## Faculty of Science and Technology

### Module code and title: 5COSC004W-Client Service Architecture Tutorial Manual

Tutorial title

Getting acquainted with the concepts and procedures of the Coding Exercise Two (Building a real Client/Server from a Dummy Client)

Tutorial type Guided and indepenent and non-marked

Week 07 14/03/19

#### Contents

Learning Goals	1
TASKS to be Performed under the instruction of the Tutor (from Task 1 to Task 10)	1
TASKS to BE PERFORMED Independently be the student (from Task 11 to Task (Formative Assessment)	•

#### **Learning Goals**

This tutorial focuses on two main learning goals:

 to get acquainted with the the concepts of the second Coding Exercise (create a client/server from a dummy client)

It is divided into two separate sections, the student will perform the first task (1-X) following the instructions of the tutor, and then, will complete the other tasks independently.

# TASKS to be Performed under the instruction of the Tutor (from Task 1 to Task 10)

- Start Netbeans in your system. If Netbeans is not present in your system, use AppsAnywhere to launch it:
   (https://www.westminster.ac.uk/sites/default/public-files/general-documents/Using%20AppsAnywhere.pdf
- 2) Given the dummy client in Figure 1, you can observe the usual testing connection test. The dummy client is connected to a dummy server that exposes two methods:

#### Faculty of Science and Technology

- a. Boolean isConnected() that returns true if the server is running, and,
- b. String sendMessage(String arg) that returns a String which contains arg.

Figure 1, Dummy Client

- 3) Now, you have to create a REAL server that implements these functions and a real client to test it.
- 4) Create a Web Application to host the Web Service. (Figure 2 and Figure 3)

Faculty of Science and Technology

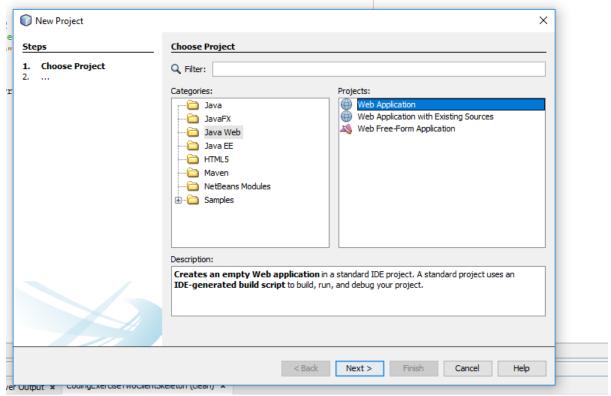


Figure 2, Create Web Application (A)

```
— ⊕ CodingExerciseTwoServer
                                     <!DOCTYPE html>
  ⊕ 🕞 Web Pages
   WEB-INF index.html
                                     To change this license header, choose License Headers in Project Properties.
                                    To change this template file, choose Tools | Templates

    Source Packages

                                5
                                    and open the template in the editor.
  ⊕ Configuration Files
                                  - <html>
                                7
                                8
                                9
                                           <title>TODO supply a title</title>
                               10
                                            <meta charset="UTF-8">
                                           <meta name="viewport" content="width=device-width, initial-scale=1.0">
                               11
                                        </head>
                               12
                               13
                                        <body>
                               14
                                            <div>TODO write content</div>
                                        </body>
                               15
                               16
                                     </html>
                               17
```

Figure 3, Create Web Application (B)

#### Faculty of Science and Technology

#### 5) Create a server package in the source code. Figure 4

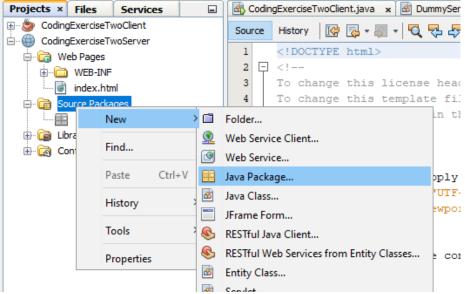


Figure 4, Create server package

#### 6) Create a Web Service. (Figure 5 and Figure 6)

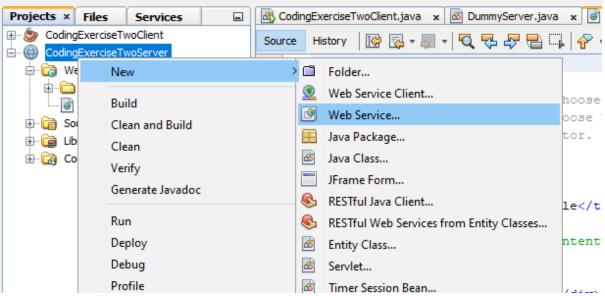


Figure 5, Create Web Service (A)

Faculty of Science and Technology

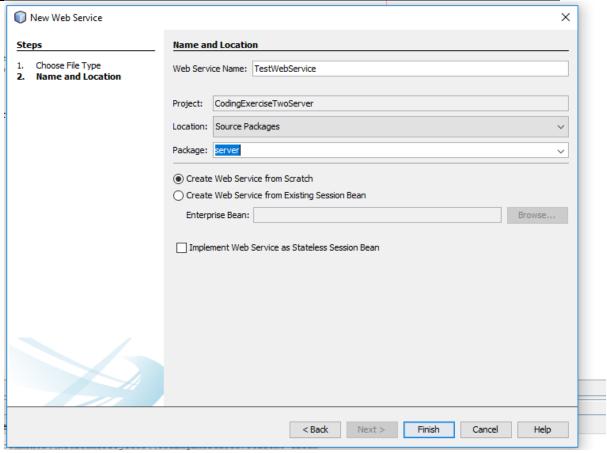


Figure 6, Create Web Service (B)

7) Add the two methods to the Web Service

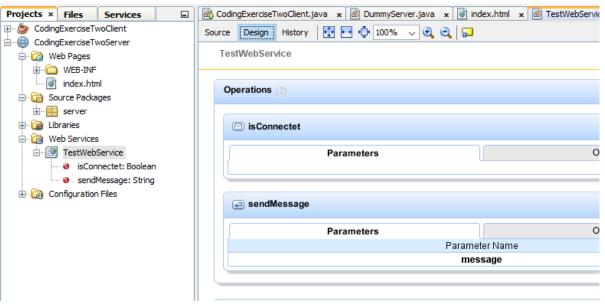


Figure 7, Adding Methods to the Web Service

8) Add the logic (Figure 9) into the empty method skeletons created by NetBeans (Figure 8)

Faculty of Science and Technology

```
IMPOIL Javan. Jws. webraiam,
12
   - /**
13
       * @author Gab
14
15
16
      @WebService(serviceName = "TestWebService")
17
      public class TestWebService {
18
19
20
          * Web service operation
21
          @WebMethod(operationName = "isConnectet")
22
<u>Q.</u>
          public Boolean isConnectet() {
   24
             //TODO write your implementation code here:
25
              return null;
26
27
          /**
28
   * Web service operation
29
          */
30
31
          @WebMethod(operationName = "sendMessage")
<u>Q</u>
   public String sendMessage(@WebParam(name = "message") String message) {
33
              //TODO write your implementation code here:
34
              return null;
35
36
37
```

Figure 8, Empty Methods Skeletons created by Netbeans

```
□ CodingExerciseTwoClient.java x DummyServer.java x index.html x inde
 Projects × Files Services
⊕ ... Some CodingExerciseTwoClient
                                                                                                  Source Design History 🔯 🐉 - 🔊 - 🔍 🔁 👺 🖶 📮 🎧 - 🚱 😂 😂 😂 🎒 🕌 🚅
IMPOIL Javan. Jws. webraiam,
      ⇒ 🕞 Web Pages
             WEB-INF index.html
                                                                                                  Source Packages
             server
                         TestWebService.java
                                                                                                                @WebService(serviceName = "TestWebService")
                                                                                                  16
      in libraries
                                                                                                   17
                                                                                                                  public class TestWebService {
      Web Services
                                                                                                   18
             - TestWebService
                                                                                                    19 🖃
                           isConnectet: Boolean
                                                                                                                                 * Web service operation

    ø sendMessage: String

      ⊕ 🕞 Configuration Files
                                                                                                                                @WebMethod(operationName = "isConnectet")
                                                                                                     ₩ 📮
                                                                                                                                public Boolean isConnectet() {
                                                                                                    24
                                                                                                                                           System.out.println("[SERVER] - Server is connected");
                                                                                                   25
                                                                                                                                            return true:
                                                                                                   26
                                                                                                   27
                                                                                                    28
                                                                                                                                   * Web service operation
                                                                                                    30
                                                                                                    31
                                                                                                                                 @WebMethod(operationName = "sendMessage")
                                                                                                     ₩ 📮
                                                                                                                                 public String sendMessage(@WebParam(name = "message") String message) {
                                                                                                                                            System.out.println("[SERVER] - Server has received message: " + message);
                                                                                                    33
                                                                                                                                            return "Server has received" + message;
                                                                                                    34
                                                                                                    35
                                                                                                    36
                                                                                                    37
```

Figure 9, Adding logic to the empty methods.

9) Deploy (Figure 10) and test the Web Service (Figure 11 and Figure 12).

Faculty of Science and Technology

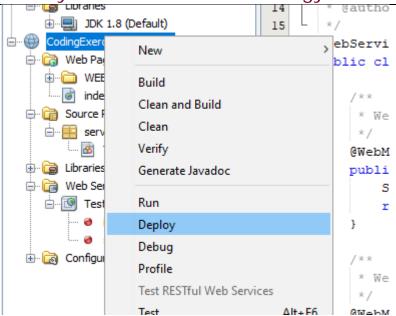


Figure 10, Deploy the Web Service

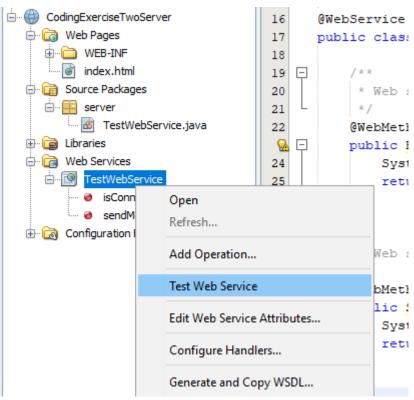


Figure 11, Test the Web Service (A)

Faculty of Science and Technology

#### TestWebService Web Service Tester

This form will allow you to test your web service implementation (WSDL File)
To invoke an operation, fill the method parameter(s) input boxes and click on the button labeled with the method name.
Methods:
public abstract java.lang.String server.TestWebService.sendMessage(java.lang.String) sendMessage (hello)
public abstract java.lang.Boolean server.TestWebService.isConnectet()  isConnectet ()

Figure 12, Test the Web Service (B)

10) Create a Java Client for the web service you have just created (The steps are detailed in Tutorial 7.A

# TASKS to BE PERFORMED Independently be the student (from Task 11 to Task 14) (Formative Assessment)

- 11) Modify the testConnection method so that the client can send its id (as a string) and that is returned from the method (e.g. Connection from client ..... succeded)
- 12) Modify the Server Class so that the server has a name (as a stgring) so that the returned string from the method testConnection is (e.g. Server....: Connection from client ..... succeded)
- 13) Modify the Server Class so that the server can add a time stamp to method testConnection (e.g. [Date and Time] Server....: Connection from client ..... succeded)
- 14) Modify the Server and Client class so that the server raises an exception in the method sendMessage that is thrown when the argument is null.