



Departamento de Engenharia de Electrónica e Telecomunicações e
de Computadores
Licenciatura em Engenharia Informática e de Computadores

Trabalho prático

(Fases 1 e 2)

Sistemas de Informação II

Semestre de Verão 2015/2016

Versão 1.01

Docente: Nuno Datia

Planeamento

As datas importantes a recordar são:

- Lançamento do enunciado: **08 de Março de 2016**
- Entrega intermédia (Fase 1): **02 de Maio de 2016**
- Entrega intermédia (Fase 2): **13 de Julho de 2016**

Cada entrega intermédia tem de incluir o relatório e código (se houver) referentes a essa fase. O relatório deve ser conciso e apresentar a justificação de todas as decisões tomadas. Tem de indicar a composição do grupo, a unidade curricular e a fase do trabalho que relata. Caso tenha adendas e/ou correcções a fazer a entregas anteriores, deve enumerar cada uma de forma explícita no relatório. Entregas sem relatório não serão consideradas.

Cada fase do trabalho terá de ser mostrada ao docente a funcionar, em data e local a combinar.

Objectivos de aprendizagem

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

- ☐ Desenvolver um modelo de dados adequado aos requisitos, normalizado até à 3NF;
- ☐ Conceber e implementar uma solução baseada em bases de dados dinâmicas, adequada aos requisitos;
- ☐ Utilizar correctamente vistas, justificando o seu uso na solução;
- ☐ Utilizar correctamente procedimentos armazenados, justificando o seu uso na solução;
- ☐ Utilizar correctamente gatilhos, justificando o seu uso na solução;
- ☐ Utilizar correctamente funções, justificando o seu uso na solução;
- ☐ Utilizar correctamente níveis de isolamento;
- ☐ Utilizar correctamente controlo transaccional.

Enunciado do trabalho (documento de requisitos do sistema)

Uma empresa denominada *ISto É Lindo* (ISEL) pretende desenvolver um sistema de venda online, que junta o conceito de leilões com vendas directas. Nesse sistema, um utilizador é caracterizado por um email (único), por um nome, por uma morada e por uma palavra passe. Por questões de segurança, a palavra passe não será guardada em claro, ficando apenas registado o resultado da função de *hash* MD5. Quando um utilizador registado pretende vender um artigo, terá de indicar um conjunto de informação obrigatória: a sua descrição, a data de início da venda, a data de fim da venda, a localização do artigo, a sua condição e o tipo de venda. As condições admissíveis, de momento, são (i) "Novo", (ii) "Usado", (iii) "Como novo" e (iv) "Velharia vintage". O tipo de venda tem um identificador (único), uma data/tempo e pode ser uma venda directa ou um leilão. Neste último tipo, tem de ser registado valor mínimo de venda, e o valor da licitação mínima, sempre entre 1 euro e 10% do valor de venda. Se o valor mínimo não for atingido o artigo não será vendido. No caso das vendas directas, apenas é necessário registar o preço de venda. Todos os artigos podem ser licitados. Se for uma venda directa, a licitação só é válida se o valor for igual ao valor de venda. Se for um leilão, a licitação tem de ser superior à última licitação, acrescida da licitação mínima. Sempre que é feita uma licitação, tem de ficar

registada a data/hora em que foi feita, que é única para o artigo. Para garantir um correcto ordenamento das licitações, todas as datas têm de ser inseridas na base de dados com precisão de milissegundos. Não são permitidas inserções de novas licitações sempre que o tempo definido para a venda tenha terminado. Depois das licitações terem terminado, o vencedor (quem fez a licitação mais alta) tem até 2 dias para formalizar a compra. Findo esse tempo, a “venda” não é possível. Uma compra é caracterizada por código (único), uma data de compra (diferente da data de licitação se for um leilão), um número de cartão de crédito e uma localização (destino). Deve estar registado no sistema o preço dos portes a pagar entre localizações, sendo estas definidas de acordo com o ISO 3166-1. O preço dos portes será acrescido ao valor de compra.

Em qualquer momento deve ser possível simular o custo total de compra, usando um valor de licitação válido e uma localização de entrega. É ainda possível um utilizador retirar uma licitação que efectuou, desde que dentro do período de venda. No entanto, essa informação não pode ser removida do sistema. É também necessário manter o registo histórico de todas as alterações de morada dos utilizadores, garantindo que se sabe a data/tempo em que foram efectuadas. O sistema não deve permitir alterações do email dos utilizadores.

Garanta que o modelo de dados exposto é consistente com as entidades descritas, escondendo sempre detalhes de implementação.

Resultados pretendidos

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

1. O modelo de dados (conceptual e relacional), incluindo todas as restrições de integridade;
2. O código T-SQL que permite:
 - (a) Criar o modelo físico;
 - (b) Inserir, remover e actualizar informação de um artigo;
 - (c) Inserir, remover e actualizar informação de um utilizador;
 - (d) Inserir, remover e actualizar informação de um local;
 - (e) Inserir uma licitação;
 - (f) Retirar uma licitação;
 - (g) Concluir a compra de um leilão;
 - (h) Realizar a compra de um artigo de venda directa;
 - (i) Determinar o valor da licitação de um artigo;
 - (j) Obter as n últimas licitações;
 - (k) Obter os portes, dadas duas localizações;
 - (l) Listar os leilões que não foram concluídos;

(m) Verificar a password de um utilizador;

(n) Testar todos os requisitos anteriores.

Data limite para entrega: 02 de Maio de 2016 até às 23:59.

A entrega deve incluir um relatório (entregue em PDF) e o código T-SQL, enviados de forma electrónica.

Nota: Deve ser possível aferir cada um dos objectivos de aprendizagem no material que entregar.

Objectivos de aprendizagem

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

- ☐ Utilizar correctamente processamento transaccional, através de mecanismos disponíveis na plataforma .NET, para garantir as propriedades ACID nas operações;
- ☐ Utilizar correctamente (ADO.NET) Entity Framework para acessos a dados;
- ☐ Utilizar correctamente ADO.NET em modo “conectado”;
- ☐ Garantir a correcta libertação de recursos, quando estes não estejam a ser utilizados;
- ☐ Garantir a correcta implementação das restrições de integridade e/ou lógica de negócio;
- ☐ Utilizar XML e XML Schema para produzir documentos válidos e bem formados a partir de dados guardados num SGBD.

Enunciado do trabalho (2ª fase)

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

1. Crie uma única aplicação .NET que permita:
 - (a) Aceder às funcionalidades **(e)** a **(m)**, descritas na alínea 2 da fase 1 deste trabalho;
 - (b) Exportar a informação do conjunto de todas licitações de um leilão que tenha terminado, obedecendo ao XML Schema em anexo;
 - (c) Usar uma implementação de acesso a dados desenvolvida usando **Entity Framework** (EF)
 - (d) Usar uma implementação de acesso a dados desenvolvida usando objectos **“conectados”** do ADO.NET.
2. Compare, usando métricas objectivas, a tecnologia EF e ADO.NET quanto à facilidade de programação e desempenho.
3. Indicar em que situações a EF e o ADO.NET apresentam vantagens/desvantagens em termos de garantia da consistência dos dados.

Data limite para entrega: 13 de Julho de 2016 até às 23:59.

A entrega deve incluir um relatório (em PDF), os projectos *Visual Studio*, o código C# e o código SQL, enviados de forma electrónica.

Nota: Deve ser possível aferir cada um dos objectivos de aprendizagem no material que entregar.

XML Schema a usar

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
3     xmlns:xs="http://www.w3.org/2001/XMLSchema">
4     <xs:element name="auction" type="auctionType"/>
5     <xs:complexType name="bidsType">
6         <xs:sequence>
7             <xs:element type="bidType" name="bid" maxOccurs="unbounded" minOccurs="0"/>
8         </xs:sequence>
9         <xs:attribute type="xs:string" name="num"/>
10    </xs:complexType>
11    <xs:complexType name="infoType">
12        <xs:sequence>
13            <xs:element type="xs:string" name="minimumBid"/>
14            <xs:element type="xs:string" name="reservationPrice"/>
15            <xs:element type="xs:string" name="initialDate"/>
16        </xs:sequence>
17    </xs:complexType>
18    <xs:complexType name="auctionType">
19        <xs:sequence>
20            <xs:element type="infoType" name="info"/>
21            <xs:element type="bidsType" name="bids"/>
22        </xs:sequence>
23        <xs:attribute type="xs:ID" name="id" use="required"/>
24    </xs:complexType>
25    <xs:complexType name="bidType">
26        <xs:simpleContent>
27            <xs:extension base="xs:string">
28                <xs:attribute type="xs:string" name="userid" use="required"/>
29                <xs:attribute type="xs:string" name="datetime" use="required"/>
30            </xs:extension>
31        </xs:simpleContent>
32    </xs:complexType>
33 </xs:schema>
```