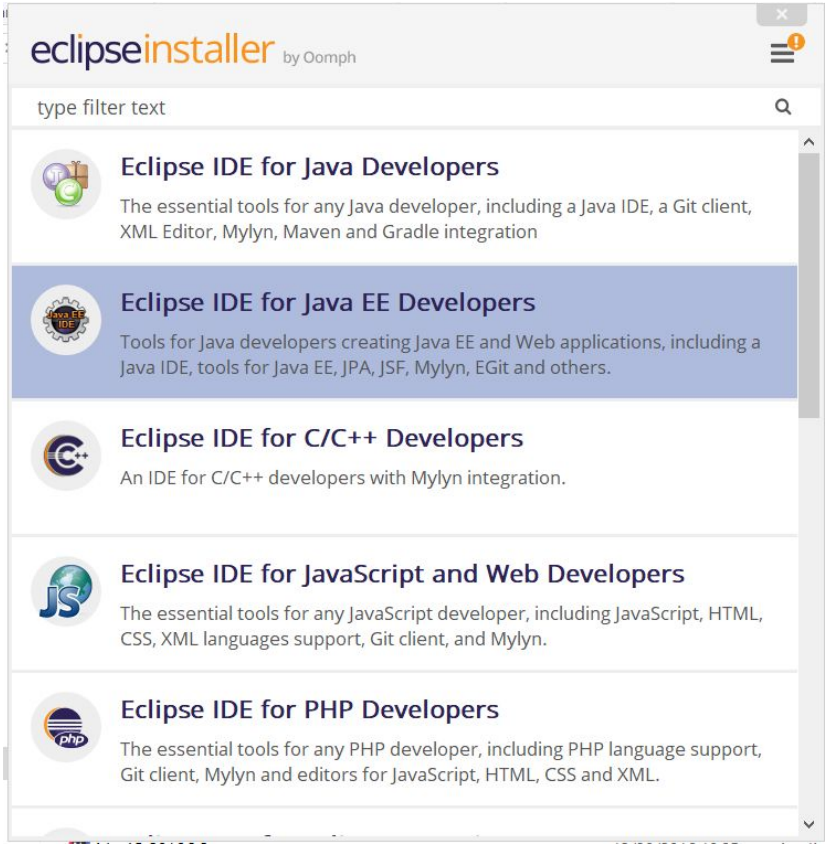
Development Manual

Contents

Software Installations

1. Java Development Kit (JDK)
    a. The gym management system has been implemented using Java Maven Project and so Java is the main software that is need to run the system.
    b. Go to http://www.oracle.com/technetwork/java/javase/downloads/index.html to download and install JDK.
    c. The installation instructions are given at https://docs.oracle.com/javase/9/index.html
2. Since the project is a maven project, all the additional dependences have been included in the pom.xml. These dependencies will be automatically downloaded and installed for the project.
3. MySql Community edition
    a. The Gym management system uses MySql, a relational database to store the data
    b. Go to https://dev.mysql.com/downloads/mysql/ to download MySql
    c. Choose the appropriate operating system and download the software.
    d. Follow the online instructions at https://dev.mysql.com/doc/refman/5.7/en/installing.html to install the MySql server..
4. Git
    a. The version control of the gym management system is maintained using GIT a version control tool
    b. Go to https://git-scm.com/downloads to download Git and follow the installation instructions at https://git-scm.com/book/en/v2/Getting-Started-Installing-Git to install it.
5. Eclipse
    a. Eclipse is a Java IDE used for programming. Any IDE can be used to setup this project but this manual will only give instructions on how to setup the project in Eclipse.
    b. Eclipse can be downloaded from http://www.eclipse.org/
    c. When installing eclipse make sure you choose the Java EE version



   d. Documentation for eclipse is available at http://www.eclipse.org/users

6. Tomcat
   a. Apache Tomcat is an open source Java Servlet Container.
   b. It can be downloaded from https://tomcat.apache.org/download-90.cgi

Eclipse Project setup

Cloning the project

1. Using the link below clone the git repository to your local file system
   https://github.com/annettekotian/cs414-f17-801-enigma.git

Setting up the database

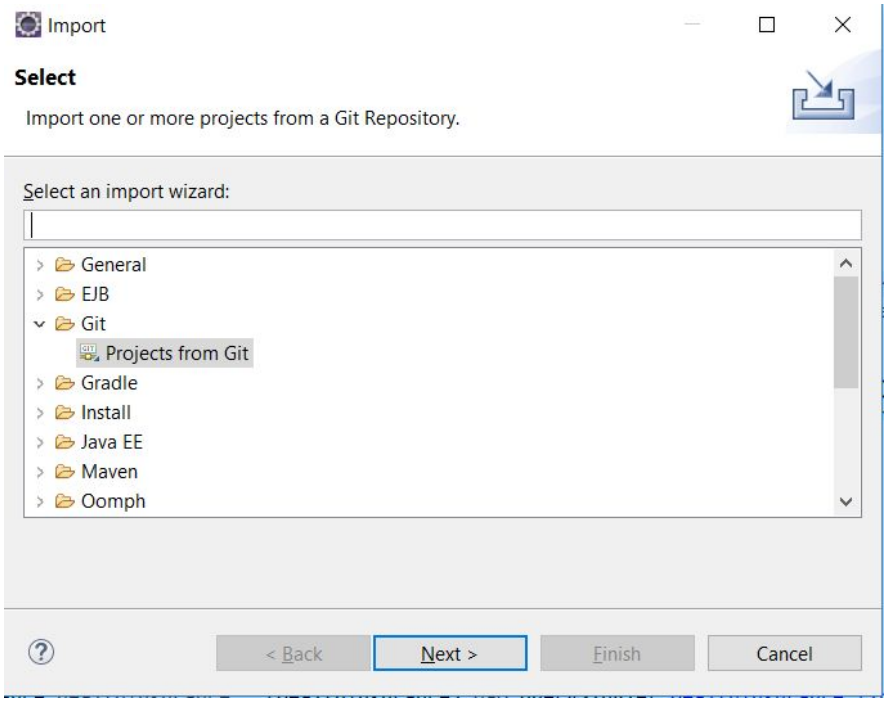
1. A database dump of the gym management system called **“gym_system_schema.sql”**, is present in the folder “db_schema” located at the root of the repository.
2. Use the following command to run the sql script file

```
mysql -u username -p database_name < file.sql
```

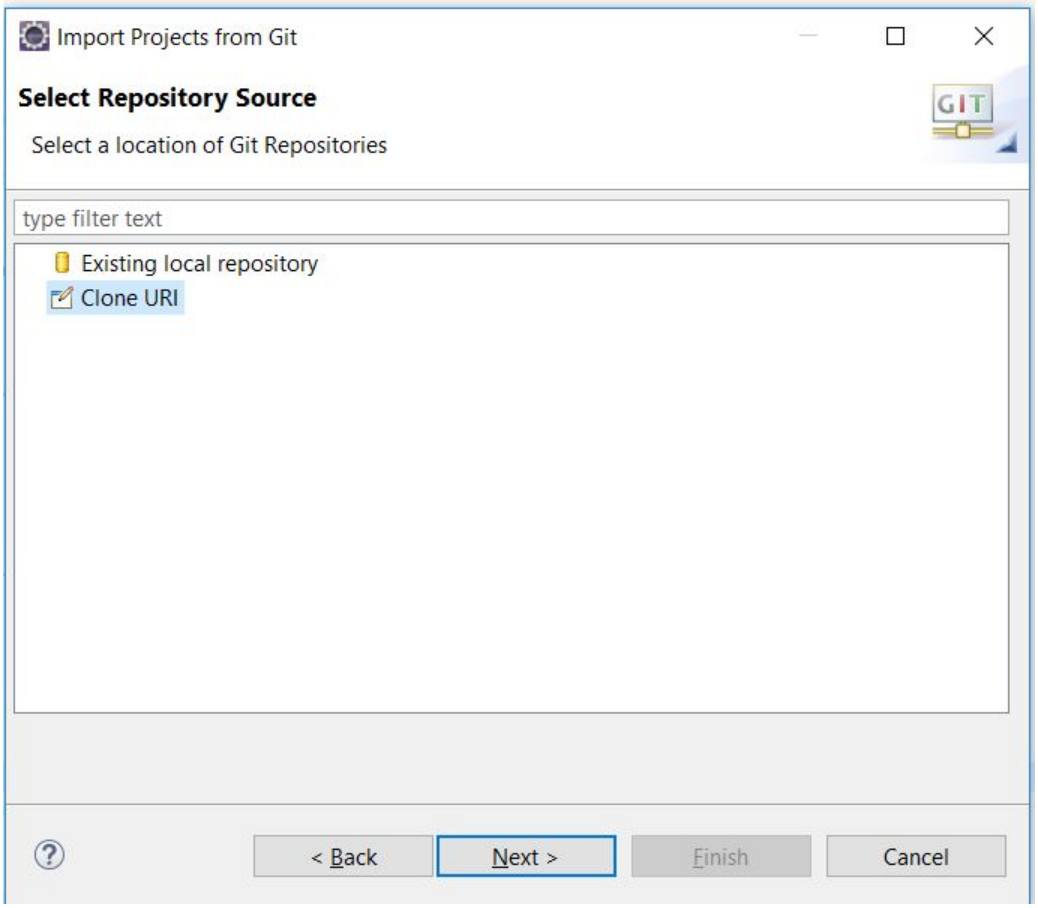
Setting up the project in Eclipse

Using the repository URI

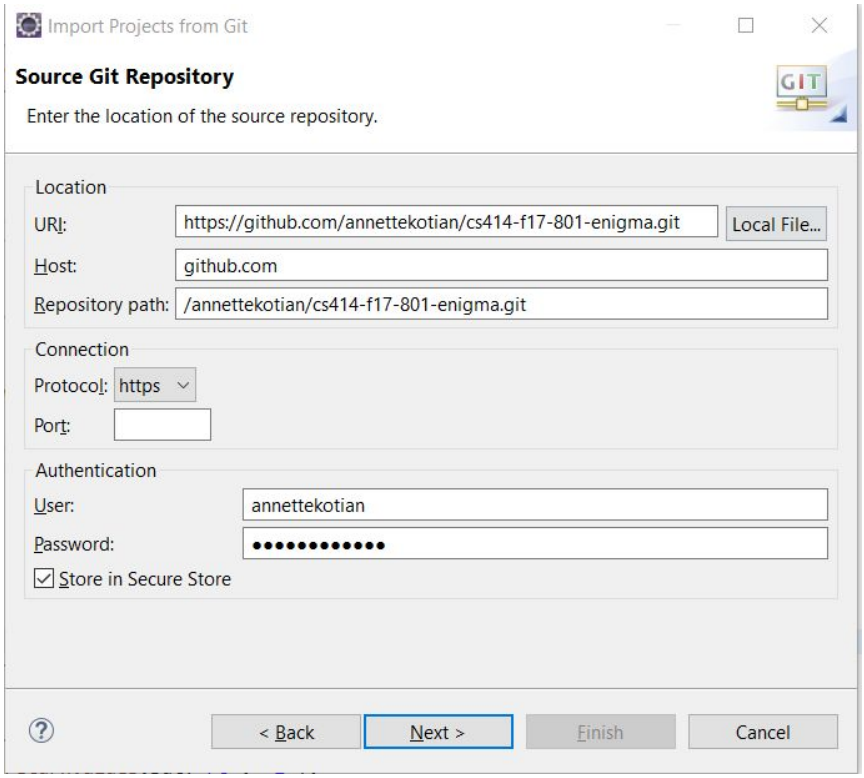
1. After opening eclipse go to File>Import>Projects from GIT



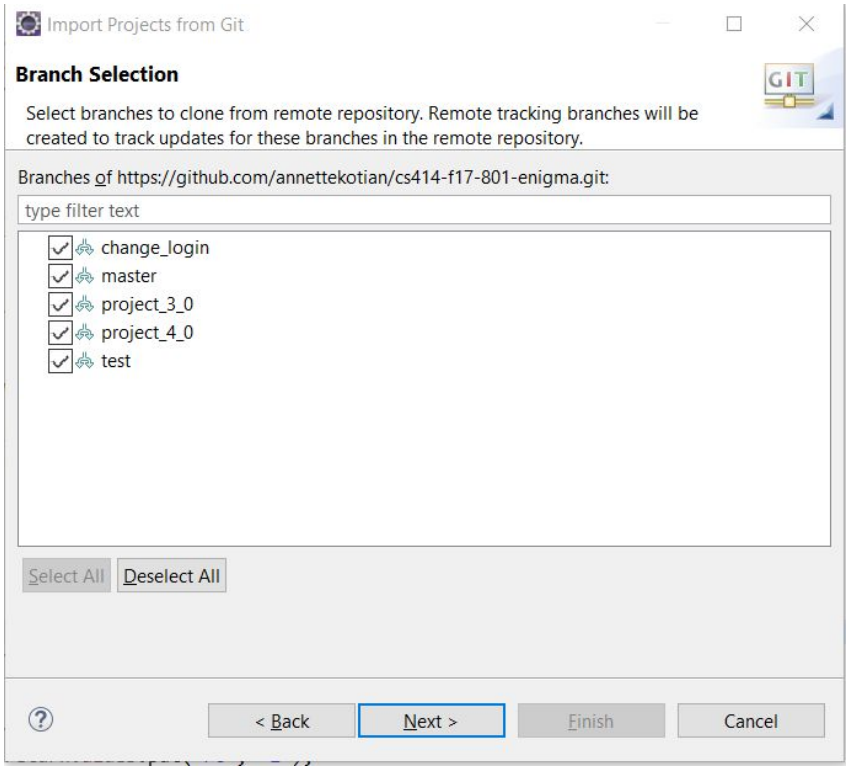
2. Choose Clone URI.



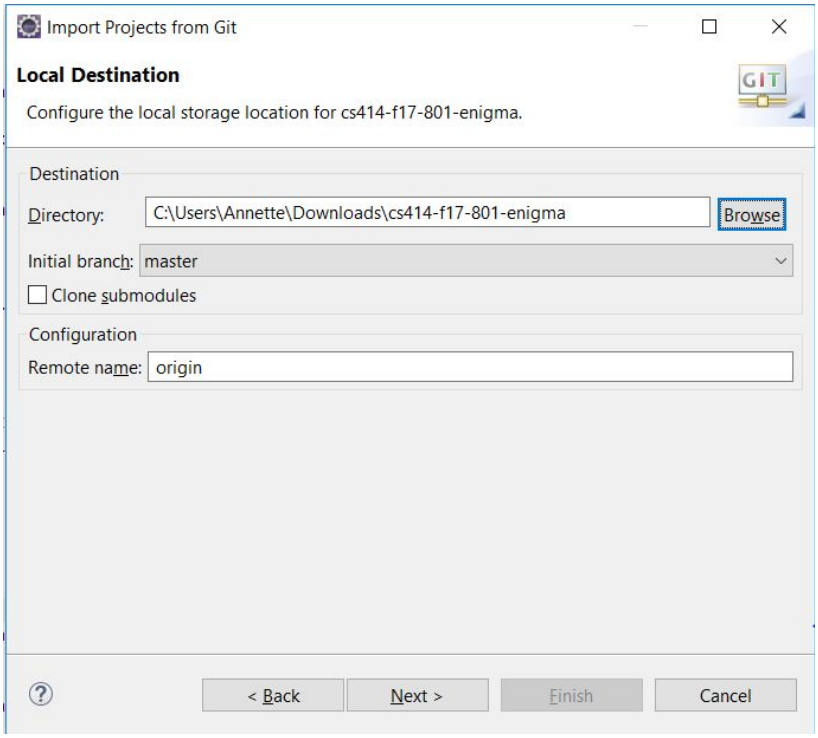
3. Click next. Enter all the details. The uri for this project is
https://github.com/annettekotian/cs414-f17-801-enigma.git



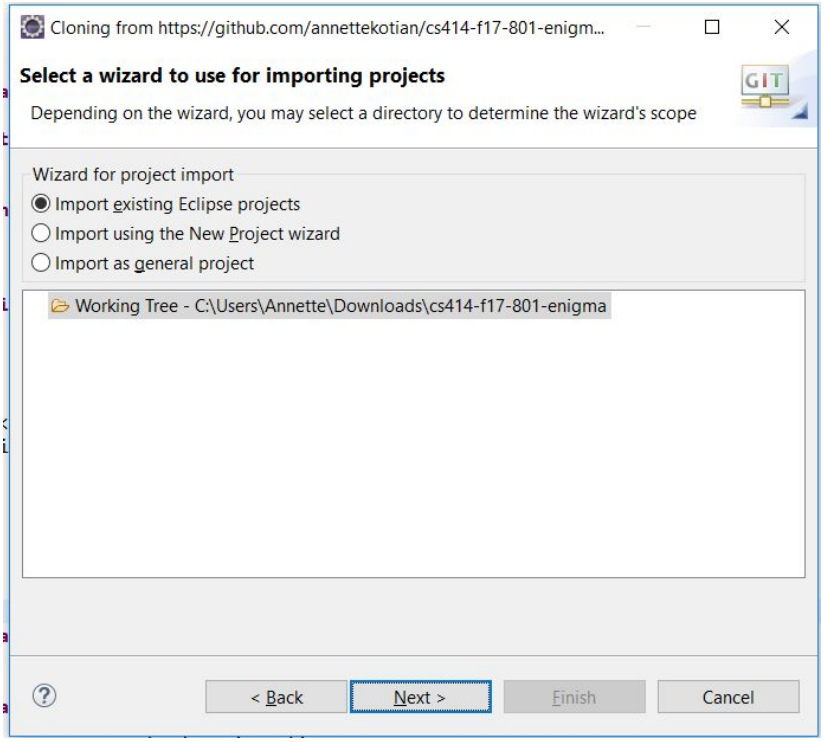
4. Click next. Choose the branches to be imported. Note that at least Master must be selected.



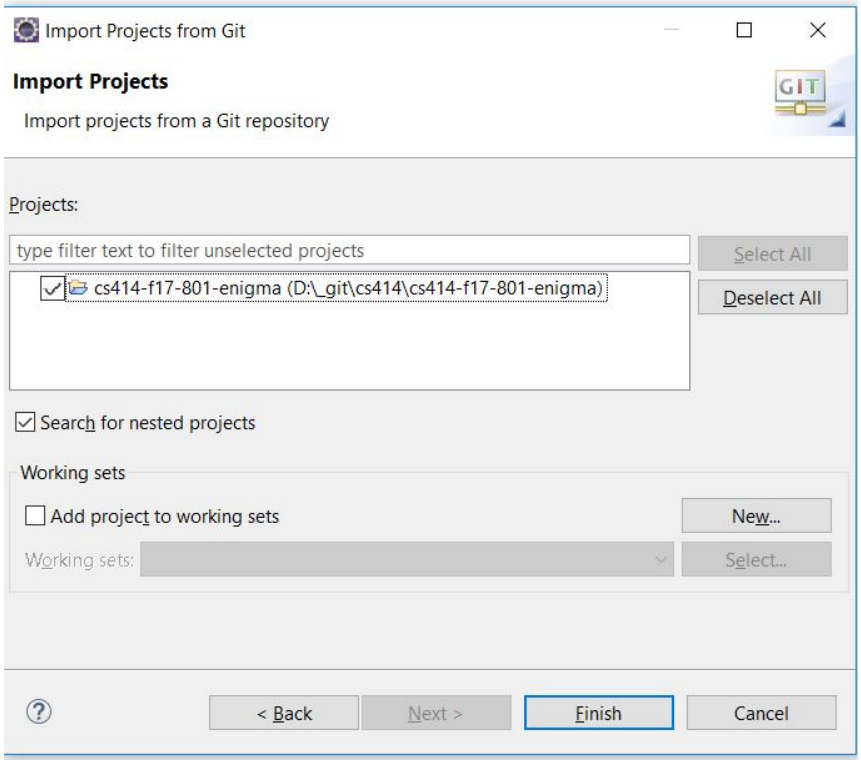
5. Click next. Choose a directory on your local system where the project can be saved



6. Click next. Click on Import existing eclipse projects

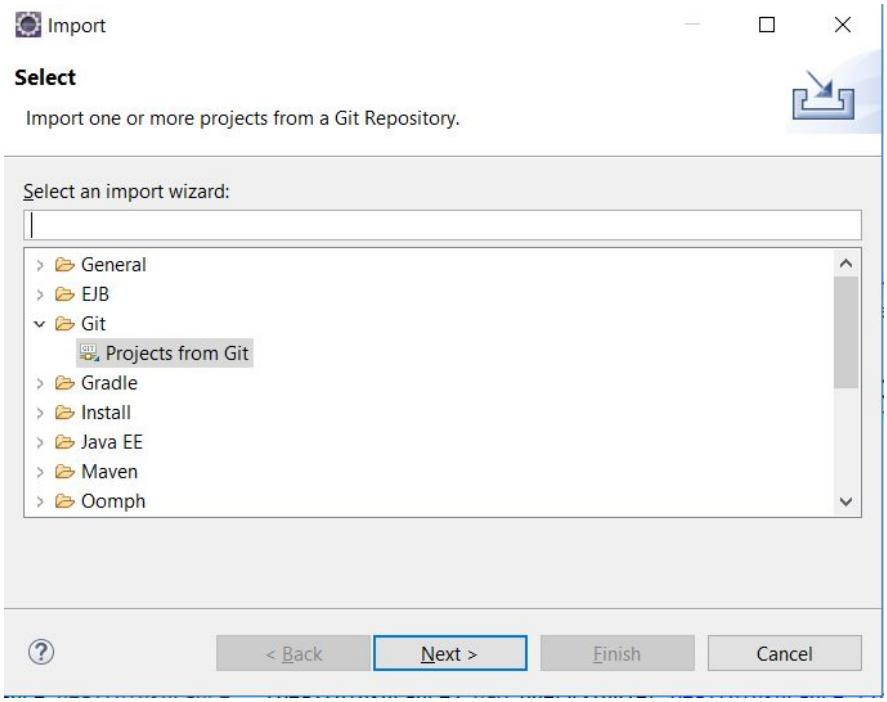


7. Click next. Select the project to be imported and click Finish.

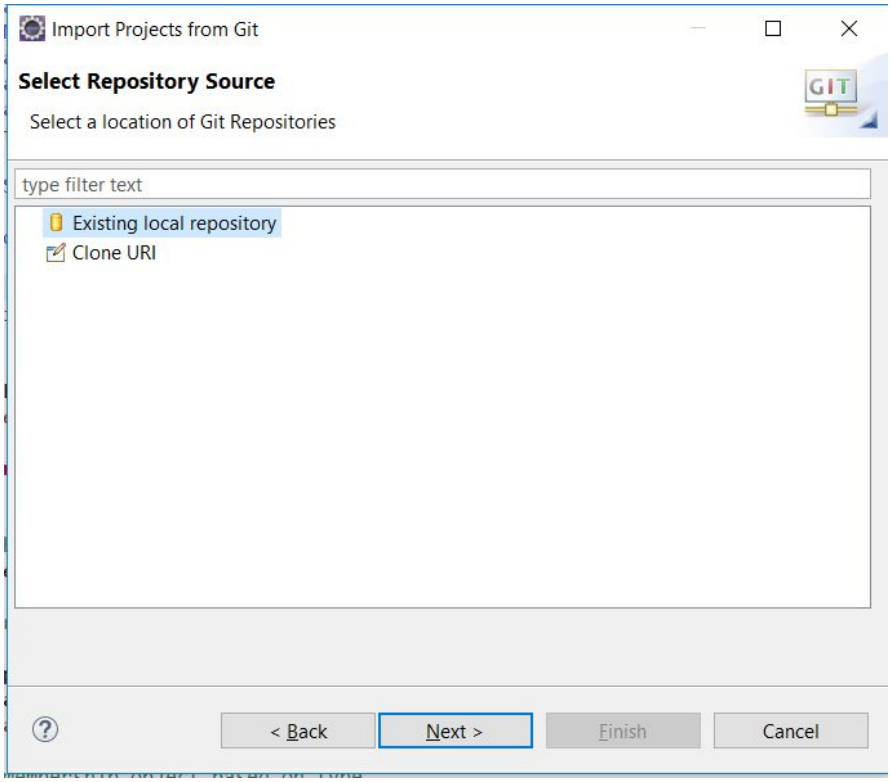


Using the repository cloned on the local system

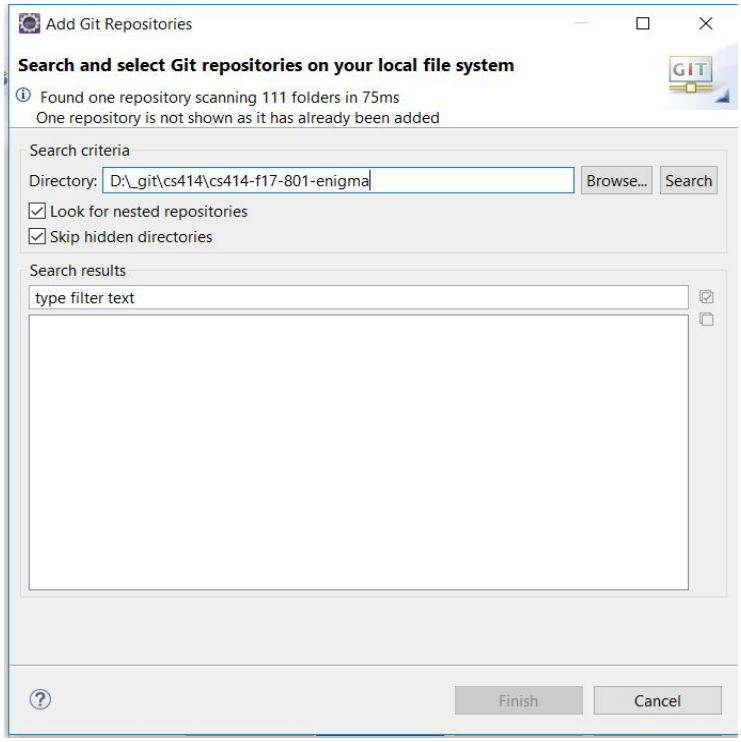
1. After opening eclipse go to File>Import>Projects from GIT



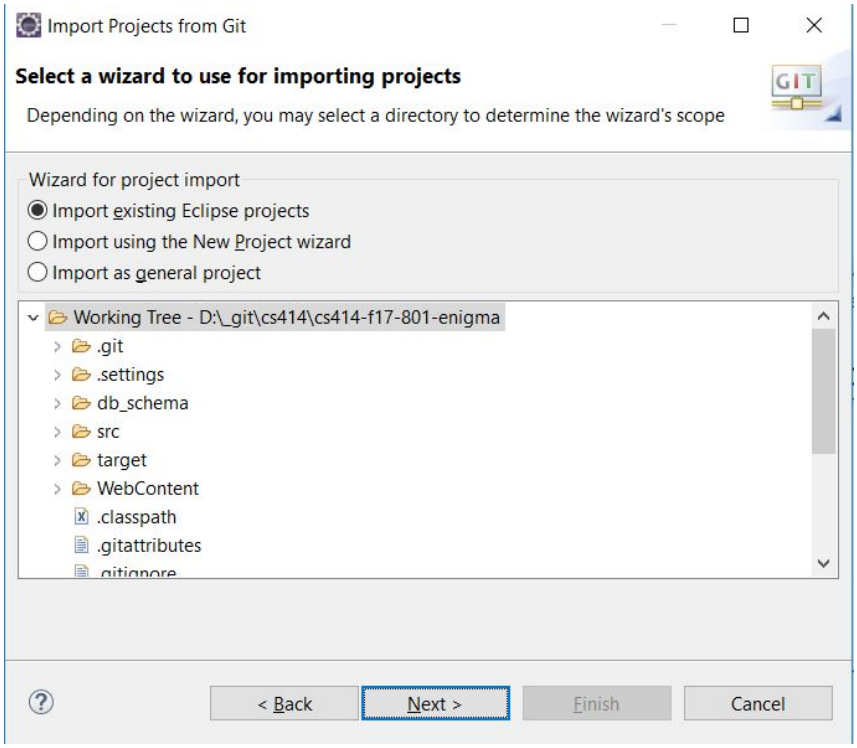
2. Click next. Choose Existing Local Repository.



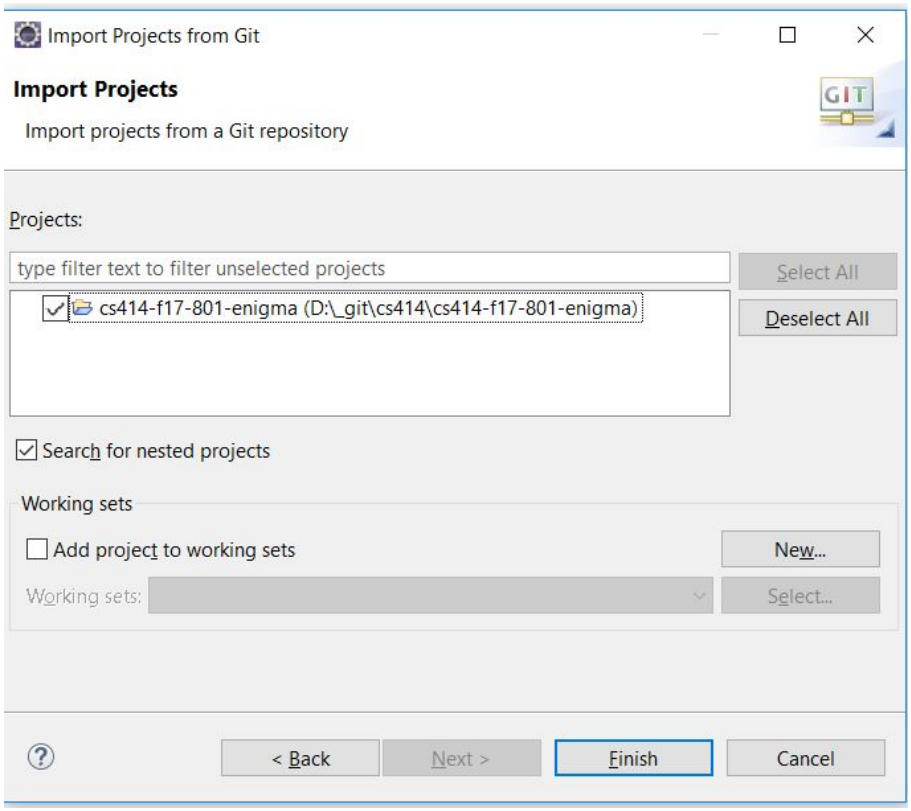
3. Click next. Click Add and choose the path the of repository cloned on your local system



4. Click on Import existing Eclipse projects



5. Click next. Select the project to be imported and click Finish

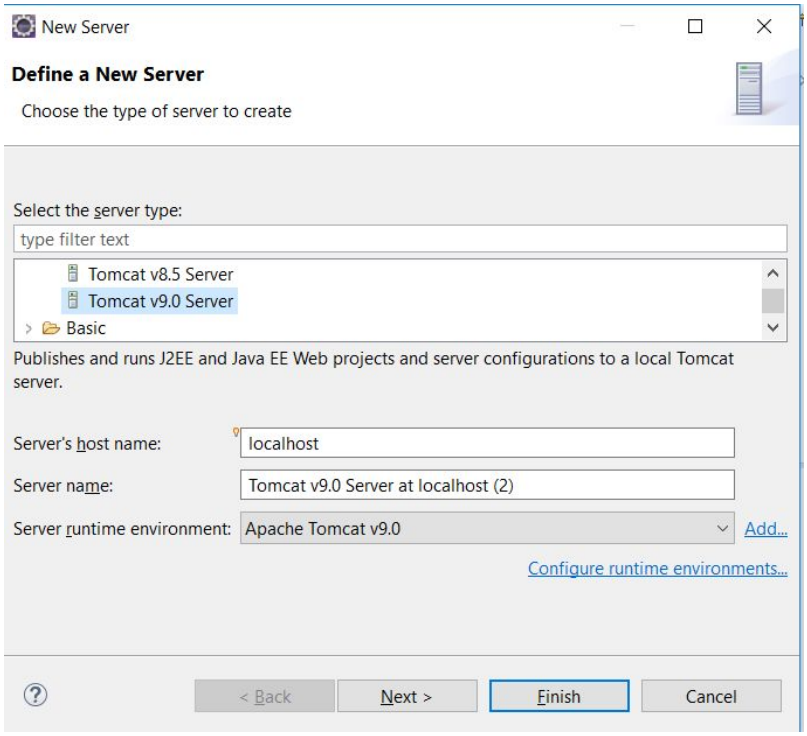


Changing the database username and password

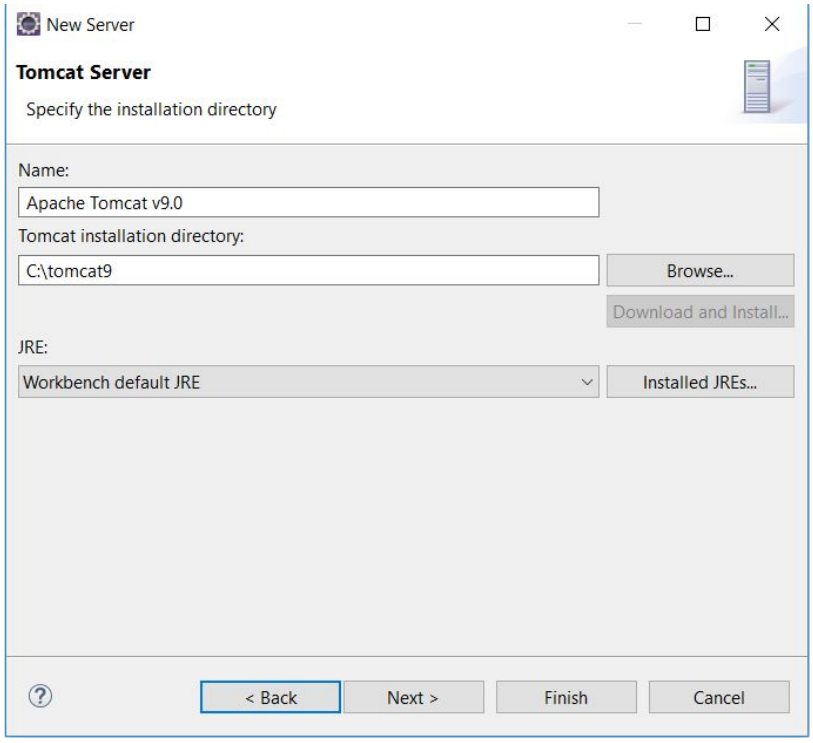
1. Open persistence.xml located in {local_repository_path}\src\main\resources\META-INF.
2. It will have a tag looking like this
   <property name="javax.persistence.jdbc.user" value="root" />
   <property name="javax.persistence.jdbc.password" value="pass123"/>
3. Set the username and password your MySql username and password

Setting up tomcat with the project

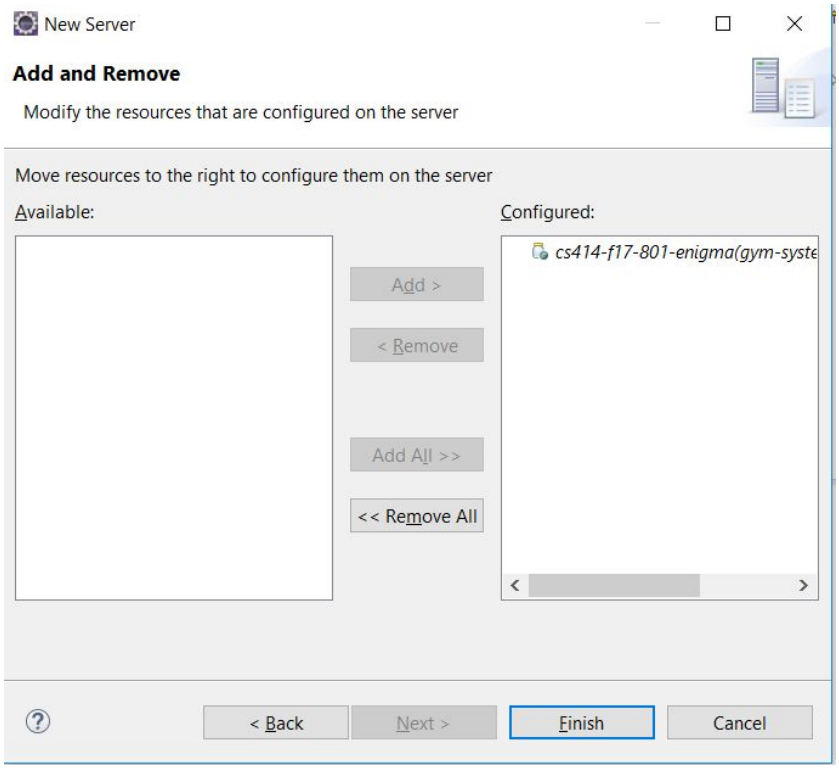
1. Open the servers window in eclipse.
2. Right click > New > Server



3. Click next. Choose the tomcat installation directory.

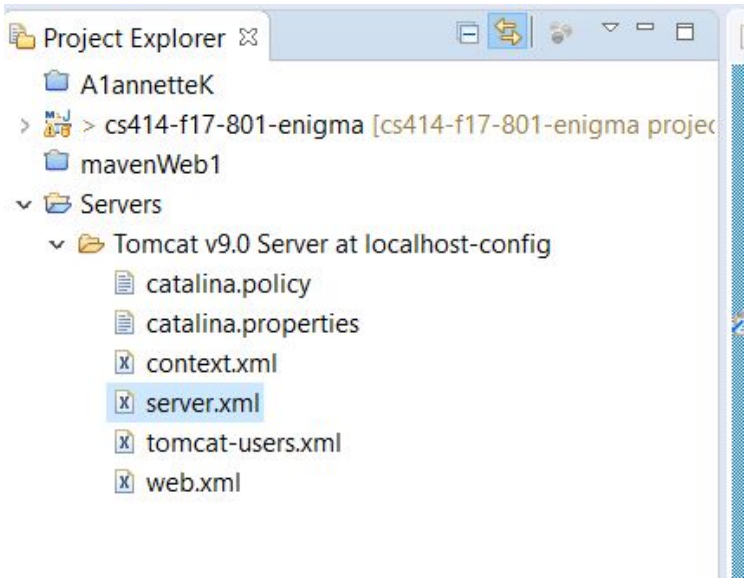


4. Click next. Choose the project to be added to the server and click Finish



Server.xml Changes

1. After tomcat has been setup, a change has to be made in server.xml of tomcat server. The project explorer window has a folder icon for server. Server.xml can be found under Servers> Tomcat v9.0 Server at localhost-config



2. The server.xml will have the following entry at the end of the file
<Context docBase="cs414-f17-801-enigma" path="/ROOT" reloadable="true" source="org.eclipse.jst.jee.server:cs414-f17-801-enigma"/>

Change this to

<Context docBase="cs414-f17-801-enigma" path="/" reloadable="true" source="org.eclipse.jst.jee.server:cs414-f17-801-enigma"/>

3. To run the project, right click on the project > Run as > Run on Server. The server should start and the web browser should display the login page of the gym system.

Setting up the image path to store machine images

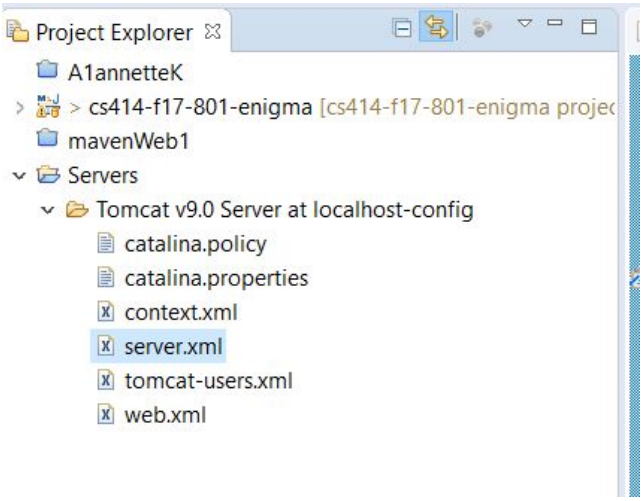
1. Adding machines to the gym system inventory requires images. Therefore to store the images and display it in the web, there are two configurations necessary.
    a. The web.xml file in {local_repository_path}\WebConent\Web-INF has the following entry
    <context-param>
        <param-name>path_to_upload</param-name>
        <param-value>PATH_ON_YOUR_LOCAL_SYSTEM</param-value>
    </context-param>
    Replace the highlighted part with a path on your local file system

b. In order to serve the images stored in the local disk as a url in the web ui, server.xml located under {tomcat_install_directory}/conf/server.xml has to to be edited. The following entry within the <Host></Host> tag has to be added.

<Context docBase="PATH_TO_YOUR_LOCAL_SYSTEM" path="/machineImages" />

Replace the highlighted part with the path where the images are stored. Note that the both the paths (point a and b) have to be the same.

Note: If your have set up tomcat with eclipse , the project explorer window, has a drop down for Servers. If you open Tomcat Server v 9.0 Server at localhost-config, it has server.xml present within it.



Login credentials

1. If the database dump has been installed successfully it has a record for the admin username and password. You can log in to the system using these credentials and further create usernames and passwords for managers and trainers.
   Username: admin
   Password: password

Eclipse Running the tests

1. The src>tests folder within the repository has all the Unit Test cases written using JUnit.

2. The package java.edu.colostate.cs.cs41.enigma has the test suite file called “AllTests.java” which will run all the test cases. In order to run the file from eclipse right click on the file>Run as> JUnit Test. This test suite will run all the test files within the test folder.
3. Additionally, there are more test files within the packages in the test folder. To run any test individually - right click on the file>Run as> JUnit Test

Maven

The project is setup with Maven integration to both run tests and package the project. Maven needs to be downloaded and installed on the location machine.

https://maven.apache.org/

Initial setup

1. Using the link below clone the git repository to your desktop
   https://github.com/annettekotian/cs414-f17-801-enigma.git

2. Open persistence.xml located in {local_repository_path}\src\main\resources\META-INF. Set the username and password your MySql username and password.
   <property name="javax.persistence.jdbc.user" value="root" />
   <property name="javax.persistence.jdbc.password" value="pass123"/>

3. A database dump of the gym management system called **“gym_system_schema.sql”**, is present in the folder “db_schema” located at the root of the repository.

4. Use the following command to run the sql script file
   ```
   mysql -u username -p database_name < file.sql
   ```

Running test

1. Change to the cloned directory. There should exist a pom.xml file which contains the Maven configuration.

2. Run the following command to execute all unit tests.

   ```
   mvn test
   ```

There should be output reporting the number of tests run along with errors and failures.

Packaging Project

1. Change to the cloned directory. There should exist a pom.xml file which contains the Maven configuration.

2. Run the following command to package a war file. Note that the file name is ROOT.war.

```
mvn clean package
```

The following is example output from a system running Linux.

```
[admin@localhost cs414-f17-801-enigma]$ mvn clean package
.
.
.
[INFO] Webapp assembled in [93 msecs]
[INFO] Building war: /home/admin/git/cs414-f17-801-enigma/target/ROOT.war
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time: 10.417s
[INFO] Finished at: Sun Oct 29 17:45:51 EDT 2017
[INFO] Final Memory: 20M/252M
[INFO] ------------------------------------------------------------------------
```

Running WAR with Tomcat

Note that at least Tomcat 8 needs to be installed on the local machine.
https://tomcat.apache.org/download-80.cgi

1. Stop any Tomcat server process. The following is example output from a system running Linux.

```
[admin@localhost ~]$ systemctl stop tomcat
```

2. Move the ROOT.war file to the Tomcat webapps directory. Location of the directory various based on OS distribution. The following is example output from a system running Linux.

```
[admin@localhost ~]$ sudo mv
/home/admin/git/cs414-f17-801-enigma/target/ROOT.war /var/lib/tomcat/webapps/
```

Note that if there is a root directory in the webapps directory, this needs to be deleted since ROOT.war will be extracted to that directory.

3. Restart the Tomcat server proces. Note that this should extract the contents of all war files in the webapps directory. The following is example output from a system running Linux.

```
[admin@localhost ~]$ systemctl start tomcat
```

4. Since the project is packaged as ROOT.war, this takes the place of the root context of the Tomcat server. This means that pointing a web browser at http://<IP addr>:<port> will return index.jsp of the ROOT.war file.