

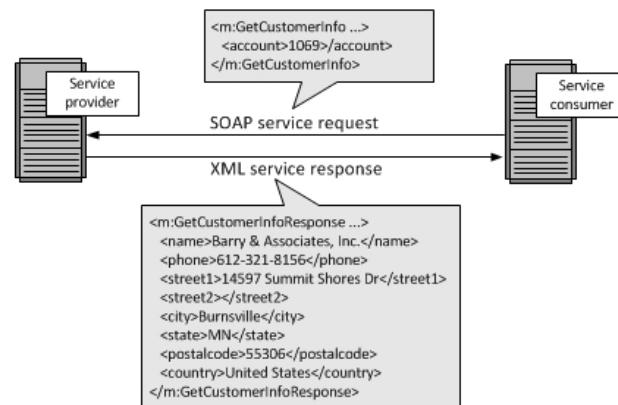
Unidade Curricular:**Integração de Sistemas de Informação****Tema da Ficha Prática:**

Este tutorial/ficha pretende apresentar a interligação entre as seguintes componentes: XML e webservices (implementados via SOAP e REST) e com implementação em Java e PHP.

Objectivos:

Pretende-se com esta ficha prática que os alunos:

- 1) implementem ou usem a implementação feita na ficha prática anterior um parsing distribuído cliente/servidor de ficheiros XML (de tamanho reduzido e de grande volume);
- 2) Criem uma tabela numa base de dados (ex. em MySQL), ler um ficheiro XML e importar os dados para essa tabela e executar queries através de web services SOAP e REST (ver figura seguinte). A implementação neste tutorial será efetuada usando as linguagens de programação Java e PHP.

SOAP.**Representation State Transfer (REST)**

Fonte: https://www.service-architecture.com/articles/web-services/web_services_explained.html

Conteúdo

Bibliografia.....	3
1. Resumo da implementação do tutorial do parser de XML usando Java	6
1.1 – Parser DOM	6
1.1.1 Com o ficheiro Simple.xml	6
1.1.2 Com o ficheiro medsamp2014.xml	7
1.2 – Parser SAX.....	7
1.2.1 Com o ficheiro note.xml.....	7
1.2.2 Com o ficheiro books.xml	8
1.2.3 Com o ficheiro medsamp2013.xml	8
1.3 – Parser JDOM	9
1.3.1 Com o ficheiro note.xml.....	9
1.3.2 Com o ficheiro books.xml	9
1.3.3 Com o ficheiro medsamp2013.xml	10
1.4 – Parser JAXP.....	10
1.4.1 Com o ficheiro simple.xml	10
1.4.2 Com o ficheiro books.xml	11
1.4.3 Com o ficheiro medsamp2014.xml	11
1.4.4 Com o ficheiro po.xml	12
2. Implementação de Webservices SOAP e REST integrando ficheiros XML em Base de Dados usando a abordagem Cliente/Servidor	13
2.1 Enquadramento.....	13
2.2 Implementação em JAVA	14
2.2.1 Importação do ficheiro books.xml, via SOAP e Parser: JAXB.....	28
2.2.2 Importação do ficheiro books.xml, via REST e Parser: JAXB	31
2.2.3 Importação do ficheiro simple.xml, via SOAP e Parser: JAXB.....	33
2.2.4 Importação do ficheiro simple.xml, via REST e Parser: JAXB	34
2.2.5 Importação do ficheiro alunos.xml, via SOAP e importação na Base de Dados	35
2.2.6 Importação do ficheiro alunos.xml, via REST e importação na Base de Dados	37
2.3 Implementação em PHP	39
2.3.1 Execução de webservice via REST para leitura de dados de uma base de dados	41
2.3.2 Importação do ficheiro books.xml via REST usando o parser DOM e/ou SAX	44
2.3.3 Execução de webservice via SOAP para leitura de dados de uma base de dados	48
2.3.4 Implementação de webservice via SOAP para leitura de ficheiro books.xml usando o parser DOM	56
2.3.5 Implementação de webservice via SOAP para leitura de ficheiro books.xml usando o parser SAX	57

Bibliografia

Para apoio a esta ficha os alunos devem consultar os apontamentos teóricos e práticos da disciplina bem como de outros recursos online, como por exemplo:

a) Validação de Ficheiros XML:

http://www.w3schools.com/xml/xml_validator.asp
<http://www.xmlvalidation.com/>
<http://validator.w3.org/>

b) Informação sobre a interpretação de ficheiros XML usando programação:

<http://www.developerfusion.com/code/2064/a-simple-way-to-read-an-xml-file-in-java/>
<http://docs.oracle.com/javase/tutorial/jaxp/stax/example.html>
http://docs.oracle.com/cd/B28359_01/appdev.111/b28394/adx_j_parser.htm
<http://www.mkyong.com/tutorials/java-xml-tutorials/>
https://www.tutorialspoint.com/java_xml/

c) Parsers XML:

- Usando a linguagem de programação JAVA:

1. Document Object Model (DOM):

<http://docs.oracle.com/javase/tutorial/jaxp/dom/readingXML.html>

2. Simple API for XML (SAX):

<http://docs.oracle.com/javase/1.5.0/docs/api/org/xml/sax/package-summary.html>
<http://www.saxproject.org/quickstart.html>

3. JDOM XML Parser

<http://www.jdom.org/>

<http://www.jdom.org/docs/oracle/dom-part1.pdf>

4. Java API for XML Processing (JAXP):

<http://www.oracle.com/technetwork/java/intro-140052.html>

5. Java Architecture for XML Binding (JAXB)

<http://jaxb.java.net/tutorial/index.html>

- Usando a linguagem de Programação PHP:

<https://www.php.net/xml>

<https://www.php.net/manual/en/refs.webservice.php>

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

Nota:

- Servirão como guia e apoios deste trabalho, a ficha prática (Java com XML) e links disponibilizados no moodle da unidade curricular.
- Para simplificar o processo, pretende-se que o servidor receba o ficheiro XML, faça o parsing do ficheiro na estrutura de armazenamento implementada (DOM, SAX, JDOM, JAXP e JAXB) e que faça o print do ficheiro do lado do servidor.

Em termos gerais, pretendeu-se através da ficha prática anterior, executar os parsers XML, usando a linguagem de programação JAVA:

```

1.1 - DOM.....  

  1.1.1 Simple.xml.....  

  1.1.2 Book.xml.....  

  1.1.3 medsamp2014.xml.....  

1.2 - SAX.....  

  1.2.1 - Books.xml.....  

  1.2.2 Note.xml.....  

  1.2.3 Ned samp2013.xml.....  

1.3 - JDOM.....  

  1.3.1 Books.xml.....  

  1.3.2 Note.xml.....  

  1.3.3 Ned samp2013.xml.....  

1.4 - JAXP.....  

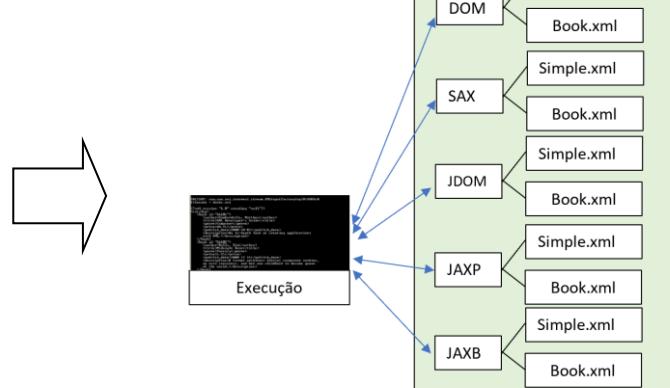
  1.4.1 Books.xml.....  

  1.4.2 Simple.xml.....  

  1.4.3 med samp2014.xml.....  

1.5 JAXB.....  

  1.5.1 Po.xml.....
```



Os ficheiros XML a submeter ao servidor via cliente, são os seguintes:

1. Um ficheiro simples obtido em: http://www.w3schools.com/xml/xml_examples.asp

w3schools.com

HTML CSS JAVASCRIPT SQL PYTHON PHP BOOTST

XML Tutorial

XML HOME
XML Introduction
XML How to use
XML Tree
XML Syntax
XML Elements
XML Attributes
XML Namespaces
XML Display
XML HttpRequest
XML Parser
XML DOM
XML XPath
XML XSLT
XML XQuery
XML XLink
XML Validator
XML DTD
XML Schema
XML Server

XML Examples

XML Examples

Try it Yourself - Examples

Viewing XML Files

View a simple XML file (note.xml)
View the same XML file with an error
View an XML CD catalog
View an XML plant catalog
View an XML food menu

Examples explained

2. O ficheiro Books.xml: [http://msdn.microsoft.com/en-us/library/ms762271\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/ms762271(v=vs.85).aspx)

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

The screenshot shows a Microsoft documentation page for the MSXML SDK. The URL is [docs.microsoft.com/en-us/previous-versions/windows/desktop/ms762271\(v=vs.85\)?redirectedfrom=MSDN](https://docs.microsoft.com/en-us/previous-versions/windows/desktop/ms762271(v=vs.85)?redirectedfrom=MSDN). The page title is "Sample XML File (books.xml)". It includes a sidebar with navigation links for XML, XSLT, XML Schemas, DOM, Helper APIs, SAX2, SOM, XML Digital Signatures, XML Error Messages, New XML Error Messages for Windows 8, XPS Documents, Extensible Storage Engine, Input Feedback Configuration, Microsoft Management Console 2.0, Host Guardian Service WMI Provider, and Help API. The main content area displays the XML code for the books.xml file:

```

<?xml version="1.0"?>
<catalog>
  <book id="bk101">
    <author>Gambardella, Matthew</author>
    <title>XML Developer's Guide</title>
    <genre>Computer</genre>
    <price>44.95</price>
    <publish_date>2000-10-01</publish_date>
    <description>A in-depth look at creating applications with XML.</description>
  </book>
  <book id="bk102">
    <author>Ralls, Kim</author>
    <title>Midnight Rain</title>
    <genre>Fantasy</genre>
    <price>5.95</price>
    <publish_date>2000-12-16</publish_date>
    <description>A former architect battles corporate zombies, an evil sorceress, and her own childhood to become queen of the world.</description>
  </book>
  <book id="bk103">
    <author>Corets, Eva</author>
  </book>

```

3. Um ficheiro de grande volume obtido em: <http://aiweb.cs.washington.edu/research/projects/xmltk/xmldata/>

The screenshot shows the XMLData Repository website at aiweb.cs.washington.edu/research/projects/xmltk/xmldata/. The page title is "XMLData Repository". It features a "Goal" section stating that the repository collects publicly available datasets in XML form and provides statistics on the datasets for research experiments. It mentions that DTDs for the datasets are included whenever possible, and dataset statistics were computed using the XML Toolkit. Below this is a "Detailed View of Datasets" section and a "Repository Contents" section. The "Repository Contents" table lists the following datasets:

Name	Description	Uncompressed Size	Preview	Date
Protein Sequence Database	Integrated collection of functionally annotated protein sequences.	683 MB		Nov 9 2001
SwissProt	SWISS-PROT is a curated protein sequence database which strives to provide a high level of annotations (such as the description of the function of a protein, its domains structure, post-translational modifications, variants, etc.), a minimal level of redundancy and high level of integration with other databases.	109 MB		1998
Auction Data	Auction data converted to XML from web sources.	23 KB		2001
DBLP Computer Science Bibliography	The DBLP server provides bibliographic information on major computer science journals and proceedings. DBLP stands for Digital Bibliography Library Project.	127 MB		Oct 2002
University Courses	Course data derived from university websites.	277 KB		1999
Nasa	Datasets converted from legacy flat-file format into XML and made available to the public.	23 MB		2001
SIGMOD Record	Index of articles from SIGMOD Record	467 KB		2001
TPC-H Relational Database Benchmark	TPC-H Benchmark, 10 MB version, in XML form. Converted to XML by Zack Ives.	603 KB		2002
Treebank (partially encrypted)	English sentences, tagged with parts of speech. The text nodes have been encrypted because they are copywritten text from the Wall Street Journal. Nevertheless, the deep recursive structure of this data makes it an interesting case for experiments.	82 MB		added Nov 2002

1. Resumo da implementação do tutorial do parser de XML usando Java

1.1 – Parser DOM

1.1.1 Com o ficheiro Simple.xml

```
C:\IS_1920\07_JAVA_com_XML\1-DOM>SETCLASSPATH.bat
C:\IS_1920\07_JAVA_com_XML\1-DOM>SET CLASSPATH=%C:\IS_1920\03_RPC\testarRPC\orpc.dev\lib\orpc.jar;
C:\IS_1920\07_JAVA_com_XML\1-DOM>java FileServer
Netbulia JavaRPC demo, not distributable!
Registered TCP transport.
Registered UDP transport.
Ficheiro recebido: simple.xml 1135 bytes transferidos
Ficheiro gravado: simple.xml
Closing socket.
```

```
C:\IS_1920\07_JAVA_com_XML\1-DOM>SETCLASSPATH.bat
C:\IS_1920\07_JAVA_com_XML\1-DOM>SET CLASSPATH=%C:\IS_1920\03_RPC\testarRPC\orpc.dev\lib\orpc.jar;
C:\IS_1920\07_JAVA_com_XML\1-DOM>java -cp ".;orpc.jar" pmapsvc
Netbulia JavaRPC demo, not distributable!
```

```
C:\IS_1920\07_JAVA_com_XML\1-DOM>SETCLASSPATH.bat
C:\IS_1920\07_JAVA_com_XML\1-DOM>SET CLASSPATH=%C:\IS_1920\03_RPC\testarRPC\orpc.dev\lib\orpc.jar;
C:\IS_1920\07_JAVA_com_XML\1-DOM>java -cp ".;orpc.jar" pmapsvc
Netbulia JavaRPC demo, not distributable!
```

```
C:\Program Files (x86)\Common Files\Oracle\Java\javapath\java.exe
TEXT: nodeName="#text" nodeValue=[WS]
ELEM: nodeName="price" localName="price"
TEXT: nodeName="#text" nodeValue="$4.50"
TEXT: nodeName="description" localName="description"
ELEM: nodeName="#text" nodeValue="Thick slices made from our homemade sourdough bread"
TEXT: nodeName="#text" nodeValue=[WS]
ELEM: nodeName="calories" localName="calories"
TEXT: nodeName="#text" nodeValue="600"
TEXT: nodeName="#text" nodeValue=[WS]
ELEM: nodeName="name" localName="name"
TEXT: nodeName="#text" nodeValue="Homestyle Breakfast"
TEXT: nodeName="#text" nodeValue=[WS]
ELEM: nodeName="price" localName="price"
TEXT: nodeName="#text" nodeValue="$6.95"
TEXT: nodeName="description" localName="description"
ELEM: nodeName="#text" nodeValue="Two eggs, bacon or sausage, toast, and our ever-popular hash browns"
TEXT: nodeName="calories" localName="calories"
TEXT: nodeName="#text" nodeValue="950"
TEXT: nodeName="#text" nodeValue=[WS]
ELEM: nodeName="description" localName="description"
TEXT: nodeName="#text" nodeValue="Two eggs, bacon or sausage, toast, and our ever-popular hash browns"
TEXT: nodeName="#text" nodeValue=[WS]
ELEM: nodeName="calories" localName="calories"
TEXT: nodeName="#text" nodeValue="950"
TEXT: nodeName="#text" nodeValue=[WS]
```

1.1.2 Com o ficheiro books.xml

```
C:\IS_1920\07_JAVA_com_XML\1-DOM>SETCLASSPATH.bat
C:\IS_1920\07_JAVA_com_XML\1-DOM>SET CLASSPATH=%C:\IS_1920\03_RPC\testarRPC\orpc.dev\lib\orpc.jar;
C:\IS_1920\07_JAVA_com_XML\1-DOM>java FileServer
Netbulia JavaRPC demo, not distributable!
Registered TCP transport.
Registered UDP transport.
Ficheiro recebido: simple.xml 1135 bytes transferidos
Ficheiro gravado: simple.xml
Closing socket.
```

```
C:\IS_1920\07_JAVA_com_XML\1-DOM>SETCLASSPATH.bat
C:\IS_1920\07_JAVA_com_XML\1-DOM>SET CLASSPATH=%C:\IS_1920\03_RPC\testarRPC\orpc.dev\lib\orpc.jar;
C:\IS_1920\07_JAVA_com_XML\1-DOM>java -cp ".;orpc.jar" pmapsvc
Netbulia JavaRPC demo, not distributable!
```

```
C:\IS_1920\07_JAVA_com_XML\1-DOM>SETCLASSPATH.bat
C:\IS_1920\07_JAVA_com_XML\1-DOM>SET CLASSPATH=%C:\IS_1920\03_RPC\testarRPC\orpc.dev\lib\orpc.jar;
C:\IS_1920\07_JAVA_com_XML\1-DOM>java -cp ".;orpc.jar" pmapsvc
Netbulia JavaRPC demo, not distributable!
```

```
C:\Program Files (x86)\Common Files\Oracle\Java\javapath\java.exe
TEXT: nodeName="#text" nodeValue=[WS]
ELEM: nodeName="price" localName="price"
TEXT: nodeName="#text" nodeValue="$4.50"
TEXT: nodeName="description" localName="description"
ELEM: nodeName="#text" nodeValue="Thick slices made from our homemade sourdough bread"
TEXT: nodeName="#text" nodeValue=[WS]
ELEM: nodeName="calories" localName="calories"
TEXT: nodeName="#text" nodeValue="600"
TEXT: nodeName="#text" nodeValue=[WS]
ELEM: nodeName="name" localName="name"
TEXT: nodeName="#text" nodeValue="Homestyle Breakfast"
TEXT: nodeName="#text" nodeValue=[WS]
ELEM: nodeName="price" localName="price"
TEXT: nodeName="#text" nodeValue="$6.95"
TEXT: nodeName="description" localName="description"
ELEM: nodeName="#text" nodeValue="Two eggs, bacon or sausage, toast, and our ever-popular hash browns"
TEXT: nodeName="#text" nodeValue=[WS]
ELEM: nodeName="calories" localName="calories"
TEXT: nodeName="#text" nodeValue="950"
TEXT: nodeName="#text" nodeValue=[WS]
ELEM: nodeName="description" localName="description"
TEXT: nodeName="#text" nodeValue="Two eggs, bacon or sausage, toast, and our ever-popular hash browns"
TEXT: nodeName="#text" nodeValue=[WS]
ELEM: nodeName="calories" localName="calories"
TEXT: nodeName="#text" nodeValue="950"
TEXT: nodeName="#text" nodeValue=[WS]
```

1.1.2 Com o ficheiro medsamp2014.xml

FileServer

```
C:\IS_1920\07_JAVA_com_XML\1-DOM>java FileServer
Netbulha JavaRPC demo, not distributable!
Registered TCP transport.
Registered UDP transport.
Ficheiro recebido: simple.xml 1135 bytes transferidos
Ficheiro gravado: simple.xml
Closing socket.
Ficheiro recebido: books.xml 4550 bytes transferidos
Ficheiro gravado: books.xml
Closing socket.
Ficheiro recebido: medsamp2014.xml 86016 bytes transferidos
Ficheiro gravado: medsamp2014.xml
Closing socket.
Ficheiro recebido: medsamp2014.xml 77824 bytes transferidos
Ficheiro gravado: medsamp2014.xml
Closing socket.
Ficheiro recebido: medsamp2014.xml 77824 bytes transferidos
Ficheiro gravado: medsamp2014.xml
Closing socket.
Ficheiro recebido: medsamp2014.xml 77824 bytes transferidos
Ficheiro gravado: medsamp2014.xml
Closing socket.
Ficheiro recebido: medsamp2014.xml 77824 bytes transferidos
Ficheiro gravado: medsamp2014.xml
Closing socket.
```

JavaRPC

```
C:\IS_1920\07_JAVA_com_XML\1-DOM>SETCLASSPATH.bat
C:\IS_1920\07_JAVA_com_XML\1-DOM>SET CLASSPATH=C:\IS_1920\03_RPC\testarRPC\orpc.dev\lib\orpc.jar;
C:\IS_1920\07_JAVA_com_XML\1-DOM>java -cp ".;orpc.jar" pmapsvc
Netbulha JavaRPC demo, not distributable!
```

FileClient

```
C:\IS_1920\07_JAVA_com_XML\1-DOM>java FileClient localhost medsamp2014.xml
Netbulha JavaRPC demo, not distributable!
Ligado ao servidor: localhost
medsamp2014.xml 86016 bytes enviados

C:\IS_1920\07_JAVA_com_XML\1-DOM>java FileClient localhost medsamp2014.xml
Netbulha JavaRPC demo, not distributable!
Ligado ao servidor: localhost
medsamp2014.xml 77824 bytes enviados

C:\IS_1920\07_JAVA_com_XML\1-DOM>java FileClient localhost medsamp2014.xml
Netbulha JavaRPC demo, not distributable!
Ligado ao servidor: localhost
medsamp2014.xml 77824 bytes enviados

C:\IS_1920\07_JAVA_com_XML\1-DOM>java FileClient localhost medsamp2014.xml
Netbulha JavaRPC demo, not distributable!
Ligado ao servidor: localhost
medsamp2014.xml 77824 bytes enviados

C:\IS_1920\07_JAVA_com_XML\1-DOM>
```

Output

Figura 3.c

1.2 – Parser SAX

1.2.1 Com o ficheiro note.xml

FileServer

```
C:\IS_1920\07_JAVA_com_XML\2-SAX>java FileServer
Netbulha JavaRPC demo, not distributable!
Registered UDP transport.
Ficheiro recebido: books.xml 4550 bytes transferidos
Ficheiro gravado: books.xml
Closing socket.
Ficheiro recebido: books.xml 4550 bytes transferidos
Ficheiro gravado: books.xml
Closing socket.
Ficheiro recebido: books.xml 4550 bytes transferidos
Ficheiro gravado: books.xml
Closing socket.
Ficheiro recebido: note.xml 145 bytes transferidos
Ficheiro gravado: note.xml
Closing socket.
Ficheiro recebido: note.xml 145 bytes transferidos
Ficheiro gravado: note.xml
Closing socket.
Ficheiro recebido: note.xml 145 bytes transferidos
Ficheiro gravado: note.xml
Closing socket.
```

JavaRPC

```
C:\IS_1920\07_JAVA_com_XML\2-SAX>java -cp ".;orpc.jar" pmapsvc
Netbulha JavaRPC demo, not distributable!
```

Output*

```
C:\IS_1920\07_JAVA_com_XML\2-SAX>javac sax/*
javac: invalid flag: sax/SAXLocalNameCount$MyErrorHandler.class
Usage: javac [options] <source files>
use -help for a list of possible options

C:\IS_1920\07_JAVA_com_XML\2-SAX>java sax/SAXLocalNameCount "C:\IS_1920\07_JAVA_com_XML\2-SAX\note.xml"
Local Name "body" occurs 1 times
Local Name "text" occurs 1 times
Local Name "heading" occurs 1 times
Local Name "from" occurs 1 times
Local Name "to" occurs 1 times

C:\IS_1920\07_JAVA_com_XML\2-SAX>
```

FileClient

```
C:\IS_1920\07_JAVA_com_XML\2-SAX>javac sax/*
javac: invalid flag: sax/SAXLocalNameCount$MyErrorHandler.class
Usage: javac [options] <source files>
use -help for a list of possible options

C:\IS_1920\07_JAVA_com_XML\2-SAX>java sax/SAXLocalNameCount "C:\IS_1920\07_JAVA_com_XML\2-SAX\note.xml"
Local Name "note" occurs 1 times
Local Name "heading" occurs 1 times
Local Name "text" occurs 1 times
Local Name "to" occurs 1 times

C:\IS_1920\07_JAVA_com_XML\2-SAX>
```

Figura 5.c

*SAXLocalNameCount.java conta quantas vezes estava presente cada elemento no ficheiro

1.2.2 Com o ficheiro books.xml

The figure shows four terminal windows comparing the performance of FileServer and JavaRPC for processing the books.xml file.

- FileServer:** Shows the server side of the FileServer application. It receives the file, processes it, and then sends it back to the client. The output shows the file being read and written multiple times.
- JavaRPC:** Shows the client side of the JavaRPC application. It connects to the server, sends the file, and receives the response. The output shows the file being sent in 4550 byte chunks.
- Output***: Shows the results of running SAXLocalNameCount.java on the FileClient side. It counts the occurrences of various XML elements like genre, catalog, book, etc.
- FileClient:** Shows the client side of the FileClient application. It connects to the server, sends the file, and receives the response. The output shows the file being sent in 4550 byte chunks.

*SAXLocalNameCount.java conta quantas vezes estava presente cada elemento no ficheiro

1.2.3 Com o ficheiro medsamp2013.xml

The figure shows four terminal windows comparing the performance of FileServer and JavaRPC for processing the medsamp2013.xml file.

- FileServer:** Shows the server side of the FileServer application. It receives the file, processes it, and then sends it back to the client. The output shows the file being read and written multiple times.
- JavaRPC:** Shows the client side of the JavaRPC application. It connects to the server, sends the file, and receives the response. The output shows the file being sent in 77824 byte chunks.
- Output***: Shows the results of running SAXLocalNameCount.java on the FileClient side. It counts the occurrences of various XML elements like note, med, etc.
- FileClient:** Shows the client side of the FileClient application. It connects to the server, sends the file, and receives the response. The output shows the file being sent in 77824 byte chunks.

*ERRO DE EXECUÇÃO: DTD do ficheiro com notações “incorrectas”. Desafio: Necessário analisar a sintaxe do DTD.

1.3 – Parser JDOM

1.3.1 Com o ficheiro note.xml

The screenshot shows four windows illustrating the interaction between a FileServer, JavaRPC, and FileClient.

- FileServer:** A command-line window showing the execution of `java FileServer`. It outputs the Java code for the JDOM parser and the registration of the TCP transport.
- JavaRPC:** A command-line window showing the execution of `java -cp ".;orpc.jar" pmapservc`. It outputs the Java code for the JavaRPC demo and the registration of the TCP transport.
- FileClient:** A command-line window showing the execution of `java FileClient localhost books.xml` and `java FileClient localhost note.xml`. It outputs the XML files sent to the server.
- Output:** An Adobe Acrobat Reader window displaying the received XML file (`note.xml`). The XML contains a heading and a body with a reminder message.

Figura 8.c

```

<?xml version="1.0" encoding="UTF-8"?>
<note>
  <to>you</to>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>

```

Figura 8.d

1.3.2 Com o ficheiro books.xml

The screenshot shows four windows illustrating the interaction between a FileServer, JavaRPC, and FileClient.

- FileServer:** A command-line window showing the execution of `java FileServer`. It outputs the Java code for the JDOM parser and the registration of the UDP transport.
- JavaRPC:** A command-line window showing the execution of `java -cp ".;orpc.jar" pmapservc`. It outputs the Java code for the JavaRPC demo and the registration of the UDP transport.
- FileClient:** A command-line window showing the execution of `java FileClient localhost books.xml`. It outputs the XML file sent to the server.
- Output:** An Adobe Acrobat Reader window displaying the received XML file (`books.xml`). The XML describes a catalog with two books, each with authors, titles, and descriptions.

Figura 7.c

```

<catalog>
  <book id="b01">
    <author>O' Brian, Tim</author>
    <title>Microsoft's .NET initiative is explored in detail, with attention to XML DOM interfaces, XSLT processing, and XSL transforms</title>
    <genre>Computer</genre>
    <publish_date>2000-12-01</publish_date>
    <description>Microsoft's .NET initiative is explored in detail, with attention to XML DOM interfaces, XSLT processing, and XSL transforms</description>
  </book>
  <book id="b02">
    <author>Gates, Mike</author>
    <title>Microsoft Visual Studio .NET is explored in depth, including the .NET Framework, C#, VB.NET, and ASP.NET are integrated into a comprehensive development environment.</title>
    <genre>Computer</genre>
    <publish_date>2000-04-16</publish_date>
    <description>Microsoft Visual Studio .NET is explored in depth, including the .NET Framework, C#, VB.NET, and ASP.NET are integrated into a comprehensive development environment.</description>
  </book>
</catalog>

```

Figura 7.d

1.3.3 Com o ficheiro medsamp2013.xml

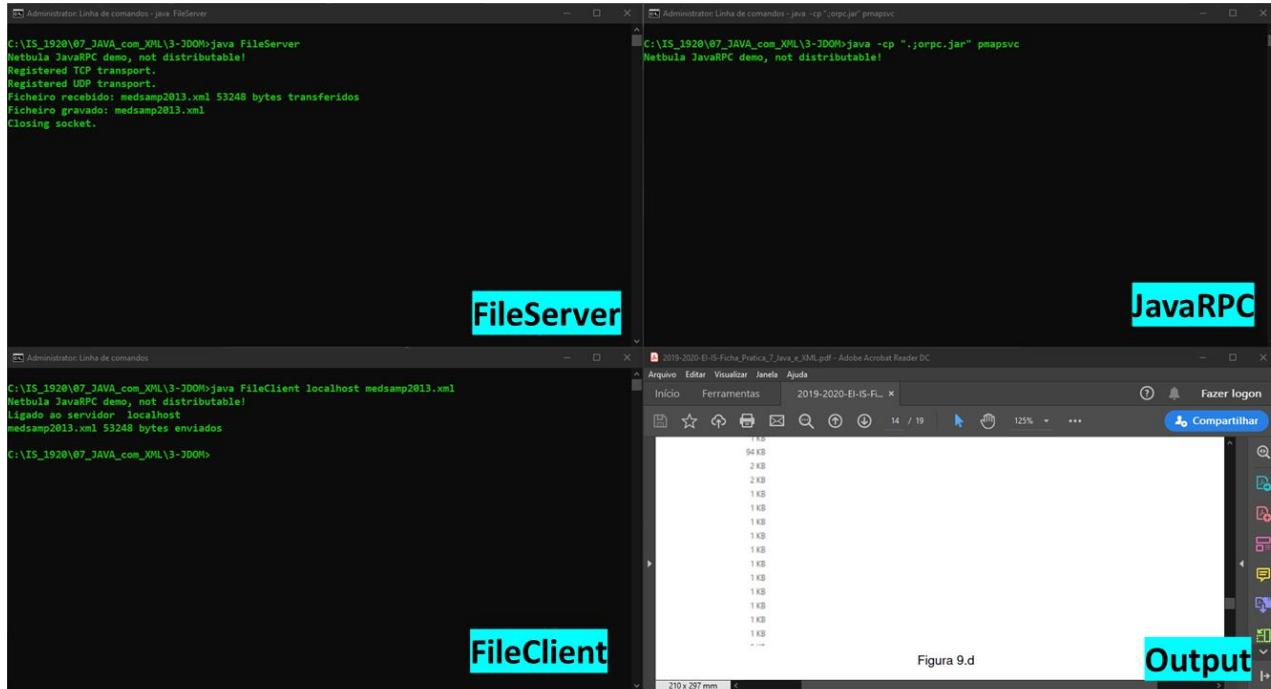
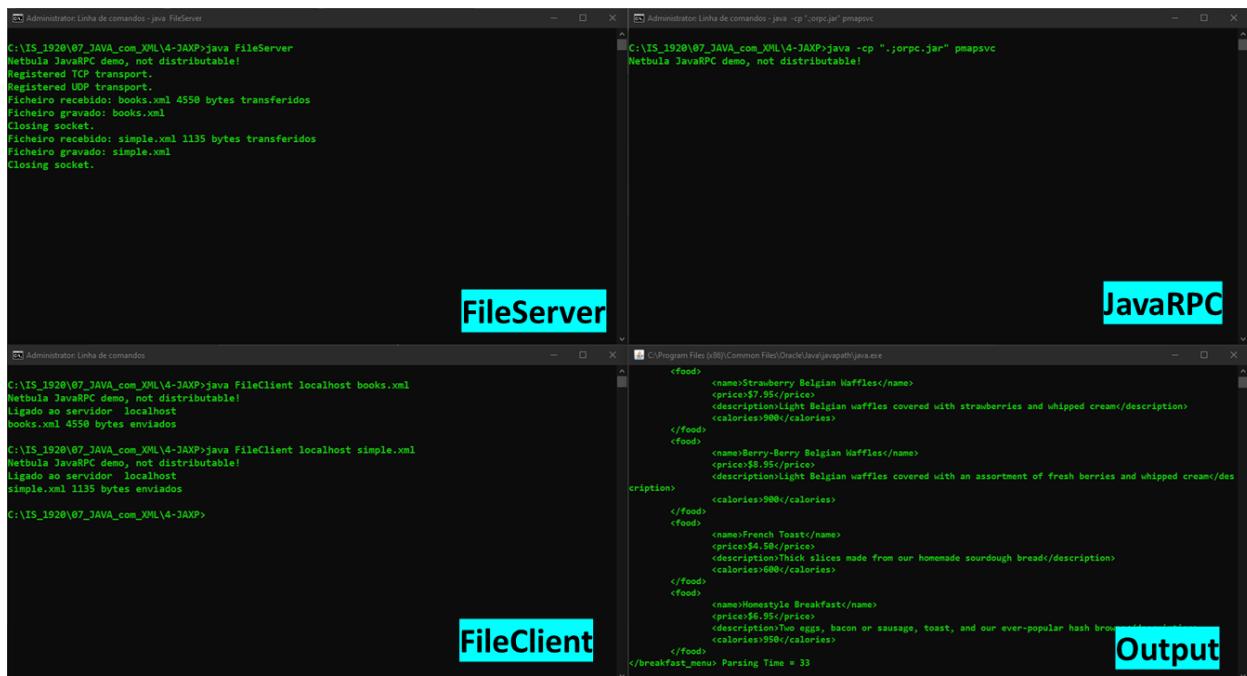


Figura 9.d

1.4 – Parser JAXP

1.4.1 Com o ficheiro simple.xml



1.4.2 Com o ficheiro books.xml

The screenshot displays four windows illustrating the interaction between a FileServer and a JavaRPC system:

- FileServer**: Shows the command-line interface (CLI) for the FileServer, which receives and processes the XML file.
- JavaRPC**: Shows the command-line interface for JavaRPC, which sends the XML file to the FileServer.
- FileClient**: Shows the command-line interface for the FileClient, which sends the XML file to the JavaRPC system.
- Output**: Shows the XML response received by the FileClient from the JavaRPC system.

```

Administrator: Linha de comandos - java FileServer
C:\IS_1920\07_JAVA_com_XML\4-JAXP>java FileServer
Netbulus JavaRPC demo, not distributable!
Registered TCP transport.
Registered UDP transport.
Ficheiro recebido: books.xml 4550 bytes transferidos
Ficheiro gravado: books.xml
Closing socket.

Administrator: Linha de comandos - java -cp ".;orpc.jar" pmapsvc
C:\IS_1920\07_JAVA_com_XML\4-JAXP>java -cp ".;orpc.jar" pmapsvc
Netbulus JavaRPC demo, not distributable!

Administrator: Linha de comandos - java -cp ".;orpc.jar" pmapsvc
C:\IS_1920\07_JAVA_com_XML\4-JAXP>java -cp ".;orpc.jar" pmapsvc
Netbulus JavaRPC demo, not distributable!

Administrator: Linha de comandos - java FileClient localhost books.xml
C:\IS_1920\07_JAVA_com_XML\4-JAXP>java FileClient localhost books.xml
Netbulus JavaRPC demo, not distributable!
Ligado ao servidor: localhost
books.xml 4550 bytes enviados
C:\IS_1920\07_JAVA_com_XML\4-JAXP>

Administrator: Linha de comandos - java -cp ".;orpc.jar" pmapsvc
C:\Program Files (x86)\Common Files\Oracle\Java\javapath>java.exe
<publish_date>2000-12-09</publish_date>
<description>Microsoft's .NET initiative is explored in detail in this deep programmer's reference.</description>
<book id="b111">
<author>O'Brian, Tim</author>
<title>MSXML 4 A Comprehensive Guide</title>
<genre>Computer</genre>
<price>36.95</price>
<publish_date>2000-12-01</publish_date>
<description>The Microsoft MSXML parser is covered in detail, with attention to XML DOM interfaces, XSLT processing, SAX and more.</description>
</book>
<book id="b112">
<author>Galas, Mike</author>
<title>Visual Studio 7: A Comprehensive Guide</title>
<genre>Computer</genre>
<price>36.95</price>
<publish_date>2001-04-16</publish_date>
<description>Microsoft Visual Studio 7 is explored in depth, looking at how Visual Basic, Visual C++, C#, and ASP+ are integrated into a comprehensive development environment.</description>
</book>
</catalog> Parsing Time = 83

```

1.4.3 Com o ficheiro medsamp2014.xml

The screenshot displays four windows illustrating the interaction between a FileServer and a JavaRPC system:

- FileServer**: Shows the command-line interface (CLI) for the FileServer, which receives and processes the XML file.
- JavaRPC**: Shows the command-line interface for JavaRPC, which sends the XML file to the FileServer.
- FileClient**: Shows the command-line interface for the FileClient, which sends the XML file to the JavaRPC system.
- Output***: Shows the XML response received by the FileClient from the JavaRPC system, including an error message.

```

Administrator: Linha de comandos - java FileServer
C:\IS_1920\07_JAVA_com_XML\4-JAXP>java FileServer
Netbulus JavaRPC demo, not distributable!
Registered TCP transport.
Registered UDP transport.
Ficheiro recebido: books.xml 4550 bytes transferidos
Ficheiro gravado: books.xml
Closing socket.
Ficheiro recebido: simple.xml 1135 bytes transferidos
Ficheiro gravado: simple.xml
Closing socket.
Ficheiro recebido: medsamp2014.xml 61440 bytes transferidos
Ficheiro gravado: medsamp2014.xml
Closing socket.

Administrator: Linha de comandos - java -cp ".;orpc.jar" pmapsvc
C:\IS_1920\07_JAVA_com_XML\4-JAXP>java -cp ".;orpc.jar" pmapsvc
Netbulus JavaRPC demo, not distributable!

Administrator: Linha de comandos - java -cp ".;orpc.jar" pmapsvc
C:\IS_1920\07_JAVA_com_XML\4-JAXP>java -cp ".;orpc.jar" pmapsvc
Netbulus JavaRPC demo, not distributable!

Administrator: Linha de comandos - java FileClient localhost books.xml
C:\IS_1920\07_JAVA_com_XML\4-JAXP>java FileClient localhost books.xml
Netbulus JavaRPC demo, not distributable!
Ligado ao servidor: localhost
books.xml 4550 bytes enviados

Administrator: Linha de comandos - java FileClient localhost simple.xml
C:\IS_1920\07_JAVA_com_XML\4-JAXP>java FileClient localhost simple.xml
Netbulus JavaRPC demo, not distributable!
Ligado ao servidor: localhost
simple.xml 1135 bytes enviados

Administrator: Linha de comandos - java FileClient localhost medsamp2014.xml
C:\IS_1920\07_JAVA_com_XML\4-JAXP>java FileClient localhost medsamp2014.xml
Netbulus JavaRPC demo, not distributable!
Ligado ao servidor: localhost
medsamp2014.xml 61440 bytes enviados

Administrator: Linha de comandos - java -cp ".;orpc.jar" pmapsvc
C:\Program Files (x86)\Common Files\Oracle\Java\javapath>java.exe
FACTORY: com.sun.xml.internal.stream.XMLInputFactoryImpl@8c647e05
filename = medsamp2014.xml
<?xml version="1.0" encoding="UTF-8"?>
ParseError at [row,col]=[1,2]
Message: The markup declarations contained or pointed to by the document type declaration must be well-formed.
Parsing Time = 392

```

*ERRO DE EXECUÇÃO: DTD do ficheiro com notações “incorrectas”. Desafio: Necessário analisar a sintaxe do DTD.

1.4.4 Com o ficheiro po.xml

The screenshot displays four terminal windows arranged in a 2x2 grid, illustrating the JavaRPC demo. The top-left window is labeled 'FileServer' and shows the command-line interface for the JavaRPC server. The top-right window is labeled 'JavaRPC' and shows the command-line interface for the JavaRPC client. The bottom-left window is labeled 'FileClient' and shows the command-line interface for the FileClient application. The bottom-right window is labeled 'Output' and shows the results of the file transfer operation.

FileServer:

```
C:\IS_1920\07_JAVA_com_XML\5-JAXB\samples\unmarshal-read>java FileServer
Netbulia JavaRPC demo, not distributable!
Registered TCP transport.
Registered UDP transport.
Ficheiro recebido: po.xml 2835 bytes transferidos
Ficheiro gravado: po.xml
Closing socket.
```

JavaRPC:

```
C:\IS_1920\07_JAVA_com_XML\5-JAXB\samples\unmarshal-read>java -cp ".;orpc.jar" pmapsvc
C:\IS_1920\07_JAVA_com_XML\5-JAXB\samples\unmarshal-read>java -cp ".;orpc.jar" pmapsvc
Netbulia JavaRPC demo, not distributable!
```

FileClient:

```
C:\IS_1920\07_JAVA_com_XML\5-JAXB\samples\unmarshal-read>java FileClient localhost po.xml
Netbulia JavaRPC demo, not distributable!
Ligado ao servidor: localhost
po.xml 2835 bytes enviados
C:\IS_1920\07_JAVA_com_XML\5-JAXB\samples\unmarshal-read>
```

Output:

```
Ship the following items to:
  Alice Smith
  123 Maple Street
  Cambridge, MA 12345
  US

  5 copies of "Nosferatu - Special Edition (1929)"
  3 copies of "The Mummy (1932)"
  3 copies of "Godzilla and Mothra: Battle for Earth/Godzilla vs. King Ghidorah"
```

2. Implementação de Webservices SOAP e REST integrando ficheiros XML em Base de Dados usando a abordagem Cliente/Servidor

Sugere-se a utilização das seguintes ferramentas/ambiente de desenvolvimento:

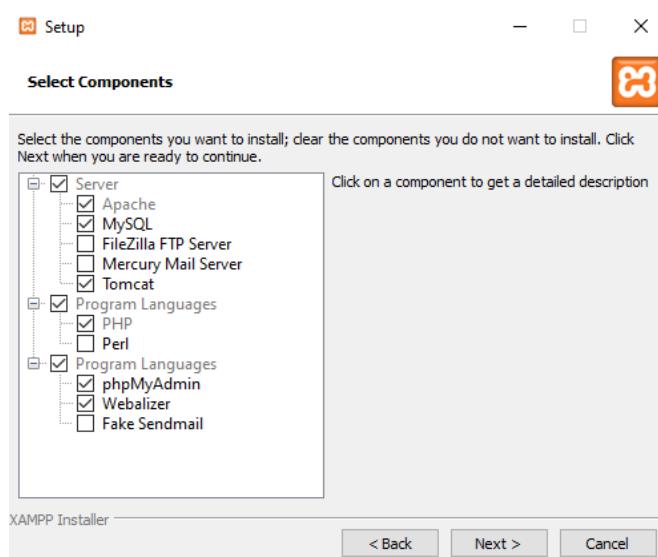
- Netbeans IDE 8.2 ALL (JAVA WebServices); Glassfish (Java WebServices – Servidor WEB); Visual Studio Code (PHP WebServices); Plataforma WAMP – 3.1.9 64bit (PHP WebServices) + Apache 2.4.39 + Mysql 5.7.26 • PHP 7.2.18; Testes: • SOAP UI • POSTMAN • BROWSER (F12 Modo Developer)

2.1 Enquadramento

Com base nos conhecimentos adquiridos da implementação de um parser em JAVA (com várias abordagens), pretende-se que se criem WebServices (SOAP e REST) implementados em Java e em PHP para receber no lado do servidor um ficheiro XML e o importar para uma tabela de uma base de dados.

Neste contexto, para:

- a implementação dos Webservices em Java iremos utilizar o IDE – Interface Desktop Environment netBeans com o módulo “mysql-connector-java-8.0.18.jar” para fazer a ligação à base de dados. Este ficheiro jar deverá estar na diretoria “build\web\WEB-INF\lib”.
- **NOTA:** As configurações de acesso à base de dados estão definidas no ficheiro “IS_TP1_JAVA\src\java\com\is\db\Alunos_ws_rest.java”.
- a implementação dos webservices em PHP iremos ter de instalar um servidor WEB. Neste caso instalaremos o XAMPP, disponível em: <https://www.apachefriends.org/index.html>

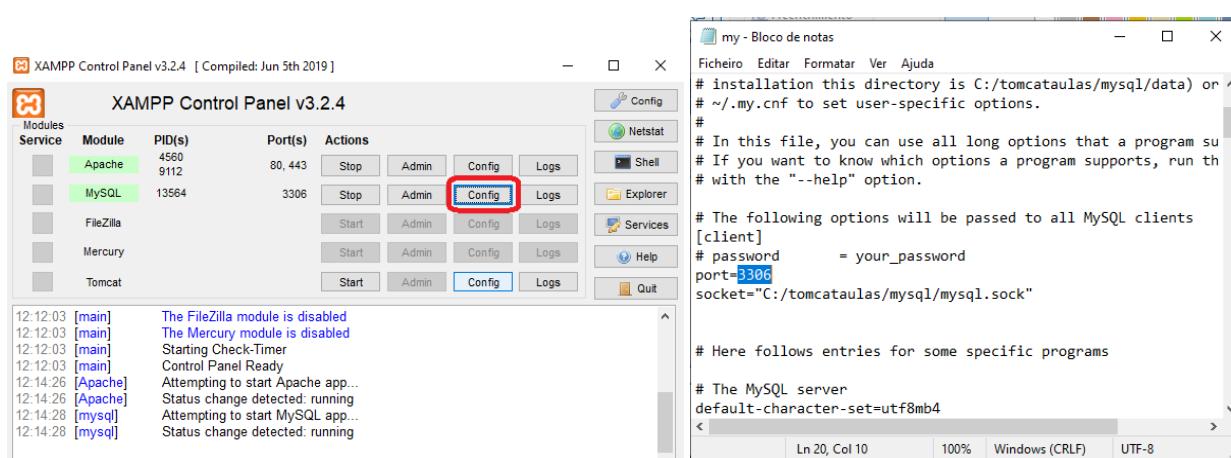


NOTA: Para simplificar a execução deste tutorial, assume-se que o servidor WEB irá ser executado **em localhost:3306** e que são definidas na base de dados mySQL as credenciais **username: userdb password: teste** (em vez de **username: root e password: root**).

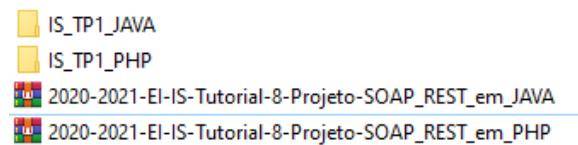
NOTA: A porta é a 3306 uma vez que por defeito o mySQL fica à escuta nessa porta. Poderá ver as configurações (ou alterar) através da configuração do ficheiro “my.ini” através do botão “Config”:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação



Considere os ficheiros ZIP com o código de implementação, “2020-2021-EI-IS-Tutorial-8-Projeto-SOAP_REST_em_JAVA.zip” e “2020-2021-EI-IS-Tutorial-8-Projeto-SOAP_REST_em_PHP.zip”:



Descompacte os ficheiros dentro de uma diretoria (ex. tutorial_8), as quais deverão ter a seguinte estrutura:

Em JAVA:	Em PHP:
<ul style="list-style-type: none"> IS_TP1_JAVA <ul style="list-style-type: none"> build empty generated-sources web <ul style="list-style-type: none"> images META-INF WEB-INF dist lib nbproject src test web 	<ul style="list-style-type: none"> IS_TP1_PHP <ul style="list-style-type: none"> REST <ul style="list-style-type: none"> db DOM SAX SOAP <ul style="list-style-type: none"> db DOM SAX

2.2 Implementação em JAVA

Para facilitar a edição dos ficheiros e a sua execução utilizou-se o IDE – Interface Desktop Environment NetBeans, disponível em: <https://netbeans.org/>.

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação



Recomenda-se a leitura desta informação:

- Getting Started with JAX-WS Web Services: <https://netbeans.org/kb/docs/websvc/jax-ws.html>
- NetBeans Developing Applications with NetBeans IDE: https://docs.oracle.com/netbeans/nb82/netbeans/NBDAG/work_app_servers.htm#NBDAG2198
- Packaging and Deploying Desktop Java Applications: https://netbeans.org/kb/articles/avase-deploy.html#Exercise_1

The screenshot shows the 'Getting Started with JAX-WS Web Services' section of the NetBeans documentation. A red box highlights the 'Consuming the Web Service' section, which includes links to 'a Java class in a Java SE Application', 'a servlet in a web application', and 'a JSP page in a web application'.

The screenshot shows the 'Downloading Apache NetBeans 12.1' page. It lists various download options including binaries, installers, and source code. A red arrow points to the 'Binaries' link.

Deployment platforms

Apache NetBeans 12.1 runs on JDK LTS releases 8 and 11, as well as on JDK 14, i.e., the current JDK release at the time of this NetBeans release.

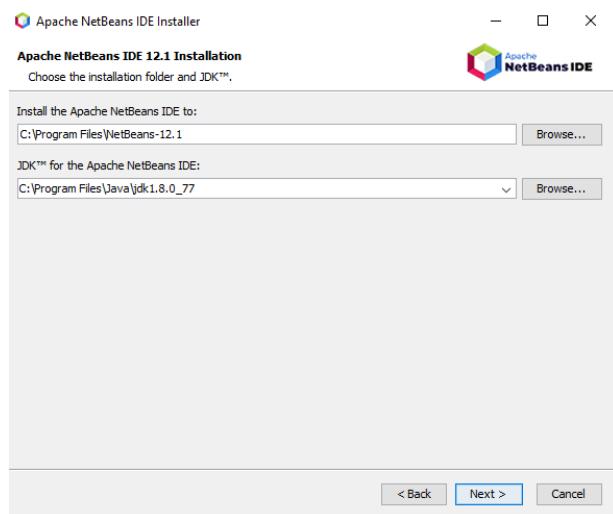
Building from source

To build Apache NetBeans 12.1 from source you need:

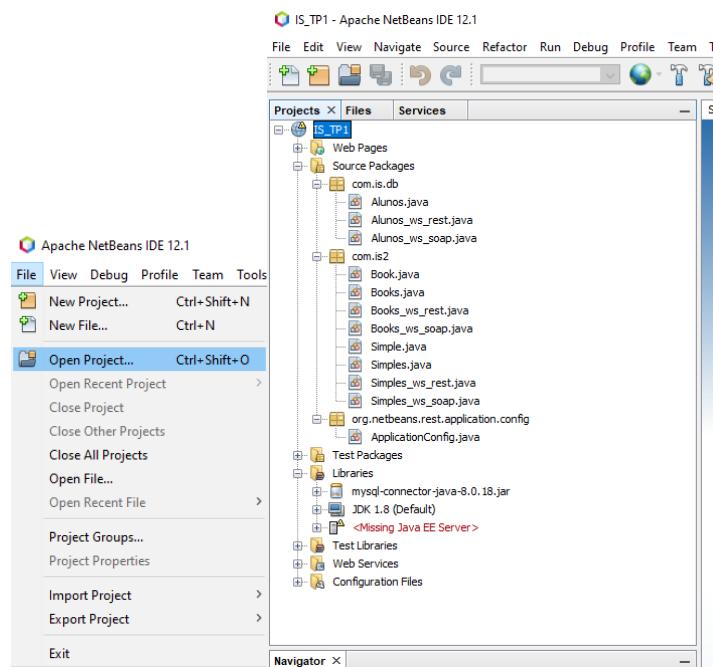
The screenshot shows the '23 Working with Web and Application Servers' page. A red box highlights the 'Table of Contents' section, which lists various topics such as 'About Working with Web and Application Servers', 'Working with Web Browsers', 'Working with Glassfish Application Servers', and 'Working with the HTTP Server-Side Monitor'.

FICHA PRÁTICA n.º 9

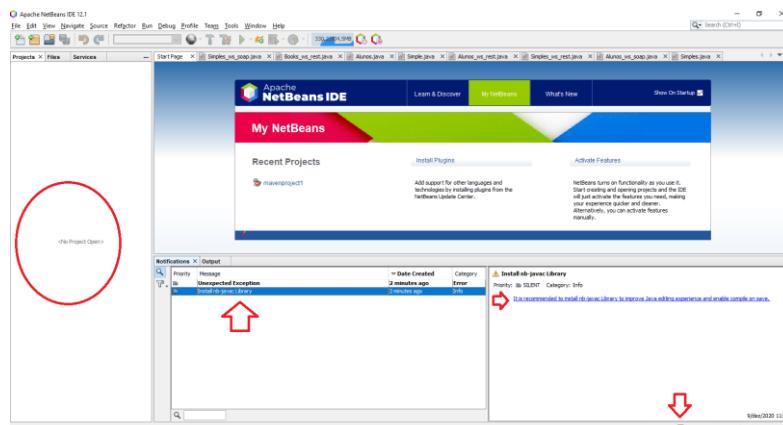
Integração de Sistemas de Informação



Importação do Projeto:



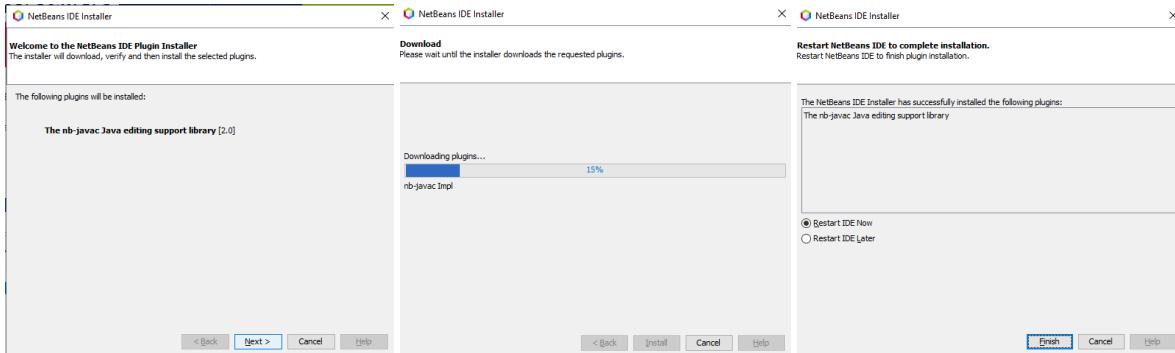
Se o projeto não foi importado, ver os erros de instalação dos plugins:



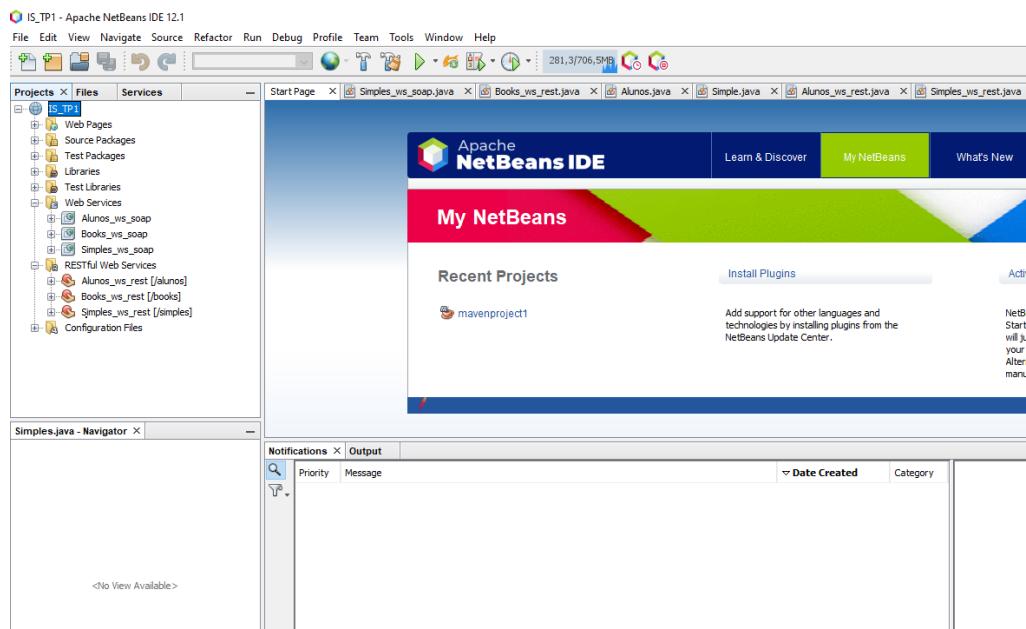
FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

Exemplo de necessidade de instalação: Is needed to install the “nb-javac Java editing support library”
Selecione a linha “Install nb-javac library” and in the link “It is recommended to install...”



Depois, importar novamente o projeto:



Atualização das Librarias/Packages:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

The screenshot shows the Apache NetBeans IDE interface. In the code editor, a Java file named Alunos.java is open, showing imports for javax.persistence and javax.validation.constraints. A red box highlights the persistence-related imports. The output window below shows the build process and errors related to missing packages like javax.ws.rs.

```

1 package com.is.db;
2
3 // Imports
4 import java.io.Serializable;
5 import javax.persistence.Basic;
6 import javax.persistence.Column;
7 import javax.persistence.Entity;
8 import javax.persistence.GeneratedValue;
9 import javax.persistence.GenerationType;
10 import javax.persistence.Id;
11 import javax.persistence.NamedQueries;
12 import javax.persistence.NamedQuery;
13 import javax.persistence.Table;
14 import javax.validation.constraints.NotNull;
15 import javax.validation.constraints.Size;
16 import javax.xml.bind.annotation.XmlRootElement;
17
18
19 @Entity
20 @Table(name = "alunos")
21 @XmlRootElement
22 // Retificação da método equals para a queries
23 @NamedQueries({
24     @NamedQuery(name = "Alunos.findAll", query = "SELECT * FROM Alunos"),
25     @NamedQuery(name = "Alunos.findById", query = "SELECT * FROM Alunos a WHERE a.id = :id"),
26     @NamedQuery(name = "Alunos.findByNome", query = "SELECT * FROM Alunos a WHERE a.nome = :nome")
27 })
28
29 public class Alunos implements Serializable {
30
31     // Attributos
32     private static final long serialVersionUID = 1L;
33 }

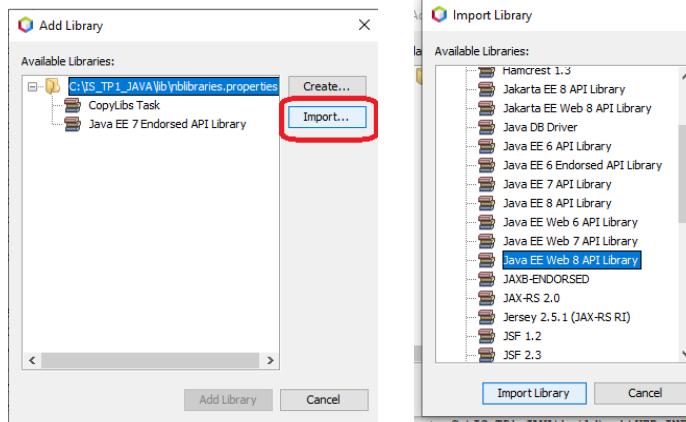
```

Output - IS_TP1 (clean,dist) X

```

Copying 1 file to C:\IS_TP1_JAVA\build\web\WEB-INF\lib
Copying 1 file to C:\IS_TP1_JAVA\build\web\WEB-INF\lib
library-inclusion-in-manifest:
Created dir: C:\IS_TP1_JAVA\build\empty
Created dir: C:\IS_TP1_JAVA\build\generated-sources\ap-source-output
Compiling 12 source files to C:\IS_TP1_JAVA\build\web\WEB-INF\classes
C:\IS_TP1_JAVA\src\java\com\is2\Books_ws_rest.java:6: error: package javax.ws.rs does not exist
import javax.ws.rs.*;
C:\IS_TP1_JAVA\src\java\com\is2\Books_ws_rest.java:7: error: package javax.ws.rs.core does not exist
import javax.ws.rs.core.*;
<

```



Importando o package Java EE API Library:

The screenshot shows the Apache NetBeans IDE interface. In the code editor, the same Alunos.java file is shown with the Java EE API library imported. The library manager window is open, showing the selected 'Java EE 7 Endorsed API Library'. A red box highlights the 'Import Library' button.

```

1 package com.is.db;
2
3 // Imports
4 import java.io.Serializable;
5 import javax.persistence.Basic;
6 import javax.persistence.Column;
7 import javax.persistence.Entity;
8 import javax.persistence.GeneratedValue;
9 import javax.persistence.GenerationType;
10 import javax.persistence.Id;
11 import javax.persistence.NamedQueries;
12 import javax.persistence.NamedQuery;
13 import javax.persistence.Table;
14 import javax.validation.constraints.NotNull;
15 import javax.validation.constraints.Size;
16 import javax.xml.bind.annotation.XmlRootElement;
17
18
19 @Entity
20 @Table(name = "alunos")
21 @XmlRootElement
22 // Retificação da método equals para a queries
23 @NamedQueries({
24     @NamedQuery(name = "Alunos.findAll", query = "SELECT * FROM Alunos"),
25     @NamedQuery(name = "Alunos.findById", query = "SELECT * FROM Alunos a WHERE a.id = :id"),
26     @NamedQuery(name = "Alunos.findByNome", query = "SELECT * FROM Alunos a WHERE a.nome = :nome")
27 })
28
29 public class Alunos implements Serializable {
30
31     // Attributos
32     private static final long serialVersionUID = 1L;
33 }

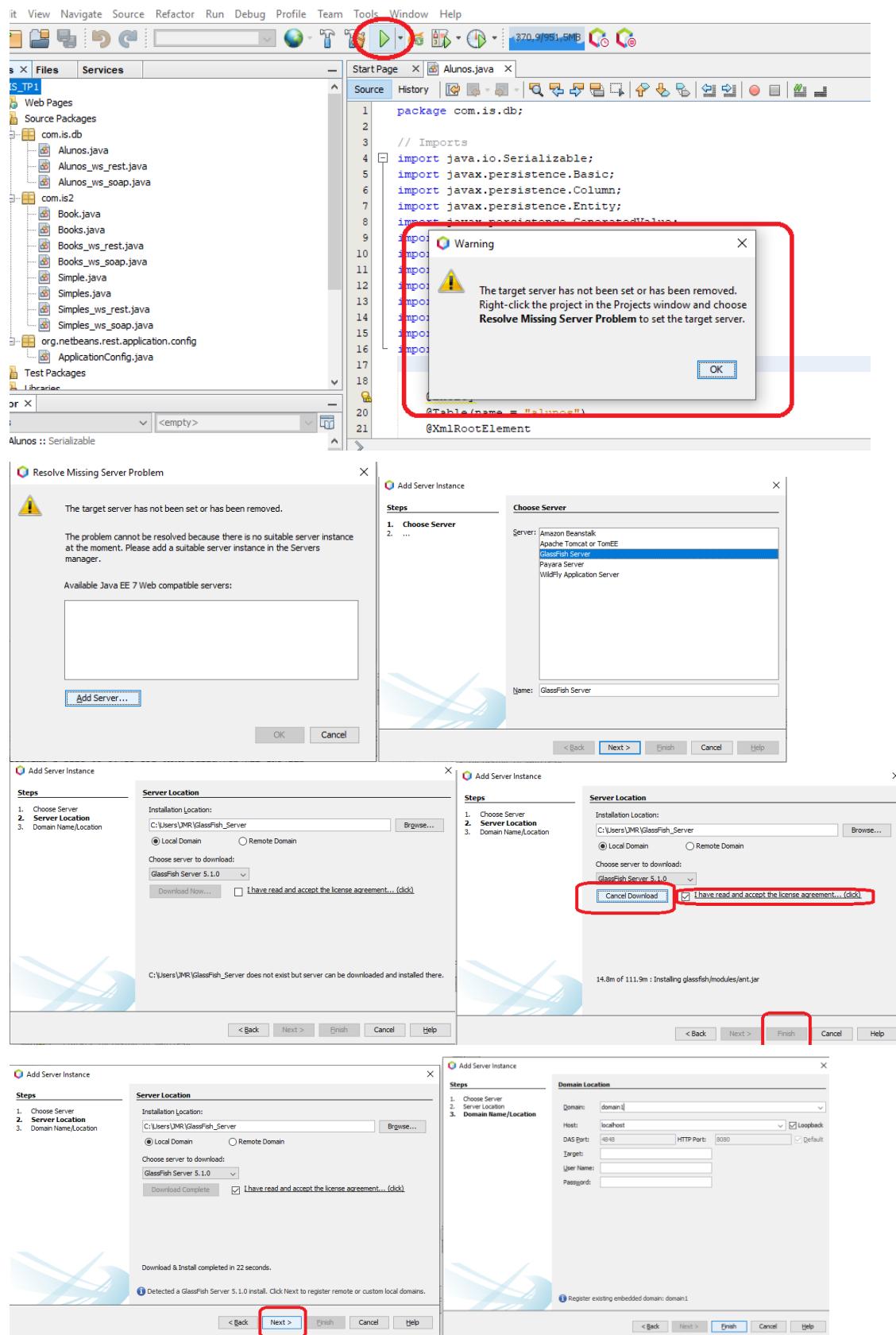
```

Preparando o arranque do projeto:

FICHA PRÁTICA n.º 9

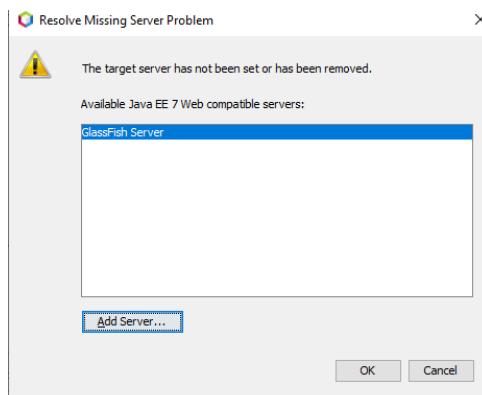
Integração de Sistemas de Informação

P1 - Apache NetBeans IDE 12.1

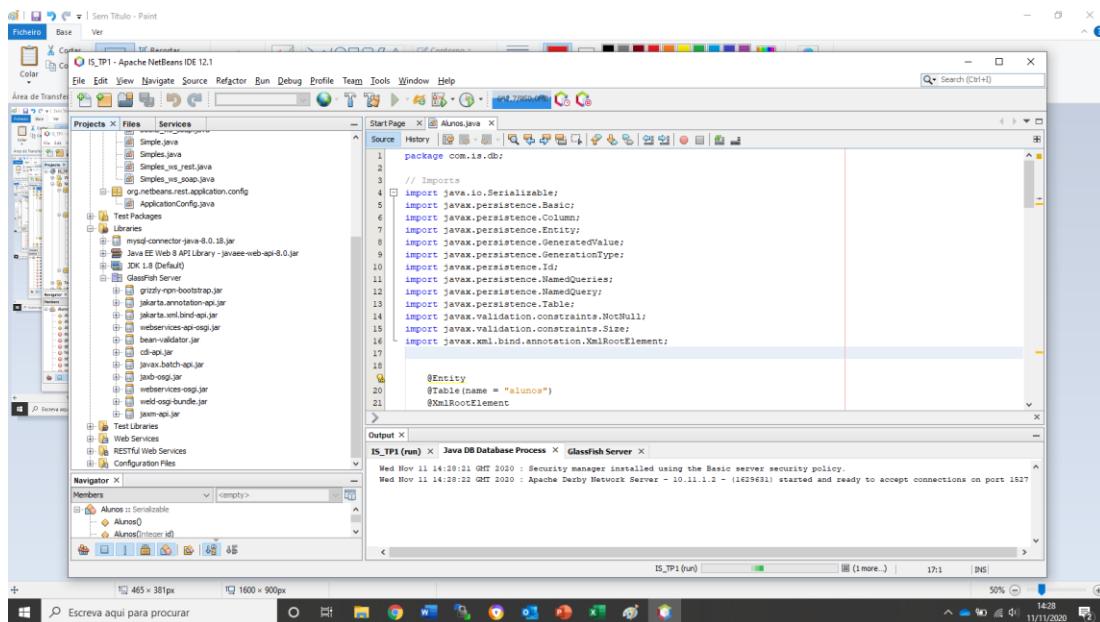


FICHA PRÁTICA n.º 9

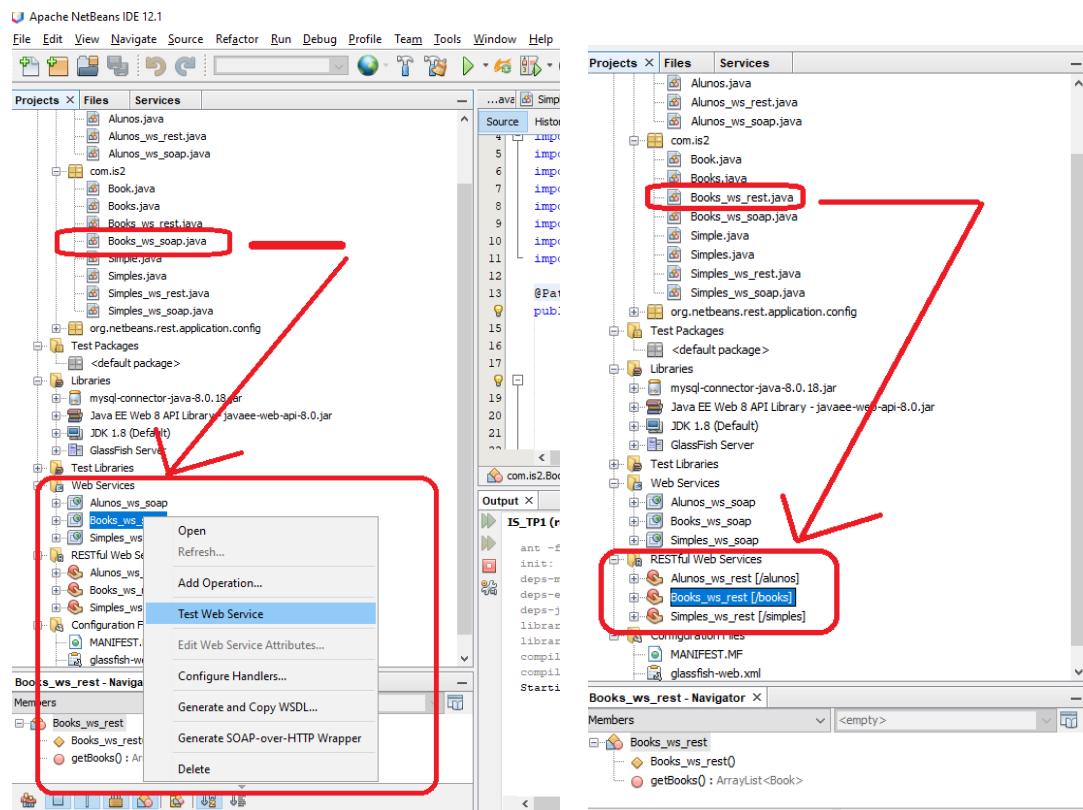
Integração de Sistemas de Informação



O projeto está pronto para ser executado:



Neste projeto o URL do tipo de webservices são diferentes:

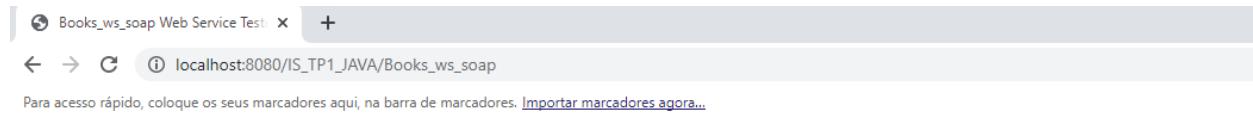


Using SOAP: http://localhost:8080/IS_TP1_JAVA/Books_ws_soap?Tester

Using REST: http://localhost:8080/IS_TP1_JAVA/webresources/books

Testando o webservice: http://localhost:8080/IS_TP1_JAVA/Simples_ws_soap?Tester ou http://localhost:8080/IS_TP1_JAVA/Books_ws_soap?Tester

NOTA: Indicar no URL a expressão ?Tester



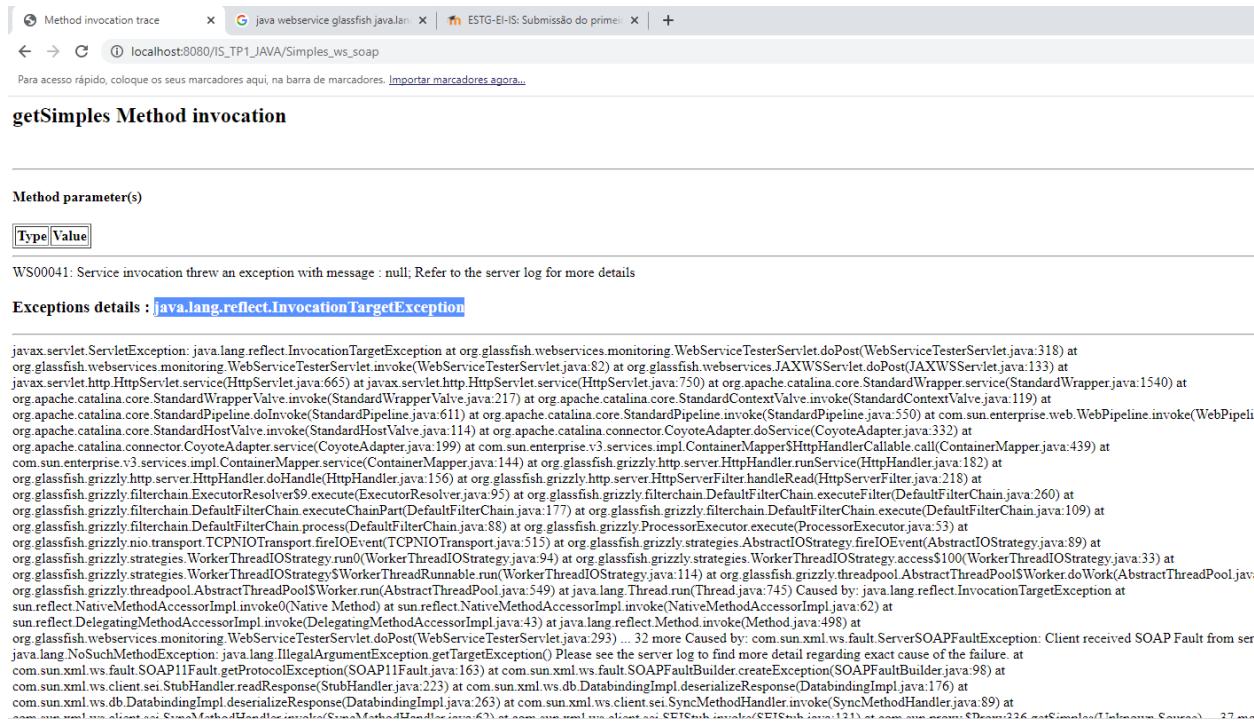
This form will allow you to test your web service implementation ([WSDL File](#))

To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.

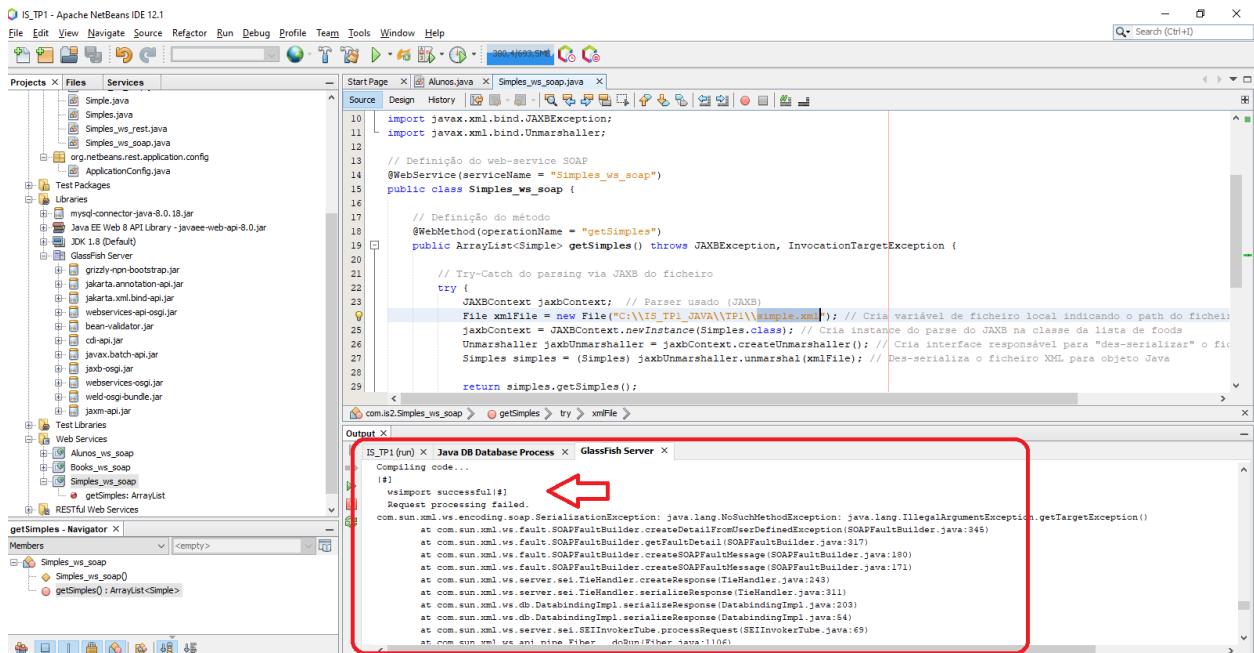
Methods :

```
public abstract java.util.List com.is2.BooksWsSoap.getBooks() throws com.is2.InvocationTargetException, com.is2.JAXBException
| getBooks | ()
```

Surgirá uma exceção:



Erro apresentado no ambiente de desenvolvimento:



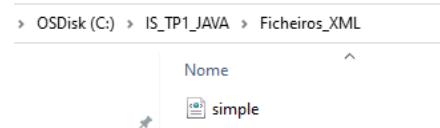
Resolução: Não consegue ler o ficheiro XML. Retificação:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

```
Start Page × Alunos.java × Simples_ws_soap.java × |  
Source Design History |                   
10 import javax.xml.bind.JAXBException;  
11 import javax.xml.bind.Unmarshaller;  
12  
13 // Definição do web-service SOAP  
14 @WebService(serviceName = "Simples_ws_soap")  
15 public class Simples_ws_soap {  
16  
17     // Definição do método  
18     @WebMethod(operationName = "getSimple")  
19     public ArrayList<Simple> getSimple() throws JAXBException, InvocationTargetException {  
20  
21         // Try-Catch do parsing via JAXB do ficheiro  
22         try {  
23             JAXBContext jaxbContext; // Parser usado (JAXB)  
24             File xmlFile = new File("C:\\\\IS_1920\\\\TP1\\\\simple.xml"); // Cria variável de ficheiro  
25             jaxbContext = JAXBContext.newInstance(Simples.class); // Cria instance do parse do J  
26             Unmarshaller jaxbUnmarshaller = jaxbContext.createUnmarshaller(); // Cria interface  
27             Simples simples = (Simples) jaxbUnmarshaller.unmarshal(xmlFile); // Deserializa o  
28  
29             return simples.getSimple();  
30  
31         } catch (JAXBException e) {  
32         }  
33  
34         return null;  
35     }  
36 }  
37  
38  
39
```

Criar uma pasta e copiar o ficheiro simple.xml para uma subpasta:



Retificar o no código:

```
// Definição do método
@WebMethod(operationName = "getSimples")
public ArrayList<Simple> getSimple() throws JAXBException, InvocationTargetException {
    // Try-Catch do parsing via JAXB do ficheiro
    try {
        JAXBContext jaxbContext; // Parser usado (JAXB)
        File xmlFile = new File("C:\\VS_TP1_JAVA\\Ficheiros_XML\\simple.xml"); // Cria variável de ficheiro local
        jaxbContext = JAXBContext.newInstance(Simples.class); // Cria instância do parse do JAXB na classe da lista
        Unmarshaller jaxbUnmarshaller = jaxbContext.createUnmarshaller(); // Cria interface responsável para "des-serializar"
        Simples simples = (Simples) jaxbUnmarshaller.unmarshal(xmlFile); // Des-serializa o ficheiro XML para o tipo
        return simples.getSimple();
    } catch (JAXBException e) {
    }
    return null;
}
```

Fazer RUN:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

The screenshot shows an IDE interface with two main panes. The top pane displays Java code for a service named 'Simples_ws_soap'. The code includes annotations like @WebMethod and @XmlRootElement, and uses JAXBContext to parse XML files. The bottom pane shows the 'Output' window for the 'GlassFish Server' run configuration. It displays the deployment process: starting the server, incrementally deploying the application, and finally a successful deployment message: 'BUILD SUCCESSFUL (total time: 32 seconds)'. A red arrow points from the deployment log back to the code in the top pane.

```

16
17     // Definição do método
18     @WebMethod(operationName = "getSimples")
19     public ArrayList<Simple> getSimples() throws JAXBException, InvocationTargetException {
20
21         // Try-Catch do parsing via JAXB do ficheiro
22         try {
23             JAXBContext jaxbContext; // Parser usado (JAXB)
24             File xmlFile = new File("C:\\IS_TP1_JAVA\\Ficheiros_XML\\simple.xml"); // Cria variável de ficheiro
25             jaxbContext = JAXBContext.newInstance(Simples.class); // Cria instance do parse do JAXB na classe
26             Unmarshaller jaxbUnmarshaller = jaxbContext.createUnmarshaller(); // Cria interface responsável para
27             Simples simples = (Simples) jaxbUnmarshaller.unmarshal(xmlFile); // Des-serializa o ficheiro XML para
28             return simples.getsimples();
29
30         } catch (JAXBException e) {
31
32         }
33
34         return null;
35     }
    
```

E os logs do servidor GlassFish:

The screenshot shows a browser window with the URL 'localhost:8080/IS_TP1_JAVA/Simples_ws_soap'. The page displays information for three endpoints:

Endpoint	Information
Service Name: {http://is2.com/}Books_ws_soap Port Name: {http://is2.com/}Books_ws_soapPort	Address: http://localhost:8080/IS_TP1_JAVA/Books_ws_soap WSDL: http://localhost:8080/IS_TP1_JAVA/Books_ws_soap?wsdl Implementation class: com.is2.Books_ws_soap
Service Name: {http://is2.com/}Simples_ws_soap Port Name: {http://is2.com/}Simples_ws_soapPort	Address: http://localhost:8080/IS_TP1_JAVA/Simples_ws_soap WSDL: http://localhost:8080/IS_TP1_JAVA/Simples_ws_soap?wsdl Implementation class: com.is2.Simples_ws_soap
Service Name: {http://db.is.com/}Alunos_ws_soap Port Name: {http://db.is.com/}Alunos_ws_soapPort	Address: http://localhost:8080/IS_TP1_JAVA/Alunos_ws_soap WSDL: http://localhost:8080/IS_TP1_JAVA/Alunos_ws_soap?wsdl Implementation class: com.db.Alunos_ws_soap

Web Services

Endpoint	Information
Service Name: {http://is2.com/}Books_ws_soap Port Name: {http://is2.com/}Books_ws_soapPort	Address: http://localhost:8080/IS_TP1_JAVA/Books_ws_soap WSDL: http://localhost:8080/IS_TP1_JAVA/Books_ws_soap?wsdl Implementation class: com.is2.Books_ws_soap
Service Name: {http://is2.com/}Simples_ws_soap Port Name: {http://is2.com/}Simples_ws_soapPort	Address: http://localhost:8080/IS_TP1_JAVA/Simples_ws_soap WSDL: http://localhost:8080/IS_TP1_JAVA/Simples_ws_soap?wsdl Implementation class: com.is2.Simples_ws_soap
Service Name: {http://db.is.com/}Alunos_ws_soap Port Name: {http://db.is.com/}Alunos_ws_soapPort	Address: http://localhost:8080/IS_TP1_JAVA/Alunos_ws_soap WSDL: http://localhost:8080/IS_TP1_JAVA/Alunos_ws_soap?wsdl Implementation class: com.db.Alunos_ws_soap

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

Ao executar o pedido no browser:

```

← → C ⓘ localhost:8080/IS_TP1_JAVA/Simples_ws_soap
Para acesso rápido, coloque os seus marcadores aqui, na barra de marcadores: Importar marcadores agora.

getSimple Method invocation

Method parameter(s)
Type Value

Method returned
java.util.List: "[com.is2.Simple@4cff5f78, com.is2.Simple@6934b989, com.is2.Simple@3c409f6e, com.is2.Simple@4dffd2fb, com.is2.Simple@7bf77fa4]"

SOAP Request
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<S:Header xmlns:n2="http://is2.com/">
<n2:getSimple/>
</S:Header>
</S:Envelope>

SOAP Response
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<S:Header xmlns:n2="http://is2.com/">
<n2:getSimpleResponse>
<return>
<name>Belgian Waffles</name>
<price>8.5</price>
<description>Two of our famous Belgian Waffles with plenty of real maple syrup</description>
<calories>650</calories>
</return>
<return>
<name>Strawberry Belgian Waffles</name>
<price>8.5</price>
<description>Light Belgian waffles covered with strawberries and whipped cream</description>
</return>
</n2:getSimpleResponse>
</S:Header>
</S:Envelope>

```

Analise a informação transferida entre o cliente e o servidor:

← → C ⓘ localhost:8080/IS_TP1_JAVA/Simples_ws_soap

Para acesso rápido, coloque os seus marcadores aqui, na barra de marcadores: Importar marcadores agora.

getSimple Method invocation

Method parameter(s)

Type Value

Method returned

java.util.List: "[com.is2.Simple@5e9de672, com.is2.Simple@2060e7ed, com.is2.Simple@49d7fe05, com.is2.Simple@1222ec14, com.is2.Simple@23bd3654]"

SOAP Request

```

<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<S:Header xmlns:n2="http://is2.com/">
<n2:getSimple/>
</S:Header>
</S:Envelope>

```

SOAP Response

```

<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<S:Header xmlns:n2="http://is2.com/">
<n2:getSimpleResponse>
<return>
<name>Belgian Waffles</name>
<price>8.5</price>
<description>Two of our famous Belgian Waffles with plenty of real maple syrup</description>
<calories>650</calories>
</return>
<return>
<name>Strawberry Belgian Waffles</name>
<price>8.5</price>
<description>Light Belgian waffles covered with strawberries and whipped cream</description>
</return>
</n2:getSimpleResponse>
</S:Header>
</S:Envelope>

```

← → C ⓘ localhost:8080/IS_TP1_JAVA/Simples_ws_soap

Para acesso rápido, coloque os seus marcadores aqui, na barra de marcadores: Importar marcadores agora.

getSimple Method invocation

Method parameter(s)

Type Value

Method returned

java.util.List: "[com.is2.Simple@5e9de672, com.is2.Simple@2060e7ed, com.is2.Simple@49d7fe05, com.is2.Simple@1222ec14, com.is2.Simple@23bd3654]"

SOAP Request

```

<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<S:Header xmlns:n2="http://is2.com/">
<n2:getSimple/>
</S:Header>
</S:Envelope>

```

SOAP Response

```

<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<S:Header xmlns:n2="http://is2.com/">
<n2:getSimpleResponse>
<return>
<name>Belgian Waffles</name>
<price>8.5</price>
<description>Two of our famous Belgian Waffles with plenty of real maple syrup</description>
<calories>650</calories>
</return>
<return>
<name>Strawberry Belgian Waffles</name>
<price>8.5</price>
<description>Light Belgian waffles covered with strawberries and whipped cream</description>
</return>
</n2:getSimpleResponse>
</S:Header>
</S:Envelope>

```

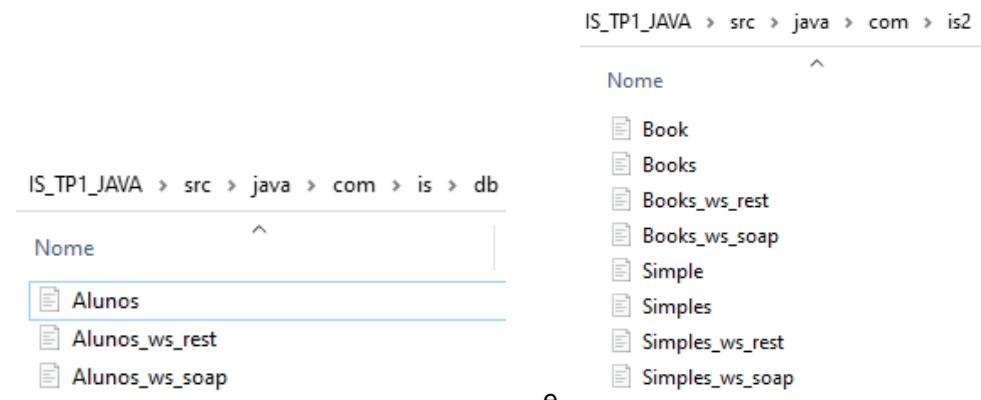
FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

Os ficheiros manipulados através dos webservices serão os seguintes três:

simple.xml	books.xml
<pre>C:\> IS_1920 > 07_JAVA_com_XML > 1-DOM > simple.xml 1 <?xml version="1.0" encoding="ISO-8859-1"?> 2 <!-- Edited by XMLSpy --> 3 <breakfast_menu> 4 <food> 5 <name>Belgian Waffles</name> 6 <price>\$5.95</price> 7 <description>Two of our famous Belgian Waffles with plenty of real maple syrup</description> 8 <calories>650</calories> 9 </food> 10 <food> 11 <name>Strawberry Belgian Waffles</name> 12 <price>\$7.95</price> 13 <description>light Belgian waffles covered with strawberries and whipped cream</description> 14 <calories>900</calories> 15 </food> 16 <food> 17 <name>Berry-Berry Belgian Waffles</name> 18 <price>\$8.95</price> 19 <description>light Belgian waffles covered with an assortment of fresh berries and whipped cream</description> 20 <calories>900</calories> 21 </food> 22 <food> 23 <name>French Toast</name> 24 <price>\$4.50</price> 25 <description>thick slices made from our homemade sourdough bread</description> 26 <calories>600</calories> 27 </food> 28 <food> 29 <name>Homestyle Breakfast</name> 30 <price>\$6.95</price> 31 <description>two eggs, bacon or sausage, toast, and our ever-popular hash browns</description> 32 <calories>950</calories> 33 </food> 34 </breakfast_menu></pre>	<pre>C:\> IS_1920 > 07_JAVA_com_XML > 1-DOM > books.xml 1 <?xml version="1.0"?> 2 <catalog> 3 <book id="bk101"> 4 <author>Gambardella, Matthew</author> 5 <title>XML Developer's Guide</title> 6 <genre>Computer</genre> 7 <price>44.95</price> 8 <publish_date>2000-10-01</publish_date> 9 <description>An in-depth look at creating applications with XML.</description> 10 </book> 11 <book id="bk102"> 12 <author>Ralls, Kim</author> 13 <title>Midnight Rain</title> 14 <genre>Fantasy</genre> 15 <price>5.95</price> 16 <publish_date>2000-12-16</publish_date> 17 <description>A former architect battles corporate zombies, an evil sorceress, and her own childhood to become queen of the world.</description> 18 </book> 19 <book id="bk103"> 20 <author>Corets, Eva</author> 21 <title>Maeve Ascendant</title> 22 <genre>Fantasy</genre> 23 <price>5.95</price> 24 <publish_date>2000-11-17</publish_date> 25 <description>After the collapse of a nanotechnology society in England, the young survivors lay the foundation for a new society.</description> 26 </book> 27 <book id="bk104"> 28 <author>Corets, Eva</author> 29 <title>Oberon's Legacy</title> 30 <genre>Fantasy</genre> 31 <price>5.95</price> 32 <publish_date>2001-03-10</publish_date> 33 <description>In post-apocalypse England, the mysterious agent known only as Oberon helps to create a new life for the inhabitants of London. Sequel to Maeve Ascendant.</description> 34 </book> 35 <book id="bk105"> 36 <author>Corets, Eva</author> 37 <title>The Sundered Grail</title> 38 <genre>Fantasy</genre> 39 <price>5.95</price> 40 <publish_date>2001-09-10</publish_date> 41 <description>The two daughters of Maeve, half-sisters,</pre>
<pre><?xml version="1.0" encoding="UTF-8"?> - <alunos> - <aluno> <id>1</id> <nome>Antonio Silva</nome> </aluno> - <aluno> <id>2</id> <nome>Maria Albertina</nome> </aluno> - <aluno> <id>3</id> <nome>Jose Ferreira</nome> </aluno> - <aluno> <id>4</id> <nome>Pedro Costa</nome> </aluno> </alunos></pre>	

Em termos de estrutura dos ficheiros com o código fonte, os mesmos são disponibilizados nas diretórias “is” e “is2”, em que na primeira apresenta a classe “Alunos.java” e os Webservices em REST e em SOAP respetivamente:



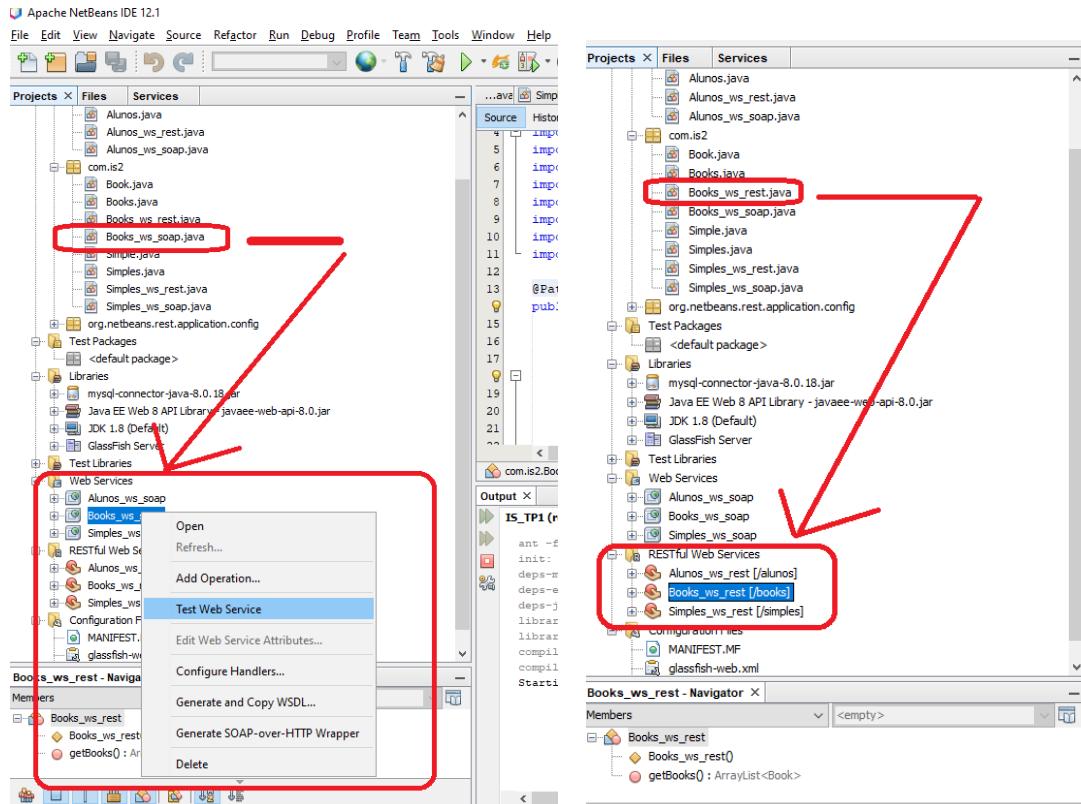
NOTA: As configurações de acesso à base de dados estão definidas no ficheiro “IS_TP1_JAVA\src\java\com\is\db\Alunos_ws_rest.java”:

```

Alunos_ws_soap.java
1 package com.is2;
2
3 // Imports
4 import java.sql.Connection;
5 import java.sql.DriverManager;
6 import java.sql.ResultSet;
7 import java.sql.SQLException;
8 import java.sql.Statement;
9 import java.util.ArrayList;
10 import javax.jws.WebService;
11 import javax.jws.WebMethod;
12
13 // Definição do web-service SOAP
14 @WebService(serviceName = "Alunos_ws_soap")
15 public class Alunos_ws_soap {
16
17     // Definição do método
18     @WebMethod(operationName = "getAlunos")
19     public ArrayList<Aluno> getAlunos() {
20         ArrayList<Aluno> alunos = new ArrayList<>();
21
22         // Try-Catch de ligação à BD e query da tabela alunos
23         try{
24             Class.forName("com.mysql.jdbc.Driver");
25             String url = "jdbc:mysql://localhost:3306/ia7useUnicode=true&useJDBCCompliantTimezoneShift=true&useLegacyDatetimeCode=false&serverTimezone=UTC";
26             Connection conn = DriverManager.getConnection(url, "userdb", "teste"); // Conexão com credenciais de ligação do MySQL
27             Statement st = conn.createStatement();
28             ResultSet rs = st.executeQuery("SELECT * FROM alunos");
29
30             while(rs.next()){
31                 Aluno aluno = new Aluno();
32                 aluno.setId(rs.getInt("id"));
33                 aluno.setNome(rs.getString("nome"));
34                 aluno.setNota(rs.getDouble("nota"));
35                 alunos.add(aluno);
36             }
37         } catch (Exception e) {
38             e.printStackTrace();
39         }
40
41         return alunos;
42     }
43
44 }

```

Neste projeto, a implementação dos WebServices são realizados em links diferentes:



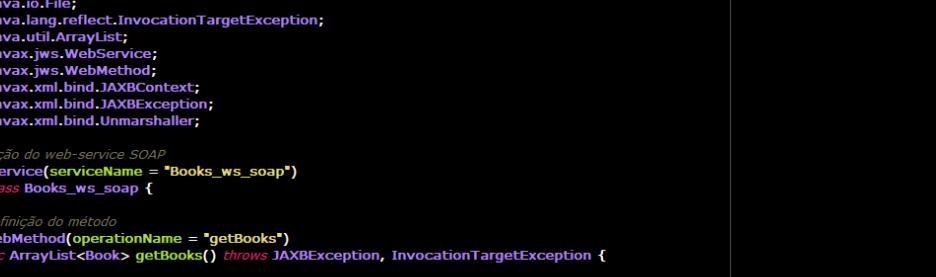
Em SOAP: http://localhost:8080/IS_TP1_JAVA/Books_ws_soap?Tester

Em REST: http://localhost:8080/IS_TP1_JAVA/webresources/books

2.2.1 Importação do ficheiro books.xml, via SOAP e Parser: JAXB

Para o carreamento da informação do ficheiro “books.xml” foram criadas as classes Book.java e Books.java

O Webservice usando SOAP e a abordagem JAXB é a seguinte:



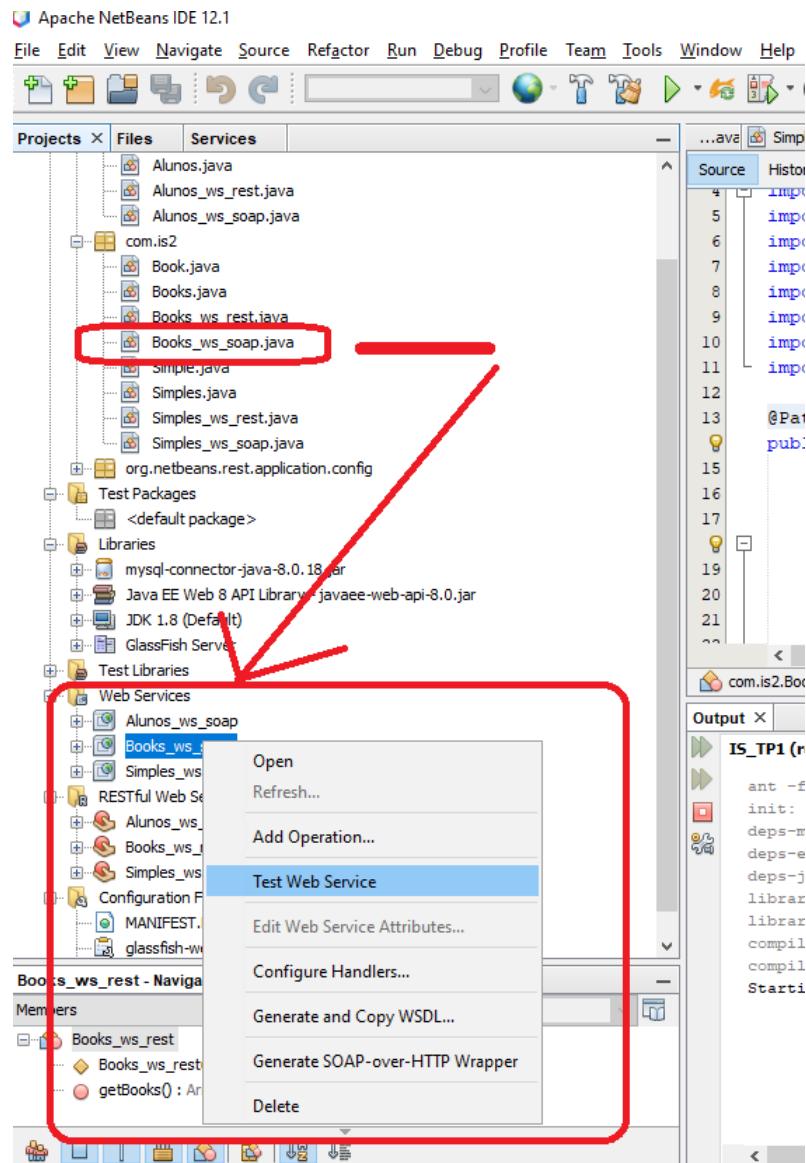
```
Books_ws_soap.java X |  
Source Design History   
1 package com.is2;  
2  
3 // Imports  
4 import java.io.File;  
5 import java.lang.reflect.InvocationTargetException;  
6 import java.util.ArrayList;  
7 import javax.jws.WebService;  
8 import javax.jws.WebMethod;  
9 import javax.xml.bind.JAXBContext;  
10 import javax.xml.bind.JAXBException;  
11 import javax.xml.bind.Unmarshaller;  
12  
13 // Definição do web-service SOAP  
14 @WebService(serviceName = "Books_ws_soap")  
15 public class Books_ws_soap {  
16  
17     // Definição do método  
18     @WebMethod(operationName = "getBooks")  
19     public ArrayList<Book> getBooks() throws JAXBException, InvocationTargetException {  
20  
21         // Try-Catch do parsing via JAXB do ficheiro  
22         try {  
23             JAXBContext jaxbContext; // Parser usado (JAXB)  
24             File xmlFile = new File("C:\\IS_1920\\TP1\\books.xml"); // Cria variável de ficheiro local indicando o path do ficheiro "books.xml"  
25             jaxbContext = JAXBContext.newInstance(Books.class); // Cria instance do parse do JAXB na classe da lista de livros  
26             Unmarshaller jaxbUnmarshaller = jaxbContext.createUnmarshaller(); // Cria interface responsável para "des-serializar" o ficheiro XML para objeto Java  
27             Books books = (Books) jaxbUnmarshaller.unmarshal(xmlFile); // Des-serializa o ficheiro XML para objeto Java  
28  
29             return books.getBooks();  
30         } catch (JAXBException e){  
31             }  
32         }  
33  
34         return null;  
35     }  
36 }  
37
```

O Teste do webservice é o seguinte:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

A partir do NetBeans, execute o webservice:



FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação



SOAP Request

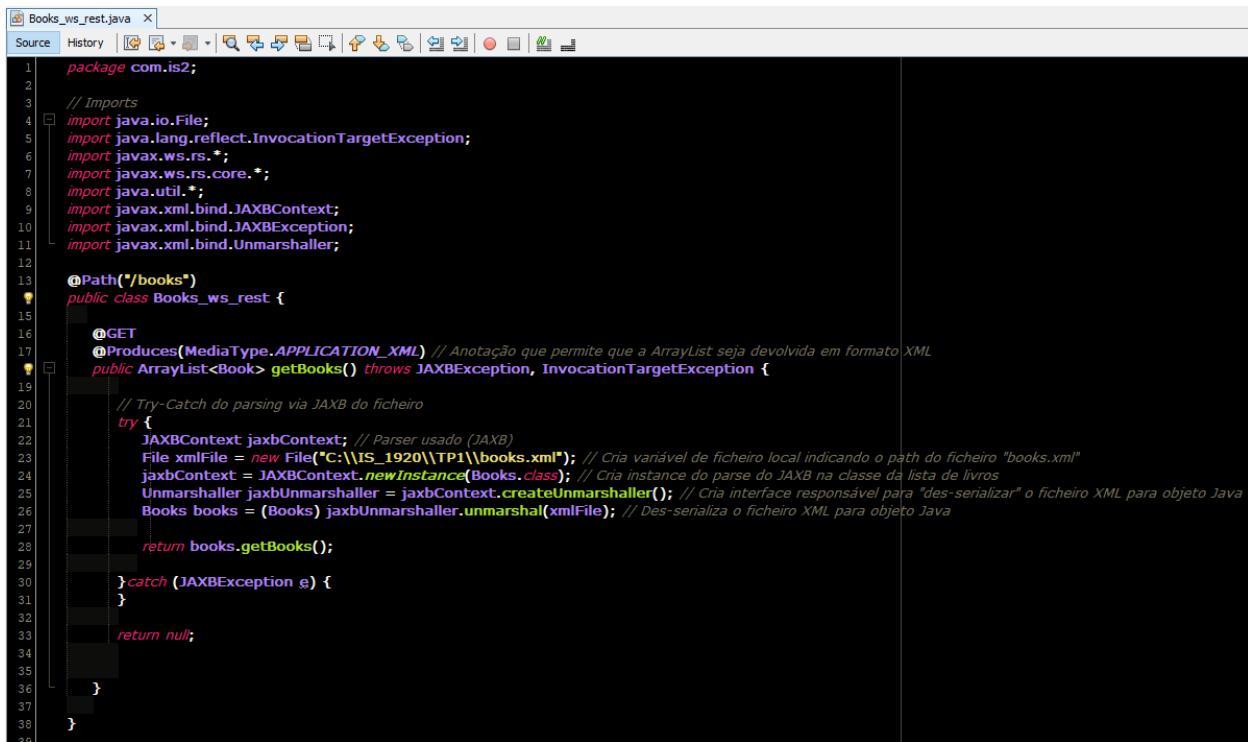
```
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="ht  
<SOAP-ENV:Header/>  
<S:Body>  
    <ns2:getBooks xmlns:ns2="http://is2.com/" />  
</S:Body>  
</S:Envelope>
```

SOAP Response

```
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="ht  
<SOAP-ENV:Header/>  
<S:Body>  
    <ns2:getBooksResponse xmlns:ns2="http://is2.com/">  
        <return>  
            <author>Gambardella, Matthew</author>  
            <title>XML Developer's Guide</title>  
            <genre>Computer</genre>  
            <price>44.95</price>  
            <publish_date>2000-10-01</publish_date>  
            <description>An in-depth look at creating applications  
with XML.</description>  
        </return>  
        <return>  
            <author>Ralls, Kim</author>  
            <title>Midnight Rain</title>  
            <genre>Fantasy</genre>  
            <price>5.95</price>  
            <publish_date>2000-12-16</publish_date>  
            <description>A former architect battles corporate zombies,  
an evil sorceress, and her own childhood to become queen  
of the world.</description>  
        </return>  
    </return>
```

2.2.2 Importação do ficheiro books.xml, via REST e Parser: JAXB

O Webservice usando REST e a abordagem JAXB é a seguinte:

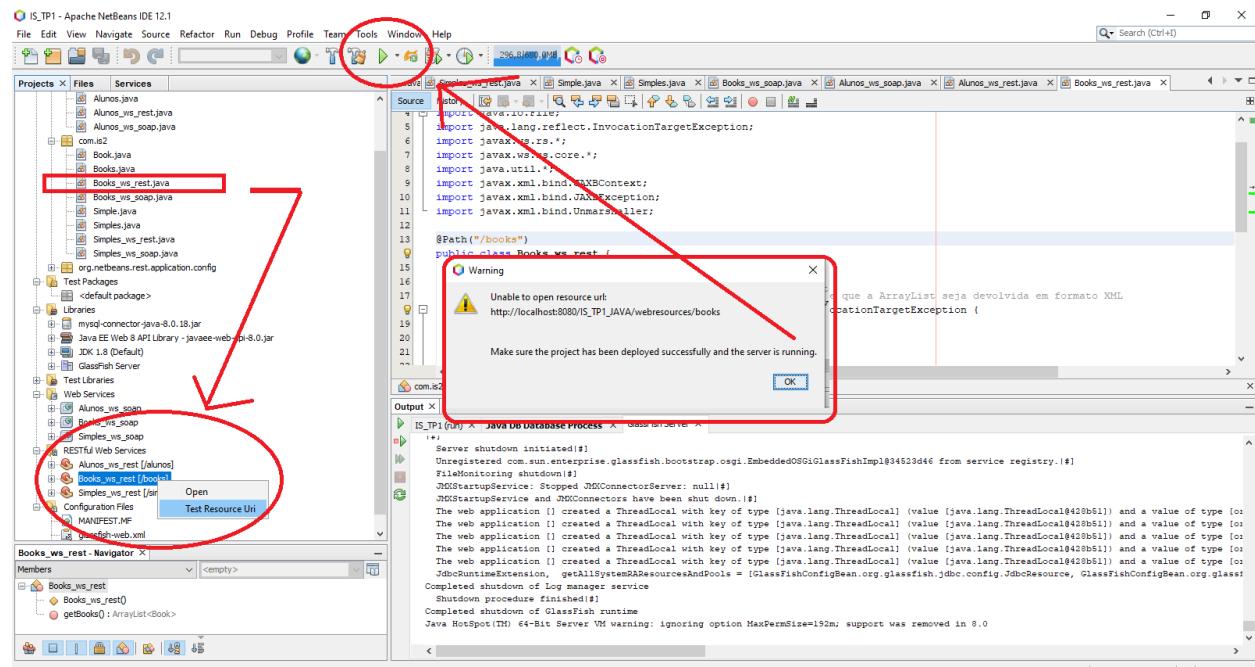


```

1 package com.is2;
2
3 // Imports
4 import java.io.File;
5 import java.lang.reflect.InvocationTargetException;
6 import javax.ws.rs.*;
7 import javax.ws.rs.core.*;
8 import java.util.*;
9 import javax.xml.bind.JAXBContext;
10 import javax.xml.bind.JAXBException;
11 import javax.xml.bind.Unmarshaller;
12
13 @Path("/books")
14 public class Books_ws_rest {
15
16     @GET
17     @Produces(MediaType.APPLICATION_XML) // Anotação que permite que a ArrayList seja devolvida em formato XML
18     public ArrayList<Book> getBooks() throws JAXBException, InvocationTargetException {
19
20         // Try-Catch do parsing via JAXB do ficheiro
21         try {
22             JAXBContext jaxbContext; // Parser usado (JAXB)
23             File xmlFile = new File("C:\\IS_1920\\TP1\\books.xml"); // Cria variável de ficheiro local indicando o path do ficheiro "books.xml"
24             jaxbContext = JAXBContext.newInstance(Books.class); // Cria instance do parse do JAXB na classe da lista de livros
25             Unmarshaller jaxbUnmarshaller = jaxbContext.createUnmarshaller(); // Cria interface responsável para "des-serializar" o ficheiro XML para objeto Java
26             Books books = (Books) jaxbUnmarshaller.unmarshal(xmlFile); // Des-serializa o ficheiro XML para objeto Java
27
28             return books.getBooks();
29
30         } catch (JAXBException e) {
31
32             return null;
33
34         }
35     }
36 }
37
38 }
39

```

Para testar o Webservice em REST:



O Teste do webservice é o seguinte: http://localhost:8080/IS_TP1_JAVA/webresources/books

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

← → ⌂ ⓘ localhost:8080/IS_TP1_JAVA/webresources/books

Para acesso rápido, coloque os seus marcadores aqui, na barra de marcadores. [Importar marcadores agora...](#)

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<books>
  <book>
    <author>Gambardella, Matthew</author>
    <title>XML Developer's Guide</title>
    <genre>Computer</genre>
    <price>44.95</price>
    <publish_date>2000-10-01</publish_date>
    <description>An in-depth look at creating applications with XML.</description>
  </book>
  <book>
    <author>Ralls, Kim</author>
    <title>Midnight Rain</title>
    <genre>Fantasy</genre>
    <price>5.95</price>
    <publish_date>2000-12-16</publish_date>
    <description>A former architect battles corporate zombies, an evil sorceress, and her own childhood to become queen of the world.</description>
  </book>
  <book>
    <author>Corets, Eva</author>
    <title>Maeve Ascendant</title>
    <genre>Fantasy</genre>
    <price>5.95</price>
    <publish_date>2000-11-17</publish_date>
    <description>After the collapse of a nanotechnology society in England, the young survivors lay the foundation for a new society.</description>
  </book>
  <book>
    <author>Corets, Eva</author>
    <title>Oberon's Legacy</title>
    <genre>Fantasy</genre>
    <price>5.95</price>
    <publish_date>2001-03-10</publish_date>
    <description>In post-apocalypse England, the mysterious agent known only as Oberon helps to create a new life for the inhabitants of London. Sequel to Maeve Ascendant.</description>
  </book>
  <book>
    <author>Corets, Eva</author>
    <title>The Sundered Grail</title>
    <genre>Fantasy</genre>
    <price>5.95</price>
    <publish_date>2001-09-10</publish_date>
    <description>The two daughters of Maeve, half-sisters, battle one another for control of England. Sequel to Oberon's Legacy.</description>
  </book>
  <book>
    <author>Randall, Cynthia</author>
    <title>Lover Birds</title>
    <genre>Romance</genre>
    <price>4.95</price>
    <publish_date>2000-09-02</publish_date>
  </book>
```

↓ Custom Request Headers

Status: 200 (OK)

Response:

Tabular View	Raw View	Sub-Resource	Headers	Http Monitor
<?xml version="1.0" encoding="UTF-8"?> <books> <book> <author>Gambardella, Matthew</author> <title>XML Developer's Guide</title> <genre>Computer</genre> <price>44.95</price> <publish_date>2000-10-01</publish_date> <description>An in-depth look at creating applications with XML.</description> </book> <book> <author>Ralls, Kim</author> <title>Midnight Rain</title> <genre>Fantasy</genre> <price>5.95</price> <publish_date>2000-12-16</publish_date> <description>A former architect battles corporate zombies, an evil sorceress, and her own childhood to become queen of the world.</description> </book> <book> <author>Corets, Eva</author> <title>Maeve Ascendant</title> <genre>Fantasy</genre> <price>5.95</price> <publish_date>2000-11-17</publish_date> <description>After the collapse of a nanotechnology society in England, the young survivors lay the foundation for a new society.</description> </book> <book> <author>Corets, Eva</author> <title>Oberon's Legacy</title> <genre>Fantasy</genre> <price>5.95</price> <publish_date>2001-03-10</publish_date> <description>In post-apocalypse England, the mysterious agent known only as Oberon helps to create a new life for the inhabitants of London. Sequel to Maeve Ascendant.</description> </book> <book> <author>Corets, Eva</author> <title>The Sundered Grail</title> <genre>Fantasy</genre> <price>5.95</price> <publish_date>2001-09-10</publish_date> <description>The two daughters of Maeve, half-sisters, battle one another for control of England. Sequel to Oberon's Legacy.</description> </book> <book> <author>Randall, Cynthia</author> <title>Lover Birds</title> <genre>Romance</genre> <price>4.95</price> <publish_date>2000-09-02</publish_date> </book>				

2.2.3 Importação do ficheiro simple.xml, via SOAP e Parser: JAXB

Para o carreamento da informação do ficheiro "simple.xml" foram criadas as classes Simple.java e Simples.java

```

Simple.java X
Source History
1 package com.is2;
2
3 // Imports
4 import java.io.Serializable;
5 import javax.xml.bind.annotation.XmlAccessType;
6 import javax.xml.bind.annotation.XmlAccessorType;
7 import javax.xml.bind.annotation.XmlRootElement;
8
9 @XmlRootElement(name = "food") // Tag raiz para cada elemento
10 @XmlAccessorType(XmlAccessType.FIELD)
11 public class Simple implements Serializable {
12     // Atributos para cada tag / elemento
13     private String name;
14     private float price;
15     private String description;
16     private int calories;
17
18     // Getters & Setters
19     public String getName() {...3 lines}
20
21     public void setName(String name) {...3 lines}
22
23     public float getPrice() {...3 lines}
24
25     public void setPrice(float price) {...3 lines}
26
27     public String getDescription() {...3 lines}
28
29     public void setDescription(String description) {...3 lines}
30
31     public int getCalories() {...3 lines}
32
33     public void setCalories(int calories) {...3 lines}
34
35 }
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51

```



```

Simples.java X
Source History
1 package com.is2;
2
3 // Imports
4 import java.util.ArrayList;
5 import javax.xml.bind.annotation.XmlAccessType;
6 import javax.xml.bind.annotation.XmlAccessorType;
7 import javax.xml.bind.annotation.XmlElement;
8 import javax.xml.bind.annotation.XmlRootElement;
9
10 @XmlRootElement(name = "breakfast_menu") // Tag raiz da lista de foods
11 @XmlAccessorType(XmlAccessType.FIELD)
12 public class Simples {
13
14     @XmlElement(name = "food")
15     private ArrayList<Simple> simples = null; // Lista de elementos de tipo food
16
17     // Getters & Setters
18     public ArrayList<Simple> getSimples() {...3 lines}
19
20     public void setSimples(ArrayList<Simple> simples) {...3 lines}
21
22
23
24
25
26
27

```

O webservice usando SOAP e a abordagem JAXB é a seguinte:

```

Simples_ws_soap.java X
Source Design History
1 package com.is2;
2
3 // Imports
4 import java.io.File;
5 import java.lang.reflect.InvocationTargetException;
6 import java.util.ArrayList;
7 import javax.jws.WebService;
8 import javax.jws.WebMethod;
9 import javax.xml.bind.JAXBContext;
10 import javax.xml.bind.JAXBException;
11 import javax.xml.bind.Unmarshaller;
12
13 // Definição do web-service SOAP
14 @WebService(serviceName = "Simples_ws_soap")
15 public class Simples_ws_soap {
16
17     // Definição do método
18     @WebMethod(operationName = "getSimples")
19     public ArrayList<Simple> getSimples() throws JAXBException, InvocationTargetException {
20
21         // Try-Catch do parsing via JAXB do ficheiro
22         try {
23             JAXBContext jaxbContext; // Parser usado (JAXB)
24             File xmlFile = new File("C:\IS_1920\TP1\simple.xml"); // Cria variável de ficheiro local indicando o path do ficheiro "simple.xml"
25             jaxbContext = JAXBContext.newInstance(Simples.class); // Cria instância do parse do JAXB na classe da lista de foods
26             Unmarshaller jaxbUnmarshaller = jaxbContext.createUnmarshaller(); // Cria interface responsável para "des-serializar" o ficheiro XML para objeto Java
27             Simples simples = (Simples) jaxbUnmarshaller.unmarshal(xmlFile); // Des-serializa o ficheiro XML para objeto Java
28
29             return simples.getSimple();
30
31         } catch (JAXBException e){
32         }
33
34         return null;
35     }
36
37 }
38

```

O teste do Webservice é o seguinte:

```

<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlso
<SOAP-ENV:Header/>
<S:Body>
<ns2:getSimple xmlns:ns2="http://is2.com/">
</S:Body>
</S:Envelope>
```

SOAP Response

```

<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlso
<SOAP-ENV:Header/>
<S:Body>
<ns2:getSimpleResponse xmlns:ns2="http://is2.com/">
<return>
<name>Belgian Waffles</name>
<price>0.0</price>
<description>Two of our famous Belgian Waffles with plenty of real maple syrup</description>
<calories>650</calories>
</return>
<return>
<name>Strawberry Belgian Waffles</name>
<price>0.0</price>
<description>Light Belgian waffles covered with strawberries and whipped cream</description>
<calories>900</calories>
</return>
<return>
<name>Berry-Berry Belgian Waffles</name>
<price>0.0</price>
<description>Light Belgian waffles covered with an assortment of fresh berries and whipped cream</description>
</return>
</return>
```

2.2.4 Importação do ficheiro simple.xml, via REST e Parser: JAXB

A definição do webservice em REST é o seguinte:

```

import java.io.File;
import java.lang.reflect.InvocationTargetException;
import javax.ws.rs.*;
import javax.ws.rs.core.*;
import java.util.*;
import javax.xml.bind.JAXBContext;
import javax.xml.bind.JAXBException;
import javax.xml.bind.Unmarshaller;

@Path("/simples")
public class Simples_ws_rest {

    @GET
    @Produces(MediaType.APPLICATION_XML) // Anotação que permite que a ArrayList seja devolvida em formato XML
    public ArrayList<Simple> getSimple() throws JAXBException, InvocationTargetException {
        // Try-Catch do parsing via JAXB do ficheiro
        try {
            JAXBContext jaxbContext; // Parser usado (JAXB)
            File xmlFile = new File("C:\\US_1920\\TP1\\simple.xml"); // Cria variável do ficheiro local indicando o path do ficheiro "simple.xml"
            jaxbContext = JAXBContext.newInstance(Simples.class); // Cria instance do parse do JAXB na classe da lista de livros
            Unmarshaller jaxbUnmarshaller = jaxbContext.createUnmarshaller(); // Cria interface responsável para "des-serializar" o ficheiro XML para objeto Java
            Simples simples = (Simples) jaxbUnmarshaller.unmarshal(xmlFile); // Des-serializa o ficheiro XML para objeto Java
            return simples.getSimple();
        } catch (JAXBException e) {
        }
        return null;
    }
}
```

O teste do webservice é o seguinte:

[↓ Custom Request Headers](#)**Status:** 200 (OK)**Response:**

Tabular View

Raw View

Sub-Resource

Headers

Http Monitor

```
<?xml version="1.0" encoding="UTF-8"?>
<simples>
<food>
    <name>Belgian Waffles</name>
    <price>0.0</price>
    <description>Two of our famous Belgian Waffles with plenty of real maple syrup</description>
    <calories>650</calories>
</food>
<food>
    <name>Strawberry Belgian Waffles</name>
    <price>0.0</price>
    <description>Light Belgian waffles covered with strawberries and whipped cream</description>
    <calories>900</calories>
</food>
<food>
    <name>Berry-Berry Belgian Waffles</name>
    <price>0.0</price>
    <description>Light Belgian waffles covered with an assortment of fresh berries and whipped cream</description>
    <calories>900</calories>
</food>
<food>
    <name>French Toast</name>
```

2.2.5 Importação do ficheiro alunos.xml, via SOAP e importação na Base de Dados

Considere-se o ficheiro alunos.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
- <alunos>
  - <aluno>
    <id>1</id>
    <nome>Antonio Silva</nome>
  </aluno>
  - <aluno>
    <id>2</id>
    <nome>Maria Albertina</nome>
  </aluno>
  - <aluno>
    <id>3</id>
    <nome>Jose Ferreira</nome>
  </aluno>
  - <aluno>
    <id>4</id>
    <nome>Pedro Costa</nome>
  </aluno>
</alunos>
```

Criou-se a classe a Classe Alunos.java:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

```

1 package com.is.db;
2
3 // Imports
4 import java.io.Serializable;
5 import javax.persistence.Column;
6 import javax.persistence.Entity;
7 import javax.persistence.GeneratedValue;
8 import javax.persistence.GenerationType;
9 import javax.persistence.Id;
10 import javax.persistence.NamedQueries;
11 import javax.persistence.NamedQuery;
12 import javax.persistence.Table;
13 import javax.validation.constraints.NotNull;
14 import javax.validation.constraints.Size;
15 import javax.xml.bind.annotation.XmlRootElement;
16
17
18 /**
19 * Entity
20 * @Table(name = "alunos")
21 * @XmlRootElement
22 * // Definição de métodos equiparados a queries
23 * @NamedQueries({
24 * @NamedQuery(name = "Alunos.findAll", query = "SELECT a FROM Alunos a"),
25 * @NamedQuery(name = "Alunos.findById", query = "SELECT a FROM Alunos a WHERE a.id = :id"),
26 * @NamedQuery(name = "Alunos.findByName", query = "SELECT a FROM Alunos a WHERE a.nome = :nome")})
27 public class Alunos implements Serializable {
28
29     // Atributos
30     private static final long serialVersionUID = 1L;
31     @Id
32     @GeneratedValue(strategy = GenerationType.IDENTITY)
33     @Basic(optional = false)
34     @Column(name = "id")
35     private Integer id;
36     @Basic(optional = false)
37     @NotNull
38     @Size(min = 1, max = 200)
39     @Column(name = "nome")
40     private String nome;
41

```

O webservice implementado em SOAP é o seguinte:

```

1 package com.is.db;
2
3 // Imports
4 import java.sql.Connection;
5 import java.sql.DriverManager;
6 import java.sql.ResultSet;
7 import java.sql.SQLException;
8 import java.sql.Statement;
9 import java.util.ArrayList;
10 import javax.jws.WebService;
11 import javax.jws.WebMethod;
12
13 // Definição do web-service SOAP
14 @WebService(serviceName = "Alunos_ws_soap")
15 public class Alunos_ws_soap {
16
17     // Definição do método
18     @WebMethod(operationName = "getAlunos")
19     public ArrayList<Alunos> getAlunos() {
20         ArrayList<Alunos> alunos = new ArrayList<>();
21
22         // Try-Catch de ligação à BD e query da tabela alunos
23         try{
24             Class.forName("com.mysql.jdbc.Driver");
25             String url = "jdbc:mysql://localhost:3306/is?useUnicode=true&useJDBCCompliantTimezoneShift=true&useLegacyDatetimeCode=false&serverTimezone=UTC";
26             Connection conn = DriverManager.getConnection(url, "root", "root"); // Conexão com credenciais de ligação do MySQL
27             Statement st = conn.createStatement();
28             ResultSet srs = st.executeQuery("SELECT * FROM alunos");
29
30             // Adição do resultado da query à resposta XML
31             while (srs.next()) {
32                 Alunos al = new Alunos();
33                 al.setId(srs.getInt("id"));
34                 al.setName(srs.getString("nome"));
35                 alunos.add(al);
36             }
37
38             return alunos;
39
40         }catch(ClassNotFoundException | SQLException e){
41             e.printStackTrace();
42         }
43
44         return null;
45     }
46

```

A execução do webservice é a seguinte:

SOAP Request

```
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
    <ns2:getAlunos xmlns:ns2="http://db.is.com/">
</S:Body>
</S:Envelope>
```

SOAP Response

```
<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<S:Body>
    <ns2:getAlunosResponse xmlns:ns2="http://db.is.com/">
        <return>
            <id>1</id>
            <nome> Antonio Silva </nome>
        </return>
        <return>
            <id>1</id>
            <nome> Maria Albertina </nome>
        </return>
        ...
    </ns2:getAlunosResponse>
</S:Body>
</S:Envelope>
```

2.2.6 Importação do ficheiro alunos.xml, via REST e importação na Base de Dados

```
package com.is.db;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import javax.ws.rs.*;
import javax.ws.rs.core.*;
import java.util.*;

@Path("/alunos")
public class Alunos_ws_rest {

    @GET
    @Produces(MediaType.APPLICATION_XML) // Anotação que permite que a ArrayList seja devolvida em formato XML
    public ArrayList<Aluno> getAlunos() {
        ArrayList<Aluno> alunos = new ArrayList<>();
        // Try-Catch de ligação à BD e query da tabela alunos
        try{
            Class.forName("com.mysql.jdbc.Driver");
            String url = "jdbc:mysql://localhost:3306/is?useUnicode=true&useJDBCCompliantTimezoneShift=true&useLegacyDatetimeCode=false&serverTimezone=UTC";
            Connection conn = DriverManager.getConnection(url, "root", "root"); // Conexão com credenciais de ligação do MySQL
            Statement st = conn.createStatement();
            ResultSet rs = st.executeQuery("SELECT * FROM alunos");

            // Adição do resultado da query à resposta XML
            while (rs.next()) {
                Aluno al = new Aluno();
                al.setId(rs.getInt("id"));
                al.setNome(rs.getString("nome"));
                alunos.add(al);
            }
        } catch (ClassNotFoundException | SQLException e){
        }
        return alunos;
    }

    @POST
    @Consumes(MediaType.APPLICATION_XML)
    public void postAluno(Aluno aluno) {
        // Implementar a lógica para inserir o novo aluno na base de dados
    }
}
```

A execução do webservice é a seguinte:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

[↓ Custom Request Headers](#)

Status: 200 (OK)

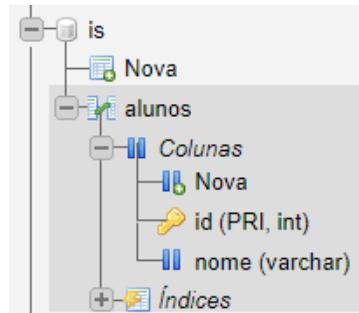
Response:

[Tabular View](#) [Raw View](#) [Sub-Resource](#) [Headers](#) [Http Monitor](#)

```
<?xml version="1.0" encoding="utf-8"?>
<alunos>
  <aluno>
    <id>1</id>
    <nome>Antonio Silva</nome>
  </aluno>
  <aluno>
    <id>2</id>
    <nome>Maria Albertina</nome>
  </aluno>
  <aluno>
    <id>3</id>
    <nome>Jose Ferreira</nome>
  </aluno>
  <aluno>
    <id>4</id>
    <nome>Pedro Costa</nome>
  </aluno>
</alunos>
```

2.3 Implementação em PHP

NOTA: No sentido de criar uma tabela na base de dados MySQL, crie uma base de dados com a designação “is” (diminutivo da designação da unidade curricular de Integração de sistemas)



#	Nome	Tipo	Agrupamento (Collation)	Atributos	Nulo	Predefinido	Comentários	Extra	Acções
<input type="checkbox"/>	1	id	int(11)		Não	Nenhum		Muda Elimina ▾ Mais	
<input type="checkbox"/>	2	nome	varchar(50)	utf8mb4_general_ci	Não	Nenhum		Muda Elimina ▾ Mais	

Os ficheiros manipulados através dos webservices serão os seguintes três:

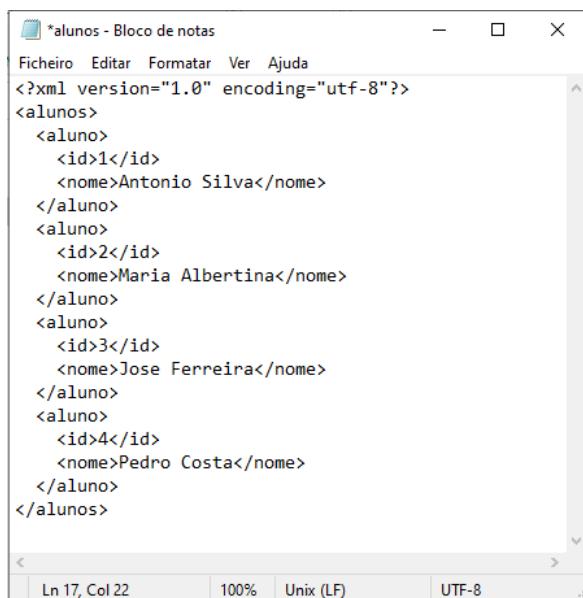
simple.xml	books.xml
<pre> <?xml version="1.0" encoding="ISO-8859-1"?> <!-- Edited by XMLSpy --> <breakfast_menu> <food> <name>Belgian Waffles</name> <price>\$5.95</price> <description>Two of our famous Belgian Waffles with plenty of real maple syrup</description> <calories>650</calories> </food> <food> <name>Strawberry Belgian Waffles</name> <price>\$7.95</price> <description>Light Belgian waffles covered with strawberries and whipped cream</description> <calories>900</calories> </food> <food> <name>Berry-Berry Belgian Waffles</name> <price>\$8.95</price> <description>Light Belgian waffles covered with an assortment of fresh berries and whipped cream</description> <calories>900</calories> </food> <food> <name>French Toast</name> <price>\$4.50</price> <description>Thick slices made from our homemade sourdough bread</description> <calories>600</calories> </food> <food> <name>Homestyle Breakfast</name> <price>\$6.95</price> <description>Two eggs, bacon or sausage, toast, and our ever-popular hash browns</description> <calories>950</calories> </food> </breakfast_menu> </pre>	<pre> <?xml version="1.0"?> <catalog> <book id="bk101"> <author>Gambardella, Matthew</author> <title>XML Developer's Guide</title> <genre>Computer</genre> <price>44.95</price> <publish_date>2000-10-01</publish_date> <description>An in-depth look at creating applications with XML.</description> </book> <book id="bk102"> <author>Ralls, Kim</author> <title>Midnight Rain</title> <genre>Fantasy</genre> <price>5.95</price> <publish_date>2000-12-16</publish_date> <description>A former architect battles corporate zombies, an evil sorceress, and her own childhood to become queen of the world.</description> </book> <book id="bk103"> <author>Corets, Eva</author> <title>Maeve Ascendant</title> <genre>Fantasy</genre> <price>5.95</price> <publish_date>2000-11-17</publish_date> <description>After the collapse of a nanotechnology society in England, the young survivors lay the foundation for a new society.</description> </book> <book id="bk104"> <author>Corets, Eva</author> <title>Oberon's Legacy</title> <genre>Fantasy</genre> <price>5.95</price> <publish_date>2001-03-10</publish_date> <description>In post-apocalypse England, the mysterious agent known only as Oberon helps to create a new life for the inhabitants of London. Sequel to Maeve Ascendant.</description> </book> <book id="bk105"> <author>Corets, Eva</author> <title>The Sundered Grail</title> <genre>Fantasy</genre> <price>5.95</price> <publish_date>2001-09-10</publish_date> <description>The two daughters of Maeve, half-sisters, </pre>
alunos.xml:	

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

```
<?xml version="1.0" encoding="UTF-8"?>
- <alunos>
  - <aluno>
    <id>1</id>
    <nome>Antonio Silva</nome>
  </aluno>
  - <aluno>
    <id>2</id>
    <nome>Maria Albertina</nome>
  </aluno>
  - <aluno>
    <id>3</id>
    <nome>Jose Ferreira</nome>
  </aluno>
  - <aluno>
    <id>4</id>
    <nome>Pedro Costa</nome>
  </aluno>
</alunos>
```

Para verificar se a importação do ficheiro XML está a ser importado corretamente, considere o ficheiro "alunos.xml"



The screenshot shows a Windows-style notepad window titled "alunos - Bloco de notas". The content of the file is an XML document with the following structure:

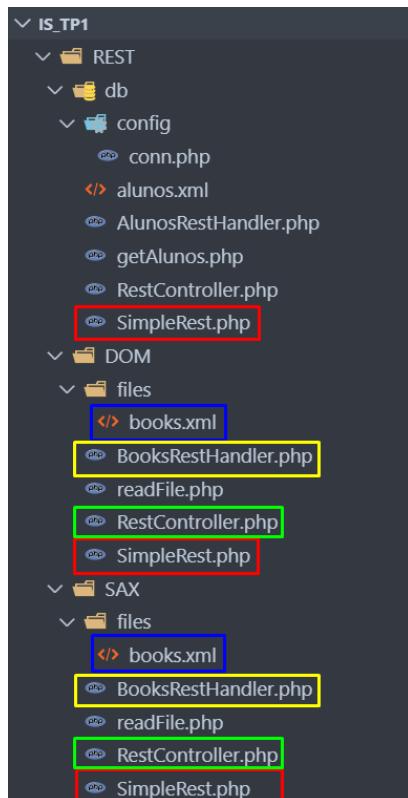
```
<?xml version="1.0" encoding="utf-8"?>
<alunos>
  <aluno>
    <id>1</id>
    <nome>Antonio Silva</nome>
  </aluno>
  <aluno>
    <id>2</id>
    <nome>Maria Albertina</nome>
  </aluno>
  <aluno>
    <id>3</id>
    <nome>Jose Ferreira</nome>
  </aluno>
  <aluno>
    <id>4</id>
    <nome>Pedro Costa</nome>
  </aluno>
</alunos>
```

The status bar at the bottom indicates "Ln 17, Col 22" and "100% Unix (LF) UTF-8".

```
<?xml version="1.0" encoding="UTF-8"?>
- <alunos>
  - <aluno>
    <id>1</id>
    <nome>Antonio Silva</nome>
  </aluno>
  - <aluno>
    <id>2</id>
    <nome>Maria Albertina</nome>
  </aluno>
  - <aluno>
    <id>3</id>
    <nome>Jose Ferreira</nome>
  </aluno>
  - <aluno>
    <id>4</id>
    <nome>Pedro Costa</nome>
  </aluno>
</alunos>
```

2.3.1 Execução de webservice via REST para leitura de dados de uma base de dados

Considere a diretoria “REST” em que os ficheiros com a mesma cor são iguais:



Considere também as configurações de acesso à base de dados:

```

<?php
$servername = "localhost:3306";
$username = "userdb";
$password = "teste";
$database = "is";

try {
    $PDO = new PDO("mysql:host=$servername;dbname=$database", $username, $password);

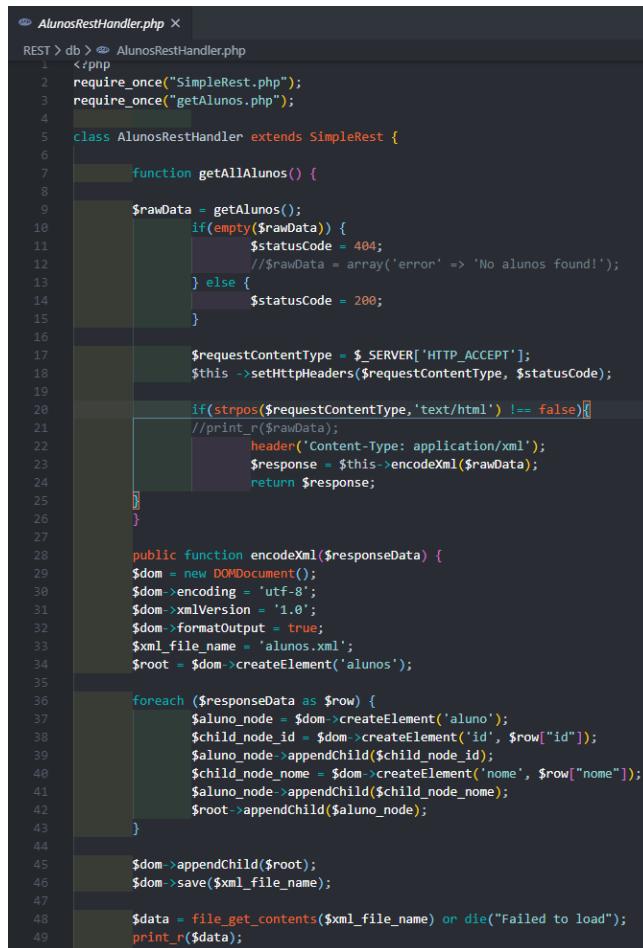
    $PDO->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $PDO->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE, PDO::FETCH_ASSOC);

} catch (PDOException $e) {
    var_dump($e);
}
?>
  
```

A implementação do Webservice poderá ser ilustrada através do ficheiro “AlunoRestHandler.php”:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

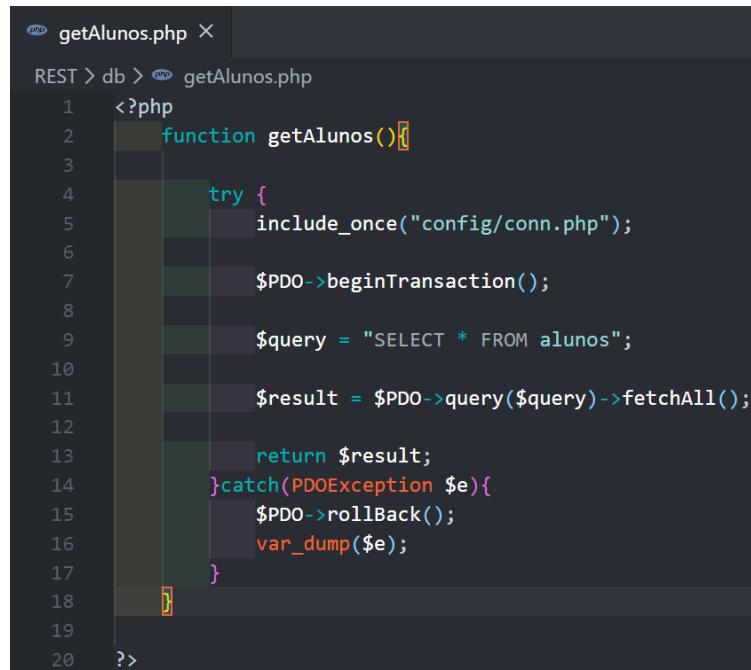


```

AlunosRestHandler.php ×
REST > db > AlunosRestHandler.php
1 <?php
2 require_once("SimpleRest.php");
3 require_once("getAlunos.php");
4
5 class AlunosRestHandler extends SimpleRest {
6
7     function getAllAlunos() {
8
9         $rawData = getAlunos();
10        if(empty($rawData)) {
11            $statusCode = 404;
12            // $rawData = array('error' => 'No alunos found!');
13        } else {
14            $statusCode = 200;
15        }
16
17        $requestContentType = $_SERVER['HTTP_ACCEPT'];
18        $this->setHttpHeaders($requestContentType, $statusCode);
19
20        if(strpos($requestContentType, 'text/html') !== false) {
21            // print_r($rawData);
22            header('Content-Type: application/xml');
23            $response = $this->encodeXml($rawData);
24            return $response;
25        }
26
27
28    public function encodeXml($responseData) {
29        $dom = new DOMDocument();
30        $dom->encoding = 'utf-8';
31        $dom->xmlVersion = '1.0';
32        $dom->formatOutput = true;
33        $xml_file_name = 'alunos.xml';
34        $root = $dom->createElement('alunos');
35
36        foreach ($responseData as $row) {
37            $aluno_node = $dom->createElement('aluno');
38            $child_node_id = $dom->createElement('id', $row["id"]);
39            $aluno_node->appendChild($child_node_id);
40            $child_node_name = $dom->createElement('nome', $row["nome"]);
41            $aluno_node->appendChild($child_node_name);
42            $root->appendChild($aluno_node);
43        }
44
45        $dom->appendChild($root);
46        $dom->save($xml_file_name);
47
48        $data = file_get_contents($xml_file_name) or die("Failed to load");
49        print_r($data);
}

```

E o Webservice de obtenção dos dados dos alunos “getAlunos.php”:



```

getAlunos.php ×
REST > db > getAlunos.php
1 <?php
2 function getAlunos(){
3
4     try {
5         include_once("config/conn.php");
6
7         $PDO->beginTransaction();
8
9         $query = "SELECT * FROM alunos";
10
11         $result = $PDO->query($query)->fetchAll();
12
13         return $result;
14     }catch(PDOException $e){
15         $PDO->rollBack();
16         var_dump($e);
17     }
18 }
19
20 ?>

```

Desenvolveu-se um controlador REST implementado no ficheiro “RestController.php”:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

```

<?php
require_once("AlunosRestHandler.php");

$view = "";
if(isset($_GET["view"])){
    $view = $_GET["view"];
}

switch($view){
    case "all":
        // to handle REST Url /mobile/list/
        $AlunosRestHandler = new AlunosRestHandler();
        $AlunosRestHandler->getAllAlunos();
        break;
}
?>

```

No ficheiro “SimpleRest.php” apresenta-se a implementação de um pedido simples REST:

```

class SimpleRest {
    private $httpVersion = "HTTP/1.1";

    public function setHttpHeaders($contentType, $statusCode){
        $statusMessage = $this -> getHttpStatusMessage($statusCode);

        header($this->httpVersion . " " . $statusCode . " " . $statusMessage);
        header("Content-Type: " . $contentType);
    }

    public function getHttpStatusMessage($statusCode){
        $httpStatus = array(
            100 => 'Continue',
            101 => 'Switching Protocols',
            200 => 'OK',
            201 => 'Created',
            202 => 'Accepted',
            203 => 'Non-Authoritative Information',
            204 => 'No Content',
            205 => 'Reset Content',
            206 => 'Partial Content',
            300 => 'Multiple Choices',
            301 => 'Moved Permanently',
            302 => 'Found',
            303 => 'See Other',
            304 => 'Not Modified',
            305 => 'Use Proxy',
            306 => '(Unused)',
            307 => 'Temporary Redirect',
            400 => 'Bad Request',
            401 => 'Unauthorized',
            402 => 'Payment Required',
            403 => 'Forbidden',
            404 => 'Not Found',
            405 => 'Method Not Allowed',
            406 => 'Not Acceptable',
            407 => 'Proxy Authentication Required',
            408 => 'Request Timeout',
            409 => 'Conflict',
            410 => 'Gone',
            411 => 'Length Required',
            412 => 'Precondition Failed',
            413 => 'Request Entity Too Large',
            414 => 'Request-URI Too Long',
            415 => 'Unsupported Media Type',
            416 => 'Requested Range Not Satisfiable',
            417 => 'Expectation Failed',
            500 => 'Internal Server Error',
            501 => 'Not Implemented',
            502 => 'Bad Gateway',
            503 => 'Service Unavailable',
            504 => 'Gateway Timeout',
            505 => 'HTTP Version Not Supported');
        return ($httpStatus[$statusCode] > $httpStatus[$statusCode] : $status[500]);
    }
}

```

Teste do Webservice (o endpoint terá que ter a view=all)

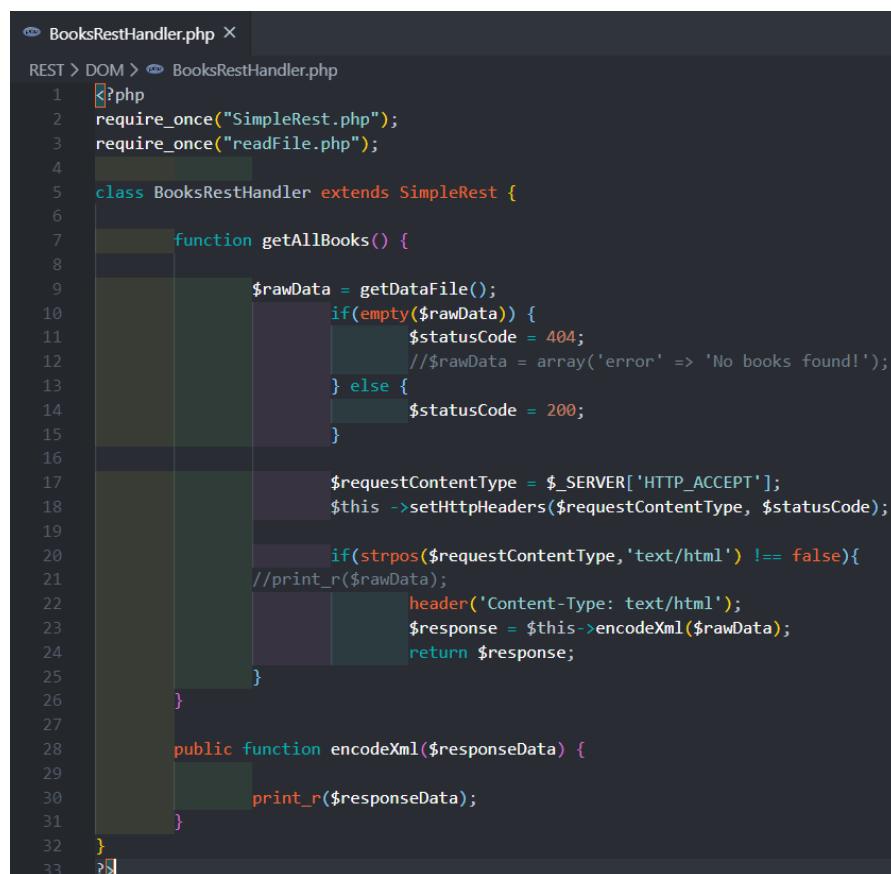
[←](#) [→](#) [C](#) [localhost/IS_TP1_PHP/REST/db/RestController.php?view=all](#)

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<alunos>
  ▼<aluno>
    <id>1</id>
    <nome>Antonio Silva</nome>
  </aluno>
  ▼<aluno>
    <id>2</id>
    <nome>Maria Albertina</nome>
  </aluno>
  ▼<aluno>
    <id>3</id>
    <nome>Jose Ferreira</nome>
  </aluno>
  ▼<aluno>
    <id>4</id>
    <nome>Pedro Costa</nome>
  </aluno>
</alunos>
```

2.3.2 Importação do ficheiro books.xml via REST usando o parser DOM e/ou SAX

Considere o ficheiro “BooksRestHandler.php”:



```
BooksRestHandler.php x
REST > DOM > BooksRestHandler.php
1 ?php
2 require_once("SimpleRest.php");
3 require_once("readFile.php");
4
5 class BooksRestHandler extends SimpleRest {
6
7     function getAllBooks() {
8
9         $rawData = getDataFile();
10        if(empty($rawData)) {
11            $statusCode = 404;
12            // $rawData = array('error' => 'No books found!');
13        } else {
14            $statusCode = 200;
15        }
16
17        $requestContentType = $_SERVER['HTTP_ACCEPT'];
18        $this->setHttpHeaders($requestContentType, $statusCode);
19
20        if(strpos($requestContentType, 'text/html') !== false){
21            //print_r($rawData);
22            header('Content-Type: text/html');
23            $response = $this->encodeXml($rawData);
24            return $response;
25        }
26    }
27
28    public function encodeXml($responseData) {
29
30        print_r($responseData);
31    }
32}
33 ?>
```

Eo ficheiro “readFile.php” para leitura do ficheiro via DOM:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

```

readFile.php ×
REST > DOM > readFile.php
1  <?php
2
3  $dom = new domDocument;
4
5  $dom->load("files/books.xml");
6
7  $dom->preserveWhiteSpace = false;
8
9  $tables = $dom->getElementsByTagName('catalog');
10
11 $rows = $tables->item(0)->getElementsByTagName('book');
12
13 foreach ($rows as $row) {
14
15     $author = $row->getElementsByTagName('author');
16     $title = $row->getElementsByTagName('title');
17     $genre = $row->getElementsByTagName('genre');
18     $price = $row->getElementsByTagName('price');
19     $publish_date = $row->getElementsByTagName('publish_date');
20     $description = $row->getElementsByTagName('description');
21
22
23     echo 'author: '.$author->item(0)->nodeValue.'<br />';
24     echo 'title: '.$title->item(0)->nodeValue.'<br />';
25     echo 'genre: '.$genre->item(0)->nodeValue;
26     echo 'price: '.$price->item(0)->nodeValue.'<br />';
27     echo 'publish_date: '.$publish_date->item(0)->nodeValue.'<br />';
28     echo 'description: '.$description->item(0)->nodeValue;
29     echo '<hr />';
30
31 }
32 
```

O ficheiro “RestController.php” e o “SimpleRest.php”:

```

RestController.php ×
REST > DOM > RestController.php
1  <?php
2
3  require_once("BooksRestHandler.php");
4
5  $view = "";
6  if(isset($_GET["view"])){
7      $view = $_GET["view"];
8
9  switch($view){
10
11      case "all":
12          // to handle REST Url /mobile/list/
13          $BooksRestHandler = new BooksRestHandler();
14          $BooksRestHandler->getAllBooks();
15          break;
16      }
17
18  }
19 
```

```

SimpleRest.php ×
REST > db > SimpleRest.php
1  <?php
2
3  class SimpleRest {
4
5      private $httpVersion = "HTTP/1.1";
6
7      public function setHttpHeaders($contentType, $statusCode){
8
9          $statusMessage = $this -> getHttpStatusMessage($statusCode);
10
11         header($this->httpVersion . " " . $statusCode . " " . $statusMessage);
12         header("Content-Type: " . $contentType);
13     }
14
15     public function getHttpStatusMessage($statusCode){
16         $httpStatus = array(
17             100 => 'Continue',
18             101 => 'Switching Protocols',
19             200 => 'OK',
20             201 => 'Created',
21             202 => 'Accepted',
22             203 => 'Non-Authoritative Information',
23             204 => 'No Content',
24             205 => 'Reset Content',
25             206 => 'Partial Content',
26             300 => 'Multiple Choices',
27             301 => 'Moved Permanently',
28             302 => 'Found',
29             303 => 'See Other',
30             304 => 'Not Modified',
31             305 => 'Use Proxy',
32             306 => '(Unused)',
33             307 => 'Temporary Redirect',
34             400 => 'Bad Request',
35             401 => 'Unauthorized',
36             402 => 'Payment Required',
37             403 => 'Forbidden',
38             404 => 'Not Found',
39             405 => 'Method Not Allowed',
40             406 => 'Not Acceptable',
41             407 => 'Proxy Authentication Required',
42             408 => 'Request Timeout',
43             409 => 'Conflict',
44             410 => 'Gone',
45             411 => 'Length Required',
46             412 => 'Precondition Failed',
47             413 => 'Request Entity Too Large',
48             414 => 'Request-URI Too Long',
49             415 => 'Unsupported Media Type',
50             416 => 'Requested Range Not Satisfiable',
51             417 => 'Expectation Failed',
52             500 => 'Internal Server Error',
53             501 => 'Not Implemented',
54             502 => 'Bad Gateway',
55             503 => 'Service Unavailable',
56             504 => 'Gateway Timeout',
57             505 => 'HTTP Version Not Supported');
58
59         return ($httpStatus[$statusCode]) ? $httpStatus[$statusCode] : $status[500];
60     }
61 
```

Teste do Webservice:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

author: Gambardella, Matthew	
title: XML Developer's Guide	
genre: Computerprice: 44.95	
publish_date: 2000-10-01	
description: An in-depth look at creating applications with XML.	
author: Ralls, Kim	
title: Midnight Rain	
genre: Fantasyprice: 5.95	
publish_date: 2000-12-16	
description: A former architect battles corporate zombies, an evil sorceress, and her own childhood to become queen of the world.	
author: Corets, Eva	
title: Maeve Ascendant	
genre: Fantasyprice: 5.95	
publish_date: 2000-11-17	
description: After the collapse of a nanotechnology society in England, the young survivors lay the foundation for a new society.	
author: Corets, Eva	
title: Oberon's Legacy	
genre: Fantasyprice: 5.95	
publish_date: 2001-03-10	
description: In post-apocalypse England, the mysterious agent known only as Oberon helps to create a new life for the inhabitants of London. Sequel to Maeve Ascendant.	
author: Corets, Eva	
title: The Sundered Grail	
genre: Fantasyprice: 5.95	
publish_date: 2001-09-10	
description: The two daughters of Maeve, half-sisters, battle one another for control of England. Sequel to Oberon's Legacy.	
author: Randall, Cynthia	
title: Lover Birds	
genre: Romanceprice: 4.95	
publish_date: 2000-09-02	
description: When Carla meets Paul at an ornithology conference, tempers fly as feathers get ruffled.	
author: Thurman, Paula	
title: Splash Splash	
genre: Romanceprice: 4.95	
publish_date: 2000-11-02	
description: A deep sea diver finds true love twenty thousand leagues beneath the sea.	

Consider o ficheiro “**readFile.php**”:

```
@@ readfile.php x
REST > SAX > @@ readfile.php
1 <?php
2
3     $tutors = array();
4     $elements = null;
5
6     function startElements($parser, $name, $attrs) {
7         global $tutors, $elements;
8
9         if(empty($name)) {
10             if($name == 'BOOK') {
11                 $tutors []= array();
12             }
13             $elements = $name;
14         }
15     }
16
17     function endElements($parser, $name) {
18         global $elements;
19
20         if(empty($name)) {
21             $elements = null;
22         }
23     }
24
25     function characterData($parser, $data) {
26         global $tutors, $elements;
27
28         if(empty($data)) {
29             if($elements == 'AUTHOR' || $elements == 'TITLE' || $elements == 'GENRE' || $elements == 'PRICE' || $elements == 'PUBLISH_DATE' || $elements == 'DESCRIPTION') {
30                 $tutors[count($tutors)-1][$elements] = trim($data);
31             }
32         }
33     }
34
35     function getDatafile() {
36         global $tutors, $elements;
37
38         $parser = xml_parser_create();
39
40         xml_set_element_handler($parser, "startElements", "endElements");
41         xml_set_character_data_handler($parser, "characterData");
42
43         if (($handle = fopen('files/hooks.xml', "r")) === false) {
44             die("could not open XML input");
45         }
46
47         while($data = fread($handle, 8096)) {
48             xml_parse($parser, $data);
49         }
50
51         xml_parser_free($parser);
52
53         $dataValue = '';
54         foreach($tutors as $course) {
55             $dataValue .= "author - " . $course['AUTHOR'] . '  
';
56             $dataValue .= "title - " . $course['TITLE'] . '  
';
57             $dataValue .= "genre - " . $course['GENRE'] . '  
';
58             $dataValue .= "price - " . $course['PRICE'] . '  
';
59             $dataValue .= "publish_date - " . $course['PUBLISH_DATE'] . '  
';
60             $dataValue .= "description - " . $course['DESCRIPTION'] . '  
';
61         }
62
63         return $dataValue;
64     }
65
66     </?php
```

Execução do webservice usando SAX (o endpoint terá que ter “?view=all”):

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

← → C ⓘ localhost/IS_TP1_PHP/REST/SAX/RestController.php?view=all

author - Gambardella, Matthew
title - XML Developer's Guide
genre - Computer
price - 44.95
publish_date - 2000-10-01
description - with XML.

author - Ralls, Kim
title - Midnight Rain
genre - Fantasy
price - 5.95
publish_date - 2000-12-16
description - of the world.

author - Corets, Eva
title - Maeve Ascendant
genre - Fantasy
price - 5.95
publish_date - 2000-11-17
description - foundation for a new society.

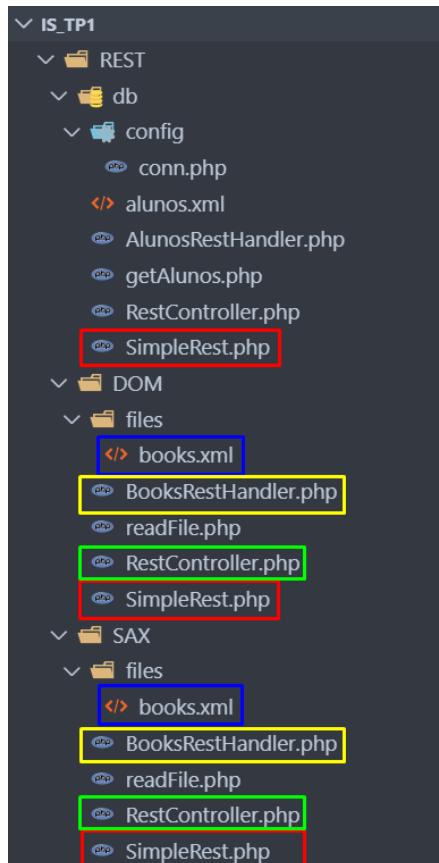
author - Corets, Eva
title - Oberon's Legacy
genre - Fantasy
price - 5.95
publish_date - 2001-03-10
description - Ascendant.

author - Corets, Eva
title - The Sundered Grail
genre - Fantasy
price - 5.95
publish_date - 2001-09-10
description - Oberon's Legacy.

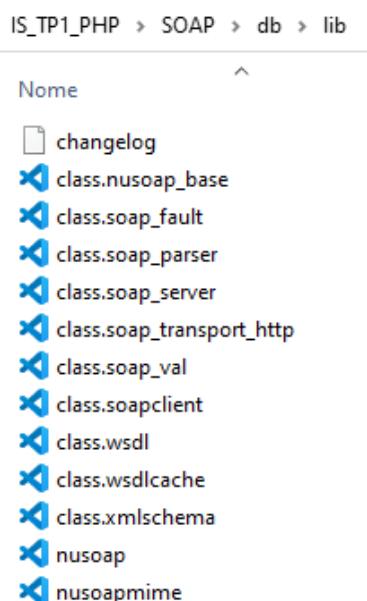
author - Randall, Cynthia
title - Lover Birds
genre - Romance
price - 4.95
publish_date - 2000-09-02
description - conference, tempers fly as feathers get ruffled.

2.3.3 Execução de webservice via SOAP para leitura de dados de uma base de dados

Considere a seguinte estrutura de ficheiros na diretoria SOAP, os ficheiros com a mesma cor são iguais:



Considere a inclusão do package “nusoap.php”:



Considere as configurações de acesso à base de dados:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

IS > IS_TP1_PHP > SOAP > db > config > conn.php

```
?<?php
    $servername = "localhost:3306";
    $username = "dbuser";
    $password = "teste";
    $database = "is";

    try {
        $PDO = new PDO("mysql:host=$servername;dbname=$database", $username, $password);

        $PDO->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
        $PDO->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE, PDO::FETCH_ASSOC);

    } catch (PDOException $e) {
        var_dump($e);
    }
?>
```

Considere o ficheiro “client.php”:

IS > IS_TP1_PHP > SOAP > db > client.php

```
?<?php
    require_once "lib/nusoap.php";

    $client = new nusoap_client(['http://localhost/IS_TP1_PHP/SOAP/db/server.php?wsdl']);

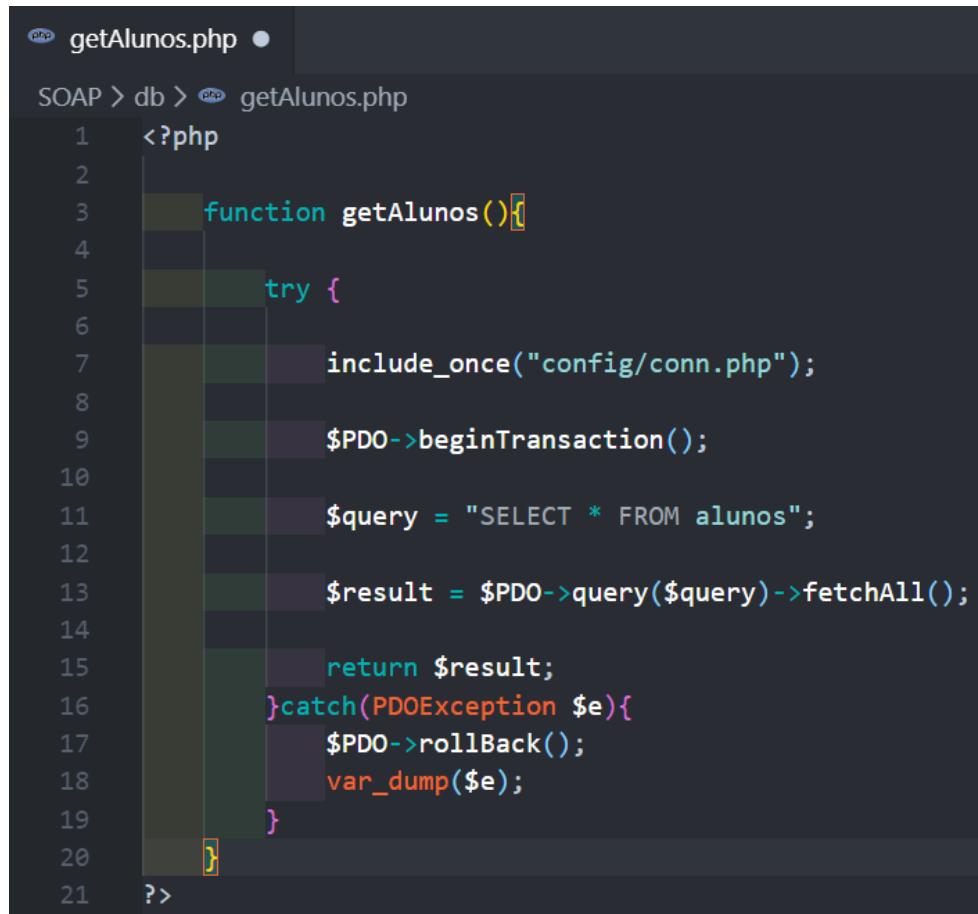
    $param = array("");

    header('Content-Type: application/xml');
    $result = $client->call('getData', $param);
    print_r($result);
?>
```

E o ficheiro “getAlunos.php”:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação



The screenshot shows a code editor with a dark theme. The file is named `getAlunos.php`. The code is a PHP script for a SOAP service. It includes a function `getAlunos()` that connects to a database using PDO, performs a SELECT query on the `alunos` table, and handles exceptions by rolling back the transaction and dumping the error.

```
<?php
function getAlunos(){
    try {
        include_once("config/conn.php");
        $PDO->beginTransaction();
        $query = "SELECT * FROM alunos";
        $result = $PDO->query($query)->fetchAll();
        return $result;
    }catch(PDOException $e){
        $PDO->rollBack();
        var_dump($e);
    }
?>
```

E o ficheiro “**server.php**”:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

```

server.php x
SOAP > db > server.php
1 <?php
2     require_once("lib/nusoap.php");
3     include_once("getAlunos.php");
4
5     $server = new nusoap_server();
6
7     $server->configureWSdl('urn:Server');
8     $server->wsdl->schemaTargetNamespace = 'urn:Server';
9
10
11     function getData($file){
12         return getDataFromFile();
13     }
14
15     function getDataFromFile(){
16         $data = getAlunos();
17
18         $dom = new DOMDocument();
19         $dom->encoding = 'utf-8';
20         $dom->xmlVersion = '1.0';
21         $dom->formatOutput = true;
22         $xml_file_name = 'alunos.xml';
23         $root = $dom->createElement('alunos');
24
25         foreach ($data as $row) {
26             $aluno_node = $dom->createElement('aluno');
27             $child_node_id = $dom->createElement('id', $row["id"]);
28             $aluno_node->appendChild($child_node_id);
29             $child_node_name = $dom->createElement('name', $row["name"]);
30             $aluno_node->appendChild($child_node_name);
31             $root->appendChild($aluno_node);
32         }
33
34         $dom->appendChild($root);
35         $dom->save($xml_file_name);
36
37         $dataDom = file_get_contents($xml_file_name) or die("Failed to load");
38
39         return $dataDom;
40     }
41
42     $server->register (
43         'getData',
44         array("file"=>"xsd:string"),
45         array("return"=>"xsd:string"),
46         'urn:Server.getData',
47         'urn:Server.getData',
48         'rpc',
49         'encoded',
50         'buscar dados a uma base de dados'
51     );
52
53     $HTTP_RAW_POST_DATA = isset( $HTTP_RAW_POST_DATA ) ? $HTTP_RAW_POST_DATA : file_get_contents("php://input");
54     $server->service($HTTP_RAW_POST_DATA);
55 }

```

Invocação do webservice:

View the [WSDL](#) for the service. Click on an operation name to view its details.

[getData](#)

← → C ⓘ localhost/IS_TP1_PHP/SOAP/db/server.php#

urn:Server

View the [WSDL](#) for the service. Click on an item to view its details.

[getData](#)

[Close](#)

```
Name: getData
Binding: urn:ServerBinding
Endpoint: http://localhost/IS_TP1_PHP/SOAP/db/server.php
SoapAction: urn:Server.getData
Style: rpc
Input:
use: encoded
namespace: urn:Server.getData
encodingStyle: http://schemas.xmlsoap.org/soap/encoding/
message: getDataRequest
parts:
file: xsd:string
Output:
use: encoded
namespace: urn:Server.getData
encodingStyle: http://schemas.xmlsoap.org/soap/encoding/
message: getDataResponse
parts:
return: xsd:string
Namespace: urn:Server.getData
Transport: http://schemas.xmlsoap.org/soap/http
Documentation: obter dados de uma base de dados
```

Se aparecer este erro e analisarem o código fonte:

← → C ⓘ localhost/IS_TP1_PHP/SOAP/db/server.php?wsdl

This page contains the following errors:

error on line 2 at column 1: Extra content at the end of the document

Below is a rendering of the page up to the first error.

← → C ⓘ view-source:localhost/IS_TP1_PHP/SOAP/db/server.php?wsdl

```
1 <br />
2 <b>Notice</b></>; Trying to access array offset on value of type bool in <b>C:\tomcataulas\htdocs\IS_TP1_PHP\SOP\dl\lib\nusoap.php</b> on line <b>5534</b><br />
3 <br />
4 <b>Notice</b></>; Trying to access array offset on value of type bool in <b>C:\tomcataulas\htdocs\IS_TP1_PHP\SOP\dl\lib\nusoap.php</b> on line <b>5534</b><br />
5 <?xml version="1.0" encoding="ISO-8859-1"?>
6 <definitions xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns="http://localhost/soap/urn:Server" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns="http://schemas.xmlsoap.org/wsdl/" targetNamespace="http://localhost/soap/urn:Server">
7 <types>
8 <xsd:schema targetNamespace="http://localhost/soap/urn:Server">
9 >
10 <xsd:import namespace="http://schemas.xmlsoap.org/soap/encoding/" />
11 <xsd:import namespace="http://schemas.xmlsoap.org/wsdl/" />
12 </xsd:schema>
13 </types>
14 <message name="getDataRequest">
15 <part name="file" type="xsd:string" /></message>
16 <message name="getDataResponse">
17 <part name="return" type="xsd:string" /></message>
18 <portType name="urn:ServerPortType">
19 <operation name="getData">
20 <documentation>obter dados de uma base de dados</documentation>
21 <input message="tns:getDataRequest"/>
22 <output message="tns:getDataResponse"/>
23 </operation>
24 </portType>
25 <binding name="urn:ServerBinding" type="tns:urn:ServerPortType">
26 <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
27 <operation name="getData">
28 <soap:operation soapAction="urn:Server.getData" style="rpc"/>
29 <input><soap:body use="encoded" namespace="urn:Server.getData" encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" /></input>
30 <output><soap:body use="encoded" namespace="urn:Server.getData" encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" /></output>
31 </operation>
32 </binding>
33 <service name="urn:Server">
34 <port name="urn:ServerPort" binding="tns:urn:ServerBinding">
35 <soap:address location="http://localhost/IS_TP1_PHP/SOAP/db/server.php"/>
36 </port>
37 </service>
38 </definitions>
```

Ao analisar a linha 5534, verifica-se que o problema está na construção dos elementos XML:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

```
C: > tomcataulas > htdocs > IS_TP1_PHP > SOAP > DOM > lib > nusoap.php
      use \SoapType has no namespace: )
5529
5530
5531
5532
5533
5534 if ($TypeDef['typeClass'] == 'element') {
5535     $elementType = 'element';
5536     if (substr($localPart, -1) == '^') {
5537         $localPart = substr($localPart, 0, -1);
5538     }
5539 } else {
5540     $elementType = 'type';
5541 }
5542 $xml .= "\n" . ' <part name="' . $partName . '" ' . $eleme
5543 }
5544 }
5545 $xml .= '</message>';
5546 }
```

Acontece que o webservice devolve um valor vazio/nulo em vez de um documento XML. Para testar esta situação acrescentou-se uma linha de código para devolver o resultado:

```
C: > tomcataulas > htdocs > IS_TP1_PHP > SOAP > db > client.php
1 <?php
2
3     require_once "lib/nusoap.php";
4
5     $client = new nusoap_client('http://localhost/IS_TP1_PHP/SOAP/db/server.php?wsdl');
6
7     $param = array("");
8
9     header('Content-Type: application/xml');
10    $result = $client->call('getData', $param);
11    echo "Devolve Resultado do Webservice:". $result;
12    print_r($result);
13 ?>
```

This page contains the following errors:
error on line 1 at column 1: Document is empty

Below is a rendering of the page up to the first error.

Name	Headers	Preview	Response	Initiator	Timing
client.php			1 Devolve Resultado do Webservice:		
favicon.ico					

Ao verificar a execução do server.php:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

This page contains the following errors:
error on line 2 at column 1: Extra content at the end of the document
Below is a rendering of the page up to the first error.

O problema encontra-se na devolução de resultados da base de dados através do comando `fetcall()`

```
htdocs > IS_TP1_PHP > SOAP > db > getAlunos.php
<?php

function getAlunos(){

    try {
        require_once("config/conn.php");

        $PDO->beginTransaction();

        $query = "SELECT * FROM alunos";

        $result = $PDO->query($query)->fetchAll();
        return $result;
    }catch(PDOException $e){
        $PDO->rollBack();
        var_dump($e);
    }
}
```

Para resolver o problema, deverá verificar os packages de instalação do pacote PDO no PHP, ou então proceder a uma forma diferente de acesso aos dados da base de dados e à devolução de resultados de uma query.

Uma solução poderá passar pela adição da extensão ou retirar o comentário da linha para o `pdo_mysql`.

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

```
php - Bloco de notas
Ficheiro Editar Formatar Ver Ajuda
; move to the new ('extension=<ext>) syntax.
;
; Notes for Windows environments :
;
; - Many DLL files are located in the extensions/ (PHP 4) or ext/ (PHP 5+)
;   extension folders as well as the separate PECL DLL download (PHP 5+).
;   Be sure to appropriately set the extension_dir directive.
;
extension=bz2
extension=curl
;extension=ffi
;extension=ftp
extension=fileinfo
extension=gd2
extension=gettext
;extension=gmp
;extension=intl
;extension=imap
;extension=ldap
extension=mbstring
extension=exif      ; Must be after mbstring as it depends on it
extension=mysql
;extension=oci8_12c ; Use with Oracle Database 12c Instant Client
;extension=odbc
;extension=openssl
;extension=pdo_firebird
extension=pdo_mysql
;extension=pdo_oci
;extension=pdo_odbc
;extension=pdo_pgsql
extension=pdo_sqlite
;extension=pgsql
;extension=shmop
; The MIBS data available in the PHP distribution must be installed.
; See http://www.php.net/manual/en/snmp.installation.php
;extension=snmp

;extension=soap
;extension=sockets
;extension=sodium
```

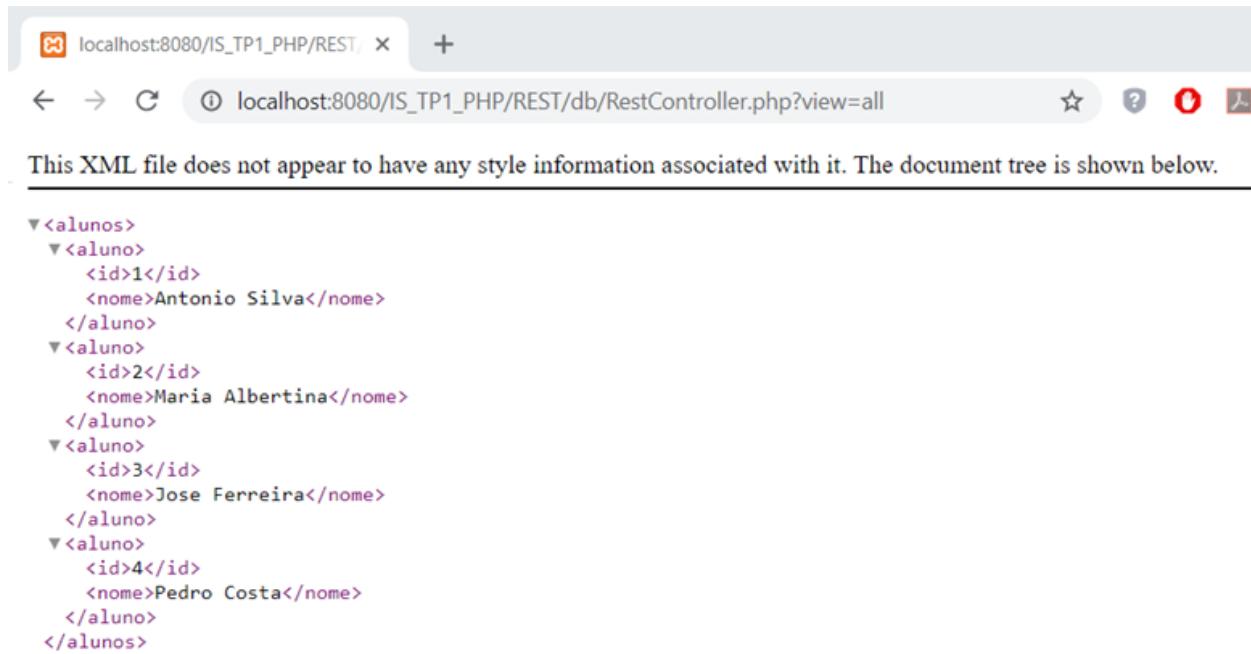
O resultado da invocação do webservice é a seguinte:

← → ⌂ i localhost/IS_TP1_PHP/SOAP/db/client.php

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="UTF-8"?>
<alunos>
  <aluno>
    <id>1</id>
    <nome>Antonio Silva</nome>
  </aluno>
  <aluno>
    <id>2</id>
    <nome>Maria Albertina</nome>
  </aluno>
  <aluno>
    <id>3</id>
    <nome>Jose Ferreira</nome>
  </aluno>
  <aluno>
    <id>4</id>
    <nome>Pedro Costa</nome>
  </aluno>
</alunos>
```

E o do acesso à BD:



This XML file does not appear to have any style information associated with it. The document tree is shown below.

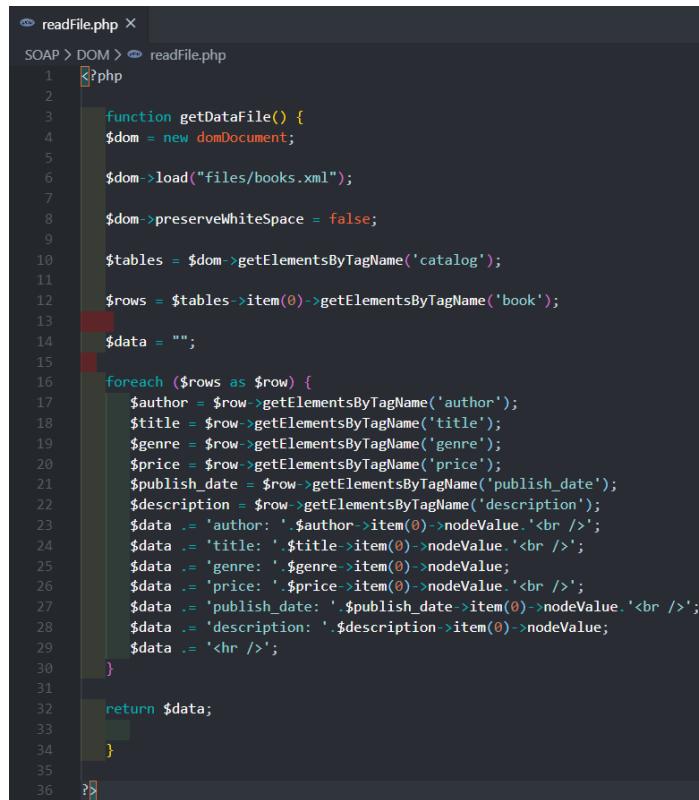
```
<?xml version="1.0"?>
<alunos>
  <aluno>
    <id>1</id>
    <nome>Antonio Silva</nome>
  </aluno>
  <aluno>
    <id>2</id>
    <nome>Maria Albertina</nome>
  </aluno>
  <aluno>
    <id>3</id>
    <nome>Jose Ferreira</nome>
  </aluno>
  <aluno>
    <id>4</id>
    <nome>Pedro Costa</nome>
  </aluno>
</alunos>
```

2.3.4 Implementação de webservice via SOAP para leitura de ficheiro books.xml usando o parser DOM

Considere o ficheiro “cliente.php”:

```
> htdocs > IS_TP1_PHP > SOAP > DOM > client.php
1  <?php
2      require_once("lib/nusoap.php");
3
4      $client = new nusoap_client('http://localhost/IS_TP1_PHP/SOAP/DOM/server.php?wsdl');
5
6      $param = array('file'=>'books.xml');
7
8      $result = $client->call('getData', $param);
9
10     print_r($result);
11 ?>
```

E o ficheiro “readFile.php”

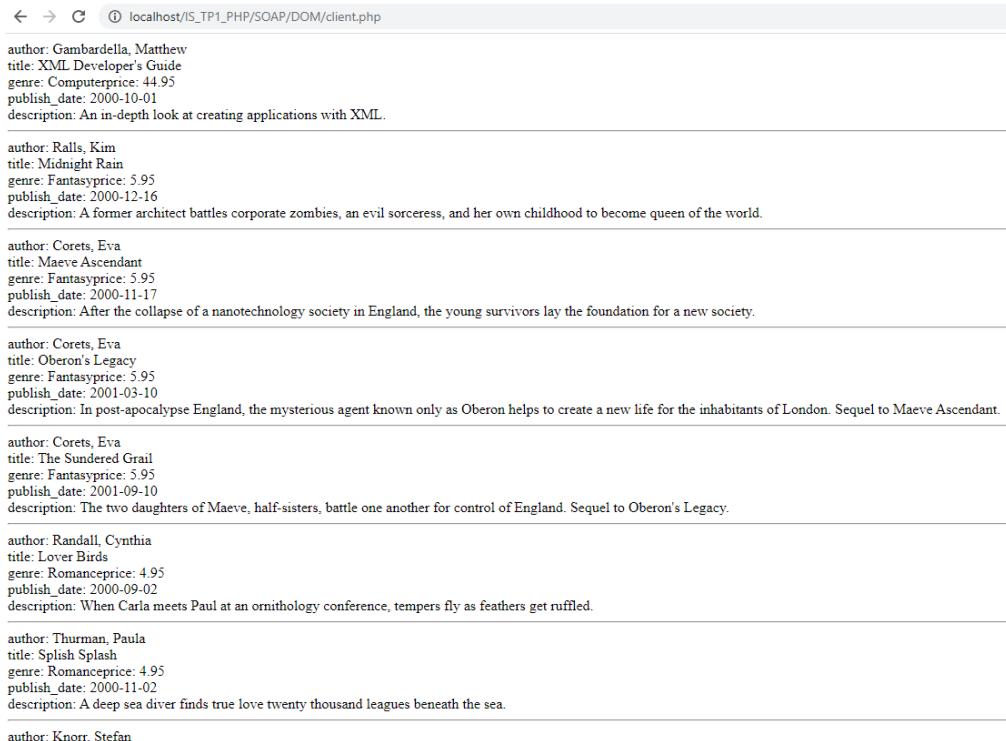


```

readFile.php x
SOAP > DOM > readFile.php
1 <?php
2
3     function getDataFile() {
4         $dom = new domDocument;
5
6         $dom->load("files/books.xml");
7
8         $dom->preserveWhiteSpace = false;
9
10        $tables = $dom->getElementsByTagName('catalog');
11
12        $rows = $tables->item(0)->getElementsByTagName('book');
13
14        $data = "";
15
16        foreach ($rows as $row) {
17            $author = $row->getElementsByTagName('author');
18            $title = $row->getElementsByTagName('title');
19            $genre = $row->getElementsByTagName('genre');
20            $price = $row->getElementsByTagName('price');
21            $publish_date = $row->getElementsByTagName('publish_date');
22            $description = $row->getElementsByTagName('description');
23            $data .= 'author: ' . $author->item(0)->nodeValue. '<br />';
24            $data .= 'title: ' . $title->item(0)->nodeValue. '<br />';
25            $data .= 'genre: ' . $genre->item(0)->nodeValue;
26            $data .= 'price: ' . $price->item(0)->nodeValue. '<br />';
27            $data .= 'publish_date: ' . $publish_date->item(0)->nodeValue. '<br />';
28            $data .= 'description: ' . $description->item(0)->nodeValue;
29            $data .= '<hr />';
30        }
31
32        return $data;
33    }
34
35
36 ?>

```

O resultado da invocação do webservice é a seguinte:



author: Gambardella, Matthew	title: XML Developer's Guide	genre: Computer	price: 49.95
publish_date: 2000-10-01	description: An in-depth look at creating applications with XML.		
author: Ralls, Kim	title: Midnight Rain	genre: Fantasy	price: 5.95
publish_date: 2000-12-16	description: A former architect battles corporate zombies, an evil sorceress, and her own childhood to become queen of the world.		
author: Corets, Eva	title: Maeve Ascendant	genre: Fantasy	price: 5.95
publish_date: 2000-11-17	description: After the collapse of a nanotechnology society in England, the young survivors lay the foundation for a new society.		
author: Corets, Eva	title: Oberon's Legacy	genre: Fantasy	price: 5.95
publish_date: 2001-03-10	description: In post-apocalypse England, the mysterious agent known only as Oberon helps to create a new life for the inhabitants of London. Sequel to Maeve Ascendant.		
author: Corets, Eva	title: The Sundered Grail	genre: Fantasy	price: 5.95
publish_date: 2001-09-10	description: The two daughters of Maeve, half-sisters, battle one another for control of England. Sequel to Oberon's Legacy.		
author: Randall, Cynthia	title: Lover Birds	genre: Romance	price: 4.95
publish_date: 2000-09-02	description: When Carla meets Paul at an ornithology conference, tempers fly as feathers get ruffled.		
author: Thurman, Paula	title: Splash Splash	genre: Romance	price: 4.95
publish_date: 2000-11-02	description: A deep sea diver finds true love twenty thousand leagues beneath the sea.		
author: Knorr, Stefan			

2.3.5 Implementação de webservice via SOAP para leitura de ficheiro books.xml usando o parser SAX

Considere o ficheiro "client.php":

```
htdocs > IS_TP1_PHP > SOAP > SAX > client.php
<?php
    require_once("lib/nusoap.php");

    $client = new nusoap_client('http://localhost/IS_TP1_PHP/SOAP/SAX/server.php?wsdl');

    $param = array('file'=>'books.xml');

    $result = $client->call('getData', $param);

    print_r($result);
?>
```

O Ficheiro “readFile.php”

```
readfile.php <
SOAP > SAX > readfile.php
1 <?php
2
3     $tutors = array();
4     $elements = null;
5
6     function startElements($parser, $name, $attrs) {
7         global $tutors, $elements;
8
9         if(empty($name)) {
10             if ($name == 'BOOK') {
11                 $tutors []= array();
12             }
13             $elements = $name;
14         }
15     }
16
17     function endElements($parser, $name) {
18         global $elements;
19
20         if(empty($name)) {
21             $elements = null;
22         }
23     }
24
25     function characterData($parser, $data) {
26         global $tutors, $elements;
27
28         if(empty($data)) {
29             if ($elements == 'AUTHOR' || $elements == 'TITLE' || $elements == 'GENRE' || $elements == 'PRICE' || $elements == 'PUBLISH_DATE' || $elements == 'DESCRIPTION') {
30                 $tutors[count($tutors)-1][$elements] = trim($data);
31             }
32         }
33     }
34
35     function getDataFile() {
36         global $tutors, $elements;
37
38         $parser = xml_parser_create();
39
40         xml_set_element_handler($parser, "startElements", "endElements");
41         xml_set_character_data_handler($parser, "characterData");
42
43         if (!($handle = fopen('files/books.xml', 'r'))) {
44             die("could not open XML input");
45         }
46
47         while($data = fread($handle, 8096)) {
48             xml_parse($parser, $data);
49         }
50
51         xml_parser_free($parser);
52
53         $dataValue = "";
54         foreach($tutors as $course) {
55             $dataValue .= "author - " . $course['AUTHOR'] . '<br/>';
56             $dataValue .= "title - " . $course['TITLE'] . '<br/>';
57             $dataValue .= "genre - " . $course['GENRE'] . '<br/>';
58             $dataValue .= "price - " . $course['PRICE'] . '<br/>';
59             $dataValue .= "publish_date - " . $course['PUBLISH_DATE'] . '<br/>';
60             $dataValue .= "description - " . $course['DESCRIPTION'] . '<br/>';
61         }
62
63         return $dataValue;
64     }
65 ?>
```

O resultado da invocação do webservice é a seguinte:

FICHA PRÁTICA n.º 9

Integração de Sistemas de Informação

← → C

localhost/IS_TP1_PHP/SOAP/SAX/client.php

author - Gambardella, Matthew
title - XML Developer's Guide
genre - Computer
price - 44.95
publish_date - 2000-10-01
description - with XML.

author - Ralls, Kim
title - Midnight Rain
genre - Fantasy
price - 5.95
publish_date - 2000-12-16
description - of the world.

author - Corets, Eva
title - Maeve Ascendant
genre - Fantasy
price - 5.95
publish_date - 2000-11-17
description - foundation for a new society.

author - Corets, Eva
title - Oberon's Legacy
genre - Fantasy
price - 5.95
publish_date - 2001-03-10
description - Ascendant.

author - Corets, Eva
title - The Sundered Grail
genre - Fantasy
price - 5.95
publish_date - 2001-09-10