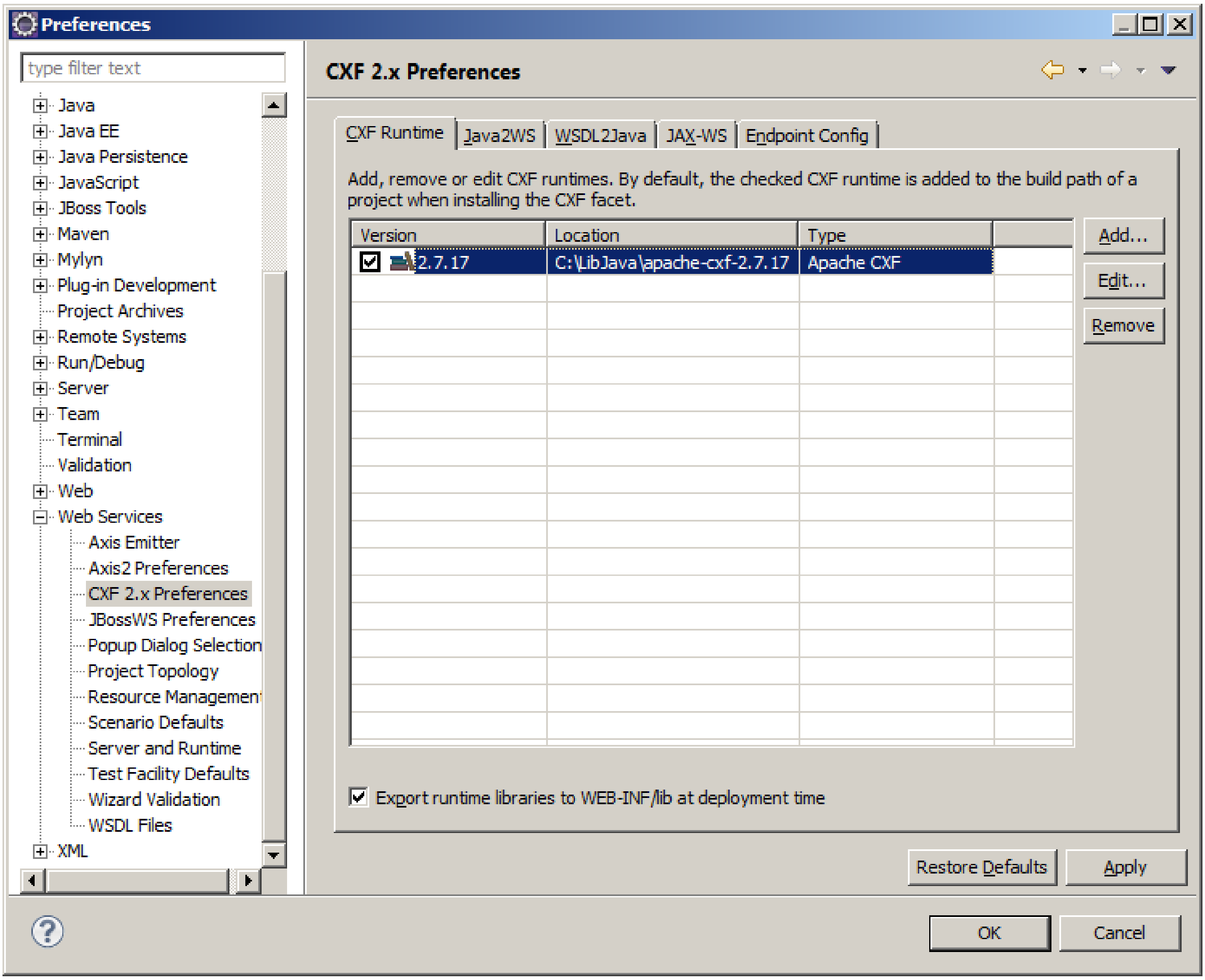
WS-Security consumindo EJB "DBS"

CRIANDO O WEBSERVICE SOBRE CXF

Download

http://mirror.nbtelecom.com.br/apache/cxf/2.7.17/apache-cxf-2.7.17.tar.gz

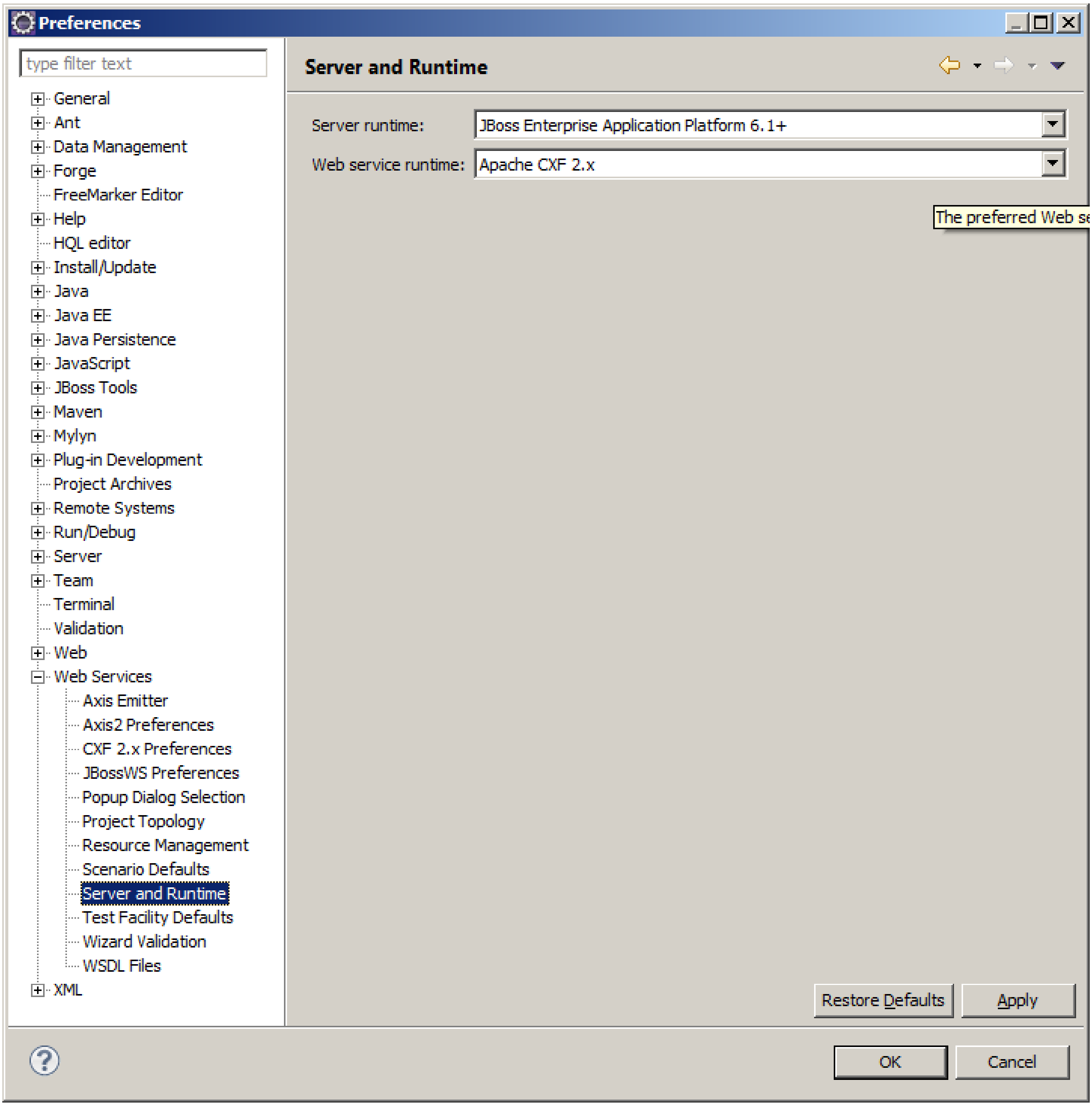


CXF 2.x Preferences

Add, remove or edit CXF runtimes. By default, the checked CXF runtime is added to the build path of a project when installing the CXF facet.

Version	Location	Type
☑ 2.7.17	C:\LibJava\apache-cxf-2.7.17	Apache CXF

☑ Export runtime libraries to WEB-INF/lib at deployment time

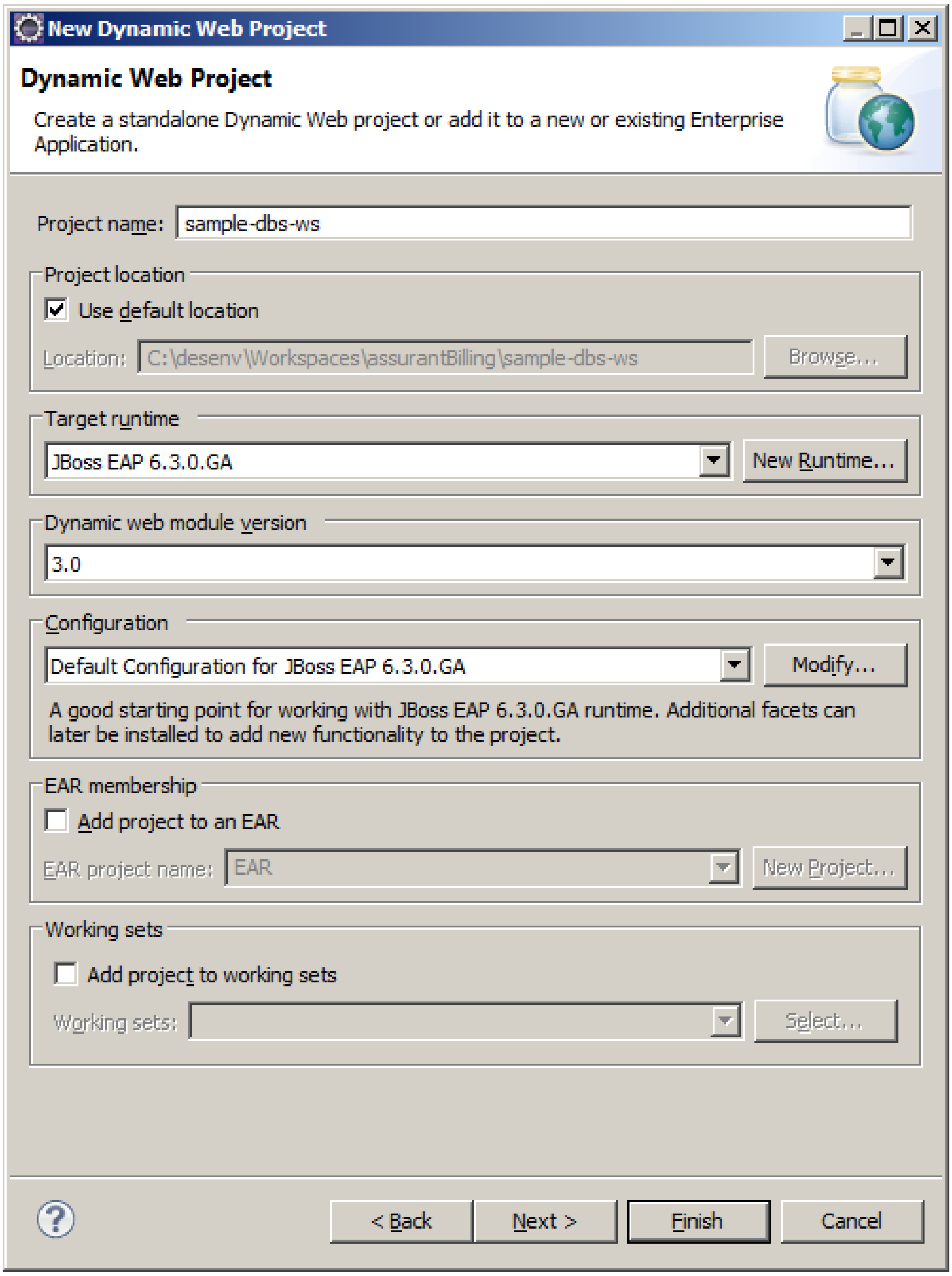


Server and Runtime

Server runtime: JBoss Enterprise Application Platform 6.1+

Web service runtime: Apache CXF 2.x

The preferred Web se



Dynamic Web Project

Create a standalone Dynamic Web project or add it to a new or existing Enterprise Application.

Project name: sample-dbs-ws

Project location

☑ Use default location

Location: C:\desenv\Workspaces\assurantBilling\sample-dbs-ws

Target runtime

JBoss EAP 6.3.0.GA

Dynamic web module version

3.0

Configuration

Default Configuration for JBoss EAP 6.3.0.GA

A good starting point for working with JBoss EAP 6.3.0.GA runtime. Additional facets can later be installed to add new functionality to the project.

EAR membership

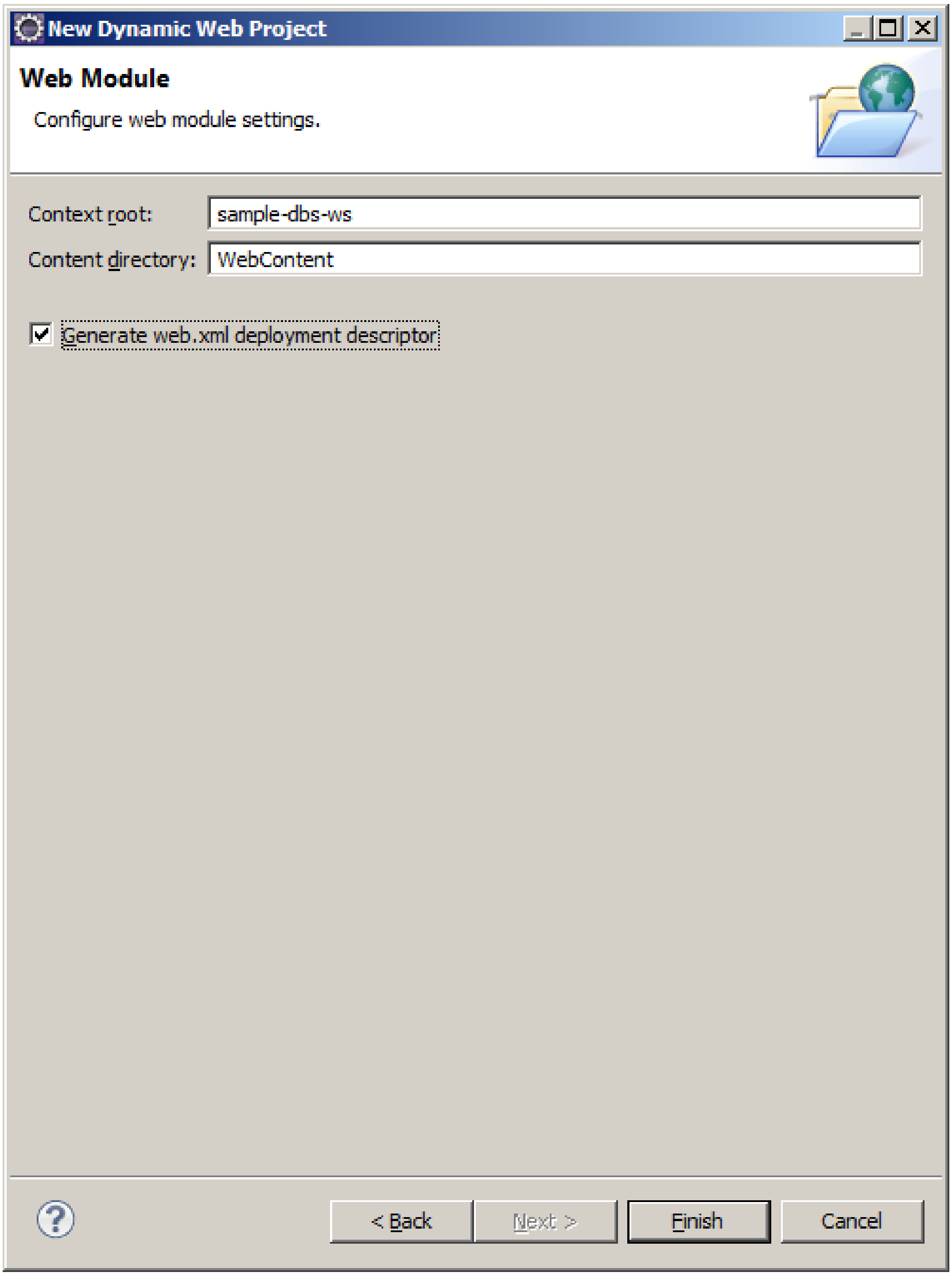
□ Add project to an EAR

EAR project name: EAR

Working sets

□ Add project to working sets

Working sets:



New Dynamic Web Project

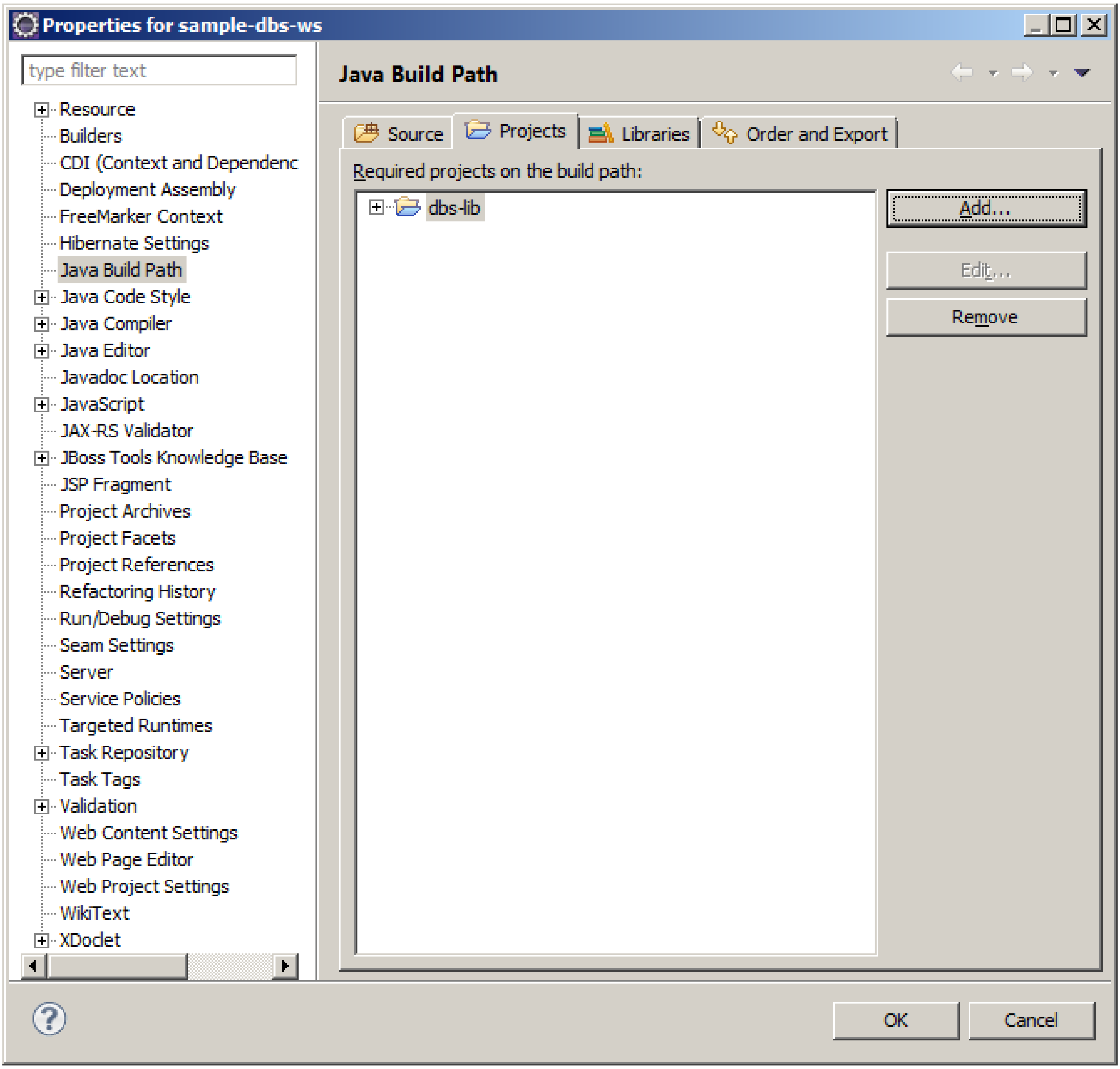
Web Module

Configure web module settings.

Context root: sample-dbs-ws

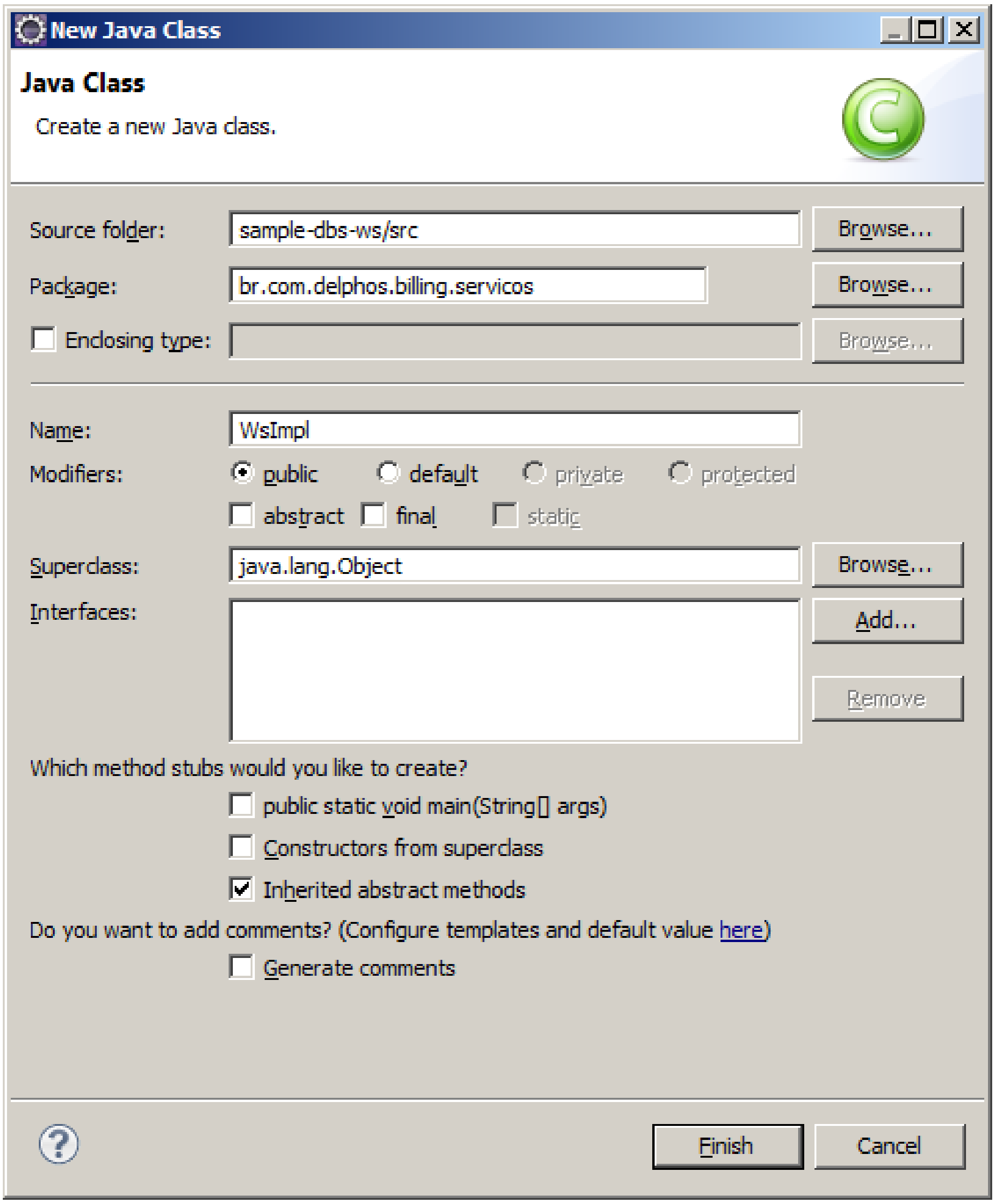
Content directory: WebContent

☑ Generate web.xml deployment descriptor



Required projects on the build path:

dbs-lib



Java Class

Create a new Java class.

Source folder: sample-dbs-ws/src

Package: br.com.delphos.billing.servicos

□ Enclosing type:

Name: WsImpl

Modifiers: ● public ○ default ○ private ○ protected

□ abstract □ final □ static

Superclass: java.lang.Object

Interfaces:

Which method stubs would you like to create?

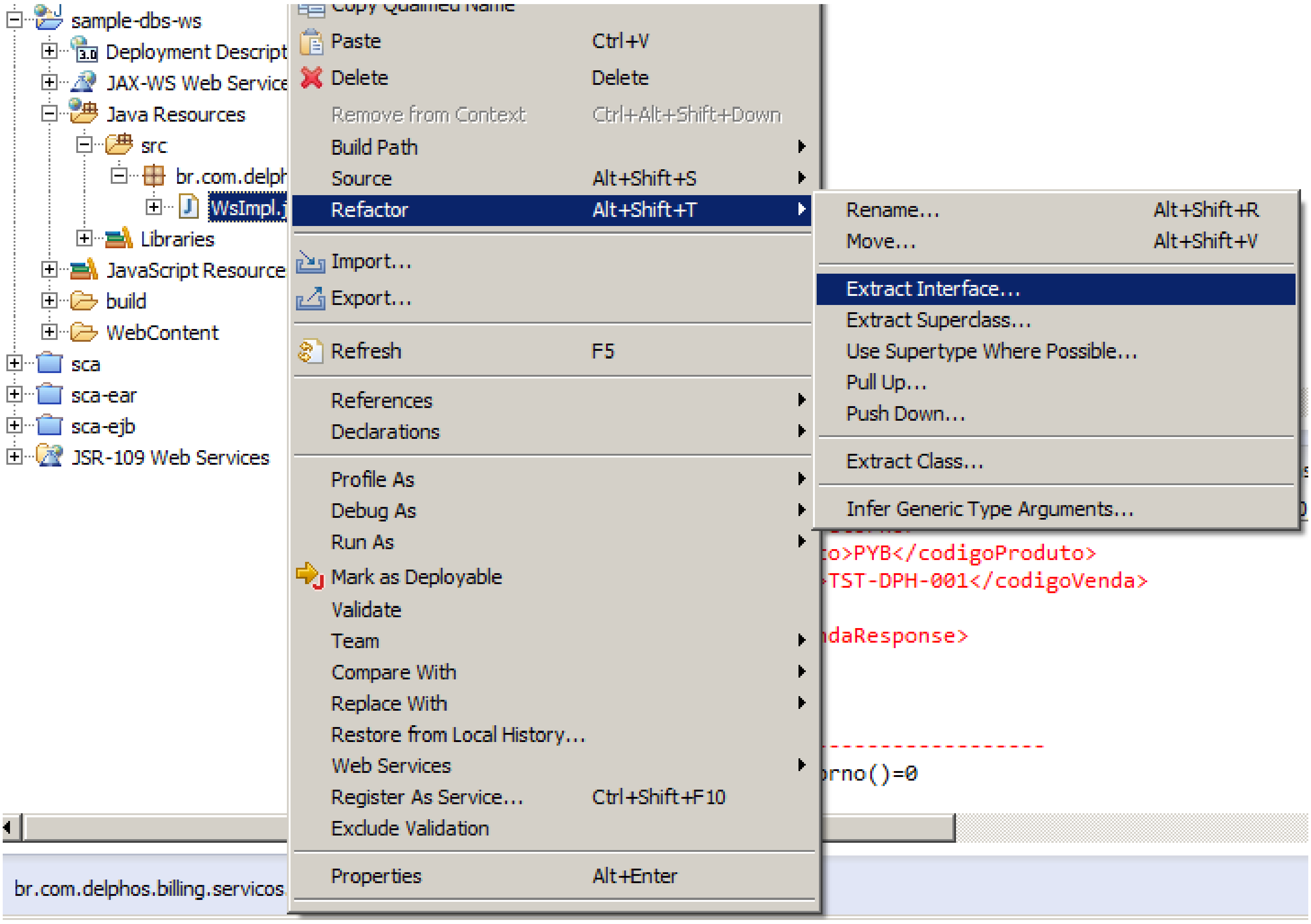
□ public static void main(String[] args)

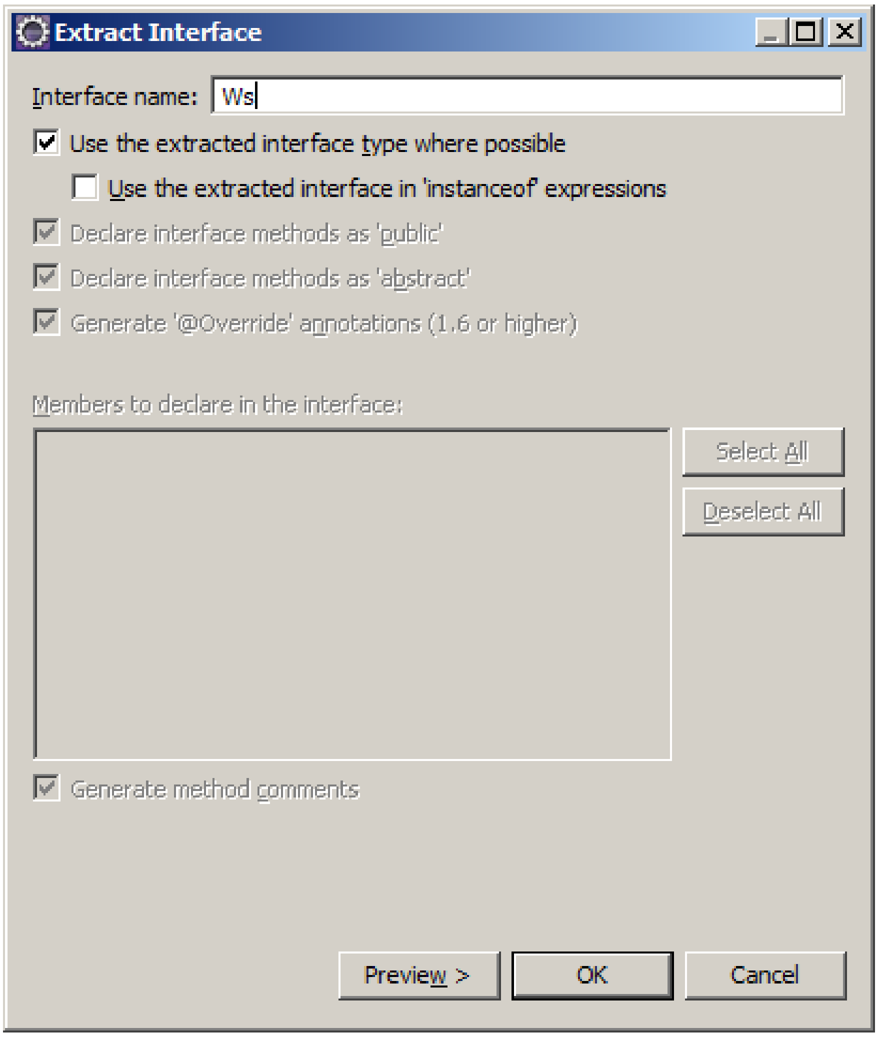
□ Constructors from superclass

☑ Inherited abstract methods

Do you want to add comments? (Configure templates and default value here)

□ Generate comments


sample-dbs-ws
Deployment Descript
JAX-WS Web Service
Java Resources
src
br.com.delph
WsImpl.j
Libraries
JavaScript Resource
build
WebContent
sca
sca-ear
sca-ejb
JSR-109 Web Services
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Remove from Context Ctrl+Alt+Shift+Down
Build Path
Source Alt+Shift+S
Refactor Alt+Shift+T
Import...
Export...
Refresh F5
References
Declarations
Profile As
Debug As
Run As
Mark as Deployable
Validate
Team
Compare With
Replace With
Restore from Local History...
Web Services
Register As Service... Ctrl+Shift+F10
Exclude Validation
Properties Alt+Enter
Rename... Alt+Shift+R
Move... Alt+Shift+V
Extract Interface...
Extract Superclass...
Use Supertype Where Possible...
Pull Up...
Push Down...
Extract Class...
Infer Generic Type Arguments...
o>PYB</codigoProduto>
TST-DPH-001</codigoVenda>
daResponse>
rno()=0
br.com.delphos.billing.servicos



Extract Interface

Interface name: Ws

☑ Use the extracted interface type where possible

☐ Use the extracted interface in 'instanceof' expressions

☑ Declare interface methods as 'public'

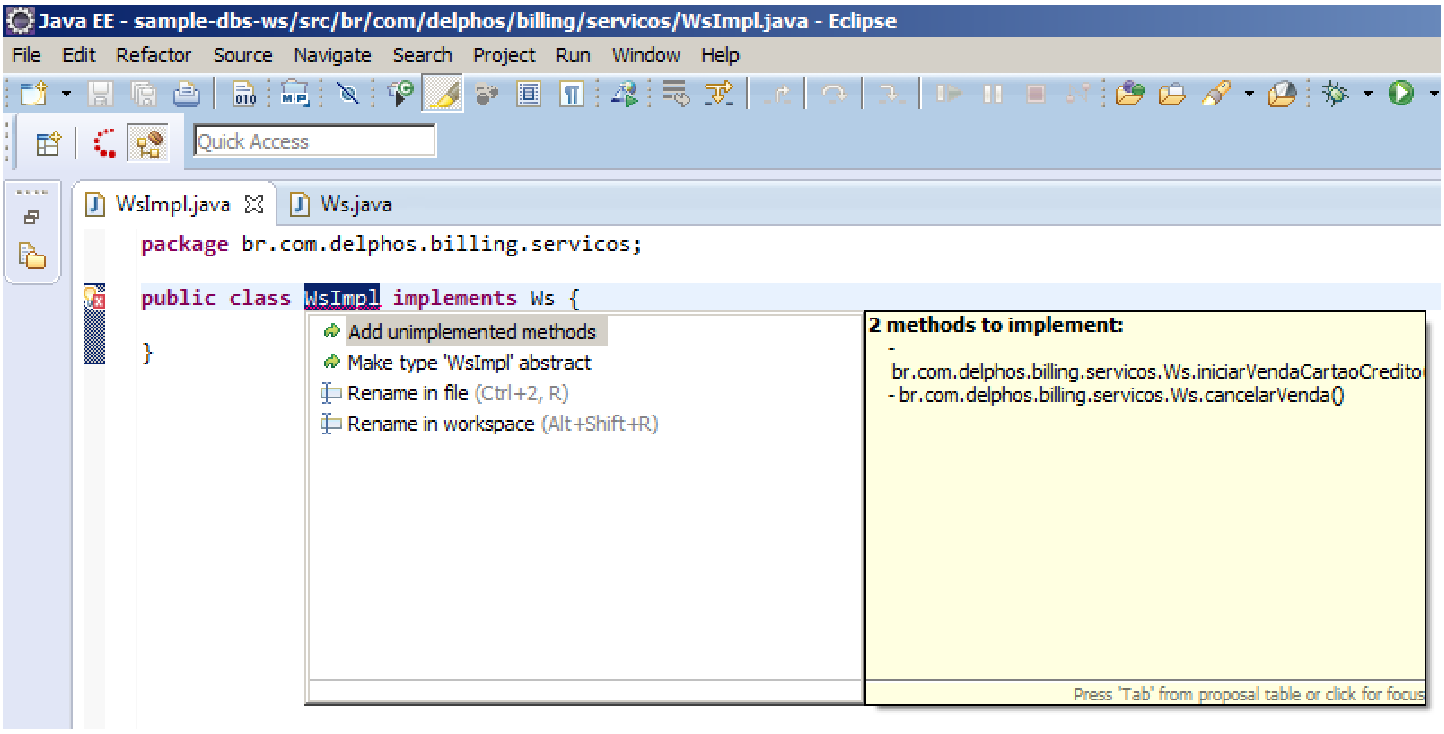
☑ Declare interface methods as 'abstract'

☑ Generate '@Override' annotations (1.6 or higher)

Members to declare in the interface:

☑ Generate method comments

```
package br.com.delphos.billing.servicos;

import br.com.delphos.billing.excecoes.BillingException;
import br.com.delphos.billing.servicos.retornos.RetornoCancelarVenda;
import br.com.delphos.billing.servicos.retornos.RetornoIniciarVendaCartaoCredito;

public interface Ws {

public RetornoIniciarVendaCartaoCredito iniciarVendaCartaoCredito(
String token, String codigoEmpresa, String codigoProduto,
String codigoSistema, String cpf, String nome, String email,
String dddCelular, String telefoneCelular, String dddContato,
String telefoneContato, String codigoVenda, String dataVenda,
String bandeiraCartao, String cpfPortadorCartao,
String nomePortadorCartao, String numeroCartao,
String vencimentoCartao, String cvvCartao, String tipoCobranca,
String quantidadeParcelas, String valorCobranca,
String dataPrimeiraCobranca, String dataFimVigencia)
throws BillingException;

public RetornoCancelarVenda cancelarVenda(String codigoEmpresa,
String codigoProduto, String codigoSistema, String token,
String cpf, String codigoVenda,
String codigoMotivoCancelamento, String valorEstorno,
String flagCancelamentoGateway) throws BillingException;

}
```


Java EE - sample-dbs-ws/src/br/com/delphos/billing/servicos/WsImpl.java - Eclipse
File Edit Refactor Source Navigate Search Project Run Window Help
Quick Access
WsImpl.java
Ws.java

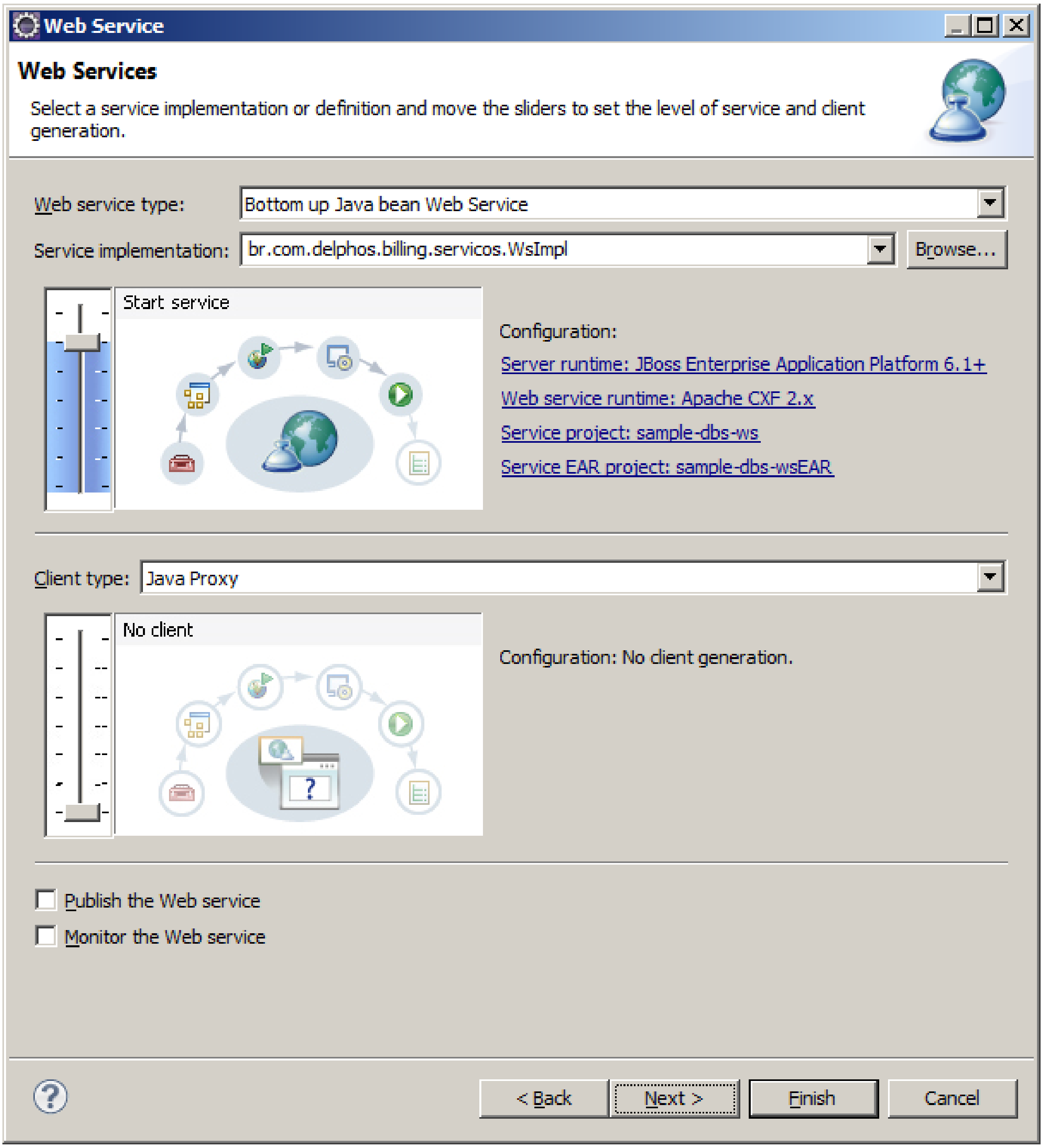
```
package br.com.delphos.billing.servicos;

public class WsImpl implements Ws {

}
```

Add unimplemented methods
Make type 'WsImpl' abstract
Rename in file (Ctrl+2, R)
Rename in workspace (Alt+Shift+R)

2 methods to implement:
- br.com.delphos.billing.servicos.Ws.iniciarVendaCartaoCredito
- br.com.delphos.billing.servicos.Ws.cancelarVenda()

Press 'Tab' from proposal table or click for focus

```
package br.com.delphos.billing.servicos;

import javax.naming.InitialContext;
import javax.naming.NamingException;

import br.com.delphos.billing.excecoes.BillingException;
import br.com.delphos.billing.servicos.retornos.RetornoCancelarVenda;
import br.com.delphos.billing.servicos.retornos.RetornoIniciarVendaCartaoCredito;

public class WsImpl implements Ws {

private BillingService getService() {

try {
InitialContext ic = new InitialContext();
return (BillingService)ic.lookup("java:global/dbs/BillingServiceBean");
} catch (NamingException e) {
// TODO Auto-generated catch block
e.printStackTrace();
}
return null;

}

@Override
public RetornoIniciarVendaCartaoCredito iniciarVendaCartaoCredito(
String token, String codigoEmpresa, String codigoProduto,
String codigoSistema, String cpf, String nome, String email,
String dddCelular, String telefoneCelular, String dddContato,
String telefoneContato, String codigoVenda, String dataVenda,
String bandeiraCartao, String cpfPortadorCartao,
String nomePortadorCartao, String numeroCartao,
String vencimentoCartao, String cvvCartao, String tipoCobranca,
String quantidadeParcelas, String valorCobranca,
String dataPrimeiraCobranca, String dataFimVigencia)
throws BillingException {
// TODO Auto-generated method stub
return getService().iniciarVendaCartaoCredito(token, codigoEmpresa, codigoProduto,
codigoSistema, cpf, nome, email, dddCelular, telefoneCelular, dddContato,
telefoneContato, codigoVenda, dataVenda, bandeiraCartao, cpfPortadorCartao,
nomePortadorCartao, numeroCartao, vencimentoCartao, cvvCartao, tipoCobranca,
quantidadeParcelas, valorCobranca, dataPrimeiraCobranca, dataFimVigencia);
}

@Override
public RetornoCancelarVenda cancelarVenda(String codigoEmpresa,
String codigoProduto, String codigoSistema, String token,
String cpf, String codigoVenda, String codigoMotivoCancelamento,
String valorEstorno, String flagCancelamentoGateway)
throws BillingException {
// TODO Auto-generated method stub
return getService().cancelarVenda(codigoEmpresa, codigoProduto, codigoSistema,
token, cpf, codigoVenda, codigoMotivoCancelamento, valorEstorno, flagCancelamentoGateway);
}

}
```



Web Service

Web Services

Select a service implementation or definition and move the sliders to set the level of service and client generation.

Web service type: Bottom up Java bean Web Service

Service implementation: br.com.delphos.billing.servicos.WsImpl

Start service

Configuration:

Server runtime: JBoss Enterprise Application Platform 6.1+

Web service runtime: Apache CXF 2.x

Service project: sample-dbs-ws

Service EAR project: sample-dbs-wsEAR

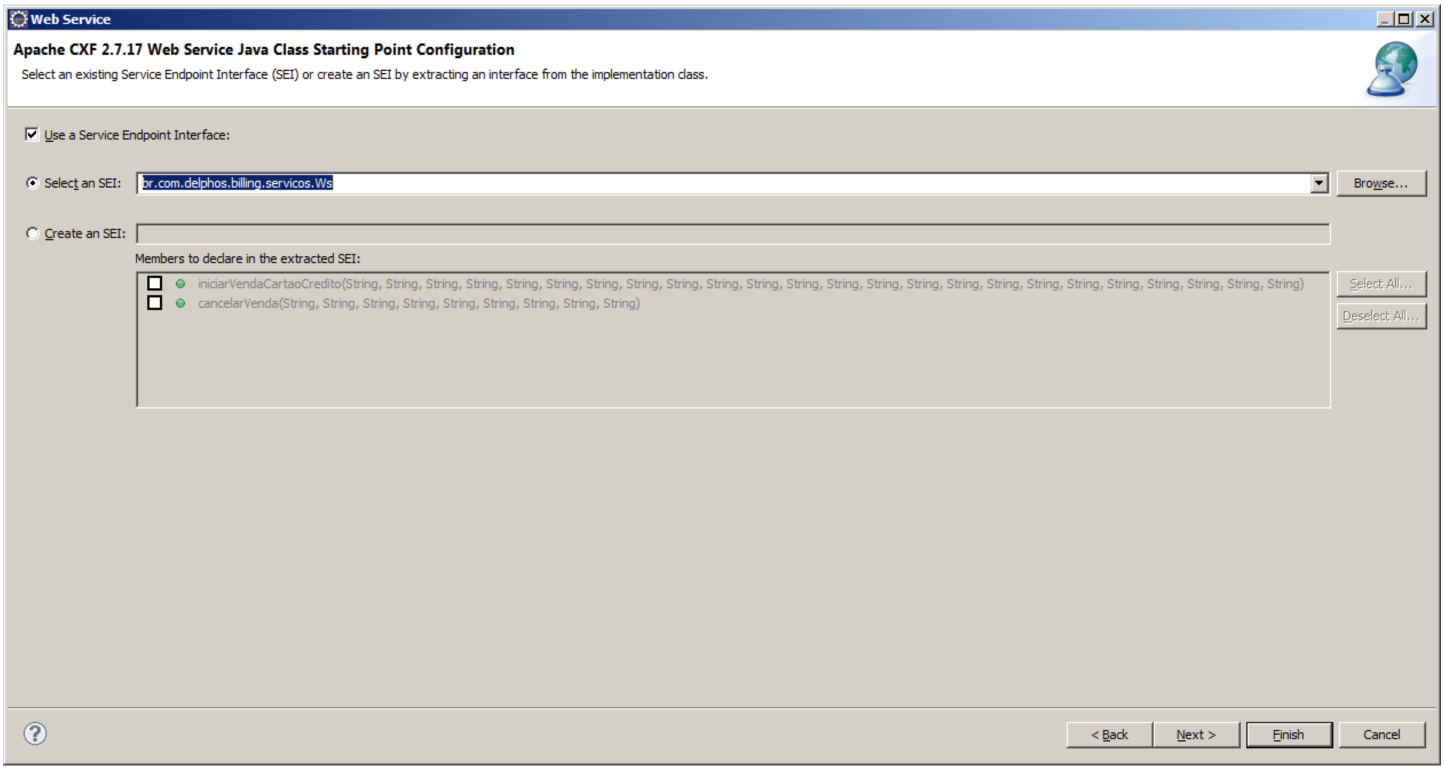
Client type: Java Proxy

No client

Configuration: No client generation.

□ Publish the Web service

□ Monitor the Web service



Apache CXF 2.7.17 Web Service Java Class Starting Point Configuration

Select an existing Service Endpoint Interface (SEI) or create an SEI by extracting an interface from the implementation class.

☑ Use a Service Endpoint Interface:

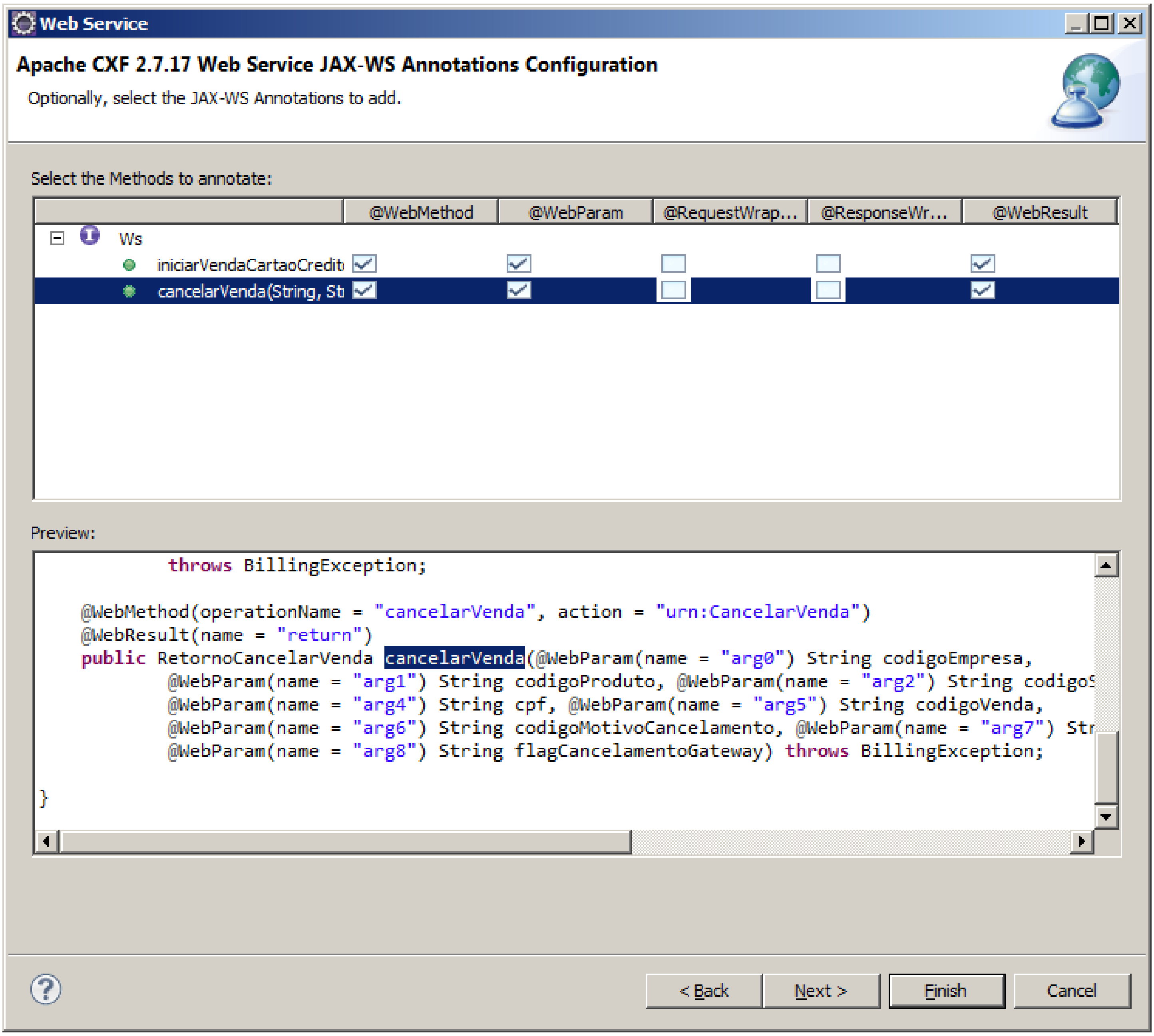
● Select an SEI: br.com.delphos.billing.servicos.Ws

○ Create an SEI:

Members to declare in the extracted SEI:

□ iniciarVendaCartaoCredito(String, String)

□ cancelarVenda(String, String, String, String, String, String, String, String, String, String)



Apache CXF 2.7.17 Web Service JAX-WS Annotations Configuration

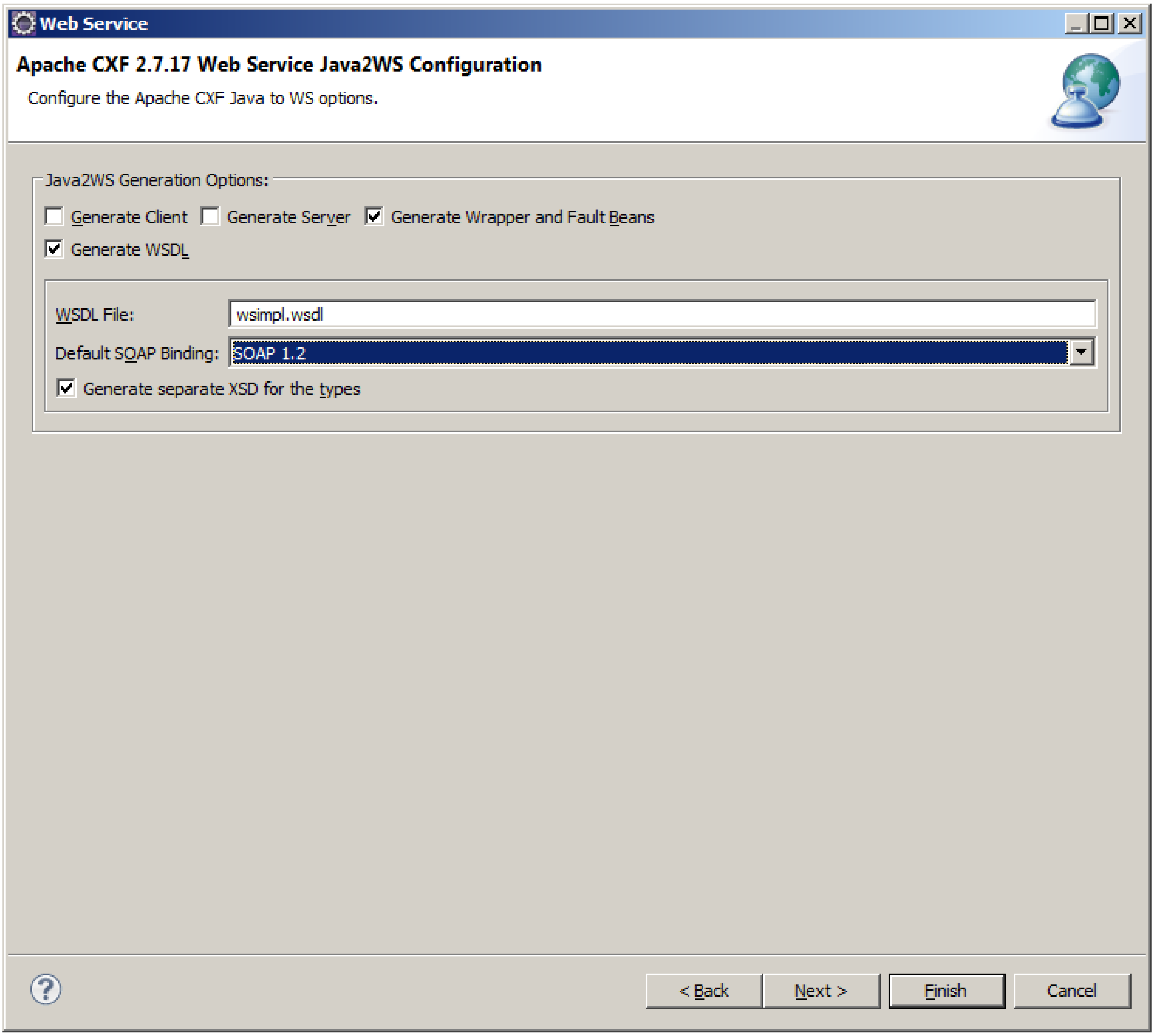
Optionally, select the JAX-WS Annotations to add.

Select the Methods to annotate:

	@WebMethod	@WebParam	@RequestWrap…	@ResponseWr…	@WebResult
Ws					
iniciarVendaCartaoCredit	☑	☑	□	□	☑
cancelarVenda(String, St	☑	☑	□	□	☑

Preview:

```
            throws BillingException;

    @WebMethod(operationName = "cancelarVenda", action = "urn:CancelarVenda")
    @WebResult(name = "return")
    public RetornoCancelarVenda cancelarVenda(@WebParam(name = "arg0") String codigoEmpresa,
            @WebParam(name = "arg1") String codigoProduto, @WebParam(name = "arg2") String codigoS
            @WebParam(name = "arg4") String cpf, @WebParam(name = "arg5") String codigoVenda,
            @WebParam(name = "arg6") String codigoMotivoCancelamento, @WebParam(name = "arg7") Str
            @WebParam(name = "arg8") String flagCancelamentoGateway) throws BillingException;

}
```



Web Service

Apache CXF 2.7.17 Web Service Java2WS Configuration

Configure the Apache CXF Java to WS options.

Java2WS Generation Options:

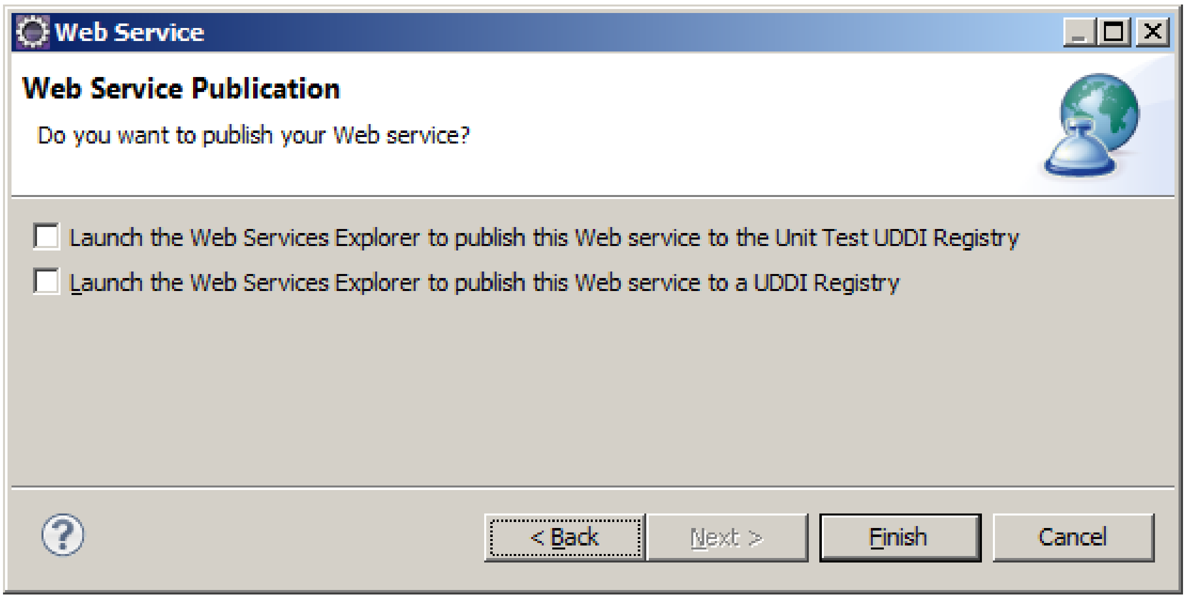
□ Generate Client □ Generate Server ☑ Generate Wrapper and Fault Beans

☑ Generate WSDL

WSDL File: wsimpl.wsdl

Default SOAP Binding: SOAP 1.2

☑ Generate separate XSD for the types



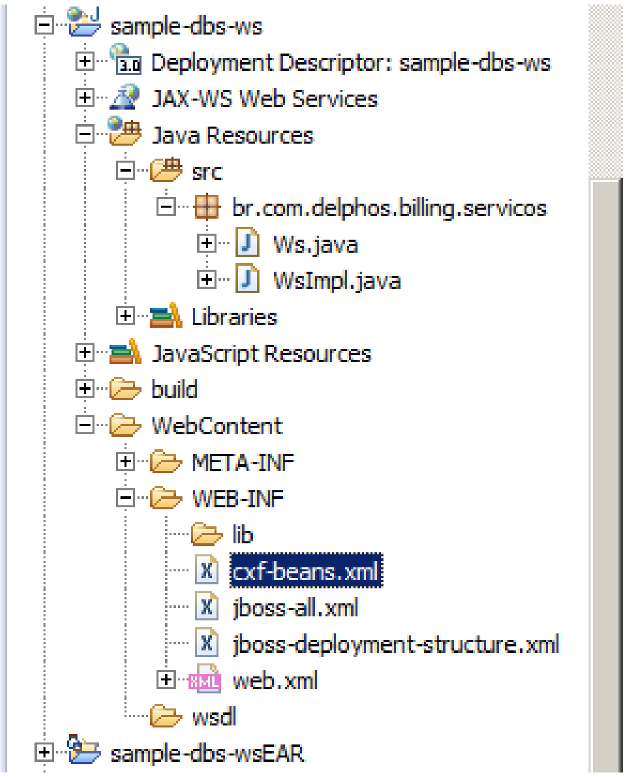
Web Service

Web Service Publication

Do you want to publish your Web service?

□ Launch the Web Services Explorer to publish this Web service to the Unit Test UDDI Registry

□ Launch the Web Services Explorer to publish this Web service to a UDDI Registry



jboss-all.xml

```
<jboss umlns="urn:jboss:1.0">

<!-- É necessário especificar uma dependência à implantação do DBSDB. -->
<!-- Esta configuração indica ao JBoss EAP que, para implantar esta -->
<!-- aplicação, é necessário que o DBSDB já esteja implantado. Isto se -->
<!-- torna necessário para que o JBoss EAP seja capaz de ordenar as -->
<!-- implantaçõpes corretamente, de forma que ele seja capaz de detectar -->
<!-- as dependências entre EJBs e injetá-las. Se isto não for feito, a -->
<!-- aplicação dependente poderá ser implantada, mas não irá ser -->
<!-- iniciada. -->
<jboss-deployment-dependencies xmlns="urn:jboss:deployment-dependencies:1.0">
<dependency name="dbsdb.jar" />
<dependency name="dbs.jar" />
</jboss-deployment-dependencies>

</jboss>
```

jboss-deployment-structure.xml

```
<!-- xmlns="urn:jboss:deployment-structure:1.2" -->
<jboss-deployment-structure>
<deployment>
<dependencies>

<module name="deployment.dbs-lib.jar" />
```

```
</dependencies>

<exclude-subsystems>
<subsystem name="webservices" />
</exclude-subsystems>

</deployment>
</jboss-deployment-structure>
```

cxf-beans.xml (Remova a linha em vermelho)

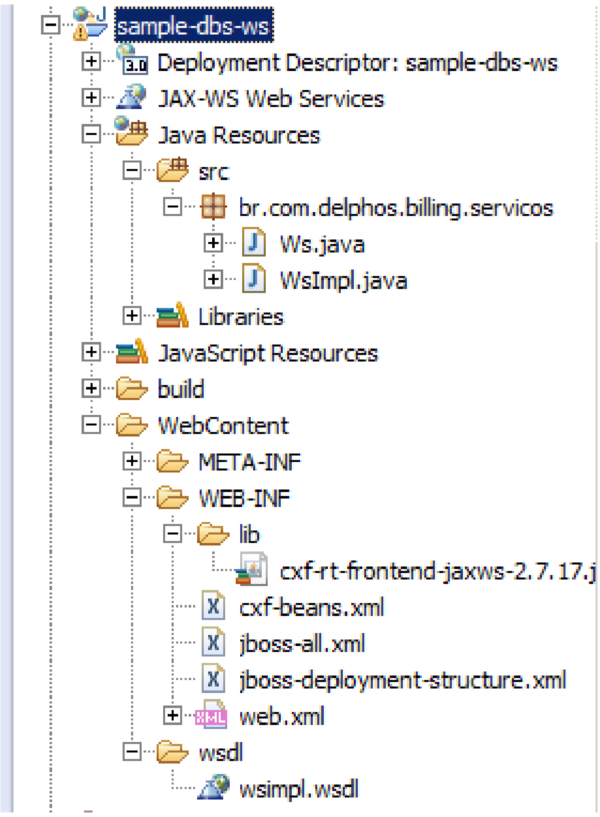
```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:jaxws="http://cxf.apache.org/jaxws"
xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-2.5.xsd http://cxf.apache.org/jaxws
http://cxf.apache.org/schemas/jaxws.xsd">
<import resource="classpath:META-INF/cxf/cxf.xml" />
<import resource="classpath:META-INF/cxf/cxf-extension-soap.xml" />
<import resource="classpath:META-INF/cxf/cxf-servlet.xml" />
<jaxws:endpoint id="ws"
implementor="br.com.delphos.billing.servicos.WsImpl"
wsdlLocation="wsdl/wsimpl.wsdl"
address="/ws">
<jaxws:features>
<bean class="org.apache.cxf.feature.LoggingFeature" />
</jaxws:features>
</jaxws:endpoint>
</beans>
```

FAÇA DEPLOY NO SERVIDOR E ACESSE

http://localhost:8080/sample-dbs-ws/services/ws?wsdl

SALVE O CÓDIGO FONTE EM /dbs-ws/WebContent/wsdl/wsimpl.wsdl

COPIE C:\LibJava\apache-cxf-2.7.17\modules\cxf-rt-frontend-jaxws-2.7.17.jar PARA /dbs-ws/WebContent/WEB-INF/lib/cxf-rt-frontend-jaxws-2.7.17.jar

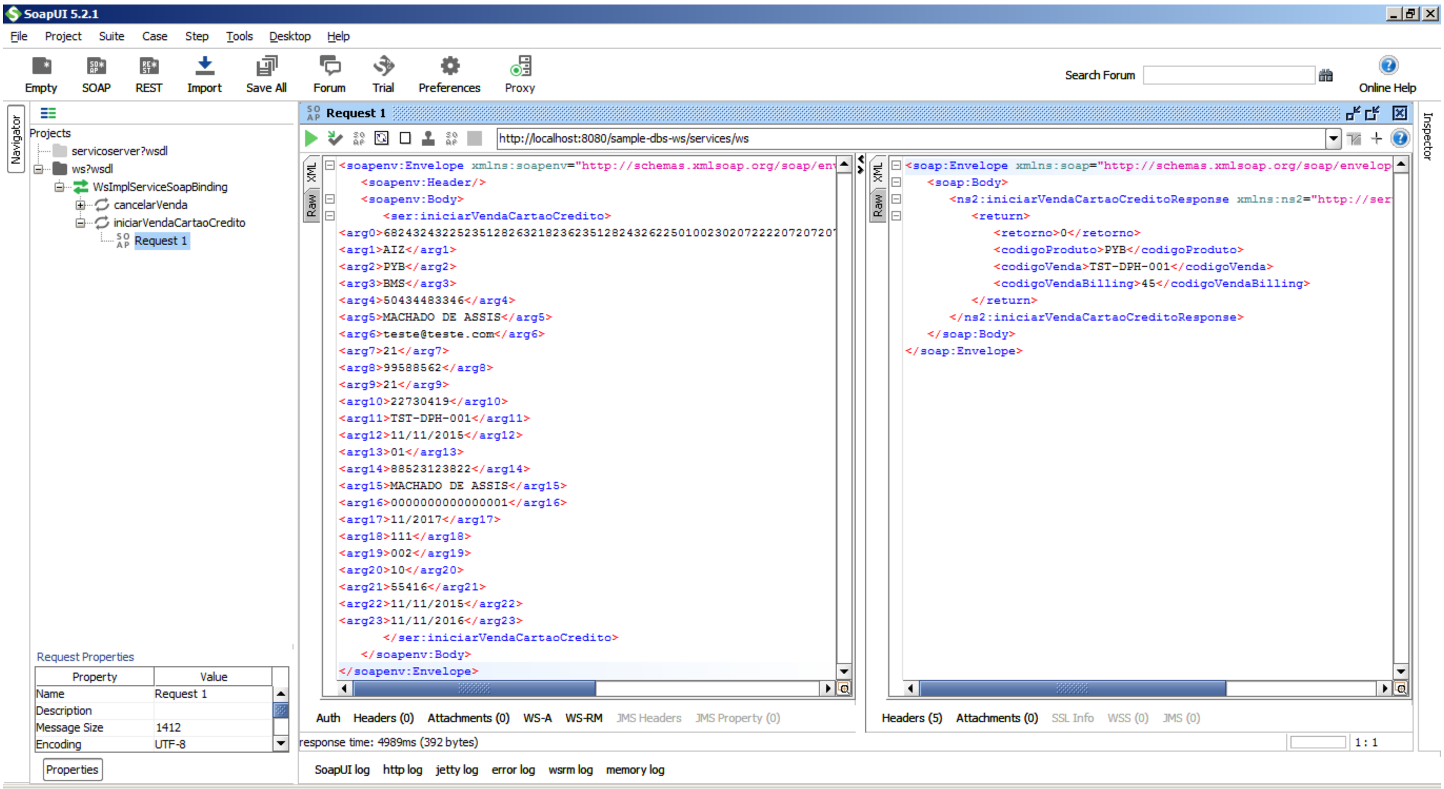


```
sample-dbs-ws
  Deployment Descriptor: sample-dbs-ws
  JAX-WS Web Services
  Java Resources
    src
      br.com.delphos.billing.servicos
        Ws.java
        WsImpl.java
    Libraries
  JavaScript Resources
  build
  WebContent
    META-INF
    WEB-INF
      lib
        cxf-rt-frontend-jaxws-2.7.17.j
      cxf-beans.xml
      jboss-all.xml
      jboss-deployment-structure.xml
      web.xml
    wsdl
      wsimpl.wsdl
```

cxf-beans.xml (Adicione a linha em vermelho)

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:jaxws="http://cxf.apache.org/jaxws"
xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-2.5.xsd http://cxf.apache.org/jaxws
http://cxf.apache.org/schemas/jaxws.xsd">
<import resource="classpath:META-INF/cxf/cxf.xml" />
<import resource="classpath:META-INF/cxf/cxf-extension-soap.xml" />
<import resource="classpath:META-INF/cxf/cxf-servlet.xml" />
<jaxws:endpoint id="ws"
implementor="br.com.delphos.billing.servicos.WsImpl"
wsdlLocation="wsdl/wsimpl.wsdl"
address="/ws">
<jaxws:features>
<bean class="org.apache.cxf.feature.LoggingFeature" />
</jaxws:features>
</jaxws:endpoint>
</beans>
```

RE-DEPLOY DA APLICAÇÃO


SoapUI 5.2.1
File Project Suite Case Step Tools Desktop Help
Empty SOAP REST Import Save All Forum Trial Preferences Proxy
Search Forum
Online Help
Navigator
Projects
servicoserver?wsdl
ws?wsdl
WsImplServiceSoapBinding
cancelarVenda
iniciarVendaCartaoCredito
Request 1
Request 1
XML
Raw
Inspector
Auth Headers (0) Attachments (0) WS-A WS-RM JMS Headers JMS Property (0)
Headers (5) Attachments (0) SSL Info WSS (0) JMS (0)
SoapUI log http log jetty log error log wsrm log memory log
Properties
1 : 1

http://localhost:8080/sample-dbs-ws/services/ws

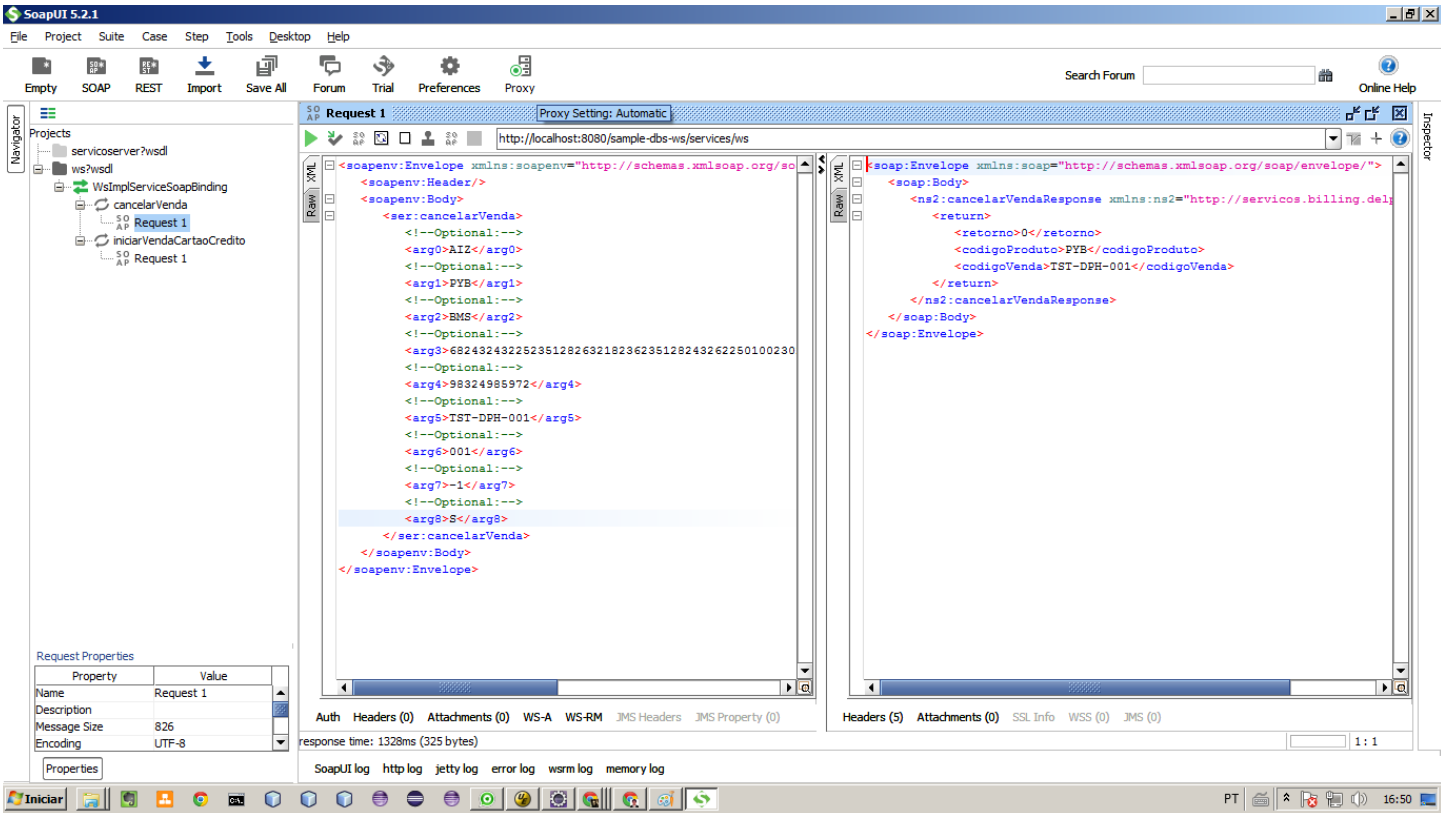
```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/env
   <soapenv:Header/>
   <soapenv:Body>
      <ser:iniciarVendaCartaoCredito>
<arg0>6824324322523512826321823623512824326225010023020722220720720
<arg1>AIZ</arg1>
<arg2>PYB</arg2>
<arg3>BMS</arg3>
<arg4>50434483346</arg4>
<arg5>MACHADO DE ASSIS</arg5>
<arg6>teste@teste.com</arg6>
<arg7>21</arg7>
<arg8>99588562</arg8>
<arg9>21</arg9>
<arg10>22730419</arg10>
<arg11>TST-DPH-001</arg11>
<arg12>11/11/2015</arg12>
<arg13>01</arg13>
<arg14>88523123822</arg14>
<arg15>MACHADO DE ASSIS</arg15>
<arg16>0000000000000001</arg16>
<arg17>11/2017</arg17>
<arg18>111</arg18>
<arg19>002</arg19>
<arg20>10</arg20>
<arg21>55416</arg21>
<arg22>11/11/2015</arg22>
<arg23>11/11/2016</arg23>
      </ser:iniciarVendaCartaoCredito>
   </soapenv:Body>
</soapenv:Envelope>
```

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelop
   <soap:Body>
      <ns2:iniciarVendaCartaoCreditoResponse xmlns:ns2="http://ser
         <return>
            <retorno>0</retorno>
            <codigoProduto>PYB</codigoProduto>
            <codigoVenda>TST-DPH-001</codigoVenda>
            <codigoVendaBilling>45</codigoVendaBilling>
         </return>
      </ns2:iniciarVendaCartaoCreditoResponse>
   </soap:Body>
</soap:Envelope>
```

Request Properties

Property	Value
Name	Request 1
Description	
Message Size	1412
Encoding	UTF-8

response time: 4989ms (392 bytes)


SoapUI 5.2.1
File Project Suite Case Step Tools Desktop Help
Empty SOAP REST Import Save All Forum Trial Preferences Proxy
Search Forum
Online Help
Navigator
Projects
servicoserver?wsdl
ws?wsdl
WsImplServiceSoapBinding
cancelarVenda
Request 1
iniciarVendaCartaoCredito
Request 1
Request 1
Proxy Setting: Automatic
XML
Raw
Inspector
Auth Headers (0) Attachments (0) WS-A WS-RM JMS Headers JMS Property (0)
Headers (5) Attachments (0) SSL Info WSS (0) JMS (0)
SoapUI log http log jetty log error log wsrm log memory log
Properties
1 : 1
Iniciar
PT
16:50

http://localhost:8080/sample-dbs-ws/services/ws

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/so
   <soapenv:Header/>
   <soapenv:Body>
      <ser:cancelarVenda>
         <!--Optional:-->
         <arg0>AIZ</arg0>
         <!--Optional:-->
         <arg1>PYB</arg1>
         <!--Optional:-->
         <arg2>BMS</arg2>
         <!--Optional:-->
         <arg3>682432432252351282632182362351282432622501002300
         <!--Optional:-->
         <arg4>98324985972</arg4>
         <!--Optional:-->
         <arg5>TST-DPH-001</arg5>
         <!--Optional:-->
         <arg6>001</arg6>
         <!--Optional:-->
         <arg7>-1</arg7>
         <!--Optional:-->
         <arg8>S</arg8>
      </ser:cancelarVenda>
   </soapenv:Body>
</soapenv:Envelope>
```

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <soap:Body>
      <ns2:cancelarVendaResponse xmlns:ns2="http://servicos.billing.delp
         <return>
            <retorno>0</retorno>
            <codigoProduto>PYB</codigoProduto>
            <codigoVenda>TST-DPH-001</codigoVenda>
         </return>
      </ns2:cancelarVendaResponse>
   </soap:Body>
</soap:Envelope>
```

Request Properties

Property	Value
Name	Request 1
Description	
Message Size	826
Encoding	UTF-8

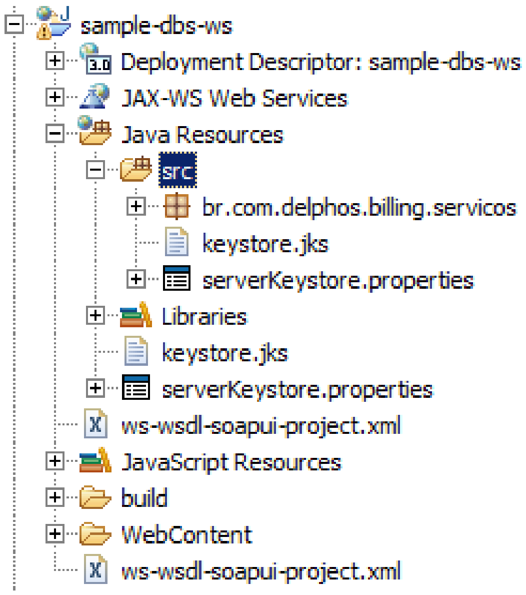
response time: 1328ms (325 bytes)

APLICANDO CERTIFICADO

```
keytool.exe -genkey -alias testkey -keystore keystore.jks -storepass storepass -dname "cn=testkey" -
keyalg RSA
keytool.exe -export -alias testkey -file publickey.rsa -keystore keystore.jks -storepass storepass
keytool.exe -import -alias testkey -file publickey.rsa -keystore clientkeystore.jks -storepass clientstorepass
```

serverKeystore.properties

```
org.apache.ws.security.crypto.provider=org.apache.ws.security.components.crypto.Merlin
org.apache.ws.security.crypto.merlin.keystore.file=keystore.jks
org.apache.ws.security.crypto.merlin.keystore.password=storepass
org.apache.ws.security.crypto.merlin.keystore.type=jks
```



```
⊟ sample-dbs-ws
    ⊞ Deployment Descriptor: sample-dbs-ws
    ⊞ JAX-WS Web Services
    ⊟ Java Resources
        ⊟ src
            ⊞ br.com.delphos.billing.servicos
              keystore.jks
            ⊞ serverKeystore.properties
        ⊞ Libraries
          keystore.jks
        ⊞ serverKeystore.properties
      ws-wsdl-soapui-project.xml
    ⊞ JavaScript Resources
    ⊞ build
    ⊞ WebContent
      ws-wsdl-soapui-project.xml
```

br.com.delphos.billing.security.ServerPasswordCallback

```
package br.com.delphos.billing.security;

import java.io.IOException;

import javax.security.auth.callback.Callback;
import javax.security.auth.callback.CallbackHandler;
import javax.security.auth.callback.UnsupportedCallbackException;
import org.apache.ws.security.WSPasswordCallback;

public class ServerPasswordCallback implements CallbackHandler {

@Override
public void handle(Callback[] arg0) throws IOException,
UnsupportedCallbackException {
```

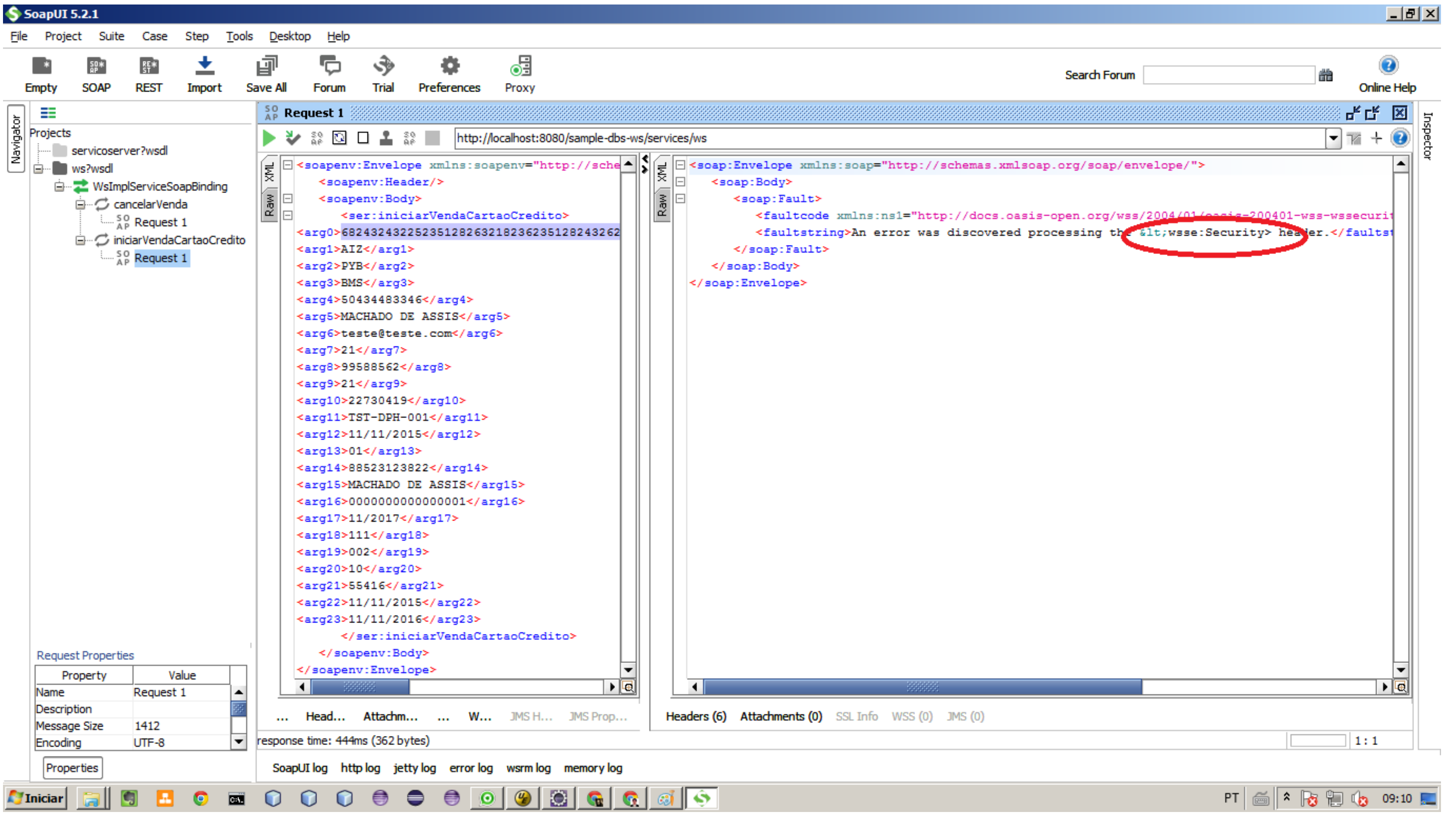
```
WSPasswordCallback pc = (WSPasswordCallback) arg0[0];

// set the password for our message.
pc.setPassword("storepass");

}

}
```

<u>cxf-beans.xml (ADICIONE AS LINHAS VERMELHAS)</u>

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:jaxws="http://cxf.apache.org/jaxws"
xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-2.5.xsd http://cxf.apache.org/jaxws
http://cxf.apache.org/schemas/jaxws.xsd">
<import resource="classpath:META-INF/cxf/cxf.xml" />
<import resource="classpath:META-INF/cxf/cxf-extension-soap.xml" />
<import resource="classpath:META-INF/cxf/cxf-servlet.xml" />
<!-- ************************************************** -->
<bean id="myPasswordCallback" class="br.com.delphos.billing.security.ServerPasswordCallback"/>
(classe criada anteriormente)
<!-- ************************************************** -->
<jaxws:endpoint id="ws"
implementor="br.com.delphos.billing.servicos.WsImpl"
address="/ws"
wsdlLocation="wsdl/wsimpl.wsdl"
xmlns:tns="http://servicos.billing.delphos.com.br/" (referencia encontrada no cabeçalho de
"wsdl/wsimpl.wsdl" )
endpointName="tns:WsImplPort" (referencia encontrada na anotação @WebService de WsImpl.java )
serviceName="tns:WsImplService" (referencia encontrada na anotação @WebService de WsImpl.java )
>
<jaxws:features>
<bean class="org.apache.cxf.feature.LoggingFeature" />
</jaxws:features>
<!-- ************************************************** -->
<jaxws:inInterceptors>
<bean class="org.apache.cxf.ws.security.wss4j.WSS4JInInterceptor">
<constructor-arg>
<map>
<entry key="user" value="testkey"/> (usuario definido na criação do certificado)
<entry key="action" value="Encrypt"/>
<entry key="passwordType" value="PasswordText"/>
<entry key="decryptionParts" value="{Content}{http://schemas.xmlsoap.org/soap/envelope/}Body"/>
<entry key="decryptionPropFile" value="serverKeystore.properties"/> (arquivo properties criado src)
<entry key="passwordCallbackRef">
```

```
<ref bean="myPasswordCallback"/> (Beans definido acima que aponta para a classe
ServerPasswordCallback que criamos anteriormente )
</entry>
</map>
</constructor-arg>
</bean>
</jaxws:inInterceptors>
<!-- ************************************************* -->
</jaxws:endpoint>
</beans>
```

RE-DEPLOY


SoapUI 5.2.1
File Project Suite Case Step Tools Desktop Help
Empty SOAP REST Import Save All Forum Trial Preferences Proxy
Search Forum
Online Help
Navigator
Projects
servicoserver?wsdl
ws?wsdl
WsImplServiceSoapBinding
cancelarVenda
Request 1
iniciarVendaCartaoCredito
Request 1
Request 1
XML
Raw
Inspector
Head... Attachm... ... W... JMS H... JMS Prop...
Headers (6) Attachments (0) SSL Info WSS (0) JMS (0)
SoapUI log http log jetty log error log wsrm log memory log
Properties
Iniciar
PT 09:10

http://localhost:8080/sample-dbs-ws/services/ws

```
<soapenv:Envelope xmlns:soapenv="http://sche
   <soapenv:Header/>
   <soapenv:Body>
      <ser:iniciarVendaCartaoCredito>
<arg0>68243243225235128263218236235128243262
<arg1>AIZ</arg1>
<arg2>PYB</arg2>
<arg3>BMS</arg3>
<arg4>50434483346</arg4>
<arg5>MACHADO DE ASSIS</arg5>
<arg6>teste@teste.com</arg6>
<arg7>21</arg7>
<arg8>99588562</arg8>
<arg9>21</arg9>
<arg10>22730419</arg10>
<arg11>TST-DPH-001</arg11>
<arg12>11/11/2015</arg12>
<arg13>01</arg13>
<arg14>88523123822</arg14>
<arg15>MACHADO DE ASSIS</arg15>
<arg16>0000000000000001</arg16>
<arg17>11/2017</arg17>
<arg18>111</arg18>
<arg19>002</arg19>
<arg20>10</arg20>
<arg21>55416</arg21>
<arg22>11/11/2015</arg22>
<arg23>11/11/2016</arg23>
      </ser:iniciarVendaCartaoCredito>
   </soapenv:Body>
</soapenv:Envelope>
```

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <soap:Body>
      <soap:Fault>
         <faultcode xmlns:ns1="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurit
         <faultstring>An error was discovered processing the &lt;wsse:Security> header.</faultst
      </soap:Fault>
   </soap:Body>
</soap:Envelope>
```

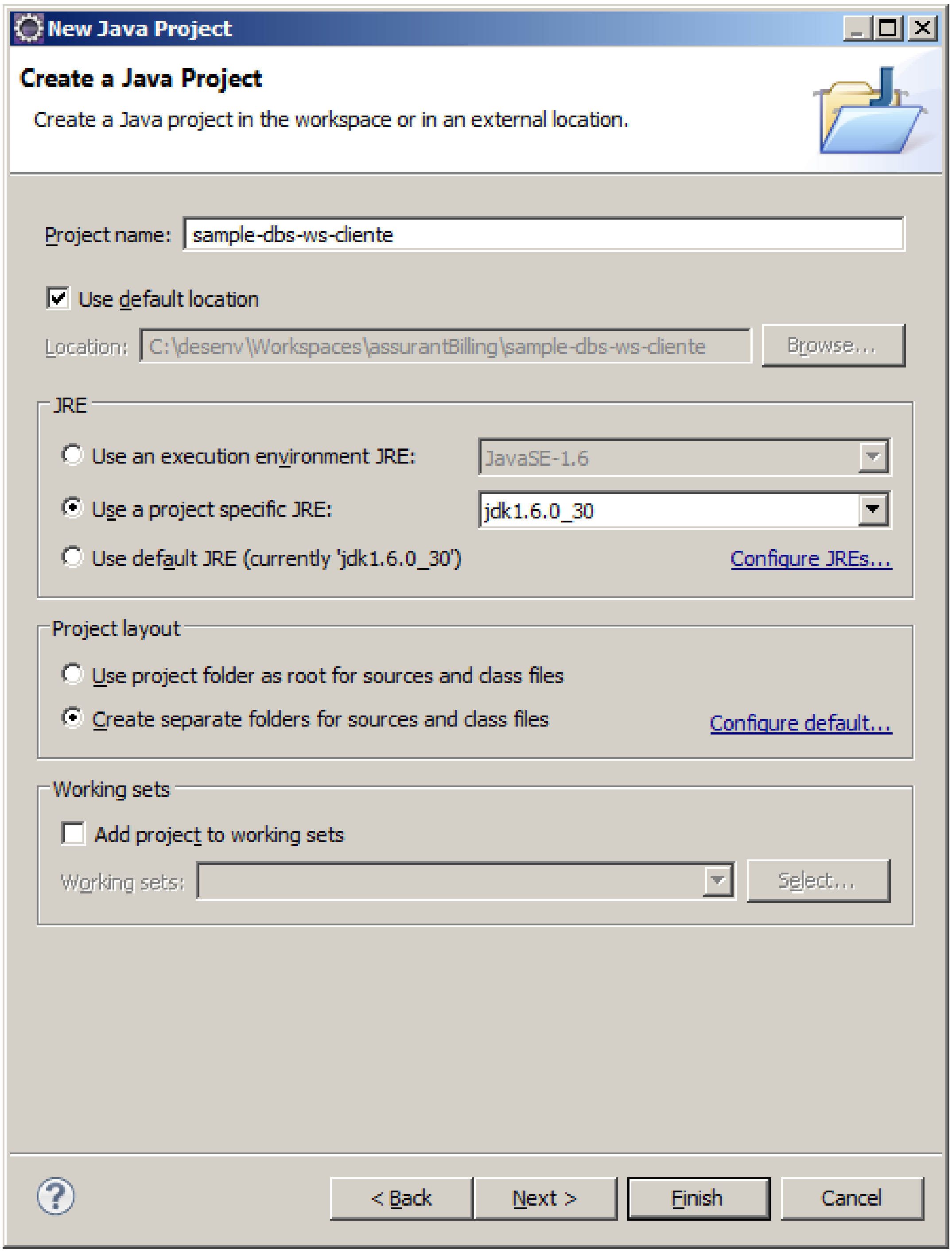
Property	Value
Name	Request 1
Description	
Message Size	1412
Encoding	UTF-8

Request Properties

response time: 444ms (362 bytes)

1 : 1

CRIANDO DO CLIENTE COM AUTENTICAÇÃO CXF



New Java Project

Create a Java Project

Create a Java project in the workspace or in an external location.

Project name: sample-dbs-ws-cliente

☑ Use default location

Location: C:\desenv\Workspaces\assurantBilling\sample-dbs-ws-cliente

JRE

○ Use an execution environment JRE: JavaSE-1.6

● Use a project specific JRE: jdk1.6.0_30

○ Use default JRE (currently 'jdk1.6.0_30')

Project layout

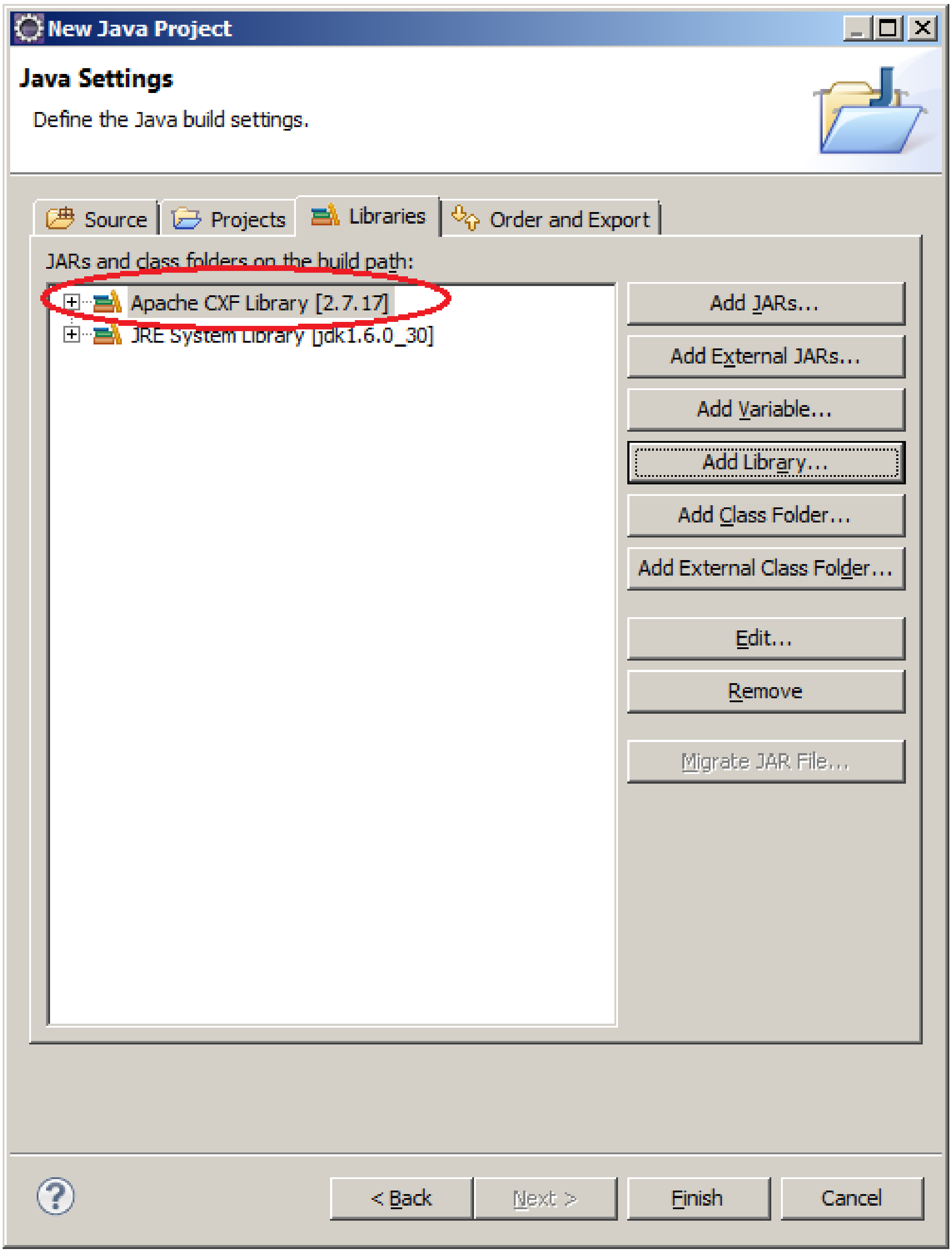
○ Use project folder as root for sources and class files

● Create separate folders for sources and class files

Working sets

□ Add project to working sets

Working sets:



New Java Project

Java Settings

Define the Java build settings.

JARs and class folders on the build path:

- Apache CXF Library [2.7.17]
- JRE System Library [jdk1.6.0_30]

NO PROMPT EXECUTE

```
"c:\LibJava\apache-cxf-2.7.17\bin\wsdl2java.bat" -fe jaxws21 -autoNameResolution -p br.com.delphos.billing.services -d "C:\desenv\Workspaces\assurantBilling\sample-dbs-ws-cliente\src" http://localhost:8080/sample-dbs-ws/services/ws?wsdl
```

CRIAR CLASSES

ClientPasswordCallback

```
package br.com.delphos.billing.security;

import java.io.IOException;

import javax.security.auth.callback.Callback;
import javax.security.auth.callback.CallbackHandler;
import javax.security.auth.callback.UnsupportedCallbackException;

import org.apache.ws.security.WSPasswordCallback;

public class ClientPasswordCallback implements CallbackHandler {

@Override
public void handle(Callback[] callbacks) throws IOException,
UnsupportedCallbackException {

WSPasswordCallback pc = (WSPasswordCallback) callbacks[0];

// set the password for our message.
pc.setPassword("clientstorepass");

}

}
```

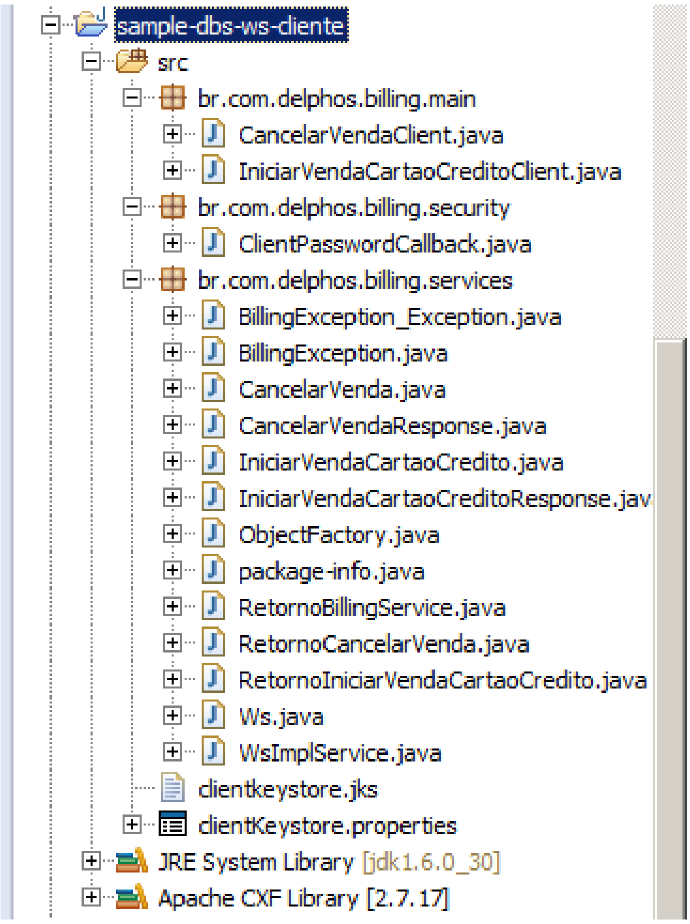
IniciarVendaCartaoCreditoClient

```
package br.com.delphos.billing.main;

import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.HashMap;
import java.util.Map;

import org.apache.commons.lang.time.DateUtils;
import org.apache.cxf.frontend.ClientProxy;
import org.apache.cxf.interceptor.LoggingInInterceptor;
import org.apache.cxf.interceptor.LoggingOutInterceptor;
import org.apache.cxf.jaxws.JaxWsProxyFactoryBean;
import org.apache.cxf.ws.security.wss4j.WSS4JOutInterceptor;
import org.apache.ws.security.handler.WSHandlerConstants;

import br.com.delphos.billing.services.BillingException_Exception;
import br.com.delphos.billing.services.RetornoIniciarVendaCartaoCredito;
import br.com.delphos.billing.services.Ws;
import br.com.delphos.billing.security.ClientPasswordCallback;
import br.com.delphos.billing.util.GerarToken;


public class IniciarVendaCartaoCreditoClient {

public static void main(String[] args) {
// TODO Auto-generated method stub


JaxWsProxyFactoryBean factory = new JaxWsProxyFactoryBean();

// Use the URL defined in the soap address portion of the WSDL
factory.setAddress("http://localhost:8080/sample-dbs-ws/services/ws");

// Utilize the class which was auto-generated by Apache CXF wsdl2java
factory.setServiceClass(Ws.class);

Object client = factory.create();
/********************************SSL*******************************************************/


// Adding Logging Interceptors
LoggingOutInterceptor loggingOutInterceptor = new LoggingOutInterceptor();
loggingOutInterceptor.setPrettyLogging(true);
ClientProxy.getClient(client).getOutInterceptors().add(loggingOutInterceptor);

LoggingInInterceptor loggingInInterceptor = new LoggingInInterceptor();
loggingInInterceptor.setPrettyLogging(true);
ClientProxy.getClient(client).getInInterceptors().add(loggingInInterceptor);

// Set up WS-Security Encryption
// Reference: https://ws.apache.org/wss4j/using.html
Map<String, Object> props = new HashMap<String, Object>();
props.put(WSHandlerConstants.USER, "testkey");
props.put(WSHandlerConstants.ACTION, WSHandlerConstants.ENCRYPT);
props.put(WSHandlerConstants.PASSWORD_TYPE, "PasswordText");
props.put(WSHandlerConstants.ENC_PROP_FILE, "clientKeystore.properties");
props.put(WSHandlerConstants.ENCRYPTION_PARTS, "{Content}
{http://schemas.xmlsoap.org/soap/envelope/}Body");
props.put(WSHandlerConstants.PW_CALLBACK_CLASS, ClientPasswordCallback.class.getName());

WSS4JOutInterceptor wss4jOut = new WSS4JOutInterceptor(props);

ClientProxy.getClient(client).getOutInterceptors().add(wss4jOut);

try {

RetornoIniciarVendaCartaoCredito iniciarVendaCartaoCredito = ((Ws)client).iniciarVendaCartaoCredito(
GerarToken.encriptar(GerarToken.CHAVE_CLIENTE)
,"AIZ"
,"PYB"
,"BMS"
,"50434483346"
,"MACHADO DE ASSIS"
,"teste@teste.com"
,"21"
,"99588562"
,"21"
,"22730419"
,"TST-DPH-001"
,new SimpleDateFormat("dd/MM/yyyy").format(new Date())
,"01"
,"88523123822"
,"MACHADO DE ASSIS"
,"0000000000000001"
,new SimpleDateFormat("MM/yyyy").format( DateUtils.addYears((new Date()), 2))
,"111"
,"002"
,"10"
,"55416"
,new SimpleDateFormat("dd/MM/yyyy").format(new Date())
,new SimpleDateFormat("dd/MM/yyyy").format( DateUtils.addYears((new Date()), 1))
);

System.out.println("iniciarVendaCartaoCredito.getRetorno()="+iniciarVendaCartaoCredito.getRetorno());


} catch (SecurityException e) {

e.printStackTrace();

} catch (IllegalArgumentException e) {

e.printStackTrace();

} catch (BillingException_Exception e) {
// TODO Auto-generated catch block
e.printStackTrace();
}


}

}
```

CancelarVendaClient

```
package br.com.delphos.billing.main;

import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.HashMap;
import java.util.Map;

import org.apache.commons.lang.time.DateUtils;
import org.apache.cxf.frontend.ClientProxy;
import org.apache.cxf.interceptor.LoggingInInterceptor;
import org.apache.cxf.interceptor.LoggingOutInterceptor;
import org.apache.cxf.jaxws.JaxWsProxyFactoryBean;
import org.apache.cxf.ws.security.wss4j.WSS4JOutInterceptor;
import org.apache.ws.security.handler.WSHandlerConstants;

import br.com.delphos.billing.services.BillingException_Exception;
import br.com.delphos.billing.services.RetornoCancelarVenda;
import br.com.delphos.billing.services.RetornoIniciarVendaCartaoCredito;
import br.com.delphos.billing.services.Ws;
import br.com.delphos.billing.security.ClientPasswordCallback;
import br.com.delphos.billing.util.GerarToken;


public class CancelarVendaClient {

public static void main(String[] args) {
// TODO Auto-generated method stub


JaxWsProxyFactoryBean factory = new JaxWsProxyFactoryBean();

// Use the URL defined in the soap address portion of the WSDL
factory.setAddress("http://localhost:8080/sample-dbs-ws/services/ws");

// Utilize the class which was auto-generated by Apache CXF wsdl2java
factory.setServiceClass(Ws.class);

Object client = factory.create();
/*******************************SSL******************************************************/


// Adding Logging Interceptors
LoggingOutInterceptor loggingOutInterceptor = new LoggingOutInterceptor();
loggingOutInterceptor.setPrettyLogging(true);
ClientProxy.getClient(client).getOutInterceptors().add(loggingOutInterceptor);

LoggingInInterceptor loggingInInterceptor = new LoggingInInterceptor();
loggingInInterceptor.setPrettyLogging(true);
ClientProxy.getClient(client).getInInterceptors().add(loggingInInterceptor);

// Set up WS-Security Encryption
// Reference: https://ws.apache.org/wss4j/using.html
Map<String, Object> props = new HashMap<String, Object>();
props.put(WSHandlerConstants.USER, "testkey");
props.put(WSHandlerConstants.ACTION, WSHandlerConstants.ENCRYPT);
props.put(WSHandlerConstants.PASSWORD_TYPE, "PasswordText");
props.put(WSHandlerConstants.ENC_PROP_FILE, "clientKeystore.properties");
props.put(WSHandlerConstants.ENCRYPTION_PARTS, "{Content}
{http://schemas.xmlsoap.org/soap/envelope/}Body");
props.put(WSHandlerConstants.PW_CALLBACK_CLASS, ClientPasswordCallback.class.getName());

WSS4JOutInterceptor wss4jOut = new WSS4JOutInterceptor(props);

ClientProxy.getClient(client).getOutInterceptors().add(wss4jOut);

try {

RetornoCancelarVenda cancelarVenda = ((Ws)client).cancelarVenda(
"AIZ",
"PYB",
"BMS",
GerarToken.encriptar(GerarToken.CHAVE_CLIENTE),
"98324985972",
"TST-DPH-001",
"001",
"-1",
"S"
);
System.out.println("cancelarVenda.getRetorno()="+cancelarVenda.getRetorno());

} catch (SecurityException e) {

e.printStackTrace();

} catch (IllegalArgumentException e) {

e.printStackTrace();

} catch (BillingException_Exception e) {
// TODO Auto-generated catch block
e.printStackTrace();
}


}

}
```

OBS.: OS DEMAIS ARQUIVOS FORAM GERADOS PELO COMANDO EM PROMPT ANTERIORMENTE



```
sample-dbs-ws-cliente
  src
    br.com.delphos.billing.main
      CancelarVendaClient.java
      IniciarVendaCartaoCreditoClient.java
    br.com.delphos.billing.security
      ClientPasswordCallback.java
    br.com.delphos.billing.services
      BillingException_Exception.java
      BillingException.java
      CancelarVenda.java
      CancelarVendaResponse.java
      IniciarVendaCartaoCredito.java
      IniciarVendaCartaoCreditoResponse.jav
      ObjectFactory.java
      package-info.java
      RetornoBillingService.java
      RetornoCancelarVenda.java
      RetornoIniciarVendaCartaoCredito.java
      Ws.java
      WsImplService.java
    clientkeystore.jks
    clientKeystore.properties
  JRE System Library [jdk1.6.0_30]
  Apache CXF Library [2.7.17]
```

EXECUTE COM "JAVA APLICATION" OS ARQUIVOS

IniciarVendaCartaoCreditoClient

```
--------------------------------------
12/11/2015 09:38:13 org.apache.cxf.services.WsService.WsPort.Ws
INFO: Inbound Message
----------------------------
ID: 1
Response-Code: 200
Encoding: UTF-8
Content-Type: text/xml;charset=UTF-8
Headers: {content-type=[text/xml;charset=UTF-8], Date=[Thu, 12 Nov 2015 11:38:13 GMT], Server=
[Apache-Coyote/1.1], transfer-encoding=[chunked]}
Payload: <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body>
<ns2:iniciarVendaCartaoCreditoResponse xmlns:ns2="http://servicos.billing.delphos.com.br/">
<return>
<retorno>0</retorno>
<codigoProduto>PYB</codigoProduto>
<codigoVenda>TST-DPH-001</codigoVenda>
<codigoVendaBilling>46</codigoVendaBilling>
</return>
</ns2:iniciarVendaCartaoCreditoResponse>
```

```
</soap:Body>
</soap:Envelope>

--------------------------------------
iniciarVendaCartaoCredito.getRetorno()=0
```

CancelarVendaClient

```
--------------------------------------
12/11/2015 09:42:09 org.apache.cxf.services.WsService.WsPort.Ws
INFO: Inbound Message
----------------------------
ID: 1
Response-Code: 200
Encoding: UTF-8
Content-Type: text/xml;charset=UTF-8
Headers: {content-type=[text/xml;charset=UTF-8], Date=[Thu, 12 Nov 2015 11:42:09 GMT], Server=[Apache-Coyote/1.1], transfer-encoding=[chunked]}
Payload: <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body>
<ns2:cancelarVendaResponse xmlns:ns2="http://servicos.billing.delphos.com.br/">
<return>
<retorno>0</retorno>
<codigoProduto>PYB</codigoProduto>
<codigoVenda>TST-DPH-001</codigoVenda>
</return>
</ns2:cancelarVendaResponse>
</soap:Body>
</soap:Envelope>

--------------------------------------
cancelarVenda.getRetorno()=0
```