



Universidade do Porto
Faculdade de Engenharia

FEUP



Rúben Filipe Melim de Nóbrega

Ferramentas para formação à medida em sistemas de e-Learning

004(047.3) LEIC
EIC5202 2004/NOBr

Porto, 2004

Faculdade de Engenharia da Universidade do Porto
Licenciatura em Engenharia Informática e Computação



Ferramentas para formação à medida em sistemas de e-Learning

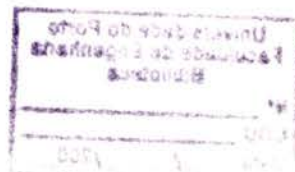
Relatório do Estágio Curricular da LEIC 2003/2004

Rúben Filipe Melim de Nóbrega

Orientador na FEUP: Francisco José de Oliveira Restivo

Orientador no uTICM: Carlos Manuel Cardoso Oliveira

Outubro de 2004



024 (047.3) LEC EIC5202 2004/NDBA

Universidade do Porto	
Faculdade de Engenharia	
Biblioteca	
Nº	81500 7
CDU	004.41 (047.3)
Data	20 / 03 / 2006

Para os meus familiares, em particular aos meus pais e irmão, pelos valores com os quais me educaram, pois agora poder-se-ão orgulhar ainda mais do seu filho e irmão.

Obrigado por tudo!

Para a Cátia, por me ter ajudado a ultrapassar tantas tormentas e por me dar sempre uma razão para acreditar em mim próprio.

Adoro-te!

Para os meus amigos, por estarem comigo nos bons e maus momentos.

Bem Hajam!

Resumo

Este relatório refere-se ao estágio intitulado “Ferramentas para formação à medida em sistemas de e-Learning”, realizado na *FEUP/uTICM*, Porto, no âmbito dos estágios da Licenciatura de Engenharia Informática e Computação, da Faculdade de Engenharia da Universidade do Porto (FEUP).

Este projecto permitiu especificar dois *plugins* complementares que estabelecem a ligação entre um sistema de *e-Learning*, a biblioteca digital e o sistema de informação, do ponto de vista comunicacional.

O primeiro *plugin* chama-se *Course Readings* e faz a integração de um sistema externo (sistema de gestão de bibliotecas) com um sistema de e-Learning (LUVIT LMS [11]), retornando a bibliografia recomendada de uma disciplina.

O segundo *plugin* chama-se *Content Readings* e adaptando o mecanismo do plugin *Course Readings*, implementará o funcionamento de um sistema de apoio à decisão, mostrando um conteúdo dependente de uma análise (*mining*) no *datawarehouse* (armazém de dados) [18].

Neste relatório, são discutidos o problema e os seus pressupostos, as tecnologias usadas, a arquitectura e a solução proposta.

Em termos gerais, este projecto permitiu:

- O estudo da tecnologia utilizada na *LUVIT LMS*;
- O estudo das normas para e-Learning *IMS/IEEE/ADL*, nomeadamente *IMS/CP*, *SCORM/RTE*;
- Implementar bibliografia em *real-time* para ser usada num sistema de e-Learning;
- Implementar o conceito de sistema de apoio à decisão em real-time, na escolha dos conteúdos mais adequados, para otimizar a aprendizagem por e-Learning;
- Desenvolver uma arquitectura para a inclusão *Just-In-Time (JIT)* de conteúdos *SCORM* [13], num curso LUVIT com base em requisitos especificados por *Simple Sequencing* [10];
- Investigação das *Redes Bayesianas* [3], estabelecendo intersecções entre a abordagem estatístico-Bayesiano e a manipulação da incerteza;
- A especificação de um ambiente computacional para aprendizagem de Redes Bayesianas, considerando as variáveis mais relevantes para a tomada de decisão, sendo um grande avanço em pesquisas na área do e-Learning;

Agradecimentos

Agradeço a toda a equipa da uTICM por todo o seu apoio, amizade e contagiante boa disposição. Em particular, agradeço ao Eng. Carlos Oliveira pela oportunidade que me proporcionou em estagiar na uTICM.

Agradeço ao Manuel Domingues e ao Pedro Faria por terem proporcionado os meios de instalação do plugin Course Readings e por me terem ensinado o verdadeiro conceito de engenharia. Sem eles, jamais este projecto teria sido concretizado. Obrigado pela vossa amizade!

Não posso deixar de referir a Fátima Lucas pelo design do poster apresentado no ICAU e ao Paulo Jesus pelo poster de estágio para a Semana de Engenharia. O meu Muito Obrigado pelo vosso sentido de estética!

Agradeço a todas as pessoas da FEUP que de algum modo apoiaram a realização do trabalho, ou que contribuíram para o seu sucesso. Um Muito Obrigado em especial para o Eng. Francisco Restivo pelo seu interesse imediato em apoiar e orientar este projecto inovador.

Finalmente, agradeço o financiamento n.º 306.012/03 – Concurso n.º 1/3.2/PRODEP/2003, correspondente ao PRODEP (Programa de Desenvolvimento Educativo para Portugal) do Ministério da Educação.

Índice de Conteúdos

1	Introdução.....	1
1.1	Apresentação da Instituição de Estágio LUVIT AB & FEUP/uTICM	1
1.2	O Plugin Course Readings na Instituição de Estágio LUVIT AB & FEUP/uTICM	3
1.3	O Projecto SADPEL na Instituição de Estágio LUVIT AB & FEUP/uTICM	3
1.4	Organização e Temas Abordados no Presente Relatório.....	4
2	Análise do problema.....	5
2.1	Plugin Course Readings	5
2.2	SADPEL (Sistema de Apoio à Decisão de uma Plataforma de e-Learning)	9
3	Revisão Tecnológica	12
3.1	Soluções Existentes ou Semelhantes.....	12
3.2	Outras Tecnologias.....	22
4	Descrição da solução proposta/Especificação.....	25
4.1	Especificação da Arquitectura.....	25
4.2	Solução Proposta.....	30
5	Desenvolvimento do Protótipo.....	42
5.1	Plugin Course Readings	42
5.2	Plugin Content Readings	45
6	Conclusões e Considerações Gerais	46
6.1	Avaliação de Resultados	47
6.2	Futuros Desenvolvimentos.....	48
6.3	Avaliação Geral do Estágio.....	48
	Referências e Bibliografia.....	49
	ANEXO A: Glossário.....	50
	ANEXO B: Apresentação do Plugin Course Readings na conferência do ICAU – Porto, Portugal, 6 de Setembro de 2004.....	54
	ANEXO C: Artigo aceite para apresentação no “e-LES’04” – Aveiro, 29 de Outubro de 2004.....	59
	ANEXO D: Slides da apresentação do artigo no “e-LES’04” – Aveiro, 29 de Outubro de 2004 62	

1 Introdução

1.1 Apresentação da Instituição de Estágio LUVIT AB & FEUP/uTICM

1.1.1 O que é a LUVIT AB?

A LUVIT [11] é uma empresa sueca de *software* cuja ênfase é na pedagogia e ensino. O *software* da LUVIT é usado para a gestão eficiente do ensino online.

LUVIT é um acrónimo para *Lund University Virtual Interactive Tool*.

A LUVIT não desenvolve o conteúdo dos cursos, mas fornece uma ferramenta técnica e pedagógica para uma forma de ensino baseado na Web. Também fornece apoio e serviços de ensino a clientes e utilizadores. Em cooperação com os seus parceiros, a LUVIT também oferece o conteúdo dos cursos, na forma de cursos já preparados ou formação personalizada.

A LUVIT AB é detida maioritariamente por três empresas. The Volito Group (empresa de investimentos), Futuremedia Plc (fornecedor de soluções de e-Learning) e a Universidade de Lund.

A empresa está sediada em Lund na Suécia.

O mercado de actuação da LUVIT é predominantemente o Europeu concentrando-se principalmente na região Nórdica, Portugal e Espanha. Estes países foram escolhidos com base no seu potencial de mercado, tendo em consideração as relações com os clientes já existentes e os mercados onde existe a procura pelos produtos multilingue da LUVIT.

1.1.1.1 A História da LUVIT AB

O *software* da LUVIT teve origem num projecto conduzido pela Universidade de Lund em 1997, quando a procura por ensino à distância era grande. O objectivo do projecto era desenvolver uma plataforma de ensino baseada na Internet.

A ferramenta tinha que ser simples de usar, mesmo para iniciados em computador, muito flexível e ser apropriada para uso na maioria das áreas das matérias e com os principais métodos pedagógicos.

A universidade de Lund descobriu que a ferramenta de ensino tinha qualidades que tornavam possível a comercialização do projecto e a empresa LUVIT foi formada em Novembro de 1988. Desde Março de 1999, a empresa tem sido operada sob o nome LUVIT AB e a partir de Março de 2000, como LUVIT AB (publ.).

1.1.1.2 Parceiros e Clientes

Os principais parceiros da LUVIT são a Futuremedia, a Microsoft e a Hewlett Packard.

Em Abril de 2003, a LUVIT entrou numa cooperação próxima com a Futuremedia Plc, uma empresa de e-Learning sediada no Reino Unido e listada no Nasdaq. A Futuremedia oferece, desde 1982, uma solução completa para ensino baseado na Web que tem em consideração os requisitos dos utilizadores, processo de formação e tecnologia.

O estatuto da LUVIT como Parceira de Ouro Certificada da Microsoft para Aplicações de Software, (Microsoft Gold Certified Partner for Software Applications), é o resultado dos investimentos de qualidade da empresa, de modo a oferecer o nível de qualidade que os clientes da LUVIT exigem. Esta certificação, é um passo na direcção de uma cooperação próxima entre a LUVIT e a Microsoft, e a primeira fase do foco estratégico da LUVIT nas plataformas Microsoft e .NET.

Desde 2000, a LUVIT e a Compaq Computer i Sverige, agora chamada Hewlett Packard Sverige (HP), têm tido um acordo de parceria que engloba várias áreas. Os resultados principais da parceria são que o software da LUVIT, com a ajuda da HP, têm sido testado e optimizado para as soluções de hardware da HP e a LUVIT pode oferecer aos seus clientes soluções gerais benéficas, consistindo tanto em hardware como em software.

Os clientes-alvo da LUVIT são diversificados e com diversos tipos de requisitos.

Nestes podem-se incluir empresas que necessitam de fornecer formação interna progressiva, cursos eficazes nos custos para os fornecedores e distribuidores, e necessitam de monitorizar os resultados e progressos dos indivíduos. Por exemplo, empresas que estão a lançar novos produtos e empresas que necessitam de aumentar a eficiência interna dos custos.

A base de clientes da LUVIT, consiste em cerca de 70 clientes activos, grandes e pequenos, incluindo empresas como a Sony Ericsson Mobile Communication, Volvo, Boehringer Ingelheim, TeliaSonera, Afnor, etc. A nível do ensino superior, têm como clientes várias das melhores universidades de diversos países tais como: Suécia, Bélgica, China, França, Noruega, Dinamarca, Lituânia, Espanha, Finlândia, Suíça, Portugal e Chile. Actualmente, em Portugal, o único cliente é a Universidade do Porto.

1.1.1.3 Área de Negócio

A LUVIT direcciona o seu negócio a dois segmentos de mercado: as empresas, e as universidades e escolas.

1.1.2 O que é o uTICM?

O *uTICM* (*Unidade das Tecnologias da Informação e Comunicação Multimédia*) é uma unidade da Faculdade de Engenharia da Universidade do Porto, que entre outros serviços, garante a administração e o suporte da plataforma de e-Learning LUVIT LMS da Universidade do Porto. Aparece neste trabalho como ligação à LUVIT AB, pois possui um acordo de *Technical Partnership*, disponibilizando o software e documentação técnica necessários.

1.2 O Plugin Course Readings na Instituição de Estágio LUVIT AB & FEUP/uTICM

Os conceitos de formação ao longo da vida e a formação à medida são cada vez mais importantes na sociedade do conhecimento em que hoje nos integramos. Necessitou do desenvolvimento de novos métodos e ferramentas para considerar esse conhecimento.

É desse ponto de vista que foi implementado um plugin que faz a integração do *Learning Management System* da LUVIT (sistema de e-Learning) com o *ALEPH* [15] (sistema de gestão de bibliotecas utilizada pela biblioteca da FEUP), e retorna a bibliografia recomendada de uma disciplina através do código desta.

Para o estagiário, a implementação do plugin foi gratificante para o início do *SAD*, pois o mesmo, para além de fornecer um serviço muito útil aos docentes e alunos, serviu de suporte para testar alguns módulos da LUVIT utilizando para tal a sua *API*, mais concretamente: a *navigation tree* e os *menus* da interface *Web*.

1.3 O Projecto SADPEL na Instituição de Estágio LUVIT AB & FEUP/uTICM

O aumento das necessidades de formação das empresas, está a ser conseguido tanto por cursos internos como externos. As empresas, através do investimento em plataformas de ensino, podem, numa maior dimensão, garantir a sua própria oferta de conhecimento através de formação interna.

O uso do *software* da LUVIT permite a redução dos cursos totais de ensino. Permite aos colaboradores terem acesso a ambientes de ensino fáceis de usar e disponíveis em várias línguas, o que é pouco importante para organizações multinacionais.

As universidades e escolas, têm uma grande necessidade de ensino baseado na *Web* por várias razões. A competição entre universidades tem aumentado ao longo dos últimos anos, porque a tecnologia tornou possível disponibilizar cursos online a estudantes localizados em lugares distantes. Ao mesmo tempo, os educadores podem trabalhar em universidades situadas a maiores distâncias das suas casas. A procura dos estudantes por ensino à distância aumentou nos últimos anos e esta tendência continua à medida que estes se apercebem dos benefícios desta nova forma de ensino quando comparada com formas mais tradicionais.

Utilizando o conhecimento adquirido da integração do plugin *Course Readings*, que é um complemento base para a criação de outras ferramentas, está a ser concebido um sistema de apoio à decisão, que será representado pelo *plugin Content Readings*, e apresentará o conteúdo mais adequado a um formando, de acordo com as necessidades deste – *Just-In-Time Learning (JIT)*.

O objectivo geral, é o de fornecer aos formadores da LUVIT melhores meios para organizar, aceder e desenvolver uma vasta gama de informação e estratégias alternativas, e estabelecer posições informais e fulcrais em relação a problemas maiores.

Mais especificamente, o objectivo, é o de organizar o conteúdo de acordo com um esquema específico de sequenciamento, fornecendo um método ordenado para apresentar o conteúdo que será mais provável corresponder às expectativas do formando.

1.4 Organização e Temas Abordados no Presente Relatório

O relatório é composto por seis Capítulos e quatro Anexos.

O capítulo 1 (Introdução), descreve todo o contexto do estágio, apresenta o tema e conclusões do estágio (com mais algum detalhe do que no resumo) e é referida a organização do relatório. É também apresentada a instituição de estágio, o projecto e o problema concreto que o estágio abordou.

O capítulo 2 (Análise do Problema) efectua uma apresentação do problema, enquadrando-o num contexto mais global; quais são os pressupostos, o que se espera obter e quais são os vários sub-problemas em que se pode desdobrar.

O capítulo 3 (Revisão Tecnológica) apresenta algumas soluções conhecidas para o problema ou para problemas semelhantes. Inclui também uma crítica qualitativa/comparativa e justifica a necessidade de desenvolvimento da solução proposta.

O capítulo 4 (Descrição da solução proposta/Especificação) especifica a solução proposta, conjuntamente com uma apresentação genérica do projecto e os seus pormenores.

O capítulo 5 (Desenvolvimento do Protótipo), explica como foi efectuada a implementação do protótipo. São explicados os problemas encontrados, previstos e imprevistos.

O capítulo 6 (Conclusões e Considerações Gerais), analisa os resultados obtidos da prototipagem: discute o sucesso/insucesso do projecto através da análise destes resultados. Foca questões pertinentes tais como: futuros desenvolvimentos e a avaliação geral do estágio. Mais concretamente, são resumidas as vantagens do produto desenvolvido e a utilidade que poderá ter para a instituição de estágio ou para os seus clientes/parceiros. Também é efectuada uma análise do que foi a integração, a formação dada pela instituição, as facilidades e dificuldades sentidas ao longo do estágio.

O Anexo A é o glossário, ideal para uma consulta rápida.

O Anexo B apresenta o plugin Course Readings tal como foi apresentado na conferência do ICAU-Porto.

O Anexo C apresenta o artigo que será apresentado no e-LES'04-Aveiro.

O Anexo D apresenta os *slides* desse mesmo artigo.

2 Análise do problema

2.1 Plugin Course Readings

2.1.1 Contexto

É extremamente útil para a geração automática de *course shells*, sem intervenção directa do docente, mas com base na informação que este introduziu no SiFEUP (*Sistema de Informação da Faculdade de Engenharia do Porto*).

Como a bibliografia de uma disciplina já existe no SiFEUP e encontra-se catalogada na biblioteca digital da FEUP, então bastará a sua integração na LUVIT LMS, evitando assim o custo do docente inserir informação já existente.

Na LUVIT LMS, esta integração fica automaticamente assegurada quando o docente activa o plugin para um determinado curso.

Para especificar este problema, foi concebida uma arquitectura geral que efectua a integração de sistemas externos na LUVIT, através de ferramentas de integração.

O plugin Course Readings enquadra-se nesta arquitectura, no sentido em que:

- O ALEPH é o sistema externo;
- A ferramenta de integração é todo o mecanismo que permite a integração do ALEPH na LUVIT;

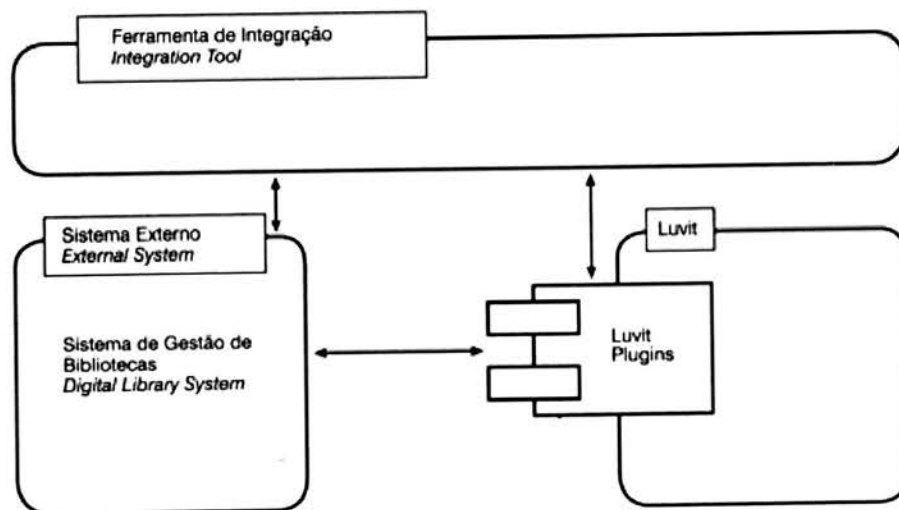


Fig. 1 – Arquitectura Geral de Integração

2.1.2 O Problema

Em termos gerais, os problemas encontrados foram os seguintes:

- O que era um plugin na LUVIT e como instalá-lo;
- Como conectar esse plugin com o ALEPH de modo a retornar a bibliografia;

A solução para estes problemas está descrita na sub-secção 4.2.1.

Para uma melhor análise deste problema, o mesmo foi descrito/especificado utilizando diagramas de caso de uso em UML [4].

Um diagrama de casos de uso de um sistema, mostra **actores** (tipos de utilizadores), **casos de uso** e relações entre eles.

Permite:

- Mostrar para que serve o sistema (a utilidade do sistema), ignorando a forma como está organizado internamente;
- Especificar o contexto do sistema: com quem interage (actores) e com que finalidade (casos de uso);
- Capturar os requisitos funcionais do sistema: casos de uso são funcionalidades do sistema, vistas pelos utilizadores;

Pode referir-se a um sistema de *software*, um sistema de negócio ou organização, um equipamento, uma classe, etc.

Existem quatro tipos de actores neste sistema: o **Utilizador**, o **Administrador do Curso**, o **Administrador do Sistema** e o **Sistema de Gestão de Bibliotecas**.

O **Administrador do Curso** é a pessoa que cria o curso e que poderá ser também um **Utilizador**. Tem permissão para poder configurar o plugin Course Readings.

O **Administrador do Sistema** tem permissão para configurar todo o sistema. É o que configura as preferências do plugin Course Readings.

O **Utilizador** é aquele que utiliza o plugin Course Readings. Tem permissão para navegar, estudar o material publicado, de submeter-se a testes e comunicar com outros participantes.

O actor **Sistema de Gestão de Bibliotecas** representa o ALEPH [15], que retornará a informação necessária para a visualização da bibliografia no LMS.

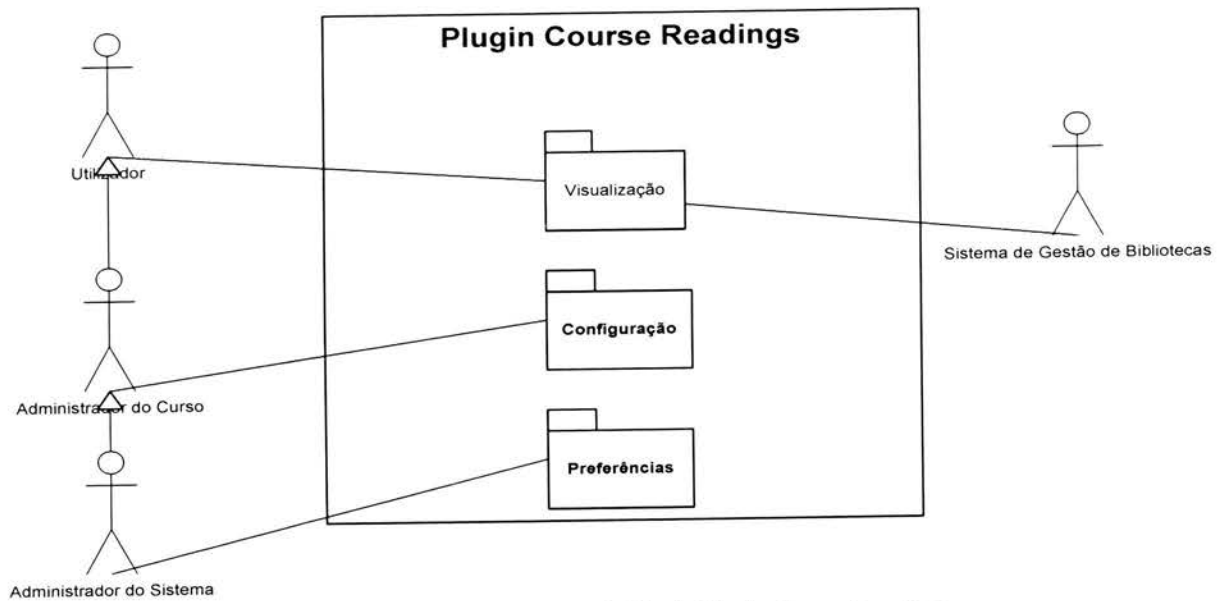


Fig. 2 – Diagrama de Casos de Uso do Plugin Course Readings

No módulo **Visualização**, o utilizador apenas poderá efectuar uma operação: visualizar a bibliografia que é retornada, quando executa o plugin que se encontra na *navigation tree*. A informação que é visualizada depende do resultado da comunicação com o *Sistema de Gestão de Bibliotecas* (neste caso, o ALEPH).

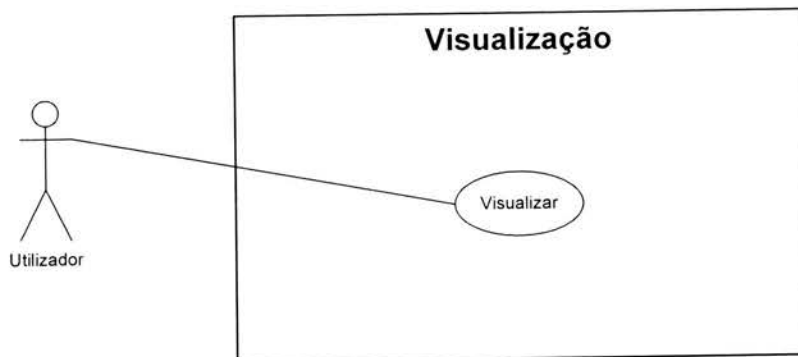


Fig. 3 – Diagrama de Casos de Uso do Módulo Visualização, do Plugin Course Readings

Seguidamente, será apresentado o módulo **Configuração**.

É de realçar, que não foi preciso criar as funcionalidades deste módulo, pois estas já fazem parte da interface do LUVIT LMS.

Criar – Usado para criar um novo documento do tipo Course Readings.

Editar – Edita um documento já criado.

Actualizar – Actualiza um documento depois de este ter sido editado.

Apagar – Apaga o documento associado ao plugin.

Publicar – Publica um documento criado, ou seja, coloca-o visível na *navigation tree*.

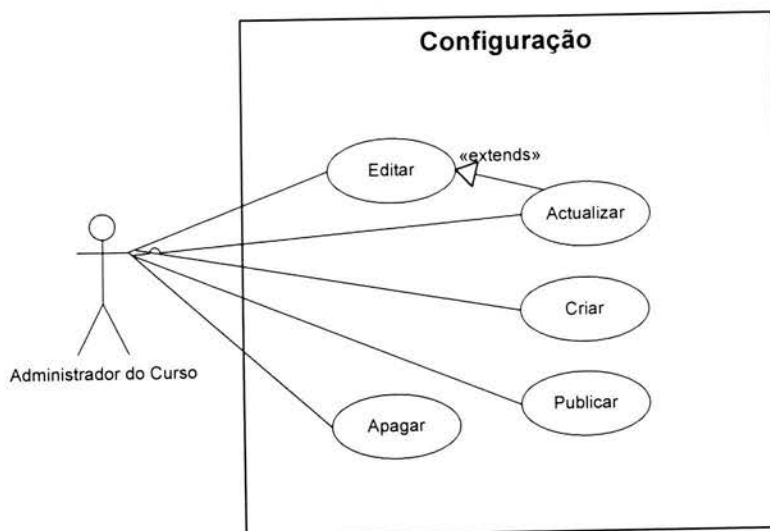


Fig. 4 – Diagrama de Casos de Uso do Módulo Configuração, do Plugin Course Readings

O módulo **Preferências** é relativo às preferências do plugin, mais concretamente:

Adicionar biblioteca digital – Representa um dos atributos de escalabilidade do plugin, no sentido em que permite adicionar uma nova biblioteca digital para retornar bibliografia.

A informação que está associada a cada biblioteca, é a seguinte:

- Nome;
- Servidor;
- Porta;
- Login;
- Password;

Editar biblioteca digital – Permite editar a informação acima descrita.

Apagar biblioteca digital – Permite apagar a informação acima descrita.

Escolher os cursos para o plugin – Permite escolher os cursos que irão utilizar o plugin.

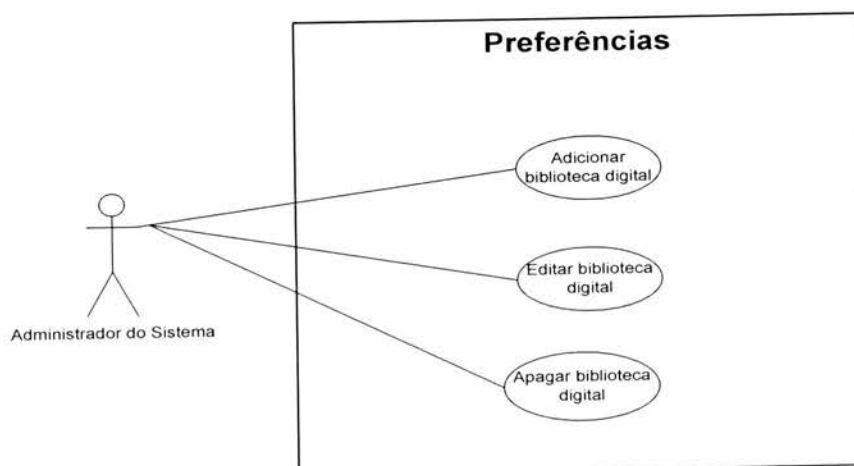


Fig. 5 – Diagrama de Casos de Uso do Módulo Preferências, do Plugin Course Readings

2.2 SADPEL (Sistema de Apoio à Decisão de uma Plataforma de e-Learning)

2.2.1 Contexto Global

Num contexto mais global, o *SADPEL* enquadra-se numa estratégia de e-Learning por parte da LUVIT, mais concretamente o sub-projecto *Semantic Web*, que engloba o serviço de *data analysis* (permite aceder a estatísticas agregadas e a dados *SCORM RTE* também agregados), a integração com o *ALEPH* [15], a produção de objectos *SCORM* com metadados associados e a sua pesquisa no *DigiTool* utilizando *OAI-PMH* [12], conforme especificado no seguinte e-Learning Workflow [16] (Oliveira 2004):

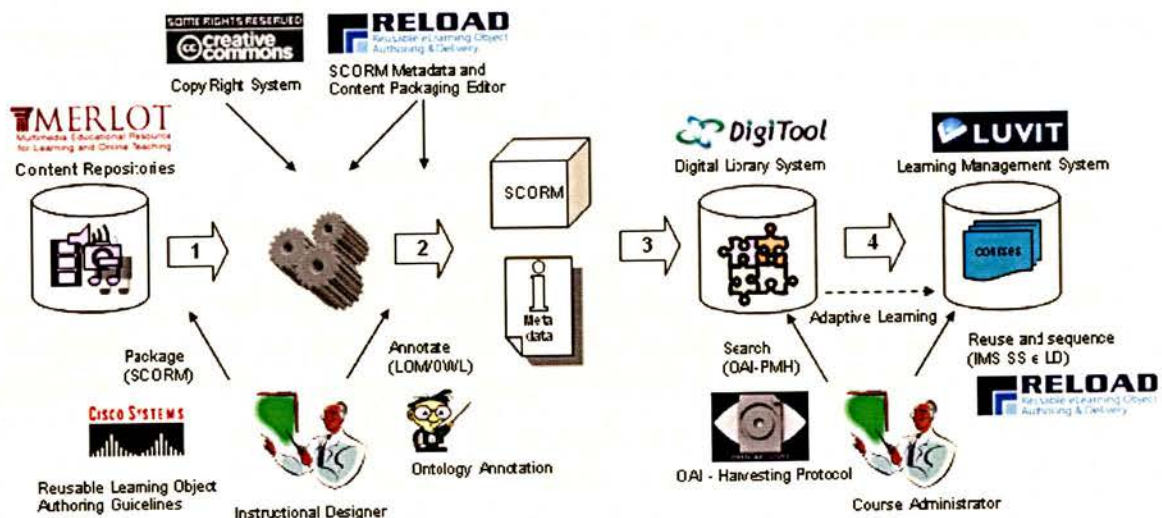


Fig. 6 – e-Learning Workflow

Descrição do Workflow:

1. O conteúdo relevante é encontrado em repositórios e adaptado, empacotado com actividades num objecto SCORM (SCO);
2. Os objectos SCORM são anotados com metadados;
3. Os SCO's são guardados num sistema de biblioteca digital, compatível com o protocolo de pesquisa OAI-PMH [12];
4. Os administradores de curso reutilizam e sequenciam os SCO's em LMS que suportam SCORM [13] ou em procedimentos automatizados baseados em *mining* de metadados que fornecem aprendizagem adaptativa aos utilizadores;

Porquê um SAD (Sistema de Apoio à Decisão)?

Os sistemas de apoio à decisão são programas informáticos de apoio à aprendizagem que apresentam um futuro muito promissor em aplicações na área da educação. Oferecem suporte ao ensino normal, podem facultar a totalidade do ensino à distância, transmitir conhecimento via *Internet*.

São novas formas de educação que podem substituir o formador humano ou acrescentar mais valias ao processo de ensino.

Desde há 25 anos que se utilizam os computadores para apoiarem a educação. Os esforços que foram feitos, não contemplavam a diversidade dos alunos e as suas necessidades específicas, sem se adaptar à sua aquisição de conhecimento, nem dando a atenção individualizada que um professor consegue dar.

Nos últimos 10 anos, a investigação no campo dos sistemas de ensino automático, deu lugar aos *Sistemas Tutores Inteligentes* (ITS, do inglês Intelligent Tutoring System). Existem já alguns exemplos de algumas implementações de ITS, que apresentam várias formas de tratar o problema do ensinar via um computador. A investigação na área dos sistemas peritos, apresenta um caminho para os SAD's de modularidade e separação entre uma plataforma de desenvolvimento e uma base de conhecimento, que começa a ser usado para permitir um mais rápido desenvolvimento de novos tutores.

Com as potencialidades que a Internet apresenta, é o local ideal para disponibilizar os sistemas de apoio e proporcionar a educação à distância por computador.

Alguns SAD's fazem o suporte ao ensino normal, facultando o acompanhamento na resolução de problemas, no entanto, as potencialidades da educação automática estão na disseminação da informação pela Internet, possibilitando ensinarem conceitos totalmente novos.

Como é que os SAD's nos têm ajudado?

O complexo ambiente para uso de recursos e gestão, necessitou do desenvolvimento de novos métodos, ferramentas para considerar nova informação e aplicar novas maneiras de pensar em relação a essa informação. Como consequência, a investigação tem desempenhado um activo contributo crescente na preparação de gestores de informação para se obter processos de decisão de alta qualidade. Esta investigação ao apoio à decisão tem incluído:

- O desenvolvimento de acesso efectivo a uma vasta gama de dados técnicos, conhecimento e informação de processos, que poderão ser relevantes ao processo de decisão;
- O desenvolvimento de novos métodos de análise de potenciais estratégias para o uso de fontes de informação e suas implicações;
- O desenvolvimento de ferramentas ou métodos, que “empacotam” estas novas abordagens para torná-las acessíveis ao gestor da informação;

Como é que procedemos a partir daqui?

Sabendo que o apoio à decisão procura melhorar a qualidade dos processos de decisão, o fornecimento do apoio à decisão precisa de ser pensado mais abertamente que o desenvolvimento de sistemas de apoio à decisão. Aliás, é em parte um facto científico e técnico, mas tendo em conta que pretende mudar os processos de decisão e por conseguinte, as decisões tomadas, é influenciado pelo contexto institucional, social e político. “Se se muda a maneira de tomar decisões, muda-se as decisões que se toma” (atribuído a Jim McNeil por Slater, 1995). “Procurando mudar a maneira que as pessoas tomam decisões acerca do uso e gestão da informação, não é um exercício académico livre de consequências, é uma iniciativa que traz responsabilidades significativas” (Walker 2000).

Isto significa que uma má ferramenta de apoio à decisão, isto é, uma ferramenta que produza informação errada ou inapropriada, é pior do que não ter nenhuma ferramenta. Um aluno que aja baseado em informação errada fornecida pelo SAD, poderá ver o seu processo de aprendizagem penalizado, se, por exemplo, na preparação para um exame, o SAD não mostrar a matéria mais apropriada. É principalmente neste aspecto, que há a necessidade de evoluir sistematicamente potenciais ferramentas de apoio à decisão.

2.2.2 Visão Geral

Seguidamente teremos uma visão geral do problema, que se enquadra no passo 4 do *e-Learning Workflow* da fig. 4.

O SADPEL terá que ser implementado, tendo em conta que o formador (ex: docente) precisa de encontrar o conteúdo apropriado, conteúdo esse, que é procurado na biblioteca digital, para depois o sequenciar e reutilizar nos cursos existentes no LMS da LUVIT.

Este processo é efectuado à medida das necessidades de cada formando (*Just-In-Time Learning*).

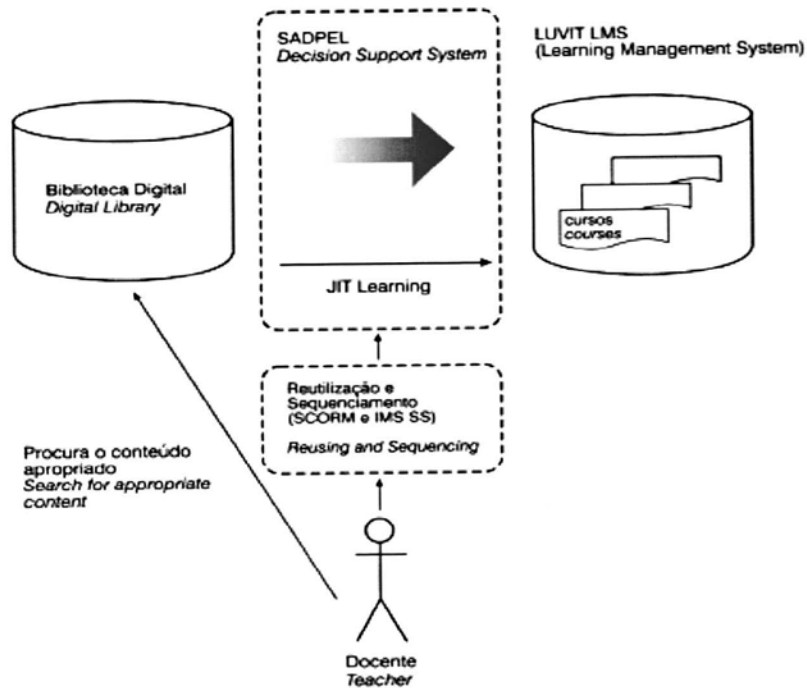


Fig. 7 – SADPEL (Visão Geral)

3 Revisão Tecnológica

3.1 Soluções Existentes ou Semelhantes

3.1.1 Plataformas de e-Learning

Actualmente, existem mais de 20 plataformas de e-Learning que competem directamente com a plataforma LUVIT LMS no mercado; as mais conhecidas são a *Blackboard* e a *WebCT*. Pode-se observar uma comparação entre as diferentes plataformas em Marshall University (2003) [14].

Face às características analisadas, a plataforma LUVIT LMS foi considerada a que reunia as condições mais adequadas para ser adquirida para o *campus* da FEUP, nomeadamente porque:

- É o sistema tecnicamente mais avançado;
- A interface de utilizador é o mais intuitivo e funcional;
- Integra funcionalidades de gestão da componente de formação contínua;
- Facilita a reutilização de conteúdos entre cursos e entre formadores;
- Oferece uma plataforma de desenvolvimento de novas funcionalidades, ao abrigo do acordo de parceria técnica, com a possibilidade de comercialização das extensões desenvolvidas;

A seguinte tabela apresenta uma análise comparativa entre os três sistemas [14].

Pelo facto de haver muitos factores de comparação, apenas foram escolhidos os mais relevantes para a concepção do SADPEL:

Tab. 1 – Comparação entre Sistemas de Formação Online

LMS Factores de Comparação	LUVIT LMS	WebCT 3.6 CE	Blackboard Bb5.5 LS
Origem	Suécia (Universidade de Lund)	Canadá (Universidade Alberta)	EUA (Universidade Cornell)
Sistemas Operativos	Windows (certificado)	Sun, Unix, Windows, Tru64	Sun, Linux, Windows
Arquitectura	Multiservidor, cluster	Multiservidor	Multiservidor
Base de Dados	SQLServer 7	Não suportada	SQL Server 7 e Oracle 8i
Gestão de ficheiros	Suportada. Permite partilha entre cursos	Suportada	Não suportada
Conteúdos IMS	Exchange Server	IMS Content Migration	IMS Import/Export

Integração com o SiFEUP	Possível sob Acordo de Technical Partner	IMS Enterprise API	Building Blocks Advanced Platform Builder (não incluído)
Gestão de Conteúdos de Formação	Assembly Tool Knowledge Pool	E-packs comerciais	Course Cartridges
Gestão da Formação	LRC Manager	Connected Learning Solution (não incluído)	Bb Community Portal (não incluído)
Preço por ano	LUVIT LMS €29.000	WebCT Campus Ed. LII \$22.000 (€24.500)	Bb 5.5 LS \$40.000 (€44.500)

A seguinte tabela apresenta as ferramentas de desenvolvimento suportadas pelas três plataformas, que interessarão ao SADPEL.

Ambas as plataformas suportam *standards* de dados que servirão para interoperarem com outras plataformas.

Os *reports* para a detecção do progresso do aluno, serão muito úteis, pois ajudarão a tomar as decisões sobre o que fazer a cada momento no SAD.

Tab. 2 – Comparação entre Sistemas de Formação Online (Ferramentas de Desenvolvimento)

Ferramentas de Desenvolvimento	Blackboard Bb5.5 LS	WebCT 3.6 CE	LUVIT LMS
A plataforma usa standards de dados abertos para comunicar com aplicações de base de dados universitárias existentes	Sim	Sim	Sim
O fornecedor da plataforma suporta a implementação do standard IMS	Sim	Sim	Sim
O componente de gestão irá criar reports para detecção do progresso do aluno	Sim	Sim	Sim

Para que o SADPEL ajude ao máximo o aluno, este deverá acompanhar o seu próprio progresso e as suas notas.

A seguinte tabela mostra duas ferramentas que a LUVIT LMS possui que ajudarão o aluno nestes aspectos.

Tab. 3 – Comparação entre Sistemas de Formação Online (Ferramentas de Estudante)

Ferramentas de Estudante	Blackboard Bb5.5 LS	WebCT 3.6 CE	LUVIT LMS
Tracking de progresso	Sim	Sim	Sim
O aluno tem acesso às suas próprias notas	Sim	Sim	Sim

Neste aspecto, a *WebCT*, apresenta um factor negativo: apresenta elevado custo de implementação.

Tendo em conta que o SADPEL envolverá muito apoio e implementação administrativa, esta não seria a plataforma ideal para o implementar, isto se fosse preciso escolher uma plataforma para suportar o SADPEL.

Tab. 4 – Comparação entre Sistemas de Formação Online (Características Administrativas)

Características Administrativas	Blackboard Bb5.5 LS	WebCT 3.6 CE	LUVIT LMS
O fornecedor da plataforma irá alojar os cursos no seu servidor	Sim	Não	Sim
A plataforma tem elevado custo de implementação com custos mínimos contínuos	Não	Sim	Não

A escolha da plataforma de e-Learning para este estágio, não dependeu de uma comparação qualitativa entre estas plataformas, uma vez que a empresa de estágio e o local de trabalho, produz e utiliza respectivamente a plataforma LUVIT LMS.

3.1.2 Redes Bayesianas

Foi o modelo escolhido para o projecto SADPEL, por ser o mais promissor, adequado e simples para fazer adaptação ao utilizador, sequenciamento curricular e escolha do conhecimento a apresentar.

A característica que acresce eficiência ao ensino com SAD's é a sua adaptabilidade ao utilizador. Esta operação é feita sobre um domínio de incerteza, pois não se conhece ao início, o que o utilizador sabe, nem quais as suas preferências. A inteligência artificial desde cedo tenta tratar este problema de incerteza e a resolução mais promissora que existe actualmente, e que é motivo de forte investigação, são as *Redes Bayesianas* [3].

As vantagens são:

- Raciocínio dedutivo e abdutivo, sendo um grande potencial para apoiar o ensino à distância;
- A teoria da probabilidade para representar a incerteza, é matematicamente bem definida, bem fundamentada e tem sido extensivamente estudada, daí a credibilidade em utilizá-la para o tratamento da incerteza em sistemas de apoio à decisão;

O seu estudo levanta os seguintes problemas:

- Se as variáveis têm valores ausentes ou não;
- A obtenção de probabilidades feitas a partir de testes ao aluno, é uma tarefa árdua;

A sua utilização levanta os seguintes problemas:

- A ineficiência computacional; a representação explícita de uma distribuição conjunta de probabilidade, requer espaço exponencialmente proporcional ao número de variáveis consideradas na rede bayesiana;

3.1.3 Abordagens de Incerteza

Para além das redes bayesianas, entre as abordagens mais tradicionais existentes para a modelagem e tratamento de incertezas, encontram-se:

- Teoria de Dempster-Shafer;
- Lógica Fuzzy;
- Redes Neurais;

Teoria de Dempster-Shafer

A vantagem da Teoria de Dempster-Shafer sobre as outras abordagens, está na capacidade deste método em modelar o afunilamento do conjunto de hipóteses, à medida em que se acumulam evidências; este procedimento reflecte o processo que caracteriza o raciocínio usado em diagnóstico e o raciocínio especializado em geral. Além disso, permite expressar ignorância parcial e total de forma extremamente adequada, ao contrário da *Teoria de Bayes* que expressa ignorância parcial atribuindo-se crença à negação da hipótese e ignorância total, dividindo-se o total da crença entre as hipóteses presentes (eventualmente, atribuindo mais crença do que realmente possuem).

As desvantagens são:

- Não há ainda procedimentos seguros que guiem o processo de declaração de crenças, o que pode introduzir agravantes ao sistema;
- Como utiliza um grande número de hipóteses (maior que na *Teoria de Bayes*), o cálculo computacional pode ser extremamente penoso. No entanto, por não precisar de probabilidades condicionais e como uma evidência em geral foca apenas um pequeno número do total das hipóteses, o cálculo pode ser bastante simplificado, pois não precisa indicar crença nula. Outro problema ainda em questionamento, é o de como as crenças devem ser propagadas;
- Implementação morosa: muito raciocínio;

Lógica Fuzzy

É a teoria sobre conjuntos lançada pelo Prof. Lofti A. Zadeh, publicada em 1965. Presume que cada elemento de um universo possa possuir pertinência parcial a um determinado conjunto. Essa teoria generaliza os conceitos da Teoria Clássica de Conjuntos. Possui aplicações nas mais variadas áreas, como por exemplo: Avaliações por questionários, Medidas de Incerteza, Inteligência Artificial, Decisão, Processamento de Imagens, Optimização, Análise de Regressão, Reconhecimento de Padrões, etc.

Os sistemas fuzzy são relativamente fáceis de se entender e de se usar na prática, e o conhecimento matemático não é tão complexo como parece à primeira vista. Entretanto, o projectista de controladores Fuzzy necessita de uma profunda compreensão de como as imprecisões e incertezas ocorrem em processos e como elas afectam as aplicações usuais da teoria de controle moderno, possibilitando a aferição da relevância e confiabilidade dos resultados obtidos através da aplicação da teoria da lógica fuzzy.

A característica especial da lógica fuzzy (também referida como lógica nebulosa e em alguns casos por Teoria das Possibilidades) é a de representar uma forma inovadora de manuseio de informações imprecisas, de forma muito distinta da Teoria das Probabilidades. A lógica fuzzy possui um método interessante de compreensão e tradução de expressões verbais, acções quotidianas de funcionamento racional, vagas, imprecisas e qualitativas, típicas na comunicação humana em valores numéricos.

A Lógica Fuzzy tem se mostrado mais adequada para tratar imperfeições da informação, do que a teoria das probabilidades.

Redes Neurais

As redes neurais artificiais, consistem num método de solucionar problemas de inteligência artificial, construindo um sistema que tenha circuitos que simulem o cérebro humano, inclusive o seu comportamento, ou seja, aprendendo, errando e fazendo descobertas.

As redes neurais possuem certas características exclusivas de sistemas biológicos. Tais características entram em conflito com os tradicionais métodos computacionais. Sistemas de computação baseados em redes neurais têm a capacidade de receber ao mesmo tempo várias entradas e distribuí-las de maneira organizada.

Num sistema de rede neuronal, a informação pode parecer ter representação redundante, porém, o facto de que ela se encontra distribuída por todos os elementos da rede significa que mesmo que parte da rede seja destruída, a informação contida nesta parte ainda estará presente na rede e poderá ser recuperada. Portanto, a redundância na representação de informações numa rede neuronal, transforma-se numa vantagem que torna o sistema tolerante a falhas.

As vantagens são:

- O modelo de rede neuronal tem muitos neurónios conectados por pesos com capacidade de adaptação que podem ser arrançados numa estrutura paralela. Por causa deste paralelismo, a falha de alguns neurónios não causam efeitos significativos para o desempenho de todo o sistema – tolerância a falhas;
- Capacidade de adaptação e aprendizagem: significa que podem lidar com dados imprecisos e situações não totalmente definidas;

As desvantagens são:

- Pouca capacidade de extrapolação do modelo;
- Custo computacional elevado;
- O tamanho da amostra tem que ser grande;

3.1.4 A utilização de normas (standards) de e-Learning

A normalização consiste num acordo sobre um conjunto específico de características que um produto ou processo deverão exhibir, ou seja, estabelece os critérios que pautarão a actividade de todas as pessoas envolvidas na indústria do e-Learning.

No documento elaborado por Singh (2000), denominado *White Paper Demystifying eLearning Standards*, o autor define que as iniciativas de normalização focalizam-se basicamente em três aspectos:

- **Portabilidade do conteúdo** – quando o conteúdo é separado do sistema de fornecimento proprietário, a organização pode consolidar, organizar e conceber as suas iniciativas de e-Learning no LMS da sua escolha. Isto possibilita a customização dos conteúdos, logo há maior flexibilidade e menores custos operacionais para as empresas;
- **Granularidade** – as novas especificações suportam a metodologia do objecto de aprendizagem permitindo a inclusão de unidades de informação mais pequenas e actualizadas. Os objectos de aprendizagem adicionam o *just-enough* à aprendizagem *Just-In-Time*;
- **Interoperabilidade** – a aplicação de interoperabilidade começa quando diferentes aplicações de e-Learning conseguem partilhar conteúdo e localizar dados. Estas especificações abrem a possibilidade de trocar e aceder a conteúdos;

O *e-Learning Consortium S3 Working Group do Masie Center*, alarga em mais quatro aspectos (para além da interoperabilidade) o campo de acção das iniciativas de normalização. No documento *Making Sense of Learning Specifications & Standards: A Decision Maker's Guide to their Adoption* (2002), pode ler-se o seguinte:

- **Reutilização** – flexibilidade na incorporação de componentes de instrução em múltiplas aplicações e contextos, isto é, consiste em possibilitar a reutilização do curso ou objecto de aprendizagem;
- **Capacidade de Gestão** – possibilidade dos sistema de gestão da formação (LMS), localizar a informação apropriada quer sobre o conteúdo, quer sobre o formando;
- **Acessibilidade** – possibilidade do formando aceder ao conteúdo adequado e na altura apropriada;
- **Durabilidade** – consiste em permitir as constantes mudanças e evoluções tecnológicas sem ser necessário redesenhar, reconfigurar e reprogramar;

Nos últimos anos, são várias as entidades que se têm debruçado sobre esta temática, sendo que no momento, os principais normativos da indústria do e-Learning provêm dos EUA. Dentro das várias organizações que estão a desenvolver iniciativas de normalização, as três maiores segundo Singh (2000), são a **Aviation Industry CBT** (Computer-Based Training) **Committee** (AICC), a **ADLNet** – Advanced Distributed Learning (ADL) Sharable Content Object Reference Model (SCORM) [13]) e o **Institute of Electrical and Electronics Engineers** (IEEE).

3.1.5 SCORM (Sharable Content Object Reference Model)

Nesta secção, serão explicados detalhadamente o SCORM e as razões da sua utilização, para melhor se entender a sua aplicação no SADPEL.

Na sub-secção 4.2.2.2, será então enquadrado o SCORM no contexto do SADPEL.

O SCORM [13] é um conjunto de especificações e linhas de orientação que estabelecem os níveis de acessibilidade, interoperabilidade, durabilidade e reutilização dos conteúdos e sistemas de aprendizagem baseados em *Web*. Até ao momento o SCORM já conheceu quatro versões.

O SCORM não é uma norma em si mesma, mas sim um modelo de referência que serve a efectividade e a aplicação real de um conjunto de normas e especificações num modelo coeso, utilizável e holístico, e assim define interrelacionamentos chave entre normas. Por isso, o SCORM é na sua essência um modelo de facto, porque não foi mapeado como um corpo de aprovação de normas, mas sim como um modelo que os governos e a indústria de aprendizagem no seu todo adoptam voluntariamente.

Para este trabalho, foi optado pelo modelo de normas SCORM [13], por várias razões:

- É o modelo utilizado pela LUVIT, mais concretamente o SCORM 1.2;
- Possui um maior grau de desenvolvimento e maturação;
- Aparentemente é o modelo de referência mais generalizado no mercado internacional, já que integra diferentes corpos de normas, em diversos âmbitos e de diferentes organizações. Além disso, também possui uma das maiores redes de laboratório especializadas;

O SCORM baseia-se em três pilares fundamentais:

- Representação de estruturas de cursos baseada em XML, para que os cursos possam transitar entre LMS's;
- Conjunto de especificações relacionadas com o ambiente em que os conteúdos vão ser executados (run-time environment), incluindo uma *API-Application Programming Interface*, um modelo de dados para os conteúdos comunicarem com o LMS e uma especificação para a forma como estes vão ser publicados e executados online;
- Especificação para criar registos na forma de metadados para os cursos e os seus respectivos conteúdos, desde a descrição dos seus recursos físicos em termos de objectos multimédia até à descrição e catalogação de objectos mais estruturados na forma de objectos de aprendizagem;

Assim, e de forma resumida, pode-se dizer que o SCORM define um modelo de agregação de conteúdos e o ambiente onde esses conteúdos vão ser executados.

3.1.5.1 Modelo de Agregação de Conteúdos (Content Aggregation Model)

No SCORM, o *Modelo de Agregação de Conteúdos* (CAM - Content Aggregation Model), representa uma forma pedagogicamente neutra de agregar recursos de aprendizagem com o objectivo de fornecer uma determinada experiência de aprendizagem. Um recurso de aprendizagem é qualquer representação de informação, utilizado numa experiência de aprendizagem. Experiências de aprendizagem consistem em actividades que são suportadas por recursos electrónicos ou não electrónicos.

O CAM suporta esta actividade através do seguinte:

- **Modelo de Conteúdos:** nomenclatura para definir as componentes do conteúdo de uma experiência de aprendizagem;
- **Metadados:** mecanismos para descrever instâncias específicas dos componentes do modelo de conteúdos;
- **Empacotamento dos conteúdos:** define como representar o comportamento de uma experiência de aprendizagem (estrutura dos conteúdos) e como empacotar os recursos de aprendizagem de forma a poderem ser instalados em diferentes ambientes;

Componentes do Modelo de Agregação de Conteúdos

O CAM é composto pelos seguintes elementos:

- **Recursos (Assets)**
- **SCO (Sharable Content Objects)**
- **Agregados de Conteúdos (Content Aggregations)**

Recursos (Assets)

É a forma mais básica de conteúdos de aprendizagem e são representações electrónicas de media (texto, imagens, animações Flash, som, páginas *Web*, questões para avaliação, etc.). Um recurso pode ser descrito através de metadados de forma a permitir a sua pesquisa em repositórios de conteúdos para reutilização.

SCO (Sharable Content Objects)

Representa uma colecção de um ou mais recursos que inclui um recurso específico de execução, utilizando o Run-Time Environment do SCORM para comunicar com o LMS. Um SCO representa o mais granular nível de recursos de aprendizagem que podem ser controlados pelo LMS.

Para ser reutilizado, o próprio SCO deve ser independente do contexto de aprendizagem. Por exemplo, pode ser reutilizado, individualmente ou de forma agregada, em diferentes experiências de aprendizagem de forma a cumprir diferentes objectivos de aprendizagem.

Um SCO só pode ser executado por um LMS. Um SCO não pode, por si só, executar outro SCO. Os benefícios destes requisitos são os seguintes:

- Qualquer LMS que suporta Run-Time Environment SCORM, pode executar SCO's e controlá-los (tracking), independentemente de quem os tenha gerado;
- Qualquer LMS que suporta Run-Time Environment SCORM, pode controlar qualquer SCO, sabendo quando é que foi iniciado e quando é que foi terminado;

Agregados de Conteúdos (Content Aggregations)

Um *agregado de conteúdos* é um mapa (estrutura de conteúdos) que pode ser utilizado para agregar recursos de aprendizagem numa unidade coesa de instrução, aplicando estruturas e associando taxionomias de aprendizagem.

Um agregado de conteúdos pode ser descrito através de metadados de forma a permitir a sua pesquisa em repositórios para reutilização.

Exemplo de uma organização taxionómica em termos curriculares:

Curso

Módulo

Sessão

Unidade

O mecanismo de ligação dos agregados de conteúdos à sua representação em metadados, é precisamente o empacotamento de conteúdos, como será explicado posteriormente.

Na versão do SCORM utilizada pela LUVIT (1.2), a navegação e sequência entre os recursos de aprendizagem, são definidas na estrutura de conteúdos utilizando pré-requisitos para cada recurso de aprendizagem ou SCO.

O LMS é responsável por interpretar a sequência pretendida descrita na estrutura de conteúdos e controlar a mesma em tempo de execução.

Este conceito é importante pois a reutilização dos recursos de aprendizagem não pode ser realizada se os mesmos contiverem informação embebida que é específica de um determinado contexto; ou seja, se um determinado SCO contiver ligações específicas a outro SCO, não pode ser utilizado num curso diferente onde o segundo SCO pode não ser aplicável ou estar disponível.

A reutilização de um determinado recurso de aprendizagem (SCO) depende da sua independência em termos de não estar ligado a um agregado específico.

O SCORM reconhece, no entanto, que determinados recursos de aprendizagem possam conter lógica interna para a realização de uma determinada tarefa (relacionada com a interactividade com o utilizador) e que normalmente não é visível para o LMS. Os botões de *Play*, *Stop*, *Rewind* e *Pause* para controlar um determinado *clip* de vídeo dentro de um SCO, podem ser considerados como um exemplo desta lógica interna.

Esta é uma área importante que os implementadores devem ter em atenção de forma a determinar quais os recursos de aprendizagem a utilizar e como irão ser agregados.

3.1.5.2 Empacotamento de Conteúdos (Content Packaging)

O objectivo do empacotamento de conteúdos é o de possibilitar uma forma normalizada de trocar recursos de aprendizagem entre diversos sistemas (LMS) e ferramentas (Sistemas de Autor).

O *empacotamento de conteúdos* define também a estrutura (ou organização) e o comportamento pretendido de um determinado conjunto de recursos de aprendizagem.

O empacotamento de conteúdos define, entre outras coisas:

- Um ficheiro de manifesto descrevendo o próprio empacotamento e que contém:
 - Metadados acerca do pacote;
 - Uma secção *Organização* opcional que define a estrutura dos conteúdos e o seu comportamento;
 - Uma lista de referências para os recursos do pacote;
- Como se deve criar um manifesto baseado em XML;

Directivas para empacotar o manifesto e todos os ficheiros relacionados num ficheiro *zip*, num CD-ROM, etc.

A parte relacionada com o Empacotamento de Conteúdos no SCORM está dividida em áreas:

- Estrutura do conteúdo;
- Descrição de um pacote IMS;
- Modelo de informação para Empacotamento de Conteúdos;
- Ligação XML ao Empacotamento de Conteúdos;
- Perfis de aplicações para Empacotamento de Conteúdos;

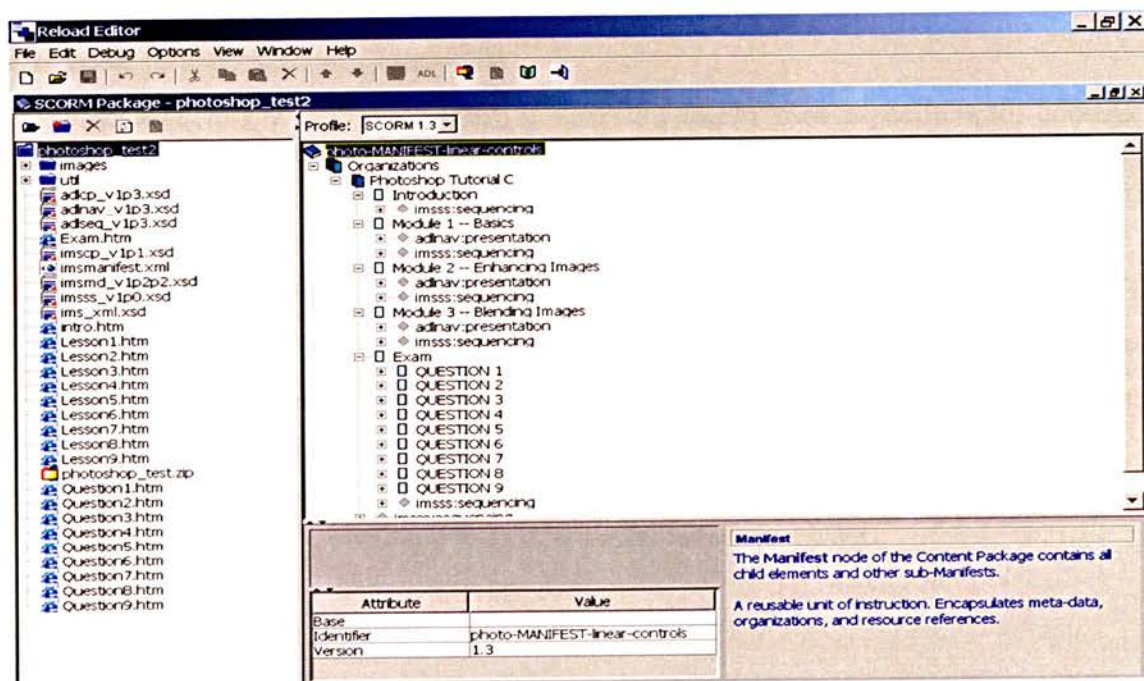


Fig. 8 – Empacotamento de Conteúdos (Content Packaging) em SCORM

Estrutura de Conteúdos

A estrutura do conteúdo pode ser considerada como o mapa de sequência/navegação através dos recursos de aprendizagem definidos no pacote de conteúdos. Deve conter não só a estrutura dos recursos de aprendizagem, mas também os comportamentos que devem ser aplicados à experiência de aprendizagem.

A estrutura do conteúdo é definida em várias partes, cada uma definindo aspectos específicos de uma colecção de recursos de aprendizagem:

- **Hierarquia dos conteúdos:** define uma representação em árvore, semelhante a um índice, que agrupa os recursos de aprendizagem numa dada ordem lógica. Representa, na maioria dos casos, a ordem pela qual o autor tenciona que o utilizador progrida no curso;
- **Sequência e Navegação:** esta parte, fornece ao LMS a informação que este necessita para saber que recursos de aprendizagem é que devem ser apresentados ao utilizador e quando. Fornece também informação de como é que as possibilidades de navegação são apresentadas ao utilizador;

3.1.5.3 Run-Time Environment

É o ambiente de execução do SCORM.

O objectivo do SCORM é permitir que os recursos de aprendizagem possam demonstrar interoperabilidade entre múltiplos LMS's. Para que tal seja possível, deverá existir uma forma comum para iniciar esses recursos (*Launch*), um mecanismo comum para que os recursos comuniquem com um LMS (*API*) e uma linguagem predefinida para o vocabulário básico dessa comunicação (*Modelo de Dados*).

3.2 Outras Tecnologias

Seguidamente, serão apresentadas outras tecnologias e a justificação do seu uso, que tiveram potencial envolvimento no desenvolvimento do projecto.

3.2.1 UML (Unified Modeling Language)

A *Unified Modeling Language (UML)*, é uma linguagem para especificação, construção, visualização e documentação de artefactos de um sistema de *software*. É uma linguagem gráfica, cujo objectivo principal é o de promover e facilitar a comunicação entre um grupo variado de intervenientes.

A UML providencia as seguintes particularidades principais:

- Semântica e notação para tratar um grande número de tópicos actuais de modelação;
- Semântica para tratar tópicos de modelação futura, relacionados em particular com a tecnologia de componentes, computação distribuída, *frameworks* e *Internet*;
- Mecanismo de extensibilidade, de modo que futuras aproximações e notações de modelação possam continuar a ser desenvolvidas sobre a UML;

A UML alarga o âmbito de aplicações alvo, comparativamente com outros métodos existentes, designadamente porque permite, por exemplo, a modelação de sistemas concorrentes, distribuídos para a *Web*, etc. A sua ênfase, é na definição de uma linguagem de modelação *standard* e por conseguinte, é independente das linguagens de programação, das ferramentas *CASE*, bem como dos processos de desenvolvimento. A razão referida é que dependendo do tipo de projecto, da ferramenta de suporte, ou da organização envolvida, devem ser adoptados distintos processos/metodologias, mantendo-se contudo a utilização da mesma linguagem de modelação.

Por todas estas razões, a UML foi escolhida para especificar, construir, visualizar e documentar (através dos diagramas que são aqui apresentados) o sistema de *software* deste projecto.

3.2.2 XML

Extended Markup Language (XML) é a linguagem que permite definir a organização de estruturas de informação para a publicação e intercâmbio de texto ou outro tipo de conteúdos.

Por exemplo, é possível descrever, em XML, um documento que depois irá ser convertido para inúmeros formatos com o auxílio de outras ferramentas.

```
<documento>
```

```
<titulo>Documento ABC</titulo>
```

```
<data>12-3-2004</data>
```

```
<autor>Desconhecido</autor>
```

```
</documento>
```

Qualquer definição de estrutura de informação realizada em XML pode ser validada, contrariamente ao que acontece com outro tipo de linguagens deste género.

O XML apresenta-se como formato ideal para o armazenamento de informação estruturada.

Foi utilizado na implementação do plugin Course Readings, para guardar a bibliografia recebida do ALEPH, para depois se poder fazer a sua transformação na informação pretendida.

Será utilizado também no SADPEL com vista à ligação aos metadados (*XML binding*) do SCORM. Esta ligação, é baseada nas especificações de ligação XML do IMS. A ligação XML define como é que os itens do modelo de informação de metadados são interpretados e transformados em XML.

A ligação descreve como é que o modelo de informação é representado em XML, indicando quais os itens que são elementos XML e os que são atributos XML, os nomes dos elementos e atributos, etc.

3.2.3 C# ASP .NET

O uso desta tecnologia deveu-se muito ao facto de estar planeada para ser utilizada futuramente pela LUVIT, em substituição das páginas ASP tradicionais.

São inúmeras as suas vantagens:

Estado e Aplicação

O ASP.NET fornece uma *framework* simples que permite aos programadores escrever a lógica de negócio que “corre” ao nível da aplicação. Os programadores podem escrever código no ficheiro de texto *global.asax* ou numa classe compilada.

Acesso a dados

O acesso a bases de dados a partir de aplicações ASP.NET, é uma técnica comum para mostrar informação dinâmica aos visitantes de um *website*. O ASP.NET, torna este processo mais fácil que nunca de aceder a bases de dados para este objectivo e fornece mecanismos de gestão dos dados da base de dados.

Desempenho

Uma grande diferença entre ASP 3.0 e ASP.NET, é a maneira como o código é executado no servidor. Com o ASP.NET, o código é compilado em classes executáveis. Com o ASP 3.0, o código precisa muitas vezes de ser interpretado, a não ser que seja mantido em *cache*. Se quisermos evitar a interpretação do código no ASP 3.0, teremos que o colocar num componente COM.

Potente

Agora com o ASP.NET, é possível aceder ao *Common Language Runtime (CLR)*. Executar no CLR, dá acesso a muitas funcionalidades disponíveis na *framework* do .NET, tais como *debugging*, segurança, acesso a dados, interoperabilidade da linguagem, etc.

Configuração

As configurações são guardadas em ficheiros XML. Cada aplicação pode ter um ficheiro distinto.

Segurança

A *framework* do .NET e do ASP.NET, fornece por omissão esquemas de autenticação e autorização para aplicações *Web*.

Migração de ASP para ASP.NET

Páginas ASP simples podem ser facilmente migradas para aplicações ASP .NET.

O ASP .NET oferece uma sintaxe completa e compatível com aplicações ASP.

4 Descrição da solução proposta/Especificação

4.1 Especificação da Arquitectura

Para a especificação da Arquitectura foram utilizados diagramas estáticos e dinâmicos existentes na linguagem de modelação UML.

Em primeiro lugar é apresentado a Arquitectura Lógica, onde o sistema é decomposto em módulos lógicos distribuídos por camadas, sendo depois precedida pela Arquitectura Física.

A arquitectura física para o trabalho, assenta essencialmente na organização do *software* em componentes de *run-time* (módulos físicos) e na topologia do *hardware* (nós e conexões) sobre a qual esses componentes são distribuídos. Estas duas ideias estão interligadas entre si, uma vez que os componentes de *run-time* estão distribuídos pelos componentes de *hardware*.

Um módulo físico, pode ser uma “biblioteca”, tabela, executável ou outro qualquer componente *run-time*.

Um componente representa normalmente um “pacote” físico de elementos lógicos tais como classes, interfaces, entre outros.

Os diagramas de distribuição e de componentes são diagramas de implementação que descrevem como a estrutura está implementada em *software* e *hardware*.

Os diagramas de distribuição [4] mostram a distribuição de *hardware* do sistema, identificando os servidores como nós do diagrama e a rede que relaciona os nós. Os componentes de *software* vão estar mapeados nestes nós.

Os diagramas de componentes [4] mostram os elementos reutilizáveis de *software* e a sua interdependência. Um componente é formado por um conjunto de classes que se encontram implementadas nele. Um componente, assim como as classes que ele possui, depende funcionalmente das classes de outro componente. O diagrama de componentes mostra esta dependência e é ainda possível mostrar a configuração do sistema de *software*, mostrando graficamente a dependência entre os diversos módulos que compõem o sistema.

4.1.1 Instalação

Explica como foi efectuada a instalação de componentes no *LEC (LUVIT Education Centre)*.

Foi utilizado o Microsoft Visual Studio .NET 2003 como ambiente de programação.

Para a instalação do plugin Course Readings, foi utilizado o *SATO (Site Administration Tool)* da LUVIT, que é uma ferramenta de administração para gerir a instalação de plugins e que está instalado no LEC (máquina *luvit1.fe.up.pt*).

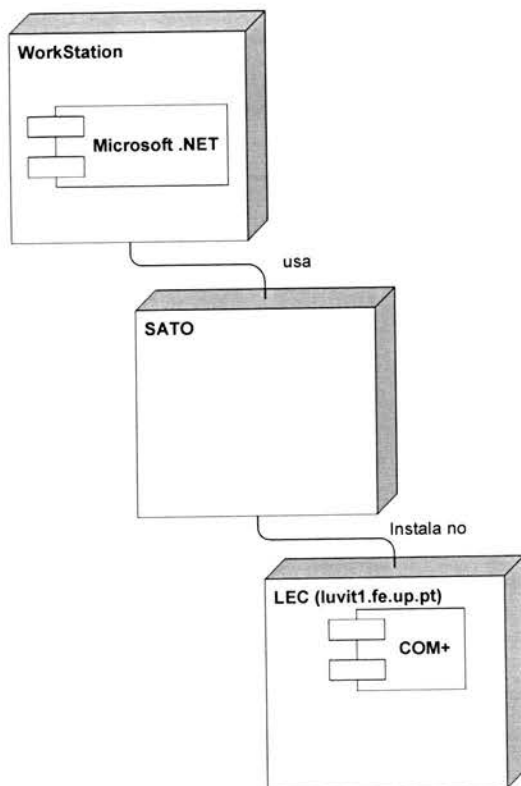


Fig. 9 – Diagrama de Distribuição da Instalação de plugins no LEC

4.1.1.1 Interoperabilidade .NET – COM

Esta secção descreve como os componentes .NET podem comunicar com componentes COM já existentes, sem necessidade de migrar esses componentes COM em componentes .NET, evitando assim os custos associados à migração.

Este processo foi necessário, pois foi preciso comunicar com os componentes usados pela LUVIT (COM), para se proceder à instalação.

Para comunicar com um componente COM, é preciso haver o *Runtime Callable Wrapper (RCW)*. O RCW, pode ser gerado usando o C# .NET ou a ferramenta *TlbImp.exe*.

Ambas as maneiras, irão ler o *type library* e usar a classe *System.Runtime.InteropServices.TypeLibConverter* para gerar o RCW. Esta classe lê o *type library* e converte estas descrições num *wrapper (RCW)*. Depois de gerar o RCW, o cliente .NET deverá importar o seu *namespace*. Depois deste passo, a aplicação cliente já poderá chamar o objecto RCW como se fosse uma chamada nativa.

Quando um cliente chama uma função, a chamada é transferida para o RCW. O RCW internamente chama a função COM nativa *coCreateInstance* criando o objecto COM que é envolvido por este RCW. O RCW converte cada chamada para a convenção COM. Uma vez que o objecto tenha sido criado com sucesso, a aplicação cliente .NET poderá aceder aos objectos COM como se fossem chamadas de objectos nativas.

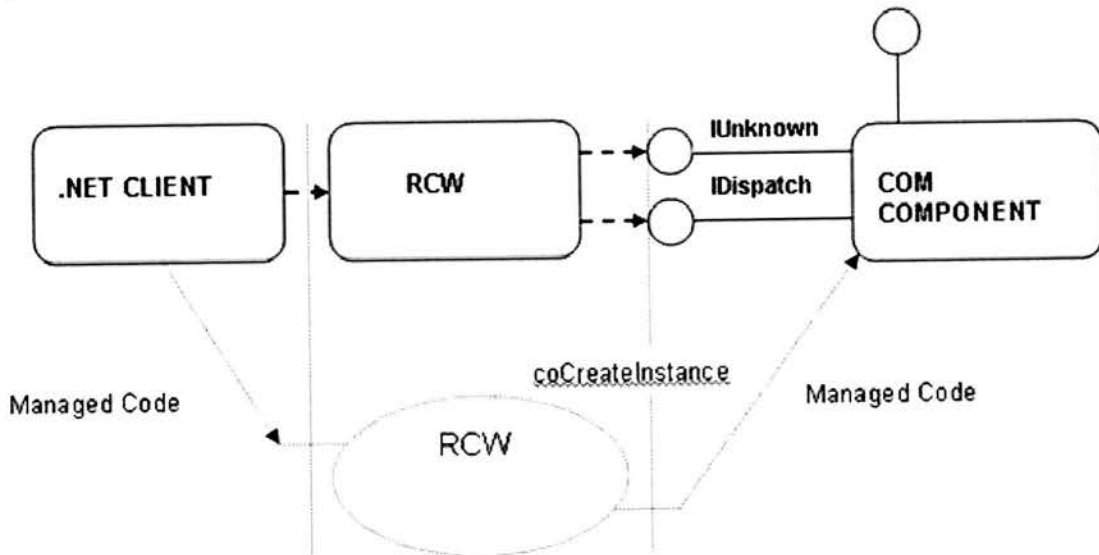


Fig. 10 – Invocação de um componente COM de um cliente .NET

4.1.2 Plugin Course Readings

4.1.2.1 Arquitectura Física

O plugin utiliza um interface *Web* (implementado em C# ASP .NET), que poderá ser acedido através de um browser *Web*.

Todos os pedidos efectuados ao ALEPH (localizado na máquina *athena.fe.up.pt*), são efectuados através do protocolo HTTP.

O ALEPH, depois de consultar a base de dados, retorna a informação pretendida em XML, através do servidor XML existente na mesma máquina.

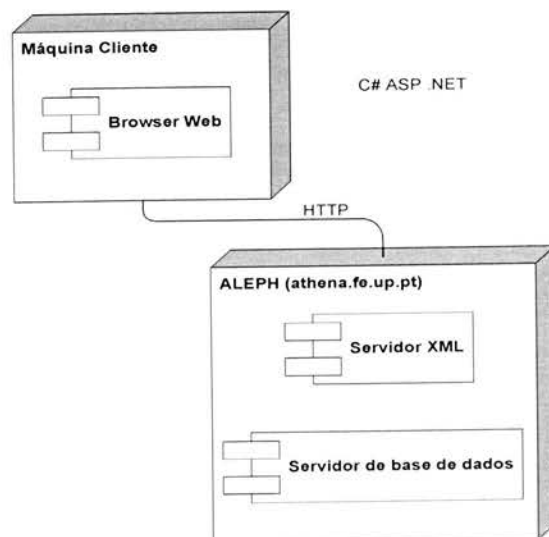


Fig. 11 – Diagrama de Distribuição do Plugin Course Readings

Seguidamente teremos o diagrama de componentes do plugin.

O plugin invoca o ALEPH, que por sua vez retorna um objecto XML. Esse objecto é guardado num ficheiro: *bibliography.xml*.

A classe *XMLParsing.cs* do plugin, usa este ficheiro para fazer o *parsing* e guardar o conteúdo pretendido.

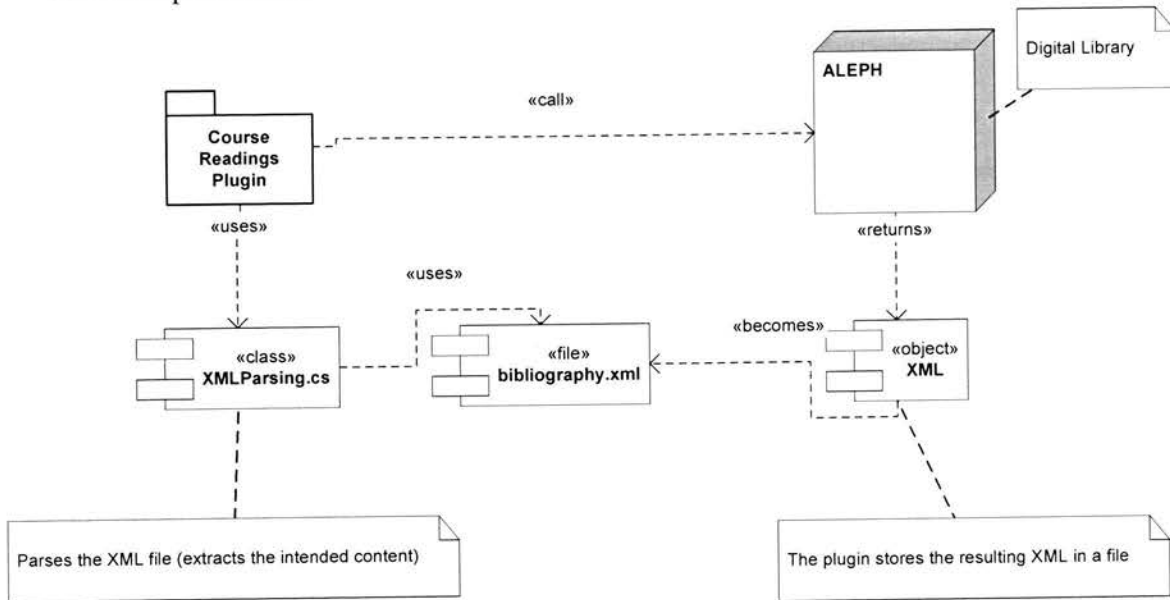


Fig. 12 – Diagrama de Componentes da Arquitectura Física do Plugin Course Readings

Numa primeira fase, utilizou-se um ficheiro para guardar a informação retornada (*bibliography.xml*). Esta solução, não seria a mais apropriada se houvesse muitos utilizadores a quererem utilizar o plugin simultaneamente, pois isso implicaria sincronização.

Foi encontrada uma solução mas simples e mais segura, que foi utilizar imediatamente a informação retornada, sem recurso a um ficheiro. Neste caso, a informação que é retornada é guardada num objecto em memória principal, do tipo *DataSet*, ficando imediatamente disponível para se efectuar o *parsing*, através da classe *StringReader* já existente no .NET.

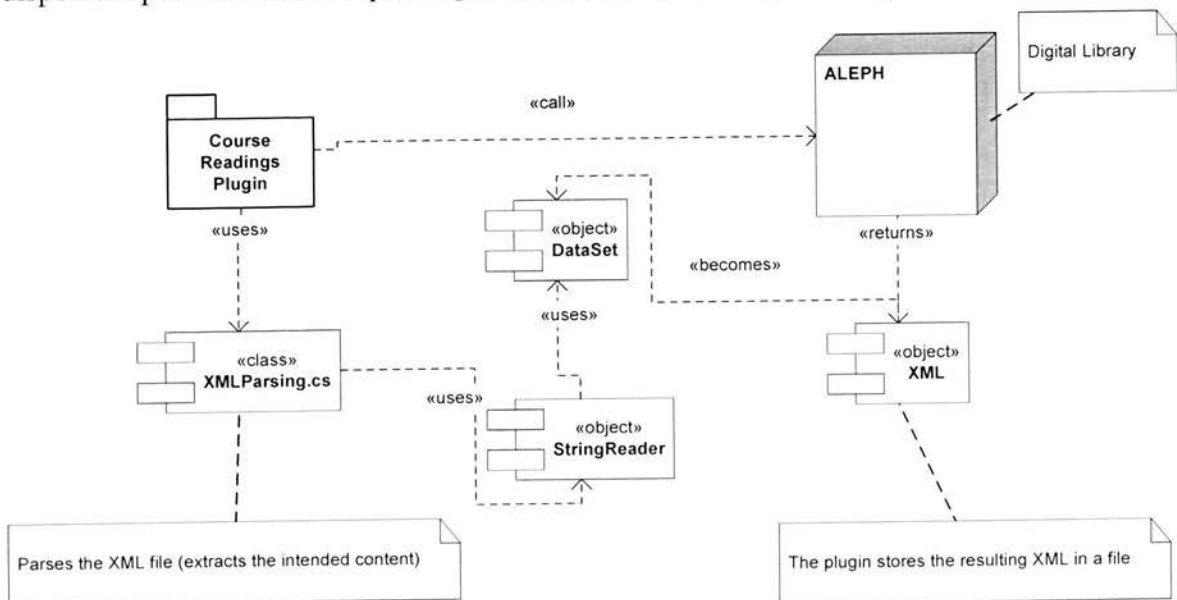


Fig. 13 – Diagrama de Componentes da Arquitectura Física do Plugin Course Readings (modificado)

4.1.3 SADPEL (Sistema de Apoio à Decisão de uma Plataforma de e-Learning)

A arquitectura do plugin Course Readings é flexível no sentido em que permite a reutilização de componentes no projecto SADPEL.

Enquanto o Course Readings suporta chamadas sobre sistemas de gestão de bibliotecas, o Content Readings irá suportar chamadas sobre um *datawarehouse* [18] para filtrar informação sobre o perfil de um utilizador, informação essa que constitui a base na selecção/escolha dos conteúdos que serão apresentados a esse mesmo utilizador.

O *datawarehouse* [18] está em fase de implementação na LUVIT e será um complemento importante para o SAD, pois:

- Contém estatísticas agregadas que permitirão ao SADPEL efectuar análises comparativas, avaliações de qualidade da formação e apoiar decisões estratégicas;

4.1.3.1 Arquitectura Lógica

A Arquitectura Lógica, aqui representada na figura, é composta por três camadas:

- Interface;
- Lógica de Negócio;
- Base de Dados;

Cada uma destas camadas, usa funcionalidades das camadas inferiores.

Interface – A camada de Interface, consiste essencialmente em páginas ASP .NET/HTML instaladas no LEC (Learning Educational Centre) da LUVIT. A *navigation tree* será um dos componentes mais utilizados para fazer a interacção entre o utilizador e o sistema.

Lógica de Negócio – Por baixo da camada Interface, tem a camada Lógica de Negócio, que é responsável pela implementação de todas as funções de negócio que são necessárias para a modelação e bom funcionamento do sistema desenvolvido. É uma camada intermédia que faz a ligação entre as camadas Base de Dados e Interface; procura os dados específicos na Base de Dados, para a Interface mostrar a informação pretendida.

Base de Dados – o SADPEL irá utilizar a bases de dado da LUVIT e do ALEPH). A LUVIT utiliza uma base de dados em SQL Server 7.



Fig. 14 – Arquitectura Lógica do SADPEL

4.1.3.2 Arquitectura Física

O utilizador poderá aceder ao plugin Content Readings por um interface *Web* (também implementado em C# ASP .NET), através de um browser *Web*.

O plugin Content Readings utilizará as funcionalidades do SADPEL através de chamadas sobre um *datawarehouse* [18] para filtrar informação sobre o perfil de um utilizador.

Os pedidos serão executados em HTTP.

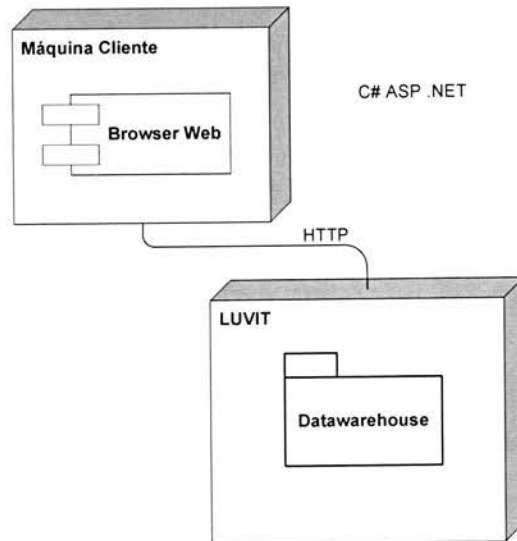


Fig. 15 – Diagrama de Distribuição da Arquitectura Física do SADPEL

4.2 Solução Proposta

4.2.1 Plugin Course Readings

O plugin consiste em componentes do tipo *Web Form* que interagem entre si.

O plugin foi dividido nas seguintes fases:

1. **Interrogação ao ALEPH;**
2. **Retorno do XML;**
3. **Parsing do XML;**
4. **Visualização da Bibliografia Recomendada;**

Cada uma das fases será explicada mais detalhadamente:

1. **Interrogação ao ALEPH** – é utilizada para fazer dois pedidos via HTTP ao ALEPH, através da sua especificação dos formatos de interrogação (*alephx*).

O primeiro pedido, retorna informação necessária em XML, para se efectuar o pedido referente à bibliografia recomendada de uma determinada disciplina. Essa informação, é a seguinte: o **índice de pesquisa** e o **número de registos de uma disciplina**. O pedido, tem o seguinte formato:

[http://athena.fe.up.pt/X?op=find&base=eup01&request="codigo da disciplina"](http://athena.fe.up.pt/X?op=find&base=eup01&request=)

X?: chamada ao serviço de XML;

op=find: operação de pesquisa para retornar o índice de pesquisa e o número de registos de uma disciplina;

base: a biblioteca base onde a pesquisa é efectuada;

request: assunto de pesquisa (neste caso, interessa-nos o código da disciplina);

O segundo pedido, interroga o ALEPH com o índice de pesquisa, retirado do XML recebido do primeiro pedido. Tem o seguinte formato:

[http://athena.fe.up.pt/X?op=present&set_no="+cadeira_id+"&set_entry=000000001&format=marc](http://athena.fe.up.pt/X?op=present&set_no=)

op=present: operação de pesquisa ao índice da disciplina;

set_no: índice de pesquisa da disciplina;

set_entry: intervalo de registos que queremos ver retornados;

format: formato da bibliografia;

2. **Retorno do XML** – o XML que é recebido como resultado da interrogação ao ALEPH, é guardado num ficheiro, para depois ser mais fácil fazer o seu *parsing*.

3. **Parsing do XML** – esta fase é necessária, pois não queremos toda a informação que é retornada. A informação que é retirada referente à bibliografia, é a seguinte:

- Título;
- Autor(es);
- URL (se existente);
- Assunto(s) associado a cada bibliografia;
- Formato do registo (ex: livro, formato digital, etc.);
- Exemplares existentes na biblioteca da FEUP;

4. **Visualização da Bibliografia Recomendada** – a visualização é efectuada recorrendo a um interface *Web* através de um browser, recorrendo a páginas ASP .NET instaladas no Learning Resource Centre (LRC) da LUVIT. Depois de a bibliografia ter sido retornada, o utilizador poderá filtrar os seus registos por **assunto** ou por **formato** (*clustering*).

4.2.2 SADPEL

Como já foi explicado na secção 3 (Revisão Tecnológica), optou-se por utilizar as redes bayesianas [2] como metodologia.

Uma *rede bayesiana* [3] é uma descrição concisa de distribuições de probabilidade. É definida por um grafo acíclico sobre nós representando variáveis aleatórias, em que cada variável tem uma matriz das probabilidades condicionadas, especificando a sua probabilidade dado a combinação de cada uma das probabilidades das variáveis das quais a variável aleatória depende directamente.

Qualquer probabilidade da rede é calculada desta forma:

$$P(x_1, \dots, x_n) = \prod_i P[x_i \mid x_{\text{pais}(x_i)}] \quad (1)$$

Em que $P[x_i \mid x_{\text{pais}(x_i)}]$ está tabelado para cada variável da rede.

As redes bayesianas [3] podem ser consideradas como diagramas que organizam o conhecimento numa dada área através de um mapeamento entre causas e efeitos. Muitas das variáveis aleatórias são independentes, o que permite ter tabelas que relacionam só um número reduzido de variáveis aleatórias.

As variáveis causas do SADPEL são: *o modelo de utilizador*.

Os efeitos são: *que páginas a mostrar* e *as opções pedagógicas a tomar*.

O algoritmo que é usado para fazer a inferência nas *redes bayesianas* é a simples propagação das probabilidades dos nós das causas (variáveis que caracterizam o utilizador), para os nós de efeito (as acções do formador através da equação que define as *redes bayesianas*) [2].

4.2.2.1 Plataforma de Desenvolvimento

A plataforma de desenvolvimento que foi utilizada, é dividida em cinco blocos, alguns dos quais dependentes do contexto da informação a ensinar, e outros intrínsecos ao SAD e independentes da base de conhecimento.

Tem-se assim um módulo de utilizador e a base de conhecimento que são construídos em função dos conhecimentos a ensinar.

O módulo de adaptação, o módulo pedagógico e o módulo de apresentação, usam a informação existente no módulo do utilizador e a base de conhecimento para dar funcionalidade e adaptabilidade à transmissão de conhecimento.

Recorreu-se outra vez à linguagem UML para a especificação/descrição da plataforma de desenvolvimento:

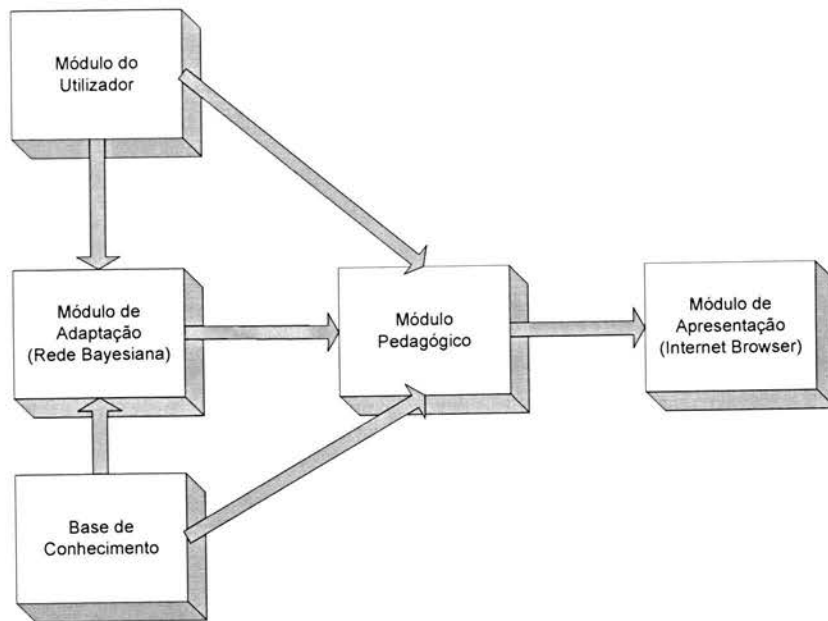


Fig. 16 – SADPEL (Plataforma de Desenvolvimento)

Módulo de Utilizador

Para caracterizar o utilizador, escolheram-se um conjunto de variáveis gerais para inferir o estilo de aprendizagem e variáveis associadas a cada bloco de informação da base de conhecimento:

Variáveis gerais do utilizador:

- Tempo médio dispendido por página;
- Costuma rever;
- Costuma aprofundar;
- Costuma ler superficialmente;
- Responde a perguntas;

Variáveis para cada bloco de informação:

- Tempo dispendido;
- Respostas certas;
- Respostas erradas;

A cada acção do utilizador, estas variáveis são ajustadas, para permitir que os módulos seguintes tenham em conta toda a informação que se possa ter do utilizador.

Base de Conhecimento

A base de conhecimento, contém a informação específica que é apresentada ao utilizador, o texto, figuras, o conjunto de perguntas, bem como a estrutura relacional entre os vários tópicos para o formador guiar o utilizador.

A estrutura interna da base de conhecimento, foi implementada num modelo de objectos, em que se cria um objecto por página, associando-lhe uma página ASP/HTML, inserindo um conjunto de perguntas e indicando as *páginas pais*.

Neste objecto (representado por uma classe), estão as variáveis que a acção do utilizador vai fazer modificar. A estrutura do objecto *UserPage* é a seguinte:

UserPage

```
{
    titulo //titulo da página
    fichHTML //o texto e imagens
    listaPerguntas //lista de perguntas
    numPerguntas //número de perguntas
    numPerguntasCertas //respostas certas
    numPerguntasErradas //respostas erradas
    tempoPagina //tempo na página
    paginaVisitada //página visitada
    tempoPadrao //tempo médio para a página
    listaPaginasPais //lista das páginas pais
    numPaginasPais //número de páginas pais
}
```

Em que *listaPerguntas* é um objecto do tipo *lista*; cada item da lista, é um objecto *Pergunta*:

```
Pergunta
{
    texto //texto da pergunta
    ajuda //texto explicativo caso erro
    resposta //resposta é sim ou não
    perguntaRespondida //pergunta já respondida
}
```

A página ASP/HTML, é concebida separadamente do formador e as perguntas relacionadas com o texto da página, são de resposta sim ou não.

Estes dados são fornecidos aos Módulos de Adaptação e Pedagógico, para poderem decidir as acções a tomar.

Módulo de Adaptação

Este módulo, é gerado a partir da estrutura de objectos criando-se uma *rede bayesiana* para inferir como está aprendido cada tópico e que tópico deve ser mostrado.

Como exemplo, irá ser apresentado um objecto com os seguintes tópicos:

Tópico A

Sub-tópico A1

Sub-tópico A2

Tópico B

Existe uma relação na informação do Tópico B e o Sub-tópico A2. A *rede bayesiana* ficaria com o formato da figura abaixo.

Cada nó *Aprendido* depende do *tempo passado* (T) e das *perguntas respondidas* (?):

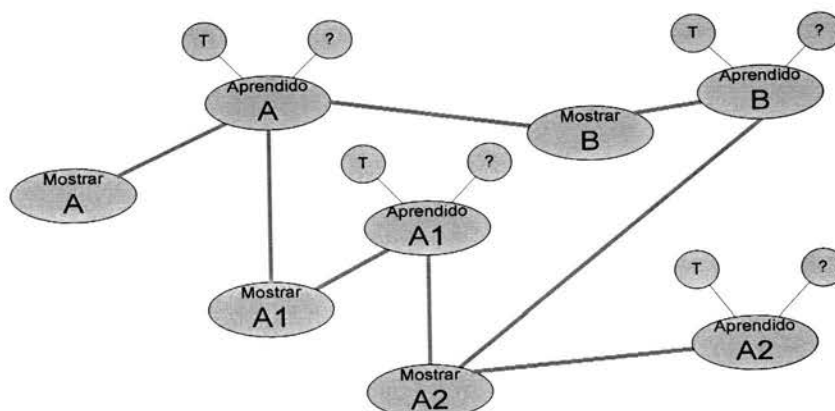


Fig. 17 – Rede Bayesiana da Base de Conhecimento

T = Tempo na página / Tempo padrão

? = (Perguntas certas - Perguntas erradas) / Número de Perguntas

Cada nó *Mostrar*, depende dos nós *Aprendido* pais e do próprio nó *Aprendido*. Todos os nós são calculados e as probabilidades que resultam desses cálculos são passadas para o Módulo Pedagógico.

Módulo Pedagógico

Este módulo toma as decisões sobre o que fazer a cada momento no SAD.

Quando o utilizador entra no sistema, é-lhe pedido para fazer um curto teste que serve para verificar os seus conhecimentos (iniciando alguns valores da *rede bayesiana*) e avaliar o seu conhecimento inicial.

Quando o utilizador achar que já aprendeu o suficiente sobre todos os temas, é-lhe apresentado um teste final. Este servirá de comparação com o teste inicial, como medida absoluta para avaliar a sua aprendizagem (se progrediu ou não).

Enquanto o utilizador está a ler uma página, o módulo pedagógico apresenta perguntas sobre o tópico. A pergunta é trocada de tempo a tempo, despertando a atenção do utilizador para que ele responda, visto não ser obrigatório responder às perguntas apresentadas.

Dependendo das respostas aos testes, o Módulo Pedagógico tem quatro opções a tomar:

- Aprofundar o tópico;
- Passagem superficial;
- Rever os tópicos anteriores;
- Ver tópicos novos;

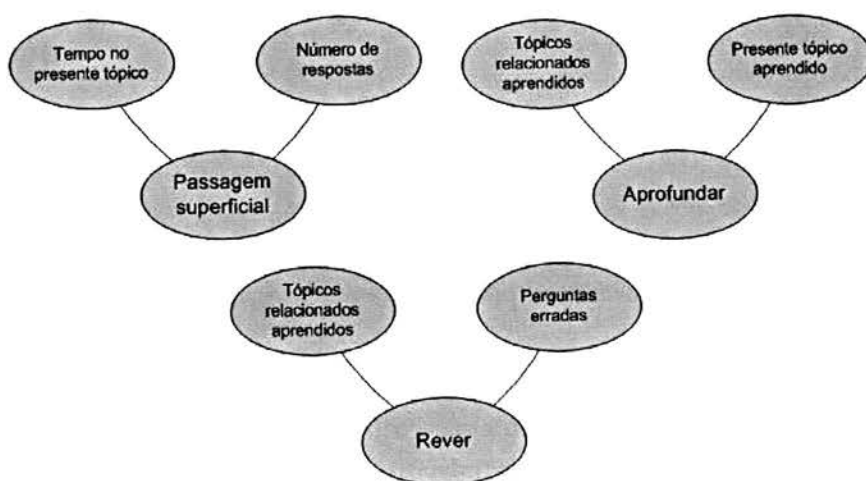


Fig. 18 – Rede Bayesiana da acção a tomar

Para tomar esta decisão, cria-se um conjunto de *redes bayesianas* que utilizam a informação que a rede do módulo da adaptação gerou. As três redes da figura acima são usadas para tomar a decisão do próximo tópico a mostrar (novo tópico), em que se compara a probabilidade resultante com as probabilidades dos nós *Mostrar* da *rede bayesiana* da Base de Conhecimento. Escolhe-se a acção cujo nó apresentar um valor probabilístico mais elevado.

$$Acção = \arg \max [aproveitar, passagem superficial, rever, \max_i (mostrar(i))] \quad (2)$$

Se a decisão for:

- **Aprofundar:** apresenta-se a página descendente da actual que seja completa;
- **Passagem Superficial:** apresenta-se a página seguinte que estiver no índice e seja “pouco profunda”;
- **Rever:** escolhe-se a página *Pai* que tiver o valor probabilístico mais elevado no nó *Mostrar* correspondente;
- **Ver tópicos novos:** significa a passagem para uma nova *rede bayesiana* baseada nas outras três opções, mas num contexto de um tópico diferente (novo);

Esta estrutura permite o sequenciamento curricular através dos tópicos do SAD, em que este determina o estilo de aprendizagem, o que é ideal para utilizar o IMS SS (IMS Simple Sequencing) [10], que é explicado mais abaixo.

Módulo de Apresentação

O Módulo de Apresentação, é o que interage com o browser da Internet. Compõe uma página com a informação gerada ou fornecida pelos blocos anteriores. A apresentação é dividida em cinco áreas:

- Navegação;
- Perguntas;
- Tópicos;
- Sugestões;
- Índice;

Navegação – tem três botões que permitem: seguir para o próximo tópico (que usa o resultado do módulo pedagógico através da fórmula (2)), ver o último tópico (que apresenta a última página) e *rever* (que mostra a página *Pai* do presente tópico que o módulo pedagógico indicar);

Perguntas – as perguntas apresentadas podem ser ou não respondidas, mas como o acompanhamento da aprendizagem do utilizador através de perguntas pode ajudar a conduzi-lo melhor pelo tutor, as perguntas são periodicamente mudadas para despertar a atenção do utilizador;

Tópicos – cada tópico apresentado, é representado por uma página ASP/HTML;

Sugestões – as sugestões são as três decisões do módulo pedagógico que apresentam um valor probabilístico mais elevado;

Índice – aqui está a estrutura convencional de um livro em que o utilizador pode também navegar;

Todas as acções do utilizador, são enviadas para o modelo do utilizador para permitir a actualização das variáveis acerca do utilizador e poder inferir melhor qual o percurso que o mesmo deve fazer pelo SAD.

4.2.2.2 O SCORM no SADPEL

O objectivo, é o de tentar aplicar algumas técnicas do SAD ao SCORM.

Porquê o SCORM?

- Porque queremos criar conteúdo curricular que detecte o desempenho e progresso do formando;
- Porque vamos utilizar um LMS para fornecer e gerir o conteúdo curricular;
- Porque será criado conteúdo que poderá ser reutilizado noutros contextos de e-Learning;
- Porque poderá ser preciso importar e executar conteúdo SCORM de outras fontes, em adição ao nosso próprio conteúdo;

O que é o Sequenciamento?

O *sequenciamento* é a ordenação eficiente de conteúdos de uma maneira que ajude o formando a obter os seus objectivos.

É a descrição de como os SCO's (Sharable Content Objects) são combinados.

Diversas estratégias de sequenciamento podem ser usadas na mesma sessão – possivelmente um diferente para cada objectivo, se aplicável.

Porquê?

Organizar o conteúdo de acordo com um esquema específico de sequenciamento, fornece um método ordenado para apresentar o conteúdo que será mais provável corresponder às expectativas do formando.

Os passos de aprendizagem são simples e confiáveis; eles permitem que o *designer* controle a ordem das experiências encontradas pelo formando.

Esta tecnologia será útil para quando o SAD precisar de entregar o objecto (SCO) mais apropriado ao formando.

4.2.2.3 IMS SS (IMS Simple Sequencing)

A especificação IMS Simple Sequencing (Sequenciamento Simples) [10]:

- É a enumeração de sequenciamento e de regras de comportamento;
- Trata o conteúdo como uma hierarquia de actividades;
- Adiciona regras descritivas aos objectos de aprendizagem (*content objects*) na hierarquia;
- Define múltiplos “objectivos” por objecto de aprendizagem (*learning object*);
- Define específicos tipos de regras que são combinadas para descrever comportamentos;

Fornece o modelo de actividades de ensino, organizado como uma árvore de *clusters* de actividades primitivas.

Esta árvore de actividades é mapeada directamente nos itens da árvore de documentos, do documento da organização mas com uma especificação separada de como é efectuado o processo de aprendizagem ao longo da árvore.

Pode ser especificado uma acção para cada item dependendo das condições.

As acções permitidas são: mudar para uma actividade diferente ou uma mudança de estado do LMS.

As condições podem se referir ao estado do LMS ou a um pedido de navegação por parte do utilizador.

A acção de mudar o estado do LMS, é usada para a especificação do progresso de objectivos. O condicionamento dos objectivos, é o mecanismo primário para definir mais actividades de ligação comportamentais em diferentes *clusters*. Os objectivos podem ser especificados separadamente do processo de aprendizagem e pode ser ou local (para actividades específicas) ou global (para diversas actividades).

A especificação não especifica quais os objectivos existentes, mas apenas normaliza num esquema de pontuação, uma sintaxe e um ambiente em *run-time*.

Na seguinte figura, poderemos ver um exemplo do Sequenciamento Simples utilizando a ferramenta de edição Reload [17]:

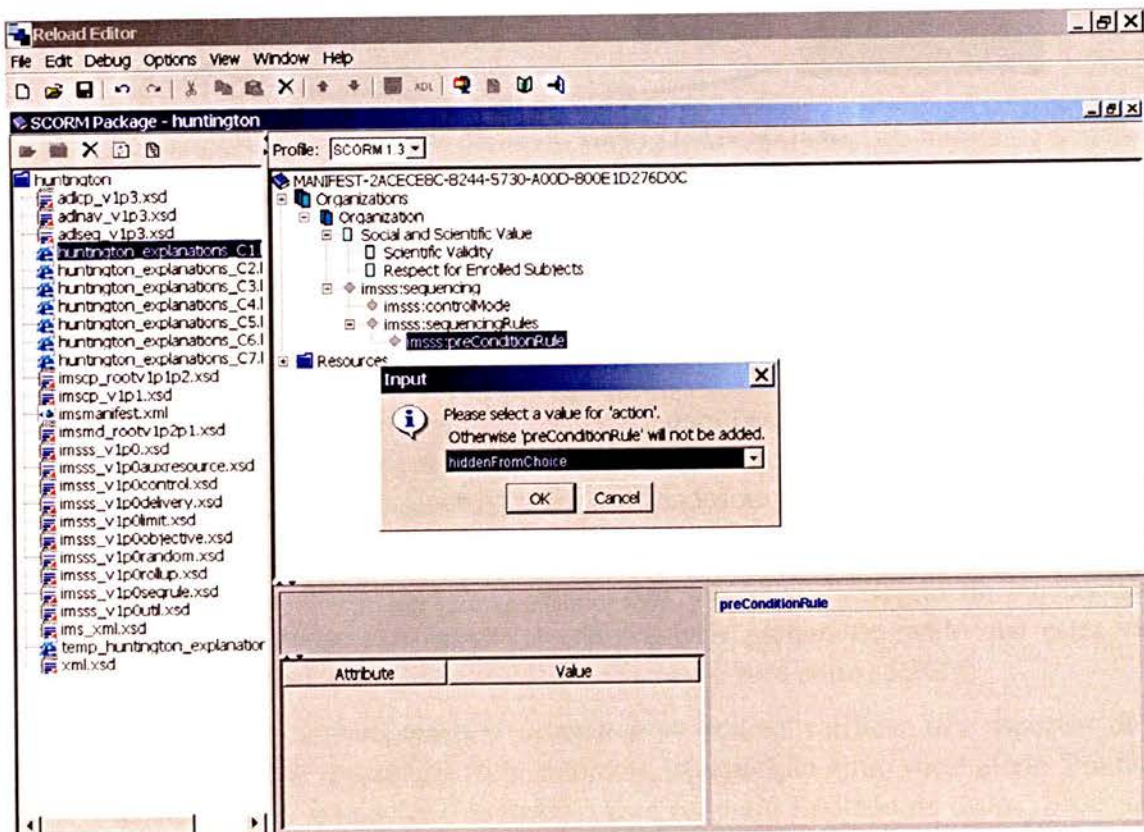


Fig. 19 – Exemplo de Sequenciamento Simples (Simple Sequencing)

4.2.2.4 Modelo Geral de Aprendizagem para o Sequenciamento Adaptativo

Esta secção descreve um modelo geral de aprendizagem para o sequenciamento adaptativo das lições, que pode ser aplicado ao modelo SCORM.

Este modelo permite:

- Que as actividades de aprendizagem sejam representadas por uma árvore de objectos de conteúdo;
- Definir regras de sequenciamento para as actividades;
- O controlo por parte do motor de sequenciamento, de todo o processo comportamental;
- Uma actividade pode ser um *assessment*;
- Os resultados da actividade modificam o sequenciamento;

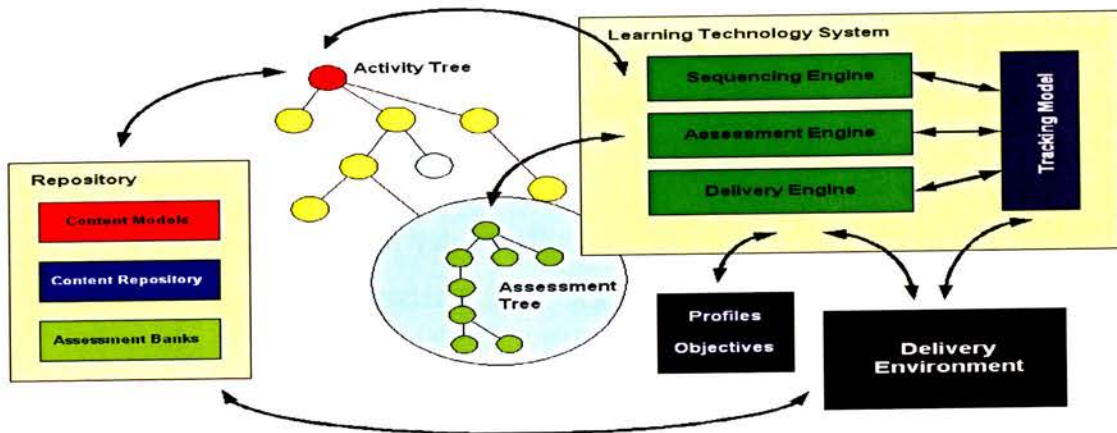


Fig. 20 – Modelo de Sequenciamento SCORM

4.2.2.5 Mapeamento do Modelo Geral de Aprendizagem ao SCORM

Esta secção descreve como cada passo no modelo de aprendizagem mapeia às capacidades do SCORM. Para manter o *Simple Sequencing* de uma forma simples de entender e implementar, os criadores da especificação limitaram o número de regras e dados que podem ser usados para fazer decisões de sequenciamento. Os passos para o mapeamento são os seguintes:

1. Embora guardar as respostas no LMS para questões individuais não seja estritamente necessário, o SCO pode reportar o desempenho do formando para questões individuais usando para tal as interações do modelo de dados no RTE;
2. O SCO pode agregar as respostas do formando às questões em objectivos e pode reportá-los usando o modelo de dados dos "objectivos". Regras de sequenciamento podem ser criadas para mapeá-las em variáveis globais de modo que estes valores possam ser usados por regras de sequenciamento para outros SCO's;
3. As regras de sequenciamento existentes só podem verificar três aspectos de uma variável global: pontuação (um número), compleição (um vocabulário limitado) e satisfação (um vocabulário limitado). Este conjunto limitado de dados, pode não ser suficiente para suportar todos os tipos de dados necessários para os algoritmos de sequenciamento;
4. O *Simple Sequencing* tem suporte para agregar uma pontuação baseada nas pontuações de todas as lições, num conjunto de lições. Mas é limitado a um único valor agregado para o conjunto de lições;

5. O *Simple Sequencing* não suporta *ranking*; as regras de cada SCO são verificadas individualmente para determinar se o SCO está disponível para o formando começar ou não, sem ter qualquer noção se um SCO é recomendado em relação a outro; Para além do mais, o Sequenciamento não tem a capacidade de verificar o conteúdo ou metadados de um SCO para determinar o *ranking*;
6. O *Simple Sequencing* não suporta a desactivação temporária de alguns SCO's. Um SCO ou está disponível ou não. O *Empacotamento de Conteúdos* tem uma opção de *isVisible* para um SCO, mas esta não pode ser configurada dinamicamente;

Templates e Modelos

Os *templates* de sequenciamento [6] descrevem potenciais comportamentos dos SCO's de acordo com várias estratégias. Os templates são criados para assistir na estruturação do conteúdo, de modo a compatibilizar com as regras de sequenciamento do SCORM.

Qualquer template ou combinação de templates pode ser combinado com outro template, criando assim uma estratégia mais complexa para um curso ou aula.

A combinação de templates permite modelos de sequenciamento [6] viáveis que podem ser adaptáveis de modo a ir de encontro aos requisitos educacionais pretendidos.

Estes templates irão ajudar a identificar novas maneiras com as quais podemos criar conteúdo SCORM atendendo às regras de sequenciamento e o verdadeiro objectivo do SCORM: criar materiais educacionais reusáveis, interoperáveis, duráveis e acessíveis.

Seguidamente, temos um exemplo simples de um template.

Para completar a agregação, o formando tem que completar o SCO.

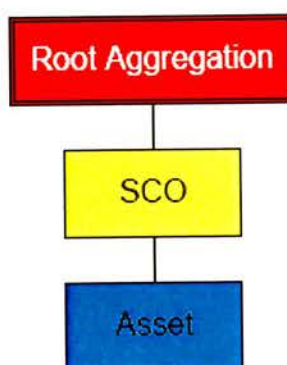


Fig. 21 – Template 1 – Um único SCO

5 Desenvolvimento do Protótipo

5.1 Plugin Course Readings

Nesta secção, apresenta-se o desenvolvimento de um protótipo como exemplo da Análise e Desenvolvimento na Implementação do projecto final. Este plugin foi concluído e encontra-se em correcto funcionamento. No sentido de o testar, foram efectuados testes a todas as suas funcionalidades especificadas nos diagramas de caso de uso.

As dificuldades encontradas na sua concepção foram de ordem muita técnica, que obrigaram a um conhecimento profundo da *API* da LUVIT. Estas dificuldades estão relacionadas com a plataforma da LUVIT:

1. Em saber o que era um plugin no LUVIT;
2. Onde é que se podia instalar esse plugin (as suas *páginas Web* e *dll* compiladas);

Para resolver este problema, foi preciso criar no *Internet Information Services (IIS)*, um directório virtual que funcionou como uma área de trabalho para guardar o conteúdo programado.

Este processo de toda uma aprendizagem permitiu saber como activar funcionalidades que ajudarão à concretização do SADPEL. Permitiu ter o conhecimento de:

- Como criar menus;
- Como manipular a *navigation tree* para inserir conteúdos;
- Que componentes deverão ser utilizados para activar o funcionamento correcto dos dois itens atrás mencionados;
- Como criar um plugin;

Como exemplo, será mostrado como é criado um novo documento no LEC para um determinado curso. Este documento é responsável por activar o plugin.

Descrição

1. Como qualquer outro documento no LUVIT Education Centre, um documento do tipo Course Readings, permite configurar o nome deste através da apresentação de uma *textbox* e a possibilidade de abri-lo numa nova janela. Esta funcionalidade só pode ser acedida pelo administrador do curso.

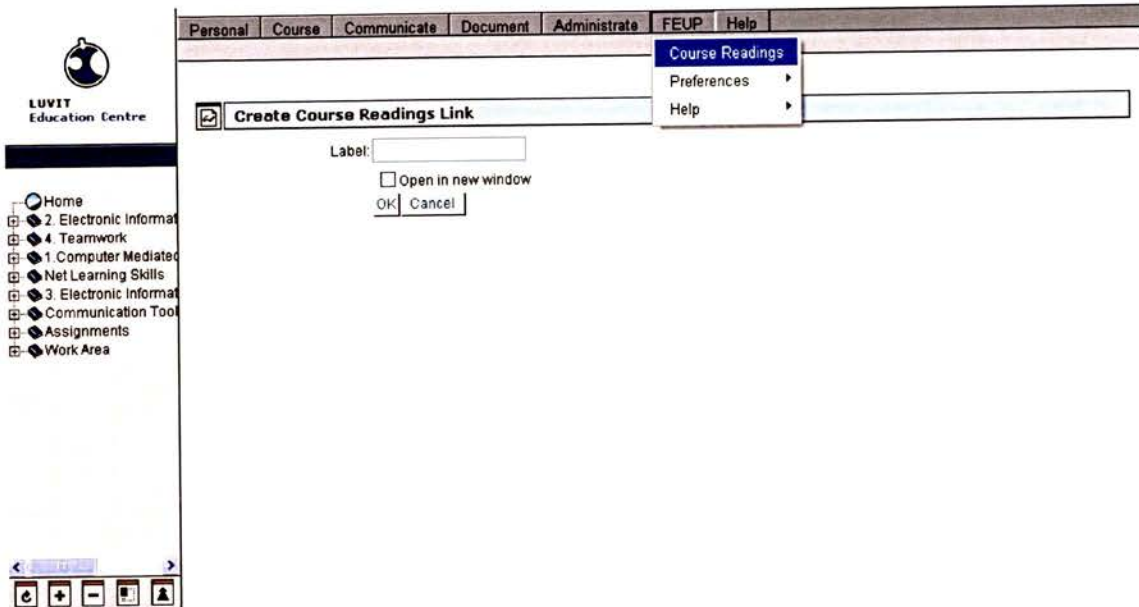


Fig. 22 - Criação de um novo documento Course Readings

2. O documento Course Readings torna-se então disponível no directório do utilizador (*My Archive*), podendo depois ser publicado na árvore de navegação de conteúdos.

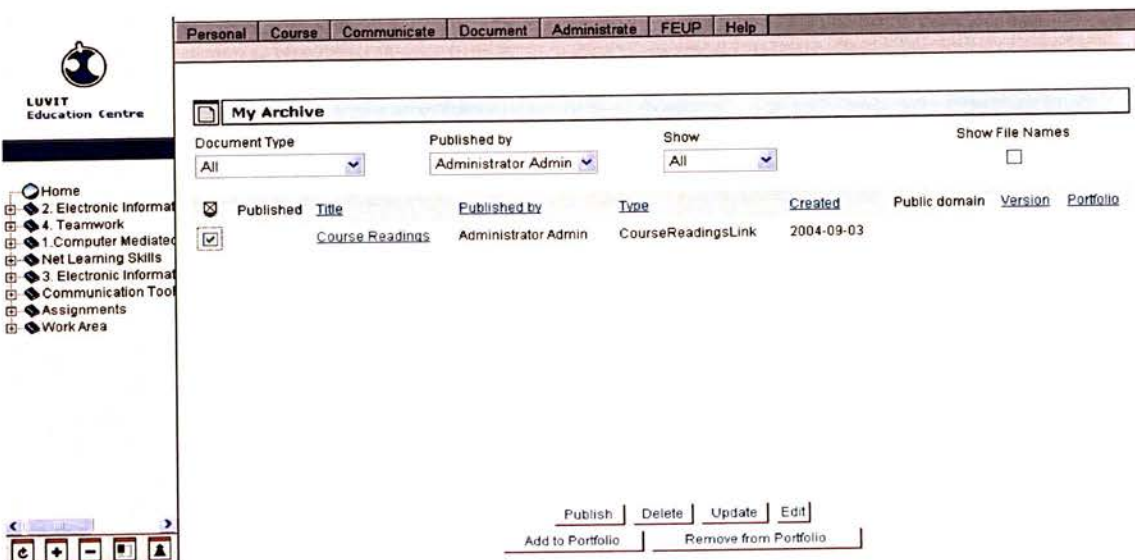


Fig. 23 - Novo documento Course Readings no My Archive

3. O professor pode seleccionar onde colocar esta informação e restringir o acesso ao documento. As estatísticas sobre documento, são automaticamente cedidas pelo Learning Management System.

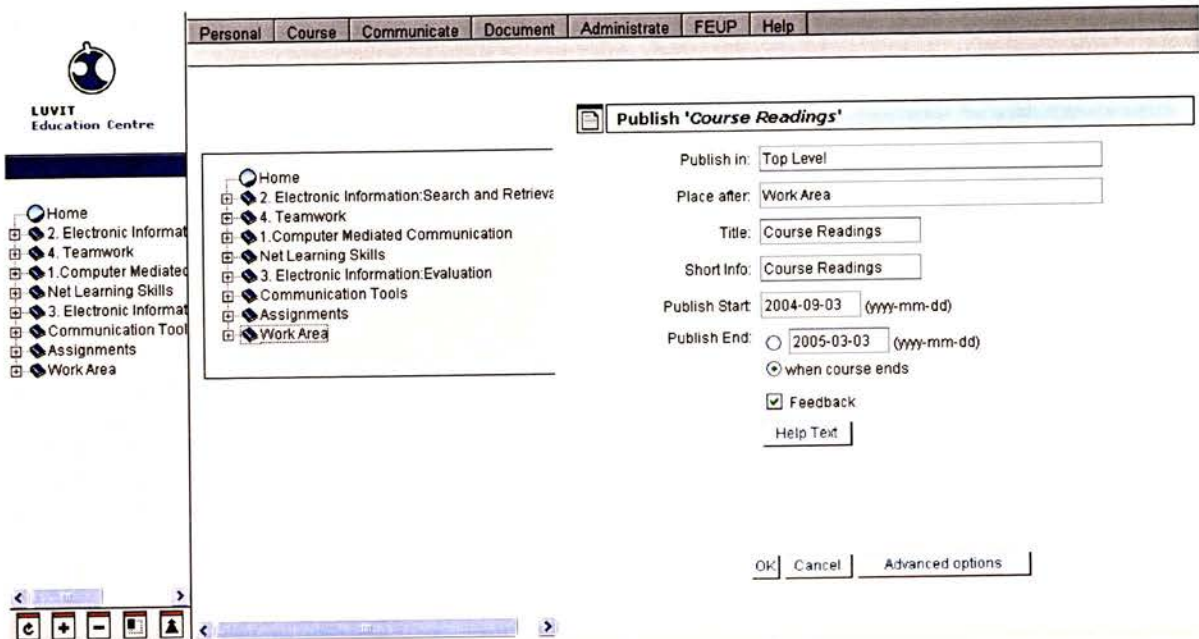


Fig. 24 - Publicação do Course Readings

4. Quando um utilizador clica no ícone do documento, o plugin "interroga" a base de dados do ALEPH e retorna a bibliografia para este curso específico. O utilizador pode também filtrar os recursos desta bibliografia, por **assunto** ou por **tipo de documento**, seguir um **URL** quando disponível ou verificar as cópias existentes na biblioteca.

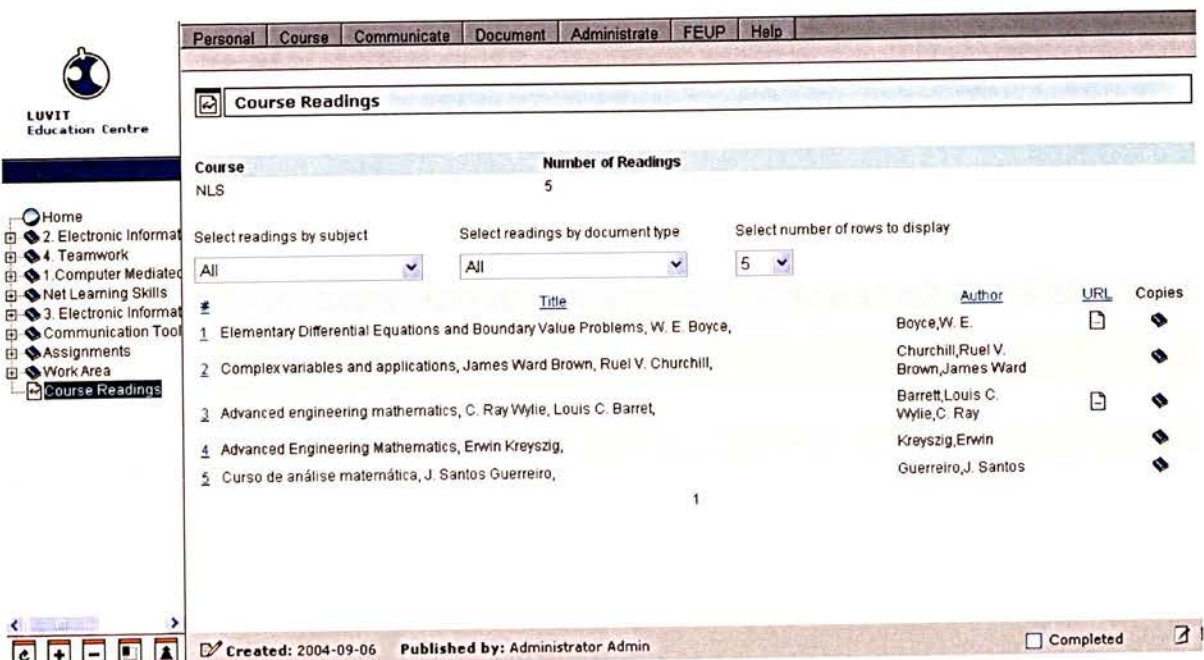


Fig. 25 - O plugin Course Readings

5.1.1 Preferências

As preferências do plugin, onde é possível adicionar (botão *New*), apagar (botão *Delete*) e editar (botão *Edit*).

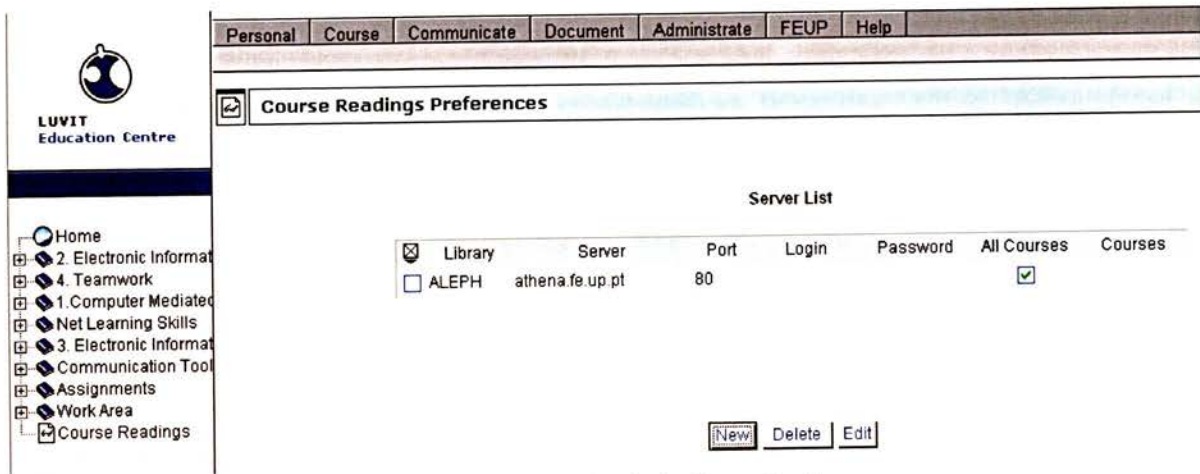


Fig. 26 – Preferências do plugin Course Readings

5.2 Plugin Content Readings

Não foi possível cumprir com a totalidade do *SADPEL*, uma vez que este se revelou demasiado ambicioso em relação ao tempo disponível para o estágio. Como tal, o protótipo do *plugin Content Readings* não foi ainda concebido.

Foram concluídas as fases de Análise e Arquitectura, faltando concluir a fase de Desenvolvimento que se encontra em curso.

6 Conclusões e Considerações Gerais

Este projecto permitiu especificar dois plugins complementares que estabelecem a ligação entre um sistema de e-Learning, a biblioteca digital e o sistema de informação, do ponto de vista comunicacional.

Para o plugin Course Readings, foram concluídas as fases de Análise, Arquitectura e Desenvolvimento.

Para o SADPEL, que implementará o plugin Content Readings, foram concluídas as fases de Análise e Arquitectura, faltando concluir a fase de Desenvolvimento.

Uma das dificuldades, foi encontrar um modelo de raciocínio baseado na incerteza, para implementar o SAD.

Para tal, foi necessário realizar as seguintes tarefas:

- a) Revisão sobre os conceitos básicos sobre Teoria dos Grafos, Teoria da Probabilidade, Teoria da Incerteza, Sistemas de Apoio à Decisão e Inteligência Artificial;
- b) Identificação do algoritmo mais promissor;
- c) Identificação das deficiências de outros algoritmos;
- d) Modelagem do método de aprendizagem (neste caso da Rede Bayesiana [3]), considerando as variáveis do domínio de aplicação para apoio à tomada de decisão;

Outra das dificuldades foi a instalação de componentes no LMS, de modo que o plugin ficasse funcional.

Uma das facilidades encontradas na instalação, foi o facto de a arquitectura da LUVIT suportar componentes *COM* (*COM* é um acrónimo para *Component Object Model* e é um standard Microsoft criado para facilitar a comunicação entre objectos programados). Para criar componentes *COM* que comunicassem com os componentes *COM* da LUVIT (programados em Microsoft Visual Basic), foi utilizado o Microsoft Visual Studio .NET 2003 como ambiente de programação. Esta ferramenta, revelou-se potentíssima em todos os aspectos, pois permitiu facilitar a comunicação acima referida sem necessidade de migrar esses componentes *COM* em componentes .NET, evitando assim os custos associados à migração.

O plugin Course Readings foi apresentado no ICAU (Conferência Internacional dos Utilizadores do ALEPH) que decorreu na FEUP. Para este propósito, foi apresentado um poster intitulado "*Integrating Library Course Readings within a Learning Management System*", seguido da demonstração do plugin.

Foi também concebida uma arquitectura genérica na forma de um artigo científico, com o objectivo de reutilizar funcionalidades comuns, entre este projecto e projectos similares, que permita um maior isolamento e independência da plataforma de e-Learning. A arquitectura, intitulada de "*Framework architecture specification for LMS plugins (a modern approach)*" [1], será apresentada na conferência do *e-LES'04-Aveiro* (e-Learning no Ensino Superior).

Mais concretamente, este estágio permitiu:

- O estudo da tecnologia utilizada na LUVIT LMS;
- O estudo das normas para e-Learning *IMS/IEEE/ADL*, nomeadamente *IMS/CP*, *SCORM/RTE*;
- Implementar bibliografia em *real-time* para ser usada num sistema de e-Learning;
- Implementar o conceito de sistema de apoio à decisão em real-time, na escolha dos conteúdos mais adequados, para otimizar a aprendizagem por e-Learning;
- Desenvolver uma arquitectura para a inclusão Just-In-Time de conteúdos SCORM, num curso LUVIT com base em requisitos especificados por *Simple Sequencing* [10];
- Investigação das Redes Bayesianas [3], estabelecendo intersecções entre a abordagem estatístico-Bayesiano e a manipulação da incerteza;
- A especificação de um ambiente computacional para aprendizagem de Redes Bayesianas [3], considerando as variáveis mais relevantes para a tomada de decisão, sendo um grande avanço em pesquisas na área do e-Learning;

A integração do estagiário na instituição de estágio foi excelente em todos os aspectos e a formação dada por esta foi altamente louvável no sentido em que permitiu a rápida familiarização com a LUVIT AB e a sua plataforma.

6.1 Avaliação de Resultados

O plugin Course Readings encontra-se completo e a sua apresentação no ICAU teve impacto no sentido em que permitiu à FEUP e em particular à uTICM, obter prestígio internacional e a estar na vanguarda de projectos deste tipo. A LUVIT também apreciou o plugin e mostrou-se bastante impressionada por todo o processo de elaboração, que permitiu trabalhar ao mais baixo nível com a plataforma LUVIT LMS [11].

O SADPEL não apresenta ainda resultados visíveis, no entanto, o relatório especifica em termos detalhados como este deverá ser desenvolvido. A funcionalidade do plugin Course Readings permite ter uma noção do que será a funcionalidade do plugin Content Readings, no sentido em que este também será instalado e utilizado de uma maneira semelhante.

A publicação do artigo científico é importante para obter reconhecimento, estabelecer relações e manter altos padrões de *benchmarking* científico em prol de uma contínua actualização.

6.2 Futuros Desenvolvimentos

Aqui serão ponderados futuros desenvolvimentos, alguns deles em fase actual de implementação, que funcionarão para melhorar/complementar as funcionalidades de todo o projecto.

- No plugin Course Readings, se um utilizador quiser ver as cópias existentes na biblioteca de um registo bibliográfico e depois efectuar a reserva do mesmo, precisa de estar autenticado no SiFEUP. Um dos próximos passos por parte da equipa de desenvolvimento da LUVIT é o de ter este mecanismo finalizado através do LDAP, de modo que ao entrar no LMS da LUVIT, um utilizador seja automaticamente autenticado no SiFEUP;
- O recurso ao *datawarehouse* [18]: será importante para a geração automática de cursos, que poderá vir a ser utilizada no novo formato do primeiro ano das licenciaturas da FEUP; devido ao crescimento da utilização da plataforma LMS, será utilizada uma estratégia de *data-mining* no *datawarehouse* [18] para possibilitar a pesquisa de padrões de comportamento na plataforma; através destes padrões, o SAD terá uma maior facilidade em decidir quais os conteúdos mais adequados para o formando;
- Estar atento a outras plataformas suportadas (actualmente *WebCT*) e estar preparado para efectuar eventuais migrações de dados entre sistemas. Em particular, considerar a evolução a prazo para cursos multidisciplinares com intervenção de docentes de outras faculdades ou a reutilização de objectos SCORM externos;
- A inclusão do SCORM 2004 [7] que permitirá, entre outras coisas usar: o *Simple Sequencing*, a navegação SCORM, o modelo de metadados standard LOM do IEEE;

6.3 Avaliação Geral do Estágio

De um modo geral, o estágio foi positivo, apesar do projecto SADPEL não estar completamente concluído.

A nível pessoal, enriqueceu os meus conhecimentos a todos os níveis, ajudou-me a trabalhar melhor em equipa e a lidar com a pressão de modo a cumprir os objectivos propostos.

O estágio permitiu a familiarização com o mundo do e-Learning, as tecnologias e arquitecturas por ele utilizadas, que cada vez mais se apoiam na migração de dados entre sistemas para evoluir o ensino à distância para níveis elevados na educação superior.

Referências e Bibliografia

- [1] Domingues M., Faria P., Nóbrega R., 2004, “*Framework Architecture Specification for LMS plugins (a modern approach)*” (artigo aceite no e-LES’04 em Aveiro, FEUP, Portugal, 2004).
- [2] Stuart J. Russell, Peter Norvig, 2002, “*Artificial Intelligence: A Modern Approach (2nd Edition)*”, Prentice Hall, 2002, ISBN 0-137-90395-2.
- [3] Gaag, Linda C. “*Bayesian Belief Networks: Odds and Ends. The Computer Journal*”, The Computer Journal, vol 39, nº 2, 1996, p 97-113.
- [4] Lopes, João, 2002. “*Acetatos de Laboratório de Engenharia de Software*”, LEIC, FEUP, Portugal.
- [5] SCORM Best Practices Guide for Content Developers. Learning Systems Architecture Lab (2003).
<http://www.lsal.cmu.edu/lsal/expertise/projects/developersguide/developersguide/guide-v1p0-20030228.pdf>.
- [6] SCORM Simple Sequencing Templates and Models. Learning Systems Architecture Lab (2003).
<http://www.lsal.cmu.edu/lsal/expertise/projects/developersguide/sstemplates/templates-v1p0-20030228.pdf>
- [7] SCORM 2004. Learning Systems Architecture Lab (2003).
http://www.lsal.cmu.edu/lsal/expertise/papers/presentations/idea2004/ideascorm_20040816.pdf
- [8] Advanced Distributed Learning Network. <http://www.adlnet.org/>
- [9] IMS Global Learning Consortium Inc. <http://www.imspjct.org/>
- [10] IMS SS. <http://www.imspjct.org/simplesequencing/>
- [11] LUVIT LMS Learning Management System.
<http://www.luvit.com/P00.m4n?language=en>
- [12] OAI. <http://www.openarchives.org/OAI/openarchivesprotocol.htm>
- [13] SCORM. <http://www.adlnet.org/index.cfm?fuseaction=scormabt>
- [14] Comparação entre Produtos de Software de Ensino Online.
<http://www.marshall.edu/it/cit/webct/compare/comparison.html>
- [15] ALEPH. <http://www.exlibris-usa.com/aleph.htm>
- [16] Oliveira, Carlos, 2004, “*Implementing an eLearning Workflow*”, artigo apresentado no e-LES’04 em Aveiro, FEUP, Portugal, 2004.
- [17] Reload Editor. <http://www.reload.ac.uk>
- [18] Domingues, Manuel, 2003, “*Concepção de um Armazém de Dados para apoio à gestão de um sistema de e-Learning*”, Relatório de Estágio Curricular da LEIC 2002/2003, FEUP, Portugal, 2003.

ANEXO A: Glossário

Este glossário identifica a terminologia com a qual o utilizador deve estar familiarizado para conseguir total benefício das secções do relatório.

Agregação de Conteúdos

Define a estrutura e os mecanismos para definir a sequência na qual os recursos de aprendizagem vão ser apresentados ao utilizador.

Asset

Um *Asset*, é um ficheiro único electrónico representando *media*, texto, imagem, *audio*, etc.

Podem ser incluídos mas não estão limitados aos tipos de ficheiro: *jpg*, *txt*, *wav*, *avi*, *html* e *gif*.

Assessment

O processo usado para avaliar as qualificações do formando ou nível de conhecimento sistematicamente.

Empacotamento de Conteúdos (Content Packaging)

Descreve a árvore estática que especifica a estrutura dos SCO's no curso, isto é, a organização geral da estrutura do curso e os objectos a ele associados.

O Empacotamento do SCORM adere estritamente ao do IMS [9], fornecendo directivas adicionais sobre empacotamento de recursos de aprendizagem digitais (Recursos, SCO, Agregados, etc.).

Ferramentas de Autor

É um programa ou software aplicativo que permite criar de modo personalizado um curso de e-Learning. Podemos distinguir como tipos de ferramentas de autor as ferramentas instrucionais, as de programação e as de *Web Authoring*. Estas ferramentas criam templates (*template-focused authoring tools*), sistemas de captura de conhecimento (*knowledge capture systems*) e de criação de ficheiros e texto.

IMS (Instructional Management System)

Instituições de educação superior dos E.U.A e os seus parceiros de venda, estabeleceram um esforço de desenvolver standards abertos baseados no mercado para o ensino online, incluindo especificações para metadados com conteúdo curricular. Presentemente, o *IMS Global Learning Consortium, Inc.* (IMS) está a desenvolver e promover especificações abertas para facilitar actividades curriculares distribuídas online, tais como a localização e uso de conteúdo educacional, a detecção do progresso do formando, troca de conteúdo estudantil entre sistemas administrativos. O IMS age de maneira a promover a adopção de especificações que permitirão ambientes de ensino e conteúdo distribuídos de modo que múltiplos autores possam trabalhar conjuntamente. O IMS conduziu um estudo nas instituições e organizações para determinar que elementos metadados do documento base do IEEE LOM eram mais fundamentais que outros. O IMS então adoptou um conjunto de elementos metadados escolhidos a partir dos resultados obtidos do estudo, da comunidade IMS nos E.U.A., Austrália, Ásia e Europa.

O standard é aberto e todos podem usá-lo.

O standard IMS [9] é definido por um número de *DTD's* (*Document Type Definition*) ou *schemas*. Estes DTD's ou schemas descrevem que elementos deverá consistir um ficheiro XML e que elementos são opcionais. O ficheiro XML é validado pelo ficheiro DTD para controlar que o ficheiro está de acordo com o standard. É também no ficheiro XML onde são guardados os dados. O standard IMS consiste neste momento das seguintes sub-partes: *Metadados*, *Empacotamento de Conteúdos* (empacotamento de cursos ou partes de cursos), *Questions & Tests* (testes), *Profiles* (utilizadores diferentes do material) e *Enterprise* (gere os dados pessoais e grupos).

Just-In-Time Learning

Característica de e-Learning, na qual os formandos têm a capacidade de aceder à informação que eles precisam, exactamente quando eles precisam.

Lightweight Directory Authentication Protocol (LDAP)

É um protocolo que permite homogeneizar os repositórios das contas de utilizadores, reduzir os tempos de resposta dos serviços de autenticação e aumentar os níveis de segurança.

Com esta infra-estrutura, para além de diversas vantagens ao nível técnico, criam-se novas oportunidades ao nível da autenticação de utilizadores. Com efeito estende-se agora, também aos sistemas operativos *Linux/Unix* departamentais, a possibilidade de autenticar utilizadores, nos seus sistemas específicos, recorrendo à informação existente centralmente e que é alimentada pelo Sistema de Informação (GAUP/SiFEUP), o que assegura a permanente actualização da informação sobre utilizadores. Desta forma, torna-se desnecessária a criação de contas nos sistemas departamentais, que além de redundantes (e por vezes desactualizados), implicam uma enorme carga de trabalho, associada à sua manutenção (criação, alteração de passwords, eliminação de contas, backups, etc.), aos administradores destes sistemas.

Learning Management System (LMS)

Um *LMS*, é uma aplicação de *software* ou tecnologia baseada na *Web* usada para planear, implementar e aceder um determinado processo de aprendizagem. Tipicamente, um LMS monitoriza a participação de um estudante e o seu desempenho. Pode também fornecer aos formandos a capacidade de usar funcionalidades interactivas tais como conferência por vídeo e fóruns de discussão.

LUVIT Education Centre (LEC)

LUVIT Education Centre (LEC) é uma aplicação baseada na *Web*, produzida especialmente para fornecer educação online. Trata conteúdos de cursos, pessoas (formadores, formandos, etc.) e muitas funcionalidades para a comunicação entre estes. Outra funcionalidade importante inclui tracking, estatísticas e publicação de conteúdos de curso.

O LRC é um repositório com objectos de cursos, enquanto que um curso que foi copiado e inicializado no LEC é um curso temporário (muitas vezes limitado no tempo).

LUVIT Resource Centre (LRC)

Este produto foi originalmente desenvolvido na Universidade de Lund em 1997 e hoje em dia é um ambiente completo para educação baseada na *Web* da LUVIT AB.

É a versão administrativa do LEC, onde são criados os cursos e transferidos conhecimento e informação pela Internet.

Manifesto (manifest)

Um *manifesto* é um ficheiro XML que descreve os componentes de pacote de conteúdos. Consiste nas seguintes secções:

- A secção de metadados descreve o pacote como um todo;
- A secção de organizações descreve um ou mais organizações hierárquicas do conteúdo (estrutura do conteúdo);
- A secção dos recursos referencia o actual recurso e os ficheiros media necessários ao conteúdo;

Metadados

Metadados são dados sobre dados. De uma perspectiva SCORM, os metadados representam um mapeamento e uso recomendado do *Learning Object Metadata* (LOM) do *Institute of Electrical and Electronics Engineers, Inc. Learning Technology Standards Committee* (IEEE LTSC) aos componentes do modelo de agregação de conteúdos (isto é, Assets, SCO's e agregações de conteúdos).

Estes metadados, são usados para descrever o asset, SCO ou agregação de conteúdos de uma maneira consistente e significativa, para activar a procura de componentes individuais através de um repositório de conteúdos.

Navigation Tree

É a árvore de navegação da LUVIT, onde todo o material dos cursos publicados é colocado e posto visível.

Objectos de Aprendizagem

São bocados de informação multimédia independentes e reutilizáveis, utilizados de forma modular na construção de um conteúdo de e-Learning.

Open Archives Initiative (OAI)

OAI é uma iniciativa para desenvolver e promover normas de interoperabilidade que têm o objectivo de facilitar a dessiminação eficiente do conteúdo.

OAI Protocol for Metadata Harvesting (OAI-PMH)

É um protocolo de *harvesting* (recolha de metadados) para partilha de metadados entre serviços.

Sharable Content Object (SCO)

Um *SCO* é uma colecção de um ou mais *assets* que é capaz de ser recebido por um browser *Web* e tem a capacidade de comunicar com um Learning Management System (LMS) através da *API* do SCORM Run-Time Environment (RTE).

Representa o mais baixo nível de granularidade dos recursos de aprendizagem que podem ser detectados por um LMS.

Sharable Content Object Reference Model (SCORM)

O *SCORM* ou *Sharable Content Object Reference Model* é um conjunto de especificações desenvolvidas pela iniciativa *Advanced Distributed Learning (ADL)* [8]. Esta iniciativa é patrocinada pelo Departamento de Defesa dos Estados Unidos da América e trabalha em cooperação com a indústria IMS e academia.

O objectivo é o de estabelecer um novo ambiente de ensino distribuído, que permita a interoperabilidade de ferramentas de ensino e conteúdos de cursos numa escala global.

As partes principais da especificação SCORM, são o modelo de metadados do IEEE-LOM, o modelo de empacotamento de conteúdos do IMS, o modelo de sequenciamento simples da IMS (IMS SS) [10] e um modelo de dados e ambiente em run-time específico do SCORM.

O modelo SCORM assume que o conteúdo é construído a partir de pequenas unidades de aprendizagem, os SCO's (Sharable Content Objects). Os cursos são desenhados criando um documento manifest do IMS descrevendo organizações. Uma organização é comparável a uma estrutura em árvore de documentos, cujos ramos consistem em SCO's e os nodos (itens) representam a estrutura interna do curso. Cada nível de estruturação (manifest, organização, item e SCO) pode ser anotado com metadados. A separação do conteúdo e organização significa que um mesmo SCO pode ser reutilizado em diferentes documentos.

Um SCO, tal como um asset, pode ser descrito com metadados para permitir a pesquisa, fornecendo informação descritiva acerca do conteúdo representado no SCO.

Com o uso dos metadados, os SCO's podem se tornar pesquisáveis nos repositórios de conteúdos, originando a sua reutilização.

Um SCO não pode invocar outro SCO.

SCORM RTE (SCORM Run-Time Environment)

Uma definição de ambiente em *run-time* que inclui um protocolo que inicia conteúdo executável baseado na *Web*, uma *API* de conteúdos LMS, e um modelo de dados definindo os dados que são trocados entre um ambiente LMS e conteúdo executável em *run-time*. A especificação *IMS Content Packaging (IMS/CP)* não endereça explicitamente os requisitos de uma ambiente em *run-time*, mas suporta na totalidade a descrição e troca de conteúdos que implementam atributos SCORM em *run-time*.

Template

Um conjunto predefinido de ferramentas ou *forms*, que estabelecem a estrutura e configuração necessárias para criar conteúdo rapidamente.

**ANEXO B: Apresentação do Plugin Course Readings na conferência do ICAU –
Porto, Portugal, 6 de Setembro de 2004**

**"Integrating Library Course Readings within a Learning
Management System"**

Carlos Oliveira: colive@fe.up.pt +

Luís Costa: lmcosta@fe.up.pt *

Manuel Domingues: mad@fe.up.pt +

Pedro Faria: pfaria@fe.up.pt +

Rúben Nóbrega: ruben.nobrega@fe.up.pt +

+Faculty of Engineering of the University of Porto/SICC/uTICM

*Faculty of Engineering of the University of Porto/SDI/DSE

Abstract:

"The objective of this work is to provide the recommended readings of a course in the Learning Management System (LMS).

This development was made for LUVIT LMS as a plugin that queries the ALEPH catalogue and returns all the records annotated with the corresponding course code.

The plugin architecture was designed so that other LMS or Library Systems can also be configured easily."

Plugin Architecture

The plugin is installed in LUVIT LMS Learning Education Centre and in the setup, the ALEPH library address is provided.

A new document type was defined in Luvit, so that the Course Readings can be managed in the workflow as any other document (publishing, access restrictions, user statistics).

A single course code was defined for both LUVIT and ALEPH systems.

The plugin gets the course code from LUVIT and calls the ALEPH interface (*alephx*).

ALEPH queries the database and returns an XML object with the resulting recordset, which becomes a file "bibliography.xml".

The plugin's XMLParsing class uses this file by parsing its data in order to extract the relevant information for visualization.

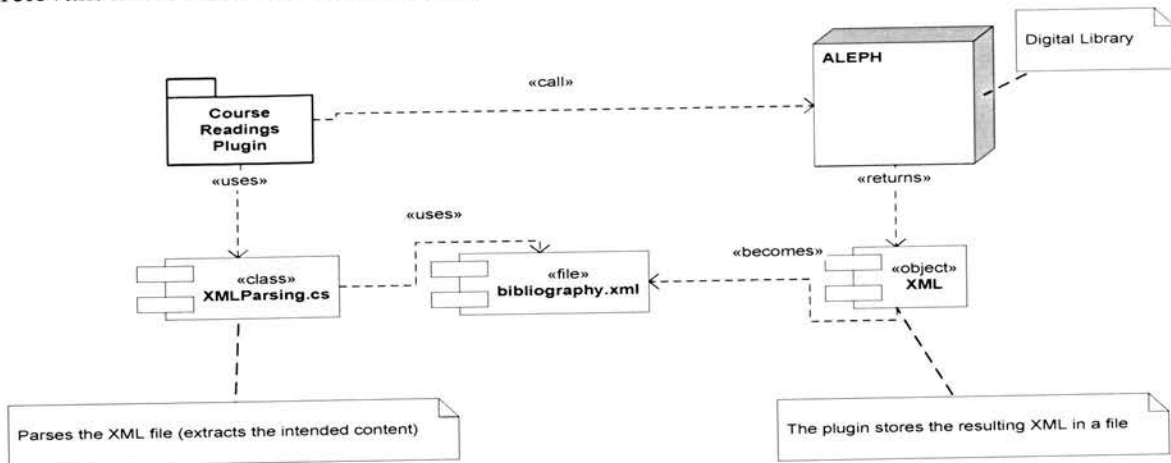


Fig. 1: Component diagram of the plugin's architecture

Plugin Development

The plugin development was divided in the following steps:

1. ALEPH query

Using the query protocol alephx two *HTTP* calls are made to ALEPH. The first call returns the query id and the resulting number of records for the requested course code, which is available as "610|a" in the UNIMARC format. The call has the following format:

`http://athena.fe.up.pt/X?op=find&base=eup01&request="coursecode"`

where:

`X?` : call to XML alephx service.

`op=find`: search operation that returns the query id and the number of records.

`base`: the library base where the search is performed.

`request`: the search data (in this case the course code).

The second call uses the query id to return a defined number of records in the intended format.

`http://athena.fe.up.pt/X?op=present&set_no="+query_id+"&set_entry=000000001&format=marc`

where:

`op=present`: record presentation operation.

`set_no`: uses the query id returned in the previous call.

`set_entry`: record set to be returned.

`format`: presentation format of the records.

2. XML result file

The resulting XML with the recordset is stored in a file for further processing.

3. XML Parsing

Parsing the recordset file the relevant fields are extracted:

Title, Author(s), Subject(s), Type of document, URL (if existent), Item list (if existent).

4. Visualization

The visualization of the resulting records is integrated in the LMS interface. The list can be filtered by subject or by document type.

Description

1- As any other document in LUVIT Education Centre, a Course Readings type document allows the setting of a presentation label and the choice to open it in a new window.

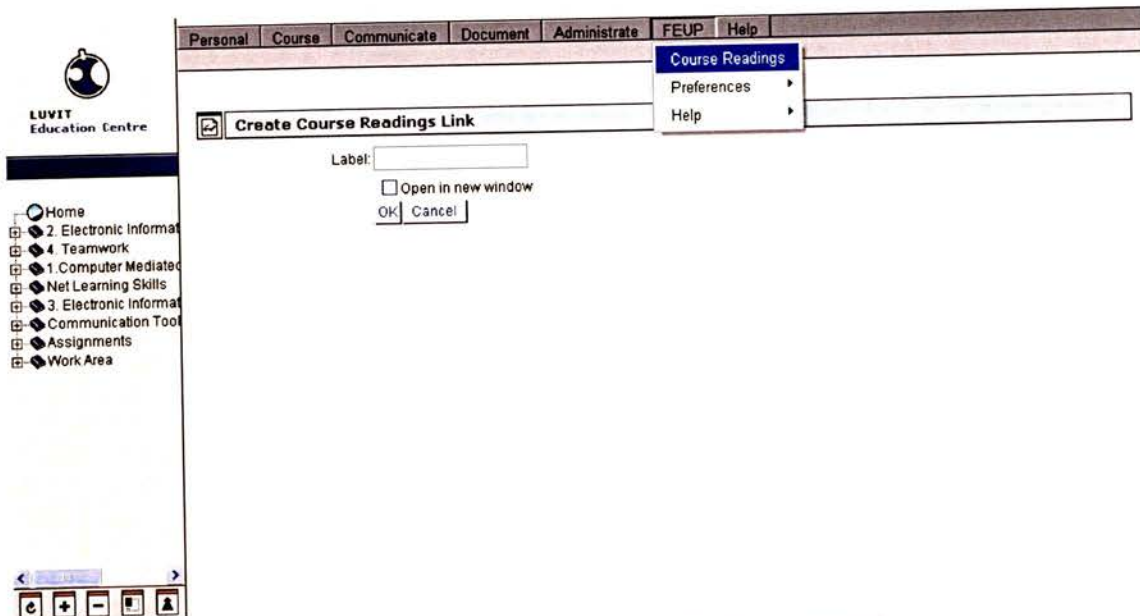


Fig. 2: Creation of a new Course Readings document

2- The Course Readings document becomes available in the user's archive and can then be published in the contents navigation tree.

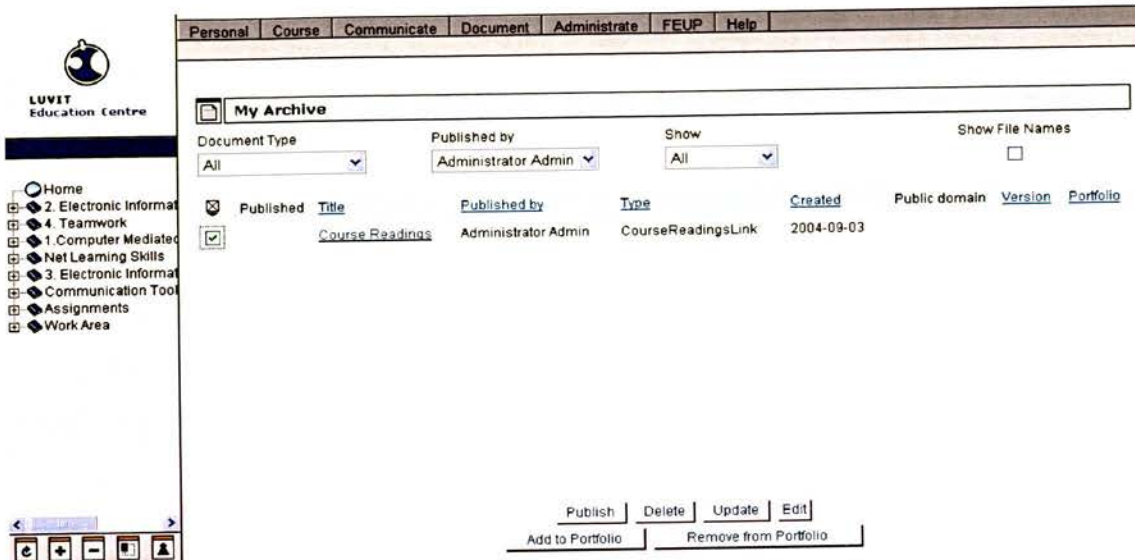


Fig. 3: New Course Readings document in My Archive

3- The teacher can select where to place this information and can restrict the access to the document. Statistics on the document's access are automatically provided by the Learning Management System.

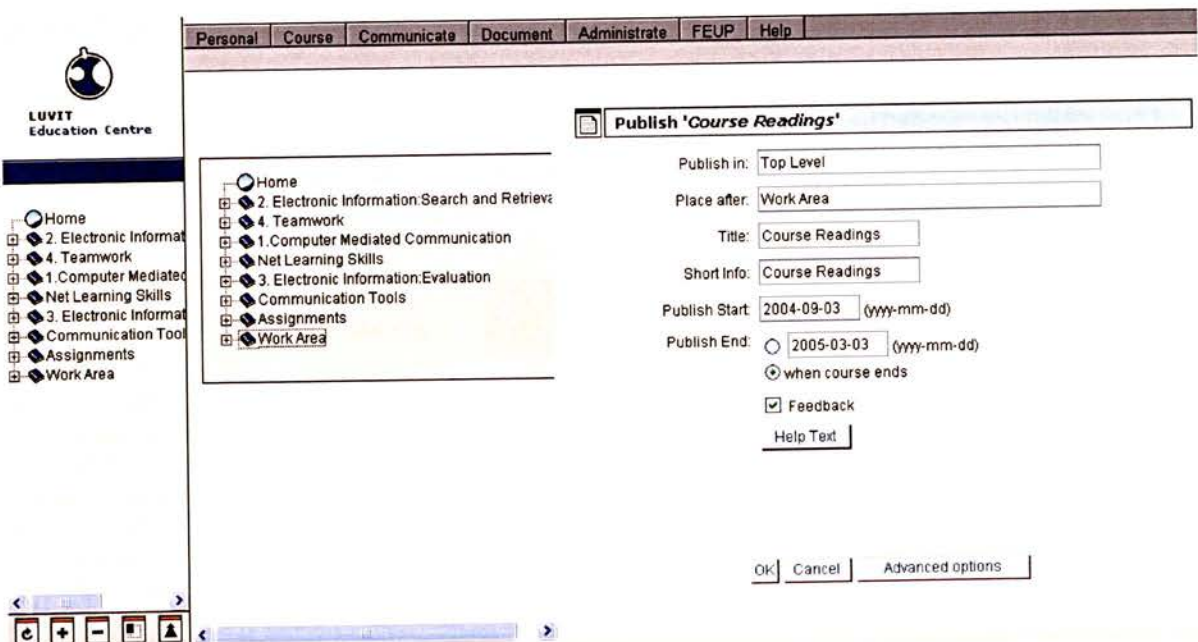


Fig. 4: Course Readings publishing

4- When a user clicks in the document's icon the plugin queries the ALEPH database and returns the course readings list for this specific course. The user can also filter these records by subject or by document type, follow an URL when available or check the existing copies in the library.


LUVIT
Education Centre

- Home
- 2. Electronic Information
- 4. Teamwork
- 1. Computer Mediated
- Net Learning Skills
- 3. Electronic Information
- Communication Tools
- Assignments
- Work Area
- Course Readings**

PersonalCourseCommunicateDocumentAdministrateFEUPHelp

Course Readings

CourseNLS

Number of Readings5

Select readings by subject

All

Select readings by document type

All

Select number of rows to display

5

#	Title	Author	URL	Copies
1	Elementary Differential Equations and Boundary Value Problems, W. E. Boyce,	Boyce, W. E.		
2	Complex variables and applications, James Ward Brown, Ruel V. Churchill,	Churchill, Ruel V. Brown, James Ward		
3	Advanced engineering mathematics, C. Ray Wylie, Louis C. Barret,	Barrett, Louis C. Wylie, C. Ray		
4	Advanced Engineering Mathematics, Erwin Kreyszig,	Kreyszig, Erwin		
5	Curso de análise matemática, J. Santos Guerreiro,	Guerreiro, J. Santos		

1

Created: 2004-09-06

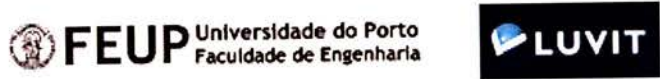
Published by: Administrator Admin

☐ Completed

Fig. 5: Course Readings plugin

Acknowledgements

We acknowledge the support of LUVIT AB to this project.



ANEXO C: Artigo aceite para apresentação no “e-LES’04” – Aveiro, 29 de Outubro de 2004



Conferência eLES'04- eLearning no Ensino Superior

Universidade de Aveiro, 27-29 Out 04

Proposta de comunicação

(submeter através de eLES04@cemed.ua.pt)

Área temática*	Tema	Outros tópicos
	Tópico	Serviços Tecnológicos de Suporte

* Utilizar lista de temas e tópicos da conferência

Título	Framework Architecture Specification for LMS plugins (A Modern Approach)
--------	--

1º autor			
Nome	Manuel Almeida Domingues		
Organização	Faculdade de Engenharia da Universidade do Porto		
Morada	Rua Dr. Roberto Frias 4200-465 Porto		
E-mail	mad@fe.up.pt	Telefone	225081583

Outros autores	
Nome	Organização
Pedro Miguel Faria	FEUP
Rubén Melim Nóbrega	FEUP

Requisitos para a apresentação (logísticos, de acesso, etc, para além de computador e projector vídeo)	

Resumo (máximo 1 página, fonte tamanho 10)
Framework Architecture Specification for LMS Plugins (A Modern Approach) Manuel Almeida Domingues, Pedro Miguel Faria, Rúben Melim Nóbrega, FEUP Abstract An efficient learning organization is a fundamental element in the learning quality. This way, it is important that e-Learning systems LMS (Learning Management Systems) answer to the expectations and people needs and that simultaneously adapt themselves to the evolution of those needs. Actually, for some needs it may already exist market solutions. However there will always be some specific needs that will hardly be covered by the existent offers. In these cases, it is necessary to implement modules for LMS. Most LMS offer mechanisms that make this job easy however these modules are dependent on LMS. Also, they are subordinated from the learning organization information systems in which they are integrated. This article approaches an abstract architecture to build modules for LMS, with the interaction of the Organization Information Systems that will allow sharing among learning organizations with some independency. Specification The architecture specification is based on the materialization of the abstract framework which is a standard for LMS[1] environments. This paper conceptualizes a framework specification which is divided in horizontal layers that interact with the adjacent or more distant levels. The more distant ones, represent orthogonal aspects to the several architecture layers, such as security, operational management and infrastructure/communication. Normally the horizontal layers are presentation, business logic and data access logic. From the point of view of IAF[1] the presentation layer can map to the applications/tools layer. The business logic layer, can be subdivided in two layers accordingly to the same

IAF[1] standard, i.e. the application services layer (this one interacts more directly with the presentation layer) and underneath this one, the common services layer that support the application services. This layer is in conformity with the OKI OSID [2] standard, which more than defining a vast set of common services for LMS platforms also offers a possible implementation. Finally we have the data access layer which accordingly with IAF[1], maps to the infrastructure/communications layer. This layer is responsible for providing a more easy way to the intra/extra system communications, using standards such as XML, WebServices, SOAP.

Architecture Implementation

To assure the quality of this implementation it is advisable the use of good practices such as software patterns[3], test driven development (TDD)[4] and aspect oriented programming (AOP)[6]. Software patterns are proven solutions for recurrent problems. The use of TDD in a wise and controlled way benefits the quality of the developed system. The tests control the development path and assure its conformance. AOP is an approach to the aspects or needs separation that cross-cut the system layers to minimize the dispersion of these same aspects.

References

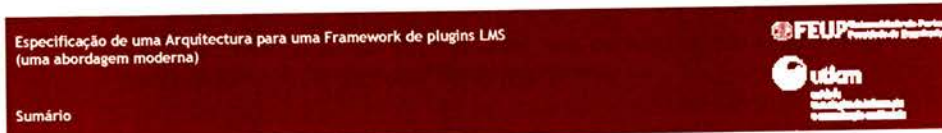
- [1] IMS Global Learning Consortium, inc – IAF IMS Abstract Framework (2004), <http://www.imsglobal.org/af/index.cfm>
- [2] OKI OSID, The Open Knowledge Initiative - Open Service Interface Definitions (2004), <http://web.mit.edu/oki/>
- [3] Microsoft Patterns & Practice (2004), <http://msdn.microsoft.com/architecture/patterns/default.aspx>
- [4] Test Driven Development (2004), <http://www.testdriven.com>
- [5] Aspect-Oriented Software Development (2004), <http://aosd.net/>

ANEXO D: Slides da apresentação do artigo no “e-LES’04” – Aveiro, 29 de Outubro de 2004



Especificação de uma Arquitectura para uma Framework de plugins LMS (uma abordagem moderna)

Manuel Domingues
Pedro Faria
Rúben Nóbrega



- Introdução
 - Contexto/Problema
 - Objectivos
- Actores envolvidos
 - Ferramentas
 - Learning Management Systems (LMS)
 - Sistemas de Informação
- Framework
- Arquitectura Genérica
- Exemplo de implementação
- Conclusões
- Referências

Especificação de uma Arquitectura para uma Framework de plugins LMS
(uma abordagem moderna)

Introdução
Contexto/Problema




- Uma organização de ensino eficiente constitui um elemento fundamental na qualidade de ensino;
- É importante que os sistemas de e-Learning (LMS - Learning Management Systems) respondam às expectativas e necessidades das pessoas e que simultaneamente se adaptem à evolução dessas mesmas necessidades;
- A necessidade de criar ferramentas para adicionar mais funcionalidades às plataformas de e-Learning, com a maior flexibilidade possível, de modo transparente para o utilizador;

Especificação de uma Arquitectura para uma Framework de plugins LMS
(uma abordagem moderna)

Introdução
Objectivos




- Fornecer aos formadores dos LMS's, melhores meios para organizar, aceder e desenvolver uma vasta gama de informação e estratégias alternativas;
- Arquitectura abstracta para construir módulos para LMS, com interacção de Sistemas de Informação nas Organizações, que permitirá a sua partilha entre organizações de ensino com alguma independência;
- Optimizar a especificação de uma API genérica e suficientemente adaptável para os objectivos propostos;

Especificação de uma Arquitectura para uma Framework de plugins LMS
(uma abordagem moderna)



Actores envolvidos

- Ferramentas
- Os Sistemas de e-Learning
- Os Sistemas de Informação nas instituições de ensino

e-learning FEUP

5

Especificação de uma Arquitectura para uma Framework de plugins LMS
(uma abordagem moderna)



Framework

- Propõe uma arquitectura de desenvolvimento de ferramentas para ambientes LMS;
- Dividida por camadas horizontais e verticais, que interagem entre os níveis adjacentes ou mais distantes;
 - As camadas horizontais contêm a apresentação, a lógica de negócio e acesso aos dados;
 - As camadas verticais representam aspectos ortogonais às várias camadas da arquitectura, tais como segurança, gestão operacional e infra-estrutura/comunicações;

e-learning FEUP

6

- O uso de práticas, como padrões de software, desenvolvimento orientado aos testes (TDD) e programação orientada a aspectos (AOP);
 - Os padrões são soluções comprovadas para problemas recorrentes;
 - O desenvolvimento TDD (Test Driven Development) beneficia a qualidade do sistema de desenvolvimento;
 - Os testes controlam o rumo do desenvolvimento e garantem a sua conformidade;
 - A AOP (Aspect Oriented Programming) é uma abordagem para o isolamento de aspectos ou necessidades que atravessam de forma transversal as camadas do sistema de forma a minimizar a dispersão desses mesmos aspectos;

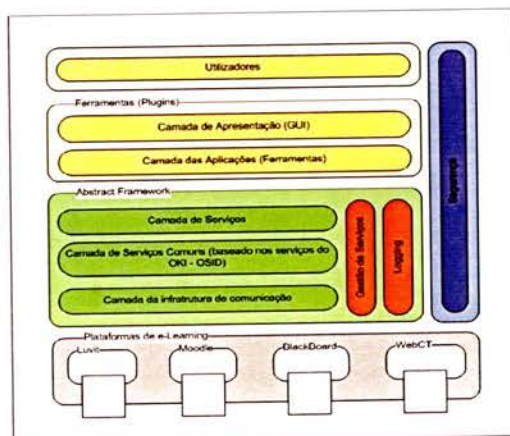


Fig. 1 - Diagrama da Arquitectura Genérica

Especificação de uma Arquitectura para uma Framework de plugins LMS
(uma abordagem moderna)

FEUP

FEUP

FEUP

Arquitectura Genérica

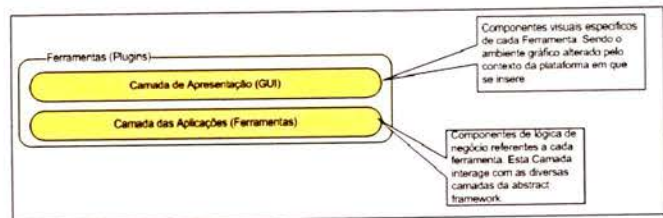


Fig. 2 - Diagrama da Camada Superior (Ferramentas)

e-learning FEUP

9

Especificação de uma Arquitectura para uma Framework de plugins LMS
(uma abordagem moderna)

FEUP

FEUP

FEUP

Arquitectura Genérica

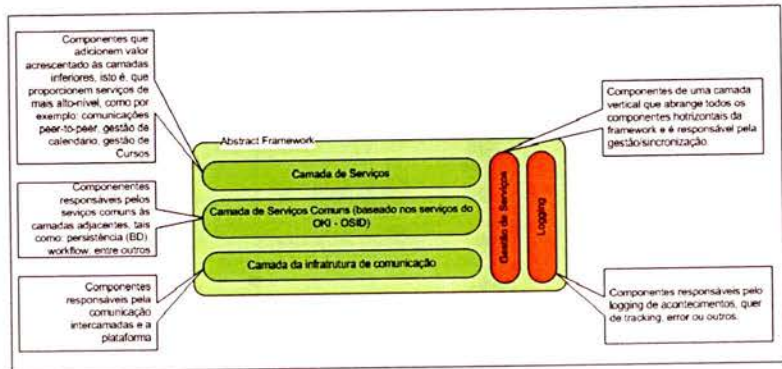


Fig. 3 - Diagrama da Camada Intermédia (Framework)

e-learning FEUP

10

Especificação de uma Arquitectura para uma Framework de plugins LMS
(uma abordagem moderna)

FEUP
Faculdade de Engenharia da Universidade do Porto

FEUP
Faculdade de Engenharia da Universidade do Porto

FEUP
Faculdade de Engenharia da Universidade do Porto

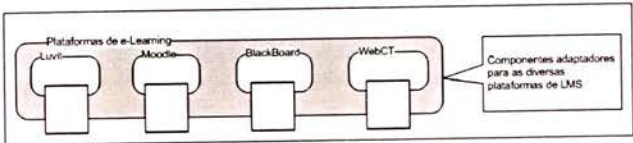


Fig. 4 - Diagrama da Camada Inferior (Adaptadores)

e-learning
FEUP

11

Especificação de uma Arquitectura para uma Framework de plugins LMS
(uma abordagem moderna)

FEUP
Faculdade de Engenharia da Universidade do Porto

FEUP
Faculdade de Engenharia da Universidade do Porto

FEUP
Faculdade de Engenharia da Universidade do Porto

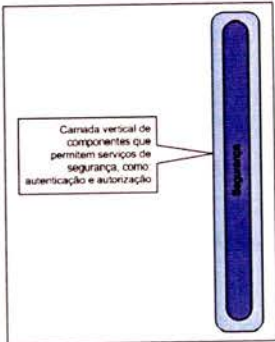


Fig. 5 - Diagrama da Camada Vertical (aspectos)

e-learning
FEUP

12

Especificação de uma Arquitectura para uma Framework de plugins LMS
(uma abordagem moderna)

Exemplo de implementação

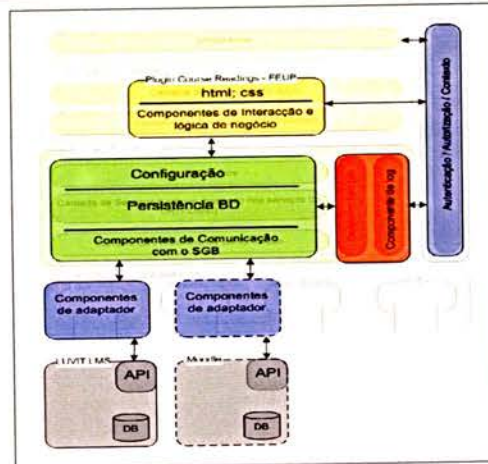


Fig. 6 - Exemplo de plugin Course Readings

Especificação de uma Arquitectura para uma Framework de plugins LMS
(uma abordagem moderna)

Conclusões

- A Framework oferece uma maior facilidade de utilização, maior abertura e segurança;
- Vantagens:
 - Modularidade
 - Encapsulamento de implementação flexíveis através de uma interface estável;
 - Reutilização
 - Define componentes genéricos que podem ser reaplicáveis;
 - Extensibilidade
 - Fornece métodos explícitos que possibilitam às aplicações estenderem as suas interfaces;

Especificação de uma Arquitectura para uma Framework de plugins LMS
(uma abordagem moderna)

Conclusões




▪ Desafios:

- Dificuldade para se manter a compatibilidade com versões anteriores, já que as frameworks tornam-se mais *maduras* com o passar do tempo e as aplicações devem evoluir em paralelo;
- Pouca padronização das API das plataformas de LMS;
- Implementação numa plataforma de programação específica. Esta característica limita o uso da framework, visto que as linguagens de programação orientadas a objecto nem sempre podem ser combinadas, não sendo interessante trabalhar com outra linguagem que não a da construção da framework;

Especificação de uma Arquitectura para uma Framework de plugins LMS
(uma abordagem moderna)

Referências




- IMS Global Learning Consortium, inc - IAF IMS Abstract Framework (2004), <http://www.imsglobal.org/af/index.cfm>
- OKI OSID, The Open Knowledge Initiative - Open Service Interface Definitions (2004), <http://www.okiproject.org>
- Microsoft Patterns & Practice (2004), <http://msdn.microsoft.com/architecture/patterns/default.aspx>
- Test Driven Development (2004), <http://www.testdriven.com>
- Aspect-Oriented Software Development (2004), <http://aosd.net/>

Tecnologias de Informação na Educação
Information Technologies in Education



Especificação de uma Arquitectura para uma Framework de plugins LMS (uma abordagem moderna)

Manuel Domingues (mad@fe.up.pt)
Pedro Faria (pfaria@fe.up.pt)
Rúben Nóbrega (ruben.nobrega@fe.up.pt)





FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

BIBLIOTECA



0000081500