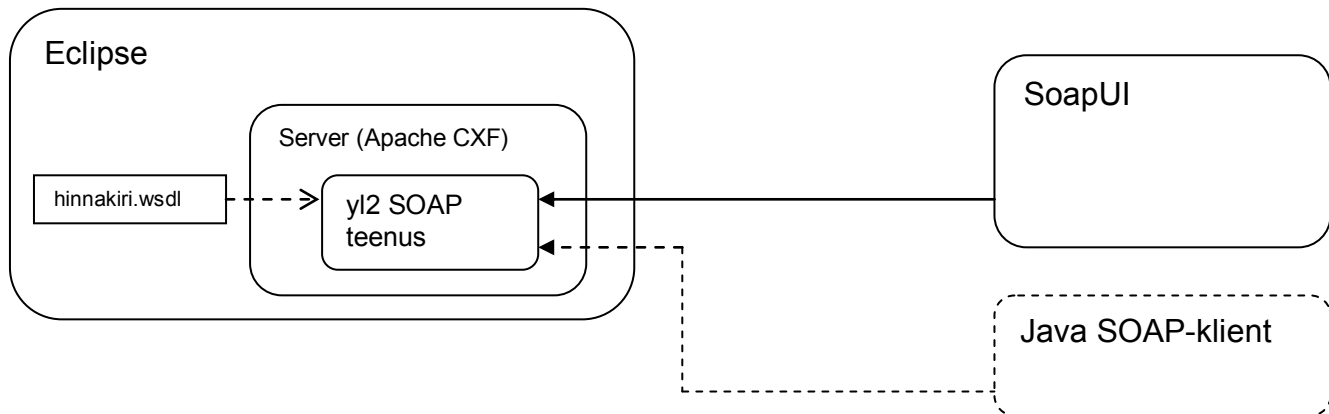


SOAP serveri tegemine , käivitamine ja testimine: Eclipse, Apache CXF ja SoapUI  
Koduülesande 2 abimaterjal.

1. Eeldused.....	2
2. Konfigureerime Eclipse .....	3
3. Teeme uue „Dynamic Web” projekti: .....	8
3.1. Paneme paika Target Runtime'i ja konfiguratsiooniks CXF-i.....	8
3.2. Lisame wsdl-i projektile.....	11
3.3. Teeme veebteenuse.....	14
4. SoapUI-ga testimine.....	24
5. Täiendame serveri osa koodi. ....	25
6. Java klient. ....	26



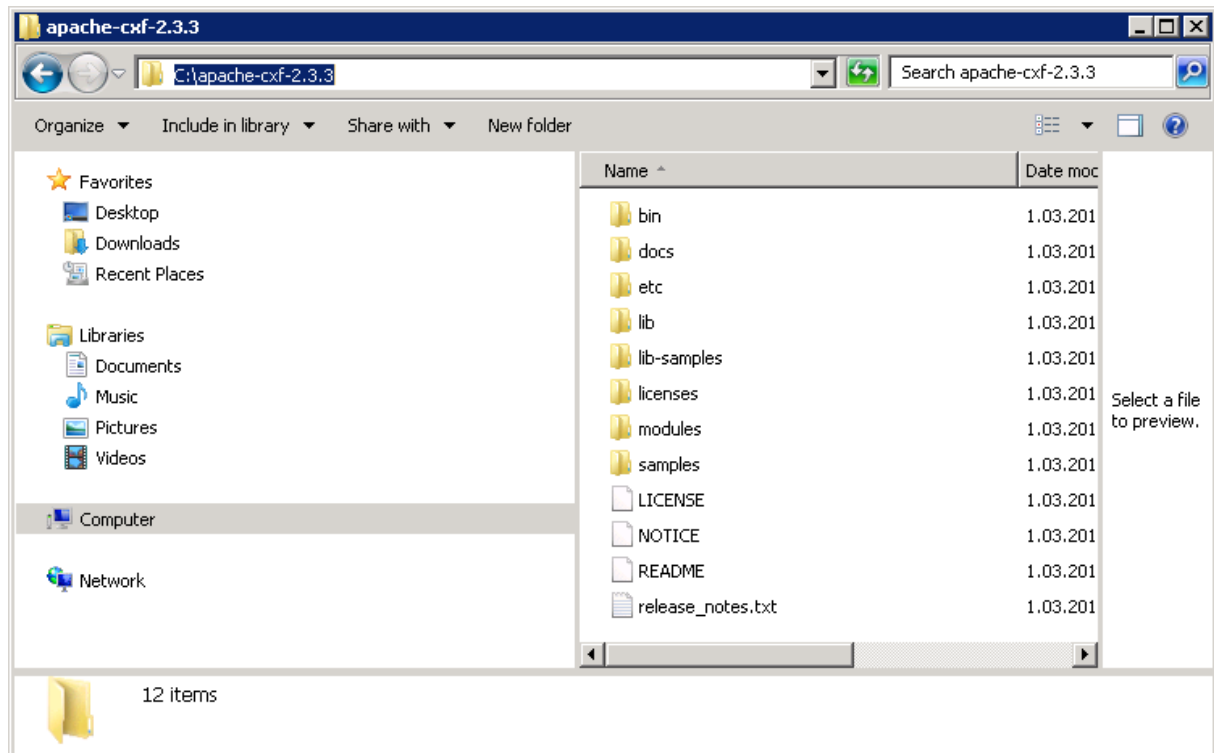
## 1. Eeldused

1) Installeeritud Apache CXF  
(lahti pakitud arvutisse)

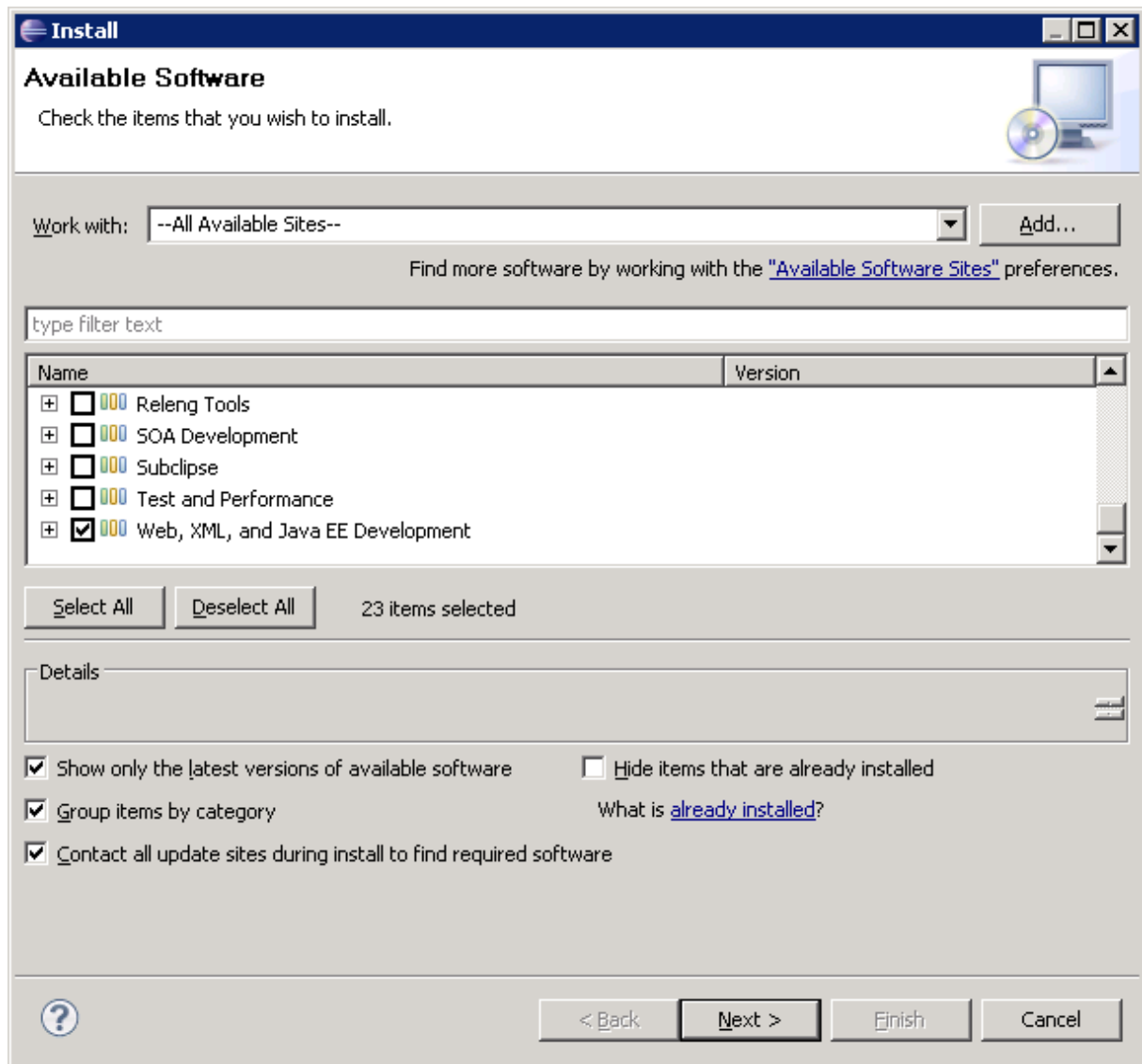
<http://cxf.apache.org/>

Tömmata alla nt. zip-versioon ja pakkida lahti oma arvutisse, antud juhendis kataloogi

2.3.3				
The 2.3.3 release is our latest release. For more information please see the <a href="#">release notes</a> .				
Description	File	MD5	SHA1	PGP
Source distribution	apache-cxf-2.3.3-src.tar.gz	apache-cxf-2.3.3-src.tar.gz.md5	apache-cxf-2.3.3-src.tar.gz.sha1	apache-cxf-2.3.3-src.tar.gz.asc
	apache-cxf-2.3.3-src.zip	apache-cxf-2.3.3-src.zip.md5	apache-cxf-2.3.3-src.zip.sha1	apache-cxf-2.3.3-src.zip.asc
Binary distribution	apache-cxf-2.3.3.tar.gz	apache-cxf-2.3.3.tar.gz.md5	apache-cxf-2.3.3.tar.gz.sha1	apache-cxf-2.3.3.tar.gz.asc
	apache-cxf-2.3.3.zip	apache-cxf-2.3.3.zip.md5	apache-cxf-2.3.3.zip.sha1	apache-cxf-2.3.3.zip.asc



2) Eclipse Web , XML, J2EE plugin installeeritud (esimese ülesande jaoks peaks see olema nagunii juba tehtud).

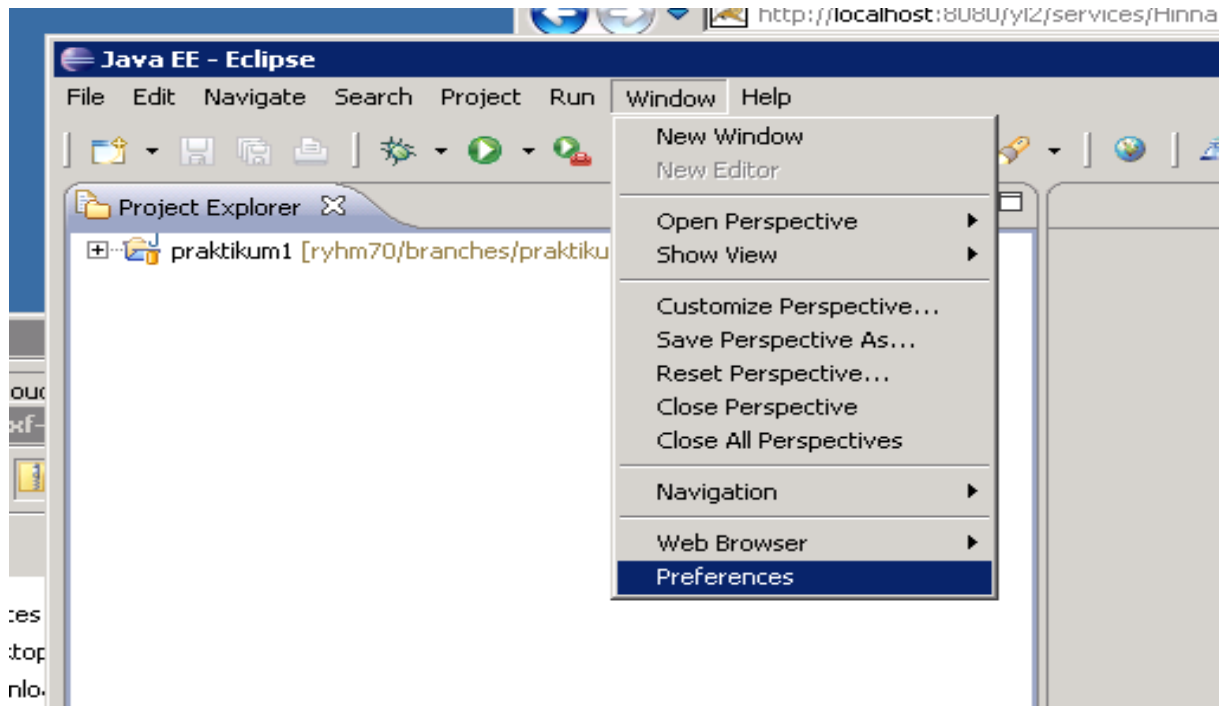


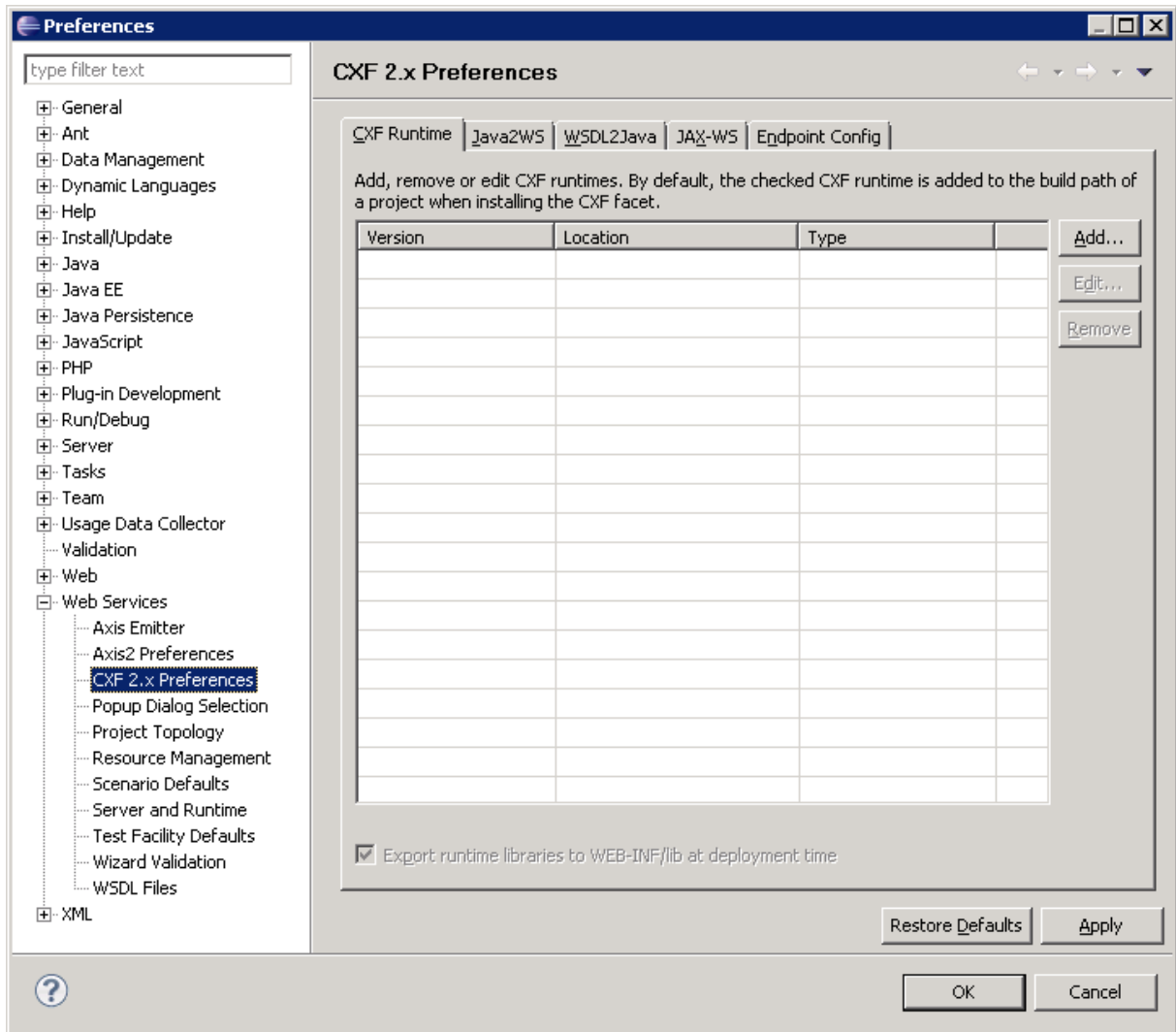
3) SoapUI on installeeritud

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

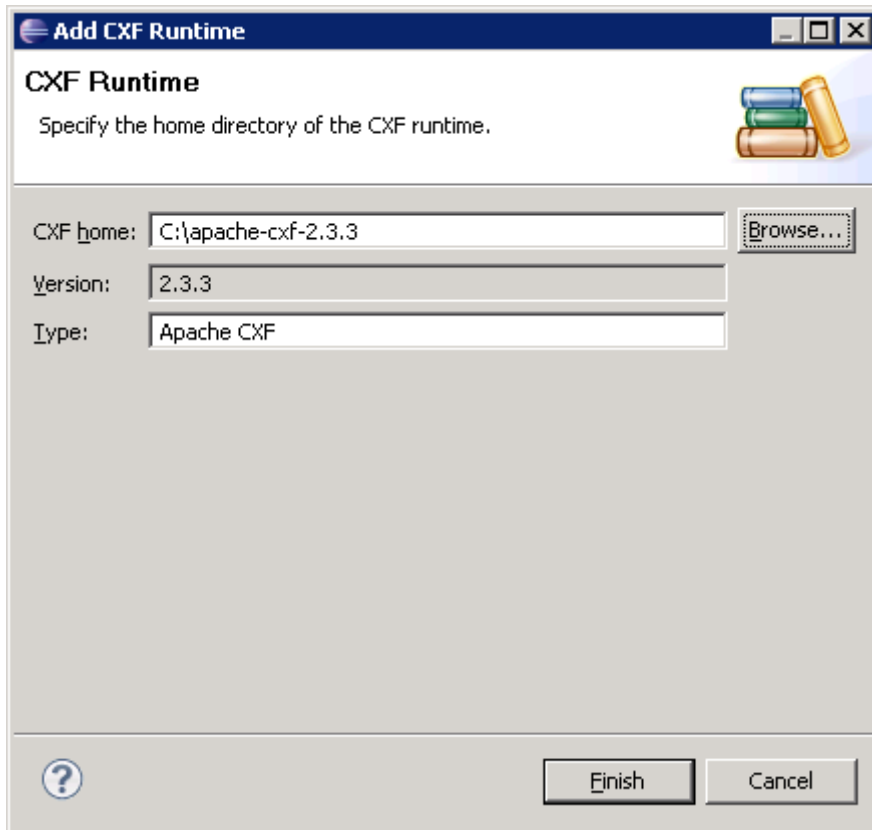
4) Eclipse subclipse plugin on installeeritud (selleks et versioone SVN serverisse saata, peaks olema ylesande 1 raames ka juba tehtud).

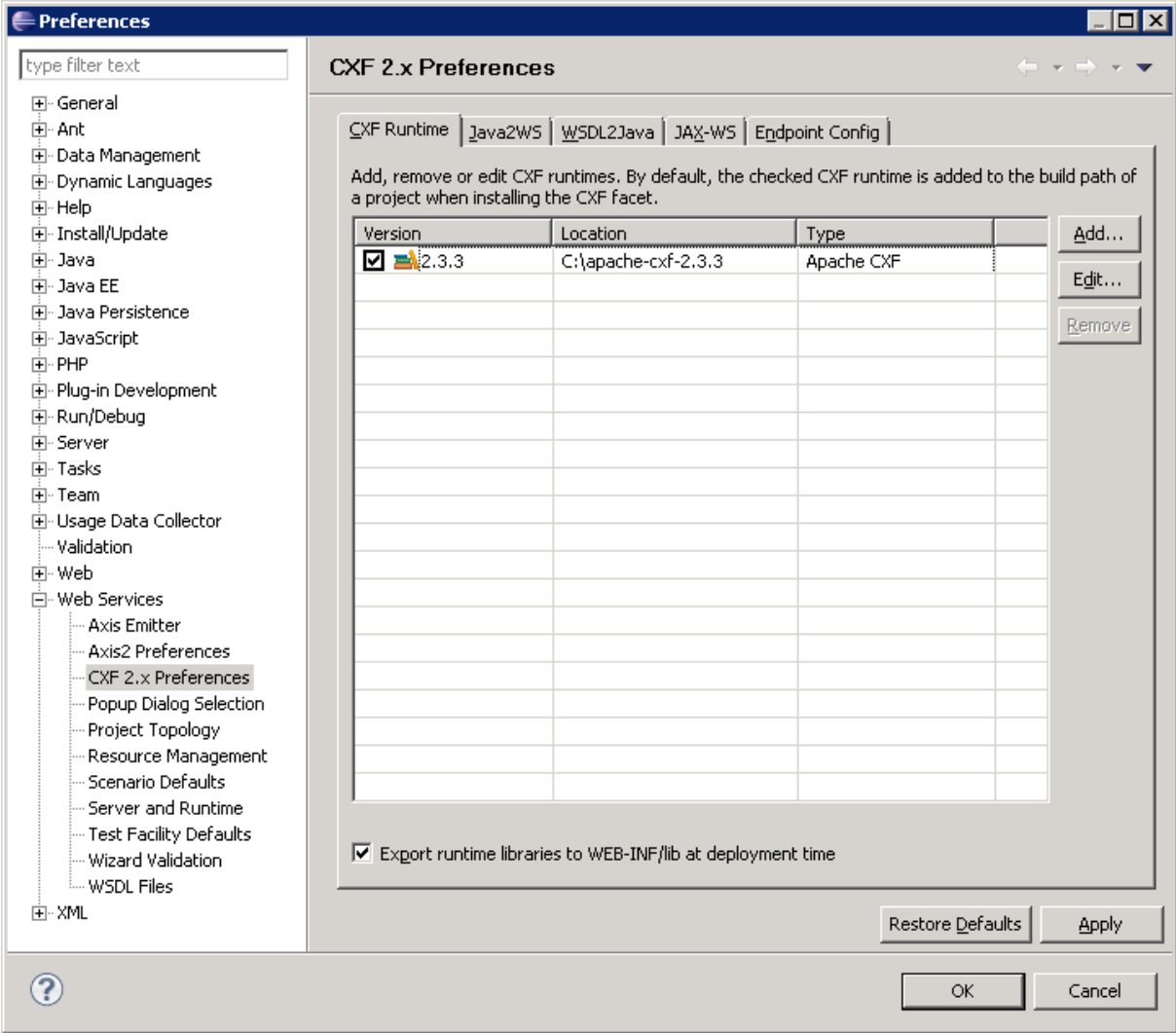
## 2. Konfigureerime Eclipse

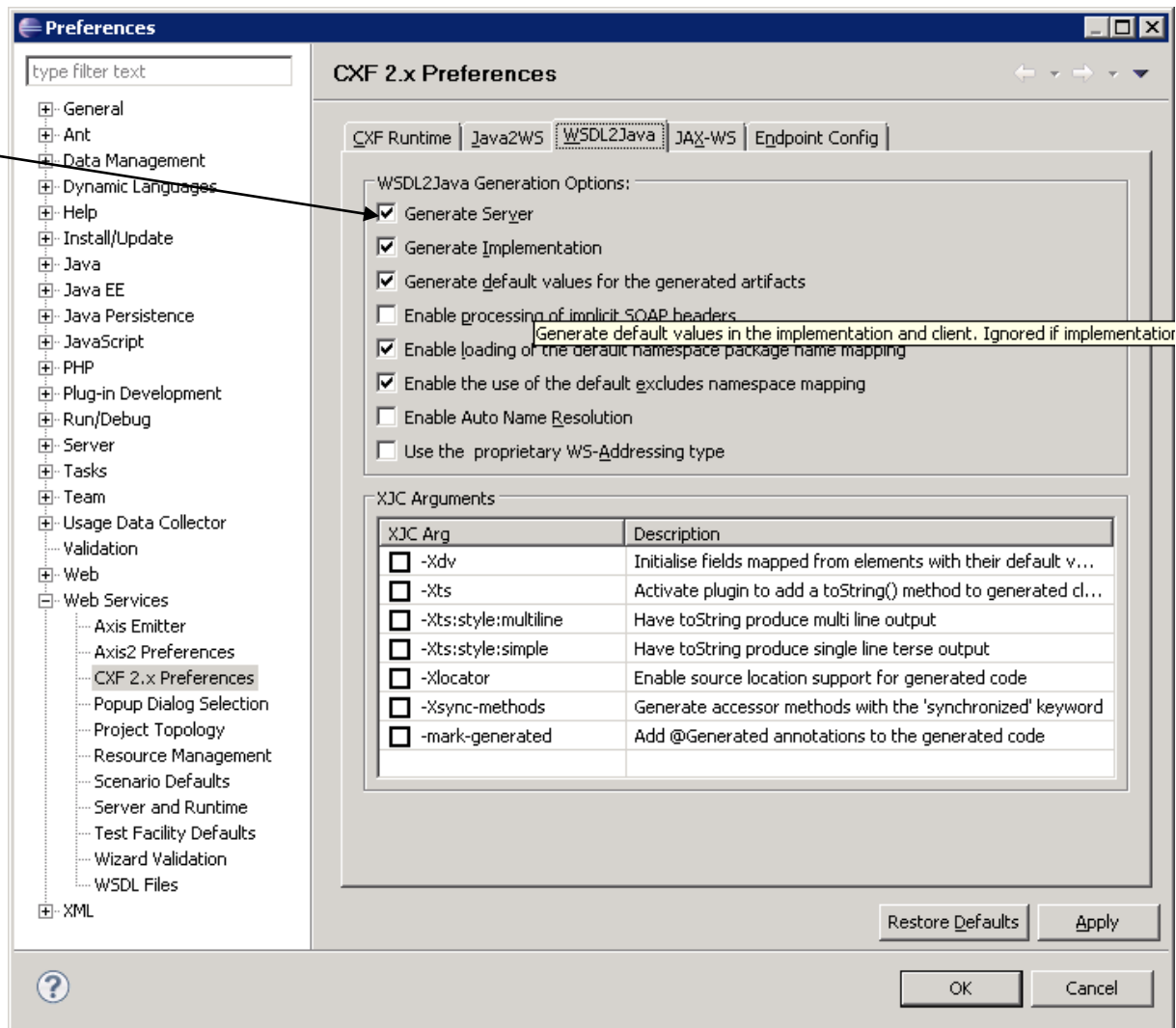




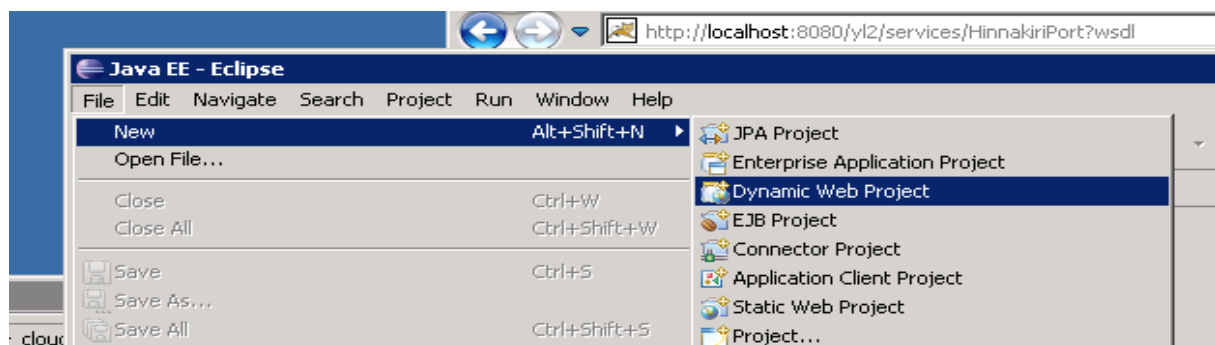
[Add..]







### 3. Teeme uue „Dynamic Web” projekti:



#### 3.1. Paneme paika Target Runtime'i ja konfiguratsiooniks CXF-i



**New Dynamic Web Project**

### Dynamic Web Project

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

Project name:

Project location

Use default location

Location:

Target runtime

Dynamic web module version

Configuration

Configures a Dynamic Web Project with CXF using Web Module v2.5 and Java v5.0

EAR membership

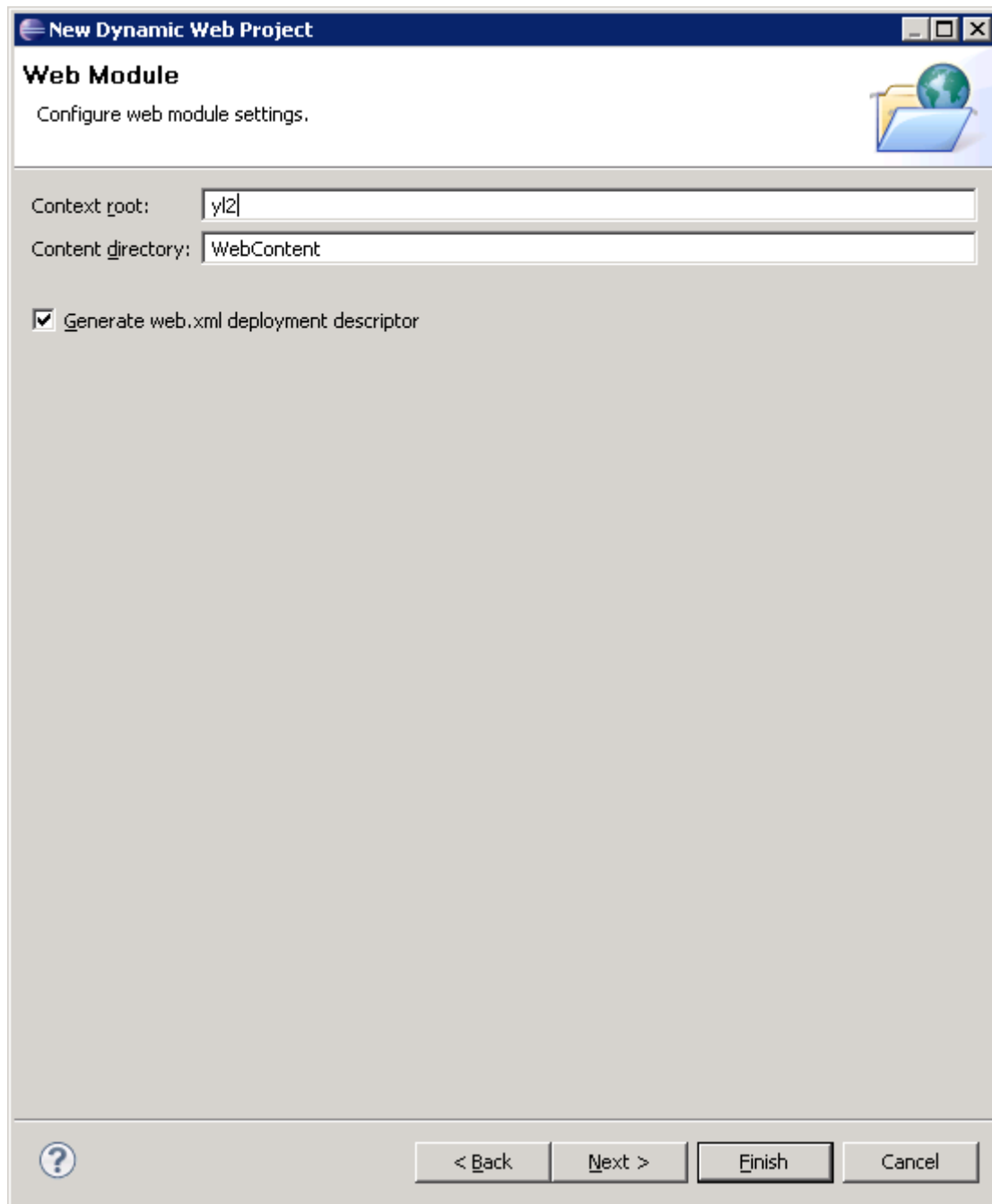
Add project to an EAR

EAR project name:

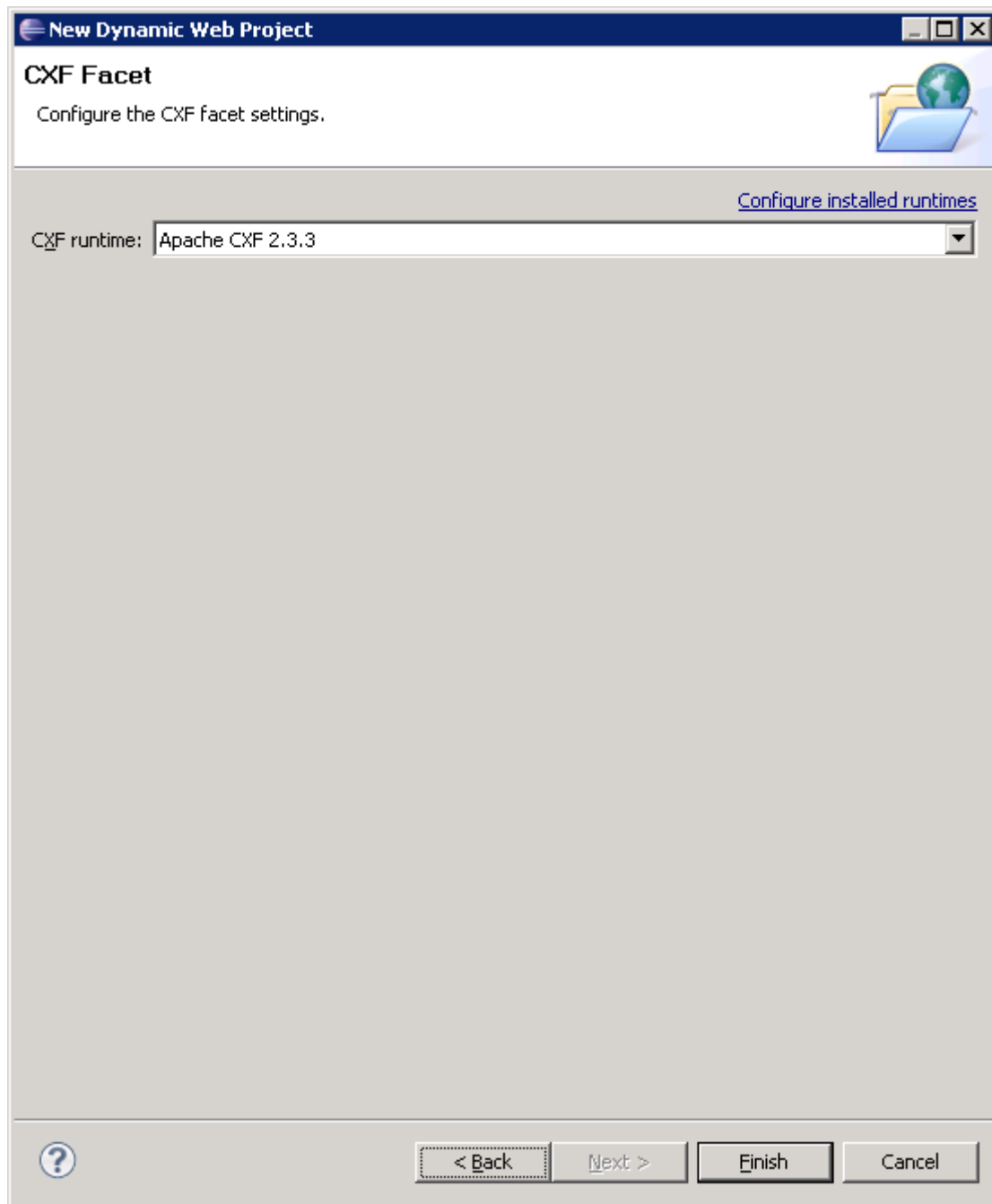
Working sets

Add project to working sets

Working sets:

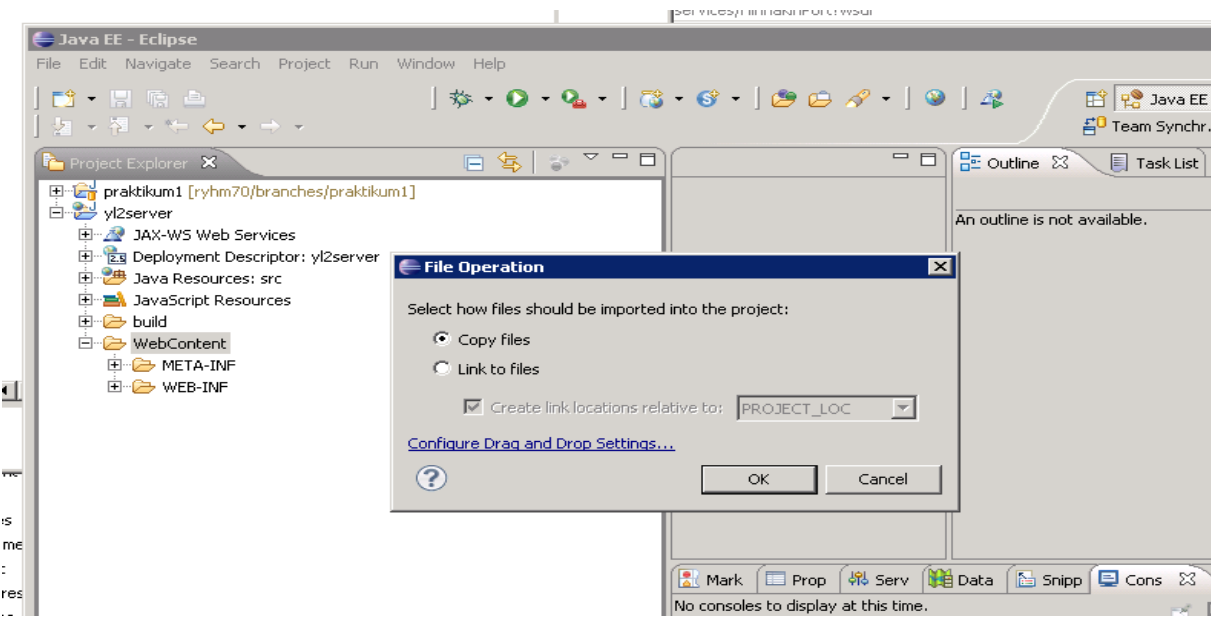
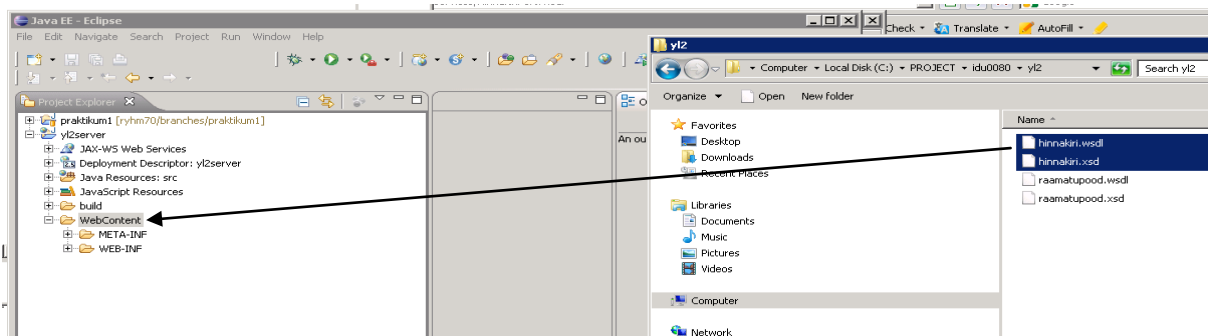


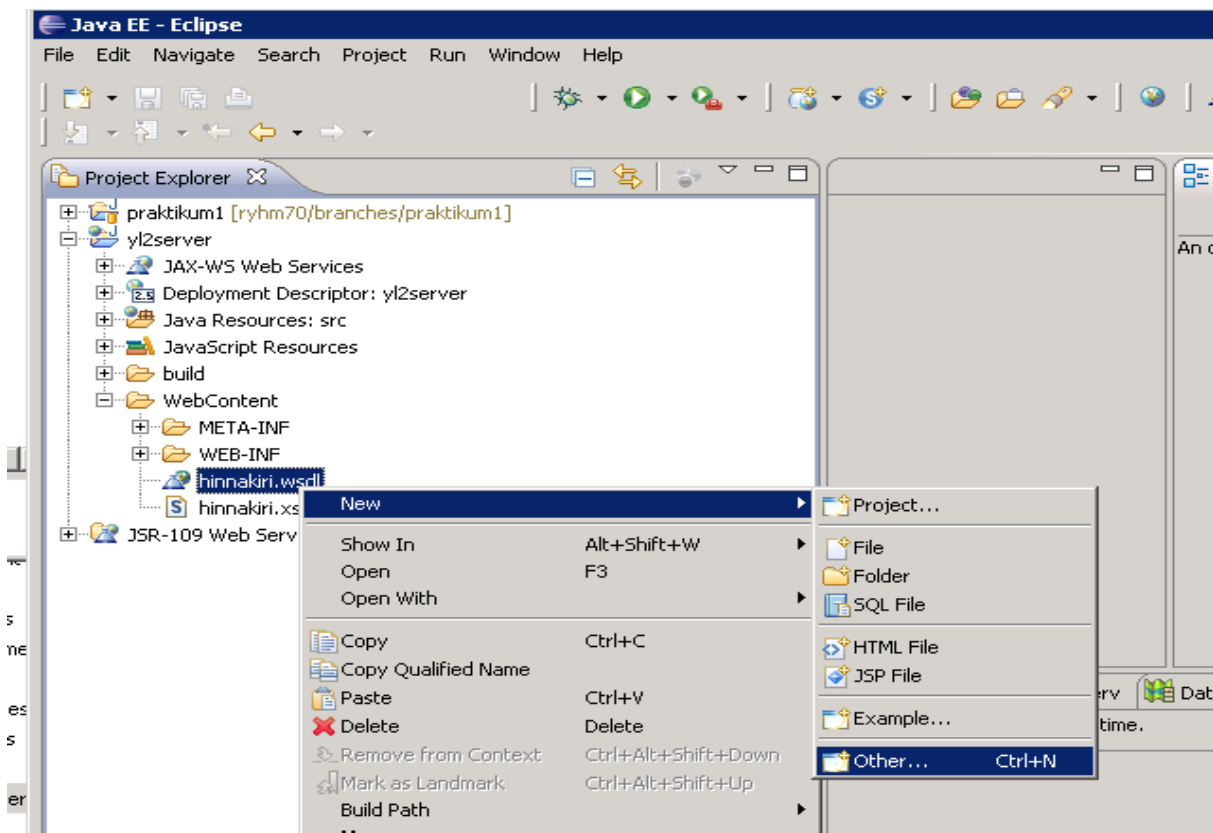
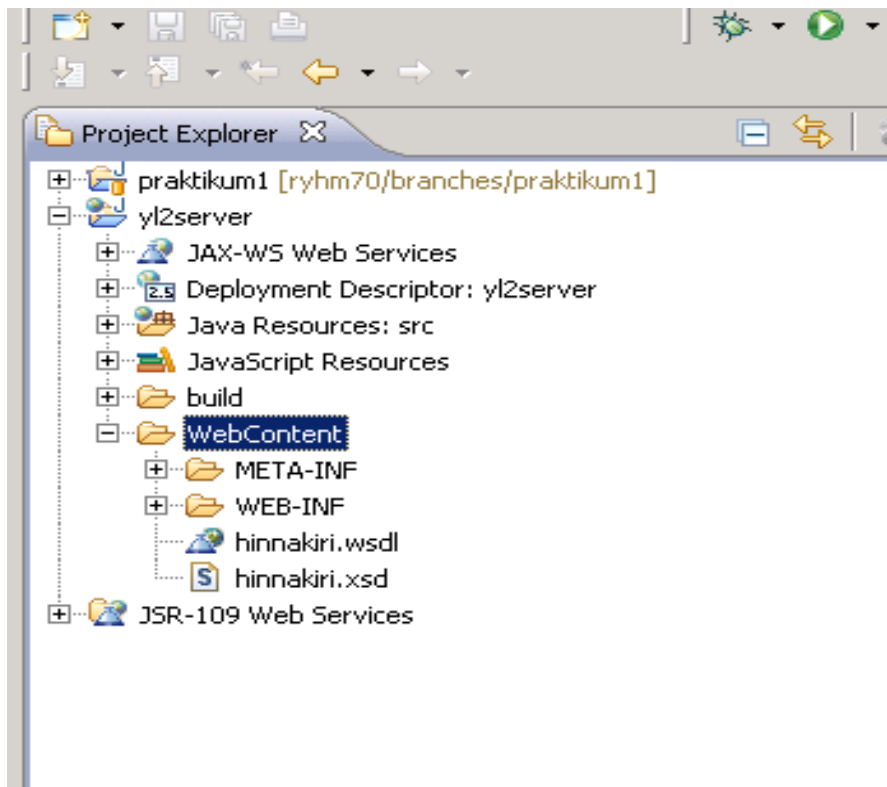
Sellel aadressil <http://localhost:8080/yl2> näeme hiljem oma teenust.



### 3.2. Lisame wsdl-i projektile.

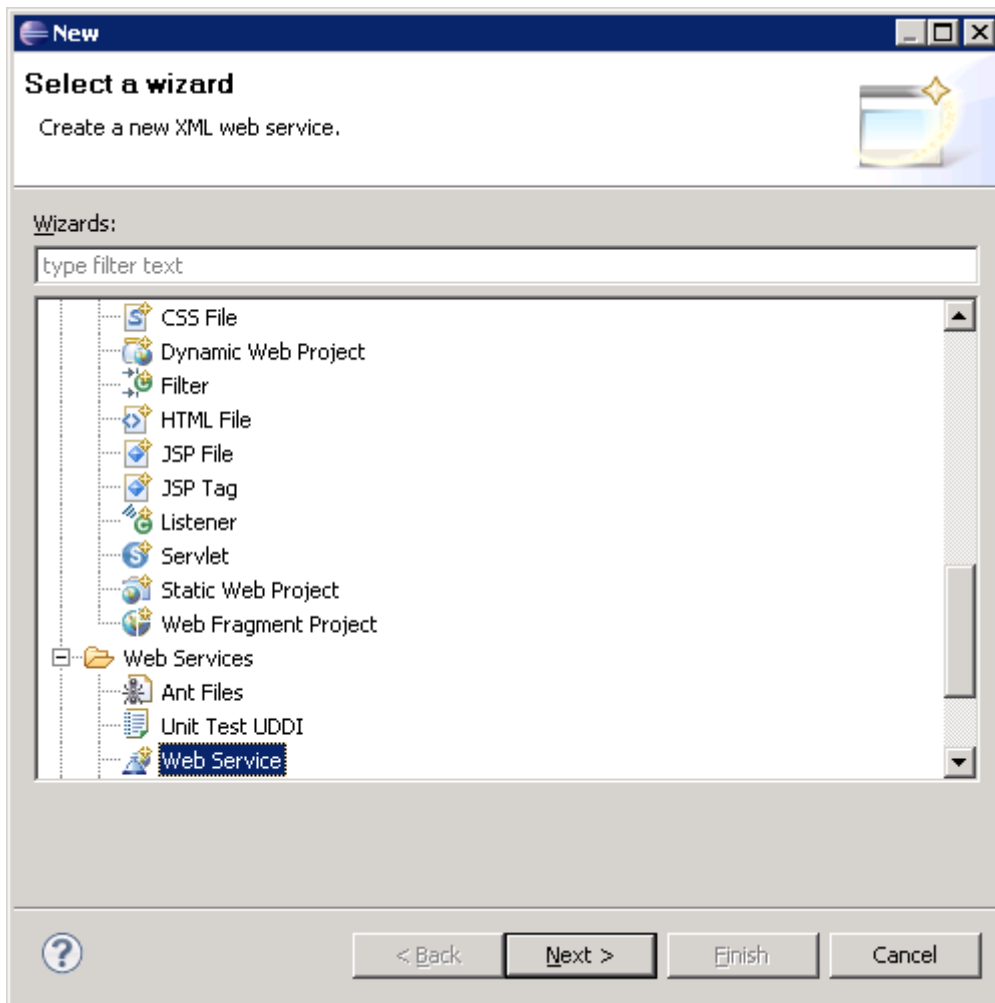
Kopeerime projekti WebContent kataloogi aine kodulehelt tõmmatud hinnakiri.xsd ja hinnakiri.xsd failid

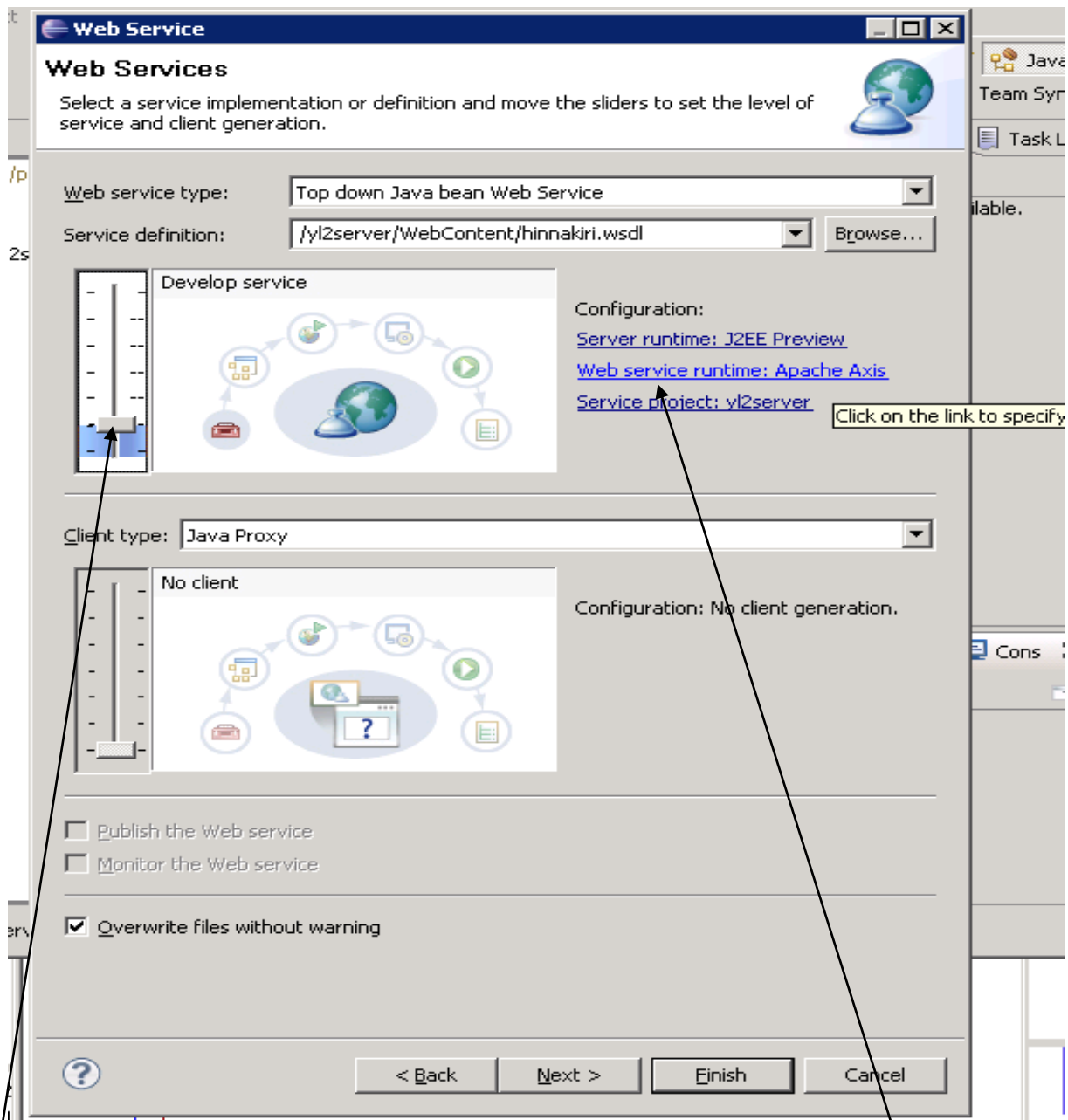




### 3.3. Teeme veebteenuse.

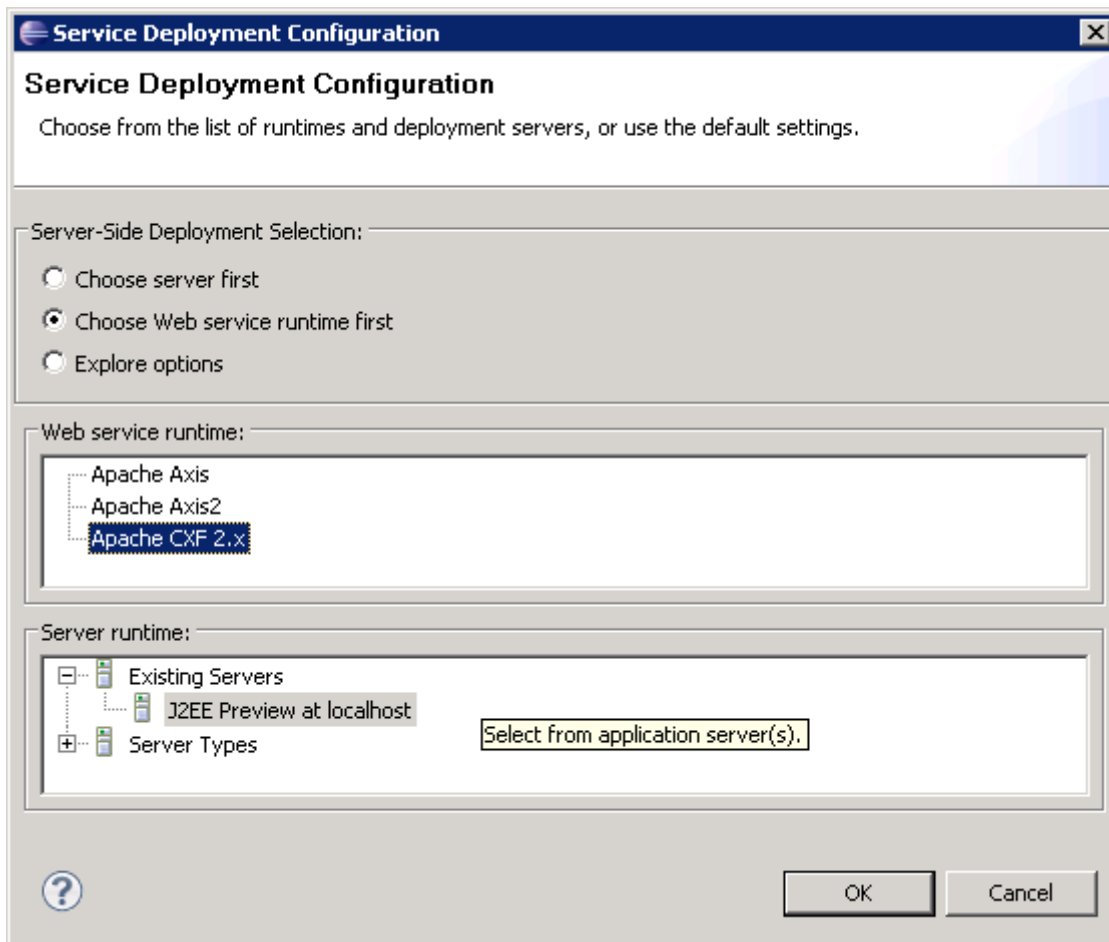
Läheme hiirega wsdl-i peale .





selle tirime alla

selle muudame ära – Axise asemel CXF





Java EE - Eclipse

File Edit Navigate Search Project

Project Explorer

- praktikum1 [ryhm70/branches/p
- yl2server
  - JAX-WS Web Services
  - Deployment Descriptor: yl2s
  - Java Resources: src
  - JavaScript Resources
  - build
  - WebContent
    - META-INF
    - WEB-INF
      - hinnakiri.wsdl
      - hinnakiri.xsd
  - JSR-109 Web Services

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: Top down Java bean Web Service

Service definition: /yl2server/WebContent/hinnakiri.wsdl [Browse...](#)

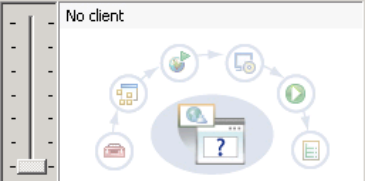
**Develop service**



Configuration:  
[Server runtime: J2EE Preview](#)  
[Web service runtime: Apache CXF 2.x](#)  
[Service project: yl2server](#)

Client type: Java Proxy

**No client**



Configuration: No client generation.

Publish the Web service  
 Monitor the Web service

Overwrite files without warning

hinnakiri.wsdl - yl2serv

**Web Service**  
**Apache CXF 2.3.3 Web Service WSDL2Java Configuration**  
 Configure the Apache CXF WSDL to Java options.

Output Directory:

Package Name:

Specify WSDL Namespace to Package Name Mappings

WSDL Namespace	Package Name
<input type="checkbox"/> http://schemas.xmlsoap.org/wsdl/	org.xmlsoap.schemas.wsdl
<input type="checkbox"/> http://schemas.xmlsoap.org/soap/en...	org.xmlsoap.schemas.soap.encoding
<input checked="" type="checkbox"/> http://www.ttu.ee/idu0080/hinnakiri/...	ee.ttu.idu0080.hinnakiri.types
<input type="checkbox"/> http://www.w3.org/2001/XMLSchema	org.w3.2001.xmlschema
<input type="checkbox"/> http://schemas.xmlsoap.org/wsdl/soap/	org.xmlsoap.schemas.wsdl.soap

Service Name:

Binding Files:

Add...  
Remove...

muudame siin („wsdl” lõpust ära)


muudame siin („types” lõppu)

valime

**Web Service**

## Apache CXF 2.3.3 Web Service WSDL2Java Configuration

The page allows the overriding of the default preferences for this Web Service generation.




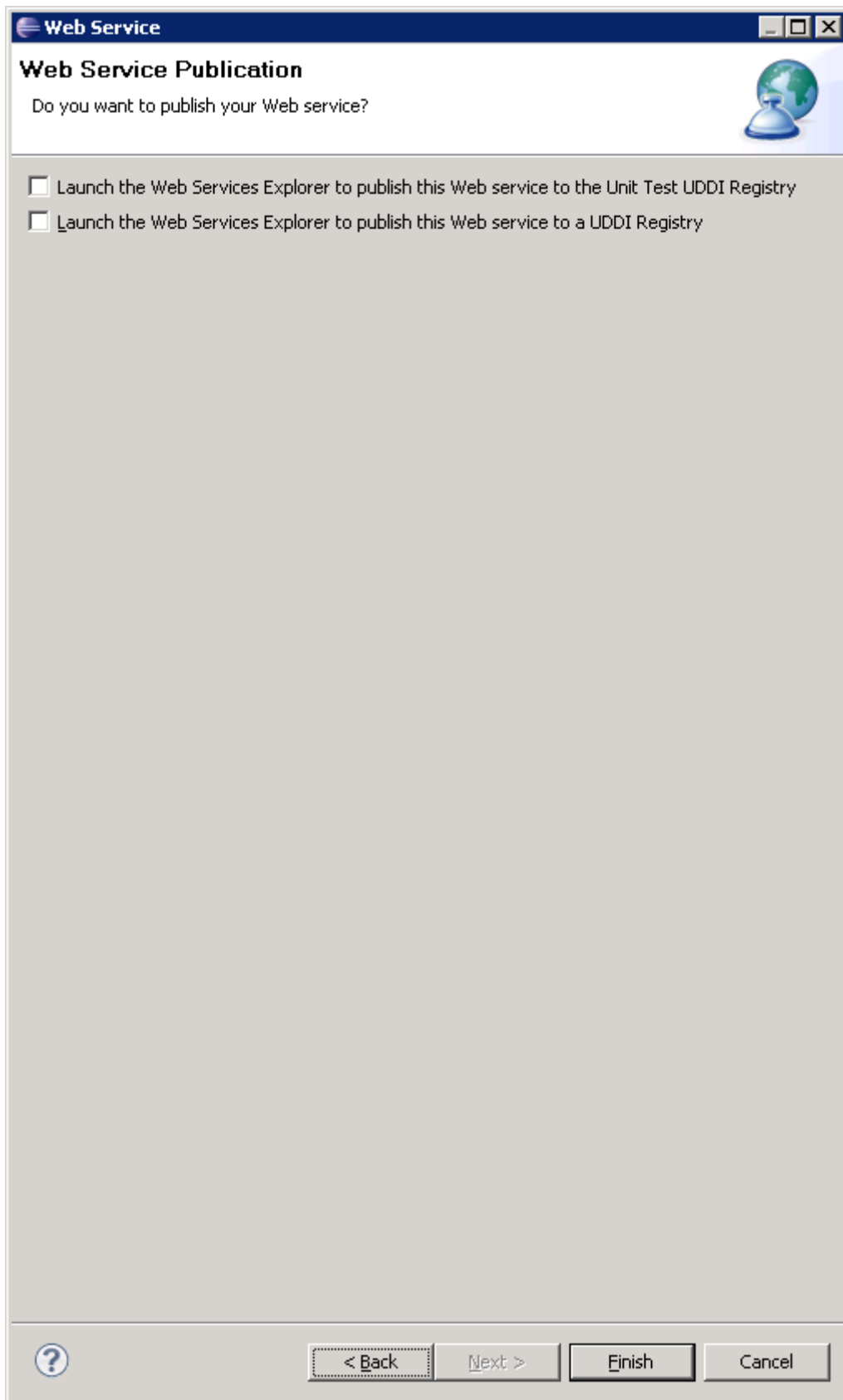
**WSDL2Java Generation Options**

- Generate Ser\_ver
- Generate Implementation
- Generate default values for the generated artifacts
- Enable processing of implicit SOAP headers
- Enable loading of the default namespace package name mapping
- Enable the use of the default excludes namespace mapping
- Enable Auto Name Resolution
- Use the Apache CXF proprietary WS-Addressing type

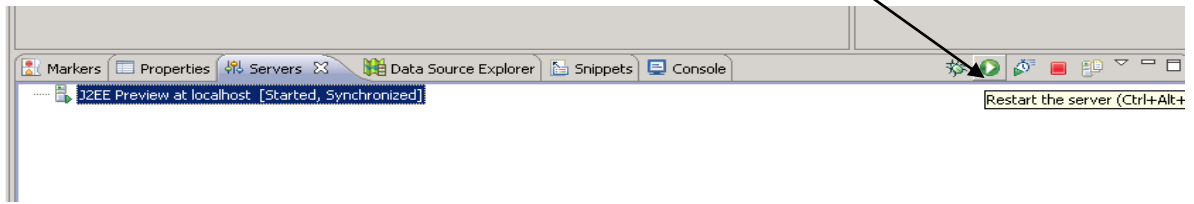
**XJC Arguments**

XJC Arg	Description
<input type="checkbox"/> -Xdv	Initialise fields mapped from elements with their default v...
<input type="checkbox"/> -Xts	Activate plugin to add a toString() method to generated ...
<input type="checkbox"/> -Xts:style:multiline	Have toString produce multi line output
<input type="checkbox"/> -Xts:style:simple	Have toString produce single line terse output
<input type="checkbox"/> -Xlocator	Enable source location support for generated code
<input type="checkbox"/> -Xsync-methods	Generate accessor methods with the 'synchronized' keyw...
<input type="checkbox"/> -mark-generated	Add @Generated annotations to the generated code





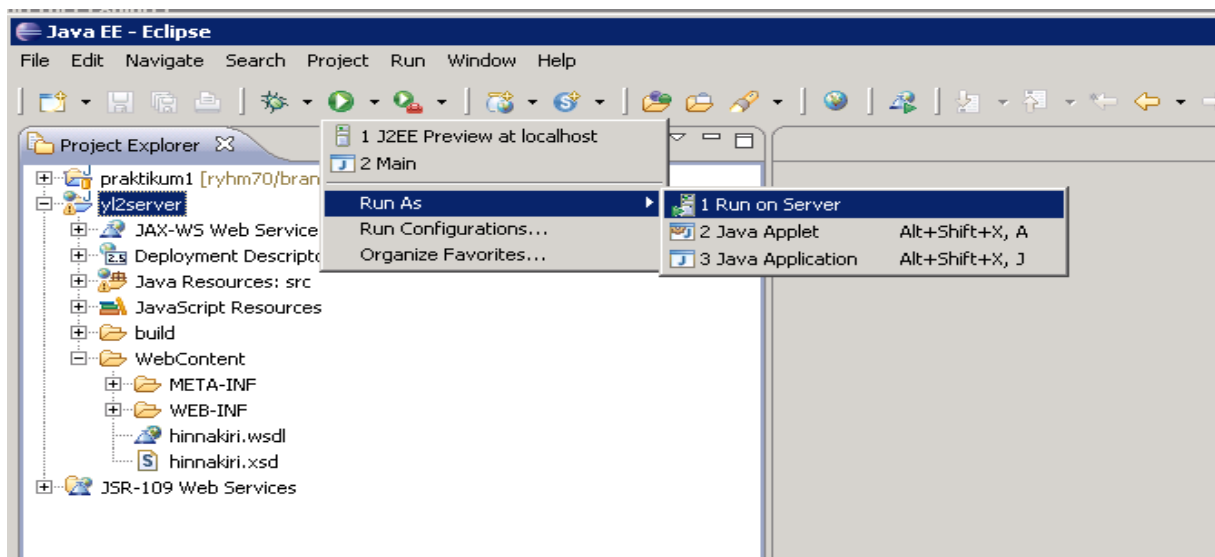
Stardime serveri (kui ei ole veel starditud automataselt)

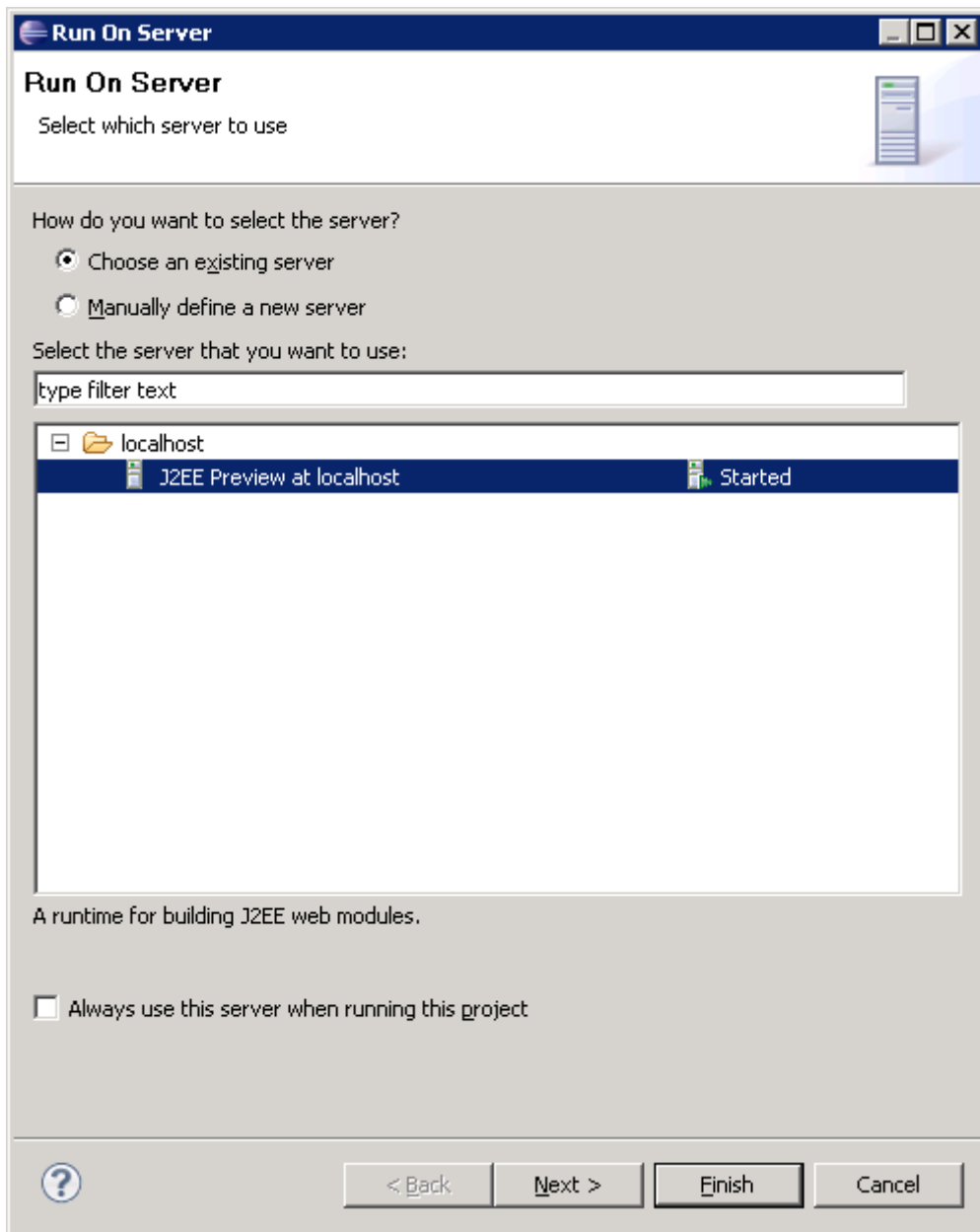


Läheme adressile

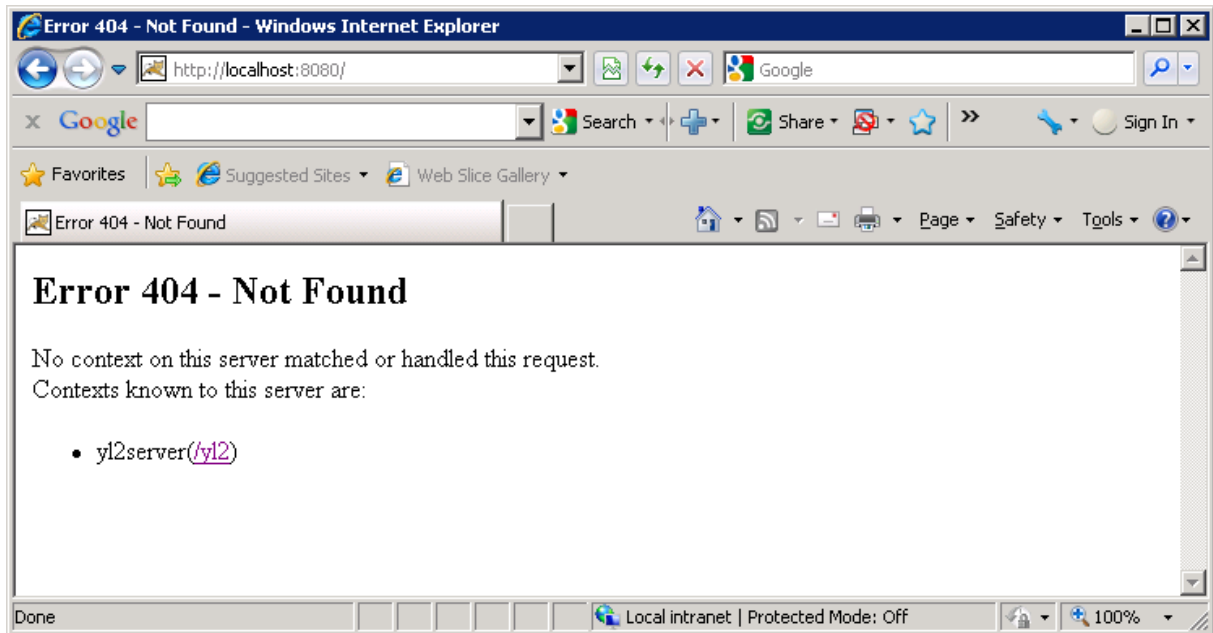
<http://localhost:8080/yl2/services/>

Kui sellel adressil midagi paista siis on meie projekti rakendus serveriga seostamata (ei ole installeeritud meie localhost serveril), teeme nii:

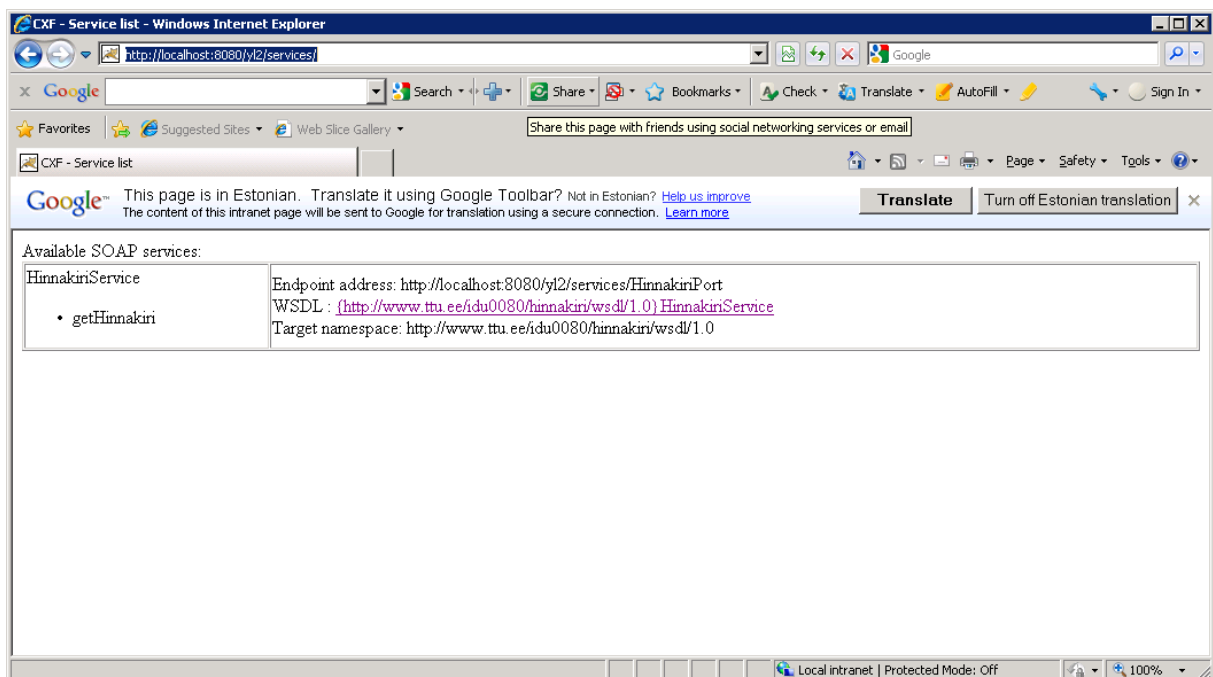




Kui kõik korras (server starditud ja projekti rakendus serveri peal ) siis peaks brauseris avanema selline pilt:



<http://localhost:8080/yl2/services/>



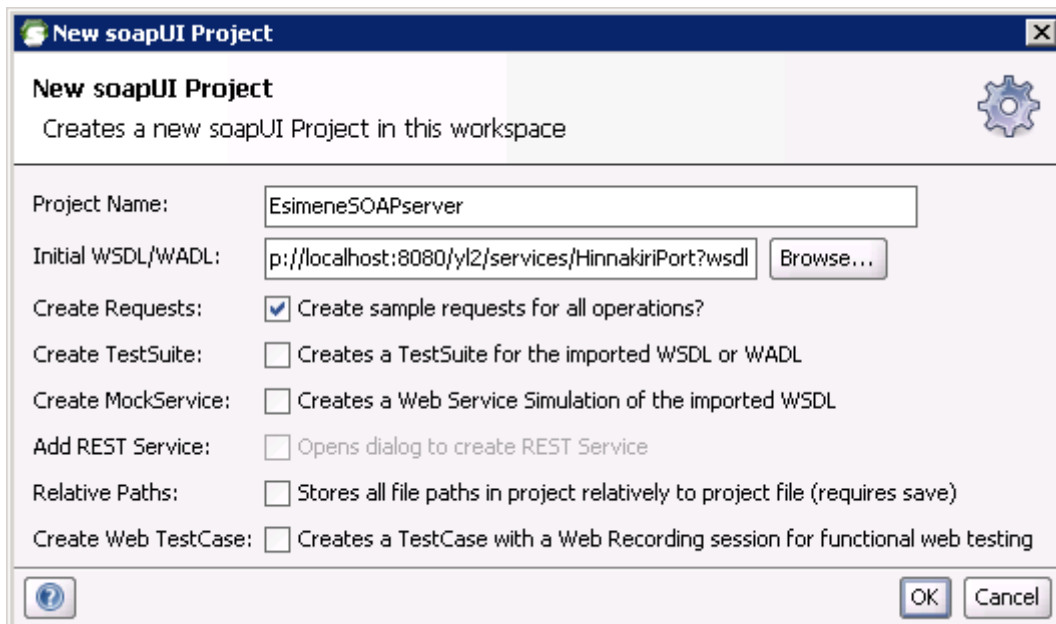
Siit saame wsdl-i aadressi SoapUI-s kasutamiseks:

<http://localhost:8080/yl2/services/HinnakiriPort?wsdl>

```
<?xml version="1.0" encoding="UTF-8" ?>
- <definitions name="HinnakiriService" targetNamespace="http://www.ttuu.ee/idu0080/hinnakiri/wsdl/1.0"
  xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:ns="http://www.ttuu.ee/idu0080/hinnakiri/1.0"
  xmlns:ns1="http://schemas.xmlsoap.org/soap/encoding/" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tns="http://www.ttuu.ee/idu0080/hinnakiri/wsdl/1.0" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
- <types>
  - <xsd:schema xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:ns="http://www.ttuu.ee/idu0080/hinnakiri/1.0"
    xmlns:ns1="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
    xmlns:tns="http://www.ttuu.ee/idu0080/hinnakiri/wsdl/1.0"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <xsd:import namespace="http://www.ttuu.ee/idu0080/hinnakiri/1.0"
      schemaLocation="http://localhost:8080/yl2/services/HinnakiriPort?xsd=hinnakiri.xsd" />
    </xsd:schema>
  </types>
- <message name="getHinnakiriResponse">
  <part element="ns:getHinnakiriResponse" name="parameters" />
</message>
<message name="NewMessage" />
```

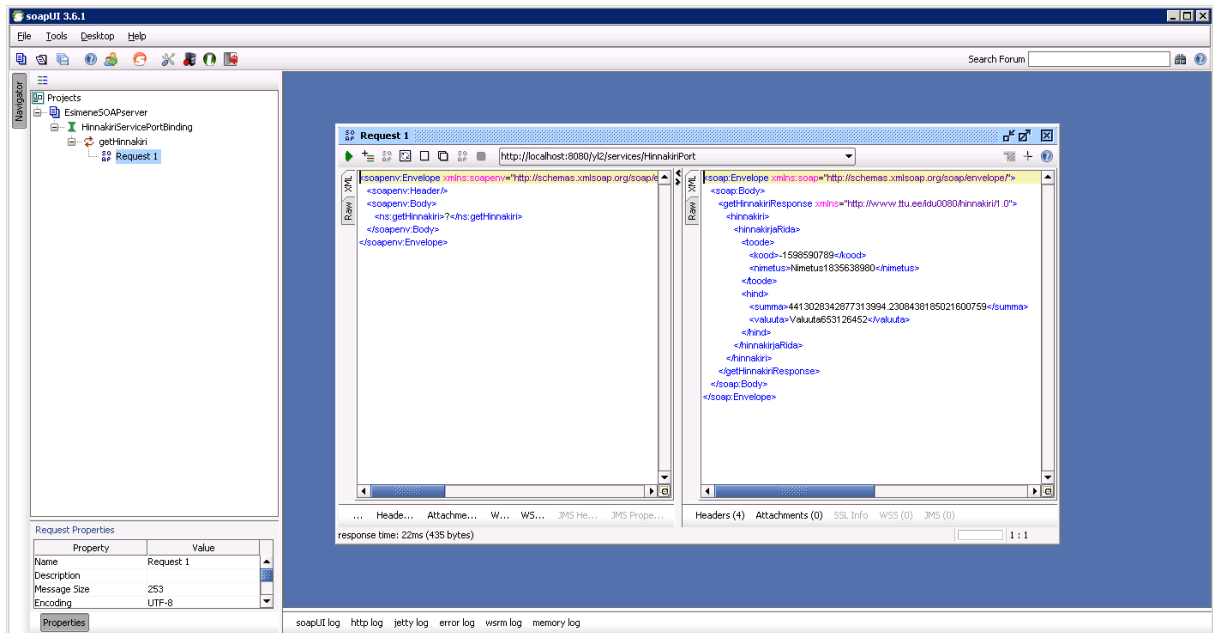
#### 4. SoapUI-ga testimine.

Avame SoapUI ja teeme uue projekti, kasutame wsdl-i aadressi ülevalt.



Avame SOAP-request-i projekti all ja käivitame:

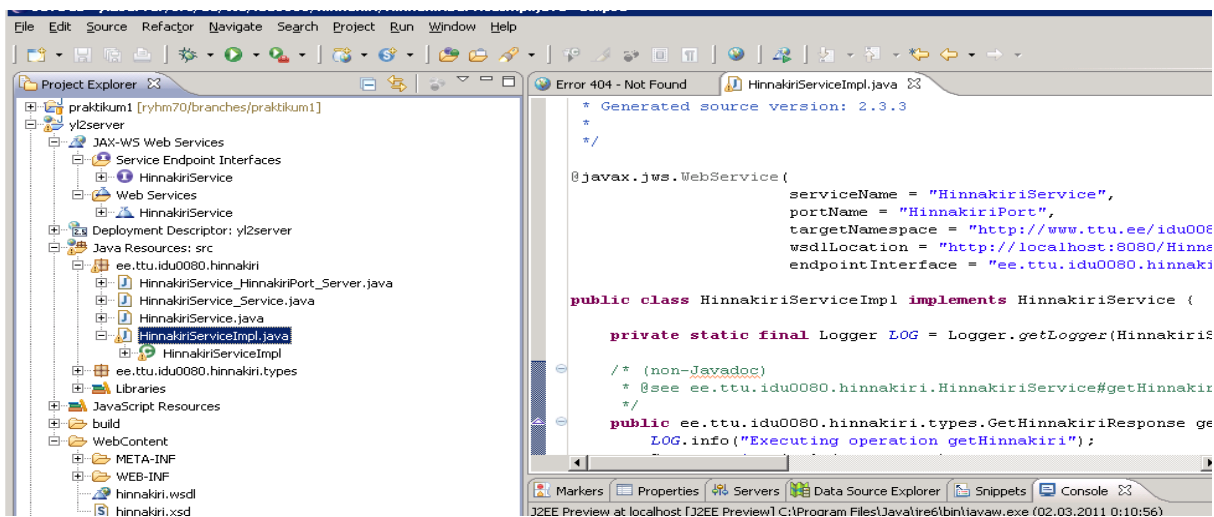




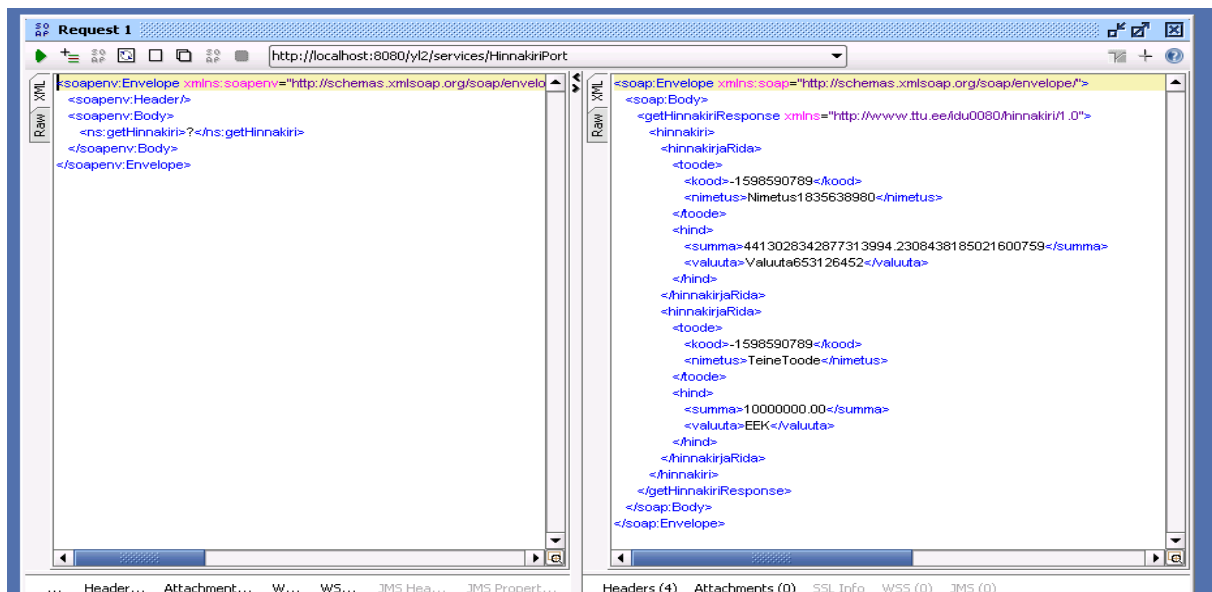
5. Täiendame serveri osa koodi.

Loeme ülesande 2 püstitust ja muudame koodi nii et tulemuseks saaksime 2 toodet.

Redigeerima peaks faili `HinnaKiriServiceImpl.java`



Tulemus võiks olla umbes selline:



## 6. Uue loodud Java projekti SVN-i lisamine.

Veenduge, et olete eelnevalt avanud SVN perspektiivi.

Võtke lahti Java EE perspektiiv ja valige oma Java project (serveri ja kliendi projektid hoidke eraldi projektidena), hiire parema nupu alt Team – Share Project, valige SVN, valige Repository location. Projekti kaust tekitage algselt /ryhm00/branches/ alla.

Kui kogemata määrasite vale SVN kausta asukoha, siis tekitage SVN-is uus kaust õigesse kohta (New - New remote folder), seejärel Java aknas valige oma projekt ja tehke Team – Switch to another branch/tag ja valige uus tekitatud kaust, seejärel tehke oma projektile Commit. Kui olete veendunud, et lokaalne koopia on edukalt õige SVN kausta külge lingitud ja projekti failid ka SVN-i tõstetud, siis võite vale SVN-i kausta ära kustutada.

Tehke branches alla Commititud versioonist Checkout ja kontrollige kas teie projekt on valmis ja töötab korralikult. Kui töötab korralikult, siis võite /branches/ all oleva projekti Tagida.

## 7. Java klient.

Uurime juhendmaterjalidest kuidas sellele SOAP-teenusele kirjutada Javas klienti. Ja proovime seda teha. Teeme sellesk eraldi Java projekti (ka „Dynamic Web project” tüüpi).