

Datos 1

Catedrático: Luis Fernando Alonzo Jerónimo

VERITAS • LIBERTAS • JUSTITIA



UFM
UNIVERSIDAD
FRANCISCO
MARROQUÍN

**FACULTAD de
CIENCIAS ECONÓMICAS**

Esquema de base de datos de una corporativa.

Nombre: David Gabriel Corzo Mcmath

Carné: 20190432

Fecha: 8 de febrero de 2021

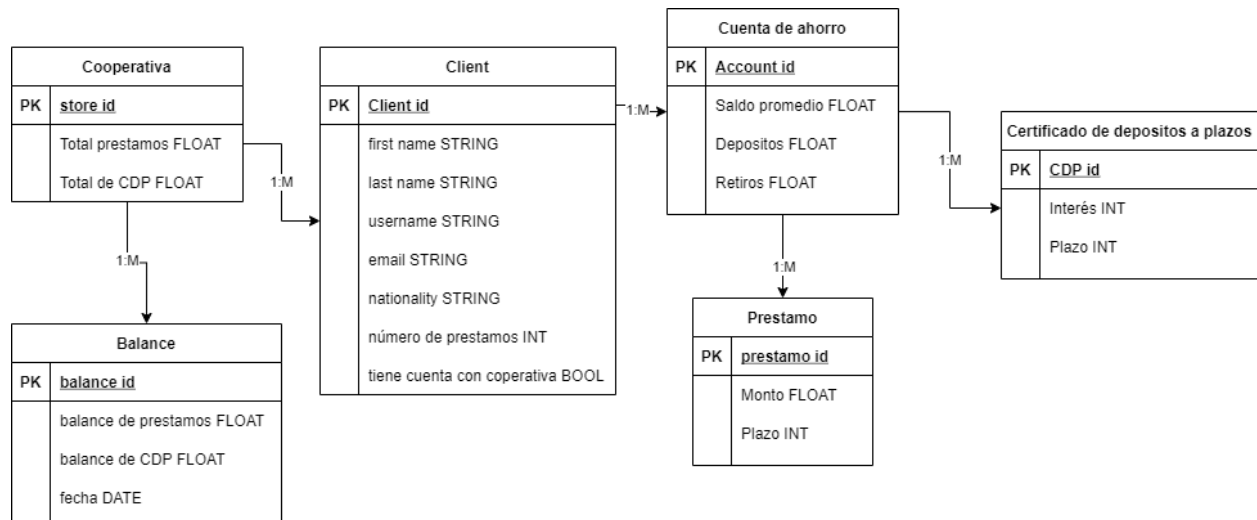
Diagrama Entidad-Relación:

Entidades:

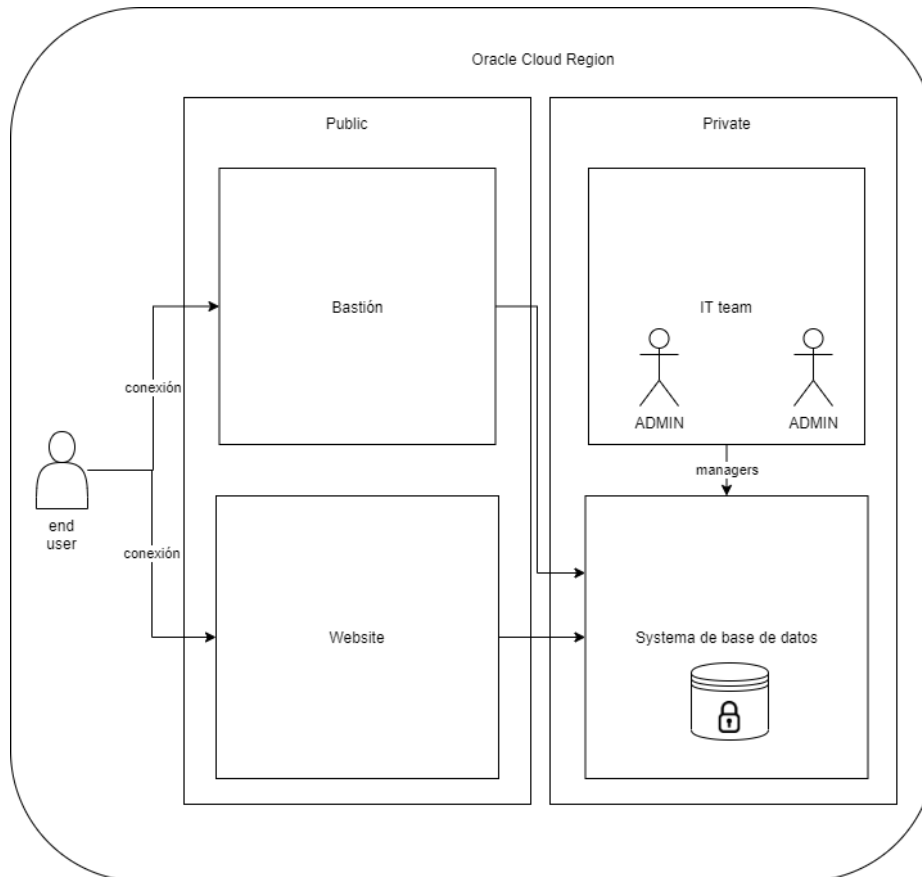
- **Cooperativa:** La entidad cooperativa se utiliza principalmente para poder ver el balance bancario, el total de préstamos que ha efectuado el banco y el total de cuentas certificadas de depósito puesto a que se debe mantener un control cercano de cuánto se presta y cuánto se retiene, en este caso el total de préstamos no debe de superar el 50% de los certificados de depósito.
- **Balance:** la entidad balance almacena los balances diarios que la cooperativa genera diariamente, como atributos se incluyen el balance de préstamos, balance de certificados de depósito y la fecha en la que se generó.
- **Cliente:** la entidad cliente comprende los atributos id (que es la llave primaria), primer nombre, apellido, usuario, correo electrónico, nacionalidad, la cantidad de préstamos que ha pedido y finalmente un booleano que nos permite ver si tiene cuenta con la cooperativa.
- **Cuenta de ahorro:** la entidad cuenta de ahorro debe de tener los atributos de: saldo promedio puesto a que se debe de estar constantemente calculando puesto a que el monto máximo permitido a otorgar en un préstamo es de cinco veces su promedio de los últimos 6 meses, adicionalmente se debe de tener los depósitos que ha realizado y los retiros que ha hecho, estas deberán de ser cantidades no negativas.
- **Préstamo:** La entidad préstamo almacena los atributos de monto y plazo puesto a que el monto otorgado debe de ser pagado por el cliente antes que el plazo se cumpla.
- **Certificado de depósitos a plazos:** la entidad certificado de depósitos a plazo almacena el interés y el plazo en el que se le va a ser reintegrado al cliente su inversión.

Relaciones:

- **Relación 1:M entre Cooperativa y balance:** esta relación tiene cardinalidad de uno a muchos puesto que una cooperativa puede y en su mayoría tendrá muchos balances diarios mientras que los balances no tendrán muchas corporativas asociadas con ellos sino solamente uno.
- **Relación 1:M Cooperativa con Cliente:** la relación tiene cardinalidad uno a muchos puesto a que clientes en esta base de datos no están asociados a muchas cooperativas sino a la cooperativa en singular, mientras la cooperativa puede y debería de poder tener muchos clientes.
- **Relación 1:M Cliente con Cuenta de Ahorro:** la cardinalidad es 1:M puesto a que un cliente puede tener muchas cuentas con la cooperativa mientras que una cuenta solamente puede estar asociada a un sólo cliente.
- **Relación 1:M Cuenta de Ahorro y Préstamo:** la cardinalidad es 1:M puesto a que la cuenta de ahorro puede tener asociada a ella muchos préstamos pero un préstamo solamente puede estar asociado a una cuenta de ahorro.
- **Relación 1:M cuenta de ahorro con certificado de depósitos a plazos:** la cardinalidad es 1:M puesto a que una cuenta de ahorro puede tener asociada a ella múltiples certificados de depósito mientras un certificado de depósito solamente puede estar asociado a una cuenta de ahorro.



Arquitectura:



Casos de uso:

En este caso se intenta crear un sistema que utilice una base de datos relacional para proveer servicios de banco a clientes y otros usuarios finales de parte de una corporación. En este caso los casos de uso para las bases de datos son en las entidades y las tablas de la base de datos relacional, esta comprende entidades como las de cooperativas, clientes, cuentas de ahorro, préstamos, certificados de depósitos y balance. Almacenar información de estas entidades es especialmente útil dado a que para hacer cualquier transacción bancaria se deben cumplir

ciertos requisitos tales como las del préstamo que debe cumplir un promedio previo a su realización.

Todo esto nos permite tener una base de datos segura y robusta en la que el usuario pueda tener una experiencia inolvidable tomando en cuenta el hecho que será fácil y versátil y usar.

Estructuras de almacenamiento: Físicas y lógicas.

Las estructuras lógicas se dividen en dos, primero las físicas. En esta aplicación se debe tomar en cuenta que las estructuras físicas de la base de datos son manejadas por el sistema operativo y se llaman data files, en dichos data files se almacena físicamente la información de la base de datos y se gestiona como cualquier otro archivo por el sistema operativo. Por otro lado están las estructuras lógicas, las estructuras lógicas permiten que Oracle conduzca operaciones en la base de datos con dichos data files mediante el sistema operativo. Las estructuras lógicas en su mayoría comprenden el sistema de manejo de bases de datos que se utilice.

Estimación de espacio y capacidad:

La estimación de espacio por entidad y relación: (Asumiendo una arquitectura de 64 bits)

- **Cooperativa: ~8 bytes.**
 - $2 * \text{FLOAT}$
 - $2 * 4 = 8 \text{ bytes}$
- **Balance: ~17 bytes**
 - $2 * \text{FLOAT} + 1 * \text{DATE}$
 - $2 * 4 + 1 * (4 + 4 + 1) = 8 + 9 = 17$
- **Cliente: ~46 bytes**
 - $5 * \text{STRINGS} + 1 * \text{INT} + 1 * \text{BOOL}$
 - $5 * 8 * (20 + 20 + 20 + 40 + 15) + 4 + 1 = 4600 \text{ bytes.}$
- **Cuenta de ahorro: ~12 bytes.**
 - $3 * \text{FLOAT}$
 - $3 * 4 = 12 \text{ bytes.}$
- **Préstamo: ~8 bytes.**
 - $1 * \text{FLOAT} + 1 * \text{INT}$
 - $1 * 4 + 1 * 4 = 8 \text{ bytes.}$
- **Certificados de depósito: ~8 bytes.**
 - $2 * \text{INT} = 2 * 4 = 8 \text{ bytes.}$

La suma del espacio ocupado por las entidades es aproximadamente 99 bytes. Hay que considerar que no todos se multiplicarán, tendremos una cooperativa, muchos balances, muchos clientes, muchas cuentas de ahorro y muchos préstamos. Multiplicar eso por 200,000 a 500,000 usuarios nos da un rango aproximado de 18,200,008-45,500,008 bytes asumiendo que cada cliente tiene una cuenta ha hecho un préstamo y un certificado de depósito, estos son variables pues puede que algunos no pidan préstamos o den depósitos. La capacidad de la aplicación tendrá que ser suficiente para almacenar todas las posibles acciones que podrían hacer los clientes.

Perfil de usuarios:

Los usuarios finales de la aplicación serán tanto los clientes de la corporativa como los empleados de la misma. Debido a este hecho en la aplicación se deben de manejar permisos, tales como por ejemplo los permisos que tendrá el usuario serán de sobre-escribir datos de su cuenta solo si cumple algunas condiciones tales como no puede modificar la cantidad de dinero que tiene a menos que haya depositado antes. Los clientes deberán cumplir ciertos pre-requisitos de la cooperativa, tales como por ejemplo tener cierta edad o cumplir con cierta nacionalidad. El banco también tendrá que tener implementado una serie de permisos, no en la base de datos sino en el lenguaje de programación que se utilice para manejar la base de datos.

- Permisos de ADMIN: Gerentes y las cabecillas de la corporación pueden tener acceso a modificar cualquier cosa en la base de datos, cualquier atributo de cualquier entidad.
- Permisos de USO: servicio al cliente podrá resolver gestiones como las de modificar la cantidad de dinero que posee un usuario en su cuenta, por supuesto después que el cliente haya depositado una cantidad de dinero.
- Permisos de MANEJO: el manejo de cuenta simplemente se le deja a los usuarios, ellos pueden pedir

Módulos:

Se implementará usando un lenguaje de programación que tenga soporte para hacer operaciones en Oracle Cloud Database. Los módulos en los que estará dividida serán básicamente tres acciones, la primera es crear nuevas instancias dinámicamente en la base de datos, tales como crear nuevos clientes con nuevas cuentas de ahorro, crear objetos de tipo préstamo, etc. La segunda sería eliminar las entidades creadas. La tercera sería poder escribir información que esté almacenada en los objetos de la base de datos. Además implementar las condiciones que hacen que todo funcione correctamente, por ejemplo no transferir de un cliente a otro si un cliente no tiene fondos, o no darle préstamos a alguien que pida un monto más grande de cinco veces su saldo promedio de los últimos seis meses.