Deliverable No. 10.1

The CHIC Portal

Grant Agreement No.:	600841
Deliverable No.:	D10.1
Deliverable Name:	The CHIC Portal
Contractual Submission Date:	30/11/2013
Actual Submission Date:	30/11/2013

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)	**X**
CO	Confidential, only for members of the consortium (including the Commission Services)	

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
Document name:	The CHIC Portal
Nature (R, P, D, O)[1]	O
Dissemination Level (PU, PP, RE, CO)[2]	RE
Version:	5.0
Actual Submission Date:	30/11/2013
Editor: Institution: E-Mail:	Georgios Zacharioudakis FORTH gzaxar@ics.forth.gr

ABSTRACT:

This deliverable describes the development process up to now of the CHIC portal which will be the main "point of entrance" to the CHIC platform, offering access and management to the data and model repositories, assisting the creation of new hypermodels by linking existing ones etc. In this document we describe the evaluation of existing portal frameworks, which resulted in the selection of the Liferay portal framework. We document the user requirements of the CHIC portal and the foreseen functionality. We describe the architecture and the components that have been identified up to now, which will be gradually implemented in the course of the project and embedded in the portal. Furthermore, we elaborate on the process of installing and configuring the development version of the CHIC portal as it is up to now.

KEYWORD LIST:

Portal, framework, portlet, web interface, Liferay, user communities, integration, web services, plugins, CHIC architecture.

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.

[1] **R**=Report, **P**=Prototype, **D**=Demonstrator, **O**=Other

[2] **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)

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.

MODIFICATION CONTROL			
Version	**Date**	**Status**	**Author**
1.0	10/11/2013	Draft	Georgios Zacharioudakis
2.0	18/11/2013	Draft	Stelios Sfakianakis
3.0	22/11/2013	Draft	Georgios Zacharioudakis
4.0	29/11/2013	Internal review	Feng Dong Fay Misichroni
5.0	29/11/2013	Final version	Georgios Zacharioudakis

List of contributors

- Georgios Zacharioudakis (FORTH)
- Stelios Sfakianakis (FORTH)
- Kostas Marias (FORTH)
- Feng Dong (BED)
- Fay Misichroni (ICCS)

Contents

1 Executive Summary

The CHIC portal will provide the primary user interface of the CHIC platform to the end users. The CHIC portal will offer access to the model repository for publishing and discovery of models and hypermodels, will support the upload and management of data into the data repository, will assist the creation of new hypermodels and will enable the users to execute them to the underlying infrastructure.

The aim of this document is to collect the user requirements for the CHIC portal and to describe its foreseen functionality in order to assist the design and implementation of the portal by the relevant Work Packages in the course of the project. In this document we evaluate the most prominent existing portal frameworks and we propose the selection of the Liferay portal framework. In addition, we provide a brief description of the portal tools that have been identified to be implemented, based on the identified CHIC components proposed by the CHIC architecture up to now. Finally, we present the current state of the development portal.

2 Introduction

The CHIC project will develop a suite of tools, services and secure infrastructure that will support accessibility and reusability of VPH mathematical and computational hypermodels. These will include a hypermodelling infrastructure consisting primarily of a hypermodelling editor and a hypermodelling execution environment, an infrastructure for semantic metadata management, a hypermodel repository, a hypermodel-driven clinical data repository, a distributed metadata repository and an in silico trial repository for the storage of executed simulation scenarios. Multiscale models and data will be semantically annotated using the ontological and annotating tools to be developed. An image processing and visualization toolkit and cloud and virtualization services will also be developed. The CHIC tools, services, infrastructure and repositories will provide the community with a collaborative interface for exchanging knowledge and sharing work in an effective and standardized way and a number of open source features and tools will enhance usability and accessibility.

The CHIC portal will be the main “point of entrance” to the wider CHIC platform, offering access to the model repositories for publishing and discovery of models and hypermodels, providing links to specific tools, supporting the upload of new data sets and the management of the existing ones, assisting the creation of new hypermodels by linking existing ones, enabling the users to execute the hypermodels to the underlying infrastructure etc.

In this document we describe the user requirements for the CHIC portal and its foreseen functionality in order to assist the design and implementation of the portal by the relevant Work Packages in the course of the project. We describe the evaluation that we performed based on the literature and the technological features of the most prominent existing portal frameworks and we propose the selection of the Liferay portal framework, based on a number of relevant criteria. In addition, we provide a brief description of the portal tools that have been identified to be implemented, based on the identified CHIC components proposed by the CHIC architecture up to now. Finally, we present the current state of the development portal.

The rest of this document is organized as follows:

In chapter 3 we elaborate on the user requirements for the CHIC portal, the users and the foreseen functionality. In chapter 4 we describe the evaluation that we carried out on the most prominent portal frameworks, the criteria for the evaluation and our proposed portal framework. In chapter 5 we describe the current status of the CHIC portal and the planning for its implementation during the course of the project, based on the mapping of CHIC components to corresponding portal modules (portlets).

3 Requirements for the CHIC portal

3.1 Stakeholders - User groups

Taking into account the CHIC Technical Annex, the deliverable D2.2 “Scenario based user needs and requirements” as well as the interaction taken place so far in the CHIC project, we expect the following general categories of users for the CHIC project.

- Researchers (biologists, mathematicians, informatics, physicists etc.)
- Clinicians
- Tool developers (IT)
- Model developers
- Administrators
- Guests (non-registered or non-authenticated portal users)

It is expected of course that a user may be related, simultaneously or at different times, to many of the groups above so this categorisation is only made to highlight the functionality that a general group of people expected from the CHIC portal. The actual user roles that the portal implementation will select and enforce might differ based also on technological factors. These technological factors will drive the need of dividing the users into distinct groups and giving them different security and authorization access, so in the portal implementation a wider list than the categorisation above is probable to be finally created and implemented based on fine-grained roles and access rights.

3.2 Foreseen functionality of the CHIC portal

The basic functionality that the CHIC portal is expected to provide, as documented in the Technical Annex of the project, is the following:

- Access, browsing and uploading of resources (data) into the data repository.
- Access, browsing and uploading of resources (models) into the model repository.
- Visual composition of hypermodels using a graphical hypermodelling editor.
- Execution of hypermodels and monitoring of the execution status.
- Visualization of hypermodel execution results.
- Tools for the annotation of models and data.

3.2.1 All users

There are functions that apply to the majority of the users. These kind of functionalities are regarded as the general functionalities provided from enterprise portals, namely:

- Access control
    - Single sign-on, authentication, authorization, sharing of information
- Management of users

- User roles, user groups and communities, permissions, personalization
- Unified user interface
- Tools management
  - Registration and publishing of new tools (portlets)
  - Execution of tools (portlets)
- Collaboration tools
  - Calendar, user forum and chat

Some of these functionalities such as the collaboration tools might come into second priority, since this type of functionalities can be achieved also by using other external tools. Nevertheless, integration and unification of user interfaces is always highly appreciated and some of these functionalities will be provided or easily implemented based on the skeleton of the tools that are provided by the underlying portal framework that will be chosen.

3.2.2 Researchers

This user group is the main focus and purpose of the CHIC project. Researchers will use the portal to explore the available models and datasets provided in the corresponding repositories, to link and combine them using the appropriate hypermodelling editor into hypermodels, to execute them and visualize the results.

3.2.3 Clinicians

This user group which is regarded as a subset of the researchers group mentioned above will mainly focus on the execution of existing tools in the portal and not particularly on the creation of new ones.

Exemplar clinical scenarios which describe the requirements of clinicians for the CHIC portal are documented in D2.2. In this deliverable we see that clinicians will use the CHIC platform, through the CHIC portal, to execute models and hypermodels seeking answers to specific clinical questions, such as those related to specific diseases (e.g. Nephroblastoma, Glioblastoma). These answers might come out of the execution of already available models and hypermodels, or, more rarely, by creating their own hypermodels. Also, clinicians will use the CHIC portal for accessing and browsing data repositories and for providing new, already anonymized or pseudonymized data.

3.2.4 Tool developers

This user group comprises the developers of IT tools, such as services, standalone applications and portlets to assist the realization of the wider CHIC platform. The tool developers, for implementation and evaluation purposes, may temporary share the roles, in terms of security and authorization permissions, of other user groups and execute their tasks. They may combine or execute models and hypermodels, access and explore repositories of models and data etc.

3.2.5 Model developers

This user group is a subset of the researchers mentioned above, aiming to browse the available models in the model repository, to publish new models and to annotate them, to explore the ability

to create new models by re-using or modifying existing ones and to execute them into the hypermodelling execution environment.

3.2.6 Administrators

Administrators in the CHIC portal may be of two categories, either for IT related or for modelling related issues. Administrators for IT issues will need to manage users and communities, manage permissions and authorization policies, publishing, integration and execution of portlets. Administrators for modelling issues will oversee the management of the model repositories, data repositories and data access and the integration and execution of the models and hypermodels into the CHIC platform.

3.2.7 Guests

We expect that for Guest users, which are non-registered or non-authenticated users, only selected, limited functionalities of the portal will be available. These functionalities will be defined in due course in collaboration with the relevant Work Packages that will examine both the legal restrictions and the security mechanisms that might apply. We expect that these functionalities will mainly be for dissemination activities, informational purposes and for attracting candidate users which relate to the other user groups (researchers, developers etc.).

4 Evaluation of portal frameworks

4.1 Portal technologies

An enterprise portal is a framework for integrating information, providing a common user interface, creating user communities and re-using pieces of information and (web) software tools. Enterprise portals have been created out of the evolution of web portals, which in turn were created from the evolution of simple web sites. Developers early realized that some pieces of information or functionality were often needed to be re-used in many sites, so this need lead to the development of some standards both for interoperability and re-usability assistance.

A portal framework usually is executed inside an application server, such as JBoss, Geronimo, Jetty, Apache Tomcat, WebLogic, WebSphere, GlassFish etc. The application server executes the portal framework as a web application and provides both the runtime and some basic functionalities such as libraries, components etc. The portal framework also uses a database for storing data such as user and personalization data, session information and other.

4.1.1 JSR

The first version of a Java Portlet Specification[3] was developed under a community process, known as Java Specification Request (JSR) 168[4] and introduced a basic programming model for the development of reusable software modules (portlets), specifying issues such as:

- Action processing and rendering, such the one commonly know from the Model-View-Controller design pattern.
- Windowing modes and states, such as what content to generate and what task to perform (minimize, maximize etc.), how much space to occupy in a web page etc.
- A data model, for storing information about the user view, the session, persistent data etc.
- Packaging format for the portlet artifacts.

The Java Portlet Specification version 2.0 was developed a few years later, on 2008, under the Java Specification Request (JSR) 286[5]. It was created to improve the 1st version of the specification (JSR 168) and it was created in alignment with the WSRP specification (see below).

Some of the major features introduced or improved were:

- The inter-portlet communication, through events and render parameters.
- The serving of dynamically generated content and resources and the handling of AJAX and JSON data.
- Introduction of portlet listeners and filters.

We expect that a portal framework to be selected for the CHIC project will at least support the JSR286 specification. Any additional standard and technology supported will also be of great benefit for the development process.

[3] http://en.wikipedia.org/wiki/Java_Portlet_Specification
[4] http://www.jcp.org/en/jsr/detail?id=168
[5] http://www.jcp.org/en/jsr/detail?id=286

4.1.2 WSRP

The WSRP specification[6] is an OASIS-approved network protocol standard, designed for communication with remote portlets and it defines a web service interface for the interaction with web services that are presentation-oriented.

As a network communication protocol, used to interact with remote containers, it does not assume or require a specific implementation, programming language or environment. As such, it does not compete with the JSR portlet specifications which define the portlets. Several existing portal frameworks provide WSRP implementations to assist developers. In its most recent version, WSRP version 2.0, it supports technologies such as AJAX and REST for the interaction with remote portlets.

4.2 Enterprise portal frameworks

There is a large list of available portal frameworks, offering both the "skeleton" of the information system to build an enterprise portal as well as a collection of already implemented portlets which can be integrated into. Many of these frameworks are offered as open source or free software[7] and below we select and compare some of them, having as primary selection filter for a portal framework to be freely available, with non-restricting licensing model and also to be widely supported and adopted by the user and developer community. Furthermore, taking into account the user requirements for the CHIC portal as described in the corresponding section above, we expect from a portal framework to provide functionality for user and access management (single sign on, user management, user roles functionality), to support a wide range of development tools (application servers, databases, web technologies) and to provide as many as possible already available tools and portlets to ease the development of the CHIC portal.

4.2.1 Liferay portal

The Liferay portal[8] is a free framework which claims to be one of the best solutions available, widely supported by the community with an extensive user base.

According the technical specifications in its website it features:

- Support for many operating systems (Linux/Unix flavours, Windows, Mac) and Java Runtimes
- Support for many Application Servers (Geronimo, GlassFish, JBoss, OracleAS, Weblogic, Websphere) and Servlet containers (Jetty, Resin, Tomcat)
- Support for many databases (IBM DB2, MySQL, Oracle, PosgreSQL, SQL server, Sybase)
- It is deployable to the cloud and virtualized environments (Amazon EC2, VMware etc)
- It uses/supports a number of important technologies (AJAX, Spring, J2EE, Groovy, Hibernate, Web services, REST, JSON)
- Identity management, Single sign-on and Security technologies (LDAP, Open SSO and others)
- Scripting languages support (Javascript, Ruby, PHPm Python)

[6] http://docs.oasis-open.org/wsrp/v2/wsrp-2.0-spec-os-01.html
[7] http://en.wikipedia.org/wiki/List_of_enterprise_portal_vendors
[8] http://www.liferay.com/products/liferay-portal/features/portal

- All major portlet standards (JSR286, WSRP etc.)
- Many themes, layouts and already available portlets
- Web publishing features (CSS etc.)
- Collaboration tools (Wiki, Blog, Calendar, Social tools, Message boards etc.)

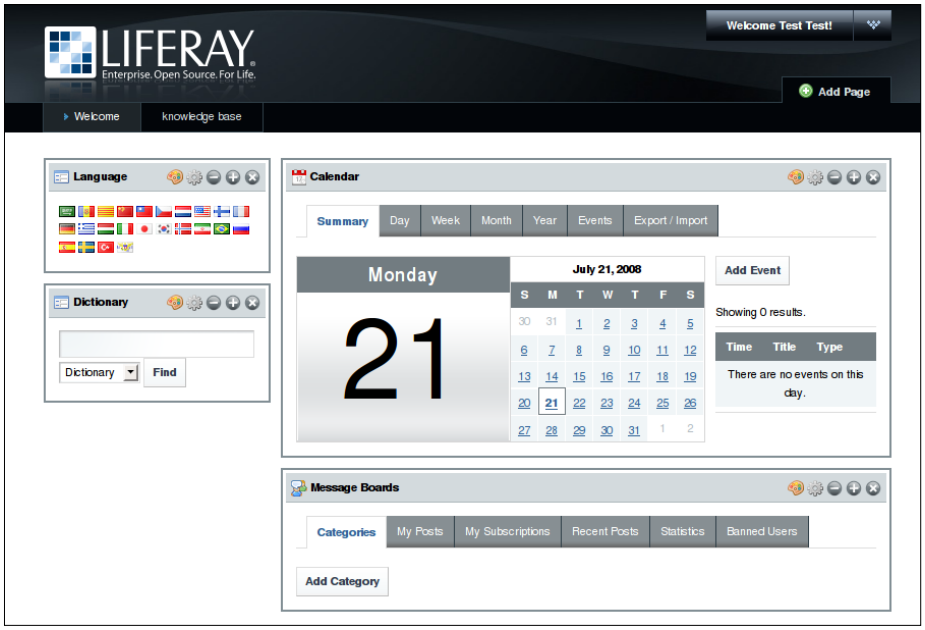

Figure 1 : User interface of the Liferay portal framework

4.2.2 Apache Jetspeed

The Apache Jetspeed[9] portal is a free framework from the well-known Apache developer community. According its website it features:

- Support of multiple application servers (Tomcat, Geronimo, Jetty, JBoss, WebLogic, WebSphere)
- Support of many databases (MySQL, PostgreSQL, Oracle, IBM DB2, Derby, MS SQL)
- Support of portlet specifications (JSR 286 etc.)
- Support of security mechanisms (J2EE security, Jetspeed single sign-on, OpenID, LDAP)
- User and role management
- A number of important technologies (Spring, Struts, MyFaces)
- A number of already available portlets (Administrative, calendar, statistics, etc.)
- Support for many development tools (Ant, Maven, JSF, Spring etc.) and customization tools

[9] http://portals.apache.org/jetspeed-2/

The Apache Jetspeed offers a list of features comparable to Liferay, with good documentation and support. In our opinion it lacks in stability and its graphical user interface and the developer community support is not as extended as Liferay's.

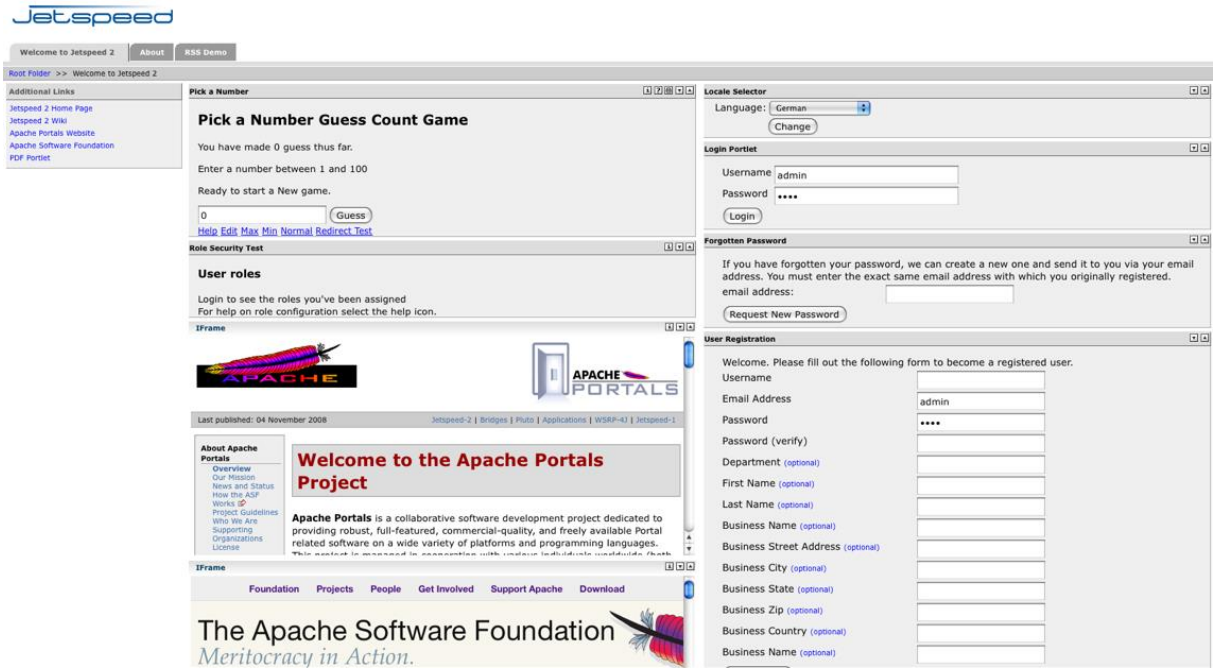

Figure 2 : User interface of the Apache Jetspeed portal framework

4.2.3 GateIn portal

The JBoss and eXo GateIn[10] portal framework resulted from the merging of the previous frameworks JBoss portal and eXo portal.

According the GateIn website, this portal framework features:

- Support for databases (MySQL etc.)
- Support for application servers (JBoss, Tomcat)
- Support for cloud deployment
- Enterprise Java (persistence, REST, Beans, etc.)
- Standards and web technologies support
- LDAP and Single sign-on functionality
- User and group management
- Website templates and page layouts

[10] http://www.jboss.org/gatein

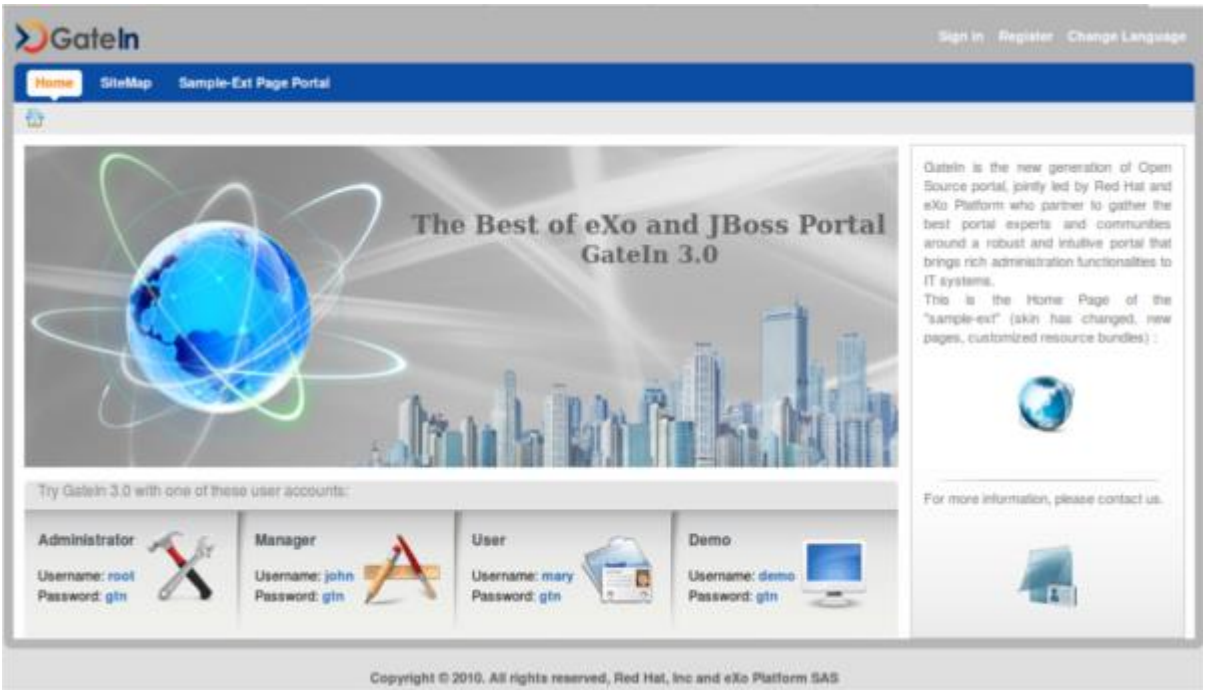

Figure 3 : User interface of the GateIn portal framework

We think that documentation of GateIn framework is average compared to the other frameworks, it does not offer such a large variety of ready to use portlets and it has less support from the user and developer community.

4.3 Basic comparison of enterprise portal frameworks

Below we outline the most important features that we seek in an enterprise portal framework and we compare the most prominent frameworks that we selected out of the list of solutions provided as free/open source software.

	Apache Jetspeed	**GateIn**	**Liferay**
Stability	Average	Average	More robust
Installation	Easy	Less easy	Easy
GUI	Average	Nice	Nice
Application Servers support	Many	Few	Many
Database support	Many	Many	Many
Standards support	Good	Good	Good
Documentation	Good	Average	Good

Available portlets	Many	Few	Many
Single sign-on	Yes	Yes	Yes
User management	Yes	Yes	Yes
License	Open / Free	Open / Free	Open / Free
Community	Average	Average	More active

Table 1 : Comparison of portal frameworks

From this comparison we see that the framework which stands out as the most favourable solution is Liferay.

4.4 Portals of other EU projects

It is important to note that other projects in the VPH field also use portals for the integration of their work and the provision of a unified user interface for accessing their infrastructure. Besides the technology factors that are mentioned above, it is also crucial to take into account the type of technology that is used in other projects in the same field, so that we enable as much as possible re-usage and interoperability. This is important not only for minimising the effort and the used resources but more importantly to create a community of tools that interoperate and add more value to the technology that is being created from all EU projects, since all projects can benefit from this kind of interoperability and can extent their exploitation options.

So, taking into account not only the technological functionality of each portal framework but also its adoption and support from other projects in the VPH area, the portal frameworks that support the usage of standards such as those described in the sections above are the first candidates to be adopted also from the CHIC project. Since the project portals usually provide only limited access to their inner workings and their sensitive information, not all projects provide access or further insight to the technology that they use or the kind of tools that they develop. To the extent that we are aware of, we know of at least two EU projects which use **Liferay framework** as the technology of choice for the creation of their portal, namely the p-medicine[11] project and the INTEGRATE [12]project, because these projects share with CHIC a number of common consortium partners and we are able to communicate this knowledge.

4.4.1 The *p-medicine* project portal

The *p-medicine* portal[13] provides a platform to clinicians, patients and researchers to collaborate, share data and expertise, and use tools and services to improve the personalized treatment of patients. The *p-medicine* portal provides access for the *p-medicine* users to the tools and services integrated into the *p-medicine* environment. The current version of the portal contains the initial functionality of the *p-medicine* platform such as the security framework, Data Mining tools, *p-medicine* Workbench, Ontology Annotator as well as ObTiMA. Further information about the p-

[11] http://p-medicine.eu/

[12] http://www.fp7-integrate.eu/

[13] http://p-medicine.eu/tools/p-medicine-portal/

medicine portal, as well as good documentation of general information about portal development, can be found in the corresponding document[14] that describes it.

As noted above, there is a number of common partners between projects CHIC and p-medicine and there has been expressed the will to collaborate and share models, data and tools between the two projects, as much as the two consortiums will mutually agree that this collaboration remains into the scope of the initial focus of their projects and does not impose significant difficulties into the architectural, technological and security level, or legal and ethical restrictions that might apply. Such collaboration might include sharing of tools, access to common repositories and other infrastructure or even co-hosting and integration of their portals. Such collaboration is subject to further discussions between the two consortiums.

4.5 Selected portal framework

Based on the factors documented above, both the technological features and the adoption also by other projects with which CHIC aims to collaborate, such as the p-medicine project, we decided to adopt **Liferay portal framework** as the portal "skeleton" on which to build upon the CHIC portal.

[14] http://p-medicine.eu/fileadmin/p-medicine/public_website/downloads/p-medicine_270089_D8-1-2_Design_and_prototype_implementation_of_the_p-medicine_portal_v1-0.pdf

5 The CHIC portal

5.1 Introduction

CHIC will deliver a system that will support the build of a user community, which is composed of clinicians, biomedical researchers, scientists working on the computational modeling of the human body, etc, to build, share, and execute complex "hypermodels" in order to gain insight into the latent mechanisms of the cancer diseases and how they affect the human organism. The CHIC platform will bring together many different components and tools in support of the envisaged user scenarios and therefore the CHIC Portal is its main façade for the users to interact with the system. As such the Portal interacts with the majority of the CHIC "building blocks" and provides a user-friendly visual interface to the CHIC users.

Taking into account the complexity of the CHIC system it's important to note that the Portal should be based on a "component-oriented" methodology, where each of the backend (sub)systems in the CHIC platform is represented by a "portlet" in the portal. There are many good reasons for such an approach:

- It presents the different functionalities of the CHIC system under a common framework
- It presents a unified design and user experience both in terms of the "look 'n' feel" and in terms of the functionality
- It distributes the development of the user interfaces to the different component developers and at the same time enforces common architectural decisions such as authentication to be implemented in the same way and reused
- It provides a single point of access for the users and therefore it fosters the CHIC dissemination and outreach.

The "modular" framework of the Portal requires a set of common functionalities shared among its portlets. The most prominent one is the authentication of the users i.e. the validation of the user credentials so that only legitimate users can gain access to the CHIC functionalities. In this respect the Portal itself needs to comply with the security framework selected in the Work Package 5 (Task 5.2 "Security tools and services") and present application programming interfaces (APIs) to the portlets for retrieving information for the currently authenticated user. Additional functionality is the provision of persistent storage for storing information relevant to the user interface, e.g. session data. This set of portal provisions can be utilized by the integrated portlets, some of which are described in the following section.

5.2 Architecture - Components

We have described in previous sections the basic functionality that is foreseen to be implemented and embedded in the CHIC portal. Up to the time of creating this document, the following CHIC components have been identified from the CHIC architecture in Work Package 5 and selected to be implemented.

CHIC component	CHIC portlet functionality	Infrastructural?
Data repository	Data repository browsing, data upload, data	No

	annotation.	
Model repository	Model discovery, model publishing, model annotation.	No
Hypermodelling orchestration environment	Model linking and execution.	No
Image processing and visualization environment	Visualization of hypermodel execution results	No
Tools registry	Sharing of the CHIC tools and services	No
Cloud infrastructure	Cloud resources monitoring.	Yes
Security framework	Single sign on, authentication and authorization services.	Yes

Table 2 : CHIC architecture components and corresponding portlets

For each of the components above the corresponding portlets should be available in the CHIC portal:

- The Data Repository portlet allows the user to upload new data and see what's available either in his own private storage area or in the public folders shared by other users.
- The Model repository portlet interacts with the corresponding repository and supports the related functionality.
- The Orchestration portlet supports the graphical design of new hypermodels by combining together existing (hypo)models and their execution, monitoring, and visualization of the results.
- The Visualization portlet supports the visualization and demonstration of the results from hypermodel executions and from the processing of data by the Image Processing and Visualization environment.
- The Tools registry portlet supports the sharing of generic and specialized (e.g. visualization) tools in the CHIC user community.
- The Cloud infrastructure portlet allows the monitoring of the cloud related services and resources and it's accessible for the CHIC administrators and cloud developers.
- The Security portlet supports the management of the users, their groups and roles, and their access rights. It is available to the CHIC security administrators only.

5.3 Installation – configuration – current development status

Liferay supports a large number of different operating systems, databases, application containers and other types of underlying technologies. In the current setup we have chosen the following key technologies:

Underlying technology	Technology chosen
Portal framework	Liferay Portal 6.2 framework
Operating system	Ubuntu 12.04.3 LTS server
Relational Database	MySQL 5.5
Application server	Tomcat 7
Java runtime	Oracle Java SE (JDK) 7u45

Table 3 : CHIC portal installation technologies

These technologies have been selected for a number of reasons, similar to the selection of the Liferay portal itself, i.e.:

- To be available with a free software license
- To be widely adopted and supported by the user and developer community
- To provide all the required functionality for the installation of Liferay
- To have adequate documentation for their installation and configuration

The current version of the CHIC portal has been set up for experimentation and development purposes and is accessible at http://139.91.190.124:8080. This URL address is likely to change or not being always accessible due to development reasons, so, for consistency reasons the stable portal URL will be available and accessible through the CHIC web site once it becomes operational.

As soon as we have a production ready cloud infrastructure in the CHIC project, as planned by the CHIC Technical Annex, we will deploy the portal on the cloud infrastructure for reasons of flexibility, robustness, performance and maintainability.

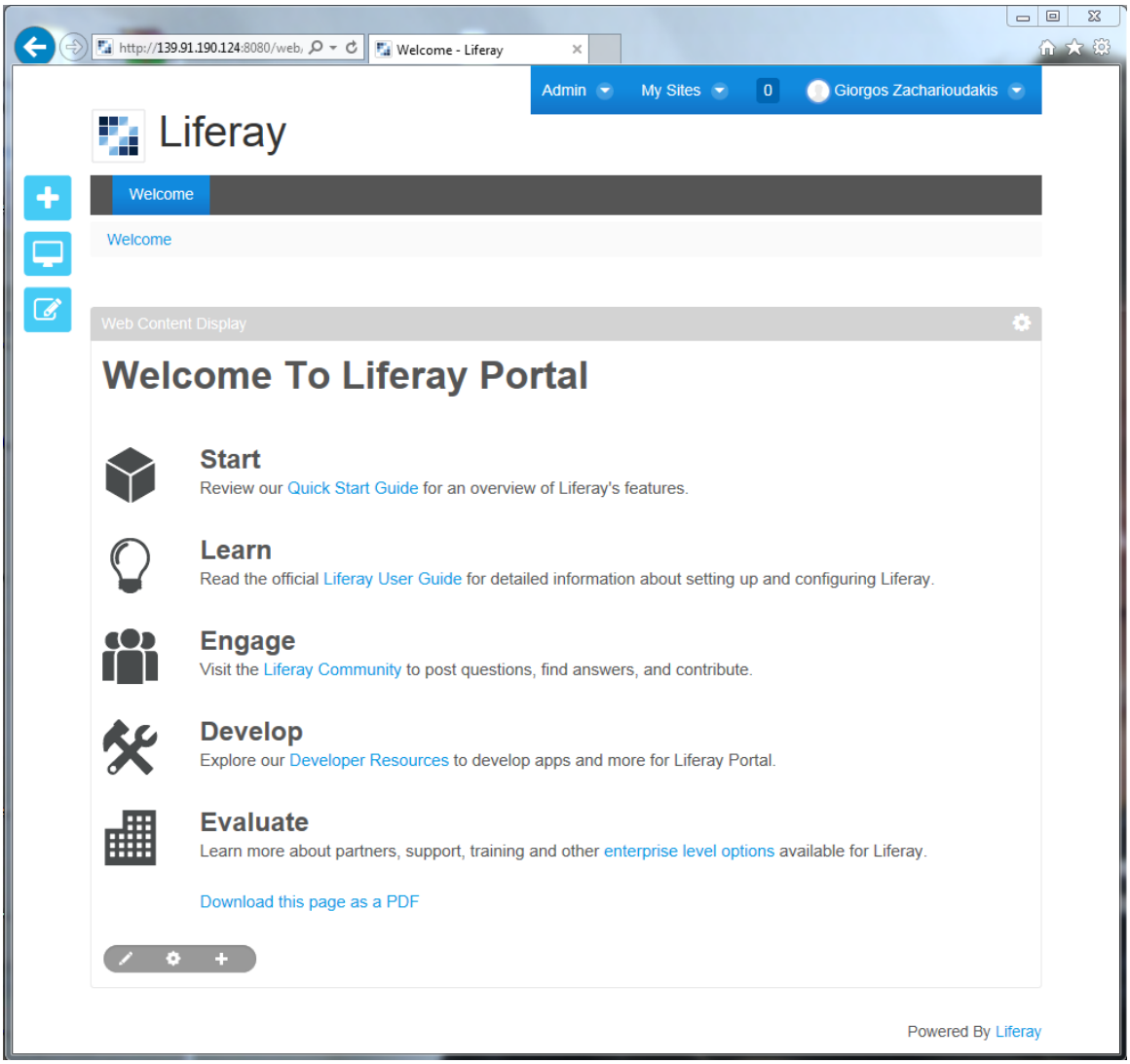

Figure 4 : Screenshot of the development version of the portal

Since the current installation of the portal is still under development, it does not offer any of the foreseen functionality yet. The planning for the development of the portal is expected to follow the development of the rest CHIC architectural elements and be delivered gradually, once the functionality of the CHIC platform is implemented. Some of the CHIC technologies that closely relate with the CHIC portal and are prerequisites for a stable and productive version of the portal are:

- The CHIC Security Framework, for the user management and the single sign on functionality.
- The CHIC cloud infrastructure, for a stable deployment of the portal and the implementation of user storage and related functionality.

The portlets of the corresponding CHIC platform tools will be developed by the corresponding Work Packages and will be integrated by the Work Package 10 which is responsible for the integration of all components into a unified platform.

6 References

[1] http://en.wikipedia.org/wiki/List_of_enterprise_portal_vendors

[2] http://en.wikipedia.org/wiki/Web_Services_for_Remote_Portlets

[3] http://en.wikipedia.org/wiki/Java_Portlet_Specification

[4] http://www.jcp.org/en/jsr/detail?id=168

[5] http://www.jcp.org/en/jsr/detail?id=286

[6] http://docs.oasis-open.org/wsrp/v2/wsrp-2.0-spec-os-01.html

[7] http://www.liferay.com/products/liferay-portal/features/portal

[8] http://portals.apache.org/jetspeed-2/

[9] http://www.jboss.org/gatein

[10] http://p-medicine.eu/

[11] http://www.fp7-integrate.eu/

[12] http://p-medicine.eu/tools/p-medicine-portal/

[13] http://p-medicine.eu/fileadmin/p-medicine/public_website/downloads/p-medicine_270089_D8-1-2_Design_and_prototype_implementation_of_the_p-medicine_portal_v1-0.pdf

Appendix 1 – Abbreviations and acronyms

AJAX	Asynchronous Javascript and XML
IT	Information Technology
JSON	JavaScript Object Notation
JSR	Java Specification Request
LDAP	Lightweight Directory Access Protocol
REST	Representational State Transfer
WSRP	Web Services for Remote Portlets
XML	Extensible Markup Language