



Departamento de Cs. e Ingeniería de la Computación
Universidad Nacional del Sur



BASES DE DATOS Segundo Cuatrimestre de 2018

Proyecto N° 3

Implementación de transacciones en un sistema de base de datos bancario

Transacciones Bancarias: Extracciones y Transferencias

Se desea proveer a los clientes la posibilidad de realizar a través de los cajeros automáticos dos tipos de transacciones sobre sus cajas de ahorro: extracciones y transferencias.

Dado que las cajas de ahorro pueden ser accedidas por más de un titular en diferentes momentos y por diferentes cajeros, se desea registrar cuál de los titulares realiza la transacción junto con la fecha, hora y código del cajero donde se realiza la misma. Antes de operar en un cajero, se solicita al cliente que inserte su tarjeta y mediante la misma se identifica al titular y la caja de ahorro sobre la cuál realizarán las transacciones.

Al momento de realizar una transacción (extracción o transferencia) se necesitarán conocer los siguientes datos:

- Código del ATM donde se realiza la transacción: Dicho código se encontrará almacenado internamente en cada cajero automático.
- Monto asociado a la transacción: Se debe verificar que dicho monto no supere el saldo disponible en la caja de ahorro.

La operatoria de las transacciones a implementar es la siguiente:

- Una extracción corresponde al retiro de cierto monto de dinero por parte de uno de los titulares de la caja de ahorro. Una vez identificado el titular y la caja de ahorro mediante la tarjeta ingresada, se solicita el monto a extraer, decrementando el saldo de dicha caja de acuerdo al monto ingresado.
- Una transferencia corresponde al movimiento de dinero hacia otra caja de ahorro del banco. Para efectuar dicho movimiento, es necesario identificar la caja de ahorro origen de los fondos y la caja de ahorro destino. La caja de ahorro origen se identifica mediante la tarjeta que es ingresada al cajero. La caja de ahorro destino se determinará luego de solicitar al cliente un número de caja de ahorro destino y verificar la existencia de la misma. Observese que el destino de los fondos corresponde a una caja de ahorro, sin considerar al titular de dicha caja.

Con la información de las cajas de ahorro origen y destino, se solicita el monto a transferir y se procede a decrementar el saldo de la caja de ahorro origen e incrementar el saldo de la caja de ahorro destino de acuerdo al monto ingresado. Por lo tanto, cuando se hace una transferencia en realidad deben almacenar dos nuevas transacciones:

- una transferencia asociada a la caja de ahorro origen y
- un depósito (por el mismo monto) asociado a la caja de ahorro destino

Como la base de datos puede ser accedida concurrentemente por varios ATM debe asegurarse que las transacciones se realicen de forma atómica y que los datos se bloqueen adecuadamente para garantizar la serializabilidad y evitar que la base de datos quede en un estado inconsistente.

Ejercicios

1. Mediante *stored procedures* implemente transacciones para:
 - a) Realizar una transferencia entre dos cajas de ahorro.
 - b) Realizar una extracción de una caja de ahorro.

Ambos *stored procedures* deberán devolver el resultado de la transacción, es decir, si la operación se realizó con éxito o no, indicando el motivo (Por ejemplo: en caso que no haya saldo suficiente, la caja de ahorro origen o destino no exista, etc).

Importante: Dado que el campo *nro.trans* de la tabla *Transaccion* es de tipo `AUTO INCREMENT`, deberá utilizar la función `LAST_INSERT_ID()` (Ver página 1766, sección 12.14 de refman-8.0-en.a4.pdf o página 615, sección 12.9.3. de refman-5.0-es.a4.pdf) para recuperar el número generado automáticamente al insertar una nueva transacción.

Los *stored procedures* deberán definirse dentro de la base de datos *banco*. Las sentencias de creación de dichos procedimientos deberán incluirse en el archivo "*banco.sql*". Además deberá extender los privilegios del usuario *atm* para que pueda ejecutar estos procedimientos.

2. **Extienda** las opciones del menú del cajero automático de la aplicación implementada en Java (correspondiente al ejercicio 2 b) del proyecto 2), para permitir realizar transferencias y extracciones desde la caja de ahorro asociada a la tarjeta ingresada. Asuma que el **código del cajero** implementado es 100. Se deberá agregar al menú las siguientes opciones:
 - Realizar transferencia: se permitirá ingresar un número de caja de ahorro destino y un monto a transferir. A continuación se deberá permitir confirmar o cancelar la transferencia. En caso de cancelar, se deberá volver al menú principal del cajero. En caso de confirmar, se realizará la transacción invocando al *stored procedure* implementado en 1a.
 - Realizar extracción: se permitirá ingresar un monto a extraer, y luego confirmar o cancelar la extracción. En caso de cancelar, se deberá volver al menú principal del cajero. En caso de confirmar, se realizará la extracción invocando al *stored procedure* implementado en 1b.

Una vez realizada la operación se deberá mostrar un mensaje que indique si la misma fué realizada con éxito o no, indicando el motivo. En el caso que la operación sea una extracción se deberá mostrar el saldo actual de la caja de ahorro.

3. Implemente un trigger asociado a la tabla *prestamo* que se active cuando se inserta un nuevo préstamo y cargue automáticamente todos pagos asociados al mismo. La sentencia de creación de dicho trigger deberá incluirse en el archivo "*banco.sql*". Recuerde eliminar de la aplicación Java el código que carga los pagos cuando se inserta un nuevo préstamo, para que no colisione con los pagos cargados automáticamente por el trigger.

Fechas y condiciones de entrega

- **Fecha límite de entrega: martes 30 de octubre de 2018** a través del curso Moodle de la materia y utilizando la tarea correspondiente habilitada para tal fin, se deberá subir un archivo comprimido (zip o rar) que contenga:
 - el archivo "*bancos.sql*" extendido con lo solicitado en los ejercicios 1 y 3
 - los archivos fuentes de la aplicación solicitada en el ejercicio 2 junto con el archivo JAR ejecutable correspondiente. **La aplicación deberá incluir las correcciones a todas las observaciones realizadas por la cátedra sobre la entrega del proyecto 2.**
- La entrega en *tiempo y forma* de este proyecto es *condición de cursado* de la materia.