

**Universidade do Minho**

*Escola de Engenharia*

Departamento de Produção e Sistemas

Mestrado integrado em Engenharia Informática

Elementos de Engenharia de Sistemas

## **Projeto de Simulação**

**“Aeroporto”**

***(Diogo Araújo nº 78485***

***Francisco Oliveira nº 78416***

***Gil Cunha nº 77249***

***Vítor Peixoto nº 79175)***

Luís Dias, António Vieira  
Braga, Setembro de 2015

## Resumo

Este projeto irá simular e modelar um aeroporto. O aeroporto é *hub* fundamental à globalização que se foca essencialmente na precisão, otimização e organização de modo a tornar todo o processo eficiente.

Essas opções levam a que o Aeroporto seja uma excelente escolha para simular e modelar através do *software Arena Simulation*.

A simulação consiste na entrada de aviões (Entidades), passando pela descarga de passageiros e malas. A partir destes procedimentos os passageiros passam pelos processos de *check-out* e levantamento de malas e posteriormente abandonam o aeroporto.

Entretanto os aviões vão aos processos de abastecimento, limpeza e assistência. Desta forma o avião está disponível para próximos passageiros.

Os passageiros (de partida) chegam ao aeroporto realizam o *check-in*, seja *online/automático* ou no balcão. Depois passam pelo controlo de segurança. A partir daí os

passageiros aguardam pelo avião (enquanto as bagagens são colocadas no avião).

Todo o processo termina quando existe a partida do avião já com passageiros e sua respetiva bagagem.

## Índice

<b>Resumo .....</b>	<b>2</b>
<b>1 Introdução .....</b>	<b>4</b>
<b>2 Aeroporto .....</b>	<b>5</b>
2.1...Criação de aviões (Chegadas) .....	5
2.2...Processo de <i>check-out</i> de passageiros.....	6
2.3...Processos de tratamento do avião .....	8
2.4...Criações de pessoas (Partidas).....	11
2.5...Entrada de passageiros para avião .....	12
<b>3 Análise do desempenho .....</b>	<b>14</b>
<b>4 Conclusão.....</b>	<b>16</b>
<b>5 Informações .....</b>	<b>18</b>

## Figuras

Figura 1 - Chegada de aviões (Criação de entidades) .....	5
Figura 2 - Exemplo de <i>assign</i> de atributos.....	5
Figura 3 - <i>Assign</i> para criação de pessoas (e bagagens).....	6
Figura 4 - Início do <i>check-out</i> .....	6
Figura 5 - Bagagens consoante probabilidade.....	7
Figura 6 - Finalização do <i>check-out</i> .....	7
Figura 7 - Processos de tratamento do avião .....	8
Figura 8 - Representação gráfica das <i>stations</i> e <i>routes</i> do tratamento de aviões .....	9
Figura 9 - Aviões para as suas pistas.....	10
Figura 10 - Representação das pistas .....	10
Figura 11 - Segurança do aeroporto .....	11
Figura 12 - Check-In das pessoas .....	11
Figura 13 - Processo de entrada de passageiros.....	12
Figura 14 - Processo final do voo .....	13

# 1 Introdução

Este projeto proposto pela unidade curricular de Elementos de Engenharia de Sistemas punha ao nosso grupo como objetivo a criação de um aeroporto no *ARENA Simulation Software*.

Neste projeto tivemos em conta diversos fatores, tais como: A chegada e a saída dos aviões e a sua respetiva carga útil (100, 200 ou 300 passageiros e 200, 400 e 550 malas no porão, respetivamente); O tempo que os passageiros demoram no processo desde que entram no aeroporto até que o avião levanta voo (check-in; revisão; separação das malas; outros). E também o processo de preparação do avião para o próximo voo, incluindo o abastecimento, limpeza e assistência técnica ao avião.

A partir destes fatores, temos como dever otimizar o funcionamento do aeroporto, através do controle do tempo de chegada dos aviões, de modo a corresponder à chegada de passageiros, evitando assim, as desnecessárias e incómodas filas de espera que milhões de pessoas enfrentam diariamente em vários locais, incluindo os aeroportos. Como não é possível controlar o número de passageiros que entram num aeroporto

teremos de reduzir o processo que vai desde o *check-in* até à entrada no avião e reduzir também o processo de preparação do avião para o próximo voo, de modo a que existam mais aviões disponíveis.

Também, e de modo a tentar criar este aeroporto o mais realista possível, pensamos em adicionar pequenos detalhes do processo de embarcação, tais como meras idas à casa de banho, a um posto multibanco ou simplesmente a uma loja, que podem atrasar o tempo que a pessoa demora a realizar o trajeto desde o *check-in* até à entrada no avião.

Para melhor aplicarmos as matérias que nos foram lecionadas traçamos como objetivo adicionar animações e imagens ao projeto com dois propósitos principalmente: tornar o projeto mais nítido no que toca ao realismo de um aeroporto e para além disso tornar o projeto mais apelativo e agradável à vista.

## 2 Aeroporto

### 2.1 Criação de aviões (Chegadas)

O aeroporto tem como base fundamental os aviões, daí serem o primeiro tipo de entidades que criamos para que tudo funcione de forma correta.

Neste caso como demonstra a *Figura 1* temos a criação de três tipos de aviões em que se diferenciam através das cargas úteis de cada um.

Para tal foi colocado um *Assign* depois da criação dos aviões para lhes dar atributos como a capacidade de passageiros e bagagens (como revela a *Figura 2*). Isto foi necessário para que os próximos passageiros tivessem ao seu dispor três tipos diferentes de aviões.

De seguida separamos o avião dos seus passageiros, utilizando o *Separate*, que devolve o original (avião) que vai para os processos de tratamento pré-voo e o “duplicado” que corresponde aos passageiros que vão depois realizar o processo de *check-out* no aeroporto.

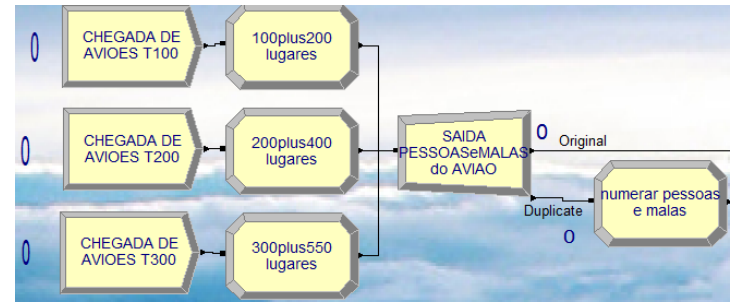


Figura 1 - Chegada de aviões (Criação de entidades)

#### Assign

Name:
100plus200 lugares
Assignments:
Attribute, lugares, 100
Attribute, porao, 200
<End of list>

Figura 2 - Exemplo de *assign* de atributos

Esse “duplicado” tem de passar a ser designado como “passageiros”, logo foi criado um *Assign* para dar certas características a esse simples “duplicado”. A partir daí, como apresenta a *Figura 3*, acaba por se atribuir um tipo de entidade, um ícone diferente e também umas variáveis de modo a controlar no futuro a entrega de malas ao seu dono corretamente.

### Assign

Name:

numerar pessoas e malas

Assignments:

Entity Picture, Picture.Person

Entity Type, PEOPLE

Attribute, meu\_numero, numerador

Variable, numerador, numerador+1

<End of list>

Figura 3 - *Assign* para criação de pessoas (e bagagens)

## 2.2 Processo de *check-out* de passageiros

O processo de *check-out* começa por dividir os passageiros e as suas respectivas bagagens (utilizando um *Separate*). Seguidamente existe um processo com um simples *Delay* para simular o tempo médio de saída dos passageiros para fora do avião em que se utilizou tempos triangulares de 2/4/8 minutos. Para as bagagens dos passageiros existe um simples *Assign* que muda o ícone da entidade para corresponder a uma bagagem.

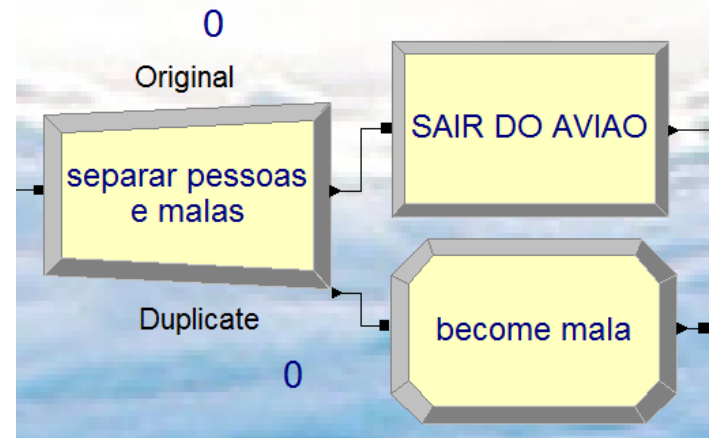


Figura 4 - Início do *check-out*

De seguida faz-se uma probabilidade da quantidade da bagagem que veio por passageiro. Desta forma utilizamos um *Decide* para colocar 50% para uma mala, 30% para 2 malas e o restante (20%) para 3 malas (como exibido na Figura 5).

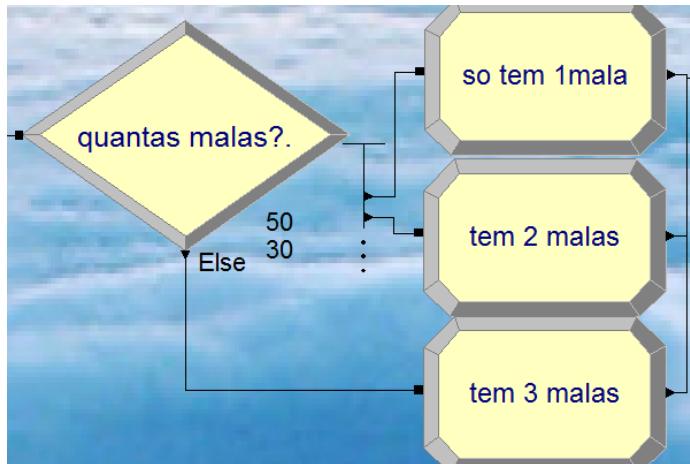


Figura 5 - Bagagens consoante probabilidade

A partir daí existe a finalização do *check-out* colocando um *Match* e *Batch*. Estes dois processos vão certificar a junção correta da bagagem e o passageiro utilizando aquele atributo que foi dado inicialmente ao passageiro e à sua bagagem. A partir daí o passageiro está pronto para sair do aeroporto.

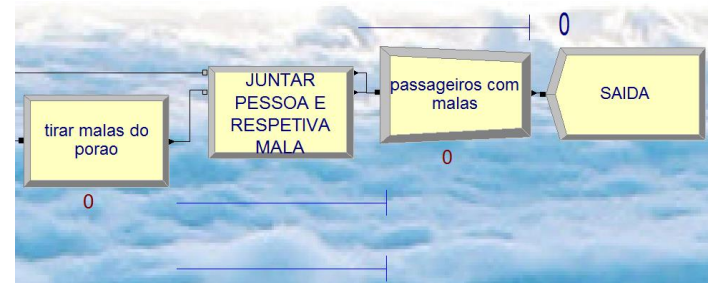


Figura 6 - Finalização do *check-out*.

## 2.3 Processos de tratamento do avião

Enquanto todo o processo de *check-out* de passageiros está a ocorrer, os aviões estão a ser tratados de forma bastante eficiente.

Como se pode observar pela *Figura 7* os aviões (já vazios) vão para a preparação do próximo voo. Aqui temos várias *stations* como:

1. Abastecimento que encaminha o avião para a rota específica que tem um tempo estipulado de 20 minutos. De seguida o avião é enviado para a Limpeza.
2. Limpeza que encaminha o avião para a rota específica que tem um tempo estipulado de 7 minutos. De seguida o avião é enviado para a Assistência.
3. Assistência que encaminha o avião para a rota específica que tem um tempo estipulado de 7 minutos. De seguida o avião é enviado para a Partida.

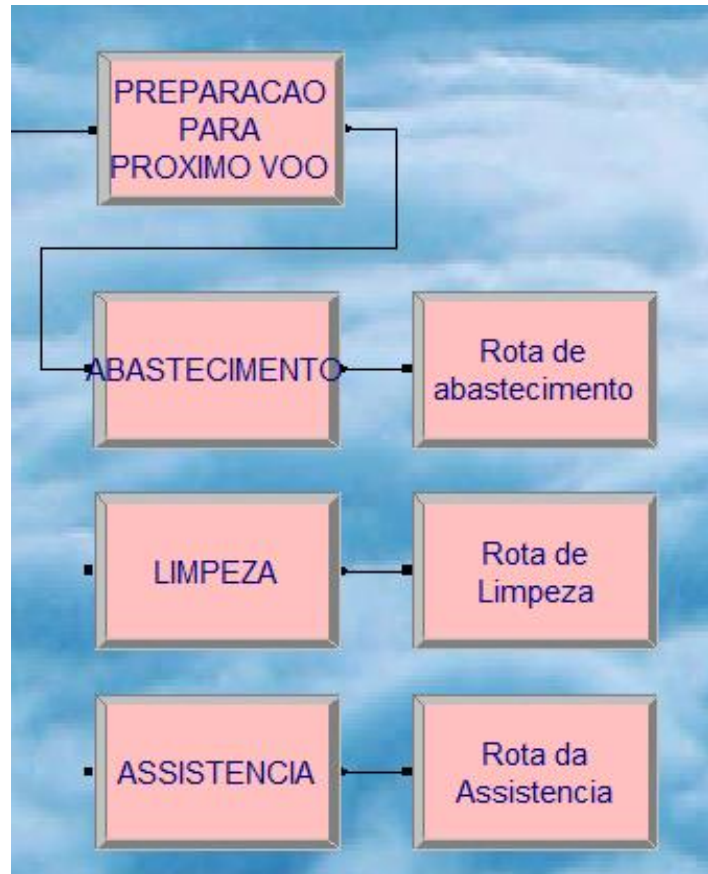


Figura 7 - Processos de tratamento do avião



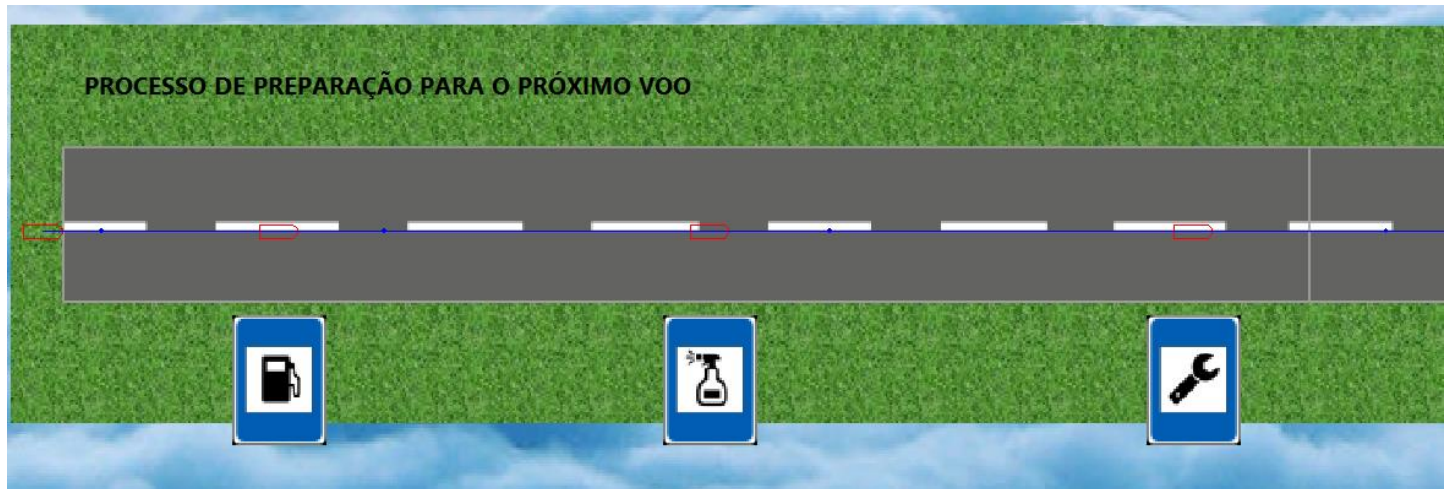


Figura 8 - Representação gráfica das *stations* e *routes* do tratamento de aviões

Seguidamente ao tratamento dos aviões eles são colocados nas suas respetivas pistas através de um *Decide* que observa o atributo do avião (que é dado no início do projeto na criação das entidades) e assim o determina. Dessa forma os aviões estão prontos para ir para a linha de embarcação respetiva.

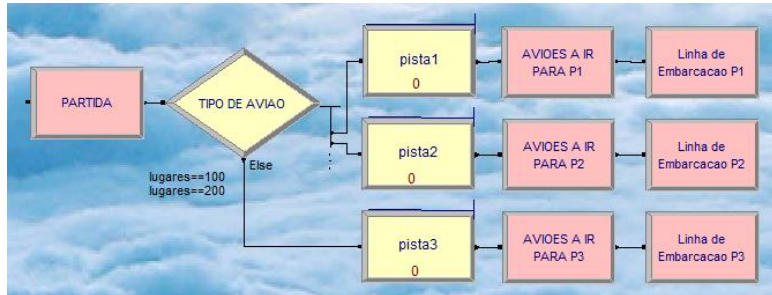


Figura 9 - Aviões para as suas pistas

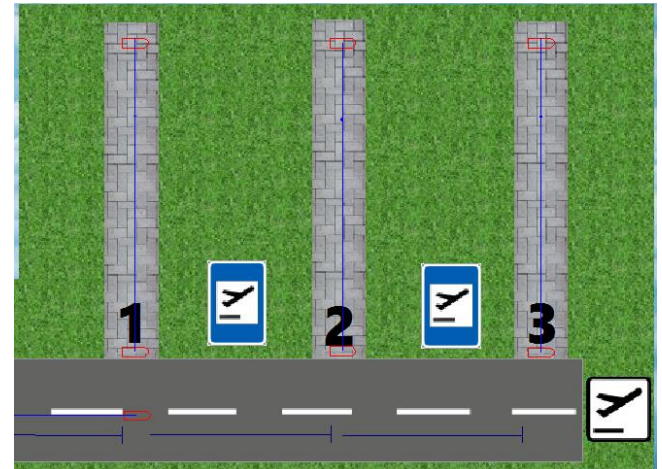


Figura 10 - Representação das pistas

## 2.4 Criações de pessoas (Partidas)

Na parte das partidas tem três entradas de futuros passageiros, que lhes vão ser dados atributos correspondentes aos tipos de avião que vão usar (Pode ser de 100/200/300 pessoas).

Posteriormente utilizamos um *Decide* (em modo de probabilidade 70%-30%) para determinar o modo de check-in que as pessoas irão utilizar.

Esses dois processos são em modo *Seize-Delay-Release* e utilizam recursos como um Funcionário ou uma Máquina respetivamente. Têm tempos triangulares de 2/3/5 e 1/2/3 respetivamente.

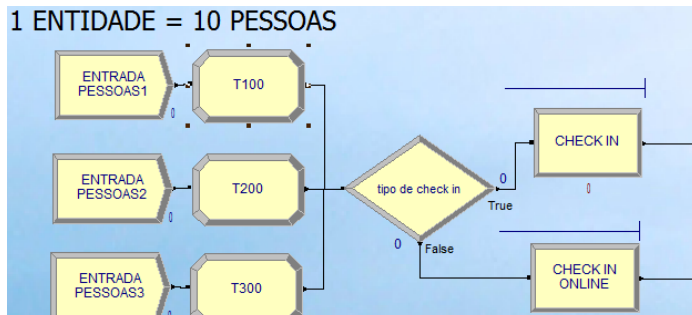


Figura 12 - Check-In das pessoas

Também colocamos a segurança em que também está em modo *Seize-Delay-Release* e utilizam um recurso que é o Segurança. Este processo tem um tempo triangular de 6/8/10 minutos.

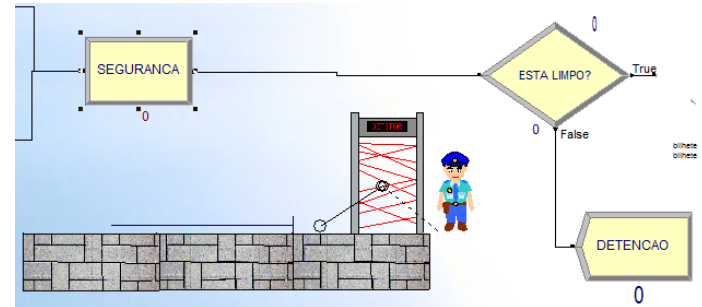


Figura 11 - Segurança do aeroporto

## 2.5 Entrada de passageiros para avião

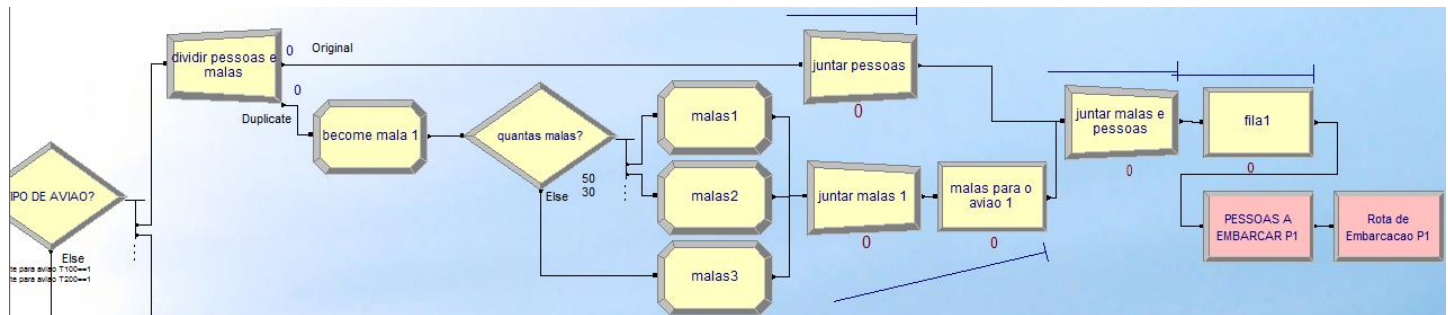


Figura 13 - Processo de entrada de passageiros

Neste processo temos a divisão das pessoas e malas em que utilizamos um *Separate*. Posteriormente criamos um *Batch* que junta grupos de pessoas (no caso da *Figura 13*, junta 10 pessoas, que equivale a 100 pessoas).

Em relação às malas é colocado um *Assign* que troca o tipo de entidade e o ícone da mesma também. Entretanto temos um *Decide* (que tem probabilidade 50%/30%/20%). Essas malas também serão juntas (no caso da *Figura 13*, junta 10 malas). O processo da colocação das malas no avião é um simples *Delay* em que tem tempo triangular de 1/2/3 minutos.

De seguida utilizou-se um *Batch* para juntar as malas e os passageiros de forma a ir para o avião.

A parte final de todo o processo se resume à *Figura 14*. Existe um *batch* final da junção entre o avião (que já estava preparado) e os passageiros e as suas bagagens. Dessa forma o avião está preparado para a decolagem e assim pronto para voar.

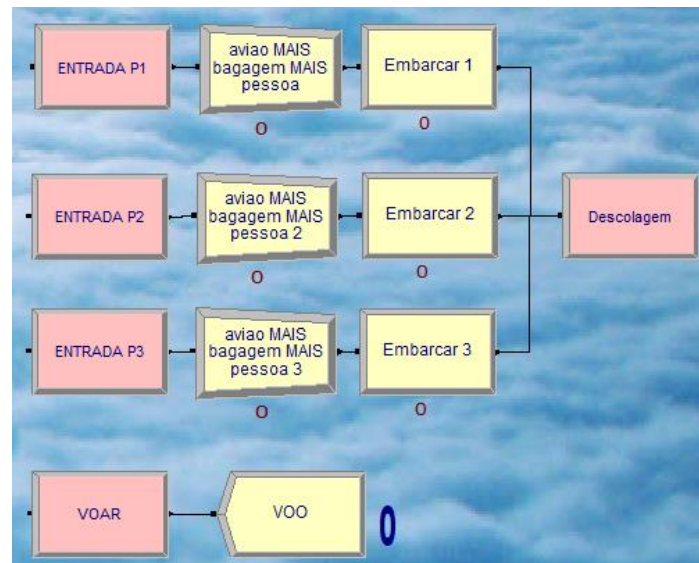


Figura 14 - Processo final do voo

### 3 Análise do desempenho

Neste projeto tivemos de adaptar o fluxo de pessoas aos horários dos aviões, porque não era possível adaptar os horários dos aviões ao fluxo de pessoas. Daí termos de controlar a entrada e saída de entidades de modo a que não haja embarço no aeroporto, isto é, haja uma transição de pessoas/aviões permanente sem ocorrer nenhum atraso e cause um congestionamento grave. Esta análise baseia-se em 16 horas de simulação feita pelo programa, sendo os valores dos dados reportados mais precisos quanto mais tempo durar a simulação.

Na primeira tentativa, verificamos que havia uma quantidade de pessoas e malas muito superior há quantidade de aviões, pelo que se formavam longas filas de espera e consequentemente uma grande acumulação de pessoas/malas, devido há falta de aviões. Com o objetivo de combater este problema, decidimos aumentar o número de aviões e restringir (controlar) a entrada de pessoas, para impedir que longas filas se formassem novamente. Todavia só estas medidas não foram necessárias, pelo que tivemos de recorrer a cálculos e comparações, de maneira a sincronizar os tempos de chegada/saída dos aviões e os processos de limpeza, abastecimento e assistência e interliga-los com o

tempo que os clientes demoram para fazer o percurso, desde a entrada no aeroporto até ao avião. Sendo o resultado:

#### Recursos

Em relação aos recursos empregados, o recurso funcionario é usado no CHECK IN, o maquina no CHECK IN ONLINE e o PROSEGUR na SEGURANÇA do aeroporto. Com estes recursos, é possível quais e quantas pessoas (entidades) passaram pelos postos referidos.

Os recursos **f1, f2 e f3** correspondem à fila 1, fila 2 e fila 3, respetivamente, assim como **p1, p2 e p3** correspondem à pista 1, pista 2 e pista 3. Estes recursos são utilizados para certificar que o Batch junta um grupo composto por pessoas e malas ao respetivo avião e que não sejam unidos (e saiam) dois aviões juntos ou dois grupos de pessoas juntas.

Quanto aos atributos:

- **meu\_numero**: serve para que posteriormente, ao utilizar o Match, as malas sejam devolvidas ao respetivo dono
- **numero\_malas**: quantas malas cada passageiro leva
- **lugares**: número de lugares
- **porão**: capacidade do porão (nº de malas)

- tipos de avião: bilhete para avião T100, bilhete para avião T200, bilhete para avião T300

## 4 Conclusão

Após uma análise exaustiva ao objetivo do nosso projeto e após diversas tentativas de encontrar a melhor solução para otimizar o funcionamento do aeroporto concluímos que, para 3 linhas de aviões com diferentes capacidades de carga (100, 200, 300 e 200, 400, 550 lugares para passageiros e malas respetivamente), com uma media de 1 avião por hora e 17 pessoas por hora, 2 funcionários e 1 máquina de check-in seria o ideal para otimizar o nosso aeroporto ao máximo, evitando assim qualquer fila de espera que tanto enfado causam às pessoas que se servem deste meio de transporte. Outros pequenos detalhes foram também adicionados para criarmos um cenário mais realista em relação aos aeroportos dos nossos dias, tal como 4 seguranças para obter a segurança necessária tanto no interior do avião, como no próprio aeroporto.

O cenário da chegada de 1 avião por hora, 17 pessoas por hora, 2 funcionários e apenas uma máquina de check-in não é o mais realista, mas o objetivo neste ponto é otimizar o funcionamento do aeroporto e para além disso, se criássemos demasiadas entidades, o software iria bloquear quando criasse 150 entidades, pois a nossa licença sob o ARENA Simulation Software é limitada à

criação de apenas 150 entidades. Assim, para obtermos um cenário mais realista teríamos de aumentar os números proporcionalmente.

Desta maneira, obtemos um aeroporto completamente funcional e eficaz, sem a criação de filas de espera e num cenário bastante realista.





## 5 Informações

Francisco José Moreira Oliveira nascido a 18/7/1997 em Santa Maria de Airão, Guimarães. Estudou na Escola Primária de Airão Santa Maria, EB 2/3 Bernardino Machado, na Escola Secundária Padre Benjamim Salgado e agora na Universidade do Minho no curso Mestrado Integrado de Engenharia Informática.

E-mail: [a78416@alunos.uminho.pt](mailto:a78416@alunos.uminho.pt)



Gil Cunha, 12 julho de 1997 (a77249@alunos.uminho.pt) Nascido em Creixomil, Guimarães, atualmente estudante universitário da Universidade do Minho, Braga, Portugal, no curso Mestrado Integrado de Engenharia Informática, tendo passado pela Escola E.B. 2,3 Egas Moniz no ensino básico e frequentado a Escola Secundária Francisco de Holanda, tendo concluído aí o secundário.



Vitor Peixoto, nascido na freguesia de Vermoim, concelho de Vila Nova de Famalicão a 30 de setembro de 1997. Iniciou o percurso escolar na EB1 Estalagem na sua freguesia natal. Prosseguiu o 2º ciclo na EB 2/3 Bernardino Machado em Joane e concluiu o 3º ciclo e ensino secundário na Escola Secundária Padre Benjamim Salgado em Joane também.

E-mail: [a79175@alunos.uminho.pt](mailto:a79175@alunos.uminho.pt)



Diogo Araújo, nascido em Braga a 8 de outubro de 1997. O meu percurso académico começou na Escola EB1 da Estrada em Ferreiros e posteriormente na Escola EB2,3 Frei Caetano Brandão terminando o ensino secundário na Escola Secundária de Maximinos.

E-mail: [a78485@alunos.uminho.pt](mailto:a78485@alunos.uminho.pt)

