**Gestão de Aeroporto**

QuAIRy *- Flying with class*

Bases de Dados

Ano Letivo 2018/2019

Turma 6 Grupo 6

Daniel Ferreira Brandão, up201705812, up201705812@fe.up.pt

Henrique José Santos, up201706898, up201706898@fe.up.pt

Pedro Miguel Moás, up201705208, up201705208@fe.up.pt

17 de março de 2019

Índice

[Tema do trabalho 3](#_Toc3127046)

[Implementação 4](#_Toc3127047)

[Diagrama de classes UML 5](#_Toc3127048)

[Conclusão 6](#_Toc3127049)

Tema do trabalho

Este projeto tem como objetivo simular uma base de dados que conterá a informação necessária para gerir um aeroporto. Isto inclui os seus funcionários e passageiros, como também informações sobre as viagens que chegam e saem do respetivo aeroporto.

Implementação

Começando pela classe **Pessoa**, sobre esta interessa saber informações básicas como nome e data de nascimento, mas para detalhes mais complexos criaram-se especializações desta, as classes **Passageiro**, que guarda apenas um número (ID) para identificação, e **Funcionário**, que guarda o seu salário, área de trabalho e categoria.

A parte central é, naturalmente, a classe **Viagem**, que guarda a data de partida, data de chegada, duração e o tipo (partida ou chegada, dependendo da direção do avião). A data de chegada poderia ser determinada através dos outros atributos, mas dependeria da diferença de fuso horário entre aeroportos, sendo necessário aceder a várias outras classes sempre que fosse preciso calcular esse atributo.

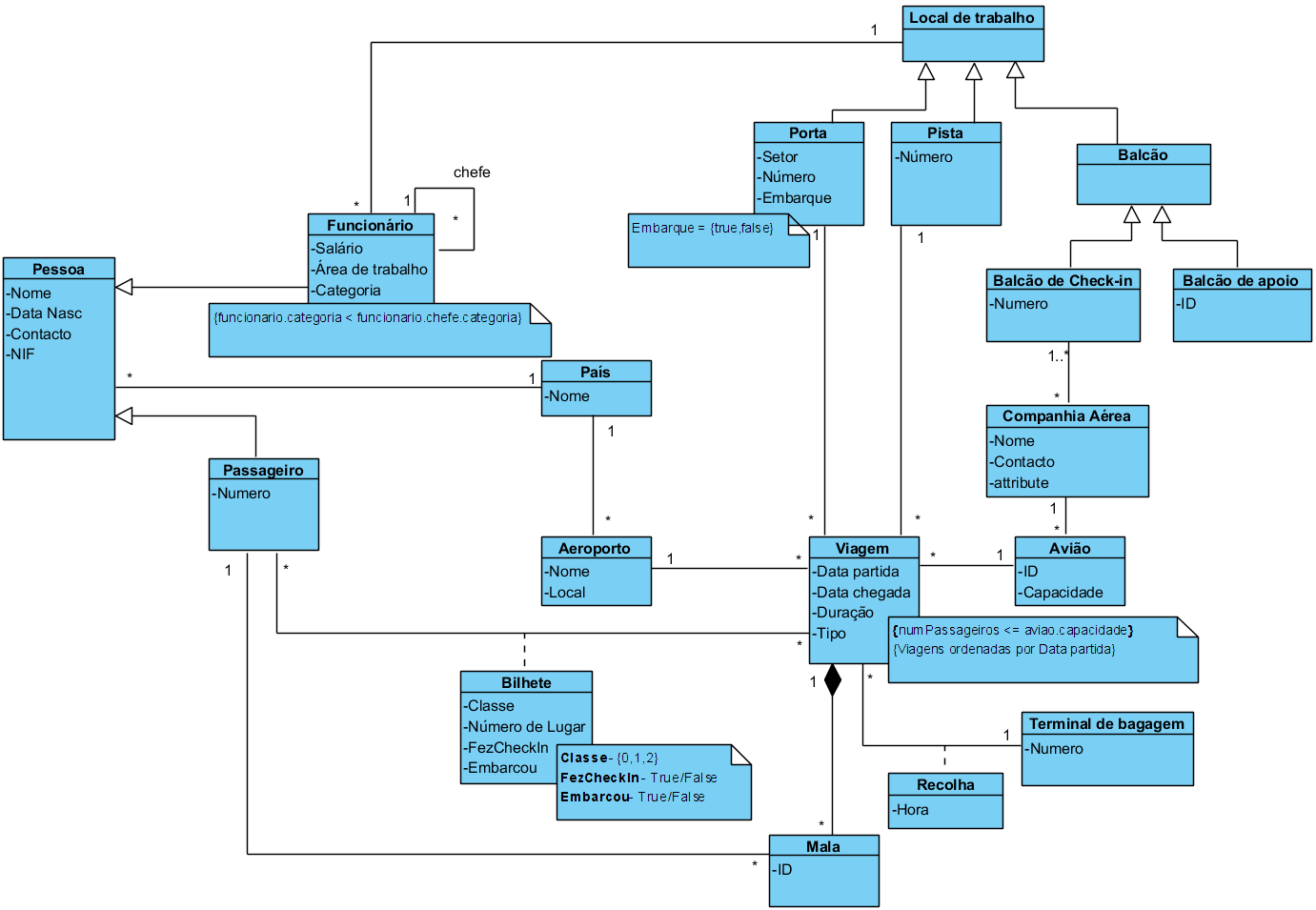
Naturalmente, uma Viagem tem um **Avião** associado (que por sua vez pode ter várias viagens), cuja classe guarda o seu ID, nome e capacidade. Um avião pertencerá a uma **Companhia Aérea** (que, certamente, poderá ter vários aviões).

Continuando, também é importante que haja uma gestão da bagagem que passa pelo Aeroporto. Assim, uma Viagem pode ter mais do que uma **Mala**. Sobre a Mala, guarda-se o seu ID, que é único. Naturalmente, a Mala não corresponde ao objeto físico em si, pois se um passageiro trouxer a mesma mala para dois voos diferentes, certamente que o Aeroporto guardará dois IDs diferentes. Esta classe tem sempre um passageiro associado, que pode ter outras malas associadas. Para além da Mala, as viagens estão sempre associadas a um **Terminal de Bagagem**, onde serão despachadas as suas malas, a uma hora de recolha guardada. Como seria de esperar, estes podem estar ligados a várias viagens.

Para completar as classes Viagem e Passageiro, também é importante notar que não só uma Viagem tem vários passageiros, como um Passageiro pode ter várias viagens, pois fica registado na base de dados do Aeroporto no caso provável que retorne. Para cada ligação entre um Passageiro e uma Viagem, guarda-se informação do “Bilhete”, com a informação sobre a Classe (Executiva – 0, Primeira Classe – 1, Económica – 2), o número de lugar no avião, se já fez Check-In, e se já embarcou, pois é sempre importante saber onde se encontra um passageiro, no caso de haver um eventual problema.

Agora relativamente à classe Funcionário, este está ligado a um **Local de Trabalho**, que é uma generalização (incompleta e disjunta) de **Pista** (desolagem / aterragem), **Porta** (embarque / desembarque), e **Balcão**. Esta última é uma generalização (completa e disjunta) de **Balcão de Apoio** e **Balcão de Check-in**. Sobre as classes Pista e Porta, ambas podem ter várias viagens associadas, mas uma Viagem apenas tem uma de cada. Por exemplo, no caso de partida do aeroporto, uma viagem está associada a uma Porta de Embarque. Um Balcão de Check-in está associado a uma ou mais companhias aéreas, e estas podem ter vários balcões também. Por fim, a um funcionário pode ter um chefe, ou ser chefe de vários funcionários.

Diagrama de classes UML



Conclusão

Nam sem libero, pulvinar in consequat vel, lacinia a urna. Fusce accumsan finibus ante eget commodo. Aenean rhoncus vitae diam ac sollicitudin. Mauris porttitor lorem et quam tristique maximus eu id erat. Etiam tempus, magna ut condimentum varius, libero est iaculis tortor, ut lobortis velit sem consectetur mauris. In erat enim, mattis ac tellus nec, mattis egestas mauris. Donec pharetra interdum mi, vel posuere justo condimentum et. In scelerisque placerat luctus. Nulla imperdiet nulla dolor, a blandit augue dignissim lacinia. Aenean eget quam volutpat, blandit tellus sit amet, volutpat augue. Phasellus sed purus vitae nibh venenatis congue at pulvinar metus. Nunc consequat urna orci, a vehicula ligula pulvinar id. Praesent quis nisl nulla. Vivamus vitae orci sit amet arcu volutpat cursus ac eu neque.

Aliquam erat volutpat. Fusce pharetra ac mauris et feugiat. Sed molestie, odio a elementum bibendum, ipsum nisi commodo ex, et malesuada felis eros id dui. Etiam ut sem sem. Fusce feugiat urna enim. Pellentesque in fringilla odio, eget condimentum ipsum. Phasellus nec interdum est. In aliquam eu leo ut vulputate. Praesent eu posuere risus, et porta lectus. In euismod ultricies augue eget cursus. Duis vel urna pharetra, suscipit metus vitae, ornare tellus. Integer ut magna orci. Etiam fermentum, tellus eget placerat mattis, nibh urna gravida orci, eget venenatis nunc ligula ut mauris. Vestibulum vehicula diam in maximus suscipit.