

PRUEBA SUFICIENCIA BASES DE DATOS II

Su misión será crear un simple sistema bancario. Piense en su experiencia personal con su cuenta bancaria. En caso de duda, opte por la solución más sencilla. No es necesario crear ningún tipo de interfaz de usuario ni hacer que estas funciones estén disponibles a través de ningún servidor. El ejercicio evoluciona como una secuencia de iteraciones. Intente completar cada iteración antes de leer la siguiente.

¿Qué debe hacer?

Iteración 1: Crear función de depósito

Siga la siguiente historia de usuario:

Característica: depositar dinero en una cuenta

Como cliente del banco

quiero depositar dinero en mi cuenta

Para aumentar mi saldo

Escenario: un cliente existente deposita dinero en su cuenta

Dado un cliente existente con identificación “Diego” con \$500.000 pesos en su cuenta

Cuando deposita \$100.000 pesos en su cuenta

Entonces el saldo de su cuenta es de \$600.000 pesos.

Iteración 2: actualizar la función de depósito

Actualmente, los usuarios pueden depositar cantidades negativas de dinero, lo cual no tiene sentido. Agregue un nuevo caso de prueba para solucionar este problema.

Iteración 3: agregar la función de retiro

Puede seguir esta historia de usuario:

Característica: retirar dinero de una cuenta

Como cliente del banco

quiero retirar dinero de mi cuenta

Para tener efectivo

Escenario: un cliente existente retira dinero de su cuenta

Dado un cliente existente con identificación “Diego” con \$100.000 pesos en su cuenta

Cuando retira \$10.000 de su cuenta

Entonces el nuevo saldo es de \$90.000 pesos.

Iteración 4: Cubrir casos fronterizos para la función de retiro

Agregue un escenario en la función de retiro para el caso en que un retiro genere un sobregiro. No se debe permitir el retiro de montos superiores al saldo de la cuenta corriente.

Iteración 5: actualizar la función de retiro

Compruebe si es posible retirar un valor negativo. Si es posible, arréglole. Agregue el caso de prueba correspondiente para solucionar este problema.

Iteración 6: agregar la función de transferencia

A partir de ahora, los usuarios podrán transferir dinero desde su cuenta a la cuenta de otro usuario. Añada esta característica. Primero escriba una historia de usuario para ello. Escriba los escenarios para el caso principal, pero también escriba escenarios para casos fronterizos, como, por ejemplo: ¿se puede transferir dinero que no se tiene? Puede que haya otros.

Iteración 7: controlar transaccionalidad

Para cada escenario es necesario registrar el usuario, el cliente, monto, operación y demás elementos que considere deben ser tenidos en cuenta para garantizar la seguridad en el sistema.

Condiciones de Éxito

- Debe crear un MER (preferiblemente en MySQL Workbench) que detalle la solución del sistema bancario, tablas, relaciones, normalización y diccionario de datos.
- Utilice procedimientos almacenados con parámetros de entrada, salida, entrada/salida para obtener la información de los escenarios.
- Cada iteración debe ser registrada para el control de la transaccionalidad, se recomienda usar disparadores (Triggers) que actualicen la información en una entidad cada vez que se realice una operación.
- Puede utilizar estructuras de control como instrucciones condicionales o repetitivas, funciones o todo aquello que puede ser útil para la resolución de los escenarios.
- Debe entregar el manual de funcionamiento indicando el paso a paso para la ejecución de los escenarios.
- El plazo máximo de entrega es el 13 de marzo.
- Se debe realizar sustentación del proyecto entregado. Se recomienda sea el mismo día en el horario de 8:15 a 10:30 p.m.

MUCHOS ÉXITOS.