

Exportação de lições Moodle para pacotes SCORM

Vítor Rocio

Universidade Aberta

vjr@univ-ab.pt

Edgar Vigário

Universidade Aberta

edgarvigario@gmail.com

Resumo

Este artigo incide sobre questões gerais relacionadas com a exportação de uma lição criada na plataforma Moodle para um pacote do modelo de referência SCORM (Sharable Content Object Reference Model), centrando a discussão sobre as características e especificações técnicas dos elementos intervenientes. Como exemplo de uma aplicação prática é utilizado o programa de exportação criado no trabalho da disciplina Projecto Final da Licenciatura em Informática da Universidade Aberta.

Palavras-chave: e-learning, xml, scorm

Abstract

This article focus on general questions related with the exportation of a lesson created in platform Moodle for a package of the model of reference SCORM (Sharable Content Object Reference Model), centering the quarrel on the characteristics and technical specifications of the intervening elements. As example of a practical application, the exportation program created in a work for the discipline Final Project in Computer Science – Universidade Aberta - will be used.

Keywords: e-learning, xml, scorm

1-Introdução

O desenvolvimento, implementação e disponibilidade de objectos de aprendizagem para e-learning são feitos por intermédio de LMS (Learning Management Systems), dos quais a plataforma Moodle (Modular Object-Oriented Dynamic Learning) [2] é um exemplo, que permitem a distribuição de conteúdos aos formandos e oferecem recursos para determinar o progresso e o desempenho durante a sua utilização.

Estes contentores de objectos foram lançados pela escola de Construtivismo Social, onde, entre outros, se destaca o trabalho do psicólogo Howard Gardner. Segundo Pollyana Notargiacomo Mustaro e Ismar Frango Silveira [3] “Felder, Kolb e Gardner, propuseram a implementação de anotações de metadados para identificar os diferentes estilos de aprendizagem presentes nos diferentes objectos de aprendizagem. Esta classificação constitui um ponto de partida para recuperar objectos de aprendizagem de um repositório de acordo com o perfil e experiência do educando”.

Deste modo o objecto de aprendizagem passa a ser considerado não tanto um ente independente do contexto mas como um instrumento que preferencialmente serve um determinado fim, sendo a ele agregados identificadores que indicam o objectivo a que se destina e que permitem a um sistema gestor tal como um LMS definir condições para a sua utilização.

Dada a existência de diferentes LMS, a interoperabilidade de conteúdos de e-learning tornou-se um requisito dos actuais ambientes de aprendizagem online, tendo o modelo de referência SCORM (Sharable Content Object Reference Model) [4] vindo a impor-se como factor comum das várias plataformas. Como exemplo, o projecto nacional e-U/Campus Virtual [5] adopta o SCORM para a interoperabilidade de conteúdos entre as várias instituições de ensino superior.

A construção de conteúdos em formato SCORM recorre normalmente à utilização de ferramentas de autoria e empacotamento próprias (RELOAD [6], eXact Packager [7]), sendo raros os casos em que é possível exportar para SCORM conteúdos construídos em plataformas de e-learning. A plataforma Dokeos [9] contém uma ferramenta compatível com o modelo SCORM, permitindo a criação de conteúdos neste formato, mas não permite a exportação de conteúdos criados na própria plataforma para SCORM. Noutros casos ([14]), a exportação para SCORM é uma funcionalidade exclusiva para conteúdos estáticos, não permitindo a sua extensão a outras ferramentas da plataforma. Em geral, a maioria das plataformas de e-learning permite importar conteúdos SCORM, mas não permite converter os seus próprios conteúdos naquele formato.

Este facto impede que os docentes que utilizam o LMS diariamente ou quase, para construir e transmitir conteúdos aos seus alunos, os partilhem facilmente com outros docentes e instituições que usam LMS diferentes.

Neste artigo é proposto um programa que transforma um conteúdo dinâmico do tipo “lição” no LMS Moodle [8] num pacote SCORM utilizável em qualquer LMS “SCORM-compliant”. A plataforma Moodle é uma das mais usadas a nível mundial, mas uma das suas lacunas é precisamente a falta de um mecanismo de exportação de conteúdos no formato SCORM.

Quanto à estrutura do artigo, na secção 2 é apresentado o modelo SCORM, com incidência nas características relevantes para este trabalho, a secção 3 ilustra, embora de um modo sucinto, as lições Moodle, a secção 4 apresenta o método de tradução que propomos para as converter em pacotes SCORM, na secção 5 é apresentada uma aplicação prática e, por último, na secção 6 são apresentadas conclusões e apontam-se direcções a seguir para um desenvolvimento futuro.

2-Modelo SCORM

O SCORM é um modelo de referência para cursos de e-learning baseados na WEB. Foi proposto pela iniciativa ADL (U.S. Department of Defense’s Advanced Distributed Learning Initiative), com o objectivo de promover a interoperabilidade de conteúdos entre várias organizações e os sistemas informáticos que utilizam. A especificação SCORM inclui dois grandes componentes:

- O modelo de agregação de conteúdos (CAM – Content Aggregation Model) [10]
- O ambiente de execução (RTE – Run-Time Environment) [11]

O CAM é o modelo para descrever os conteúdos, incluindo meta-dados, empacotamento e sua organização/estrutura. O RTE define uma API (Application Program Interface) e um modelo de dados para a comunicação entre o LMS e o curso SCORM.

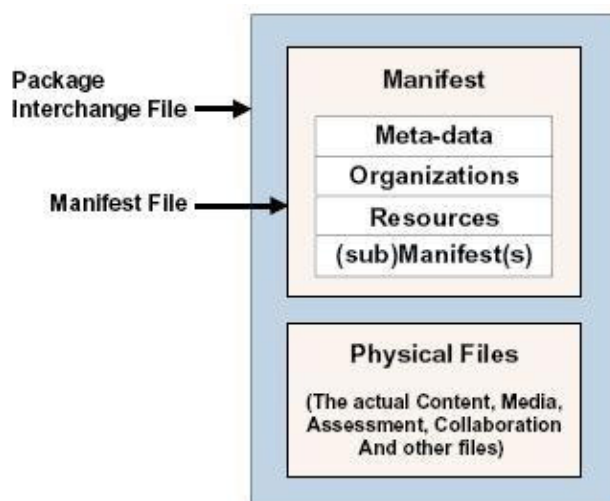


Figura 1 – Diagrama conceitual do conteúdo de um pacote SCORM

2.1-Manifesto

Um curso SCORM é materializado num ficheiro.zip (cf. Figura 1, imagem retirada de The Scorm Content Agregation Model publicada pela ADL em Outubro de 2001 [10]) com todos os recursos¹, incluindo entre eles um documento, o manifesto, que descreve o curso de acordo com o CAM. Este documento, escrito em XML, tem sempre o nome imsmanifest.xml e inclui toda a informação necessária para que qualquer sistema “SCORM-compliant” possa ministrar o curso aos formandos.

O manifesto, numa das suas configurações possíveis, pode ser constituído unicamente por dois blocos principais, o primeiro referenciando os elementos do curso e os configura em opções de menu quando da execução do pacote SCORM (informação contida entre as marcas <organizations></organizations>) e o segundo contendo a identificação dos recursos ou seja, dos ficheiros necessários para a execução da lição (informação contida entre as marcas <resources></resources>).

A figura 2 ilustra um manifesto simples, com três recursos (duas páginas HTML e um ficheiro JavaScript, usado para estabelecer uma interface com a API do SCORM). Entre as marcas <item></item> são identificadas as duas páginas, o que vai determinar o seu reconhecimento como opções de um menu quando o pacote for executado.

¹ i.e.: os ficheiros usados para mostrar os conteúdos através de um web browser – ficheiros HTML, imagens, vídeos, animações, etc.

```
<?xml version="1.0" encoding="UTF-8"?>
<manifest identifier="MANIFEST-XXX">
  <organizations default="ORG-XXX">
    <organization identifier="ORG-XXX" structure="hierarchical">
      <title>Organization</title>
      <item identifier="ITEM-19" isvisible="true" identifierref="RES-19">
        <title>decisao</title>
      </item>
      <item identifier="ITEM-20" isvisible="true" identifierref="RES-20">
        <title>maquina</title>
      </item>
    </organization>
  </organizations>
  <resources>
    <resource identifier="RES-19" type="webcontent" href="19.htm">
      <file href="19.htm" />
    </resource>
    <resource identifier="RES-20" type="webcontent" href="20.htm">
      <file href="20.htm" />
    </resource>
    <resource identifier="RES-APIJava" type="webcontent" href="APIWrapper.js">
      <file href="APIWraper.js" />
    </resource>
  </resources>
</manifest>
```

Figura 2 – Exemplo de documento imsmanifest.xml

2.2-API do SCORM

A API do SCORM é o mecanismo de comunicação utilizado para enviar e receber dados entre o Learning Management System (LMS) da plataforma de aprendizagem e o pacote SCORM. Estes dados definidos pelo Data Model são um conjunto padronizado que se espera reconhecíveis pela plataforma e pelo modelo.

O pacote SCORM, por intermédio da API, acede a identificadores do LMS que referem recursos de uma gestão de aprendizagem como a pontuação, número de tentativas por página para um determinado aluno, objectivos a concretizar, limites de tempo, etc. (figura 3).

Do estudo efectuado foram definidos três tipos de utilização da API (diferentes gradações de cinzento na figura 4), determinados pela informação que pretendemos exportar em conjunto com os conteúdos da lição. Assim, o terceiro tipo é definido pela utilização de um vasto leque de identificadores do LMS e de funções da API. O segundo tipo é definido por uma parca utilização quer dos identificadores, sendo estes normalmente definidos pelo programador para o efeito pretendido, quer de funções da API, como é exemplificado na figura 8, onde são utilizados os identificadores `cmi.objectives.0` a `cmi.objectives.n` e as funções do `LMSSetValue` e do `LMSInitialize` e `LMSFinalize`. O primeiro tipo é definido por não serem utilizados nenhuns recursos nem funções, exportando-se unicamente os conteúdos da lição.

Deste modo, consoante a informação que pretendemos exportar, determinamos o número e o tipo dos recursos acedidos e o conjunto de marcas do documento `moodle.xml` que

necessitamos de considerar e avaliar. A título de exemplo do referido sobre as marcas a avaliar no documento, atente-se na figura 5, onde o texto com o fundo em sombreado realça as marcas utilizadas na elaboração de um programa que unicamente exporta os conteúdos de uma lição.

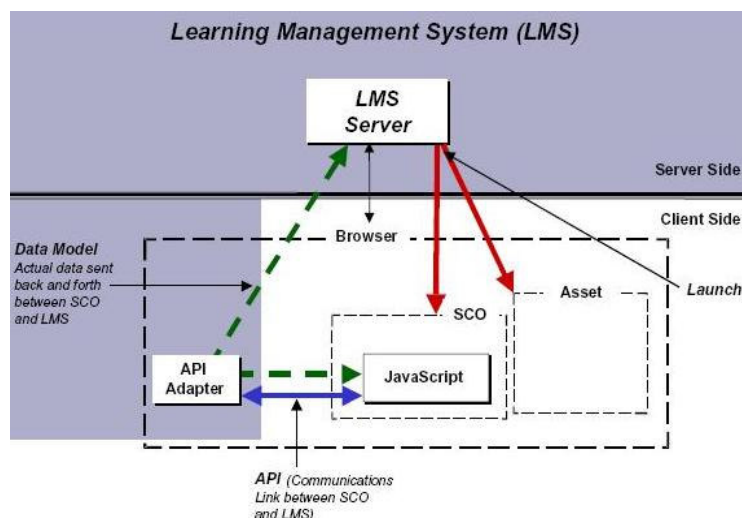


Figura 3 – API e modelo de dados no ambiente de execução SCORM

| recursos da API utilizados | informação da plataforma Moodle a exportar |
|---|---|
| <p>{funções API Scorm}</p> <p>{identificadores LMS}</p> | <p>conteudos de uma lição</p> <p>+ gestão utilizadores</p> |
| <p>{LMSGetValue(),LMSSetValue(),LMSInitialize()}</p> <p>{cmi.objectives.n.id}</p> | <p>conteudos de uma lição</p> <p>+ sistema de pontuação</p> |
| <p>{0}</p> | <p>conteudos de uma lição</p> |

Figura 4 – Relação entre informação da plataforma a exportar e os recursos da API

3- Lições na Plataforma Moodle

A plataforma Moodle permite aos docentes a criação de tipos de recursos e actividades que podem disponibilizar aos seus alunos no âmbito das disciplinas que leccionam.

Existem actividades que não se adequam ao modelo SCORM, como fóruns de discussão, chats, trabalhos (i.e. espaço para upload de ficheiros), no entanto, uma das actividades do Moodle, a lição, pode ser traduzida num pacote SCORM.

As lições incluem páginas HTML estáticas e interactivas, definindo percursos possíveis de aprendizagem. O percurso é definido pelo aluno, quer pela escolha explicita do caminho a seguir, quer pelas respostas dadas a questões interactivas apresentadas nas páginas (que definem implicitamente o percurso).

Uma lição Moodle é assim constituída por uma série de páginas HTML que podem ser ou um menu de escolhas possíveis, ou uma “página de perguntas” de um dos seguintes tipos: escolha múltipla; verdadeiro/falso; resposta curta; resposta numérica; comparação; ensaio.

Quanto às respostas, estas podem também ser ou não pontuadas; se o forem, é atribuída ao aluno uma classificação no final da lição.

3.1- Ficheiro de backup

Quando do procedimento de salvaguarda de uma disciplina, a plataforma Moodle cria um documento XML contendo informações relativas à disciplina e aos seus constituintes (elementos do grupo de estudo, lições, pacotes SCORM, imagens, etc.). Quando do procedimento de salvaguarda de uma disciplina, a plataforma Moodle cria um documento XML contendo informações relativas à disciplina e aos seus constituintes (elementos do grupo de estudo, lições, pacotes SCORM, imagens, etc.).

```
<?xml version="1.0" encoding="UTF-8"?>
<MOODLE_BACKUP>
  <INFO>
    <NAME>backup-mecan1-23072006-1302.zip</NAME>
    <MOODLE_VERSION>2005060240</MOODLE_VERSION>
    <MOODLE_RELEASE>1.5.4</MOODLE_RELEASE>
    <BACKUP_VERSION>2005070500</BACKUP_VERSION>
    <BACKUP_RELEASE>1.5.3</BACKUP_RELEASE>
    <DATE>1153656166</DATE>
    <ORIGINAL_WWWROOT>http://localhost/moodle</ORIGINAL_WWWROOT>
    <ZIP_METHOD>internal</ZIP_METHOD>
    <DETAILS>
      <MOD>...</MOD> ...
    </DETAILS>
  </INFO>
  <COURSE>
    <HEADER> ... </HEADER>
    <BLOCKS> <BLOCK>... </BLOCK> ... </BLOCKS>
    <SECTIONS> <SECTION> ... </SECTION> ... </SECTIONS>
    <USERS> <USER> ... </USER> ... </USERS>
    <GRADEBOOK></GRADEBOOK>
    <MODULES>
      <MOD> ... </MOD> ...
      <MOD> ... <PAGES> <PAGE> ... </PAGE> ... </PAGES> </MOD>
      <MOD> ... <SCOE> <SCO> ... </SCO> ... </SCOE> </MOD>
    </MODULES>
  </COURSE>
</MOODLE_BACKUP>
```

Figura 5 – Marcas constituintes do documento moodle.xml

Quando do procedimento de salvaguarda de uma disciplina, a plataforma Moodle cria um documento XML contendo informações relativas à disciplina e aos seus constituintes (elementos do grupo de estudo, lições, pacotes SCORM, imagens, etc.).

Tabela 1 – Marcas que contêm informação sobre diferentes tipos de página e respostas

| | Marca | Identifica |
|----------|------------|--|
| Página | PAGEID | Numero de página determinado pelo Moodle |
| | QTYPE | Tipo de página |
| | QTOPTION | Subtipo de página |
| | TITLE | Titulo da página |
| | CONTENTS | Texto da pergunta |
| Resposta | ID | Numero de página determinado pelo Moodle |
| | JUMPTO | Próxima página a ser executada |
| | SCORE | Valor da resposta |
| | ANSWERTEXT | Texto da resposta |

Como a informação da lição que pretendemos exportar está neste documento referenciada, é necessário, antes da construção de um programa de exportação, efectuar um estudo prévio sobre o que as marcas referenciam e o que o seu conteúdo determina. Como exemplo de um estudo deste tipo são apresentadas, na tabela 1, as marcas contidas entre <PAGE></PAGE> e <ANSWER></ANSWER> que contêm informação sobre os diferentes tipos de páginas e respostas. As marcas QTYPE e QTOPTION definem o tipo de página de acordo com a tabela 2.

Tabela 2 – Tipos de páginas identificados pelos valores QTYPE e QTOPTION

| QTYPE | QTOPTION | Tipo de página |
|-------|----------|---|
| 1 | 0 | Resposta curta |
| 1 | 1 | Resposta curta – distingue maiúsculas de minúsculas |
| 2 | 0 | Verdadeiro ou falso |
| 3 | 0 | Escolha múltipla |
| 3 | 1 | Escolha múltipla – Múltiplas respostas |
| 5 | 0 | Comparação |
| 8 | 0 | Resposta numérica |
| 10 | 0 | Ensaio |
| 20 | 0 | Menu |

A marca JUMPTO pode ter como valores -1, 0, 9, -40, ou um número de outra página constituinte da lição, tendo como acções consequentes executar a página seguinte, executar a mesma página, terminar a lição, executar a página anterior ou executar a página cujo conteúdo de PAGEID corresponde ao numero indicado.

4-Elementos e Especificações Técnicas Condicionantes (ETC)

São considerados como elementos a lição do Moodle, o pacote SCORM, a API do SCORM e o programa de exportação (figura 6). Como ETC definem-se especificações técnicas de interesse na concepção e execução do programa de exportação. No pacote SCORM, as ETC decorrem da configuração do elemento ou seja a existência de um conjunto de ficheiros considerados recursos e de um documento XML que corresponda à descrição feita em 2.1.

No modo de utilização da API as ETC são o próprio modo (ver 2.2). Nas lições Moodle as ETC são a necessidade da existência de diferentes páginas WEB que sejam de um dos sete tipos indicados na secção 3.1 e que se mimetize a sua sequência de execução.

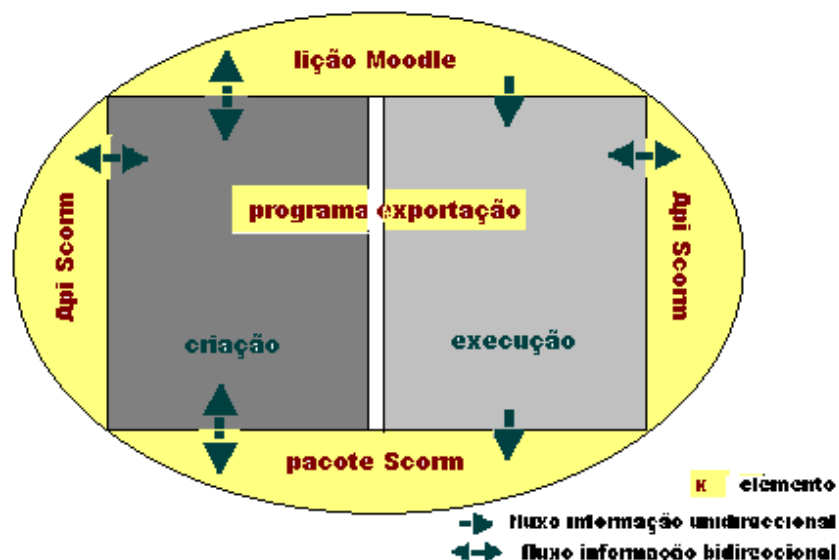


Figura 6 – Diagrama conceitual para a elaboração e execução de um programa de exportação com os diferentes elementos identificados e fluxos de informação

4.1-Programa de Exportação

Este elemento gere as ETC dos outros elementos, com o objectivo de na sua execução ler um documento moodle.xml (que contém a lição) e produzir um conjunto de ficheiros com as características necessárias para a construção de um pacote SCORM.

Temos assim que considerar quando da construção do seu código:

- o tipo de utilização da API, o que determina desde logo a escolha das informações consideradas relevantes no documento moodle.xml;
- como criar ficheiros que reproduzam o conteúdo e tipo das páginas WEB da lição bem como o seu fluxo de execução;
- o modo de construir o documento imsmanifest.xml.

É também necessário considerar que na sua execução o programa tem de:

- ler o documento moodle.xml, criar estruturas de dados para salvaguarda das informações consideradas relevantes obtidas na leitura e desprezar as outras;
- a partir da informação salvaguardada nas estruturas de dados, construir ficheiros numa linguagem WEB que simulem as páginas da lição do Moodle e o seu modo de execução;
- construir o documento imsmanifest.xml.

5-Exemplo de uma aplicação prática

O objectivo desta secção consiste em apresentar uma solução prática que torne mais clara a discussão anterior, utilizando para tal o programa referido no resumo que exporta lições da plataforma Moodle (versão 1.5) para o modelo de referência SCORM.

Na concepção deste programa, em primeiro lugar, optou-se por uma utilização da API do segundo tipo como definido em 2.2, em que unicamente eram exportados os conteúdos de uma lição sendo criado um sistema de pontuações. Em seguida, foi efectuado um estudo sobre como referenciar as diferentes páginas de uma lição no documento moodle.xml.

Por fim, foi criado o programa em C# que, através de um processo em que são construídos os diversos ficheiros em código HTML, efectua a exportação. Neste programa, foram definidas duas classes, uma de páginas e outra de respostas cujos atributos propriedades e métodos são listados na figura 7, e, como facilmente se verifica, reflectem quase na íntegra a estrutura de marcas existentes no moodle.xml (quadro 1), com a excepção dos atributos *pagant* e *pagsgt*, necessários para a construção de uma sequência de execução quando esta não se encontrar estabelecida na lição original.

| |
|---|
| <p>Página</p> <p>pageid: string pagsgt: string pagant: string qtoption: string title: string contents: string lanswers: List<resposta></p> <p>pagina(string pageid, string qtoption, string title, string contents, List<resposta> lanswers)</p> <p>string rspcorrectas()</p> |
| <p>Resposta</p> <p>id: string jumpto: string score: string answertext: string response: string</p> <p>pagina(string id, string jumpto, string score, string answertext, string response)</p> |

Figura 7 – Atributos, construtores e métodos das classes: página e resposta

Na sua execução o programa segue as actividades referidas em 4.1. Na primeira, lê-se o documento moodle.xml com recurso ao construtor da classe XmlNodeReader. XmlNodeReader [12] e “preenchem-se” as estruturas de dados (figura 7). Na segunda, procede-se a uma identificação do fluxo de execução da lição com o preenchimento dos atributos *pagant* e *pagsgt* da classe página construindo-se uma página final para apresentar as pontuações.

Na terceira, constroem-se os ficheiros HTML e as funções JavaScript neles embebidas a partir do conteúdo das estruturas de dados, construindo-se um ficheiro HTML para cada página no qual são inclusas diferentes funções JavaScript², embebidas consoante o tipo de página, sendo no entanto comuns as funções que asseguram a “ligação” à API do SCORM (figura 8).

Convém referir que para a utilização destas funções é necessária a inclusão do ficheiro APIWrapper.js [13] da autoria da Concurrent Technologies Corporation que estabelece uma interface entre a API e os ficheiros HTML (figura 3).

Na quarta, é construído o ficheiro imsmanifest.xml, tendo-se optado por uma configuração simples do ficheiro imsmanifest.xml semelhante à da figura 2.

Para finalizar esta secção são apresentadas as figuras 9 e 10, onde se exhibe, na primeira, uma página WEB executada na plataforma Moodle e, na segunda, a mesma página executada na plataforma Intralearn da Universidade Aberta. O pacote SCORM que contém estas páginas foi gerado pelo programa de exportação e carregado em cada uma das plataformas.

```
function iniciaLMS(){ doLMSInitialize();doLMSSetValue("cmi.objectives.0.score.raw","0"); ...
doLMSSetValue("cmi.objectives.n-1.score.raw","0"); }
function scoreLMS(){ doLMSSetValue("cmi.objectives.n.score.raw","0"); }
function terminaLMS(){doLMSFinalize();}
function mudaSCORE(pontuacao){doLMSSetValue("cmi.objectives.n.score.raw",
pontuacao.toString()); }
```

Figura 8 – Funções que no programa utilizam a API do SCORM

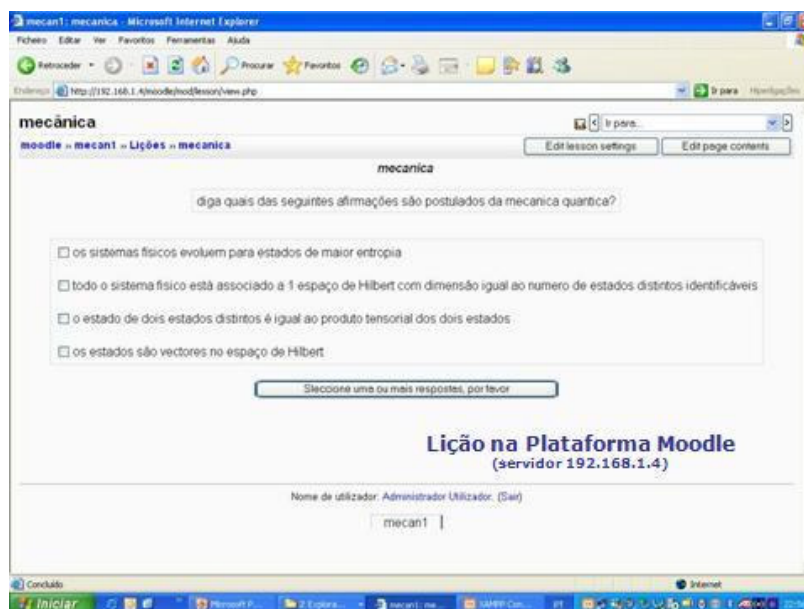


Figura 9 – Lição na plataforma Moodle (servidor 192.168.1.4)

² Escolheu-se esta linguagem por se pretender utilizar uma linguagem WEB que pudesse ser utilizada tanto num servidor como num cliente

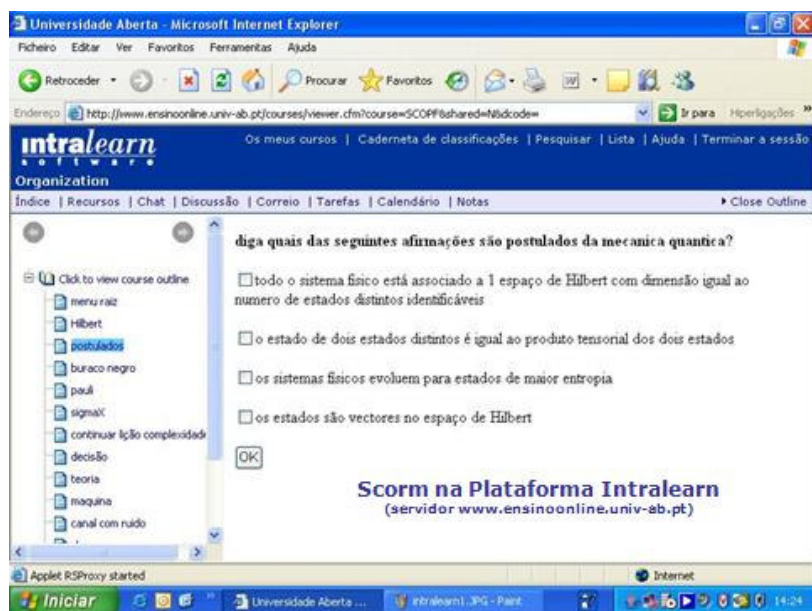


Figura 10 – Lição na plataforma intralearn (servidor www.ensinoonline.univ-ab.pt)

6-Conclusão

Neste artigo, definiu-se uma estratégia para a exportação de lições da plataforma Moodle baseada numa análise do conteúdo do ficheiro de backup e na escolha de um modo de utilização da API do modelo de referência SCORM, tendo sido definidas e identificadas as especificações técnicas que a condicionam.

A exportação de conteúdos em formato SCORM promove a interoperabilidade dos mesmos em várias plataformas, como demonstrámos no caso particular das lições e das plataformas Moodle e Intralearn. A conversão da estrutura da lição Moodle em formato SCORM não implica apenas uma conversão de ficheiros XML, mas a construção de um pacote com vários recursos web, além do manifesto SCORM. Por esta razão, o projecto foi implementado com recurso a uma linguagem imperativa, em detrimento de linguagens de transformação XML (por exemplo XSLT [15] ou FXT [16]).

Embora não tenha sido feito um estudo para um modo de utilização da API do terceiro tipo, como definido em 2.2, pressupõe-se que soluções técnicas apresentadas neste artigo para o primeiro e segundos tipos, tais como a tradução das diferentes páginas constituintes numa lição em ficheiros executáveis numa linguagem WEB, inserido-lhe as funções da API que permitem utilizar o LMS da plataforma e a implementação de estruturas de dados que salvaguardem as informações consideradas relevantes para a exportação, podem para ele ser extrapoladas.

Em desenvolvimentos futuros, pretendemos abordar a inclusão de ficheiros no conteúdo textual das lições como é, por exemplo, o caso em que no texto de uma pergunta ou resposta está incluída uma imagem, pormenor que não afectando em muito a estratégia de exportação é bastante importante para uma utilização prática da mesma.

Estando a plataforma Moodle implementada em PHP, torna-se também interessante a implementação do programa de exportação nesta linguagem, como inclusão no software da própria plataforma. Uma outra possibilidade futura é a extensão do programa para lidar com outro tipo de recursos (mini-testes, referendos).

Como principais dificuldades encontradas na realização do programa de exportação, podemos referir a grande quantidade de informação existente no documento moodle.xml e o modo como a extracção desta informação se dificulta quando pretendemos exportar recursos da LMS relativos à gestão dos alunos. Estas dificuldades levaram à definição dos três modos de utilização da API e a que se tenha optado por um modo de utilização do segundo tipo. Refere-se também a grande flexibilidade que a plataforma Moodle permite na construção de lições, o que obrigou à construção de alternativas para colmatar informação que não podia ser recuperada do documento moodle.xml.

Como exemplo, podemos considerar a já previamente discutida introdução na classe página dos atributos *pagant* e *pagseg*, aliada a um rastreamento prévio no documento dos redireccionamentos existentes para contornar situações em que a sequência de execução da lição na plataforma foi construída de modo a ser exclusivamente gerida pelo browser, onde é executada.

Bibliografia

[1] Edgar Vigário, Exportação de lições criadas na plataforma Moodle para pacotes SCORM - Relatório do trabalho da disciplina Projecto Final do curso de Licenciatura em Informática da Universidade Aberta, Julho 2006.

[2] Moodle (<http://moodle.org>)

[3] Pollyana Notargiacomo Mustaro e Ismar Frango Silveira. Learning Objects: Adaptive Retrieval through Learning Styles, Interdisciplinary Journal of Knowledge and Learning Objects, Volume 2, 2006

[4] Iniciativa ADL (<http://www.adlnet.gov/>)

[5] Projecto e-U/Campus Virtual (<http://www.e-u.pt/>)

[6] Ferramenta RELOAD (<http://www.reload.ac.uk/>)

[7] Giunti Labs eXact Packager (<http://www.giuntilabs.com/info.php?vvu=13>)

[8] Jason Cole. Using Moodle, O' Reilly, Julho 2005

[9] Dokeos 1.8: The SCORM machine (<http://www.dokeos.com/scorm.php>)

[10] The SCORM Content Aggregation Model, Advanced Distributed Learning Technical Team, Outubro 2001

[11] The SCORM Run-Time Environment, Advanced Distributed Learning Technical Team, Outubro 2001

[12] XmlNodeReader Class

(<http://msdn2.microsoft.com/en-us/library/system.xml.xmlnodereader.xmlnodereader.aspx>)

[13] SCORM General Common Questions

(<http://www.adlnet.gov/help/CommonQuestions/SCORMGeneralQuestions.aspx>)

[14] SCORM export for the E-Portfolio add-on. <http://www.moodlekurse.org> (Login as guest)

[15] XSL Transformations (<http://www.w3.org/TR/xslt>)

[16] XStream (<http://xstream.codehaus.org/>)