



# Deliverable No. 10.1.A

## The CHIC Portal

### -Development guidelines

Grant Agreement No.: 600841  
 Deliverable No.: D10.1.A  
 Deliverable Name: The CHIC Portal -Development guidelines  
 Contractual Submission Date: -  
 Actual Submission Date: 08/04/2014

Dissemination Level		
PU	Public	
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	X





COVER AND CONTROL PAGE OF DOCUMENT	
Project Acronym:	CHIC
Project Full Name:	Computational Horizons In Cancer (CHIC): Developing Meta- and Hyper-Multiscale Models and Repositories for In Silico Oncology
Deliverable No.:	D10.1.A
Document name:	The CHIC Portal –Development guidelines
Nature (R, P, D, O) <sup>1</sup>	O
Dissemination Level (PU, PP, RE, CO) <sup>2</sup>	CO
Version:	1
Actual Submission Date:	08/04/2014
Editor: Institution: E-Mail:	Giorgos Zacharioudakis FORTH <a href="mailto:gzaxar@ics.forth.gr">gzaxar@ics.forth.gr</a>

#### ABSTRACT:

This document provides information regarding the development of the CHIC portal. It is a working document that integrates the manuals, guidelines, templates and any other necessary information to implement the portal software elements (portlets).

#### KEYWORD LIST:

Portal, portlets, development, guidelines.

*The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 600841.*

*The author is solely responsible for its content, it does not represent the opinion of the European Community and the Community is not responsible for any use that might be made of data appearing therein.*

<sup>1</sup> R=Report, P=Prototype, D=Demonstrator, O=Other

<sup>2</sup> PU=Public, PP=Restricted to other programme participants (including the Commission Services), RE=Restricted to a group specified by the consortium (including the Commission Services), CO=Confidential, only for members of the consortium (including the Commission Services)

<b>MODIFICATION CONTROL</b>			
<b>Version</b>	<b>Date</b>	<b>Status</b>	<b>Author</b>
1.0	08/04/2014	Submitted to consortium partners	Giorgos Zacharioudakis

#### List of contributors

- Vangelis Kritsotakis, FORTH
- Ioannis Karatzanis, FORTH
- Giorgos Zacharioudakis, FORTH

## Contents

1	EXECUTIVE SUMMARY.....	6
2	GLOSSARY AND DESCRIPTION OF PORTAL COMPONENTS.....	7
2.1	WEB PORTAL.....	7
2.2	ENTERPRISE PORTAL.....	7
2.3	LIFERAY .....	8
2.4	PORTLETS.....	10
3	HOW TO SET UP A SIMPLE, DEVELOPMENT LIFERAY INSTALLATION FOR EXPERIMENTATION AND DEVELOPMENT.....	13
3.1	STEPS TO FOLLOW IN SHORT.....	13
3.2	REQUIRED STEPS IN DETAIL .....	13
4	CREATING A PORTLET .....	25
4.1	ECLIPSE AND THE LIFERAY IDE.....	25
4.2	DEVELOPING APPS WITH LIFERAY IDE.....	25
4.3	INSTALLING LIFERAY IDE .....	25
4.4	SETTING UP LIFERAY IDE.....	29
4.5	CREATING A NEW LIFERAY PLUGIN PROJECT .....	33
5	MORE INFORMATION.....	43
	APPENDIX 1 – ABBREVIATIONS AND ACRONYMS .....	44

## 1 Executive Summary

The goal of WP10 of the CHIC project is to integrate the various components of the CHIC architecture into one coherent and functional platform. An important part of work will be delivered through task T10.1, the CHIC Portal, which will be the main “point of entrance”, the main user interface of the CHIC platform with the end users. In this task all the portal components that will be developed in the other Work Packages will be integrated into one, common and unified portal.

This document is an Appendix to the deliverable D10.1 “The CHIC Portal”, having as objective to provide all the necessary technical information that is required to install and implement portal software components (portlets). This document is a working document that will be updated when necessary, in order to reflect all the latest information that is required by the software developers.

## 2 Glossary and description of portal components

### 2.1 Web Portal

A web portal<sup>3</sup> is a specially-designed website which combines information from diverse sources in a uniform way. Usually, each information source gets its dedicated area on the page for displaying information (a portlet). Typically, the user can configure which portlets to be displayed into the portal. Variants of portals include Mashup (web application hybrid) and intranet "dashboards" for executives and managers. Very often design emphasis is on a certain "metaphor" for configuring and customizing the presentation of the content and the chosen implementation framework and/or code libraries. In addition, the role of the user in an organization may determine which content can be added to the portal or deleted from the portal configuration.

### 2.2 Enterprise Portal

An enterprise portal<sup>4</sup>, also known as an enterprise information portal (EIP), is a framework for integrating information, people and processes across organizational boundaries. Enterprise portals provide a secure unified access point, often in the form of a web-based user interface, and are designed to aggregate and personalize information through application-specific portlets.

The advantage of enterprise portals is based on the de-centralized content contribution and content management, which implies that the information is always updated. Another distinguishing characteristic that they have is that they cater for customers, vendors and others beyond an organization's boundaries. This differs from a corporate portal which is structured for roles within an organization.



Figure 1 Example of an Enterprise Portal

<sup>3</sup> Web Portal - [http://en.wikipedia.org/wiki/Web\\_portal](http://en.wikipedia.org/wiki/Web_portal)

<sup>4</sup> Enterprise Portal - [http://en.wikipedia.org/wiki/Enterprise\\_portal](http://en.wikipedia.org/wiki/Enterprise_portal)

## 2.3 Liferay

Liferay Portal<sup>5</sup> is a free and open source java based enterprise portal that was initially developed in 2000 in order to provide an enterprise portal solution for non-profit organizations. Currently Liferay is maintained by a company named Liferay, Inc, that has distributed two main type versions:

- **Liferay Portal Community Edition** — This version contains the latest features and support through the active community.
- **Liferay Portal Enterprise Edition** — This is a commercial offering that provides services including updates and full support. This release goes through additional quality assurance cycles and is usually available around 1 or 2 months after the Community Edition and comes under a subscription.

In general Liferay is written in java and runs on all major application servers/servlet containers, databases and operating systems. It provides a robust platform with all the standard applications that someone might need to run on a site, along with an easy to use development framework for new applications or customization. In addition, Liferay Portal is developed using an open source methodology, with solid code base, and has been proved to be reliable and stable in mission critical deployments in diverse industries.

Below are the most important features of the Liferay framework<sup>6</sup>:

- **Simplified UI Development** - Liferay Portal simplifies the development of internal, external, and channel websites--notably those that allow users to login for personalized services or views and those that require a workflow approval process to update content and integrate or aggregate multiple existing services. Liferay Portal provides a single presentation layer for integrating all enterprise systems into a single easy to use interface for end users.
- **Flexible Enterprise Integration Framework** - Liferay Portal is a central presentation layer platform that allows users, administrators and developers to integrate content and services from backend or legacy applications. Liferay supports multiple methods to integrate, including SOAP, REST, RSS, as well as proprietary API's.
- **Out-of-the-box Tools** - Liferay Portal provides more out of the box portlets than any other portal on the market—Choose from over 60 to customize your work environment. Included are Liferay CMS and Liferay Collaboration offering web publishing, content management, collaboration and social networking.
- **Secure Single Sign On (SSO)** - Aggregate and access your content and applications in one place. Liferay Portal can pull all of your different systems together and make them available by logging in just once via the secure SSO.
- **Custom Fields** - Administrators can customize, edit, add, and change user attributes (name, company, date, or any other information that needs to be tracked or logged by your system) directly from Liferay, without needing to modify their database. Liferay 6 extends this support to Pages, Web Content, Blogs Entries, Documents and its folders, images and its folders, bookmarks and its folders, forum messages, calendar events and wiki pages. Liferay also offers a framework so that developers can leverage this functionality in their custom portlets to support custom fields for their own entities.

<sup>5</sup> Liferay - <http://www.liferay.com/>

<sup>6</sup> Liferay portal's features - <http://www.liferay.com/products/liferay-portal/features/portal>



- **Rules Engine Integration** - Advanced personalization via Drools allows for "contextual personalization", the specification of content, functionality, and features based on a user's location, activities, and their other self-defined profile attributes.
- **User Groups, Organizations and Sites** - Liferay users can be intuitively grouped into a hierarchy of "organizations" or cross-organizational "user groups," providing flexibility and ease of administration. For example, members of different geographies such as Americas and EMEA can be grouped into organizations, whereas project based or departmental teams such as a "Website redesign" that cross disciplines can be created as user groups. Liferay provides support for "sites" where both organizations and user groups can be added to a separate web property with its own set of pages, content management system, shared calendar, and authorizations. A user can belong to multiple sites and easily navigate between them.
- **SOA Framework** - Liferay Portal is developed using an open SOA strategy that makes it the choice of enterprises worldwide for enterprise application integration. Integrate your existing HR, Accounting or Sales systems and any other sources of important data.
- **User Personalization** - Depending on what an administrator allows, users can personalize portal pages by adding, removing, positioning content or portlet attributes (e.g., zip code for weather portlet). These pages can be made public (published as a website with a unique friendly URL) or kept private.
- **Role Based Content Delivery** - Portals allow multiple user types to access a single URL and access a unique page view depending on the users role, group, organization or personal preferences. Administrator controlled and user customized, Liferay Portal provides a central platform for determining enterprise content policy, including who can edit and publish content, files, communities, files and applications.
- **Single-Click Configuration** - A fast, responsive interface makes Liferay Portal extremely easy and enjoyable to use for everyone in your organization. Typically time-consuming tasks such as altering a page layout, adding new applications and content, and changing look and feel can all be done in a couple of clicks without ever refreshing the page.
- **Dynamic Drag & Drop** - Liferay Portal was the first portal to offer this feature, allowing users to move different elements around in the portal by simply dragging and dropping them into place.
- **Workflow Framework** - Liferay Portal allows you to incorporate workflow into any of your own custom portlets by leveraging Liferay's workflow APIs. This provides a better end user experience; saves your development team a lot of effort by reusing Liferay's workflow administration tools; and provides seamless integration with all supported workflow engines.
- **User-Driven Workflow & Approval** - Not only is there embedded workflow for content, Liferay Portal allows users to create their own workflow and define the number of approval paths based on their own unique business requirements and operational needs. For example, administrators can now implement an approval process for new document uploads before they appear in the Document Library.
- **Auditing & Performance Monitoring** - To help administrators monitor the portal's performance and better optimize resources, Liferay Portal gives administrators access to key performance statistics (hits/page, avg time/hit, max time per request, and more) for all portlets and portal pages. Meanwhile, our Portal Auditing feature allows administrators to track and manage user activity within the portal.

- **Working from Your Desktop with Liferay Sync** - Working on both the Liferay Portal and your desktop becomes a seamless experience with Liferay Sync. Liferay Sync automatically syncs files from the document library to your local desktop. Changes from local files update automatically in the remote Document Library.
- **Search & Tagging** - Tag web content, documents, message board threads and more to dynamically share important or interesting content with other portal users. Users can then search for relevant information through faceted search, allowing one to filter results by specific criteria within specific portlets, communities, the entire portal and even external integrated applications.
- **Multi-language Support** - International or multi-lingual organizations get out of the box support for 30+ languages. Users can toggle between different language settings with just one click. You can also easily add other languages.
- **OpenSocial** - Support for OpenSocial 1.1 creates new avenues for developers to add social capabilities and dimensions in their websites. With OpenSocial, users can manage and deploy web-based social applications built from gadgets directly to pages and sites.

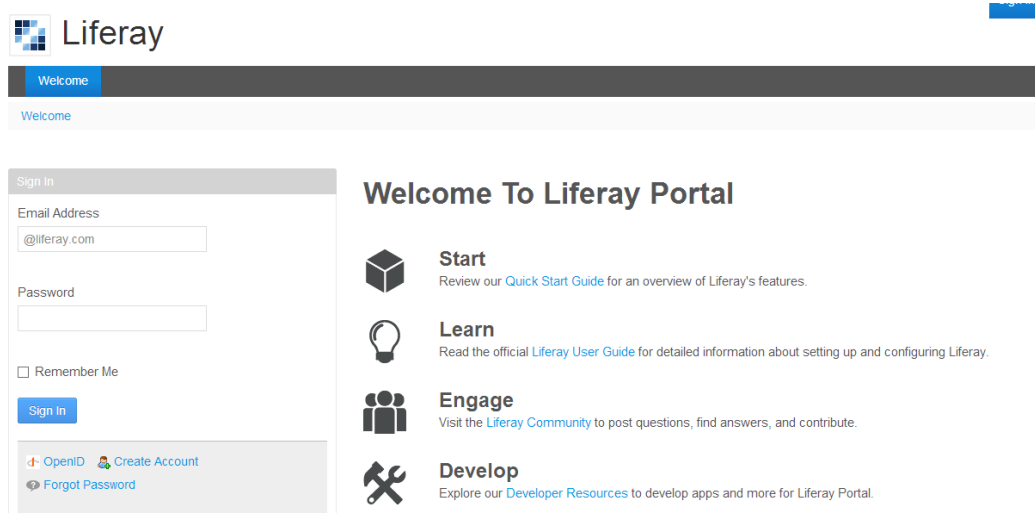



Figure 2 Liferay's welcome page (screenshot)

## 2.4 Portlets

Portlets<sup>7</sup> are pluggable user interface software components that are managed and displayed in a web portal. They produce fragments of markup code (HTML, XHTML, WML) that are grouped into a portal. In order to further understand how portlets look like, we can imagine a portal page being displayed as a collection of non-overlapping windows in a desktop, where each window displays a portlet. Hence a portlet or a collection of portlets resembles a web-based application that is hosted in a portal. Some typical examples of portlet applications are email, weather reports, discussion forums, and news.

<sup>7</sup> Portlet - <http://en.wikipedia.org/wiki/Portlet>


**LIFERAY**  
 Enterprise. Open Source. For Life.

Welcome Joe Bloggs!

[Main](#)
[Add Page](#)

**Message Boards**

[Categories](#)
[My Posts](#)
[Recent Posts](#)
[Statistics](#)

[Add Category](#)

**Blogs**

[Entries](#)
[Categories](#)

[Add Entry](#)

**Polls**

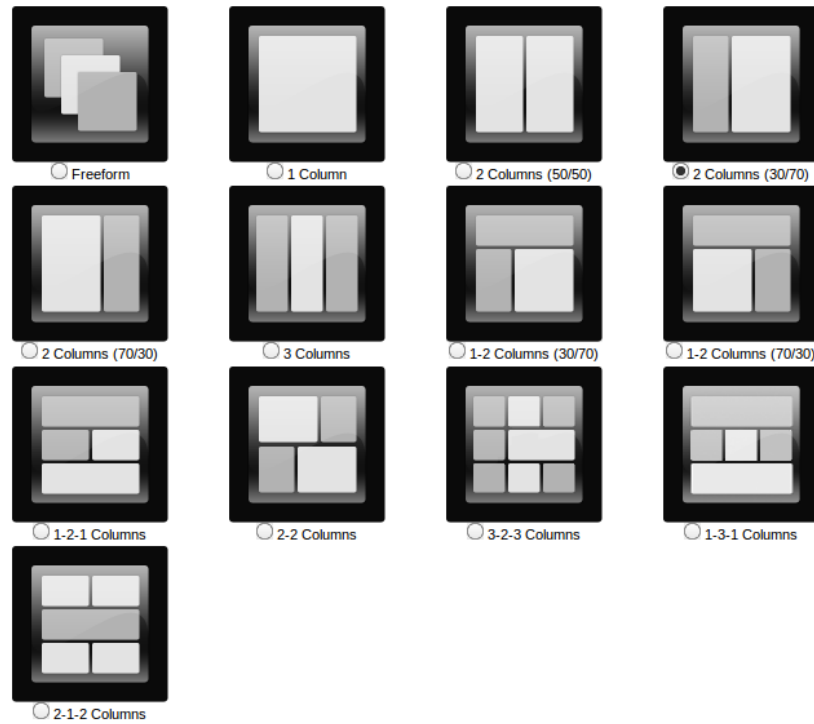
[Add Question](#)

**Currency Converter**

Convert:   To

Currency	British Pound (GBP)	Chinese Yuan (CNY)	Euro (EUR)	Japanese Yen (JPY)	U.S. Dollar (USD)
GBP	1	0.0655	0.6603	0.0042	0.511
CNY	15.2619	1	10.0767	0.0648	7.7984
EUR	1.5146	0.0992	1	0.0064	0.7739
JPY	235.58	15.4359	155.5425	1	120.375
USD	1.9571	0.1282	1.2921	0.0083	1

Figure 3 A page in liferay that includes four portlets (screenshot)



**Figure 4** Available layout of any page in Liferay with respect to the page's portlets

### 3 How to set up a simple, development Liferay installation for experimentation and development.

**Environment:** Windows, Tomcat Server and Liferay 6.2CE with MySQL

**Note:** In the following guidelines we used windows 7 operating system. However the same steps can be applied to any environment. The only things that have to be applied differently are: setting the environment variables and downloading the software that corresponds to the desired OS.

#### 3.1 Steps to follow in short

- Install the **Java JDK** and set the required environment variables (JAVA\_HOME, path)
- Download **MySQL** server and start it
- Download **HeidiSQL** and install it (or any other software for managing MySQL)
- Connect to **MySQL** server and create the Database for Liferay
- Download **Liferay 6.2** Tomcat Bundles
- Configure database settings in **portal-ext.properties** file and place properties file in the **Liferay Home** directory.
- Start the tomcat server and access the **Liferay Portal**
- Complete the required basic **configurations** for the newly installed portal

#### 3.2 Required Steps in detail

##### 3.2.1 Install JDK and set the required environment variables (JAVA\_HOME, path) (Skip this step if you have already a working JDK environment)

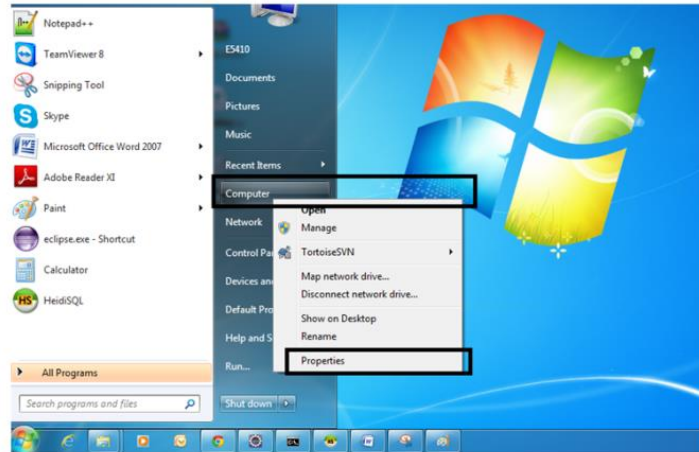
- Download *Java SDK 1.6* from the following location:  
<http://www.oracle.com/technetwork/java/javasebusiness/downloads/java-archive-downloads-javase6-419409.html>

For current Java releases, please consult the Oracle Software Download page.

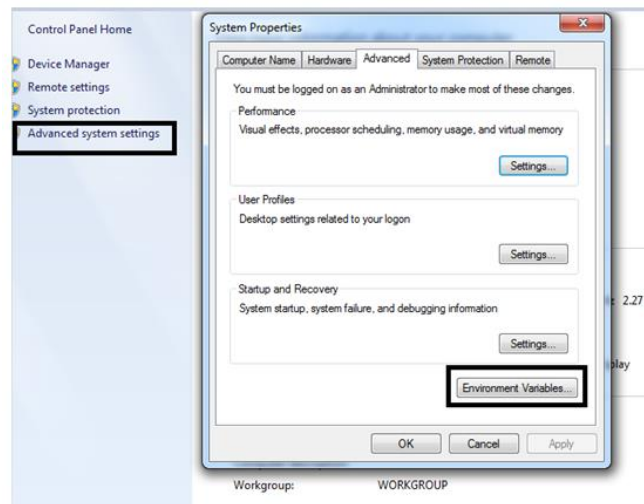
For more information on the transition of products from the legacy Sun download system to the Oracle Technology Network, visit the SDLC Decommission page announcement.

- Java SE Development Kit 6u45
- Java SE Runtime Environment 6u45
- **Java SE Development Kit 6u43**
- Java SE Runtime Environment 6u43
- Java SE Development Kit 6u41
- Java SE Runtime Environment 6u41
- Java SE Development Kit 6u39
- Java SE Runtime Environment 6u39
- Java SE Development Kit 6u38

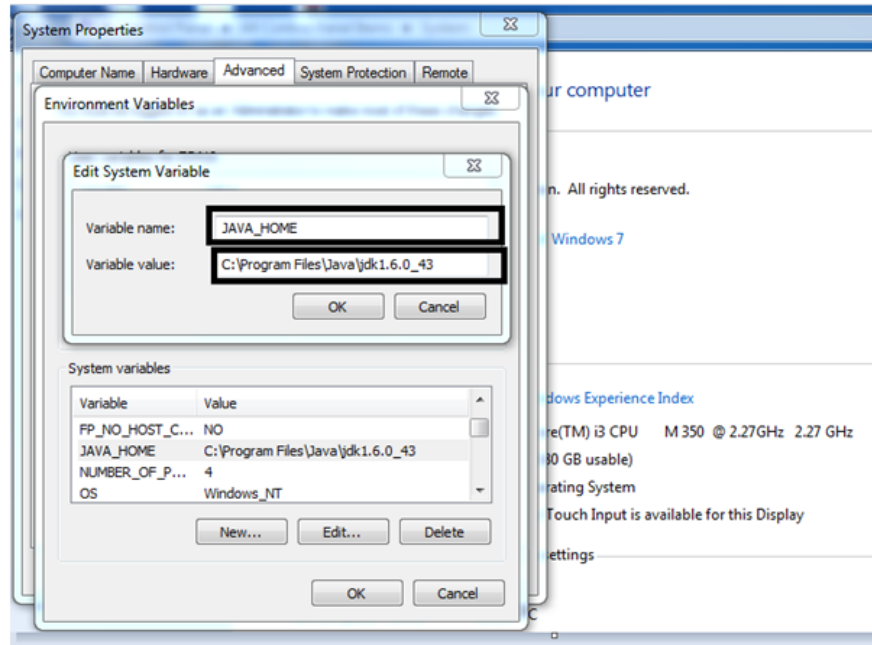
- Set the required environment variables:
  - Go to computer properties



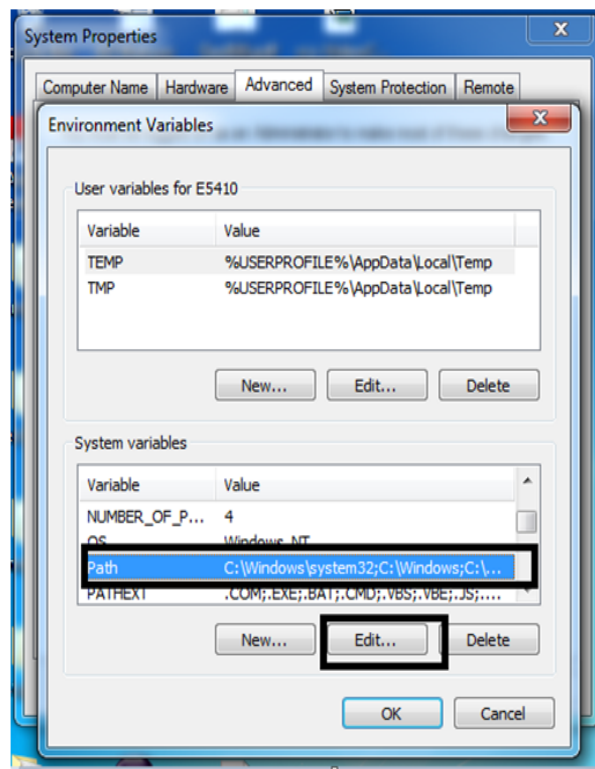
- Access the System's environment variables:

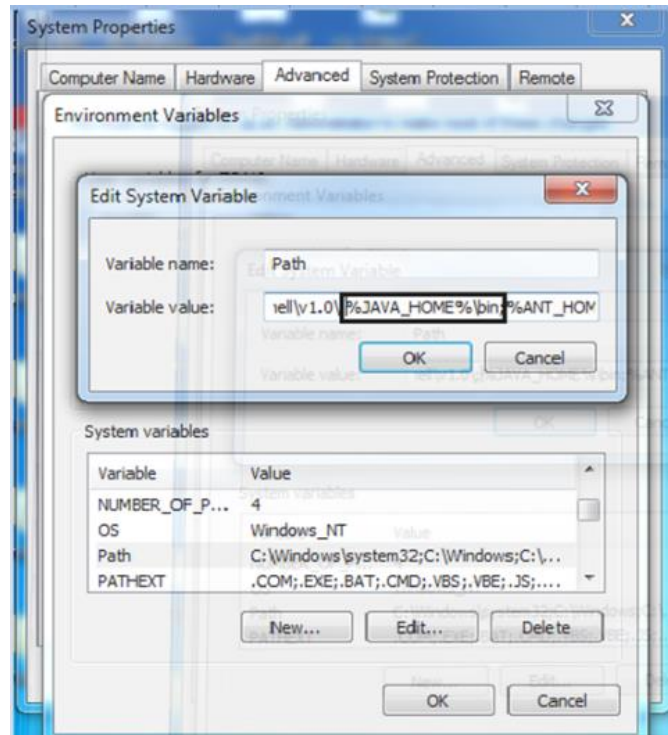


- Create the **"JAVA\_HOME"** variable (if it doesn't exist) as follows:
  - Variable Name: **JAVA\_HOME**
  - Variable Value: The path where java is installed (*JDK home directory*)



- Edit the system path variable and specify the java path:
  - Select the “**Path**” variable
  - Edit it
  - Add `;%JAVA_HOME%/bin;` to the existed path value (each value is separated by semicolon).





### 3.2.2 Download MySQL server and Start it (*Skip this step if you have already a working MySQL installation*)

- Download **MySQL sever 5.1** or **5.5** from following location:  
<http://dev.mysql.com/downloads/mysql/5.1.html#downloads>

Other Downloads:		
Windows (x86, 32-bit), ZIP Archive	5.1.70	115.8M
<a href="#">(mysql-noinstall-5.1.70-win32.zip)</a> <a href="#">MDS: 8ec8ce6dc1b76ae5117912eae3d17d15</a>   <a href="#">Signature</a>		
Windows (x86, 64-bit), ZIP Archive	5.1.70	119.1M
<a href="#">(mysql-noinstall-5.1.70-win64.zip)</a> <a href="#">MDS: 92ac90da15e1ca3e85d3b2289a0d512</a>   <a href="#">Signature</a>		

- Extract the *zip* file in your desired location
- Start *MySQL* database server
  - Open command prompt and go *MySQL* bin directory (*mysql-5.1.70-winx64/bin*) then use following command to start *MySQL* database server: *mysqld --console*

```

C:\Windows\system32\cmd.exe - mysql - console
D:\liferay-spring-mysql-workspace->cd mysql-5.1.70-winx64
D:\liferay-spring-mysql-workspace\mysql-5.1.70-winx64>cd bin
D:\liferay-spring-mysql-workspace\mysql-5.1.70-winx64\bin>mysql --console
131210 9:09:03 [Note] Plugin 'FEDERATED' is disabled.
131210 9:09:03 InnoDB: Initializing buffer pool, size = 8.0M
131210 9:09:03 InnoDB: Completed initialization of buffer pool
InnoDB: The log sequence number in ibdata files does not match
InnoDB: the log sequence number in the ib_logfiles!
131210 9:09:03 InnoDB: Database was not shut down normally!
InnoDB: Starting crash recovery.
InnoDB: Reading tablespace information from the .ibd files...
InnoDB: Restoring possible half-written data pages from the doublewrite
InnoDB: buffer...
131210 9:09:04 InnoDB: Started; log sequence number 0 261871982
131210 9:09:04 [Note] Event Scheduler: Loaded 0 events
131210 9:09:04 [Note] mysqld: ready for connections.
Version: '5.1.70-community' socket: '' port: 3306 MySQL Community Server (GPL)

```



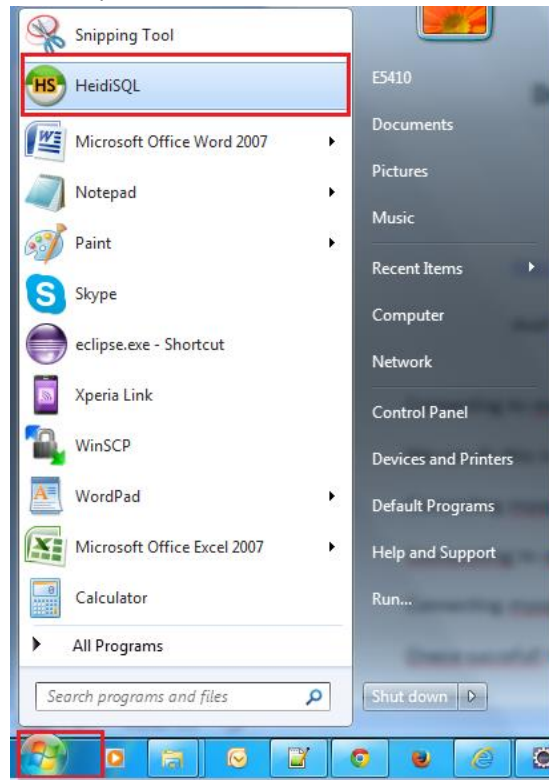
### 3.2.3 Download HeidiSQL and install it (*or use any other alternative MySQL management software*)

- Download **HeidiSQL** to access *MySQL* server databases. This is a lightweight tool that provides a simple user interface for accessing the database and managing the tables.
  - Download *HeidiSQL* from following location  
[http://www.heidisql.com/installers/HeidiSQL\\_8.0.0.4396\\_Setup.exe](http://www.heidisql.com/installers/HeidiSQL_8.0.0.4396_Setup.exe)

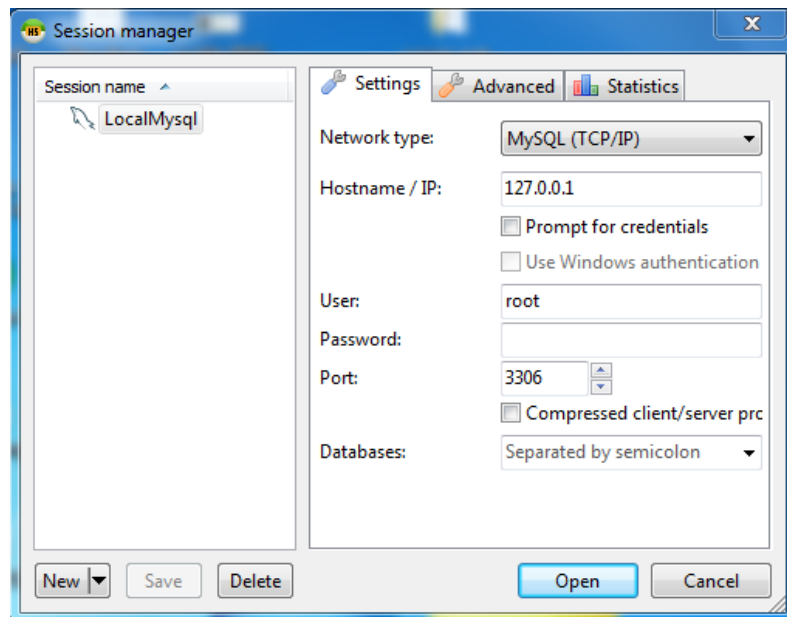
#### Download HeidiSQL 8.0

- [Installer](#)
- [Portable](#)
- [Sourcecode](#)

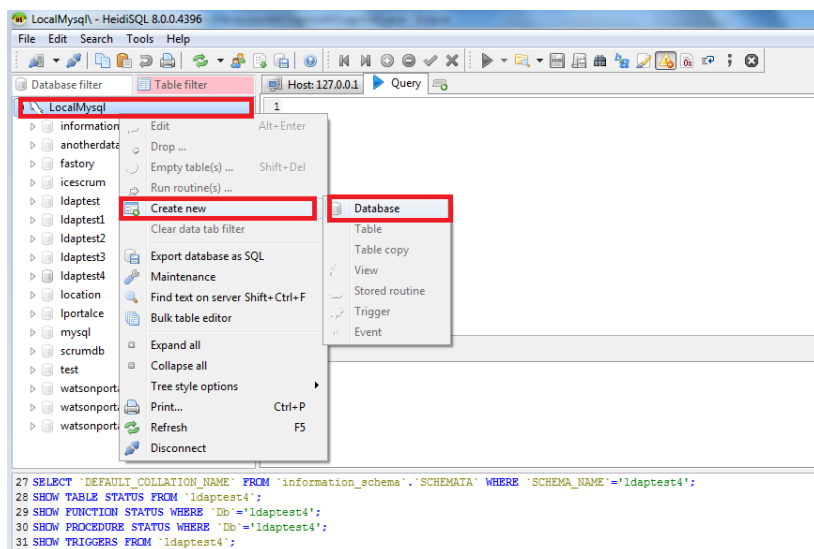
- Install it by running the file and following the steps
- Connect to *MySQL* server and create the Database for Liferay  
 We can do this in two ways:
  1. Connecting *MySQL* from *HeidiSQL* and create Liferay Data base
    - Once successful installation of *HeidiSQL*, go to windows start and click on *HeidiSQL*



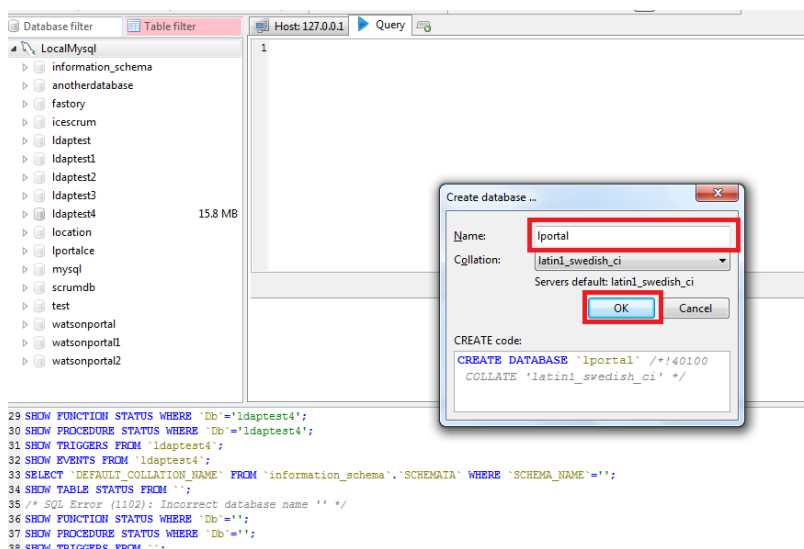
- Click on New and enter the following details:  
**Host Name:** 127.0.0.1 Or localhost  
**User:** root  
**Password:** (leave that empty)



- Create the *lportal* database for the *Liferay Portal*. Right click on root node and select create new database



- Name the database "*lportal*" and click OK

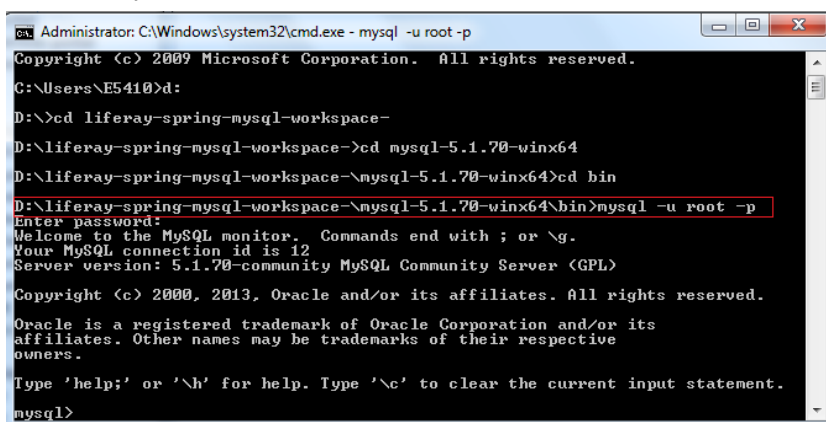


2. Connect to *MySQL* from *MS DOS Command Prompt* and create the data base for liferay.

This is another way to connect to *MySQL* server.

- Open *MS DOS Command Prompt*
- Navigate to the *MySQL* server *bin* directory
- Enter the following command: `mysql -u root -p`
- Press enter

It will ask for password; however we don't need to enter one, since we haven't set any



- create the data base by entering the following command:  
`create database lportal;`

```
Administrator: C:\Windows\system32\cmd.exe - mysql -u root -p
D:\liferay-spring-mysql-workspace\mysql-5.1.70-win64>cd bin
D:\liferay-spring-mysql-workspace\mysql-5.1.70-win64\bin>mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 5.1.70-community MySQL Community Server (GPL)

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

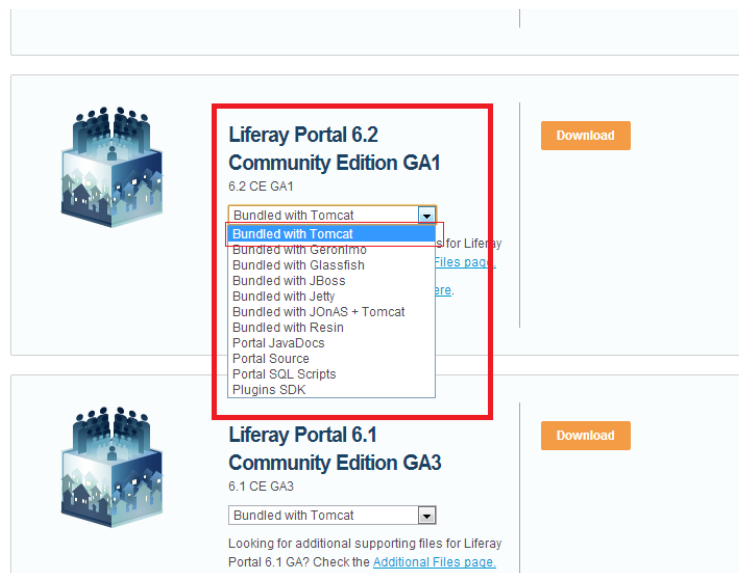
mysql> create database meera
->
Query OK, 1 row affected (0.00 sec)

mysql> create database lportal
->
Query OK, 1 row affected (0.00 sec)

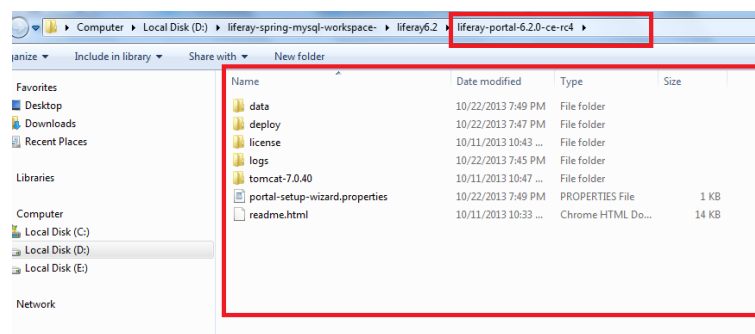
mysql>
```

### 3.2.4 Download Liferay 6.2 Tomcat bundle.

- Download **Liferay 6.2 Tomcat bundle** from the following location:  
<https://www.liferay.com/downloads/liferay-portal/available-releases>



- Extract the **Liferay 6.2 Tomcat Bundle Zip** file in your desired location
- Open to extracted folder (we should call this folder “**Liferay Home**”)



### 3.2.5 Configure database setting in portal-ext.properties file and place properties file in Liferay home directory

- Create a file in the *Liferay Home* folder and name it **portal-ext.properties**

- Add the following *MySQL* configurations:

```
jdbc.default.driverClassName=com.mysql.jdbc.Driver
```

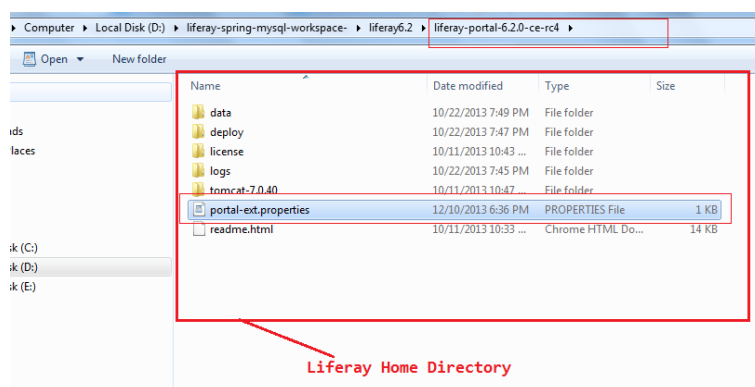
```
jdbc.default.url=jdbc:mysql://localhost:3306/lportal?useUnicode=true&characterEncoding=UTF-
```

```
8&useFastDateParsing=false
```

```
jdbc.default.username=root
```

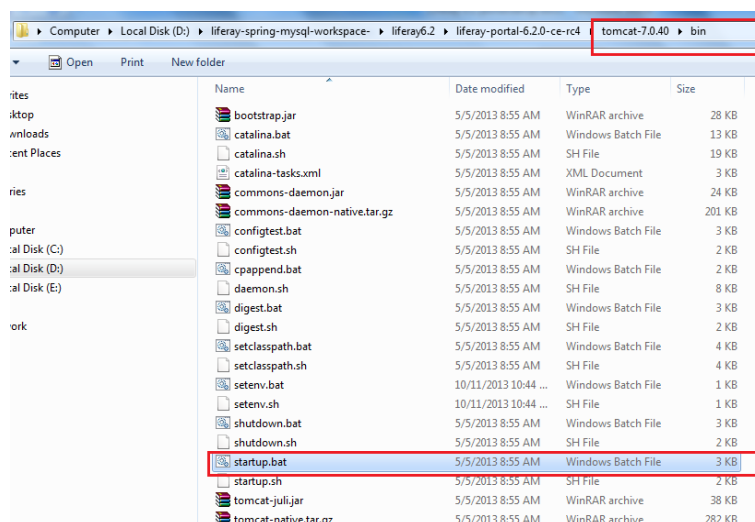
```
jdbc.default.password=
```

(Previously we created a database named “lportal” in *MySQL*. If that database was named differently, it should be specified respectively in the “portal-ext.properties” file)



### 3.2.6 Start Tomcat server and access Liferay Portal

- Go to the *Liferay Home* folder and open tomcat’s (tomcat-7.0.40) directory
- Start tomcat by executing the file *bin/startup.bat*



A command prompt console will open, showing log information of liferay. We should know that the server has started as soon as we see “Server startup in xxxxx ms”

```

Dec 11, 2013 1:47:07 AM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory D:\liferay-spring-mysql-workspace-\lif
eray6.2\liferay-portal-6.2.0-ce-rc4\tomcat-7.0.40\webapps\calendar-portlet
Dec 11, 2013 1:47:18 AM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory D:\liferay-spring-mysql-workspace-\lif
eray6.2\liferay-portal-6.2.0-ce-rc4\tomcat-7.0.40\webapps\marketplace-portlet
Dec 11, 2013 1:47:23 AM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory D:\liferay-spring-mysql-workspace-\lif
eray6.2\liferay-portal-6.2.0-ce-rc4\tomcat-7.0.40\webapps\resources-importer-web
Dec 11, 2013 1:47:25 AM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory D:\liferay-spring-mysql-workspace-\lif
eray6.2\liferay-portal-6.2.0-ce-rc4\tomcat-7.0.40\webapps\welcome-theme
Dec 11, 2013 1:47:26 AM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["http-bio-8080"]
Dec 11, 2013 1:47:26 AM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["ajp-bio-8009"]
Dec 11, 2013 1:47:26 AM org.apache.catalina.startup.Catalina start
INFO: Server startup in 234494 ms
Dec 11, 2013 1:47:48,377 ERROR [http-bio-8080-exec-3][AutoLoginFilter:87] Current URL / gene
rates exception: com.liferay.portal.kernel.exception.SystemException: com.liferay
y.util.EncryptorException: com.liferay.util.EncryptorException: javax.crypto.Ill
legalBlockSizeException: Input length must be multiple of 16 when decrypting with
padded cipher
  
```

- Go to the following URL (by using any browser) in order to access Liferay  
<http://localhost:8080/>

(**Note:** Tomcat server runs by default on the port 8080. In case this port is engaged by some other service, an address bind exception will be thrown. In that case you should use another port for your Tomcat server).

### 3.2.7 Complete required basic configurations for liferay portal

Once Liferay portal is accessed, the basic configuration page will be displayed

- Fill in all the required information. Some short explanation of what needs to be entered is given here:

**Portal Name:** anything you like

**Default Language:** Choice from the select box

**Admin First Name:** Anything you like

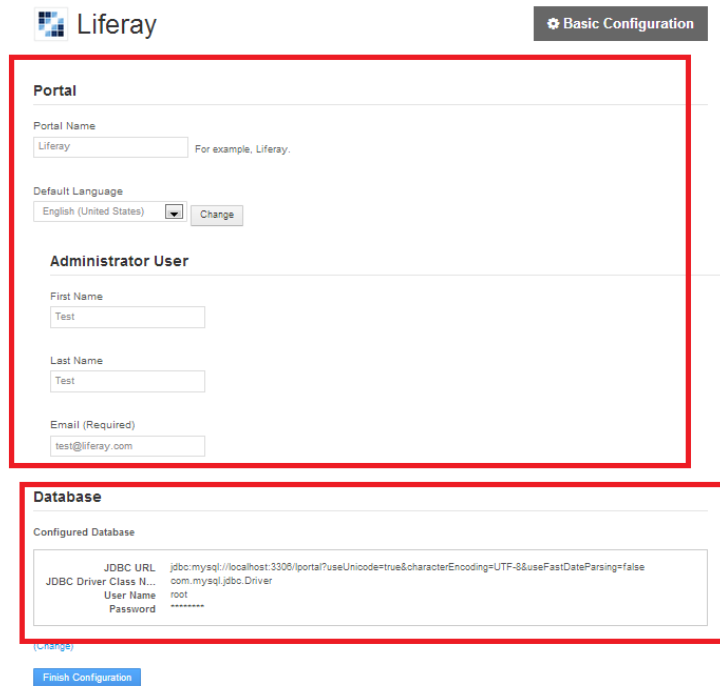
**Admin Last Name:** Anything you like

**Admin Email:** Anything you like (the default is [test@liferay.com](mailto:test@liferay.com))

(**Note:** The Admin’s Email will be used for logging into the portal).

In the data base section you can find *lportal*’s configurations.

- Click the *Finish Configuration* button in order to save the applied configuration settings.



**Liferay** Basic Configuration

**Portal**

Portal Name  
Liferay For example, Liferay.

Default Language  
English (United States) Change

**Administrator User**

First Name  
Test

Last Name  
Test

Email (Required)  
test@liferay.com

**Database**


Configured Database

JDBC URL	jdbc:mysql://localhost:3306/lportal?useUnicode=true&characterEncoding=UTF-8&useFastDateParsing=false
JDBC Driver Class Name	com.mysql.jdbc.Driver
User Name	root
Password	*****

(Changes)

Finish Configuration

All saved configuration settings are stored in the *portal-setup-wizard.properties* file located into the *Liferay Home* directory.



**Liferay** Basic Configuration

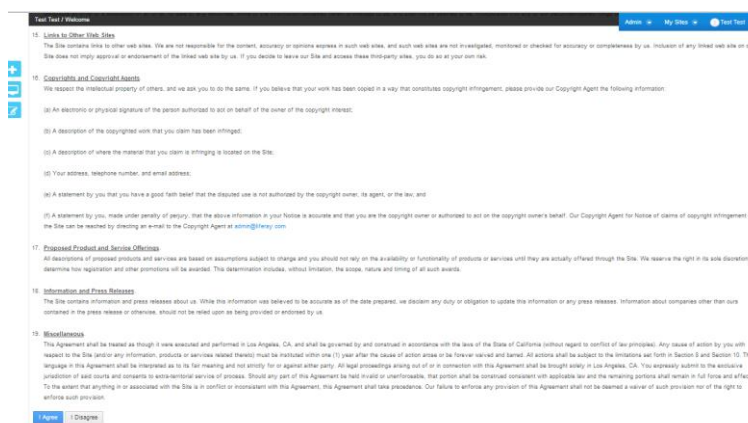
Your configuration was saved successfully.

The configuration was saved in **D:/liferay-spring-mysql-workspace-/liferay6.2/liferay-portal-6.2.0-ce-rc4/portal-setup-wizard.properties**

Go to My Portal

Powered By Liferay

- Press the “Go to My Portal” button in order to proceed. That will navigate us to the “Liferay Terms and condition” page, where we simply click on the “I Agree” button located at the bottom of the page



Test Portal - Welcome

Admin My Sites Text Text

15. **Links to Other Web Sites**  
The Site contains links to other web sites. We are not responsible for the content, accuracy or opinions expressed in such web sites, and such web sites are not investigated, monitored or checked for accuracy or completeness by us. Inclusion of any linked web site on our Site does not imply approval or endorsement of the linked web site by us. If you decide to leave our Site and access these third-party sites, you do so at your own risk.

16. **Copyrights and Copyright Infringement**  
We respect the intellectual property of others, and we ask you to do the same. If you believe that your work has been copied in a way that constitutes copyright infringement, please provide our Copyright Agent the following information:  
(a) An electronic or physical signature of the person authorized to sue on behalf of the owner of the copyright interest;  
(b) A description of the copyrighted work that you claim has been infringed;  
(c) A description of where the material that you claim is infringing is located on the Site;  
(d) Your address, telephone number, and email address;  
(e) A statement by you that you have a good faith belief that the disputed use is not authorized by the copyright owner, its agent, or the law; and  
(f) A statement by you, made under penalty of perjury, that the above information in your notice is accurate and that you are the copyright owner or authorized to act on the copyright owner's behalf. Our Copyright Agent for notice of claims of copyright infringement on the Site can be reached by emailing an e-mail to the Copyright Agent at [esom@liferay.com](mailto:esom@liferay.com).

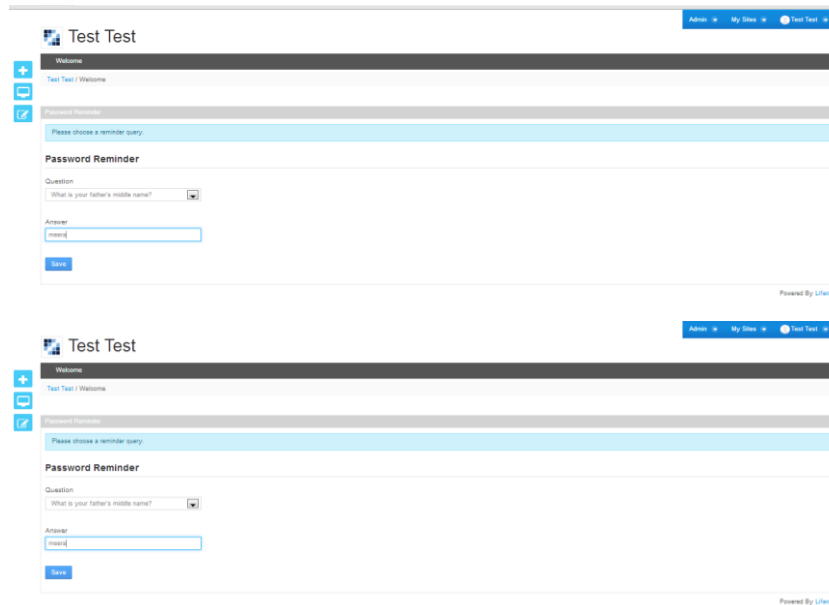
17. **Processed Product and Service Offerings**  
All descriptions of processed products and services are based on assumptions subject to change and you should not rely on the availability or functionality of products or services until they are actually offered through the Site. We reserve the right to discontinue or alter the products and services at any time without notice, and we are not responsible for any loss or damage resulting from the discontinuation or alteration of any product or service.

18. **Information and Press Releases**  
The Site contains information and press releases about us. While this information was believed to be accurate as of the date prepared, we disclaim any duty or obligation to update this information or any press releases. Information about companies other than ours contained in the press release or otherwise, should not be relied upon as being provided or endorsed by us.

19. **Waiver of Rights**  
This Agreement shall be treated as though it were executed and performed in Los Angeles, CA, and shall be governed by and construed in accordance with the laws of the State of California (without regard to conflict of law principles). Any issue of action by you with respect to this Agreement shall be resolved as to its fair meaning and not strictly for or against either party. All legal proceedings arising out of or in connection with this Agreement shall be brought solely in Los Angeles, CA. You expressly submit to the exclusive jurisdiction of said courts and consents to extrajurisdictional service of process. Should any part of this Agreement be held invalid or unenforceable, that portion shall be construed consistent with applicable law and the remaining portions shall remain in full force and effect. To the extent that anything in or associated with the Site is in conflict or inconsistent with this Agreement, this Agreement shall take precedence. Our failure to enforce any provision of this Agreement shall not be deemed a waiver of such provision nor of the right to enforce such provision.

Agree Disagree

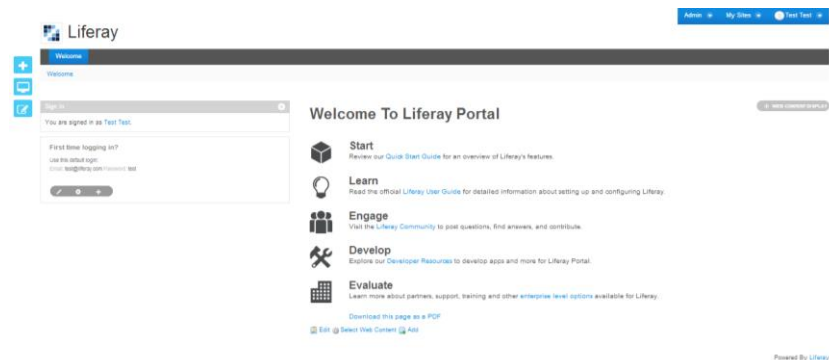
- Enter a new password for the administrator in order to use it during the login procedure



The image shows two screenshots of a Liferay Portal page titled 'Test Test'. The page has a sidebar with navigation icons and a main content area. The main content area contains a 'Password Reminder' form. The form has a section titled 'Please choose a reminder query:' followed by a 'Question' field with a dropdown menu. The question is 'What is your father's middle name?'. Below the question is an 'Answer' field with the text 'mario' entered. A 'Save' button is at the bottom right of the form. The page is powered by Liferay.

(**Note:** Admin's email and password are the portal administrator's credentials. The portal administrator has full control over the portal.)

As soon as the whole procedure has been completed, we will be navigated to the portal's *welcome page*



The image shows a screenshot of the Liferay Portal 'Welcome To Liferay Portal' page. The page has a sidebar with navigation icons and a main content area. The main content area contains a 'Welcome To Liferay Portal' message and a list of links: Start, Learn, Engage, Develop, and Evaluate. The page is powered by Liferay.

You can find liferay's user guide here: <https://www.liferay.com/documentation/liferay-portal/6.2/user-guide>



## 4 Creating a portlet

### 4.1 *Eclipse and the Liferay IDE*

**Eclipse** is the most popular and well known Java IDE and it provides a wide variety of features. **Liferay IDE** is a plugin for Eclipse that extends its functionality to facilitate developing all types of Liferay plugins. Liferay IDE uses the Plugins SDK underneath, but you don't need to know the SDK unless you're performing an advanced operation not directly supported by Liferay IDE. In this document we will focus to the Liferay IDE plugin for Eclipse. To develop applications for Liferay Portal Enterprise Edition (EE), which is out of scope for this document, use Liferay Developer Studio which extends Liferay IDE.

### 4.2 *Developing Apps with Liferay IDE*

Liferay IDE is an extension for Eclipse IDE and supports development of plugin projects for the Liferay Portal platform. You can install Liferay IDE as a set of Eclipse plugins from an update site. The latest version of Liferay IDE supports development of portlets, hooks, layout templates, themes, and Ext plugins. To use Liferay IDE, you need the Eclipse Java EE developer package using Indigo or a later version.

In this section we'll show you how to install Liferay IDE, set up projects for your applications, and deploy them to your portal. We'll get you started with the basics of developing your Liferay application in Liferay IDE.

### 4.3 *Installing Liferay IDE*

To install and set up Liferay IDE, follow the instructions below.

#### 4.3.1 Requirements

Make sure you have a supported Java JRE and Eclipse release:

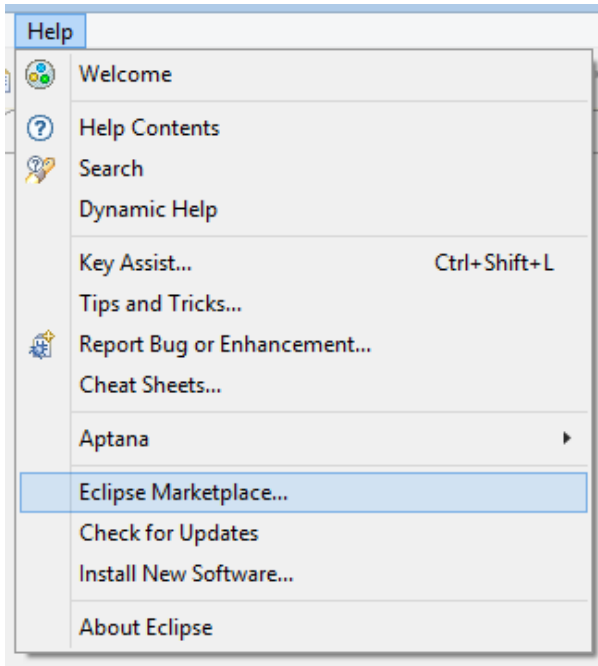
1. Java 6.0 JRE or greater.
2. Liferay IDE must be run in one of the following Eclipse releases:
  - a. Eclipse Kepler Java EE (4.3.x)
  - b. Eclipse Juno Java EE (4.2.x)
  - c. Eclipse Indigo Java EE (3.7.x)

#### 4.3.2 Installing Liferay IDE using Eclipse Marketplace

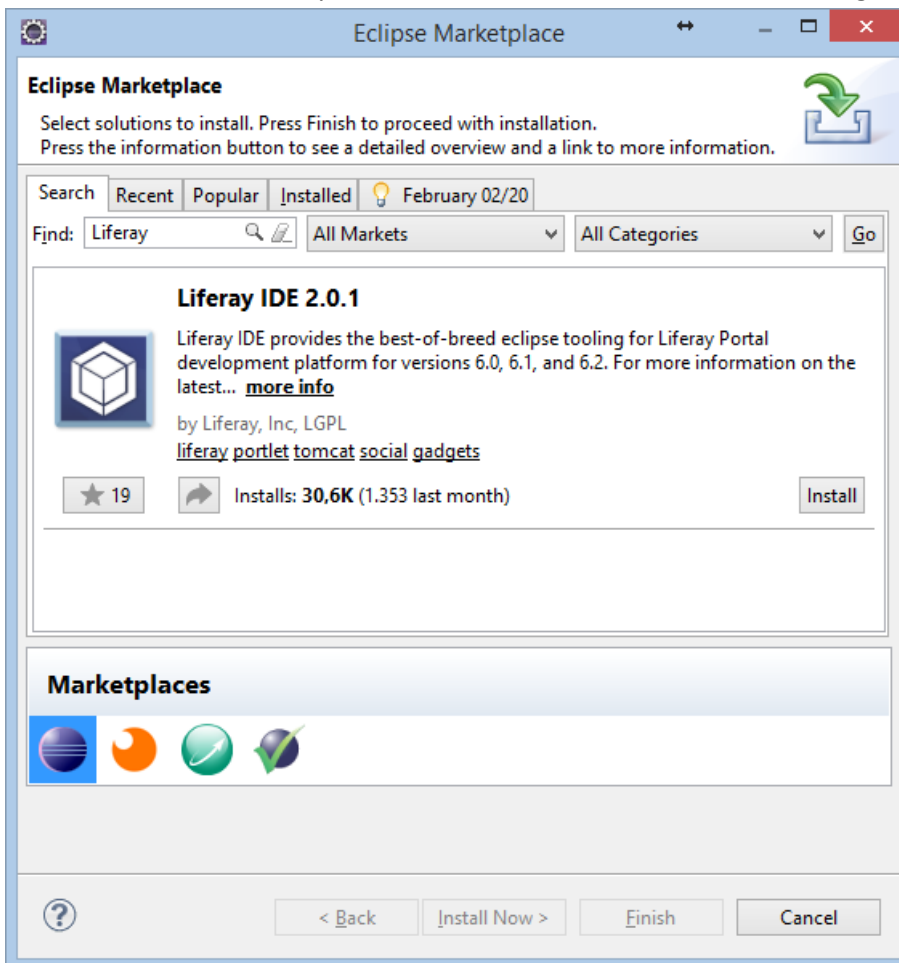
To install Liferay IDE through the Eclipse Marketplace, follow these steps:

1. Install Eclipse Kepler, Juno, or Indigo from the Eclipse website.
2. Run the Eclipse executable file (e.g., eclipse.exe).

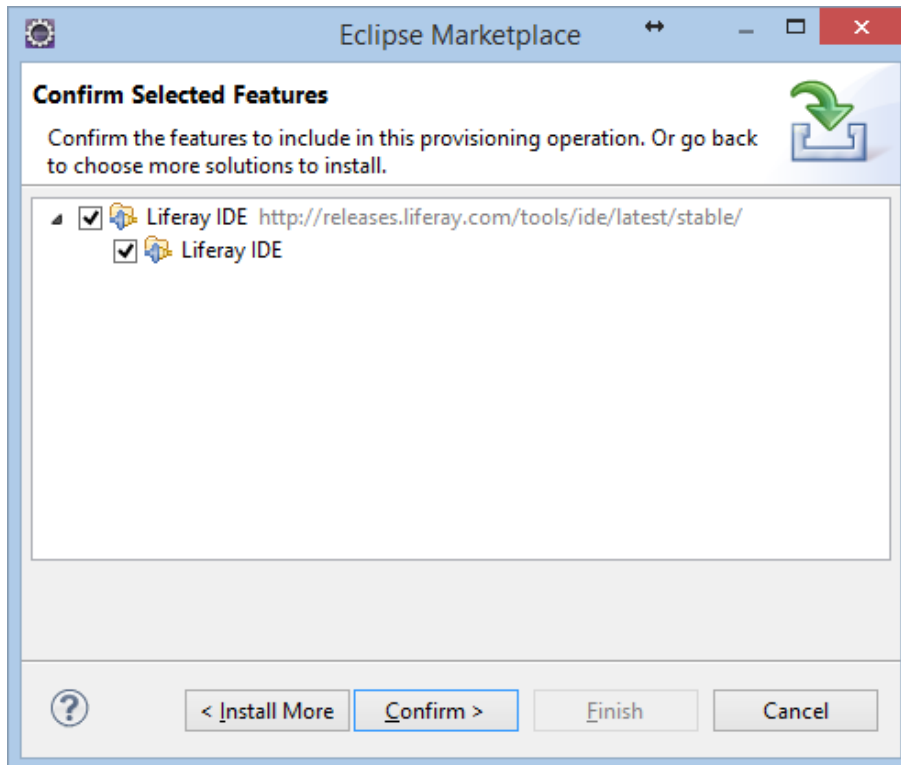
- When Eclipse opens, go to **Help → Eclipse Marketplace...**



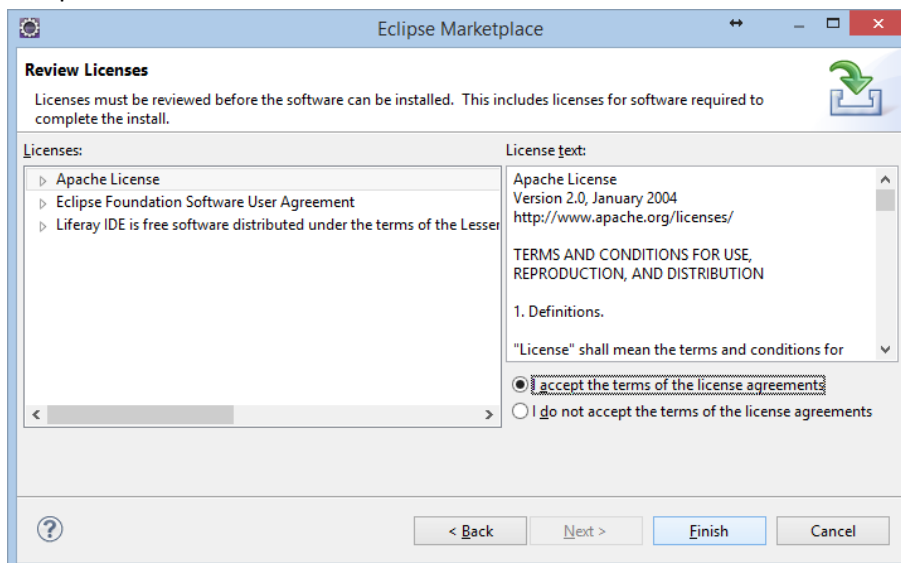
- In the Find field, enter Liferay and press Enter. From the filtered results the first should be the latest version of Liferay IDE. Click the Install button at the bottom right of the result.



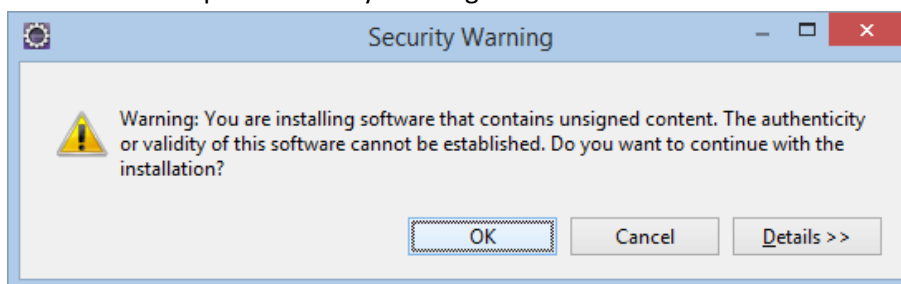
5. Liferay IDE features should be detected. Select them and click **Next**.



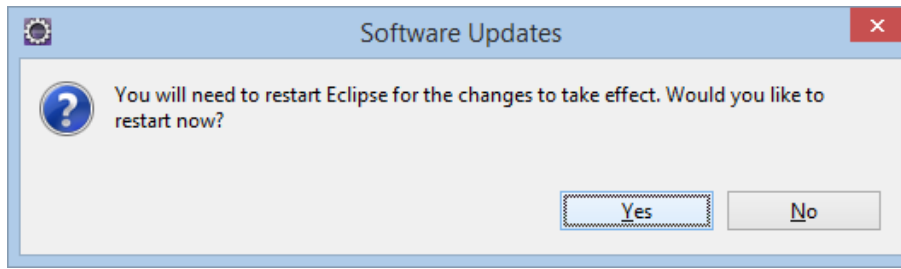
6. After calculating dependencies, click **Next**, accept the license agreement, and click **Finish** to complete the installation.



7. Press OK to accept the security warning.



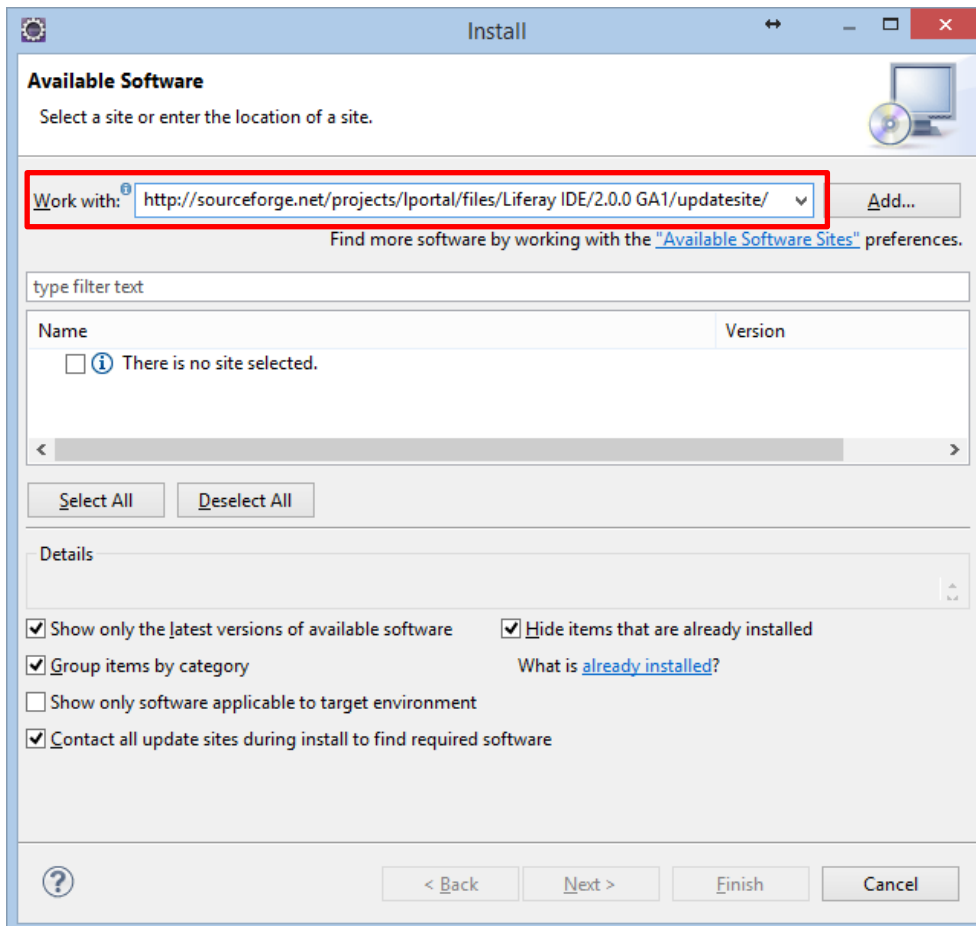
8. Restart Eclipse to verify that Liferay IDE is properly installed.



### 4.3.3 Alternative Installation using an Eclipse update URL

To install Liferay IDE and specify an Eclipse update URL, follow these steps:

9. Install Eclipse Kepler, Juno, or Indigo from the Eclipse website.
10. Run the Eclipse executable file (e.g., eclipse.exe).
11. When Eclipse opens, go to **Help → Install New Software....**
12. In the Work with field, enter the update site URL  
**[http://sourceforge.net/projects/lportal/files/Liferay IDE/2.0.0 GA1/updatesite/](http://sourceforge.net/projects/lportal/files/Liferay%20IDE/2.0.0%20GA1/updatesite/)**  
and press Enter.



13. Liferay IDE features should be detected. Select them and click **Next**.

14. After calculating dependencies, click **Next**, **accept the license agreement**, and click **Finish** to complete the installation.
15. **Restart** Eclipse to verify that Liferay IDE is properly installed.
16. After restarting Eclipse, go to **Help → About Eclipse**; if you see a Liferay IDE icon badge as in the screenshot below, it's properly installed.



## 4.4 Setting up Liferay IDE

Now that you have Liferay IDE installed, you need to perform some basic setup. This section describes the setup steps to perform so you can develop your Liferay portal and test your customizations.

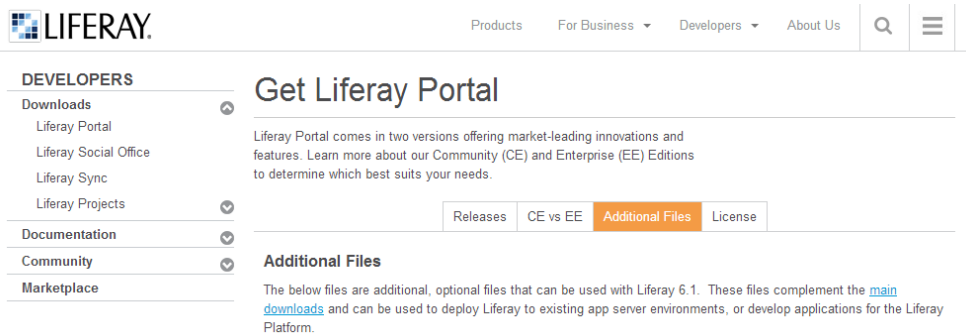
Before setting up Liferay IDE, let's make sure you have all the appropriate software packages installed.

### 4.4.1 Requirements

Before setting up Liferay IDE, you need to have appropriate versions of Liferay Portal, Liferay Plugins SDK and/or Maven, and Eclipse. Make sure you satisfy these requirements before proceeding:

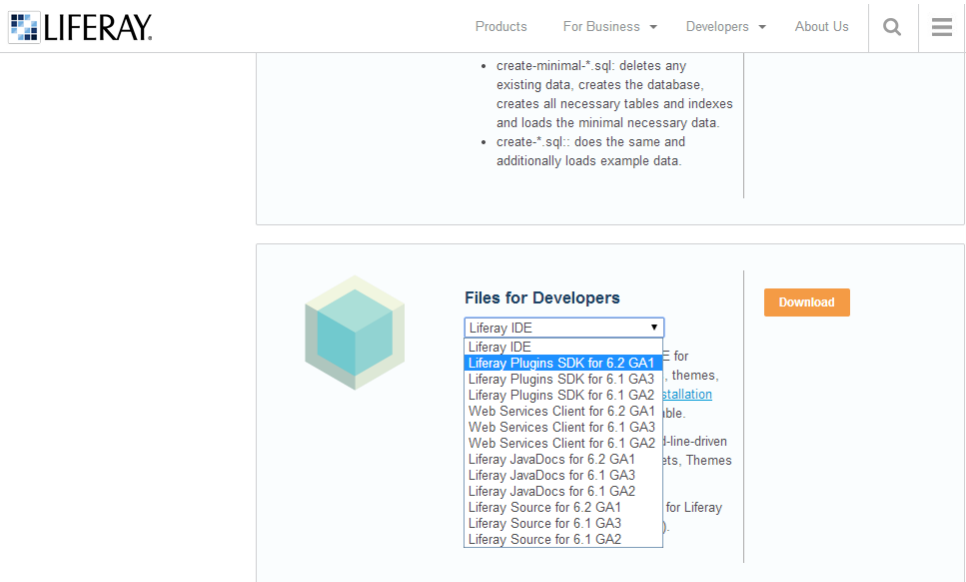
1. Liferay Portal 6.0.5 or greater is installed (downloaded, unzipped and configured so that it can be operational).
2. Liferay Plugins SDK 6.0.5 or greater is downloaded and unzipped, and/or any version of Maven is installed. If you're using the Plugins SDK, make sure the Plugins SDK version matches the Liferay Portal version.
  - a. In order to install the Liferay SDK, you have to visit the download section of the Liferay Official Site, and from the Developers area navigate to the third tab named

"Additional Files".



The screenshot shows the Liferay Portal website. The top navigation bar includes 'Products', 'For Business', 'Developers', and 'About Us'. The left sidebar has a 'DEVELOPERS' section with links to 'Downloads', 'Documentation', 'Community', and 'Marketplace'. The main content area is titled 'Get Liferay Portal' and contains a section for 'Additional Files'. This section includes a list of files with descriptions and a 'Download' button.

- b. Scroll to the bottom of the page of the additional files, and you will see a block called "Files for Developers". From the dropdown select the appropriate version of the Liferay Plugins SDK, for your installation of Liferay and finally click on the Download button.



The screenshot shows the 'Files for Developers' section of the Liferay Portal website. It features a dropdown menu with various Liferay SDKs and a 'Download' button. The dropdown menu is open, showing options like 'Liferay IDE', 'Liferay Plugins SDK for 6.2 GA1', and 'Liferay Plugins SDK for 6.1 GA3'.

3. You've installed an appropriate Eclipse IDE version for Java EE Development, and the Liferay IDE extension—see the Installation section if you haven't already done this.

#### 4.4.2 Liferay Plugins SDK Setup

Before you begin creating new Liferay plugin projects, a supported Liferay Plugins SDK and/or Maven installation and Liferay Portal must be installed and configured in your Liferay IDE. If you're thinking, "Wait a second! You told me earlier that the Plugins SDK and Maven could be used without Liferay IDE!", then you're right. In the second half of this chapter, we'll explain how to use the Plugins SDK and Maven on its own, with a text editor. Here, we explain the easiest way to use the Plugins SDK: by running it from Liferay IDE.

1. In Eclipse, open the **Installed Plugin SDKs** dialog box—from your **Windows** dropdown menu, click **Preferences → Liferay → Installed Plugin SDKs**.
2. Click **Add** to bring up the Add SDK Dialog.

3. Browse to your Plugins SDK installation. The default name is the directory name; you can change it if you want.
4. Select **OK** and verify that your SDK was added to the list of Installed Liferay Plugin SDKs.

#### 4.4.3 Liferay Portal Runtime and Server Setup

You can run Liferay on any application server supported by Liferay Portal. Here, for demonstration purposes, we'll set up our Liferay runtime on the Tomcat application server. The steps you'd follow for any other supported application server would be similar. For a list of Liferay bundles with other application servers, please visit Liferay's Downloads page. For instructions on installing Liferay manually on other application servers, please refer to the Installation and Setup chapter of Using Liferay Portal 6.2.

1. In Eclipse, open the **Server Runtime Environments** dialog box—go to **Window → Preferences → Server → Runtime Environments**.

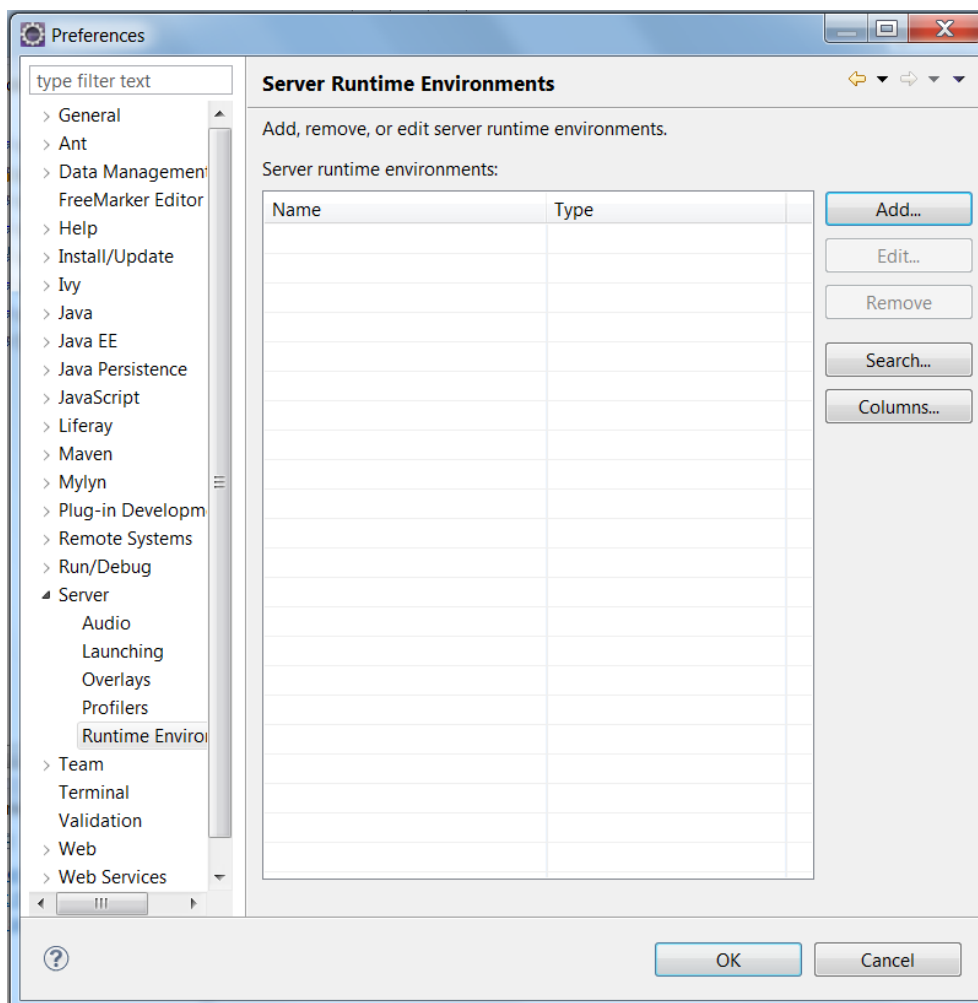


Figure 5 Liferay IDE provides wizards for creating new Liferay server runtime environments

2. Click **Add** to add a new Liferay runtime; find **Liferay v6.2** (Tomcat 7) under the **Liferay, Inc.** category and click **Next**.
3. Click **Browse** and select your liferay-portal-6.2.x directory.

4. If you've selected the Liferay portal directory and a bundle JRE is present, it is automatically selected as the server's launch JRE. If no JRE bundle is present, then you must select the JRE to use for launch by clicking Installed JREs....

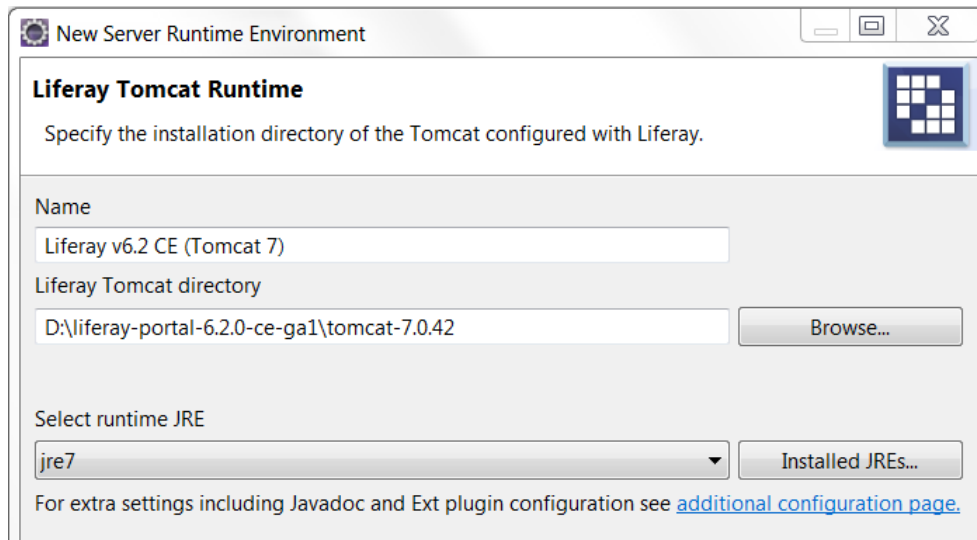


Figure 6 If you have multiple JREs installed on your system, choose the one which should run Liferay

5. Click **Finish**; you should see your Liferay portal runtime listed in **Preferences → Server & Runtime Environments**.
6. Click **OK** to save your runtime preferences.
7. If you haven't created a server, create one now from the Servers view in Liferay IDE; then you can test the server.

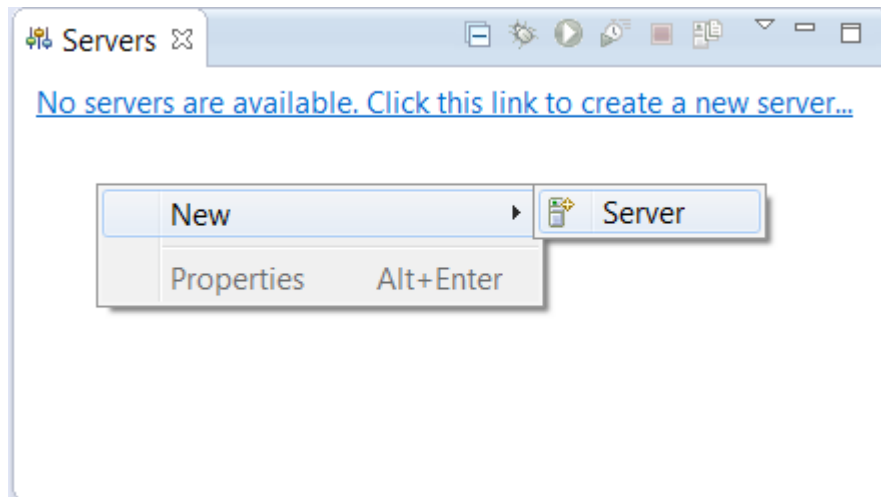


Figure 7 If you haven't created a Liferay server yet, you can do so from the Servers tab in Liferay IDE

Scroll to the **Liferay, Inc** folder and select **Liferay v6.2... Server**. Choose the **Liferay v6.2...** runtime environment that you just created.

Now your server is set up. Let's launch it and perform some tests!



## 4.4.4 Launching and Testing Your Liferay Server

Once your Liferay Portal Server is set up, you can launch it from the Servers tab in Eclipse. You have a few options for launching and stopping the server once it's selected in the Servers tab.

From the **Servers** tab:

- Click on the green **Start the Server** button to launch it (or use Ctrl+Alt+R).
- Click on the red **Stop the Server** button to stop it (or use Ctrl+Alt+S). You'll only see this button if the server is running.
- Right click the server and select **Start**.
- Right click on the server and select **Stop**.

Once the server is launched, you can open Liferay portal home from the **Servers** tab by right clicking your Liferay Tomcat server and selecting Open Liferay Portal Home.

Next, you'll learn to create new Liferay projects in Liferay IDE.

## 4.5 Creating a New Liferay Plugin Project

Plugins for Liferay Portal must be created inside of a Liferay project. A Liferay project is essentially a root directory with a standardized structure containing the project's (and each of its plugins') necessary files. Since each plugin type requires a different folder and file structure, let's create a project to illustrate the process.

If you've been following our Liferay IDE configuration instructions, your Plugins SDK and Liferay portal server have already been configured in Liferay IDE. Now let's create a new Liferay plugin project in Liferay IDE.

1. Go to **File → New → Other...**
2. From the wizard go to: **Liferay → Liferay Plugin Project** and click **Next**.
3. In the project creation wizard, you'll name and configure your project. We'll create a plugin project that we'll use throughout this guide. First, we'll create a bare bones plugin project; then, we'll manually add an additional plugin to the project and add additional configurations.
  - a. Provide both a Project Name, which is used to name the project's directory, and a Display Name, which is used to identify the plugin when adding it to a page in Liferay Portal. Our demonstration project will have the project name **test-portlet** and the display name **Test**.
  - b. Leave the Use default location checkbox checked. By default, the default location is set to your current workspace. If you'd like to change where your plugin project is saved in your file system, uncheck the box and specify your alternate location.
  - c. Select the Ant (liferay-plugins-sdk) option for your build type. If you'd like to use Maven for your build type, navigate to the Developing Plugins Using Maven section for details.

- d. Your newly configured SDK and Liferay Runtime should already be selected. If you haven't yet pointed Liferay IDE to a Plugins SDK, click Configure SDKs to open the Installed Plugin SDKs management wizard. You can also access the New Server Runtime Environment wizard if you need to set up your runtime server; just click the New Liferay Runtime button next to the Liferay Portal Runtime dropdown menu.
- e. Under Plugin Type, indicate which plugin type your project will hold by selecting one from the list. You can choose from Portlet, Service Builder Portlet, Hook, Layout Template, Theme, or Ext. Liferay IDE provides handy wizards for creating new Liferay projects. For our demonstration project we will create a portlet, so make sure that Portlet is selected.

**New Liferay Plugin Project**

Create a new project configured as a Liferay plugin

Project name:

Display name:

☒ Use default location

Location:

Build type:

Plugins SDK:

Liferay runtime:

Select the plugin type

☒ **Portlet**  
Create an application to run as a portlet in Liferay portal.

☐ **Service Builder Portlet**  
Create a portlet plugin that uses the Liferay Service Builder framework.

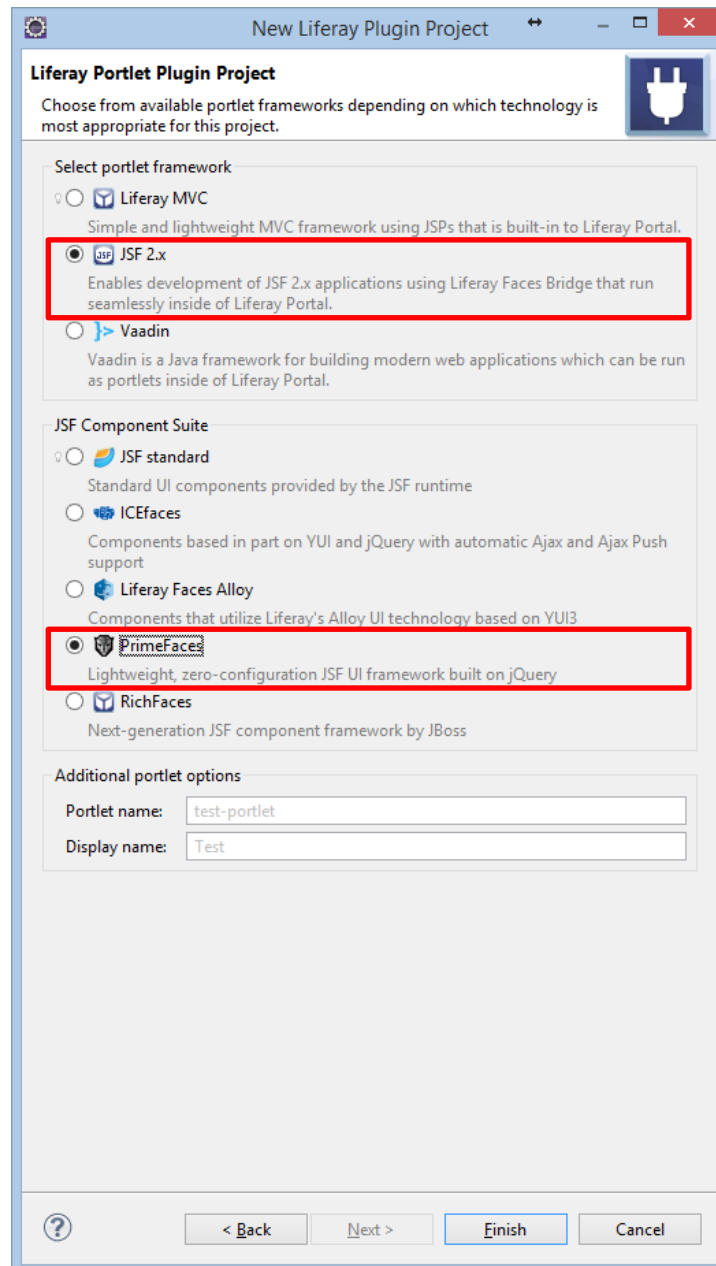
☐ **Hook**  
Override or extend Liferay's default behavior and functionality.

☐ **Layout Template**  
Add new custom layouts for Liferay pages.

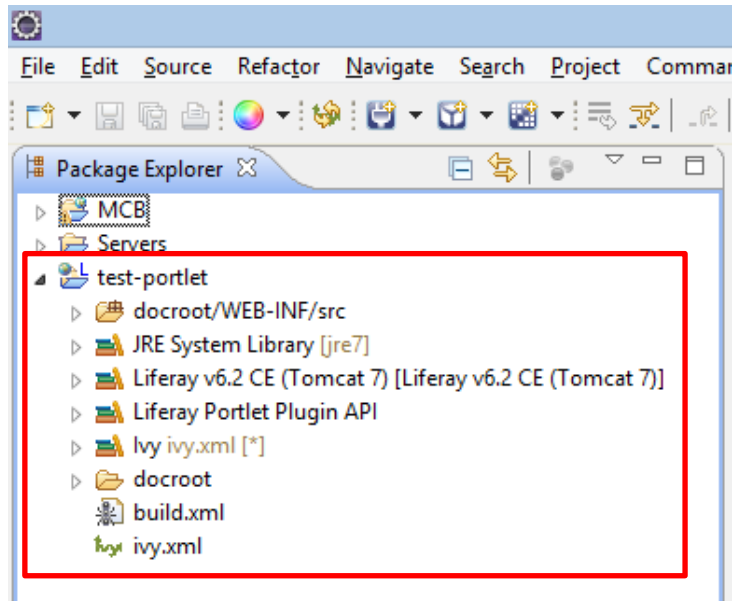
☐ **Theme**  
Build a custom look and feel for the portal.

☐ **Ext**  
Light-weight extension environment for Liferay as a plugin.

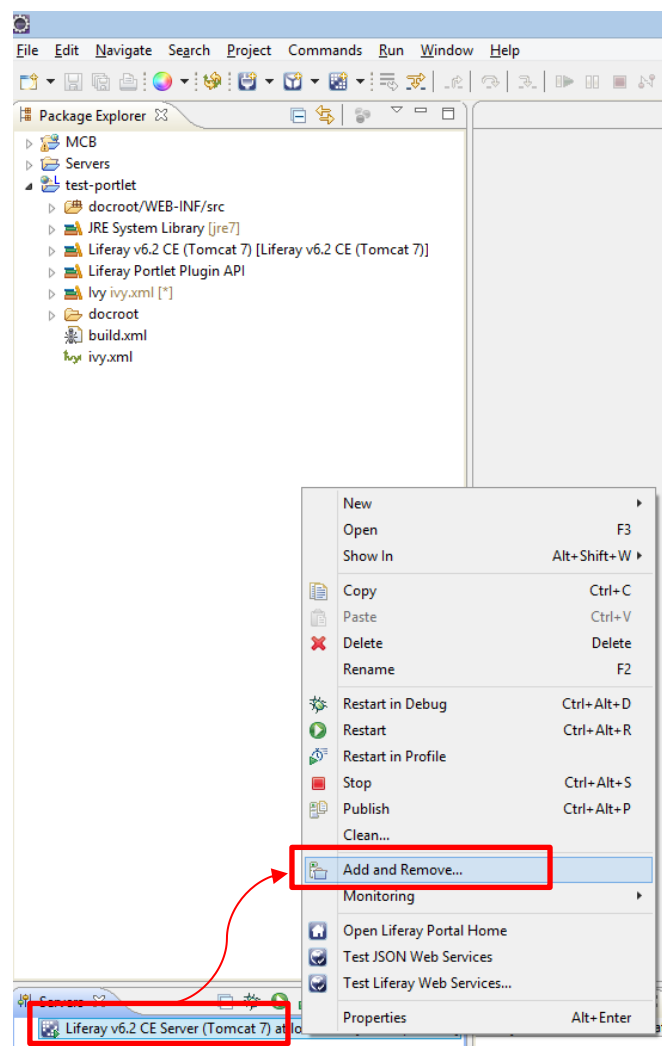
4. We will create the test-portlet using the JSF framework, so check the JSF 2.x option. Also we will use the Primefaces suite so go on and select it as well.



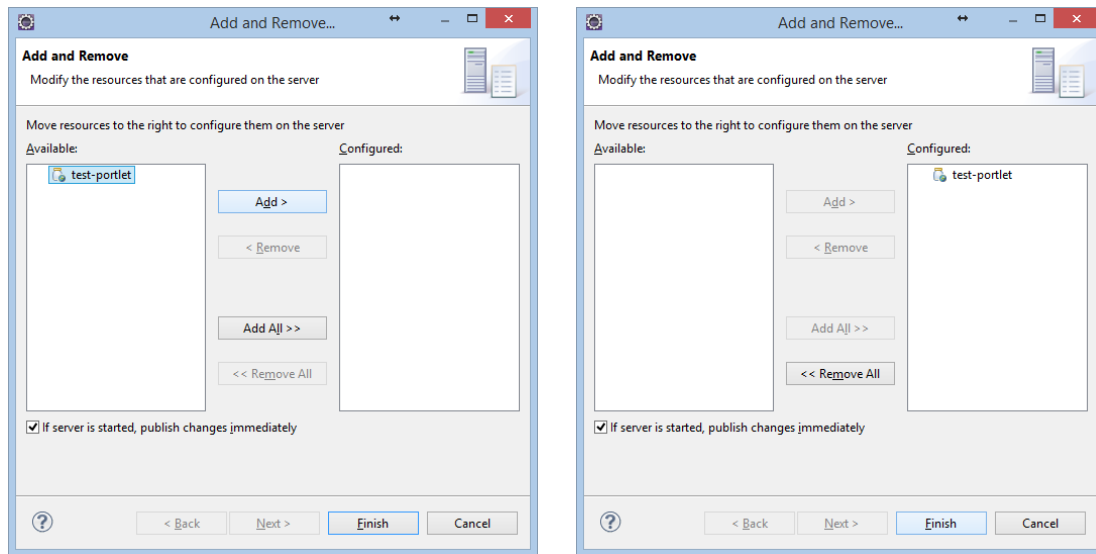
5. Liferay IDE creates the project and all the necessary structure (folders and configuration files). The following snapshot depicts the structure created automatically by the Liferay IDE plugin and the Eclipse.



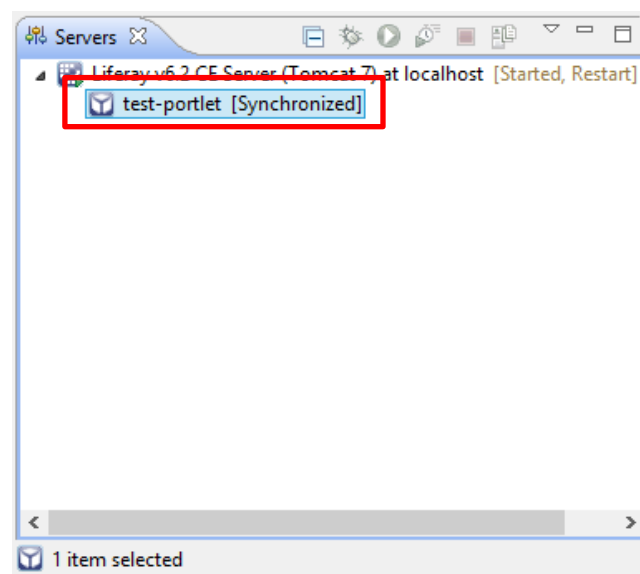
6. Now we have to add our newly created portlet to the local Liferay portal. **Right click** on the **Liferay v6.2 CE Server** at the Server tab (bottom left) and then click the **Add and Remove...** option from the context menu.



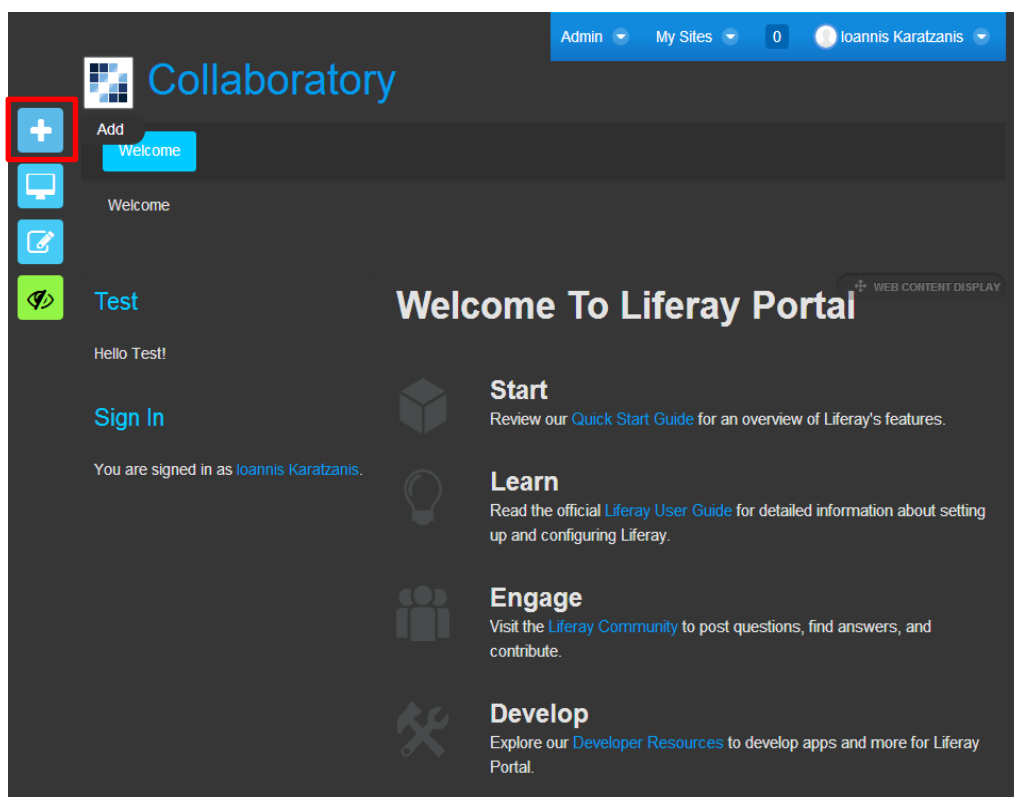
7. From the dialog window that appears, select the test-portlet from the **Available** portlets column and click the **Add >** button to move it to the **Configured** portlets. Then click Finish.



8. The test-portlet should appear below our server's local installation in the Servers tab (bottom left), as displayed at the following image. The portlet will be installed to the server, a procedure that will take some seconds to complete. At the status bar will appear a process bar that will help you understand when the procedure is complete. In the same time deployment messages appear in Eclipse's Console window, with information regarding the deployment process and potential debugging information. If the portlet has been installed correctly the Synchronized status label should be appear beside it. (Notice: Sometimes there might be a need to restart the server in order to complete the deployment process. This can be done by the red squared button at the top of the Servers tab, and then -when the server has stopped- with the green start button.)



9. Use your browser to go to the local installation of the Liferay portal (*localhost:8080*) and sign in (using your credentials). Then click the Add button at the left as shown at the image bellow.



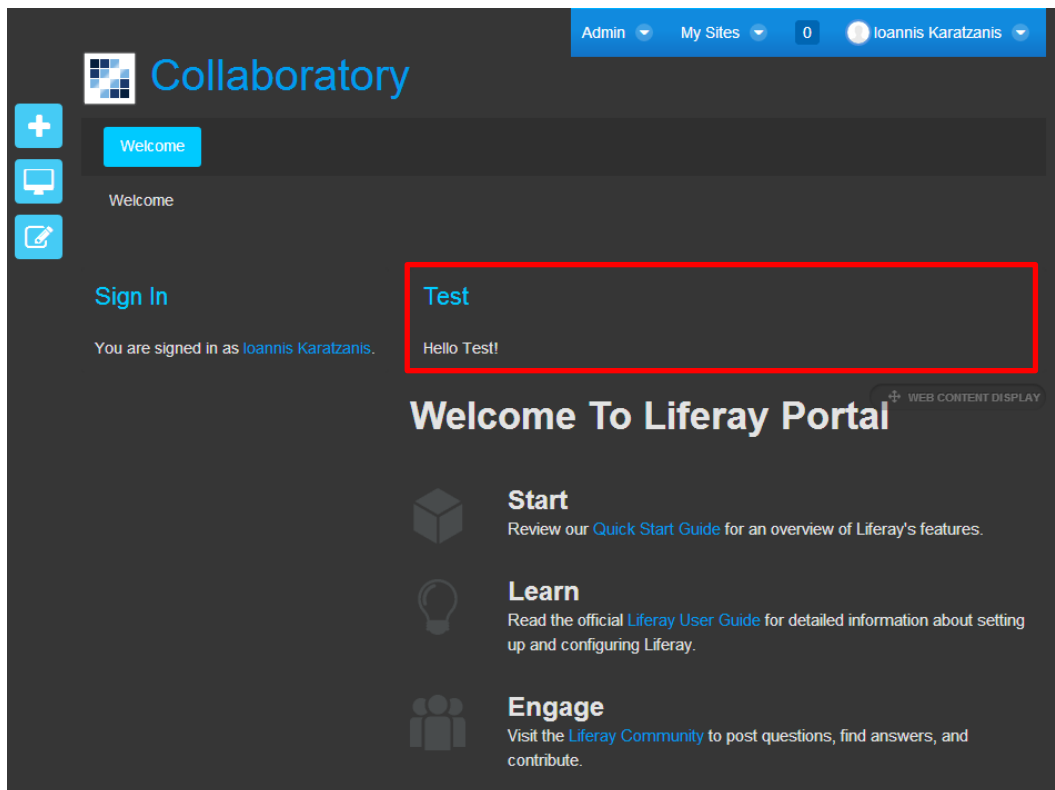
10. A panel will slide from the left of the screen with all the relative configuration options for adding portlets, content or other components to the portal. Go to the Application tab and at the filter field write the word test, this way the applications will be filtered and only our test-portlet will be shown. There are two ways to add the portlet to the current page. Either you click the Add option beside the test-portlet (and the portlet appears somewhere on the page where you have to move it to the place you want), either you click and drag the portlet to the desired place on the page.

The image consists of two screenshots illustrating the process of adding a portlet to the Liferay portal.

**Top Screenshot:** Shows the 'Add' dialog box. The 'Applications' tab is selected. In the 'Sample' dropdown menu, 'Test' is highlighted. Below the dropdown, there are checkboxes for 'Can Be Added Once' and 'Can Be Added Several Times'. A button 'Install More Applications' is visible at the bottom.

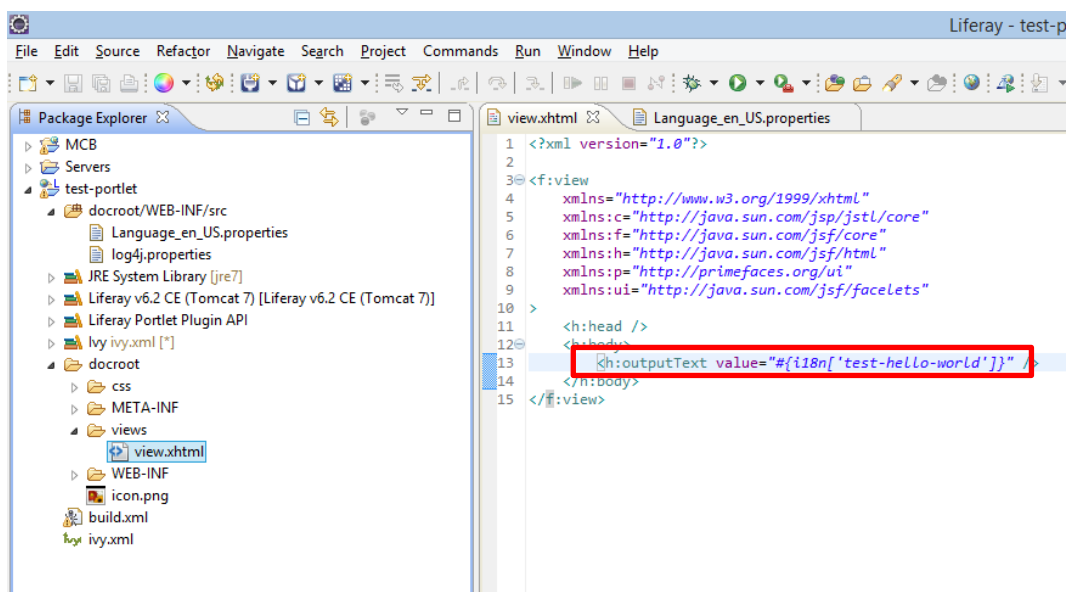
**Bottom Screenshot:** Shows the 'Collaboratory' page. The 'Test' portlet has been added to the page. A red arrow points from the 'Test' portlet in the 'Add' dialog to the 'Test' portlet on the page. The portlet on the page displays the message: 'Test will only be shown after you refresh the current page.'

Once the portlet is placed on the page, the page will have to be refreshed in order to show the changes. As it is shown in the following image, the test-portlet is now fully functional, displaying its title ("Test") and its content ("Hello Test!"). Let's see how this works.



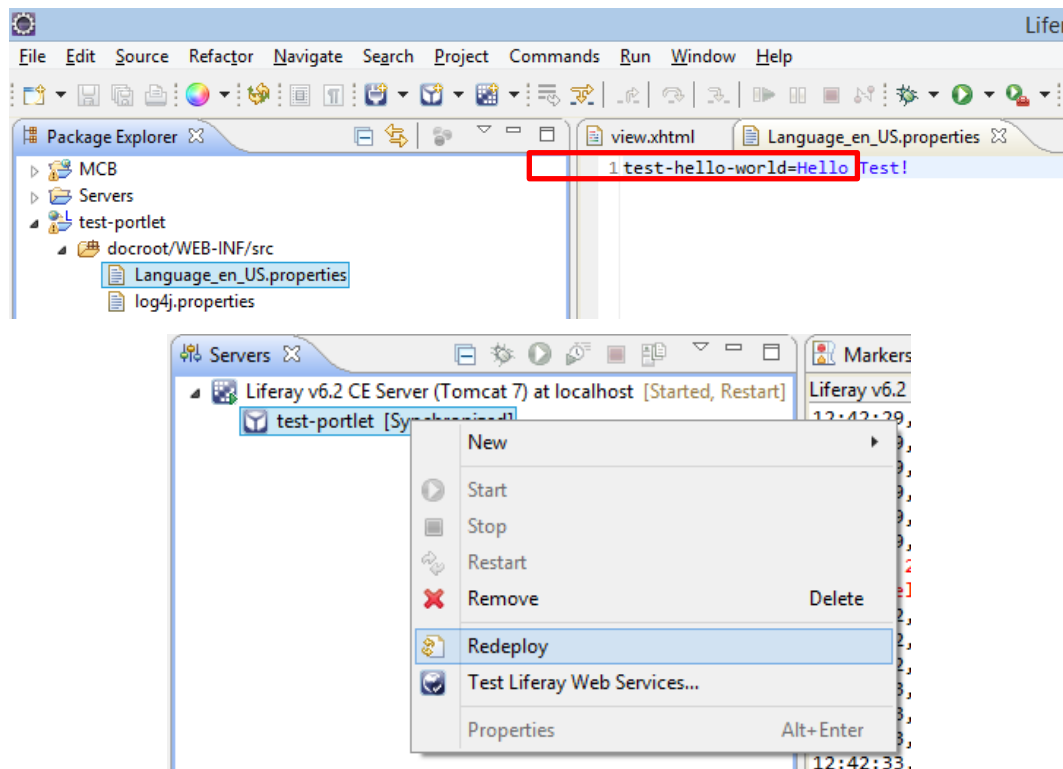
11. Open the **view.xhtml** file from the **test-portlet/docroot/views** folder.

The line in the red rectangle is the code of our test-portlet responsible for the message displayed in the portlet. The option value is set to display the contents of the test-hello-world variable as defined in the **language\_en\_US.properties** file.



12. Now let's open the **language\_en\_US.properties** file from the **test-portlet/docroot/WEB-INF/src** folder. Change the **"Hello Test!"** string to **"Hello World!"** If you go on and refresh the page in your browser you probably won't see any difference. That is because only the changes on the xhtml files are usually shown immediately, while for any other file type most of the times it is needed to redeploy the portlet first.



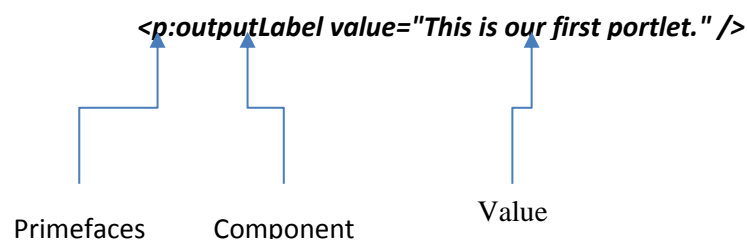


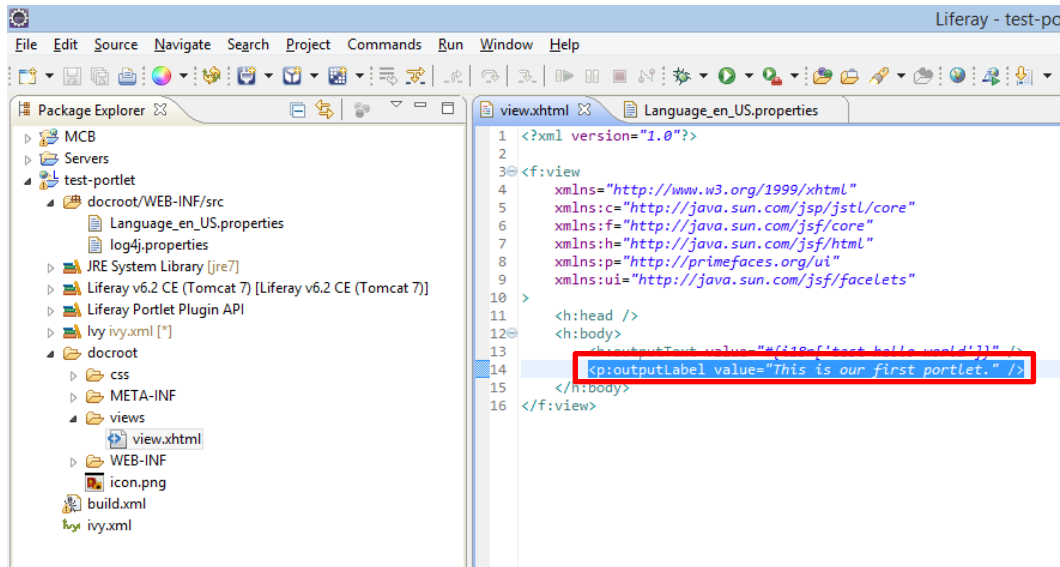
After redeploying the portlet, the new content will be shown at the page once you refresh it.

13. In order to add some extra content to our portlet, let's go back to **the view.xhtml** page. Right after the **h tag** add the following line (as shown in the image below).

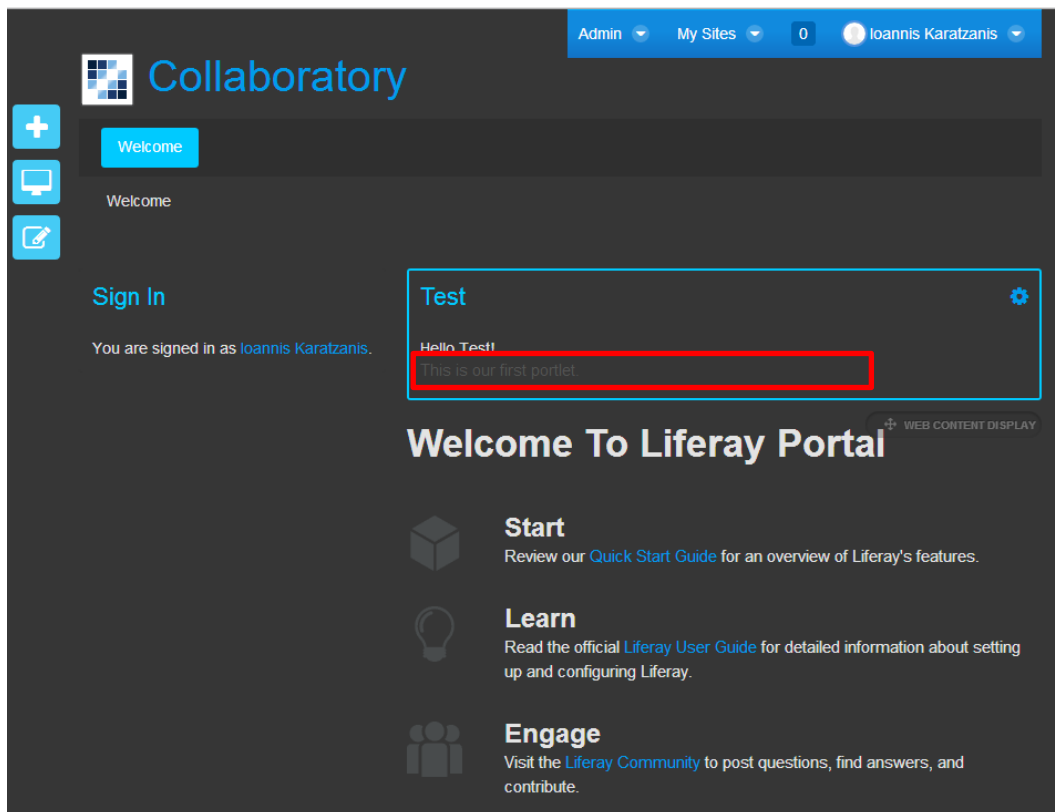
**`<p:outputLabel value="This is our first portlet." />`**

At the line above we have used a *Primefaces* component. Let's analyze it.





14. Refresh the page and the new line will be shown.



## 5 More Information

For more information on developing portlets for the Liferay portal please visit the official website of Liferay at the following link:

<https://www.liferay.com/documentation/liferay-portal/6.2/development>

## Appendix 1 – Abbreviations and acronyms

<i>API</i>	Application Programming Interface
<i>CMS</i>	Content Management System
<i>EIP</i>	Enterprise Information Portal
<i>EMEA</i>	Europe, Middle East, Africa
<i>HTML</i>	HyperText Markup Language
<i>IDE</i>	Integrated Development Environment
<i>JDK</i>	Java Development Kit
<i>JRE</i>	Java Runtime Environment
<i>JSF</i>	JavaServer Faces
<i>REST</i>	Representational State Transfer
<i>RSS</i>	Rich Site Summary – RDF Site Summary – Really Simple Syndication
<i>SOA</i>	Service Oriented Architecture
<i>SOAP</i>	Simple Object Access Protocol
<i>SSO</i>	Single Sign On
<i>URL</i>	Uniform Resource Locator
<i>WML</i>	Wireless Markup Language
<i>XHTML</i>	Extensible HTML