



Universidade do Minho

Mestrado Integrado em Engenharia Informática

Agentes Inteligentes

Grupo 2

J. Diogo
Monteiro
(A83638)

Benjamim
Oliveira
(PG42815)

Gonçalo Borges
(PG42643)

2 de dezembro de 2020

Conteúdo

1	Introdução	3
2	Agentes	4
2.1	Agente Estação	4
2.2	Agente Utilizador	4
2.3	Agente Interface	4
2.4	Agente Gestor	4
3	Interação dos Agentes	5
3.1	Utilizador - Gestor	5
3.1.1	Utilizador envia Posição Atual	5
3.1.2	Gestor responde com Estação Próxima	5
3.2	Utilizador - Estação	6
3.2.1	Utilizador Requisita o Aluguer	6
3.2.2	Estação Confirma o Pedido	6
3.2.3	Utilizador informa a sua Presença	7
3.2.4	Estação envia Proposta	7
3.2.5	Utilizador responde à Proposta	7
3.3	Estação - Interface	7
4	Arquitetura	8
5	Conclusão	10

1 Introdução

No âmbito da unidade curricular de Agentes Inteligentes, concebemos uma arquitetura distribuída baseada em agentes para um sistema de partilha de bicicletas.

Este sistema de partilha de bicicletas tem-se tornado bastante popular nomeadamente em metrópoles devido à facilidade em alugar e devolver, neste caso, uma bicicleta e devido a ser prático.

Um utilizador pode requisitar uma bicicleta e posteriormente efetuar a sua devolução. A requisição e devolução é feita em estações dedicadas a este fim. Por norma as estações estão distanciadas 100 metros e cada estação possui um número máximo fixo de bicicletas que pode armazenar.

Contudo este sistema sofre de um problema que visaremos em solucionar. Trata-se do reequilíbrio de partilha de bicicletas, ou seja, estações que ficam sobrelotadas e consequentemente estações com falta de bicicletas. Afim de desenhar esta arquitetura, delegamos funções e ações distintas para os agentes apresentados ao longo do relatório.

2 Agentes

Neste ponto serão descritos os agentes no que toca a função e funcionamento dos mesmos. Utilizamos os três agentes sugeridos pela equipa docente (Agentes Estação, Utilizador e Interface), no entanto consideramos que seria benéfico ao desenvolvimento do nosso trabalho o acréscimo de um quarto agente, o agente gestor. Passamos então á descrição e apresentação dos mesmos.

2.1 Agente Estação

Este agente, como o próprio nome indica é referente á estação. Neste projeto representa o que pode ser considerado como um dos pontos cruciais no problema de equilíbrio que pretendemos simular.

Cada estação possui uma APE, ou **Área de Proximidade de uma Estação**. Um Agente Estação pode, no caso de ter falta de bicicletas na sua estação, tentar aliciar utilizadores em viagem a terminarem a mesma e deixarem a sua bicicleta na estação. Para isto o Agente Estação tem a possibilidade de fazer descontos sobre o preço da viagem, ajustando o desconto conforme a necessidade (ou falta dela) de aumentar o número de bicicletas na estação.

2.2 Agente Utilizador

Com o Agente Utilizador simulamos a atuação de uma pessoa que utilize um SPB (Sistema de Partilha de Bicicletas). No caso simulamos o comportamento que o mesmo teria usando um *smartphone*, fornecendo constantemente dados de localização ao Agente Gestor. A partir do momento que este estiver a 3/4 da viagem pode aceitar descontos propostos por Agentes Estação em troca de terminar a viagem prematuramente, deixando a bicicleta antes do destino objetivo.

2.3 Agente Interface

Este agente é utilizado com o intuito de nos permitir visualizar, ao longo do tempo de execução, o estado do sistema em termos de equilíbrio.

Para este ponto poderá vir a ser desenvolvida uma interface de visualização gráfica.

2.4 Agente Gestor

Tendo em conta o alto número de mensagens que seriam trocadas entre os agentes Utilizador e Estação decidimos implementar um agente que servisse como intermediário. Assim, ao invés de existirem comunicações de todos os agentes utilizador com todos os agentes estação temos o Agente Gestor a receber localizações de todos os agente e a informar os mesmo quando estes se encontrarem dentro de alguma APE. Nesse caso o Utilizador irá então comunicar diretamente com a estação.

3 Interação dos Agentes

Nesta secção vamos explicar todas as interações entre os nossos agentes, que serão essenciais ao bom funcionamento do nosso sistema. O objetivo será que o leitor entenda os vários tipos de comunicação que possam existir, tal como a sua necessidade. Para tal, vamos tentar explicar todos de forma sucinta.

3.1 Utilizador - Gestor

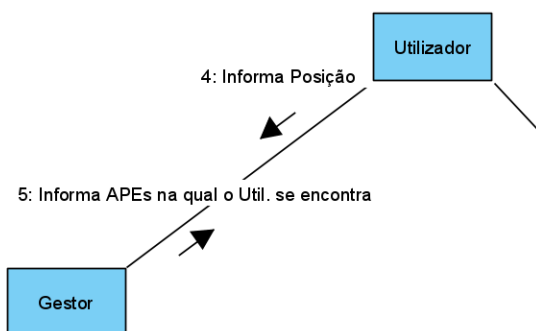


Figura 1: Comunicação entre Utilizador e Gestor

Tal como explicado anteriormente o Gestor é um Agente cuja única função é servir de intermediário entre o utilizador e todas as estações pelas qual ele passar. Para esse efeito, este irá receber continuamente comunicação da nova localização de todos os Agentes Utilizadores que se encontrem em viagem:

3.1.1 Utilizador envia Posição Atual

A partir do momento que o Agente Utilizador aluga uma bicicleta até ao momento em que a entrega, irá, tal como referido anteriormente, comunicar a sua posição ao Agente Gestor. O Gestor, por sua vez, utilizará a nova localização de todos os Utilizadores que com ele comunicaram para calcular se já fizeram 3/4 do seu caminho e se se encontram dentro de uma zona APE.

Performative: INFORMATIVE

3.1.2 Gestor responde com Estação Próxima

Quando o Agente Gestor finalmente deteta que o Utilizador já fez 3/4s do seu caminho e se encontra dentro da zona APE de uma ou mais estações, passa a enviar uma resposta com ao Utilizador com a informação das mesmas Estações. Assim, este pode passar à fase de comunicação direta, em que vai conseguir receber as propostas vindas da Estação.

Performative: INFORMATIVE

3.2 Utilizador - Estação

Todos estes tipos de comunicação entre Utilizador e Estação, ocorrerão em 2 situações diferentes. A primeira, será quando o Utilizador for gerado aleatoriamente numa das estações e terá de proceder a pedir o aluguer de uma bicicleta. O segundo, será quando o Utilizador que prossegue em viagem já fez 3/4 do seu caminho até ao destino e se encontra na zona APE da Estação.

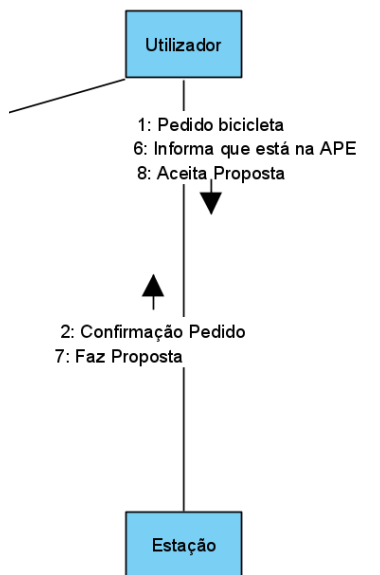


Figura 2: Comunicação entre Utilizador e Estação

3.2.1 Utilizador Requisita o Aluguer

Quando o Agente Utilizador é gerado com a localização de um Agente Estação aleatório terá de proceder a fazer-lhe um pedido do aluguer de uma bicicleta.

Performative: REQUEST

3.2.2 Estação Confirma o Pedido

Após receber o REQUEST de uma bicicleta por parte de um Agente Utilizador, o Agente Estação terá de verificar se tem bicicletas de vago. Caso existam, responderá ao Utilizador que pode proceder ao aluguer, caso não existam terá de adicionar o Utilizador a uma queue e informar que terá de esperar até que haja uma nova vaga.

Performative: ACCEPT // REFUSE

3.2.3 Utilizador informa a sua Presença

Após o Agente Utilizador, que segue viagem, ser informado que se encontra dentro da zona APE de um Agente Estação, irá proceder à fase de comunicação direta. Basicamente, o Utilizador irá informar a estação que se encontra na sua zona, para que assim possa receber proposta de entrega e descontos.

Performative: INFORMATIVE

3.2.4 Estação envia Proposta

O Agente Estação, utilizando a informação sobre todos os Utilizadores existentes na sua APE e sobre o número de bicicletas livres que possui, irá calcular um possível valor de desconto que estará disposta a oferecer ao Utilizador de forma a que este disponibilize/entregue a sua bicicleta. Mediante a necessidade de bicicletas, este valor de desconto poderá aumentar ou diminuir.

Performative: REQUEST

3.2.5 Utilizador responde à Proposta

Uma vez recebido o desconto/proposta da estação o utilizador irá então decidir se aceita ou não o mesmo. Para tal deverá ter em conta a distância que falta percorrer vs o desconto a receber e assim tomar uma decisão, informando a estação da mesma.

Performative: ACCEPT // REFUSE

3.3 Estação - Interface

A Interface é responsável por manter atualizado o número de bicicletas de cada estação. Sempre que uma viagem é iniciada ou terminada, o número de bicicletas é ajustado pela Interface de forma a manter um registo da capacidade de cada Estação.

Performative: INFORMATIVE

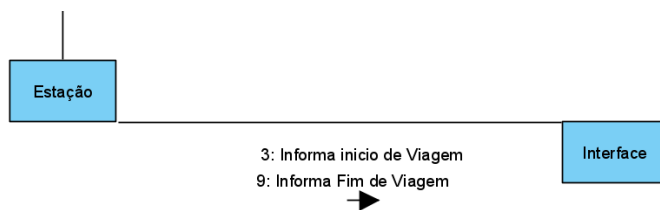


Figura 3: Comunicação entre Estação e Interface

```

graph TD
    subgraph "Agent Utilizador"
        Start(( )) --> P1[Poseição Atribuída]
        P1 --> P2[Pedido Bicicleta]
        P2 --> D1{ }
        D1 -- Não --> P3[Sai da estação]
        D1 -- Sim --> P4[Inicia viagem]
        P4 --> P5[Atualiza posição]
        P5 --> D2{Distância percorrida > 3/4 total?}
        D2 -- Não --> P5
        D2 -- Sim --> P6[Informa Gestor da sua localização]
        P6 --> D3{Encontra-se dentro de alguma APE?}
        D3 -- Não --> P5
        D3 -- Sim --> P7[Informa as estações que se encontra na sua APE]
        P7 --> D4{Utilizador aceita proposta?}
        D4 -- Não --> P8[Faz proposta ao utilizador]
        D4 -- Sim --> P9[Termina viagem]
        P9 --> End(( ))
    end

    subgraph "Agent Estação"
        P2 --> P10[Regista Pedido]
        P10 --> D5{Há bicicleta disponível?}
        D5 -- Não --> P11[Adicionado à que se ainda não estiver na mesma]
        P11 --> D5
        D5 -- Sim --> P12[Informa user]
        P12 --> P4
        P7 --> P13[A estação tem falta de bicicletas?]
        P13 -- Não --> D4
        P13 -- Sim --> P8
    end

    subgraph "Agent Gestor"
        P6 --> P14[Verifica se o utilizador está dentro de APEs, retorna os casos positivos]
        P14 --> D3
    end

    subgraph "Agent Interface"
        P12 --> P15[Atualiza dados relativos à estação]
        P15 --> P16[Atualiza Dados]
        P16 --> P17[Atualiza dados relativos à estação]
    end

```


Caso a Estação em consideração tenha falta de bicicletas, é feita uma proposta ao Utilizador para devolver a bicicleta nessa estação. O Utilizador pode aceitar ou recusar. Em caso de recusa, a posição do utilizador volta a ser atualizada e o processo repete-se até encontrar uma estação disponível a receber a bicicleta. Caso aceite, a viagem é terminada, a Estação atualiza os dados de término da viagem e a Interface atualiza o número de bicicletas presentes nessa estação.

5 Conclusão

Tendo em conta os pontos anteriores, a nossa arquitetura considera 4 agentes. Cada ação, consoante a natureza da mesma, despoletará uma reação dos agentes.

A nossa arquitetura considera o processo End-to-End da requisição de uma bicicleta.

De forma a evitar o desequilíbrio no número de bicicletas presentes em cada estação, é considerada a posição do utilizador e são propostas alterações no local de devolução da bicicleta conforme a lotação das estações.