*Script de Base de datos:*

**//Creacion del esquema**

create schema lifebank\_db;

**//Creacion de tabla para el manejo de usuarios**

create table lifebank\_db.usuario (

usr\_id serial not null,

usr\_usuario VARCHAR(25) not null,

usr\_password VARCHAR(500) not null,

usr\_nombre VARCHAR(50) not null,

usr\_apellido VARCHAR(50) not null,

usr\_correo VARCHAR(50) not null,

usr\_estado CHAR (1) not null,

usr\_conteo\_fallido int DEFAULT 0,

usr\_CREATED\_BY varchar(25) NOT NULL DEFAULT USER,

usr\_CREATED\_DATE timestamp NOT NULL DEFAULT NOW(),

usr\_MODIFIED\_BY varchar(25) NULL,

usr\_MODIFIED\_DATE timestamp NULL,

constraint PK\_USUARIO primary key (usr\_id)

);

COMMENT ON TABLE lifebank\_db.usuario IS 'Informacion de las crendenciales de los usuarios' ;

INSERT INTO lifebank\_db.usuario (usr\_id, usr\_usuario, usr\_password, usr\_nombre, usr\_apellido, usr\_correo,urs\_estado,urs\_conteo\_fallido, usr\_created\_by, usr\_created\_date, usr\_modified\_by, usr\_modified\_date) VALUES(1, 'wcenteno', '04e85978d5ed597ecfaf4ed222aaab05ca4c32b2898532ce0fe7cdf2cfc1dd5f502e40fdf0a205f47e22d422e5146b2f893c5817ef5e3da0ace6c72941f84040', 'Wilfredo', 'Centeno', 'wilfredo@gmail.com','A',0, 'postgres', '2019-05-09 17:16:48.878', NULL, NULL);

INSERT INTO lifebank\_db.usuario (usr\_id, usr\_usuario, usr\_password, usr\_nombre, usr\_apellido, usr\_correo,urs\_estado,urs\_conteo\_fallido, usr\_created\_by, usr\_created\_date, usr\_modified\_by, usr\_modified\_date) VALUES(2, 'eflores', '04e85978d5ed597ecfaf4ed222aaab05ca4c32b2898532ce0fe7cdf2cfc1dd5f502e40fdf0a205f47e22d422e5146b2f893c5817ef5e3da0ace6c72941f84040', 'Emilio', 'Flores', 'emilio@gmail.com','A',0, 'postgres', '2019-05-09 17:16:48.878', NULL, NULL);

**//Creacion de tabla para el manejo de los token se sesión de los usuarios**

create table lifebank\_db.usuario\_token(

tkn\_id serial not null,

tkn\_token VARCHAR(500) not null,

tkn\_usr\_id int not null,

tkn\_date\_expiration timestamp NOT NULL,

tkn\_created\_date timestamp not null default now(),

constraint PK\_USUARIO\_TOKEN primary key (tkn\_id),

constraint FK\_USUARIO foreign key(tkn\_usr\_id) references lifebank\_db.usuario (usr\_id)

);

COMMENT ON TABLE lifebank\_db.usuario\_token IS 'Informacion de las sessiones de los usuarios' ;

**//Creacion de tabla para el manejo de los tipos de productos que posee el banco (Cuenta Bancaria, Prestamos y Tarjetas de créditos)**

create table lifebank\_db.tipo\_producto(

tip\_id serial not null,

tip\_nombre varchar(100) not null,

constraint PK\_TIPO\_CUENTAS primary key (tip\_id)

);

COMMENT ON TABLE lifebank\_db.tipo\_producto IS 'Tipo de cuentas en bancos';

INSERT into lifebank\_db.tipo\_producto(tip\_id,tip\_nombre) values (1,'Cuentas Personales Ahorro');

INSERT into lifebank\_db.tipo\_producto(tip\_id,tip\_nombre) values (2,'Tarjeta de Creditos');

INSERT into lifebank\_db.tipo\_producto(tip\_id,tip\_nombre) values (3,'Prestamos Personal');

**//Creacion de tabla para el manejo de los cuentas que posee un cliente**

create table lifebank\_db.cuentas\_bancarias(

cub\_id varchar(50) not null,

cub\_nombre varchar(100) not null,

cub\_tip\_id int not null,

cub\_usr\_id int not null,

cub\_saldo decimal(18,2),

cub\_estado char(1) default 'A',

cub\_created\_date timestamp DEFAULT now(),

constraint PK\_CUENTA primary key (cub\_id),

constraint FK\_CUE\_TIPO\_CUENTA foreign key(cub\_tip\_id) references lifebank\_db.tipo\_producto (tip\_id),

constraint FK\_CUE\_USUARIO foreign key(cub\_usr\_id) references lifebank\_db.usuario (usr\_id)

);

insert into lifebank\_db.cuentas\_bancarias (cub\_id,cub\_nombre,cub\_tip\_id,cub\_usr\_id,cub\_saldo) values ('001','Cuenta de Ahorro 1',1,1,200);

insert into lifebank\_db.cuentas\_bancarias (cub\_id,cub\_nombre,cub\_tip\_id,cub\_usr\_id,cub\_saldo) values ('002','Cuenta de Ahorro 2',1,1,500);

insert into lifebank\_db.cuentas\_bancarias (cub\_id,cub\_nombre,cub\_tip\_id,cub\_usr\_id,cub\_saldo) values ('003','Cuenta de Ahorro 1',1,2,500);

**//Creacion de tabla para el manejo de los prestamos que posee un usuario**

create table lifebank\_db.prestamos\_bancarias(

prb\_id varchar(50) not null,

prb\_nombre varchar(100) not null,

prb\_tip\_id int not null,

prb\_usr\_id int not null,

prb\_saldo decimal(18,2),

prb\_interes\_mensual decimal(18,2),

prb\_total\_interes decimal(18,2),

prb\_periodo\_meses int not null,

prb\_estado char(1) default 'A',

prb\_fecha\_inicio timestamp,

prb\_created\_date timestamp DEFAULT now(),

constraint PK\_PREST\_BANCA primary key (prb\_id),

constraint FK\_\_PRB\_TIPO\_CUENTA foreign key(prb\_tip\_id) references lifebank\_db.tipo\_producto (tip\_id),

constraint FK\_PRB\_USUARIO foreign key(prb\_usr\_id) references lifebank\_db.usuario (usr\_id)

);

insert into lifebank\_db.prestamos\_bancarias (prb\_id,prb\_nombre,prb\_tip\_id,prb\_usr\_id,prb\_saldo,prb\_periodo\_meses,prb\_interes\_mensual,prb\_total\_interes,prb\_fecha\_inicio) values ('001','Prestamos Personal 1',3,1,1500,36,2.5,550,'2019-05-09');

**//Creacion de tabla para el manejo de las tarjetas de créditos que posee un usuario**

create table lifebank\_db.tarjeta\_creditos(

tc\_id varchar(50) not null,

tc\_nombre varchar(100) not null,

tc\_tip\_id int not null,

tc\_usr\_id int not null,

tc\_limite decimal(18,2),

tc\_saldo decimal(18,2),

tc\_costo\_membresia int not null,

tc\_interes\_mensual decimal(18,2),

tc\_total\_interes decimal(18,2),

tc\_estado char(1) DEFAULT 'A',

tc\_fecha\_corte int,

tc\_fecha\_pago int,

tc\_fecha\_cancelacion timestamp,

tc\_created\_date timestamp DEFAULT now(),

constraint PK\_TARJETA\_CREDITO primary key (tc\_id),

constraint FK\_TC\_TIPO\_CUENTA foreign key(tc\_tip\_id) references lifebank\_db.tipo\_producto (tip\_id),

constraint FK\_TC\_USUARIO foreign key(tc\_usr\_id) references lifebank\_db.usuario (usr\_id)

);

insert into lifebank\_db.tarjeta\_creditos(tc\_id,tc\_nombre,tc\_tip\_id,tc\_usr\_id,tc\_limite,tc\_saldo,tc\_costo\_membresia,tc\_interes\_mensual,tc\_total\_interes,tc\_fecha\_corte,tc\_fecha\_pago) values ('001','Tarjeta de Cretito clsica',2,1,700,700,25,3.5,0,4,15);

**//Creacion de tabla para el manejo de las tipo de transacciones(Abono o Cargo)**

create table lifebank\_db.tipo\_transaccion(

trs\_id serial not null,

trs\_nombre varchar(50) not null,

constraint PK\_TIPO\_TRANS primary key (trs\_id)

);

insert into lifebank\_db.tipo\_transaccion (trs\_id, trs\_nombre) values(1,'Abono');

insert into lifebank\_db.tipo\_transaccion (trs\_id, trs\_nombre) values(2,'Cargo');

**//Creacion de tabla para el manejo de las transacciones**

create table lifebank\_db.transacciones(

tra\_id serial not null,

tra\_id\_cuenta varchar(50) not null,

tra\_cantidad decimal(18,2) not null,

tra\_tipo\_cuenta int not null,

tra\_fecha timestamp not null,

tra\_tipo int not null,

tra\_descripcion varchar(100) not null,

tra\_referencia varchar(50) not null,

tra\_fecha\_creacion timestamp default now(),

constraint PK\_TRANS primary key (tra\_id)

);

**//Creacion de tabla para el manejo de los beneficiarios que posee un usuario.**

create table lifebank\_db.cuentas\_beneficiarios(

ben\_id serial not null,

ben\_num\_cuenta varchar(50) not null,

ben\_nombre varchar(100) not null,

ben\_correo varchar(50) not null,

ben\_tip\_cuenta int not null,

ben\_usr\_id int not null,

constraint PK\_BENEFI primary key (ben\_id),

constraint FK\_BEN\_TIPO\_CUENTA foreign key(ben\_tip\_cuenta) references lifebank\_db.tipo\_producto (tip\_id),

constraint FK\_BEN\_USUARIO foreign key(ben\_usr\_id) references lifebank\_db.usuario (usr\_id)

);

*Repositorio de los Servicio:*

<https://github.com/wcenteno2019>

* Servicio Authentication: **lfb-authentication-svc**
* Servicio Integrador para la comunicacion entre sitio web y el procesamiento: **lfb-integrator-svc**
* Servicio transacional para realización de los diferentes transacciones: **lfb-transacional-svc**

Repositorio: <https://github.com/wcenteno2019/lfb-authentication-svc>

**Método de Validación de Usuario*.***

|  |  |  |  |
| --- | --- | --- | --- |
| Servicio**: lfb-authentication-svc** | | | ITEM-1 El servicio tendrá la lógica para validar los usuarios del sistema, aplicando el método de encriptación SHA-512 (base64(user)+base64(pass), y le retornara si son validar un jwt para que pueda transaccionar en el sitio. |
| URL servicio: | | | **http://localhost:9092/lfb/authentication/validar-user** |
| Metodo | | | POST |
| Entrada: Donde user es el usuario del cliente que desea ingresar y password la contraseña que tiene almacenada en la base de datos. | | | {  "user":"wcenteno",  "password":"Emilio1234"  } |
| Salida: | | | |
| **Resultado** | **Header** | **Body:** | |
| Exitoso | HTTP 200 | | {  "tkn": "eyJhbGciOiJIUzUxMiJ9.eyJuYmYiOjE1NTc2Nzc1NDgsImlkQ2xpZW50ZSI6MSwiaXAiOiIwOjA6MDowOjA6MDowOjEiLCJmZWNoYUV4cGlyYWNpb24iOnsibW9udGgiOiJNQVkiLCJ5ZWFyIjoyMDE5LCJkYXlPZk1vbnRoIjoxMiwiZGF5T2ZXZWVrIjoiU1VOREFZIiwiZGF5T2ZZZWFyIjoxMzIsImhvdXIiOjEwLCJtaW51dGUiOjQyLCJuYW5vIjoxNDMwMDAwMDAsInNlY29uZCI6MjgsIm1vbnRoVmFsdWUiOjUsImNocm9ub2xvZ3kiOnsiaWQiOiJJU08iLCJjYWxlbmRhclR5cGUiOiJpc284NjAxIn19fQ.0-WO0okP69WCcfgsmYQ2A4j3iOSk70CUo9ltnjZPO3nRfLmvEyVaz6kko6xeuqGh-7XgWnFPfqg3msbHsL831g"  } |
| Error en credenciales | HTTP 400 | | Null |
| Error | HTTP 401 | | Null |

Repositorio: <https://github.com/wcenteno2019/lfb-integrator-svc>

**Método devuelve la cuenta que posee un Cliente***.*

|  |  |  |  |
| --- | --- | --- | --- |
| Servicio**: lfb-integrator-svc** | | | ITEM-3 El servicio devolverá todas las cuentas que posee un cliente (Cuenta Bancaria, Prestamos y Tarjeta de Credito) |
| URL servicio: | | | http://localhost:90/lfb/integrator/getAllAccount/{**idUsuario**} |
| Metodo | | | GET |
| Entrada: Parámetros: **idUsuario**. Y Header (**"authorization"**) | | | [http://localhost:90/lfb/integrator/getAllAccount/{**1**](http://localhost:90/lfb/integrator/getAllAccount/%7b1)  **Header:**  **Authorization = “**eyJhbGciOiJIUzUxMiJ9.eyJuYmYiOjE1NTc2Nzc1NDgsImlkQ2xpZW50ZSI6MSwiaXAiOiIwOjA6MDowOjA6MDowOjEiLCJmZWNoYUV4cGlyYWNpb24iOnsibW9udGgiOiJNQVkiLCJ5ZWFyIjoyMDE5LCJkYXlPZk1vbnRoIjoxMiwiZGF5T2ZXZWVrIjoiU1VOREFZIiwiZGF5T2ZZZWFyIjoxMzIsImhvdXIiOjEwLCJtaW51dGUiOjQyLCJuYW5vIjoxNDMwMDAwMDAsInNlY29uZCI6MjgsIm1vbnRoVmFsdWUiOjUsImNocm9ub2xvZ3kiOnsiaWQiOiJJU08iLCJjYWxlbmRhclR5cGUiOiJpc284NjAxIn19fQ.0-WO0okP69WCcfgsmYQ2A4j3iOSk70CUo9ltnjZPO3nRfLmvEyVaz6kko6xeuqGh-7XgWnFPfqg3msbHsL831g” |
| Salida: | | | |
| **Resultado** | **Header** | **Body:** | |
| Exitoso | HTTP 200 | | {  "accounts": {  "loan": [  {  "id": "1",  "name": "Prestamos Personal 1"  }  ],  "creditCard": [  {  "id": "1",  "name": "Tarjeta de Cretito clsica"  }  ],  "personal": [  {  "id": "1",  "name": "Cuenta de Ahorro 1"  },  {  "id": "2",  "name": "Cuenta de Ahorro 2"  }  ]  }  } |
| Error | HTTP 400 | | Null |

**Método para realizar transferencia entre cuentas**

|  |  |  |  |
| --- | --- | --- | --- |
| Servicio**: lfb-integrator-svc** | | | ITEM-5 El servicio realizara la transferencia entre cuentas propias |
| URL servicio: | | | http://localhost:90/lfb/integrator/transfer |
| Metodo | | | POST |
| Entrada: Header (**"authorization"**) | | | [Body:](http://localhost:90/lfb/integrator/getAllAccount/%7b1) {  "cuentaOrigen":"002",  "cuentaDestino":"001",  "monto": 6.50,  "idUsuario":1  }  **Header:**  **Authorization = “**eyJhbGciOiJIUzUxMiJ9.eyJuYmYiOjE1NTc2Nzc1NDgsImlkQ2xpZW50ZSI6MSwiaXAiOiIwOjA6MDowOjA6MDowOjEiLCJmZWNoYUV4cGlyYWNpb24iOnsibW9udGgiOiJNQVkiLCJ5ZWFyIjoyMDE5LCJkYXlPZk1vbnRoIjoxMiwiZGF5T2ZXZWVrIjoiU1VOREFZIiwiZGF5T2ZZZWFyIjoxMzIsImhvdXIiOjEwLCJtaW51dGUiOjQyLCJuYW5vIjoxNDMwMDAwMDAsInNlY29uZCI6MjgsIm1vbnRoVmFsdWUiOjUsImNocm9ub2xvZ3kiOnsiaWQiOiJJU08iLCJjYWxlbmRhclR5cGUiOiJpc284NjAxIn19fQ.0-WO0okP69WCcfgsmYQ2A4j3iOSk70CUo9ltnjZPO3nRfLmvEyVaz6kko6xeuqGh-7XgWnFPfqg3msbHsL831g” |
| Salida: | | | |
| **Resultado** | **Header** | **Body:** | |
| Exitoso | HTTP 200 | | {  "numAutorizacion": "LFB345549546"  } |
| Error Cuentas no son del cliente enviado | HTTP 404 | | Null |
| Saldo insuficiente en cuenta origen | HTTP 409 | | Null |
| Error General | HTTP 500 | | Null |

**Método para realizar pago de tarjeta propio**

|  |  |  |  |
| --- | --- | --- | --- |
| Servicio**: lfb-integrator-svc** | | | ITEM-6 El servicio realizara el pago de la tarjeta de crédito con una cuentas propias |
| URL servicio: | | | http://localhost:90/lfb/integrator/ pago-tarjeta |
| Metodo | | | POST |
| Entrada: Header (**"authorization"**) | | | [Body:](http://localhost:90/lfb/integrator/getAllAccount/%7b1) {  "cuentaTarjeta":"001",  "cuentaPersonal":"001",  "monto": 21.54,  "idUsuario":1  }**Header:**  **Authorization = “**eyJhbGciOiJIUzUxMiJ9.eyJuYmYiOjE1NTc2Nzc1NDgsImlkQ2xpZW50ZSI6MSwiaXAiOiIwOjA6MDowOjA6MDowOjEiLCJmZWNoYUV4cGlyYWNpb24iOnsibW9udGgiOiJNQVkiLCJ5ZWFyIjoyMDE5LCJkYXlPZk1vbnRoIjoxMiwiZGF5T2ZXZWVrIjoiU1VOREFZIiwiZGF5T2ZZZWFyIjoxMzIsImhvdXIiOjEwLCJtaW51dGUiOjQyLCJuYW5vIjoxNDMwMDAwMDAsInNlY29uZCI6MjgsIm1vbnRoVmFsdWUiOjUsImNocm9ub2xvZ3kiOnsiaWQiOiJJU08iLCJjYWxlbmRhclR5cGUiOiJpc284NjAxIn19fQ.0-WO0okP69WCcfgsmYQ2A4j3iOSk70CUo9ltnjZPO3nRfLmvEyVaz6kko6xeuqGh-7XgWnFPfqg3msbHsL831g” |
| Salida: | | | |
| **Resultado** | **Header** | **Body:** | |
| Exitoso | HTTP 200 | | {  "numAutorizacion": " LFB270493098"  } |
| Error Cuentas no son del cliente enviado | HTTP 404 | | Null |
| Saldo insuficiente en cuenta origen | HTTP 409 | | Null |
| Error General | HTTP 500 | | Null |

**Método para realizar pago de préstamo propio**

|  |  |  |  |
| --- | --- | --- | --- |
| Servicio**: lfb-integrator-svc** | | | ITEM-7 El servicio realizara la el pago del préstamo con un cuenta propia |
| URL servicio: | | | http://localhost:90/lfb/integrator/ pago-prestamo |
| Metodo | | | POST |
| Entrada: Header (**"authorization"**) | | | [Body:](http://localhost:90/lfb/integrator/getAllAccount/%7b1) {  "cuentaTarjeta":"001",  "cuentaPersonal":"001",  "monto": 21.54,  "idUsuario":1  }**Header:**  **Authorization = “**eyJhbGciOiJIUzUxMiJ9.eyJuYmYiOjE1NTc2Nzc1NDgsImlkQ2xpZW50ZSI6MSwiaXAiOiIwOjA6MDowOjA6MDowOjEiLCJmZWNoYUV4cGlyYWNpb24iOnsibW9udGgiOiJNQVkiLCJ5ZWFyIjoyMDE5LCJkYXlPZk1vbnRoIjoxMiwiZGF5T2ZXZWVrIjoiU1VOREFZIiwiZGF5T2ZZZWFyIjoxMzIsImhvdXIiOjEwLCJtaW51dGUiOjQyLCJuYW5vIjoxNDMwMDAwMDAsInNlY29uZCI6MjgsIm1vbnRoVmFsdWUiOjUsImNocm9ub2xvZ3kiOnsiaWQiOiJJU08iLCJjYWxlbmRhclR5cGUiOiJpc284NjAxIn19fQ.0-WO0okP69WCcfgsmYQ2A4j3iOSk70CUo9ltnjZPO3nRfLmvEyVaz6kko6xeuqGh-7XgWnFPfqg3msbHsL831g” |
| Salida: | | | |
| **Resultado** | **Header** | **Body:** | |
| Exitoso | HTTP 200 | | {  "numAutorizacion": "LFB108196243"  } |
| Error Cuentas no son del cliente enviado | HTTP 404 | | Null |
| Saldo insuficiente en cuenta origen | HTTP 409 | | Null |
| Error General | HTTP 500 | | Null |

Repositorio: <https://github.com/wcenteno2019/lfb-transacional-svc>

**Método devuelve la cuenta que posee un Cliente***.*

|  |  |  |  |
| --- | --- | --- | --- |
| Servicio**: lfb-transacional-svc** | | | ITEM-3 El servicio devolverá todas las cuentas que posee un cliente (Cuenta Bancaria, Prestamos y Tarjeta de Credito) |
| URL servicio: | | | http://localhost:9091/lfb/transacciones/obtener-cuentas/{**idUsuario**} |
| Metodo | | | GET |
| Entrada: Parámetros: **idUsuario** | | | <http://localhost:9091/lfb/transacciones/obtener-cuentas/1> |
| Salida: | | | |
| **Resultado** | **Header** | **Body:** | |
| Exitoso | N/A | | {  "cuenta": [  {  "id": "001",  "name": "Cuenta de Ahorro 1",  "tipo": 1  },  {  "id": "002",  "name": "Cuenta de Ahorro 2",  "tipo": 1  },  {  "id": "001",  "name": "Prestamos Personal 1",  "tipo": 3  },  {  "id": "001",  "name": "Tarjeta de Cretito clsica",  "tipo": 2  }  ],  "status": 200,  "message": "SUCCES"  } |
| Cliente no posee cuentas | N/A | | {  "cuenta": [ ],  "status": 201,  "message": "NO DATA"  } |
| Error General | N/A | | {  "cuenta": [ ],  "status": 202,  "message": "NO DATA"  } |

**Método para realizar transferencia entre cuentas**

|  |  |  |  |
| --- | --- | --- | --- |
| Servicio**: lfb-transacional-svc** | | | ITEM-5 El servicio realizara la transferencia entre cuentas propias |
| URL servicio: | | | http://localhost:9091/lfb/transacciones/transfer |
| Metodo | | | POST |
| Entrada: | | | [Body:](http://localhost:90/lfb/integrator/getAllAccount/%7b1) {  "cuentaOrigen":"002",  "cuentaDestino":"001",  "monto": 6.50,  "idUsuario":1  } |
| Salida: | | | |
| **Resultado** | **Header** | **Body:** | |
| Exitoso | N/A | | {  "status": "200",  "message": "SUCESS",  "numAutotizacion": " LFB345549546"  } |
| Error Cuentas no son del cliente enviado | N/A | | {  "status": "404",  "message": "ERROR EN CUENTAS",  "numAutotizacion": ""  } |
| Saldo insuficiente en cuenta origen | N/A | | {  "status": "409",  "message": "SALDO INSUFICIENTE",  "numAutotizacion": ""  } |
| Error General | N/A | | {  "status": "500",  "message": "EROR",  "numAutotizacion": ""  } |

**Método para realizar pago de tarjeta propio**

|  |  |  |  |
| --- | --- | --- | --- |
| Servicio**: lfb-transacional-svc** | | | ITEM-6 El servicio realizara el pago de la tarjeta de crédito con una cuentas propias |
| URL servicio: | | | http://localhost:9091/lfb/transacciones/pago-tarjeta |
| Metodo | | | POST |
| Entrada: | | | [Body:](http://localhost:90/lfb/integrator/getAllAccount/%7b1) {  "cuentaTarjeta":"001",  "cuentaPersonal":"001",  "monto": 21.54,  "idUsuario":1  } |
| Salida: | | | |
| **Resultado** | **Header** | **Body:** | |
| Exitoso | N/A | | {  "status": "200",  "message": "SUCESS",  "numAutotizacion": " LFB91545395"  } |
| Error Cuentas no son del cliente enviado | N/A | | {  "status": "404",  "message": "ERROR EN CUENTAS",  "numAutotizacion": ""  } |
| Saldo insuficiente en cuenta origen | N/A | | {  "status": "409",  "message": "SALDO INSUFICIENTE",  "numAutotizacion": ""  } |
| Error General | N/A | | {  "status": "500",  "message": "EROR",  "numAutotizacion": ""  } |

**Método para realizar pago de préstamo propio**

|  |  |  |  |
| --- | --- | --- | --- |
| Servicio**: lfb-transacional-svc** | | | ITEM-7 El servicio realizara la el pago del préstamo con un cuenta propia |
| URL servicio: | | | http://localhost:9091/lfb/transacciones/pago-prestamo |
| Metodo | | | POST |
| Entrada: | | | [Body:](http://localhost:90/lfb/integrator/getAllAccount/%7b1) {  "cuentaTarjeta":"001",  "cuentaPersonal":"001",  "monto": 21.54,  "idUsuario":1  } |
| Salida: | | | |
| **Resultado** | **Header** | **Body:** | |
| Exitoso | N/A | | {  "status": "200",  "message": "SUCESS",  "numAutotizacion": " LFB108196243"  } |
| Error Cuentas no son del cliente enviado | N/A | | {  "status": "404",  "message": "ERROR EN CUENTAS",  "numAutotizacion": ""  } |
| Saldo insuficiente en cuenta origen | N/A | | {  "status": "409",  "message": "SALDO INSUFICIENTE",  "numAutotizacion": ""  } |
| Error General | N/A | | {  "status": "500",  "message": "EROR",  "numAutotizacion": ""  } |