DESIGNING A USER INTERFACE FOR REPOSITORIES OF LEARNING OBJECTS

ABSTRACT

This paper presents the design of a user interface for repositories of learning objects. It integrates several tasks, such as submission, browse, search, and comment/review of learning objects, on a single screen layout. This design is being implemented on the web front-end of crimsonHex, a repository of specialized learning objects, developed as part of EduJudge, a European project that aims to bring automatic evaluation of programming problems to eLearning systems.

KEYWORDS

E-Learning, Repositories, Interoperability, Learning Objects.

1. INTRODUCTION

This paper addresses the design of a user interface for repositories of Learning Objects (LO). With this user interface, different user profiles - archivists, authors, reviewers - will manage collections of LO persisted in a repository. This design is meant for the front-end of the crimsonHex (Leal, 2009).

The crimsonHex system is a repository of specialised learning objects used in the EduJudge project. The goal of this project is to open the repository of programming problems of the UVA Online Judge (online-judge.uva.es) to pedagogical uses in secondary and higher education. The EduJudge project foresees the integration with a specific collection of services. Nevertheless, our goal is to position crimsonHex as a service provider for a larger set of systems with an automatic evaluation engine, such as a LMS or even programming contest management systems. The EduJudge project includes three types of components:

- Learning Objects Repository (LOR): to store the exercises and to retrieve those suited to a particular learner profile;
- Learning Management System (LMS): to present the exercises to the learners;
- Evaluation Engine (EE): to automatic evaluate and grade the students' attempts to solve exercises.

The remainder of this paper is organised as follows. The next section recalls the evolution of e-Learning systems with emphasis on repositories, the users' expectations regarding these systems and the features needed to make them interoperable and based in standards. In the following section we highlight one of the main components of the crimsonHex repository: the Web Manager, focusing on the management and browsing facilities it provides. Finally, we conclude with a summary of the main contribution of this work and a perspective of future work.

2. RELATED WORK

Component oriented systems are predominant in most of eLearning platforms. Despite their success, they have also been target of criticism: their tools are too general and they are difficult to integrate with other eLearning systems (Dagger et al., 2007). These issues led to a new generation of service oriented eLearning platforms, easier to integrate with other systems. One of the key services of eLearning platforms is the repository. A repository of learning objects can be defined as a 'system that stores electronic objects and meta-data about those objects' (Holden, 2004:1). The need for this kind of repositories is growing as more educators are eager to use digital educational contents and more of it is available. One of the best examples is the repository Merlot (Multimedia Educational Resource for Learning and Online Teaching). The repository provides pointers to online learning materials and includes a search engine. The Jorum Team made a comprehensive survey (Jorum Team, 2006) of the existing repositories and noticed that most of these systems do not store actual learning objects. They just store meta-data describing LOs, including pointers to

their locations on the Web, and sometimes these pointers are dangling. Although some of these repositories list a large number of pointers to LOs, they have few instances in any category, such as programming problems. Last but not least, the LOs listed in these repositories must be manually imported into a LMS. An evaluation engine cannot query the repository and automatically import the LO it needs. In summary, most of the current repositories are specialised search engines of LOs and not adequate for interact with other eLearning systems, such as, feeding an automatic evaluation engine.

Based in other surveys (Holden, 2004: 15-18) the users are concerned with issues that are not completely addressed by the existing systems, such as interoperability. Some major interoperability efforts (Hatala, 2004) were made in eLearning, such as, NSDL, POOL, ELENA/Edutella, EduSource and IMS Digital Repositories (IMS DRI). The IMS DRI specification was created by the IMS Global Learning Consortium (IMS GLC) and provides a functional architecture and reference model for repository interoperability. The IMS DRI provides recommendations for common repository functions, namely the submission, search and download of LOs. It recommends the use of web services to expose the repository functions based on the Simple Object Access Protocol (SOAP) protocol, defined by W3C. Despite the SOAP recommendation, other web service interfaces could be used, such as, Representational State Transfer (REST) (Fielding, 2000).

3. THE USER INTERFACE

The architecture of crimsonHex repository is divided in three main components:

- **the Core** exposes the main features of the repository, both to external services, such as the LMS and the EE, and to internal components - the Web Manager and the Importer;
- **the Web Manager** allows the creation, revision, versioning, uploading/downloading of LOs and related meta-data, enforcing compliance with controlled vocabularies;
- **the Importer** populates the repository with existing legacy repositories. In the remainder we focus on the Core component, more precisely, its functions, communication model and implementation.

The Core component of the crimsonHex repository provides a minimal set of operations exposed as web services and based in the IMS DRI specification. In this section we detail the Web Manager component as a Web user interface for managing a repository of learning objects. The component uses the DRI API to communicate with the core functions of the repository, promoting the interoperability with other repositories that complies with the DRI infrastructure.

In the design of the Web Manager component one of our major concerns was usability, and to promote it we followed established user interface design principles (Shneiderman, 1998). The main feature of the resulting design is the use of a single screen common to all user profiles. This type of design breaks with the traditional structure of web interfaces used by other repositories (ADL, 2003). To design this user interface we started with the identification of task and usage profiles, task objects and task actions. Then we selected a suitable interaction style and finally we created a screen layout.

3.1 User Profiles and Actions

In the start of the design process we identified the following task profiles:

- **Archivist** - a person responsible for a set of activities related with the collection management, such as the creation of collections or the assignment of learners and reviewers to collections;
- **Author** - a person that develops and submits LOs to the repository. The submission of LOs will be enforced to comply with controlled vocabularies defined in meta-data standards (IEEE LOM) and possible extensions. This class of users will contribute with new learning objects and receives reviews from specialists;
- **Reviewer** - a person that controls the quality of the repository by validating the submitted LOs;
- **Consumer**: a person that browses (part) of the repository and has limited access to its content (LO, usage reports, reviews, comments).

We assume that users will have different usage profiles. On one hand, many will be novice or first-time users, especially among authors and consumers. On the other hand, we expect some users, especially reviewers and archivists, to use it frequently, tending to become experts in its use.

After the identification of users and usage profiles we proceeded to identify the tasks they need to perform on this interface. We clearly identified LOs and collections of LOs as our task objects, each with a number of associated task actions, depending on user profiles. Task actions over LOs include: viewing, reviewing, downloading, and commenting. Task actions on collections include creating, removing and authoring/uploading (LOs to that collection).

3.2 Screen Layout

To define the screen layout of crimsonHex we sought an interaction style balancing intuitiveness and expressiveness. We first considered direct manipulation of task objects. Although it provides a convenient way to select objects, it is not possible to map all the identified task actions to basic mouse interaction (click, point and drag). We found form filling adequate for entering data for the complex tasks, such as search, authoring and reviewing. Finally, we decided to blend these two interaction styles, using a form of direct manipulation for task object selection and form-filling for executing task actions.

Based on this blended interaction style we defined a screen layout - a single screen with specific areas for task object selection and task actions. Task object selection is needed by all users, although selectable content depends on the user's profile, thus it can be implemented by a common tree-based control. Different task actions require specific forms or panels that also share a common control on the users interface. Since the number of task actions is comparatively small we chose a tabbed control to aggregate them. The tab configuration shown to users depends both on their profile and on the current task object selection.

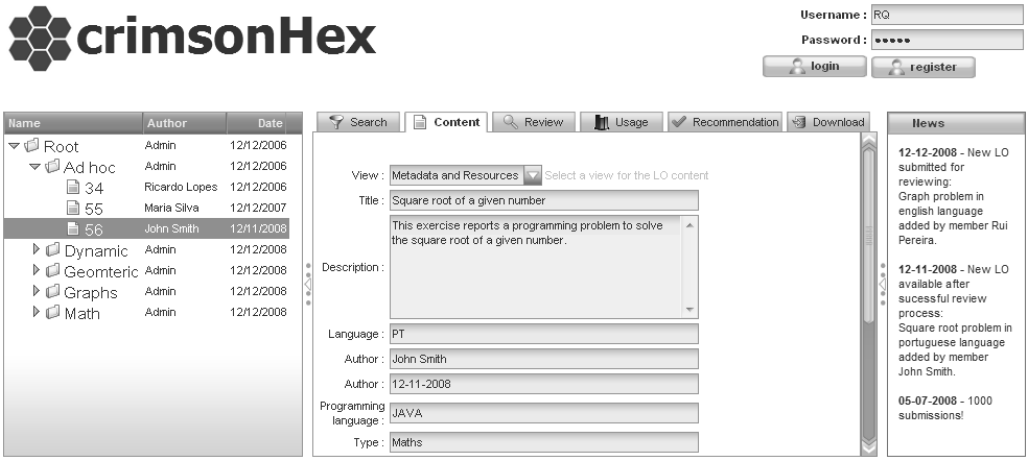

Figure 1. Component diagram of the crimsonHex repository

Figure 1 shows the user interface layout of the repository with two main areas: selection on the left side and action on middle. In the selection area the user navigates trough the repository structure to select task objects. In the action area the user executes task actions on the selected task objects. Secondary areas in this layout are the header, used for authentication and registration, and the right side, used for news and statistics.

As a rule, all available task actions have an associated tab, thus helping novice users to recognise which are the available actions. However, some of these task actions can be executed directly over selected task objects, without requiring additional data. In general, these task actions are meant for frequent users and will be bound to contextual menus on the tree-control, as well as to accelerator keys.

3.3 Implementation

The Web Manager component was developed using an Ajax framework to enable the implementation of the single screen design resulting from the last section. We selected the Google Web Toolkit (GWT), an open source Java software development framework that allows a rapid development of AJAX applications in Java.

When the application is deployed, the GWT cross-compiler translates Java classes of the GUI to JavaScript files and guarantees cross-browser portability. The framework supports also asynchronous remote procedure calls. This way, tasks that require high computational resources (e.g., complex searching within the repository) can be triggered asynchronously, increasing the user interface's responsiveness. The complex controls required by the selection and action areas are provided by SmartGWT, a GWT API's for SmartClient, a Rich Internet Application (RIA) system.

The Web Manager component is organised in two main packages: the back-end (server) and the front-end (client). The back-end includes all the service implementations triggered by the user interface. These implementations rely on the gateway class for managing the communication with the Core of the repository. A single class implementing the Gateway design pattern concentrates the interaction with the core component. To interact with other DRI compliant repositories only this class will have to be re-implemented.

4. CONCLUSION

In this paper we present the design and implementation of a management component for crimsonHex, a repository of specialized learning objects designed for interoperability with other eLearning systems. This repository is being developed as part of a project for using automatic evaluation of programming problems in the eLearning scope.

The main contribution of this work was a novel user interface design, with a single screen layout, that differs from other web based interfaces of repositories of learning objects. Also, the architecture of the resulting management component simplifies its adaption to any DRI compliant repository.

The Web Manager component is still a work in progress. Our immediate work in this project is to conclude its implementation. As future work we plan to integrate an authoring tool in the front-end of crimsonHex, a kind of feature lacking in most repositories of learning objects.

ACKNOWLEDGEMENT

This work is part of the project entitled "Integrating Online Judge into effective e-learning", with project number 135221-LLP-1-2007-1-ES-KA3-KA3MP. This project has been funded with support from the European Commission. This communication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

REFERENCES

Academic ADL Co-Lab. 2003. *Learning Repositories included in Learning Repository Investigation*, [Online], Available: http://www.academiccolab.org/resources/DraftRepositoriesList.pdf

Dagger, D., O'Connor, A., Lawless, S., Walsh, E., Wade, V., 2007. *Service Oriented eLearning Platforms:From Monolithic Systems to Flexible Services*,[Online], Available: https://www.cs.tcd.ie/~slawless/papers/ieee2007.pdf.

Fielding, R., 2000. *Architectural Styles and the Design of &etwork-based Software Architectures* (Phd dissertation), [Online], Available: http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm.

Hatala, M., Richards, G., Eap, T., Willms, J., 2004. *The EduSource Communication Language: Implementing Open &etwork for Learning Repositories and Services.* Proceedings of the 2004 ACM symposium on Applied computing.

Holden, C., 2004. *What We Mean When We Say "Repositories" User Expectations of Repository Systems*, Academic ADL Co-Lab, [Online], Available: http://www.hewlett.org/NR/rdonlyres/158FC043-A56F43C6-ABA7-zB9A62656FCB/0/RepoSurvey2004-1.pdf.

JORUM team, 2006. *E-Learning Repository Systems Research Watch*, [Online], Available: http://www.jorum.ac.uk/docs/pdf/Repository_Watch_final_05012006.pdf.

Leal, J. P., Queirós, R., 2009. *CrimsonHex: a Service Oriented Repository of Specialised Learning Objects*, ICEIS 2009: 11th International Conference on Enterprise Information Systems.

Shneiderman, B. 1998. *Designing the Users Interface - Strategies for Effective Human-Computer Interaction*, third edition, Addison-Wesley, ISBN: 0-201-69497-2.