



Documentación del Proyecto Final

API REST SISTEMA BANCARIO

Introducción.

Este proyecto es una API REST implementada en Java Spring Boot para gestionar operaciones bancarias básicas en un sistema de banco. La API permite realizar operaciones de clientes, cuentas, depósitos, retiros y transferencias.

Información de interés.

- **API Online:** A cada URL de los endpoints, le agregué su correspondiente enlace completo subido en Railway.app. Se me ocurrió esta opción para presentar una más simple accesibilidad al proyecto. Por lo que probé, todas las entity se guardan en la memoria del servidor, es 100% funcional.
- **BanelcoService:** En caso que la “cuentaDestino” no esté en el banco se invoca al servicio y si el número de “cuentaDestino” no es un número divisible por “2”, no realiza la transferencia.
- **VerazService:** Simulo una invocación a un servicio de Veraz. En caso de que el DNI del cliente sea un numero primo, lo detecta como deudor y lanza una excepción.
- **InterpolService:** Simulo una invocación a un servicio de Interpol. Si el DNI es igual a 26456437, lanza una excepción que tiene pedido de captura internacional.
- Agregué una excepción que salta cuando el endpoint es incorrecto.
- Agregue el versionado de la API en los endpoint.

1. Controlador: Cliente

1.1. Crear Cliente

Método: POST

URL: /api/v1/cliente

URL ONLINE: <https://laboratorioiiiprojectofinal-production.up.railway.app/api/v1/cliente>

Descripción: Crea un nuevo cliente en el sistema. Donde “tipoPersona” tiene que ser “F”, para personas físicas, o “J”, para personas jurídicas.



Request Body: json

```
{  
  "nombre": "string",  
  "apellido": "string",  
  "dni": 12345678,  
  "fechaNacimiento": "YYYY-MM-DD",  
  "tipoPersona": "string"  
}
```

Respuesta:

Body: json

```
{  
  "nombre": "Juan Jose",  
  "apellido": "Chaparro",  
  "dni": 37389808,  
  "fechaNacimiento": "1993-01-16",  
  "tipoPersona": "F",  
  "banco": "Banco UTN - TUP",  
  "fechaAlta": "2024-09-03",  
  "cuentas": [],  
  "edad": 31  
}
```

1.2. Obtener Cliente

Método: GET

URL: /api/v1/cliente/{dni}

URL ONLINE: <https://laboratorioiiiiprojectofinal-production.up.railway.app/api/v1/cliente/{dniCliente}>

Descripción: Obtiene la información de un cliente basado en su DNI.

Parámetros de Ruta:

- **dni (long):** El DNI del cliente a obtener.



Respuesta:

Body:

```
{
  "nombre": "Juan Jose",
  "apellido": "Chaparro",
  "dni": 37389808,
  "fechaNacimiento": "1993-01-16",
  "tipoPersona": "F",
  "banco": "Banco UTN - TUP",
  "fechaAlta": "2024-09-03",
  "cuentas": [],
  "edad": 31
}
```

2. Controlador: Cuenta

2.1. Crear Cuenta

Método: POST

URL: /api/v1/cuenta

URL ONLINE: <https://laboratorioiiiiprojectofinal-production.up.railway.app/api/v1/cuenta>

Descripción: Crea una nueva cuenta en el sistema. Donde “tipoCuenta” tiene que ser “CC”, para cuenta corriente, o “CA”, para caja de ahorro, y “moneda” tiene que ser “P”, para pesos, o “D”, para dólares.
Cada Cliente puede tener una caja de ahorro en pesos, una caja de ahorro en dólares y una cuenta corriente en pesos.

Request Body: json

```
{
  "tipoCuenta": "string",
  "dni": 12345678,
  "moneda": "string"
}
```



Respuesta:

Body:

```
{
  "numeroCuenta": 37823283,
  "fechaCreacion": "2024-09-03T21:38:20.5782707",
  "balance": 0.0,
  "tipoCuenta": "CUENTA_CORRIENTE",
  "moneda": "PESOS"
}
```

2.2 Mostrar Cuenta

Método: GET

URL: /api/v1/cuenta/{idCuenta}

URL ONLINE: <https://laboratorioiiiiprojectofinal-production.up.railway.app/api/v1/cuenta/{idCuenta}>

Descripción: Obtiene la información de una cuenta basada en su ID.

Parámetros de Ruta:

- **idCuenta** (long): El ID de la cuenta a obtener.

Respuesta:

Body: json

```
{
  "numeroCuenta": 37823283,
  "fechaCreacion": "2024-09-03T21:38:20.5782707",
  "balance": 0.0,
  "tipoCuenta": "CUENTA_CORRIENTE",
  "moneda": "PESOS"
}
```



2.3. Obtener Cuentas de un Cliente

Método: GET

URL: /api/v1/cuenta/{idcliente}/cuentas

URL ONLINE: <https://laboratorioiiiiprojectofinal-production.up.railway.app/api/v1/cuenta/{idcliente}/cuentas>

Descripción: Obtiene todas las cuentas asociadas a un cliente basado en su ID.

Parámetros de Ruta:

- **idcliente** (long): El ID del cliente.

Respuesta:

Body: json

```
[
  {
    "numeroCuenta": 37823283,
    "fechaCreacion": "2024-09-03T21:38:20.5782707",
    "balance": 10000.0,
    "tipoCuenta": "CUENTA_CORRIENTE",
    "moneda": "PESOS"
  },
  ...
]
```

2.4. Obtener Movimientos de una Cuenta

Método: GET

URL: /api/v1/cuenta/{idCuenta}/movimientos

URL ONLINE: <https://laboratorioiiiiprojectofinal-production.up.railway.app/api/v1/cuenta/{idCuenta}/movimientos>

Descripción: Obtiene los movimientos de una cuenta basada en su ID.



Parámetros de Ruta:

- **idCuenta** (long): El ID de la cuenta.

Respuesta:

Body: json

```
{
  "numeroCuenta": 37823283,
  "movimientosCuenta": [
    {
      "fecha": "2024-09-03T21:41:49.0654037",
      "tipo": "DEPOSITO",
      "monto": 10000.0,
      "descripcion": "Depósito en cuenta: 37823283",
      "numMovimiento": 625236945
    },
    ...
  ]
}
```

3 Controlador: Depósitos

3.1. Realizar Depósito

Método: POST

URL: /api/v1/deposito

URL ONLINE: <https://laboratorioiiiiprojectofinal-production.up.railway.app/api/v1/deposito>

Descripción: Realiza un depósito en una cuenta especificada. Donde “moneda” tiene que ser “P”, para pesos, o “D”, para dólares, y tiene que coincidir con la cuenta.

Request Body: json

```
{
  "cuenta": 12345678,
  "monto": 1000.00,
  "moneda": "P"
}
```



Respuesta:

Body: json

```
{
  "estado": "EXITOSA",
  "mensaje": "Deposito Exitoso. Número de movimiento: 637952245. Realizado el 2024-09-03T21:40:36.561641900"
}
```

4 Controlador: Retiros

4.1. Realizar Retiro

Método: POST

URL: /api/v1/retiro

URL ONLINE: <https://laboratorioiiiprojectofinal-production.up.railway.app/api/v1/retiro>

Descripción: Realiza un retiro de una cuenta especificada. Donde “moneda” tiene que ser “P”, para pesos, o “D”, para dólares, y tiene que coincidir con la cuenta.

Request Body: json

```
{
  "cuenta": 12345678,
  "monto": 500.00,
  "moneda": "P"
}
```

Respuesta:

Body: json

```
{
  "estado": "EXITOSA",
  "mensaje": "Retiro Exitoso. Número de movimiento: 952369607. Realizado el 2024-09-03T21:43:18.685166900"
}
```



5 Controlador: Transferencia

5.1. Realizar Transferencia

Método: POST

URL: /api/v1/transferencia

URL ONLINE: <https://laboratorioiiiiprojectofinal-production.up.railway.app/api/v1/transferencia>

Descripción: Realiza una transferencia de fondos entre dos cuentas. Donde “moneda” tiene que ser “P”, para pesos, o “D”, para dólares, y tiene que coincidir entre las dos cuentas.

Request Body: json

```
{  
  "cuentaOrigen": 12345678,  
  "cuentaDestino": 87654321,  
  "monto": 500.00,  
  "moneda": "P"  
}
```

Respuesta:

Body: json

```
{  
  "estado": "EXITOSA",  
  "mensaje": "Transferencia Exitosa. Número de transferencia: 874812548.  
Realizado el 2024-09-03T21:43:44.627562700"  
}
```