Uma imagem com texto

Descrição gerada automaticamente Instituto Superior de Engenharia de Coimbra

Departamento de Engenharia Informática e Sistemas

Licenciatura em Engenharia Informática

Unidade Curricular: Integração de Dados

2020/2021

Relatório de Trabalho Prático

*Integração de dados com XML*

­

Isabel Ramos Castro 2018013160

Índice

[Estrutura do projeto 3](#_Toc73198495)

[Interface Gráfico 4](#_Toc73198496)

[HttpRequestFunctions 4](#_Toc73198497)

[XMLJDomFunctions 5](#_Toc73198498)

[XPathFunctions 6](#_Toc73198499)

[JDOMFunctions\_Validar 6](#_Toc73198500)

[JDOMFunctions\_XSLT 8](#_Toc73198501)

[SaxonFunctions\_XQuery 8](#_Toc73198502)

# Estrutura do projeto

O projeto criado no Netbeans divide-se em vários ficheiros:

− HttpRequestFunctions

− XMLJDomFunctions

− XPathFunctions

− JDOMFunctions\_Validar

− JDOMFunctions\_XSLT

− SaxonFunctions\_XQuery

− Jogador

− Wrappers

− XML\_functions

− Interface

− main

Interface Gráfica  
A figura seguinte ilustra a interface gráfica implementado para o projeto. A  
área de texto maior é o espaço onde serão apresentados a maioria dos resultados das diversas opções do menu.

HttpRequestFunctions  
Este ficheiro foi fornecido pela professora nas aulas práticas, tem 2 funções que extraem dados da web:  
public static int httpRequest1(String link, String pesquisa, String outFile)  
public static void httpRequest2(String link, String pesquisa, String outFile)

Ambas recebem um link de uma página web e uma string (pesquisa) com o que se pretende pesquisar, estas funções juntam o link e a pesquisa formando o URL a que vão aceder, e de seguida guardam a informação da página obtida num ficheiro html, cujo o nome é passado por argumento para a função (variável outFile).  
A única diferença entre estas duas funções é a codificação usada para os caracteres, pois pode variar de site para site, neste projeto utilizei apenas a função httpRequest1.  
A função httpRequest1 fornecida não devolvia qualquer valor, no entanto no decorrer do desenvolvimento do programa, pareceu pertinente colocar esta função a  
devolver um valor inteiro, é importante frisar que este inteiro só é apanhado quando a função é chamada na função “encontra\_linha” do ficheiro “WrappersW”, esta implementação surgiu da necessidade de informar o utilizador se o filme procurado foi encontrado nas duas páginas ou apenas em uma delas, mais detalhes na secção “Filme”

# XMLJDomFunctions

Ficheiro fornecido nas aulas práticas tem 3 funções:

public static Document lerDocumentoXML(String caminhoFicheiro)

public static void escreverDocumentoParaFicheiro(Document doc, String caminhoFicheiro)

public static String escreverDocumentoString(Document doc)

Tal como o nome indica a função lerDocumentoXML lê o documento XML, para que posteriormente se possa pesquisar, alterar ou transformar o seu conteúdo, o ficheiro XML tem um nome que é passado como argumento da função através da variável caminhoFicheiro, a função devolve um documento com o conteúdo do ficheiro XML lido.

A função escreverDocumentoParaFicheiro cria em disco um documento, caso ele não exista, caso exista altera-o, nesse documento fica o conteúdo de um documento XML que está previamente em memória. Na variável doc está guardado o conteúdo XML, na variável caminhoFicheiro o nome do ficheiro que vai ser criado/alterado.

Com a função escreverDocumentoString é possível aceder a um documento com o conteúdo de um ficheiro XML e guardar todo o seu conteúdo numa variável do tipo String, a função recebe como parâmetro um documento (doc), previamente guardado na memória.

Para que estas funções funcionem corretamente é necessário adicionar ao projeto a API JDOM, neste caso adicionei a biblioteca jdom-2.0.6.jar.

XPathFunctions  
Este ficheiro foi fornecido nas aulas práticas e tem apenas duas funções:  
static XdmValue executaXpath(String xp, String xmlFile)  
static String listaResultado(XdmValue lista)

A função executaXpath é responsável pela pesquisa XPath num ficheiro XML, a  
variável xp é uma String com a pesquisa XPath e a variável xmlFile tem o nome do  
ficheiro XML onde a pesquisa vai ser efetuada, esta função devolve a lista de resultados que encontrou.  
A função listaResultados recebe o resultado da função anterior (executaXpath) e  
devolve uma String com o resultado da pesquisa efetuada anteriormente.

# JDOMFunctions\_Validar

Este ficheiro foi fornecido nas aulas práticas e é composto por 4 funções:

public static Document validarDTD(String caminhoFicheiro)

public static Document validarXSD(String caminhoFicheiro)

public static int validarDocumentoDTD(String xmlFile, String DTDFile)

public static int validarDocumentoXSD(String xmlFile, String XSDFile)

A função validarDTD abre o ficheiro XML indicado na variável caminhoFicheiro e introduz o cabeçalho necessário para a validação posterior, a função validarXSD faz exatamente o mesmo, desta vez, introduz o cabeçalho correspondente à validação por XSD, caso o ficheiro seja válido por XSD a função validarXSD devolve um documento com conteúdo do ficheiro XML validado, caso o ficheiro seja válido por DTD a função validarDTD devolve um documento com conteúdo do ficheiro XML validado.

A função validarDocumentoDTD, recebe como parâmetros o nome do ficheiro DTD com a estrutura do ficheiro XML pretendido (DTDFile) e o nome do ficheiro XML que se pretende verificar (xmlFile) esta função devolve -1 caso o ficheiro seja inválido e 1 caso seja válido.

A função validarDocumentoXSD recebe como parâmetros o nome do ficheiro XSD com a estrutura do ficheiro XML pretendido (XSDFile) e o nome do ficheiro XML que se pretende verificar (xmlFile), caso o ficheiro seja inválido a função devolve -1 caso seja válido devolve 1.

Para o correto funcionamento destas funções é necessário o recurso à API JDOM e SAXON, neste projeto adicionei duas bibliotecas da API SAXON, saxon9.jar e saxon-s9api.jar, para além da já mencionada anteriormente jdom-2.0.6.jar.

# JDOMFunctions\_XSLT

Este ficheiro foi fornecido no decorrer das aulas práticas e inclui duas funções:

public static Document transformaDocumento(Document XMLdoc, String xmlFile, String xslFile)

public static void transformaDocumento2(String xmlFile,String xslFile, String sOutFile)

A função transformaDocumento recebe um documento com o conteúdo XML a transformar, o nome do ficheiro XML a transformar e o nome do ficheiro XSL com o template de transformação e devolve um documento com o conteúdo XML/HTML resultante.

A função transformaDocumento2 recebe o nome do ficheiro XML a transformar, o nome do ficheiro XSL com o template de transformação e o nome do ficheiro TXT onde ficará guardado o conteúdo resultante da operação.

SaxonFunctions\_XQuery  
Este ficheiro fornecido nas aulas práticas tem 3 funções:  
public static void xQueryToText(String outputFile, String queryFile)  
public static void xQueryToHtml(String outputFile, String queryFile)  
public static void xQueryToXml(String outputFile, String queryFile)

A função xQueryToText recebe como parâmetros o nome do ficheiro TXT onde  
será guardado o resultado outputFile e o nome do ficheiro onde se encontra a query queryFile.  
A função xQueryToHtml recebe como parâmetros o nome do ficheiro HTML onde  
será guardado o resultado outputFile e o nome do ficheiro onde se encontra a query  
queryFile.  
A função xQueryToXml recebe como parâmetros o nome do ficheiro XML onde  
será guardado o resultado outputFile e o nome do ficheiro onde se encontra a query  
queryFile.