**Explicación sobre el Diagrama de la Base de Datos**

El modelo de datos está diseñado para cubrir las operaciones y entidades principales de un banco. A continuación, se describe cada tabla y sus relaciones:

**1. Clientes**

Propósito: Almacena la información personal de los clientes del banco.

Campos principales: IdCliente, Nombres, Apellidos, NombreCompleto, DPI, Teléfono, CorreoPersonal.

Relaciones: Cada cliente puede tener varias cuentas asociadas.

**2. Empleados**

Propósito: Guarda los datos de los empleados del banco.

Campos principales: IdEmpleado, Nombres, Apellidos, DPI, Teléfono, CorreoPersonal, FK\_IdSucursal.

Relaciones: Los empleados pueden estar asociados a una sucursal y pueden estar relacionados con pagos realizados o autorizados.

**3. Usuarios**

Propósito: Representa las credenciales de acceso al sistema (login).

Campos principales: IdUsuario, Usuario, Contraseña, Token, FK\_IdRol, FK\_IdPersona (referencia a Cliente o Empleado).

Relaciones: Un usuario puede pertenecer a un cliente o a un empleado y tiene un rol asignado.

**4. TipoRol**

Propósito: Define los diferentes roles del sistema (Empleado, Cliente).

Campos principales: IdRol, TipoRol.

Relaciones: Los usuarios se asocian a un rol para determinar permisos y accesos.

**5. Cuentas**

Propósito: Almacena las cuentas bancarias de los clientes.

Campos principales: IdCuenta, TipoCuenta, NumeroCuenta, Saldo, FK\_IdCliente, FK\_IdSucursal, FechaApertura.

Relaciones: Cada cuenta pertenece a un cliente y a una sucursal.

**6. Sucursales**

Propósito: Guarda la información de las sucursales del banco.

Campos principales: IdSucursal, Nombre, Dirección, Teléfono, Correo.

Relaciones: Las cuentas y empleados pueden estar vinculados a una sucursal.

**7. Pagos**

Propósito: Registra los movimientos de pagos, transferencias y depósitos.

Campos principales: IdPago, TipoPago, Monto, Descripción, FK\_NumeroCuentaOrigen, FK\_NumeroCuentaDestino, FK\_IdEmpleado, FK\_IdCliente, FechaPago.

Relaciones: Un pago puede estar asociado a cuentas de origen y destino, así como a un empleado que lo realizó o autorizó.

**8. Bitácoras**

Propósito: Lleva el registro de acciones importantes realizadas en el sistema (auditoría).

Campos principales: IdBitacora, Acción, Descripción, FechaBitacora.

Relaciones: Permite rastrear eventos clave para fines de auditoría y seguridad.

**Explicación sobre el Proyecto desarrollado**

**Seguridad:** La autenticación se realiza mediante JWT, protegiendo los endpoints sensibles.

**Auditoría:** Todas las acciones importantes quedan registradas en la bitácora.

**Escalabilidad:** El modelo permite agregar nuevos tipos de roles, cuentas, sucursales y funcionalidades fácilmente.

**Separación de responsabilidades:** Cada entidad y relación está bien definida para evitar redundancias y asegurar la integridad de los datos.

**Ejemplo de Flujo de Uso:**

* **Login:** Un usuario (cliente o empleado) inicia sesión y recibe un token JWT.
* **Operaciones:** El usuario puede consultar cuentas, realizar pagos o transferencias.
* **Bitácora:** Cada acción relevante es registrada para fines de auditoría.
* **Consultas:** Se pueden consultar clientes, cuentas, pagos y movimientos según los permisos del usuario.

**Notas adicionales:**

Se ha implementado la mayoría de lógica para que funcione la API Rest, como bien detalle anteriormente, he implementado JWT para la seguridad de uso de los endpoints, también he creado una colección en Postman, solo para hacer las pruebas necesarias. Me consta, que bien puede faltar más validaciones o incluso atributos según los modelos a como yo los he implementado para crear una API mucho más robusta y segura. De mi parte, eso es todo, quedo atento a cualquier retroalimentación, muchas gracias.