



Universidade do Minho

Escola de Engenharia

Mestrado Integrado Engenharia Informático

Unidade Curricular de Bases de Dados

Ano Lectivo de 2015/2016

Sistema de Base de Dados de uma companhia aérea

Grupo 03:

A68876 – Beatriz Loureiro

A67687 – José Santos

A70676 – Marcos Luís

A70644 – Ruben Santos

Janeiro, 2015

Data de Recepção	
Responsável	
Avaliação	
Observações	

Sistema de Base de Dados de uma companhia aérea

Grupo 03:

A68876 – Beatriz Loureiro

A67687 – José Santos

A70676 – Marcos Luís

A70644 – Ruben Santos

Janeiro, 2015

Resumo

Foi definido como objetivo deste trabalho a elaboração de um sistema de base de dados, capaz de gerir a informação acerca de uma companhia aérea, fornecendo informações sobre os componentes que constituem um voo.

Este relatório descreve o que foi feito na construção deste sistema, nomeadamente o levantamento de requisitos, a modelação do seu esquema conceptual, do seu esquema lógico e por fim o físico.

Área de Aplicação: Análise de requisitos, desenho e arquitetura de Sistemas de Bases de Dados.

Palavras-Chave: Requisitos, Relacionamentos, Esquema conceptual, Companhia Aérea, Avião, Voo, Passageiro, Base de Dados, modelo lógico, modelo físico, relações, transações, triggers, normalização, integridade, vistas, MySQL.

Índice

1. Introdução	1
1.1. Contextualização	1
1.2. Apresentação do Caso de Estudo	1
1.3. Motivação e Objetivos	2
1.4. Estrutura do Relatório	3
2. Requisitos	4
3. Modelação Concretual	5
3.1. Identificação dos tipos de entidade	5
3.2. Identificação dos tipos de relacionamento	5
3.3. Identificação e associação dos atributos com entidades ou relacionamentos	7
3.4. Identificação dos domínios dos atributos	10
3.5. Descrição dos tipos de entidades	13
3.6. Esquema Conceptual	13
4. Modelo Lógico	15
4.1. Passagem do modelo de dados conceptual local para um seu correspondente modelo de dados lógico	15
4.2. Validação do modelo de dados lógico através da normalização	18
4.3. Validação do modelo com as transações do utilizador - elaboração do correspondente mapa de transações	22
4.4. Elaboração e validação do esquema lógico da base de dados	23
4.5. Regras de integridade	23
4.6. Definição do tamanho inicial da base de dados e análise do seu crescimento futuro	25
4.7. Revisão do modelo lógico final com os futuros utilizadores do sistema de bases de dados	26
5. Modelo Físico	27
5.1. Tradução do modelo lógico para um SGBD e consequente implementação	27
5.1.1 Implementação das relações base	27
5.1.2 Implementação do Esquema Físico - Povoação	33
5.1.3 Restrições Gerais	41
5.2. Desenho da representação física - esquema físico	42
5.3. Análise de transações	42
5.4. Estimativa dos requisitos do espaço em disco agora com base no SGBD escolhido.	45
5.5. Definição das vistas dos utilizadores e regras de acesso	53
6. Conclusões e Trabalho Futuro	57
7. Ferramentas utilizadas	59

Bibliografia	60
Lista de Siglas e Acrónimos	61

Índice de Figuras

Figura 1 - Relação entre Voo e Piloto	5
Figura 2 - Relação entre Avião e Voo	5
Figura 3 - Relação entre Voo e Passageiro	6
Figura 4 - Relação entre Voo e Bagagem	6
Figura 5 - Relação entre Passageiro e Bagagem	6
Figura 6 - Versão mais recente do esquema conceptual	14
Figura 7- Esquema Lógico	17
Figura 8 - 1FN (Avião)	18
Figura 9 - 1FN (Voo)	19
Figura 10 - 1FN (Piloto)	19
Figura 11 - 1FN (Bagagem)	19
Figura 12 - 1FN MV (Piloto)	20
Figura 13 - 1FN MV (Passageiro)	20
Figura 14 - Relação Avião	28
Figura 15 - Relação Voo	29
Figura 16 - Relação Passageiro	29
Figura 17 - Relação Telemóvel	30
Figura 18 - Relação NacionalidadePassageiro	30
Figura 19 - Relação Piloto	31
Figura 20 - Relação Bagagem	31
Figura 21 - Relação VooPiloto	32
Figura 22 - Relação VooPassageiro	32
Figura 23 - Relação VooBagagem	33
Figura 24 - Gráfico do espaço ocupado pelo nº de aviões	46
Figura 25 - Gráfico do espaço ocupado pelo nº de pilotos	48
Figura 26 - Gráfico espaço ocupado por voos, passageiros e bagagens	52
Figura 27 - Versão 1 do Esquema conceptual	66
Figura 28 - Versão 2 do Esquema Conceptual	66

Índice de Tabelas

Tabela 1 - Multiplicidade das novas tabelas	24
Tabela 2 - Tamanho ocupado pelos atributos da Tabela Avião	45
Tabela 3 - Crescimento futuro e espaço ocupado pelo nº de aviões	46
Tabela 4 - Tamanho ocupado pelos atributos da tabela Piloto	47
Tabela 5 - Crescimento futuro e espaço ocupado pelo nº de pilotos	47
Tabela 6 - Tamanho ocupado pelos atributos da tabela NacionalidadePiloto	48
Tabela 7 - Tamanho ocupado pelos atributos da tabela Voo	48
Tabela 8 - Crescimento futuro e espaço ocupado pelo nº de Voos	49
Tabela 9 Tamanho ocupado pelos atributos da tabela VooPiloto	49
Tabela 10 -Tamanho ocupado pelos atributos da tabela Passageiro	50
Tabela 11 - Crescimento futuro e espaço ocupado pelo nº de Passageiros	50
Tabela 12 - Tamanho ocupado pelos atributos da tabela Telemovel	50
Tabela 13 - Tamanho ocupado pelos atributos da tabela NacionalidadePassageiro	51
Tabela 14 - Tamanho ocupado pelos atributos da tabela VooPassageiro	51
Tabela 15 - Tamanho ocupado pelos atributos da tabela Bagagem	51
Tabela 16 - Crescimento futuro e espaço ocupado pelo nº de Bagagens	52
Tabela 17 - Tamanho ocupado pelos atributos da tabela VooBagagem	52
Tabela 18 - Dicionário de Dados dos relacionamento	63
Tabela 19 - Dicionário de dados das entidades	63
Tabela 20 - Dicionário de dados dos atributos das entidades	65

1. Introdução

1.1. Contextualização

Uma companhia aérea é uma empresa que presta serviços na área do transporte aéreo de passageiros e mercadorias, podendo ter operações regulares, sazonais ou esporádicas. A primeira companhia aérea surgiu na Alemanha no início do século XX, abrindo caminho para centenas de novas companhias e a um novo mercado que atualmente é bastante lucrativo e competitivo. A aviação foi-se tornando num meio de transporte popular devido à rapidez das viagens e a sua comodidade em relação aos outros meios de transporte.

Excluindo os voos militares, os voos privados e voos exclusivos de carga, atualmente são feitos em média 102465 [1¹] voos diariamente, e em 2014 foram transportados 3 bilhões de passageiros, havendo cerca de 365 companhias aéreas dedicadas ao transporte comercial de passageiros.

A Delta Airlines, uma companhia norte-americana, transportou em 2014 cerca de 129 milhões de passageiros tornando-a na maior companhia aérea mundial e possui na sua frota 1,280 aeronaves.

1.2. Apresentação do Caso de Estudo

Neste trabalho focamo-nos apenas na aviação comercial, deixando de parte a aviação privada, de carga e militar.

Para melhor entender o conceito deste problema apresentamos como exemplo a TAP, que é a maior companhia aérea portuguesa, operando, em média, cerca de 2500 voos por semana. Dispõe de uma moderna frota de 77 aviões: 61 aeronaves de modelo Airbus, às quais se juntam mais 16 aeronaves que operam com as cores da PGA (Portugália Airlines). Todos os voos realizados em nome da companhia são minuciosamente controlados, desde o estado de cada avião que é regularmente examinado, aos dados dos passageiros que fizeram certo voo.

¹ [1]<http://www.garfors.com/2014/06/100000-flights-day.html>

Assim sendo, decidimos implementar uma pequena base de dados correspondente a uma companhia aérea de média dimensão que nos permitirá obter informação referida a um específico voo. Para tal, foi necessário criar uma lista de requisitos que conterá todos os dados que concordamos serem essenciais para a construção da nossa base de dados.

Para cada avião é indispensável armazenar o seu número, o combustível, a sua designação (serie, modelo, ano), a capacidade de cada classe e a data da última inspeção. Adicionalmente, um voo executado por um avião tem o seu próprio número de identificação, o dia e as horas de partida e chegada, os preços de cada classe, a origem e o destino da viagem. À necessidade de controlar as informações do voo, juntam-se os pilotos que pilotam o avião, que são identificados com o número, estatuto, experiência, nacionalidade, morada (país, cidade, rua, código postal) e salário. A cada passageiro que decide comprar um bilhete para determinado voo, é atribuído um numero, a classe, nacionalidade, telemóvel, nome e idade do individuo. Usualmente, os passageiros transportam bagagem com os seus pertences que decidimos também controlar. A bagagem é identificada por um número específico, o seu nível de fragilidade, a descrição, peso e dimensões (comprimento, altura, largura).

1.3. Motivação e Objetivos

Tendo em conta a contextualização acima apresentada, sabemos que uma companhia aérea, sendo de tamanho médio/grande, é um sistema de um elevado grau de complexidade de gerir. É fundamental ter informação completa, atualizada e sempre disponível para uma gestão de recursos eficiente e que seja lucrativa para uma empresa, sendo uma base de dados bem construída um dos pilares de funcionamento das companhias aéreas.

Posto isto, na elaboração da lista de requisitos deparamo-nos com inúmeras possibilidades de entidades e relacionamentos, confirmando assim a complexidade esperada.

A liberdade que disponha-mos para criar o nosso sistema tornou a escolha deste caso bastante apelativa. A capacidade de reunir, observar e tratar dados sobre vários componentes do funcionamento de uma empresa deste tipo é igualmente atrativo. Num aspecto pessoal, o campo da aviação é do interesse dos elementos do grupo.

O principal objetivo desta base de dados é modelar um protótipo de um sistema que permita a visualização de informações relativamente a uma companhia aérea. Para modelar a base de dados será necessário fazer um estudo dos principais componentes de uma companhia aérea como por exemplo dos pilotos da mesma. A companhia aérea em questão será de carácter fictício não sendo assim utilizados nenhuns dados reais.

1.4. Estrutura do Relatório

Este relatório descreve detalhadamente, o processo de estudo e conceção do trabalho prático, que se resume no levantamento de requisitos e respetiva modelação conceptual, lógica e física. A metodologia usada acompanha a metodologia do livro *Database Systems – A Practical Approach to Design, Implementation and Management*, 4ª Edição (Conolly e Begg, 2005).

O primeiro capítulo contém a exposição do problema através da sua contextualização. Segue-se, depois, a apresentação do caso de estudo e a motivação/objetivos que nos levaram a escolher este tema para o nosso trabalho.

No segundo capítulo enunciamos como procedemos ao levantamento de requisitos.

No terceiro, introduzimos o modelo conceptual definindo detalhadamente as fases enunciadas na metodologia. Sendo assim subdividimos em: Identificação dos tipos de entidade, Identificação dos tipos de relacionamento, Identificação e associação dos atributos com entidades ou relacionamentos, identificação dos domínios dos atributos, descrição dos tipos de entidades e por fim esquema conceptual.

Na quarta fase, fizemos a transição do modelo conceptual para o modelo lógico, aperfeiçoando a estrutura inicial até esta ficar solidificada, incluindo todas fases a ela associada.

No quinto capítulo, que foi a fase final do projeto, procedemos à implementação física da base de dados, apresentando o Sistema de Gestão de Bases de Dados escolhido e também alguns exemplos de transações.

No sexto capítulo expomos as ferramentas utilizadas para cada vertente deste trabalho desde o relatório até às ferramentas de desenvolvimento de um SGBD.

Por fim, apresentámos as nossas conclusões e o trabalho futuro.

2. Requisitos

Para a realização de levantamento de requisitos, após sermos contactados para elaborar uma base de dados de uma companhia aérea, foi necessária uma deslocação até aos locais. Para juntamente com as entidades deste meio obter uma certa veracidade nos termos e requisitos necessários para prestar de forma coerente o serviço solicitado. Após uma reunião com o cliente para discussão sobre cada ponto indispensável e relevante que pretendíamos para satisfazer os objetivos da base de dados a que nos propomos. Deste modo, obtivemos inúmeros requisitos que de seguida são nomeados e explicados.

O funcionamento de uma companhia aérea requer a cooperação tanto dos seus pilotos e aviões, como dos seus passageiros.

Cada avião pode efetuar várias viagens desde que os horários dos voos não coincidam. Como tal, o mesmo voo não pode ser efetuado por mais que um avião. Uma entidade importante a acrescentar seria o piloto, que exerce a função de pilotar um avião num voo até ao destino previsto. Normalmente o número requisitado de pilotos por avião são 2 (piloto e copiloto) para voos de pequenas distâncias e 3-4 para voos de longas distâncias ou ultralargas distâncias. Assim sendo, vários pilotos participam em vários voos.

Quanto ao passageiro, está também disponível a possibilidade de participar em vários voos e por consequência em vários aviões, havendo, pois, a necessidade de controlar o número de clientes que entra num avião. Cada passageiro tem opção de não levar bagagem ou levar uma ou mais bagagens consigo. Portanto, podemos afirmar que um passageiro pode transportar várias bagagens, mas uma bagagem pertence apenas a um passageiro. Por fim, visto que um passageiro usualmente viaja acompanhado de bagagem, surge assim a necessidade de criar um relacionamento de várias bagagens para vários aviões.

3. Modelação Conceitual

3.1. Identificação dos tipos de entidade

Após a análise dos requisitos anteriormente mencionados, foi possível nomear e concluir que as entidades em baixo apresentadas são as essenciais para este projeto:

- Avião
- Voo
- Piloto
- Passageiro
- Bagagem

3.2. Identificação dos tipos de relacionamento

Nesta fase, depois de já termos identificado entidades procedemos à identificação dos relacionamentos existentes entre as mesmas. No anexo 1, na tabela 1 apresentamos o dicionário de dados relativo aos atributos das entidades.

- A entidade Voo, é uma entidade crucial no nosso modelo uma vez que esta se relaciona com todas as outras entidades. Há um relacionamento entre a entidade Voo e a entidade Piloto, pois um piloto pode realizar vários voos e cada voo é efetuado por vários pilotos então cardinalidade deste relacionamento é N:M.



Figura 1 - Relação entre Voo e Piloto

- A entidade Voo relaciona-se também com a entidade Avião na perspectiva em que um avião pode realizar vários voos e cada voo é obrigatoriamente efetuado exatamente por um avião, assim sendo, a relação entre as entidades avião e voo tem cardinalidade 1: N.

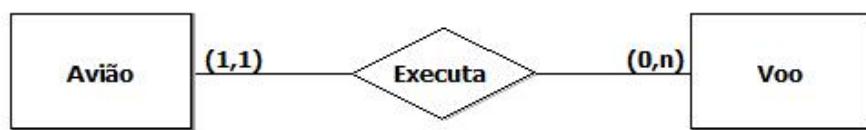


Figura 2 - Relação entre Avião e Voo

- A entidade Voo relaciona-se ainda com a entidade Passageiro com cardinalidade N:M, dando a possibilidade que um passageiro possa usufruir de vários voos e num voo participam vários passageiros, mas ainda há possibilidade de existir um voo sem passageiros.

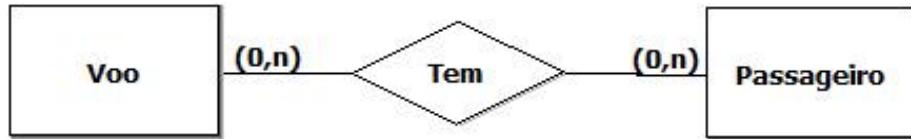


Figura 3 - Relação entre Voo e Passageiro

- Por último, entre a entidade voo e a entidade bagagem há um relacionamento com cardinalidade N:M, uma vez que num voo transporta várias bagagens e estas podem ser transportadas em vários voos.



Figura 4 - Relação entre Voo e Bagagem

- A entidade Passageiro relaciona-se com a entidade Bagagem, assumindo que um passageiro tem a opção de carregar uma bagagem e esta tem de estar obrigatoriamente associada a um passageiro, apresentando assim cardinalidade 1:N.

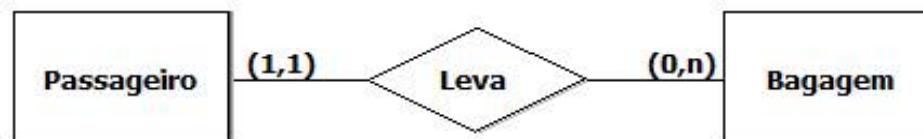


Figura 5 - Relação entre Passageiro e Bagagem

3.3. Identificação e associação dos atributos com entidades ou relacionamentos

Esta fase, seguimos para a caracterização das entidades, focando-nos na atribuição e na descrição dos atributos.

Desta forma, a entidade **avião** é caracterizada pelos seguintes atributos:

- **Nr. Identificador:** número único de cada avião servirá para o identificar na base de dados, sendo então chave primária desta entidade.
- **Ano de fabrico:** representa o ano em que o avião foi fabricado.
- **Nr de série:** representa a que serie o avião corresponde.
- **Modelo:** representa o respetivo modelo do avião.
- **Nr de lugares de primeira classe:** representa o número total de passageiros que a primeira classe pode acomodar.
- **Nr de capacidade de lugares de segunda classe:** representa o número total de passageiros que a segunda classe pode acomodar.
- **Nr de capacidade de lugares de terceira classe:** representa o número total de passageiros que a terceira classe pode acomodar.
- **Data da sua ultima inspeção:** representa a data da última vez que o avião foi inspecionado permitindo assim saber quanto tempo e até quando o avião pode voar ate ter que ser revisto novamente, verificando se a aeronave está apta para realizar um voo.
- **Nível de combustível atual:** representa a atual quantidade de combustível sabendo assim se será necessário adicionar combustível para a realização do próximo voo e também qual a quantidade que restou do voo anterior.

Ex: O avião 121 do ano 2011 sendo este um Boeing 747-8 tendo classe económica, business e primeira classe, com capacidade de 8 passageiros, 80 passageiros e 298 passageiros respetivamente, tendo sido revisto pela última vez a 10/15 e com nível de combustível em reserva.

De seguida, apresentamos os atributos da entidade **Voo**:

- **Nr identificador:** atribuição dada pela companhia aérea para rastreio de cada voo sendo este um atributo único.
- **Data de partida:** cada voo tem início num determinado dia.
- **Data de chegada:** cada voo tem como data de chegada um determinado dia.
- **Hora de partida:** cada voo tem associado uma hora de partida.
- **Hora de chegada:** cada voo tem uma hora de chegada associada.
- **Preço:** cada voo tem vários tipos de lugares cada um com o seu respetivo preço.
- **Local de origem:** país e respetivo aeroporto no qual vai ser o local de partida do avião.
- **Local de chegada:** país e respetivo aeroporto no qual vai ser o local de chegada do avião.
- **Quantidade:** Número de passageiros que participam no voo.
- **Custo operacional:** Custo base de se realizar o respetivo voo.

Ex: O voo 17 com partida no Porto no aeroporto Francisco Sá Carneiro prevista para 15/7/2015 às 18:00h tem como destino Londres no aeroporto London Heathrow com chegada prevista às 21:00h do dia 15/7/15. Tem bilhetes disponíveis ao preço de 100 euros em classe económica, 200 euros em classe business e 400 euros em primeira classe.

A entidade **Piloto** tem os seguintes atributos:

- **Nr identificador:** atribuição dada pela companhia aérea para rastreio do historial de cada piloto que faz parte da companhia.
- **Estatuto:** nível de competências do piloto em questão.
- **Nome:** nome completo do piloto.

- **Experiência:** número de horas de voo que um piloto já exerceu.
- **Nacionalidade:** é um atributo multi-valor uma vez que o piloto pode ter mais que uma nacionalidade.
- **Morada:** pais, cidade, rua e código postal do local de residência fixa do piloto.
- **Salário:** salário mensal.

Ex: O voo 17 terá como um dos pilotos, o piloto Miguel Serafim da Costa de nacionalidade Portuguesa com residência em Portugal, Lisboa Rua Dr. Carlos Lobo Oliveira 35 4ºesq. 4900-444, exercendo a função de copiloto, tendo 1700 horas completas de voo e com salário mensal de 5000 euros.

Os atributos que caracterizam a entidade **Passageiro** são:

- **Nr identificador:** atribuição dada pela companhia aérea para cada passageiro.
- **Classe:** o passageiro em questão irá numa determinada classe.
- **Nome:** nome completo do passageiro.
- **Telemóvel:** é um atributo multivalor, na medida em que um passageiro pode ter mais que um número de telemóvel para onde pode ser contactado no caso de um eventual contratempo.
- **Nacionalidade:** atributo multivalor uma vez que o passageiro pode ter mais que uma nacionalidade.
- **Idade:** idade do passageiro para fins estatísticos.

Ex: No voo 17 encontra-se o passageiro com nome Sandra Maria Moreira, de 35 anos com nacionalidade portuguesa e contacto 914325465 e voa em 1ª Classe.

Por último, a entidade **Bagagem** é especificada por os seguintes atributos:

- **Nr identificador:** atribuição dada pela companhia aérea para cada bagagem.
- **Fragilidade:** nível de fragilidade de cada bagagem.

- **Descrição:** consiste em saber o tipo de carga.
- **Peso:** o peso de bagagem em questão.
- **Dimensões:** altura, comprimento e largura.

Ex: A bagagem com a identificação OPO15HW760, sendo uma mala de porão, com fragilidade reduzida.

A mala foi despachada no check-in com 34 Kg e as suas dimensões são 50x20x60 cm.

No anexo 1, na tabela 3 apresentamos o dicionário de dados relativo aos atributos das entidades.

3.4. Identificação dos domínios dos atributos

Na entidade Avião:

- **Nr:** Caracterizado por 4 dígitos de 0 a 9 sem ordem definida.
Ex: '0021';
- **Combustível:** No máximo 4 dígitos de 0 a 9, que representa centenas de litros;
Ex: '5' que corresponde a 500 litros;
- **Série:** Combinação de caracteres alfanuméricos, com os dois primeiros caracteres correspondentes aos dois primeiros caracteres da marca que o fabricou; Ex: 'AI11031' para uma aeronave fabricada pela Airbus;
- **Modelo:** Combinação de caracteres alfanuméricos agrupados no máximo em 4 e separados por um '-'; Ex: 'A330-300' para um Airbus deste modelo;
- **Ano:** Ano de fabrico do tipo data; Ex: '1993';
- **1^a classe/2^a classe/3^a classe:** Caracterizado por no máximo 3 caracteres, que representam a capacidade do avião em cada classe; Ex: '50', para número de lugares de 2^a classe que o avião possui;
- **Inspeção:** Atributo do tipo data que indica a data em que é necessário realizar a próxima inspeção; Ex: '2015-12-23' para uma inspeção a fazer nesta data;

Na entidade Voo:

- **Nr:** Caracterizado por no máximo 3 caracteres, que abreviam a companhia em questão, separados de outros 3 caracteres por um ‘-’; Ex: ‘TP-032’ para um voo da TAP Portugal;
- **HoraPartida/HoraChegada:** Atributo do tipo tempo; Ex: ‘19:09’;
- **DataPartida/DataChegada:** Atributo do tipo data; Ex: ‘2015-12-30’;
- **Preço 1ªclasse/ Preço 2ªclasse/ Preço 3ªclasse:** Composto por um número que pode não ser inteiro; Ex: ‘212,12’ para um bilhete que custou 212 euros e 12 cêntimos;
- **Origem/Destino:** Caracterizado pelo código do aeroporto, com 3 caracteres, separado por um ‘-’ do nome completo do aeroporto; Ex: ‘MCT- Muscat International Airport’;
- **QuantidadePassg.:** Número inteiro de no máximo 3 dígitos; Ex: ‘213’ para um voo com 213 passageiros;
- **CustoOperacional:** Número que pode não ser inteiro com no máximo 8 dígitos; Ex: ‘15345,20’ para um voo com um custo de realização de 15345 euros e 20 cêntimos;

Na entidade Passageiro:

- **Nr:** Caraterizado por 4 dígitos de 0 a 9 sem ordem definida. Ex: 0011;
- **Nacionalidade:** Combinação de caracteres de tamanho até 75, que representam um país. Ex: “Portugal”;
- **Telemóvel:** Combinação de caracteres alfanuméricos que representam um número, com o primeiro dígito como sendo o carater ‘+’. Ex:”+351911111111”;
- **Nome:** Combinação de caracteres de tamanho até 75, que representam o nome do passageiro. Ex: “José Santos”;
- **Idade:** Caraterizado como inteiro que representa a idade do passageiro em anos. Ex:22;

Na entidade Bagagem:

- **Nr:** Conjunto de caracteres até 75 que permite um número identificador único para cada bagagem. Ex: OPO15H0240;
- **Fragilidade:** Conjunto de caracteres até 75, que permite descrever a fragilidade da bagagem. Ex: "Muito frágil";
- **Descrição:** Combinação de caracteres até 75, que permite descrever a bagagem. Ex: "Mala cor rosada";
- **Peso:** Caraterizado por valor que pode conter casas decimais, que representa o peso da bagagem em Kg. Ex:15.5;
- **Comprimento:** Caraterizado por valor que pode conter casas decimais, que representa o comprimento da bagagem em cm. Ex:117;
- **Largura:** Caraterizado por valor que pode conter casas decimais, que representa a largura da bagagem em cm. Ex:64;
- **Altura:** Caraterizado por valor que pode conter casas decimais, que representa a altura da bagagem em cm.Ex: 89;

Na entidade Piloto:

- **Nr:** Caraterizado por 4 dígitos de 0 a 9 sem ordem definida. Ex: 0099;
- **Nome:** Combinação de caracteres de tamanho até 75, que representam o nome do passageiro. Ex: "José Santos";
- **Estatuto:** Combinação de caracteres de tamanno até 75, que representa o estatuto que o piloto tem. Ex:"Comandante";
- **Experiência:** Numero inteiro, máximo 2 digitos que carateriza os anos de experiência. Ex:20;
- **País/Cidade/Rua:** Combinação de carateres de tamanho até 75, que representam respetivamente, o país, a cidade e a rua. Ex: "Portugal", "Porto", "Rua da Cernelha";

- **Cod_Postal:** Combinação de caracteres de tamanho até 75, que representa o código postal do piloto. Ex: "4007-007";
- **Salario:** Caraterizado por valor que pode conter casas decimais, que carateriza o valor que o piloto aufera mensalmente em euros. Ex: 5000.00;
- **Nacionalidade:** Combinação de caracteres de tamanho até 75, que representam um país. Ex: "Portugal";

No anexo 1, na tabela 3 apresentamos o dicionário de dados relativo aos atributos das entidades.

3.5. Descrição dos tipos de entidades

Nesta secção procedemos à descrição das entidades, no entanto no anexo 1 está exposta a tabela 2 que é referente ao dicionário de dados das entidades.

- A entidade avião serve para representar os inúmeros aviões que a companhia aérea possui.
- A entidade voo serve para representar os voos efetuados pelos seus transportes aéreos.
- A entidade piloto refere-se aos pilotos contratados pela companhia aérea para realizarem os voos.
- A entidade Passageiro caracteriza o nosso cliente, ou seja, quem usufrui dos serviços proporcionados pela companhia aérea.
- A entidade Bagagem refere-se à possibilidade de cada passageiro poder ou não transportar consigo a sua bagagem sendo esta também caracterizada e registada no sistema.

3.6. Esquema Conceptual

Dadas todas estas considerações iniciais, apresentamos agora, o modelo conceitual. Como é óbvio, estas considerações e o acompanhamento detalhado das várias fases da metodologia foram fundamentais na conceção do mesmo, pelo que sem elas, não seria possível compor o modelo conceptual que apresentamos de seguida, de modo a que fosse simples e percutível.

Nos anexos 2 encontram-se as versões mais antigas que mostram como o esquema conceitual foi evoluindo e aperfeiçoado ao longo do tempo em conjunto com o professor.

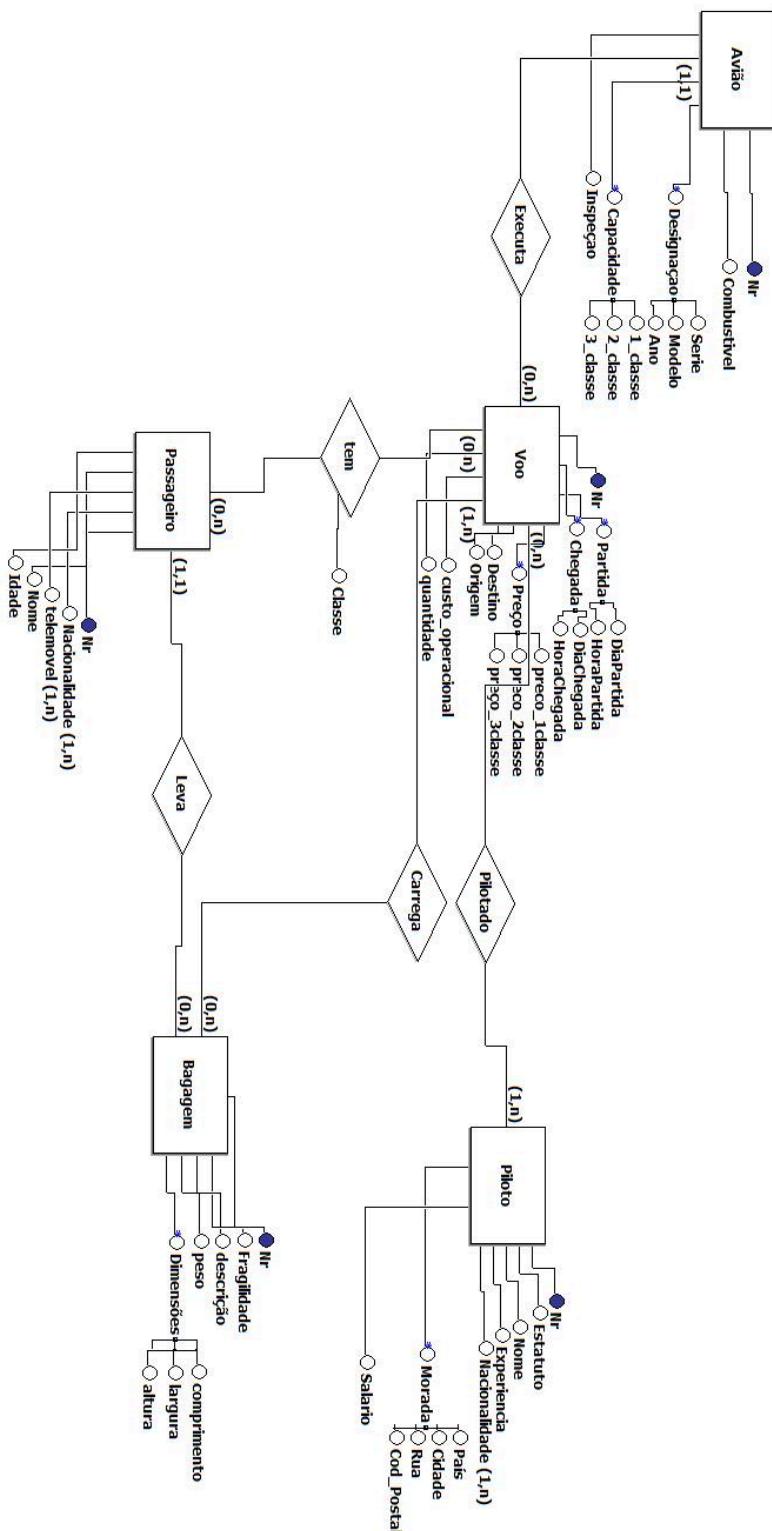


Figura 6 - Versão mais recente do esquema conceptual

4. Modelo Lógico

4.1. Passagem do modelo de dados conceptual local para um seu correspondente modelo de dados lógico

- **Derivar relações para o modelo lógico de dados**

4.1.1 Entidades fortes

Para cada entidade forte definida anteriormente, foi criada uma relação com os seus atributos, todos simples. Todas as nomes das relações começam com letra maiúscula e aos nomes dos atributos foram removidos os espaços e retirada a acentuação.

4.1.2 Entidades fracas

Neste modelo de dados não há nenhuma ocorrência de entidades fracas.

4.1.3 Relacionamentos binários 1:N

Para cada relacionamento 1:N, foi criada uma cópia da chave primária da relação-pai e colocada na relação-filho como chave estrangeira. Esta chave tem sempre o nome da relação-pai. Exemplos deste relacionamento são: Avião executa voo e Passageiro leva Bagagem.

4.1.4 Relacionamentos 1:1 e relacionamentos recursivos

Não existe nenhum destes relacionamentos no modelo.

4.1.5 Relacionamentos superclasse/subclasse

Não existe nenhum destes relacionamentos no modelo.

4.1.6 Relacionamentos N:M

Para cada relacionamento N:M, foi criada uma relação cujo nome é a concatenação dos nomes das relações presentes no relacionamento. De seguida, foram-lhes colocadas as cópias das chaves primárias das relações originais, criando assim dois relacionamentos 1:N. O relacionamento deste tipo são: Voo tem Passageiro, Voo carrega Bagagem e Voo é pilotado por Piloto.

4.1.7 Relacionamentos Complexos

Um relacionamento complexo é um relacionamento de grau igual ou superior a 3, ou seja, quando 3 ou mais entidades intervêm. No modelo relacional, estes casos devem ser decompostos de modo a encontrar uma entidade intermédia e que as entidades fiquem relacionadas com relacionamentos de grau 2. Neste modelo não temos nenhum relacionamento deste tipo.

4.1.8 Atributos Multi Valor

Cada um dos três atributos MV no modelo dão origem a uma relação cujo nome é o do atributo. Nestas relações, temos uma chave primária composta, colocando uma cópia da chave primária da relação à qual pertencem os atributos (ficando também como chave estrangeira) e ainda um atributo identificador. Os atributos MV no modelo são: nacionalidade e telemóvel em Passageiro; nacionalidade em Piloto.

• Esquema lógico

No seguimento deste processo iterativo de construção de uma base de dados, geramos agora o nosso modelo lógico de acordo com o modelo conceptual apresentado. Nesta etapa surgem as tabelas correspondentes às diversas entidades assim como dos vários tipos de relacionamentos. Através da conclusão desta fase, já teremos um modelo representativo dos requisitos do sistema.

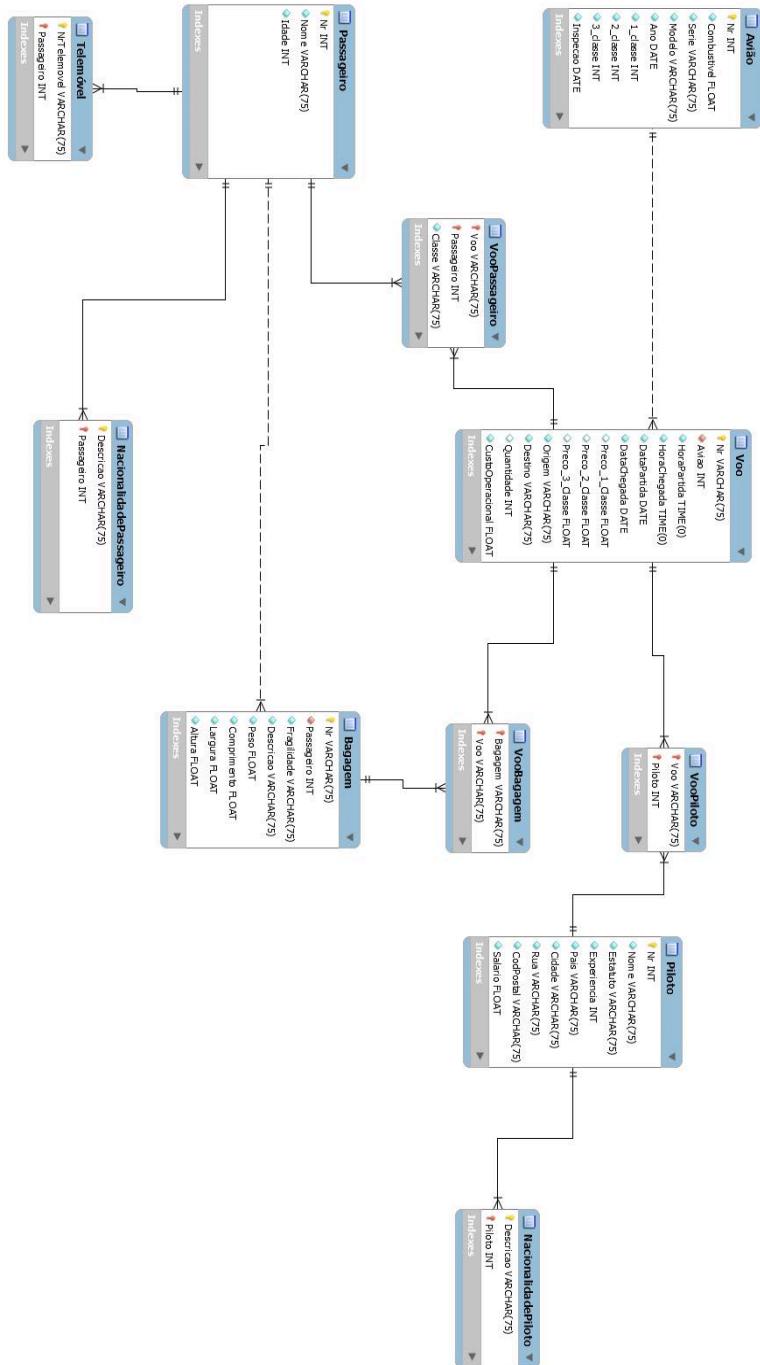


Figura 7- Esquema Lógico

4.2. Validação do modelo de dados lógico através da normalização

Na tradução do modelo conceptual para o modelo lógico existem, por vezes, algumas anomalias que podem tornar a base de dados inconsistente. Sendo assim procedemos à validação do nosso modelo através da normalização, que é caracterizada por ser um processo sequencial.

No nosso caso de estudo, iremos abordar as 3 primeiras formas, fundamentando as decisões tomadas.

• 1^a Forma Normal

A primeira forma normal verifica-se quando todos os valores de todos os atributos forem atómicos, isto é, se não for possível decompô-los. A relação em que a interseção de cada linha e coluna contém um e apenas um valor.

Primeiramente, analisámos as tabelas do modelo à procura de grupos repetidos, que existiriam em tabelas não normalizadas, mas concluímos que esta situação não se verifica, nesta fase identificámos a chave primária e os atributos MV e compostos.

Atributos compostos:

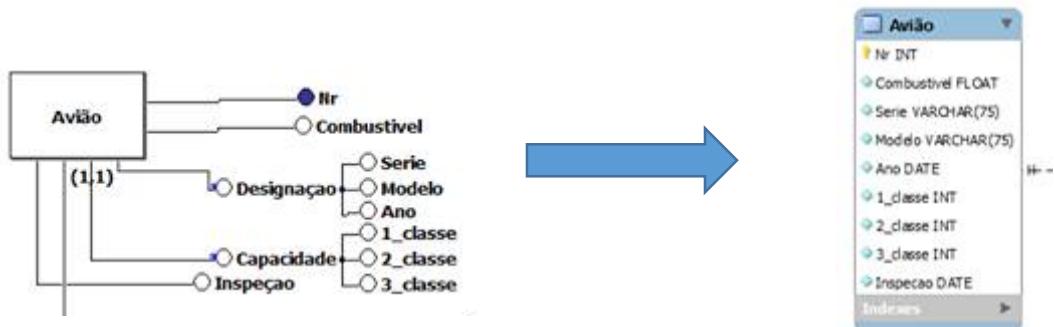


Figura 8 - 1FN (Avião)

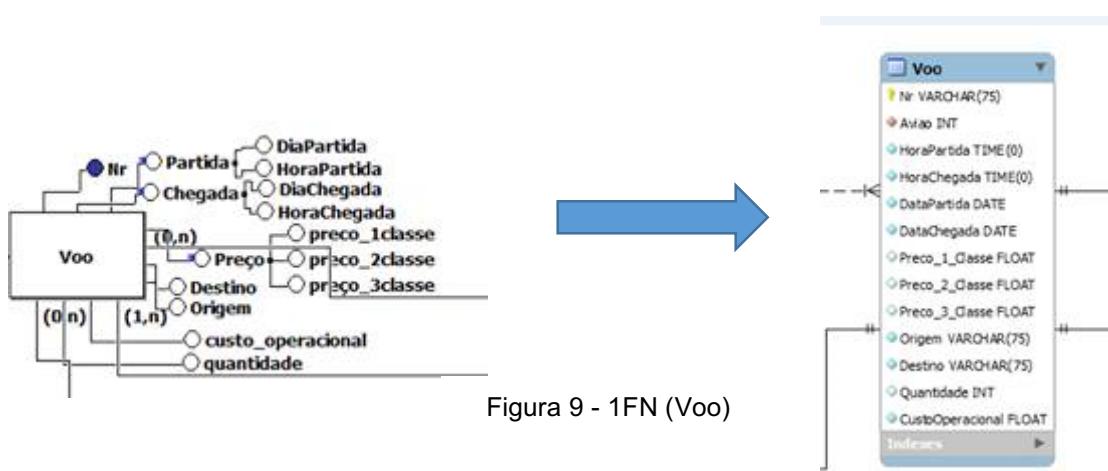


Figura 9 - 1FN (Voo)

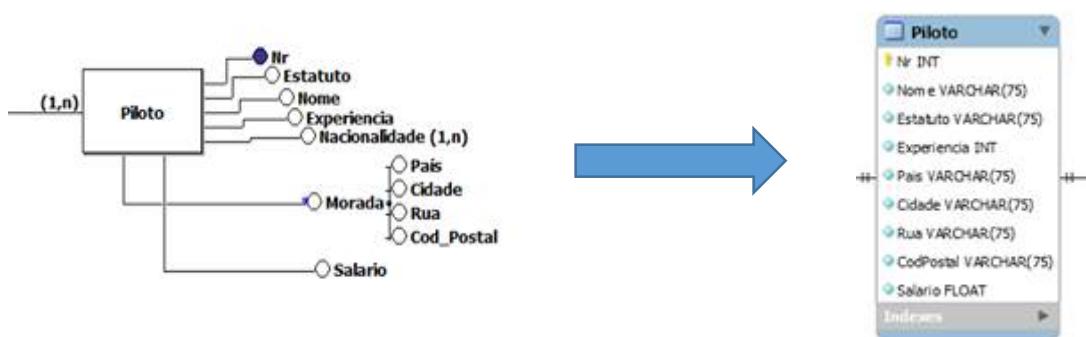


Figura 10 - 1FN (Piloto)



Figura 11 - 1FN (Bagagem)

Atributos Multi-valor:

No modelo conceptual existem duas entidades que possuem atributos MV: nacionalidade e telemóvel (entidade Passageiro) e nacionalidade (entidade Piloto). Na transição para o modelo lógico, o grupo teve em conta que os atributos MV dão origem a uma tabela adicional pelo que de imediato, também se confirma que é respeitada a primeira forma normal.

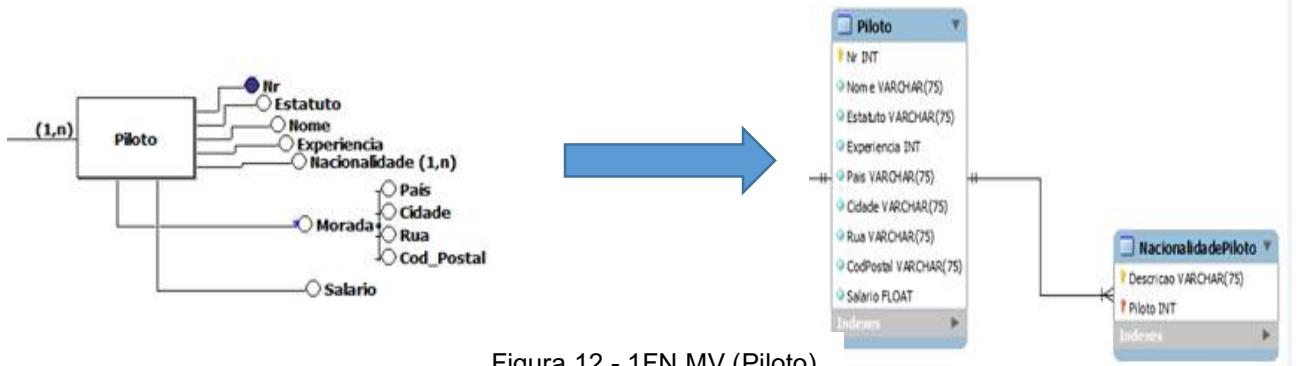


Figura 12 - 1FN MV (Piloto)

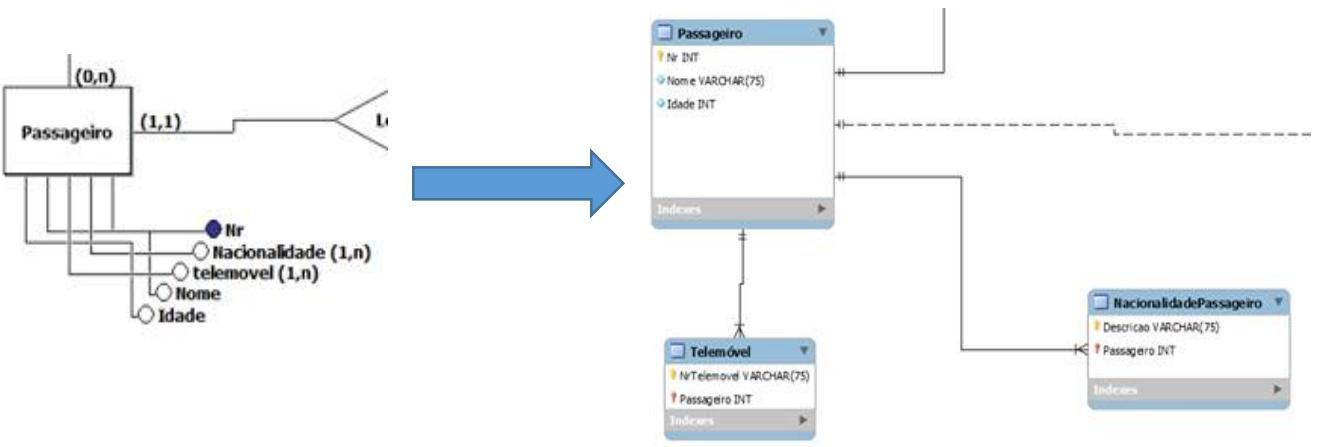


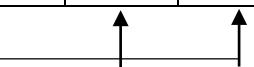
Figura 13 - 1FN MV (Passageiro)

■ 2^a Forma normal:

Na segunda forma normal, o objetivo é eliminar redundâncias e analisar as dependências parciais. A relação neste momento já se encontra na 1ºForma Normal e cada atributo que não pertença à chave primária depende totalmente desta e não apenas parcialmente. Neste caso a base de dados encontra-se já na 2ºForma Normal não havendo casos em que se verificam dependências parciais. De seguida temos dois exemplos em que se observa a dependência total: na primeira tabela correspondente a Passageiro os atributos 'Nome' e 'Idade' dependem da chave primaria 'Nr' verificando assim a 2ºForma Normal; já no segundo exemplo correspondente a tabela VooPassageiro observa-se que o atributo 'Classe' que não pertence a chave primária mas depende totalmente desta, que é composta por 'Voo' e 'Passageiro'.

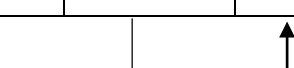
Passageiro

Nr	Nome	Idade



VooPassageiro

Voo	Passageiro	Classe



■ **3^a Forma normal**

O objetivo da 3^a forma é análise das dependências transitivas. Como exemplo observemos a tabela avião existe a capacidade correspondente a cada classe não havendo a necessidade de existir um atributo capacidade total pois pode ser calculada através da soma dos outros 3 atributos.

Nr	Combustivel	Serie	Modelo	Ano	1_classe	2_classe	3_classe	Inspecção



A 3FN é aquela em que normalmente termina o processo de normalização, contudo em alguns casos muito específicos, a 3FN ainda transporta alguns problemas. Nas relações em causa não consideramos que houvesse necessidade em prosseguir com o processo de normalização. No entanto, esta remete-nos assim para dois objetivos, em geral conflituosos: por um lado pretendem-se sistemas flexíveis, sem problemas de redundância; por outro lado exigem-se sistemas com alto desempenho. Torna-se, portanto, necessário chegar a um compromisso, o ideal seria um esquema equilibrado que nunca ponha em risco a integridade da base de dados, mas que ao mesmo tempo tenha um desempenho aceitável, pois só assim será utilizado.

4.3. Validação do modelo com as transações do utilizador - elaboração do correspondente mapa de transações

Neste projeto decidimos implementar três das mais importantes transações que existem no contexto desta base de dados, tendo consciência que posteriormente poderão ser desenvolvidas outras transações para aperfeiçoar o sistema. A primeira transação definida torna possível a inserção de um passageiro num determinado voo e numa determinada classe. Já a segunda transação permite à companhia aérea adicionar um voo e consequentemente designar um piloto para a realização do mesmo", por final foi implementada uma transação que insere uma nova bagagem e atribui-lhe um voo.

- **Inserir novo passageiro**

A primeira transação designada como "novo passageiro" começa por inserir um novo passageiro na tabela Passageiro incluído o seu Nr, nome e idade, de seguida insere também na tabela Telemóvel e na tabela NacionalidadePassageiro as informações pedidas. Por final para fazer corresponder este novo passageiro a um voo e a uma classe temos que inserir na tabela VooPassageiro o voo e a classe que pretende desfrutar.

- **Inserir novo Voo**

Nesta transação partimos do princípio que será a ação mais importante numa companhia aérea pois é o serviço que esta mesma presta. Para isso começamos por inserir o novo voo na tabela Voo destacando por exemplo o seu número, as horas e datas de chegada e partida, destino, entre outros. Por fim nomeamos um piloto que o irá executar e para isso apenas temos que recorrer a inserção na tabela VooPiloto o número do piloto correspondente.

- **Inserir novo bagagem**

Por fim foi criada uma transação que permite o registo de uma bagagem executando uma inserção na tabela Bagagem e posteriormente atribui-la a um voo correspondente, inserindo na tabela VooBagagem o número do voo.

Em anexo apresentamos um mapa no modelo lógico que corresponde as diferentes transações e as suas respetivas ações.

Na secção 5.3 apresentamos as transações implementadas em SQL.

4.4. Elaboração e validação do esquema lógico da base de dados

Com base no esquema lógico elaborado na figura 7 foi necessário proceder à validação do mesmo seguindo a metodologia específica apresentada no livro de bases de dados recomendado. Este parâmetro completa a informação já descrita no capítulo “Informação dos domínios dos atributos” e no dicionário de dados apresentado na explicação do esquema conceptual.

4.5. Regras de integridade

As restrições de integridade garantem uma base de dados completa, precisa e consistente. Para atingir tal objetivo é importante que se respeitem as regras que se seguem:

- **Necessidade de valores** – No desenvolvimento do dicionário de dados, alguns atributos têm, obrigatoriamente, que possuir um valor válido, ou seja, não nulo. Foi decidido que a maior parte dos atributos pertencentes ao modelo lógico tenham um valor válido, sendo estes importantes para controlar as informações dos voos, dos passageiros, etc.
De seguida, apresentamos alguns exemplos de atributos que, apesar de não serem chaves primárias, são considerados essenciais:
 - **Inspecção**: acedendo às informações de um determinado avião, é possível verificar a data da próxima inspecção para concluir que o mesmo está operacional.
 - **DataPartida**: atributo da entidade voo que indica a hora em que o avião partirá.
 - **Nome**: atributo da entidade passageiro que identifica o passageiro, não podendo assim ser nulo.
- **Restrições do domínio do atributo** – Todos os atributos têm de ter associado uma gama de valores que podem tomar. O domínio dos atributos pode ser consultado no anexo 1 (Tabela 3), referente aos dicionários de dados.
- **Multiplicidade** – Na passagem do modelo conceptual para o modelo lógico surgiram novos relacionamentos, como é o caso dos atributos multi-valor, onde passa a existir um relacionamento de cardinalidade 1:N com a tabela adicional. A nível dos relacionamentos N:M os relacionamentos com a nova tabela passam a ser ambos de 1:N.

Sendo assim apresentamos na tabela abaixo as novas tabelas e respetiva multiplicidade:

Tabela 1 - Multiplicidade das novas tabelas

Entidade	Multiplicidade	Multiplicidade	Entidade
Avião	1	n	Voo
Voo	1	n	VooPiloto
VooPiloto	n	1	Piloto
Piloto	1	n	NacionalidadePiloto
Voo	1	n	VooBagagem
VooBagagem	n	1	Bagagem
Voo	1	n	VooPassageiro
VooPassageiro	n	1	Passageiro
Passageiro	1	n	Bagagem
Passageiro	1	n	Telemóvel
Passageiro	1	n	NacionalidadePassageiro

- **Integridade da entidade** – Esta regra diz que a chave primária de uma entidade nunca pode assumir valores nulos. Pela análise do dicionário de dados em anexo conseguimos comprovar que esta regra está bem explícita.

- **Integridade referencial** – O valor de uma chave estrangeira numa relação “filho” deverá ser e existir na chave primária da relação “pai” a ela associada. Portanto, sempre que uma relação “pai” sofre uma alteração ou remoção, essa alteração tem que se refletir nas chaves estrangeiras às quais está ligada.

No caso da entidade voo, é indispensável que a chave estrangeira “avião” exista, pois não é possível efetuar um voo sem que seja atribuído um avião para o efetuar. Para qualquer outro caso do modelo lógico apresentado, decidimos que as chaves estrangeiras contendo um valor existente é mandatário.

Para o segundo problema de integridade referencial, usando as entidades Passageiro e Bagagem, é essencial considerar os seguintes casos:

- **Caso 1: Inserir um registo na relação filho** – Apenas se verifica se a chave estrangeira na relação filho têm alguma correspondência com o valor da chave primária na relação pai, ou seja, para assegurar a integridade referencial, verificamos que a chave estrangeira, Passageiro, do novo elemento da entidade Bagagem está definido como valor existente na entidade Passageiro.

- **Caso 2: Eliminar um registo da relação filho** – Uma remoção na relação filho não afeta a integridade em qualquer aspeto, ou seja, se for eliminado um tuplo da entidade Bagagem, a integridade referencial não é afetada.
- **Caso 3: Atualização o registo da chave estrangeira na relação filho** – O procedimento é totalmente igual ao de inserir um registo na relação filho, ou seja, verificar que o atributo passageiro, que é chave estrangeira do novo registo da tabela Bagagem, é um valor existente da tabela Passageiro.
- **Caso 4: Inserir um registo na relação pai** – Uma inserção na relação pai não interfere com a integridade referencial. Contudo de modo manter a informação atualizada e consistente é necessário proceder à atualização nas respetivas relações filho. Sendo assim, inserindo um registo na entidade Passageiro, não afeta a integridade do referencial. Simplesmente torna-se um passageiro sem Bagagem.
- **Caso 5: Remover registo na relação pai** – No caso da remoção de um registo na relação pai a integridade referencial pode ser posta em causa, na medida em que pode existir um registo numa relação filho sem uma entidade pai. Se um registo pertencente à entidade Passageiro é eliminado, a integridade referencial não é garantida, se existe um registo da entidade Bagagem que referece o determinado passageiro. A estratégia adotada para prevenir este tipo de acontecimento foi a NO ACTION, que previne a eliminação de um passageiro se o mesmo estiver associado a uma bagagem.

4.6. Definição do tamanho inicial da base de dados e análise do seu crescimento futuro

O tamanho inicial da nossa Base de Dados, ou seja, a nossa script de povoamento inicial para testes, contem as seguintes quantidades de registos:

Aviões: 15

Voos: 17

Pilotos: 17

Passageiros: 16

Bagagens: 15

Nacionalidade de Passageiros: 16

Nacionalidade de Pilotos: 17

Telemóveis de Passageiros: 16

Passageiros no Voo: 17

Pilotos nos Voo:17

Bagagens nos Voo: 21

No Capítulo 5 analisamos com mais pormenor o tamanho inicial da base de dados e estimamos o seu crescimento futuro assim como o espaço que ocupa em disco.

4.7. Revisão do modelo lógico final com os futuros utilizadores do sistema de bases de dados

Para finalizar esta fase, o modelo lógico deve ser revisto com o utilizador. Este processo é extremamente importante na medida em que o utilizador deve ser capaz de reconhecer, que o modelo concebido é uma representação real dos requisitos da empresa a ser modelada. Este passo é também uma forma de avaliar o trabalho feito face aos requisitos do utilizador.

Sendo assim, foi avaliada toda a documentação associada ao modelo lógico, ou seja, o dicionário de dados de entidades, relacionamentos e atributos e validar a conversão para o modelo relacional. Na revisão do modelo foi possível identificar todas as tabelas importantes no modelo e constatar com o cliente que correspondem aos objetos considerados importantes no problema e às suas necessidades. Na revisão do dicionário de relacionamentos foi justificada a forma como as relações estão associadas e o utilizador verificou que estes relacionamentos correspondem à realidade do problema. Já na revisão dos atributos o utilizador pode verificar mais concretamente a informação associada a cada relação e validar o domínio de cada atributo, constatando que estes correspondem aos requisitos anteriormente apresentados.

Por fim, na revisão geral do modelo lógico realizou-se a verificação de algumas das perguntas mais frequentes no problema de modo a testar as capacidades do mesmo. O modelo foi aprovado pelo cliente como sendo uma solução que dá resposta a todas as necessidades pretendidas.

5. Modelo Físico

Nesta secção realizamos a tradução do modelo lógico anteriormente apresentado para o modelo físico a ser implementado no SGBD.

Para a construção da Base de Dados autoproposta, foi usado como sistema de base de dados o MySQL Workbench da Oracle. A utilização deste sistema foi assumida uma vez que, nas aulas práticas da Unidade Curricular, foi usada esta solução, facilitando assim a implementação da nossa base de dados.

Para avançarmos com a implementação foi necessário refletir sobre as funcionalidades presentes no sistema de base de dados a usar. Das aulas práticas verificamos que o MySQL suporta o seguinte:

- Definição de chaves primárias e chaves estrangeiras;
- Definição do tipo de dados pretendido, isto é, o sistema permite que os atributos sejam definidos como NOT NULL;
- Restrições de integridade nas relações;
- Definição de restrições de integridade;

5.1. Tradução do modelo lógico para um SGBD e consequente implementação

A primeira fase da modelação física corresponde à tradução do modelo lógico para o SGBD que pretendemos implementar. Nela iniciamos com a implementação das relações base.

5.1.1 Implementação das relações base

O objetivo é decidir como representar as relações identificadas no modelo lógico no sistema de base de dados escolhido (MySQL).

Neste ponto, para cada relação apresentamos o domínio e a estrutura das relações como os valores por defeito, valores nulos e possíveis restrições, informação que pode também ser obtida através do dicionário de dados.

- Tabela Avião

Nr	[Número inteiro positivo não nulo]
Combustivel	[Número real positivo não nulo]
Serie	[Sequência de caracteres não nulo]
Modelo	[Sequência de caracteres não nulo]
Ano	[Variável do tipo data não nula]
1_classe	[Número inteiro positivo não nulo]
2_classe	[Número inteiro positivo não nulo]
3_classe	[Número inteiro positivo não nulo]
Inspecao	[Variável do tipo data não nulo]

```

CREATE TABLE IF NOT EXISTS `CompanhiaAerea`.`Avião` (
`Nr` INT NOT NULL,
`Combustivel` FLOAT NOT NULL,
`Serie` VARCHAR(75) NOT NULL,
`Modelo` VARCHAR(75) NOT NULL,
`Ano` Date NOT NULL,
`1_classe` INT NOT NULL,
`2_classe` INT NOT NULL,
`3_classe` INT NOT NULL,
`Inspecao` Date NOT NULL,
PRIMARY KEY (`Nr`))
ENGINE = InnoDB;

```

Figura 14 - Relação Avião

- Tabela Voo

Nr	[Sequência de caracteres não nula]
Aviao	[Número inteiro não nulo]
HoraPartida	[Variável de tipo tempo não nula]
HoraChegada	[Variável do tipo tempo não nula]
DataPartida	[Variável do tipo data não nula]
DataChegada	[Variável do tipo data não nula]
Preco_1_Classe	[Número real não nulo]
Preco_2_Classe	[Número real não nulo]
Preco_3_Classe	[Número real não nulo]
Origem	[Sequência de caracteres não nulo]
Destino	[Sequência de caracteres não nulo]
Quantidade	[Número inteiro]
CustoOperacional	[Número real não nulo]

```

CREATE TABLE IF NOT EXISTS `CompanhiaAerea`.`Voo`(
    `Nr` VARCHAR(75) NOT NULL,
    `Aviao` INT NOT NULL,
    `HoraPartida` TIME NOT NULL,
    `HoraChegada` TIME NOT NULL,
    `DataPartida` DATE NOT NULL,
    `DataChegada` DATE NOT NULL,
    `Preco_1_Classe` FLOAT,
    `Preco_2_Classe` FLOAT,
    `Preco_3_Classe` FLOAT,
    `Origem` VARCHAR(75) NOT NULL,
    `Destino` VARCHAR(75) NOT NULL,
    `Quantidade` INT,
    `CustoOperacional` FLOAT NOT NULL,
    PRIMARY KEY (`Nr`),
    CONSTRAINT `fk_Avião_Voo`
        FOREIGN KEY(`Aviao`)
        REFERENCES `CompanhiaAerea`.`Avião`(`Nr`)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

Figura 15 - Relação Voo

- **Tabela Passageiro**

Nr	[Número inteiro não nulo]
Nome	[Sequência de caracteres (máximo 75) não nula]
Idade	[Sequência de caracteres (máximo 75) não nula]

```

CREATE TABLE IF NOT EXISTS `CompanhiaAerea`.`Passageiro`(
    `Nr` INT NOT NULL,
    `Nome` VARCHAR(75) NOT NULL,
    `Idade` INT NOT NULL,
    PRIMARY KEY(`Nr`))
ENGINE = InnoDB;

```

Figura 16 - Relação Passageiro

- **Tabela Telemóvel**

Passageiro	[Número inteiro não nulo]
NrTelemovel	[Sequência de caracteres (máximo 75) não nula]

```
CREATE TABLE IF NOT EXISTS `CompanhiaAerea`.`Telemóvel`(
  `Passageiro` INT NOT NULL,
  `NrTelemovel` VARCHAR(75) NOT NULL,
  PRIMARY KEY(`Passageiro`, `NrTelemovel`),
  CONSTRAINT `fk_Telemovel_Passageiro`
    FOREIGN KEY(`Passageiro`)
    REFERENCES `CompanhiaAerea`.`Passageiro`(`Nr`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

Figura 17 - Relação Telemóvel

- **Tabela NacionalidadePassageiro**

Passageiro	[Número inteiro não nulo]
Descricao	[Sequência de caracteres (máximo 75) não nula]

```
CREATE TABLE IF NOT EXISTS `CompanhiaAerea`.`NacionalidadePassageiro`(
  `Passageiro` INT NOT NULL,
  `Descricao` VARCHAR(75) NOT NULL,
  PRIMARY KEY(`Passageiro`, `Descricao`),
  CONSTRAINT `fk_Nacionalidade_Passageiro`
    FOREIGN KEY(`Passageiro`)
    REFERENCES `CompanhiaAerea`.`Passageiro`(`Nr`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

Figura 18 - Relação NacionalidadePassageiro

- **Tabela Piloto**

Nr	[Número inteiro positivo não nulo]
Nome	[Sequência de caracteres (máximo 75) não nula]
Estatuto	[Sequência de caracteres (máximo 75) não nula]
Experiencia	[Número inteiro positivo não nulo]
Pais	[Sequência de caracteres (máximo 75) não nula]
Cidade	[Sequência de caracteres (máximo 75) não nula]

Rua	[Sequência de caracteres (máximo 75) não nula]
CodPostal	[Sequência de caracteres (máximo 75) não nula]
Salario	[Número real positivo não nulo]

```

CREATE TABLE IF NOT EXISTS `CompanhiaAerea`.`Piloto`(
  `Nr` INT NOT NULL,
  `Nome` VARCHAR(75) NOT NULL,
  `Estatuto` VARCHAR(75) NOT NULL,
  `Experiencia` INT NOT NULL,
  `Pais` VARCHAR(75) NOT NULL,
  `Cidade` VARCHAR(75) NOT NULL,
  `Rua` VARCHAR(75) NOT NULL,
  `CodPostal` VARCHAR(75) NOT NULL,
  `Salario` FLOAT NOT NULL,
  PRIMARY KEY(`Nr`)
)
ENGINE = InnoDB;

```

Figura 19 - Relação Piloto

- **Tabela Bagagem**

Nr	[Sequência de caracteres (máximo 75) não nula]
Fragilidade	[Sequência de caracteres (máximo 75) não nula]
Descricao	[Sequência de caracteres (máximo 75) não nula]
Peso	[Número real positivo não nulo]
Comprimento	[Número real positivo não nulo]
Largura	[Número real positivo não nulo]
Altura	[Número real positivo não nulo]
Passageiro	[Número inteiro positivo não nulo]

```

CREATE TABLE IF NOT EXISTS `CompanhiaAerea`.`Bagagem`(
  `Nr` VARCHAR(75) NOT NULL,
  `Fragilidade` VARCHAR(75),
  `Descricao` VARCHAR(75) NOT NULL,
  `Peso` FLOAT NOT NULL,
  `Comprimento` FLOAT NOT NULL,
  `Largura` FLOAT NOT NULL,
  `Altura` FLOAT NOT NULL,
  `Passageiro` INT NOT NULL,
  PRIMARY KEY(`Nr`),
  CONSTRAINT `fk_Bagagem_Passageiro1`
  FOREIGN KEY (`Passageiro`)
  REFERENCES `CompanhiaAerea`.`Passageiro`(`Nr`)
  ON DELETE NO ACTION
  ON UPDATE NO ACTION
)
ENGINE = InnoDB;

```

Figura 20 - Relação Bagagem

- **Tabela VooPiloto**

Voo [Sequência de caracteres não nula (máximo 75)]
 Piloto [Número inteiro não nulo]

```
CREATE TABLE IF NOT EXISTS `CompanhiaAerea`.`VooPiloto`(
    `Voo` VARCHAR(75) NOT NULL,
    `Piloto` INT NOT NULL,
    PRIMARY KEY(`Voo`, `Piloto`),
    CONSTRAINT `fk_VooPiloto_Piloto1`
        FOREIGN KEY (`Piloto`)
        REFERENCES `CompanhiaAerea`.`Piloto`(`Nr`)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION,
    CONSTRAINT `fk_VooPiloto_Voo1`
        FOREIGN KEY (`Voo`)
        REFERENCES `CompanhiaAerea`.`Voo`(`Nr`)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

Figura 21 - Relação VooPiloto

- **Tabela VooPassageiro**

Voo [Sequência de caracteres não nula (máximo 75)]
 Passageiro [Número inteiro não nulo]
 Classe [Sequência de caracteres (máximo 75) não nula]

```
CREATE TABLE IF NOT EXISTS `CompanhiaAerea`.`VooPassageiro`(
    `Voo` VARCHAR(75) NOT NULL,
    `Passageiro` INT NOT NULL,
    `Classe` VARCHAR(75),
    PRIMARY KEY(`Voo`, `Passageiro`),
    CONSTRAINT `fk_VooPassageiro_Voo1`
        FOREIGN KEY (`Voo`)
        REFERENCES `CompanhiaAerea`.`Voo`(`Nr`)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION,
    CONSTRAINT `fk_VooPassageiro_Passageiro1`
        FOREIGN KEY (`Passageiro`)
        REFERENCES `CompanhiaAerea`.`Passageiro`(`Nr`)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

Figura 22 - Relação VooPassageiro

- **Tabela VooBagagem**

Voo [Sequência de caracteres não nula (máximo 75)]
 Bagagem [Sequência de caracteres não nula (máximo 75)]

```
|CREATE TABLE IF NOT EXISTS `CompanhiaAerea`.`VooBagagem` (
    `Voo` VARCHAR(75) NOT NULL,
    `Bagagem` VARCHAR(75) NOT NULL,
    PRIMARY KEY(`Voo`, `Bagagem`),
    CONSTRAINT `fk_VooBagagem_Voo1`
        FOREIGN KEY (`Voo`)
        REFERENCES `CompanhiaAerea`.`Voo` (`Nr`)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION,
    CONSTRAINT `fk_VooBagagem_Bagagem1`
        FOREIGN KEY (`Bagagem`)
        REFERENCES `CompanhiaAerea`.`Bagagem` (`Nr`)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

Figura 23 - Relação VooBagagem

5.1.2 Implementação do Esquema Físico - Povoação

- **Tabela Avião**

```
INSERT INTO Aviao
(Nr, Combustivel, Serie, Modelo, Ano, 1_classe, 2_classe, 3_classe, Inspecao)
VALUES
(1,70,'CS-TOH', 'A330-223', '2007-06-20', 10,16,200, '2016-01-11'),
(2,70,'CS-TOI', 'A330-223', '2007-07-12', 10,14,200, '2016-02-12'),
(3,70,'CS-TQO', 'A330-303', '2014-05-12', 4,10,200, '2016-03-15'),
(4,90,'CS-TOD', 'A340-612', '1994-12-22', 15,8,250, '2016-02-23'),
(5,50,'CS-TJE', 'A321-211', '2000-10-01', 2,10,120, '2016-01-30'),
(6,45,'CS-TNT', 'A320-214', '2009-10-17', 0,10,90, '2016-12-27'),
(7,40,'CS-TTR', 'A219-111', '2008-11-23', 0,5,80, '2016-06-22'),
(8,40,'CS-TPK', 'ERJ-145ER', '1997-05-06', 0,20,50, '2016-07-07'),
(9,40,'CS-TPC', 'F100', '1991-10-24', 0,0,70, '2016-08-09'),
(10,25,'CS-TRU', 'ATR72-202', '2011-08-15', 0,0,70, '2016-04-10'),
(11,150,'A6-EEE', 'A380-861', '2012-12-17', 20,40,400, '2016-04-19'),
(12,100,'A6-EPF', 'B777-300ER', '2015-10-01', 10,20,300, '2016-03-15'),
(13,90,'A7-BCU', 'B787-800', '2011-10-01', 4,10,220, '2016-05-08'),
(14,95,'A7-ALG', 'A350-941', '2015-04-01', 5,30,247, '2016-11-02'),
(15,50,'N949AN', 'B737-823', '2008-10-01', 0,0,135, '2016-09-05');
```

Nr	Combustivel	Serie	Modelo	Ano	1_classe	2_classe	3_classe	Inspecao
1	70	CS-TOH	A330-223	2007-06-20	10	16	200	2016-01-11
2	70	CS-TOI	A330-223	2007-07-12	10	14	200	2016-02-12
3	70	CS-TOQ	A330-303	2014-05-12	4	10	200	2016-03-15
4	90	CS-TOD	A340-612	1994-12-22	15	8	250	2016-02-23
5	50	CS-TJE	A321-211	2000-10-01	2	10	120	2016-01-30
6	45	CS-TNT	A320-214	2009-10-17	0	10	90	2016-12-27
7	40	CS-TTR	A219-111	2008-11-23	0	5	80	2016-06-22
8	40	CS-TPK	ERJ-145ER	1997-05-06	0	20	50	2016-07-07
9	40	CS-TPC	F100	1991-10-24	0	0	70	2016-08-09
10	25	CS-TRU	ATR72-202	2011-08-15	0	0	70	2016-04-10
11	150	A6-EEE	A380-861	2012-12-17	20	40	400	2016-04-19
12	100	A6-EPF	B777-300ER	2015-10-01	10	20	300	2016-03-15
13	90	A7-BCU	B787-800	2011-10-01	4	10	220	2016-05-08
14	95	A7-ALG	A350-941	2015-04-01	5	30	247	2016-11-02
15	50	N949AN	B737-823	2008-10-01	0	0	135	2016-09-05

• Tabela Voo

INSERT INTO Voo
(Nr, Aviao, HoraPartida, HoraChegada, DataPartida, DataChegada,
Preco_1_Classe, Preco_2_Classe, Preco_3_Classe, Origem,
Destino, Quantidade, CustoOperacional)

VALUES
('TP-032', 1, '06:30:00.00', '08:30:00.00', '2016-01-12', '2016-01-12', 700, 328.40, 120.12, 'LIS-LISBON PORTELA AIRPORT', 'LHR-LONDON HEATHROW AIRPORT', 230, 30.300),
('TP-123', 3, '07:05:00.00', '10:40:00.00', '2016-01-15', '2016-01-15', 1200, 500.98, 130.12, 'OPO-PORTO INTERNATIONAL AIRPORT', 'SVO-MOSCOW SHEREMETYEVO INTERNATIONAL AIRPORT', 320, 50.500),
('TP-008', 4, '08:30:00.00', '11:20:00.00', '2016-01-17', '2016-08-17', 1500, 123.68, 150.50, 'CDG-PARIS CHARLES DE GAULLE AIRPORT', 'FAO-FARO AIRPORT', 320, 20.000),
('W2-130', 6, '08:35:00.00', '17:00:00.00', '2016-02-11', '2016-02-11', 2000, 400, 200.50, 'FNC-FUNCHAL MADEIRA AIRPORT', 'CCS-SIMÓN BOLÍVAR INTERNATIONAL AIRPORT', 110, 70.000),
('TP-070', 5, '09:10:00.00', '16:30:00.00', '2016-10-13', '2016-10-13', 800, 345.90, 170.90, 'BOS-BOSTON LOGAN INTERNATIONAL AIRPORT', 'PDL-JOÃO PAULO II AIRPORT', 90, 90.600),
('TP-005', 7, '09:26:00.00', '17:45:00.00', '2016-03-13', '2016-03-13', 700, 236.40, 90.50, 'OPO-PORTO INTERNATIONAL AIRPORT', 'ORD-CHICAGO OHARE INTERNATIONAL AIRPORT', 410, 100.000),
('TP-235', 9, '10:30:00.00', '19:30:00.00', '2016-02-15', '2016-02-15', 3974, 1230.70, 110.50, 'JFK-JOHN F. KENNEDY INTERNATIONAL AIRPORT', 'LIS-LISBON PORTELA AIRPORT', 160, 170.000),
('PG-726', 8, '11:45:00.00', '21:45:00.00', '2016-02-09', '2016-02-09', 6789, 2450.06, 175.50, 'LIS-LISBON PORTELA AIRPORT', 'LAX-LOS ANGELES INTERNATIONAL AIRPORT', 250, 48.000),
('SA-340', 10, '11:50:00.00', '15:00:00.00', '2016-02-02', '2016-02-02', 500, 200, 80.90, 'PDX-PORTO SANTO AIRPORT', 'MAD-ADOLFO SUÁREZ MADRID-BARAJAS AIRPORT', 234, 48.000),
('EK-060', 12, '12:00:00.00', '19:30:00.00', '2016-04-23', '2016-04-23', 3400, 980, 350.90, 'DXB-DUBAI INTERNATIONAL AIRPORT', 'LIS-LISBON PORTELA AIRPORT', 410, 90.000),
('QR-053', 11, '13:50:00.00', '21:50:00.00', '2016-03-30', '2016-03-30', 2345, 1230.34, 240.50, 'OPO-PORTO INTERNATIONAL AIRPORT', 'DOH-DHAH HAMAD INTERNATIONAL AIRPORT', 320, 110.000),
('TP-720', 14, '14:28:00.00', '23:49:00.00', '2016-02-28', '2016-02-28', 1700, 567, 160.50, 'GRU-SÃO PAULO-GUARULHOS INTERNATIONAL AIRPORT', 'FAO-FARO AIRPORT', 154, 115.400),
('FR-100', 13, '16:30:00.00', '00:30:00.00', '2016-11-15', '2016-11-16', 1345, 723.70, 190.90, 'OPO-PORTO INTERNATIONAL AIRPORT', 'GIG-RIO DE JANEIRO-GALEÃO INTERNATIONAL AIRPORT', 329, 90.900),
('TP-930', 3, '17:40:00.00', '20:00:00.00', '2016-02-13', '2016-02-13', 2367, 563.45, 230.50, 'AMS-AMSTERDAM AIRPORT SCHIPHOL', 'LIS-LISBON PORTELA AIRPORT', 156, 40.000),
('AF-450', 15, '18:35:00.00', '21:50:00.00', '2016-02-29', '2016-02-29', 3467, 604.90, 260.90, 'FCO-LEONARDO DA VINCI-FIUMICINO AIRPORT', 'OPO-PORTO INTERNATIONAL AIRPORT', 234, 30.400),
('LH-550', 12, '20:00:00.00', '02:30:00.00', '2016-02-16', '2016-02-17', 1983, 403.40, 270.50, 'IST-ISTANBUL ATATÜRK AIRPORT', 'PDL-JOÃO PAULO II AIRPORT', 158, 40.000),
('TP-050', 1, '22:55:00.00', '03:25:00.00', '2016-02-18', '2016-02-19', 1845, 503.09, 280.90, 'FAO-FARO AIRPORT', 'MXP-MILAN-MALPENSA AIRPORT', 290, 20.000);

Nr	Aviao	HoraPartida	HoraChegada	DataPartida	DataChegada	Preco_1_Classe	Preco_2_Classe	Preco_3_Classe	Origem	Destino	Quantidade	CustoOperacional
AF-450	15	18:35:00	21:50:00	2016-02-29	2016-02-29	3467	604.9	260.9	FCO-LEONARDO DA VINCI-FIUMICINO AIRPORT	OPO-PORTO INTERNATIONAL AIRPORT	234	30.4
EK-060	12	12:00:00	19:30:00	2016-04-23	2016-04-23	3400	980	350.9	DXB-DUBAI INTERNATIONAL AIRPORT	LIS-LISBON PORTELA AIRPORT	410	90
FR-100	13	16:30:00	00:30:00	2016-11-15	2016-11-16	1345	723.7	190.9	OPO-PORTO INTERNATIONAL AIRPORT	GIG-RIO DE JANEIRO-GALEÃO INTERNATIONAL AIRPORT	329	90.9
LH-550	12	20:00:00	02:30:00	2016-02-16	2016-02-17	1983	403.4	270.5	IST-ISTANBUL ATATÜRK AIRPORT	PDL-JOÃO PAULO II AIRPORT	158	40
PG-726	8	11:45:00	21:45:00	2016-02-09	2016-02-09	6789	2450.06	175.5	LIS-LISBON PORTELA AIRPORT	LAX-LOS ANGELES INTERNATIONAL AIRPORT	250	170
QR-053	11	13:50:00	21:50:00	2016-03-30	2016-03-30	2345	1230.34	240.5	OPO-PORTO INTERNATIONAL AIRPORT	DOH-DHAH HAMAD INTERNATIONAL AIRPORT	320	110
SA-340	10	11:50:00	15:00:00	2016-02-02	2016-02-02	500	200	80.9	PDX-PORTO SANTO AIRPORT	MAD-ADOLFO SUÁREZ MADRID-BARAJAS AIRPORT	234	40
TP-005	7	09:20:00	17:45:00	2016-03-13	2016-03-13	700	236.4	90.5	OPO-PORTO INTERNATIONAL AIRPORT	ORD-CHICAGO OHARE INTERNATIONAL AIRPORT	410	100
TP-008	4	08:30:00	11:20:00	2016-08-17	2016-08-17	1500	123.6	150.5	CDG-PARIS CHARLES DE GAULLE AIRPORT	FAO-FARO AIRPORT	320	20
TP-032	1	06:30:00	08:30:00	2016-01-12	2016-01-12	700	320.4	120.12	LIS-LISBON PORTELA AIRPORT	LHR-LONDON HEATHROW AIRPORT	230	30.3

- **Tabela Passageiro**

```
INSERT INTO Passageiro
(Nr, Nome, Idade)
VALUES
(2368, 'Alberto Fernandez Mola' ,57),
(4971, 'Joana Empadão Cunha' ,49),
(6861, 'James Roberto Affleck' ,08),
(9581, 'Bernardo Silva Culatra' ,07),
(2294, 'André Inocêncio Batman' ,65),
(8596, 'Thomas Müller Carrey' ,03),
(9610, 'Claudio Marchisio Mola' ,08),
(6007, 'Patrícia Castro Zê' ,67),
(8009, 'Joana Costa Carrey' ,30),
(0542, 'Ana Sofia Spears' ,32),
(2993, 'Patrícia Castro Silva' ,52),
(0188, 'Claudio Marchisio Sá' ,15),
(4617, 'John McMany Carrey' ,47),
(8603, 'Angela Merkel Santos' ,78),
(2443, 'Angela Merkel Rato' ,44),
(0233, 'Claudio Marchisio Rato' ,27),
(4896, 'Marcos Luís Guimarães' ,39),
(3979, 'Célia Natália Sá' ,46),
(4179, 'Maria Albertina Culatra' ,68),
(3264, 'Célia Natália Rocha' ,70),
(1911, 'José Gabriel MacLess' ,63),
(8887, 'André Inocêncio Affleck' ,01),
(6177, 'João Miguel Tone' ,24),
(1455, 'José Gabriel Carrey' ,47),
(3415, 'João Miguel Rocha' ,44),
(7129, 'João Francesinhas Bola' ,48);
```

Nr	Nome	Idade
188	Claudio Marchisio Sá	15
233	Claudio Marchisio Rato	27
542	Ana Sofia Spears	32
1455	José Gabriel Carrey	47
1911	José Gabriel MacLess	63
2294	André Inocêncio Batman	65
2368	Alberto Fernandez Mola	57
2443	Angela Merkel Rato	44
2993	Patrícia Castro Silva	52
3264	Célia Natália Rocha	70
3415	João Miguel Rocha	44
3979	Célia Natália Sá	46
4179	Maria Albertina Culatra	68
4617	John McMany Carrey	47
4896	Marcos Luís Guimarães	39
4971	Joana Empadão Cunha	49
6007	Patrícia Castro Zê	67
6177	João Miguel Tone	24
6861	James Roberto Affleck	8
7129	João Francesinhas Bola	48
8009	Joana Costa Carrey	30
8596	Thomas Müller Carrey	3
8603	Angela Merkel Santos	78
8887	André Inocêncio Affleck	1
9581	Bernardo Silva Culatra	7
9610	Claudio Marchisio Mola	8

- Tabela Telemóvel

```
INSERT INTO Telemovel
(Passageiro, NrTelemovel)
VALUES
(2368, '+351968764321'),
(4971, '+351938827544'),
(4971, '+351933327544'),
(6861, '+351423684863'),
(9581, '+351333383216'),
(2294, '+351504424206'),
(8596, '+351047845108'),
(9610, '+351857876864'),
(6007, '+351777412537'),
(8009, '+351682064818'),
(0542, '+351537330201'),
(2993, '+351532255358'),
(2993, '+351653714665'),
(0188, '+351241724307'),
(4617, '+351478858232'),
(8603, '+351311307046'),
(2443, '+351457335713'),
(2443, '+351466203102'),
(0233, '+351758840856'),
(4896, '+351353676761'),
(3979, '+351018010104'),
(4179, '+351704418738'),
(3264, '+351452802406'),
(1911, '+351352608352'),
(1911, '+351760364468'),
(8887, '+351062825718'),
(6177, '+351806566737'),
(1455, '+351375458301'),
(3415, '+351145648744'),
(3415, '+351877618788'),
(7129, '+351751408371');
```

Passageiro	NrTelemovel
188	+351241724307
233	+351758840856
542	+351537330201
1455	+351375458301
1911	+351352608352
1911	+351760364468
2294	+351504424206
2368	+351968764321
2443	+351457335713
2443	+351466203102
2993	+351532255358
2993	+351653714665
3264	+351452802406
3415	+351145648744
3415	+351877618788
3979	+351018010104
4179	+351704418738
4617	+351478858232
4896	+351353676761
4971	+351933327544
4971	+351938827544
6007	+351777412537
6177	+351806566737
6861	+351423684863
7129	+351751408371
8009	+351682064818
8596	+351047845108
8603	+351311307046
8887	+351062825718
9581	+351333383216
9610	+351857876864

- **Tabela NacionalidadePassageiro**

```

CREATE TABLE IF NOT EXISTS `CompanhiaAerea`.`NacionalidadePassageiro` (
    `Passageiro` INT NOT NULL,
    `Descricao` VARCHAR(75) NOT NULL,
    PRIMARY KEY(`Passageiro`,`Descricao`),
    CONSTRAINT `fk_Nacionalidade_Passageiro`
    FOREIGN KEY(`Passageiro`)
    REFERENCES `CompanhiaAerea`.`Passageiro`(`Nr`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
) ENGINE = InnoDB;

```

Passageiro	Descricao
188	Alemã
188	Holandesa
233	Americana
542	Portuguesa
1455	Portuguesa
1911	Albanesa
1911	Espanhola
1911	Grega
2294	Italiana
2368	Portuguesa
2443	Espanhola
2443	Portuguesa
2443	Sul Africana
2993	Alemã
2993	Americana
3264	Brasileira
3415	Alemã
3415	Checa
3415	Eslovaca
3979	Holandesa
3979	Sueca
4179	Espanhola
4617	Italiana
4896	Italiana
4971	Inglesa
4971	Portuguesa
6007	Americana
6177	Polaca
6177	Ucraniana
6861	Brasileira
6861	Portuguesa
7129	Portuguesa
8009	Portuguesa
8596	Portuguesa
8603	Portuguesa
8887	Francesa
9581	Americana
9610	Francesa
9610	Portuguesa

- **Tabela Piloto**

```

INSERT INTO Piloto
(Nr, Nome, Estatuto, Experiencia, Pais, Cidade, Rua, CodPostal, Salario)
VALUES
(23, 'Ruben Santos', 'Comandante', 12000, 'Portugal', 'Viana do Castelo', 'Rua do Céu', '4925-074', 12.345),
(34, 'Luís Silva', 'Co-Piloto', 1000, 'Portugal', 'Lisboa', 'Rua das Zemites', '4002-074', 5.000),
(3, 'José Santos', 'Comandante', 15000, 'Portugal', 'Porto', 'Rua Moinho de Vento', '4500-044', 13.345),
(45, 'Marcos Luís', 'Comandante', 12500, 'Portugal', 'Viana do Castelo', 'Rua Fábrica do Pêlo', '4400-074', 13.500),
(56, 'Gonçalo Guedes', 'Co-Piloto', 2000, 'Portugal', 'Faro', 'Rua João Allen', '4300-074', 2.345),
(6, 'Bernardo Silva', 'Co-Piloto', 3200, 'Portugal', 'Coimbra', 'Rua da Bandeira', '4700-074', 1.345),
(7, 'Alberto Martínez', 'Comandante', 9600, 'Espanha', 'Barcelona', 'Calle de las Comas', '93747', 11.345),
(43, 'Beatriz Loureiro', 'Co-Piloto', 6050, 'Portugal', 'Braga', 'Rua Nova', '4203-076', 3.345),
(20, 'Jean Pierre', 'Comandante', 11563, 'França', 'Lyon', 'Oliveress', '12458', 10.345),
(10, 'Wolfgang Stark', 'Comandante', 25050, 'Alemanha', 'Munich', 'Schübert', '83874', 12.345),
(17, 'João Almeida', 'Co-Piloto', 3089, 'Portugal', 'Setúbal', 'Rua da Pista', '4890-123', 2.345),
(12, 'Kevin G. Hayden', 'Co-Piloto', 7456, 'Estados Unidos', 'Austin,TX', '23rd Street', '90210', 4.345),
(2, 'Oscar Mendonza', 'Comandante', 19987, 'Paraguai', 'Assunção', 'Calle de las Flores', '38494', 12.345),
(14, 'Raheem Al Said', 'Co-Piloto', 1010, 'Jordânia', 'Amman', 'Mintaq Al Gar', '12345', 1.345),
(15, 'Horácio Gomes', 'Co-Piloto', 2245, 'Portugal', 'Lisboa', 'Rua Gomes Almeida', '5648-082', 9.345),
(70, 'Pedro Castro', 'Comandante', 13694, 'Portugal', 'Porto', 'Rua São Bento', '4648-082', 12.345),
(65, 'Kevin Lahore', 'Co-Piloto', 4694, 'South Africa', 'Durban', 'Freedom Street', '24213', 3.345);

```

Nr	Nome	Estatuto	Experiencia	País	Cidade	Rua	CodPostal	Salario
2	Oscar Mendonza	Comandante	19987	Paraguai	Assunção	Calle de las Flores	38494	12.345
3	José Santos	Comandante	15000	Portugal	Porto	Rua Moinho de Vento	4500-044	13.345
6	Bernardo Silva	Co-Piloto	3200	Portugal	Coimbra	Rua da Bandeira	4700-074	1.345
7	Alberto Martínez	Comandante	9600	Espanha	Barcelona	Calle de las Comas	93747	11.345
10	Wolfgang Stark	Comandante	25050	Alemanha	Munich	Schübert	83874	12.345
12	Kevin G. Hayden	Co-Piloto	7456	Estados Unidos	Austin, TX	23rd Street	90210	4.345
14	Raheem Al Said	Co-Piloto	1010	Jordânia	Amman	Mintaq Al Gar	12345	1.345
15	Horácio Gomes	Co-Piloto	2245	Portugal	Lisboa	Rua Gomes Almeida	5648-082	9.345
17	João Almeida	Co-Piloto	3089	Portugal	Setúbal	Rua da Pista	4890-123	2.345
20	Jean Pierre	Comandante	11563	França	Lyon	Oliveress	12458	10.345
23	Ruben Santos	Comandante	12000	Portugal	Viana do...	Rua do Céu	4925-074	12.345
34	Luís Silva	Co-Piloto	1000	Portugal	Lisboa	Rua das Zemites	4002-074	5
43	Beatriz Loureiro	Co-Piloto	6050	Portugal	Braga	Rua Nova	4203-076	3.345
45	Marcos Luís	Comandante	12500	Portugal	Viana do...	Rua Fábrica do Pêlo	4400-074	13.5
56	Gonçalo Guedes	Co-Piloto	2000	Portugal	Faro	Rua João Allen	4300-074	2.345
65	Kevin Lahore	Co-Piloto	4694	South Africa	Durban	Freedom Street	24213	3.345
70	Pedro Castro	Comandante	13694	Portugal	Porto	Rua São Bento	4648-082	12.345

- Tabela NacionalidadePiloto**

```
INSERT INTO NacionalidadePiloto
(Piloto, Descricao)
VALUES
(23, 'Portuguesa'),
(34, 'Portuguesa'),
(3, 'Inglesa'),
(3, 'Irlandesa'),
(45, 'Portuguesa'),
(56, 'Brasileira'),
(56, 'Portuguesa'),
(6, 'Americana'),
(7, 'Italiana'),
(43, 'Portuguesa'),
(43, 'Angolana'),
(20, 'Portuguesa'),
(10, 'Francesa'),
(10, 'Luxemburguesa'),
(17, 'Americana'),
(12, 'Portuguesa'),
(2, 'Portuguesa'),
(2, 'Moçambicana'),
(14, 'Americana'),
(15, 'Alemã'),
(70, 'Alemã'),
(70, 'Argelina'),
(65, 'Holandesa');
```

Piloto	Descricao
2	Moçambicana
2	Portuguesa
3	Inglesa
3	Irlandesa
6	Americana
7	Italiana
10	Francesa
10	Luxemburguesa
12	Portuguesa
14	Americana
15	Alemã
17	Americana
20	Portuguesa
23	Portuguesa
34	Portuguesa
43	Angolana
43	Portuguesa
45	Portuguesa
56	Brasileira
56	Portuguesa
65	Holandesa
70	Alemã
70	Argelina

- **Tabela Bagagem**

```
INSERT INTO Bagagem
(Nr, Fragilidade, Descricao, Peso, Comprimento, Largura, Altura, Passageiro)
VALUES
('OPO15H0240', 'Reduzida', 'Mala de Porão', 32.5, 40, 30, 50, 2368),
('LIS15LA240', 'Mediana', 'Prancha de Surf', 10, 120, 10, 40, 4971),
('LIS16LA240', 'Reduzida', 'Mala de Porão', 20, 40, 30, 50, 6861),
('JFK16F1212', 'Reduzida', 'Mala de Porão', 15, 40, 30, 50, 6861),
('BOS16A2113', 'Elevada', 'Animal de Estimação', 5, 20, 12, 10, 9581),
('CCS16C4569', 'Reduzida', 'Mala de Porão', 15, 40, 30, 50, 2294),
('MAD16V4568', 'Reduzida', 'Mala de Porão', 25, 40, 30, 50, 8596),
('PDL16N3574', 'Mediana', 'Animal de Estimação', 5, 20, 12, 10, 6007),
('DXB1600023', 'Reduzida', 'Mala de Porão', 20, 40, 30, 50, 8603),
('LIS16L2034', 'Reduzida', 'Mala de Porão', 20, 40, 30, 50, 4896),
('AMS16M0056', 'Elevada', 'Instrumento Musical', 12, 20, 30, 10, 3979),
('IST16X0002', 'Mediana', 'Mala de Porão', 25, 40, 30, 50, 1911),
('DOH16Z0058', 'Reduzida', 'Mala de Porão', 30, 40, 30, 50, 8887),
('GIG16K4859', 'Reduzida', 'Mala de Porão', 20, 40, 30, 50, 6177),
('GRU16W9304', 'Elevada', 'Instrumento Musical', 5, 20, 12, 10, 3415),
('IST16X0002', 'Mediana', 'Mala de Porão', 25, 40, 30, 50, 1911),
('JFK16F1212', 'Reduzida', 'Mala de Porão', 15, 40, 30, 50, 6861),
('LIS15LA240', 'Mediana', 'Prancha de Surf', 10, 120, 10, 40, 4971),
('LIS16L2034', 'Reduzida', 'Mala de Porão', 20, 40, 30, 50, 4896),
('LIS16LA240', 'Reduzida', 'Mala de Porão', 20, 40, 30, 50, 6861),
('MAD16V4568', 'Reduzida', 'Mala de Porão', 25, 40, 30, 50, 8596),
('OPO15H0240', 'Reduzida', 'Mala de Porão', 32.5, 40, 30, 50, 2368),
('PDL16N3574', 'Mediana', 'Animal de Estimação', 5, 20, 12, 10, 6007);
```

Nr	Fragilidade	Descricao	Peso	Comprimento	Largura	Altura	Passageiro
AMS16M0056	Elevada	Instrumento Musical	12	20	30	10	3979
BOS16A2113	Elevada	Animal de Estimação	5	20	12	10	9581
CCS16C4569	Reduzida	Mala de Porão	15	40	30	50	2294
DOH16Z0058	Reduzida	Mala de Porão	30	40	30	50	8887
DXB16O0023	Reduzida	Mala de Porão	20	40	30	50	8603
GIG16K4859	Reduzida	Mala de Porão	20	40	30	50	6177
GRU16W9304	Elevada	Instrumento Musical	5	20	12	10	3415
IST16X0002	Mediana	Mala de Porão	25	40	30	50	1911
JFK16F1212	Reduzida	Mala de Porão	15	40	30	50	6861
LIS15LA240	Mediana	Prancha de Surf	10	120	10	40	4971
LIS16L2034	Reduzida	Mala de Porão	20	40	30	50	4896
LIS16LA240	Reduzida	Mala de Porão	20	40	30	50	6861
MAD16V4568	Reduzida	Mala de Porão	25	40	30	50	8596
OPO15H0240	Reduzida	Mala de Porão	32.5	40	30	50	2368
PDL16N3574	Mediana	Animal de Estimação	5	20	12	10	6007

- **Tabela VooPassageiro**

```
INSERT INTO voopassageiro
(Voo, Passageiro, Classe)
VALUES
('TP-032', 2368, '1_classe'),
('TP-032', 4971, '2_classe'),
('TP-032', 6861, '3_classe'),
('TP-032', 9581, '1_classe'),
('TP-032', 2294, '2_classe'),
('TP-123', 8596, '3_classe'),
('TP-123', 9610, '1_classe'),
('TP-123', 6007, '1_classe'),
('TP-123', 8009, '3_classe'),
('TP-123', 0542, '2_classe'),
('TP-008', 2993, '3_classe'),
('U2-130', 0188, '2_classe'),
('TP-070', 4617, '3_classe'),
('TP-005', 8603, '2_classe'),
('TP-235', 2443, '3_classe'),
('PG-726', 0233, '1_classe'),
('SA-340', 4896, '3_classe'),
('EK-060', 3979, '3_classe'),
('QR-053', 4179, '1_classe'),
('TP-720', 3264, '3_classe'),
('FR-100', 1911, '3_classe'),
('TP-930', 8887, '1_classe'),
('AF-450', 6177, '3_classe'),
('LH-550', 1455, '3_classe'),
('TP-050', 3415, '3_classe'),
('TP-050', 7129, '3_classe'),
('TP-070', 4617, '3_classe'),
('TP-123', 542, '2_classe'),
('TP-123', 6007, '1_classe'),
('TP-123', 8009, '3_classe'),
('TP-123', 8596, '3_classe'),
('TP-123', 9610, '1_classe'),
('TP-235', 2443, '3_classe'),
('TP-720', 3264, '3_classe'),
('TP-930', 8887, '1_classe'),
('U2-130', 188, '2_classe');
```

Voo	Passageiro	Classe
AF-450	6177	3_classe
EK-060	3979	3_classe
FR-100	1911	3_classe
LH-550	1455	3_classe
PG-726	233	1_classe
QR-053	4179	1_classe
SA-340	4896	3_classe
TP-005	8603	2_classe
TP-008	2993	3_classe
TP-032	2294	2_classe
TP-032	2368	1_classe
TP-032	4971	2_classe
TP-032	6861	3_classe
TP-032	9581	1_classe
TP-050	3415	3_classe
TP-050	7129	3_classe
TP-070	4617	3_classe
TP-123	542	2_classe
TP-123	6007	1_classe
TP-123	8009	3_classe
TP-123	8596	3_classe
TP-123	9610	1_classe
TP-235	2443	3_classe
TP-720	3264	3_classe
TP-930	8887	1_classe
U2-130	188	2_classe

- **Tabela VooBagagem**

```
INSERT INTO voobagagem
(Voo,Bagagem)
VALUES
('TP-032','OPO15H0240'),
('TP-123','GRU16W9304'),
('TP-008','LIS15LA240'),
('U2-130','GIG16K4859'),
('U2-130','JFK16F1212'),
('U2-130','BOS16A2113'),
('U2-130','DOH16Z0058'),
('TP-070','LIS16LA240'),
('TP-005','DOH16Z0058'),
('TP-235','IST16X0002'),
('PG-726','AMS16M0056'),
('SA-340','LIS16L2034'),
('EK-060','DXB1600023'),
('QR-053','PDL16N3574'),
('QR-053','LIS16LA240'),
('QR-053','GIG16K4859'),
('QR-053','BOS16A2113'),
('TP-720','MAD16V4568'),
('FR-100','CCS16C4569'),
('TP-930','BOS16A2113'),
('AF-450','JFK16F1212');
```

Voo	Bagagem
PG-726	AMS16M0056
QR-053	BOS16A2113
TP-930	BOS16A2113
U2-130	BOS16A2113
FR-100	CCS16C4569
TP-005	DOH16Z0058
U2-130	DOH16Z0058
EK-060	DXB1600023
QR-053	GIG16K4859
U2-130	GIG16K4859
TP-123	GRU16W9304
TP-235	IST16X0002
AF-450	JFK16F1212
U2-130	JFK16F1212
TP-008	LIS15LA240
SA-340	LIS16L2034
QR-053	LIS16LA240
TP-070	LIS16LA240
TP-720	MAD16V4568
TP-032	OPO15H0240
QR-053	PDL16N3574

- **Tabela VooPiloto**

```
INSERT INTO voopiloto
(Voo,Piloto)
VALUES
('TP-032',23),
('TP-123',3),
('TP-008',45),
('U2-130',7),
('TP-070',20),
('TP-005',10),
('TP-235',2),
('PG-726',34),
('SA-340',56),
('EK-060',43),
('QR-053',14),
('TP-720',15),
('FR-100',17),
('TP-930',6),
('AF-450',12),
('LH-550',70),
('TP-050',65);
```

Voo	Piloto
TP-235	2
TP-123	3
TP-930	6
U2-130	7
TP-005	10
AF-450	12
QR-053	14
TP-720	15
FR-100	17
TP-070	20
TP-032	23
PG-726	34
EK-060	43
TP-008	45
SA-340	56
TP-050	65
LH-550	70

5.1.3 Restrições Gerais

Apresentamos aqui casos que não implementamos, mas que se fosse necessário impor restrições faríamos.

Consideramos o seguinte:

- Um piloto só pode ingressar na companhia aérea se tiver mais de 500 horas de experiência, isto é, só podem ser introduzidos pilotos na base de dados que tenham mais do que as horas referidas em cima;

Relativamente à restrição da experiência do piloto, a forma que encontramos de a implementar foi com recurso a um trigger que verifica antes de cada inserção na tabela Piloto, se o novo registo contém a experiência mínima para ingressar na companhia aérea.

```
DELIMITER //
CREATE TRIGGER experienciaPiloto
BEFORE INSERT ON Piloto
FOR EACH ROW
BEGIN

DECLARE msg VARCHAR(200);

IF ((new.Experiencia) <= 500)
THEN
    SET msg = 'O piloto não tem a experiência mínima para ingressar na companhia aérea';
    SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = msg;
END IF;
END //
```

Encontramos ainda outras restrições possíveis de serem implementadas e que acrescentariam consistência e coerência aos dados inseridos:

- A capacidade em cada uma das classes tem de ser superior ou igual a 0;
- A hora e data de chegada têm de ser posteriores à hora e data de partida;
- O número de passageiros num voo, não pode exceder a capacidade do avião que irá efetuar o voo;
- O peso de uma bagagem é superior a 0;
- A data da inspeção não pode ser inferior à data atual;
- A hora e data de chegada têm de ser posteriores à hora e data de partida;

5.2. Desenho da representação física - esquema físico

A representação física e o seu respetivo esquema é idêntico à representação lógica que já foi apresentada na figura 7.

5.3. Análise de transações

- Transação Inserir Novo Passageiro

```
DELIMITER $$  
CREATE PROCEDURE inserirPassageiro(IN NrPass INT, IN nomePass VARCHAR(75),  
                                     IN idadePass INT, IN NrtelemovelPass VARCHAR(75),  
                                     IN DescricaoPass VARCHAR(75), IN NrVooPass VARCHAR(75),  
                                     IN ClassePass VARCHAR(75))  
  
BEGIN  
DECLARE erro BOOL DEFAULT 0;  
DECLARE CONTINUE HANDLER FOR SQLEXCEPTION SET erro=1;  
START TRANSACTION;  
  
INSERT INTO Passageiro  
          (Nr, Nome, Idade)  
      VALUES  
          (NrPass, nomePass, idadePass);  
  
INSERT INTO Telemovel  
          (NrTelemovel, Passageiro)  
      VALUES  
          (NrtelemovelPass, NrPass);  
  
INSERT INTO NacionalidadePassageiro  
          (Descricao, Passageiro)  
      VALUES  
          (DescricaoPass, NrPass);  
  
INSERT INTO VooPassageiro  
          (Voo, Passageiro, Classe)  
      VALUES  
          (NrVooPass, NrPass, ClassePass);  
  
IF erro  
THEN ROLLBACK;  
ELSE COMMIT;  
END IF;  
END $$
```

Houve a necessidade implementar um trigger nomeado atualizaQuantidade que na realização da transação Novo Passageiro incrementa uma unidade a quantidade de passageiros do voo em que o passageiro foi inserido

```
DELIMITER $$  
CREATE TRIGGER atualizaQuantidade  
AFTER INSERT ON voopassageiro  
FOR EACH ROW  
BEGIN  
  
    UPDATE Voo  
    SET Quantidade=Quantidade+1  
    WHERE voo.Nr=new.Voo;  
  
END $$
```

- **Transação Inserir Novo Voo**

```
DELIMITER $$
CREATE PROCEDURE `InsereVoo`(
    IN NrVoo VARCHAR(75),
    IN HoraPartidaVoo TIME,
    IN HoraChegadaVoo TIME,
    IN DataPartidaVoo DATE,
    IN DataChegadaVoo DATE,
    IN Preco_1_ClasseVoo FLOAT,
    IN Preco_2_ClasseVoo FLOAT,
    IN Preco_3_ClasseVoo FLOAT,
    IN OrigemVoo VARCHAR(75),
    IN DestinoVoo VARCHAR(75),
    IN QuantidadeVoo VARCHAR(75),
    IN CustoOperacionalVoo FLOAT,
    IN PilotoVoo INT)
BEGIN
    DECLARE Erro BOOL DEFAULT 0;
    DECLARE CONTINUE HANDLER FOR SQLEXCEPTION SET Erro = 1;
    START TRANSACTION;

        INSERT INTO Voo
        (Nr, HoraPartida, HoraChegada, DataPartida, DataChegada,
        Preco_1_Classe, Preco_2_Classe, Preco_3_Classe, Origem,
        Destino, Quantidade, CustoOperacional)
        VALUES
        (HoraPartidaVoo,HoraChegadaVoo, DataPartidaVoo, DataChegadaVoo,Preco_1_ClasseVoo,
        Preco_2_ClasseVoo, Preco_3_ClasseVoo, OrigemVoo, DestinoVoo, QuantidadeVoo,
        CustoOperacionalVoo);

        INSERT INTO voopiloto
        (Voo,Piloto)
        VALUES
        (AviaoVoo, PilotoVoo);

    IF Erro THEN
        ROLLBACK;
    ELSE
        COMMIT;
    END IF;
END $$
```

- **Transação Inserir Nova Bagagem**

- Um passageiro que viaja num voo tem a possibilidade de associar uma ou mais bagagens ao seu voo. Este procedimento só pode ser executado se o passageiro estiver previamente na base de dados e se o voo existir.
- A transação inicia-se com inserção da nova bagagem na tabela Bagagem. É criado então um novo registo com um Nr de Bagagem único e as suas características – Fragilidade, Descrição, Peso, Comprimento, Largura, Altura. Associa-se também a este registo o dono da bagagem (passageiro).
- Posto isto, é feita uma inserção na tabela VooBagagem, onde é acrescentada a bagagem criada ao voo em que o passageiro viaja.

```
DELIMITER $$

CREATE PROCEDURE `CheckInBagagem`(
    IN Voo VARCHAR(75),
    IN DonoBagagem INT,
    IN NrBagagem VARCHAR(75),
    IN FragilidadeBagagem VARCHAR(75),
    IN DescricaoBagagem VARCHAR(75),
    IN PesoBagagem FLOAT,
    IN ComprimentoBagagem FLOAT,
    IN LarguraBagagem FLOAT,
    IN AlturaBagagem FLOAT)

BEGIN
    DECLARE Erro BOOL DEFAULT 0;
    DECLARE CONTINUE HANDLER FOR SQLEXCEPTION SET Erro = 1;
    START TRANSACTION;

    INSERT INTO Bagagem
    (Nr, Fragilidade, Descricao, Peso, Comprimento, Largura, Altura, Passageiro)
    VALUES
    (NrBagagem, FragilidadeBagagem, DescricaoBagagem,
    PesoBagagem, ComprimentoBagagem, LarguraBagagem, AlturaBagagem, DonoBagagem);

    INSERT INTO VooBagagem
    (Voo, Bagagem)
    VALUES
    (Voo, DonoBagagem);

    IF Erro THEN
        ROLLBACK;
    ELSE
        COMMIT;
    END IF;
END $$
```

5.4. Estimativa dos requisitos do espaço em disco agora com base no SGBD escolhido.

Na construção de uma base dados o espaço de ocupação dos dados requeridos para a sua implementação é um dos fatores importantes, por isso é necessário o cálculo de uma estimativa realista quer para a presente implementação, quer também para expansões futuras. É importante desenvolver um modelo que seja extensível e tenha a capacidade de evoluir para suportar novos requisitos com um efeito mínimo sobre os utilizadores existentes. Assim sendo, para determinar que existem alterações significativas previsíveis no futuro da companhia aérea implementada e para avaliar que o nosso modelo lógico de dados pode sustentar estas mudanças e acomodar novos requisitos apresentamos de seguida as estimativas e os cálculos que realizámos.

Começamos por calcular o espaço de cada tabela, ou seja, analisar o espaço que cada atributo ocupa, tendo como fundamento a tabela disponibilizada pela Microsoft com todos os tamanhos para os vários tipos de domínio. Através deste método conseguimos assim obter quanto vai ocupar cada registo de uma determinada tabela com certos atributos. Por fim, conseguimos obter o espaço no tempo presente de cada tabela multiplicando pela cardinalidade atual.

De seguida temos o cálculo de espaço estimado para cada uma das tabelas apresentada no modelo lógico do projeto e também futuras previsões e estimativas para o seu crescimento com respetivas representações gráficas.

Tabela 2 - Tamanho ocupado pelos atributos da Tabela Avião

Avião		
Atributos	Domínio	Tamanho (bytes)
Nr	INTEGER	4
Combustível	FLOAT	8
Série	VARCHAR	8
Modelo	VARCHAR	10
Ano	DATE	3
1_classe	INTEGER	4
2_classe	INTEGER	4
3_classe	INTEGER	4
Inspeção	DATE	3

Espaço estimado (1 registo) = $4 + 8 + 8 + 10 + 3 + 4 + 4 + 4 + 3 = 48$ bytes

Espaço estimado (cardinalidade atual = 15) = $48 * 15 = 720$ bytes

Estimando-se um aumento futuro de clientes a recorrerem aos serviços da companhia aérea existirá a necessidade de aumentar a frota adicionando aeronaves, consequentemente o espaço

de disco requerido vai aumentar gradualmente sendo que 48 bytes representa cada avião adicionado à frota.

Tabela 3 - Crescimento futuro e espaço ocupado pelo nº de aviões

Ano	Nº Aviões	Espaço Ocupado (bytes)
2016	15	720
2017	17	816
2018	20	960
2019	22	1056
2020	24	1152

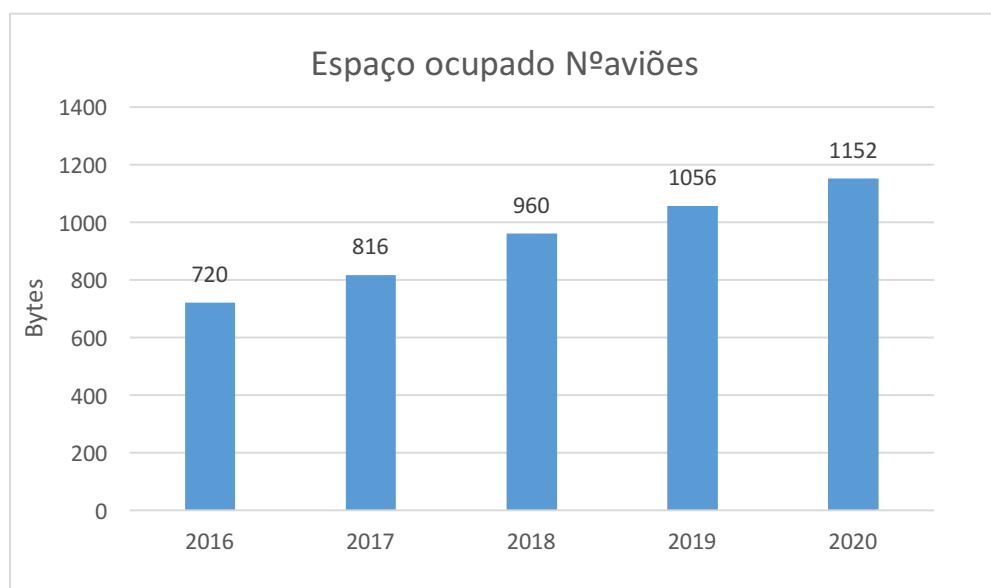


Figura 24 - Gráfico do espaço ocupado pelo nº de aviões

Tabela 4 - Tamanho ocupado pelos atributos da tabela Piloto

Piloto		
Atributos	Domínio	Tamanho
Nr	INTEGER	4
Nome	VARCHAR	75
Estatuto	VARCHAR	20
Experiencia	INTEGER	4
Pais	VARCHAR	50
Cidade	VARCHAR	50
Rua	VARCHAR	75
CodPostal	VARCHAR	20
Salario	FLOAT	8

Espaço estimado (1 registo) = 4 + 75 + 20 + 4 + 50 + 50 + 75 + 20 + 8 = 306 bytes

Espaço estimado (cardinalidade atual = 17) = 306 * 17 = 5202 bytes

Com o crescimento futuro da companhia consequentemente haverá a necessidade de contratação de novos pilotos para o manuseamento das aeronaves. Por cada registo de um piloto inserido ocupa 306 bytes.

Tabela 5 - Crescimento futuro e espaço ocupado pelo nº de pilotos

Ano	Nº Pilotos	Espaço ocupado (bytes)
2016	30	9180
2017	34	10404
2018	40	12240
2019	44	13464
2020	48	14688

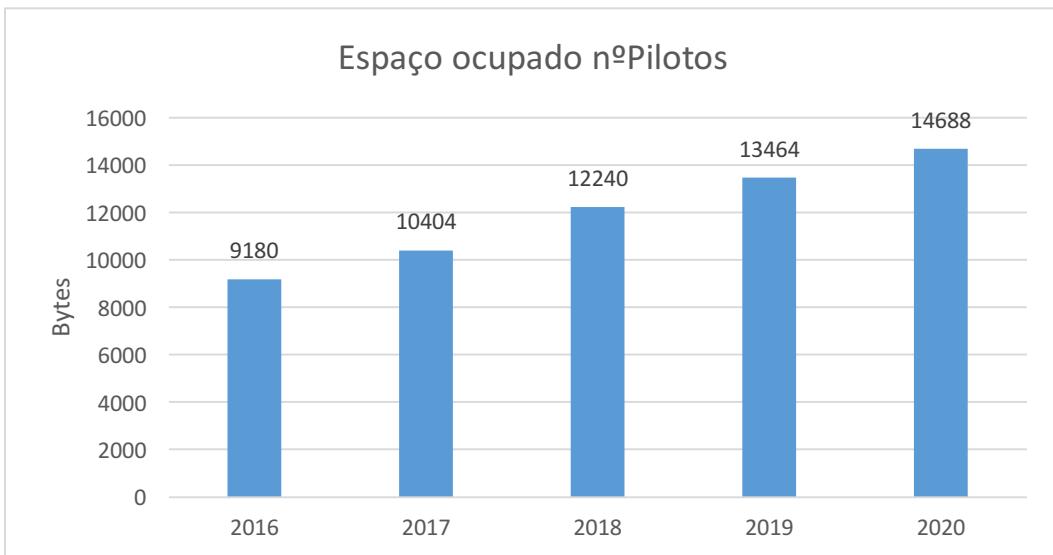


Figura 25 - Gráfico do espaço ocupado pelo nº de pilotos

Tabela 6 - Tamanho ocupado pelos atributos da tabela NacionalidadePiloto

NacionalidadePiloto		
Atributos	Domínio	Tamanho
Descricao	VARCHAR	75
Piloto	INTEGER	4

Espaço estimado (1 registo) = 4 + 75 = 79 bytes

Espaço estimado (cardinalidade atual = 17) = 79 * 17 = 1343 bytes

Sendo a nacionalidade um multivalue da entidade piloto esta mesma irá aumentar igualmente como o piloto sendo o espaço de disco requerido 79 bytes.

Tabela 7 - Tamanho ocupado pelos atributos da tabela Voo

Voo		
Atributos	Domínio	Tamanho
Nr	VARCHAR	8
Aviao	INTEGER	4
HoraPartida	TIME	5
HoraChegada	TIME	5
DataPartida	DATE	3

DataChegada	DATE	3
Preco_1_Classe	FLOAT	8
Preco_2_Classe	FLOAT	8
Preco_3_Classe	FLOAT	8
Origem	VARCHAR	75
Destino	VARCHAR	75
Quantidade	INTEGER	4
CustoOperacional	FLOAT	8

Espaço estimado (1 registo) = $8 + 4 + 5 + 5 + 3 + 3 + 8 + 8 + 8 + 75 + 75 + 4 + 8 = 214$ bytes

Espaço estimado (cardinalidade atual = 17) = $210 * 17 = 3638$ bytes

Prevendo um aumento futuro de clientes a recorrerem aos serviços da companhia aérea, os próprios serviços irão aumentar, sendo assim, estima-se um aumento de espaço de disco necessário para o aumento da ocorrência de voos. Cada registo de um voo ocupa aproximadamente 210 bytes.

Tabela 8 - Crescimento futuro e espaço ocupado pelo nº de Voos

Ano	Nº de voos	Espaço ocupado (bytes)	Espaço ocupado (MB)
2016	30	2343300	2,34
2017	34	2655740	2,66
2018	40	3124400	3,12
2019	44	3436840	3,44
2020	48	3749280	3,75

Tabela 9 Tamanho ocupado pelos atributos da tabela VooPiloto

VooPiloto		
Atributos	Domínio	Tamanho
Voo	INTEGER	4
Piloto	INTEGER	4

Espaço estimado (1 registo) = $4 + 4 = 8$ bytes

Espaço estimado (cardinalidade atual = 17) = $8 * 17 = 136$ bytes

Tabela 10 -Tamanho ocupado pelos atributos da tabela Passageiro

Passageiro		
Atributos	Domínio	Tamanho
Nr	INTEGER	4
Nome	VARCHAR	75
Idade	INTEGER	4

Espaço estimado (1 registo) = 4 + 75 + 4 = 83 bytes

Espaço estimado (cardinalidade atual = 16) = 83 * 16 = 1328 bytes

Como sendo o crescimento de serviços prestados um dos principais objetivos da empresa estima-se o aumento de passageiros a recorrerem aos serviços que a companhia aérea fornece. Sendo assim o espaço de disco para a informação destes aumentará gradualmente, ocupando 83 bytes por cada novo passageiro.

Tabela 11 - Crescimento futuro e espaço ocupado pelo nº de Passageiros

Ano	Nº Passageiros	Tamanho ocupado(bytes)	Tamanho ocupado (MB)
2016	120	3635400	3,64
2017	136	4120120	4,12
2018	160	4847200	4,85
2019	176	5331920	5,33
2020	192	5816640	5,82

Tabela 12 - Tamanho ocupado pelos atributos da tabela Telemovel

Telemovel		
Atributos	Domínio	Tamanho
NrTelemovel	VARCHAR	30
Passageiro	INTEGER	4

Espaço estimado (1 registo) = 30 + 4 = 34 bytes

Espaço estimado (cardinalidade atual = 16) = 34 * 16 = 544 bytes

Sendo o telemóvel um multivvalor da entidade passageiro este mesmo irá aumentar igualmente como o passageiro sendo o espaço de disco requerido 34 bytes.

Tabela 13 - Tamanho ocupado pelos atributos da tabela NacionalidadePassageiro

NacionalidadePassageiro		
Atributos	Domínio	Tamanho
Descrição	VARCHAR	50
Passageiro	INTEGER	4

Espaço estimado (1 registo) = $50 + 4 = 54$ bytes

Espaço estimado (cardinalidade atual = 16) = $54 * 16 = 864$ bytes

Sendo a nacionalidade um multivvalor da entidade passageiro esta mesma irá aumentar igualmente como o passageiro sendo o espaço de disco requerido 54 bytes.

Tabela 14 - Tamanho ocupado pelos atributos da tabela VooPassageiro

VooPassageiro		
Atributos	Domínio	Tamanho
Voo	INTEGER	4
Passageiro	INTEGER	4
Classe	VARCHAR	10

Espaço estimado (1 registo) = $4 + 4 + 10 = 18$ bytes

Espaço estimado (cardinalidade atual = 16) = $18 * 16 = 288$ bytes

Tabela 15 - Tamanho ocupado pelos atributos da tabela Bagagem

Bagagem		
Atributos	Domínio	Tamanho
Nr	VARCHAR	12
Fragilidade	VARCHAR	50
Descricao	VARCHAR	75
Peso	FLOAT	8
Comprimento	FLOAT	8
Largura	FLOAT	8
Altura	FLOAT	8
Passageiro	INTEGER	4

Espaço estimado (1 registo) = $12 + 50 + 75 + 8 + 8 + 8 + 8 + 4 = 173$ bytes

Espaço estimado (cardinalidade atual = 15) = $173 * 15 = 2595$ bytes

Estima-se um aumento do número de bagagens, como consequência do aumento do número de clientes, havendo assim a necessidade de aumentar o espaço de disco para a informação relativamente a esta mesma. Cada unidade de bagagem inserida aumentará o espaço de disco em 173 bytes.

Tabela 16 - Crescimento futuro e espaço ocupado pelo nº de Bagagens

Ano	Nº de Bagagens	Espaço Ocupado (bytes)	Espaço ocupado (MB)
2016	140	8840300	8,84
2017	159	10040055	10,04
2018	186	11744970	11,74
2019	205	12944725	12,94
2020	224	14144480	14,14

Tabela 17 - Tamanho ocupado pelos atributos da tabela VooBagagem

VooBagagem		
Atributos	Domínio	Tamanho
Bagagem	FLOAT	8
Voo	INTEGER	4

Espaço estimado (1 registo) = 8 + 4 = 12 bytes

Espaço estimado (cardinalidade atual = 21) = 12 * 21 = 252 bytes

No total necessita-se presentemente de 16910 bytes de espaço de disco para armazenar a base de dados corrente, tendo sido este valor calculado através do tamanho de espaço de dados referidos na tabela fornecida pela Microsoft, como já foi referido anteriormente, apenas alterando o tamanho do VARCHAR consoante o atributo em causa.

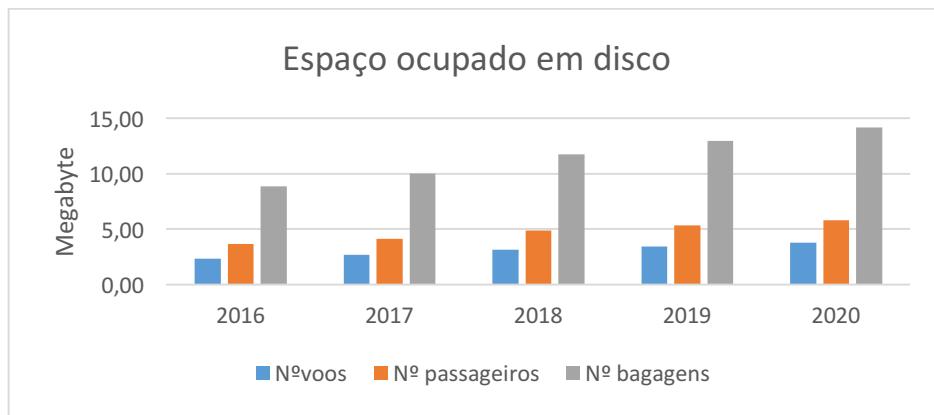


Figura 26 - Gráfico espaço ocupado por voos, passageiros e bagagens

5.5. Definição das vistas dos utilizadores e regras de acesso

Um dos pontos importantes no desenvolvimento de um Sistema de Gestão de Base de Dados é a restrição das vistas de cada utilizador. Assumimos que a nossa implementação tem os seguintes tipos de utilizadores:

- **Administrador** – Tem acesso a toda a Base de Dados e pode executar todas as ações que entender;
- **Gestor de recursos da companhia aérea** – Tem acesso a toda a informação, não tendo permissões para remover registos nem para eliminar tabelas;
- **Funcionário de reserva** – Tem apenas acesso aos voos e à lista de passageiros nos voos, aos passageiros e às bagagens. Tem permissão para inserir um passageiro e uma bagagem na base de dados, mas não tem no entanto permissões para eliminar registos;
- **Cliente** – Tem acesso apenas aos voos, exceto o seu custo operacional e a sua quantidade (número de passageiros);

Quanto ao **gestor de recursos da companhia aérea**, sabemos que tem acesso a todas as informações, tal como o administrador, não tendo, no entanto, hipótese de eliminar tabelas.

```
CREATE USER 'gestorfunc'@'localhost'
IDENTIFIED BY 'gestorpassword';

REVOKE DROP, CREATE
ON *.* 
FROM 'gestorfunc'@'localhost';
```

Quanto à vista do **cliente** temos, a criação do utilizador cliente e a definição dos seus privilégios.

```
CREATE USER 'cliente'@'localhost'
IDENTIFIED BY 'password';

REVOKE INSERT, UPDATE, DELETE, DROP, CREATE
ON *.* 
FROM 'cliente'@'localhost';

GRANT SELECT
ON cliente.*
TO 'cliente'@'localhost';
```

```

CREATE
    ALGORITHM = UNDEFINED
    DEFINER = `cliente`@`localhost`
    SQL SECURITY DEFINER
VIEW `companhiaaaerea`.`cliente` AS
SELECT
    `v`.`Nr` AS `Nr`,
    `v`.`Aviao` AS `Aviao`,
    `v`.`HoraPartida` AS `HoraPartida`,
    `v`.`HoraChegada` AS `HoraChegada`,
    `v`.`DataPartida` AS `DataPartida`,
    `v`.`DataChegada` AS `DataChegada`,
    `v`.`Preco_1_Classe` AS `Preco_1_classe`,
    `v`.`Preco_2_Classe` AS `Preco_2_classe`,
    `v`.`Preco_3_Classe` AS `Preco_3_classe`,
    `v`.`Origem` AS `Origem`,
    `v`.`Destino` AS `Destino`
FROM
    `companhiaaaerea`.`voo` `v`

```

Quanto à vista de um **funcionário** temos a definição dos seguintes privilégios:

```

GRANT UPDATE
ON Passageiro.*
TO 'reservafunc'@'localhost';

GRANT UPDATE
ON Voo.*
TO 'reservafunc'@'localhost';

GRANT UPDATE
ON VooPassageiro.*
TO 'reservafunc'@'localhost';

GRANT UPDATE
ON Bagagem.*
TO 'reservafunc'@'localhost';

GRANT UPDATE
ON VooBagagem.*
TO 'reservafunc'@'localhost';

GRANT SELECT
ON ViewVooPassageiro.*
TO 'reservafunc'@'localhost';

GRANT SELECT
ON ViewVooBagagem.*
TO 'reservafunc'@'localhost';

GRANT SELECT
ON ViewPassageiro.*
TO 'reservafunc'@'localhost';

GRANT SELECT
ON ViewBagagem.*
TO 'reservafunc'@'localhost';

CREATE USER 'reservafunc'@'localhost'
    IDENTIFIED BY 'password';

REVOKE DELETE, DROP, CREATE
    ON *.*
    FROM 'reservafunc'@'localhost';

GRANT INSERT
ON Passageiro.*
TO 'reservafunc'@'localhost';

GRANT INSERT
ON Bagagem.*
TO 'reservafunc'@'localhost';

GRANT INSERT
ON VooPassageiro.*
TO 'reservafunc'@'localhost';

GRANT INSERT
ON VooBagagem.*
TO 'reservafunc'@'localhost';

GRANT SELECT
ON ViewVoo.*
TO 'reservafunc'@'localhost';

```

Vista do Funcionário de Reservas para os voos:

```
CREATE
  ALGORITHM = UNDEFINED
  DEFINER = `reservafunc`@`localhost`
  SQL SECURITY DEFINER
VIEW `companhiaaerea`.`viewvoo` AS
SELECT
  `v`.`Nr` AS `Nr`,
  `v`.`HoraPartida` AS `HoraPartida`,
  `v`.`HoraChegada` AS `HoraChegada`,
  `v`.`DataPartida` AS `DataPartida`,
  `v`.`DataChegada` AS `DataChegada`,
  `v`.`Preco_1_Classe` AS `Preco_1_Classe`,
  `v`.`Preco_2_Classe` AS `Preco_2_Classe`,
  `v`.`Preco_3_Classe` AS `Preco_3_Classe`,
  `v`.`Origem` AS `Origem`,
  `v`.`Destino` AS `Destino`,
  (((`a`.`1_classe` + `a`.`2_classe`) + `a`.`3_classe`) - `v`.`Quantidade`) AS `LugaresLivres`
FROM
  (`companhiaaerea`.`voo` `v`
  JOIN `companhiaaerea`.`avião` `a` ON ((`a`.`Nr` = `v`.`Aviao`)))
```

Vista do Funcionário de Reservas para a lista de passageiros dos voos:

```
CREATE
  ALGORITHM = UNDEFINED
  DEFINER = `reservafunc`@`localhost`
  SQL SECURITY DEFINER
VIEW `companhiaaerea`.`viewvoopassageiro` AS
SELECT
  `vp`.`Voo` AS `Voo`,
  `vp`.`Passageiro` AS `Passageiro`,
  `vp`.`Classe` AS `Classe`
FROM
  `companhiaaerea`.`voopassageiro` `vp`
```

Vista do Funcionário de Reservas para a lista de bagagens dos voos:

```
CREATE
  ALGORITHM = UNDEFINED
  DEFINER = `reservafunc`@`localhost`
  SQL SECURITY DEFINER
VIEW `companhiaaerea`.`viewvoobagagem` AS
SELECT
  `vb`.`Voo` AS `Voo`,
  `vb`.`Bagagem` AS `Bagagem`,
  `bp`.`Passageiro` AS `NrPassageiro`
FROM
  (`companhiaaerea`.`voobagagem` `vb`
  JOIN `companhiaaerea`.`bagagem` `bp` ON ((`bp`.`Nr` = `vb`.`Bagagem`)))
```

Vista do Funcionário de Reservas para os passageiros:

```
CREATE
ALGORITHM = UNDEFINED
DEFINER = `reservafunc`@`localhost`
SQL SECURITY DEFINER
VIEW `companhiaaerea`.`viewpassageiro` AS
SELECT
    `companhiaaerea`.`passageiro`.`Nr` AS `Nr`,
    `companhiaaerea`.`passageiro`.`Nome` AS `Nome`,
    `companhiaaerea`.`passageiro`.`Idade` AS `Idade`
FROM
    `companhiaaerea`.`passageiro`
```

Vista do Funcionário de Reservas para as bagagens:

```
CREATE
ALGORITHM = UNDEFINED
DEFINER = `reservafunc`@`localhost`
SQL SECURITY DEFINER
VIEW `companhiaaerea`.`viewbagagem` AS
SELECT
    `companhiaaerea`.`bagagem`.`Nr` AS `Nr`,
    `companhiaaerea`.`bagagem`.`Fragilidade` AS `Fragilidade`,
    `companhiaaerea`.`bagagem`.`Descricao` AS `Descricao`,
    `companhiaaerea`.`bagagem`.`Peso` AS `Peso`,
    `companhiaaerea`.`bagagem`.`Comprimento` AS `Comprimento`,
    `companhiaaerea`.`bagagem`.`Largura` AS `Largura`,
    `companhiaaerea`.`bagagem`.`Altura` AS `Altura`,
    `companhiaaerea`.`bagagem`.`Passageiro` AS `Passageiro`
FROM
    `companhiaaerea`.`bagagem`
```

6. Conclusões e Trabalho Futuro

Com a realização deste trabalho prático enriquecemos os nossos conhecimentos sobre os conteúdos apreendidos nas aulas e consolidamos as nossas competências relacionadas com o processo de criação e desenvolvimento de um SGBD.

Na fase inicial do projeto de Base de Dados foi proposto a implementação de um sistema de base de dados em que a escolha do tema ficava ao nosso critério, no nosso caso decidimos que seria um sistema de Base de Dados de uma companhia aérea. De seguida, foi executado a análise e levantamento de requisitos necessários para a sua implementação.

Através do levantamento de requisitos reconhecemos que a base de dados necessitava de várias entidades fundamentais, tal como voo, sendo esta a principal pois é o único serviço que a companhia aérea pode prestar. Partindo desta entidade outras se desenvolveram consequentemente, tais como: o avião e os pilotos essenciais para o desempenho dos voos.

Sendo o objetivo da companhia aérea servir a necessidade dos possíveis clientes, houve a necessidade de criar duas entidades, passageiro e bagagem. Em relação aos atributos, após o levantamento de requisitos foi relativamente fácil defini-los. Desta forma, poucas alterações foram efetuadas ao longo deste projeto, pois no nosso ponto de vista, a definição feita correspondia ao que pretendíamos.

Respetivamente aos relacionamentos entre entidades existiram alguns percalços na definição e interpretação dos mesmos, no entanto as dúvidas foram esclarecidas em conjunto com o professor.

Após a execução da fase inicial do projeto construiu-se o requerido modelo conceptual que nos dá uma pré-visualização sobre parte do funcionamento de uma companhia aérea com os seus devidos requisitos validados e um esquema coerente.

De seguida, na modelação lógica desenvolveu-se a transformação do modelo conceptual para modelo lógico utilizando como ferramenta o Workbench também utilizado nas aulas lecionadas, procedendo de seguida ao processo de normalização até à 3ºForma Normal e várias regras de integridade assegurando que a base de dados se encontrava consistente e sem redundância. Foram implementados ainda nesta fase algumas transações ilustradas no respetivo mapa de transações.

Por fim, foi implementado o modelo físico passando assim ao processo de criar as relações base e o seu povoamento, assim como a implementação das restrições gerais do sistema e das transições nomeadas no modelo lógico do projeto. Nesta fase também realizamos o cálculo da estimativa de espaço de disco que iria ser necessária incluindo projeções futuras ilustradas por gráficos temporais. São apresentadas ainda as vistas do utilizador e respetivas regras de acesso. Relativamente ao trabalho futuro a ser realizado existe perfeita noção que há possibilidade de melhoramento e otimização no projeto. Uma possível alteração ao nosso modelo seria a criação de uma entidade nacionalidade, eliminando-se assim a existência dos atributos MV da nacionalidade do Piloto e Passageiro, resultando numa otimização da base de dados.

otimizando assim a base de dados e removendo assim dois atributos MV. Com esta alteração podíamos ainda dar resposta, por exemplo, à pergunta “Quais os passageiros e pilotos de nacionalidade Portuguesa?”. Futuramente podíamos ainda adicionar mais registos à Base de Dados, uma vez que o seu tamanho atual apenas tem os registos suficientes para a parte experimental das funcionalidades implementadas e garantir que esta é escalável. Outro aspeto seria a criação de mais restrições gerais, tais como as que foram sugeridas no ponto 5.1.3. Em termos de implementações em ambiente real tornou-se claro que este projeto é apenas um trabalho académico não sendo praticável devido às dimensões que uma companhia aérea real acarreta, mas não deixa de dar uma perspetiva do funcionamento básico de uma base de dados e consideramos que foi suficiente para aplicar as noções lecionadas durante o semestre.

7. Ferramentas utilizadas

[01] **brModelo 2.0.0:**

Utilizado para desenhar o modelo conceptual.

[02] **MySQLWorkbench 6.2:** Utilizado para desenhar o modelo lógico e implementar a base de dados física.

[03] **Microsoft Office 2016:** Apresentando um bom leque de ferramentas, o grupo decidiu usar o Word para a criação do presente documento e o PowerPoint para a apresentação do projeto.

[04] **Microsoft SQL Server 2014:** Por sugestão do docente, o grupo utilizou esta ferramenta para implementar a Base de Dados.

Bibliografia

- [1] Connolly, T. M. & Begg, C.E., 2005. Database System - A Practical Approach to Design, Implementation and Management 4th Edition
- [2] Gouveia, F., "Fundamentos de Bases de Dados", Coleção Tecnologias da Informação, FCA, Editora de Informática, 2014.
- [3] <http://dev.mysql.com/doc/refman/5.7/en/storage-requirements.html>. Acesso 18 Janeiro 2016

Lista de Siglas e Acrónimos

BD Base de Dados

MV Multi-valor

Ex. Exemplo

Anexos

Os anexos de seguida apresentados são uma ferramenta adicional necessária para uma melhor interpretação, completando assim alguns tópicos e secções que são abordados no relatório.

Assim sendo, os anexos estão divididos em três partes:

- Anexo 1 – Referente aos dicionários de dados
- Anexo 2 – Alusiva às versões do modelo conceptual
- Anexo 3 – Mapa de transações

I. Anexo 1 – Dicionário de Dados

Entidade	Multiplicidade	Relacionamento	Multiplicidade	Entidade
Avião	(1,1)	Executa	(0,n)	Voo
Avião	(0,n)	Transporta	(0,n)	Passageiro
Voo	(0,n)	Tem	(0,n)	Passageiro
Voo	(0,n)	Pilotado	(1,n)	Piloto
Voo	(1,n)	Carrega	(0,n)	Bagagem
Passageiro	(1,1)	Leva	(0,n)	Bagagem

Tabela 18 - Dicionário de Dados dos relacionamentos

Entidade	Descrição	Sinónimos	Ocorrência
Avião	Aviões pertencentes à companhia aérea	Aeronave	Avião executa vários voos; Avião transporta várias pessoas
Voo	Voos executados pelos transportes da companhia aérea	Viagem	Voo tem vários passageiros; Voo é pilotado por pilotos; Voo carrega bagem
Piloto	Pilotos da companhia que pilotam os aparelhos aéreos	Tripulante	Piloto pilota aviões
Passageiro	Passageiros que usufruem dos serviços da companhia aérea	Cliente	Passageiro leva bagagem
Bagagem	Bagagem que os passageiros levam com eles	Mala	Bagagem é carregada no avião

Tabela 19 - Dicionário de dados das entidades

Entidade	Atributos	Descrição	Domínio	NULL	MV	Tamanho
Avião	Nr (PK)	Número identificador do avião	INTEGER	Não	Não	4
	Combustível	Combustível presente no avião	FLOAT	Não	Não	8
	Série	Número de série do avião	VARCHAR	Não	Não	75
	Modelo	Modelo do avião	VARCHAR	Não	Não	75
	Ano	Ano de fabrico do avião	DATE	Não	Não	3
	1_classe	Número de lugares 1ª classe do avião	INTEGER	Não	Não	4
	2_classe	Número de lugares 2ª classe do avião	INTEGER	Não	Não	4
	3_classe	Número de lugares 3ª classe do avião	INTEGER	Não	Não	4
	Inspeção	Data da próxima inspeção	DATE	Não	Não	3
Voo	Nr (PK)	Número identificador do voo	VARCHAR	Não	Não	75
	HoraPartida	Hora de partida do voo	TIME	Não	Não	5
	HoraChegada	Hora de chegada do voo	TIME	Não	Não	5
	DataPartida	Data de partida do voo	DATE	Não	Não	3
	DataChegada	Data de chegada	DATE	Não	Não	3
	Preço_1classe	Preço de lugar de 1ª classe	FLOAT	Sim	Não	8
	Preço_2classe	Preço de lugar de 2ª classe	FLOAT	Sim	Não	8
	Preço_3classe	Preço de lugar de 3ª classe	FLOAT	Sim	Não	8
	Origem	Aeroporto de origem	VARCHAR	Não	Não	75
	Destino	Aeroporto de destino	VARCHAR	Não	Não	75
	Quantidade	Quantidade de passageiros no voo	INTEGER	Sim	Não	4
Passageiro	CustoOperacional	Custo Operacional do voo	FLOAT	Não	Não	8
	Nr (PK)	Número identificador do passageiro	INTEGER	Não	Não	8
	Classe		VARCHAR	Não	Não	75

		Classe onde viaja o passageiro				
	Nacionalidade	Nacionalidade do Passageiro	VARCHAR	Não	Sim	75
	Telemóvel	Contacto do passageiro	VARCHAR	Não	Sim	75
	Nome	Nome do Passageiro	VARCHAR	Não	Não	75
	Idade	Idade do Passageiro	INTEGER	Não	Não	4
Piloto	Nr (PK)	Número identificador do piloto	INTEGER	Não	Não	4
	Nome	Nome do Piloto	VARCHAR	Não	Não	75
	Estatuto	Estatuto do Piloto no cockpit	VARCHAR	Não	Não	75
	Experiência	Anos de Experiência	INTEGER	Não	Não	4
	País	País onde reside o piloto	VARCHAR	Não	Não	75
	Cidade	Cidade onde reside o piloto	VARCHAR	Não	Não	75
	Rua	Rua onde reside o piloto	VARCHAR	Não	Não	75
	Cod_Postal	Código Postal do local onde reside	VARCHAR	Não	Não	75
	Salário	Salário mensal que o piloto aufera	FLOAT	Não	Não	8
	Nacionalidade	Nacionalidade do Piloto	VARCHAR	Não	Sim	75
Bagagem	Nr (PK)	Número identificador da bagagem	VARCHAR	Não	Não	75
	Fragilidade	Fragilidade da bagagem	VARCHAR	Não	Não	75
	Descrição	Tipo da bagagem transportada	VARCHAR	Não	Não	75
	Peso	Peso em Kg da bagagem	FLOAT	Não	Não	8
	Comprimento	Comprimento da bagagem	FLOAT	Não	Não	8
	Largura	Largura da bagagem	FLOAT	Não	Não	8
	Altura	Altura da bagagem	FLOAT	Não	Não	8

Tabela 20 - Dicionário de dados dos atributos das entidades

II. Anexo 2 – Versões do Esquema Conceptual

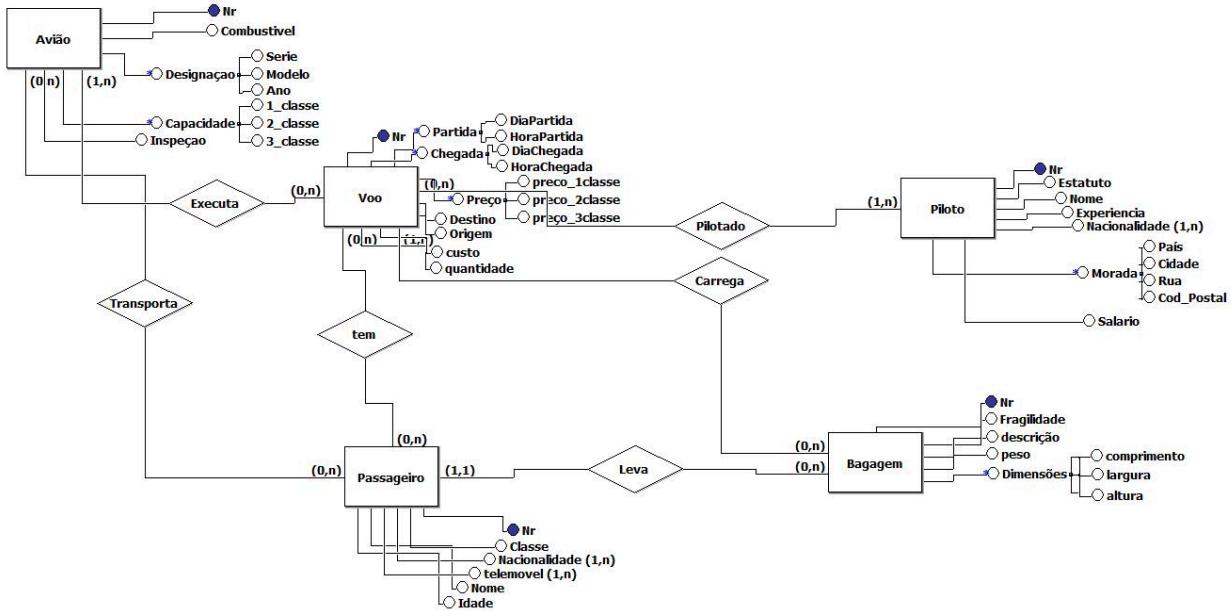


Figura 27 - Versão 1 do Esquema conceptual

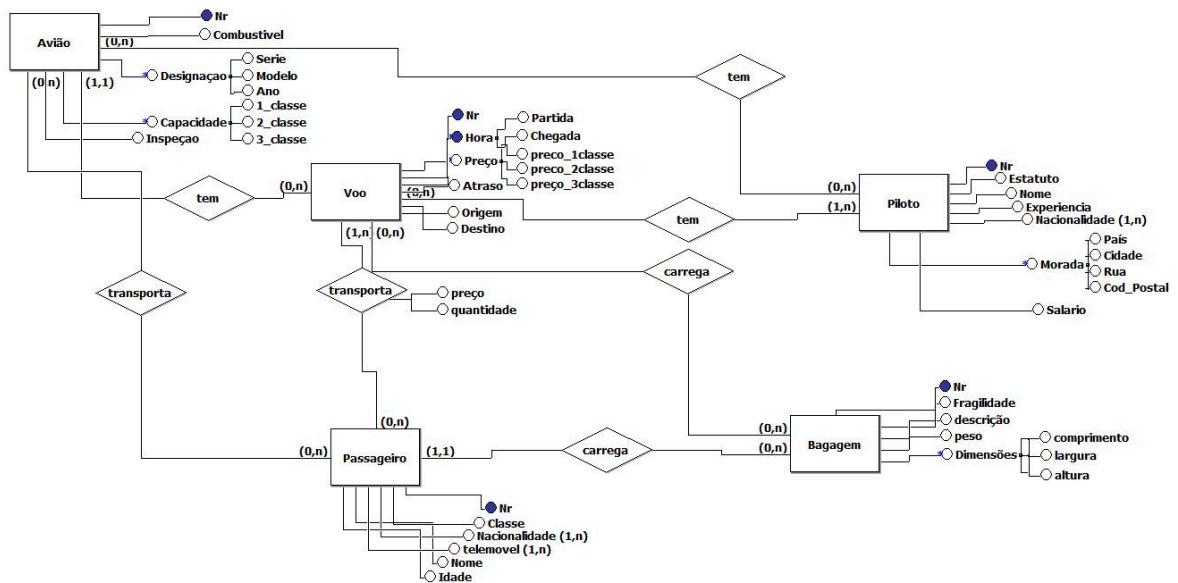
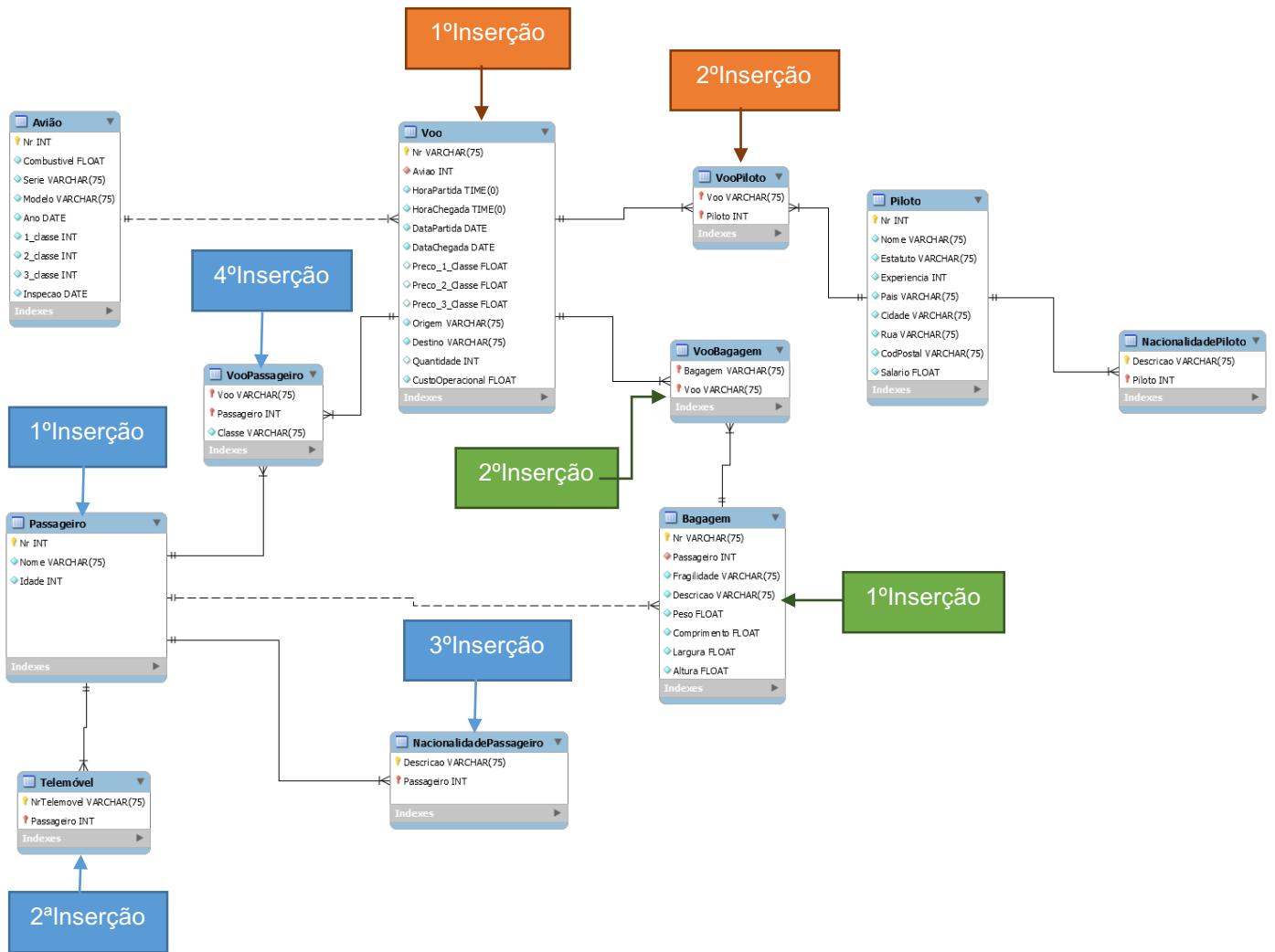


Figura 28 - Versão 2 do Esquema Conceptual

III. Anexo 3 – Mapa de Transações



1ºTransação Novo Passageiro

2ªTransação Novo Voo

3ºTransação Nova Bagagem