



Estrategia

Nro Grupo: 37

Grupo: TPGDD

2016

Integrantes

Nombre	Legajo
Martínez Rodríguez, Jorge	140.831-8
Uviña, María Guadalupe	143.470-6
Chungara, Daniel	136.644-0

Índice

<u>Decisiones de Diseño</u>	3
<u>Migración de Datos</u>	4
<u>DER</u>	6
<u>Explicación procedures, function y triggers</u>	6
<u>Explicación de la Aplicación Desktop</u>	9

Decisiones de Diseño.

En la aplicación se utilizaron patrones de diseño como al Abstract Factory. Se busco un diseño de ventanas que fuera ameno para el usuario con el fin de facilitar un manejo intuitivo, y eficiente.

Sobre el DER se modelaron las entidades de acuerdo a las restricciones de los requerimientos, teniendo en presentes las buenas prácticas.

Sobre el script, se decidió hacer una análisis previo de los datos a migrar, pruebas sobre conjuntos determinados para comprobar supuestos y posteriormente realizar la migración con optimización. También se definieron varios índices con este objeto.

Migración de Datos

Analizando los datos de la tabla maestra se observa que los mismos cumplen el siguiente patrón. El cual se tomo como supuesto para hacer la migración. Se toma una abreviatura para para el representante de los conjuntos de campos que empiezan con el mismo nombre.

1. Todas las tuplas son distintas de NULL para los campos de Publicacion_.
2. Para todas las tuplas que los campos Publ_Cli_ son NOT NULL se cumple entonces que los campos Publ_Empresa son NULL y viceversa.
3. Para todas las tuplas el comportamiento de los demás campos es el siguiente:

Caso	Cli_	Compra_	Oferta_	Calificacion_	Item_	Factura_	Forma_
A	NULL	NULL	NULL	NULL	NULL	NULL	NULL
B	NOT NULL	NOT NULL	NULL	NULL	NULL	NULL	NULL
C	NOT NULL	NULL	NOT NULL	NULL	NULL	NULL	NULL
D	NOT NULL	NOT NULL	NULL	NOT NULL	NULL	NULL	NULL
E	NULL	NULL	NULL	NULL	NOT NULL	NOT NULL	NOT NULL

Casos:

- A. Se hace una publicación, estado activa.
- B. Se realiza compra de una publicación
- C. Se realiza oferta de una subasta.
- D. Se califica la compra
- E. Se Factura al vendedor (se incluyen en una sola factura todos los items).

Se realizo el DER con los requerimientos del nuevo sistema y se normalizo. Lo que produjo la creación de las siguientes entidades, que se mencionan y se explica los supuestos a la hora de realizar la migración desde la tabla maestra.

Sobre la migración de las Relaciones

- a. Funcionalidades: Se agregan las funcionalidades del sistema.
- b. Roles: Se agregan los roles especificados en el sistema.
- c. FormasPago: se agregan las especificadas en el sistema.
- d. Localidades: se decidió dejarla en blanco para la migración.
- e. Estados: se agregan las especificadas en el sistema.
- f. Usuarios: para generar el username se tomo los caracteres que están antes de la @ del mail de cada usuario que es empresa o cliente. Y se les dio el mismo password a todos, el cual podrán cambiar cuando entren al sistema. Para el administradorGeneral se le creó la tupla con el fin de no tener NULL. Como fecha de creación la actual. La reputación del usuario como vendedor se realiza posteriormente a terminar la migración de otras tablas y sobre las relaciones del nuevo sistema, que son un menor número de tuplas en general. Por cuestiones de optimización
- g. Empresas y Clientes: se migraron las tuplas específicas para estas entidades, poniendo los campos comunes en el Usuario, como se puede ver en el DER.
- h. Visibilidades: se migran de la tabla maestra. La prioridad definida es de mayor a menor 1 a 5, Platino a Gratis. Se tomo como que no hubo costo de Envío.
- i. Publicaciones: se migran de la tabla maestra. Dado los análisis realizados. Se tomo como supuesto luego del análisis de los datos que todas las publicaciones están en estado finalizada, pues tienen facturas. Excepto 3 de ellas. Las facturas y monto del ítem que es cero corresponden a las publicaciones gratuitas.
- j. ComprasInmediatas: luego de hacer varios análisis se determino que se venden todas las cantidades de las compras inmediatas por lo que se tomo supuesto que el stockDisponible es 0 y unidades vendidas es el total del stock de la publicación de referencia, Entonces en aras de disminuir el tiempo de carga del script, se saca el uso de funciones y cursores y se deja en el anexo.
- k. Subastas: Se migran de la tabla maestra. Se toma como supuesto que comienzan con un valor base 0, que irá aumentando a medida que se

hacen ofertas. Se define el valorActual como el valor con que cierra la subasta o sea, el valor mayor de los Items, siendo los ítems de menor valor los de comisión.

- l. Ofertas: se migran de la tabla maestra.
- m. Compras: se migran de la tabla maestra. Se tomo como supuesto que es obligatorio calificar una sola vez la compra.
- n. Calificaciones: se migran de la tabla maestra. La tabla maestra tenía un rango de estrellas de 1 – 10, mientras que el nuevo sistema tiene un rango de 1 – 5, por lo que se uso una escala de reducción 2:1.
- o. Facturas: se migran de la tabla maestra.
- p. Items: se migran de la tabla maestra.

DER:

Se puede observar en una imagen adjunta con el nombre del equipo.

Explicación procedures, function y triggers

`trigger TPGDD.TRIGGERUPDATECALIFICADACOMPRA`

Se usa cuando ingresa una calificación y se debe actualizar el campo calificada de la relación Compras a 1, para decir con esto que está calificada.

`trigger TPGDD.updateReputacionVendedorTrigger`

Para actualizar la reputación del vendedor cuando es ingresada una calificación en Calificaciones.

`trigger TPGDD.updateStockPublicacionTrigger`

Para actualizar el stock de la publicación cuando una compra es ingresada en Compras, según la cantidad de unidades que se compro.

`trigger TPGDD.updateValorActualSubastaTrigger`

Para actualizar el valor de la Subasta, según la oferta válida que se ha ingresado en Ofertas.

`trigger TPGDD.deleteUsuariosRolesTrigger`

Para eliminar el rol de un usuario, cuando este es dado de baja.(se deshabilita para el usuario) . Cuando se actualiza la relación Roles.

`trigger TPGDD.generarFacturacionPorPublicar`

Cuando se actualiza la relación Publicaciones. Se genera la factura por publicar.

`create trigger TPGDD.generarFacturacionPorComprar`

Cuando se inserta una compra en Compras, se genera la factura correspondiente. Incluye la generación del Item que tendrá dicha factura.

`TRIGGER TPGDD.ClienteCalifPendienteCompTrigger`

`TRIGGER TPGDD.ClienteCalifPendienteOferTrigger`

Ambos son para evitar que un cliente compre o oferte cuando tiene pendiente calificar más de 3 compras. Se da en los ingresos de Compras y Ofertas respectivamente.

`TRIGGER TPGDD.bajaLogicaPausaPublicacionesTrigger`

En usuarios , cuando se le da de baja(lógica) a un usuario , se pausan todas las publicaciones que tiene activas.

`TRIGGER TPGDD.activaUsuarioActivaPublicacionesTrigger`

En usuarios cuando se reactiva a un usuario que estaba en baja(lógica) se activan las publicaciones de este usuario.

`TRIGGER TPGDD.generaCompdeSubastaTrigger`

En publicaciones se genera la compra cuando se hace cambia el estado a finalizada de una publicación que es una Subasta.

`PROCEDURE TPGDD.peoresVendedoresSP(@codigoVisbilidad numeric(18,0),
@numeroTrimestre int, @year int)`

Vendedores con mayor cantidad de productos no vendidos.

`PROCEDURE TPGDD.mejoresCompradoresSP(@idRubro int, @numeroTrimestre int, @year int)`

Recibe el numero del trimestre, el año y el rubro. Selecciona clientes con mayor cantidad de productos comprados, por mes y por año, dentro de un rubro particular.

`PROCEDURE TPGDD.mejoresVendedoresPorCantidadFacturasSP(@numeroTrimestre int,
@year int)`

Recibe el numero del trimestre y el año. Selecciona vendedores con mayor cantidad de facturas dentro de un mes y año particular

```
PROCEDURE TPGDD.mejoresVendedoresPorMontoFacturadoSP(@numeroTrimestre int, @year int)
```

Recibe el numero del trimestre y el año. Selecciona vendedores con mayor monto facturado dentro de un mes y año particular.

```
PROCEDURE tpgdd.migraReputacionUser
```

Para actualizar la reputación luego que ya se migraron las tuplas a las relaciones.

```
FUNCTION TPGDD.convierteEstrellas(@cantidadEstrellas int)
```

Para convertir por escala de reducción 2:1 la cantidad de estrellas. Recibe la cantidad de estrellas a convertir y devuelve las estrellas convertidas.

Se utilizan procedures para realizar los ABMs de Usuarios, Visibilidad, Rubro y Rol, que a su desencadenan otros procedures para otras entidades. Los cuales no se detallan por ser generales de todo ABM.

Procedimientos para listados estadísticos:

```
PROCEDURE TPGDD.peoresVendedoresSP
```

```
PROCEDURE TPGDD.mejoresCompradoresSP
```

```
PROCEDURE TPGDD.mejoresVendedoresPorCantidadFacturasSP
```

```
PROCEDURE TPGDD.mejoresVendedoresPorMontoFacturadoSP
```

Para mejorar la expresividad, claridad y mantenibilidad se decidió en todos los procedimientos almacenados utilizar funciones auxiliares. Como se puede apreciar en la siguiente captura:

```
--*****
-----
--2. Clientes con mayor cantidad de productos comprados, por mes y por año, dentro
--de un rubro particular
-----
create PROCEDURE TPGDD.mejoresCompradoresSP(@idRubro int, @numeroTrimestre int, @year int)
AS
BEGIN

    select top 5 C.idCliente, TPGDD.cantidadProductosComprados(C.idCliente, @idRubro, @numeroTrimestre, @year) as CantidadProductosComprados
    from TPGDD.Clientes C
    order by 2 desc

END
go
```

Procedimientos para ABMs:

Sobre ABMs de USUARIOS:

En todos los casos se intento encapsular la lógica repetida en procedimientos cómo:


```

procedure TPGDD.modificarUsuarioSP
procedure TPGDD.insertarUsuarioSP
procedure TPGDD.validarCamposObligatoriosUsuarioSP
procedure TPGDD.validarUnicidadCuitRazonSocialSP
procedure TPGDD.validarUnicidadTipoNumeroDocumentoSP
procedure TPGDD.validarUnicidadUsuarioSP

```

Son llamados internamente por :

```

procedure TPGDD.darAltaUsuarioCliente
procedure TPGDD.darAltaUsuarioEmpresa
procedure TPGDD.darBajaUsuario
procedure TPGDD.modificarUsuarioClienteSP
procedure TPGDD.modificarUsuarioEmpresaSP

```

Además se decidió que la mayoría de las validaciones se hicieran dentro de los procedimientos, para que dotar de claridad el código C# ya que éste sólo llamara a los procedimientos y si ocurre un error lo atrapa la excepción y lo notificara al usuario. De esta manera los formularios se transforman en los controladores

Se realizó el testing de los procedures y de los triggers, los cuales se encuentran en la carpeta anexo.

Explicación de la aplicación:

Se entrega un prototipo, que está en desarrollo.

Configuración: para establecer la hora del sistema se deberá ir al archivo App.config y establecer la hora en el formato dd/mm/aa.

Se puede entrar al sistema con **Usuario: “admin”, password: “ w23e”** como se muestra en la figura 1.

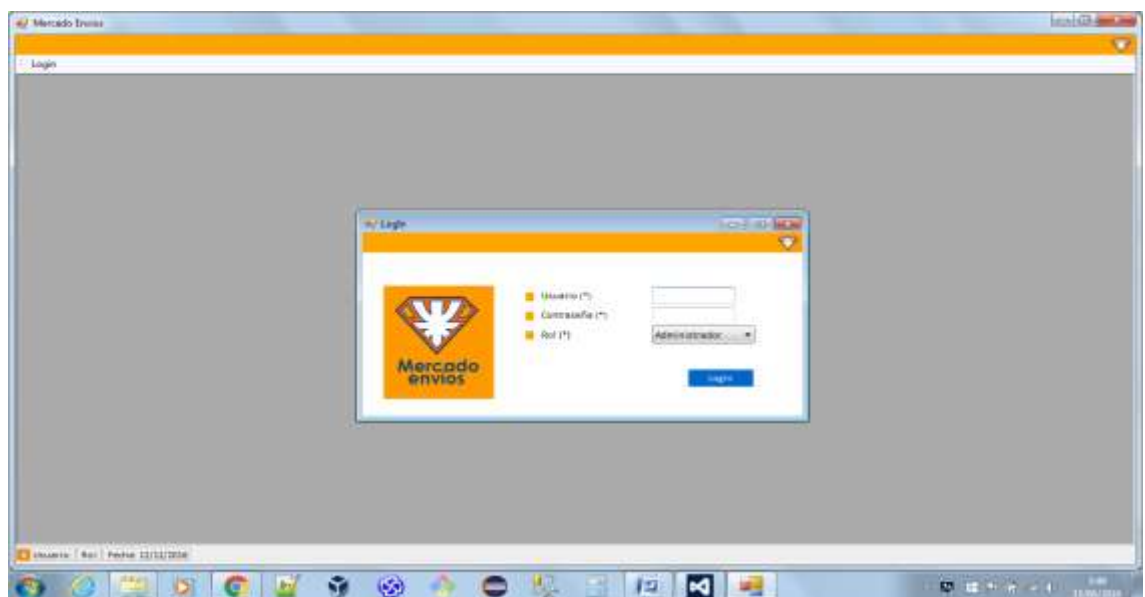


Figura 1.

. Posteriormente se tendrá acceso al formulario principal desde el que se tendrá acceso a los demás formularios, y se podrá cambiar la contraseña ver figura 2.

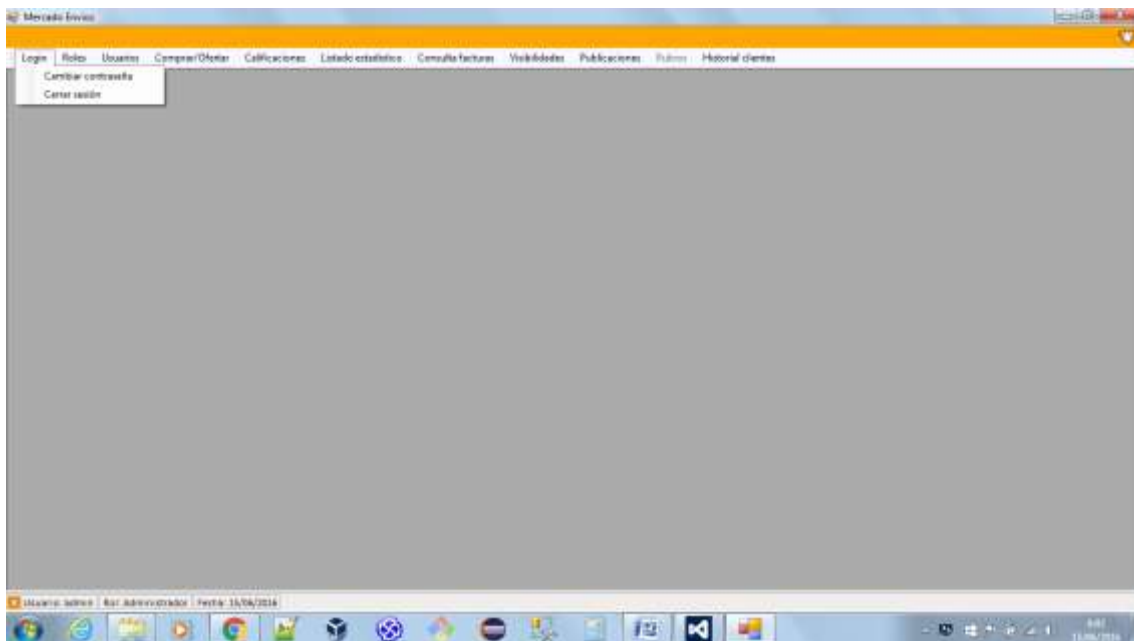


Figura 2.

Se observa la validación de los formularios, ver figura 3.

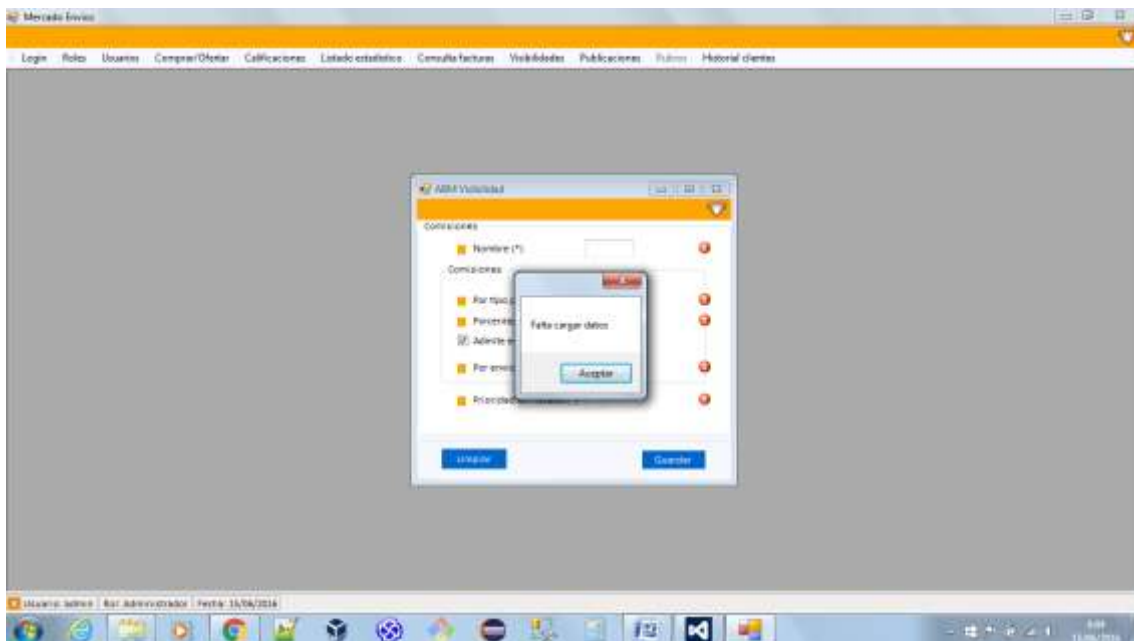


Figura 3.

Algunas de las pantallas que contiene la aplicación son las siguientes, ver figura 4, y figura 5.

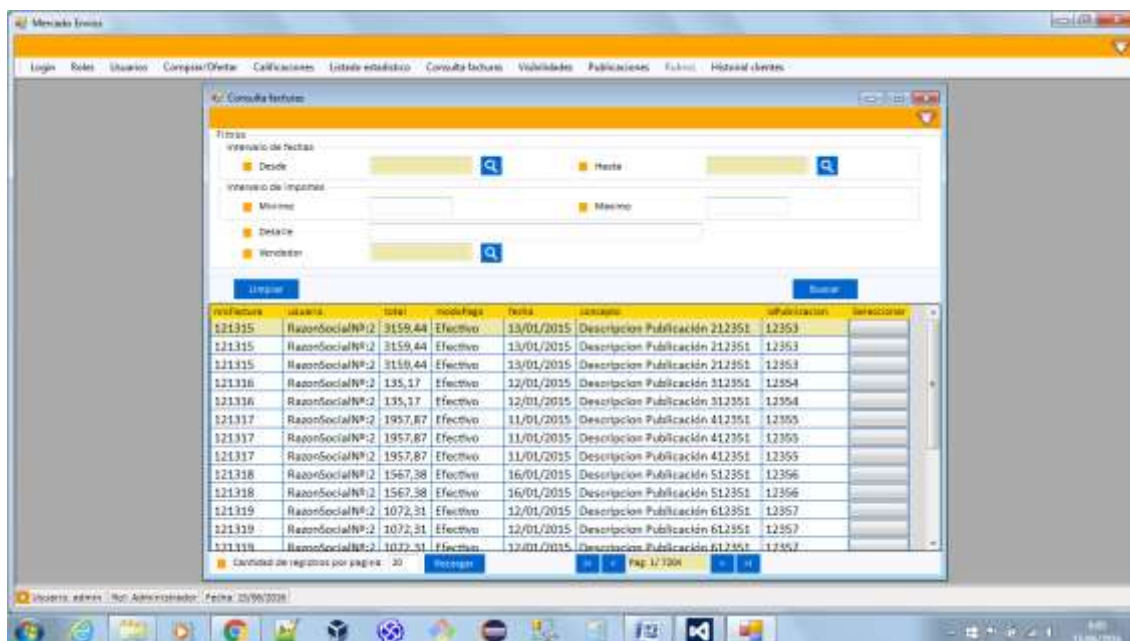


Figura 4.

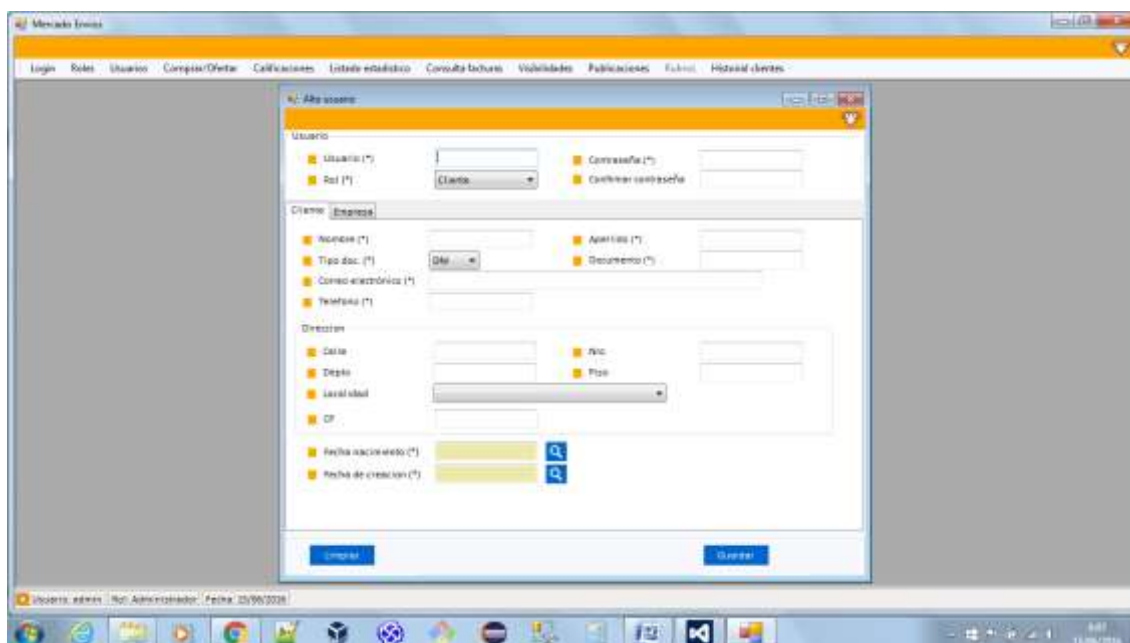


Figura 5.