



## **Plataformas de e-Learning Orientadas a Serviços**

**Paulo Hélio dos Santos Alves**

Dissertação apresentado à  
**Escola Superior de Tecnologia e Gestão**  
**Instituto Politécnico de Bragança**  
para obtenção do grau de **Mestre em**  
**Sistemas de Informação**

Orientador:

**Paulo Alexandre Vara Alves**

Esta dissertação não inclui as críticas e sugestões feitas pelo Júri

**Outubro de 2010**

Á Nina

Ao Pai, à Mãe e à Mana

# Agradecimentos

Agradeço a todos os que tornaram possível a realização deste trabalho, nomeadamente:

Ao meu orientador Professor Doutor Paulo Alves, pela ajuda e incentivo que me foi dando ao longo do tempo.

A toda a minha família, em especial aos meus pais por todo o apoio incondicional que me prestaram.

À Anabela Martins, pelo apoio, entrega e paciência que teve para comigo para a realização deste trabalho, o meu muito obrigado.

# Resumo

Na era da economia e das sociedades baseadas no conhecimento, o acesso à informação é um dos factores de maior relevância no bom funcionamento das organizações. Para as organizações estarem preparados para estes desafios, implica uma mudança para um paradigma de formação ao longo da vida e o recurso às tecnologias da informação (TI), de modo a tornar as organizações mais eficientes, evitando o desperdício de recursos por vezes vitais numa organização. O e-learning assume-se como a solução para a formação sem limitações de espaço e de tempo.

Com o crescimento das TI no seio das organizações leva a uma necessidade de adaptação constante das aplicações às estratégias organizacionais. Assim, o desenvolvimento de software assume um papel relevante na missão das organizações, em que qualquer aplicação numa empresa deverá ser um objecto de trabalho catalisador.

Foi com base na necessidade de flexibilidade das arquitecturas de software que surgiram as arquitecturas orientadas a serviços, prevendo o crescimento organizacional de modo a tornar possível a transformação de aplicações rígidas em componentes de software. A flexibilidade e a interoperabilidade entre os diversos componentes e tecnologias permitem uma maior diversidade de serviços para disponibilização final.

Neste trabalho pretende-se efectuar um estudo das potencialidades das arquitecturas SOA aplicadas ao e-learning, de modo a aumentar a competitividade das empresas e das organizações.

Neste sentido é proposto um modelo de arquitectura de software para plataformas de e-learning orientadas a serviços, com o objectivo de tornar os paradigmas de negócio empresariais mais flexíveis e adaptados à necessidade de constante mutação das organizações.

**Palavras Chave:** e-learning, SOA, organizações, flexibilidade

# Abstract

Now days, in times of economic society and knowledge, the access to information are a major factor for the good function of organizations. To be prepared for that challenge any organization must change their paradigm to lifelong learning. Supported by new Information Technologies (IT), it is possible to increase the efficiency and avoid waist resources that could be vital in the organization. E-learning assumes such a solution for the training areas, without limits for space and time.

With growing of IT adoption in organizations, it's necessary to be prepared for continues adaptations of business strategies. For that, the software development assumes important role on the mission of organizations, such all applications must be a catalyst object for work development in organization.

It was based on flexibility that in the area of software architectures it appears the service oriented architectures, having the main goal to respond to organizations growing, making possible to transform rigid applications in software components.

Flexibility and interoperability with many components and technologic allow a bigger set of final service available.

In this dissertation, we pretend to study the potential of SOA architectures applicable to e-learning, to sustain the growing of applications in organizations.

For that, it's propose an architecture model for e-learning platforms with service oriented approach, with the objective to support the flexibility of business organization paradigms and to be easily adapted to new organization requirements.

**Keywords:** e-learning, SOA, organization, flexibility

# Índice

<b>1</b>	<b>Introdução .....</b>	<b>11</b>
1.1	Motivação .....	11
1.2	Objecto de estudo.....	12
1.3	Objectivos .....	13
1.4	Organização da dissertação.....	14
<b>2</b>	<b>Arquitecturas SOA em ambientes E-learning .....</b>	<b>15</b>
2.1	E-learning.....	15
2.1.1	LMS – Learning Management System.....	17
2.1.2	Normas para conteúdos de e-learning .....	20
2.1.2.1	SCORM.....	20
2.1.2.2	AICC .....	24
2.2	Arquitectura SOA .....	25
2.2.1	Introdução às Arquitecturas de Software .....	25
2.2.1.1	Motivação da Arquitectura de Software.....	26
2.2.1.2	Elementos arquitecturais .....	26
2.2.2	Definição e Contextualização.....	27
2.2.3	Arquitectura da informação.....	29
2.2.4	Arquitectura das camadas de referência da aplicação SOA.....	30
2.2.5	O Futuro, Arquitectura Orientada à Web (WOA)? .....	31
<b>3</b>	<b>Especificação de uma plataforma de e-learning orientada a serviços .....</b>	<b>34</b>
3.1	Requisitos.....	35
3.1.1	Disponibilidade de acesso à plataforma .....	35

3.1.2	Login na plataforma .....	35
3.1.3	Página Inicial.....	35
3.1.4	Inscrição num curso .....	35
3.1.5	Página de cursos matriculados .....	36
3.1.6	Frequência de curso.....	36
3.1.7	Performance .....	36
3.2	Casos de Uso.....	36
3.2.1	Caso de uso login na plataforma .....	36
3.2.2	Caso de uso frequência de curso .....	37
3.2.3	Caso de uso performance .....	38
3.3	Diagrama de Actividades.....	39
3.4	Diagrama de UML .....	41
3.5	Modelo E-R.....	42
3.5.1	Modelo E-R de Gestão de Utilizadores.....	43
3.5.2	Modelo E-R de Gestão de conteúdos .....	45
3.5.3	Diagrama de Classes de Serviços.....	46
3.5.4	Diagrama de Sequência dos Web Services .....	48
<b>4</b>	<b>Desenvolvimento da Plataforma SKyLearn.....</b>	<b>50</b>
4.1	Tecnologias adoptadas .....	50
4.2	Desenvolvimento de Interface Gráfico. ....	51
4.3	Desenvolvimento da camada de <i>middleware</i> .....	54
4.4	Implementação da interface .....	56
<b>5</b>	<b>Validação da Plataforma SKyLearn.....</b>	<b>59</b>
5.1	Metodologia de investigação .....	60
5.2	Análise do inquérito .....	61
5.3	Síntese dos resultados .....	66

<b>6 Conclusão .....</b>	<b>67</b>
6.1 Trabalho Futuro .....	68
6.2 Conclusão.....	69
<b>Questionário.....</b>	<b>72</b>
II – Plataforma de e-learning SKyLearn .....	75



# Lista de Tabelas

Tabela 1 - Discriminativo do modelo de dados de gestão do LMS.....	44
Tabela 2 - Discriminativo do modelo de dados de gestão de conteúdos .....	46

# Lista de Figuras

Figura 1 - Plataformas de E-learning em Portugal, Ano 2007 .....	19
Figura 2 - Custo de produção de conteúdo e-learning .....	21
Figura 3 - Integração de SCORM com varios LMS.....	21
Figura 4 - SCORM Run-Time Enviroment.....	23
Figura 5 - Integração de conteúdo AICC com LMS .....	25
Figura 6-Business Process Reengineering Cycle .....	28
Figura 7 - Arquitectura tradicional de aplicações ' <i>stand-alone</i> ' .....	29
Figura 8 - Arquitectura da informação de referência .....	30
Figura 9 - Arquitectura dos componentes de referência.....	31
Figura 10 - Visão da arquitetura orientada a Web. ....	32
Figura 11 - Evolução de SOA para WOA.....	32
Figura 16 - Diagrama de Actividades do Modelo de Negocio SCORM .....	40
Figura 17 - Diagrama UML de Arquitectura.....	41
Figura 18- Modelo E-R, Gestão de Utilizadores.....	43
Figura 19 - Modelo E-R da Gestão de Conteúdos.....	45
Figura 20 - Diagrama de Classes do serviço de Users .....	47
Figura 21 - Ciclo de vida de um Webservice .....	48
Figura 22 - Ciclo de vida de um pedido a um Webservice.....	49
Figura 23 - Homepage de apresentação da plataforma SkyLearn .....	51
Figura 24 - Layout da Homepage da plataforma SkyLearn .....	52
Figura 25- Homepage após login na plataforma SKyLearn.....	56
Figura 26 - Área de Cursos na plataforma SKyLearn .....	57
Figura 27 - Etapas da Metodologia de Investigação [Alves, 2007].....	61
Figura 28 - Gráfico de Características de uma plataforma de e-learning.....	63
Figura 29 - Gráfico do tipo de preferência para suporte e estrutura dos conteúdos de e-learning .....	64
Figura 30 - Classificação da plataforma SkyLearn .....	65

# Capítulo 1

## 1 Introdução

### 1.1 Motivação

A cada dia que passa, o ser humano torna-se cada vez mais dependente das Tecnologias e Informação e Comunicação (TIC) para a realização das mais variadas tarefas do dia-a-dia.

Numa altura em que o público-alvo de todas as áreas de negócio é cada vez mais exigente e com perfis mais segmentados, os tempos de disponibilização e excesso de informação tratada pelos métodos tradicionais aumenta de forma incontrolável. Deste modo, os tempos de espera por resultados e análises que os gestores necessitam, devem ser cada vez menores, para facilitar a tomada de decisões em tempo útil.

A eficiência de uma organização reflecte-se não só na qualidade dos seus produtos e/ou serviços como também no *know-how* e empenho dos seus colaboradores em vencer na concretização dos objectivos com sucesso.

As deficiências de informação e formação dentro de uma organização, obrigam a um reconhecimento das necessidades formativas de todos os seus colaboradores nas suas valências.

As plataformas de e-learning, surgem como um meio de suporte à formação dos colaboradores de uma organização para que os gestores possam ter sob a sua responsabilidade colaboradores com *know-how* suficiente para darem resposta às necessidades do mercado de forma ágil e capaz.

## 1.2 Objecto de estudo

Nos dias de hoje, com a competitividade que existe no mercado, não basta ser bom, é preciso ser o melhor, e para isso a tomada de decisões tendo por base informação actualizada é o trunfo das organizações para serem líderes.

Mais importante do que ter acesso à informação, é ter a capacidade de manuseá-la para que seja possível a tomada de decisão, com a “Informação certa, na hora certa, para a pessoa certa”.

A internet permite-nos através das novas tecnologias, ter acesso a um vasto conjunto de informação e recursos de comunicação interpessoal. O E-learning sendo um sinónimo de um processo de ensino-aprendizagem on-line, surge como uma tecnologia que faculta a interacção não só com conteúdos informáticos, como também com pessoas.

As implementações de um sistema de *E-learning* para suporte à formação parte da necessidade de as organizações terem colaboradores preparados para os desafios constantes de um mercado global altamente competitivo.

Este trabalho tem como principal objectivo apresentar uma arquitectura de software para plataformas de e-learning orientadas a serviços, indo de encontro às necessidades do mundo empresarial, no que diz respeito à formação dos seus quadros tendo em vista a sua preparação continua para enfrentar os desafios de um mercado cada vez mais competitivo.

A plataforma resultante desta arquitectura foi desenvolvida em linguagem JAVA e assente em uma arquitectura orientada a serviços (SOA). Esta plataforma de *E-learning*, disponibiliza aos seus utilizadores a possibilidade de acederem a diferentes conteúdos no formato SCORM, sem terem que ausentar-se das suas organizações, para receberem formação, permitindo-lhes assim, que desenvolvam novas soluções, com custos mais atenuados.

As arquitecturas SOA aliadas aos sistemas de *E-learning*, trazem assim um valor acrescentado para uma organização, na medida em que, este pode ser embebido dentro do sistema de informação da empresa, como um serviço que interage com todos os serviços/departamentos da organização.

### 1.3 Objectivos

O principal objectivo deste trabalho centra-se na especificação de um modelo para plataformas de e-learning inter-operáveis, baseadas no referencial SCORM e em arquitecturas orientadas a serviços, que permitam potenciar a formação e a aprendizagem ao longo da vida em contexto laboral.

Para a concretização deste objectivo geral foram definidos os seguintes objectivos específicos:

- Especificação de uma arquitectura para plataformas de e-learning baseadas no referencial SCORM;
- Definição de um modelo de interoperabilidade entre sistemas de gestão da aprendizagem e os conteúdos baseado numa arquitectura orientadas a serviços;
- Implementação de uma plataforma baseada na referência SCORM para dar suporte à necessidade da aprendizagem ao longo da vida.

Na especificação de uma arquitectura de e-learning baseada no referencial SCORM, pretende-se definir e especificar os requisitos básicos de uma plataforma de e-learning com capacidade para reproduzir conteúdos elaborados segundo o referencial SCORM.

Tendo em conta o ciclo de vida de um conteúdo, pretende-se que a plataforma execute todos os pedidos efectuados pelo conteúdo á plataforma, quer sejam pedidos de leitura como de escrita de dados.

Na definição do modelo de interoperabilidade entre sistemas, pretende-se demonstrar, recorrendo a arquitectura orientada a serviços, a usabilidade de serviços, para tal recorre-se á implementação de um protótipo onde todas as acções efectuadas pelo conteúdo SCORM, são efectuadas recorrendo a serviços.

## 1.4 Organização da dissertação

Esta dissertação foi elaborada no âmbito do Mestrado em Sistemas de Informação, leccionado pela Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Bragança, e divide-se em cinco capítulos.

No primeiro capítulo, introdução, é abordada a motivação, o objecto de estudo, os objectivos e a estrutura da dissertação.

No segundo capítulo deu-se ênfase ao estudo do estado da arte das arquitecturas orientadas a serviços (SOA), assim como do e-learning. Começando por definir o conceito de e-learning, e explicitando as normas de conteúdos SCORM e AICC. Este capítulo apresenta também as arquitecturas SOA, apresentando uma introdução às arquitecturas de software, seguida da motivação da mesma, passando por definir os elementos arquitecturais. Outros tópicos abordados neste capítulo incluem a definição e contextualização, a arquitectura da informação, e, a arquitectura das camadas de referência da aplicação SOA. Foi também feita uma breve referência ao que será o futuro de uma arquitectura orientada à Web.

O terceiro capítulo, refere-se a metodologia de desenvolvimento e às bases da arquitectura que sustentam o protótipo SKyLearn. Começamos por definir os requisitos iniciais que a aplicação irá responder, em seguida apresentamos os casos de uso dos requisitos preponderantes para esta dissertação, apresentando os modelos UML e E-R.

O quarto capítulo é descrito o processo de desenvolvimento do protótipo SKyLearn, nomeadamente as opções técnicas para implementação de cada requisito e o modo de funcionamento do protótipo.

No quinto capítulo é apresentada a validação do protótipo em função da questão em torno da problemática desta tese, da metodologia seguida para validar a aplicação, enquadramento, análise dos resultados e conclusões.

No sexto capítulo apresentamos as conclusões finais desta dissertação, com a discussão dos resultados e apresentação de trabalhos futuros.

## Capítulo 2

# 2 Arquitecturas SOA em ambientes E-learning

## 2.1 E-learning

Podendo ter diferentes significados, E-learning é um termo que pode ser entendido como um fragmento de conteúdos, ou uma infra-estrutura técnica. Ainda há quem o caracterize como uma forma de auto-estudo on-line, ou uma aprendizagem e colaboração em tempo real e ao ritmo e disponibilidade de cada um.

Pelo facto de ser uma mais-valia para as organizações, pelo valor acrescentado que pode representar, o E-learning é um método eficaz de aprendizagem, e de elevada importância estratégica para as organizações, baseado em tecnologias da Internet, de forma a facultar um abrangente conjunto de soluções que reforcem tanto o conhecimento como o empenho dos colaboradores.

Rosset[Rossett, 2001] define e-learning como “formação disponível em um servidor que está acessível a todos por internet”.

E-learning identifica-se como um modelo de ensino à distância, que tem como suporte as tecnologias de informação (TI). A sustentabilidade deste modelo provem de um conjunto de soluções que levam a uma melhoria dos conhecimentos e subseqüentemente ao aumento do produtividade[Rosenberg, 2001].

O modelo de ensino tradicional, tendo como ponto forte a interacção presencial entre o tutor e os seus formandos apresenta algumas desvantagens, das quais se destaca:

- Limitação do espaço físico, uma vez que tutor e aluno deverão estar em contacto no mesmo espaço;
- Restrição de horários;
- Limitação da capacidade de ajuizar sobre o conteúdo de aprendizagem por parte do tutor perante o formando ou aluno.

Em contrapartida o modelo de e-learning, baseando em TI, e no lema “*anytime and anywhere*”, oferece como principais vantagens<sup>1</sup>:

- Interação do aluno com os conteúdos pode ou não estar a ser monitorizado por um tutor;
- Anytime – A qualquer hora o formando poderá aceder aos conteúdos, estando ou não o tutor presente;
- Anywhere - Não tem restrições de espaço, pois são conteúdos virtuais, e que estão disponíveis em qualquer computador que o formando possa usar;
- Formação individualizada, em que o aluno acede a um conteúdo, onde poderá efectuar a aprendizagem ao seu próprio ritmo;
- Normalização da produção de conteúdos, sendo reproduzidos em qualquer LMS compatível com as normas pela qual os conteúdos foram produzidos, reduzindo assim os custos de produção.

Estas são algumas das razões pelas quais um sistema de e-learning fornece soluções flexíveis e de acordo com o tempo e com a metodologia de ensino/formação de cada organização.

As características diferenciadoras dos modos de formação de e-learning (síncrono ou assíncrono) estão relacionadas com o tipo de conteúdos disponível para cada um dos cursos.

Conteúdos que dependam de outros, seja de tutoria como de *inputs* externos, que obriguem a estarem todos os envolvidos em contacto para poderem frequentar a formação, designa-se de formação síncrona, também por vezes chamada de *blended-learning*. [Bielawski e Metcalf, 2003]

Quando um conteúdo é atómico e totalmente independente de sistemas externos, o que permite ao formando executa-lo e a apreende-lo em qualquer altura, designa-se de formação

---

<sup>1</sup> Fonte: [http://www.e-learningguru.com/articles/art1\\_3.htm](http://www.e-learningguru.com/articles/art1_3.htm)



assíncrona, uma vez que, não sendo necessária a intervenção de outros, o formando é auto-suficiente para interagir com o conteúdo e apreender.

No e-learning o formando passa a ter um papel fundamental, e a aprendizagem tem predominância sobre o ensino, permitindo mais facilmente o desenvolvimento das competências, tendo como base a capacidade do formando de aprender sem restrições de espaço nem de tempo.

Os sistemas de aprendizagem depararam-se com problemas de compatibilidade entre os sistemas utilizados para visualizar os conteúdos, afectando assim os custos de formação.

Numa perspectiva de uniformização de conteúdos, diversas organizações, como a AICC, (Aviation Industry Computer-Based Training Committee) começaram a estudar formas de otimizar e normalizar a produção de conteúdos, para assim poderem ser reutilizados entre diversas organizações [Lucena, 2003]

Para além da AICC, mais organizações como o Departamento de Defesa dos Estados Unidos, num esforço de normalizar, produziram a norma SCORM (Sharable Content Object Reference Model), tendo sido adoptado por uma grande maioria de organizações como o *standard* a seguir, visto que contemplava a interoperabilidade entre o formando e o conteúdo formativo<sup>2</sup>.

Existem outras normas referenciais que tem como objectivo a interacção o formando e os conteúdos, nomeadamente as aprovadas pelo IMS, Global Learning Consortium, Inc.

### **2.1.1 LMS – Learning Management System**

Existem plataformas de e-learning que permitem não só efectuar a disponibilização de conteúdos, como também a gestão de cursos e utilizadores, monitorização da aprendizagem e colaboração online entre utilizadores., chamadas de LMS - Learning Management Systems.

Algumas dessas plataformas, já têm módulos de monitorização avançada que permitem a obtenção de relatórios pormenorizados sobre as actividades no sistema, tais como a monitorização do percurso dos formando em cada curso, o tipo de intervenção e colaboração, os resultados obtidos, entre outros.

---

<sup>2</sup> Fonte: <http://www.scorm.com/scorm-explained/business-of-scorm/>

Das mais populares destacamos algumas, tais como o Moodle<sup>3</sup>, Sakai<sup>4</sup>, e BlackBoard<sup>5</sup>.

O Sakai é uma plataforma de colaboração livre, distribuída e desenvolvida em ambiente *open-source*, que permite um ambiente de aprendizagem colaborativo, através de um conjunto de ferramentas de e-learning que inclui portefólios, cursos online e projectos.<sup>6</sup>

Por sua vez o Moodle, é uma das plataformas de e-learning mais populares, sendo também gratuita, foi também desenvolvida em regime de *open-source*, em linguagem PHP, permitindo a gestão e a criação de ambientes de aprendizagem colaborativos[Rice, 2006]

Um curso na plataforma Moodle apresenta as seguintes vantagens:

- Motivação dos alunos
- Produção de conteúdos
- Partilha de conteúdos
- Gestão do ambiente de aprendizagem (Cursos, Alunos, etc)
- Avaliações dos alunos
- Controlo de acessos

A Blackboard é uma plataforma comercial, com participação da Microsoft, tendo grande penetração no ambiente académico. Actualmente está em declínio devido aos valores de licenciamento e ao amadurecimento das soluções *open-source*, como é o caso do Moodle.

Em 2007 foi efectuado em Portugal um estudo intitulado de “Estudo das Plataformas de Formação à Distância em Portugal”<sup>7</sup>, onde se pretendia perceber a realidade em Portugal da utilização de plataformas LMS. Os resultados deste estudo indicam que a plataforma Moodle é a mais utilizada com uma grande vantagem em relação às plataformas comerciais ( Figura 1).

---

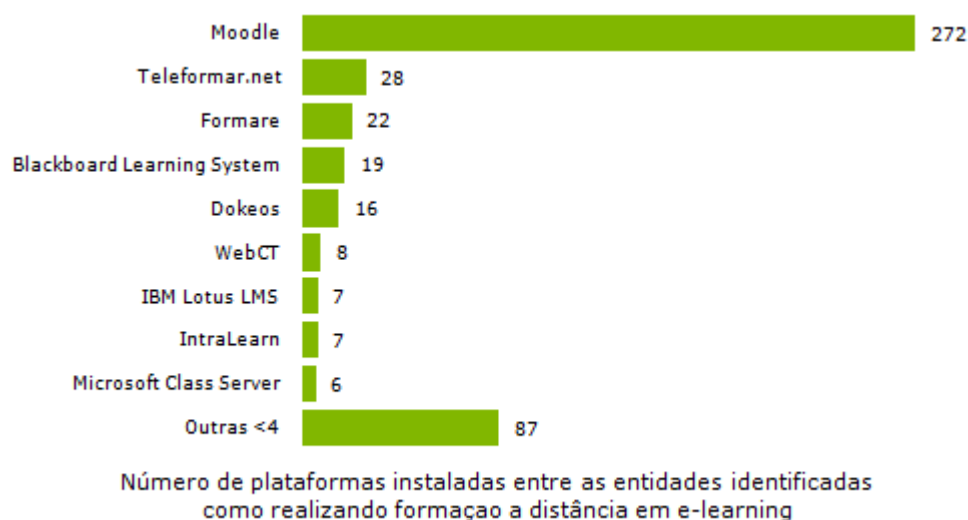
<sup>3</sup> <http://moodle.org>

<sup>4</sup> <http://sakaiproject.org>

<sup>5</sup> <http://www.blackboard.com>

<sup>6</sup> Fonte: <http://confluence.sakaiproject.org/display/CONF/Welcome+to+the+Sakai+wiki>

<sup>7</sup> Fonte: [http://www.elearning-pt.com/lms2/index.php?option=com\\_wrapper&Itemid=34](http://www.elearning-pt.com/lms2/index.php?option=com_wrapper&Itemid=34)



**Figura 1 - Plataformas de E-learning em Portugal, Ano 2007<sup>8</sup>**

O Moodle é uma plataforma bastante popular devido à sua facilidade de instalação, escalabilidade, ao seu suporte das normas de e-learning, tais como SCORM, AICC, entre outras, acrescentando o facto de ser uma plataforma *open-source* o que permite uma utilização sem grandes custos acrescidos.

Todas as plataformas têm as suas mais-valias, dependendo do âmbito de utilização, o que faz com que seja um sector com bastante possibilidade de escolha. Contudo em nenhuma das plataformas é disponibilizado um conjunto de serviços de integração em outras plataformas fora do âmbito de e-learning, como por exemplo a possibilidade de uma organização poder adicionar um módulo de e-learning na sua intranet.

É no conceito de distribuir o e-learning, possibilitando a integração da framework SCORM e da gestão de aprendizagem em qualquer contexto aplicacional, que as organizações passaram a encarar a formação em contexto de trabalho, sendo considerado o e-learning como mais um módulo da organização a par das restantes componentes de software.

---

<sup>8</sup> Fonte: [http://www.ed-rom.com/?pt=moodle\\_blackboard](http://www.ed-rom.com/?pt=moodle_blackboard)

## 2.1.2 Normas para conteúdos de e-learning

Um conteúdo de e-learning para ser inserido num contexto, e reproduzido, necessita de um sistema de LMS, (Learning Management System). Este sistema deverá operacionalizar a produção do contexto físico de um curso, turmas, ou conjuntos de formandos, para que assim se possa disponibilizar os conteúdos de uma forma lógica e com um objectivo de aprendizagem [Rozanski e Woods, 2005].

Assim um LMS não é mais do que uma aplicação, dotada de capacidade para agendar, gerir, disponibilizar e acompanhar o decorrer dos cursos de e-learning, disponibilizando informações constantes e reais em consonância com o resultado da aprendizagem do formando.

Apesar dos avanços tecnológicos, a maioria dos LMS ainda não tem suporte para o referencial SCORM(Sharable Content Object Reference Model), utilizando assim ferramentas internas para a autoria, fechando os conteúdos a determinado LMS.

Apesar de conseguirem interagir com algumas normas de e-learning, os LMS não conseguem garantir uma interacção completa com um conteúdo produzido numa ferramenta externa ao LMS. Este factor representa uma das áreas com maior relevo nas áreas do e-learning no sentido de se conseguir uma interoperabilidade plena entre as várias plataformas de e.learning e as ferramentas de autoria.

As principais normas que suportam a portabilidade de conteúdos e a interoperabilidade entre LMS são apresentadas de seguida.

### 2.1.2.1 SCORM

Sharable Content Object Reference Model (SCORM), é uma especificação *standard* para conteúdos de e-learning. A norma SCORM teve a sua origem nos Estados Unidos da América através do Departamento de Defesa, quando em 1997 se propuseram a criar um standard para que os conteúdos para sistemas de educação à distancia se mantivessem uniformes, permitindo padronizar a disseminação de conteúdos de e-learning e que simultaneamente que fosse possível comunicar com qualquer sistema de gestão da aprendizagem, LMS.

Em Janeiro de 2000, a ADL(Advanced Distributed Learning), consórcio criado pelo governo dos Estados Unidos da América e algumas organizações privadas, apresenta o SCORM 1.0, como produto de estudo e fusão de diversas normas ao nível da produção de conteúdos.



Figura 2 - Custo de produção de conteúdo e-learning<sup>9</sup>

Actualmente a versão em uso é o [SCORM 2004 4th Edition](#)<sup>10</sup>, lançada em Março de 2009.

Em termos de funcionamento o SCORM, é HTTP-based, uma vez que é focalizado em ambientes Web, e é baseado em funções de Javascript, que permitem efectuar a comunicação entre o conteúdo e o LMS.

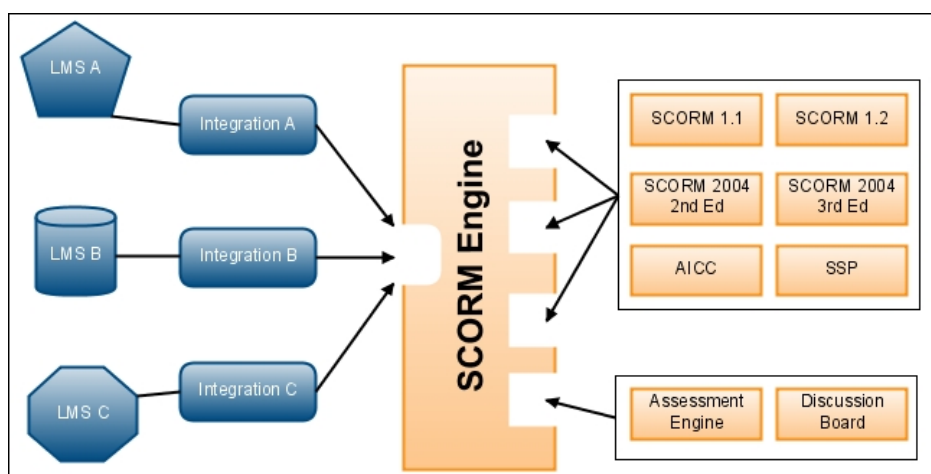


Figura 3 - Integração de SCORM com varios LMS<sup>11</sup>

<sup>9</sup> Fonte: <http://www.scorm.com/>

<sup>10</sup> Fonte: <http://scorm.com/scorm-explained/business-of-scorm/scorm-versions/>

<sup>11</sup> Fonte: <http://www.scorm.com/>

O referencial SCORM pode ser dividido em 3 modelos :

### **CAM: Content Aggregation Model**

O modelo CAM define como os componentes institucionais estão descritos e organizados “*packages*”. Um “*content package*” inclui todos os ficheiros físicos e um ficheiro XML descritor do conteúdo e da sua estrutura (*imsmanifest.xml*). O ficheiro descritor é composto pelas seguintes secções:

- *Metadados*: onde se descreve o “*package*” como um todo.
- *Organizations*: onde se descreve a estrutura do “*package*”. É possível várias organizações, onde cada uma definirá uma hierarquia de itens para ela.
- *Resources*: definições dos recursos de aprendizagem associados, normalmente são os conteúdos de aprendizagem.

A norma SCORM define dois tipos de recursos, os Assets, e os SCOs,( Sharable Content Objects). Um Asset é um conteúdo de aprendizagem básico, como por exemplo uma página HTML, um documento PDF ou um vídeo. Não utiliza a comunicação com o LMS e este pressuposto como completo no momento da sua inicialização. Em contraste, um SCO pode ser composto por Assets, contudo deverá comunicar com o sistema LMS em *run-time* e não deverá ter ligações directas para outros SCOs para não comprometer assim a integridade de informações do seu *Package* Este conceito vai de encontro à definição de objecto de aprendizagem que deve ser um unidade granular reutilizável em qualquer contexto de aprendizagem.

### **RTE: Run-Time Environment**

Uma das aplicações mais interessantes que o SCORM tem é a Run-Time Environment, onde é especificado como o SCO é “lançado”, e como irá comunicar com o LMS.

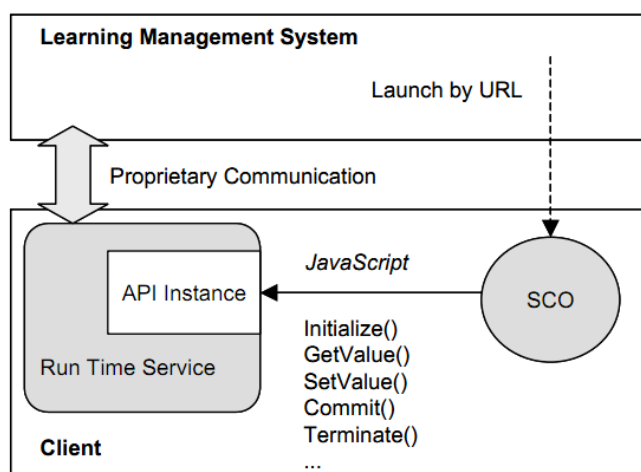


Figura 4 - SCORM Run-Time Environment<sup>12</sup>

O LMS utiliza o URL definido pelo SCO, no ficheiro `imsmanifest.xml`, para poder disponibilizá-lo via HTTP num browser. Cada SCO deverá poder ser lançado desta forma. Na disponibilização num browser, esta deverá ser efectuada numa janela criada a partir da aplicação base ou num elemento frame, para que o SCO através dos algoritmos de procura, possa encontrar a API SCORM e com ela efectuar a comunicação com o LMS.

Cada comunicação é inicializada com o método `Initialize()` e terminada com o `Terminate()`, antes de ser removido da janela do browser. Entre estas duas chamadas é possível efectuar a leitura e escrita de atributos SCORM através dos métodos `GetValue` e `SetValue`. A gestão do fluxo de informação é efectuada pelo LMS, assim o SCO não poderá saber se os seus dados estão em alguma cache, ou se já foram guardados de alguma forma em algum sistema. Em todo o caso é possível em qualquer altura efectuar um pedido para a persistência desses dados, através do método `Commit()`. Existem ainda funções de erros, onde se poderá efectuar o tratamento de diversos erros que poderão ocorrer no conteúdo ao longo do seu ciclo de vida.

As interacções do SCO com o LMS são guardadas num modelo de dados especificado pelo IEEE. Os elementos do modelos de dados são de uso opcional por parte do SCO, contudo o LMS deverá poder proporcionar o seu uso. Este elementos servem para diferente propósitos, efectuar o registo do percursos em um determinado curso, registar os resultados e os seus objectivos. Os elementos do modelo de dados,

<sup>12</sup> Fonte: <http://www.scorm.com/>

permitem-nos efectuar uma interrupção do conteúdo em qualquer altura, guardando os valores que nesse momento o SCO gerou, proporcionando assim a possibilidade de poder ser retomada a aprendizagem em qualquer altura posterior.

### **SN: Sequencing and Navigation**

A versão SCORM2004 introduziu o conceito de sequência e navegação, aplicando as especificações da norma IMS - Simple Sequencing Specification, onde se define que o conteúdo deverá ser sequenciado através da inicialização do formando ou através de eventos do sistema de navegação. Para que a sequenciação seja possível, um *content package* deverá ser transformado numa árvore de actividades para o formando, onde cada ramo contém conteúdos de aprendizagem associados. A progressão entre as diversas actividades de aprendizagem poderá ser sequencial, não sequencial, directo ou adaptativo, seguindo as regras de sequenciação definido numa árvore de actividades.

#### **2.1.2.2 AICC**

A norma AICC foi uma das pioneiras em conteúdos e-learning. Num esforço de normalização dos conteúdos de formação, numa época em que o e-learning era mais conhecido como *Computer-Based-Training(CBT)* e *Computer-Management Instrution(CMI)*, em Fevereiro de 1998 a Aviation Industry CBT Committee (AICC), apresentou uma norma de produção de conteúdos, para poder certificar uma uniformização da formação dentro da sua organização. Um dos exemplos dessa normalização foi a empresa Boeing, fabricante de aeronaves, que para poder certificar as peças que comprava teve que implementar um sistema de formação com conteúdos baseados na norma AICC, para que fosse possível a certificação de que todos os seus fornecedores teriam acesso à mesma informação.

O SCORM é baseado no AICC, onde inicialmente se tratava da comunicação entre o ficheiro do conteúdo e o LMS. Mais tarde, evoluindo para um suporte Web, a norma passou a incluir a comunicação via HTTP entre o conteúdo e o LMS, (*HTTP-based AICC/CMI Protocol –* também conhecida por *HACP*), foi ainda evoluída uma vez mais, para suportar javascript, *EMCAScript-based*, numa tentativa de aproximação ao SCORM.

Contudo devido à sua especificação HACP, o AICC é uma norma muito utilizada ainda nos dias correntes, devido a sua flexibilidade e à sua independência relativamente ao



*EMCAScript-based*, permite uma transposição entre browsers sem problemas que por vezes assolam o SCORM.

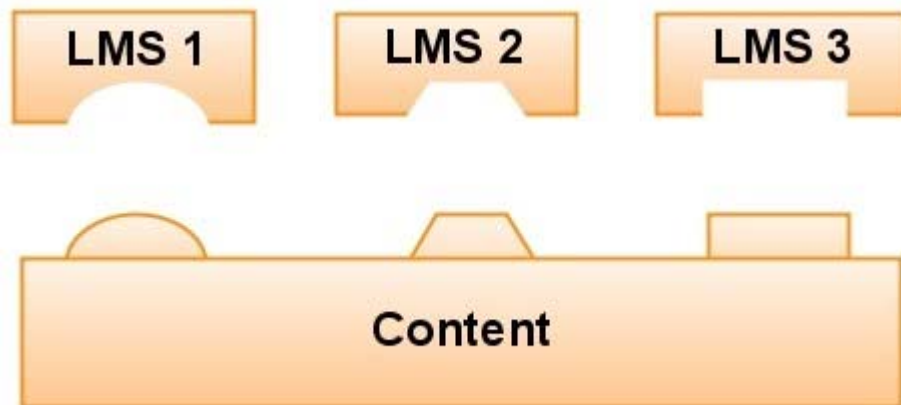


Figura 5 - Integração de conteúdo AICC com LMS<sup>13</sup>

## 2.2 Arquitectura SOA

### 2.2.1 Introdução às Arquitecturas de Software

A concepção de um projecto de software passa por uma fase designada por análise de Design de Software. Esta fase, deverá ser dividida em duas grandes áreas: Design de Alto-nível ou Arquitectural e Design Detalhado. Ambas têm um papel preponderante no sucesso do projecto de desenvolvimento.

Desde o primeiro relatório técnico datado de 1970, *Software Engineering Techniques*, vários autores tem vindo a propor a definição do termo de arquitectura de software [Buxton e Randell, 1970].

Das definições mais consensuais é a definida pelo Padrão ISO/IEEE-2000[IEEE e ISO/IEC, 2007], em que o propósito deste conceito foi o de ajudar a gerar um consenso entre estudantes, professores, autores e profissionais, sobre o uso de uma arquitectura de software.

---

<sup>13</sup> Fonte: <http://www.scorm.com/>

Assim, segundo a definição de arquitectura de software do padrão ISO/IEEE-2000, é a organização fundamental de um sistema em que todos os seus componentes, fluxos e processos que conduzem o seu design e desenvolvimento.

### **2.2.1.1 Motivação da Arquitectura de Software**

O processo de desenvolvimento de um software deverá ter capacidade para lidar com a complexidade inerente a qualquer problema, devendo ser um dos objectivos do arquitecto, que o software a ser produzido resolva o problema proposto e/ou alvo de desenvolvimento. Assim, o software deve, não só resolver o problema, como também resolvê-lo respondendo de forma assertiva aos requisitos iniciais.

A estrutura e organização em que se vai compor o desenvolvimento do produto são o que chamamos de arquitectura. Os atributos de qualidade do software se deve em grande parte à sua arquitectura, surge a necessidade de estudá-la. Com o estudo das características e técnicas do projecto, a arquitectura deverá elaborar a melhor estratégia para desenvolver os melhores produtos de software.

### **2.2.1.2 Elementos arquitecturais**

A arquitectura deve definir os elementos base que constituem o software. Os elementos arquitecturais base dependem da granularidade em que o processo de desenvolvimento pode ser dividido, ou seja, dependem da forma como o software é particionado em pedaços menores, de forma a responderem aos requisitos de negócio pretendidos.

Existem dois tipos de elementos arquitecturais: elementos estáticos e elementos dinâmicos.

Os elementos estáticos dependem das partes do sistema e da sua organização.

Este tipo de elemento define o sistema durante a fase de design e é constituído por elementos de software (ex: módulos, pacotes, classes, procedimentos), elementos de dados (e.g., entidades, ficheiros de propriedades com informação para o sistema, ou classes estáticas que contenham dados *hardcoded*), e elementos de hardware (ex: terminais onde o software irá ser executado, sistemas de apoio ao software, rede de comunicações, impressoras, etc.).

Os elementos estáticos não consistem unicamente as partes estáticas do sistema, mas também como eles irão interagir entre si. As composições e o relacionamento entre elementos estáticos formam o aspecto estático que compõe a arquitectura do sistema.

Os elementos dinâmicos definem o comportamento do sistema. Ou seja, os elementos que reflectem num determinado instante de execução o comportamento do sistema, estando neste elementos incluídos: processos, módulos, protocolos ou classes que contenham de alguma forma regras de negócio associadas a requisitos [IEEE e ISO/IEC, 2007].

### **2.2.2 Definição e Contextualização**

As arquitecturas orientadas a serviços (Service-oriented Architecture - SOA), são um novo paradigma nas arquitecturas de software.

Apesar da forte evolução das tecnologias de informação, continuamos a assistir a grandes dificuldades por parte das organizações em garantir o sucesso dos Sistemas de Informação.

Se acompanharmos os projectos de desenvolvimento ou de implementação de aplicações, vamos facilmente encontrar a especificação dos requisitos como uma das principais causas de insucesso.

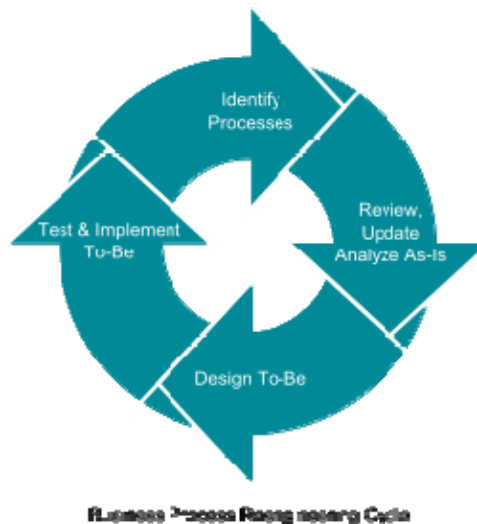
Em quase todos os projectos é identificada, na fase final, a necessidade de adicionar ou alterar os requisitos iniciais. Este facto tem um impacto significativo no plano do projecto e consequentemente na satisfação do cliente. Importará clarificar que o cliente não é o utilizador. O cliente é a organização.

Assim, o que é pretendido não é uma nova aplicação, mas sim melhores condições para cumprir a estratégia de negócio da empresa. Compreende-se que, neste quadro, não deveriam existir projectos de implementação de novas aplicações informáticas, mas apenas projectos de melhoria dos processos da organização, com suporte de aplicações informáticas. Trata-se de projectos de mudança da organização e não apenas de projectos tecnológicos. Não pode então iniciar-se um projecto deste tipo por entrevistas aos utilizadores para levantar os requisitos de informação, como é tradicional.

A abordagem por processos, normalmente designada por BPM (*Business Process Modelling*), surge como uma oportunidade para resolver problemas clássicos das organizações, nomeadamente no que respeita aos sistemas de informação.

Este conceito vem possibilitar às organizações uma visão global e integrada das tarefas de cada um dos colaboradores, permitindo deste modo, uma adaptação mais rápida e adequada, às exigências do mercado que se tem tornado cada vez mais competitivo.

Na década de 90, o BPR (*Business Process Re-engineering*) foi lançado por Hammer[Hammer, 1993], como a evolução necessária para conseguir resultados mais eficazes na melhoria das organizações, tendo até aí o TQM (*Total Quality Management*), adquirido uma notoriedade relevante.



**Figura 6-Business Process Reengineering Cycle<sup>14</sup>**

A partir dessa altura emergiram várias teorias de gestão que criaram as bases do BPM de hoje, sendo que este pode ser definido como uma integração de várias metodologias com a finalidade clara de garantir alinhamento estratégico de todos os processos das organizações.

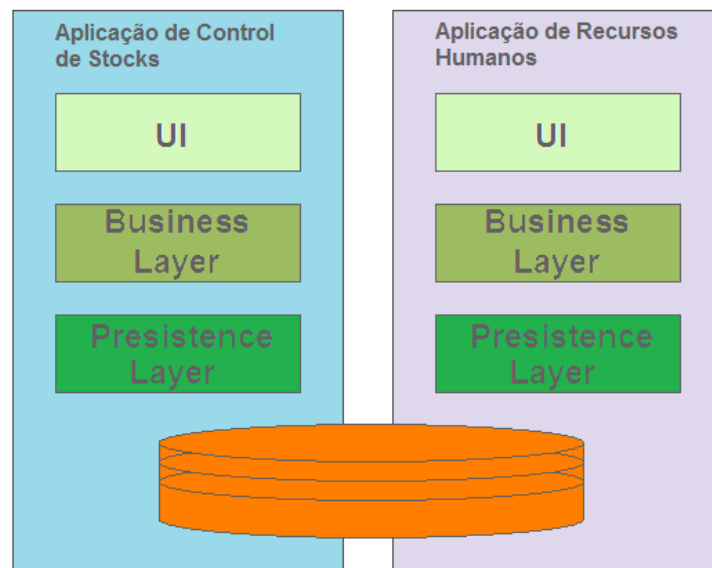
Com o aparecimento do BPM, a SOA ganhou valor, pois a filosofia de se criarem processos de negócio associados à organização fez com que o SOA se transformasse na arquitectura de software com maior viabilidade para responder aos novos requisitos propostos pelo BPM.

Com a abertura desta nova corrente de arquitectura de software, o desenho das aplicações começou a ser pensado na necessidade do reaproveitamento de elementos de software, o que leva a que uma aplicação seja pensada de forma mais granular, isto é, deverá ser o mais ágil possível, uma vez que desta forma poder-se-á efectuar uma manutenção e uma gestão mais eficaz em termos tempos e recursos.

O crescimento da Internet contribuiu para o aumento exponencial das aplicações distribuídas, abandonando-se assim as aplicações *stand-alone* em que todos os recursos estavam focalizados única e exclusivamente para responderem aos requisitos do utilizador.

---

<sup>14</sup> Fonte: [http://en.wikipedia.org/wiki/Business\\_process\\_reengineering#cite\\_ref-1](http://en.wikipedia.org/wiki/Business_process_reengineering#cite_ref-1)



**Figura 7 - Arquitectura tradicional de aplicações 'stand-alone'**

A SOA leva o conceito de “foco no requisito” mais à frente. Deixa de responder ao utilizador final e passa a responder à organização e aos modelos de negócio.

Assim, este tipo de arquitectura de software passa a ter uma importância mais significativa em termos de gestão, pois uma organização poderá efectuar uma reengenharia de processos sem ter que efectuar alterações profundas no seu sistema de informação, levando assim a uma redução drástica dos custos inerentes às mudanças.

### **2.2.3 Arquitectura da informação**

A arquitectura da informação materializa os elementos estáticos e dinâmicos do sistema de informações da SOA usando camadas, sistemas e fluxos de informação.

Inicialmente deve apresentar numa primeira versão, tendo em conta diversos aspectos tais como a segurança e a gestão de sistemas, que também são aspectos chave da implementação da SOA.

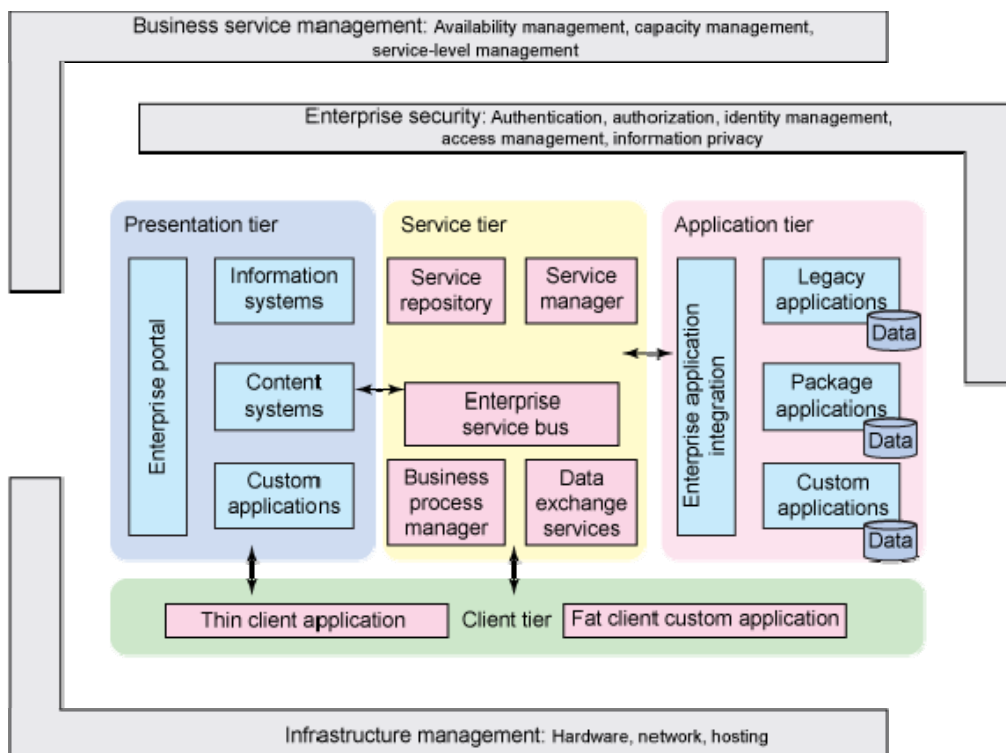


Figura 8 - Arquitectura da informação de referência<sup>15</sup>

A arquitectura do sistema deverá juntar todos os elementos arquitecturais num ciclo de vida da SOA para mostrar como se encaixam juntos no ambiente, com os processos de sistemas e os fluxos de informação por este gerados.

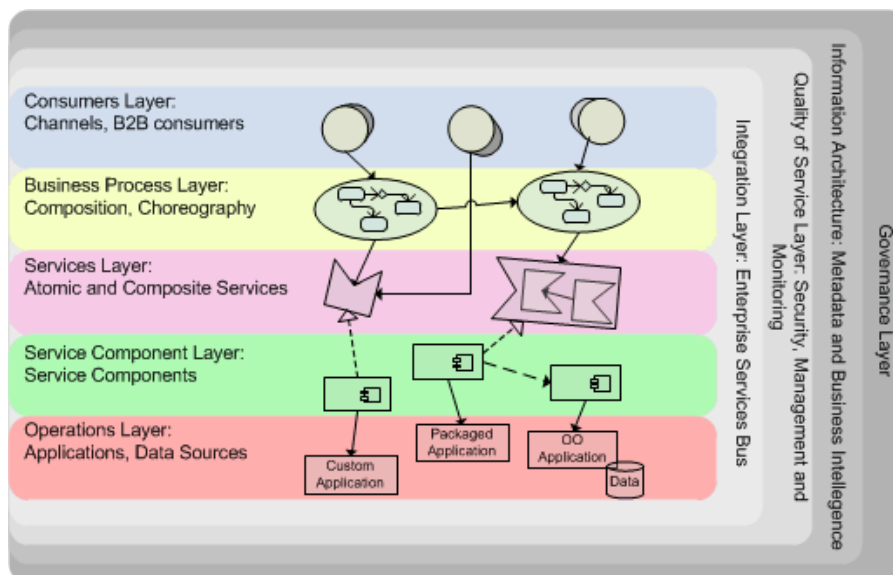
Assim, a arquitectura da informação deverá ter como missão a apresentação de uma arquitectura de alto nível, esquematizando os sistemas envolvidos, de forma fácil de identificar, onde todos os sistemas estejam representados, para que seja possível, uma análise isolada de cada um dos componentes do sistema de informação.

## 2.2.4 Arquitectura das camadas de referência da aplicação SOA

A arquitectura das camadas de referência da aplicação engloba todas as camadas de serviços de forma a conseguir efectuar um rastreio de todos os componentes envolvidos na plataforma

<sup>15</sup> Fonte: <http://www.ibm.com/developerworks/br/rational/library/10/soa-togaf-part2-service-oriented-architecture/index.html>

do sistema a desenvolver, agrupando-os em pacotes e subsistemas da arquitectura da informação.



**Figura 9 - Arquitectura dos componentes de referência<sup>16</sup>**

Este modelo deverá ser utilizado para análise do sistema organizacional, da arquitectura de dados, análise de componentes de serviços e processo de negócios, análise de impacto de arquitectura, planeamento de projecto e actividades de planeamento de gestão do sistema.

Este modelo tem como principal função ser uma estrutura dentro da organização, demonstrando a sua influência sobre o sistema de informação e a sua gestão.

### 2.2.5 O Futuro, Arquitectura Orientada à Web (WOA)?

A arquitectura orientada à Web surgiu há alguns anos por Gall<sup>17</sup>. Desenvolveu o conceito que desde então tem vindo a crescer e a ganhar valor no mundo das TI.

<sup>16</sup> <http://www.ibm.com/developerworks/br/rational/library/10/soa-togaf-part2-service-oriented-architecture/index.html>

<sup>17</sup> Fonte: [http://blogs.gartner.com/nick\\_gall/2008/11/19/woa-putting-the-web-back-in-web-services/](http://blogs.gartner.com/nick_gall/2008/11/19/woa-putting-the-web-back-in-web-services/)



Figura 10 - Visão da arquitetura orientada a Web.<sup>18</sup>

Hinchliffe publicou recentemente [A Web-Oriented Architecture \(WOA\) Un-Manifesto](#)<sup>19</sup>, onde detalha uma série de princípios que orientam esta arquitetura.

No futuro o WOA não deverá ser uma arquitetura que irá alterar os princípios do SOA, mas sim irá dar uma nova importância à Internet dentro de uma organização, devendo passar a ser vista não como uma parte integrante das ferramentas mas sim como uma componente de negócio da própria organização.

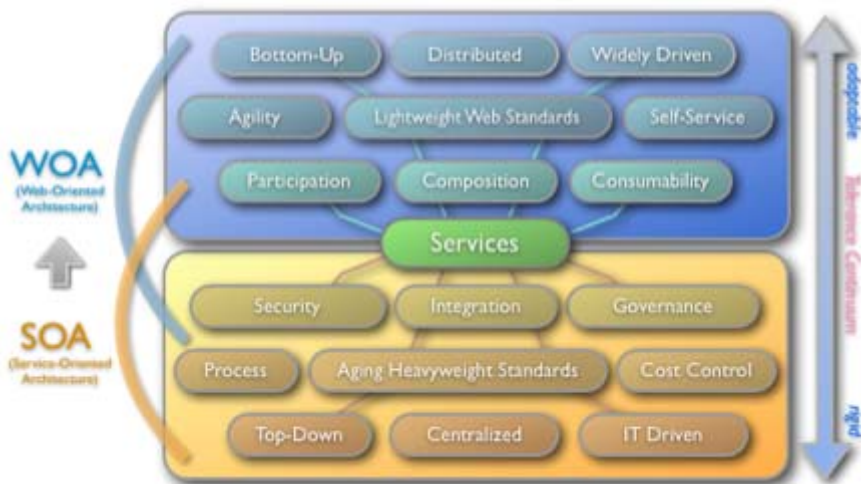


Figura 11 - Evolução de SOA para WOA<sup>20</sup>

<sup>18</sup> Fonte: <http://hinchcliffe.org/archive/2009/12/14/18179.aspx>

<sup>19</sup> Fonte: <http://hinchcliffe.org/archive/2009/12/14/18179.aspx>

<sup>20</sup> Fonte: <http://www.zdnet.com/blog/hinchcliffe/>



A disponibilização de serviços das organizações em ambiente Web, irá permitir uma interoperabilidade crescente entre as organizações, o que por sua vez levará a que cada uma produza serviços com valor acrescentado para utilização interna e para partilha de serviços, o que levará inevitavelmente a uma poupança de recursos.

## Capítulo 3

# 3 Especificação de uma plataforma de e-learning orientada a serviços

Uma plataforma de e-learning para funcionar de forma independente dos conteúdos, deverá ter associada a si, uma Framework de Javascript, de modo a capturar os eventos gerados durante a navegação nos conteúdos.

Habitualmente essa Framework é gerida pela aplicação em si, isto é, encontra-se embebida nela, sendo da responsabilidade do conteúdo indicar a informação gerada pela aprendizagem. No entanto, o que se pretende é eliminar as dependências directas das regras de negócio que o e-learning tem, relativamente aos conteúdos, permitindo assim que a plataforma e-learning (Learning Management System - LMS) consiga interpretar as mensagens geradas e consequentemente tomar acções relevantes referentes às actividades de aprendizagem/formação.

Note-se que para isso, os serviços disponibilizados devem passar a ser dotados de inteligência de negócio relativamente à aprendizagem, e que passem a ter utilidade não só numa plataforma LMS como também noutra plataforma qualquer externa ao ambiente de aprendizagem de e-learning, passando assim a ser entendido como um serviço externo.

As aplicações práticas podem ser diversas, se atendermos a que um conteúdo pode ser disponibilizado não só para uma plataforma de visualização em ambiente de aprendizagem, como também num modo de apresentação de conteúdos.

## **3.1 Requisitos**

### **3.1.1 Disponibilidade de acesso à plataforma**

A plataforma deverá estar disponível para todo e qualquer utilizador que nela pretenda efectuar cursos. Os utilizadores podem ser aluno ou formando, tutor e administrador. Por uma questão de simplificação o perfil de aluno ou formando será designado ao longo do texto de Formando, uma vez que a plataforma desenvolvida está mais vocacionada para a formação. Para isso deverá o utilizador interessado, efectuar o registo através do menu de subscrição o que deverá desencadear o processo de inscrição do utilizador na plataforma.

### **3.1.2 Login na plataforma**

O utilizador quando já devidamente inscrito deverá ter a possibilidade de efectuar o login e entrar na plataforma, através das credenciais de segurança fornecidas por e-mail.

### **3.1.3 Página Inicial**

Na primeira página de acesso, o aluno ou formando deverá ter à sua disposição uma listagem agrupada com os cursos que poderá efectuar naquele momento.

A listagem está organizada por grupo, que deverá ter um nome condizente com os cursos disponibilizados.

### **3.1.4 Inscrição num curso**

A inscrição do aluno para a frequência de um curso deverá ser efectuada na página inicial, nos detalhes do curso publicitado, no único botão disponível nessa área.

Aquando da inscrição, o curso deixa de estar disponível como publicitado e passa a estar nos cursos matriculados.

### **3.1.5 Página de cursos matriculados**

A página de cursos matriculados, fornece uma listagem de todos os cursos onde o aluno está matriculado. Estes cursos não aparecem na página inicial, de cursos publicitados.

### **3.1.6 Frequência de curso**

Na listagem de cursos matriculados, o aluno tem no menu de acções, a opção de “play”. Esta opção deverá despoletar o curso, abrindo sobre a aplicação uma janela onde o conteúdo deverá ser mostrado.

### **3.1.7 Performance**

No menu performance, na listagem de cursos matriculados, o aluno tem a opção de verificar o estado da sua matrícula, assim como os valores da avaliação que obteve durante a frequência do curso.

## **3.2 Casos de Uso**

Os modelos de caso de uso têm como finalidade a descrição pormenorizada dos processos associados aos requisitos enunciados no [ponto 3.1](#).

Para explicar os processos aplicados a todos os requisitos, foi restringida a explicação mais detalhada apenas aos requisitos com maior importância no âmbito da presente dissertação.

### **3.2.1 Caso de uso login na plataforma**

**Âmbito:**

Login na Plataforma

**Cenário:**

Acesso à Plataforma

**Actores:**

Formando

**Descrição do processo:**

Na pagina inicial, são apresentadas duas caixas de texto, nas quais o Formando deverá introduzir o *username* e a *password*.

**Pré-condições:**

O *username* e *password* do Formando deve estar previamente registadas na base de dados.

**Pós-condições:**

Acesso foi efectuado. O formando visualiza quais os cursos disponíveis para frequência. O Formando visualiza os cursos nos quais efectuou matrícula.

**Descrição do fluxo de processo:**

O formando irá colocar as carenciais de acesso há plataforma, (*username* e *password*), condição prévia, e após validação do sistema irá ser efectuada o redireccionamento para a página de entrada na plataforma SKyLearn.

**Problemas em aberto:**

Neste momento não existem problemas em aberto.

### 3.2.2 Caso de uso frequência de curso

**Âmbito:**

Frequência do Curso

**Cenário:**

Listagem de cursos onde o formando esta inscrito.

**Actores:**

Formando

**Descrição do Processo**

Na listagem de cursos, o formando tem acesso à acção de visualização do curso, onde lhe abrirá uma nova janela interior, para apresentação do curso.

**Pré-condições:**

O formando deverá estar previamente inscrito em pelo menos um curso disponível para frequência.

**Pós-condições:**

O formando terá acesso a um interface gráfico interior, onde será apresentado o conteúdo.

Ao longo da apresentação, deverão ser registadas as acções do formando e a avaliação final.

**Descrição do fluxo de processo:**

O formando efectuará um clique sobre o botão de apresentação, que por sua vez deverá abrir uma janela interior para disponibilização do conteúdo.

Na janela interior, o curso é apresentado numa área delimitada, onde existe um botão no canto superior direito para terminar a apresentação em qualquer altura.

A API SCORM, efectuará a comunicação entre os conteúdos e a plataforma, efectuando todos os registos necessários ao suporte do conteúdo.

**Problemas em aberto:**

Neste momento não existem problemas em aberto, tendo sido o comportamento especificado com a norma SCORM, e respondendo a eventos gerados durante a navegação nos conteúdos SCORM.

### **3.2.3 Caso de uso performance**

**Âmbito:**

Após a frequência do curso.

**Cenário:**

Lista de cursos matriculados.

**Actores:**

Formando.

**Descrição do Processo**

Na listagem de cursos, o formando tem acesso à acção de visualização dos seus resultados de aprendizagem, relacionados com o curso que frequentou. Estes resultados deverão ser apresentados num painel que será acrescentado abaixo do curso, aumentando assim a altura útil disponível para os conteúdos do curso.

**Pré-condições:**

O formando deverá estar previamente inscrito em pelo menos um curso disponível para frequência.

**Pós-condições:**

O formando terá acesso a um interface gráfico, onde será apresentado o resultado pedagógico do curso.

**Descrição do fluxo de processo:**

O formando efectuará um clique sobre o botão de performance, que por sua vez deverá acrescentar um painel na linha do curso cuja performance pretende verifica.

No painel de performance, o curso é apresentado numa área delimitada, onde deverá ter informações relativas à data de início do curso, assim como a sua data final, (caso o formando já tenha terminado o curso), assim como o resultado da sua frequência e o resultado final do curso, (*in progress, passed, completed*). Quando o utilizador ainda não iniciou o curso o seu resultado é nulo, não tendo assim nada a apresentar.

**Problemas em aberto:**

Neste caso, o encerramento do curso deverá ficar ao abrigo do conteúdo, assim sendo a plataforma deverá limitar-se a registar os indicadores do curso ao longo da sua frequência.

O ponto acima não será propriamente um problema, pois o pretendido é que os conteúdos SCORM sejam completamente independentes das plataformas, ou seja eles possuírem a inteligência de indicar se o formando frequentou o curso com sucesso.

### **3.3 Diagrama de Actividades**

Para cada um dos requisitos existe um diagrama de actividades. O objectivo do estudo nesta dissertação é predominantemente o SCORM, pretendendo-se demonstrar apenas a actividade de frequência de um curso com conteúdo SCORM e não as componentes de gestão de cursos, alunos, docentes, entre outras, associadas aos LMS.

As actividades ocorridas na visualização de um curso, são próprias de um conteúdo. Assim os pedidos efectuados deverão ser os necessários para satisfazer as necessidades de negócio do próprio conteúdo.

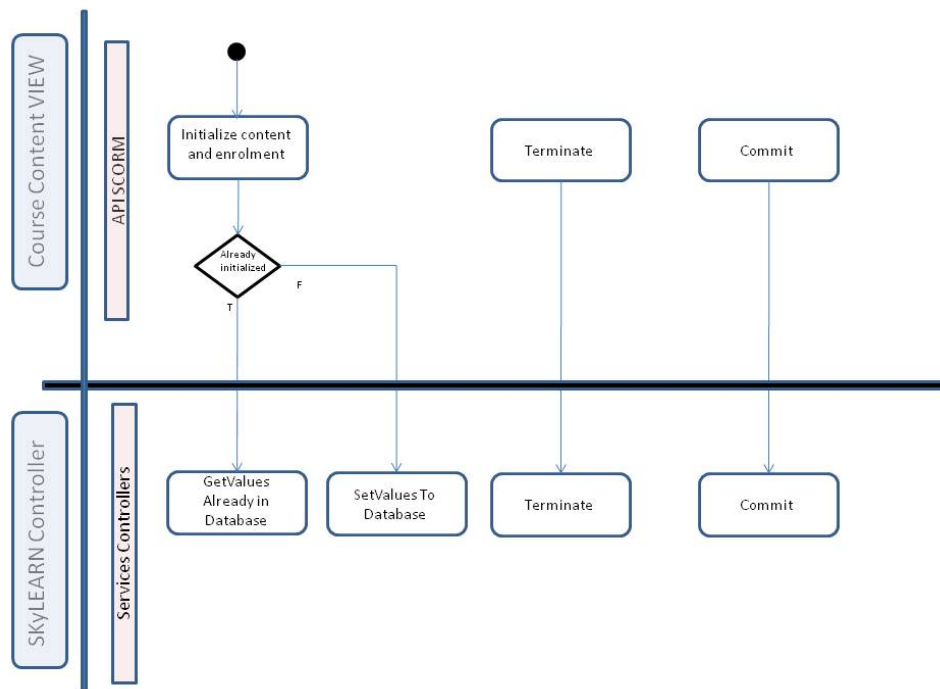


Figura 12 - Diagrama de Actividades do Modelo de Negocio SCORM

Um conteúdo SCORM tem como primeira actividade a sua inicialização. Neste momento, a plataforma inicializará todos os componentes envolvidos no curso caso seja um curso frequentado pela primeira vez. Inicializará o estado de frequência do curso, a sua data de inicio, os seus registos SCORM (*location, session time, etc*).

Caso seja um curso que tenha sido inicializado previamente, a API do *Controller* da plataforma, irá verificar e actualizar os valores para os novos obtidos nesta sessão aberta.

Ao longo do curso, devem ser efectuados registos relativamente à localização no curso(*location*), ao tempo de sessão(*session time*), assim como a registos mais complexos, tais como interacções e objectivos do conteúdo.

A plataforma irá também registar dados relativamente a navegação para outros conteúdos (norma ADL), em que irá registar url's de pedidos que vão ser despontados em determinado momento durante a navegação através dos conteúdos.

Estes registos e a extracção de informação são efectuados através de dois métodos:

- **GetValue**, que tem como argumento somente uma chave identificativa que permite à API do controlador descobrir qual o elemento pretendido pelo conteúdo (*GetValue(KEY), GetValue('cmi.location')*);
- **SetValue**, que tem como parmetros uma chave identificativa do elemento a escrever e o valor (*SetValue(KEY, VALUE), SetValue('cmi.location',10)*).



No momento do fecho da janela com os conteúdos do curso, o conteúdo invocará o método *Terminate*, que irá efectuar o cálculo e a persistência (*Commit*) dos registos que foram gravados durante o percurso de aprendizagem.

### 3.4 Diagrama de UML

Neste diagrama UML são apresentados os serviços SCORM associados, pois não se pretende que a plataforma seja em si o modelo de estudo, mas sim somente os serviços e a sua forma de invocação.

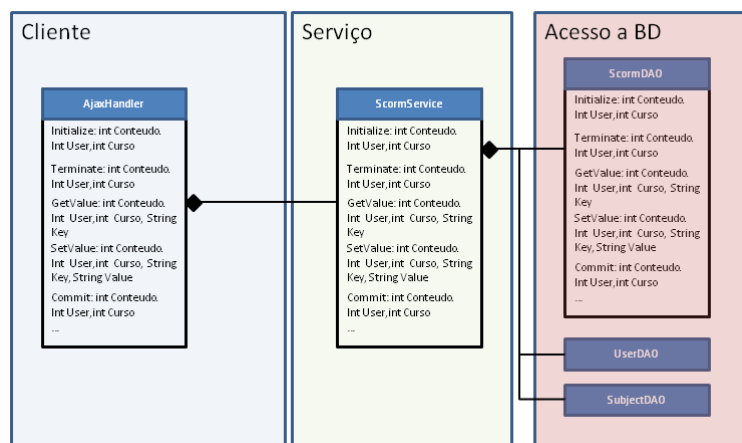


Figura 13 - Diagrama UML de Arquitectura

As chamadas da API SCORM para um cliente Java foram efectuadas recorrendo a uma Framework de *AJAX*, o *DWR*<sup>21</sup>, em que é possível mapear para uma classe, chamadas a métodos de *Javascript*.

Este tipo de abordagem foi pensada para que os pedidos SCORM fossem logo processados e entregues.

Para a elaboração dos serviços recorreu-se à Framework *SPRING*<sup>22</sup>, onde a configuração dos serviços é efectuada através de um ficheiro XML, onde se poderá configurar qual o tipo de Java Beans a utilizar.

<sup>21</sup> <http://directwebremoting.org/>

Os serviços são instanciados no momento da criação do contexto da aplicação, onde é efectuada a leitura do ficheiro XML, passado por parâmetro para o *listener* do contexto através da configuração da aplicação Web no ficheiro *web.xml*.

A vantagem da utilização do *container* da Framework SPRING é a sua capacidade de adaptação à exposição quer seja como serviços de J2EE quer seja como Web Services. Esta vantagem foi crucial para a escolha da Framework.

A camada de acesso à Base de Dados tem duas formas de acesso, via JPA<sup>23</sup> e JDBC.

A JPA, é uma Framework de persistência Java, que permite a utilização de entidades relacionais como objectos Java. Deste modo, apresenta características idênticas à Framework Hibernate, permitindo a utilização de mapeamentos do modelo de dados em objectos Java directos, mapeando os relacionamentos do modelo de dados, em relacionamentos lógicos entre objectos Java.

O JDBC, permite um acesso mais directo à base de dados, sendo que em detrimento da simplicidade de manuseamento das entidades, a velocidade será a sua principal vantagem, assim a escolha de acessos JDBC ao invés de se manter o JPA foi devido à necessidade de uma maior rapidez no acesso para gravação e consulta dos dados, para a resposta aos pedidos do SCORM.

### 3.5 Modelo E-R

O Modelo de dados apresentado, contém apenas as tabelas base para o funcionamento da norma SCORM.

Pretende-se apenas reflectir as acções mais habituais num ambiente de aprendizagem, e que estão contempladas neste modelo.

---

<sup>22</sup> <http://www.springsource.org/>

<sup>23</sup> <http://www.oracle.com/technetwork/articles/javaee/jpa-137156.html>

### 3.5.1 Modelo E-R de Gestão de Utilizadores

Para demonstrar o modelo de gestão de cursos e alunos, para que assim seja possível verificar que dentro de um ambiente de e-learning existe a ligação para um utilizador e um conteúdo, permitindo registar acções do utilizador sobre esse conteúdo.

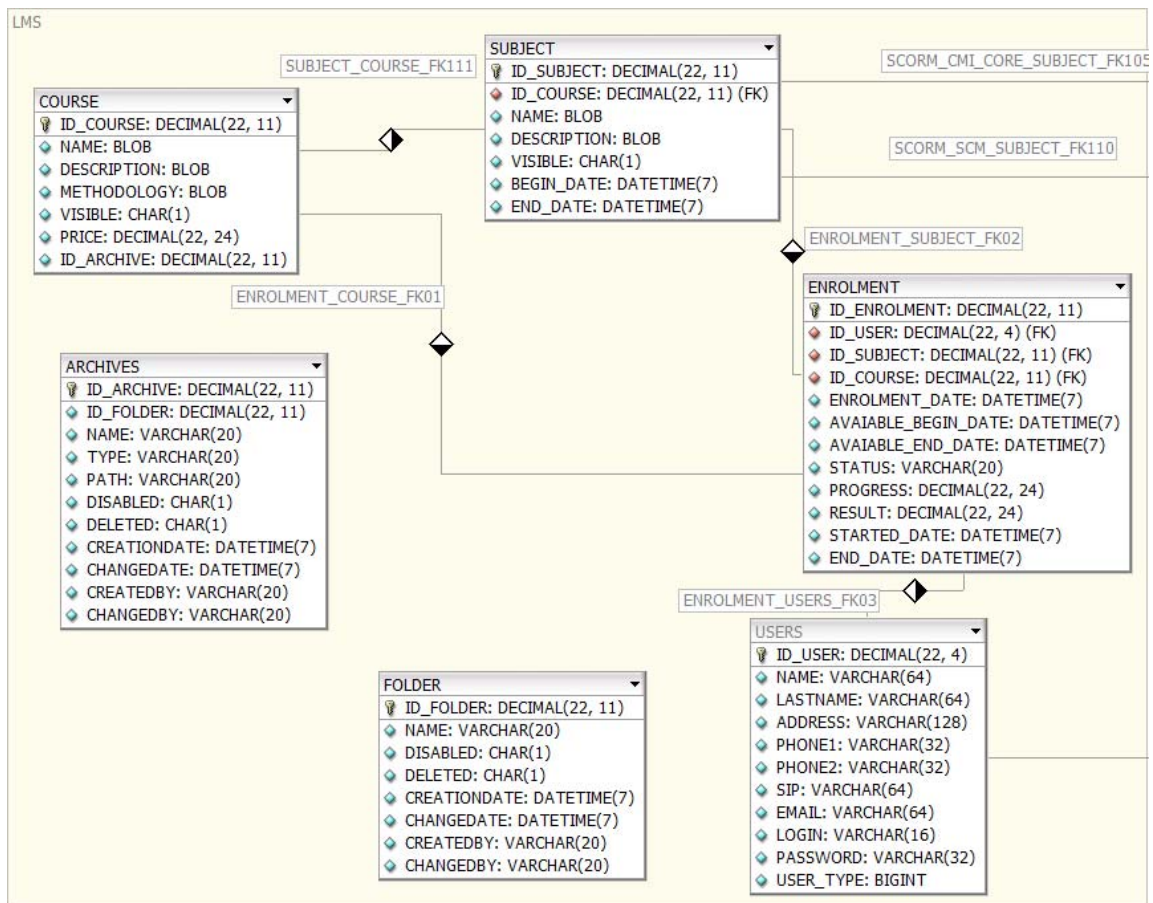


Figura 14- Modelo E-R, Gestão de Utilizadores

As descrições das tabelas acima apresentadas são:

**Tabela 1 - Discriminativo do modelo de dados de gestão do LMS**

<i>Tabela</i>	<i>Comentário Discriminativo</i>
USERS	Tabela que contém informação relevante aos utilizadores registados na plataforma de e-learning, esta tabela tem como chave primária o campo ID_USER
COURSES	Tabela que contém informação relativa aos cursos a que podem estar agregados os utilizadores. Nesta tabela poderemos definir algum género de informação que irá ser mostrada ao utilizador, assim como numa plataforma de venda de cursos online poderá indicar o preço de um determinado curso.  Contem ainda campos que podem ser utilizados para acesso à informação de um curso, somente a utilizadores devidamente registados na plataforma.
SUBJECT	Tabela SUBJECT contém os módulos requeridos em cada curso, ou seja, existe a relação de um curso para várias disciplinas (representada pela tabela SUBJECT), e uma disciplina é somente relacionada com um único curso.  Esta tabela contém informação de apresentação, assim como informação que poderá ser utilizada como negócio na disponibilização do curso, os campos BEGIN_DATE e END_DATE.
ENROLMENT	Tabela representativa da associação de um utilizador a uma disciplina. Contém de forma resumida a informação do utilizador relativamente ao conteúdo de uma disciplina. Regista também o resultado do progresso, se o utilizador chegou ou não ao fim do curso, o status da disciplina relativamente ao utilizador, se o utilizador completou com sucesso o curso, e o resultado obtido nessa disciplina (nota final)

As tabelas de gestão de utilizadores não contém nenhuma referencia directa às tabelas de registo dos conteúdos, apenas contém informação referente a regras de negócio dos serviços da Framework.

### 3.5.2 Modelo E-R de Gestão de conteúdos

O modelo de gestão dos conteúdos, dentro de um ambiente de e-learning contém a ligação do conteúdo à disciplina associada, e ao utilizador. As tabelas que se seguem, demonstram como a informação é registada pela Framework SCORM, e que deriva da interacção entre a disciplina, o conteúdo e o formando.

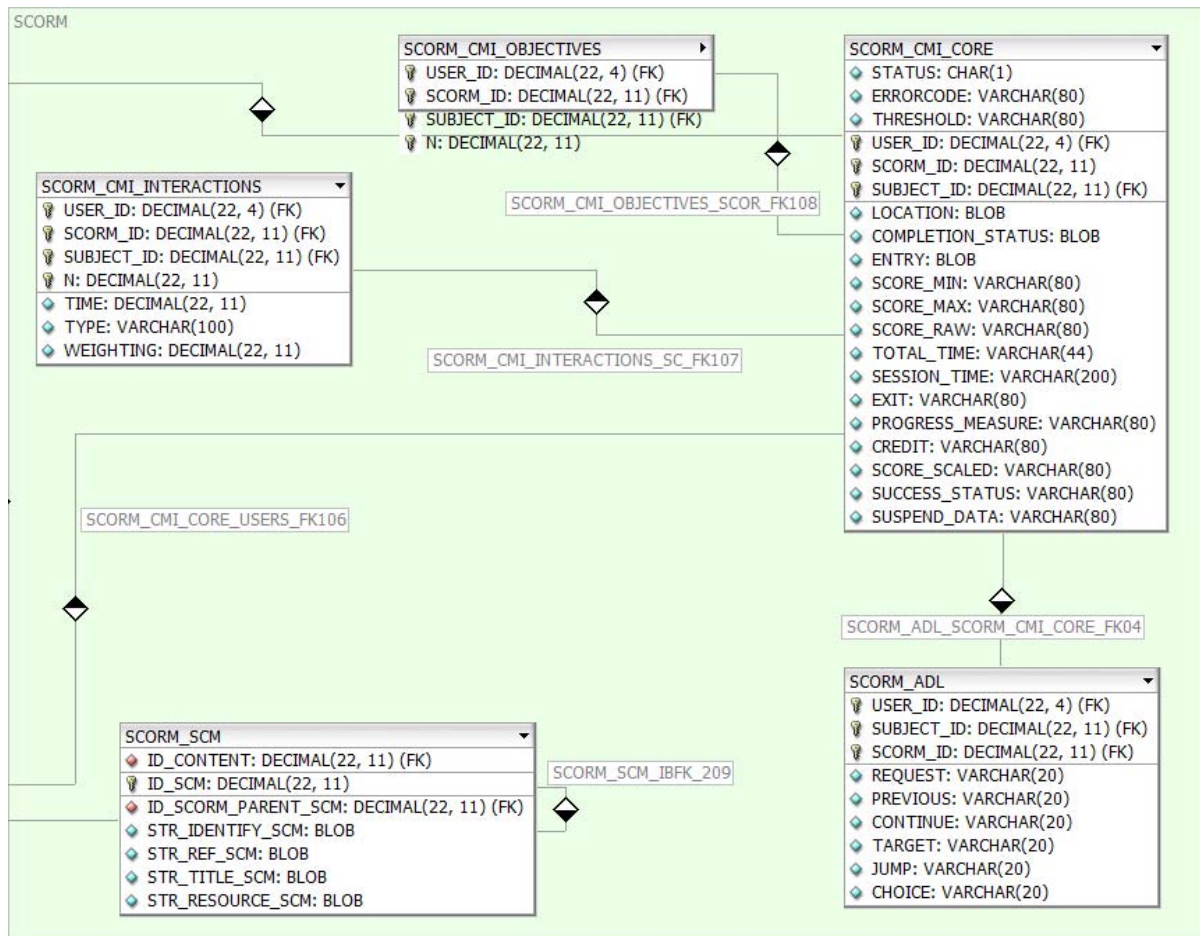


Figura 15 - Modelo E-R da Gestão de Conteúdos

As descrições das tabelas acima apresentadas são as seguintes:

Tabela 2 - Discriminativo do modelo de dados de gestão de conteúdos

<i>Tabela</i>	<i>Comentário Discriminativo</i>
SCORM_SCM	Tabela que contém informação relevante aos conteúdos presentes na plataforma. Esta informação é independente do utilizador e da turma, pois esta tabela é produto do resultado final do <i>parsing</i> do ficheiro de Manifest do conteúdo, (imsmanifest.xml)
SCORM_CMI_CORE	Tabela que contém informação relativa ao curso, disciplina, utilizador e resultados da interacção com os conteúdos. Esta tabela regista as localizações dentro do conteúdo, assim como o resultado final, o estado(completo ou incompleto), tempo do conteúdo em sessão, assim como os limites do desempenho do formando.
SCORM_ADL	Tabela SUBJECT, contém informação referente ao redireccionamento do conteúdo para outro contexto, ou outro conteúdo. Como o próprio nome indica isto provém da implementação da NORMA SCORM2004, nomeadamente a questão de sequenciação dos conteúdos.
SCORM_CMI_OBJECTIVES	Tabela representativa de objectivos, esta tabela tem como finalidade registar resultados das actividades do conteúdo.
SCORM_CMI_INTERACTIONS	Tabela representativa das iterações sobre um conteúdo distribuído num <i>Content Package</i> , Norma SCORM2004 v.3, variante IMS.

### 3.5.3 Diagrama de Classes de Serviços

Uma arquitectura SOA, não força obrigatoriamente ao uso de Web-Services publicados da forma mais tradicional conhecida, com um ficheiro XML descritor de serviço e por sua vez, publicado, assim no caso da plataforma em estudo foram publicados serviços num *container*

J2EE, o Spring, não ficando acessíveis a sistemas produzidos por outras tecnologias que não JAVA.

O uso destes serviços deve-se à sua capacidade de transformação mediante uma parametrização para Webservices públicos, em que permitem a um cliente, qualquer que seja a sua ligação, o acesso ao serviço publicado. Assim, e para facilitar a flexibilização destes serviços, foram desenhados da seguinte forma:

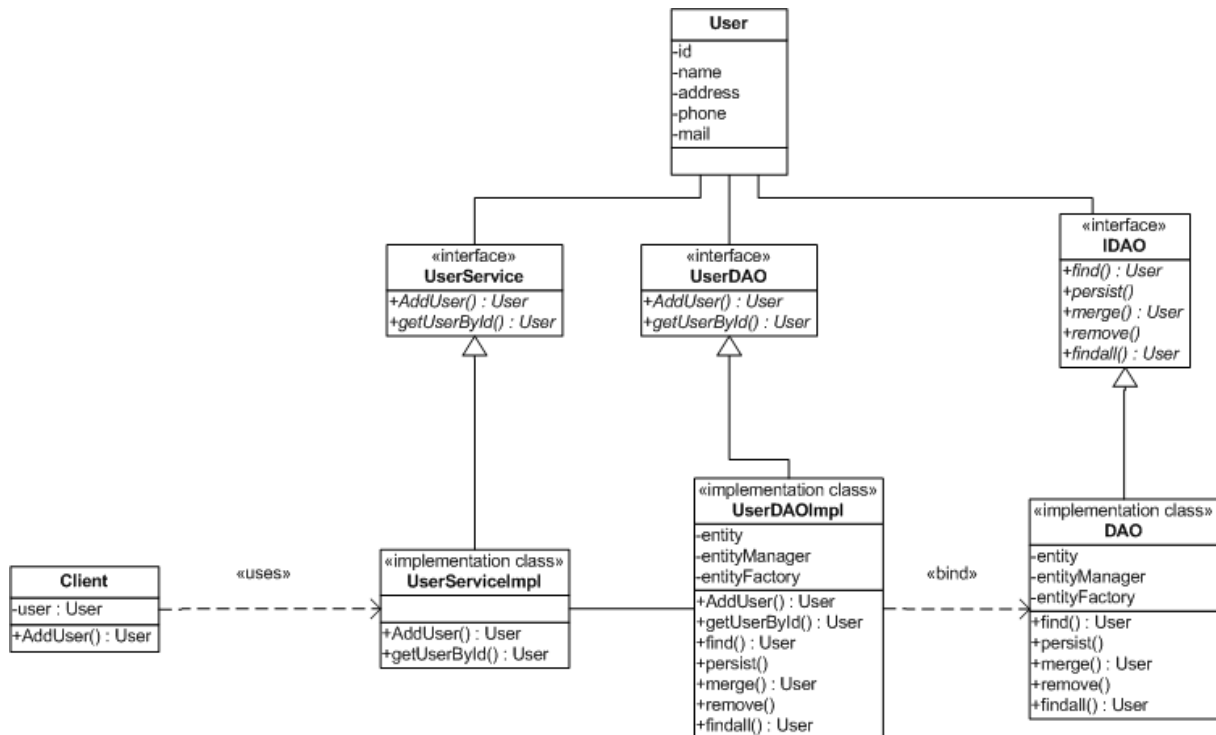


Figura 16 - Diagrama de Classes do serviço de Users

A descrição de cada serviço está na classe que o implementa, assim, com vista a uma possível expansão, as classes que implementam os serviços, como é neste caso a UserServiceImpl, contém a definição da sua natureza, ou seja, é nela que se vai definir se irá ser um Webservice, ou um serviço publicado por um *container*.

Toda a aplicação esta elaborada no sentido de permitir uma expansão para novas tecnologias, sendo somente necessário a implementação das *interfaces*, e por sua vez definir a tecnologia de publicação do serviço, (AXIS<sup>24</sup>, REST<sup>25</sup>, SPRING, EJB<sup>26</sup>).

<sup>24</sup> <http://ws.apache.org/axis/>

<sup>25</sup> <http://www.oracle.com/technetwork/articles/javase/index-137171.html>

### 3.5.4 Diagrama de Sequência dos Web Services

Uma arquitetura SOA utiliza o paradigma de *find-bind-execute*, para a utilização dos serviços. O princípio base deste paradigma baseia-se no ciclo de vida do serviço, onde um cliente/consumidor do serviço, realiza uma pesquisa para o encontrar, após encontrar o descritivo dele, pode ser um documento WSDL<sup>27</sup>, no caso de um Webservice, efectua uma ligação ao serviço, ficando assim pronto para o consumir.

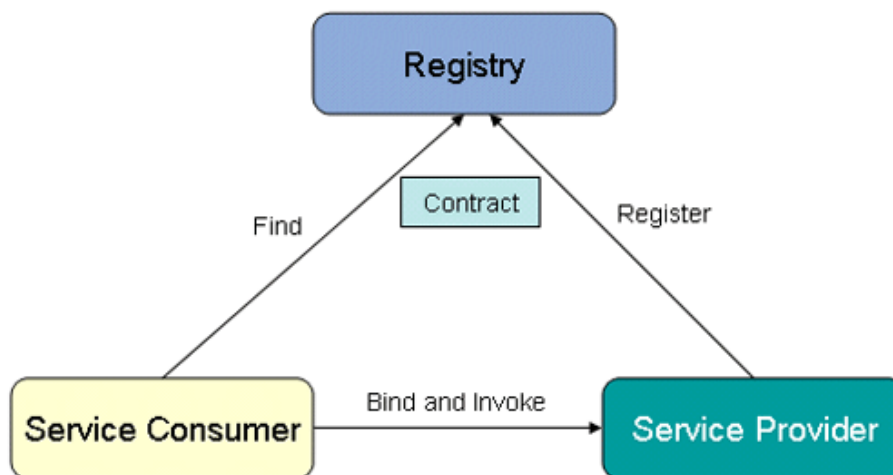


Figura 17 - Ciclo de vida de um Webservice<sup>28</sup>

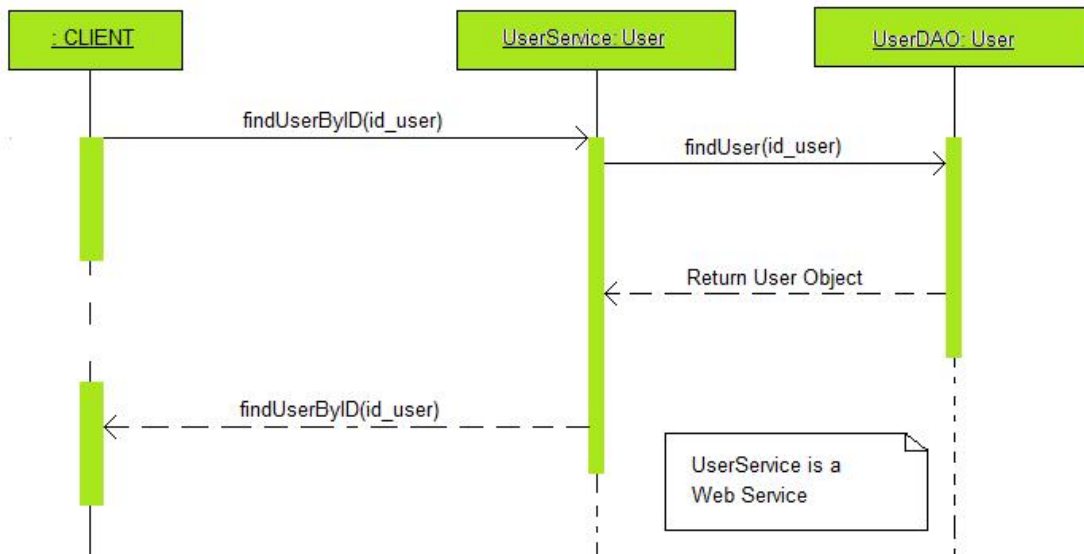
No lado do servidor, ou produtor do Webservice, necessita apenas de o registar e publicar o seu WSDL, que por sua vez contem a especificação do serviço.

<sup>26</sup> <http://www.oracle.com/technetwork/java/index-jsp-140203.html>

<sup>27</sup> Web Service Definition Language – Documento XML para especificação de um webservice, [www.w3.org/TR/wsd1](http://www.w3.org/TR/wsd1)

<sup>28</sup> Fonte: <http://www.oracle.com/technetwork/articles/javase/index-142519.html>





**Figura 18 - Ciclo de vida de um pedido a um Webservice<sup>29</sup>**

Uma chamada a um serviço tem como limites do ciclo de vida desde a elaboração do pedido, até à entrega da resposta a esse mesmo pedido.

Assim, na figura acima o cliente efectua o pedido ao serviço, UserService, valida-o, caso este seja um pedido válido, efectua o pedido, findUserByID, retornando-o ao interface de ligação, que por sua vez o irá entregar ao cliente.

<sup>29</sup> Fonte: <http://www.devx.com/enterprise/Article/28722/0/page/2>

## Capítulo 4

# 4 Desenvolvimento da Plataforma SKyLearn

A plataforma SKyLearn, foi desenvolvida na presente dissertação de mestrado, como objecto de estudo do impacto das arquitecturas SOA em ambientes de e-learning.

O Desenvolvimento da plataforma foi efectuado em linguagem JAVA, versão 1.6.0\_22.

Os servidores aplicacional e de base de dados, foram Apache Tomcat 6.0 e Oracle Database 10g Express Edition Release 10.2.0.1.0 respectivamente.

### 4.1 Tecnologias adoptadas

O tema em estudo nesta dissertação, impacto das arquitecturas SOA em ambientes de e-learning, obriga a uma elevada flexibilidade de padrões de desenvolvimento, sendo que neste ponto a linguagem de programação JAVA demonstra as suas mais-valias.

Sendo uma linguagem de alto nível, baseada na orientação a objectos, está intrínseco na sua filosofia o conceito de desenvolvimento "*as service*", sendo que a sua evolução natural foi para o conceito de empresarial, em que providencia Frameworks para responderem às mais diversas aplicações empresariais<sup>30</sup>.

Algumas dessas Framework foram utilizadas no desenvolvimento desta aplicação, tais como JPA (Java Persistence API), JNDI (Java Naming and Directory Interface), entre outras.

---

<sup>30</sup> Fonte: [http://www.oracle.com/technology/sample\\_code/tech/java/j2ee/javacookbook/overview.html](http://www.oracle.com/technology/sample_code/tech/java/j2ee/javacookbook/overview.html)

O servidor aplicacional, Apache Tomcat, foi escolhido devido às suas características de "Servlet Container", o que o torna um servidor muito leve relativamente a um servidor J2EE aplicacional.

O motor de base de dados utilizado foi o Oracle XE, Express Edition, devido à sua fiabilidade e rapidez, uma vez que possui com base o motor de nível empresarial do Oracle 10g, no entanto com a limitação de 4Gb de informação.

## 4.2 Desenvolvimento de Interface Gráfico.

A necessidade de efectuar um interface gráfico de uma forma rápida em HTML reflectiu a escolha efectuada, numa framework Web de fácil manuseamento e de elevada fiabilidade.

Assim, Framework Stripes<sup>31</sup> foi a escolhida, devido à sua fácil utilização, suporte para J2EE e ter necessidades de configurações mínimas<sup>32</sup>.



Figura 19 - Homepage de apresentação da plataforma SkyLearn

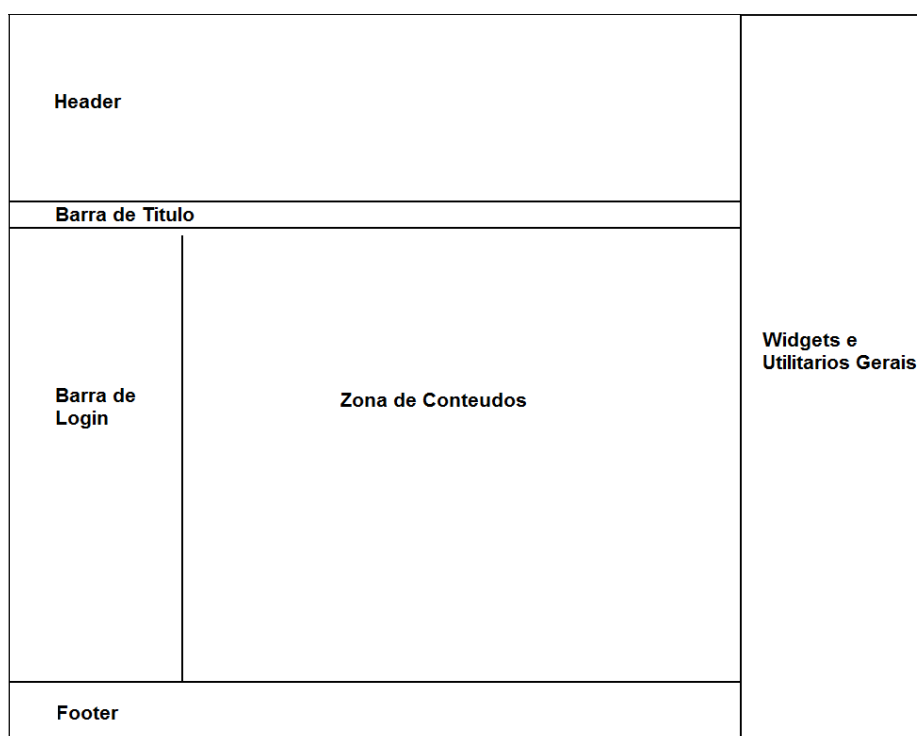
<sup>31</sup> Fonte: <http://www.stripesframework.org/>

<sup>32</sup> Fonte: <http://www.devx.com/Java/Article/31921>

O interface gráfico, foi elaborado a pensar numa utilização profissional, fora do mundo académico, tendo uma filosofia de protótipo, sem um rigor ao nível de design que um produto final deverá ter.

Na elaboração do protótipo, teve-se o cuidado de apresentar um layout com áreas bastante distintas e com o objectivo de agrupar informação e acções mediante o seu tipo de utilização.

Assim o layout da plataforma divide-se da seguinte forma:



**Figura 20 - Layout da Homepage da plataforma SkyLearn**

A área de Header surge na forma de dar resposta a acções gerais sobre a aplicação, incluindo alguns menus gerais, como notícias, subscrição da aplicação e área de contactos.

A barra de título, de uma forma geral indica qual a acção concretizada, apresentando o seu título nesta barra. A área de Login, está sempre presente na Homepage da aplicação, devido à necessidade de proporcionar ao utilizador a possibilidade de efectuar o login sempre e em qualquer área da Homepage. Na zona de conteúdos, onde se apresenta as informações seleccionadas previamente pelas acções.

A Área de Footer servirá apenas para futuras acções de publicidade. A Área de Widget e utilitários, é composta por utilitários gerais, tais como calendário e relógio. Esta área está presente em todo o ciclo de vida da aplicação.

Na elaboração do *layout* recorreu-se a uma das mais-valias da framework Web escolhida, uma vez que traz de forma nativa, ferramentas de layout.

Na pagina de definição do layout indica-se quais as áreas que serão incluídas, exemplo:

```
<stripes:layout-component name="header">
    <jsp:include page="header.jsp" />
</stripes:layout-component>
<div id="pageContent">
<div class="sectionTitle">
    <fmt:message key="header.login.title" /></div>
    <stripes:messages />
    <stripes:layout-component name="contents" />
</div>
```

Na renderização de um pedido efectuado a uma qualquer página onde esteja definido que o seu componente seja a área de Contents, irá ser renderizado nessa área definida por a página de *layout*.

```
<stripes:layout-render name="/pages/layout/standard.jsp" title="news">
<stripes:layout-component name="contents">
<stripes:errors action="/shared/news/News.action" />
<stripes:form action="/shared/news/News.action" >
    <c:forEach items="${actionBean.newsList}" var="new">
        <div>${new}</div>
    </c:forEach>
</stripes:form>
</stripes:layout-component>
</stripes:layout-render/>
```

Esta Framework permite também a apresentação de erros a cada ciclo de vida de cada pedido, isto é, quando é efectuado um pedido, é possível devolver erros sobre a forma de uma lista de mensagens que irá ser apresentada no espaço definido pela *tag* <stripes:error/>.

No interface foi também utilizado Javascript, para efectuar e utilizar diversos widgets.

As bibliotecas utilizadas foram JQUERY<sup>33</sup> e DWR<sup>34</sup>. O JQUERY, é uma framework *open-source* que contém diversos *plugins*, e *widgets* associados, o que permite uma flexibilidade em termos de interoperabilidade entre o interface e a camada de *middleware*, possibilitando a apresentação de páginas Web com diversos *widgets*, (calendário, tabuladores, menus deslizantes, etc...).

O DWR foi a framework utilizada para efectuar as chamadas AJAX<sup>35</sup> aos serviços SCORM.

### 4.3 Desenvolvimento da camada de *middleware*.

A camada de *middleware* foi desenvolvida em Java recorrendo às Frameworks Spring para serviços, e JPA/Hibernate e JDBC para acessos à base de dados.

A Framework Spring permite a exposição de serviços como WebServices ou como serviços J2EE. Assim, mediante o tipo de serviço é possível deixa-lo acessível através de WebServices que funcionam sobre HTTP, ou como serviços J2EE, apenas possíveis de consumir por sistemas Java.

Na nossa arquitectura SOA, os serviços da plataforma foram expostos como serviços J2EE, já que o cliente/consumidor iriam ser Servlets, que são componentes base do J2EE.

Exemplo da exposição de um serviço:

```
@Service
public class ServiceExample implements IServiceExample {
    @Autowired
    private IDaoExample daoExample;

    public List<IModelExample> searchMessages() {
        return daoExample.searchMessages();
    }
}
```

---

<sup>33</sup> <http://jquery.com/>

<sup>34</sup> DWR – Direct Web Remoting, <http://directwebremoting.org>

<sup>35</sup> Ajax - acrônimo em língua inglesa de Asynchronous Javascript And XML.

Os serviços Spring são descritos em um ficheiro XML, onde se poderá configurar várias propriedades relativas ao serviço.

Exemplo:

```
<bean      name="/service/ServiceExample"      id="ServiceExampleBean"
class="pt.elearn.core.service.ServiceExample" singleton="true"/>
```

Neste exemplo, o serviço é configurado em modo *singleton*, onde irá ser apenas instanciado uma única vez no contexto da aplicação para todos os clientes.

É possível efectuar configurações relativamente às *datasources* (fontes de dados) da camada de dados, contudo a plataforma SKyLearn ao utilizar JPA com Hibernate, para acesso a camada de dados, foi possível utilizar a *datasource* definida pelo JNDI no servidor aplicacional.

Assim a framework de persistência JPA, e o JDBC utilizam o mesmo acesso à camada de dados, centralizando assim o esquema de *polling de connection*, deixando a gestão dessas componentes a encargo do *datasource*.

Exemplo de configuração do datasource:

```
<Resource      name="jdbc/eLEARN"      auth="Container"      scope="Shareable"
type="oracle.jdbc.pool.OracleDataSource"
driverclassname="oracle.jdbc.driver.OracleDriver"
factory="oracle.jdbc.pool.OracleDataSourceFactory"
connectionCachingEnabled="true"      connectionCacheName="ELearningCache"
connectionCacheProperties="{InitialLimit=5,
MinLimit=5,      MaxLimit=40,      InactivityTimeout=30,
ConnectionWaitTimeout=120,      PropertyCheckInterval=120,
ValidateConnection=true}"
url="jdbc:oracle:thin:@127.0.0.1:1521:XE"      user="ELEARN_SYSTEM"
password="ELEARN_SYSTEM" setDefaultAutoCommit="false"/>
```

Na especificação da unidade de persistência, efectuada através do ficheiro *persistence.xml*, é efectuada a ligação entre essa unidade e o recurso JNDI da *datasource*:

```

<persistence-unit name="ELEARNING_SYSTEM" transaction
type="RESOURCE_LOCAL">
<provider>org.hibernate.ejb.HibernatePersistence</provider>
<non-jta-data-source>java:/comp/env/jdbc/eLEARN</non-jta-data-source>

```

## 4.4 Implementação da interface

Apesar do aspecto simples da interface, o protótipo foi pensado de forma a contemplar formações empresariais, ou seja fora do contexto académico. Devido a esse objectivo, a plataforma apresenta poucas acções de gestão.

SKyLearn Technologies@ALPHA APPLICATION

Welcome: Paulo Hélio Alves | Logout

114336

October 2010

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Publicity courses | Enrollment courses

Name	Description	Methodology	Actions
Curso 1	Teste de Curso 1	Presencial	+
Curso 2	Teste de Curso 2	Presencial	+

• Candidatos a Belém reagem à recandidatura de Cavaco  
 Tue, 24 Oct 2010 22:24:00 GMT  
 Ainda antes da declaração oficial no CCB, o candidato presidencial Manuel Alegre afirmou não existir qualquer "novidade" na "recandidatura de direi...

• Conselho Económico e Social aprova parecer crítico ao Orçamento  
 Tue, 24 Oct 2010 21:58:00 GMT  
 O Conselho Económico e Social aprovou hoje, terça-feira, um parecer crítico sobre a proposta de Orçamento do Estado para 2011, mas defendendo que ...

• CDS-PP "estará empenhado" na reeleição de Cavaco Silva  
 Tue, 24 Oct 2010 21:34:00 GMT  
 O porta-voz democrata-cristão anunciou que o CDS-PP estará "empenhado" na reeleição de Cavaco Silva, e afirmou que seria desejável, caso vença a...

BUILT USING

Figura 21- Homepage após login na plataforma SKyLearn

A apresentação de dois cenários, através da listagem de cursos publicitados e de cursos onde o utilizador está matriculado, implica que o formando tenha o foco na aprendizagem, sem necessitar de procurar informações adicionais dispersas pela plataforma.



Na área de apresentação do conteúdo, o utilizador só deverá estar focalizado no conteúdo no qual estará a estudar.

De salientar que a API SCORM efectua pedido por meio de AJAX, utilizando a Framework DWR, que por sua vez irá invocar pedidos que efectuaram as acções relativas ao SCORM.

The screenshot displays the SKyLearn Technologies@ALPHA APPLICATION interface. At the top, it shows a user welcome message: 'Welcome, Paulo Hélio Alves | Logout'. A large green score '120231' is visible in the top right corner. The main content area is titled 'Play of the game' and contains text explaining the rules of golf, a photograph of a golf course, and a Creative Commons license notice. A sidebar on the right shows a calendar for October 2010 and a list of news items related to the 'Candidatura de Cavaco'.

Figura 22 - Área de Cursos na plataforma SKyLearn

No *middleware*, a Framework Spring, apresenta grandes vantagens, pois mesmo em ambientes que não sejam puramente J2EE, como acontece num servidor Tomcat, esta permite a disponibilização de componentes capazes de colmatar a falta de um servidor aplicacional J2EE.

A utilização do JPA/Hibernate, efectuando o mapeamento de algumas tabelas do modelo de dados para objectos Java, permite a utilização do modelo relacional da base de dados, como um modelo orientado a objectos. Contudo nas tabelas associadas ao SCORM, em que os tempos de acesso deverão ser baixos, estas Frameworks de ORM(Object-relation mapping), como é o caso do Hibernate, tem tempos de acesso muito elevados. Isto deve-se ao facto de efectuar uma abstracção do modelo de dados e a necessidade de efectuar um mapeamento da

entidade JAVA em tabelas do modelo de dados, assim como o *parsing* das *queries*, o que implica uma latência muito elevada.

Assim, no sentido de minimizar este problema, o acesso à base de dados por parte dos serviços SCORM é efectuada através de JDBC, onde é privilegiado o uso de *preparestatements*, tendo assim em conta os tempos utilizados pelo motor de base de dados para efectuar o *parsing* das *queries*.

Deste modo, reduzindo as camadas por onde passa um pedido, este é resolvido de uma forma mais rápida, contudo a questão de manutenção aplicacional e a transparência das camadas fica comprometida.

No desenvolvimento desta plataforma foi equacionada a questão da fiabilidade/facilidade versus velocidade, tendo-se optado por utilizar o conector JDBC em serviços que deverão ser necessariamente muito rápidos.

## Capítulo 5

### 5 Validação da Plataforma SKyLearn

A plataforma SkyLearn, foi sujeita a uma validação, com a finalidade de garantir que o objectivo proposto na especificação inicialmente definida, fosse totalmente conferido pelo utilizador final.

Assim, para a validação da plataforma SkyLearn, foi escolhido o método de inquirição a um grupo de utilizadores.

Esta validação baseou-se em sessões informais com um grupo de alunos da Escola Superior de Tecnologia e Gestão, do Instituto Politécnico de Bragança, que se voluntariaram para efectuarem testes experimentais sobre a plataforma, o que permitiu obter indicadores de qualidade, usabilidade e satisfação pelo utilizador final.

A resposta à validação foi realizada de modo formal, recorrendo ao método de recolha de dados, através de inquéritos, solicitados aos mesmos alunos referidos anteriormente.

A principal questão que era pretendida validar e analisar com este capítulo, é o impacto de uma arquitectura SOA numa plataforma de *e-learning*.

No âmbito da validação geral, apresentou-se também uma Framework de serviços, possíveis de utilização com o referencial SCORM.

No sentido de validar as questões subjacentes a esta dissertação, é apresentada uma análise qualitativa da plataforma SkyLearn, em fase de protótipo, e por último uma discussão dos resultados da validação, assim como da sua arquitectura.

## 5.1 Metodologia de investigação

Este desenvolvimento, integra-se na área da Tecnologias de informação, onde o conhecimento, os métodos de investigação, quer quantitativos quer qualitativos, ou ambos são o seu fundamento.

Os métodos de apreciação quantitativos, sendo métodos que se baseiam em modelos matemáticos, têm por vezes a desvantagem do desconhecimento da realidade envolta no âmbito de uma experiência, o que pode traduzir numa análise comprometida dos resultados expectáveis. Assim, os métodos qualitativos surgem com a necessidade colmatar as falhas dos modelos matemáticos em estudos, onde envolvem pessoas, em que a quantificação por vezes é comprometida.

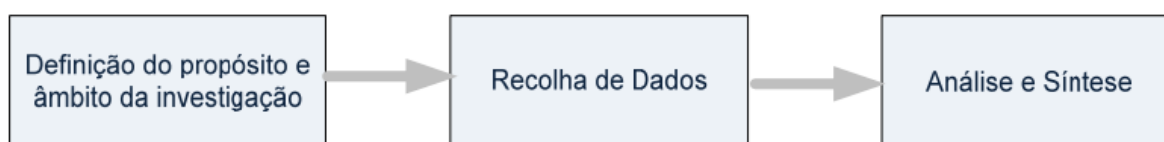
A vantagem de uma análise qualitativa em detrimento de uma análise quantitativa é a possibilidade de compreensão das pessoas e dos contextos que as rodeiam. A análise qualitativa tem como base epistemológica a investigação positiva, interpretativa e crítica, onde um conjunto de factores pode ser descrito por um conjunto de propriedades mensuráveis independentes do observador [Boland 1985].

Segundo Orlikowski [Orlikowski 1991], as principais metodologias a ter em atenção para análises de pesquisas em sistemas de informação são as seguintes:

- *Survey* – Método qualitativo, onde os dados recolhidos através de questionários ou entrevistas são analisados para posterior conclusão.
- *Experiencias em campo* – A utilização de métodos experimentais e reais dentro de ambientes controlados, como organizações. Tem como vantagem a iteração directa com o objecto de estudo, contudo como não é possível determinar todas as variáveis do sistema, não é possível garantir a repetição dos resultados da experiencia.
- *Case Study* – Investigação de fenómenos ocorridos dentro de contextos fechados, como organizações. É muito utilizado em Sistemas de Informação devido à sua focalização no problema dentro de um contexto, ou realidade, a grande desvantagem desta metodologia é estar muito centrada no objecto de estudo, o que impede a generalização de resultados.

Tendo em conta o âmbito desta dissertação, a análise que se pretende efectuar é predominantemente qualitativa.

O processo de investigação definido para esta dissertação tem como base a definição do seu propósito e do âmbito da investigação.



**Figura 23 - Etapas da Metodologia de Investigação [Alves, 2007]**

Após a definição do propósito e âmbito desta investigação, avaliar o protótipo SKyLearn, foi efectuada uma recolha de dados usando com instrumento de recolha um inquérito aos utilizadores da Plataforma SkyLearn. Por fim efectuou-se a análise dos dados dos inquéritos e apresentou-se os resultados de validação.

## **5.2 Análise do inquérito**

Para o estudo deste protótipo, a população considerada foram os alunos da unidade curricular de Multimédia do Curso de Informática de Gestão, administrado na Escola Superior de Tecnologia e Gestão de Bragança no ano de 2010, que voluntariamente se dispuseram para a validação da Plataforma SkyLearn.

A análise dos resultados inclui a utilização da plataforma SkyLearn, da sua arquitectura e de um conteúdo multimédia produzido com a norma SCORM.

Na plataforma não foram contempladas para análise, as ferramentas de gestão da plataforma, nem os conceitos de usabilidade e design gráfico da plataforma.

Assim para um universo de dezoito alunos obtivemos os seguintes resultados:

Na resposta à pergunta “Em média, quantas horas por dia utiliza a Internet”, obteve-se o resultado médio de 4.16h por dia.

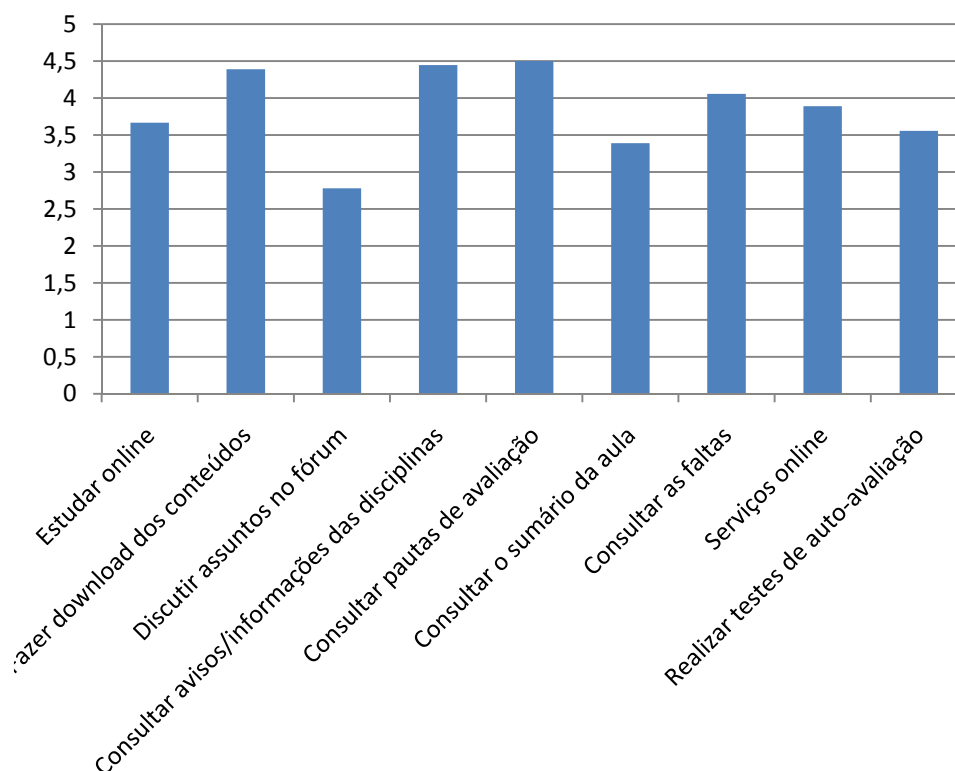
Para a análise da familiaridade com outra ferramenta de *e-learning* questionou-se os inquiridos acerca do tempo médio de utilização da plataforma IPB virtual, obtendo-se o resultado médio de 0.9h por dia.

Quanto ao *e-learning*, pretendia-se avaliar qual a informação e actividades mais importantes numa plataforma de *e-learning*, mais especificamente foram efectuadas as seguintes perguntas para análise de importância:

- Estudar online
- Fazer download dos conteúdos
- Discutir assuntos no fórum
- Consultar avisos/informações das disciplinas
- Consultar pautas de avaliação
- Consultar o sumário da aula
- Consultar as faltas
- Serviços online
- Realizar testes de auto-avaliação

Obtivemos os seguintes valores apresentados no seguinte gráfico:

## Características de uma plataforma de e-learning



**Figura 24 - Gráfico de Características de uma plataforma de e-learning**

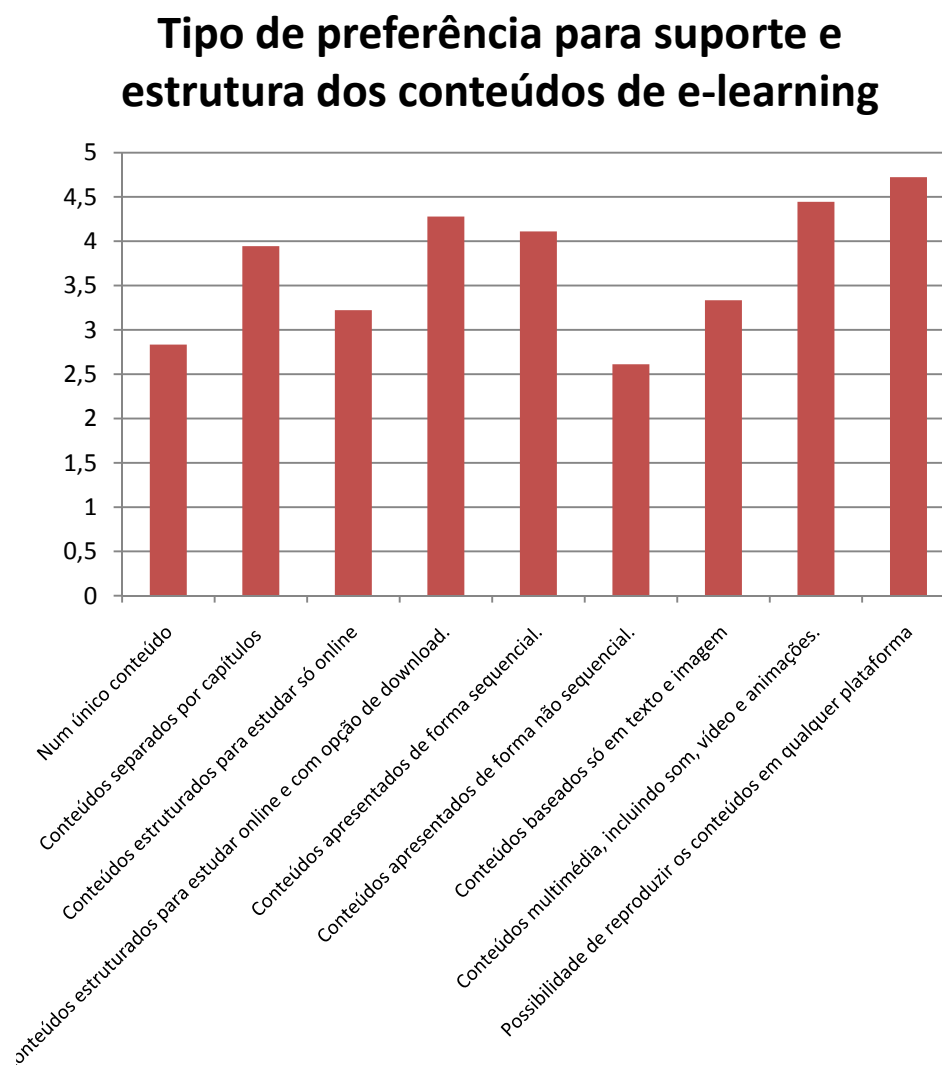
A maioria dos inquiridos utiliza as plataformas como complemento de estudo. Assim pode-se inferir que as actividades mais importantes enumeradas foram a realização de downloads dos conteúdos, consulta de pautas e consulta de avisos/informações das unidades curriculares.

Na questão da avaliação do tipo de conteúdos que os inquiridos preferiam como forma de disponibilização, foram apresentadas as seguintes opções:

- Num único conteúdo
- Em conteúdos separados por capítulos
- Conteúdos estruturados para estudar só online
- Conteúdos estruturados para estudar online e com opção de download.
- Conteúdos apresentados de forma sequencial.
- Conteúdos apresentados de forma não sequencial.
- Conteúdos baseados só em texto e imagem

- Conteúdos multimédia, incluindo som, vídeo e animações.
- Possibilidade de reproduzir os conteúdos em qualquer plataforma

Obtivemos os seguintes valores apresentados no seguinte gráfico:



**Figura 25 - Gráfico do tipo de preferência para suporte e estrutura dos conteúdos de e-learning**

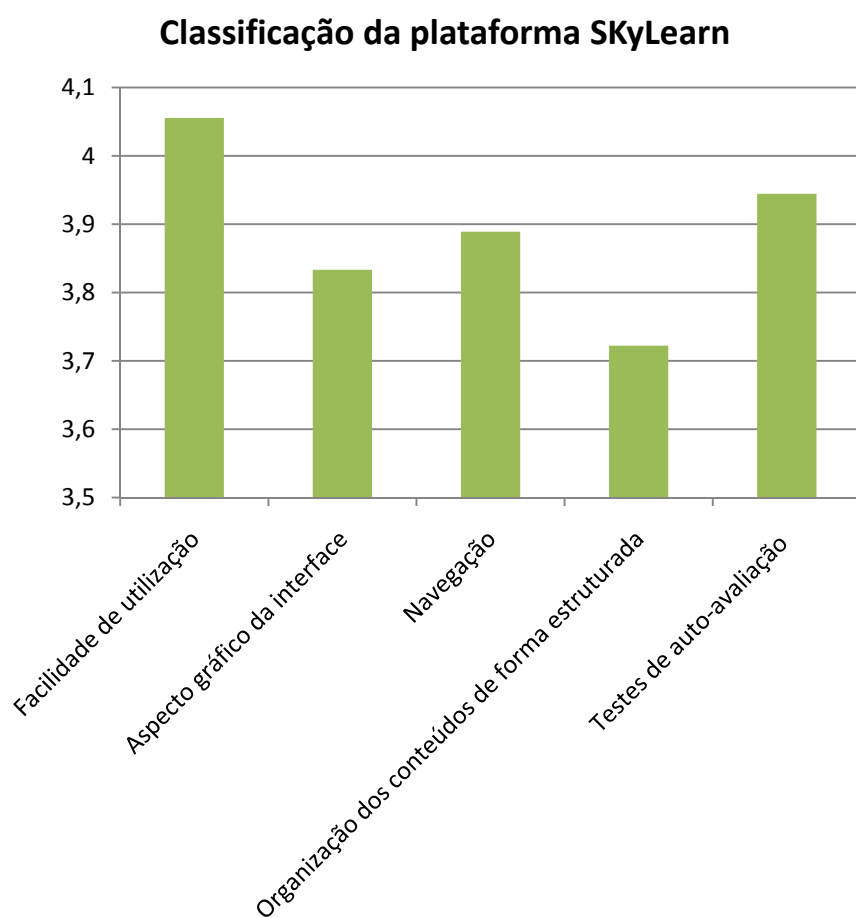
Verificou-se que os inquiridos pretendem conteúdos multimédia que sejam compatíveis com outras plataformas de *e-learning* e disponíveis tanto *online*, como com a possibilidade de efectuarem download para reprodução *offline*.

Na questão de classificação da plataforma SKyLine, foram efectuadas as seguintes perguntas:



- Facilidade de utilização
- Aspecto gráfico da interface
- Navegação
- Organização dos conteúdos de forma estruturada
- Testes de auto-avaliação

Obtivemos os seguintes resultados:



**Figura 26 - Classificação da plataforma SkyLearn**

Estes resultados sugerem a facilidade de utilização da plataforma como uma das suas mais-valias, seguido da integração dos testes de auto-avaliação suportados pelo referencial SCORM.

### 5.3 Síntese dos resultados

Neste capítulo, especificou-se a validação do protótipo da plataforma SkyLearn, testada e validada em termos de arquitectura, usabilidade e compatibilidade por um grupo de indivíduos, utilizadores de outras plataformas de *e-learning*.

A questão científica que orientou esta dissertação – Impacto da arquitectura SOA em ambientes de e-learning - foi validada através do desenvolvimento de uma plataforma de *e-learning* que suporta conteúdos produzidos segundo a norma SCORM 2004.

A validação do protótipo foi realizada através da realização de um inquérito ao grupo que efectuou os testes da plataforma. Com base nos resultados, destaca-se a importância de permitir utilizar conteúdos compatíveis com a norma SCORM, tornando portáteis os conteúdos, e de permitir reproduzir conteúdos multimédia (conteúdo de teste).

No entanto, como protótipo, prevê-se que necessite de alterações ao nível de interface gráfico e de alguns ajustes em termos de requisitos de forma a responder aos requisitos de negócio reais.

O SkyLearn foi desenvolvido com base em requisitos muito generalizados, de forma a apresentar-se o mais flexível possível. É por isso pertinente, que seja adaptado a um motor de regras de negócio, para que os cursos sejam avaliados e validados respondendo a essas regras, de modo a que, o mesmo conteúdo, desde que devidamente parametrizado, satisfaça as necessidades de diferentes organizações.

## Capítulo 6

### 6 Conclusão

Uma plataforma de e-learning potencia uma formação adequada a um público que tenha necessidades de potenciar os seus recursos, economizando os tempos dispendidos em necessidades formativas.

Assim, o e-learning surge como uma solução ágil, uma vez que a flexibilidade e a disponibilidade são as suas características mais importantes permitindo uma gestão eficaz de recursos, evitando assim desperdícios de tempo.

De notar que referimos o recurso Tempo, pois esta dimensão do ponto de vista económico é o recurso mais dispendioso que existe, sendo necessário efectuar o maior aproveitamento dele.

O e-learning é apenas mais uma ferramenta utilizada nas organizações que para fazer face aos contextos económicos, evitam o desperdício de recursos. A formação dos colaboradores nas organizações, apesar de dispendiosa na sua forma tradicional, é necessária, pois a sobrevivência de uma organização, advém da qualidade dos seus produtos ou prestação de serviços.

A diversidade de aplicações de e-learning existentes no mercado, enumerando algumas, como o Moodle, BlackBoard, Kall, Sakai, entre outras, tem como publico alvo a comunidade académica, ou seja, junto de uma plataforma de e-learning, juntam ferramentas de gestão dos cursos com algum grau de complexidade, vocacionadas para o âmbito académico.

Tendo em conta que o conceito de Intranets, sistemas de Gestão integrados com CRM, (Customer Relationship Management), se popularizou, as empresas de fornecimento de software apostam em aplicações do tipo “chave na mão”, ao passo que as plataformas de e-learning tem seguido um rumo diferente nas organizações não procurando a integração com

as diversas aplicações existentes. A gestão dos colaboradores e dos planos de formação são tarefas que são muitas das vezes realizadas nas aplicações CRM que tem que ser replicadas nas plataformas de e-learning. Assim, na questão central desta dissertação, a arquitectura proposta assenta numa total independência dos sistemas de aprendizagem, com a informação utilizada no cerne de uma organização, sendo apenas disponibilizado a informação fulcral e requerida pela organização.

As principais características da plataforma apresentada, a SkyLearn, explora a flexibilidade das arquitecturas orientadas a serviços aplicadas ao e-learning, tendo como base tecnológica a linguagem JAVA, uma camada de serviços, que disponibilizados através de variados ‘containers’, seja WebServices, sejam serviços J2EE, seja através de SPRING, permite a qualquer cliente invocar e efectuar os registos necessários para os cursos de e-learning produzidos com o padrão SCORM, de modo a serem reproduzidos em qualquer sistema.

Embora o tempo de observação e de validação da plataforma SkyLearn tenha sido curto, foi efectuado um inquérito de utilização a um grupo de alunos da unidade curricular de Multimédia do curso de Informática e Gestão.

Através da realização de um inquérito, observou-se que a grande maioria dos alunos utilizam as plataformas de e-learning para fazer download de conteúdos, tendo sido referido que a possibilidade de apresentar conteúdos multimédia foi uma mais-valia apresentada pela plataforma em estudo.

## **6.1 Trabalho Futuro**

Ao longo deste trabalho foram surgindo diversas ideias e aplicações, o que significa que a investigação é apenas o inicio de uma “caminhada” que não sabemos nunca onde iremos acabar, se algum dia conseguirmos dar por concluído o nosso trabalho.

As validações das questões de investigação que orientaram este trabalho foram baseadas numa análise qualitativa. Deste modo, não foi possível, por questões de limitação de tempo, efectuar estudos mais exaustivos que permitissem a formulação de teorias científicas mais assertivas. Assim propõe-se para um trabalho futuro um estudo mais alargado no âmbito das arquitecturas de software, de forma a contemplar novas correntes e estilos de modo a melhorar as teorias apresentadas.

Apesar do protótipo SKyLearn ser uma plataforma de e-learning com funções basilares, abre caminho ao desenvolvimento de novos campos na área do e-learning, como as ferramentas de autoria e a tutoria de cursos.

A integração com ferramentas de redes sociais, como o Facebook e Twitter, são campos que não devem ser descurados, uma vez que os alunos são utilizadores das redes sociais, o que induzirá que um trabalho avançado nesse campo traria uma mais-valia sobre a investigação de integração em aplicações online como uma “3rd party”.

Para além das propostas apresentadas, a área de formação em e-learning é uma área emergente devido à filosofia de partilha de conhecimento. Assim, a investigação nesta dissertação contribui para a apresentação de novas e melhores soluções, no sentido da integração do e-learning em sistemas autónomos e independentes, respondendo assim aos constantes desafios das organizações.

## **6.2 Conclusão**

Tendo como base um padrão de e-learning, o referencial SCORM, e o conceito de arquitectura de software orientada a serviços, SOA, o foco do estudo baseou-se na especificação de uma arquitectura orientada a serviços, tendo como resultado a elaboração de protótipo, o SKyLearn.

Este estudo cumpriu os seus objectivos propostos, pois permitiu a elaboração de uma plataforma de e-learning, onde as implementações do referencial SCORM foram conseguidas através da invocação de serviços, independentes da plataforma.

A centralização das regras de negócio da plataforma num “core” disponibilizado através de serviços a um cliente, garante a independência da camada de middleware do cliente, assumindo assim as virtudes e mais-valias de uma arquitectura SOA.

# Bibliografia

F. Neumann, R. Geys, SCORM and the Learning Grid, Friedrich-Alexander-Universität Erlangen-Nürnberg, 2005

IEEE, Systems and Software Engineering - Recommended Practice for Architectural Description of Software-Intensive Systems. *ISO/IEC 42010 IEEE Std 1471-2000 First edition*, c1-24, 2007

J. Ward, P. Griffiths, "Strategic Planning for Information Systems", John Wiley and Sons Ltd, 2002

L. Bielawski e D. Metcalf., Blended eLearning, Integrating Knowledge, Performance Support, and Online Learning, HRD Press, Inc. Amherst, Massachusetts, 2003.

L. Osterweil, *Software Engineering Techniques*. Technical report. Rome, Italy: NATO Science Committee, 2007

Lucena, Novas Tecnologias no E-learning: Desafios e Oportunidades para o Design. São Paulo, [http://www.abed.org.br/revistacientifica/Revista\\_PDF\\_Doc/2003\\_Novas\\_Tecnologias\\_Beto\\_Lucena.pdf](http://www.abed.org.br/revistacientifica/Revista_PDF_Doc/2003_Novas_Tecnologias_Beto_Lucena.pdf), 2003

M. Hammer; C. James," Reengineering the Corporation: A Manifesto for Business Revolution". (1st ed.) New York, NY: HarperBusiness, 1993

M. Rosenberg "E-Learning: Strategies for Delivering Knowledge in the Digital Age, 2001

N. Rozanski et al., *Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives*. Addison-Wesley Professional, 2005

P. Alves, E-generation: especificação de uma arquitectura para Intranets educacionais baseada em agentes, Tese de Doutoramento, Universidade do Minho, 2007

P. Dewayne, Foundations for the study of software architecture, SIGSOFT, Software Engineering Notes, 17(4):408211;52, 1992.

R. Boland, Phenomenology: A Preferred Approach to Research in Information Systems, in *Research Methods in Information Systems*, E. Mumford, R.A. Hirschheim, G. Fitzgerald, and T. WoodHarper (eds.), pp. 193-201, NorthHolland, Amsterdam, 1985

W. Orlikowski, Studying Information Technology in Organizations: Research Approaches and Assumptions *Information Systems Research*, 2, 1, 1-28, 1991

W. Rice, *Moodle, E-Learning Course Development*, PACKT Publishing, Birmingham, 2006

# **Apêndice A**

## **Questionário**



## Inquérito sobre a utilização do SKyLearn

Este inquérito tem o objectivo de avaliar a utilização da plataforma de e-learning SKyLearn em estudo na Tese de Mestrado em Sistemas de Informação a decorrer na ESTiG-IPB.

**É importante que a sua resposta traduza com sinceridade a sua opinião.**

**Os dados são confidenciais.**

Ano curricular em que se encontra matriculado: \_\_\_\_

Curso: \_\_\_\_\_

Sexo: F  M

Ano de nascimento: \_\_\_\_\_

Data de preenchimento: 12 / 10 / 2010

### I - E-learning

1. Em média quantas horas por dia utiliza a Internet: \_\_\_\_\_ horas
2. Em média quantas horas por dia utiliza a plataforma IPB.Virtual: \_\_\_\_\_ horas
3. Indique o grau de importância das seguintes características de uma plataforma de e-learning:

Muito Reduzida

Muito Elevada



Características de uma plataforma de e-learning	1	2	3	4	5
Estudar online					
Fazer download dos conteúdos					
Discutir assuntos no fórum					
Consultar avisos/informações das disciplinas					
Consultar pautas de avaliação					
Consultar o sumário da aula					
Consultar as faltas					
Serviços <i>online</i>					
Realizar testes de auto-avaliação					

4. Indique o tipo de preferência para suporte e estrutura dos conteúdos de e-learning:

Conteúdos	Muito Reduzida			Muito Elevada	
	1	2	3	4	5
Num único conteúdo					
Em conteúdos separados por capítulos					
Conteúdos estruturados para estudar só online					
Conteúdos estruturados para estudar online e com opção de download.					
Conteúdos apresentados de forma sequencial.					
Conteúdos apresentados de forma não sequencial.					
Conteúdos baseados só em texto e imagem					
Conteúdos multimédia, incluindo som, vídeo e animações.					
Possibilidade de reproduzir os conteúdos em qualquer plataforma					

5. Indique qual a importância do uso de **standards** no **e-learning**.

---



---



---

6. Já ouviu falar do referencial **SCORM**?

Sim  Não

7. Caso tenha respondido **SIM**, por favor indique qual a importância do **SCORM** para o e-learning.

---



---



---

## II – Plataforma de e-learning SKyLearn



8. Classifique a plataforma SKyLearn quanto a:

Plataforma SKyLearn	Nada relevante			Extremamente relevante	
	1	2	3	4	5
Facilidade de utilização					
Aspecto gráfico da interface					
Navegação					
Organização dos conteúdos de forma estruturada					
Testes de auto-avaliação					

9. Indique **três** pontos positivos da plataforma SkyLearn.

---

---

---

10. Apresente duas sugestões que possam melhorar a estrutura da plataforma de e-learning SKyLearn.

---

---

---

11. Descreva a sua experiência na utilização do e-learning.

---

---

---

Obrigado pela sua colaboração.