Versão 1 – Documento em Atualização

Integração de Sistemas e Tecnologias

Interoperabilidade de Sistemas e Tecnologias

Tutorial 9 – WebServices, XML, SOAP, Java, PHP..

Escola Superior de Tecnologia e Gestão
Instituto Politécnico de Viana do Castelo
Braga, Portugal

Jorge Ribeiro
jribeiro@estg.ipvc.pt

- Exercício – DOM (simple.xml, books.xml, medsamp.xml)
- Exercício – SAX (simple.xml, books.xml, medsamp.xml, note.xml)
- Exercício – JDOM (books.xml, medsamp.xml,note.xml)
- Exercício – JAXP (simple.xml, books.xml, medsamp.xml)
- Exercício – JAXB (po.xml)
- SOAP – JAVA:
    - Exemplo nº1 – Adicionar dois números,
    - Exemplo nº2 – Devolver o ficheiro Books.xml para leitura
    - Exemplo nº3 – Devolver dados da base de dados
- SOAP – PHP:
    - Exemplo nº1: Exemplo dado pelo sitepoint.com
    - Exemplo nº2: Devolver o ficheiro Books.xml para leitura
    - Exemplo nº3: Devolver registos da base de dados em XML
- REST – JAVA
    - Exemplo nº1: Devolver o ficheiro Books.xml para leitura
    - Exemplo nº2: Acesso à base de dados e devolução de registos em XML
- REST – PHP
    - Exemplo nº1: Acesso à base de dados e devolução de registos em XML
    - Exemplo nº2: Enviar conteúdo do ficheiro Books.xml
- JAVA – SOAP – DOM, SAX, JDOM, JAXP, JAXB
- JAVA – REST – DOM, SAX, JDOM, JAXP, JAXB
- PHP – SOAP – DOM, SAX, JDOM, JAXP, JAXB
- PHP – REST – DOM, SAX, JDOM, JAXP, REST - JAXB
- PHP – SOAP – JDOM com FileUpload

Ambiente de Desenvolvimento para o Tutorial 9

Netbeans IDE 8.2 ALL (JAVA WebServices)

Glassfish (Java WebServices – Servidor WEB)

Visual Studio Code (PHP WebServices)

WAMP – 3.1.9 64bit (PHP WebServices)

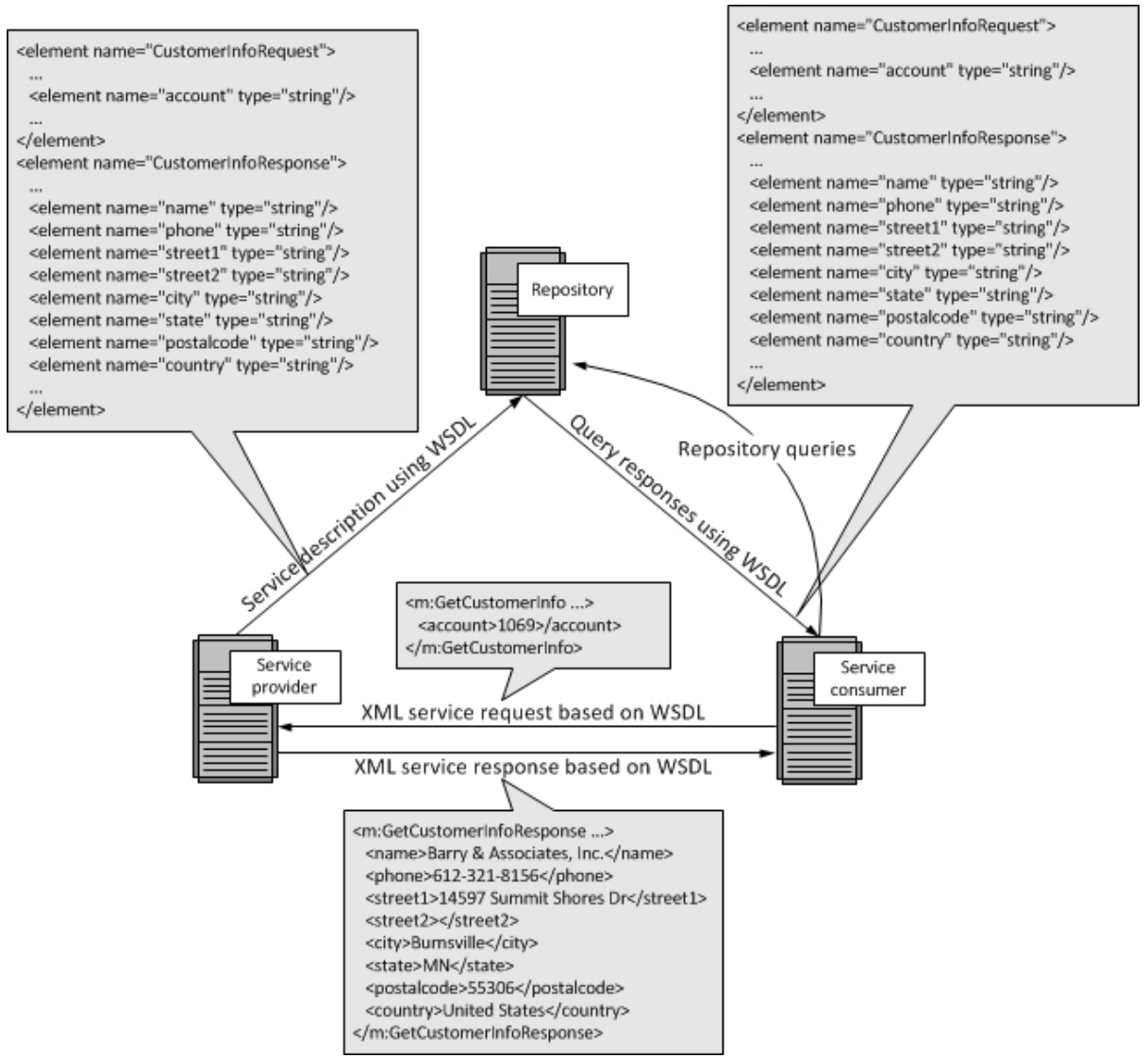
- **Apache 2.4.39**
- **Mysql 5.7.26**
- **PHP 7.2.18**

Testes:

- **SOAP UI**
- **POSTMAN**
- **BROWSER**

SOAP

SOAP was originally part of the specification that included the Web Services Description Language (WSDL) and Universal Description, Discovery, and Integration (UDDI). It is used now without WSDL and UDDI. Instead of the discovery process described in the History of the Web Services Specification section below, SOAP messages are hard-coded or genereated without the use of a repository. The interaction is illustrated in the figure below.



SOAP – JAVA

Exemplo nº1 – Adicionar dois números

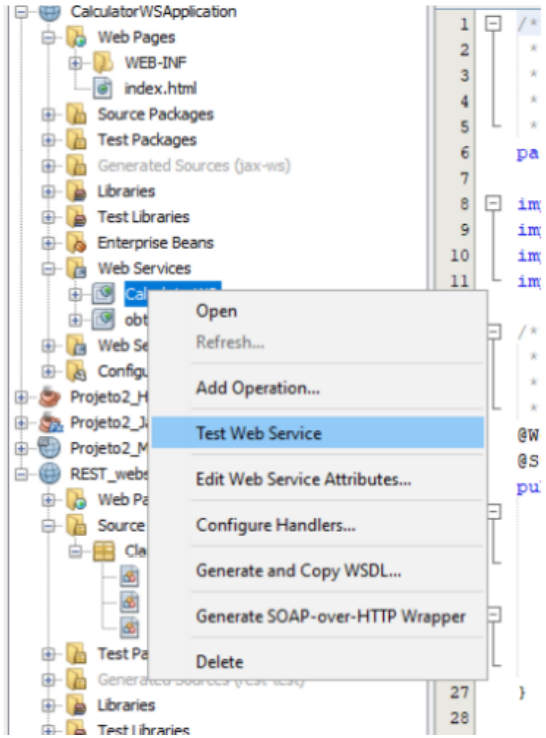
Código

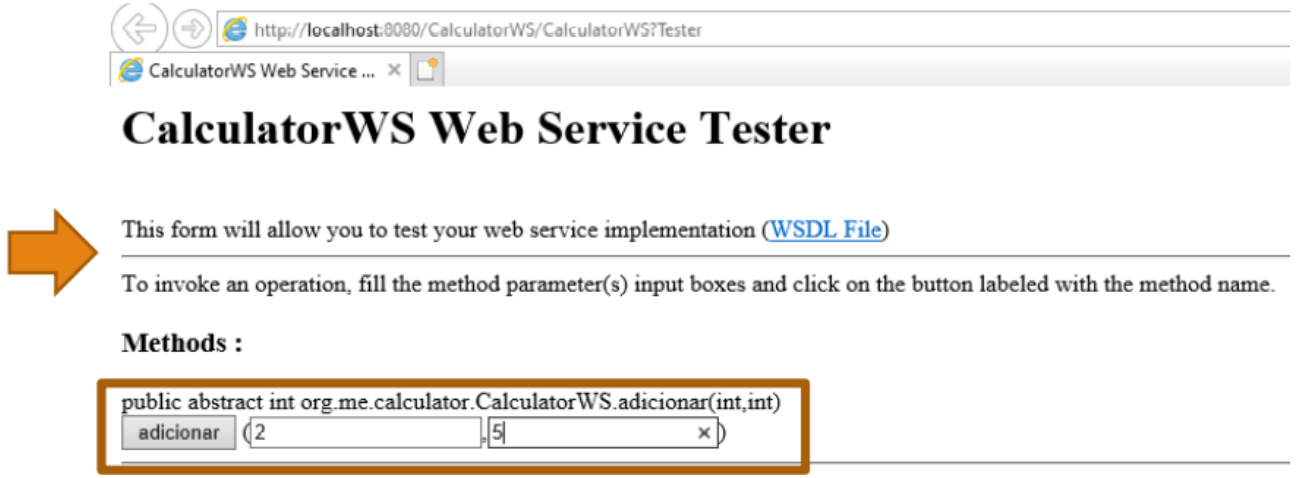
```
@WebService(serviceName = "CalculatorWS")
@Stateless()
public class CalculatorWS {
    /**
     * Web service operation
     */
    @WebMethod(operationName = "adicionar")
    public int adicionar(@WebParam(name = "a") int a, @WebParam(name = "b") int b) {
        return a+b;
    }
}
```

SOAP – JAVA

Exemplo nº1 – Adicionar dois números

Testes – Browser (Opção "Test Web Service" do Netbeans)

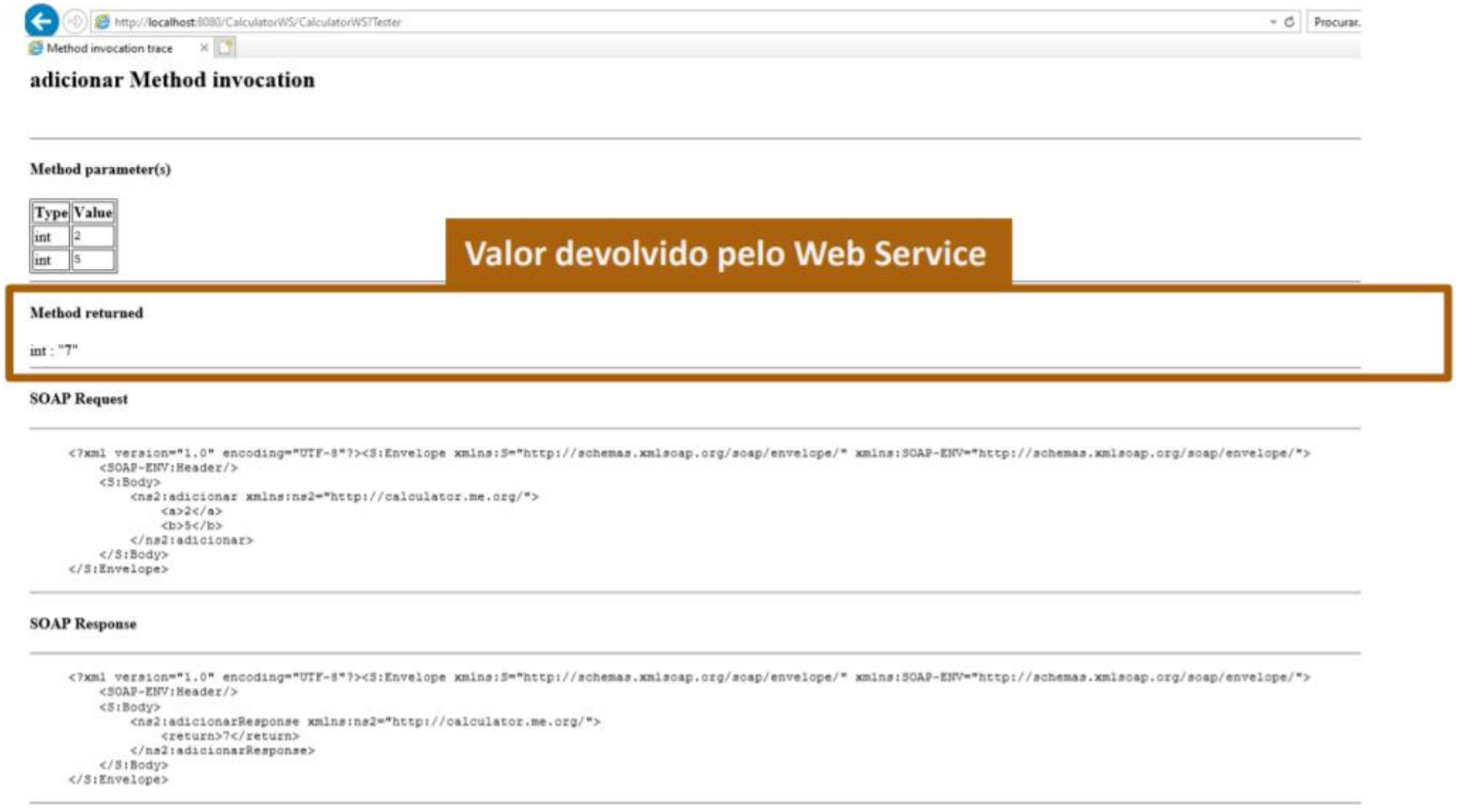




SOAP – JAVA

Exemplo nº1 – Adicionar dois números

Testes – Browser (Opção "Test Web Service" do Netbeans)



SOAP – JAVA

Exemplo nº1 – Adicionar dois números

Testes – Browser (Opção "Test Web Service" do Netbeans)

http://localhost:8080/CalculatorWS/CalculatorWS?Tester — Procurar.

Method invocation trace

adicionar Method invocation

Method parameter(s)

Type	Value
int	2
int	5

Method returned

int : "7"

Formato do Pedido SOAP

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:adicionar xmlns:ns2="http://calculator.me.org/">
            <a>2</a>
            <b>5</b>
        </ns2:adicionar>
    </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:adicionarResponse xmlns:ns2="http://calculator.me.org/">
            <return>7</return>
        </ns2:adicionarResponse>
    </S:Body>
</S:Envelope>
```

SOAP – JAVA

Exemplo nº1 – Adicionar dois números

Testes – Browser (Opção "Test Web Service" do Netbeans)

http://localhost:8080/CalculatorWS/CalculatorWS?Tester — Procurar.

Method invocation trace

adicionar Method invocation

Method parameter(s)

Type	Value
int	2
int	5

Method returned

int : "7"

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:adicionar xmlns:ns2="http://calculator.me.org/">
            <a>2</a>
            <b>5</b>
        </ns2:adicionar>
    </S:Body>
</S:Envelope>
```

Resposta SOAP do servidor

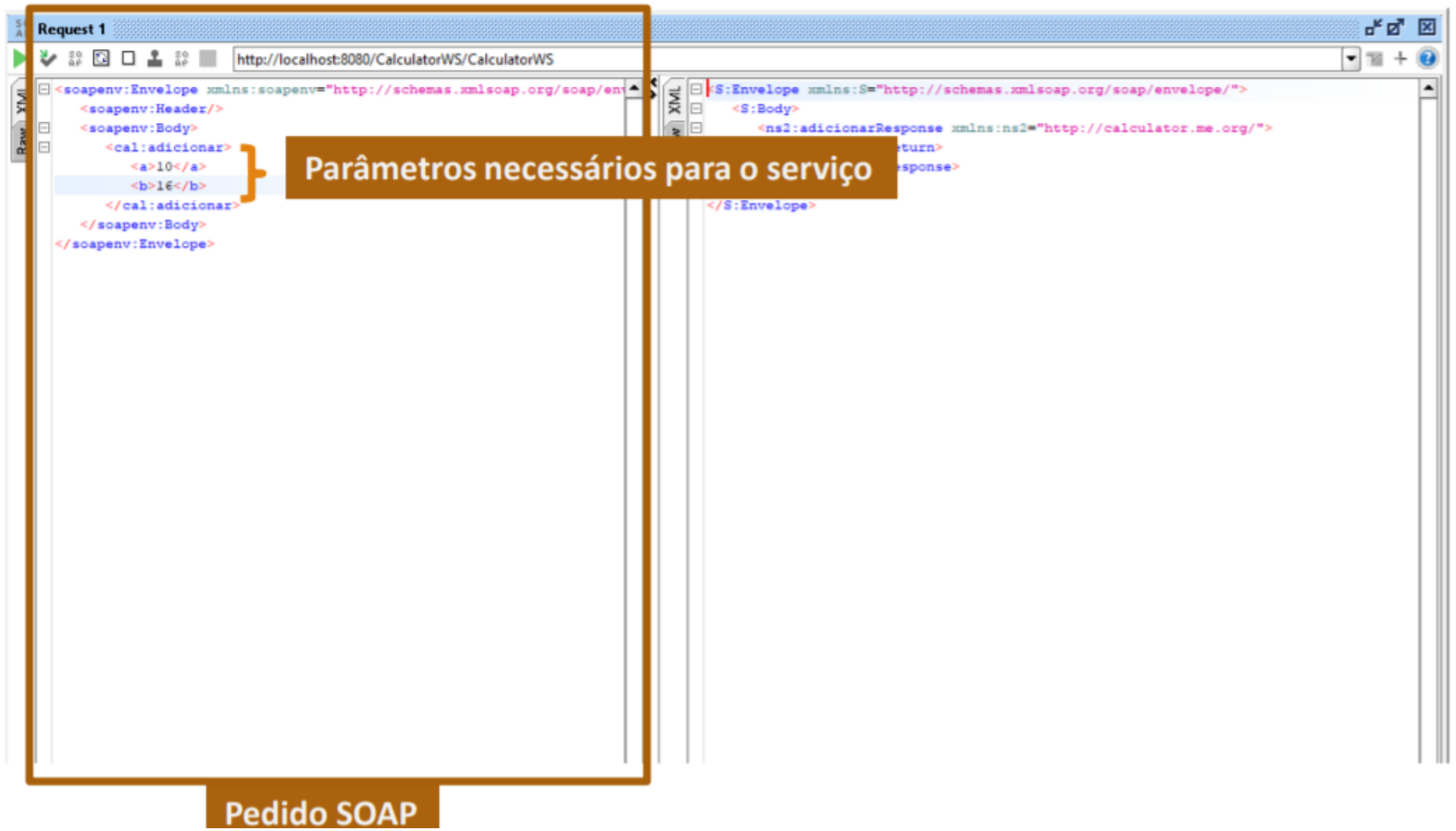
SOAP Response

```
<?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:adicionarResponse xmlns:ns2="http://calculator.me.org/">
            <return>7</return>
        </ns2:adicionarResponse>
    </S:Body>
</S:Envelope>
```

SOAP – JAVA

Exemplo nº1 – Adicionar dois números

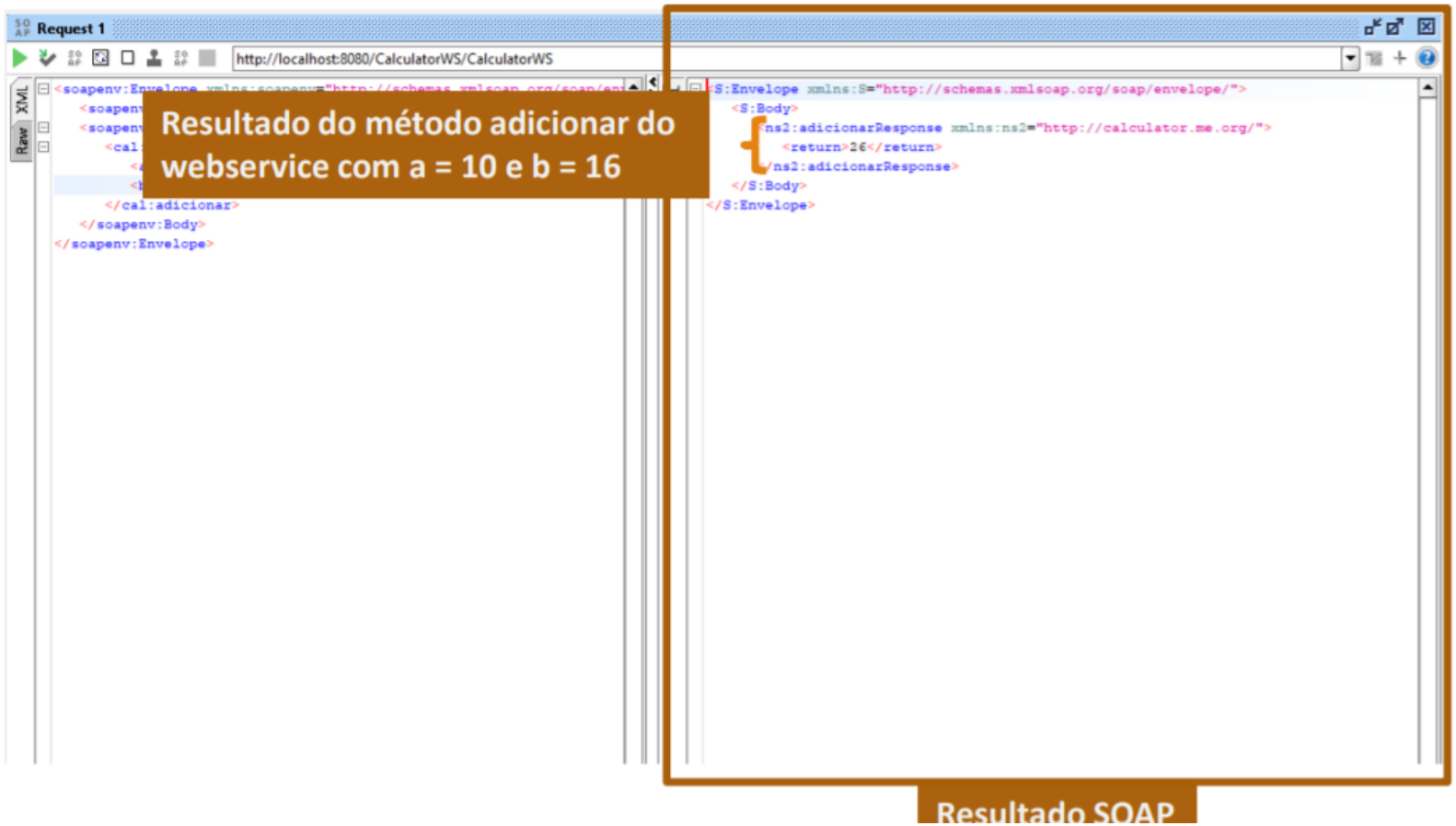
Testes – SoapUI



SOAP – JAVA

Exemplo nº1 – Adicionar dois números

Testes – SoapUI



SOAP – JAVA

Exemplo nº2 – Devolver o ficheiro Books.xml para leitura

Código

```
@WebService(serviceName = "obtemXML")
public class obtemXML {

    private final String caminhoBooksXml = "D:\\IPVC\\5ºS. - Integração de Sistemas\\Trabalhos Práticos\\TP1 - Webservices SOAP e REST\\JAVA\\REST_webservice\\books.xml";

    /**
     * This is a sample web service operation
     */
    @WebMethod(operationName = "obtemXML")
    public String obtemXML() {
        try {
            byte[] encoded = Files.readAllBytes(Paths.get(caminhoBooksXml));
            String retorno = new String(encoded, "utf-8");

            return retorno;
        } catch (IOException ex) {
            return "ERRO: " + ex.getMessage();
        }
    }
}
```

SOAP – JAVA

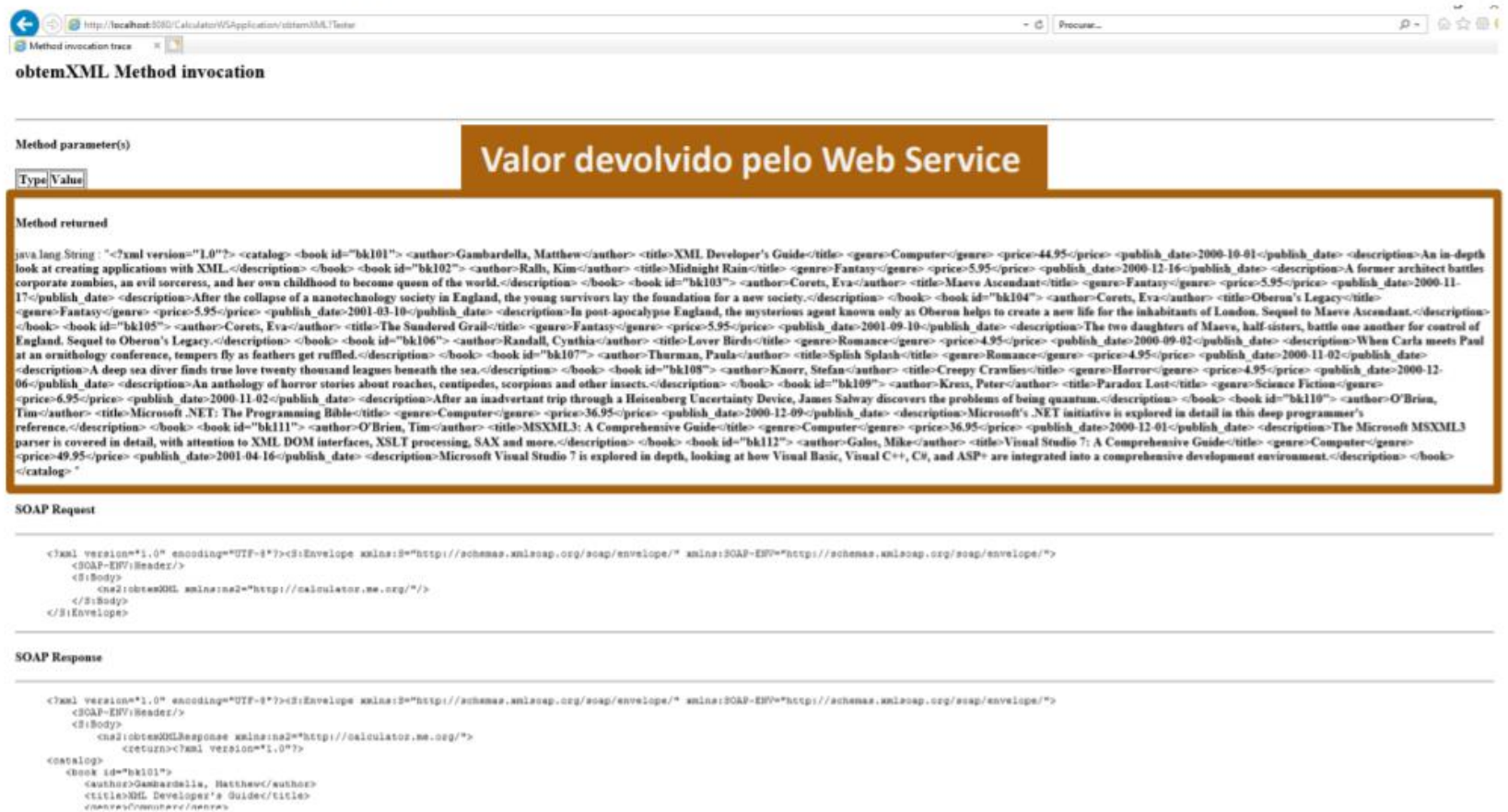
Exemplo nº2 – Devolver o ficheiro Books.xml para leitura

Testes – Browser (Opção "Test Web Service" do Netbeans)

SOAP – JAVA

Exemplo nº2 – Devolver o ficheiro Books.xml para leitura

Testes – Browser (Opção "Test Web Service" do Netbeans)



Method invocation trace

obtemXML Method invocation

Method parameter(s)

Type	Value

Valor devolvido pelo Web Service

Method returned

java.lang.String : "<?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, battle one another for control of England. Sequel to Oberon's Legacy.</description> </book> <book id="bk106"> <author>Randall, Cynthia</author> <title>Lover Birds</title> <genre>Romance</genre> <price>4.95</price> <publish_date>2000-09-02</publish_date> <description>When Carla meets Paul at an ornithology conference, tempers fly as feathers get ruffled.</description> </book> <book id="bk107"> <author>Thurman, Paula</author> <title>Splish Splash</title> <genre>Romance</genre> <price>4.95</price> <publish_date>2000-11-02</publish_date> <description>A deep sea diver finds true love twenty thousand leagues beneath the sea.</description> </book> <book id="bk108"> <author>Knorr, Stefan</author> <title>Creepy Crawlies</title> <genre>Horror</genre> <price>4.95</price> <publish_date>2000-12-06</publish_date> <description>An anthology of horror stories about roaches, centipedes, scorpions and other insects.</description> </book> <book id="bk109"> <author>Kress, Peter</author> <title>Paradox Lost</title> <genre>Science Fiction</genre> <price>6.95</price> <publish_date>2000-11-02</publish_date> <description>After an inadvertant trip through a Heisenberg Uncertainty Device, James Salway discovers the problems of being quantum.</description> </book> <book id="bk110"> <author>O'Brien, Tim</author> <title>Microsoft .NET: The Programming Bible</title> <genre>Computer</genre> <price>36.95</price> <publish_date>2000-12-09</publish_date> <description>Microsoft's .NET initiative is explored in detail in this deep programmer's reference.</description> </book> <book id="bk111"> <author>O'Brien, Tim</author> <title>MSXML3: A Comprehensive Guide</title> <genre>Computer</genre> <price>36.95</price> <publish_date>2000-12-01</publish_date> <description>The Microsoft MSXML3 parser is covered in detail, with attention to XML DOM interfaces, XSLT processing, SAX and more.</description> </book> <book id="bk112"> <author>Galos, Mike</author> <title>Visual Studio 7: A Comprehensive Guide</title> <genre>Computer</genre> <price>49.95</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> "

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:obtemXML xmlns:ns2="http://calculator.me.org/"/>
    </S:Body>
</S:Envelope>
```

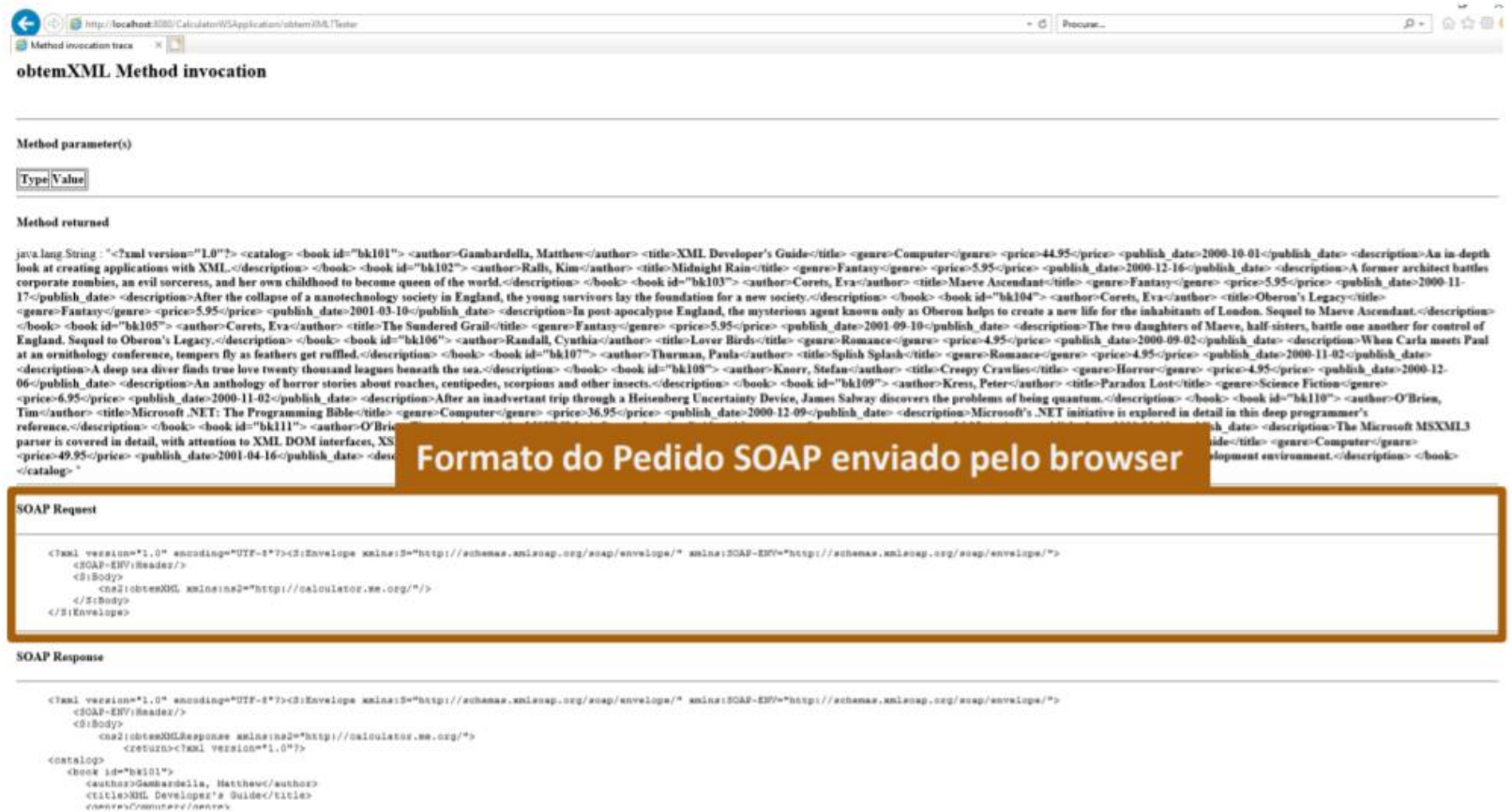
SOAP Response

```
<?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:obtemXMLResponse xmlns:ns2="http://calculator.me.org/">
            <return><?xml version="1.0"?>
<catalog>
   <book id="bk101">
      <author>Gambardella, Matthew</author>
      <title>XML Developer's Guide</title>
```

SOAP – JAVA

Exemplo nº2 – Devolver o ficheiro Books.xml para leitura

Testes – Browser (Opção "Test Web Service" do Netbeans)



SOAP – JAVA

Exemplo nº2 – Devolver o ficheiro Books.xml para leitura

Testes – Browser (Opção "Test Web Service" do Netbeans)

http://localhost:8080/CalculatorWSApplication/obtemXMLTester

Method invocation trace

obtemXML Method invocation

Method parameter(s)

Type	Value

Method returned

java.lang.String : "<?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, battle one another for control of England. Sequel to Oberon's Legacy.</description> </book> <book id="bk106"> <author>Randall, Cynthia</author> <title>Lover Birds</title> <genre>Romance</genre> <price>4.95</price> <publish_date>2000-09-02</publish_date> <description>When Carla meets Paul at an ornithology conference, tempers fly as feathers get ruffled.</description> </book> <book id="bk107"> <author>Thurman, Paula</author> <title>Splish Splash</title> <genre>Romance</genre> <price>4.95</price> <publish_date>2000-11-02</publish_date> <description>A deep sea diver finds true love twenty thousand leagues beneath the sea.</description> </book> <book id="bk108"> <author>Knorr, Stefan</author> <title>Creepy Crawlies</title> <genre>Horror</genre> <price>4.95</price> <publish_date>2000-12-06</publish_date> <description>An anthology of horror stories about roaches, centipedes, scorpions and other insects.</description> </book> <book id="bk109"> <author>Kress, Peter</author> <title>Paradox Lost</title> <genre>Science Fiction</genre> <price>6.95</price> <publish_date>2000-11-02</publish_date> <description>After an inadvertant trip through a Heisenberg Uncertainty Device, James Salway discovers the problems of being quantum.</description> </book> <book id="bk110"> <author>O'Brien, Tim</author> <title>Microsoft .NET: The Programming Bible</title> <genre>Computer</genre> <price>36.95</price> <publish_date>2000-12-09</publish_date> <description>Microsoft's .NET initiative is explored in detail in this deep programmer's reference.</description> </book> <book id="bk111"> <author>O'Brien, Tim</author> <title>MSXML3: A Comprehensive Guide</title> <genre>Computer</genre> <price>36.95</price> <publish_date>2000-12-01</publish_date> <description>The Microsoft MSXML3 parser is covered in detail, with attention to XML DOM interfaces, XSLT processing, SAX and more.</description> </book> <book id="bk112"> <author>Galos, Mike</author> <title>Visual Studio 7: A Comprehensive Guide</title> <genre>Computer</genre> <price>49.95</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> "

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:obtemXML xmlns:ns2="http://calculator.me.or
    </S:Body>
</S:Envelope>
```

Formato da resposta SOAP dada pelo servidor

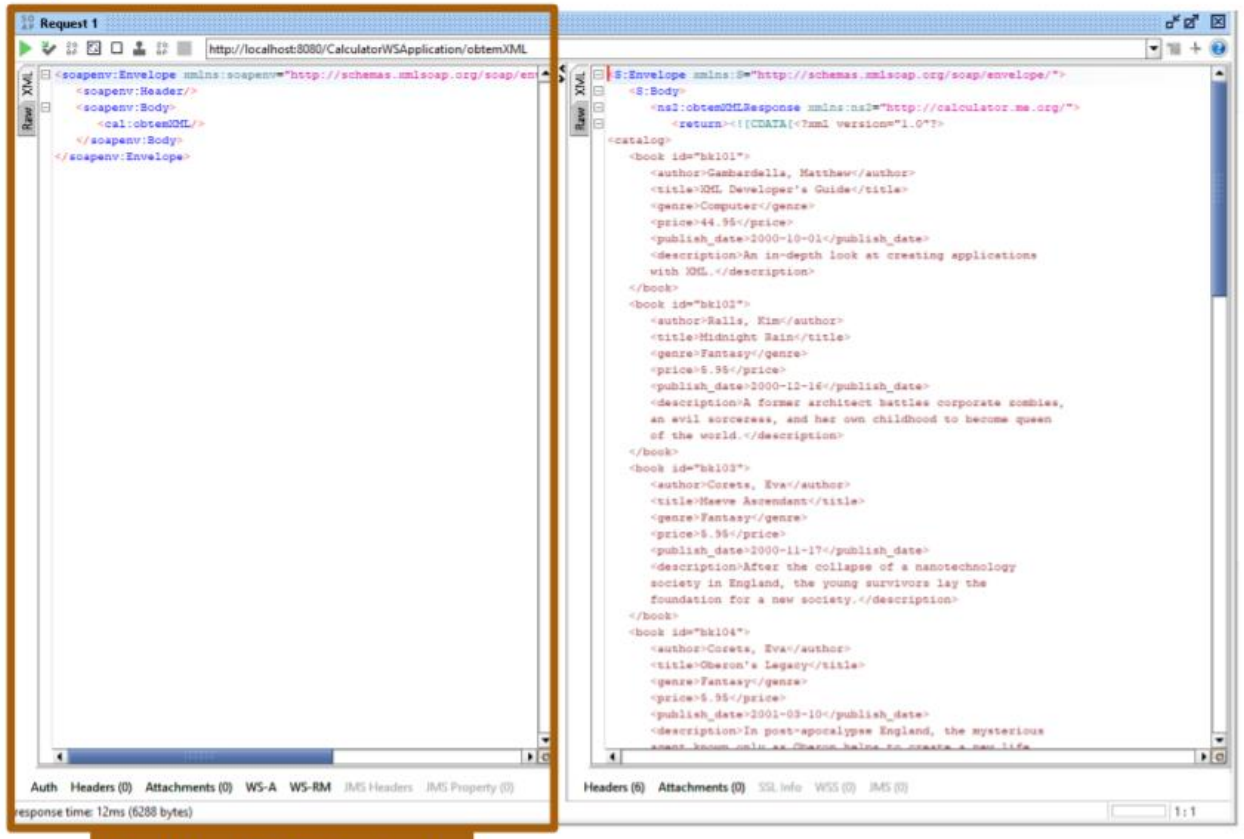
SOAP Response

```
<?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:obtemXMLResponse xmlns:ns2="http://calculator.me.org/">
            <return><?xml version="1.0"?>
<catalog>
   <book id="bk101">
      <author>Gambardella, Matthew</author>
      <title>XML Developer's Guide</title>
      <genre>Computer</genre>
```

SOAP – JAVA

Exemplo nº2 – Devolver o ficheiro Books.xml para leitura

Testes – SoapUI

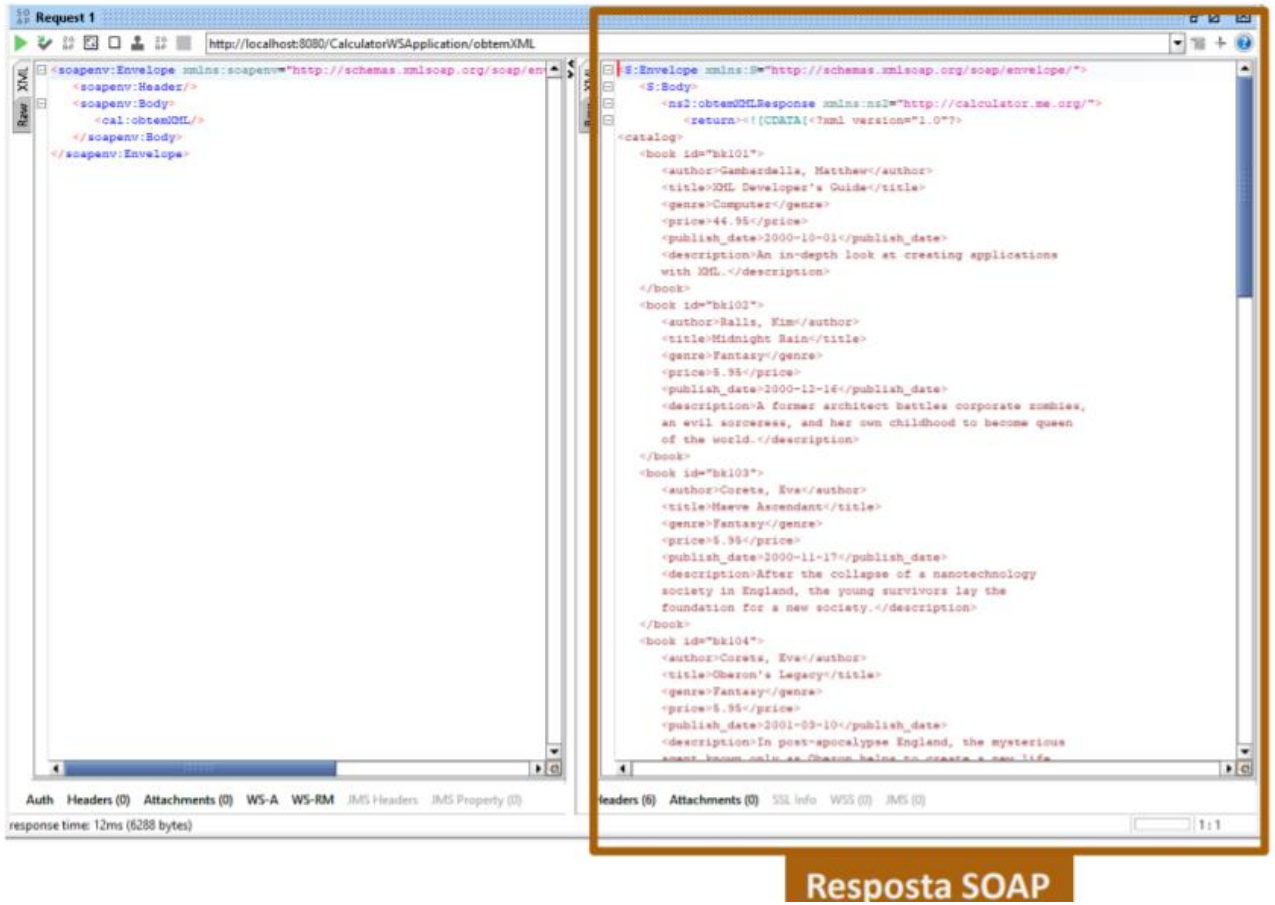


Pedido SOAP Enviado

SOAP – JAVA

Exemplo nº2 – Devolver o ficheiro Books.xml para leitura

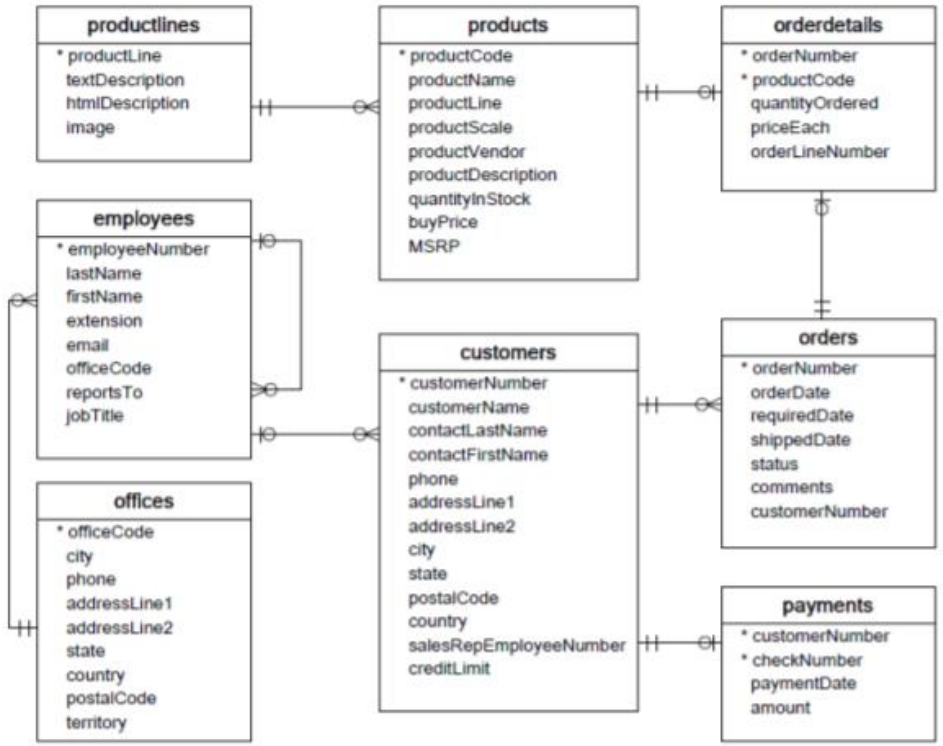
Testes – SoapUI



SOAP – JAVA

Exemplo nº3 – Devolver dados da base de dados

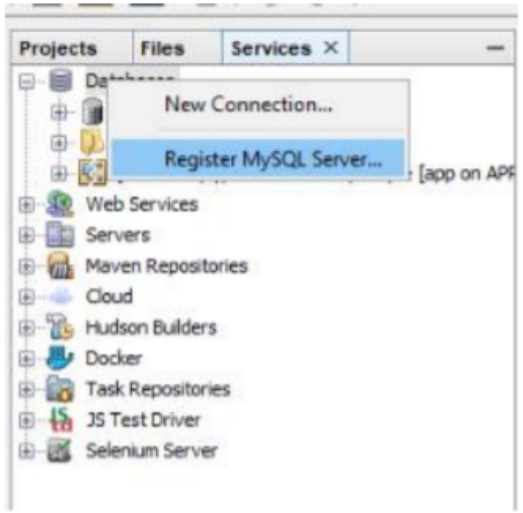
Os registos utilizados neste exemplo encontram-se disponíveis na base de dados de testes disponível em http://www.mysqltutorial.org/mysql-sample-database.aspx

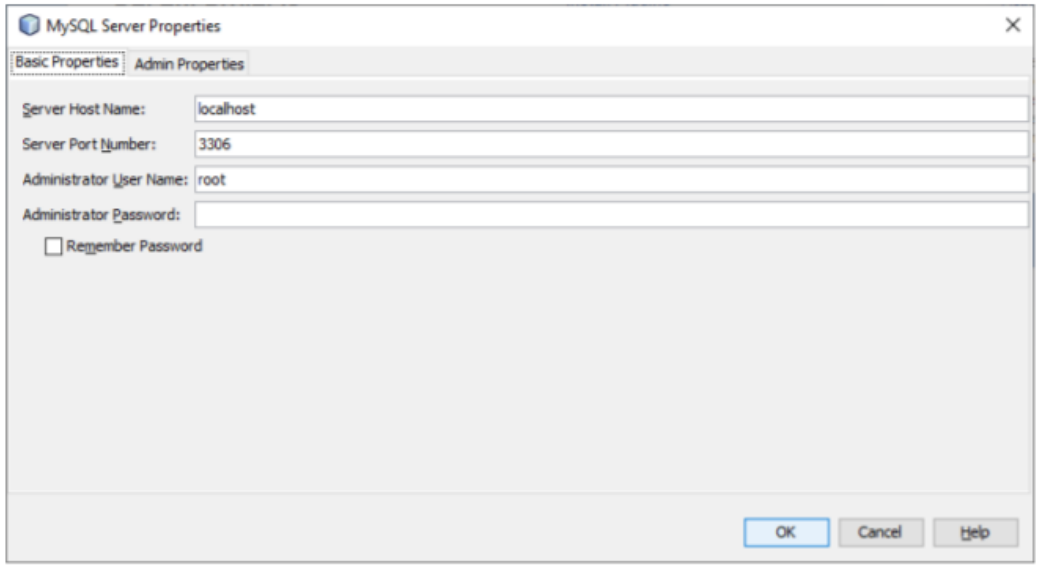


SOAP – JAVA

Exemplo nº3 – Devolver dados da base de dados

Inicialmente, é necessário criar a ligação à base de dados através do conetor JDBC.





https://www.youtube.com/watch?v=ET6BlRZY_Jw

SOAP – JAVA

Exemplo nº3 – Devolver dados da base de dados

Após criar a ligação, é necessário criar um projeto do tipo "Java Web Application". Nesse novo projeto, deve ser criada uma classe "session bean", na qual ficará o método que acede à base de dados para obter informação. Esse método, terá o seguinte código:

```
@Stateless
public class NewSessionBean {

    public Employees obtemEmployees(){

        Employees employees = new Employees();

        //String driver_JDBC = "com.mysql.cj.jdbc.Driver";
        String url_JDBC = "jdbc:mysql://localhost:3306/classicmodels";
        String utilizador = "root";
        String pass = "";
        try{
            Class.forName("com.mysql.jdbc.Driver").newInstance();
            Connection conn = DriverManager.getConnection(url_JDBC, utilizador, pass);
            Statement stmt = conn.createStatement();
            ResultSet result = stmt.executeQuery("SELECT * FROM employees");
            while(result.next())
                employees.getEmployees().add(Employee.criarEmployee(result.getInt(1), result.getString(3), result.getString(2)));
            conn.close();

        } catch(Exception e){
            System.out.println("Ocorreu um erro" + e.getMessage());
            employees.getEmployees().add(Employee.criarEmployee(1, "OCORREU UM ERRO - ", e.getMessage()));
        }

        return employees;

    }
}
```

SOAP – JAVA

Exemplo nº3 – Devolver dados da base de dados

São necessárias também as seguintes classes:

```
*/
@XmlRootElement(name="employee")
public class Employee implements Serializable{
    private int identificador;
    private String primeironome;
    private String ultimonome;

    public Employee() ()

    public Employee(int identificador, String primeironome, String ultimonome) {
        this.identificador = identificador;
        this.primeironome = primeironome;
        this.ultimonome = ultimonome;
    }

    public static Employee criarEmployee(int identificador, String primeironome, String ultimonome) {
        return new Employee(identificador, primeironome, ultimonome);
    }

    public int getIdentificador() {
        return identificador;
    }

    public void setIdentificador(int identificador) {
        this.identificador = identificador;
    }

    public String getPrimeironome() {
        return primeironome;
    }

    public void setPrimeironome(String primeironome) {
        this.primeironome = primeironome;
    }

    public String getUltimonome() {
        return ultimonome;
    }

    public void setUltimonome(String ultimonome) {
        this.ultimonome = ultimonome;
    }
```

SOAP – JAVA

Exemplo nº3 – Devolver dados da base de dados

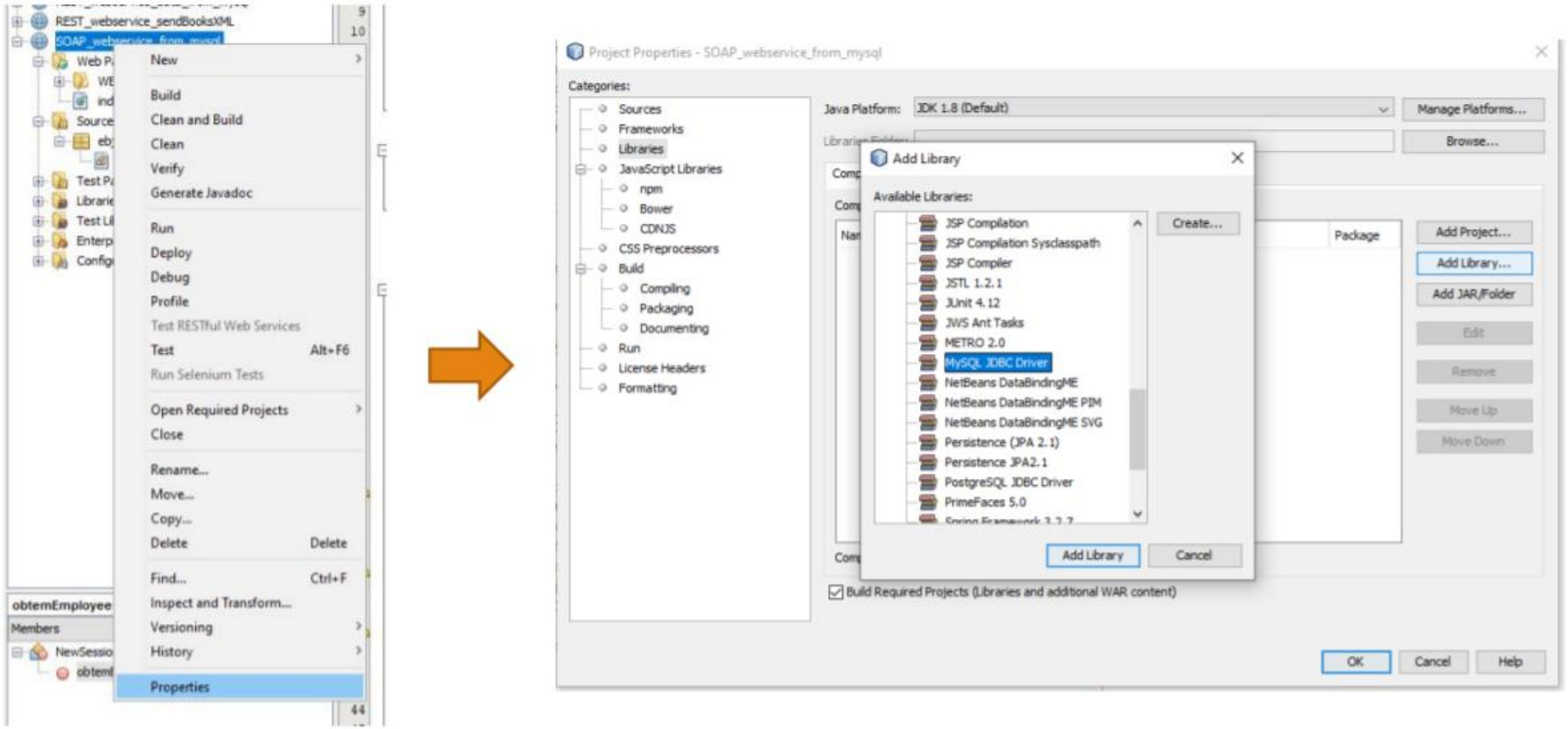
e:

```
@XmlRootElement(name = "employees")
public class Employees {
    private List<Employee> employees = new ArrayList<>();

    public Employees() {

    }

    public List<Employee> getEmployees() {
        return employees;
    }

    public void setEmployees(List<Employee> employees) {
        this.employees = employees;
    }


}
```

SOAP – JAVA

Exemplo nº3 – Devolver dados da base de dados

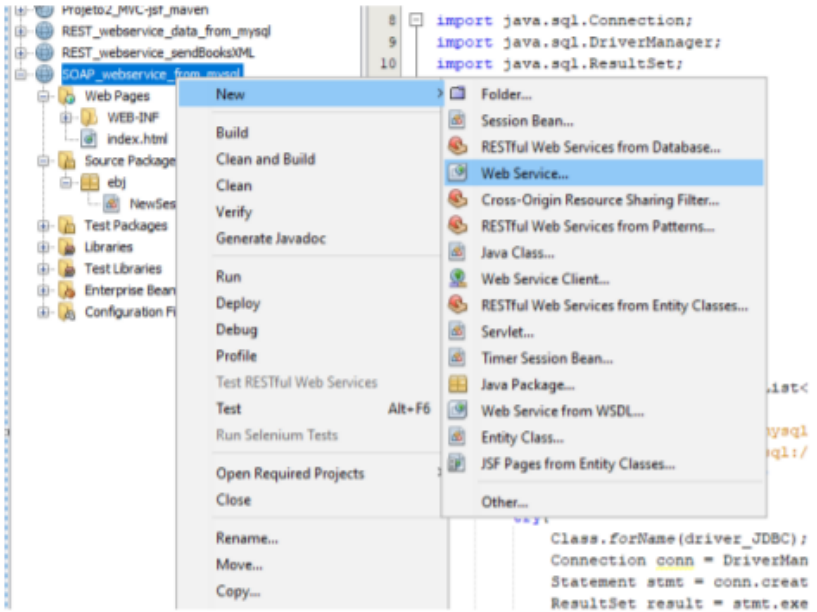
A libraria do Driver Mysql deve ser associada ao projeto

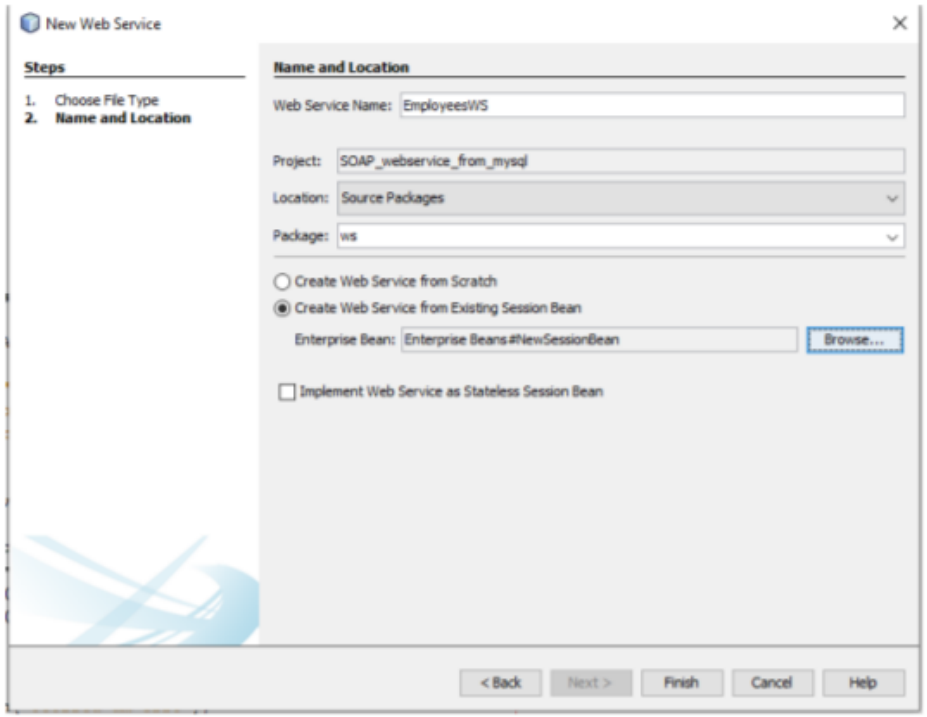


SOAP – JAVA

Exemplo nº3 – Devolver dados da base de dados

Após criada a ligação, session bean e adicionada a libraria, é necessário criar um novo web service para usar a session bean.

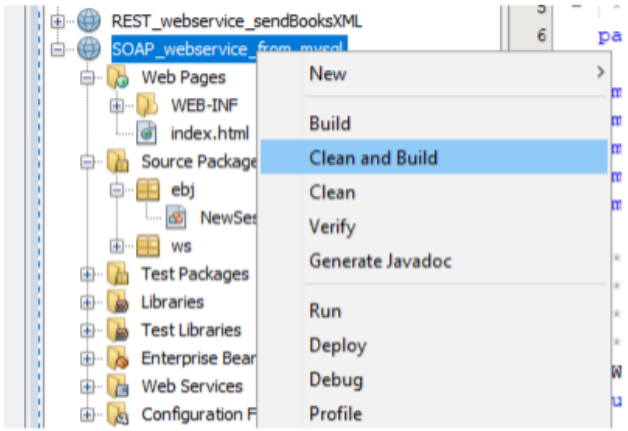


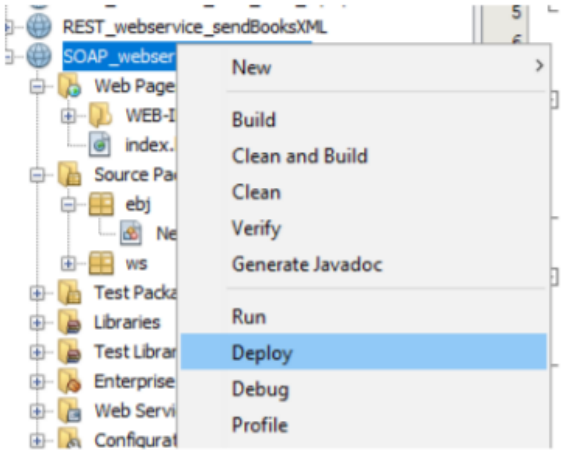


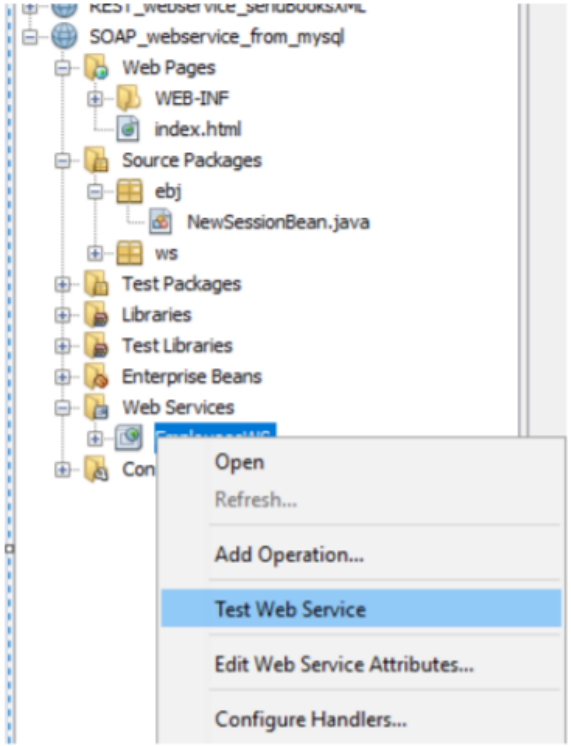
SOAP – JAVA

Exemplo nº3 – Devolver dados da base de dados

Agora, é necessário Construir o projeto através do "Build", e dar Deploy ao mesmo, para que possa ser testado.







Versão 1 – Documento em Atualização

SOAP – JAVA

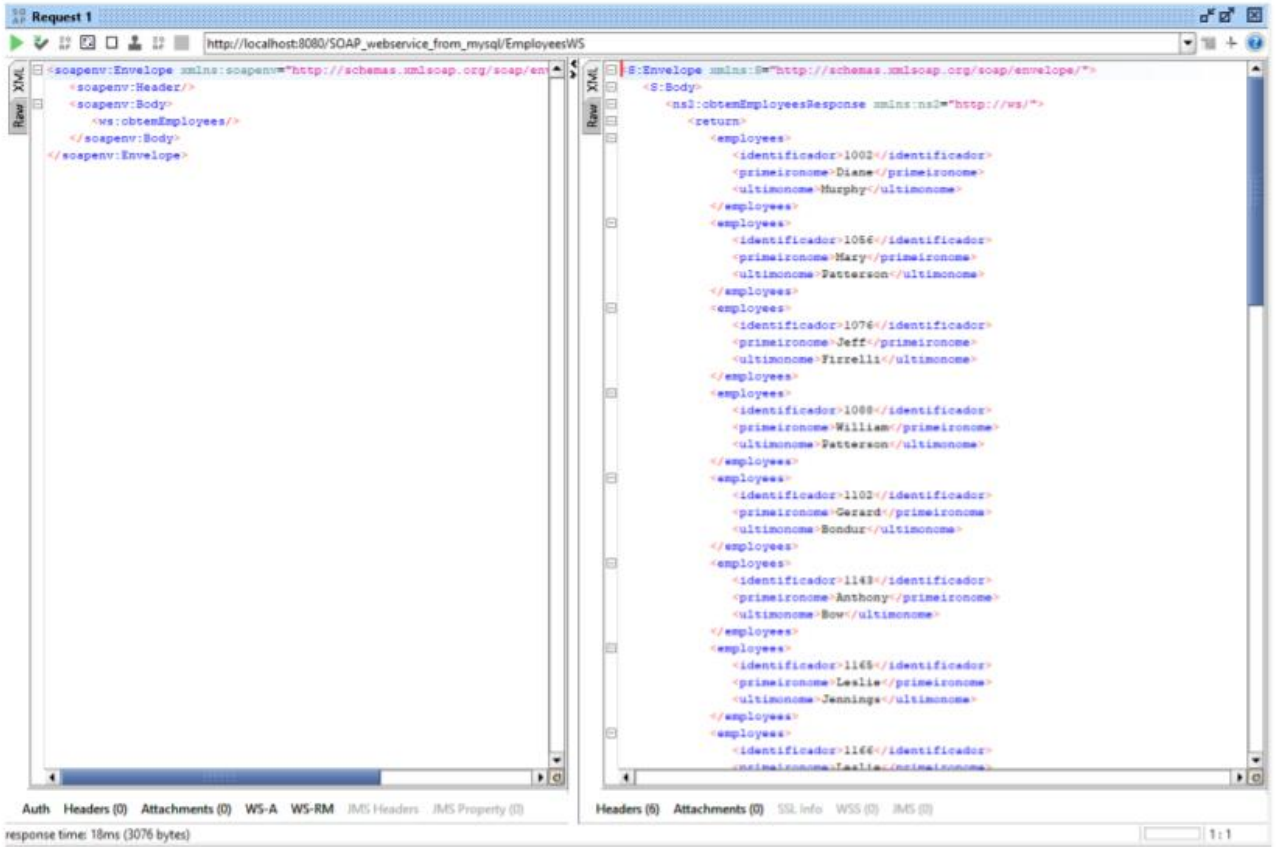
Exemplo nº3 – Devolver dados da base de dados

Testes – Browser (Opção "Test Web Service" do Netbeans)

SOAP – JAVA

Exemplo nº3 – Devolver dados da base de dados

Testes – SoapUI



SOAP – PHP

Exemplo nº1: Exemplo dado pelo sitepoint.com

Código Servidor (envio da informação + criação automática do ficheiro wsdl)

SendBooks.php | RecebeBooks.php

```php
<?php
require_once "nusoap.php";
function getProd($category) {
    if ($category == "books") {
        return join(",", array(
            "The WordPress Anthology",
            "PHP Master: Write Cutting Edge Code",
            "Build Your Own Website the Right Way"));
    }
    else {
        return "No products listed under that category";
    }
}

$server = new soap_server();
$server->configureWSDL('AddService', 'ObtemBooksString');
$server->wsdl->schemaTargetNamespace = 'ObtemBooksString';
$server->register("getProd",
    array(),
    array('return' => 'xsd:string')
);
$server->service(file_get_contents('php://input'));


?>
```

https://www.sitepoint.com/web-services-with-php-and-soap-1/

SOAP – PHP

Exemplo nº1: Exemplo dado pelo sitepoint.com

Código Cliente

SendBooks.php | RecebeBooks.php

RecebeBooks.php > ...

```php
<?php
require_once "nusoap.php";
$client = new nusoap_client("http://localhost/IS_WebServices/SendBooks.php");

$error = $client->getError();
if ($error) {
    echo "<h2>Constructor error</h2><pre>" . $error . "</pre>";
}

$result = $client->call("getProd", array("category" => "books"));

if ($client->fault) {
    echo "<h2>Fault</h2><pre>";
    print_r($result);
    echo "</pre>";
}
else {
    $error = $client->getError();
    if ($error) {
        echo "<h2>Error</h2><pre>" . $error . "</pre>";
    }
    else {
        echo "<h2>Books</h2><pre>";
        echo $result;
        echo "</pre>";
    }
}

?>
```

SOAP – PHP

Exemplo nº1: Exemplo dado pelo sitepoint.com

Teste – Broswer (Cliente php)

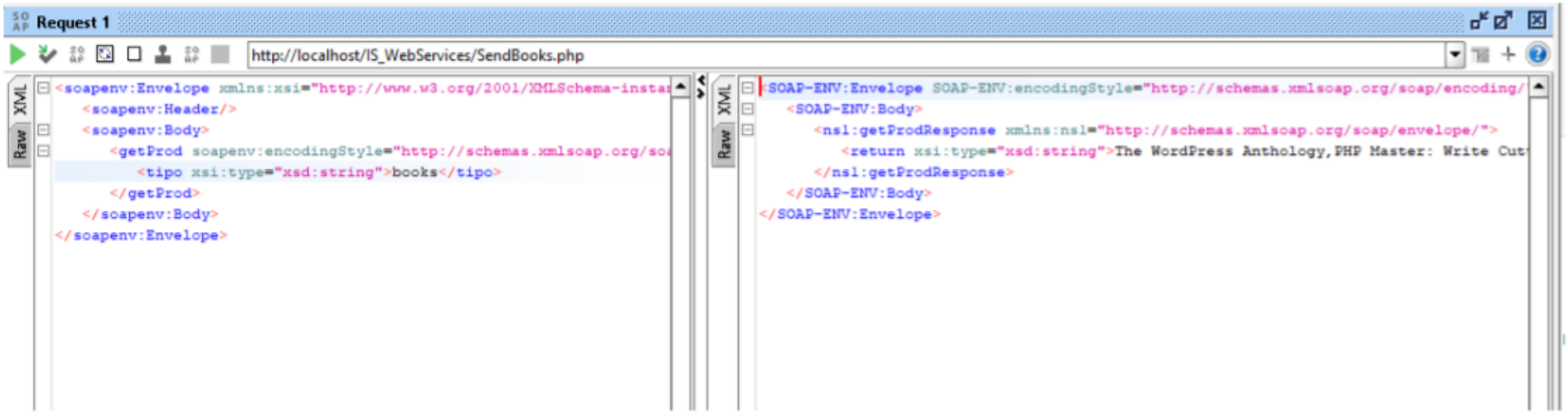
Books

The WordPress Anthology,PHP Master: Write Cutting Edge Code,Build Your Own Website the Right Way

SOAP – PHP

Exemplo nº1: Exemplo dado pelo sitepoint.com

Teste – SOAP UI



SOAP – PHP

Exemplo nº2: Devolver o ficheiro Books.xml para leitura

Código Servidor – Baseado no exemplo 1

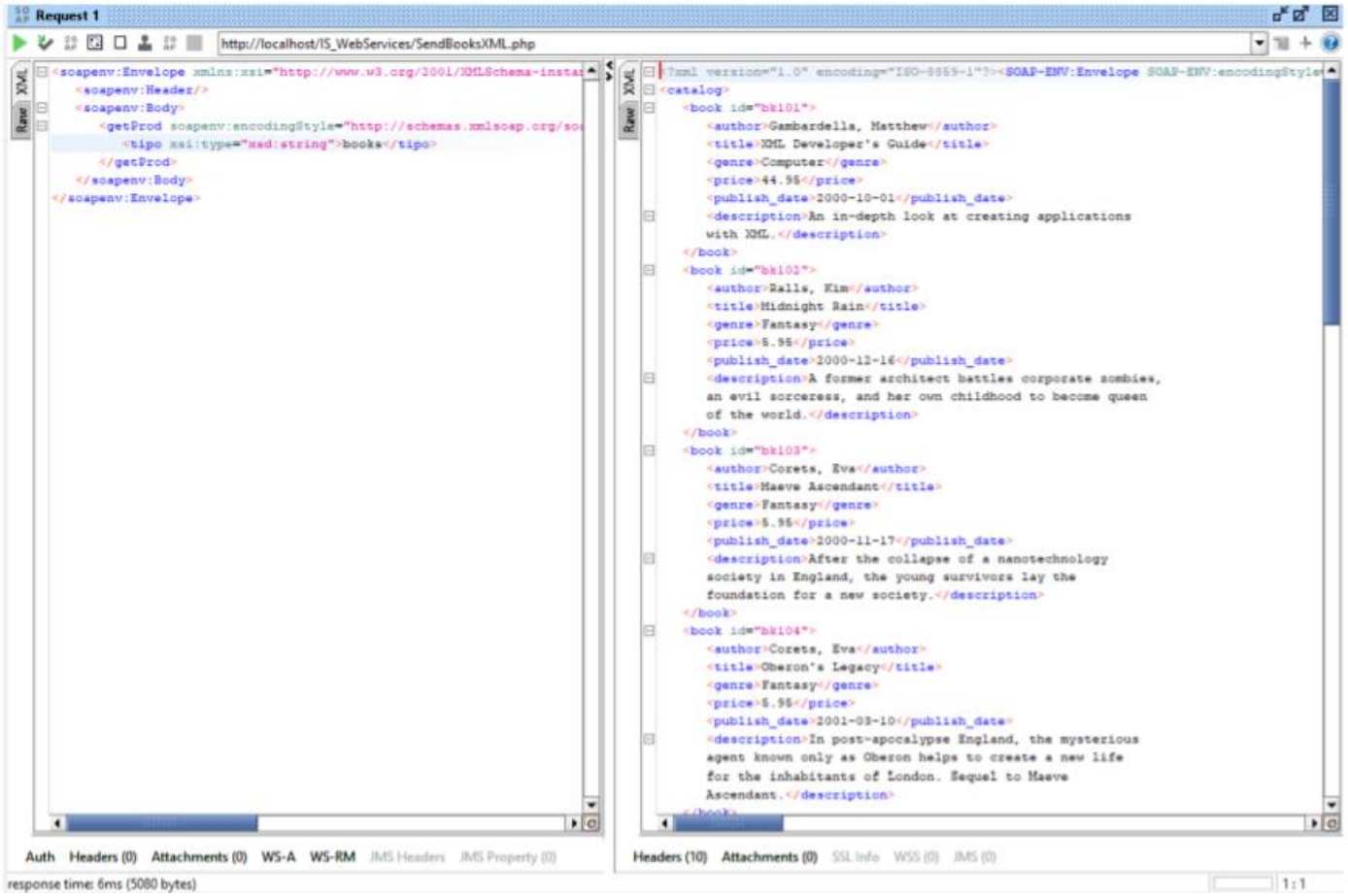
SendBooksXML.php > ...

```php
<?php
require_once "nusoap.php";
function getProd($category) {
    if ($category == "books") {
        return file_get_contents("books.xml");
    }
    else {
            return "No products listed under that category";
    }
}

$server = new soap_server();
$server->configureWSDL('AddService', 'ObtemBooksString');
$server->wsdl->schemaTargetNamespace = 'ObtemBooksString';
$server->register("getProd",
    array('tipo' => 'xsd:string'),
    array('return' => 'xsd:xml')
);
$server->service(file_get_contents('php://input'));
?>
```

SOAP – PHP

Exemplo nº2: Devolver o ficheiro Books.xml para leitura

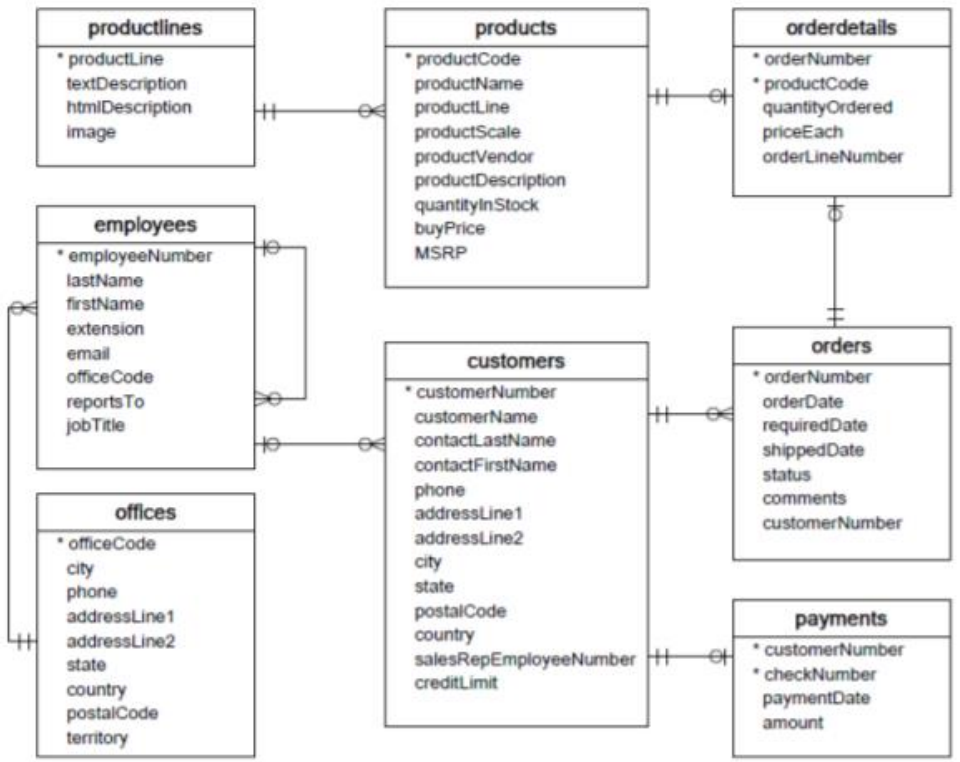
Teste – SOAP UI



SOAP – PHP

Exemplo nº3: Devolver registos da base de dados em XML

Os registos utilizados neste exemplo encontram-se disponíveis na base de dados de testes disponível em http://www.mysqltutorial.org/mysql-sample-database.aspx



SOAP – PHP

Exemplo nº3: Devolver registos da base de dados em XML

Código Servidor – Parte 1

```php
<?php
require_once "nusoap.php";
function getBooks() {
    /* máximo de registos */
    $number_of_posts = 10;
    /* número de registo de inicio a listar */
    $inicio = 1 * $number_of_posts;

    /* ligar à base de dados */
    $link = mysqli_connect('localhost','root','') or die('Cannot connect to the DB');
    mysqli_select_db($link, 'classicmodels') or die('Cannot select the DB');

    /* obtem dados */
    $query = "SELECT customerNumber, customerName, phone, addressLine1, city, country FROM customers LIMIT $inicio, $number_of_posts";
    $result = mysqli_query($link, $query) or die('Errant query:  '.$query);

    $arr = array();
    if(mysqli_num_rows($result)) {
        while($linha = mysqli_fetch_assoc($result)) {
            $arr[] = array('customers'=>$linha);
        }
    }
```

SOAP – PHP

Exemplo nº3: Devolver registos da base de dados em XML

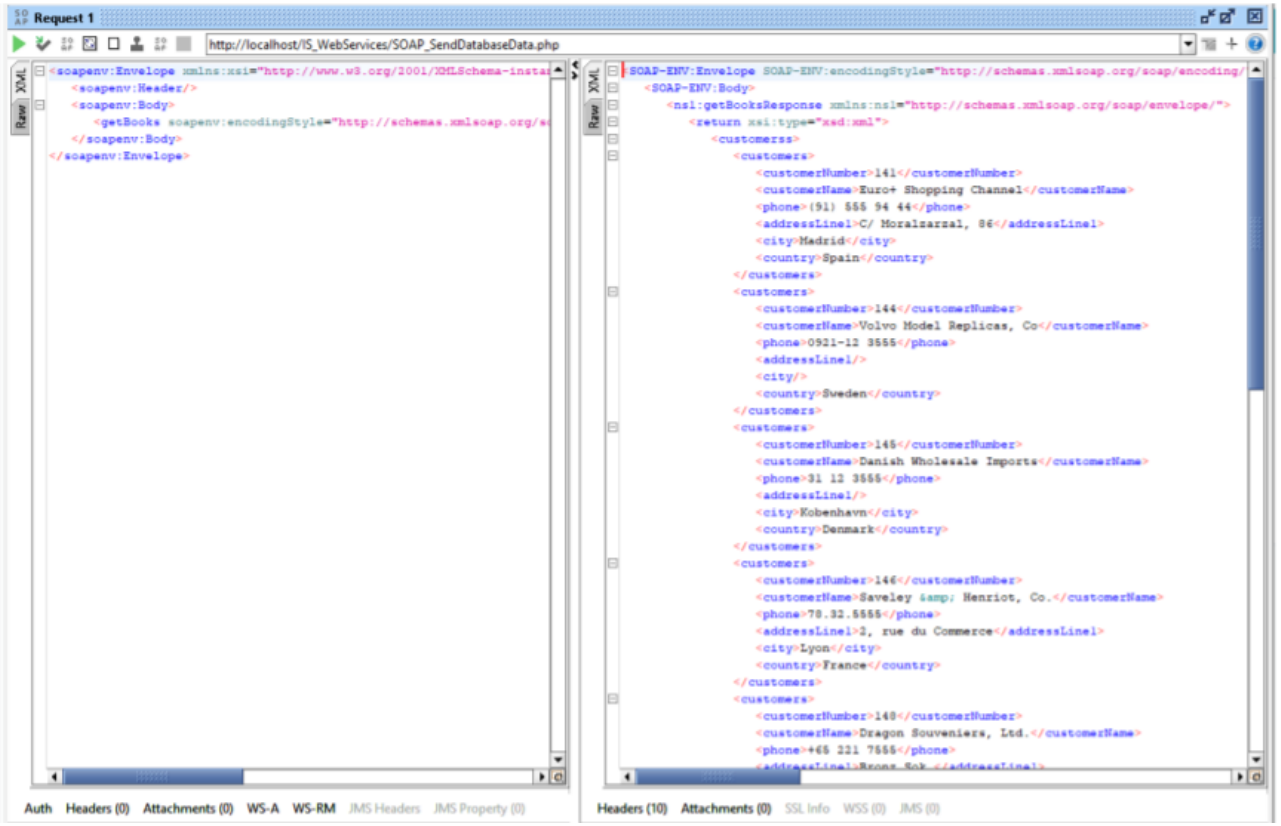
Código Servidor – Parte 2

```php
    $stringXML = "";

    header('Content-type: text/xml');
    $stringXML .= '<customerss>'; // tag inicial do documento
    foreach($arr as $index => $cliente) { // percorre array obtido da base de dados
        if(is_array($cliente)) {
            foreach($cliente as $key => $value) { // cada $cliente é um registo da base de dados representado num array de 1 elemento | obtem esse elemento para $value
                $stringXML .= '<'.$key.'>'; // inicia tag
                if(is_array($value)) {
                    foreach($value as $tag => $val) { // percorre todos os elementos de cada registo obtido da base de dados
                        $stringXML .= '<'.$tag.'>'.htmlentities($val).'</'.$tag.'>'; // echo a cada elemento
                    }
                }
                $stringXML .= '</'.$key.'>'; // termina a tag
            }
        }
    }
    $stringXML .= '</customerss>';

    return $stringXML;
}

$server = new soap_server();
$server->configureWSDL('AddService', 'ObtemBooksString');
$server->wsdl->schemaTargetNamespace = 'ObtemBooksString';
$server->register("getBooks", array(), array('return' => 'xsd:xml'));
$server->service(file_get_contents('php://input'));
?>
```

SOAP – PHP

Exemplo nº3: Devolver registos da base de dados em XML

Teste – SOAP UI



REST – JAVA

Exemplo nº1: Devolver o ficheiro Books.xml para leitura

Código – Parte 1 (Utiliza o DOM Parser)

```
@Path("obtemXML")
public class ObtemXML {

    @Context
    private UriInfo context;

    public ObtemXML() {}

    @GET
    @Produces(MediaType.APPLICATION_XML)
    public String getXml()    {
        try {
            File xmlFile = new File("D:\\IPVC\\5ºS. - Integração de Sistemas\\Trabalhos Práticos\\TP1 - Webservices SOAP e REST\\JAVA\\REST_webservice\\books.xml");
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
            DocumentBuilder builder = factory.newDocumentBuilder();
            Document doc = builder.parse(xmlFile);

            String xml = "";
            xml += parseWholeXML(doc.getDocumentElement());

            return xml;
        } catch (SAXException ex) {
            return "ERRO: " + ex.getMessage();
        } catch (IOException ex) {
            return "ERRO: " + ex.getMessage();
        } catch (ParserConfigurationException ex) {
            return "ERRO: " + ex.getMessage();
        }
    }
```

https://netbeans.org/kb/docs/websvc/rest.html

https://www.mkyong.com/java/how-to-read-xml-file-in-java-dom-parser/

REST – JAVA

Exemplo nº1: Devolver o ficheiro Books.xml para leitura

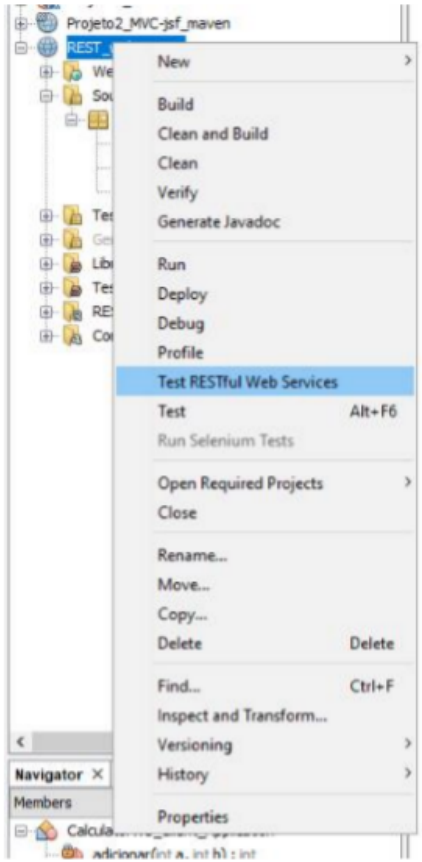
Código – Parte 2

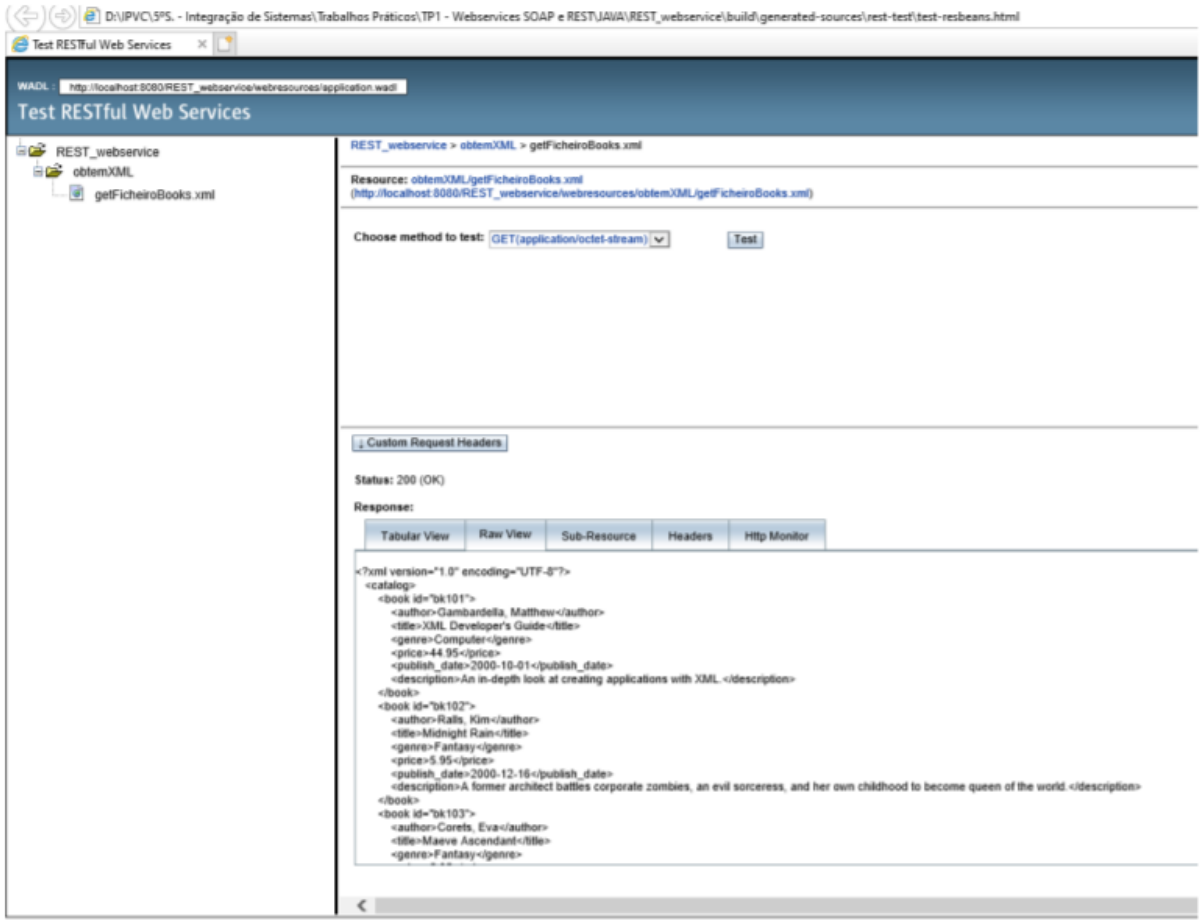
```
private static String parseWholeXML(Node startingNode){
    NodeList childNodes = startingNode.getChildNodes();
    String s = "";
    //inicializa tag
    s += "<" + startingNode.getNodeName() + ">\n";

    // percorre todos os filhos da tag dada
    for(int i=0; i<childNodes.getLength(); i++){
        Node childNode = childNodes.item(i);
        if(childNode.getNodeType() == Node.ELEMENT_NODE){
            // se possuir tag dentro da tag atual, executa recursivamente parseWholeXML() com a nova tag encontrada
            s += parseWholeXML(childNode);
        }else{
            if(!childNode.getTextContent().trim().isEmpty()){
                //adiciona texto que está dentro da tag
                s += childNode.getTextContent() + "\n";
            }
        }
    }
    // finaliza tag
    s += "</" + startingNode.getNodeName() + ">\n";

    return s;
}
```

REST – JAVA

Exemplo nº1: Devolver o ficheiro Books.xml para leitura

Teste – Opção "Test RESTful Web Services" do Netbeans

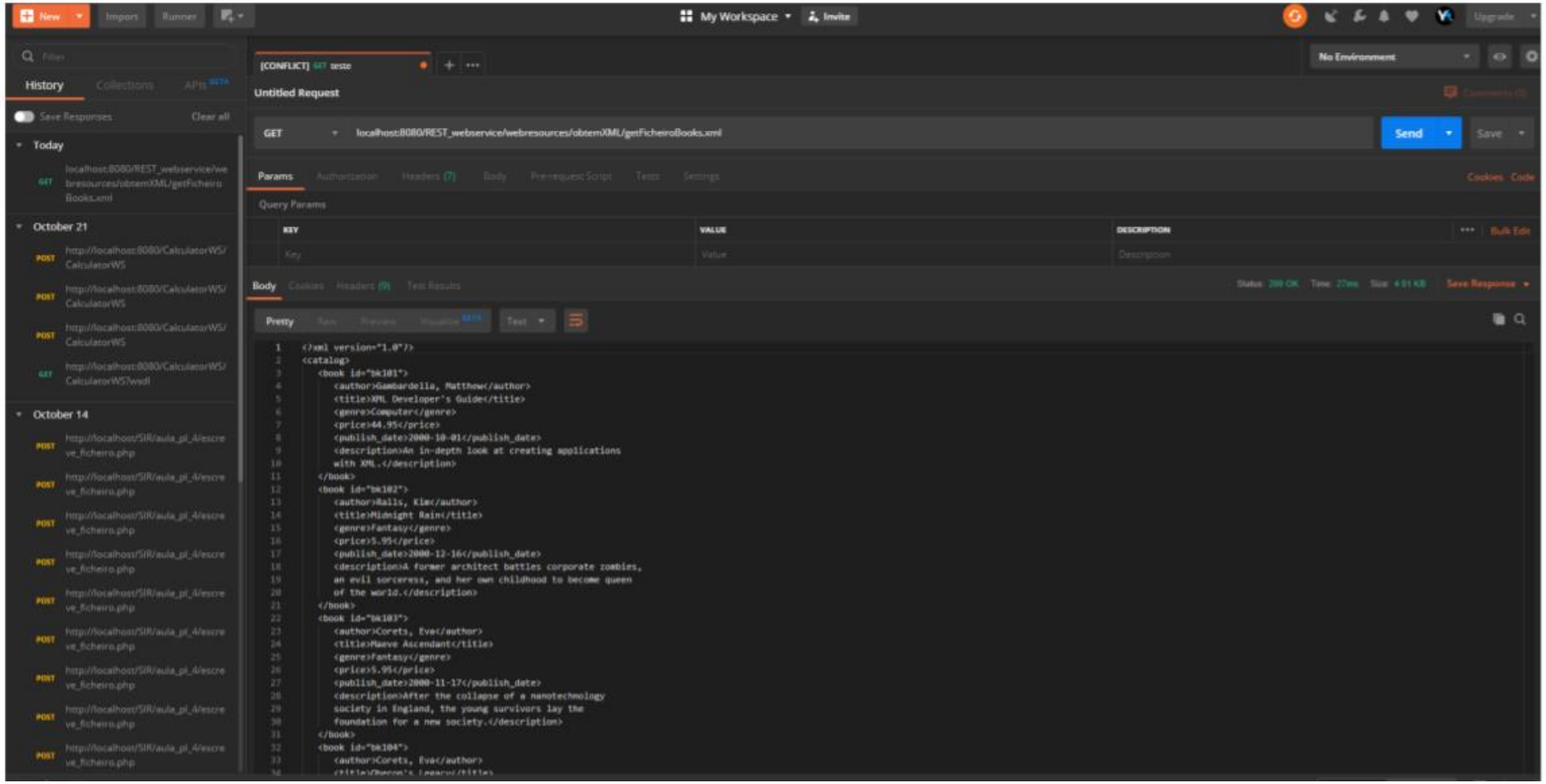




REST – JAVA

Exemplo nº1: Devolver o ficheiro Books.xml para leitura

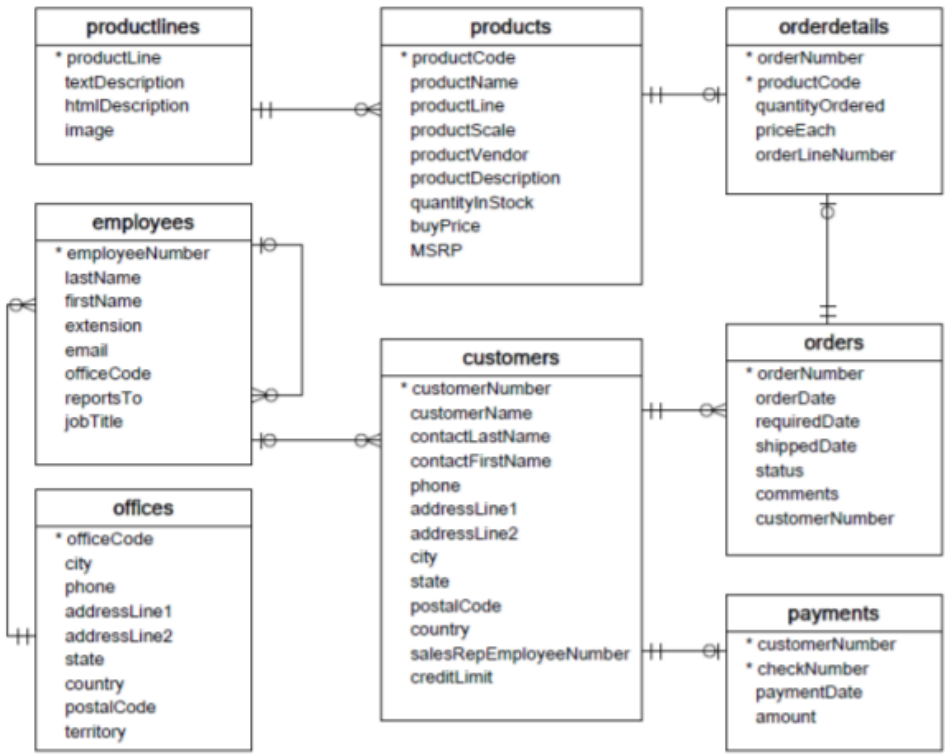
Teste – POSTMAN



REST – JAVA

Exemplo nº2: Acesso à base de dados e devolução de registos em XML

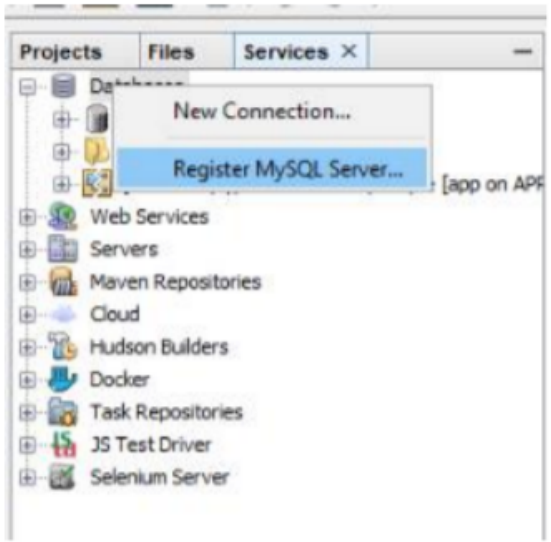
Os registos utilizados neste exemplo encontram-se disponíveis na base de dados de testes disponível em http://www.mysqltutorial.org/mysql-sample-database.aspx

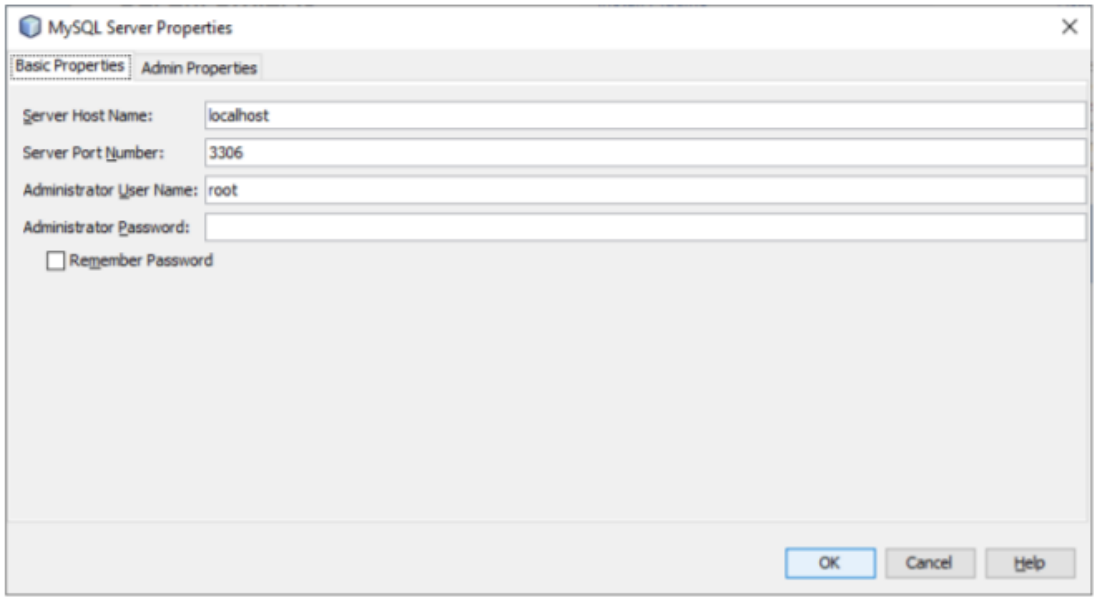


REST – JAVA

Exemplo nº2: Acesso à base de dados e devolução de registos em XML

1 – Associar a base de dados mysql ao netbeans.

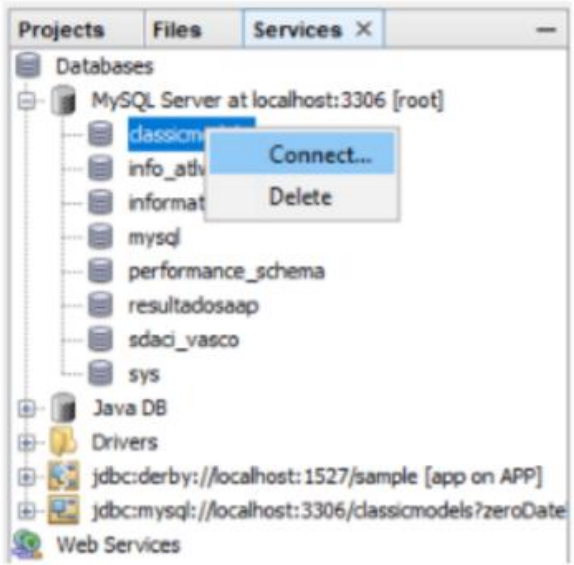




REST – JAVA

Exemplo nº2: Acesso à base de dados e devolução de registos em XML

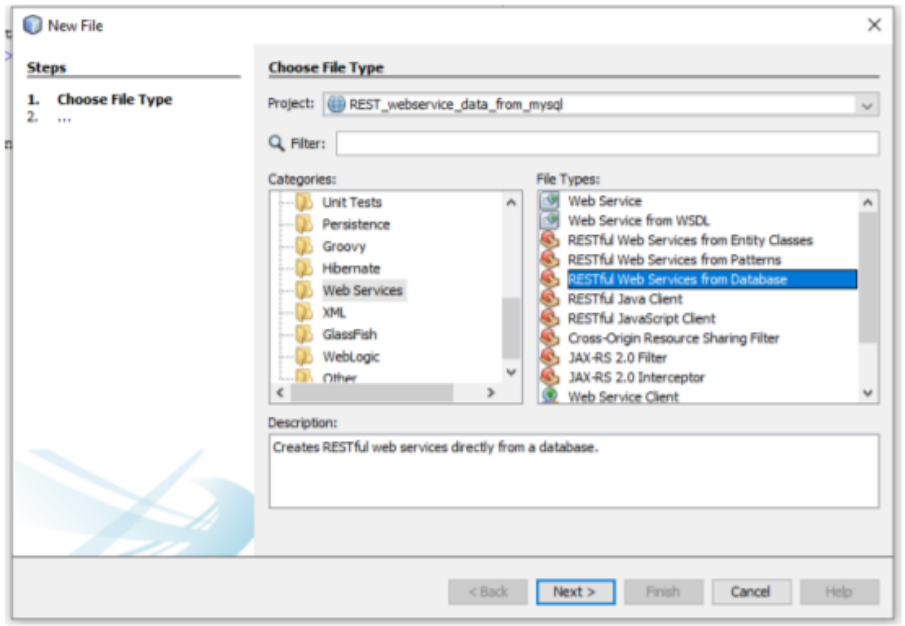
2 – Adicionar a conexão à base de dados.

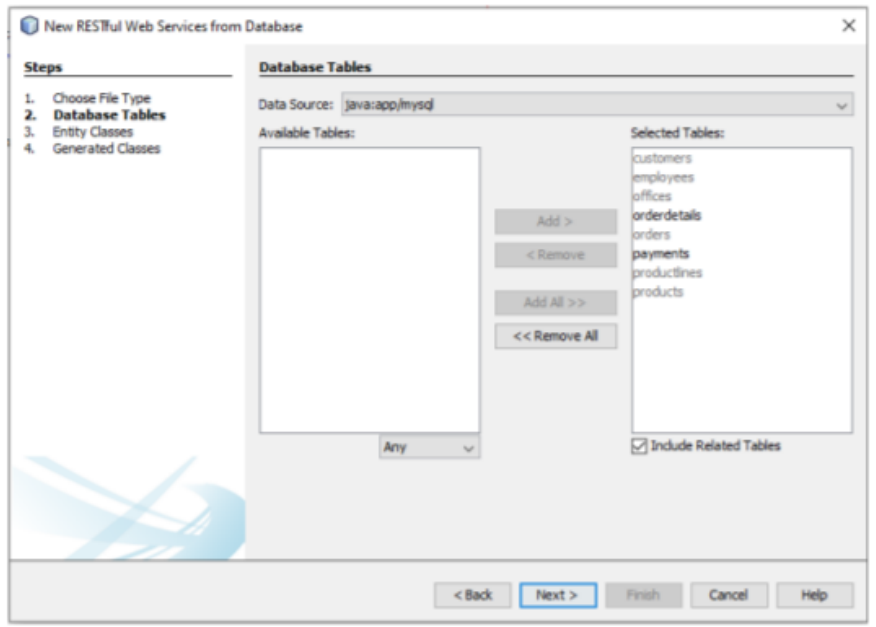


REST – JAVA

Exemplo nº2: Acesso à base de dados e devolução de registos em XML

3 – Criar uma Java Web > Web Application e gerar classes a partir da base de dados





REST – JAVA

Exemplo nº2: Acesso à base de dados e devolução de registos em XML

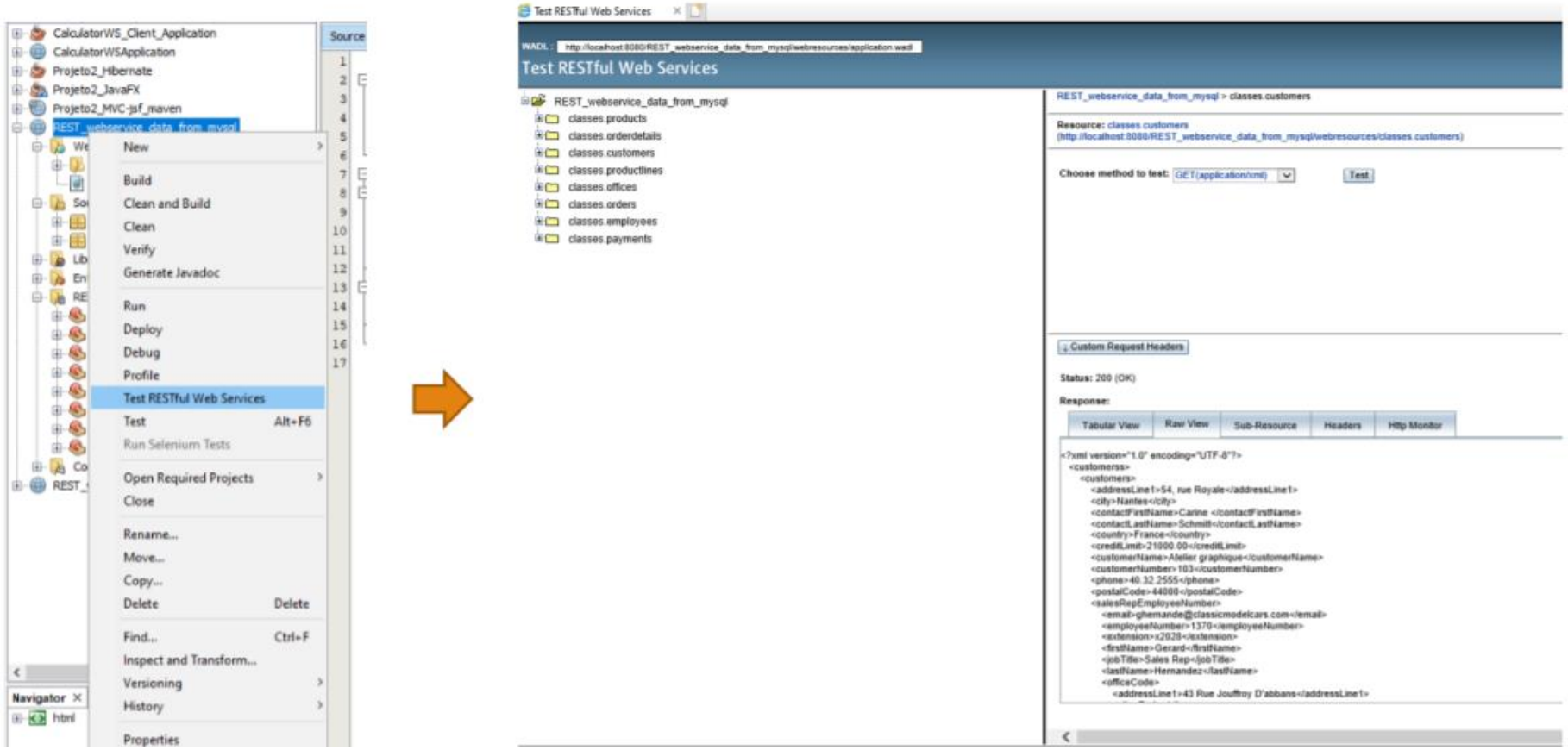
Excerto do código gerado, permite utilizar os web services com XML ou JSON

```
@GET
@Path("{id}")
@Produces({MediaType.APPLICATION_XML, MediaType.APPLICATION_JSON})
public Customers find(@PathParam("id") Integer id) {
    return super.find(id);
}

@GET
@Override
@Produces({MediaType.APPLICATION_XML, MediaType.APPLICATION_JSON})
public List<Customers> findAll() {
    return super.findAll();
}

@GET
@Path("{from}/{to}")
@Produces({MediaType.APPLICATION_XML, MediaType.APPLICATION_JSON})
public List<Customers> findRange(@PathParam("from") Integer from, @PathParam("to") Integer to) {
    return super.findRange(new int[]{from, to});
}
```

REST – JAVA

Exemplo nº2: Acesso à base de dados e devolução de registos em XML

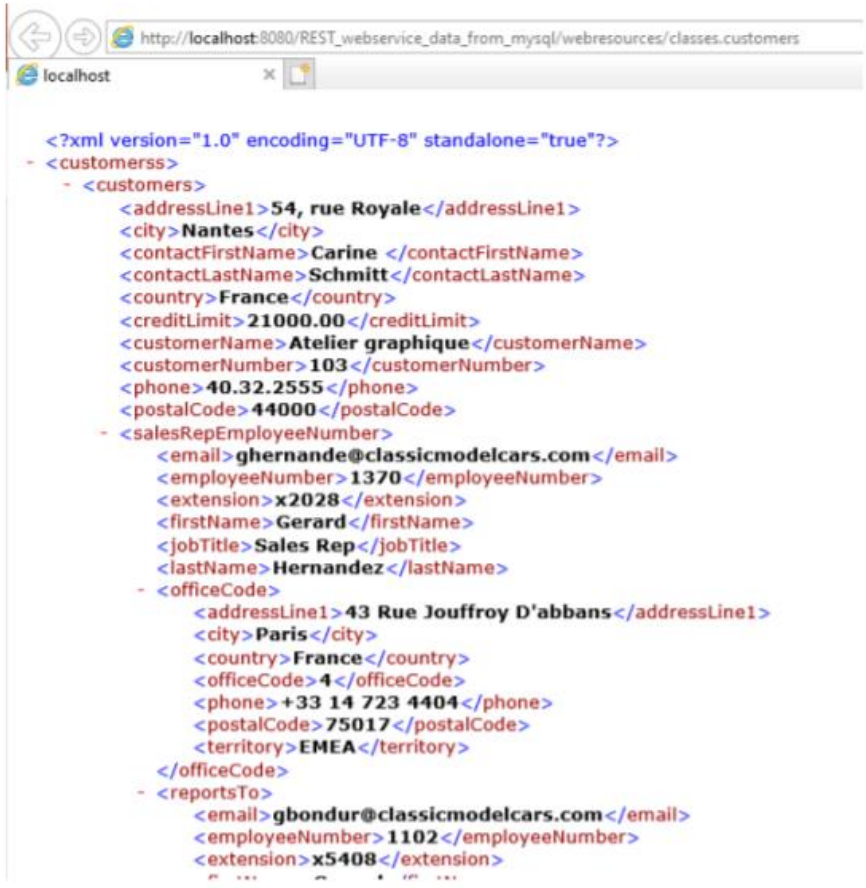
TESTE: Browser (Página gerada pelo netbeans)



REST – JAVA

Exemplo nº2: Acesso à base de dados e devolução de registos em XML

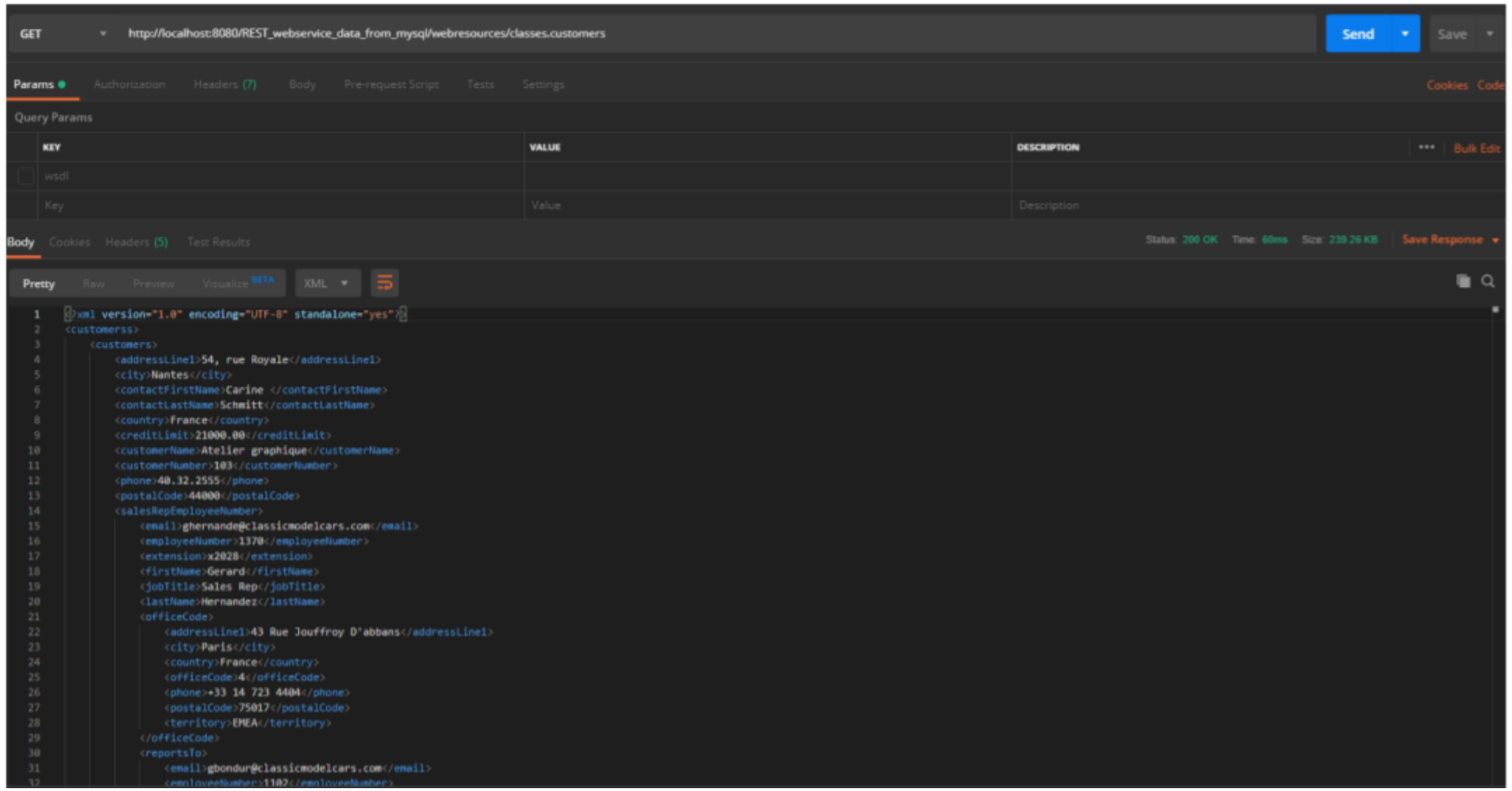
TESTE: Browser (Link direto)



REST – JAVA

Exemplo nº2: Acesso à base de dados e devolução de registos em XML

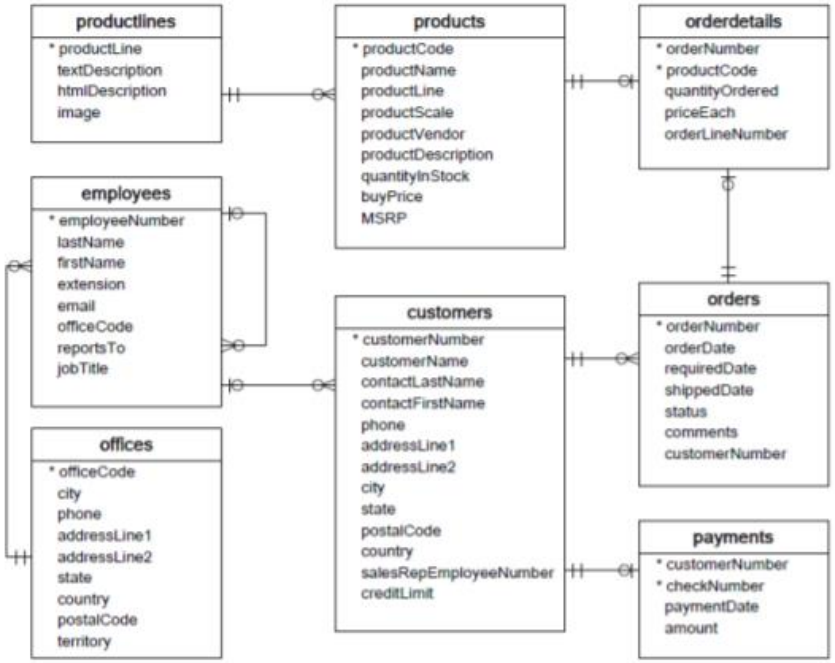
TESTE: POSTMAN



REST – PHP

Exemplo nº1: Acesso à base de dados e devolução de registos em XML

Os registos utilizados neste exemplo encontram-se disponíveis na base de dados de testes disponível em http://www.mysqltutorial.org/mysql-sample-database.aspx. O web service foi criado seguindo o tutorial em: https://davidwalsh.name/web-service-php-mysql-xml-json



https://davidwalsh.name/web-service-php-mysql-xml-json

REST – PHP

Exemplo nº1: Acesso à base de dados e devolução de registos em XML

Código:

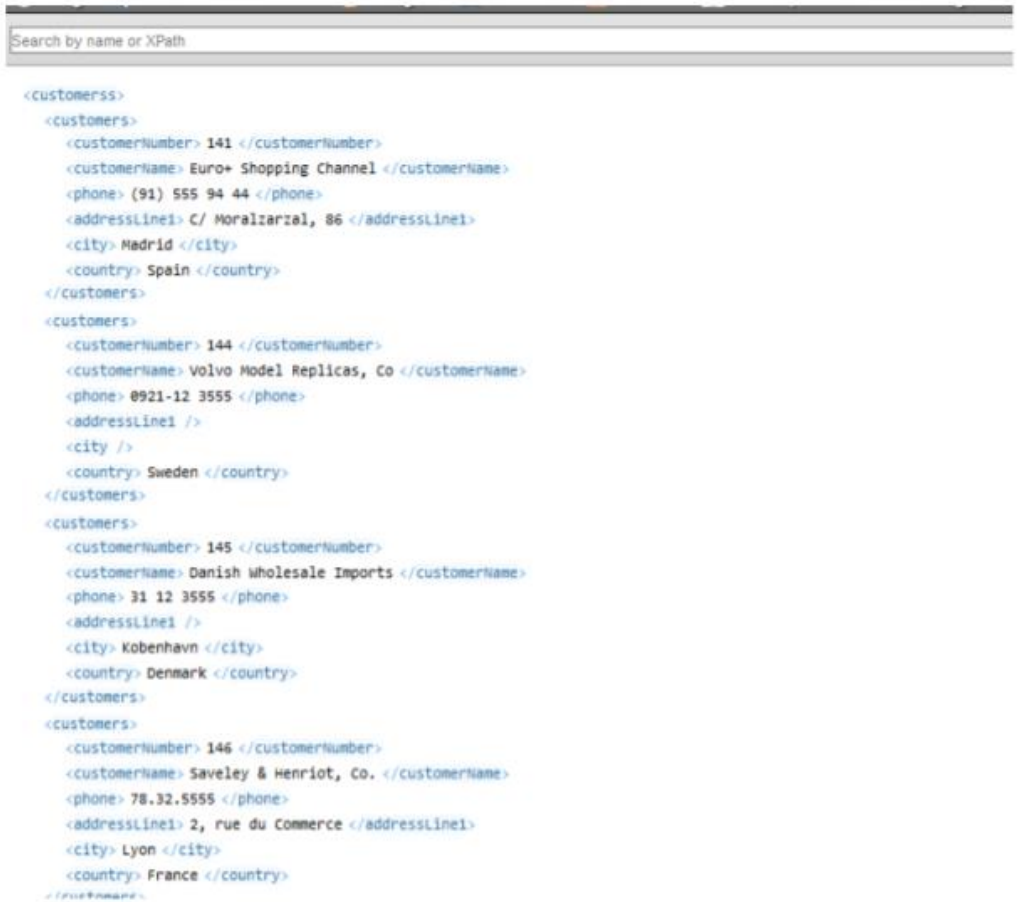
REST_get_customers_from_mysql_xml.php

```php
<?php
    /* máximo de registos */
    $number_of_posts = 10;
    /* número de registo de início a listar */
    $inicio = 1 * $number_of_posts;

    /* ligar à base de dados */
    $link = mysqli_connect('localhost','root','') or die('Cannot connect to the DB');
    mysqli_select_db($link, 'classicmodels') or die('Cannot select the DB');

    /* obtem dados */
    $query = "SELECT customerNumber, customerName, phone, addressLine1, city, country FROM customers LIMIT $inicio, $number_of_posts";
    $result = mysqli_query($link, $query) or die('Errant query:  '.$query);

    $arr = array();
    if(mysqli_num_rows($result)) {
        while($linha = mysqli_fetch_assoc($result)) {
            $arr[] = array('customers'=>$linha);
        }
    }

    header('Content-type: text/xml');
    echo '<customerss>'; // tag inicial do documento
    foreach($arr as $index => $cliente) { // percorre array obtido da base de dados
        if(is_array($cliente)) {
            foreach($cliente as $key => $value) { // cada $cliente é um registo da base de dados representado num array de 1 elemento | obtem esse elemento para $value
                echo '<',$key,'>'; // inicia tag
                if(is_array($value)) {
                    foreach($value as $tag => $val) { // percorre todos os elementos de cada registo obtido da base de dados
                        echo '<',$tag,'>',htmlentities($val),'</',$tag,'>'; // echo a cada elemento
                    }
                }
                echo '</',$key,'>'; // termina a tag
            }
        }
    }
    echo '</customerss>';


    // -- termina a ligação à base de dados
    mysqli_close($link);
?>
```

REST – PHP

Exemplo nº1: Acesso à base de dados e devolução de registos em XML

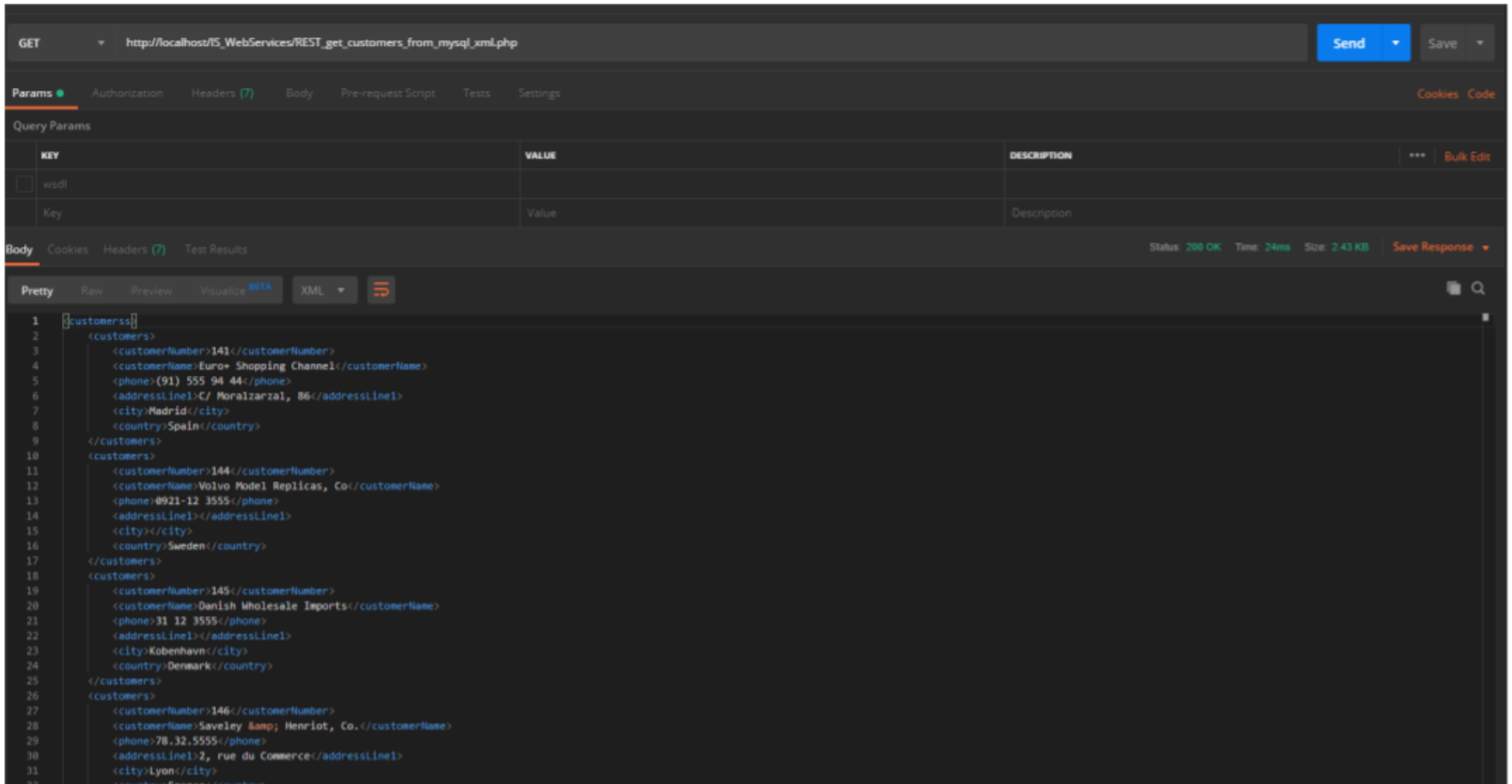
TESTE: Browser (Aceder ao link do serviço)



REST – PHP

Exemplo nº1: Acesso à base de dados e devolução de registos em XML

TESTE: POSTMAN



REST – PHP

Exemplo nº2: Enviar conteúdo do ficheiro Books.xml

Código:

```php
<?php
    $xml = file_get_contents("books.xml");

    header('Content-type: text/xml');
    echo $xml;
?>
```

REST – PHP

Exemplo nº2: Enviar conteúdo do ficheiro Books.xml

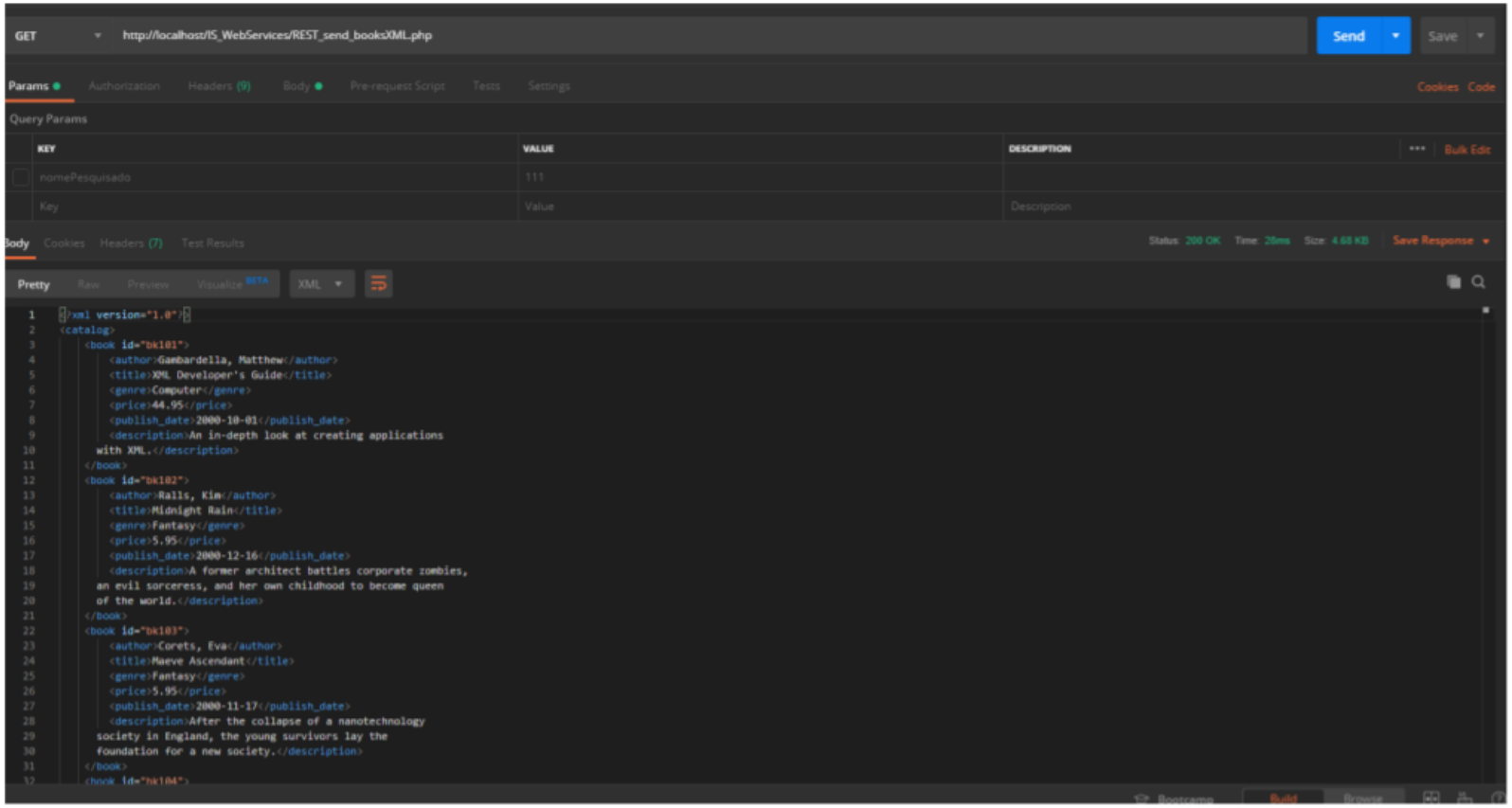
TESTE: Browser (Aceder ao link do serviço)

```
<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 Lor
    </description>
  </book>
  <book id="bk105">
```

REST – PHP

Exemplo nº2: Enviar conteúdo do ficheiro Books.xml

TESTE: Postman



JAVA – SOAP - DOM

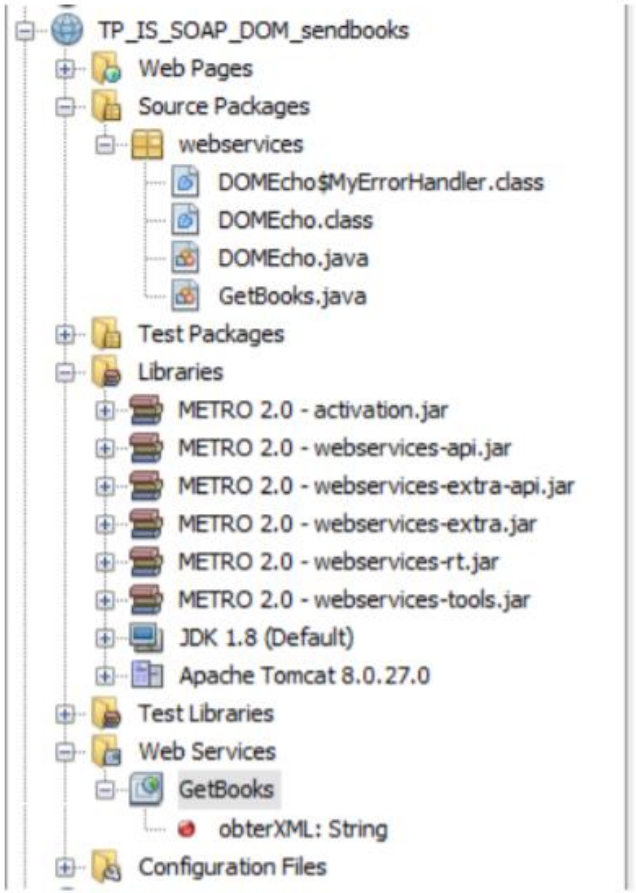
Dados Necessários

De modo a que os seguintes Web Services funcionem corretamente, é necessário:

- Necessário colocar o ficheiro books.xml em C:\testar\
- Necessário colocar o ficheiro po.xml em C:\testar\ ***(apenas para os ws para o parser JAXB)***

JAVA – SOAP - DOM

Estrutura do Projeto Criado



DOMEcho.java – ficheiro que trata do parsing DOM

GetBooks.java – classe do web service que invoca o DOMEcho

JAVA – SOAP - DOM

Parte do Código DOMEcho.java

```
public class DOMEcho {
    static final String outputEncoding = "UTF-8";
    private PrintWriter out;
    private int indent = 0;
    private final String basicIndent = "  ";
    static final String JAXP_SCHEMA_LANGUAGE = "http://java.sun.com/xml/jaxp/properties/schemaLanguage";
    static final String W3C_XML_SCHEMA = "http://www.w3.org/2001/XMLSchema";
    static final String JAXP_SCHEMA_SOURCE = "http://java.sun.com/xml/jaxp/properties/schemaSource";

    static String a = new String();

    DOMEcho(PrintWriter out) {
        this.out = out;
    }

    private void printlnCommon(Node n) {
        a += " nodeName=\"" + n.getNodeName() + "\"";
        String val = n.getNamespaceURI();
        if (val != null) {
            a += " uri=\"" + val + "\"";
        }
        val = n.getPrefix();
        if (val != null) {
            a += " pre=\"" + val + "\"";
        }
        val = n.getLocalName();
        if (val != null) {
            a += " local=\"" + val + "\"";
        }
        val = n.getNodeValue();
        if (val != null) {
            a += " nodeValue=";
            if (val.trim().equals("")) {
                // Whitespace
                a += "[WS]";
            } else {
                a += "\"" + n.getNodeValue() + "\"";
            }
        }
        a += "\n";
    }

    private void outputIndentation() {
        for (int i = 0; i < indent; i++) {
```

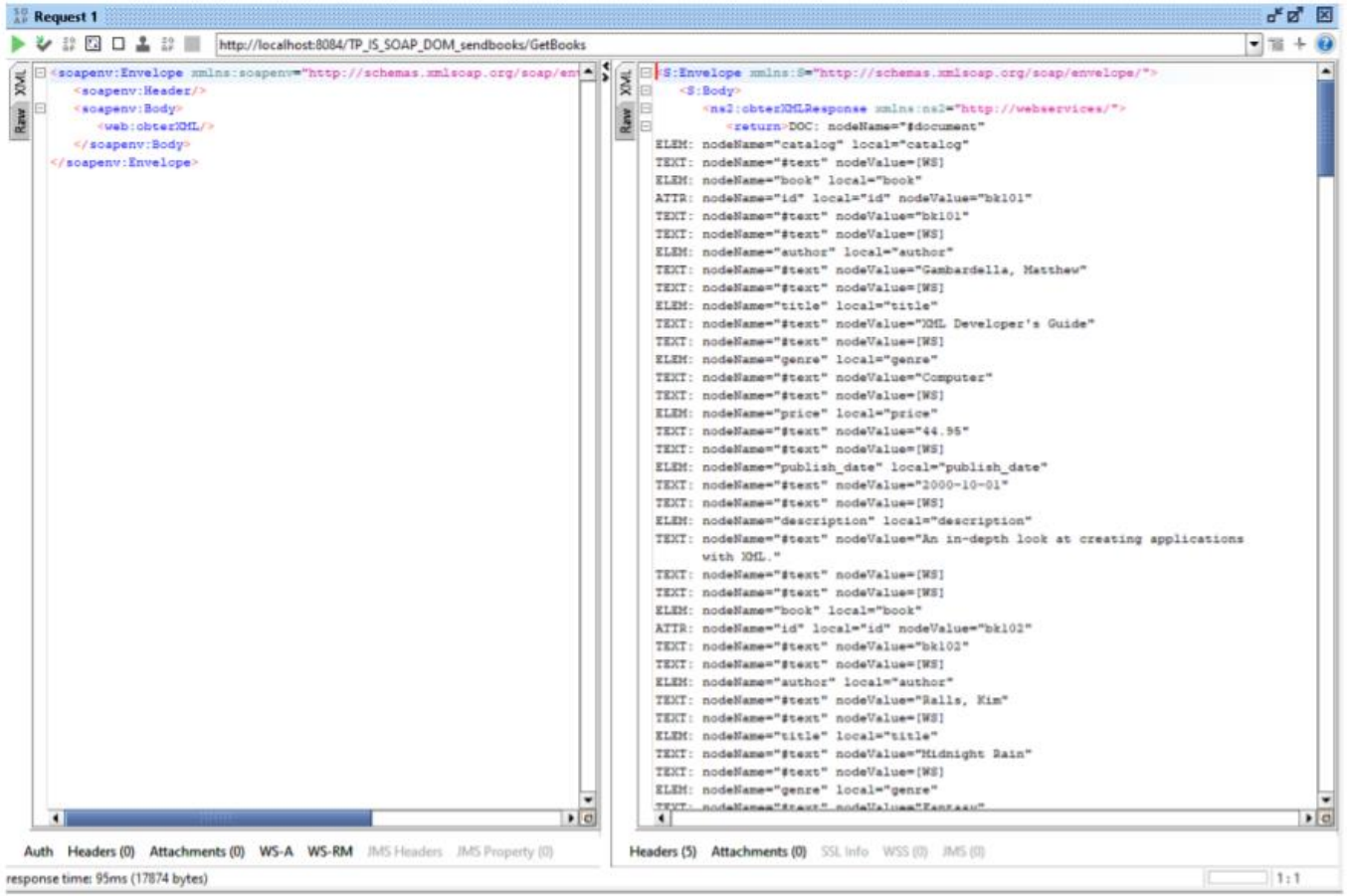
JAVA – SOAP - DOM

Código do WS GetBooks.java

```
 */
package webservices;

import java.util.logging.Level;
import java.util.logging.Logger;
import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.WebParam;

/**
 *
 * @author Vasco Alves
 */
@WebService(serviceName = "GetBooks")
public class GetBooks {

    @WebMethod(operationName = "obterXML")
    public String hello() {
        try {
            return DOMEcho.obterStringXML("C:\\testar\\books.xml");
        } catch (Exception ex) {
            return "Ocorreu um erro! -> " + ex.getMessage();
        }
    }
}
```

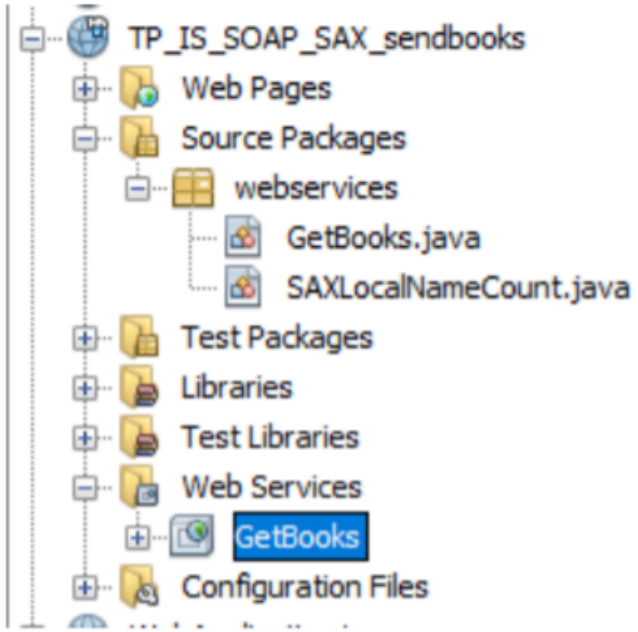
JAVA – SOAP - DOM

Teste SoapUI



JAVA – SOAP - SAX

Estrutura do Projeto



SAXLocalNameCount.java – ficheiro que conta o número de ocorrências dos nós XML usando o SAX

GetBooks.java – classe do web service que invoca o SAXLocalNameCount. Nesta classe é identificado o ficheiro XML a usar

JAVA – SOAP - SAX

Parte do Código de SAXLocalNameCount.java

```
public static String obterDados(String filename) throws Exception {
    boolean dtdValidate = false;
    boolean xsdValidate = false;
    String schemaSource = null;

    if (filename == null) {
        filename = "C:\\testar\\books.xml";
    }

    SAXParserFactory spf = SAXParserFactory.newInstance();
    spf.setNamespaceAware(true);
    spf.setValidating(dtdValidate || xsdValidate);
    SAXParser saxParser = spf.newSAXParser();
    if (schemaSource != null) {
        saxParser.setProperty(JAXP_SCHEMA_SOURCE, new File(schemaSource));
    }

    XMLReader xmlReader = saxParser.getXMLReader();
    xmlReader.setContentHandler(new SAXLocalNameCount());
    xmlReader.setErrorHandler(new MyErrorHandler(System.err));
    xmlReader.parse(convertToFileURL(filename));

    return teste;
}
```

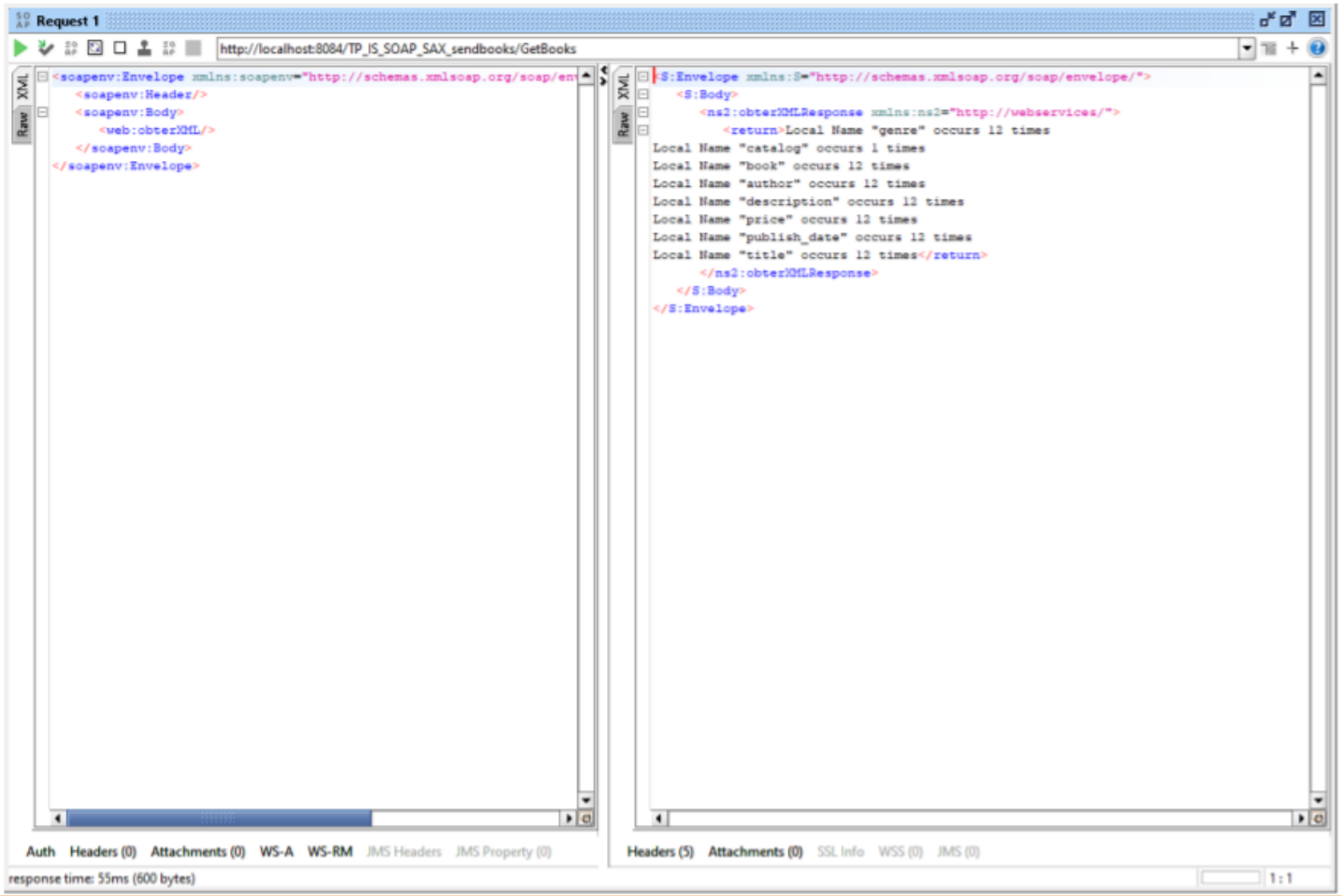
JAVA – SOAP - SAX

Código do WS GetBooks.java

```
package webservices;

import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.WebParam;

/**
 *
 * @author Vasco Alves
 */
@WebService(serviceName = "GetBooks")
public class GetBooks {

    @WebMethod(operationName = "obterXML")
    public String obterXML() {
        try{
            return SAXLocalNameCount.obterDados("C:\\testar\\books.xml");
        }catch(Exception e){
            return "ocorreu um erro...";
        }
    }
}
```

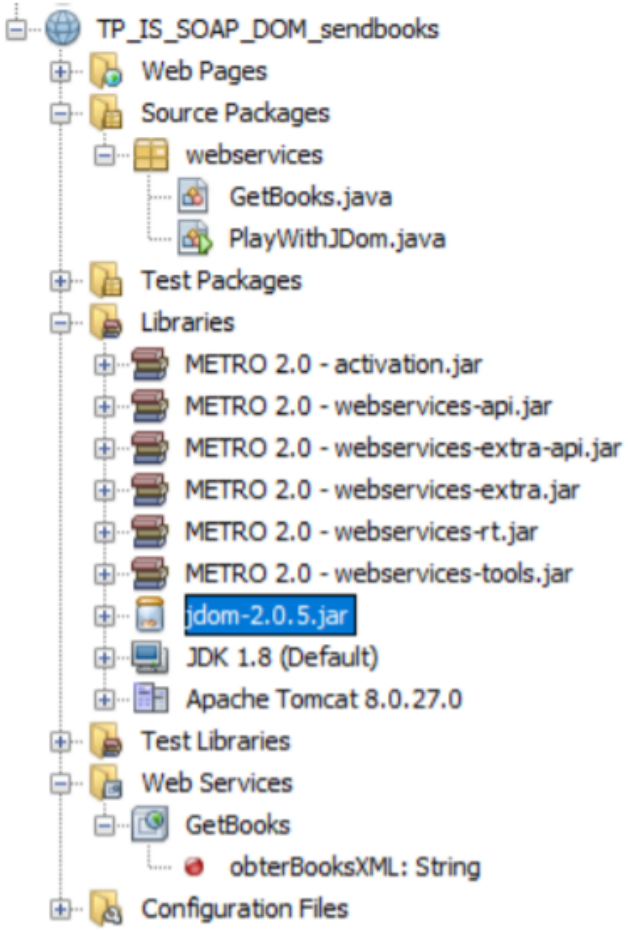
JAVA – SOAP - SAX

Teste SoapUI



JAVA – SOAP - JDOM

Estrutura do Projeto



PlayWithJDom.java – ficheiro que chama o JDOM

GetBooks.java – classe do web service que invoca o PlayWithJDOM. Nesta classe é identificado o ficheiro XML a usar

Necessário adicionar jdom-2.0.5.jar às librarias do projeto

JAVA – SOAP - JDOM

Parte do Código de PlayWithJDOM.java

```java
import java.io.File;
import java.io.IOException;
import java.io.*;
import org.jdom2.Document;
import org.jdom2.Element;
import org.jdom2.JDOMException;
import org.jdom2.input.SAXBuilder;
import org.jdom2.output.DOMOutputter;
import org.jdom2.output.XMLOutputter;

public class PlayWithJDom  {
    XMLOutputter printer = null;

    public PlayWithJDom () {
        printer = new XMLOutputter();
    }

    public String parseAndPlay(File anXmlDocFile){
        SAXBuilder parser = new SAXBuilder();
        try {
            Document doc = parser.build(anXmlDocFile);
            return this.printJDom(doc);
        } catch (JDOMException ex) {
            ex.printStackTrace();
        } catch (IOException ex) {
            ex.printStackTrace();
        }
        return "Ocorreu um erro";
    }

    public String printJDom(Document aJDOMdoc) throws IOException{
        return printer.outputString(aJDOMdoc);
    }
```

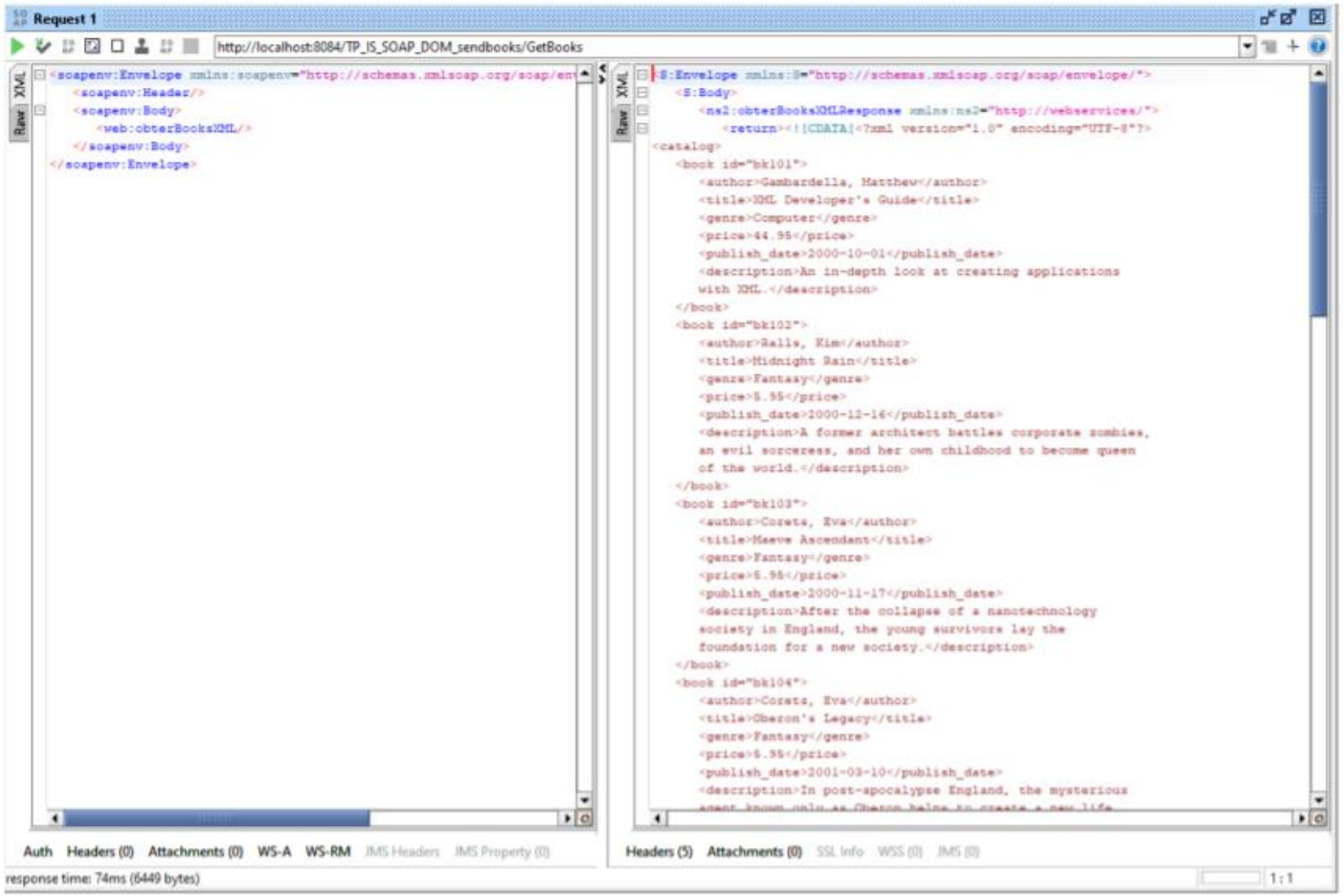
JAVA – SOAP - JDOM

Código do WS GetBooks.java

```
package webservices;

import java.io.File;
import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.WebParam;

/**
 *
 * @author Vasco Alves
 */
@WebService(serviceName = "GetBooks")
public class GetBooks {

    @WebMethod(operationName = "obterBooksXML")
    public String hello() {
        PlayWithJDom play = new PlayWithJDom();
        String XML = play.parseAndPlay(new File("C:\\testar\\books.xml"));
        return XML;
    }
}
```

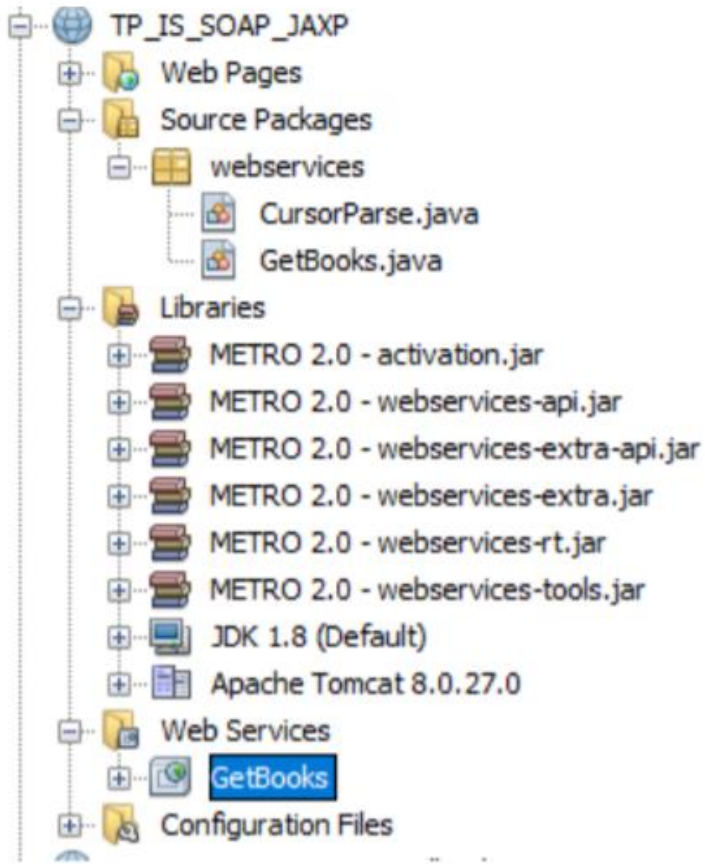
JAVA – SOAP - JDOM

Teste SoapUI



JAVA – SOAP - JAXP

Estrutura do Projeto



CursorParse.java – ficheiro que executa o parser JAXP no ficheiro XML

GetBooks.java – classe do web service que invoca o CursorParse. Nesta classe é identificado o ficheiro XML a usar

JAVA – SOAP - JAXP

Parte do Código de CursorParse.java

```
public static String obterDados(String filename) throws Exception {
    File file = null;
    try {
        if(filename == null)
            filename = "C:\\testar\\books.xml";
        file = new File(filename);
        if (!file.exists()) {
            System.out.println("File: " + filename + " does not exist.");
            System.exit(0);
        }
    } catch (ArrayIndexOutOfBoundsException aioobe){
        printUsage();
        System.exit(0);
    } catch (Exception ex){
        printUsage();
        ex.printStackTrace() ;
    }

    XMLInputFactory xmlif = null ;
    try{
        xmlif = XMLInputFactory.newInstance();
        xmlif.setProperty(XMLInputFactory.IS_REPLACING_ENTITY_REFERENCES,Boolean.TRUE);
        xmlif.setProperty(XMLInputFactory.IS_SUPPORTING_EXTERNAL_ENTITIES,Boolean.FALSE);
        xmlif.setProperty(XMLInputFactory.IS_COALESCING , Boolean.FALSE);
    }catch(Exception ex){
        ex.printStackTrace();
    }
    dados += "FACTORY: " + xmlif + "\n";
    dados += "filename = "+ filename + "\n\n";
    long starttime = System.currentTimeMillis() ;

    try{
        XMLStreamReader xmlr = xmlif.createXMLStreamReader(filename, new FileInputStream(file));
```

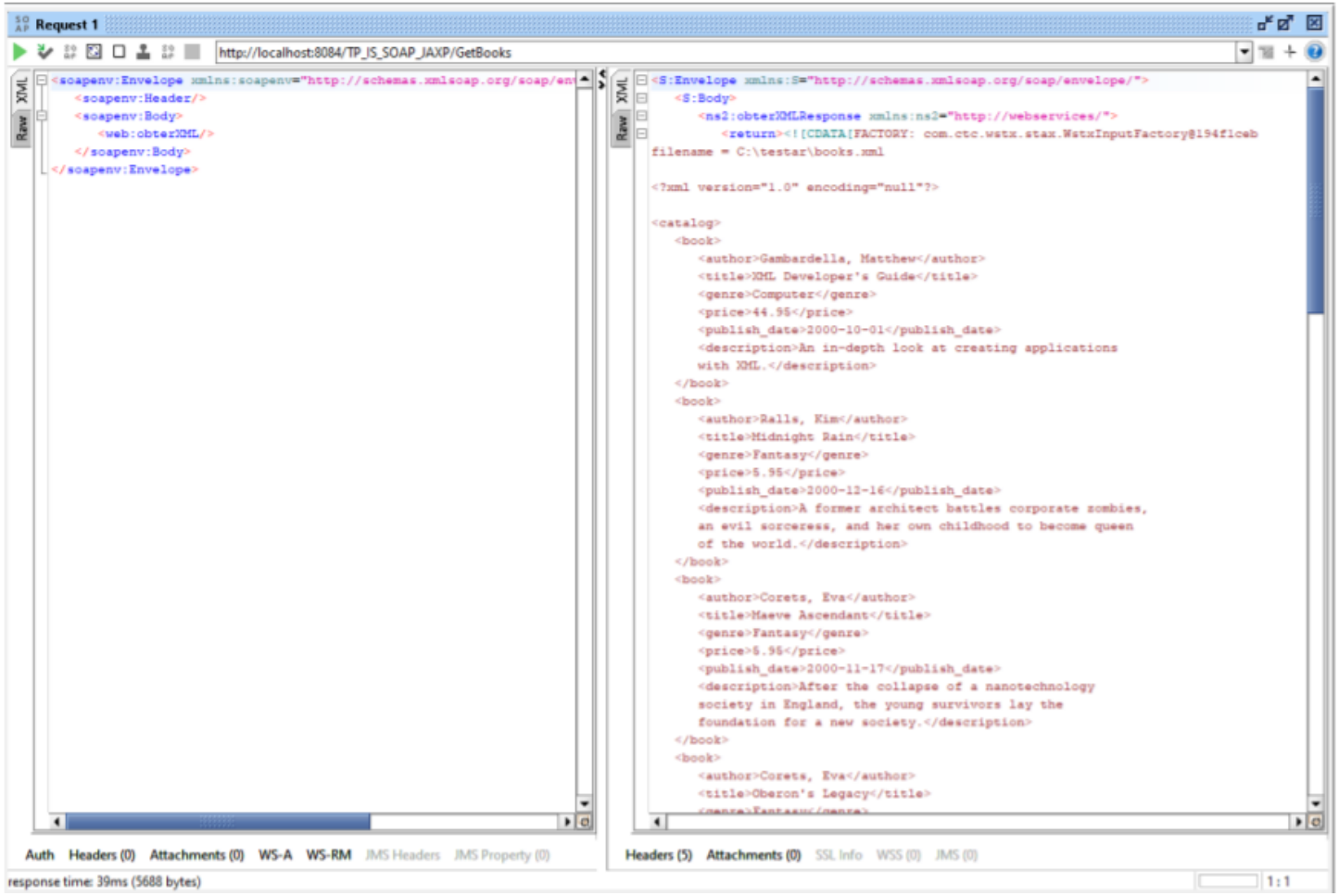
JAVA – SOAP - JAXP

Código do WS GetBooks.java

```java
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.WebParam;

/**
 *
 * @author Vasco Alves
 */
@WebService(serviceName = "GetBooks")
public class GetBooks {

    @WebMethod(operationName = "obterXML")
    public String obterXML() {
        try {
            return CursorParse.obterDados("C:\\testar\\books.xml");
        } catch (Exception ex) {
            return "Ocorreu um erro...";
        }
    }
}
```

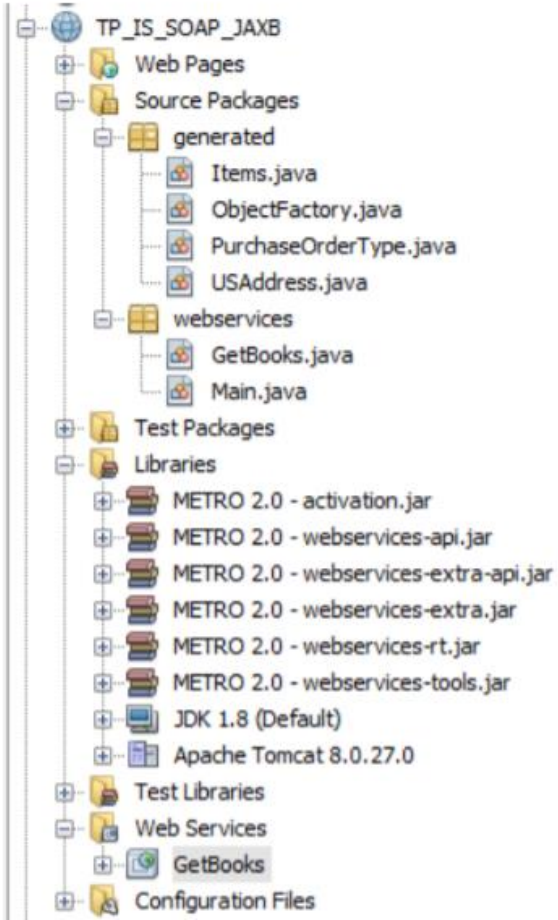
JAVA – SOAP - JAXP

Teste SoapUI



JAVA – SOAP - JAXB

Estrutura do Projeto



Main.java – ficheiro que executa o parser JAXB no ficheiro XML

GetBooks.java – classe do web service que invoca o Main. Nesta classe é identificado o ficheiro XML a usar

JAVA – SOAP - JAXB

Parte do Código de Main.java

```
public static String obterDados(String filename) {
    File file = null;
    try {
        if(filename == null)
            filename = "C:\\testar\\po.xml";
        file = new File(filename);
        JAXBContext jc = JAXBContext.newInstance( "generated" );

        Unmarshaller u = jc.createUnmarshaller();
        JAXBElement<?> poElement = (JAXBElement<?>)u.unmarshal( new FileInputStream( file ));
        PurchaseOrderType po = (PurchaseOrderType)poElement.getValue();

        dados += "Ship the following items to: \n";

        USAddress address = po.getShipTo();
        displayAddress( address );

        Items items = po.getItems();
        displayItems(items);

        return dados;
    } catch( JAXBException je ) {
        return "Ocorreu um erro...";
    } catch( IOException ioe ) {
        return "Ocorreu um erro...";
    }
}
```

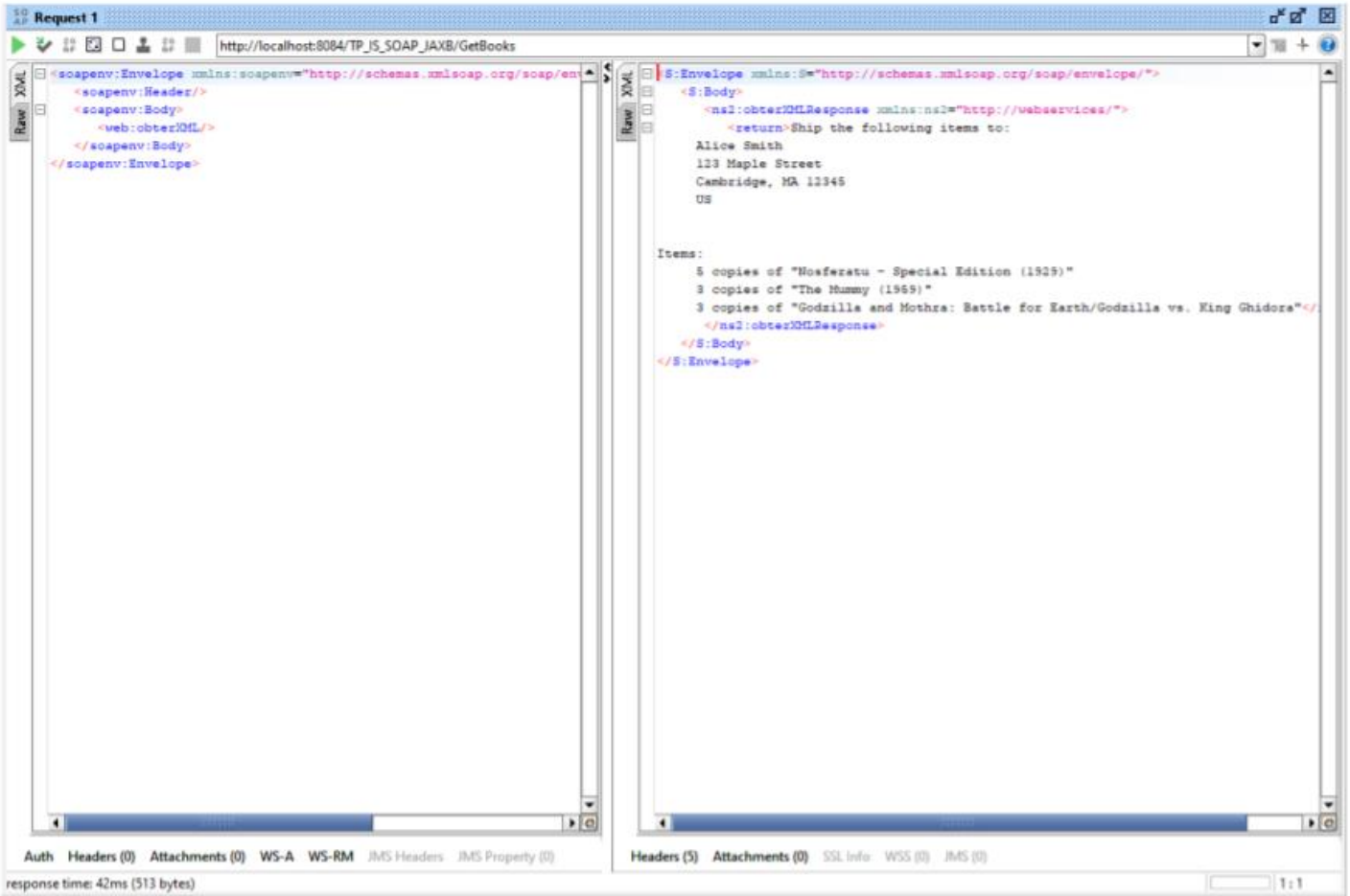
JAVA – SOAP - JAXB

Código do WS GetBooks.java

```
package webservices;

import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.WebParam;

/**
 *
 * @author Vasco Alves
 */
@WebService(serviceName = "GetBooks")
public class GetBooks {

    @WebMethod(operationName = "obterXML")
    public String obterXML() {
        return Main.obterDados("C:\\testar\\po.xml");
    }
}
```

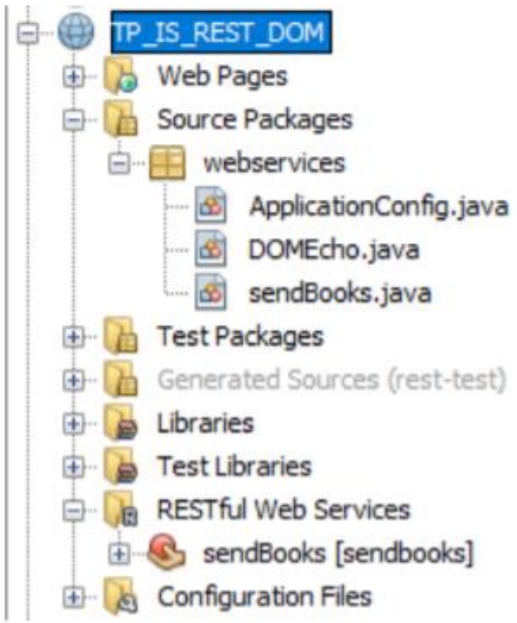
JAVA – SOAP - JAXB

Teste SoapUI



JAVA – REST - DOM

Estrutura do Projeto Criado



DOMEcho.java – ficheiro que trata do parsing DOM

GetBooks.java – classe do web service que invoca o DOMEcho

JAVA – REST - DOM

Parte do Código DOMEcho.java

```
public class DOMEcho {
    static final String outputEncoding = "UTF-8";
    private PrintWriter out;
    private int indent = 0;
    private final String basicIndent = "  ";
    static final String JAXP_SCHEMA_LANGUAGE = "http://java.sun.com/xml/jaxp/properties/schemaLanguage";
    static final String W3C_XML_SCHEMA = "http://www.w3.org/2001/XMLSchema";
    static final String JAXP_SCHEMA_SOURCE = "http://java.sun.com/xml/jaxp/properties/schemaSource";

    static String a = new String();

    DOMEcho(PrintWriter out) {
        this.out = out;
    }

    private void printlnCommon(Node n) {
        a += " nodeName=\"" + n.getNodeName() + "\"";
        String val = n.getNamespaceURI();
        if (val != null) {
            a += " uri=\"" + val + "\"";
        }
        val = n.getPrefix();
        if (val != null) {
            a += " pre=\"" + val + "\"";
        }
        val = n.getLocalName();
        if (val != null) {
            a += " local=\"" + val + "\"";
        }
        val = n.getNodeValue();
        if (val != null) {
            a += " nodeValue=";
            if (val.trim().equals("")) {
                // Whitespace
                a += "[WS]";
            } else {
                a += "\"" + n.getNodeValue() + "\"";
            }
        }
        a += "\n";
    }

    private void outputIndentation() {
        for (int i = 0; i < indent; i++) {
            out.print(basicIndent);
```

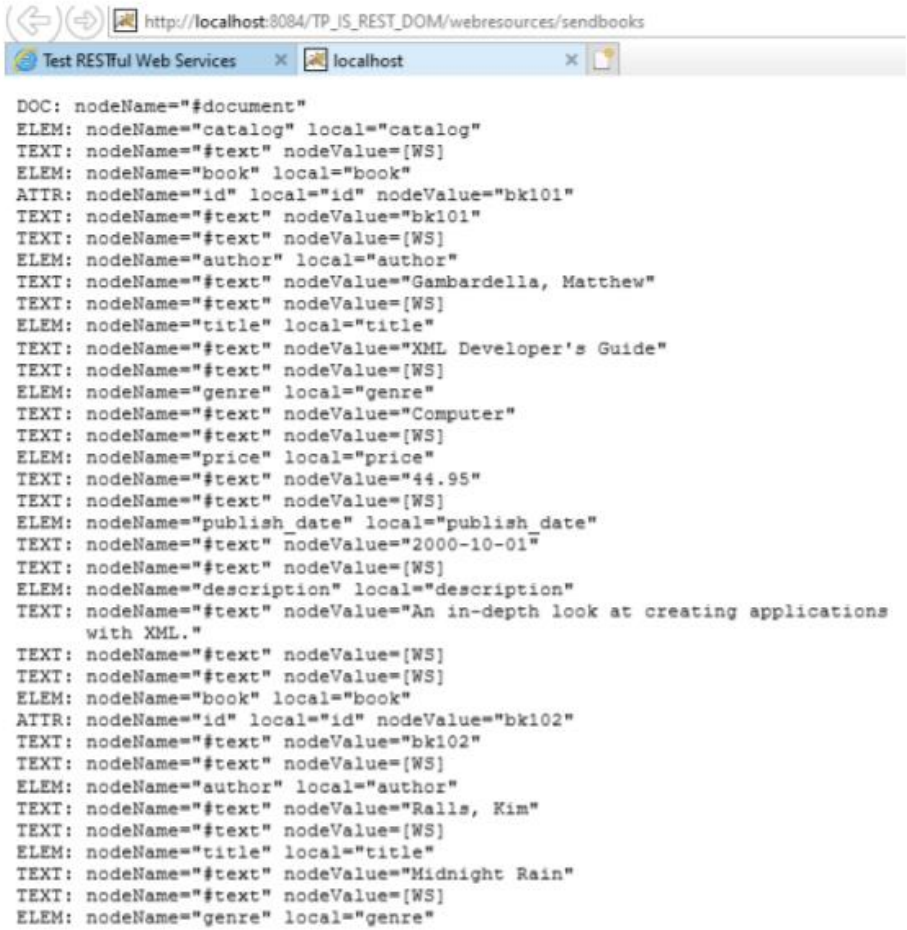
JAVA – REST - DOM

Código do WS sendBooks.java

```java
package webservices;

import javax.ws.rs.core.Context;
import javax.ws.rs.core.UriInfo;
import javax.ws.rs.Produces;
import javax.ws.rs.Consumes;
import javax.ws.rs.GET;
import javax.ws.rs.Path;
import javax.ws.rs.PUT;
import javax.ws.rs.core.MediaType;

@Path("sendbooks")
public class sendBooks {

    @Context
    private UriInfo context;

    public sendBooks() {}

    @GET
    @Produces(MediaType.APPLICATION_XML)
    public String getXml() {
        try {
            return DOMEcho.obterStringXML("C:\\testar\\books.xml");
        } catch (Exception ex) {
            return "Ocorreu um erro! -> " + ex.getMessage();
        }
    }
}
```

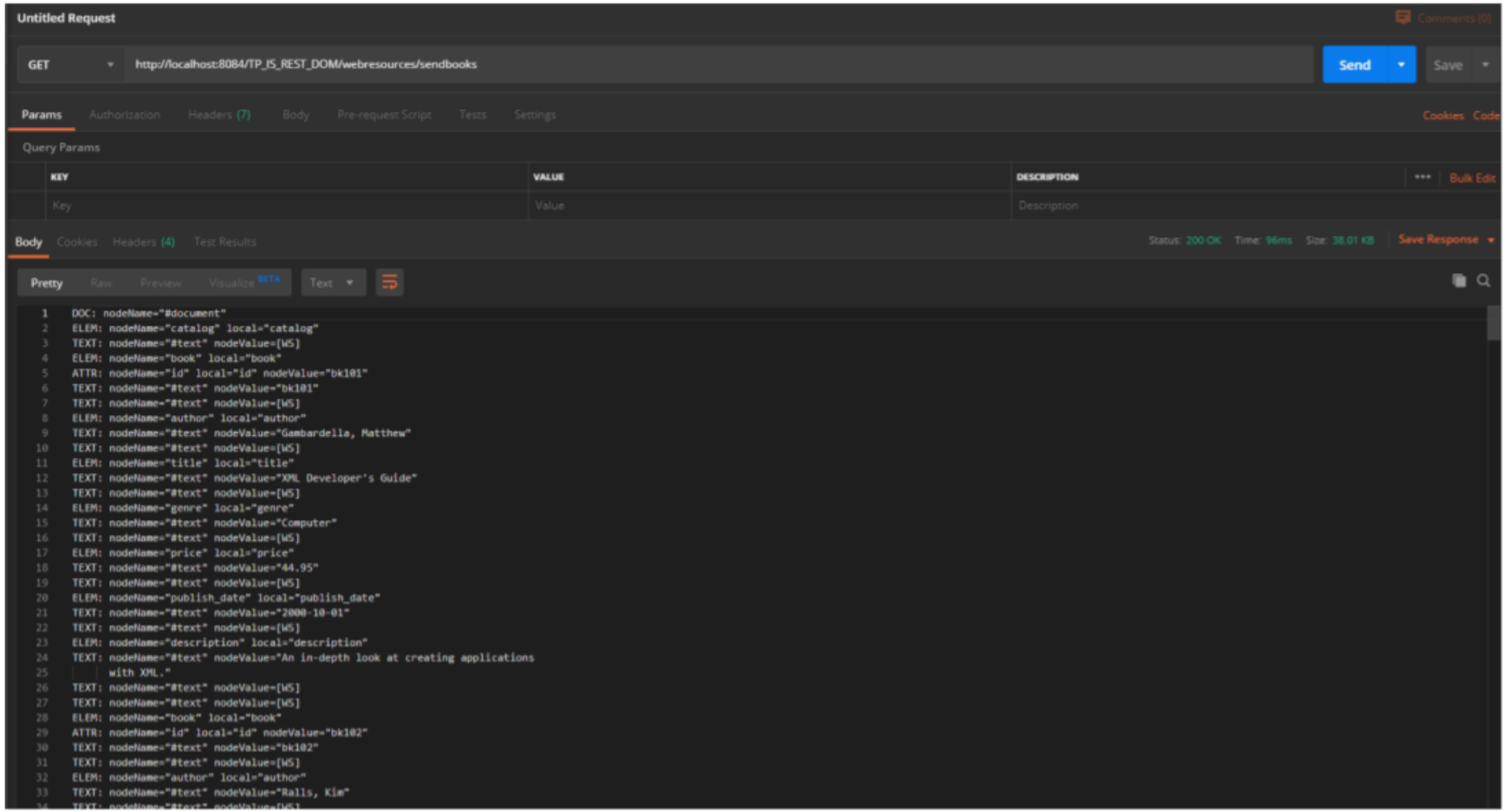
JAVA – REST - DOM

Teste Browser



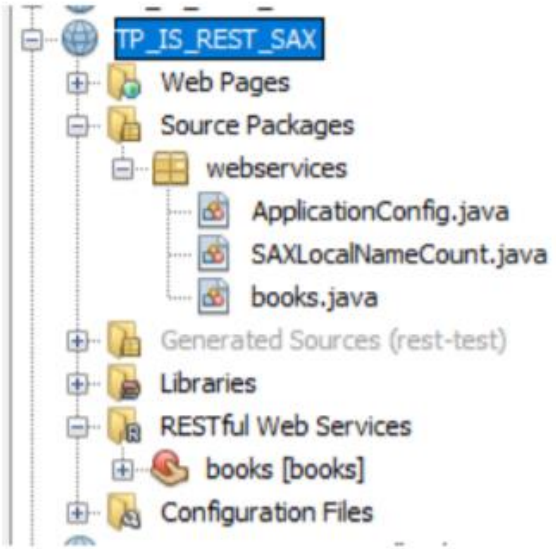
JAVA – REST - DOM

Teste POSTMAN



JAVA – REST - SAX

Estrutura do Projeto



SAXLocalNameCount.java – ficheiro que conta o número de ocorrências dos nós XML usando o SAX

GetBooks.java – classe do web service que invoca o SAXLocalNameCount. Nesta classe é identificado o ficheiro XML a usar

JAVA – REST - SAX

Parte do Código de SAXLocalNameCount.java

```
public static String obterDados(String filename) throws Exception {
    boolean dtdValidate = false;
    boolean xsdValidate = false;
    String schemaSource = null;

    if (filename == null) {
        filename = "C:\\testar\\books.xml";
    }

    SAXParserFactory spf = SAXParserFactory.newInstance();
    spf.setNamespaceAware(true);
    spf.setValidating(dtdValidate || xsdValidate);
    SAXParser saxParser = spf.newSAXParser();
    if (schemaSource != null) {
        saxParser.setProperty(JAXP_SCHEMA_SOURCE, new File(schemaSource));
    }

    XMLReader xmlReader = saxParser.getXMLReader();
    xmlReader.setContentHandler(new SAXLocalNameCount());
    xmlReader.setErrorHandler(new MyErrorHandler(System.err));
    xmlReader.parse(convertToFileURL(filename));

    return teste;
}
```

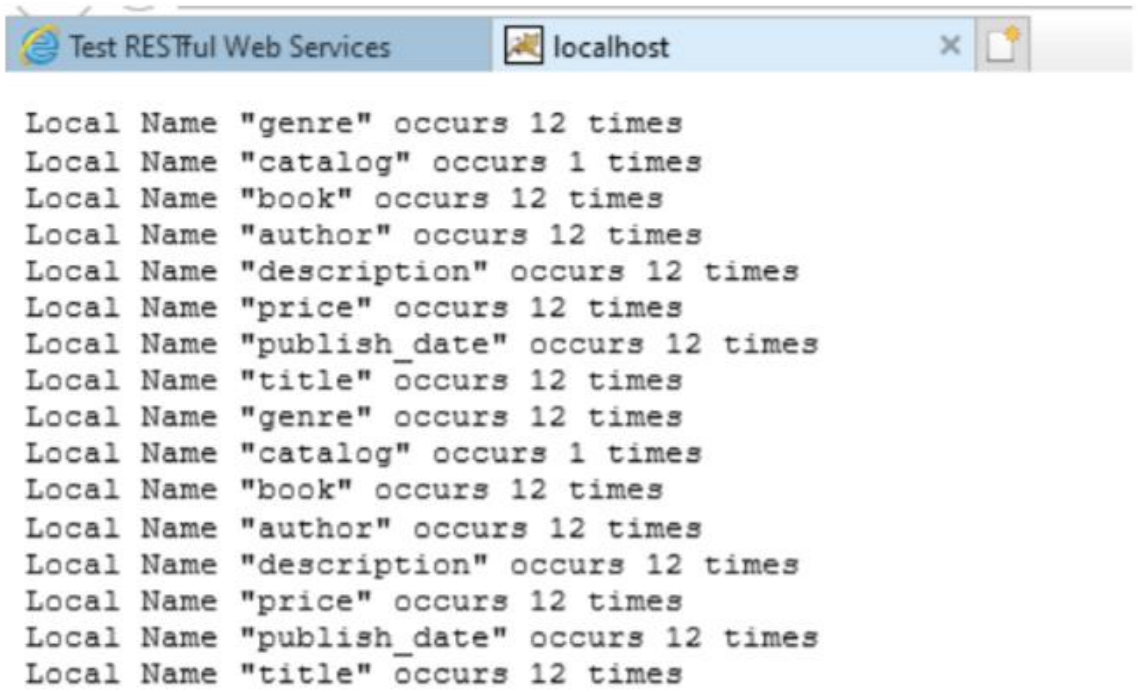
JAVA – REST - SAX

Código do WS books.java

```
1   package webservices;
2
3   import javax.ws.rs.core.Context;
4   import javax.ws.rs.core.UriInfo;
5   import javax.ws.rs.Produces;
6   import javax.ws.rs.Consumes;
7   import javax.ws.rs.GET;
8   import javax.ws.rs.Path;
9   import javax.ws.rs.PUT;
10  import javax.ws.rs.core.MediaType;
11
12  @Path("books")
13  public class books {
14
15      @Context
16      private UriInfo context;
17
18      public books() {}
19
20      @GET
21      @Produces(MediaType.TEXT_PLAIN)
22      public String getXml() {
23          try{
24              return SAXLocalNameCount.obterDados("C:\\testar\\books.xml");
25          }catch(Exception e){
26              return "ocorreu um erro...";
27          }
28      }
29  }
```

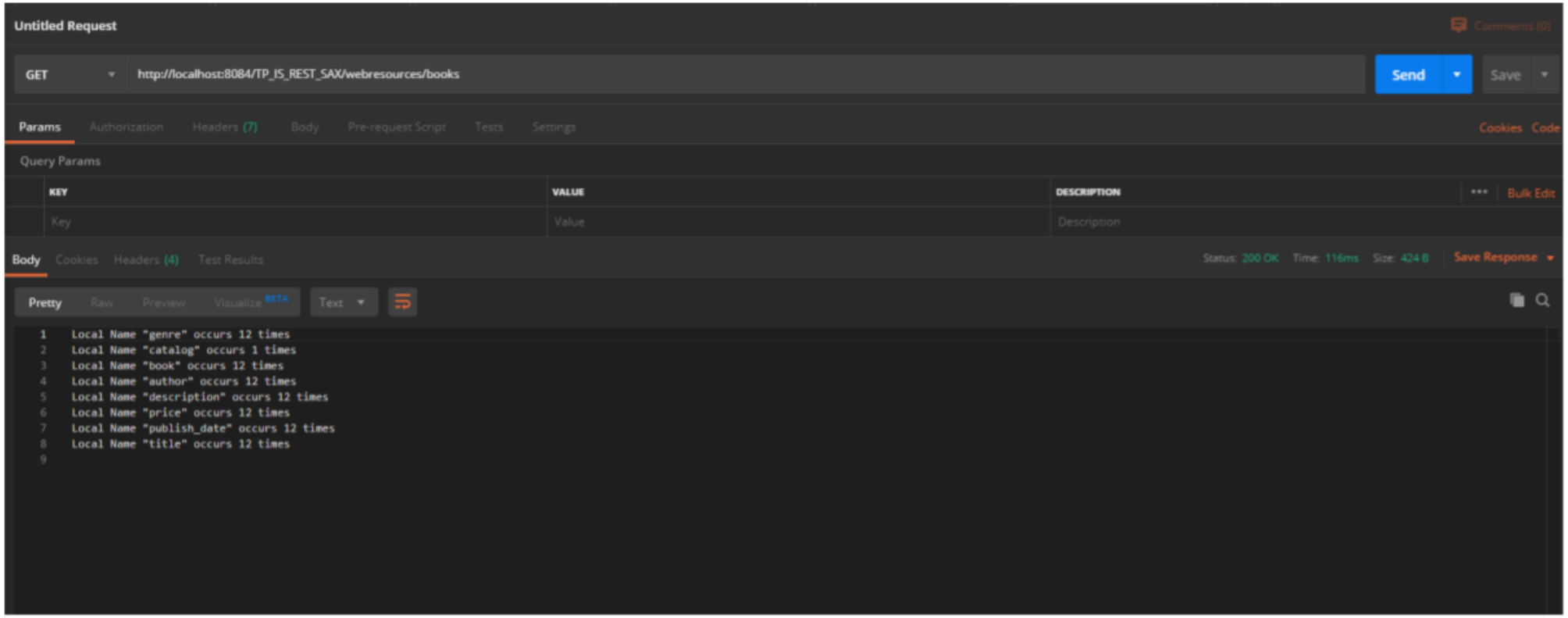
JAVA – REST - SAX

Teste Browser



JAVA – REST - SAX

Teste POSTMAN



JAVA – REST - JDOM

Estrutura do Projeto

PlayWithJDom.java – ficheiro que chama o JDOM

GetBooks.java – classe do web service que invoca o PlayWithJDOM. Nesta classe é identificado o ficheiro XML a usar

Necessário adicionar jdom-2.0.5.jar às librarias do projeto

JAVA – REST - JDOM

Parte do Código de PlayWithJDOM.java

```
import java.io.File;
import java.io.IOException;
import java.io.*;
import org.jdom2.Document;
import org.jdom2.Element;
import org.jdom2.JDOMException;
import org.jdom2.input.SAXBuilder;
import org.jdom2.output.DOMOutputter;
import org.jdom2.output.XMLOutputter;

public class PlayWithJDom  {
    XMLOutputter printer = null;

    public PlayWithJDom () {
        printer = new XMLOutputter();
    }

    public String parseAndPlay(File anXmlDocFile){
        SAXBuilder parser = new SAXBuilder();
        try {
            Document doc = parser.build(anXmlDocFile);
            return this.printJDom(doc);
        } catch (JDOMException ex) {
            ex.printStackTrace();
        } catch (IOException ex) {
            ex.printStackTrace();
        }
        return "Ocorreu um erro";
    }

    public String printJDom(Document aJDOMdoc) throws IOException{
        return printer.outputString(aJDOMdoc);
    }
```

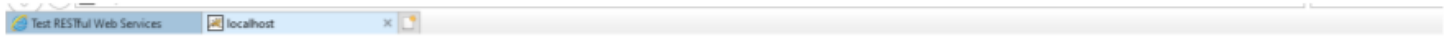
JAVA – REST - JDOM

Código do WS books.java

```
1   package webservices;
2
3   import java.io.File;
4   import javax.ws.rs.core.Context;
5   import javax.ws.rs.core.UriInfo;
6   import javax.ws.rs.Produces;
7   import javax.ws.rs.Consumes;
8   import javax.ws.rs.GET;
9   import javax.ws.rs.Path;
10  import javax.ws.rs.PUT;
11  import javax.ws.rs.core.MediaType;
12
13  @Path("books")
14  public class books {
15
16      @Context
17      private UriInfo context;
18
19      public books() {}
20
21      @GET
22      @Produces(MediaType.APPLICATION_XML)
23      public String getXml() {
24          PlayWithJDom play = new PlayWithJDom();
25          String XML = play.parseAndPlay(new File("C:\\testar\\books.xml"));
26          return XML;
27      }
28  }
29
```

JAVA – REST - JDOM

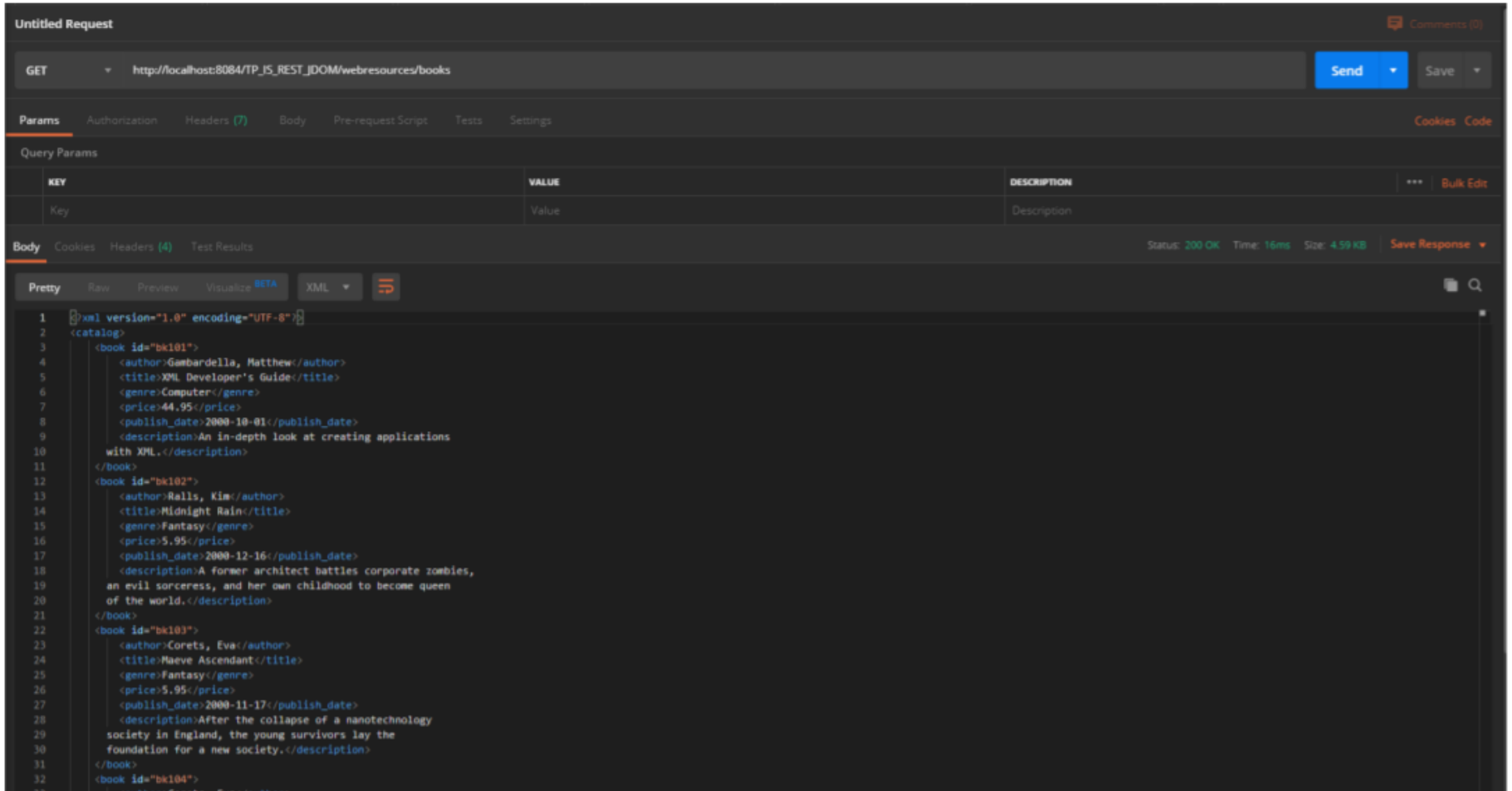
Teste Browser




```
<?xml version="1.0" encoding="UTF-8"?>
<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, battle one another for control of England. Sequel to Oberon's Legacy.</description>
  </book>
  <book id="bk106">
    <author>Randall, Cynthia</author>
    <title>Lover Birds</title>
    <genre>Romance</genre>
    <price>4.95</price>
    <publish_date>2000-09-02</publish_date>
    <description>When Carla meets Paul at an ornithology conference, tempers fly as feathers get ruffled.</description>
  </book>
  <book id="bk107">
    <author>Thurman, Paula</author>
    <title>Splish Splash</title>
    <genre>Romance</genre>
    <price>4.95</price>
    <publish_date>2000-11-02</publish_date>
    <description>A deep sea diver finds true love twenty thousand leagues beneath the sea.</description>
  </book>
```

JAVA – REST - JDOM

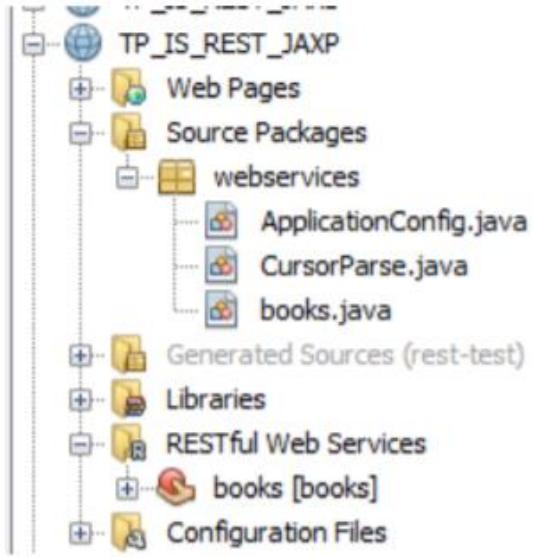
Teste POSTMAN



Versão 1 – Documento em Atualização

JAVA – REST - JAXP

Estrutura do Projeto



CursorParse.java – ficheiro que executa o parser JAXP no ficheiro XML

GetBooks.java – classe do web service que invoca o CursorParse. Nesta classe é identificado o ficheiro XML a usar

JAVA – REST - JAXP

Parte do Código de CursorParse.java

```
public static String obterDados(String filename) throws Exception {
    File file = null;
    try {
        if(filename == null)
            filename = "C:\\testar\\books.xml";
        file = new File(filename);
        if (!file.exists()) {
            System.out.println("File: " + filename + " does not exist.");
            System.exit(0);
        }
    } catch (ArrayIndexOutOfBoundsException aioobe){
        printUsage();
        System.exit(0);
    } catch (Exception ex){
        printUsage();
        ex.printStackTrace() ;
    }

    XMLInputFactory xmlif = null ;
    try{
        xmlif = XMLInputFactory.newInstance();
        xmlif.setProperty(XMLInputFactory.IS_REPLACING_ENTITY_REFERENCES,Boolean.TRUE);
        xmlif.setProperty(XMLInputFactory.IS_SUPPORTING_EXTERNAL_ENTITIES,Boolean.FALSE);
        xmlif.setProperty(XMLInputFactory.IS_COALESCING , Boolean.FALSE);
    }catch(Exception ex){
        ex.printStackTrace();
    }
    dados += "FACTORY: " + xmlif + "\n";
    dados += "filename = "+ filename + "\n\n";
    long starttime = System.currentTimeMillis() ;

    try{
        XMLStreamReader xmlr = xmlif.createXMLStreamReader(filename, new FileInputStream(file));
```

JAVA – REST - JAXP

Código do WS books.java

```
1   package webservices;
2
3   import javax.ws.rs.core.Context;
4   import javax.ws.rs.core.UriInfo;
5   import javax.ws.rs.Produces;
6   import javax.ws.rs.Consumes;
7   import javax.ws.rs.GET;
8   import javax.ws.rs.Path;
9   import javax.ws.rs.PUT;
10  import javax.ws.rs.core.MediaType;
11
12  /**
13   * REST Web Service
14   *
15   * @author Vasco Alves
16   */
17  @Path("books")
18  public class books {
19
20      @Context
21      private UriInfo context;
22
23      public books() {}
24
25      @GET
26      @Produces(MediaType.TEXT_PLAIN)
27      public String getText() {
28          try {
29              return CursorParse.obterDados("C:\\testar\\books.xml");
30          } catch (Exception ex) {
31              return "Ocorreu um erro...";
32          }
33      }
34  }
```

JAVA – REST - JAXP

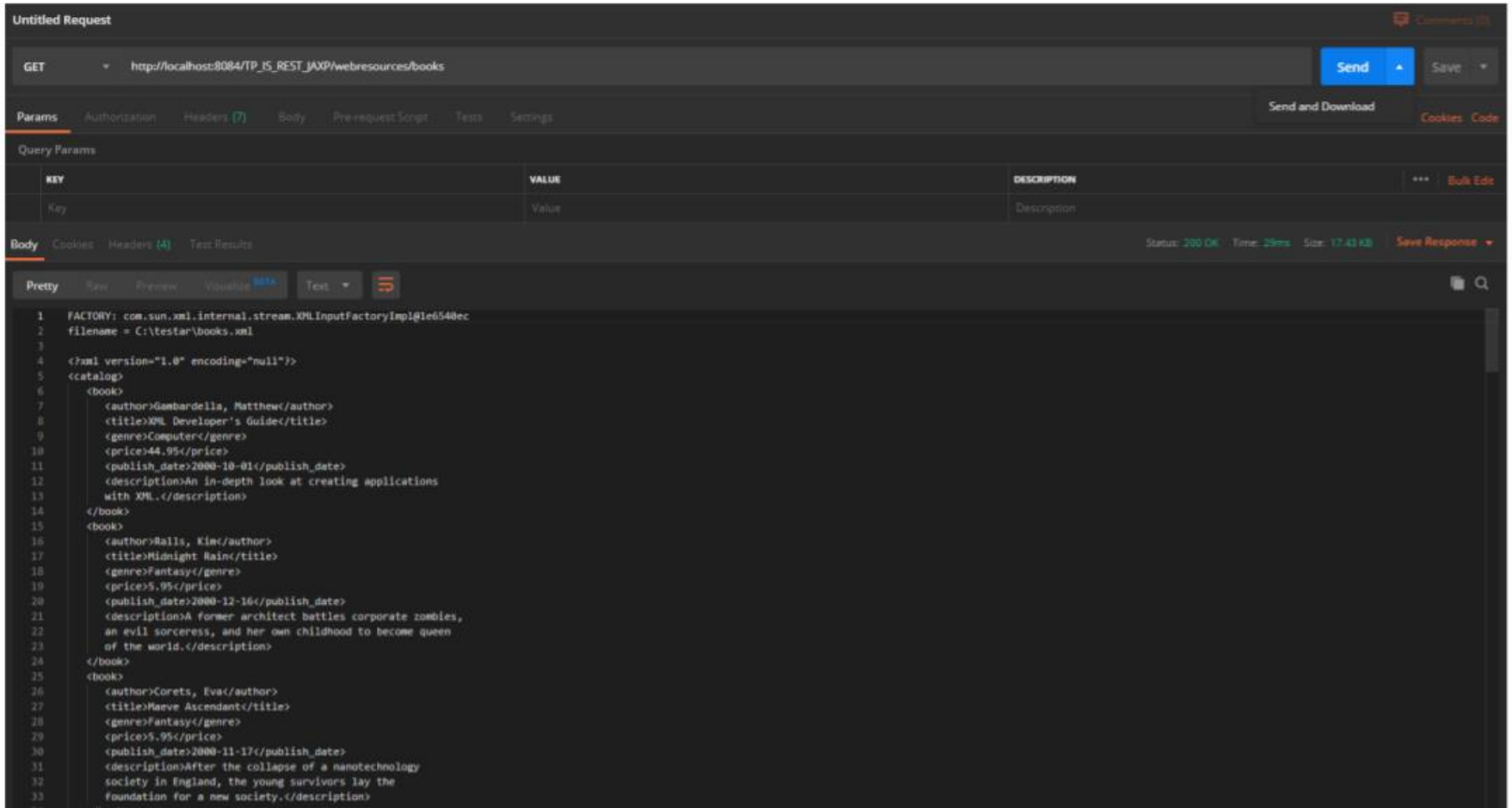
Teste Browser

localhost

```
FACTORY: com.sun.xml.internal.stream.XMLInputFactoryImpl@1e6540ec
filename = C:\testar\books.xml

<?xml version="1.0" encoding="null"?>
<catalog>
   <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>
```

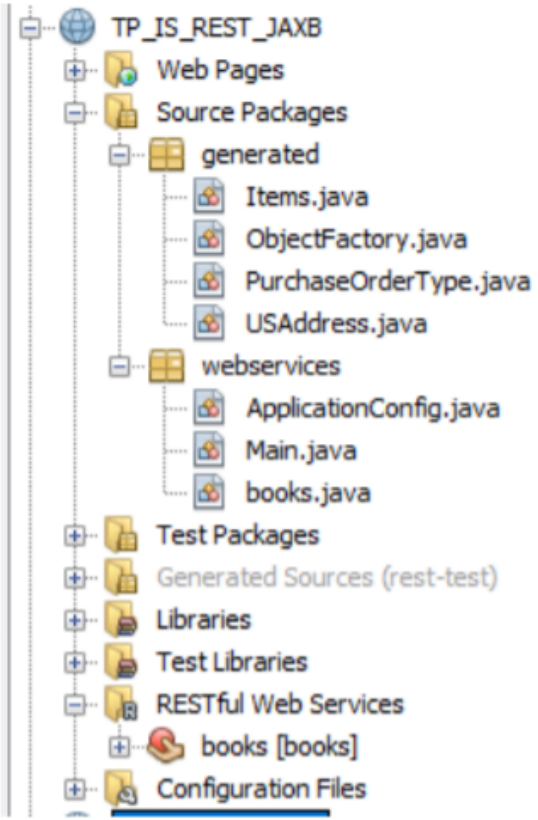
JAVA – REST - JAXP

Teste POSTMAN



JAVA – REST - JAXB

Estrutura do Projeto



Main.java – ficheiro que executa o parser JAXB no ficheiro XML

GetBooks.java – classe do web service que invoca o Main. Nesta classe é identificado o ficheiro XML a usar

JAVA – REST - JAXB

Parte do Código de Main.java

```
public static String obterDados(String filename) {
    File file = null;
    try {
        if(filename == null)
            filename = "C:\\testar\\po.xml";
        file = new File(filename);
        JAXBContext jc = JAXBContext.newInstance( "generated" );

        Unmarshaller u = jc.createUnmarshaller();
        JAXBElement<?> poElement = (JAXBElement<?>)u.unmarshal( new FileInputStream( file ));
        PurchaseOrderType po = (PurchaseOrderType)poElement.getValue();

        dados += "Ship the following items to: \n";

        USAddress address = po.getShipTo();
        displayAddress( address );

        Items items = po.getItems();
        displayItems(items);

        return dados;
    } catch( JAXBException je ) {
        return "Ocorreu um erro...";
    } catch( IOException ioe ) {
        return "Ocorreu um erro...";
    }
}
```

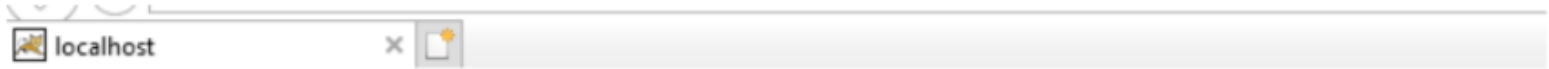
JAVA – REST - JAXB

Código do WS books.java

```java
package webservices;

import javax.ws.rs.core.Context;
import javax.ws.rs.core.UriInfo;
import javax.ws.rs.Produces;
import javax.ws.rs.Consumes;
import javax.ws.rs.GET;
import javax.ws.rs.Path;
import javax.ws.rs.PUT;
import javax.ws.rs.core.MediaType;

/**
 * REST Web Service
 *
 * @author Vasco Alves
 */
@Path("books")
public class books {

    @Context
    private UriInfo context;

    public books() {}

    @GET
    @Produces(MediaType.TEXT_PLAIN)
    public String getText() {
        try {
            return CursorParse.obterDados("C:\\testar\\books.xml");
        } catch (Exception ex) {
            return "Ocorreu um erro...";
        }
    }
}
```

JAVA – REST - JAXB

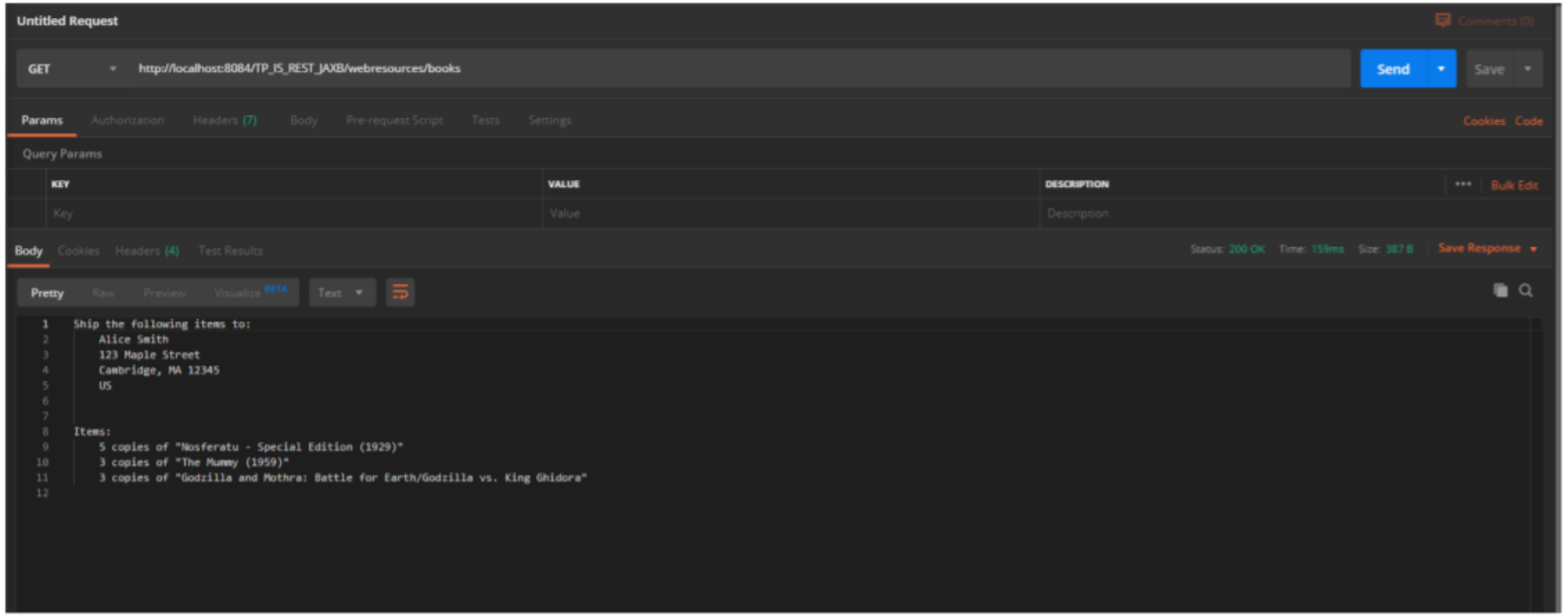
Teste Browser




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


Items:
        5 copies of "Nosferatu - Special Edition (1929)"
        3 copies of "The Mummy (1959)"
        3 copies of "Godzilla and Mothra: Battle for Earth/Godzilla vs. King Ghidora"
Ship the following items to:
        Alice Smith
        123 Maple Street
        Cambridge, MA 12345
        US


Items:
        5 copies of "Nosferatu - Special Edition (1929)"
        3 copies of "The Mummy (1959)"
        3 copies of "Godzilla and Mothra: Battle for Earth/Godzilla vs. King Ghidora"
```

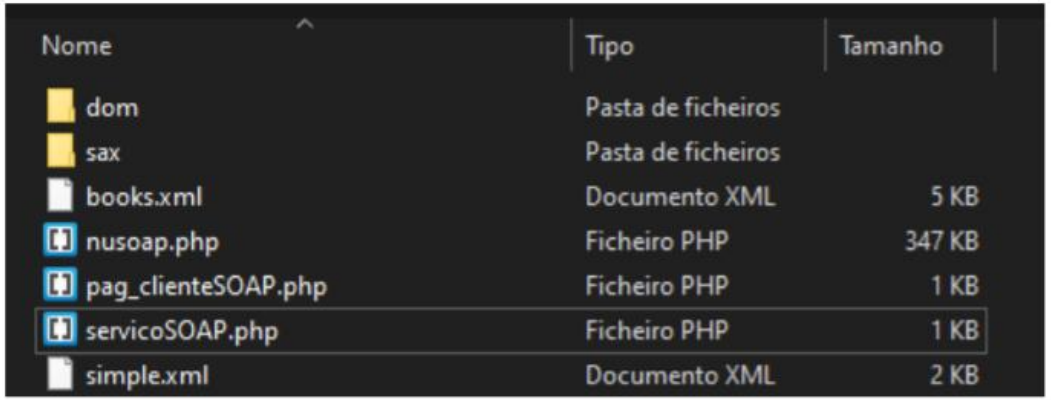
JAVA – REST - JAXB

Teste POSTMAN



PHP – SOAP - DOM

Estrutura do Projeto Criado



servicoSOAP.php – ficheiro que trata de ler o ficheiro XML com o DOM e de o enviar ao cliente num envelope SOAP

pag_ClienteSOAP.php – cliente SOAP em php que chama o servicoSOAP.php

PHP – SOAP - DOM

Parte do Código servicoSOAP.php

```php
<?php
    require_once "nusoap.php";
    function obterDados($nomeFicheiroACorrer) {
        // -- verifica se shell_exec está ativo
        if(!function_exists('shell_exec')) {
            die("A função shell_exec deve estar ativa!");
        }

        $output = shell_exec('java dom/DOMEcho ' . $nomeFicheiroACorrer);
        $XML = "";

        $XML = "DOC: nodeName=" . explode('DOC: nodeName=', $output)[1];
        $XML = explode(' Parsing Time = ', $XML)[0];

        return $XML;

    }

    // -- cria servidor SOAP e emvia
    $server = new soap_server();
    $server->configureWSDL('AddService', 'obtemXML');
    $server->wsdl->schemaTargetNamespace = 'obtemXML';
    $server->register("obterDados",
        array('tipo' => "xsd:string"),
        array('return' => 'xsd:string')
    );
    $server->service(file_get_contents('php://input'));
?>
```

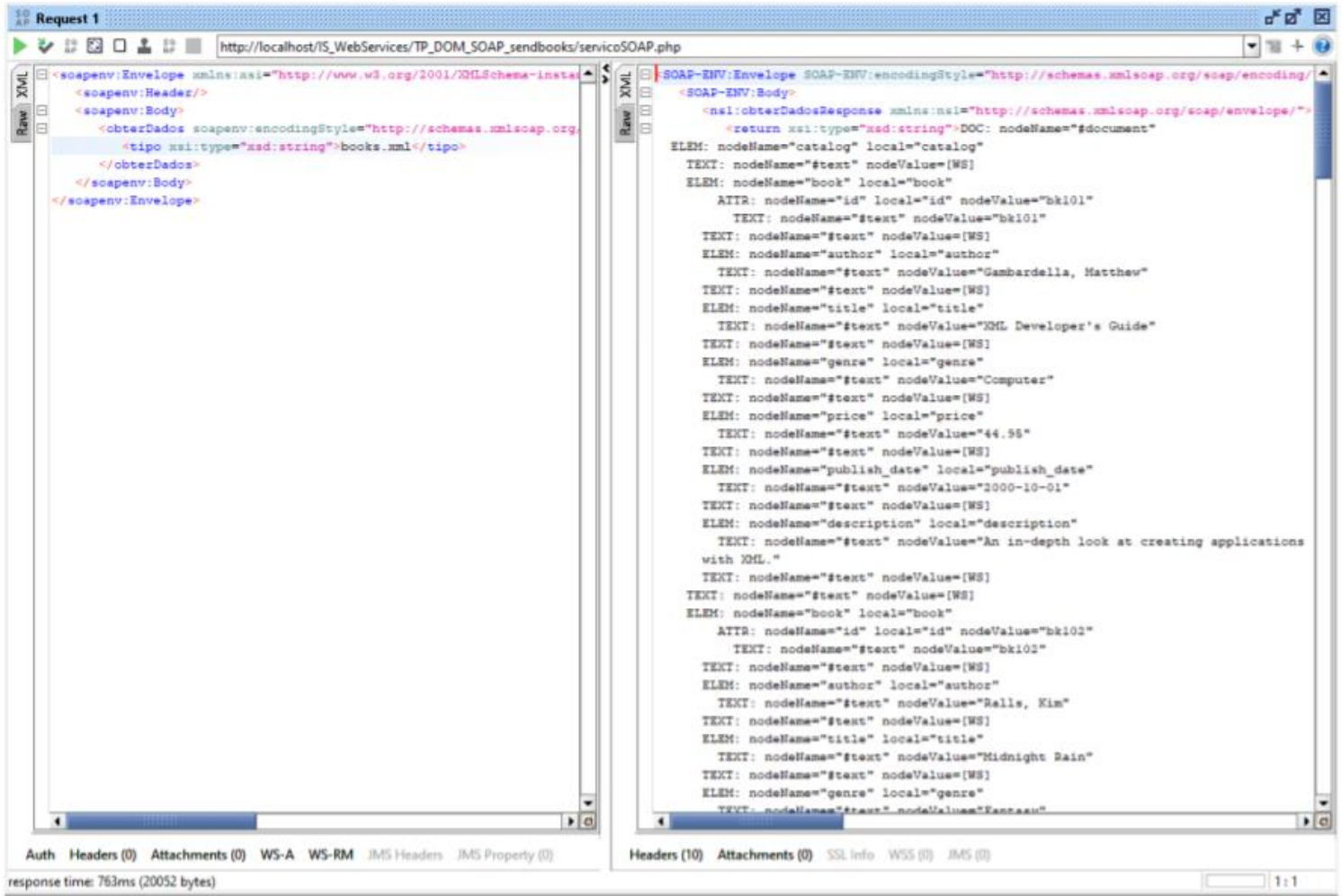
PHP – SOAP - DOM

Código do pag_clienteSOAP.php

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

$client = new nusoap_client("http://localhost/IS_WebServices/TP_DOM_SOAP_sendbooks/servicoSOAP.php");


$error = $client->getError();
if ($error) {
    echo "<h2>Constructor error</h2><pre>" . $error . "</pre>";
}

$result = $client->call("obterDados", array("nome" => "books.xml"));

if ($client->fault) {
    echo "<h2>Fault</h2><pre>" . $result . "</pre>";
} else {
    $error = $client->getError();
    if ($error) {
        echo "<h2>Error</h2><pre>" . $error . "</pre>";
    }else {
        header("content-type: text/plain");
        echo $result;
    }
}
```

PHP – SOAP - DOM

Teste SoapUI



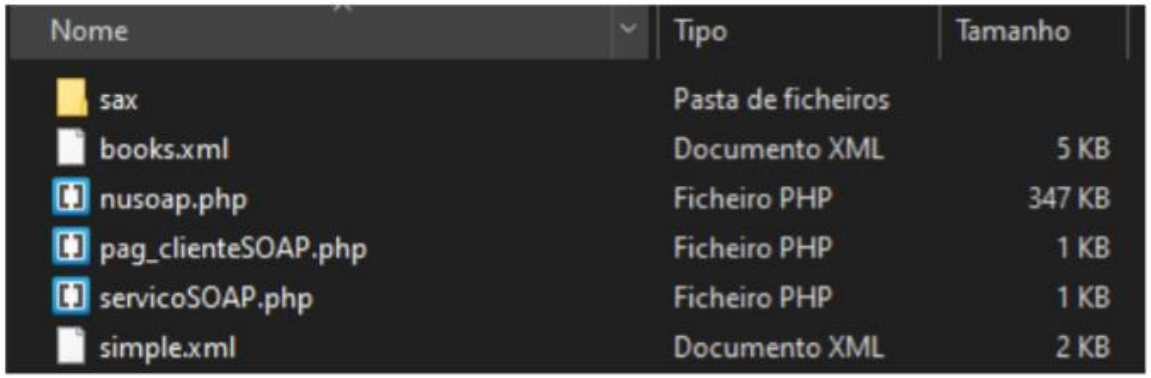
PHP – SOAP - DOM

Teste Browser (pag_clienteSOAP.php)

```
DOC: nodeName="#document"
  ELEM: nodeName="catalog" local="catalog"
    TEXT: nodeName="#text" nodeValue=[WS]
    ELEM: nodeName="book" local="book"
        ATTR: nodeName="id" local="id" nodeValue="bk101"
          TEXT: nodeName="#text" nodeValue="bk101"
      TEXT: nodeName="#text" nodeValue=[WS]
      ELEM: nodeName="author" local="author"
        TEXT: nodeName="#text" nodeValue="Gambardella, Matthew"
      TEXT: nodeName="#text" nodeValue=[WS]
      ELEM: nodeName="title" local="title"
        TEXT: nodeName="#text" nodeValue="XML Developer's Guide"
      TEXT: nodeName="#text" nodeValue=[WS]
      ELEM: nodeName="genre" local="genre"
        TEXT: nodeName="#text" nodeValue="Computer"
      TEXT: nodeName="#text" nodeValue=[WS]
      ELEM: nodeName="price" local="price"
        TEXT: nodeName="#text" nodeValue="44.95"
      TEXT: nodeName="#text" nodeValue=[WS]
      ELEM: nodeName="publish_date" local="publish_date"
        TEXT: nodeName="#text" nodeValue="2000-10-01"
      TEXT: nodeName="#text" nodeValue=[WS]
      ELEM: nodeName="description" local="description"
        TEXT: nodeName="#text" nodeValue="An in-depth look at creating applications
      with XML."
        TEXT: nodeName="#text" nodeValue=[WS]
    TEXT: nodeName="#text" nodeValue=[WS]
    ELEM: nodeName="book" local="book"
        ATTR: nodeName="id" local="id" nodeValue="bk102"
          TEXT: nodeName="#text" nodeValue="bk102"
      TEXT: nodeName="#text" nodeValue=[WS]
      ELEM: nodeName="author" local="author"
        TEXT: nodeName="#text" nodeValue="Ralls, Kim"
      TEXT: nodeName="#text" nodeValue=[WS]
      ELEM: nodeName="title" local="title"
        TEXT: nodeName="#text" nodeValue="Midnight Rain"
      TEXT: nodeName="#text" nodeValue=[WS]
      ELEM: nodeName="genre" local="genre"
        TEXT: nodeName="#text" nodeValue="Fantasy"
      TEXT: nodeName="#text" nodeValue=[WS]
      ELEM: nodeName="price" local="price"
        TEXT: nodeName="#text" nodeValue="5.95"
      TEXT: nodeName="#text" nodeValue=[WS]
      ELEM: nodeName="publish_date" local="publish_date"
```

PHP – SOAP - SAX

Estrutura do Projeto Criado



servicoSOAP.php – ficheiro que trata de ler o ficheiro XML com o SAX e de o enviar ao cliente num envelope SOAP

pag_ClienteSOAP.php – cliente SOAP em php que chama o servicoSOAP.php

PHP – SOAP - SAX

Parte do Código servicoSOAP.php

```php
<?php
require_once "nusoap.php";
function obterDados($nomeFicheiroACorrer) {
    // -- verifica se shell_exec está ativo
    if(!function_exists('shell_exec')) {
        die("A função shell_exec deve estar ativa!");
    }

    $output = shell_exec('java sax/SAXLocalNameCount ' . $nomeFicheiroACorrer);
    $XML = "";

    $XML = explode('Local Name', $output, 1);
    $XML = substr($output, strpos($output, "Local Name"));

    return $XML;

}

@var soap_server $server
$server = new soap_server();
$server->configureWSDL('AddService', 'obtemXML');
$server->wsdl->schemaTargetNamespace = 'obtemXML';
$server->register("obterDados",
    array('tipo' => "xsd:string"),
    array('return' => 'xsd:string')
);
$server->service(file_get_contents('php://input'));
?>
```

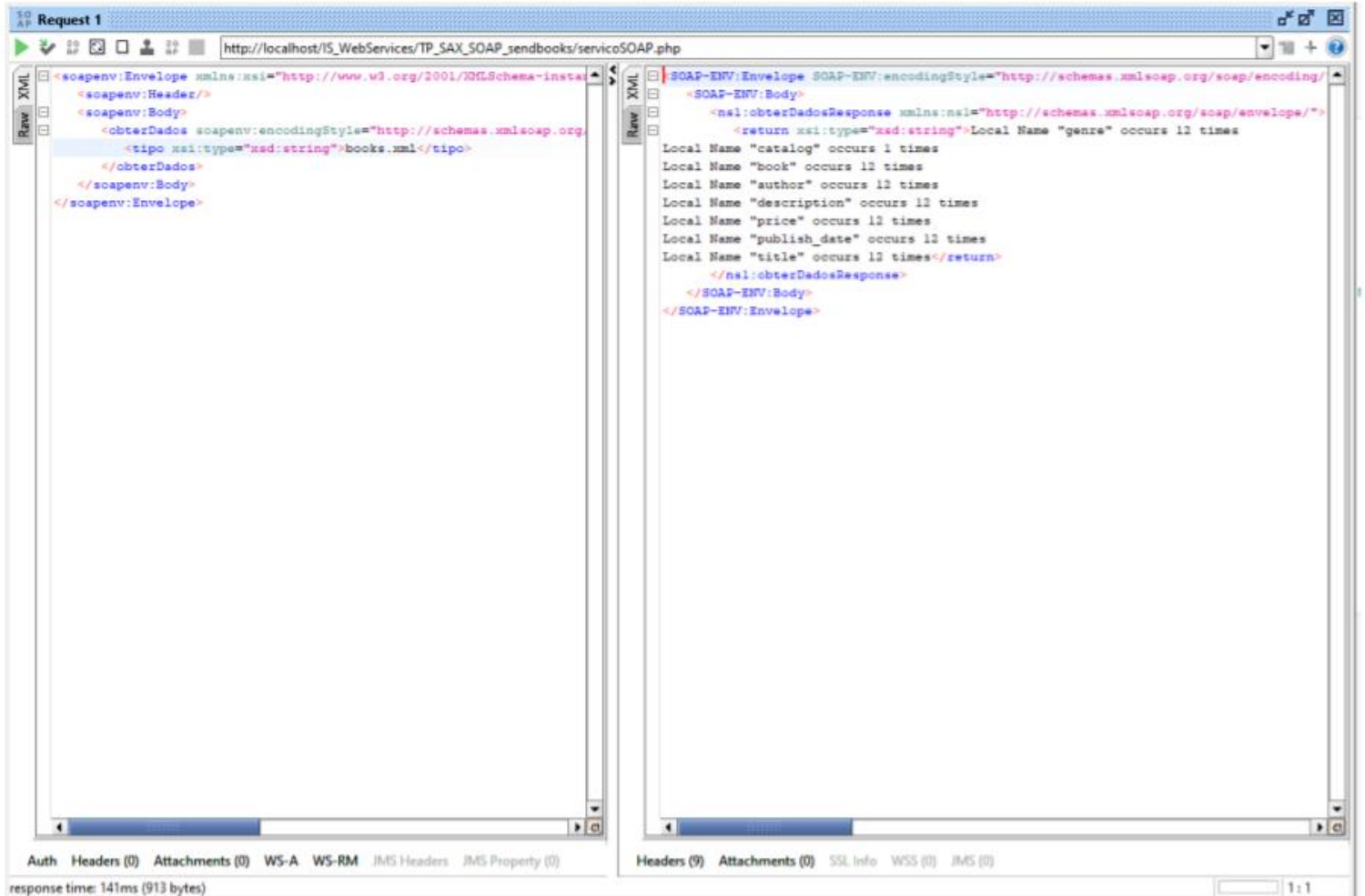
PHP – SOAP - SAX

Código do pag_clienteSOAP.php

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

$client = new nusoap_client("http://localhost/IS_WebServices/TP_SAX_SOAP_sendbooks/servicoSOAP.php");


$error = $client->getError();
if ($error) {
    echo "<h2>Constructor error</h2><pre>" . $error . "</pre>";
}

$result = $client->call("obterDados", array("nome" => "books.xml"));

if ($client->fault) {
    echo "<h2>Fault</h2><pre>" . $result . "</pre>";
} else {
    $error = $client->getError();
    if ($error) {
        echo "<h2>Error</h2><pre>" . $error . "</pre>";
    }else {
        header("content-type: text/plain");
        echo $result;
    }
}
?>
```

PHP – SOAP - SAX

Teste SoapUI



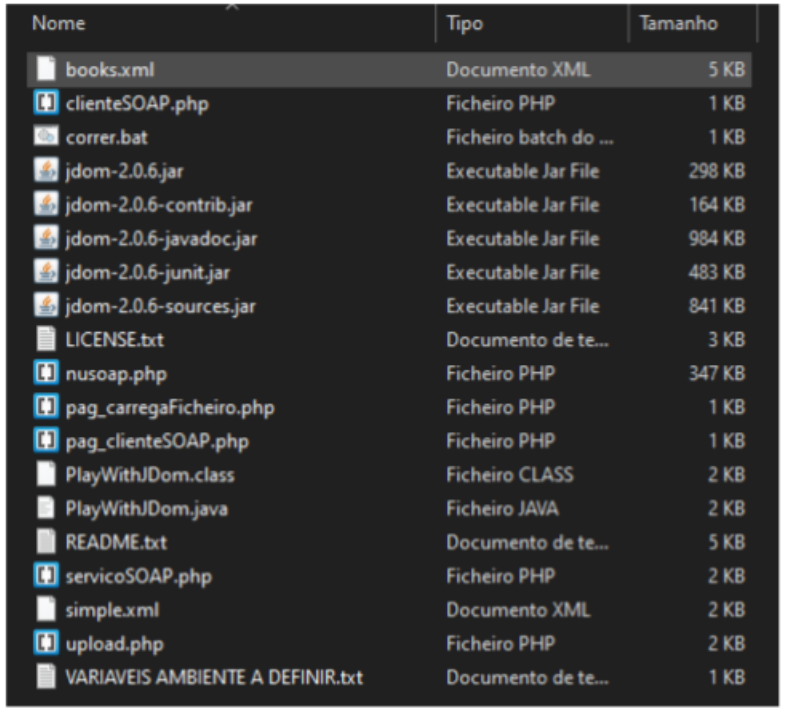
PHP – SOAP - SAX

Teste Browser (pag_clienteSOAP.php)

```
Local Name "genre" occurs 12 times
Local Name "catalog" occurs 1 times
Local Name "book" occurs 12 times
Local Name "author" occurs 12 times
Local Name "description" occurs 12 times
Local Name "price" occurs 12 times
Local Name "publish_date" occurs 12 times
Local Name "title" occurs 12 times
```

PHP – SOAP - JDOM

Estrutura do Projeto Criado



Nome	Tipo	Tamanho
books.xml	Documento XML	5 KB
clienteSOAP.php	Ficheiro PHP	1 KB
correr.bat	Ficheiro batch do ...	1 KB
jdom-2.0.6.jar	Executable Jar File	298 KB
jdom-2.0.6-contrib.jar	Executable Jar File	164 KB
jdom-2.0.6-javadoc.jar	Executable Jar File	984 KB
jdom-2.0.6-junit.jar	Executable Jar File	483 KB
jdom-2.0.6-sources.jar	Executable Jar File	841 KB
LICENSE.txt	Documento de te...	3 KB
nusoap.php	Ficheiro PHP	347 KB
pag_carregaFicheiro.php	Ficheiro PHP	1 KB
pag_clienteSOAP.php	Ficheiro PHP	1 KB
PlayWithJDom.class	Ficheiro CLASS	2 KB
PlayWithJDom.java	Ficheiro JAVA	2 KB
README.txt	Documento de te...	5 KB
servicoSOAP.php	Ficheiro PHP	2 KB
simple.xml	Documento XML	2 KB
upload.php	Ficheiro PHP	2 KB
VARIAVEIS AMBIENTE A DEFINIR.txt	Documento de te...	1 KB

servicoSOAP.php – ficheiro que trata de ler o ficheiro XML com o JDOM e de o enviar ao cliente num envelope SOAP

pag_ClienteSOAP.php – cliente SOAP em php que chama o servicoSOAP.php

PHP – SOAP - JDOM

Parte do Código servicoSOAP.php

```php
<?php
require_once "nusoap.php";
function obterDados($nomeFicheiroACorrer) {
    // -- verifica se shell_exec está ativo
    if(!function_exists('shell_exec')) {
        die("A função shell_exec deve estar ativa!");
    }

    // -- gera uma string aleatoria para colocar no nome do ficheiro bat a correr, de modo a evitar sobreposição se o web service for chamado mais que uma vez ao mesmo tempo
    function generateRandomString($length = 10) {
        return substr(str_shuffle(str_repeat($x='0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ', ceil($length/strlen($x)) )),1,$length);
    }

    $nomeFicheiro = generateRandomString(15) . ".bat";


    // -- gerar .bat para correr o JDOM
    // -- -- necessário devido à necessidade de definir várias variaveis PATH e CLASSPATH
    $textoFicheiro = 'SET PATH=%PATH%;C:\testar\orpc.dev\bin\win32
    SET PATH=%PATH%;"C:\Program Files\Java\jdk1.8.0_221\bin"
    SET CLASSPATH=%CLASSPATH%;C:\testar\orpc.dev\lib\orpc.jar;C:\testar\jdom\jdom-2.0.6.jar;
    java PlayWithJDom ' . $nomeFicheiroACorrer;
    // -- coloca conteudo no correr.bat
    file_put_contents($nomeFicheiro, $textoFicheiro);


    // -- corre o ficheiro .bat gerado.
    $output = shell_exec($nomeFicheiro);
    $XML = explode('<?xml version="1.0" encoding="UTF-8"?>', $output)[1];

    // -- remove ficheiro gerado
    unlink($nomeFicheiro);

    return $XML;

}

// -- cria servidor SOAP e envia
$server = new soap_server();
$server->configureWSDL('AddService', 'obtemXML');
$server->wsdl->schemaTargetNamespace = 'obtemXML';
$server->register("obterDados",
    array('tipo' => "xsd:string"),
    array('return' => 'xsd:string')
);
$server->service(file_get_contents('php://input'));
?>
```

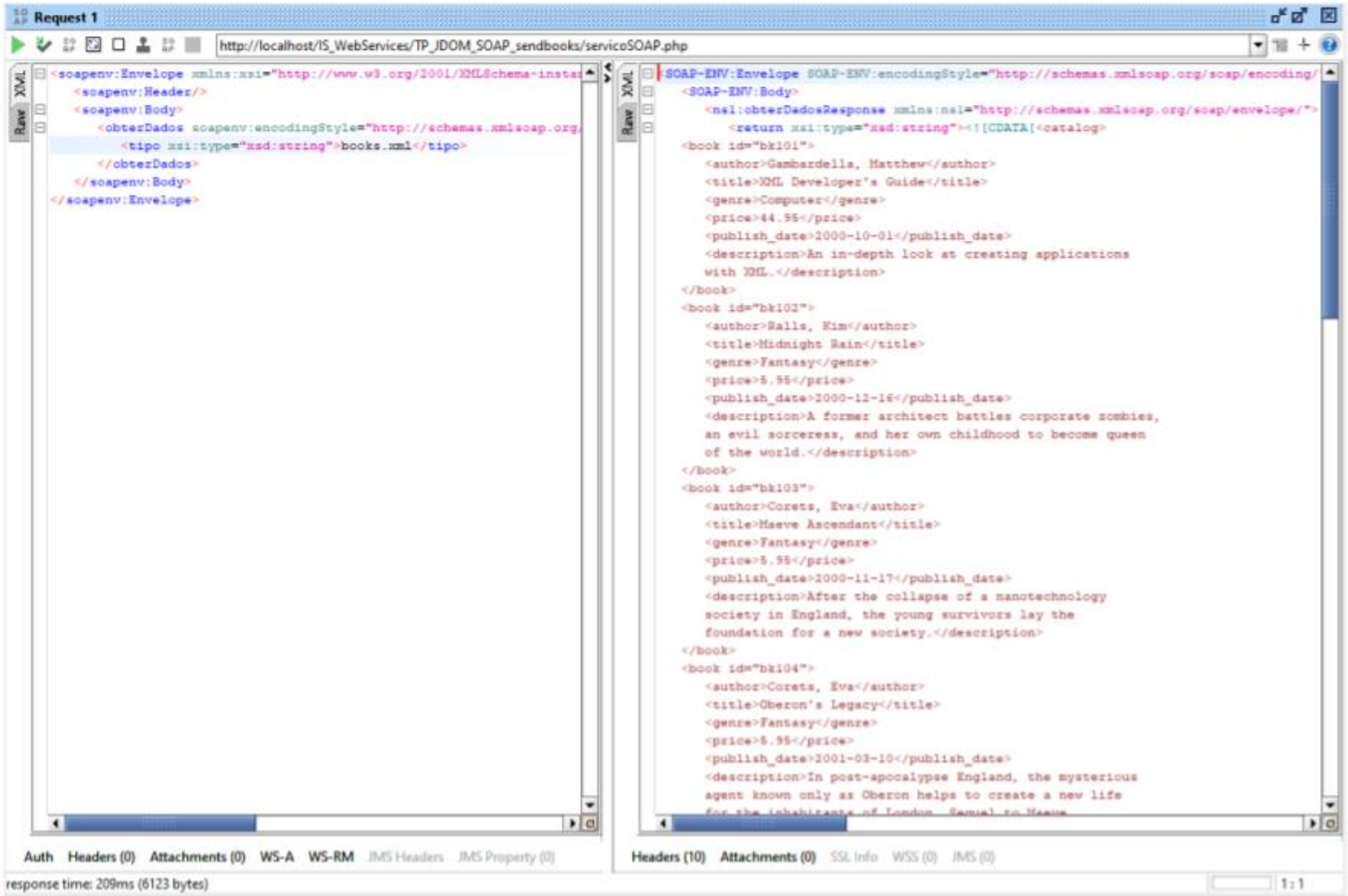
PHP – SOAP - JDOM

Código do pag_clienteSOAP.php

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

$client = new nusoap_client("http://localhost/IS_WebServices/TP_JDOM_SOAP_sendbooks/servicoSOAP.php");


$error = $client->getError();
if ($error) {
    echo "<h2>Constructor error</h2><pre>" . $error . "</pre>";
}

$result = $client->call("obterDados", array("nome" => "books.xml"));

if ($client->fault) {
    echo "<h2>Fault</h2><pre>" . $result . "</pre>";
}
else {
    $error = $client->getError();
    if ($error) {
        echo "<h2>Error</h2><pre>" . $error . "</pre>";
    }
    else {
        header("content-type: text/xml");
        echo $result;
    }
}



?>
```

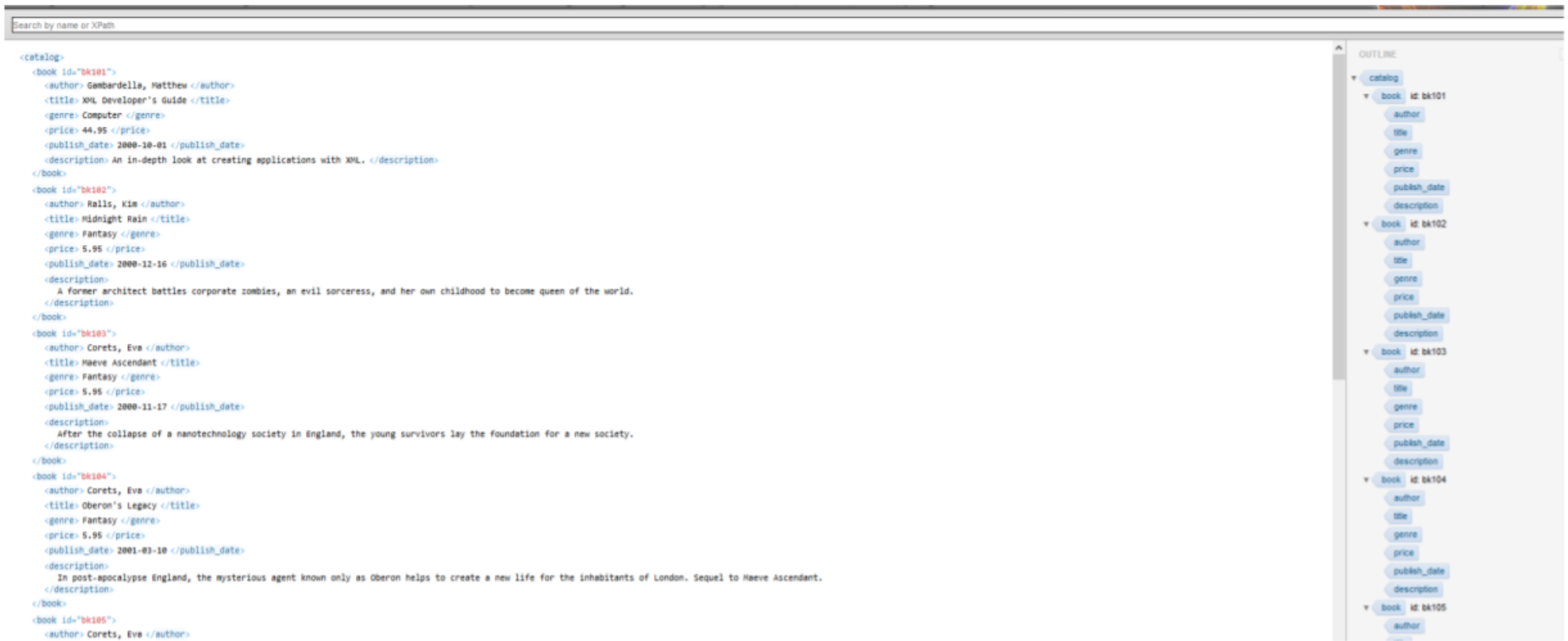
PHP – SOAP - JDOM

Teste SoapUI



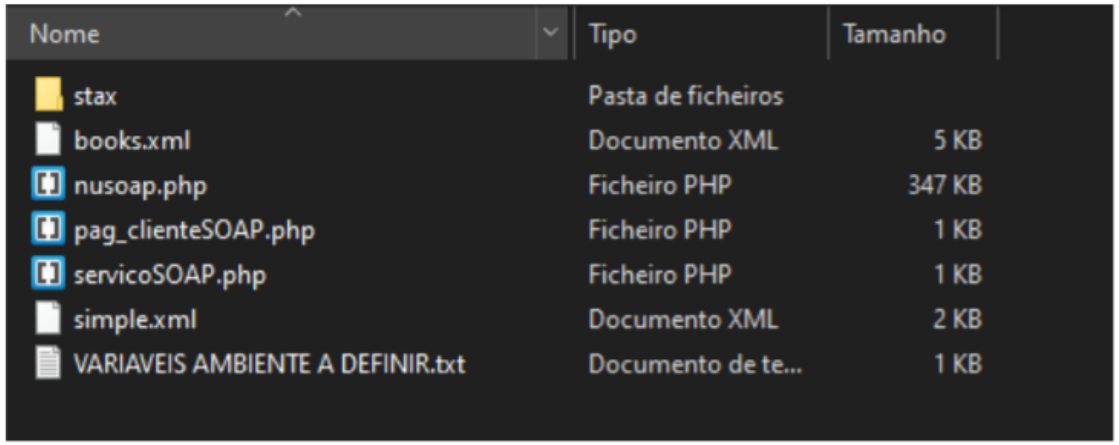
PHP – SOAP - JDOM

Teste Browser (pag_clienteSOAP.php)



PHP – SOAP - JAXP

Estrutura do Projeto Criado



servicoSOAP.php – ficheiro que trata de ler o ficheiro XML com o JAXP e de o enviar ao cliente num envelope SOAP

pag_ClienteSOAP.php – cliente SOAP em php que chama o servicoSOAP.php

PHP – SOAP - JAXP

Parte do Código servicoSOAP.php

```php
<?php
require_once "nusoap.php";
function obterDados($nomeFicheiroACorrer) {
    // -- verifica se shell_exec está ativo
    if(!function_exists('shell_exec')) {
        die("A função shell_exec deve estar ativa!");
    }

    $output = shell_exec('java stax/cursor/CursorParse ' . $nomeFicheiroACorrer);
    $XML = "";

    $XML = explode('<?xml version="1.0" ', $output)[1];
    $XML = explode('?>', $XML)[1];

    $XML = explode(' Parsing Time = ', $XML)[0];

    return $XML;

}

// -- cria servidor SOAP e envia
$server = new soap_server();
$server->configureWSDL('AddService', 'obtemXML');
$server->wsdl->schemaTargetNamespace = 'obtemXML';
$server->register("obterDados",
    array('tipo' => "xsd:string"),
    array('return' => 'xsd:string')
);
$server->service(file_get_contents('php://input'));
?>
```

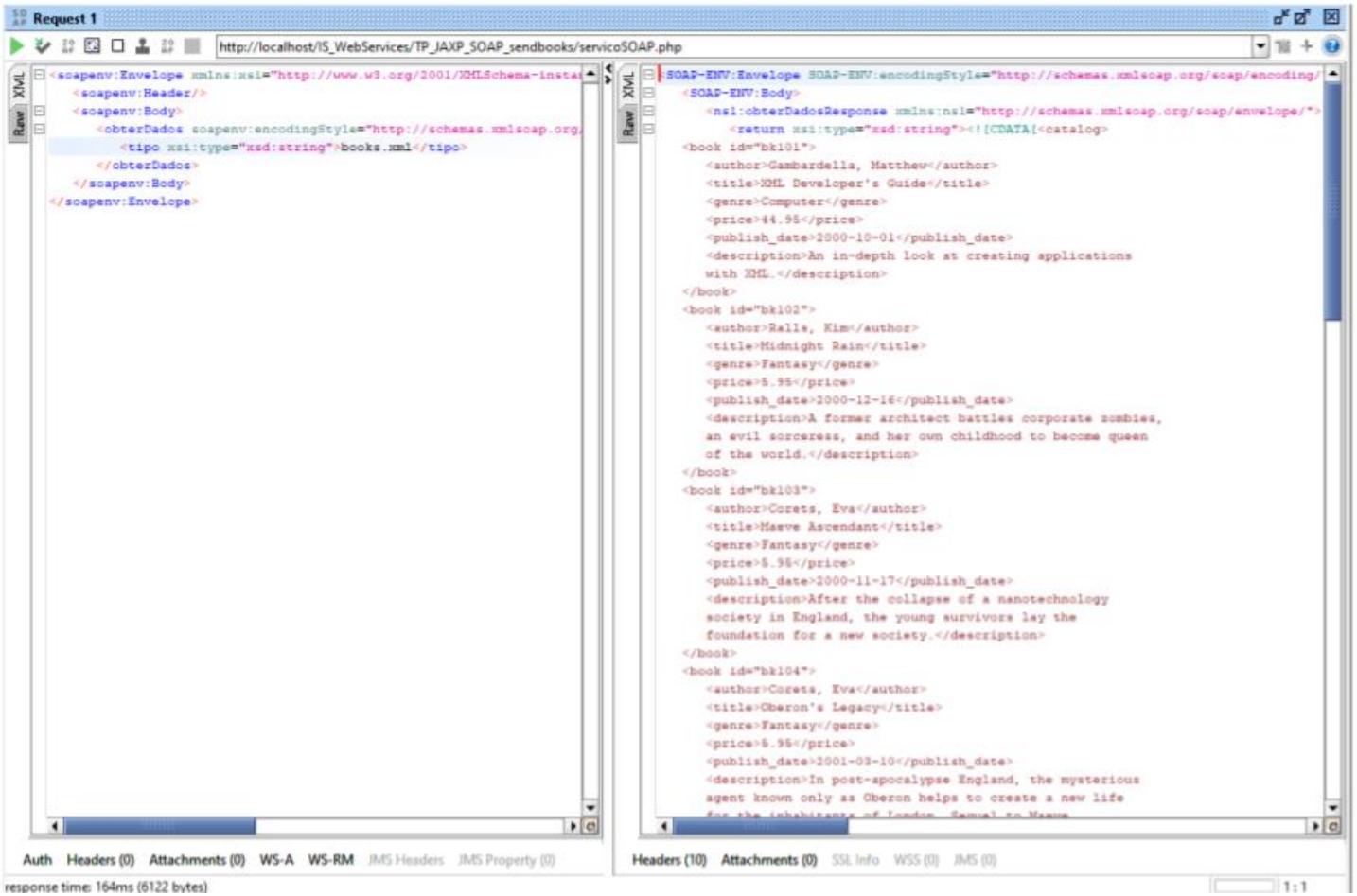
PHP – SOAP - JAXP

Código do pag_clienteSOAP.php

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

$client = new nusoap_client("http://localhost/IS_WebServices/TP_JAXP_SOAP_sendbooks/servicoSOAP.php");


$error = $client->getError();
if ($error) {
    echo "<h2>Constructor error</h2><pre>" . $error . "</pre>";
}

$result = $client->call("obterDados", array("nome" => "books.xml"));

if ($client->fault) {
    echo "<h2>Fault</h2><pre>" . $result . "</pre>";
} else {
    $error = $client->getError();
    if ($error) {
        echo "<h2>Error</h2><pre>" . $error . "</pre>";
    }else {
        header("content-type: text/xml");
        echo $result;
    }
}

```

PHP – SOAP - JAXP

Teste SoapUI



PHP – SOAP - JAXP

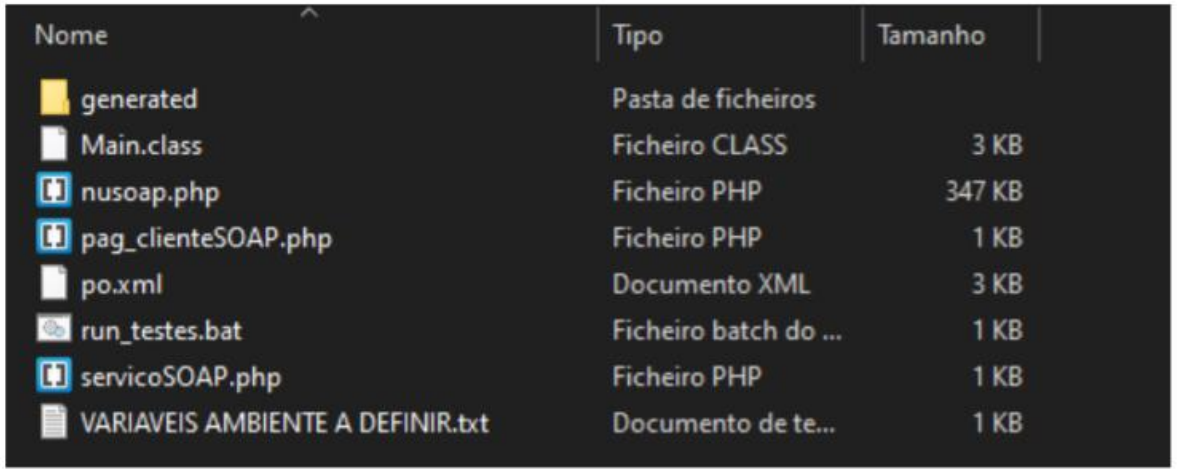
Teste Browser (pag_clienteSOAP.php)

Search by name or XPath


```
<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">
```

PHP – SOAP - JAXB

Estrutura do Projeto Criado



Nome	Tipo	Tamanho
generated	Pasta de ficheiros	
Main.class	Ficheiro CLASS	3 KB
nusoap.php	Ficheiro PHP	347 KB
pag_clienteSOAP.php	Ficheiro PHP	1 KB
po.xml	Documento XML	3 KB
run_testes.bat	Ficheiro batch do ...	1 KB
servicoSOAP.php	Ficheiro PHP	1 KB
VARIAVEIS AMBIENTE A DEFINIR.txt	Documento de te...	1 KB

servicoSOAP.php – ficheiro que trata de ler o ficheiro XML com o JAXB e de o enviar ao cliente num envelope SOAP

pag_ClienteSOAP.php – cliente SOAP em php que chama o servicoSOAP.php

PHP – SOAP - JAXB

Parte do Código servicoSOAP.php

```
P_JAXB_SOAP_sendpo > servicoSOAP.php > ...
1   <?php
2   require_once "nusoap.php";
3   function obterDados($nomeFicheiroACorrer) {
4       // -- verifica se shell_exec está ativo
5       if(!function_exists('shell_exec')) {
6           die("A função shell_exec deve estar ativa!");
7       }
8
9       $output = shell_exec('java Main ' . $nomeFicheiroACorrer);
10      $XML = $output;
11      return $XML;
12
13  }
14
15  // -- cria servidor SOAP e emvia
16  $server = new soap_server();
17  $server->configureWSDL('AddService', 'obtemXML');
18  $server->wsdl->schemaTargetNamespace = 'obtemXML';
19  $server->register("obterDados",
20      array('tipo' => "xsd:string"),
21      array('return' => 'xsd:string')
22  );
23  $server->service(file_get_contents('php://input'));
24  ?>
```

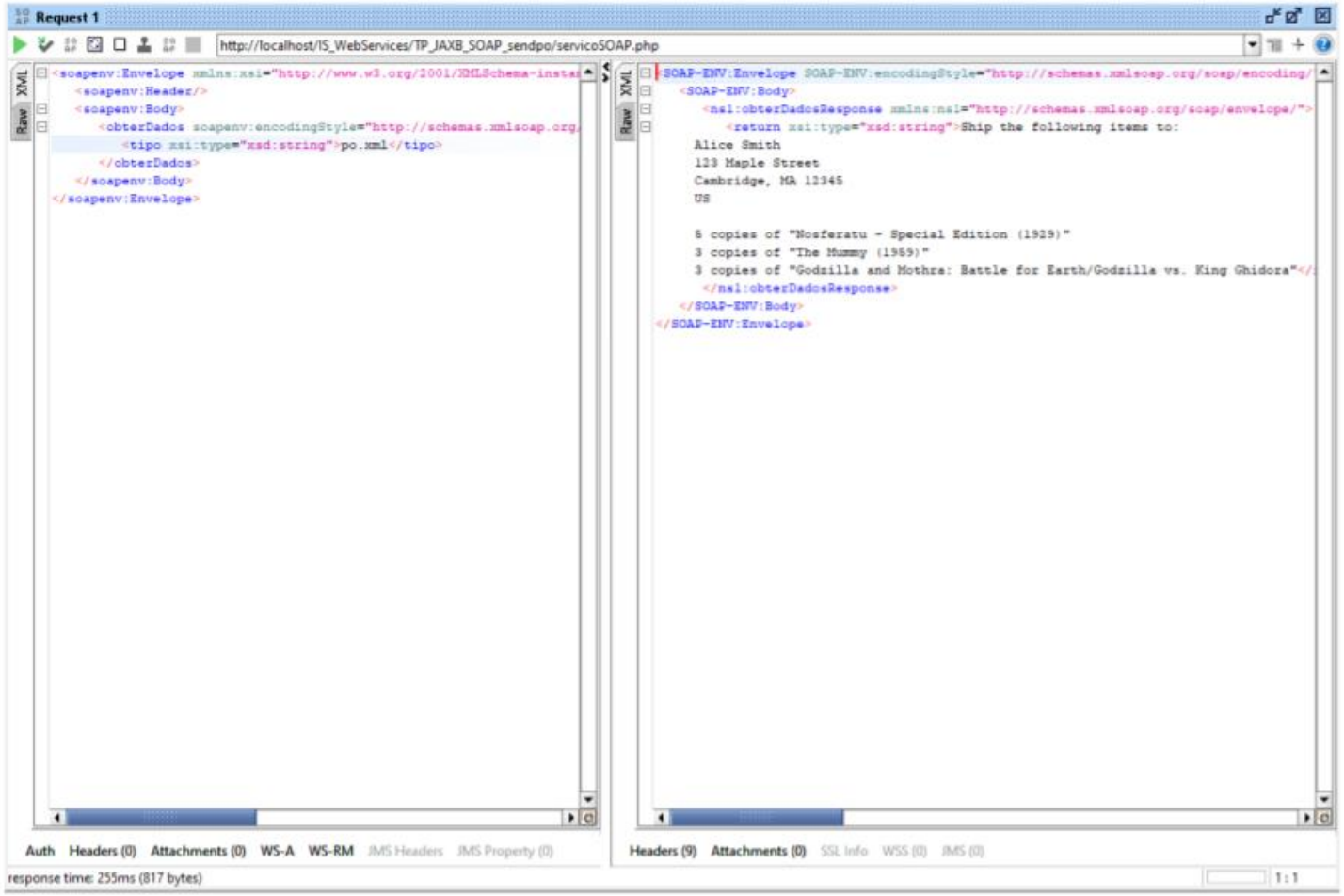
PHP – SOAP - JAXB

Código do pag_clienteSOAP.php

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

$client = new nusoap_client("http://localhost/IS_WebServices/TP_JAXB_SOAP_sendpo/servicoSOAP.php");


$error = $client->getError();
if ($error) {
    echo "<h2>Constructor error</h2><pre>" . $error . "</pre>";
}

$result = $client->call("obterDados", array("nome" => "po.xml"));

if ($client->fault) {
    echo "<h2>Fault</h2><pre>" . $result . "</pre>";
} else {
    $error = $client->getError();
    if ($error) {
        echo "<h2>Error</h2><pre>" . $error . "</pre>";
    }else {
        header("content-type: text/plain");
        echo $result;
    }
}
```

PHP – SOAP - JAXB

Teste SoapUI



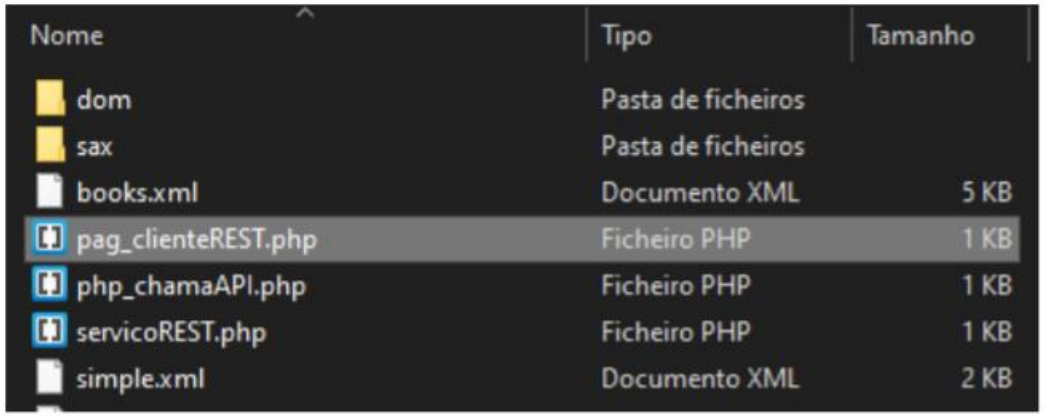
PHP – SOAP - JAXB

Teste Browser (pag_clienteSOAP.php)

```
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 (1959)"
        3 copies of "Godzilla and Mothra: Battle for Earth/Godzilla vs. King Ghidora"
```

PHP – REST - DOM

Estrutura do Projeto Criado



servicoREST.php – ficheiro que trata de ler o ficheiro XML com o DOM e de o enviar ao client

pag_ClienteREST.php – cliente REST em php que chama o servicoREST.php através do php_chamaAPI.php.

PHP – REST - DOM

Parte do Código servicoREST.php

TP_DOM_REST_sendbooks > servicoREST.php > ...

```php
<?php
function obterDados($nomeFicheiroACorrer) {
    // -- verifica se shell_exec está ativo
    if(!function_exists('shell_exec')) {
        die("A função shell_exec deve estar ativa!");
    }

    $output = shell_exec('java dom/DOMEcho ' . $nomeFicheiroACorrer);
    $XML = "";

    $XML = "DOC: nodeName=" . explode('DOC: nodeName=', $output)[1];
    $XML = explode(' Parsing Time = ', $XML)[0];

    return $XML;

}

// -- envia dados

// -- -- nome de ficheiro enviado por get
if(!empty($_GET['nomeFicheiroXML'])){
    echo obterDados($_GET['nomeFicheiroXML']);
} else {
    echo "Ocorreu um erro -- não foi dado um nome de ficheiro";
}


?>
```

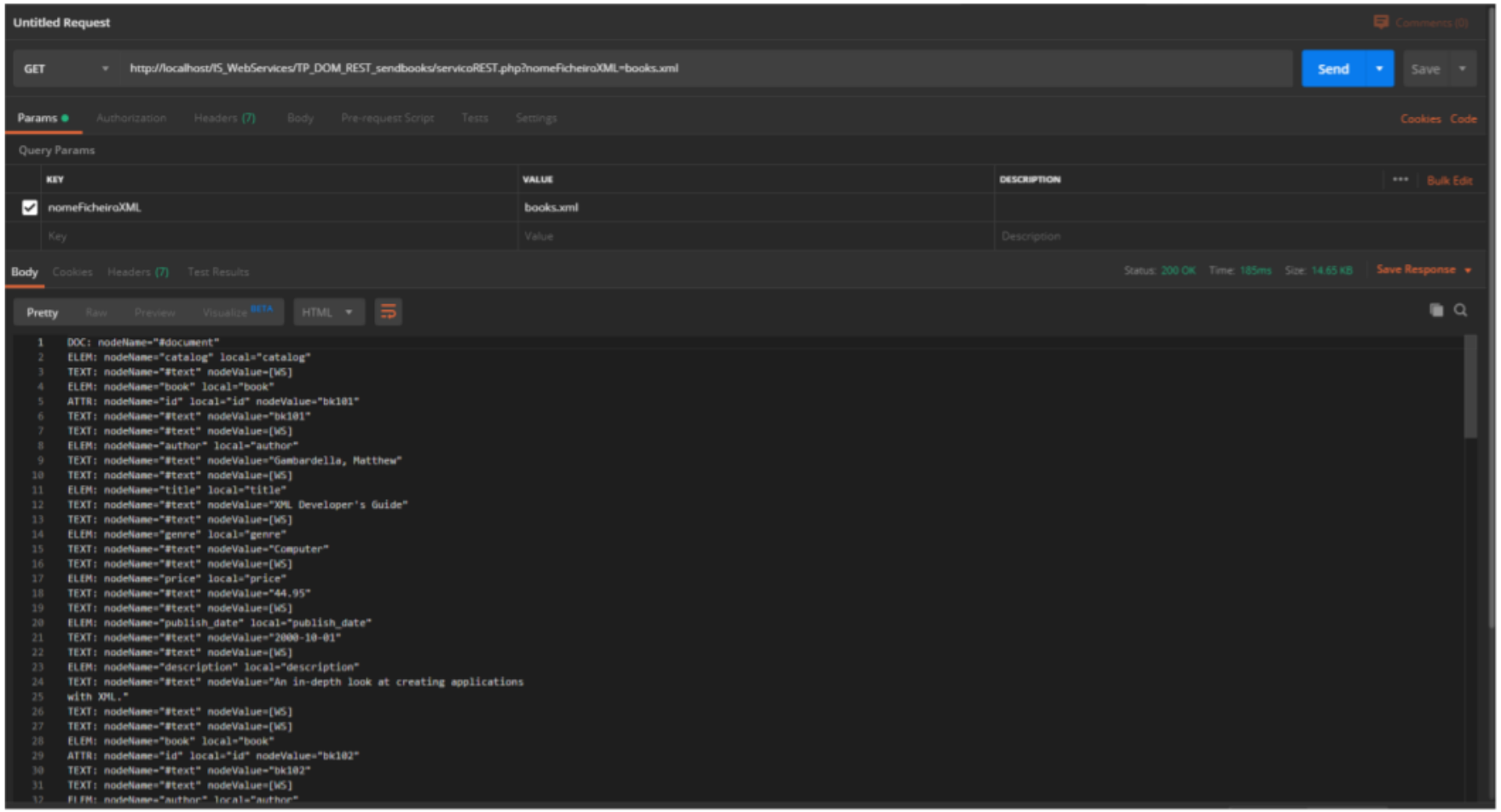
PHP – REST - DOM

Código do pag_clienteREST.php

```php
<?php
    require("php_chamaAPI.php");
    $url = "http://localhost/IS_WebServices/TP_DOM_REST_sendbooks/servicoREST.php";
    $nomeFicheiro = "books.xml";

    $make_call = callAPI('GET', $url . "?nomeFicheiroXML=" . urldecode($nomeFicheiro), null);
    $response = $make_call;

    header("Content-Type: text/plain");
    echo $response;
?>
```

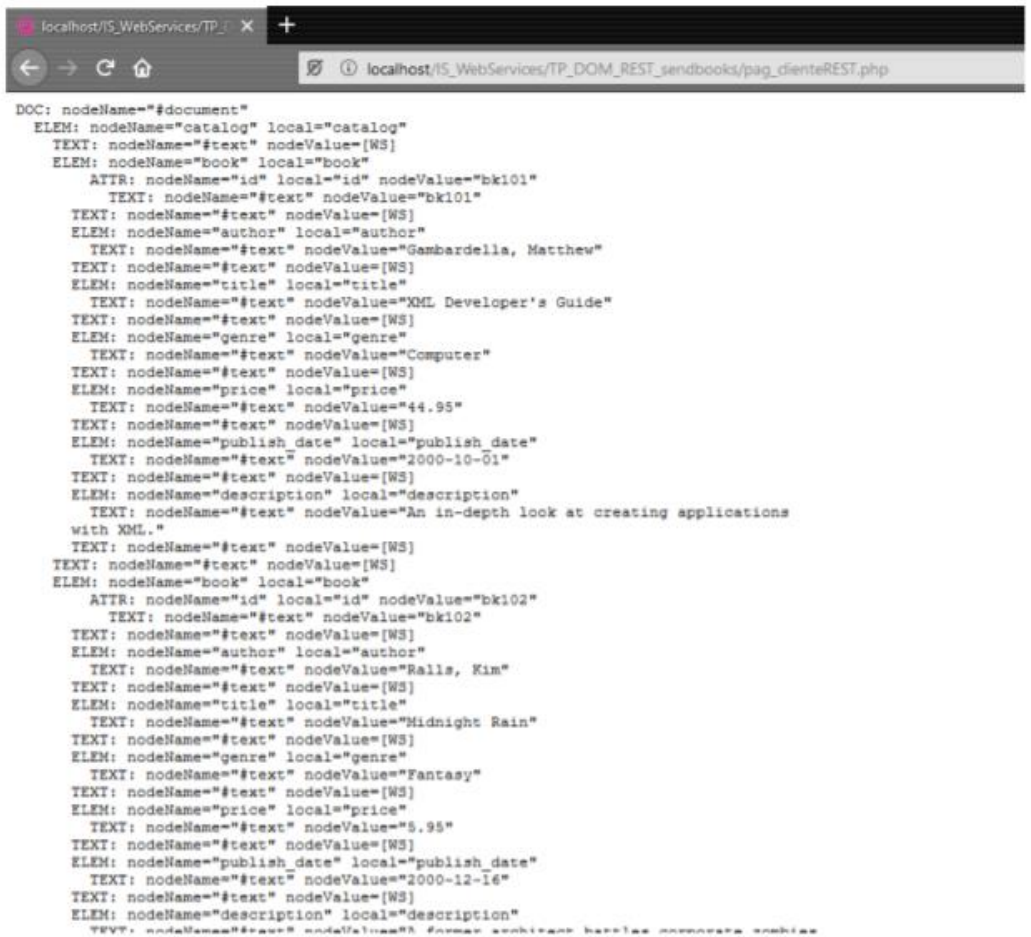
PHP – REST - DOM

Teste POSTMAN



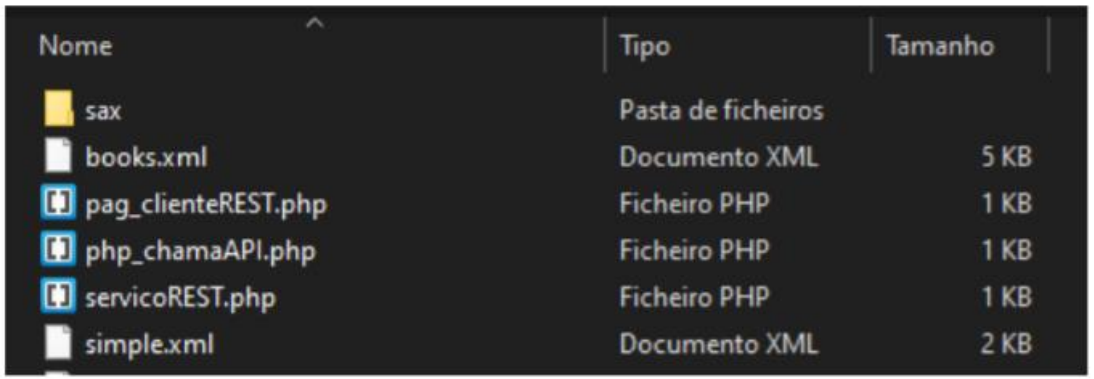
PHP – REST - DOM

Teste Browser (pag_clienteREST.php)



PHP – REST - SAX

Estrutura do Projeto Criado



servicoREST.php – ficheiro que trata de ler o ficheiro XML com o SAX e de enviar o resultado ao cliente

pag_ClienteREST.php – cliente REST em php que chama o servicoREST.php através do php_chamaAPI.php.

PHP – REST - SAX

Parte do Código servicoREST.php

TP_SAX_REST_sendbooks > servicoREST.php > ...

```php
<?php
function obterDados($nomeFicheiroACorrer) {
    // -- verifica se shell_exec está ativo
    if(!function_exists('shell_exec')) {
        die("A função shell_exec deve estar ativa!");
    }

    $output = shell_exec('java sax/SAXLocalNameCount ' . $nomeFicheiroACorrer);
    $XML = "";

    $XML = explode('Local Name', $output, 1);
    $XML = substr($output, strpos($output, "Local Name"));

    return $XML;
}

// -- envia dados

// -- -- nome de ficheiro enviado por get
if(!empty($_GET['nomeFicheiroXML'])){
    echo obterDados($_GET['nomeFicheiroXML']);
} else {
    echo "Ocorreu um erro -- não foi dado um nome de ficheiro";
}
?>
```

PHP – REST - SAX

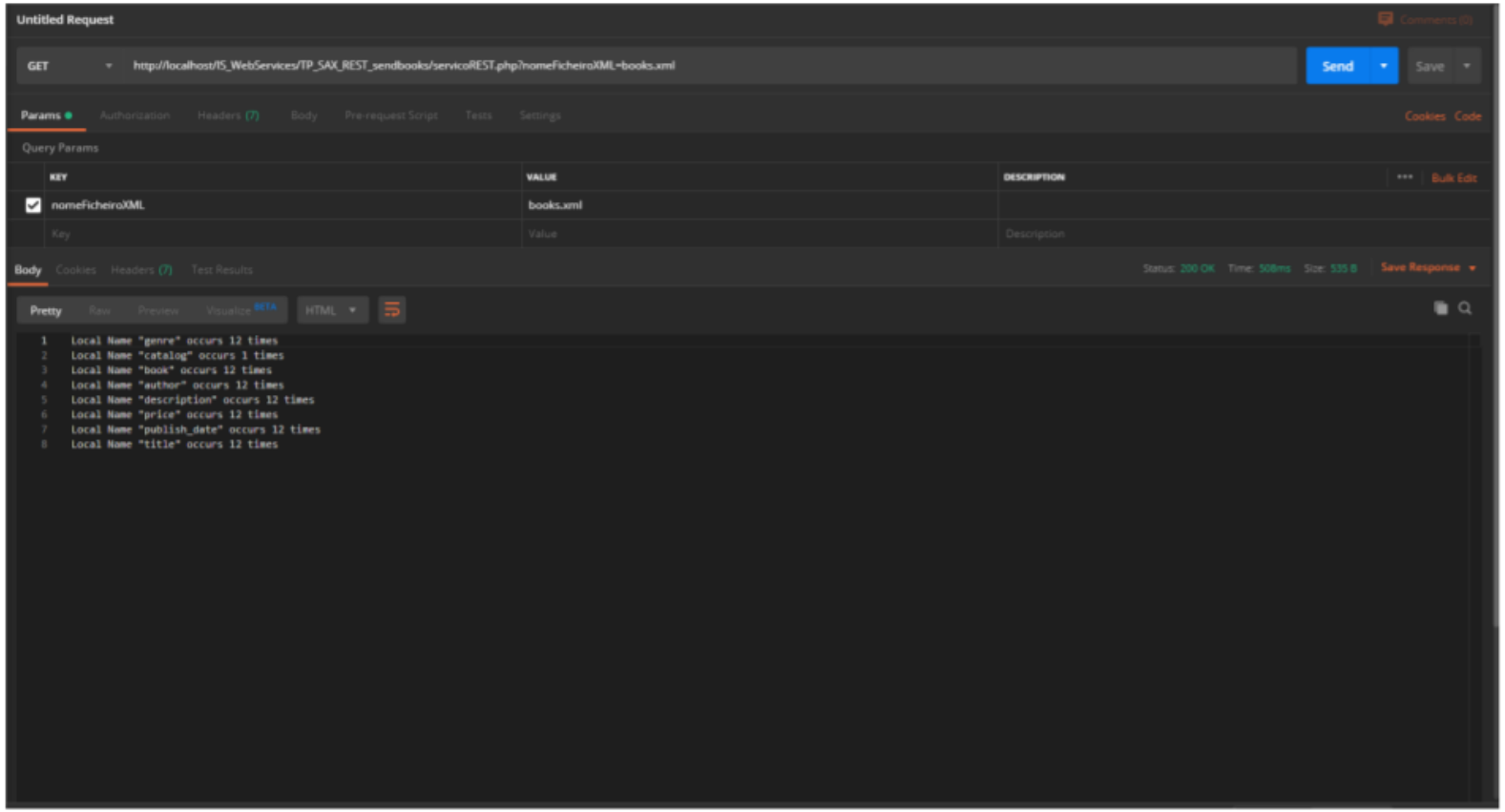
Código do pag_clienteREST.php

TP_SAX_REST_sendbooks > pag_clienteREST.php > ...

```php
<?php
    require("php_chamaAPI.php");
    $url = "http://localhost/IS_WebServices/TP_SAX_REST_sendbooks/servicoREST.php";
    $nomeFicheiro = "books.xml";

    $make_call = callAPI('GET', $url . "?nomeFicheiroXML=" . urldecode($nomeFicheiro), null);
    $response = $make_call;

    header("Content-Type: text/plain");
    echo $response;
?>
```

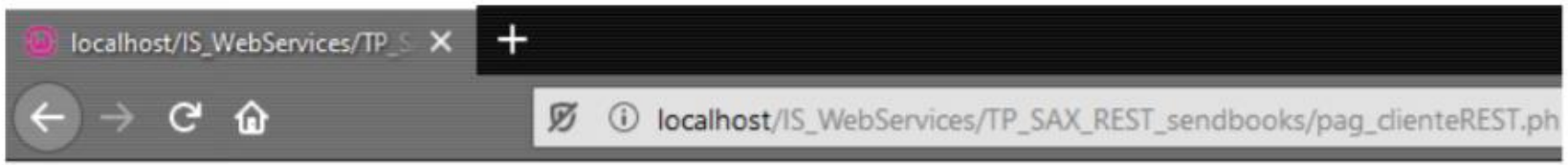
PHP – REST - SAX

Teste POSTMAN



PHP – REST - SAX

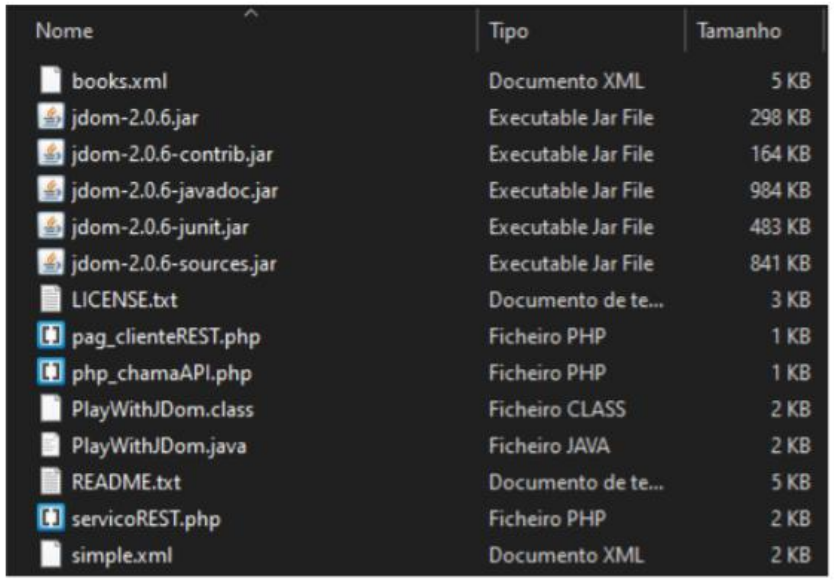
Teste Browser (pag_clienteREST.php)




```
Local Name "genre" occurs 12 times
Local Name "catalog" occurs 1 times
Local Name "book" occurs 12 times
Local Name "author" occurs 12 times
Local Name "description" occurs 12 times
Local Name "price" occurs 12 times
Local Name "publish_date" occurs 12 times
Local Name "title" occurs 12 times
```

PHP – REST - JDOM

Estrutura do Projeto Criado



Nome	Tipo	Tamanho
books.xml	Documento XML	5 KB
jdom-2.0.6.jar	Executable Jar File	298 KB
jdom-2.0.6-contrib.jar	Executable Jar File	164 KB
jdom-2.0.6-javadoc.jar	Executable Jar File	984 KB
jdom-2.0.6-junit.jar	Executable Jar File	483 KB
jdom-2.0.6-sources.jar	Executable Jar File	841 KB
LICENSE.txt	Documento de te...	3 KB
pag_clienteREST.php	Ficheiro PHP	1 KB
php_chamaAPI.php	Ficheiro PHP	1 KB
PlayWithJDom.class	Ficheiro CLASS	2 KB
PlayWithJDom.java	Ficheiro JAVA	2 KB
README.txt	Documento de te...	5 KB
servicoREST.php	Ficheiro PHP	2 KB
simple.xml	Documento XML	2 KB

servicoREST.php – ficheiro que trata de ler o ficheiro XML com o JDOM e de o enviar ao client

pag_ClienteREST.php – cliente REST em php que chama o servicoREST.php através do php_chamaAPI.php.

PHP – REST - JDOM

Parte do Código servicoREST.php

```php
<?php
function obterDados($nomeFicheiroACorrer) {
    // -- verifica se shell_exec está ativo
    if(!function_exists('shell_exec')) {
        die("A função shell_exec deve estar ativa!");
    }

    // -- gera uma string aleatoria para colocar no nome do ficheiro bat a correr, de modo a evitar sobreposição se o web service for chamado mais que uma vez ao mesmo tempo
    function generateRandomString($length = 10) {
        return substr(str_shuffle(str_repeat($x='0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ', ceil($length/strlen($x)) )),1,$length);
    }

    $nomeFicheiro = generateRandomString(15) . ".bat";


    // -- gerar .bat para correr o JDOM
    // -- -- necessário devido à necessidade de definir várias variaveis PATH e CLASSPATH
    $textoFicheiro = 'SET PATH=%PATH%;C:\testar\orpc.dev\bin\win32
    SET PATH=%PATH%;"C:\Program Files\Java\jdk1.8.0_221\bin"
    SET CLASSPATH=%CLASSPATH%;C:\testar\orpc.dev\lib\orpc.jar;C:\testar\jdom\jdom-2.0.6.jar;
    java PlayWithJDom ' . $nomeFicheiroACorrer;
    // -- coloca conteudo no correr.bat
    file_put_contents($nomeFicheiro, $textoFicheiro);


    // -- corre o ficheiro .bat gerado.
    $output = shell_exec($nomeFicheiro);
    $XML = explode('<?xml version="1.0" encoding="UTF-8"?>', $output)[1];

    // -- remove ficheiro gerado
    unlink($nomeFicheiro);

    return $XML;

}

// -- envia dados

// -- -- nome de ficheiro enviado por get
if(!empty($_GET['nomeFicheiroXML'])){
    echo obterDados($_GET['nomeFicheiroXML']);
} else {
    echo "Ocorreu um erro -- não foi dado um nome de ficheiro";
}

?>
```

PHP – REST - JDOM

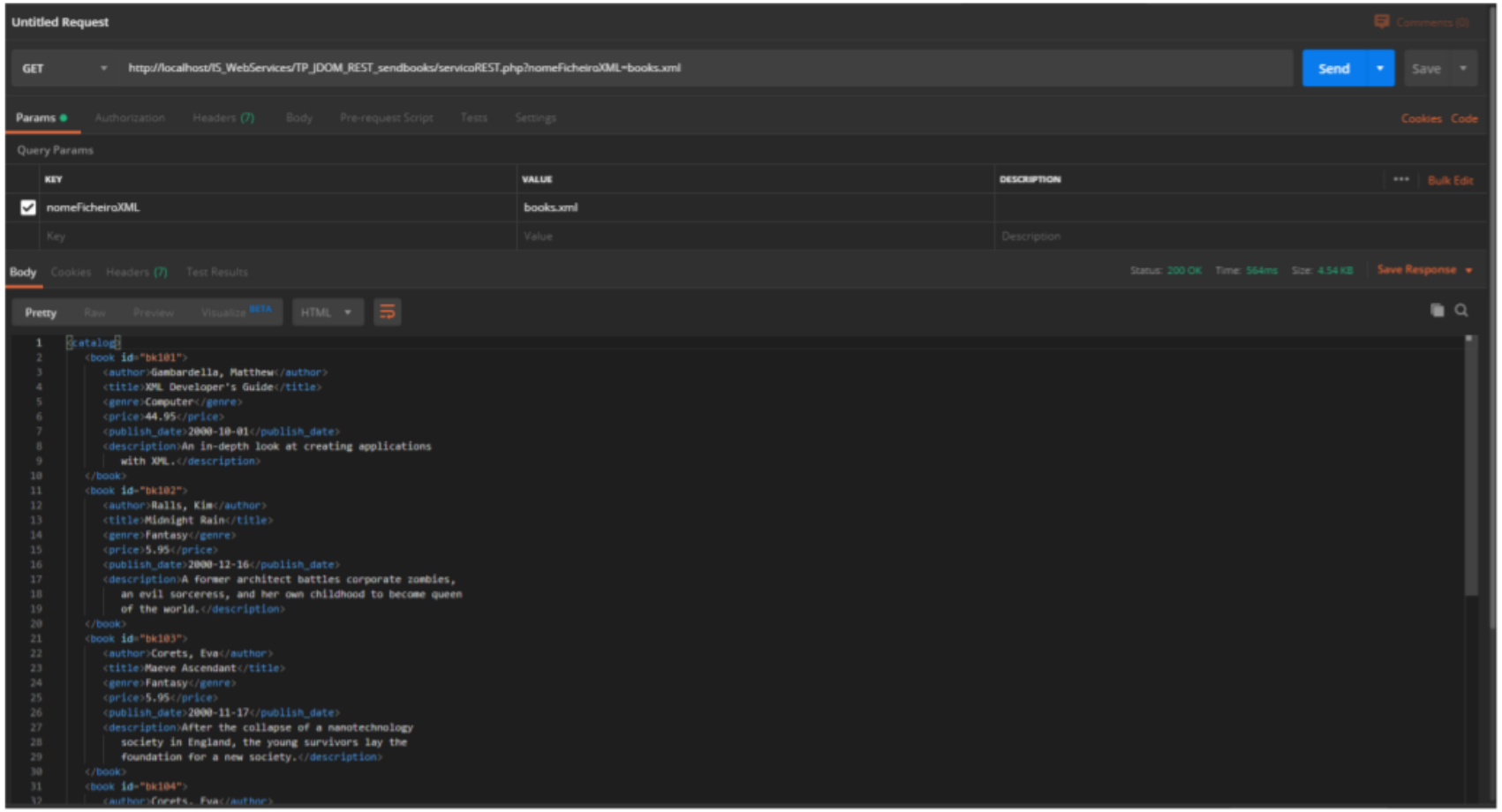
Código do pag_clienteREST.php

TP_JDOM_REST_sendbooks > pag_clienteREST.php > ...

```php
<?php
    require("php_chamaAPI.php");
    $url = "http://localhost/IS_WebServices/TP_JDOM_REST_sendbooks/servicoREST.php";
    $nomeFicheiro = "books.xml";

    $make_call = callAPI('GET', $url . "?nomeFicheiroXML=" . urldecode($nomeFicheiro), null);
    $response = $make_call;

    header("Content-Type: text/xml");
    echo $response;
?>
```

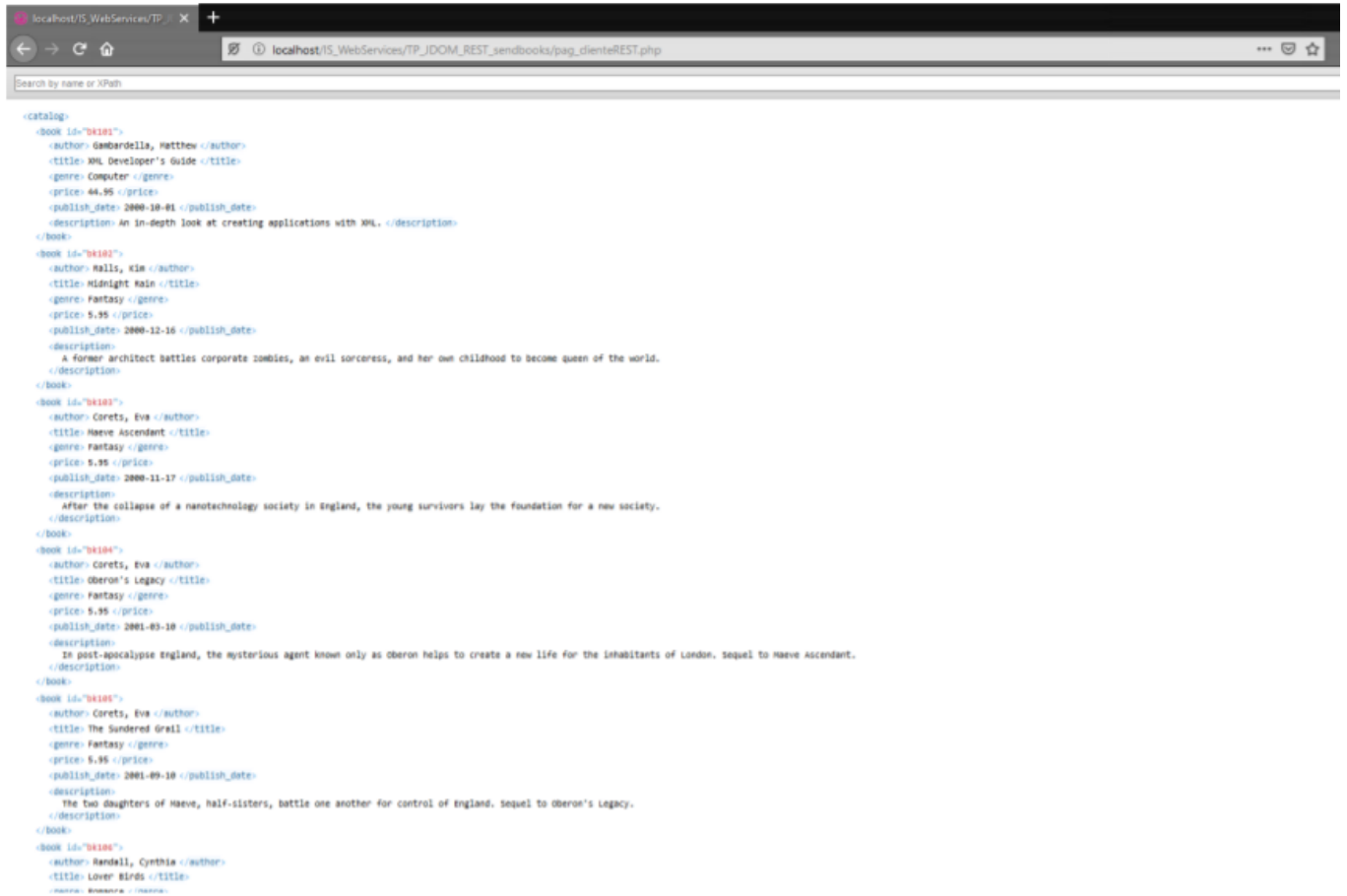
PHP – REST - JDOM

Teste POSTMAN



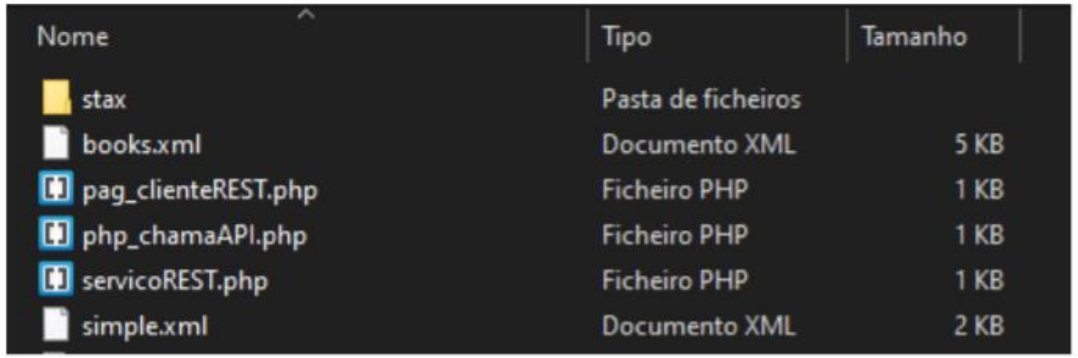
PHP – REST - JDOM

Teste Browser (pag_RESTphp)



PHP – REST - JAXP

Estrutura do Projeto Criado



servicoREST.php – ficheiro que trata de ler o ficheiro XML com o JAXP e de o enviar ao client

pag_ClienteREST.php – cliente REST em php que chama o servicoREST.php através do php_chamaAPI.php.

PHP – REST - JAXP

Parte do Código servicoREST.php

TP_JAXP_REST_sendbooks > servicoREST.php > ...

```php
<?php
function obterDados($nomeFicheiroACorrer) {
    // -- verifica se shell_exec está ativo
    if(!function_exists('shell_exec')) {
        die("A função shell_exec deve estar ativa!");
    }

    $output = shell_exec('java stax/cursor/CursorParse ' . $nomeFicheiroACorrer);
    $XML = "";

    $XML = explode('<?xml version="1.0" ', $output)[1];
    $XML = explode('?>', $XML)[1];

    $XML = explode(' Parsing Time = ', $XML)[0];

    return $XML;
}

// -- envia dados

// -- -- nome de ficheiro enviado por get
if(!empty($_GET['nomeFicheiroXML'])){
    echo obterDados($_GET['nomeFicheiroXML']);
} else {
    echo "Ocorreu um erro -- não foi dado um nome de ficheiro";
}

?>
```

PHP – REST - JAXP

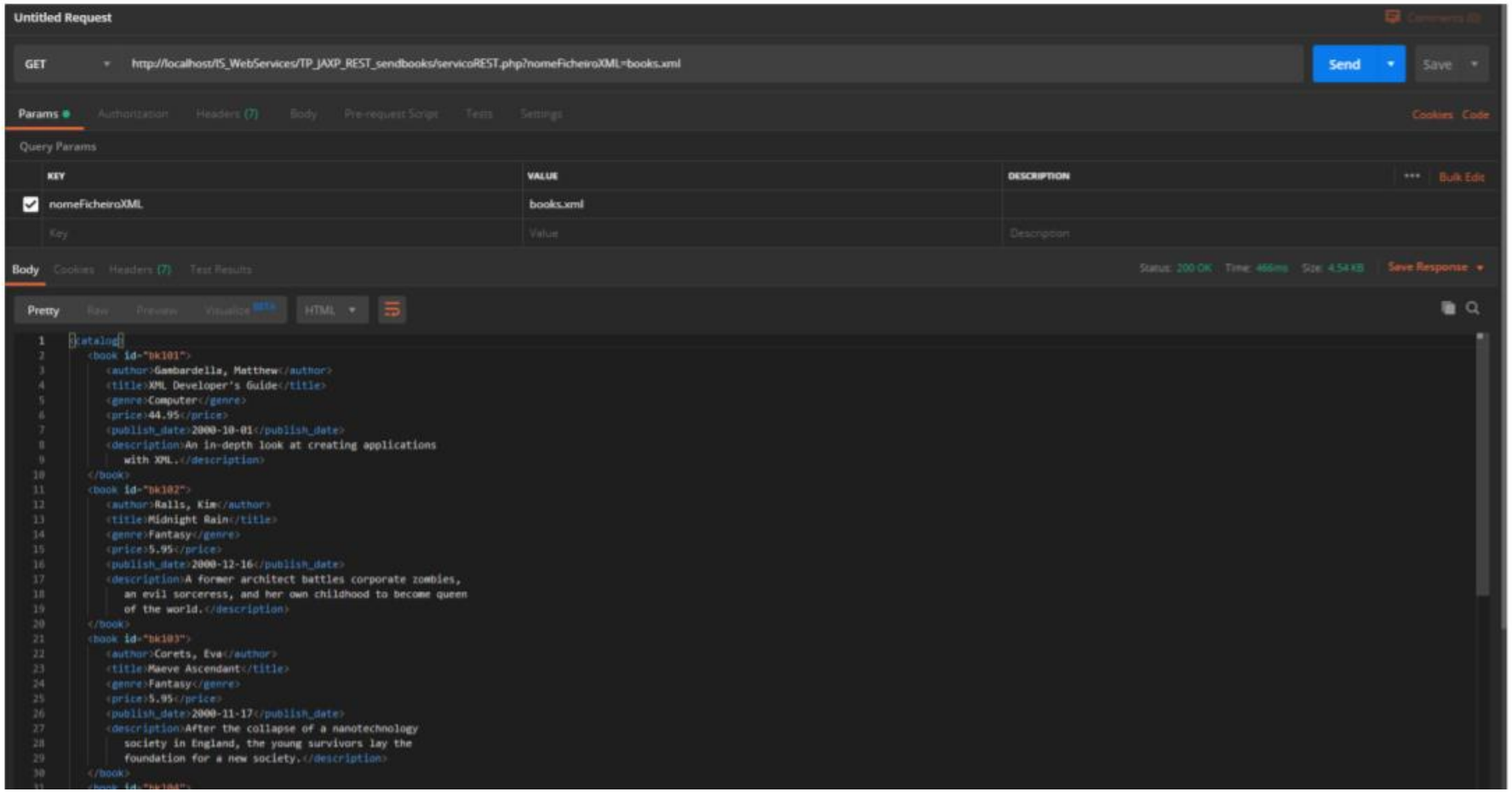
Código do pag_clienteREST.php

TP_JAXP_REST_sendbooks > pag_clienteREST.php > ...

```php
<?php
    require("php_chamaAPI.php");
    $url = "http://localhost/IS_WebServices/TP_JAXP_REST_sendbooks/servicoREST.php";
    $nomeFicheiro = "books.xml";

    $make_call = callAPI('GET', $url . "?nomeFicheiroXML=" . urldecode($nomeFicheiro), null);
    $response = $make_call;

    header("Content-Type: text/xml");
    echo $response;
?>
```

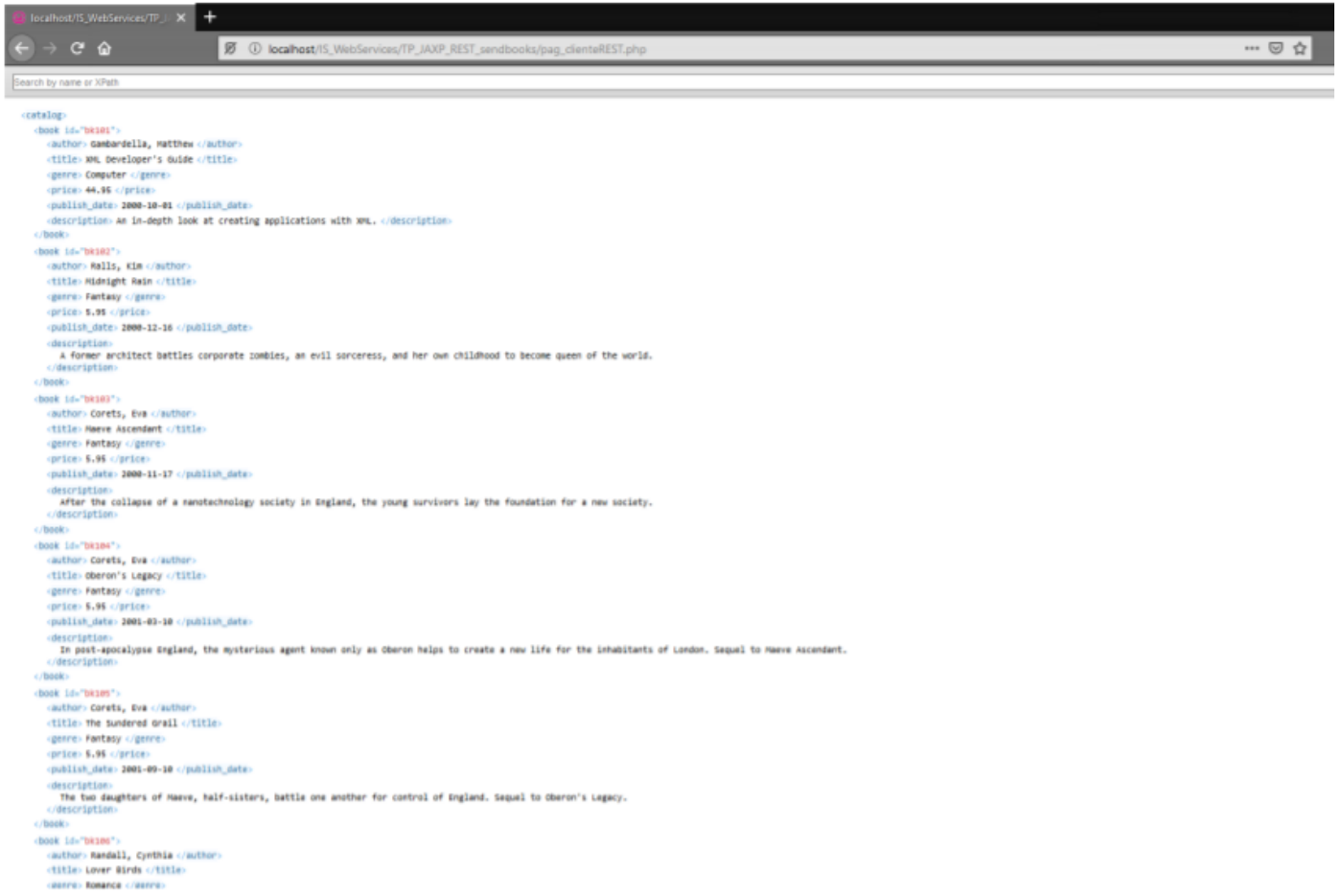
PHP – REST - JAXP

Teste POSTMAN



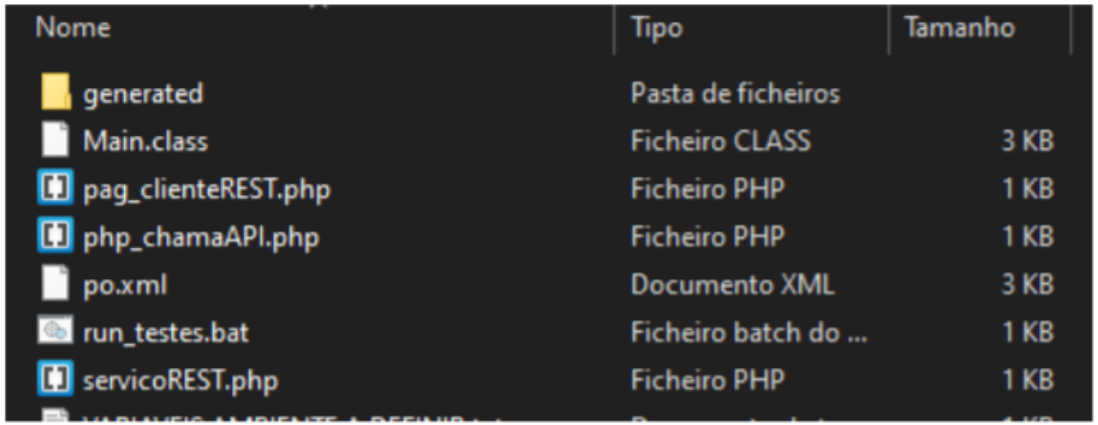
PHP – REST - JAXP

Teste Browser (pag_clienteREST.php)



PHP – REST - JAXB

Estrutura do Projeto Criado



servicoREST.php – ficheiro que trata de ler o ficheiro XML com o JAXB e de o enviar ao client

pag_ClienteREST.php – cliente REST em php que chama o servicoREST.php através do php_chamaAPI.php.

PHP – REST - JAXB

Parte do Código servicoREST.php

TP_JAXB_REST_sendpo > servicoREST.php > ...

```php
<?php
function obterDados($nomeFicheiroACorrer) {
    // -- verifica se shell_exec está ativo
    if(!function_exists('shell_exec')) {
        die("A função shell_exec deve estar ativa!");
    }

    $output = shell_exec('java Main ' . $nomeFicheiroACorrer);
    $XML = $output;
    return $XML;
}

// -- envia dados

// -- -- nome de ficheiro enviado por get
if(!empty($_GET['nomeFicheiroXML'])){
    echo obterDados($_GET['nomeFicheiroXML']);
} else {
    echo "Ocorreu um erro -- não foi dado um nome de ficheiro";
}
?>
```

PHP – REST - JAXB

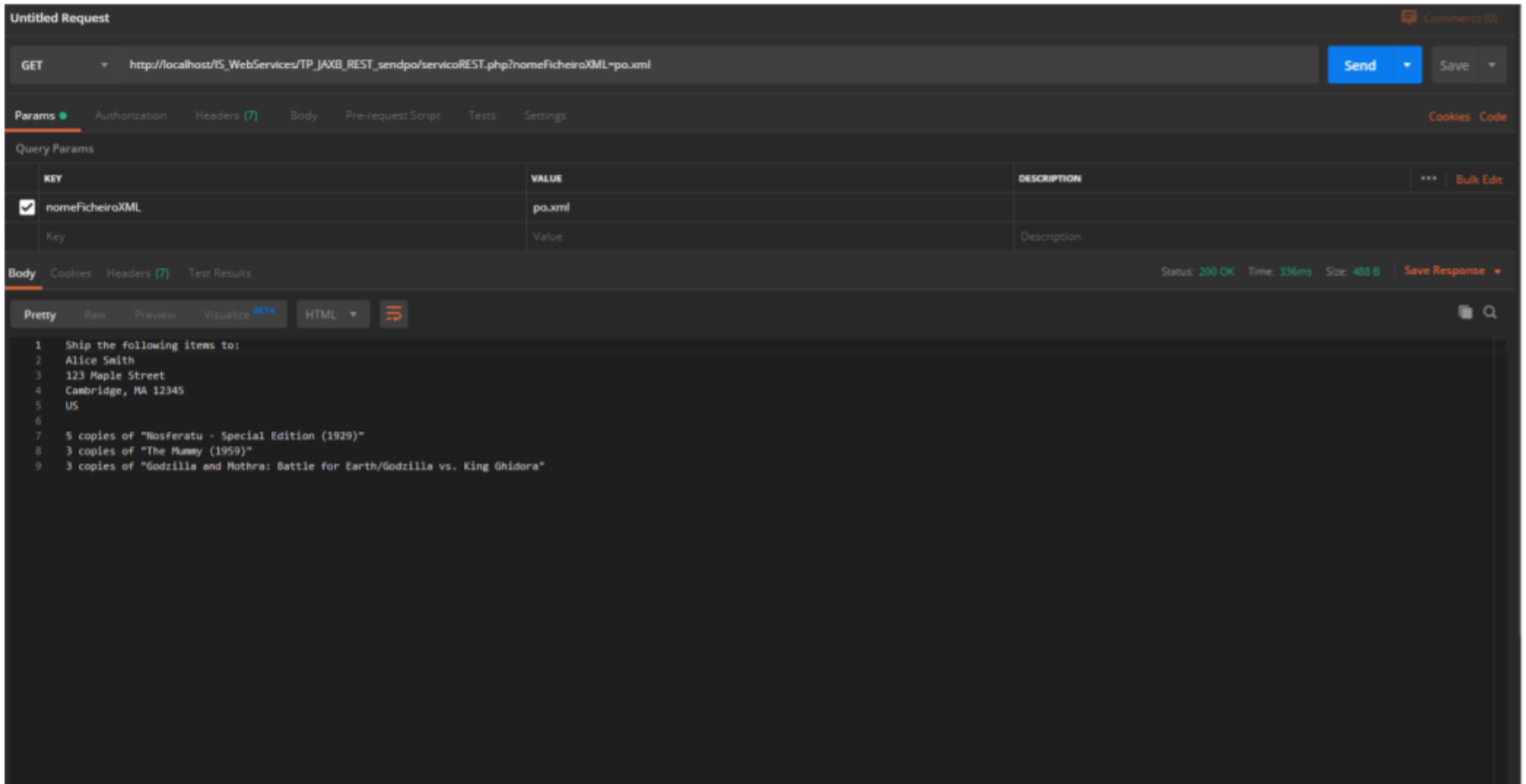
Código do pag_clienteREST.php

TP_JAXB_REST_sendpo > pag_clienteREST.php > ...

```php
<?php
    require("php_chamaAPI.php");
    $url = "http://localhost/IS_WebServices/TP_JAXB_REST_sendpo/servicoREST.php";
    $nomeFicheiro = "po.xml";

    $make_call = callAPI('GET', $url . "?nomeFicheiroXML=" . urldecode($nomeFicheiro), null);
    $response = $make_call;

    header("Content-Type: text/plain");
    echo $response;
?>
```

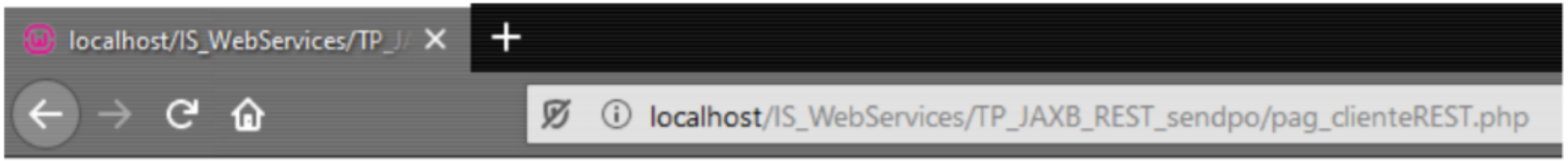
PHP – REST - JAXB

Teste POSTMAN



PHP – REST - JAXB

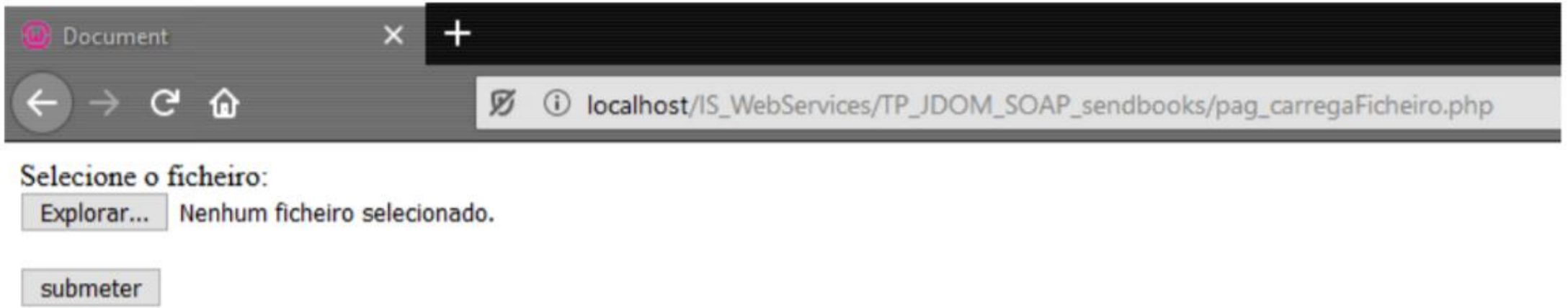
Teste Browser (pag_clienteREST.php)




```
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 (1959)"
        3 copies of "Godzilla and Mothra: Battle for Earth/Godzilla vs. King Ghidora"
```

PHP – SOAP – JDOM com FileUpload

Página PHP para submeter o ficheiro xml:



PHP – SOAP – JDOM com FileUpload

Código:

TP_JDOM_SOAP_sendbooks > pag_carregaFicheiro.php

```php
<!DOCTYPE html>
<html lang="en">
    <head>
        <?php
            header("Access-Control-Allow-Origin: *");
            header("Access-Control-Allow-Methods: OPTIONS, GET, POST, PUT, DELETE");
            header("Access-Control-Allow-Headers: Content-Type");
        ?>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <meta http-equiv="X-UA-Compatible" content="ie=edge">
        <title>Document</title>
        <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
    </head>
    <body>

        <form action="upload.php" enctype="multipart/form-data" method="POST">
            Selecione o ficheiro:
            <br>
            <input type="file" name="ficheiro" />
            <br><br>
            <input type="submit" value="submeter" />
        </form>
    </body>
</html>
```

PHP – SOAP – JDOM com FileUpload

Código para upload:

TP_JDOM_SOAP_sendbooks > upload.php > ...

```php
<?php
    header("Content-type: text/xml");
    include("clienteSOAP.php");

    if(!isset($_FILES['ficheiro'])){
        exit('Ficheiro não enviado');
    }
    // verificar erro
    if($_FILES['ficheiro']['error']){
        exit('Erro no upload');
    }
    // -- tamanho do ficheiro
    if($_FILES['ficheiro']['size'] > 40000000){
        exit("ficheiro demasiado grande");
    }
    // -- tipo ficheiro
    if($_FILES['ficheiro']['type'] != "text/xml"){
        exit("tipo de ficheiro não suportado");
    }
    $name = $_FILES['ficheiro']['name'];
    $ficheiro_final = $name;

    if(file_exists($ficheiro_final)){
        //-- tocar ficheiro
        unlink($ficheiro_final);
    }

    if(!move_uploaded_file($_FILES['ficheiro']['tmp_name'], $ficheiro_final)){
        exit("Ocorreu um erro ao carregar!");
    }
    echo '<?xml version="1.0" encoding="ISO-8859-1"?>';
    echo clienteSOAP($ficheiro_final);
?>
```

PHP – SOAP – JDOM com FileUpload

Código da function clienteSOAP(string nomeFicheiro):

```php
<?php
require_once "nusoap.php";
function clienteSOAP($nomeFicheiroCarregado = "books.xml"){

    $client = new nusoap_client("http://localhost/IS_WebServices/TP_JDOM_SOAP_sendbooks/servicoSOAP.php");
    $error = $client->getError();
    if ($error) {
        echo "<h2>Constructor error</h2><pre>" . $error . "</pre>";
    }

    $result = $client->call("obterDados", array("nome" => $nomeFicheiroCarregado));

    if ($client->fault) {
        return "<h2>Fault</h2><pre>" . $result . "</pre>";
    }
    else {
        $error = $client->getError();
        if ($error) {
            return "<h2>Error</h2><pre>" . $error . "</pre>";
        }
        else {
            return $result;
        }
    }
}

?>
```

PHP – SOAP – JDOM com FileUpload

Teste Upload e visualização com o simple.xml:

PHP – SOAP – JDOM com FileUpload

Teste Upload e visualização com o simple.xml: