



FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Desenvolvimento de uma Base de Dados para uma Companhia Aérea

Bases de Dados - 2015/2016

João Ferreira up201405163@fe.up.pt

Nuno Silva up201404380@fe.up.pt

Pedro Amaro up201405163@fe.up.pt

Dirigido por:

Carla Alexandra Teixeira Lopes ctl@fe.up.pt

Índice

Introdução	2
Descrição do Contexto	3
Definições dos principais conceitos	4
Cálculo do preço da reserva	5
Diagrama UML	7
Esquema Relacional	8
Instruções LDD-SQL para a criação da Base de Dados	9
Instruções LMD-SQL para a modificação da Base de Dados	13
Conclusão	17
Referências	18

Introdução

Este relatório foi feito no âmbito da unidade curricular "Base de Dados" e centra-se no estudo da melhor maneira de criar uma base de dados, para uma companhia aérea, na perspectiva do utilizador. Com esta base de dados, o utilizador poderá consultar informações sobre os voos disponíveis, ver o preço de uma reserva e ajustar várias opções em relação aos voos de cada passageiro.

Descrição do contexto

Esta base de dados foca-se no armazenamento de informação relevante para um utilizador. Será possível gerir a oferta de voos, tal como todos os detalhes dos voos em questão (origem, destino e lugares disponíveis). Um utilizador, ao fazer uma reserva, que pode ser constituída por vários voos, necessita de indicar os passageiros pertencentes à mesma. Para além disso, o utilizador terá oportunidade de fazer escolhas sobre o voo de cada passageiro (refeição predileta, refeições extra, malas e animais de estimação).

Surgiu um impasse com o cálculo do preço, sendo que foi recomendado o desenvolvimento e a explicitação de um algoritmo exemplificativo que calculasse o preço da reserva.

Definições dos principais conceitos

Uma reserva é constituída por passageiros e vários voos, numa certa ordem, tendo um preço associado, que varia com vários fatores, nomeadamente se a reserva for de ida e de volta (se a origem do primeiro voo e o destino do ultimo coincidirem).

Cada voo, associado a um trajeto, tem uma lotação limitada de passageiros, assim como datas e horas de partida/chegada referentes ao mesmo. Um trajeto tem um nome identificativo, um aeroporto de partida e outro de chegada (por exemplo, “TP0087” seria entre Lisboa e São Paulo).

Um passageiro, em cada voo, pode escolher as refeições que irá consumir, assim como a sua quantidade, a sua classe, e pode, inclusive, levar no porão do avião malas *standard* (até 23kg, com dimensões menores que 20cm x 30cm x 50cm) ou malas não *standard* (até 32kg, com dimensões menores que 50cm x 75cm x 150cm), assim como até 2 animais de estimação.

Finalmente, cada aeroporto serve uma cidade, num dado país e possui um código identificativo de 3 caracteres (código IATA).

Cálculo do preço da reserva

Como foi dito anteriormente, não é possível conhecer a equação exata usada pelas companhias aéreas para calcular o preço das suas reservas, tornando-se necessário assumir uma expressão intuitivamente correta para o cálculo do preço de cada reserva. Devido à dificuldade deste problema, foi decidido dedicar uma secção à descrição do algoritmo desenvolvido.

Sendo $mRetorno$ o modificador do preço de retorno (será 1 se a reserva for apenas de ida, ou 0,7 se for de ida e de volta), $nVoos$ o número de voos, $nPax$ o número de passageiros e $Pvoo(i,j)$ o preço do voo i para o passageiro j , o preço de uma reserva pode ser dado por:

$$Preserva = mRetorno \sum_{i=0}^{nVoos} \sum_{j=0}^{nPax} Pvoo(i,j)$$

O preço de um voo i , para um passageiro j , é calculado tendo em conta um modificador de classe $mClasse(i,j)$ (1 para a classe turística e 3 para a primeira classe), um modificador baseado no número de passageiros na reserva $mColectivo(j)$, sendo que cada passageiro adicional na reserva significa uma diminuição no preço do voo, segundo a fórmula:

$$mColectivo(j) = 5/16 * (4/5)^j + 3/4$$

Tendo ainda em conta o preço base do voo $Pbase(i)$ e o preço de qualquer extra que o passageiro tenha comprado para este voo $Pextra(i,j)$, o preço do voo é calculado da seguinte forma:

$$Pvoo(i,j) = mClasse(i,j) mColectivo(j) Pbase(i,d,p) + Pextra(i,j)$$

O preço base de um voo i é baseado na distância $dist(i)$ deste e em 3 modificadores: $mDias(d)$, $mCheio(p)$ e $mDiaEspecial$. Os primeiros dois podem ser calculados da seguinte forma: sendo d o número de dias restantes até à data de partida do voo e p a percentagem de ocupação da lotação atual do avião:

$$mDias(d) = 6,75 \times 10^{-6} d^2$$

$$mCheio(p) = \frac{2}{5} (4 - 3p)$$

O modificador $mDiaEspecial$ é 1,5 se a data de partida do voo coincidir com um feriado nacional e 1 nos outros dias. Assim pode-se calcular o preço base:

$$P_{base}(i, d, p) = (35 + 0,08 \text{ dist}(i)) m_{DiaEspecial} m_{Cheio}(p) m_{Dias}(d)$$

Finalmente, o preço extra de cada passageiro j num voo i são os gastos extras nos seguintes campos: malas, animais de estimação e refeições, e pode ser calculado da seguinte forma:

$$P_{extra}(i, j) = P_{malas}(i, j) + P_{animais}(i, j) + P_{refeicao}(i, j)$$

Os campos à direita desta equação podem ser determinados segundo os seguintes critérios:

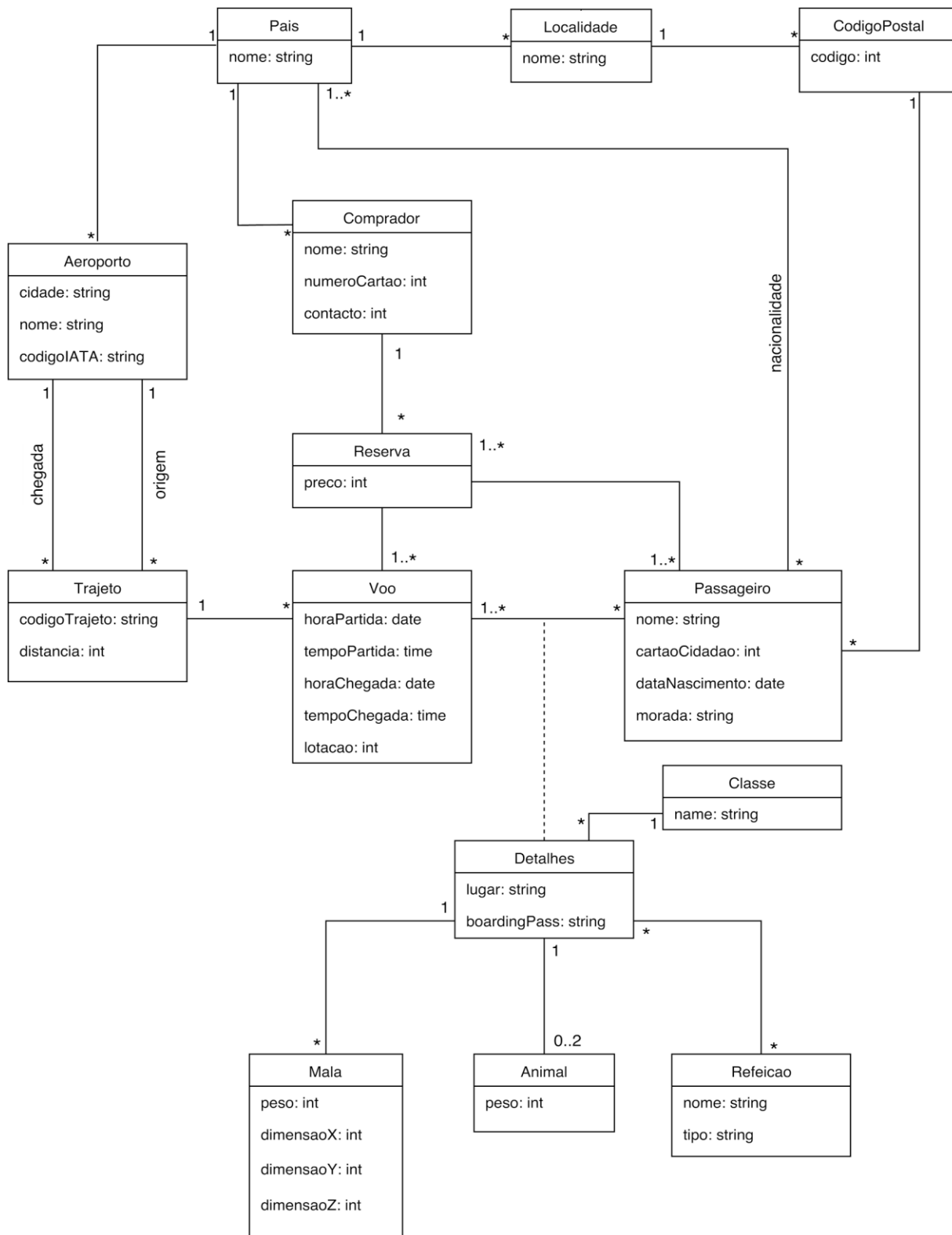
Preço de malas <i>standard</i>					
N	1	2	3	4	n
Preço / voo	0	20	50	100	+100 / mala

Os preços de malas não *standard* são determinados a partir da soma do preço por volume (150/m³) e do preço por peso (10/kg).

Preço para animais		
N	1	2
Preço / voo	50	150

Preço das refeições			
N	1	2	n
Preço / voo	0	5	+10 / refeição

Diagrama UML



Esquema Relacional

Pais(idPais, nome)
Localidade(idLocalidade, idPais-Pais, nome)
CodigoPostal(idCodigoPostal, idLocalidade-Localidade, codigo)
Comprador(idComprador, nome, numeroCartao, contato, idPais-Pais)
Reserva(idReserva, preco, idComprador-Comprador)
Aeroporto(idAeroporto, cidade, nome, codigoIATA, idPais-Pais)
Trajeto(idTrajeto, idAeroportoOrigem-Aeroporto, idAeroportoChegada-Aeroporto, codigoTrajeto, distancia)
Voo(idVoo, idTrajeto-Trajeto, dataPartida, dataChegada, lotacao)
Passageiro(idPassageiro, nome, cartaoCidadao, dataNascimento, morada, idPais-Pais, idCodigoPostal-CodigoPostal)
Classe(idClasse, nome)
Mala(idMala, idDetalhes-Detalhes, peso, dimensaoX, dimensaoY, dimensaoZ)
Animal(idAnimal, idDetalhes-Detalhes, peso)
Refeicao(idRefeicao, nome, tipo)
Detalhes(idDetalhes, lugar, boardingPass, idClasse-Classe)
ReservaPassageiro(idPassageiro-Passageiro, idReserva-Reserva)
ReservaVoo(idVoo-Voo, idReserva-Reserva)
PassageiroVooDetalhes(idVoo-Voo, idPassageiro-Passageiro, idDetalhes-Detalhes)
DetalhesRefeicao(idDetalhes-Detalhes, idRefeicao-Refeicao)

Instruções LDD-SQL para a criação da Base de Dados

```
PRAGMA FOREIGN_KEYS=ON;

.mode columns
.headers on
.nullvalue NULL

DROP TABLE IF EXISTS Pais;
CREATE TABLE Pais (
    idPais NUMBER PRIMARY KEY,
    nome TEXT
);

DROP TABLE IF EXISTS Localidade;
CREATE TABLE Localidade (
    idLocalidade NUMBER PRIMARY KEY,
    idPais NUMBER,
    nome TEXT,
    FOREIGN KEY(idPais) REFERENCES Pais(idPais)
);

DROP TABLE IF EXISTS CodigoPostal;
CREATE TABLE CodigoPostal (
    idCodigoPostal NUMBER PRIMARY KEY,
    idLocalidade NUMBER,
    codigo NUMBER,
    FOREIGN KEY(idLocalidade) REFERENCES Localidade(idLocalidade)
);

DROP TABLE IF EXISTS Comprador;
CREATE TABLE Comprador (
    idComprador NUMBER PRIMARY KEY,
    nome TEXT,
    numeroCartao NUMBER,
    contato NUMBER,
    idPais NUMBER,
    FOREIGN KEY(idPais) REFERENCES Pais(idPais)
);

DROP TABLE IF EXISTS Reserva;
CREATE TABLE Reserva (
    idReserva NUMBER PRIMARY KEY,
    preco NUMBER,
    idComprador NUMBER,
    FOREIGN KEY(idComprador) REFERENCES Comprador(idComprador)
);

DROP TABLE IF EXISTS Aeroporto;
CREATE TABLE Aeroporto (
    idAeroporto NUMBER PRIMARY KEY,
    cidade TEXT,
```

```

        nome TEXT,
        codigoIATA TEXT,
        idPais NUMBER,
        FOREIGN KEY(idPais) REFERENCES Pais(idPais)
    );

DROP TABLE IF EXISTS Trajeto;
CREATE TABLE Trajeto (
    idTrajeto NUMBER PRIMARY KEY,
    idAeroportoOrigem NUMBER,
    idAeroportoChegada NUMBER,
    codigoTrajeto TEXT,
    distancia NUMBER,
    FOREIGN KEY(idAeroportoOrigem) REFERENCES Aeroporto(idAeroporto),
    FOREIGN KEY(idAeroportoChegada) REFERENCES Aeroporto(idAeroporto)
);

DROP TABLE IF EXISTS Voo;
CREATE TABLE Voo (
    idVoo NUMBER PRIMARY KEY,
    idTrajeto NUMBER,
    dataPartida DATE,
    dataChegada DATE,
    lotacao NUMBER,
    FOREIGN KEY(idTrajeto) REFERENCES Trajeto(idTrajeto)
);

DROP TABLE IF EXISTS Passageiro;
CREATE TABLE Passageiro (
    idPassageiro NUMBER PRIMARY KEY,
    nome TEXT,
    cartaoCidadao NUMBER,
    dataNascimento DATE,
    morada TEXT,
    idPais NUMBER,
    idCodigoPostal NUMBER,
    FOREIGN KEY(idPais) REFERENCES Pais(idPais),
    FOREIGN KEY(idCodigoPostal) REFERENCES CodigoPostal(idCodigoPostal)
);

DROP TABLE IF EXISTS Classe;
CREATE TABLE Classe (
    idClasse NUMBER PRIMARY KEY,
    nome TEXT
);

DROP TABLE IF EXISTS Mala;
CREATE TABLE Mala (
    idMala NUMBER PRIMARY KEY,
    idDetalhes NUMBER,
    peso NUMBER,
    dimensaoX NUMBER,

```

```

        dimensaoY NUMBER,
        dimensaoZ NUMBER,
        FOREIGN KEY(idDetalhes) REFERENCES Detalhes(idDetalhes)
    );

DROP TABLE IF EXISTS Animal;
CREATE TABLE Animal (
    idAnimal NUMBER PRIMARY KEY,
    idDetalhes NUMBER,
    peso NUMBER,
    FOREIGN KEY(idDetalhes) REFERENCES Detalhes(idDetalhes)
);

DROP TABLE IF EXISTS Refeicao;
CREATE TABLE Refeicao (
    idRefeicao NUMBER PRIMARY KEY,
    nome TEXT,
    tipo TEXT
);

DROP TABLE IF EXISTS Detalhes;
CREATE TABLE Detalhes (
    idDetalhes NUMBER PRIMARY KEY,
    lugar TEXT,
    boardingPass TEXT,
    idClasse NUMBER,
    FOREIGN KEY(idClasse) REFERENCES Classe(idClasse)
);

DROP TABLE IF EXISTS ReservaPassageiro;
CREATE TABLE ReservaPassageiro (
    idPassageiro NUMBER,
    idReserva NUMBER,
    FOREIGN KEY(idPassageiro) REFERENCES Passageiro(idPassageiro),
    FOREIGN KEY(idReserva) REFERENCES Reserva(idReserva),
    CONSTRAINT idReservaPassageiro PRIMARY KEY (idPassageiro, idReserva)
);

DROP TABLE IF EXISTS ReservaVoo;
CREATE TABLE ReservaVoo (
    idVoo NUMBER,
    idReserva NUMBER,
    FOREIGN KEY(idVoo) REFERENCES Voo(idVoo),
    FOREIGN KEY(idReserva) REFERENCES Reserva(idReserva),
    CONSTRAINT idReservaVoo PRIMARY KEY (idVoo, idReserva)
);

DROP TABLE IF EXISTS PassageiroVooDetalhes;
CREATE TABLE PassageiroVooDetalhes (
    idVoo NUMBER,
    idPassageiro NUMBER,
    idDetalhes NUMBER,

```

```
FOREIGN KEY(idVoo) REFERENCES Voo(idVoo),  
FOREIGN KEY(idPassageiro) REFERENCES Passageiro(idPassageiro),  
FOREIGN KEY(idDetalhes) REFERENCES Detalhes(idDetalhes),  
CONSTRAINT idPassageiroVooDetalhes PRIMARY KEY (idVoo, idPassageiro)  
);
```

```
DROP TABLE IF EXISTS DetalhesRefeicao;
```

```
CREATE TABLE DetalhesRefeicao (  
    idDetalhes NUMBER,  
    idRefeicao NUMBER,  
    FOREIGN KEY(idDetalhes) REFERENCES Detalhes(idDetalhes),  
    FOREIGN KEY(idRefeicao) REFERENCES Refeicao(idRefeicao),  
    CONSTRAINT idDetalhesRefeicao PRIMARY KEY (idDetalhes, idRefeicao)  
);
```

Instruções LMD-SQL para o preenchimento da Base de Dados

```
PRAGMA FOREIGN_KEYS=ON;

.mode columns
.headers on
.nullvalue NULL

INSERT INTO Classe VALUES (0, "Turistica");
INSERT INTO Classe VALUES (1, "Primeira Classe");

INSERT INTO Refeicao VALUES (0, "Galinha", "Carne");
INSERT INTO Refeicao VALUES (1, "Bife", "Carne");
INSERT INTO Refeicao VALUES (2, "Bacalhau", "Peixe");
INSERT INTO Refeicao VALUES (3, "Salada", "Vegetariana");

INSERT INTO Pais VALUES ( 0, "Portugal");
INSERT INTO Pais VALUES ( 1, "Espanha");
INSERT INTO Pais VALUES ( 2, "Franca");
INSERT INTO Pais VALUES ( 3, "Belgica");
INSERT INTO Pais VALUES ( 4, "Reino Unido");
INSERT INTO Pais VALUES ( 5, "Brazil");
INSERT INTO Pais VALUES ( 6, "Russia");
INSERT INTO Pais VALUES ( 7, "Polonia");
INSERT INTO Pais VALUES ( 8, "Republica Checa");
INSERT INTO Pais VALUES ( 9, "Gana");
INSERT INTO Pais VALUES (10, "Estados Unidos");
INSERT INTO Pais VALUES (11, "Angola");

INSERT INTO Aeroporto VALUES ( 0, "Madrid", "Madrid-Barajas", "MAD", 1);
INSERT INTO Aeroporto VALUES ( 1, "Lisboa", "Portela", "LIS", 0);
INSERT INTO Aeroporto VALUES ( 2, "Barcelona", "Barcelona", "BCN", 1);
INSERT INTO Aeroporto VALUES ( 3, "Recife", "Int'l do Recife", "REC", 5);
INSERT INTO Aeroporto VALUES ( 4, "Moscovo", "Int'l Domodedovo", "DME", 6);
INSERT INTO Aeroporto VALUES ( 5, "Warsaw", "Frederic Chopin de Varsovia", "WAW", 7);
INSERT INTO Aeroporto VALUES ( 6, "Praga", "Int'l Ruzyně", "PRG", 8);
INSERT INTO Aeroporto VALUES ( 7, "Acra", "Int'l Kotoka", "ACC", 9);
INSERT INTO Aeroporto VALUES ( 8, "Santa Cruz", "Madeira", "FNC", 0);
INSERT INTO Aeroporto VALUES ( 9, "Lajes", "Base Aerea das Lajes", "TER", 0);
INSERT INTO Aeroporto VALUES (10, "Newark", "Int'l de Newark", "KEWR", 10);
INSERT INTO Aeroporto VALUES (11, "Miami", "Int'l de Miami", "KMIA", 10);
INSERT INTO Aeroporto VALUES (12, "Luanda", "Quatro de Fevereiro", "LAD", 11);
INSERT INTO Aeroporto VALUES (13, "Fortaleza", "Int'l de Fortaleza", "FOR", 5);
INSERT INTO Aeroporto VALUES (14, "Manchester", "Manchester", "MAN", 4);
INSERT INTO Aeroporto VALUES (15, "Oostende", "Ostend-Bruges Int'l", "OST", 3);
INSERT INTO Aeroporto VALUES (16, "Londres", "Londres Heathrow", "LHR", 4);
INSERT INTO Aeroporto VALUES (17, "Paris", "Paris-Orly", "ORY", 2);
INSERT INTO Aeroporto VALUES (18, "Porto", "Francisco Sa Carneiro", "OPO", 0);

INSERT INTO Trajeto VALUES ( 0, 0, 1, "TAP1017", 514);
INSERT INTO Trajeto VALUES ( 1, 1, 0, "TAP1018", 512);
```

```

INSERT INTO Trajeto VALUES ( 2, 2, 1, "TAP1045", 995);
INSERT INTO Trajeto VALUES ( 3, 1, 3, "TAP11" , 5864);
INSERT INTO Trajeto VALUES ( 4, 1, 4, "TAP1232", 3918);
INSERT INTO Trajeto VALUES ( 5, 5, 1, "TAP1261", 2752);
INSERT INTO Trajeto VALUES ( 6, 6, 1, "TAP1305", 2231);
INSERT INTO Trajeto VALUES ( 7, 1, 7, "TAP1511", 3802);
INSERT INTO Trajeto VALUES ( 8, 8, 18, "TAP1714", 1192);
INSERT INTO Trajeto VALUES ( 9, 1, 9, "TAP1823", 1556);
INSERT INTO Trajeto VALUES (10, 1, 10, "TAP203" , 5439);
INSERT INTO Trajeto VALUES (11, 1, 12, "TAP287" , 5787);
INSERT INTO Trajeto VALUES (12, 1, 13, "TAP31" , 5618);
INSERT INTO Trajeto VALUES (13, 15, 1, "TAP3383", 1670);
INSERT INTO Trajeto VALUES (14, 16, 16, "TAP367" , 1566);
INSERT INTO Trajeto VALUES (15, 1, 17, "TAP448" , 1439);
INSERT INTO Trajeto VALUES (16, 17, 1, "TAP449" , 1439);
INSERT INTO Trajeto VALUES (17, 17, 18, "TAP457" , 1201);
INSERT INTO Trajeto VALUES (18, 18, 17, "TAP458" , 1201);

INSERT INTO Voo VALUES (0, 12, "2016-04-25 14:34", "2016-04-25 17:38", 200);
INSERT INTO Voo VALUES (1, 4, "2016-04-25 19:28", "2016-04-26 02:08", 200);
INSERT INTO Voo VALUES (2, 0, "2016-04-25 19:20", "2016-04-25 19:15", 150);
INSERT INTO Voo VALUES (3, 3, "2016-04-25 13:19", "2016-04-25 16:49", 320);
INSERT INTO Voo VALUES (4, 10, "2016-04-25 13:03", "2016-04-25 15:13", 250);
INSERT INTO Voo VALUES (5, 11, "2016-04-26 13:13", "2016-04-26 20:54", 140);
INSERT INTO Voo VALUES (6, 13, "2016-04-25 18:40", "2016-04-25 19:45", 200);
INSERT INTO Voo VALUES (7, 18, "2016-04-23 19:45", "2016-04-23 22:35", 220);
INSERT INTO Voo VALUES (8, 16, "2016-04-25 20:44", "2016-04-25 22:00", 220);
INSERT INTO Voo VALUES (9, 17, "2016-04-25 22:48", "2016-04-25 23:26", 200);
INSERT INTO Voo VALUES (10, 15, "2016-04-25 18:18", "2016-04-25 21:25", 210);
INSERT INTO Voo VALUES (11, 14, "2016-04-25 20:01", "2016-04-25 22:20", 217);
INSERT INTO Voo VALUES (12, 1, "2016-04-25 19:10", "2016-04-25 21:01", 200);
INSERT INTO Voo VALUES (13, 5, "2016-04-25 18:40", "2016-04-25 19:45", 200);
INSERT INTO Voo VALUES (14, 2, "2016-04-25 19:18", "2016-04-25 20:36", 150);
INSERT INTO Voo VALUES (15, 6, "2016-04-25 18:57", "2016-04-25 21:04", 190);
INSERT INTO Voo VALUES (16, 7, "2016-04-25 18:40", "2016-04-25 19:45", 200);
INSERT INTO Voo VALUES (17, 9, "2016-04-25 17:06", "2016-04-25 21:21", 203);
INSERT INTO Voo VALUES (18, 8, "2016-04-25 19:20", "2016-04-25 20:55", 196);

INSERT INTO Localidade VALUES (0, 0, "Soito");
INSERT INTO CodigoPostal VALUES (0, 0, 6320631);
INSERT INTO Comprador VALUES (0, "Joao Silva", 4485088038682341, 271123321, 0);
INSERT INTO Reserva VALUES (0, NULL, 0);
INSERT INTO ReservaVoo VALUES (0, 0);

INSERT INTO Passageiro VALUES (0, "Joao Silva", 12312312, "1980-01-01", "390,
Estrada Municipal 538-1", 0, 0);
INSERT INTO ReservaPassageiro VALUES (0, 0);

```

```

INSERT INTO Detalhes VALUES (0, "34A", "2 207 365 3958 3309 0", 0);
INSERT INTO Mala VALUES (0, 0, 22, 40, 20, 10);
INSERT INTO Animal VALUES (0, 0, 5);
INSERT INTO Animal VALUES (1, 0, 7);
INSERT INTO DetalhesRefeicao VALUES (0, 1);
INSERT INTO PassageiroVooDetalhes VALUES (0, 0, 0);

INSERT INTO Localidade VALUES (1, 0, "Macieira de Sarnes");
INSERT INTO CodigoPostal VALUES (1, 1, 3700714);
INSERT INTO Comprador VALUES (1, "Ricardo Rodrigues", 4539675847475440, 234555123, 0);
INSERT INTO Reserva VALUES (1, NULL, 1);
INSERT INTO ReservaVoo VALUES (0, 1);
INSERT INTO ReservaVoo VALUES (1, 1);

INSERT INTO Passageiro VALUES (1, "Ricardo Rodrigues", 32132132, "1970-06-23", "47, Travessa 1o de Dezembro", 0, 1);
INSERT INTO Passageiro VALUES (2, "Sara Costa", 23123123, "1975-09-05", "47, Travessia 1o de Dezembro", 0, 1);
INSERT INTO Passageiro VALUES (3, "Carlos Rodrigues", 33322211, "1999-04-20", "47, Travessia 1o de Dezembro", 0, 1);
INSERT INTO ReservaPassageiro VALUES (1, 1);
INSERT INTO ReservaPassageiro VALUES (2, 1);
INSERT INTO ReservaPassageiro VALUES (3, 1);

INSERT INTO Detalhes VALUES (1, "22A", "5 321 503 2103 0569 2", 0);
INSERT INTO Mala VALUES (1, 1, 24, 60, 40, 20);
INSERT INTO Animal VALUES (2, 1, 4);
INSERT INTO DetalhesRefeicao VALUES (1, 0);
INSERT INTO PassageiroVooDetalhes VALUES (0, 1, 1);

INSERT INTO Detalhes VALUES (2, "22B", "5 321 503 2103 0569 3", 0);
INSERT INTO Mala VALUES (3, 2, 13, 50, 30, 25);
INSERT INTO DetalhesRefeicao VALUES (2, 3);
INSERT INTO PassageiroVooDetalhes VALUES (0, 2, 2);

INSERT INTO Detalhes VALUES (3, "22C", "5 321 503 2103 0569 4", 0);
INSERT INTO DetalhesRefeicao VALUES (3, 0);
INSERT INTO PassageiroVooDetalhes VALUES (0, 3, 3);

INSERT INTO Detalhes VALUES (4, "36B", "8 842 038 7309 3050 5", 0);
INSERT INTO Mala VALUES (2, 4, 24, 60, 40, 20);
INSERT INTO Animal VALUES (3, 4, 4);
INSERT INTO DetalhesRefeicao VALUES (4, 0);
INSERT INTO PassageiroVooDetalhes VALUES (1, 1, 4);

```



```

INSERT INTO Detalhes VALUES (5, "36A", "8 842 038 7309 3050 4", 0);
INSERT INTO Mala VALUES (4, 5, 13, 50, 30, 25);
INSERT INTO DetalhesRefeicao VALUES (5, 3);
INSERT INTO PassageiroVooDetalhes VALUES (1, 2, 5);

INSERT INTO Detalhes VALUES (6, "2A", "8 842 038 7309 3050 3", 1);
INSERT INTO DetalhesRefeicao VALUES (6, 0);
INSERT INTO PassageiroVooDetalhes VALUES (1, 3, 6);


INSERT INTO Localidade VALUES (2, 0, "PORTELA");
INSERT INTO Localidade VALUES (3, 0, "SACAVEM");
INSERT INTO CodigoPostal VALUES (2, 2, 2685181);
INSERT INTO CodigoPostal VALUES (3, 3, 2685894);
INSERT INTO Comprador VALUES (2, "Antonio Amorim", 4556024191973791, 210456456, 0);
INSERT INTO Reserva VALUES (2, NULL, 2);
INSERT INTO ReservaVoo VALUES (5, 2);
INSERT INTO ReservaVoo VALUES (6, 2);


INSERT INTO Passageiro VALUES (4, "Antonio Amorim", 88899911, "1967-06-07", "234, Impasse I", 0, 3);
INSERT INTO Passageiro VALUES (5, "Armando Vieira", 44477744, "1954-03-20", "832, Beco I", 0, 2);
INSERT INTO ReservaPassageiro VALUES (4, 2);
INSERT INTO ReservaPassageiro VALUES (5, 2);


INSERT INTO Detalhes VALUES (7, "1A", "5 843 792 9732 2983 4", 1);
INSERT INTO Mala VALUES (5, 7, 30, 40, 30, 20);
INSERT INTO DetalhesRefeicao VALUES (7, 2);
INSERT INTO PassageiroVooDetalhes VALUES (5, 4, 7);


INSERT INTO Detalhes VALUES (8, "2A", "5 843 792 9732 2983 5", 1);
INSERT INTO Mala VALUES (6, 8, 31, 40, 30, 20);
INSERT INTO DetalhesRefeicao VALUES (8, 2);
INSERT INTO PassageiroVooDetalhes VALUES (5, 5, 8);


INSERT INTO Detalhes VALUES (9, "1A", "8 230 832 0482 3043 1", 1);
INSERT INTO Mala VALUES (7, 9, 30, 40, 30, 20);
INSERT INTO DetalhesRefeicao VALUES (9, 2);
INSERT INTO PassageiroVooDetalhes VALUES (6, 4, 9);


INSERT INTO Detalhes VALUES (10, "2A", "8 230 832 0482 3043 2", 1);
INSERT INTO Mala VALUES (8, 10, 31, 40, 30, 20);
INSERT INTO DetalhesRefeicao VALUES (10, 2);
INSERT INTO PassageiroVooDetalhes VALUES (6, 5, 10);

```

Conclusão

Neste projeto chegou-se à conclusão que o desenvolvimento de uma base de dados e algoritmo para calcular o preço das suas reservas não é uma tarefa simples e exige uma extensa pré-análise de modo a que as decisões tomadas sejam as mais corretas. Existem várias formas de estruturar a base de dados, que foram consideradas até chegarmos ao modelo atual. A maximização do lucro num dado voo foi um problema difícil de resolver, e pode ser facilmente melhorado com acesso à base de dados de uma companhia aérea.

Referências

International Air Transport Association. "IATA Codes". iata.org.

<http://www.iata.org/services/Pages/codes.asp> (acedido a 19 de março de 2016).

TAP Portugal. "TAP Portugal". flytap.com.

<http://www.flytap.com> (acedido a 19 de março de 2016)