



# Integração de Dados

2018/2019

## Trabalho Prático de Avaliação

**Nota prévia:** O enunciado é propositadamente vago, genérico e incompleto em alguns pontos. O que se pretende é que os alunos avaliem as várias opções existentes e escolham a que considerarem mais apropriada para cada uma das situações com que se deparem. Todas as escolhas devem ser referidas e devidamente justificadas no relatório a entregar.

### 1. Objetivos

Com este trabalho pretende-se criar um programa em Java (com interface gráfica) composto por vários *Wrappers* que obtenham dados de fontes heterogéneas, distribuídas, autónomas e que possibilitem ao utilizador a visualização dos dados de forma integrada.

O utilizador terá ainda a possibilidade de fazer pesquisas, acrescentar dados que respeitem os esquemas adotados e gerar ficheiros com informação selecionada.

Para a realização deste trabalho deve usar a Linguagem Java, Expressões regulares e a API JDOM2 para a manipulação de XML estudada nas aulas práticas.

### 2. Resultados da Aprendizagem

Com este trabalho prático, pretende-se que se adquiram as seguintes competências:

- Saber analisar uma situação típica de Integração de Dados e apresentar propostas válidas para um modelo de integração funcional, eficaz e correto;
- Capacidade de criação e manipulação de ficheiros XML;
- Utilização de expressões regulares;
- Capacidade de realização de pesquisa de informação em ficheiros XML, usando XPath ou XQuery;
- Capacidade de efetuar transformações de ficheiros XML;
- Capacidade de efetuar validação de ficheiros XML usando DTD e/ou XSD;
- Capacidade de criar aplicações Java.

### 3. Descrição do Trabalho

O trabalho consiste na criação de uma aplicação integradora que apresente uma vista unificada de informação relativa aos contratos públicos realizados e publicitados pelos Municípios do distrito de Coimbra. As fontes de dados a usar são:

- Municípios.txt – Ficheiro com o nome dos municípios e os respetivos números de contribuinte (NIF);
- <https://www.anmp.pt> (**Municípios -> Municípios de A a Z -> Centro**) ou <https://www.anmp.pt/anmp/pro/mun1/mun101w2.php?dis=06> ) – Informações detalhadas dos Municípios
- <http://www.base.gov.pt> (**entidades -> + -> Lista de contratos onde esta entidade foi adjudicante**) – Informações dos contratos públicos

Podem ser usadas outras fontes de dados, mas a sua utilização deve ser devidamente justificada. Se a estrutura das fontes de dados usadas não for equivalente às sugeridas acima, poderão existir penalizações na nota final.

- As três fontes de dados são heterogéneas, autónomas e distribuídas, contendo informação relevante sobre as diversas Câmaras Municipais e os respetivos contratos.

- O objetivo do trabalho prático consiste em efetuar a **integração de dados** provenientes destas fontes de dados e construir dois ficheiros XML que agreguem a informação de forma organizada e coerente.
- Ficheiro **camaras.xml** deve conter, para cada Câmara Municipal do Distrito de Coimbra (19), uma lista de informações detalhadas. Esta informação deve ser obtida do website <http://www.anmp.pt/index.php/municipios/municipios-de-a-a-v>. Deve recolher pelo menos a seguinte informação:
  - Identificador único (gerado pelo aluno), Nome do Município, Nome do Presidente, E-mail, Site Institucional, Telefone, Número de Freguesias, Área, Número Habitantes, Brasão e outras informações que considere relevantes.
- Ficheiro **contratos.xml** deve conter informação sobre os contratos realizados por cada câmara. Esta informação, à exceção do identificador, deve ser obtida através do website <http://www.base.gov.pt> e deve contemplar a seguinte informação mínima.

- Identificador único (gerado pelo aluno), NIF, Objeto do Contrato, Preço Contratual, Data da Publicação, Nome do Município, Adjudicatário, e outras informações que considere relevantes.

**Nota:** Para obter o NIF do Município pode utilizar o ficheiro municipios.txt ou aceder ao website [www.nif.pt](http://www.nif.pt)

O esquema a adotar na vista unificada deve ser decidido pelos alunos e validado usando o XSD e o DTD apropriado.

Depois de realizado o processo de integração dos dados, o utilizador poderá fazer pesquisas sobre a vista unificada.

## 4. Tarefas a Realizar

De seguida, encontram-se as tarefas principais a desenvolver neste trabalho prático. As descrições são genéricas e os exemplos apresentados servem apenas para uma melhor compreensão do que é pretendido. Os alunos devem ser criativos e apresentar uma solução integradora, completa e funcional que permita efetuar uma grande diversidade de pesquisas.

### 4.1. Definir o esquema global

A primeira parte do trabalho consiste em procurar todos os atributos acima descritos para uma lista com informações de Câmara Municipais do distrito de Coimbra e os respetivos contratos. O ficheiro **municipios.txt** contém uma lista com os municípios da região de Coimbra e o respetivo NIF. A informação recolhida deve ser organizada nos dois ficheiros XML acima descritos. Os alunos devem validar os modelos de dados escolhidos para os ficheiros XML usando XSD. Devem também construir os DTD correspondentes. Nota que os caracteres especiais do ficheiro foram introduzidos propositadamente.

## 4.2. Implementar Wrappers

Implementar os *Wrappers* que permitam obter a informação relevante de cada fonte de dados. Estes *Wrappers* devem ser implementados usando expressões regulares. **No relatório deve ser descrito detalhadamente** cada um dos *Wrappers*, indicando que informação é retirada por cada um deles da fonte de dados em que cada um opera.

Para cada atributo a encontrar, deve(m) ser selecionada(s) a(s) fonte(s) de dado(s) relevante(s). No caso de encontrar inconsistências ou conflitos, os alunos terão de propor uma solução.

Para saber como implementar os *Wrappers*, deve analisar a fonte das páginas HTML onde vai procurar a informação.

Use a função *httpRequest* dada nas aulas práticas para aceder às páginas e gravá-las em disco.

O número e a estrutura dos *Wrappers* depende da forma e da quantidade de informação que se quer encontrar e deve ser analisada pelos estudantes.

## 4.3. Gerar ficheiros XML de acordo com 4.1

A informação obtida pelos *Wrappers* deve ser usada para a criação dos ficheiros XML de acordo com os modelos apresentados em 4.1.

Os ficheiros devem ser validados usando os XSD/DTD escolhidos.

Esta tarefa deve ser feita usando a API JDOM2 dado nas aulas práticas.

## 4.4. Definir Pesquisas

Permitir ao utilizador efetuar diferentes pesquisas:

1) Sobre os ficheiros XML criados;

2) Sobre as fontes de dados originais

- **Exemplo: obter detalhes de uma data introduzida pelo utilizador**

- Para esse município os detalhes já se encontram no ficheiro **contratos.xml**? → Mostrar dados recolhidos.

- Para esse **Município** não existem dados no ficheiro? → Procurar nas fontes de dados usando os *Wrappers*, mostrar ao utilizador os dados recolhidos e adicionar a nova informação nos ficheiros XML, respeitando os modelos decididos em 4.1. Se não existirem concursos ou não forem encontrados, informar o utilizador do insucesso da pesquisa.

- Os passos 1) e 2) devem ser aplicados a diferentes tipos de pesquisas:

- Exemplos de pesquisas:

- Procurar contratos por data específica;
- Procurar contratos por autor da publicação;
- Procurar contratos por adjudicatário;
- Procurar qual o contrato de maior valor de uma Câmara Municipal específica;
- Procurar qual o contrato de maior valor de todas as Câmaras.
- Introduzir uma câmara e obter todos os dados da mesma;
- Top 5 das Câmaras que gastaram mais com contratos;
- Procurar Câmara por nome do presidente;
- Efetuar pesquisas que combinem dois ou mais atributos;
- Mais pesquisas serão valorizadas.

As pesquisas devem ser definidas pelos alunos, devendo ser variadas e versáteis, com combinação de um diferente número de atributos.

A escolha dos atributos de pesquisa devem ser escolhidos pelo utilizador usando Interface Java.

Os resultados devem ser apresentados de forma atrativa e organizada usando Interface Java.

As pesquisas devem ser implementadas usando XPath. Use o API JAXEN dados nas aulas práticas.

## 4.5. Editar e eliminar informação

A interface deve possibilitar ao utilizador editar um determinado atributo dos ficheiros XML. Por exemplo, alterar/apagar/acrescentar contratos de um valor específico, alterar o nome do presidente, número de habitantes, acrescentar/eliminar *emails*, etc. O utilizador também pode eliminar os **contratos** de um dia específico do ficheiro **contratos.xml** ou uma Câmara do ficheiro **camaras.xml**

A informação dos dois ficheiros deve estar coerente e relacionada, usando um campo à escolha do aluno. A remoção de uma Câmara Municipal do ficheiro **camaras.xml** deve ter como consequência a remoção de todas as informações do ficheiro **contratos.xml**. Após as alterações, o modelo deve ser validado, para garantir a sua integridade.

## 4.6 Gerar ficheiros de Output

O programa deve possibilitar ao utilizador gerar ficheiros de resultados. Estes ficheiros devem ser transformações dos ficheiros XML da vista global.

Duas transformações **obrigatórias**:

- Gerar um ficheiro HTML contendo o brasão de todos as Câmaras Municipais listadas no ficheiro XML;
- Gerar um novo ficheiro XML que junte atributos dos dois ficheiros (**contratos.xml** e **camaras.xml**)

Transformações adicionais serão valorizadas. Devem implementar as transformações usando XSLT

## 5. Normas para Realização do Trabalho

O trabalho deverá ser realizado **individualmente ou em grupos de até três alunos**.

A indicação dos elementos do grupo de trabalho é obrigatória e deve ser feita por *e-mail* ([filipe.sa@isec.pt](mailto:filipe.sa@isec.pt)) até dia **21 de novembro de 2019**.

O trabalho final deve ser entregue até **15 de janeiro de 2019** às 23h55 GMT. Os trabalhos serão sujeitos a **defesa obrigatória** na aula prática de **16 de janeiro de 2019**. Neste dia, deverá ser apresentado o programa a funcionar, entrega de um relatório em papel e todo o código desenvolvido.

## 6. Critérios de Avaliação

O trabalho vale **6 valores** na nota final da Unidade Curricular.

Será avaliado segundo os seguintes critérios:

- Qualidade e correção na implementação das tarefas solicitadas;
- Funcionalidade do programa;
- Originalidade e diversificação dos conteúdos abordados, nomeadamente as funcionalidades extras;
- Justificação das opções tomadas;

- Qualidade do relatório entregue;
- Qualidade da defesa.