

PROCEDIMIENTO DE INSTALACION

Versión 1.0

Contenido

Base de datos	3
Modelo Entidad Relación	7
Fuentes.....	7

Base de datos

Lo primero que hay que hacer es crear la base de datos en postgresSQL, las tablas y el usuario. Para crear la base de datos ejecutamos el siguiente DDL.

```
-- Database: "Banco_PagosOnline"

-- DROP DATABASE "Banco_PagosOnline";

CREATE DATABASE "Banco_PagosOnline"

WITH OWNER = postgres

    ENCODING = 'UTF8'

    TABLESPACE = pg_default

    LC_COLLATE = 'Spanish_Colombia.1252'

    LC_CTYPE = 'Spanish_Colombia.1252'

    CONNECTION LIMIT = -1;
```

Una vez creada la base de datos, se establece una conexión con ella y se ejecuta el script para cada una de las siguientes 3 tablas:

```
-- Table: cliente

-- DROP TABLE cliente;

CREATE TABLE cliente

(

    id serial NOT NULL, -- Identificador unico de un cliente

    nombre character varying(50) NOT NULL, -- Nombre completo del cliente

    direccion character varying(20), -- Dirección de la residencia del cliente

    telefono character varying(10), -- Telefono del cliente

    CONSTRAINT "ClientePK" PRIMARY KEY (id)

)

WITH (

    OIDS=FALSE
```

```

);

ALTER TABLE cliente

OWNER TO postgres;

COMMENT ON TABLE cliente

IS 'Representa a los clientes del banco';

COMMENT ON COLUMN cliente.id IS 'Identificador unico de un cliente';

COMMENT ON COLUMN cliente.nombre IS 'Nombre completo del cliente';

COMMENT ON COLUMN cliente.direccion IS 'Dirección de la residencia del cliente';

COMMENT ON COLUMN cliente.telefono IS 'Telefono del cliente';


-- Table: cuenta

-- DROP TABLE cuenta;

CREATE TABLE c
uenta
(
    id serial NOT NULL, -- Identificador unico de la cuenta
    numero character varying(20) NOT NULL, -- Numero de la cuenta bancaria de un cliente
    saldo integer NOT NULL DEFAULT 0, -- Cantidad de dinero que tiene la cuenta bancaria
    id_cliente integer NOT NULL, -- Relaciona un cliente con la cuenta
    CONSTRAINT "Cuenta_PK" PRIMARY KEY (id),
    CONSTRAINT "Cuenta_Cliente_FK" FOREIGN KEY (id_cliente)
        REFERENCES cliente (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE CASCADE
)

WITH (
    OIDS=FALSE

```

```

);

ALTER TABLE cuenta

OWNER TO postgres;

COMMENT ON TABLE cuenta

IS 'Representa una cuenta bancaria de un cliente del banco';

COMMENT ON COLUMN cuenta.id IS 'Identificador unico de la cuenta';

COMMENT ON COLUMN cuenta.numero IS 'Numero de la cuenta bancaria de un cliente';

COMMENT ON COLUMN cuenta.saldo IS 'Cantidad de dinero que tiene la cuenta bancaria';

COMMENT ON COLUMN cuenta.id_cliente IS 'Relaciona un cliente con la cuenta';


-- Table: movimiento

-- DROP TABLE movimiento;

CREATE TABLE movimiento

(

id serial NOT NULL, -- Identificador unico del movimiento

tipo integer NOT NULL, -- Indica el tipo de cuenta (1 = debito, 2 = credito)

fecha timestamp with time zone NOT NULL DEFAULT now(), -- Indica la fecha en la que fue
realizado el movimiento

valor integer NOT NULL, -- Indica el valor del movimiento

id_cuenta integer NOT NULL, -- Identificador de la cuenta a la que pertenece este movimiento

saldo integer NOT NULL,

CONSTRAINT "Movimiento_PK" PRIMARY KEY (id),

CONSTRAINT "Movimiento_Cuenta_FK" FOREIGN KEY (id_cuenta)

REFERENCES cuenta (id) MATCH SIMPLE

ON UPDATE NO ACTION ON DELETE CASCADE

)

```

```

WITH (

    OIDS=FALSE

);

ALTER TABLE movimiento

    OWNER TO postgres;

COMMENT ON TABLE movimiento

    IS 'Representa los movimientos que ha tenido una cuenta de un cliente';

COMMENT ON COLUMN movimiento.id IS 'Identificador unico del movimiento';

COMMENT ON COLUMN movimiento.tipo IS 'Indica el tipo de cuenta (1 = debito, 2 = credito)';

COMMENT ON COLUMN movimiento.fecha IS 'Indica la fecha en la que fue realizado el
movimiento';

COMMENT ON COLUMN movimiento.valor IS 'Indica el valor del movimiento';

COMMENT ON COLUMN movimiento.id_cuenta IS 'Identificador de la cuenta a la que pertenece
este movimiento';

```

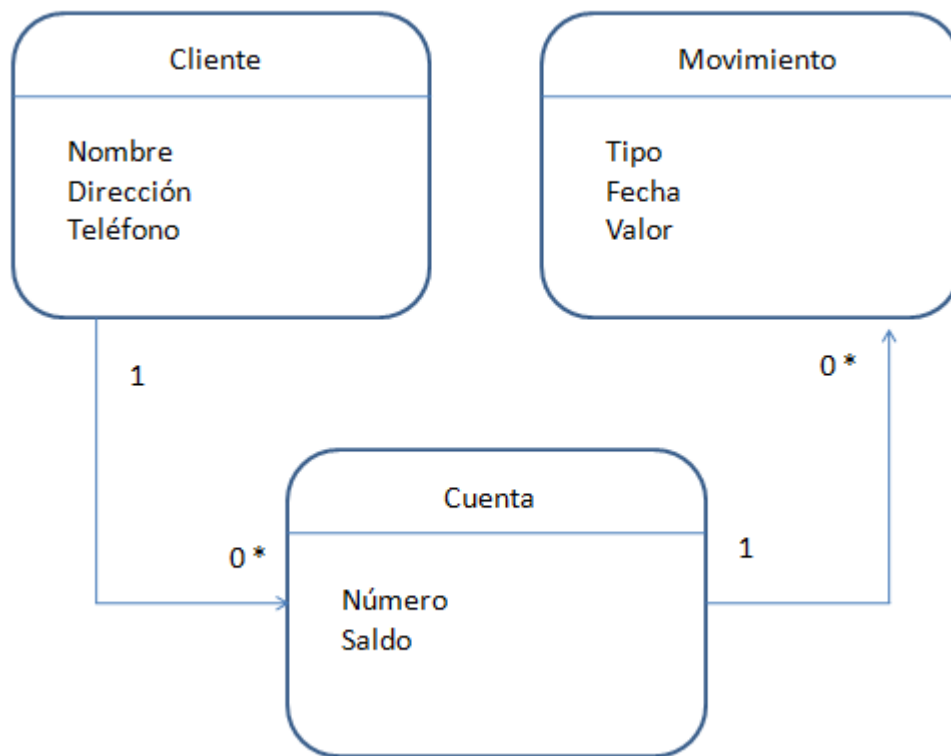
Una vez creada la base de datos y la tabla, se procede a configurar la cadena de conexion del archivo de conexion de hibernate llamado "datasource-tx-hp.xml" ubicado en la ruta "src/main/resource".

```

<bean id="dataSource"
      class="org.springframework.jdbc.datasource.DriverManagerDataSource">
    <property name="driverClassName">
        <value>org.postgresql.Driver</value>
    </property>
    <property name="url">
        <value>
            jdbc:postgresql://[NombreServidorDB]:5432/Banco_PagosOnline
        </value>
    </property>
    <property name="username" value="postgres" />
    <property name="password" value="[Clave]" />
</bean>

```

Modelo Entidad Relación



Fuentes

El proyecto fue desarrollado sobre STS (Eclipse junio, con plug-ins para Spring 3.0). Las dependencias se adjuntan y el archivo POM.xml indica cuales son.