



Universidade do Minho

# Agentes e Sistemas Multiagente

MEI - 1º ANO - 2º SEMESTRE

UNIVERSIDADE DO MINHO

## AEROPORTO

TRABALHO PRÁTICO 2023

URL : [HTTPS://GITHUB.COM/DUDANPM/ASM.GIT](https://github.com/DUDANPM/ASM.git)

## Grupo 06

*Trabalho realizado por:*

Duarte Moreira  
Pedro Tavares  
Tiago Costa

*Número*

PG50349  
PG50698  
PG50777

Braga, 17 de maio de 2023

# Índice

<b>1</b>	<b>Introdução</b>	<b>4</b>
<b>2</b>	<b>Contextualização</b>	<b>5</b>
<b>3</b>	<b>Características e tipos de agentes</b>	<b>6</b>
3.1	Agentes . . . . .	6
3.2	<i>Behaviours</i> . . . . .	6
3.3	<i>Performatives</i> . . . . .	7
<b>4</b>	<b>Arquitetura</b>	<b>8</b>
4.1	Protocolo geral . . . . .	8
4.2	Interações dos agentes . . . . .	9
4.2.1	Diagrama de comunicação . . . . .	9
4.2.2	Diagrama de atividades . . . . .	10
4.2.3	Diagrama de sequência . . . . .	11
<b>5</b>	<b>Resultados obtidos</b>	<b>13</b>
<b>6</b>	<b>Conclusão</b>	<b>16</b>
<b>7</b>	<b>Anexos</b>	<b>18</b>

## Lista de figuras

4.1	Diagrama de Classes . . . . .	8
4.2	Diagrama de Comunicação . . . . .	9
4.3	Diagrama de Atividades . . . . .	10
4.4	Diagrama de Sequência: operação de aterragem . . . . .	11
4.5	Diagrama de Sequência: operação de descolagem . . . . .	12
5.1	Inicialização das variáveis . . . . .	13
5.2	Estado inicial do aeroporto . . . . .	13
5.3	<i>Performatives</i> iniciais . . . . .	14
5.4	Momento em que o avião abandone a gare . . . . .	14
5.5	Momento em que o avião reúne as condições para aterrar . . . . .	15
5.6	Momento em que o avião ocupa a gare . . . . .	15
5.7	Momento em que o avião estacionado altera o seu estado e pede para descolar . . . . .	15

# Abreviaturas

*SMA* Sistemas Multiagente

*UML* *Unified Modeling Language*

# 1. Introdução

Um Sistema Multiagente compreende um conjunto de entidades que cooperam de forma a solucionar um dado problema; o que, normalmente, estará para lá das suas capacidades individuais.

A importância dos SMA assenta na especificidade de cada agente e nas suas capacidades de relacionamento. A interação entre agentes num SMA obriga, assim, à análise, definição, especificação e implementação de um certo conjunto base de funcionalidades, aqui concretizadas em termos das camadas:

- A plataforma de comunicação;
- A linguagem de comunicação;
- A ontologia;

Em resumo, será através da ativação de protocolos de comunicação que se efetuará a troca e a decodificação de mensagens. Por outro lado, são também necessários protocolos de interação que tornem possível o diálogo, i.e., a troca estruturada de informação através de mensagens[1].

O objetivo deste sistema multiagente consiste na elaboração de um conjunto de agentes inteligentes autónomos que simulem o comportamento de um aeroporto. O sistema final deverá ser capaz de gerir as chegadas e partidas de aviões num aeroporto.

Para tal, é utilizada, para o desenvolvimento de agentes em *Python*, a biblioteca *SPADE*.

Ao longo deste documento serão apresentados todos os passos e decisões tomadas para a conceção e desenvolvimento da arquitetura distribuída do sistema. Para isso, será apresentada uma contextualização geral da nossa solução, seguida da respetiva arquitetura, e por fim, o protocolo de comunicação associado aos agentes. São aplicadas metodologias *Agent UML* com o intuito de formalizar os protocolos de interação entre os agentes existentes no sistema.

## 2. Contextualização

Como mencionado anteriormente, este SMA consiste num sistema capaz de simular um aeroporto. Assim para a realização do mesmo, foi seguida a proposta apresentada no enunciado que deu origem a um total de 4 agentes: agente avião (*AviaoAgent*), agente torre de controlo (*TorreControloAgent*), agente gestor de gares (*GestorGaresAgent*) e agente de informação (*InfoAgent*).

O objetivo principal do sistema passa por, tanto a torre de controlo como o gestor de gares, conseguirem atender e satisfazer todos os pedidos que chegam por parte de aviões quer seja para aterrar e/ou descolar.

Antes de se "iniciar" o aeroporto podem ser definidas algumas constantes, como o número de gares, o número de pistas e o número de aviões a querer aterrar, assim como o número de aviões a querer descolar numa fase inicial. Como sugerido, existem dois tipos de aviões: **comerciais** e de transporte de **mercadorias**, no entanto foi adicionado pelo grupo um terceiro tipo **privado**, para simular, por exemplo, jatos privados. De igual forma, existem 3 tipos de gares, as quais apenas permitem estacionar aviões do respetivo tipo.

Após o começo do programa, são geradas as pistas, os aviões e as gares respetivamente. As pistas e as gares são geradas com posições aleatórias. Uma pista pode atender todo tipo de aviões. O tipo das gares geradas é dividido entre comerciais e de mercadorias, sendo que uma fica reservada para aviões privados. Ou seja, se o número de gares for igual a 7, então haverá 1 privada, 3 comerciais e 3 de mercadorias. O tipo dos aviões gerados é aleatório, sendo que o número de aviões com intenção de descolar é ajustado ao número de gares livres para o tipo dos mesmos. Ou seja, se forem gerados 5 aviões comerciais a querer descolar mas só houver 2 gares do tipo comercial, os restantes 3 serão rejeitados. Por outro lado os agentes torre de controlo, gestor de gares, e info são únicos, ou seja, apenas é criado um de cada para o correto funcionamento do sistema.

Como característica adicional, o grupo decidiu também implementar o sistema de forma a que os aviões estivessem sujeitos a condições atmosféricas. Estas, como esperado, afetam os tempos, tanto de aterragem como de descolagem dos aviões.

Assim sendo, o funcionamento geral deste SMA resume-se em dois fluxos principais: aterragem e descolagem. O primeiro acaba por ser o mais complexo e demorado uma vez que começa com um *landing request* do avião para a torre de controlo, que posteriormente terá de assegurar todas as condições para o mesmo aterrar e estacionar em segurança, como por exemplo, verificar o estado do tempo, a existência de uma pista livre, comunicar com o gestor para reservar uma gare, entre outros. Após reunidas as condições necessárias e o avião se encontrar na gare, passado  $x$  tempo, este transita para um estado diferente, querendo agora descolar. Este processo de descolagem é mais simples e inicia-se com um *takeoff request* do avião para o gestor. Por sua vez, este tem apenas de garantir com a torre que existe uma pista livre e que as condições atmosféricas são favoráveis à descolagem. Após realizada, o avião sai do sistema.

## 3. Características e tipos de agentes

### 3.1 Agentes

Neste SMA existem 4 tipos de agentes diferentes, sendo eles:

- **AviaoAgent**: contém um objeto da classe **aviao** para representar o seu estado.
- **GestorGaresAgent**: contém uma lista de elementos da classe *enum*, **Metereologia**, e uma lista de elementos da classe **gare**.
- **TorreControloAgent**: contém duas listas de elementos da classe **aviao**, uma lista de elementos da classe **gare** e por fim uma lista de elementos da classe **Metereologia**.
- **InfoAgent**: apenas mantém contacto com a torre de controlo para apresentar informações do aeroporto.

A construção e caracterização dos agentes, assim como, as classes envolvidas é abordada de forma mais pormenorizada na secção 4.1.

### 3.2 Behaviours

De seguida apresentam-se todos os *behaviours* utilizados ao longo do ciclo de vida de cada um dos agentes.

- AviaoAgent
  - **RequestLandingOrTakeOff**: *behaviour* do tipo *OneShotBehaviour* usado pelos aviões para informarem a torre, se quiserem aterrar, ou o gestor, se quiserem descolar, da sua intenção.
  - **AwaitLandingOrTakeOff**: *behaviour* do tipo *CyclicBehaviour* usado pelos aviões para aguardar por uma resposta afirmativa ou negativa quer da torre ou do gestor.
- GestorGaresAgent
  - **GaresInfo**: *behaviour* do tipo *OneShotBehaviour* usado pelo gestor para enviar para a torre de controlo toda a informação dos estados das gares.
  - **Control**: *behaviour* do tipo *CyclicBehaviour* usado pelo gestor para controlar basicamente todas as comunicações que faz, desde envios a receções de mensagens da torre ou de aviões com intenção de descolar.
- TorreControloAgent
  - **Info**: *behaviour* do tipo *PeriodicBehaviour* usado pela torre para comunicar a situação atual do aeroporto para o agente de informação, de **x** em **x** tempo.

- **Control:** *behaviour* do tipo *CyclicBehaviour* usado pela torre para controlar basicamente todas as comunicações que faz, desde envios a receções de mensagens do gestor ou de aviões com intenção de aterrar.
- InfoAgent
  - **Info:** *behaviour* do tipo *CyclicBehaviour* usado pelo agente para receber as informações vindas da torre e apresentá-las para o utilizador.

### 3.3 *Performatives*

De seguida apresentam-se todas as *performatives* estabelecidas para a comunicação entre os agentes dentro do sistema.

- AviaoAgent
  - **landing\_request:** pedido de aterragem.
  - **takeoff\_request:** pedido de descolagem.
  - **free\_lane\_inform:** informar a torre da desocupação de uma pista.
  - **free\_gare\_inform:** informar o gestor da desocupação de uma gare.
  - **occupied\_gare\_inform:** informar o gestor da ocupação de uma gare.
  - **change\_state\_inform:** após **x** tempo de ter estacionado, informar o gestor que pretende agora descolar.
- GestorGaresAgent
  - **gares\_inform:** informar a torre do estado das gares.
  - **free\_lane\_request:** pedido de pista livre à torre.
  - **free\_gares\_accept:** aceitar o pedido de gares livres da torre.
  - **free\_gares\_refuse:** rejeitar o pedido de gares livres da torre.
  - **takeoff\_request\_accept:** aceitar o pedido de descolagem de um avião.
  - **takeoff\_request\_refuse:** rejeitar o pedido de descolagem de um avião.
  - **gare\_request\_accept:** aceitar o pedido de uma gare por parte da torre.
- TorreControloAgent
  - **global\_inform:** informar o agente de informação do estado geral do aeroporto.
  - **gare\_request:** pedido de reserva de gare ao gestor.
  - **free\_gares\_request:** pedido de gares livres ao gestor.
  - **free\_lane\_accept:** aceitar o pedido de pista livre do gestor.
  - **free\_lane\_refuse:** rejeitar o pedido de pista livre do gestor.
  - **landing\_request\_accept:** aceitar o pedido de aterragem de um avião.
  - **landing\_request\_refuse:** rejeitar o pedido de aterragem de um avião.



## 4. Arquitetura

Ao longo deste capítulo serão apresentados todos os diagramas realizados para auxiliar na construção do sistema.

### 4.1 Protocolo geral

O primeiro diagrama, Fig 4.1, representa a versão final do diagrama de classes realizado, onde podemos ter uma visão geral da estrutura e relações das classes do SMA.

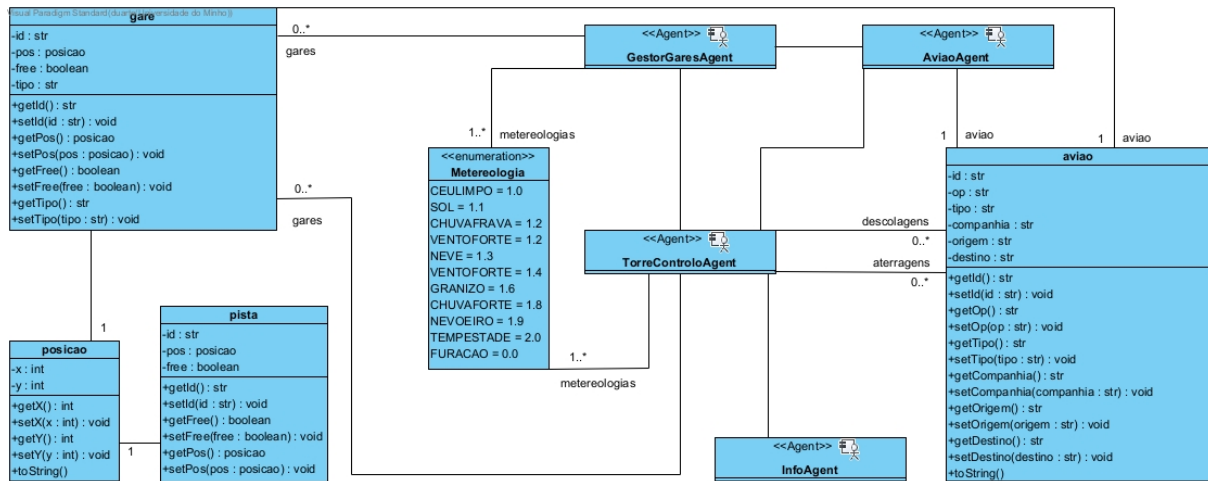


Figura 4.1: Diagrama de Classes

Como já mencionado anteriormente e como podemos ver no diagrama de classes, o sistema apresenta 4 agentes. Começando pelo **AviaoAgent**, este apresenta apenas um objeto do tipo **aviao** no seu estado. Assim cada avião terá atribuído a si um **id**, uma **operação** (aterrar, descolar, ou finalizado), um **tipo**, uma **companhia** associada, uma **origem** e um **destino**.

Já o **GestorGaresAgent** apresenta uma lista de diferentes condições meteorológicas para poder comunicar aos aviões que querem levantar voo das mesmas. Para além disso, como gestor, tem acesso a uma lista de gares para conseguir gerir o estado das mesmas. Cada **gare** é constituída por um **id**, uma **posição**, um **boolean** para indicar a ocupação da gare, um **tipo** e, se for o caso, o **avião** que lá se encontra estacionado.

O agente apresentado no centro, **TorreControloAgent**, é o que reúne mais informação, contendo duas **listas de aviões**, uma correspondente aos que querem aterrar e outra aos que querem descolar. Para além disso, contém uma **lista de gares** que é constantemente atualizada pelo gestor, e, da mesma forma que este, também tem acesso a uma lista com as diferentes condições atmosféricas.

O último agente, **InfoAgent**, não apresenta nenhuma variável, uma vez que este apenas recebe a informação transmitida pela torre de controlo e a apresenta para o utilizador.

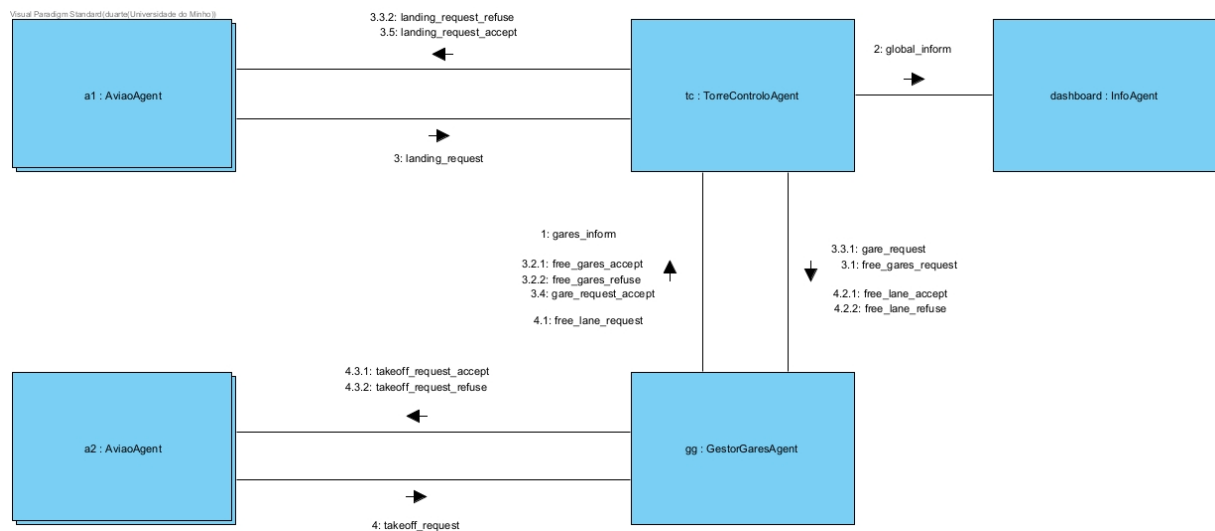
## 4.2 Interações dos agentes

Já nesta secção serão apresentados alguns diagramas que transmitem de forma clara, e alguns mais detalhada, a comunicação e comportamento interno dos diferentes agentes envolvidos no SMA.

### 4.2.1 Diagrama de comunicação

Começando pelo diagrama de comunicação, Fig 4.2, podemos observar aqui o fluxo das mensagens trocadas entres os agentes, para os 4 tipos de atividades que ocorrem no sistema, sendo elas:

1. Atividade em que o gestor partilha a informação das gares com a torre.
2. Atividade em que a torre partilha a informação global do aeroporto com o agente de informação.
3. Atividade de aterragem de um avião.
4. Atividade de decolagem de um avião.



**Figura 4.2:** Diagrama de Comunicação

## 4.2.2 Diagrama de atividades

De forma semelhante ao diagrama anterior, neste diagrama de atividades, Fig 4.3, como o próprio nome sugere, temos uma representação minimalista, da comunicação entre os agentes, nas diferentes atividades que acontecem ao longo do ciclo de vida do sistema.

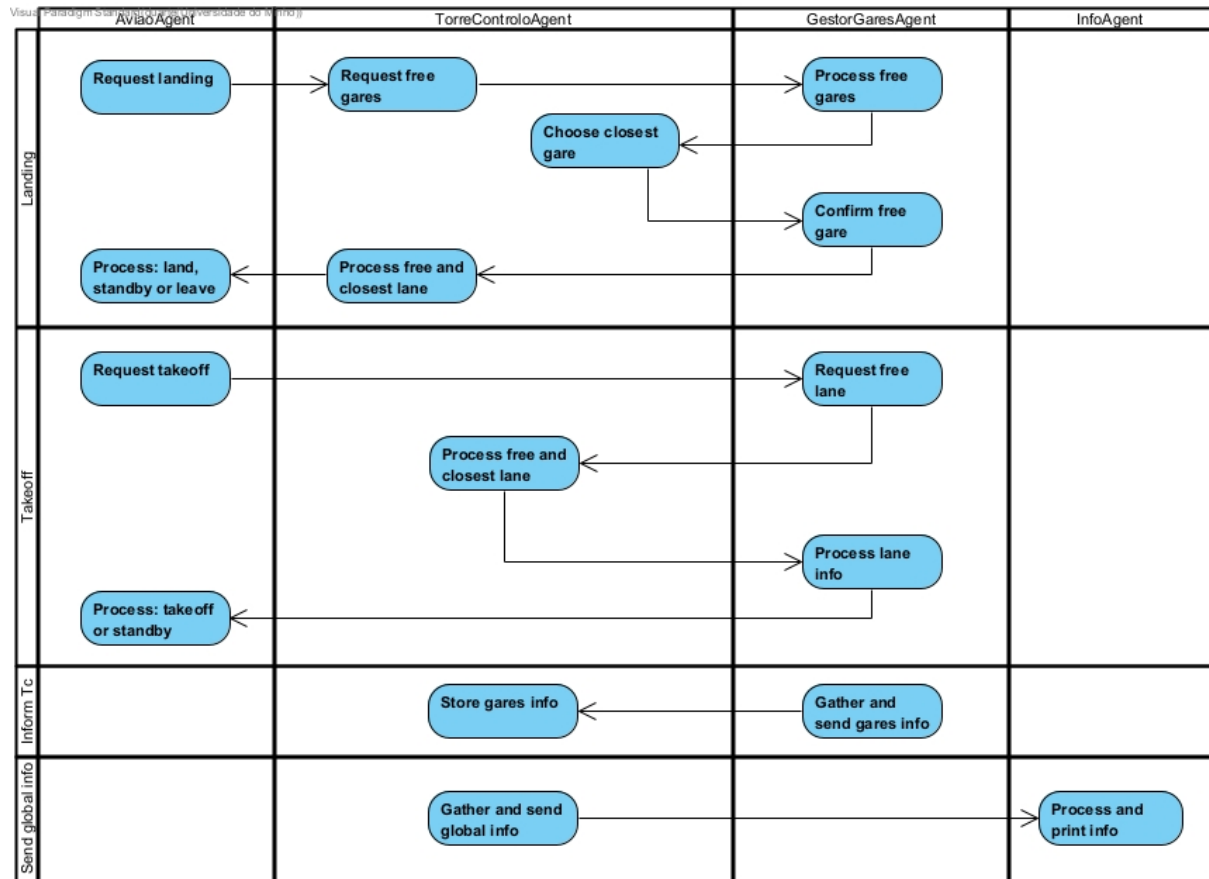
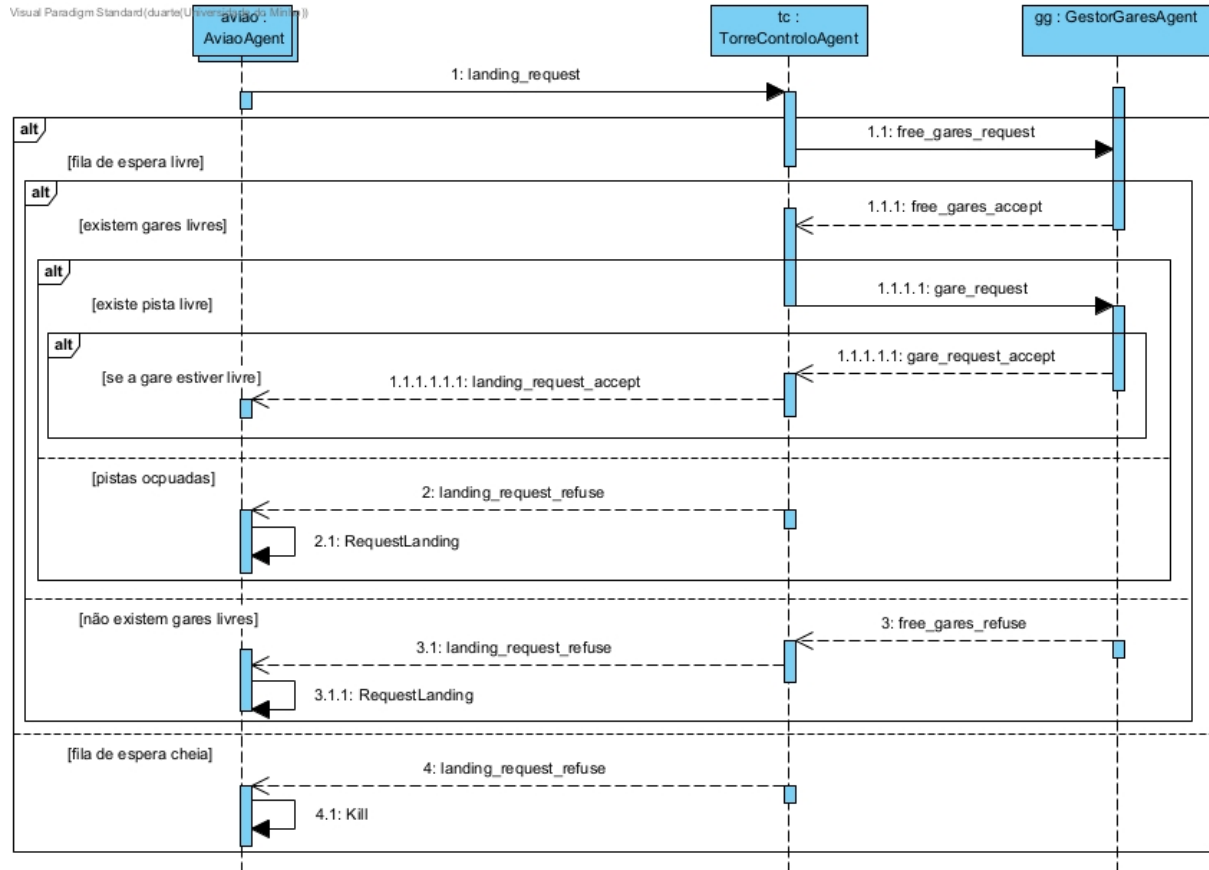


Figura 4.3: Diagrama de Atividades

### 4.2.3 Diagrama de sequência

Passando agora para um nível mais pormenorizado, na Fig 4.4, apresenta-se o diagrama de sequência associado à operação de aterragem de um avião. Aqui podemos observar todas as invocações dos métodos associados a esta operação, assim como, todos os desfechos de um pedido de aterragem.



**Figura 4.4:** Diagrama de Sequência: operação de aterragem

Numa fase inicial, após receber o pedido, a primeira condição necessária de garantir será verificar se a fila de espera não se encontra cheia na torre de controlo. Caso isso aconteça, esta informa o avião que terá que se dirigir para outro aeroporto e este acaba por abandonar o sistema.

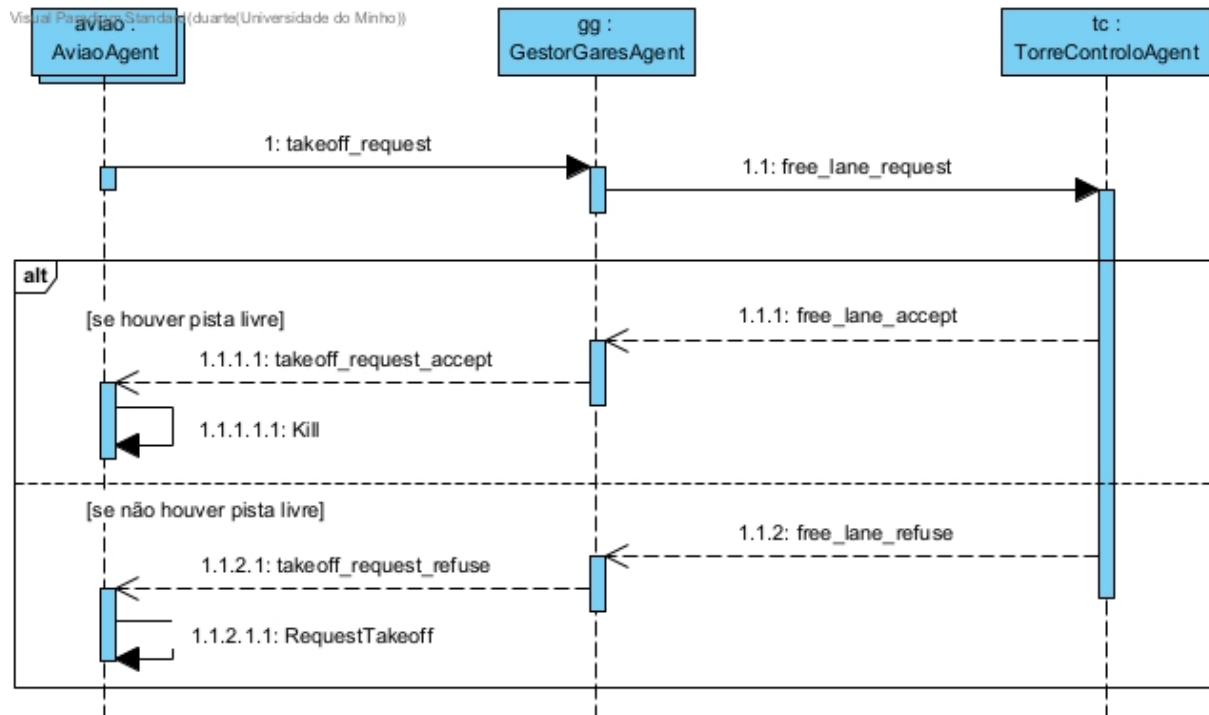
Caso contrário, a torre procede e o próximo passo será pedir ao gestor uma lista de gares livres para o avião em questão. Caso todas as gares estejam ocupadas, o gestor retornará um *refuse* à torre, que da mesma maneira, o reencaminhará para o avião. No entanto, este não abandonará o sistema mas irá aguardar *x* unidades de tempo para reenviar o pedido de aterragem.

Por outro lado, caso a torre receba uma lista de gares livres para o avião aterrar este irá calcular o par (pista,gare) mais próximos um do outro. Caso exista não exista uma pista livre, será enviado um *refuse* ao avião com o objetivo de aguardar e voltar a enviar o pedido.

Se houver uma pista livre, significa que o par foi calculado e a torre terá de confirmar com o gestor que a gare em questão não foi ocupada entretanto. Caso a gare já tenha sido ocupada (caso não representado no diagrama), o gestor recalcula a lista de gares livres e o processo repete-se, passo 1.1.

Caso contrário, o gestor reserva essa gare, comunica à torre, que por sua vez, aceitará o pedido de aterragem do avião, enviando-lhe a pista e a gare processadas.

Passando agora para o segundo diagrama de sequência realizado, Fig 4.5, este é direcionado à operação de descolagem de um avião, relativamente mais simples do que a operação vista anteriormente. Isto deve-se ao facto de que a condição necessária para um avião descolar é apenas, a necessidade de uma pista livre.



**Figura 4.5:** Diagrama de Sequência: operação de descolagem

Assim sendo, esta dá-se início num avião estacionado numa gare, que faz um pedido de descolagem ao gestor. De seguida, este realiza um pedido de pista livre à torre. No caso de haver uma pista livre, a torre transmite esse conhecimento ao gestor que por sua vez a reencaminha para o avião. A pista é reservada durante o tempo necessário para a deslocação e descolagem do mesmo. No final da operação o avião abandona o sistema.

Caso contrário, ou seja, se a torre rejeitar o pedido de pista livre ao gestor, este rejeita o pedido do avião. Após  $x$  unidades de tempo, o mesmo volta a realizar o pedido de descolagem.

## 5. Resultados obtidos

No presente capítulo serão apresentados os resultados obtidos, ou seja, o sistema em funcionamento. De forma a não tornar a visualização da informação muito excessiva, decidiu-se, demonstrar um exemplo muito simples apenas com um avião a descolar e outro a aterrar. Apesar de simplista cobre todas as atividades implementadas no sistema. Anexadas a este relatório, na secção 7, encontram-se algumas imagens que demonstram um exercício em que o sistema se encontra sobrelotado de aviões a querer aterrar e descolar.

Assim sendo, numa fase inicial, quando iniciamos o programa, Fig 5.1, definimos que queremos 3 gares, uma para cada tipo (comercial, mercadorias e privado), apenas uma pista, 1 avião a querer aterrar e 1 a querer descolar.

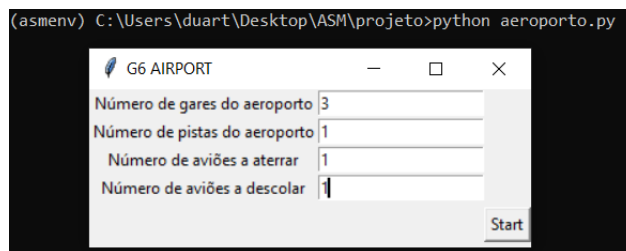


Figura 5.1: Inicialização das variáveis

Assim, o estado inicial do nosso aeroporto será o apresentado na Fig 5.2. Como esperado, temos o **aviao1** a querer aterrar, e o **aviao2** estacionado na **gare2**, correspondente ao seu tipo, a querer descolar. É de notar que a **pista1** já se encontra com o campo **Free** a **False**, isto porque, o pedido de um dos aviões já foi aceite, ou seja, a pista ficou reservada para o mesmo.

```
#####
##### LANDINGS
Aviao=[ID=aviao1@desktop-jh2ka3p Operação=aterrar Tipo=mercadorias Companhia=PEGASUS AIRLINES Origem=Lulea Kallax Destino=G6 AIRPORT]
#####
##### TAKEOFFS
Aviao=[ID=aviao2@desktop-jh2ka3p Operação=descolar Tipo=comercial Companhia=easyJet Origem=G6 AIRPORT Destino=Chiang Rai]
#####
##### GARES
Gare [ID=gare1 Posicao=(54,11) Free=True Tipo=privado Aviao=None]
Gare [ID=gare2 Posicao=(43,55) Free=False Tipo=comercial Aviao=[ID=aviao2@desktop-jh2ka3p Operação=descolar Tipo=comercial Companhia=easyJet Origem=G6 AIRPORT Destino=Chiang Rai]]
Gare [ID=gare3 Posicao=(1,76) Free=True Tipo=mercadorias Aviao=None]
#####
##### PISTAS
Pista [ID=pista1 Posicao=(2,99) Free=False]
#####
```

Figura 5.2: Estado inicial do aeroporto

Podemos confirmar o ocorrido pela Fig 5.3, onde o pedido de descolagem foi aceite, ou seja, a única pista livre teve de ser reservada para a descolagem do **aviao2**. Outra informação adicional que podemos retirar deste conjunto inicial de *performatives* recebidas pelos agentes, é o estado do tempo associado à descolagem do avião, que irá alterar o tempo da manobra.

```
(asmenv) C:\Users\duart\Desktop\ASM\projeto>python aeroporto.py
Agent info@desktop-jh2ka3p starting...
Agent torrecontrole@desktop-jh2ka3p starting...
Agent gestorgares@desktop-jh2ka3p starting...
TorreControle: gares_inform
Agent aviao1@desktop-jh2ka3p starting...
Agent aviao2@desktop-jh2ka3p starting...
TorreControle: landing_request
GestorGares: takeoff_request
GestorGares: free_gares_request
TorreControle: free_lane_request
TorreControle: free_gares_accept
GestorGares: free_lane_accept
Aviao: landing_request_refuse
Aviao: takeoff_request accept
TAKEOFF -> METEREOLOGIA: Metereologia.GRANIZO Valor: 1.6
```

Figura 5.3: *Performatives* iniciais

Na Fig 5.4, podemos observar o momento em que o **aviao2** abandona a **gare2**, informando o gestor do acontecimento. No entanto, apesar de já ter deixado a gare e esta se encontrar disponível, o mesmo não acontece para a **pista1**, uma vez que este ainda necessita dela para descolar.

```
GestorGares: free_gare_inform
TorreControle: gares_inform

#####
##### LANDINGS
#####
Aviao=[ID=aviao1@desktop-jh2ka3p Operação=aterrar Tipo=mercadorias Companhia=PEGASUS AIRLINES Origem=Lulea Kallax Destino=G6 AIRPORT]
#####
##### TAKEOFFS
#####
#####
##### GARES
#####
Gare [ID=gare1 Posicao=(54,11) Free=True Tipo=privado Aviao=None]
Gare [ID=gare2 Posicao=(43,55) Free=True Tipo=comercial Aviao=None]
Gare [ID=gare3 Posicao=(1,76) Free=True Tipo=mercadorias Aviao=None]
#####
##### PISTAS
#####
Pista [ID=pista1 Posicao=(2,99) Free=False]
#####
```

Figura 5.4: Momento em que o avião abandone a gare

Passando para a Fig 5.5, podemos aqui testemunhar o momento em que o **aviao2** libertou a pista e informou a torre. Reúnem-se agora as condições necessárias para o **aviao1** aterrar, o que pode ser confirmado nas *performatives* finais.

```
TorreControlo: free_lane_inform

#####
##### LANDINGS
#####
Aviao=[ID=aviao1@desktop-jh2ka3p Operação=aterrar Tipo=mercadorias Companhia=PEGASUS AIRLINES Origem=Lulea Kallax Destino=G6 AIRPORT]
#####
##### TAKEOFFS
#####
##### GARES
#####
Gare [ID=gare1 Posicao=(54,11) Free=True Tipo=privado Aviao=None]
Gare [ID=gare2 Posicao=(43,55) Free=True Tipo=comercial Aviao=None]
Gare [ID=gare3 Posicao=(1,76) Free=True Tipo=mercadorias Aviao=None]
#####
##### PISTAS
#####
Pista [ID=pista1 Posicao=(2,99) Free=True]
#####

TorreControlo: landing_request
GestorGares: free_gares_request
TorreControlo: free_gares_accept
GestorGares: gare_request
TorreControlo: gare_request_accept
TorreControlo: gares_inform
Aviao: landing_request_accept
LANDING -> METEREOLÓGIA: Metereologia.NEVOEIRO Valor: 1.9
```

**Figura 5.5:** Momento em que o avião reúne as condições para aterrar

Já na Fig 5.6, encontramos o momento em que o **aviao1** estaciona na gare destinada para o mesmo.

```
GestorGares: occupied_gare_inform
TorreControlo: gares_inform

#####
##### LANDINGS
#####
##### TAKEOFFS
#####
##### GARES
#####
Gare [ID=gare1 Posicao=(54,11) Free=True Tipo=privado Aviao=None]
Gare [ID=gare2 Posicao=(43,55) Free=True Tipo=comercial Aviao=None]
Gare [ID=gare3 Posicao=(1,76) Free=False Tipo=mercadorias Aviao=[ID=aviao1@desktop-jh2ka3p Operação=finalizado Tipo=mercadorias Companhia=PEGASUS AIRLINES Origem=Lulea Kallax Destino=G6 AIRPORT]]
#####
##### PISTAS
#####
Pista [ID=pista1 Posicao=(2,99) Free=True]
#####
```

**Figura 5.6:** Momento em que o avião ocupa a gare

Finalmente na Fig 5.7 temos as *performatives* associadas ao momento em que o **aviao1** que já se encontrava estacionado, passado *x* unidades de tempo, altera o seu estado, pretendendo agora descolar para abandonar o aeroporto. Assim o processo repete-se da mesma forma que aconteceu para o **aviao2**.

```
GestorGares: change_state_inform
GestorGares: takeoff_request
TorreControlo: gares_inform
TorreControlo: free_lane_request
GestorGares: free_lane_accept
Aviao: takeoff_request_accept
TAKEOFF -> METEREOLÓGIA: Metereologia.VENTOFORTE Valor: 1.4
```

**Figura 5.7:** Momento em que o avião estacionado altera o seu estado e pede para descolar



## 6. Conclusão

Dado por concluído o trabalho, acreditamos que o objetivo principal do projeto como sendo conceber e implementar um sistema multiagente que simule um aeroporto foi atingido, assim como todas as regras propostas para o funcionamento do mesmo. Apesar disso, alguns ajustes podem ainda ser aplicados, como por exemplo, tornar mais eficiente a troca de mensagens entre os agentes, e também, aplicar mais algum tipo de mecanismo de negociação (um exemplo seria ter em conta as companhias aéreas dos aviões) para tornar o sistema mais complexo e realista.

Um ponto a repensar na nossa implementação seria a forma de como o conteúdo das mensagens trocadas entre os agentes é encapsulado. O grupo está a fazê-lo recorrendo a dicionários, o que pode não ser a melhor forma de o fazer. Uma alternativa seria criar uma classe para cada tipo de mensagem diferente.

Apesar disso, o grupo faz um balanço positivo no que toca ao global do trabalho realizado.

## Bibliografia

- [1] Paulo Novais e Cesar Analide. *Agentes Inteligentes*. URL: [https://elearning.uminho.pt/webapps/blackboard/execute/content/file?cmd=view&content\\_id=\\_\\_1321044\\_\\_1&course\\_id=\\_\\_55478\\_\\_1](https://elearning.uminho.pt/webapps/blackboard/execute/content/file?cmd=view&content_id=__1321044__1&course_id=__55478__1).

## 7. Anexos

```
#####
##### LANDINGS
Aviao=[ID=aviao6@desktop-jh2ka3p Operação=aterrar Tipo=mercadorias Companhia=AIR EUROPA Origem=Tambov Destino=G6 AIRPORT]
Aviao=[ID=aviao1@desktop-jh2ka3p Operação=aterrar Tipo=comercial Companhia=SMART WINGS Origem=Coral Harbour Destino=G6 AIRPORT]
Aviao=[ID=aviao4@desktop-jh2ka3p Operação=aterrar Tipo=mercadorias Companhia=Saudia Airlines Origem=Estrasburgo Destino=G6 AIRPORT]
Aviao=[ID=aviao9@desktop-jh2ka3p Operação=aterrar Tipo=comercial Companhia=EMIRATES Origem=Altay Destino=G6 AIRPORT]
Aviao=[ID=aviao2@desktop-jh2ka3p Operação=aterrar Tipo=mercadorias Companhia=RYANAIR Origem=Ulyanovsk Baratayevka Destino=G6 AIRPORT]
#####
##### TAKEOFFS
Aviao=[ID=aviao12@desktop-jh2ka3p Operação=descolar Tipo=comercial Companhia=EUROWINGS Origem=G6 AIRPORT Destino=Dillingham]
Aviao=[ID=aviao18@desktop-jh2ka3p Operação=descolar Tipo=comercial Companhia=AEGEAN Origem=G6 AIRPORT Destino=Tiksi]
Aviao=[ID=aviao11@desktop-jh2ka3p Operação=descolar Tipo=mercadorias Companhia=EMIRATES Origem=G6 AIRPORT Destino=Istanbul]
Aviao=[ID=aviao13@desktop-jh2ka3p Operação=descolar Tipo=mercadorias Companhia=FLYDUBAI Origem=G6 AIRPORT Destino=Lulea Kallax]
#####
##### GARES
Gare [ID=gare1 Posicao=(97,6) Free=True Tipo=privado Aviao=None]
Gare [ID=gare2 Posicao=(36,91) Free=False Tipo=comercial Aviao=[ID=aviao12@desktop-jh2ka3p Operação=descolar Tipo=comercial Companhia=EUROWINGS Origem=G6 AIRPORT Destino=Dillingham]]
Gare [ID=gare3 Posicao=(92,16) Free=False Tipo=comercial Aviao=[ID=aviao18@desktop-jh2ka3p Operação=descolar Tipo=comercial Companhia=AEGEAN Origem=G6 AIRPORT Destino=Tiksi]]
Gare [ID=gare4 Posicao=(78,5) Free=False Tipo=mercadorias Aviao=[ID=aviao11@desktop-jh2ka3p Operação=descolar Tipo=mercadorias Companhia=EMIRATES Origem=G6 AIRPORT Destino=Istanbul]]
Gare [ID=gare5 Posicao=(92,61) Free=False Tipo=mercadorias Aviao=[ID=aviao13@desktop-jh2ka3p Operação=descolar Tipo=mercadorias Companhia=FLYDUBAI Origem=G6 AIRPORT Destino=Lulea Kallax]]
#####
##### PISTAS
Pista [ID=pista1 Posicao=(6,44) Free=False]
Pista [ID=pista2 Posicao=(30,94) Free=False]
Pista [ID=pista3 Posicao=(73,92) Free=False]
#####
```

```
GestorGares: free_gare_inform
TorreControlo: gares_inform
GestorGares: free_gare_inform
TorreControlo: gares_inform

#####
##### LANDINGS
Aviao=[ID=aviao6@desktop-jh2ka3p Operação=aterrar Tipo=mercadorias Companhia=AIR EUROPA Origem=Tambov Destino=G6 AIRPORT]
Aviao=[ID=aviao1@desktop-jh2ka3p Operação=aterrar Tipo=comercial Companhia=SMART WINGS Origem=Coral Harbour Destino=G6 AIRPORT]
Aviao=[ID=aviao4@desktop-jh2ka3p Operação=aterrar Tipo=mercadorias Companhia=Saudia Airlines Origem=Estrasburgo Destino=G6 AIRPORT]
Aviao=[ID=aviao9@desktop-jh2ka3p Operação=aterrar Tipo=comercial Companhia=EMIRATES Origem=Altay Destino=G6 AIRPORT]
Aviao=[ID=aviao2@desktop-jh2ka3p Operação=aterrar Tipo=mercadorias Companhia=RYANAIR Origem=Ulyanovsk Baratayevka Destino=G6 AIRPORT]
#####
##### TAKEOFFS
Aviao=[ID=aviao12@desktop-jh2ka3p Operação=descolar Tipo=comercial Companhia=EUROWINGS Origem=G6 AIRPORT Destino=Dillingham]
#####
##### GARES
Gare [ID=gare1 Posicao=(97,6) Free=True Tipo=privado Aviao=None]
Gare [ID=gare2 Posicao=(36,91) Free=False Tipo=comercial Aviao=[ID=aviao12@desktop-jh2ka3p Operação=descolar Tipo=comercial Companhia=EUROWINGS Origem=G6 AIRPORT Destino=Dillingham]]
Gare [ID=gare3 Posicao=(92,16) Free=True Tipo=comercial Aviao=None]
Gare [ID=gare4 Posicao=(78,5) Free=True Tipo=mercadorias Aviao=None]
Gare [ID=gare5 Posicao=(92,61) Free=True Tipo=mercadorias Aviao=None]
#####
##### PISTAS
Pista [ID=pista1 Posicao=(6,44) Free=False]
Pista [ID=pista2 Posicao=(30,94) Free=False]
Pista [ID=pista3 Posicao=(73,92) Free=False]
#####
```

```

TorreControlo: landing_request
TorreControlo: landing_request
Aviao: landing_request_refuse
Agent aviao10@desktop-jh2ka3p: vou dirigir-me para outro aeroporto.
TorreControlo: landing_request
Aviao: landing_request_refuse
Agent aviao5@desktop-jh2ka3p: vou dirigir-me para outro aeroporto.
GestorGares: free_gares_request
TorreControlo: landing_request
GestorGares: takeoff_request
TorreControlo: landing_request
GestorGares: free_gares_request
Aviao: takeoff_request_refuse
TorreControlo: landing_request
GestorGares: free_gares_request
TorreControlo: landing_request
GestorGares: free_gares_request
TorreControlo: free_gares_accept
GestorGares: free_gares_request
TorreControlo: free_gares_accept
Aviao: landing_request_refuse
TorreControlo: free_gares_accept
Aviao: landing_request_refuse
TorreControlo: free_gares_accept
Aviao: landing_request_refuse
TorreControlo: free_gares_accept
Aviao: landing_request_refuse
TorreControlo: free_gares_accept
Aviao: landing_request_refuse
Aviao: landing_request_refuse

```

```

TorreControlo: free_lane_inform
TorreControlo: landing_request
GestorGares: takeoff_request
TorreControlo: landing_request
GestorGares: free_gares_request
TorreControlo: landing_request
GestorGares: free_gares_request
TorreControlo: landing_request
GestorGares: free_gares_request
TorreControlo: landing_request
GestorGares: free_gares_request
TorreControlo: free_lane_request
GestorGares: free_gares_request
TorreControlo: free_gares_accept
GestorGares: free_lane_accept
TorreControlo: free_gares_accept
GestorGares: gare_request
Aviao: takeoff_request_accept
TAKEOFF -> METEOROLOGIA: Metereologia.SOL Valor: 1.1
TorreControlo: free_gares_accept
GestorGares: gare_request
TorreControlo: free_gares_accept
Aviao: landing_request_refuse
TorreControlo: free_gares_accept
Aviao: landing_request_refuse
TorreControlo: gare_request_accept
Aviao: landing_request_refuse
TorreControlo: gares_inform
Aviao: landing_request_accept
LANDING -> METEOROLOGIA: Metereologia.CHUVAFRACA Valor: 1.2
TorreControlo: gare_request_accept
TorreControlo: gares_inform
Aviao: landing_request_accept
LANDING -> METEOROLOGIA: Metereologia.NEVE Valor: 1.3

```

```

=====
===== LANDINGS
=====
Aviao[ID=aviao4@desktop-jh2ka3p Operacao=aterrar Tipo=mercadorias Companhia=Saudia Airlines Origem=Estrasburgo Destino=G6 AIRPORT]
Aviao[ID=aviao9@desktop-jh2ka3p Operacao=aterrar Tipo=comercial Companhia=EMIRATES Origem=Altay Destino=G6 AIRPORT]
Aviao[ID=aviao2@desktop-jh2ka3p Operacao=aterrar Tipo=mercadorias Companhia=RYANAIR Origem=Ulyanovsk Baratayevka Destino=G6 AIRPORT]
=====
===== TAKEOFFS
=====
===== GARES
=====
Gare [ID=gare1 Posicao=(97,6) Free=True Tipo=privado Aviao=None]
Gare [ID=gare2 Posicao=(36,91) Free=True Tipo=comercial Aviao=None]
Gare [ID=gare3 Posicao=(92,16) Free=False Tipo=comercial Aviao=[ID=aviao1@desktop-jh2ka3p Operacao=finalizado Tipo=comercial Companhia=SMART WINGS Origem=Coral Harbour Destino=G6 AIRPORT]]
Gare [ID=gare4 Posicao=(78,5) Free=False Tipo=mercadorias Aviao=[ID=aviao6@desktop-jh2ka3p Operacao=finalizado Tipo=mercadorias Companhia=AIR EUROPA Origem=Tamboy Destino=G6 AIRPORT]]
Gare [ID=gare5 Posicao=(92,61) Free=False Tipo=mercadorias Aviao=None]
=====
===== PISTAS
=====
Pista [ID=pista1 Posicao=(6,44) Free=True]
Pista [ID=pista2 Posicao=(30,94) Free=True]
Pista [ID=pista3 Posicao=(73,92) Free=False]
=====

```

```

#####
##### LANDINGS
#####
Aviao=[ID=aviao4@desktop-jh2ka3p Operação=aterrar Tipo=mercadorias Companhia=Saudia Airlines Origem=Estrasburgo Destino=G6 AIRPORT]
Aviao=[ID=aviao0@desktop-jh2ka3p Operação=aterrar Tipo=comercial Companhia=EMIRATES Origem=Altay Destino=G6 AIRPORT]
Aviao=[ID=aviao2@desktop-jh2ka3p Operação=aterrar Tipo=mercadorias Companhia=RYANAIR Origem=Ulyanovsk Baratayevka Destino=G6 AIRPORT]
#####
##### TAKEOFFS
#####
Aviao=[ID=aviao1@desktop-jh2ka3p Operação=descolar Tipo=comercial Companhia=SMART WINGS Origem=G6 AIRPORT Destino=International Falls]
Aviao=[ID=aviao0@desktop-jh2ka3p Operação=descolar Tipo=mercadorias Companhia=AIR EUROPA Origem=G6 AIRPORT Destino=Wolf Point]
#####
##### GARES
#####
Gare [ID=gare1 Posicao=(97,6) Free=True Tipo=privado Aviao=None]
Gare [ID=gare2 Posicao=(36,91) Free=True Tipo=comercial Aviao=None]
Gare [ID=gare3 Posicao=(92,16) Free=False Tipo=comercial Aviao=[ID=aviao1@desktop-jh2ka3p Operação=descolar Tipo=comercial Companhia=SMART WINGS Origem=G6 AIRPORT Destino=International Falls]]
Gare [ID=gare4 Posicao=(78,5) Free=False Tipo=mercadorias Aviao=[ID=aviao0@desktop-jh2ka3p Operação=descolar Tipo=mercadorias Companhia=AIR EUROPA Origem=G6 AIRPORT Destino=Wolf Point]]
Gare [ID=gare5 Posicao=(92,61) Free=False Tipo=mercadorias Aviao=None]
#####
##### PISTAS
#####
Pista [ID=pista1 Posicao=(6,44) Free=False]
Pista [ID=pista2 Posicao=(30,94) Free=False]
Pista [ID=pista3 Posicao=(73,92) Free=False]
#####
TorreControlo: landing_request
TorreControlo: landing_request
Aviao: landing_request_refuse
GestorGares: free_gares_request
TorreControlo: free_gares_refuse
Aviao: landing_request_refuse
GestorGares: free_gare_inform
TorreControlo: gares_inform

```

```

#####
##### LANDINGS
#####
Aviao=[ID=aviao4@desktop-jh2ka3p Operação=aterrar Tipo=mercadorias Companhia=Saudia Airlines Origem=Estrasburgo Destino=G6 AIRPORT]
#####
##### TAKEOFFS
#####
Aviao=[ID=aviao2@desktop-jh2ka3p Operação=descolar Tipo=mercadorias Companhia=RYANAIR Origem=G6 AIRPORT Destino=Damasco]
#####
##### GARES
#####
Gare [ID=gare1 Posicao=(97,6) Free=True Tipo=privado Aviao=None]
Gare [ID=gare2 Posicao=(36,91) Free=False Tipo=comercial Aviao=[ID=aviao9@desktop-jh2ka3p Operação=finalizado Tipo=comercial Companhia=EMIRATES Origem=Altay Destino=G6 AIRPORT]]
Gare [ID=gare3 Posicao=(92,16) Free=True Tipo=comercial Aviao=None]
Gare [ID=gare4 Posicao=(78,5) Free=False Tipo=mercadorias Aviao=None]
Gare [ID=gare5 Posicao=(92,61) Free=False Tipo=mercadorias Aviao=[ID=aviao2@desktop-jh2ka3p Operação=descolar Tipo=mercadorias Companhia=RYANAIR Origem=G6 AIRPORT Destino=Damasco]]
#####
##### PISTAS
#####
Pista [ID=pista1 Posicao=(6,44) Free=False]
Pista [ID=pista2 Posicao=(30,94) Free=True]
Pista [ID=pista3 Posicao=(73,92) Free=True]
#####

```

```

TorreControlo: free_lane_inform

#####
##### LANDINGS
#####
##### TAKEOFFS
#####
##### GARES
#####
Gare [ID=gare1 Posicao=(97,6) Free=True Tipo=privado Aviao=None]
Gare [ID=gare2 Posicao=(36,91) Free=True Tipo=comercial Aviao=None]
Gare [ID=gare3 Posicao=(92,16) Free=True Tipo=comercial Aviao=None]
Gare [ID=gare4 Posicao=(78,5) Free=True Tipo=mercadorias Aviao=None]
Gare [ID=gare5 Posicao=(92,61) Free=True Tipo=mercadorias Aviao=None]
#####
##### PISTAS
#####
Pista [ID=pista1 Posicao=(6,44) Free=True]
Pista [ID=pista2 Posicao=(30,94) Free=True]
Pista [ID=pista3 Posicao=(73,92) Free=True]
#####

```