ESTRATEGIAS BOBBY\_TABLES

DER:

* Cada intento de login (fallido o correcto) se guarda en la tabla de intentos, el usuario posee un contador de intentos fallidos por cada acceso.
* Se le dio identidad a las funcionalidades y se creó una relación de muchos a muchos con roles para asegurar la seguridad de acceso en la aplicación.
* Un usuario está directamente relacionado con una persona, pueden existir usuarios (administradores por ejemplo) sin personas. Pero no tendría sentido una persona sin usuario.
* Las tarjetas tienen los primeros 12 dígitos del número encriptados y los últimos cuatro sin encriptar. Se decidió que tenía más sentido encriptar los 12 que los 16.
* Las tarjetas y las cuentas están directamente ligadas con el usuario que es dueño de las mismas.
* Una operación está planeada para garantizar herencia (a nivel objetos) con depósitos, transferencias y retiros de efectivo, ya que tienen una interfaz en común.
* Las subclases de operación tienen sus atributos específicos asignados.
* Una comisión está directamente ligada con una operación a pesar de que las únicas operaciones facturables sean las transferencias para asegurar flexibilidad. A su vez, cada operación conoce también la cuenta y el tipo de cuenta siempre y cuando se trate de una operación por cambio de tipo de cuenta (o alta de la misma).
* Una cuenta posee una fecha de expiración que se calcula a la hora de la creación de la misma según el tipo de cuenta que se elija y a la hora de facturación cuando se abone los gastos de cuenta. (Si paga 10 abonos Oro, se le garantiza 10 veces el tiempo de expiración)

MIGRACIÓN:

* Se tomaron todos los países de la tabla maestra (los tres lugares donde figuraban con nombre y código) y se realizó una unión para abarcar la mayor cantidad de países registrados.
* Se asignaron costos hardcodeados a los tipos de cuenta ya que en la tabla maestra no había información al respecto. Se le permite al administrador elegir el costo a consignar desde el cambio en adelante.
* Se le asignó al pasaporte (como tipo de documento) un nuevo ID para facilitar la aplicación desktop. A cada cliente se le asignó luego el nuevo ID.
* Los usuarios se cargaron teniendo en cuenta que cada persona tenía que tener sí o sí un usuario, por lo tanto se les creo un usuario default cuyo username sería el mail y password vacía. Al entrar a la aplicación se recibirá un mensaje de usuario caducado y se le pedirá elegir un nuevo usuario y contraseña.
* Por defecto, se le agregó a cada persona como nacionalidad el país de residencia. A su vez, se le asignó de localidad una cadena vacía.
* Por defecto, a cada usuario se le agregó el rol de cliente.
* Por defecto, a cada cuenta se le asignó el tipo gratuita y el estado pendiente de activación. A su vez, se le asignó de saldo 0.
* Para las tarjetas se utilizó el encriptado SHA1.
* La información del cheque se acortó, teniendo en cuenta que la información se repite con la de la operación de retiro de efectivo.
* Se aseguró que cada operación tenga un registro donde coincida el ID con las correspondientes tablas de operaciones (depósitos, transferencias y retiros de efectivo).
* Para el caso de las transferencias que no contaban con un código, se utilizó una tabla temporal que nos asegurara unicidad de información y código para la operación y la transferencia. Así como el código de operación en la tabla facturas.
* Al momento de la creación de cuentas, las cuentas de destino con su información correspondiente fue desestimada ya que no hay manera de ligarlo con un usuario.

APLICACIÓN:

* Para reactivar un rol se debe ir al menú de alta roles, si se escribiese un nombre de rol que se encuentra deshabilitado, la aplicación provee herramientas para reactivarlo.
* Suponemos que un depósito sólo puede realizarse en una cuenta propia (del usuario), teniendo en cuenta que no tiene costos por depósito.
* Durante la baja de cliente, si el cliente a dar de baja posee un solo rol, se dará de baja también el usuario.
* Se decidió que el modificar cuenta permita cambiar el estado de la cuenta (principalmente para dejar que el administrador vuelva a dar de alta una cuenta dada de baja) pero esto me permite dar de baja desde modificar. Se decidió que lo mejor era hacerlo a pesar del problema.
* Fue decidido que durante la modificación de categoría de cuenta si la cuenta cambiada es igual a la actual se realiza el cambio igual, ya que eso genera una nueva fecha de vencimiento y una nueva comisión.
* Para el alta de cuentas se debe ingresar un cliente ya que una cuenta sin cliente no tiene sentido en la base.
* En varios lugares, debido a los números autogenerados, se decidió devolver el mismo luego de generar la operación (Ejemplo: el cheque en retiro de efectivo).

EXTRAS:

* Se decidió separar cada funcionalidad en ítems para tener el mayor control de quién tiene acceso a qué pantalla.
* Se decidió, por la manera en la que se diseñó la tarjeta, que no tenía sentido que el usuario pudiera editar los campos de la misma.
* Tomando en cuenta que no tenemos una operación que se ejecute diariamente, se decidió que la inhabilitación por falta de pago se produzca cuando el usuario desee rendir sus facturas.
* Se eligió como fecha de vencimiento de la tarjeta la fecha de emisión más tres años.
* Se pensó a la factura como una agrupación de comisiones que fueron pagadas. Una comisión se lleva a cabo una vez que se produce un efecto que genera un costo en el cliente.

FUNCIONALIDADES FALTANTES:

* Modificar Tarjeta de Crédito.
* Desasociar Tarjeta de Crédito.
* Asociar Tarjeta de Crédito desde el ABM Clientes (Tiene ciertas irrgularidades).
* Facturar.
* Consulta saldos (administrador).
* Permitir abonar N abonos de tipo cuenta. (Se nos pasó por alto)