**1. Create an account:** we use the Account and Cards table, which will create an ID with the user's full name, birthdate, tax number and ID number. The account id will generate the account's IBAN. When creating an account, the system shall automatically create two debit cards, called "Physical Card" and "Virtual Card", so will create the Cards table.

**2.** **Close an account:** we will use the Account table to delete the user's records. If the user has money in the account, they will need to transfer money to another account for the balance to become 0.

**3.View account balance:** we will use the Account table to check the account’s balance.

**4**. **View transactions:** will use the Transfers and Account table, to get the sender and receiver data.

* Filter by date: put the start and end date to see the transactions within those dates.
* By type (expense/income) all transactions are expense where SenderId = id of own, and if income RecieverId=id of own.
* By sender/recipient: filtering by recipient or sender id.
* Association of image and notes to a transaction -notes can be added after transactions.

**4.1 Change transaction’s notes:** we will use Transfers table to change the notes.

**4.2 Get a proof of transaction:** we will use Transfers table, with senderId, metadata and date.

**5.Transfer Money:** will use the Transfers table and Account.

**5.1 Make a bank transfer by IBAN:** will use the Transfers table and Account, the transfers enter the metadata attribute which will generate a destination Iban through the Account table.

**5.2** **Make a service payment:** will use the Transfers table and Account, where the Transfers table will generate a reference for the payment, the reciverId will be generated from the Account.

**5.3** **Make a government payment:** will use the Transfers table and Account, when we select metadata with payments to the government a ReciverId with an entity will be generated by the Account table.

**5.4** **Make a telco payment:** will use the Transfers table and Telco Provider, where we select the metadata with the option mobile phone charging, a mobile phone number will be inserted for charging, and the name and id from the Telco Provider will be given by the TelcoProviders table.

**6. View** **Schedule a transfer:** we use the StandingOrder table, and Account. Where the SenderId will be generated from the Account table,the receiverId will fetch the Id in the Account table .

**6.1 Schedule a transfer:** we use the StandingOrder table and Account. To schedule the transfer, the user will provide IBAN, an amount, frequency, and the date.

**6.2 Change a schedule transfer:** we use StandingOrder table,so we can change the reciverId and the amount of the transfer.

**6.3 Delete a schedule transfer:** we use StandingOrder table, to delete scheduled transfer we can use the reciverId or the date of the transfer.

**7. View Direct debits:**

**7.1 Toggle a direct debit:** will use the table DebitDirect and Transfers. Will create an Id in the table of transfers when a direct debit is finished. In the table DebitDirect is possible active or disabled this option of the payment and is possible to view all last debits made filtering by date.

**8.View cards:** we use the Cards and Account table, the card ID is generated from the account ID. When creating a card, the expiry date must be valid for 2 years after the creation of the card.

**8.1 Toggler online payments:** use Cards table to active or disabled the option with online payments.

**8.2 Toggler NFC payments:** use Cards table to active or disabled the option with NFC payments.

**8.3 Change PIN:** Account and Cards table will need, to access to the account we need Account that will generate Cards table where is possible to change pin code for 4 digits.

**9.Get Documents:**

**9.1 Get IBAN proof:** We are going to use the table Account, where we use id attribute to generate the IBAN and the account number, we use full name attribute to be able to generate the proof of Iban.

**9.2 Get account statements:** Weuse Account and Transfers table, use id from account and when enter in the transfers table to take information it will be need date attribute twice to define the filter of starting date and ending date. We also use amount in euros, sender, and receiver to show transfer statement emission.

**10. Change account information:** We will use Account and Address table. We need attributes such as Tax Number and the address Id that will generate new table Address where will be possible change information from the user.

**Account Entity**

|  |  |  |
| --- | --- | --- |
| Atribute | data type | description |
| id | long | Unique, primary key, from the id will be generate IBAN. The type is classified as long because they can store more numbers and don’t have decimal places. All values ​​in Long are divided by 100. |
| Fullname | varchar | The first and last name of the user. |
| birthdate | date | The system will ensure the user is at least 18 years old. |
| addressid | long |  |
| email | varchar | Email address of the user, the system will ensure the given email isn't already in use. |
| taxnumber | varchar | The tax number of the user |
| idnumber | int | The identification number of the user |
| ctreatedat | datetime | Date of creation of the account. |
| balance | long | The amount in euro that is available. The type is classified as long because they can store more numbers and don’t have decimal places. All values ​​in Long are divided by 100. |

**Cards Entity**

|  |  |  |
| --- | --- | --- |
| Atribute | data type | description |
| id | long | Unique, primary key, which identifies the cards |
| name | varchar | Name of the card, the system will generate two debit cards, called “Physical Card” and "Virtual Card |
| expirydate | date | Expiry date of the card |
| pincode | int | Pin code to gain access to the system |
| onlinepayments | Boolean | Online payments are Boolean type, active or disabled |
| nfcpayments | Boolean | NFC payments are Boolean type, active or disabled |
| accountid | long | Number account ID, the long keyword is a data type that can store whole numbers. |
| createdat | datetime | Will register date of the creation of the card |

**Transfers Entity**

|  |  |  |
| --- | --- | --- |
| Atribute | data type | description |
| id | long | Unique, primary key, which identifies each transfer |
| Date | datetime | Is who will send the transfer, will be generated from the account id. |
| senderid | long | Is who will receive the transfer, will be generated from the account id. |
| reciverid | long |  |
| amount | long | Amount that will be transferred. |
| metadata | json | Is the type of transfers in json format, which can be a transfer via IBAN, payment by ATM reference, government payments and mobile phone top-ups. |
| notes | varchar | Can be associate descriptive notes |
| images | image | Can be associate images |

**TelcoProviders Entity**

|  |  |  |
| --- | --- | --- |
| Atribute | data type | description |
| id | long | Unique, primary key, which identifies the Telco Providers |
| Name | varchar | The id that will be associate to the name:   * *Lycamobile GT Mobile* * *MEO* * *MEO Card* * *MEO Card - PT Hello /PT Card* * *MEO Card - TelefoneHello* * *MEO Escola Digital* * *Moche* * *NOS* * *NOS - Escola Digital* * *SAPO* * *Sapo ADSL* * *UZO* * *Via Card* * *Vodafone* * WTF |
| Updatedat | datetime | Date of the last update of database |

**StandingOrder Entity**

|  |  |  |
| --- | --- | --- |
| Atribute | data type | description |
| id | long | Unique, primary key, which identifies the Standing order. |
| ReciverID | long | Will generate from ID of table account or from TelcoProviders. |
| amount | long | The amount in euro that will be paid. |
| Frequency | Enum | “frequency” can be scheduled daily, weekly, monthly, and annually. |
| Senderid | long | Will generate from Id of table Account. |

**DirectDebit Entity**

|  |  |  |
| --- | --- | --- |
| Atribute | data type | description |
| id | long | Unique, primary key, which identifies the direct debit. |
| active | Boolean | Shows the active or not active status of the direct debit. |
| amount | long | The amount in euro connected to this debit. |
| receiverid | long | Will generate from ID of table account or from TelcoProviders. |
| Senderid | long | Will generate from Id of table Account. |
| LASTDEBIT | long | Is related to the id of table transfers, which allows checking which transfers are made by date and was the last. |

**Address Entity**

|  |  |  |
| --- | --- | --- |
| Atribute | data type | description |
| id | long | Unique, primary key, which identifies the address of the user |
| Line1 | varchar | The address of the user |
| line2 | varchar | The address of the user (optional) |
| postalcode | varchar | (####-###) |
| city | varchar |  |
| district | varchar |  |