



INSTITUTO TECNOLÓGICO  
"CORDILLERA"

**CARRERA DE ANÁLISIS DE SISTEMAS**

CONTROL DE PEDIDOS E INVENTARIO DE PRODUCTOS MEDIANTE UNA  
APLICACIÓN WEB PARA LA EMPRESA DE MATERIAL ELÉCTRICO  
"IMPORT"

Autor: Cuichán Lincango Iván Mauricio

Tutor: Ingeniero Richard Mafla

Quito, Abril 2014

## DECLARACIÓN DE APROBACIÓN TUTOR

En mi calidad de tutor del trabajo sobre el tema: **CONTROL DE PEDIDOS E INVENTARIO DE PRODUCTOS MEDIANTE UNA APLICACIÓN WEB PARA LA EMPRESA DE MATERIAL ELÉCTRICO "IMPORT"**, presentado por el ciudadano: **CUICHÁN LINCANGO IVÁN MAURICIO**, estudiante de la Escuela de Análisis y Sistemas, considero que dicho informe reúne los requisitos y méritos suficientes para ser sometido a la evaluación por parte del Tribunal de Grado, que el Honorable Consejo de Escuela designe, para su correspondiente estudio y calificación.

Quito, Abril del 2014

---

Tutor.

Ing. Richard Mafla

---

Lector.

Ing. Hugo Heredia



---

## DECLARATORIA

El abajo firmante, declara que los contenidos y los resultados obtenidos en el presente proyecto, como requerimiento previo para la obtención del Título de Tecnólogo en Análisis y Sistemas, son absolutamente originales, auténticos y personales y de exclusiva responsabilidad legal y académica de los autores.

---

Cuichán Lincango Iván Mauricio

C.C. 171754799-4



## CONTRATO DE CESIÓN SOBRE DERECHOS PROPIEDAD INTELECTUAL

Comparecen a la celebración del presente contrato de cesión y transferencia de derechos de propiedad intelectual, por una parte, el estudiante **CUICHÁN LINCANGO IVAN MAURICIO**, por sus propios y personales derechos, a quien en lo posterior se le denominará el "CEDENTE"; y, por otra parte, el INSTITUTO SUPERIOR TECNOLÓGICO CORDILLERA, representado por su Rector el Ingeniero Ernesto Flores Córdova, a quien en lo posterior se lo denominará el "CESIONARIO". Los comparecientes son mayores de edad, domiciliados en esta ciudad de Quito Distrito Metropolitano, hábiles y capaces para contraer derechos y obligaciones, quienes acuerdan al tenor de las siguientes cláusulas:

**PRIMERA: ANTECEDENTE.-** a) El Cedente dentro del pensum de estudio en la carrera de análisis de sistemas que imparte el Instituto Superior Tecnológico Cordillera, y con el objeto de obtener el título de Tecnólogo en Análisis de Sistemas, el estudiante participa en el proyecto de grado denominado "**CONTROL DE PEDIDOS E INVENTARIO DE PRODUCTOS MEDIANTE UNA APLICACIÓN WEB PARA LA EMPRESA DE MATERIAL ELÉCTRICO IMPORT**", el cual incluye la creación y desarrollo del programa de ordenador o software, para lo cual ha implementado los conocimientos adquiridos en su calidad de alumno. b) Por iniciativa y responsabilidad del Instituto Superior Tecnológico Cordillera se desarrolla la creación del programa de ordenador, motivo por el cual se regula de forma clara la cesión de los derechos de autor que genera la obra literaria y que es producto del proyecto de grado, el mismo que culminado es de plena aplicación técnica, administrativa y de reproducción.

**SEGUNDA: CESIÓN Y TRANSFERENCIA.-** Con el antecedente indicado, el Cedente libre y voluntariamente cede y transfiere de manera perpetua y gratuita todos los derechos patrimoniales del programa de ordenador descrito en la cláusula anterior a favor del Cesionario, sin reservarse para sí ningún privilegio especial (código fuente, código objeto, diagramas de flujo, planos, manuales de uso, etc.). El Cesionario podrá explotar el programa de ordenador por cualquier medio o procedimiento tal cual lo establece el Artículo 20 de la Ley de Propiedad Intelectual, esto es, realizar, autorizar o prohibir, entre otros: a) La reproducción del programa de ordenador por cualquier forma o procedimiento; b) La comunicación pública del software; c) La distribución pública de ejemplares o copias, la comercialización, arrendamiento o alquiler del programa de ordenador; d) Cualquier transformación o modificación del programa de ordenador; e) La protección y registro en el IEPI el programa de ordenador a nombre del Cesionario; f) Ejercer la protección jurídica del programa de ordenador; g) Los demás derechos establecidos en la Ley de Propiedad Intelectual y otros cuerpos legales que normen sobre la cesión de derechos de autor y derechos patrimoniales.

**TERCERA: OBLIGACIÓN DEL CEDENTE.-** El cedente no podrá transferir a ningún tercero los derechos que conforman la estructura, secuencia y organización del programa de ordenador que es objeto del presente contrato, como tampoco emplearlo o utilizarlo a título personal, ya que siempre se deberá guardar la exclusividad del programa de ordenador a favor del Cesionario.

**CUARTA: CUANTIA.-** La cesión objeto del presente contrato, se realiza a título gratuito y por ende el Cesionario ni sus administradores deben cancelar valor alguno o regalías por este contrato y por los derechos que se derivan del mismo.

**QUINTA: PLAZO.-** La vigencia del presente contrato es indefinida.

**SEXTA: DOMICILIO, JURISDICCIÓN Y COMPETENCIA.-** Las partes fijan como su domicilio la ciudad de Quito. Toda controversia o diferencia derivada de éste, será resuelta directamente entre las partes y, si esto no fuere factible, se solicitará la asistencia de un Mediador del Centro de Arbitraje y Mediación de la Cámara de Comercio de Quito. En el evento que el conflicto no fuere resuelto mediante este procedimiento, en el plazo de diez días calendario desde su inicio, pudiendo prorrogarse por mutuo acuerdo este plazo, las partes someterán sus controversias a la resolución de un árbitro, que se sujetará a lo dispuesto en la Ley de Arbitraje y Mediación, al Reglamento del Centro de Arbitraje y Mediación de la Cámara de comercio de Quito, y a las siguientes normas: a) El árbitro será seleccionado conforme a lo establecido en la Ley de Arbitraje y Mediación; b) Las partes renuncian a la jurisdicción ordinaria, se obligan a acatar el laudo arbitral y se comprometen a no interponer ningún tipo de recurso en contra del laudo arbitral; c) Para la ejecución de medidas cautelares, el árbitro está facultado para solicitar el auxilio de los funcionarios públicos, judiciales, policiales y administrativos, sin que sea necesario recurrir a juez ordinario alguno; d) El procedimiento será confidencial y en derecho; e) El lugar de arbitraje serán las instalaciones del centro de arbitraje y mediación de la Cámara de Comercio de Quito; f) El idioma del arbitraje será el español; y, g) La reconvenición, caso de haberla, seguirá los mismos procedimientos antes indicados para el juicio principal.

**SÉPTIMA: ACEPTACIÓN.-** Las partes contratantes aceptan el contenido del presente contrato, por ser hecho en seguridad de sus respectivos intereses.

En aceptación firman a los 22 días del mes de Abril del dos mil catorce.

f) \_\_\_\_\_  
C.C. 171754779-4  
**CEDENTE**

f) \_\_\_\_\_  
Instituto Superior Tecnológico Cordillera  
**CESIONARIO**



---

## AGRADECIMIENTO

Gracias infinita al INSTITUTO TECNÓLOGICO SUPERIOR "CORDILLERA", Carrera de Análisis en Sistemas, por haberme abierto las puertas para emprender la formación académica profesional, y en especial mis sinceros agradecimientos al director de tesis Ing. Richard Mafla por su ayuda y guía oportuna en el desarrollo de mi proyecto.

Con mucho amor y cariño a mis padres Laura Lincango y Luis Cuichán, a mis hermanos Erika Cuichán y Aldair Cuichán, que gracias a su ejemplo, motivación, consejos, esfuerzo, amor brindado generaron la fortaleza que me ayudó a sobrellevar las dificultades que surgieron en el camino trazado para conseguir esta meta, tan importante dentro de mi vida profesional.

También deseo dedicar este logro a toda mi familia y amigos que estuvieron pendientes y animándome para alcanzalo con éxito.



## **DEDICATORIA**

A Dios nuestra fuerza de amor y vida

A mis padres por su apoyo incondicional



## ÍNDICE GENERAL

| <b>CONTENIDO</b>   | <b>Pág.</b> |
|--|-------------|
| DECLARACIÓN DE APROBACIÓN DE TUTOR Y LECTOR .....          | i           |
| DECLARACIÓN DE AUDITORÍA DEL ESTUDIANTE .....              | ii          |
| CONTRATO DE CESIÓN SOBRE DERECHOS PROPIEDAD INTELECTUAL..  | iii         |
| AGRADECIMIENTOS .....                                      | v           |
| DEDICATORIA .....  | vi          |
| CONTENIDO .....  | vii         |
| ÍNDICE DE TABLAS .....                                     | xi          |
| ÍNDICE DE FIGURAS .....                                    | xii         |
| RESUMEN EJECUTIVO .....                                    | xvi         |
| ABSTRACT .....   | xvii        |
| INTRODUCCIÓN .....   | xviii       |
| <br>   |             |
| <b>Capítulo I: Antecedentes .....</b>                      | <b>1</b>    |
| 1.01 Contexto .....  | 1           |
| 1.02 Justificación e Importancia.....                      | 2           |
| 1.03 Matriz T .....  | 3           |
| 1.03.01 Análisis de la matriz T .....                      | 4           |
| <b>Capítulo II.....</b>                                    | <b>5</b>    |
| 2.01 Mapeos de Involucrados.....                           | 5           |
| 2.02 Mapeos de Involucrados.....                           | 6           |
| 2.03 Matriz de Análisis de Involucrados .....              | 6           |
| 2.03.01 Análisis de involucrados .....                     | 7           |
| <b>Capítulo III .....</b>                                  | <b>8</b>    |
| 3.01 Árbol de Problemas .....                              | 8           |
| 3.01 Análisis del árbol de problemas.....                  | 9           |
| 3.02 Árbol de Objetivos .....                              | 10          |
| 3.02.01 Análisis del árbol de objetivos. ....              | 11          |
| <b>Capítulo IV .....</b>                                   | <b>12</b>   |
| 4.01 Matriz de Análisis de Alternativas .....              | 12          |
| 4.01.01 Análisis de la matriz de alternativas.....         | 12          |
| 4.02 Matriz de Análisis de Impactos de los Objetivos. .... | 12          |



|   | <b>Pág.</b> |
|---|-------------|
| 4.02.01 Análisis sobre la matriz de impacto de los objetivos..... | 12          |
| 4.03 Diagrama de estrategias.....                                 | 13          |
| 4.03.01 Análisis de la matriz de alternativas.....                | 14          |
| 4.04 Matriz de Marco Lógico.....                                  | 14          |
| 4.04.01 Análisis de la matriz del marco lógico.....               | 14          |
| <b>Capítulo V.....</b>  | <b>18</b>   |
| 5.01 Justificación Técnica.....                                   | 18          |
| 5.02 Análisis y Diseño.....                                       | 20          |
| 5.02.01 Diagrama de Caso de Uso General.....                      | 20          |
| 5.02.02 Diseño de Casos de Uso.....                               | 21          |
| 5.02.03 Diagramas de Secuencia.....                               | 32          |
| 5.02.04 Diagrama de Colaboración.....                             | 38          |
| 5.02.05 Diagrama de Componentes.....                              | 43          |
| 5.02.06 Diagrama de Clases.....                                   | 45          |
| 5.02.07 Diagrama Lógico.....                                      | 46          |
| 5.02.07 Diagrama Físico.....                                      | 47          |
| 5.03 Desarrollo.....  | 48          |
| 5.03.01 Arquitectura del Sistema.....                             | 48          |
| 5.03.02 Lista de Comandos.....                                    | 50          |
| 5.03.03 Estándares de Base de Datos.....                          | 52          |
| 5.03.04 Estándares de Diseño UML.....                             | 53          |
| 5.03.05 Diseño de Interfaces.....                                 | 63          |
| 5.04 Pruebas.....   | 74          |
| 5.04.01 Pruebas de Integridad.....                                | 75          |
| 5.04.02 Pruebas de Módulo.....                                    | 75          |
| 5.04.03 Pruebas de Interfaz de Usuario.....                       | 76          |
| <b>Capítulo VI.....</b>   | <b>77</b>   |
| 6.01 Recursos.....  | 77          |
| 6.02 Presupuesto.....   | 78          |
| 6.03 Cronograma de Actividades.....                               | 78          |
| <b>Capítulo VII.....</b>  | <b>79</b>   |
| 7.01 Conclusiones.....  | 79          |

|  | <b>Pág.</b> |
|--|-------------|
| 7.02 Recomendaciones .....                               | 79          |
| <b>Anexos .....</b>                                      | <b>81</b>   |
| A.01 Matriz de Análisis de Involucrados .....            | 82          |
| A.02 Matriz de Análisis de Alternativas .....            | 84          |
| A.03 Matriz de Análisis de Impacto de los Objetivos..... | 85          |
| A.04 Matriz de Marco Lógico .....                        | 86          |
| A.05 Cronograma de Actividades .....                     | 89          |
| A.06 Instalación de Programas.....                       | 90          |
| A.06.01 Instalación del Servidor de Base de Datos.....   | 91          |
| A.06.02 Instalación del NetBeans IDE 7.2.....            | 99          |
| A.07 Script de la Base de Datos en Mysql .....           | 106         |
| A.08 Manual Técnico .....                                | 114         |
| A.09 Manual de Usuario .....                             | 149         |
| A.09.01 Login .....                                      | 149         |
| A.09.02 Bodega .....                                     | 149         |
| A.09.03 Secretaria.....                                  | 153         |
| A.09.04 Administrador .....                              | 157         |



## ÍNDICE TABLAS

| <b>Contenido</b>  | <b>Pág.</b> |
|---|-------------|
| Tabla 1: Matriz T .....                                       | 3           |
| Tabla 2: Matriz de Análisis de Involucrados .....             | 82          |
| Tabla 3: Matriz de Análisis de Alternativas.....              | 84          |
| Tabla 4: Matriz de Análisis de Impacto de los Objetivos ..... | 85          |
| Tabla 4: Matriz de Análisis de Impacto de los Objetivos ..... | 86          |
| Tabla 5: UC Consultar Producto .....                          | 21          |
| Tabla 6: UC Consultar Producto .....                          | 21          |
| Tabla 7: UC Realizar Pedido .....                             | 22          |
| Tabla 8: UC Catálogo de Producto .....                        | 23          |
| Tabla 9: UC Control de inventario.....                        | 24          |
| Tabla 10: UC Devolución de Producto .....                     | 25          |
| Tabla 11: UC Alerta Stock.....                                | 26          |
| Tabla 12: UC Informes.....                                    | 27          |
| Tabla 13: UC Facturar.....                                    | 28          |
| Tabla 14: UC Cancelar Factura.....                            | 29          |
| Tabla 15: UC Anular Factura.....                              | 30          |
| Tabla 16: UC Devolucion de Producto .....                     | 31          |
| Tabla 17: Listado de Comandos HTML .....                      | 50          |
| Tabla 18: Tipos de Datos de una DB .....                      | 52          |
| Tabla 19: Ingreso al Sistema .....                            | 63          |
| Tabla 20: Administracion de Clientes.....                     | 64          |
| Tabla 21: Agregar Cliente.....                                | 64          |
| Tabla 22: Administrar Proveedor.....                          | 65          |
| Tabla 23: Agregar Proveedor .....                             | 66          |
| Tabla 24: Administrar Categoria de Producto .....             | 66          |
| Tabla 25: Agregar Categoria.....                              | 67          |
| Tabla 26: Adminstrar Bodega .....                             | 68          |
| Tabla 27: Nueva Bodega.....                                   | 68          |
| Tabla 28: Administrar Orden de Requisición .....              | 69          |
| Tabla 29: Nueva Orden de Requisición .....                    | 70          |
| Tabla 30: Catalogo .....                                      | 70          |



---

|  | <b>Pág.</b> |
|--|-------------|
| Tabla 31: Administración de Productos .....          | 71          |
| Tabla 32: Agregar Productos .....                    | 72          |
| Tabla 33: Guías de Remisión .....                    | 73          |
| Tabla 34: Nueva guía de Remisión .....               | 73          |
| Tabla 35: Administración Facturas de Proveedor ..... | 74          |
| Tabla 36: Recursos .....                             | 77          |
| Tabla 37: Presupuesto .....                          | 78          |
| Tabla 38: Cronograma de Actividades .....            | 89          |



## ÍNDICE DE FIGURAS

| <b>Contenido</b>  | <b>Pág.</b> |
|---|-------------|
| Figura 1: Matriz de Involucrados .....                        | 6           |
| Figura 2: Árbol de Problemas .....                            | 8           |
| Figura 3: Árbol de Objetivos .....                            | 10          |
| Figura 4: Matriz de Marco Lógico Matriz de Marco Lógico ..... | 86          |
| Figura 5: Diagrama de Caso de Uso General .....               | 20          |
| Figura 6: CU001 Consultar Producto .....                      | 21          |
| Figura 7: CU002 Realizar Pedido .....                         | 22          |
| Figura 8: CU003 Catálogo de Producto .....                    | 23          |
| Figura 9: CU004 Control de inventario .....                   | 24          |
| Figura 10: CU005 Devolución de Producto .....                 | 25          |
| Figura 11: CU006 Alerta Stock .....                           | 26          |
| Figura 12: CU007 Informes .....                               | 27          |
| Figura 13: CU008 Facturar .....                               | 28          |
| Figura 14: CU009 Cancelar Factura .....                       | 29          |
| Figura 15: CU010 Anular Factura .....                         | 30          |
| Figura 16: CU011 Devolución de Producto .....                 | 31          |
| Figura 17: DS001 Consultar Producto .....                     | 32          |
| Figura 18: DS002 Realizar Pedido .....                        | 32          |
| Figura 19: DS003 Catálogo de Producto .....                   | 33          |
| Figura 20: DS004 Control de inventario .....                  | 33          |
| Figura 21: DS005 Devolución de Producto .....                 | 34          |
| Figura 22: DS006 Alerta Stock .....                           | 34          |
| Figura 23: DS007 Informes .....                               | 35          |
| Figura 24: DS008 Facturar .....                               | 35          |
| Figura 25: DS009 Cancelar Factura .....                       | 36          |
| Figura 26: DS010 Anular Factura .....                         | 36          |
| Figura 27: DS011 Devolución de Producto .....                 | 37          |
| Figura 28: DC001 Consultar Producto .....                     | 38          |
| Figura 29: DC002 Realizar Pedido .....                        | 38          |
| Figura 30: DC003 Catálogo de Producto .....                   | 39          |
| Figura 31: DC004 Control de inventario .....                  | 39          |



---

|  | <b>Pág.</b> |
|--|-------------|
| Figura 32: DC005 Devolución de Producto.....         | 40          |
| Figura 33: DC006 Alerta Stock .....                  | 40          |
| Figura 34: DC007 Informes .....                      | 41          |
| Figura 35: DC008 Facturar .....                      | 41          |
| Figura 36: DC009 Cancelar Factura .....              | 42          |
| Figura 37: DC010 Anular Factura.....                 | 42          |
| Figura 38: DC011 Devolución de Producto.....         | 43          |
| Figura 39: Capa Acceso a Datos .....                 | 43          |
| Figura 40: Capa Lógica del Negocio .....             | 44          |
| Figura 41: Capa Presentación.....                    | 44          |
| Figura 42: Diagrama de Clases .....                  | 45          |
| Figura 43: Diagrama Lógico .....                     | 46          |
| Figura 44: Diagrama Físico.....                      | 47          |
| Figura 45: Arquitectura del Sistema .....            | 49          |
| Figura 46: Ingreso al Sistema.....                   | 63          |
| Figura 47: Administrar Clientes.....                 | 63          |
| Figura 48: Agregar Clientes .....                    | 64          |
| Figura 49: Administrar Proveedor .....               | 65          |
| Figura 50: Agregar Proveedor.....                    | 65          |
| Figura 51: Administrar Categorías de Productos ..... | 66          |
| Figura 52: Nueva Categoría .....                     | 67          |
| Figura 53: Adminisarar Bodega.....                   | 67          |
| Figura 54: Agregar Bodega.....                       | 68          |
| Figura 55: Administrar Ordenes de Requisición.....   | 69          |
| Figura 56: Nueva Orden de Requisición.....           | 69          |
| Figura 57: Catalogo.....                             | 70          |
| Figura 58: Administrar Producto .....                | 71          |
| Figura 59: Agregar Producto.....                     | 72          |
| Figura 60: Administrar Guías de Remisión .....       | 72          |
| Figura 61: Nueva Guía de Remisión.....               | 73          |
| Figura 62: Administrar Facturas de Proveedor .....   | 74          |
| Figura 63: Instalación de Wmapserver .....           | 90          |



---

|  | <b>Pág.</b> |
|--|-------------|
| Figura 64: Términos de licencia de Wampserver .....                    | 90          |
| Figura 65: Directorio de Instalación de WampServer.....                | 91          |
| Figura 66: Accesos Directos de WampServer .....                        | 91          |
| Figura 67: Install de WampServer .....                                 | 92          |
| Figura 68: Procesos de Instalación de WampServer.....                  | 92          |
| Figura 69: Explorador de Firefox por defecto para WampServer.....      | 93          |
| Figura 70: Explorador de Windows como browser para el WampServer ..... | 93          |
| Figura 71: Servidor SMTP y un Email de WampServer.....                 | 94          |
| Figura 72: Completando la Instalación de WampServer .....              | 94          |
| Figura 73: Icono del WampServer en la Barra de Tareas .....            | 95          |
| Figura 74: Cambio de Idioma en el WampServer.....                      | 95          |
| Figura 75: Inicio de Servicio de WampServer .....                      | 95          |
| Figura 76: Pantalla de Presentación de WampServer. ....                | 96          |
| Figura 77: Cambio de Puerto .....                                      | 96          |
| Figura 78: Texto ID.....   | 97          |
| Figura 79: Cambio por puerto 82.....                                   | 97          |
| Figura 80: Cambio en localhost .....                                   | 97          |
| Figura 81: Cambio texto de Puerto .....                                | 98          |
| Figura 82: Explorador .....  | 98          |
| Figura 83: Instalador Netbeans .....                                   | 99          |
| Figura 84: Ejecutar como administrador.....                            | 99          |
| Figura 85: Esperando Instalador .....                                  | 99          |
| Figura 86: Carga de Instalador.....                                    | 100         |
| Figura 87: Pantallada de Instalador.....                               | 100         |
| Figura 88: Términos de licencia.....                                   | 101         |
| Figura 89: Aceptar Terminos .....                                      | 101         |
| Figura 90: Ruta de Instalación .....                                   | 102         |
| Figura 91: Ventana de Instalación.....                                 | 102         |
| Figura 92: Instalar .....  | 103         |
| Figura 93: Esperar Instalación .....                                   | 103         |
| Figura 94: Concluir Instalación.....                                   | 104         |
| Figura 95: Acceso Directo .....  | 104         |



---

|  | <b>Pág.</b> |
|--|-------------|
| Figura 96: Abrir Programa .....                  | 104         |
| Figura 97: Ventana principal de NetBeans .....   | 105         |
| Figura 98: Programa.....                         | 105         |
| Figura 99: Ingreso al Sistema.....               | 149         |
| Figura 100: Crear Bodega .....                   | 149         |
| Figura 101: Administra Bodega .....              | 150         |
| Figura 102: Crear Proveedor .....                | 150         |
| Figura 103: Administrar Proveedor .....          | 151         |
| Figura 104: Crear Categoría.....                 | 151         |
| Figura 105: Administrar Categoría .....          | 152         |
| Figura 106: Nuevo Producto .....                 | 152         |
| Figura 107: Administrar Producto .....           | 153         |
| Figura 108: Facturas Proveedor .....             | 153         |
| Figura 109: Orden de Requisición .....           | 154         |
| Figura 110: Guías de Remisión.....               | 155         |
| Figura 111: Menu.....                            | 157         |
| Figura 112: Administrar Usuarios.....            | 157         |
| Figura 113: Ordenes de Compras.....              | 158         |
| Figura 114: Administrar Ordenes de Compras ..... | 158         |



---

## RESUMEN EJECUTIVO

El presente proyecto tiene como objetivo general el control de pedidos e inventario dentro de la empresa de material eléctrico IMPORT HNR ubicado en Quito, parroquia de Cotocollao. En el cual se realizará un análisis de requerimientos y necesidades, que permitan proponer el desarrollo de un sistema web de control de pedidos e inventario.

Mediante esta aplicación se busca mejorar la atención al público y recepción de pedidos, como también elevar la eficacia de la empresa. La seguridad al realizar consultas de productos de una manera más eficaz, precisa y facilitar la toma de decisiones de la empresa cuya acción es mejorar el tiempo de servicio.

El sistema permitirá mantener la información en forma digital y actualizada de cada cliente, proveedor, producto y pedido, que será de mucha ayuda para el personal de la empresa.

El administrador, el personal y los clientes son los beneficiarios directos del sistema ya que se ajusta a sus necesidades con la finalidad de optimizar recursos, procesos y actividades.

Este sistema será diseñado para no limitarse solo a esta empresa sino que sirva para otras empresas que dispongan con similar servicio.

---

## ABSTRACT

This project has the overall objective order and inventory control within the company IMPORT HNR electrical equipment located in Quito, Cotacollao parish. In which an analysis of needs and requirements, allowing propose the development of a web control system will take orders and inventory.

Using this application is to improve service to the public and receiving orders, as well as enhance the effectiveness of the company. The security product queries more efficiently, accurately and facilitates decision -making of the company whose action is to improve the service time.

The system permitirá keep information in digital form and updated customer, supplier, product, and order, which will be helpful for the company staff.

The manager, the staff and customers are the direct beneficiaries of the system as it fits your needs in order to optimizar resources, processes and activities.

This system will be designed to not be limited only to the company but to serve with other companies that have similar service.



---

## INTRODUCCIÓN

El objetivo primordial de toda empresa es obtener utilidades y la obtención de utilidades obviamente reside en gran parte de Ventas, ya que éste es el motor de la empresa, sin embargo, si la función del inventario no opera con efectividad, ventas no tendrá material suficiente para poder trabajar, el cliente quedaría insatisfecho,

Muchas de las empresas se encuentran en constantes cambios e innovación en la atención a los clientes que requieren de sus servicios y productos ya que es de suma importancia que el cliente se sienta satisfecho por la atención brindada por los empleados de dicha institución.

La siguiente tesis presenta de manera simple la información de seguimiento al desarrollo de un sistema con el propósito de optimizar los procesos de pedidos e inventarios que actualmente se lleva a cabo en la empresa.

La toma de pedidos es parte vital de las empresas comerciales, productoras e incluso de servicio, es indispensable para estas empresas tener un control eficiente de los productos. Esto es para mejorar un desempeño en sus actividades y un mayor aprovechamiento de los recursos humanos y de tiempo. Por ende las empresas están automatizando los procesos y así mantener una información actualizada y ordenada.

Con la investigación realizada se puede concluir en que el desarrollo y la implementación del sistema se lograrán satisfacer a las necesidades y los requerimientos del personal, a los clientes y así brindar un servicio de calidad.

## Capítulo I: Antecedentes

### 1.01 Contexto

El control de inventario es el proceso de operaciones para definir el costo total del inventario. Mantener un nivel de almacenamiento adecuado para cumplir con los pedidos. El control de inventario es una operación principal de una compañía u organización que maneja la recepción, almacenamiento y distribución de elementos.

Las empresas se encuentran en constantes cambios e innovación en la atención a los clientes que requieren de sus servicios y productos ya que es de suma importancia que el cliente se sienta satisfecho por la atención brindada por los empleados de dicha institución.

Por ejemplo, la información asociada a las características de un producto tales como precio, marca, proveedor, cantidad, etc. se los lleva de forma física y se los guarda en carpetas. Además llevan un registro manuscrito de los productos que posee en inventario con sus respectivos movimientos de entrada y salida.

Un sistema de inventario es un conjunto de políticas y controles que supervisan los niveles de inventario y establece cuales son los niveles que debe mantenerse, cuando hay que ordenar un pedido y de qué tamaño deben hacerse. Una forma práctica de establecer un sistema de inventario es llevar la cuenta de cada artículo que sale del almacén y colocar una orden por más existencias cuando los inventarios lleguen a un nivel predeterminado.

Con estos antecedentes y en función a la investigación realizada, es indispensable implementar una herramienta tecnológica automatizada que mejore y controle los procesos de atención de una manera rápida, clara y precisa, para facilitar la gestión los administradores, los mismos que analizan e interpretan los pedidos que

genera el sistema, optimizando del tiempo, y garantizar la integridad de la información.

El desarrollo del proyecto llevara a cabo toda la investigación necesaria para cumplir con el objetivo de la propuesta.

## **1.02 Justificación e Importancia**

El presente proyecto busca crear un modelo que ayude y permita el control de toma de pedidos e inventarios para optimizar el proceso de atención al cliente.

Es conveniente desarrollar la herramienta informática e implementarla, porque está orientada a mejorar los problemas que afronta actualmente dicha empresa en todos los procesos de control, mediante modelo integral que dispone la herramienta informática para optimizar el tiempo en la atención al cliente. La toma de pedidos es parte vital de las empresas comerciales, productoras e incluso de servicio, es indispensable para estas empresas tener un control eficiente de los productos, bodegas y caja. Esto es para mejorar un desempeño en sus actividades y un mayor aprovechamiento de los recursos humanos, de tiempo, materiales, financiero.

El proyecto se lo va a realizar debido a que en la empresa surge la necesidad de complementar nuevos procedimientos, técnicas y estrategias para servir a sus clientes. Desarrollar el sistema tiene como objetivo automatizar el control de inventario. Se busca mejorar el sistema de atención al público y recepción de pedidos de la empresa. Otros beneficios con que contara la aplicación, son el control de diferentes tipos de usuarios, capacidad de implementarse en empresas que manejen producto terminado, obtención de información actualizada y detallada, reducción de tiempos en entrega de pedidos.

### 1.03 Matriz T

En la siguiente matriz describela situaciones mediante la cual describiremos los extremosa evitar, la problemática en el presente y lo que buscamos al hacer nuestra investigación. Además describe las fuerzas mediante la cual son calificadas con la intensidad y el potencial de campo en la escala de 1 al 5.

**Tabla 1**

*Matriz T del Problema Central*

| ANÁLISIS DE FUERZAS T   |   |           |          |           |   |
|---|---|-----------|----------|-----------|---|
| Situación Empeorada   | Situación Actual  |           |          |           | Situación Mejorada  |
| <b>Cientes descontentos por información y atención incorrecta que se le facilita.</b> | <b>Procesos y registros de pedidos realizados de forma manualmente.</b> |           |          |           | <b>Mejorar la calidad de atención de la empresa a los clientes.</b> |
| <b>Fuerzas Impulsadoras</b>   | <b>I</b>  | <b>PC</b> | <b>I</b> | <b>PC</b> | <b>Fuerzas Bloqueadoras</b>   |
| Optimizar el proceso de atención.   | 3   | 4         | 3        | 4         | Procesos incorrectos en los registros de pedidos.                   |
| Estabilizar la administración de la información.                                      | 3   | 4         | 2        | 4         | Demasiado tiempo de espera en la atención.                          |
| El uso de la información de manera digital.   | 4   | 3         | 3        | 4         | Confusión con la información de los pedidos.                        |
| Mejorar el ambiente de trabajo.   | 2   | 3         | 2        | 3         | La ausencia de personal en atender a los clientes.                  |
| Apoyo al desarrollo de la empresa.  | 1   | 3         | 1        | 4         | No existe reporte de pedidos  |
| La gestión de pedidos desde cualquier lugar.  | 2   | 4         | 3        | 4         | Registro de datos en forma manual.                                  |
| Evitar cambios en información.  | 2   | 3         | 2        | 4         | Limitación para que los cliente sean informados.                    |

*Nota:* Establece las fuerzas bloqueadoras e impulsadoras.

---

### **1.03.01 Análisis de la matriz T**

Los problemas que afronta la empresa son por los registros solicitados a los clientes que se los realiza en forma manual, así como también reportes, por tal motivo el sistema web permitirá optimizar los procesos en la atención al cliente, estabilizar la administración de pedidos y mejorar el ambiente de trabajo e impulsando a la empresa a utilizar herramientas tecnológicas.

---

## Capítulo II: Involucrados

### 2.01 Mapeos de Involucrados

Los clientes concurren a la empresa para realizar pedidos y elevar su calidad de ventas que será beneficiado con el presente proyecto para la gestión de pedidos, administración de reportes, e inventario.

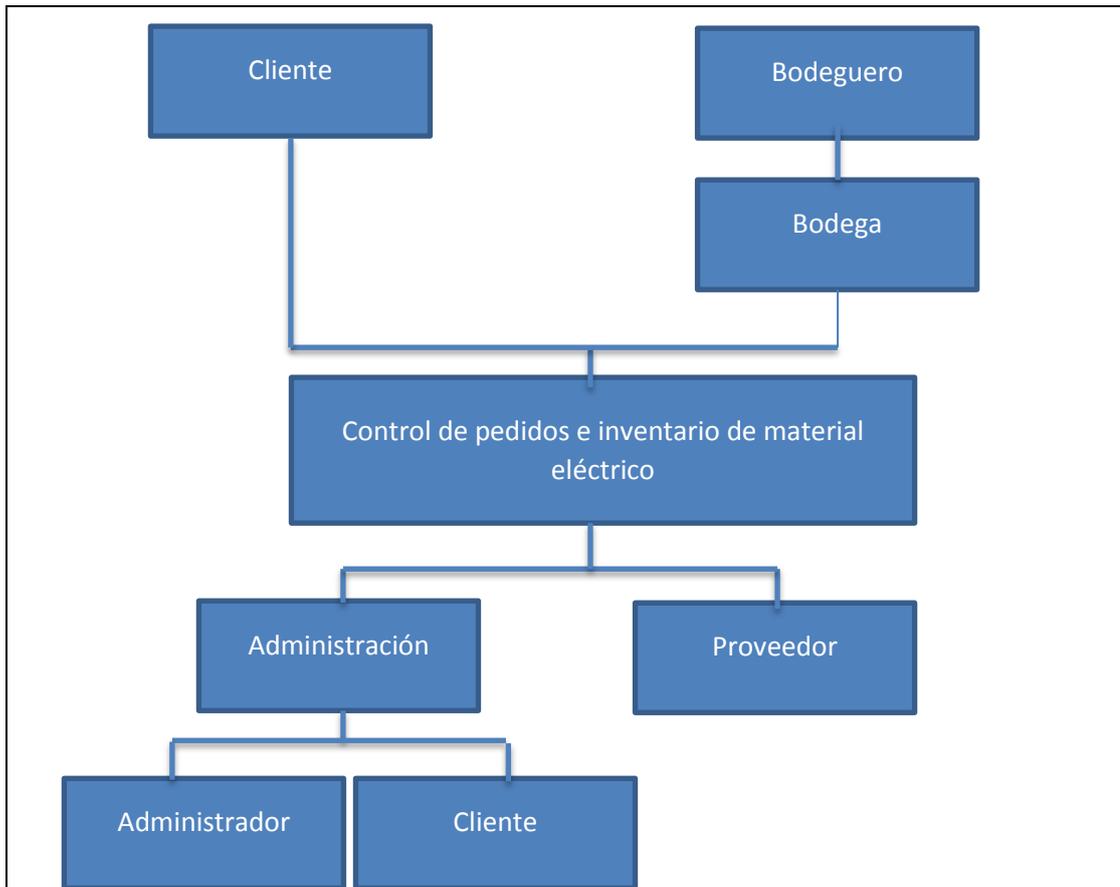
Los Empleados revisan, analizan, consultan, realizan los informes de forma manual, indican el procedimientos que realizan a los clientes atendidos. Se interesará en los módulos que se implementan en el proyecto, ayudará a solventar y definir los resultados esperados con la toma de decisiones que habitualmente realiza a las personas atendidas en la empresa.

El Bodeguero colabora en la atención a los clientes, con el propósito de informar, notificar, priorizar las actividades y la asignación de pedidos, tiene la dificultad de realizar las búsquedas de los pedidos de forma manual, el proyecto apoyará la coordinación de los procesos que realiza.

El Administrador es el responsable directo de lograr los objetivos, normas, mandatos, leyes, que rigen en la empresa y necesita ejercer su autoridad a través de factores en el cual ayuden a su ética, liderazgo y permita ejercer un cumplimiento considerable y satisfactorio.

Los Clientes son individuos que analizan el sector y describen que no hay la suficiente información sobre los procesos que realiza la empresa, por tal motivo el presente proyecto permitirá brindar la información necesaria a través de las disposiciones que emita la empresa.

## 2.02 Mapeos de Involucrados



*Figura 1:* Matriz de Involucrados Mapeo de Involucrados: El análisis de involucrados es un instrumento esencial que permite:

- Identificar a aquellas personas interesadas en un determinado proyecto.
- Establecer quienes pudieran ser afectadas por los objetivos del mismo.
- Explorar quiénes y cómo pueden contribuir el logro de los objetivo.

## 2.03 Matriz de Análisis de Involucrados

Tiene como propósito analizar a los involucrados con su respectivo interés, los problemas percibidos, en el desarrollo de sus actividades sociales a través de los recursos que mantiene el proyecto para comprobar la efectividad de su producción.

(Ver Anexo A.01).

---

## **Conclusión**

Con los problemas observados y analizados mediante los involucrados, la solución que se considera en el presente proyecto, tiene como finalidad mejorar los procesos, análisis y control de inventarios, adicionalmente ayudando a al cliente en obtener una atención de calidad.

### **2.03.01 Análisis de involucrados**

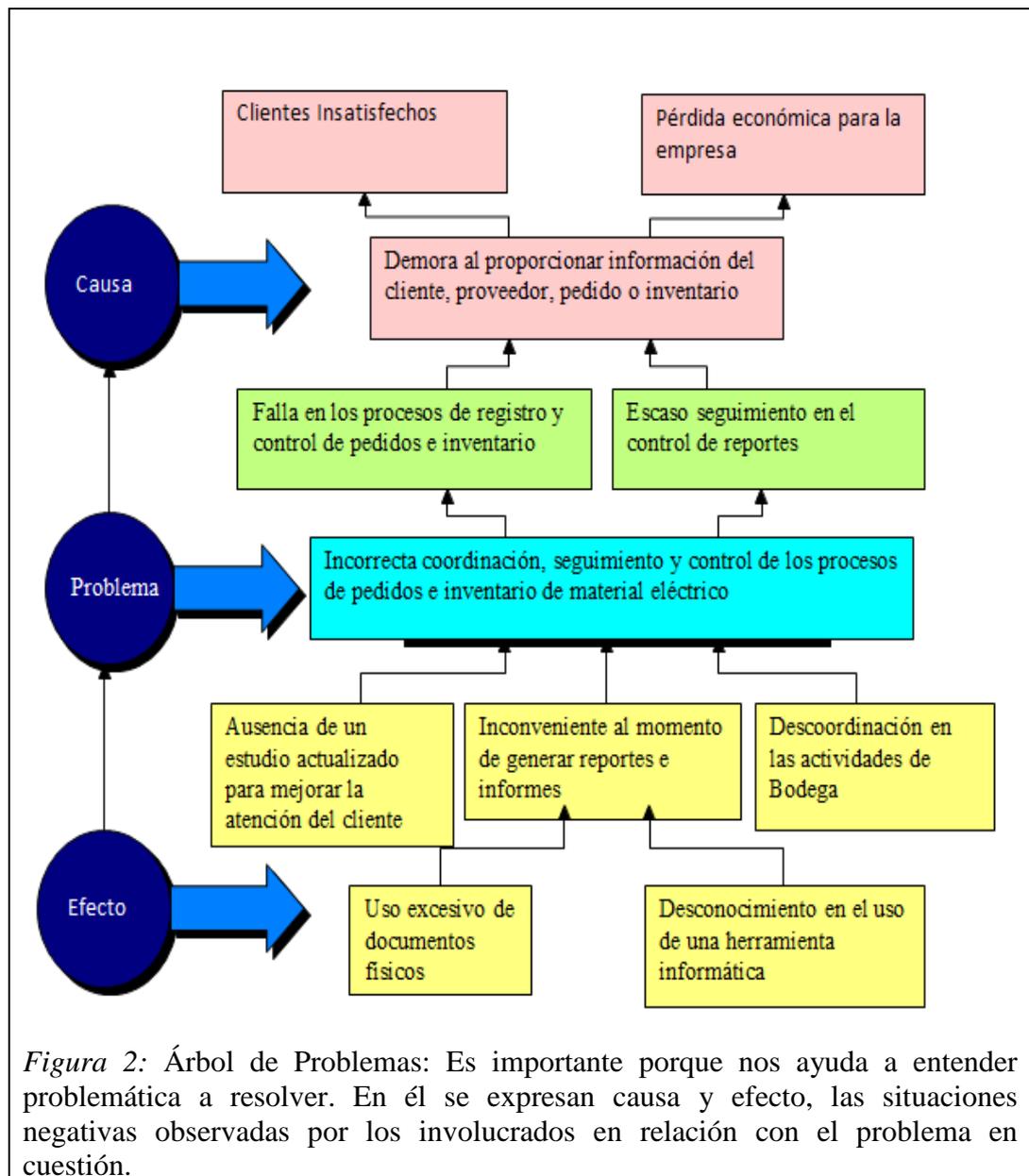
En la matriz mostrada anteriormente se analizarán las estrategias que realizarán los diferentes tipos de actores mencionados en el mapeo de involucrados, buscando soluciones a los distintos problemas presentados para el desarrollo del proyecto.

También se ha realizado el respectivo análisis de involucrados, los cuales muestran intereses sobre el problema central, generando diferentes interrogantes en cada uno de dichos involucrados permitiendo encontrar cuales son los puntos favorables y no favorables para su mejor desempeño dentro de la empresa, y así llegar a cumplir su objetivo, que es poder solucionar el problema central antes mencionado.

### Capítulo III: Árbol de Problemas - Objetivos

#### 3.01 Árbol de Problemas

En el siguiente árbol de problemas se describe los aspectos necesarios que nos permiten entender de mejor manera la problemática a resolver, listando las causas y sus efectos para de esta manera poder identificar todo el conjunto de problemas sobre el cual se concentran enfocados los objetivos del proyecto.



*Figura 2:* Árbol de Problemas: Es importante porque nos ayuda a entender problemática a resolver. En él se expresan causa y efecto, las situaciones negativas observadas por los involucrados en relación con el problema en cuestión.



---

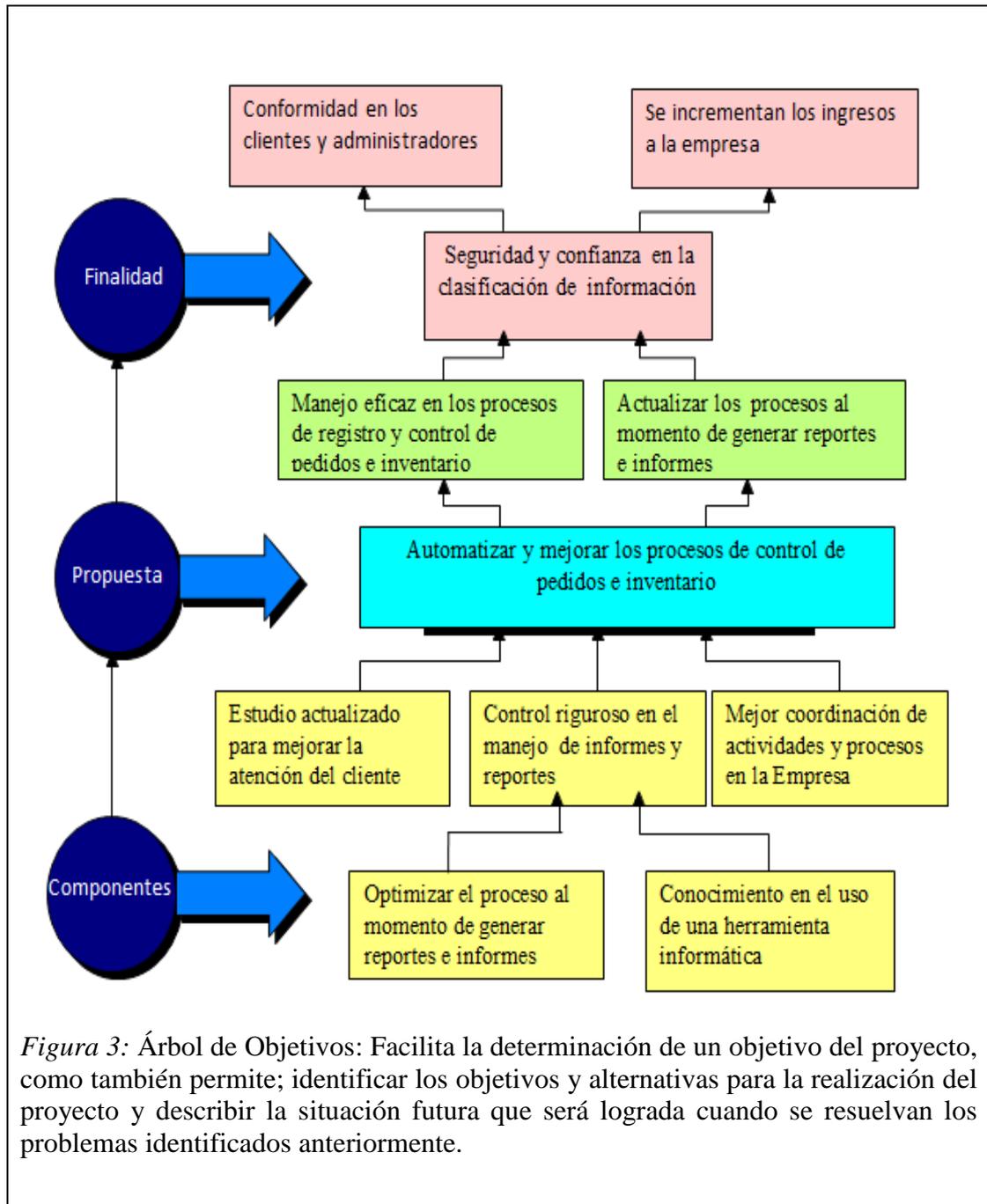
### **3.01 Análisis del árbol de problemas.**

El desconocimiento de nuevas herramientas tecnológicas, provoca la dificultad para dar un seguimiento a los requerimientos de un determinado cliente, así mismo el deficiente control de pedidos generan una defectuosa organización de los documentos e información de la empresa IMPORT. Ocasionando que la atención de cada cliente sea poca satisfactoria, incumpliendo con las expectativas del administrador, produciendo fallas en los procesos de control de pedidos e inventario siendo uno de los más principales procesos de la Empresa IMPORT.

Los pedidos mediante vía telefónica provocan una pérdida de tiempo tanto al cliente, como al administrador lo que ocasiona que disminuya la productividad. La pérdida de credibilidad en la gestión de cada proceso provoca un incremento de tiempo de archivo y disminución en la productividad, debido a la complejidad de los procesos y el excesivo papeleo. Los mismos que provocan un disgusto al administrador por la falta de organización en los documentos.

### 3.02 Árbol de Objetivos

A continuación se describe la solución a todos los inconvenientes identificados en el árbol de problemas, convirtiendo a los componentes en medios para encarar el problema de desarrollo identificado y proporcionar el sistema necesario para mejorar los procesos de pedidos e inventario de la empresa.



*Figura 3:* Árbol de Objetivos: Facilita la determinación de un objetivo del proyecto, como también permite; identificar los objetivos y alternativas para la realización del proyecto y describir la situación futura que será lograda cuando se resuelvan los problemas identificados anteriormente.



---

### **3.02.01 Análisis del árbol de objetivos.**

Se reduce el tiempo en el proceso para control de pedidos e inventario, induce a la facilidad para el manejo de la información de cada cliente, al igual que se genera los respectivos reportes e informes, fomentando una correcta organización en las áreas de bodega, secretaria y administración de la empresa IMPORT.

La satisfacción en los administradores y personal por la precisión en los procesos, se verá reflejada en el trabajo y desempeño de cada uno, gracias a la sencillez del manejo de información y el limitado papeleo; incrementando un control riguroso de la organización y manejo de la documentación.

---

## Capítulo IV: Análisis de Alternativas

### 4.01 Matriz de Análisis de Alternativas

En la matriz de alternativas tomamos en cuenta los objetivos a cumplir por medio de los indicadores que señalan el porcentaje de importancia, los mismos que aclaran los impactos que generan los propósitos para mejorar el registro de pedidos el control inventarios, tiene como finalidad categorizar las propuestas que se encuentran involucradas en el proyecto.(Ver Anexo A.02 )

#### 4.01.01 Análisis de la matriz de alternativas

Mide las soluciones en la calidad de atención a los clientes tomando en cuenta los procesos con los que se ampara en las capacitaciones sobre herramientas tecnológicas al personal técnico y administrativo a fin de orientar en los registros de control de pedidos e inventario, además se verifica un alto índice de apoyo en las distintas factibilidades presentadas, contando con el apoyo necesario para culminar con éxito los requerimientos, la búsqueda de soluciones y el apoyo tecnológico para la empresa IMPORT.

### 4.02 Matriz de Análisis de Impactos de los Objetivos.

Posteriormente de realizar un análisis de cada objetivo se verificó la factibilidad los mismos que interpretan la incidencia de los impacto que genera los propósitos y metas, tiene como finalidad categorizar las propuestas y posteriormente comparar los objetivos que se encuentran involucrados en el proyecto (Ver anexo A. 03).

#### 4.02.01 Análisis sobre la matriz de impacto de los objetivos

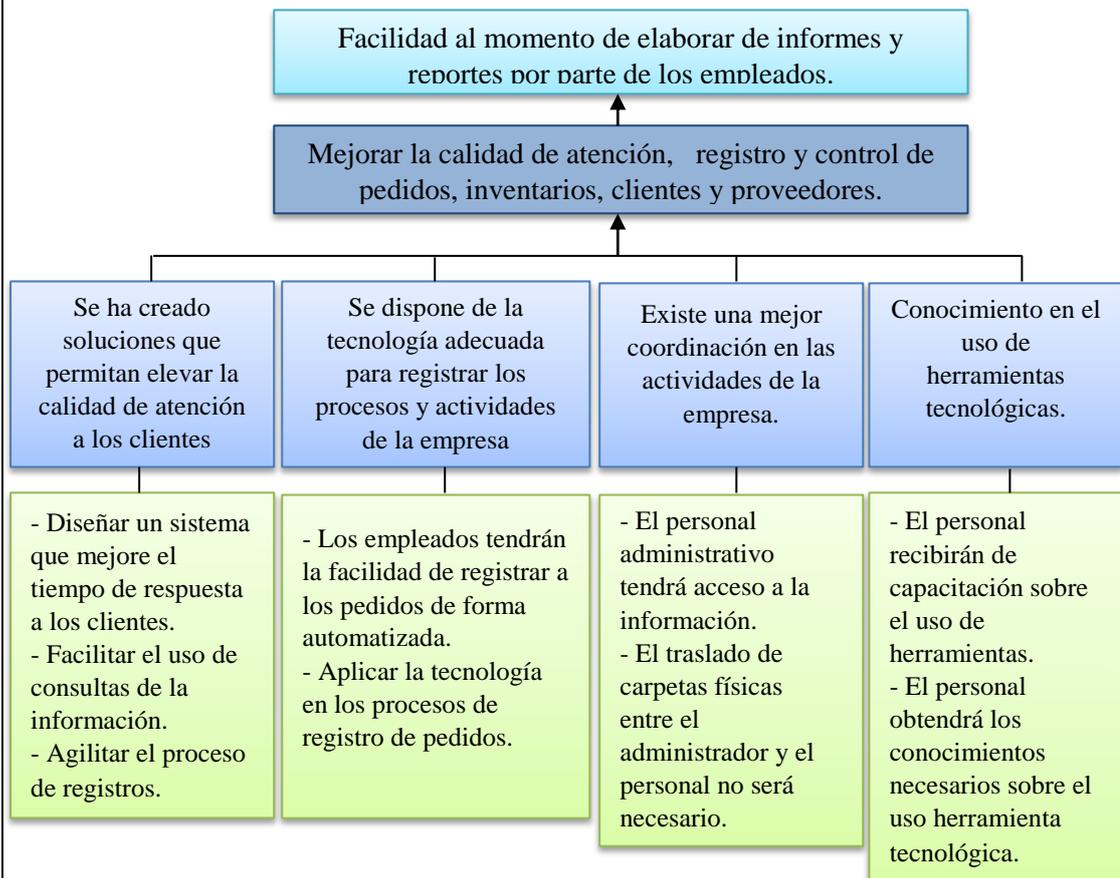
Luego de observar que la factibilidad a conseguir en la optimización de los procesos de pedidos e inventario son adecuados en el desarrollo del proyecto, además se conseguirá mejorar el ambiente social y cultural, permitiendo responder todas las

expectativas de los beneficiarios y fortaleciendo la participación de los administradores y el personal, en los procesos y actividades que realizan en la empresa.

#### 4.03 Diagrama de estrategias

Este Diagrama toma en cuenta el árbol de problemas y objetivos con la finalidad de mostrar las actividades que se van a realizar dentro de la solución al problema, identificando la finalidad, el propósito y componentes del mismo.

**Tabla 2**  
Matriz de Análisis de Impacto de los Objetivos



-En este diagrama se detallan las actividades que se van a realizar para mantener el seguimiento y control actualizado de pedidos e inventarios.

#### **4.03.01 Análisis de la matriz de alternativas**

Nos dan la posibilidad de encontrar soluciones necesarias que permitan elevar la calidad de atención, registro de pedidos y control de inventarios, verificando que nuestra intención es facilitar la elaboración de informes a los administradores, para lo cual contamos con la tecnología y las capacitaciones necesarias, las actividades y los procesos a realizar van en conjunto de las soluciones encontradas para el proyecto.

#### **4.04 Matriz de Marco Lógico**

En esta matriz permite describir los resultados anunciados en el proyecto, analizando los indicadores, medios de verificación y supuestos del proyecto, mostrando el estado inicial del problema y como se ha ido desarrollando las soluciones, además se analiza los elementos de las posibles dificultades que se pueden dar.

(Ver anexo A. 04).

##### **4.04.01 Análisis de la matriz del marco lógico**

Un **indicador** se usa para describir un problema. Los indicadores nos permiten recolectar datos y son útiles para poder medir los resultados obtenidos, procesos o acciones específicos, con el fin de obtener el análisis de una situación, comparar las características o para evaluar las variaciones de un evento.

Permite identificar las diferencias existentes entre los resultados planeados y obtenidos como base para la toma de decisiones, fijar el rumbo y alinear los esfuerzos hacia la consecución de las metas establecidas con el fin de lograr una mejora en los servicios.

---

## **Tipos de indicadores**

**Indicadores de cumplimiento:** tiene que ver con la conclusión de una tarea. Los indicadores de cumplimiento están relacionados con las razones que indican el grado de consecución de tareas y/o trabajos.

**Indicadores de evaluación:** tiene que ver con el rendimiento que se obtiene de una tarea, trabajo o proceso. Los indicadores de evaluación están relacionados con las razones y/o los métodos que ayudan a identificar nuestras fortalezas, debilidades y oportunidades de mejora.

**Indicadores de eficiencia:** tiene que ver con la actitud y la capacidad para llevar a cabo un trabajo o una tarea con el mínimo de recursos. Los indicadores de eficiencia están relacionados con las razones que indican los recursos invertidos en la consecución de tareas y/o trabajos.

**Indicadores de eficacia:** tiene que ver con hacer efectivo un intento o propósito. Los indicadores de eficacia están relacionados con las razones que indican capacidad o acierto en la consecución de tareas y/o trabajos.

**Indicadores de gestión:** tiene que ver con administrar y/o establecer acciones concretas para hacer realidad las tareas y/o trabajos programados y planificados. Los indicadores de gestión están relacionados con las razones que permiten administrar realmente un proceso.

Los **medios de verificación** corresponden a las fuentes de información que se utilizarán para obtener los datos necesarios para calcular los indicadores. La existencia de esta columna en la Matriz Marco Lógico tiene la ventaja de obligar a quién formula el proyecto y define los indicadores a identificar fuentes

existentes de información. O bien, si estas no están disponibles, a incluir en el diseño del proyecto actividades orientadas a recoger la información requerida.

Al preparar un proyecto, así como al evaluarlo y al planificar su ejecución es necesario recurrir a una serie de **supuestos**. Ello es consecuencia de que se está trabajando sobre un futuro hipotético e incierto. Lamentablemente, asociado a cada supuesto hay un riesgo: que el supuesto no se cumpla. Dependiendo del mismo impacto que el no cumplimiento del supuesto, este puede verse demorado, puede haber incrementado su costo, puede lograr sólo parte de sus objetivos.

Los procesos de registro de pedidos y control de inventarios desde el 2009, se han desarrollado de forma manual e inadecuada, partir de abril del 2014 se optimizará los procesos y actividades de manera automática, continuando hasta el 2016 o la disponibilidad de una nueva versión del software.

Los componentes planteados han permitido mejorar los procesos y actividades, con el fin de alcanzar todas las metas propuestas en el proyecto, incentivando al personal a utilizar la herramienta implementada.

Permite analizar a través de los indicadores, la optimización de los procesos de registro y control, tomando en cuenta el año inicial así como también el punto de terminación debido a que se toman hasta esa instancia los inconvenientes presentados.

Los servicios prestados a los clientes por parte de los empleados no ha sido optimizados dando como resultado una calidad de atención de tipo media-baja, a partir de abril del 2013, la calidad de atención será mejorada, actualizada y beneficiará a todos los clientes y personal de la empresa IMPORT.

Los procesos de registros y control de pedidos e inventarios serán desarrollados de forma rápida y automática. Los resultados de las encuestas realizadas han determinado que los procesos, las actividades y la atención brindada lo hacen de forma inadecuada.

Las encuestas elaboradas a los empleados sobre el uso de la herramienta tecnológica, han demostrado que tienen bajo conocimiento sobre el manejo de las aplicaciones, por lo tanto será necesario realizar un curso de capacitación sobre el uso y utilidad del software.

El análisis realizado a la situación actual de la Empresa, ha permitido crear nuevas estrategias, el cual optimice los procesos que realiza de forma eficiente y actualizada con la finalidad de brindar un buen servicio a los clientes.

Las necesidades que requiere el desarrollador del proyecto están marcadas en el presupuesto y los requerimientos se basan entorno y en función mediante el cual el proyecto se encuentre en la fase de desarrollo.

---

## Capítulo V: Desarrollo del Sistema

### 5.01 Justificación Técnica

Permite definir el diseño e identificar las actividades y procesos de control de pedidos e inventarios de forma clara y precisa, el cual es de vital importancia para la empresa IMPORT.

La metodología RUP (Proceso Unificado de Rational) es dirigida por los casos de uso, centrado en la arquitectura, iterativo e incremental, esta metodología en sus 4 fases (Inicio, elaboración, desarrollo y transición) establece la estructura que nos ayudara a la construcción de la presente aplicación. El diseño está orientado a un modo grafico llamado UML (Lenguaje de Modelado Unificado) lo que nos permitirá visualizar, especificar, construir y documentar un sistema, incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema y expresiones de lenguajes de programación, esquemas de base de datos. Lo cual ha permitido entender los procesos, procedimientos y actividades que se realizan en cada uno de los departamentos de la empresa IMPORT.

Los procesos serán mostrados mediante diagramas como: diagrama de caso de uso, diagrama de clases, diagrama de secuencia, diagrama de colaboración, diagrama de componentes, modelos lógico y físico.

Ayudara visualizar la necesidad y procedimientos que se realizan dentro de la organización además de crear y distribuir los diferentes modelos lógicos y físicos que dispone el sistema.

El sistema está diseñado en base a objetivos y condiciones, el sistema es de tres capas (presentación, negocio y datos) es una arquitectura cliente – aplicación - servidor siendo su objetivo la separación de la lógica de negocios de la lógica de



---

diseños. Además, permite distribuir el trabajo de creación de una aplicación por niveles.

La metodología y arquitectura ayudara a documentar el sistema de forma apropiada, cumpliendo con los requisitos necesarios que ayudaran a fomentar la elaboración de la aplicación.

## 5.02 Análisis y Diseño

### 5.02.01 Diagrama de Caso de Uso General

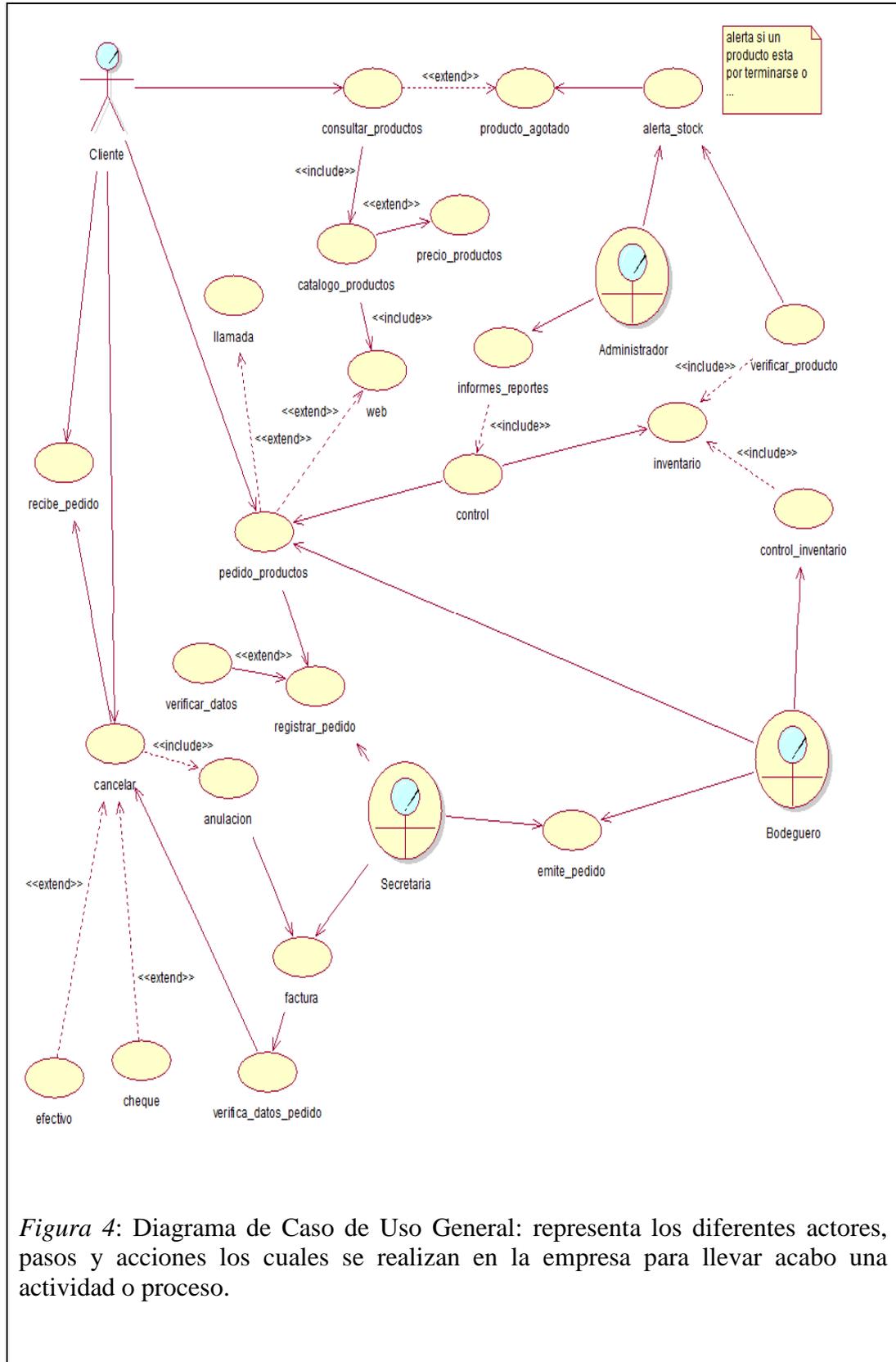
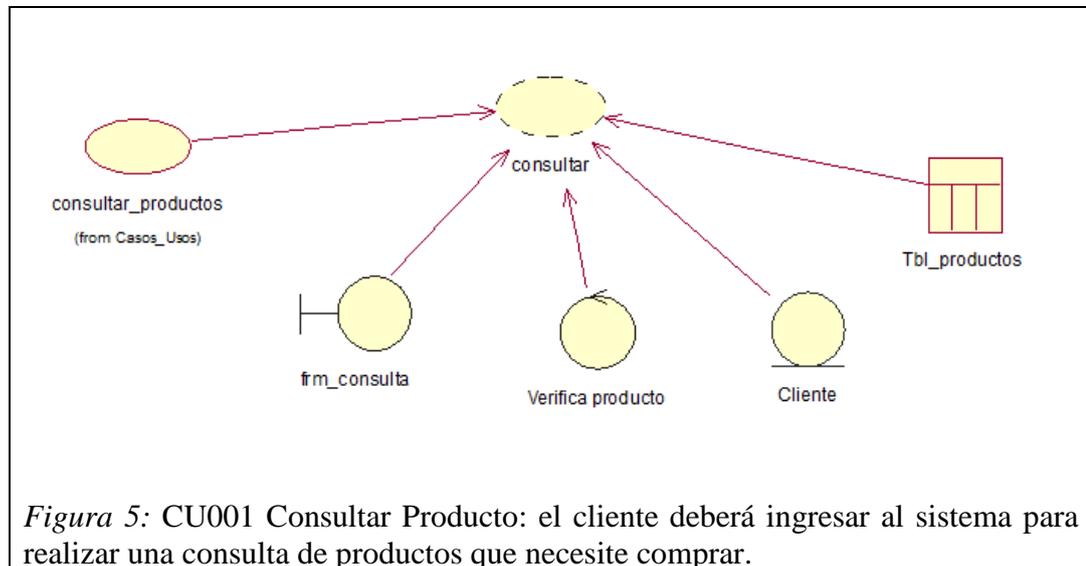


Figura 4: Diagrama de Caso de Uso General: representa los diferentes actores, pasos y acciones los cuales se realizan en la empresa para llevar a cabo una actividad o proceso.

## 5.02.02 Análisis y Diseño

### 5.02.02.01 CU 001 Consultar Producto



**Tabla 6**

#### Análisis CU-001 Consultar Producto

|                           |  |
|---------------------------|--|
| <b>ID</b>                 | CASO 001   |
| <b>Nombre:</b>            | Consultar Producto   |
| <b>Actores:</b>           | Cliente  |
| <b>Precondiciones:</b>    | Consultar producto vía web o llamada.<br>Disponer de un teléfono.<br>Mantener la información actualizada y ordenada. |
| <b>Flujo de Eventos:</b>  | 1. El cliente realiza una consulta de productos.<br>2. Consulta precios de productos.                                |
| <b>Flujo Alternativo:</b> | En caso que el producto no exista en el sistema web, el cliente realizara la consulta vía llamada telefónica.        |
| <b>Poscondiciones:</b>    | El flujo alternativo entrara en funcionalidad en caso de alguna contrariedad durante el proceso.                     |

5.02.02.02 CU 002 Realizar Pedidos

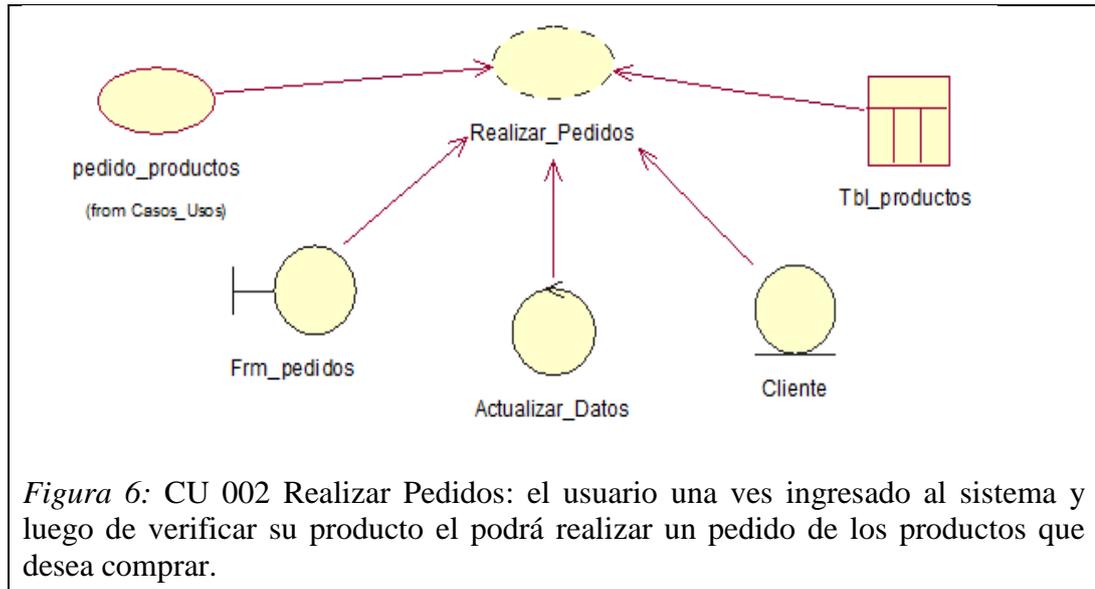


Tabla 7

Análisis CU-002 Realizar Pedido

|                           |   |
|---------------------------|---|
| <b>ID</b>                 | CASO 002  |
| <b>Nombre:</b>            | Realizar Pedido   |
| <b>Actores:</b>           | Cliente   |
| <b>Precondiciones:</b>    | Para realizar un pedido debe haber el producto en stock.<br>Disponer de un computador.<br>Mantener la información actualizada y ordenada. |
| <b>Flujo de Eventos:</b>  | 1. El cliente realiza su pedido vía web.<br>2. Realiza pedido de varios productos.  |
| <b>Flujo Alternativo:</b> | En caso que el producto no exista en el sistema web, el cliente realizara la consulta vía llamada telefónica.                             |
| <b>Poscondiciones:</b>    | El flujo alternativo entrara en funcionalidad en caso de alguna contrariedad durante el proceso.  |

5.02.02.03

CU 003 Catálogo de Productos

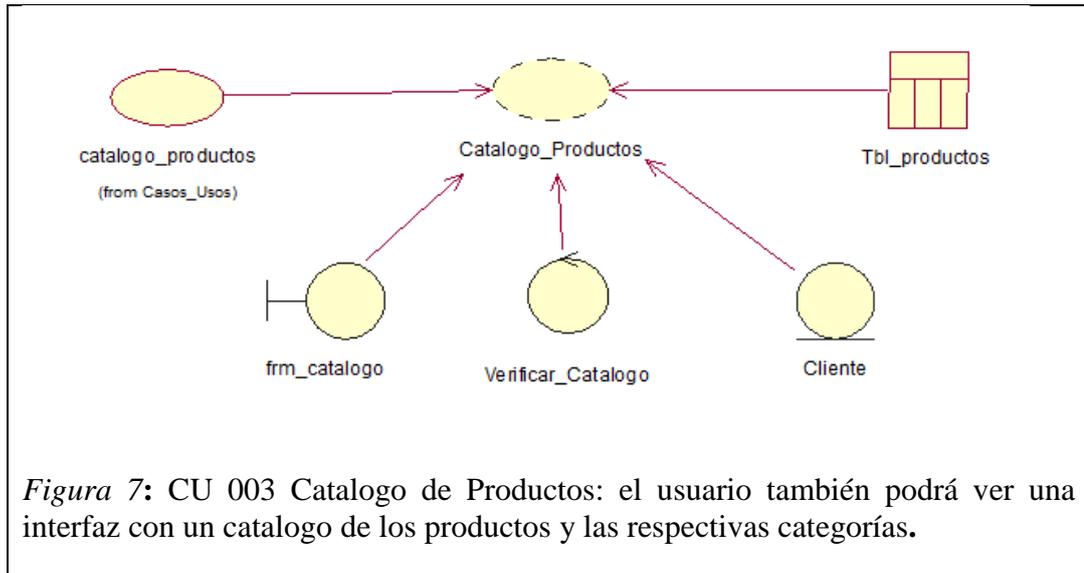


Tabla 8

Análisis CU-003 Catálogo de Productos

|                           |  |
|---------------------------|--|
| <b>ID</b>                 | CASO 003   |
| <b>Nombre:</b>            | Catálogo de Producto   |
| <b>Actores:</b>           | Cliente  |
| <b>Precondiciones:</b>    | Visualiza catálogo de producto vía web.<br>Disponer de un Computador.<br>Mantener la información actualizada y ordenada. |
| <b>Flujo de Eventos:</b>  | 1. El cliente visualiza un catálogo de productos.<br>2. Visualiza categorías de productos.                               |
| <b>Flujo Alternativo:</b> | En caso de que el pedido no exista se registre en el sistema web, el cliente realizara su pedido vía llamada telefónica. |
| <b>Poscondiciones:</b>    | El flujo alternativo entrara en funcionalidad en caso de alguna contrariedad durante el proceso.                         |

5.02.02.04

CU 004 Control de Inventario

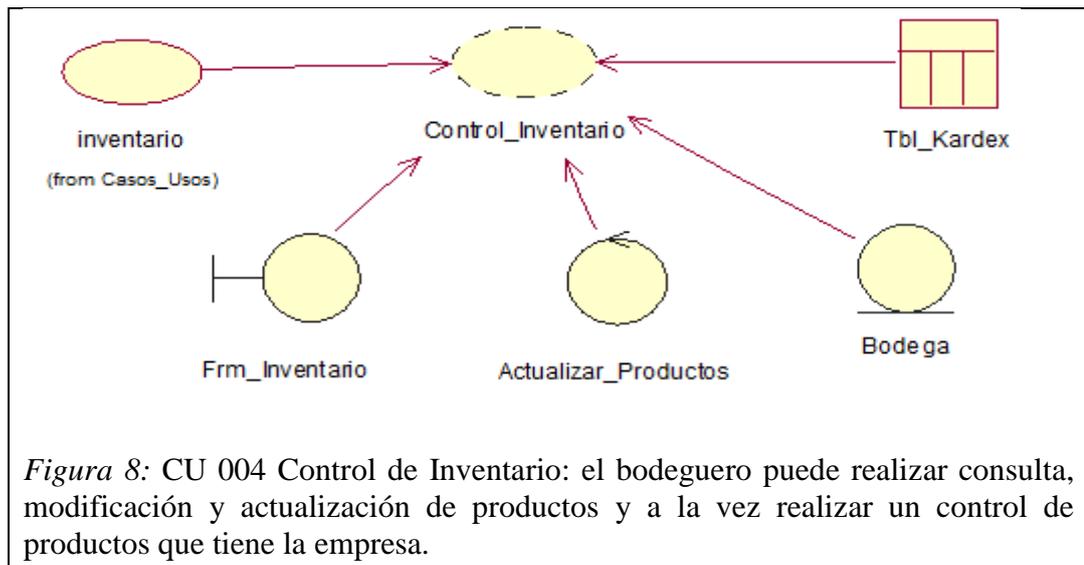


Tabla 9

Análisis CU-004 Control de inventario

|                           |   |
|---------------------------|---|
| <b>ID</b>                 | CASO 004  |
| <b>Nombre:</b>            | Control de inventario   |
| <b>Actores:</b>           | Bodeguero   |
| <b>Precondiciones:</b>    | Se realiza un control del inventario de todos los productos.<br>Disponer de un computador.<br>Mantener la información actualizada y ordenada. |
| <b>Flujo de Eventos:</b>  | 1. El bodeguero realiza un control de productos.<br>2. Realizar un control actualizado.   |
| <b>Flujo Alternativo:</b> | En caso de que el control no se lo realice correctamente, el bodeguero deberá poner un informe de retraso en realizar un control.             |
| <b>Poscondiciones:</b>    | El flujo alternativo entrara en funcionalidad en caso de alguna contrariedad durante el proceso.  |

5.02.02.05 CU 005 Devolución

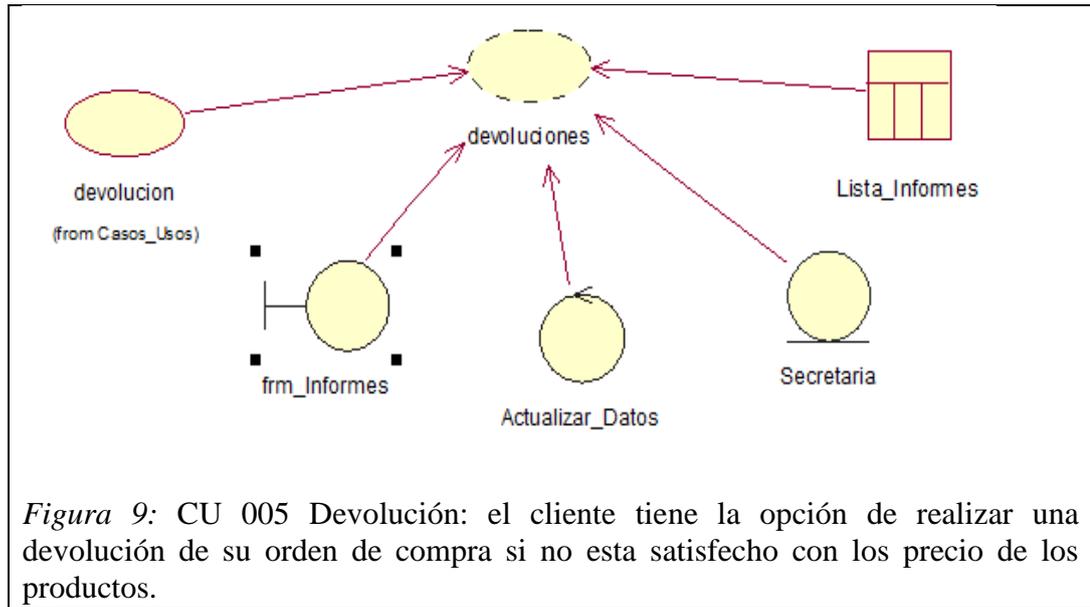


Figura 9: CU 005 Devolución: el cliente tiene la opción de realizar una devolución de su orden de compra si no esta satisfecho con los precio de los productos.

Tabla 10

Análisis CU-005 Devolución

|                           |   |
|---------------------------|---|
| <b>ID</b>                 | CASO 005  |
| <b>Nombre:</b>            | Devolución de Producto  |
| <b>Actores:</b>           | Cliente   |
| <b>Precondiciones:</b>    | El cliente realiza una devolución de producto a Bodega.<br>Verificar datos de devolución.<br>Se realiza un informe de devolución. |
| <b>Flujo de Eventos:</b>  | 1. El cliente realiza una devolución de producto.<br>2. Actualizar datos de productos e inventario.                               |
| <b>Flujo Alternativo:</b> | En caso que el producto sea devuelto, el cliente solicita el despacho del producto correcto.                                      |
| <b>Poscondiciones:</b>    | El flujo alternativo entrara en funcionalidad en caso de alguna contrariedad durante el proceso.                                  |

5.02.02.06

CU 006 Alerta de Stock

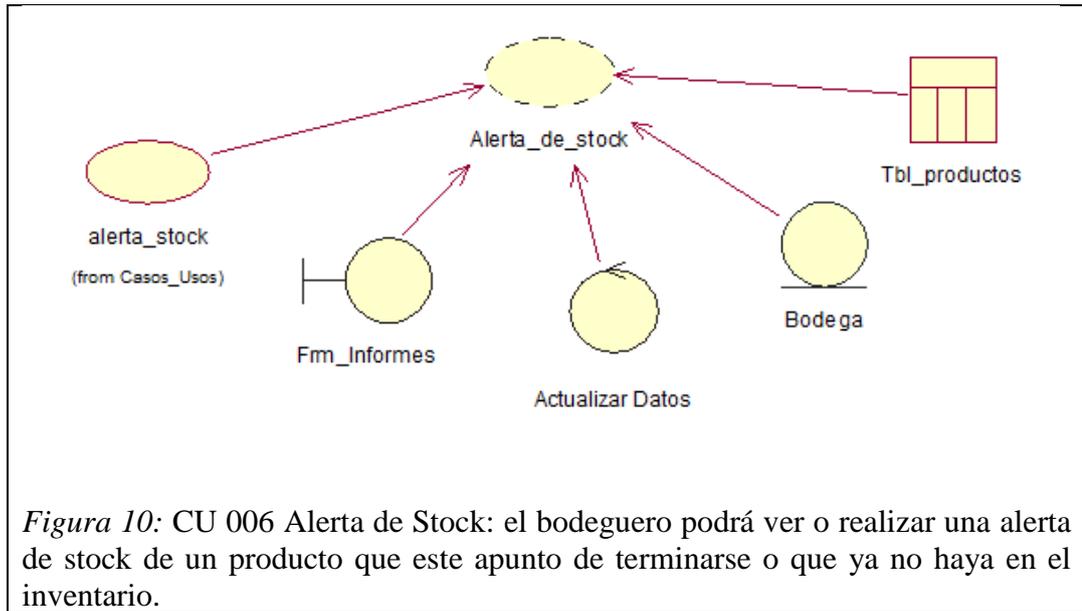


Figura 10: CU 006 Alerta de Stock: el bodeguero podrá ver o realizar una alerta de stock de un producto que este apunto de terminarse o que ya no haya en el inventario.

**Tabla 11**

*Análisis CU-006 Alerta Stock*

|                           |  |
|---------------------------|--|
| <b>ID</b>                 | CASO 006   |
| <b>Nombre:</b>            | Alerta Stock   |
| <b>Actores:</b>           | Bodeguero  |
| <b>Precondiciones:</b>    | Consultar existencia de productos.<br>Verificar datos de productos.<br>Mantener la información actualizada y ordenada. |
| <b>Flujo de Eventos:</b>  | 1. Realizar un chequeo en inventario sobre el stock de productos.<br>2. Actualizar datos de stock.                     |
| <b>Flujo Alternativo:</b> | En caso que el producto no exista en inventario, el bodeguero emitirá una alerta de stock bajo sobre el producto.      |
| <b>Poscondiciones:</b>    | El flujo alternativo entrara en funcionalidad en caso de alguna contrariedad durante el proceso.                       |

5.02.02.07

Caso 007 Informes o Reportes

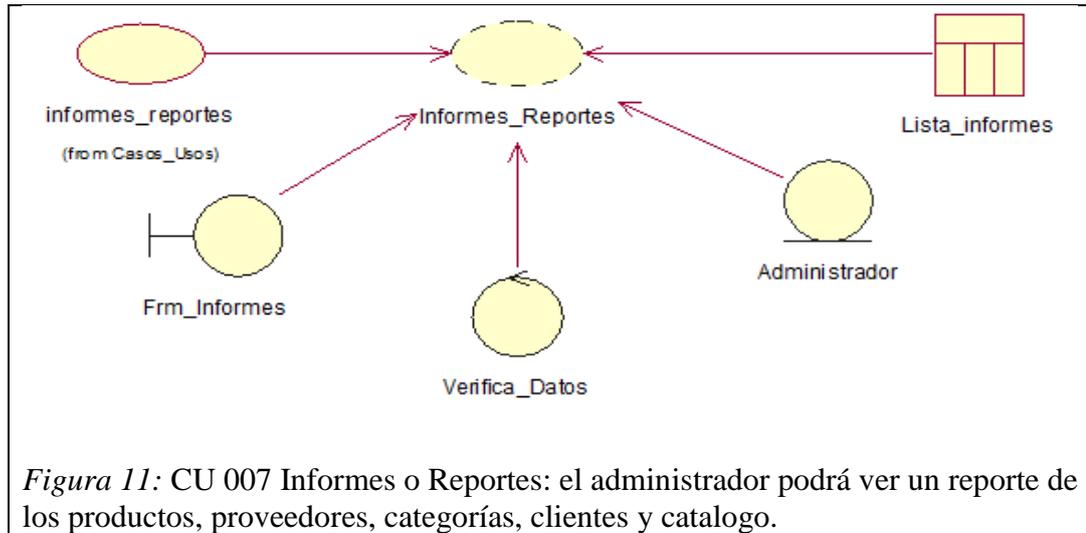


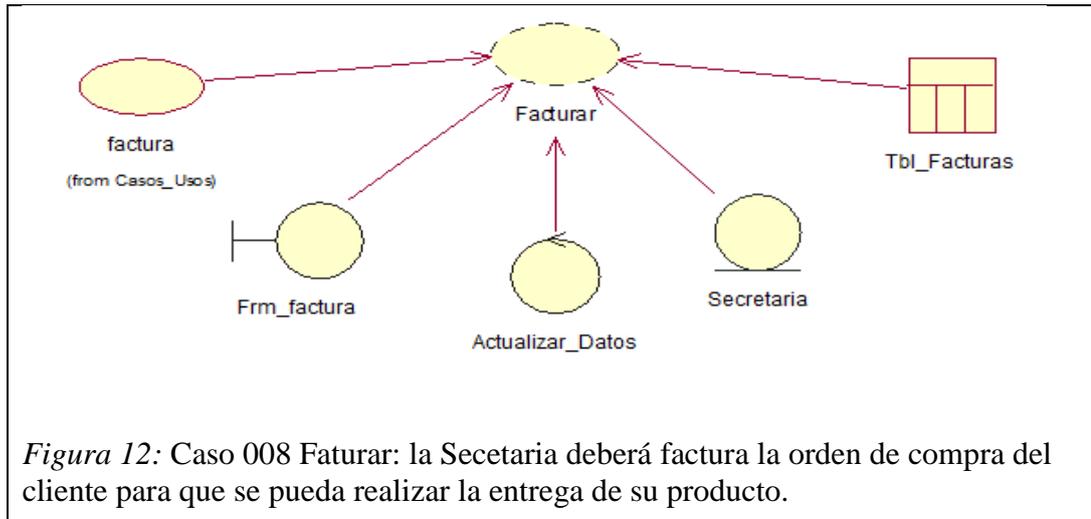
Figura 11: CU 007 Informes o Reportes: el administrador podrá ver un reporte de los productos, proveedores, categorías, clientes y catalogo.

Tabla 12

Análisis CU-007 Informes

|                           |   |
|---------------------------|---|
| <b>ID</b>                 | CASO 007  |
| <b>Nombre:</b>            | Informes  |
| <b>Actores:</b>           | Administrador, Secretaria, Bodega   |
| <b>Precondiciones:</b>    | Los informes deben ser entregados al administrador en periodos mensuales.<br>Los informes deben presentarse de forma detallada y especifica.<br>Mantener la información actualizada y ordenada. |
| <b>Flujo de Eventos:</b>  | 1. Ingresar las fechas de consulta para la generación de los informes.<br>2. Imprimir los informes.<br>3. Los informes deben ser entregados al administrador.                                   |
| <b>Flujo Alternativo:</b> | En el caso de que no se puedan generar los informes, estos deberán ser comunicados al administrador.  |
| <b>Poscondiciones:</b>    | El flujo alternativo entrara en funcionalidad en caso de alguna contrariedad durante el proceso.  |

5.02.02.08 Caso 008 Facturar



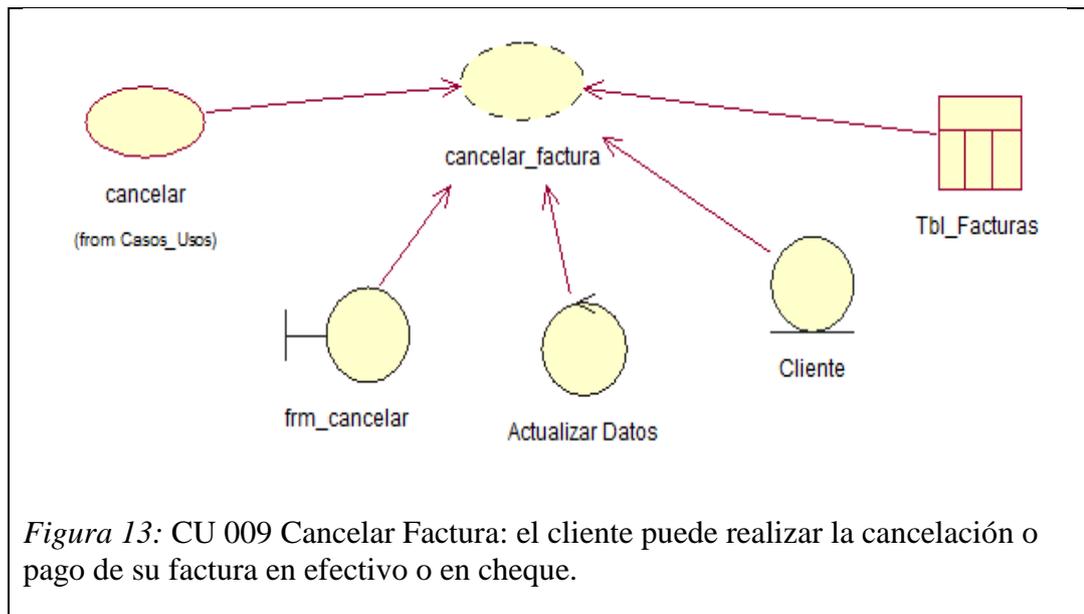
**Tabla 13**

*Análisis CU-008 Facturar*

|                           |  |
|---------------------------|--|
| <b>ID</b>                 | CASO 008   |
| <b>Nombre:</b>            | Facturar   |
| <b>Actores:</b>           | Secretaria   |
| <b>Precondiciones:</b>    | Los datos de la factura deben de ser correctos.<br>La factura debe ser entregada al cliente.<br>La información debe ser actualizada. |
| <b>Flujo de Eventos:</b>  | 1. Ingresar información de venta.<br>2. Imprimir factura.  |
| <b>Flujo Alternativo:</b> | En caso de que se registre incorrectamente la información, el sistema permitirá a la secretaria la corrección de la misma.           |
| <b>Poscondiciones:</b>    | El flujo alternativo entrara en funcionalidad en caso de alguna contrariedad durante el proceso.                                     |

5.02.02.09

CU 009 Cancelar Factura



**Tabla 14**

*Análisis CU-009 Cancelar Factura*

|                           |  |
|---------------------------|--|
| <b>ID</b>                 | CASO 009   |
| <b>Nombre:</b>            | Cancelar Factura   |
| <b>Actores:</b>           | Cliente  |
| <b>Precondiciones:</b>    | El cliente realiza el pago de su factura.<br>Se verifica si los datos son correctos.<br>La información debe ser actualizada. |
| <b>Flujo de Eventos:</b>  | 1. Recibe factura de su compra.<br>2. Consulta descuentos de su compra.  |
| <b>Flujo Alternativo:</b> | En caso de que no pueda pagar en efectivo, el cliente realizara su pago en cheque.   |
| <b>Poscondiciones:</b>    | El flujo alternativo entrara en funcionalidad en caso de alguna contrariedad durante el proceso.                             |

5.02.02.10 Caso 010 Anular Factura

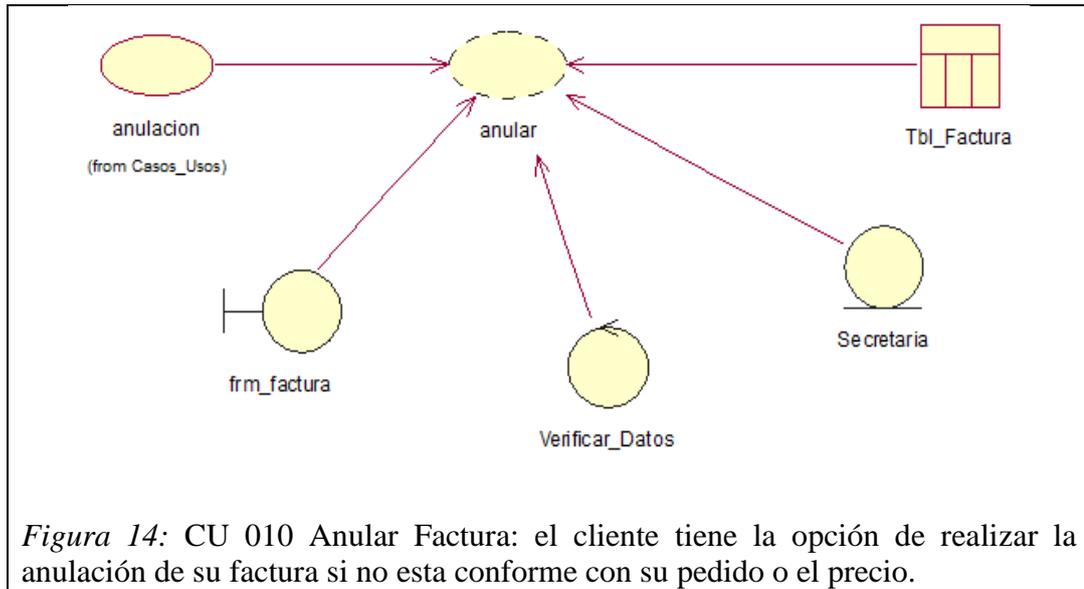


Figura 14: CU 010 Anular Factura: el cliente tiene la opción de realizar la anulación de su factura si no esta conforme con su pedido o el precio.

Tabla 15

Análisis CU-010 Anular Factura

|                           |   |
|---------------------------|---|
| <b>ID</b>                 | CASO 010  |
| <b>Nombre:</b>            | Anular Factura  |
| <b>Actores:</b>           | Cliente   |
| <b>Precondiciones:</b>    | El cliente solicita la anulación de su factura.<br>Se verifica si los datos son correctos y si su pedido es procedente.<br>La información debe ser actualizada. |
| <b>Flujo de Eventos:</b>  | 1. Recibe informe de anulación de la factura de su compra.<br>2. Consulta anulación de su compra.<br>3. Recibe documento de anulación.                          |
| <b>Flujo Alternativo:</b> | En caso de que no se pueda anular la factura, la secretaria realizara una nota de crédito.  |
| <b>Poscondiciones:</b>    | El flujo alternativo entrara en funcionalidad en caso de alguna contrariedad durante el proceso.  |

5.02.02.11

Caso 011 Devolución de Producto

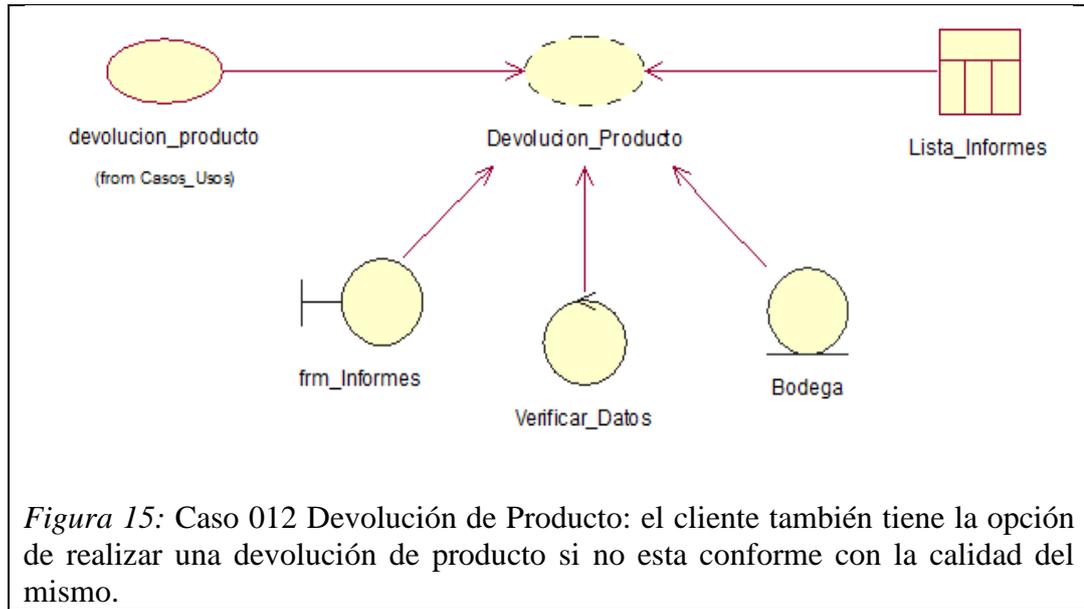


Figura 15: Caso 012 Devolución de Producto: el cliente también tiene la opción de realizar una devolución de producto si no esta conforme con la calidad del mismo.

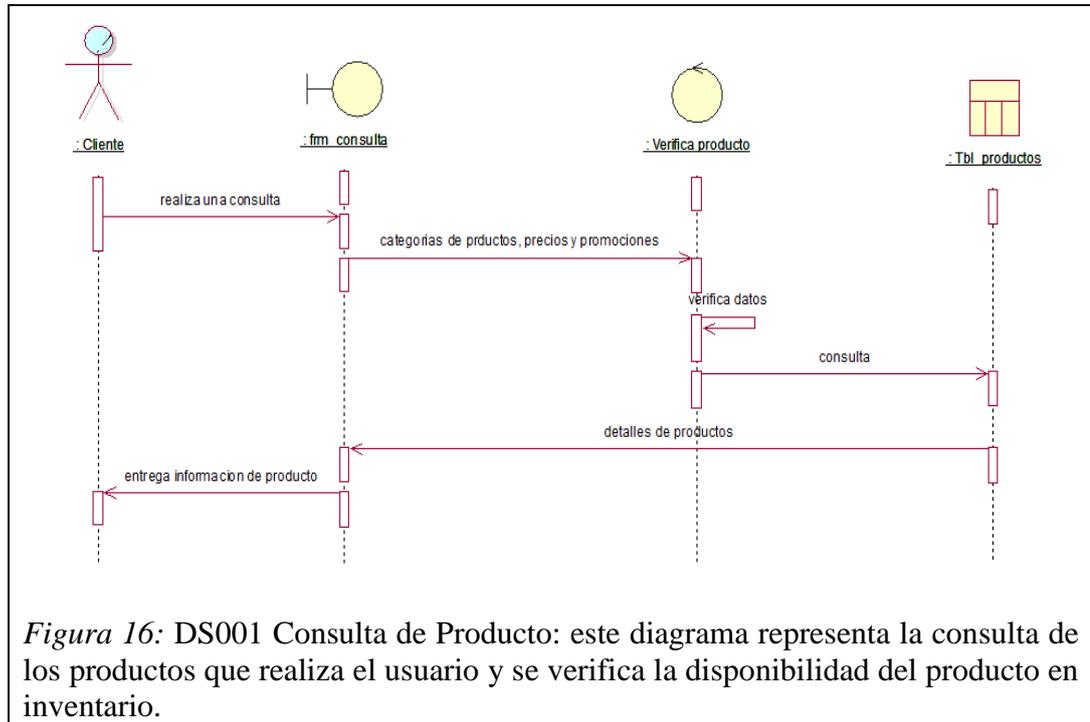
Tabla 16

Análisis CU-011 Devolución de Producto

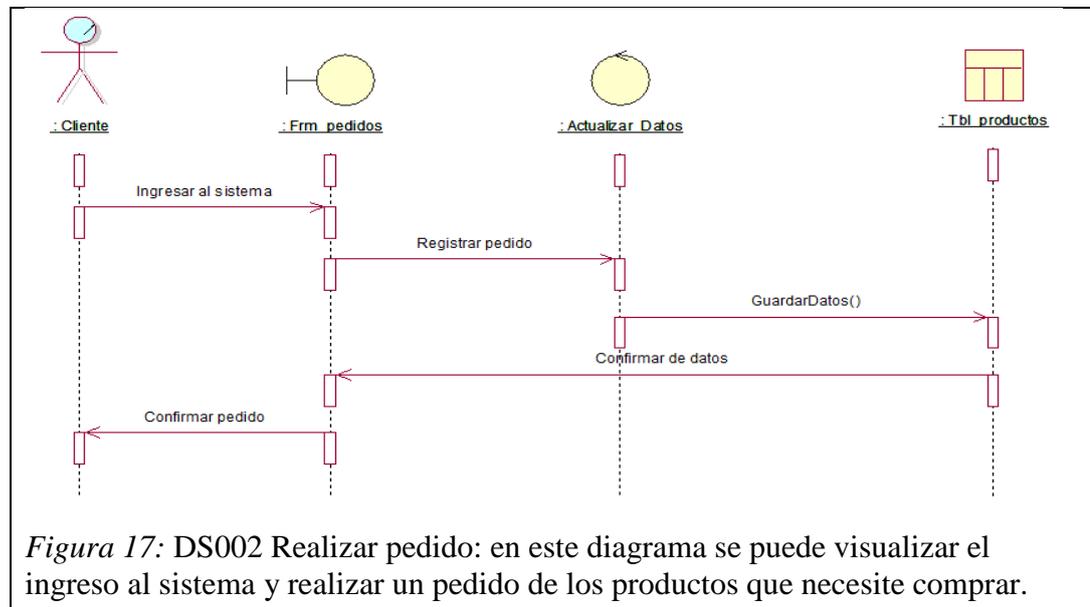
|                           |   |
|---------------------------|---|
| <b>ID</b>                 | CASO 012  |
| <b>Nombre:</b>            | Devolución de Producto  |
| <b>Actores:</b>           | Cliente   |
| <b>Precondiciones:</b>    | El cliente realiza devolución de producto.<br>Se verifica si los datos son correctos o incorrectos.<br>La información debe ser actualizada. |
| <b>Flujo de Eventos:</b>  | 1. Realiza devolución de producto.<br>2. Se realiza la anulación de su factura.   |
| <b>Flujo Alternativo:</b> | En caso de que no se pueda cancelar el producto, el cliente realizará una devolución.   |
| <b>Poscondiciones:</b>    | El flujo alternativo entrará en funcionalidad en caso de alguna contrariedad durante el proceso.  |

### 5.02.03 Diagramas de Secuencia

#### 5.02.03.01 DS001 Consulta de Producto



#### 5.02.03.02 DS002 Realizar pedido



### 5.02.03.03 DS003 Catalogo de Producto

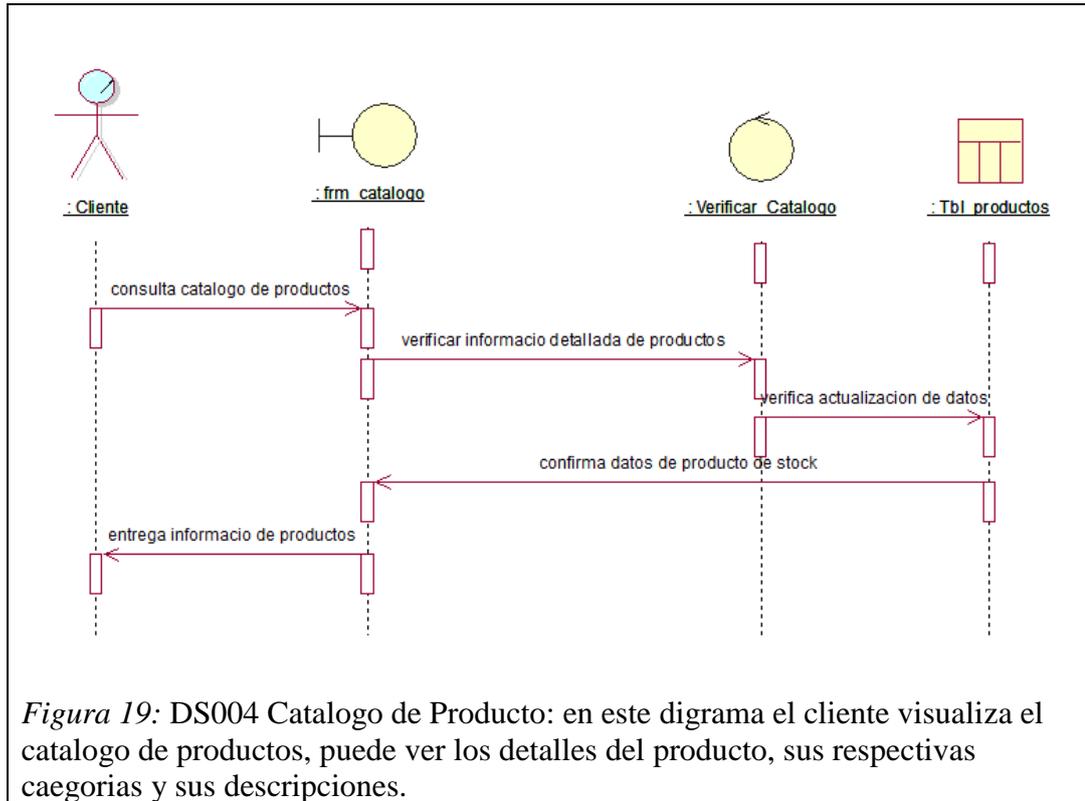


Figura 19: DS004 Catalogo de Producto: en este digrama el cliente visualiza el catalogo de productos, puede ver los detalles del producto, sus respectivas caegorias y sus descripciones.

### 5.02.03.04 DS004 Control de inventario

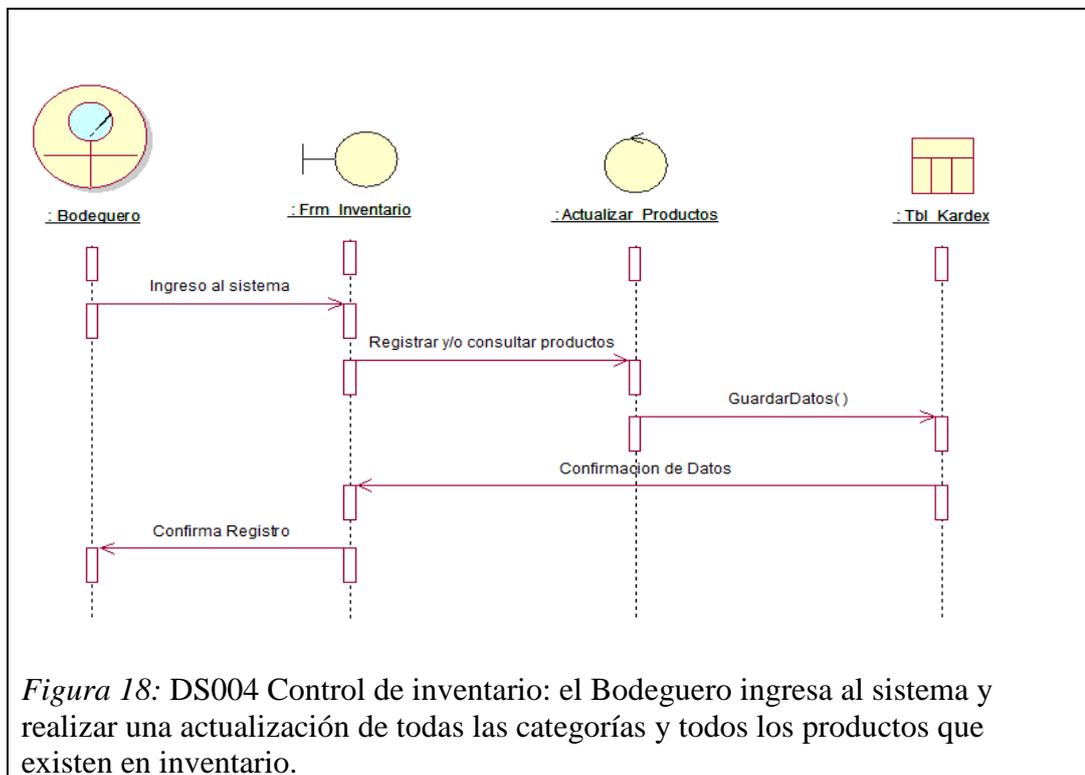
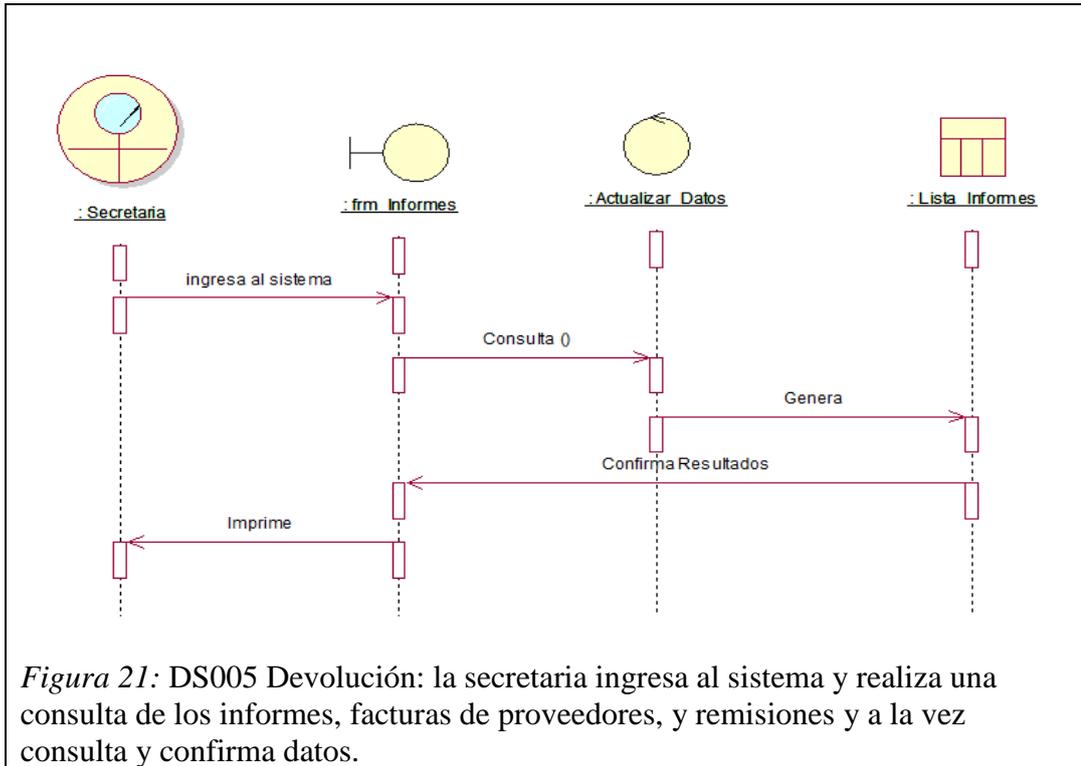
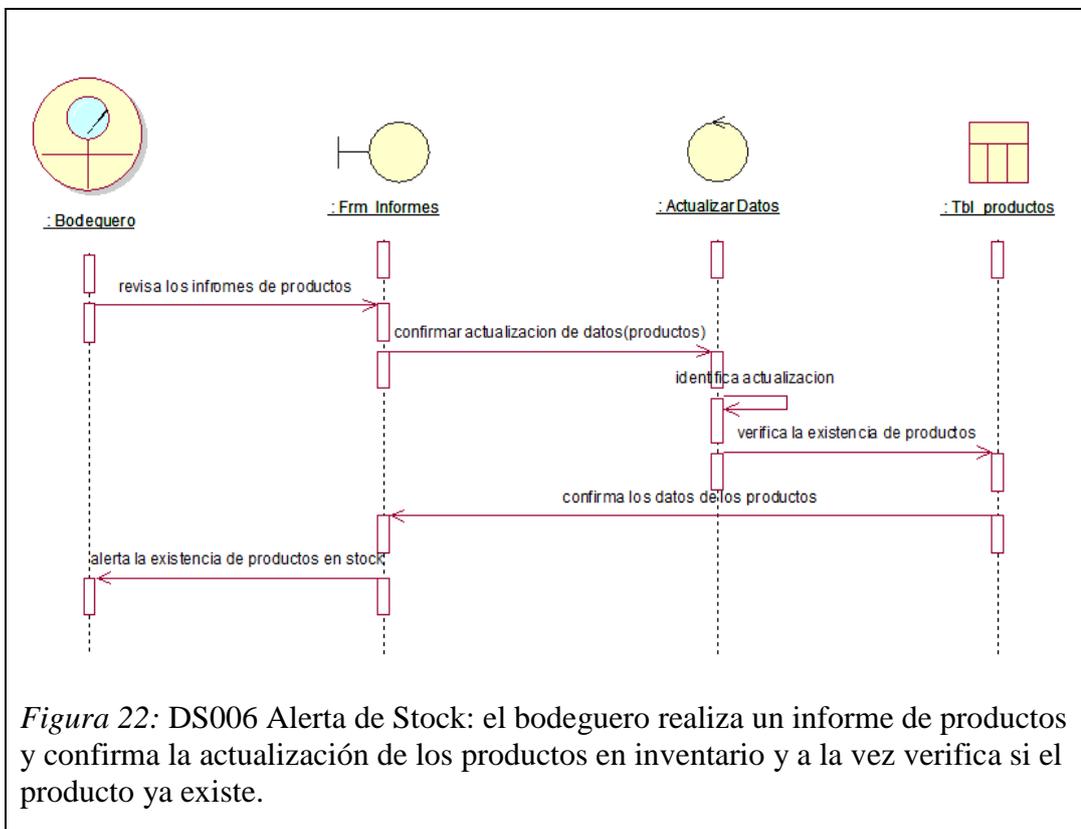


Figura 18: DS004 Control de inventario: el Bodeguero ingresa al sistema y realizar una actualización de todas las categorías y todos los productos que existen en inventario.

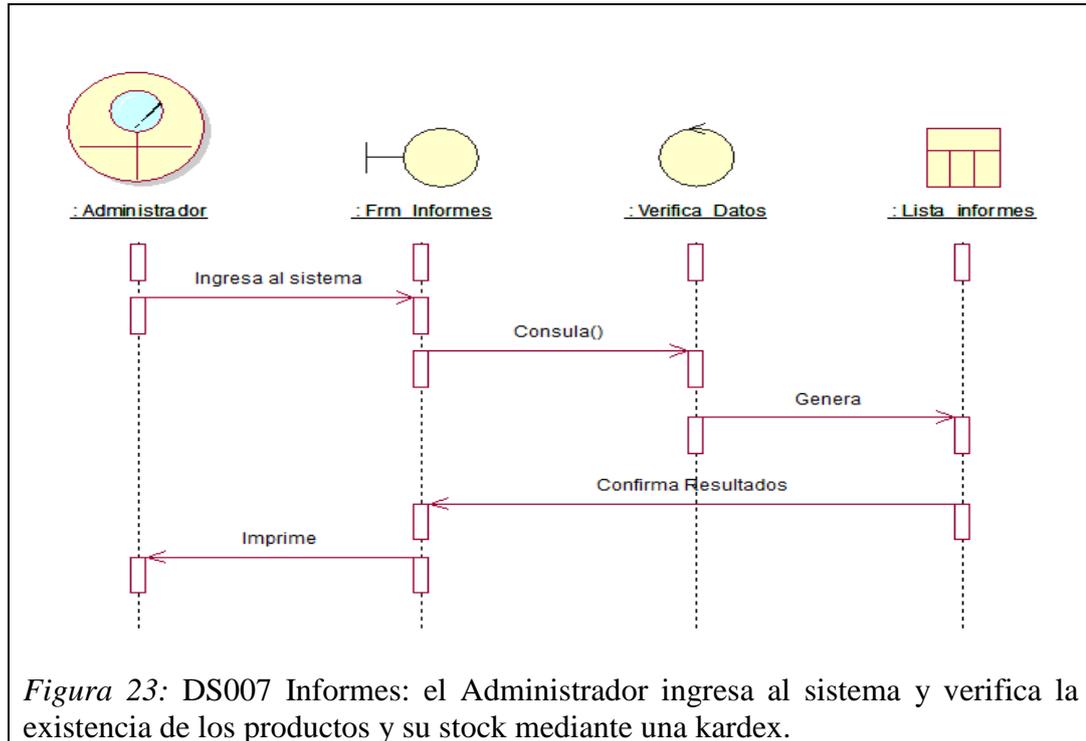
### 5.02.03.05 DS005 Devolución



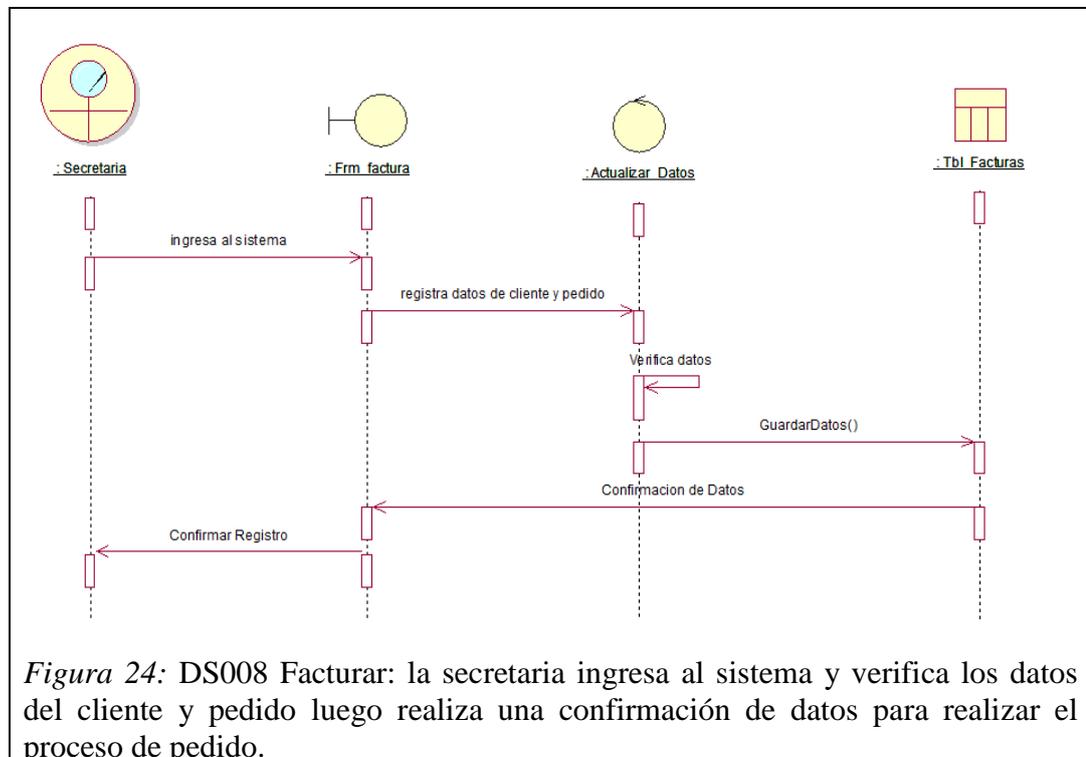
### 5.02.03.06 DS006 Alerta de Stock



### 5.02.03.07 DS007 Informes



### 5.02.03.08 DS008 Facturar



### 5.02.03.09 DS009 Cancelar Factura

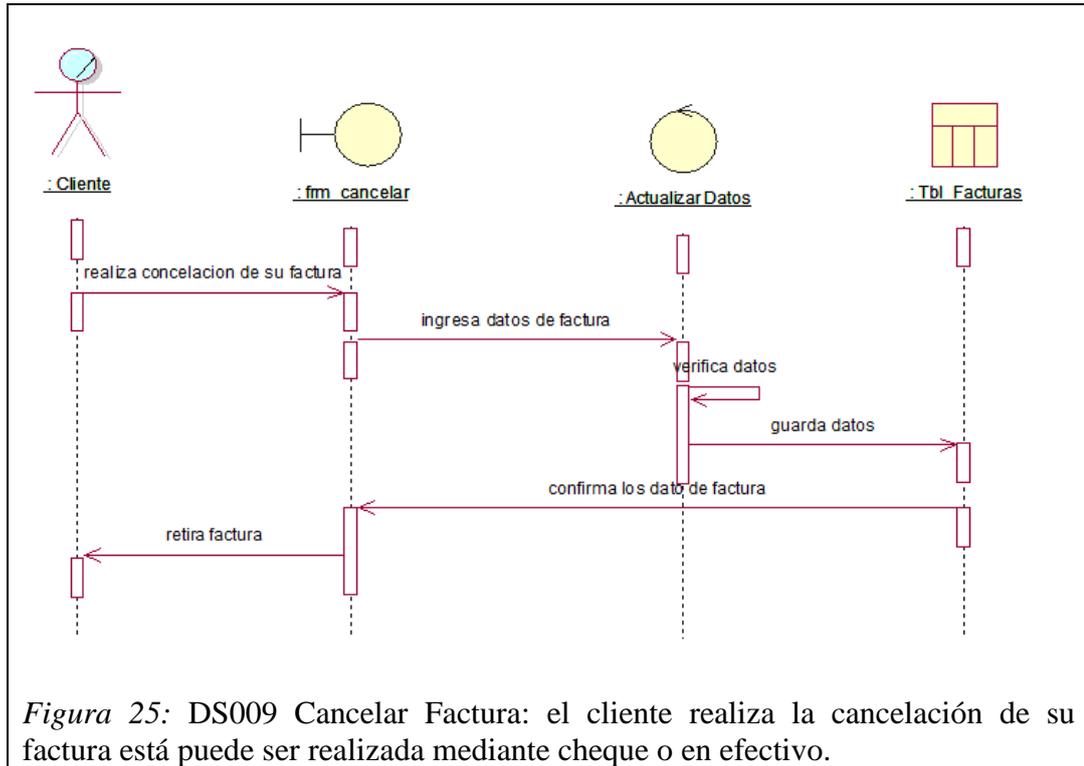


Figura 25: DS009 Cancelar Factura: el cliente realiza la cancelación de su factura está puede ser realizada mediante cheque o en efectivo.

### 5.02.03.10 DS010

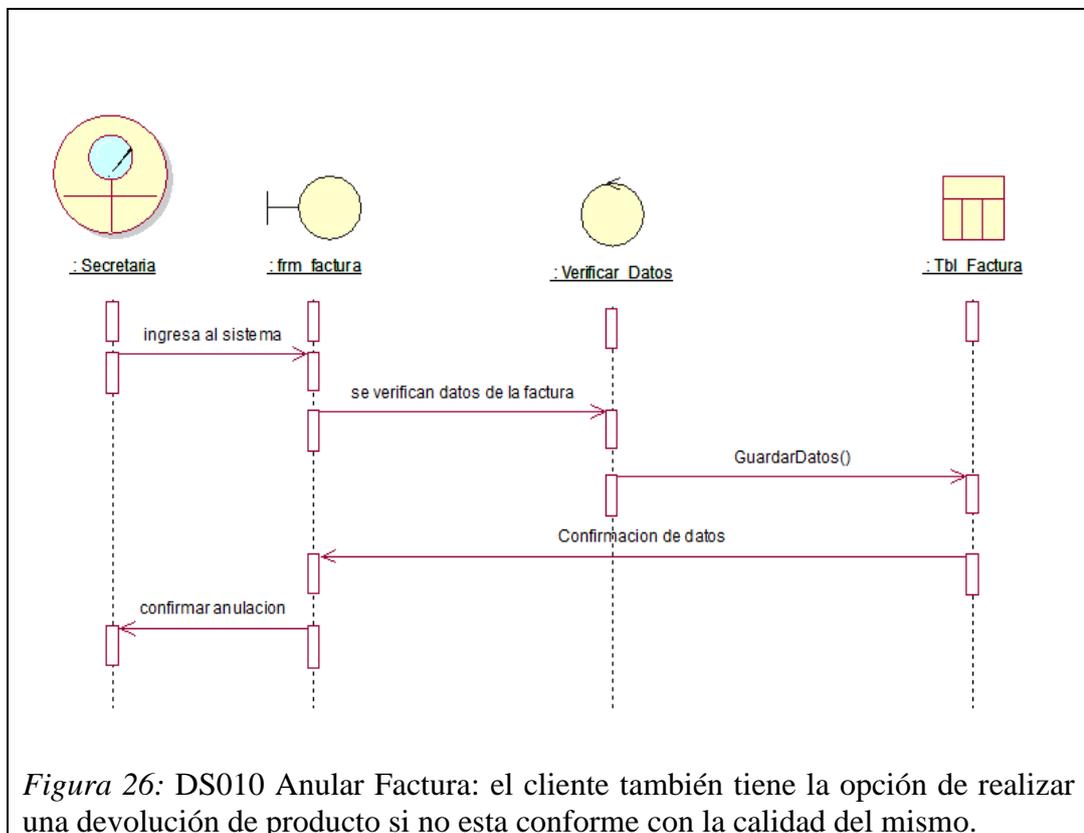
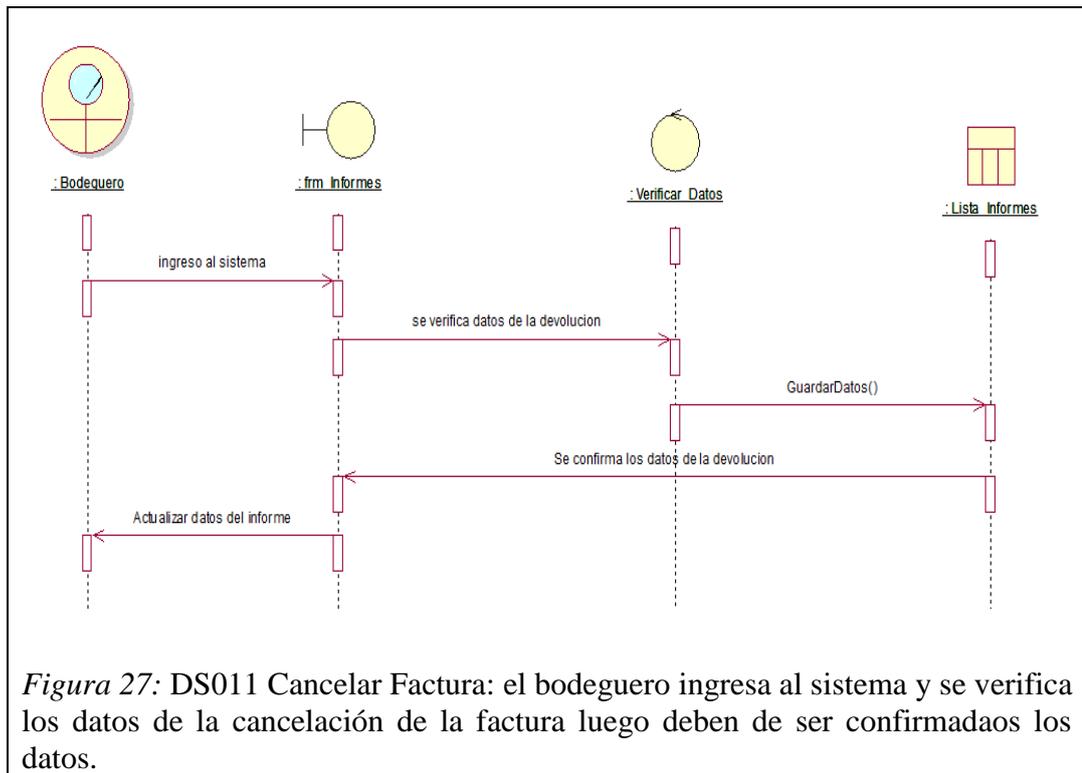


Figura 26: DS010 Anular Factura: el cliente también tiene la opción de realizar una devolución de producto si no esta conforme con la calidad del mismo.

### 5.02.03.11 DS011 Cancelar Factura



#### Descripción de los Diagramas de Secuencia

Los diagramas de secuencia se los realiza para detallar acciones que se pueden realizar para una tarea determinada; en las figuras anteriores se observó las actividades y procesos internos del sistema, es decir, de esta forma podemos analizar que el inicio de los procesos comienza con el cliente ingresando al sistema, donde podrá revisar catálogo los productos y una vez dentro del sistema podrá realizar un pedido, podrá ver la orden de compra en la que podrá modificar la cantidad de productos que desea comprar,

Luego se eso la secretaria estará dentro del sistema para procesar la orden de compra del cliente y así poder facturar su orden de compra.

### 5.02.04 Diagrama de Colaboración

#### 5.02.04.01 DC001 Consulta de Producto

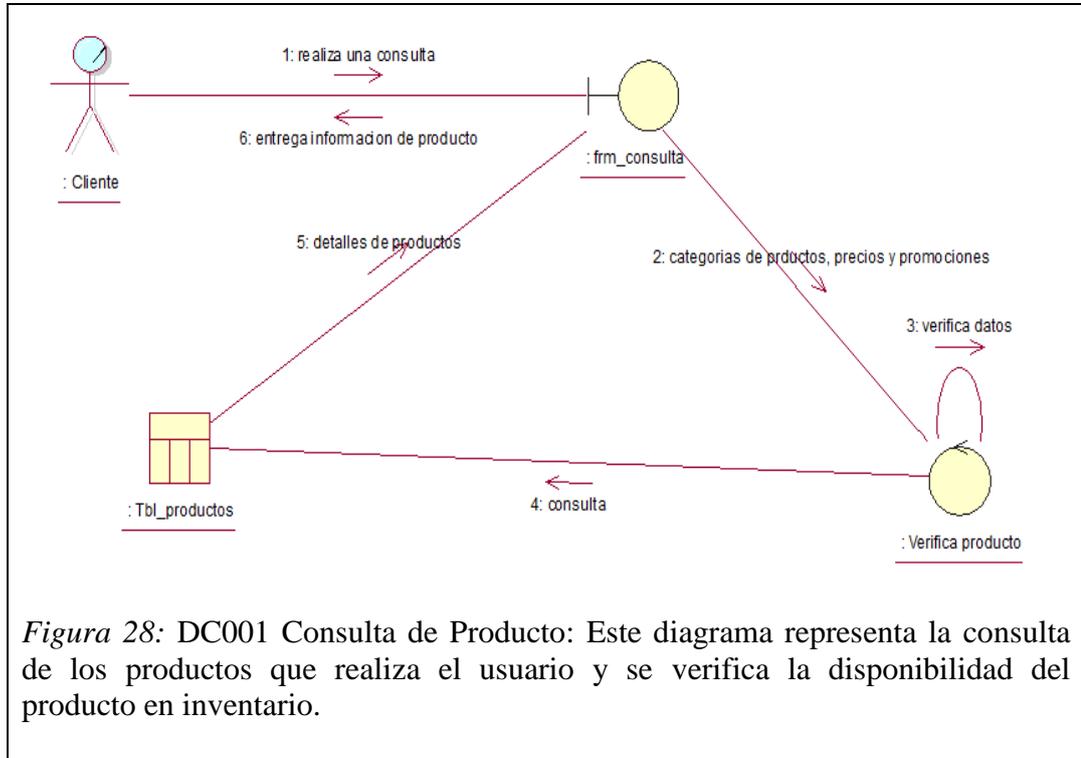


Figura 28: DC001 Consulta de Producto: Este diagrama representa la consulta de los productos que realiza el usuario y se verifica la disponibilidad del producto en inventario.

#### 5.02.03.02 DC002

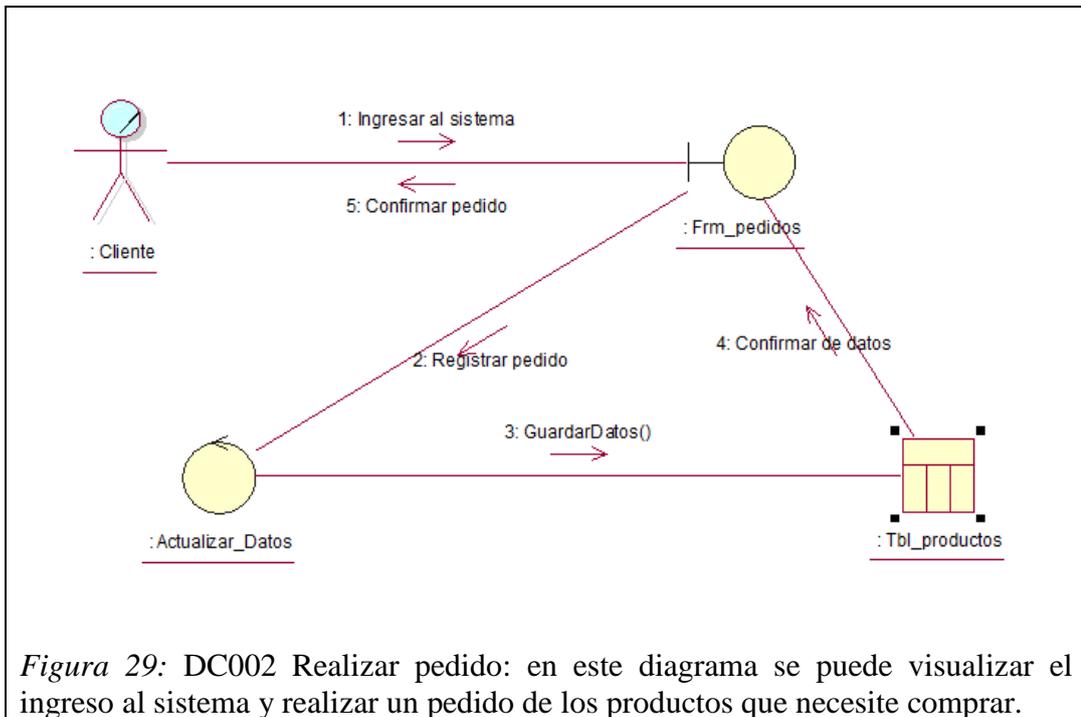


Figura 29: DC002 Realizar pedido: en este diagrama se puede visualizar el ingreso al sistema y realizar un pedido de los productos que necesite comprar.

### 5.02.03.03 DC003 Catalogo de Producto

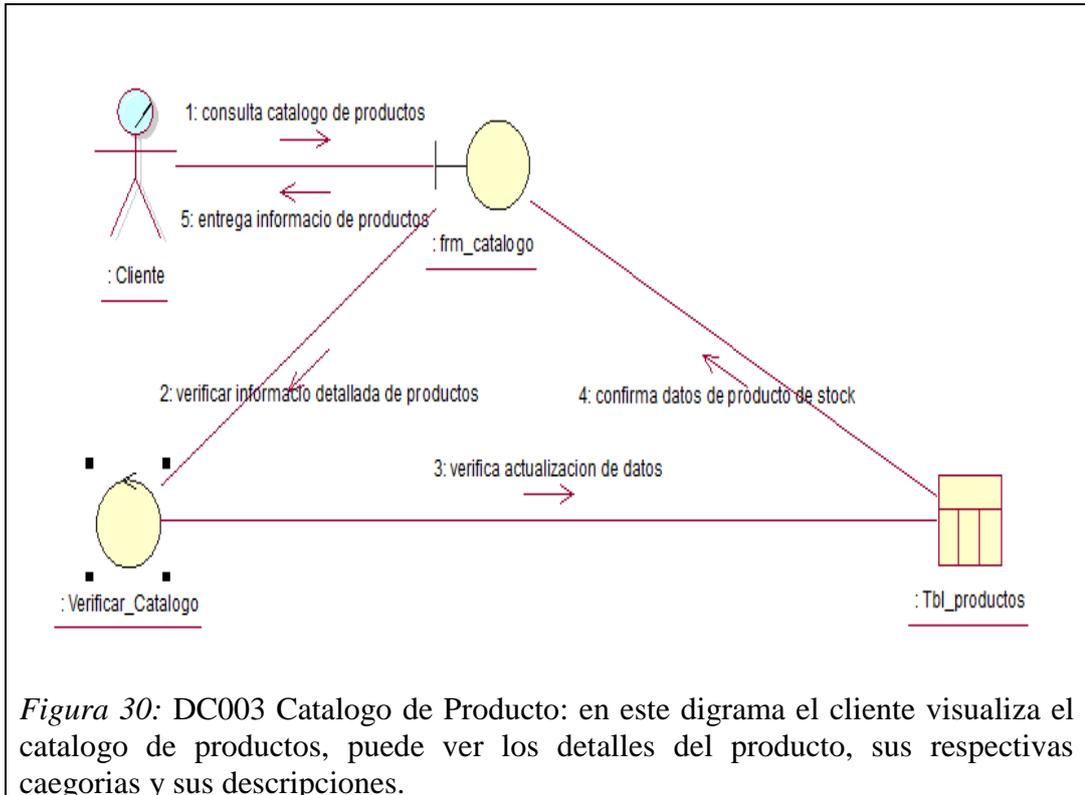


Figura 30: DC003 Catalogo de Producto: en este digrama el cliente visualiza el catalogo de productos, puede ver los detalles del producto, sus respectivas caegorias y sus descripciones.

### 5.02.03.04 DC004 Control de inventario

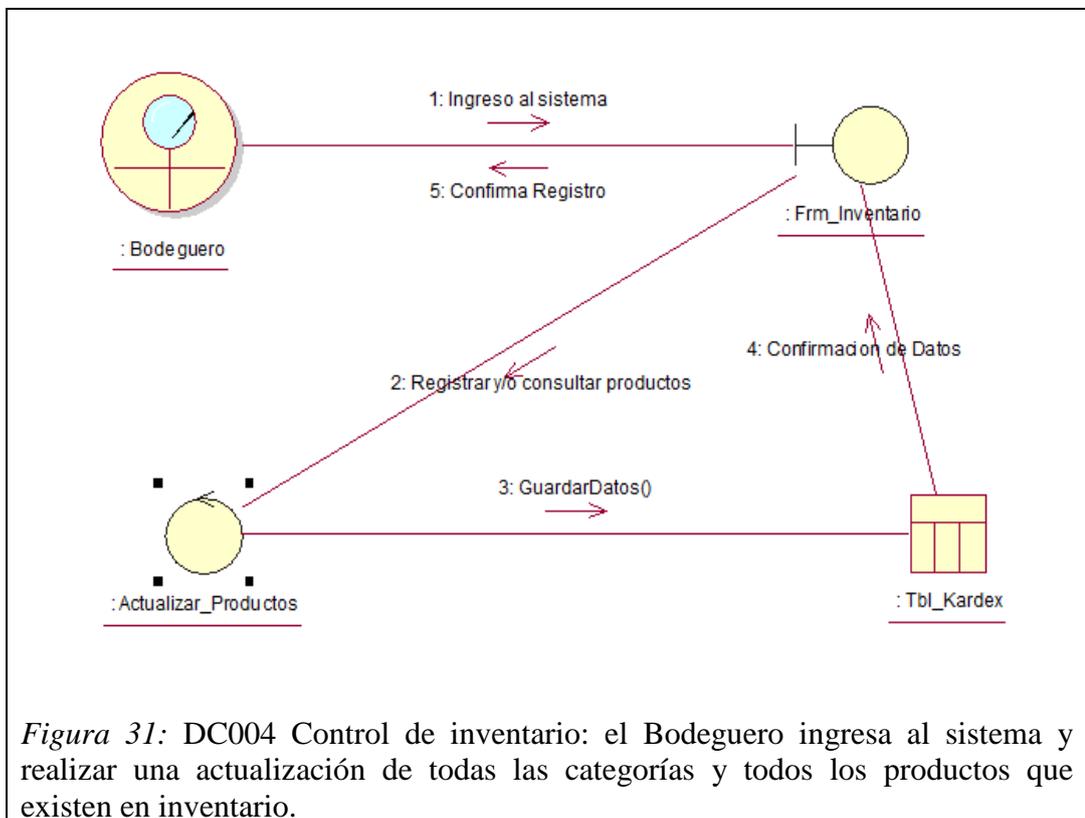


Figura 31: DC004 Control de inventario: el Bodeguero ingresa al sistema y realizar una actualización de todas las categorías y todos los productos que existen en inventario.

### 5.02.03.05 DC005 Devolución

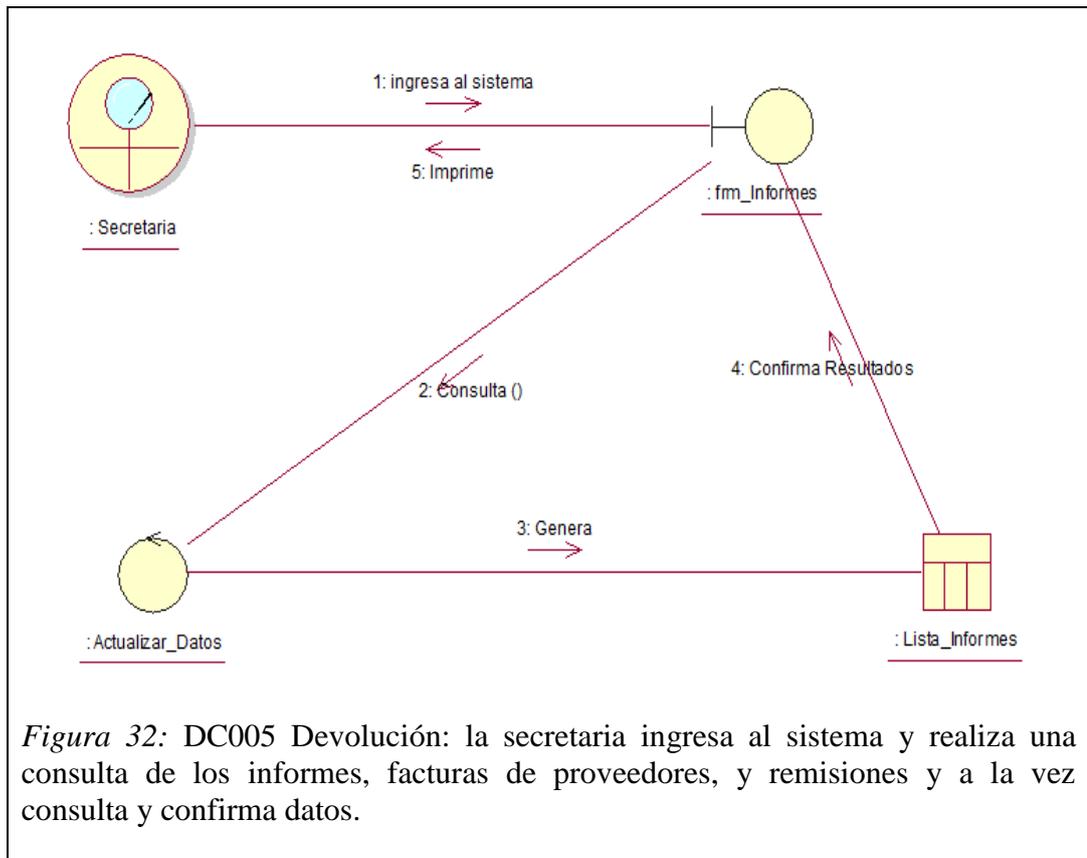


Figura 32: DC005 Devolución: la secretaria ingresa al sistema y realiza una consulta de los informes, facturas de proveedores, y remisiones y a la vez consulta y confirma datos.

### 5.02.03.06 DC006 Alerta de Stock

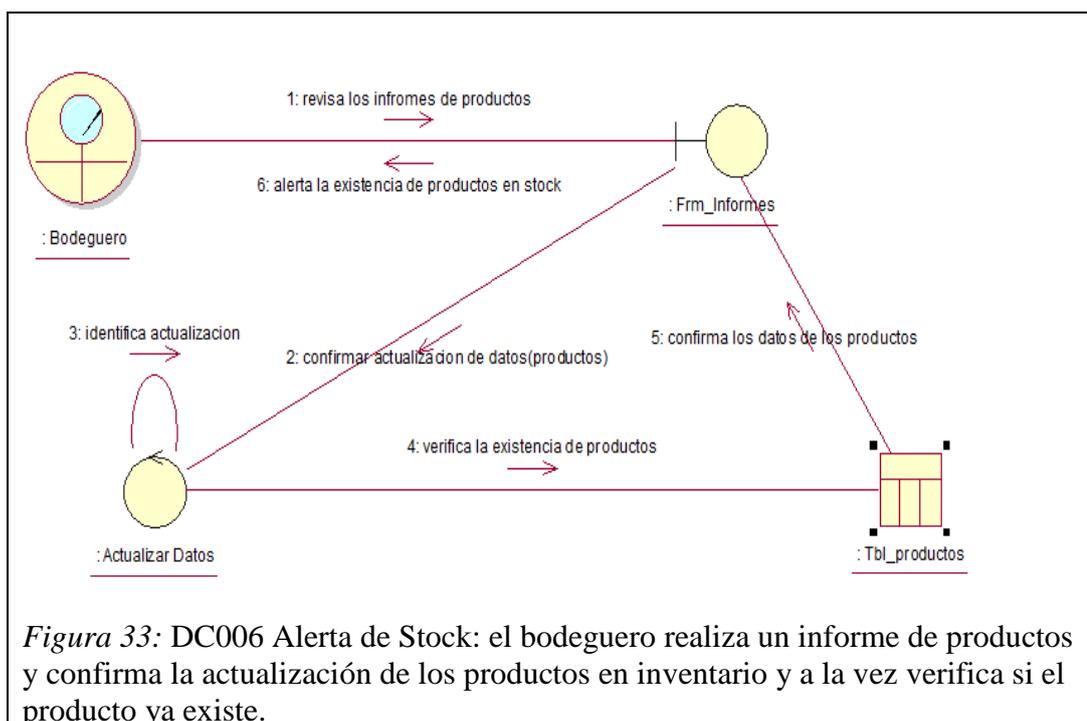


Figura 33: DC006 Alerta de Stock: el bodeguero realiza un informe de productos y confirma la actualización de los productos en inventario y a la vez verifica si el producto ya existe.

### 5.02.03.07 DC007 Informes

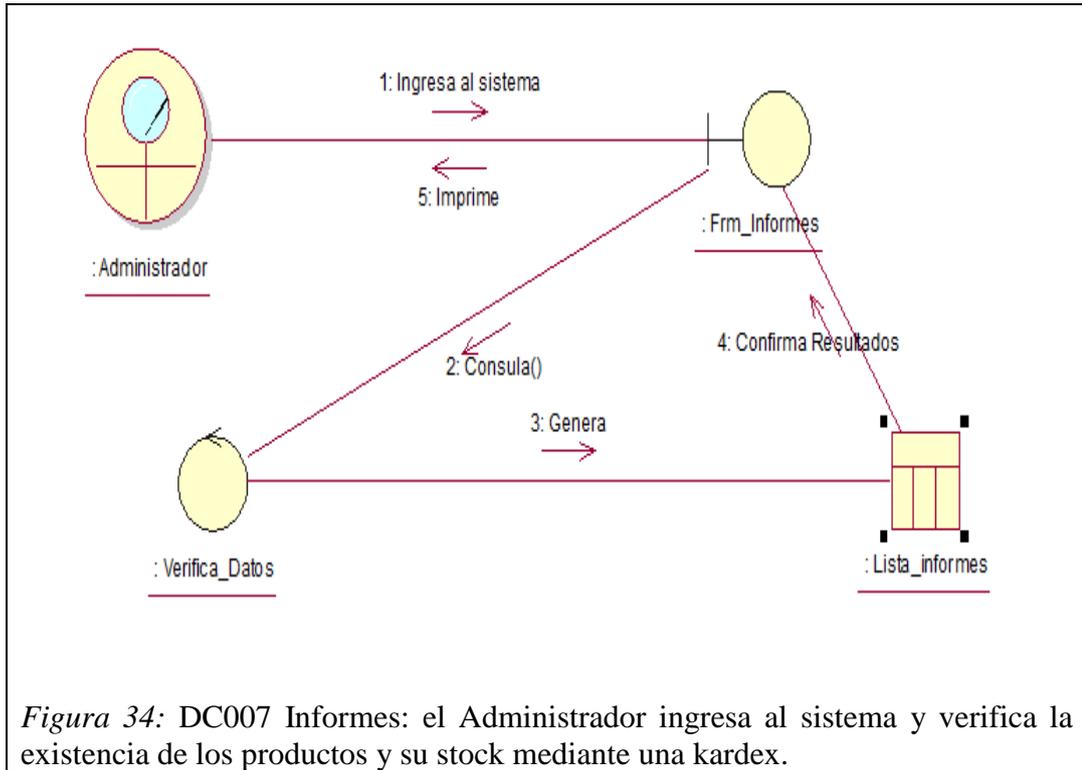


Figura 34: DC007 Informes: el Administrador ingresa al sistema y verifica la existencia de los productos y su stock mediante una kardex.

### 5.02.03.08 DC008 Facturar

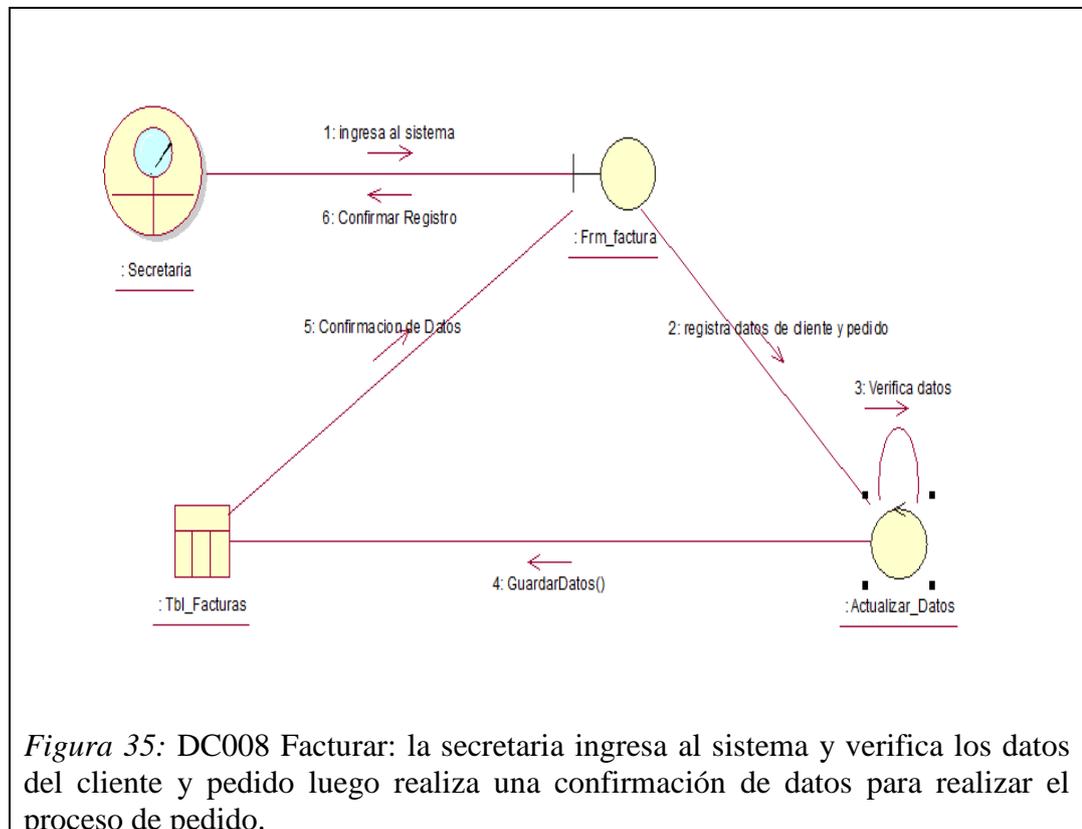


Figura 35: DC008 Facturar: la secretaria ingresa al sistema y verifica los datos del cliente y pedido luego realiza una confirmación de datos para realizar el proceso de pedido.

### 5.02.03.09 DC009 Cancelar Factura

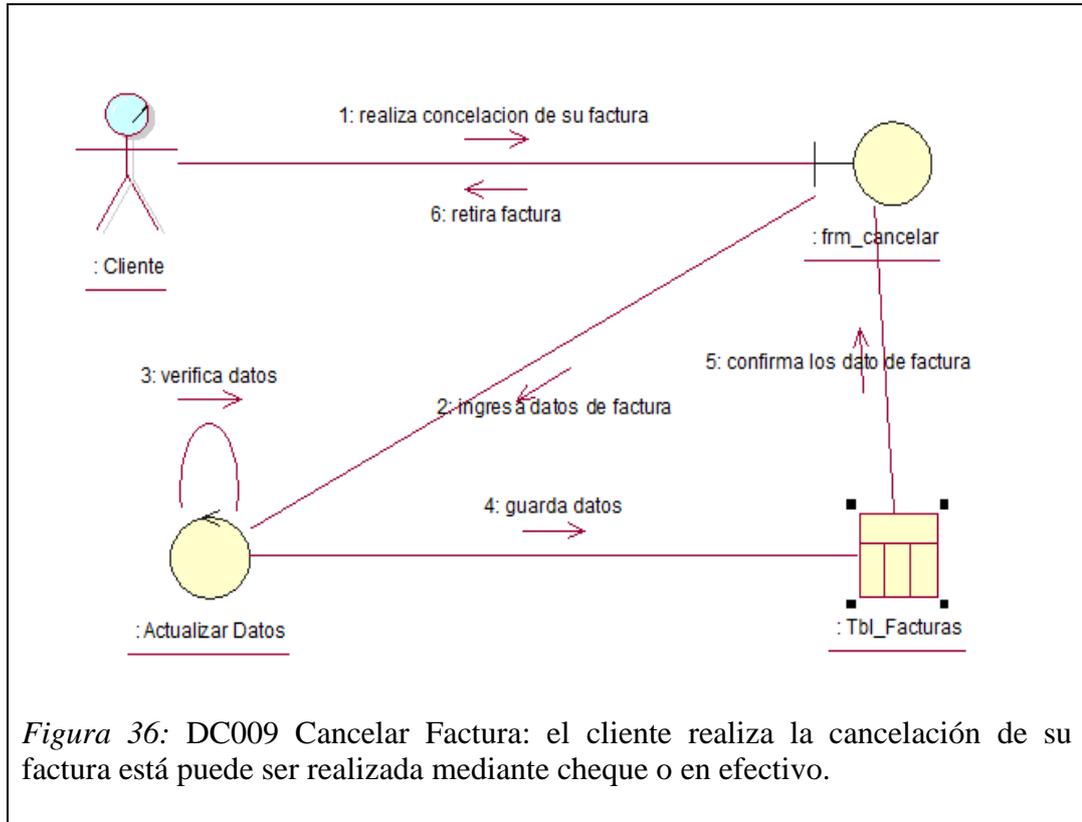


Figura 36: DC009 Cancelar Factura: el cliente realiza la cancelación de su factura está puede ser realizada mediante cheque o en efectivo.

### 5.02.03.10 DC010 Anular Factura

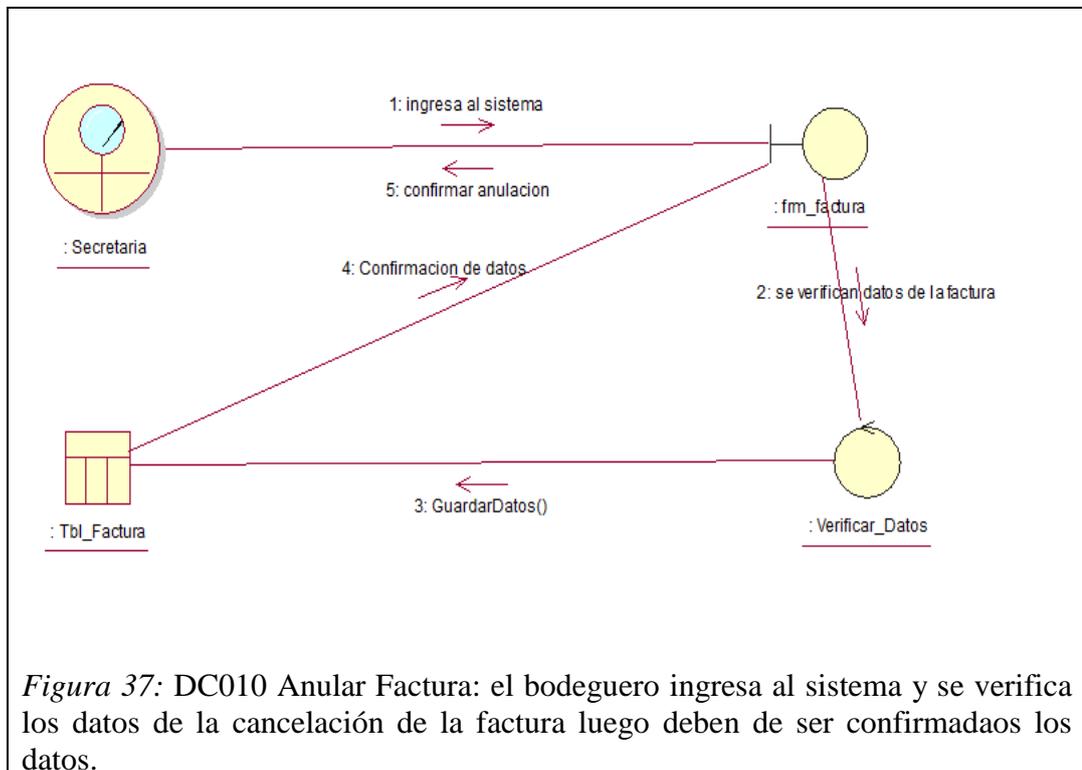
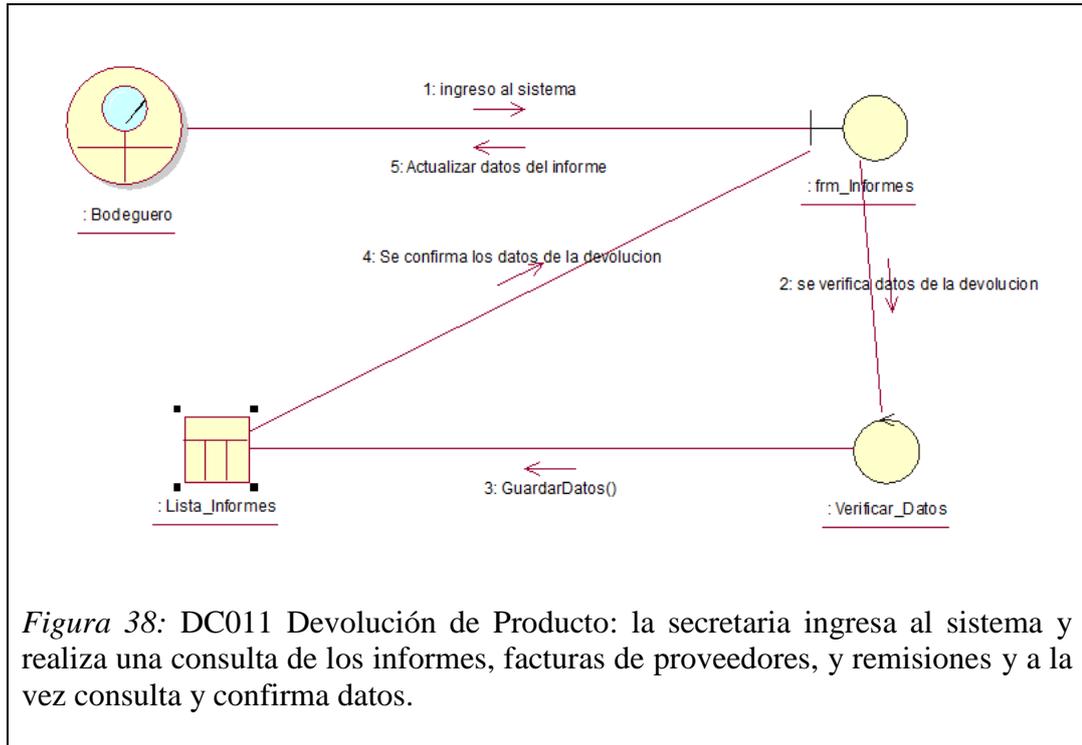


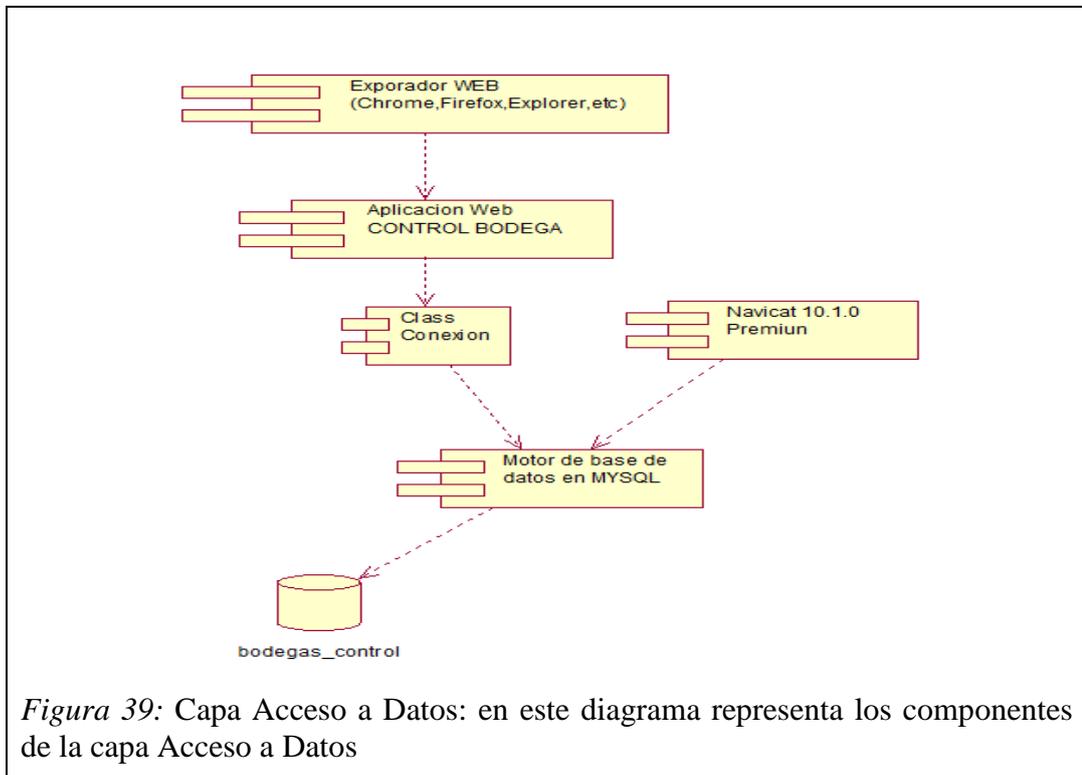
Figura 37: DC010 Anular Factura: el bodeguero ingresa al sistema y se verifica los datos de la cancelación de la factura luego deben de ser confirmadaos los datos.

### 5.02.03.11 DC011



### 5.02.05 Diagrama de Componentes

#### 5.02.05.01 Capa Acceso a Datos



### 5.02.05.02 Capa Lógica del Negocio

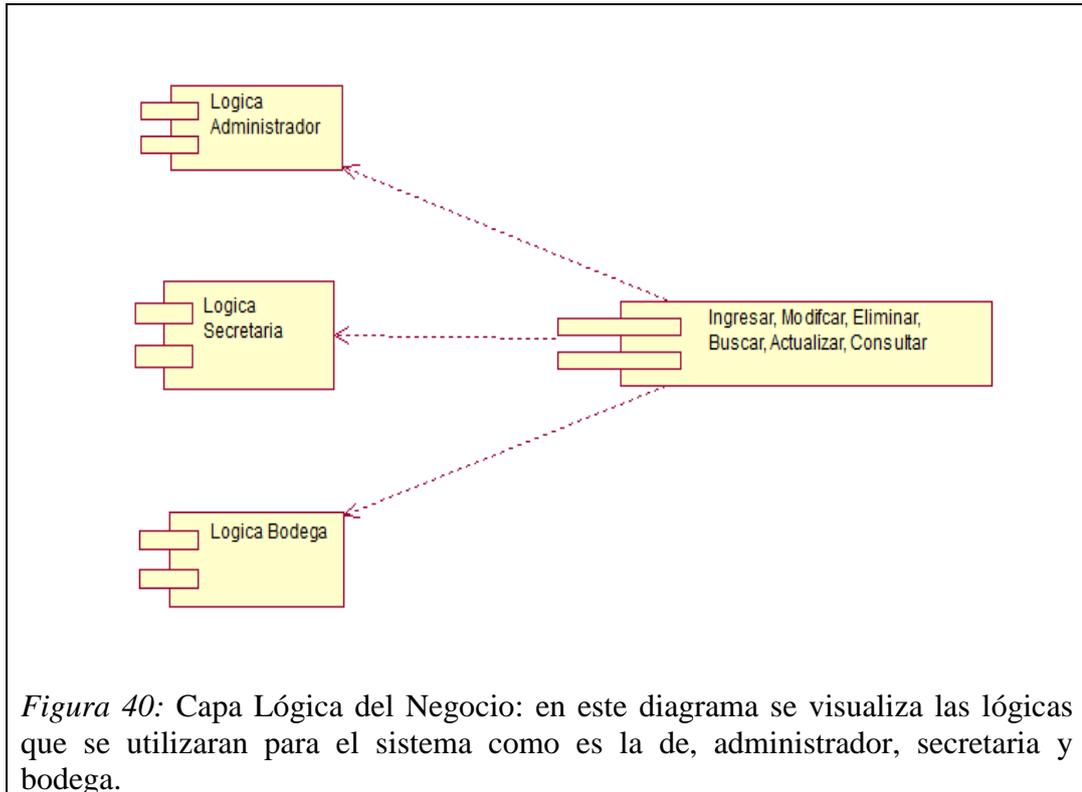


Figura 40: Capa Lógica del Negocio: en este diagrama se visualiza las lógicas que se utilizarán para el sistema como es la de, administrador, secretaria y bodega.

### 5.02.05.03 Capa Presentación

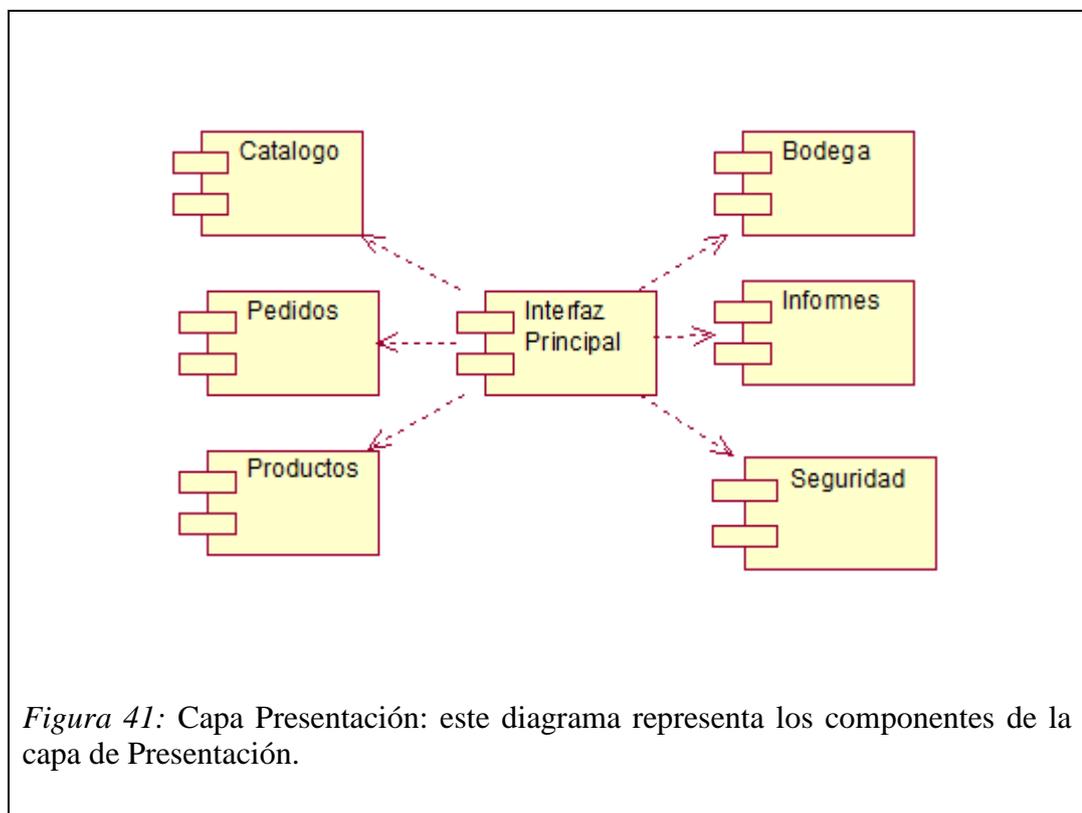


Figura 41: Capa Presentación: este diagrama representa los componentes de la capa de Presentación.

### 5.02.06 Diagrama de Clases

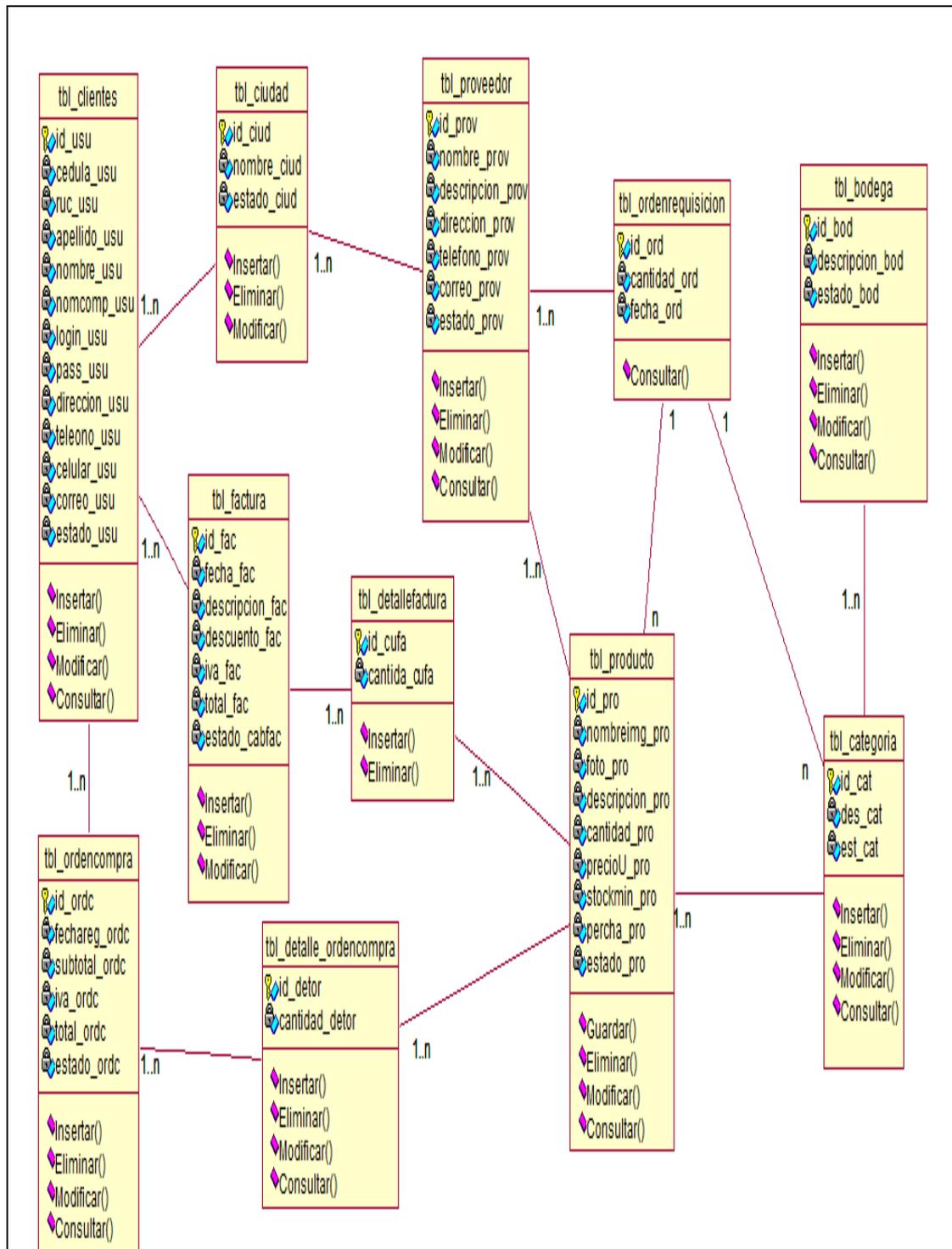


Figura 42: Diagrama de Clases: en el diagrama de Clases se representa la estructura del sistema que se va a utilizar (la base de datos del sistema con sus tablas, atributos y procesos).

### 5.02.07 Diagrama Lógico

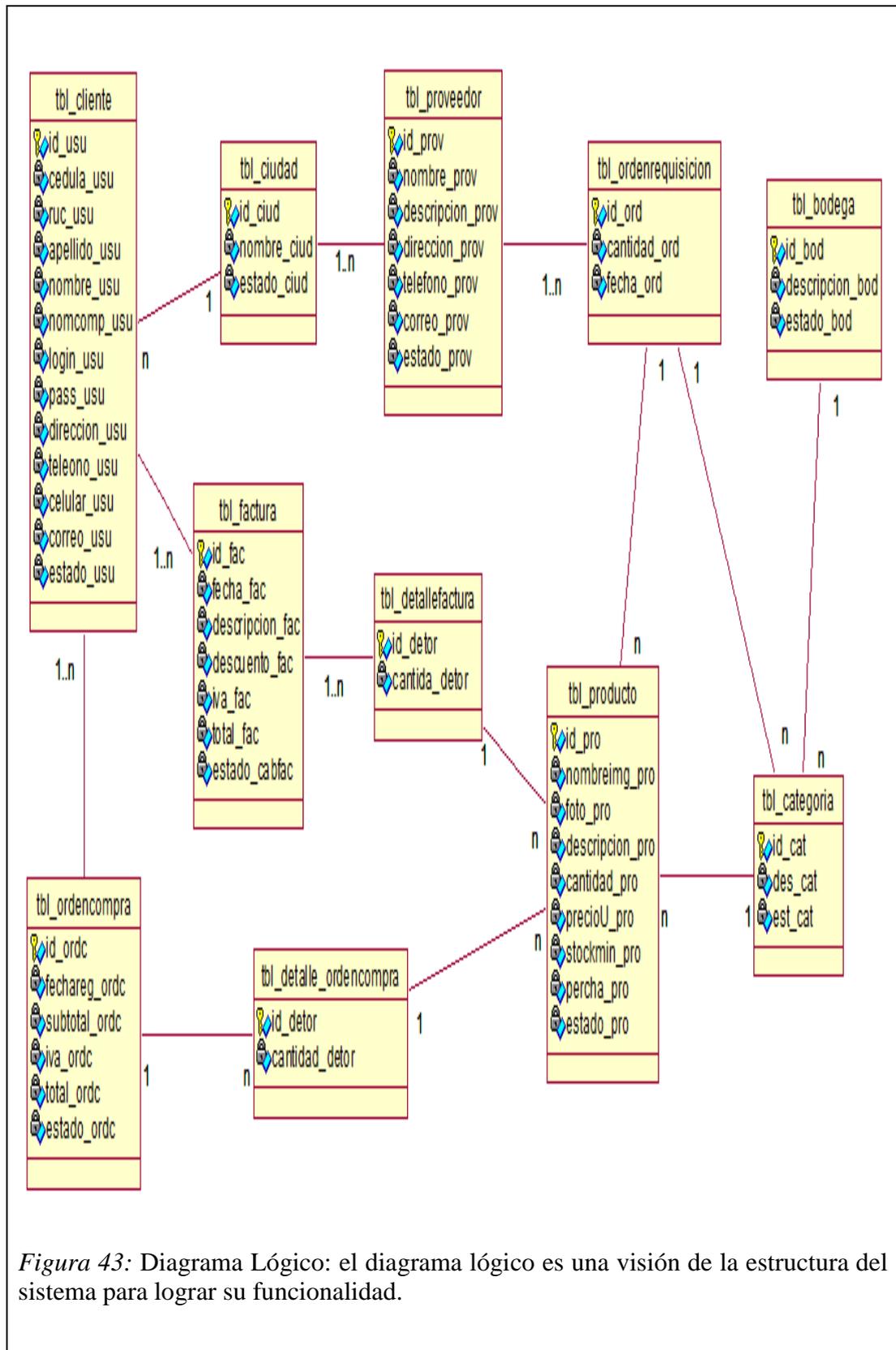


Figura 43: Diagrama Lógico: el diagrama lógico es una visión de la estructura del sistema para lograr su funcionalidad.

### 5.02.07 Diagrama Físico

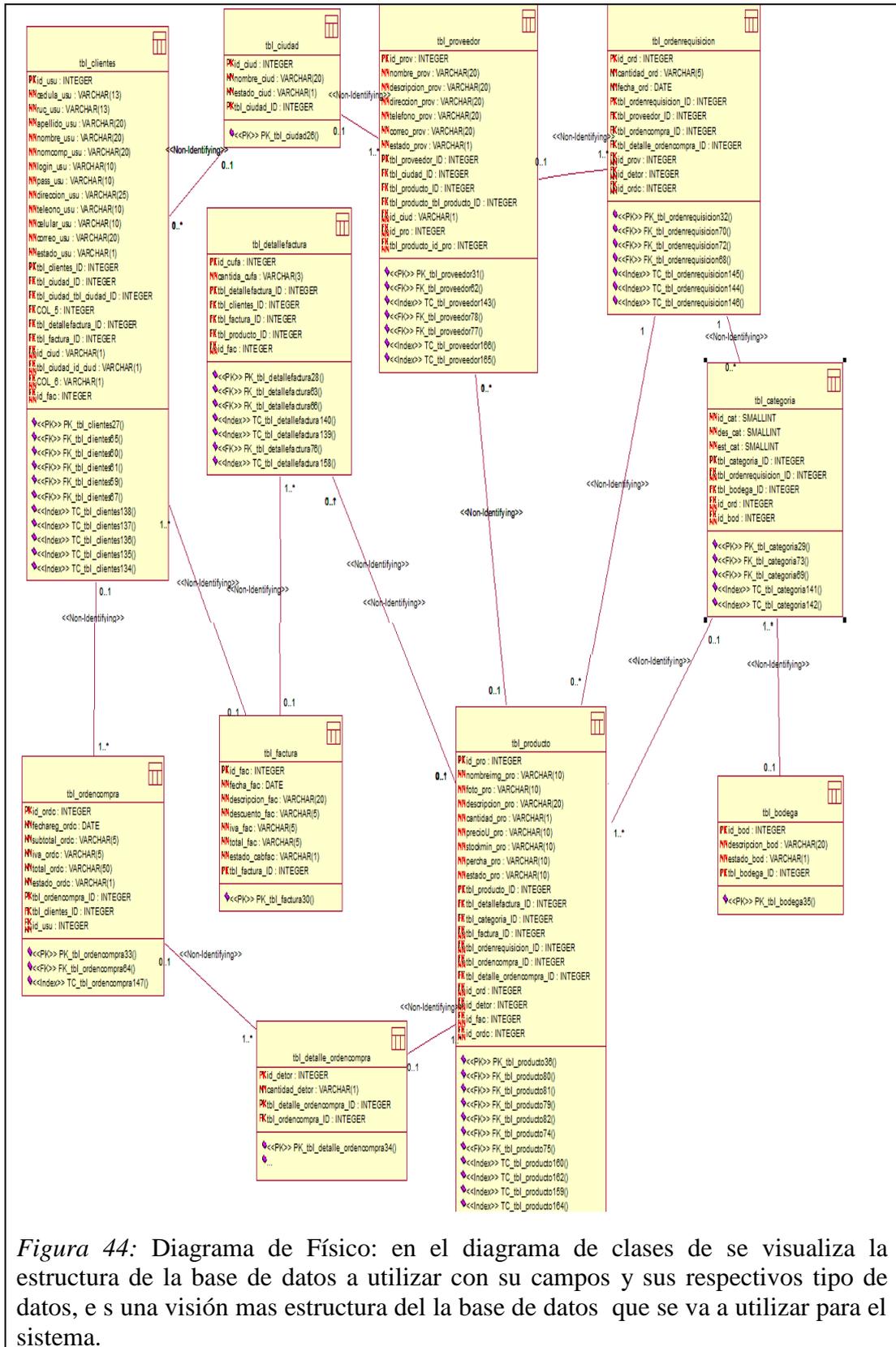


Figura 44: Diagrama de Físico: en el diagrama de clases de se visualiza la estructura de la base de datos a utilizar con su campos y sus respectivos tipo de datos, e s una visión mas estructura del la base de datos que se va a utilizar para el sistema.

## **5.03 Desarrollo**

### **5.03.01 Arquitectura del Sistema**

#### **5.03.01.01 Capa de Presentación**

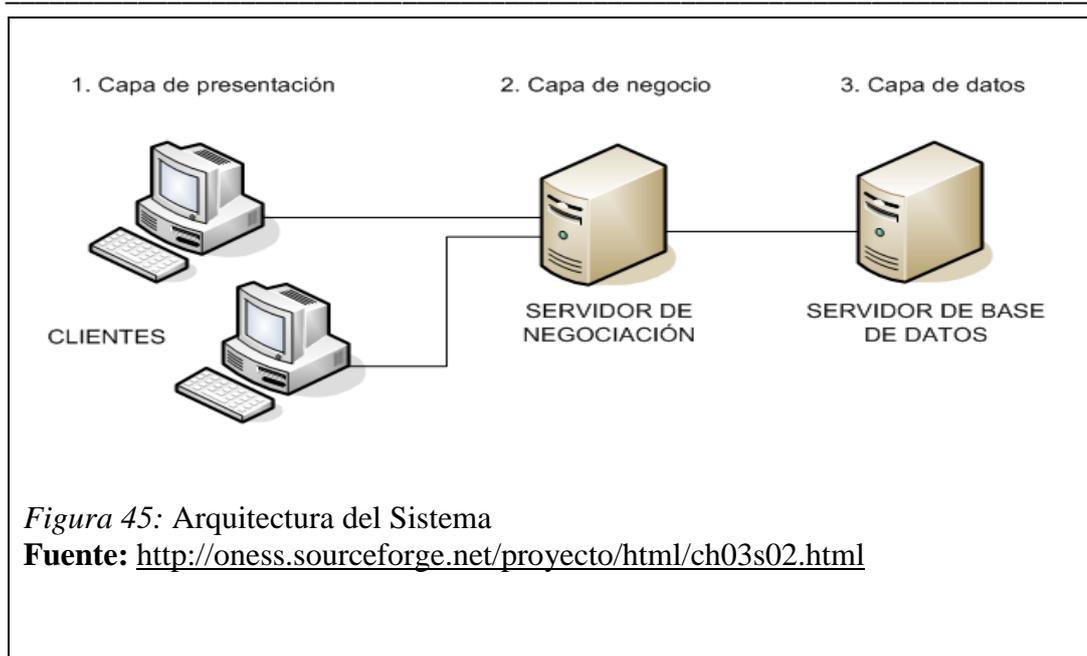
Es la que ve el usuario (también se la denomina "capa de usuario"), presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). También es conocida como interfaz gráfica y debe tener la característica de ser "amigable" (entendible y fácil de usar) para el usuario. Esta capa se comunica únicamente con la capa de negocio.

#### **5.03.01.02 Capa de Negocio**

Es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos almacenar o recuperar datos de él. También se consideran aquí los programas de aplicación.

#### **5.03.01.03 Capa de Datos**

Es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.



## MÓDULOS

### Módulo de Seguridad

Permite proteger la información que se procesa en los sistemas a través del control de acceso y las diferentes limitaciones para cada tipo de usuario, el único usuario que controlar los privilegios y toda la información es el administrador del sistema.

### Módulo de Mantenimiento

Permite al administrador del sistema agregar, modificar, actualizar y eliminar la información con el propósito de sustentar y crear una persistencia en la información. Los mantenimientos con los que cuenta el sistema son a clientes, proveedores, productos, bodega y empresa así como también las modificaciones de los datos de los turnos facturas y consultas.

### Módulo Lógica-Negocios

Permite crear las validaciones de los procesos que realiza el sistema con la finalidad de que exista integridad en la información.

### 5.03.02 Lista de Comandos Html

**Tabla 3**

*Listado de Comandos HTML*

| DETALLE DEL COMANDO               | ABRE  | CIERRA                             |
|-----------------------------------|---|------------------------------------|
| <b>Principio de Documento</b>     | .. <b>&lt;HTML&gt;</b> ...                        | <b>&lt;/HTML&gt;</b>               |
| <b>Encabezado y Título</b>        | <b>&lt;head&gt;&lt;title&gt;</b> (titulo archivo) | <b>&lt;/title&gt;&lt;/head&gt;</b> |
| <b>Color de Fondo de Página</b>   | <b>&lt;body bgcolor="#RRVVAA"&gt;</b>             | ----                               |
| <b>Imagen de Fondo</b>            | <b>&lt;body background="*.gif (o jpg)"&gt;</b>    | ----                               |
| <b>Imagen Individual</b>          | <b>&lt;img src="YYY.gif (o jpg)"&gt;</b>          | ----                               |
| <b>Imágen (Ancho y Alto)</b>      | <b>&lt;img src=".." width="n" height="n"&gt;</b>  | -----                              |
| <b>Espacio "libre"</b>            | <b>&lt;img src=".." hspace="n"&gt;</b>            | -----                              |
| <b>Color del Texto (parcial)</b>  | <b>&lt;font color="#RRVVAA"&gt;</b>               | <b>&lt;/font&gt;</b>               |
| <b>Tamaño del Texto (parcial)</b> | <b>&lt;font size="n"&gt;</b>                      | <b>&lt;/font&gt;</b>               |
| <b>Punto y a Parte</b>            | <b>&lt;br&gt;</b>                                 | -----                              |
| <b>Línea (s) en blanco</b>        | <b>&lt;p&gt;</b>                                  | -----                              |
| <b>"As you See"</b>               | <b>&lt;pre&gt;</b>                                | <b>&lt;/pre&gt;</b>                |

|  |   |  |
|--|---|--|
| <b>Línea Embebida</b>                  | <code>&lt;hr="n"&gt;</code>   | -----  |
| <b>Negrita*Cursiva*Subra<br/>y</b>     | <code>&lt;b&gt; * &lt;I&gt; * &lt;U&gt;</code>                          | <code>&lt;/b&gt;* &lt;/I&gt;* &lt;/U&gt;</code>                  |
| <b>Subíndice * Superíndice</b>         | <code>&lt;sub&gt; * &lt;sup&gt;</code>                                  | <code>&lt;/sub&gt; * &lt;/sup&gt;</code>                         |
| <b>Centrado*Izquierda*De<br/>recha</b> | <code>&lt;center&gt;* &lt;left&gt;* &lt;right&gt;</code>                | <code>&lt;/center&gt;* &lt;/left<br/>&gt;* &lt;/right&gt;</code> |
| <b>Enlace "lejano"</b>                 | <code>&lt;a href="http://..."&gt;</code>                                | <code>....&lt;/a&gt;</code>                                      |
| <b>Enlace "ab.html" en tu<br/>PC</b>   | <code>&lt;a href="ab.html"&gt;</code>                                   | <code>....&lt;/a&gt;</code>                                      |
| <b>Enlace misma página</b>             | <code>&lt;a href="#pepe"&gt;</code>                                     | <code>....&lt;/a&gt;</code>                                      |
| <b>Enlace Correo</b>                   | <code>&lt;a href="mailto: dir.mail..."&gt;</code>                       | <code>...&lt;/a&gt;</code>                                       |
| <b>Combo</b>                           | <code>&lt;select&gt;&lt;option&gt;Seleccione&lt;/op<br/>tion&gt;</code> | <code>&lt;/select&gt;</code>                                     |
| <b>Tabla</b>                           | <code>&lt;table&gt;&lt;tr&gt;&lt;td&gt;&lt;/td&gt;&lt;/tr&gt;</code>    | <code>&lt;/table&gt;</code>                                      |

*Nota:* Describe una Lista de Comandos html

### 5.03.03 Estándares de Base de Datos

**Tabla 4**

*Estándares de una Base de Datos*

| <b>Tipo de Datos</b>     | <b>Característica</b>   |
|--------------------------|---|
| <b>TinyInt</b>           | Es un número entero con o sin signo. Con signo el rango de valores válidos va desde -128 a 127. Sin signo, el rango de valores es de 0 a 255  |
| <b>Bit ó Bool</b>        | Un número entero que puede ser 0 ó 1  |
| <b>SmallInt</b>          | Número entero con o sin signo. Con signo el rango de valores va desde -32768 a 32767. Sin signo, el rango de valores es de 0 a 65535.   |
| <b>Integer, Int</b>      | Número entero con o sin signo. Con signo el rango de valores va desde -2147483648 a 2147483647. Sin signo el rango va desde 0 a 429.4967.295  |
| <b>Float</b>             | Número pequeño en coma flotante de precisión simple. Los valores válidos van desde -3.402823466E+38 a -1.175494351E-38, 0 y desde 1.175494351E-38 a 3.402823466E+38.                            |
| <b>xReal,<br/>Double</b> | Número en coma flotante de precisión doble. Los valores permitidos van desde -1.7976931348623157E+308 a -2.2250738585072014E-308.   |
| <b>Date</b>              | Tipo fecha, almacena una fecha. El rango de valores va desde el 1 de enero del 1001 al 31 de diciembre de 9999. El formato de almacenamiento es de año-mes-día                                  |
| <b>DateTime</b>          | Combinación de fecha y hora. El rango de valores va desde el 1 de enero del 1001 a las 0 horas, 0 minutos y 0 segundos al 31 de diciembre del 9999 a las 23 horas, 59 minutos y 59 segundos. El |

|            |   |
|------------|---|
|            | formato de almacenamiento es de año-mes-día<br>horas:minutos:segundos   |
| Time       | Almacena una hora. El rango de horas va desde -838 horas, 59 minutos y 59 segundos a 838, 59 minutos y 59 segundos. El formato de almacenamiento es de 'HH:MM:SS' |
| Char(n)    | Almacena una cadena de longitud fija. La cadena podrá contener desde 0 a 255 caracteres.  |
| VarChar(n) | Almacena una cadena de longitud variable. La cadena podrá contener desde 0 a 255 caracteres.  |

*Nota:* Describe estándares de una base de datos

### 5.03.04 Estándares de Diseño UML

#### Diagrama de Clases

Un diagrama de clases sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia y de uso.

Un diagrama de clases está compuesto por los siguientes elementos:

- Clase: atributos, métodos y visibilidad.
- Relaciones: Herencia, Composición, Agregación, Asociación y Uso.

#### Elementos

**Clase.**-Es la unidad básica que encapsula toda la información de un Objeto (un objeto es una instancia de una clase). A través de ella podemos modelar el entorno en estudio (una Casa, un Auto, una Cuenta Corriente, etc.).

En UML, una clase es representada por un rectángulo que posee tres divisiones:



En donde:

- **Superior:** Contiene el nombre de la Clase
- **Intermedio:** Contiene los atributos (o variables de instancia) que caracterizan a la Clase (pueden ser private, protected o public).
- **Inferior:** Contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno (dependiendo de la visibilidad: private, protected o public).

Atributos y Métodos:

- **Atributos:**

Los atributos o características de una Clase pueden ser de tres tipos, los que definen el grado de comunicación y visibilidad de ellos con el entorno, estos son:

- **public** (+, 
  - **Métodos:**

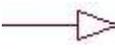
Los métodos u operaciones de una clase son la forma en como ésta interactúa con su entorno, éstos pueden tener las características:

- **public** (+, 
  - **Relaciones entre Clases:**

Ahora ya definido el concepto de Clase, es necesario explicar cómo se pueden interrelacionar dos o más clases (cada uno con características y objetivos diferentes).

Antes es necesario explicar el concepto de cardinalidad de relaciones: En UML, la cardinalidad de las relaciones indica el grado y nivel de dependencia, se anotan en cada extremo de la relación y éstas pueden ser:

- **uno o muchos:** 1..\* (1..n)
- **0 o muchos:** 0..\* (0..n)
- **número fijo:** m (m denota el número).

**Herencia (Especialización/Generalización):** 

Indica que una subclase hereda los métodos y atributos especificados por una Super Clase, por ende la Subclase además de poseer sus propios métodos y atributos, poseerá las características y atributos visibles de la Super Clase (public y protected)

**Agregación:** 

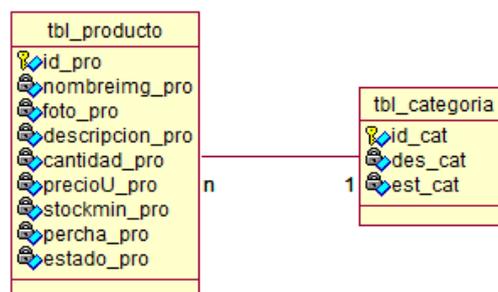
Para modelar objetos complejos, no bastan los tipos de datos básicos que proveen los lenguajes: enteros, reales y secuencias de caracteres. Cuando se requiere componer objetos que son instancias de clases definidas por el desarrollador de la aplicación, tenemos dos posibilidades:

- **Por Valor:** Es un tipo de relación estática, en donde el tiempo de vida del objeto incluido está condicionado por el tiempo de vida del que lo incluye. Este tipo de relación es comúnmente llamada **Composición** (el Objeto base se construye a partir del objeto incluido, es decir, es "parte/todo").
- **Por Referencia:** Es un tipo de relación dinámica, en donde el tiempo de vida del objeto incluido es independiente del que lo incluye. Este tipo de relación es comúnmente llamada **Agregación** (el objeto base utiliza al incluido para su funcionamiento).

**Asociación:** 

La relación entre clases conocida como Asociación, permite asociar objetos que colaboran entre sí. Cabe destacar que no es una relación fuerte, es decir, el tiempo de vida de un objeto no depende del otro.

Ejemplo:



Una categoría puede tener asociados muchos Productos, en cambio un producto solo puede tener asociado una categoría.

**Fuente:** <http://users.dcc.uchile.cl/~psalinas/uml/modelo.html>

## Casos de Uso (Use Case)

### Introducción

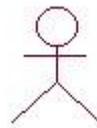
El diagrama de casos de uso representa la forma en como un Cliente (Actor) opera con el sistema en desarrollo, además de la forma, tipo y orden en como los elementos interactúan (operaciones o casos de uso).

Un diagrama de casos de uso consta de los siguientes elementos:

- Actor.
- Casos de Uso.
- Relaciones de Uso, Herencia y Comunicación.

### Elementos

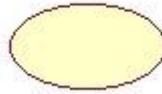
- **Actor:**



Una definición previa, es que un **Actor** es un rol que un usuario juega con respecto al sistema. Es importante destacar el uso de la palabra rol, pues con esto se especifica que un Actor no necesariamente representa a una persona en particular, sino más bien la labor que realiza frente al sistema.

Como ejemplo a la definición anterior, tenemos el caso de un sistema de ventas en que el rol de Vendedor con respecto al sistema puede ser realizado por un Vendedor o bien por el Jefe de Local.

- **Caso de Uso:**



Es una operación/tarea específica que se realiza tras una orden de algún agente externo, sea desde una petición de un actor o bien desde la invocación desde otro caso de uso.

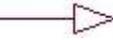
- **Relaciones:**

- **Asociación** 

Es el tipo de relación más básica que indica la invocación desde un actor o caso de uso a otra operación (caso de uso). Dicha relación se denota con una flecha simple.

- **Dependencia o Instanciación** 

Es una forma muy particular de relación entre clases, en la cual una clase depende de otra, es decir, se instancia (se crea). Dicha relación se denota con una flecha punteada.

- **Generalización** 

Este tipo de relación es uno de los más utilizados, cumple una doble función dependiendo de su estereotipo, que puede ser de **Uso** (<<uses>>) o de **Herencia** (<<extends>>).

Este tipo de relación está orientado exclusivamente para casos de uso (y no para actores).

**Extends:** Se recomienda utilizar cuando un caso de uso es similar a otro (características).

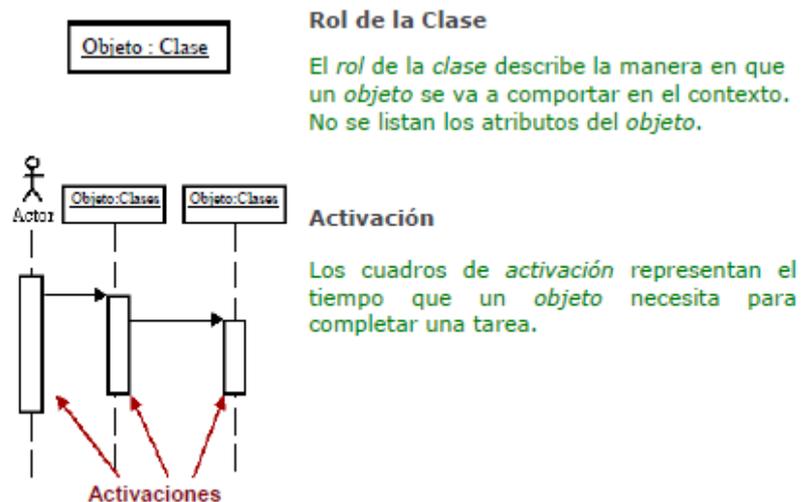
**Uses:** Se recomienda utilizar cuando se tiene un conjunto de características que son similares en más de un caso de uso y no se desea mantener copiada la descripción de la característica.

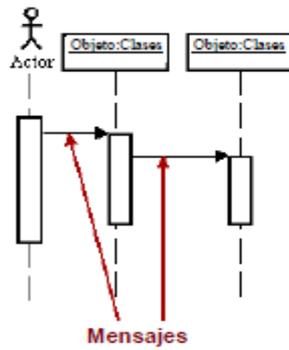
De lo anterior cabe mencionar que tiene el mismo paradigma en diseño y modelamiento de clases, en donde está la duda clásica de **usar** o **heredar**.

**Fuente:** <http://users.dcc.uchile.cl/~psalinas/uml/casosuso.html>

### Diagrama de Secuencias

Los diagramas de clases y los de objetos representan información estática. No obstante, en un sistema funcional, los objetos interactúan entre sí, y tales interacciones suceden con el tiempo. El diagrama de secuencias UML muestra la mecánica de la interacción con base en tiempos.

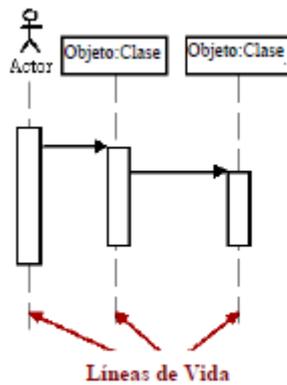




### Mensajes

Los *mensajes* son flechas que representan comunicaciones entre *objetos*. Las medias flechas representan *mensajes* asincrónicos. Los *mensajes* asincrónicos son enviados desde un *objeto* que no va a esperar una respuesta del receptor para continuar con sus tareas.

| Flecha | Tipo de mensaje |
|--------|-----------------|
|        | Simple          |
|        | Sincrónico      |
|        | Asincrónico     |
|        | Rechazado       |
|        | Time out        |



### Líneas de Vida

Las *líneas de vida* son verticales y en línea de puntos, ellas indican la presencia del *objeto* durante el tiempo.

**Fuente:** <http://webbress.com.ar>

### Diagrama de Colaboraciones

Los diagramas de colaboración representan una combinación de información tomada de los diagramas de clases, de secuencias y de casos de uso, describiendo el comportamiento, tanto de la estructura estática, como de la estructura dinámica de un sistema.

Objeto : Clase

### Rol de la Clase

El rol de la clase describe cómo se comporta un objeto. Los atributos del objeto no se listan.

### Rol de las Asociaciones

Los roles de asociación describen cómo se va a comportar una asociación en una situación particular. Se usan líneas simple etiquetadas con un estereotipo\*. (ver al final del documento)

<<global>>

1.4 [condición]  
nombre del mensaje

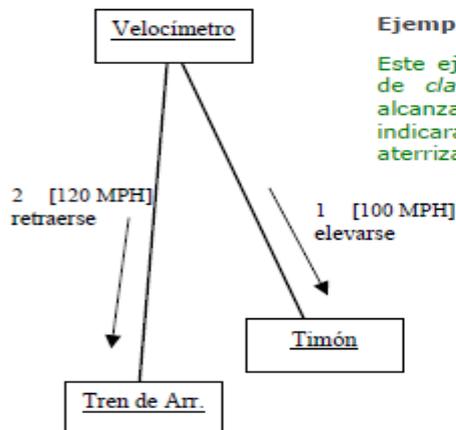


1.4 \* [loop]  
nombre del mensaje



### Mensajes

Contrariamente a los *diagramas de secuencias*, los *diagramas de colaboración* no tienen una manera explícita para denotar el tiempo, por lo que entonces numeran a los mensajes en orden de ejecución. La numeración puede anidarse; por ejemplo, para mensajes anidados al mensaje número 1: 1.1, 1.2, 1.3, etc. . La condición para un mensaje se suele colocar entre corchetes. Para indicar un loop se usa \* después de la numeración.



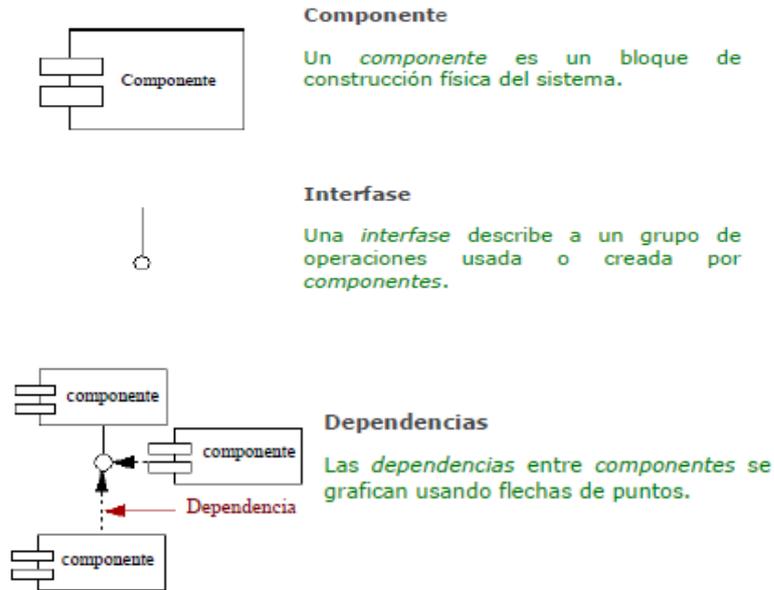
### Ejemplo de Diagrama de Colaboración

Este ejemplo agrega un velocímetro al conjunto de clases que constituyen a un "Avión". Al alcanzar una cierta velocidad el velocímetro indicará al timón que debe elevarse y al tren de aterrizaje que debe retraerse.

Fuente: <http://webbress.com.ar>

## Diagrama de Componentes

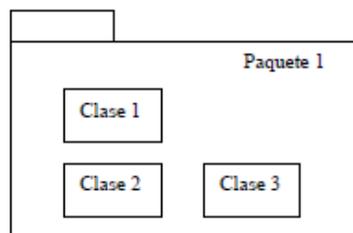
Un diagrama de componentes describe la organización de los componentes físicos de un sistema.



**Fuente:** <http://webbress.com.ar>

### Paquetes en UML

En algunas ocasiones se encontrará con la necesidad de organizar los elementos de un diagrama en un grupo. Tal vez quiera mostrar que ciertas clases o componentes son parte de un subsistema en particular. Para ello, se pueden agrupar en un paquete, que se representa por una carpeta tabular.



**Fuente:** <http://webbress.com.ar>

### 5.03.05 Diseño de Interfaces

#### Sistema de control de pedidos e inventarios

##### Login



Figura 46: Ingreso al Sistema: el usuario realiza un ingreso de datos al login para ingresar al sistema.

Tabla 19

##### Ingreso al Sistema

| Ítem | Representación | Descripción        |
|------|----------------|--------------------|
| A    | Caja de Texto  | Usuario            |
| B    | Caja de Texto  | Contraseña         |
| C    | Botón          | Ingreso al Sistema |

##### Administración de Clientes

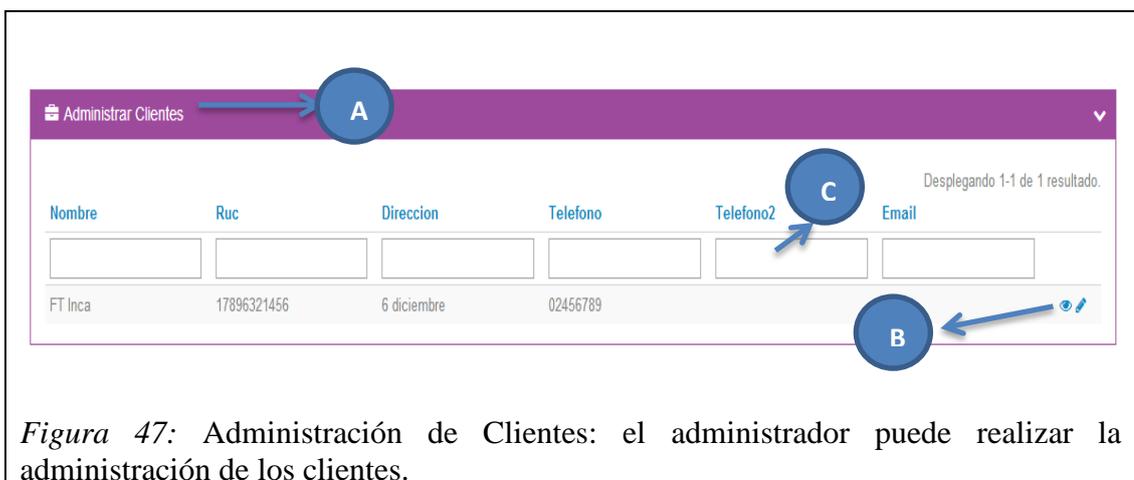


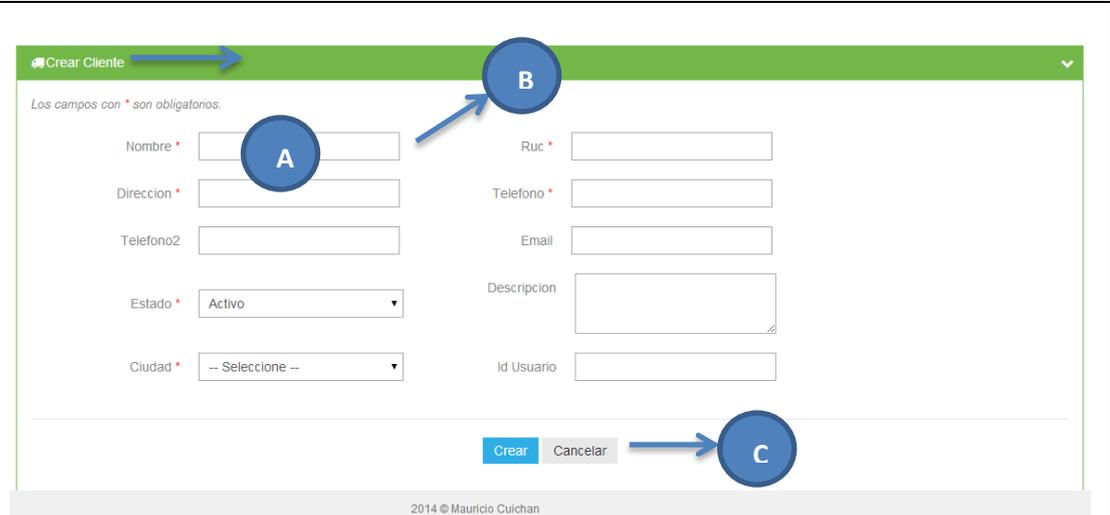
Figura 47: Administración de Clientes: el administrador puede realizar la administración de los clientes.

**Tabla 20**

Administración de Clientes

| Ítem | Representación             | Descripción                |
|------|----------------------------|----------------------------|
| A    | Administración de Clientes | Administración de Clientes |
| B    | Búsqueda                   | Sistema de Búsqueda        |
| C    | Administración             | Mostrar y Editar Producto  |

**Agregar Cliente**



Crear Cliente

Los campos con \* son obligatorios.

Nombre \*  Ruc \*

Dirección \*  Teléfono \*

Teléfono2  Email

Estado \* Activo  Descripción

Ciudad \* -- Seleccione --  Id Usuario

Crear Cancelar

2014 © Mauricio Cuichan

*Figura 48: Agregar Clientes: el administrador puede realizar el ingreso de un nuevo cliente llenando los campos de registro.*

**Tabla 21**

*Agregar Clientes*

| Ítem | Representación | Descripción           |
|------|----------------|-----------------------|
| A    | Nuevo Cliente  | Agregar Nuevo Cliente |
| B    | Registros      | Campos de Registros   |
| C    | Botón          | Crear y Cancelar      |

## Administrar Proveedor

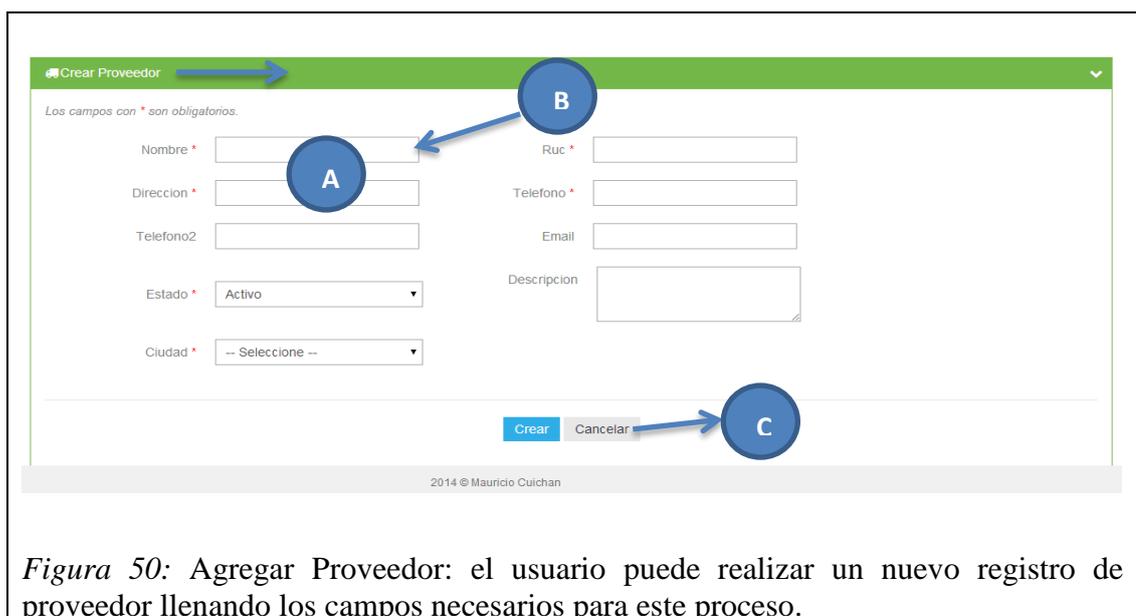


**Tabla 22**

### Administrar Proveedor

| Ítem | Representación          | Descripción                |
|------|-------------------------|----------------------------|
| A    | Administrar Proveedores | Administrar de Proveedores |
| B    | Búsqueda                | Sistema de Búsqueda        |
| C    | Administración          | Mostrar y Editar           |

## Agregar Proveedor

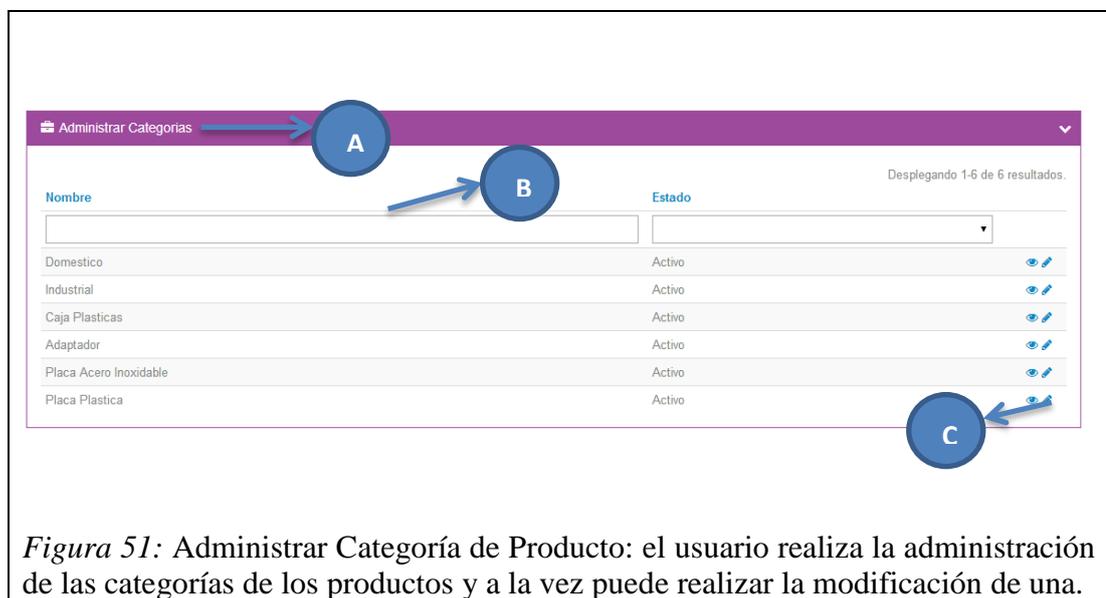


**Tabla N° 23**

*Agregar Proveedor*

| Ítem | Representación | Descripción           |
|------|----------------|-----------------------|
| A    | Nuevo Cliente  | Agregar Nuevo Cliente |
| B    | Registros      | Campos de Registros   |
| C    | Botón          | Crear y Cancelar      |

### Administración de Categorías de Productos



*Figura 51: Administrar Categoría de Producto: el usuario realiza la administración de las categorías de los productos y a la vez puede realizar la modificación de una.*

**Tabla N° 24**

*Administrar Categoría de Producto*

| Ítem | Representación        | Descripción           |
|------|-----------------------|-----------------------|
| A    | Administrar Categoría | Administrar Categoría |
| B    | Búsqueda              | Sistema de Búsqueda   |
| C    | Administrar           | Mostrar y Editar      |

## Agregar Categoría

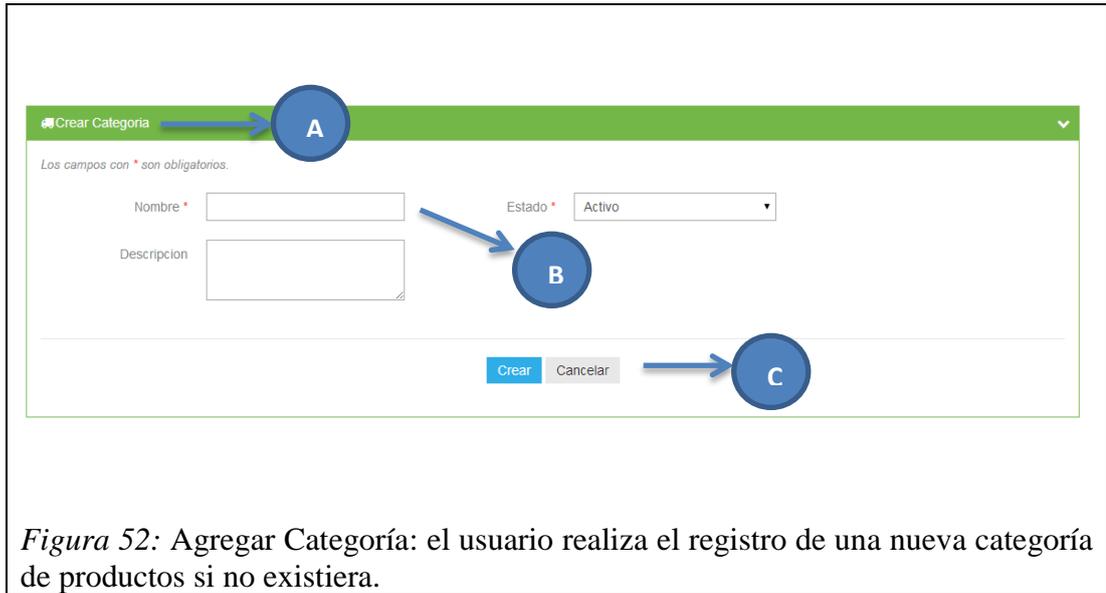


Figura 52: Agregar Categoría: el usuario realiza el registro de una nueva categoría de productos si no existiera.

Tabla 25

### Agregar Categoría

| Ítem | Representación    | Descripción         |
|------|-------------------|---------------------|
| A    | Agregar Categoría | Agregar Categoría   |
| B    | Búsqueda          | Sistema de Búsqueda |
| C    | Botón             | Crear y Cancelar    |

## Administrar Bodega

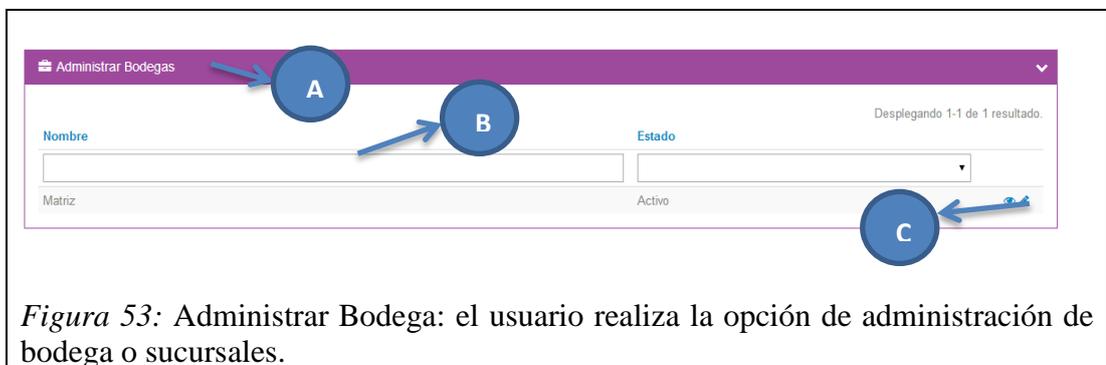


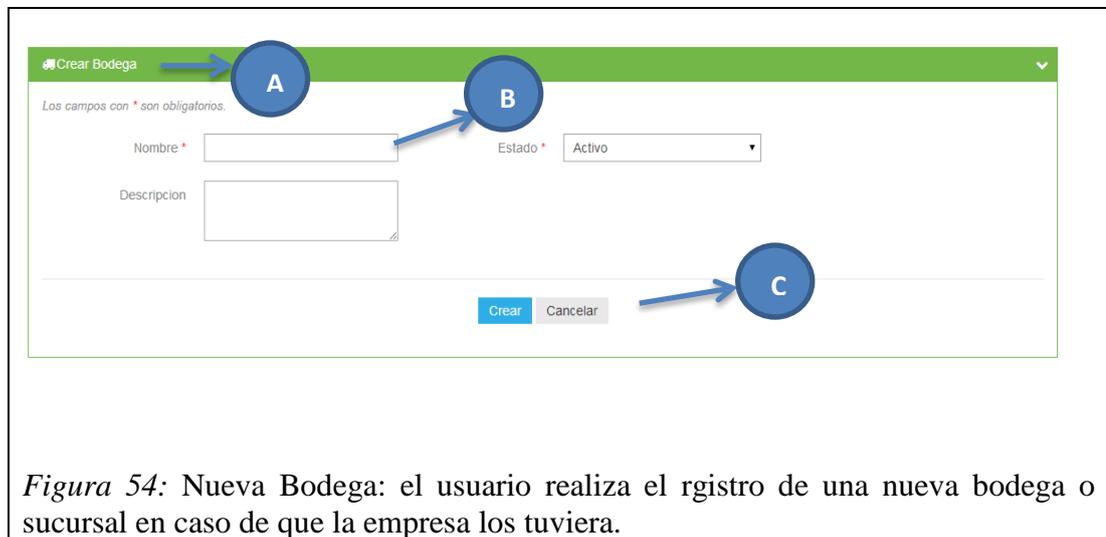
Figura 53: Administrar Bodega: el usuario realiza la opción de administración de bodega o sucursales.

**Tabla 26**

*Administrar Bodega*

| Ítem | Representación     | Descripción         |
|------|--------------------|---------------------|
| A    | Administrar Bodega | Administrar Bodega  |
| B    | Búsqueda           | Sistema de Búsqueda |
| C    | Administración     | Mostrar y Editar    |

**Agregar Bodega**



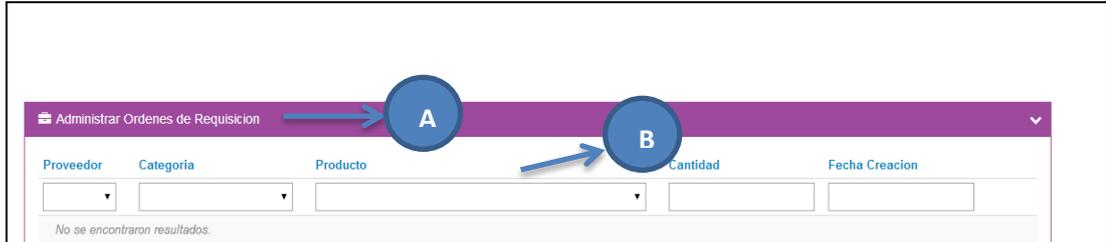
*Figura 54:* Nueva Bodega: el usuario realiza el registro de una nueva bodega o sucursal en caso de que la empresa los tuviera.

**Tabla 27**

*Nueva Bodega*

| Ítem | Representación | Descripción          |
|------|----------------|----------------------|
| A    | Nueva Bodega   | Agregar Nuevo Bodega |
| B    | Registros      | Campos de Registros  |
| C    | Botón          | Crear y Cancelar     |

## Administrar Órdenes de Requisición



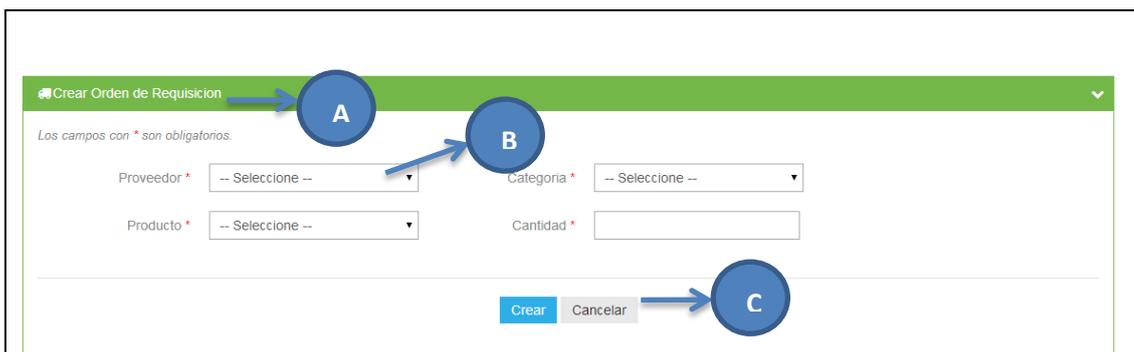
*Figura 55:* Administrar Órdenes de Requisición: el usuario realiza la administración de una orden de requisición y a la vez puede modificar y actualizar la orden.

**Tabla 28**

### Administrar Órdenes de Requisición

| Ítem     | Representación                     | Descripción                        |
|----------|------------------------------------|------------------------------------|
| <b>A</b> | Administrar Ordenes de Requisición | Administrar Ordenes de Requisición |
| <b>B</b> | Búsqueda                           | Sistema de Búsqueda                |

### Nueva Orden de Requisición



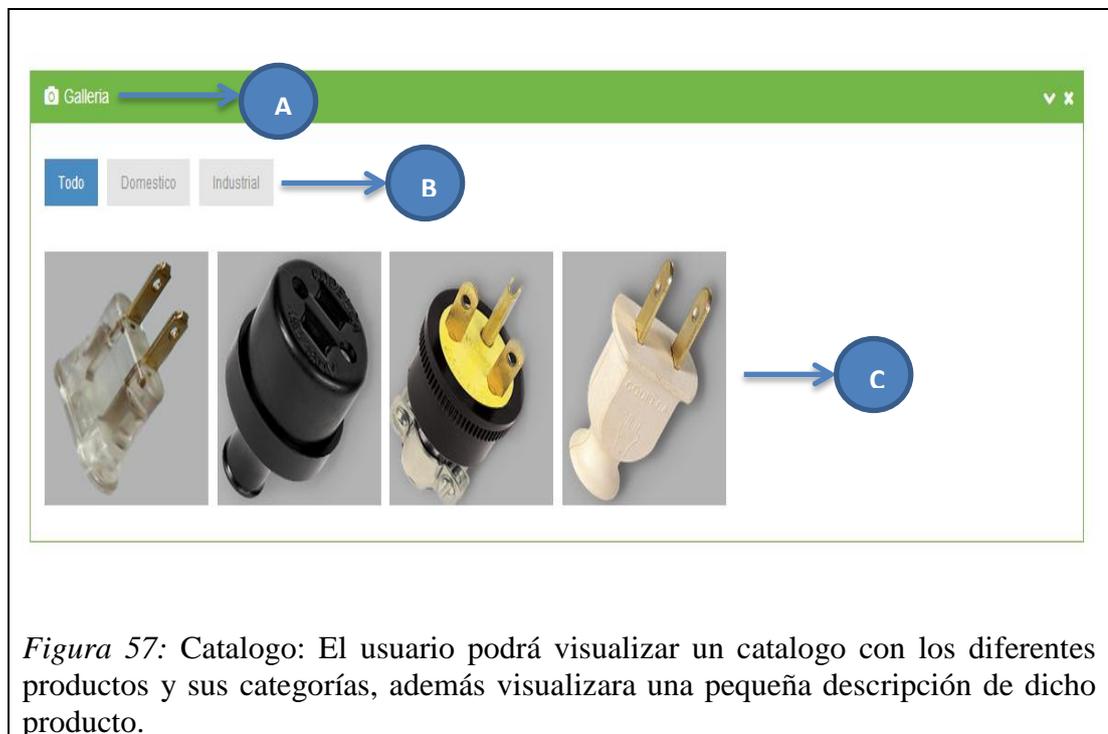
*Figura 56:* Nueva Orden de Requisición: el usuario en esta pestaña puede realizar el registro de una nueva orden de requisición llenando los datos necesarios.

**Tabla 29**

*Nueva Orden de Requisición*

| Ítem | Representación | Descripción         |
|------|----------------|---------------------|
| A    | Nuevo Orden    | Agregar Nuevo Orden |
| B    | Registros      | Campos de Registros |
| C    | Botón          | Crear y Cancelar    |

**Catálogo de productos**



*Figura 57: Catalogo: El usuario podrá visualizar un catalogo con los diferentes productos y sus categorías, además visualizara una pequeña descripción de dicho producto.*

**Tabla 30**

*Catalogo*

| Ítem | Representación | Descripción |
|------|----------------|-------------|
| A    | Catalogo       | Catalogo    |
| B    | Categoría      | Categoría   |
| C    | Productos      | Productos   |

## Administración de Producto

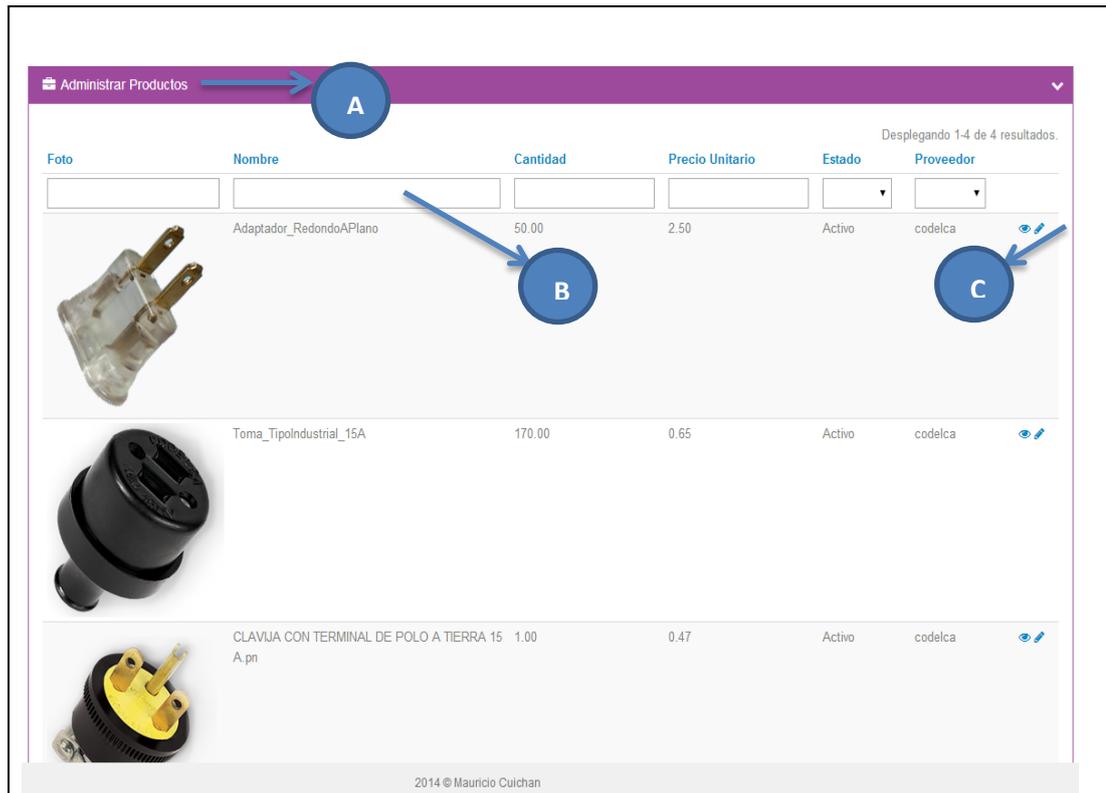


Figura 58: Administración de Productos: El usuario realizara un administración de los productos, además realizara cambios o estado los productos.

**Tabla 31**

### Administración de Productos

| Ítem | Representación              | Descripción                 |
|------|-----------------------------|-----------------------------|
| A    | Administración de Productos | Administración de Productos |
| B    | Búsqueda                    | Sistema de Búsqueda         |
| C    | Administración              | Mostrar y Editar Producto   |

## Control de Productos



Figura 59: Agregar Productos: el usuario puede realizar la opción de registrar un nuevo producto llenando la información necesaria para su venta.

Tabla 32

### Agregar Productos

| Ítem | Representación              | Descripción                 |
|------|-----------------------------|-----------------------------|
| A    | Administración de Productos | Administración de Productos |
| B    | Búsqueda                    | Sistema de Búsqueda         |
| C    | Administración              | Mostrar y Editar Producto   |

## Administrar Guías de Remisión



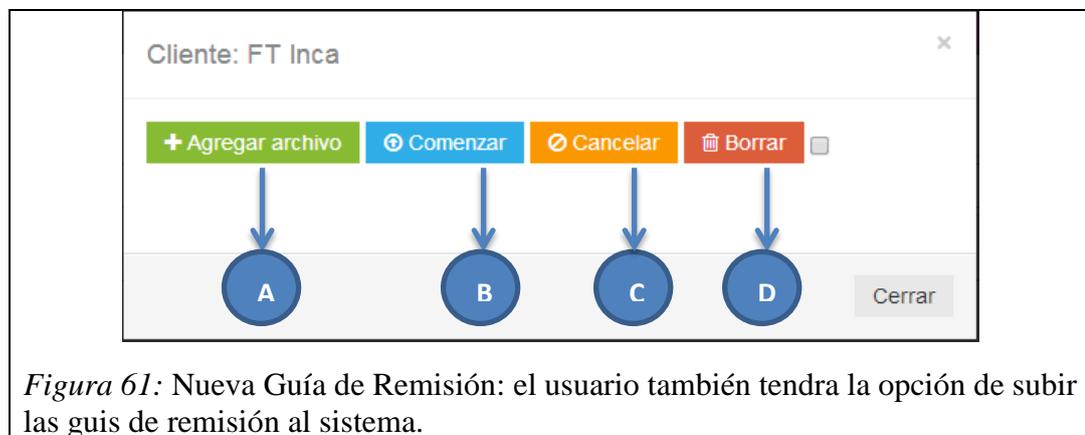
Figura 60: Administrar Guías de Remisión: el usuario realiza la administración de guías de remisión en el cual podrá subir los archivos en pdf del cliente.

**Tabla 33**

*Administrar Guías de Remisión*

| Ítem | Representación                | Descripción                   |
|------|-------------------------------|-------------------------------|
| A    | Administrar Guías de Remisión | Administrar Guías de Remisión |
| B    | Búsqueda                      | Sistema de Búsqueda           |
| C    | Cargar Archivo                | Subir Archivo                 |

**Nueva Guía de Remisión**



**Tabla 34**

*Nueva Guía de Remisión*

| Ítem | Representación  | Descripción     |
|------|-----------------|-----------------|
| A    | Agregar Archivo | Agregar Archivo |
| B    | Comenzar        | Comenzar        |
| C    | Cancelar        | Cancelar        |
| D    | Borrar          | Borrar          |

## Administrar Facturas de Proveedor

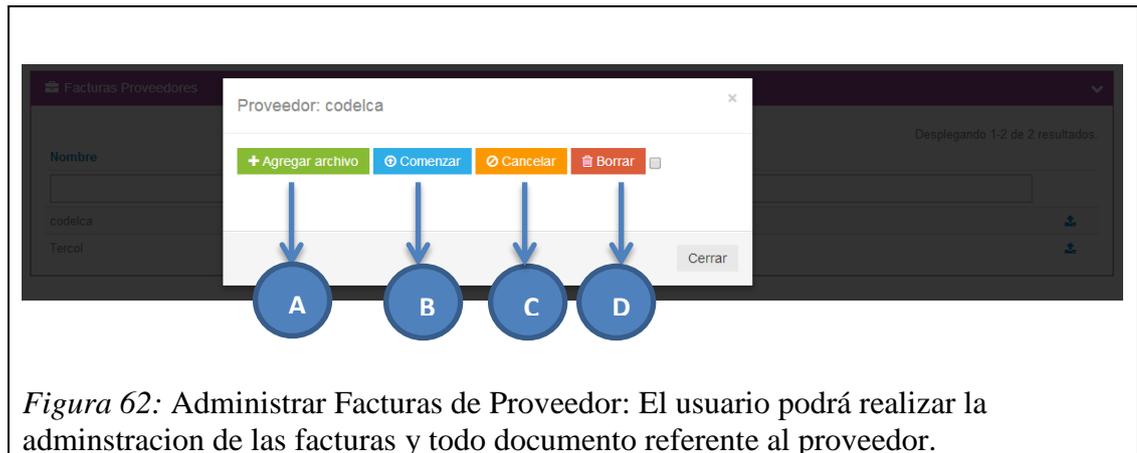


Figura 62: Administrar Facturas de Proveedor: El usuario podrá realizar la administración de las facturas y todo documento referente al proveedor.

### Tabla N° 35:

#### Administrar Facturas de Proveedor

| Ítem | Representación  | Descripción     |
|------|-----------------|-----------------|
| A    | Agregar Archivo | Agregar Archivo |
| B    | Comenzar        | Comenzar        |
| C    | Cancelar        | Cancelar        |
| D    | Borrar          | Borrar          |

### 5.04 Pruebas

Tiene como finalidad, identificar que la información existente en el sistema y los componentes cumplan con los estándares de seguridad y calidad, verificando que los objetos y las clases tengan relación directa con los componentes de software y que sean probados y verificados.

Es necesario que cada una de las clases mantengan una persistencia, considerando que estas clases en lo posterior se transformarán en tablas los objetos que contiene cada una de estas clases deben tener la capacidad de ser reutilizables y consistentes en el desenvolvimiento del proyecto.

#### **5.04.01 Pruebas de Integridad**

Se realiza en la fase de diseño del modelo lógico colocando la persistencia en las clases, momentos antes de generar el modelo físico del negocio.

Este tipo de prueba dará como resultado la integridad de los datos que son manipulados por los objetos y métodos que tienen cada una de las clases participantes.

#### **5.04.02 Pruebas de Módulo.**

##### **5.04.02.01 Módulo de Seguridad**

El usuario podrá realizar su debido ingreso y manipulación de los datos que el administrador le hubiese asignado tomando en cuenta que solo podrá realizar cambios en su cuenta.

El ingreso erróneo en cualquiera de estas dos cajas de captura de datos (usuario, clave), enviará automáticamente a la página principal del sistema.

El número de cédula es indispensable para el sistema ya que servirán como identificadores para los procesos en los cuales se va a desempeñar, por lo tanto el ingreso de estos nuevos valores es importante que sea verificado al momento de capturar los datos en la caja de texto. Se demuestra que el ingreso erróneo de los valores será marcado de color rojo, brindando una alerta oportuna para que el usuario pueda verificar y corregir al mismo tiempo.

##### **5.04.02.02 Módulo Mantenimiento**

Las pruebas en el módulo de mantenimiento permite verificar el acceso idóneo de información hacia la base de datos, por lo tanto la base de datos



debe tener integridad en los campos que manejan, mediante un código de acceso y en cada tabla, teniendo una función que me permita generar códigos secuenciales sin que estos se repitan y puedan generar errores de duplicidad de claves primarias.

#### **5.04.03 Pruebas de Interfaz de Usuario.**

Esta prueba se realiza a través de la utilización y el ambiente con la que el usuario se desenvuelve en realizar las distintas operaciones y actividades de la empresa, a la vez midiendo el tiempo de respuesta que tardan en refrescar y responder los navegadores en el ambiente browser.

Además se verifica la comunicación que existe entre los datos desde la lógica del negocio y la base de datos.

## Capítulo VI: Aspectos Administrativos

### 6.01 Recursos

Tabla 36

Recursos

| RECURSOS HUMANOS           |                       |                                     |   |
|----------------------------|-----------------------|-------------------------------------|---|
| HUMANO                     | NOMBRE                | ACTIVIDAD                           | RESPONSABILIDAD                               |
| <b>Tutor</b>               | Ing. Richard Mafla    | Director del Proyecto               | Guía e instruye al desarrollador del proyecto |
| <b>Administrador</b>       | Ing. Héctor Naranjo   | Autoriza la elaboración del Sistema | Toma decisiones                               |
| <b>Empleados</b>           | Empleados             | Proporciona Información             | Atiende a los clientes                        |
| <b>Recursos Físicos</b>    | Equipo de Computo     |                                     |   |
|                            | Impresora             |                                     |   |
| <b>Recursos Económicos</b> | Alimentos-Refrigerios |                                     |   |
|                            | Transporte            |                                     |   |
| <b>Recursos Renovables</b> | Hojas                 |                                     |   |
|                            | Energía               |                                     |   |
|                            | Anillados             |                                     |   |
|                            | Empastados            |                                     |   |
| <b>Recurso Didáctico</b>   | Internet              |                                     |   |
| <b>Otros Recursos</b>      | Seminarios            |                                     |   |
|                            | Tutorías              |                                     |   |

*Nota:* Determina y analiza los recursos necesarios.

**Fuente:** Mauricio Cuichán

## 6.02 Presupuesto

**Tabla 37**

*Presupuesto*

| <b>PRESUPUESTO DEL PROYECTO</b> |                       |                       |                    |
|---------------------------------|-----------------------|-----------------------|--------------------|
| <b>CANT.</b>                    | <b>DESCRIPCIÓN</b>    | <b>VALOR UNITARIO</b> | <b>VALOR TOTAL</b> |
| <b>540</b>                      | Impresiones Color     | \$ 0,25               | \$ 135,00          |
| <b>540</b>                      | Impresiones B/N       | \$ 0,10               | \$ 54,00           |
| <b>1</b>                        | Materiales de Oficina | \$ 20,00              | \$ 20,00           |
| <b>320</b>                      | Transporte            | \$ 0,25               | \$ 80,00           |
| <b>2</b>                        | Anillados             | \$ 2,50               | \$ 5,00            |
| <b>2</b>                        | Empastados            | \$ 3,50               | \$ 7,00            |
| <b>6</b>                        | Internet              | \$ 21,00              | \$ 126,00          |
| <b>1</b>                        | Seminarios            | \$ 520,00             | \$ 520,00          |
| <b>1</b>                        | Tutorías              | \$ 200,00             | \$ 200,00          |
|                                 |                       | <b>TOTAL</b>          | <b>\$ 1.147,00</b> |

Nota: determina el presupuesto para el proyecto

**Fuente:** Mauricio Cuichán

## 6.03 Cronograma de Actividades

Un cronograma es la interpretación en una gráfica de tiempo la cronología de un hecho o trabajo que se representa en un par de ejes de coordenadas, el eje de abscisas se divide en fracciones de tiempo, por ejemplo, días, semanas, meses, años, y en el eje de coordenadas se describe la tarea a realizar o la tarea realizada marcando el tiempo.

Si el desarrollo del proyecto está planificado para construirse en 6 meses iniciando el lunes 02/11/13 hasta lunes 21/04/14, con las actividades que se detallan anexados a continuación (Ver Anexo A.05).

---

## Capítulo VII: Conclusiones y Recomendaciones

### 7.01 Conclusiones

- Con la implementación del sistema lograremos la optimización en el control de inventarios a través de los módulos con los que cuenta el sistema los cuales permiten el funcionamiento de la misma.
- Los administradores tendrán la opción de realizar la búsqueda de la información que genera el sistema y agiliza los procesos operacionales que se realizan en la actualidad, aportando nuevas estrategias en el desarrollo de las actividades.
- El sistema permitirá un control de cada cliente, proveedor, producto y bodega para facilitar la información con eficiencia y rapidez.
- Esta Empresa contara con el aporte tecnológico beneficiando a sus clientes, empleados y a las personas aledañas del sector.

### 7.02 Recomendaciones

- El personal y los administradores tiene la obligación de revisar los manuales para el buen desempeño del sistema.
- Los errores que se presenten al momento de la manipulación del sistema se deberá reportar al personal técnico para la solución inmediata.



- 
- Es necesario que los equipos tecnológicos que contaran el personal debe estar actualizados con la última tecnología para el desempeño de sus funciones.
  
  - El acceso a los mantenimientos del sistema deben ser exclusivos del administrador, con el propósito de mantener la persistencia y la integridad de los datos, permitiendo que la seguridad del software se encuentre en un estado óptimo.



# ANEXOS

## A.01 Matriz de Análisis de Involucrados

**Tabla 5**

*Matriz de Análisis de Involucrados*

| Actores Involucrados | Intereses sobre el problema central                       | Problemas Percibidos   | Recursos, Mandatos y Capacidades                    | Intereses sobre el Proyecto                 | Conflictos Potenciales                                |
|----------------------|---|--|---|---|---|
| CLIENTE              | Recibir una mejor atención de calidad y comodidad.        | No hay suficiente información sobre los servicios que brinda la empresa. | Recurso Humano                                      | Obtener una atención de calidad.            | Pérdida de tiempo al realizar un pedido.              |
| EMPLEADO             | Mejorar los procesos de gestión de pedidos automatizados. | No es posible generar informes de pedidos.                               | Reportes, presupuestos<br>Recurso Humano            | Ayudar a procesar la gestión de pedidos.    | Registro manuscrito de los pedidos.                   |
| BODEGUERO            | Mejorar el gestión de pedidos y automatizarlos.           | No se puede realizar consultas sobre los pedidos.                        | Presupuestos<br>Recurso Humano                      | Ayudar a procesar la generación de pedidos. | Mala coordinación entre el personal de la empresa.    |
| ADMINISTRADOR        | Ejercer su autoridad de coordinar la empresa.             | Poca autoridad en el control de procesos.                                | Registros, reportes, presupuestos<br>Recurso Humano | Mejorar los procesos de control.            | Mejorar la organización y coordinación de la empresa. |

|           |  |  |                |   |   |
|-----------|--|--|----------------|---|---|
| PROVEEDOR | Mejorar el control de los entregables en la empresa. | No hay un adecuado control de los proveedores. | Recurso Humano | Optimizar el control que se realizan a los proveedores. | Optimizar el control y la comunicación entre la empresa con el proveedor. |
|-----------|--|--|----------------|---|---|

*Nota:* Analiza la matriz de involucrados

## A.02 Matriz de Análisis de Alternativas

**Tabla 6**

*Matriz de Análisis de Alternativas*

| Objetivos   | Impact. del propósito | Factib. Técnica | Factib. Financ. | Factib. Social | Factib. política | Total | Categoría |
|---|-----------------------|-----------------|-----------------|----------------|------------------|-------|-----------|
| Solucionar necesidades, los cuales permitan elevar la calidad de atención a los clientes. | 4                     | 3               | 4               | 2              | 4                | 17    | Alto      |
| Automatizar y mejorar los procesos de gestión de pedidos.                                 | 4                     | 4               | 4               | 3              | 2                | 17    | Alto      |
| Optimizar el proceso de registro de la información de los pedidos.                        | 4                     | 4               | 4               | 2              | 2                | 16    | Alto      |
| Conocimiento del estado en el que se encuentra la actividad, proceso o registro.          | 4                     | 3               | 4               | 3              | 3                | 17    | Alto      |
| <b>TOTAL</b>  | 16                    | 14              | 16              | 10             | 11               | 67    |           |

*Nota:* Matriz de Análisis de Alternativas

### A.03 Matriz de Análisis de Impacto de los Objetivos

**Tabla 7**

*Matriz de Análisis de Impacto de los Objetivos*

| AUMENTO DE LA PRODUCTIVIDAD Y CONFORMIDAD EN LAS ACTIVIDADES Y PROCESOS, DEBIDO A LA AGILIDAD DE PEDIDOS Y CONTROL DE INVENTARIOS. | <i>Factibilidad de Lograse</i><br>(Alta-Media-Baja)<br>( 4 - 2 - 1 )  | <i>Relevancia</i><br>(Alta-Media-Baja)<br>( 4 - 2 - 1 )  | <i>Sostenibilidad</i><br>(Alta-Media-Baja)<br>( 4 - 2 - 1 )   | <i>Total</i>   |
|--|---|--|---|--|
|  | -Los beneficios son mayores que los costos.<br>- Cuenta con financiamiento por parte de la empresa.<br>-Protege el uso de los recursos.<br>-Es aceptable y conveniente para los beneficiarios<br>-Existe tecnología adecuada para su realización<br>-Se cuenta con soporte político-institucional.<br>-Mejora el entorno social con los clientes. | -Responde a las expectativas de los beneficiarios<br>-Es una prioridad sentida por los beneficiarios<br>-Beneficia a grupos de mayor carencia y vulnerabilidad<br>-Los beneficios son deseados por los beneficiarios | -Fortalece la participación de los empleados y clientes.<br>-Fortalece la Organización local<br>-Los beneficiarios están interesados en el proyecto<br>-la Empresa respalda el financiamiento del proyecto a futuro | <b>60 puntos</b><br><b>22 a 32</b><br><b>BAJA</b><br><b>33 a 44</b><br><b>MEDIA</b><br><b>BAJA</b><br><b>45 a 66</b><br><b>MEDIA</b><br><b>ALTA</b><br><b>67 a 88</b><br><b>ALTA</b> |
|  | <b>28 Puntos</b>  | <b>16 puntos</b>   | <b>16 puntos</b>  |  |

*Nota:* Matriz de Análisis de Impacto de los Objetivos

#### A.04 Matriz de Marco Lógico

Figura 19: Matriz de Marco Lógico

| RESUMEN NARRATIVO   | INDICADORES  | MEDIOS DE VERIFICACIÓN  | SUPUESTOS   |
|---|--|---|---|
| <b>FIN DEL PROYECTO</b><br><br><b>Facilidad al momento de elaborar los informes y reportes de las actividades y procesos de la empresa.</b>   | -En el tiempo de uso del sistema se espera que incremente un 60% la satisfacción de las nuevas estrategias de registro y control.  | - Las encuestas realizadas determina que los informes que se generan son escasos en relación a lo requerido por las autoridades.<br><br>- Resultados de encuestas realizadas a los clientes y personal de la empresa. | - Los administradores y el personal, no tendrán la dificultad en la generación de informes y en el registro de pedidos.   |
| <b>PROPÓSITO DEL PROYECTO</b><br><br><b>Mejorar la calidad de atención, y mantener un control actualizado de los registros de pedidos, inventarios, clientes y proveedores.</b>           | Aumento de un 70% en el manejo eficaz en los procesos y actividades de la empresa.<br><br>- El número de demora en la atención se reduce en (+/-15 minutos)              | - Encuestas a los empleados sobre el uso de la herramienta tecnológica.<br><br>- Encuestas a los clientes sobre el servicio que reciben la empresa.   | - Se logra optimizar el tiempo de actividades y procesos de gestión y registros.  |
| <b>COMPONENTES DEL PROYECTO</b><br><br><b>1. Se ha creado soluciones que permitan elevar la calidad de atención a los clientes.</b><br><br><b>2. Se dispone de la tecnología adecuada</b> | - Los registros y el control de los pedidos aumentarían un 75% con el proceso de automatización de todas las actividades.<br><br>- El acceso a la información aumentaría | - Informes sobre el estado en que se encuentra la empresa y el tipo de atención que realizan a los clientes.<br><br>- Consultas sobre el  | - Los administradores y el personal optaron con nuevos conocimientos y están dispuestos a utilizar la tecnología mediante la cual van a realizar sus labores con la dedicación de brindar comodidad a |

|  |  |  |   |
|--|--|--|---|
| <p>para registrar la información de los pedidos.</p> <p>3. Existe una mejor coordinación en el personal.</p>   | <p>90% con la implementación del sistema tecnológico.</p> <p>-Aumento de un 95% en la coordinación de los procesos y actividades de la empresa.</p>  | <p>interés que tiene el personal sobre la tecnología a implementar.</p> <p>-Se confirman en los informes de los procesos control de pedidos e inventarios que se realizan en la empresa.</p>   | <p>todas los clientes y proveedores.</p> <p>-El personal se encuentra 100% capacitado para ofrecer los servicios con las nuevas estrategias en sus procesos de control.</p>   |
| <p><b>ACTIVIDADES DEL PROYECTO</b></p> <p>1.1 Implementar un sistema que mejore el tiempo de respuesta a los clientes.</p> <p>1.2 Facilitar el uso de la información de la empresa.</p> <p>1.3 Agilizar el proceso de registro de pedidos.</p> <p>2.1 los empleados tendrán la facilidad de registrar los procesos de gestión de pedidos.</p> <p>2.2 los empleados podrán tener acceso a la información desde cualquier lugar.</p> <p>2.3 El traslado de</p> | <p><b>PRESUPUESTO</b></p> <p><b>Materiales</b></p> <p>Impresiones Color (\$ 135,00)</p> <p>Impresiones B/N (\$ 54,00)</p> <p>Materiales de Oficina (\$ 20,00)</p> <p>Alimentos-Refrigerios (\$ 180,00)</p> <p>Transporte (\$ 80,00)</p> <p>Anillados (\$ 5,00)</p> <p>Empastados (\$ 7,00)</p> <p>Internet (\$ 126,00)</p> <p>Seminarios (\$ 520,00)</p> <p>Tutorías</p> | <p><b>MEDIOS DE VERIFICACIÓN</b></p> <p>- Informe sobre los equipos de desarrollo a utilizar.</p> <p>Verificar módulo de seguridad.</p> <p>Verificar módulo mantenimiento y módulo de reglas de negocio.</p> <p>Cuantificar las entrevistas realizadas</p> <p>Información disponible para la realización del sistema por parte de los empleados de la empresa.</p> | <p><b>SUPUESTOS</b></p> <p>Personal altamente capacitado para el manejo operativo.</p> <p>Beneficiarios directos satisfactorios con la aplicación proporcionada.</p> <p>Agilidad y seguridad en los procesos operativos de la empresa.</p> <p>Incremento de ingresos para la empresa.</p> |

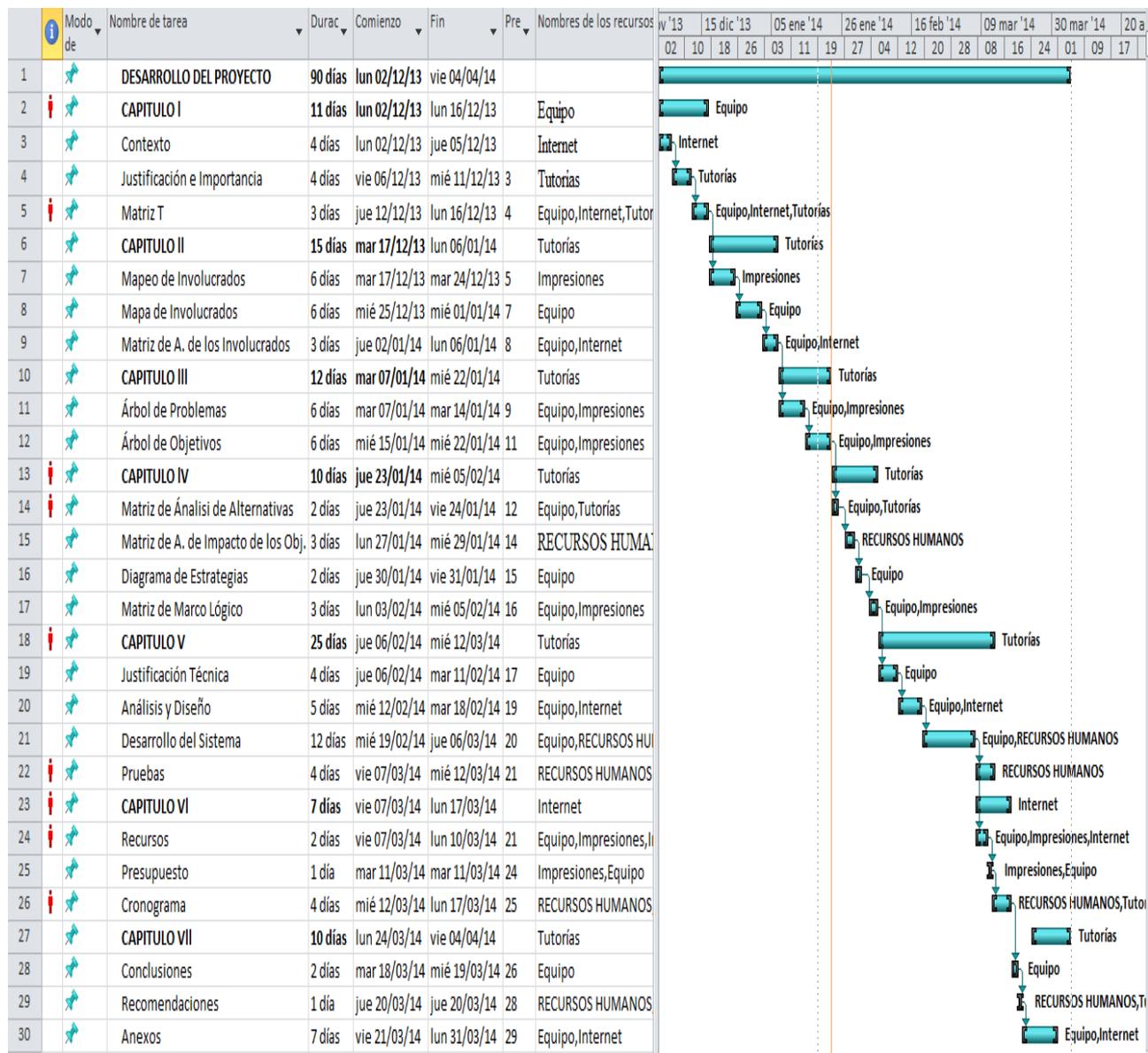


|  |                   |  |  |
|--|-------------------|--|--|
| <b>carpetas físicas de los pedidos no serán necesarios.</b>                                  | \$ 200,00         |  |  |
|  | Total Presupuesto |  |  |
|  | (\$ 1327.00)      |  |  |
| <b>2.4 El personal obtendrá los conocimientos necesarios sobre el uso de la herramienta.</b> |                   |  |  |
| <b>2.5 La implementación del proyecto cuenta con todo el respaldo necesario.</b>             |                   |  |  |

## A.05 Cronograma de Actividades

Tabla 38

### Cronograma de Actividades



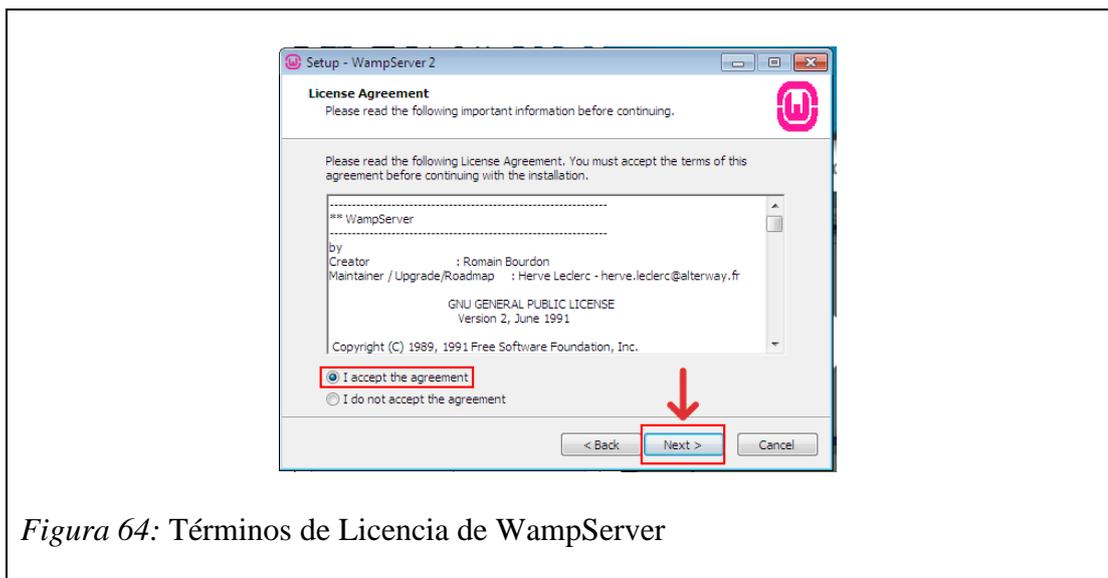
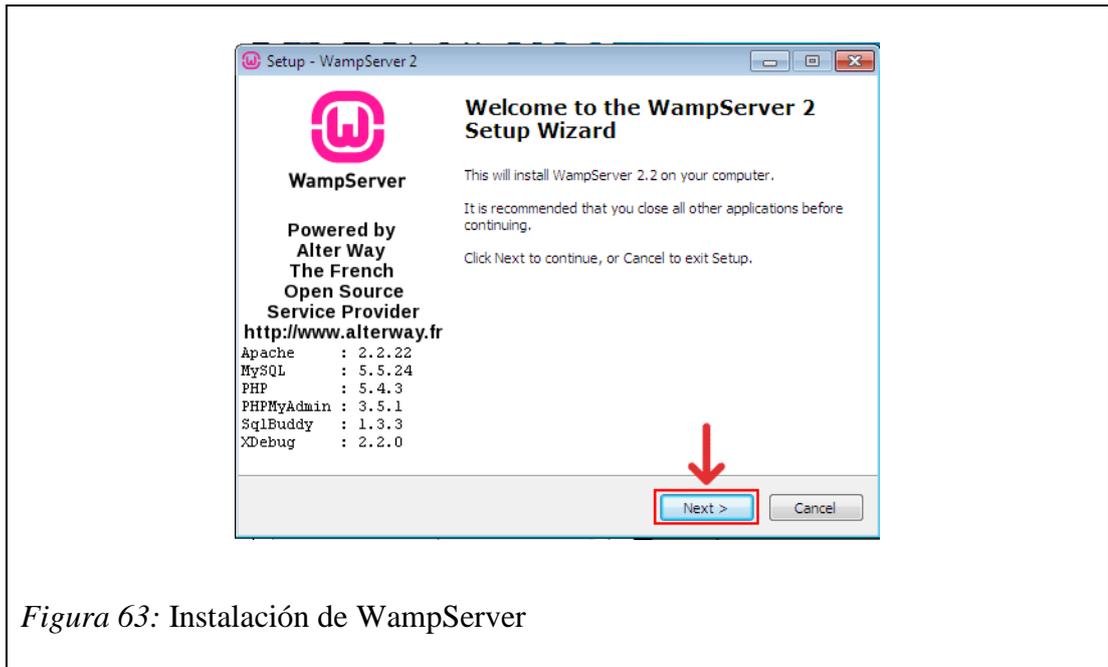
Nota: Determina el cronograma de actividades del proyecto

## A.06 Instalación de Programas

### A.06.01 Instalación del Servidor de Base de Datos

En el nuestro proyecto el servidor de base de datos será el Mysql para lo cual utilizaremos en WampServer.

En la pantalla de bienvenida presionamos next



Seleccionamos el directorio de la instalación y presionamos next.

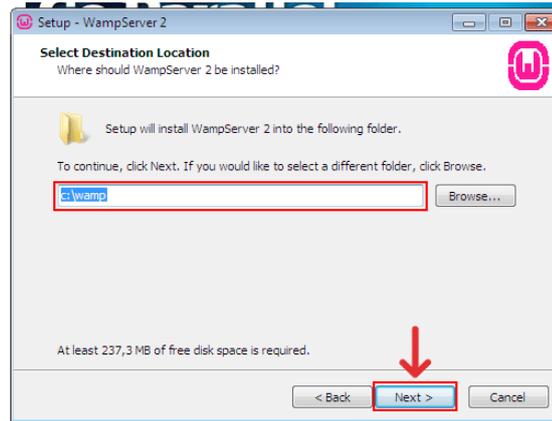


Figura 65: Directorio de Instalación de WampServer

Escogemos los accesos directos deseados y presionamos next.

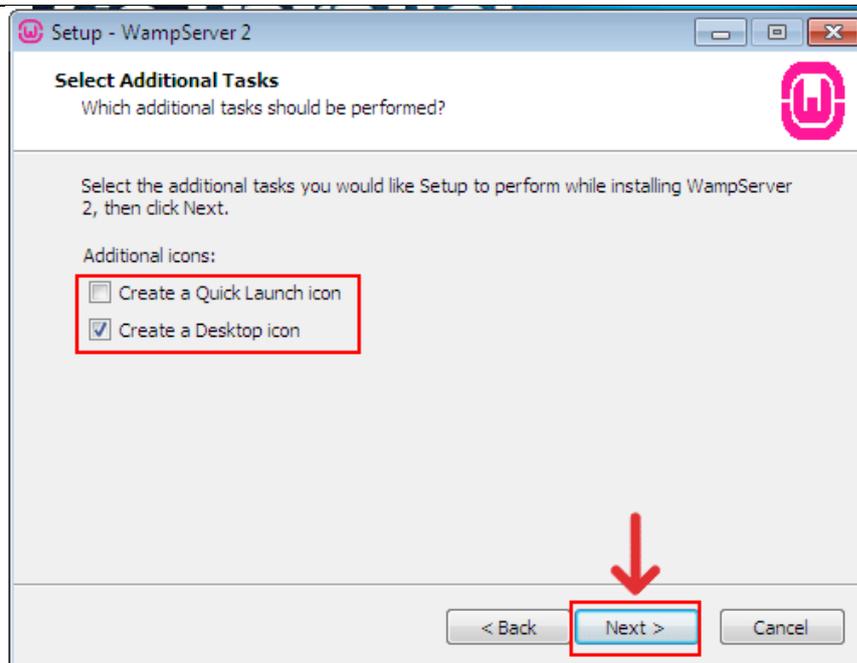
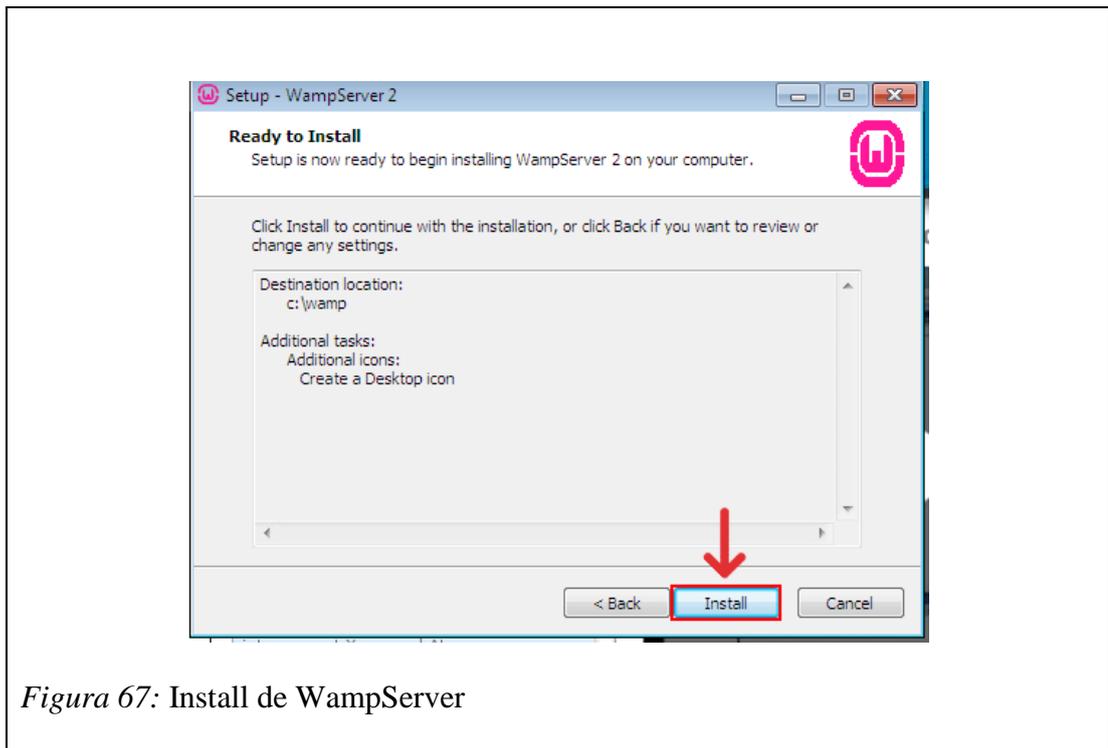


Figura 66: Accesos Directos de WampServer

Presionamos Install



Esperamos a que el proceso de instalación finalice.



Si queremos que nuestro Firefox sea el explorador web por defecto de WampServer presionamos Si caso contrario No.

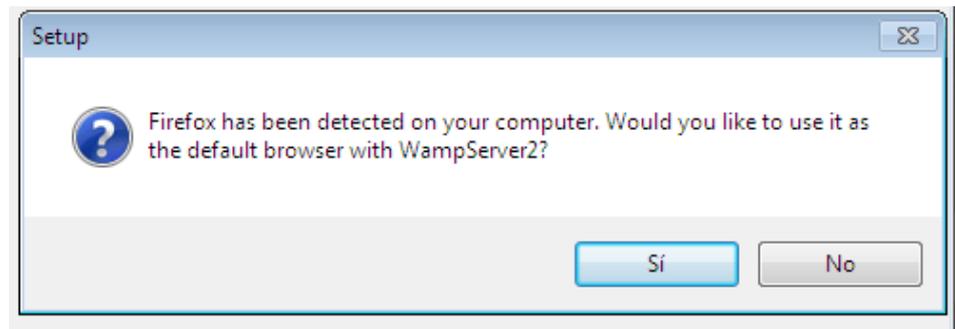


Figura 69: Explorador de Firefox por defecto para WampServer

En caso de que no encuentre ningún explorador browser lo pondrá como por defecto el explorer.exe, simplemente presionamos en abrir.

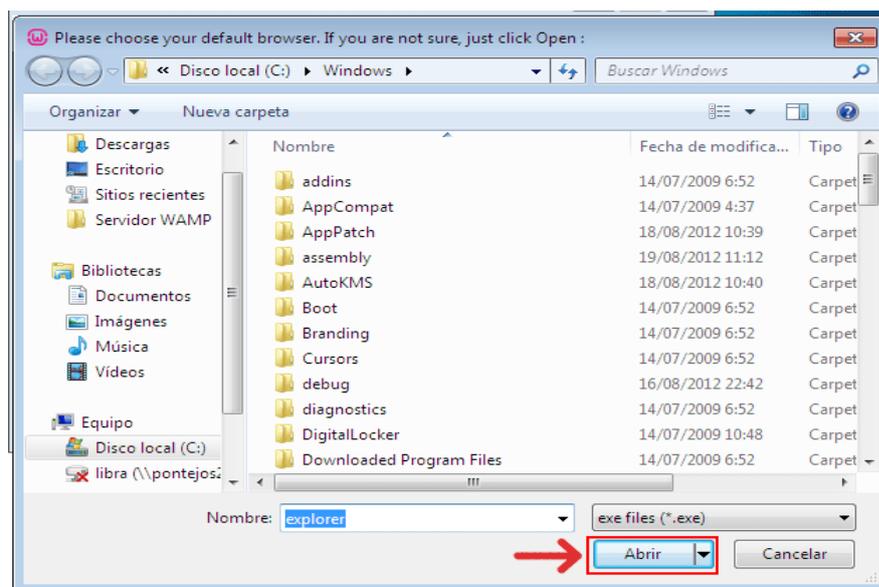


Figura 70: Explorador de Windows como browser para el WampServer

Digitamos el nombre del servidor SMTP y el Email si nuestro servidor es local dejamos los campos tal como nos muestra y presionamos next.



Figura 71: Servidor SMTP y un Email de WampServer

Una vez completada la instalación activamos el inicio automático y presionamos finish.



Figura 72: Completando la Instalación de WampServer.



Figura 73: Icono del WampServer en la Barra de Tareas

Para cambiar el idioma nos dirigimos a Language y luego escogemos en idioma deseado.

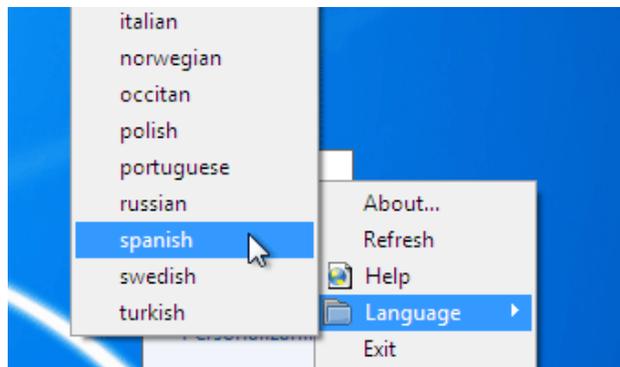


Figura 74: Cambio de Idioma en el WampServer

Para iniciar el servicio presionamos Encender.

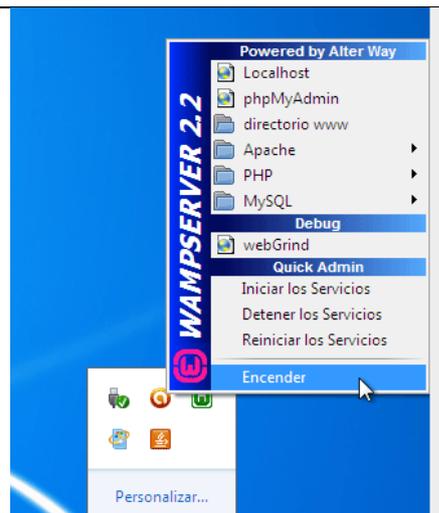


Figura 75: Inicio del Servicio de WampServer

Nos dirigimos al explorador web y escribimos localhost en la barra de direcciones y obtendremos ya nuestro WampServer instalado.



Figura 76: Pantalla de Presentación de WampServer

### Cambiar Puerto en wampserver

Primeramente ejecutamos el servidor y le hacemos clic en Apache->httpd.conf tal como se muestra en la siguiente imagen:

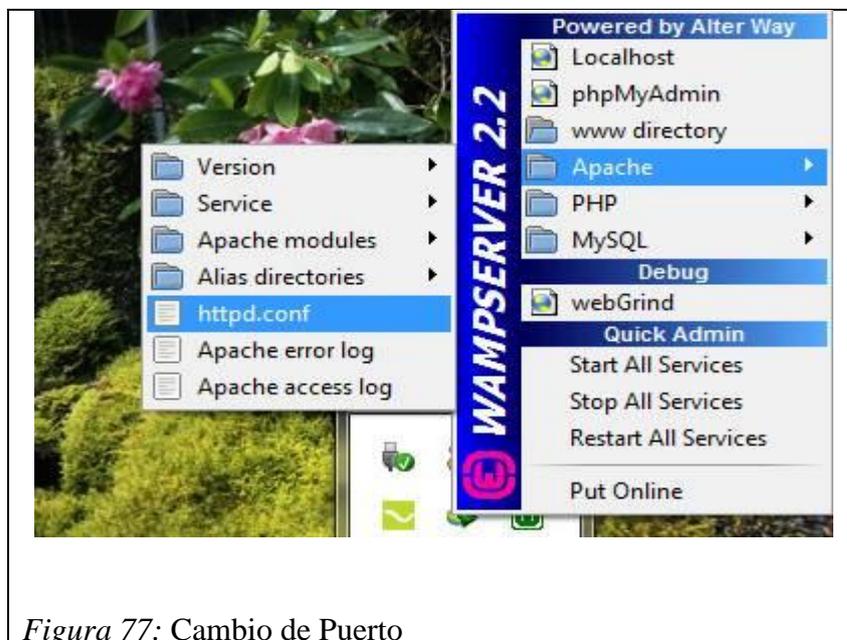


Figura 77: Cambio de Puerto

Nos abrirá un documento de texto buscamos el siguiente párrafo:

CONTROL DE PEDIDOS E INVENTARIO DE PRODUCTOS MEDIANTE UNA APLICACION WEB PARA LA EMPRESA DE MATERIAL ELÉCTRICO "IMPORT"

```
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
# Change this to listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#Listen 12.34.56.78:80
Listen 80
# Dynamic Shared Object (DSO) Support
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding 'LoadModule' lines at this location so the
```

Figura 78: Texto ID

Y donde dice 80 le cambiamos por 82 y debe quedarnos así:

```
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
# Change this to listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#Listen 12.34.56.78:82
Listen 82
# Dynamic Shared Object (DSO) Support
```

Figura 79: Cambio por puerto 82

Ahora más abajo hay otra línea la cual cambiamos de la misma forma como se muestra en la siguiente imagen cambiar de 80 a 82:

```
option: "Powered by
Host: "localhost:82";
ServerName: "phpMyAdmin"; Ac
```

Figura 80: Cambio en localhost

Ahora cerramos y cuando nos pregunte guardar los cambios le decimos que sí.

Ahora nos vamos a "C:\wamp" y abrimos wampmanager.ini con notepad++ y

buscamos la línea:288, 298 y 296 y reemplazamos *localhost* por *localhost: 80* en las tres líneas de be quedarnos así:

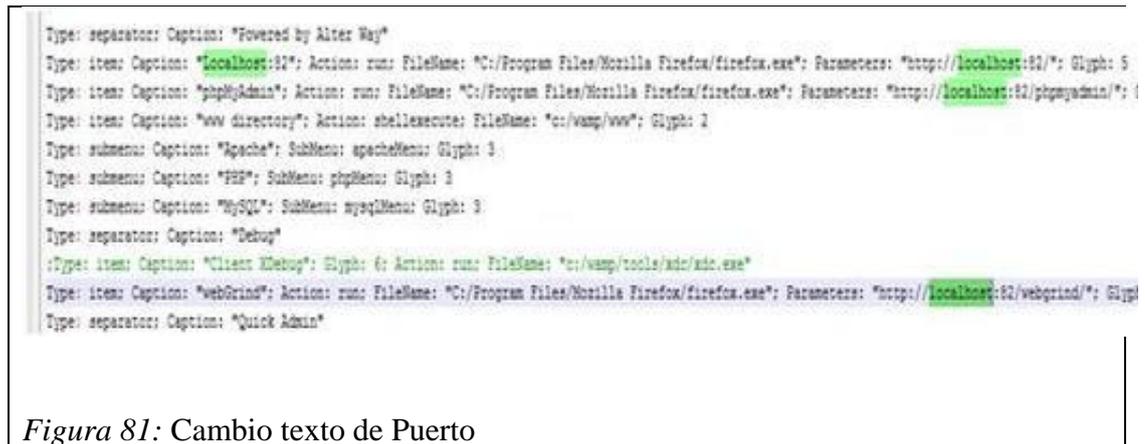


Figura 81: Cambio texto de Puerto

Son en cuatro lugares en los cuales se debe cambiar como se muestra en la imagen siguiente, recuerda solo aumentar dos puntos y escribir 80 como se muestra en la imagen de arriba.

Después cerramos y guardamos los cambios y volvemos a reiniciar WAMP

SERVER recuerda que el explorador tienes que usar ahora ya no localhost si no

localhost: 80 como se muestra en la siguiente imagen:

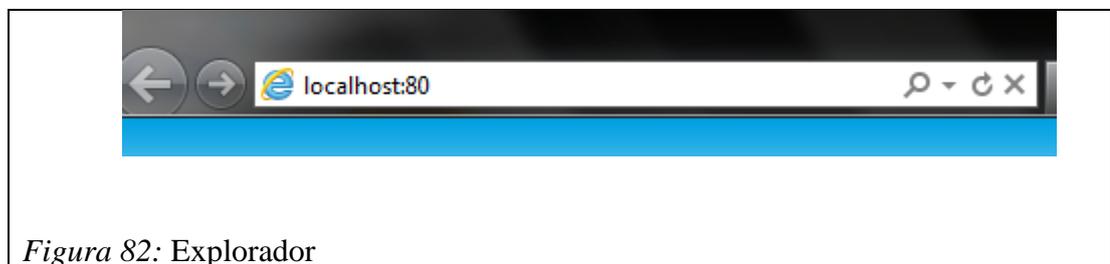


Figura 82: Explorador

Y listo ya tienes acceso a tu servidor local.

### A.06.02 Instalación del NetBeans IDE 7.2

Se debe ejecutar como administrador el archivo que contiene el software que vamos a trabajar el cual será la versión NetBeans 7.2.

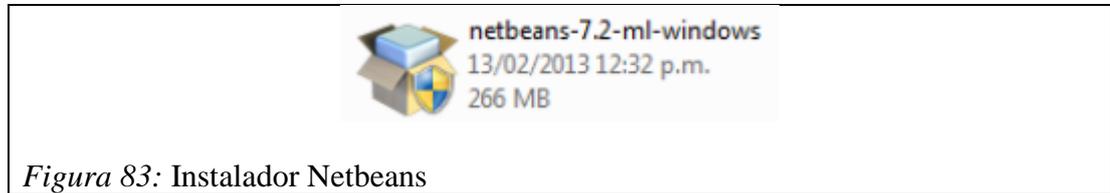


Figura 83: Instalador Netbeans

Dar clic derecho sobre el archivo y seleccionar la opción Ejecutar como administrador.

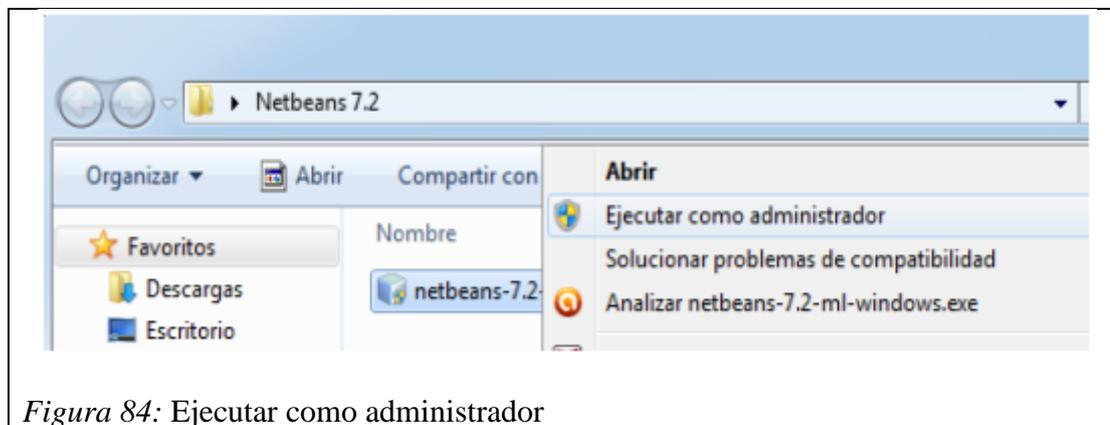


Figura 84: Ejecutar como administrador

Comenzará a configurarse el instalador.

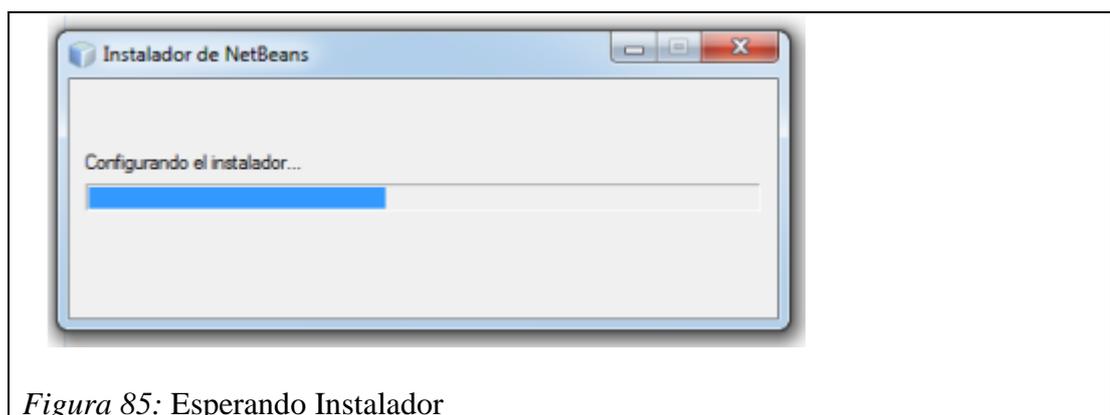


Figura 85: Esperando Instalador

Instalador de NetBeans IDE, esperar a que termine de cargar

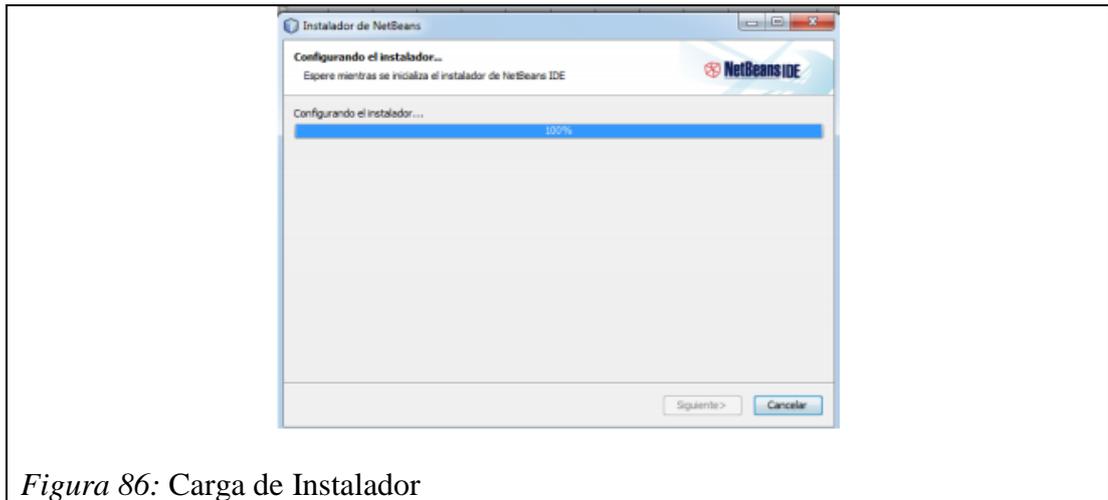


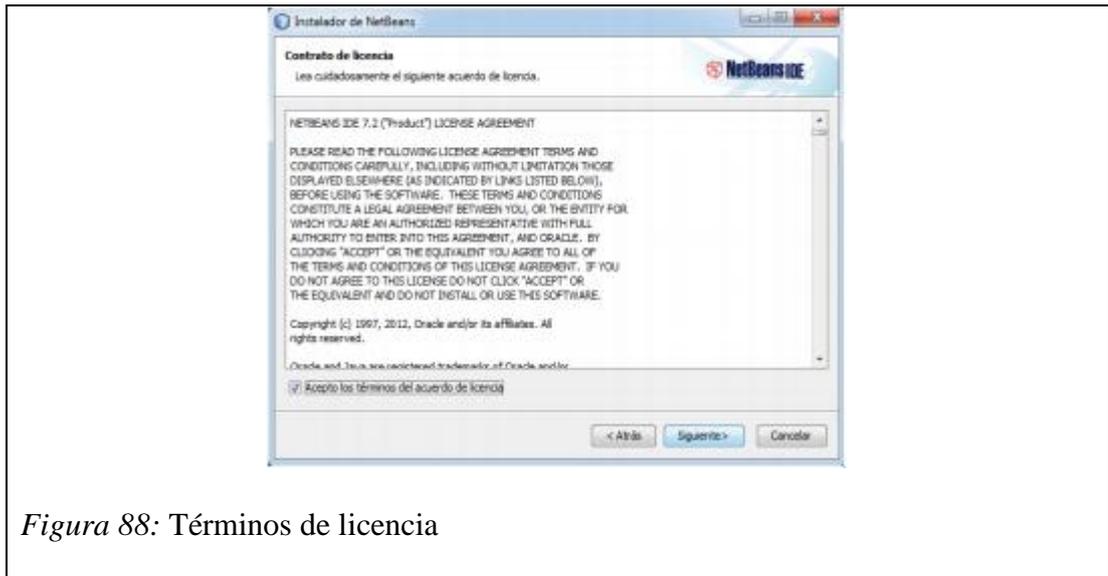
Figura 86: Carga de Instalador

Aparecerá la pantalla de bienvenida, presionar el botón siguiente

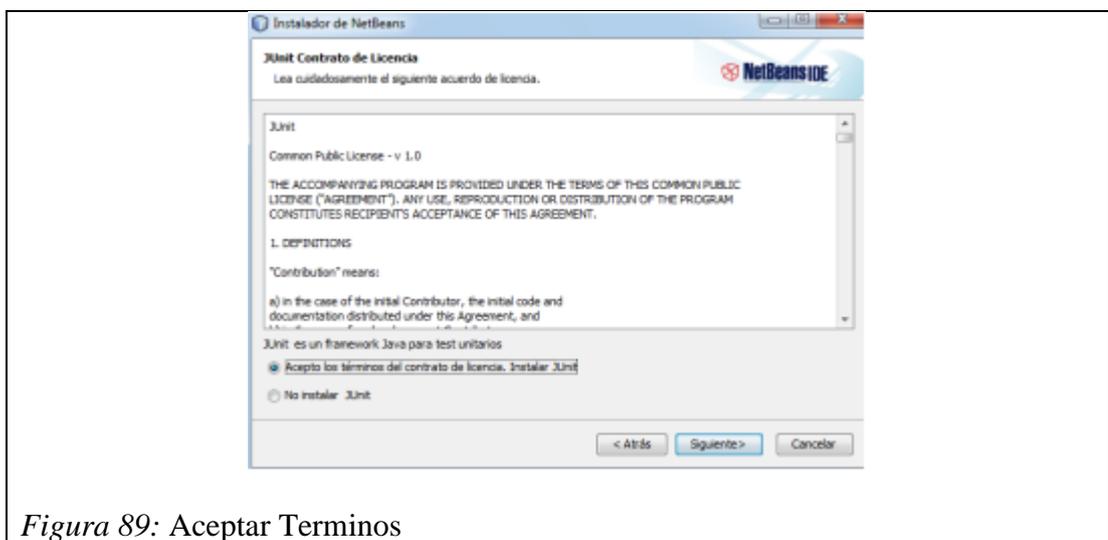


Figura 87: Pantallada de Instalador

Nos mostrara el contrato de licencia para el uso de NetBeans IDE 7.2 seleccionamos la casilla que dice Acepto los términos del acuerdo de licencia y presionamos siguiente.

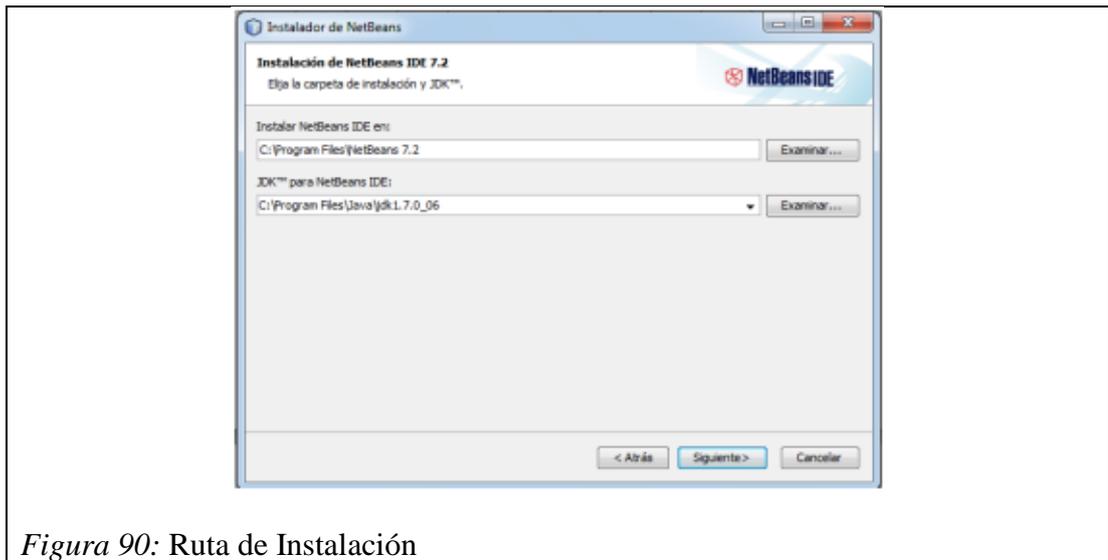


Nos muestra el contrato de licencia para JUnit, activamos la casilla Acepto los términos del contrato de licencia. Instalar JUnit y presionamos el botón de siguiente.

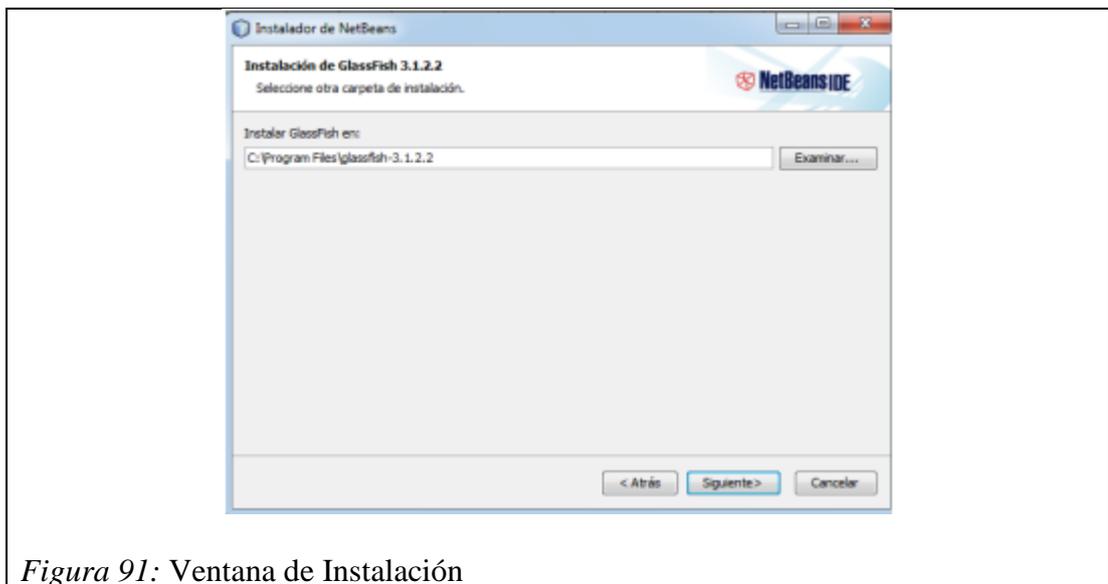


La siguiente figura indica la ruta en dónde se desea que se instale el programa.

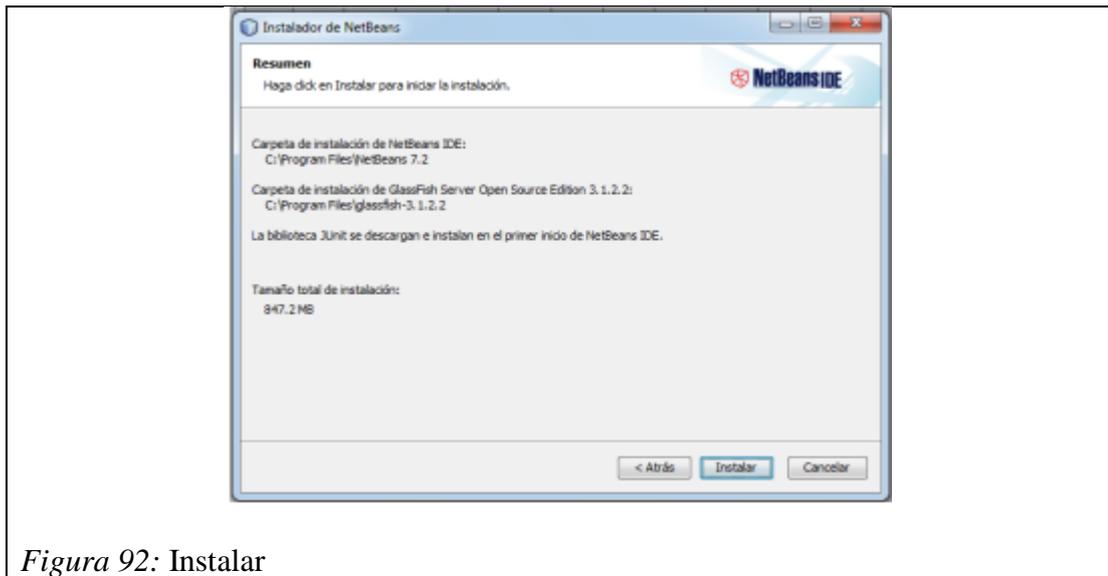
Asimismo, indicar la ruta del jdk que se tiene instalado, presionar el botón siguiente.



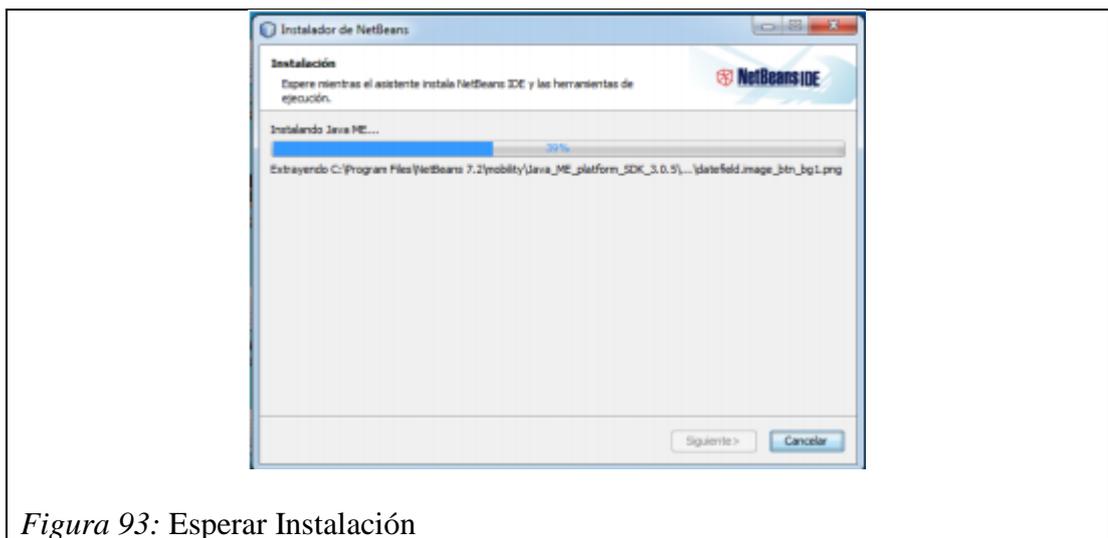
Aparece la ventana de instalación de Glassfish 3.1.2.2, presionar el botón siguiente.



Aparecerá el resumen de los elementos que se van a instalar. Presionar el botón instalar para comenzar con la instalación de NetBeans IDE 7.2.



Comenzará la instalación del programa esperar a que termine de la instalación.



Finalmente, desactiva la casilla Apoye al proyecto NetBeans proporcionando datos de uso de manera anónima y presiona el botón terminar para concluir con la instalación.



Figura 94: Concluir Instalación

Aparecerá en su escritorio una imagen donde muestra el acceso directo al programa instalado NetBeans IDE 7.2

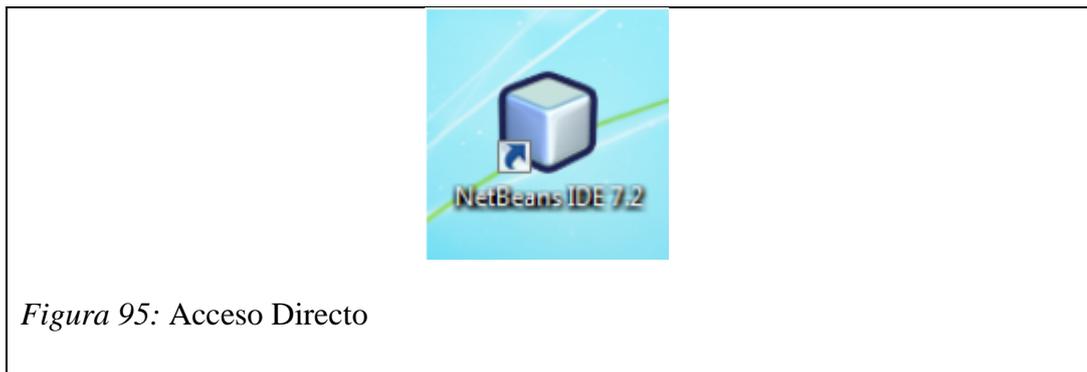


Figura 95: Acceso Directo

Para entrar a NetBeans dar doble clic sobre la imagen y nos mostrara una imagen como la de la siguiente figura.



Figura 96: Abrir Programa

Aparecerá la ventana principal de NetBeans IDE 7.2. Presiona el X que se encuentra en la pestaña Página de inicio para cerrar la ventana.

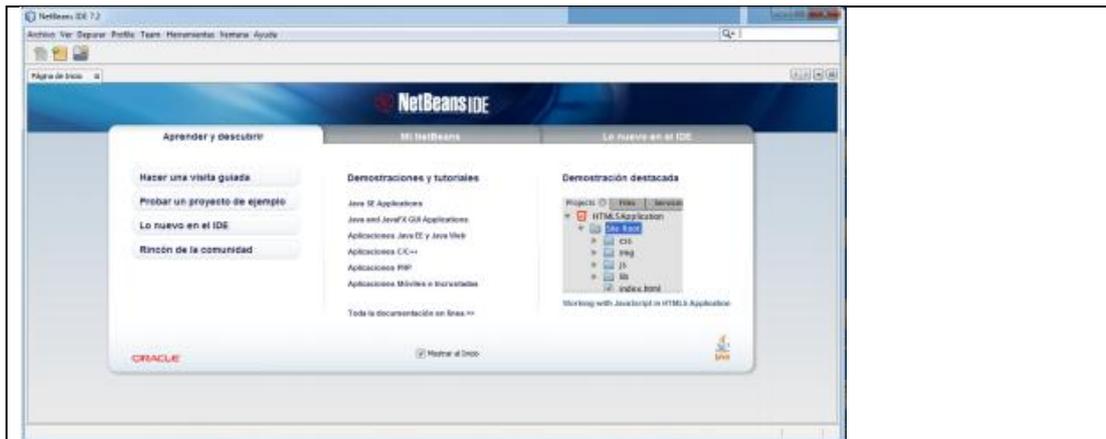


Figura 97: Ventana principal de NetBeans

Quedará la pantalla de trabajo de NetBeans IDE 7.2

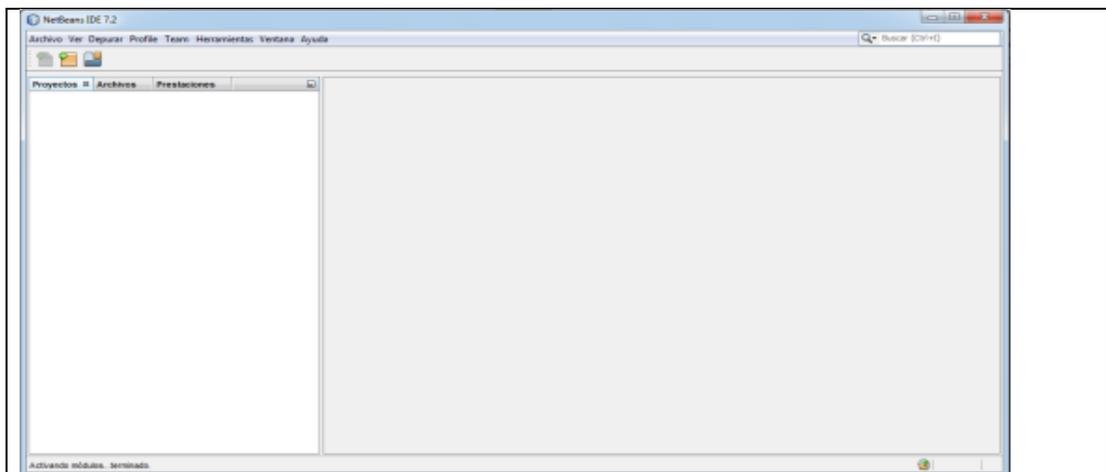


Figura 98: Programa

Ventana de ejecución del programa netbeans Id 7.2

## A.07 Script de la Base de Datos en Mysql

```
-- phpMyAdmin SQL Dump
-- version 3.5.1
-- http://www.phpmyadmin.net
-- Servidor: localhost
-- Tiempo de generación: 08-04-2014 a las 19:08:30
-- Versión del servidor: 5.5.24-log
-- Versión de PHP: 5.4.3

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";

/*!40101 SET
@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;

/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;

/*!40101 SET
@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;

/*!40101 SET NAMES utf8 */;

-- Base de datos: `facturacion_hnr`
-----

-- Estructura de tabla para la tabla `bodega`
CREATE TABLE IF NOT EXISTS `bodega` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nombre` varchar(45) NOT NULL,
  `estado` enum('ACTIVO','INACTIVO') NOT NULL,
  `descripcion` text,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;

-- Volcado de datos para la tabla `bodega`
INSERT INTO `bodega` (`id`, `nombre`, `estado`, `descripcion`) VALUES
(1, 'Matriz', 'ACTIVO', 'ninguna');
-----

-- Estructura de tabla para la tabla `categoria`
CREATE TABLE IF NOT EXISTS `categoria` (
```

```
`id` int(11) NOT NULL AUTO_INCREMENT,  
`nombre` varchar(45) NOT NULL,  
`estado` enum('ACTIVO','INACTIVO') NOT NULL,  
`descripcion` text,  
PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=7 ;  
-- Volcado de datos para la tabla `categoria`  
INSERT INTO `categoria` (`id`, `nombre`, `estado`, `descripcion`) VALUES  
(1, 'Domestico', 'ACTIVO', 'ninguna'),  
-----  
-- Estructura de tabla para la tabla `ciudad`  
CREATE TABLE IF NOT EXISTS `ciudad` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `nombre` varchar(45) NOT NULL,  
  `estado` enum('ACTIVO','INACTIVO') NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;  
-- Volcado de datos para la tabla `ciudad`  
INSERT INTO `ciudad` (`id`, `nombre`, `estado`) VALUES  
(1, 'Quito', 'ACTIVO');  
-----  
-- Estructura de tabla para la tabla `cliente`  
CREATE TABLE IF NOT EXISTS `cliente` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `nombre` varchar(45) NOT NULL,  
  `ruc` varchar(20) NOT NULL,  
  `direccion` varchar(45) NOT NULL,  
  `telefono` varchar(22) NOT NULL,  
  `telefono2` varchar(45) DEFAULT NULL,  
  `email` varchar(45) DEFAULT NULL,  
  `estado` enum('ACTIVO','INACTIVO') NOT NULL,  
  `descripcion` text,
```

```
`ciudad_id` int(11) NOT NULL,  
`id_usuario` int(11) DEFAULT NULL,  
PRIMARY KEY (`id`),  
UNIQUE KEY `ruc_UNIQUE` (`ruc`),  
KEY `fk_cliente_ciudad_idx` (`ciudad_id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;  
-- Volcado de datos para la tabla `cliente`  
INSERT INTO `cliente` (`id`, `nombre`, `ruc`, `direccion`, `telefono`, `telefono2`,  
`email`, `estado`, `descripcion`, `ciudad_id`, `id_usuario`) VALUES  
(1, 'FT Inca', '17896321456', '6 diciembre', '02456789', NULL, NULL, 'ACTIVO',  
NULL, 1, 1);  
-----  
-- Estructura de tabla para la tabla `detalle_factura`  
CREATE TABLE IF NOT EXISTS `detalle_factura` (  
`factura_id` int(11) NOT NULL,  
`producto_id` int(11) NOT NULL,  
`cantidad` decimal(10,2) NOT NULL,  
PRIMARY KEY (`factura_id`, `producto_id`),  
KEY `fk_detalle_factura_factura1_idx` (`factura_id`),  
KEY `fk_detalle_factura_producto1_idx` (`producto_id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
-----  
-- Estructura de tabla para la tabla `detalle_orden_compra`  
CREATE TABLE IF NOT EXISTS `detalle_orden_compra` (  
`orden_compra_id` int(11) NOT NULL,  
`producto_id` int(11) NOT NULL,  
`cantidad` decimal(10,2) NOT NULL,  
PRIMARY KEY (`orden_compra_id`, `producto_id`),  
KEY `fk_detalle_orden_compra_producto1_idx` (`producto_id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
-- Volcado de datos para la tabla `detalle_orden_compra`  
INSERT INTO `detalle_orden_compra` (`orden_compra_id`, `producto_id`,  
`cantidad`) VALUES
```

(3, 1, '1.00'),

```
-----  
-- Estructura de tabla para la tabla `factura`  
CREATE TABLE IF NOT EXISTS `factura` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `cliente_id` int(11) NOT NULL,  
  `usuario_id` int(11) NOT NULL,  
  `fecha_creacion` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP  
ON UPDATE CURRENT_TIMESTAMP,  
  `subtotal` decimal(10,2) DEFAULT NULL,  
  `descuento` decimal(10,2) DEFAULT NULL,  
  `iva` decimal(10,2) DEFAULT NULL,  
  `total` decimal(10,2) DEFAULT NULL,  
  `estado` enum('FACTURADO','CANCELADO','ANULADO') NOT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE KEY `id_UNIQUE` (`id`),  
  KEY `fk_factura_cliente1_idx` (`cliente_id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;  
-----
```

```
-- Estructura de tabla para la tabla `orden_compra`  
CREATE TABLE IF NOT EXISTS `orden_compra` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `cliente_id` int(11) NOT NULL,  
  `fecha_creacion` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP  
ON UPDATE CURRENT_TIMESTAMP,  
  `subtotal` decimal(10,2) DEFAULT NULL,  
  `iva` decimal(10,2) DEFAULT NULL,  
  `total` decimal(10,2) DEFAULT NULL,  
  `estado` enum('PENDIENTE','PROCESADO') NOT NULL,  
  PRIMARY KEY (`id`),  
  KEY `fk_orden_compra_cliente1_idx` (`cliente_id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;  
-- Volcado de datos para la tabla `orden_compra`
```

```
INSERT INTO `orden_compra` (`id`, `cliente_id`, `fecha_creacion`, `subtotal`, `iva`,  
`total`, `estado`) VALUES
```

```
(3, 1, '2014-04-04 16:03:53', NULL, NULL, NULL, 'PENDIENTE');
```

```
-----  
-- Estructura de tabla para la tabla `orden_requisicion`
```

```
CREATE TABLE IF NOT EXISTS `orden_requisicion` (
```

```
  `id` int(11) NOT NULL AUTO_INCREMENT,
```

```
  `proveedor_id` int(11) NOT NULL,
```

```
  `categoria_id` int(11) NOT NULL,
```

```
  `producto_id` int(11) NOT NULL,
```

```
  `cantidad` decimal(10,2) NOT NULL,
```

```
  `fecha_creacion` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP  
ON UPDATE CURRENT_TIMESTAMP,
```

```
  PRIMARY KEY (`id`),
```

```
  KEY `fk_orden_requisicion_proveedor1_idx` (`proveedor_id`),
```

```
  KEY `fk_orden_requisicion_categoria1_idx` (`categoria_id`),
```

```
  KEY `fk_orden_requisicion_producto1_idx` (`producto_id`)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

```
-----  
-- Estructura de tabla para la tabla `producto`
```

```
CREATE TABLE IF NOT EXISTS `producto` (
```

```
  `id` int(11) NOT NULL AUTO_INCREMENT,
```

```
  `nombre` varchar(45) NOT NULL,
```

```
  `foto` text,
```

```
  `cantidad` decimal(10,2) NOT NULL,
```

```
  `precio_unitario` decimal(10,2) NOT NULL,
```

```
  `percha` varchar(45) DEFAULT NULL,
```

```
  `estado` enum('ACTIVO','INACTIVO') NOT NULL,
```

```
  `descripcion` varchar(45) DEFAULT NULL,
```

```
  `proveedor_id` int(11) NOT NULL,
```

```
  `categoria_id` int(11) NOT NULL,
```

```
  `bodega_id` int(11) NOT NULL,
```

```
  `stock_minimo` decimal(10,2) NOT NULL,
```

```

PRIMARY KEY (`id`),
KEY `fk_producto_proveedor1_idx` (`proveedor_id`),
KEY `fk_producto_categoria1_idx` (`categoria_id`),
KEY `fk_producto_bodega1_idx` (`bodega_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=10 ;
-- Volcado de datos para la tabla `producto`
INSERT INTO `producto` (`id`, `nombre`, `foto`, `cantidad`, `precio_unitario`,
`percha`, `estado`, `descripcion`, `proveedor_id`, `categoria_id`, `bodega_id`,
`stock_minimo`) VALUES
(1, 'Adaptador_RedondoAPlano', '1.png', '50.00', '2.50', NULL, 'ACTIVO', 'el chufflis
ese', 1, 4, 1, '0.00'),
-----
-- Estructura de tabla para la tabla `producto_bodega`
CREATE TABLE IF NOT EXISTS `producto_bodega` (
`cantidad` decimal(10,2) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
-----
-- Estructura de tabla para la tabla `proveedor`
CREATE TABLE IF NOT EXISTS `proveedor` (
`id` int(11) NOT NULL AUTO_INCREMENT,
`nombre` varchar(45) NOT NULL,
`ruc` varchar(20) NOT NULL,
`direccion` varchar(45) NOT NULL,
`telefono` varchar(22) NOT NULL,
`telefono2` varchar(45) DEFAULT NULL,
`email` varchar(45) DEFAULT NULL,
`estado` enum('ACTIVO','INACTIVO') NOT NULL,
`descripcion` text,
`ciudad_id` int(11) NOT NULL,
PRIMARY KEY (`id`),
UNIQUE KEY `ruc_UNIQUE` (`ruc`),
KEY `fk_proveedor_ciudad1_idx` (`ciudad_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;

```

-- Volcado de datos para la tabla `proveedor`

```
INSERT INTO `proveedor` (`id`, `nombre`, `ruc`, `direccion`, `telefono`,  
`telefono2`, `email`, `estado`, `descripcion`, `ciudad_id`) VALUES  
(1, 'Codelca', '1799955536001', 'cotocollao', '2548698', NULL, NULL, 'ACTIVO',  
NULL, 1),
```

-- Restricciones para tablas volcadas

-- Filtros para la tabla `cliente`

```
ALTER TABLE `cliente`  
  
ADD CONSTRAINT `fk_cliente_ciudad` FOREIGN KEY (`ciudad_id`)  
REFERENCES `ciudad` (`id`) ON DELETE NO ACTION ON UPDATE NO  
ACTION;
```

-----  
-- Filtros para la tabla `detalle\_factura`

```
ALTER TABLE `detalle_factura`  
  
ADD CONSTRAINT `fk_detalle_factura_factura1` FOREIGN KEY (`factura_id`)  
REFERENCES `factura` (`id`) ON DELETE NO ACTION ON UPDATE NO  
ACTION,  
  
ADD CONSTRAINT `fk_detalle_factura_producto1` FOREIGN KEY  
(`producto_id`) REFERENCES `producto` (`id`) ON DELETE NO ACTION ON  
UPDATE NO ACTION;
```

-- Filtros para la tabla `detalle\_orden\_compra`

```
ALTER TABLE `detalle_orden_compra`  
  
ADD CONSTRAINT `fk_detalle_orden_compra_orden_compra1` FOREIGN KEY  
(`orden_compra_id`) REFERENCES `orden_compra` (`id`) ON DELETE NO  
ACTION ON UPDATE NO ACTION,  
  
ADD CONSTRAINT `fk_detalle_orden_compra_producto1` FOREIGN KEY  
(`producto_id`) REFERENCES `producto` (`id`) ON DELETE NO ACTION ON  
UPDATE NO ACTION;
```

-- Filtros para la tabla `factura`

```
ALTER TABLE `factura`  
  
ADD CONSTRAINT `fk_factura_cliente1` FOREIGN KEY (`cliente_id`)  
REFERENCES `cliente` (`id`) ON DELETE NO ACTION ON UPDATE NO  
ACTION;
```

-- Filtros para la tabla `orden\_compra`

```
ALTER TABLE `orden_compra`  
  
ADD CONSTRAINT `fk_orden_compra_cliente1` FOREIGN KEY (`cliente_id`)  
REFERENCES `cliente` (`id`) ON DELETE NO ACTION ON UPDATE NO  
ACTION;
```

-- Filtros para la tabla `orden\_requisicion`

```
ALTER TABLE `orden_requisicion`
```

```
ADD CONSTRAINT `fk_orden_requisicion_categoria1` FOREIGN KEY  
(`categoria_id`) REFERENCES `categoria` (`id`) ON DELETE NO ACTION ON  
UPDATE NO ACTION,
```

```
ADD CONSTRAINT `fk_orden_requisicion_producto1` FOREIGN KEY  
(`producto_id`) REFERENCES `producto` (`id`) ON DELETE NO ACTION ON  
UPDATE NO ACTION,
```

```
ADD CONSTRAINT `fk_orden_requisicion_proveedor1` FOREIGN KEY  
(`proveedor_id`) REFERENCES `proveedor` (`id`) ON DELETE NO ACTION ON  
UPDATE NO ACTION;
```

-- Filtros para la tabla `producto`

```
ALTER TABLE `producto`
```

```
ADD CONSTRAINT `fk_producto_bodega1` FOREIGN KEY (`bodega_id`)  
REFERENCES `bodega` (`id`) ON DELETE NO ACTION ON UPDATE NO  
ACTION,
```

```
ADD CONSTRAINT `fk_producto_categoria1` FOREIGN KEY (`categoria_id`)  
REFERENCES `categoria` (`id`) ON DELETE NO ACTION ON UPDATE NO  
ACTION,
```

```
ADD CONSTRAINT `fk_producto_proveedor1` FOREIGN KEY (`proveedor_id`)  
REFERENCES `proveedor` (`id`) ON DELETE NO ACTION ON UPDATE NO  
ACTION;
```

-- Filtros para la tabla `proveedor`

```
ALTER TABLE `proveedor`
```

```
ADD CONSTRAINT `fk_proveedor_ciudad1` FOREIGN KEY (`ciudad_id`)  
REFERENCES `ciudad` (`id`) ON DELETE NO ACTION ON UPDATE NO  
ACTION;
```

```
/*!40101 SET  
CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
```

```
/*!40101 SET  
CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
```

```
/*!40101 SET  
COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

## A.08 Manual Técnico

### A.08.01 Objetivo del Manual

Facilitar la estructura técnica de cada módulo, igualmente como los detalles técnicos de forma clara mediante la descripción del sistema, para que los usuarios técnicos obtengan una mayor visión de la estructura y funcionamiento del sistema.

### A.08.02 Capa de Datos

#### Conexión

```
// 'db' => array(  
//   'connectionString' => 'sqlite:' . dirname(__FILE__) . '/../data/testdrive.db',  
// ),  
// MySQL database  
'db'=>array(  
  'connectionString' => 'mysql:host=localhost;dbname=facturacion_hnr',  
  'emulatePrepare' => true,  
  'username' => 'root',  
  'password' => "",  
  'charset' => 'utf8',  
),
```

#### Login

```
<?php  
/**  
 * LoginForm class.  
 * LoginForm is the data structure for keeping  
 * user login form data. It is used by the 'login' action of 'SiteController'.  
 */  
class LoginForm extends CFormModel  
{  
    public $username;  
    public $password;  
    public $rememberMe;  
    private $_identity;  
    /**  
     * Declares the validation rules.  
     * The rules state that username and password are required,  
     * and password needs to be authenticated.  
     */
```

```
public function rules()
{
    return array(
        // username and password are required
        array('username, password', 'required'),
        // rememberMe needs to be a boolean
        array('rememberMe', 'boolean'),
        // password needs to be authenticated
        array('password', 'authenticate'),
    );
}
/**
 * Declares attribute labels.
 */
public function attributeLabels()
{
    return array(
        'rememberMe'=>'Remember me next time',
    );
}
/**
 * Authenticates the password.
 * This is the 'authenticate' validator as declared in rules().
 */
public function authenticate($attribute,$params)
{
    if(!$this->hasErrors())
    {
        $this->_identity=new UserIdentity($this->username,$this-
>password);
        if(!$this->_identity->authenticate())
            $this->addError('Incorrect username or
password.');
```

```
$this->_identity=new UserIdentity($this->username,$this-
>password);
    $this->_identity->authenticate();
    }
    if($this->_identity->errorCode===UserIdentity::ERROR_NONE)
    {
        $duration=$this->rememberMe ? 3600*24*30 : 0; // 30 days
        Yii::app()->user->login($this->_identity,$duration);
        return true;
    }
    else
        return false;
    }
}
<?php
/* @var $this SiteController */
/* @var $model LoginForm */
/* @var $form CActiveForm */
$this->pageTitle=Yii::app()->name . ' - Login';
$this->breadcrumbs=array(
    'Login',
);
?>
<h1>Login</h1>
<p>Please fill out the following form with your login credentials:</p>
<div class="form">
<?php $form=$this->beginWidget('CActiveForm', array(
    'id'=>'login-form',
    'enableClientValidation'=>true,
    'clientOptions'=>array(
        'validateOnSubmit'=>true,
    ),
)); ?>
<p class="note">Fields with <span class="required">*</span> are
required.</p>
<div class="row">
    <?php echo $form->labelEx($model,'username'); ?>
    <?php echo $form->textField($model,'username'); ?>
    <?php echo $form->error($model,'username'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model,'password'); ?>
    <?php echo $form->passwordField($model,'password'); ?>
    <?php echo $form->error($model,'password'); ?>
</div>
```

```

    <p class="hint">
        Hint: You may login with
    <kbd>demo</kbd>/<kbd>demo</kbd> or <kbd>admin</kbd>/<kbd>admin</kbd>.
    </p>
</div>

<div class="row rememberMe">
    <?php echo $form->checkBox($model,'rememberMe'); ?>
    <?php echo $form->label($model,'rememberMe'); ?>
    <?php echo $form->error($model,'rememberMe'); ?>
</div>
<div class="row buttons">
    <?php echo CHtml::submitButton('Login'); ?>
</div>
<?php $this->endWidget(); ?>
</div><!-- form -->

```

### A.08.03 Capa Lógica Del Negocio

#### Cliente

```

<?php
/**
 * This is the model base class for the table "cliente".
 * DO NOT MODIFY THIS FILE! It is automatically generated by AweCrud.
 * If any changes are necessary, you must set or override the required
 * property or method in class "Cliente".
 *
 * Columns in table "cliente" available as properties of the model,
 * followed by relations of table "cliente" available as properties of the model.
 * @property integer $id
 * @property string $nombre
 * @property string $ruc
 * @property string $direccion
 * @property string $telefono
 * @property string $telefono2
 * @property string $email
 * @property string $estado
 * @property string $descripcion
 * @property integer $ciudad_id
 * @property integer $id_usuario
 * @property Ciudad $ciudad
 * @property Factura[] $facturas
 * @property OrdenCompra[] $ordenCompras

```

```
*/
abstract class BaseCliente extends AweActiveRecord {
    public static function model($className=__CLASS__) {
        return parent::model($className);
    }
    public function tableName() {
        return 'cliente';
    }
    public static function representingColumn() {
        return 'nombre';
    }
    public function rules() {
        return array(
            array('nombre, ruc, direccion, telefono, estado, ciudad_id', 'required'),
            array('ciudad_id, id_usuario', 'numerical', 'integerOnly'=>true),
            array('email', 'email'),
            array('nombre, direccion, telefono2, email', 'length', 'max'=>45),
            array('ruc', 'length', 'max'=>20),
            array('telefono', 'length', 'max'=>22),
            array('estado', 'length', 'max'=>8),
            array('descripcion', 'safe'),
            array('estado', 'in', 'range' => array('ACTIVO','INACTIVO')), // enum,
            array('telefono2, email, descripcion, id_usuario', 'default', 'setOnEmpty' =>
true, 'value' => null),
            array('id, nombre, ruc, direccion, telefono, telefono2, email, estado,
descripcion, ciudad_id, id_usuario', 'safe', 'on'=>'search'),
        );
    }

    public function relations() {
        return array(
            'ciudad' => array(self::BELONGS_TO, 'Ciudad', 'ciudad_id'),
            'facturas' => array(self::HAS_MANY, 'Factura', 'cliente_id'),
            'ordenCompras' => array(self::HAS_MANY, 'OrdenCompra', 'cliente_id'),
        );
    }
}
/**
 * @return array customized attribute labels (name=>label)
 */
public function attributeLabels() {
    return array(
        'id' => Yii::t('app', 'ID'),
        'nombre' => Yii::t('app', 'Nombre'),
        'ruc' => Yii::t('app', 'Ruc'),
    );
}
```

```
'direccion' => Yii::t('app', 'Direccion'),
'telefono' => Yii::t('app', 'Telefono'),
'telefono2' => Yii::t('app', 'Telefono2'),
'email' => Yii::t('app', 'Email'),
'estado' => Yii::t('app', 'Estado'),
'descripcion' => Yii::t('app', 'Descripcion'),
'ciudad_id' => Yii::t('app', 'Ciudad'),
'id_usuario' => Yii::t('app', 'Id Usuario'),
'ciudad' => null,
'facturas' => null,
'ordenCompras' => null,
);
}
public function search() {
    $criteria = new CDbCriteria;
    $criteria->compare('id', $this->id);
    $criteria->compare('nombre', $this->nombre, true);
    $criteria->compare('ruc', $this->ruc, true);
    $criteria->compare('direccion', $this->direccion, true);
    $criteria->compare('telefono', $this->telefono, true);
    $criteria->compare('telefono2', $this->telefono2, true);
    $criteria->compare('email', $this->email, true);
    $criteria->compare('estado', $this->estado, true);
    $criteria->compare('descripcion', $this->descripcion, true);
    $criteria->compare('ciudad_id', $this->ciudad_id);
    $criteria->compare('id_usuario', $this->id_usuario);
    return new CActiveDataProvider($this, array(
        'criteria' => $criteria,
    ));
}
public function behaviors() {
    return array_merge(array(
    ), parent::behaviors());
}
}
```

## Producto

<?php

```
/**
 * This is the model base class for the table "producto".
 * DO NOT MODIFY THIS FILE! It is automatically generated by AweCrud.
 * If any changes are necessary, you must set or override the required
 * property or method in class "Producto".
 * Columns in table "producto" available as properties of the model,
 * followed by relations of table "producto" available as properties of the model.
 * @property integer $id
 * @property string $nombre
 * @property string $foto
 * @property string $cantidad
 * @property string $precio_unitario
 * @property string $percha
 * @property string $stock_minimo
 * @property string $estado
 * @property string $descripcion
 * @property integer $proveedor_id
 * @property integer $categoria_id
 * @property integer $bodega_id
 * @property Factura[] $facturas
 * @property OrdenCompra[] $ordenCompras
 * @property OrdenRequisicion[] $ordenRequisiciones
 * @property Proveedor $proveedor
 * @property Categoria $categoria
 * @property Bodega $bodega
 */
abstract class BaseProducto extends AweActiveRecord {
    public static function model($className=__CLASS__) {
        return parent::model($className);
    }
    public function tableName() {
        return 'producto';
    }
    public static function representingColumn() {
        return 'nombre';
    }
    public function rules() {
        return array(
            array('nombre, cantidad, precio_unitario, estado, proveedor_id, categoria_id,
bodega_id', 'required'),
            array('proveedor_id, categoria_id, bodega_id', 'numerical',
'integerOnly'=>true),
            array('nombre, percha, descripcion', 'length', 'max'=>45),
            array('cantidad, precio_unitario, stock_minimo', 'length', 'max'=>10),
```

```
array('estado', 'length', 'max'=>8),
array('foto', 'safe'),
array('estado', 'in', 'range' => array('ACTIVO','INACTIVO')), // enum,
array('foto', 'percha', 'stock_minimo', 'descripcion', 'default', 'setOnEmpty' =>
true, 'value' => null),
array('id', 'nombre', 'foto', 'cantidad', 'precio_unitario', 'percha', 'stock_minimo',
estado, 'descripcion', 'proveedor_id', 'categoria_id', 'bodega_id', 'safe', 'on'=>'search'),
);
}
public function relations() {
return array(
'facturas' => array(self::MANY_MANY, 'Factura',
'detalle_factura(producto_id, factura_id)'),
'ordenCompras' => array(self::MANY_MANY, 'OrdenCompra',
'detalle_orden_compra(producto_id, orden_compra_id)'),
'ordenRequisiciones' => array(self::HAS_MANY, 'OrdenRequisicion',
'producto_id'),
'proveedor' => array(self::BELONGS_TO, 'Proveedor', 'proveedor_id'),
'categoria' => array(self::BELONGS_TO, 'Categoria', 'categoria_id'),
'bodega' => array(self::BELONGS_TO, 'Bodega', 'bodega_id'),
);
}
/**
 * @return array customized attribute labels (name=>label)
 */
public function attributeLabels() {
return array(
'id' => Yii::t('app', 'ID'),
'nombre' => Yii::t('app', 'Nombre'),
'foto' => Yii::t('app', 'Foto'),
'cantidad' => Yii::t('app', 'Cantidad'),
'precio_unitario' => Yii::t('app', 'Precio Unitario'),
'percha' => Yii::t('app', 'Percha'),
'stock_minimo' => Yii::t('app', 'Stock Mínimo'),
'estado' => Yii::t('app', 'Estado'),
'descripcion' => Yii::t('app', 'Descripcion'),
'proveedor_id' => Yii::t('app', 'Proveedor'),
'categoria_id' => Yii::t('app', 'Categoria'),
'bodega_id' => Yii::t('app', 'Bodega'),
'facturas' => null,
'ordenCompras' => null,
'ordenRequisiciones' => null,
'proveedor' => null,
'categoria' => null,
```

```
        'bodega' => null,
    );
}
public function search() {
    $criteria = new CDbCriteria;
    $criteria->compare('id', $this->id);
    $criteria->compare('nombre', $this->nombre, true);
    $criteria->compare('foto', $this->foto, true);
    $criteria->compare('cantidad', $this->cantidad, true);
    $criteria->compare('precio_unitario', $this->precio_unitario, true);
    $criteria->compare('percha', $this->percha, true);
    $criteria->compare('stock_minimo', $this->stock_minimo, true);
    $criteria->compare('estado', $this->estado, true);
    $criteria->compare('descripcion', $this->descripcion, true);
    $criteria->compare('proveedor_id', $this->proveedor_id);
    $criteria->compare('categoria_id', $this->categoria_id);
    $criteria->compare('bodega_id', $this->bodega_id);
    return new CActiveDataProvider($this, array(
        'criteria' => $criteria,
    ));
}
public function behaviors() {
    return array_merge(array(
    ), parent::behaviors());
}
}
```

### **Proveedor**

```
<?php
```

```
/**
```

- \* This is the model base class for the table "proveedor".
- \* DO NOT MODIFY THIS FILE! It is automatically generated by AweCrud.
- \* If any changes are necessary, you must set or override the required
- \* property or method in class "Proveedor".
- \* Columns in table "proveedor" available as properties of the model,
- \* followed by relations of table "proveedor" available as properties of the model.
- \* @property integer \$id
- \* @property string \$nombre
- \* @property string \$ruc
- \* @property string \$direccion
- \* @property string \$telefono
- \* @property string \$telefono2
- \* @property string \$email
- \* @property string \$estado



```
* @property string $descripcion
* @property integer $ciudad_id
*
* @property Producto[] $productos
* @property Ciudad $ciudad
*/
abstract class BaseProveedor extends AweActiveRecord {
    public static function model($className=__CLASS__) {
        return parent::model($className);
    }
    public function tableName() {
        return 'proveedor';
    }
    public static function representingColumn() {
        return 'nombre';
    }
    public function rules() {
        return array(
            array('nombre, ruc, direccion, telefono, estado, ciudad_id', 'required'),
            array('ciudad_id', 'numerical', 'integerOnly'=>true),
            array('email', 'email'),
            array('nombre, direccion, telefono2, email', 'length', 'max'=>45),
            array('ruc', 'length', 'max'=>20),
            array('telefono', 'length', 'max'=>22),
            array('estado', 'length', 'max'=>8),
            array('descripcion', 'safe'),
            array('estado', 'in', 'range' => array('ACTIVO','INACTIVO')), // enum,
            array('telefono2, email, descripcion', 'default', 'setOnEmpty' => true, 'value'
=> null),
            array('id, nombre, ruc, direccion, telefono, telefono2, email, estado,
descripcion, ciudad_id', 'safe', 'on'=>'search'),
        );
    }
    public function relations() {
        return array(
            'productos' => array(self::HAS_MANY, 'Producto', 'proveedor_id'),
            'ciudad' => array(self::BELONGS_TO, 'Ciudad', 'ciudad_id'),
        );
    }
    /**
     * @return array customized attribute labels (name=>label)
     */
    public function attributeLabels() {
        return array(
```

```
'id' => Yii::t('app', 'ID'),
'nombre' => Yii::t('app', 'Nombre'),
'ruc' => Yii::t('app', 'Ruc'),
'direccion' => Yii::t('app', 'Direccion'),
'telefono' => Yii::t('app', 'Telefono'),
'telefono2' => Yii::t('app', 'Telefono2'),
'email' => Yii::t('app', 'Email'),
'estado' => Yii::t('app', 'Estado'),
'descripcion' => Yii::t('app', 'Descripcion'),
'ciudad_id' => Yii::t('app', 'Ciudad'),
'productos' => null,
'ciudad' => null,
);
}
public function search() {
    $criteria = new CDbCriteria;
    $criteria->compare('id', $this->id);
    $criteria->compare('nombre', $this->nombre, true);
    $criteria->compare('ruc', $this->ruc, true);
    $criteria->compare('direccion', $this->direccion, true);
    $criteria->compare('telefono', $this->telefono, true);
    $criteria->compare('telefono2', $this->telefono2, true);
    $criteria->compare('email', $this->email, true);
    $criteria->compare('estado', $this->estado, true);
    $criteria->compare('descripcion', $this->descripcion, true);
    $criteria->compare('ciudad_id', $this->ciudad_id);
    return new CActiveDataProvider($this, array(
        'criteria' => $criteria,
    ));
}
public function behaviors() {
    return array_merge(array(
    ), parent::behaviors());
}
}
```

## Categoría

```
<?php
```

```
/**
```

- \* This is the model base class for the table "categoria".
- \* DO NOT MODIFY THIS FILE! It is automatically generated by AweCrud.
- \* If any changes are necessary, you must set or override the required
- \* property or method in class "Categoria".

```
* Columns in table "categoria" available as properties of the model,
* followed by relations of table "categoria" available as properties of the model.
* @property integer $id
* @property string $nombre
* @property string $estado
* @property string $descripcion
* @property Producto[] $productos
*/
```

```
abstract class BaseCategoria extends AweActiveRecord {
    public static function model($className=__CLASS__) {
        return parent::model($className);
    }
    public function tableName() {
        return 'categoria';
    }
    public static function representingColumn() {
        return 'nombre';
    }
    public function rules() {
        return array(
            array('nombre, estado', 'required'),
            array('nombre', 'length', 'max'=>45),
            array('estado', 'length', 'max'=>8),
            array('descripcion', 'safe'),
            array('estado', 'in', 'range' => array('ACTIVO','INACTIVO')), // enum,
            array('descripcion', 'default', 'setOnEmpty' => true, 'value' => null),
            array('id, nombre, estado, descripcion', 'safe', 'on'=>'search'),
        );
    }
    public function relations() {
        return array(
            'productos' => array(self::HAS_MANY, 'Producto', 'categoria_id'),
        );
    }
}
/**
 * @return array customized attribute labels (name=>label)
 */
public function attributeLabels() {
    return array(
        'id' => Yii::t('app', 'ID'),
        'nombre' => Yii::t('app', 'Nombre'),
        'estado' => Yii::t('app', 'Estado'),
        'descripcion' => Yii::t('app', 'Descripcion'),
        'productos' => null,
    );
}
```

```
);
}
public function search() {
    $criteria = new CDbCriteria;
    $criteria->compare('id', $this->id);
    $criteria->compare('nombre', $this->nombre, true);
    $criteria->compare('estado', $this->estado, true);
    $criteria->compare('descripcion', $this->descripcion, true);
    return new CActiveDataProvider($this, array(
        'criteria' => $criteria,
    ));
}
public function behaviors() {
    return array_merge(array(
    ), parent::behaviors());
}
}
```

## Bodega

```
<?php
/**
 * This is the model base class for the table "bodega".
 * DO NOT MODIFY THIS FILE! It is automatically generated by AweCrud.
 * If any changes are necessary, you must set or override the required
 * property or method in class "Bodega".
 * Columns in table "bodega" available as properties of the model,
 * followed by relations of table "bodega" available as properties of the model.
 * @property integer $id
 * @property string $nombre
 * @property string $estado
 * @property string $descripcion
 * @property Producto[] $productos
 */
abstract class BaseBodega extends AweActiveRecord {
    public static function model($className=__CLASS__) {
        return parent::model($className);
    }
    public function tableName() {
        return 'bodega';
    }
    public static function representingColumn() {
        return 'nombre';
    }
}
```

```
}
public function rules() {
    return array(
        array('nombre, estado', 'required'),
        array('nombre', 'length', 'max'=>45),
        array('estado', 'length', 'max'=>8),
        array('descripcion', 'safe'),
        array('estado', 'in', 'range' => array('ACTIVO','INACTIVO')), // enum,
        array('descripcion', 'default', 'setOnEmpty' => true, 'value' => null),
        array('id, nombre, estado, descripcion', 'safe', 'on'=>'search'),
    );
}
public function relations() {
    return array(
        'productos' => array(self::HAS_MANY, 'Producto', 'bodega_id'),
    );
}
/**
 * @return array customized attribute labels (name=>label)
 */
public function attributeLabels() {
    return array(
        'id' => Yii::t('app', 'ID'),
        'nombre' => Yii::t('app', 'Nombre'),
        'estado' => Yii::t('app', 'Estado'),
        'descripcion' => Yii::t('app', 'Descripcion'),
        'productos' => null,
    );
}
public function search() {
    $criteria = new CDbCriteria;
    $criteria->compare('id', $this->id);
    $criteria->compare('nombre', $this->nombre, true);
    $criteria->compare('estado', $this->estado, true);
    $criteria->compare('descripcion', $this->descripcion, true);
    return new CActiveDataProvider($this, array(
        'criteria' => $criteria,
    ));
}
public function behaviors() {
    return array_merge(array(
    ), parent::behaviors());
}
}
```

## Orden de Compra

```

<?php
/**
 * This is the model base class for the table "orden_compra".
 * DO NOT MODIFY THIS FILE! It is automatically generated by AweCrud.
 * If any changes are necessary, you must set or override the required
 * property or method in class "OrdenCompra".
 *
 * Columns in table "orden_compra" available as properties of the model,
 * followed by relations of table "orden_compra" available as properties of the
 model.
 * @property integer $id
 * @property integer $cliente_id
 * @property string $fecha_creacion
 * @property string $subtotal
 * @property string $iva
 * @property string $total
 * @property string $estado
 * @property Producto[] $productos
 * @property Cliente $cliente
 */
abstract class BaseOrdenCompra extends AweActiveRecord {
    public static function model($className=__CLASS__) {
        return parent::model($className);
    }
    public function tableName() {
        return 'orden_compra';
    }
    public static function representingColumn() {
        return 'fecha_creacion';
    }
    public function rules() {
        return array(
            array('cliente_id, estado', 'required'),
            array('cliente_id', 'numerical', 'integerOnly'=>true),
            array('subtotal, iva, total', 'length', 'max'=>10),
            array('estado', 'length', 'max'=>9),
            array('estado', 'in', 'range' => array('PENDIENTE','PROCESADO')), // enum,
            array('subtotal, iva, total', 'default', 'setOnEmpty' => true, 'value' => null),
            array('id, cliente_id, fecha_creacion, subtotal, iva, total, estado', 'safe',
'on'=>'search'),
        );
    }
}

```

```
public function relations() {
    return array(
        'productos' => array(self::MANY_MANY, 'Producto',
'detalle_orden_compra(orden_compra_id, producto_id)'),
        'cliente' => array(self::BELONGS_TO, 'Cliente', 'cliente_id'),
    );
}
/**
 * @return array customized attribute labels (name=>label)
 */
public function attributeLabels() {
    return array(
        'id' => Yii::t('app', 'ID'),
        'cliente_id' => Yii::t('app', 'Cliente'),
        'fecha_creacion' => Yii::t('app', 'Fecha Creacion'),
        'subtotal' => Yii::t('app', 'Subtotal'),
        'iva' => Yii::t('app', 'Iva'),
        'total' => Yii::t('app', 'Total'),
        'estado' => Yii::t('app', 'Estado'),
        'productos' => null,
        'cliente' => null,
    );
}
public function search() {
    $criteria = new CDbCriteria;
    $criteria->compare('id', $this->id);
    $criteria->compare('cliente_id', $this->cliente_id);
    $criteria->compare('fecha_creacion', $this->fecha_creacion, true);
    $criteria->compare('subtotal', $this->subtotal, true);
    $criteria->compare('iva', $this->iva, true);
    $criteria->compare('total', $this->total, true);
    $criteria->compare('estado', $this->estado, true);
    return new CActiveDataProvider($this, array(
        'criteria' => $criteria,
    ));
}
public function behaviors() {
    return array_merge(array(
        'CTimestampBehavior' => array(
            'class' => 'zii.behaviors.CTimestampBehavior',
            'createAttribute' => 'fecha_creacion',
            'updateAttribute' => null,
            'timestampExpression' => new CDbExpression('NOW()'),
        )
    ));
}
```

```

    ), parent::behaviors());
}
}

```

### Detalle Orden de Compra

```

<?php
/**
 * This is the model base class for the table "detalle_orden_compra".
 * DO NOT MODIFY THIS FILE! It is automatically generated by AweCrud.
 * If any changes are necessary, you must set or override the required
 * property or method in class "DetalleOrdenCompra".
 * Columns in table "detalle_orden_compra" available as properties of the model,
 * and there are no model relations.
 * @property integer $orden_compra_id
 * @property integer $producto_id
 * @property string $cantidad
 */
abstract class BaseDetalleOrdenCompra extends AweActiveRecord {
    public static function model($className=__CLASS__) {
        return parent::model($className);
    }
    public function tableName() {
        return 'detalle_orden_compra';
    }
    public static function representingColumn() {
        return array(
            'orden_compra_id',
            'producto_id',
        );
    }
    public function rules() {
        return array(
            array('orden_compra_id, producto_id, cantidad', 'required'),
            array('orden_compra_id, producto_id', 'numerical', 'integerOnly'=>true),
            array('cantidad', 'length', 'max'=>10),
            array('orden_compra_id, producto_id, cantidad', 'safe', 'on'=>'search'),
        );
    }
    public function relations() {
        return array(
        );
    }
}
/**
 * @return array customized attribute labels (name=>label)

```

```
*/  
public function attributeLabels() {  
    return array(  
        'orden_compra_id' => Yii::t('app', 'Orden Compra'),  
        'producto_id' => Yii::t('app', 'Producto'),  
        'cantidad' => Yii::t('app', 'Cantidad'),  
    );  
}  
public function search() {  
    $criteria = new CDbCriteria;  
    $criteria->compare('orden_compra_id', $this->orden_compra_id);  
    $criteria->compare('producto_id', $this->producto_id);  
    $criteria->compare('cantidad', $this->cantidad, true);  
    return new CActiveDataProvider($this, array(  
        'criteria' => $criteria,  
    ));  
}  
public function behaviors() {  
    return array_merge(array(  
    ), parent::behaviors());  
}  
}
```

## Orden Requisición

```
<?php  
/**  
 * This is the model base class for the table "orden_requisicion".  
 * DO NOT MODIFY THIS FILE! It is automatically generated by AweCrud.  
 * If any changes are necessary, you must set or override the required  
 * property or method in class "OrdenRequisicion".  
 * Columns in table "orden_requisicion" available as properties of the model,  
 * followed by relations of table "orden_requisicion" available as properties of the  
 model.  
 * @property integer $id  
 * @property integer $proveedor_id
```

```

* @property integer $categoria_id
* @property integer $producto_id
* @property string $cantidad
* @property string $fecha_creacion
* @property Proveedor $proveedor
* @property Categoria $categoria
* @property Producto $producto
*/

abstract class BaseOrdenRequisicion extends AweActiveRecord {

    public static function model($className=__CLASS__) {
        return parent::model($className);
    }
    public function tableName() {
        return 'orden_requisicion';
    }
    public static function representingColumn() {
        return 'cantidad';
    }
    public function rules() {
        return array(
            array('proveedor_id, categoria_id, producto_id, cantidad', 'required'),
            array('proveedor_id, categoria_id, producto_id', 'numerical',
'integerOnly'=>true),
            array('cantidad', 'length', 'max'=>10),
            array('id, proveedor_id, categoria_id, producto_id, cantidad, fecha_creacion',
'safe', 'on'=>'search'),
        );
    }
    public function relations() {
        return array(
            'proveedor' => array(self::BELONGS_TO, 'Proveedor', 'proveedor_id'),
            'categoria' => array(self::BELONGS_TO, 'Categoria', 'categoria_id'),
            'producto' => array(self::BELONGS_TO, 'Producto', 'producto_id'),
        );
    }
}
/**
 * @return array customized attribute labels (name=>label)
 */
public function attributeLabels() {
    return array(
        'id' => Yii::t('app', 'ID'),
        'proveedor_id' => Yii::t('app', 'Proveedor'),
        'categoria_id' => Yii::t('app', 'Categoria'),
    );
}

```

```
'producto_id' => Yii::t('app', 'Producto'),
'cantidad' => Yii::t('app', 'Cantidad'),
'fecha_creacion' => Yii::t('app', 'Fecha Creacion'),
'proveedor' => null,
'categoria' => null,
'producto' => null,
);
}
public function search() {
    $criteria = new CDbCriteria;
    $criteria->compare('id', $this->id);
    $criteria->compare('proveedor_id', $this->proveedor_id);
    $criteria->compare('categoria_id', $this->categoria_id);
    $criteria->compare('producto_id', $this->producto_id);
    $criteria->compare('cantidad', $this->cantidad, true);
    $criteria->compare('fecha_creacion', $this->fecha_creacion, true);
    return new CActiveDataProvider($this, array(
        'criteria' => $criteria,
    ));
}
public function behaviors() {
    return array_merge(array(
        'CTimestampBehavior' => array(
            'class' => 'zii.behaviors.CTimestampBehavior',
            'createAttribute' => 'fecha_creacion',
            'updateAttribute' => null,
            'timestampExpression' => new CDbExpression('NOW()'),
        )
    ), parent::behaviors());
}
}
```

## Factura

```
<?php
/**
 * This is the model base class for the table "factura".
 * DO NOT MODIFY THIS FILE! It is automatically generated by AweCrud.
 * If any changes are necessary, you must set or override the required
 * property or method in class "Factura".
 * Columns in table "factura" available as properties of the model,
 * followed by relations of table "factura" available as properties of the model.
 * @property integer $id
```

```

* @property integer $cliente_id
* @property integer $usuario_id
* @property string $fecha_creacion
* @property string $subtotal
* @property string $descuento
* @property string $iva
* @property string $total
* @property string $estado
* @property Producto[] $productos
* @property Cliente $cliente
*/

abstract class BaseFactura extends AweActiveRecord {
    public static function model($className=__CLASS__) {
        return parent::model($className);
    }
    public function tableName() {
        return 'factura';
    }
    public static function representingColumn() {
        return 'fecha_creacion';
    }
    public function rules() {
        return array(
            array('cliente_id, usuario_id, estado', 'required'),
            array('cliente_id, usuario_id', 'numerical', 'integerOnly'=>true),
            array('subtotal, descuento, iva, total', 'length', 'max'=>10),
            array('estado', 'length', 'max'=>9),
            array('estado', 'in', 'range' =>
array('FACTURADO','CANCELADO','ANULADO')), // enum,
            array('subtotal, descuento, iva, total', 'default', 'setOnEmpty' => true, 'value'
=> null),
            array('id, cliente_id, usuario_id, fecha_creacion, subtotal, descuento, iva,
total, estado', 'safe', 'on'=>'search'),
        );
    }
    public function relations() {
        return array(
            'productos' => array(self::MANY_MANY, 'Producto',
'detalle_factura(factura_id, producto_id)'),
            'cliente' => array(self::BELONGS_TO, 'Cliente', 'cliente_id'),
        );
    }
    /**
     * @return array customized attribute labels (name=>label)

```

```
*/  
public function attributeLabels() {  
    return array(  
        'id' => Yii::t('app', 'ID'),  
        'cliente_id' => Yii::t('app', 'Cliente'),  
        'usuario_id' => Yii::t('app', 'Usuario'),  
        'fecha_creacion' => Yii::t('app', 'Fecha Creacion'),  
        'subtotal' => Yii::t('app', 'Subtotal'),  
        'descuento' => Yii::t('app', 'Descuento'),  
        'iva' => Yii::t('app', 'Iva'),  
        'total' => Yii::t('app', 'Total'),  
        'estado' => Yii::t('app', 'Estado'),  
        'productos' => null,  
        'cliente' => null,  
    );  
}  
public function search() {  
    $criteria = new CDbCriteria;  
    $criteria->compare('id', $this->id);  
    $criteria->compare('cliente_id', $this->cliente_id);  
    $criteria->compare('usuario_id', $this->usuario_id);  
    $criteria->compare('fecha_creacion', $this->fecha_creacion, true);  
    $criteria->compare('subtotal', $this->subtotal, true);  
    $criteria->compare('descuento', $this->descuento, true);  
    $criteria->compare('iva', $this->iva, true);  
    $criteria->compare('total', $this->total, true);  
    $criteria->compare('estado', $this->estado, true);  
    return new CActiveDataProvider($this, array(  
        'criteria' => $criteria,  
    ));  
}  
public function behaviors() {  
    return array_merge(array(  
        'CTimestampBehavior' => array(  
            'class' => 'zii.behaviors.CTimestampBehavior',  
            'createAttribute' => 'fecha_creacion',  
            'updateAttribute' => null,  
            'timestampExpression' => new CDbExpression('NOW()'),  
        )  
    ), parent::behaviors());  
}  
}
```

## Detalle Factura

```

<?php
/**
 * This is the model base class for the table "detalle_factura".
 * DO NOT MODIFY THIS FILE! It is automatically generated by AweCrud.
 * If any changes are necessary, you must set or override the required
 * property or method in class "DetalleFactura".
 * Columns in table "detalle_factura" available as properties of the model,
 * and there are no model relations.
 * @property integer $factura_id
 * @property integer $producto_id
 * @property string $cantidad
 */
abstract class BaseDetalleFactura extends AweActiveRecord {
    public static function model($className=__CLASS__) {
        return parent::model($className);
    }
    public function tableName() {
        return 'detalle_factura';
    }
    public static function representingColumn() {
        return array(
            'factura_id',
            'producto_id',
        );
    }
    public function rules() {
        return array(
            array('factura_id, producto_id, cantidad', 'required'),
            array('factura_id, producto_id', 'numerical', 'integerOnly'=>true),
            array('cantidad', 'length', 'max'=>10),
            array('factura_id, producto_id, cantidad', 'safe', 'on'=>'search'),
        );
    }
    public function relations() {
        return array(
        );
    }
}
/**
 * @return array customized attribute labels (name=>label)
 */
public function attributeLabels() {
    return array(
        'factura_id' => Yii::t('app', 'Factura'),
        'producto_id' => Yii::t('app', 'Producto'),
    );
}

```

```
        'cantidad' => Yii::t('app', 'Cantidad'),
    );
}
public function search() {
    $criteria = new CDbCriteria;
    $criteria->compare('factura_id', $this->factura_id);
    $criteria->compare('producto_id', $this->producto_id);
    $criteria->compare('cantidad', $this->cantidad, true);
    return new CActiveDataProvider($this, array(
        'criteria' => $criteria,
    ));
}
public function behaviors() {
    return array_merge(array(
    ), parent::behaviors());
}
}
```

#### A.08.04 Capa de Presentación

##### Login

```
<DOCTYPE> <html>
<head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <linkrel="stylesheet" type="text/css" href="/facturacion-
hnr/assets/d06d9a7a/css/bootstrap.css" />
    <linkrel="stylesheet" type="text/css" href="/facturacion-
hnr/assets/d06d9a7a/css/bootstrap-responsive.css" />
    <linkrel="stylesheet" type="text/css" href="/facturacion-
hnr/assets/d06d9a7a/css/bootstrap-yii.css" />
    <linkrel="stylesheet" type="text/css" href="/facturacion-
hnr/assets/d06d9a7a/css/jquery-ui-bootstrap.css" />
    <linkrel="stylesheet" type="text/css" href="/facturacion-
hnr/assets/d06d9a7a/css/bootstrap-notify.css" />
    <linkrel="stylesheet" type="text/css" href="/facturacion-
hnr/assets/51cc686f/estilos.css" />
    <scripttype="text/javascript" src="/facturacion-
hnr/assets/abcf4ddb/jquery.js"></script>
    <scripttype="text/javascript" src="/facturacion-
hnr/assets/d06d9a7a/js/bootstrap.bootbox.min.js"></script>
    <scripttype="text/javascript" src="/facturacion-
hnr/assets/d06d9a7a/js/bootstrap.notify.js"></script>
```

```
<script type="text/javascript" src="/facturacion-
hnr/assets/d06d9a7a/js/bootstrap.js"></script>
<title>HNR - Login Ui</title>
<meta content="width=device-width, initial-scale=1.0" name="viewport" />
<link rel="stylesheet" type="text/css" href="/facturacion-
hnr/themes/truulo/css/fonts/font-awesome.css" />
<link href="/facturacion-hnr/themes/truulo/css/style.css" rel="stylesheet" />
<link href="/facturacion-hnr/themes/truulo/css/style-responsive.css"
rel="stylesheet" />
<head>
  <body class="lock">
  </div>
  <div class="forgot-hint pull-right">
    <a href="/facturacion-hnr/crue/ui/pwdrec">¿Olvidó su
    contraseña?</a> <a href="/facturacion-
    hnr/crue/ui/registration">Registrarse</a> </div>
  </div>
</form> </div>
<script type="text/javascript">
/*<![CDATA[*]
jQuery(function($) {
jQuery('body').tooltip({'selector': '[rel=tooltip]'});
jQuery('body').popover({'selector': '[rel=popover]'});
});
/*]]>*/
</script>
</body>
<!-- END BODY -->
</html>
```

## Cliente

```
<?php
/** @var ClienteController $this */
/** @var Cliente $model */
/** @var AweActiveForm $form */
$form = $this->beginWidget('ext.AweCrud.components.AweActiveForm', array(
    'type' => 'horizontal',
    'id' => 'cliente-form',
    'enableAjaxValidation' => false,
```

```
'clientOptions' => array('validateOnSubmit' => true, 'validateOnChange' =>
false,),
'enableClientValidation' => true,
));?>

<div class="widget green ">
  <div class="widget-title ">
    <h4>
      <i class="icon-truck"></i><?php echo Yii::t('AweCrud.app', 'Create') . ' ' .
Cliente::label();?> </h4>
    <span class="tools">
      <a class="icon-chevron-down"></a>
    </span>
  </div>
  <div class="widget-body">
    <p class="note">
      <?php echo Yii::t('AweCrud.app', 'Fields with') ?> <span
class="required">*</span>
      <?php echo Yii::t('AweCrud.app', 'are required') ?>. </p>

      <?php echo $form->textFieldRow($model, 'nombre',
array('maxlength' => 45)) ?>

      <?php echo $form->textFieldRow($model, 'ruc', array('maxlength' =>
20)) ?>

      <?php echo $form->textFieldRow($model, 'direccion',
array('maxlength' => 45)) ?>

      <?php echo $form->textFieldRow($model, 'telefono',
array('maxlength' => 22)) ?>

      <?php echo $form->textFieldRow($model, 'telefono2',
array('maxlength' => 45)) ?>

      <?php echo $form->textFieldRow($model, 'email', array('maxlength'
=> 45)) ?>

      <?php echo $form->dropDownListRow($model, 'estado',
array('ACTIVO' => 'Activo','INACTIVO' => 'Inactivo',)) ?>

      <?php echo $form->textAreaRow($model,'descripcion',array('rows'=>3, 'cols'=>50)) ?>
```

```
<?php echo $form->dropDownListRow($model, 'ciudad_id', array("
=> ' -- Seleccione -- ') + CHtml::listData(Ciudad::model()->findAll(), 'id',
Ciudad::representingColumn())) ?>

<?php echo $form->textFieldRow($model, 'id_usuario') ?>
<div class="form-actions">
<div class="form-actions-float">
<?php $this->widget('bootstrap.widgets.TbButton', array(
    'buttonType'=>'submit',
    'type'=>'primary',
    'label'=>$model->isNewRecord ? Yii::t('AweCrud.app', 'Create') :
Yii::t('AweCrud.app', 'Save'),
    )); ?>
<?php $this->widget('bootstrap.widgets.TbButton', array(
    //'buttonType'=>'submit',
    'label'=> Yii::t('AweCrud.app', 'Cancel'),
    'htmlOptions' => array('onclick' => 'javascript:history.go(-1)')
    )); ?>
</div>
</div>
</div>
</div>
<?php $this->endWidget(); ?>
```

## Productos

```
<link rel="stylesheet" href="<?php echo Yii::app()->baseUrl; ?>/css/bootstrap-
fileupload.css" />
<?php
/** @var ProductoController $this */
/** @var Producto $model */
/** @var AweActiveForm $form */
$form = $this->beginWidget('ext.AweCrud.components.AweActiveForm', array(
    'type' => 'horizontal',
    'id' => 'producto-form',
    'htmlOptions' => array('enctype' => 'multipart/form-data'),
    'enableAjaxValidation' => false,
    'clientOptions' => array('validateOnSubmit' => true, 'validateOnChange' => false,),
    'enableClientValidation' => true,
    ));
?>
<div class="widget green ">
```

```
<div class="widget-title ">
  <h4>
    <i class="icon-truck"></i><?php echo Yii::t('AweCrud.app', 'Create') . ' '.
Producto::label(); ?> </h4>
  <span class="tools">
    <a class="icon-chevron-down"></a>
  </span>
</div>
<div class="widget-body">
  <p class="note">
    <?php echo Yii::t('AweCrud.app', 'Fields with') ?> <span
class="required">*</span>
    <?php echo Yii::t('AweCrud.app', 'are required') ?>.
  </p>
  <?php echo $form->textFieldRow($model, 'nombre', array('maxlength' => 45))
?>

  <?php // echo $form->textAreaRow($model, 'foto', array('rows' => 3, 'cols' =>
50)) ?>

  <?php echo $form->textFieldRow($model, 'cantidad', array('maxlength' => 10))
?>

  <?php echo $form->textFieldRow($model, 'precio_unitario', array('maxlength'
=> 10)) ?>

  <?php echo $form->textFieldRow($model, 'percha', array('maxlength' => 45))
?>

  <?php echo $form->textFieldRow($model, 'stock_minimo', array('maxlength'
=> 10)) ?>

  <?php echo $form->dropDownListRow($model, 'estado', array('ACTIVO' =>
'Activo', 'INACTIVO' => 'Inactivo',)) ?>

  <?php echo $form->textFieldRow($model, 'descripcion', array('maxlength' =>
45)) ?>

  <?php echo $form->dropDownListRow($model, 'proveedor_id', array(" => ' --
Seleccione -- ') + CHtml::listData(Proveedor::model()->findAll(), 'id',
Proveedor::representingColumn())) ?>
```

```
<?php echo $form->dropDownListRow($model, 'categoria_id', array(" => ' --
Seleccione -- ') + CHtml::listData(Categoria::model()->findAll(), 'id',
Categoria::representingColumn())) ?>
```

```
<?php echo $form->dropDownListRow($model, 'bodega_id', array(" => ' --
Seleccione -- ') + CHtml::listData(Bodega::model()->findAll(), 'id',
Bodega::representingColumn())) ?>
```

```
<div class="control-group">
  <label class="control-label">Image Upload</label>
  <div class="controls">
    <div data-provides="fileupload" class="fileupload fileupload-new">
      <div style="width: 200px; height: 150px;" class="fileupload-new
thumbnail">
        ">
      </div>
      <div style="max-width: 200px; max-height: 150px; line-height: 20px;"
class="fileupload-preview fileupload-exists thumbnail"></div>
      <div>
        <span class="btn btn-file"><span class="fileupload-new">Seleccione
imagen</span>
          <span class="fileupload-exists">Cambiar</span>
          <input name="Producto[foto]" type="file"
class="default"></span>
          <a data-dismiss="fileupload" class="btn fileupload-exists"
href="#">Remover</a>
        </div>
      </div>
    </div>
  </div>
</div>
<div class="form-actions">
  <div class="form-actions-float">
    <?php
    $this->widget('bootstrap.widgets.TbButton', array(
      'buttonType' => 'submit',
      'type' => 'primary',
      'label' => $model->isNewRecord ? Yii::t('AweCrud.app', 'Create') :
Yii::t('AweCrud.app', 'Save'),
    ));
  <?php
  $this->widget('bootstrap.widgets.TbButton', array(
    //'buttonType'=>'submit',
```

```
'label' => Yii::t('AweCrud.app', 'Cancel'),
'htmlOptions' => array('onclick' => 'javascript:history.go(-1)')
));
?>
</div>
</div>
</div>
</div>
<script type="text/javascript" src="<?php echo Yii::app()->baseUrl; ?>/js/bootstrap-
fileupload.js"></script>
<?php $this->endWidget(); ?>
```

### Orden de Compra

```
<?php
/** @var OrdenCompraController $this */
/** @var OrdenCompra $model */
/** @var AweActiveForm $form */
$form = $this->beginWidget('ext.AweCrud.components.AweActiveForm', array(
    'type' => 'horizontal',
    'id' => 'orden-compra-form',
    'enableAjaxValidation' => false,
    'clientOptions' => array('validateOnSubmit' => true, 'validateOnChange' =>
false,),
    'enableClientValidation' => true,
));?>

<div class="widget green ">
    <div class="widget-title ">
        <h4>
            <i class="icon-truck"></i><?php echo Yii::t('AweCrud.app', 'Create') . ' ' .
OrdenCompra::label();?> </h4>
            <span class="tools">
                <a class="icon-chevron-down"></a>
            </span>
        </div>
        <div class="widget-body">
            <p class="note">
                <?php echo Yii::t('AweCrud.app', 'Fields with') ?> <span
class="required">*</span>
                <?php echo Yii::t('AweCrud.app', 'are required') ?>. </p>
```

```
<?php echo $form->dropDownListRow($model, 'cliente_id', array("
=> ' -- Seleccione -- ') + CHtml::listData(Cliente::model()->findAll(), 'id',
Cliente::representingColumn())) ?>
```

```
<?php echo $form->textFieldRow($model, 'subtotal',
array('maxlength' => 10)) ?>
```

```
<?php echo $form->textFieldRow($model, 'iva', array('maxlength' =>
10)) ?>
```

```
<?php echo $form->textFieldRow($model, 'total', array('maxlength'
=> 10)) ?>
```

```
<?php echo $form->dropDownListRow($model, 'estado',
array('PENDIENTE' => 'Pendiente', 'PROCESADO' => 'Procesado',)) ?>
<div class="form-actions">
<div class="form-actions-float">
<?php $this->widget('bootstrap.widgets.TbButton', array(
'buttonType'=>'submit',
'type'=>'primary',
'label'=>$model->isNewRecord ? Yii::t('AweCrud.app', 'Create') :
Yii::t('AweCrud.app', 'Save'),
)); ?>
<?php $this->widget('bootstrap.widgets.TbButton', array(
//'buttonType'=>'submit',
'label'=> Yii::t('AweCrud.app', 'Cancel'),
'htmlOptions' => array('onclick' => 'javascript:history.go(-1)')
)); ?>
</div>
</div>
</div>
<?php $this->endWidget(); ?>
```

## Orden de Requisicion

```
<?php
/** @var OrdenRequisicionController $this */
/** @var OrdenRequisicion $model */
/** @var AweActiveForm $form */
$form = $this->beginWidget('ext.AweCrud.components.AweActiveForm', array(
'type' => 'horizontal',
```

```

'id' => 'orden-requisicion-form',
'enableAjaxValidation' => false,
'clientOptions' => array('validateOnSubmit' => true, 'validateOnChange' =>
false,),
'enableClientValidation' => true,
));?>

<div class="widget green ">
  <div class="widget-title ">
    <h4>
      <i class="icon-truck"></i><?php echo Yii::t('AweCrud.app', 'Create') . ' '.
OrdenRequisicion::label();?>    </h4>
    <span class="tools">
      <a class="icon-chevron-down"></a>
    </span>
  </div>
  <div class="widget-body">
    <p class="note">
      <?php echo Yii::t('AweCrud.app', 'Fields with') ?> <span
class="required">*</span>
      <?php echo Yii::t('AweCrud.app', 'are required') ?>.    </p>

      <?php echo $form->dropDownListRow($model, 'proveedor_id',
array(" => ' -- Seleccione -- ') + CHtml::listData(Proveedor::model()->findAll(), 'id',
Proveedor::representingColumn())) ?>

      <?php echo $form->dropDownListRow($model, 'categoria_id',
array(" => ' -- Seleccione -- ') + CHtml::listData(Categoria::model()->findAll(), 'id',
Categoria::representingColumn())) ?>

      <?php echo $form->dropDownListRow($model, 'producto_id', array("
=> ' -- Seleccione -- ') + CHtml::listData(Producto::model()->findAll(), 'id',
Producto::representingColumn())) ?>

      <?php echo $form->textFieldRow($model, 'cantidad',
array('maxlength' => 10)) ?>

    <div class="form-actions">
      <div class="form-actions-float">
        <?php $this->widget('bootstrap.widgets.TbButton', array(
          'buttonType'=>'submit',
          'type'=>'primary',

```

```
        'label'=>$model->isNewRecord ? Yii::t('AweCrud.app', 'Create') :
Yii::t('AweCrud.app', 'Save'),
    )); ?>
    <?php $this->widget('bootstrap.widgets.TbButton', array(
        //'buttonType'=>'submit',
        'label'=> Yii::t('AweCrud.app', 'Cancel'),
        'htmlOptions' => array('onclick' => 'javascript:history.go(-1)')
    )); ?>
    </div>
</div>
</div>
<?php $this->endWidget(); ?>
```

### Proveedor

```
<?php
/** @var ProveedorController $this */
/** @var Proveedor $model */
/** @var AweActiveForm $form */
$form = $this->beginWidget('ext.AweCrud.components.AweActiveForm', array(
    'type' => 'horizontal',
    'id' => 'proveedor-form',
    'enableAjaxValidation' => false,
    'clientOptions' => array('validateOnSubmit' => true, 'validateOnChange' =>
false,),
    'enableClientValidation' => true,
));?>

<div class="widget green ">
    <div class="widget-title ">
        <h4>
            <i class="icon-truck"></i><?php echo Yii::t('AweCrud.app', 'Create') . ' ' .
Proveedor::label();?> </h4>
        <span class="tools">
            <a class="icon-chevron-down"></a>
        </span>
    </div>
    <div class="widget-body">
        <p class="note">
            <?php echo Yii::t('AweCrud.app', 'Fields with') ?> <span
class="required">*</span>
            <?php echo Yii::t('AweCrud.app', 'are required') ?>. </p>
```

```

        <?php echo $form->textFieldRow($model, 'nombre',
array('maxlength' => 45)) ?>

        <?php echo $form->textFieldRow($model, 'ruc', array('maxlength' =>
20)) ?>

        <?php echo $form->textFieldRow($model, 'direccion',
array('maxlength' => 45)) ?>

        <?php echo $form->textFieldRow($model, 'telefono',
array('maxlength' => 22)) ?>

        <?php echo $form->textFieldRow($model, 'telefono2',
array('maxlength' => 45)) ?>

        <?php echo $form->textFieldRow($model, 'email', array('maxlength'
=> 45)) ?>

        <?php echo $form->dropDownListRow($model, 'estado',
array('ACTIVO' => 'Activo','INACTIVO' => 'Inactivo',)) ?>
        <?php echo $form-
>textAreaRow($model,'descripcion',array('rows'=>3, 'cols'=>50)) ?>

        <?php echo $form->dropDownListRow($model, 'ciudad_id', array("
=> ' -- Seleccione -- ') + CHtml::listData(Ciudad::model()->findAll(), 'id',
Ciudad::representingColumn())) ?>
        <div class="form-actions">
        <div class="form-actions-float">
            <?php $this->widget('bootstrap.widgets.TbButton', array(
                'buttonType'=>'submit',
                'type'=>'primary',
                'label'=>$model->isNewRecord ? Yii::t('AweCrud.app', 'Create') :
                Yii::t('AweCrud.app', 'Save'),
            )); ?>
            <?php $this->widget('bootstrap.widgets.TbButton', array(
                //'buttonType'=>'submit',
                'label'=> Yii::t('AweCrud.app', 'Cancel'),
                'htmlOptions' => array('onclick' => 'javascript:history.go(-1)')
            )); ?>
        </div>
    </div>
</div>

```



</div>

<?php \$this->endWidget(); ?>

## A.09 Manual de Usuario

### A.09.01 Login

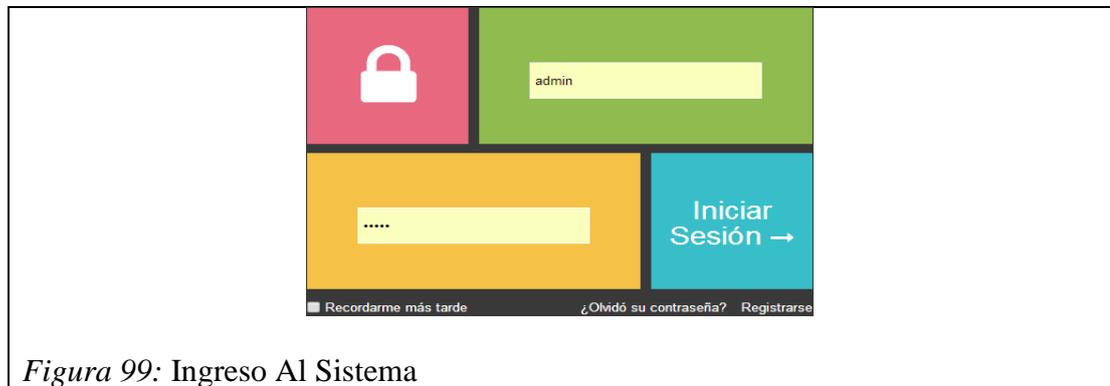


Figura 99: Ingreso Al Sistema

El cliente deberá ingresar usuario y contraseña para ingresar al sistema. Luego de haber ingresado sus datos deberá dar clic en **Iniciar Sesión**

### A.09.02 Bodega

#### Bodega

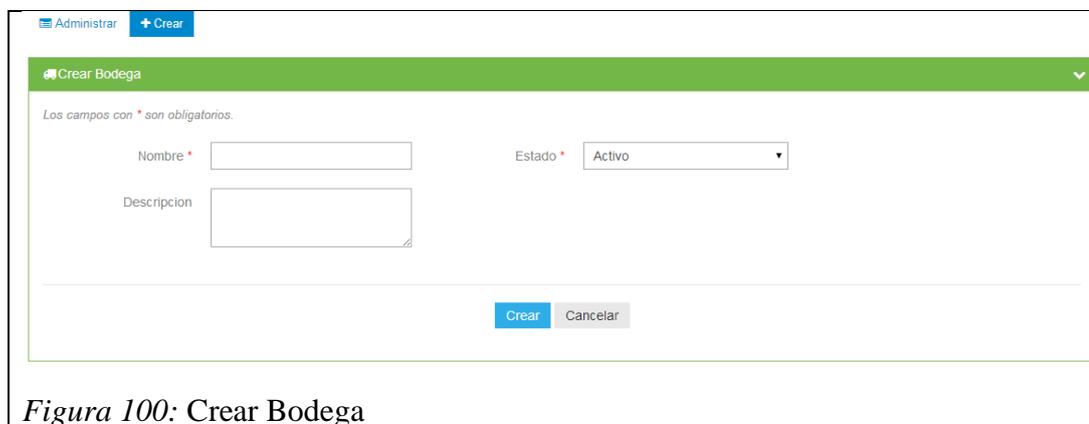


Figura 100: Crear Bodega

El usuario realizará la creación de una nueva bodega, registrara su nomdre, si esta en Actividad y escribirá una descripción de la bodega, si la empresa contará con sucursales en las cuales existan bodegas

## Administrar Bodega



Administrar + Crear

Administrar Bodegas

Desplegando 1-1 de 1 resultado.

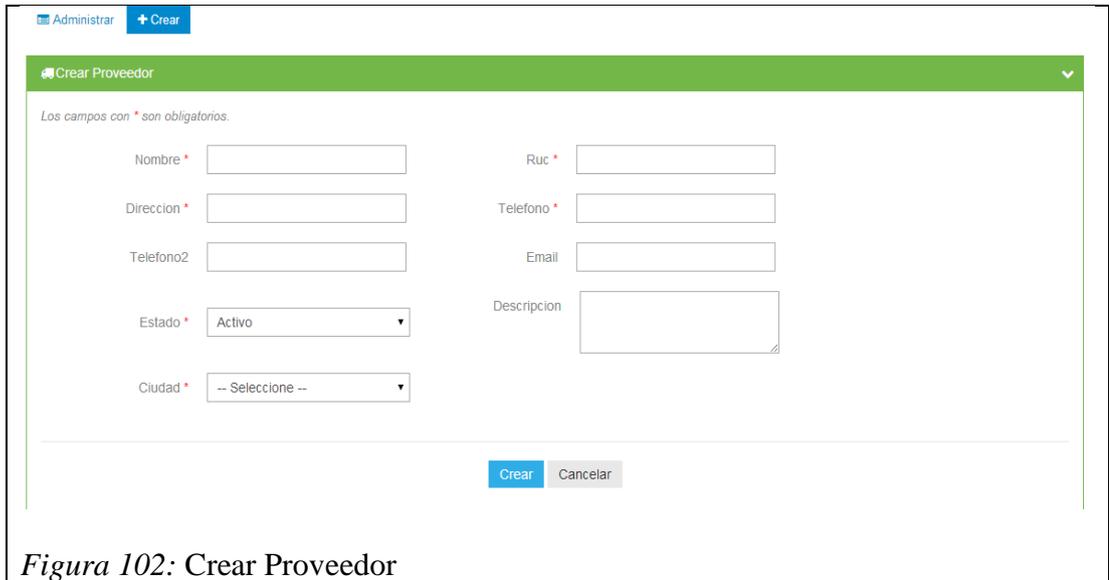
| Nombre               | Estado                              |
|----------------------|-------------------------------------|
| <input type="text"/> | <input type="text" value="Activo"/> |

Matriz

Figura 101: Administrar Bodega

El usuario realizará la administración (modificar, actualizar) de las diferentes Bodegas que existan.

## Proveedor



Administrar + Crear

Crear Proveedor

Los campos con \* son obligatorios.

Nombre \*

Ruc \*

Direccion \*

Telefono \*

Telefono2

Email

Estado \*

Descripcion

Ciudad \*

Crear Cancelar

Figura 102: Crear Proveedor

En esta ventana se realizará la creación de proveedor llenando los respectivos registros (nombre, ruc, dirección, teléfono, teléfono2, email, estado, descripción, ciudad).

## Administración Proveedor



Administración Proveedores

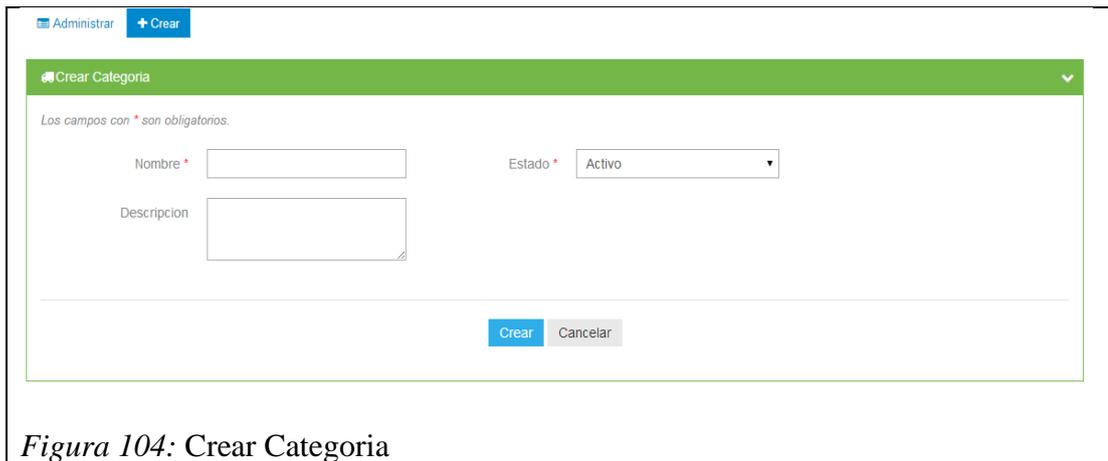
Desplegando 1-2 de 2 resultados.

| Nombre  | Ruc           | Dirección  | Teléfono   | Teléfono2 | Email |
|---------|---------------|------------|------------|-----------|-------|
| Codelca | 179995536001  | cotocollao | 2548698    |           |       |
| Tercol  | 1717547994001 | quinta     | 0057999999 |           |       |

Figura 103: Administrar Proveedor

Se podrá modificar los datos de Proveedor y/o eliminar un proveedor que ya no despache productos a la empresa.

## Categoría



Crear Categoría

Los campos con \* son obligatorios.

Nombre \*  Estado \* Activo

Descripción

Crear Cancelar

Figura 104: Crear Categoría

Aquí se deberá llenar los registros para la creación de una nueva categoría de productos si la empresa lo necesitara.

## Administración de Categoría

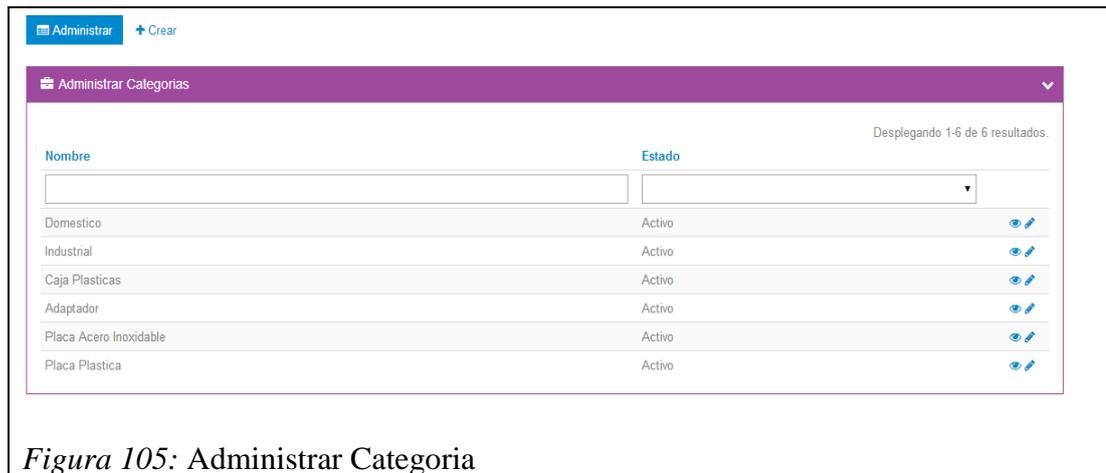


Figura 105: Administrar Categoría

En la pestaña administrar categoriase puede modificar o eliminar una categoría que ya no este en la empresa y a la vez se puede poner inactividad a una categoría que ya no se vaya a utilizarse.

## Productos

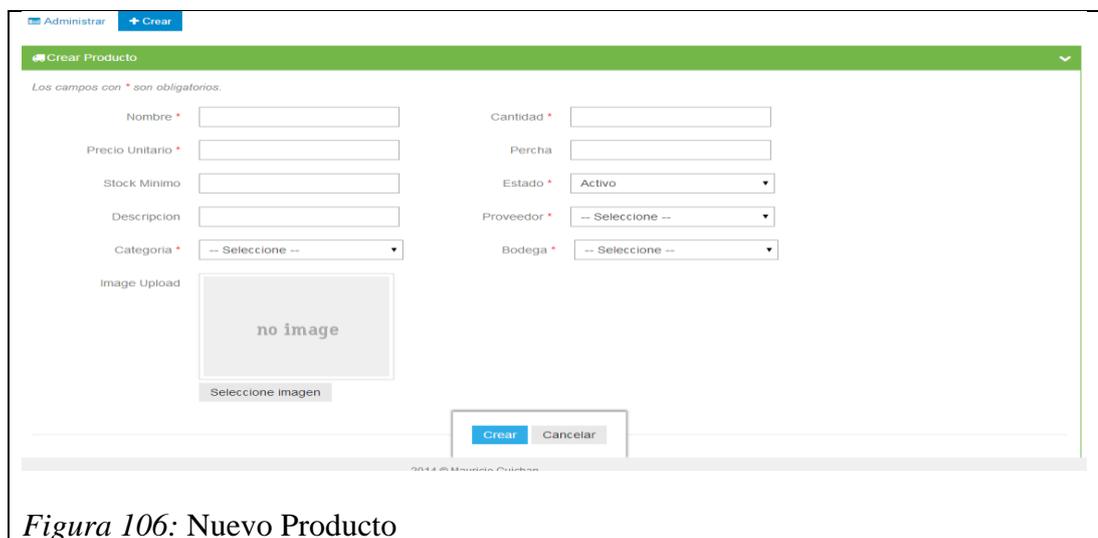


Figura 106: Nuevo Producto

El usuario registrar un nuevo producto llenando los registros necesarios para la información de dicho producto, seleccionar una imagen del producto, cotegoria, y su respectivo proveedor.

## Administrar Producto

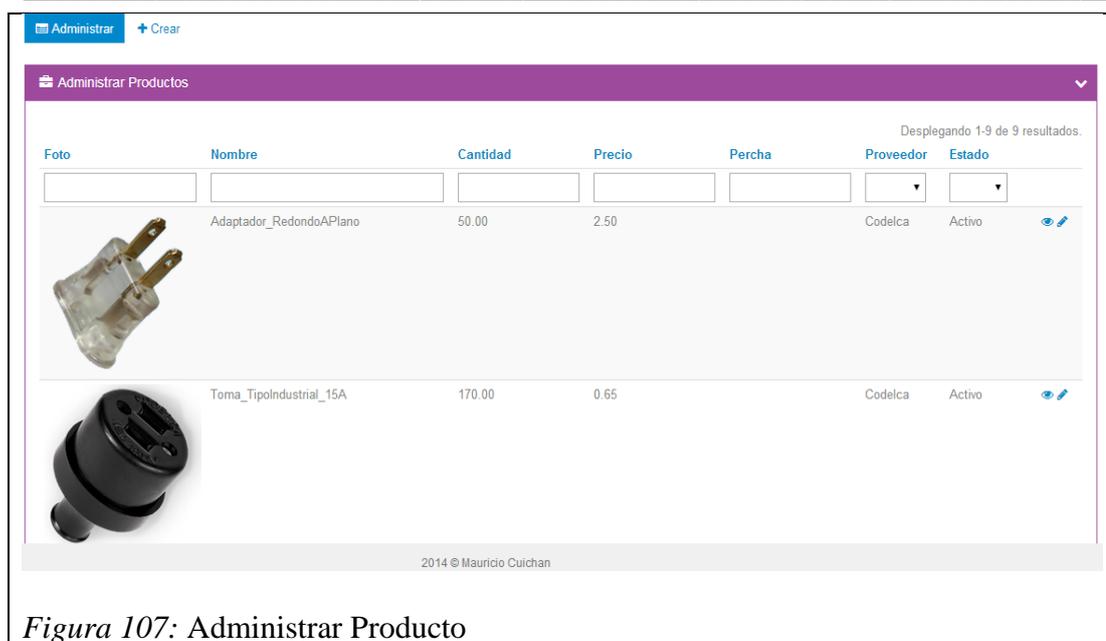


Figura 107: Administrar Producto

En la administración de productos se puede modificar y/o consultar productos que estén activos o inactivos, ver cantidad, precio y su proveedor. Además puede realizar búsquedas de acuerdo al nombre proveedor.

### A.09.03 Secretaria

#### Facturas de Proveedor

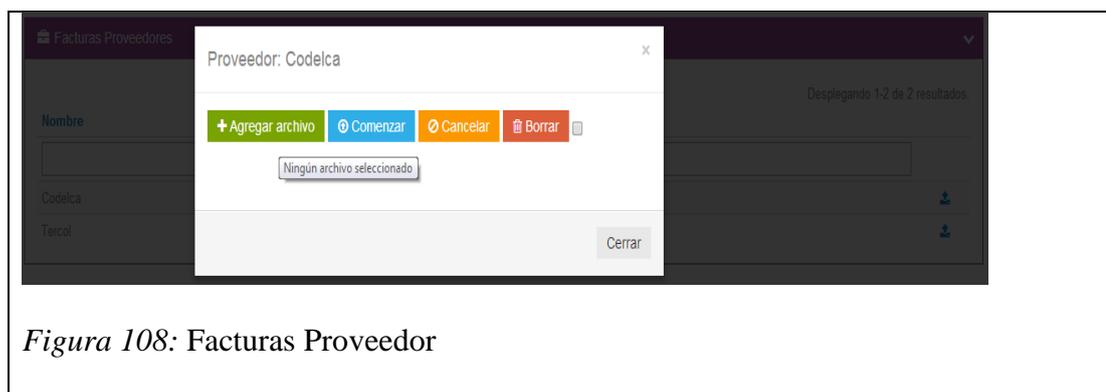
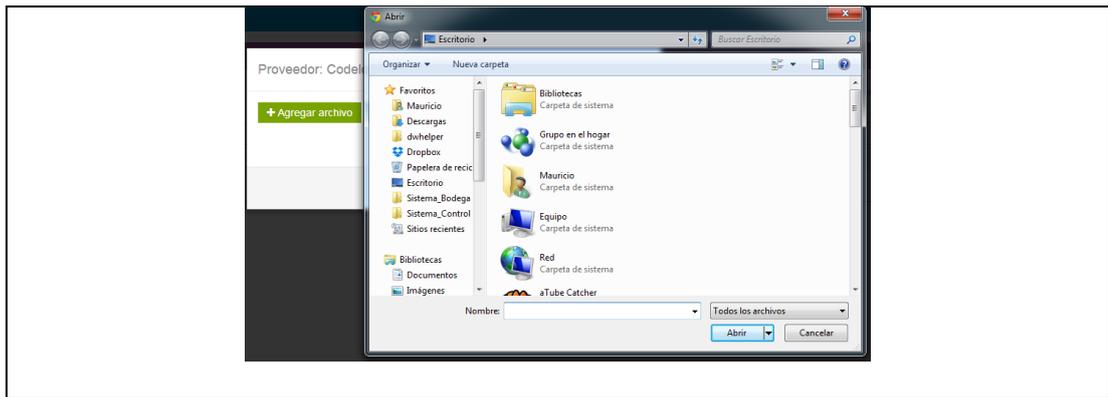
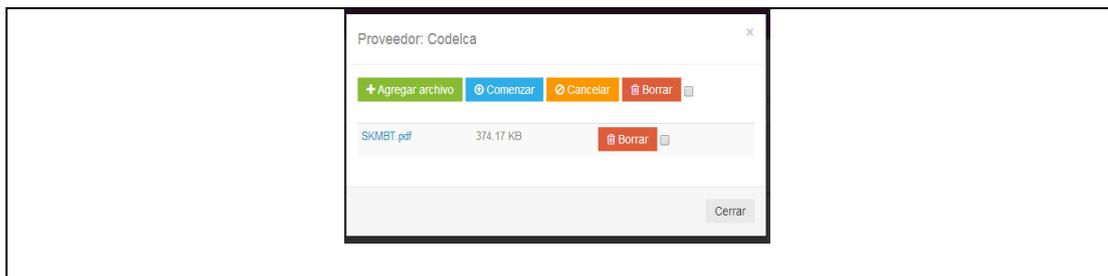


Figura 108: Facturas Proveedor

Aquí en esta pestaña el usuario podrá cargar las facturas de proveedor o todo documento referente a ese proveedor documento de pedido y observaciones en formato digital.



Se podrá seleccionar el archivo para poder subirlo al sistema y almacenarlo.



## Orden de Requisición

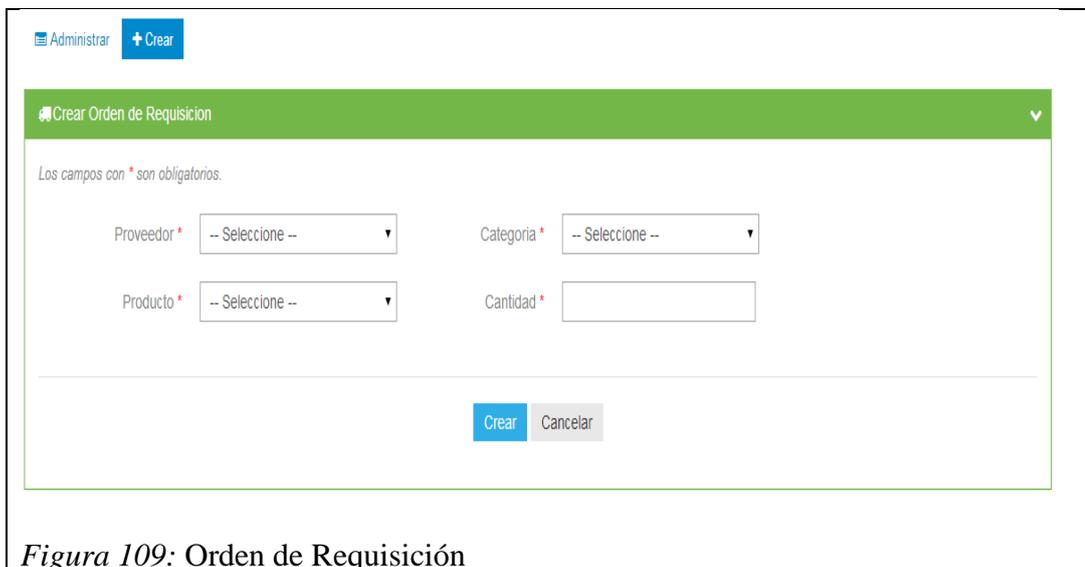
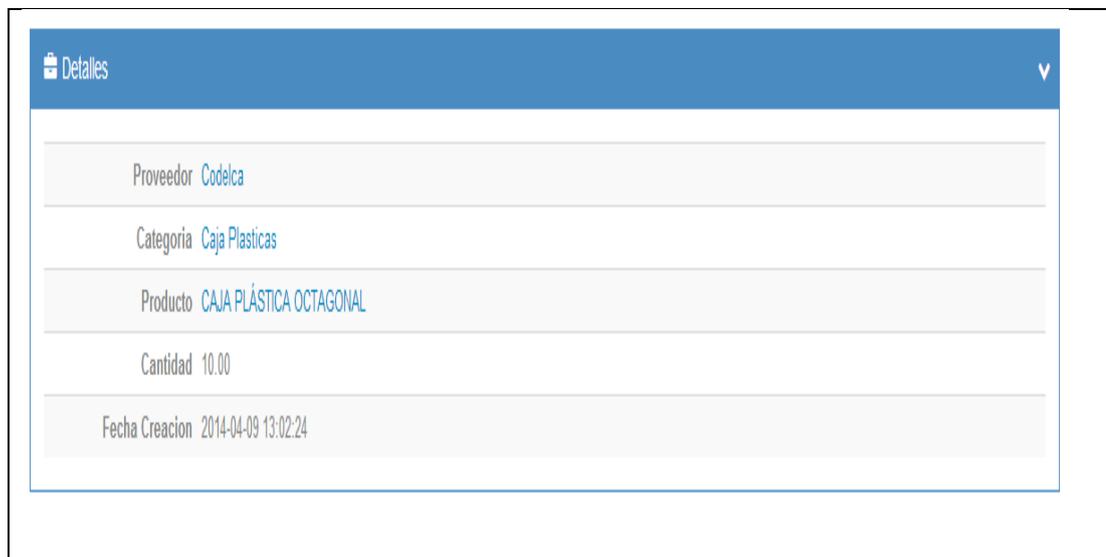


Figura 109: Orden de Requisición

En esta pestaña se puede crear una orden de requisición para realizar un pedido a los proveedores del producto, cantidad y categoría.

La cual se creará un detalle con todos los datos ingresados y la fecha de creación de dicho pedido.



### Guías de Remisión

Aquí en esta pestaña el usuario podrá cargar las guías de remisión.

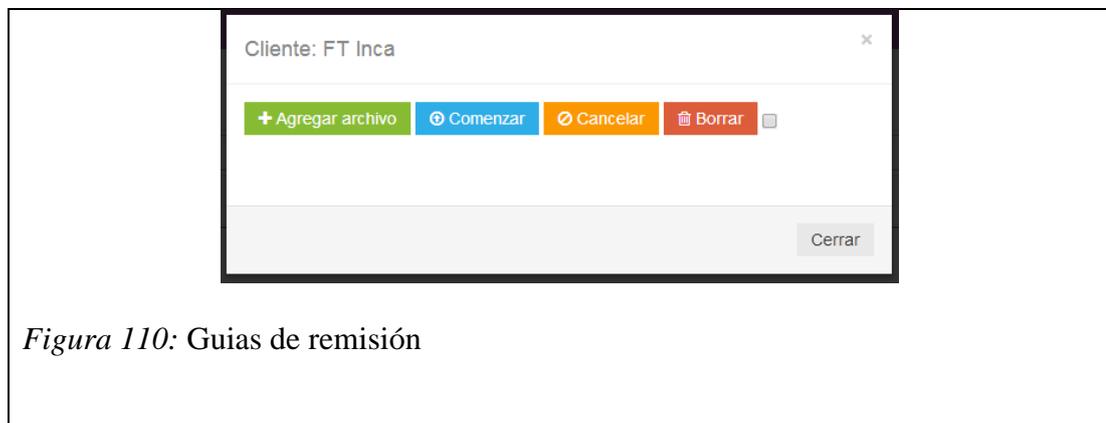
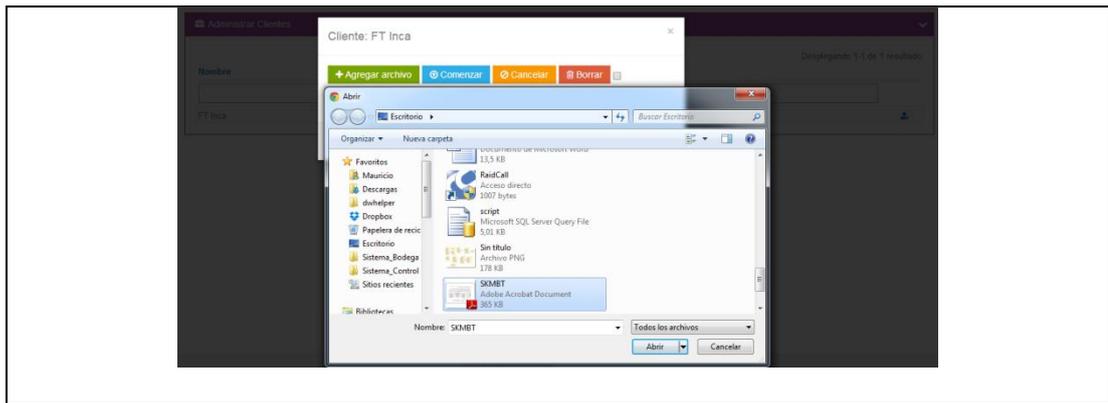
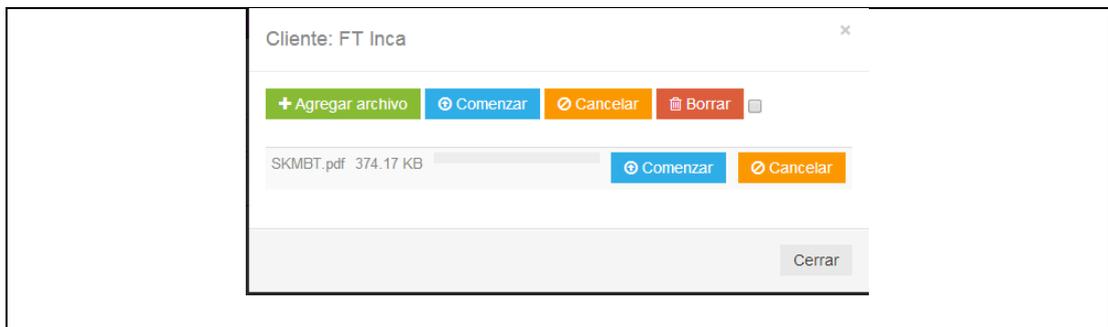


Figura 110: Guías de remisión

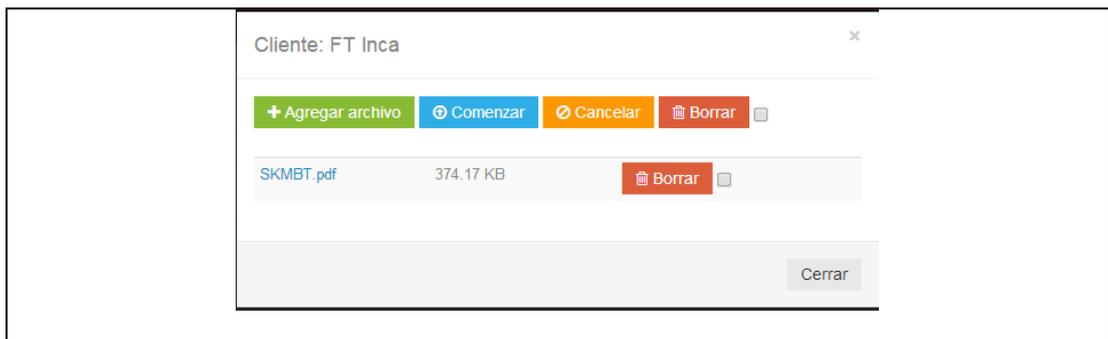
Se debe seleccionar el archivo que desee cargar.



Y se le da clic en comesar para subir la remisión seleccionada.



En esta ventana se puede borrar el archivo en caso de que el archivo no sea el documento indicado.



### A.09.04 Administrador

El administrador tendrá el acceso a todas las ventanas del sistema.

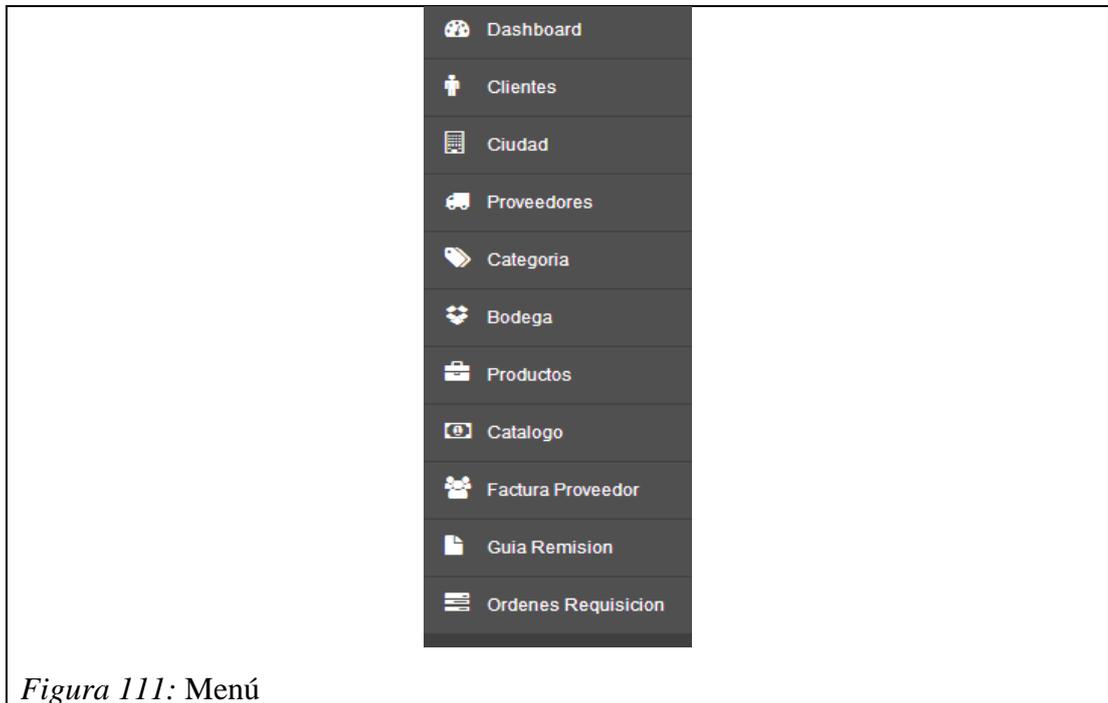


Figura 111: Menú

Además de poder verificar que usuario inicio seccion y en que momento los hizo. El administrador puede modificar los accesos a los clientes y/o empleados de la empresa.



Figura 112: Administrar Usuarios

En esta ventana puede administrar las órdenes de compras de los usuarios y a la ves modificar la cantidad de productos de su pedido.



**Figura 113:** Orden de Compras

En esta ventana puede ver que cliente realizo una orden de compra y su costo total en productos



**Figura 114:** Administrar Orden de Compras