



Universidade do Minho

MESTRADO INTEGRADO EM ENGENHARIA
INFORMÁTICA

SISTEMAS INTELIGENTES: AGENTES INTELIGENTES

AGENTES E SISTEMAS
MULTIAGENTE: SISTEMAS DE
PARTILHA DE BICICLETAS

GRUPO 5

A84656 - HUGO CUNHA
A86268 - MARIA PIRES
A85242 - MARIA REGUEIRAS
A84167 - SUSANA MARQUES

3 DE JANEIRO DE 2021

Conteúdo

1	Introdução	5
1.1	Estrutura do documento	5
2	Contextualização	6
3	Agentes	7
3.1	Agente Estação	7
3.2	Agente Utilizador	8
3.2.1	Características	8
3.3	Agente Monitor	8
3.4	Agente Interface	8
3.4.1	Características	8
4	Proposta de Resolução Inicial	9
4.1	Conflitos	9
4.2	Arquitetura do sistema	10
4.3	Protocolos de Comunicação	11
4.3.1	Protocolo entre Agente Estação e Agente Utilizador	11
4.3.2	Protocolo entre Agente Estação e Agente Monitor	12
4.3.3	Protocolo entre Agente Estação e Agente Interface	12
4.3.4	Protocolo entre Agente Utilizador e Agente Monitor	13
4.3.5	Protocolo entre Agente Utilizador e Agente Interface	13
4.4	Estado interno do Agente Utilizador	14
5	Alterações à Proposta de Resolução Inicial	15
5.1	Arquitetura do Sistema	15
5.2	Protocolos de Comunicação	16
5.3	Estado interno do Agente Utilizador	16
6	Implementação	18
6.1	Inicialização do Sistema e dos Agentes	18
6.2	Realização de uma Viagem	18
6.3	Caso de Emergência	20
6.4	Continuidade da Simulação	20
7	Interface	21
7.1	Taxa de ocupação das estações	21
7.2	Número de utilizadores no Sistema	22
7.3	Distribuição de preços por estação	22

Lista de Figuras

4.1	Arquitetura do Sistema	10
4.2	Panorama geral de comunicação entre agentes	11
4.3	Comunicação Agente Estação e Agente Utilizador	12
4.4	Comunicação Agente Estação e Agente Monitor	12
4.5	Comunicação Agente Estação e Agente Interface	13
4.6	Comunicação Agente Utilizador e Agente Monitor	13
4.7	Comunicação Agente Utilizador e Agente Interface	14
4.8	Estado interno do Agente Utilizador	14
5.1	Arquitetura do Sistema	15
5.2	Panorama geral da comunicação entre agentes	16
5.3	Estado interno do Agente Utilizador	17
7.1	Interface final	21
7.2	Taxa de ocupação das estações	22
7.3	Número de utilizadores no Sistema	22
7.4	Distribuição de preços por estação	23

Lista de Tabelas

4.1	Taxa de ocupação das estações de bicicletas	10
4.2	Preços a praticar em função da taxa de ocupação	10

1 | Introdução

O presente relatório foi elaborado no âmbito da unidade curricular *Agentes Inteligentes*, integrada no perfil de especialização *Sistemas Inteligentes*, onde será exposto o processo de resolução da primeira fase do trabalho prático. O objetivo principal deste projeto é a conceção e desenvolvimento de uma sistema multiagente (SMA), que permite aos utilizadores alugar bicicletas para realizar viagens curtas. Os utilizadores alugam e devolvem as bicicletas em estações de bicicleta dedicadas, podendo surgir o problema de reequilíbrio da partilha (PRPB), isto é, desequilíbrios na gestão de ocupação da estação, pelo que é necessário manter um estado equilibrado. Nesta segunda fase, o objetivo seria implementar a arquitetura concebida inicialmente, bem como uma interface gráfica que permite a visualização dos eventos do sistema.

1.1 Estrutura do documento

O relatório encontra-se dividido em oito partes. Em primeiro lugar, na secção 1, *Introdução*, são apresentados os objetivos e o tema proposto.

Na secção 2 é elaborada uma contextualização referente ao projeto.

De seguida, na secção 3, são descritos os diversos agentes que irão integrar o SMA final. Na secção 4 é apresentada a proposta inicial de uma arquitetura para resolver o problema. De seguida, na secção, 5 são expostas as alterações efetuadas à arquitetura aquando da sua implementação.

Na secção 6 é descrito o processo de construção da arquitetura concebida.

Na secção 7, *Interface*, são apresentados os componentes gráficos desenvolvidos que permitem observar o estado do sistema. Finalmente, na secção 8 conclui-se esta segunda fase e apresenta-se uma perspetiva do trabalho futuro.

2 | Contextualização

Recentemente, face ao aumento do aquecimento global e poluição em geral, a procura por transportes sustentáveis incrementou substancialmente e parte da população começou a deslocar-se no seu dia-a-dia de bicicleta, pois para além do impacto ambiental nulo, estes meios de transporte são também mais económicos, saudáveis e facilitam a movimentação em meios urbanos em pequenas distancias. Assim, foram nascendo sistemas de aluguer e partilha de bicicletas. Existem diversas modalidades de aluguer de bicicletas, contudo o Sistema de Partilha de Bicicletas (SPB) consiste na disponibilização de bicicletas em várias estações espalhadas num mapa para que os ciclistas possam usufruir destas durante um período de tempo para se deslocar até ao seu destino.

3 | Agentes

Após a análise do problema proposto concebeu-se que o SMA seria composto pelos seguintes agentes:

- Agente de Estação;
- Agente Utilizador;
- Agente Monitor;
- Agente Interface.

De seguida serão descritas detalhadamente as características dos agentes enumerados nesta secção.

3.1 Agente Estação

Agente que será responsável por caracterizar uma estação do Sistema de Partilha de Bicicletas, mantendo os seus dados devidamente atualizados. Este apresenta a capacidade de interagir com o agente Interface para atualizar o estado do primeiro e com o Agente Monitor para verificar se um agente Utilizador se encontra na área de Proximidade do Agente Estação. Para caracterizar esta área o agente usa o raio de uma circunferência em metros como medida necessária para tal.

- Posição: Característica que representa a localização da estação no mapa. É caracterizada por dois valores que representam os dois eixos de um modelo 2D: *Eixo x*; *Eixo y*.
- Capacidade Atual: Valor que representa o número atual de bicicletas que estão na estação.
- Limite: Valor que representa o número máximo de bicicletas que a estação pode guardar.
- isFull: Característica que permite saber se uma estação está cheia para **não** permitir que a mesma seja indicada a um utilizador como estação adequada para deixar a bicicleta.

3.2 Agente Utilizador

O Agente Utilizador corresponde ao papel que um utilizador que alugue uma bicicleta irá ter no sistema. Este interage com o Agente Monitor para lhe informar a sua localização e para averiguar se a mesma pertence à APE de alguma estação e com o Agente Interface para também lhe informar a sua localização atual.

3.2.1 Características

- **Posição Atual:** Posição que será atualizada periodicamente, informando assim conhecimento da localização atual do utilizador. Tal como no agente Estação, a posição atual do Agente Utilizador é caracterizada no formato: **Eixo x; Eixo y**.
- **Destino:** Posição que o utilizador decide como seu destino final no formato: *Eixo x; Eixo y*.
- **Personalidade:** Aspecto que irá influenciar a decisão relativa às ofertas do sistema. Esta personalidade indica o preço e distancia máximos que uma utilizador está disposto a aceitar durante o processo de devolução de uma bicicleta a uma estação.

3.3 Agente Monitor

Este agente tem o propósito de verificar se a posição de um utilizador se encontra numa APE de alguma estação e caso seja verdade, informar essa estação para que haja uma proposta da mesma.

- **Estações:** Informação sobre todas as estações que se encontram ativas no sistema.

3.4 Agente Interface

A existência deste agente tem como objetivo a representação visual dos vários aspetos de cada uma das estações, de forma a auxiliar o utilizador na sua tomada de decisão.

3.4.1 Características

- **Estações:** Informação sobre todas as estações que se encontram ativas no sistema.
- **Utilizador:** Informação sobre a posição do utilizador no sistema.

4 | Proposta de Resolução Inicial

Após a apresentação dos agentes que constituirão o SMA, neste capítulo será exposta a forma como estes interagem uns com os outros criando um sistema dinâmico e distribuído.

4.1 Conflitos

O grupo iniciou a concepção da arquitetura através a exploração dos vários conflitos que podem surgir no sistema. Serão listados, de seguida, os problemas apurados e a proposta de resolução destes. É expectável que no decorrer da implementação do projeto surjam conflitos adicionais.

Conflito: No caso de uma estação ter uma taxa de ocupação muito alta é necessário reequilibrar o sistema.

- **Resolução:** Aumentar o preço de depósito na estação.
- **Vantagens:** Menor fluxo de bicicletas para aquela estação.
- **Desvantagens:** Estações isoladas forçam o utilizador a pagar mais ou ficar à espera.

Conflito: No caso de uma estação ter uma taxa de ocupação muito baixa é necessário garantir que, caso os utilizadores queiram alugar bicicletas nessa estação, haja bicicletas disponíveis.

- **Resolução:** Diminuir o preço de depósito na estação.
- **Vantagens:** Maior fluxo de bicicletas para aquela estação.
- **Desvantagens:** Estações isoladas forçam o utilizador a ficar à espera.

Conflito: No caso de um utilizador receber propostas igualmente vantajosas.

- **Resolução:** Depende da personalidade do utilizador.
- **Vantagens:** Variabilidade garantida na população.
- **Desvantagens:** Imprevisibilidade no sistema.

Definiu-se que a taxa de ocupação de uma estação obedeceria aos valores presentes na seguinte tabela.

Taxa de Ocupação	Valor
Alto	80%
Médio	20-80%
Baixo	20%

Tabela 4.1: Taxa de ocupação das estações de bicicletas

Para resolver os conflitos que surgem com a escassez ou excesso de bicicletas presentes numa estação os valores do aluguer a pagar pelo cliente são calculados através da multiplicação de um preço base (PB) com um valor estabelecido, dependente da taxa de ocupação da estação em causa.

Taxa de Ocupação	Preço
Alto	$PB \times 1.8$
Médio	$PB \times 1$
Baixo	$PB \times 0,2$

Tabela 4.2: Preços a praticar em função da taxa de ocupação

4.2 Arquitetura do sistema

Nesta secção apresenta-se a arquitetura concebida numa fase inicial para o sistema através do diagrama de classes presente na figura 4.1.

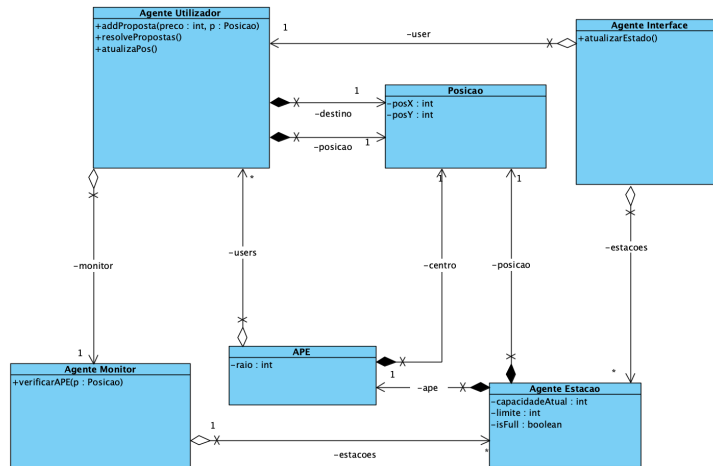


Figura 4.1: Arquitetura do Sistema

Determinadas relações identificadas serão abstraídas através dos mecanismos de comunicação do JADE, e é bastante provável que ao desenvolver o projeto surjam novas relações ou as apresentadas inicialmente sejam alteradas.

4.3 Protocolos de Comunicação

Para que o SMA se comporte de forma correta e dinâmica é necessário que os agentes comuniquem entre si e respeitem os protocolos de comunicação estabelecidos. Na figura 4.2 é apresentado o panorama geral de comunicação entre os agentes, no qual se pode observar as *performatives* empregues entre cada par.

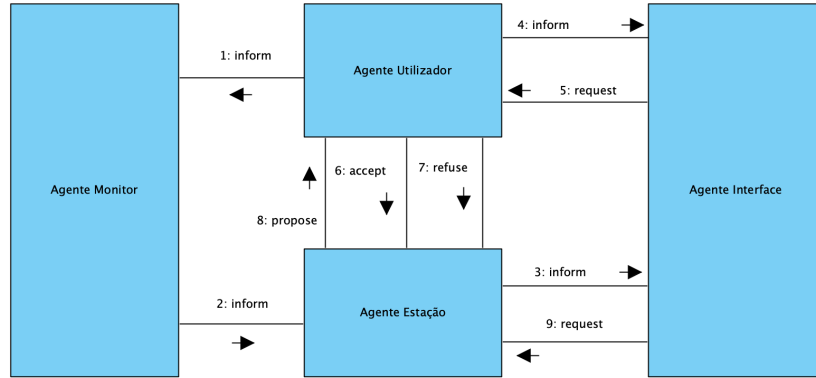


Figura 4.2: Panorama geral de comunicação entre agentes

O Agente Utilizador comunica com o Agente Monitor através de mensagens do tipo *inform*, indicando-lhe, a sua posição no momento e aquela que pretende alcançar. O Agente Monitor comunica ao Agente Estação a existência de Utilizadores na sua APE através de mensagens do tipo *inform*. O Agente Estação envia propostas (*propose*) de preços ao Agente Utilizador, o qual pode aceitar (*accept*) ou recusar (*refuse*). O Agente Interface pede (*request*) ao Agente Utilizador a sua posição, e o segundo responde-lhe (*inform*) com as coordenadas da sua localização no momento. O Agente Interface comunica também com o Agente Estação pedindo-lhe através de mensagens do tipo *request* informações sobre o estado da estação, sendo que o ultimo lhe responde através de uma mensagem do tipo *inform*.

Nas subsecções seguintes serão definidos e explicados detalhadamente os protocolos de comunicação entre cada par.

4.3.1 Protocolo entre Agente Estação e Agente Utilizador

O protocolo de comunicação apresentado na figura 4.3 modela a comunicação entre o Agente Estação e Agente Utilizador. O Agente Estação envia propostas de preços ao Agente Utilizador, o qual pode aceitar ou recusar consoante o seu interesse.

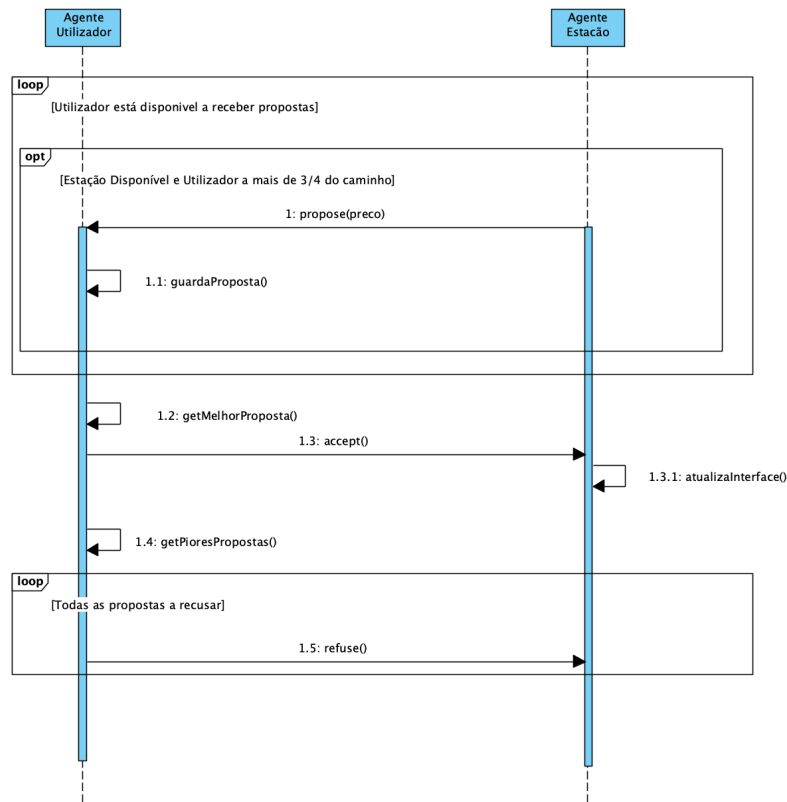


Figura 4.3: Comunicação Agente Estação e Agente Utilizador

4.3.2 Protocolo entre Agente Estação e Agente Monitor

O protocolo de comunicação apresentado na figura 4.4 modela a informação enviada pelo Agente Estação ao Agente Monitor.

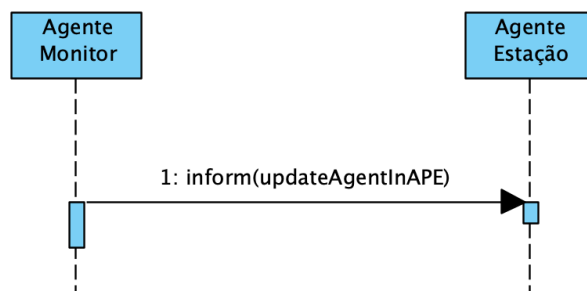


Figura 4.4: Comunicação Agente Estação e Agente Monitor

4.3.3 Protocolo entre Agente Estação e Agente Interface

O protocolo de comunicação apresentado na figura 4.5 estabelece a troca de informação sobre o estado de uma estação entre o Agente Estação e Agente

Interface.

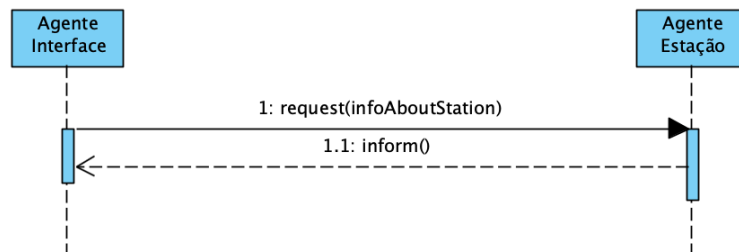


Figura 4.5: Comunicação Agente Estação e Agente Interface

4.3.4 Protocolo entre Agente Utilizador e Agente Monitor

Na figura seguinte é definido o protocolo de comunicação entre o Agente Utilizador e Agente Monitor, no qual o primeiro informa o segundo da sua posição atual e é averiguado por parte do Agente Monitor se essa posição pertence à APE de alguma estação.

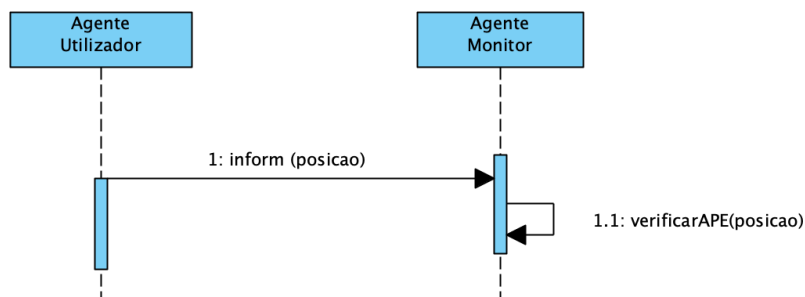


Figura 4.6: Comunicação Agente Utilizador e Agente Monitor

4.3.5 Protocolo entre Agente Utilizador e Agente Interface

O protocolo de comunicação apresentado na figura 4.7 estabelece a troca de informação entre o Agente Utilizador e Agente Interface sobre a posição atual do primeiro.

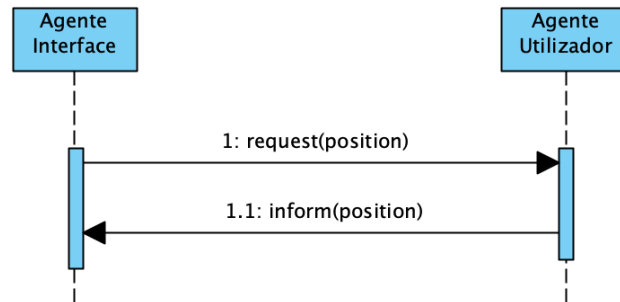


Figura 4.7: Comunicação Agente Utilizador e Agente Interface

4.4 Estado interno do Agente Utilizador

O Agente Utilizador é o agente que apresenta mais dinamismo no sistema, podendo encontrar-se em quatro estados distintos. Na figura 4.8 é exposto num diagrama de máquina de estados o seu comportamento interno.

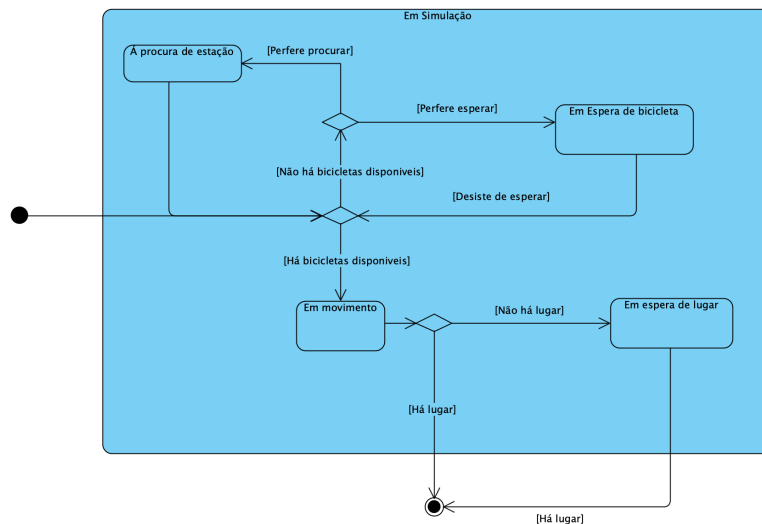


Figura 4.8: Estado interno do Agente Utilizador

O Agente Utilizador nasce numa estação do sistema e caso existam bicicletas disponíveis aluga a bicicleta e inicia a sua deslocação, caso contrário, dependendo da escolha do utilizador, pode esperar até que haja bicicletas disponíveis na estação em que se encontra ou então deslocar-se até outra estação. Após realizar a sua viagem e se encontrar próximo do destino é feita a escolha da estação para depositar a bicicleta, no entanto, escolhendo uma localização sobre-lotada há a possibilidade de ficar de novo em espera até conseguir depositar a sua bicicleta. As ações do agente no sistema terminam com o pagamento e depósito da bicicleta numa estação.

5 | Alterações à Proposta de Resolução Inicial

Após a primeira fase do projeto, foram realizadas algumas alterações à arquitetura primária proposta após perceber melhor como o sistema iria reagir. Para tal, alteraram-se alguns dos diagramas apresentados anteriormente para fins representativos como forma de demonstrar a versão final. Os protocolos serão abordados mais a pormenor ao longo do capítulo dedicado à implementação para uma compreensão mais clara e simples do funcionamento do sistema.

5.1 Arquitetura do Sistema

Arquiteturalmente, o sistema manteve a sua base com a existência dos 4 agentes propostos inicialmente. No entanto, novas classes foram criadas assim como as variáveis que compunham as já existentes.

Na figura 5.1 encontra-se a visão mais abrangente das classes e agentes que se encontram na versão final do sistema. Algumas adições importantes passam pela criação da classe **Personalidade** e **ConfigVars** assim como o acerto das variáveis e relações entre as próprias classes.

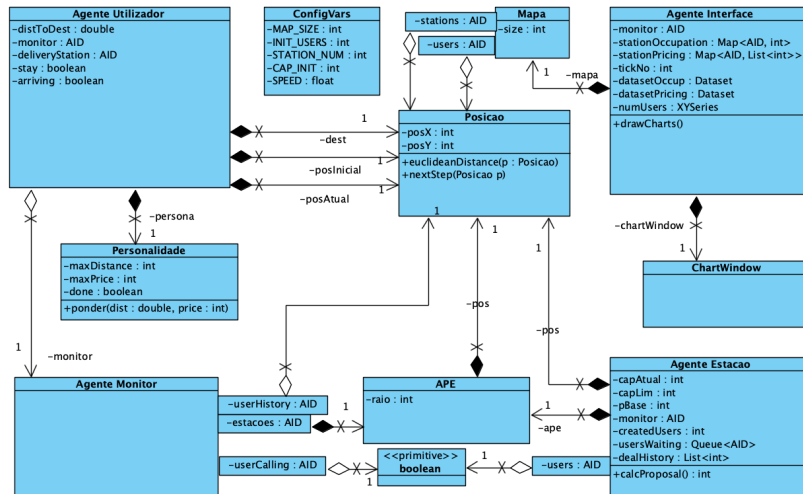


Figura 5.1: Arquitetura do Sistema

5.2 Protocolos de Comunicação

Após considerar mais casos e situações que poderiam acontecer no sistema, chegou-se à conclusão que também seriam alteradas as comunicações existentes entre os agentes assim como a adição de novas. Na figura seguinte encontra-se o resumo final destas.

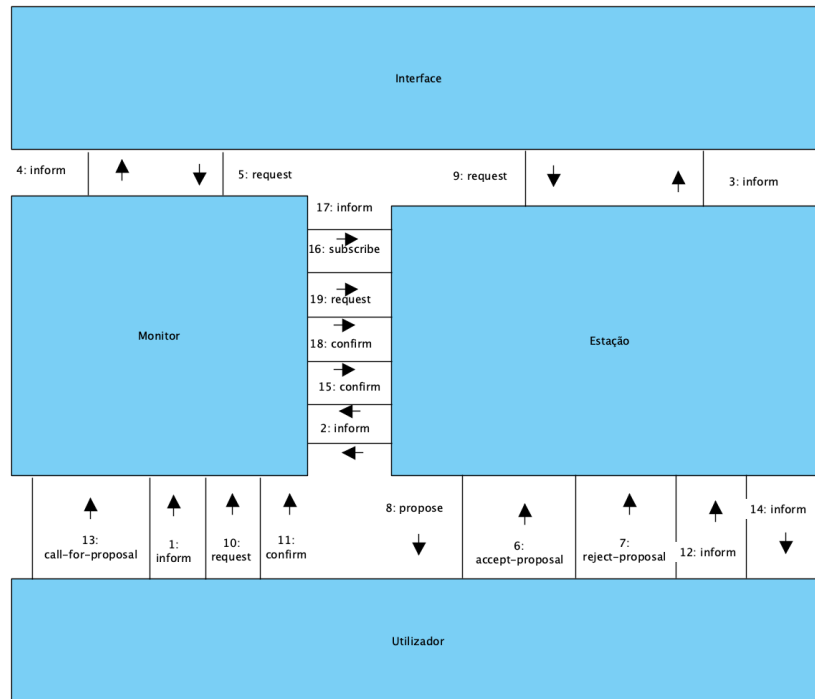


Figura 5.2: Panorama geral da comunicação entre agentes

5.3 Estado interno do Agente Utilizador

Relativamente ao Agente Utilizador e as suas ações, poucas foram as alterações realizadas. No entanto, a sua versão final pode ser observada na figura 5.3.

Nesta o utilizador pode encontrar-se em 4 estados: começa **em movimento** até encontrar uma estação na qual pretende parar entrando em estado de **entrega** da bicicleta. De seguida, este pode ter de **esperar por um lugar** ou terminar logo a sua atividade.

Adicionalmente, este pode encontrar-se em **emergência** quando não quer aceitar propostas das estações ou não consegue receber pedidos destas, chegando ao seu destino sem conseguir estacionar. Assim, após pedir uma recomendação ao Agente Monitor, este aceita a sua sugestão entrando no estado de **entrega**.

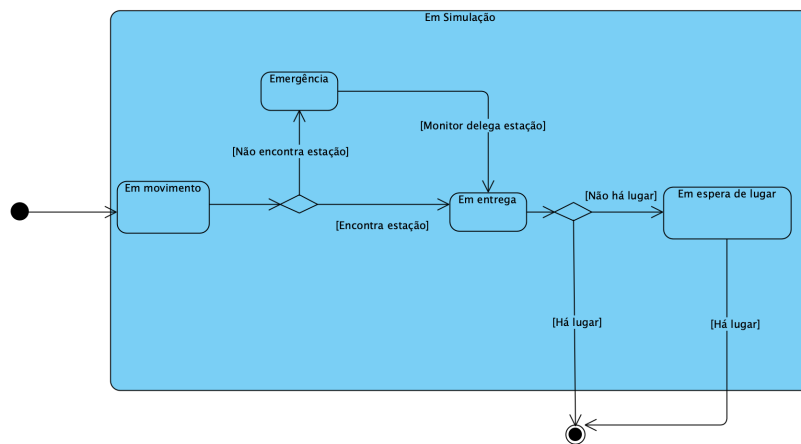


Figura 5.3: Estado interno do Agente Utilizador

6 | Implementação

No capítulo que se segue serão apresentadas todas as decisões tomadas durante o processo de implementação do sistema assim como as suas razões e motivações.

6.1 Inicialização do Sistema e dos Agentes

A base da utilização do *JADE* passa por conseguir criar agentes que interagem entre si num *Container*. Assim, no sistema existe uma classe chamada **MainContainer** onde são inicializadas as características desejadas para a simulação. Para isso criou-se uma classe auxiliar onde se encontram todos os valores necessários para a configuração inicial da simulação chamada **ConfigVars**.

Na classe **MainContainer** é criado o *Container* onde o sistema de agentes se encontra. Primeiramente são gerados os agentes **Estação** com uma posição aleatória dentro de um limite do mapa, seguido do agente **Monitor** uma vez que este assume que as estações têm posições estáticas podendo realizar logo no seu *setup* a procura das estações existentes no *Directory Facilitator*, onde guarda um *HashMap* com os **AID**'s e posições das mesmas.

Seguidamente, são criados os agentes **Utilizador** iniciais procuram imediatamente pelo agente **Monitor** iniciando a sequência de mensagens que permite observar os movimentos do **Utilizador** ao longo do seu percurso.

No final da inicialização, o agente **Interface** é criado. Este necessita que os agentes **Monitor** e **Estação** já estejam a correr pois este agente é responsável por recolher estatísticas que só estão disponíveis depois destes agentes estarem ativos. Por essa razão existe algum tempo de espera entre a inicialização de todos os agentes pelo que é necessário cerca de 4 segundos para o agente **Interface** estar totalmente funcional.

Na classe **ConfigVars** existem os valores que controlam a quantidade inicial de utilizadores no sistema e o tamanho do mapa, bem como o número de estações e capacidade das mesmas. Também se incluiu um multiplicador para controlar a velocidade do sistema ao modificar o período de todos os **TickerBehavior**.

6.2 Realização de uma Viagem

Neste segmento é explicado como é que o sistema e os seus agentes interagem entre si para realizar uma viagem de um utilizador que parte num ponto do mapa

e se desloca até um destino. São mencionadas ainda algumas das comunicações que ocorrem entre os agentes e o porquê destas abordagens.

Os agentes **Utilizador** são criados inicialmente em posições correspondentes a posições de estações isto porque faz sentido que, na vida real, este se tenha de dirigir a uma estação reservar uma bicicleta e por isso partir desse mesmo local. Contudo não é reduzido o número de bicicletas nas mesmas sendo que a criação de mais utilizadores é explicada no capítulo de **Continuidade da Simulação**.

O utilizador é criado com uma **Posicao** aleatória como destino e antes de iniciar o seu percurso estabelece a comunicação com o **Monitor**. A cada unidade de tempo (*Tick*) o utilizador executa um passo onde muda a sua posição atual aumentando uma unidade na direção do destino (em linha reta). Este comunica a sua nova posição ao monitor através de um *INFORM* que a regista e calcula, baseado na ultima posição do utilizador, se este entrou ou saiu de uma **APE** de alguma estação. Caso tenha acontecido este envia aos agentes **Estação** competentes um sinal em forma de *INFORM* de que houve uma mudança. Deste modo diminuímos a quantidade de mensagens a circular no sistema de um típico *Broadcast* para

$$\mathcal{O}(\#Utilizadores \times \#Estacoes \text{ que interagem com } utilizador \text{ num dado passo})$$

Assim que o utilizador atinge os 3/4 do seu caminho passa a enviar mensagens do tipo *CFP* em vez do tipo *INFORM* indicando ao Monitor que este se encontra pronto para receber propostas das estações para depositar a bicicleta. Esta mensagem é passada aos agentes estação que abrangem o utilizador onde se inicia o processo de negociação com o mesmo.

A cada *Tick* cada uma dessas Estações envia uma proposta ao utilizador em forma de *PROPOSE* onde é incluído o preço calculado tendo em consideração sua *Taxa de Ocupação* inflacionando ou descontando caso esta tenha ultrapassado os *Thresholds* estabelecidos.

Cada utilizador tem uma **Personalidade** que analisa propostas e aceita ou recusa baseando-se nos valores máximos de preço e distância à estação que está disposto a aceitar. A personalidade do utilizador é criada com valores aleatórios aquando a criação do mesmo.

Caso o utilizador aceite a proposta envia ao monitor um *CONFIRM* para este redirecionar confirmações a todas as estações onde a sua APE abrange o utilizador, assinalando assim o fim da emissão de propostas novas.

Paralelamente envia uma mensagem *ACCEPT-PROPOSAL* à estação escolhida alterando o seu destino para a posição desta, entrando no estado de entrega da bicicleta e revertendo o tipo de mensagens de rastreamento da posição para o monitor de volta para *INFORM*.

Quando o utilizador chega à estação escolhida, informa a estação da sua chegada, podendo depositar a bicicleta e abandonar o sistema caso haja espaço na estação ou então entrar para uma *Queue* na qual fica à espera de uma mensagem da estação para entregar a sua bicicleta, quando houver espaço.

6.3 Caso de Emergência

Embora o sistema se balance autonomamente existem sempre situações de emergência provocadas pela base inerentemente aleatória deste sistema que afeta a distribuição das estações no mapa e o destino ponderado bem como a personalidade de cada utilizador, instigando em certos casos uma situação onde o utilizador termina o seu percurso desejado sem chegar a um acordo com nenhuma estação.

Nestas situações dizemos que o utilizador entra num estado de **emergência** no qual depende de uma negociação unilateral mediada pelo monitor. Para tal, o utilizador envia-lhe um pedido de ajuda na forma de *REQUEST* para este lhe conseguir sugerir uma estação à qual se pode dirigir para depositar a bicicleta. Este procura a estação mais próxima do utilizador e requisita através de um *REQUEST* à estação escolhida um pedido de proposta forçada para o utilizador. Embora o utilizador não possa recusar esta próxima proposta, esta vem com o preço calculado normalmente sem descontar nem taxar pelo facto de se encontrar numa situação de emergência.

Uma vez aceite a proposta, o processo retoma o seu seguimento normal com as comunicações habituais.

6.4 Continuidade da Simulação

Para garantir que a simulação não termina no fim de todos os utilizadores iniciais entregarem as suas bicicletas e que este sistema simule alguma oferta e procura, utilizadores são criados periodicamente nas estações tendo em consideração o número de bicicletas presentes nas mesmas.

A quantidade e periodicidade de utilizadores criados está dependente da sua capacidade atual bem como do registo de transações guardado pela estação. Esta, apetrechada de um *Random* limitado, determina a quantidade de bicicletas a criar em cada *Tick* ao manipular a distribuição uniforme para criar uma distribuição com maior probabilidade de gerar mais utilizadores quando a estação apresenta maior taxa de preços inflacionados e vice versa. Desta forma obtemos algo remanescente de uma histerese permitindo à estação um controlo mais suave sobre a sua taxa de ocupação ao escoar mais bicicletas quando esta se encontra alta e menos caso contrário.

Adicionalmente, os resultados obtidos tornam-se mais interessantes acabando por simular algo mais semelhante à vida real e com alguma estabilidade.

7 | Interface

O último agente existente no sistema é o agente **Interface** cuja função é demonstrar dados sobre o sistema e a simulação em tempo real. Este encontra-se um pouco à parte pois é algo externo ao sistema que apenas monitoriza o que vai acontecendo e nunca interfere com o decurso dos eventos.

Para tal este interage com o agente **Monitor** e os agentes **Estação** pedindo em forma de *REQUESTs* e de 2 em 2 ticks informações sobre os históricos de preços por transação em cada estação, a ocupação destas e o número de utilizadores no sistema. Os agentes respondem então aos pedidos devolvendo-as em forma de *INFORMs*.

Com estas informações, este exibe gráficos dinâmicos que se adaptam ao longo da simulação como se pode observar pela figura 7.1 os quais foram desenvolvidos com recurso à biblioteca *JFreeChart*.

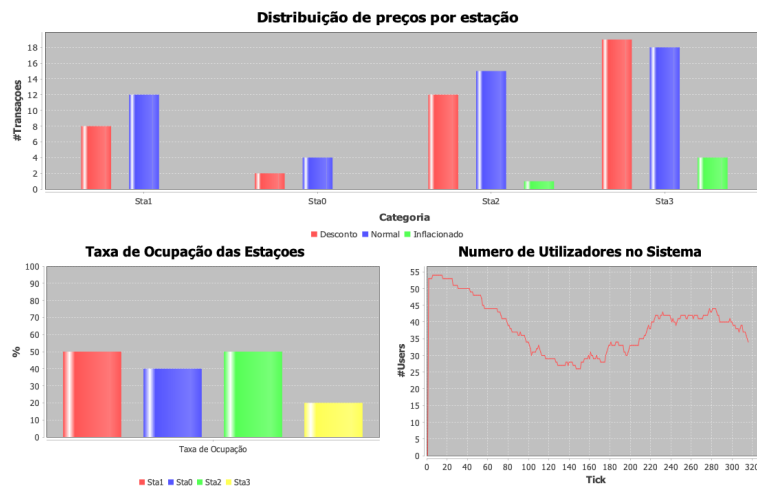


Figura 7.1: Interface final

7.1 Taxa de ocupação das estações

Para representar a taxa de ocupação de cada estação, isto é, o número de bicicletas estacionadas, criou-se um histograma de barras através dos métodos fornecidos pelo *JFreeChart*. O histograma é dinâmico apresentando uma barra

por cada estação existente no sistema. Na figura seguinte apresenta-se, a título de exemplo, um gráfico de um sistema constituído por quatro estações.

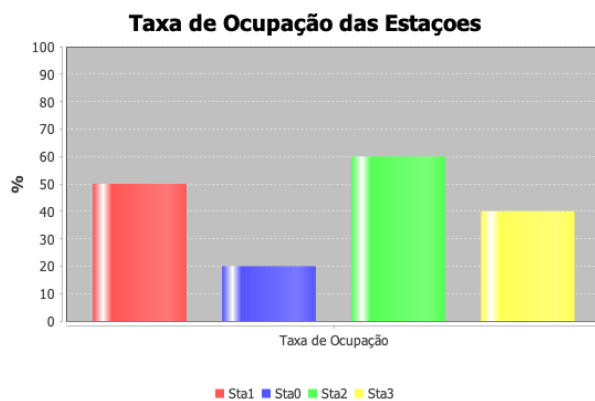


Figura 7.2: Taxa de ocupação das estações

7.2 Número de utilizadores no Sistema

Na figura 7.3 podemos observar a curva que representa o número de utilizadores ativos com o passar do tempo no sistema.

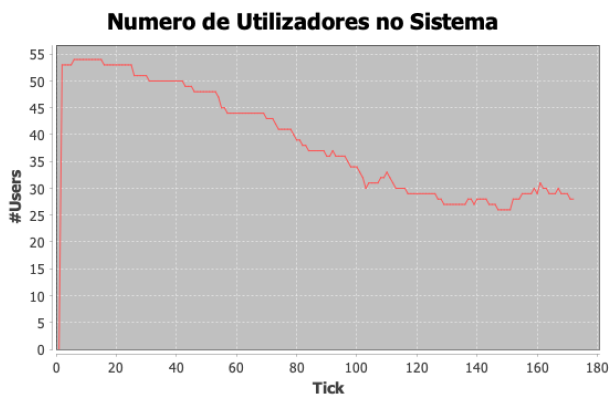


Figura 7.3: Número de utilizadores no Sistema

7.3 Distribuição de preços por estação

O histograma apresentado na figura 7.4 é constituído por três barras para cada estação, estas representam o número de propostas feitas por cada estação a clientes consoante o seu grau de ocupação e o número de transações realizadas de cada tipo.

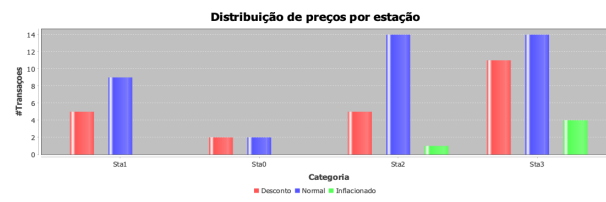


Figura 7.4: Distribuição de preços por estação

8 | Conclusão

Neste relatório foi apresentado todo o planeamento e processo de desenvolvimento da segunda fase da resolução do problema de Sistemas de Partilha de Bicicletas. Este trabalho revelou-se como uma boa forma de por os conhecimentos do grupo à prova (tanto teóricos como práticos).

Partindo de um resumo da fase anterior e seguindo para a posterior menção das alterações realizadas, falou-se sobre a implementação e as motivações que estavam na sua base. A utilidade de um sistema deste tipo no mundo real foi compreendida e ainda se pode refletir na expansão dele e como os agentes interagiriam entre si nesse caso.

Em suma, o grupo considerou o trabalho como algo em que não só demonstrou a sua utilidade como reforçou os conhecimentos sobre a cadeira no geral.

Bibliografia

- [1] Contardo, C., Morency, C., Rousseau, L.-M. (2012). Balancing a dynamic public bike-sharing system (Vol. 4). Cirrelet Montreal.
- [2] Kahn, Robert. (2012). Bicycle Sharing in the USA-State of the Art. *Ite Journal*. 82. 32-36.
- [3] Midgley, P. (2009). The role of smart bike-sharing systems in urban mobility.
- [4] Álvaro Lozano Murciego, De JF, G. Villarrubia, Bajo J. (2018). Multi-Agent System for Demand Prediction and Trip Visualization in Bike Sharing Systems. ResearchGate.
- [5] Odell J., Parunak H.V.D., Bauer B. (2003). Extending UML for Agent-Based Systems. In: Kilov H., Baclawski K. (eds) *Practical Foundations of Business System Specifications*. Springer, Dordrecht.