

# DATABASE PROJECT

## *Completion of Database Administrator Course - IFRS*

In the first phase of the project, we will implement the Transactional Database alongside the development team. In the second phase of the project, we will connect the database with the CRM system, directly with the development (Backend) team for real-time data feeding.

The chosen Database for this project is PostgreSQL, in an on-premises environment. Below are the Conceptual, Logical, and Physical models for the implementation of the Transactional Database.

### **CONCEPTUAL MODEL**

#### **Entities**

1. Company
2. Revenue
3. Expense
4. Cost Center
5. Customer
6. Bank Transaction
7. Supplier
8. Service
9. Invoice
10. Payment

#### **Attributes**

##### **Company:**

- Name
- ID

##### **Revenue:**

- ID
- Amount
- Issue Date
- Customer (associated Customer ID)
- Payment Method

##### **Expense:**

- ID
- Amount
- Issue Date
- Supplier (associated Supplier ID)
- Cost Center (associated Cost Center ID)
- Payment Method

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

## **Cost Center:**

- ID
- Description
- Type

## **Customer:**

- ID
- Name
- CNPJ (National Register of Legal Entities)
- Email
- Phone
- Address

## **Bank Transaction:**

- ID
- Amount
- Issue Date
- Type (Revenue/Expense)
- Bank Account (associated Bank Account ID)
- Supplier (associated Supplier ID, if applicable)
- Customer (associated Customer ID, if applicable)

## **Supplier:**

- ID
- Name
- CNPJ
- Email
- Phone
- Address
- Bank
- Agency
- Bank Account

## **Service:**

- ID
- Description
- Unit Price
- Nature (Taxable/Non-Taxable)
- Payment Method

## **Invoice:**

- ID
- Number
- Issue Date
- Total Amount
- Customer (associated Customer ID)
- Services (provided Service IDs)

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

## Payment:

- ID
- Amount
- Payment Date
- Invoice (associated Invoice ID)
- Payment Method

## Relacionamentos

1. Bank - Income (1: )
2. Bank - Expenses (1: )
3. Expenses - Costs (N:1)
4. Income - Customers (N:1)
5. Expenses - Suppliers (N:1)
6. Transaction - Account (N:1)
7. Transaction - Customers/Suppliers (N:1)
8. Invoice - Service (N: )
9. Payment - Invoice (1: )
10. Account - Income (1: )
11. Account - Expenses (1: )
12. Account - Payment (1: )
13. Account - Transaction (1: )
14. Transaction - Transaction\_details (1: )
15. Invoice - Invoice\_details (1: )
16. Invoice\_details - Billing (N:1)
17. Income - Invoice (N:1)
18. Expenses - Invoice (N:1)

## DIMENSIONAL MODEL

## Tables Specifications

### ACCOUNT

- account\_id
- account\_id
- open\_dt
- last\_settlement\_date
- bill\_cycle\_id
- prev\_balance
- account\_class\_flag
- write\_off
- is\_vip
- payment\_term\_id
- account\_status

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

## INCOME

- inc\_id
- income\_status
- REC\_conf\_status
- REC\_cod
- REC\_client\_name
- REC\_social\_name
- amount\_received
- income\_collaborator\_id
- income\_collaborator\_name
- OCC\_isReceived
- OCC\_isPaid
- account\_id
- income\_category\_id
- income\_category\_name

## INVOICE

- INV\_id
- INV\_status
- REC\_cod
- REC\_client\_name
- REC\_social\_name
- INV\_value
- INV\_due\_date
- INV\_issue\_date
- INV\_number
- invoice\_category\_id
- INV\_isPaid
- account\_id
- income\_category\_id
- income\_category\_name

## INVOICE DETAILS

- INC\_id
- income\_status
- REC\_conf\_status
- REC\_cod
- REC\_client\_name
- REC\_social\_name
- invoice\_value
- invoice\_category\_value

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

- due\_date
- value\_date
- isPaid
- account\_id
- collaborator\_id
- product\_id
- income\_category\_id
- income\_category\_name
- service\_id
- service\_description

## **TRANSACTION**

- trans\_id
- REC\_cod
- trans\_dt
- account\_id
- transaction\_status
- account\_origin
- isDebit
- OCC\_isProcessed
- OCC\_isPaid
- OCC\_isInvoiced
- is\_conciliated
- collaborator\_id
- account\_destination
- payment\_method\_id
- payment\_method\_status
- invoice\_category\_value
- account\_destination\_id
- invoice\_id
- OCC\_center\_id

## **TRANSACTION\_DETAILS**

- transcd\_id
- trans\_id
- payment\_method\_id
- payment\_method\_name
- payment\_method\_value
- invoice\_id
- invoice\_value
- service\_id
- cost\_center\_id

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

## **PAYMENT**

- account\_id
- REC\_cod
- payment\_status
- account\_class\_flag
- bank\_id
- collaborator\_name
- collaborator\_id
- account\_name
- collaborator\_contact
- payment\_cod
- account\_status
- OCC\_isPaid
- OCC\_center\_id
- OCC\_center\_name
- OCC\_center\_description
- payment\_method\_id
- payment\_method\_name
- OCC\_center\_isReceived
- OCC\_center\_payment\_status
- OCC\_center\_payment\_status\_flag
- invoice\_category\_value
- account\_destination\_id
- payment\_method\_status

## **BANK**

- account\_id
- REC\_cod
- payment\_status
- account\_class\_flag
- bank\_id
- collaborator\_name
- collaborator\_id
- account\_name
- bank\_open\_dt
- bank\_contact
- payment\_cod
- account\_status
- bank\_status
- bank\_cod
- bank\_number
- OCC\_isPaid
- OCC\_center\_id
- OCC\_center\_name

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

- OCC\_center\_description
- payment\_method\_id
- payment\_method\_name
- OCC\_center\_isReceived
- OCC\_center\_payment\_status
- OCC\_center\_payment\_status\_flag
- invoice\_category\_value
- account\_destination\_id
- payment\_method\_status
- bank\_name\_suggestion

## **EXPENSES**

- account\_id
- expenses\_status
- expense\_class\_flag
- REC\_cod
- REC\_client\_name
- REC\_social\_name
- expense\_description
- OCC\_isProcessed
- OCC\_isPaid
- OCC\_expense\_isProcessed
- expense\_due\_date
- expense\_open\_dt
- account\_destination
- expense\_destination
- collaborator\_id
- payment\_method\_id
- payment\_method\_status
- expense\_value
- expense\_cod
- expense\_center\_id
- expense\_center\_name
- expense\_center\_description
- is\_expense\_flag
- invoice\_category\_value
- account\_destination\_id
- invoice\_value
- payment\_method\_status

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

## **COSTS**

- cost\_center\_id
- cost\_center\_description
- OCC\_center\_cod
- OCC\_center\_status
- OCC\_center\_contact
- OCC\_center\_address
- OCC\_center\_type
- OCC\_center\_account\_id
- OCC\_center\_payment\_id
- OCC\_center\_open\_dt
- OCC\_center\_name

## **CUSTOMERS**

- REC\_cod
- REC\_client\_name
- REC\_social\_name
- REC\_status
- REC\_client\_contact
- account\_id
- isVIP
- OCC\_center\_cod
- OCC\_center\_status
- OCC\_center\_contact
- OCC\_center\_address
- account\_open\_dt
- account\_payment\_id
- account\_payment\_status
- account\_status
- invoice\_category\_value
- total\_value

## **SERVICE**

- product\_id
- account\_category\_description
- account\_class\_flag
- account\_status
- account\_open\_dt
- collaborator\_id
- collaborator\_contact
- collaborator\_name
- account\_payment\_id
- service\_description



# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

- product\_open\_dt
- product\_value
- service\_isProcessed
- product\_class\_flag
- product\_center\_id
- product\_center\_name
- product\_value\_flag
- isVIP
- payment\_cod
- account\_center\_id
- product\_charge\_value

## **SUPPLIERS**

- REC\_id
- supplier\_status
- supplier\_class\_flag
- REC\_cod
- REC\_client\_name
- REC\_social\_name
- supplier\_address
- supplier\_contact
- supplier\_open\_dt
- account\_status
- supplier\_payment\_status
- supplier\_payment\_id
- supplier\_payment\_status\_flag
- supplier\_cod
- supplier\_number
- supplier\_center\_id
- supplier\_center\_name
- supplier\_center\_description
- OCC\_center\_isReceived
- OCC\_center\_payment\_status
- OCC\_center\_payment\_status\_flag
- account\_destination\_id
- payment\_method\_status
- supplier\_name\_suggestion

## **BILLING**

- bill\_id
- REC\_cod
- REC\_client\_name
- REC\_social\_name
- bill\_due\_date

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

- bill\_issue\_date
- bill\_number
- bill\_value
- invoice\_id
- OCC\_center\_isProcessed
- OCC\_center\_isPaid
- OCC\_center\_isInvoiced
- is\_conciliated
- collaborator\_id
- account\_id
- account\_name
- service\_id
- product\_id
- payment\_cod
- OCC\_center\_id
- OCC\_center\_status
- product\_category\_value
- invoice\_category\_value

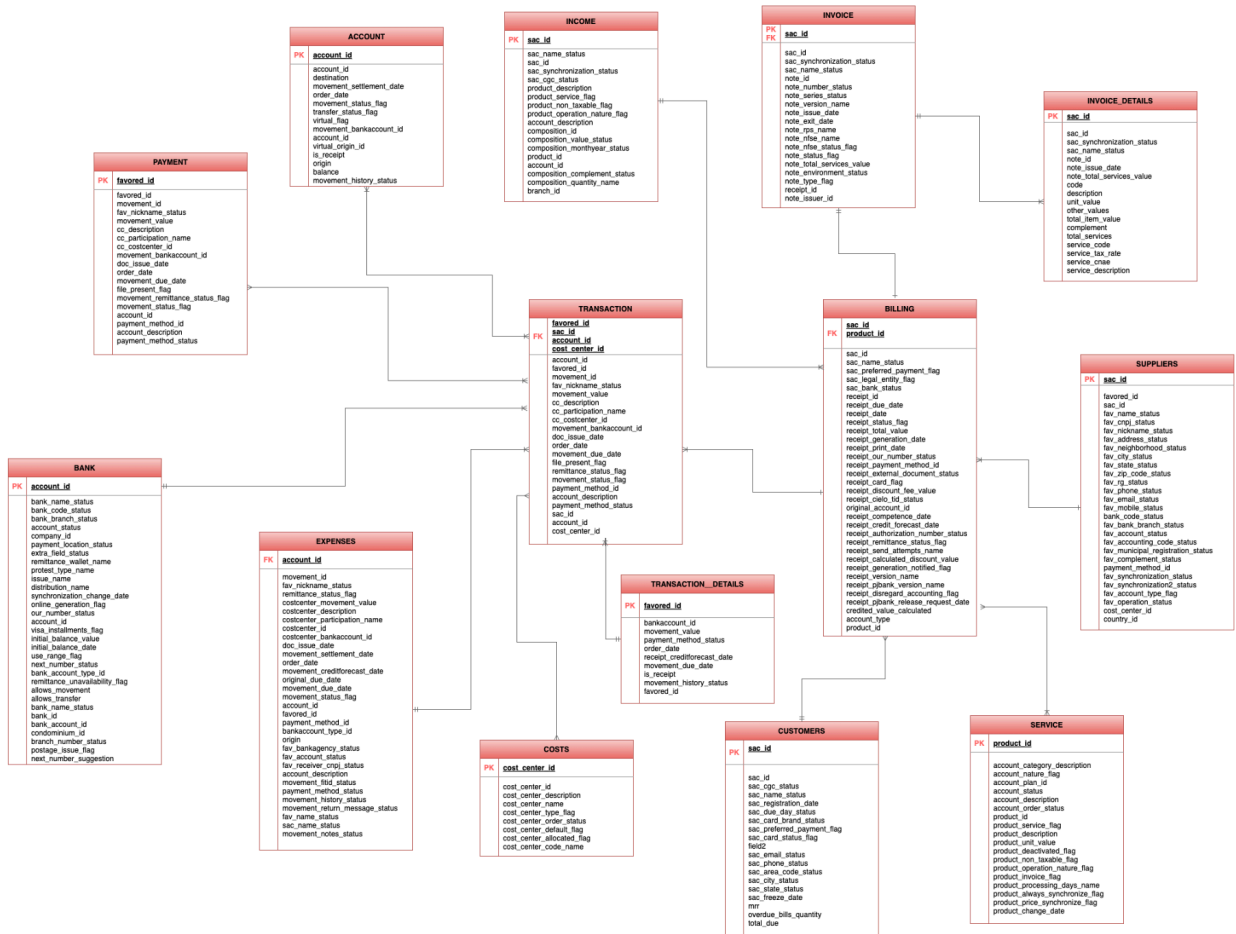
# DATABASE PROJECT

Completion of Database Administrator Course - IFRS

## LOGICAL MODEL

Diagrama de Entidade-Relacionamento (ERD)  
Projeto de Banco de Dados Transaccional

Finalização de Curso - IFRS



# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

## PHYSICAL MODEL

### Implement

- Account

```
CREATE TABLE ACCOUNT (  
    account_id INT PRIMARY KEY,  
    open_dt DATE,  
    last_settlement_date DATE,  
    bill_cycle_id INT,  
    prev_balance DECIMAL(10, 2),  
    account_class_flag CHAR(1),  
    write_off CHAR(1),  
    is_vip CHAR(1),  
    payment_term_id INT,  
    account_status CHAR(1)  
);
```

- Income

```
CREATE TABLE INCOME (  
    inc_id INT PRIMARY KEY,  
    income_status CHAR(1),  
    REC_conf_status CHAR(1),  
    REC_cod INT,  
    REC_client_name VARCHAR(100),  
    REC_social_name VARCHAR(100),  
    amount_received DECIMAL(10, 2),  
    income_collaborator_id INT,  
    income_collaborator_name VARCHAR(100),  
    OCC_isReceived CHAR(1),  
    OCC_isPaid CHAR(1),  
    account_id INT,  
    income_category_id INT,  
    income_category_name VARCHAR(100),  
    FOREIGN KEY (account_id) REFERENCES ACCOUNT(account_id)  
);
```

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

```
CREATE INDEX idx_income_collaborator_id ON INCOME
(income_collaborator_id);
CREATE INDEX idx_income_account_id ON INCOME (account_id);
```

- Invoice

```
CREATE TABLE INVOICE (
    INV_id INT PRIMARY KEY,
    INV_status CHAR(1),
    REC_cod INT,
    REC_client_name VARCHAR(100),
    REC_social_name VARCHAR(100),
    INV_value DECIMAL(10, 2),
    INV_due_date DATE,
    INV_issue_date DATE,
    INV_number INT,
    invoice_category_id INT,
    INV_isPaid CHAR(1),
    account_id INT,
    income_category_id INT,
    income_category_name VARCHAR(100),
    FOREIGN KEY (account_id) REFERENCES ACCOUNT(account_id)
);
```

```
CREATE INDEX idx_invoice_category_id ON INVOICE
(invoice_category_id);
CREATE INDEX idx_invoice_account_id ON INVOICE (account_id);
```

- Invoice\_details

```
CREATE TABLE INVOICE_DETAILS (
    INC_id INT PRIMARY KEY,
    income_status CHAR(1),
    REC_conf_status CHAR(1),
    REC_cod INT,
    REC_client_name VARCHAR(100),
    REC_social_name VARCHAR(100),
    invoice_value DECIMAL(10, 2),
    invoice_category_value DECIMAL(10, 2),
```

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

```
due_date DATE,  
value_date DATE,  
isPaid CHAR(1),  
account_id INT,  
collaborator_id INT,  
product_id INT,  
income_category_id INT,  
income_category_name VARCHAR(100),  
service_id INT,  
service_description VARCHAR(255),  
FOREIGN KEY (account_id) REFERENCES ACCOUNT(account_id)  
);
```

```
CREATE INDEX idx_invoice_details_collaborator_id ON INVOICE_DETAILS  
(collaborator_id);  
CREATE INDEX idx_invoice_details_account_id ON INVOICE_DETAILS  
(account_id);
```

- Transaction

```
CREATE TABLE TRANSACTION (  
    trans_id INT PRIMARY KEY,  
    REC_cod INT,  
    trans_dt DATE,  
    account_id INT,  
    transaction_status CHAR(1),  
    account_origin INT,  
    isDebit CHAR(1),  
    OCC_isProcessed CHAR(1),  
    OCC_isPaid CHAR(1),  
    OCC_isInvoiced CHAR(1),  
    is_conciliated CHAR(1),  
    collaborator_id INT,  
    account_destination INT,  
    payment_method_id INT,  
    payment_method_status CHAR(1),  
    invoice_category_value DECIMAL(10, 2),  
    account_destination_id INT,  
    invoice_id INT,  
    OCC_center_id INT,  
    FOREIGN KEY (account_id) REFERENCES ACCOUNT(account_id)  
);
```

**Author: Juliana Santimaria**  
**Date: 06-20-2024**

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

```
CREATE INDEX idx_transaction_account_id ON TRANSACTION (account_id);
CREATE INDEX idx_transaction_collaborator_id ON TRANSACTION
(collaborator_id);
```

- Transaction\_details

```
CREATE TABLE TRANSACTION_DETAILS (
    transcd_id INT PRIMARY KEY,
    trans_id INT,
    payment_method_id INT,
    payment_method_name VARCHAR(100),
    payment_method_value DECIMAL(10, 2),
    invoice_id INT,
    invoice_value DECIMAL(10, 2),
    service_id INT,
    cost_center_id INT,
    FOREIGN KEY (trans_id) REFERENCES TRANSACTION(trans_id)
);
```

```
CREATE INDEX idx_transaction_details_trans_id ON TRANSACTION_DETAILS
(trans_id);
```

- Payment

```
CREATE TABLE PAYMENT (
    account_id INT PRIMARY KEY,
    REC_cod INT,
    payment_status CHAR(1),
    account_class_flag CHAR(1),
    bank_id INT,
    collaborator_name VARCHAR(100),
    collaborator_id INT,
    account_name VARCHAR(100),
    collaborator_contact VARCHAR(100),
    payment_cod INT,
    account_status CHAR(1),
    OCC_isPaid CHAR(1),
    OCC_center_id INT,
    OCC_center_name VARCHAR(100),
    OCC_center_description VARCHAR(255),
    payment_method_id INT,
    payment_method_name VARCHAR(100),
```

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

```
OCC_center_isReceived CHAR(1),
OCC_center_payment_status CHAR(1),
OCC_center_payment_status_flag CHAR(1),
invoice_category_value DECIMAL(10, 2),
account_destination_id INT,
payment_method_status CHAR(1),
FOREIGN KEY (account_id) REFERENCES ACCOUNT(account_id)
);
```

```
CREATE INDEX idx_payment_collaborator_id ON PAYMENT (collaborator_id);
CREATE INDEX idx_payment_bank_id ON PAYMENT (bank_id);
```

- Bank

```
CREATE TABLE BANK (
    account_id INT PRIMARY KEY,
    REC_cod INT,
    payment_status CHAR(1),
    account_class_flag CHAR(1),
    bank_id INT,
    collaborator_name VARCHAR(100),
    collaborator_id INT,
    account_name VARCHAR(100),
    bank_open_dt DATE,
    bank_contact VARCHAR(100),
    payment_cod INT,
    account_status CHAR(1),
    bank_status CHAR(1),
    bank_cod INT,
    bank_number INT,
    OCC_isPaid CHAR(1),
    OCC_center_id INT,
    OCC_center_name VARCHAR(100),
    OCC_center_description VARCHAR(255),
    payment_method_id INT,
    payment_method_name VARCHAR(100),
    OCC_center_isReceived CHAR(1),
    OCC_center_payment_status CHAR(1),
    OCC_center_payment_status_flag CHAR(1),
    invoice_category_value DECIMAL(10, 2),
    account_destination_id INT,
    payment_method_status CHAR(1),
```



# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

```
bank_name_suggestion VARCHAR(100),  
FOREIGN KEY (account_id) REFERENCES ACCOUNT(account_id)  
);
```

```
CREATE INDEX idx_bank_collaborator_id ON BANK (collaborator_id);  
CREATE INDEX idx_bank_bank_cod ON BANK (bank_cod);
```

- Expenses

```
CREATE TABLE EXPENSES (  
    account_id INT PRIMARY KEY,  
    expenses_status CHAR(1),  
    expense_class_flag CHAR(1),  
    REC_cod INT,  
    REC_client_name VARCHAR(100),  
    REC_social_name VARCHAR(100),  
    expense_description VARCHAR(255),  
    OCC_isProcessed CHAR(1),  
    OCC_isPaid CHAR(1),  
    OCC_expense_isProcessed CHAR(1),  
    expense_due_date DATE,  
    expense_open_dt DATE,  
    account_destination INT,  
    expense_destination INT,  
    collaborator_id INT,  
    payment_method_id INT,  
    payment_method_status CHAR(1),  
    expense_value DECIMAL(10, 2),  
    expense_cod INT,  
    expense_center_id INT,  
    expense_center_name VARCHAR(100),  
    expense_center_description VARCHAR(255),  
    is_expense_flag CHAR(1),  
    invoice_category_value DECIMAL(10, 2),  
    account_destination_id INT,  
    invoice_value DECIMAL(10, 2),  
    payment_method_status CHAR(1),  
    FOREIGN KEY (account_id) REFERENCES ACCOUNT(account_id)  
);
```

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

```
CREATE INDEX idx_expenses_collaborator_id ON EXPENSES
(collaborator_id);
CREATE INDEX idx_expenses_account_id ON EXPENSES (account_id);
```

- **Costs**

```
CREATE TABLE COSTS (
    cost_center_id INT PRIMARY KEY,
    cost_center_description VARCHAR(255),
    OCC_center_cod INT,
    OCC_center_status CHAR(1),
    OCC_center_contact VARCHAR(100),
    OCC_center_address VARCHAR(255),
    OCC_center_type CHAR(1),
    OCC_center_account_id INT,
    OCC_center_payment_id INT,
    OCC_center_open_dt DATE,
    OCC_center_name VARCHAR(100)
);
```

- **Customers**

```
CREATE TABLE CUSTOMERS (
    REC_cod INT PRIMARY KEY,
    REC_client_name VARCHAR(100),
    REC_social_name VARCHAR(100),
    REC_status CHAR(1),
    REC_client_contact VARCHAR(100),
    account_id INT,
    isVIP CHAR(1),
    OCC_center_cod INT,
    OCC_center_status CHAR(1),
    OCC_center_contact VARCHAR(100),
    OCC_center_address VARCHAR(255),
    account_open_dt DATE,
    account_payment_id INT,
    account_payment_status CHAR(1),
    account_status CHAR(1),
    invoice_category_value DECIMAL(10, 2),
    total_value DECIMAL(10, 2),
    FOREIGN KEY (account_id) REFERENCES ACCOUNT(account_id)
);
```

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

```
CREATE INDEX idx_customers_account_id ON CUSTOMERS (account_id);
```

- Service

```
CREATE TABLE SERVICE (  
    product_id INT PRIMARY KEY,  
    account_category_description VARCHAR(255),  
    account_class_flag CHAR(1),  
    account_status CHAR(1),  
    account_open_dt DATE,  
    collaborator_id INT,  
    collaborator_contact VARCHAR(100),  
    collaborator_name VARCHAR(100),  
    account_payment_id INT,  
    service_description VARCHAR(255),  
    product_open_dt DATE,  
    product_value DECIMAL(10, 2),  
    service_isProcessed CHAR(1),  
    product_class_flag CHAR(1),  
    product_center_id INT,  
    product_center_name VARCHAR(100),  
    product_value_flag CHAR(1),  
    isVIP CHAR(1),  
    payment_cod INT,  
    account_center_id INT,  
    product_charge_value DECIMAL(10, 2),  
    FOREIGN KEY (account_center_id) REFERENCES ACCOUNT(account_id)  
);
```

```
CREATE INDEX idx_service_collaborator_id ON SERVICE (collaborator_id);  
CREATE INDEX idx_service_account_center_id ON SERVICE  
(account_center_id);
```

- Suppliers

```
CREATE TABLE SUPPLIERS (  
    REC_id INT PRIMARY KEY,  
    supplier_status CHAR(1),  
    supplier_class_flag CHAR(1),  
    REC_cod INT,  
    REC_client_name VARCHAR(100),  
    REC_social_name VARCHAR(100),
```

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

```
supplier_address VARCHAR(255),
supplier_contact VARCHAR(100),
supplier_open_dt DATE,
account_status CHAR(1),
supplier_payment_status CHAR(1),
supplier_payment_id INT,
supplier_payment_status_flag CHAR(1),
supplier_cod INT,
supplier_number INT,
supplier_center_id INT,
supplier_center_name VARCHAR(100),
supplier_center_description VARCHAR(255),
OCC_center_isReceived CHAR(1),
OCC_center_payment_status CHAR(1),
OCC_center_payment_status_flag CHAR(1),
account_destination_id INT,
payment_method_status CHAR(1),
supplier_name_suggestion VARCHAR(100),
FOREIGN KEY (account_destination_id) REFERENCES ACCOUNT(account_id)
);
```

```
CREATE INDEX idx_suppliers_supplier_cod ON SUPPLIERS (supplier_cod);
```

- **Billing**

```
CREATE TABLE BILLING (
    bill_id INT PRIMARY KEY,
    REC_cod INT,
    REC_client_name VARCHAR(100),
    REC_social_name VARCHAR(100),
    bill_due_date DATE,
    bill_issue_date DATE,
    bill_number INT,
    bill_value DECIMAL(10, 2),
    invoice_id INT,
    OCC_center_isProcessed CHAR(1),
    OCC_center_isPaid CHAR(1),
    OCC_center_isInvoiced CHAR(1),
    is_conciliated CHAR(1),
    collaborator_id INT,
    account_id INT,
    account_name VARCHAR(100),
```

**Author: Juliana Santimaria**  
**Date: 06-20-2024**

# DATABASE PROJECT

*Completion of Database Administrator Course - IFRS*

```
service_id INT,  
product_id INT,  
payment_cod INT,  
OCC_center_id INT,  
OCC_center_status CHAR(1),  
product_category_value DECIMAL(10, 2),  
invoice_category_value DECIMAL(10, 2),  
FOREIGN KEY (account_id) REFERENCES ACCOUNT(account_id)  
);
```

```
CREATE INDEX idx_billing_account_id ON BILLING (account_id);  
CREATE INDEX idx_billing_collaborator_id ON BILLING  
(collaborator_id);
```