

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ciencias y Sistemas
Laboratorio Sistemas de Bases de Datos 1 B
1er Semestre 2024
Ing. Luis Fernando Espino
Aux. Edin Emanuel Montenegro Vásquez



Enunciado

PROYECTO #2

Introducción

La integridad y la gestión eficiente de los datos son aspectos cruciales para las empresas de servicios financieros, donde la precisión y la confiabilidad de la información son fundamentales. En este proyecto, se propone diseñar y desarrollar una base de datos para una empresa de servicios financieros que se dedica a la gestión de cuentas de clientes, transacciones y análisis de datos financieros. El objetivo principal es garantizar la integridad de los datos y proporcionar funcionalidades que permitan una gestión eficiente de cuentas, transacciones y análisis de riesgos financieros.

Objetivos

General:

Adquirir conocimientos y habilidades en la gestión de bases de datos relacionales aplicadas al sector financiero.

Específicos:

- Aplicar las primeras formas normales para la elaboración de un modelo relacional más optimizado.
- Asegurar la integridad relacional y aplicar prácticas de seguridad.
- Diseñar un sistema de base de datos que permita la gestión de cuentas de clientes, transacciones y análisis de datos financieros.
- Implementar procedimientos almacenados, funciones y triggers para garantizar la integridad de los datos y facilitar la gestión de la información financiera.
- Desarrollar funcionalidades específicas para la gestión de cuentas, transacciones y análisis de riesgos financieros.

Descripción del problema

La empresa de servicios financieros requiere una solución de base de datos que le permita gestionar de manera eficiente las cuentas de sus clientes, registrar y procesar transacciones financieras, y realizar análisis de datos para evaluar riesgos financieros. Se busca una solución integral que garantice la integridad de los datos y facilite la toma de decisiones financieras. Se le requiere a usted, como estudiante del curso de Sistemas de Bases de Datos 1, para brindar una solución inicial en fase beta, a nivel de base de datos para que usted sea el encargado de proponer, diseñar e implementar todo el flujo de los datos asegurando la persistencia e integridad de toda la información que se registre.

Se le dará la libertad de crear sus propios procedimientos, funciones y disparadores automatizados para el buen funcionamiento de la base de

datos de la entidad financiera; pero respetando las principales funcionalidades que se le requieren.

Requerimientos del proyecto

Diseño:

Se debe de realizar todo el proceso de modelado de la base de datos necesario para su creación. Lo cual consistirá en:

1. Modelo conceptual
2. Modelo lógico
3. Modelo relacional (Diagrama ER)

Funcionalidades:

La gerencia administrativa de la entidad financiera ha brindado todos los requerimientos funcionales de manera detallada para que usted los pueda implementar correctamente. Los cuales se detallan a continuación:

1. Registro de clientes

Se debe de almacenar los datos de todo cuentahabiente, incluyendo información personal y detalles de contacto. **También es necesario guardar la fecha exacta de su creación.**

registrarCliente

Parámetro	Tipo de dato	Observación
IdCliente	Integer(10)	
Nombre	Varchar (40)	*Validar que sólo sean letras
Apellidos	Varchar (40)	*Validar que sólo sean letras
Teléfono(s)	Varchar(12)	*Obviar código de área
Correo(s)	Varchar (40)	*Validar que sea un formato válido
Usuario	Varchar(40)	*Validar Disponibilidad (no repetido)
Contraseña	Varchar(200)	*Almacenar de manera encriptada

2. Registro de tipo de cliente

Se debe de categorizar cada cliente almacenado en la base de datos, ya que cada cliente tiene diferentes características.

registrarTipoCliente

Parámetro	Tipo de dato	Observación
IdTipo	Integer(10)	*Se debe de crear de manera auto incrementable empezando en 1
Nombre	Varchar(20)	
Descripción	Varchar(40)	*Validar que sólo sean letras

3. Registro de cuentas bancarias

Sirve para agregar la cuenta o cuentas que tengan los clientes de esta entidad financiera. **Se debe validar que no exista el número de cuenta.**

registrarCuenta

Parámetro	Tipo de dato	Observación
Id_cuenta	Integer(10)	
Monto_apertura	Decimal number (12,2)	*Este campo nunca debe modificarse, validar que sea positivo
Saldo_cuenta	Decimal number (12,2)	Validar que sea ≥ 0
Descripción	Varchar(50)	
Fecha de apertura	DATETIME	*Se debe de agregar automáticamente.
Otros detalles	Varchar (100)	*Campo no es obligatorio

4. Tipo de cuenta

Este servirá para ingresar los tipos de cuentas validos en el sistema financiero. Ya que en toda entidad financiera existen tipos de cuenta tales como, ahorro, monetaria, etc.

registrarTipoCuenta

Parámetro	Tipo de dato	Observación
-----------	--------------	-------------

Código	Integer(10)	*Se debe de crear de manera auto incrementable empezando en 1
Nombre	Varchar(20)	
Descripcion	Varchar(100)	

5. Productos y servicios

Como toda entidad financiera, un cliente puede pedir productos o servicios financieros, tales como prestamos, créditos, asesoramiento, tarjeta de crédito, débito, banca virtual, pagos electrónicos, etc. Este servirá para agregar estos servicios mencionados.

crearProductoServicio

Parámetro	Tipo de dato	Observación
Código del producto / servicio	Integer	*Autoincrementable
Tipo	Integer (1)	*1 para servicio; 2 producto
Costo	Decimal number (12,2)	* Este campo solo es obligatorio, si se trata de un servicio ya que estos tienen un precio definido.
Descripción producto / servicio	Varchar (100)	

6. Compras

Servirá para que el cliente pueda acceder a los servicios y productos que la entidad financiera ofrece.

realizarCompra

Parámetro	Tipo de dato	Observación
Id_compra	Integer	*Autoincrementable
Fecha	Date	
Importe compra	Decimal / Number (12,2)	* Este campo solo es obligatorio si se trata de un producto.

Otros Detalles	Varchar (40)	Campo es opcional
-----------------------	--------------	-------------------

7. Depósitos

Servirá para que el cliente pueda depositar dinero a la cuenta correspondiente.

realizarDeposito

Parámetro	Tipo de dato	Observación
Id_compra	Integer	*Auto incrementable
Fecha	Date	
Monto	Decimal / Number (12,2)	
Otros Detalles	Varchar (40)	Campo es opcional

8. Débitos

Servirá para que el cliente pueda retirar dinero de la cuenta correspondiente.

realizarDebito

Parámetro	Tipo de dato	Observación
Id_compra	Integer	*Auto incrementable
Fecha	Date	
Monto	Decimal / Number (12,2)	
Otros Detalles	Varchar (40)	Campo es opcional

9. Transacciones

Este servirá como toda transacción funciona en una entidad financiera, luego de realizar una compra, deposito, debito, etc. Se debe de validar el saldo en cuenta de cliente y realizar el crédito o debito correspondiente en la cuenta respectiva.

asignarTransaccion

Parámetro	Tipo de dato	Observación
Id transacción	Integer	Auto incrementable.

Fecha	Date	
Otros Detalles	Varchar (40)	Campo es opcional

10. Registrar tipo transacciones

Servirá para poder tener un listado del tipo de transacciones validas en este sistema financiero (compras, depósitos, etc.).

registrarTipoTransaccion

Parámetro	Tipo de dato	Observación
Código transacción	Integer	Auto incrementable.
Nombre	Varchar(20)	
Descripcion	Varchar(40)	

Procesamiento de datos:

La gerencia administrativa de la entidad financiera le solicita que se extraiga la información necesaria a través de los datos almacenados con el fin de poder generar reportes y constancias necesarias para los distintos procesos administrativos y de los clientes.

Para ello es necesario elaborar los siguientes procedimientos:

1. Consultar Saldo Cliente

Debe de retornar el saldo de la cuenta proporcionada por el cliente.

consultarSaldoCliente

Parámetro	Tipo de dato	Observación
Numero de cuenta	Numérico	* Mostrar error si no existe

Resultado esperado:

✓ Nombre Cliente

- ✓ Tipo de cliente
- ✓ Tipo de cuenta
- ✓ Saldo cuenta
- ✓ Saldo apertura

2. Consultar cliente

Debe retornar la información de un cliente según su id.

consultarCliente

Parámetro	Tipo de dato	Observación
idCliente	Numérico	*Si no existe mostrar error

Resultado esperado:

- ✓ Id cliente
- ✓ Nombre completo (concatenado)
- ✓ Fecha de creación
- ✓ Usuario
- ✓ Teléfono(s)
- ✓ Correo(s)
- ✓ No dé cuenta(s) que posee (cantidad)
- ✓ Tipo(s) de cuenta que posee

3. Consultar movimientos cliente

Devuelve todos los movimientos (transacciones) realizados por un cliente específico.

consultarMovsCliente

Parámetro	Tipo de dato	Observación
idCliente	Numérico	*Si no existe mostrar error

Resultado esperado:

- ✓ Id transacción
- ✓ Tipo transacción
- ✓ Monto
- ✓ Devolver el tipo de servicio que se hizo (compra, debito, crédito, etc.)

- ✓ No. cuenta
- ✓ Tipo cuenta.

4. Consultar clientes por tipo de cuenta

Debe retornar el listado de clientes que tienen el tipo de cuenta ingresado.

Si no se encuentra debe mostrar un error.

consultarTipoCuentas

Parámetro	Tipo de dato	Observación
Id de tipo de cuenta	Numérico	*Si no existe mostrar error

Resultado esperado:

- ✓ Código de tipo de cuenta
- ✓ Nombre cuenta
- ✓ Cantidad de clientes que poseen ese tipo de cuenta

5. Consultar movimientos generales por rango de fechas

Debe retornar el listado de transacciones realizados en ese rango de fechas de todos los clientes.

consultarMovsGenFech

Parámetro	Tipo de dato	Observación
Fecha Inicio	Date	*Validar formato de fecha (DD/MM/YY)
Fecha Fin	Date	*Validar formato de fecha (DD/MM/YY)

Resultado esperado:

- ✓ Id transacción
- ✓ Tipo transacción
- ✓ Devolver el tipo de servicio que se hizo (compra, debito, crédito, etc.)
- ✓ Nombre Cliente
- ✓ No cuenta
- ✓ Tipo de cuenta

- ✓ Fecha
- ✓ Monto
- ✓ Otros detalles

6. Consulta de Movimientos por Rango de Fechas para un Cliente Específico

Permite consultar los movimientos financieros realizados por un cliente específico dentro de un rango de fechas determinado. Los movimientos incluyen depósitos, débitos, compras u otras transacciones realizadas en la cuenta del cliente durante el período especificado.

consultarMovsFechClien

Parámetro	Tipo de dato	Observación
Id cliente	Number	*Mostrar error si no existe
Fecha Inicio	Date	*Validar formato de fecha (DD/MM/YY)
Fecha Fin	Date	*Validar formato de fecha (DD/MM/YY)

Resultado esperado:

- ✓ Id transacción
- ✓ Tipo transacción
- ✓ Devolver el tipo de servicio que se hizo (compra, debito, crédito, etc.)
- ✓ Nombre Cliente
- ✓ No cuenta
- ✓ Tipo de cuenta
- ✓ Fecha
- ✓ Monto
- ✓ Otros detalles

7. Consultar productos y servicios

Consulta el listado de servicios y productos que tiene la entidad financiera

consultarDesasignacion

Parámetro	Tipo de dato	Observación
-----------	--------------	-------------

Resultado esperado:

- ✓ Código de servicio / producto
- ✓ Nombre
- ✓ Descripción
- ✓ Tipo

Historial de transacciones

Se debe llevar un control de transacciones por medio de una tabla adicional en donde se registrará automáticamente (por medio de triggers) cada vez que ocurra una inserción, modificación o eliminación en una tabla. Se debe de almacenar la fecha y hora exacta, y el nombre de la tabla que se vio afectada.

Tabla Ejemplo:

Fecha	Descripción	Tipo
YYYY-MM-DD hh:mm:ss	Se ha realizado una acción en la tabla "Nombre".	INSERT/UPDATE/DELETE

Tipos utilizados

Cientes

Id	Nombre	Descripción
1	Individual Nacional	Este tipo de cliente es una persona individual de nacionalidad guatemalteca.

2	Individual Extranjero	Este tipo de cliente es una persona individual de nacionalidad extranjera.
3	Empresa PyMe	Este tipo de cliente es una empresa de tipo pequeña o mediana.
4	Empresa S.C	Este tipo de cliente corresponde a las empresa grandes que tienen una sociedad colectiva.

Cuentas

Id	Nombre	Descripción
1	Cuenta de Cheques	Este tipo de cuenta ofrece la facilidad de emitir cheques para realizar transacciones monetarias.
2	Cuenta de Ahorros	Esta cuenta genera un interés anual del 2%, lo que la hace ideal para guardar fondos a largo plazo.
3	Cuenta de Ahorro Plus	Con una tasa de interés anual del 10%, esta cuenta de ahorros ofrece mayores rendimientos.
4	Pequeña Cuenta	Una cuenta de ahorros con un interés semestral del 0.5%, ideal para pequeños ahorros y movimientos.
5	Cuenta de Nómina	Diseñada para recibir depósitos de sueldo y realizar pagos, con acceso a servicios bancarios básicos.
6	Cuenta de Inversión	Orientada a inversionistas, ofrece opciones de inversión y rendimientos más altos que una cuenta de ahorros estándar.

Productos / Servicios

Id	Nombre	Descripción	Costo
1	Servicio de tarjeta de debito	Tipo = 1	Q. 10.00
2	Servicio de chequera	Tipo = 1	Q. 10.00
3	Servicio de asesoramiento financiero	Tipo = 1	Q. 400.00
4	Servicio de banca personal	Tipo = 1	Q. 5.00
5	Seguro de vida	Tipo = 1	Q. 30.00
6	Seguro de vida plus	Tipo = 1	Q. 100.00
7	Seguro de automóvil	Tipo = 1	Q.300.00
8	Seguro de automóvil plus	Tipo = 1	Q. 500.00
9	Servicio de deposito	Tipo = 1	Q. 00.05
10	Servicio de Debito	Tipo = 1	Q. 00.10
11	Pago de energía Eléctrica (EEGSA)	Tipo = 2	Este varía al realizar en compras (el valor de pago de factura).
12	Pago de agua potable (Empagua)	Tipo = 2	Este varía al realizar en compras (el valor

			de pago de factura).
13	Pago de Matricula USAC	Tipo = 2	Este varía al realizar en compras (el valor de pago de factura).
14	Pago de curso vacaciones USAC	Tipo = 2	Este varía al realizar en compras (el valor de pago de factura).
15	Pago de servicio de internet	Tipo = 2	Este varía al realizar en compras (el valor de pago de factura).
16	Servicio de suscripción plataformas streaming	Tipo = 2	Este varía al realizar en compras (el valor de pago de factura).
17	Servicios Cloud	Tipo = 2	Este varía al realizar en compras (el valor de pago de pago

			de factura).
--	--	--	-----------------

Entregables

- Modelos conceptual, lógico y relacional
- Script de la base de datos y datos de carga
- Funciones, procedimientos y triggers implementados
- Para poder calificarse el estudiante debe cargar los datos de los tipos especificados en su tabla correspondiente.

Restricciones

- El gestor de base de datos a utilizar puede ser **MySQL, Oracle o SQL Server**.
- **Use el nombre dado para cada función o procedimiento con la misma notación que se tiene "camelCase"**
- El proyecto es individual.
- Sistema operativo libre.
- Pueden utilizar Docker o una instancia local para su base de datos.
- Todo puede ser presentado desde su gestor nativo o IDE.
- El código y todo lo relacionado a su proyecto debe alojarse en el repositorio creado para el primer proyecto, en una nueva carpeta con el nombre **Proyecto 2**.

- En caso de copias totales o parciales será anulado y reportado a escuela.

Entrega

- Fecha límite: domingo 29 de abril de 2024 hasta las 09:00 hrs.
- La entrega se hará vía UEDi por medio del enlace a su repositorio.