUESTA ESPERADA DE LA APLICACIÓN</b>	<b>COINCIDE</b>		<b>RESPUESTA DEL SISTEMA</b>
		<b>SI</b>	<b>NO</b>	
1. Seleccionar opción de administración de Proveedores	Desplegar plantilla de usuarios	X		
2. Consulta de datos de Proveedores.				
Datos Correctos	Se presenta una tabla con los datos de los proveedores existentes.	X		
Datos Incorrectos	Se presenta un mensaje de error.	X		
3. Edición de datos de Proveedor.				
Datos Correctos	Se actualizan los datos del Proveedor.		X	No se actualizan algunos de los datos del proveedor.
Datos Incorrectos	Se presentan mensajes de error.	X		
4. Registro de datos de Proveedor.				
Datos Correctos	Se crea el registro del Proveedor.		X	Se crea el registro pero no con todos los datos ingresados.
Datos Incorrectos	Se presentan mensajes de error.	X		
<b>c. Post condiciones</b>				
La base de datos debe quedar actualizada.				
<b>2. RESULTADOS DE LA PRUEBA</b>				
<b>Defectos y desviaciones</b>				<b>Veredicto</b>
NA				<b>Falló</b>
<b>Observaciones / Modificaciones Sugeridas</b>				<b>Probador</b>

Revisar las funciones de persistencia a la Base de Datos pues hay datos que no se están guardando. También revisar los "id" o "name" en las etiquetas "<input>" del HTML.	<b>Firma:</b>
	<b>Nombre: Julio Alberto Lárraga</b>
	<b>Fecha: 2013/04/15</b>

**Tabla IV – 15:** Prueba de Caso de Uso (Administración de Proveedores)

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

INFORMACIÓN GLOBAL DE LA PRUEBA				
<b>CASO DE USO:</b>	F3  Administración de Productos	<b>VERSIÓN DE EJECUCIÓN</b>	1.0	
		<b>FECHA EJECUCIÓN</b>	15/04/2013	
		<b>MODULO DEL SISTEMA</b>	Administración	
<b>Descripción del caso de prueba:</b>	Consulta, ingreso y edición de productos.			
<b>1. CASO DE USO</b>				
<b>a. Precondiciones</b>				
Información de productos.				
<b>b. Pasos de la prueba</b>				
PASOS	RESPUESTA ESPERADA DE LA APLICACIÓN	COINCIDE		RESPUESTA DEL SISTEMA
		SI	NO	
1. Seleccionar opción de administración de Productos	Desplegar plantilla de productos	X		
2. Consulta de datos de Productos.				
Datos Correctos	Se presenta una tabla con los datos de los productos		X	No se realiza la búsqueda de acuerdo a los datos de búsqueda

	existentes.			ingresados.
Datos Incorrectos	Se presenta un mensaje de error.		X	No se realiza la búsqueda de acuerdo a los datos de búsqueda ingresados.
3. Edición de datos de Producto.				
Datos Correctos	Se actualizan los datos del Producto.		X	No se registra la imagen ingresada.
Datos Incorrectos	Se presentan mensajes de error.	X		
4. Registro de datos de Producto.				
Datos Correctos	Se crea el registro del Producto.		X	No se registra la imagen ingresada.
Datos Incorrectos	Se presentan mensajes de error.	X		
<b>c. Post condiciones</b>				
La base de datos debe quedar actualizada.				
<b>2. RESULTADOS DE LA PRUEBA</b>				
<b>Defectos y desviaciones</b>				<b>Veredicto</b>
NA				<b>Falló</b>
<b>Observaciones / Modificaciones Sugeridas</b>				<b>Probador</b>
Revisar las funciones de búsqueda en el controlador de lista de Productos al igual que el HTML de listado de Productos.				
Modificar la función de subida de imágenes, pues no está realizando su función correctamente.				<b>Firma:</b> <b>Nombre: Julio Alberto Lárraga</b> <b>Fecha: 2013/04/15</b>

**Tabla IV – 16:** Prueba de Caso de Uso (Administración de Productos)

Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

INFORMACIÓN GLOBAL DE LA PRUEBA				
<b>CASO DE USO:</b>	F4	Administración de Clientes	<b>VERSIÓN DE EJECUCIÓN</b>	1.0
			<b>FECHA EJECUCIÓN</b>	15/04/2013
			<b>MODULO DEL SISTEMA</b>	Administración
<b>Descripción del caso de prueba:</b>	Consulta, ingreso y edición de clientes.			
<b>1. CASO DE USO</b>				
<b>a. Precondiciones</b>				
Información de clientes.				
<b>b. Pasos de la prueba</b>				
PASOS	RESPUESTA ESPERADA DE LA APLICACIÓN	COINCIDE		RESPUESTA DEL SISTEMA
		SI	NO	
1. Seleccionar opción administración de Clientes	Desplegar plantilla de clientes	X		
2. Consulta de datos de Clientes.				
Datos Correctos	Se presenta una tabla con los datos de los clientes existentes.	X		
Datos Incorrectos	Se presenta un mensaje de error.		X	No presenta ningún mensaje de error
3. Edición de datos de Cliente.				
Datos Correctos	Se actualizan los datos del Cliente.	X		
Datos Incorrectos	Se presentan mensajes de error.		X	No presenta ningún mensaje de error

4. Registro de datos de Cliente.				
Datos Correctos	Se crea el registro del Cliente.	X		
Datos Incorrectos	Se presentan mensajes de error.		X	No presenta ningún mensaje de error
<b>c. Post condiciones</b>				
La base de datos debe quedar actualizada.				
<b>2. RESULTADOS DE LA PRUEBA</b>				
<b>Defectos y desviaciones</b>				<b>Veredicto</b>
NA				<b>Falló</b>
<b>Observaciones / Modificaciones Sugeridas</b>			<b>Probador</b>	
Programar funciones de validación para todas las interfaces de Cliente, ya que me permite ingresar cualquier dato sin validarlo.			<b>Firma:</b> <b>Nombre: Julio Alberto Lárraga</b> <b>Fecha: 2013/04/15</b>	

**Tabla IV – 17:** Prueba de Caso de Uso (Administración de Clientes)

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.4. IMPLANTACIÓN

La presente fase describe el conjunto de tareas necesarias para instalar y probar el sistema de software desarrollado, para que así la transición a los usuarios se realice efectiva y eficientemente. Se debe tener en mente que este aplicativo es web, por lo tanto cuando se hable de instalación se referirá a las actividades de subir al servidor web todos los componentes necesarios para la correcta corrida del sistema.

Esta fase a través de un plan de implantación, busca definir la lista de tareas o agenda de implantación que incluye la preparación del ambiente para la transición y ejecución del sistema, el

entrenamiento de los usuarios y detalles de los futuros procedimientos de soporte, todo esto a través de un cronograma de actividades y responsables necesarios para la oportuna implantación.

También se van a definir las dependencias y los recursos necesarios tanto de software como de hardware para poder instalar el sistema, es decir, los condicionantes de la implantación.

#### 4.1.4.1. Plan de Implantación

##### 4.1.4.1.1. Tipo de Implantación

La Tabla IV – 9 muestra el servidor web y el dominio en el cual estará funcionando la aplicación web de este proyecto.

<b>Tipo del Sistema</b>	<b>Aplicación Web</b>
<b>Servidor Web</b>	ixwebhosting
<b>Dominio</b>	lubrimotorautomotriz.com

**Tabla IV – 18:** Catalogo de Piezas de Software

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

##### 4.1.4.1.2. Responsabilidades

Para poder llevar a cabo la implantación del sistema desarrollado, es necesario establecer un compromiso por parte de los clientes y el grupo de desarrolladores.

Las tareas o responsabilidades que deberá llevar a cabo el equipo, se enfocan en la realización de pruebas del sistema, para asegurarnos de que el sistema puede ser llevado a otro ambiente de trabajo diferente a las computadoras personales del grupo de desarrolladores.

Una vez que se haya subido el servidor web la versión final del sistema, se pedirá al administrador y a los usuarios del sistema la realización de las pruebas de usabilidad, para posteriormente realizar los ajustes pertinentes en el sistema.



##### 4.1.4.1.3. Cronograma

La implantación del sistema en Lubrimotor Automotriz es responsabilidad de los desarrolladores. A continuación se va a proponer un cronograma de las actividades, y de sus entregables.






















Se debe tener en consideración que el Informe de Recursos y el Acta de Entrega Recepción del Sistema que se citará continuación estarán en el Anexo IV (Informe de Recursos de Hardware y de Software y Acta de Entrega y Recepción).

**Cronograma de Implantación**

Lubrimotor Automotriz (LA)

Terminado   
 Pendiente 

➤ Entregables

Actividad	Responsable	Enero				Marzo				Abril				Mayo				Junio			
		S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4
Análisis de Recursos	Desarrolladores																				
➤ Informe de Recursos (Anexos)	Desarrolladores																				
Preparar los equipos / Internet	IT - LA																				
Adquisición de Dominio y Servidor Web	Desarrolladores																				
Subir Versión Final al Servidor Web	Desarrolladores																				
Ejecutar Pruebas de Usabilidad	Usuarios - Administrador																				
Realizar Ajustes	Desarrolladores																				
Subir versión final a Servidor Web	Desarrolladores																				
Entrenamiento a Administrador del Sistema	Desarrolladores - Administrador																				
Entrenamiento a Usuarios	Desarrolladores - Usuarios																				
➤ Manual Técnico	Desarrolladores																				
➤ Manual de Usuarios	Desarrolladores																				
➤ Acta Entrega Recepción de Sistema	Desarrolladores - LA																				

**Tabla IV – 19:** Cronograma de Implementación

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.4.1.3. Documentación de Soporte

##### 4.1.4.1.3.1. Creación del Manual Técnico

El manual técnico está compuesto por todas las especificaciones descritas en el análisis y el diseño del sistema. Especialmente, las especificaciones de las clases con todos sus atributos, así como la relación existente entre ellas y el proceso de implantación para futuros cambios estarán disponibles en el CD.

El técnico que desee conocer la estructura interna del software podrá hacerlo fácilmente explorando el archivo ManualTecnico.docx.

##### 4.1.4.1.3.2. Creación del Manual de Usuario

El manual del usuario tiene todas las instrucciones para la utilización del software que se ha desarrollado. Debido a que este muestra todas las interfaces gráficas con el usuario, se decidió ubicarlo en el Anexo IV. De esta manera se permitió que el usuario pueda tener acceso a la documentación del manual en cualquier momento desde el archivo ManualUsuario.docx.

##### 4.1.4.1.3.3. Personal de Soporte

El personal de soporte en primera línea será el Administrador del Sistema asignado por Lubrimotor Automotriz, esta persona será la que brinde el soporte a todos los usuarios en el sistema, por otro lado el equipo conformado por los desarrolladores de éste proyecto, María Gabriela Bedón Melo y Julio Alberto Lárraga Mena serán el soporte auxiliar durante los seis primeros meses, después de que las partes firmen el Acta de Entrega y Recepción del Sistema. En casos de que existan fallas en la aplicación deberá el Administrador asignado, documentarla con evidencias, el formato para reportar estas fallas será la que se muestra en la Tabla IV – 20 y el medio de contacto será vía email o vía telefónica.

#### **Reporte de Fallas**

SISTEMA:	MODULO DEL SISTEMA:	N° Fallo	PAG.
Fecha de Detección:			
Responsable:			

Proceso de Detección:
Detalles Adicionales: Adjuntar pantallas
Prioridad en el Negocio:

**Tabla IV - 20:** Formato de Reporte de Fallos

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.4.1.4. Entrenamiento

Se tiene pensado que el entrenamiento total del administrador y de los usuarios se haga durante cuatro días, y que posteriormente ellos por su parte sigan practicando y puedan así conocer el sistema. El primer entrenamiento lo recibirá el Administrador del Sistema designado previamente por la empresa y posteriormente los usuarios complementarios que harán uso del sistema, para explicarlo la Tabla IV - 46.

Es importante que durante el entrenamiento y soporte de los usuarios del sistema, el dueño y el administrador del sistema estén presentes. En estos cuatro días se realizará acompañamiento a los usuarios en el manejo del aplicativo en las actividades diarias.

Rol	Descripción	Tiempo
<b>Administrador</b>	Este actor se encargara del soporte posterior por lo cual el entrenamiento será enfocado a superuser del sistema	16 horas
<b>Usuarios Finales</b>	El entrenamiento tendrá como foco el de usuarios finales, que usan el sistema en el día a día.	16 horas

**Tabla IV - 21:** Formato de Reporte de Fallos

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.4.2. Pruebas con Usuarios

##### 4.1.4.2.1. Resultados de las Pruebas

INFORMACIÓN GLOBAL DE LA PRUEBA		
<b>VERSIÓN DE EJECUCIÓN</b>	1.0	
<b>FECHA EJECUCIÓN</b>	2013-04-15	
<b>MODULO DEL SISTEMA</b>	Administración	
1. PRUEBA DE USABILIDAD		
PREGUNTA	CRITERIOS DE EVALUACIÓN	RESPUESTA
1. ¿Es simple el vocabulario utilizado?	1 = El vocabulario es demasiado técnico. 2 = El vocabulario presenta algunas dificultades de comprensión. 3 = El vocabulario es completamente comprensible.	3
2. ¿Hay algún tipo de asistencia para los usuarios que hacen uso del sistema por primera vez?	1 = No existe ninguna ayuda. 2 = Se encuentra ayuda en algunas partes. 3 = Existen ayudas en todo el sistema.	1
3. ¿El sistema es fácil de operar para alguien que no recibió capacitación en su operación?	1 = El sistema es de difícil comprensión. 2 = El sistema es fácil de operar en algunas de sus funcionalidades. 3 = El sistema es completamente fácil de operar.	2
4. ¿Se entienden la interfaz y su contenido?	1 = No se entiende su interfaz. 2 = La interfaz se entiende en algunas partes.	2

	3 = La interfaz es completamente entendible.	
5. ¿Resulta fácil identificar un objeto o una acción?	1 = Es difícil identificar los objetos o acciones. 2 = Se pueden identificar los objetos y acciones en algunas partes del sistema. 3 = Todos los objetos y acciones son fácilmente identificables.	1
6. ¿Resulta fácil entender el resultado de una acción?	1 = Los resultados de las acciones no son entendibles. 2 = Los resultados de las acciones son entendibles en algunas partes o la mayor parte del sistema. 3 = Todos los resultados de las acciones son entendibles.	2
7. ¿Está diseñada la interfaz para facilitar la realización eficiente de las tareas de la mejor forma posible?	1 = La interfaz es difícil de usar. 2 = La interfaz es difícil de usar en algunas partes del sistema. 3 = La interfaz es completamente sencilla de usar.	1
8. ¿Son apropiados los mensajes presentado por el sistema?	1 = Los mensajes no son apropiados. 2 = Los mensajes son apropiados en algunas partes del sistema. 3 = Todos los mensajes son apropiados y fáciles de comprender.	1
9. ¿Actúa el sistema en la prevención de errores?	1 = El sistema no previene errores del usuario. 2 = El sistema previene algunos o la mayoría de los errores del usuario. 3 = El sistema previene cualquier error que pueda cometer el usuario.	2

<p>10. ¿El sistema informa claramente sobre los errores presentados?</p>	<p>1 = El sistema no informa de manera adecuada sobre los errores cometidos.                  2 = El sistema informa de manera adecuada algunos o la mayoría de los errores cometidos por el usuario.                  3 = El sistema informa de forma adecuada todos los errores cometidos por el usuario.</p>	<p><b>2</b></p>
<p>11. ¿Se utiliza mensajes y textos descriptivos?</p>	<p>1 = Los mensajes de texto no son descriptivos.                  2 = La mayoría de los textos son descriptivos o fáciles de interpretar                  3 = Todos los textos son descriptivos o fáciles de interpretar.</p>	<p><b>1</b></p>
<p>12. ¿Se proporciona información visual de dónde está el usuario, qué está haciendo y qué puede hacer a continuación?</p>	<p>1 = No se presenta ninguna información visual ni otro tipo de ayuda.                  2 = Presenta ayudas en algunas partes del sistema.                  3 = Las ayudas son apropiadas y están distribuidas a los largo del sistema.</p>	<p><b>1</b></p>
<p><b>Observaciones / Modificaciones Sugeridas</b></p>		<p><b>Probador</b></p>
<p>Mejorar la interfaz de usuario en cuanto a color e imágenes.</p> <p>Estudiar la navegabilidad del sitio pues es un tanto confusa la interacción para moverse entre opciones del sistema.</p> <p>Utilizar expresiones más claras en los mensajes error, actualmente son mensajes muy técnicos.</p> <p>Si es posible agregar una ayuda del sistema.</p>		

	<b>Firma:</b>
	<b>Nombre:</b>
	<b>Fecha:</b> 2013-04-15

**Tabla IV - 22:** Resultado de la Prueba

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

<b>INFORMACIÓN GLOBAL DE LA PRUEBA</b>		
<b>VERSIÓN DE EJECUCIÓN</b>	1.0	
<b>FECHA EJECUCIÓN</b>	2013-04-15	
<b>MODULO DEL SISTEMA</b>	Administración	
<b>1. PRUEBA DE USABILIDAD</b>		
<b>PREGUNTA</b>	<b>CRITERIOS DE EVALUACIÓN</b>	<b>RESPUESTA</b>
1. ¿Es simple el vocabulario utilizado?	1 = El vocabulario es demasiado técnico. 2 = El vocabulario presenta algunas dificultades de comprensión. 3 = El vocabulario es completamente comprensible.	<b>3</b>
2. ¿Hay algún tipo de asistencia para los usuarios que hacen uso del sistema por primera vez?	1 = No existe ninguna ayuda. 2 = Se encuentra ayuda en algunas partes. 3 = Existen ayudas en todo el sistema.	<b>1</b>
3. ¿El sistema es fácil de operar para alguien que no recibió	1 = El sistema es de difícil comprensión.	<b>2</b>

capacitación en su operación?	<p>2 = El sistema es fácil de operar en algunas de sus funcionalidades.</p> <p>3 = El sistema es completamente fácil de operar.</p>	
4. ¿Se entienden la interfaz y su contenido?	<p>1 = No se entiende su interfaz.</p> <p>2 = La interfaz se entiende en algunas partes.</p> <p>3 = La interfaz es completamente entendible.</p>	<b>2</b>
5. ¿Resulta fácil identificar un objeto o una acción?	<p>1 = Es difícil identificar los objetos o acciones.</p> <p>2 = Se pueden identificar los objetos y acciones en algunas partes del sistema.</p> <p>3 = Todos los objetos y acciones son fácilmente identificables.</p>	<b>3</b>
6. ¿Resulta fácil entender el resultado de una acción?	<p>1 = Los resultados de las acciones no son entendibles.</p> <p>2 = Los resultados de las acciones son entendibles en algunas partes o la mayor parte del sistema.</p> <p>3 = Todos los resultados de las acciones son entendibles.</p>	<b>3</b>
7. ¿Está diseñada la interfaz para facilitar la realización eficiente de las tareas de la mejor forma posible?	<p>1 = La interfaz es difícil de usar.</p> <p>2 = La interfaz es difícil de usar en algunas partes del sistema.</p> <p>3 = La interfaz es completamente sencilla de usar.</p>	<b>1</b>
8. ¿Son apropiados los mensajes presentado por el sistema?	<p>1 = Los mensajes no son apropiados.</p> <p>2 = Los mensajes son apropiados en algunas partes del sistema.</p> <p>3 = Todos los mensajes son apropiados y fáciles de comprender.</p>	<b>1</b>

<p>9. ¿Actúa el sistema en la prevención de errores?</p>	<p>1 = El sistema no previene errores del usuario.                  2 = El sistema previene algunos o la mayoría de los errores del usuario.                  3 = El sistema previene cualquier error que pueda cometer el usuario.</p>	<p>2</p>
<p>10. ¿El sistema informa claramente sobre los errores presentados?</p>	<p>1 = El sistema no informa de manera adecuada sobre los errores cometidos.                  2 = El sistema informa de manera adecuada algunos o la mayoría de los errores cometidos por el usuario.                  3 = El sistema informa de forma adecuada todos los errores cometidos por el usuario.</p>	<p>1</p>
<p>11. ¿Se utiliza mensajes y textos descriptivos?</p>	<p>1 = Los mensajes de texto no son descriptivos.                  2 = La mayoría de los textos son descriptivos o fáciles de interpretar                  3 = Todos los textos son descriptivos o fáciles de interpretar.</p>	<p>1</p>
<p>12. ¿Se proporciona información visual de dónde está el usuario, qué está haciendo y qué puede hacer a continuación?</p>	<p>1 = No se presenta ninguna información visual ni otro tipo de ayuda.                  2 = Presenta ayudas en algunas partes del sistema.                  3 = Las ayudas son apropiadas y están distribuidas a lo largo del sistema.</p>	<p>1</p>
<p><b>Observaciones / Modificaciones Sugeridas</b></p>		<p><b>Probador</b></p>
<p>Mejorar mensajes del sistema.                   Que la interfaz de usuario sea más amigable.</p>		

Mejorar navegabilidad del sistema.  Colocar componentes claros e intuitivos para la interacción del usuario.	
	<b>Firma:</b>
	<b>Nombre:</b>
	<b>Fecha:</b>

**Tabla IV - 23:** Resultado de la Prueba

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

4.1.4.2.2. Versión Final de Pruebas

INFORMACIÓN GLOBAL DE LA PRUEBA				
<b>CASO DE USO:</b>	F1  Administración de Usuarios	<b>VERSIÓN DE EJECUCIÓN</b>	1.0	
		<b>FECHA EJECUCIÓN</b>	18/04/2013	
		<b>MODULO DEL SISTEMA</b>	Administración	
<b>Descripción del caso de prueba:</b>	Consulta, ingreso y edición de usuarios.			
<b>1. CASO DE USO</b>				
<b>a. Precondiciones</b>				
Ninguna.				
<b>b. Pasos de la prueba</b>				
PASOS	RESPUESTA ESPERADA DE LA APLICACIÓN	COINCIDE		RESPUESTA DEL SISTEMA
		SI	NO	
1. Seleccionar opción de administración de Usuarios	Desplegar plantilla de usuarios	X		

2. Consulta de datos de Usuarios.				
Datos Correctos	Se presenta una tabla con los datos de los usuarios existentes.	X		
Datos Incorrectos	Se presenta un mensaje de error.	X		
3. Edición de datos de Usuario.				
Datos Correctos	Se actualizan los datos del Usuario.	X		
Datos Incorrectos	Se presentan mensajes de error.	X		
4. Registro de datos de Usuario.				
Datos Correctos	Se crea el registro del Usuario.	X		
Datos Incorrectos	Se presentan mensajes de error.	X		
<b>c. Post condiciones</b>				
La base de datos debe quedar actualizada. El usuario ingresado puede acceder al sistema.				
<b>2. RESULTADOS DE LA PRUEBA</b>				
<b>Defectos y desviaciones</b>				<b>Veredicto</b>
				<b>Pasó</b>
<b>Observaciones / Modificaciones Sugeridas</b>			<b>Probador</b>	
Se han corregido todos los errores de funcionamiento, sin embargo no se han cambiado los mensajes del sistema ni las validaciones del formulario.				
			<b>Firma:</b> <b>Nombre: Julio Alberto Lárraga</b> <b>Fecha: 2013/04/18</b>	

**Tabla IV – 24:** Prueba de Caso de Uso (Administración de Usuarios)

Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

INFORMACIÓN GLOBAL DE LA PRUEBA					
<b>CASO DE USO:</b>	F2	Administración de Proveedores	<b>VERSIÓN DE EJECUCIÓN</b>		1.0
			<b>FECHA EJECUCIÓN</b>		18/04/2013
			<b>MODULO DEL SISTEMA</b>		Administración
<b>Descripción del caso de prueba:</b>	Consulta, ingreso y edición de proveedores				
<b>1. CASO DE USO</b>					
<b>a. Precondiciones</b>					
Información de proveedores.					
<b>b. Pasos de la prueba</b>					
PASOS	RESPUESTA ESPERADA DE LA APLICACIÓN	COINCIDE		RESPUESTA DEL SISTEMA	
		SI	NO		
1. Seleccionar opción de administración de Proveedores	Desplegar plantilla de usuarios	X			
2. Consulta de datos de Proveedores.					
Datos Correctos	Se presenta una tabla con los datos de los proveedores existentes.	X			
Datos Incorrectos	Se presenta un mensaje de error.	X			
3. Edición de datos de Proveedor.					
Datos Correctos	Se actualizan los datos del Proveedor.	X			

Datos Incorrectos	Se presentan mensajes de error.	X		
4. Registro de datos de Proveedor.				
Datos Correctos	Se crea el registro del Proveedor.	X		
Datos Incorrectos	Se presentan mensajes de error.	X		
<b>c. Post condiciones</b>				
La base de datos debe quedar actualizada.				
<b>2. RESULTADOS DE LA PRUEBA</b>				
<b>Defectos y desviaciones</b>				<b>Veredicto</b>
NA				Pasó
<b>Observaciones / Modificaciones Sugeridas</b>			<b>Probador</b>	
NA			<b>Firma:</b> <b>Nombre: Julio Alberto Lárraga</b> <b>Fecha: 2013/04/18</b>	

**Tabla IV – 25:** Prueba de Caso de Uso (Administración de Proveedores)

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

<b>INFORMACIÓN GLOBAL DE LA PRUEBA</b>			
<b>CASO DE USO:</b>	F3	<b>VERSIÓN DE EJECUCIÓN</b>	1.0
		<b>FECHA EJECUCIÓN</b>	18/04/2013
	Administración de Productos	<b>MODULO DEL</b>	Administración

		SISTEMA		
<b>Descripción del caso de prueba:</b>	Consulta, ingreso y edición de productos.			
<b>1. CASO DE USO</b>				
<b>a. Precondiciones</b>				
Información de productos.				
<b>b. Pasos de la prueba</b>				
PASOS	RESPUESTA ESPERADA DE LA APLICACIÓN	COINCIDE		RESPUESTA DEL SISTEMA
		SI	NO	
1. Seleccionar opción de administración de Productos	Desplegar plantilla de productos	X		
2. Consulta de datos de Productos.				
Datos Correctos	Se presenta una tabla con los datos de los productos existentes.	X		
Datos Incorrectos	Se presenta un mensaje de error.	X		
3. Edición de datos de Producto.				
Datos Correctos	Se actualizan los datos del Producto.	X		
Datos Incorrectos	Se presentan mensajes de error.	X		
4. Registro de datos de Producto.				
Datos Correctos	Se crea el registro del Producto.	X		
Datos Incorrectos	Se presentan mensajes de error.	X		

c. Post condiciones	
La base de datos debe quedar actualizada.	
2. RESULTADOS DE LA PRUEBA	
Defectos y desviaciones	Veredicto
NA	Falló
Observaciones / Modificaciones Sugeridas	Probador
NA	<b>Firma:</b> <b>Nombre: Julio Alberto Lárraga</b> <b>Fecha: 2013/04/18</b>

**Tabla IV – 26:** Prueba de Caso de Uso (Administración de Productos)

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

INFORMACIÓN GLOBAL DE LA PRUEBA			
<b>CASO DE USO:</b>	F4	<b>VERSIÓN DE EJECUCIÓN</b>	1.0
		<b>FECHA EJECUCIÓN</b>	18/04/2013
	Administración de Clientes	<b>MODULO DEL SISTEMA</b>	Administración
<b>Descripción del caso de prueba:</b>	Consulta, ingreso y edición de clientes.		
1. CASO DE USO			
a. Precondiciones			
Información de clientes.			
b. Pasos de la prueba			
PASOS	RESPUESTA ESPERADA	COINCIDE	RESPUESTA DEL SISTEMA

	DE LA APLICACIÓN	SI	NO	
1. Seleccionar opción administración de Clientes	Desplegar plantilla de clientes	X		
2. Consulta de datos de Clientes.				
Datos Correctos	Se presenta una tabla con los datos de los clientes existentes.	X		
Datos Incorrectos	Se presenta un mensaje de error.	X		
3. Edición de datos de Cliente.				
Datos Correctos	Se actualizan los datos del Cliente.	X		
Datos Incorrectos	Se presentan mensajes de error.	X		
4. Registro de datos de Cliente.				
Datos Correctos	Se crea el registro del Cliente.	X		
Datos Incorrectos	Se presentan mensajes de error.	X		
<b>c. Post condiciones</b>				
La base de datos debe quedar actualizada.				
<b>2. RESULTADOS DE LA PRUEBA</b>				
<b>Defectos y desviaciones</b>				<b>Veredicto</b>
NA				<b>Pasó</b>
<b>Observaciones / Modificaciones Sugeridas</b>			<b>Probador</b>	
Se han realizado todas las modificaciones pertinentes en cuanto a errores de persistencia y búsqueda de datos.				
			<b>Firma:</b>	

Nombre: Julio Alberto Lárraga

Fecha: 2013/04/18

**Tabla IV – 27:** Prueba de Caso de Uso (Administración de Clientes)

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

INFORMACIÓN GLOBAL DE LA PRUEBA		
<b>VERSIÓN DE EJECUCIÓN</b>	1.0	
<b>FECHA EJECUCIÓN</b>	2013-04-18	
<b>MODULO DEL SISTEMA</b>	Administración	
1. PRUEBA DE USABILIDAD		
PREGUNTA	CRITERIOS DE EVALUACIÓN	RESPUESTA
1. ¿Es simple el vocabulario utilizado?	1 = El vocabulario es demasiado técnico. 2 = El vocabulario presenta algunas dificultades de comprensión. 3 = El vocabulario es completamente comprensible.	3
2. ¿Hay algún tipo de asistencia para los usuarios que hacen uso del sistema por primera vez?	1 = No existe ninguna ayuda. 2 = Se encuentra ayuda en algunas partes. 3 = Existen ayudas en todo el sistema.	2
3. ¿El sistema es fácil de operar para alguien que no recibió capacitación en su operación?	1 = El sistema es de difícil comprensión. 2 = El sistema es fácil de operar en algunas de sus funcionalidades.	3

	3 = El sistema es completamente fácil de operar.	
4. ¿Se entienden la interfaz y su contenido?	1 = No se entiende su interfaz. 2 = La interfaz se entiende en algunas partes. 3 = La interfaz es completamente entendible.	<b>3</b>
5. ¿Resulta fácil identificar un objeto o una acción?	1 = Es difícil identificar los objetos o acciones. 2 = Se pueden identificar los objetos y acciones en algunas partes del sistema. 3 = Todos los objetos y acciones son fácilmente identificables.	<b>3</b>
6. ¿Resulta fácil entender el resultado de una acción?	1 = Los resultados de las acciones no son entendibles. 2 = Los resultados de las acciones son entendibles en algunas partes o la mayor parte del sistema. 3 = Todos los resultados de las acciones son entendibles.	<b>3</b>
7. ¿Está diseñada la interfaz para facilitar la realización eficiente de las tareas de la mejor forma posible?	1 = La interfaz es difícil de usar. 2 = La interfaz es difícil de usar en algunas partes del sistema. 3 = La interfaz es completamente sencilla de usar.	<b>2</b>
8. ¿Son apropiados los mensajes presentado por el sistema?	1 = Los mensajes no son apropiados. 2 = Los mensajes son apropiados en algunas partes del sistema. 3 = Todos los mensajes son apropiados y fáciles de comprender.	<b>3</b>
9. ¿Actúa el sistema en la prevención de errores?	1 = El sistema no previene errores del usuario.	<b>2</b>

	<p>2 = El sistema previene algunos o la mayoría de los errores del usuario.</p> <p>3 = El sistema previene cualquier error que pueda cometer el usuario.</p>	
<p>10. ¿El sistema informa claramente sobre los errores presentados?</p>	<p>1 = El sistema no informa de manera adecuada sobre los errores cometidos.</p> <p>2 = El sistema informa de manera adecuada algunos o la mayoría de los errores cometidos por el usuario.</p> <p>3 = El sistema informa de forma adecuada todos los errores cometidos por el usuario.</p>	<b>3</b>
<p>11. ¿Se utiliza mensajes y textos descriptivos?</p>	<p>1 = Los mensajes de texto no son descriptivos.</p> <p>2 = La mayoría de los textos son descriptivos o fáciles de interpretar</p> <p>3 = Todos los textos son descriptivos o fáciles de interpretar.</p>	<b>2</b>
<p>12. ¿Se proporciona información visual de dónde está el usuario, qué está haciendo y qué puede hacer a continuación?</p>	<p>1 = No se presenta ninguna información visual ni otro tipo de ayuda.</p> <p>2 = Presenta ayudas en algunas partes del sistema.</p> <p>3 = Las ayudas son apropiadas y están distribuidas a lo largo del sistema.</p>	<b>3</b>
<b>Observaciones / Modificaciones Sugeridas</b>		<b>Probador</b>
<p>Se han realizado modificaciones de estilos en los html, así como en los mensajes del sistema.</p> <p>La gráfica y color del diseño tiene sus limitantes, ya que va de acuerdo a la imagen de la empresa.</p>		
		<b>Firma:</b>

	<b>Nombre:</b>
	<b>Fecha:</b> 2013-04-18

**Tabla IV - 28:** Resultado de la Prueba

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.1.5. POSTMORTEM

Se considera vital la retroalimentación como parte final del Ciclo 1, pues nos dará un aprendizaje inicial en el siguiente ciclo. Es así que la Fase de Postmortem tendrá como objetivo realizar una evaluación de la calidad del proyecto, se hará un análisis de resultados desde el punto de vista del sistema y cada una de las fases, así como también de la interacción de los desarrolladores entre ellos y entre el cliente.

##### 4.1.5.1. Análisis de Resultados

El análisis de resultados se realizará por cada una de las fases del ciclo con la finalidad de que en el siguiente ciclo se pueda tomar estas sugerencias o comentarios por fase.

Fase	Comentario
<b>Análisis de Requerimientos</b>	· Los casos de uso deben ser lo más puntuales posibles, no deben incluir el detalle de cada administración
	· Es mucho más didáctico realizar un flujo para explicar el comportamiento de cada caso de uso.
	· Los diagramas de casos de uso realizables son los conectores para comprender como los desarrolladores transformar los requerimientos del cliente en los diagramas de secuencia.
<b>Diseño del Sistema</b>	· En los diagramas de secuencia es crucial poner todos los controladores que interactuaran en el sistema, pues clarifican el desarrollo del aplicativo.
	· Es fundamental dedicar el tiempo a diseñar el sistema con todos sus detalles, pues un buen diseño de un sistema hace que perdure en el tiempo y que únicamente

	<p>se realicen ajuste de forma más no de fondo.</p>
	<ul style="list-style-type: none"> <li>· El diseño de los formatos de pruebas son importantes ya que una vez listo el sistema se podrán ejecutarlas.</li> </ul>
<b>Construcción y pruebas</b>	<ul style="list-style-type: none"> <li>· La transformación del Modelo E/R al Modelo Relacional clarifica las reglas que se usaran para ir de la fase de diseño a la fase de construcción.</li> </ul>
	<ul style="list-style-type: none"> <li>· Es necesario estandarizar las estrategias de programación ya que clarifica el entendimiento del código y facilita la programación de los módulos.</li> </ul>
	<ul style="list-style-type: none"> <li>· El catálogo de piezas de desarrollo orientará las partes de la programación es preciso el orden en las mismas.</li> </ul>
	<ul style="list-style-type: none"> <li>· La aplicación de las pruebas de Desarrollo son clave en esta fase mientras que las pruebas de usabilidad se ejecutarán en la fase de implantación.</li> </ul>
<b>Implantación</b>	<ul style="list-style-type: none"> <li>· El cronograma de implantación debe contener los entregables tanto del análisis de software como del acta de entrega/recepción del sistema.</li> </ul>
	<ul style="list-style-type: none"> <li>· El manual técnico debe estar muy bien documentado, pues al ser el sistema de propiedad del cliente, el administrador debe entenderlo fácilmente.</li> </ul>
	<ul style="list-style-type: none"> <li>· El entrenamiento debe realizarse con todos los usuarios y el acompañamiento en los primeros días es fundamental para despejar dudas y detectar posibles fallas.</li> </ul>
	<ul style="list-style-type: none"> <li>· El soporte por parte de los desarrolladores cubrirá los seis primeros meses, en este periodo se dará pronta respuesta a las fallas detectadas y documentadas bajo el formato establecido.</li> </ul>

**Tabla IV - 29:** Diagrama Ciclo 1

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

Una vez analizadas las oportunidades de mejora en cada fase, se desea analizar la interacción con el cliente y entre los desarrolladores, para optimizar recursos en la ejecución del ciclo 2. A continuación los siguientes comentarios.

- La comunicación continua con el cliente fue muy importante, para ello una oportuna planificación fue esencial.
- La clara documentación de los elementos de aprobación ayudo para mantener una relación clara con el cliente.
- La disponibilidad incondicional por parte de las partes fue un punto a favor para cumplir con los tiempos establecidos.
- El manejo de estándares entre los desarrolladores ayudo a mantener durante todo el proyecto una clara comunicación tanto de la programación como de toda la documentación generada.
- En un principio no se realizó un plan de visibilidad con actividades, entregables y responsables, pero en la mitad del ciclo al detectar bajo cumplimiento en las tareas encomendadas, se vio la necesidad de alimentar éste plan con su respectivo cronograma cada semana.

## 4.2. CICLO 2

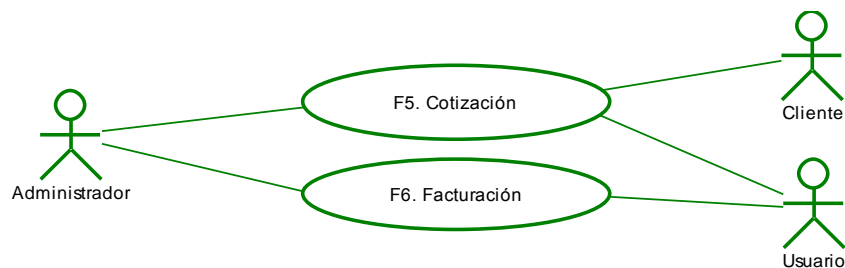
Cumpliendo con la metodología de mini cascada, y después de recibir retroalimentación por parte del cliente en el Ciclo 2 se trabajara los procesos que contendrá este aplicativo web, es así que las siguientes fases estarán enfocadas en la Facturación y en la Cotización.

### 4.2.1. ANALISIS DE REQUERIMIENTOS

Se analizara ahora los requerimientos en cuento a los procesos que el sistema contará, estos requerimientos serán recibidos por parte del cliente y se tratara de transformarlos en funcionalidades especificas y que puedan plasmar la necesidad de la empresa en su día a día.

#### 4.2.1.1. Diagrama General

Este diagrama contendrá únicamente los procesos que en el presente ciclo se tratara.



**Figura IV - 25:** Diagrama Ciclo 2

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.2.1.2. Diagrama de Casos de Uso

Los diagramas de casos de uso de los procesos serán de igual forma que las Administraciones del ciclo anterior, un primer ensayo de transformar los requerimientos del cliente en diagramas claros.

##### 4.2.1.2.1 Proceso de Cotización

**Caso de Uso:** Cotización

**Identificador:** F5

El proceso a continuación se refiere a que los clientes de la empresa tengan acceso a realizar cotizaciones de los productos que dispone la compañía y de esta forma pueda analizar el precio previo a su adquisición.



**Figura IV - 26:** Diagrama de Proceso de Cotización

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

La Tabla IV – 1 explicará claramente el flujo que el autor debe hacer en este proceso.

**Caso de Uso:** Cotización

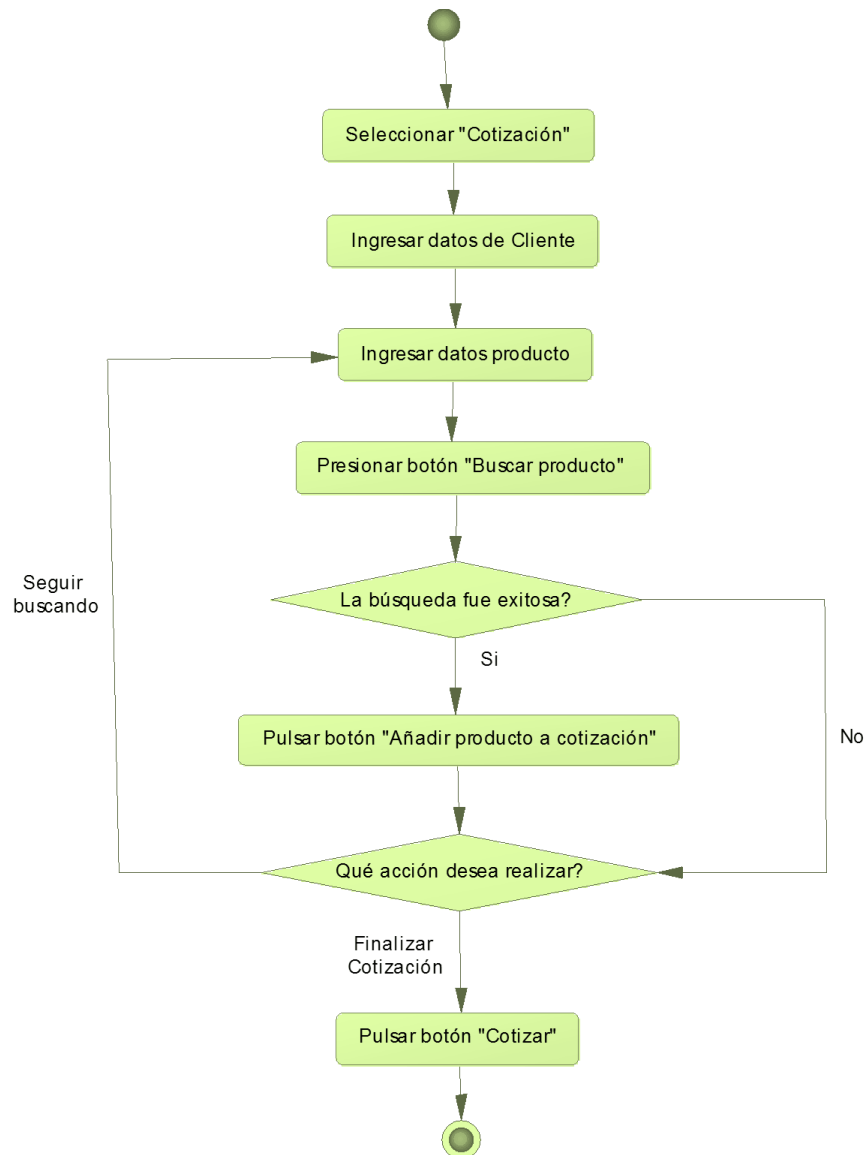
**Identificador:** F5

**Actores:** Administrador / Usuario / Cliente

**Pre-condición:**

Los clientes desean averiguar los precios de los productos antes de adquirirlos.

**Flujo:**



**Pos-condición:**

Se muestra la información requerida

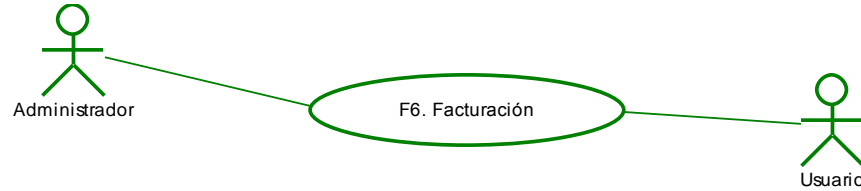
**Tabla IV – 30:** Tabla de Flujo para Cotización

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

4.2.1.2.3. Proceso de Facturación

**Caso de Uso: Facturación**

**Identificador: F6**



**Figura IV - 27:** Diagrama del Proceso de Facturación

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

Las tareas de facturación se deberán realizar por parte de los actores que interactúan con el aplicativo para el manejo de usuarios. Por lo tanto, este proceso debe regirse a un flujo realizado por el usuario el cual se puede determinar de manera correcta en la Tabla IV - 2.

**Caso de Uso:** Facturación

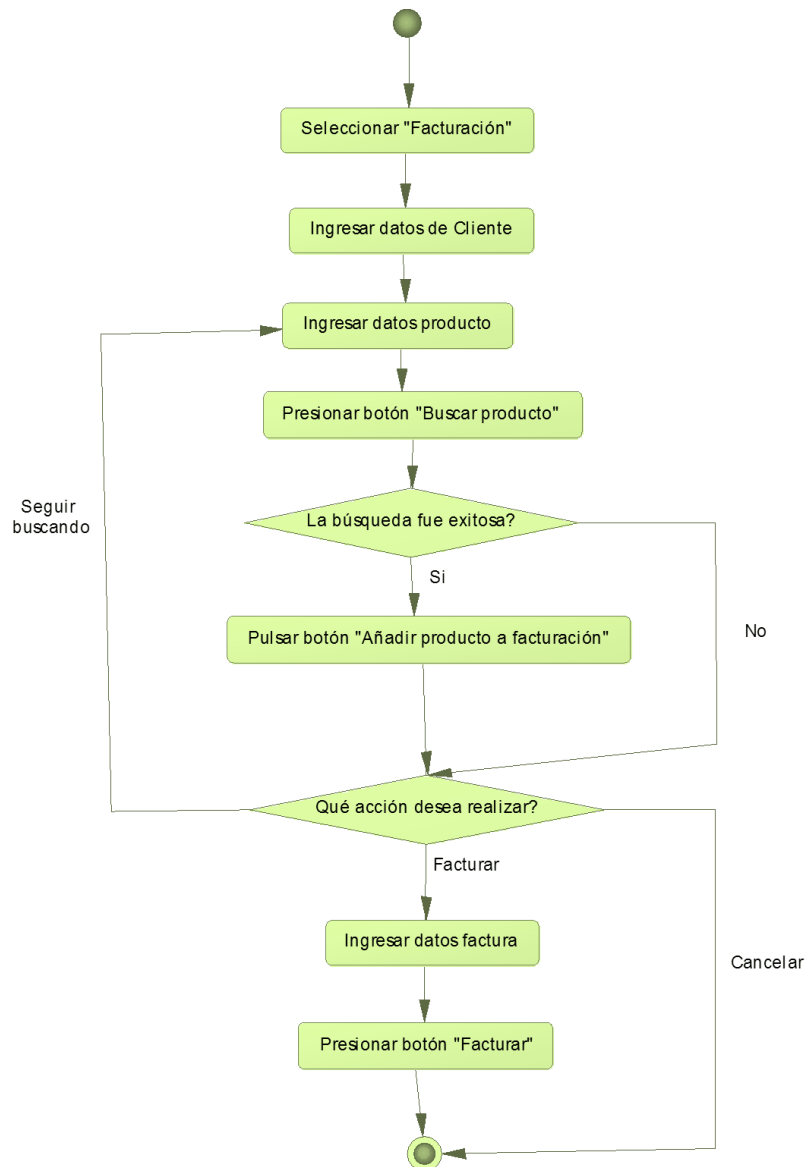
**Identificador:** F6

**Actores:** Administrador / Usuario

**Pre-condición:**

Cliente adquiera producto de la empresa.

**Flujo:**



**Pos-condición:**

La base queda actualizada

Se genera un documento físico que es la factura.

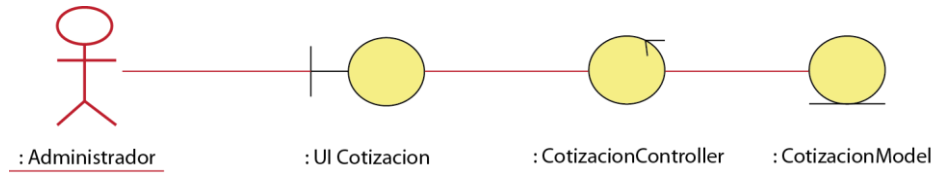
**Tabla IV – 31:** Tabla de Flujo para la Facturación

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.2.1.3. Diagrama de Casos de Uso Realizables

**Caso de Uso:** Cotización

**Identificador:** F5

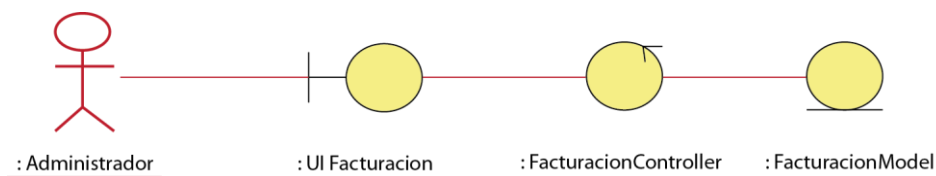


**Figura IV – 28:** Diagrama de Proceso de Cotización

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

**Caso de Uso:** Facturación

**Identificador:** F6



**Figura IV – 29:** Diagrama del Proceso de Facturación

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.2.2. DISEÑO DEL SISTEMA

##### 4.2.2.1. Diagramas de Secuencia

Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada caso de uso. A continuación se mostrará cada diagrama de secuencia:

Caso de Uso: Cotización

Identificador: F5

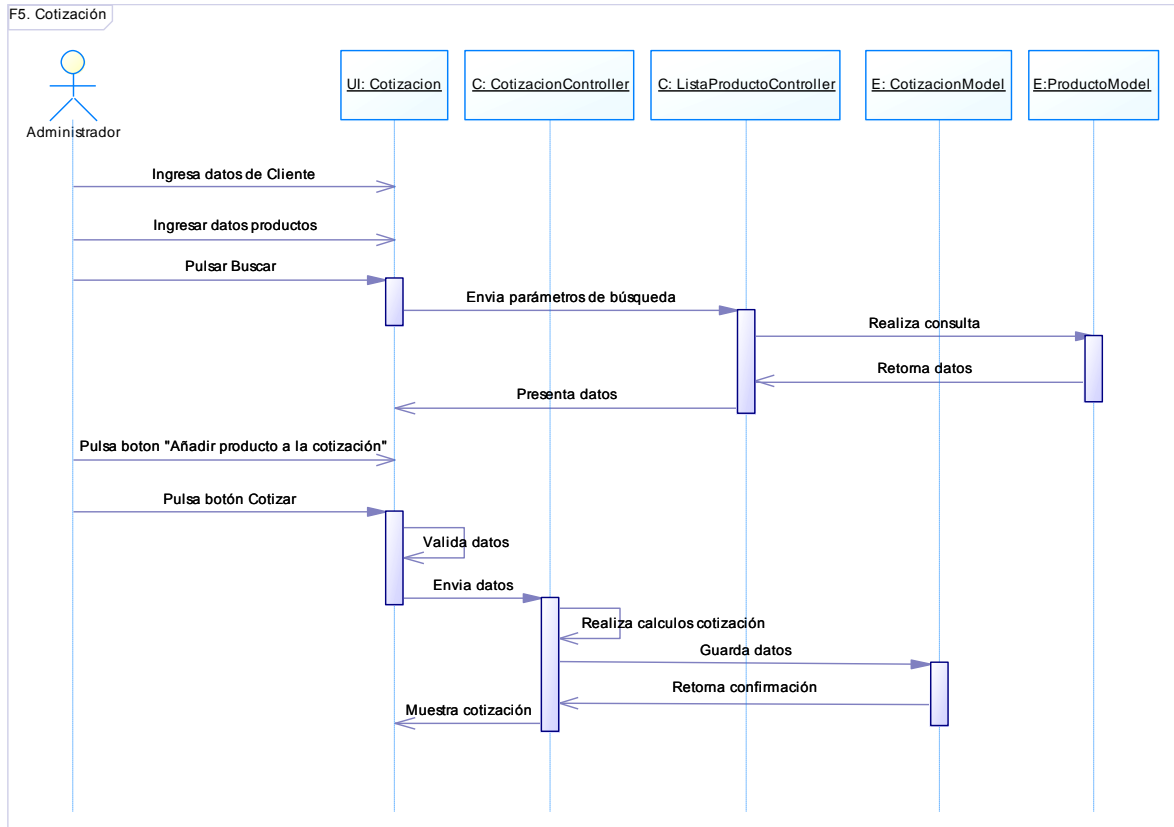


Figura IV – 30: Diagrama de Secuencia para el proceso de Cotización

Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

Caso de Uso: Facturación

Identificador: F6

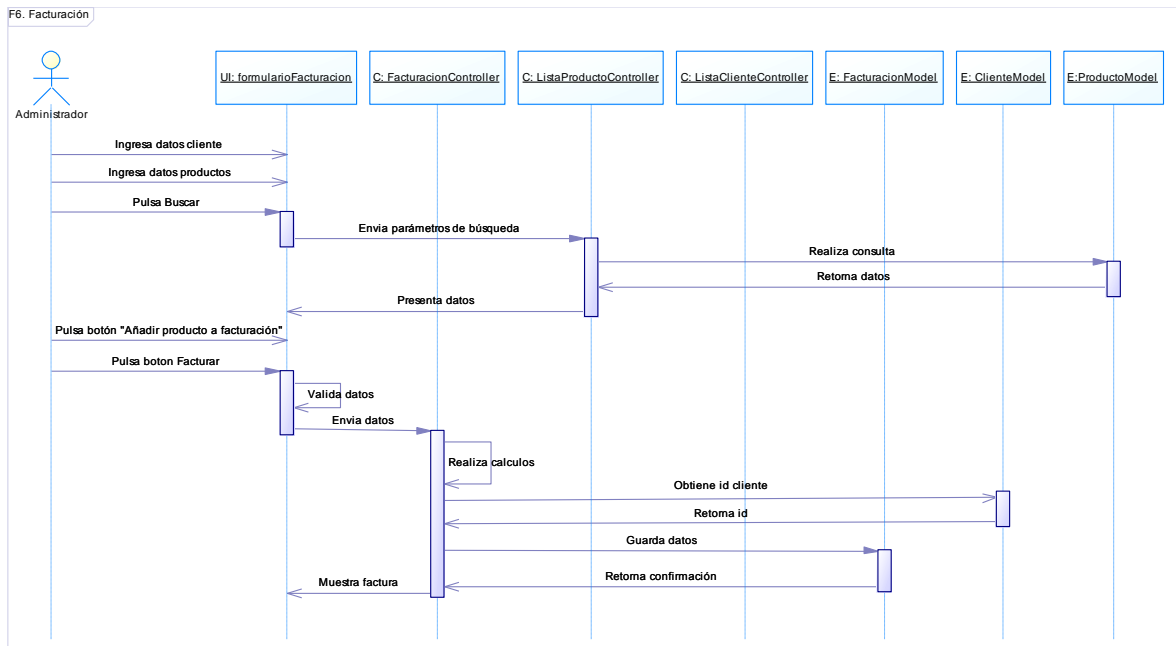


Figura IV – 31: Diagrama de Secuencia para el proceso de facturación

Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

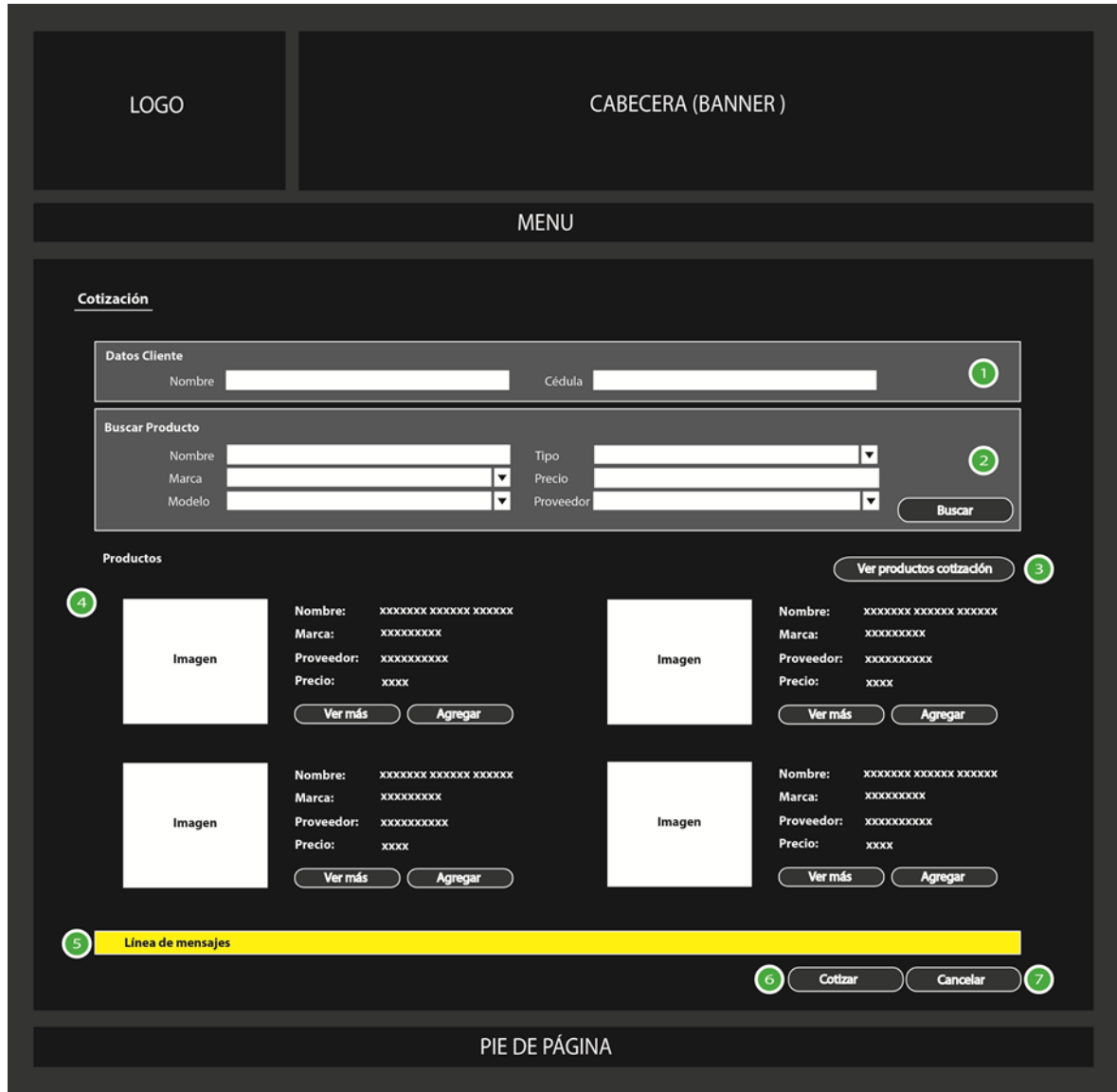
4.2.2.2. Diseño de Clases

4.2.2.2.1. Diseño UI

Caso de Uso: Cotización

Identificador: F5

Interfaz: UI\_F5\_01\_Cotizacion



**Figura IV – 32:** Diseño de interfaz para el proceso de Cotización

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

1. Datos Cliente. Datos primordiales y opcionales del cliente. Solo para registro, no actúa como relación con la cotización.
2. Buscar Producto. Datos para la búsqueda de productos.
3. Ver productos cotización. Ver los productos agregados a la cotización en un cuadro de dialogo que también permitirá quitarlos.

4. Productos resultados de la búsqueda, disponibles para agregarlos a la cotización.
5. Línea de mensajes. Área de la interfaz donde se presentan los mensajes del sistema.
6. Botón “Cotizar”. El sistema mostrará el mensaje de confirmación de cotización.
7. Botón “Cancelar”. El sistema mostrará el mensaje de confirmación de cancelación.

**Caso de Uso:** Facturación

**Identificador:** F6

**Interfaz:** UI\_F6\_01\_Facturación

The wireframe illustrates the layout for the 'Facturación' (Billing) process. It is organized into several key sections:

- LOGO** and **CABECERA (BANNER)**: Header area at the top.
- MENU**: A horizontal navigation bar below the header.
- Facturación**: The main content area, containing:
  - Datos Cliente**: A form with fields for Nombre, Dirección, Cédula, and Teléfono.
  - Buscar Producto**: A search form with fields for Nombre, Marca, Modelo, Tipo, Precio, and Proveedor, along with a 'Buscar' button.
  - Productos**: A grid of product cards. Each card includes an 'Imagen' placeholder, fields for Nombre, Marca, Proveedor, and Precio, and buttons for 'Ver más' and 'Agregar'. A 'Ver productos facturación' button is also present.
  - Summary**: A table at the bottom right showing Subtotal, IVA, and Total.
  - Línea de mensajes**: A yellow horizontal bar at the bottom of the main content area.
  - Buttons**: 'Facturar' and 'Cancelar' buttons located at the bottom right of the interface.
- PIE DE PÁGINA**: Footer area at the very bottom.

**Figura IV – 33:** Diseño de interfaz para el proceso de Facturación

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

1. Datos Cliente. Datos primordiales del cliente para la emisión de la factura.
2. Buscar Producto. Datos para la búsqueda de productos.
3. Ver productos facturación. Ver los productos agregados a la facturación en un cuadro de dialogo que también permitirá quitarlos.
4. Productos resultados de la búsqueda, disponibles para agregarlos a la facturación.
5. Línea de mensajes. Área de la interfaz donde se presentan los mensajes del sistema.
6. Botón “Facturar”. El sistema mostrará el mensaje de confirmación de facturación.
7. Botón “Cancelar”. El sistema mostrará el mensaje de confirmación de cancelación.

#### 4.2.2.2.2. Diseño de Clases de Control

Caso de Uso	Control	Nombre	Descripción
F5	C_F5_01_CotizacionController	CotizacionController	Contiene las funciones necesarias para el proceso de cotización.
F6	C_F6_02_FacturacionController	FacturacionController	Contiene las funciones necesarias para el proceso de facturación.

**Tabla IV – 34:** Diseño de Clases de Control

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.2.2.2.3. Diseño de Clases Entidad

En el ciclo 1 se definió el diseño de clases entidad para los dos ciclos.

#### 4.2.2.3. Plan de Pruebas

En el ciclo 1 se definió el plan de pruebas para los dos ciclos.

### 4.2.3. CONSTRUCCIÓN Y PRUEBAS

Considerando que en el ciclo 1 ya se seleccionaron las herramientas de desarrollo, se elaboró el modelo conceptual, modelo físico, diccionario de datos y se definió los estándares y estrategias para los dos ciclos. En este ciclo en la fase de construcción y pruebas se elaborará únicamente el catalogo de piezas de software junto con las pruebas en ambiente de desarrollo.

4.2.3.1. Catálogo de piezas de software construidos

Nombre	Tipo	Descripción
<b>controllers</b>	directorio	Capa de controladores de la aplicación.
cotizacion.php	archivo PHP	Controlador que contiene todas las funciones necesarias para el proceso de cotización.
facturacion.php	archivo PHP	Controlador que contiene todas las funciones necesarias para el proceso de facturación.
<b>models</b>	directorio	Capa de modelos de la aplicación.
cotizacion_model.php	archivo PHP	Modelo que maneja los datos de las cotizaciones.
facturacion_model.php	archivo PHP	Modelo que maneja los datos de las facturas.
<b>views</b>	directorio	Capa de vistas de la aplicación.
cotizacionForm_view.php	archivo PHP	Archivo que muestra la pantalla de cotización.
cotizacionPrint_view.php	archivo PHP	Archivo que muestra una versión imprimible o exportable de la cotización.
facturacionForm_view.php	archivo PHP	Archivo que muestra la pantalla de facturación.
facturacionPrint_view.php	archivo PHP	Archivo que muestra una versión imprimible o exportable de la factura.

**Tabla IV – 35:** Catálogo de piezas de software construidos  
**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

4.2.3.2. Aplicación Pruebas

4.2.3.2.1. Pruebas en Ambiente de Desarrollo

Las pruebas funcionales en ambiente de desarrollo fueron ejecutadas dos por cada caso de uso, sin embargo a continuación se muestran una por cada caso de uso y las demás se encontraran en el Anexo IV.

INFORMACIÓN GLOBAL DE LA PRUEBA			
<b>CASO DE USO:</b>	F5	<b>VERSIÓN DE</b>	1.0

		<b>EJECUCIÓN</b>		
		<b>FECHA EJECUCIÓN</b>		20/04/2013
	Cotización	<b>MODULO DEL SISTEMA</b>		Administración
<b>Descripción del caso de prueba:</b>	Proceso de cotización de productos.			
<b>1. CASO DE USO</b>				
<b>a. Precondiciones</b>				
Ninguna.				
<b>b. Pasos de la prueba</b>				
PASOS	RESPUESTA ESPERADA DE LA APLICACIÓN	COINCIDE		RESPUESTA DEL SISTEMA
		SI	NO	
1. Seleccionar opción Cotización	Desplegar plantilla de cotización	X		
2. Consulta de datos de Productos.				
Datos Correctos	Se presenta una tabla con los datos de los productos existentes.	X		
Datos Incorrectos	Se presenta un mensaje de error.	X		
3. Añadir producto a cotización.	Se añade el producto a la cotización.	X		
4. Cotizar	Se ejecuta el proceso de cotización y muestra una pantalla con el detalle de la cotización.		X	El sistema presenta un error no estandarizado al presionar el botón de Cotizar
5. Imprimir Cotización	Imprime la cotización.		X	No se pudo a llegar a esta opción puesto que no funcionó el cotizar
6. Exportar a PDF Cotización	Exporta a PDF la cotización.		X	No se pudo a llegar a esta opción puesto que no funcionó el cotizar

<b>c. Post condiciones</b>	
La base de datos debe quedar actualizada.	
<b>2. RESULTADOS DE LA PRUEBA</b>	
<b>Defectos y desviaciones</b>	<b>Veredicto</b>
Al ejecutar el proceso de cotización, el sistema no desplego la pantalla esperada, sino que mostro una pantalla con un error no estandarizado.	<b>Falló</b>
<b>Observaciones / Modificaciones Sugeridas</b>	<b>Probador</b>
Revisar el código del controlador de cotizaciones, ya que no se ejecuto el proceso. El fallo puede estar en alguna función o en el archivo modelo de acceso a la base de datos.	<b>Firma:</b> <b>Nombre: Julio Alberto Lárraga</b> <b>Fecha: 2013/04/20</b>

**Tabla IV –36:** Prueba de Caso de Uso (Proceso de cotización)

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

<b>INFORMACIÓN GLOBAL DE LA PRUEBA</b>			
<b>CASO DE USO:</b>	F6  Proceso de facturación	<b>VERSIÓN DE EJECUCIÓN</b>	1.0
		<b>FECHA EJECUCIÓN</b>	20/04/2013
		<b>MODULO DEL SISTEMA</b>	Administración
<b>Descripción del caso de prueba:</b>	Proceso de facturación de productos.		
<b>1. CASO DE USO</b>			
<b>a. Precondiciones</b>			
Información de cliente y productos.			
<b>b. Pasos de la prueba</b>			
<b>PASOS</b>	<b>RESPUESTA ESPERADA</b>	<b>COINCIDE</b>	<b>RESPUESTA DEL SISTEMA</b>

	DE LA APLICACIÓN	SI	NO	
1. Seleccionar opción Facturación	Desplegar plantilla de facturación	X		
2. Consulta de datos de Clientes.				
Datos Correctos	Se presenta un campo de autocompletado con los nombres de los clientes registrados en la BDD.	X		
Datos Incorrectos	Se presenta un mensaje de error.	X		
3. Consulta de datos de Productos.				
Datos Correctos	Se presenta una tabla con los datos de los productos existentes.	X		
Datos Incorrectos	Se presentan mensajes de error.	X		
4. Añadir producto a facturación.	Se añade el producto a la facturación.		X	No se agrega el producto a la facturación.
5. Facturar	Se ejecuta el proceso de cotización y muestra una pantalla con el detalle de la facturación.		X	Al ejecutar el botón facturar se presenta una pantalla en blanco.
6. Imprimir factura	Imprime la facturación.		X	No se muestra la opción de imprimir
7. Exportar a PDF factura	Exporta a PDF la facturación.		X	No se muestra la opción de exportar a PDF
<b>c. Post condiciones</b>				
La base de datos debe quedar actualizada.				
<b>2. RESULTADOS DE LA PRUEBA</b>				
<b>Defectos y desviaciones</b>				<b>Veredicto</b>

Al ejecutar el proceso de facturación aparece una pantalla en blanco sin mensaje alguno.	<b>Falló</b>
<b>Observaciones / Modificaciones Sugeridas</b>	<b>Probador</b>
Revisar el código del controlador de facturación y el código ajax en el HTML del formulario de cotización ya que no se agregan los productos.	<b>Firma:</b> <b>Nombre: Julio Alberto Lárraga</b> <b>Fecha: 2013/04/20</b>

**Tabla IV – 37:** Prueba de Caso de Uso (Proceso de cotización)

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.2.4. IMPLANTACIÓN

En el ciclo 1 se describió el conjunto de tareas necesarias para instalar y probar el modulo de administraciones del sistema.

En este ciclo, la presente fase a través de un plan de implantación, busca definir la lista de tareas o agenda de implantación que incluye la preparación del ambiente para la actualización y ejecución del sistema total, el entrenamiento de los usuarios y detalles de los futuros procedimientos de soporte, todo esto a través de un cronograma de actividades y responsables necesarios para la oportuna implantación.

##### 4.2.4.1. Plan de Implantación

##### 4.2.4.1.2. Responsabilidades

Las tareas o responsabilidades que deberá llevar a cabo el equipo, se enfocan en la realización de pruebas del sistema, del modulo desarrollado en el ciclo 2.

Una vez que se haya subido el servidor web la versión final del sistema, se pedirá al administrador

y a los usuarios del sistema la realización de las pruebas de usabilidad, para posteriormente realizar los ajustes pertinentes en el sistema.



#### 4.2.4.1.3. Cronograma

La implantación del sistema en Lubrimotor Automotriz es responsabilidad de los desarrolladores. A continuación se va a proponer un cronograma de las actividades, y de sus entregables.

Se debe tener en consideración que el Informe de Recursos y el Acta de Entrega Recepción del Sistema que se citará continuación estarán en el Anexo IV (Informe de Recursos de Hardware y de Software y Acta de Entrega y Recepción).

**Cronograma de Implantación**

Lubrimotor Automotriz (LA)

Terminado   
 Pendiente 

➤ Entregables

Actividad	Responsable	Enero				Marzo				Abril				Mayo				Junio				
		S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	
Subir Versión Final al Servidor Web	Desarrolladores																					
Ejecutar Pruebas de Usabilidad Ciclo2	Usuarios - Administrador																					
Realizar Ajustes	Desarrolladores																					
Subir versión final a Servidor Web	Desarrolladores																					
Entrenamiento a Administrador del Sistema	Desarrolladores - Administrador																					
Entrenamiento a Usuarios	Desarrolladores - Usuarios																					
➤ Manual Técnico	Desarrolladores																					
➤ Manual de Usuarios	Desarrolladores																					
➤ Acta Entrega Recepción de Sistema	Desarrolladores - LA																					

**Tabla IV – 38:** Cronograma de Implantación Ciclo 2  
 Elaborado por: María Gabriela Bedón, Julio Alberto Lárraga

#### 4.2.4.1.4. Documentación de Soporte

##### 4.2.4.1.4.1. Manual Técnico

Se aumentarán las especificaciones técnicas correspondientes al ciclo, en el manual técnico creado en el Ciclo 1. El técnico que desee conocer la estructura interna del software podrá hacerlo fácilmente explorando el archivo ManualTecnico.docx.

##### 4.2.4.1.4.2. Manual de Usuario

Se aumentarán las instrucciones de uso de los procesos desarrollados al manual de Usuario en el Ciclo 1.

##### 4.2.4.1.4.3. Personal de Soporte

El personal de soporte será el mismo definido en la fase de implantación del ciclo 1.

##### 4.2.4.1.4. Entrenamiento

Se tiene pensado que el entrenamiento total del administrador y de los usuarios se haga durante cuatro días, y que posteriormente ellos por su parte sigan practicando y puedan así conocer el sistema. El primer entrenamiento lo recibirá el Administrador del Sistema designado previamente por la empresa y posteriormente los usuarios complementarios que harán uso del sistema, para explicarlo la Tabla IV - 39.

Es importante que durante el entrenamiento y soporte de los usuarios del sistema, el dueño y el administrador del sistema estén presentes. En estos cuatro días se realizará acompañamiento a los usuarios en el manejo del aplicativo en las actividades diarias.

<b>Rol</b>	<b>Descripción</b>	<b>Tiempo</b>
<b>Administrador</b>	Este actor se encargara del soporte posterior por lo cual el entrenamiento será enfocado a superuser del sistema	16 horas

<b>Usuarios</b>	El entrenamiento tendrá como foco el de 16 horas
<b>Finales</b>	usuarios finales, que usan el sistema en el día a día.

**Tabla IV - 39:** Formato de Reporte de Fallos

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

4.1.4.2. Pruebas con Usuarios

4.1.4.2.1. Resultados de las Pruebas

INFORMACIÓN GLOBAL DE LA PRUEBA		
<b>VERSIÓN DE EJECUCIÓN</b>	1.0	
<b>FECHA EJECUCIÓN</b>	2013-04-24	
<b>MODULO DEL SISTEMA</b>	Administración	
1. PRUEBA DE USABILIDAD		
PREGUNTA	CRITERIOS DE EVALUACIÓN	RESPUESTA
1. ¿Es simple el vocabulario utilizado?	1 = El vocabulario es demasiado técnico. 2 = El vocabulario presenta algunas dificultades de comprensión. 3 = El vocabulario es completamente comprensible.	3
2. ¿Hay algún tipo de asistencia para los usuarios que hacen uso del sistema por primera vez?	1 = No existe ninguna ayuda. 2 = Se encuentra ayuda en algunas partes. 3 = Existen ayudas en todo el sistema.	1

<p>3. ¿El sistema es fácil de operar para alguien que no recibió capacitación en su operación?</p>	<p>1 = El sistema es de difícil comprensión.                  2 = El sistema es fácil de operar en algunas de sus funcionalidades.                  3 = El sistema es completamente fácil de operar.</p>	<p><b>2</b></p>
<p>4. ¿Se entienden la interfaz y su contenido?</p>	<p>1 = No se entiende su interfaz.                  2 = La interfaz se entiende en algunas partes.                  3 = La interfaz es completamente entendible.</p>	<p><b>2</b></p>
<p>5. ¿Resulta fácil identificar un objeto o una acción?</p>	<p>1 = Es difícil identificar los objetos o acciones.                  2 = Se pueden identificar los objetos y acciones en algunas partes del sistema.                  3 = Todos los objetos y acciones son fácilmente identificables.</p>	<p><b>1</b></p>
<p>6. ¿Resulta fácil entender el resultado de una acción?</p>	<p>1 = Los resultados de las acciones no son entendibles.                  2 = Los resultados de las acciones son entendibles en algunas partes o la mayor parte del sistema.                  3 = Todos los resultados de las acciones son entendibles.</p>	<p><b>2</b></p>
<p>7. ¿Está diseñada la interfaz para facilitar la realización eficiente de las tareas de la mejor forma posible?</p>	<p>1 = La interfaz es difícil de usar.                  2 = La interfaz es difícil de usar en algunas partes del sistema.                  3 = La interfaz es completamente sencilla de usar.</p>	<p><b>1</b></p>
<p>8. ¿Son apropiados los mensajes presentado por el sistema?</p>	<p>1 = Los mensajes no son apropiados.                  2 = Los mensajes son apropiados en algunas partes del sistema.</p>	<p><b>1</b></p>

	3 = Todos los mensajes son apropiados y fáciles de comprender.	
9. ¿Actúa el sistema en la prevención de errores?	1 = El sistema no previene errores del usuario. 2 = El sistema previene algunos o la mayoría de los errores del usuario. 3 = El sistema previene cualquier error que pueda cometer el usuario.	2
10. ¿El sistema informa claramente sobre los errores presentados?	1 = El sistema no informa de manera adecuada sobre los errores cometidos. 2 = El sistema informa de manera adecuada algunos o la mayoría de los errores cometidos por el usuario. 3 = El sistema informa de forma adecuada todos los errores cometidos por el usuario.	2
11. ¿Se utiliza mensajes y textos descriptivos?	1 = Los mensajes de texto no son descriptivos. 2 = La mayoría de los textos son descriptivos o fáciles de interpretar 3 = Todos los textos son descriptivos o fáciles de interpretar.	1
12. ¿Se proporciona información visual de dónde está el usuario, qué está haciendo y qué puede hacer a continuación?	1 = No se presenta ninguna información visual ni otro tipo de ayuda. 2 = Presenta ayudas en algunas partes del sistema. 3 = Las ayudas son apropiadas y están distribuidas a lo largo del sistema.	1
<b>Observaciones / Modificaciones Sugeridas</b>		<b>Probador</b>
Mejorar el diseño y ubicación de los componentes en las pantallas de facturación y cotización.		Mayra Andrade

Mejorar la interfaz para que sea más intuitiva y amigable al usuario.	
	<b>Firma:</b> MA
	<b>Nombre:</b> Mayra Andrade
	<b>Fecha:</b> 2013-04-15

**Tabla IV -40:** Resultado de las Pruebas

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

4.2.4.2.2. Versión Final de Pruebas

INFORMACIÓN GLOBAL DE LA PRUEBA				
<b>CASO DE USO:</b>	F5  Cotización	<b>VERSIÓN DE EJECUCIÓN</b>	1.0	
		<b>FECHA EJECUCIÓN</b>	23/04/2013	
		<b>MODULO DEL SISTEMA</b>	Administración	
<b>Descripción del caso de prueba:</b>	Proceso de cotización de productos.			
<b>1. CASO DE USO</b>				
<b>a. Precondiciones</b>				
Ninguna.				
<b>b. Pasos de la prueba</b>				
PASOS	RESPUESTA ESPERADA DE LA APLICACIÓN	COINCIDE		RESPUESTA DEL SISTEMA
		SI	NO	
1. Seleccionar opción Cotización	Desplegar plantilla de cotización	X		
2. Consulta de datos de				

Productos.				
Datos Correctos	Se presenta una tabla con los datos de los productos existentes.	X		
Datos Incorrectos	Se presenta un mensaje de error.	X		
3. Añadir producto a cotización.	Se añade el producto a la cotización.	X		
4. Cotizar	Se ejecuta el proceso de cotización y muestra una pantalla con el detalle de la cotización.	X		
5. Imprimir Cotización	Imprime la cotización.	X		
6. Exportar a PDF Cotización	Exporta a PDF la cotización.	X		
<b>c. Post condiciones</b>				
La base de datos debe quedar actualizada.				
<b>2. RESULTADOS DE LA PRUEBA</b>				
<b>Defectos y desviaciones</b>				<b>Veredicto</b>
				<b>Pasó</b>
<b>Observaciones / Modificaciones Sugeridas</b>				<b>Probador</b>
Todos los errores en sentencias SQL y lógica de programación fueron corregidos y debidamente comentados.				<b>Firma:</b> <b>Nombre: Julio Alberto Lárraga</b> <b>Fecha: 2013/04/23</b>

**Tabla IV – 41:** Prueba de Caso de Uso (Proceso de cotización)

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

INFORMACIÓN GLOBAL DE LA PRUEBA				
<b>CASO</b>	<b>DE</b>	F6	<b>VERSIÓN</b>	1.0

<b>USO:</b>	Proceso de facturación	<b>DE EJECUCIÓN</b>		
		<b>FECHA EJECUCIÓN</b>		23/04/2013
		<b>MODULO DEL SISTEMA</b>		Administración
<b>Descripción del caso de prueba:</b>	Proceso de facturación de productos.			
<b>1. CASO DE USO</b>				
<b>a. Precondiciones</b>				
Información de cliente y productos.				
<b>b. Pasos de la prueba</b>				
PASOS	RESPUESTA ESPERADA DE LA APLICACIÓN	COINCIDE		RESPUESTA DEL SISTEMA
		SI	NO	
1. Seleccionar opción Facturación	Desplegar plantilla de facturación	X		
2. Consulta de datos de Clientes.				
Datos Correctos	Se presenta un campo de autocompletado con los nombres de los clientes registrados en la BDD.	X		
Datos Incorrectos	Se presenta un mensaje de error.	X		
3. Consulta de datos de Productos.				
Datos Correctos	Se presenta una tabla con los datos de los productos existentes.	X		
Datos Incorrectos	Se presentan mensajes de error.	X		
4. Añadir producto a	Se añade el producto a la	X		

facturación.	facturación.			
5. Facturar	Se ejecuta el proceso de cotización y muestra una pantalla con el detalle de la facturación.	X		
6. Imprimir factura	Imprime la facturación.	X		
7. Exportar a PDF factura	Exporta a PDF la facturación.	X		
<b>c. Post condiciones</b>				
La base de datos debe quedar actualizada.				
<b>2. RESULTADOS DE LA PRUEBA</b>				
<b>Defectos y desviaciones</b>				<b>Veredicto</b>
				<b>Pasó</b>
<b>Observaciones / Modificaciones Sugeridas</b>			<b>Probador</b>	
El error en pruebas anteriores fue debido a inconsistencias en el código javascript y Ajax en el HTML Este fue corregido y documentado.			<b>Firma:</b> <b>Nombre: Julio Alberto Lárraga</b> <b>Fecha: 2013/04/23</b>	

**Tabla IV – 42:** Prueba de Caso de Uso (Proceso de cotización)

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

<b>INFORMACIÓN GLOBAL DE LA PRUEBA</b>	
<b>VERSIÓN DE EJECUCIÓN</b>	1.0
<b>FECHA EJECUCIÓN</b>	2013-04-25

<b>MODULO DEL SISTEMA</b>	Administración	
<b>1. PRUEBA DE USABILIDAD</b>		
<b>PREGUNTA</b>	<b>CRITERIOS DE EVALUACIÓN</b>	<b>RESPUESTA</b>
1. ¿Es simple el vocabulario utilizado?	1 = El vocabulario es demasiado técnico. 2 = El vocabulario presenta algunas dificultades de comprensión. 3 = El vocabulario es completamente comprensible.	<b>3</b>
2. ¿Hay algún tipo de asistencia para los usuarios que hacen uso del sistema por primera vez?	1 = No existe ninguna ayuda. 2 = Se encuentra ayuda en algunas partes. 3 = Existen ayudas en todo el sistema.	<b>2</b>
3. ¿El sistema es fácil de operar para alguien que no recibió capacitación en su operación?	1 = El sistema es de difícil comprensión. 2 = El sistema es fácil de operar en algunas de sus funcionalidades. 3 = El sistema es completamente fácil de operar.	<b>2</b>
4. ¿Se entienden la interfaz y su contenido?	1 = No se entiende su interfaz. 2 = La interfaz se entiende en algunas partes. 3 = La interfaz es completamente entendible.	<b>2</b>
5. ¿Resulta fácil identificar un objeto o una acción?	1 = Es difícil identificar los objetos o acciones. 2 = Se pueden identificar los objetos y acciones en algunas partes del sistema. 3 = Todos los objetos y acciones son	<b>3</b>

	fácilmente identificables.	
6. ¿Resulta fácil entender el resultado de una acción?	<p>1 = Los resultados de las acciones no son entendibles.</p> <p>2 = Los resultados de las acciones son entendibles en algunas partes o la mayor parte del sistema.</p> <p>3 = Todos los resultados de las acciones son entendibles.</p>	<b>3</b>
7. ¿Está diseñada la interfaz para facilitar la realización eficiente de las tareas de la mejor forma posible?	<p>1 = La interfaz es difícil de usar.</p> <p>2 = La interfaz es difícil de usar en algunas partes del sistema.</p> <p>3 = La interfaz es completamente sencilla de usar.</p>	<b>2</b>
8. ¿Son apropiados los mensajes presentado por el sistema?	<p>1 = Los mensajes no son apropiados.</p> <p>2 = Los mensajes son apropiados en algunas partes del sistema.</p> <p>3 = Todos los mensajes son apropiados y fáciles de comprender.</p>	<b>2</b>
9. ¿Actúa el sistema en la prevención de errores?	<p>1 = El sistema no previene errores del usuario.</p> <p>2 = El sistema previene algunos o la mayoría de los errores del usuario.</p> <p>3 = El sistema previene cualquier error que pueda cometer el usuario.</p>	<b>3</b>
10. ¿El sistema informa claramente sobre los errores presentados?	<p>1 = El sistema no informa de manera adecuada sobre los errores cometidos.</p> <p>2 = El sistema informa de manera adecuada algunos o la mayoría de los errores cometidos por el usuario.</p> <p>3 = El sistema informa de forma adecuada todos los errores cometidos por el usuario.</p>	<b>3</b>

11. ¿Se utiliza mensajes y textos descriptivos?	<p>1 = Los mensajes de texto no son descriptivos.</p> <p>2 = La mayoría de los textos son descriptivos o fáciles de interpretar</p> <p>3 = Todos los textos son descriptivos o fáciles de interpretar.</p>	<b>2</b>
12. ¿Se proporciona información visual de dónde está el usuario, qué está haciendo y qué puede hacer a continuación?	<p>1 = No se presenta ninguna información visual ni otro tipo de ayuda.</p> <p>2 = Presenta ayudas en algunas partes del sistema.</p> <p>3 = Las ayudas son apropiadas y están distribuidas a lo largo del sistema.</p>	<b>3</b>
<b>Observaciones / Modificaciones Sugeridas</b>		<b>Probador</b>
No hubo mayor cambio, ya que en el ciclo 1 el diseño e interfaz quedo estandarizado.		Susana Aguirre
		<p><b>Firma:</b> SA</p> <p><b>Nombre:</b> Susana Aguirre</p> <p><b>Fecha:</b> 2013-04-25</p>

**Tabla IV – 43:** Prueba de Caso de Uso (Proceso de cotización)

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

#### 4.2.5. POSTMORTEM

Es así que la Fase de Postmortem tendrá como objetivo realizar una evaluación de la calidad del proyecto, se hará un análisis de resultados desde el punto de vista del sistema y cada una de las fases, así como también de la interacción de los desarrolladores entre ellos y entre el cliente.

##### 4.2.5.1. Análisis de Resultados

El análisis de resultados se realizará por cada una de las fases del ciclo.

Fase	Comentario
<b>Análisis de Requerimientos</b>	· Los casos de uso deben ser lo más puntuales posibles, no deben incluir el detalle de cada proceso
	· Se debe analizar y entender desde el principio las entidades involucradas en un proceso.
<b>Diseño del Sistema</b>	· En los diagramas de secuencia es crucial poner todos los controladores que interactuaran en el sistema, pues clarifican el desarrollo del aplicativo.
	· Es fundamental elaborar un diseño bien estructurado que facilite el desarrollo, más cuando se programan procesos que involucran varias clases.
<b>Construcción y pruebas</b>	· El catálogo de piezas de desarrollo orientará las partes de la programación es preciso el orden en las mismas.
	· La aplicación de las pruebas es una de las partes importantes pues nos ayudan a evitar a futuro los errores comunes a la hora de programar.
<b>Implantación</b>	· El cronograma de implantación debe contener los entregables tanto del análisis de software como del acta de entrega/recepción del sistema.
	· El manual técnico debe estar muy bien documentado, pues al ser el sistema de propiedad del cliente, el administrador debe entenderlo fácilmente.

**Tabla IV - 44:** Diagrama Ciclo 2

**Elaborado por:** María Gabriela Bedón, Julio Alberto Lárraga

Una vez analizadas las oportunidades de mejora en cada fase, se desea analizar la interacción con el cliente y entre los desarrolladores. A continuación los siguientes comentarios.

- Una buena comunicación entre miembros del equipo y clientes facilita de gran manera el avance del proyecto.
- La base de un buen sistema es un buen análisis y diseño.

- Se necesita siempre tener un buen diseño que acompañe la programación, puesto que no solamente depende del funcionamiento del sistema sino de lo amigable que sea para el usuario.
- Mantener un estilo de programación, rehusando componentes es primordial para tener un sistema consistente.

## **CAPITULO 5**

### **CONCLUSIONES Y RECOMENCACIONES**

La última fase al presentar en este documento trata sobre las enseñanzas que el proceso de desarrollo del sistema nos ha dado. Por lo mismo, este capítulo comprende las conclusiones y recomendaciones a las que el equipo de trabajo ha llegado.

Se ha cumplido el objetivo principal del proyecto que ha sido el enseñarnos una metodología de trabajo que podamos utilizar a futuro.

#### **5.1 CONCLUSIONES**

- La programación orientada a objetos definitivamente facilita la conversión de los objetos de la vida real con los del software generado, reduciendo tiempos en el desarrollo por sus características como reutilización de código fuente.
- El ciclo de vida en minicascada, permite que entre cada ciclo de vida haya un espacio de retroalimentación o de reflexión para mejorar ciertos problemas, optimizar recursos y tomarlos como futuro aprendizaje en el siguiente ciclo.
- La representación de las necesidades del cliente a través de los diagramas de caso permiten definir en lenguaje del cliente todos los requerimientos, sin embargo los diagramas de casos de uso realizables son los únicos que conectan los requerimientos con la factibilidad del software, pues aquí se analiza hasta donde se llegará con cada caso de uso, para posteriormente diseñar la herramienta.
- Una vez aprobados los diseños del sistema por el usuario, son el respaldo que los desarrolladores cuentan para proceder con la programación, un buen diseño se mantiene en el tiempo, y únicamente se realiza ajustes de forma.
- La implantación de un sistema web debe tener un plan de pruebas que no solo abarque la funcionalidad sino que considere la adaptación y presentación del sistema en los diferentes navegadores. De igual manera, es necesario poseer un ambiente definido y controlado en el internet sobre el cual el equipo de trabajo y el cliente puedan hacer pruebas sin dañar ni presentar información no necesaria en el sitio principal.

- Al poder desarrollar un sistema utilizando herramientas y lenguaje GNU hemos podido encontrar amplio soporte en internet, así como también gran cantidad de aditamentos o plugins que se podrían incorporar en el sitio web, potenciando la funcionalidad del sitio
- La planificación entre los integrantes del proyecto y el cliente es fundamental, pues construye un fuerte cimiento de comunicación desde el levantamiento de información hasta la implantación final del aplicativo

## 5.2. RECOMENDACIONES

- La implementación entregada ha sido una versión beta del sistema el cual puede ser mejorado y cambiado conforme los requerimientos varíen. Todo sistema que trabaje en la web debería ser actualizado y modificado conforme la tecnología y los requerimientos varíen, para mantenerse siempre en el mejor rendimiento posible.
- Para poder empezar con el mantenimiento del sistema es necesario leer la documentación de la presente tesis, principalmente la fase análisis y diseño ya que ahí se podrá encontrar información acerca de los componentes necesarios para la correcta ejecución y que fueron inicialmente aprobados.
- En la implementación del sistema existieron varias ideas que no pudieron ser puestas en práctica en la codificación del sitio, en especial por el tema de límites de tiempo. Estas, sin embargo, pueden ser aplicadas en posteriores revisiones del sitio de ser necesarias y serían mejoras a la presentación y funcionalidad del sitio.
- Los diagramas de caso de uso entregables son cruciales, aun cuando no sufran cambios con los caso de uso, normalmente se hace caso omiso de estos diagramas pues los desarrolladores y diseñadores son las mismas personas y se asume que es lo mismo.
- Se deberá considerar la planificación entre los integrantes del equipo y el cliente como parte imprescindible de una buena práctica de la metodología usada, independientemente de cuál fuera esta.

- La retroalimentación constante entre las partes hará que se optimice el enorme manera los recursos, para cumplir esto la aplicación de minicascada como metodología del ciclo de vida fundamentará y apalancará esta optimización.

## **ANEXOS**

### **MANUAL TECNICO**

#### **SISTEMA DE ADMINISTRACIÓN DE CLIENTES Y PRODUCTOS DE LA EMPRESA LUBRIMOTOR AUTOMOTRIZ.**

### **CONTENIDO**

#### **INTRODUCCION**

1. REQUERIMIENTOS TÉCNICOS
2. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO
3. INSTALACION WEB SERVER
4. CONFIGURACION DEL APLICATIVO
5. CASOS DE USO
6. FLUJOS
7. MODELO ENTIDAD RELACION
8. DICCIONARIO DE DATOS DEL MODELO ENTIDAD RELACIÓN

### **OBJETIVOS**

Brindar la información necesaria para poder realizar la instalación y configuración del aplicativo

Específicos

- Representar la funcionalidad técnica de la estructura, diseño y definición del aplicativo
- Definir claramente el procedimiento de instalación del aplicativo.
- Detallar la especificación de los requerimientos de Hardware y Software necesarios para la instalación de la aplicación.
- Describir las herramientas utilizadas para el diseño y desarrollo del prototip

### **INTRODUCCION**

Este manual describe los pasos necesarios para cualquier persona que tenga ciertas bases de sistemas pueda realizar la instalación del aplicativo creado para la empresa “Lubrimotor Automotriz”.

Es importante tener en cuenta que en el presente manual se hace mención a las especificaciones mínimas de hardware y software para la correcta instalación del aplicativo.

## **1. REQUERIMIENTOS TÉCNICOS.**

### **REQUERIMIENTOS MÍNIMOS DE HARDWARE**

- ✓ Procesador: Core
  
- ✓ Memoria RAM: Mínimo: 1 Gigabytes (GB)
  
- ✓ Disco Duro: 500 Gb.

### **REQUERIMIENTOS MÍNIMOS DE SOFTWARE**

- ✓ Privilegios de administrador
  
- ✓ Sistema Operativo: Windows NT/98/Me/2000/2003/XP/Vista, Sistema Operativo basado en Linux

## **2. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO**

### **2.1. PHP**

Es un Lenguaje de Programación para trabajar páginas WEB ofreciendo la ventaja de mezclarse con HTML. Las ejecuciones son realizadas en el Servidor y el cliente es el encargado de recibir los resultados de la ejecución. Si el cliente realiza una petición, se ejecuta el intérprete de PHP y se genera el contenido de manera dinámica. Permite conexión con varios tipos de Bases de Datos como: MySQL,

Oracle, Postgress, SQL Server, etc. permitiendo aplicaciones robustas sobre la

WEB. Este lenguaje de programación puede ser ejecutado en la gran mayoría de sistemas operacionales y puede interactuar con Servidores WEB populares.

## **2.2 MYSQL**

Es un manejador de Bases de Datos, el cual permite múltiples hilos y múltiples usuarios, fue desarrollado como software libre.

Aunque se puede usar sobre varias plataformas es muy utilizado sobre LINUX. Es libre para uso en Servidores WEB.

Ofrece ventajas tales como fácil adaptación a diferentes entornos de desarrollo, Interacción con Lenguajes de Programación como PHP, Java Script y fácil Integración con distintos sistemas operativos.

## **2.3 APACHE**

Es un Servidor WEB desarrollado por el grupo Apache. Su código fuente se puede distribuir y utilizar de forma libre.

Está disponible para diferentes plataformas de Sistemas Operativos entre otros Windows, Linux, Mac y NetWare

.

Ofrece ventajas tales como independencia de plataforma, haciendo posible el cambio de plataforma en cualquier momento; creación de contenidos dinámicos, permitiendo crear sitios mediante lenguajes PHP.

Además de ser libre su soporte técnico es accesible ya que existe una comunidad que está disponible en foros, canales IRC y servidores de noticias, donde hay gran cantidad de usuarios disponibles para cuando surge algún problema.

## **3. INSTALACION DE WEB SERVER**

Descargar al PC el fichero de instalación wampserver2.2e-php5.4.3-httpd2.2.22-mysql5.5.24-32b.exe el cuál se encuentra en la página: <http://sourceforge.net/Projects/wampserver/> una vez descargado, dar clic sobre él para iniciar la instalación. La instalación se debe hacer con una cuenta de Administrador o con derechos de administrador.

Durante la instalación aparecerán las siguientes pantallas:

La primera ventana que aparecerá es la de bienvenida al Setup Wizard de la instalación, pulsa el botón Instalación y administración WampServer 2.2a 64 bits para continuar

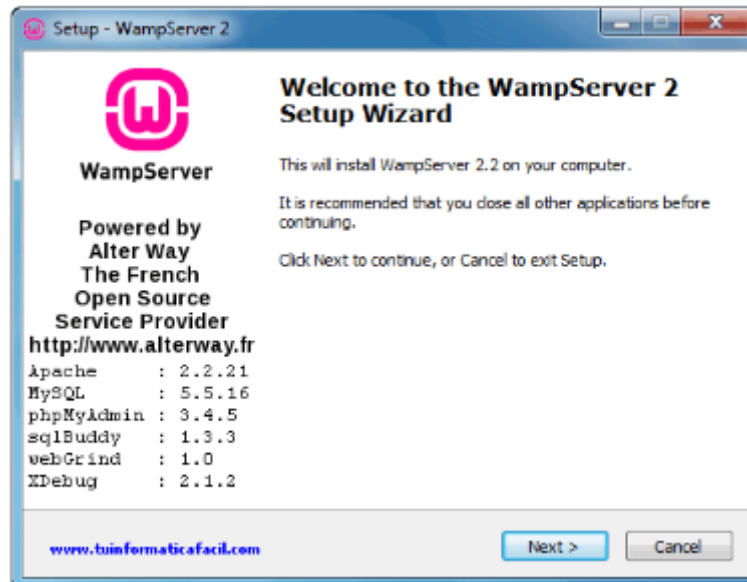


Figura No 1 - WapServer Instalación 1

La siguiente ventana muestra el acuerdo de licencia, selecciona –I accept the agreement - y botón “Next”. Instalación y administración WampServer 2.2a 64 bits para continuar.

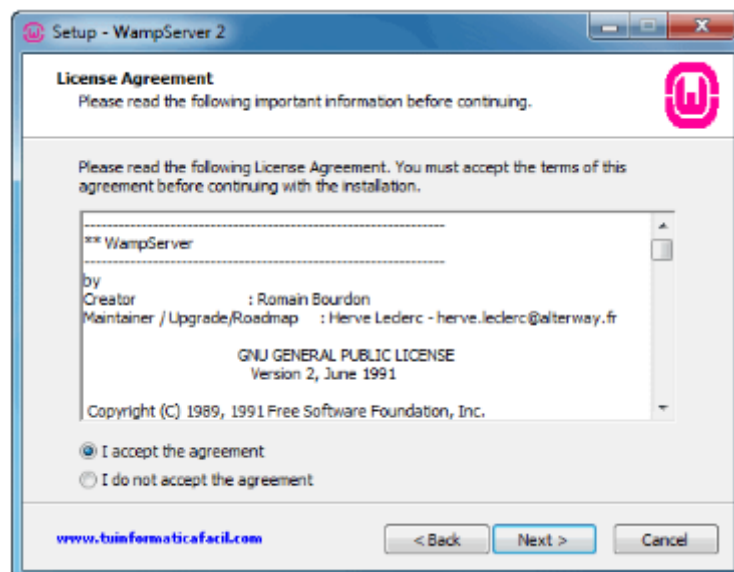
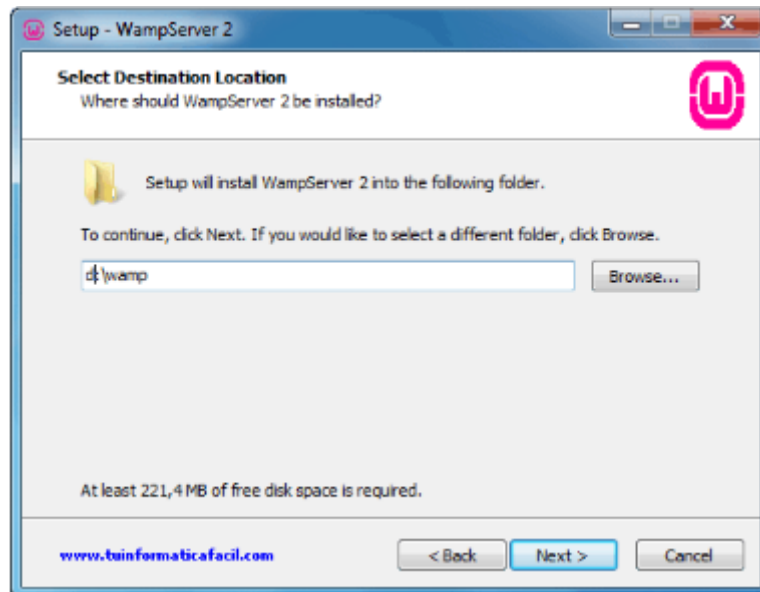


Figura No 2 - WapServer Instalación 2

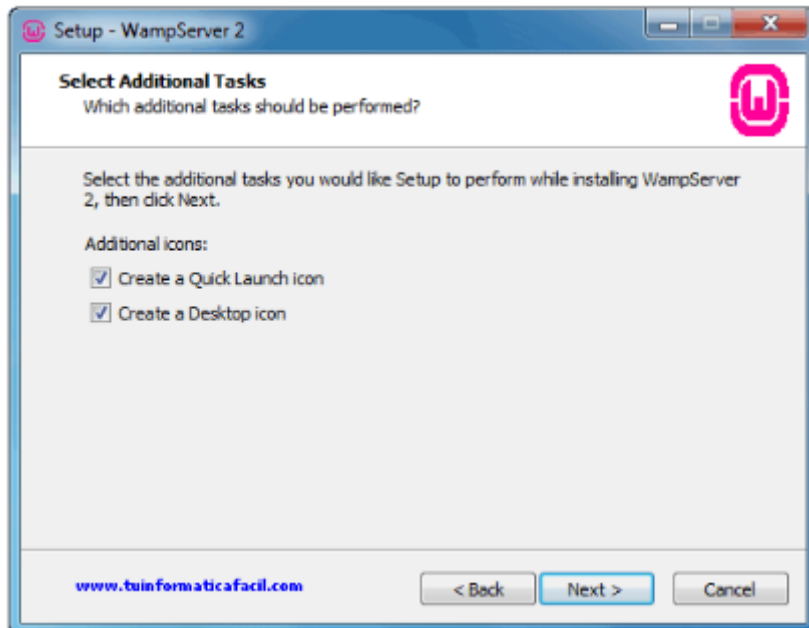
En esta ventana tienes que especificar el directorio donde se va a realizar la instalación, el Setup muestra una directorio por defecto - c:\wamp -, donde realizará la instalación, si quieres instalar en otra localización, yo he cambiado a la unidad d:, pulsa el botón

Instalación y administración WampServer 2.2 64 bits, una vez tengas el directorio destino decidido pulsa el botón Instalación y administración WampServer 2.2 64 bits “Next” para continuar



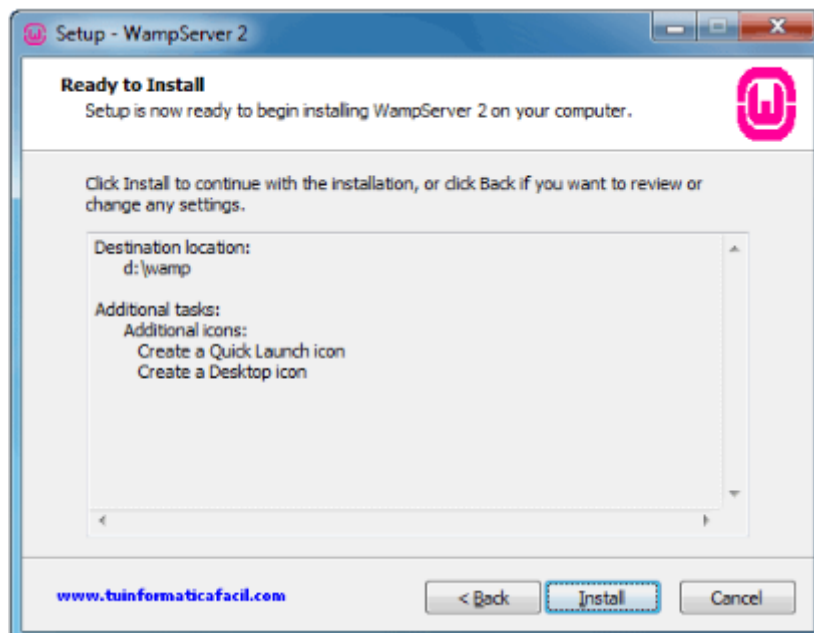
**Figura No 3-WapServer Instalación 3**

En esta ventana selecciona los iconos que instalador creara automáticamente, icono en el inicio rápido o/y icono en el escritorio, pulsa el botón Instalación y administración WampServer 2.2a 64 bits “Next” para continuar.



**Figura No 4-WapServer Instalación 4**

Finalmente obtendrás una ventana con el resumen de las tareas que va a realizar el Setup de WampServer 2.2a 64 bits, pulsa el botón Instalación y administración de WampServer 2.2a 64 bits “Install” y la instalación comenzará



**Figura No 5--WapServer Instalación 5**

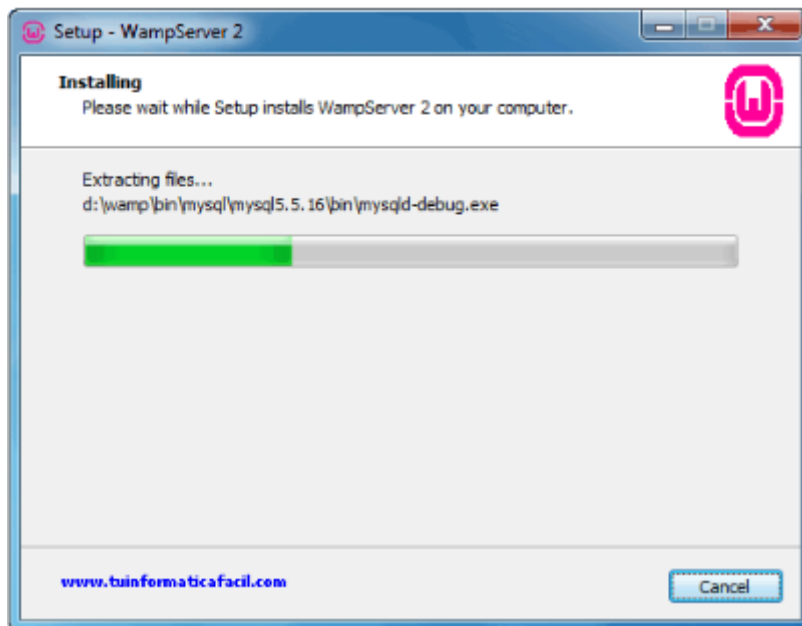


Figura No 6-WapServer Instalación 6

Durante el proceso de instalación aparece el explorador de archivos para que selecciones el navegador que vas a utilizar por defecto cada vez que algunas de las herramientas incluidas en WampServer, como por ejemplo PhpMyAdmin, por defecto WampServer utilizará el navegador Internet Explorer de Microsoft, si deseas utilizar otro debes buscar el ejecutable del navegador que quieras usar y pulsa el botón – Abrir -para seleccionarlo.

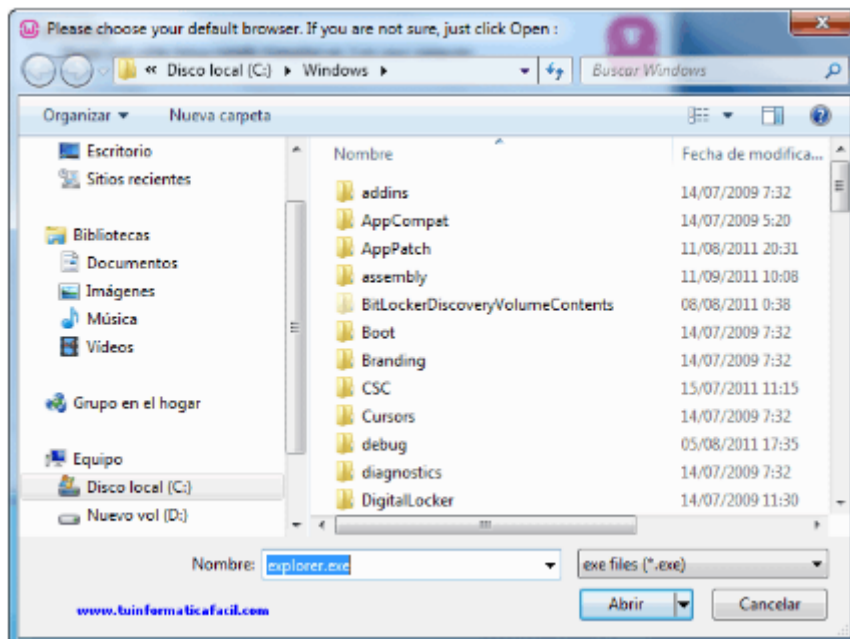
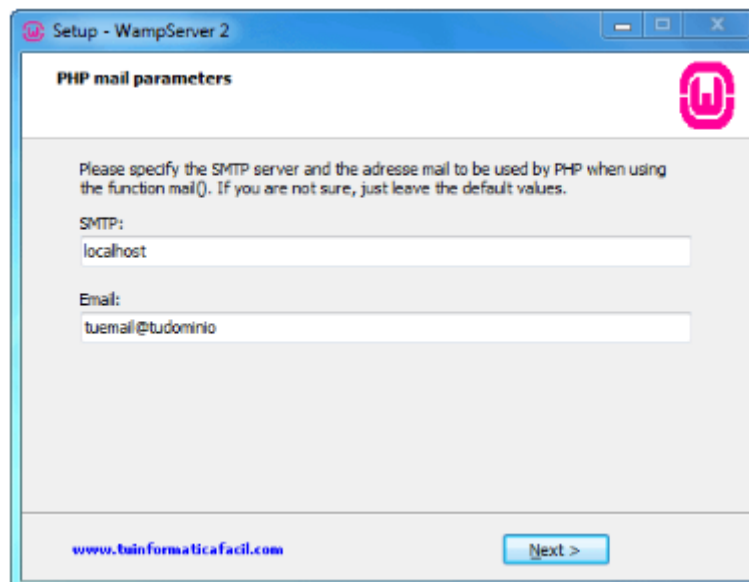


Figura No 7 - WapServer Instalación 7

Si en el PC o servidor donde estas instalando tienes montado un servidor de correo (servidor SMTP) introduce el nombre del servidor de correo en campo SMTP y una cuenta de correo valida en el campo Email, esto será usado por la función mail() de PHP como remitente para correos de salida. Si no sabes o no estás seguro de tener un servidor de correo instalado, deja los valores que aparecen por defecto.

Pulsa el botón Instalación y administración WampServer 2.2a 64 bits “Next” para continuar.



**Figura No 8 - WapServer Instalación 8**

La instalación finalizo.

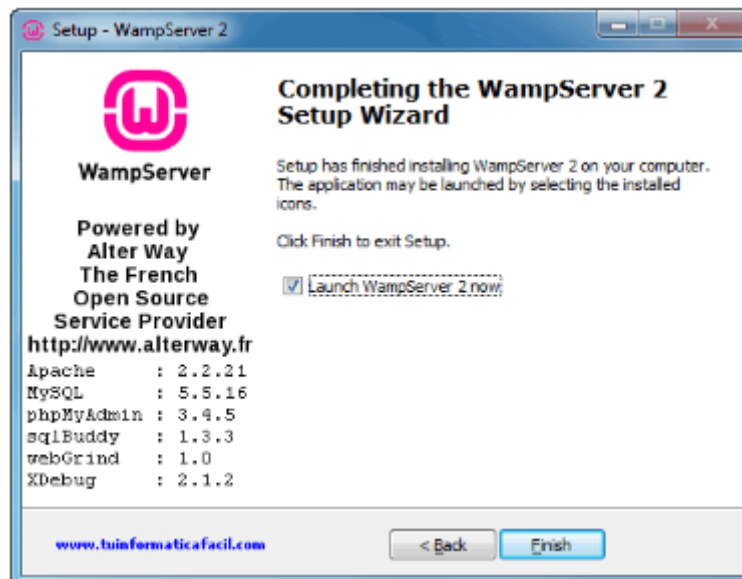


Figura No 9 - WapServer Instalación 9

### 3. CONFIGURACIÓN DEL APLICATIVO

Se coloca la carpeta llamada sistema en la ruta donde se encuentra instalado el aplicativo normalmente es: C:\wamp\www o en este caso en el dominio de la empresa “lubrimotorautomotriz.com”.

Luego se buscan los archivos que mencionan a continuación y se configuran

#### 4.1 Reemplazar archivo controller

aplication\config

config.php

\$config['base\_url'] coloca como aparece en internet

http://www.lubrimotorautomotriz.com/sistema

Guardar

#### 4.2 Configurar la base de datos

aplication\database.php

\$db['default']['hostname'] = '98.130.0.108';

\$db['default']['username'] = 'C365242\_lubauto';

```
$db['default']['password'] = 'lubaUTo2013';  
$db['default']['database'] = 'C365242_lubauto_sys';  
$db['default']['dbdriver'] = 'mysql';
```

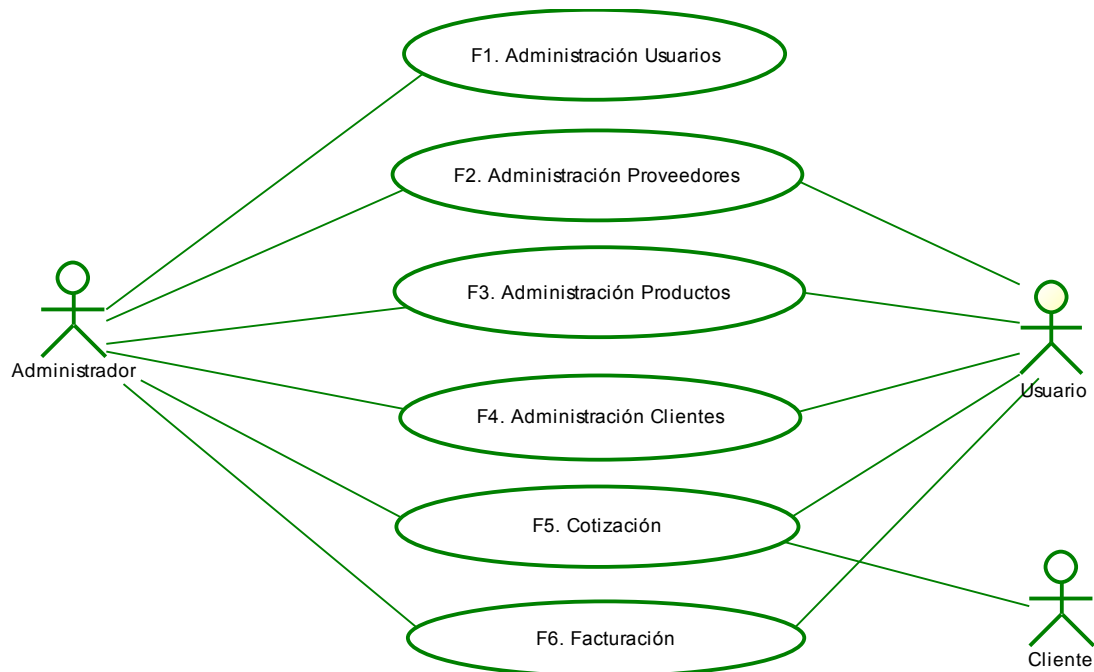
## 5. CASOS DE USO

Para el aplicativo se va a definir dos roles o tipos de usuarios que interactuarán con el sistema de información de diferentes modos.

**Administrador:** tendrá la labor de ingresar, eliminar, modificar, copiar y realizar diferentes acciones sobre los registros de las tablas de la base de datos.

**Usuario:** tendrá similares permisos al del administrador, a excepción del manejo de usuarios.

**Cliente:** Cualquier persona que entre a la página web de la empresa y realice una cotización de productos.



**Figura IV - 1:** Diagrama General Casos de Uso

## 6. FLUJOS

**Caso de Uso:** Administración de Usuarios

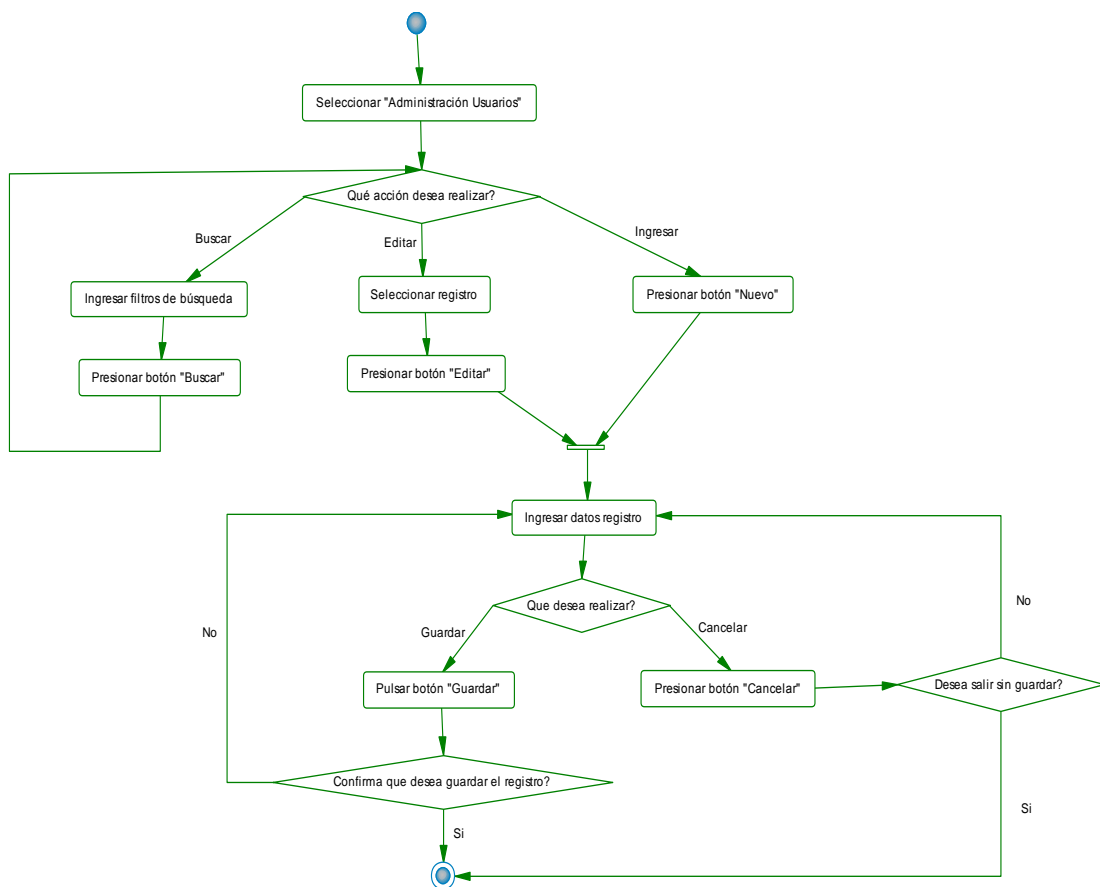
**Identificador:** F1

**Actores:** Administrador

**Pre-condición:**

Nuevas contrataciones de personal en la empresa implican crear nuevos usuarios.  
Se necesitan actualizar los datos de un usuario.

**Flujo:**



**Pos-condición:**

La base queda actualizada  
Los nuevos usuarios pueden hacer uso del sistema.

**Tabla IV – 1:** Tabla de Flujo para la Administración de Usuarios

**Caso de Uso:** Administración de Proveedores

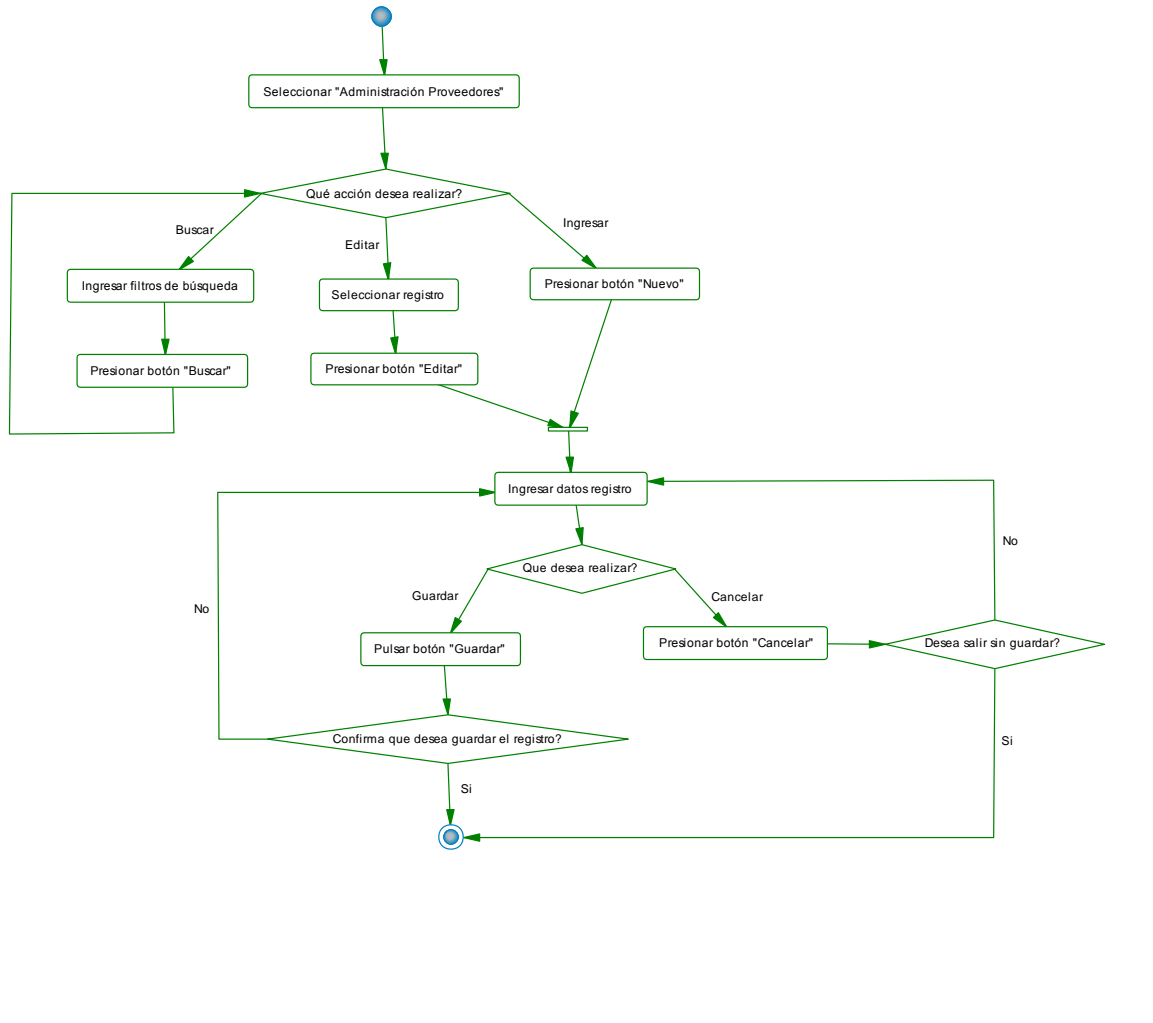
**Identificador:** F2

**Actores:** Administrador/ Usuario

**Pre-condición:**

La empresa necesita adquirir los productos de una nueva marca, y esto implica registrar un nuevo proveedor.

**Flujo:**



**Pos-condición:**

La base queda actualizada  
Se podrán ingresar los productos del nuevo proveedor.

**Tabla IV – 2:** Tabla de Flujo para la Administración de Proveedores

**Caso de Uso:** Administración de Productos

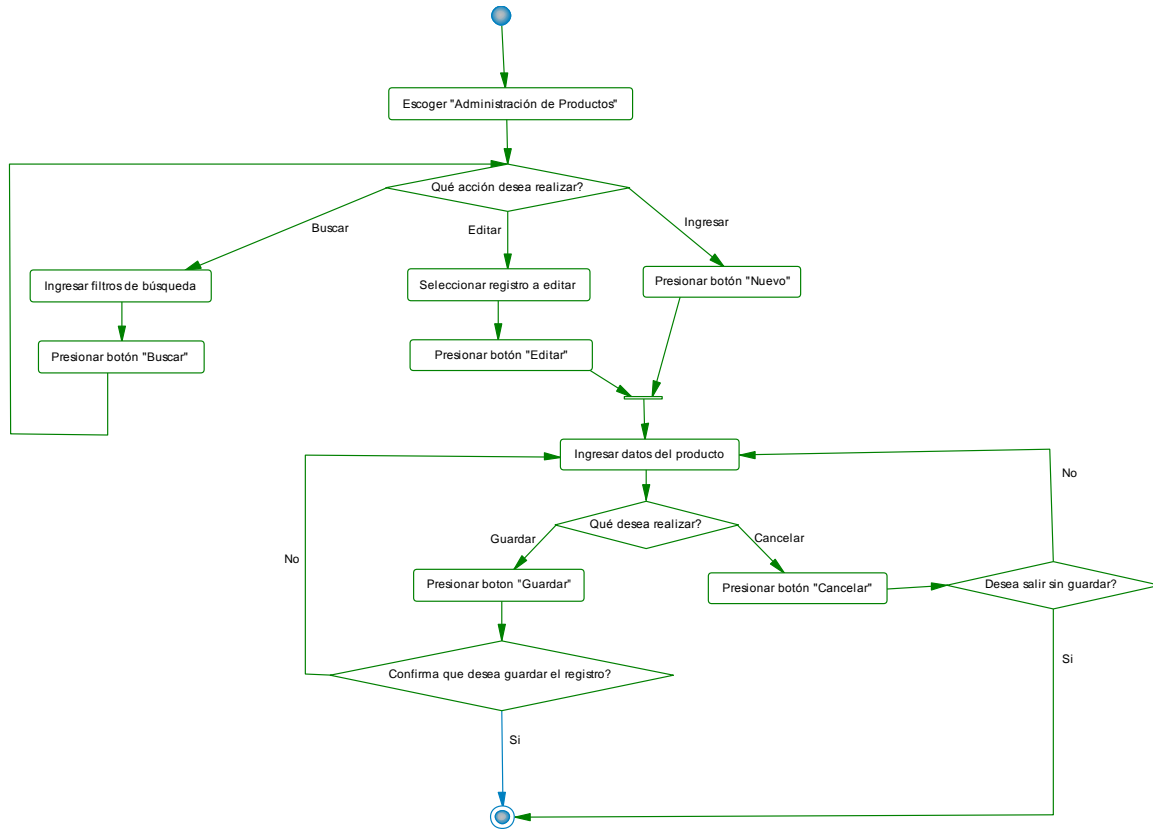
**Identificador:** F3

**Actores:** Administrador/ Usuario

**Pre-condición:**

El negocio adquiere nuevos productos, y el administrador debe registrarlos.  
 Hay cambios en el precio, descripción o detalle de los productos.  
 Hay cambios en la disponibilidad de los productos

**Flujo:**



**Pos-condición:**

La base queda actualizada  
 Los productos quedan actualizados y disponibles para ser usados en cotizaciones o en el proceso de facturación.

**Tabla IV – 3:** Tabla de Flujo para la Administración de Productos

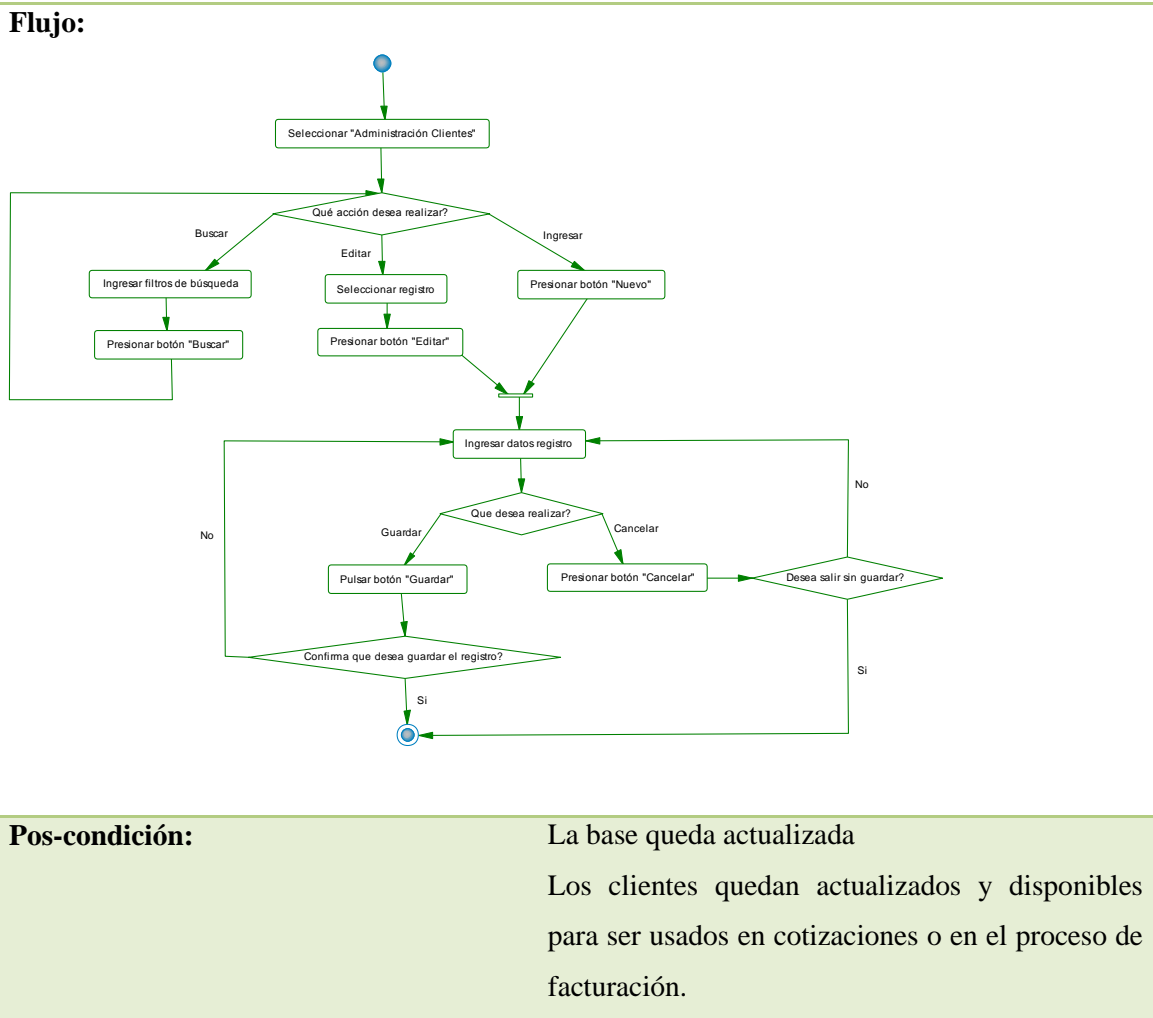
**Caso de Uso:** Administración de Clientes

**Identificador:** F4

**Actores:** Administrador/Usuario

**Pre-condición:**

Para los procesos de facturación y cotización la empresa necesita registrar el cliente, con el propósito de mantener un historial y una cartera de clientes.



**Tabla IV – 4:** Tabla de Flujo para la Administración de Clientes

**Caso de Uso:** Cotización

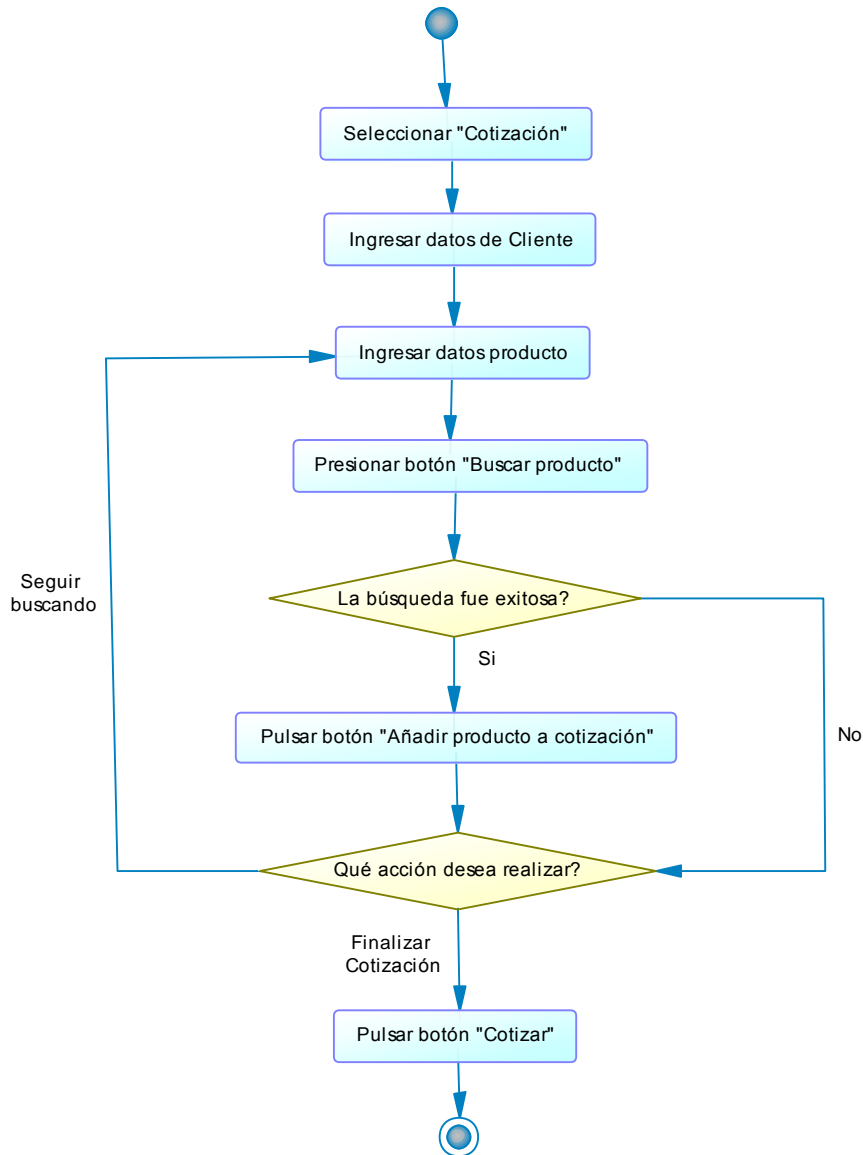
**Identificador:** F5

**Actores:** Administrador / Usuario / Cliente

**Pre-condición:**

Los clientes desean averiguar los precios de los productos antes de adquirirlos.

**Flujo:**



**Pos-condición:** Se muestra la información requerida

**Tabla IV – 1:** Tabla de Flujo para Cotización

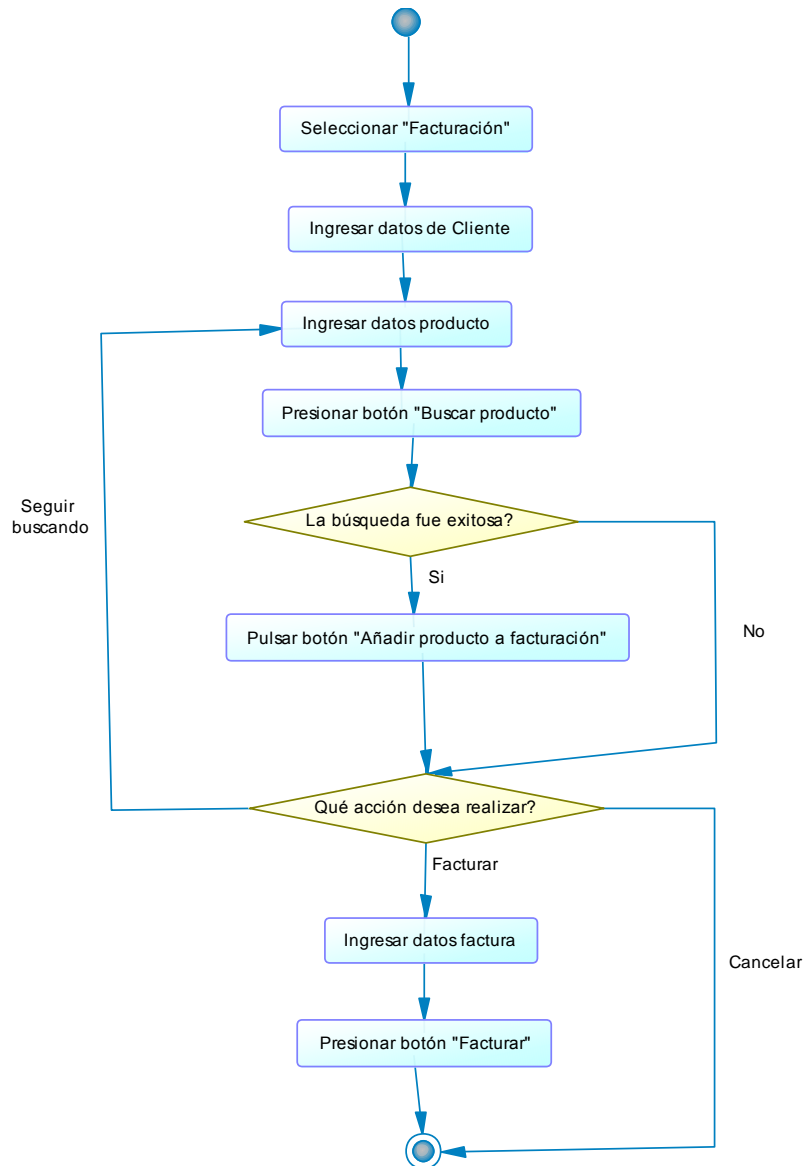
**Caso de Uso:** Facturación

**Identificador:** F6

**Actores:** Administrador / Usuario

**Pre-condición:**  
Cliente adquiera producto de la empresa.

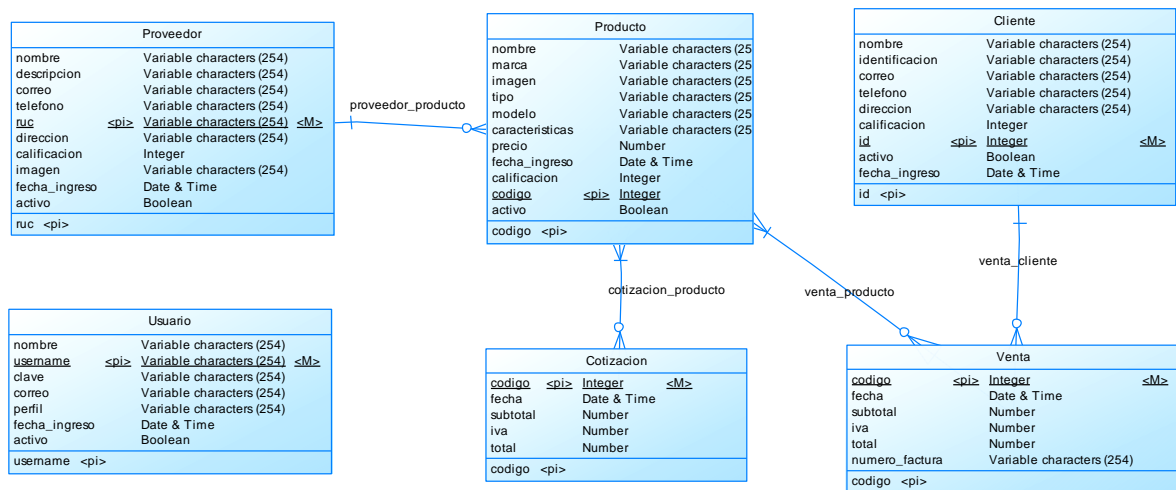
**Flujo:**



**Pos-condición:** La base queda actualizada  
Se genera un documento físico que es la factura.

**Tabla IV – 2:** Tabla de Flujo para la Facturación

**7. MODELO ENTIDAD RELACIÓN**



## 8. DICCIONARIO DE DATOS DEL MODELO ENTIDAD RELACIÓN

Entidades	Atributos	Tipo de Dato	Descripción
<b>Usuario</b>	username	varchar(50)	<b>Clave</b> única de registro de usuarios
	nombre	varchar(254)	Nombre del Usuario
	clave	varchar(254)	Contraseña para ingreso al sistema
	correo	varchar(254)	Correo electrónico del usuario
	activo	tinyint(4)	Estado del Usuario
	fecha_ingreso	datetime	Fecha en la que el Usuario fue creado
	perfil	varchar(50)	Nombre del perfil del usuario
<b>Producto</b>	codigo	int	<b>Clave</b> única de registro de productos
	nombre	varchar(254)	Nombre del Producto
	marca	varchar(254)	Marca del Producto
	imagen	varchar(254)	Imagen del Producto
	tipo	varchar(254)	Tipo de Producto
	modelo	varchar(254)	Modelo del Producto
	caracteristicas	varchar(254)	Características que tiene el Producto
	precio	numeric(8,0)	Precio del Producto
	fecha_ingreso	datetime	Fecha de Creación del Producto
	calificacion	varchar(254)	La calificación del Producto

	activo	tinyint(1)	Estado del Producto
	proveedor_ruc	varchar(13)	Clave foránea de la tabla Proveedor
<b>Cliente</b>	id	int	<b>Clave</b> única de registro de clientes
	nombre	varchar(254)	Nombre del Cliente
	identificacion	varchar(254)	Cedula o pasaporte del Cliente
	correo	varchar(254)	Correo electrónico del Cliente
	telefono	varchar(254)	Número de teléfono del Cliente
	direccion	varchar(254)	Domicilio del Cliente
	calificacion	int	La calificación del Cliente
	activo	tinyint(1)	Estado del Cliente
	fecha_ingreso	datetime	Fecha de creación del Cliente
<b>Proveedor</b>	ruc	varchar(13)	<b>Clave</b> única de registro de proveedores
	nombre	varchar(254)	Nombre del Proveedor
	descripcion	varchar(254)	Descripción del Proveedor
	correo	varchar(254)	Correo electrónico del Proveedor
	telefono	varchar(254)	Número de teléfono del Proveedor
	direccion	varchar(254)	Dirección del Proveedor
	calificacion	int	Calificación del Proveedor
	imagen	varchar(254)	Imagen del Proveedor
	fecha_ingreso	datetime	Fecha de creación del Proveedor
	activo	tinyint(1)	Estado del Proveedor
<b>Venta</b>	codigo	int(11)	<b>Clave</b> única de registro de facturas
	cliente_id	int(11)	Clave foránea
	fecha	datetime	Fecha que se realiza la factura
	subtotal	numeric(8,0)	Suma de todos los productos adquiridos en la factura
	iva	numeric(8,0)	Porcentaje de IVA
	total	numeric(8,0)	Suma de subtotal más IVA
<b>Cotización</b>	codigo	int	<b>Clave</b> única de registro de Cotizaciones
	fecha	datetime	Fecha que se realiza la Cotización
	subtotal	numeric(8,0)	Suma de todos los productos seleccionados en la

Cotización			
	iva	numeric(8,0)	Porcentaje de IVA
	total	numeric(8,0)	Suma de subtotal más IVA
<b>cotizacion_producto</b>	codigo	int(11)	<b>Clave</b> primaria. Clave foránea de la tabla cotización.
	Pro_codigo	int(11)	<b>Clave</b> primaria Clave foránea de la tabla producto.
<b>venta_producto</b>	codigo	int(11)	<b>Clave</b> primaria. Clave foránea de la tabla venta.
	Pro_codigo	int(11)	<b>Clave</b> primaria Clave foránea de la tabla producto.

## **MANUAL DE USUARIO**

### **Introducción**

En el presente manual tiene el propósito de guiar a los usuarios para el uso correcto del sistema, tanto como administradores como usuarios registrados. Presentando el proceso a seguir para las diferentes funcionalidades que esta posee, por ejemplo el registro de los usuarios, la búsqueda de productos, administración de productos, etc.

En esta sección del manual presentamos el funcionamiento los componentes de que la mayoría de usuarios visualiza a tras vez de su explorador.

### **Proceso de Usuarios**

Esta es una de las funcionalidades que debería ser de las más utilizadas dentro del sitio. Para un correcto uso se recomienda seguir las recomendaciones presentadas a continuación.

### **Ingreso al Sistema**

A través de la página de inicio visualizaremos esta ventana de login la cual permite ingresar el usuario y contraseña (password) que hemos obtenido luego de haber registrado en el sistema. También tenemos la posibilidad de registrarnos o de definir un nuevo password en caso de olvidarlo. Estas acciones se pueden manejar desde el módulo ubicado en la mayoría de las páginas, y que se muestra en la figura I – 1.



Usuario

Clave

© Copyright 2013 Lubrimotor Automotriz. Todos los Derechos Reservados. xhtml - css

**Figura I - 1:** Componente de login  
**Elaborado por:** María Gabriela Bedón, Alberto Lárraga Mena

### **Registro de usuarios nuevos**

Si deseamos registrarnos al sistema debemos comunicarnos con el Administrador, pues es la única persona que puede crear nuevos usuarios en el sistema

### **Perfil**

Si deseamos ver nuestra información ingresada podemos acceder al enlace de Perfil el cual nos despliega una vista similar a la que podemos observar en la figura I - 02.



Nidia Guayaquil

- Archivo
  - Administración >
  - Procesos >
  - Salir
- Proveedores
  - Productos
  - Clientes

### Productos

Mostrar 10 registros por página  Buscar

Nombre	Proveedor	Marca	Estado
Aceite	Clicka Imp.	Havoline	Activo
cortadora triple hoja	Clicka Imp.	Steehl	Activo
cortadora triple hoja	Clicka Imp.	Kawasaki	Activo

Mostrando 1 a 3 de 3 registros

**Figura I - 2:** Ventana de Perfil Front-End  
**Elaborado por:** María Gabriela Bedón, Alberto Lárraga Mena

### Búsqueda de Proveedores

El sitio nos presenta la factibilidad de buscar un proveedor, búsqueda que se puede realizar en base al nombre, área tecnológica o dirección del proveedor. De igual manera se ha provisto de la funcionalidad para buscar un proveedor en base a la inicial de su nombre. Además poseemos un botón para la búsqueda de todos los proveedores en el sitio. La sección mostrada en la figura Anexo IV - 9 muestra un ejemplo de la ventana de búsqueda.



Nidia Guayaquil

- Archivo
- Administración >
- Procesos >
- Salir

### Proveedores

Mostrar 10 registros por página  Buscar

RUC	Nombre	Correo	Estado
1722651773001	Clicka Imp.	contact@clicka.ec	Activo
1722651774001	Stihl	info@stihl.com	Activo

Mostrando 1 a 2 de 2 registros

© Copyright 2013 Lubrimotor Automotriz. Todos los Derechos Reservados. xhtml - css

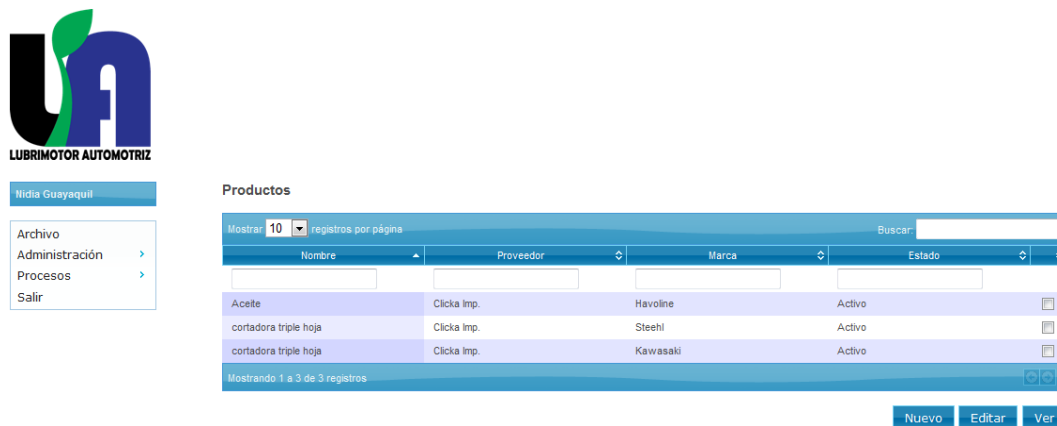
**Figura Anexo I - 3:** Ventana de Búsqueda de Proveedores Front-End

**Elaborado por:** María Gabriela Bedón, Alberto Lárraga Mena

Una vez finalizada la búsqueda se presentarán los datos en una lista, para que el usuario pueda escoger el proveedor del que desea visualizar la información.

### Búsqueda de Productos

Para poder solventar las difíciles búsquedas de los productos que buscamos el sistema nos permite filtrar de diferentes formas nuestra información. Por lo tanto, la ventana de búsqueda, así como los resultados que se presentarán en esta se muestran en la figura



**Figura I - 4:** Ventana búsqueda Producto  
**Elaborado por:** María Gabriela Bedón, Alberto Lárraga Mena

### Búsqueda de Clientes

La búsqueda de clientes es el mismo proceso que el de Productos y proveedores, se debe realizar la consulta para encontrarlo, caso contrario se debe crear uno nuevo o editar su información en caso de ser necesario.



Cientes

Mostrar 10 registros por página

Nombre	Identificación	Correo	Estado
Belén Lárraga	1722641763	belen.larraga@hotmail.com	Activo
Julio Alberto Lárraga Mena	1722651773	albert@clicka.ec	Activo

Mostrando 1 a 2 de 2 registros

Nuevo Editar Ver

© Copyright 2013 Lubrimotor Automotriz. Todos los Derechos Reservados. xhtml - css

**Figura I - 5:** Ventana búsqueda Clientes  
**Elaborado por:** María Gabriela Bedón, Alberto Lárraga Mena

## Proceso de Ventas

El Proceso de Ventas inicia con la disponibilidad de productos y creación de los clientes, con esta pre-condición, se debe crear una Venta en la siguiente figura I -7.



Ventas

Mostrar 10 registros por página

Código	Fecha	Cliente
No hay datos		

Mostrando 0 a 0 de 0 registros

Nuevo Ver

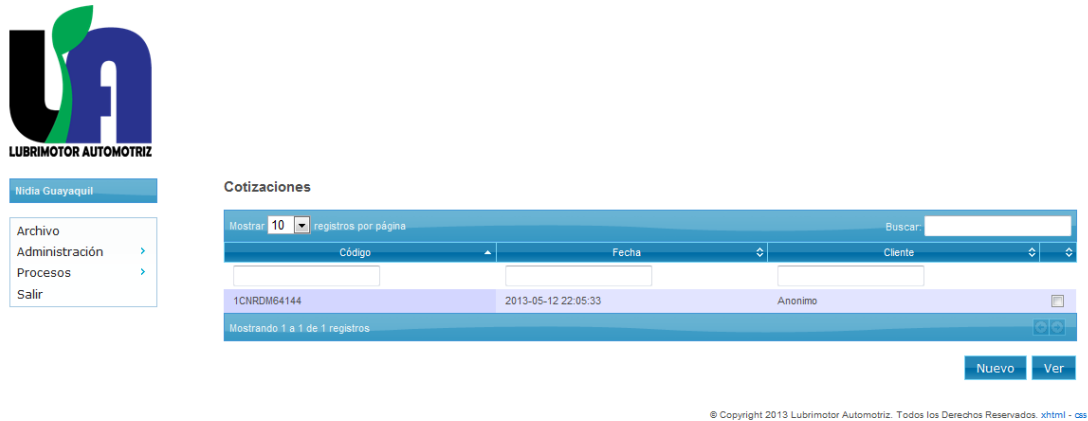
© Copyright 2013 Lubrimotor Automotriz. Todos los Derechos Reservados. xhtml - css

**Figura I - 7:** Ventana de Cotización de un Producto  
**Elaborado por:** María Gabriela Bedón, Alberto Lárraga Mena

## Cotizaciones

Una vez identificado el producto y el proveedor tenemos la posibilidad de solicitar una cotización la cual será evaluada por la empresa y los proveedores. La pantalla para el envío del

correo electrónico al proveedor, y que creará el registro de solicitud de cotización se muestra en la figura I – 6.



**Figura I - 6: Ventana de Cotización de un Producto**  
**Elaborado por:** María Gabriela Bedón, Alberto Lárraga Mena

Al decidir solicitar la cotización, un correo electrónico se enviará automáticamente, y se presentará una ventana de éxito o error al proceso realizado.

## **IFORME DE RECURSOS**

### **Informe de Análisis de Recursos de Hardware y Software**

Este informe se hace con la finalidad de documentar el análisis de los recursos que son necesarios para que el sistema de Administración de Usuarios, Clientes, Producto, Proveedores y de los procesos de Facturación y Cotización de la empresa Lubrimotor Automotriz funcione correctamente. Para los usuarios los requerimientos son mínimos, de manera que debe poseer un ordenador funcional y un browser, al igual que posibilidad para navegar en el internet. Por lo mismo, los requerimientos de software indicados son:

#### **REQUERIMIENTOS MÍNIMOS DE HARDWARE**

- ✓ Procesador: Core
- ✓ Memoria RAM: Mínimo: 1 Gigabytes (GB)
- ✓ Disco Duro: 500 Gb.

#### **REQUERIMIENTOS MÍNIMOS DE SOFTWARE**

- ✓ Privilegios de administrador
- ✓ Sistema Operativo: Windows NT/98/Me/2000/2003/XP/Vista, Sistema Operativo basado en Linux

#### **Entregado por:**

  
María Gabriela Bedón Melo

  
Julio Alberto Larraga Mena

Quito, 03 Abril del 2013

## ACTA ENTREGA RECEPCIÓN

### Acta de Entrega Recepción de Software

En la ciudad de Ibarra, siendo las 9h00 del día Viernes 10 de Mayo del 2013, en las oficinas de Lubrimotor Automotriz, ubicado en la Av. Mariano Acosta y Sánchez y Cifuentes; se levanta el acta de entrega-recepción del sistema Lubrimotor Automotriz, que administra clientes, proveedores, usuarios, productos y que tiene como procesos automatizados el de Facturación y Cotización.

Para tal efecto se reunieron las partes, el dueño de la empresa Germán Bedón Aguirre, y los desarrolladores María Gabriela Bedón y Alberto Larraga Mena.

La entrega formal del software contiene la culminación del entrenamiento a los usuarios así como también de los respectivos manuales técnicos y para usuario final. Se entrega el código fuente desarrollado con el respectivo índice de programación para futuros cambios, en caso de querer modificar el alcance del producto.

#### Entregado por:

  
María Gabriela Bedón Melo

  
Julio Alberto Larraga Mena

#### Recibido por:

  
Germán Bedón Aguirre

## INDICE DE PROGRAMACIÓN

Nombre	Tipo
<a href="http://www.lubrimotorautomotriz.com/sistema/">www.lubrimotorautomotriz.com/sistema/</a>	directorio
_MACOSX	directorio
application	directorio
cache	directorio
config	directorio
autoload.php	archivo php
config.php	archivo php
database.php	archivo php
routes.php	archivo php
controllers	directorio
home.php	archivo php
usuario.php	archivo php
usuarioList.php	archivo php
proveedor.php	archivo php
proveedorList.php	archivo php
producto.php	archivo php
productoList.php	archivo php
cliente.php	archivo php
clienteList.php	archivo php
cotizacion.php	archivo php
cotizacionList.php	archivo php
facturacion.php	archivo php
facturacionList.php	archivo php
core	directorio
errors	directorio
helpers	directorio
hooks	directorio
language	directorio
libraries	directorio
logs	directorio
models	directorio
usuario_model.php	archivo php
proveedor_model.php	archivo php
producto_model.php	archivo php
cliente_model.php	archivo php
cotizacion_model.php	archivo php
facturacion_model.php	archivo php
third_party	directorio
views	directorio
cotizacion	directorio
cotizacionConfirmation_view.php	archivo php
cotizacionForm_view.php	archivo php
cotizacionList_view.php	archivo php
cotizacionPrint_view.php	archivo php
productosCotizacion_view.php	archivo php
facturacion	directorio

	facturacionConfirmation_view.php	archivo php
	facturacionForm_view.php	archivo php
	facturacionList_view.php	archivo php
	facturacionPrint_view.php	archivo php
	productosFacturacion_view.php	archivo php
	index_view.php	archivo php
	loginView.php	archivo php
	principal_view.php	archivo php
	usuarioList_view.php	archivo php
	usuarioForm_view.php	archivo php
	proveedorList_view.php	archivo php
	proveedorForm_view.php	archivo php
	productoList_view.php	archivo php
	productoForm_view.php	archivo php
	clienteList_view.php	archivo php
	clienteForm_view.php	archivo php
	index.html	archivo html
public		directorio
	css	directorio
	estilos_admin.css	archivo css
	global.css	archivo css
	menu.css	archivo css
	ui.notify	archivo css
	images	directorio
	logo.png	imagen
	productos	directorio
	proveedores	directorio
	js	directorio
	DataTables-1.9.1	directorio
	jquery-ui-1.10.0.custom	directorio
	noty-master	directorio
	jquery.notify.js	archivo js
system		directorio
templates		directorio
codegen.php		archivo php
index.php		archivo php
license.doc		archivo doc