





# INFORME BASE DE DATOS

SQUAD 2 - ALKEMY

---



# Informe Base de Datos - Squad 2 Alkemy

---

## 1. Descripción General

La base de datos corresponde a un sistema de billetera virtual, diseñado para gestionar usuarios, sus cuentas y las transacciones realizadas. El modelo se compone de cuatro tablas principales:

- **rol**: Define los tipos de usuarios dentro del sistema (por ejemplo, *administrador* o *cliente*).
- **usuario**: Almacena información personal de los usuarios y su rol correspondiente.
- **cuenta**: Registra las cuentas asociadas a los usuarios, incluyendo saldo, moneda y fecha de creación.
- **transaccion**: Guarda el historial de movimientos, como transferencias entre cuentas.

Estas entidades se relacionan entre sí mediante claves foráneas que garantizan la integridad referencial de los datos.

## 2. Estructura de Tablas y Relaciones

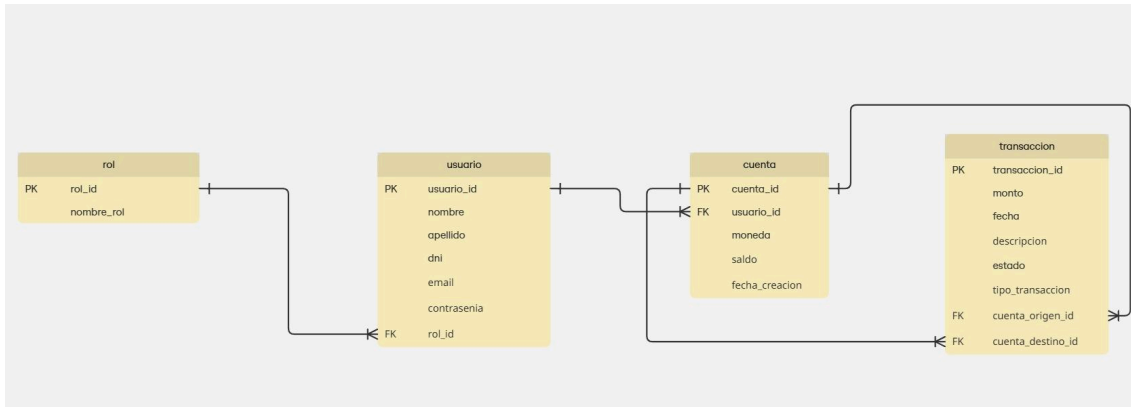
- **Claves primarias**: Todas las tablas poseen una clave primaria (ID) con valor autoincremental, que garantiza la unicidad de los registros.
- **Claves foráneas**:
  - **usuario.rol\_id** referencia a **rol.rol\_id**.
  - **cuenta.usuario\_id** referencia a **usuario.usuario\_id**.
  - **transaccion.cuenta\_origen\_id** y **cuenta\_destino\_id** referencian a **cuenta.cuenta\_id**.

### 2.1 Tabla: rol

Contiene los tipos de usuario del sistema. Incluye un campo **rol\_id** como clave primaria autoincremental y un campo de descripción (**nombre\_rol**).

### 2.2 Tabla: usuario

Almacena los datos personales de los usuarios, incluyendo *nombre*, *apellido*, *DNI*, *email* y *contraseña*. Cada usuario está vinculado a un rol mediante la clave foránea **rol\_id**.



### 2.3 Tabla: cuenta

Representa las cuentas asociadas a los usuarios. Contiene información como *tipo de moneda, saldo disponible y fecha de creación*. Se vincula a la tabla usuario a través de la clave **usuario\_id**.

### 2.4 Tabla: transaccion

Registra las transacciones realizadas entre cuentas. Cada transacción incluye el *monto, fecha, descripción, estado (pendiente, aprobado, cancelado), tipo de transacción y las cuentas de origen y destino (cuenta\_origen\_id, cuenta\_destino\_id)*

## 3. Datos de Ejemplo Insertados

Para validar la funcionalidad del sistema, se insertaron los siguientes datos:

- **Roles:** administrador, cliente
- **Usuarios:** Juan López, María Sánchez, Carlos García
- **Cuentas:** asociadas a Juan y María
- **Transacciones:** movimientos entre las cuentas de Juan y María

## 4. Consultas SQL Realizadas

Durante el proceso de validación, se realizaron las siguientes consultas SQL:

1. Selección de todos los registros de cada tabla.
2. Selección de usuarios con rol '*cliente*'.
3. Identificación de usuarios que realizaron transferencias.
4. Cálculo del saldo final estimado de Juan López.

5. Inserción de un nuevo usuario con rol '*administrador*'.
6. Actualización de contraseña para el usuario con ID = 4.
7. Eliminación del usuario con ID = 3.

## **5. Conclusión**

La base de datos fue correctamente diseñada e implementada, cumpliendo con los requisitos funcionales de un sistema de billetera virtual. La estructura relacional garantiza integridad de datos, escalabilidad y permite realizar operaciones clave como la gestión de usuarios, cuentas y transacciones. Las consultas ejecutadas validan la operatividad del sistema sin necesidad de realizar modificaciones estructurales adicionales.