



INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Área Departamental de Engenharia de Electrónica e Telecomunicações e de Computadores

LEIC

Sistemas de Informação 2

Semestre de Inverno 2021-2022

Trabalho prático

v1.00

Afonso Remédios e Nuno Datia

Planeamento

As datas importantes a recordar são:

- Lançamento do enunciado: **26 de Outubro de 2020**
- Entrega intermédia (Fase 1): **04 de Dezembro de 2021** (≈ 6 semanas)
- Entrega intermédia (Fase 2): **22 de Janeiro de 2021** (≈ 6 semanas)

Cada entrega intermédia deve apresentar o relatório e código (se houver) referentes **exclusivamente** a essa fase. O relatório deve seguir um dos modelos fornecidos, obrigatoriamente, sob pena de penalização na nota. Este deve ser conciso e apresentar a justificação de todas as decisões tomadas. 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 zip (e o pdf com o relatório) gerado deve seguir o padrão: 'TPSI2-2122SI-GrupoNN-TTTFaseN.ext' (N representa um dígito, TTT indica a turma, e 'ext' a extensão do ficheiro). Por exemplo: TPSI2-2122SI-Grupo14-51DFase1.zip.

Objectivos de aprendizagem

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

- Desenvolver um modelo de dados adequado aos requisitos do sistema, normalizado até à 3NF;
- Conceber e implementar uma solução baseada em bases de dados dinâmicas, adequada aos requisitos;
- Utilizar corretamente controlo transacional;
- Utilizar corretamente níveis de isolamento;
- Utilizar corretamente vistas, justificando o seu uso na solução;
- Utilizar corretamente procedimentos armazenados, justificando o seu uso na solução;
- Utilizar corretamente gatilhos, justificando o seu uso na solução;
- Utilizar corretamente funções, justificando o seu uso na solução;
- Conseguir testar, usando T-SQL, todas as funcionalidades indicadas nos requisitos;
- Conseguir disponibilizar a solução, a partir de uma base de dados vazia, com recurso o código T-SQL;
- Escrever um relatório técnico sobre as decisões tomadas e o trabalho desenvolvido.

Enunciado do trabalho (Documento de requisitos do sistema)

A **Maintain4ver** é uma empresa de manutenção que pretende implementar um sistema de informação para a gestão de manutenção de activos físicos.

A empresa gere um conjunto de **activos**. Estes são caracterizados por um **identificador** (único), o **nome**, a **data de aquisição** (em dd-mm-aaaa), o **estado**, a **marca**, o **modelo** e a **localização**. Todos os campos são obrigatórios, com excepção da marca e modelo. O estado só pode tomar 2 valores: “0” ou “1”, i.e., desactivado ou operacional. Um activo tem um **tipo** caracterizado por um **identificador** e uma **descrição**. Um activo pode ser constituído por outros activos, e.g. uma piscina constituída por uma bomba de água, onde ambos são activos, constituindo uma hierarquia de activos. O tipo do activo de topo da hierarquia tem de ser igual ao(s) tipo(s) do(s) activo(s) “filho(s)”. Pretende-se manter o registo histórico do valor comercial do activo, em euros, registando-se a data (no formato dd-mm-aaaa) em que a alteração ocorreu.

A empresa tem vários **funcionários**, sendo estes descritos pelo seu **nome completo**, a **data de nascimento**, endereço (inclui **morada**, **código postal** e **localidade**), **profissão**, **telefone** (fixo e/ou móvel), **e-mail** e o **número de identificação** (CC e/ou NIF).

Na manutenção dos activos físicos existem **intervenções**. Uma intervenção pode ser **periódica**, sendo registada a periodicidade (em meses), e **não periódica**. Para cada intervenção é registado um **número único**, uma **descrição**, um **estado**, o **valor monetário** (em euros), a **data de início e de fim** (em dd-mm-aaaa). A descrição pode tomar como valores “avaria” ou “rutura”, “inspecção”, sendo estes meramente indictivos. Sobre estado, os valores possíveis são “por atribuir”, “em análise”, “em execução” ou “concluído”. A data de intervenção deve ser superior à data de aquisição do activo intervencionado.

Cada intervenção é realizada por uma **equipa de manutenção** constituída, no mínimo, por 2 elementos. Um deles será o supervisor. A equipa é caracterizada por um **código de equipa**, uma **localização**, e um **número de elementos que a constitui**. A equipa completa tem pelo menos 2 elementos durante a intervenção ao activo. Se for necessário, podem-se adicionar/alterar elementos à equipa, durante a intervenção. Cada equipa só consegue realizar uma intervenção de cada vez, podendo ter até 3 intervenções atribuídas. Se não for possível atribuir uma equipa, o estado da intervenção deve ser “por atribuir”. Note-se que a equipe pode mudar ao longo da resolução da intervenção, devendo ficar registadas todas as equipas envolvidas, garantindo que é possível ordená-las de forma cronológica.

Os **elementos de uma equipa** são caracterizados pelo seu **nome completo**, a **data de nascimento**, **endereço** (inclui **morada**, **código postal** e **localidade**), **profissão**, **telefone** (fixo e/ou móvel), **e-mail** e o **número de identificação** (CC e/ou NIF). Existe uma pessoa da empresa que gere o activo e que não pode participar na equipa que faz a intervenção desse activo. Cada elemento da equipa tem re-

gistado as suas competências. A **competência** de um elemento tem uma **descrição** e um **identificador** (único). Uma pessoa só pode pertencer a uma equipa. A **atribuição** de uma intervenção a uma equipa só é possível se a descrição for compatível com as competências da equipa.

Resultados pretendidos

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

1. Uma ilustração do modelo de dados (conceptual e relacional), incluindo todas as restrições de integridade e regras de negócio, indicando a solução de implementação de cada uma;
2. O código T-SQL que permite, sem erros, em 5 scripts:
 - (a) Criar o modelo físico (1 *script* autónomo);
 - (b) Remover o modelo físico (1 *script* autónomo);
 - (c) Preencher a base de dados com registos de teste (1 *script* autónomo);
 - (d) Mecanismo que permite Inserir, remover e atualizar informação de uma pessoa. Não podem ser usados insert, update e delete directos. (1 *script* autónomo que contém as alíneas 2d a 2k);
 - (e) Obter o código de uma equipa livre, dada uma descrição de intervenção, capaz de resolver o problema. Em caso de haver várias equipas deve escolher-se a que teve uma intervenção atribuída à mais tempo;
 - (f) Criar o procedimento `p_criaInter` que permite criar uma intervenção;
 - (g) Implementar o mecanismo que permite criar uma equipa;
 - (h) Actualizar (adicionar ou remover) os elementos de uma equipa e associar as respectivas competências;
 - (i) Criar uma função para produzir a listagem (código, descrição) das intervenções de um determinado ano;
 - (j) Actualizar o estado de uma intervenção;
 - (k) Criar uma vista que mostre o resumo das intervenções (atributos de intervenção e activo), que possibilite a alteração do estado de uma ou mais intervenções;
 - (l) Testar as funcionalidades de 2d a 2k (1 *script* autónomo), com indicação dos testes que falham.

Garanta que, para os utilizadores da base de dados, todas as funcionalidades de 2d a 2k produzem os resultados esperados, sendo sempre garantidas as restrições de integridade e regras de negócio. Deve reutilizar o código entre alíneas. Deve indicar no relatório a forma de testar as funcionalidades. Além disso, deve ser claro quando um teste falha.

Data limite para entrega: 04 de Dezembro de 2021 até às 23:59.

A entrega deve incluir um relatório, para além do código T-SQL, enviados de forma electrónica através do Moodle. O relatório é **entregue** em formato PDF (obrigatório).

Nota: Deve ser possível aferir cada um dos objectivos de aprendizagem no material que entregar. Trabalhos que apresentem erros sintáticos e/ ou que não criem os objectos pretendidos são considerados como não conformes, não validando os objectivos de aprendizagem

Objectivos de aprendizagem

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

- Desenvolver uma aplicação em C#, que use diferentes implementações de acesso a dados;
- Utilizar corretamente processamento transaccional, concretizado através de mecanismos disponíveis na plataforma .NET;
- Organizar o código de acesso a dados para permitir usar processamento transaccional, iniciado nesse código ou existente no contexto de chamada;
- Utilizar corretamente ADO.NET em modo “conectado”, para acessos a dados;
- Utilizar corretamente (ADO.NET) Entity Framework para acessos a dados;
- Garantir a correta libertação de ligações e recursos, quando estes não estejam a ser utilizados;
- Garantir a correta implementação das restrições de integridade e/ou lógica de negócio;
- Organizar o código de acesso a dados usando padrões de desenho, nomeadamente, Data Mapper, Lazy load, Data Transfer Object e Virtual Proxy;
- Escrever um relatório técnico sobre o trabalho desenvolvido.

Enunciado do trabalho (2ª fase)

Nesta fase do trabalho pretende-se que os alunos criem uma aplicação que use diferentes *frameworks* de acesso a dados. A aplicação deve ter como meta a reutilização de código, fácil manutenção e eficiência, e ser independente do modo de acesso a dados.

Resultados pretendidos

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

1. Criação de uma aplicação .NET que, usando uma implementação de acesso a dados desenvolvida usando objetos “conectados” do ADO.NET, permita:
 - (a) Aceder às funcionalidades 2e a 2i, descritas na fase 1 deste trabalho;
 - (b) Implementar a funcionalidade 2f, descrita na fase 1 deste trabalho, sem usar qualquer procedimento armazenado;
 - (c) Implementar a atribuição de uma intervenção usando conjuntamente as funcionalidades 2e, 2f e 2j;
2. Alteração (ou configuração) da aplicação desenvolvida na alínea 1, de forma a que esta use uma implementação de acesso a dados desenvolvida usando Entity Framework (EF). Esta alteração deve ter o mínimo impacto possível;
3. Apresentar testes comparativos de desempenho das tecnologias EF e ADO, na implementação da 1c.
4. Usando EF com optimistic locking, trocar as competências de dois elementos de uma equipe. Apresente uma mensagem de erro adequada em caso de alteração concorrente conflituante que inviabilize a operação. No relatório deve estar descrita a forma como as situações de erro foram criadas para teste desta alínea.

Data limite para entrega: 22 de Janeiro de 2021 até às 23:59.

A entrega deve incluir um relatório (em formato PDF), o código SQL de criação, remoção e preenchimento da base de dados, os projetos Visual Studio e o código C# , submetidos de forma eletrónica via moodle. O código deve vir configurado para funcionar com o SGBD disponibilizado pelo ISEL, sendo indicado como se altera a `ConnectionString` para outro servidor. Todos os *scripts* de criação dos objectos no SGBD devem ser submetidos como parte do trabalho.

Nota: Deve ser possível aferir cada um dos objetivos de aprendizagem no material que entregar. Trabalhos que apresentem erros sintáticos e/ ou que não criem os objectos pretendidos são considerados como não conformes, não validando os objectivos de aprendizagem.