TP Gestión De Datos 2016- Mercado Envió

Leonardo Lopez…………………….. 1475915

Alexis Taberna……………………….1496931

Brian Lobo…………………………...1496414

Email del responsable del grupo: [leoceit@gmail.com](mailto:leoceit@gmail.com)

Estrategia

Hipótesis inicial: consideremos que el tp se basa en demostrar los conocimientos aprendidos en la cursada, por ende la mayoría de la lógica propia del negocio se encuentra en el motor y no tanto en c#.

Aclaraciones:

El archivo de configuración se encuentra en (ruta\_del\_archivo.txt), la fecha de inicio del sistema la seteamos en (una\_fecha).

Cuando se inicia la aplicación se barre la base de datos para saber que publicación esta vencida, las que están vencidas las finalizamos y cerramos la factura con la fecha del archivo de configuración.

Con las subastas hacemos lo mismo y adjudicamos la compra al usuario ganador.

Tablas “Factura” y “Item\_factura”:

Los campos “total\_facturar” y “forma\_pago” serán nulos a lo largo de todo el ciclo del sistema, ya que el enunciado dice:

“Tener en cuenta que solo deberá generarse la facturación dado que el pago de facturas no compete a este trabajo práctico”.

Como dice en el enunciado:

“cuando una publicación pasa a estar activa se debe generar en ese mismo instante la facturación correspondiente al tipo de visibilidad elegida”. Hacemos exactamente eso en la Tabla “FACTURA” y a cada publicación le corresponde una única factura.

Después, cuando se realiza una venta , se crea un “item\_factura” por la compra que se hizo, y el mismo “item\_factura” pertenece a la factura propia de la publicación. Entonces en la tabla “ITEM\_FACTURA” facturamos el envió (si corresponde) ,la cantidad y el precio unitario.

La "factura\_fecha" toma la fecha del sistema (o fecha del .txt) cuando en una publicación el stock llega a cero, o cuando la publicación esta vencida, ósea, "factura\_fecha" representa la finalización de una publicación.

También agregamos la funcionalidad de envío haciendo que se pueda elegir habilitar la opción desde la creación de la publicación y cada vez que se genera una compra/oferta elegir si es o no con envío. La comisión por la misma la elegimos como un valor constante de $100.

A continuación solo se describen las decisiones que tomamos para las diferentes funcionalidades, consideramos que esta aplicación está implementando actualmente todo lo pedido en el enunciado.

**Login y Seguridad**

Se agregó la funcionalidad “CAMBIAR\_PASSWORD” debido que esta acción seria parte de ABM de usuario, pero esta solo puede ser ejecutada por el rol de administrador, a diferencia de “CAMBIAR\_PASSWORD” que es una funcionalidad para todos los roles. Osea gracias a esto, todos los roles pueden cambiar su contraseña.

**ABM Rol**

**ABM Cliente y Empresa**

El administrador es el único que tiene acceso a esta funcionalidad.

Empresa: En caso que se haga un ABM de una empresa con “razon\_social” o “cuit” repetidos, no se ejecutara dicha transacción y se le notificara al usuario del error.

Cliente: En caso que se haga un ABM de un cliente con “dni” repetido, no se ejecutara dicha transacción y se le notificara al usuario del error.

**ABM Rubro**

No se implementa, ya que no era requerido en el enunciado.

**ABM Visibilidad**

Para la baja de visibilidad decidimos tomar el consejo del ayudante del grupo de google, y emitir un mensaje de error, si no es posible borrar la visibilidad.

**Generar Publicación**

Se puede crear en estado “borrador” o “activa” dependiendo de la elección del usuario.

La fecha de inicio de publicación la lee del archivo de configuración.

La fecha de vencimiento es a elección del usuario.

Lo modelamos para que una publicación pueda tener un único rubro.

Para el caso que el cliente nuevo con una publicación gratis, no se le cobra costo de visibilidad, independientemente de la elección del cliente al elegir el tipo de visibilidad. Si el cliente por ejemplo, elige el tipo de “visibilidad gratis” se factura esa visibilidad, perdiendo así la el beneficio de publicación si costo para futuras publicaciones.

**Comprar/Ofertar**

El usuario hace los filtros requeridos en el enunciado, decidimos que el resultado de las búsquedas sea en páginas de 10.

Cuando el usuario selecciona una publicación para verla (ya sea compra o subasta), antes de comprar, hacemos 3 validaciones (delegamos dicha lógica en el motor SQL):

1) el cliente no sea el mismo.

2) la publicación no esté en estado pausado.

3) el usuario no tenga más de 3 publicaciones sin calificar.

Por cada una se le notifica al usuario en una pantalla de dicho error.

Después, al momento de comprar/ofertar también validamos en cada caso (delegamos esta lógica en C#):

1) compra: validamos que la cantidad elegida no supere el stock.

2) Oferta: validamos que supere el monto de subasta y que los valores sean enteros.

Por cada una se le notifica al usuario en una pantalla de dicho error.

**Historial Cliente**

Mostramos en pantalla la cantidad de compras sin calificar y un resumen de estrellas dadas.

También mostramos un historial con las compras y subastas ganadas, en paginas de 10.

**Calificar Vendedor**

Se muestra un listado con todas las compras no calificadas por el usuario, ahí mismo el usuario selecciona que con que puntuación calificarla y una descripción (opcional). En ese mismo instante, la compra deja de estar pendiente de calificar

También se muestra la venta con todo lo pedido en el enunciado:

1) Resumen se estrellas dadas y las ultimas 5 compras.

**Consulta de facturas realizadas al vendedor**

Lo implementamos de manera tal, que solo el administrador tenga acceso a esta funcionalidad. En el cual se puede filtrar por un monto, un rango de montos, una fecha, un rango de fechas o cualquier combinación entre estas.

Consideramos que el filtro de “esta dirigido” como el “id” del vendedor.

Quitamos el filtro de “factura detalle” ya que nuestro diseño del módulo de facturación no aplica ese campo por motivos explicados en las aclaraciones.

**Listado Estadístico**

El listado estadístico es un solo panel en el cual primero se ingresa el año de consulta, luego se ingresa el trimestre; Ingresados estos dos se puede seleccionar entre cuatro tipos de listado. Una vez seleccionado el tipo de listado se puede agregar un filtro (respectivo a cada tipo de listado) y además si se quiere consultar por un mes en específico se puede filtrar por un solo mes, siempre y cuando este en el trimestre especificado anteriormente.

# **Tomas de decisiones en la migración de datos**

En general las estructuras de los objetos usados para realizar la migración se basan en 3 partes:

* Creación de vistas con la información la cual se va a insertar
* Creación de procedimientos que ejecutan un insert dentro de la tabla para hacer el pasaje de los datos
* Borrado de todos los objetos usados para la migración correspondiente a esa tabla.

# **Tablas**

**Usuario**:

La estructura que usamos para almacenar los clientes se basa en la herencia, existe el tipo usuario el cual puede ser un cliente, empresa o administrador y contiene los datos en común de estos 3, como en la tabla Maestra (desde ahora abrevio diciendo maestra) no existe el concepto de usuario, se tenía que insertar un usuario por cada uno siendo el nombre de usuario y la contraseña, en caso del cliente el DNI y en caso de la empresa su Razón Social.

Para el caso del cliente se debería hacer un select con la unión de los diferentes casos que son:

Los que compraron algo de una empresa, los que compraron algo de otro cliente y los que vendieron algo, pero como todos arrojaban los mismos 28 registros se terminó usando el que sea más rápido.

Para el caso de las empresas solo se usó un select distinct de todas las publicaciones de las empresas, ya que la maestra no guarda más que publicaciones y datos referentes a ella.

**Visibilidad:**

Dado que la maestra siempre tenía estos 5 registros, estos se insertaron a mano.

**Estado de Publicación:**

Siguiendo con el enunciado se insertó a mano esos registros.

**Tipo de Publicación:**

Solo existen 2 tipos según pudimos identificar, por lo que también fueron insertados a mano.

**Rubros**:

El rubro que está en la maestra fue considerada como la descripción corta, y la descripción larga se puso en Null ya que no teníamos datos acerca de ella, supusimos que se llenaría en la aplicación.

Algo a tener en cuenta es que los datos que se repiten porque son un tipo o demás, las pasamos en mayúsculas para evitar errores en búsquedas o demás.

**Calificaciones**:

Las calificaciones las consideramos con estrellas, siendo un valor entero y entre 1 y 5, como la maestra tiene valores de 1 al 10, dividimos esa calificación por 2 redondeando para arriba.

Para la creación de la FK, en algunas tablas se agregó un campo con el código provisto por la maestra para establecer la relación con el inner join, que luego de que las FK sean creadas, son borradas.

**Factura**:

Simplemente se leen los datos y se insertan agregando en la tabla una columna con el código dado por la tabla maestra.

**Item** **Factura**:

Insertado normal.

**Publicaciones**:

Normal salvo que se realiza la migración de las publicaciones de clientes y de empresas, por lo que se tiene que hacer un case para comparar, ésta se hace por el nick del usuario, como es el DNI o la razón social y sé sabe que estos son únicos se puede hacer la conexión así para la FK.

Como no encontramos alguna publicación que no esté activa, se migraron todas en estado activo.

**Compras**:

Solo pueden comprar los usuarios, así que no se tiene que hacer un case dependiendo si uso el DNI o la RS, el resto es normal.

**Ofertas**:

Según observamos, todas las ofertas que tenían el máximo ofrecido se concretaba, por esa razón tomamos esta condición para verificar en cada oferta si la oferta propuesta por el cliente es el máximo de esa publicación, depende de cual sea el caso, se sabe si se concretó o no.

**Roles**:

Como son los 3 comentados en el enunciado los pusimos a mano y habilitados.

**Funcionalidad**:

Son las ventanas de nuestra aplicación y las que se pedían en el trabajo.

# **Eliminación de innecesarios**

Eliminamos las columnas y con ella todos sus datos de las tablas que, para hacer la FK con las demás, tuvimos que usar identificadores dadas por la maestra.

# **Creación del Usuario inicial**

Se crea el usuario pedido por el trabajo dándole todo el control.