WebLD: A Web Portal to Design IMS LD Units of Learning

Eduardo Sánchez, Manuel Lama, Ricardo R. Amorim and Angel Negrete
Department of Electronics and Computer Science
University of Santiago de Compostela
eduardos@usc.es, lama@dec.usc.es, rramorin@usc.es, angel.negrete@rai.usc.es

Abstract

This paper presents WebLD, a web authoring tool to create units of learning following the IMS Learning Design (IMS LD) specification. WebLD presents two main features that distinguish it from other tools: (1) it is a web application, therefore available anytime, anywhere from the Internet, and (2) it includes semantic technology, based on our IMS LD ontology, that enables the automatic validation of IMS LD documents.

1. Introduction

The IMS Learning Design specification (IMS LD) is an Educational Modelling Language (EML) that provides a model of semantic notation to describe both the content and processes of units of study [1]. There exist two main authoring tools to create IMS LD Units of Learning (UoLs): Reload LD Editor [2] and CooperAuthor [3]. Some limitations can be found on these tools. Because of their nature as desktop applications, they are not accessible “anywhere”, and they require software installation on every workplace. Moreover, there is no rigorous semantic validation, and it is then possible to create semantically inconsistent UoLs. To overcome these limitations, we present WebLD, a web authoring tool aimed at creating learning designs and validating them with semantic technology.

2. Requirements

The main requirements of the application are:

- FR2. Creation, list and edition of Activities.
- FR8. Creation, list and edition of Role-Parts.
- FR9. Export and import of UoLs in IMS LD format.
- FR10. Semantic validation of UoLs.

2. Design

Some popular design patterns have been used: Model-View-Controller (MVC), InterceptingFilter, FrontController, TransferObject, and Data Access Object. MVC is used as a common pattern to implement the presentation layer, the filters are used to validate and pre-processed every client request, FrontController determines a unique access point for every user action, and the other two patterns manage both data transfer and data access.

3. Implementation

Figure 1 shows the WebLD user interface. The following technologies have been used to implement the application:

- The J2EE platform has been used as the main development technology.
- Struts was chosen as the framework to implement the web/presentation tier, which means the views, forms, actions and the controller component.
- Xerces2 Java Parser 2.9.0, an XML Schema processor, to implement the IMS LD export/import component.
- MySQL Server 5.0 as the database manager.
• Jboss 4.0.5 was chosen as the application server. In this way, we allow the future implementation of the business logic with EJB 3.0 components.

• Java and F-Logic to implement the validation component. The key element in this component is the IMS LD ontology that provides axioms and rules to guarantee the semantic coherence of each UoL document [4].

5. Discussion

There are some opened issues that need to be addressed shortly: (1) the IMS LD validator component, which was implemented as a web service, and, in order to reduce the response time for the validation action, it should be integrated directly in the application; and (2) the usability of the graphical interface, which demands a learning design metaphor to facilitate the transition between the current and the new design practices.

Acknowledgements

Xunta de Galicia, for their financial support under the project PGIDT06SIN20601PR.

References


