

Sistemas Distribuídos — Trabalho Prático

Reserva de voos

Engenharia Informática
Universidade do Minho

2021/2022

Informações gerais

- Cada grupo deve ser constituído por 4 elementos.
- Deve ser entregue o código fonte e um relatório de até 6 páginas (A4, 11pt) no formato PDF.
- O trabalho deve ser entregue até às 23:59 do dia 12 de janeiro de 2022 no *eLearning*.
- A apresentação do trabalho será em dia e hora a agendar posteriormente por cada grupo.
- Cada grupo deve organizar a sua apresentação de forma a que todos os elementos participem espontaneamente.

Resumo

Neste projeto pede-se a implementação de uma plataforma de reserva de voos sob a forma de um par cliente-servidor em Java utilizando *sockets* e *threads*. A essência do serviço é permitir aos utilizadores reservar viagens constituídas por vários voos, sendo informados pelo servidor quando a reserva foi feita com sucesso. Uma viagem reservada pode ser cancelada antes do encerramento do dia correspondente.

Para simplificar, não são tidos em conta os horários dos voos, assumindo que é sempre possível fazer escalas (se existe um voo $A \rightarrow B$ e outro $B \rightarrow C$ num dado dia, é possível um passageiro fazer $A \rightarrow B \rightarrow C$ se houver disponibilidade de lugares em ambos os voos). Assume-se que os mesmos voos se repetem todos os dias.

Funcionalidade Básica

Este serviço deverá suportar a seguinte funcionalidade básica:

1. Autenticação e registo de utilizador, dado o seu nome e palavra-passe. Sempre que um utilizador desejar interagir com o serviço deverá estabelecer uma conexão e ser autenticado pelo servidor.
2. Autenticação de um utilizador especial de administração (admin).
3. Inserção de informação sobre voos (origem, destino, capacidade) pelo administrador.
4. Encerramento de um dia por parte do administrador, impedindo novas reservas para esse dia e cancelamento de reservas desse dia.

5. Reserva de uma viagem indicando o percurso completo com todas as escalas (por exemplo *Porto* → *London* → *Tokyo*, ou seja, de *Porto* a *Tokyo* com uma escala em *London*) e um intervalo de datas possíveis, deixando ao serviço a escolha de uma data em que a viagem seja possível. O servidor deverá responder com o código de reserva.
6. Cancelamento da reserva de uma viagem, usando o código de reserva, a pedido do utilizador a que pertence.
7. Obtenção da lista de todas os voos existentes (lista de pares *origem*→*destino*), a pedido do utilizador.

Funcionalidades Adicionais

A resposta será valorizada pelas seguintes funcionalidades adicionais:

1. Obtenção de uma lista com todos os percursos possíveis para viajar entre uma origem e um destino, limitados a duas escalas (três voos). Minimize a quantidade de dados transferidos.
2. Possibilidade de um cliente continuar a executar outras operações enquanto se espera pela conclusão duma reserva, incluindo novas operações de reserva.

Cliente

Deverá ser disponibilizado um cliente que ofereça uma interface com o utilizador que permita suportar a funcionalidade descrita acima. Este cliente deverá ser escrito em Java usando *threads* e *sockets* TCP.

Servidor

O servidor deverá ser escrito também em Java, usando *threads* e *sockets* TCP, mantendo em memória a informação relevante para suportar as funcionalidades acima descritas, receber conexões e input dos clientes, bem como fazer chegar a estes a informação pretendida. O protocolo entre cliente e servidor deverá ser num formato binário, através de código desenvolvido no trabalho, podendo recorrer apenas a `Data[Input|Output]Stream`. Para o serviço não ficar vulnerável a clientes lentos, não deverá ter *threads* do servidor a escrever em mais do que um socket, devendo as escritas serem feitas por threads associadas a esse socket.