

**Universidad Autónoma
Metropolitana Unidad Azcapotzalco
División de Ciencias Básicas e Ingeniería**

Ingeniería en Computación

Sistema de gestión de facturación y cotización

Edgar Eduardo González Herrera. Matricula 205202251

Trimestre 11-I

Asesora del proyecto

Rafaela Blanca Silva López. No Económico 17114.

Contenido

Antecedentes	6
Objetivo general	6
Objetivos particulares	6
Justificación	6
Descripción Técnica.	6
Diseño lógico del sistema.....	7
Código fuente.....	7
Clases.....	7
com.uam.cofa.actions.....	7
AdministradorFacturasAction.java.....	7
AdministrarCotizacionesAction.java.....	10
CotizacionesAction.java.....	14
DireccionesAction.java.....	20
DownloadAction.java.....	22
FacturasAction.java.....	24
InicioAction.java.....	27
ModosDeContactoAction.java.....	28
ProductosAction.java.....	31
UsuariosAction.java.....	37
ValidarUsuariosAction.java.....	40
com.uam.cofa.business.....	43
AdministradorCotizaciones.java.....	43
AdministradorDirecciones.java.....	44
AdministradorUsuarios.java.....	46
ValidadorUsuarios.java.....	46
com.uam.cofa.business.local.....	47
AdministradorFacturasImpl.java.....	47
AdministradorProductosImpl.java.....	51
AdministradorUsuariosImpl.java.....	53
Cotizador.java.....	56
Facturador.java.....	59
ValidadorUsuariosImpl.java.....	60

com.uam.cofa.data.componentesUsuario.....	61
DatosDeAccesoDTO.java.....	62
DireccionDTO.java.....	64
MedioDeContactoDTO.java.....	66
OcupacionDTO.java.....	68
UsuarioDTO.java.....	69
com.uam.cofa.data.facturasCotizaciones.....	71
CotizacionDTO.java.....	71
FacturaDTO.java.....	75
ProductoEnCotizacionDTO.java.....	79
SubtipoProductoDTO.java.....	81
TipoProductoDTO.java.....	82
MarcaDTO.java.....	85
ProductoDTO.java.....	86
ProductoEnCarrito.java.....	90
com.uam.cofa.utilities.....	93
EncriptadorPassword.java.....	93
GeneradorPDF.java.....	94
UtileriasAdministrarSistema.java.....	98
Archivos de configuración xml.....	100
cotizaciones.xml.....	100
facturas.xml.....	101
productos.xml.....	101
usuarios.xml.....	102
validacion.xml.....	103
struts.xml.....	104
spring.xml.....	105
DireccionesAction-validation.xml.....	108
web.xml.....	108
decorators.xml.....	110
WEB content.....	110
css.....	111
Style_menuDesplegable.css.....	111
style.css.....	112
Layouts	124
header_inicio.jspf.....	124
header_administrador.jspf.....	125

header_registrado.jspf.....	125
footer.jspf.....	126
internal_layout.jsp.....	126
Páginas.....	128
Cotizaciones	128
forma_generarCotizaciones.jsp.....	128
forma_seleccionarCantidad.jsp.....	131
informacionCotizacion.jsp.....	132
misCotizaciones.jsp.....	136
pantallaCotizacion.jsp.....	140
Facturas.....	142
generarFacturas.jsp.....	142
informacionDeFactura.jsp.....	143
listar.jsp.....	146
misFacturas.jsp.....	149
Productos.....	152
administrarProdcutos.jsp.....	152
forma_insertarProducto.jsp.....	153
forma_modifcarProducto.jsp.....	155
listar.jsp.....	157
listarParaModificar.jsp.....	159
Usuarios.....	161
forma_insertarCliente.jsp.....	161
forma_insertarDireccion.jsp.....	163
forma_insertarModosDeContacto.jsp.....	165
registroExitoso.jsp.....	167
forma_validarUsuario.jsp.....	168
Error.jsp.....	169
index_registrado.jsp.....	170
index.jsp.....	172
Manual del usuario	175
Ingresar al sistema	175
Registro de nuevo usuario	175
Consultar productos	175
Generar cotización.....	176
Consultar cotizaciones.....	176
Ver cotización a detalle.....	176

Cerrar cotización.....	176
Cancelar cotización.....	177
Consultar mis facturas	177
Ver detalle de factura	177
Descargar factura.....	177
Manual usuario administrador.....	177
Ingresar al sistema	177
Consultar cotizaciones.....	178
Administración de facturas.....	178
Generar facturas.....	178
Ver detalle de factura	178
Descargar factura.....	178
Administración de productos	178
Agregar producto.....	179
Modificar producto.....	179
Borrar producto	179
Manual de instalación.....	180
Requisitos de instalación.....	180
Instalación de jdk.....	180
Instalación de mysql-server.....	181
Restauración de la base de datos.....	182
Instalación servidor tomcat.....	182
Montar la aplicación web.....	182
Ligas de descarga.....	183
Licenciamiento Creative Commons.....	183
Bibliografía	183
<u>Objetivo general</u>	<u>3</u>
<u>Objetivos particulares</u>	<u>3</u>
<u>Justificación</u>	<u>3</u>
<u>Descripción Técnica.</u>	<u>3</u>
<u>Diseño lógico del sistema.....</u>	<u>4</u>
<u>Código fuente.....</u>	<u>4</u>
<u>Clases.....</u>	<u>4</u>
<u>com.uam.cofa.actions.....</u>	<u>4</u>
<u>AdministradorFacturasAction.java.....</u>	<u>4</u>
<u>AdministrarCotizacionesAction.java.....</u>	<u>7</u>
<u>CotizacionesAction.java.....</u>	<u>11</u>

<u>DireccionesAction.java</u>	17
<u>DownloadAction.java</u>	19
<u>FacturasAction.java</u>	21
<u>InicioAction.java</u>	24
<u>ModosDeContactoAction.java</u>	25
<u>ProductosAction.java</u>	28
<u>UsuariosAction.java</u>	34
<u>ValidarUsuariosAction.java</u>	37
<u>com.uam.cofa.business</u>	40
<u>AdministradorCotizaciones.java</u>	40
<u>AdministradorDirecciones.java</u>	41
<u>AdministradorUsuarios.java</u>	43
<u>ValidadorUsuarios.java</u>	43
<u>com.uam.cofa.business.local</u>	44
<u>AdministradorFacturasImpl.java</u>	44
<u>AdministradorProductosImpl.java</u>	48
<u>AdministradorUsuariosImpl.java</u>	50
<u>Cotizador.java</u>	53
<u>Facturador.java</u>	56
<u>ValidadorUsuariosImpl.java</u>	57
<u>com.uam.cofa.data.componentesUsuario</u>	58
<u>DatosDeAccesoDTO.java</u>	59
<u>DireccionDTO.java</u>	61
<u>MedioDeContactoDTO.java</u>	63
<u>OcupacionDTO.java</u>	65
<u>UsuarioDTO.java</u>	66
<u>com.uam.cofa.data.facturasCotizaciones</u>	68
<u>CotizacionDTO.java</u>	68
<u>FacturaDTO.java</u>	72
<u>ProductoEnCotizacionDTO.java</u>	76
<u>SubtipoProductoDTO.java</u>	78
<u>TipoProductoDTO.java</u>	79
<u>MarcaDTO.java</u>	82
<u>ProductoDTO.java</u>	83
<u>ProductoEnCarrito.java</u>	87
<u>com.uam.cofa.utilities</u>	90
<u>EncriptadorPassword.java</u>	90

GeneradorPDF.java	91
UtileriasAdministrarSistema.java	95
Archivos de configuración xml	97
cotizaciones.xml	97
facturas.xml	98
productos.xml	98
usuarios.xml	99
validacion.xml	100
struts.xml	101
spring.xml	102
DireccionesAction-validation.xml	105
web.xml	105
decorators.xml	107
WEB content	107
css	108
Style_menuDesplegable.css	108
style.css	109
Layouts	121
header_inicio.jspf	121
header_administrador.jspf	122
header_registrado.jspf	122
footer.jspf	123
internal_layout.jsp	123
Páginas	125
Cotizaciones	125
forma_generarCotizaciones.jsp	125
forma_seleccionarCantidad.jsp	128
informacionCotizacion.jsp	129
misCotizaciones.jsp	133
pantallaCotizacion.jsp	137
Facturas	139
generarFacturas.jsp	139
informacionDeFactura.jsp	140
listar.jsp	143
misFacturas.jsp	146
Productos	149
administrarProdcutos.jsp	149

forma_insertarProducto.jsp.....	150
forma_modifcarProducto.jsp.....	152
listar.jsp.....	154
listarParaModificar.jsp.....	156
Usuarios.....	158
forma_insertarCliente.jsp.....	158
forma_insertarDireccion.jsp.....	160
forma_insertarModosDeContacto.jsp.....	162
registroExitoso.jsp.....	164
forma_validarUsuario.jsp.....	165
Error.jsp.....	166
index_registrado.jsp.....	167
index.jsp.....	169
Manual del usuario	172
 Ingresar al sistema	172
 Registro de nuevo usuario	172
 Consultar productos	172
 Generar cotización.....	173
 Consultar cotizaciones.....	173
 Ver cotización a detalle.....	173
 Cerrar cotización.....	173
 Cancelar cotización.....	174
 Consultar mis facturas	174
 Ver detalle de factura	174
 Descargar factura.....	174
Manual usuario administrador.....	174
 Ingresar al sistema	174
 Consultar cotizaciones.....	175
 Administración de facturas.....	175
 Generar facturas.....	175
 Ver detalle de factura	175
 Descargar factura.....	175
 Administración de productos	175
 Agregar producto.....	176
 Modificar producto.....	176
 Borrar producto	176
Manual de instalación.....	177

<u>Requisitos de instalación.....</u>	<u>177</u>
<u>Instalación de jdk.....</u>	<u>177</u>
<u>Instalación de mysql-server.....</u>	<u>178</u>
<u>Restauración de la base de datos.....</u>	<u>179</u>
<u>Instalación servidor tomcat.....</u>	<u>179</u>
<u>Montar la aplicación web.....</u>	<u>179</u>
<u>Ligas de descarga.....</u>	<u>180</u>
<u>Bibliografía</u>	<u>180</u>

Antecedentes

Existe una gran variedad de proyectos que se han desarrollado para la gestión de facturación y cotización. Estos proyectos se han desarrollado en el ámbito comercial, desarrollados por instituciones educativas o desarrollados como proyectos *open source*.

Un software CRM (*Customer Relationship Management*) proporciona apoyo a la gestión de las relaciones con los clientes, a la venta y al marketing. La mayoría de los productos que tienen una funcionalidad similar a lo que queremos desarrollar se catalogan como software CRM.

Los proyectos que más se asemejan a lo que se tiene como objetivo principal en este proyecto tienen tres características esenciales que son: ser sistemas Web, *open source* y desarrollados en ambiente Java. A continuación se mencionan y describen un conjunto de proyectos que cuentan con módulos para facturación y cotización Web y además cuentan con las tres características mencionadas arriba.

- Hipergate – Es un CRM que contiene módulos para la automatización de ventas, servicio al cliente, marketing por correo electrónico, gestión de contenidos, gestión de fallos, funcionalidades para el gerente de proyecto, calendario, foros, intercambio de archivos, directorios. Basado en Java / JSP para PostgreSQL, Oracle, SQL Server.

<http://www.hipergate.org/> Fecha de consulta 18/02/2010.

- Ohioedge CRM Server es un producto de software para la gestión empresarial diseñado específicamente para medianas y grandes empresas. Contiene funcionalidades para satisfacer la gestión de ventas, generación de trabajo y cumplimiento actividades.

<http://www.ohioedge.com/> Fecha de consulta 18/02/2010.

- OFBiz – Es un proyecto de software para empresas licenciado bajo la licencia de Apache versión 2.0. Es un proyecto *open source* con funcionalidades de un CRM las cuales incluyen: gestión de catálogos de venta, gestión de pedidos (ventas y compras) y gestión de clientes.

<http://ofbiz.apache.org/> Fecha de consulta 28/02/2010

Los proyectos mencionados anteriormente en relación con el que se propone en este documento, ofrece actuar como un punto de venta Web que gestione el proceso de cotizaciones y facturación actuando de manera genérica. Es decir que el sistema sea capaz de llevar a cabo el proceso de facturación y cotización independientemente del tipo de productos o servicios con los que la PyME comercie.

Objetivo general

Desarrollar un punto de venta en ambiente Web que permita gestionar la cotización y facturación de una PyME.

Objetivos particulares

- Elaborar el diseño lógico del Sistema de Gestión de Cotizaciones y Facturación.
- Diseñar una interfaz Web para el módulo para la validación de los usuarios.
- Diseñar una interfaz Web para la gestión de cotizaciones de productos o servicios.
- Diseñar una interfaz Web para la gestión del proceso de facturación de productos o servicios.
- Diseñar una interfaz Web para el módulo de administración del sistema
- Diseñar una interfaz Web para el módulo generación de reportes
- Programar el módulo para la validación de los usuarios.
- Programar el módulo de generación automática de cotizaciones de productos o servicios.
- Programar el módulo de gestión y seguimiento del proceso de facturación.
- Programar el módulo de administración del sistema.
- Programar el módulo para la generación de reportes de movimientos realizados en el sistema.
- Elaborar el reporte final del proyecto.

Justificación

El aumento de la población que accede a Internet para consultar precios de bienes y servicios o bien realizar la compra de un producto o contratación de un servicio, obliga a las empresas que deseen aumentar su cartera de clientes a incorporar un punto de venta accesible por medio de Internet. La razón para poner en marcha este proyecto es el de reducir la complejidad en el proceso de montar un sitio Web dedicado a la cotización y facturación de productos. El software desarrollado al concluir el proyecto se encargara de generar sitios Web dedicados a facturación y cotización automáticamente a partir de una base de datos que contenga la información de los productos o servicios con los que comercialice la PyME.

La razón por la cual un Ingeniero en Computación puede realizar este proyecto es por que se requieren conocimientos sobre diseño de algoritmos, diseño y manipulación de bases de datos, diseño de aplicaciones Web dinámicas, métodos y procesos de Ingeniería del Software y condiciones para la calidad de un producto de software. El cursar el plan de estudios de la Licenciatura en Ingeniería en Computación implica la adquisición de los conocimientos anteriormente mencionados.

Como se muestra en los antecedentes existen productos enfocados en procesos para la atención a clientes como el proceso de facturación y cotización de productos, entre los productos de software mencionados anteriormente no hay uno que tenga la funcionalidad

de generar un punto de venta Web automáticamente a partir de la base de datos de productos o servicios que se le conecte y que este orientado a PyMEs.

La posible continuación del proyecto podría ser desarrollar a partir de este proyecto un software CRM-Académico con todas las funcionalidades que contienen la mayoría de ellos, más otras funcionalidades que puedan ser propuestas por nuestra comunidad universitaria, que permitiera realizar una preinscripción, inscripción y pago en línea. Podría realizarse un software *open source* CRM desarrollado por los alumnos de Ingeniería en Computación de la UAM Azcapotzalco. Cada modulo adicional podría ser contemplado como propuesta de proyecto terminal.

Descripción Técnica.

El sistema a desarrollar generara un punto de venta Web, el cual se encarga de realizar cotizaciones de productos o servicios, y en el caso de que el usuario desee la adquisición de un bien deberá llevar el seguimiento del proceso de facturación. Los productos que se cotizaran y facturan se tomara de la base de datos de productos que le conecte al sistema. Dentro de las opciones de administración del sistema se contara con opciones para agregar, modificar o eliminar productos o servicios de la base de datos. Además se podrán generar reportes previamente definidos acerca de la funcionalidad que ha tenido el sistema.

Los bloques principales de software con los que contara proyecto son:

- Módulo para la validación de usuarios.

Se encargara de permitir el acceso al sistema a usuarios administradores y a usuarios no administradores, permitiéndoles el acceso a diferentes recursos del sistema según su condición de usuario.

- Módulo para generación automática de cotizaciones de productos o servicios.

Integra la solicitud de creación de cotizaciones, seguimiento, o cancelación de la cotización.

- Módulo para la gestión y seguimiento del proceso de facturación.

Integra la solicitud de creación de facturas con base en la cotización cerrada. Se encarga del seguimiento a la facturación, pago o cancelación de factura.

- Módulo para administración del sistema

Permite la administración de los datos requeridos en la aplicación, modificación del catalogo de productos o servicios de la PyME así como la adecuación de los procesos de seguimiento del la cotización y facturación. En este modulo se darán de alta a los usuarios del sistema tanto administradores como no administradores.

- Módulo para la generación de reportes.

Se encargara de generar reportes que presenten los movimientos que los usuarios han realizado en el sistema.

Diseño lógico del sistema

Código fuente

Clases

com.uam.cofa.actions

AdministradorFacturasAction.java

```
package com.uam.cofa.actions;

import java.sql.Date;
import java.util.List;
import com.opensymphony.xwork2.ActionSupport;
import com.uam.cofa.business.AdministradorFacturas;
import com.uam.cofa.data.componentesUsuario.UsuarioDTO;
import com.uam.cofa.data.facturasCotizaciones.CotizacionDTO;
import com.uam.cofa.data.facturasCotizaciones.FacturaDTO;
import com.uam.cofa.data.facturasCotizaciones.ProductoEnCotizacionDTO;
import com.uam.cofa.utilities.GeneradorPDF;
import com.uam.cofa.business.local.Facturador;

/**
 * @author Eduardo González
 * Contiene métodos necesarios para llevar acabo el proceso de
 * facturación.
 * */

public class AdministradorFacturasAction extends ActionSupport{

    private List<CotizacionDTO> _cotizaciones;
    private AdministradorFacturas _administradorFacturas;
    private int _cotizacionesSeleccionadas[];
    private Date _fechaLimiteDePago;
    private List<FacturaDTO> _facturas;

    /**
     * Método para el control en Struts. Sirve para abrir la página
     de administración
     * de facturas.
     * @return String
     */
    public String pantallaAdministracion() {
        return "pantallaAdministracion";
    }
}
```

```

/**
 * @return String
 * Se realiza una asignación del dato miembro _facturas y
regresa una cadena
 * para el control en Struts y así mostrar las facturas.
 */
public String listar(){
    _facturas=_administradorFacturas.listar();
    return "listar";
}

/**
 * @return String
 * Se realiza una asignación del dato miembro _cotizaciones y
regresa una cadena
 * para el control en Struts y así mostrar las cotizaciones
cerradas listas para
 * facturar.
 */
public String generarFacturas(){

_cotizaciones=_administradorFacturas.listarCotizacionesCerradas();
    return "generarFacturas";
}

/**
 * @return String
 * Toma los valores seleccionados del dato miembro
_cotizacionesSeleccionadas[]
 * y en base a ellos realiza el proceso de facturación (crea
nuevas facturas en
 * la base de datos y genera sus respectivos comprobantes PDF)
 */
public String seleccionaCotizacionesAFacturar(){
    Facturador facturador =new Facturador();
    setFechaLimiteDePago(new Date(2011,05,12));
    GeneradorPDF testPDF=new GeneradorPDF();
    int num=_cotizacionesSeleccionadas.length;
    CotizacionDTO cotizacion=new CotizacionDTO();
    FacturaDTO factura=new FacturaDTO();
    FacturaDTO
ultimaFactura=_administradorFacturas.obtenUltimaFactura();
    for (int i=0;i<num;i++){

cotizacion=_administradorFacturas.buscaCotizacion(_cotizacionesSelec
cionadas[i]);
        factura=facturador.generaFactura(cotizacion,
ultimaFactura);

        ultimaFactura=_administradorFacturas.guardar(factura);
        List<ProductoEnCotizacionDTO>
productos=_administradorFacturas.listaProductosEnCotizacion(cotizacio
n.getId());
        UsuarioDTO
usuario=_administradorFacturas.buscaUsuario(ultimaFactura.getIdUsuari
o());

        testPDF.creaPDF(ultimaFactura,productos,cotizacion,usuario);
        cotizacion=new CotizacionDTO();

```

```

        factura=new FacturaDTO();
        usuario=new UsuarioDTO();
        productos.clear();
    }
    return "facturasGeneradas";
}

/*Declaración de los métodos get*/

public List<CotizacionDTO> getCotizaciones() {
    return _cotizaciones;
}

public int[] getCotizacionesSeleccionadas() {
    return _cotizacionesSeleccionadas;
}

public Date getFechaLimiteDePago() {
    return _fechaLimiteDePago;
}

public List<FacturaDTO> getFacturas() {
    return _facturas;
}

/*Delcaración de los métodos set*/

public void setCotizaciones(List<CotizacionDTO> _cotizaciones)
{
    this._cotizaciones = _cotizaciones;
}

public void setAdministradorFacturas( AdministradorFacturas
administradorFacturas ){
    _administradorFacturas = administradorFacturas;
}

public void setCotizacionesSeleccionadas(int[]
_cotizacionesSeleccionadas) {
    this._cotizacionesSeleccionadas =
_cotizacionesSeleccionadas;
}

public void setFechaLimiteDePago(Date _fechaLimiteDePago) {
    this._fechaLimiteDePago = _fechaLimiteDePago;
}

public void setFacturas(List<FacturaDTO> facturas) {
    this._facturas = facturas;
}
}

```

AdministrarCotizacionesAction.java

```
package com.uam.cofa.actions;
```

```

import com.opensymphony.xwork2.ActionSupport;
import java.util.List;
import javax.servlet.http.HttpSession;
import org.apache.struts2.ServletActionContext;
import com.uam.cofo.data.componentesUsuario.UsuarioDTO;
import com.uam.cofo.data.facturasCotizaciones.*;
import com.uam.cofo.business.*;

/**
 * @author Eduardo González
 * Contiene métodos necesarios para llevar acabo el proceso de
 cotización, así
 * como métodos para el seguimiento de las cotizaciones.
 * */

public class AdministrarCotizacionesAction extends ActionSupport{

    private List<CotizacionDTO> _cotizaciones;
    private AdministradorCotizaciones _administradorCotizaciones;
    private AdministradorFacturas _administradorFacturas;
    private List<ProductoEnCotizacionDTO> _productosEnCotizacion;
    private String _cotizacionSeleccionada;
    private CotizacionDTO _cotizacion;
    private int _tipo;
    private int _idCotizacion;

    /**
     * @return String
     * Realiza una asignación del dato miembro _cotizaciones, para
 que sean
     * mostrados en la pantalla listar (cotizaciones). Lista las
 cotizaciones
     * que se encuentren en estado Registrada.
     * */
    public String listar(){
        System.out.println("listando cotizaciones desde admin");

        _cotizaciones=_administradorCotizaciones.listarCotizaciones(1);
        return "listar";
    }

    /**
     * @return String
     * Es similar al método listar pero lista las cotizaciones del
 usuario que
     * acceda al sistema.
     * */
    public String listarPorUsuario(){
        setTipo(1);
        HttpSession session =
ServletActionContext.getRequest().getSession();
        UsuarioDTO
usuario=(UsuarioDTO)session.getAttribute("usuario");

        _cotizaciones=_administradorCotizaciones.listarCotizacionesPorUsuario
(usuario.getId(),1);
        System.out.println(_cotizaciones.size());
        return "listarMisCotizaciones";
    }
}

```



```

    /**
     * @return String
     * Es similar al método listarPorUsuario(), pero lista las
cotizaciones que
     * esten cerradas.
     */
    public String listarPorUsuarioCerradas () {
        setTipo(2);
        HttpSession session =
ServletActionContext.getRequest().getSession();
        UsuarioDTO
usuario=(UsuarioDTO)session.getAttribute("usuario");

        _cotizaciones=_administradorCotizaciones.listarCotizacionesPorUsuario
(usuario.getId(),2);
        System.out.println(_cotizaciones.size());
        return "listarMisCotizacionesCerradas";
    }

    /**
     * @return String
     * Toma el valor de el dato miembro _idCotizacion, busca la
cotización que
     * corresponda con el id y la pone estado cerrado.
     */
    public String cerrarCotizacion () {

        _cotizacion=_administradorCotizaciones.bucarCotizacion(_idCotizacion)
;
        _administradorCotizaciones.cerrarCotizacion(_cotizacion);
        return "cotizacionCerrada";
    }

    /**
     * @return String "cotizacionCerrada"
     * Toma el valor de el dato miembro _idCotizacion, busca la
cotización que
     * corresponda con el id y la pone estado cancelado.
     */
    public String cancelarCotizacion () {

        _cotizacion=_administradorCotizaciones.bucarCotizacion(_idCotizacion)
;

        _administradorCotizaciones.cancelarCotizacion(_cotizacion);
        return "cotizacionCerrada";
    }

    /**Realiza una asignación de los datos miembro _cotizacion y
     * _productosEnCotizacion para mostrar los detalles de la
cotización
     * correspondiente a _idCotizacion.
     */
    public String informacionCotizacion () {

        _cotizacion=_administradorFacturas.buscaCotizacion(_idCotizacion);

```

```

_productosEnCotizacion=_administradorFacturas.listaProductosEnCotizac
ion(_cotizacion.getId());
        return "muestraInformacionCotizacion";
    }

    /*Metodos get*/

    public List<CotizacionDTO> getCotizaciones() {
        return _cotizaciones;
    }

    public String getCotizacionSeleccionada() {
        return _cotizacionSeleccionada;
    }

    public int getTipo() {
        return _tipo;
    }

    public CotizacionDTO getCotizacion() {
        return _cotizacion;
    }

    public List<ProductoEnCotizacionDTO> getProductosEnCotizacion()
{
        return _productosEnCotizacion;
    }

    public int getIdCotizacion() {
        return _idCotizacion;
    }

    /*Metodos set*/

    public void setCotizaciones(List<CotizacionDTO> _cotizaciones)
{
        this._cotizaciones = _cotizaciones;
    }

    public void
setAdministradorCotizaciones( AdministradorCotizaciones
administradorCotizaciones ){
        _administradorCotizaciones = administradorCotizaciones;
    }

    public void setCotizacionSeleccionada(String
cotizacionSeleccionada) {
        this._cotizacionSeleccionada = cotizacionSeleccionada;
    }

    public void setTipo(int tipo) {
        this._tipo = tipo;
    }

    public void setAdministradorFacturas( AdministradorFacturas
administradorFacturas ){

```

```

        _administradorFacturas = administradorFacturas;
    }

    public void setCotizacion(CotizacionDTO _cotizacion) {
        this._cotizacion = _cotizacion;
    }

    public void setProductosEnCotizacion(
        List<ProductoEnCotizacionDTO>
        _productosEnCotizacion) {
        this._productosEnCotizacion = _productosEnCotizacion;
    }

    public void setIdCotizacion(int _idCotizacion) {
        this._idCotizacion = _idCotizacion;
    }
}

```

CotizacionesAction.java

```

Package com.uam.cofa.actions;

import java.util.*;

import com.opensymphony.xwork2.ActionSupport;
import com.uam.cofa.business.*;
import com.uam.cofa.business.local.*;
import com.uam.cofa.data.productosServicios.*;
import com.uam.cofa.data.facturasCotizaciones.*;
import com.uam.cofa.data.componentesUsuario.*;
import javax.servlet.http.HttpSession;
import org.apache.struts2.ServletActionContext;

/**
 * @author Eduardo González
 * Contiene métodos necesarios para llevar acabo el proceso de
 registrar una
 * cotización.
 * */

public class CotizacionesAction extends ActionSupport{

    private List<SubtipoProductoDTO> _listaSubtiposProductos;
    private List<TipoProductoDTO> _listaTiposProductos;
    private List<ProductoDTO> _productos = new
ArrayList<ProductoDTO>();
    private AdministradorProductos _administradorProductos;
    private String _id;
    private String _productoSeleccionado;
    private LinkedList<ProductoEnCarrito>
_listasProductosCotizacion;
    private String [] _cantidadesDeProductos;
    private CotizacionDTO _cotizacion;
    private List<CotizacionDTO> _cotizaciones;
    private AdministradorCotizaciones _administradorCotizaciones;
    private int _idProducto;

```

```

/**
 * @return String
 *Método para el control en Struts. Sirve para abrir la página
 *forma_generarCotizaciones.
 * */
public String forma_generarCotizaciones () {
    setListaTiposProductos ();
    setListaSubtiposProductos ();
    setListaProductosCotizacion ();
    _productos = _administradorProductos.listarProductos ();
    int k=1;
    if(k==1)
        return "forma_generarCotizaciones";
    else{
ingresar ");
        return "error";
    }
}

/**
 * @return String
 *Método para el control en Struts. Sirve para abrir la página
 *forma_seleccionarCantidad donde se selecciona cuántos
productos se
 *desean de los agregados en el carritoDeCompras.
 * */
public String forma_seleccionarCantidad () {

    _listaProductosCotizacion=obtenerCarrito ().getListaProductos ();
        return "forma_seleccionarCantidad";
}

/**
 * @return String
 * Realiza una asignación del dato miembro _productos para
mostrar los productos
 * en la página forma_generarCotizaciones
 * */
public String listarPorTipo () {
    setListaTiposProductos ();
    setListaSubtiposProductos ();
    setListaProductosCotizacion ();
    _productos =
_administradorProductos.listarProductosPorTipo (_id);
        return "forma_generarCotizaciones";
}

/**
 * @return String
 * Busca el producto correspondiente al _idProducto y lo agrega
al
 * carrito de compras guardado como variable se sesión.
 * */
public String agregarProductoACarrito () {
    CarritoDeCompras carritoDeCompras=obtenerCarrito ();
    ProductoDTO
aux=_administradorCotizaciones.buscaProducto (_idProducto);
    carritoDeCompras.agregar (aux);
}

```

```

        return "agregadoACarrito";
    }

    /**
     * @return String
     * Busca el producto correspondiente al _idProducto y lo
remueve del
     * carrito de compras guardado como variable de sesión.
     */
    public String removerProductoDeCarrito() {
        CarritoDeCompras carritoDeCompras=obtenerCarrito();
        ProductoDTO
aux=_administradorCotizaciones.buscaProducto(_idProducto);
        carritoDeCompras.remover(aux);
        return "removidoDeCarrito";
    }

    /**
     * @return String
     * Obtiene los valores de el dato miembro
_cantidadesDeProductos[] y lo
     * relaciona con el producto correspondiente en el carrito de
compras.
     */
    public String obtenerCantidadDeProductos() {
        Cotizador cotizador=new Cotizador();
        _cotizacion=new CotizacionDTO();

        _listaProductosCotizacion=obtenerCarrito().getListaProductos();
        int size=_cantidadesDeProductos.length;
        CarritoDeCompras carrito=obtenerCarrito();
        for (int i=0;i<size;i++){
            carrito.estableceCantidad(i,new
Integer(_cantidadesDeProductos[i]));
        }

        _cotizacion=cotizador.cotiza(_listaProductosCotizacion,_cotizacion);
        return "pantallaCotizacion";
    }

    /**
     * @return String
     * Realiza una asignación del dato miembro _cotizaciones, para
que sean
     * mostrados en la pantalla listar (cotizaciones). Lista las
cotizaciones
     * que se encuentren en estado Registrada.
     */
    public String listarPorUsuario() {
        HttpSession session =
ServletActionContext.getRequest().getSession();
        UsuarioDTO
usuario=(UsuarioDTO)session.getAttribute("usuario");
        //se consultan las cotizaciones registradas

        _cotizaciones=_administradorCotizaciones.listarCotizacionesPorUsuario
(usuario.getId(),1);
        System.out.println(_cotizaciones.size());
        return "listarMisCotizaciones";
    }

```

```

    }

    /**
     * @return String
     * En este método se toman los productos guardados en el
     carrito de compras,
     * se le envia al objeto Cotizador la lista de productos los
     cotiza y la guarda
     * en la base de datos.
     * */
    public String guardarCotizacion() {
        HttpSession session =
ServletActionContext.getRequest().getSession();
        UsuarioDTO
usuario=(UsuarioDTO)session.getAttribute("usuario");
        _cotizacion=new CotizacionDTO();

        _listaProductosCotizacion=obtenerCarrito().getListaProductos();
        Cotizador cotizador=new Cotizador();

        _cotizacion=cotizador.cotiza(_listaProductosCotizacion,_cotizacion);
        _cotizacion.setIdUsuario(usuario.getId());

        _cotizacion=_administradorCotizaciones.insertar(_cotizacion);

        _administradorCotizaciones.guardarProductosDeCotizacion(obtenerCarrito(),_cotizacion.getId());
        vaciarCarrito();
        return "cotizacionGuardada";
    }

    /**
     * @return CarritoDeCompras
     * Regresa el objeto carrito de compras guardado en sesión en
     el momento de
     * logearse.
     * */
    public CarritoDeCompras obtenerCarrito() {
        HttpSession session =
ServletActionContext.getRequest().getSession();
        CarritoDeCompras
carritoDeCompras=(CarritoDeCompras)session.getAttribute("carritoDeCompras");
        return carritoDeCompras;
    }

    /**
     * Se obtiene el objeto CarritoDeCompras de sesión y se realiza
     una asignación
     * a un carrito de compras nuevo.
     * */
    public void vaciarCarrito() {
        HttpSession session =
ServletActionContext.getRequest().getSession();
        CarritoDeCompras carritoDeCompras=new CarritoDeCompras();

        session.setAttribute("carritoDeCompras",carritoDeCompras);
    }

```

```

    /*Metodos get*/

    public List<SubtipoProductoDTO>
getListaSubtiposProductos() {
        return _listaSubtiposProductos;
    }

    public List<TipoProductoDTO> getListaTiposProductos() {
        return _listaTiposProductos;
    }

    public String getId() {
        return _id;
    }

    public List<ProductoDTO> getProductos()
    {
        return _productos;
    }

    public String getProductoSeleccionado() {
        return _productoSeleccionado;
    }

    public String[] getCantidadesDeProductos() {
        return _cantidadesDeProductos;
    }

    public CotizacionDTO getCotizacion() {
        return _cotizacion;
    }

    public List<CotizacionDTO> getCotizaciones() {
        return _cotizaciones;
    }

    public int getIdProducto() {
        return _idProducto;
    }

    public LinkedList<ProductoEnCarrito>
getListaProductosCotizacion() {
        return _listaProductosCotizacion;
    }

    /*Metodos set*/

    public void setId(String _id) {
        this._id = _id;
    }

    public void
setAdministradorProductos( AdministradorProductos
administradorProductos )
    {
        _administradorProductos = administradorProductos;
    }

```

```

        public void setListaSubtiposProductos () {
            this._listaSubtiposProductos =
            _administradorProductos.consultaSubtiposDeProducto ();
        }

        public void setListaTiposProductos () {
            this._listaTiposProductos =
            _administradorProductos.consultaTiposDeProducto ();
        }

        public void setProductos ( List<ProductoDTO> productos )
        {
            _productos = productos;
        }

        public void setProductoSeleccionado (String
        _productoSeleccionado) {
            this._productoSeleccionado =
            _productoSeleccionado;
        }

        public void setCantidadesDeProductos (String[]
        cantidadesDeProductos) {
            this._cantidadesDeProductos =
            cantidadesDeProductos;
        }

        public void setCotizacion (CotizacionDTO cotizacion) {
            this._cotizacion = cotizacion;
        }

        public void setCotizaciones (List<CotizacionDTO>
        _cotizaciones) {
            this._cotizaciones = _cotizaciones;
        }

        public void
        setAdministradorCotizaciones ( AdministradorCotizaciones
        administradorCotizaciones ) {
            _administradorCotizaciones =
            administradorCotizaciones;
        }

        public void setIdProducto (int _idProducto) {
            this._idProducto = _idProducto;
        }

        public void setListaProductosCotizacion () {
            CarritoDeCompras
            carritoDeCompras=obtenerCarrito ();
            this._listaProductosCotizacion =
            carritoDeCompras.getListaProductos ();
        }
    }

```

DireccionesAction.java

```
package com.uam.cofa.actions;

import com.uam.cofa.business.AdministradorDirecciones;
import com.uam.cofa.data.componentesUsuario.DireccionDTO;
import com.opensymphony.xwork2.ActionSupport;
import java.util.ArrayList;
import java.util.List;

import javax.servlet.http.HttpSession;

import org.apache.struts2.ServletActionContext;

/**
 * @author Eduardo Gonzalez
 */

public class DireccionesAction extends ActionSupport{

    List<DireccionDTO> _direcciones = new
ArrayList<DireccionDTO>();
    private AdministradorDirecciones _administradorDirecciones;
    private DireccionDTO _direccion;

    /**
     * @return String
     * Se realiza una asignación del dato miembro _direcciones para
ser
     * mostradas en la página listarDirecciones
     */
    public String listar(){
        _direcciones =
_administradorDirecciones.listarDirecciones();
        System.out.println(_direcciones.size());
        return "listar";
    }

    /**
     * @return String
     * Guarda el objeto _direccion en sesión para su posterior
guardado en la
     * base de datos al terminar el registro de usuario.
     */
    public String insertarDireccion(){
        HttpSession session =
ServletActionContext.getRequest().getSession();
        session.setAttribute("direccionNueva",_direccion);
        return "direccionInsertada";
    }

    /*Declaración de métodos get*/

    /**
     * @return the direcciones

```

```

    */
    public List<DireccionDTO> getDirecciones ()
    {
        return _direcciones;
    }

    public DireccionDTO getDireccion () {
        return _direccion;
    }

    /*Declaración de métodos set*/

    public void setDireccion(DireccionDTO _direccion) {
        this._direccion = _direccion;
    }

    /**
     * @param direcciones the direcciones to set
     */
    public void setDirecciones( List<DireccionDTO> direcciones )
    {
        _direcciones = direcciones;
    }

    public void
setAdministradorDirecciones( AdministradorDirecciones
administradorDirecciones )
    {
        _administradorDirecciones = administradorDirecciones;
    }
}

```

DownloadAction.java

```

package com.uam.cofa.actions;

import java.io.File;
import java.io.FileInputStream;
import java.io.InputStream;
import com.opensymphony.xwork2.ActionSupport;

public class DownloadAction extends ActionSupport{

    private InputStream fileInputStream;
    private String _folioFactura;

    /**
     * @return String
     * Método que controla la descarga del archivo pdf
generado como comprobante
     * de la factura.
     */
    public String execute() throws Exception {

```

```

        //System.out.println("folio de la factura:
"+_folioFactura);
        //fileInputStream = new FileInputStream(new
File("C:/Users/Mario/Documents/workspace/COFA/WebContent/WEB-
INF/files/facturas/Factura"+_folioFactura+".pdf"));
        fileInputStream = new FileInputStream(new
File("Factura"+_folioFactura+".pdf"));
        return SUCCESS;
    }

    public void setFolioFactura(String folioFactura) {
        this._folioFactura = folioFactura;
    }

    public InputStream getFileInputStream() {
        return fileInputStream;
    }

    public String getFolioFactura() {
        return _folioFactura;
    }
}
}

```

FacturasAction.java

```

package com.uam.cofa.actions;

import java.util.List;

import javax.servlet.http.HttpSession;

import org.apache.struts2.ServletActionContext;

import com.opensymphony.xwork2.ActionSupport;
import com.uam.cofa.business.AdministradorCotizaciones;
import com.uam.cofa.business.AdministradorFacturas;
import com.uam.cofa.data.componentesUsuario.UsuarioDTO;
import com.uam.cofa.data.facturasCotizaciones.CotizacionDTO;
import com.uam.cofa.data.facturasCotizaciones.FacturaDTO;
import com.uam.cofa.data.facturasCotizaciones.ProductoEnCotizacionDTO;
import com.uam.cofa.utilities.GeneradorPDF;

/**
 * @author Eduardo González
 * Contiene métodos para la consulta de facturas por parte de los
 * usuarios
 * */

public class FacturasAction extends ActionSupport{

    private AdministradorFacturas _administradorFacturas;

```

```

private List<FacturaDTO> _facturas;
private String _facturaSerializada;
private FacturaDTO _facturaADetalle;
private CotizacionDTO _cotizacionFacturada;
private List<ProductoEnCotizacionDTO> _productosEnCotizacion;
private int _idFactura;

/**
 * @return String
 * Realiza una asignación del dato miembro _facturas (las
facturas del usuario)
 * de que la sesión este iniciada.
 * */
public String misFacturas () {
    HttpSession session =
ServletActionContext.getRequest ().getSession ();
    UsuarioDTO
usuario=(UsuarioDTO)session.getAttribute ("usuario");

    _facturas=_administradorFacturas.listarPorUsuario(usuario.getId());
    return "misFacturas";
}

/**
 * @return String
 * Realiza la búsqueda de la factura que corresponda con el
_idFactura
 * y se le asigana al dato miembro _facturaADetalle para que
sean mostrados
 * los datos de la facrura seleccionada.
 * */
public String mostrarInformacionDeFactura () {
    HttpSession session =
ServletActionContext.getRequest ().getSession ();
    UsuarioDTO
usuario=(UsuarioDTO)session.getAttribute ("usuario");
    FacturaDTO
factura=_administradorFacturas.buscaFactura (_idFactura);

    _cotizacionFacturada=_administradorFacturas.buscaCotizacion(factura.g
etIdCotizacion ().intValue ());
    _facturaADetalle=factura;

    _productosEnCotizacion=_administradorFacturas.listaProductosEnCotizac
ion(factura.getIdCotizacion ().intValue ());
    System.out.println("los pordcutos
son:"+_productosEnCotizacion.size ());
    return "mostrarInformacionDeFactura";
}

/*Metodos get*/

public List<FacturaDTO> getFacturas () {
    return _facturas;
}

```

```

public String get_facturaSerializada() {
    return _facturaSerializada;
}

public FacturaDTO getFacturaADetalle() {
    return _facturaADetalle;
}

public CotizacionDTO getCotizacionFacturada() {
    return _cotizacionFacturada;
}

public List<ProductoEnCotizacionDTO> getProductosEnCotizacion()
{
    return _productosEnCotizacion;
}

public int getIdFactura() {
    return _idFactura;
}
/*Metodos set*/

public void setFacturas(List<FacturaDTO> _facturas) {
    this._facturas = _facturas;
}

public void setAdministradorFacturas(
    AdministradorFacturas _administradorFacturas) {
    this._administradorFacturas = _administradorFacturas;
}

public void set_facturaSerializada(String _facturaSerializada)
{
    this._facturaSerializada = _facturaSerializada;
}

public void setFacturaADetalle(FacturaDTO _facturaADetalle) {
    this._facturaADetalle = _facturaADetalle;
}

public void setCotizacionFacturada(CotizacionDTO
_cotizacionFacturada) {
    this._cotizacionFacturada = _cotizacionFacturada;
}

public void setProductosEnCotizacion(
    List<ProductoEnCotizacionDTO>
_prodcutosEnCotizacion) {
    this._productosEnCotizacion = _prodcutosEnCotizacion;
}

public void setIdFactura(int _idFactura) {
    this._idFactura = _idFactura;
}
}

```

InicioAction.java

```
package com.uam.cofa.actions;

import java.util.Map;
import java.util.TreeMap;
import javax.servlet.http.HttpSession;
import org.apache.struts2.ServletActionContext;
import com.opensymphony.xwork2.ActionSupport;

/**
 * @author Eduardo González
 */

public class InicioAction extends ActionSupport {

    /**@return String
     * Método para iniciar la aplicación web
     */
    public String index(){
        HttpSession session =
ServletActionContext.getRequest().getSession();
        if(session.getAttribute("UsuarioLogado")==null )
            return "no_registrado";
        return "registrado";
    }

    public String indexRegistrado(){
        return "indexRegistrado";
    }

    /**@return String
     * Invalida la sesión del usuario y regresa a la página
principal.
     */
    public String salir(){
        HttpSession session =
ServletActionContext.getRequest().getSession();
        session.invalidate();
        return "salir";
    }

}

```

ModosDeContactoAction.java

```
package com.uam.cofa.actions;

import com.opensymphony.xwork2.ActionSupport;
```

```

import com.opensymphony.xwork2.validator.annotations.*;
import com.uam.cofa.data.componentesUsuario.MedioDeContactoDTO;
import java.util.List;
import java.util.LinkedList;

import javax.servlet.http.HttpSession;

import org.apache.struts2.ServletActionContext;
import com.opensymphony.xwork2.validator.annotations.*;

@Validation

public class ModosDeContactoAction extends ActionSupport{

    private String _correo;
    private String _confirmacionCorreo;
    private String _telefonoCasa;
    private String _telefonoCelular;
    private String _telefonoTrabajo;
    private String _fax;
    private LinkedList<MedioDeContactoDTO>
    _listaModosDeContacto=new LinkedList ();

    /**
     * @return String
     * Método para el control en Struts. Sirve para abrir la página
donde se dan
     * de alta los modos de contacto.
     */
    public String forma_insertarModosDeContacto () {
        return "forma_insertarModosDeContacto";
    }

    /**
     * @return String
     * Analiza los datos miembro
    _correo, _telefonoCasa, _telefonoCelular,
     * _telefonoTrabajo y _fax y si datos válidos para el tipo de
contacto que
     * representan son almacenados en la base de datos.
     */
    public String insertarModosDeContacto () {
        if (_correo.length () > 1)
            guardaEnListaModoDeContacto (2, _correo);
        if (_telefonoCasa.length () > 1)
            guardaEnListaModoDeContacto (3, _telefonoCasa);
        if (_telefonoCelular.length () > 1)
            guardaEnListaModoDeContacto (4, _telefonoCelular);
        if (_telefonoTrabajo.length () > 1)
            guardaEnListaModoDeContacto (5, _telefonoTrabajo);
        if (_fax.length () > 1)
            guardaEnListaModoDeContacto (1, _fax);
        HttpSession session =
ServletActionContext.getRequest ().getSession ();

        session.setAttribute ("listaModosDeContacto", _listaModosDeContacto);
        return "modosDeContactoInsertados";
    }
}

```

```

public String listar(){
    return "modosDeContactoInsertados";
}

/**Guarda en la lista ListaModosDeContacto objetos de tipo
ModoDeContactoDTO
*Recibe como parametro el id del tipo de contacto y la cadena
correspondiente
*a esa forma de contacto
* */
public void guardaEnListaModoDeContacto(int
modoDeContacto,String contacto){
    MedioDeContactoDTO medioDeContacto=new
MedioDeContactoDTO(modoDeContacto,contacto);
    _listaModosDeContacto.add(medioDeContacto);
}

/*Declaración de los métodos get*/

@RequiredStringValidator(message = "El correo es requerido")
@emailValidator(message = "Dirección de correo inválida")
public String getCorreo() {
    return _correo;
}

public String getConfirmacionCorreo() {
    return _confirmacionCorreo;
}

@RequiredStringValidator(message = "El teléfono es requerido")
public String getTelefonoCasa() {
    return _telefonoCasa;
}

public String getTelefonoCelular() {
    return _telefonoCelular;
}

public String getTelefonoTrabajo() {
    return _telefonoTrabajo;
}

public String getFax() {
    return _fax;
}

/*Declaración de los métodos set*/

public void setCorreo(String _correo) {
    this._correo = _correo;
}

public void setConfirmacionCorreo(String _confirmacionCorreo) {
    this._confirmacionCorreo = _confirmacionCorreo;
}

```



```

public void setTelefonoCasa(String _telefonoCasa) {
    this._telefonoCasa = _telefonoCasa;
}

public void setTelefonoCelular(String _telefonoCelular) {
    this._telefonoCelular = _telefonoCelular;
}

public void setTelefonoTrabajo(String _telefonoTrabajo) {
    this._telefonoTrabajo = _telefonoTrabajo;
}

public void setFax(String _fax) {
    this._fax = _fax;
}
}

```

ProductosAction.java

```

package com.uam.cofa.actions;

import com.uam.cofa.business.AdministradorProductos;
import com.opensymphony.xwork2.ActionContext;
import com.opensymphony.xwork2.ActionSupport;
import com.uam.cofa.data.productosServicios.*;
import com.uam.cofa.utilities.UtileriasAdministrarSistema;
import net.handle.hdllib.Util;
import java.io.File;
import java.util.*;
import javax.servlet.http.HttpSession;
import org.apache.struts2.ServletActionContext;

public class ProductosAction extends ActionSupport{

    public UtileriasAdministrarSistema _utileriasSistema=new
    UtileriasAdministrarSistema();
    List<ProductoDTO> _productos = new ArrayList<ProductoDTO>();
    private AdministradorProductos _administradorProductos;
    private ProductoDTO _producto;
    private Map<Integer,String> _marcas;
    private Map<Integer,String> _subtiposDeProductos;
    private List<SubtipoProductoDTO> _listaSubtiposProductos;
    private List<TipoProductoDTO> _listaTiposProductos;
    private String _id;
    private File _upload;
    private String _productoAModificarSerializado ;
    private ProductoDTO _productoAModificar;
    private int _idProducto;

    /**
     * @return String
     * Realiza la asignación del dato miembro _productos con el
    resultado encontrado
     * en la base de datos(tabla productos), ademas llama a los
    métodos

```

```

    * setListaTiposProductos() y setListaSubtiposProductos() para
cargar los
    * catálogos de tipos y subtipos de productos .
    * */
    public String listar(){
        setListaTiposProductos();
        setListaSubtiposProductos();
        _productos = _administradorProductos.listarProductos();
        return "listar";
    }

    /**
    * Es igual que el método listar pero en este el resultado va
para el jsp
    * listarParaModificar.
    * */
    public String listarParaModificar(){
        setListaTiposProductos();
        setListaSubtiposProductos();
        _productos = _administradorProductos.listarProductos();
        return "listarParaModificar";
    }

    /**
    * @return String
    * Realiza una asignación del dato miembro _productos con una
consulta realizada
    * mostrando solo los productos pertenecientes al subtipo _id
    * */
    public String listarPorTipo(){
        setListaTiposProductos();
        setListaSubtiposProductos();
        _productos =
_administradorProductos.listarProductosPorTipo(_id);
        return "listar";
    }

    /**
    * @return String
    * Es igual que el método listarPorTipo pero en este el
resultado va para el jsp
    * listarParaModificar.
    * */
    public String listarPorTipo_modificar(){
        setListaTiposProductos();
        setListaSubtiposProductos();
        _productos =
_administradorProductos.listarProductosPorTipo(_id);
        return "listarParaModificar";
    }

    /**
    *@return String
    *Ayuda a mostrar el jsp forma_modificarProducto, además
realiza la asignación
    *del dato miembro _productoAModificar con el producto
correspondiente al
    *_idProducto
    */
    public String forma_modificarProducto(){
        setMarcas();

```

```

        setListaSubtiposProductos();
        setSubtiposDeProductos();

_productoAModificar=_administradorProductos.buscarProducto(_idProducto);
    }
    return "forma_modificarProducto";
}

/**
 * @return String
 * Actualiza el producto _producto
 * */
public String modificar(){
    _producto.setImagen(Util.getBytesFromFile(_upload));
    _administradorProductos.actualizar(_producto);
    return "productoModificado";
}

/**
 * @return String
 * Borra el producto _producto
 * */
public String borrar(){

_productoAModificar=_administradorProductos.buscarProducto(_idProducto);
    _administradorProductos.borrar(_productoAModificar);
    return "productoModificado";
}

/**
 * @return String
 * Ayuda a mostrar la froma_insertarProducto
 * */
public String forma_insertarProducto(){
    setMarcas();
    setListaSubtiposProductos();
    setSubtiposDeProductos();
    return "forma_insertarProducto";
}

/**
 * @return String
 * Inserta en la base de datos el producto _producto
 * */
public String insertar(){
    _producto.setImagen(Util.getBytesFromFile(_upload));
    System.out.println(_producto.toString());
    _administradorProductos.insertar(_producto);
    return "insertado";
}

public String administrar(){
    return "administrar";
}

/**
 * @return the productos
 * */

```

```

public List<ProductoDTO> getProductos ()
{
    return _productos;
}

public ProductoDTO getProducto () {
    return _producto;
}

public Map<Integer, String> getMarcas () {
    return _marcas;
}

public Map<Integer, String> getSubtiposDeProductos () {
    return _subtiposDeProductos;
}

public List<SubtipoProductoDTO> getListaSubtiposProductos () {
    return _listaSubtiposProductos;
}

public List<TipoProductoDTO> getListaTiposProductos () {
    return _listaTiposProductos;
}

public String getProductoAModificarSerializado () {
    return _productoAModificarSerializado;
}

public ProductoDTO getProductoAModificar () {
    return _productoAModificar;
}

public File getUpload () {
    return _upload;
}

public String getId () {
    return _id;
}

public int getIdProducto () {
    return _idProducto;
}

/**
 * @param productos the productos to set
 */
public void setProdcutos ( List<ProductoDTO> productos )
{
    _productos = productos;
}

public void setProducto(ProductoDTO _producto) {
    this._producto = _producto;
}

public void setAdministradorProductos ( AdministradorProductos
administradorProductos )

```

```

    {
        _administradorProductos = administradorProductos;
    }

    public void setListaSubtiposProductos () {
        this._listaSubtiposProductos =
        _administradorProductos.consultaSubtiposDeProducto ();
    }

    public void setListaTiposProductos () {
        this._listaTiposProductos =
        _administradorProductos.consultaTiposDeProducto ();
    }

    public void setMarcas () {
        List<MarcaDTO>
        listaMarcas=_administradorProductos.consultaMarcas ();

        _marcas=_utileriasSistema.cargarCatalogoDeMarcas (listaMarcas);
    }

    public void setSubtiposDeProductos () {

        _subtiposDeProductos=_utileriasSistema.cargarCatalogoDeSubtipos (_list
        aSubtiposProductos);
    }

    public void setProductoAModificar (String _productoAModificar)
    {
        this._productoAModificarSerializado =
        _productoAModificar;
    }

    public void setProductoAModificarSerializado (
        String _productoAModificarSerializado) {
        this._productoAModificarSerializado =
        _productoAModificarSerializado;
    }

    public void setProductoAModificar (ProductoDTO
    _productoAModificar) {
        this._productoAModificar = _productoAModificar;
    }

    public void setUpload (File _upload) {
        this._upload = _upload;
    }

    public void setId (String _id) {
        this._id = _id;
    }

    public void setIdProducto (int _idProducto) {
        this._idProducto = _idProducto;
    }
}

```

UsuariosAction.java

```
package com.uam.cofa.actions;

import com.uam.cofa.business.AdministradorUsuarios;
import com.uam.cofa.data.componentesUsuario.*;
import com.uam.cofa.utilities.*;
import com.opensymphony.xwork2.ActionSupport;
import java.util.*;
import javax.servlet.http.HttpSession;
import org.apache.struts2.ServletActionContext;

/**
 * @author Eduardo Gonzalez
 */

public class UsuariosAction extends ActionSupport{

    List<UsuarioDTO> _usuarios = new ArrayList<UsuarioDTO>();
    private AdministradorUsuarios _administradorUsuarios;
    private UsuarioDTO _usuario=new UsuarioDTO();
    private DatosDeAccesoDTO _datosDeAcceso=new
DatosDeAccesoDTO();
    private String _confirmacionPassword;
    private Map<Integer,String> _ocupaciones;

    public String listar()
    {
        _usuarios = _administradorUsuarios.listarUsuarios();
        System.out.println(_usuarios.size());
        return "listar";
    }

    /**
     * @return String
     * Ayuda a mostrar el jsp forma_insertarCliente
     */
    public String forma_insertarCliente(){
        setOcupaciones();
        return "forma_insertarCliente";
    }

    /**Guarda los datos en sesión para su posterior inserción en
la base
     * en el método insetar datosCompletos()*/
    public String insertarCliente(){
        _usuario.setIdTipoDeUsuario(2);
        HttpSession session =
ServletActionContext.getRequest().getSession();
        session.setAttribute("usuarioAInsertar",_usuario);
        session.setAttribute("datosDeAccesoNuevos",_datosDeAcceso);
        return "clienteInsertado";
    }
}
```

```

    /**
     * @return String
     * Obtiene todos los datos que se guardaron en sesion ya que
 se deben realizar
     * todas las inserciones en esta parte para que se realice un
 registro completo*/
    public String insertarDatosCompletos () {
        EncriptadorPassword encriptador=new
EncriptadorPassword ();
        HttpSession session =
ServletActionContext.getRequest ().getSession ();
        UsuarioDTO
usuario=(UsuarioDTO)session.getAttribute ("usuarioAInsertar");
        DatosDeAccesoDTO
datosDeAcceso=(DatosDeAccesoDTO)session.getAttribute ("datosDeAccesoNu
evos");

        datosDeAcceso.setPassword(encriptador.encriptaMD5 (datosDeAcceso.getPa
ssword ()));

        DireccionDTO
direccion=(DireccionDTO)session.getAttribute ("direccionNueva");
        LinkedList<MedioDeContactoDTO>
medios=(LinkedList)session.getAttribute ("listaModosDeContacto");
        MedioDeContactoDTO aux=new MedioDeContactoDTO ();

        usuario=_administradorUsuarios.insertarUsuario (usuario);
        datosDeAcceso.setIdUsuario (usuario.getId ());
        direccion.setIdUsuario (usuario.getId ());
        _administradorUsuarios.insertaDireccion (direccion);

        _administradorUsuarios.insertaDatosDeAcceso (datosDeAcceso);
        for(int i=0;i<medios.size ();i++){
            aux=medios.get (i);
            aux.setIdUsuario (usuario.getId ());

        _administradorUsuarios.insertaMedioDeContacto (aux);
            aux=new MedioDeContactoDTO ();
        }
        return "registroNuevoUsuarioExitoso";
    }

    /**
     * @return the usuarios
     */
    public List<UsuarioDTO> getUsuarios ()
    {
        return _usuarios;
    }

    /**
     * @param usuarios the usuarios to set
     */
    public void setUsuarios ( List<UsuarioDTO> usuarios )
    {
        _usuarios = usuarios;
    }

```

```

        public void setAdministradorUsuarios( AdministradorUsuarios
administradorUsuarios )
        {
            _administradorUsuarios = administradorUsuarios;
        }

        public UsuarioDTO getUsuario() {
            return _usuario;
        }

        public void setUsuario(UsuarioDTO usuario) {
            _usuario=usuario;
        }

        public DatosDeAccesoDTO getDatosDeAcceso() {
            return _datosDeAcceso;
        }

        public void setDatosDeAcceso(DatosDeAccesoDTO datosDeAcceso) {
            _datosDeAcceso=datosDeAcceso;
        }

        public String getConfirmacionPassword() {
            return _confirmacionPassword;
        }

        public void setConfirmacionPassword(String
confirmacionPassword) {
            _confirmacionPassword=confirmacionPassword;
        }

        public Map<Integer,String> getOcupaciones() {
            return _ocupaciones;
        }

        public void setOcupaciones() {
            UtileriasAdministrarSistema aux=new
UtileriasAdministrarSistema();
            List <OcupacionDTO>
listaOcupaciones=_administradorUsuarios.consultaOcupaciones();

            this._ocupaciones=aux.cargarCatalogoDeOcupaciones(listaOcupaciones);
        }
    }
}

```

ValidarUsuariosAction.java

```

package com.uam.cofa.actions;

import com.uam.cofa.business.*;
import com.uam.cofa.data.componentesUsuario.*;
import com.opensymphony.xwork2.ActionContext;
import com.opensymphony.xwork2.ActionSupport;
import java.util.ArrayList;

```



```

import java.util.*;
import com.uam.cofa.data.productosServicios.*;

import javax.servlet.http.HttpSession;

import org.apache.struts2.ServletActionContext;

/**
 * @author Eduardo Gonzalez
 */
public class ValidarUsuarioAction extends ActionSupport
{
    List<DatosDeAccesoDTO> _listaDatosDeAcceso = new
ArrayList<DatosDeAccesoDTO>();
    private DatosDeAccesoDTO _datosDeAcceso=new DatosDeAccesoDTO();
    private ValidadorUsuarios _validadorUsuarios;
    private AdministradorUsuarios _administradorUsuarios;
    private String _mensaje;

    public String listar()
    {
        _listaDatosDeAcceso = _validadorUsuarios.listarDatosDeUsuario();
        System.out.println(_listaDatosDeAcceso.size());
        return "listar";
    }

    /**
     * @return String
     */
    public String forma_validarUsuario(){
        return "forma_validarUsuario";
    }

    /**
     * @return String
     * Valida al usuario y guarda variables de sesión necesarias para
el
     * funcionamiento de la aplicación.
     */
    public String validarUsuario(){
        System.out.println("Estoy validando al usuario:");
        CarritoDeCompras carritoDeCompras=new CarritoDeCompras();
        Integer
idUsuario=_validadorUsuarios.validarUsuario(_datosDeAcceso);
        if(!idUsuario.equals(new Integer(-1))){
            HttpSession session =
ServletActionContext.getRequest().getSession();

session.setAttribute("UsuarioLogado",_datosDeAcceso.getLogin());
            session.setAttribute("tipoDeUsuario",new Integer(1));

session.setAttribute("carritoDeCompras",carritoDeCompras);
            setMensaje("Bienvenido(a) :
"+_datosDeAcceso.getLogin());
            System.out.println("Datos validos");

```

```

        UsuarioDTO
usuario=_validadorUsuarios.buscarUsuario(idUsuario);
        session.setAttribute("usuario",usuario);
        return "validado";
    }
    else{
        setMensaje("Datos inválidos. Favor de verificar ");
        System.out.println("Datos no validos");
        return "error";
    }
}

/**
 * @return the datosDeAcceso
 */
public List<DatosDeAccesoDTO> getListaDatosDeAcceso ()
{
    return _listaDatosDeAcceso;
}

/**
 * @param listaDatosDeAcceso the datosDeAcceso to set
 */
public void setListaDatosDeAcceso ( List<DatosDeAccesoDTO>
listaDatosDeAcceso )
{
    _listaDatosDeAcceso = listaDatosDeAcceso;
}

public void setValidadorUsuarios ( ValidadorUsuarios
validadorUsuarios )
{
    _validadorUsuarios = validadorUsuarios;
}

public DatosDeAccesoDTO getDatosDeAcceso () {
    return _datosDeAcceso;
}

public void setDatosDeAcceso (DatosDeAccesoDTO datosDeAcceso) {
    _datosDeAcceso=datosDeAcceso;
}

public String getMensaje () {
    return _mensaje;
}

public void setMensaje (String mensaje) {
    _mensaje=mensaje;
}

public void set_administradorUsuarios (
    AdministradorUsuarios _administradorUsuarios) {
    this._administradorUsuarios = _administradorUsuarios;
}
}
}

```

com.uam.cofa.business

AdministradorCotizaciones.java

```
package com.uam.cofa.business;

import org.springframework.transaction.annotation.Propagation;
import org.springframework.transaction.annotation.Transactional;
import java.util.List;
import com.uam.cofa.data.facturasCotizaciones.*;
import com.uam.cofa.data.productosServicios.CarritoDeCompras;
import com.uam.cofa.data.productosServicios.ProductoDTO;

/**
 * @author Eduardo Gonzalez
 * Contiene los métodos para la administración de las cotizaciones
 */

@Transactional( rollbackFor=Exception.class )
public interface AdministradorCotizaciones {

    @Transactional( propagation=Propagation.SUPPORTS, readOnly =
true )
    List<CotizacionDTO> listarCotizaciones(int tipo);
    List<CotizacionDTO> listarCotizacionesPorUsuario(int
idUsuario,int tipo);
    CotizacionDTO insertar(CotizacionDTO cotizacion);
    boolean guardarProductosDeCotizacion(CarritoDeCompras
carrito,Integer idCotizacion);
    void guardarProductoDeCotizacion(ProductoEnCotizacionDTO
producto,Integer idCotizacion);
    void cerrarCotizacion(CotizacionDTO cotizacion);
    void cancelarCotizacion(CotizacionDTO cotizacion);
    CotizacionDTO buscarCotizacion(int idCotizacion);
    ProductoDTO buscaProducto(int idProducto);

}

```

AdministradorDirecciones.java

```
Package com.uam.cofa.business;

import com.uam.cofa.data.componentesUsuario.DireccionDTO;
import org.springframework.transaction.annotation.Propagation;
import org.springframework.transaction.annotation.Transactional;
import java.util.List;

/**
 * @author Eduardo Gonzalez
 * Contiene los métodos para la administración de las direcciones
 */

@Transactional( rollbackFor=Exception.class )

```

```

public interface AdministradorDirecciones {

    @Transactional( propagation=Propagation.SUPPORTS, readOnly =
true )
    List<DireccionDTO> listarDirecciones ();
    boolean insertaDireccion(DireccionDTO direccion);
}

```

```

package com.uam.cofa.business;

```

```

import org.springframework.transaction.annotation.Propagation;
import org.springframework.transaction.annotation.Transactional;
import java.util.List;

```

```

import com.uam.cofa.data.componentesUsuario.UsuarioDTO;
import com.uam.cofa.data.factorasCotizaciones.*;

```

```

/**
 * @author Eduardo Gonzalez
 * Contiene los métodos para la administración de las facturas
 */

```

```

@Transactional( rollbackFor=Exception.class )
public interface AdministradorFacturas {

    @Transactional( propagation=Propagation.SUPPORTS, readOnly =
true )
    List<FacturaDTO> listar ();
    List<FacturaDTO> listarPorUsuario(int idUsuario);
    List<CotizacionDTO> listarCotizacionesCerradas ();
    FacturaDTO obtenUltimaFactura ();
    FacturaDTO guardar(FacturaDTO factura);
    CotizacionDTO buscaCotizacion(int idCotizacion);
    List<ProductoEnCotizacionDTO> listaProductosEnCotizacion(int
idCotizacion);
    UsuarioDTO buscaUsuario(int id);
    FacturaDTO buscaFactura(int idFactura);
}

```

```

package com.uam.cofa.business;

```

```

import com.uam.cofa.data.productosServicios.*;
import org.springframework.transaction.annotation.Propagation;
import org.springframework.transaction.annotation.Transactional;
import java.util.*;

```

```

/**
 * @author Eduardo Gonzalez
 * Contiene los métodos para la administración de los productos
 */

```

```

@Transactional( rollbackFor=Exception.class )

```

```

public interface AdministradorProductos {

    @Transactional( propagation=Propagation.SUPPORTS, readOnly =
true )
    List<ProductoDTO> listarProductos ();
    List<MarcaDTO> consultaMarcas ();
    List<SubtipoProductoDTO> consultaSubtiposDeProducto ();
    List<TipoProductoDTO> consultaTiposDeProducto ();
    List<ProductoDTO> listarProductosPorTipo (String tipo);
    boolean insertar (ProductoDTO producto);
    boolean actualizar (ProductoDTO producto);
    boolean borrar (ProductoDTO producto);
    ProductoDTO buscarProducto (int idProducto);

}

```

AdministradorUsuarios.java

```

package com.uam.cofa.business;

import com.uam.cofa.data.componentesUsuario.*;

import org.springframework.transaction.annotation.Propagation;
import org.springframework.transaction.annotation.Transactional;
import java.util.*;

/**
 * @author Eduardo Gonzalez
 * Contiene los métodos para la administración de los usuarios
 */

@Transactional( rollbackFor=Exception.class )
public interface AdministradorUsuarios {

    @Transactional( propagation=Propagation.SUPPORTS, readOnly =
true )
    List<UsuarioDTO> listarUsuarios ();
    UsuarioDTO insertarUsuario (UsuarioDTO usuario);
    boolean insertaDatosDeAcceso (DatosDeAccesoDTO datosDeAcceso);
    List<OcupacionDTO> consultaOcupaciones ();
    boolean insertaMedioDeContacto (MedioDeContactoDTO
medioDeContacto);
    boolean insertaDireccion (DireccionDTO direccion);

}

```

ValidadorUsuarios.java

```

package com.uam.cofa.business;

import com.uam.cofa.data.componentesUsuario.DatosDeAccesoDTO;
import com.uam.cofa.data.componentesUsuario.UsuarioDTO;

import org.springframework.transaction.annotation.Propagation;
import org.springframework.transaction.annotation.Transactional;
import java.util.List;

```

```

/**
 * @author Eduardo Gonzalez
 * Contiene los métodos para la validación de los usuarios
 */

@Transactional( rollbackFor=Exception.class )
public interface ValidadorUsuarios
{
    @Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
    List<DatosDeAccesoDTO> listarDatosDeUsuario ();
    Integer validarUsuario (DatosDeAccesoDTO datosDeAcceso);
    UsuarioDTO buscarUsuario (Integer idUsuario);
}

```

com.uam.cofa.business.local

AdministradorFacturasImpl.java

```

package com.uam.cofa.business.local;

import java.util.List;

import org.hibernate.Query;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import com.uam.cofa.business.*;
import com.uam.cofa.data.componentesUsuario.DatosDeAccesoDTO;
import com.uam.cofa.data.componentesUsuario.UsuarioDTO;
import com.uam.cofa.data.facturasCotizaciones.CotizacionDTO;
import com.uam.cofa.data.facturasCotizaciones.FacturaDTO;
import com.uam.cofa.data.facturasCotizaciones.ProductoEnCotizacionDTO;

/**
 * @author Eduardo Gonzalez
 * Implementa los métodos declarados en la interfaz
 * com.uam.cofa.business.AdministracionFacturas
 */

public class AdministradorFacturasImpl implements
AdministradorFacturas{

    private SessionFactory _sessionFactory;
    private static final int _CERRADA=2;

    /**
     * @return List<FacturaDTO>
     * Realiza una consulta en la tabla facturas y regresa una
lista con
     * las facturas encontradas
     */
    public List<FacturaDTO> listar () {

```

```

        System.out.println("listando facturas");
        Session session=getSession();
        Query query= session.createQuery("SELECT f FROM
FacturaDTO f");
        return query.list();
    }

    /**
     * @return List<FacturaDTO>
     * @param idUsuario
     * Realiza una consulta en la tabla facturas y regresa una
lista con
     * las facturas encontradas del usuario idUsuario
     */
    public List<FacturaDTO> listarPorUsuario(int idUsuario){
        System.out.println("listando facturas por usuario ");
        Session session=getSession();
        Query query= session.createQuery("SELECT f FROM
FacturaDTO f WHERE f.idUsuario="+idUsuario);
        return query.list();
    }

    /**
     * @return List<CotizacionDTO>
     * Realiza una consulta en la tabla cotizaciones y regresa una
lista con
     * las cotizaciones encontradas del tipo cerradas.
     */
    public List<CotizacionDTO> listarCotizacionesCerradas(){
        System.out.println("consultando cotizaciones cerradas");
        Session session=getSession();
        Query query=session.createQuery("SELECT c FROM
CotizacionDTO c WHERE c.tipo="+_CERRADA);
        return query.list();
    }

    /**
     * @return FacturaDTO
     * Regresa la ultima factura registrada en la base de datos
     */
    public FacturaDTO obtenUltimaFactura(){
        Session session=getSession();
        Query query=session.createQuery("SELECT max(f.id) from
FacturaDTO f");
        Integer max=(Integer)query.list().get(0);
        query=session.createQuery("SELECT f FROM FacturaDTO f
WHERE f.id="+max);
        query.setMaxResults(1);
        FacturaDTO factura = (FacturaDTO) query.uniqueResult();
        return factura;
    }

    /**
     * @return FacturaDTO
     * @param idFactura
     * Regresa la factura correspondiente al parametro idFactura
     */
    public FacturaDTO buscaFactura(int idFactura){
        Session session=getSession();
        System.out.println("buscando la factura con id:
"+idFactura);
    }

```

```

        Query query=session.createQuery("SELECT f FROM FacturaDTO
f WHERE f.id="+idFactura);
        return (FacturaDTO)query.list().get(0);
    }

    /**
     * @return FacturaDTO
     * @param factura
     * Realiza una inserción en la base de datos de una nueva
factura correspondiente
     * al parámetro factura
     * */
    public FacturaDTO guardar(FacturaDTO factura){
        Session session=getSession();
        session.beginTransaction();
        session.save(factura);
        session.getTransaction().commit();
        return factura;
    }

    /**
     * @return CotizacionDTO
     * @param idCotizacion
     * Realiza una consulta en la tabla cotizaciones y regresa un
objeto de
     * tipo CotizacionDTO correspondiente al parámetro
idCotizacion
     */
    public CotizacionDTO buscaCotizacion(int idCotizacion){
        Session session=getSession();
        Query query=session.createQuery("SELECT c FROM
CotizacionDTO c WHERE c.id="+idCotizacion);
        CotizacionDTO
cotizacion=(CotizacionDTO)query.list().get(0);
        return cotizacion;
    }

    /**
     * @return List<ProductoEnCotizacionDTO>
     * @param idCotizacion
     * Realiza una consulta en la tabla productosencotizacion y
regresa una lista
     * con todos los productos relacionados con la cotización
correspondiente al idCotizacion
     * */
    public List<ProductoEnCotizacionDTO>
listaProductosEnCotizacion(int idCotizacion){
        Session session=getSession();
        Query query=session.createQuery("SELECT p FROM
ProductoEnCotizacionDTO p WHERE p.idCotizacion="+idCotizacion);
        return query.list();
    }

    /**
     * @return UsuarioDTO
     * @param idUsuario
     * Realiza una consulta en la tabla usuarios y regresa un
objeto de
     * tipo UsuarioDTO correspondiente al parámetro idUsuario
     */
    public UsuarioDTO buscaUsuario(int idUsuario){

```



```

        Session session=getSession();
        Query query=session.createQuery("SELECT u FROM UsuarioDTO
u WHERE u.id="+idUsuario);
        return (UsuarioDTO)query.list().get(0);
    }

    private Session getSession(){
        return _sessionFactory.getCurrentSession();
    }

    public void setSessionFactory(SessionFactory _sessionFactory) {
        this._sessionFactory = _sessionFactory;
    }
}

```

AdministradorProductosImpl.java

```

package com.uam.cofa.business.local;

import com.uam.cofa.business.AdministradorProductos;
import com.uam.cofa.data.productosServicios.*;
import java.util.*;
import org.hibernate.*;

/**
 * @author Eduardo Gonzalez
 * Implementa los métodos declarados en la interfaz
 * com.uam.cofa.business.AdministracionProductos
 */

public class AdministradorProductosImpl implements
AdministradorProductos{

    private SessionFactory _sessionFactory;

    /**
     * @return List<ProductoDTO>
     * Realiza una consulta a la base de datos y regresa una
lista con objetos
     * ProdcutoDTO correspondientes */
    public List<ProductoDTO> listarProductos()
    {
        Session session = getSession();
        Query query = session.createQuery( "SELECT p FROM
ProductoDTO p WHERE p.estado=1" );
        return query.list();
    }

    /**
     * @return List<ProductoDTO>
     * @param tipo
     * Realiza una consulta a la tabla productos y regresa una
lista con objetos
     * de tipo ProdcutoDTO en base al tipo o subtipo recibido
como parametro */

```

```

public List<ProductoDTO> listarProductosPorTipo(String tipo){
    StringTokenizer st=new StringTokenizer(tipo);
    st.nextToken();
    String idTipo=st.nextToken("");
    String consulta="SELECT p FROM ProductoDTO p WHERE
p.idSubtipo="+idTipo+" AND p.estado=1";
    Session session=getSession();
    Query query = session.createQuery(consulta);
    return query.list();
}

/**
 * @return boolean
 * @param producto
 * Agrega un producto nuevo a la base*/
public boolean insertar(ProductoDTO producto){
    Session session=getSession();
    session.beginTransaction();
    session.save(producto);
    session.getTransaction().commit();
    return true;
}

public boolean actualizar(ProductoDTO producto){
    Session session=getSession();
    session.beginTransaction();
    session.update(producto);
    session.getTransaction().commit();
    return true;
}

public boolean borrar(ProductoDTO producto){
    Session session=getSession();
    session.beginTransaction();
    producto.setEstado(new Integer(0));
    System.out.println("aca esta el
error:"+producto.toString());
    session.update(producto);
    session.getTransaction().commit();
    return true;
}

/**Realiza una consulta en la base de datos,obtiene las
marcas de los
    productos y los regresa en una lista. */
public List<MarcaDTO> consultaMarcas(){
    Session session=getSession();
    Query query = session.createQuery("SELECT m FROM
MarcaDTO m");
    return query.list();
}

/**Realiza una consulta en la base de datos,obtiene los
subtipos
    * de los productos y los regresa en una lista. */
public List<SubtipoProductoDTO> consultaSubtiposDeProducto(){
    Session session=getSession();
    Query query = session.createQuery("SELECT s FROM
SubtipoProductoDTO s");
    return query.list();
}

```

```

        /**Realiza una consulta en la base de datos,obtiene los tipos
        * de los productos y los regresa en una lista. */
        public List<TipoProductoDTO> consultaTiposDeProducto(){
            Session session=getSession();
            Query query = session.createQuery("SELECT t FROM
TipoProductoDTO t");
            return query.list();
        }

        public ProductoDTO buscarProducto(int idProducto){
            Session session=getSession();
            Query query=session.createQuery("SELECT p FROM
ProductoDTO p WHERE p.id="+idProducto);
            return (ProductoDTO)query.list().get(0);
        }

        public void setSessionFactory( SessionFactory sessionFactory
)
        {
            _sessionFactory = sessionFactory;
        }

        private Session getSession()
        {
            return _sessionFactory.getCurrentSession();
        }
    }

```

AdministradorUsuariosImpl.java

```

package com.uam.cofa.business.local;

import com.uam.cofa.business.AdministradorUsuarios;
import com.uam.cofa.data.componentesUsuario.*;

import java.util.*;
import org.hibernate.*;

/**
 * @author Eduardo Gonzalez
 * Implementa los métodos declarados en la interfaz
 * com.uam.cofa.business.AdministracionUsuarios
 */

public class AdministradorUsuariosImpl implements
AdministradorUsuarios{

    private SessionFactory _sessionFactory;

    /*Lista los usuarios registrados existentes*/

    public List<UsuarioDTO> listarUsuarios()
    {
        System.out.println("listando usuarios");
        Session session = getSession();
    }

```

```

        Query query = session.createQuery( "SELECT t FROM
UsuarioDTO t" );
        return query.list();
    }

    /**
     * @return UsuarioDTO
     * inserta un nuevo usuario en la base de datos (tabla de
    usuarios )*/
    public UsuarioDTO insertarUsuario(UsuarioDTO usuario){
        System.out.println("agregando usuario");
        Session session = getSession();
        session.beginTransaction();
        session.save(usuario);
        session.getTransaction().commit();
        return usuario;
    }

    /**
     * @return boolean
     * @param datosDeAcceso
     * Inserta los datos de acceso del usuario (login y
    password)*/
    public boolean insertaDatosDeAcceso(DatosDeAccesoDTO
datosDeAcceso){
        System.out.println("agregando datos de acceso");
        Session session = getSession();
        session.beginTransaction();
        session.save(datosDeAcceso);
        session.getTransaction().commit();
        return true;
    }

    /**
     * @return boolean
     * @param medioDeContacto
     * Inserta los datos correspondientes a la forma en que se
    contactara al usuario*/
    public boolean insertaMedioDeContacto(MedioDeContactoDTO
medioDeContacto){
        System.out.println("agregando medios de contacto");
        Session session = getSession();
        session.beginTransaction();
        session.save(medioDeContacto);
        session.getTransaction().commit();
        return true;
    }

    public boolean insertaDireccion(DireccionDTO direccion){
        System.out.println("Insertando direccion");
        Session session=getSession();
        session.beginTransaction();
        session.save(direccion);
        session.getTransaction().commit();
        return true;
    }

    /**

```

```

        * @return List<OcupacionDTO>
        * obtiene una lista de las ocupaciones registradas en la
tabla de ocupaciones
        en la BD*/
        public List<OcupacionDTO> consultaOcupaciones () {
            Session session = getSession();
            Query query = session.createQuery( "SELECT o from
OcupacionDTO o " );
            return query.list();
        }

        public void setSessionFactory( SessionFactory sessionFactory
)
        {
            _sessionFactory = sessionFactory;
        }

        private Session getSession()
        {
            return _sessionFactory.getCurrentSession();
        }
    }
}

```

Cotizador.java

```

package com.uam.cofa.business.local;

import com.uam.cofa.data.productosServicios.*;
import com.uam.cofa.data.facturasCotizaciones.*;
import java.util.*;
import javax.servlet.http.HttpSession;
import org.apache.struts2.ServletActionContext;

/**
 * @author Eduardo Gonzalez
 * Cotizador es un clase que contiene metodos necesarios para darle
 * funcionalidad al módulo de generar cotizaciones.
 */

public class Cotizador {

    private static final float _IVA=(float)0.16;

    /**Recibe como parámetro un objeto de tipo ProductoDTO y
    calcula su IVA
    *en base a su precio y el dato miembro disponibles que en este
    caso hace
    *referencia a el número de prodcutos del mismo tipo que se
    cotizaran
    */
    public String calculaIVADeProducto(ProductoDTO producto) {
        Float iva=producto.getPrecioUnitario()*_IVA;
        System.out.println("Este es el IVA:"+iva);
        return iva.toString();
    }
}

```

```

    /**Recibe como parámetro un objeto de tipo ProductoDTO y
    calcula el subtotal
    *en base a su precio y el dato miembro disponibles que en este
    caso hace
    *referencia a el número de productos del mismo tipo que se
    cotizaran
    */
    public float calculaSubtotalDeProducto(ProductoDTO producto) {
        float
subtotal=(producto.getPrecioUnitario()*producto.getDisponibles()*(1-
_IVA));
        return subtotal;
    }

    /**Recibe una lista de objetos de tipo ProductoEnCarrito y
    obtiene el iva de todos
    los productos cotizados*/
    public float calculaIvaGeneral(LinkedList<ProductoEnCarrito>
productos) {
        int size=productos.size();
        float ivaTotal=(float)0.0;
        ProductoEnCarrito aux=new ProductoEnCarrito();
        for(int i=0;i<size;i++) {
            aux=(ProductoEnCarrito)productos.get(i);
            ivaTotal+=aux.calculaIva();
        }
        return ivaTotal;
    }

    /**Recibe una lista de objetos de tipo ProductoEnCarrito y
    obtiene el importe parcial
    * de todos los productos cotizados*/
    public float
calculaImporteParcialGeneral(LinkedList<ProductoEnCarrito> productos)
{
        int size=productos.size();
        float importeParcial=(float)0.0;
        ProductoEnCarrito aux=new ProductoEnCarrito();
        for(int i=0;i<size;i++) {
            aux=(ProductoEnCarrito)productos.get(i);
            importeParcial+=aux.calculaImporteParcial();
        }
        return importeParcial;
    }

    /**Recibe una lista de objetos de tipo ProductoEnCarrito y
    obtiene el
    * importe total de todos los productos cotizados. En base al
    iva y al
    * importe parcial*/
    public float
calculaImporteTotalGeneral(LinkedList<ProductoEnCarrito> productos) {
        return calculaImporteParcialGeneral(productos)
+calculaIvaGeneral(productos);
    }

    /**Recibe una lista de objetos de tipo ProductoEnCarrito y un
    objeto de
    *tipo cotizacion y llena algunos elementos de este en base a
    la lista de

```

```

        *productos
        */
        public CotizacionDTO cotiza(LinkedList<ProductoEnCarrito>
productos,CotizacionDTO cotizacion) {
            cotizacion.setIva(calculaIvaGeneral(productos));

cotizacion.setImporteParcial(calculaImporteParcialGeneral(productos))
;

cotizacion.setImporteTotal(calculaImporteTotalGeneral(productos));
            return cotizacion;
        }

        /*Recibe un carrito de compras y regresa un lista de objetos de
tipo
        * ProductoEnCotizacion que es el la representacion de objetos
tipo ProductoEnCarrito
        */
        LinkedList<ProductoEnCotizacionDTO>
transformaDeCarritoACotizacion(CarritoDeCompras carrito,Integer
idCotizacion) {
            ProductoEnCarrito prCarrito=new ProductoEnCarrito();
            ProductoEnCotizacionDTO prCotizacion=new
ProductoEnCotizacionDTO();
            prCotizacion.setIdCotizacion(idCotizacion);
            LinkedList<ProductoEnCotizacionDTO>
productosEnCotizacion=new LinkedList();
            LinkedList<ProductoEnCarrito>
productosEnCarrito=carrito.getListaProductos();
            int numeroDeProductos=carrito.numeroDeProductos();
            for(int i=0;i<numeroDeProductos;i++){
                prCarrito=productosEnCarrito.get(i);

                prCotizacion.setIdProducto(prCarrito.getProducto().getId());
                prCotizacion.setCantidad(prCarrito.getCantidad());

prCotizacion.setPrecioUnitario(prCarrito.getProducto().getPrecioUnita
rio());

                prCotizacion.setImporte(prCarrito.getImporteParcial());
                prCotizacion.setIva(prCarrito.getIva());
                prCotizacion.setTotal(prCarrito.getImporteTotal());
                //System.out.println(prCotizacion.toString());
                productosEnCotizacion.add(prCotizacion);
                prCotizacion=new ProductoEnCotizacionDTO();
                prCarrito=new ProductoEnCarrito();
            }
            return productosEnCotizacion ;
        }
    }
}

```

Facturador.java

```
package com.uam.cofa.business.local;
```

```

import com.uam.cofa.data.facturasCotizaciones.CotizacionDTO;
import com.uam.cofa.data.facturasCotizaciones.FacturaDTO;
import java.util.Calendar;
import java.util.GregorianCalendar;

/**
 * @author Eduardo Gonzalez
 * Contiene métodos necesarios para generar una nueva factura*/

public class Facturador {

    /*Recibe el último folio generado en una factura y regresa el
folio recibido incrementado en uno*/
    public Integer generaFolio(Integer folioAnterior){
        return folioAnterior+1;
    }

    public Integer generaNumeroDeSerie(Integer numeroDeSerie){
        return numeroDeSerie+1;
    }

    public String generaNumeroDeAprobacion(Integer folio){
        Calendar c = new GregorianCalendar();
        String year=Integer.toString(c.get(Calendar.YEAR));
        int num=folio+1;
        return ""+year+"-"+num;
    }

    /*Genera un factura en base a la cotizacion enviada y a la ultima
factura registrada*/
    public FacturaDTO generaFactura(CotizacionDTO
cotizacion,FacturaDTO ultimaFactura){
        FacturaDTO factura=new FacturaDTO();
        factura.setIdEstado(1);
        factura.setIdUsuario(cotizacion.getIdUsuario());
        factura.setIdCotizacion(cotizacion.getId());

        factura.setSerie(generaNumeroDeSerie(ultimaFactura.getSerie()));
        factura.setFolio(generaFolio(ultimaFactura.getFolio()));

        factura.setNumeroDeaprobacion(generaNumeroDeAprobacion(factura.getFolio()));
        return factura;
    }
}

```

ValidadorUsuariosImpl.java

```

package com.uam.cofa.business.local;

import com.uam.cofa.business.ValidadorUsuarios;
import com.uam.cofa.data.componentesUsuario.DatosDeAccesoDTO;
import com.uam.cofa.data.componentesUsuario.UsuarioDTO;
import java.util.List;

```



```

import org.hibernate.*;
import java.sql.*;
import org.hibernate.criterion.*;
import com.uam.cofa.utilities.EncriptadorPassword;

/**
 * @author Eduardo Gonzalez
 * Implementa los métodos declarados en la interfaz
 * com.uam.cofa.business.ValidadorUsuarios
 */

public class ValidadorUsuariosImpl implements ValidadorUsuarios
{
    private SessionFactory _sessionFactory;

    public List<DatosDeAccesoDTO> listarDatosDeUsuario ()
    {
        System.out.println("listando datos de usuario");
        Session session = getSession();
        Query query = session.createQuery( "SELECT t FROM
DatosDeAccesoDTO t" );
        return query.list();
    }

    public Integer validarUsuario(DatosDeAccesoDTO datosDeAcceso) {
        Integer idUsuario=new Integer(-1);
        System.out.println("validando usuario");
        EncriptadorPassword enc=new EncriptadorPassword();
        String login=datosDeAcceso.getLogin();
        Session session = getSession();
        Query query = session.createQuery( "SELECT t FROM
DatosDeAccesoDTO t where t.login='"+login+"'" );
        query.setMaxResults(1);
        DatosDeAccesoDTO datos = (DatosDeAccesoDTO)
query.uniqueResult();

        System.out.println(enc.encriptaMD5(datosDeAcceso.getPassword()));
        if(datos!=null){

            if(enc.encriptaMD5(datosDeAcceso.getPassword()).equals(datos.getPassw
ord()))
                idUsuario=datos.getIdUsuario();
        }
        return idUsuario;
    }

    /*Busca un usuario que corresponda con el id mandado como
parámetro*/
    public UsuarioDTO buscarUsuario(Integer idUsuario){
        int id=idUsuario;
        UsuarioDTO usuario=new UsuarioDTO();
        Session session = getSession();
        String s_query="SELECT t FROM UsuarioDTO t where t.id="+id;
        System.out.println("el query para buscar un usuario:
"+s_query);
        Query query = session.createQuery(s_query);
        query.setMaxResults(1);
        usuario= (UsuarioDTO) query.uniqueResult();
        System.out.println(usuario.toString());
    }
}

```

```

        return usuario;
    }

    public void setSessionFactory( SessionFactory sessionFactory )
    {
        _sessionFactory = sessionFactory;
    }

    private Session getSession()
    {
        return _sessionFactory.getCurrentSession();
    }
}

```

com.uam.cofa.data.componentesUsuario

DatosDeAccesoDTO.java

```

package com.uam.cofa.data.componentesUsuario;

import java.io.Serializable;
import org.hibernate.annotations.Proxy;
import javax.persistence.*;
import static javax.persistence.GenerationType.AUTO;
/**
 * @author Eduardo Gonzalez
 */
@Entity @Table( name="datosdeacceso" )
public class DatosDeAccesoDTO implements Serializable
{
    private Integer _id;
    private String _login;
    private String _password;
    private Integer _idUserario;

    /**
     * @return the id
     */
    @Id @Column( name="id" ) @GeneratedValue( strategy=AUTO )
    public Integer getId()
    {
        return _id;
    }

    /**
     * @return the login
     */
    @Column( name="login" )
    public String getLogin()
    {
        return _login;
    }

    /**
     * @return the password

```

```

    */
    @Column( name="password" )
    public String getPassword()
    {
        return _password;
    }

    @Column(name="Usuarios_id")
    public Integer getIdUsuario() {
        return _idUsuario;
    }

    /**
     * @param id the id to set
     */
    public void setId( int id )
    {
        _id = id;
    }

    /**
     * @param login the login to set
     */
    public void setLogin( String login )
    {
        _login = login;
    }

    /**
     * @param password the password to set
     */
    public void setPassword( String password )
    {
        _password = password;
    }

    public void setIdUsuario(Integer usuarios_id) {
        _idUsuario = usuarios_id;
    }

    @Override
    public String toString() {
        return "DatosDeAccesoDTO [_id=" + _id + ", _login=" + _login
            + ", _password=" + _password + ", _idUsuario=" +
            _idUsuario + " ]";
    }
}

```

DireccionDTO.java

```

package com.uam.cofa.data.componentesUsuario;

import java.io.Serializable;
import org.hibernate.annotations.Proxy;
import javax.persistence.*;

```

```

import static javax.persistence.GenerationType.AUTO;

/**
 * @author Eduardo Gonzalez
 */

@Entity @Table( name="direcciones" )
public class DireccionDTO implements Serializable{

    private Integer _id;
    private Integer _idUser;
    private String _calle ;
    private String _numeroInt;
    private String _numeroExt ;
    private String _colonia ;
    private String _ciudad ;
    private String _codigoPostal;

    /*Declaracion de getters*/

    @Id @Column( name="id" ) @GeneratedValue( strategy=AUTO )
    public Integer getId() {
        return _id;
    }

    @Column (name="Usuarios_id")
    public Integer getIdUsuario() {
        return _idUser;
    }

    @Column (name="calle")
    public String getCalle() {
        return _calle;
    }

    @Column (name="numeroInt")
    public String getNumeroInt() {
        return _numeroInt;
    }

    @Column (name="numeroExt")
    public String getNumeroExt() {
        return _numeroExt;
    }

    @Column (name="colonia")
    public String getColonia() {
        return _colonia;
    }

    @Column (name="ciudad")
    public String getCiudad() {
        return _ciudad;
    }

    @Column (name="codigoPostal")
    public String getCodigoPostal() {
        return _codigoPostal;
    }
}

```

```

/*Declaracion de setters*/

public void setId(Integer id) {
    this._id = id;
}
public void setIdUsuario(Integer idUsuario) {
    this._idUsuario = idUsuario;
}
public void setCalle(String calle) {
    this._calle = calle;
}
public void setNumeroInt(String numeroInt) {
    this._numeroInt = numeroInt;
}
public void setNumeroExt(String numeroExt) {
    this._numeroExt = numeroExt;
}
public void setColonia(String colonia) {
    this._colonia = colonia;
}
public void setCiudad(String ciudad) {
    this._ciudad = ciudad;
}

public void setCodigoPostal(String codigoPostal) {
    this._codigoPostal = codigoPostal;
}

@Override
public String toString() {
    return "DireccionDTO [_id=" + _id + ", _idUsuario=" +
_idUsuario
        + ", _calle=" + _calle + ", _numeroInt=" +
_numeroInt
        + ", _numeroExt=" + _numeroExt + ",
_colonia=" + _colonia
        + ", _ciudad=" + _ciudad + ", _codigoPostal="
+ _codigoPostal
        + " ]";
}
}

```

MedioDeContactoDTO.java

```

package com.uam.cofa.data.componentesUsuario;

import java.io.Serializable;
import javax.persistence.*;
import static javax.persistence.GenerationType.AUTO;

@Entity @Table (name="modosdecontacto")
public class MedioDeContactoDTO implements Serializable{

```

```

private Integer _id;
private Integer _idTipoDeContacto;
private Integer _idUsuario;
private String _contacto;

public MedioDeContactoDTO(int idTipo,String contacto){
    _idTipoDeContacto=new Integer(idTipo);
    _contacto=contacto;
}

public MedioDeContactoDTO(){

}

/*Métodos get*/
@Id @Column(name="id") @GeneratedValue(strategy=AUTO)
public Integer getId() {
    return _id;
}

@Column (name="TiposDeContaco_id")
public Integer getIdTipoDeContacto() {
    return _idTipoDeContacto;
}

@Column (name="Usuarios_id")
public Integer getIdUsuario() {
    return _idUsuario;
}

@Column (name="contacto")
public String getContacto() {
    return _contacto;
}

/*Métodos set*/

public void setId(Integer _id) {
    this._id = _id;
}

public void setIdTipoDeContacto(Integer _idTipoDeContacto) {
    this._idTipoDeContacto = _idTipoDeContacto;
}

public void setIdUsuario(Integer _idUsuario) {
    this._idUsuario = _idUsuario;
}

public void setContacto(String _contacto) {
    this._contacto = _contacto;
}

@Override
public String toString() {
    return "MedioDeContactoDTO [_id=" + _id + ",
_idTipoDeContacto="
        + _idTipoDeContacto + ", _idUsuario=" +
_idUsuario
        + ", _contacto=" + _contacto + " ]";
}
}

```

OcupacionDTO.java

```
package com.uam.cofa.data.componentesUsuario;

import java.io.Serializable;
import org.hibernate.annotations.Proxy;
import javax.persistence.*;
import static javax.persistence.GenerationType.AUTO;

/**
 * @author Eduardo Gonzalez
 */

@Entity @Table(name="Ocupaciones")
public class OcupacionDTO implements Serializable{

    private Integer _id;
    private String _ocupacion;

    @Id @Column( name="id" ) @GeneratedValue( strategy=AUTO )
    public Integer getId() {
        return _id;
    }

    @Column (name="ocupacion")
    public String getOcupacion() {
        return _ocupacion;
    }

    public void setId(int _id) {
        this._id = _id;
    }

    public void setOcupacion(String _ocupacion) {
        this._ocupacion = _ocupacion;
    }

}
```

UsuarioDTO.java

```
package com.uam.cofa.data.componentesUsuario;

import java.io.Serializable;
import org.hibernate.annotations.Proxy;
import javax.persistence.*;
import static javax.persistence.GenerationType.AUTO;
import com.uam.cofa.data.*;

/**
 * @author Eduardo Gonzalez
 */

@Entity @Table( name="usuarios" )
```

```

public class UsuarioDTO implements Serializable{

    private Integer _id;
    private String _nombres;
    private String _apellidoPaterno;
    private String _apellidoMaterno;
    private Integer _idOcupacion;
    private Integer _idTipoDeUsuario;
    private Integer _edad;
    private String _rfc;

    @Id @Column( name="id" ) @GeneratedValue( strategy=AUTO )
    public Integer getId() {
        return _id;
    }
    public void setId(Integer id) {
        this._id = id;
    }

    @Column( name="nombres" )
    public String getNombres() {
        return _nombres;
    }
    public void setNombres(String nombres) {
        this._nombres = nombres;
    }

    @Column (name="apellidoPaterno")
    public String getApellidoPaterno() {
        return _apellidoPaterno;
    }
    public void setApellidoPaterno(String apellidoPaterno) {
        this._apellidoPaterno = apellidoPaterno;
    }

    @Column (name="apellidoMaterno")
    public String getApellidoMaterno() {
        return _apellidoMaterno;
    }
    public void setApellidoMaterno(String apellidoMaterno) {
        this._apellidoMaterno = apellidoMaterno;
    }

    @Column (name="Ocupaciones_id")
    public Integer getIdOcupacion() {
        return _idOcupacion;
    }
    public void setIdOcupacion(Integer idOcupacion) {
        this._idOcupacion = idOcupacion;
    }

    @Column (name="TiposDeUsuario_id")
    public Integer getIdTipoDeUsuario() {
        return _idTipoDeUsuario;
    }
    public void setIdTipoDeUsuario(Integer idTipoDeUsuario) {
        this._idTipoDeUsuario = idTipoDeUsuario;
    }
}

```



```

@Column (name="edad")
public Integer getEdad() {
    return _edad;
}
public void setEdad(Integer edad) {
    this._edad = edad;
}

@Column (name="RFC")
public String getRfc() {
    return _rfc;
}
public void setRfc(String rfc) {
    this._rfc = rfc;
}

@Override
public String toString() {
    return "UsuarioDTO [_id=" + _id + ", _nombres=" +
_nombres
        + ", _apellidoPaterno=" + _apellidoPaterno
        + ", _apellidoMaterno=" + _apellidoMaterno +
", _idOcupacion="
        + _idOcupacion + ", _idTipoDeUsuario=" +
_idTipoDeUsuario
        + ", _edad=" + _edad + ", _rfc=" + _rfc +
"]";
}
}
}

```

com.uam.cofa.data.facturasCotizaciones

CotizacionDTO.java

```

package com.uam.cofa.data.facturasCotizaciones;

import java.io.Serializable;
import javax.persistence.*;
import static javax.persistence.GenerationType.AUTO;
import java.sql.Timestamp;
import java.sql.Date;
import java.util.StringTokenizer;

@Entity @Table(name="cotizaciones")
public class CotizacionDTO implements Serializable {

    private Integer _id;
    private Integer _tipo=new Integer(1);
    private Integer _idUserio;
    private Timestamp _fechaDeCotizacion;
    private Date _fechaDeCaducidad=new Date(2012,11,20);
    private Float _importeParcial;
}

```

```

private Float _importeTotal;
private Float _descuentoTotal=new Float(0.0);
private Float _iva;

/* ----- Metodos get ----- */

@Id @Column (name="id") @GeneratedValue (strategy=AUTO)
public Integer getId() {
    return _id;
}

@Column (name="TipoDeCotizacion_id")
public Integer getTipo() {
    return _tipo;
}

@Column (name="Usuarios_id")
public Integer getIdUsuario() {
    return _idUsuario;
}

@Column (name="fechaDeCotizacion") @GeneratedValue
(strategy=AUTO)
public Timestamp getFechaDeCotizacion() {
    return _fechaDeCotizacion;
}

@Column (name="fechaDeVigencia")
public Date getFechaDeCaducidad() {
    return _fechaDeCaducidad;
}

@Column (name="importeParcial")
public Float getImporteParcial() {
    return _importeParcial;
}

@Column (name="importeTotal")
public Float getImporteTotal() {
    return _importeTotal;
}

@Column (name="descuentoTotal")
public Float getDescuentoTotal() {
    return _descuentoTotal;
}

@Column (name="IVA")
public Float getIva() {
    return _iva;
}

/*----- Metodos set -----*/

public void setId(Integer id) {
    this._id = id;
}

public void setTipo(Integer tipo) {
    this._tipo = tipo;
}

public void setIdUsuario(Integer idUsuario) {

```

```

        this._idUserario = idUsuario;
    }
    public void setFechaDeCotizacion(Timestamp fechaDeCotizacion) {
        this._fechaDeCotizacion = fechaDeCotizacion;
    }
    public void setFechaDeCaducidad(Date fechaDeCaducidad) {
        this._fechaDeCaducidad = fechaDeCaducidad;
    }
    public void setImporteParcial(Float importeParcial) {
        this._importeParcial = importeParcial;
    }
    public void setImporteTotal(Float importeTotal) {
        this._importeTotal = importeTotal;
    }
    public void setDescuentoTotal(Float descuentoTotal) {
        this._descuentoTotal = descuentoTotal;
    }
    public void setIva(Float iva) {
        this._iva = iva;
    }
}

/*----- Termina la declaracion de metodos set -----*/

@Override
public String toString() {
    return "Cotizacion [_id=" + _id + ", _tipo=" + _tipo + ",
_idUsuario="
        + _idUserario + ", _fechaDeCotizacion=" +
_fechaDeCotizacion
        + ", _fechaDeCaducidad=" + _fechaDeCaducidad
        + ", _importeParcial=" + _importeParcial + ",
_importeTotal="
        + _importeTotal + ", _descuentoTotal=" +
_descuentoTotal
        + ", _iva=" + _iva + "];"
}

public String serializar(){
    StringBuilder aux=new StringBuilder();
    aux.append(_id);aux.append("/**/");
    aux.append(_tipo);aux.append("/**/");
    aux.append(_idUserario);aux.append("/**/");
    aux.append(_fechaDeCotizacion);aux.append("/**/");
    aux.append(_fechaDeCaducidad);aux.append("/**/");
    aux.append(_importeParcial);aux.append("/**/");
    aux.append(_importeTotal);aux.append("/**/");
    aux.append(_descuentoTotal);aux.append("/**/");
    aux.append(_iva);aux.append("/**/");
    return aux.toString();
}

public CotizacionDTO deserializar(String cotizacionSerializada)
{
    CotizacionDTO aux=new CotizacionDTO();
    StringTokenizer st=new
StringTokenizer(cotizacionSerializada,"/**/");
    aux.setId(new Integer(st.nextToken()));
    aux.setTipo(new Integer(st.nextToken()));
    aux.setIdUsuario(new Integer(st.nextToken()));
}

```

```

        aux.setFechaDeCotizacion(Timestamp.valueOf(st.nextToken()));
        aux.setFechaDeCaducidad(Date.valueOf(st.nextToken()));
        aux.setImporteParcial(new Float(st.nextToken()));
        aux.setImporteTotal(new Float(st.nextToken()));
        aux.setDescuentoTotal(new Float(st.nextToken()));
        aux.setIva(new Float(st.nextToken()));
        return aux;
    }
}

```

FacturaDTO.java

```

package com.uam.cofa.data.facturasCotizaciones;

import java.sql.Timestamp;
import java.sql.Date;
import java.util.StringTokenizer;
import java.io.Serializable;
import javax.persistence.*;
import static javax.persistence.GenerationType.AUTO;

@Entity @Table(name="facturas")
public class FacturaDTO implements Serializable{

    private Integer _id;
    private Integer _idEstado;
    private Integer _idUsuario;
    private Integer _idCotizacion;
    private Integer _serie;
    private Integer _folio;
    private Timestamp _fechaDeExpedicion;
    private Date _fechaLimiteDePago;
    private String _numeroDeaprobacion;
    private Date _fechaDePago;

    /*Metodos get*/

    @Id @Column(name="id") @GeneratedValue(strategy=AUTO)
    public Integer getId() {
        return _id;
    }

    @Column(name="EstadosFactura_id")
    public Integer getIdEstado() {
        return _idEstado;
    }

    @Column(name="Usuarios_id")
    public Integer getIdUsuario() {
        return _idUsuario;
    }

    @Column(name="Cotizaciones_id")
    public Integer getIdCotizacion() {
        return _idCotizacion;
    }
}

```

```

@Column(name="serie")
public Integer getSerie() {
    return _serie;
}

@Column(name="folio")
public Integer getFolio() {
    return _folio;
}

@Column(name="fechaDeExpedicion")
@GeneratedValue(strategy=AUTO)
public Timestamp getFechaDeExpedicion() {
    return _fechaDeExpedicion;
}

@Column(name="fechaLimiteDePago")
public Date getFechaLimiteDePago() {
    return _fechaLimiteDePago;
}

@Column(name="numeroDeAprobacion")
public String getNumeroDeaprobacion() {
    return _numeroDeaprobacion;
}

@Column(name="fechaDePago")
public Date getFechaDePago() {
    return _fechaDePago;
}

/*Metodos set*/

public void setId(Integer _id) {
    this._id = _id;
}

public void setIdEstado(Integer _idEstado) {
    this._idEstado = _idEstado;
}

public void setIdUsuario(Integer _idUsuario) {
    this._idUsuario = _idUsuario;
}

public void setIdCotizacion(Integer _idCotizacion) {
    this._idCotizacion = _idCotizacion;
}

public void setSerie(Integer _serie) {
    this._serie = _serie;
}

public void setFolio(Integer _folio) {
    this._folio = _folio;
}

public void setFechaDeExpedicion(Timestamp _fecha_deExpedicion)
{
    this._fechaDeExpedicion = _fecha_deExpedicion;
}

public void setFechaLimiteDePago(Date _fechaLimiteDePago) {
    this._fechaLimiteDePago = _fechaLimiteDePago;
}

public void setNumeroDeaprobacion(String _numeroDeaprobacion) {

```

```

        this._numeroDeaprobacion = _numeroDeaprobacion;
    }
    public void setFechaDePago(Date _fechaDePago) {
        this._fechaDePago = _fechaDePago;
    }

    public String serializar(){
        StringBuilder aux=new StringBuilder();
        aux.append(_id);aux.append("/**/");
        aux.append(_idEstado);aux.append("/**/");
        aux.append(_idUsuario);aux.append("/**/");
        aux.append(_idCotizacion);aux.append("/**/");
        aux.append(_serie);aux.append("/**/");
        aux.append(_folio);aux.append("/**/");

        aux.append(_fechaDeExpedicion.toString());aux.append("/**/");
        aux.append(_numeroDeaprobacion);aux.append("/**/");
        return aux.toString();
    }

    public FacturaDTO deserializar(String facturaSerializada){
        FacturaDTO factura=new FacturaDTO();
        StringTokenizer st=new
StringTokenizer(facturaSerializada,"/**/");
        factura.setId(new Integer(st.nextToken()));
        factura.setIdEstado(new Integer(st.nextToken()));
        factura.setIdUsuario(new Integer(st.nextToken()));
        factura.setIdCotizacion(new Integer(st.nextToken()));
        factura.setSerie(new Integer(st.nextToken()));
        factura.setFolio(new Integer(st.nextToken()));

        factura.setFechaDeExpedicion(Timestamp.valueOf(st.nextToken()));
        factura.setNumeroDeaprobacion(st.nextToken());
        return factura;
    }

    /*Termino de getters and setters*/

    @Override
    public String toString() {
        return "FacturaDTO [_id=" + _id + ", _idEstado=" +
_idEstado
        + ", _idUsuario=" + _idUsuario + ",
_idCotizacion="
        + _idCotizacion + ", _serie=" + _serie + ",
_folio=" + _folio
        + ", _fechaDeExpedicion=" +
_fechaDeExpedicion
        + ", _fechaLimiteDePago=" +
_fechaLimiteDePago
        + ", _numeroDeaprobacion=" +
_numeroDeaprobacion
        + ", _fechaDePago=" + _fechaDePago + "];"
    }
}

```

ProductoEnCotizacionDTO.java

```
package com.uam.cofa.data.facturasCotizaciones;

import java.io.Serializable;
import javax.persistence.*;

import static javax.persistence.GenerationType.AUTO;

@Entity @Table(name="productosencotizaciones")
public class ProductoEnCotizacionDTO implements Serializable{

    private Integer _id;
    private Integer _idProducto;
    private Float _precioUnitario;
    private Integer _cantidad;
    private Float _iva;
    private Float _importe;
    private Float _total;
    private Integer _idCotizacion;

    /*Metodos get*/
    @Id @Column( name="id" ) @GeneratedValue( strategy=AUTO )
    public Integer getId() {
        return _id;
    }

    @Column( name="id_producto")
    public Integer getIdProducto() {
        return _idProducto;
    }

    @Column( name="precio_unitario")
    public Float getPrecioUnitario() {
        return _precioUnitario;
    }

    @Column( name="cantidad")
    public Integer getCantidad() {
        return _cantidad;
    }

    @Column( name="iva")
    public Float getIva() {
        return _iva;
    }

    @Column( name="importe")
    public Float getImporte() {
        return _importe;
    }

    @Column( name="total")
    public Float getTotal() {
        return _total;
    }

    @Column( name="id_cotizacion")
    public Integer getIdCotizacion() {
```

```

        return _idCotizacion;
    }

    /*Metodos set*/

    public void setId(Integer id) {
        this._id = id;
    }
    public void setIdProducto(Integer idProducto) {
        this._idProducto = idProducto;
    }
    public void setPrecioUnitario(Float precioUnitario) {
        this._precioUnitario = precioUnitario;
    }
    public void setCantidad(Integer cantidad) {
        this._cantidad = cantidad;
    }
    public void setIva(Float iva) {
        this._iva = iva;
    }
    public void setImporte(Float importe) {
        this._importe = importe;
    }
    public void setTotal(Float total) {
        this._total = total;
    }
    public void setIdCotizacion(Integer idCotizacion) {
        this._idCotizacion = idCotizacion;
    }
}

/*Termino de la declaracion de los metodos set*/

@Override
public String toString() {
    return "ProductoEnCotizacionDTO [_id=" + _id + ",
_idProducto="
        + _idProducto + ", _precioUnitario=" +
_precioUnitario
        + ", _cantidad=" + _cantidad + ", _iva=" +
_iva + ", _importe="
        + _importe + ", _total=" + _total + ",
_idCotizacion="
        + _idCotizacion + "];"
    }
}

```

SubtipoProductoDTO.java

```

package com.uam.cofa.data.productosServicios;

import java.io.Serializable;
import javax.persistence.*;
import static javax.persistence.GenerationType.AUTO;

```



```

@Entity @Table(name="subtiposdeproductos")
public class SubtipoProductoDTO implements Serializable{

    private Integer _id;
    private Integer _idTipoDeProducto;
    private String _subtipo;

    @Id @Column(name="id") @GeneratedValue (strategy=AUTO)
    public Integer getId() {
        return _id;
    }

    @Column (name="TipoDeProducto_id")
    public Integer getIdTipoDeProducto() {
        return _idTipoDeProducto;
    }

    @Column (name="subtipo")
    public String getSubtipo() {
        return _subtipo;
    }

    public void setId(Integer _id) {
        this._id = _id;
    }

    public void setIdTipoDeProducto(Integer _tipoDeProducto) {
        this._idTipoDeProducto = _tipoDeProducto;
    }

    public void setSubtipo(String _subtipo) {
        this._subtipo = _subtipo;
    }

}

```

TipoProductoDTO.java

```

package com.uam.cofa.data.productosServicios;

import java.io.Serializable;
import javax.persistence.*;
import static javax.persistence.GenerationType.AUTO;

@Entity @Table(name="tipodeproducto")
public class TipoProductoDTO {

    private Integer _id;
    private String _tipo;

    @Id @Column(name="id") @GeneratedValue (strategy=AUTO)
    public Integer getId() {
        return _id;
    }

```

```

    }

    @Column (name="tipo")
    public String getTipo() {
        return _tipo;
    }

    public void setId(Integer _id) {
        this._id = _id;
    }
    public void setTipo(String _tipo) {
        this._tipo = _tipo;
    }
}

}

}



---



package com.uam.cofa.data.productosServicios;

import java.util.LinkedList;

/**
 * La clase representa un listaProductos de compras que guarda
 * objetos de tipo ProductoDTO.
 * @author (Eduardo Gonzalez)
 * @version (1.0)
 */

public class CarritoDeCompras {

    private LinkedList<ProductoEnCarrito> _listaProductos=new
LinkedList();

    public LinkedList<ProductoEnCarrito> getListaProductos() {
        return _listaProductos;
    }

    public void setListaProductos(LinkedList<ProductoEnCarrito>
_listaProductos) {
        this._listaProductos = _listaProductos;
    }

    /**
     * @return boolean
     * @param producto
     *Agrega un objeto de tipo ProductoEnCarrito a la
listaProductos de compras si no esta ya agregado.
     *Agrega el prodcuto si no esta ya agregado y regresa true,
de lo contrario no lo agrega
     *y regresa false
     */
    public boolean agregar(ProductoDTO producto){
        ProductoEnCarrito aux=new ProductoEnCarrito();
        aux.setProducto(producto);
        aux.setIdProducto(producto.getId());
        aux.setNombre(producto.getNombre());
        aux.setPrecioUnitario(producto.getPrecioUnitario());
    }
}

```

```

        if(estaEnListaProductos(producto)!=-1) return false;
        else _listaProductos.add(aux);
        return true;
    }

    /**
     * @param indice
     * @param cantidad
     * Establece la cantidad de productos que se escogieran de
cada tipo
     * Recibe como parametros el id del del objeto de tipo
ProductoEnCarrito y
     * la cantidad de prodcutos que se desean. El lugar donde
se pone la
     * cantidad es el dato miembro (de
ProductoEnCarrito)cantidad.
     */
    public void estableceCantidad(int indice,int cantidad){
        ProductoEnCarrito aux=_listaProductos.get(indice);
        aux.setCantidad(cantidad);
        _listaProductos.set(indice,aux);
    }

    /**
     * @return boolean
     * Recibe un objeto de tipo PorductoEnCarrito, si se
encuentra el
     * objeto recibido lo remueve del listaProductos y reresas
true,
     * de lo contrario regresa false*/
    public boolean remover(ProductoDTO producto){
        ProductoEnCarrito aux=new ProductoEnCarrito();
        aux.setProducto(producto);
        return _listaProductos.remove(aux);
    }

    /**Recibe un objeto de tipo ProductoDTO y regresa el indice
en que se encuentra*/
    public int estaEnListaProductos(ProductoDTO producto){
        ProductoEnCarrito aux=new ProductoEnCarrito();
        aux.setProducto(producto);
        int indice;
        indice=_listaProductos.indexOf(aux);
        return indice;
    }

    /**Regresa el tamaño de la lista _listaProductos*/
    public int numeroDeProductos(){
        return _listaProductos.size();
    }
}

```

MarcaDTO.java

```
package com.uam.cofa.data.productosServicios;
```

```

import java.io.Serializable;
import javax.persistence.*;
import static javax.persistence.GenerationType.AUTO;

@Entity @Table(name="marcasdeproductos")
public class MarcaDTO implements Serializable{

    private Integer _id;
    private String _marca;

    /*metodos get*/
    @Id @Column (name="id") @GeneratedValue(strategy=AUTO)
    public Integer getId() {
        return _id;
    }

    @Column(name="marca")
    public String getMarca() {
        return _marca;
    }

    /*Metodos set*/

    public void setId(Integer id) {
        this._id = id;
    }
    public void setMarca(String marca) {
        this._marca = marca;
    }

}

```

ProductoDTO.java

```

package com.uam.cofa.data.productosServicios;

import java.io.Serializable;
import java.util.Arrays;
import javax.persistence.*;
import static javax.persistence.GenerationType.AUTO;
import java.util.*;

/**
 * @author Eduardo Gonzalez
 */

@Entity @Table(name="productos")
public class ProductoDTO implements Serializable{

    private Integer _id=new Integer(0);
    private String _nombre;
    private Integer _codigoDeBarras=new Integer(0);
    private Integer _descuento=new Integer(0);
    private Float _precioUnitario=new Float(0.0);

```

```

private String _descripcion;
private Integer _disponibles=new Integer(0);
private Integer _idMarca=new Integer(0);
private Integer _idSubtipo=new Integer(0);
private byte[] _imagen;
private Integer _estado=new Integer(1);

/*Metodos getters*/
@Id @Column( name="id" ) @GeneratedValue( strategy=AUTO )
public Integer getId() {
    return _id;
}

@Column (name="nombre")
public String getNombre() {
    return _nombre;
}

@Column (name="codigoDeBarras")
public Integer getCodigoDeBarras() {
    return _codigoDeBarras;
}

@Column (name="descuento")
public Integer getDescuento() {
    return _descuento;
}

@Column (name="precioUnitario")
public Float getPrecioUnitario() {
    return _precioUnitario;
}

@Column (name="descripcion")
public String getDescripcion() {
    return _descripcion;
}

@Column (name="disponibles")
public Integer getDisponibles() {
    return _disponibles;
}

@Column (name="MarcasDeProductos_id")
public Integer getIdMarca() {
    return _idMarca;
}

@Column (name="SubtiposDeProductos_id")
public Integer getIdSubtipo() {
    return _idSubtipo;
}

@Lob @Column (name="imagen")
public byte[] getImagen() {
    return _imagen;
}

@Column (name="estado")

```

```

public Integer getEstado() {
    return _estado;
}

/*Metodos Setters*/

public void setId(Integer _id) {
    this._id = _id;
}
public void setNombre(String _nombre) {
    this._nombre = _nombre;
}
public void setCodigoDeBarras(Integer _codigoDeBarras) {
    this._codigoDeBarras = _codigoDeBarras;
}
public void setDescuento(Integer _descuento) {
    this._descuento = _descuento;
}
public void setPrecioUnitario(Float _precioUnitario) {
    this._precioUnitario = _precioUnitario;
}
public void setDescripcion(String _descripcion) {
    this._descripcion = _descripcion;
}
public void setDisponibles(Integer _disponibles) {
    this._disponibles = _disponibles;
}
public void setIdMarca(Integer _idMarca) {
    this._idMarca = _idMarca;
}
public void setIdSubtipo(Integer _idSubtipo) {
    this._idSubtipo = _idSubtipo;
}

public void setImagen(byte[] _imagen) {
    this._imagen = _imagen;
}

public void setEstado(Integer _estado) {
    this._estado = _estado;
}

@Override
public String toString() {
    return "ProductoDTO [_id=" + _id + ", _nombre=" + _nombre
        + ", _codigoDeBarras=" + _codigoDeBarras + ",
    _descuento="
        + _descuento + ", _precioUnitario=" +
    _precioUnitario
        + ", _descripcion=" + _descripcion + ",
    _disponibles="
        + _disponibles + ", _idMarca=" + _idMarca +
    ", _idSubtipo="

```

```

        + _idSubtipo + ", _imagen=" +
Arrays.toString(_imagen) + "];
    }

    public String serializar(){
        StringBuilder aux=new StringBuilder();
        aux.append(_id);aux.append("/--/");
        aux.append(_nombre);aux.append("/--/");
        aux.append(_codigoDeBarras);aux.append("/--/");
        aux.append(_descuento);aux.append("/--/");
        aux.append(_precioUnitario.toString());aux.append("/--/");
        aux.append(_idMarca);aux.append("/--/");
        aux.append(_idSubtipo);aux.append("/--/");
        return aux.toString();
    }

    public ProductoDTO deserializar(String productoSerializado){
        ProductoDTO aux=new ProductoDTO();
        StringTokenizer st=new
StringTokenizer(productoSerializado,"/--/");
        aux.setId(new Integer(st.nextToken()));
        aux.setNombre(st.nextToken());
        aux.setCodigoDeBarras(new Integer(st.nextToken()));
        aux.setDescuento(new Integer(st.nextToken()));
        aux.setPrecioUnitario(new Float(st.nextToken()));
        aux.setIdMarca(new Integer(st.nextToken()));
        aux.setIdSubtipo(new Integer(st.nextToken()));
        return aux;
    }

    public boolean equals(Object obj){
        ProductoDTO aux=(ProductoDTO)obj;
        if(this._id.equals(aux.getId())) return true;
        return false;
    }
}
}

```

ProductoEnCarrito.java

```

package com.uam.cofa.data.productosServicios;

public class ProductoEnCarrito {

    private static final float _IVA=(float)0.16;

    private ProductoDTO _producto;
    private Integer _idProducto;
    private int _cantidad=1 ;
    private float _iva;
    private float _importeParcial;
    private float _importeTotal;
    private String _nombre;
    private Float _precioUnitario;
}

```

```

/*----- Metodos get -----*/
public ProductoDTO getProducto() {
    return _producto;
}

public Integer getIdProducto() {
    return _idProducto;
}

public int getCantidad() {
    return _cantidad;
}
public float getIva() {
    return _iva;
}
public float getImporteParcial() {
    return _importeParcial;
}
public float getImporteTotal() {
    return calculaImporteTotal();
}

public String getNombre() {
    return _nombre;
}

public Float getPrecioUnitario() {
    return _precioUnitario;
}
/*----- Metodos set -----*/

public void setProducto(ProductoDTO _producto) {
    this._producto = _producto;
}
public void setCantidad(int _cantidad) {
    this._cantidad = _cantidad;
}
public void setIva() {
    this._iva =
(_cantidad*_producto.getPrecioUnitario())*_IVA;
}
public void setImporteParcial(float _importeParcial) {
    this._importeParcial =
(_cantidad*_producto.getPrecioUnitario())*(1-_IVA);
}
public void setImporteTotal(float _total) {
    this._importeTotal = calculaImporteTotal();
}

public void setIdProducto(Integer _idProducto) {
    this._idProducto = _idProducto;
}

public void setNombre(String _nombre) {
    this._nombre = _nombre;
}

public void setPrecioUnitario(Float _precioUnitario) {
    this._precioUnitario = _precioUnitario;
}

```



```

    }

    /*-----terminan los metodos set-----*/

    @Override
    public String toString() {
        return "ProductoEnCarrito [_producto=" + _producto + ",
        _cantidad="
            + _cantidad + ", _iva=" + _iva + ",
        _importeParcial="
            + _importeParcial + ", _total=" +
        _importeTotal + "];";
    }

    @Override
    public boolean equals(Object obj){
        ProductoEnCarrito aux=(ProductoEnCarrito)obj;
        if(aux.getProducto().getId().equals(_producto.getId()))
            return true;
        return false;
    }

    public float calculaIva(){
        _iva=( _cantidad * _producto.getPrecioUnitario())*_IVA;
        return _iva;
    }

    public float calculaImporteParcial(){

        _importeParcial=( _cantidad*_producto.getPrecioUnitario())*(1-
        _IVA);
        return _importeParcial;
    }

    public float calculaImporteTotal(){
        _importeTotal= calculaIva()+calculaImporteParcial();
        return _importeTotal;
    }
}

```

com.uam.cofa.utilities

EncriptadorPassword.java

```

package com.uam.cofa.utilities;

import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;

/**
 * Contiene los metodos necesarios para encirptar el password de un
 usuario.

```

```

*
* @author (Eduardo Gonzalez)
* @version (07/02/2011)
*/

public class EncriptadorPassword {

    /**Recibe una cadena y regresa la representacion encriptada con
MD5
    de la cadena ingresada*/
    public String encriptaMD5(String s) {
        String r = null;
        try {
            if (s != null) {
                MessageDigest algorithm
=MessageDigest.getInstance("MD5");
                algorithm.reset();
                algorithm.update(s.getBytes());
                byte bytes[] = algorithm.digest();
                StringBuffer sb = new StringBuffer();
                for (int i = 0; i < bytes.length; i++) {
                    String hex = Integer.toHexString(0xff &
bytes[i]);

                    if (hex.length() == 1)
                        sb.append('0');
                    sb.append(hex);
                }
                r = sb.toString();
            }
        }
        catch (NoSuchAlgorithmException e) {
        }
        return r;
    }
}

```

GeneradorPDF.java

```

package com.uam.cofa.utilities;

import java.io.FileOutputStream;
import java.util.Date;
import java.util.List;
import java.util.Calendar;
import com.itextpdf.text.*;
import com.itextpdf.text.pdf.PdfPCell;
import com.itextpdf.text.pdf.PdfPTable;
import com.itextpdf.text.pdf.PdfWriter;
import com.uam.cofa.data.componentesUsuario.UsuarioDTO;
import com.uam.cofa.data.facturasCotizaciones.CotizacionDTO;

```

```

import com.uam.cofa.data.facturasCotizaciones.FacturaDTO;
import
com.uam.cofa.data.facturasCotizaciones.ProductoEnCotizacionDTO;

public class GeneradorPDF {

    //private static String FILE =
"C:/Users/Mario/Documents/workspace/COFA/WebContent/WEB-
INF/files/facturas/Factura";
    private static String FILE = "Factura";
    private static Font catFont = new
Font(Font.FontFamily.TIMES_ROMAN, 18,
Font.BOLD);
    private static Font redFont = new
Font(Font.FontFamily.TIMES_ROMAN, 12,
Font.NORMAL, BaseColor.RED);
    private static Font subFont = new
Font(Font.FontFamily.TIMES_ROMAN, 14,
Font.BOLD);
    private static Font smallBold = new
Font(Font.FontFamily.TIMES_ROMAN, 12,
Font.BOLD);

    public static String obtenFecha() {
        Calendar c=Calendar.getInstance();
        Date fecha= new Date();
        String dia = Integer.toString(c.get(Calendar.DATE));
        String mes = Integer.toString(c.get(Calendar.MONTH)+1);
        String annio = Integer.toString(c.get(Calendar.YEAR));
        String segundo = Integer.toString(c.get(Calendar.SECOND));
        String minuto = Integer.toString(c.get(Calendar.MINUTE));
        String hora = Integer.toString(c.get(Calendar.HOUR));
        return annio+"/"+mes+"/"+dia+" "+hora+":"+minuto+":segundo;
    }

    // iText allows to add metadata to the PDF which can be viewed
in your Adobe
    // Reader
    // under File -> Properties
    private static void addMetaData(Document document,String
autor,String asunto,String titulo) {
        document.addTitle(titulo);
        document.addSubject(asunto);
        document.addAuthor(autor);
        document.addCreator(autor);
    }

    private static void addTitlePage(Document document) throws
DocumentException {
        Paragraph preface = new Paragraph();
        preface.add(new Paragraph("Factura", catFont));
        addEmptyLine(preface, 1);
        preface.add(new Paragraph("Negocio con nombre",subFont));
        preface.add(new Paragraph("RFC: GOHE851004KL6",subFont));
        preface.add(new Paragraph("Domicilio fiscal: Calle 1 No.2
Delegacion Gustavo A Madero",subFont));
    }
}

```

```

        preface.add(new Paragraph("Telefono: 56243582", subFont));
        addEmptyLine(preface, 3);
        document.add(preface);
    }

    private static void addContent(Document document, FacturaDTO
factura, List<ProductoEnCotizacionDTO> productos, CotizacionDTO
cotizacion, UsuarioDTO usuario) throws DocumentException {
        Paragraph preface = new Paragraph();
        preface.add(new Paragraph("Folio: "+factura.getFolio(),
smallBold));
        preface.add(new Paragraph("Serie:
"+factura.getSerie(), smallBold));
        preface.add(new Paragraph("Lugar de expedicion: Mexico
DF", smallBold));
        preface.add(new Paragraph("Fecha de expedicion:
"+obtenFecha(), smallBold));
        preface.add(new Paragraph("Cliente:
"+usuario.getNombres()+" "+usuario.getApellidoPaterno()+"
"+usuario.getApellidoMaterno(), smallBold));
        preface.add(new Paragraph("RFC:
"+usuario.getRfc(), smallBold));
        //preface.add(new Paragraph("Direccion: ", smallBold));
        addEmptyLine(preface, 3);
        document.add(preface);
        PdfPTable table=createTable(productos, cotizacion);
        document.add(table);
    }

    private static PdfPTable
createTable(List<ProductoEnCotizacionDTO> productos, CotizacionDTO
cotizacion) throws BadElementException {
        PdfPTable table = new PdfPTable(6);

        PdfPCell c1 = new PdfPCell(new Phrase("Producto"));
        c1.setHorizontalAlignment(Element.ALIGN_CENTER);
        table.addCell(c1);

        c1 = new PdfPCell(new Phrase("Cantidad"));
        c1.setHorizontalAlignment(Element.ALIGN_CENTER);
        table.addCell(c1);

        c1 = new PdfPCell(new Phrase("Precio unitario"));
        c1.setHorizontalAlignment(Element.ALIGN_CENTER);
        table.addCell(c1);

        c1 = new PdfPCell(new Phrase("IVA"));
        c1.setHorizontalAlignment(Element.ALIGN_CENTER);
        table.addCell(c1);

        c1 = new PdfPCell(new Phrase("Importe"));
        c1.setHorizontalAlignment(Element.ALIGN_CENTER);

```

```

table.addCell(c1);

c1 = new PdfPCell(new Phrase("Total"));
c1.setHorizontalAlignment(Element.ALIGN_CENTER);
table.addCell(c1);

table.setHeaderRows(1);
int size=productos.size();
for(int i=0;i<size;i++){
    ProductoEnCotizacionDTO producto=productos.get(i);
    table.addCell(producto.getIdProducto().toString());
    table.addCell(producto.getCantidad().toString());

table.addCell("$"+producto.getPrecioUnitario().toString());
    table.addCell("$"+producto.getIva().toString());
    table.addCell("$"+producto.getImporte().toString());
    table.addCell("$"+producto.getTotal().toString());
}

table.addCell("");
table.addCell("");
table.addCell("TOTALES");
table.addCell("$"+cotizacion.getIva().toString());
table.addCell("$"+cotizacion.getImporteParcial().toString());
table.addCell("$"+cotizacion.getImporteTotal().toString());

return table;
}

private static void addEmptyLine(Paragraph paragraph, int
number) {
    for (int i = 0; i < number; i++) {
        paragraph.add(new Paragraph(" "));
    }
}

public void creaPDF(FacturaDTO
factura,List<ProductoEnCotizacionDTO> productos,CotizacionDTO
cotizacion,UsuarioDTO usuario){
    try {
        Document document = new Document();
        PdfWriter.getInstance(document, new
FileOutputStream(FILE+factura.getFolio().toString()+".pdf"));
        document.open();
        addMetaData(document, "COFA", "Factura", "Factura
"+factura.getFolio());
        addTitlePage(document);

addContent(document, factura, productos, cotizacion, usuario);
        document.close();
    } catch (Exception e) {
        e.printStackTrace();
    }
}
}
}

```

UtileriasAdministrarSistema.java

```
package com.uam.cofa.utilities;

import java.util.List;
import java.util.Map;
import java.util.TreeMap;
import com.uam.cofa.data.componentesUsuario.*;
import com.uam.cofa.data.productosServicios.*;

/*Autor: Eduardo Gonzalez
 * Contiene metodos para colaborar con las tareas de adminiccion del
 sistema */

public class UtileriasAdministrarSistema {

    /**Recibe una lista con objetos de tipo OcupacionDTO y lo
 convierte en
 un objeto de tipo Map (utilidad para generar <s:select>*/
 public Map<Integer,String> cargarCatalogoDeOcupaciones(List
 listaOcupaciones) {
 Map <Integer,String> ocupaciones=new TreeMap();
 for (int i=0;i<listaOcupaciones.size();i++) {
 OcupacionDTO aux=(OcupacionDTO) listaOcupaciones.get(i);
 ocupaciones.put(aux.getId(),aux.getOcupacion());
 }
 return ocupaciones;
 }

 /**Recibe una lista con objetos de tipo MarcaDTO y lo convierte
 en
 un objeto de tipo Map (utilidad para generar <s:select>*/
 public Map<Integer,String> cargarCatalogoDeMarcas(List
 listaMarcas) {
 Map <Integer,String> marcas=new TreeMap();
 for (int i=0;i<listaMarcas.size();i++) {
 MarcaDTO aux=(MarcaDTO) listaMarcas.get(i);
 marcas.put(aux.getId(),aux.getMarca());
 }
 return marcas;
 }

 /**Recibe una lista con objetos de tipo SubtipoProductoDTO y lo
 convierte en
 un objeto de tipo Map (utilidad para generar <s:select>*/
 public Map<Integer,String> cargarCatalogoDeSubtipos(List
 listaSubtipos) {
 Map <Integer,String> subtipos=new TreeMap();
 for (int i=0;i<listaSubtipos.size();i++) {
 SubtipoProductoDTO
 aux=(SubtipoProductoDTO) listaSubtipos.get(i);
 subtipos.put(aux.getId(),aux.getSubtipo());
 }
 return subtipos;
 }
}
```

```
}  
}
```

Archivos de configuración xml

cotizaciones.xml

```
<?xml version="1.0" encoding="UTF-8"?>  
  
<!DOCTYPE struts PUBLIC  
    "-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"  
    "http://struts.apache.org/dtds/struts-2.0.dtd">  
  
<struts>  
<constant name="struts.devMode" value="true"/>  
<constant name="struts.objectFactory"  
    value="org.apache.struts2.spring.StrutsSpringObjectFactory" />  
  
    <package name="cotizaciones" extends="struts-default">  
  
        <action name="cotizaciones_*"  
            class="com.uam.cofa.actions.CotizacionesAction"  
method="{1}">  
            <result name="forma_generarCotizaciones"/WEB-  
INF/pages/cotizaciones/forma_generarCotizaciones.jsp</result>  
            <result name="agregadoACarrito"  
type="redirectAction">cotizaciones_forma_generarCotizaciones</result>  
            <result name="removidoDeCarrito"  
type="redirectAction">cotizaciones_forma_generarCotizaciones</result>  
            <result name="forma_seleccionarCantidad"/WEB-  
INF/pages/cotizaciones/forma_seleccionarCantidad.jsp</result>  
            <result name="pantallaCotizacion"/WEB-  
INF/pages/cotizaciones/pantallaCotizacion.jsp</result>  
            <result name="listarMisCotizaciones"/WEB-  
INF/pages/cotizaciones/misCotizaciones.jsp</result>  
            <result name="cotizacionGuardada"  
type="redirectAction">cotizaciones_listarPorUsuario</result>  
            <result name="error"/WEB-  
INF/pages/error.jsp</result>  
  
        </action>  
  
        <action name="adminCotizaciones_*"  
            class="com.uam.cofa.actions.AdministrarCotizacionesAction"  
method="{1}">  
            <result name="listar"/WEB-  
INF/pages/cotizaciones/listarCotizaciones.jsp</result>  
            <result name="listarMisCotizaciones"/WEB-  
INF/pages/cotizaciones/misCotizaciones.jsp</result>  
            <result name="listarMisCotizacionesCerradas"/WEB-  
INF/pages/cotizaciones/misCotizaciones.jsp</result>
```

```

        <result name="cotizacionCerrada"
type="redirectAction">adminCotizaciones_listarPorUsuarioCerradas</res
ult>
        <result name="muestraInformacionCotizacion">/WEB-
INF/pages/cotizaciones/informacionCotizacion.jsp</result>
    </action>

</package>

</struts>

```

facturas.xml

```

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE struts PUBLIC
    "-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"
    "http://struts.apache.org/dtds/struts-2.0.dtd">

<struts>
<constant name="struts.devMode" value="true"/>
<constant name="struts.objectFactory"
    value="org.apache.struts2.spring.StrutsSpringObjectFactory" />

    <package name="administracionFacturas" extends="struts-default">

        <action name="administradorFacturas_*"
            class="com.uam.cofa.actions.AdministradorFacturasAction"
method="{1}">
            <result name="pantallaAdministracion">/WEB-
INF/pages/facturas/pantallaAdministracion.jsp</result>
            <result name="generarFacturas">/WEB-
INF/pages/facturas/generarFacturas.jsp</result>
            <result name="listar">/WEB-
INF/pages/facturas/listar.jsp</result>
            <result name="facturasGeneradas"
type="redirectAction">administradorFacturas_listar</result>
        </action>

        <action name="facturas_*"
class="com.uam.cofa.actions.FacturasAction" method="{1}">
            <result name="misFacturas">/WEB-
INF/pages/facturas/misFacturas.jsp</result>
            <result name="mostrarInformacionDeFactura">/WEB-
INF/pages/facturas/informacionDeFactura.jsp</result>
        </action>

    </package>

</struts>

```

productos.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE struts PUBLIC
    "-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"
    "http://struts.apache.org/dtds/struts-2.0.dtd">

<struts>
<constant name="struts.devMode" value="true"/>
<constant name="struts.objectFactory"
    value="org.apache.struts2.spring.StrutsSpringObjectFactory" />

    <package name="productos" extends="struts-default">

        <action name="productos_*"
            class="com.uam.cofa.actions.ProductosAction" method="{1}">
            <result name="forma_insertarProducto">/WEB-
INF/pages/productos/forma_insertarProducto.jsp</result>
            <result name="administrar">/WEB-
INF/pages/productos/administrarProductos.jsp</result>
            <result name="insertado"
type="redirectAction">productos_listar</result>
            <result name="listar">/WEB-
INF/pages/productos/listar.jsp</result>
            <result name="listarParaModificar">/WEB-
INF/pages/productos/listarParaModificar.jsp</result>
            <result name="forma_modificarProducto">/WEB-
INF/pages/productos/forma_modificarProducto.jsp</result>
            <result name="productoModificado"
type="redirectAction">productos_listar</result>
        </action>

    </package>
</struts>
```

usuarios.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE struts PUBLIC
    "-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"
    "http://struts.apache.org/dtds/struts-2.0.dtd">

<struts>
<constant name="struts.devMode" value="true"/>
<constant name="struts.objectFactory"
    value="org.apache.struts2.spring.StrutsSpringObjectFactory" />

    <package name="usuarios" extends="struts-default">

        <action name="usuarios_*"
            class="com.uam.cofa.actions.UsuariosAction" method="{1}">
```

```

        <result name="forma_insertarCliente">/WEB-
INF/pages/usuarios/forma_insertarCliente.jsp</result>
        <result name="input">/WEB-
INF/pages/usuarios/forma_insertarCliente.jsp</result>
        <result name="insertado" type="redirectAction"
>usuarios_listar</result>
        <result name="clienteInsertado">/WEB-
INF/pages/usuarios/forma_insertarModosDeContacto.jsp</result>
        <result type="redirectAction">usuarios_listar</result>
        <result name="listar">/WEB-
INF/pages/listarUsuarios.jsp</result>
        <result name="registroNuevoUsuarioExitoso">/WEB-
INF/pages/usuarios/registroExitoso.jsp</result>
    </action>

    <action name="direcciones_*"
class="com.uam.cofa.actions.DireccionesAction" method="{1}">
        <result name="listar">/WEB-
INF/pages/listarDirecciones.jsp</result>
        <!-- <result name="direccionInsertada"> /WEB-
INF/pages/usuarios/registroExitoso.jsp</result> -->
        <result name="direccionInsertada"
type="redirectAction">usuarios_insertarDatosCompletos</result>
        <result name="input">/WEB-
INF/pages/usuarios/forma_insertarDireccion.jsp</result>
    </action>

    <action name="modosDeContacto_*"
class="com.uam.cofa.actions.ModosDeContactoAction"
method="{1}">
        <result name="modosDeContactoInsertados">/WEB-
INF/pages/usuarios/forma_insertarDireccion.jsp</result>
        <result name="forma_insertarModosDeContacto">/WEB-
INF/pages/usuarios/forma_insertarModosDeContacto.jsp</result>
        <result name="error">/WEB-
INF/pages/usuarios/forma_insertarModosDeContacto.jsp</result>
        <result name="input">/WEB-
INF/pages/usuarios/forma_insertarModosDeContacto.jsp</result>
    </action>

</package>
</struts>

```

validacion.xml

```

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE struts PUBLIC
    "-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"
    "http://struts.apache.org/dtds/struts-2.0.dtd">

<struts>
<constant name="struts.devMode" value="true"/>
<constant name="struts.objectFactory"
    value="org.apache.struts2.spring.StrutsSpringObjectFactory" />

```

```

<package name="validacion" extends="struts-default">

    <action name="validar_*"
        class="com.uam.cofa.actions.ValidarUsuarioAction" method="{1}">
        <result name="listar">/WEB-
INF/pages/listarDatosDeAcceso.jsp</result>
        <result name="forma_validarUsuario">/WEB-
INF/pages/forma_validarUsuario.jsp</result>
        <result name="validado">/WEB-
INF/index_registrado.jsp</result>
        <result name="error">/WEB-
INF/pages/forma_validarUsuario.jsp</result>
    </action>

</package>
</struts>

```

struts.xml

```

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE struts PUBLIC
    "-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"
    "http://struts.apache.org/dtds/struts-2.0.dtd">

<struts>
<constant name="struts.devMode" value="true"/>
<constant name="struts.objectFactory"
    value="org.apache.struts2.spring.StrutsSpringObjectFactory" />

<include file="curso.xml"/>
<include file="usuarios.xml"/>
<include file="validacion.xml"/>
<include file="cotizaciones.xml"/>
<include file="productos.xml"/>
<include file="facturas.xml"/>

<package name="/" extends="struts-default">

    <action name="index_*"
        class="com.uam.cofa.actions.InicioAction" method="{1}">

        <result name="registrado">/WEB-INF/index.jsp</result>
        <result name="no_registrado">/WEB-INF/index.jsp</result>
        <result name="salir">/WEB-INF/index.jsp</result>
        <result name="success">/WEB-INF/index.jsp</result>
        <result name="indexRegistrado">/WEB-
INF/index_registrado.jsp</result>
    </action>

    <action name="download"
class="com.uam.cofa.actions.DownloadAction">

```

```

        <result name="success" type="stream">
            <param name="contentType">application/octet-stream</param>
            <param name="inputName">fileInputStream</param>
            <param
name="contentDisposition">attachment;filename="Factura.pdf"</param>
            <param name="bufferSize">1024</param>
        </result>
    </action>

</package>
</struts>

```

spring.xml

```

<?xml version="1.0" encoding="UTF-8"?>

<beans
xmlns="http://www.springframework.org/schema/beans"
xmlns:tx="http://www.springframework.org/schema/tx"
xmlns:context="http://www.springframework.org/schema/context"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-2.5.xsd
http://www.springframework.org/schema/tx
http://www.springframework.org/schema/tx/spring-tx-2.5.xsd
http://www.springframework.org/schema/context
http://www.springframework.org/schema/context/spring-context-2.5.xsd"
default-autowire="byName" >

    <bean id="dataSource"
class="org.apache.commons.dbcp.BasicDataSource" destroy-
method="close">
        <property name="driverClassName">
            <!-- <value>org.postgresql.Driver</value> -->
            <value>com.mysql.jdbc.Driver</value>
        </property>
        <property name="url">
            <value>jdbc:mysql://localhost:3306/mydb?
autoReconnect=true</value>
            <!-- <value>jdbc:postgresql://localhost:5432/curso?
autoReconnect=true</value> -->
        </property>
        <property name="maxWait" value="10000"/>
        <property name="username" value="root"/>
        <property name="password" value="root"/>
    </bean>

    <bean id="transactionManager" class=
"org.springframework.jdbc.datasource.DataSourceTransactionManage
r"/>

    <tx:annotation-driven/>

    <bean id="sessionFactory"
class="org.springframework.orm.hibernate3.annotation.AnnotationSessio
nFactoryBean">
        <property name="annotatedClasses">

```

```

        <list>
            <value>com.uam.cofa.data.componentesUsuario.DatosDeAccesoDTO<
/val
            <value>com.uam.cofa.data.CursoDTO</value>
            <value>com.uam.cofa.data.componentesUsuario.UsuarioDTO</value
>
            <value>com.uam.cofa.data.componentesUsuario.MedioDeContactoDT
O</value>

<value>com.uam.cofa.data.componentesUsuario.DireccionDTO</value>

<value>com.uam.cofa.data.componentesUsuario.OcupacionDTO</value>

    <value>com.uam.cofa.data.productosServicios.ProductoDTO</value>

    <value>com.uam.cofa.data.productosServicios.MarcaDTO</value>

<value>com.uam.cofa.data.productosServicios.SubtipoProductoDTO</value
>

<value>com.uam.cofa.data.productosServicios.TipoProductoDTO</value>

<value>com.uam.cofa.data.facturasCotizaciones.CotizacionDTO</value>

<value>com.uam.cofa.data.facturasCotizaciones.ProductoEnCotizacionDTO
</value>

<value>com.uam.cofa.data.facturasCotizaciones.FacturaDTO</value>
    </list>
</property>
<property name="hibernateProperties">
    <props>
        <prop key="hibernate.dialect">
            org.hibernate.dialect.MySQLDialect
        </prop>
    </props>
</property>
</bean>

<bean id="validadorUsuarios"
    class="com.uam.cofa.business.local.ValidadorUsuariosImpl"
/>

<bean id="administradorCursos"
    class="com.uam.cofa.business.local.AdministradorCursosImpl"
/>

<bean id="administradorUsuarios"
    class="com.uam.cofa.business.local.AdministradorUsuariosImpl"
/>

<bean id="administradorDirecciones"
    class="com.uam.cofa.business.local.AdministradorDireccionesIm
pl"

```

```

    />

    <bean id="administradorProductos"
        class="com.uam.cofa.business.local.AdministradorProductosImpl"
    />

    <bean id="administradorCotizaciones"
        class="com.uam.cofa.business.local.AdministradorCotizacionesI
mpl"
    />

    <bean id="administradorFacturas"
        class="com.uam.cofa.business.local.AdministradorFacturasImpl"
    />

</beans>

```

DireccionesAction-validation.xml

```

<!DOCTYPE validators PUBLIC "-//OpenSymphony Group//XWork Validator
1.0.2//EN"
    "http://www.opensymphony.com/xwork/xwork-validator-1.0.2.dtd">

<validators>
    <field name="direccion.calle">
        <field-validator type="requiredstring">
            <message>La calle es requerida.</message>
        </field-validator>
    </field>

    <field name="direccion.numeroExt">
        <field-validator type="requiredstring">
            <message>El numero exterior es requerido.</message>
        </field-validator>
    </field>

    <field name="direccion.colonia">
        <field-validator type="requiredstring">
            <message>El nombre de la colonia es requerido.</message>
        </field-validator>
    </field>

    <field name="direccion.codigoPostal">
        <field-validator type="requiredstring">
            <message>El código postal es requerido.</message>
        </field-validator>
    </field>

    <field name="direccion.ciudad">
        <field-validator type="requiredstring">
            <message>El nombre de la ciudad es requerido.</message>
        </field-validator>
    </field>

</validators>

```

web.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
    http://java.sun.com/xml/ns/javaee/web-
app_2_5.xsd"
  id="WebApp_ID" version="2.5"
>
  <display-name>Aplicación</display-name>

  <!--***** LISTENERS *****-->

  <listener>
  <listener-class>
    org.springframework.web.context.ContextLoaderListener
  </listener-class>
  </listener>

  <!--***** FILTERS *****-->
  <

  <filter>
    <filter-name>sitemesh</filter-name>
    <filter-
class>com.opensymphony.module.sitemesh.filter.PageFilter</filter-
class>
    </filter>

  <filter-mapping>
    <filter-name>sitemesh</filter-name>
    <url-pattern>*.action</url-pattern>
  </filter-mapping>

  <filter>
    <filter-name>struts2</filter-name>
    <filter-
class>org.apache.struts2.dispatcher.FilterDispatcher</filter-class>
    </filter>

  <filter-mapping>
    <filter-name>struts2</filter-name>
    <url-pattern>/*</url-pattern>
  </filter-mapping>
```

```

<!--***** ADDITIONAL CONFIGURATION
*****-->

<welcome-file-list>
  <welcome-file>index.jsp</welcome-file>
</welcome-file-list>

<context-param>
  <param-name>contextConfigLocation</param-name>
  <param-value>classpath:spring.xml</param-value>
</context-param>

</web-app>

```

decorators.xml

```

<?xml version="1.0" encoding="ISO-8859-1"?>

<decorators defaultdir="/WEB-INF/layouts">
  <excludes>
    <pattern>/css/*</pattern>
    <pattern>/img/*</pattern>
    <pattern>/scripts/*</pattern>
    <pattern>/struts/dojo/*</pattern>
    <pattern>/resources/*</pattern>
    <pattern>/index.jsp</pattern>
    <pattern>/jlogin.jsp</pattern>
    <pattern>/construction.jsp</pattern>
    <pattern>/confirmation*</pattern>
    <pattern>/informacionDeFactura.jsp</pattern>
  </excludes>

  <decorator name="internal_layout" page="internal_layout.jsp">
    <pattern>/*</pattern>
  </decorator>
</decorators>

```

WEB content

CSS

Style_menuDesplegable.css

```

ul#css3menu1, ul#css3menu1 ul{
  margin:0;list-style:none;padding:0;background-
color:#dedede;border-width:1px;border-style:solid;border-
color:#5f5f5f;-moz-border-radius:5px;-webkit-border-
radius:5px;border-radius:5px;}

```



```

ul#css3menu1 ul{
    display:none;position:absolute;left:0;top:100%;-moz-box-
shadow:3.5px 3.5px 5px #000000;-webkit-box-shadow:3.5px 3.5px 5px
#000000;box-shadow:3.5px 3.5px 5px #000000;padding:0 10px
10px;background-color:#FFFFFF;border-radius:6px;-moz-border-
radius:6px;-webkit-border-radius:6px;border-color:#d4d4d4;}
ul#css3menu1 li:hover>{*
    display:block;}
ul#css3menu1 li:hover{
    position:relative;}
ul#css3menu1 ul ul{
    position:absolute;left:100%;top:0;}
ul#css3menu1{
    display:block;font-size:0;float:left;}
ul#css3menu1 li{
    display:block;white-space:nowrap;font-size:0;float:left;}
ul#css3menu1>li,ul#css3menu1 li{
    margin:0;}
ul#css3menu1 a:active, ul#css3menu1 a:focus{
    outline-style:none;}
ul#css3menu1 a{
    display:block;vertical-align:middle;text-align:left;text-
decoration:none;font:bold 14px Trebuchet MS;color:#000000;text-
shadow:#FFF 0 0 1px;cursor:pointer;padding:10px;background-
color:#c1c1c1;background-image:url("../img/mainbk.png");background-
repeat:repeat;background-position:0 0;border-width:0 0 0 1px;border-
style:solid;border-color:#C0C0C0;}
ul#css3menu1 ul li{
    float:none;margin:10px 0 0;}
ul#css3menu1 ul a{
    text-align:left;padding:4px;background-
color:#FFFFFF;background-image:none;border-width:0;border-
radius:0px;-moz-border-radius:0px;-webkit-border-radius:0px;font:14px
Tahoma;color:#000;text-decoration:none;}
ul#css3menu1 li:hover>a{
    background-color:#0c97e2;border-color:#C0C0C0;border-
style:solid;font:bold 14px Trebuchet MS;color:#FFFFFF;text-
decoration:none;text-shadow:#FFF 0 0 1px;background-
image:url("../img/mainbk.png");background-position:0 100px;}
ul#css3menu1 img{
    border:none;vertical-align:middle;margin-right:10px;}
ul#css3menu1 img.over{
    display:none;}
ul#css3menu1 li:hover > a img.def{
    display:none;}
ul#css3menu1 li:hover > a img.over{
    display:inline;}
ul#css3menu1 span{
    display:block;overflow:visible;background-position:right
center;background-repeat:no-repeat;padding-right:0px;}
ul#css3menu1 ul li:hover>a{
    background-color:#FFFFFF;background-image:none;font:14px
Tahoma;color:#0978b3;text-decoration:none;}
ul#css3menu1 li.topfirst>a{
    height:20px;line-height:20px;border-radius:5px 0 0 5px;-moz-
border-radius:5px 0 0 5px;-webkit-border-radius:5px;-webkit-border-
top-right-radius:0;-webkit-border-bottom-right-radius:0;}

```

```

ul#css3menu1 li.topfirst: hover>a{
    line-height: 20px;}
ul#css3menu1 li.topmenu>a{
    height: 20px; line-height: 20px;}
ul#css3menu1 li.topmenu: hover>a{
    line-height: 20px;}
ul#css3menu1 li.toplast>a{
    height: 20px; line-height: 20px; border-radius: 0 5px 5px 0; -moz-
border-radius: 0 5px 5px 0; -webkit-border-radius: 0; -webkit-border-top-
right-radius: 5px; -webkit-border-bottom-right-radius: 5px;}
ul#css3menu1 li.toplast: hover>a{
    line-height: 20px;}

```

style.css

```


/*
Design by Eduardo Gonzalez
*/

body {
    margin: 0;
    padding: 0;
    background: #FFFFFF;
    font-family: Arial, Helvetica, sans-serif;
    font-size: 12px;
    color: #A19D9A;
}

h1, h2, h3 {
    margin: 0;
    padding: 0;
    font-weight: normal;
    color: #3E3530;
}

h1 {
    font-size: 2em;
}

h2 {
    font-size: 2.8em;
}

h3 {
    font-size: 1.6em;
}

p, ul, ol {
    margin-top: 0;
    line-height: 180%;
}

ul, ol {


```

```

}

a {
    text-decoration: none;
    color: #339CCB;
}

a:hover {
}

#wrapper {
    margin: 0 auto;
    padding: 0;
}

/* Header */

#header-wrapper {
    height: 100px;
    background: #1582AB;
    border-bottom: 10px solid #2AA1CC;
}

#header {
    width: 1220px;
    margin: 0 auto;
    padding: 0px 0px 0px 30px;
}

/* Logo */

#logo {
    float: left;
    width: 350px;
    margin: 0;
    padding: 0;
    color: #000000;
    background: url(images/img02.jpg) no-repeat left 15px;
}

#logo h1, #logo p {
}

#logo h1 {
    margin-left: 60px;
    padding: 40px 0px 0px 80px;
    letter-spacing: -1px;
    text-transform: lowercase;
    font-size: 3.8em;
    background: red;
}

#logo p {
    margin: 0;
    padding: 0px 0 0 62px;
    font: normal 14px Georgia, "Times New Roman", Times, serif;
    font-style: italic;
}

```

```

        color: #FFFFFF;
    }

#logo a {
    border: none;
    background: none;
    text-decoration: none;
    color: #FFFFFF;
}

/* Search */

#search {
    float: right;
    width: 280px;
    height: 60px;
    padding: 20px 0px 0px 0px;
}

#search form {
    height: 41px;
    margin: 0;
    padding: 10px 0 0 20px;
}

#search fieldset {
    margin: 0;
    padding: 0;
    border: none;
}

#search-text {
    width: 170px;
    padding: 6px 5px 2px 5px;
    border: 1px solid #DEDEDE;
    background: #FFFFFF;
    text-transform: lowercase;
    font: normal 11px Arial, Helvetica, sans-serif;
    color: #5D781D;
}

#search-submit {
    width: 50px;
    height: 22px;
    border: none;
    background: #B9B9B9;
    color: #000000;
}

/*para el formulario de registro de nuevo usuario (dar de alta nuevas cosas)*/

#register {
    float: left;
    width: 280px;
    height: 60px;
    padding: 20px 0px 0px 0px;
}

```

```

#register form {
    height: 41px;
    margin: 0;
    padding: 10px 0 0 20px;
}

#register fieldset {
    margin: 0;
    padding: 0;
    border: none;
}

#register-text {
    width: 170px;
    padding: 6px 5px 2px 5px;
    border: 1px solid #DEDEDE;
    background: #FFFFFF;
    font: normal 11px Arial, Helvetica, sans-serif;
    color: #5D781D;
}

#register-select {
    width: 182px;
    /*padding: 6px 5px 2px 5px;*/
    border: 1px solid #DEDEDE;
    background: #FFFFFF;
    text-transform: uppercase;
    font: normal 11px Arial, Helvetica, sans-serif;
    color: #5D781D;
}

#register-data {
    width: 170px;
    padding: 6px 5px 2px 5px;
    border: 1px solid #DEDEDE;
    background: #FFFFFF;
    font: strong 11px Arial, Helvetica, sans-serif;
    color: #5D781D;
}

#register-submit {
    width: 70px;
    height: 22px;
    border: none;
    background: #B9B9B9;
    color: #000000;
}

/* Menu */

#menu {

```

```

        float: right;
        width: 600px;
        height: 90px;
        margin: 0 auto;
        padding: 0;
    }

    #menu ul {
        margin: 0;
        padding: 60px 0px 0px 0px;
        list-style: none;
        line-height: normal;
    }

    #menu li {
        float: right;
    }

    #menu a {
        display: block;
        margin-right: 1px;
        padding: 10px 15px 15px 15px;
        text-decoration: none;
        text-align: center;
        font-family: Arial, Helvetica, sans-serif;
        font-size: 13px;
        font-weight: bold;
        color: #FFFFFF;
        border: none;
    }

    #menu a:hover, #menu .current_page_item a {
        background: #2AA1CC;
        text-decoration: none;
    }

    #menu .current_page_item a {
    }

    /* Page */

    #page {
        width: 980px;
        margin: 0 auto;
        padding: 0px 0px 0px 0px;
    }

    #page-bgtop {
        padding: 20px 0px;
    }

    /* Content */

    #content {
        float: left;
        width: 650px;
        padding: 0px 0px 0px 0px;
    }

```

```

.post {
    margin-bottom: 40px;
    border-bottom: 1px dotted #E7E2DC;
}

.post .title {
    height: 41px;
    padding: 7px 0px 0px 30px;
    letter-spacing: -.5px;
}

.post .title a {
    color: #333333;
    border: none;
}

.post .meta {
    margin-bottom: 30px;
    padding: 0px 30px 0px 30px;
    text-align: left;
    font-family: Arial, Helvetica, sans-serif;
    font-size: 13px;
    font-weight: bold;
    font-style: italic;
}

.post .meta .date {
    float: left;
}

.post .meta .posted {
    float: right;
}

.post .meta a {
}

.post .entry {
    padding: 0px 30px 20px 30px;
    padding-bottom: 20px;
    text-align: justify;
}

.links {
    padding-top: 20px;
    font-size: 12px;
    font-weight: bold;
}

/* Sidebar */

#sidebar {
    float: right;
    width: 280px;
    margin: 0px;
    padding: 0px 20px 0px 0px;
}

```

```

        color: #787878;
    }

    #sidebar ul {
        margin: 0;
        padding: 0;
        list-style: none;
    }

    #sidebar li {
        margin: 0;
        padding: 0;
    }

    #sidebar li ul {
        margin: 0px 0px;
        padding-bottom: 30px;
    }

    #sidebar li li {
        line-height: 35px;
        border-bottom: 1px dotted #E7E2DC;
        margin: 0px 30px;
        border-left: none;
    }

    #sidebar li li span {
        display: block;
        margin-top: -20px;
        padding: 0;
        font-size: 11px;
        font-style: italic;
    }

    #sidebar li li a {
        padding: 0px 0px 0px 15px;
        background: url(images/img01.jpg) no-repeat left 5px;
    }

    #sidebar h2 {
        height: 38px;
        padding-left: 30px;
        letter-spacing: -.5px;
        font-size: 1.8em;
    }

    #sidebar p {
        margin: 0 0px;
        padding: 0px 30px 20px 30px;
        text-align: justify;
    }

    #sidebar a {
        border: none;
        color: #339CCB;
    }

```



```

#sidebar a:hover {
    text-decoration: underline;
    color: #8A8A8A;
}

/* Calendar */

#calendar {
}

#calendar_wrap {
    padding: 20px;
}

#calendar table {
    width: 100%;
}

#calendar tbody td {
    text-align: center;
}

#calendar #next {
    text-align: right;
}

/* Footer */

#footer {
    height: 50px;
    /*width: 980px;*/
    margin: 0 auto;
    padding: 0px 0 15px 0;
    background: #1582AB;
    border-top: 1px solid #1885B0;
    font-family: Arial, Helvetica, sans-serif;
}

#footer p {
    margin: 0;
    padding-top: 20px;
    line-height: normal;
    font-size: 9px;
    text-transform: uppercase;
    text-align: center;
    color: #FFFFFF;
}

#footer a {
    color: #FFFFFF;
}

/*Parte para la definicion de tablas*/

table.visual, th.visual, td.visual {

```

```

border: 1px solid #D4E0EE;
border-collapse: collapse;
font-family: "Trebuchet MS", Arial, sans-serif;
color: #555;
}

caption.visual {
font-size: 150%;
font-weight: bold;
margin: 5px;
}

td.visual, th.visual {
padding: 4px;
}

thead.visual th.visual {
text-align: center;
background: #E6EDF5;
color: #4F76A3;
font-size: 100% !important;
}

tbody.visual th.visual {
font-weight: bold;
}

tbody.visual tr.visual { background: #FCFDFE; }

tbody.visual tr.visual.odd { background: #F7F9FC; }

table.visual a:link {
color: #718ABE;
text-decoration: none;
}

table.visual a:visited {
color: #718ABE;
text-decoration: none;
}

table.visual a:hover {
color: #718ABE;
text-decoration: underline !important;
}

tfoot.visual th.visual, tfoot.visual td.visual {
font-size: 85%;
}

div.contenedor{
position: absolute;
width: 500px;
height: 270px;
border: solid 2px #F5F7F7;
background: #FFFFFF;
}

```

```

z-index: 500;
-moz-box-shadow:3.5px 3.5px 5px #000000;
-webkit-box-shadow:3.5px 3.5px 5px #000000;
box-shadow:3.5px 3.5px 5px #000000;
border-radius:6px;
-moz-border-radius:6px;
-webkit-border-radius:6px;
font:bold 12px Trebuchet MS;
color:#000000;text-shadow:#FFF 0 0 1px;

overflow-x: auto;
overflow-y: auto;

/*filter:alpha(opacity=20);*/
/* CSS3 standard */
/*opacity:0.6;*/

}

```

Layouts

header_inicio.jspf

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>

<div id="header-wrapper">
    <div id="header">
        <div id="logo">
            <h1><a href="#">COFA </a></h1>
            <p>sistema de cotizaciones y facturas </p>
        </div>

        <div id="menu">
            <ul>
                <li><a class=""
href="usuarios_forma_insertarCliente.action">Registrarse</a></li>
                <li><a class=""
href="productos_listar.action">Productos</a></li>
                <li><a class=""
href="#">Servicios</a></li>
                <li class="current_page_item"><a
href="index.action">Home</a></li>
            </ul>
        </div>
    </div>
</div>

```

header_administrador.jspf

```
<div id="header-wrapper">
  <div id="header">
    <div id="logo">
      <h1><a href="#">COFA </a></h1>
      <p>sistema de cotizaciones y facturas </p>
    </div>

    <div id="menu">

      <ul>
        <li><a
href="index_salir.action">Salir</a></li>
        <li><a
href="productos_administrar.action">Productos</a></li>
        <li><a href="#">Servicios</a></li>
        <li><a
href="adminCotizaciones_listar.action">Cotizaciones</a></li>
        <li><a
href="administradorFacturas_pantallaAdministracion.action">Facturas</
a></li>
        <li class="current_page_item"><a
href="index_indexRegistrado.action">Home</a></li>
      </ul>
    </div>
  </div>
</div>
```

header_registrado.jspf

```
<div id="header-wrapper">
  <div id="header">
    <div id="logo">
      <h1><a href="#">COFA </a></h1>
      <p>sistema de cotizaciones y facturas </p>
    </div>

    <div id="menu">

      <ul>
        <li><a
href="index_salir.action">Salir</a></li>
        <li><a
href="productos_listar.action">Productos</a></li>
        <li><a href="#">Servicios</a></li>
        <li><a
href="cotizaciones_forma_generarCotizaciones.action">Cotizaciones</a>
</li>
        <li><a
href="facturas_misFacturas.action">Mis Facturas</a></li>
        <li class="current_page_item"><a
href="index_indexRegistrado.action">Home</a></li>
      </ul>
    </div>
  </div>
```

```

        </ul>
    </div>
</div>
</div>

```

footer.jspf

```

<br><br><br><br><br><br><br><br><br><br><br><br><br><br>
<div id="footer">
    <p>Copyright (c) 2011 COFA. Eduardo González Herrera.</p>
</div>

```

internal_layout.jsp

```

<%@ taglib uri="http://www.opensymphony.com/sitemesh/decorator"
prefix="decorator" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jstl/fmt" prefix="fmt" %>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ page import="com.uam.cofa.data.componentesUsuario.UsuarioDTO;" %>

<decorator:usePage id="html_page"/>

<html>
    <head>
        <title><decorator:title default="COFA"/></title>
        <link rel="shortcut icon" href="$
{ pageContext.request.contextPath }/img/favicon.ico">
        <link rel="stylesheet" href="$
{ pageContext.request.contextPath }/css/style.css">
        <decorator:head/>
    </head>

    <body>

        <%
        HttpSession sesionOk = request.getSession();
        %>

        <table id="pagina">

            <tr>
                <td colspan="2">

                    <div align="center" class="breadcrumb">

                        <%
                        //HttpSession sesionOk =
request.getSession();
                        if
(sesionOk.getAttribute("UsuarioLogado") == null) {
                            %>

                            <%@include file="/WEB-
INF/layouts/header_inicio.jspf" %>

```

```

                <%> else{
                    UsuarioDTO
usuario=(UsuarioDTO) sesionOk.getAttribute("usuario");

                if(usuario.getIdTipoDeUsuario().equals(new Integer(1))){
                    <%>
                    <%@include file="/WEB-
INF/layouts/header_administrador.jspf" %>
                    <%>
                    else
                if(usuario.getIdTipoDeUsuario().equals(new Integer(2))){
                    <%>
                    <%@include file="/WEB-
INF/layouts/header_registrado.jspf" %>
                    <%> %>

                <%> %>

                </div>
            </td>
        </tr>

        <tr>
            <td width="700" id="contenido">
                <div align="center">
                    <br>
                    <decorator:body/> <!-- The content page -->
                    <br>
                    <hr width="80%" size="3">
                </div>
            </td>
        </tr>

        <tr>
            <td>
                <div align="center" class="breadcrumb">
                    <%@include file="/WEB-
INF/layouts/footer.jspf"%>
                </div>
            </td>
        </tr>
    </table>

</body>

<head>
    <META HTTP-EQUIV="PRAGMA" CONTENT="NO-CACHE">
</head>
</html>

```

Páginas

Cotizaciones

forma_generarCotizaciones.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<link rel="stylesheet" href="css/style_menuDesplegable.css"
type="text/css" />
<title>COFA</title>
<s:head theme="ajax"/>

</head>
<body>

    Generación de cotizaciones<br><br>
    <a href="adminCotizaciones_listarPorUsuario.action">
        </img>
    </a>
    <br><br>
    <!-- aca se crea el menu de tipos y subtipos de productos -->
    <ul id="css3menu1" class="topmenu">
    <c:forEach var="tipo" items="{ listaTiposProductos }">
        <li class="topmenu"><a href="" title="{tipo.tipo}">$
{tipo.tipo}</a>
        <ul>
            <c:forEach var="subtipo" items="$
{ listaSubtiposProductos }">
                <c:if test="{tipo.id ==
subtipo.idTipoDeProducto }">
                    <li>
                        <a
href="cotizaciones_listarPorTipo.action?id='subtipo ${subtipo.id}'"
title="{subtipo.subtipo}">$
{subtipo.subtipo}</a>
                    </li>
                </c:if>
            </c:forEach>
        </ul>
    </li>
    </c:forEach>
</ul>
```

```

</ul>
<!-- Termina de generarse en menu de los productos -->
<br>.<br>.<br>.<br>.<br>

<table class="visual" align="center" width="80%" border="1">
<thead class="visual"><tr class="visual">
  <th colspan="10" class="visual" align="center">
    <div class="label">Lista productos</div>
  </th>
</tr></thead>
<thead class="visual"><tr class="visual">
  <th class="visual" align="center"><div
class="label">Imagen</div></th>
  <th class="visual" align="center"><div
class="label">Nombre</div></th>
  <th class="visual" align="center"><div
class="label">Marca</div></th>
  <th class="visual" align="center"><div
class="label">Precio</div></th>
  <th class="visual" align="center"><div
class="label">clasificación</div></th>
  <th class="visual" align="center"><div
class="label">Agregar</div></th>
</tr></thead>

  <c:forEach varStatus="status" var="producto" items="$
{ productos }">
    <tr class="odd">
      <td>${ producto.imagen }</td>
      <td>${ producto.nombre }</td>
      <td>${ producto.idMarca }</td>
      <td>${ producto.precioUnitario }</td>
      <td>${ producto.idSubtipo }</td>
      <td align="center">
        <a
href="cotizaciones_agregarProductoACarrito.action?idProducto=$
{ producto.id}">
          </img>
        </a>
      </td>
    </tr>
  </c:forEach>
</table>

<br><br><br><br><br>

<table class="visual" align="center" width="80%" border="1">
<thead class="visual"><tr class="visual">
  <th colspan="10" class="visual" align="center">
    <div class="label">Lista de productos a cotizar </div>
  </th>
</tr></thead>
<thead class="visual"><tr class="visual">
  <th class="visual" align="center"><div
class="label">Nombre</div></th>
  <th class="visual" align="center"><div
class="label">Precio</div></th>

```



```

        <th class="visual" align="center"><div
class="label">Remover</div></th>
    </tr></thead>

    <c:forEach varStatus="status" var="productoEnCarrito"
items="{ listaProductosCotizacion }">
        <tr class="odd">
            <td>${ productoEnCarrito.nombre }</td>
            <td>${ productoEnCarrito.precioUnitario }</td>
            <td align="center">
                <a
href="cotizaciones_removeProductoDeCarrito.action?idProducto=${
productoEnCarrito.idProducto}">
                    </img>
                </a>
            </td>
        </tr>
    </c:forEach>
</table>

    <br><br>
    <a href="cotizaciones_forma_seleccionarCantidad.action">
        </img>
    </a>

</body>
</html>

```

forma_seleccionarCantidad.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>
</head>
<body>

<br>

<table class="visual" align="center" width="80%" border="1">
    <thead class="visual"><tr class="visual">
        <th colspan="10" class="visual" align="center">
            <div class="label">Selecciona cuantos productos
deseas</div>

```

```

        </th>
    </tr></thead>
    <tr>
        <th class="visual" align="center"><div
class="label">Nombre</div></th>
        <th class="visual" align="center"><div
class="label">Precio</div></th>
        <th class="visual" align="center"><div
class="label">cantidad</div></th>
    </tr>
    <s:form action="cotizaciones_obtenerCantidadDeProductos">
        <c:forEach varStatus="status" var="productoEnCarrito"
items="{ listaProductosCotizacion }">
            <tr class="visual">
                <td class="visual">${ productoEnCarrito.nombre }</td>
                <td class="visual">${ $
{ productoEnCarrito.precioUnitario }</td>
                <td class="visual" align="right">
                    <s:textfield name="cantidadesDeProductos"
id="register-text" value="1" />
                </td>
            </tr>
        </c:forEach>
        <s:submit value="enviar"></s:submit>
    </s:form>
</table>

<br><br><br><br><br>

</body>
</html>

```

informacionCotizacion.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib prefix="sx" uri="/struts-dojo-tags"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>
</head>
<body>

    <table align="left">
        <thead class="visual"><tr>
            <th colspan="10" class="visual" align="left">
                <div class="label">Insumos</div>

```

```

        </th>
    </tr></thead>

    <tr class="visual">
        <th class="visual">Producto</th>
        <th class="visual">Precio <br>Unitario</th>
        <th class="visual">Cantidad</th>
        <th class="visual">Iva</th>
        <th class="visual">Importe</th>
        <th class="visual">Total</th>
    </tr>

    <c:forEach varStatus="status" var="producto" items="$
{productosEnCotizacion}">
        <tr class="visual">
            <td class="visual">${
{producto.idProducto}}</td>
            <td class="visual">${
{producto.precioUnitario}}</td>
            <td class="visual">${{producto.cantidad}}</td>
            <td class="visual">${
{producto.iva}}</td>
            <td class="visual">${
{producto.importe}}</td>
            <td class="visual">${
{producto.total}}</td>
        </tr>
    </c:forEach>
    <tr>
        <td class="visual"></td>
        <td class="visual"></td>
        <td class="visual"><strong>Total</strong></td>
        <td class="visual"><strong>${
{cotizacion.iva}}</strong></td>
        <td class="visual"><strong>${
{cotizacion.importeParcial}}</strong></td>
        <td class="visual"><strong>${
{cotizacion.importeTotal}}</strong></td>
    </tr>

</table>

</body>
</html>

```

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>

```

```

<script type="text/javascript">

```

```

function muestra_oculta(id,idCotizacion) {

```

```

        if (document.getElementById){ //se obtiene el id
        var el = document.getElementById(id); //se define la variable
"el" igual a nuestro div
        el.style.display = (el.style.display == 'none') ? 'block' :
'none'; //damos un atributo display:none que oculta el div
        if(el.style.display== 'block'){

MostrarConsulta('adminCotizaciones_informacionCotizacion.action?
idCotizacion='+idCotizacion,id);
        }
        }
}

/*
    window.onload = function(){//hace que se cargue la función lo
que predetermina que div estará oculto hasta llamar a la función
nuevamente

    muestra_oculta('contenido_a_mostrar');//"contenido_a_mostrar"
es el nombre que le dimos al DIV
    }*/

function objetoAjax(){
    var xmlhttp=false;
    try {
        xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
    } catch (e) {
        try {
            xmlhttp = new
ActiveXObject("Microsoft.XMLHTTP");
        } catch (E) {
            xmlhttp = false;
        }
    }

    if (!xmlhttp && typeof XMLHttpRequest!='undefined') {
        xmlhttp = new XMLHttpRequest();
    }
    return xmlhttp;
}

function MostrarConsulta(datos,id){
    divResultado =document.getElementById(id);
    ajax=objetoAjax();
    ajax.open("GET", datos);
    ajax.onreadystatechange=function() {
        if (ajax.readyState==4) {
            divResultado.innerHTML = ajax.responseText;
        }
    }
    ajax.send(null);
}
</script>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>
</head>

```

```

<body>

    <table class="visual" align="center" width="80%" border="1">
        <thead class="visual"><tr class="visual">
            <th colspan="10" class="visual" align="center">
                <div class="label">Cotizaciones</div>
            </th>
        </tr></thead>
        <thead class="visual"><tr>
            <th class="visual" align="center"><div
class="visual">Id</div></th>
            <th class="visual" align="center"><div class="visual">Id
Usuario</div></th>
            <th class="visual" align="center"><div class="visual">Fecha
de cotización</div></th>
            <th class="visual" align="center"><div class="visual">Importe
parcial</div></th>
            <th class="visual" align="center"><div
class="visual">IVA</div></th>
            <th class="visual" align="center"><div class="visual">Importe
total</div></th>
        </tr></thead>

        <c:forEach varStatus="status" var="cotizacion" items="$
{ cotizaciones }">
            <tr class="odd">
                <td class="visual">
                    ${ cotizacion.id }
                    
                    <div class="contenedor" id="${cotizacion.id }"
style="display:none;" ></div>
                </td>
                <td class="visual"> ${ cotizacion.idUsuario }</td>
                <td class="visual"> ${ cotizacion.fechaDeCotizacion }
            </td>
                <td class="visual">${ ${cotizacion.importeParcial}</td>
                <td class="visual">${ ${cotizacion.iva}</td>
                <td class="visual">${ ${cotizacion.importeTotal}</td>
            </tr>
        </c:forEach>

    </table>

</body>
</html>

```

misCotizaciones.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>

```

```

<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>

<script type="text/javascript">

function muestra_oculta(id,idCotizacion){
    if (document.getElementById){ //se obtiene el id
        var el = document.getElementById(id); //se define la variable
"el" igual a nuestro div
        el.style.display = (el.style.display == 'none') ? 'block' :
'none'; //damos un atributo display:none que oculta el div
        if(el.style.display== 'block'){

MostrarConsulta('adminCotizaciones_informacionCotizacion.action?
idCotizacion='+idCotizacion,id);
        }
    }
}

/*
    window.onload = function(){//hace que se cargue la función lo
que predetermina que div estará oculto hasta llamar a la función
nuevamente

    muestra_oculta('contenido_a_mostrar');//"contenido_a_mostrar"
es el nombre que le dimos al DIV
}*/

function objetoAjax(){
    var xmlhttp=false;
    try {
        xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
    } catch (e) {
        try {
            xmlhttp = new
ActiveXObject("Microsoft.XMLHTTP");
        } catch (E) {
            xmlhttp = false;
        }
    }

    if (!xmlhttp && typeof XMLHttpRequest!='undefined') {
        xmlhttp = new XMLHttpRequest();
    }
    return xmlhttp;
}

function MostrarConsulta(datos,id){
    divResultado =document.getElementById(id);
    ajax=objetoAjax();
    ajax.open("GET", datos);
    ajax.onreadystatechange=function() {

```

```

        if (ajax.readyState==4) {
            divResultado.innerHTML = ajax.responseText;
        }
    }
    ajax.send(null);
}
</script>

```

```

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>COFA</title>
</head>
<body>

```

```

    <br>

```

```

    <c:if test="${tipo==1}">
        Mis cotizaciones registradas
    </c:if>

```

```

    <c:if test="${tipo==2}">
        Mis cotizaciones cerradas
    </c:if>

```

```

    <br><br>

```

```

    <a href="adminCotizaciones_listarPorUsuario.action">
        </img>
    </a>

```

```

    <a
href="adminCotizaciones_listarPorUsuarioCerradas.action">
        </img>
    </a>

```

```

    <br>

```

```

<table class="visual" align="center" width="80%" border="1">
    <thead class="visual"><tr class="visual">
        <th colspan="10" class="visual" align="center">
            <div class="label">Mis cotizaciones</div>
        </th>
    </tr></thead>
    <thead class="visual"><tr>
        <th class="visual" align="center"><div
class="visual">Id</div></th>
        <th class="visual" align="center"><div class="visual">Fecha
de cotización</div></th>
        <th class="visual" align="center"><div class="visual">Importe
parcial</div></th>
        <th class="visual" align="center"><div
class="visual">IVA</div></th>
        <th class="visual" align="center"><div class="visual">Importe
total</div></th>

```

```

                <c:if test="\${tipo==1}">
                    <th class="visual" align="center"><div
class="visual">Cerrar <br> cotización</div></th>
                    <th class="visual" align="center"><div
class="visual">Cancelar <br> cotización</div></th>
                </c:if>
                <c:if test="\${tipo==2}">
                    <th class="visual" align="center"><div
class="visual">Cancelar <br> cotización</div></th>
                </c:if>

            </tr></thead>

            <c:forEach varStatus="status" var="cotizacion" items="\$
{ cotizaciones }">
                <tr class="odd">
                    <td class="visual">
                        ${ cotizacion.id }
                        
                        <div class="contenedor" id="\${cotizacion.id }"
style="display:none;" ></div>
                    </td>
                    <td class="visual"> ${ cotizacion.fechaDeCotizacion }
</td>
                    <td class="visual"> \$ ${cotizacion.importeParcial}</td>
                    <td class="visual"> \$ ${cotizacion.iva}</td>
                    <td class="visual"> \$ ${cotizacion.importeTotal}</td>
                    <c:if test="\${ tipo==1 }">
                        <td align="center">
                            <a
href="adminCotizaciones_cerrarCotizacion.action?idCotizacion=\$
{cotizacion.id}">
                                </img>
                            </a>
                        </td>
                        <td align="center">
                            <a
href="adminCotizaciones_cancelarCotizacion.action?idCotizacion=\$
{cotizacion.id}">
                                </img>
                            </a>
                        </td>
                    </c:if>

                    <c:if test="\${ tipo==2 }">

                        <td align="center">
                            <a
href="adminCotizaciones_cancelarCotizacion.action?idCotizacion=\$
{cotizacion.id}">

```



```

                </img>
                </a>
            </td>
        </c:if>
    </tr>
</c:forEach>

</table>

</body>
</html>

```

pantallaCotizacion.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>
</head>
<body>
<center>

<br>

<table class="visual" align="center" width="80%" border="1">
    <thead class="visual"><tr class="visual">
        <th colspan="10" class="visual" align="center">
            <div class="label">Cotización</div>
        </th>
    </tr></thead>
    <thead class="visual"><tr>
        <th class="visual" align="center"><div
class="visual">Nombre</div></th>
        <th class="visual" align="center"><div
class="visual">Precio<br>unitario</div></th>
        <th class="visual" align="center"><div
class="visual">Cantidad</div></th>
        <th class="visual" align="center"><div
class="visual">IVA</div></th>
        <th class="visual" align="center"><div
class="visual">Importe</div></th>

```

```

        <th class="visual" align="center"><div
class="visual">Total</div></th>
    </tr></thead>

    <c:forEach varStatus="status" var="productoEnCarrito"
items="{ listaProductosCotizacion }">
        <tr class="odd">
            <td class="visual">${ productoEnCarrito.nombre }</td>
            <td class="visual">$ $
{ productoEnCarrito.precioUnitario }</td>
            <td class="visual">${ productoEnCarrito.cantidad }</td>
            <td class="visual">$ ${productoEnCarrito.iva}</td>
            <td class="visual">$ $
{productoEnCarrito.importeParcial}</td>
            <td class="visual">$ $
{productoEnCarrito.importeTotal}</td>
        </tr>
    </c:forEach>

</table>

<br><br>

<table class="visual">
    <thead class="visual"><tr class="visual">
        <th colspan="10" class="visual" align="center">
            <div class="label">Totales</div>
        </th>
    </tr></thead>

    <thead class="visual">
        <tr class="visual">
            <th class="visual">Importe</th>
            <th class="visual">IVA</th>
            <th class="visual">Total</th>
        </tr>
        <tr class="visual">
            <td class="visual">${s:property
value="cotizacion.importeParcial"}</s:property></td>
            <td class="visual">${s:property
value="cotizacion.iva"}</s:property></td>
            <td class="visual">${s:property
value="cotizacion.importeTotal"}</s:property></td>
        </tr>
    </thead>
</table>

<br><br>
<a href="cotizaciones_guardarCotizacion.action">
    </img>
</a>

</center>
</body>
</html>

```

Facturas

generarFacturas.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>
<s:head theme="ajax"/>
</head>
<body>

    <center>

        <table class="visual" align="center" width="80%" border="1">
        <thead class="visual"><tr class="visual">
            <th colspan="10" class="visual" align="center">
                <div class="label">Lista de cotizaciones cerradas</div>
            </th>
        </tr></thead>
        <thead class="visual"><tr>
            <th class="visual" align="center"><div
class="visual">Id</div></th>
            <th class="visual" align="center"><div class="visual">Id
Usuario</div></th>
            <th class="visual" align="center"><div class="visual">Fecha
de cotización</div></th>
            <th class="visual" align="center"><div class="visual">Importe
parcial</div></th>
            <th class="visual" align="center"><div
class="visual">IVA</div></th>
            <th class="visual" align="center"><div class="visual">Importe
total</div></th>
            <th class="visual" align="center"><div
class="visual">Facturar</div></th>
        </tr></thead>

        <s:form
action="administradorFacturas_seleccionaCotizacionesAFacturar"
method="post">

            <c:forEach varStatus="status" var="cotizacion" items="$
{ cotizaciones }">
                <tr class="odd">
                    <td class="visual">${ cotizacion.id }</td>
                    <td class="visual"> ${ cotizacion.idUsuario }</td>
                    <td class="visual"> ${ cotizacion.fechaDeCotizacion }
                </td>

                    <td class="visual">${ cotizacion.importeParcial}</td>
                    <td class="visual">${ cotizacion.iva}</td>
```

```

                <td class="visual">$ ${cotizacion.importeTotal}</td>
                <td class="visual">
                    <input type="checkbox"
name="cotizacionesSeleccionadas" value="${cotizacion.id}">
                </td>
            </tr>
        </c:forEach>

        <tr></tr><tr>
            <td>fecha limite</td>
            <td><s:datetimepicker name="fechaLimiteDePago"
type="date" displayFormat="yyyy-MM-dd"></s:datetimepicker></td>
            <s:submit value="enviar" id="search-submit"></s:submit> <tr>
            </s:form>
        </table>

</center>
</body>
</html>

```

informacionDeFactura.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib prefix="sx" uri="/struts-dojo-tags"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>
</head>
<body>

    <table align="left">
        <thead class="visual"><tr>
            <th colspan="10" class="visual" align="left">
                <div class="label">Información de empresa</div>
            </th>
        </tr></thead>
        <tr class="visual">
            <td class="visual">Empresa</td>
            <td class="visual">Negocio con nombre</td>
        </tr>
        <tr class="visual">
            <td class="visual">RFC</td>
            <td class="visual">GOHE851004KL6</td>
        </tr>
        <tr class="visual">

```

```

        <td class="visual">Domicilio fiscal</td>
    <td class="visual">Calle 1 No.2 Delegación Gustavo A.
Madero</td>
</tr>
    <tr class="visual">
        <td class="visual">Teléfono</td>
        <td class="visual">56243582</td>
    </tr>

<thead class="visual"><tr>
    <th colspan="10" class="visual" align="left">
        <div class="label">Datos de factura</div>
    </th>
</tr></thead>

<tr class="visual">
    <td class="visual">Folio</td>
    <td class="visual">${facturaADetalle.folio }</td>
</tr>
<tr class="visual">
    <td class="visual">Serie</td>
    <td class="visual">${facturaADetalle.serie }</td>
</tr>
<tr class="visual">
    <td class="visual">Fecha de Expedición</td>
    <td class="visual">${
{facturaADetalle.fechaDeExpedicion}</td>
</tr>
<tr class="visual">
    <td class="visual">Cliente</td>
    <td class="visual">RFC</td>
</tr>
<tr class="visual">
    <td class="visual">RFC de cliente</td>
    <td class="visual">RFC</td>
</tr>

<thead class="visual"><tr>
    <th colspan="10" class="visual" align="left">
        <div class="label">Insumos</div>
    </th>
</tr></thead>

<tr class="visual">
    <th class="visual">Producto</th>
    <th class="visual">Precio <br>Unitario</th>
    <th class="visual">Cantidad</th>
    <th class="visual">Iva</th>
    <th class="visual">Importe</th>
    <th class="visual">Total</th>
    <c:forEach varStatus="status" var="producto" items="$
{productosEnCotizacion}">
        <tr class="visual">
            <td class="visual">${
{producto.idProducto}</td>
            <td class="visual">${
{producto.precioUnitario}</td>
            <td class="visual">${{producto.cantidad}</td>
            <td class="visual">${ ${producto.iva}</td>
            <td class="visual">${ ${producto.importe}</td>
            <td class="visual">${ ${producto.total}</td>

```

```

        </tr>
    </c:forEach>
    <td class="visual"></td>
    <td class="visual"></td>
    <td class="visual"><strong>Total</strong></td>
    <td class="visual"><strong>$ $
{cotizacionFacturada.iva}</strong></td>
    <td class="visual"><strong>$ $
{cotizacionFacturada.importeParcial}</strong></td>
    <td class="visual"><strong>$ $
{cotizacionFacturada.importeTotal}</strong></td>
    </tr>

</table>

<!--
<s:property value="facturaADetalle.toString()" />
<br><br>
<s:property value="cotizacionFacturada.toString()" />
-->
</body>
</html>

```

listar.jsp

```

<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>

<script type="text/javascript">

function muestra_oculta(id,idFactura) {
    if (document.getElementById){ //se obtiene el id
        var el = document.getElementById(id); //se define la variable
"el" igual a nuestro div
        el.style.display = (el.style.display == 'none') ? 'block' :
'none'; //damos un atributo display:none que oculta el div
        if(el.style.display== 'block'){

            MostrarConsulta('facturas_mostrarInformacionDeFactura.action?
idFactura='+idFactura,id);
        }
    }
}

```

```

        /*
        window.onload = function(){//hace que se cargue la función lo
que predetermina que div estará oculto hasta llamar a la función
nuevamente

        muestra_oculta('contenido_a_mostrar');//"contenido_a_mostrar"
es el nombre que le dimos al DIV
        }*/

function objetoAjax(){
    var xmlhttp=false;
    try {
        xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
    } catch (e) {
        try {
            xmlhttp = new
ActiveXObject("Microsoft.XMLHTTP");
        } catch (E) {
            xmlhttp = false;
        }
    }

    if (!xmlhttp && typeof XMLHttpRequest!='undefined') {
        xmlhttp = new XMLHttpRequest();
    }
    return xmlhttp;
}

function MostrarConsulta(datos,id){
    divResultado =document.getElementById(id);
    ajax=objetoAjax();
    ajax.open("GET", datos);
    ajax.onreadystatechange=function() {
        if (ajax.readyState==4) {
            divResultado.innerHTML = ajax.responseText;
        }
    }
    ajax.send(null);
}
</script>

</head>
<body>

    <br><br>
    <table class="visual" align="center" width="80%" border="1">
    <thead class="visual"><tr class="visual">
        <th colspan="10" class="visual" align="center">
            <div class="label">Facturas vigentes</div>
        </th>
    </tr></thead>
    <thead class="visual"><tr class="visual">
        <th class="visual" align="center"><div
class="label">Folio</div></th>
        <th class="visual" align="center"><div
class="label">Estado</div></th>
        <th class="visual" align="center"><div class="label">Id
usuario</div></th>

```

```

        <th class="visual" align="center"><div class="label">Fecha de
expedición</div></th>
        <th class="visual" align="center"><div class="label">Fecha
límite de pago</div></th>
        <th class="visual" align="center"><div
class="label">Descargar pdf</div></th>

</tr></thead>
<c:forEach varStatus="status" var="factura" items="$
{ facturas }">
    <tr>
        <td>
            ${ factura.folio }
            
            <div class="contenedor" id="${ factura.folio }"
style="display:none;" ></div>
        </td>
        <td>${ factura.idEstado }</td>
        <td>${ factura.idUsuario }</td>
        <td>${ factura.fechaDeExpedicion }</td>
        <td>${ factura.fechaLimiteDePago }</td>
        <td align="center">
            <a href="download.action?folioFactura=$
{factura.folio}">
                
            </a>
        </td>
    </tr>
</c:forEach>
</table>

</body>
</html>

```

misFacturas.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib prefix="sx" uri="/struts-dojo-tags"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>

```



```

<!-- <script src="{ pageContext.request.contextPath }/js/ajax.js"
type="text/javascript"></script> -->
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>

<script type="text/javascript">

function muestra_oculta(id,idFactura){
    if (document.getElementById){ //se obtiene el id
        var el = document.getElementById(id); //se define la variable
"el" igual a nuestro div
        el.style.display = (el.style.display == 'none') ? 'block' :
'none'; //damos un atributo display:none que oculta el div
        MostrarConsulta('facturas_mostrarInformacionDeFactura.action?
_idFactura='+idFactura,id);
    }
}

/*
    window.onload = function(){//hace que se cargue la función lo
que predetermina que div estará oculto hasta llamar a la función
nuevamente

    muestra_oculta('contenido_a_mostrar');//"contenido_a_mostrar"
es el nombre que le dimos al DIV
}*/

function objetoAjax(){
    var xmlhttp=false;
    try {
        xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
    } catch (e) {
        try {
            xmlhttp = new
ActiveXObject("Microsoft.XMLHTTP");
        } catch (E) {
            xmlhttp = false;
        }
    }

    if (!xmlhttp && typeof XMLHttpRequest!='undefined') {
        xmlhttp = new XMLHttpRequest();
    }
    return xmlhttp;
}

function MostrarConsulta(datos,id){
    divResultado =document.getElementById(id);
    ajax=objetoAjax();
    ajax.open("GET", datos);
    ajax.onreadystatechange=function() {
        if (ajax.readyState==4) {
            divResultado.innerHTML = ajax.responseText;
        }
    }
    ajax.send(null);
}
</script>

```


pantallaAdministracion.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>
</head>
<body>
<center>

    Aca se listaran las facturas que estan vigentes
    <br><br>

    <a href="administradorFacturas_generarFacturas.action" >
        </img>
    </a>
    <a href="administradorFacturas_listar.action" >
        </img>
    </a>
</center>
</body>
</html>
```

Productos

administrarProdcutos.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>
</head>
<body>
<center>

    Aca se administraran los productos
    <br><br>
    <a href="productos_forma_insertarProducto.action" >
```

```

        </img>
    </a>
    <a href="productos_listarParaModificar.action" >
        </img>
    </a>
</center>
</body>
</html>

```

forma_insertarProducto.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>
</head>
<body>

    <s:form action="productos_insertar" enctype="multipart/form-data"
method="post">
        <table >
            <tr>
                <td>Nombre</td>
                <td> <s:textfield name="producto.nombre" size="30"
id="register-text"/> </td>
            </tr>

            <tr>
                <td>Marca</td>
                <td>
                    <s:select name="producto.idMarca" id="register-text"
headerKey="-1"
headerValue="-- Seleccione marca --"
list="marcas"
/>
                </td>
            </tr>
        </table>
    </form>

```

```

        <tr>
        <td>Clasificación</td>
        <td>
            <s:select name="producto.idSubtipo" id="register-text"
            headerKey="-1"
            headerValue="-- Seleccione subtipo --"
            list="subtiposDeProductos"
            />
        </td>
        </tr>

        <tr>
        <td>Precio </td>
        <td><s:textfield name="producto.precioUnitario" size="30"
id="register-text"/> </td>
        </tr>

        <tr>
        <td>Descripcion</td>
        <td><s:textarea name="producto.descripcion" cols="70" rows="7"
id="register-text"/></td>
        </tr>

        <tr>
        <td>Imagen</td>
        <td><s:file name="upload" /></td>
        </tr>

        <tr>
        <td>Disponibles</td>
        <td><s:textfield name="producto.disponibles" size="30"
id="register-text"/></td>
        </tr>

        <tr>
        <td></td>
        <td><s:submit value=" Insertar " id="register-submit"/></td>
        </tr>

    </table>

</s:form>

</body>
</html>

```

forma_modificarProducto.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

```

```

<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>
</head>
<body>

<h1>Modificar producto</h1>

<br><br>

<s:form action="productos_modificar" enctype="multipart/form-data"
method="post">

    <input type="hidden" name="producto.id" value="$
{ productoAModificar.id}">

    <table >
        <tr>
            <td>Nombre</td>
            <td><input type="text" name="producto.nombre" size="30"
id="register-text" value="{productoAModificar.nombre}"/> </td>
        </tr>

        <tr>
            <td>Marca</td>
            <td>
                <s:select name="producto.idMarca" id="register-text"
headerKey="-1"
headerValue="-- Seleccione marca --"
list="marcas"
/>
            </td>
        </tr>

        <tr>
            <td>Clasificación</td>
            <td>
                <s:select name="producto.idSubtipo" id="register-text"
headerKey="-1"
headerValue="-- Seleccione subtipo --"
list="subtiposDeProductos"
/>
            </td>
        </tr>

        <tr>
            <td>Precio </td>
            <td>
                <input type="text" name="producto.precioUnitario"
size="30" id="register-text"

```

```

        value="{productoAModificar.precioUnitario}" />
    </td>
</tr>

<tr>
<td>Descripcion</td>
<td><textarea name="producto.descripcion" cols="100" rows="10"
id="register-text" >

        </textarea>
</td>
</tr>

<tr>
<td>Imagen</td>
<td><s:file name="upload" /></td>
</tr>

<!--
<tr>
<td>Disponibles</td>
<td><input type="text" name="producto.disponibles" size="30"
id="register-text" value="{productoAModificar.disponibles}"/></td>
</tr>
-->

<tr>
<td></td>
<td><s:submit value=" Insertar " id="register-submit"/></td>
</tr>

</table>

</s:form>

</body>
</html>

```

listar.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<link rel="stylesheet" href="css/style_menuDesplegable.css"
type="text/css" />
<title>COFA</title>
</head>
<body>

    <!-- aca se crea el menu de tipos y subtipos de productos -->
    <ul id="css3menu1" class="topmenu">
    <c:forEach var="tipo" items="{ listaTiposProductos }">
        <li class="topmenu"><a href="" title="{tipo.tipo}">${
{tipo.tipo}</a>
        <ul>
            <c:forEach var="subtipo" items="{
{ listaSubtiposProductos }">
                <c:if test="{tipo.id ==
subtipo.idTipoDeProducto }">
                    <li>
                        <a
href="productos_listarPorTipo.action?id='subtipo ${subtipo.id}'"
title="{subtipo.subtipo}">${
{subtipo.subtipo}</a>
                    </li>
                </c:if>
            </c:forEach>
        </ul>
    </li>
</c:forEach>
</ul>
<!-- Termina de generarse en menu de los productos -->
<br>.<br>.<br>.<br>.<br>

    <table class="visual" align="center" width="80%" border="1">
    <thead class="visual"> <tr class="visual">
        <th colspan="10" class="visual" align="center">
            <div class="label">Lista productos</div>
        </th>
    </tr> </thead>
    <thead class="visual"><tr>
        <th class="visual" align="center"><div
class="label">Imagen</div></th>
        <th class="visual" align="center"><div
class="label">Nombre</div></th>
        <th class="visual" align="center"><div
class="label">Marca</div></th>
        <th class="visual" align="center"><div
class="label">Precio</div></th>
        <th class="visual" align="center"><div
class="label">Clasificación</div></th>
    </tr></thead>

    <c:forEach varStatus="status" var="producto" items="{
{ productos }">
        <tr class="odd">

```



```

        <!-- <td>" /></td> -->
        <td>
            <s:url id="image" action="getPortrait"
namespace="/user" >
                <s:param name="producto"
value="producto.imagen" />
            </s:url>
            " />
<br/>
        </td>
        <td>${ producto.nombre }</td>
        <td>${ producto.idMarca }</td>
        <td>${ producto.precioUnitario }</td>
        <td>${ producto.idSubtipo }</td>
    </tr>
</c:forEach>
</table>

</body>
</html>

```

listarParaModificar.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<link rel="stylesheet" href="css/style_menuDesplegable.css"
type="text/css" />
<title>COFA</title>
</head>
<body>

    <!-- aca se crea el menu de tipos y subtipos de productos -->
    <ul id="css3menu1" class="topmenu">
    <c:forEach var="tipo" items="${ listaTiposProductos }">
        <li class="topmenu"><a href="" title="${tipo.tipo}">${
{tipo.tipo}</a>
    </li>
    </ul>

```

```

        <c:forEach var="subtipo" items="$
{ listaSubtiposProductos }">
            <c:if test="{tipo.id ==
subtipo.idTipoDeProducto }">
                <li>
                    <a
href="productos_listarPorTipo_modificar.action?id='subtipo $
{subtipo.id}' "
                    title="{subtipo.subtipo}">$
{subtipo.subtipo}</a>
                </li>
            </c:if>
        </c:forEach>
    </ul>
</li>
</c:forEach>
</ul>
<!-- Termina de generarse en menu de los productos -->
<br>.<br>.<br>.<br>.<br>

<table class="visual" align="center" width="80%" border="1">
<thead class="visual"> <tr class="visual">
    <th colspan="10" class="visual" align="center">
        <div class="label">Lista productos</div>
    </th>
</tr> </thead>
<thead class="visual"><tr>
    <th class="visual" align="center"><div
class="label">Imagen</div></th>
    <th class="visual" align="center"><div
class="label">Nombre</div></th>
    <th class="visual" align="center"><div
class="label">Marca</div></th>
    <th class="visual" align="center"><div
class="label">Precio</div></th>
    <th class="visual" align="center"><div
class="label">Clasificación</div></th>
    <th class="visual" align="center"><div
class="label">Borrar</div></th>
    <th class="visual" align="center"><div
class="label">modificar</div></th>
</tr></thead>

    <c:forEach varStatus="status" var="producto" items="$
{ productos }">
        <tr class="odd">
            <!-- <td>" /></td> -->
            <td>
                <s:url id="image" action="getPortrait"
namespace="/user" >
                    <s:param name="producto"
value="producto.imagen" />
                </s:url>
                " />
            </td>
        </tr>
    </c:forEach>
</table>

```

```

        </td>
        <td>${ producto.nombre }</td>
        <td>${ producto.idMarca }</td>
        <td>${ producto.precioUnitario }</td>
        <td>${ producto.idSubtipo }</td>
        <td align="center">
            <a href="productos_borrar.action?idProducto=${
{ producto.id }}">
                </img>
            </a>
        </td>
        <td align="center">
            <a href="productos_forma_modificarProducto.action?
idProducto=${ producto.id }}">
                </img>
            </a>
        </td>
    </tr>
</c:forEach>
</table>

</body>
</html>

```

Usuarios

forma_insertarCliente.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>
</head>
<body>

<s:fielderror theme="xhtml"/>

<s:form action="usuarios_insertarCliente" >
    <table >
        <tr>

```

```

        <td><strong>Nombres</strong></td>
        <td> <s:textfield name="usuario.nombres" size="30"
id="register-text"/> </td>
    </tr>

    <tr>
        <td><strong>Apellido paterno</strong></td>
        <td> <s:textfield name="usuario.apellidoPaterno" size="30"
id="register-text"/></td>
    </tr>

    <tr>
        <td><strong>Apellido materno</strong></td>
        <td> <s:textfield name="usuario.apellidoMaterno" size="30"
id="register-text"/> </td>
    </tr>

    <tr>
        <td><strong>Ocupación</strong> </td>
        <td>
            <s:select name="usuario.idOcupacion" id="register-text"
headerKey="-1"
headerValue="-- Selecciona ocupacion --"
list="ocupaciones"
/>
        </td>
    </tr>

    <tr>
        <td><strong>Edad</strong></td>
        <td><s:textfield name="usuario.edad" size="30" id="register-
text"/></td>
    </tr>

    <tr>
        <td><strong>RFC</strong></td>
        <td><s:textfield name="usuario.rfc" size="30" id="register-
text"/>
        <td><a href="http://www.consisarh.com.mx/rfc.html"
target="_blank"> Consultar RFC</a></td>
    </tr>

    <tr>
        <td><strong>login</strong></td>
        <td><s:textfield name="datosDeAcceso.login" size="30"
id="register-text"/></td>
    </tr>

    <tr>
        <td><strong>password</strong></td>
        <td><s:password name="datosDeAcceso.password" size="30"
id="register-text"/></td>
    </tr>

    <!--
    <tr>

```

```

        <td>confirmar password</td>
        <td><s:password name="confirmacionPassword" size="30"
id="register-text"/></td>
    </tr>

    -->

    <tr>
    <td></td>
    <td><s:submit value="  Insertar  " id="register-submit"/></td>
    </tr>

</table>

</s:form>

</body>
</html>

```

forma_insertarDireccion.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>Registrar direccion</title>
</head>
<body>

    <!-- <s:fielderror theme="xhtml"/> -->

    <s:form action="direcciones_insertarDireccion" >
        <table >
            <tr>
                <td><strong>Calle</strong></td>
                <td> <s:textfield name="direccion.calle" size="30"
id="register-text"/> </td>
            </tr>

            <tr>
                <td><strong>Número interior</strong></td>
                <td> <s:textfield name="direccion.numeroInt" size="30"
id="register-text"/></td>
            </tr>

```

```

        <tr>
        <td><strong>Número exterior</strong></td>
        <td> <s:textfield name="direccion.numeroExt" size="30"
id="register-text"/> </td>
        </tr>

        <tr>
        <td><strong>Colonia</strong></td>
        <td> <s:textfield name="direccion.colonia" size="30"
id="register-text"/></td>
        </tr>

        <tr>
        <td><strong>Código postal</strong></td>
        <td><s:textfield name="direccion.codigoPostal" size="30"
id="register-text"/></td>
        </tr>

        <tr>
        <td><strong>Ciudad</strong></td>
        <td><s:textfield name="direccion.ciudad" size="30"
id="register-text"/></td>
        </tr>

        <tr>
        <td></td>
        <td><s:submit value=" Insertar " id="register-submit"/></td>
        </tr>

    </table>

</s:form>

</body>
</html>

```

forma_insertarModosDeContacto.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">

```

```

<title>Insert title here</title>
</head>
<body>

    <s:form action="modosDeContacto_insertarModosDeContacto" >
    <table >
        <tr>
            <td><strong>Correo</strong></td>
            <td> <s:textfield name="correo" size="30" id="register-text" />
        </td>
        </tr>

        <!--
        <tr>
            <td>Confirmar correo</td>
            <td> <s:textfield name="confirmacionCorreo" size="30"
id="register-text"/></td>
        </tr>
        -->

        <tr>
            <td><strong>Teléfono casa</strong></td>
            <td> <s:textfield name="telefonoCasa" size="30" id="register-
text"/> </td>
        </tr>

        <tr>
            <td><strong>Teléfono celular</strong></td>
            <td> <s:textfield name="telefonoCelular" size="30"
id="register-text"/></td>
        </tr>

        <tr>
            <td><strong>Teléfono trabajo</strong></td>
            <td><s:textfield name="telefonoTrabajo" size="30" id="register-
text"/></td>
        </tr>

        <tr>
            <td><strong>Fax</strong></td>
            <td><s:textfield name="fax" size="30" id="register-text"/></td>
        </tr>

        <tr>
            <td><s:submit value=" Insertar " id="register-submit"/></td>
        </tr>

    </table>

</s:form>

```

```
</body>
</html>
```

registroExitoso.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>Insert title here</title>
</head>
<body>

    <div id="wrapper">

        <!-- end #header -->
        <div id="page">
            <div id="page-bgtop">
                <div id="page-bgbtm">
                    <div id="content">
                        <div class="post">
                            <h2 class="title"><a
href="#">Gracias por registrarte.</a></h2>

                                <div style="clear:
both;">&nbsp;</div>

                                    <div class="entry">
                                        <h2>Tu registro ha
finalizado correctamente.</h2> <br>
<p class="links"><a href="index.action">Accede aqui para
continuar</a>

                                            </div>

                                                </div>

                                                    <div style="clear: both;"></div>

                                                        </div>
<!-- end #content -->

                    <div style="clear: both;">&nbsp;</div>
                </div>
            </div>
        </div>
    </div>
<br><br><br><br><br><br><br><br><br><br><br><br><br><br><br>
<!-- end #page -->
```



```
</div>
```

```
</body>
```

```
</html>
```

forma_validarUsuario.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>Insert title here</title>
</head>
<body>
```

Ingresar tus datos

```
<br>
```

```
<h1><s:property value="mensaje"/></h1>
```

```
<s:form action="validar_validarUsuario" >
```

```
  <table>
```

```
    <tr>
```

```
      <td>login</td>
```

```
      <td><s:textfield name="datosDeAcceso.login" size="30"/></td>
```

```
    </tr>
```

```
    <tr>
```

```
      <td>password</td>
```

```
      <td><s:password name="datosDeAcceso.password" size="30"/></td>
```

```
    </tr>
```

```
    <tr>
```

```
      <td></td>
```

```
      <td><s:submit value="Enviar"/></td>
```

```
    </tr>
```

```
  </table>
```

```
</s:form>
```

```
<br>.<br>.<br>.<br>.<br>.<br>.<br>.<br>.<br>.
```

```
</body>
</html>
```

Error.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>Insert title here</title>
</head>
<body>

<s:actionerror theme="xhtml"/>

</body>
</html>
```

index_registrado.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>
</head>

<body>

<div id="wrapper">

    <div id="page">
```



```

                                <div style="clear:
both;"></div>
                                </li>
                                <br/><br/>
                                </ul>
                                </div>
                                <!-- end #sidebar -->
                                <div style="clear: both;">&nbsp;</div>
                                </div>
                                </div>
                                <!-- end #page -->
</div>

</body>
</html>

```

index.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="s" uri="/struts-tags" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://acsinet-solutions.com/pager" prefix="pager"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>COFA</title>
</head>

<body>

<div id="wrapper">

    <div id="page">
        <div id="page-bgtop">
            <div id="page-bgbtm">
                <div id="content">
                    <div class="post">

                                <h2 class="title"><a
href="#">Bienvenido a cofa</a></h2>

                                <div style="clear:
both;">&nbsp;</div>

                                <div class="entry">

```



```

        </s:form>
declaracion del formulario -->        <!-- Termina la

                                        </div>
both;"></div>                        <div style="clear:
                                        </li>

                                        </ul>
                                        </div>
                                        <!-- end #sidebar -->
                                        <div style="clear: both;">&nbsp;</div>
                                </div>
                                </div>
                                </div>
                                <!-- end #page -->
</div>

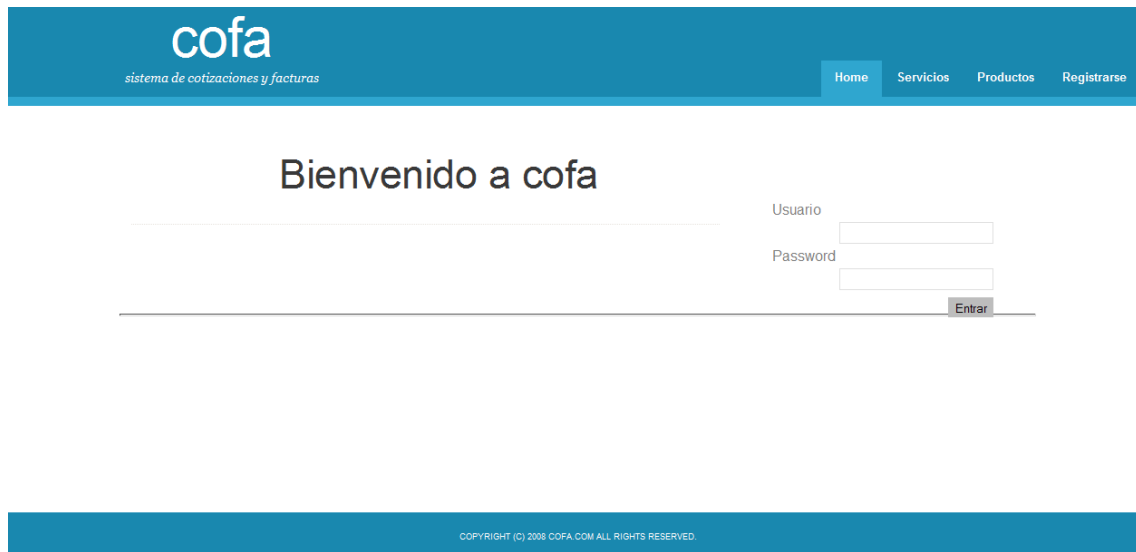
</body>
</html>

```

Manual del usuario

Ingresar al sistema

En esta pantalla introducirá sus datos de acceso con los que se registro, si no está registrado consulte la parte de registro de nuevo usuario.



cofa
sistema de cotizaciones y facturas

Home Servicios Productos Registrarse

Bienvenido a cofa

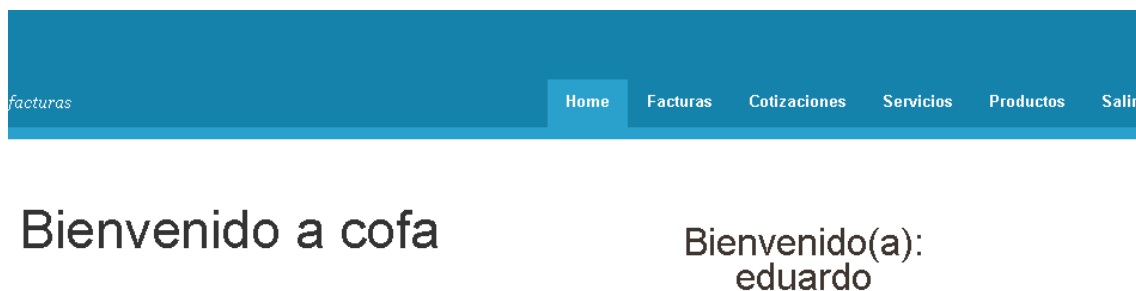
Usuario

Password

Entrar

COPYRIGHT (C) 2008 COFA.COM ALL RIGHTS RESERVED.

Si sus datos de acceso son correctos se desplegará la siguiente pantalla, desplegando el menú para usuarios registrados.



facturas

Home Facturas Cotizaciones Servicios Productos Salir

Bienvenido a cofa

Bienvenido(a):
eduardo

Si los datos introducidos no son válidos se mostrara la siguiente pantalla, permitiéndole el acceso hasta que introduzca datos de acceso válidos.

Ingresar tus datos
Datos inválidos. Favor de verificar

login

password

Enviar

Registro de nuevo usuario

En la pantalla principal deberá seleccionar la opción registrarse.

Bienvenido a cofa

Usuario

Password

Entrar

Al seleccionar la opción de mostrara la siguiente pantalla donde tendrá que ingresar sus datos personal es.

Nombres

Apellido paterno

Apellido materno

Ocupación

Edad

RFC

[Consultar RFC](#)

login

password

Es necesario que introduzca su RFC si no cuenta con el acceda a la opción **consultar RFC** de la pantalla anterior y se mostrará en otra pestaña la pantalla siguiente donde podrá consultar su RFC.



Calcular RFC y CURP

Apellido paterno:

Apellido materno:

Nombre(s):

Fecha de nacimiento (dd/mm/aaaa):

Entidad Federativa de nacimiento:

Sexo:

RFC: RARY8608165F2

CURP: RARY860816MJCNMT07

Una vez introducidos sus datos personales es momento de ingresar las formas en que se le podrá contactar. Nota el correo electrónico y el teléfono fijo son indispensables.

Correo

Teléfono casa

Teléfono celular

Teléfono trabajo

Fax

Ahora deberá ingresar su dirección.

Calle

Número interior

Número exterior

Colonia

Código postal

Ciudad

Una vez que haya registrado todos sus datos se desplegará la siguiente pantalla. Nota el registro debe realizarse de manera completa, de lo contrario ningún dato se almacenará en nuestra base de datos.

Gracias por registrarte.
Tu registro ha finalizado
correctamente.

[Accede aquí para continuar](#)

Consultar productos

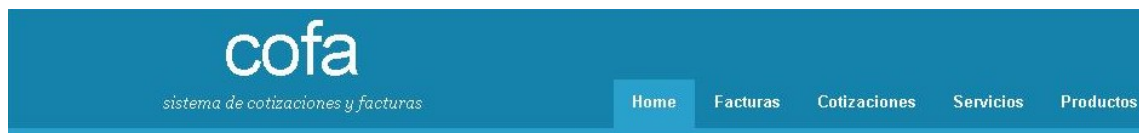
Para consultar los productos existentes en el menú principal selecciona la opción productos.



Bienvenido a cofa

Bienvenido(a):
eduardo

Se desplegará la siguiente pantalla, la cual cuenta con un menú para búsqueda especializada de productos, basta con dar click a cualquiera de las categorías para la búsqueda especializada.



Generación de cotizaciones

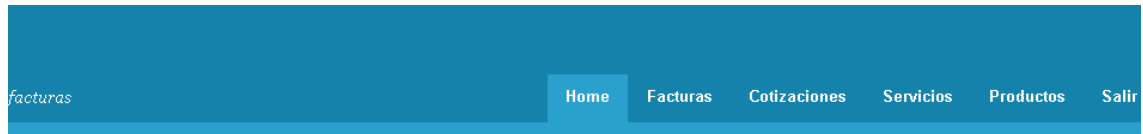
[Ver mis cotizaciones](#)

Celulares y accesorios	Computación	Consolas y videojuegos	Electrónica	Audio	Vídeo	Hogar
Celulares y accesorios						
Tarjetas de memoria						
Teléfonos celulares						

Lista productos						
Imagen	Nombre	Marca	Precio	clasificación	Agregar	
[B@17c3056]	memoria 8 GB	1	\$ 252.0	1	✓	
[B@8cd4db]	Nokia N900	8	\$ 5320.0	2	✓	
	Celular Nokia C3	8	\$ 1800.0	2	✓	
	Celular Sony Ericsson W705	9	\$ 2597.0	2	✓	
	Sony Ericsson U8	9	\$ 4199.0	2	✓	
	Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	✓	
	Lap Top Dell Latitude D620 Intel Core Duo Wifi	3	\$ 3000.0	3	✓	

Generar cotización

Para generar una cotización acceda a la opción cotizaciones.



Bienvenido a cofa

Bienvenido(a):
eduardo

Se desplegará una pantalla donde podrá seleccionar los productos que desee agregar al carrito de compras. Para agregar un producto al carrito basta con dar click en cualquiera de las palomas verdes de la tabla de productos.



Generación de cotizaciones

Ver mis
Cotizaciones

Celulares y accesorios Computación Consolas y videojuegos Electrónica Audio Video Hogar

Lista productos						
Imagen	Nombre	Marca	Precio	clasificación	Agregar	
[B@fb0b0]	memoria 8 GB	1	\$ 2220.0	4	✓	
[B@18ff27]	Nokia N900	8	\$ 5320.0	2	✓	
	Celular Nokia C3	8	\$ 1800.0	2	✓	
	Celular Nokia C3	8	\$ 1800.0	2	✓	
	Celular Sony Ericsson W705	9	\$ 2597.0	2	✓	
	Sony Ericsson U8	9	\$ 4199.0	2	✓	
	Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	✓	
	Lap Top Dell Latitude D620 Intel Core Duo Wifi	3	\$ 3000.0	3	✓	
[B@b8501d]	Mario Land 2 En Excelente Estado	4	\$ 1750.0	8	✓	
[B@93a985]	aass	3	\$ 25200.0	2	✓	
[B@6399ae]	Transmisor Fm 4 En 1 Coche Usb	2	\$ 4990.0	9	✓	
[B@56d6cf]	lavadora	8	\$ 450057.0	4	✓	

Lista de productos a cotizar				
Nombre	Marca	Precio	clasificación	Remover

Seleccionar
numero
de productos

Cuando agregue un producto al carrito de compras se desplegarán en la tabla inferior de la pantalla, para remover el producto basta con dar click en cualquiera de los taches rojos de la

tabla. Al seleccionar todos los productos que desee de click en el botón **seleccionar número de productos**.

[B@4ba4d6	lavadora	8	\$ 450057.0	4	<input checked="" type="checkbox"/>
-----------	----------	---	-------------	---	-------------------------------------

Lista de productos a cotizar				
Nombre	Marca	Precio	clasificación	Remove
Celular Sony Ericsson W705	9	\$ 2597.0	2	<input checked="" type="checkbox"/>
Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	<input checked="" type="checkbox"/>
memoria 8 GB	1	\$ 2220.0	4	<input checked="" type="checkbox"/>
Nokia N900	8	\$ 5320.0	2	<input checked="" type="checkbox"/>

Seleccionar número de productos

Se desplegara la siguiente pantalla donde deberá introducir la cantidad de productos que desee de cada tipo y después dar click en el botón enviar.

cofa
sistema de cotizaciones y facturas

[Home](#) | [Mis Facturas](#) | [Cotizaciones](#) | [Servicios](#) | [Productos](#) | [Salir](#)

Selecciona cuantos productos deseas		
Nombre	Precio	cantidad
Celular Sony Ericsson W705	\$ 2597.0	<input type="text" value="2"/>
Lap Top Toshiba 3 Gb Ram C645d	\$ 6898.0	<input type="text" value="4"/>
memoria 8 GB	\$ 2220.0	<input type="text" value="2"/>
Nokia N900	\$ 5320.0	<input type="text" value="5"/>

Se desplegará la siguiente pantalla donde se mostrará la cotización realizada en base a los productos seleccionados, en esta parte tendrá la opción de guardar la cotización si lo desea. Nota para generar una factura se debe guardar la cotización ya demás que la cotización este cerrada.

Cotización					
Nombre	Precio unitario	Cantidad	IVA	Importe	Total
Celular Sony Ericsson W705	\$ 2597.0	2	\$ 831.04	\$ 4362.96	\$ 5194.0
Lap Top Toshiba 3 Gb Ram C645d	\$ 6898.0	4	\$ 4414.7197	\$ 23177.281	\$ 27592.0
memoria 8 GB	\$ 2220.0	2	\$ 710.39996	\$ 3729.6	\$ 4440.0
Nokia N900	\$ 5320.0	5	\$ 4256.0	\$ 22344.0	\$ 26600.0

Totales		
Importe	IVA	Total
\$53613.844	\$10212.16	\$63826.004

Guardar Cotización

Si selecciona la opción de guardar cotización de desplegar la siguiente pantalla.

Cotizaciones Pendientes Cotizaciones Cerradas

Mis cotizaciones				
Id	Fecha de cotización	Importe parcial	IVA	Importe total
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004

Consultar cotizaciones

Para consultar las cotizaciones realizadas en el menú principal acceda a la opción **cotizaciones** y después **ver mis cotizaciones**

Bienvenido a cofa

Bienvenido(a):
eduardo

Se desplegara la siguiente pantalla donde podrá consultar sus cotizaciones realizadas.

Mis cotizaciones registradas

Cotizaciones Pendientes Cotizaciones Cerradas

Mis cotizaciones						
Id	Fecha de cotización	Importe parcial	IVA	Importe total	Cerrar cotización	Cancelar cotización
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	✓	✗

Ver cotización a detalle

Para ver una cotización a detalle, en la pantalla ver cotizaciones dar click en el símbolo +/-, se desplegaran los detalles de la cotización, para ocultarlo basta con dar click de nuevo en el símbolo +/-.

Mis cotizaciones cerradas

Cotizaciones Pendientes Cotizaciones Cerradas

Mis cotizaciones					
Id	Fecha de cotización	Importe parcial	IVA	Importe total	Cancelar cotización
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	✗
Insumos					
		\$ 1847.52		\$ 11547.0	✗
Producto	Precio Unitario	Cantidad	Iva	Importe	Total
	\$ 2597.0	2	\$ 831.04	\$ 4362.96	\$ 5194.0
	\$ 6898.0	4	\$ 4414.7197	\$ 23177.281	\$ 27592.0
	\$ 2220.0	2	\$ 710.39996	\$ 3729.6	\$ 4440.0
	\$ 5320.0	5	\$ 4256.0	\$ 22344.0	\$ 26600.0
Total			\$ 10212.16	\$ 53613.844	\$ 63826.00

Cerrar cotización

En la pantalla ver cotizaciones seleccione la opción cotizaciones pendientes y se desplegara la siguiente pantalla.

Mis cotizaciones registradas

Cotizaciones Pendientes

Cotizaciones Cerradas

Mis cotizaciones						
Id	Fecha de cotización	Importe parcial	IVA	Importe total	Cerrar cotización	Cancelar cotización
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	✓	✗

Para cerrar la cotización basta con dar click en la paloma verde de la cotización. Nota , para iniciar el proceso de facturación la cotización deberá estar cerrada.

Cancelar cotización

Para cancelar la cotización basta con dar click en la tache rojo de la cotización

Mis cotizaciones registradas

Cotizaciones Pendientes

Cotizaciones Cerradas

Mis cotizaciones						
Id	Fecha de cotización	Importe parcial	IVA	Importe total	Cerrar cotización	Cancelar cotización
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	✓	✗

Consultar mis facturas

Para consultar la facturas se deberá seleccionar ene le menú principal la opción facturas

Bienvenido a cofa

Bienvenido(a):
eduardo

Se desplegará la siguiente pantalla donde podrá consultar sus facturas

cofa
sistema de cotizaciones y facturas

Home Mis Facturas Cotizaciones Servicios Productos Salir



Mis facturas				
Folio	Estado	Fecha de expedición	Fecha límite de pago	Descarga pdf
12314 +/-	1	2011-04-05 11:44:55.0		
12315 +/-	1	2011-04-05 11:44:56.0		

Ver detalle de factura

Para ver los detalles de la factura en la pantalla ver facturas dar click en el símbolo +/-, se desplegarán los detalles de la cotización, para ocultarlo basta con dar click de nuevo en el símbolo +/-.

cofa
sistema de cotizaciones y facturas

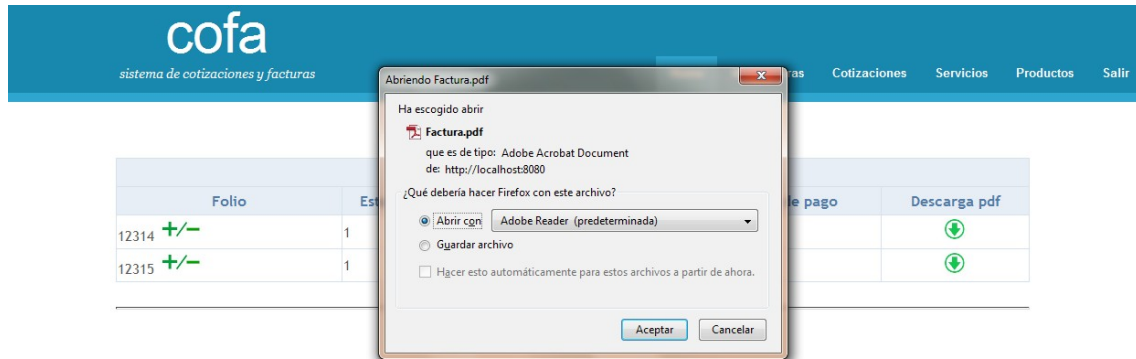
Home Mis Facturas Cotizaciones Servicios Productos Salir

Mis facturas				
Folio	Estado	Fecha de expedición	Fecha límite de pago	Descarga pdf
12314 +/-	1	2011-04-05 11:44:55.0		
12315 +/-	1	2011-04-05 11:44:56.0		

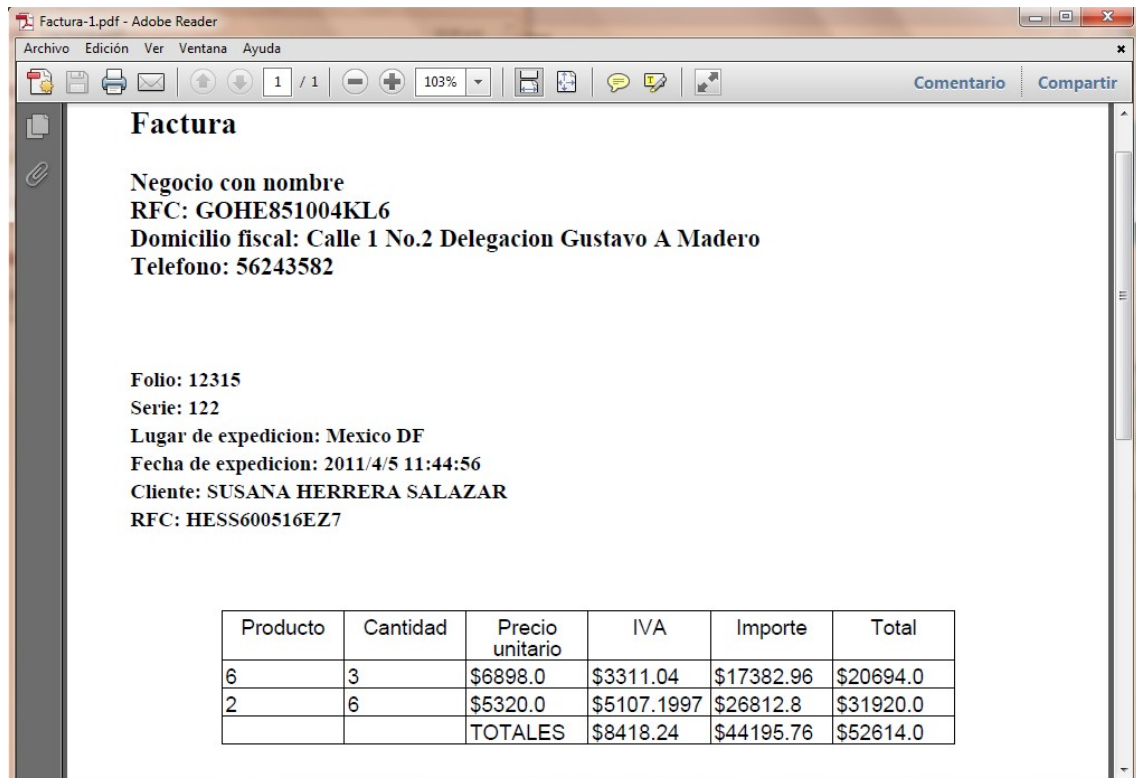
empresa		registro con número	
RFC	GOHE851004KL6		
Domicilio fiscal	Calle 1 No.2 Delegación Gustavo A. Madero		
Teléfono	56243582		
Datos de factura			
Folio	12315		
Serie	122		
Fecha de Expedición	2011-04-05 11:44:56.0		

Descargar factura

Para descargar la factura en formato PDF basta con dar click en el icono descargar factura (flecha verde apuntando hacia abajo)



Este es ejemplo de una factura descargada.



Manual usuario administrador

Ingresar al sistema

En esta pantalla introducirá sus datos de acceso con permisos de administrador.



Bienvenido a cofa

Usuario

Password



Bienvenido a cofa

Bienvenido(a):
eduardo



Ingresar tus datos
Datos inválidos. Favor de verificar

login

password

Consultar cotizaciones

Para consultar las cotizaciones en el menú principal acceda a la opción cotizaciones.



Bienvenido a cofa

Bienvenido(a):
eduardo



Cotizaciones					
Id	Id Usuario	Fecha de cotización	Importe parcial	IVA	Importe total
7 +/-	9	2011-04-06 15:57:20.0	\$ 22092.002	\$ 4208.0	\$ 26300.002
8 +/-	9	2011-04-06 15:57:52.0	\$ 14784.0	\$ 2816.0	\$ 17600.0

Insumos					
Producto	Precio Unitario	Cantidad	Iva	Importe	Total
	\$ 2220.0	1	\$ 355.19998	\$ 1864.8	\$ 2220.0
	\$ 1800.0	3	\$ 864.0	\$ 4536.0	\$ 5400.0
	\$ 1000.0	2	\$ 1506.7000	\$ 8782.2	\$ 6000.0

2011 COFA. EDUARDO GONZÁLEZ HERRERA.

Administración de facturas

Para la administración de las facturas en el menú principal acceder a la opción facturas



Bienvenido a cofa

Bienvenido(a):
eduardo

Se desplegará la siguiente pantalla de administración de facturas.



Generar facturas

Para generar facturas acceder a la opción generar facturas.



Se desplegará la siguiente pantalla, que lista las cotizaciones en estado cerrado. Para generar facturas se deberá seleccionar las cotizaciones que se deseen facturar y poner la fecha limite de pago de la cotización.





Al enviar los datos se desplegara la siguiente pantalla que muestra las facturas registradas.

Facturas vigentes						
Folio	Estado	Id usuario	Fecha de expedición	Fecha límite de pago	Descargar pdf	
12313 +/-	1	1	2011-03-24 12:36:28.0	2011-10-25		
12314 +/-	1	9	2011-04-05 11:44:55.0			
12315 +/-	1	9	2011-04-05 11:44:56.0			
12316 +/-	1	10	2011-04-05 19:30:25.0			
12317 +/-	1	9	2011-04-06 19:30:02.0			
12318 +/-	1	9	2011-04-06 19:30:02.0			

Ver detalle de factura

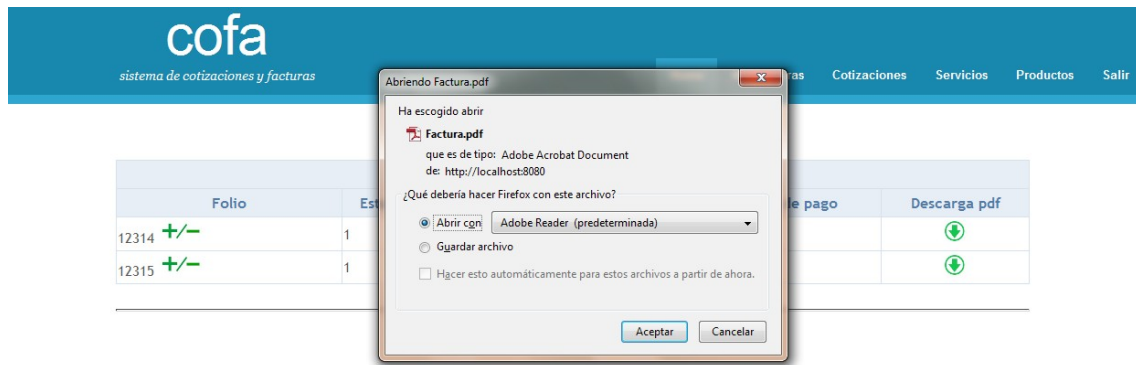
Para ver los detalles de la factura en la pantalla ver facturas dar click en el símbolo +/-, se desplegaran los detalles de la cotización, para ocultarlo basta con dar click de nuevo en el símbolo +/-.

Mis facturas				
Folio	Estado	Fecha de expedición	Fecha límite de pago	Descarga pdf
12314 +/-	1	2011-04-05 11:44:55.0		
12315 +/-	1	2011-04-05 11:44:56.0		

Empresa		Negocio con número	
RFC	GOHE851004KL6		
Domicilio fiscal	Calle 1 No.2 Delegación Gustavo A. Madero		
Teléfono	56243582		
Datos de factura			
Folio	12315		
Serie	122		
Fecha de Expedición	2011-04-05 11:44:56.0		

Descargar factura

Para descargar la factura en formato PDF basta con dar click en el icono descargar factura (flecha verde apuntando hacia abajo)



Administración de productos

Para la administración de productos deberá seleccionar en el menú principal la opción de productos.



Bienvenido a cofa

Bienvenido(a):
eduardo

Se mostrara la siguiente pantalla



Agregar producto

Para agregar un nuevo producto seleccionar la opción agregar

Aca se administraran los productos



Se desplegara la siguiente pantalla donde se deberán ingresar los datos de un producto nuevo.

Nombre

Marca

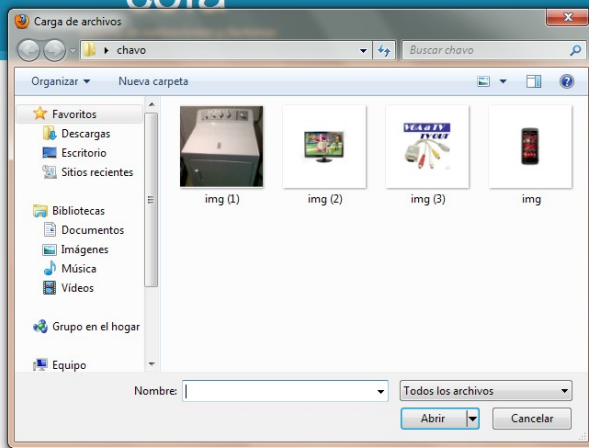
Clasificación

Precio

Descripción

Imagen

Disponibles



Disponibles

Modificar producto

Para modificar un producto seleccione la opción consultar

Aca se administraran los productos

Agregar Consultar

Se mostrara la siguiente pantalla donde se podrá elegir el producto a modificar.

Celulares y accesorios Computación Consolas y videojuegos Electrónica Audio Video Hogar

Lista productos						
Imagen	Nombre	Marca	Precio	Clasificación	Borrar	modificar
	memoria 8 GB	1	\$ 2220.0	4	X	
	Nokia N900	8	\$ 5320.0	2	X	
	Celular Nokia C3	8	\$ 1800.0	2	X	
	Celular Sony Ericsson W705	9	\$ 2597.0	2	X	
	Sony Ericsson U8	9	\$ 4199.0	2	X	
	Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	X	
	Lap Top Dell Latitude D620 Intel Core Duo Wifi	3	\$ 3000.0	3	X	
	Mario Land 2 En Excelente Estado	4	\$ 1750.0	8	X	

Al seleccionar un producto dando click en el icono de modificar se mostrara la siguiente pantalla con los datos del producto seleccionado.

Modificar producto

Nombre

Marca

Clasificación

Precio

Descripcion

Imagen

Borrar producto

cofa
sistema de cotizaciones y facturas

Home Facturas Cotizaciones Servicios Productos Salir

Aca se administraran los productos
















Agregar

Consultar

cofa
sistema de cotizaciones y facturas

Home Facturas Cotizaciones Servicios Productos Salir

Celulares y accesorios Computación Consolas y videojuegos Electrónica Audio Video Hogar

Lista productos						
Imagen	Nombre	Marca	Precio	Clasificación	Borrar	modificar
	memoria 8 GB	1	\$ 2220.0	4	X	
	Nokia N900	8	\$ 5320.0	2	X	
	Celular Nokia C3	8	\$ 1800.0	2	X	
	Celular Sony Ericsson W705	9	\$ 2597.0	2	X	
	Sony Ericsson U8	9	\$ 4199.0	2	X	
	Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	X	
	Lap Top Dell Latitude D620 Intel Core Duo Wifi	3	\$ 3000.0	3	X	
	Mario 1 and 2 En Excelente Estado	4	\$ 1750.0	8	X	

Manual de instalación

Requisitos de instalación

Para instalar la aplicación se necesitan seguir 5 pasos que se muestran a continuación y se encuentran presentados a detalle a lo largo de este documento.

1. Instalación de JDK
2. Instalación de MySQL-server
3. Restauración de la base de datos
4. Instalación del servidor de aplicaciones TOMCAT
5. Montar la aplicación web.

Instalación de jdk

Entramos a la pagina de descarga del JDK SE Development kit 6u24 (la liga de descarga para el archivo se encuentra en la parte final del documento).

SDN Home > Download Center >

Java SE Development Kit 6u24

Provide Information, then Continue to Download

Select Platform and Language for your download:

Platform: Windows

Language: Multi-language

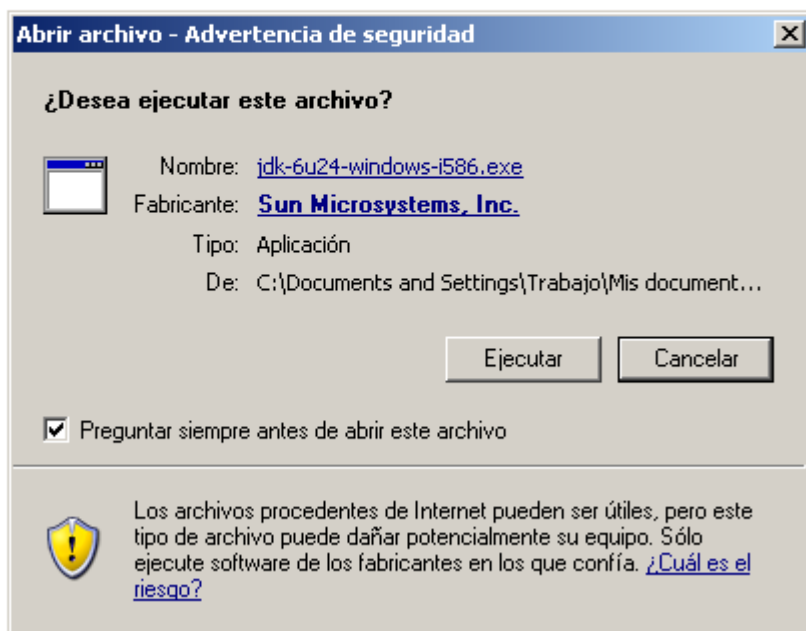
I agree to the [Java SE Development Kit 6u24 License Agreement](#).

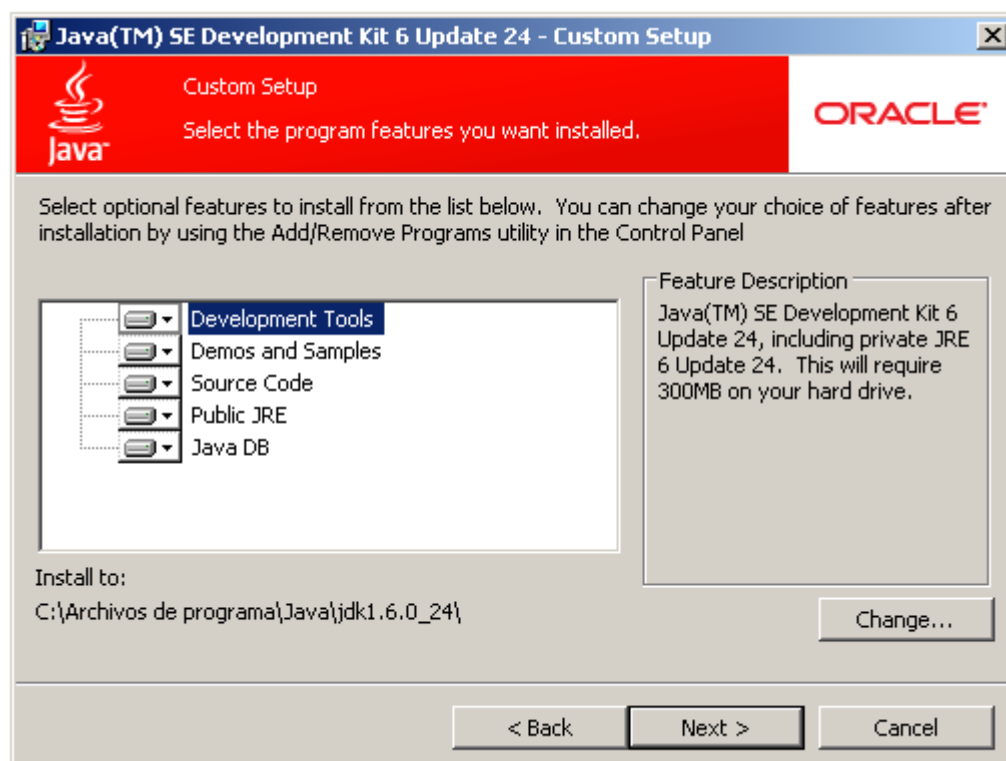
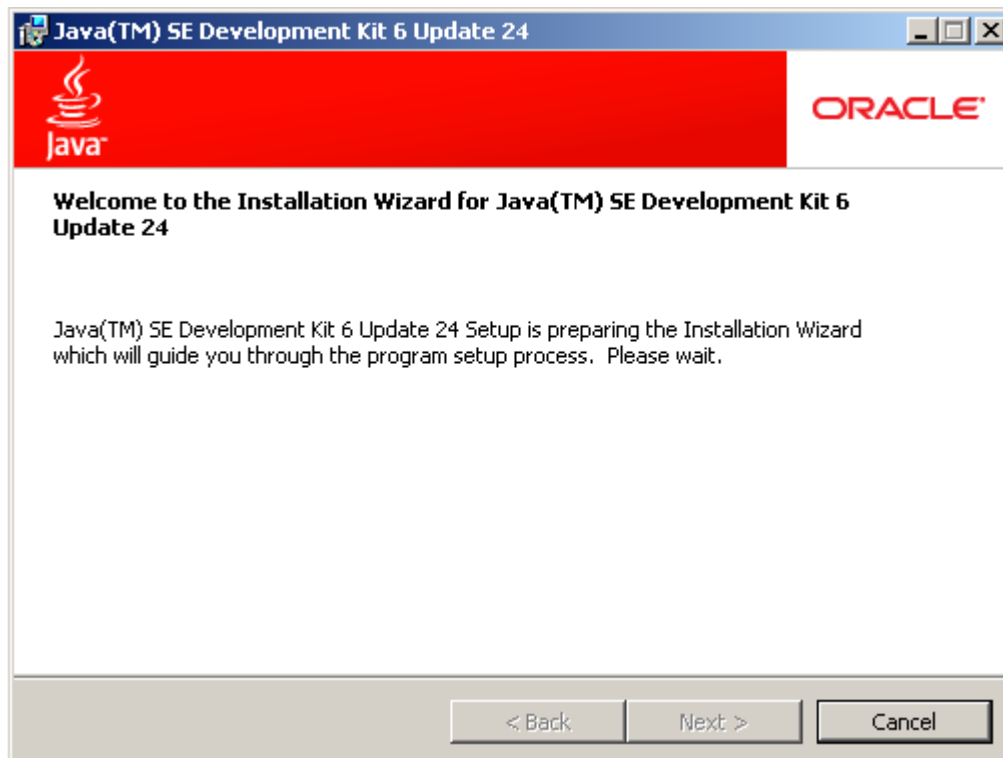
Continue »

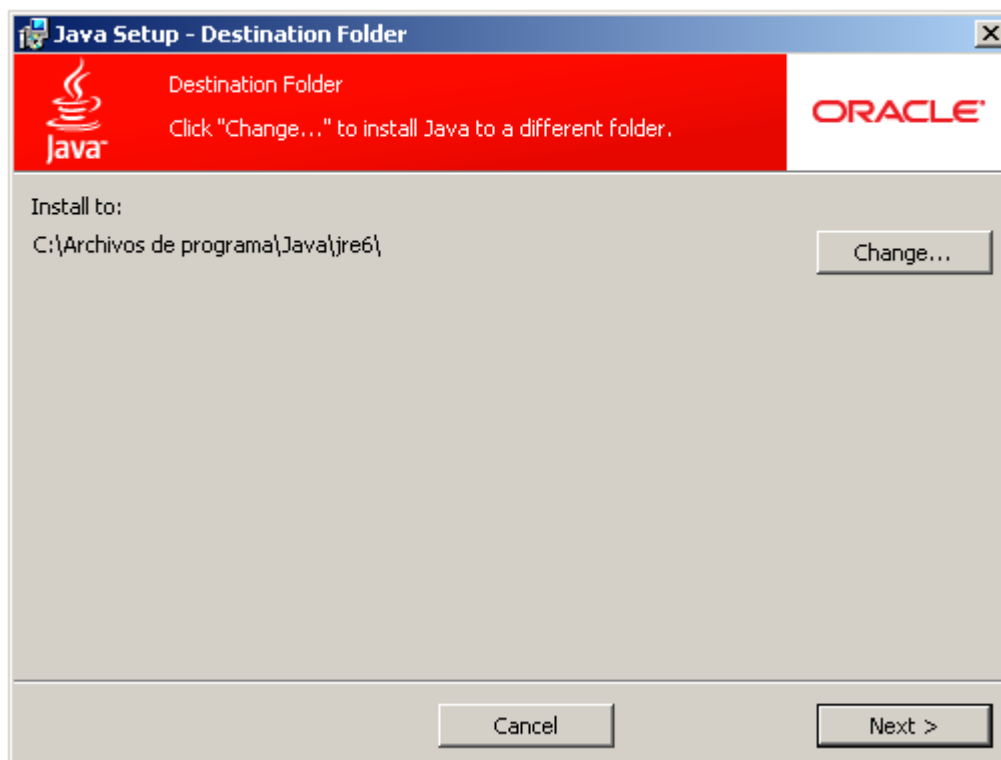
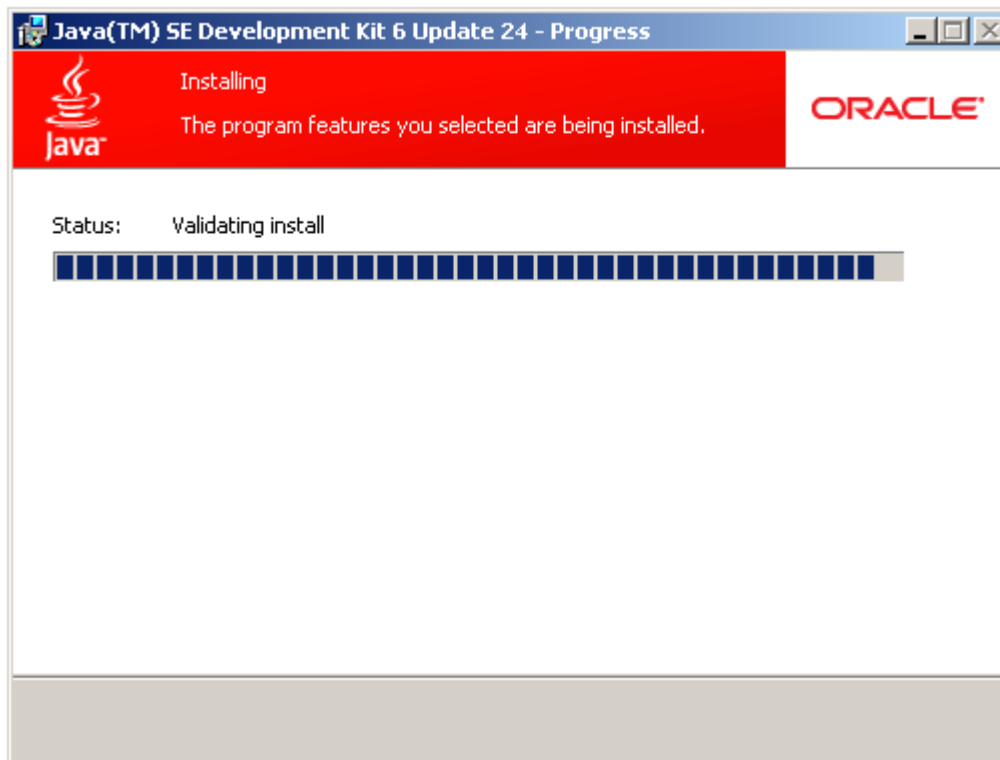
Una vez descargado el archivo lo ejecutamos



jdk-6u24-windows-i586









Aquí se termina de instalar el JDK.

Instalación de mysql-server

Descargamos el manejador de base de datos mysql-5.5.11 (6u24 (la liga de descarga para el archivo se encuentra en la parte final del documento)).

Una vez descargado el archivo lo ejecutamos



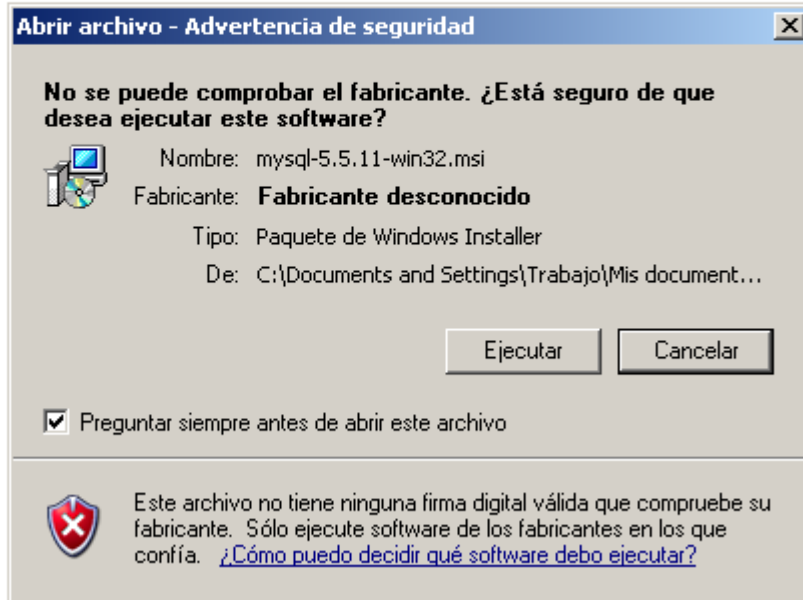
jdk-6u24-windows-i586



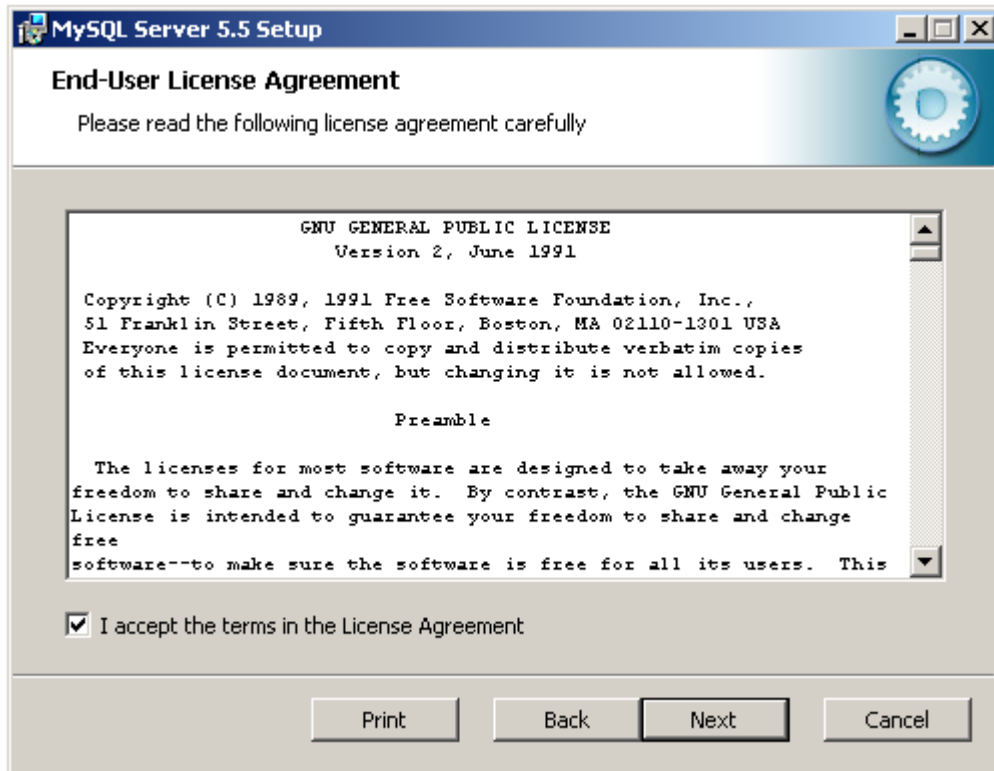
mysql-5.5.11-win32
Paquete de Windows Installer
28.318 KB



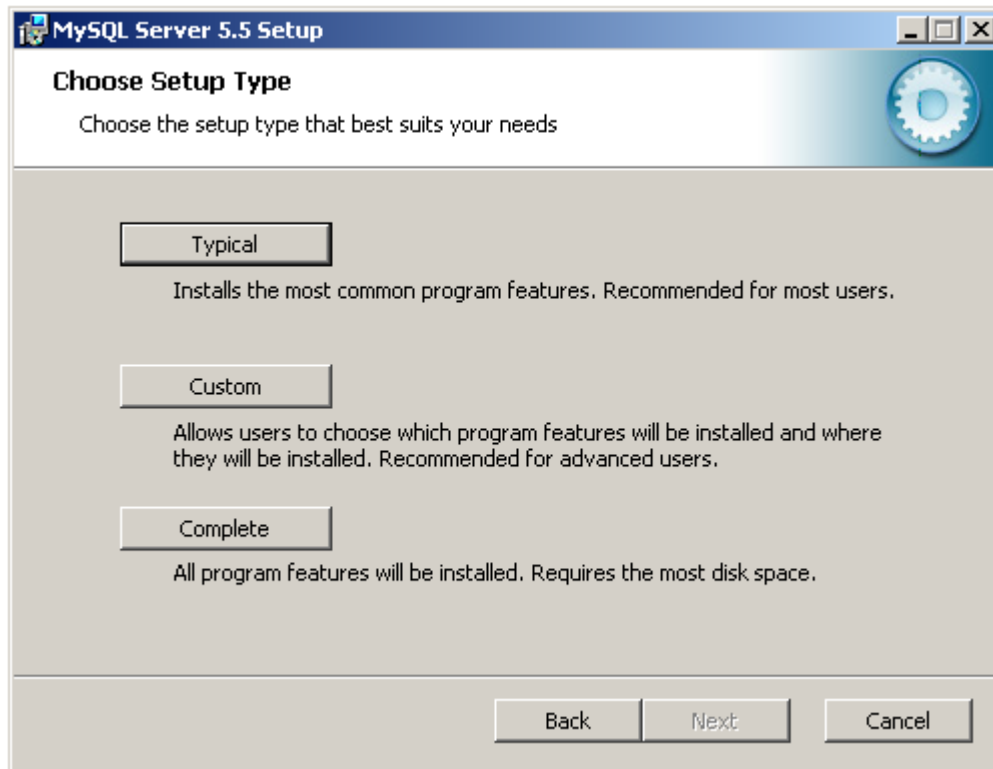
my
Pa
25



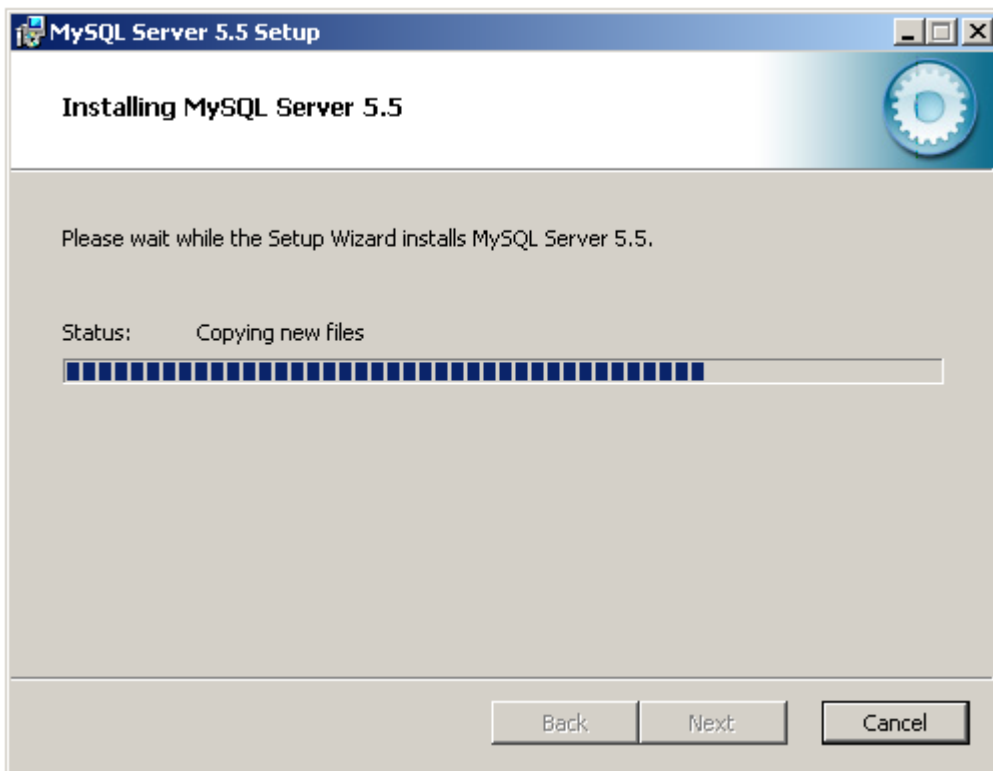
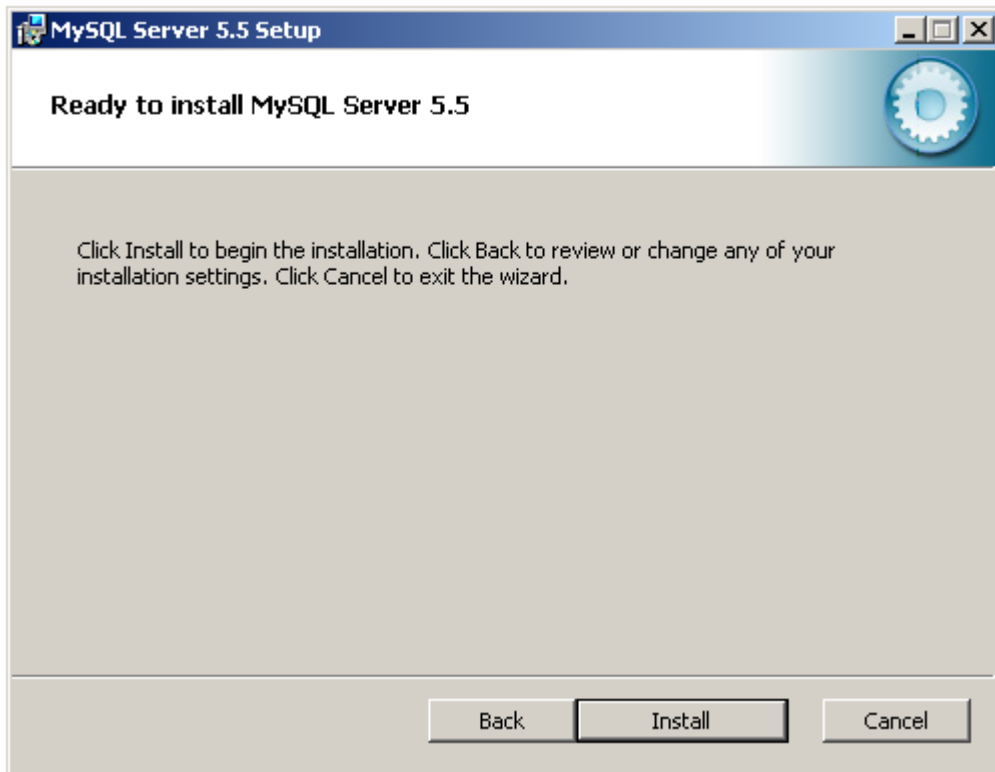
Aceptamos los términos

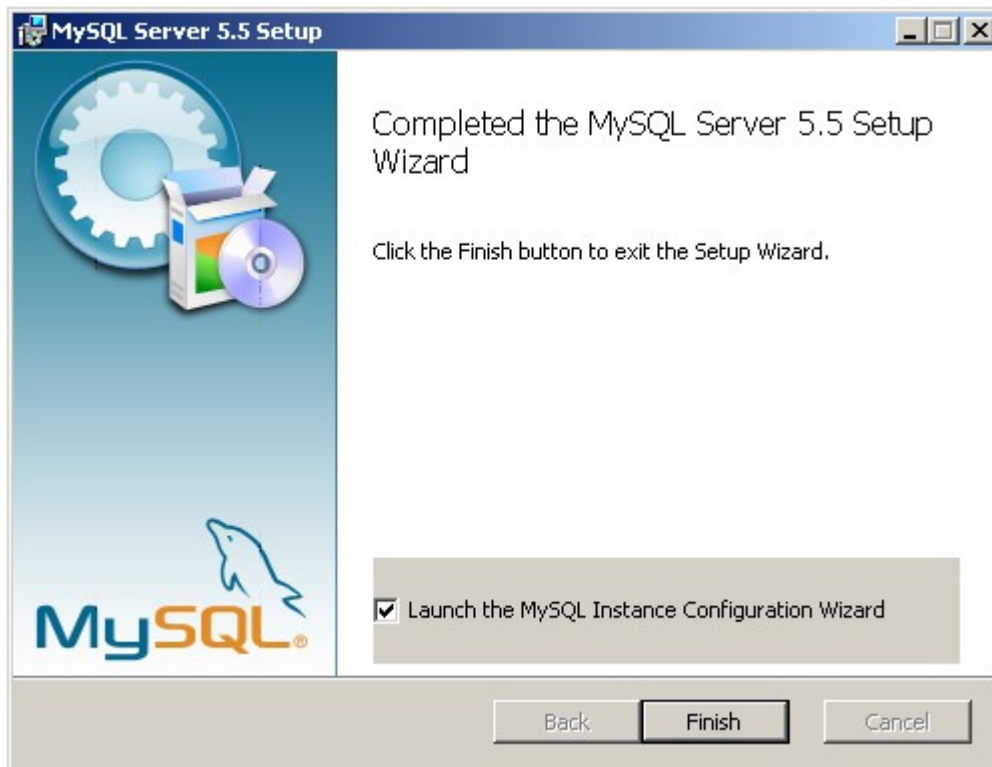
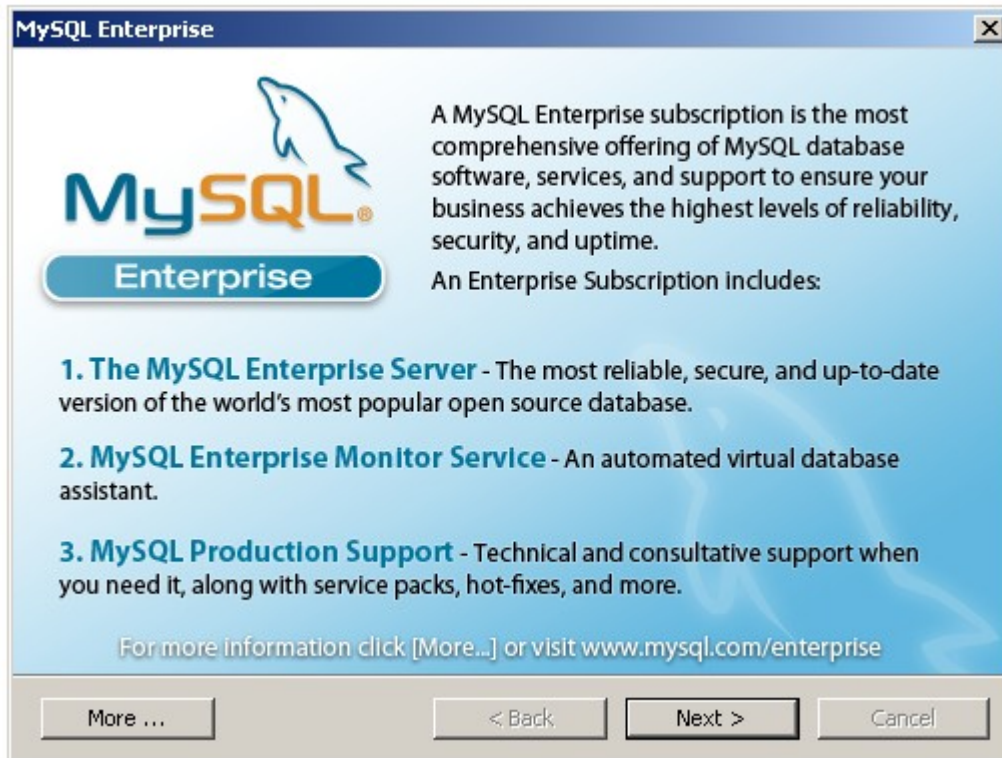


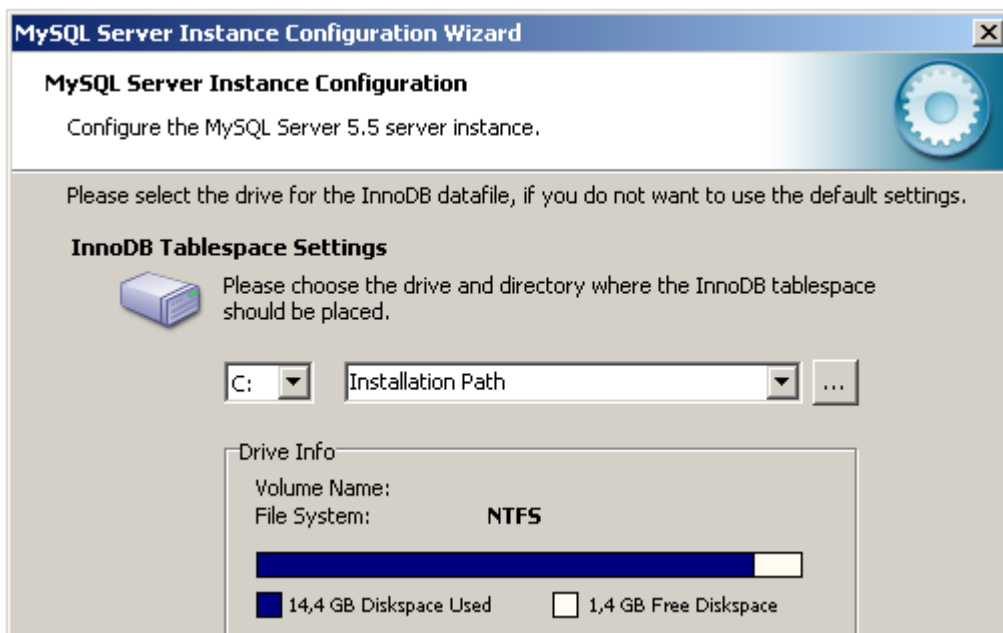
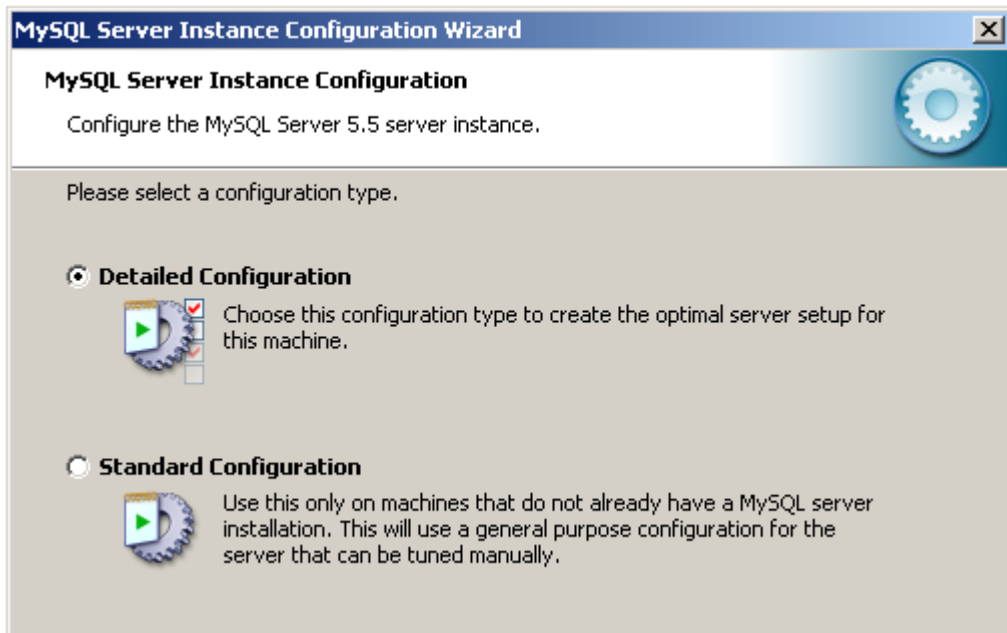
Seleccionamos la instalación completa



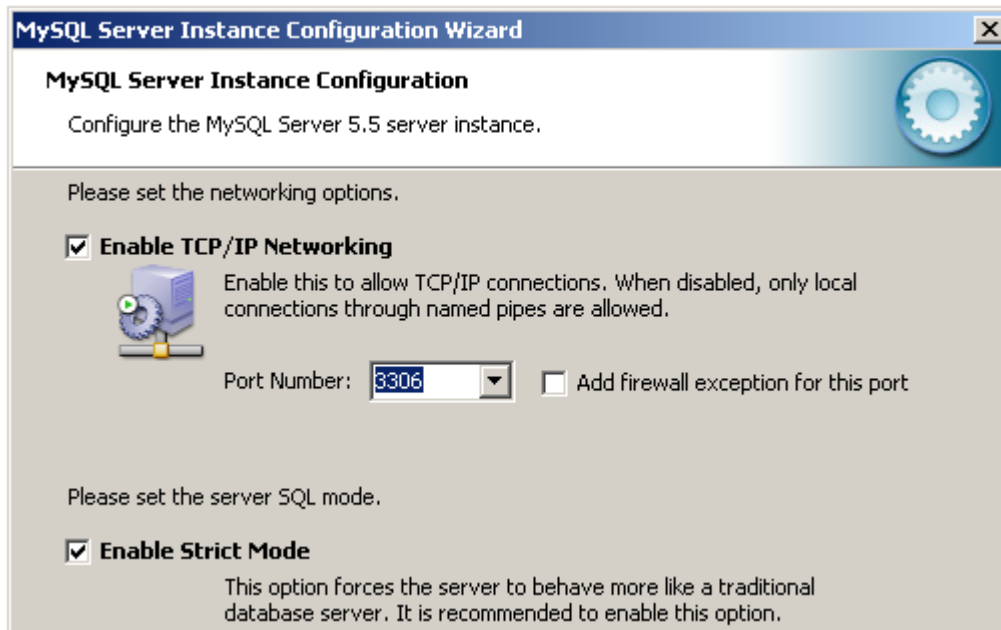
Seleccionamos la opción de instalar



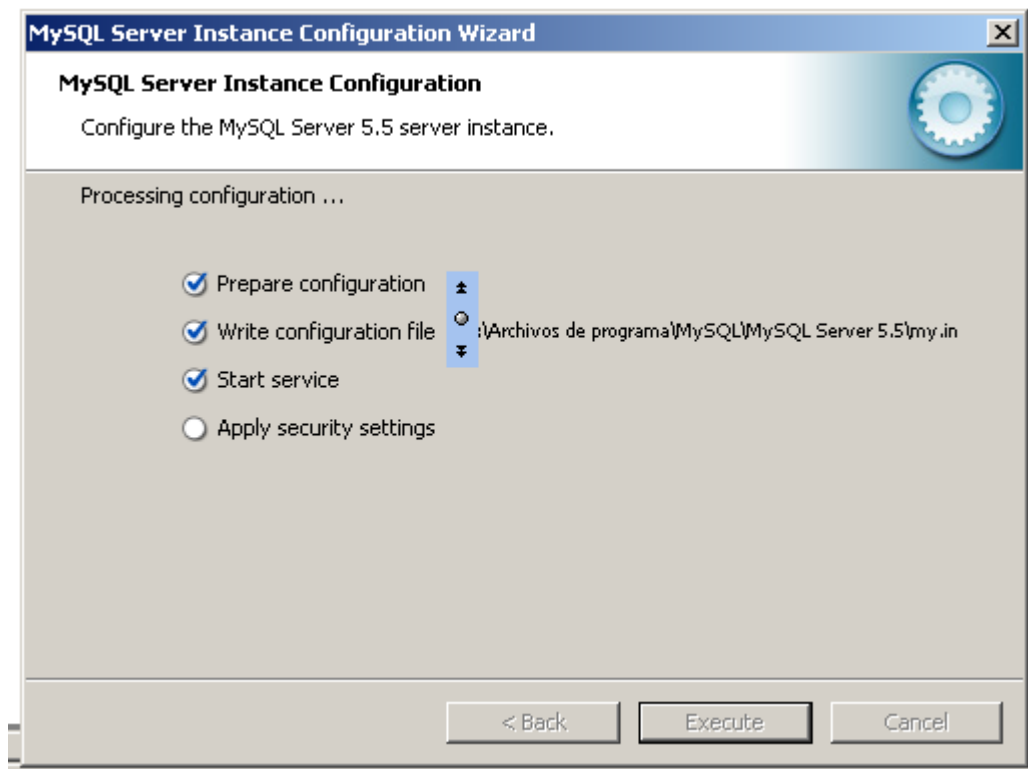
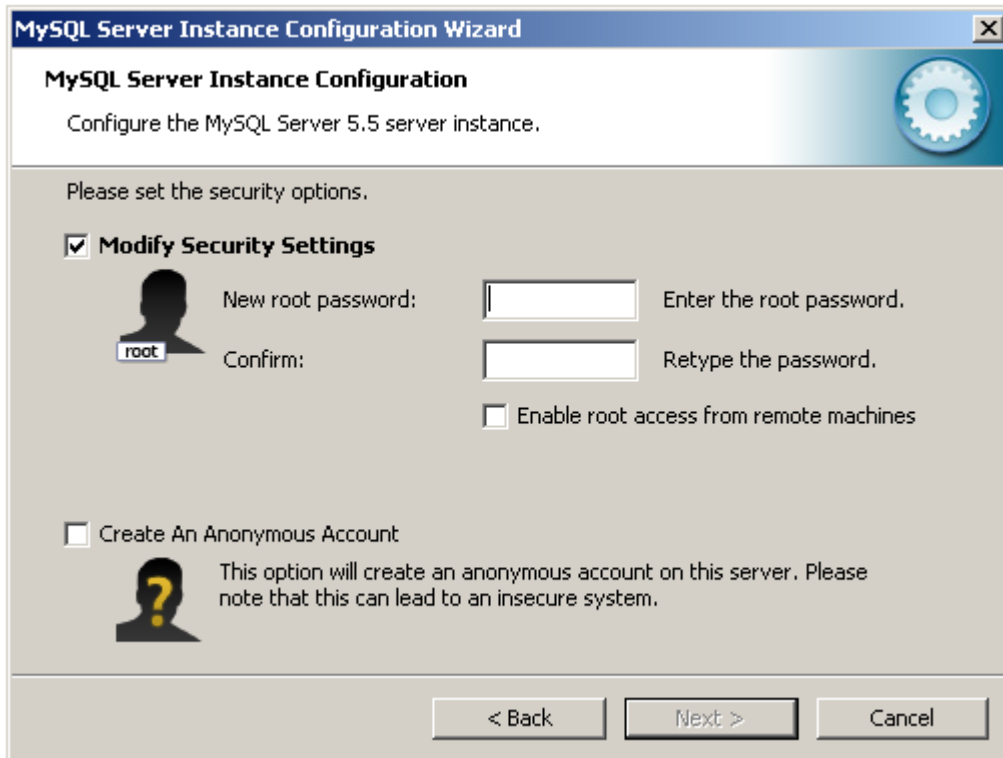




En esta parte seleccionamos el puerto.



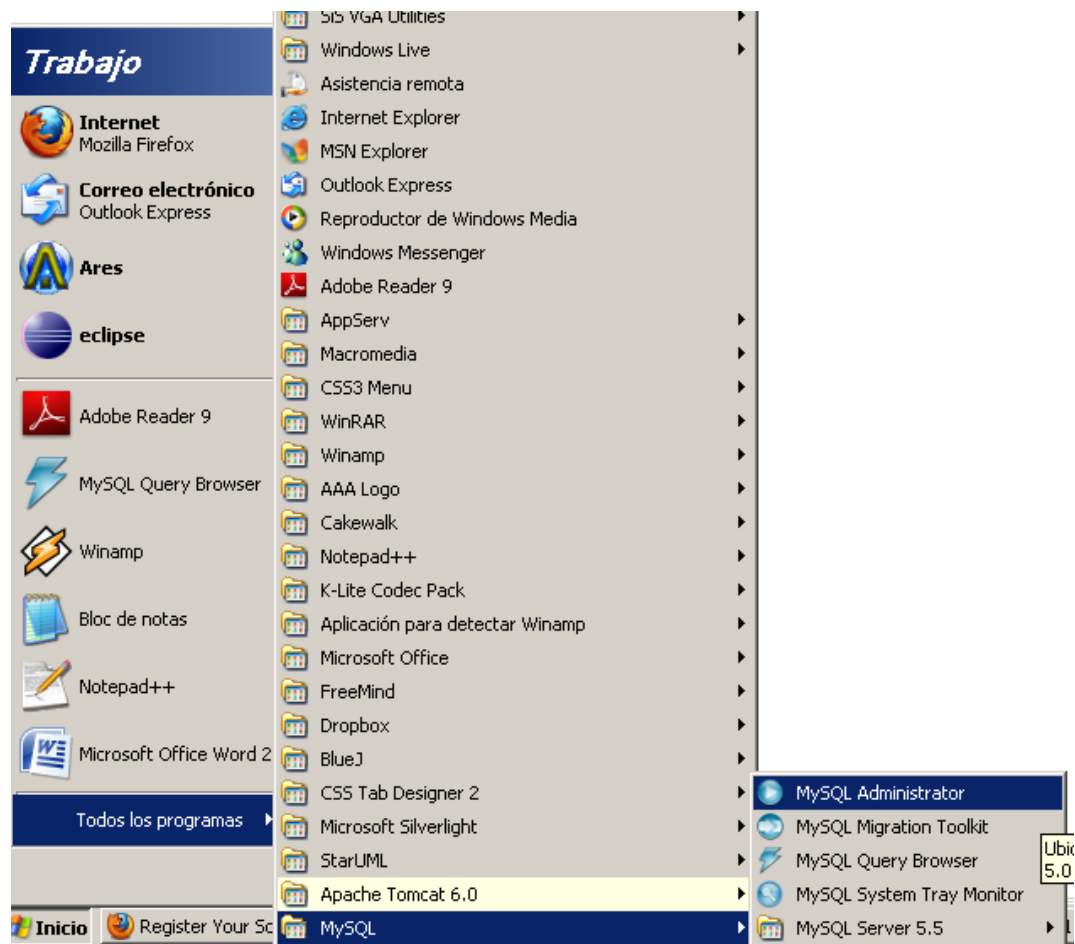
Introducimos la contraseña para el administrador



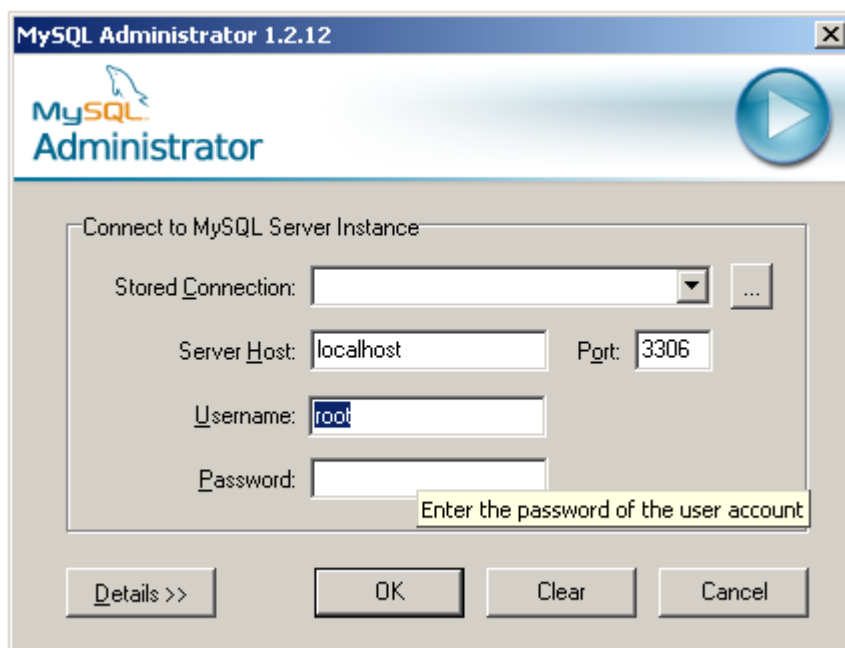
Aquí se termina de instalar el manejador de base de datos mysql .

Restauración de la base de datos

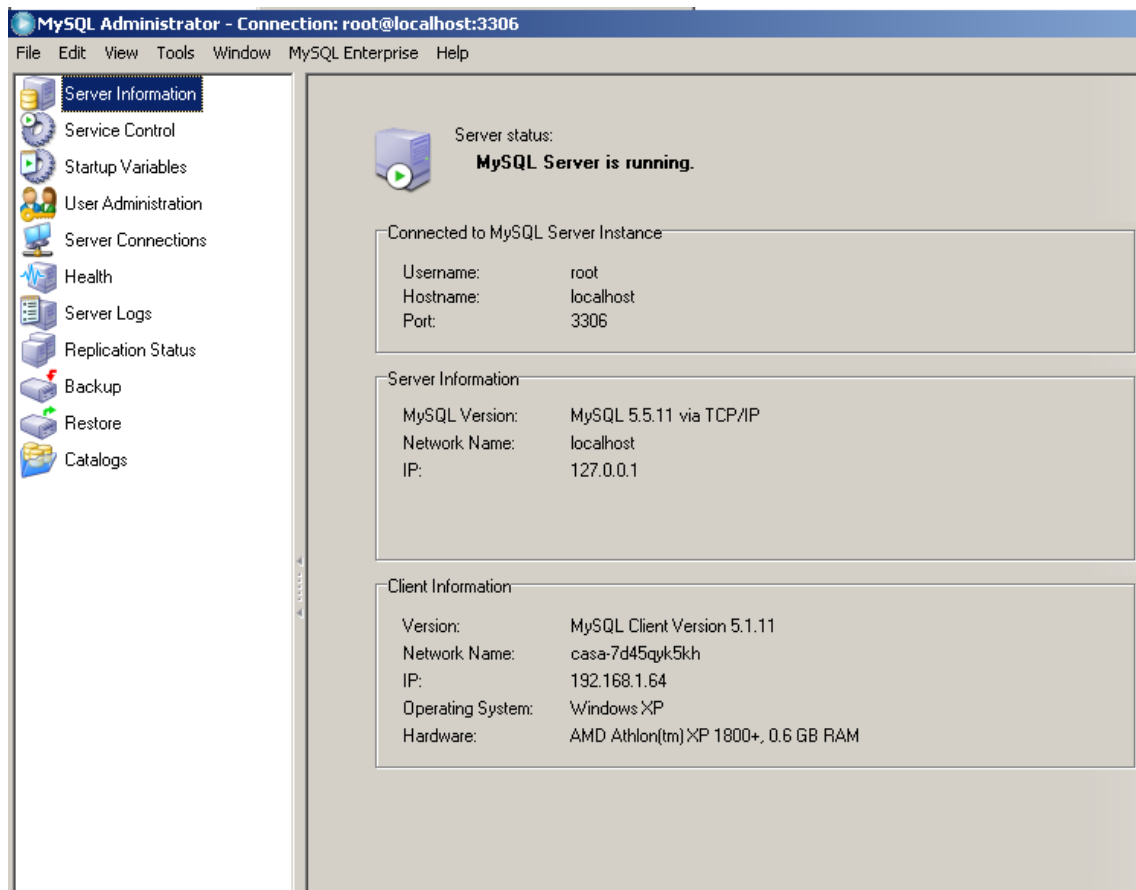
Una vez instalado mysql-server ejecutamos el MySQL-Administrator



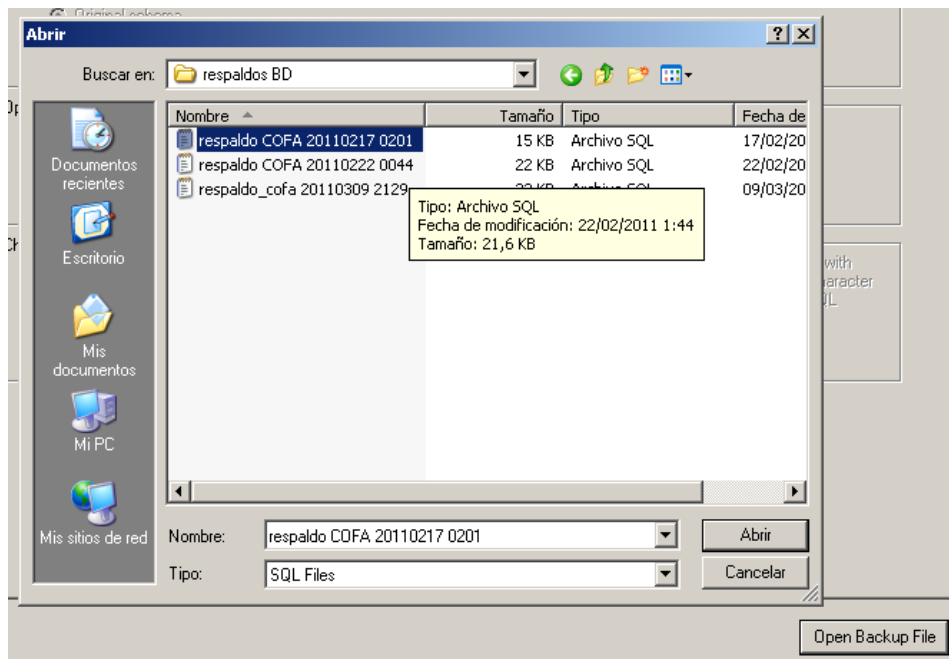
Introducimos el password ingresado al momento de la instalación.



Seleccionamos la opción Restore



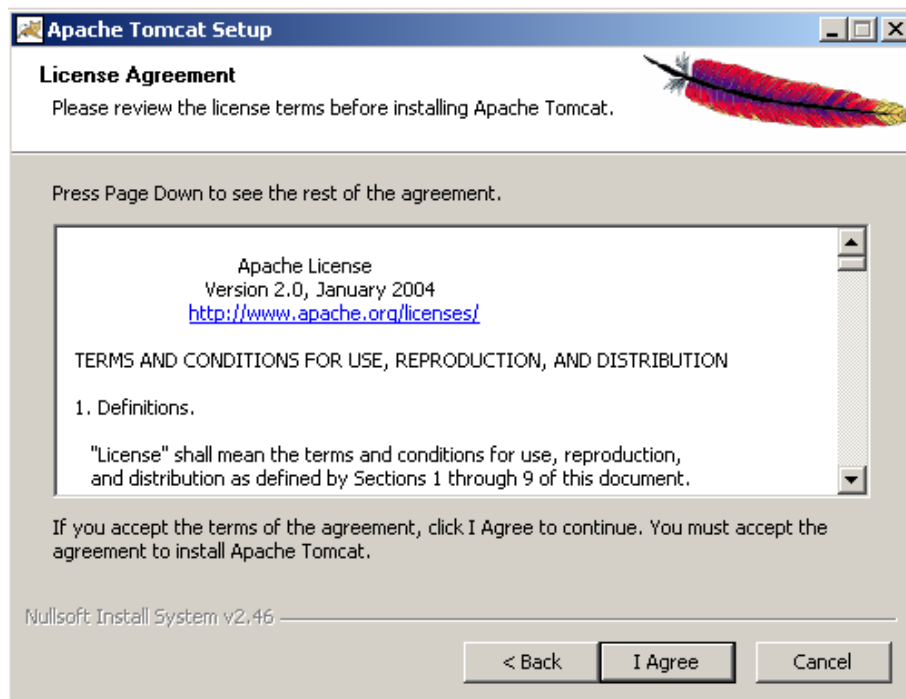
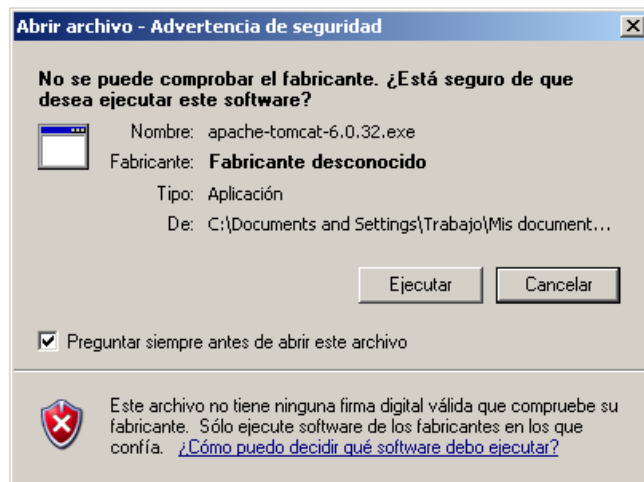
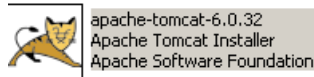
Abrimos el archivo SQL que contiene la declaración de la base de datos.



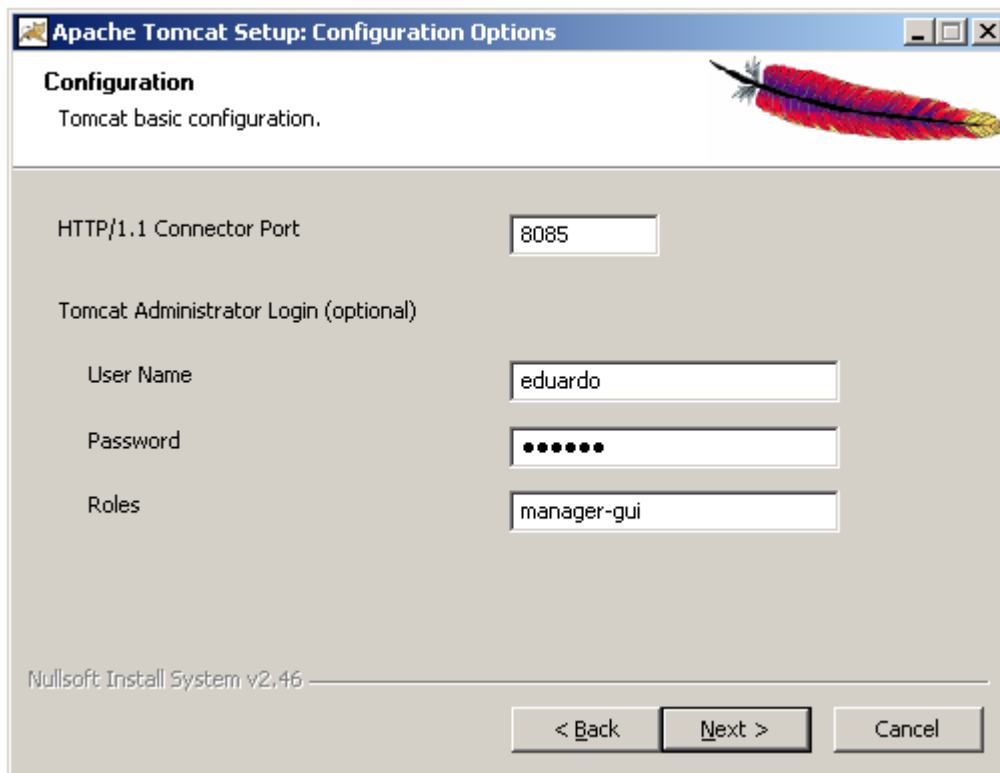
Después elegimos la opción start backup. Aquí terminan los pasos para la creación de la base de datos

Instalación servidor tomcat

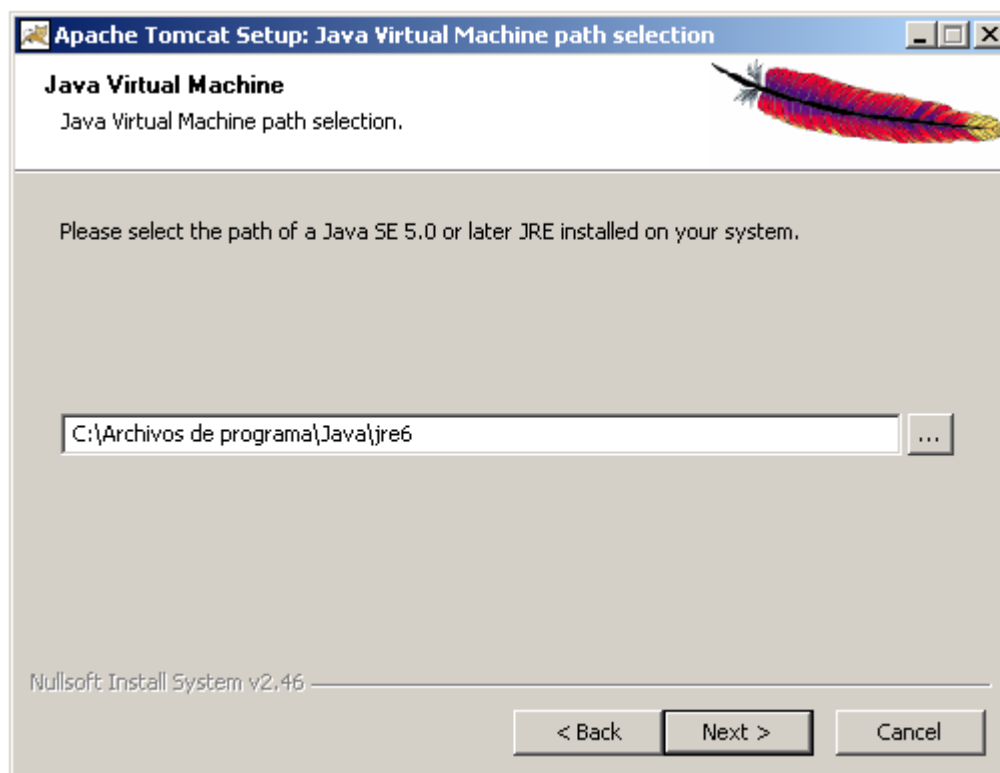
Descargamos el servidor de aplicaciones tomcat-6.0. Una vez descargado lo ejecutamos



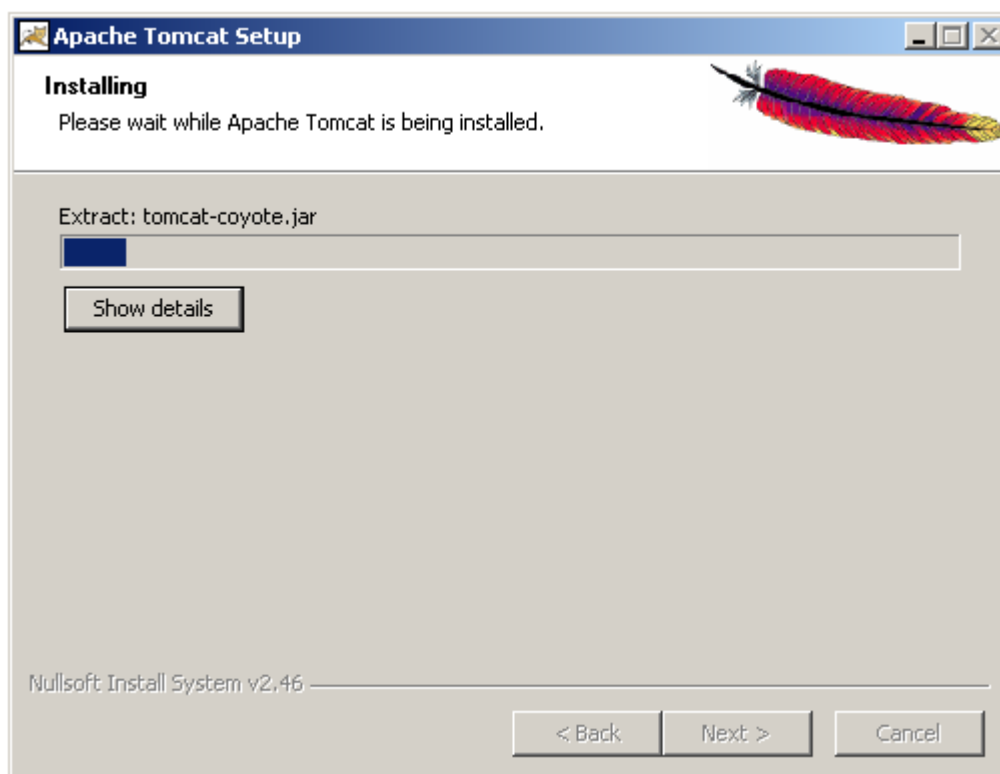
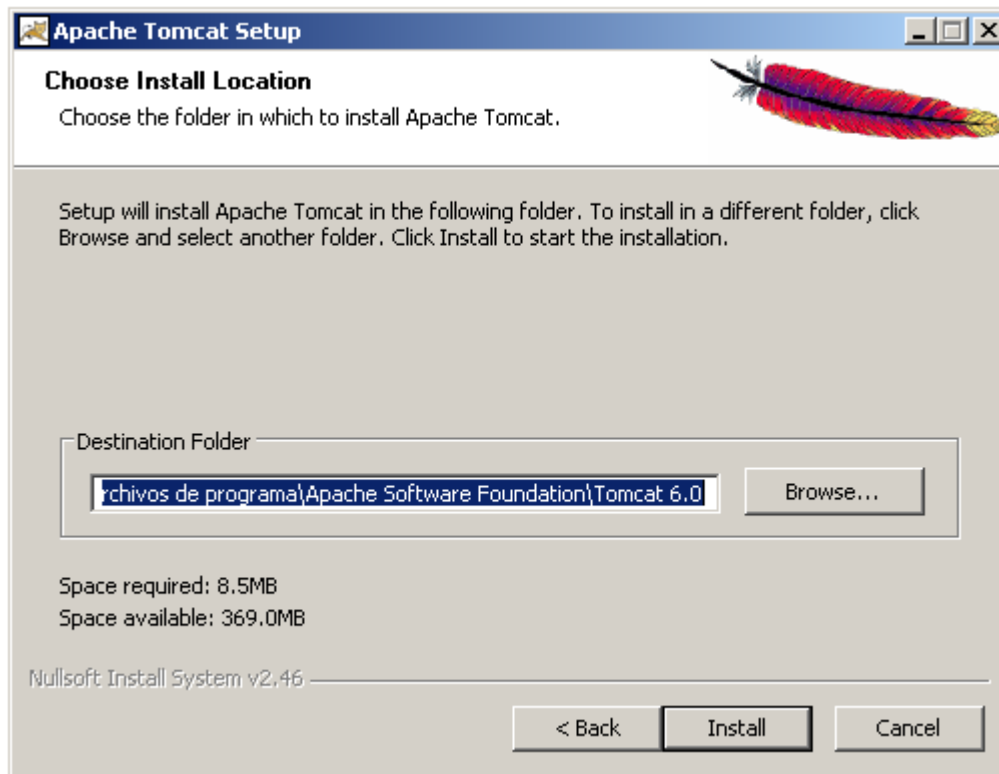
Aquí introducimos el nombre de usuario y password de administrador.

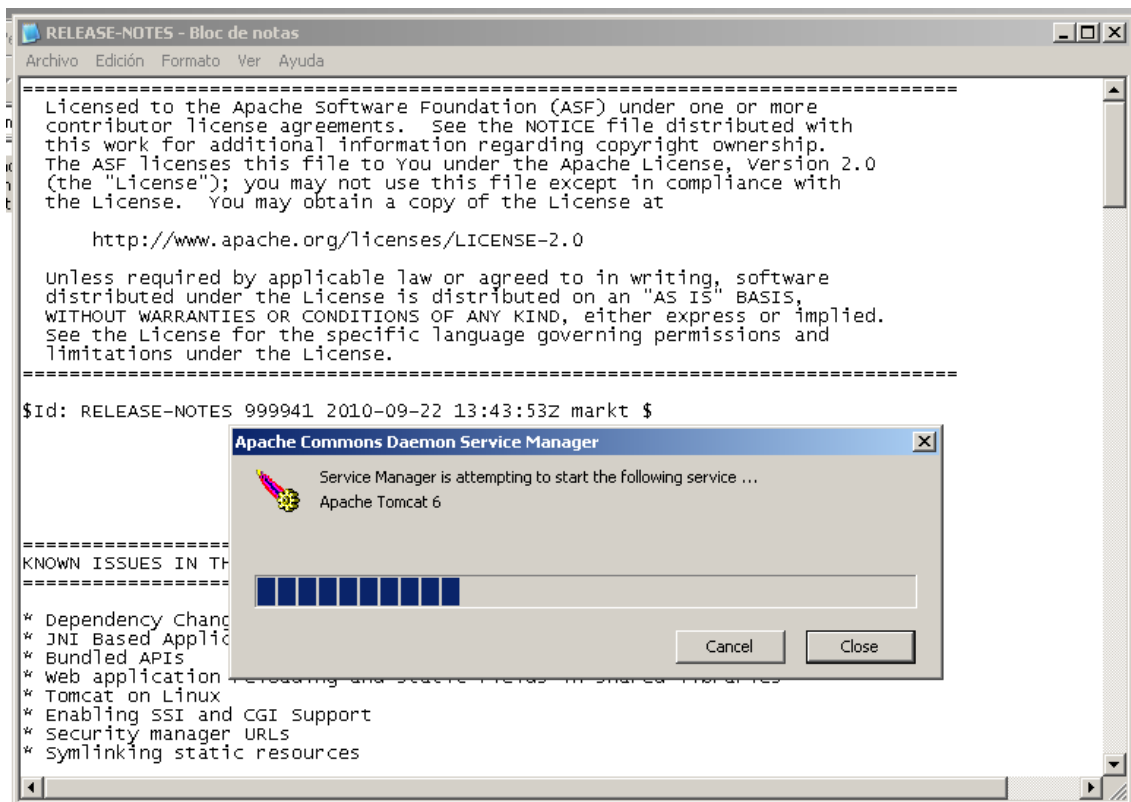
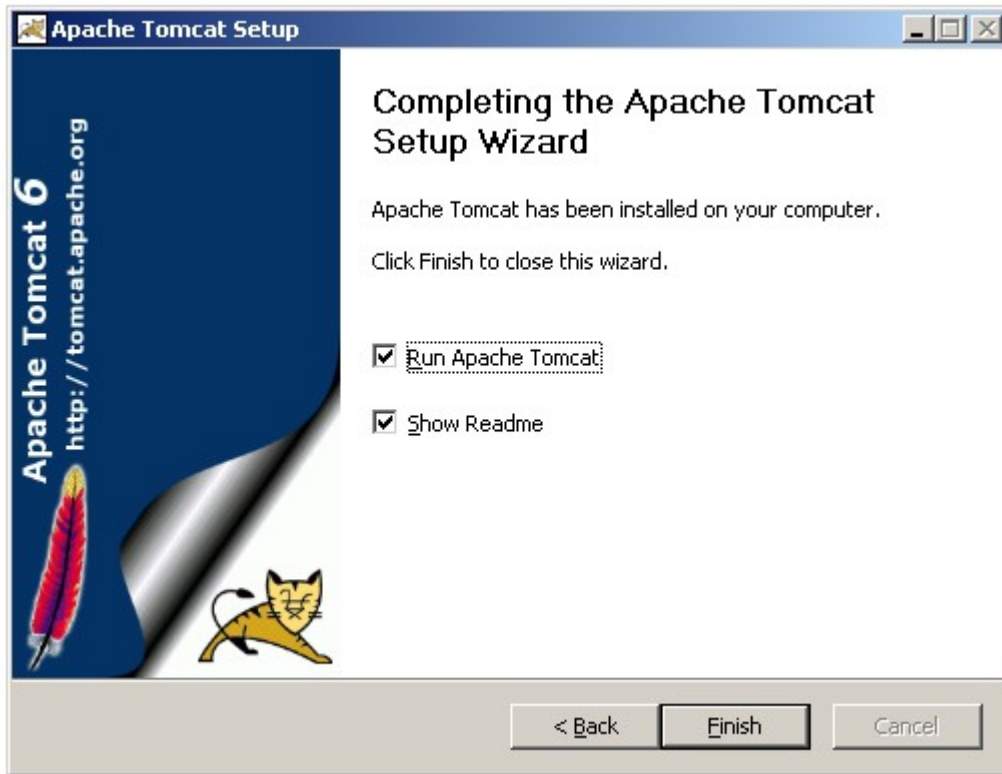


Aquí seleccionamos la tura del JDK que instalamos en la primera parte



Seleccionamos la ruta de instalación de tomcat



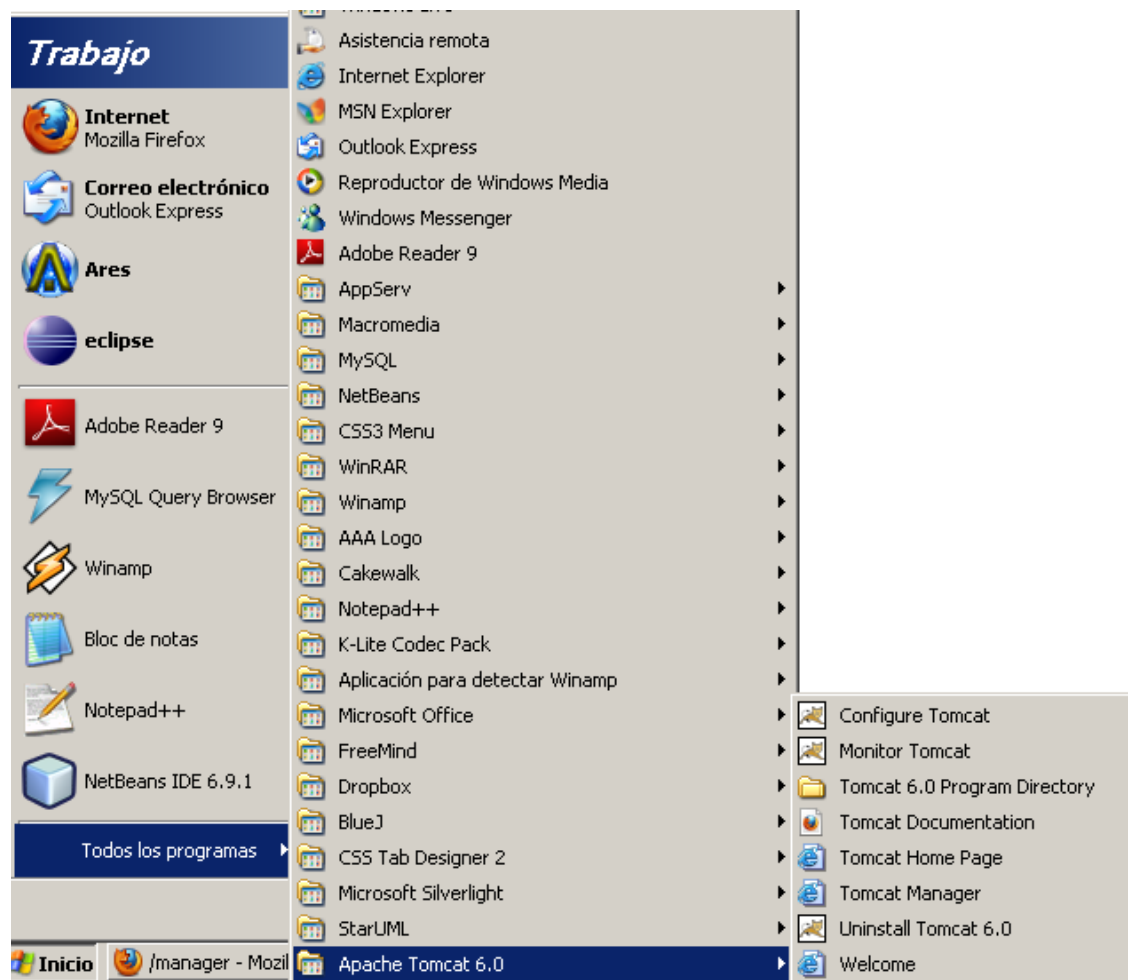


En esta parte tomcat se termina de instalar.

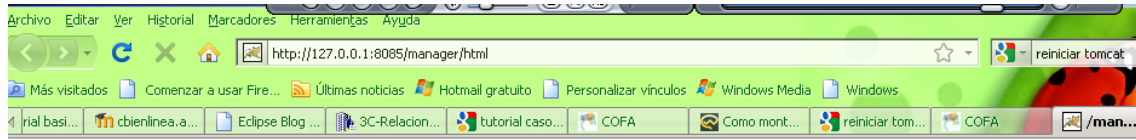
Montar la aplicación web

En la ruta donde se instaló tomcat entramos a la carpeta webapps y pegamos dentro de ella el archivo COFA.war.

Abrimos el tomcat manager



Una vez abierto el gestor de aplicaciones de tomcat, vemos las aplicaciones que están actualmente en el servidor, en donde está nuestra aplicación (COFA) seleccionamos la opción de arrancar.



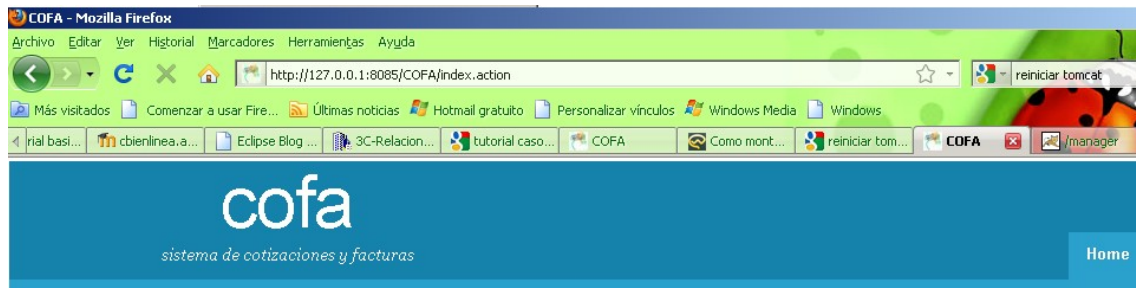
Gestor de Aplicaciones Web de Tomcat

Mensaje: OK

Gestor
[Listar Aplicaciones](#) [Ayuda HTML de Gestor](#) [Ayuda de Gestor](#) [Estado de S](#)

Aplicaciones				
Trayectoria	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Welcome to Tomcat	true	0	Arrancar Parar Recargar Replegar <input type="button" value="Expirar sesiones"/> sin trabajar ≥ <input type="text" value="30"/> minutos
/COFA	Aplicación	true	1	Arrancar Parar Recargar Replegar <input type="button" value="Expirar sesiones"/> sin trabajar ≥ <input type="text" value="30"/> minutos
/docs	Tomcat Documentation	true	0	Arrancar Parar Recargar Replegar <input type="button" value="Expirar sesiones"/> sin trabajar ≥ <input type="text" value="30"/> minutos

Para acceder a la aplicación introducimos el siguiente URL en el navegador y listo .



Bienvenido a cofa

Usuario
 Password

Ligas de descarga

- TOMCAT - <http://tomcat.apache.org/download-60.cgi>
- MySQL-server - <http://www.mysql.com/downloads/mirror.php?id=401954>
- JDK - https://cds.sun.com/is-bin/INTERSHOP.enfinity/WFS/CDS-CDS_Developer-Site/en_US/-/USD/ViewProductDetail-Start?ProductRef=jdk-6u24-oth-JPR@CDS-CDS_Developer

Ligas consultadas el 12/04/2011

Conclusiones

El crecimiento constante de la población que accede a Internet para la consulta o consumo de bienes o servicios, obliga a que cualquier empresa considere tener un punto de venta en Internet, además debe de estar en funcionamiento para los clientes en el menor tiempo posible, el proyecto desarrollado presenta una base de un sistema de cotizaciones y facturación electrónica. Debido a que cada negocio tiene necesidades particulares el proyecto puede estar en continuo crecimiento y adaptándose a las necesidades del negocio.

Los frameworks utilizados para el desarrollo de este proyecto permiten un alto grado de escalabilidad y mantenibilidad, además de estar desarrollado en su totalidad con software libre.

A lo largo del desarrollo de este proyecto me di cuenta que ningún sistema estará completo, siempre debe de estar en constante desarrollo y observación, pero algo que se debe de tomar en cuenta al desarrollar un sistema es proporcionar una base sólida para los desarrolladores que contribuyan en el diseño e implementación de un sistema.

Licenciamiento Creative Commons

Este trabajo cuenta con una licencia Creative Commons, los términos de la licencia se pueden consultar en:

<http://creativecommons.org/licenses/by-sa/3.0/>

Bibliografía

- <http://www.manageability.org/blog/stuff/open-source-crm>
Fecha de consulta 28/02/2010
- <http://www.hipergate.org/> Fecha de consulta 18/02/2010.

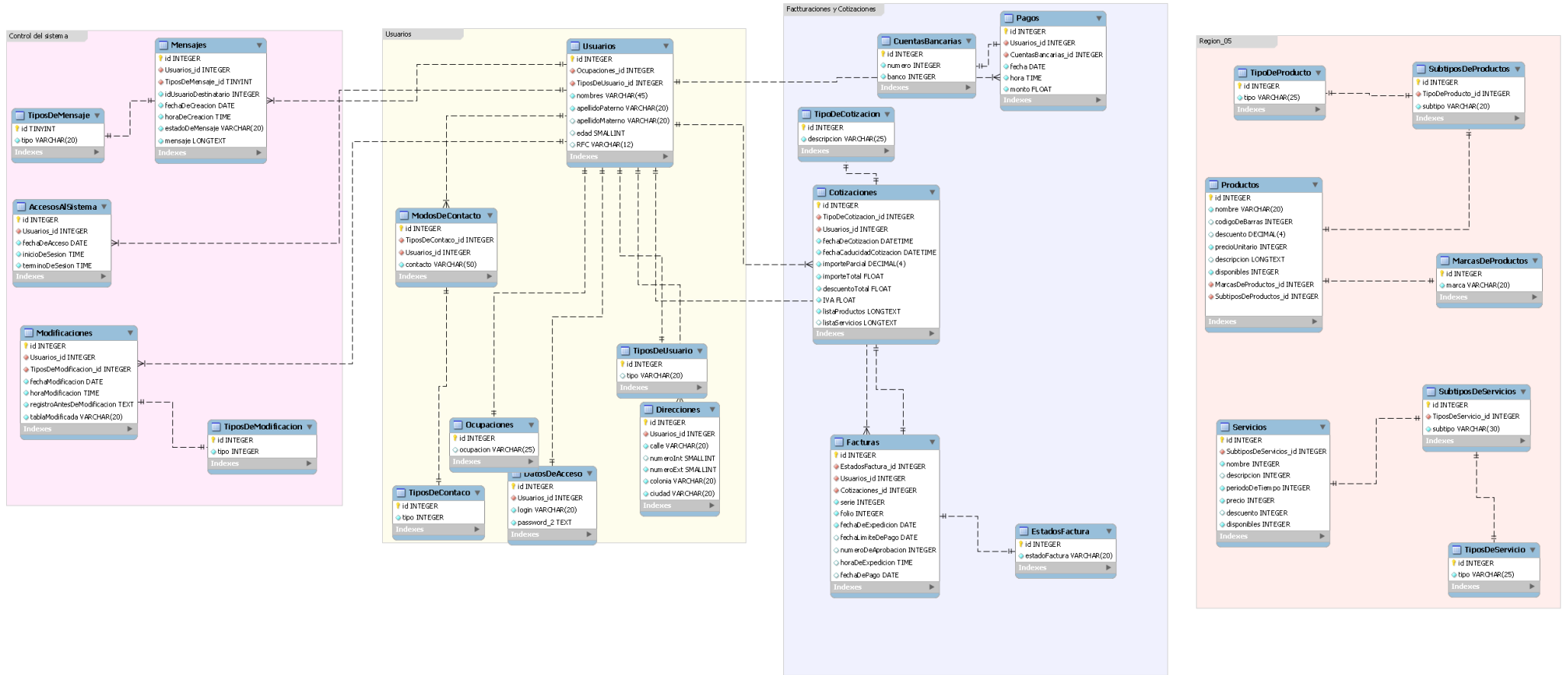
- <http://www.ohioedge.com/> Fecha de consulta 18/02/2010.
- <http://ofbiz.apache.org/> Fecha de consulta 28/02/2010

Documentación diseño lógico del sistema

Diseño de la base de datos	3
Diagramas de la base de datos	3
Diagrama entidad relación.....	3
Diagrama modelo relacional.....	4
Declaración de la base de datos	5
Diccionario de datos	13
Tabla marcasDeProductos	13
Tabla datosDeAcceso	13
Tabla productos	13
Tabla tiposDeCotizacion.....	14
Tabla cotizaciones.....	14
Tabla direcciones	15
Tabla productosEnCotizacion.....	16
Tabla tipoDeProducto	16
Tabla tipoDeUsuario	16
Tabla estadosFactura	16
Tabla modosDeContacto.....	17
Tabla tiposDeContacto.....	17
Tabla usuarios	17
Tabla facturas.....	18
Tabla Ocupaciones.....	18
Tabla subtiposDeProductos	19
Diagramas de clases	20
Modelo de casos de uso	22
Diagrama de caso de uso para validación de usuarios	22
Diagrama de casos de uso de la gestión de productos.....	22
Diagrama de casos de uso de la gestión de cotizaciones.....	23
Diagrama de casos de uso de la gestión de facturas	23
Diagrama de casos de uso de la gestión de usuarios.....	24
Documentación de actores	24
Documentación de casos de uso.....	25
Validar usuario	25
Caso de uso registro de nuevo usuario.....	27

Caso de uso agregar producto	30
Caso de uso modificar producto	32
Caso de uso borrar producto	35
Caso de uso generar cotización	36
Caso de uso cerrar cotización	39
Caso de uso cancelar cotización.....	41
Caso de uso ver detalle de cotización	43
Caso de uso generar facturas.....	44
Caso de uso consultar facturas	46
Caso de uso descargar factura	47
Diagramas de navegación	48
Diagrama de navegación para la ruta de facturas	48
Diagrama de navegación del cliente	49
Diagrama de navegación general.....	49
Diagrama de navegación para productos-servicios	50

Diagrama modelo relacional



Declaración de la base de datos

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL';
```

```
CREATE SCHEMA IF NOT EXISTS `mydb` ;
USE `mydb` ;
```

```
-----
-- Table `mydb`.`TiposDeUsuario`
-----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`TiposDeUsuario` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `tipo` VARCHAR(20) NULL ,
  PRIMARY KEY (`id` )
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;
```

```
-----
-- Table `mydb`.`Ocupaciones`
-----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Ocupaciones` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `ocupacion` VARCHAR(25) NULL ,
  PRIMARY KEY (`id` )
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;
```

```
-----
-- Table `mydb`.`Usuarios`
-----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Usuarios` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `Ocupaciones_id` INT(11) NOT NULL ,
  `TiposDeUsuario_id` INT(11) NOT NULL ,
  `nombres` VARCHAR(45) NOT NULL ,
  `apellidoPaterno` VARCHAR(20) NOT NULL ,
  `apellidoMaterno` VARCHAR(20) NULL ,
  `edad` SMALLINT UNSIGNED NULL ,
  `RFC` VARCHAR(12) NULL ,
  PRIMARY KEY (`id` ) ,
  INDEX `Usuarios_FKIndex1` (`TiposDeUsuario_id` ASC) ,
  INDEX `Usuarios_FKIndex2` (`Ocupaciones_id` ASC) ,
  CONSTRAINT `fk_{CBD9AF9A-FFF4-45B2-BD84-CF87E1DE8637}`
    FOREIGN KEY (`TiposDeUsuario_id` )
    REFERENCES `mydb`.`TiposDeUsuario` (`id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_{FB09B6E2-5CE1-4049-AA2E-0D5C3935835A}`
    FOREIGN KEY (`Ocupaciones_id` )
    REFERENCES `mydb`.`Ocupaciones` (`id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
```

```
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;
```

```
-----
-- Table `mydb`.`MarcasDeProductos`
-----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`MarcasDeProductos` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `marca` VARCHAR(20) NOT NULL ,
  PRIMARY KEY (`id`) )
```

```
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;
```

```
-----
-- Table `mydb`.`TipoDeProducto`
-----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`TipoDeProducto` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `tipo` VARCHAR(25) NOT NULL ,
  PRIMARY KEY (`id`) )
```

```
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;
```

```
-----
-- Table `mydb`.`SubtiposDeProductos`
-----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`SubtiposDeProductos` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `TipoDeProducto_id` INT(11) NOT NULL ,
  `subtipo` VARCHAR(20) NOT NULL ,
  PRIMARY KEY (`id`) ,
  INDEX `SubtiposDeProductos_FKIndex1` (`TipoDeProducto_id` ASC) ,
  CONSTRAINT `fk_{F2A11D02-AA35-4398-9D22-A1A9C3541475}`
    FOREIGN KEY (`TipoDeProducto_id`)
    REFERENCES `mydb`.`TipoDeProducto` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
```

```
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;
```

```
-----
-- Table `mydb`.`Productos`
-----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Productos` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `nombre` VARCHAR(20) NOT NULL ,
  `codigoDeBarras` INT(11) NULL ,
  `descuento` DECIMAL(4) NULL ,
  `precioUnitario` INT(11) NOT NULL ,
  `descripcion` LONGTEXT NULL ,
  `disponibles` INT(11) NOT NULL ,
  `MarcasDeProductos_id` INT(11) NOT NULL ,
  `SubtiposDeProductos_id` INT(11) NOT NULL ,
  PRIMARY KEY (`id`) ,
  INDEX `fk_Productos_MarcasDeProductos1` (`MarcasDeProductos_id` ASC)
,
```

```

INDEX `fk_Productos_SubtiposDeProductos1` (`SubtiposDeProductos_id`
ASC) ,
CONSTRAINT `fk_Productos_MarcasDeProductos1`
  FOREIGN KEY (`MarcasDeProductos_id` )
  REFERENCES `mydb`.`MarcasDeProductos` (`id` )
  ON DELETE NO ACTION
  ON UPDATE NO ACTION,
CONSTRAINT `fk_Productos_SubtiposDeProductos1`
  FOREIGN KEY (`SubtiposDeProductos_id` )
  REFERENCES `mydb`.`SubtiposDeProductos` (`id` )
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

```

```

-----
-- Table `mydb`.`TiposDeServicio`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`TiposDeServicio` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `tipo` VARCHAR(25) NOT NULL ,
  PRIMARY KEY (`id` )
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

```

```

-----
-- Table `mydb`.`SubtiposDeServicios`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`SubtiposDeServicios` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `TiposDeServicio_id` INT(11) NOT NULL ,
  `subtipo` VARCHAR(30) NOT NULL ,
  PRIMARY KEY (`id` ) ,
  INDEX `SubtiposDeServicios_FKIndex1` (`TiposDeServicio_id` ASC) ,
  CONSTRAINT `fk_{505DABBB-8782-4326-B0EA-A7E7938797AE}`
    FOREIGN KEY (`TiposDeServicio_id` )
    REFERENCES `mydb`.`TiposDeServicio` (`id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

```

```

-----
-- Table `mydb`.`Servicios`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`Servicios` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `SubtiposDeServicios_id` INT(11) NOT NULL ,
  `nombre` INT(11) NOT NULL ,
  `descripcion` INT(11) NULL ,
  `periodoDeTiempo` INT(11) NOT NULL ,
  `precio` INT(11) NOT NULL ,
  `descuento` INT(11) NULL ,
  `disponibles` INT(11) NOT NULL ,
  PRIMARY KEY (`id` ) ,
  INDEX `Servicios_FKIndex1` (`SubtiposDeServicios_id` ASC) ,
  CONSTRAINT `fk_{36A37B70-7801-49D5-BA5E-D0AD7DBEE34F}`
    FOREIGN KEY (`SubtiposDeServicios_id` )

```

```

REFERENCES `mydb`.`SubtiposDeServicios` (`id` )
ON DELETE NO ACTION
ON UPDATE NO ACTION)
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

-----
-- Table `mydb`.`TipoDeCotizacion`
-----
CREATE TABLE IF NOT EXISTS `mydb`.`TipoDeCotizacion` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `descripcion` VARCHAR(25) NOT NULL ,
  PRIMARY KEY (`id` )
)
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

-----
-- Table `mydb`.`Cotizaciones`
-----
CREATE TABLE IF NOT EXISTS `mydb`.`Cotizaciones` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `TipoDeCotizacion_id` INT(11) NOT NULL ,
  `Usuarios_id` INT(11) NOT NULL ,
  `fechaDeCotizacion` DATETIME NOT NULL ,
  `fechaCaducidadCotizacion` DATETIME NOT NULL ,
  `importeParcial` DECIMAL(4) NOT NULL ,
  `importeTotal` FLOAT NOT NULL ,
  `descuentoTotal` FLOAT NOT NULL ,
  `IVA` FLOAT NOT NULL ,
  `listaProductos` LONGTEXT NOT NULL ,
  `listaServicios` LONGTEXT NULL ,
  PRIMARY KEY (`id` ) ,
  INDEX `Cotizaciones_FKIndex1` (`Usuarios_id` ASC) ,
  INDEX `Cotizaciones_FKIndex2` (`TipoDeCotizacion_id` ASC) ,
  CONSTRAINT `fk_{617CD48E-864D-459A-A21B-53ECD0C662E2}`
    FOREIGN KEY (`Usuarios_id` )
    REFERENCES `mydb`.`Usuarios` (`id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_{70A01D0D-B31F-4AB4-BCFA-C1B33ECF6472}`
    FOREIGN KEY (`TipoDeCotizacion_id` )
    REFERENCES `mydb`.`TipoDeCotizacion` (`id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

-----
-- Table `mydb`.`EstadosFactura`
-----
CREATE TABLE IF NOT EXISTS `mydb`.`EstadosFactura` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `estadoFactura` VARCHAR(20) NOT NULL ,
  PRIMARY KEY (`id` )
)
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

```

-- Table `mydb`.`Facturas`

```
CREATE TABLE IF NOT EXISTS `mydb`.`Facturas` (  
  `id` INT(11) NOT NULL AUTO_INCREMENT ,  
  `EstadosFactura_id` INT(11) NOT NULL ,  
  `Usuarios_id` INT(11) NOT NULL ,  
  `Cotizaciones_id` INT(11) NOT NULL ,  
  `serie` INT(11) NOT NULL ,  
  `folio` INT(11) NOT NULL ,  
  `fechaDeExpedicion` DATE NOT NULL ,  
  `fechaLimiteDePago` DATE NULL ,  
  `numeroDeAprobacion` INT(11) NULL ,  
  `horaDeExpedicion` TIME NULL ,  
  `fechaDePago` DATE NULL ,  
  PRIMARY KEY (`id`) ,  
  INDEX `Facturas_FKIndex1` (`Cotizaciones_id` ASC) ,  
  INDEX `Facturas_FKIndex2` (`Usuarios_id` ASC) ,  
  INDEX `Facturas_FKIndex3` (`EstadosFactura_id` ASC) ,  
  CONSTRAINT `fk_{2753A132-B7B2-4B5F-A779-4A5C147BCE99}`  
    FOREIGN KEY (`Cotizaciones_id` )  
    REFERENCES `mydb`.`Cotizaciones` (`id` )  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_{D708C724-0B92-4BF3-980A-4BE49833A616}`  
    FOREIGN KEY (`Usuarios_id` )  
    REFERENCES `mydb`.`Usuarios` (`id` )  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_{B329873E-9394-4208-89BE-BA324C50AE76}`  
    FOREIGN KEY (`EstadosFactura_id` )  
    REFERENCES `mydb`.`EstadosFactura` (`id` )  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
PACK_KEYS = 0  
ROW_FORMAT = DEFAULT;
```

-- Table `mydb`.`TiposDeModificacion`

```
CREATE TABLE IF NOT EXISTS `mydb`.`TiposDeModificacion` (  
  `id` INT(11) NOT NULL AUTO_INCREMENT ,  
  `tipo` INT(11) NOT NULL ,  
  PRIMARY KEY (`id`) ,  
  UNIQUE INDEX `tipo_UNIQUE` (`tipo` ASC) )  
PACK_KEYS = 0  
ROW_FORMAT = DEFAULT;
```

-- Table `mydb`.`Modificaciones`

```
CREATE TABLE IF NOT EXISTS `mydb`.`Modificaciones` (  
  `id` INT(11) NOT NULL AUTO_INCREMENT ,  
  `Usuarios_id` INT(11) NOT NULL ,  
  `TiposDeModificacion_id` INT(11) NOT NULL ,  
  `fechaModificacion` DATE NOT NULL ,  
  `horaModificacion` TIME NOT NULL ,  
  `registroAntesDeModificacion` TEXT NOT NULL ,  
  `tablaModificada` VARCHAR(20) NOT NULL ,
```

```

PRIMARY KEY (`id`) ,
INDEX `Modificaciones_FKIndex1` (`TiposDeModificacion_id` ASC) ,
INDEX `Modificaciones_FKIndex2` (`Usuarios_id` ASC) ,
CONSTRAINT `fk_{547828AB-A94D-49A1-9572-ECD08B98D230}`
  FOREIGN KEY (`TiposDeModificacion_id` )
  REFERENCES `mydb`.`TiposDeModificacion` (`id` )
  ON DELETE NO ACTION
  ON UPDATE NO ACTION,
CONSTRAINT `fk_{E00D1747-32B7-4C6C-8DE8-D7BD07A92084}`
  FOREIGN KEY (`Usuarios_id` )
  REFERENCES `mydb`.`Usuarios` (`id` )
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

```

```

-----
-- Table `mydb`.`AccesosAlSistema`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`AccesosAlSistema` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `Usuarios_id` INT(11) NOT NULL ,
  `fechaDeAcceso` DATE NOT NULL ,
  `inicioDeSesion` TIME NOT NULL ,
  `terminoDeSesion` TIME NOT NULL ,
  PRIMARY KEY (`id`) ,
  INDEX `AccesosAlSistema_FKIndex1` (`Usuarios_id` ASC) ,
  CONSTRAINT `fk_{08A36E2D-F7D6-4FC1-8533-299AA33F7D3E}`
    FOREIGN KEY (`Usuarios_id` )
    REFERENCES `mydb`.`Usuarios` (`id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

```

```

-----
-- Table `mydb`.`TiposDeContaco`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`TiposDeContaco` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `tipo` INT(11) NOT NULL ,
  PRIMARY KEY (`id`) )
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

```

```

-----
-- Table `mydb`.`ModosDeContacto`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`ModosDeContacto` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `TiposDeContaco_id` INT(11) NOT NULL ,
  `Usuarios_id` INT(11) NOT NULL ,
  `contacto` VARCHAR(50) NOT NULL ,
  PRIMARY KEY (`id`) ,
  INDEX `ModosDeContacto_FKIndex1` (`Usuarios_id` ASC) ,
  INDEX `ModosDeContacto_FKIndex2` (`TiposDeContaco_id` ASC) ,
  CONSTRAINT `fk_{A10CDF04-A728-4B5A-9149-308940E34477}`
    FOREIGN KEY (`Usuarios_id` )

```

```

REFERENCES `mydb`.`Usuarios` (`id` )
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `fk_{D2FAA45B-1C00-447D-A15F-A6D9E4CB0247}`
FOREIGN KEY (`TiposDeContaco_id` )
REFERENCES `mydb`.`TiposDeContaco` (`id` )
ON DELETE NO ACTION
ON UPDATE NO ACTION)
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

```

```

-----
-- Table `mydb`.`CuentasBancarias`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`CuentasBancarias` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `numero` INT(11) NOT NULL ,
  `banco` INT(11) NOT NULL ,
  PRIMARY KEY (`id` )

```

```

PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

```

```

-----
-- Table `mydb`.`Pagos`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`Pagos` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `Usuarios_id` INT(11) NOT NULL ,
  `CuentasBancarias_id` INT(11) NOT NULL ,
  `fecha` DATE NOT NULL ,
  `hora` TIME NOT NULL ,
  `monto` FLOAT NOT NULL ,
  PRIMARY KEY (`id` ) ,
  INDEX `Pagos_FKIndex1` (`CuentasBancarias_id` ASC) ,
  INDEX `Pagos_FKIndex2` (`Usuarios_id` ASC) ,
  CONSTRAINT `fk_{31B58494-890D-4871-8CE9-FB740AC74A79}`
FOREIGN KEY (`CuentasBancarias_id` )
REFERENCES `mydb`.`CuentasBancarias` (`id` )
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `fk_{9B182327-ACF3-47AE-BA4E-F4D69CA7CB4E}`
FOREIGN KEY (`Usuarios_id` )
REFERENCES `mydb`.`Usuarios` (`id` )
ON DELETE NO ACTION
ON UPDATE NO ACTION)

```

```

PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

```

```

-----
-- Table `mydb`.`TiposDeMensaje`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`TiposDeMensaje` (
  `id` TINYINT UNSIGNED NOT NULL AUTO_INCREMENT ,
  `tipo` VARCHAR(20) NOT NULL ,
  PRIMARY KEY (`id` )

```

```

PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

```

-- Table `mydb`.`Mensajes`

```
CREATE TABLE IF NOT EXISTS `mydb`.`Mensajes` (  
  `id` INT(11) NOT NULL AUTO_INCREMENT ,  
  `Usuarios_id` INT(11) NOT NULL ,  
  `TiposDeMensaje_id` TINYINT UNSIGNED NOT NULL ,  
  `idUsuarioDestinatario` INT(11) NOT NULL ,  
  `fechaDeCreacion` DATE NOT NULL ,  
  `horaDeCreacion` TIME NOT NULL ,  
  `estadoDeMensaje` VARCHAR(20) NOT NULL ,  
  `mensaje` LONGTEXT NOT NULL ,  
  PRIMARY KEY (`id`) ,  
  INDEX `Mensajes_FKIndex1` (`TiposDeMensaje_id` ASC) ,  
  INDEX `Mensajes_FKIndex2` (`Usuarios_id` ASC) ,  
  CONSTRAINT `fk_{289F8E1A-CB1C-4380-BFA9-1ADEF1C8276D}`  
    FOREIGN KEY (`TiposDeMensaje_id` )  
    REFERENCES `mydb`.`TiposDeMensaje` (`id` )  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_{24017EFA-A5BD-4978-A8AD-09B9B625DA78}`  
    FOREIGN KEY (`Usuarios_id` )  
    REFERENCES `mydb`.`Usuarios` (`id` )  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
PACK_KEYS = 0  
ROW_FORMAT = DEFAULT;
```

-- Table `mydb`.`Direcciones`

```
CREATE TABLE IF NOT EXISTS `mydb`.`Direcciones` (  
  `id` INT(11) NOT NULL AUTO_INCREMENT ,  
  `Usuarios_id` INT(11) NOT NULL ,  
  `calle` VARCHAR(20) NOT NULL ,  
  `numeroInt` SMALLINT UNSIGNED NULL ,  
  `numeroExt` SMALLINT UNSIGNED NOT NULL ,  
  `colonia` VARCHAR(20) NOT NULL ,  
  `ciudad` VARCHAR(20) NOT NULL ,  
  PRIMARY KEY (`id`) ,  
  INDEX `Direcciones_FKIndex1` (`Usuarios_id` ASC) ,  
  CONSTRAINT `fk_{73D2464C-9A58-47B9-A4D3-026BA5DCEC41}`  
    FOREIGN KEY (`Usuarios_id` )  
    REFERENCES `mydb`.`Usuarios` (`id` )  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
PACK_KEYS = 0  
ROW_FORMAT = DEFAULT;
```

-- Table `mydb`.`DatosDeAcceso`

```
CREATE TABLE IF NOT EXISTS `mydb`.`DatosDeAcceso` (  
  `id` INT(11) NOT NULL AUTO_INCREMENT ,  
  `Usuarios_id` INT(11) NOT NULL ,  
  `login` VARCHAR(20) NOT NULL ,  
  `password_2` TEXT NOT NULL ,  
  PRIMARY KEY (`id`) ,
```

```

INDEX `DatosDeAcceso_FKIndex1` (`Usuarios_id` ASC) ,
CONSTRAINT `fk_{09545DF8-5EAC-4ABA-AB89-E9C3B0A88535}`
FOREIGN KEY (`Usuarios_id` )
REFERENCES `mydb`.`Usuarios` (`id` )
ON DELETE NO ACTION
ON UPDATE NO ACTION)
PACK_KEYS = 0
ROW_FORMAT = DEFAULT;

```

```

SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;

```

Diccionario de datos

Tabla marcasDeProductos

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.
Marca	VARCHAR			NO	Descripción de la marca

Tabla datosDeAcceso

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.
Login	VARCHAR			NO	El login de acceso del usuario
Password	TINYTEXT			NO	El password del usuario, se almacenará encriptado con MD5
Usuarios_id	INT		FK	NO	Identificador que conecta con la tabla usuarios.

Tabla productos

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.

MarcasDeProductos_id	INT		FK	NO	Identificador que conecta con la tabla marcasDeProductos
Nombre	VARCHAR			NO	El nombre del producto
CodigoDeBarras	VARCHAR				El código de barras del producto
Descuento	FLOAT				Indica el porcentaje de descuento que tiene el producto
PrecioUnitario	FLOAT			NO	El precio del producto
Descripción	MEDIUMTEXT				Contiene una descripción de los detalles del producto
Disponibles	INT				La cantidad de productos que hay en existencia
SubtiposDeProductos_id	INT		FK	NO	Identificador que conecta con la tabla subtiposDeProductos
Imagen	BLOB				Se guarda la imagen como un dato BLOB
Estado	INT			NO	El estado en que se encuentra el producto, puede ser activo, inactivo o agotado.

Tabla tiposDeCotizacion

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.
Descripción	VARCHAR			NO	La descripción del estado de la cotización

Tabla cotizaciones

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
-------	------	----------------	---------------	------	-------------

Id	INT	PK	NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.	
tipoDeCotizacion_id	INT		FK	NO	Identificador que conecta con la tabla tipoDecotizacion que indica el estado de la cotización
Usuarios_id	INT		FK	NO	El id del usuario que genero la cotización
fechaDeCotizacion	TIMESTAMP			NO	La fecha en que se genero la cotización, es generada automáticamente por el manejador de la base de datos.
fechaDeVigencia	DATE				La fecha de vigencia de la cotización ya que pueden aumentar de precio los productos cotizados
importeParcial	FLOAT			NO	La suma de los precios de los productos sin iva.
importeTotal	FLOAT			NO	La suma de los precios de los productos con iva
descuentoTotal	FLOAT			NO	La suma de los descuentos de los productos
IVA	FLOAT			NO	La suma del IVA de cada producto

Tabla direcciones

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.
Usuarios_id	INT		FK	NO	El id del usuario al que pertenece la dirección
Calle	VARCHAR			NO	Nombre de la calle
numeroInt	VARCHAR				Numero interior, no obligatorio
numeroExt	VARCHAR			NO	Numero exterior obligatorio
Colonia	VARCHAR			NO	El nombre de la colonia
Ciudad	VARCHAR			NO	El nombre de la ciudad
codigoPostal	VARCHAR			NO	Código postal

Tabla productosEnCotizacion

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.
Id_producto	INT		FK	NO	Id del producto que se esta cotizando correspondiente a la tabla productos
precioUnitario	FLOAT			NO	Precio unitario al momento de realizarse la cotización
Cantidad	INT				La cantidad de productos de un mismo tipo cotizados
IVA	FLOAT			NO	El IVA del producto al momento de realizarse la cotización
Importe	FLOAT			NO	El precio del producto sin IVA al momento de realizarse la cotización
Total	FLOAT			NO	La suma del IVA mas el importe
Id_cotizacion	INT		FK	NO	El id de la cotización al que pertenece el producto

Tabla tipoDeProducto

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.
Tipo	VARCHAR			NO	Descripción del tipo de producto

Tabla tipoDeUsuario

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.
Tipo	VARCHAR			NO	Descripción del tipo de producto

Tabla estadosFactura

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.
Estado	VARCHAR			NO	Descripción del estado de la factura

Tabla modosDeContacto

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.
tipoDeContacto_id	INT		FK	NO	El id del tipo de contacto correspondiente en la tabla tipoDeContacto
Usuarios_id	INT		FK	NO	El id del usuario al que pertenece el contacto
Contacto	VARCHAR			NO	El contacto(teléfono, correo, fax o teléfono celular)

Tabla tiposDeContacto

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.
Tipo	VARCHAR			NO	Descripción del tipo, puede ser correo, teléfono , teléfono celular, etc.

Tabla usuarios

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.
Ocupacion_id	INT		FK	NO	El id de la ocupación
tipoDeUsuario_id	INT		FK	NO	El id del tipo de usuario

Nombres	VARCHAR	NO	Nombres del usuario
ApellidoPaterno	VARCHAR	NO	Apellido paterno del usuario
ApellidoMaterno	VARCHAR		Apellido materno del usuario
Edad	INT		Edad del usuario
RFC	VARCHAR	NO	RFC del usuario

Tabla facturas

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.
estadoFactura_id	INT		FK	NO	El id del estado de la factura
Usuarios_id	INT		FK	NO	El usuario al que pertenece la factura
Cotizaciones_id	INT		FK	NO	Cada factura tiene asociada una cotización, este dato es el id de la cotización asociada
Serie	INT			NO	Numero de serie de la factura
Folio	INT			NO	Folio de la factura
fechaDeExpedicion	TIMESTAMP			NO	Fecha en que se registro la factura, dato generado automáticamente por el manejador de la base de datos
fechaLimiteDePago	DATE				Fecha limite de pago
numeroDeAprobacion	VARCHAR			NO	El número de aprobación de la factura
fechaDePago	DATETIME				Registra la fecha en que se registro el pago de la factura.

Tabla Ocupaciones

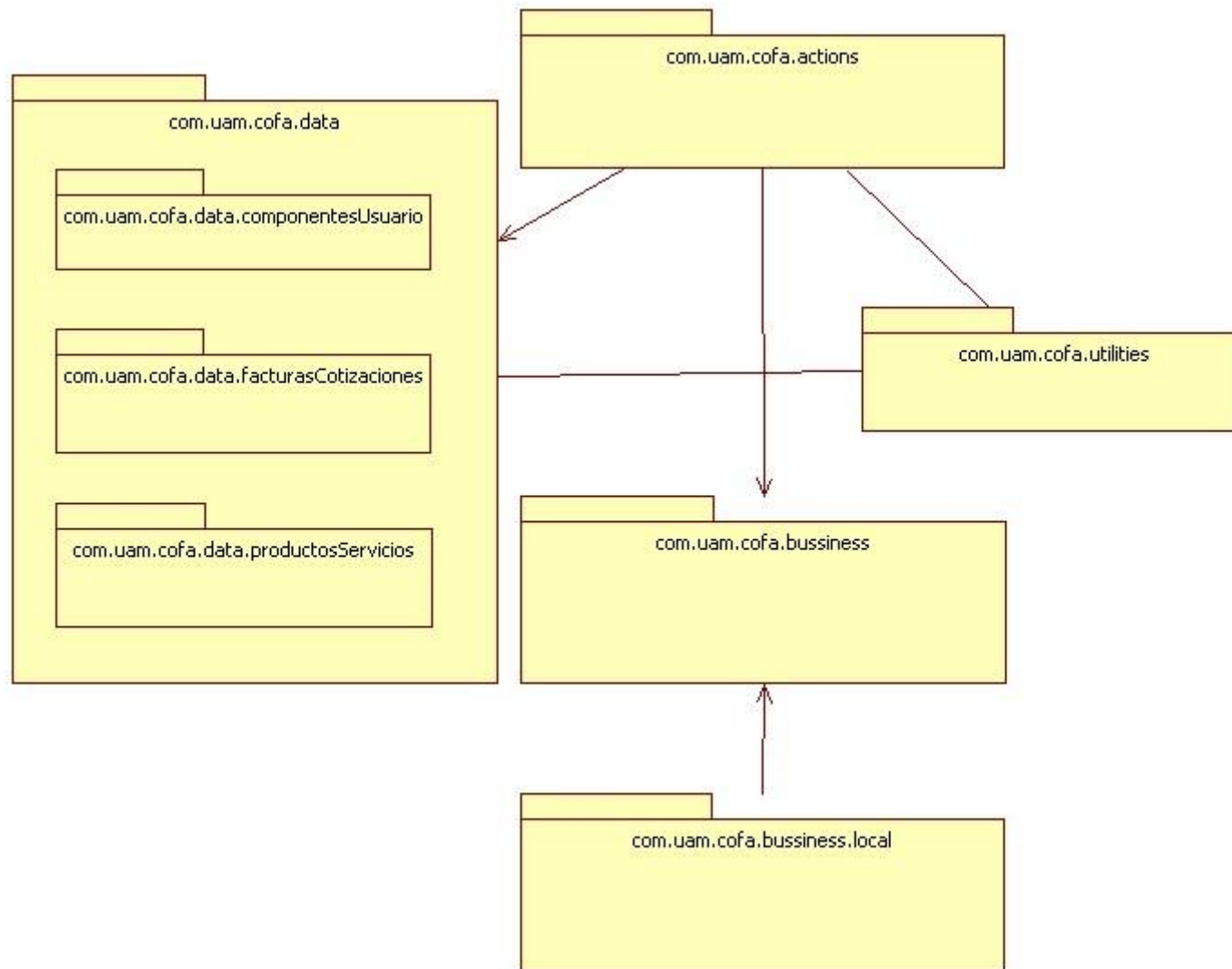
Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero

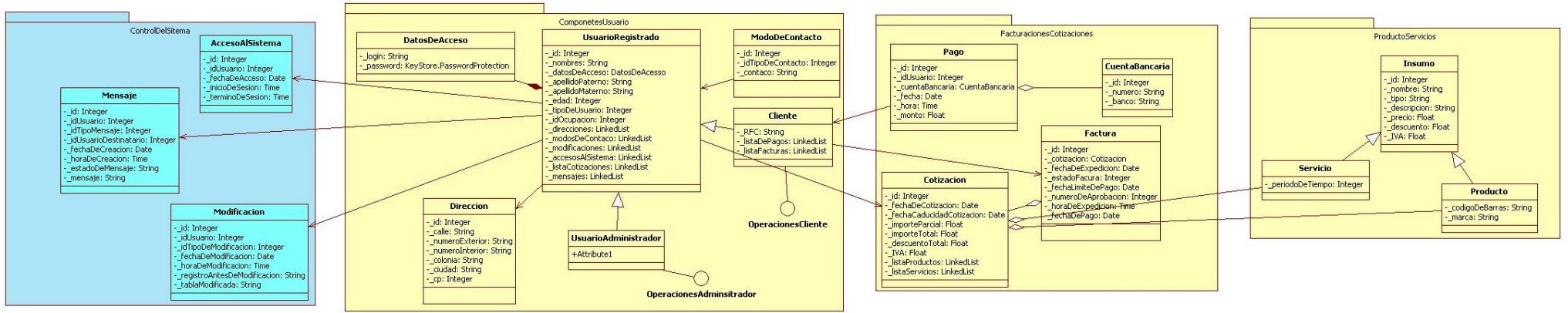
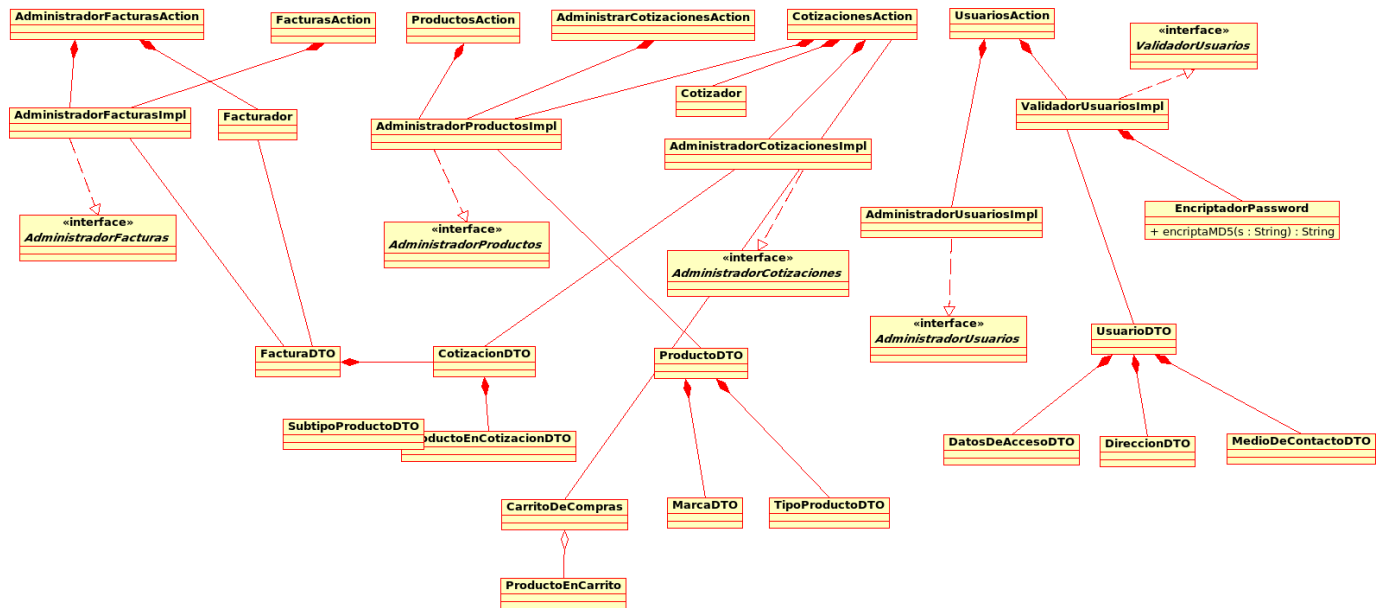
			secuencial autoincrementable en cada nuevo registro.
Ocupación	VARCHAR	NO	Descripción de la ocupación

Tabla subtiposDeProductos

Campo	Tipo	Llave primaria	Llave foránea	Nulo	Comentarios
Id	INT	PK		NO	Identificador único. Numero secuencial autoincrementable en cada nuevo registro.
tipoDeProdcuto_id	INT		FK	NO	Id correspondiente al tipo de producto
subtipo	VARCHAR			NO	Descripción del subtipo

Diagramas de clases





Modelo de casos de uso

Los servicios generales que el portal proporcionará a los usuarios son: Gestión de cotizaciones, gestión de facturas, gestión productos y gestión de usuarios.

Diagrama de caso de uso para validación de usuarios

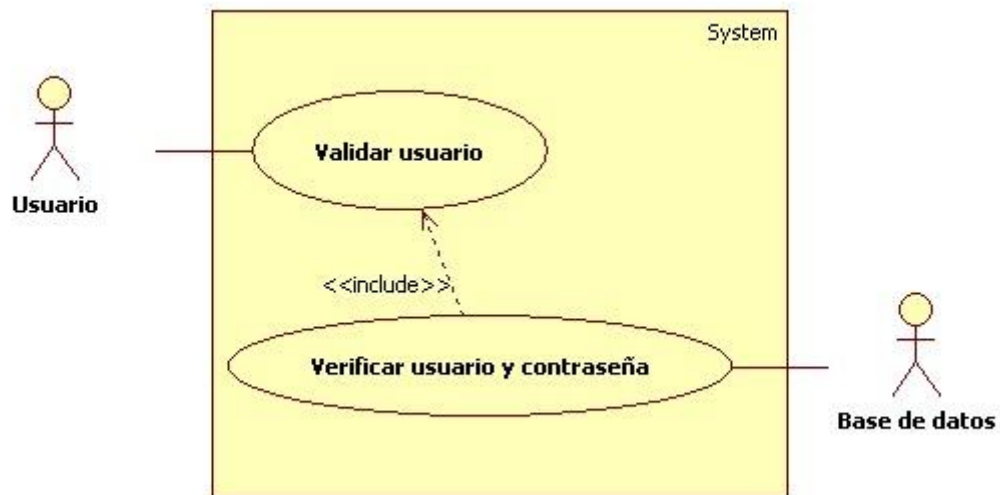


Diagrama de casos de uso de la gestión de productos.

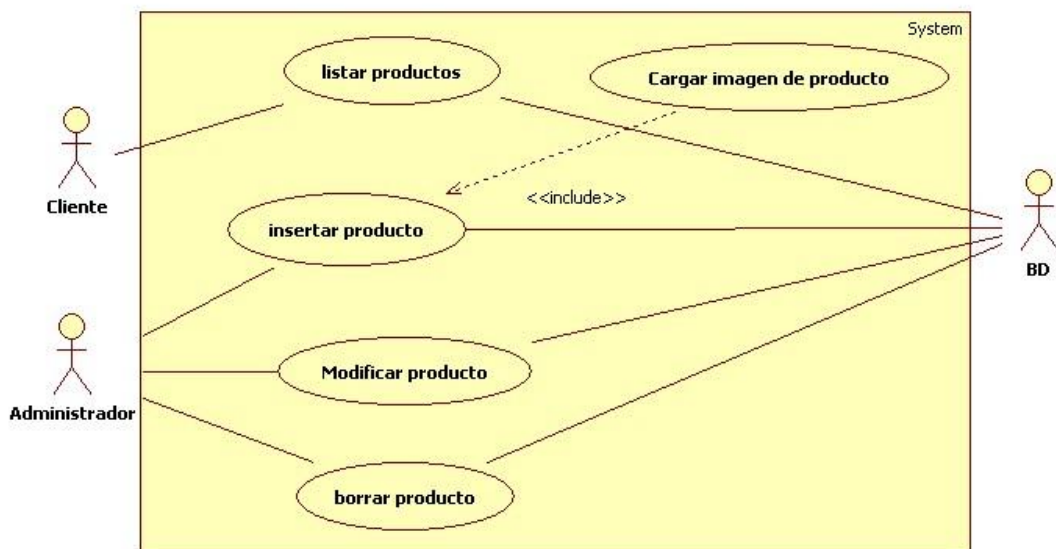
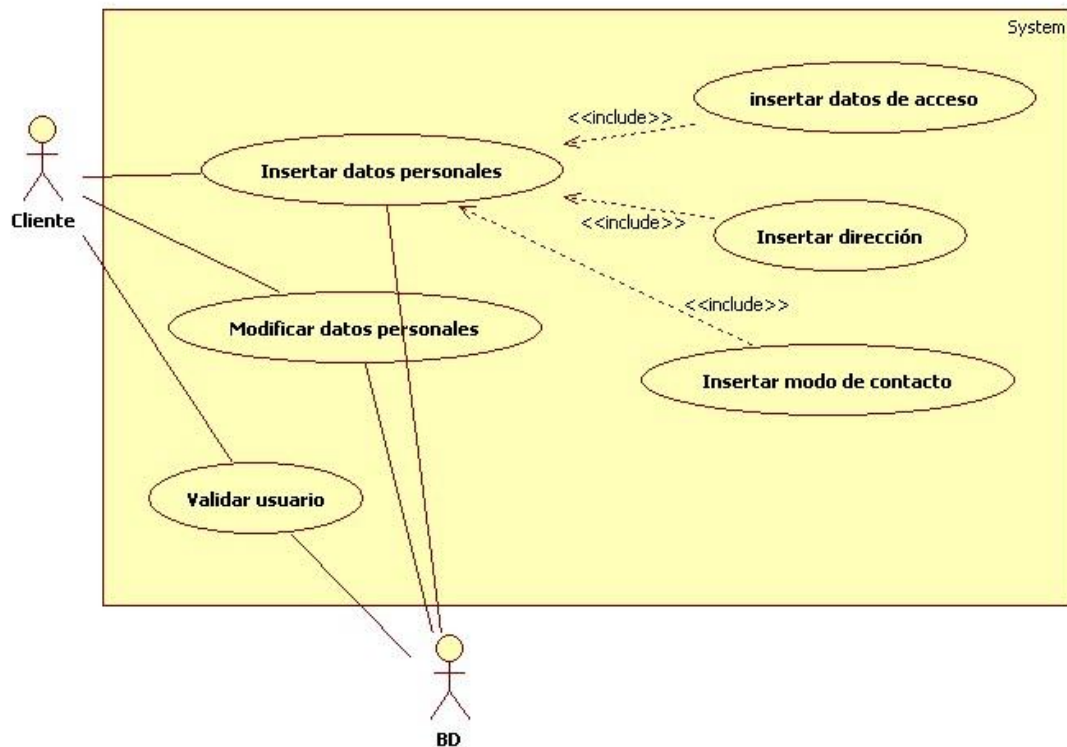


Diagrama de casos de uso de la gestión de usuarios



Documentación de actores

Actor	Administrador del sistema
Casos	Generar factura, cancelar factura, cancelar cotización, mostrar información de cotización, mostrar cotizaciones, descargar comprobante PDF, ver detalles de factura, mostrar factura, insertar producto, modificar producto, listar productos borrar producto
Tipo	Primario
Descripción	Es el usuario que se encarga de la administración de los procesos de facturación, cotización y gestión de productos

Actor	Usuario registrado
Casos	Cancelar cotización, cerrar cotización, generar cotización, guardar cotización, insertar dirección, modificar dirección, borrar dirección, descargar comprobante PDF, ver detalles de factura, mostrar factura, insertar modo de contacto, insertar datos personales, seleccionar cantidad de productos.

Tipo	Primario
Descripción	Representa a cualquier usuario (no administrador) que acceda al sistema

Actor	Base de datos
Casos	Generar factura, cancelar factura, cancelar cotización, mostrar información de cotización, mostrar cotizaciones, ver detalles de factura, mostrar factura, insertar producto, modificar producto, listar productos, borrar producto , cerrar cotización, generar cotización, guardar cotización, insertar dirección, modificar dirección, borrar dirección, insertar modo de contacto, insertar datos personales.
Tipo	Secundario
Descripción	Representa la base de datos donde se registran los movimientos realizados por los usuarios.

Documentación de casos de uso

Validar usuario

El usuario introduce su clave de usuario y su password.

The image shows a web page for COFA. At the top, there is a dark blue header with the COFA logo and the tagline 'sistema de cotizaciones y facturas'. To the right of the logo are navigation links: 'Home', 'Servicios', 'Productos', and 'Registrarse'. Below the header, the main content area has the heading 'Bienvenido a cofa'. Underneath this heading is a login form with two input fields: 'Usuario' and 'Password'. Below the 'Password' field is a button labeled 'Entrar'. At the bottom of the page, there is a dark blue footer with the text 'COPYRIGHT (C) 2008 COFA.COM ALL RIGHTS RESERVED'.

El password es encriptado con MD5 se envía al controlador, se consulta en la base de datos, si los datos son válidos se muestra la siguiente pantalla.



Bienvenido a cofa

Bienvenido(a):
eduardo

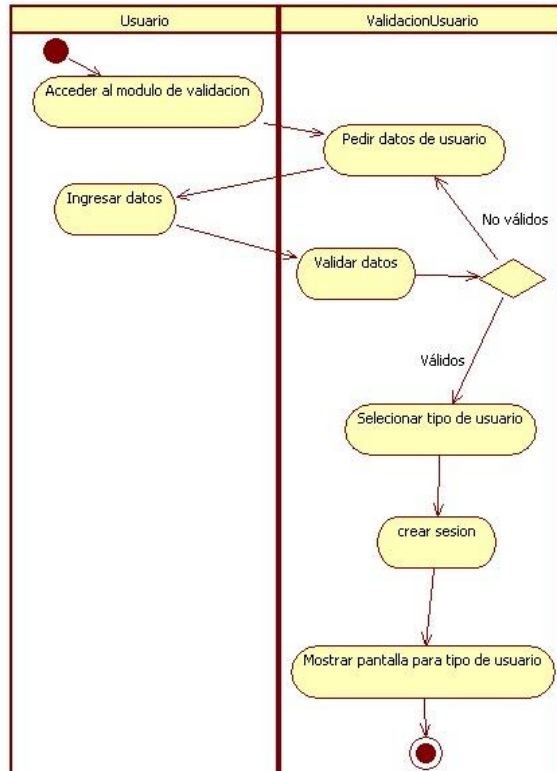
Si los datos no corresponden con los que se encuentran en la base de datos se redirecciona a esta pantalla.



Ingresar tus datos
Datos inválidos. Favor de verificar

login

password



Caso de uso registro de nuevo usuario

Se selecciona la opción registrarse

The screenshot shows the COFA website header with navigation links: Home, Servicios, Productos, and Registrarse. The main heading is 'Bienvenido a cofa'. Below it is a registration form with fields for 'Usuario' and 'Password', and an 'Entrar' button. The footer contains the text: 'COPYRIGHT (C) 2008 COFA.COM ALL RIGHTS RESERVED.'

Al seleccionar la opción se muestra la siguiente pantalla. En esta pantalla se capturan datos personales del usuario.

Nombres

Apellido paterno

Apellido materno

Ocupación

Edad

RFC

[Consultar RFC](#)

login

password

Al presionar el botón insertar, los datos enviados se guardan en sesión para su posterior almacenamiento en la base de datos ya que la inserción se realiza hasta que el usuario capture todos sus datos (datos personales, dirección y modos de contacto).

Correo

Teléfono casa

Teléfono celular

Teléfono trabajo

Fax

Calle

Número interior

Número exterior

Colonia

Código postal

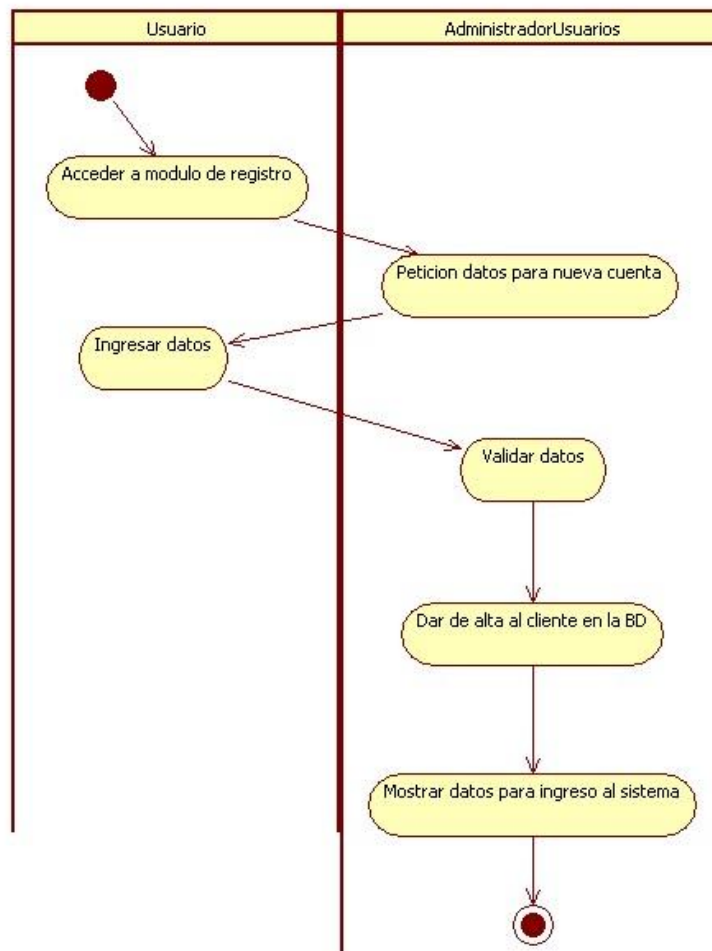
Ciudad

Al terminar de capturar todos los datos se realiza la inserción en la base de datos y se muestra la siguiente pantalla.



Gracias por registrarte.
Tu registro ha finalizado
correctamente.

[Accede aquí para continuar](#)



Caso de uso agregar producto

Al entrar como usuario administrador se selecciona la opción de productos y aparece la siguiente pantalla.

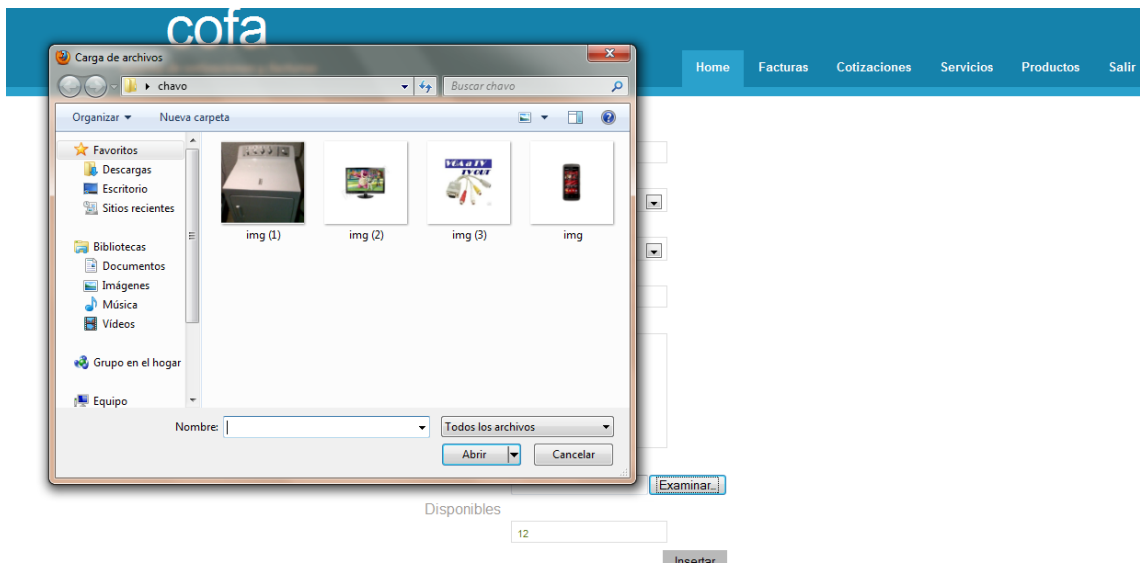


Se selecciona la opción de agregar y aparece la siguiente pantalla.

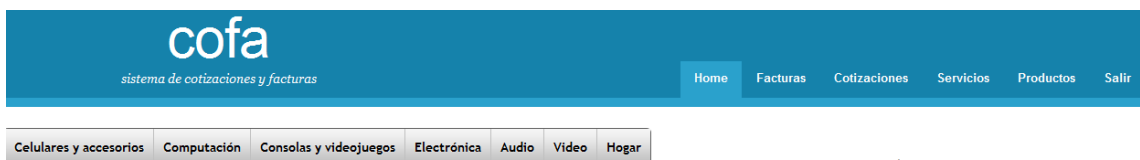
The screenshot shows the 'Agregar producto' form. The header is identical to the previous screenshot. The form fields are as follows:

- Nombre:
- Marca: (dropdown menu)
- Clasificación: (dropdown menu)
- Precio:
- Descripcion:
- Imagen: (with an 'Examinar...' button)
- Disponibles:

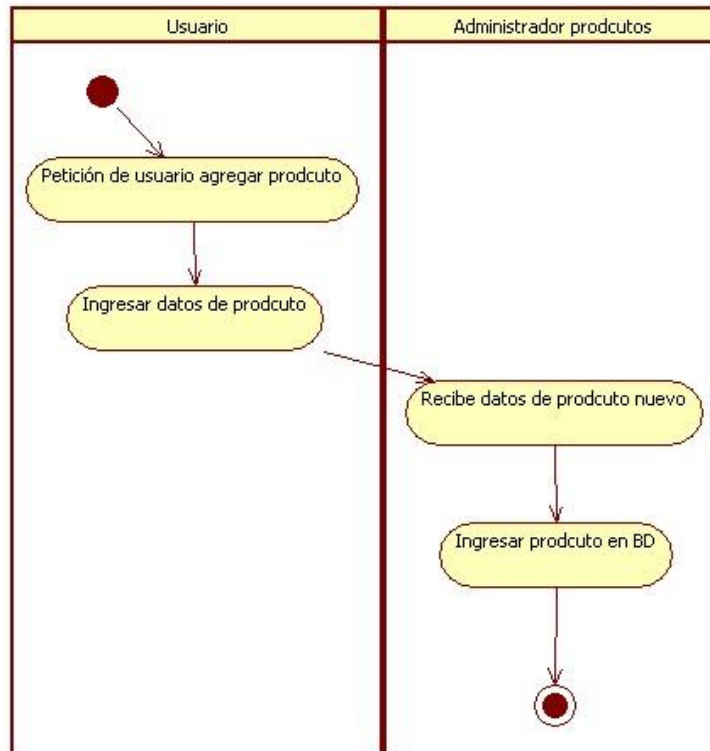
At the bottom right of the form area, there is an 'Insertar' button.



Al insertar los datos del producto se muestra la pantalla con los productos actuales en la base de datos.



Lista productos						
Imagen	Nombre	Marca	Precio	Clasificación	Borrar	modificar
	memoria 8 GB	1	\$ 2220.0	4	✗	
	Nokia N900	8	\$ 5320.0	2	✗	
	Celular Nokia C3	8	\$ 1800.0	2	✗	
	Celular Sony Ericsson W705	9	\$ 2597.0	2	✗	
	Sony Ericsson U8	9	\$ 4199.0	2	✗	
	Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	✗	
	Lap Top Dell Latitude D620 Intel Core Duo Wifi	3	\$ 3000.0	3	✗	
	Mario Land 2 En Excelente Estado	4	\$ 1750.0	8	✗	



















Caso de uso modificar producto

Se selecciona la opción productos y se escoge la parte de consultar



Al seleccionar la opción se muestra la siguiente pantalla donde se listan los productos para seleccionar el producto que se desee modificar.

Lista productos						
Imagen	Nombre	Marca	Precio	Clasificación	Borrar	modificar
	memoria 8 GB	1	\$ 2220.0	4	X	
	Nokia N900	8	\$ 5320.0	2	X	
	Celular Nokia C3	8	\$ 1800.0	2	X	
	Celular Sony Ericsson W705	9	\$ 2597.0	2	X	
	Sony Ericsson U8	9	\$ 4199.0	2	X	
	Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	X	
	Lap Top Dell Latitude D620 Intel Core Duo Wifi	3	\$ 3000.0	3	X	
	Marin Land 2 En Excelente Estado	4	\$ 1750.0	8	X	

Se selecciona el producto a modificar, el controlador se encarga de obtener los datos del producto y desplegarlos en la forma_modificarProducto.

Modificar producto

Nombre

Marca

Clasificación

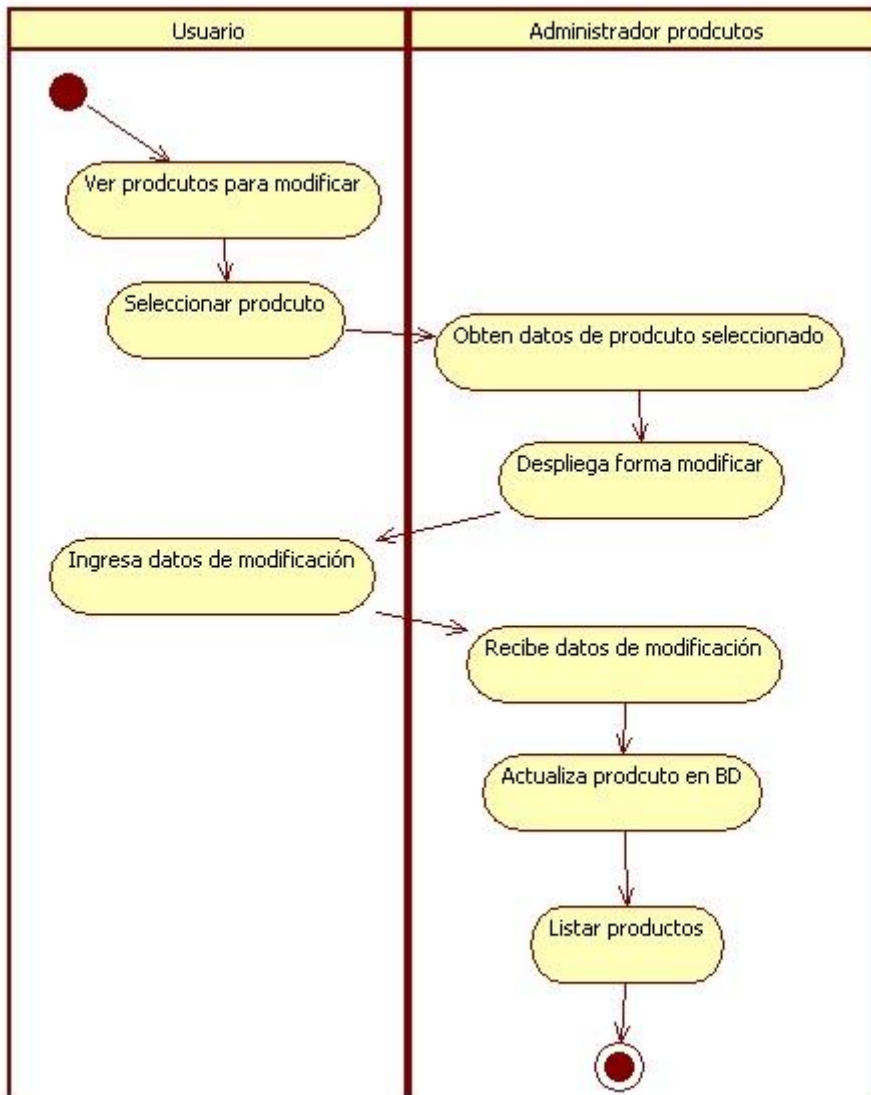
Precio

Descripcion

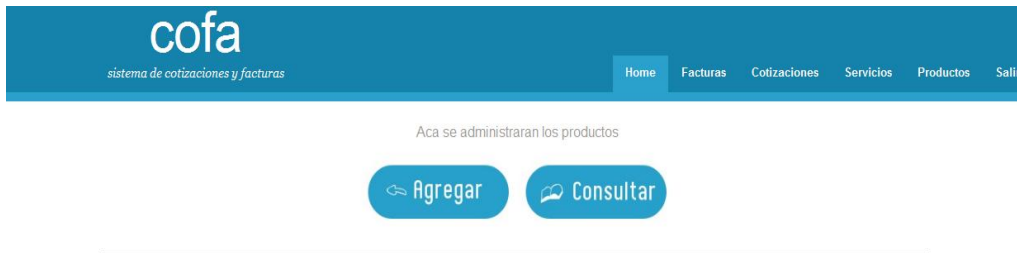
Imagen

El controlador obtiene los datos ingresados para la modificación del producto y realiza la actualización del producto en la base de datos.

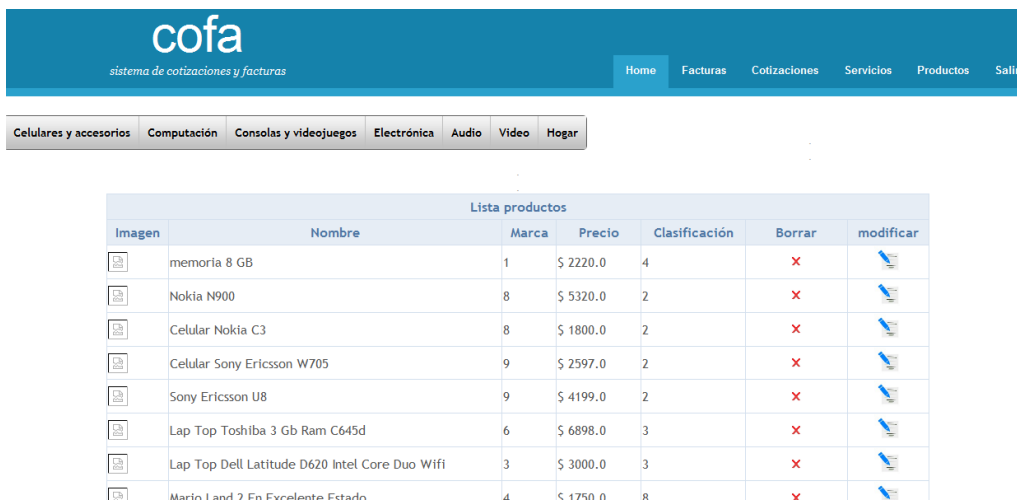
Lista productos						
Imagen	Nombre	Marca	Precio	Clasificación	Borrar	modificar
	memoria 8 GB	1	\$ 2220.0	4	✖	
	Nokia N900	8	\$ 5320.0	2	✖	
	Celular Nokia C3	8	\$ 1800.0	2	✖	
	Celular Sony Ericsson W705	9	\$ 2597.0	2	✖	
	Sony Ericsson U8	9	\$ 4199.0	2	✖	
	Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	✖	
	Lap Top Dell Latitude D620 Intel Core Duo Wifi	3	\$ 3000.0	3	✖	
	Mario Land 2 En Excelente Estado	4	\$ 1750.0	8	✖	



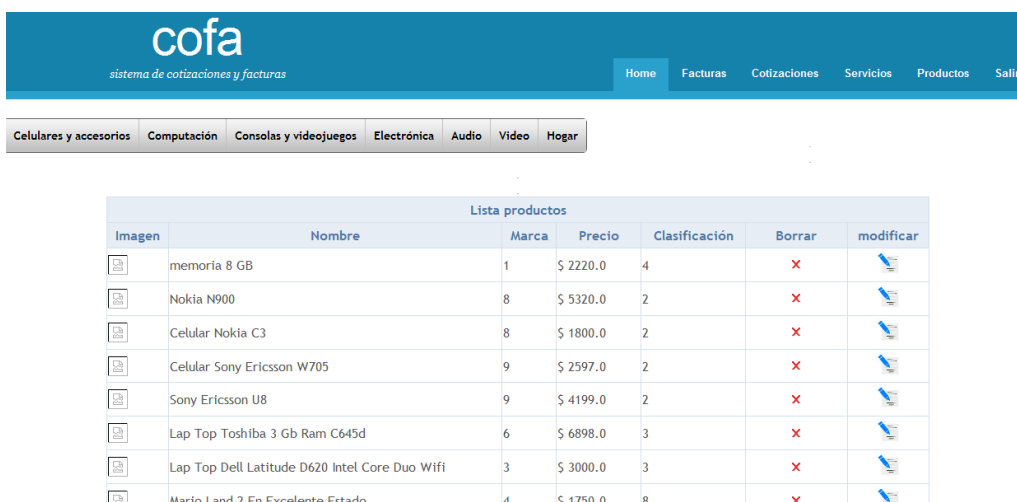
Caso de uso borrar producto

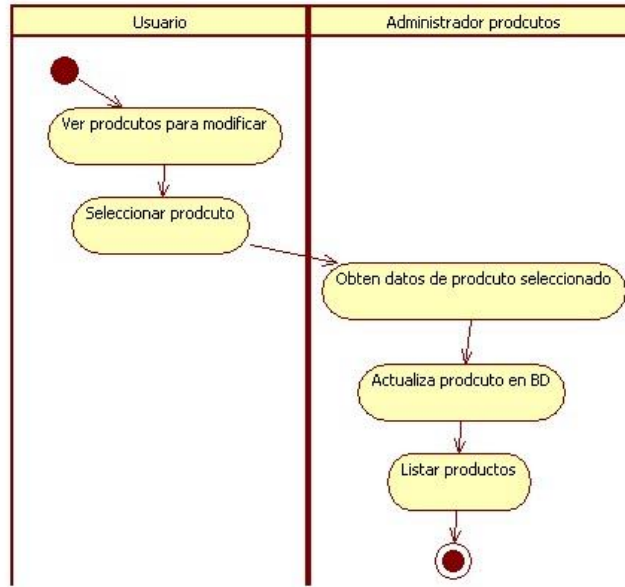


Se selecciona el producto a cancelar



El controlador obtiene los datos del producto a cancelar seleccionado y realiza la actualización del producto en la base de datos. El producto no es removido de la base de datos, solo es cambiado su estado a cancelado. Al realizar la actualización del producto el controlador redirecciona para listar los productos existentes.





Caso de uso generar cotización

Al elegir la opción de cotizaciones se muestra la siguiente pantalla donde hay dos listas, los productos existentes y los productos agregados al carrito de compras. Al presionar la palomita se le envía al controlador la petición de agregar el producto al carrito de compras, y se actualiza la segunda lista con los productos seleccionados.

Generación de cotizaciones

Ver mis Cotizaciones

- Celulares y accesorios
- Computación
- Consolas y videojuegos
- Electrónica
- Audio
- Video
- Hogar

Lista productos						
Imagen	Nombre	Marca	Precio	clasificación	Agregar	
[B@fb0b0	memoria 8 GB	1	\$ 2220.0	4	✓	
[B@18ff27	Nokia N900	8	\$ 5320.0	2	✓	
	Celular Nokia C3	8	\$ 1800.0	2	✓	
	Celular Nokia C3	8	\$ 1800.0	2	✓	
	Celular Sony Ericsson W705	9	\$ 2597.0	2	✓	
	Sony Ericsson U8	9	\$ 4199.0	2	✓	
	Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	✓	
	Lap Top Dell Latitude D620 Intel Core Duo Wifi	3	\$ 3000.0	3	✓	
[B@b8501d	Mario Land 2 En Excelente Estado	4	\$ 1750.0	8	✓	
[B@93a985	aass	3	\$ 25200.0	2	✓	
[B@6399ae	Transmisor Fm 4 En 1 Coche Usb	2	\$ 4990.0	9	✓	
[B@56d6cf	lavadora	8	\$ 450057.0	4	✓	

Lista de productos a cotizar				
Nombre	Marca	Precio	clasificación	Remover

Seleccionar número de productos

[B@4ba4d6	lavadora	8	\$ 450057.0	4	✓
-----------	----------	---	-------------	---	---

Lista de productos a cotizar				
Nombre	Marca	Precio	clasificación	Remover
Celular Sony Ericsson W705	9	\$ 2597.0	2	✗
Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	✗
memoria 8 GB	1	\$ 2220.0	4	✗
Nokia N900	8	\$ 5320.0	2	✗

Seleccionar número de productos

Cuando se han seleccionado los productos a cotizar se pasa a la selección de la cantidad de productos que se desean.

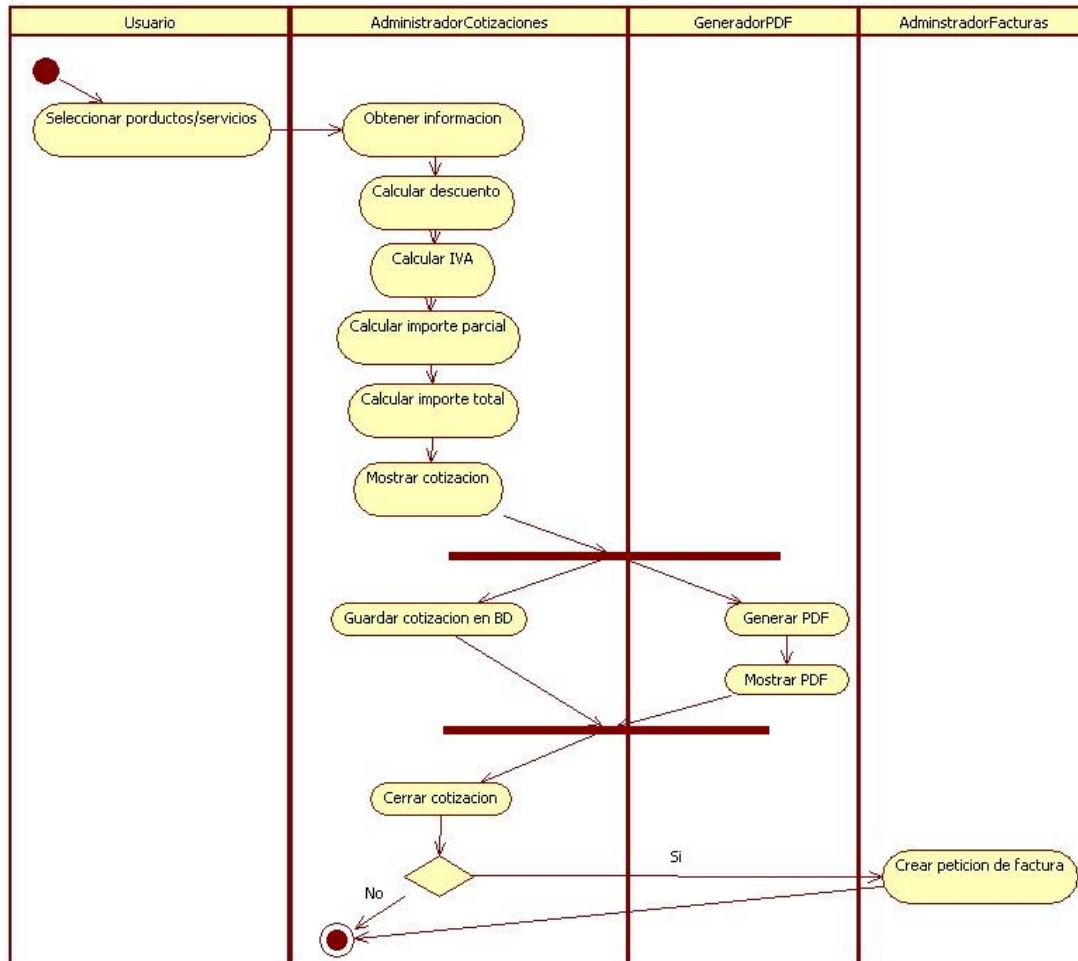
Selecciona cuantos productos deseas

Nombre	Precio	cantidad
Celular Sony Ericsson W705	\$ 2597.0	2
Lap Top Toshiba 3 Gb Ram C645d	\$ 6898.0	4
memoria 8 GB	\$ 2220.0	2
Nokia N900	\$ 5320.0	5

El controlador obtiene los datos del carrito de compras y realiza los cálculos necesarios para la cotización auxiliándose de un objeto de tipo Cotizador.

Cotización					
Nombre	Precio unitario	Cantidad	IVA	Importe	Total
Celular Sony Ericsson W705	\$ 2597.0	2	\$ 831.04	\$ 4362.96	\$ 5194.0
Lap Top Toshiba 3 Gb Ram C645d	\$ 6898.0	4	\$ 4414.7197	\$ 23177.281	\$ 27592.0
memoria 8 GB	\$ 2220.0	2	\$ 710.39996	\$ 3729.6	\$ 4440.0
Nokia N900	\$ 5320.0	5	\$ 4256.0	\$ 22344.0	\$ 26600.0

Totales		
Importe	IVA	Total
\$53613.844	\$10212.16	\$63826.004



Caso de uso cerrar cotización

cofa
sistema de cotizaciones y facturas

Home Mis Facturas Cotizaciones Servicios Productos Salir

Generación de cotizaciones

Ver mis Cotizaciones

Celulares y accesorios Computación Consolas y videojuegos Electrónica Audio Video Hogar

Lista productos

Imagen	Nombre	Marca	Precio	clasificación	Agregar
[B@fb0b0	memoria 8 GB	1	\$ 2220.0	4	✓
[B@18ff27	Nokia N900	8	\$ 5320.0	2	✓
	Celular Nokia C3	8	\$ 1800.0	2	✓

Mis cotizaciones registradas

Cotizaciones Pendientes

Cotizaciones Cerradas

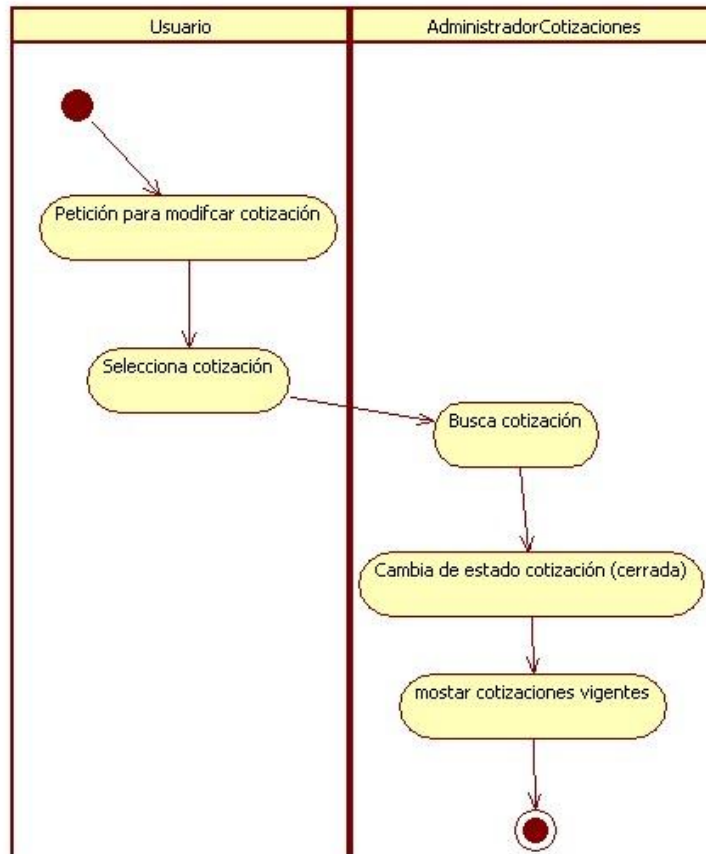
Mis cotizaciones						
Id	Fecha de cotización	Importe parcial	IVA	Importe total	Cerrar cotización	Cancelar cotización
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	✓	✗

Mis cotizaciones cerradas

Cotizaciones Pendientes

Cotizaciones Cerradas

Mis cotizaciones						
Id	Fecha de cotización	Importe parcial	IVA	Importe total	Cancelar cotización	
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	✗	
Insumos						
		\$ 1847.52		\$ 11547.0	✗	
Producto	Precio Unitario	Cantidad	Iva	Importe	Total	
	\$ 2597.0	2	\$ 831.04	\$ 4362.96	\$ 5194.0	
	\$ 6898.0	4	\$ 4414.7197	\$ 23177.281	\$ 27592.0	
	\$ 2220.0	2	\$ 710.39996	\$ 3729.6	\$ 4440.0	
	\$ 5320.0	5	\$ 4256.0	\$ 22344.0	\$ 26600.0	
		Total	\$ 10212.16	\$ 53613.844	\$ 63826.00	



Caso de uso cancelar cotización

cofa
sistema de cotizaciones y facturas

Home Mis Facturas Cotizaciones Servicios Productos Salir

Generación de cotizaciones

Ver mis Cotizaciones

Celulares y accesorios Computación Consolas y videojuegos Electrónica Audio Video Hogar

Lista productos

Imagen	Nombre	Marca	Precio	clasificación	Agregar
[B@fb0b0]	memoria 8 GB	1	\$ 2220.0	4	✓
[B@18ff27]	Nokia N900	8	\$ 5320.0	2	✓
	Celular Nokia C3	8	\$ 1800.0	2	✓

Mis cotizaciones registradas

Cotizaciones
Pendientes

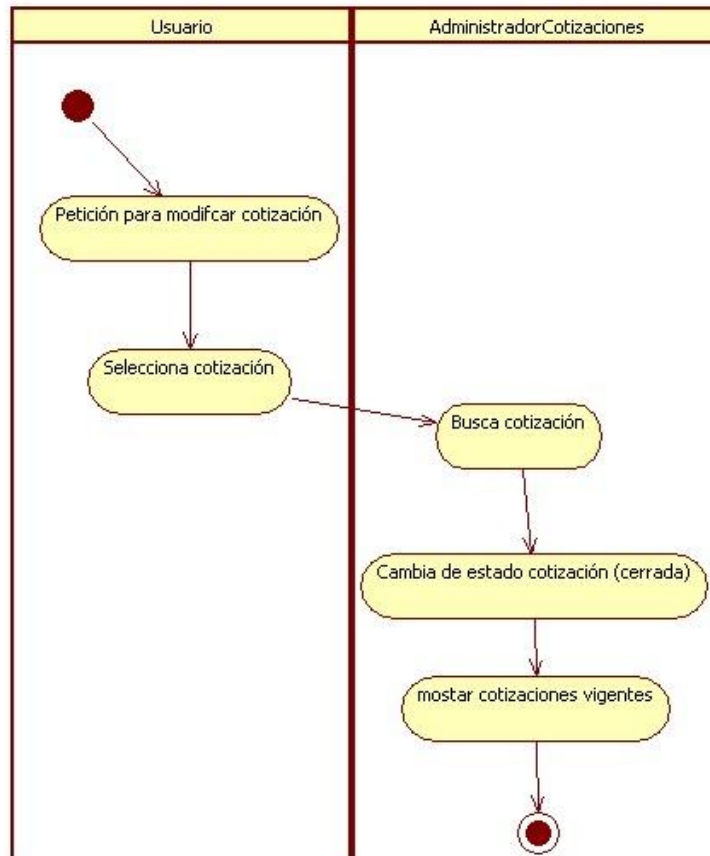
Cotizaciones
Cerradas

Mis cotizaciones						
Id	Fecha de cotización	Importe parcial	IVA	Importe total	Cerrar cotización	Cancelar cotización
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	✓	✗

Cotizaciones
Pendientes

Cotizaciones
Cerradas

Mis cotizaciones				
Id	Fecha de cotización	Importe parcial	IVA	Importe total
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004

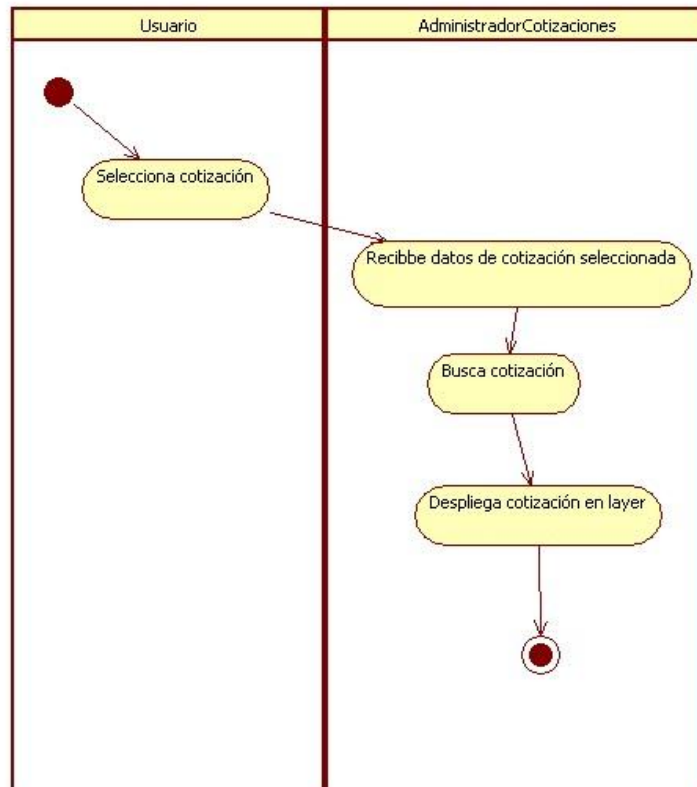


Caso de uso ver detalle de cotización

Mis cotizaciones cerradas

[Cotizaciones Pendientes](#)
[Cotizaciones Cerradas](#)

Mis cotizaciones						
Id	Fecha de cotización		Importe parcial	IVA	Importe total	Cancelar cotización
5 +/-	2011-04-06 15:15:03.0		\$ 53613.844	\$ 10212.16	\$ 63826.004	✗
Insumos						
lucto	Precio Unitario	Cantidad	Iva	Importe	Total	
	\$ 2597.0	2	\$ 831.04	\$ 4362.96	\$ 5194.0	
	\$ 6898.0	4	\$ 4414.7197	\$ 23177.281	\$ 27592.0	
	\$ 2220.0	2	\$ 710.39996	\$ 3729.6	\$ 4440.0	
	\$ 5320.0	5	\$ 4256.0	\$ 22344.0	\$ 26600.0	
Total			\$ 10212.16	\$ 53613.844	\$ 63826.00	



Caso de uso generar facturas

cofa
sistema de cotizaciones y facturas

Home Facturas Cotizaciones Servicios Productos Salir

Aca se listaran las facturas que estan vigentes

Generar Facturas

Consultar

Lista de cotizaciones cerradas

Id	Id Usuario	Fecha de cotización	Importe parcial	IVA	Importe total	Facturar
5	9	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	<input checked="" type="checkbox"/>
6	9	2011-04-06 15:21:26.0	\$ 9699.48	\$ 1847.52	\$ 11547.0	<input checked="" type="checkbox"/>

fecha limite

enviar

Lista de cotizaciones cerradas

Id	Id Usuario	Fecha de cotización	Importe parcial	IVA	Importe total	Facturar
5	9	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	<input checked="" type="checkbox"/>
6	9	2011-04-06 15:21:26.0	\$ 9699.48	\$ 1847.52	\$ 11547.0	<input checked="" type="checkbox"/>

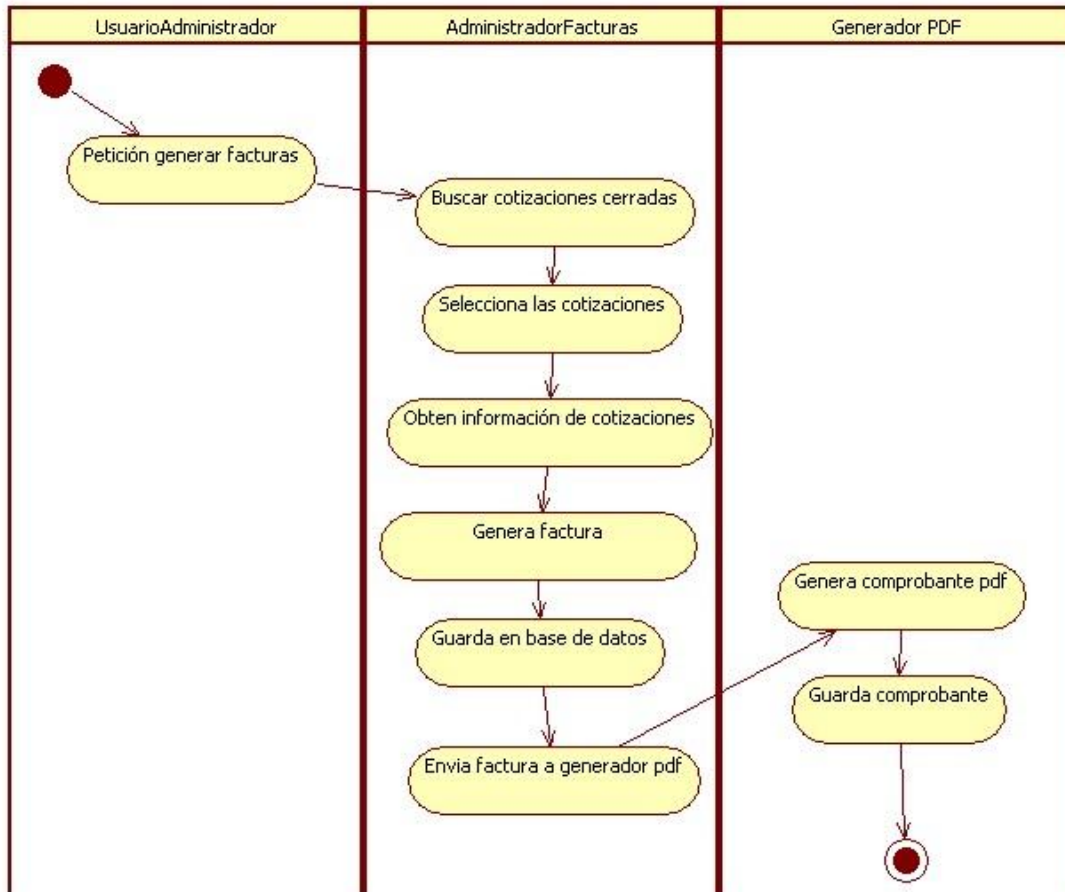
fecha limite

abril

L	M	J	V	S	D
28	29	30	31	1	2
4	5	6	7	8	9
11	12	13	14	15	16
18	19	20	21	22	23
25	26	27	28	29	30
2	3	4	5	6	7
2010 2011 2012					

Facturas vigentes

Folio	Estado	Id usuario	Fecha de expedición	Fecha límite de pago	Descargar pdf
12313 +/-	1	1	2011-03-24 12:36:28.0	2011-10-25	
12314 +/-	1	9	2011-04-05 11:44:55.0		
12315 +/-	1	9	2011-04-05 11:44:56.0		
12316 +/-	1	10	2011-04-05 19:30:25.0		
12317 +/-	1	9	2011-04-06 19:30:02.0		
12318 +/-	1	9	2011-04-06 19:30:02.0		

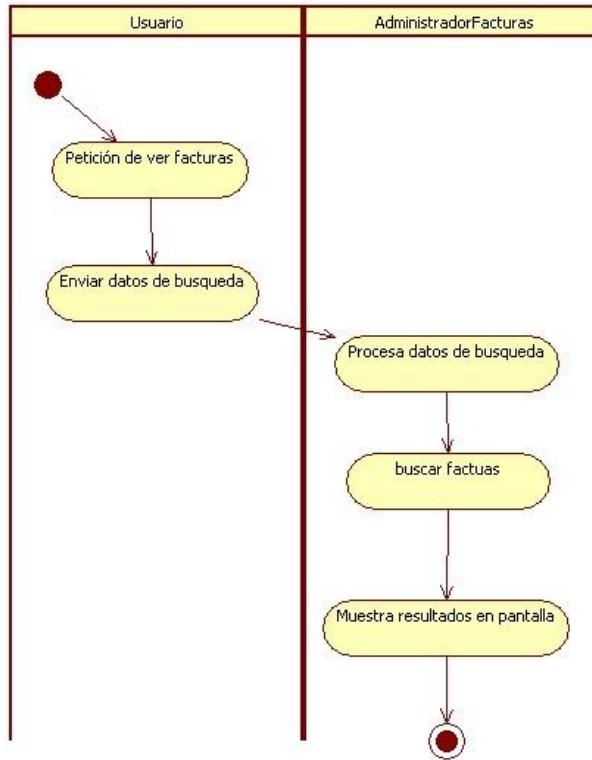


Caso de uso consultar facturas

cofa
sistema de cotizaciones y facturas

Home Mis Facturas Cotizaciones Servicios Productos Salir

Mis facturas				
Folio	Estado	Fecha de expedición	Fecha límite de pago	Descarga pdf
12314 +/-	1	2011-04-05 11:44:55.0		
12315 +/-	1	2011-04-05 11:44:56.0		



Caso de uso descargar factura

cofa
sistema de cotizaciones y facturas

Inicio Cotizaciones Servicios Productos Salir

Folio	Est	Fecha de pago	Descarga pdf
12314 +/-	1		⬇
12315 +/-	1		⬇

Abriendo Factura.pdf

Ha escogido abrir **Factura.pdf** que es de tipo: Adobe Acrobat Document de: http://localhost:8080

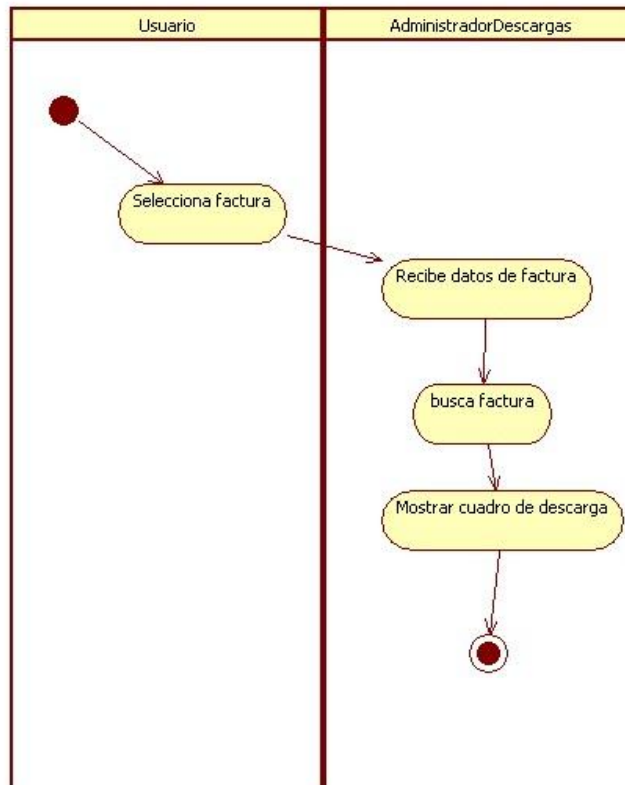
¿Qué debería hacer Firefox con este archivo?

Abrir con Adobe Reader (predeterminada)

Guardar archivo

Hacer esto automáticamente para estos archivos a partir de ahora.

Aceptar Cancelar



Diagramas de navegación

Diagrama de navegación para la ruta de facturas

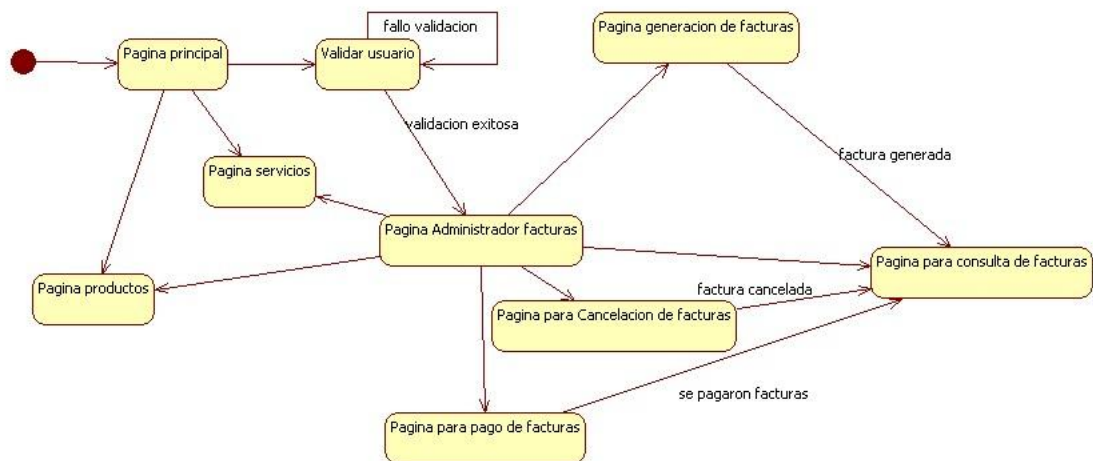


Diagrama de navegación del cliente

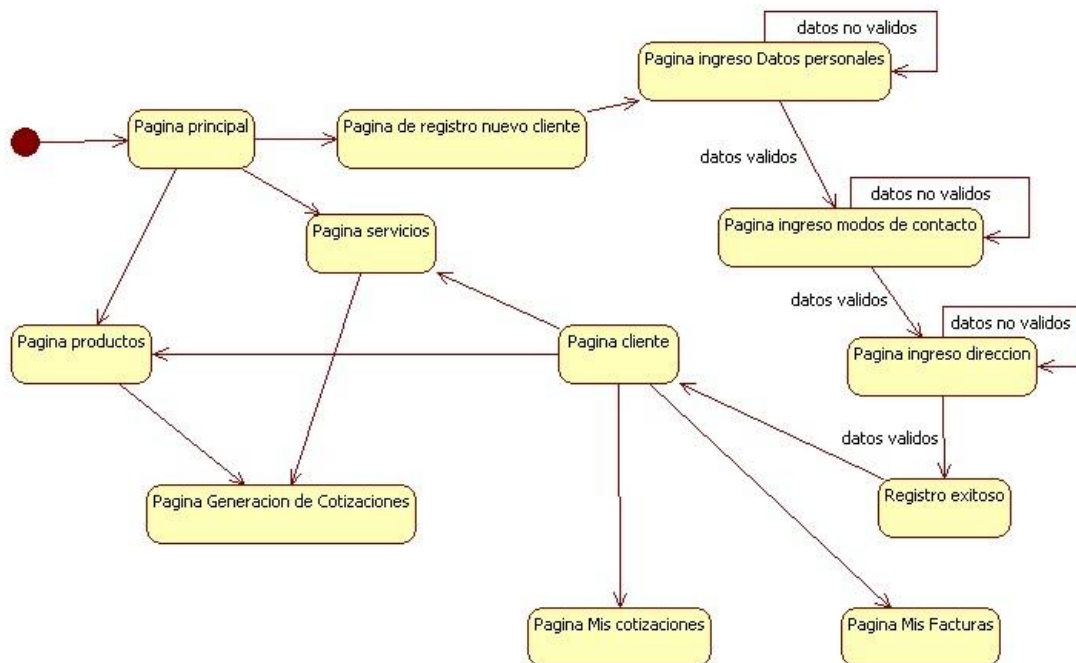


Diagrama de navegación general

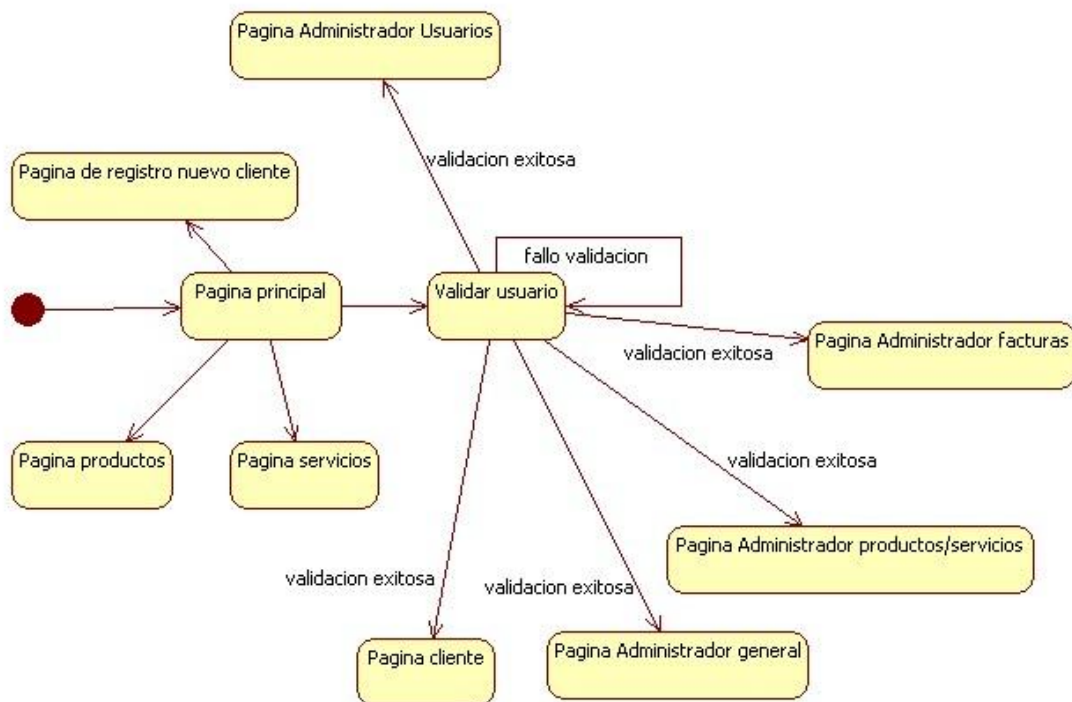
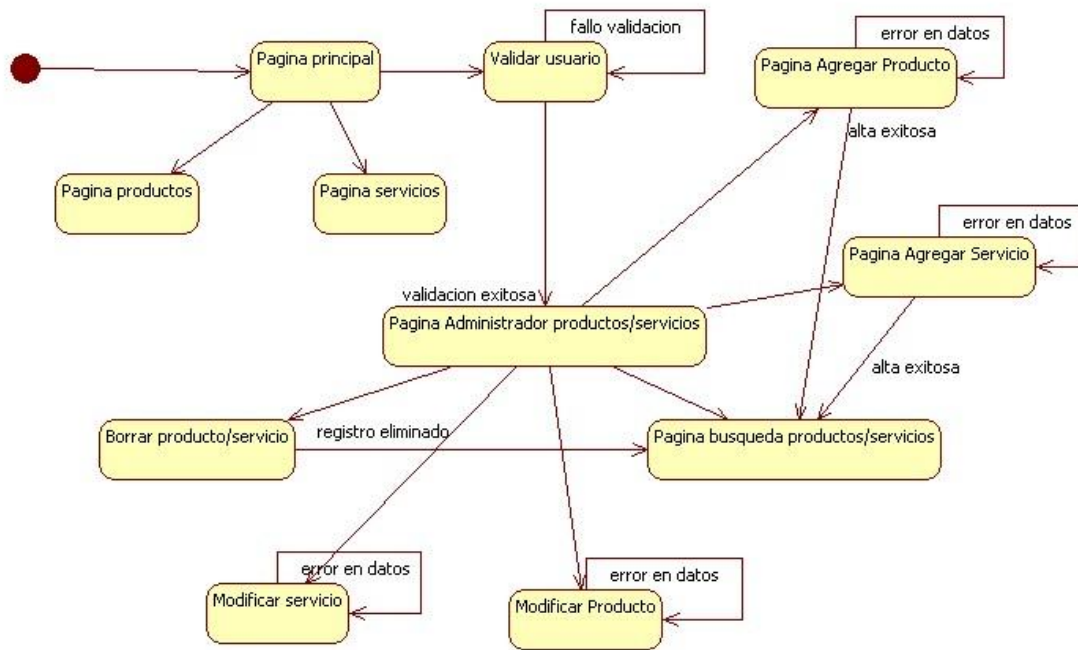


Diagrama de navegación para productos-servicios



Manual de instalación

Requisitos de instalación.....	2
Instalación de jdk	2
Instalación de mysql-server	6
Restauración de la base de datos.....	13
Instalación servidor tomcat.....	16
Montar la aplicación web.....	20
Ligas de descarga	22

Requisitos de instalación

Para instalar la aplicación se necesitan seguir 5 pasos que se muestran a continuación y se encuentran presentados a detalle a lo largo de este documento.

1. Instalación de JDK
2. Instalación de MySQL-server
3. Restauración de la base de datos
4. Instalación del servidor de aplicaciones TOMCAT
5. Montar la aplicación web.

Instalación de jdk

Entramos a la pagina de descarga del JDK SE Development kit 6u24 (la liga de descarga para el archivo se encuentra en la parte final del documento).

SDN Home > Download Center >

Java SE Development Kit 6u24

Provide Information, then Continue to Download

Select Platform and Language for your download:

Platform:

Language: Multi-language

I agree to the [Java SE Development Kit 6u24 License Agreement](#).

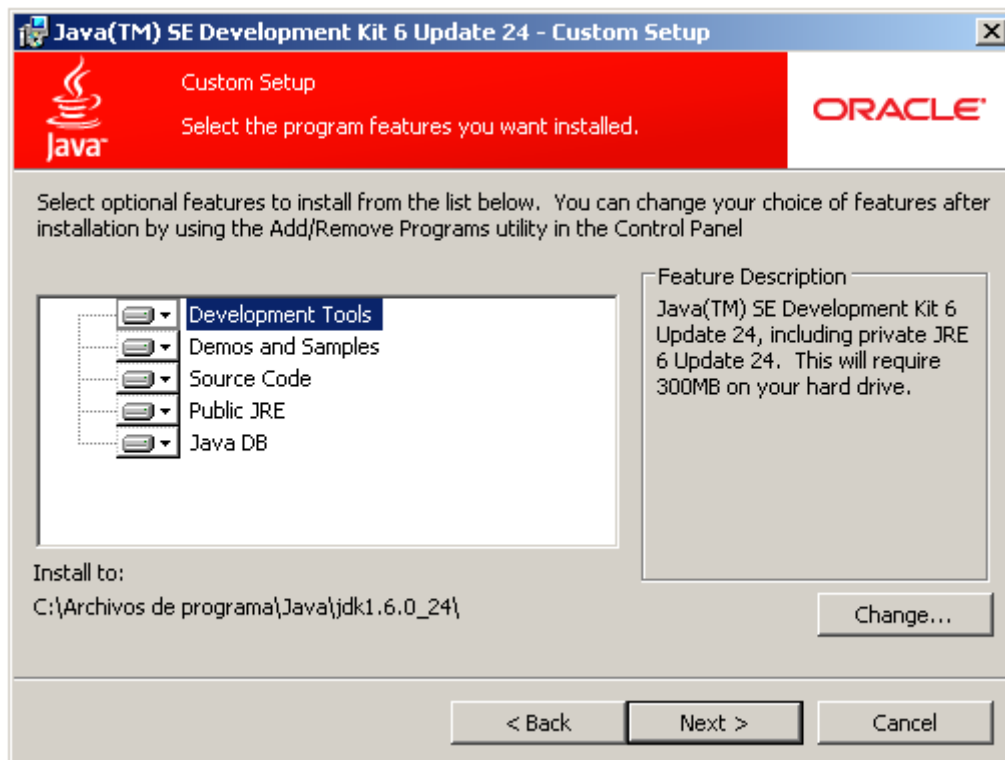
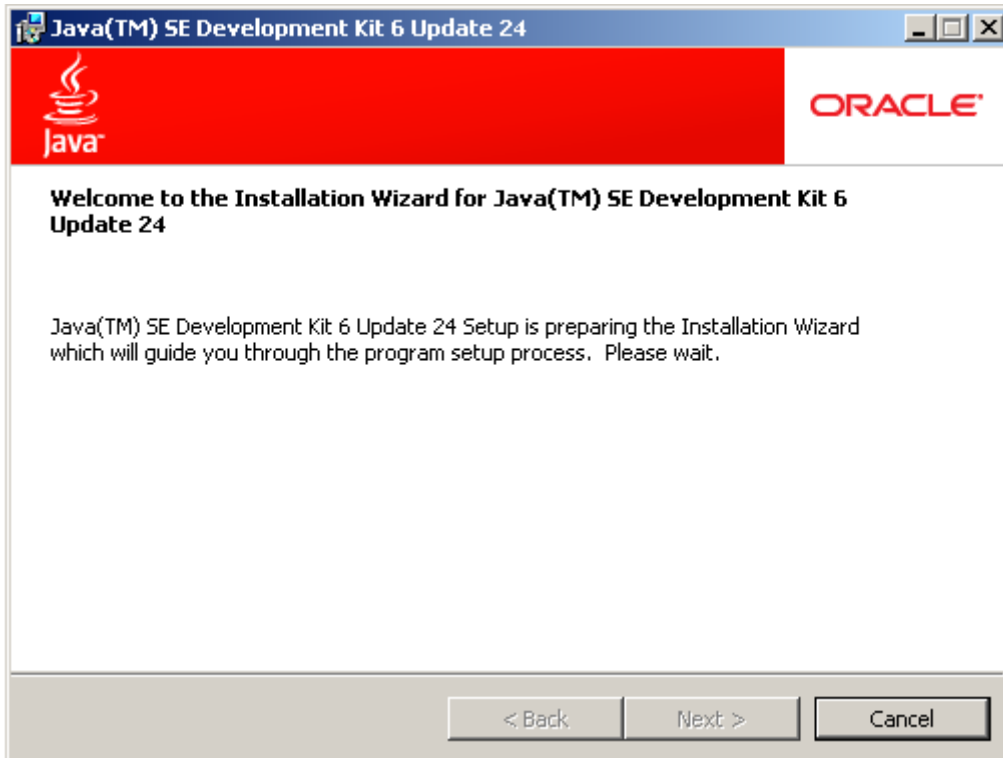
Continue »

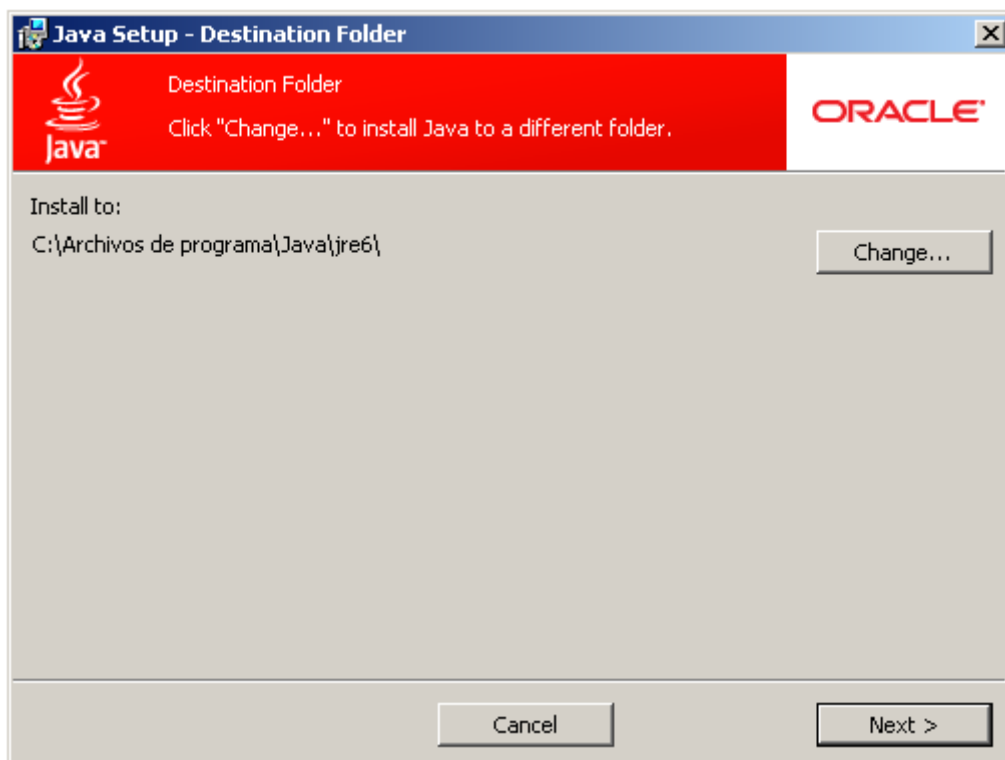
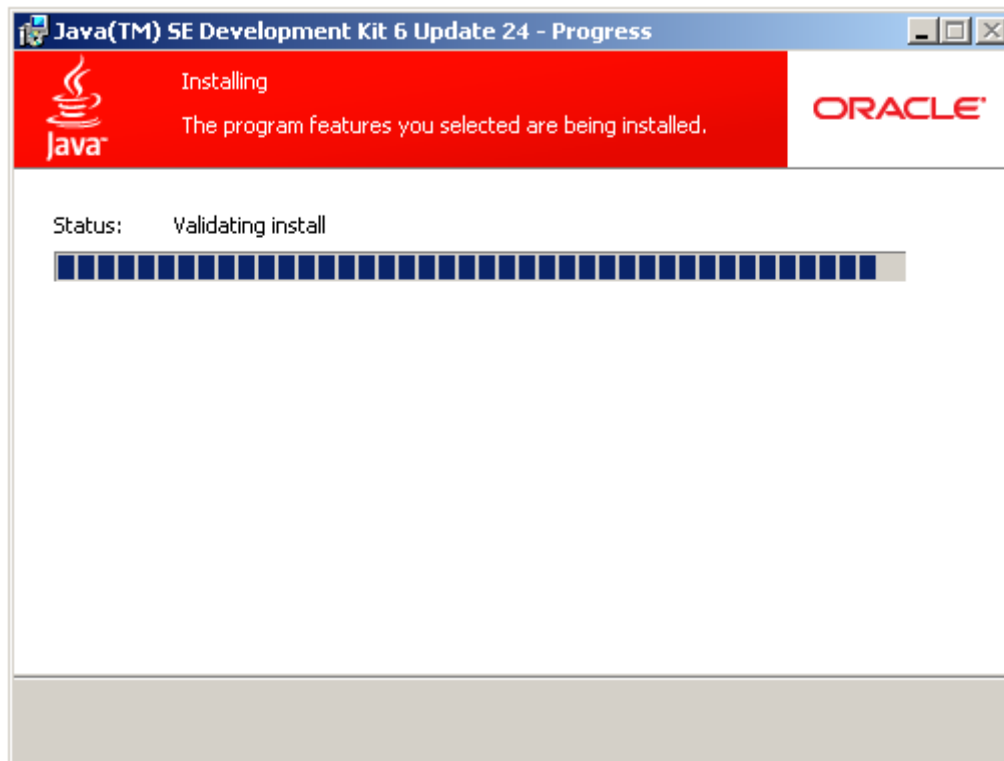
Una vez descargado el archivo lo ejecutamos



jdk-6u24-windows-i586









Aquí se termina de instalar el JDK.

Instalación de mysql-server

Descargamos el manejador de base de datos mysql-5.5.11 (6u24 (la liga de descarga para el archivo se encuentra en la parte final del documento).

Una vez descargado el archivo lo ejecutamos



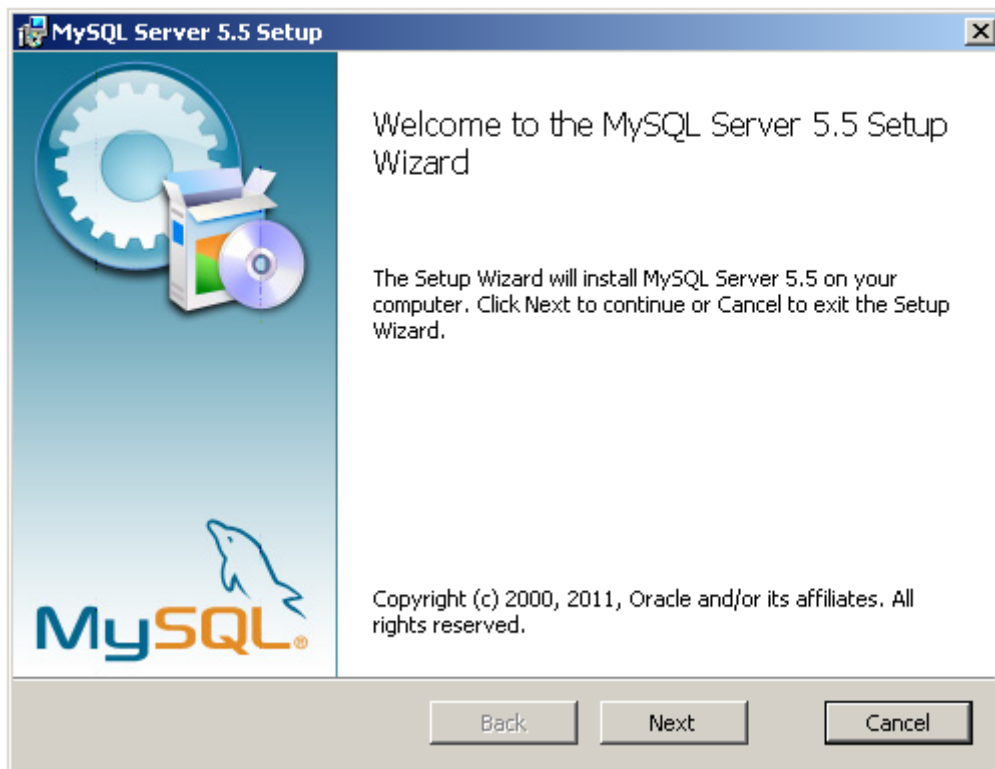
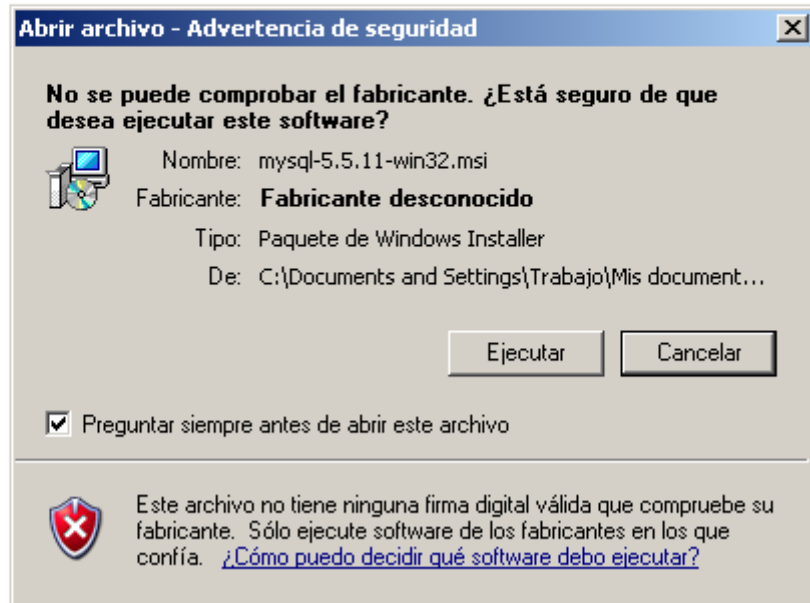
jdk-6u24-windows-i586



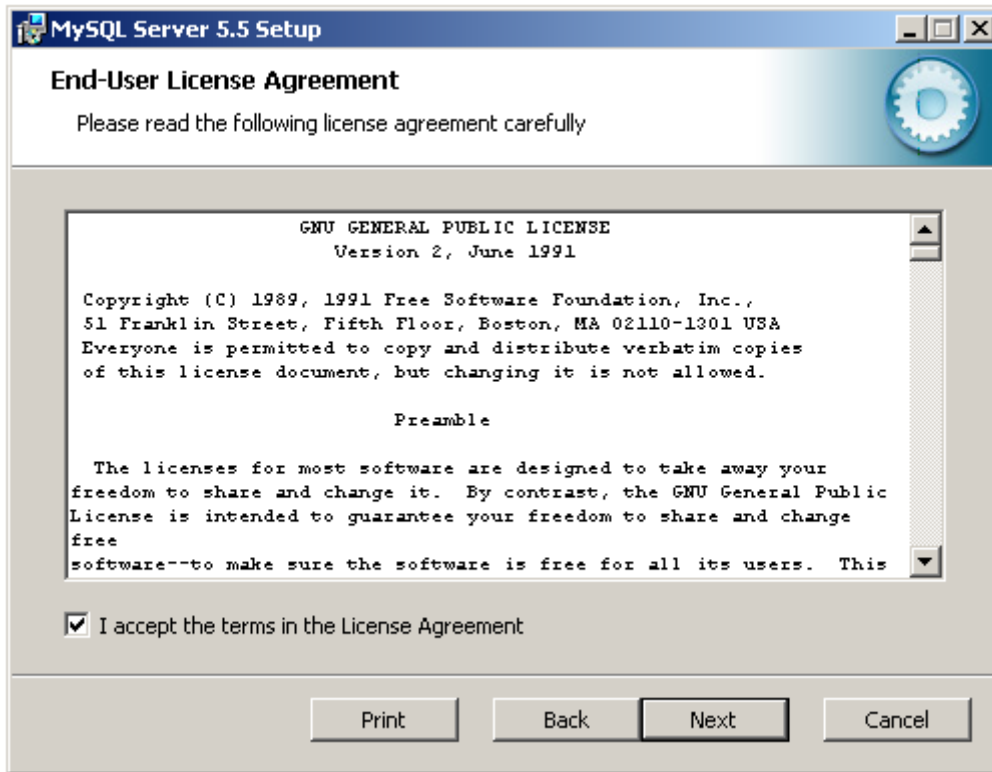
mysql-5.5.11-win32
Paquete de Windows Installer
28.318 KB



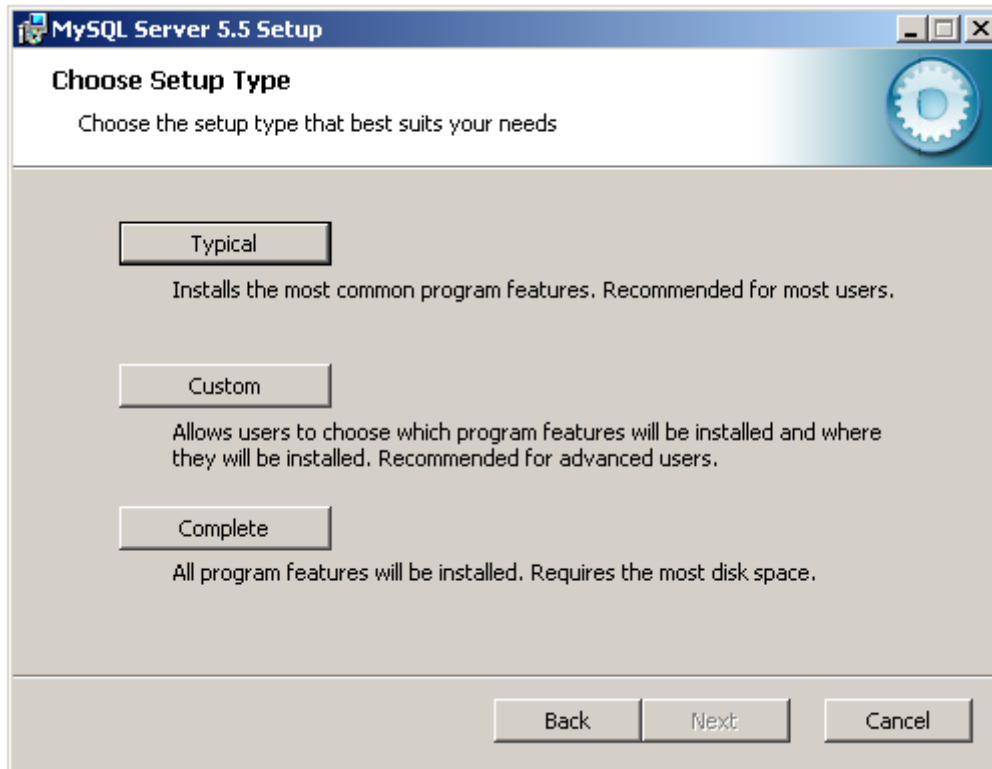
my
Pa
25



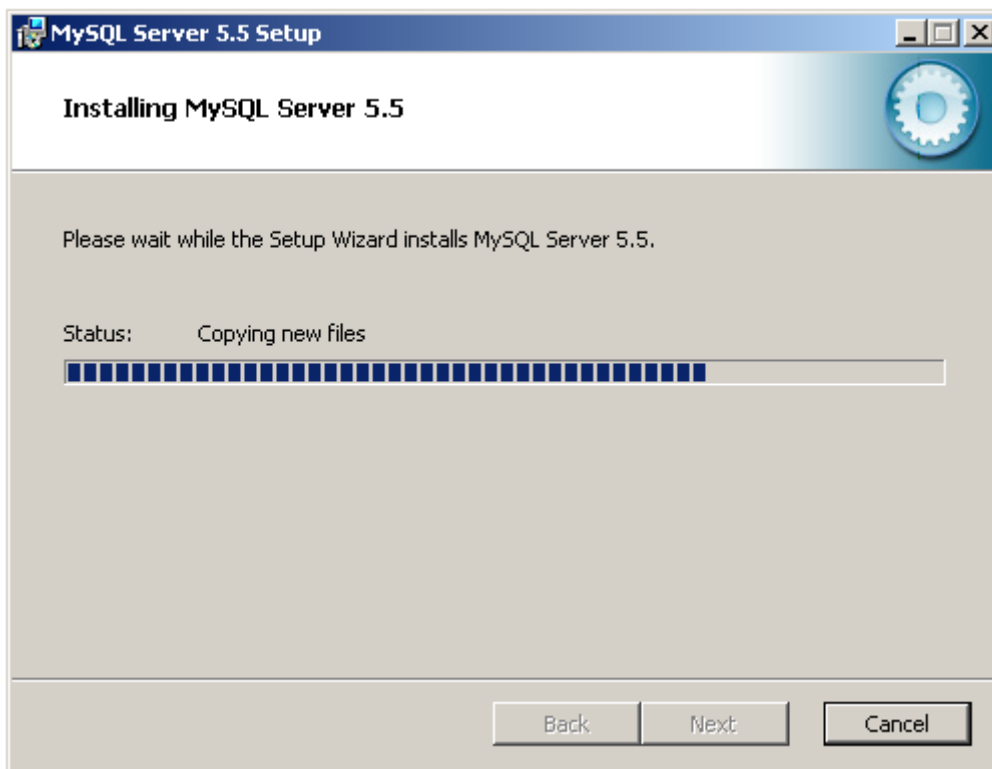
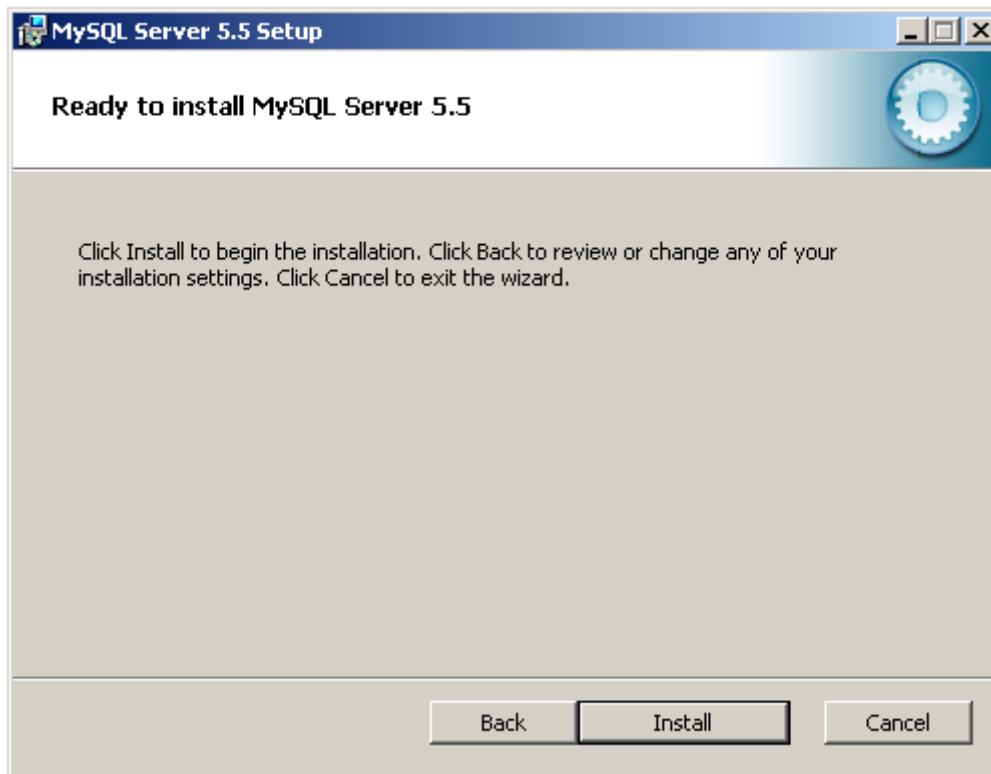
Aceptamos los términos

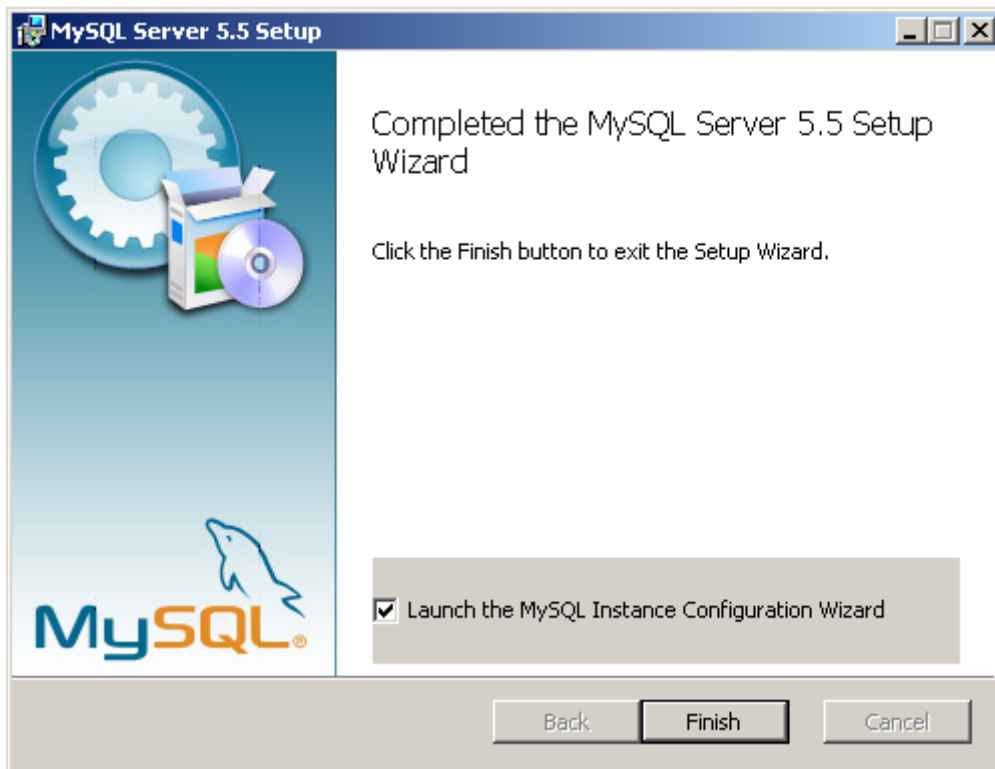
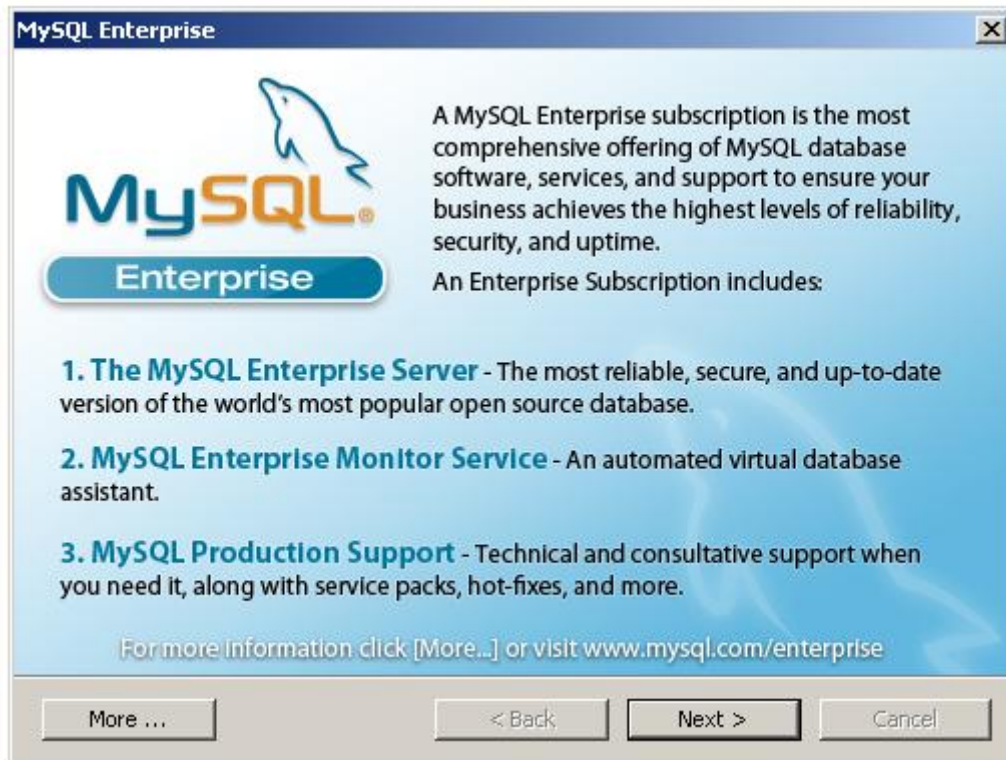


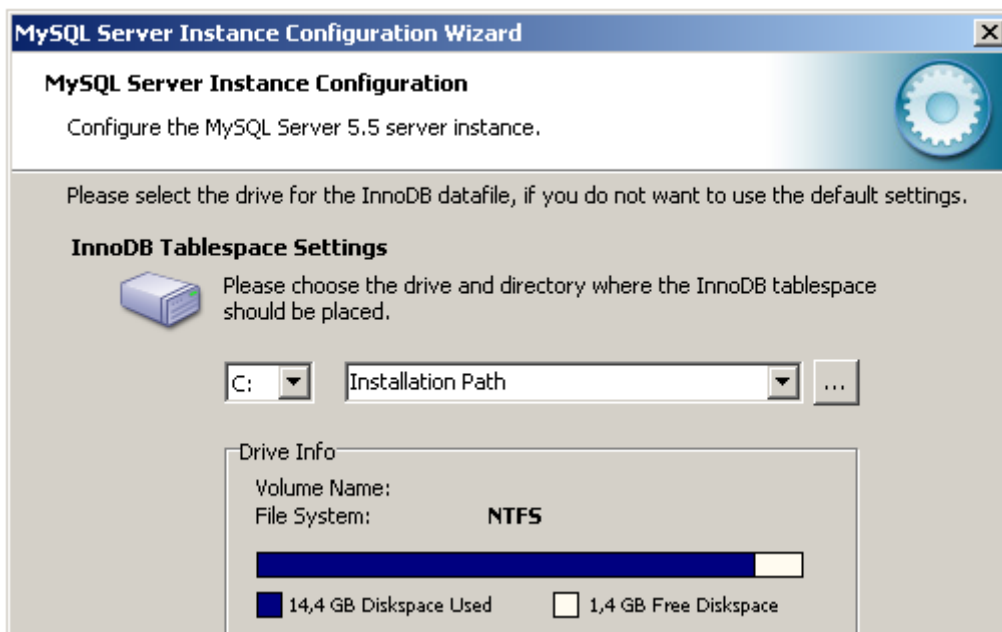
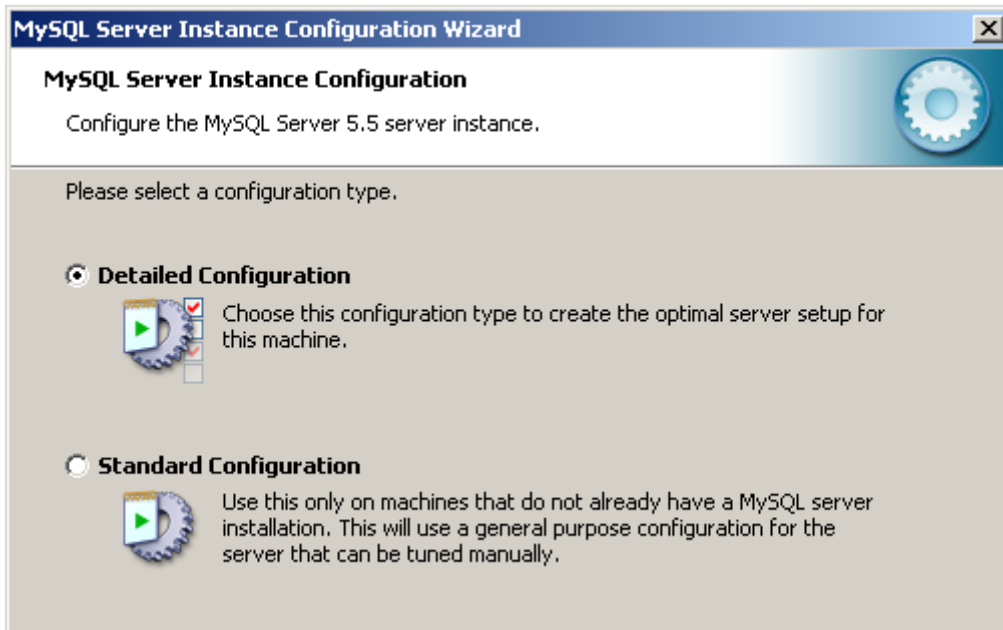
Seleccionamos la instalación completa



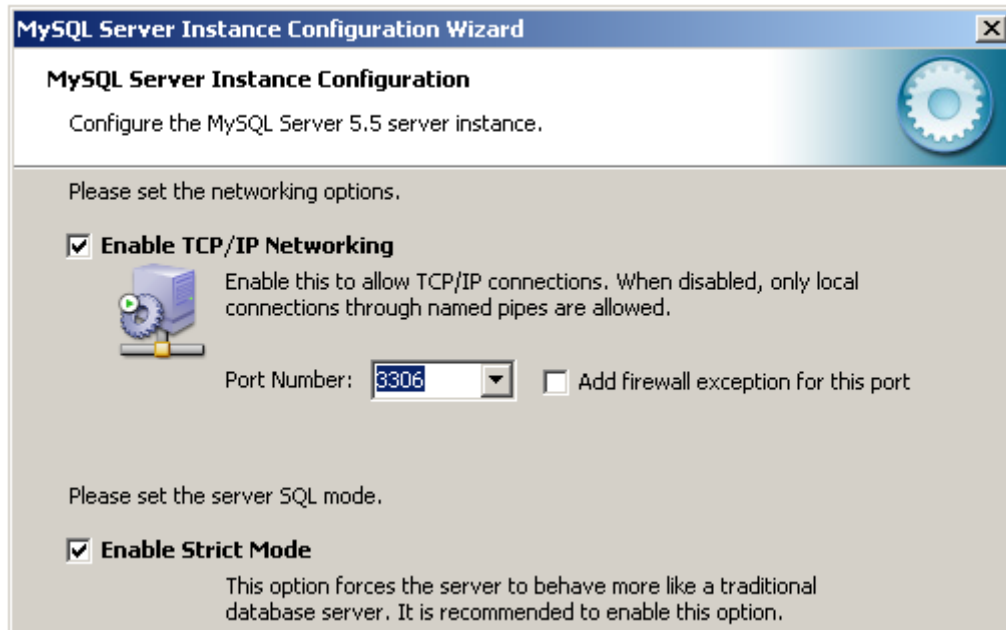
Seleccionamos la opción de instalar



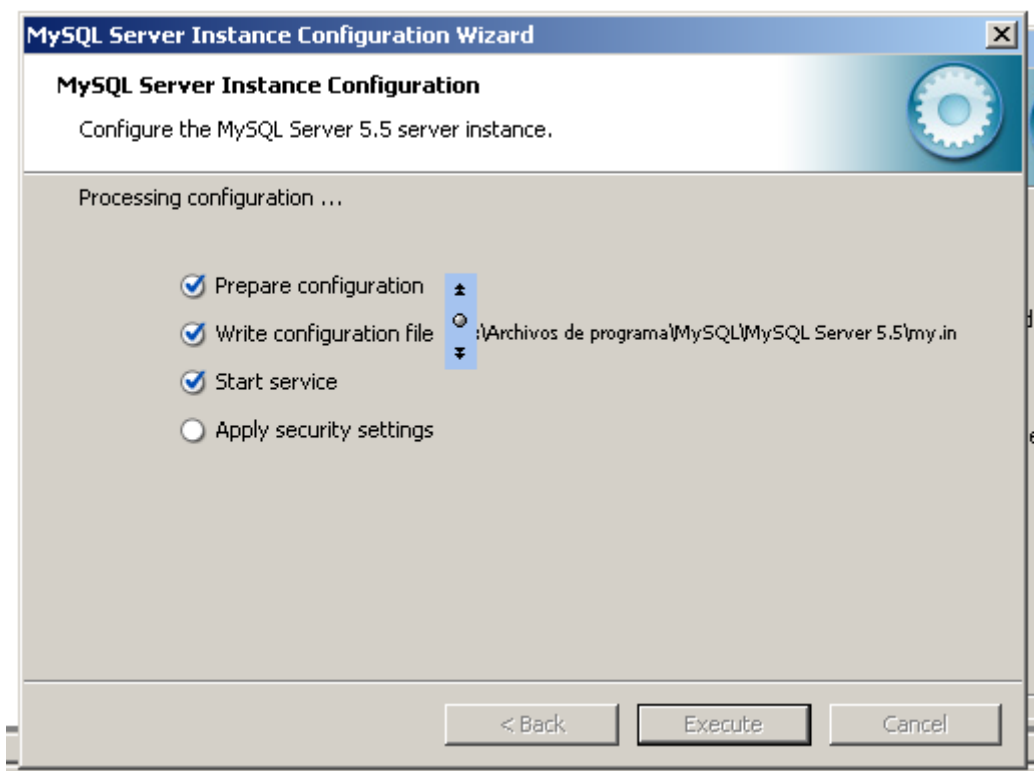
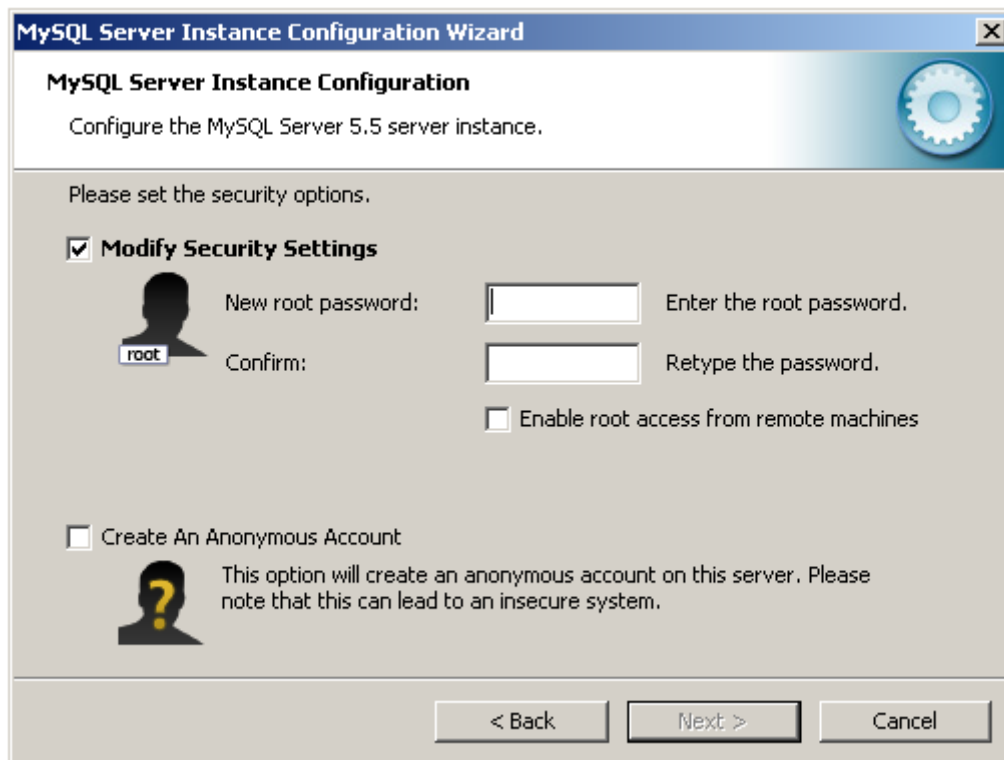




En esta parte seleccionamos el puerto.



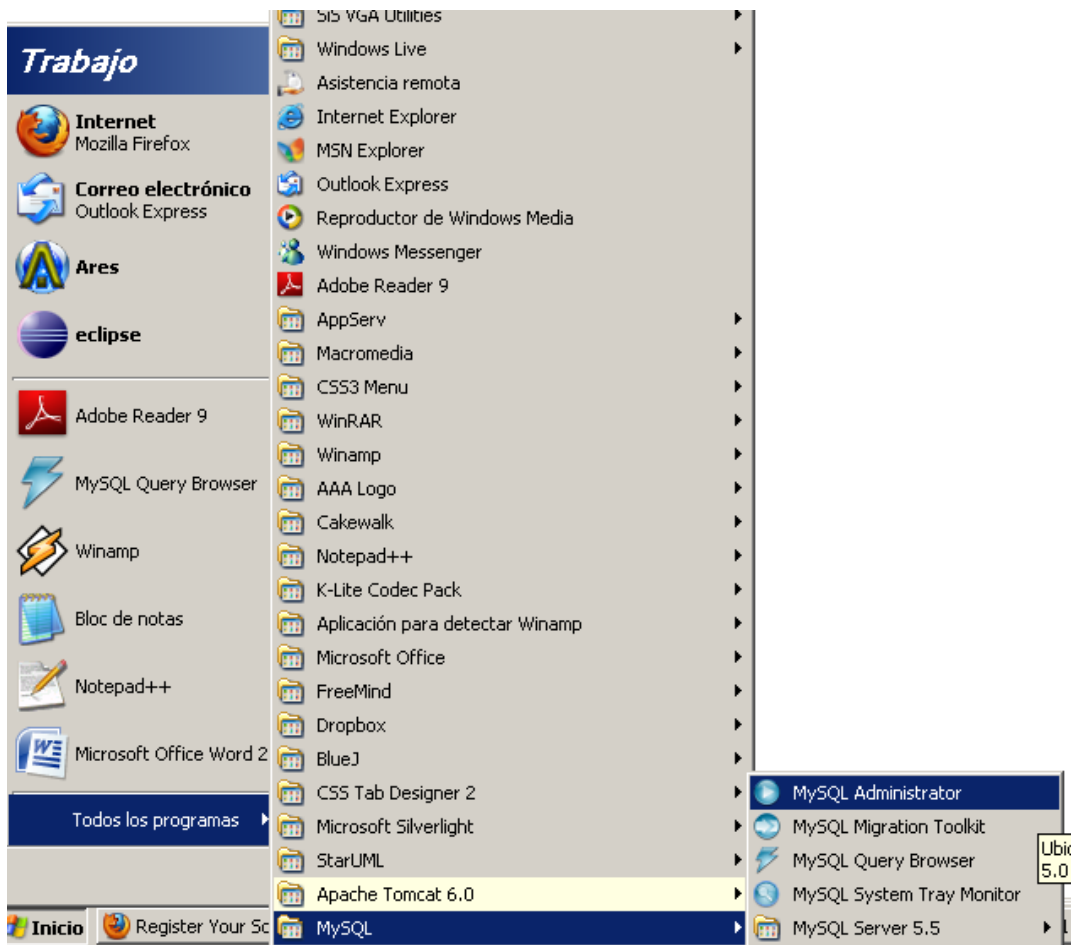
Introducimos la contraseña para el administrador



Aquí se termina de instalar el manejador de base de datos mysql .

Restauración de la base de datos

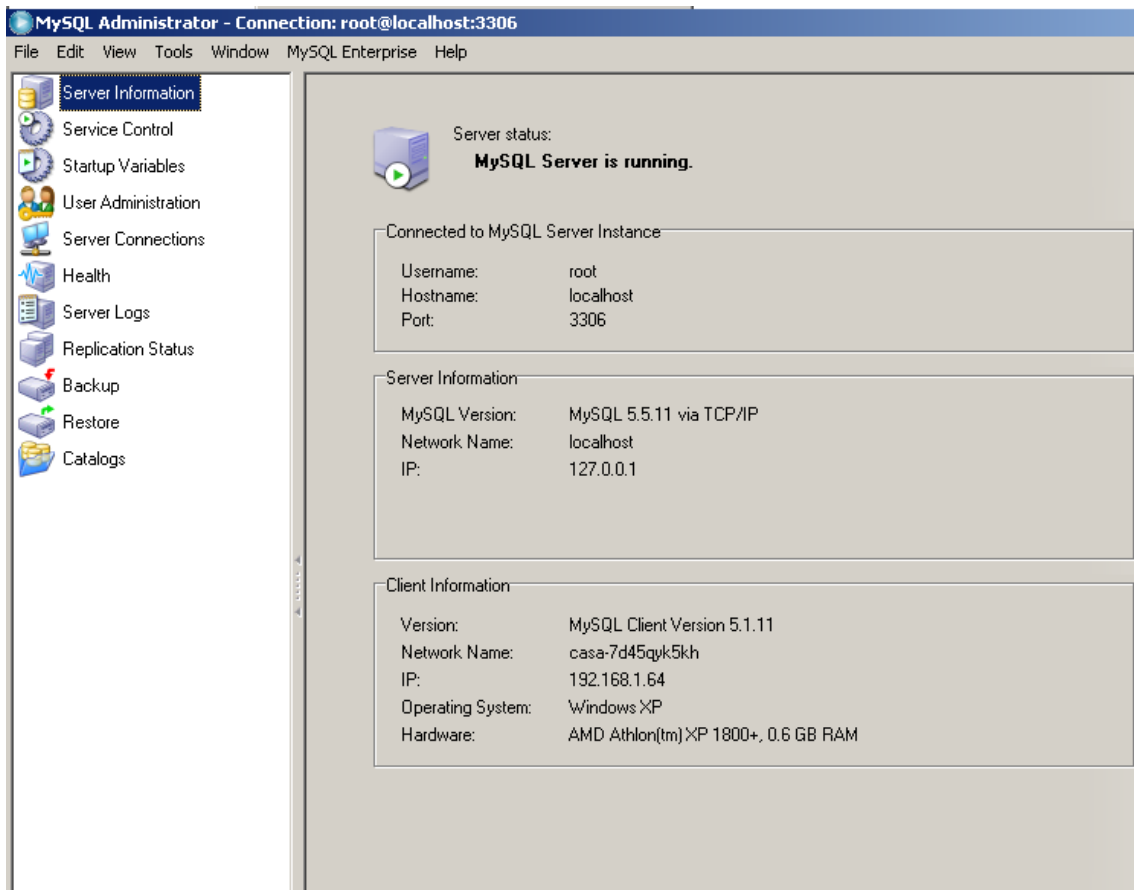
Una vez instalado mysql-server ejecutamos el MySQL-Administrator



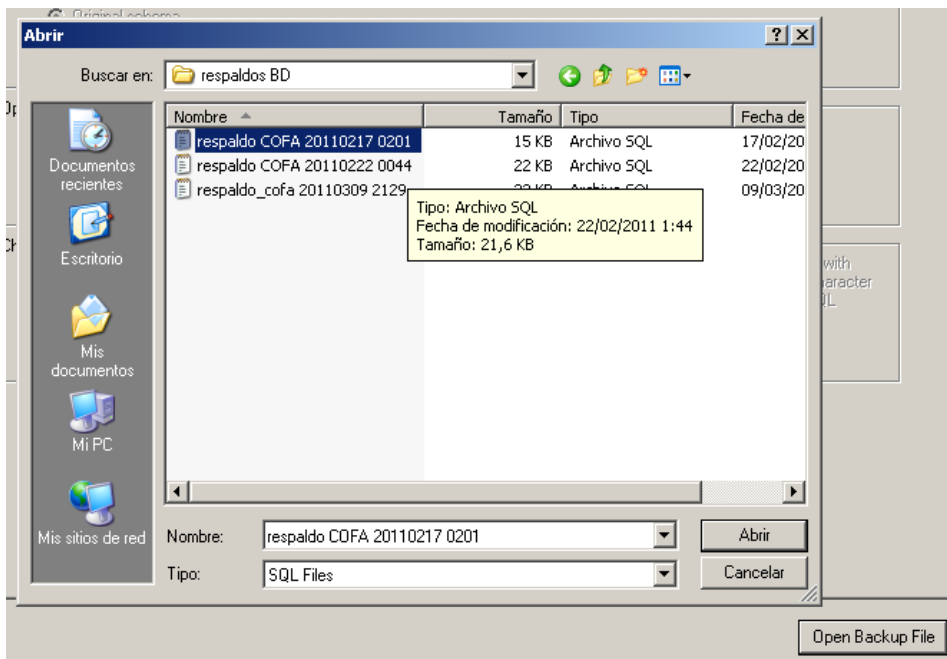
Introducimos el password ingresado al momento de la instalación.



Seleccionamos la opción Restore



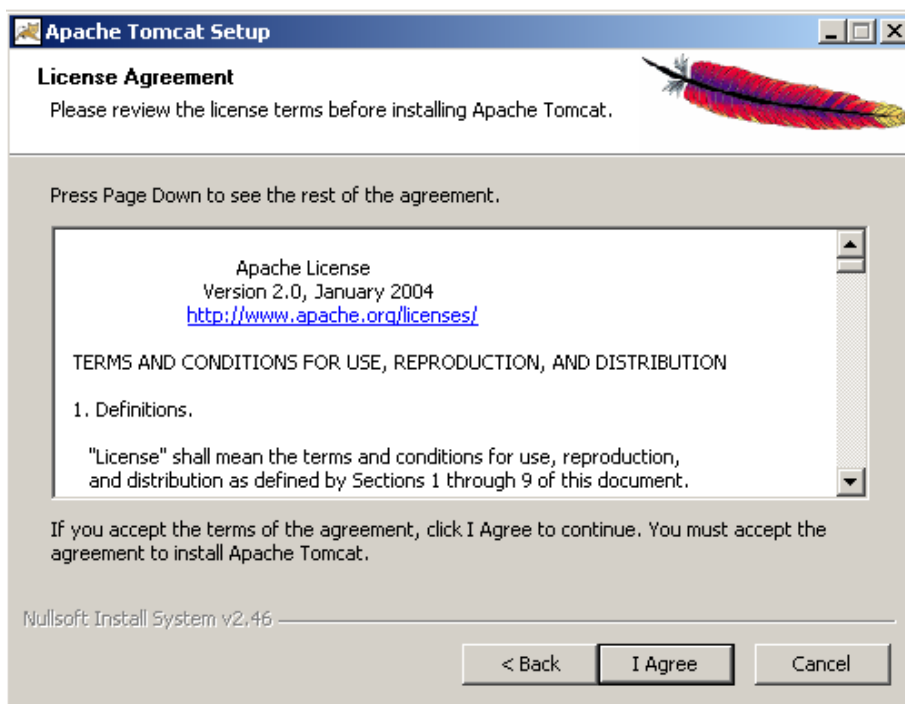
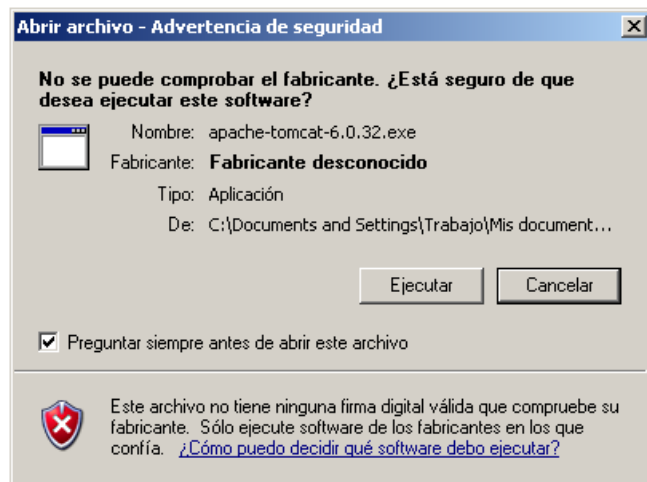
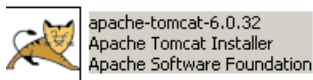
Abrimos el archivo SQL que contiene la declaración de la base de datos.



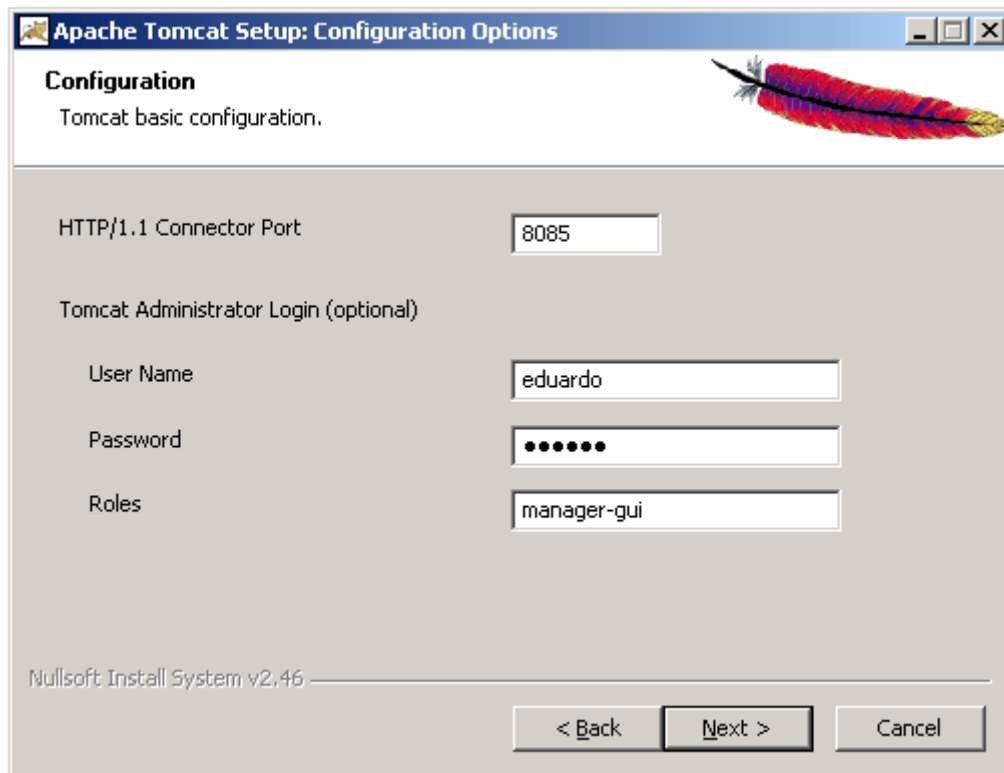
Después elegimos la opción start backup. Aquí terminan los pasos para la creación de la base de datos

Instalación servidor tomcat

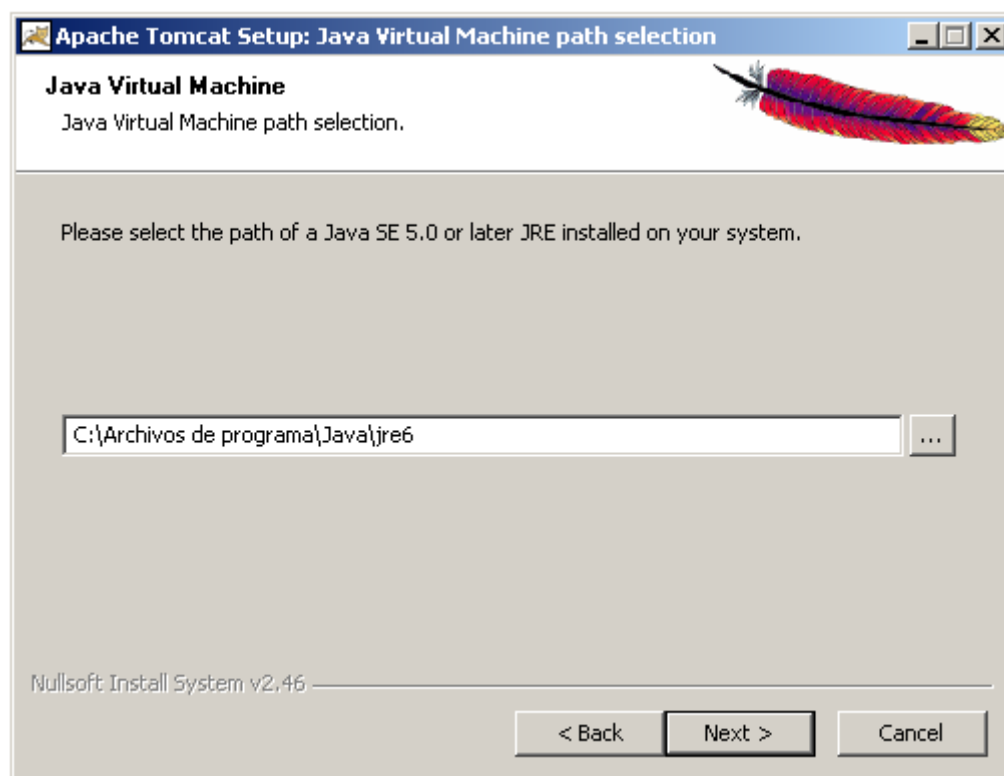
Descargamos el servidor de aplicaciones tomcat-6.0. Una vez descargado lo ejecutamos



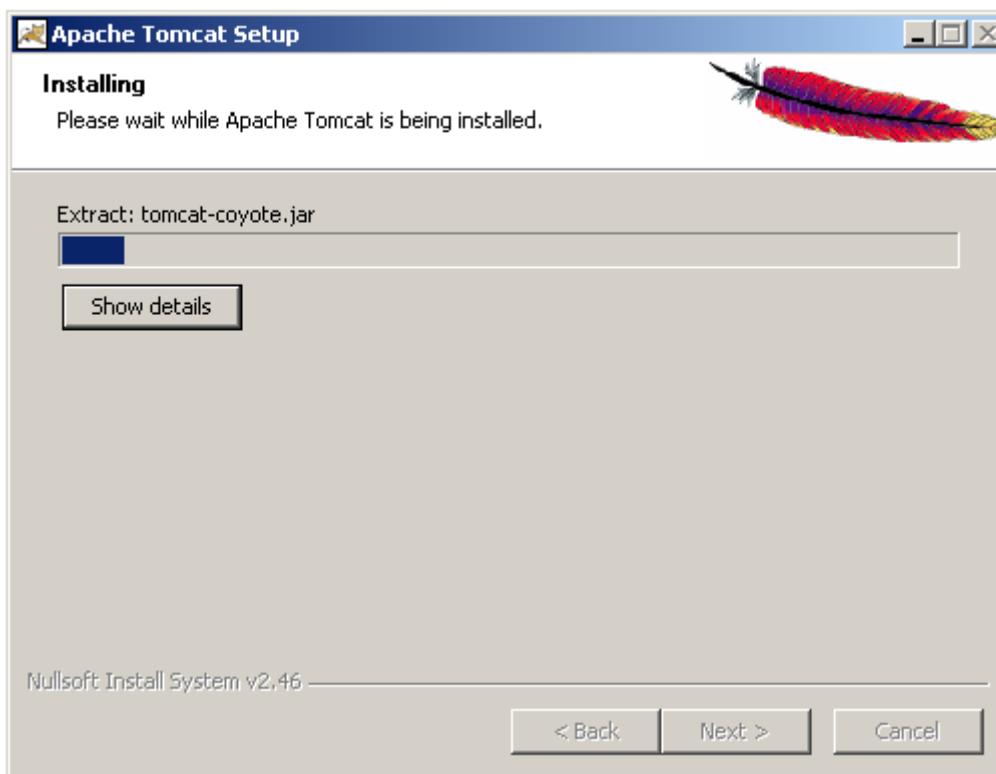
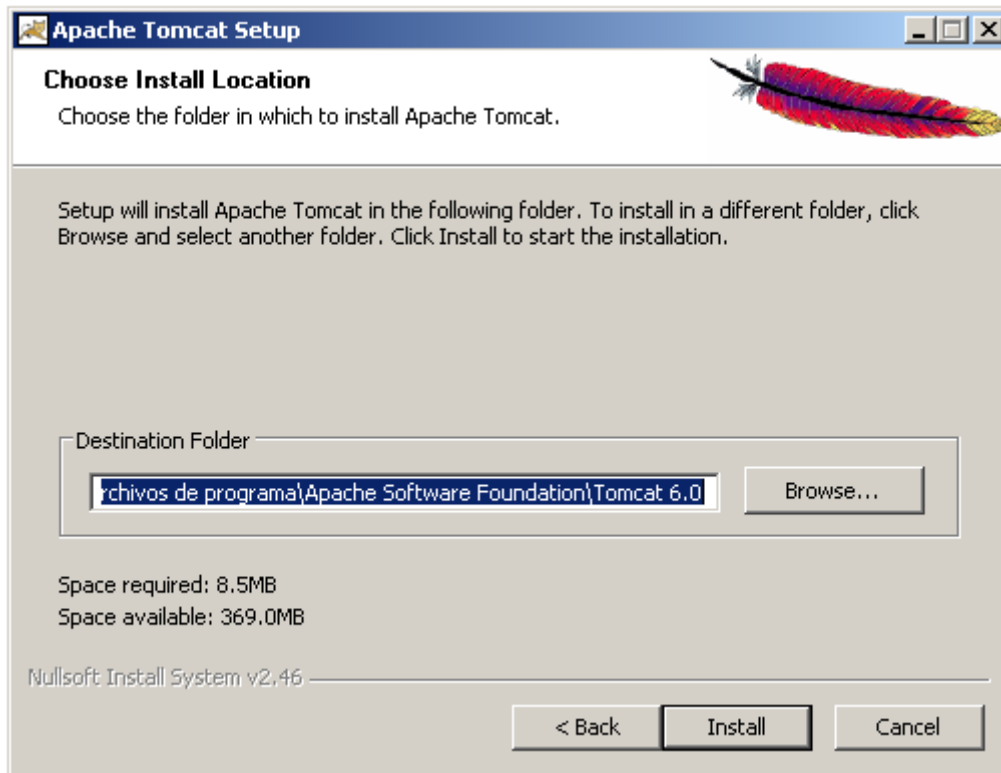
Aquí introducimos el nombre de usuario y password de administrador.

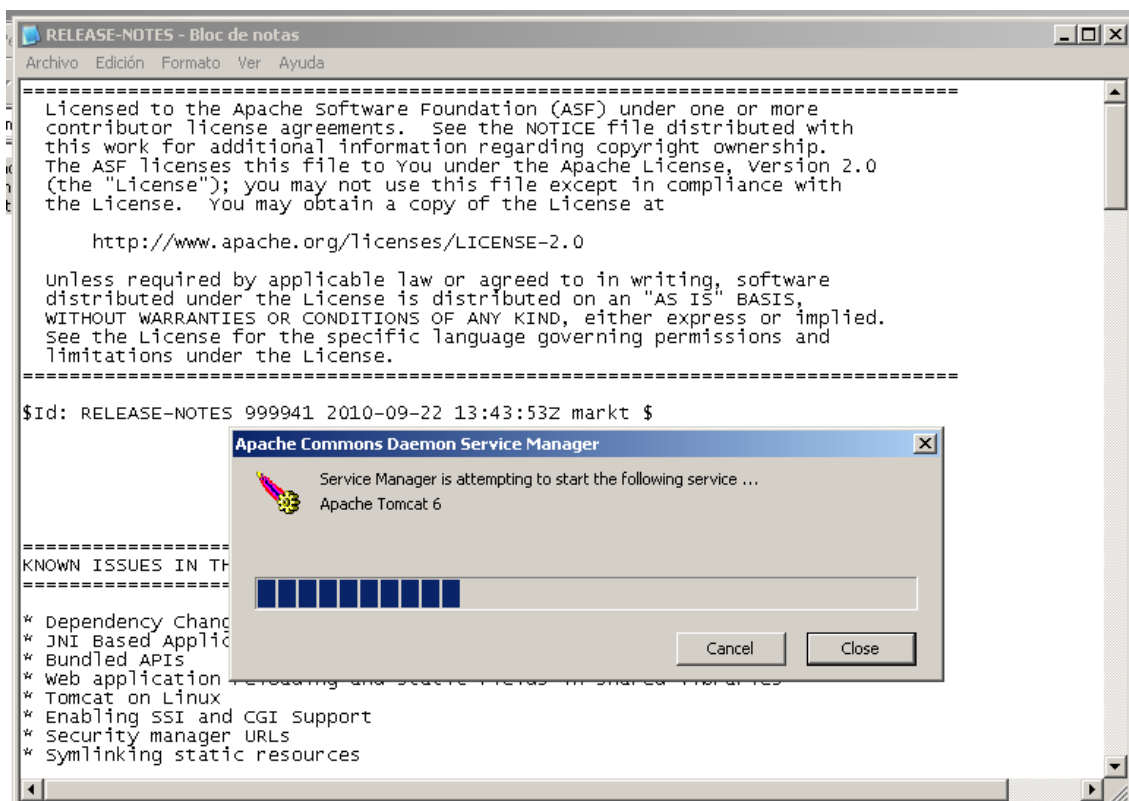
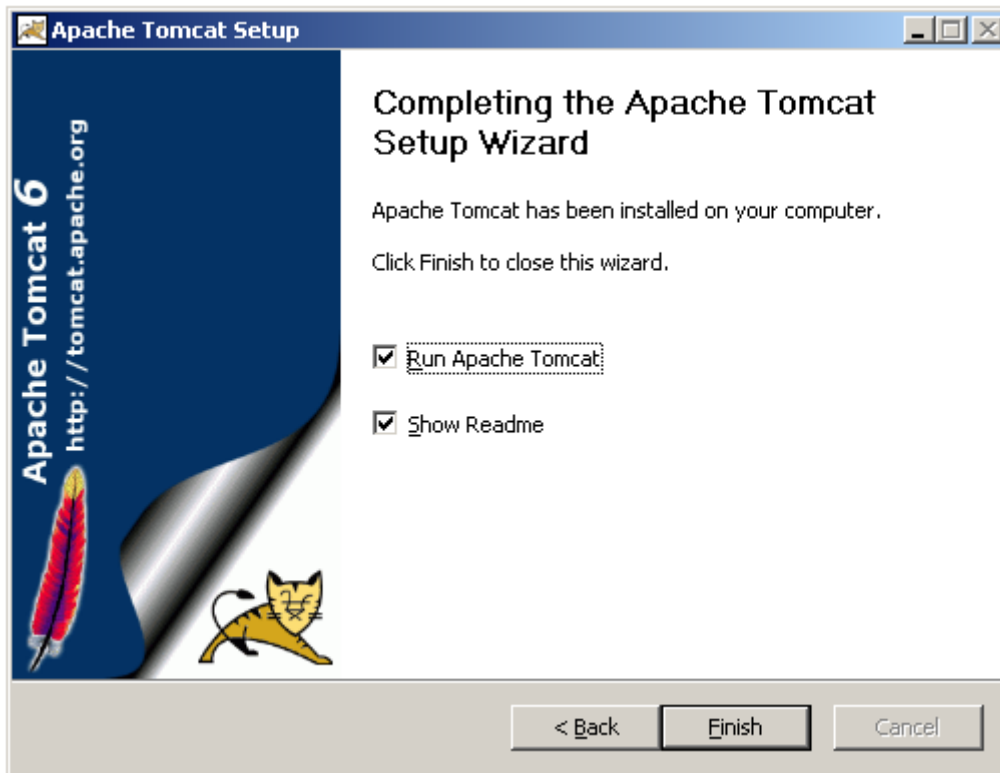


Aquí seleccionamos la tura del JDK que instalamos en la primera parte



Seleccionamos la ruta de instalación de tomcat



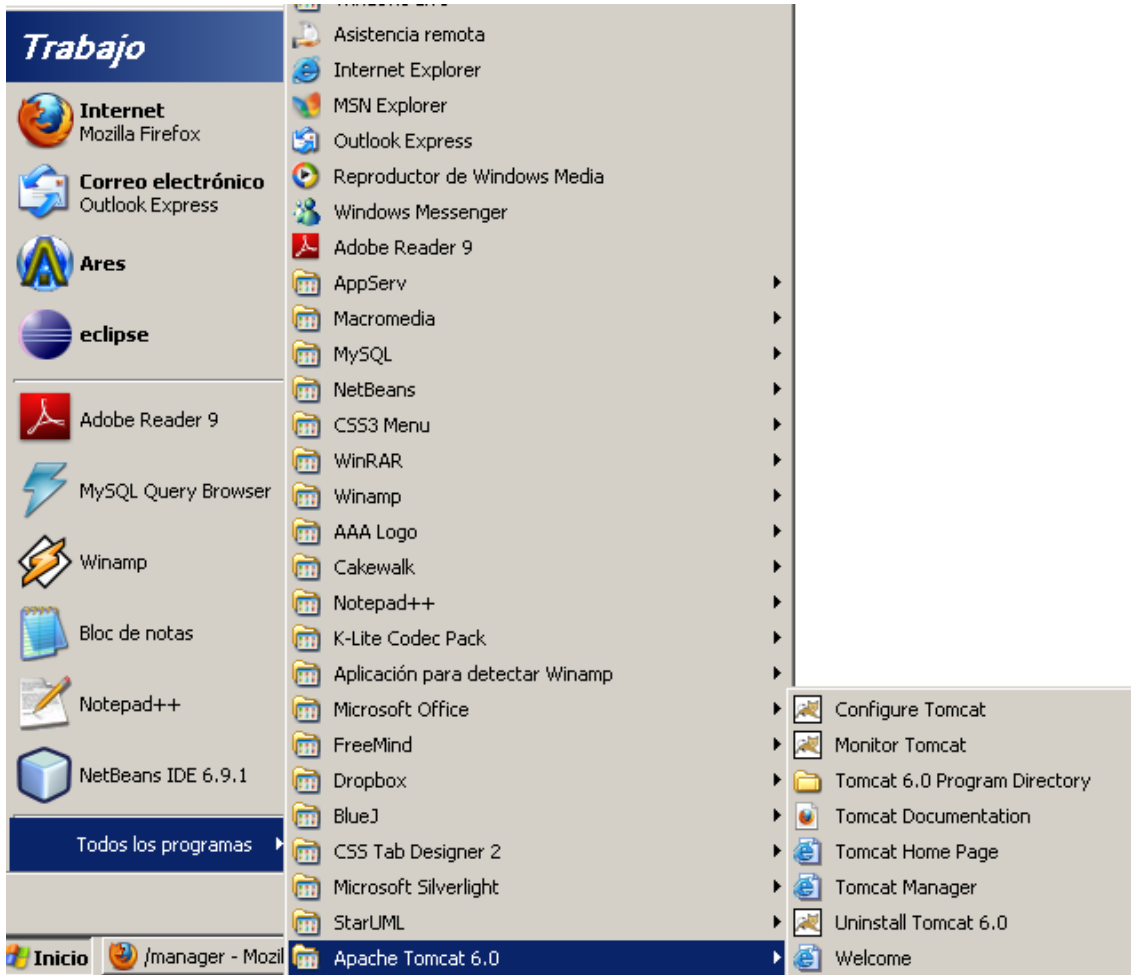


En esta parte tomcat se termina de instalar.

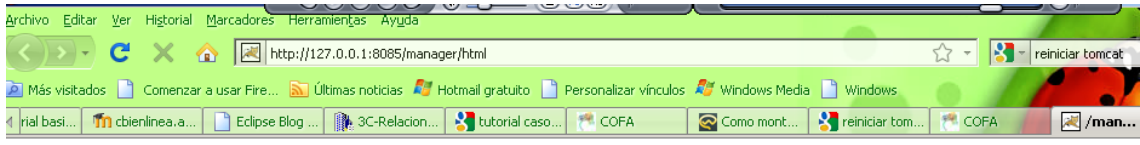
Montar la aplicación web

En la ruta donde se instaló tomcat entramos a la carpeta webapps y pegamos dentro de ella el archivo COFA.war.

Abrimos el tomcat manager



Una vez abierto el gestor de aplicaciones de tomcat, vemos las aplicaciones que están actualmente en el servidor, en donde está nuestra aplicación (COFA) seleccionamos la opción de arrancar.



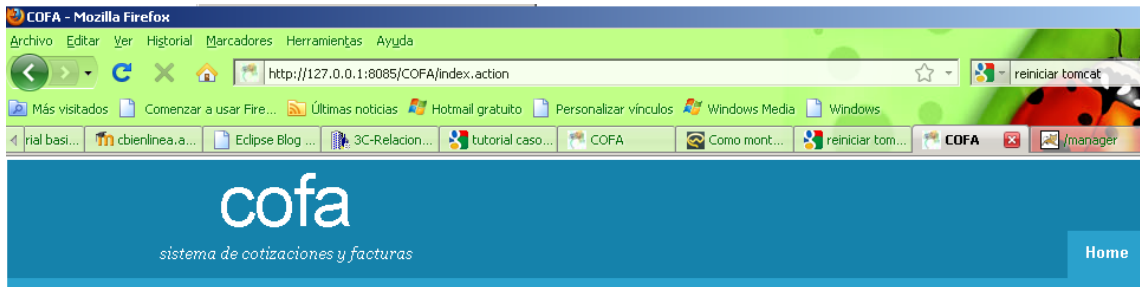
Gestor de Aplicaciones Web de Tomcat

Mensaje:	OK
----------	----

Gestor			
Listar Aplicaciones	Ayuda HTML de Gestor	Ayuda de Gestor	Estado de S...

Aplicaciones				
Trayectoria	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Welcome to Tomcat	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/COFA	Aplicación	true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/docs	Tomcat Documentation	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos

Para acceder a la aplicación introducimos el siguiente URL en el navegador y listo .



Bienvenido a cofa

	Usuario	<input type="text"/>
	Password	<input type="password"/>

Ligas de descarga

- TOMCAT - <http://tomcat.apache.org/download-60.cgi>
- MySQL-server - <http://www.mysql.com/downloads/mirror.php?id=401954>
- JDK - https://cds.sun.com/is-bin/INTERSHOP.enfinity/WFS/CDS-CDS_Developer-Site/en_US/-/USD/ViewProductDetail-Start?ProductRef=jdk-6u24-oth-JPR@CDS-CDS_Developer

Ligas consultadas el 12/04/2011

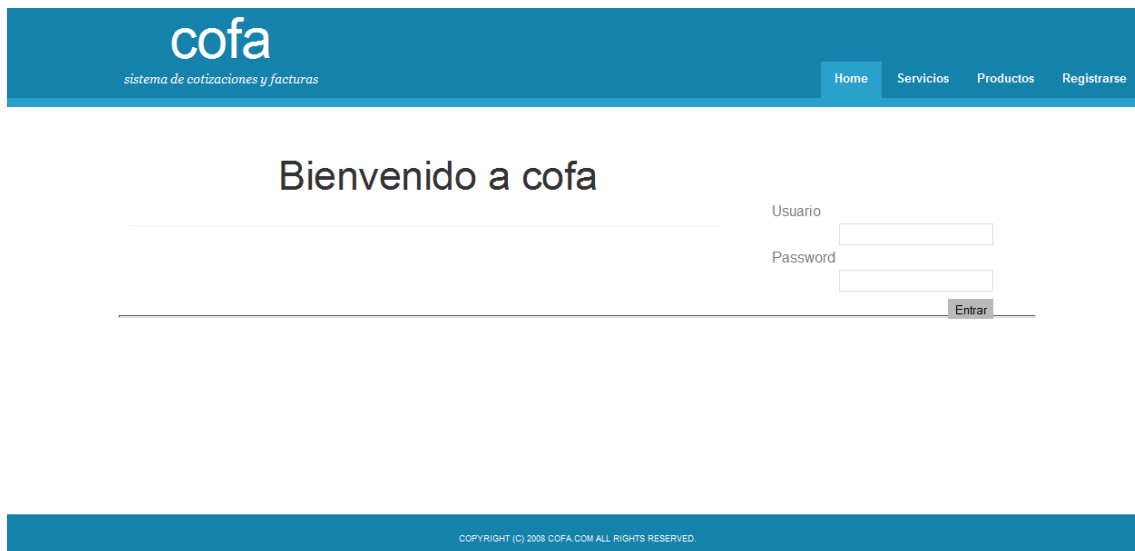
Manual de usuario

Manual del usuario	2
Ingresar al sistema	2
Registro de nuevo usuario	3
Consultar productos.....	6
Generar cotización	7
Consultar cotizaciones	9
Ver cotización a detalle.....	10
Cerrar cotización	10
Cancelar cotización	11
Consultar mis facturas.....	11
Ver detalle de factura.....	12
Descargar factura	13
Manual usuario administrador	14
Ingresar al sistema	14
Consultar cotizaciones	15
Administración de facturas	15
Generar facturas	16
Ver detalle de factura.....	17
Descargar factura	18
Administración de productos.....	18
Agregar producto	19
Modificar producto	20
Borrar producto.....	21

Manual del usuario

Ingresar al sistema

En esta pantalla introducirá sus datos de acceso con los que se registro, si no está registrado consulte la parte de registro de nuevo usuario.



cofa
sistema de cotizaciones y facturas

Home Servicios Productos Registrarse

Bienvenido a cofa

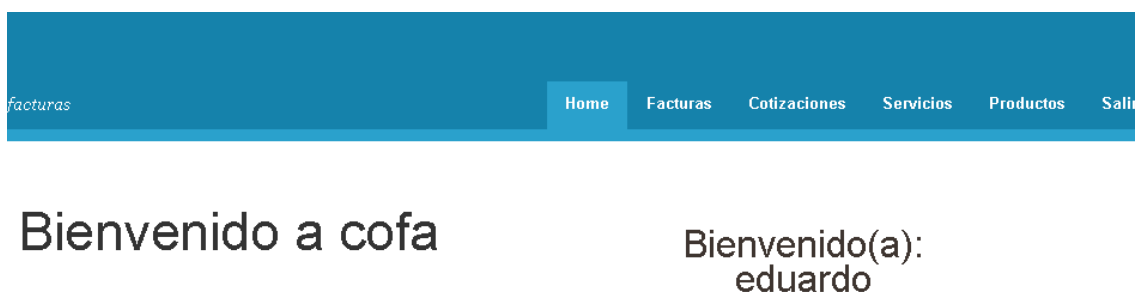
Usuario

Password

Entrar

COPYRIGHT (C) 2008 COFA.COM ALL RIGHTS RESERVED.

Si sus datos de acceso son correctos se desplegará la siguiente pantalla, desplegando el menú para usuarios registrados.



facturas

Home Facturas Cotizaciones Servicios Productos Salir

Bienvenido a cofa

Bienvenido(a):
eduardo

Si los datos introducidos no son válidos se mostrara la siguiente pantalla, permitiéndole el acceso hasta que introduzca datos de acceso válidos.

Ingresa tus datos

Datos inválidos. Favor de verificar

login

password

Enviar

Registro de nuevo usuario

En la pantalla principal deberá seleccionar la opción registrarse.

Bienvenido a cofa

Usuario

Password

Entrar

Al seleccionar la opción de mostrara la siguiente pantalla donde tendrá que ingresar sus datos personal es.

Nombres

Apellido paterno

Apellido materno

Ocupación

Edad

RFC

[Consultar RFC](#)

login

password

Es necesario que introduzca su RFC si no cuenta con el acceda a la opción **consultar RFC** de la pantalla anterior y se mostrará en otra pestaña la pantalla siguiente donde podrá consultar su RFC.



Calcular RFC y CURP

Apellido paterno:

Apellido materno:

Nombre(s):

Fecha de nacimiento (dd/mm/aaaa): 01 ENE 1901

Entidad Federativa de nacimiento: AGUASCALIENTES

Sexo: Hombre

RFC: RARY8608165F2

CURP: RARY860816MJCNT07

Una vez introducidos sus datos personales es momento de ingresar las formas en que se le podrá contactar. Nota el correo electrónico y el teléfono fijo son indispensables.

Correo

Teléfono casa

Teléfono celular

Teléfono trabajo

Fax

Ahora deberá ingresar su dirección.

Calle

Número interior

Número exterior

Colonia

Código postal

Ciudad

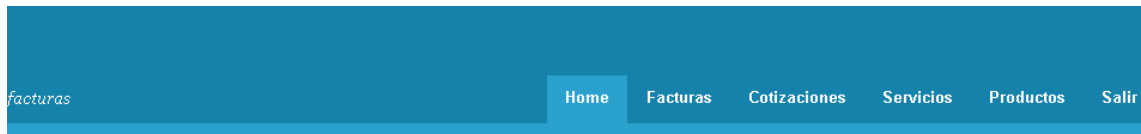
Una vez que haya registrado todos sus datos se desplegar la siguiente pantalla. Nota el registro debe realizarse de manera completa, de lo contrario ningún dato se almacenará en nuestra base de datos.

Gracias por registrarte.
Tu registro ha finalizado
correctamente.

[Accede aqui para continuar](#)

Consultar productos

Para consultar los productos existentes en el menú principal selecciona la opción productos.



Bienvenido a cofa

Bienvenido(a):
eduardo

Se desplegará la siguiente pantalla, la cual cuenta con un menú para búsqueda especializada de productos, basta con dar click a cualquiera de las categorías para la búsqueda especializada.



Generación de cotizaciones

Ver mis cotizaciones

Celulares y accesorios Computación Consolas y videojuegos Electrónica Audio Video Hogar

Celulares y accesorios
Tarjetas de memoria
Teléfonos celulares

Lista productos

imagen	Nombre	Marca	Precio	clasificación	Agregar
[B@17c3056	memoria 8 GB	1	\$ 252.0	1	✓
[B@8cd4db	Nokia N900	8	\$ 5320.0	2	✓
	Celular Nokia C3	8	\$ 1800.0	2	✓
	Celular Sony Ericsson W705	9	\$ 2597.0	2	✓
	Sony Ericsson U8	9	\$ 4199.0	2	✓
	Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	✓
	Lap Top Dell Latitude D620 Intel Core Duo Wifi	3	\$ 3000.0	3	✓

Generar cotización

Para generar una cotización acceda a la opción cotizaciones.



Bienvenido a cofa

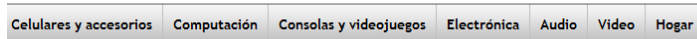
Bienvenido(a):
eduardo

Se desplegara una pantalla donde podrá seleccionar los productos que desee agregar al carrito de compras. Para agregar un producto al carrito basta con dar click en cualquiera de las palomas verdes de la tabla de productos.



Generación de cotizaciones

Ver mis
Cotizaciones



Lista productos						
Imagen	Nombre	Marca	Precio	clasificación	Agregar	
[B@fb0b0	memoria 8 GB	1	\$ 2220.0	4	✓	
[B@18ff27	Nokia N900	8	\$ 5320.0	2	✓	
	Celular Nokia C3	8	\$ 1800.0	2	✓	
	Celular Nokia C3	8	\$ 1800.0	2	✓	
	Celular Sony Ericsson W705	9	\$ 2597.0	2	✓	
	Sony Ericsson U8	9	\$ 4199.0	2	✓	
	Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	✓	
	Lap Top Dell Latitude D620 Intel Core Duo Wifi	3	\$ 3000.0	3	✓	
[B@b8501d	Mario Land 2 En Excelente Estado	4	\$ 1750.0	8	✓	
[B@93a985	aass	3	\$ 25200.0	2	✓	
[B@6399ae	Transmisor Fm 4 En 1 Coche Usb	2	\$ 4990.0	9	✓	
[B@56d6cf	lavadora	8	\$ 450057.0	4	✓	

Lista de productos a cotizar				
Nombre	Marca	Precio	clasificación	Remove

Seleccionar
numero
de productos

Cuando agregue un producto al carrito de compras se desplegarán en la tabla inferior de la pantalla, para remover el producto basta con dar click en cualquiera de los taches rojos de la tabla. Al seleccionar todos los productos que desee de click en el botón **seleccionar número de productos**.

[B@4ba4d6	lavadora	8	\$ 450057.0	4	<input checked="" type="checkbox"/>
-----------	----------	---	-------------	---	-------------------------------------

Lista de productos a cotizar				
Nombre	Marca	Precio	clasificación	Remover
Celular Sony Ericsson W705	9	\$ 2597.0	2	<input checked="" type="checkbox"/>
Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	<input checked="" type="checkbox"/>
memoria 8 GB	1	\$ 2220.0	4	<input checked="" type="checkbox"/>
Nokia N900	8	\$ 5320.0	2	<input checked="" type="checkbox"/>

Seleccionar número de productos

Se desplegara la siguiente pantalla donde deberá introducir la cantidad de productos que desee de cada tipo y después dar click en el botón enviar.

cofa
sistema de cotizaciones y facturas

Home Mis Facturas Cotizaciones Servicios Productos Salir

Selecciona cuantos productos deseas		
Nombre	Precio	cantidad
Celular Sony Ericsson W705	\$ 2597.0	<input type="text" value="2"/>
Lap Top Toshiba 3 Gb Ram C645d	\$ 6898.0	<input type="text" value="4"/>
memoria 8 GB	\$ 2220.0	<input type="text" value="2"/>
Nokia N900	\$ 5320.0	<input type="text" value="5"/>

Se desplegará la siguiente pantalla donde se mostrará la cotización realizada en base a los productos seleccionados, en esta parte tendrá la opción de guardar la cotización si lo desea. Nota para generar una factura se debe guardar la cotización ya demás que la cotización este cerrada.

Cotización					
Nombre	Precio unitario	Cantidad	IVA	Importe	Total
Celular Sony Ericsson W705	\$ 2597.0	2	\$ 831.04	\$ 4362.96	\$ 5194.0
Lap Top Toshiba 3 Gb Ram C645d	\$ 6898.0	4	\$ 4414.7197	\$ 23177.281	\$ 27592.0
memoria 8 GB	\$ 2220.0	2	\$ 710.39996	\$ 3729.6	\$ 4440.0
Nokia N900	\$ 5320.0	5	\$ 4256.0	\$ 22344.0	\$ 26600.0

Totales		
Importe	IVA	Total
\$53613.844	\$10212.16	\$63826.004

Guardar Cotización

Si selecciona la opción de guardar cotización de desplegara la siguiente pantalla.

Cotizaciones Pendientes Cotizaciones Cerradas

Mis cotizaciones				
Id	Fecha de cotización	Importe parcial	IVA	Importe total
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004

Consultar cotizaciones

Para consultar las cotizaciones realizadas en el menú principal acceda a la opción **cotizaciones** y después **ver mis cotizaciones**

Bienvenido a cofa

Bienvenido(a):
eduardo

Se desplegara la siguiente pantalla donde podrá consultar sus cotizaciones realizadas.

The screenshot shows the COFA system interface. At the top, there is a navigation bar with the COFA logo and the tagline "sistema de cotizaciones y facturas". The navigation menu includes "Home", "Mis Facturas", "Cotizaciones", "Servicios", "Productos", and "Salir". Below the navigation bar, the text "Mis cotizaciones registradas" is displayed. There are two buttons: "Cotizaciones Pendientes" and "Cotizaciones Cerradas". Below these buttons is a table titled "Mis cotizaciones" with the following data:

Id	Fecha de cotización	Importe parcial	IVA	Importe total	Cerrar cotización	Cancelar cotización
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	✓	✗

Ver cotización a detalle

Para ver una cotización a detalle, en la pantalla ver cotizaciones dar click en el símbolo +/-, se desplegaran los detalles de la cotización, para ocultarlo basta con dar click de nuevo en el símbolo +/-.

The screenshot shows the COFA system interface with the "Cotizaciones Cerradas" button selected. The table "Mis cotizaciones" is displayed with the following data:

Id	Fecha de cotización	Importe parcial	IVA	Importe total	Cancelar cotización
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	✗
Insumos					
		\$ 1847.52	\$ 11547.0		✗

Below the main table, a detailed view of the quotation items is shown in a scrollable window:

lucto	Precio Unitario	Cantidad	Iva	Importe	Total
	\$ 2597.0	2	\$ 831.04	\$ 4362.96	\$ 5194.0
	\$ 6898.0	4	\$ 4414.7197	\$ 23177.281	\$ 27592.0
	\$ 2220.0	2	\$ 710.39996	\$ 3729.6	\$ 4440.0
	\$ 5320.0	5	\$ 4256.0	\$ 22344.0	\$ 26600.0
Total			\$ 10212.16	\$ 53613.844	\$ 63826.00

Cerrar cotización

En la pantalla ver cotizaciones seleccione la opción cotizaciones pendientes y se desplegara la siguiente pantalla.

Mis cotizaciones registradas

Cotizaciones
PendientesCotizaciones
Cerradas

Mis cotizaciones						
Id	Fecha de cotización	Importe parcial	IVA	Importe total	Cerrar cotización	Cancelar cotización
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	✓	✗

Para cerrar la cotización basta con dar click en la paloma verde de la cotización. Nota , para iniciar el proceso de facturación la cotización deberá estar cerrada.

Cancelar cotización

Para cancelar la cotización basta con dar click en la tache rojo de la cotización

Mis cotizaciones registradas

Cotizaciones
PendientesCotizaciones
Cerradas

Mis cotizaciones						
Id	Fecha de cotización	Importe parcial	IVA	Importe total	Cerrar cotización	Cancelar cotización
5 +/-	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	✓	✗

Consultar mis facturas

Para consultar la facturas se deberá seleccionar en el menú principal la opción facturas

Bienvenido a cofa

Bienvenido(a):
eduardo

Se desplegara la siguiente pantalla donde podrá consultar sus facturas

cofa

sistema de cotizaciones y facturas

Home

Mis Facturas

Cotizaciones

Servicios

Productos

Salir

Mis facturas				
Folio	Estado	Fecha de expedición	Fecha límite de pago	Descarga pdf
12314 +/-	1	2011-04-05 11:44:55.0		
12315 +/-	1	2011-04-05 11:44:56.0		

Ver detalle de factura

Para ver los detalles de la factura en la pantalla ver facturas dar click en el símbolo +/-, se desplegaran los detalles de la cotización, para ocultarlo basta con dar click de nuevo en el símbolo +/-.

cofa

sistema de cotizaciones y facturas

Home



Mis Facturas

Cotizaciones

Servicios

Productos

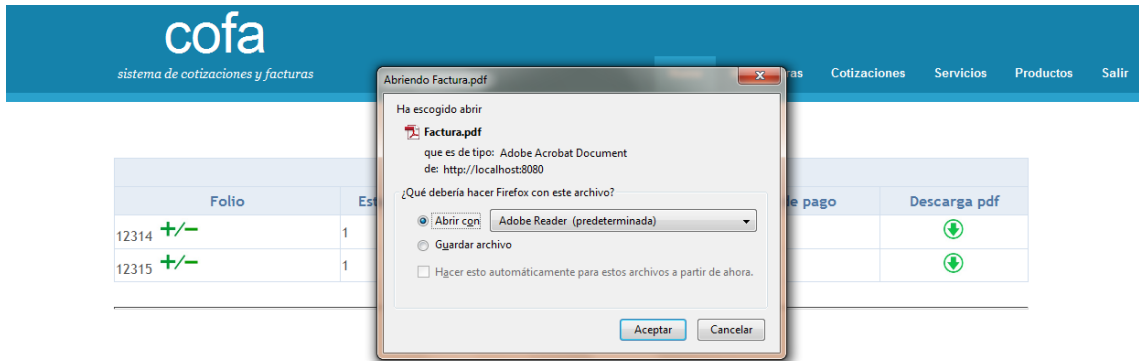
Salir

Mis facturas				
Folio	Estado	Fecha de expedición	Fecha límite de pago	Descarga pdf
12314 +/-	1	2011-04-05 11:44:55.0		
12315 +/-	1	2011-04-05 11:44:56.0		

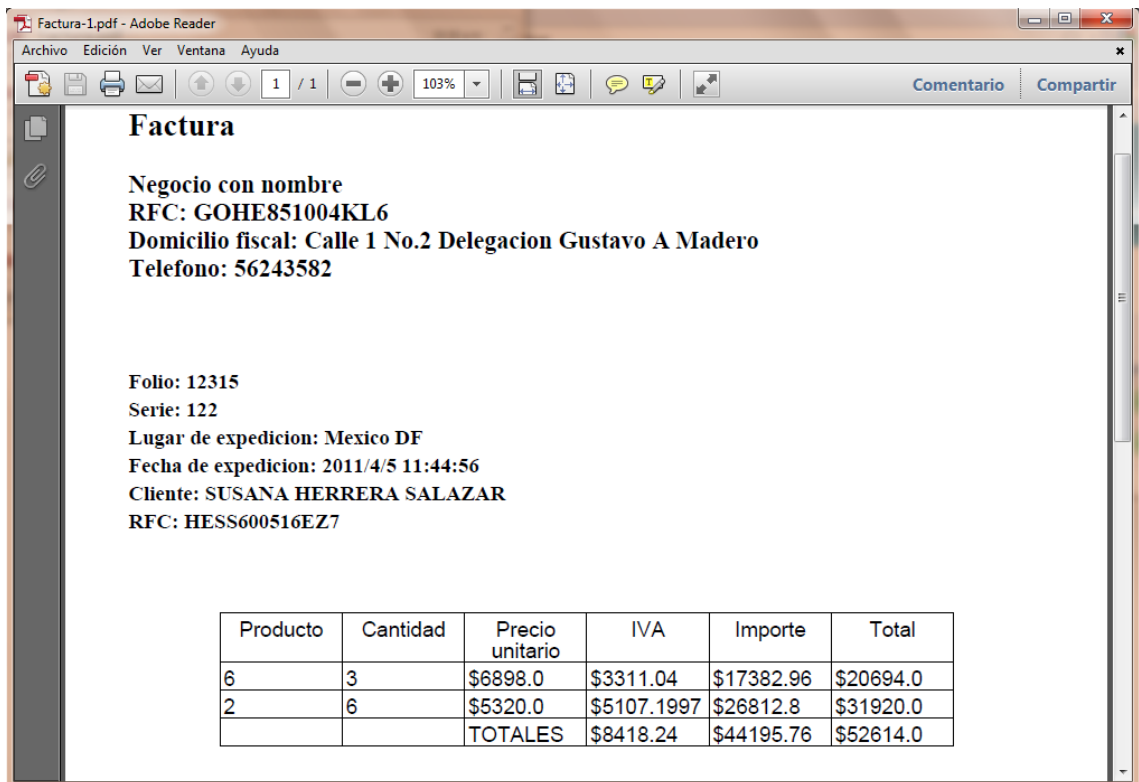
Datos de factura	
RFC	GOHE851004KL6
Domicilio fiscal	Calle 1 No.2 Delegación Gustavo A. Madero
Teléfono	56243582
Folio	12315
Serie	122
Fecha de Expedición	2011-04-05 11:44:56.0

Descargar factura

Para descargar la factura en formato PDF basta con dar click en el icono descargar factura (flecha verde apuntando hacia abajo)



Este es ejemplo de una factura descargada.



Manual usuario administrador

Ingresar al sistema

En esta pantalla introducirá sus datos de acceso con permisos de administrador.



Bienvenido a cofa

Formulario de inicio de sesión con campos para Usuario y Password, y un botón Entrar.

COPYRIGHT (C) 2008 COFA.COM ALL RIGHTS RESERVED.



Bienvenido a cofa

Bienvenido(a):
eduardo



Ingresar tus datos
Datos inválidos. Favor de verificar

Formulario de inicio de sesión con campos para login y password, y un botón Enviar.

Consultar cotizaciones

Para consultar las cotizaciones en el menú principal acceda a la opción cotizaciones.



Bienvenido a cofa

Bienvenido(a):
eduardo



Cotizaciones					
Id	Id Usuario	Fecha de cotización	Importe parcial	IVA	Importe total
7 +/-	9	2011-04-06 15:57:20.0	\$ 22092.002	\$ 4208.0	\$ 26300.002
8 +/-	9	2011-04-06 15:57:52.0	\$ 14784.0	\$ 2816.0	\$ 17600.0

Insumos					
Producto	Precio Unitario	Cantidad	Iva	Importe	Total
	\$ 2220.0	1	\$ 355.19998	\$ 1864.8	\$ 2220.0
	\$ 1800.0	3	\$ 864.0	\$ 4536.0	\$ 5400.0
	\$ 1000.0	2	\$ 1506.7000	\$ 3013.4	\$ 6026.8

Administración de facturas

Para la administración de las facturas en el menú principal acceder a la opción facturas

Bienvenido a cofa

Bienvenido(a):
eduardo

Se desplegará la siguiente pantalla de administración de facturas.



Generar facturas

Para generar facturas acceder a la opción generar facturas.



Se desplegará la siguiente pantalla, que lista las cotizaciones en estado cerrado. Para generar facturas se deberá seleccionar las cotizaciones que se deseen facturar y poner la fecha limite de pago de la cotización.

Lista de cotizaciones cerradas

Id	Id Usuario	Fecha de cotización	Importe parcial	IVA	Importe total	Facturar
5	9	2011-04-06 15:15:03.0	\$ 53613.844	\$ 10212.16	\$ 63826.004	<input checked="" type="checkbox"/>
6	9	2011-04-06 15:21:26.0	\$ 9699.48	\$ 1847.52	\$ 11547.0	<input checked="" type="checkbox"/>

fecha limite

Al enviar los datos se desplegara la siguiente pantalla que muestra las facturas registradas.

Facturas vigentes

Folio	Estado	Id usuario	Fecha de expedición	Fecha limite de pago	Descargar pdf
12313 +/-	1	1	2011-03-24 12:36:28.0	2011-10-25	
12314 +/-	1	9	2011-04-05 11:44:55.0		
12315 +/-	1	9	2011-04-05 11:44:56.0		
12316 +/-	1	10	2011-04-05 19:30:25.0		
12317 +/-	1	9	2011-04-06 19:30:02.0		
12318 +/-	1	9	2011-04-06 19:30:02.0		

Ver detalle de factura

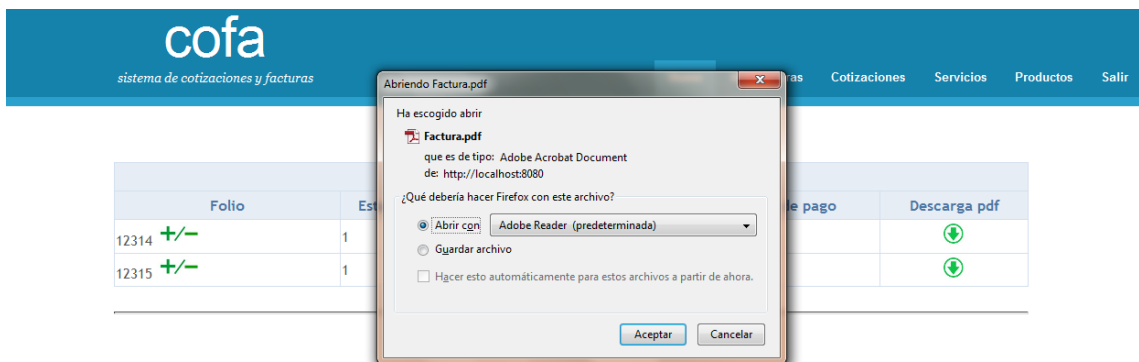
Para ver los detalles de la factura en la pantalla ver facturas dar click en el símbolo +/-, se desplegaran los detalles de la cotización, para ocultarlo basta con dar click de nuevo en el símbolo +/-.

Mis facturas					
Folio	Estado	Fecha de expedición	Fecha límite de pago	Descarga pdf	
12314 +/-	1	2011-04-05 11:44:55.0			
12315 +/-	1	2011-04-05 11:44:56.0			

RFC	GOHE851004KL6
Domicilio fiscal	Calle 1 No.2 Delegación Gustavo A. Madero
Teléfono	56243582
Datos de factura	
Folio	12315
Serie	122
Fecha de Expedición	2011-04-05 11:44:56.0

Descargar factura

Para descargar la factura en formato PDF basta con dar click en el icono descargar factura (flecha verde apuntando hacia abajo)



Administración de productos

Para la administración de productos deberá seleccionar en el menú principal la opción de productos.

Bienvenido a cofa

Bienvenido(a):
eduardo

Se mostrara la siguiente pantalla



Agregar producto

Para agregar un nuevo producto seleccionar la opción agregar



Se desplegara la siguiente pantalla donde se deberán ingresar los datos de un producto nuevo.

Nombre

Marca

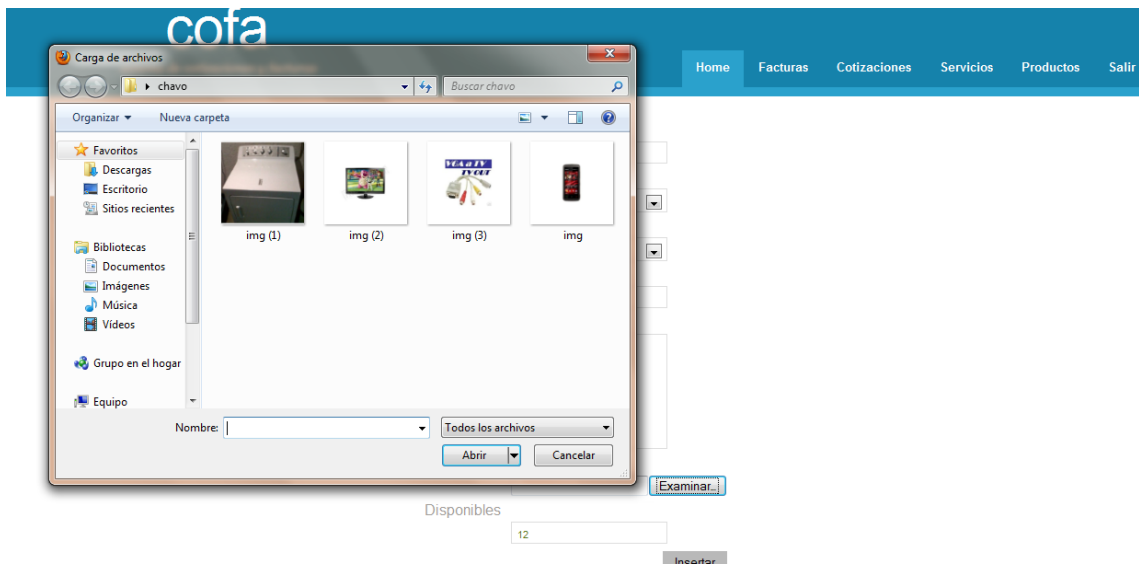
Clasificación

Precio

Descripción

Imagen

Disponibles



Modificar producto

Para modificar un producto seleccione la opción consultar

Acá se administraran los productos



Se mostrara la siguiente pantalla donde se podrá elegir el producto a modificar.

Lista productos						
Imagen	Nombre	Marca	Precio	Clasificación	Borrar	modificar
	memoria 8 GB	1	\$ 2220.0	4	✗	
	Nokia N900	8	\$ 5320.0	2	✗	
	Celular Nokia C3	8	\$ 1800.0	2	✗	
	Celular Sony Ericsson W705	9	\$ 2597.0	2	✗	
	Sony Ericsson U8	9	\$ 4199.0	2	✗	
	Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	✗	
	Lap Top Dell Latitude D620 Intel Core Duo Wifi	3	\$ 3000.0	3	✗	
	Marin Land 2 En Excelente Estado	4	\$ 1750.0	8	✗	

Al seleccionar un producto dando click en el icono de modificar se mostrara la siguiente pantalla con los datos del producto seleccionado.

Modificar producto

Nombre

Marca

Clasificación

Precio

Descripcion

Imagen

Borrar producto

Acá se administraran los productos

Celulares y accesorios

Computación
















Consolas y videojuegos

Electrónica

Audio

Video

Hogar

Lista productos						
Imagen	Nombre	Marca	Precio	Clasificación	Borrar	modificar
	memoria 8 GB	1	\$ 2220.0	4	X	
	Nokia N900	8	\$ 5320.0	2	X	
	Celular Nokia C3	8	\$ 1800.0	2	X	
	Celular Sony Ericsson W705	9	\$ 2597.0	2	X	
	Sony Ericsson U8	9	\$ 4199.0	2	X	
	Lap Top Toshiba 3 Gb Ram C645d	6	\$ 6898.0	3	X	
	Lap Top Dell Latitude D620 Intel Core Duo Wifi	3	\$ 3000.0	3	X	
	María Land 2 En Excelente Estado	4	\$ 1750.0	R	X	