

# Trabalho Experimental

## Fase 1 – Trabalho C

Mapeamento do diagrama E-R para o modelo relacional  
Normalização do modelo relacional até à 3.ª Forma Normal  
Implementação do modelo físico da base de dados  
Criação do diagrama da base de dados

---

### Licenciatura em Engenharia Informática

#### Base de Dados

Paulo Nogueira Martins  
Daniel Moreira Lopes Alexandre

#### **Autores**

Diogo António Costa Medeiros n.º 70633

Pedro Miguel Cunha da Silva n.º 70649

Rui João Barros Pinto n.º 70648

Vila Real, abril 2021

## ÍNDICE

1. INTRODUÇÃO .....	3
2. OBJETIVOS DO TRABALHO PRÁTICO .....	4
3. ENQUADRAMENTO TEÓRICO .....	3
4. DESENVOLVIMENTO .....	4
4.1 MAPEAMENTO DO DIAGRAMA E-R PARA O MODELO RELACIONAL .....	4
4.2 NORMALIZAÇÃO DO MODELO RELACIONAL ATÉ À 3.ª FORMA NORMAL.....	6
4.2.1 MODELO NORMALIZADO ATÉ À 3.ª FORMA NORMAL.....	12
4.3 IMPLEMENTAÇÃO DO MODELO FÍSICO DA BASE DE DADOS .....	14
4.4 CRIAÇÃO DO DIAGRAMA DA BASE DE DADOS.....	17
5 CONCLUSÃO .....	18
6 BIBLIOGRAFIA .....	18

## 1. INTRODUÇÃO

Por definição, uma base de dados é uma coleção organizada de informação estruturada, ou dados, tipicamente armazenada eletronicamente num sistema computacional. A base de dados é habitualmente controlada por um sistema de gestão de base de dados (SGBD) (What Is A Database | Oracle Portugal, 2021). Por outro lado, os dados são apenas elementos ou valores discretos que, isoladamente, não têm qualquer valor, só se transformam em informação quando relacionados ou interpretados de alguma forma.

## 2. ENQUADRAMENTO TEÓRICO

Um diagrama Entidade – Relacionamento (E-R) expressa graficamente a estrutura lógica de uma base de dados. Enquanto uma entidade representa um conjunto de objetos (concreto ou abstrato) do mundo real que possuem características comuns, o relacionamento representa a associação entre entidades. Todas as entidades devem possuir atributos que representam as suas propriedades elementares.

O modelo relacional descreve dados a um nível lógico e visual, abstraindo os detalhes de baixo nível do armazenamento de dados (Silberschatz, Korth, & Sudarshan, 2020). Para obter o modelo relacional, faz-se o mapeamento do diagrama E-R. Este mapeamento tem início na representação em tabelas das entidades e suas relações. Os atributos especiais tomam diferentes papéis no modelo relacional. Os identificadores das entidades são representados por chaves primárias no modelo relacional. As relações são mapeadas com o auxílio de atributos especiais – chaves estrangeiras.

A normalização é um processo sistemático, definido por um conjunto de regras bem definidas, que visa eliminar fontes de redundância nos dados. O processo de normalização ocorre através de um conjunto de fases que conduzem a base de dados a estados onde a redundância se torna cada vez menor. A cada um destes estados dá-se o nome de forma normal (FN).

Um modelo relacional pode encontrar-se na primeira forma normal (1FN) se os domínios de todos os seus atributos forem atômicos, ou seja, se não contiverem subestruturas (tais como atributos multivalorados e compostos) (Silberschatz, Korth, & Sudarshan, 2020). Já na segunda forma normal (2FN), todos os atributos não chave devem ser determinados pela totalidade da chave primária. Por fim, na terceira forma normal (3FN), não existem atributos não chave dependentes funcionalmente de outro atributo não chave (Kroenke & Auer, 2016).

### 3. OBJETIVOS DO TRABALHO PRÁTICO

Dado um diagrama Entidade-Relacionamento (E-R) de uma base de dados, foram solicitados o mapeamento do diagrama E-R para o modelo relacional; a normalização do mesmo até à 3.<sup>a</sup> Forma Normal; a implementação do modelo físico da base de dados com as respetivas restrições de integridade, recorrendo à linguagem SQL, bem como a criação do diagrama da base de dados.

### 4. DESENVOLVIMENTO

#### 4.1 MAPEAMENTO DO DIAGRAMA E-R PARA O MODELO RELACIONAL

Pessoas(ID, NIF, Nome, Apelido, Telefone, End\_Morada, End\_CP, End\_Localidade)

Pacientes(ID\_Pac)

- **ID\_Pac** referencia Pessoas

Funcionarios(ID\_Func, Salario)

- **ID\_Func** referencia Pessoas

Medicos(ID\_Med, Especialidade)

- **ID\_Med** referencia Funcionarios

Enfermeiros(ID\_Enf, Turno, Horas\_Extra)

- **ID\_Enf** referencia Funcionarios

Auxiliares(ID\_Aux, Antiguidade, Servico)

- **ID\_Aux** referencia Funcionarios

Inquerito(ID\_Pae, ID\_Func, Data, Descricao)

- **ID\_Pac** referencia Pacientes
- **ID\_Func** referencia Fucionarios

Nota: ID\_Op são distintos para cada operação. No entanto, de forma a facilitar o acesso e gestão dos dados da tabela Operar, as chaves estrangeiras compõem a chave primária com este atributo.

Operar(ID\_Op, ID\_Med, ID\_Enf, ID\_Pae, Data, Duracao) -- Entidade Associativa

- **ID\_Med** referencia Medicos
- **ID\_Enf** referencia Enfermeiros
- **ID\_Pac** referencia Pacientes

Agendar(ID\_Op, ID\_Med, ID\_Enf, ID\_Pae, ID\_Aux, Data, Data\_Agend, Local)

- **(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac)** referencia Operar
- **ID\_Aux** referencia Auxiliares

Pagar(ID\_Op, ID\_Med, ID\_Enf, ID\_Pae, ID\_Paciente, ID\_Aux, Data, Preco)

- **(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac)** referencia Operar
- **ID\_Paciente** referencia Pacientes
- **ID\_Aux** referencia Auxiliares

Alergias(ID\_Alerg, Tipo)

Paciente\_Alergia(ID\_Pae, ID\_Alerg)

- **ID\_Pac** referencia Pacientes
- **ID\_Alerg** referencia Alergias

## 4.2 NORMALIZAÇÃO DO MODELO RELACIONAL ATÉ À 3.ª FORMA NORMAL

Pessoas(ID, NIF, Nome, Apelido, Telefone, End\_Morada, End\_CP, End\_Localidade)

### Dependências funcionais:

ID → NIF, Nome, Apelido, Telefone, End\_Morada, End\_CP, End\_Localidade

End\_CP → End\_Localidade

NIF → Nome, Apelido, Telefone

1FN✓      2FN✓      3FN✗

### Normalização:

CPs(CP, Localidade)

1FN✓      2FN✓      3FN✓

NIFs(NIF, Nome, Apelido, Telefone)

1FN✓      2FN✓      3FN✓

Pessoas(ID, ~~NIF~~, Morada, ~~CP~~)

- **NIF** referencia NIFs
- **CP** referencia CPs

1FN✓      2FN✓      3FN✓

Pacientes(ID\_Pac)

- **ID\_Pac** referencia Pessoas

1FN✓      2FN✓      3FN✓

Alergias(ID\_Alerg, Tipo)

**Dependências funcionais:**

ID\_Alerg → Tipo

1FN✓          2FN✓          3FN✓

Paciente\_Alergia(ID\_Pac, ID\_Alerg)

- **ID\_Pac** referencia Pacientes
- **ID\_Alerg** referencia Alergias

1FN✓          2FN✓          3FN✓

Funcionarios(ID\_Func, Salario)

- **ID\_Func** referencia Pessoas

**Dependências funcionais:**

ID\_Func → Salario

1FN✓          2FN✓          3FN✓

Medicos(ID\_Med, Especialidade)

- **ID\_Med** referencia Funcionarios

**Dependências funcionais:**

ID\_Med → Especialidade

1FN✓          2FN✓          3FN✓

Enfermeiros(~~ID\_Enf~~, Turno, Horas\_Extra)

- **ID\_Enf** referencia Funcionarios

**Dependências funcionais:**

ID\_Enf → Turno, Horas\_Extra

1FN✓          2FN✓          3FN✓

Auxiliares(~~ID\_Aux~~, Antiguidade, Servico)

- **ID\_Aux** referencia Funcionarios

**Dependências funcionais:**

ID\_Aux → Antiguidade, Servico

1FN✓          2FN✓          3FN✓

Inquerito(~~ID\_Pac~~, ~~ID\_Func~~, Data, Descricao)

- **ID\_Pac** referencia Pacientes
- **ID\_Func** referencia Funcionarios

**Nota:** Assume-se que ‘Data’ é específica o suficiente (inclui horas e minutos) para distinguir inquéritos distintos realizados no mesmo dia.

**Dependências funcionais:**

ID\_Pac, Data → Descricao

1FN✓          2FN✗



### Normalização:

Descricoes(ID\_Pac, Data, Descricao)

- **ID\_Pac** referencia Pacientes

1FN✓

2FN✓

3FN✓

Inquerito(ID\_Pac, ID\_Func, Data)

- **(ID\_Pac, Data)** referencia Descricoes
- **ID\_Func** referencia Funcionarios

1FN✓

2FN✓

3FN✓

Operar(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac, Data, Duracao) -- Entidade Associativa

- **ID\_Med** referencia Medicos
- **ID\_Enf** referencia Enfermeiros
- **ID\_Pac** referencia Pacientes

### Dependências funcionais:

ID\_Op → Data, Duracao

1FN✓

2FN✗

### Normalização:

Info\_Op(ID\_Op, Data, Duracao)

1FN✓

2FN✓

3FN✓

Operar(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac)

- **ID\_Op** referencia Info\_Op
- **ID\_Med** referencia Medicos
- **ID\_Enf** referencia Enfermeiros
- **ID\_Pac** referencia Pacientes

1FN✓            2FN✓            3FN✓

Agendar(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac, ID\_Aux, Data, Data\_Agend, Local)

- **(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac)** referencia Operar
- **ID\_Aux** referencia Auxiliares

#### **Dependências funcionais:**

(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac), Data → Local

1FN✓            2FN✗

#### **Normalização:**

Local\_Op(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac, Data, Local)

- **(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac)** referencia Operar

1FN✓            2FN✓            3FN✓

Agendar(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac, ID\_Aux, Data, Data\_Agend)

- **(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac, Data)** referencia Local\_Op

1FN✓            2FN✓            3FN✓

Pagar(~~ID\_Op~~, ~~ID\_Med~~, ~~ID\_Enf~~, ~~ID\_Pae~~, ~~ID\_Paciente~~, ~~ID\_Aux~~, Data, Preço)

- (ID\_Op, ID\_Med, ID\_Enf, ID\_Pac) referencia Operar
- ID\_Paciente referencia Pacientes
- ID\_Aux referencia Auxiliares

### Dependências funcionais:

(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac) → Preço

1FN✓          2FN✗

### Normalização:

Preco\_Pag(~~ID\_Op~~, ~~ID\_Med~~, ~~ID\_Enf~~, ~~ID\_Pae~~, Preço)

- (ID\_Op, ID\_Med, ID\_Enf, ID\_Pac) referencia Operar

1FN✓          2FN✓          3FN✓

Pagar(~~ID\_Op~~, ~~ID\_Med~~, ~~ID\_Enf~~, ~~ID\_Pae~~, ~~ID\_Paciente~~, ~~ID\_Aux~~, Data)

- (ID\_Op, ID\_Med, ID\_Enf, ID\_Pac) referencia Preco\_Pag
- ID\_Paciente referencia Pacientes
- ID\_Aux referencia Auxiliares

1FN✓          2FN✓          3FN✓

#### 4.2.1 MODELO NORMALIZADO ATÉ À 3.<sup>a</sup> FORMA NORMAL

CPs(CP, Localidade)

NIFs(NIF, Nome, Apelido, Telefone)

Pessoas(ID, ~~NIF~~, Morada, ~~CP~~)

- **NIF** referencia NIFs
- **CP** referencia CPs

Pacientes(ID\_Pac)

- **ID\_Pac** referencia Pessoas

Alergias(ID\_Alerg, Tipo)

Paciente\_Alergia(ID\_Pac, ID\_Alerg)

- **ID\_Pac** referencia Pacientes
- **ID\_Alerg** referencia Alergias

Funcionarios(ID\_Func, Salario)

- **ID\_Func** referencia Pessoas

Medicos(ID\_Med, Especialidade)

- **ID\_Med** referencia Funcionarios

Enfermeiros(~~ID\_Enf~~, Turno, Horas\_Extra)

- **ID\_Enf** referencia Funcionarios

Auxiliares(~~ID\_Aux~~, Antiguidade, Servico)

- **ID\_Aux** referencia Funcionarios

Descricoes(~~ID\_Pae~~, Data, Descricao)

- **ID\_Pac** referencia Pacientes

Inquerito(~~ID\_Pae~~, Data, ~~ID\_Func~~)

- **(ID\_Pac, Data)** referencia Descricoes
- **ID\_Func** referencia Funcionarios

Info\_Op(ID\_Op, Data, Duracao)

Operar(~~ID\_Op~~, ~~ID\_Med~~, ~~ID\_Enf~~, ~~ID\_Pae~~)

- **ID\_Op** referencia Info\_Op
- **ID\_Med** referencia Medicos
- **ID\_Enf** referencia Enfermeiros
- **ID\_Pac** referencia Pacientes

Local\_Op(~~ID\_Op~~, ~~ID\_Med~~, ~~ID\_Enf~~, ~~ID\_Pae~~, Data, Local)

- **(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac)** referencia Operar

Agendar(~~ID\_Op~~, ~~ID\_Med~~, ~~ID\_Enf~~, ~~ID\_Pae~~, ~~ID\_Aux~~, Data, Data\_Agend)

- **(ID\_Op, ID\_Med, ID\_Enf, ID\_Pac, Data)** referencia Local\_Op

Preco\_Pag(ID\_Op, ID\_Med, ID\_Enf, ID\_Pae, Preco)

- (ID\_Op, ID\_Med, ID\_Enf, ID\_Pac) referencia Operar

Pagar(ID\_Op, ID\_Med, ID\_Enf, ID\_Pae, ID\_Paciente, ID\_Aux, Data)

- (ID\_Op, ID\_Med, ID\_Enf, ID\_Pac) referencia Preco\_Pag
- ID\_Paciente referencia Pacientes
- ID\_Aux referencia Auxiliares

### 4.3 IMPLEMENTAÇÃO DO MODELO FÍSICO DA BASE DE DADOS

```
1. -- Criação da base de dados
2. USE master;
3. CREATE DATABASE Hospital;
4. USE Hospital;
5. -- Eliminação da base de dados
6. USE master;
7. DROP DATABASE Hospital;
8. -- Criação das tabelas da BD Hospital
9. CREATE TABLE CPs (
10.     CP          CHAR(8),
11.     Localidade  VARCHAR(50) NOT NULL,
12.     PRIMARY KEY (CP),
13.     CHECK (CP LIKE '[0-9][0-9][[0-9][0-9]-[0-9][0-9][0-9]')
14. );
15.
16. CREATE TABLE NIFs (
17.     NIF          INTEGER,
18.     Nome          VARCHAR(50) NOT NULL,
19.     Apelido       VARCHAR(50) NOT NULL,
20.     Telefone      INTEGER NOT NULL,
21.     PRIMARY KEY (NIF),
22.     CHECK (NIF >= 100000000 AND NIF <= 999999999),
23.     CHECK ((Telefone >= 200000000 AND Telefone < 300000000) OR
24.         (Telefone >= 900000000 AND Telefone <= 999999999))
25. );
26. CREATE TABLE Pessoas (
27.     ID           INTEGER CHECK (ID > 0),
28.     NIF          INTEGER NOT NULL UNIQUE,
29.     Morada       VARCHAR(50) NOT NULL,
30.     CP          CHAR(8) NOT NULL,
31.     PRIMARY KEY (ID),
32.     FOREIGN KEY (NIF) REFERENCES NIFs(NIF),
33.     FOREIGN KEY (CP) REFERENCES CPs(CP)
34. );
35.
36. CREATE TABLE Pacientes (
37.     ID_Pac       INTEGER,
38.     PRIMARY KEY (ID_Pac),
```

```

39.         FOREIGN KEY (ID_Pac) REFERENCES Pessoas(ID)
40.     );
41.
42. CREATE TABLE Alergias(
43.     ID_Alerg INTEGER          CHECK (ID_Alerg > 0),
44.     Tipo      VARCHAR(50)    NOT NULL,
45.     PRIMARY KEY (ID_Alerg)
46. );
47.
48. CREATE TABLE Paciente_Alergia(
49.     ID_Pac  INTEGER,
50.     ID_Alerg INTEGER,
51.     PRIMARY KEY (ID_Pac, ID_Alerg),
52.     FOREIGN KEY (ID_Pac) REFERENCES Pacientes(ID_Pac),
53.     FOREIGN KEY (ID_Alerg) REFERENCES Alergias(ID_Alerg)
54. );
55.
56. CREATE TABLE Funcionarios(
57.     ID_Func INTEGER,
58.     Salario MONEY NOT NULL CHECK (Salario > 0),
59.     PRIMARY KEY (ID_Func),
60.     FOREIGN KEY (ID_Func) REFERENCES Pessoas(ID)
61. );
62.
63. CREATE TABLE Medicos(
64.     ID_Med      INTEGER,
65.     Especialidade VARCHAR(50) NOT NULL
66.     PRIMARY KEY (ID_Med),
67.     FOREIGN KEY (ID_Med) REFERENCES Funcionarios(ID_Func)
68. );
69.
70. CREATE TABLE Enfermeiros(
71.     ID_Enf      INTEGER,
72.     Turno        VARCHAR(50)    NOT NULL,
73.     Horas_Extra  INTEGER        NOT NULL CHECK (Horas_Extra >= 0),
74.     PRIMARY KEY (ID_Enf),
75.     FOREIGN KEY (ID_Enf) REFERENCES Funcionarios(ID_Func)
76. );
77.
78. CREATE TABLE Auxiliares(
79.     ID_Aux      INTEGER,
80.     Antiguidade  INTEGER        NOT NULL DEFAULT 0,
81.     Servico      VARCHAR(50)    NOT NULL,
82.     CHECK (Antiguidade >= 0),
83.     PRIMARY KEY (ID_Aux),
84.     FOREIGN KEY (ID_Aux) REFERENCES Funcionarios(ID_Func)
85. );
86.
87. CREATE TABLE Descricoes(
88.     ID_Pac      INTEGER,
89.     Data_Inq    DATETIME NOT NULL DEFAULT GETDATE(),
90.     Descricao   VARCHAR(100), -- Duracao opcional
91.     PRIMARY KEY (ID_Pac, Data_Inq),
92.     FOREIGN KEY (ID_Pac) REFERENCES Pacientes(ID_Pac)
93. );
94.
95. CREATE TABLE Inquerito(
96.     ID_Pac      INTEGER,
97.     Data_Inq    DATETIME,
98.     ID_Func      INTEGER,

```

```

99.          PRIMARY KEY (ID_Pac, Data_Inq, ID_Func),
100.         FOREIGN KEY (ID_Func) REFERENCES Funcionarios(ID_Func),
101.         FOREIGN KEY (ID_Pac, Data_Inq) REFERENCES Descricoes(ID_Pac,
Data_Inq),
102. );
103.
104. CREATE TABLE Info_Op(
105.         ID_Op    INTEGER IDENTITY(1,1),
106.         Data_Op  DATETIME NOT NULL,
107.         Duracao  INTEGER DEFAULT NULL, -- Duracao opcional
108.         PRIMARY KEY (ID_Op)
109. );
110.
111. CREATE TABLE Operar(
112.         ID_Op    INTEGER,
113.         ID_Med   INTEGER,
114.         ID_Enf   INTEGER,
115.         ID_Pac   INTEGER,
116.         PRIMARY KEY (ID_Op, ID_Med, ID_Enf, ID_Pac),
117.         FOREIGN KEY (ID_Op) REFERENCES Info_Op(ID_Op),
118.         FOREIGN KEY (ID_Med) REFERENCES Medicos(ID_Med),
119.         FOREIGN KEY (ID_Enf) REFERENCES Enfermeiros(ID_Enf),
120.         FOREIGN KEY (ID_Pac) REFERENCES Pacientes(ID_Pac)
121. );
122.
123. CREATE TABLE Local_Op(
124.         ID_Op    INTEGER,
125.         ID_Med   INTEGER,
126.         ID_Enf   INTEGER,
127.         ID_Pac   INTEGER,
128.         Data_Op  DATETIME,
129.         Local_Op VARCHAR(50) NOT NULL,
130.         PRIMARY KEY (ID_Op, ID_Med, ID_Enf, ID_Pac, Data_Op),
131.         FOREIGN KEY (ID_Op, ID_Med, ID_Enf, ID_Pac) REFERENCES
Operar(ID_Op, ID_Med, ID_Enf, ID_Pac)
132. );
133.
134. CREATE TABLE Agendar(
135.         ID_Op    INTEGER,
136.         ID_Med   INTEGER,
137.         ID_Enf   INTEGER,
138.         ID_Pac   INTEGER,
139.         ID_Aux   INTEGER,
140.         Data_Op  DATETIME,
141.         Data_Agend DATETIME DEFAULT GETDATE(),
142.         PRIMARY KEY (ID_Op, ID_Med, ID_Enf, ID_Pac, ID_Aux, Data_Op,
Data_Agend),
143.         FOREIGN KEY (ID_Op, ID_Med, ID_Enf, ID_Pac, Data_Op) REFERENCES
Local_Op(ID_Op, ID_Med, ID_Enf, ID_Pac, Data_Op)
144. );
145.
146. CREATE TABLE Preco_Pag(
147.         ID_Op    INTEGER,
148.         ID_Med   INTEGER,
149.         ID_Enf   INTEGER,
150.         ID_Pac   INTEGER,
151.         Preco    MONEY NOT NULL CHECK (Preco >= 0),
152.         PRIMARY KEY (ID_Op, ID_Med, ID_Enf, ID_Pac),
153.         FOREIGN KEY (ID_Op, ID_Med, ID_Enf, ID_Pac) REFERENCES
Operar(ID_Op, ID_Med, ID_Enf, ID_Pac)

```

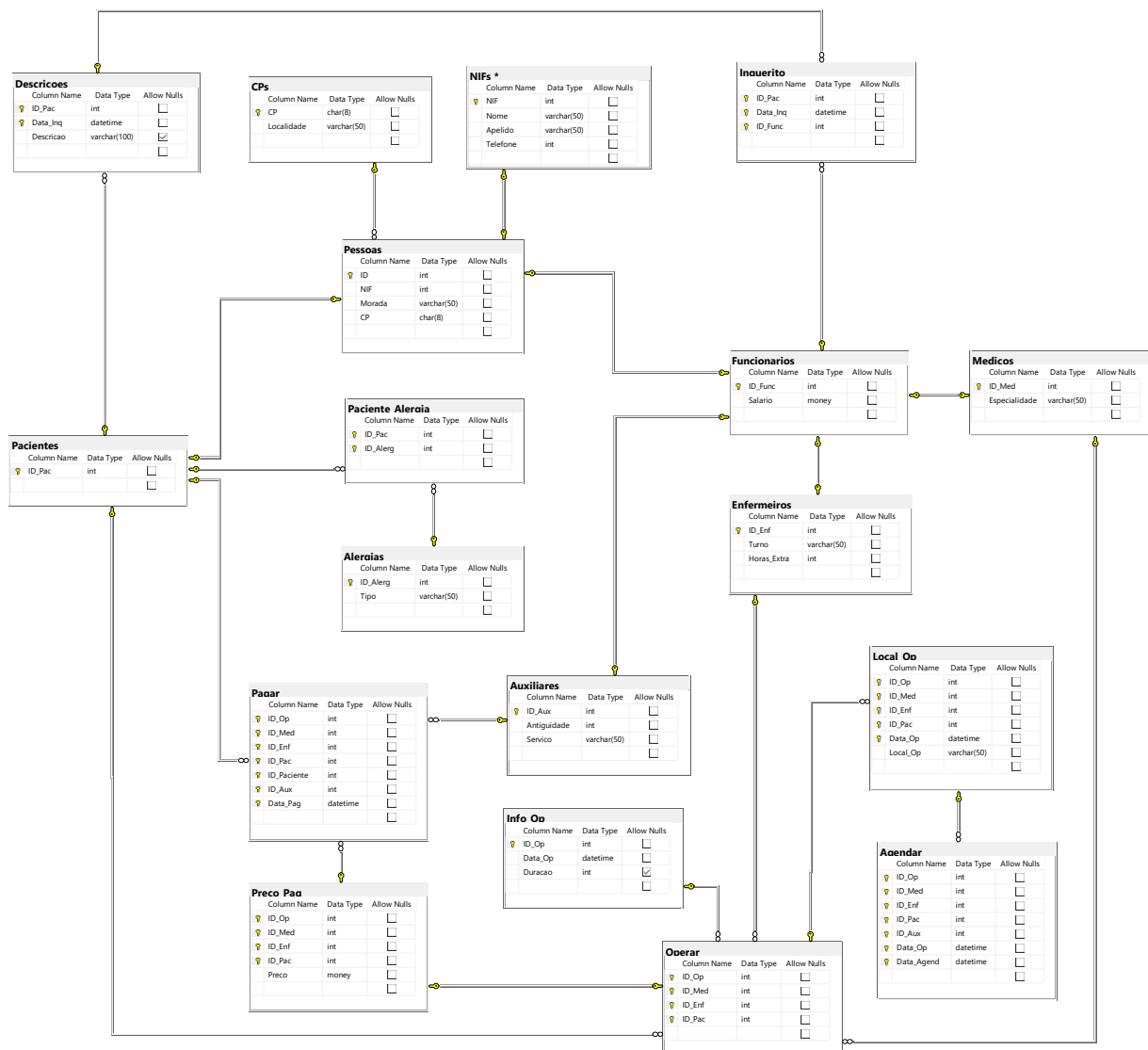


```

154. );
155.
156. CREATE TABLE Pagar (
157.     ID_Op          INTEGER,
158.     ID_Med          INTEGER,
159.     ID_Enf          INTEGER,
160.     ID_Pac          INTEGER,
161.     ID_Paciente     INTEGER,
162.     ID_Aux          INTEGER,
163.     Data_Pag        DATETIME DEFAULT GETDATE(),
164.     PRIMARY KEY (ID_Op, ID_Med, ID_Enf, ID_Pac, ID_Paciente,
ID_Aux, Data_Pag),
165.     FOREIGN KEY (ID_Paciente) REFERENCES Pacientes(ID_Pac),
166.     FOREIGN KEY (ID_Aux) REFERENCES Auxiliares,
167.     FOREIGN KEY (ID_Op, ID_Med, ID_Enf, ID_Pac) REFERENCES
Preco_Pag(ID_Op, ID_Med, ID_Enf, ID_Pac)
168. );

```

#### 4.4 CRIAÇÃO DO DIAGRAMA DA BASE DE DADOS



## 5 CONCLUSÃO

Face ao trabalho desenvolvido, crê-se ter atingido os objetivos definidos para esta etapa, nomeadamente o desenvolvimento do modelo relacional e sua normalização, e a implementação do modelo físico da base de dados.

## 6 BIBLIOGRAFIA

- Kroenke, D. M., & Auer, D. J. (2016). *Database Processing: Fundamentals, Design, and Implementation*. Edinburgh Gate, Harlow, Essex CM20 2JE, England: Pearson Education.
- Silberschatz, A., Korth, H. F., & Sudarshan, S. (2020). *Database System Concepts*. 2 Penn Plaza, New York, NY 10121: McGraw-Hill Education.
- What Is A Database | Oracle Portugal*. (23 de abril de 2021). Obtido de Oracle Portugal: <https://www.oracle.com/pt/database/what-is-database/>