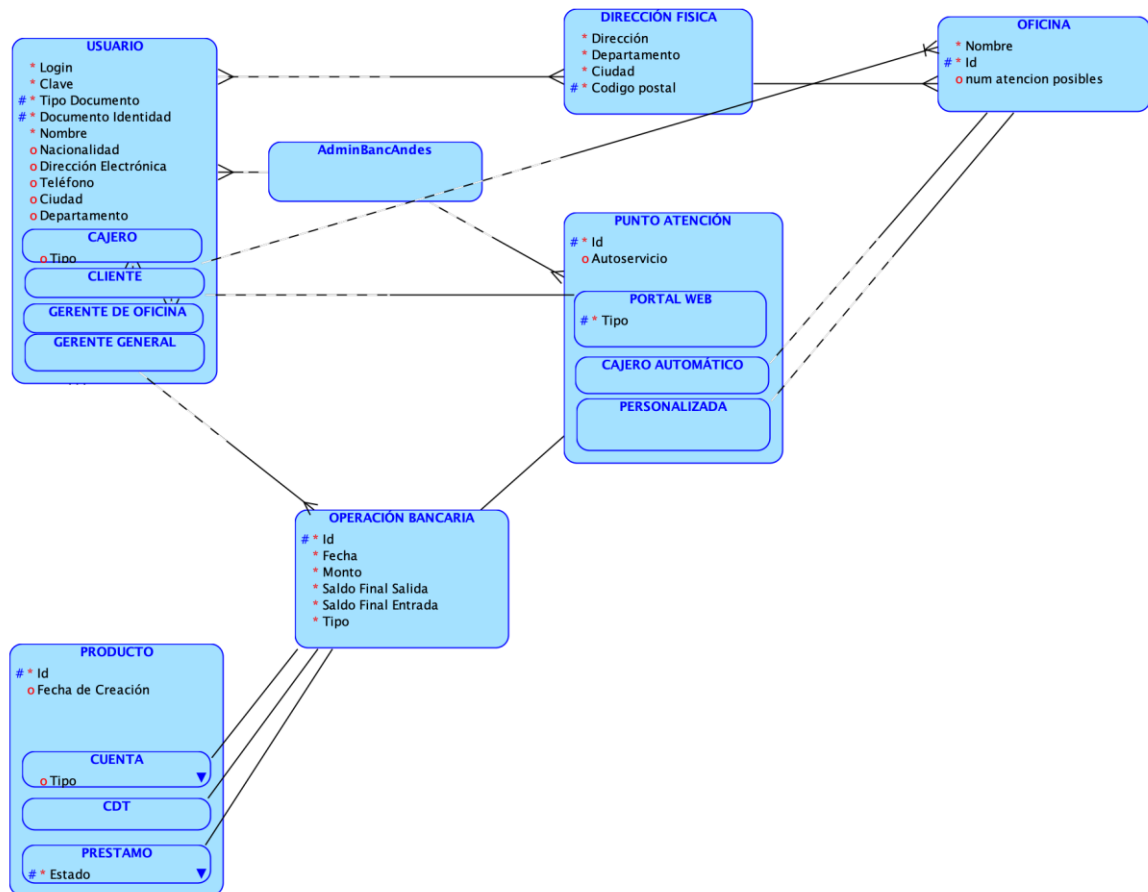


Juan Esteban Quiroga 202013216, Mateo Rincón 202221402, Luis Pinilla 202221306

[illegible]

Modelo relacional Explicado



Usuario-Dirección Física:

Como parte del registro que tienen los usuarios, deben poner su dirección, a modo de evitar la duplicidad de datos, escogimos guardar esa información en otra entidad. De esa manera, también podemos decir que un usuario puede tener múltiples direcciones y por eso la relación es de tipo mucho a muchos, no obstante, una dirección no está necesariamente asignada a muchos usuarios, por eso es así la obligatoriedad.

Dirección- Oficina:

En este caso, la oficina tiene una dirección asignada por obligatoriedad. Es decir, no puede existir una oficina si no tiene una dirección. Por otro lado, una dirección puede tener muchas oficinas asociadas. Por ejemplo, la dirección de un edificio que tenga múltiples oficinas de la empresa.

Oficina-Cajero Automático:

Una oficina puede o no tener un cajero automático, pero si existe un cajero automático necesariamente tiene que estar asociado a una oficina. Por eso la relación solo es obligatoria para el cajero automático.

Punto de Atención Personalizada:

Igual que el cajero automático, una oficina puede o no tener un punto de atención personalizada, pero si existe un punto de atención personalizada, tiene que estar en una oficina.

Punto atención- Operación :

Según si es un punto de atención personalizada, cajero, un punto regular o una entidad web, todas van son puntos en los que se solicita ejecutar operaciones por los usuarios (incluyendo los gerenciales). Y cada una de esas operaciones está asociada si o si a uno de los puntos de atención.

Operación Cuenta, Operación CDT y Operación Préstamo:

Se refiere a los requerimientos funcionales en los que los usuarios hacen solicitudes (clientes y gerenciales). Para los usuarios son: abrir cuenta, transferir dinero, retirar dinero, pagar créditos etc. Por parte de los gerentes son relaciones como: cerrar cuenta por motivos de seguridad, denegar crédito etc. Esta relación es obligatoria para ambos lados, porque no se puede ejecutar ninguna de estas acciones si no existe la operación y si no hay un producto asociado. Además, una operación solo puede relacionarse a UN producto, pues no se puede sacar dinero o solicitar múltiples préstamos en la misma operación, por eso la relación es uno a uno.

Gerente General- Operación y Cliente-Operación:

Los únicos usuarios que hacen operaciones son los clientes y los gerentes. A diferencia de otros usuarios, el gerente es el único que puede solicitar operaciones sin un punto de atención. El Gerente y los clientes pueden o no hacer solicitudes, pero si hay una solicitud, tiene que estar conectada si o si a un usuario. Además estos usuarios pueden hacer múltiples solicitudes. Por ese motivo la relación es de uno a muchos y además es opcional para el usuario.

Cliente- Portal Web:

Este es una relación de uno a uno, pues un cliente solo puede acceder a un portal web (solo existe una aplicación o una página). Además un usuario puede decidir si accede o no al portal web. Por eso motivo la relación es uno a uno y opcional para el usuario.

Cliente- oficina:

Un cliente tendría que ir a una oficina si quiere acceder a los productos bancarios si no quiere usar el portal web. El usuario tiene muchas opciones de oficina a las que puede ir, y por eso es una relación de uno a muchos.

AdminBankandes- usuario y AdminBank andes - Punto de Atención:

Este es un super usuario que tiene el poder de crear las cuentas de los usuarios y también tiene el poder de eliminar y crear puntos de atención. Estas dos relaciones son las más importante a la hora de que la empresa pueda expandirse. Existe un único super usuario de este tipo y puede o no crear múltiples usuarios o puntos de atención. Por eso la relación es de uno a muchos en ambos casos.

Cajero-Cliente:

El cajero tiene el deber de atender a muchos clientes. Un único cajero puede atender a muchos clientes y si un cliente fue atendido necesariamente fue por un cajero. Por ese motivo, la relación es de uno a muchos y además es obligatoria en ambos casos

Diseño de pruebas:

Pruebas de unicidad:

- Tratar de ingresar más de un usuario con un mismo login (Rf 1)
- Tratar de ingresar una dirección con la misma dirección ciudad y departamento
- Tratar de crear una oficina con una dirección o nombre repetidos (Rf2)
- Tratar de borrar un punto de atención(Rf2)
- Consultar cuenta no creada(Rfc1)
- Consultar cuenta no perteneciente al cliente (Rfc1)
- Consultar cuenta no ligada a la oficina(Rfc1)
- Consultar cliente inexistente (Rfc2)
- Consultar cliente (Rfc2)
- Consultar extracto bancario de una cuenta inexistente (Rfc3)
- Consultar extracto bancario de una cuenta (Rfc3)

Pruebas de integridad:

- Intentar crear producto (cuenta o prestamo) con referencia a un cliente(Rf3, Rf5)
- Intentar crear producto (cuenta o prestamo) con referencia a un cliente inexistente(Rf3, Rf5)
- Intentar realizar una operación ligada a un producto que no se haya creado
- Intentar realizar una operación ligada a un producto existente
- Intentar acceder a un cajero automático o puntos personalizados en un punto físico u oficina que no cuente con cajeros automáticos o puntos personalizados.
- Intentar acceder a un cajero automático o puntos personalizados en un punto físico u oficina que cuente con cajeros automáticos o puntos personalizados.
- Tratar de crear un punto de atención ligado a una oficina inexistente (Rf2)
- Tratar de crear un punto de atención ligado a una oficina existente (Rf2)
- Tratar de desactivar una cuenta con saldo(Rf4)
- Tratar de desactivar una cuenta(Rf4)
- Tratar de hacer una operación sobre una cuenta desactivada(Rf4)
- Tratar de cerrar un préstamo(Rf5)
- Tratar de cerrar un préstamo con saldo pendiente mayor a cero(Rf5)
-

Pruebas de integridad de acuerdo con restricciones de chequeo:

- Intentar crear un usuario con tipo de documento no permitido(Rf1)
- Intentar crear un empleado que no sea gerente de oficina o cajero(Rf1)
- Intentar crear una dirección con un departamento que no esté en la lista de departamentos
- Intentar crear una dirección con una ciudad que no esté en la lista de ciudades
- Intentar crear un préstamo con un día corte que no esté entre 0 y 30(Rf5)
- Intentar crear un préstamo con interés negativo(Rf5)
- Intentar crear un punto físico que no sea cajero o punto personalizado
- Intentar crear un portal web que no sea página o app

Análisis de normalización de las tablas:

Cómo parte del algoritmo modificado de Chen y teniendo en cuenta que una forma normal implica que ya estuviera en la forma normal anterior podemos hacer el análisis de normalización de nuestras tablas de la siguiente manera:

Paso 1 – Primera forma normal.

Para hacer que nuestras tablas se encontrasen en primera forma normal debíamos identificar aquellos atributos de los cuáles se podría tener más de uno en una columna y los separamos en distintos registros o tupla para garantizar que no haya más de un registro por casilla. Esto puede referenciar al ejemplo del tablero de ajedrez, donde solo puede haber una pieza por casilla. En nuestra situación pudimos identificar esto de varias maneras. Un ejemplo es que a la hora de ingresar un cliente nos dimos cuenta de que puede tener más de una cuenta asociada por lo que si tuviésemos una columna llamada “cuentas” estaríamos entrando en un problema ya que se podría prestar para que haya más de una cuenta por casilla. Ejemplos cómo este hay varios en el modelo, pero este fue el más claro.

Paso 2 – Segunda forma normal

La segunda forma normal se enfoca en que no se den dependencias parciales. Es decir, un atributo de la tabla tiene que depender de la llave primaria, de lo contrario debería estar en otra tabla. De la mano con el ejemplo de la primera normal también pudimos identificar esto con la relación entre los clientes y sus cuentas. Por ejemplo, ciertos atributos como lo son el tipo de cuenta o el saldo de la misma son cosas que no dependen del login de un usuario. Por esto se tomó la decisión de crear una nueva tabla que contuviera la información de cada cuenta ya que los atributos propios de las cuentas si están ligados a la llave primaria de la cuenta más no a la llave primaria del dueño de la cuenta. Por esto en cada cuenta incluimos el id del usuario para establecer la relación, pero no hay necesidad de incluir toda la información del usuario en la tabla de las cuentas o viceversa. Igualmente, como en el paso 1 identificamos estas dependencias parciales en la mayoría de las tablas por lo que para alcanzar la segunda forma normal tuvimos que dividir nuestras tablas grandes en tablas más pequeñas pero que cumplieran con los requisitos para que estuviesen en segunda forma normal.

Paso 3 – tercera forma normal

Hacer que nuestras tablas llegasen a estar en tercera forma normal realmente no fue muy complicado. Esto es porque la tercera forma normal se enfoca en no tener dependencias transitivas o sea que no sea posible llegar a un atributo en una tabla mediante otro atributo que se encontrara en la misma tabla. No pudimos identificar muchos casos pero si tuvimos un pequeño dilema con las direcciones. Al principio pensamos que al tener el id de una dirección podíamos tener tablas de direcciones y usar el id solo como llave primaria. Sin embargo, nos dimos cuenta de que si la única llave primaria es el id nada nos está garantizando que no pueda existir otra dirección igual, pero con un id distinto. Por esto identificamos que el menor conjunto posible para identificar una dirección tenía que ser la unión de departamento, ciudad y dirección. Así, obtuvimos una llave primaria compuesta por tres atributos. Esto fue algo notable en nuestro modelo ya que si hubiéramos asumido que al identificar una dirección por solo un id nos hubiéramos deshecho de una dependencia transitiva realmente hubiéramos cometido un error. Por esto también es que tener varios atributos de una dirección no es una dependencia transitiva, sino que los tres atributos forman la llave primaria.

Paso 4 - Forma Normal de Boyce-Codd (BCNF)

La forma normal de Boyce-Codd(BCNF) es particular porque en varios casos al llegar a la tercera forma normal ya se encuentra en BCNF. Sin embargo, en nuestro caso solo hubo una tabla posiblemente no se encontraba de esta manera y es la tabla de direcciones. Esto es porque es la única que tenía una llave compuesta. Sin embargo, retomando lo que nos pasó en la tercera forma normal, identificamos que un id no bastaba para identificar las direcciones y garantizar que no tuviéramos datos redundantes. Después de esto llegamos a la conclusión de que no había un conjunto de elementos más pequeño al que escogimos que nos pudiera proveer con toda la información que necesitábamos para identificar una dirección y garantizar que no hubiera redundancia de datos y es por esto que aunque no fue muy claro al principio si podemos decir que tanto esa tabla como el resto se encuentran en forma normal de Boyce-Codd.

Justificación tablas:

- Usuarios: Esta representa la entidad usuario, que funciona como entidad padre de los diferentes tipos de usuarios que tiene BankAndes.
- Cientes: Esta representa a la entidad cliente, que hereda la información de usuario, pues es un tipo de usuario.
- Empleado: Esta representa a la entidad empleado, que hereda la información de usuario, pues es un tipo de usuario. También tiene su tipo de empleado y a la oficina a la que está ligado.
- Gerente general: Esta representa a la entidad gerente general, que hereda la información de usuario, pues es un tipo de usuario. Si bien él es un tipo de empleado, tiene atributos y responsabilidades diferentes a los otros empleados.
- Direcciones_f: Representa la entidad direcciones física, con una llave compuesta por direccion, ciudad y departamento para asegurar que sea única.
- Direcciones-Usuario: Representa la relacion de 1* a * entre direcciones_f y usuario. La tabla lleva las PK de ambas tablas.
- Productos: Representa la entidad producto, su llave primaria es su id.
- Cuentas: Representa la entidad cuenta, esta hereda de producto. Su llave primaria es el id.Productos.
- Prestamos: Representa la entidad prestamo, esta hereda de producto. Su llave primaria es su id.Productos.
- Operaciones: Representa la entidad operaciones, esta tambien contiene como atributos las relaciones de composición con producto salida, cuenta y punto de atencion.
- Puntos de atencion: Representa la entidad punto de atencion.
- Puntos fisicos: Representa la entidad punto fisico, que hereda de puntos de atención. Además, como atributo esta guarda la oficina a la que está relacionada.
- Portales web: Representa la entidad portal web, que hereda de puntos de atención.
- Oficinas: Representa la entidad oficina, esta lleva toda la información relacionada con la oficina.