



INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Departamento de Engenharia de Electrónica e Telecomunicações e de Computadores

Licenciatura de Engenharia Informática e de Computadores

Introdução a Sistemas de Informação

Semestre de Inverno 2022-2023

Transporte de passageiros

Trabalho prático
(Fase 3)

Afonso Remédios, Matilde Pato e Nuno Datia

Planeamento

As datas importantes a recordar são:

- Lançamento do enunciado: **16 de Outubro 2022**
- Entrega intermédia (Fase 1): **14 de Outubro de 2022**
- Entrega intermédia (Fase 2): **2 de Dezembro de 2022**
- Entrega intermédia (Fase 3): **6 de Janeiro de 2023**

Cada entrega intermédia deve apresentar o relatório e código (se houver) referentes **exclusivamente** a essa fase. O relatório deve seguir um dos *templates* fornecidos, obrigatoriamente, sob pena de penalização. Este deve ser conciso e apresentar a justificação de todas as decisões tomadas (ver Critérios de Avaliação). A capa do relatório deve indicar a composição do grupo, a unidade curricular e a fase do trabalho que relata. Caso tenha adendas e/ou correcções a fazer a modelos já entregues, deve indicá-las de forma explícita no relatório seguinte.

O pdf (e, o zip) gerado deve seguir o nome da seguinte forma: 'TPNISI-2223SI-Grupo-**AP-NN**.ext' (AP representa o acrónimo do docente, NN representa um dígito, e 'ext' a extensão do ficheiro), e.g.: TP1ISI-2223SI-Grupo-mp-01.pdf.

5 de Dezembro de 2022, Afonso Remédios, Matilde Pato e Nuno Datia

Objectivos de aprendizagem

No final da **terceira fase do trabalho**, os alunos devem ser capazes de:

- Estabelecer uma ligação ao SGBD pretendido, correctamente parametrizada, utilizando JDBC;
- Utilizar correctamente comandos parametrizados para executar operações em JDBC (*prepared statement*);
- Utilizar correctamente transações para garantir atomicidade nas operações, utilizando JDBC;
- Gerir correctamente o tempo de vida das ligações JDBC;
- Garantir a libertação de recursos, quando estes não estejam a ser utilizados;
- Utilizar correctamente o tipo `ResultSet`;
- Implementar **todas as restrições de integridade** aplicacionais que não foram possíveis em PostgreSQL.

Enunciado do trabalho (3ª fase)

Considerando o texto com os requisitos do sistema, apresentados na primeira fase do trabalho, e o modelo de dados implementado na segunda fase, deve garantir, no programa desenvolvido em Java que:

1. o proprietário tem no máximo 20 veículos;
2. o proprietário nunca é um condutor e vice-versa;
3. os atributos `ncconducao` e `matricula` respeitam as restrições de integridade de coluna (**Senão a implementou anteriormente. Só foram consideradas como aceites as expressões regulares**);

4. todos os veículos registados com mais de 5 anos, à data actual, deverão passar para uma nova tabela `VEICULO_OLD`, de forma automática, com os mesmos atributos de `VEICULO` e duas colunas adicionais com informação sobre (1) o número total de viagens efectuadas, e (2) os quilómetros totais percorridos. Para este último valor, poderá recorrer à classe que se encontra disponível em Geodatasource (<https://www.geodatasource.com/developers/java>);
5. deverá ser permitido alterar a designação em `TIPOVEICULO` sempre que surja uma nova. Esta mudança, poderá obrigar a fazer alterações ao nível do tipo.

Os alunos terão agora a oportunidade de utilizar uma API JDBC para, através de uma aplicação Java, acederem e manipularem os dados existentes no modelo físico criado na fase anterior. As alterações/actualizações/consultas irão ser feitas a nível da aplicação e não directamente na BD.

Sempre que se justificar devem ser usados os mecanismos transacionais necessários para garantir a atomicidade das operações. Também devem utilizar mecanismos que evitem ataques de “*SQL injection*”, bem como que evitem problemas de formatações, e.g. campos de datas.

Resultados pretendidos

Tendo em conta os objectivos de aprendizagem, deverão ser produzidos os seguintes resultados:

1. Uma aplicação Java (executado independente do ambiente de desenvolvimento) que permita realizar as seguintes operações **(não deve fazer qualquer alteração à BD já construída, com excepção da adição da nova tabela referida acima)**:
 - (a) Opção para inserir um novo condutor;
 - (b) Opção para colocar um veículo fora de serviço a partir da matrícula, i.e. este veículo não poderá realizar mais viagens. Assim sendo, passará para a nova tabela `VEICULO_OLD`;
 - (c) Opção para calcular as horas totais, os quilómetros e o custo total de um veículo. O utilizador deverá escolher o veículo dentro da lista dos existentes. Sugiro que apresente a lista de veículos disponíveis;
 - (d) Opções que implementem as alíneas 2.c, 2.d, 3.b e 3.c (da Fase 2), garantindo que todos os parâmetros variáveis são alteráveis na interface com o utilizador. E.g. em 2.c “(...) com mais viagens no ano de 2021” pode ser substituído por outro ano à escolha do utilizador; e, em 3.b “Dado o número de identificação fiscal (...)”, o utilizador poderá inserir o nome do proprietário em vez do número.

Todas as instruções devem vir indicadas (e, explicadas) no “relatório” que dá suporte a este trabalho. Se considerar necessário, pode a título ilustrativo, mostrar tabela(s) de resultados.

Data limite para entrega: 6 de Janeiro de 2023 até às 23:59.

A entrega deve incluir um “relatório” e o código, enviados de forma electrónica através do Moodle. O relatório **é entregue** em formato PDF. No final deverá colocar o código e o pdf num único um arquivo ZIP.

Notas:

1. Deve garantir a correcta implementação de todas as funcionalidades, incluindo o acesso a dados;
2. Deve criar em Java um modelo de dados que mapeie as relações utilizadas para objectos em memória;
3. A lógica de interface com o utilizador deve estar em classes separadas da lógica de acesso a dados;
4. Deve ser possível aferir cada um dos objectivos de aprendizagem no material que entregar.

Todo o código entregue tem de ser executado independente do ambiente de desenvolvimento, em linha de comandos. Os alunos têm de fornecer as instruções de execução, assumindo como único pré-requisito a existência da máquina virtual Java 1.8.