Relatório sistemas distribuidos

6-1-2019

Nomes aqui

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

Sumário

[Introdução 2](#_Toc534290457)

[Teste teste teste teste Trânsito Professor Tempo demora escrever Boa noite pra outra pessoa Mas até agora Consciência Public static estranho é muito estranho dormir Enter no it back space backspace backspace control come out public static string in the sink in the file open parenthesis it isn't today open brackets be out in brackets pencils 2](#_Toc534290458)

[Funcionamento do programa 3](#_Toc534290459)

[Funcionamento do protocolo 4](#_Toc534290460)

Introdução

Neste trabalho prático pretende-se que implemente um sistema similar ao conhecido Uber. A ideia é ser capaz de chamar uma viatura com condutor para uma deslocação do ponto A a B.

De seguida descreve-se o que se pretende para cada um dos componentes cliente e servidor.

## Cliente (consumidor final)

O cliente consumidor final começa por se registar no serviço. Após este registo passa a ter a possibilidade de efetuar pedidos de viaturas com condutor. A um cliente deverá ser permitido:

* Registar;
* Logar;
* Inserir origem da viagem;
* Inserir destino da viagem;
* Solicitar uma viatura com condutor para uma viagem específica;
* Atribuir uma pontuação (1 a 5) ao condutor para uma viagem específica;
* Visualizar o seu histórico de viagens e respetiva pontuação atribuída;
* Sair.

## Cliente (condutor)

O cliente condutor começa por se registar no serviço. Após este registo passa a ter a possibilidade de aceitar pedidos de viaturas com condutor caso esteja disponível para o efeito. A um cliente deverá ser permitido:

* Registar;
* Logar;
* Alterar estado (disponível / indisponível)
* Receber pedidos de viatura com condutor para uma viagem específica e aceitar ou rejeitar os mesmos;
* Visualizar o seu histórico de viagens e respetiva pontuação recebida;
* Sair.

## Servidor

O servidor deve estar preparado para receber ligações de vários clientes em simultâneo. Deverá ainda ser capaz de guardar o registo de todas as informações relevantes (clientes, viagens, pontuações, etc).

Todas os pedidos de viagens do servidor para os clientes condutor devem ser efetuados via *multicast*.

## Detalhes

Deve ser dada especial atenção aos seguintes detalhes que deverão estar cuidadosamente implementados na solução:

* Existência de um protocolo de comunicação ao nível da aplicação claro e documentado.
* Suporte para múltiplas *threads* e controlo sempre que exista acesso a recursos partilhados. O trabalho deve incluir um relatório explicando sucintamente a implementação e justificando quaisquer opções tomadas.

Funcionamento do programa

Funcionamento do protocolo