



Lab 12: GIT + GoogleMaps

Introduction to Web Programming

AA 2016/2017

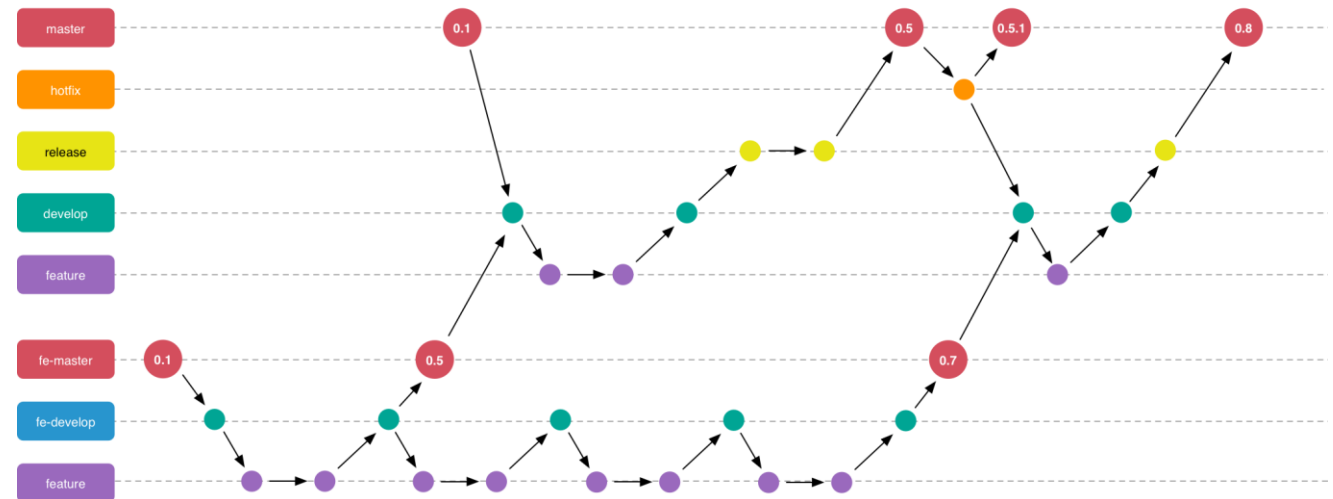
Stefano Chirico

Git: Introduction

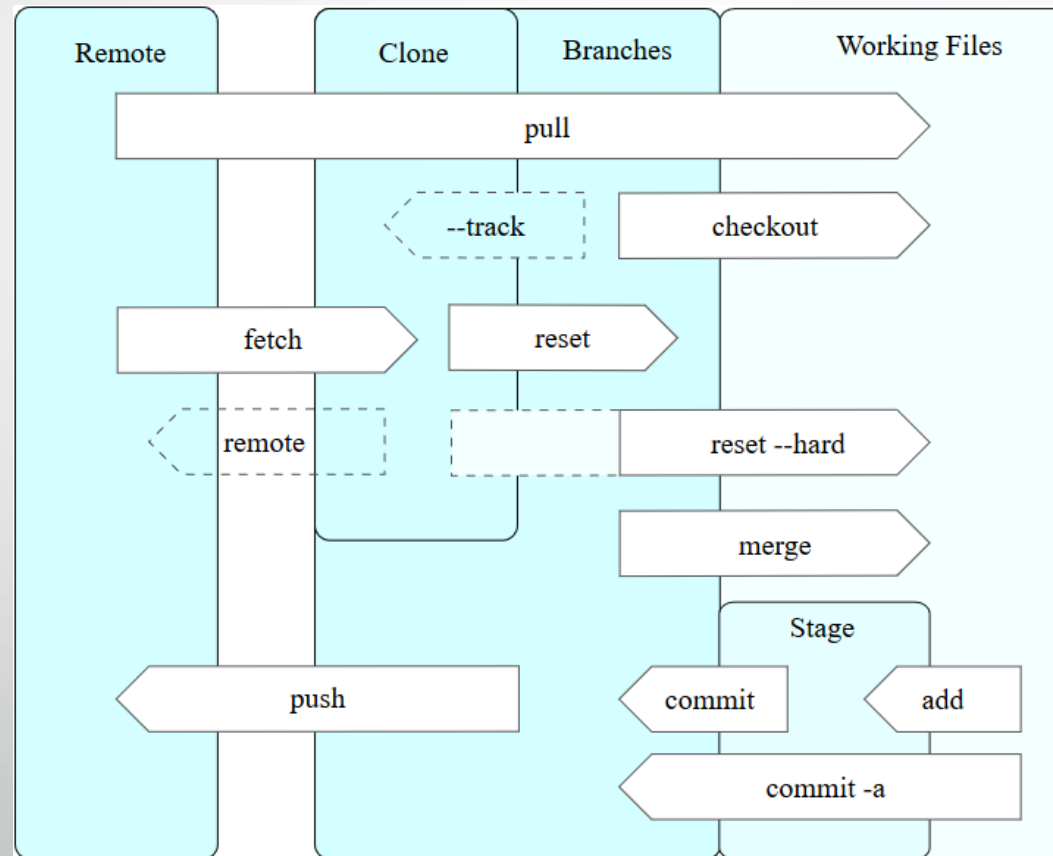
- **GitHub** is a web-based Git repository hosting service. It offers all of the distributed revision control and **source code management** (SCM) functionality of Git as well as adding its own features. Unlike Git, which is strictly a command-line tool, GitHub provides a Web-based graphical interface and desktop as well as mobile integration. It also provides access control and several collaboration features such as bug tracking, feature requests, task management, and wikis for every project.
- **Git** is a **version control system** that is widely used for software development and other version control tasks. It is a distributed revision control system with an emphasis on speed, data integrity, and support for distributed, non-linear workflows. Git was created by Linus Torvalds in 2005 for development of the Linux kernel, with other kernel developers contributing to its initial development.

Git: Introduction

Branching



Git: How it works



GitHub

Create account

- <https://github.com/join>

Join GitHub

The best way to design, build, and ship software.



Step 1:
Set up a personal account



Step 2:
Choose your plan



Step 3:
Help us tailor your experience

Create your personal account

Username

This will be your username — you can enter your organization's username next.

Email Address

You will occasionally receive account related emails. We promise not to share your email with anyone.

Password

Use at least one lowercase letter, one numeral, and seven characters.

By clicking on "Create an account" below, you are agreeing to the [Terms of Service](#) and the [Privacy Policy](#).

Create an account

You'll love GitHub

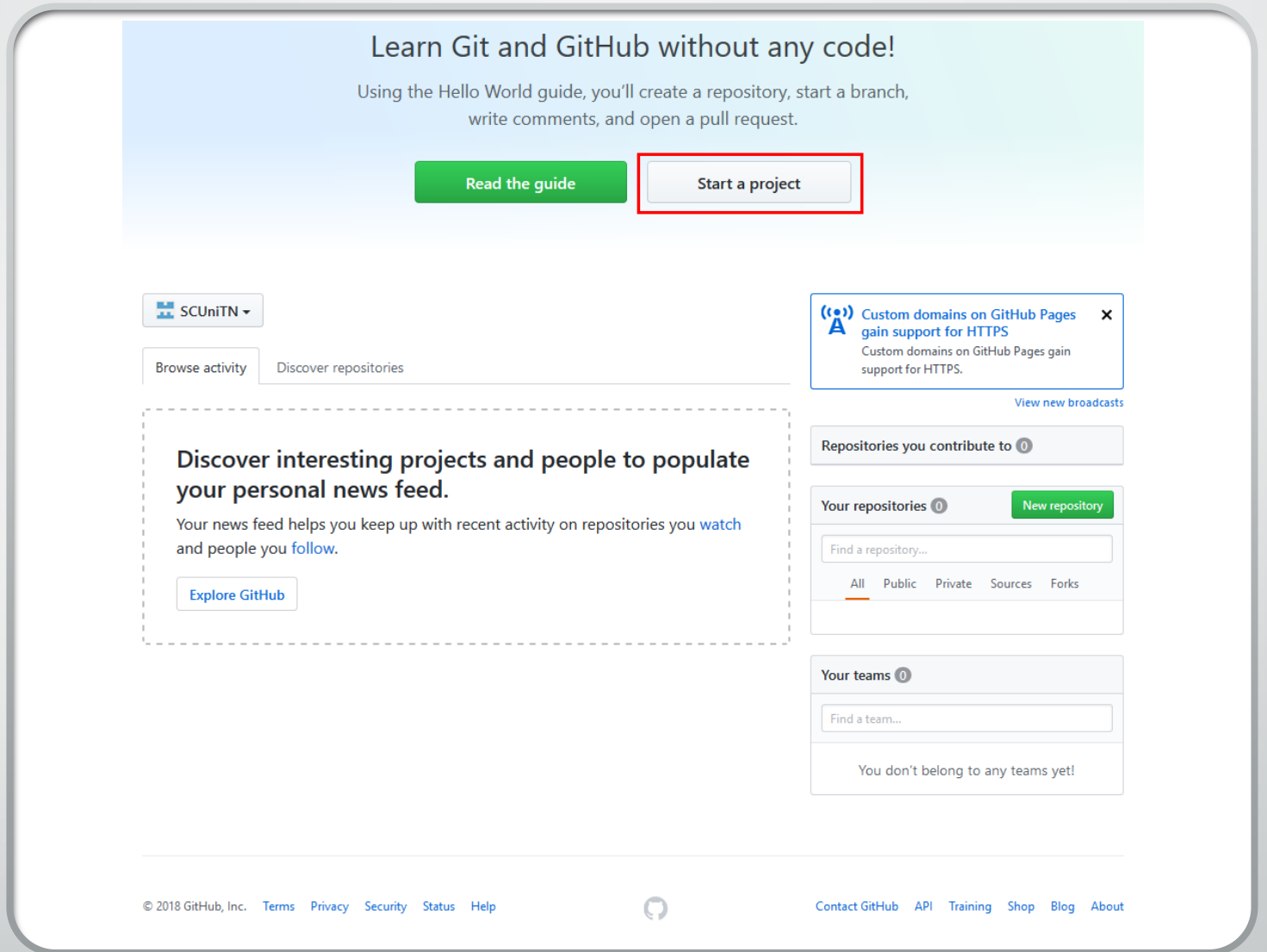
Unlimited collaborators

Unlimited public repositories

- ✓ Great communication
- ✓ Friction-less development
- ✓ Open source community

GitHub

Start a project



GitHub

Create a new repository

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

Repository name

SCUnitN /

Great repository names are short and memorable. Need inspiration? How about [legendary-octo-giggle](#).

Description (optional)

☒ **Public**
Anyone can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.


☐ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None**

Add a license: **None** [i](#)

Create repository

© 2018 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Help](#)



[Contact GitHub](#) [API](#) [Training](#) [Shop](#) [Blog](#) [About](#)

GitHub

Create a new repository


1. Digit the repository name
2. Check to initialize the repository with a README file
3. Click "Create repository" button

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

Repository name

SCUniTN / Lab12 

Great repository names are short and memorable. Need inspiration? How about [refactored-meme](#).

Description (optional)

☒ Public

Anyone can see this repository. You choose who can commit.

☐ Private

You choose who can see and commit to this repository.


2 ☒

Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None

Add a license: None



3

Create repository

© 2018

© 2018 GitHub, Inc.


[Terms](#)

[Privacy](#)

[Security](#)

[Status](#)

[Help](#)



[Contact GitHub](#)

[API](#)

[Training](#)

[Shop](#)

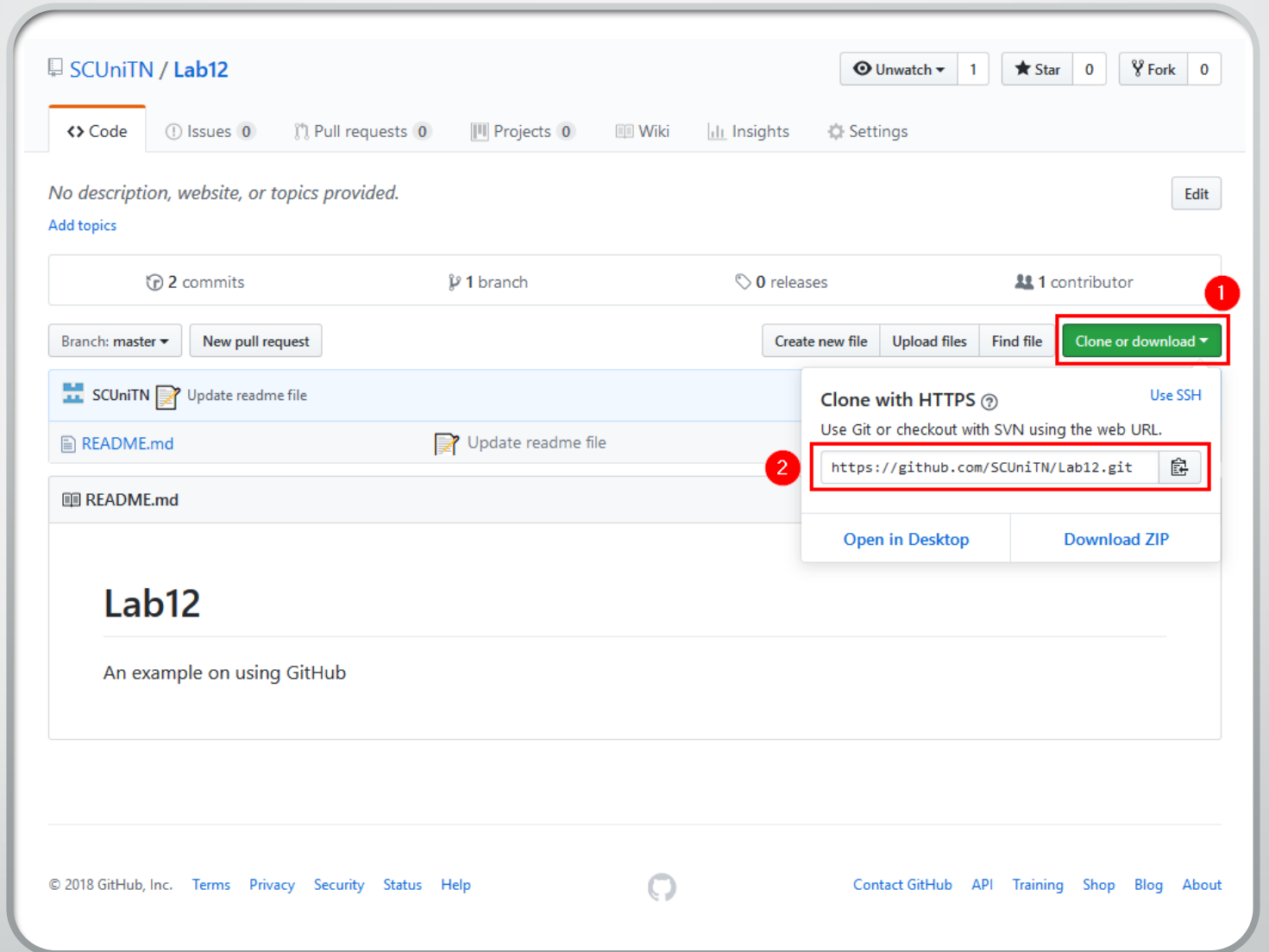
[Blog](#)

[About](#)

GitHub

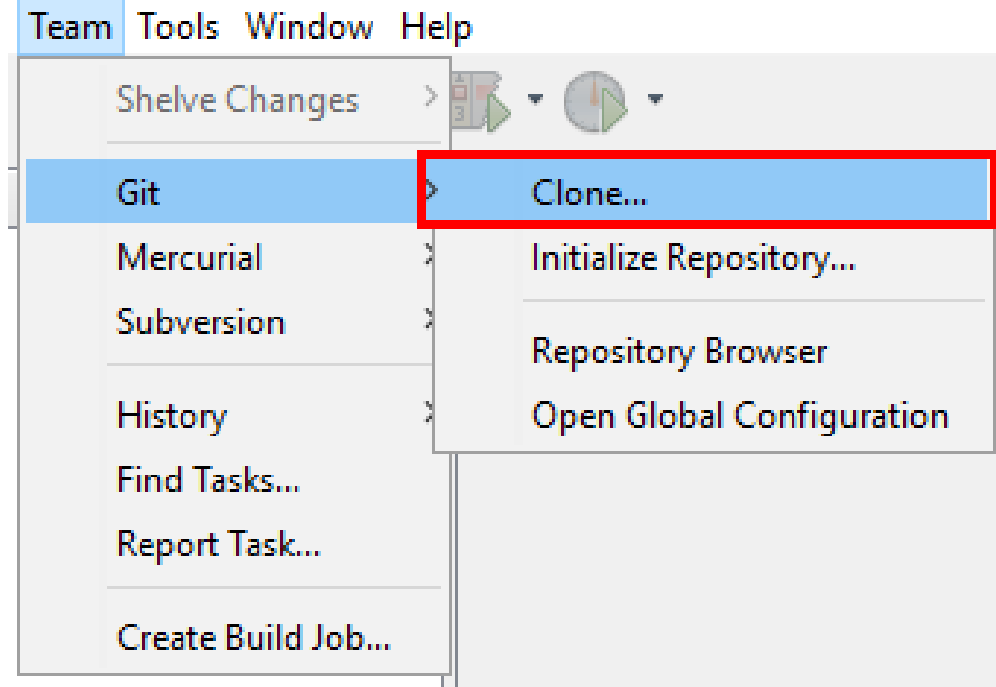
Clone repository

1. Click "Clone or download"
2. Click the button to copy the project address to the clipboard



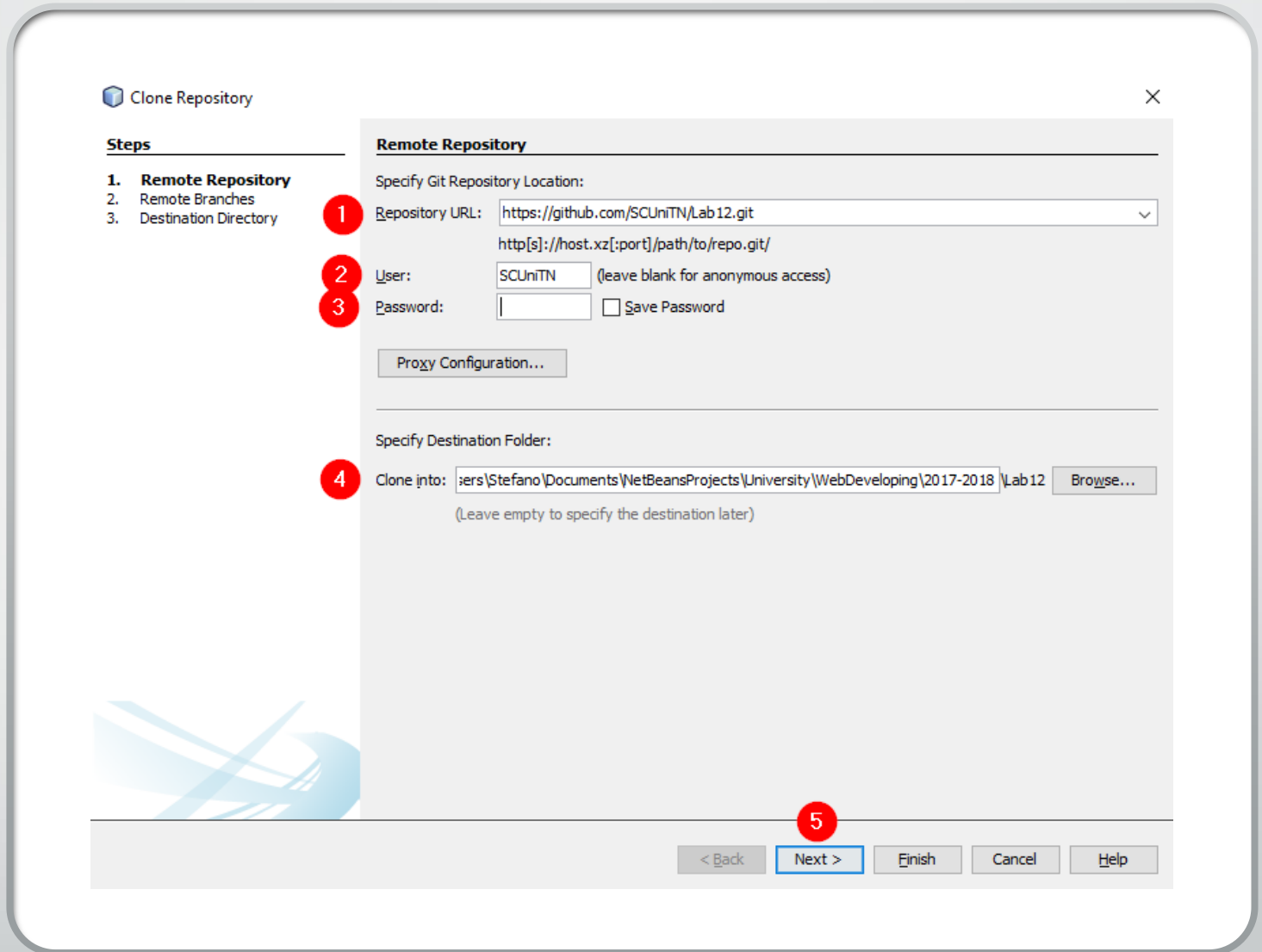
NetBeans

Clone repository



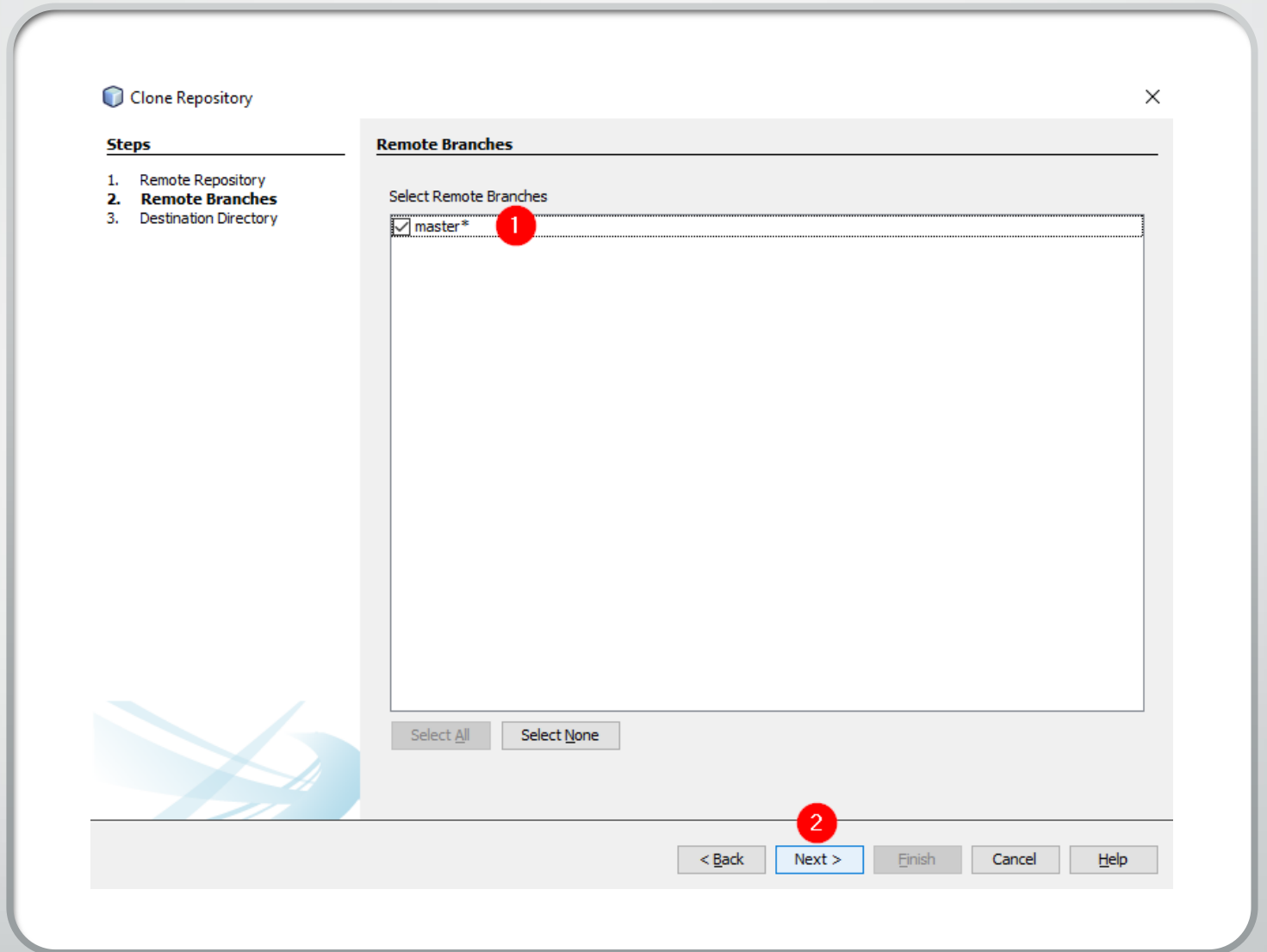
NetBeans

Clone repository



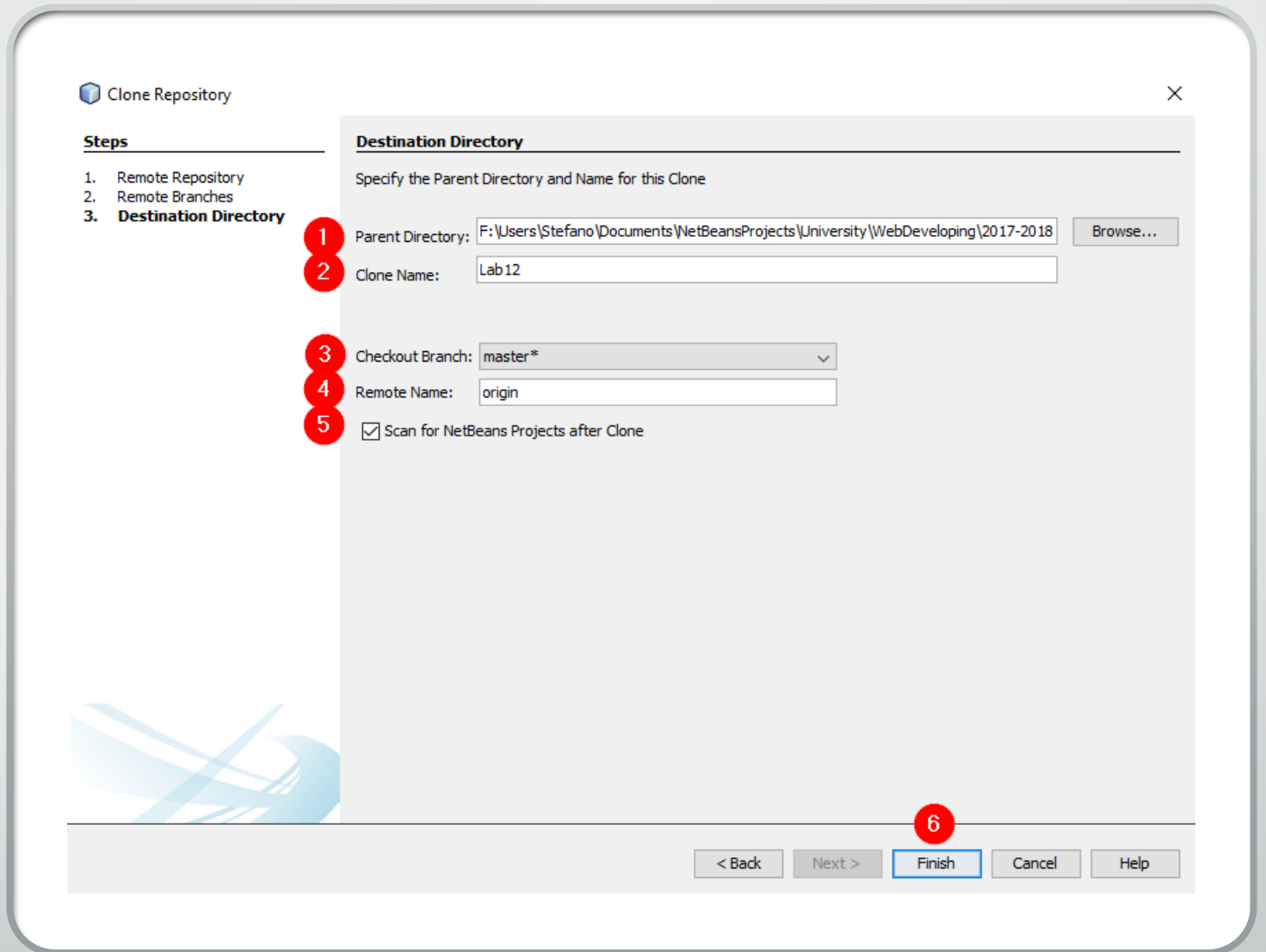
NetBeans

Remote branches



NetBeans

Destination directory



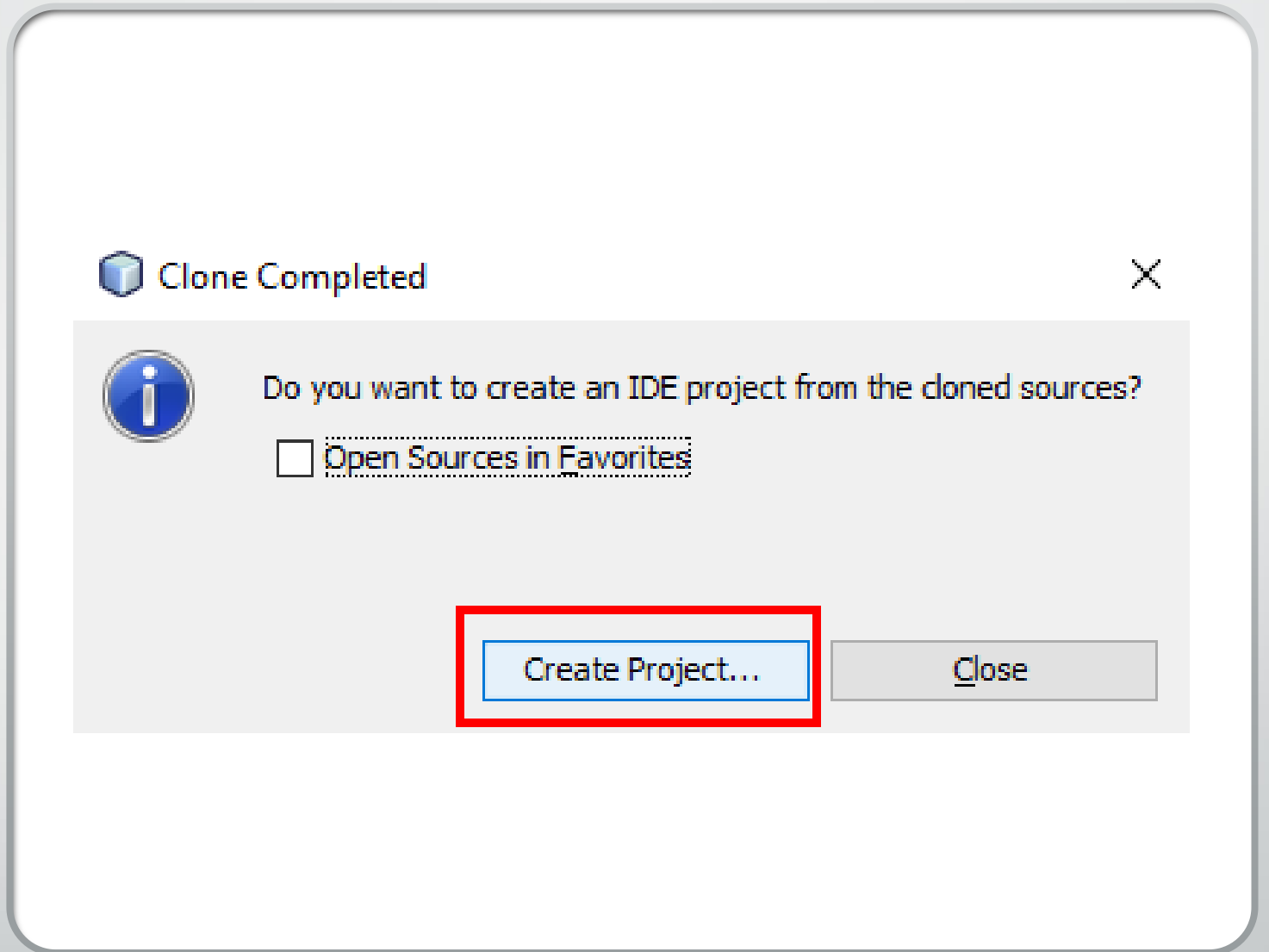
The image shows the 'Clone Repository' dialog box in NetBeans. The 'Steps' list on the left indicates the current step is '3. Destination Directory'. The main area is titled 'Destination Directory' and contains the following fields and options:

- Parent Directory:** A text field containing the path 'F:\Users\Stefano\Documents\NetBeansProjects\University\WebDeveloping\2017-2018'. A red circle with the number '1' is next to this field. A 'Browse...' button is to the right.
- Clone Name:** A text field containing 'Lab12'. A red circle with the number '2' is next to this field.
- Checkout Branch:** A dropdown menu showing 'master*'. A red circle with the number '3' is next to this field.
- Remote Name:** A text field containing 'origin'. A red circle with the number '4' is next to this field.
- Scan for NetBeans Projects after Clone:** A checked checkbox. A red circle with the number '5' is next to this checkbox.

At the bottom right, there are four buttons: '< Back', 'Next >', 'Finish', and 'Help'. A red circle with the number '6' is positioned above the 'Finish' button.

NetBeans

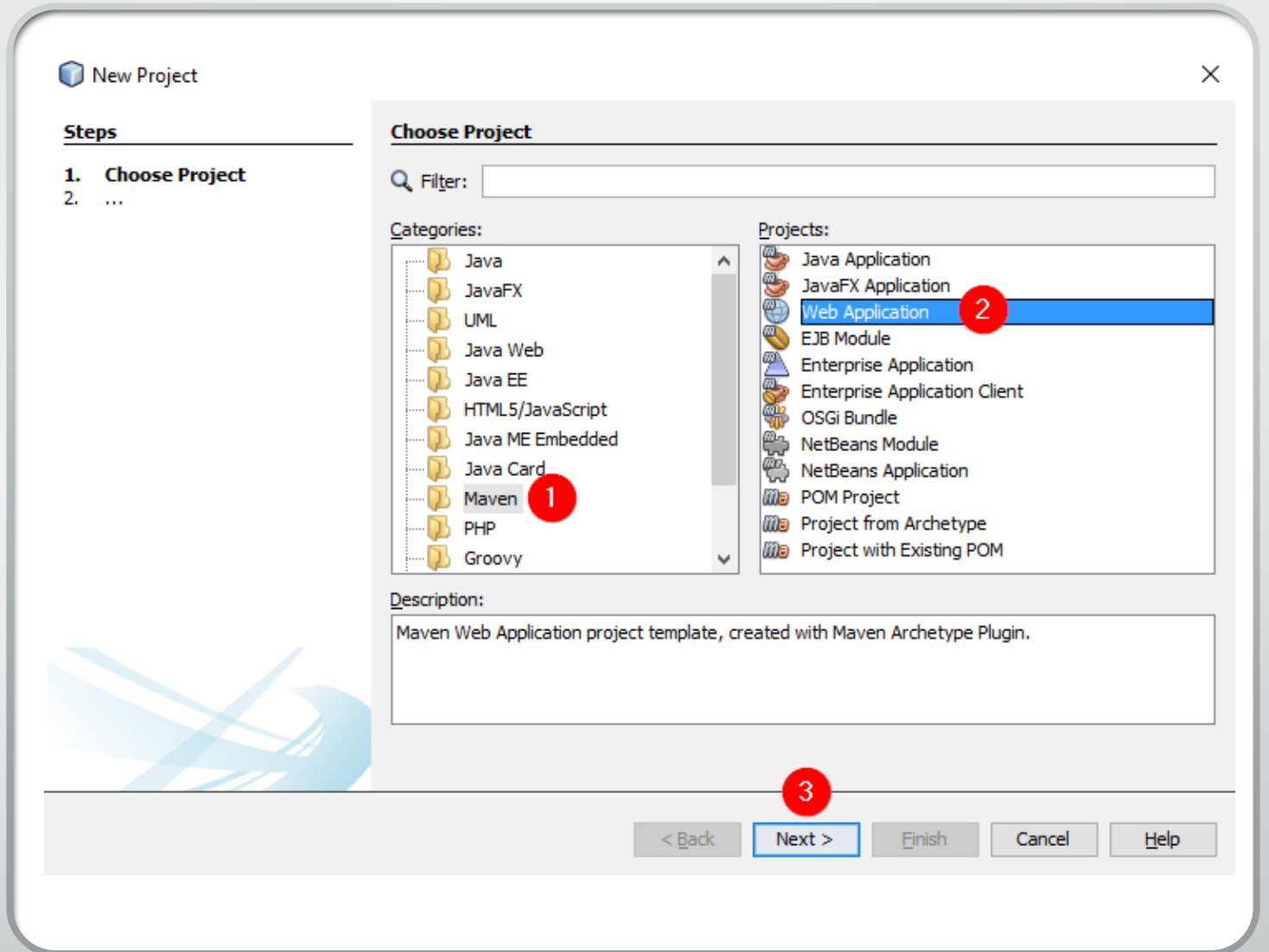
Clone Completed



NetBeans

New Project

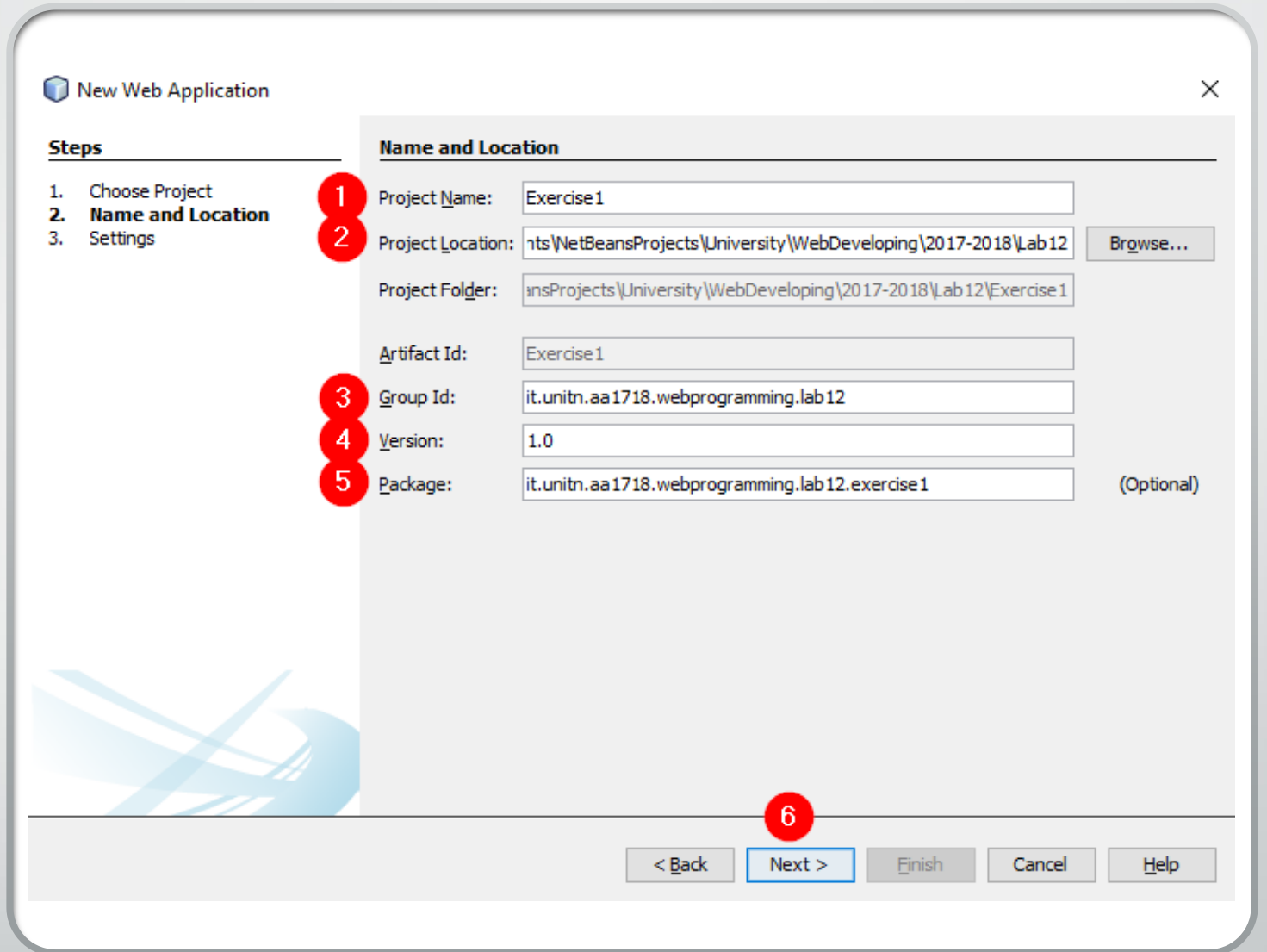
- Choose Project



NetBeans

New Web Application

- Name and Location



The image shows the 'New Web Application' dialog box in NetBeans. The dialog is titled 'New Web Application' and has a close button (X) in the top right corner. It is divided into two main sections: 'Steps' and 'Name and Location'.

Steps:

1. Choose Project
- 2. Name and Location**
3. Settings

Name and Location:

1 Project Name: Exercise 1

2 Project Location: nts\NetBeansProjects\University\WebDeveloping\2017-2018\Lab 12 Browse...

Project Folder: nsProjects\University\WebDeveloping\2017-2018\Lab 12\Exercise 1

Artifact Id: Exercise 1

3 Group Id: it.unitn.aa1718.webprogramming.lab12

4 Version: 1.0

5 Package: it.unitn.aa1718.webprogramming.lab12.exercise1 (Optional)

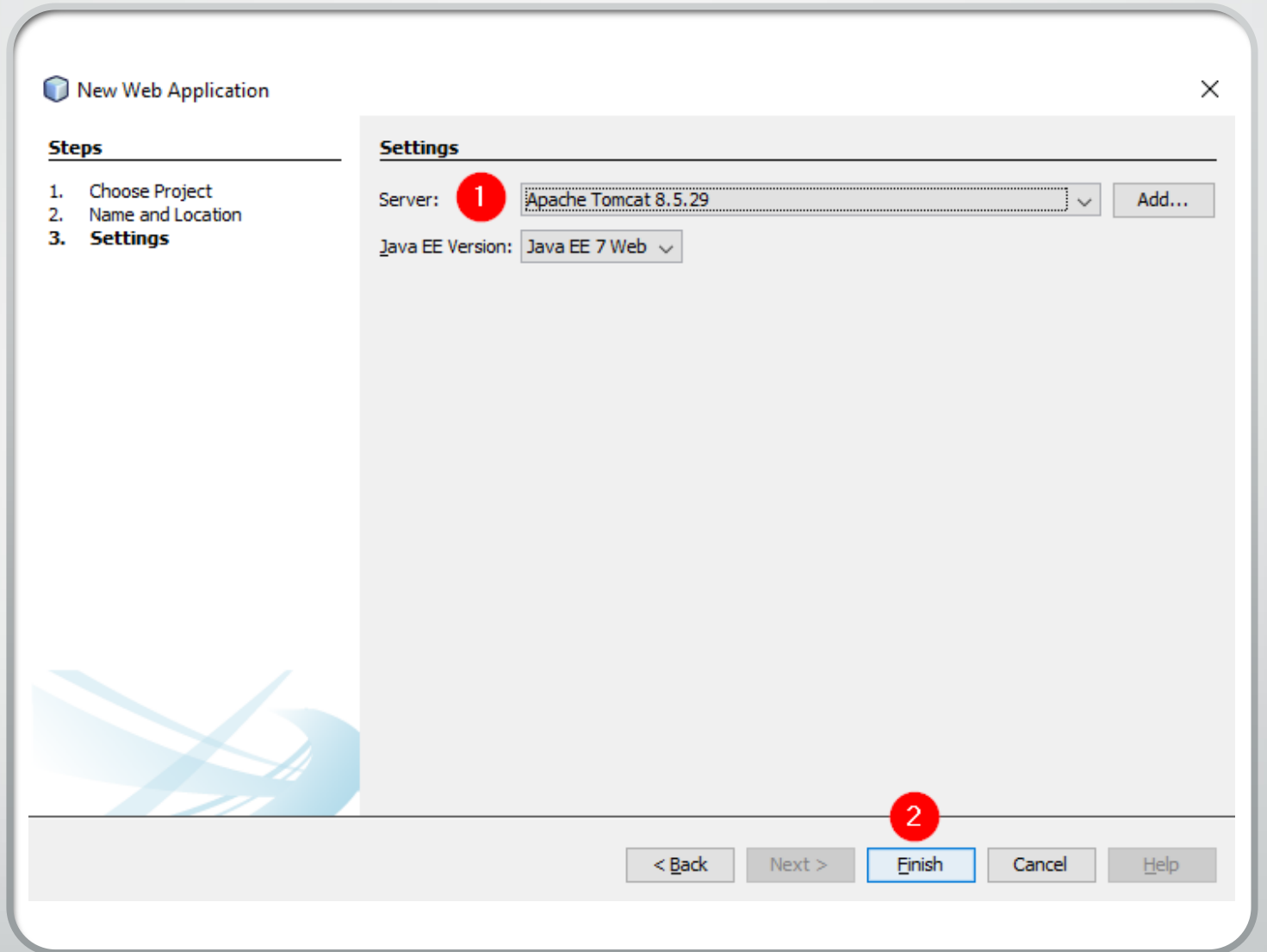
6

< Back Next > Finish Cancel Help

NetBeans

New Web Application

- Settings



Web Service SOAP

- **SOAP**, originally an acronym for **Simple Object Access Protocol**, is a protocol specification for exchanging structured information in the implementation of web services in computer networks. It uses **XML** Information Set for its message format, and relies on application layer protocols, most notably Hypertext Transfer Protocol (**HTTP**) or Simple Mail Transfer Protocol (**SMTP**), for **message negotiation and transmission**.

Web Service SOAP

- **Advantages**

- SOAP's neutrality characteristic explicitly makes it suitable for use with any transport protocol. Implementations often use HTTP as a transport protocol, but obviously other popular transport protocols can be used. For example, SOAP can also be used over SMTP, JMS and Message Queues.
- SOAP, when combined with HTTP post/response exchanges, tunnels easily through existing firewalls and proxies, and consequently doesn't require modifying the widespread computing and communication infrastructures that exist for processing HTTP post/response exchanges.

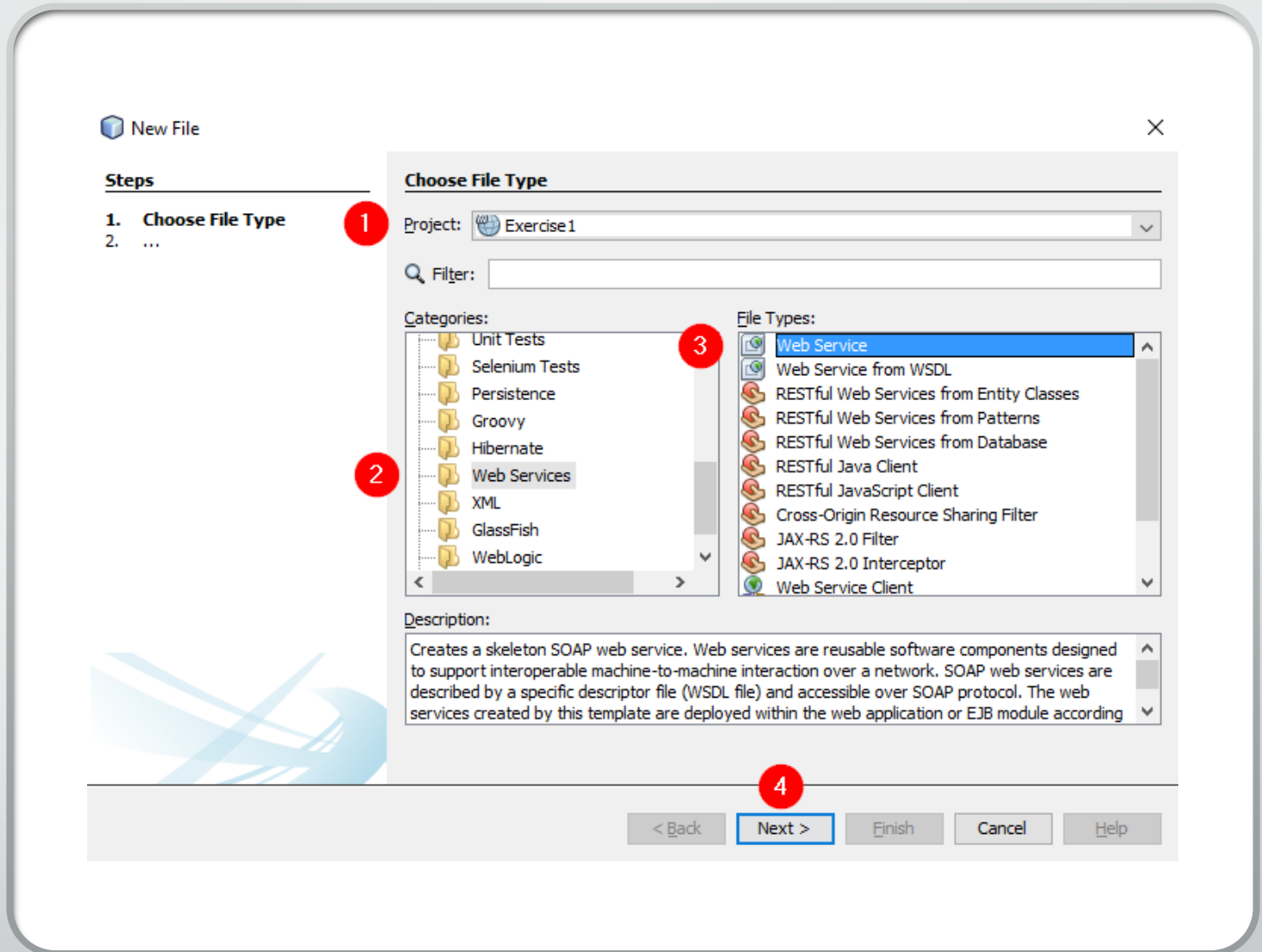
- **Disadvantages**

- When using standard implementations and the default SOAP/HTTP binding, the XML info-set is serialized as XML. To improve performance for the special case of XML with embedded binary objects, the Message Transmission Optimization Mechanism was introduced.
- When relying on HTTP as a transport protocol and not using WS-Addressing or an ESB, the roles of the interacting parties are fixed. Only one party (the client) can use the services of the other.
- The verbosity of the protocol led to the domination in the field by services using the HTTP protocol more directly.

NetBeans

New File

- Choose File Type



NetBeans

New Web Service

- Name and Location

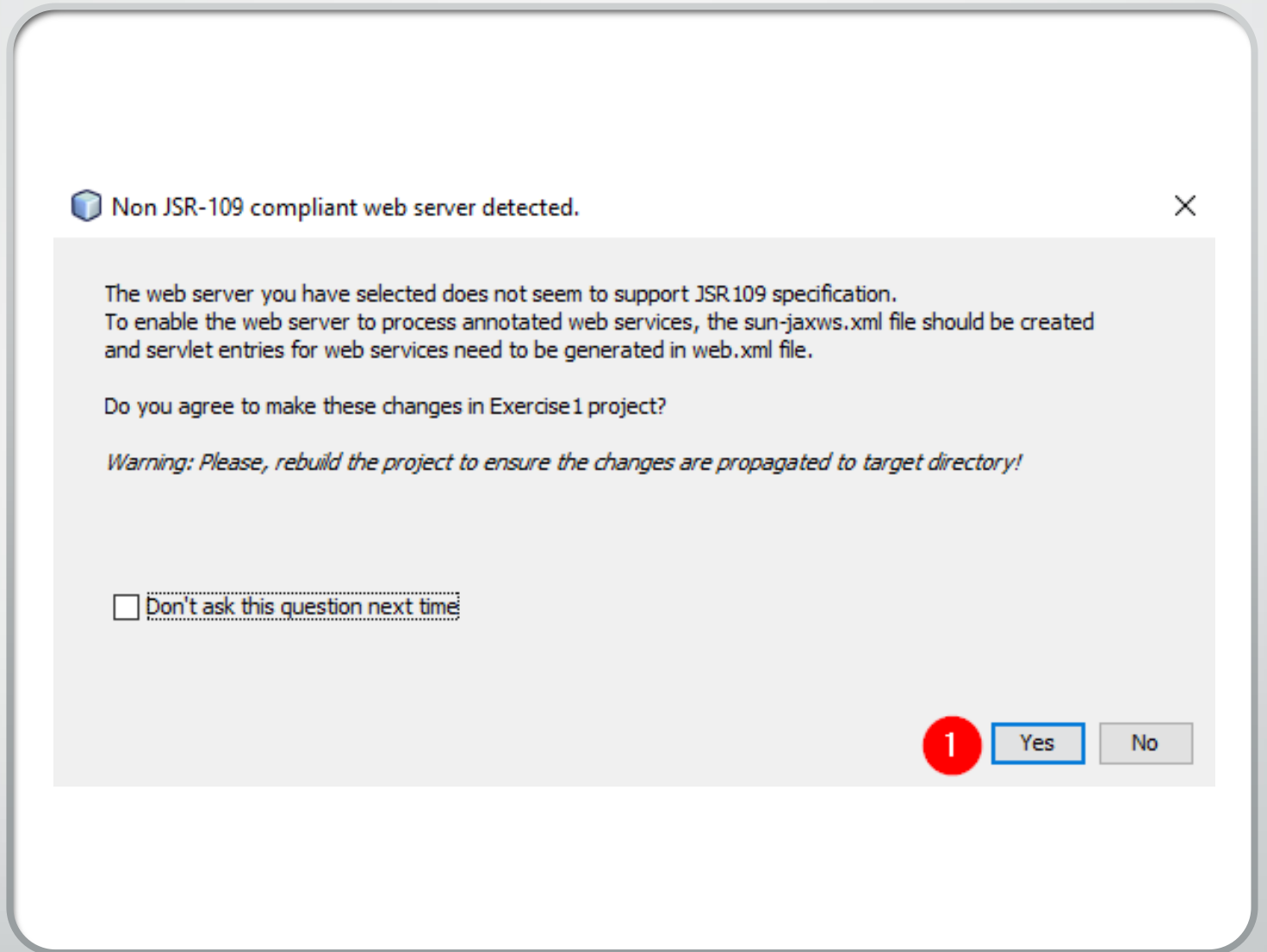
The screenshot shows the 'New Web Service' dialog box in NetBeans. The dialog is titled 'New Web Service' and has a close button (X) in the top right corner. On the left, under the 'Steps' section, there are two steps: '1. Choose File Type' and '2. Name and Location'. The 'Name and Location' section is active. It contains the following fields and options:

- Web Service Name:** A text field containing 'MapGroupGenerator', marked with a red circle '1'.
- Project:** A text field containing 'Exercise1'.
- Location:** A dropdown menu showing 'Source Packages'.
- Package:** A dropdown menu showing 'it.unitn.aa1718.webprogramming.lab12.exercise1.services', marked with a red circle '2'.
- Creation Method:** Two radio buttons. The first is 'Create Web Service from Scratch', which is selected and marked with a red circle '3'. The second is 'Create Web Service from Existing Session Bean'.
- Enterprise Bean:** A text field and a 'Browse...' button, located below the radio buttons.
- Implement Web Service as Stateless Session Bean:** A checkbox that is currently unchecked.

At the bottom of the dialog, there are four buttons: '< Back', 'Next >', 'Finish' (which is highlighted with a blue border and marked with a red circle '4'), 'Cancel', and 'Help'.

NetBeans

Create sun-jaxws.xml



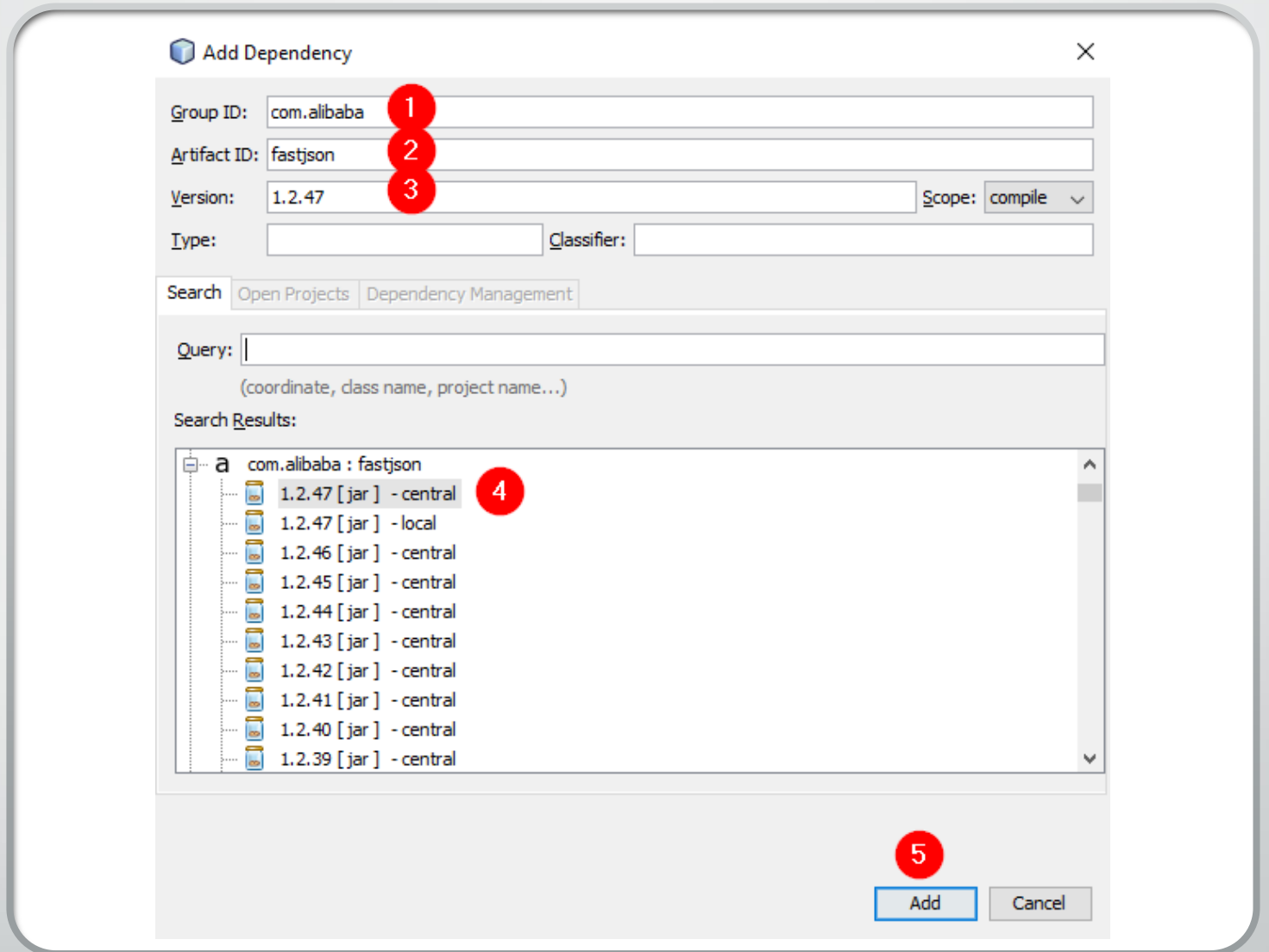
NetBeans

Add FastJson dependency

To parse the output using the JSON format, we need to add a library.

You can use the Gson library shown on the last lesson or you can use the fastjson library.

Add the Maven dependency.



NetBeans

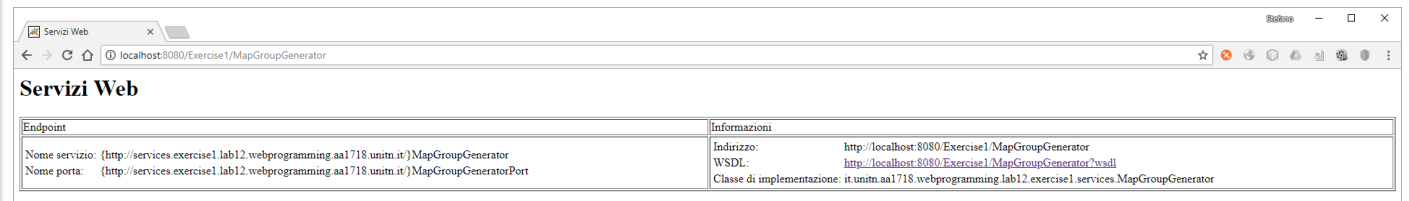
Coding...

```
@WebMethod(operationName = "getPublicURLMapNeighbors")
public String getPublicURLMapNeighbors(
    @WebParam(name = "userId") Integer userId,
    @WebParam(name = "shopIds") List<Integer> shopIds,
    @WebParam(name = "aggregationTypeId") Integer aggregationTypeId,
    @WebParam(name = "mapTypeId") Integer mapTypeId,
    @WebParam(name = "centerLat") Double centerLat,
    @WebParam(name = "centerLng") Double centerLng,
    @WebParam(name = "swLat") Double swLat,
    @WebParam(name = "swLng") Double swLng,
    @WebParam(name = "neLat") Double neLat,
    @WebParam(name = "neLng") Double neLng,
    @WebParam(name = "width") Integer divWidth,
    @WebParam(name = "height") Integer divHeight,
    @WebParam(name = "zoom") Integer zoom
) {
    List<Shop> shops = new ArrayList<>();

    String title = "Google Map: Shops neighbors";
```


NetBeans

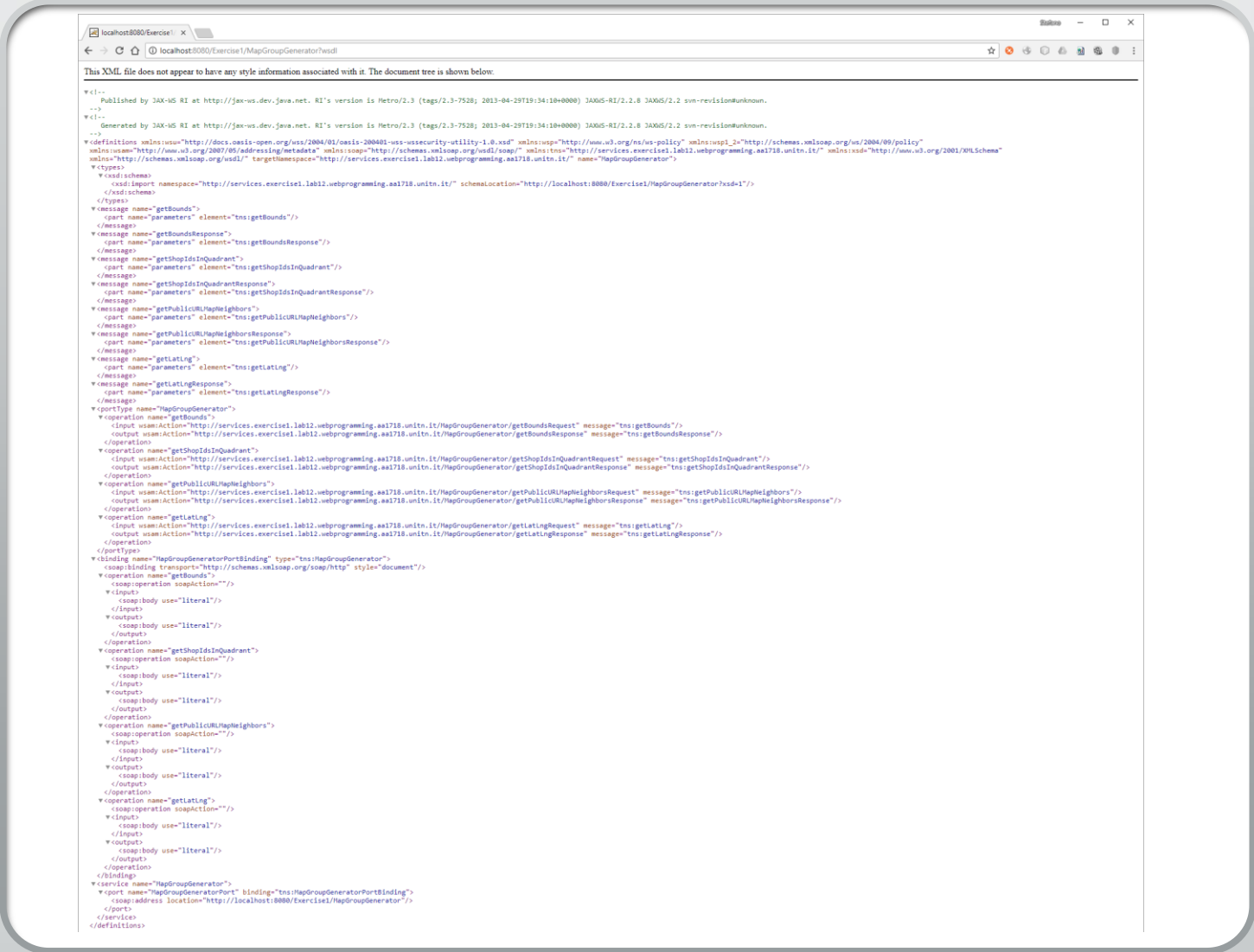
- **Get WSDL**
- **Web Service Description Language**



Endpoint	Informazioni
Nome servizio: {http://services.exercise1.lab12.webprogramming.aa1718.univr.it}MapGroupGenerator	Indirizzo: http://localhost:8080/Exercise1/MapGroupGenerator
Nome porta: {http://services.exercise1.lab12.webprogramming.aa1718.univr.it}MapGroupGeneratorPort	WSDL: http://localhost:8080/Exercise1/MapGroupGenerator?wsdl
	Classe di implementazione: it.univr.aa1718.webprogramming.lab12.exercise1.services.MapGroupGenerator

NetBeans

- Get WSDL
- Web Service Description Language



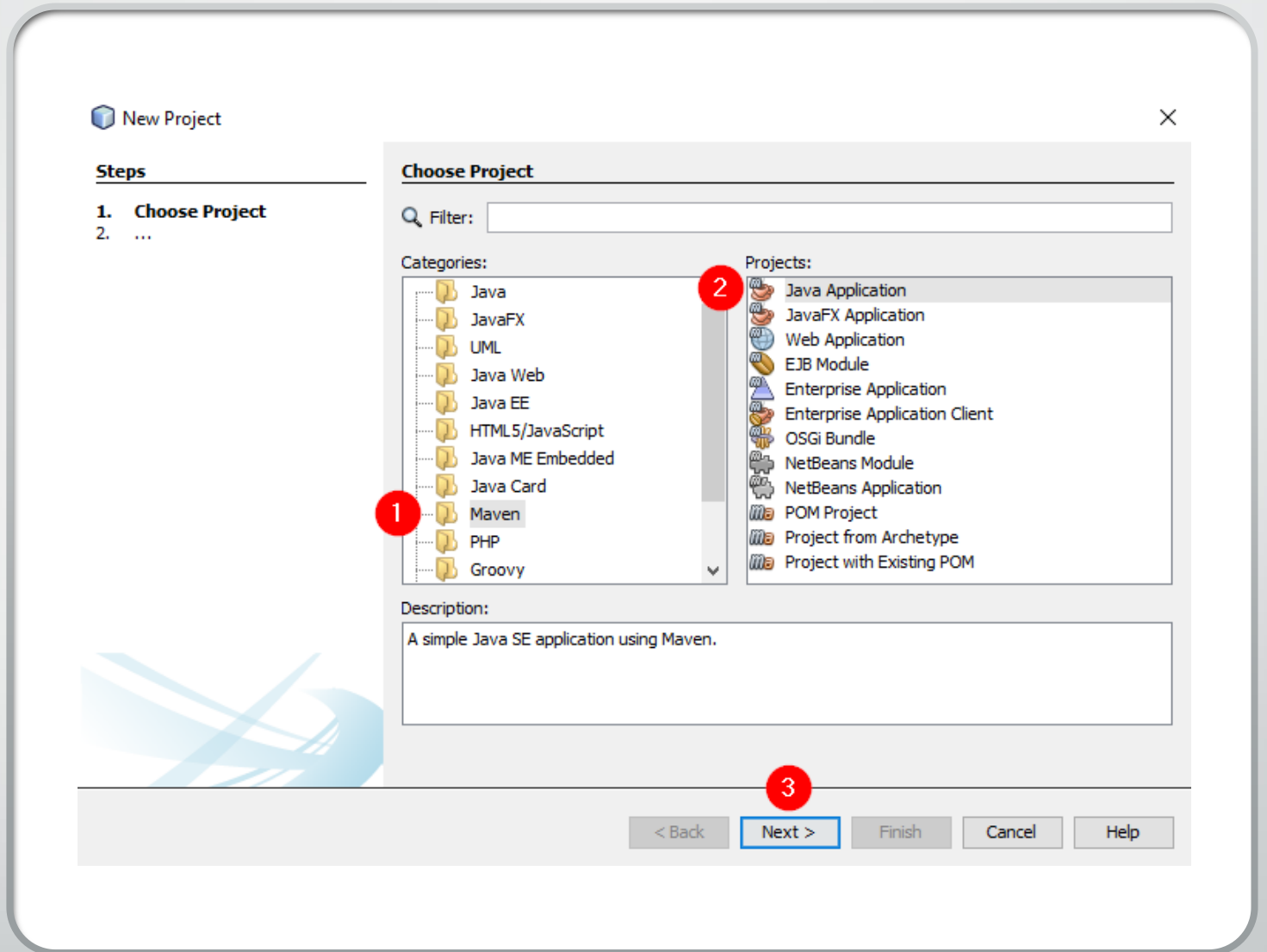
This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml-
  Published by JAX-WS RI at http://jax-ws.dev.java.net. RI's version is Metro/2.3 (tags/2.3-7528; 2013-04-29T19:34:10+0000) JAXWS-RI/2.2.8 JAXWS/2.2 svn-revision=unknown.
  --
  Generated by JAX-WS RI at http://jax-ws.dev.java.net. RI's version is Metro/2.3 (tags/2.3-7528; 2013-04-29T19:34:10+0000) JAXWS-RI/2.2.8 JAXWS/2.2 svn-revision=unknown.
  -->
  <definitions xmlns:wsa="http://docs.oasis-open.org/ws/2004/08/oasis-200408-wss-wssecurity-utility-1.0.xsd" xmlns:wsap="http://www.w3.org/ns/ws-policy" xmlns:wspl_2="http://schemas.xmlsoap.org/ws/2004/09/policy"
  xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://services.exercisel.lab12.webprogramming.aal718.unin.it/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://schemas.xmlsoap.org/wsdl/" targetNamespace="http://services.exercisel.lab12.webprogramming.aal718.unin.it/" name="MapGroupGenerator">
    <types>
      <xsd:schema
        xmlns:schema="http://services.exercisel.lab12.webprogramming.aal718.unin.it/" schemaLocation="http://localhost:8080/Exercise1/MapGroupGenerator?xsd=1"/>
      </xsd:schema>
    </types>
    <message name="getBounds">
      <part name="parameters" element="tns:getBounds"/>
    </message>
    <message name="getBoundsResponse">
      <part name="parameters" element="tns:getBoundsResponse"/>
    </message>
    <message name="getShopIdsInQuadrant">
      <part name="parameters" element="tns:getShopIdsInQuadrant"/>
    </message>
    <message name="getShopIdsInQuadrantResponse">
      <part name="parameters" element="tns:getShopIdsInQuadrantResponse"/>
    </message>
    <message name="getPublicURLMapNeighbors">
      <part name="parameters" element="tns:getPublicURLMapNeighbors"/>
    </message>
    <message name="getPublicURLMapNeighborsResponse">
      <part name="parameters" element="tns:getPublicURLMapNeighborsResponse"/>
    </message>
    <message name="getLattling">
      <part name="parameters" element="tns:getLattling"/>
    </message>
    <message name="getLattlingResponse">
      <part name="parameters" element="tns:getLattlingResponse"/>
    </message>
    <portType name="MapGroupGenerator">
      <operation name="getBounds">
        <input wsam:Action="http://services.exercisel.lab12.webprogramming.aal718.unin.it/MapGroupGenerator/getBoundsRequest" message="tns:getBounds"/>
        <output wsam:Action="http://services.exercisel.lab12.webprogramming.aal718.unin.it/MapGroupGenerator/getBoundsResponse" message="tns:getBoundsResponse"/>
      </operation>
      <operation name="getShopIdsInQuadrant">
        <input wsam:Action="http://services.exercisel.lab12.webprogramming.aal718.unin.it/MapGroupGenerator/getShopIdsInQuadrantRequest" message="tns:getShopIdsInQuadrant"/>
        <output wsam:Action="http://services.exercisel.lab12.webprogramming.aal718.unin.it/MapGroupGenerator/getShopIdsInQuadrantResponse" message="tns:getShopIdsInQuadrantResponse"/>
      </operation>
      <operation name="getPublicURLMapNeighbors">
        <input wsam:Action="http://services.exercisel.lab12.webprogramming.aal718.unin.it/MapGroupGenerator/getPublicURLMapNeighborsRequest" message="tns:getPublicURLMapNeighbors"/>
        <output wsam:Action="http://services.exercisel.lab12.webprogramming.aal718.unin.it/MapGroupGenerator/getPublicURLMapNeighborsResponse" message="tns:getPublicURLMapNeighborsResponse"/>
      </operation>
      <operation name="getLattling">
        <input wsam:Action="http://services.exercisel.lab12.webprogramming.aal718.unin.it/MapGroupGenerator/getLattlingRequest" message="tns:getLattling"/>
        <output wsam:Action="http://services.exercisel.lab12.webprogramming.aal718.unin.it/MapGroupGenerator/getLattlingResponse" message="tns:getLattlingResponse"/>
      </operation>
    </portType>
    <binding name="MapGroupGeneratorPortBinding" type="tns:MapGroupGenerator">
      <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
      <operation name="getBounds">
        <soap:operation soapAction="">
          <input>
            <soap:body use="literal"/>
          </input>
          <output>
            <soap:body use="literal"/>
          </output>
        </operation>
      <operation name="getShopIdsInQuadrant">
        <soap:operation soapAction="">
          <input>
            <soap:body use="literal"/>
          </input>
          <output>
            <soap:body use="literal"/>
          </output>
        </operation>
      <operation name="getPublicURLMapNeighbors">
        <soap:operation soapAction="">
          <input>
            <soap:body use="literal"/>
          </input>
          <output>
            <soap:body use="literal"/>
          </output>
        </operation>
      <operation name="getLattling">
        <soap:operation soapAction="">
          <input>
            <soap:body use="literal"/>
          </input>
          <output>
            <soap:body use="literal"/>
          </output>
        </operation>
      </binding>
    <service name="MapGroupGenerator">
      <port name="MapGroupGeneratorPort" binding="tns:MapGroupGeneratorPortBinding">
        <soap:address location="http://localhost:8080/Exercise1/MapGroupGenerator?wsdl">
        </soap:address>
      </port>
    </service>
  </definitions>
```

NetBeans

New Project

- Choose Project



NetBeans

New Project

- Name and Location

New Java Application

Steps

1. Choose Project
2. **Name and Location**

Name and Location

Project Name: Exercise 1-Services-Client

Project Location: ts\NetBeansProjects\University\WebDeveloping\2017-2018\Lab 12 Browse...

Project Folder: ersity\WebDeveloping\2017-2018\Lab 12\Exercise 1-Services-Client

Artifact Id: Exercise 1-Services-Client

Group Id: it.unitn.aa1718.webprogramming.lab 12

Version: 1.0

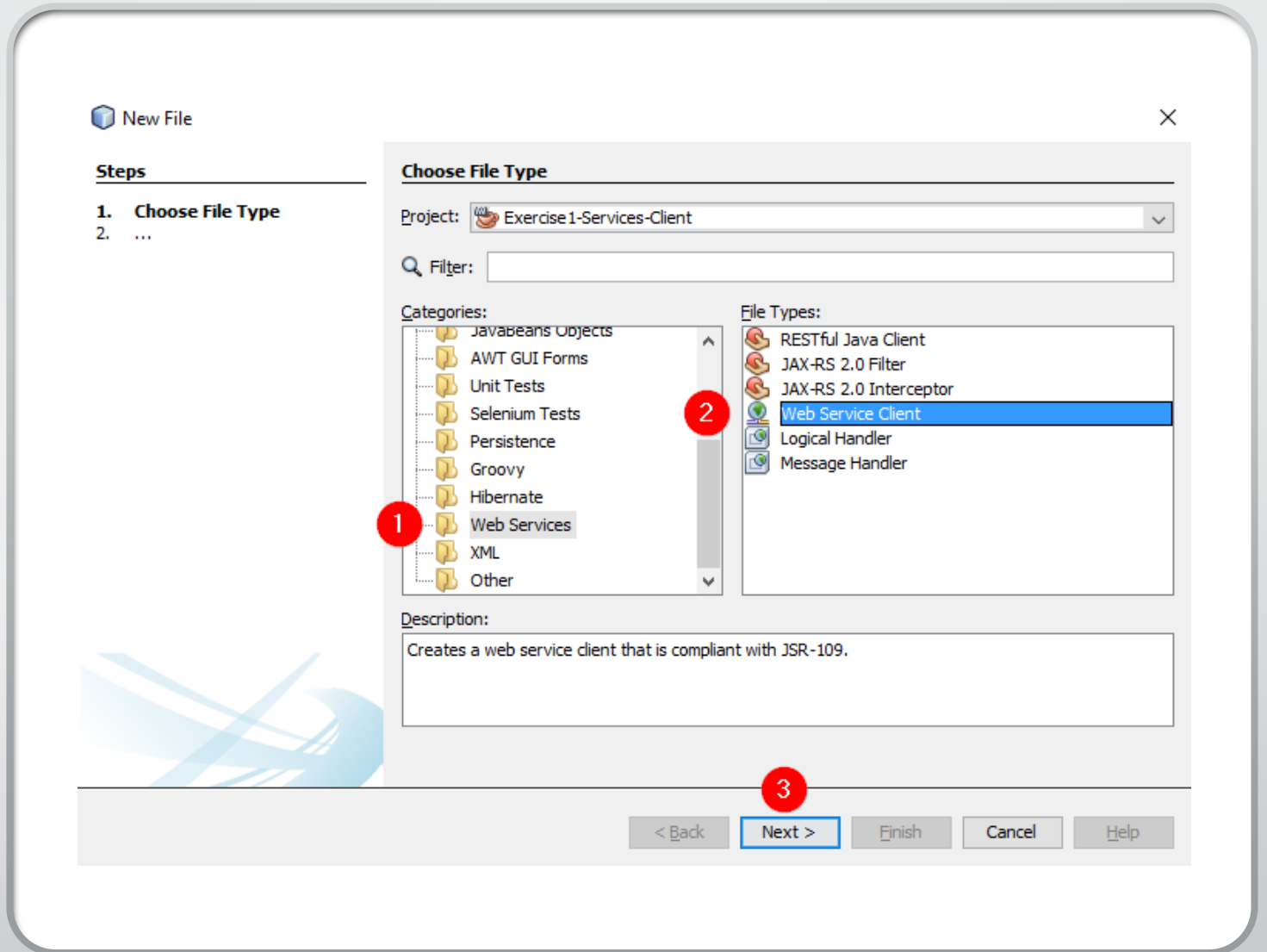
Package: it.unitn.aa1718.webprogramming.lab 12.exercise 1.services (Optional)

< Back Next > **Finish** Cancel Help

NetBeans

New File

- Choose File Type



NetBeans

New Web Service Client

- WSDL and Client Location

New Web Service Client

Steps

1. Choose File Type
2. **WSDL and Client Location**

WSDL and Client Location

Specify the WSDL file of the Web Service.

☐ Project:

☐ Local File:

1 ☒ WSDL URL:

☐ IDE Registered:

Specify a package name where the client java artifacts will be generated:

Project: Exercise 1-Services-Client

2 Package:

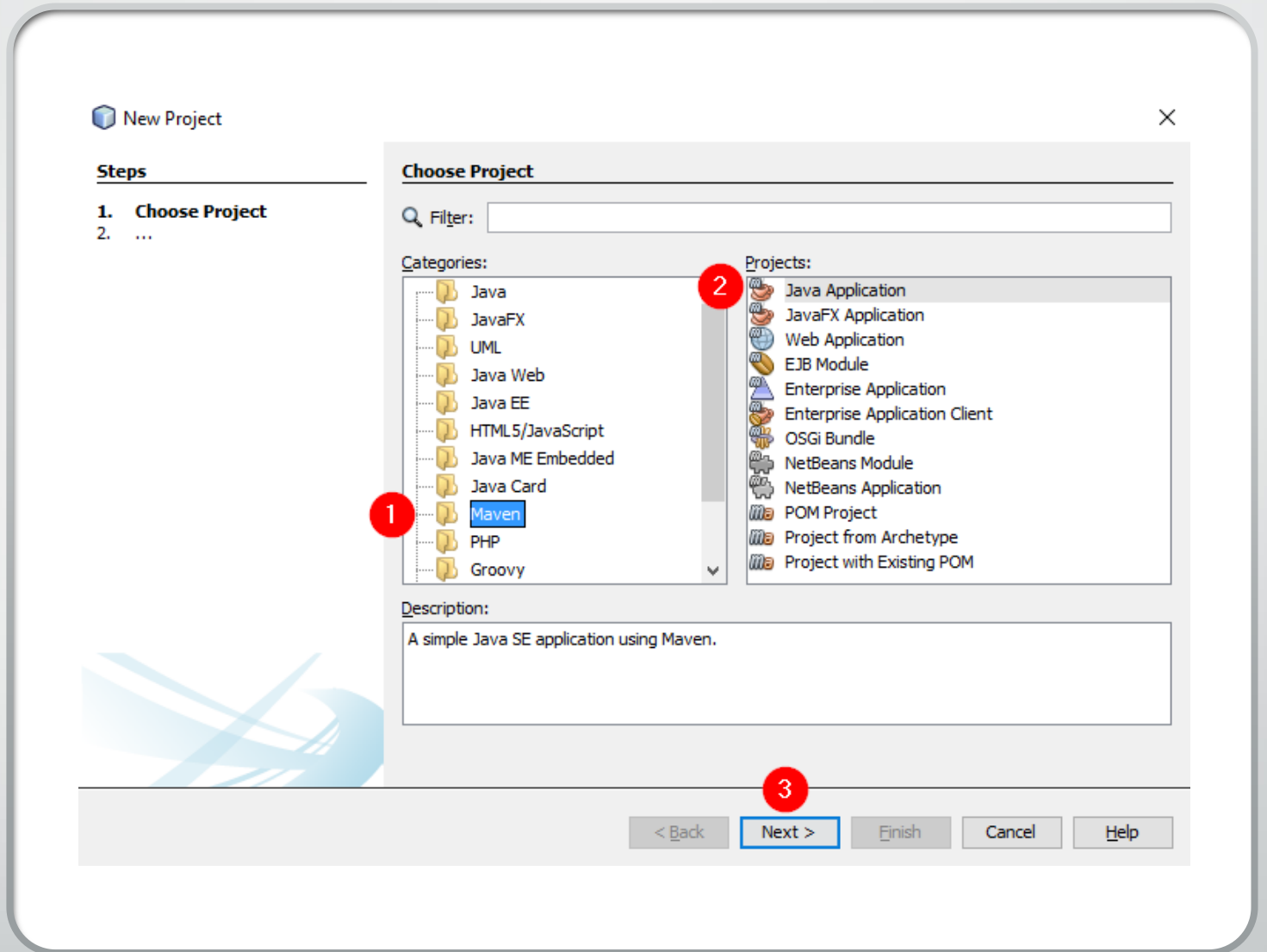
☐ Generate Dispatch code

3

NetBeans

New Project

- Choose Project



NetBeans

New Java Application

- Name and Location

New Java Application

Steps

1. Choose Project
2. **Name and Location**

Name and Location

Project Name: Exercise 1-Client

Project Location: rts\NetBeansProjects\University\WebDeveloping\2017-2018\Lab 12 Browse...

Project Folder: jects\University\WebDeveloping\2017-2018\Lab 12\Exercise 1-Client

Artifact Id: Exercise 1-Client

Group Id: it.unitn.aa1718.webprogramming.lab 12

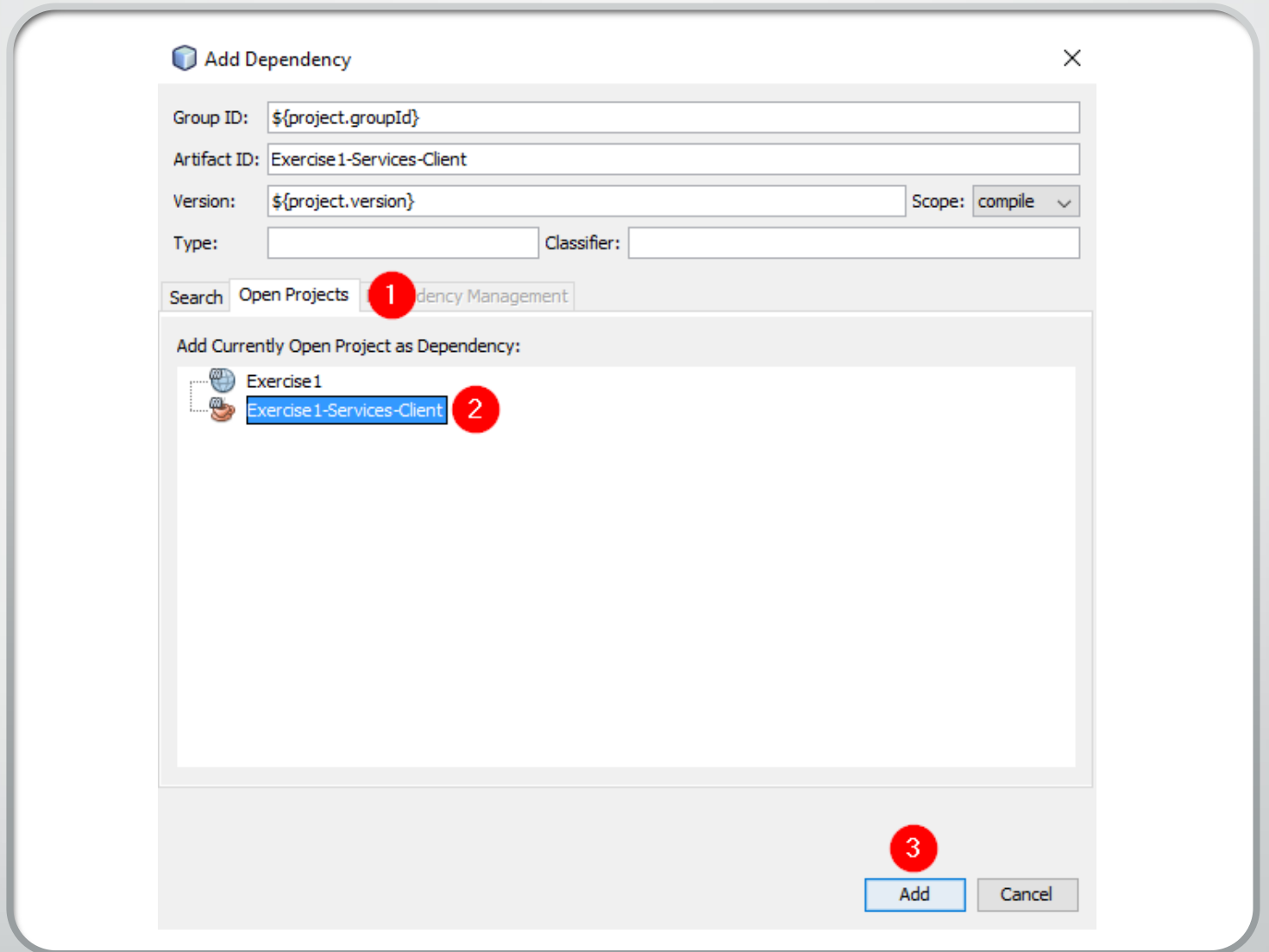
Version: 1.0

Package: it.unitn.aa1718.webprogramming.lab 12.exercise 1.client (Optional)

< Back Next > **Finish** Cancel Help

NetBeans

Add project dependency



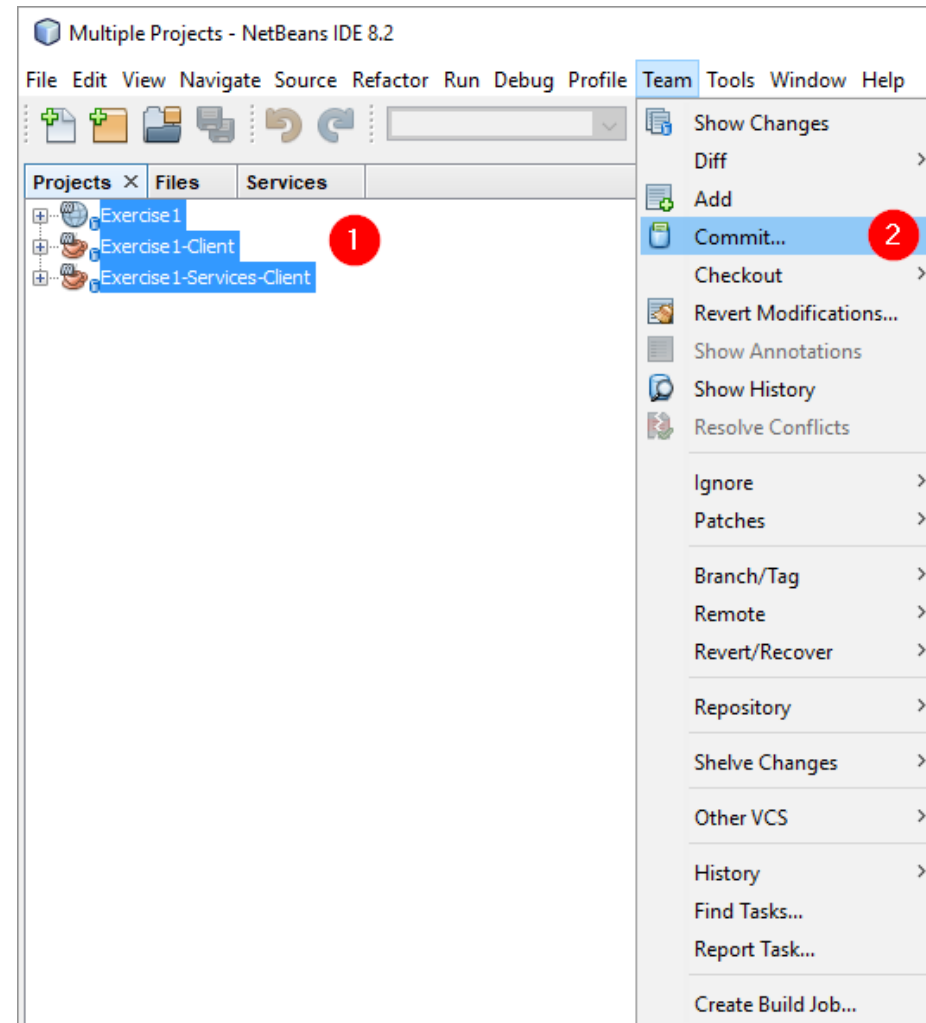
NetBeans

Coding...

```
public class Client {  
    public static void main(String[] args) {  
        MapGroupGenerator service = new MapGroupGenerator_Service().getMapGroupGeneratorPort();  
        List<Object> latLng = service.getLatLng(2131943431);  
        if (latLng.size() == 2) {  
            Double distance = 100.0;  
            Double centerLat = (Double) latLng.get(0);  
            Double centerLng = (Double) latLng.get(1);  
  
            Double ddistance = Math.sqrt(2 * Math.pow(distance, 2));  
            ddistance = ddistance / 6371;  
  
            Double topBrng = Math.toRadians(315);  
            Double bottomBrng = Math.toRadians(135);  
  
            centerLat = Math.toRadians(centerLat);  
            centerLng = Math.toRadians(centerLng);  
        }  
    }  
}
```

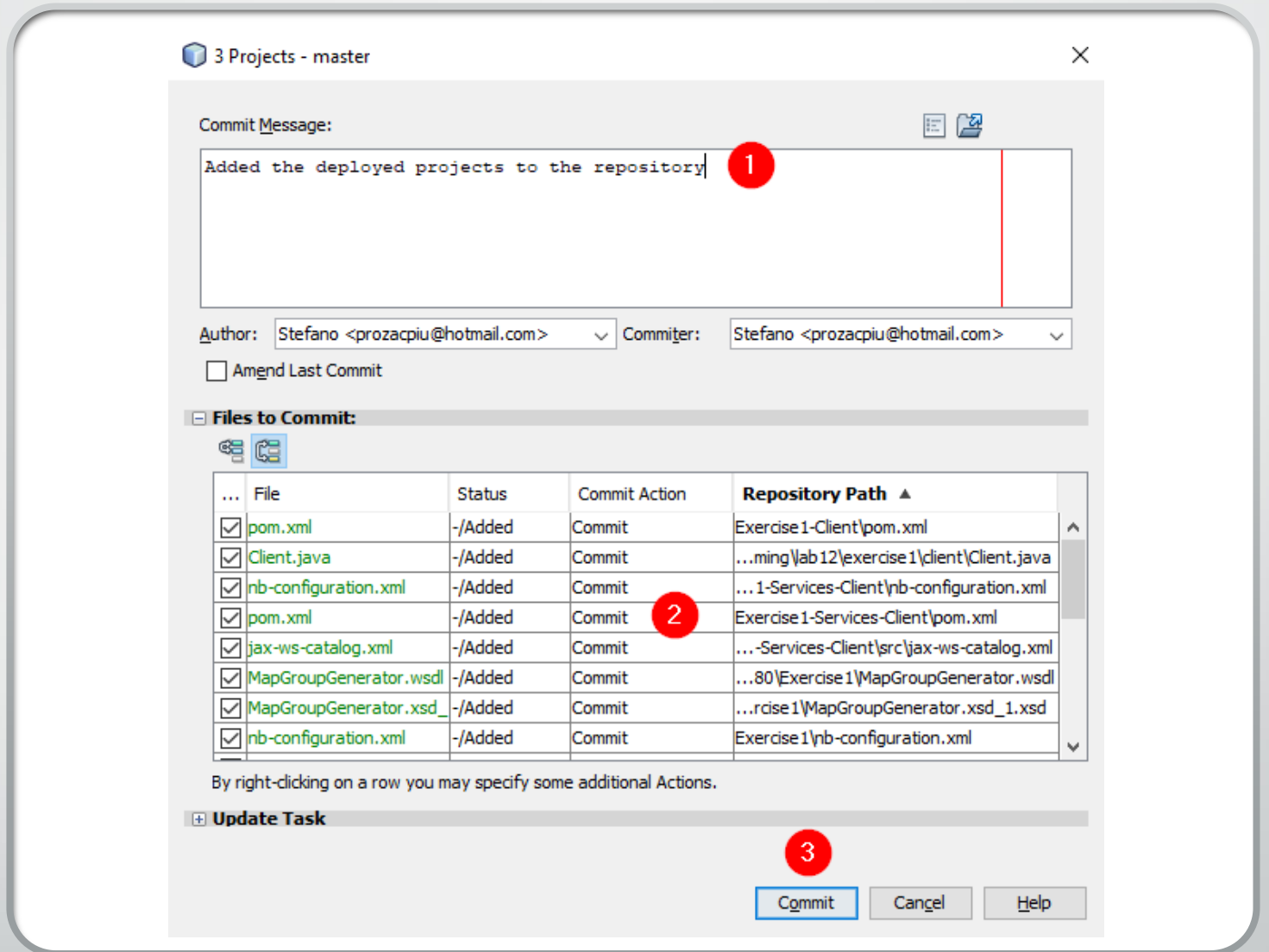
GitHub

Add new/modified files to the repository.



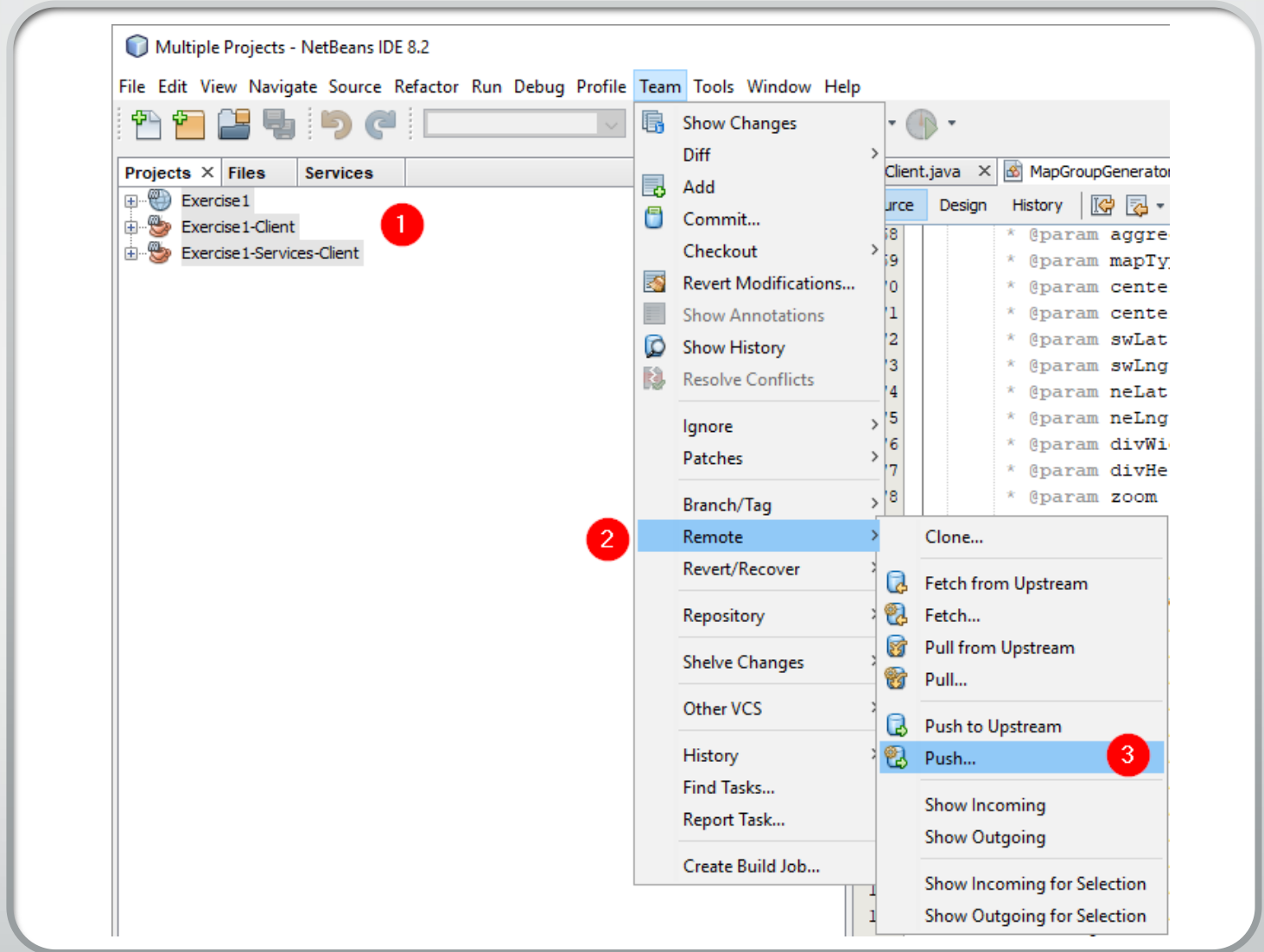
GitHub

Commit the projects



GitHub

Push to Remote Repository



GitHub

Push to Remote Repository

- Remote Repository

Push to Remote Repository

Steps

1. **Remote Repository**
2. Select Local Branches
3. Update Local References

Remote Repository

☒ **Select Configured Git Repository Location:**

origin:https://SCUnitN@github.com/SCUnitN/Lab12.git

☐ **Specify Git Repository Location:**

Remote Name: origin ☒ Persist Remote

Repository URL: https://github.com/SCUnitN/Lab12.git

http[s]://host.xz[:port]/path/to/repo.git/

User: SCUnitN (leave blank for anonymous access)

Password: ☐ Save Password

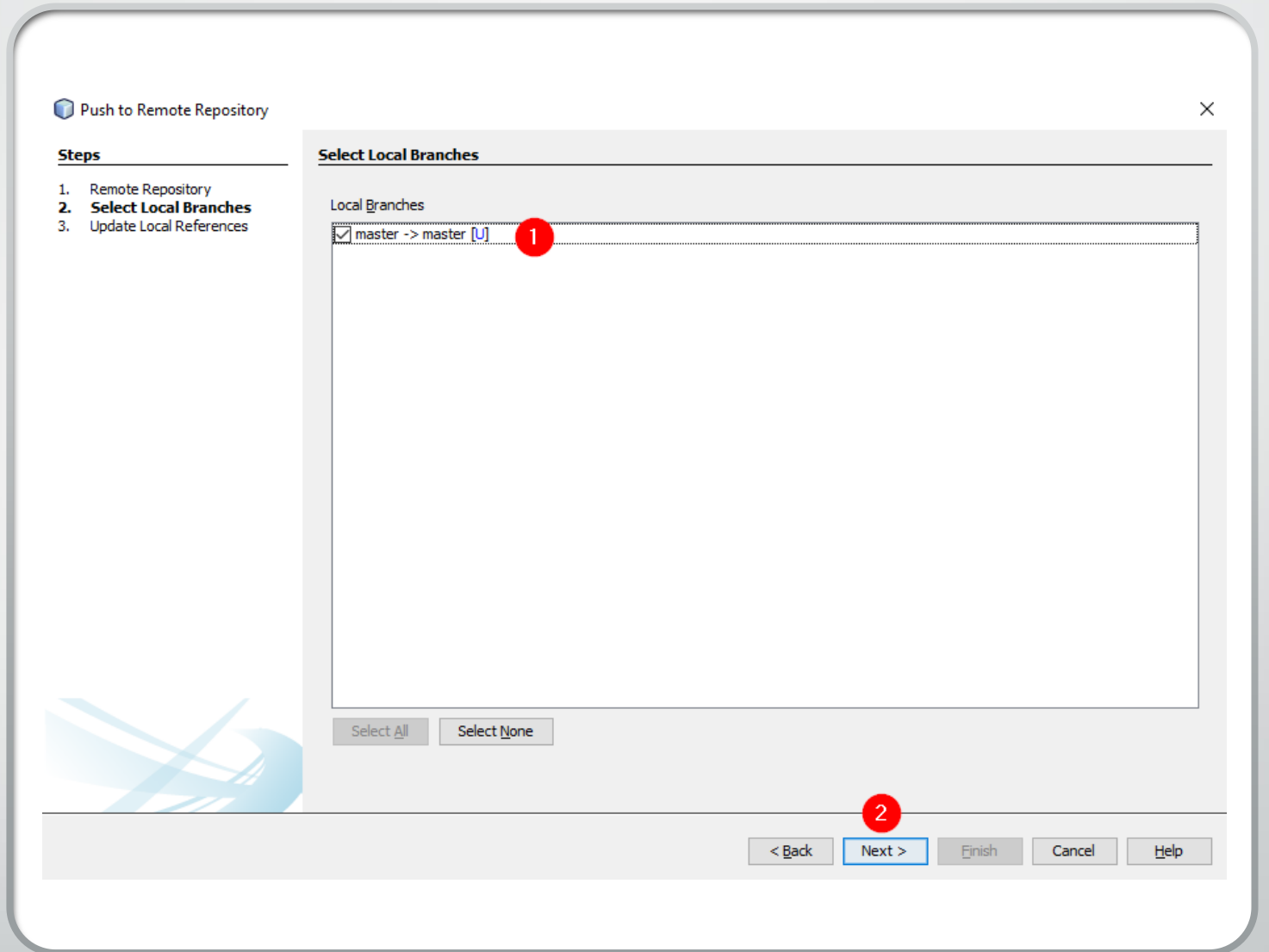
Proxy Configuration...

< Back **Next >** Finish Cancel Help

GitHub

Push to Remote Repository

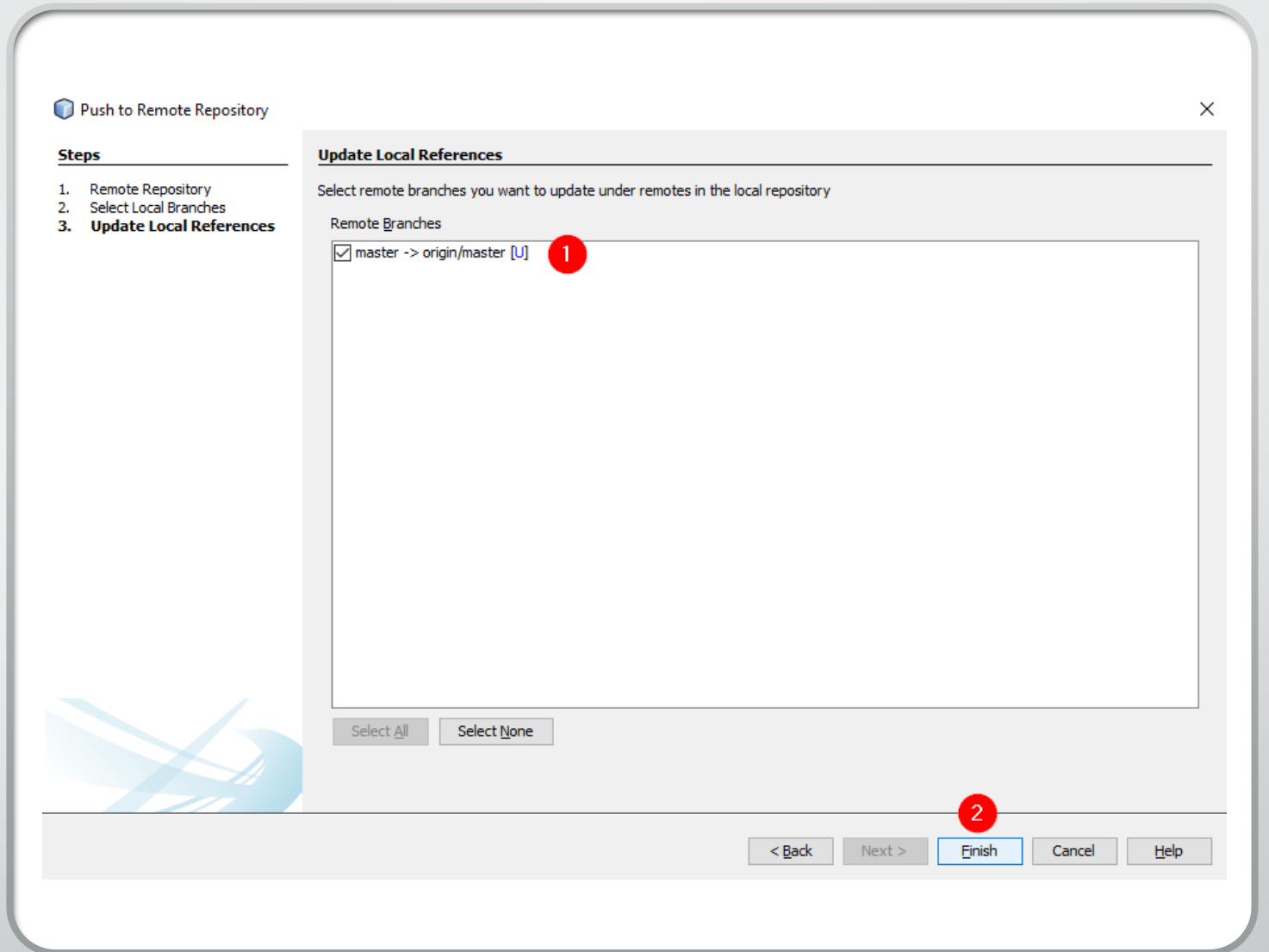
- Select Local Branches



GitHub

Push to Remote Repository

- Update Local References



References

- <https://netbeans.org/kb/docs/ide/git.html>
- https://docs.oracle.com/netbeans/nb82/netbeans/NBDAG/version_control.htm#NBDAG237
- <http://www.foo.be/cours/dess-20122013/b/OReilly%20Version%20Control%20with%20GIT.pdf>
- <https://github.com/>
- <https://en.wikipedia.org/wiki/SOAP>
- <https://developers.google.com/maps/>