**Estrategias TP GDD 2019 2° Cuatrimestre**

**Login y Seguridad:**

* Primera pantalla del sistema donde se debe completar usuario y contraseña. En caso de no completar alguno, aparece un mensaje indicando que debe hacerse.
* Una vez completados, se valida a través de un Stored Procedure y puede suceder que:
  + Se muestra un mensaje indicando que el usuario es inexistente o incorrecto. En caso de ser un error de password, se suma un intento fallido.
  + Se indica que el usuario no está habilitado (esto puede ser por intentos fallidos acumulados o por una inhabilitación manual).
  + El usuario puede ingresar: se muestran las opciones que tiene disponibles y se reinician los intentos fallidos.

**ABM Cliente:**

* La búsqueda se realiza con una consulta dinámica donde los campos vacíos se reemplazan por “1=1”.
* Para dar de alta un cliente se verifica por constraint que el DNI no se repita, si bien el mail puede considerarse como unique, encontramos en la tabla maestra que clientes distintos tienen mismo mail.
* Al darse de baja un cliente se actualiza el campo habilitado.

**ABM Proveedor:**

**Ventana Proveedor:**

* Para las búsquedas:
  + El CUIT es búsqueda exacta.
  + La razón social, nombre de contacto y email son de búsqueda no exacta.
  + El rubro es un desplegable que trae sus valores de la tabla correspondiente.
  + El checkbox “Mostrar deshabilitados” permite mostrar también a aquellos usuarios que no se encuentran habilitados.

**Bajas:**

* Las bajas se realizan con el botón “Habilitar/Deshabilitar”: luego de realizar la búsqueda y seleccionar a un proveedor, este botón lo habilitará/deshabilitará según se encuentre deshabilitado/habilitado respectivamente. Es una baja lógica, no física.
* El botón “Limpiar” borra todos los campos completados.

**Tanto en Alta como en Modificación:**

* Se realizan validaciones a nivel aplicación:
  + Todos los campos son obligatorios.
  + El CUIT debe tener el formato correcto: dos dígitos, un guion, ocho dígitos, un guion, un dígito.

- El teléfono y el código postal deben contener solo números

**Ventana Modificación:**

* Para ingresar, debe buscarse primero el proveedor a modificar y seleccionarlo.
* Al modificar un proveedor migrado, todos los campos que no existían previamente vendrán vacíos y deberán ser completados para poder guardar la modificación.

**ABM Rol:**

* El nombre del rol debe ser único
* Un usuario nuevo puede asignarse un rol nuevo, en ese caso las funcionalidades funcionan igual que si fuese Administrador General
* Al dar de baja un rol, se actualiza el campo habilitado, que activa un Trigger que quita el rol a los usuarios que lo tenían asignado

**Registro de Usuario:**

* Se realiza con una Transaction que contiene múltiples Stored Procedures
* Primero se verifica que el nombre de usuario no esté siendo usado
* Dependiendo de que rol se selecciona, se abre el correspondiente formulario
* Luego se verifica que el Cliente o Proveedor no se repita
* Por último, se inserta en las Tablas y se Actualizan las Foreing Keys

**Carga Crédito:**

* En la tabla maestra aparece que las cargas fueron realizadas por efectivo, decidimos agregar esa opción en nuestra aplicación a fines prácticos ya que, si el pago es con tarjeta o con efectivo no se controla si en realidad se pagó o no
* En el caso de las tarjetas se distingue entre Débito y Crédito mediante la tabla Tipo\_Pago y se almacena solo el número y el nombre del titular, la clave no se almacena por motivos de seguridad

**Creación de Ofertas**

* Como el código de la oferta no se menciona en el enunciado como un campo obligatorio a registrar, y su utilización es transparente para el usuario en el resto del sistema, decidimos mantenerlo solo para la migración, pero en el nuevo sistema no tiene ningún propósito y estará en NULL.
* Si el usuario no es un proveedor, el campo para este vendrá sin completar y al presionar “Seleccionar” podrá elegir de todos los proveedores para cuál crear la oferta.

Si por el contrario se trata de un usuario proveedor, el campo ya viene cargado con su CUIT.

* Se realizan validaciones a nivel aplicación:
  + Todos los campos son obligatorios.
  + El precio de lista no puede ser mayor ni igual que el precio de oferta.
  + La cantidad máxima por cliente no puede ser mayor que la cantidad disponible.
  + La fecha de vencimiento no puede ser menor o igual a la fecha de publicación.
* El botón “Limpiar todo” borra todos los datos completados.
* Las ofertas no pueden ser creadas con una fecha de publicación anterior a la configurada en el sistema.
* La fecha de vencimiento es por lo menos un día después de la fecha de publicación.

**Comprar Ofertas**

* El sistema muestra por pantallas todas las ofertas disponibles, para que no se muestren ofertas vencidas la tabla se llena a partir de la función Ofertas\_Disponibles que retorna una tabla verificando que las ofertas no estén vencidas
* Para efectuar la compra, se utiliza el Store Procedure nueva\_compra, que verifica que el usuario tenga saldo disponible.
* Para controlar el saldo, ese campo en la tabla Cliente tiene una constraint de CHECK que controla que el saldo nunca puede ser menor que cero

**Consumir Ofertas**

* La Aplicación muestra los cupones que pertenecen al proveedor, a través de la función cupones\_proveedor que además de retornar los cupones del proveedor suministrado por parámetro, devuelve solo los que no estén vencidos ni tampoco los que hayan sido entregados

**Facturación**

* El sistema permite y obliga a completar todos los campos: Proveedor, Fecha desde y hasta.
* El botón “Limpiar todo” borra todos los datos completados y reinicia las fechas a la fecha del sistema.
* Una vez completados los datos, el sistema se comportará de la siguiente manera, todo a través de un Stored Procedure:
  + Si existe una facturación creada para ese proveedor y en ese rango de fechas, trae los datos.
  + Si no existen compras en ese intervalo de fecha, arroja un mensaje indicando esta situación y no hace nada más.
  + Si existen compras y no existe una facturación ya creada en ese rango, crea una nueva factura y va relacionando los ítems en la tabla “Detalle\_Facturación” con ella.
  + Cada vez que se inserta/actualiza/elimina un elemento en la tabla “Detalle\_Facturación”, existe un trigger que actualiza el total de la factura correspondiente.

**Listado Estadístico**

**Otras Consideraciones**