

UNIVERSIDAD AUTÓNOMA METROPOLITANA UNIDAD AZCAPOTZALCO

DIVISIÓN DE CIENCIAS BÁSICAS E INGENIERÍA

LICENCIATURA EN INGENIERÍA EN COMPUTACIÓN

PROYECTO TERMINAL

“SISTEMA DE GESTIÓN PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS.”

Alumna: Stephanie Varela Peña

Matricula: 205307831

Trimestre 11-P

Asesorado por:

M. en C. Silva López Rafaela Blanca

17114

Jueves 8 de septiembre del 2011

OBJETIVO GENERAL	3
OBJETIVOS PARTICULARES.....	3
INTRODUCCIÓN	3
ANTECEDENTES.....	3
REFERENCIAS INTERNAS:.....	3
REFERENCIAS EXTERNAS:	3
JUSTIFICACIÓN.....	4
DESCRIPCIÓN TÉCNICA.....	5
ESPECIFICACIONES TÉCNICAS	7
ANÁLISIS Y DISEÑO DE LA BASE DE DATOS RELACIONAL	9
DICCIONARIO DE DATOS	9
DIAGRAMA ENTIDAD-RELACIÓN.....	18
ESQUEMA FÍSICO	19
DISEÑO E IMPLEMENTACIÓN DE SISTEMA DE GESTIÓN PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS. 29	
DIAGRAMAS UML	29
CASOS DE USO	32
DOCUMENTACIÓN DE ACTORES	34
MAPA DE NAVEGACIÓN.....	49
CÓDIGO FUENTE	50
• <i>Modulo Validar Usuario</i>	<i>50</i>
• <i>Modulo Realizar Solicitud y Autorización.....</i>	<i>58</i>
• <i>Modulo Realizar Cotización.....</i>	<i>80</i>
• <i>Modulo Administrador de Proveedores</i>	<i>101</i>
• <i>Archivos XML.....</i>	<i>125</i>
BIBLIOGRAFÍA	134

Objetivo General

- Diseñar e implementar un sistema de gestión para la adquisición de bienes y servicios, que permita a un departamento realizar el levantamiento de un pedido contra una lista de proveedores mejorando así la rapidez, facilidad de uso sin generar gasto alguno.

Objetivos Particulares

- Diseñar y construir el módulo de Solicitud y cotización de un bien o servicio.
- Diseñar y construir el módulo de autorización de adquisiciones y levantamiento de pedidos
- Diseñar y construir el módulo de Administración de Proveedores

Introducción

El Sistema de Gestión para la Adquisición de Bienes y Servicios es un sistema que se encarga de realizar un levantamiento de una orden de compra a proveedores, el empleado realiza una solicitud que es enviada a la administración está a su vez realiza la cotización correspondiente, el gerente da la autorización a la administración quien realiza el pedido.

Antecedentes

Referencias Internas:

Dentro de la UAM-A existen varios Proyectos Terminales los cuales son Sistemas de Gestión, pero ninguno tiene una similitud con este Sistema de Gestión para la adquisición de Bienes o Servicios.

Referencias Externas:

El Software ERP(Enterprise Resource Planning) [1] Es un sistema de gestión comercial con funciones extendidas, pensado para el planeamiento y el análisis para la toma de decisiones. A diferencia de la gestión tradicional, un ERP permite presupuestar, sacar estadísticas, generar proyecciones en base a históricos (por ejemplo, de compras y ventas, de stocks,

etc.), lo que permite optimizar la *performance* comercial. A su vez, es un sistema pensado para ser utilizado como única fuente de información para toda la empresa, por lo que supone abarca una mayor cantidad de funciones sin necesidad de utilizar sistemas y utilitarios complementarios para cubrir necesidades particulares.



[3] Isis Classic [2] es un sistema de gestión de compras, el cual es un software de gestión comercial diseñado para la administración eficiente de las pequeñas y medianas empresas.



[5] Husky Gestión Comercial [4] es un sistema informático totalmente integrado, que está dirigido a las PyMES como una simple y poderosa herramienta para informatizar la gestión administrativa de la empresa. Desarrollado bajo Windows, éste sistema administrativo apunta a agilizar las tareas rutinarias de su empresa brindándole las herramientas necesarias para el control de stocks, facturación, remitos, cuentas a cobrar y a pagar, comprobantes a vencer, caja y bancos, cheques, liquidar comisiones a vendedores, confeccionar presupuestos, emitir los libros de IVA, obtener estadísticas de ventas y compras, análisis de gastos, etc.

Como se puede ver un sistema ERP contiene muchos módulos, que en ocasiones no todas son necesarios para una PyME, muchas veces no resulta tan sencillo el uso de estos y además es un costo alto que en ocasiones es inalcanzable para una PyME.

A diferencia de los software mencionados esta propuesta de Sistema de Gestión para la Adquisición de Bienes y Servicios, pretende brindar un uso sencillo para los usuarios, sin costo y un sistema que sólo contiene las funciones necesarias.

Justificación

Actualmente surgen distintas problemáticas en el ámbito laboral, y muchas de ellas pueden ser resueltas con el manejo adecuado de la tecnología y nuevos sistemas, una de éstas problemáticas es el largo y tedioso proceso para la solicitud de material y servicio, ya que si no se cuenta con un software que simplifique lo anterior mencionado, entonces puede volverse un proceso complicado para el departamento, disminuyendo así la eficacia laboral y el logro de los diferentes objetivos establecidos.

Las soluciones ERP son muy grandes y costosas, no pueden estar al alcance de una PyME. El presente proyecto pretende ofrecer una solución que esté al alcance de cualquier tipo de empresa.

Lo que se quiere lograr con éste proyecto de Sistema de Gestión para la Adquisición de Bienes y Servicios es tener un proceso más sencillo, rápido y gratuito para el usuario, permitiendo así el buen desarrollo para distintos departamentos de la Universidad Autónoma Metropolitana (UAM) generando así una mayor eficacia y logro de objetivos establecidos.

En base a los conocimientos adquiridos y la conjunción de éstos a lo largo de la carrera de Ingeniería en Computación como Base de Datos, Ingeniería de Software y Metodologías de la Información, por mencionar algunos; es posible el desarrollo de un Sistema de Gestión que cuente con las características necesarias para el fácil, sencillo y buen uso de éste, pudiéndose adquirir sin generar ningún gasto para la UAM, como el software comercial lo hace.

El futuro del sistema propuesto, consistirá en aumentar su funcionalidad, incorporando más módulos como facturación, ventas, etc.

Descripción Técnica

El Sistema de Gestión para la Adquisición de Bienes y Servicios es un sistema que se encarga de realizar un levantamiento de una orden de compra contra una lista de proveedores almacenada en el sistema, el empleado realiza una solicitud que es enviada a la administración está a su vez realiza la cotización correspondiente, el gerente da la autorización a la administración quien realiza el pedido

El sistema contará con los siguientes módulos:

a) Módulo de Solicitud y cotización de un bien o servicio.

El módulo de Solicitud de un bien o servicio, permite al empleado realizar una petición para solicitar el bien o servicio (La entrada de este módulo son los datos donde el empleado realiza una solicitud de adquisición). El administrador entra al modulo de Solicitud y Cotizaciones de un Bien o Servicio verifica si hay peticiones, si las hay solicita al sistema que envíe una petición solicitando la cotización al proveedor correspondiente.

b) Módulo de autorización de adquisiciones y levantamiento de pedidos

El módulo de autorización de adquisiciones y levantamiento de pedidos recibe la cotización enviada por el proveedor la cual fue solicitada en el modulo anterior como ya

fue mencionado, la cotización se presentan a la gerencia para que sea autorizada la compra y se registre en el sistema. El sistema enviará el pedido al proveedor que se le solicito previamente la cotización.

c) Módulo de Administración de Proveedores

En el módulo de Administración de Proveedores, se da de alta proveedores, se realizan actualizaciones y se generan reportes de los mismos.

Una vez terminado de diseñar y construir los módulos antes mencionados, se desarrollara la Interfaz de Integración de los mismos.



Figura 1. Diagrama de Caso de Uso General

Especificaciones Técnicas

La plataforma de desarrollo para éste sistema será con un Sistema Operativo Windows Xp, lenguaje de programación Java, Entorno de desarrollo Eclipse¹, Sistema de Gestión de Base de Datos MySQL, Servidor de Aplicación Tomcat², Frameworks³ catalogados para Java los cuales son: Hibernate, Struts , Springs

La plataforma de Ejecución puede realizarse en un Sistema Operativo Windows/Linux.

Lo antes ya mencionado se muestra en el siguiente Diagrama de Bloques de Aplicación.

Nivel 6

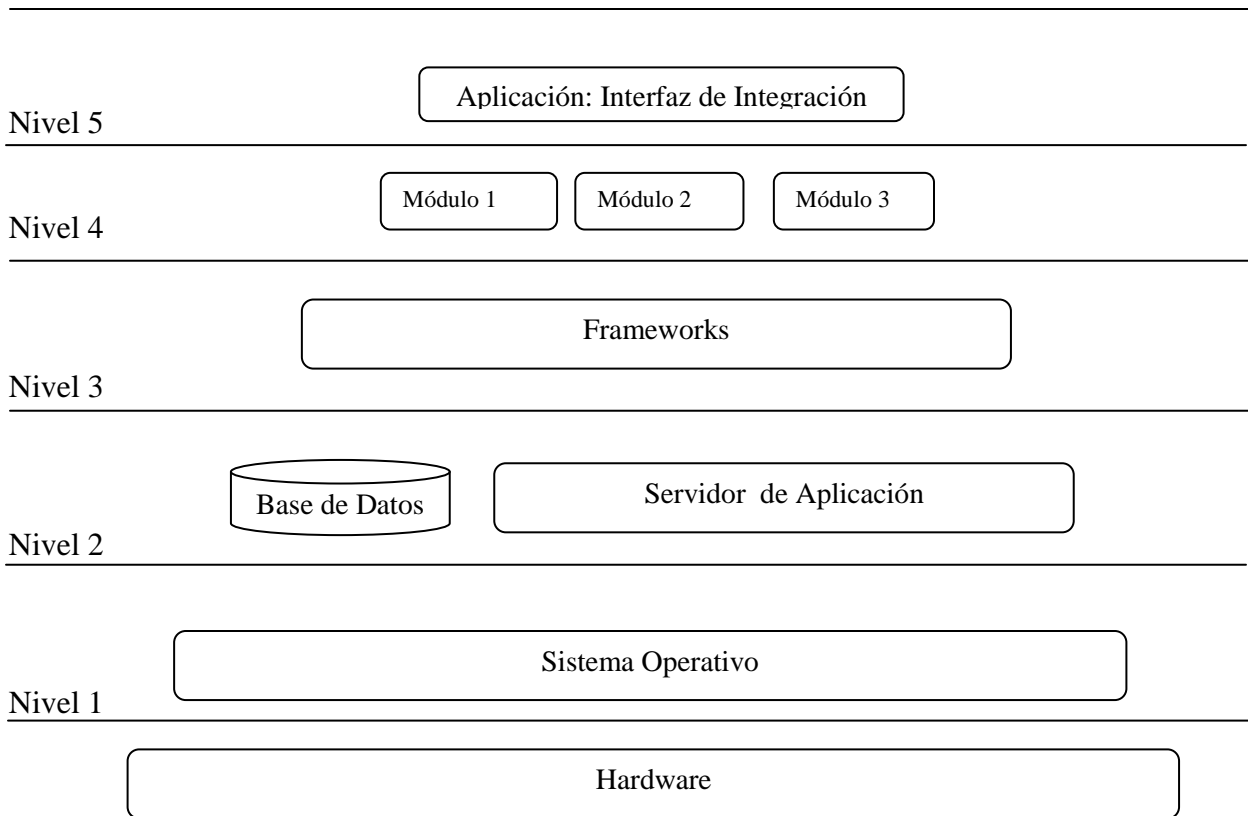


Figura 2. Diagrama de Bloques de la Aplicación [7], [8]

Nivel 1 : Hardware que se ocupara en el desarrollo del Sistema de Gestión anteriormente dicho.

¹ Eclipse[9].- Es un entorno de Desarrollo Integrado de código abierto.

² Servidor de Aplicación Tomcat [10].- Servidor Web con soporte de Servlets y JSPs

³ Framework[11].-Es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular, que sirve como referencia para enfrentar y resolver nuevos problemas de índole similar.

Nivel 2: El Sistema Operativo (Windows) que se ocupara en el desarrollo del Sistema de Gestión anteriormente dicho.

Nivel 3: Base de Datos y Servidor de Aplicación que ocupara el Sistema

Nivel 4: Frameworks que ocupara el Sistema.

Nivel 5: Módulos que serán construidos en el Sistema:

Módulo 1: Módulo de Solicitud y cotización de un bien o servicio.

Módulo 2: Módulo de autorización de adquisiciones y levantamiento de pedidos.

Módulo 3: Módulo de Administración de Proveedores

Nivel 6: Interfaz de Integración del Sistema, la cual se refiere a la agrupación de los distintos módulos.

Licencia de Software Libre – Creative Commons.

Reconocimiento (Attribution): En cualquier explotación de la obra autorizada por la licencia hará falta reconocer la autoría.

Compartir Igual (Share alike): La explotación autorizada incluye la creación de obras derivadas siempre que mantengan la misma licencia al ser divulgadas.



Figura 3. Logo de la licencia *Creative Commons*

Reconocimiento – Compartir Igual (by-sa): Se permite la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Análisis y Diseño de la Base de Datos Relacional

Diccionario de Datos

Proveedores

Nombre del Atributo	Descripción	Llave Primaria	Llave Secundaria	Valor Único	Tipo de Datos	Formato
Número_proveedor	Identifica al proveedor mediante un numero consecutivo	X		X	Numero(s) entero corto positivo	Integer
Razón_social	Nombre y firma de la Empresa				Caracteres alfanuméricos	Longtext
RFC	Registro Federal De Contribuyentes del Proveedores				Caracteres alfanuméricos	Longtext
Dirección	Indica dirección física del proveedor				Caracteres alfanuméricos	Longtext
Teléfono	Indica teléfono del proveedor				Numeros enteros positivos	Longtext
Email	Indica email del proveedor				Caracteres alfanuméricos	Longtext
Comentarios	Indica comentarios relevantes acerca del proveedor				Caracteres alfanuméricos	Longtext
Tiempo_entrega	Indica la fecha de				Fecha y hora	Datetime

	entrega del proveedor				dd-mm-aaaa,hh:mm:ss	
Estado_proveedor	Indica el estado del Proveedor				Caracteres alfabéticos	char

Adquisiciones

Nombre del Atributo	Descripción	Llave Primaria	Llave Secundaria	Valor Único	Tipo de Datos	Formato
Número_adquisición	Identifica la adquisición mediante un numero consecutivo				Numero(s) entero corto positivo	Integer
Número_proveedor	Identifica al proveedor mediante un numero consecutivo				Numero(s) entero corto positivo	Integer
Monto	Proporciona el monto total de la adquisición				Números enteros positivos	Float
Fecha solicitud	Proporciona la fecha en la que se realiza la solicitud				Fecha y hora dd-mm-aaaa,hh:mm:ss	Datetime
Autorización	Indica si se				Caracteres	char

	autoriza o no la adquisición				alfabéticos	
Observaciones	Proporciona observaciones relevantes acerca de la adquisición				Caracteres alfanuméricos	Longtext
Estado_adquisicion	Indica el estado de la adquisición				Caracteres alfabéticos	char

Pedidos

Nombre del Atributo	Descripción	Llave Primaria	Llave Secundaria	Valor Único	Tipo de Datos	Formato
Número_pedido	Identifica al pedido mediante un numero consecutivo	X		X	Numero(s) entero corto positivo	Integer
Número_adquisición	Identifica la adquisición mediante un numero consecutivo				Numero(s) entero corto positivo	Integer
Número_cliente	Identifica al cliente mediante un numero consecutivo				Numero(s) entero corto positivo	Integer

Fecha_solicitud	Proporciona la fecha en la que se realiza la solicitud				Fecha y hora dd-mm- aaaa,hh:mm:ss	Datetime
Fecha_registro_pedido	Proporciona la fecha en la que se registro el pedido				Fecha y hora dd-mm- aaaa,hh:mm:ss	Datetime
Fecha_pago	Proporciona la fecha de pago en la que se realizo pedido				Fecha y hora dd-mm- aaaa,hh:mm:ss	Datetime
Monto	Proporciona el Monto total del pedido				Números enteros positivos	Float
Tipo_pago	Indica la forma en la que se efectuó el pago				Caracteres alfabéticos	Char
Observaciones	Proporciona observaciones relevantes acerca de la adquisición				Caracteres alfanuméricos	Longtext
Estado_pedido	Indica el estado del pedido				Caracteres alfabéticos	Char

Detalle de Pedido

Nombre del Atributo	Descripción	Llave Primaria	Llave Secundaria	Valor Único	Tipo de Datos	Formato
Número de pedido	Identifica al pedido mediante un numero consecutivo	X		X	Numero(s) entero corto positivo	Integer
Número_item	Proporciona un numero consecutivo que ordena al producto dentro del pedido				Numero(s) entero corto positivo	Integer
número_producto	Identifica al producto mediante un numero consecutivo				Numero(s) entero corto positivo	Integer
cantidad	Proporciona la cantidad que se pidió de cierto producto				Numero(s) entero corto positivo	Integer
precio_unitario	Proporciona el precio de una unidad.				Numero(s) entero corto positivo	Float
especificaciones	Características del pedido				Caracteres alfanuméricos	Longtext

Usuarios

Nombre del Atributo	Descripción	Llave Primaria	Llave Secundaria	Valor Único	Tipo de Datos	Formato
Id_usuario	Identifica al usuario mediante un numero consecutivo	X			Numero(s) entero corto positivo	Integer
nombre	Nombre del usuario				Caracteres	Longtext
Login	Proporciona una cadena de caracteres				Caracteres	Longtext
Password	Proporciona una cadena de caracteres				Caracteres	Longtext
Perfil	Clasifica al usuario				Numero(s) entero corto positivo	Integer

Solicitudes

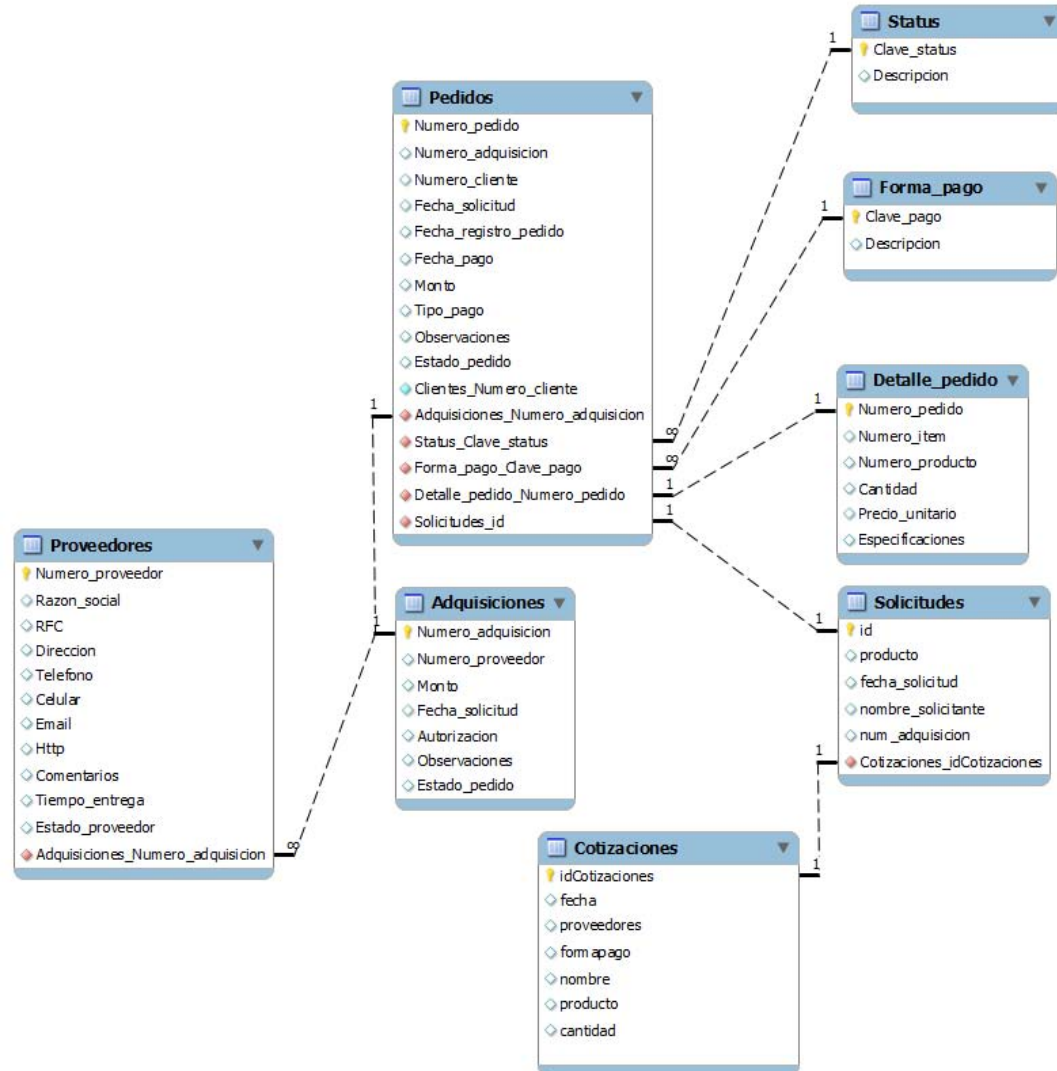
Nombre del Atributo	Descripción	Llave Primaria	Llave Secundaria	Valor Único	Tipo de Datos	Formato
Id	Identifica la solicitud mediante un numero consecutivo				Numero(s) entero corto positivo	Integer
Producto	Nombre del producto				Caracteres	Longtext
Cantidad	Proporciona un numero entero				Numero entero positivo	Int
fecha_solicitud	Proporciona la fecha de solicitud				Fecha y hora dd-mm-aaaa,hh:mm:ss	Datetime
nombre_solicitud	Proporciona el nombre de quien realiza la solicitud				Caracteres	Longtext
num_adquisicion	Proporciona el numero de solicitud que ha realizado el usuario				Numero(s) entero corto positivo	int

Cotizaciones

Nombre del Atributo	Descripción	Llave Primaria	Llave Secundaria	Valor Único	Tipo de Datos	Formato
Id	Identifica la solicitud mediante un numero consecutivo				Numero(s) entero corto positivo	Integer
Fecha	Proporciona la fecha de cotizacion				Fecha y hora dd-mm-aaaa,hh:mm:ss	Datetime
Proveedores	Proporciona el nombre del Proveedor				Caracteres	Longtext
formapago	Proporciona una descripción de pago(efectivo o tarjeta)				Caracteres	Longtext
nombre	Proporciona el nombre de quien realiza la cotización				Caracteres	Longtext

Cantidad	Proporciona un numero entero				Numero entero positivo	int
Producto	Nombre del producto				Caracteres	Longtext

Diagrama Entidad-Relación



Esquema Físico

```
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` DEFAULT CHARACTER  
SET latin1 COLLATE latin1_swedish_ci ;
```

```
CREATE SCHEMA IF NOT EXISTS `Proyecto Terminal` ;
```

```
-----
```

```
-- Table `mydb`.`Status`
```

```
-----
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Status` (  
  
  `Clave_status` CHAR NOT NULL ,  
  
  `Descripcion` LONGTEXT NULL DEFAULT NULL ,  
  
  PRIMARY KEY (`Clave_status`))  
  
ENGINE = InnoDB;
```

```
-----  
-- Table `mydb`.`Forma_pago`
```

```
-----  
CREATE TABLE IF NOT EXISTS `mydb`.`Forma_pago` (  
  `Clave_pago` CHAR NOT NULL ,  
  `Descripcion` LONGTEXT NULL DEFAULT NULL ,  
  PRIMARY KEY (`Clave_pago`))  
ENGINE = InnoDB;
```

```
-----  
-- Placeholder table for view `mydb`.`view1`
```

```
-----  
CREATE TABLE IF NOT EXISTS `mydb`.`view1` (`id` INT);
```

```
-----  
-- View `mydb`.`view1`
```

```
-----  
DROP TABLE IF EXISTS `mydb`.`view1`;
```

```
-----  
-- Table `Proyecto Terminal`.`Adquisiciones`
```

```

-----
CREATE TABLE IF NOT EXISTS `Proyecto Terminal`.`Adquisiciones` (
  `Numero_adquisicion` INT(11) NOT NULL DEFAULT NULL ,
  `Numero_proveedor` INT(11) NULL DEFAULT NULL ,
  `Monto` FLOAT NULL DEFAULT NULL ,
  `Fecha_solicitud` DATETIME NULL DEFAULT NULL ,
  `Autorizacion` CHAR NULL DEFAULT NULL ,
  `Observaciones` LONGTEXT NULL DEFAULT '\0' ,
  `Estado_pedido` CHAR NULL ,
  PRIMARY KEY (`Numero_adquisicion`)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
-----

```

```

-- Table `Proyecto Terminal`.`Detalle_pedido`
-----

```

```

CREATE TABLE IF NOT EXISTS `Proyecto Terminal`.`Detalle_pedido` (
  `Numero_pedido` INT(11) NOT NULL DEFAULT NULL ,
  `Numero_item` INT(11) NULL DEFAULT NULL ,
  `Numero_producto` INT(11) NULL DEFAULT NULL ,
  `Cantidad` INT(11) NULL DEFAULT NULL ,

```

```

`Precio_unitario` FLOAT NULL DEFAULT NULL ,
`Especificaciones` LONGTEXT NULL DEFAULT '\0' ,
PRIMARY KEY (`Numero_pedido`))
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
-----
-- Table `Proyecto Terminal`.`Cotizaciones`
-----

CREATE TABLE IF NOT EXISTS `Proyecto Terminal`.`Cotizaciones` (
  `idCotizaciones` INT NOT NULL ,
  `fecha` DATETIME NULL ,
  `proveedores` VARCHAR(45) NULL ,
  `formapago` VARCHAR(45) NULL ,
  `nombre` VARCHAR(45) NULL ,
  `producto` VARCHAR(45) NULL ,
  `cantidad` INT NULL ,
  PRIMARY KEY (`idCotizaciones`))
ENGINE = InnoDB;

```

-- Table `Proyecto Terminal`.`Solicitudes`

```
CREATE TABLE IF NOT EXISTS `Proyecto Terminal`.`Solicitudes` (  
  `id` INT(11) NOT NULL AUTO_INCREMENT ,  
  `producto` VARCHAR(45) NULL DEFAULT NULL ,  
  `fecha_solicitud` DATETIME NULL DEFAULT NULL ,  
  `nombre_solicitante` VARCHAR(45) NULL DEFAULT NULL ,  
  `num_adquisicion` INT NULL DEFAULT '0' ,  
  `Cotizaciones_idCotizaciones` INT NOT NULL ,  
  PRIMARY KEY (`id`) ,  
  INDEX `fk_Solicitudes_Cotizaciones1` (`Cotizaciones_idCotizaciones`  
  ASC) ,  
  CONSTRAINT `fk_Solicitudes_Cotizaciones1`  
    FOREIGN KEY (`Cotizaciones_idCotizaciones` )  
    REFERENCES `Proyecto Terminal`.`Cotizaciones` (`idCotizaciones` )  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = latin1;
```

-- Table `Proyecto Terminal`.`Pedidos`

```
CREATE TABLE IF NOT EXISTS `Proyecto Terminal`.`Pedidos` (  
  `Numero_pedido` INT(11) NOT NULL AUTO_INCREMENT ,  
  `Numero_adquisicion` INT(11) NULL DEFAULT NULL ,  
  `Numero_cliente` INT(11) NULL DEFAULT NULL ,  
  `Fecha_solicitud` DATETIME NULL DEFAULT NULL ,  
  `Fecha_registro_pedido` DATETIME NULL DEFAULT NULL ,  
  `Fecha_pago` DATETIME NULL DEFAULT NULL ,  
  `Monto` FLOAT NULL DEFAULT NULL ,  
  `Tipo_pago` CHAR NULL DEFAULT NULL ,  
  `Observaciones` LONGTEXT NULL DEFAULT '\0' ,  
  `Estado_pedido` CHAR NULL ,  
  `Clientes_Numero_cliente` INT(11) NOT NULL ,  
  `Adquisiciones_Numero_adquisicion` INT(11) NOT NULL ,  
  `Status_Clave_status` CHAR NOT NULL ,  
  `Forma_pago_Clave_pago` CHAR NOT NULL ,  
  `Detalle_pedido_Numero_pedido` INT(11) NOT NULL ,  
  `Solicitudes_id` INT(11) NOT NULL ,
```



```

PRIMARY KEY (`Numero_pedido`),

INDEX `fk_Pedidos_Adquisiciones1`
(`Adquisiciones_Numero_adquisicion` ASC),

INDEX `fk_Pedidos_Status1` (`Status_Clave_status` ASC),

INDEX `fk_Pedidos_Forma_pago1` (`Forma_pago_Clave_pago` ASC),

INDEX `fk_Pedidos_Detalle_pedido1` (`Detalle_pedido_Numero_pedido`
ASC),

INDEX `fk_Pedidos_Solicitudes1` (`Solicitudes_id` ASC),

CONSTRAINT `fk_Pedidos_Adquisiciones1`

FOREIGN KEY (`Adquisiciones_Numero_adquisicion`)

REFERENCES `Proyecto Terminal`.`Adquisiciones`
(`Numero_adquisicion`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk_Pedidos_Status1`

FOREIGN KEY (`Status_Clave_status`)

REFERENCES `mydb`.`Status` (`Clave_status`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk_Pedidos_Forma_pago1`

FOREIGN KEY (`Forma_pago_Clave_pago`)

```

```
REFERENCES `mydb`.`Forma_pago` (`Clave_pago` )
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `fk_Pedidos_Detalle_pedido1`
FOREIGN KEY (`Detalle_pedido_Numero_pedido` )
REFERENCES `Proyecto Terminal`.`Detalle_pedido` (`Numero_pedido` )
ON DELETE NO ACTION
ON UPDATE NO ACTION,
CONSTRAINT `fk_Pedidos_Solicitudes1`
FOREIGN KEY (`Solicitudes_id` )
REFERENCES `Proyecto Terminal`.`Solicitudes` (`id` )
ON DELETE NO ACTION
ON UPDATE NO ACTION)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

-- Table `Proyecto Terminal`.`Proveedores`

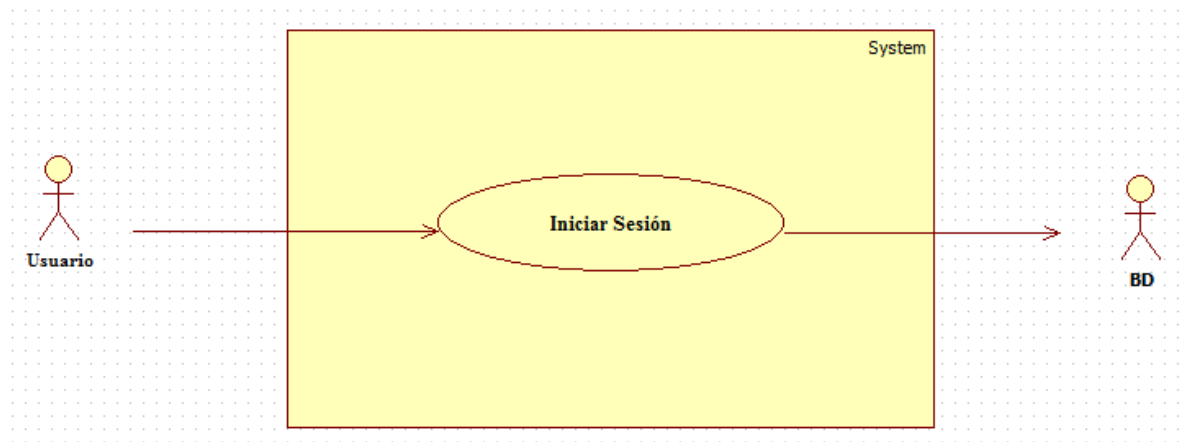
```
CREATE TABLE IF NOT EXISTS `Proyecto Terminal`.`Proveedores` (  
  `Numero_proveedor` INT(11) NOT NULL AUTO_INCREMENT ,  
  `Razon_social` LONGTEXT NULL DEFAULT NULL ,  
  `RFC` LONGTEXT NULL DEFAULT NULL ,  
  `Direccion` LONGTEXT NULL DEFAULT NULL ,  
  `Telefono` LONGTEXT NULL DEFAULT NULL ,  
  `Celular` LONGTEXT NULL DEFAULT NULL ,  
  `Email` LONGTEXT NULL DEFAULT NULL ,  
  `Http` LONGTEXT NULL ,  
  `Comentarios` LONGTEXT NULL DEFAULT '\0' ,  
  `Tiempo_entrega` DATETIME NULL ,  
  `Estado_proveedor` CHAR NULL ,  
  `Adquisiciones_Numero_adquisicion` INT(11) NOT NULL ,  
  PRIMARY KEY (`Numero_proveedor`),  
  INDEX `fk_Proveedores_Adquisiciones1`  
  (`Adquisiciones_Numero_adquisicion` ASC) ,  
  CONSTRAINT `fk_Proveedores_Adquisiciones1`
```

```
FOREIGN KEY (`Adquisiciones_Numero_adquisicion` )  
  
REFERENCES `Proyecto Terminal`.`Adquisiciones`  
(`Numero_adquisicion` )  
  
ON DELETE NO ACTION  
  
ON UPDATE NO ACTION)  
  
ENGINE = InnoDB  
  
DEFAULT CHARACTER SET = latin1;  
  
SET SQL_MODE=@OLD_SQL_MODE;  
  
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;  
  
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

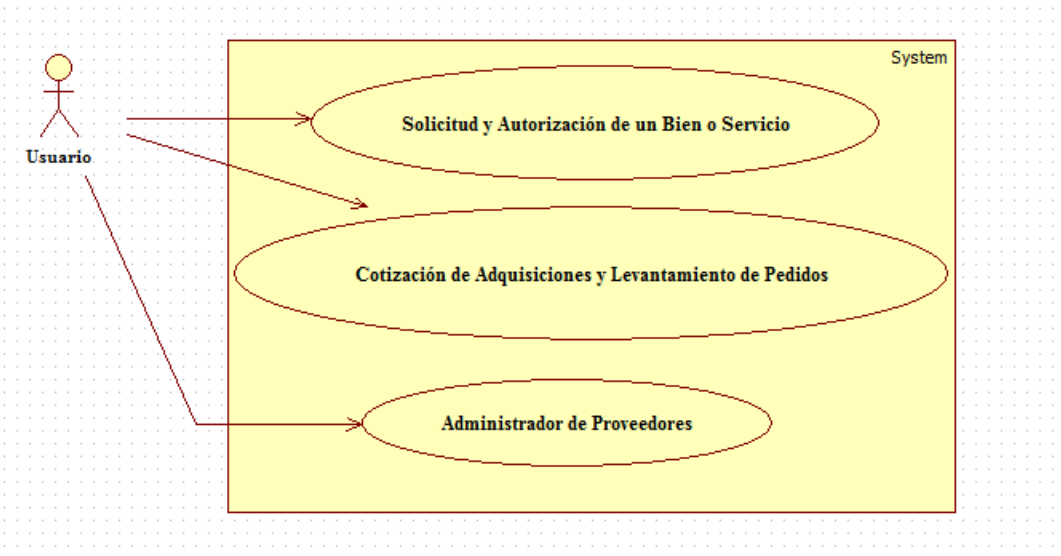
Diseño e Implementación de Sistema de Gestión para la Adquisición de Bienes y Servicios

Diagramas UML

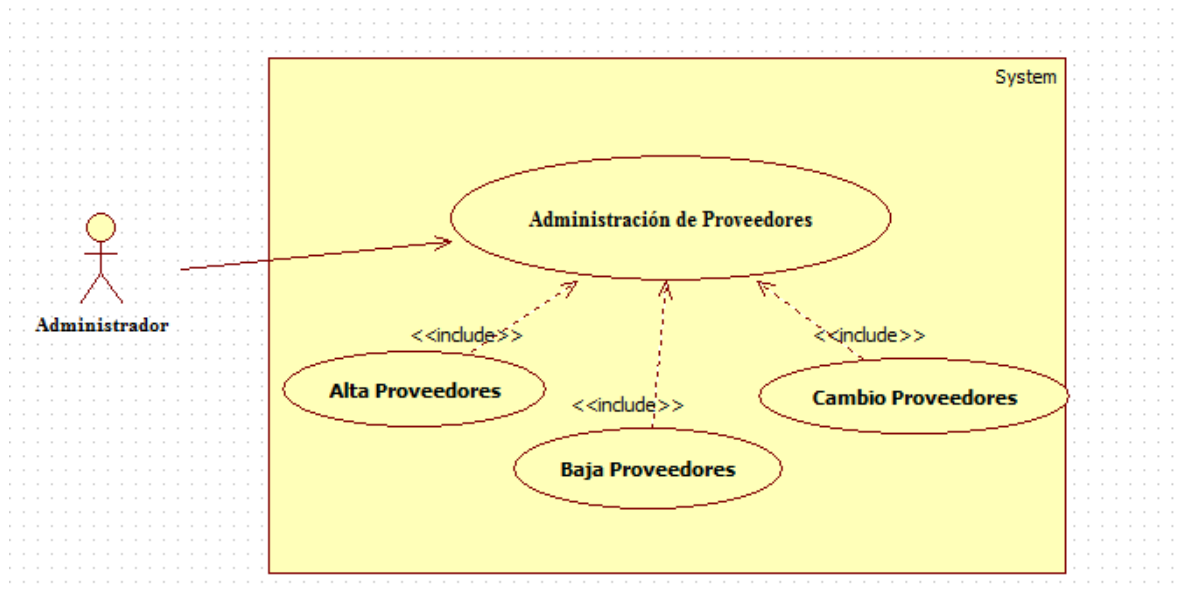
Iniciar Sesión



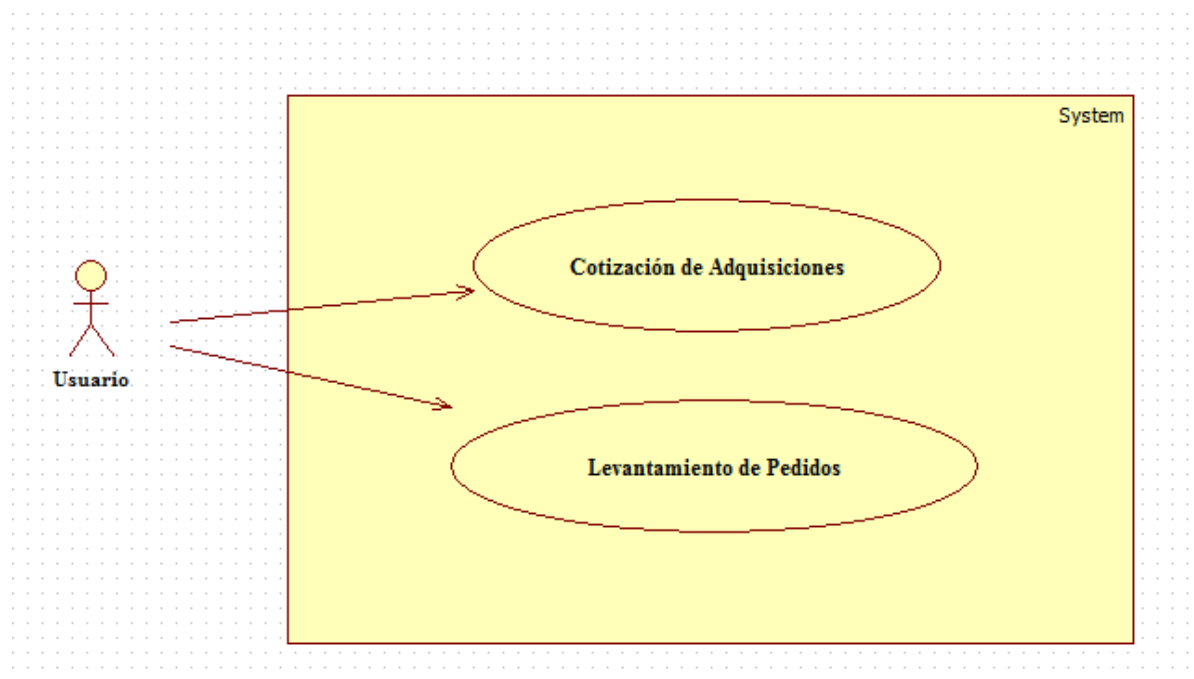
General



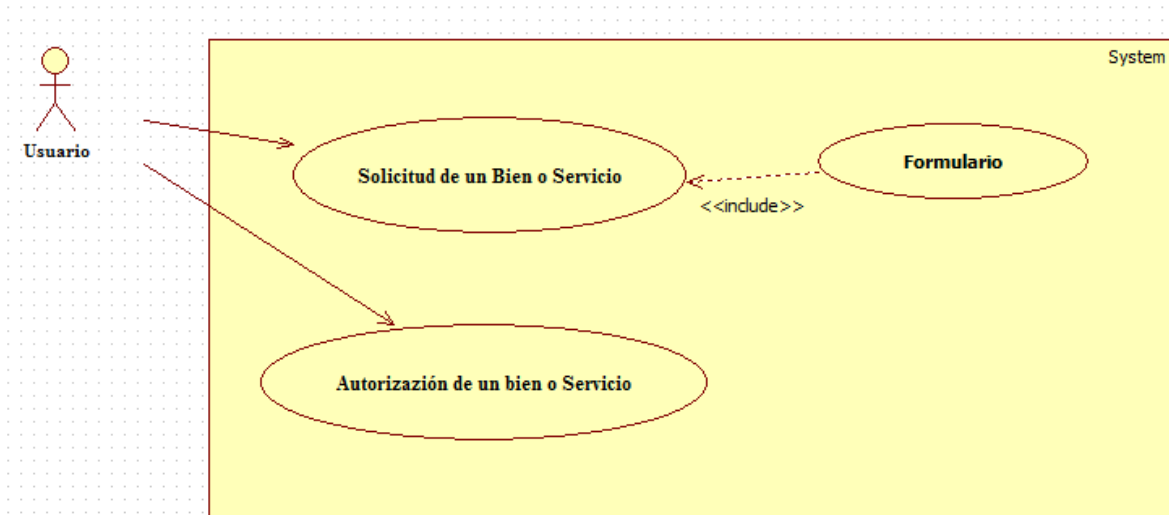
Administrador de Proveedores



Cotización de Adquisiciones y Levantamiento de Pedidos



Solicitud y Autorización de un Bien o Servicio



Casos de Uso

SISTEMA DE GESTION PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS

PROYECTO TERMINAL 2

Bienvenido al Sistema para la Adquisición de Bienes y Servicios

Proporciona tu cuenta de usuario y contraseña

Cuenta de Usuario:

Contraseña:

Alumna: Stephanie Varela Peña

Iniciar Sesión Pantalla Principal (P-1)

Caso de Uso	Iniciar Sesión
Actores	Usuario, BD
Tipo	Inclusión
Propósito	Verificar que el usuario este dado de alto en el sistema
Resumen	El usuario en la P-1 introducirá sus datos (login y password) para que el sistema los valide.
Pre -Condiciones	El usuario debe esta registrado en la Base de Datos
Post Condiciones	El usuario habrá ingresado al sistema y se le presentara la pantalla para que elija otras posibles opciones
Flujo Principal	Se presenta la pantalla principal (P-1) donde el usuario puede elegir "Accesar" (S-1) ó introducir sus datos (E-1)
Sub Flujo	(S-1) Se despliega la pantalla (P-2), ya que son validados los datos ingresados por el usuario (E-2).
Excepciones	(S-1) En caso de que los datos no sean válidos, se mostrará la misma pantalla (P-1). (E-2) En caso de que uno de los datos no coincida con la información en la BD, se mostrará la misma pantalla (P-1).



Menú
Interfaz de Integración (P-2)

Caso de Uso	Menú
Actores	Usuario, BD
Tipo	Básico
Propósito	Que el usuario elija la opción que desea realizar
Resumen	El usuario puede elegir alguna de las posibles acciones que desea realizar en el sistema.
Pre -Condiciones	El usuario debe estar registrado en el Sistema
Post Condiciones	Se cambiara a una ventana diferente dependiendo la acción elegida
Flujo Principal	Se ejecutara el caso de uso iniciar sesión, posteriormente se mostrara la pantalla (P-2) para el elegir el modulo a ingresar.
Sub Flujo	(S-1) Realizar Solicitud. Ingresar al Modulo de Realizar Solicitud (S-2) Realizar Cotización. Ingresar al Modulo de Realizar Cotización (S-3) Visualizar Autorizaciones. Ingresar al Modulo de Visualizar Autorizaciones (S-4) Proveedores. Ingresar al Modulo de Proveedores
Excepciones	Ninguna

SISTEMA DE GESTION PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS
PROYECTO TERMINAL 1

Administrador Proveedores

Proveedores										
id proveedor	razon social	nombre	rfc	direccion	Telefono	email	comentarios	tiempo entrega	estado proveedor	
<u>1</u>	Femsa coca cola	COCA-COLA FEMSA	CO:Ca	Av. sta fe 160	55678900	coca@kof.com.mx	ninguno	0	Disponible	<input type="button" value="Borrar Proveedor"/>
<u>2</u>	MAX	office MAX	OFFMAX	naucalpan #72	67789345	office@kof.com.mx	ningunpo	5	Disponible	<input type="button" value="Borrar Proveedor"/>
<u>3</u>	informatic	papeleria informatic	PAINFO	tlanepantla #35	223335566	ninguno	solo surte los dias lunes	1	No Disponible	<input type="button" value="Borrar Proveedor"/>

[salir](#)

Alumna:Stephanie Varela Peña

Pantalla Administrador Proveedores (P-3)

SISTEMA DE GESTION PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS
PROYECTO TERMINAL 1

Dar de Alta un Proveedor

Datos

Nombre	<input type="text"/>
Razón Social	<input type="text"/>
RFC	<input type="text"/>
Dirección	<input type="text"/>
Teléfono	<input type="text"/>
E-mail	<input type="text"/>
Comentarios	<input style="height: 40px;" type="text"/>
Tiempo Entrega(en días)	<input type="text"/>
Estado Proveedor (Disponible o No Disponible)	<input type="text"/>

Alumna:Stephanie Varela Peña

Pantalla Alta Proveedor (P-4)

Caso de Uso	Alta Proveedor
Actores	Usuario, BD
Tipo	Básico
Propósito	Que el usuario ingrese los datos de un nuevo proveedor
Resumen	Registrar un proveedor.
Pre -Condiciones	No debe de existir el proveedor en BD
Post Condiciones	Registrar los datos del proveedor en la BD
Flujo Principal	Se eligió la opción Insertar Proveedor en la pantalla (P-3), se llenan los campos correspondientes en la (P-4) y al elegir Insertar Proveedor se registra en la base de datos.
Sub Flujo	(S-1) Cancelar. Se regresa a la pantalla (P-3)
Excepciones	(E-1) En caso de no llenar un campo solicitado, se deberá mencionar al usuario que lo ingrese.

Ver Proveedor

Datos	
id	1
razon social	Femsa coca cola
nombre	COCA-COLA FEMSA
rfc	CO:Ca
direccion	Av. sta fe 160
Telefono	55678900
email	coca@kof.com.mx
tiempo entrega	0
estado Proveedor	Disponible

Actualizar Proveedor

Cancelar

Alumna:Stephanie Varela Peña

Ver Proveedor

Datos	
id	1
razon social	Femsa coca cola
nombre	COCA-COLA FEMSA
rfc	CO:Ca
direccion	Av. sta fe 160
Telefono	55678900
email	coca@kof.com.mx
tiempo entrega	0
estado Proveedor	Disponible

Actualizar Proveedor

Cancelar

Alumna:Stephanie Varela Peña

Pantalla Ver Proveedor (P-5)

SISTEMA DE GESTIÓN PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS
PROYECTO TERCERA E

Actualizar Proveedor

Datos	
Nombre	COCA-COLA PEMSA
Razón Social	Pemsa coca cola
RFC	CO-CA
Dirección	Av. 2da fe 100
Teléfono	55510000
E-mail	coca@cof.com.mx
Comentarios	ninguno
Tiempo Entrega(en días)	0
Estado Proveedor (Disponible o No Disponible)	Disponible

Alumna:Stephanie Varela Peña

Pantalla Actualizar Proveedor (P-6)

Caso de Uso	Actualizar Proveedor
Actores	Usuario, BD
Tipo	Básico
Propósito	Modificar los datos de un proveedor registrado
Resumen	Modificar los datos de un proveedor
Pre -Condiciones	Se almacenaran las modificaciones realizadas en la BD
Post Condiciones	Registrar los datos del proveedor en la BD
Flujo Principal	Se eligió la opción Actualizar Proveedor (P-5), se modifican los datos que se requieran (P-6) y se da nuevamente clic en el botón Actualizar Proveedor
Sub Flujo	(S-1) Cancelar. Se regresa a la pantalla (P-5)
Excepciones	(E-1) En caso de no llenar un campo solicitado, se deberá mencionar al usuario que lo ingrese.

SISTEMA DE GESTION PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS
PROTECTOR TERRIBIL 2

Administrador Proveedores

Proveedores										
id proveedor	razon social	nombre	rfc	direccion	Telefono	email	comentarios	tiempo entrega	estado proveedor	
1	Femsa coca cola	COCA-COLA FEMSA	CO:Ca	Av. sta fe 160	55678900	coca@kof.com.mx	ninguno	0	Disponible	<input type="button" value="Borrar Proveedor"/>
2	MAX	office MAX	OFFMAX	naucalpan #72	67789345	office@kof.com.mx	ningunpo	5	Disponible	<input type="button" value="Borrar Proveedor"/>
3	informatic	papeleria informatic	PAINFO	tlanepantla #35	223335566	ninguno	solo surte los dias lunes	1	No Disponible	<input type="button" value="Borrar Proveedor"/>

[salir](#)

Alumna:Stephanie Varela Peña

Pantalla Borrar Proveedor (P-3)

Caso de Uso	Borrar Proveedor
Actores	Usuario, BD
Tipo	Básico
Propósito	Borrar los datos de un proveedor registrado
Resumen	Borra los datos de un proveedor que se encuentra en la BD
Pre -Condiciones	El proveedor debe de estar registrado en la BD
Post Condiciones	El proveedor será eliminado de la BD y ya no se visualizara en la Pantalla (P-3)
Flujo Principal	Se eligió la opción Borrar Proveedor en la Pantalla (P-3), se da un clic en el botón para que el Proveedor sea eliminado
Sub Flujo	(S-1) Salir se va la Pantalla (P-1) (S-2)Insertar Proveedor , se va a la Pantalla (P-4)
Excepciones	Ninguna

SISTEMA DE GESTION PARA LA ADOUSIACIÓN DE BIENES Y SERVICIOS

PROYECTO TERMINAL 2

MENU
[Solicitudes](#)
[Autorizaciones](#)
[Cotización](#)
[Proveedores](#)

Solicitudes

Solicitudes						
id Solicitud	producto	cantidad	fecha	nombre solicitante	adquisicion	
15	hojas	60	08/09/2011	stephanie	1	<input type="button" value="Borrar Solicitud"/>

[salir](#)

Alumna:Stephanie Varela Peña

Pantalla Solicitudes (P-7)

SISTEMA DE GESTION PARA LA ADOUSIACIÓN DE BIENES Y SERVICIOS

PROYECTO TERMINAL 2

MENU
[Solicitudes](#)
[Autorizaciones](#)
[Cotización](#)
[Proveedores](#)

Realizar una Solicitud

Datos	
Producto	<input type="text"/>
Cantidad	<input type="text"/>
fecha de Solicitud	<input type="text"/> <input type="button" value="Calendar"/>
Nombre Solicitante	<input type="text"/>
Numero de Adquisición	<input type="text"/>

Pantalla Realizar Solicitud (P-8)

Caso de Uso	Alta Solicitud
Actores	Usuario, BD
Tipo	Básico
Propósito	Que el usuario ingrese los datos para realizar una solicitud
Resumen	Registrar una nueva solicitud
Pre -Condiciones	No debe de existir la solicitud en BD
Post Condiciones	Registrar los datos necesarios para realizar una solicitud
Flujo Principal	Se eligió la opción Realizar Solicitud en la pantalla (P-7), se llenan los campos correspondientes en (P-8) y al elegir Insertar Solicitud se registra en la base de datos.
Sub Flujo	(S-1) Cancelar. Se regresa a la pantalla (P-7)
Excepciones	(E-1) En caso de no llenar un campo solicitado, se deberá mencionar al usuario que lo ingrese.

SISTEMA DE GESTION PARA LA ADOUSIACIÓN DE BIENES Y SERVICIOS
PROYECTO TERRIBI 2

Ver Solicitud

Datos	
Numero de Pedido	5
Producto	1
Cantidad	20
Fecha de Solicitud (La fecha no se puede modificar)	17/08/2011
Nombre solicitante	1
Numero de Adquisicion	3

Actualizar Solicitud Salir

Alumna:Stephanie Varela Peña

Pantalla Ver Solicitud (P-9)



Pantalla Actualizar Solicitud (P-10)

Caso de Uso	Actualizar Solicitud
Actores	Usuario, BD
Tipo	Básico
Propósito	Modificar los datos de una solicitud registrada
Resumen	Modificar los datos de una solicitud
Pre -Condiciones	Se almacenaran las modificaciones realizadas en la BD
Post Condiciones	Registrar los datos de la solicitud en la BD
Flujo Principal	Se eligió la opción Actualizar Proveedor (P-9), se modifican los datos que se requieran (P-10) y se da nuevamente clic en el botón Actualizar Solicitud
Sub Flujo	(S-1) Cancelar. Se regresa a la pantalla (P-9)
Excepciones	(E-1) En caso de no llenar un campo solicitado, se deberá mencionar al usuario que lo ingrese.

SISTEMA DE GESTION PARA LA ADOUSIACIÓN DE BIENES Y SERVICIOS

PROYECTO TERMINAL 2

MENU

- [Solicitudes](#)
- [Autorizaciones](#)
- [Cotización](#)
- [Proveedores](#)

Solicitudes

Solicitudes						
id Solicitud	producto	cantidad	fecha	nombre solicitante	adquisicion	
15	hojas	60	08/09/2011	stephanie	1	<input type="button" value="Borrar Solicitud"/>

[salir](#)

Alumna:Stephanie Varela Peña

Pantalla Borrar Solicitud (P-11)

Caso de Uso	Borrar Solicitud
Actores	Usuario, BD
Tipo	Básico
Propósito	Borrar los datos de una solicitud registrada
Resumen	Borra los datos de una solicitud que se encuentra en la BD
Pre -Condiciones	La solicitud debe de estar registrada en la BD
Post Condiciones	La solicitud será eliminado de la BD y ya no se visualizara en la Pantalla (P-11)
Flujo Principal	Se eligió la opción Borrar Solicitud en la Pantalla (P-11), se da un clic en el botón para que la solicitud sea eliminada
Sub Flujo	(S-1) Salir se va la Pantalla (P-1) (S-2)Realizar Solicitud , se va a la Pantalla (P-8)
Excepciones	Ninguna



Pantalla Autorizaciones (P-12)

Caso de Uso	Autorizar Solicitud
Actores	Usuario, BD
Tipo	Básico
Propósito	Autorizar los datos de una solicitud registrada
Resumen	Autoriza los datos de una solicitud que se encuentra en la BD
Pre -Condiciones	La solicitud debe de estar registrada en la BD
Post Condiciones	La solicitud será autorizada y se pasara al modulo de cotizacion Pantalla (P-13)
Flujo Principal	Se eligió la opción Auotirzar Solicitud en la Pantalla (P-12), se da un clic en el botón para que la solicitud sea Autorizada
Sub Flujo	(S-1) Salir se va la Pantalla (P-1)
Excepciones	Ninguna

Caso de Uso	No Autorizar Solicitud
Actores	Usuario, BD
Tipo	Básico
Propósito	No Autorizar los datos de una solicitud registrada
Resumen	No Autoriza los datos de una solicitud que se encuentra en la BD y elimina el registro en el modulo de autorizaciones y de solicitudes
Pre -Condiciones	La solicitud debe de estar registrada en la BD
Post Condiciones	La solicitud será No autorizada y se eliminara de el Modulo de Autorizaciones (P-12) y de Solicitudes (P-7)
Flujo Principal	Se eligió la opción No Auotirzar Solicitud en la Pantalla (P-12), se da un clic en el botón para que la solicitud sea Autorizada
Sub Flujo	(S-1) Salir se va la Pantalla (P-1)
Excepciones	Ninguna

SISTEMA DE GESTION PARA LA ADOUSIACIÓN DE BIENES Y SERVICIOS

PROYECTO TERMINAL 2

MENU
[Solicitudes](#)
[Autorizaciones](#)
[Cotización](#)
[Proveedores](#)

Cotizaciones

Cotizaciones						
id Cotizacion	Nombre	Fecha Cotización	Forma de Pago	Producto (s)	Cantidad	Proveedores
3	gerardo	14/09/2011	efectivo	hojas	5	OFFICE MAX

[salir](#)

Alumna:Stephanie Varela Peña

Pantalla Cotizaciones (P-12)

SISTEMA DE GESTION PARA LA ADOUSIACIÓN DE BIENES Y SERVICIOS

PROYECTO TERMINAL 2

MENU
[Solicitudes](#)
[Autorizaciones](#)
[Cotización](#)
[Proveedores](#)

Realizar Cotización

Datos	
Nombre	<input type="text"/>
Fecha de Cotización	<input type="text"/> <input type="button" value="Calendar"/>
Forma de Pago	<input type="text"/>
Producto(s)	<input type="text"/>
Cantidad	<input type="text"/>
Proveedores	<input type="text"/>

Pantalla Realizar Cotizacion (P-13)

Caso de Uso	Realizar Cotización
Actores	Usuario, BD
Tipo	Básico
Propósito	Que el usuario ingrese los datos para realizar una cotización
Resumen	Registrar una nueva cotización
Pre -Condiciones	No debe de existir la solicitud en BD
Post Condiciones	Registrar los datos necesarios para realizar una cotización
Flujo Principal	Se eligió la opción Realizar Cotización en la pantalla (P-12), se llenan los campos correspondientes en (P-13) y al elegir Realizar Cotización se registra en la base de datos.
Sub Flujo	(S-1) Cancelar. Se regresa a la pantalla (P-12)
Excepciones	(E-1) En caso de no llenar un campo solicitado, se deberá mencionar al usuario que lo ingrese.

Pantalla Borrar Cotizacion (P-12)

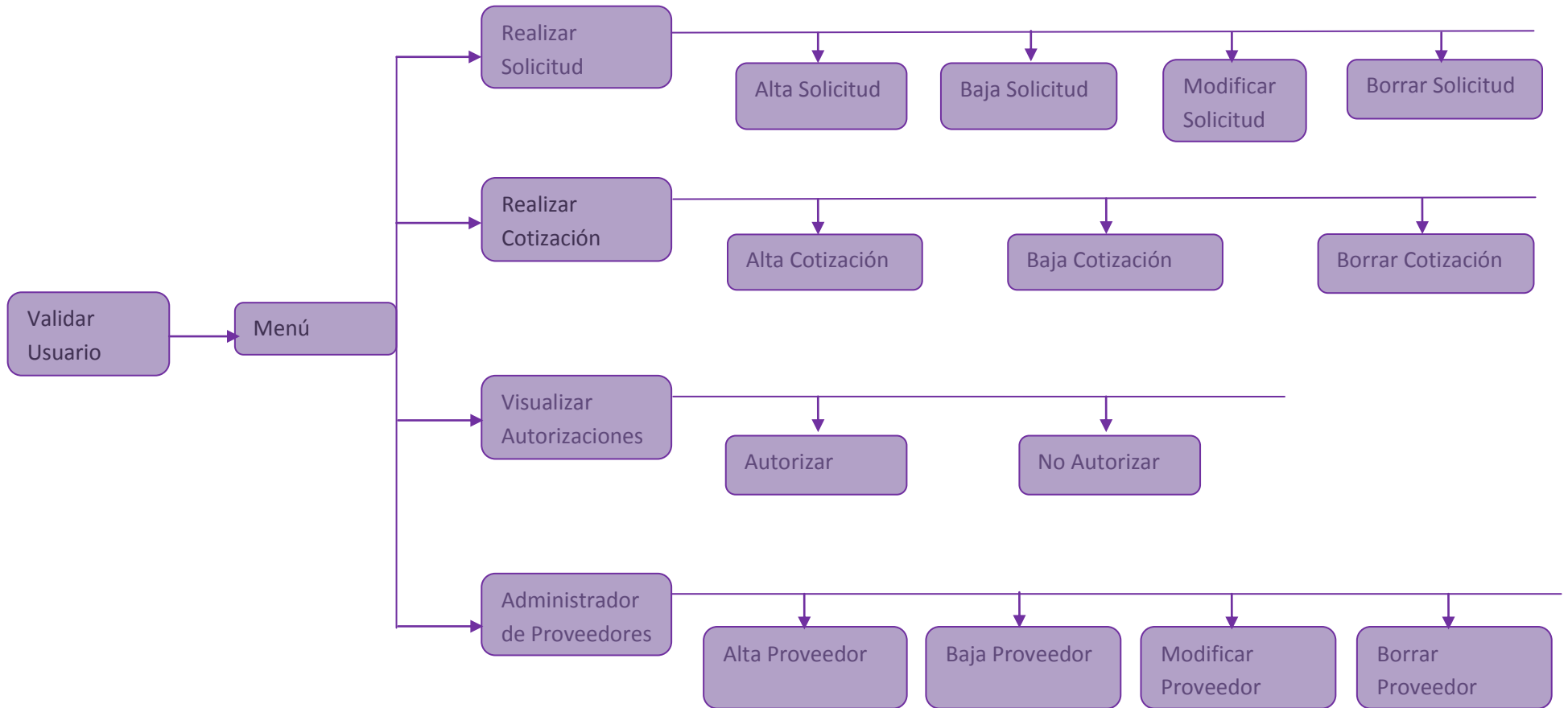
Caso de Uso	Borrar Cotización
Actores	Usuario, BD
Tipo	Básico
Propósito	Borrar los datos de una cotización registrada
Resumen	Borra los datos de una cotización que se encuentra en la BD
Pre -Condiciones	La cotización debe de estar registrada en la BD
Post Condiciones	La cotización será eliminada de la BD y ya no se visualizara en la Pantalla (P-14)
Flujo Principal	Se eligió la opción Borrar Cotización en la Pantalla (P-12), se da un clic en el botón para que la Cotización sea eliminada
Sub Flujo	(S-1) Salir se va la Pantalla (P-1) (S-2)Realizar Cotización , se va a la Pantalla (P-13)
Excepciones	Ninguna

Documentación de Actores

Actor	Usuario
Casos	Validar Usuario , Realizar Solicitud, Realizar Cotización, Visualizar Autorizaciones, Administrador de Proveedores
Tipo	Básico
Descripción	Es el actor principal y representa a cualquier persona que desee utilizar el sistema de gestión de un establecimiento de video

Actor	Base de Datos
Casos	Validar Usuario , Realizar Solicitud, Realizar Cotización, Visualizar Autorizaciones, Administrador de Proveedores
Tipo	Secundario
Descripción	Representa una base de Datos donde se guarda toda la información relacionada con el Sistema de Gestión para la Adquisición de Bienes y Servicios

Mapa de Navegación



Código Fuente

- Modulo Validar Usuario

→ Action

LoginAction.java

```
/**
 *
 */
package com.proyectoterminal.pt2.action;

import com.proyectoterminal.pt2.data.UsuarioDTO;
import com.proyectoterminal.pt2.logica.AdministradorUsuarios;

import com.opensymphony.xwork2.ActionSupport;
import org.apache.struts2.interceptor.SessionAware;

import java.util.Map;

/**
 * @author steph
 */
public class LoginAction extends ActionSupport implements SessionAware
{
    private UsuarioDTO _usuario;
    private Map _sessionData;
    private AdministradorUsuarios _administradorUsuarios;

    public String forma_login()
    {
        _sessionData.remove( "usuario" );
        return "forma_login";
    }

    public String login() throws Exception
    {
        _usuario = _administradorUsuarios.leerUsuarioLoginPassword(
            _usuario.getLogin(), _usuario.getPassword() );
        if ( _usuario == null )
        {
            addActionMessage( "Usuario Incorrecto" );
            _sessionData .put( "usuario" , new UsuarioDTO() ) ;
            return "error";
        }
        _sessionData .put( "usuario" , _usuario ) ;
        //NOOOOreturn "menu";
        if(_usuario.getPerfil() == 1)
            return "menu";
        else
    }
}
```

```

return "error";
}

public String salir() throws Exception
{
    sessionData.put( "usuario", null );
    return "error";
}

public String menuRedirect() throws Exception
{
    _usuario = (UsuarioDTO) _sessionData.get( "usuario" );
    if(_usuario != null)
    {
        if (_usuario.getPerfil() == 1)
            return "menu";
        else
            return "forma_login";
    }
    else return "error";
}

public void setSession( Map sessionData )
{
    _sessionData = sessionData;
}

public void setAdministradorUsuarios(
AdministradorUsuarios administradorUsuarios )
{
    _administradorUsuarios = administradorUsuarios;
}
public UsuarioDTO getUsuario()
{
    return _usuario;
}
public void setUsuario( UsuarioDTO usuario )
{
    _usuario = usuario;
}

}

```

→ Data

UsuarioDTO.java

```
/**
 *
 */
package com.proyectoterminal.pt2.data;

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

/**
 * @author steph
 */
@Entity @Table( name="usuarios" )
public class UsuarioDTO implements Serializable
{
    private int _id;
    private String _nombre;
    private String _login;
    private String _password;
    private int _perfil;

    //////////////////////////////////////////////////METODOS ACCESORS//////////////////////////////////////
    @Id @Column( name="id_usuario" ) @GeneratedValue( strategy=IDENTITY )
    public int getId()
    {
        return _id;
    }

    @Column( name="nombre" )
    public String getNombre()
    {
        return _nombre;
    }
    public void setId( int id )
    {
        _id = id;
    }
    public void setNombre( String nombre )
    {
        _nombre = nombre;
    }
    @Column( name="login" )
    public String getLogin()
    {
        return _login;
    }

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

```

}

@Column( name="Perfil" )
public int getPerfil()
{
    return _perfil;
}

public void setLogin( String login )
{
    _login = login;
}

public void setPassword( String password )
{
    _password = password;
}

public void setPerfil( int perfil )
{
    _perfil = perfil;
}
}

```

→ **Lógica**

AdministradorUsuarios.java

```

package com.proyectoterminal.pt2.logica;
import com.proyectoterminal.pt2.data.UsuarioDTO;

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

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

    @Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
    public List<UsuarioDTO> leerUsuarios( );

    @Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
    public Map<Long, String> cargarUsuarios( );

    @Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
    public UsuarioDTO leerUsuarioLoginPassword(String login, String pass);
}

```

→ Logica.local

AdministradorUsuariosImpl.java

```
/**
 *
 */
package com.proyectoterminal.pt2.logica.local;

import com.proyectoterminal.pt2.data.UsuarioDTO;
import com.proyectoterminal.pt2.logica.AdministradorUsuarios;

import org.hibernate.*;

import java.util.*;

/**
 * @author steph
 */
public class AdministradorUsuariosImpl implements AdministradorUsuarios
{

    private SessionFactory _sessionFactory;

    @Override
    public List<UsuarioDTO> leerUsuarios()
    {
        Session session = _sessionFactory.getCurrentSession();
        Query query = session.getNamedQuery( "leer_usuarios" );
        return query.list();
    }

    @Override
    public Map<Long, String> cargarUsuarios()
    {
        List<UsuarioDTO> t = leerUsuarios();
        Iterator<UsuarioDTO> i = t.iterator();
        UsuarioDTO tt = new UsuarioDTO();
        Map<Long,String> mapa = new TreeMap<Long,String>();
        while(i.hasNext())
        {
            tt = i.next();
            mapa.put( new Long(tt.getId()),tt.getNombre() );
        }
        return mapa ;
    }

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

    private Session getSession()
```

```

    {
    return _sessionFactory.getCurrentSession();
    }

    @Override
    public UsuarioDTO leerUsuarioLoginPassword( String login, String
password )
    {
    Session session = _sessionFactory.getCurrentSession();
    Query query = session.getNamedQuery( "leer_usuario_login_pass" );
    query.setString( 0, login );
    query.setString( 1, password );
    List<UsuarioDTO> lista = query.list();
    return lista.isEmpty() ? null : lista.get( 0 );
    }
}

```

→ JSP

Login.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" %>
<html>
<head>
<title>Bienvenido</title>
<link rel="stylesheet"
href="{pageContext.request.contextPath}/css/estilo.css">
</head>
<body>
<table border="0" cellspacing="11" cellpadding="0" width="600">
<tr>
<td valign="top" align="center" >
<div class="step">
Bienvenido al Sistema para la Adquisición de Bienes y Servicios
</div>
<s:form action="login_login" method="post">
<table id="border">
<tr>
<td colspan="2" align="center">
Proporciona tu cuenta de usuario y contrase&ntilde;a
</td>
</tr>
<tr><td align="center"><table>
<tr class="form">
<td align="right" class="formTit">
<div class="label"> Cuenta de Usuario: </div>
</td>
<td>
<s:textfield name="usuario.login" size="25" theme="simple"/>

```

```

</td>
</tr>
<tr class="form">
<td align="right" align="right" class="formTit">
<div class="label">Contrase&ntilde;a:</div>
</td>
<td>
<s:password name="usuario.password" size="25" theme="simple"/>
</td>
</tr>
<tr>
<td colspan="2" align="center">
<s:submit value="Enviar" theme="simple"/>
</td>
</tr>
</table>
</td>
</tr>
</table>
</s:form>
</td>
</tr>
</table>
</body>
</html>

```

Menu.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>Menu</title>
<link rel="stylesheet" href="css/estilo.css">
</head>
<body>
    <table border="0" cellspacing="0" cellpadding="0" width="800">

        <td width="25">
        </td>
        <td valign="top" align="center" width="625">
            <div class="section">
                Bienvenido al Sistema de Adquisición de Bienes y Servicios
            </div>
            <hr>
            <form method="get" action="http://aplicacion.com">
                <div class="step">
                    MENU
                </div>
            <hr>
            <div class="instructions">

```


El sistema contiene los siguientes Modulos:
</div>
<hr>

```
<table border="0" cellspacing="0" cellpadding="0" width="100%">
<tr>
<td align="left" colspan="3" height="20"><br>

<ul>
<li>
<a
href="http://localhost:8080/pt2Web/pt2/solicitudes_listar.action">Solicit
ud y Autorización </a></li>

<li>
<a
href="http://localhost:8080/pt2Web/pt2/cotizaciones_listar.action">Cotiza
ción</a></li>
<li>
<a
href="http://localhost:8080/pt2Web/proveedores_listar.action">Proveedores
</a></li>
</ul>
</tr>
</td>
</table>

</table>
</body>
</html>
```

- Modulo Realizar Solicitud y Autorización

→ Action

SolicitudesAction.java

```

/**
 *
 */
package com.proyectoterminal.pt2.action;

import com.proyectoterminal.pt2.data.SolicitudDTO;
import com.proyectoterminal.pt2.logica.AdministradorSolicitudes;
import com.proyectoterminal.pt2.logica.local.FiltroSolicitudes;

import com.opensymphony.xwork2.ActionSupport;
import com.opensymphony.xwork2.Preparable;

import org.apache.struts2.interceptor.SessionAware;

import java.util.*;

/**
 * @author steph
 */
public class SolicitudesAction extends ActionSupport implements
SessionAware, Preparable
{

    private SolicitudDTO _solicitud;
    private SolicitudDTO _solicitudSave;
    private List<SolicitudDTO> _solicitudes = new ArrayList<SolicitudDTO>();
    private long _id;

    private FiltroSolicitudes _filtro;

    private Map _sessionData;
    private AdministradorSolicitudes _administradorSolicitudes;

    public String listar()
    {
        _sessionData.put( "idSolicitud" , null );
        FiltroSolicitudes filter = new FiltroSolicitudes();
        filter.setNombre( "" );
        _solicitudes = _administradorSolicitudes.leerSolicitudes( filter );

        return "listar";
    }
}

```

```

//AUTORIZACION
public String listar1()
{
    _sessionData .put( "idSolicitud" , null ) ;
    FiltroSolicitudes filter = new FiltroSolicitudes();
    filter.setNombre( "" );
    _solicitudes = _administradorSolicitudes.leerSolicitudes( filter );

    return "listarautorizacion";
}

public String ver()
{
    if( _id == 0)
        _id = ((Long)_sessionData.get( "idSolicitud" )).longValue();
    _solicitud = _administradorSolicitudes.leerSolicitudes( _id );
    _sessionData .put( "idSolicitud" , id );

    return "ver";
}

public String ver2()
{
    _id = ((Long)_sessionData.get( "idSolicitud" )).longValue();
    _solicitud = _administradorSolicitudes.leerSolicitudes( _id );
    _sessionData .put( "idSolicitud" , id );
    return "ver";
}

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

public String insertar()
{
    //inserta a la base de datos
    System.out.println("insertando en la base");
    _administradorSolicitudes.insertarSolicitud( _solicitudSave );
    addActionMessage( "La solicitud ha sido insertado" );

    return "listar_redirect";
}

public String forma_actualizar()
{
    // _proveedor.setId( ((Long)_sessionData.get( "idProveeedor"
    )).longValue() );
    _solicitud = _administradorSolicitudes.leerSolicitudes(
_solicitud.getId() );
    _solicitudSave = cargarSolicitud( _solicitud );

    return "forma_actualizar";
}

```

```

public String actualizar()
{
//actualiza a la base de datos
System.out.println("actualizando en la base");
_administradorSolicitudes.actualizarSolicitud( _solicitudSave );
_sessionData .put( "idSolicitud" , _solicitudSave .getId() ) ;
return "ver_redirect";
}

public String borrar()
{
System.out.println("borrando en la base");
_administradorSolicitudes.borrarSolicitud( _id );

return "listar_redirect";
}

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

public SolicitudDTO getSolicitudSave()
{
return _solicitudSave;
}
public void setSolicitudSave(SolicitudDTO solicitudSave )
{
_solicitudSave = solicitudSave;
}

public FiltroSolicitudes getFiltro()
{
return _filtro;
}
public void setFiltro( FiltroSolicitudes filtro )
{
_filtro = filtro;
}

public SolicitudDTO getSolicitud()
{
return _solicitud;
}
public void setSolicitud( SolicitudDTO solicitud )
{
_solicitud = solicitud;
}
public List<SolicitudDTO> getSolicitudes()
{
return _solicitudes;
}
public void setSolicitudes( List<SolicitudDTO> solicitudes )

```

```

    {
        _solicitudes = solicitudes;
    }
    public long getId()
    {
        return _id;
    }
    public void setId( long id )
    {
        _id = id;
    }

    public void setSession( Map sessionData )
    {
        _sessionData = sessionData;
    }

    public void prepare() throws Exception
    {
        _filtro = ( FiltroSolicitudes ) _sessionData.get( "filtroSolicitudes"
);
        if( _filtro == null )
        {
            _filtro = new FiltroSolicitudes();
            _sessionData .put( "filtroSolicitudes" , _filtro ) ;
        }
    }

    public void setAdministradorSolicitudes (AdministradorSolicitudes
administradorSolicitudes)
    {
        _administradorSolicitudes = administradorSolicitudes;
    }

    private SolicitudDTO cargarSolicitud( SolicitudDTO solicitud )
    {
        SolicitudDTO s = new SolicitudDTO();
        s.setId( solicitud.getId() );
        s.setProducto( solicitud.getProducto() );
        s.setNombre( solicitud.getNombre() );
        s.setCantidad( solicitud.getCantidad() );
        s.setAdquisicion( solicitud.getAdquisicion() );
        s.setFecha( solicitud.getFecha() );

        return s;
    }
}

```

→ Data

SolicitudDTO.java

```
/**
 *
 */
package com.proyectoterminal.pt2.data;

import javax.persistence.*;
import static javax.persistence.GenerationType.IDENTITY;
import java.io.Serializable;
import java.util.Calendar;

/**
 * @author steph
 */
@Entity @Table( name="solicitudes" )
public class SolicitudDTO implements Serializable
{
    private long _id;
    private String _producto;
    private int _cantidad;
    private Calendar _fecha = Calendar.getInstance();
    private String _nombre;
    private int _adquisicion;

    @Id @Column( name="id" ) @GeneratedValue( strategy=IDENTITY )
    public long getId()
    {
        return _id;
    }
    @Column( name="producto" )
    public String getProducto()
    {
        return _producto;
    }
    @Column( name="cantidad" )
    public int getCantidad()
    {
        return _cantidad;
    }
    @Column( name="fecha_solicitud" )
    public Calendar getFecha()
    {
        return _fecha;
    }
    @Column( name="nombre_solicitante" )
    public String getNombre()
    {
        return _nombre;
    }
}
```

```

@Column( name="num_adquisicion" )
public int getAdquisicion()
{
    return _adquisicion;
}
public void setId( long id )
{
    _id = id;
}
public void setProducto( String producto )
{
    _producto = producto;
}
public void setCantidad( int cantidad )
{
    _cantidad = cantidad;
}
public void setFecha( Calendar fecha )
{
    _fecha = fecha;
}
public void setNombre( String nombre )
{
    _nombre = nombre;
}
public void setAdquisicion( int adquisicion )
{
    _adquisicion = adquisicion;
}
}

```

→ Interceptores

SolicitudesInterceptor.java

```

/**
 *
 */

package com.proyectoterminal.pt2.interceptors;

import com.proyectoterminal.pt2.data.UsuarioDTO;

import com.opensymphony.xwork2.ActionInvocation;
import com.opensymphony.xwork2.interceptor.Interceptor;

import java.util.Map;

/**
 * @author steph
 */
public class SolicitudesInterceptor implements Interceptor
{
    @Override

```

```

public void destroy()
{
    System.out.println("Destruyendo SolicitudInterceptor...");
}

@Override
public void init()
{
    System.out.println("Inicializando SolicitudesInterceptor...");
}

@Override
public String intercept( ActionInvocation invocation ) throws Exception
{
    Map<?, ?> session = invocation.getInvocationContext().getSession();
    UsuarioDTO usuario = ( UsuarioDTO )session.get( "usuario" );
    if( usuario != null &&
        usuario.getPerfil() == 1 )
        return invocation.invoke();
    else return "error";
}
}
}

```

→ Logica

AdministradorSolicitudes.java

```

/**
 *
 */
package com.proyectoterminal.pt2.logica;
import com.proyectoterminal.pt2.data.ProveedorDTO;
import com.proyectoterminal.pt2.data.SolicitudDTO;
import com.proyectoterminal.pt2.logica.local.*;

import org.springframework.transaction.annotation.Propagation;
import org.springframework.transaction.annotation.Transactional;

import java.util.List;

/**
 * @author steph
 */
@Transactional( rollbackFor=Exception.class )
public interface AdministradorSolicitudes
{
    @Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
    public SolicitudDTO leerSolicitudes(long id);
}

```



```

@Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
public List<SolicitudDTO> leerSolicitudesQuery();

@Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
public long contarSolicitudes();

@Transactional (propagation = Propagation.SUPPORTS, readOnly= true)
public List <SolicitudDTO> leerSolicitudes(FiltroSolicitudes filter);

@Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
public List<SolicitudDTO> buscarTemasCriterioNombre( FiltroSolicitudes
filter );

public long insertarSolicitud(SolicitudDTO solicitud);

public void actualizarSolicitud(SolicitudDTO solicitud);

public void borrarSolicitud(long id);
}

```

→ Logica.local

AdministradorSolicitudesImpl.java

```

/**
 *
 */
package com.proyectoterminal.pt2.logica.local;

import com.proyectoterminal.pt2.data.SolicitudDTO;
import com.proyectoterminal.pt2.logica.AdministradorSolicitudes;

import org.hibernate.*;
import org.hibernate.criterion.Order;
import org.hibernate.criterion.Property;

import java.util.List;

/**
 * @author steph
 */
public class AdministradorSolicitudesImpl implements
AdministradorSolicitudes
{
    ///////////////variables de parametros////////////////
    private int _estado;
    private String _salida;
    private List<Integer> _indices;
    private SolicitudDTO _solicitud;
    private AdministradorSolicitudes administradorSolicitudes;
}

```

```

////////// PROPIEDADES //////////
public void setEstado( int estado )
{
    _estado = estado;
    System.out.println(_estado);
}

public void setSalida( String salida )
{
    _salida = salida;
    System.out.println(_salida);
}

public void setIndices( List<Integer> indices )
{
    _indices = indices;
    System.out.println(_indices.get( 0 ).intValue());
    System.out.println(_indices.get( 1 ).intValue());
}

public void setSolicitud( SolicitudDTO solicitud )
{
    this._solicitud = solicitud;

    System.out.println(_solicitud.getId());
    System.out.println(_solicitud.getNombre());
}
}
//////////

//////////variables de configuracion//////////
private SessionFactory _sessionFactory;

public SolicitudDTO leerSolicitudes(long id)
{
    Session session = getSession();
    return ( SolicitudDTO )session.get( SolicitudDTO.class, id );
}

public long insertarSolicitud( SolicitudDTO solicitud )
{
    Session session = getSession();
    session.save( solicitud );
    return solicitud.getId();
}

public void setSessionFactory( SessionFactory sessionFactory )

```

```

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

public void actualizarSolicitud( SolicitudDTO solicitud )
{
    Session session = getSession();
    session.update( solicitud );
}

public void borrarSolicitud( long id )
{
    Session session = getSession();
    SolicitudDTO solicitud =
(SolicitudDTO)session.load(SolicitudDTO.class,id);
    session.delete(solicitud);
}

public List<SolicitudDTO> leerSolicitudesQuery()
{
    Session session = getSession();
    Query query = session.createQuery( "SELECT s FROM SolicitudDTO s" );
    return query.list();
}

@Override
public long contarSolicitudes()
{
    Session session = getSession();
    Query query = session.getNamedQuery( "contar_solicitudes" );
    long count = ( Long )query.uniqueResult();
    return count;
}

@Override
public List<SolicitudDTO> leerSolicitudes( FiltroSolicitudes filter )
{
    Session session = getSession();
    Query query = session.getNamedQuery( "leer_solicitudes_filtro" );
    query.setString( "nombre", "%" + filter.getNombre() + "%" );
    return query.list();
}

@Override
public List<SolicitudDTO> buscarTemasCriterioNombre( FiltroSolicitudes
filter )

```

```

    {
        Criteria criteria = getSession().createCriteria( SolicitudDTO.class
);
        String nombre = "%" + filter.getNombre() + "%";
        criteria.add( Property.forName( "nombre" ).like( nombre ) );
        criteria.addOrder( Order.asc( "id" ) );
        return criteria.list();
    }
}

```

FiltroSolicitudes.java

```

package com.proyectoterminal.pt2.logica.local;

public class FiltroSolicitudes
{
    private String _nombre;
    public String getNombre()
    {
        return _nombre;
    }
    public void setNombre( String nombre )
    {
        _nombre = nombre;
    }
}

```

→ JSP

Forma_actualizar.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" %>
<html>
<head>

<title>forma_actualizar</title>
<link href="{pageContext.request.contextPath }/css/datepicker.css"
rel="stylesheet" type="text/css"/>
<link rel="stylesheet"
href="{pageContext.request.contextPath}/css/style.css">
<script type="text/javascript"
src="{pageContext.request.contextPath}/js/datepicker.js">
</script>

```

```

</head>
<body>
<s:form action="solicitudes_actualizar" method="post">

<table border="0" cellspacing="0" cellpadding="0" width="100%">

<tr>
<td align="center" valign="top" width="150" class="menu">

</td>
<td width="25">
</td>
<td valign="top" align="center" width="625">
<div class="section">
Actualizar Solicitud
</div>
<table align="center" width="80%" border="1">
<tr>
<td class="form" align="center" colspan="2">
<div class="label">Datos</div>
</td>
</tr><!--

<tr>
<td class="form" align="center">
<div class="label">Numero de Pedido</div></td>
<td>
<s:hidden name="solicitudSave.id" />
<s:textfield name="solicitudSave.id" size="25" cssClass="fontForma"
/></td>
</tr>

--><tr>
<td class="form" align="center"><div class="label">Producto</div></td>
<td><s:textfield name="solicitudSave.producto" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Cantidad</div></td>
<td><s:textfield name="solicitudSave.cantidad" size="25"
cssClass="fontForma" />
</td>
</tr>
<!--

<tr>
<td class="form" align="center"><div class="label">fecha</div></td>
<td>
<s:textfield name="temaSave.fecha.time" cssClass="w16em dateFormat-d-s1-
m-s1-Y" size="15" theme="simple" />
</td>
</tr>

-->

```

```

<tr>
<td class="form" align="center"><div class="label">Nombre
Solicitante</div></td>
<td><s:textfield name="solicitudSave.nombre" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Numero de
Adquisición</div></td>
<td><s:textfield name="solicitudSave.adquisicion" size="25"
cssClass="fontForma" />
</td>
</tr>

</table>
<table>

<tr>
<td align="center">
<s:submit value="Actualizar Solicitud"/>
</td>

<a
href="{pageContext.request.contextPath}/pt2/solicitudes listar.action"><
/a>

<td align="center">
<input type="button" value=" Cancelar "
onClick="window.location='{pageContext.request.contextPath}/pt2/solicitu
des_ver.action'">

</td>
</tr>

</table>

<br>
</td>
</tr>

</table>
</s:form>
</body>
</html>

```

Forma_insertar.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" %>
<html>
<head>

<title>Realizar Solicitud</title>
<link rel="stylesheet"
href="{pageContext.request.contextPath}/css/style.css">
<link href="../css/datepicker.css" rel="stylesheet" type="text/css" />
<script type="text/javascript"
src="{pageContext.request.contextPath}/js/livevalidation.js">
</script>

<script type="text/javascript"
src="{pageContext.request.contextPath}/js/datepicker.js">
</script>

</head>
<body>
<table border="0" cellspacing="0" cellpadding="0" width="100%">
<tr>
<td align="center" colspan="3" height="20">

</td>
</tr>
<tr>
<td align="center" valign="top" width="150" class="menu">

</td>
<td width="25">
</td>
<td valign="top" align="center" width="625">
<div class="section">
Realizar una Solicitud
</div>
<hr>
<s:form action="solicitudes_insertar" method="post">
<table align="center" width="80%" border="1">

<tr>
<td class="form" align="center" colspan="2">
<div class="label">Datos</div>
</td>
</tr>

<!--

<tr>
<td class="form" align="center"><div class="label">Numero de
Pedido</div></td>
```

```

<td><s:textfield name="solicitudSave.id" size="25" cssClass="fontForma"
/>
</td>
</tr>

-->

<tr>
<td class="form" align="center"><div class="label">Producto</div></td>
<td><s:textfield name="solicitudSave.producto" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Cantidad</div></td>
<td><s:textfield name="solicitudSave.cantidad" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">fecha de
Solicitud</div></td>
<td>
<s:textfield name="solicitudSave.fecha.time" cssClass="w16em dateformat-
d-sl-m-sl-Y"
size="15" theme="simple" /></td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Nombre
Solicitante</div></td>
<td><s:textfield name="solicitudSave.nombre" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Numero de
Adquisición</div></td>
<td><s:textfield name="solicitudSave.adquisicion" size="25"
cssClass="fontForma" />
</td>
</tr>

</table>
<table>

<tr>
<td align="center">

```



```

<s:submit value="Insertar Solicitud"/>
</td>

<a
href="{pageContext.request.contextPath}/pt2/solicitudes listar.action"><
/a>

<td align="center">
<input type="button" value=" Cancelar "
onClick="window.location='{pageContext.request.contextPath}/pt2/solicitu
des_listar.action'">

</td>
</tr>

</table>
</s:form>
<br>
</td>
</tr>
<tr>
<td align="center" colspan="3">

</td>
</tr>
</table>
</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" %>
<html>
<head>
<title>REALIZAR SOLICITUD</title>
<link rel="stylesheet"
href="{pageContext.request.contextPath}/css/estilo.css">
<link href="../css/datepicker.css" rel="stylesheet" type="text/css" />
<script type="text/javascript" src="{pageContext.request.contextPath}/js/datepicker.js"></script>

</head>
<body>
<table border="0" cellspacing="0" cellpadding="0" width="100%">

<tr>
<td align="center" valign="top" width="150" class="menu">

</td>
<td width="25">
</td>
<td valign="top" align="center" width="625">
<div class="section">
Solicitudes

```

```

</div>
<hr>
<s:form action="solicitudes_listar" method="post">
</s:form>
<table align="center" width="80%" border="1">
<tr>
<td colspan="11" class="form" align="center">
<div class="label">Solicitudes</div>
</td>
</tr>
<tr>
<td class="form" align="center"><div class="label">id
Solicitud</div></td>
<td class="form" align="center"><div class="label">producto </div></td>
<td class="form" align="center"><div class="label">cantidad </div></td>
<td class="form" align="center"><div class="label">fecha</div></td>
<td class="form" align="center"><div class="label">nombre
solicitante</div></td>
<td class="form" align="center"><div class="label">adquisicion</div></td>
</td>
</tr>

<c:forEach var="solicitud" items="${ solicitudes }">
<tr>
<td>
<a href="${pageContext.request.contextPath}/solicitudes_ver.action?id=${
solicitud.id }">${ solicitud.id }</a>

</td>

<td>${ solicitud.producto }</td>
<td>${ solicitud.cantidad }</td>
<td><fmt:formatDate pattern="dd/MM/yyyy" value="${ solicitud.fecha.time
}"/></td>
<td>${ solicitud.nombre }</td>
<td>${ solicitud.adquisicion }</td>

<td align="center">
<input type="button" value=" Borrar Solicitud"
onClick="window.location='${pageContext.request.contextPath}/pt2/solicitu
des_borrar.action?id=${ solicitud.id }'">
</td></tr>

</c:forEach>

</table>
<center>
<table>
<tr>
<td align="center">
<input type="button" value=" Realizar Solicitud "
onClick="window.location='${pageContext.request.contextPath}/pt2/solicitu
des_forma_insertar.action'">

</td>
<td align="center">

```

```

<a href="{pageContext.request.contextPath}/pt2/login_salir.action">
salir</a>

</td>

</table>
</center>
<br>
</td>
</tr>

</table>
</body>
</html>

```

Ver.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" %>

<html>
  <head>
    <title>Ver Solicitud</title>
    <link rel="stylesheet" href="{pageContext.request.contextPath
}/css/estilo.css">
    <!--<link href="../css/datepicker.css" rel="stylesheet" type="text/css"
/>
    <script type="text/javascript"
src="{pageContext.request.contextPath}/js/datepicker.js"></script>

--></head>
  <body>
    <table border="0" cellspacing="0" cellpadding="0" width="100%">

      <tr>
        <td align="center" valign="top" width="150" class="menu">

          </td>
          <td width="25">
          </td>
          <td valign="top" align="center" width="625">

            <div class="section">
              Ver Solicitud
            </div>

            <hr>

```

```

<s:form action="solicitudes_forma_actualizar" method="post">

    <table align="center" width="80%" border="1">

        <tr>
            <td class="form" align="center" colspan="2">
                <div class="label">Datos</div>
            </td>
        </tr>

        <tr>
            <td class="form" align="center"><div class="label">Numero
de Pedido</div></td>
            <td>${ solicitud.id }</td>
        </tr>

        <tr>
            <td class="form" align="center"><div
class="label">Producto</div></td>
            <td><s:hidden name="solicitud.id" />${ solicitud.producto
}</td>
        </tr>

        <tr>
            <td class="form" align="center"><div
class="label">Cantidad</div></td>
            <td>${ solicitud.cantidad}</td>
        </tr>

        <tr>
            <td class="form" align="center"><div class="label">Fecha
de Solicitud <br>(La fecha no se puede modificar)
            </div></td>
            <td><fmt:formatDate pattern="dd/MM/yyyy" value="${
solicitud.fecha.time }"/></td>
        </tr>

        <tr>
            <td class="form" align="center"><div
class="label">Nombre solicitante</div></td>
            <td>${ solicitud.nombre }</td>
        </tr>

        <tr>
            <td class="form" align="center"><div
class="label">Numero de Adquisicion</div></td>
            <td>${ solicitud.adquisicion}</td>
        </tr>

    </table>
</table>

    <tr>
        <td align="center">
            <s:submit value="Actualizar Solicitud"/>
        </td>
    </tr>

```

```

</td>
<td align="center">
<a
href="{pageContext.request.contextPath}/pt2/solicitudes_listar.action?id
Solicitud={ solicitud.id }">
</a>
<td align="center">
<input type="button" value=" Salir"
onClick="window.location='{pageContext.request.contextPath}/pt2/solicitu
des_listar.action'">

</td>
</tr>

</table>

</s:form>
<br>
</td>
</tr>

<tr>

</tr>
</table>
</body>
</html>

```

Autorizaciones

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" %>
<html>
<head>
<title>AUTORIZACIONES</title>
<link rel="stylesheet"
href="{pageContext.request.contextPath}/css/estilo.css">
<link href="../css/datepicker.css" rel="stylesheet" type="text/css" />
<script type="text/javascript" src="{
pageContext.request.contextPath}/js/datepicker.js"></script>

</head>
<body>
<table border="0" cellspacing="0" cellpadding="0" width="100%">

<tr>
<td align="center" valign="top" width="150" class="menu">

</td>

```

```

<td width="25">
</td>
<td valign="top" align="center" width="625">
<div class="section">
Autorizaciones
</div>
<hr>
<s:form action="solicitudes_listar1" method="post">
</s:form>
<table align="center" width="80%" border="1">
<tr>
<td colspan="11" class="form" align="center">
<div class="label">Autorizaciones</div>
</td>
</tr>
<tr>
<td class="form" align="center"><div class="label">id
Solicitud</div></td>
<td class="form" align="center"><div class="label">producto </div></td>
<td class="form" align="center"><div class="label">cantidad </div></td>
<td class="form" align="center"><div class="label">fecha</div></td>
<td class="form" align="center"><div class="label">nombre
solicitante</div></td>
<td class="form" align="center"><div class="label">adquisicion</div></td>
</td>
</tr>

<c:forEach var="solicitud" items="{ solicitudes }">
<tr>
<td>
<a href="{pageContext.request.contextPath}/solicitudes_ver.action?id={
solicitud.id }">${ solicitud.id }</a>

</td>

<td>${ solicitud.producto }</td>
<td>${ solicitud.cantidad }</td>
<td><fmt:formatDate pattern="dd/MM/yyyy" value="{ solicitud.fecha.time
}"/></td>
<td>${ solicitud.nombre }</td>
<td>${ solicitud.adquisicion }</td>

<td align="center">
<input type="button" value=" Autorizar Solicitud"
onClick="window.location='{pageContext.request.contextPath}/pt2/cotizaci
ones_listar.action?id={ cotizacion.id }'">
</td>

<td align="center">
<input type="button" value=" No Autorizar Solicitud"
<%out.print("Al eliminar la Autorizacion, la solicitud quedara eliminada
tambi\u00e9n");%>
onClick="window.location='{pageContext.request.contextPath}/pt2/solicitu
des_borrar1.action?id={ solicitud.id }'">

```

```
</td>

</tr>

</c:forEach>

</table>
<center>
<table>
<tr>
<td align="center">

<a href="{pageContext.request.contextPath}/pt2/login_salir.action">
salir</a>

</td>

</table>
</center>
<br>
</td>
</tr>

</table>
</body>
</html>
```

- Modulo Realizar Cotización

→ Action

CotizacionAction.java

```
/**
 *
 */
package com.proyectoterminal.pt2.action;

import com.proyectoterminal.pt2.data.CotizacionDTO;
import com.proyectoterminal.pt2.logica.AdministradorCotizaciones;
import com.proyectoterminal.pt2.logica.local.FiltroCotizaciones;

import com.opensymphony.xwork2.ActionSupport;
import com.opensymphony.xwork2.Preparable;

import org.apache.struts2.interceptor.SessionAware;

import java.util.*;

/**
 * @author steph
 */
public class CotizacionesAction extends ActionSupport implements
SessionAware, Preparable
{

    private CotizacionDTO _cotizacion;
    private CotizacionDTO _cotizacionSave;
    private List<CotizacionDTO> _cotizaciones = new
ArrayList<CotizacionDTO>();
    private long _id;

    private FiltroCotizaciones _filtro;

    private Map _sessionData;
    private AdministradorCotizaciones _administradorCotizaciones;

    public String listar()
    {
        _sessionData.put( "idCotizacion" , null ) ;
        FiltroCotizaciones filter = new FiltroCotizaciones();
        filter.setNombre( "" );
        _cotizaciones = _administradorCotizaciones.leerCotizaciones( filter
);

        return "listar";
    }
}
```



```

}

public String ver()
{
    if(_id == 0)
        _id = ((Long)_sessionData.get( "idCotizacion")).longValue();
    _cotizacion = _administradorCotizaciones.leerCotizaciones( _id );
    _sessionData .put( "idCotizacion", _id );

    return "ver";
}

public String ver2()
{
    _id = ((Long)_sessionData.get( "idCotizacion" )).longValue();
    _cotizacion= _administradorCotizaciones.leerCotizaciones( _id );
    _sessionData .put( "idCotizacion" , _id ) ;
    return "ver";
}

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

public String insertar()
{
    //inserta a la base de datos
    System.out.println("insertando en la base");
    _administradorCotizaciones.insertarCotizacion( _cotizacionSave );
    addActionMessage( "La Cotizacion ha sido insertado" );

    return "listar_redirect";
}

public String borrar()
{
    System.out.println("borrando en la base");
    _administradorCotizaciones.borrarCotizacion( _id );

    return "listar_redirect";
}

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

public CotizacionDTO getCotizacionSave()
{
    return _cotizacionSave;
}

```

```

public void setCotizacionSave(CotizacionDTO cotizacionSave )
{
    _cotizacionSave = cotizacionSave;
}

public FiltroCotizaciones getFiltro()
{
    return _filtro;
}
public void setFiltro( FiltroCotizaciones filtro )
{
    _filtro = filtro;
}

public CotizacionDTO getCotizacion()
{
    return _cotizacion;
}
public void setSolicitud( CotizacionDTO cotizacion )
{
    _cotizacion = cotizacion;
}
public List<CotizacionDTO> getCotizaciones()
{
    return _cotizaciones;
}
public void setCotizaciones( List<CotizacionDTO> cotizaciones )
{
    _cotizaciones = cotizaciones;
}
public long getId()
{
    return _id;
}
public void setId( long id )
{
    _id = id;
}

public void setSession( Map sessionData )
{
    _sessionData = sessionData;
}

public void prepare() throws Exception
{
    _filtro = ( FiltroCotizaciones ) _sessionData.get( "filtroCotizaciones"
);
    if( _filtro == null )
    {
        _filtro = new FiltroCotizaciones();
        _sessionData .put( "filtroCotizaciones" , filtro ) ;
    }
}

```

```

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

    private CotizacionDTO cargarCotizacion( CotizacionDTO cotizacion )
    {
        CotizacionDTO c = new CotizacionDTO();
        c.setId( cotizacion.getId() );
        c.setNombre( cotizacion.getNombre() );
        c.setFormaPago( cotizacion.getFormaPago() );
        c.setFecha( cotizacion.getFecha() );
        c.setProducto( cotizacion.getProducto() );
        c.setCantidad( cotizacion.getCantidad() );
        c.setProveedores( cotizacion.getProveedores() );

        return c;
    }
}

```

→ Data

CotizacionDTO.java

```

/**
 *
 */
package com.proyectoterminal.pt2.data;

import javax.persistence.*;
import static javax.persistence.GenerationType.IDENTITY;
import java.io.Serializable;
import java.util.Calendar;

/**
 * @author steph
 */
@Entity @Table( name="cotizaciones" )
public class CotizacionDTO implements Serializable
{
    private long _id;
    private String _nombre;
    private Calendar _fecha= Calendar.getInstance();
}

```

```

private String _formaPago;
private String _producto;
private int _cantidad;
private String _proveedores;

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

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

@Column( name="producto" )
public String getProducto()
{
    return _producto;
}

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

@Column( name="fecha" )
public Calendar getFecha()
{
    return _fecha;
}

public void setFecha( Calendar fecha )
{
    _fecha = fecha;
}

public void setProducto( String producto )
{
    _producto = producto;
}

public void setCantidad( int cantidad )
{
    _cantidad = cantidad;
}

public void setNombre( String nombre )
{
    _nombre = nombre;
}

public void setId( long id )
{

```

```

    _id = id;
}

@Column( name="formapago" )
public String getFormaPago()
{
    return _formaPago;
}

@Column( name="proveedores" )
public String getProveedores()
{
    return _proveedores;
}

public void setFormaPago( String formaPago )
{
    _formaPago = formaPago;
}
public void setProveedores( String proveedores )
{
    _proveedores = proveedores;
}
}

```

→ Interceptores

CotizacionesInterceptor.java

```

package com.proyectoterminal.pt2.interceptors;

import com.proyectoterminal.pt2.data.UsuarioDTO;

import com.opensymphony.xwork2.ActionInvocation;
import com.opensymphony.xwork2.interceptor.Interceptor;

import java.util.Map;

public class CotizacionesInterceptor implements Interceptor
{

    @Override
    public void destroy()
    {
        System.out.println("Destruyendo CotizacionesInterceptor...");
    }
}

```

```

    }

    @Override
    public void init()
    {
        System.out.println("Inicializando CotizacionesInterceptor...");
    }

    @Override
    public String intercept( ActionInvocation invocation ) throws Exception
    {
        Map<?, ?> session = invocation.getInvocationContext().getSession();
        UsuarioDTO usuario = ( UsuarioDTO )session.get( "usuario" );
        if( usuario != null &&
            usuario.getPerfil() == 1 )
            return invocation.invoke();
        else return "error";
    }
}

```

→ Logica

AdministradorCotizaciones.java

```

/**
 *
 */
package com.proyectoterminal.pt2.logica;

import com.proyectoterminal.pt2.data.CotizacionDTO;
import com.proyectoterminal.pt2.logica.local.*;

import org.springframework.transaction.annotation.Propagation;
import org.springframework.transaction.annotation.Transactional;

import java.util.List;

/**
 * @author steph
 */
@Transactional( rollbackFor=Exception.class )
public interface AdministradorCotizaciones
{
    @Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
    public CotizacionDTO leerCotizaciones(long id);
}

```

```

@Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
public List<CotizacionDTO> leerCotizacionesQuery();

@Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
public long contarCotizaciones();

@Transactional (propagation = Propagation.SUPPORTS, readOnly= true)
public List <CotizacionDTO> leerCotizaciones(FiltroCotizaciones
filter);

@Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
public List<CotizacionDTO> buscarTemasCriterioNombre(
FiltroCotizaciones filter );

public long insertarCotizacion(CotizacionDTO cotizacion);

public void actualizarCotizacion(CotizacionDTO cotizacion);

public void borrarCotizacion(long id);
}

```

→ Logica.local

AdministradorCotizaciones.java

```

/**
 *
 */
package com.proyectoterminal.pt2.logica.local;

import com.proyectoterminal.pt2.data.CotizacionDTO;

import com.proyectoterminal.pt2.logica.AdministradorCotizaciones;

import org.hibernate.*;
import org.hibernate.criterion.Order;
import org.hibernate.criterion.Property;

import java.util.List;

/**
 * @author steph
 */
public class AdministradorCotizacionesImpl implements
AdministradorCotizaciones
{
    ///////////////variables de parametros////////////////
    private int _estado;
}

```

```

private String _salida;
private List<Integer> _indices;
private CotizacionDTO _cotizacion;
private AdministradorCotizaciones administradorCotizaciones;

////////// PROPIEDADES //////////
public void setEstado( int estado )
{
    _estado = estado;
    System.out.println(_estado);
}

public void setSalida( String salida )
{
    _salida = salida;
    System.out.println(_salida);
}

public void setIndices( List<Integer> indices )
{
    _indices = indices;
    System.out.println(_indices.get( 0 ).intValue());
    System.out.println(_indices.get( 1 ).intValue());
}

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

    System.out.println(_cotizacion.getId());
    System.out.println(_cotizacion.getNombre());
}
//////////

//////////variables de configuracion//////////
private SessionFactory _sessionFactory;

public CotizacionDTO leerCotizaciones(long id)
{
    Session session = getSession();
    return ( CotizacionDTO )session.get( CotizacionDTO.class, id );
}

public long insertarCotizacion( CotizacionDTO cotizacion )
{
    Session session = getSession();
    session.save( cotizacion );
    return cotizacion.getId();
}

```



```

    }

    public void setSessionFactory( SessionFactory sessionFactory )
    {
        _sessionFactory = sessionFactory;
    }
    private Session getSession()
    {
        return _sessionFactory.getCurrentSession();
    }

    public void actualizarCotizacion( CotizacionDTO cotizacion )
    {
        Session session = getSession();
        session.update( cotizacion );
    }

    public void borrarCotizacion( long id )
    {
        Session session = getSession();
        CotizacionDTO cotizacion =
(CotizacionDTO)session.load(CotizacionDTO.class, id);
        session.delete(cotizacion);
    }

    public List<CotizacionDTO> leerCotizacionesQuery()
    {
        Session session = getSession();
        Query query = session.createQuery( "SELECT a FROM CotizacionDTO a" );
        return query.list();
    }

    @Override
    public long contarCotizaciones()
    {
        Session session = getSession();
        Query query = session.getNamedQuery( "contar_cotizaciones" );
        long count = ( Long )query.uniqueResult();
        return count;
    }

    @Override
    public List<CotizacionDTO> leerCotizaciones( FiltroCotizaciones filter
)
    {
        Session session = getSession();
        Query query = session.getNamedQuery( "leer_cotizaciones_filtro" );
        query.setString( "nombre", "%" + filter.getNombre() + "%" );
        return query.list();
    }
}

```

```

@Override
public List<CotizacionDTO> buscarTemasCriterioNombre(
FiltroCotizaciones filter )
{
    Criteria criteria = getSession().createCriteria( CotizacionDTO.class
);
    String nombre = "%" + filter.getNombre() + "%";
    criteria.add( Property.forName( "nombre" ).like( nombre ) );
    criteria.addOrder( Order.asc( "id" ) );
    return criteria.list();
}
}
}

```

FiltroCotizaciones.java

```

package com.proyectoterminal.pt2.logica.local;

public class FiltroCotizaciones
{
    private String _nombre;
    public String getNombre()
    {
        return _nombre;
    }
    public void setNombre( String nombre )
    {
        _nombre = nombre;
    }
}
}

```

→ JSP

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" %>
<html>
<head>

```

```

<title>COTIZACIÓN</title>
<link rel="stylesheet"
href="{pageContext.request.contextPath}/css/estilo.css">
<link href="../css/datepicker.css" rel="stylesheet" type="text/css" />
<script type="text/javascript" src="{
pageContext.request.contextPath}/js/datepicker.js"></script>

</head>
<body>
<table border="0" cellspacing="0" cellpadding="0" width="100%">

<tr>
<td align="center" valign="top" width="150" class="menu">

</td>
<td width="25">
</td>
<td valign="top" align="center" width="625">
<div class="section">
Cotizaciones
</div>
<hr>
<s:form action="cotizaciones_listar" method="post">
</s:form>
<table align="center" width="80%" border="1">
<tr>
<td colspan="11" class="form" align="center">
<div class="label">Cotizaciones</div>
</td>
</tr>
<tr>
<td class="form" align="center"><div class="label">id
Cotizacion</div></td>
<td class="form" align="center"><div class="label">Nombre </div></td>
<td class="form" align="center"><div class="label">Fecha Cotización
</div></td>
<td class="form" align="center"><div class="label">Forma de
Pago</div></td>
<td class="form" align="center"><div class="label">Producto(s)</div></td>
<td class="form" align="center"><div class="label">Cantidad</div></td>
<td class="form" align="center"><div class="label">Proveedores</div></td>
</tr>

<c:forEach var="cotizacion" items="{ cotizaciones }">
<tr>
<td>
<a href="{pageContext.request.contextPath}/cotizaciones_ver.action?id={
cotizacion.id }">{ cotizacion.id }</a>
</td>
<td>{ cotizacion.nombre }</td>
<td><fmt:formatDate pattern="dd/MM/yyyy" value="{ cotizacion.fecha.time
}"/></td>
<td>{ cotizacion.formaPago }</td>
<td>{ cotizacion.producto }</td>
<td>{ cotizacion.cantidad }</td>
<td>{ cotizacion.proveedores }</td>

```

```

<td align="center">
<input type="button" value=" Borrar Cotización "
onClick="window.location='{pageContext.request.contextPath}/pt2/cotizaci
ones_borrar.action?id=${ cotizacion.id }'">
</td>

<td align="center">
<input type="button" value=" Enviar Cotización "
onClick="window.location='{pageContext.request.contextPath}/pt2/cotizaci
ones_listar.action?id=${ cotizacion.id }'">
</td>

</c:forEach>

</table>
<center>
<table>
<tr>
<td align="center">
<input type="button" value=" Realizar Cotización "
onClick="window.location='{pageContext.request.contextPath}/pt2/cotizaci
ones_forma_insertar.action'">

</td>
<td align="center">

<a href="{pageContext.request.contextPath}/pt2/login_salir.action">
salir</a>

</td>

</table>
</center>
<br>
</td>
</tr>

</table>
</body>
</html>

```

Ver.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" %>

```

```

<html>
  <head>
    <title>Ver Cotización</title>
    <link rel="stylesheet" href="{pageContext.request.contextPath
}/css/estilo.css">
    <!--<link href="../css/datepicker.css" rel="stylesheet" type="text/css"
/>
    <script type="text/javascript"
src="{pageContext.request.contextPath}/js/datepicker.js"></script>

    --></head>
  <body>
    <table border="0" cellspacing="0" cellpadding="0" width="100%">

      <tr>
        <td align="center" valign="top" width="150" class="menu">

          </td>
          <td width="25">
          </td>
          <td valign="top" align="center" width="625">

            <div class="section">
              Ver Cotización
            </div>

            <hr>
            <s:form action="cotizaciones_forma_actualizar" method="post">

              <table align="center" width="80%" border="1">

                <tr>
                  <td class="form" align="center" colspan="2">
                    <div class="label">Datos</div>
                  </td>
                </tr>

                <tr>
                  <td class="form" align="center"><div class="label">id
Cotización</div></td>
                  <td>${ cotizacion.id }</td>
                </tr>

                <tr>
                  <td class="form" align="center"><div
class="label">Nombre</div></td>
                  <td>${ cotizacion.nombre}</td>
                </tr>

                <td class="form" align="center"><div class="label">Fecha
de Cotización <br>(La fecha no se puede modificar)
                </div></td>

```

```

        <td><fmt:formatDate pattern="dd/MM/yyyy" value="{
cotizacion.fecha.time }"/></td>
    </tr>
    <tr>
        <td class="form" align="center"><div class="label">Forma
de Pago</div></td>
        <td>${ cotizacion.formaPago }</td>
    </tr>

    <tr>
        <td class="form" align="center"><div
class="label">Producto(s)</div></td>
        <td>${ cotizacion.producto }</td>
    </tr>

    <tr>
        <td class="form" align="center"><div
class="label">Cantidad</div></td>
        <td>${ cotizacion.cantidad }</td>
    </tr>

    <tr>
        <td class="form" align="center"><div
class="label">Proveedores</div></td>
        <td>${ cotizacion.proveedores}</td>
    </tr>

    </table>
<table>

    <tr>
<td align="center">
<s:submit value="Actualizar Cotizacion"/>
</td>
<td align="center">

<a
href="{pageContext.request.contextPath}/pt2/cotizaciones_listar.action?idCotizacion=${ cotizacion.id }">
</a>

<td align="center">
<input type="button" value="Regresar"
onClick="window.location='{pageContext.request.contextPath}/pt2/cotizaci
ones_listar.action'">

</td>
</tr>

    </table>

</s:form>
    <br>
</td>
</tr>

```

```

        <tr>
        </tr>
    </table>
</body>
</html>

```

Forma_actualizar.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" %>
<html>
<head>

<title>Actualizar Cotización</title>
<link rel="stylesheet"
href="{pageContext.request.contextPath}/css/style.css">
<link href="../css/datepicker.css" rel="stylesheet" type="text/css" />
<script type="text/javascript"
src="{pageContext.request.contextPath}/js/livevalidation.js">
</script>

<script type="text/javascript"
src="{pageContext.request.contextPath}/js/datepicker.js">
</script>

</head>
<body>
<table border="0" cellspacing="0" cellpadding="0" width="100%">
<tr>
<td align="center" colspan="3" height="20">

</td>
</tr>
<tr>
<td align="center" valign="top" width="150" class="menu">

</td>
<td width="25">
</td>
<td valign="top" align="center" width="625">
<div class="section">
Actualizar Cotización
</div>
<hr>
<s:form action="cotizaciones_actualizar" method="post">
<table align="center" width="80%" border="1">

<tr>
<td class="form" align="center" colspan="2">
<div class="label">Datos</div>

```

```

</td>
</tr>

<!--<tr>
<td class="form" align="center"><div class="label">Id
Cotizacion</div></td>
<td><s:textfield name="cotizacionSave.id" size="25" cssClass="fontForma"
/>
</td>
</tr>

--><tr>
<td class="form" align="center"><div class="label">Nombre</div></td>
<td><s:textfield name="cotizacionSave.nombre" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Fecha de
Cotización</div></td>
<td>
<s:textfield name="cotizacionSave.fecha.time" cssClass="w16em dateformat-
d-sl-m-sl-Y"
size="15" theme="simple" /></td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Forma de
Pago</div></td>
<td>
<s:textfield name="cotizacionSave.formaPago" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Producto(s)</div></td>
<td>
<s:textfield name="cotizacionSave.producto" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Cantidad</div></td>
<td>
<s:textfield name="cotizacionSave.cantidad" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>

```



```

<td class="form" align="center"><div class="label">Proveedores</div></td>
<td>
<s:textfield name="cotizacionSave.proveedores" size="25"
cssClass="fontForma" />
</td>
</tr>

</table>
<table>

<tr>
<td align="center">
<s:submit value="Actualizar Cotización"/>
</td>

<a
href="{pageContext.request.contextPath}/pt2/cotizaciones_listar.action">
</a>

<td align="center">
<input type="button" value=" Cancelar "
onClick="window.location='{pageContext.request.contextPath}/pt2/cotizaci
ones_listar.action'">

</td>
</tr>

</table>
</s:form>
<br>
</td>
</tr>
<tr>
<td align="center" colspan="3">

</td>
</tr>
</table>
</body>
</html>

```

Forma_insertar.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" %>
<html>
<head>

<title>Realizar Cotización</title>
<link rel="stylesheet"
href="{pageContext.request.contextPath}/css/style.css">
<link href="../css/datepicker.css" rel="stylesheet" type="text/css" />
<script type="text/javascript"
src="{pageContext.request.contextPath}/js/livevalidation.js">
</script>

<script type="text/javascript"
src="{pageContext.request.contextPath}/js/datepicker.js">
</script>

</head>
<body>
<table border="0" cellspacing="0" cellpadding="0" width="100%">
<tr>
<td align="center" colspan="3" height="20">

</td>
</tr>
<tr>
<td align="center" valign="top" width="150" class="menu">

</td>
<td width="25">
</td>
<td valign="top" align="center" width="625">
<div class="section">
Realizar Cotización
</div>
<hr>
<s:form action="cotizaciones_insertar" method="post">
<table align="center" width="80%" border="1">

<tr>
<td class="form" align="center" colspan="2">
<div class="label">Datos</div>
</td>
</tr>

<!--<tr>
<td class="form" align="center"><div class="label">Id
Cotizacion</div></td>
<td><s:textfield name="cotizacionSave.id" size="25" cssClass="fontForma"
/>
</td>
</tr>
```

```

--><tr>
<td class="form" align="center"><div class="label">Nombre</div></td>
<td><s:textfield name="cotizacionSave.nombre" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Fecha de
Cotización</div></td>
<td>
<s:textfield name="cotizacionSave.fecha.time" cssClass="w16em dateformat-
d-sl-m-sl-Y"
size="15" theme="simple" /></td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Forma de
Pago</div></td>
<td>
<s:textfield name="cotizacionSave.formaPago" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Producto(s)</div></td>
<td>
<s:textfield name="cotizacionSave.producto" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Cantidad</div></td>
<td>
<s:textfield name="cotizacionSave.cantidad" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Proveedores</div></td>
<td>
<s:textfield name="cotizacionSave.proveedores" size="25"
cssClass="fontForma" />
</td>
</tr>

</table>
<table>

<tr>
<td align="center">

```

```
<s:submit value="Realizar Cotización"/>
</td>

<a
href="{pageContext.request.contextPath}/pt2/cotizaciones_listar.action">
</a>

<td align="center">
<input type="button" value=" Cancelar "
onClick="window.location='{pageContext.request.contextPath}/pt2/cotizaci
ones_listar.action'">

</td>
</tr>

</table>
</s:form>
<br>
</td>
</tr>
<tr>
<td align="center" colspan="3">

</td>
</tr>
</table>
</body>
</html>
```

- Modulo Administrador de Proveedores

→ Action

ProveedoresAction.java

```
/**
 *
 */
package com.proyectoterminal.pt2.action;

import com.proyectoterminal.pt2.data.ProveedorDTO;
import com.proyectoterminal.pt2.logica.AdministradorProveedores;
import com.proyectoterminal.pt2.logica.local.FiltroProveedores;

import com.opensymphony.xwork2.ActionSupport;
import com.opensymphony.xwork2.Preparable;

import org.apache.struts2.interceptor.SessionAware;

import java.util.*;

/**
 * @author steph
 */
public class ProveedoresAction extends ActionSupport implements
SessionAware, Preparable
{

    private ProveedorDTO _proveedor;
    private ProveedorDTO _proveedorSave;
    private List<ProveedorDTO> _proveedores = new ArrayList<ProveedorDTO>();
    private long _id;

    private FiltroProveedores _filtro;

    private Map _sessionData;
    private AdministradorProveedores _administradorProveedores;

    public String listar()
    {
        _sessionData .put( "idProveedor" , null ) ;
        FiltroProveedores filter = new FiltroProveedores();
        filter.setNombre( "" );
        _proveedores = _administradorProveedores.leerProveedores( filter );

        return "listar";
    }
}
```

```

public String ver()
{
    if(_id == 0)
        _id = ((Long)_sessionData.get( "idProveedor")).longValue();
    _proveedor = _administradorProveedores.leerProveedores( _id );
    _sessionData .put( "idProveedor", _id );

    return "ver";
}

public String ver2()
{
    _id = ((Long)_sessionData.get( "idProveedor" )).longValue();
    _proveedor = _administradorProveedores.leerProveedores( _id );
    _sessionData .put( "idProveedor" , _id ) ;
    return "ver";
}

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

public String insertar()
{
    //inserta a la base de datos
    System.out.println("insertando en la base");
    _administradorProveedores.insertarProveedor( _proveedorSave );
    addActionMessage( "El proveedor ha sido insertado" );

    return "listar_redirect";
}

public String forma_actualizar()
{
    // _proveedor.setId( ((Long)_sessionData.get( "idProveedor"
    )).longValue() );
    _proveedor = _administradorProveedores.leerProveedores(
    _proveedor.getId() );
    _proveedorSave = cargarProveedor( _proveedor );

    return "forma_actualizar";
}

public String actualizar()
{
    //actualiza a la base de datos
    System.out.println("actualizando en la base");
    _administradorProveedores.actualizarProveedor( _proveedorSave );
    _sessionData .put( "idProveedor" , _proveedorSave .getId() ) ;
    return "ver_redirect";
}

public String borrar()

```

```

{
    System.out.println("borrando en la base");
    _administradorProveedores.borrarProveedor( _id );

    return "listar_redirect";
}

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

public ProveedorDTO getProveedorSave()
{
    return _proveedorSave;
}
public void setProveedorSave(ProveedorDTO proveedorSave )
{
    _proveedorSave = proveedorSave;
}

public FiltroProveedores getFiltro()
{
    return _filtro;
}
public void setFiltro( FiltroProveedores filtro )
{
    _filtro = filtro;
}

public ProveedorDTO getProveedor()
{
    return _proveedor;
}
public void setProveedor( ProveedorDTO proveedor )
{
    _proveedor = proveedor;
}
public List<ProveedorDTO> getProveedores()
{
    return _proveedores;
}
public void setProveedores( List<ProveedorDTO> proveedores )
{
    _proveedores = proveedores;
}
public long getId()
{
    return _id;
}
public void setId( long id )
{
    _id = id;
}
}

```

```

public void setSession( Map sessionData )
{
    _sessionData = sessionData;
}

public void prepare() throws Exception
{
    _filtro = ( FiltroProveedores ) _sessionData.get( "filtroProveedores"
);
    if( _filtro == null )
    {
        _filtro = new FiltroProveedores();
        _sessionData .put( "filtroProveedores" , _filtro ) ;
    }
}

public void setAdministradorProveedores (AdministradorProveedores
administradorProveedores)
{
    _administradorProveedores = administradorProveedores;
}

private ProveedorDTO cargarProveedor( ProveedorDTO proveedor )
{
    ProveedorDTO p = new ProveedorDTO();
    p.setComentarios(proveedor.getComentarios());
    p.setDireccion(proveedor.getDireccion());
    p.setEmail( proveedor.getEmail() );
    p.setEstadoProveedor( proveedor.getEstadoProveedor() );
    p.setId( proveedor.getId() );
    p.setNombre( proveedor.getNombre() );
    p.setRazonsocial( proveedor.getRazonsocial() );
    p.setRfc( proveedor.getRfc() );
    p.setTelefono( proveedor.getTelefono() );
    p.setTiempoentrega( proveedor.getTiempoentrega() );

    return p;
}
}

```


→ Data

ProveedorDTO.java

```
/**
 *
 */
package com.proyectoterminal.pt2.data;

import java.io.Serializable;
import org.hibernate.annotations.Proxy;

import javax.persistence.*;

import static javax.persistence.GenerationType.IDENTITY;

/**
 * @author steph
 */
@Entity @Table( name="proveedores" )
@Proxy( lazy=true )
public class ProveedorDTO implements Serializable
{
    private long _id;
    private String _razonsocial;
    private String _nombre;
    private String _rfc;
    private String _direccion;
    private int _telefono;
    private String _email;
    private String _comentarios;
    private int _tiempoentrega;
    private String _estadoProveedor;

    //////////////////////////////////////METODOS ACCESOS////////////////////////////////////

    @Id @Column( name="id_proveedor" ) @GeneratedValue( strategy=IDENTITY )
    public long getId()
    {
        return _id;
    }
    @Column( name="razonSocial" )
    public String getRazonsocial()
    {
        return _razonsocial;
    }
    @Column( name="nombre" )
    public String getNombre()
    {
        return _nombre;
    }
    @Column( name="rfc" )
```

```

public String getRfc()
{
    return _rfc;
}
@Column( name="direccion" )
public String getDireccion()
{
    return _direccion;
}
@Column( name="telefono" )
public int getTelefono()
{
    return _telefono;
}
@Column( name="email" )
public String getEmail()
{
    return _email;
}
@Column( name="comentarios" )
public String getComentarios()
{
    return _comentarios;
}
@Column( name="tiempoentrega" )
public int getTiempoentrega()
{
    return _tiempoentrega;
}
@Column( name="estadoProveedor" )
public String getEstadoProveedor()
{
    return _estadoProveedor;
}

////////////////////SETTERS////////////////////////////////////
public void setId( long id )
{
    _id = id;
}
public void setRazonsocial( String razonsocial )
{
    _razonsocial = razonsocial;
}
public void setNombre( String nombre )
{
    _nombre = nombre;
}
public void setRfc( String rfc )
{
    _rfc = rfc;
}
public void setDireccion( String direccion )
{
    _direccion = direccion;
}
public void setTelefono( int telefono )

```

```

    {
        _telefono = telefono;
    }
    public void setEmail( String email )
    {
        _email = email;
    }
    public void setComentarios( String comentarios )
    {
        _comentarios = comentarios;
    }
    public void setTiempoentrega( int tiempoentrega )
    {
        _tiempoentrega = tiempoentrega;
    }
    public void setEstadoProveedor( String estadoProveedor )
    {
        _estadoProveedor = estadoProveedor;
    }
}

```

→ Interceptores

ProveedoresInterceptor.java

```

/**
 *
 */
package com.proyectoterminal.pt2.interceptors;

import com.proyectoterminal.pt2.data.UsuarioDTO;

import com.opensymphony.xwork2.ActionInvocation;
import com.opensymphony.xwork2.interceptor.Interceptor;

import java.util.Map;

/**
 * @author steph
 */
public class ProveedoresInterceptor implements Interceptor
{
    @Override
    public void destroy()
    {
        System.out.println("Destruyendo ProveedoresInterceptor...");
    }

    @Override
    public void init()

```

```

    {
        System.out.println("Inicializando ProveedoresInterceptor...");
    }

    @Override
    public String intercept( ActionInvocation invocation ) throws Exception
    {
        Map<?, ?> session = invocation.getInvocationContext().getSession();
        UsuarioDTO usuario = ( UsuarioDTO )session.get( "usuario" );
        if( usuario != null &&
            usuario.getPerfil() == 1 )
            return invocation.invoke();
        else return "error";
    }
}

```

→ Logica

AdministradorProveedores.java

```

/**
 *
 */
package com.proyectoterminal.pt2.logica;
import com.proyectoterminal.pt2.data.ProveedorDTO;
import com.proyectoterminal.pt2.logica.local.*;

import org.springframework.transaction.annotation.Propagation;
import org.springframework.transaction.annotation.Transactional;

import java.util.List;

/**
 * @author steph
 */
@Transactional( rollbackFor=Exception.class )
public interface AdministradorProveedores
{
    @Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
    public ProveedorDTO leerProveedores(long id);

    @Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
    public List<ProveedorDTO> leerProveedoresQuery();

    @Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
    public long contarProveedores();

    @Transactional (propagation = Propagation.SUPPORTS, readOnly= true)

```

```

    public List <ProveedorDTO> leerProveedores(FiltroProveedores filter);

    @Transactional( propagation=Propagation.SUPPORTS, readOnly = true )
    public List<ProveedorDTO> buscarTemasCriterioNombre( FiltroProveedores
filter );

    public long insertarProveedor(ProveedorDTO proveedor);

    public void actualizarProveedor(ProveedorDTO proveedor);

    public void borrarProveedor(long id);
}

```

→ Logica.local

AdministradorProveedoresImpl.java

```

/**
 *
 */
package com.proyectoterminal.pt2.logica.local;

import com.proyectoterminal.pt2.action.ProveedoresAction;
import com.proyectoterminal.pt2.data.ProveedorDTO;
import com.proyectoterminal.pt2.logica.AdministradorProveedores;

import org.hibernate.*;
import org.hibernate.criterion.Order;
import org.hibernate.criterion.Property;

import java.util.List;

/**
 * @author steph
 */
public class AdministradorProveedoresImpl implements
AdministradorProveedores
{
    //////////////variables de parametros////////////////
    private int _estado;
    private String _salida;
    private List<Integer> _indices;
    private ProveedorDTO _proveedor;
    private AdministradorProveedores administradorProveedores;

    ////////////// PROPIEDADES //////////////////
    public void setEstado( int estado )
    {
        _estado = estado;
        System.out.println(_estado);
    }
}

```

```

public void setSalida( String salida )
{
    _salida = salida;
    System.out.println(_salida);
}

public void setIndices( List<Integer> indices )
{
    _indices = indices;
    System.out.println(_indices.get( 0 ).intValue());
    System.out.println(_indices.get( 1 ).intValue());
}

public void setProveedor( ProveedorDTO proveedor )
{
    this._proveedor = proveedor;

    System.out.println(_proveedor.getId());
    System.out.println(_proveedor.getNombre());
}

//////////variables de configuracion//////////
private SessionFactory _sessionFactory;

public ProveedorDTO leerProveedores(long id)
{
    Session session = getSession();
    return ( ProveedorDTO )session.get( ProveedorDTO.class, id );
}

public long insertarProveedor( ProveedorDTO proveedor )
{
    Session session = getSession();
    session.save( proveedor );
    return proveedor.getId();
}

public void setSessionFactory( SessionFactory sessionFactory )
{
    _sessionFactory = sessionFactory;
}
private Session getSession()
{
    return _sessionFactory.getCurrentSession();
}

```

```

public void actualizarProveedor( ProveedorDTO proveedor )
{
    Session session = getSession();
    session.update( proveedor );
}

public void borrarProveedor( long id )
{
    Session session = getSession();
    ProveedorDTO proveedor =
(ProveedorDTO)session.load(ProveedorDTO.class,id);
    session.delete(proveedor);
}

public List<ProveedorDTO> leerProveedoresQuery()
{
    Session session = getSession();
    Query query = session.createQuery( "SELECT p FROM ProveedorDTO p" );
    return query.list();
}

@Override
public long contarProveedores()
{
    Session session = getSession();
    Query query = session.getNamedQuery( "contar_proveedores" );
    long count = ( Long )query.uniqueResult();
    return count;
}

@Override
public List<ProveedorDTO> leerProveedores( FiltroProveedores filter )
{
    Session session = getSession();;
    Query query = session.getNamedQuery( "leer_proveedores_filtro" );
    query.setString( "nombre", "%" + filter.getNombre() + "%" );
    return query.list();
}

@Override
public List<ProveedorDTO> buscarTemasCriterioNombre( FiltroProveedores
filter )
{
    Criteria criteria = getSession().createCriteria( ProveedorDTO.class
);
    String nombre = "%" + filter.getNombre() + "%";
    criteria.add( Property.forName( "nombre" ).like( nombre ) );
    criteria.addOrder( Order.asc( "id" ) );
    return criteria.list();
}

```

```
}  
  
}
```

FiltroProveedores.java

```
/**  
 *  
 */  
package com.proyectoterminal.pt2.logica.local;  
  
/**  
 * @author steph  
 */  
public class FiltroProveedores  
{  
  
    private String _nombre;  
    public String getNombre()  
    {  
        return _nombre;  
    }  
    public void setNombre( String nombre )  
    {  
        _nombre = nombre;  
    }  
  
}
```

→ JSP

forma_actualizar.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" %>  
<html>  
<head>  
<title>Actualizar Proveedores</title>  
<link rel="stylesheet"  
href="{pageContext.request.contextPath}/css/estilo.css">  
<link href="../css/datepicker.css" rel="stylesheet" type="text/css" />  
<script type="text/javascript"  
src="{pageContext.request.contextPath}/js/livevalidation.js">  
</script>  
  
<script type="text/javascript"  
src="{pageContext.request.contextPath}/js/datepicker.js">  
</script>
```



```

</head>
<body>
<s:form action="proveedores_actualizar" method="post">

<table border="0" cellspacing="0" cellpadding="0" width="100%">

<tr>
<td align="center" valign="top" width="150" class="menu">

</td>
<td width="25">
</td>
<td valign="top" align="center" width="625">
<div class="section">
Insertar Proveedor
</div>
<table align="center" width="80%" border="1">
<tr>
<td class="form" align="center" colspan="2">
<div class="label">Datos</div>
</td>
</tr>
<tr>
<td class="form" align="center">
<div class="label">Nombre</div></td>
<td>
<s:hidden name="proveedorSave.id" />
<s:textfield name="proveedorSave.nombre" size="25" cssClass="fontForma"
/></td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Razón
Social</div></td>
<td><s:textfield name="proveedorSave.razonsocial" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">RFC</div></td>
<td><s:textfield name="proveedorSave.rfc" size="25" cssClass="fontForma"
/>
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Direccion</div></td>
<td><s:textfield name="proveedorSave.direccion" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Telefono</div></td>
<td><s:textfield name="proveedorSave.telefono" size="25"
cssClass="fontForma" />

```

```

</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">E-mail</div></td>
<td><s:textfield name="proveedorSave.email" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center">
<div class="label">Comentarios</div>
</td>
<td><s:textarea name="proveedorSave.comentarios" rows="4" cols="50"
/></td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Tiempo Entrega (en
dias)</div></td>
<td><s:textfield name="proveedorSave.tiempoentrega" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Estado
Proveedor (Disponible o No Disponible)</div></td>
<td><s:textfield name="proveedorSave.estadoProveedor" size="25"
cssClass="fontForma" />
</td>
</tr>

</table>
<table>

<tr>
<td align="center">
<s:submit value="Actualizar Proveedor"/>
</td>
<td align="center">
<a
href="{pageContext.request.contextPath}/pt2/proveedores_listar.action">

</a>
</td>
</tr>

</table>

<br>
</td>
</tr>

</table>

```

```

</s:form>
</body>
</html>

```

forma_insertar.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" %>
<html>
<head>
<title>Realizar el Alta de Proveedores</title>
<link rel="stylesheet"
href="{pageContext.request.contextPath}/css/estilo.css">
<link href="../css/datepicker.css" rel="stylesheet" type="text/css" />
<script type="text/javascript" src="$
{pageContext.request.contextPath}/js/datepicker.js"></script>

</head>
<body>
<table border="0" cellspacing="0" cellpadding="0" width="100%">

<tr>
<td align="center" valign="top" width="150" class="menu">

</td>
<td width="25">
</td>
<td valign="top" align="center" width="625">
<div class="section">
Dar de Alta un Proveedor
</div>
<hr>
<s:form action="proveedores_insertar" method="post">
<table align="center" width="80%" border="1">
<tr>
<td class="form" align="center" colspan="2">
<div class="label">Datos</div>
</td>
</tr>
<tr>
<td class="form" align="center"><div class="label">Nombre</div></td>
<td><s:textfield name="proveedorSave.nombre" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Razón
Social</div></td>
<td><s:textfield name="proveedorSave.razonsocial" size="25"
cssClass="fontForma" />
</td>
</tr>

```

```

<tr>
<td class="form" align="center"><div class="label">RFC</div></td>
<td><s:textfield name="proveedorSave.rfc" size="25" cssClass="fontForma"
/>
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Direccion</div></td>
<td><s:textfield name="proveedorSave.direccion" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Telefono</div></td>
<td><s:textfield name="proveedorSave.telefono" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">E-mail</div></td>
<td><s:textfield name="proveedorSave.email" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center">
<div class="label">Comentarios</div>
</td>
<td><s:textarea name="proveedorSave.comentarios" rows="4" cols="50"
/></td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Tiempo Entrega (en
días)</div></td>
<td><s:textfield name="proveedorSave.tiempoentrega" size="25"
cssClass="fontForma" />
</td>
</tr>

<tr>
<td class="form" align="center"><div class="label">Estado
Proveedor (Disponible o No Disponible)</div></td>
<td><s:textfield name="proveedorSave.estadoProveedor" size="25"
cssClass="fontForma" />
</td>
</tr>

</table>
<table>

<tr>
<td align="center">

```

```

<s:submit value="Insertar Proveedor"/>
</td>

<td align="center">
<a
href="{pageContext.request.contextPath}/pt2/proveedores_listar.action">

</a>
</td>

</tr>

</table>
</s:form>
<br>
</td>
</tr>

</table>
</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" %>
<html>
<head>
<title>ADMINISTRADOR PROVEEDORES</title>
<link rel="stylesheet"
href="{pageContext.request.contextPath}/css/estilo.css">
</head>
<body>
<table border="0" cellspacing="0" cellpadding="0" width="100%">

<tr>
<td align="center" valign="top" width="150" class="menu">

</td>
<td width="25">
</td>
<td valign="top" align="center" width="625">
<div class="section">
Administrador Proveedores
</div>
<hr>
<s:form action="proveedores_listar" method="post">
<div class="instructions">
Buscar Proveedores
</div>
<br>
<table border="1" cellpadding="10">
<tr>

```

```

<td align="center">
<table>
<tr class="form">
<td align="right">
<div class="label">
Patrón:
</div>
</td>
<td>
<input name="pattern" size="10" value="E*">
</td>
<td>
<s:submit value="Buscar"/>
</td>
</tr>
</table>
</td>
</tr>
</table>
<br>
</s:form>
<table align="center" width="80%" border="1">
<tr>
<td colspan="11" class="form" align="center">
<div class="label">Proveedores</div>
</td>
</tr>
<tr>
<td class="form" align="center"><div class="label">id
proveedor</div></td>
<td class="form" align="center"><div class="label">razon social
</div></td>
<td class="form" align="center"><div class="label">nombre </div></td>
<td class="form" align="center"><div class="label">rfc</div></td>
<td class="form" align="center"><div class="label">direccion</div></td>
<td class="form" align="center"><div class="label">Telefono</div></td>
<td class="form" align="center"><div class="label">email</div></td>
<td class="form" align="center"><div class="label">comentarios</div></td>
<td class="form" align="center"><div class="label">tiempo
entrega</div></td>
<td class="form" align="center"><div class="label">estado
proveedor</div></td>
<td class="form" align="center"></td>
</tr>

<c:forEach var="proveedor" items="{ proveedores }">
<tr>
<td>
<a href="{pageContext.request.contextPath}/proveedores_ver.action?id={
proveedor.id }">${ proveedor.id }</a>

</td>
<td>${ proveedor.razonsocial }</td>
<td>${ proveedor.nombre}</td>
<td>${ proveedor.rfc }</td>
<td>${ proveedor.direccion }</td>
<td>${ proveedor.telefono }</td>

```

```

<td>${ proveedor.email }</td>
<td>${ proveedor.comentarios }</td>
<td>${ proveedor.tiempoentrega }</td>
<td>${ proveedor.estadoProveedor}</td>
<td align="center">
<input type="button" value=" Borrar Proveedor"
onClick="window.location='${pageContext.request.contextPath}/pt2/proveedo
res_borrar.action?id=${ proveedor.id }'">

</td>
</tr>
</c:forEach>

</table>
<center>
<table>
<tr>
<td align="center">
<input type="button" value=" Insertar Proveedor "
onClick="window.location='${pageContext.request.contextPath}/pt2/proveedo
res_forma_insertar.action'">

</td>
<td align="center">



<a href=" ${
pageContext.request.contextPath}/pt2/login_salir.action">
salir</a>

</td>
</tr>
</table>
</center>
<br>
</td>
</tr>

</table>
</body>
</html>

```

ver.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" %>

<html>
  <head>
    <title>Ver Proveedor</title>
    <link rel="stylesheet" href="{pageContext.request.contextPath
} /css/estilo.css">
  </head>
  <body>
    <table border="0" cellspacing="0" cellpadding="0" width="100%">

      <tr>
        <td align="center" valign="top" width="150" class="menu">

          </td>
          <td width="25">
          </td>
          <td valign="top" align="center" width="625">

            <div class="section">
              Ver Proveedor
            </div>

            <hr>
            <s:form action="proveedores_forma_actualizar" method="post">

              <table align="center" width="80%" border="1">

                <tr>
                  <td class="form" align="center" colspan="2">
                    <div class="label">Datos</div>
                  </td>
                </tr>

                <tr>
                  <td class="form" align="center"><div
class="label">id</div></td>
                  <td>${ proveedor.id }</td>
                </tr>

                <tr>
                  <td class="form" align="center"><div class="label">razon
social</div></td>
                  <td><s:hidden name="proveedor.id" />${
proveedor.razonsocial }</td>
                </tr>

                <tr>
                  <td class="form" align="center"><div
class="label">nombre</div></td>
                  <td>${ proveedor.nombre}</td>
                </tr>
              </table>
            </s:form>
          </td>
        </tr>
      </table>
    </body>
  </html>
```



```

        </tr>
        <tr>
            <td class="form" align="center"><div
class="label">rfc</div></td>
            <td>${ proveedor.rfc}</td>
        </tr>
        <tr>
            <td class="form" align="center"><div
class="label">direccion</div></td>
            <td>${ proveedor.direccion }</td>
        </tr>

        <tr>
            <td class="form" align="center"><div
class="label">Telefono</div></td>
            <td>${ proveedor.telefono }</td>
        </tr>

        <tr>
            <td class="form" align="center"><div
class="label">email</div></td>
            <td>${ proveedor.email }</td>
        </tr>
        <tr>
            <td class="form" align="center"><div
class="label">tiempo entrega</div></td>
            <td>${ proveedor.tiempoentrega }</td>
        </tr>
        <tr>
            <td class="form" align="center"><div
class="label">estado Proveedor</div></td>
            <td>${ proveedor.estadoProveedor}</td>
        </tr>
        <tr>
            </td>
        </tr>
    </table>
</table>

    <tr>
        <td align="center">
            <s:submit value="Actualizar Proveedor"/>
        </td>
        <td align="center">
            <a
href="${pageContext.request.contextPath}/pt2/proveedores_listar.action?id
Proveedor=${ proveedor.id }">
                
            </a>
        </td>
    </tr>

</table>

</s:form>

```

```

        <br>
    </td>
</tr>

<tr>

</tr>
</table>
</body>
</html>

```

Test.java

```

/**
 *
 */
package com.proyectoterminal.pt2.test;

import com.proyectoterminal.pt2.data.*;

import com.proyectoterminal.pt2.logica.*;
import com.proyectoterminal.pt2.logica.local.FiltroProveedores;
import com.proyectoterminal.pt2.logica.local.FiltroSolicitudes;

import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

import java.util.*;

/**
 * @author steph
 */
public class Test
{
    /**
     * @param args
     */
    public static void main( String[] args )
    {
        try {
            new Test().execute();
        }
        catch ( Exception e )
        {
            e.printStackTrace() ;
        }
    }

    private void execute() throws Exception
    {
        System.out.println("inicia prueba");
    }
}

```

```

        //probarProveedor();

        System.out.println("Prueba Usuarios");
        //probarUsuario();
        System.out.println("prueba terminada");

        System.out.println("inicia Solicitud");
        //probarSolicitud();

    }

    private void probarSolicitud()
    {
        AdministradorSolicitudes administrador =
getAdministradorSolicitudes();

        FiltroSolicitudes filtro = new FiltroSolicitudes();
        filtro.setNombre( "hola" );
        List<SolicitudDTO> lista = administrador.leerSolicitudesQuery();
        System.out.println("# de Solicitudes por Filtro: " +
lista.size());
        for( SolicitudDTO solicitud : lista )
        {
            System.out.println(solicitud.getNombre());
        }
    }

    private void probarUsuario()
    {
        AdministradorUsuarios administrador = getAdministradorUsuarios();
        Map<Long,String> mapa = administrador.cargarUsuarios();
        System.out.println("Mapa Usuarios: no de registros " +
mapa.size());

    }

    private void probarProveedor()
    {
        AdministradorProveedores administrador =
getAdministradorProveedores();

        // ProveedorDTO proveedor = new ProveedorDTO();
        /* proveedor.setId( 1 );
        proveedor.setRazonsocial( "CO:CA" );
        proveedor.setNombre( "coca-cola" );
        proveedor.setRfc( "C.A" );
        proveedor.setDireccion( "santa fe" );
        proveedor.setTelefono( 55723891);
        proveedor.setEmail( "coca@kof.mx" );
        proveedor.setComentarios( "ninguno" );
        proveedor.setTiempoentrega( 2); */
        //administrador.insertarProveedor( proveedor );
        /* ProveedorDTO proveedor = administrador.leerProveedores( 2 );
        System.out.println("Proveedor Nombre: " + proveedor.getNombre());

```

```

        System.out.println("Proveedor Razon Social: " +
proveedor.getRazonsocial());
        System.out.println("Proveedor Rfc: " + proveedor.getRfc());*/
    /*
        System.out.println("Proveedor Direccion: " +
proveedor.getDireccion());
        System.out.println("Proveedor Telefono: " +
proveedor.getTelefono());
        System.out.println("Proveedor EMail: " + proveedor.getEmail());
        System.out.println("Proveedor Comentarios: " +
proveedor.getComentarios());
        System.out.println("Proveedor Tiempo Entrega: " +
proveedor.getTiempoentrega());
    */
    /*
        proveedor.setNombre( "oficce" );
        proveedor.setRazonsocial( "OF." );
        proveedor.setRfc( "OF12CCI" );
        administrador.actualizarProveedor( proveedor );
        administrador.leerProveedores( 2 );
        System.out.println("Proveedor Nombre: " + proveedor.getNombre());
        System.out.println("Proveedor Razon Social: " +
proveedor.getRazonsocial());
        System.out.println("Proveedor Rfc: " + proveedor.getRfc());*/
    /*
        administrador.borrarProveedor( 2 );*/

    /*
        List<ProveedorDTO> lista = administrador.leerProveedoresQuery();
        System.out.println("# de proveedores: " + lista.size());*/
    /*
        long cuenta = administrador.contarProveedores();
        System.out.println("# de temas: " + cuenta);
    */
    FiltroProveedores filtro = new FiltroProveedores();
    filtro.setNombre( "coca-cola" );
    List<ProveedorDTO> lista = administrador.leerProveedores( filtro
);
    System.out.println("# de Proveedores por Filtro: " +
lista.size());
    for( ProveedorDTO proveedor : lista )
    {
        System.out.println(proveedor.getNombre());
    }

    /*
        FiltroProveedores filtro = new FiltroProveedores();
        filtro.setNombre( "coca-cola" );
        List<ProveedorDTO> temas =
administrador.buscarTemasCriterioNombre( filtro );
        Iterator<ProveedorDTO> i = temas.iterator();
        ProveedorDTO t = new ProveedorDTO();
        while(i.hasNext()) {
            t = i.next();
            System.out.println("proveedor ID: " + t.getId() + " proveedor
nombre: " +
            t.getNombre());
        }
    */
}

```

```

    private AdministradorProveedores getAdministradorProveedores()
    {
        ApplicationContext context = new ClassPathXmlApplicationContext
        ( "spring-dev.xml" );
        return( AdministradorProveedores)context.getBean(
"administradorProveedores" );
    }

    private AdministradorUsuarios getAdministradorUsuarios()
    {
        ApplicationContext context = new ClassPathXmlApplicationContext
        ( "spring-dev.xml" );
        return ( AdministradorUsuarios)context.getBean(
"administradorUsuarios" );
    }

    private AdministradorSolicitudes getAdministradorSolicitudes()
    {
        ApplicationContext context = new ClassPathXmlApplicationContext
        ( "spring-dev.xml" );
        return ( AdministradorSolicitudes)context.getBean(
"administradorSolicitudes" );
    }
}

```

- Archivos XML

Base.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>
    <package name="base" extends="struts-default" abstract="true">

        <interceptors>

            <interceptor name="proveedores_interceptor"
class="com.proyectoterminal.pt2.interceptors.ProveedoresInterceptor" />

            <interceptor name="solicitudes_interceptor"
class="com.proyectoterminal.pt2.interceptors.SolicitudesInterceptor" />

            <interceptor name="cotizaciones_interceptor"
class="com.proyectoterminal.pt2.interceptors.CotizacionesInterceptor" />

```

```

<interceptor name="map_to_request"
  class="com.proyectoterminal.interceptors.MapToRequestInterceptor"
  />

<interceptor-stack name="customStack">
  <interceptor-ref name="exception"/>
  <interceptor-ref name="map_to_request"/>
  <interceptor-ref name="servletConfig"/>
  <interceptor-ref name="prepare"/>
  <interceptor-ref name="i18n"/>
  <interceptor-ref name="debugging"/>
  <interceptor-ref name="profiling"/>
  <interceptor-ref name="checkbox"/>
  <interceptor-ref name="params">
    <param name="excludeParams">dojo\..*</param>
  </interceptor-ref>
  <interceptor-ref name="conversionError"/>
  <interceptor-ref name="validation">
    <param name="excludeMethods">input,back, cancel,browse</param>
  </interceptor-ref>
  <interceptor-ref name="workflow">
    <param name="excludeMethods">input,back, cancel,browse</param>
  </interceptor-ref>
</interceptor-stack>

  <interceptor-stack name="proveedorStack">
<interceptor-ref name="proveedores_interceptor" />
<interceptor-ref name="customStack" />
</interceptor-stack>

  <interceptor-stack name="solicitudStack">
<interceptor-ref name="solicitudes_interceptor" />
<interceptor-ref name="customStack" />
</interceptor-stack>

  <interceptor-stack name="cotizacionStack">
<interceptor-ref name="cotizaciones_interceptor" />
<interceptor-ref name="customStack" />
</interceptor-stack>

</interceptors>

  <default-interceptor-ref name="customStack"/>

</package>
</struts>

```

Queries.hbm.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-mapping PUBLIC
"-//Hibernate/Hibernate Mapping DTD//EN"
"http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">
<hibernate-mapping>

<query name="contar_proveedores">
SELECT COUNT(*) FROM ProveedorDTO p
</query>

<query name="leer_proveedores_filtro">
FROM ProveedorDTO p WHERE (p.nombre LIKE :nombre )
</query>

<query name="contar_solicitudes">
SELECT COUNT(*) FROM SolicitudDTO s
</query>

<query name="leer_solicitudes_filtro">
FROM SolicitudDTO s WHERE (s.nombre LIKE :nombre)
</query>

<query name="contar_cotizaciones">
SELECT COUNT(*) FROM CotizacionDTO a
</query>

<query name="leer_cotizaciones_filtro">
FROM CotizacionDTO a WHERE (a.nombre LIKE :nombre)
</query>

<query name="leer_usuarios">
FROM UsuarioDTO c
</query>

<query name="leer_usuario_login_pass">
FROM UsuarioDTO u WHERE
(
(u.login = ? )AND
(u.password = ?)
)
</query>

</hibernate-mapping>
```

Spring-dev.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" default-lazy-init="true">

<!-- declaro del datasource -->
<bean id="dataSource"
      class="org.apache.commons.dbcp.BasicDataSource"
      destroy-method="close">
  <property name="driverClassName"
    value="com.mysql.jdbc.Driver"/>

  <property name="url">

<value>jdbc:mysql://localhost:3306/proyecto?autoReconnect=true</value>
  </property>
  <property name="username" value = "root"></property>
  <property name="password" value = "root"></property>
</bean>

<bean id="transactionManager"
class="org.springframework.orm.hibernate3.HibernateTransactionManager"/>

<bean id="sessionFactory"
class="org.springframework.orm.hibernate3.annotation.AnnotationSessionFac
toryBean">
<property name="annotatedClasses">
<list>
<value>com.proyectoterminal.pt2.data.ProveedorDTO</value>
<value>com.proyectoterminal.pt2.data.UsuarioDTO</value>
<value>com.proyectoterminal.pt2.data.SolicitudDTO</value>
<value>com.proyectoterminal.pt2.data.CotizacionDTO</value>

</list>
</property>
<property name="hibernateProperties">
<props>
<prop key="hibernate.dialect">
org.hibernate.dialect.MySQLDialect
</prop>
</props>
</property>
```



```

<property name="mappingResources">
<list><value>queries.hbm.xml</value></list>
</property>
</bean>
<tx:annotation-driven/>
<import resource="spring-services.xml"/>

</beans>

```

Spring-services.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" default-lazy-init="true">

<bean id="administradorProveedores"
class="com.proyectoterminal.pt2.logica.local.AdministradorProveedoresImpl
">
<property name="salida" value="/tmp/reporte.pdf"/>
<property name="estado" value="100"/>
<property name="indices">
<list>
<value>100</value>
<value>200</value>
</list>
</property>

<property name="proveedor">
<bean class="com.proyectoterminal.pt2.data.ProveedorDTO">
<property name="id" value="1"/>
<property name="nombre" value="Proveedor"/>
</bean>
</property>
</bean>

<bean id="administradorUsuarios"
class="com.proyectoterminal.pt2.logica.local.AdministradorUsuariosImpl"/>

<bean id="administradorSolicitudes"
class="com.proyectoterminal.pt2.logica.local.AdministradorSolicitudesImpl
">
<property name="salida" value="/tmp/reporte.pdf"/>
<property name="estado" value="100"/>
<property name="indices">
<list>

```

```
<value>100</value>
<value>200</value>
</list>
</property>

<property name="solicitud">
<bean class="com.proyectoterminal.pt2.data.SolicitudDTO">
<property name="id" value="1"/>
<property name="nombre" value="Solicitud"/>
</bean>
</property>
</bean>

<bean id="administradorCotizaciones"
class="com.proyectoterminal.pt2.logica.local.AdministradorCotizacionesImp
1">
<property name="salida" value="/tmp/reporte.pdf"/>
<property name="estado" value="100"/>
<property name="indices">
<list>
<value>100</value>
<value>200</value>
</list>
</property>

<property name="cotizacion">
<bean class="com.proyectoterminal.pt2.data.CotizacionDTO">
<property name="id" value="1"/>
<property name="nombre" value="Cotizacion"/>
</bean>
</property>
</bean>

</beans>
```

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" />
<constant name="struts.ui.theme" value="simple"/>

<!--<include file="struts-extensions.xml"/>
-->

<include file="base.xml"/>

<package name="pt2" extends="base">
<action name="principal" class=
"com.proyectoterminal.pt2.action.PrincipalAction">
<result name="success"/>principal.jsp</result>
</action>

<action name="proveedores_*"
class="com.proyectoterminal.pt2.action.ProveedoresAction" method="{1}">
<interceptor-ref name="proveedorStack"/>
<result name="menu"/>WEB-INF/login/menu.jsp</result>
<result name="listar"/>WEB-INF/proveedores/listar.jsp</result>
<result name="ver"/>WEB-INF/proveedores/ver.jsp</result>
<result name="forma_actualizar"/>WEB-
INF/proveedores/forma_actualizar.jsp</result>
<result name="forma_insertar"/>WEB-
INF/proveedores/forma_insertar.jsp</result>
<result name="listar_redirect"
type="redirectAction">proveedores_listar</result>
<result name="ver_redirect"
type="redirectAction">proveedores_ver</result>

<result name="ver_redirect"
type="redirectAction">proveedores_ver?idProveedores=$
{ idProveedor }</result>

<result name="error" type="redirectAction">
<param name="actionName">login_forma_login</param>
</result>
</action>

<action name="solicitudes_*"
class="com.proyectoterminal.pt2.action.SolicitudesAction" method="{1}">
<!--<interceptor-ref name="solicitudStack"/>
--><result name="menu"/>WEB-INF/login/menu.jsp</result>
<result name="listar"/>WEB-INF/solicitud/listar.jsp</result>
```

```

<result name="listarautorizacion"/WEB-
INF/autorizaciones/listar.jsp</result>
<result name="ver"/WEB-INF/solicitud/ver.jsp</result>
<result name="forma_actualizar"/WEB-
INF/solicitud/forma_actualizar.jsp</result>
<result name="forma_insertar"/WEB-
INF/solicitud/forma_insertar.jsp</result>
<result name="listar_redirect" type="redirectAction">solicitudes_listar
</result>
<result name="ver_redirect"
type="redirectAction">solicitudes_ver</result>

<result name="ver_redirect"
type="redirectAction">solicitudes_ver?idSolicitudes=${ idSolicitud
}</result>

```

```

<result name="error" type="redirectAction">
<param name="actionName">login_forma_login</param>
</result>
</action>

```

```

<action name="cotizaciones_*"
class="com.proyectoterminal.pt2.action.CotizacionesAction" method="{1}">
<!--<interceptor-ref name="cotizacionStack"/>
--><result name="menu"/WEB-INF/login/menu.jsp</result>
<result name="listar"/WEB-INF/cotizaciones/listar.jsp</result>
<result name="ver"/WEB-INF/cotizaciones/ver.jsp</result>
<result name="forma_actualizar"/WEB-
INF/cotizaciones/forma_actualizar.jsp</result>
<result name="forma_insertar"/WEB-
INF/cotizaciones/forma_insertar.jsp</result>
<result name="listar_redirect" type="redirectAction">cotizaciones_listar
</result>
<result name="ver_redirect"
type="redirectAction">cotizaciones_ver</result>

<result name="ver_redirect"
type="redirectAction">cotizaciones_ver?idCotizaciones=${
idCotizacion}</result>

```

```

<result name="error" type="redirectAction">
<param name="actionName">login_forma_login</param>
</result>
</action>

```

```

<action name="login_*"
class="com.proyectoterminal.pt2.action.LoginAction" method="{1}">
<result name="menu"/WEB-INF/login/menu.jsp</result>
<result name="forma_login"/WEB-INF/login/login.jsp</result>

```

```
<result name="error">/WEB-INF/login/login.jsp</result>
<result name="login_redirect" type="redirectAction">login_login</result>
<result name="menuRedirect" type="redirectAction">
<param name="actionName">login_menu</param>
</result>
<result name="error" type="redirectAction">
<param name="actionName">login_forma_login</param>
</result>
<result name="salir" type="redirectAction">
<param name="actionName">login_salir</param>
</result>

</action>

</package>
</struts>
```

Bibliografía

- [1] DDS Sistemas Informáticos, “FAQ ACERCA DEL SOFTWARE ERP”
Disponible: <http://www.ddsoftware.com.ar/software-gestion-comercial.html> Consultado el 23 de Junio del 2010
- [2] Quality Soft Argentina S.A., “Software de Gestion Comercial Isis Classic”
Disponible: <http://www.sistemaisis.com/sistemaisisclassic.htm> Consultado el 2 de Junio del 2010
- [3] <http://www.sistemaisis.com/cabecera-isis.jpg> Consultado el 15 de Junio del 2010
- [4] Husky Software, “Husky Gestión Comercial”
Disponible: <http://www.huskysoft.com/siles02.htm> Consultado el 2 de Junio del 2010
- [5] <http://www.actualsys.com.ar/dat/logo-husky-software.jpg> Consultado el 15 de Junio del 2010
- [6] Mastermagazine, “Definicion de ERP”
Disponible: <http://www.mastermagazine.info/termino/4908.php> Consultado el 7 de Junio del 2010
- [7] I. Sommerville, “*Ingenieria del Software*”, 7ª. Ed. Pearson Addison Wesley, 2005, p.275
- [8] A. Tanenbaum y A. Woodhull, “*Sistemas Operativos: Diseño e implementacion*”, Segunda Ed. Prentice Hall, 1998, p.2
- [9] Wikipedia, “Eclipse(Software)”
Disponible: [http://es.wikipedia.org/wiki/Eclipse_\(software\)](http://es.wikipedia.org/wiki/Eclipse_(software)) Consultado el 28 de Junio del 2010
- [10] Wikipedia, “tomcat”
Disponible: <http://es.wikipedia.org/wiki/Tomcat> Consultado el 28 de Junio del 2010
- [11] Wikipedia, “Framework”
Disponible: <http://es.wikipedia.org/wiki/Framework> Consultado el 28 de Junio del 2010
- [12] K. Kendall y J. Kendall, “*Análisis y Diseño de Sistemas*”, 6ª.Ed. Pearson Prentice Hall,2005

UNIVERSIDAD AUTÓNOMA METROPOLITANA UNIDAD AZCAPOTZALCO

DIVISIÓN DE CIENCIAS BÁSICAS E INGENIERÍA

LICENCIATURA EN INGENIERÍA EN COMPUTACIÓN

MANUAL DE USUARIO



Stephanie Varela Peña

Proyecto Terminal 2

1. Iniciar Sesión

Para iniciar sesión debes de ingresar , Tu Usuario y Contraseña que se te debio de haber proporcionado por el area de Sistemas. Al dar un clic en Accesar, se desplegara la siguiente ventana (Menu)



The screenshot shows a login window with a blue header and background. The header contains the text "SISTEMA DE GESTION PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS" and "PROYECTO TERMINAL 2". Below the header, the text "Bienvenido al Sistema para la Adquisición de Bienes y Servicios" is displayed. Underneath, the instruction "Proporciona tu cuenta de usuario y contraseña" is followed by two input fields: "Usuario:" and "Contraseña:". A button labeled "Accesar" is positioned below the password field. At the bottom of the window, a horizontal line is followed by the text "Alumna:Stephanie Varela Peña".

2. Menú

Aquí podrás Visualizar los diferentes Módulos y escoger el modulo dependiendo de la operación que desees realizar.

SISTEMA DE GESTION PARA LA ADOUSIACIÓN DE BIENES Y SERVICIOS

PROYECTO TERMINAL 2

MENU

[Solicitudes](#)
[Autorizaciones](#)
[Cotización](#)
[Proveedores](#)

Bienvenido al Sistema de Adquisición de Bienes y Servicios

MENU

El sistema contiene los siguientes Modulos:

- [Solicitudes](#)
- [Autorizaciones](#)
- [Cotización](#)
- [Proveedores](#)

Alumna: Stephanie Varela Peña

3. Solicitud

Se despliegan las solicitudes realizadas ,en esta parte se puede Realizar una nueva Solcitud o borrar una existente

SISTEMA DE GESTION PARA LA ADOQUISIÓN DE BIENES Y SERVICIOS

PROYECTO TERMINADO

MENU

- [Solicitudes](#)
- [Autorizaciones](#)
- [Cotización](#)
- [Proveedores](#)

Solicitudes

Solicitudes					
id Solicitud	producto	cantidad	fecha	nombre solicitante	adquisicion
15	hojas	60	08/09/2011	stephanie	1

Realizar Solicitud [salir](#)

Alumna:Stephanie Varela Peña

3.1 Realizar Solicitud

Una vez dando clic en Realizar Solicitud, se despliega el siguiente formato para capturar los datos de la nueva solicitud que se quiere ingresar , Si se elige la opción insertar solicitud quedara registrada en la BD , de lo contrario se elige el botón cancelar que regresa a la pantallas donde se visualizan todas las solicitudes



The screenshot displays a web application interface. At the top, a banner reads "SISTEMA DE GESTION PARA LA ADOUSIACIÓN DE BIENES Y SERVICIOS" with "PROYECTO TERMINAL 2" in the bottom right corner. On the left, a "MENU" section lists "Solicitudes", "Autorizaciones", "Cotización", and "Proveedores". The main area is titled "Realizar una Solicitud" and contains a form with the following fields:

Datos	
Producto	<input type="text"/>
Cantidad	<input type="text"/>
fecha de Solicitud	<input type="text"/> 
Nombre Solicitante	<input type="text"/>
Numero de Adquisición	<input type="text"/>

Below the form are two buttons: "Insertar Solicitud" and "Cancelar".

3.2 Actualizar una Solicitud

Das un clic en el número que aparece dentro de la columna Id Solicitud

SISTEMA DE GESTION PARA LA ADOUSIACIÓN DE BIENES Y SERVICIOS
PROYECTO TERMINAL 2

MENU
[Solicitudes](#)
[Autorizaciones](#)
[Cotización](#)
[Proveedores](#)

Solicitudes

id Solicitud	producto	cantidad	fecha	nombre solicitante	adquisicion	
15	hojas	60	08/09/2011	stephanie	1	Borrar Solicitud

Realizar Solicitud [salir](#)

Alumna:Stephanie Varela Peña

Se mostrara la siguiente pantalla , para actualizar datos debes de dar clic en botón Actualizar Solicitud, si no deseas realizar un actualización da clic en salir

**SISTEMA DE GESTION PARA LA ADQUISICIÓN
DE BIENES Y SERVICIOS**

PROYECTO TECNIBAI 2

Ver Solicitud

Datos	
Numero de Pedido	5
Producto	1
Cantidad	20
Fecha de Solicitud (La fecha no se puede modificar)	17/08/2011
Nombre solicitante	1
Numero de Adquisicion	3

Actualizar Solicitud

Salir

Alumna:Stephanie Varela Peña

Realizas las modificaciones que así desees. Una vez terminado das clic en el botón de Actualizar Solicitud , si no realizaras ningún cambio , da un clic en el boton Cancelar

SISTEMA DE GESTION PARA LA ADOUSIACIÓN DE BIENES Y SERVICIOS

PROYECTO TERMINAL 2

MENU

- [Solicitudes](#)
- [Autorizaciones](#)
- [Cotización](#)
- [Proveedores](#)

Actualizar Solicitud

Datos	
Numero de Pedido	15
Producto	hojas
Cantidad	60
Nombre Solicitante	stephanie
Numero de Adquisición	1

Actualizar Solicitud Cancelar

Alumna:Stephanie Varela Peña

3.3 Borrar Solicitud

Para borrar una solicitud te posiciones en la solicitud que quieras eliminar y das un clic en Borrar Solicitud

SISTEMA DE GESTION PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS
PROYECTO TERMINAL 2

MENU
[Solicitudes](#)
[Autorizaciones](#)
[Cotización](#)
[Proveedores](#)

Solicitudes

Solicitudes						
id Solicitud	producto	cantidad	fecha	nombre solicitante	adquisicion	
15	hojas	60	08/09/2011	stephanie	1	Borrar Solicitud

Realizar Solicitud [salir](#)

Alumna:Stephanie Varela Peña

4. Autorizaciones

Una vez accedando al modulo de Autorizaciones

SISTEMA DE GESTION PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS
PROYECTO TERMINAL 2

MENU
[Solicitudes](#)
[Autorizaciones](#)
[Cotización](#)
[Proveedores](#)

Autorizaciones

id Solicitud	producto	cantidad	fecha	nombre solicitante	adquisicion	
15	hojas	60	08/09/2011	stephanie	1	<input type="button" value="Autorizar Solicitud"/> <input type="button" value="No Autorizar Solicitud"/>

[salir](#)

Alumna:Stephanie Varela Peña

Para Autorizar una solicitud ,das clic en el botón de Autorizar Solicitud , esta ya no se visualizara dentro de esta pantalla y pasaras al módulo de Cotización

Para No Autorizar Solicitud , das clic en el botón y esta solicitud será eliminada de la BD

5. Modulo de Cotización

Para acceder al modulo de cotizacion debes de darle un clic en el menu



The screenshot shows a web application interface with a light blue background. At the top, there is a banner with the text "SISTEMA DE GESTION PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS" in bold blue letters. Below the banner, the text "PROYECTO TERMINAL 2" is visible in the bottom right corner. On the left side, there is a vertical menu with the heading "MENU" and four items: "Solicitudes", "Autorizaciones", "Cotización", and "Proveedores", all in purple text. The main content area on the right has the heading "Bienvenido al Sistema de Adquisición de Bienes y Servicios" in blue. Below this, there is another "MENU" heading in blue, followed by the text "El sistema contiene los siguientes Modulos:" in black. A bulleted list follows, containing "Solicitudes", "Autorizaciones", "Cotización", and "Proveedores" in purple. At the bottom of the page, the text "Alumna: Stephanie Varela Peña" is displayed in black.

El modulo de Cotización se visualiza de la siguiente forma, donde podemos

Realizar un cotización, borrar una cotización, enviar una cotización y visualizar la cotización

SISTEMA DE GESTION PARA LA ADOQUISIÓN DE BIENES Y SERVICIOS
PROYECTO TERMINAL 2

MENU
[Solicitudes](#)
[Autorizaciones](#)
[Cotización](#)
[Proveedores](#)

Cotizaciones

Cotizaciones						
id Cotizacion	Nombre	Fecha Cotización	Forma de Pago	Producto (s)	Cantidad	Proveedores
3	gerardo	14/09/2011	efectivo	hojas	5	OFFICE MAX

Realizar Cotización salir Borrar Cotización Enviar Cotización

Alumna:Stephanie Varela Peña

5.1 Realizar Cotización

Introducir los datos para realizar una cotización, una vez finalizado debes dar clic en botón Realizar Cotización para que quede registrada en la BD, si no deseas realizar una cotización da clic en el botón cancelar



The screenshot shows a web application interface. At the top, a banner reads "SISTEMA DE GESTION PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS" and "PROYECTO TERMINAL 2". On the left, a "MENU" section contains links for "Solicitudes", "Autorizaciones", "Cotización", and "Proveedores". The main area is titled "Realizar Cotización" and contains a form with the following fields:

Datos	
Nombre	<input type="text"/>
Fecha de Cotización	<input type="text"/> 
Forma de Pago	<input type="text"/>
Producto(s)	<input type="text"/>
Cantidad	<input type="text"/>
Proveedores	<input type="text"/>

At the bottom of the form are two buttons: "Realizar Cotización" and "Cancelar".

5.2 Visualizar Cotizacion

Si deseas visualizar la cotización en un formato diferente da un clic en el numero que aparece dentro de la Columna Id Cotización y se mostrara la siguiente pantalla



SISTEMA DE GESTION PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS

PROYECTO TERMINAL 2

MENU

- [Solicitudes](#)
- [Autorizaciones](#)
- [Cotización](#)
- [Proveedores](#)

Ver Cotización

Datos	
id Cotización	3
Nombre	gerardo
Fecha de Cotización (La fecha no se puede modificar)	14/09/2011
Forma de Pago	efectivo
Producto(s)	hojas
Cantidad	5
Proveedores	OFFICE MAX

Alumna:Stephanie Varela Peña

5.3 Borrar Cotización , Enviar Cotización

Para borrar la Cotización, posicónate sobre la cotización que quieras eliminar y da un clic en el botón de Borrar Cotizacion .

Para envíar la cotización debes de dar clic en el botón enviar cotización el cual se encuentra a un lado de la opción de Borrar Cotizacion .

SISTEMA DE GESTION PARA LA ADOUSICIÓN DE BIENES Y SERVICIOS
PROYECTO TERMINAL 2

MENU
[Solicitudes](#)
[Autorizaciones](#)
[Cotización](#)
[Proveedores](#)

Cotizaciones

id Cotizacion	Nombre	Fecha Cotización	Forma de Pago	Producto (s)	Cantidad	Proveedores	
3	gerardo	14/09/2011	efectivo	hojas	5	OFFICE MAX	<input type="button" value="Borrar Cotización"/> <input type="button" value="Enviar Cotización"/>

[salir](#)

Alumna:Stephanie Varela Peña

6. Modulo de Proveedores

El modulo de Proveedores se visualiza de la siguiente forma, donde podemos

Dar de alta un proveedor , modificar los datos de un proveedor y borrar el proveedor

SISTEMA DE GESTION PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS
PROTEGO TEJEMAR S

Administrador Proveedores

Proveedores										
id proveedor	razon social	nombre	rfc	direccion	Telefono	email	comentarios	tiempo entrega	estado proveedor	
1	Femsa coca cola	COCA-COLA FEMSA	CO:Ca	Av. sta fe 160	55678900	coca@kof.com.mx	ninguno	0	Disponible	<input type="button" value="Borrar Proveedor"/>
2	MAX	office MAX	OFFMAX	naucalpan #72	67789345	office@kof.com.mx	ningunpo	5	Disponible	<input type="button" value="Borrar Proveedor"/>
3	informatic	papeleria informatic	PAINFO	tlanepantla #35	223335566	ninguno	solo surte los dias lunes	1	No Disponible	<input type="button" value="Borrar Proveedor"/>

[salir](#)

Alumna:Stephanie Varela Peña

6.1 Dar de alta un Proveedor

Damos un clic en el botón de Insertar Proveedor, y en la siguiente pantalla se capturan los datos del proveedor que queremos dar de alta en la BD

SISTEMA DE GESTIÓN PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS
PROYECTO TERCERA 3

Dar de Alta un Proveedor

Datos	
Nombre	<input type="text"/>
Razón Social	<input type="text"/>
RFC	<input type="text"/>
Dirección	<input type="text"/>
Teléfono	<input type="text"/>
E-mail	<input type="text"/>
Comentarios	<input type="text"/>
Tiempo Entrega (en días)	<input type="text"/>
Estado Proveedor (Disponible o No Disponible)	<input type="text"/>

Alumna: Stephanie Varela Peña

6.2 Actualizar Proveedor

Para actualizar los datos de un proveedor, das un clic en el numero que aparece dentro de la columna id Proveedor

SISTEMA DE GESTION PARA LA ADOUSIACIÓN DE BIENES Y SERVICIOS
PROTEGO, TELMIABI 2

Administrador Proveedores

Proveedores										
id proveedor	razon social	nombre	rfc	direccion	Telefono	email	comentarios	tiempo entrega	estado proveedor	
<u>1</u>	Femsa coca cola	COCA-COLA FEMSA	CO:Ca	Av. sta fe 160	55678900	coca@kof.com.mx	ninguno	0	Disponible	Borrar Proveedor
<u>2</u>	MAX	office MAX	OFFMAX	naucalpan #72	67789345	office@kof.com.mx	ningunpo	5	Disponible	Borrar Proveedor
<u>3</u>	informatic	papeleria informatic	PAINFO	tlanepantla #35	223335566	ninguno	solo surte los dias lunes	1	No Disponible	Borrar Proveedor

Insertar Proveedor salir

Alumna:Stephanie Varela Peña

En la siguiente pantalla seleccionas la opción Actualizar Proveedor, sino das un clic en cancelar

SISTEMA DE GESTION PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS
PROYECTO TERRIBIL 9

Ver Proveedor

Datos	
id	1
razon social	Femsa coca cola
nombre	COCA-COLA FEMSA
rfc	CO:Ca
direccion	Av. sta fe 160
Telefono	55678900
email	coca@kof.com.mx
tiempo entrega	0
estado Proveedor	Disponible

Alumna: Stephanie Varela Peña

En la siguiente pantalla modificas los datos que necesites del proveedor y das clic en Actualizar Proveedor para que queden actualizados, sino necesitas actualizar , das clic en cancelar

SISTEMA DE GESTION PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS
PROYECTO TERMINAL 2

Actualizar Proveedor

Detos	
Nombre	COCA-COLA FEMSA
Razón Social	Femsa coca cola
RFC	CO-CA
Direccion	Av. sta fe 100
Telefono	55870900
E-mail	coca@kof.com.mx
Comentarios	n.ninguno
Tiempo Entrego(en dias)	0
Estado Proveedor (Disponible o No Disponible)	Disponible

Alumna: Stephanie Varela Peña

6.3 Borrar Proveedor

Para borrar el proveedor, posíciónate sobre el proveedor que quieras eliminar y da un clic en el botón de Borrar Proveedor para que sea eliminado.

SISTEMA DE GESTIÓN PARA LA ADQUISICIÓN DE BIENES Y SERVICIOS
PROYECTO TECNIBI 2

Administrador Proveedores

Proveedores										
id proveedor	razon social	nombre	rfc	direccion	Telefono	email	comentarios	tiempo entrega	estado proveedor	
<u>1</u>	Fensa coca cola	COCA-COLA FEMSA	CO:Ca	Av. sta fe 160	55678900	coca@kof.com.mx	ninguno	0	Disponible	Borrar Proveedor
<u>2</u>	MAX	office MAX	OFFMAX	naucalpan #72	67789345	office@kof.com.mx	ningunpo	5	Disponible	Borrar Proveedor
<u>3</u>	informatic	papeleria informatic	PAINFO	tlanepantla #35	223335566	ninguno	solo surte los dias lunes	1	No Disponible	Borrar Proveedor

Insertar Proveedor salir

Alumna:Stephanie Varela Peña