

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/305463108>

# Integração do Moodle com Sistemas de Gestão Académica

Article · January 2014

---

CITATIONS

0

READS

1,461

2 authors, including:



Edson Livongue

University of Coimbra

1 PUBLICATION 0 CITATIONS

[SEE PROFILE](#)

# Integração do Moodle com Sistemas de Gestão Académica

Abel Zacarias

Departamento de Engenharia Informática  
Universidade de Coimbra  
Coimbra Portugal  
Email:uc2013102031@student.uc.pt

Edson C. F. Livongue

Departamento de Engenharia Informática  
Universidade de Coimbra  
Coimbra Portugal  
Email: uc2012127191@student.uc.pt

**Resumo -** Este trabalho sugere um modelo de integração do moodle com um sistema de gestão académica, fundamentado no modelo de arquitetura de serviços, principalmente os Web service, demonstrando através da extensão de um modelo já proposta de integração do moodle com sistemas de gestão académicas. Visando assim permitir uma integração que não seja só unidirecional mas também bidireccional em os sistemas em causas. Ao fim é apresentado um novo modelo teórico que permita a aplicação desse tipo de integração bidireccional.

**Keywords—**Moodle; Integração; Sistemas de Gestão Académica, ambientes virtuais de aprendizagens.

## INTRODUÇÃO

A integração é um trabalho de sincronização de dados entre os dois sistemas.

A sincronização consiste em mapear um par correspondente de cada registo nos dois sistemas. Digamos assim, que cada registo na aplicação acadêmica deve ter uma cópia no sistema do Moodle ou vice-versa.

A maioria das demandas de integração, visam integrar os seguintes dados:

1. Utilizador.
2. Curso;
3. Disciplina;
4. Matrícula;
5. Nota

Contudo, a integração de aplicações e sistemas não é uma tarefa fácil, comenta Moul (2003) isto porque, a maioria dos fornecedores apenas possibilitam integração dos seus próprios sistemas e aplicações relacionadas, tornando assim difícil o processo de integração com sistemas diferentes.

## MOTIVAÇÃO

Como em qualquer empresa, as instituições de ensino também padecem da falta de integração entre os sistemas, principalmente com a chegada e a proliferação do uso dos Ambientes Virtuais de Aprendizagem (AVA). A integração do AVA em instituição de ensino deve, necessariamente, ser com um Sistema de Gestão Académica SGA.

Este trabalho é um projecto da cadeira de Aplicações Telemáticas que tem como objectivos apresentar um estudo para a integração de um ambiente virtual de aprendizagem,

especificamente o Moodle com um Sistema de Gestão Académica.

## OBJETIVOS

Fazer um estudo para a integração de um ambiente virtual de aprendizagem, especificamente o Moodle com um Sistema de Gestão Académica.

Propõe-se apresentar uma solução de integração entre o AVA Moodle e um Sistema de Gestão Académica seguindo o modelo sugerido por (MOURA e BERNARDINO, 2010) que propõem integração utilizando os benefícios que SOA (*Service-Oriented Architecture*) nos apresenta, através da construção de Web Service, bem como identificar a adequada de integração entre Sistemas de Gestão Académica.

A Plataforma Moodle é um sistema de gestão de ensino online. É uma ferramenta de trabalho do professor. Em um ambiente institucional de ensino, o seu uso atende as demandas da gestão de atividades pedagógicas do corpo docente. No entanto, não atende as demandas de gestão académica e administrativa.

### A. 2.1- Apresentação geral do Estado da Arte

Pretende-se a partir deste ponto demonstrar os principais conceitos e marcos teóricos relacionados com o estudo da integração entre Sistemas de Gestão Académica e ambientes virtuais de aprendizagem, especificamente o Moodle.

Torna-se relevante, quando se fala da integração de sistemas abordar aspectos sobre Arquitectura Orientada a Serviço (SOA), assim, mostra-se alguns aspectos fundamentais a cerca do assunto, posteriormente fazer-se-á a apresentação de conceitos fundamentais sobre Web Service. Logo após explica-se conceitos que envolve, Sistema de Gestão Académica (SGA), Ambiente Virtual, Moodle. Por fim apresentaremos o *Modelo conceitual de Integração* entre o Moodle e o SGA proposto por (MOURA e BERNARDINO, 2010).

### B. 2.2- Arquitectura Orientada a Serviço (SOA)

SOA – do inglês *Service-Oriented Architecture* (Arquitectura Orientada a serviço), é um paradigma de desenvolvimento de aplicações cujo objectivo é criar módulos funcionais chamados serviços com baixo acoplamento, permitindo a

reutilização de código (SAMPAIO, 2006). SOA estabelece um modelo arquitectónico que visa a aprimorar a eficiência, a agilidade e a produtividade de uma empresa através de serviços (ERL, 2009).

SOA é na verdade um novo jeito de desenvolver aplicações. Suas origens advém da orientação a objectos tradicional, a diferença entre esses dois paradigmas consiste no fato que na *orientação a objectos* o que se deseja quando se cria uma classe é caracterizar um objecto em si através das propriedades e suas ações através dos métodos. Na *orientação a serviço* o foco principal não é um objeto em si, mas sim o que o objeto(s) é (são) capaz de gerar para determinadas necessidades. Por exemplo, em nosso estudo de caso necessitaremos que as informações do registo dos alunos e professores que estão presentes em um sistema de gestão acadêmica também se tornem disponíveis no sistema Moodle, no entanto ao invés de nos preocuparmos com o objeto aluno, enviamos solicitações através de SOA das informações do aluno sem nos preocupar como ele foi desenvolvido. A computação *orientada a serviço* representa, portanto, uma nova geração da plataforma da computação distribuída.

Um serviço, do ponto de vista da arquitetura SOA, é uma função de um sistema computacional que é disponibilizado para outro sistema. Um serviço deve funcionar de forma independente do estado de outros serviços, exceto nos casos de serviços de processos (process services), e deve possuir uma interface bem definida. Normalmente, a comunicação entre o sistema cliente e aquele que disponibiliza o serviço é realizada através de web services. (Wiki 2013).

### 2.2.1- Elementos de SOA

Os elementos essenciais para alcançar o sucesso em SOA são: Visão conceitual de SOA, serviços, tecnologia, governança SOA e estratégia SOA, indicadores e cultura e comportamento [Marks e Bell, 2006].

### 2.2.2 Visão conceitual de SOA

SOA é uma proposta de como as funcionalidades de TI (Tecnologia de Informação) podem ser planificadas, projetadas e entregues como serviços de negócio modulares para atingir um determinado benefício de negócio. Para alcançar os objetivos SOA, considerações organizacionais e de comportamento devem ser entendidas e alteradas primeiro, de forma gradual e ao longo do tempo.

### 2.2.3- Serviços

Os serviços são os artefactos centrais de SOA, os ativos primários de arquitetura. Juntamente com os serviços deve ser definido um modelo de projeto de serviços que assegure reutilização, interoperabilidade e integração por todos os processos de negócio e plataforma de tecnologias.

### 2.2.4- Tecnologia

A tecnologia é essencial para apoiar e alcançar SOA, mas iniciativa SOA não se resume em uma tecnologia. A tecnologia deve ser disponibilizada para atingir os seguintes objetivos:

- Permitir que os serviços operem de forma confiável e segura apoiando os objetivos do negócio;

- Permitir evoluir na arquitetura de TI existente, possibilitando, por exemplo, que sistemas legados possam ser utilizados em processos de negócio a fim de apoiar os objetivos SOA. Em muitas organizações sistemas legados são os principais contribuidores de serviços para o SOA.

## 2.3- Web Services

Devido aos avanços tecnológicos, a popularização da Internet e a evolução das redes de computadores, resultando no surgimento de aplicações distribuídas, cada vez mais aumenta a necessidade de compartilhamento de informações. Em outras palavras, aumenta a necessidade de interoperabilidade entre sistemas. Surgem então os Web Services, como solução para uma melhor comunicação entre sistemas distribuídos completamente distintos.

Assim sendo, o Web Services é uma ferramenta com bastante potencial e ideal actualmente para a partilha de informação entre sistemas. A comunicação entre os serviços é padronizada, possibilitando a independência de plataforma e de linguagem de programação. Por exemplo, um sistema de desenvolvido em Java e executado em um servidor Linux pode acessar, com transparência, um serviço feito em .Net sendo executado por um servidor Microsoft.

### 2.3.1- Definição

Segundo Menéndez, há uma definição bastante simples para um Web Services: É uma aplicação que aceita solicitações de outros sistemas através da Internet. Segundo James Snell, Web Services são interfaces acessíveis de rede, para as funcionalidades da aplicação, que utilizam em sua construção tecnologias padrões da Internet.

Web Service é um componente de software que se comunica usando protocolos abertos e que usa como base o padrão XML. Sua principal funcionalidade é prover integração entre aplicações distintas através da *World Wide Web* utilizando componentes, especificações e padrões tecnológicos que são bem definidas no mercado de TI como um todo.

### 2.3.2- Arquitetura

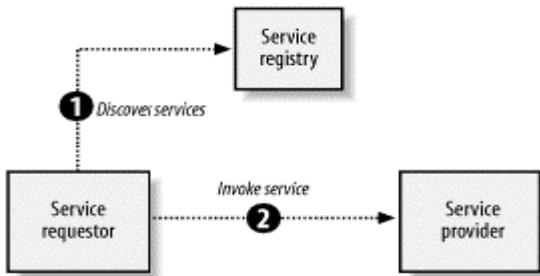
A arquitetura dos Web Services é baseada na interação de três personagens: Provedor de Serviços, Consumidor de Serviços e Registro dos Serviços. A interação destes personagens envolve as operações de publicação, pesquisa e ligação. Vejamos a definição de cada uma destas funções [KREGER2001]:

- ✓ **Provedor de serviços:** o provedor de serviços é a entidade que cria o Web Service. Ele disponibiliza o serviço para que alguém possa utilizá-lo. Mas, para que isto ocorra, ele precisa descrever o Web Service em um formato padrão, que seja compreensível para qualquer um, que precise usar esse serviço, também publicar os detalhes sobre seu Web Service em um registro central que esteja disponível.
- ✓ **Consumidor de serviços:** qualquer um que utilize um Web Service criado por um provedor de serviços é chamado de consumidor de serviços. Este conhece a funcionalidade do Web Service, a partir da descrição

disponibilizada pelo provedor de serviços, recuperando os seus detalhes através de uma pesquisa sobre o registo publicado. Através desta pesquisa, também o consumidor de serviços pode obter o mecanismo para ligação com este Web Service.

- ✓ **Registro dos serviços:** Um registo de serviços é a localização central onde o provedor de serviços pode relacionar seus Web Services, e no qual um consumidor de serviços pode pesquisá-los. O registo dos serviços contém informações como detalhes de uma empresa, quais os serviços que ela fornece e a descrição técnica de cada um deles.

Portanto, o provedor de serviços define a descrição do serviço para o Web Service e publica esta para o consumidor de serviços no registo de serviços. O consumidor de serviços utiliza a descrição do serviço publicada para se ligar ao provedor de serviços e invocar ou interagir com a implementação do Web Service.



#### Web Service Roles [OREILLY], p.11.

Este modelo de integração representa uma alternativa a outros modelos conhecidos, como as chamadas de procedimento remoto (RPCs). Mesmo não tendo o mesmo desempenho e maturidade, o uso de Web services é uma excelente alternativa quando há necessidade de baixo acoplamento entre as aplicações.

O grupo de trabalho do W3C responsável pela arquitetura dos Web services os define assim: “Um Web service é um aplicativo projetado para suportar interoperabilidade entre máquinas através de uma rede. Deve possuir uma interface apresentada em WSDL (Web Services Description Language), que descreve como outros sistemas podem interagir com ele através da utilização de mensagens SOAP (Simple Object Access Protocol). Essas mensagens são tipicamente transportadas usando HTTP (Hypertext Transfer Protocol) com uma formatação XML (Extensible Markup Language) e em conjunto com outras convenções relacionadas à Web”.



[Ihttp://www.nce.ufsj.br/conceito/artigos/2005/01p1-1.htm](http://www.nce.ufsj.br/conceito/artigos/2005/01p1-1.htm)

#### 2.3.3- Classificação dos componentes dos webservices

Curbera, Mukhi e Weerawarana (2001) afirmam que a arquitetura de webservices é uma plataforma que tem por objetivo integrar aplicações na Web. Os autores usam como critério de classificação a funcionalidade das especificações. De acordo com este critério, a plataforma de serviços Web é classificada em quatro conjuntos de especificações que têm em comum o uso da linguagem XML. Esses conjuntos são:

1. Descrição de serviços - utilizada para definir as operações, mensagens e os tipos de dados de um serviço, também mantendo as informações sobre como acceder os serviços;
2. Publicação e descoberta de serviços - contém os protocolos que possibilitam a localização da descrição dos serviços.
3. Descrição de composição de serviços - contém os modelos e linguagens utilizadas para descrever como se dará a interação dos serviços;
4. Protocolos de comunicação - utilizados para definir, estabelecer e manter a comunicação entre as aplicações, também contendo a descrição dos formatos das mensagens utilizadas no estabelecimento da comunicação entre aplicações.

Os quatro conjuntos apresentam uma característica comum, que é o uso da linguagem XML, padrão aberto e extensível que permite a integração e troca de dados entre componentes distintos, garantindo a interoperabilidade necessária para a arquitetura.

#### 2.3.4- Descrição de serviços

A especificação WSDL (Web Services Description Language), atualmente na versão 2.0, é uma linguagem baseada em XML que descreve de forma padronizada e independente de plataforma como e onde os serviços podem ser conectados e utilizados através da rede.

Um documento WSDL representando um contrato entre o provedor de serviços e seus clientes e definindo os serviços como uma coleção de portas na rede. Os documentos WSDL apresentam uma dicotomia entre a definição da mensagem (parte abstrata) e sua implementação (parte concreta), o que permite o reuso das definições das mensagens e port types. Um documento WSDL é composto por sete elementos XML

que representam as partes abstrata e concreta. Na parte abstrata temos quatro elementos:

1-Types - fornece a definição dos tipos de dados para descrever as mensagens trocadas entre aplicações, normalmente representadas por um documento XSD (XML Schema Definition);

2- Message - representa a informação que será trocada através das definições dos tipos de dados;

3-Porttype - é um conjunto de operações suportadas por um ou mais end points, onde cada operação se refere a uma mensagem de entrada, saída ou erro;

4-operation - descreve a ação suportada pelo serviço; Na parte concreta temos os outros três elementos:

1-Binding - define uma especificação de protocolo e formato de dados para as mensagens definidas em um port type;

2-Port - é um end point, representa a combinação de um binding e um endereço de rede;

3-Service - é uma coleção de portas. Cada elemento port se relaciona com um elemento binding particular, indicando qual interface e qual protocolo de comunicação estão sendo utilizados nessa implementação.

### **2.3.5-Publicação e descoberta de serviços Web**

A especificação UDDI, atualmente na versão 3.0, especifica mecanismos para a publicação, descoberta e integração de serviços. Tais mecanismos definem as estruturas de dados necessárias para sua descrição e classificação, bem como uma interface baseada no protocolo SOAP que permite o acesso a essas informações.

Resumidamente, o registo de serviços UDDI possui dois tipos de cliente. O primeiro envolve as aplicações que desejam publicar serviços e suas interfaces, o segundo tipo envolve os clientes que desejam obter e se ligar a serviços Web.

Conceitualmente, o protocolo UDDI apresenta três papéis, representados sob a forma de XML Schemas (NEWCOMER, 2002). Os papéis são:

1-Páginas Brancas - contém identificadores sobre o contato técnico do serviço oferecido;

2-Páginas Amarelas - contém informações genéricas sobre os tipos e localização dos serviços disponíveis;

3-Páginas Verdes - contém informações técnicas sobre um determinado serviço Web.

A especificação UDDI define quatro estruturas de dados, também descritas como documentos XML, onde cada elemento descreve o tratamento dado ao serviço Web, (KREGER, 2001; 2002; NEWCOMER, 2002).

1-BusinessEntity - estrutura de alto nível (páginas brancas) que contém, para cada serviço, as informações (nome, categoria, identificadores, entre outros) sobre a organização que publicou o serviço Web;

2- BusinessService - contém informações descritivas sobre serviços Web (páginas amarelas), tais como nome e descrição do serviço publicado;

3- BindingTemplate - contém informações técnicas sobre o serviço Web tais como forma de acesso e endereços dos pontos acesso ao serviço (páginas verdes);

4- TModel - mecanismo usado para a troca de definições abstratas (metadados) sobre um serviço Web, contém as descrições do serviço Web, opcionalmente aponta para documentos WSDL.

### **2.3.6-.Descrição.de.composições.de.serviços.Web**

A composição de serviços Web se relaciona com a automação de processos de negócio e requer uma especificação que controle a troca de mensagens entre os serviços Web.

As especificações que descrevem a composição de serviços Web devem seguir uma série de princípios:

1-Estar em consonância com as demais especificações da W3C;

2-Permitir a composição de serviços mais complexos a partir de serviços Web mais simples;

3-Ser capaz de manter alta independência entre serviços com baixo acoplamento;

4-Facilitar a implementação de processos de negócio e a comunicação entre seus serviços; definir serviços obrigatórios e opcionais;

5-Indicar a ordem de execução e as rotas que serão seguidas no caso de sucesso ou falha na execução de um dado serviço Web.

Inicialmente, existiam duas especificações de descrição de composição de serviços:

A WSFL (Web Services Flow Language) e a XLANG. Entretanto, elas foram substituídas pela especificação BPEL4WS (Business Process Execution Language for WebServices). BPEL4WS une características das linguagens XLANG e WSFL tornando-se mais completa que suas antecessoras (ANDREWS et al., 2003).

A especificação BPEL4WS, atualmente na versão 1.1, possui elementos que possibilitam a descrição de workflows baseados na interação de serviços Web. A linguagem também define um modelo e uma gramática para a descrição do comportamento de um processo de negócio e ainda define mecanismos para o tratamento de exceção (ANDREWS et al., 2003).

### **2.3.7-Protocolos de comunicação**

A especificação SOAP (Simple Object Access Protocol) é um protocolo de transporte que rege a troca de mensagens entre aplicações em ambientes distribuídos e descentralizados. Seu conteúdo é composto por informações e estruturas de dados (KREGER, 2001; COYLE, 2002).

A especificação SOAP 1.2 contém três partes principais:

1- Modelo de empacotamento - envelope SOAP - define o conteúdo da mensagem SOAP;

2- Mecanismo de serialização - conjunto de regras de codificação - define os tipos de dados utilizados na aplicação;

3- Mecanismo de comunicação – convenção - define as chamadas e respostas através de procedimentos remotos.

Toda mensagem SOAP é um documento XML. Ela obrigatoriamente contém os elementos <Envelope> e <Body> e opcionalmente os elementos <Header> e <Fault>.

1- O elemento <Envelope> é a raiz do documento XML e representa a mensagem propriamente dita;

2- O elemento <Body> contém a carga de informações (operações e parâmetros) que são entregues ao destinatário da mensagem. Este elemento, de acordo com a especificação SOAP, pode conter um elemento <Fault>, que, quando presente, pode ser utilizado no processamento de falhas do serviço Web. O elemento <Fault> descreve erros de chamada de métodos remotos ou mantém informações acerca do tipo de erro. Portanto, ele conta com os elementos <FaultCode>, <FaultActor>, <FaultString> e <Detail>;

3- O elemento <Header> expande uma mensagem SOAP. Ele define algumas características opcionais e acordos negociáveis entre as partes; seu conteúdo deve ser aceito pelas aplicações que estiverem se comunicando.

A especificação SOAP, a rigor, não depende do protocolo HTTP. Apesar de ser a combinação mais amplamente utilizada nos dias de hoje, a SOAP pode ser utilizada por outros protocolos da camada de transporte como, por exemplo, FTP, SMTP (COYLE,2002).

Newcomer (2002) considera a especificação SOAP um tipo de extensão do protocolo HTTP, pois ela envia e recebe mensagens XML através das operações requisição e resposta do protocolo HTTP. Para que este processo ocorra, é necessário utilizar um servidor de páginas que seja capaz de atuar como um processador SOAP. O exemplo abaixo ilustra uma mensagem SOAP sendo enviada através do método POST do protocolo HTTP v 1.1

.

POST /HoplitasBooks HTTP/1.1

Host: localhost

Content-Type: text/xml; charset="utf-8"

Content-Length: nnnn

SOAPAction: http://equipe.nce.ufrj.br/serra

<soapenv:Header>

<soapenv:criptograph>

soapenv:mustUnderstand="0" xsi:type="xsd:string">yes

</soapenv:criptograph>

<soapenv:priority>

soapenv:mustUnderstand="0" xsi:type="xsd:string">high

</soapenv:priority>

</soapenv:Header>

<soapenv:Body>

<validateInfo>

soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

<Username xsi:type="xsd:string">SergioSerra</Username>

```
<BirthDate xsi:type="xsd:string">03/02/1965</BirthDate>
<IDCardType xsi:type="xsd:string">CPF</IDCardType>
<IDCardNumber
xsi:type="xsd:string">45608</IDCardNumber>
</validateInfo>
</soapenv:Body>
</soapenv:Envelope>
```

Exemplo - Mensagem SOAP contendo os elementos <Envelope>, <Header> e <Body>. No exemplo acima, o campo SOAPAction é utilizado por servidores de páginas ou firewalls para filtrar ou rotear uma mensagem SOAP. Este campo pode ser nulo ou conter o nome de algum método SOAP. O cabeçalho da mensagem se refere a uma possível prioridade de execução, que ainda não foi estabelecida. Ele faz uso do atributo mustUnderstand que exige do receptor o suporte a transação solicitada pelo emissor da mensagem. Caso o receptor da mensagem não suporte a transação, ele originará uma mensagem SOAP de falha.

O corpo da mensagem possui uma operação definida pelo atributo ValidateInfo e um pequeno conjunto de parâmetros definidos pelos elementos <UserName>, <Birthdate>, <IDCardType>, <IDcardNumber>.

### 3- AMBIENTES VIRTUAIS DE APRENDIZAGEM

Por ambientes podemos entender tudo aquilo que envolve pessoas, natureza ou coisas, objetos técnicos. Já o virtual vem do latim medieval *virtualis*, derivado por sua vez de *virtus*, força, potência. Muitas pessoas utilizam a expressão virtual que designar alguma coisa que não existe. Lévy (1996) em seu livro O que é o virtual? Esclarece que o virtual não se opõe ao real e sim ao atual. Virtual é o que existe em potência e não em ato. Citando o exemplo da árvore e da semente, Lévy explica que toda semente é potencialmente uma árvore, ou seja, não existe em ato, mas existe em potência. Ao contrário dos exemplos citados no parágrafo anterior o virtual faz parte do real, não se opondo a ele. Por isso nem tudo que é virtual necessariamente se atualizará. Ainda no exemplo da semente, caso um pássaro à coma a mesma jamais poderá vir a ser uma árvore.

Transpondo essa ideia para a realidade educacional podemos aferir que quando estamos interagindo com outros sujeitos e objetos técnicos construindo uma prática de significação podemos tanto virtualizar quanto atualizar este processo. Vale destacar que a atualização é um processo que parte, quase sempre, de uma problemática para uma solução já a “virtualização passa de uma solução dada a um (outro) problema”. (LEVY,1996, p. 18). Logo, virtualizar é problematizar, questionar é processo de criação.

Ainda segundo Lucena (2013) Ma Ambiente Virtual de Aprendizagem (AVA) é um software que utiliza técnicas pedagógicas bem definidas para o ensino através de ferramentas online, como a internet. Um outro termo utilizado para o AVA é o *LMS* (*Learning Management System*), estes dois termos são considerados sinônimos. Muitos autores utilizam também o termo EAD (Educação à Distância) para

definir um AVA. No entanto, existe uma sutil diferença entre os termos:

O EAD engloba todos os meios que distanciam as figuras de professores e alunos utilizando qualquer meio, que não presencial, para a tarefa de ensino-aprendizagem, tais como: CD, DVD, TV, correspondências e também através da internet;

No AVA, a distância física entre alunos e professores é somente através da Internet.

Neste sentido pode-se afirmar que um ambiente virtual é um espaço fecundo de significação onde seres humanos e objetos técnicos interagem potencializando assim, a construção de conhecimentos, logo a aprendizagem. Então todo ambiente virtual é um ambiente de aprendizagem? Se entendermos aprendizagem como um processo sócio técnico onde os sujeitos interagem na e pela cultura sendo esta um campo de luta, poder, diferença e significação, espaço para construção de saberes e conhecimento, então pode-se afirmar que sim.

Utilizaremos o termo AVA em nosso trabalho, visto que utilizaremos somente a internet como meio de auxílio ao Ensino-Aprendizagem.

### 3.1- Vantagens dos Ambientes Virtuais de Aprendizagem

Um Ambiente Virtual de Aprendizagem pode ser vantajosamente inserido em um dispositivo pedagógico para estender o espaço da aula, mas ele é imprescindível quando se trata de Educação a Distância. Ele pode oferecer funcionalidades básicas ou sofisticadas.

Algumas vantagens dos AVAs na perspectiva de vários autores como Gugleby (2002) e Santos (2000) são: flexibilidade temporal; independência geográfica;

- Flexibilidade temporal, os cursos e serviços estão disponíveis 24 horas por dia, proporcionando ao educando adaptar a sua aprendizagem ao horário e a duração que melhor disponibilize.
- Flexibilidade no acesso a aprendizagem, é a possibilidade de o aluno aprender a um ritmo próprio, aprofundar os conteúdos, associar a teoria às múltiplas atividades práticas e escolher o método que melhor se adapta ao seu estilo e possibilidade.
- Independência geográfica, é possível por meio da educação à distância, em particular o e-learning. Não existem fronteiras de tempo e espaço, abrindo caminhos de formação para as pessoas que tenham dificuldades de frequentar o ensino presencial, pela localidade geográfica ou pela impossibilidade de se ausentar do local de trabalho.
- Estimula a auto aprendizagem, permitindo o desenvolvimento pessoal e contínuo dos indivíduos, conferindo-lhes maior autonomia. Nesta modalidade o centro das atenções é o utilizador, ao contrário do que acontece no ensino presencial, onde os elementos centrais são o professor ou conteúdos.

### 3.2- Desvantagens dos AVAs

Para Santos (2000) e Duggleby (2002), as desvantagens apresentadas pelo e-learning são as seguintes:

- Ausência de relação humana entre educador/educando, pode traduzir-se em isolamento, desmotivação, ausência de autodisciplina o que implica resultados nada satisfatórios.
- Conteúdos mais gerais, a componente prática é menor, isto porque o investimento em simulação é elevado.
- Não se adapta a todos os perfis de destinatários. O e-learning é uma das modalidades de ensino que exige muita responsabilidade e maturidade, por parte do aluno, fatores essenciais para a eficácia dessa estratégia de aprendizagem do que a formação presencial e, algumas experiências de uso de computadores e de utilização de Internet.
- Exige alguns conhecimentos tecnológicos. A falta de habilidade com as tecnologias e Internet por parte dos alunos, torna-se muito difícil a adaptação do AVA como estratégia ideal para aprimorar os seus conhecimentos. Visto que osAVAs se baseiam nas tecnologias informática e multimídia.

Atualmente existem diversas opções de Ambientes Virtuais de Aprendizagem. Alguns são gratuitos e outros pagos. Podem ser de *software livre* ou proprietário. Cada um dispõe de facilidades diferentes.

Apresenta-se a seguir alguns exemplos de ambientes virtuais de aprendizagem: Ex: [Moodle](#), [SOLAR](#), [TelEduc](#), [Blackboard](#) etc.

Como se pode verificar, os AVAs apresentam vantagens e desvantagens cabendo a cada um dos seus utilizadores explorar ao máximo essas ferramentas para que seus objectivos sejam alcançados.

A escolha dos AVAs anteriormente mencionados, depende principalmente da facilidade que as mesmas oferecem para o acesso a informação.

No nosso trabalho elegemos o Moodle como a ferramenta para efectuar interligação com o Sistema de Gestão Académica pelas vantagens acrescida que oferece.

## 3- SISTEMAS DE GESTÃO ACADÉMICA

Os sistemas de gestão académica são sistemas de informação que, normalmente, disponibilizam funcionalidades de controle quanto a dados sobre registos de discentes, docentes, cursos, perfis curriculares: disciplinas, requisitos, equivalências, associações; oferta de turmas a cada período letivo; pré-matrícula e matrícula de discentes; lançamento de notas *online* pelos docentes; histórico escolar; registo de diplomas, entre outras funcionalidades que auxiliam o controle administrativo das instituições de ensino.

Numa instituição de ensino, um sistema deste tipo é uma ferramenta fundamental para o controle dos processos administrativos e da gestão acadêmica. No geral, os sistemas

de gestão académica propõem-se a controlar e a agilizar os processos da instituição, possibilitando a consolidação de informações importantes para a gestão, através da análise de dados, tais como matrículas, aproveitamento académico, frequência, evasão, entre outros indicadores.

Está disponível num sistema de gestão académica de uma instituição todas as informações referentes à vida académica do aluno e do professor. Isso permite que diretores contem com uma radiografia completa das atividades da instituição permitindo monitorar o desempenho acadêmico e financeiro, além de permitir planejar a evolução futura e tomar decisões (Lyceum, 2013).

Existem varias instituições de ensino que desenvolveram os seus próprios sistemas de gestão Académicas. Como exemplo temos o Nónio que é um sistema de gestão académica na Universidade de Coimbra, que de entre outras funcionalidades permite as seguintes:

- ✓ Candidaturas a cursos;
- ✓ Matrículas e Inscrições;
- ✓ Consultar pautas de avaliação;
- ✓ Consultar Sumários e Materiais de Apoio;
- ✓ Consultar Informação sobre Cursos, Menores e Disciplinas;
- ✓ Participar em fóruns de discussão;
- ✓ Obter o seu suplemento ao Diploma.

#### 4- MOODLE.

No ponto anterior falou-se dos AVAs e sua importância bem como as suas vantagens e desvantagens para o processo de ensino.

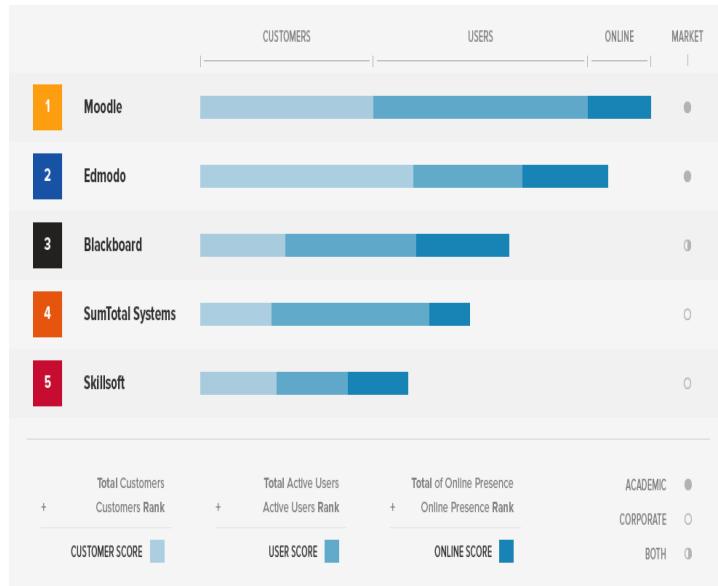
Para a nossa proposta de Integração de Sistema de Gestão Académica optamos pela escolha do Moodle, a seguir faremos uma abordagem acerca do Moodle, que vantagens e desvantagens oferece para a integração.

O Moodle – *acrônimo de Modular Object - Oriented Dynamic Learning Environment* – é um software concebido na filosofia *open source* de apoio a aprendizagem que disponibiliza aos usuários desta ferramenta um ambiente virtual de aprendizagem (AVA), de acordo com (MOODLE, 2013) moodle é um sistema de gestão de cursos - *Course Management System (CMS)*, também conhecido como *Learning Management System (LMS)* ou um Ambiente Virtual de Aprendizagem (AVA). Constitui-se em um sistema de administração de atividades educacionais destinados a criação de comunidades on-line, em ambientes virtuais voltados para a aprendizagem colaborativa. Permite, de maneira simplificada, a um estudante ou a um professor integrar-se, estudando ou lecionando, num curso a sua escolha (Moodle About, 2013).

Como AVA, o Moodle foi desenvolvido por vários especialistas que contribuíram para o seu crescimento em diversos estágios (Al-Ajlan & Zedan, 2008). O Moodle disponibiliza toda a informação necessária e essencial para individuos interessados em contribuir cada vez mais para a sua evolução. O Moodle está disponível para *download* a partir do URL: <http://moodle.org/downloads>, em várias versões, as quais apresentam diferentes estados de desenvolvimento.

Segundo Moodle (2011) o Moodle é um dos tipos de SGA que é distribuído livremente uma vez que é um software Open Source (sob os termos da Licença Pública GNU). Dadas as características de um software Open Source, significa que, o Moodle tem direitos de autor, mas são concedidas algumas liberdades que permitem a qualquer entidade que o queira utilizar de parametrizar, ou fazer algo a mais sobre o código fonte do mesmo, mas não deve modificar nem eliminar a licença e copyrights originais, e aplicar esta mesma licença a qualquer software derivado do mesmo.

Tal como foi anteriormente dito, existem inúmeras ferramentas que apoiam a ideia concebida pelo AVA. Em 2012 (DUNN) apresentou um resultado com as vinte melhores plataformas para AVA. Na **Figura que se segue**, apresentamos alguns destes AVA's onde pode-se verificar o Moodle em primeiro lugar, justificando mais uma vez o motivo da escolha do Moodle para a integração com o Sistema de Gestão Académica.



<<http://www.edudemic.com/2012/10/the-20-1>

#### 4.1- Funcionalidades do Moodle

Eis algumas das funcionalidades que irão interessar ao administrador do sistema:

- Moodle pode ser instalado em Unix, Linux, Windows, Mac OS X, Netware e qualquer outro sistema que suporte PHP.
- Moodle é desenhado de forma modular, e permite uma grande flexibilidade para adicionar, configurar ou remover funcionalidades, em vários níveis.
- Moodle permite upgrade simplificado de uma versão para outra mais recente: possui uma sistemática interna que permite fazer atualização de suas bases de dados e reparar-se automaticamente.

- Moodle requer apenas uma base de dados (que pode ser compartilhado com outras aplicações, se necessário).
- Moodle suporta uma variedade de bases de dados.
- Moodle promove uma interação sócio-construtivista, que inclui colaboração, reflexão crítica, permitindo máxima interação e integração entre a comunidade virtual.
- Moodle pode ser aplicado como opção totalmente virtual ou como complemento/suporte a turmas presenciais.
- Moodle tem uma interface clara, limpa e simples, compatível com qualquer browser, sem maiores exigências de tecnologia.
- Lista de cursos mostra descrição sumária dos cursos disponíveis, informando, inclusive, se estão disponíveis para acesso de visitantes.
- Os cursos podem ser enquadrados em categorias. Um site Moodle pode comportar e fazer busca a centenas de cursos.
- Áreas para entradas de textos (pesquisas, postagem para fórum, entradas diversas de textos) permitem edição (negrito, imagens, sublinhados, etc.) de forma fácil, usando uma interface html bem simples (WYSIWYG HTML), acessível a qualquer utilizador.

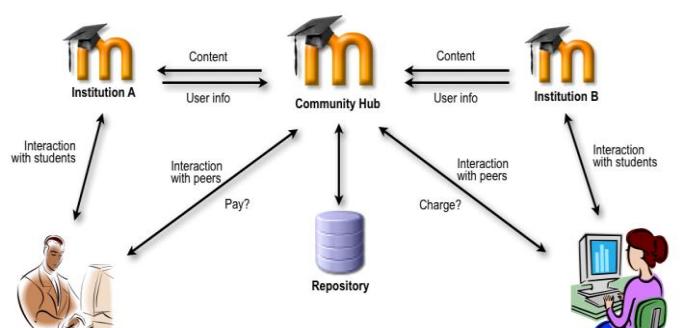
#### 4.2- Gestão do Moodle

- O site é gerido por um administrador, definido durante o setup inicial.
- O layout geral pode ser facilmente alterado a partir de temas pré-configurados, com fontes, cores e padrões que podem ser modificados facilmente e adaptado às necessidades de cada organização.
- Módulos e Plug-ins podem ser adicionados a partir da instalação inicial do Moodle.
- Pacote para língua portuguesa disponível, além de mais 40 outros idiomas, se necessário.
- Caso necessário, o código fonte do programa pode ser alterado para adaptar-se às necessidades, por tratar-se de código aberto (software livre - GPL license).

#### C. 4.3- Administração de Contas de Utilizadores Moodle

- O objetivo é minimizar a necessidade de envolvimento do administrador apenas ao necessário.
- Suporta uma gama de mecanismos de autenticação por módulos plug-in de autenticação, permitindo fácil integração com sistemas já existentes na organização.
- Método de definição de e-mail padrão: os usuários podem criar suas próprias contas de login. Os endereços são validados por confirmação.

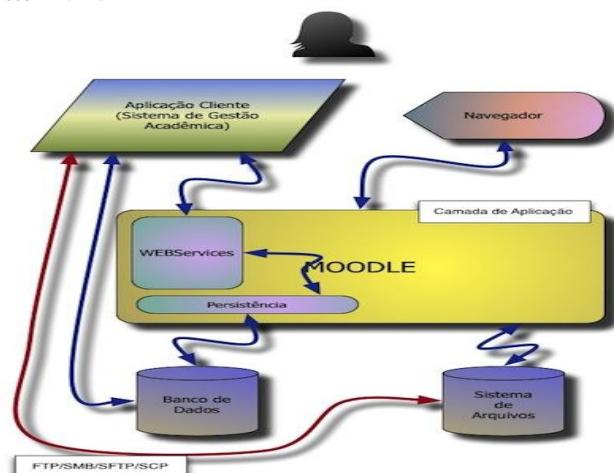
- Base de Dados externa para autenticação: qualquer base de dados que contenha pelo menos dois campos pode ser usado como fonte externa de autenticação.
- LDAP method: contas de login podem ser verificados via um servidor LDAP. O administrador pode especificar que campos usar.
- IMAP, POP3, NNTP: contas de login são verificadas em servidor de mail ou news. SSL, certificados e TLS são também suportados.
- Dados externos: qualquer base contendo ao menos dois campos pode ser usada como fonte para autenticação externa.
- Cada pessoa necessita de apenas uma conta, e com ela pode ter diferentes acessos.
- Uma conta de administrador controla a criação de cursos e cria instrutores relacionando usuários a cursos.
- Cada conta do tipo criação de cursos pode criar cursos e agir e servir como instrução.
- Os professores podem ter seus privilégios editados de forma que não possam editar/excluir conteúdos de cursos (útil para instrutores não fixos)
- Segurança: professores podem gerar chaves de acesso aos cursos, de forma a não permitir que qualquer indivíduo aceda suas aulas. Eles podem passar essas chaves de acesso por e-mail.
- Professores podem inscrever e excluir alunos manualmente, se desejarem.
- Alunos podem ser excluídos automaticamente, caso ultrapassem período de inatividade definido pelo administrador.
- Alunos são estimulados a criar um perfil on line, incluindo fotos, descrição, hobbies, etc. E-mails podem ser protegidos de exibição caso solicitado.
- Cada aluno pode escolher a linguagem que prefere usar (inglês, francês, português, espanhol, etc.)
- O professor também pode "forçar" uma linguagem para um curso específico.



<http://www.google.com.br/url?sa=i&source=1>

## 6- INTEGRAÇÕES COM MOODLE

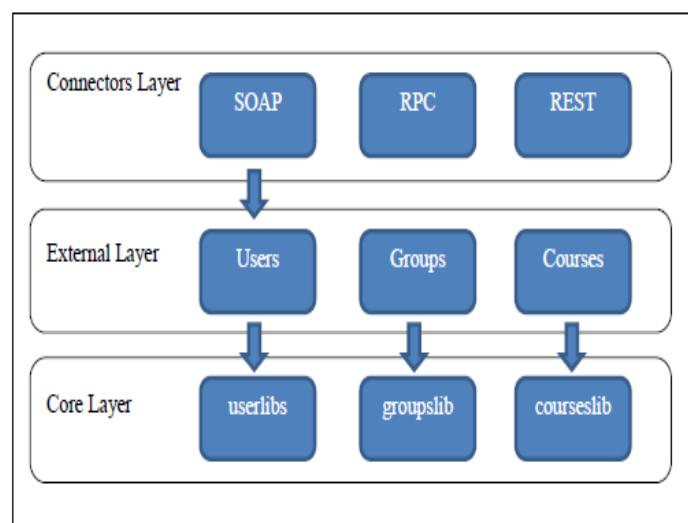
Com o objectivo de tornar o *Moodle* uma aplicação para integração entre sistemas diversos, a comunidade que desenvolve o Moodle, em janeiro de 2008, apresentaram uma proposta de criação de uma infraestrutura de Web Service para este AVA.



<http://www.google.com.br/url?sa=i&source=2>

Só em julho de 2010 é que a tarefa ficou concluída com o lançamento do Moodle 2.0 (Moodle Releases, 2013). Disponibilizando para os utilizadores Moodle acesso as funções de núcleo do AVA, permitindo interoperabilidade com sistemas externos.

Apresenta-se a seguir a arquitectura WebService Moodle:



MOURA, R.; BERNARDINO, J. Porto, v. 5, p. 1

1- Connectors layer: Implementa vários protocolos da camada de mensagem: SOAP (Simple Object Application Protocol), REST (Representational State Transfer), XML-RPC (eXtensible Markup Language-Remote Procedure Call) e AMF (Action MessageFormat). Nesta camada é possível integrar outros protocolos em função das necessidades de integração com outros sistemas. Esta camada é também

responsável por aceitar ligações HTTP (Hypertext Transfer Protocol), criar sessões e atuar como ponte para a camada externa. (MOURA e BERNARDINO, 2010).

- 2- External layer: Constituída por um conjunto de ficheiros com funções que permitem comunicar com a camada inferior, sendo responsáveis por garantir que o acesso às funções do núcleo apenas é realizado por quem tem capacidade e permissões para o fazer. Permite ainda validar os parâmetros utilizados e chamar a função correta do núcleo. (MOURA e BERNARDINO, 2010).
- 3- Core Layer: Núcleo do Moodle onde existem funções interessantes para serem utilizadas por sistemas externos e por isso, necessitam de ser publicadas. São exemplos destas funções as relacionadas com utilizadores, cursos, grupos, calendários, etc. (MOURA e BERNARDINO, 2010).

### 6.1- Motivação para Integração de Sistemas

A Revista (2012) apresenta uma series de pontos que servem de motivação para integração e alguns deles baixo apresentam-se:

- **Otimização do tempo:** no que concerne à realização das atividades administrativas, pedagógicas e de comunicação, considerando que, para executar as suas atividades, o docente necessita alternar entre os dois sistemas, o que é improdutivo. Isso pode confundir o docente, especialmente no caso das funcionalidades presentes nos dois sistemas, o que pode levar o docente a executar a atividade no ambiente errado.
- **Segurança da informação:** é preciso que exista garantia e confiabilidade nos dados. O usuário precisa sentir-se seguro quanto aos dados armazenados e às informações geradas pelo sistema.
- **Facilidade de uso:** os usuários almejam trabalhar em sistemas fáceis de utilizar e intuitivos, pois as dificuldades para aprender e explorar o “novo” são minimizadas, e os benefícios da ferramenta tornam-se mais evidentes. A integração transparente ao usuário, na qual ele não perceba a existência dos dois ambientes, sinaliza que a prática docente seria facilitada e simplificada.
- **Acompanhamento do desempenho do discente:** a integração dos dados e informações presentes nos dois sistemas podem produzir conhecimento para ajudar o docente a compreender o que acontece no aprendizado do discente ou da turma. Esse conhecimento pode auxiliar o docente a refletir sobre sua prática e, dessa forma, realizar os ajustes necessários de forma que o objetivo do ensino seja alcançado.
- **Acesso a informações cadastrais e acadêmicas do discente:** auxilia no processo de avaliação, pois fornece argumentos para que o docente possa avaliar o discente, ponderando sobre seu desempenho em semestres

anteriores, ou em outras disciplinas. Essas informações possivelmente auxiliam ao docente refletir e replanear sua prática.

## 6.2- Funcionalidade de integração de um Sistema de Gestão Académica e Moodle

Segundo ainda um inquérito a Revista (2012) apresentou as seguintes funcionalidades entre os sistemas de gestão académica e o Moodle:

- Existência de ata eletrônica para as atividades enviadas por e-mail ou postadas no fórum. Tal funcionalidade objetiva garantir que as atividades enviadas pelo discente foram recebidas.
- Existência de relatórios: relação dos discentes matriculados por turma, emissão de ata de presença, ementa da disciplina, equivalência entre as disciplinas, perfil curricular do curso.
- Funcionalidade para controle dos prazos definidos para a execução e entrega das atividades dos discentes.
- Funcionalidades que possibilitem o planeamento, registo e acompanhamento da disciplina, além do controle das atividades planejadas e as realizadas, de modo a alertar o docente quanto ao não cumprimento do planeamento. Possibilitar à coordenação e ao discente acompanhar o planeamento da disciplina.
- Importação/exportação de planeamentos a partir de documentos texto, planilhas e de unidades de aprendizagem no padrão IMS LD10.
- Funcionalidades de comunicação entre docente e discente, de forma que auxiliem o discente a superar suas dificuldades e auxiliem o docente na reflexão da sua prática.
- Funcionalidade de registo e acompanhamento do desempenho do discente: simplificar e viabilizar o registo e acompanhamento do desempenho do discente e possibilitar ao discente o acompanhamento de seu desempenho.
- Relatórios: que mostrem a evolução do processo de avaliação do discente e da turma, de forma a comparar o desempenho do discente em outras disciplinas, juntamente à verificação do desempenho de turmas anteriores, de modo a criar estratégias para auxiliar a reflexão docente e o replaneamento da disciplina. E relatórios que forneçam subsídios à equipe pedagógica para repensar novas estratégias, e responder questionamentos como: É prudente mudar de professor? O problema está na interação? O problema está no material? O problema é o embasamento? Que possibilite aferir se o problema é pontual em determinada cidade, ou em determinada disciplina. Que possibilite investigar o desempenho do discente na disciplina e orientá-lo na condução de

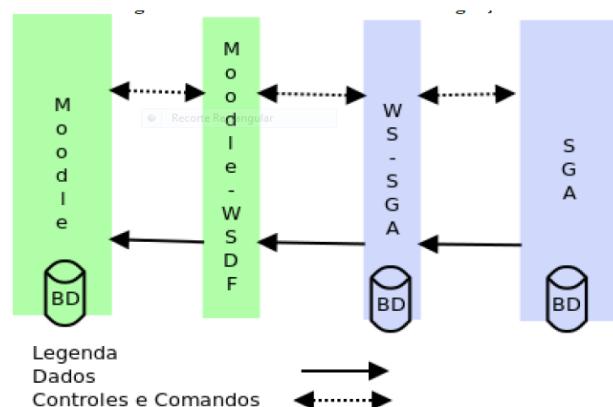
sua vida acadêmica, considerando os problemas de legislação.

- Funcionalidades que possibilitem o acesso às informações académicas como notas, frequência, calendário e plano de ensino.
- Funcionalidades que possibilitem a realização de atividades online, e que possam ser corrigidas e registadas automaticamente no sistema de gestão académica.
- O sistema de gestão da aprendizagem precisa considerar a estrutura de cursos de longa duração, por exemplo, cursos, universitários.
- Necessidade de login único em ambos os sistemas

## 6.3- Proposta de solução (Moura e Bernardino)

### Descrição Geral

A proposta de solução é baseada no modelo conceitual de (Moura e Benardino 2010) que a seguir se apresenta, e como se vê ambos os sistemas apresentam a estrutura de web service.



Fonte: (MOURA e BERNARDINO, 2010)

Onde:

- Moodle – AVA, Ambiente Virtual de Aprendizagem;
- SGA – Sistema de Gestão Acadêmica;
- *Moodle DF-WS* – infraestrutura de *Web service* do Moodle. Esta camada só existe na versão 2.0 ou superior do Moodle;
- *WS-SGA* – *Web service* de acesso ao sistema de gestão acadêmica.

Como se observa na figura acima, os dados são unidirecionais, do sistema de gestão acadêmica ao ambiente virtual (*Moodle*), ou seja nunca um dado será passado do *Moodle* para o *SGA*. Paralelamente ao fluxo dos dados deste modelo se observa, ainda na que o fluxo dos comandos e controles se dá em todas as direções, ou seja, tanto o *Moodle*

*DF-WS* envia e recebe informações para o *WS-SGA* como também de forma contrária. E por fim, vemos que os sistemas das extremidades só se comunicam com seus respectivos *web service*, ou seja, não existe comunicação direta entre os sistemas.

Em toda e qualquer instituição de ensino, os dados existentes no SGA são dados fidedignos e utilizados como fonte de informação para a emissão de certificados e documentos oficiais que atestam o percurso académico do aluno. Assim, no modelo proposto, considera-se que os dados existentes no SGA se sobrepõem a qualquer informação existente no Moodle que conflitu com a existente no SGA. (MOURA e BERNARDINO, 2010)

A abordagem proposta permite garantir a integridade dos dados existentes no SGA, uma vez que os acessos são apenas de leitura e controlados pelo Web Service.

A criação de utilizadores, cursos, disciplinas e a associação dos utilizadores a disciplinas é efetuada de forma simples, rápida e segura. Com a infraestrutura existente e, considerando o modelo de integração proposto à base de Web Services, muitos outros serviços podem ser desenvolvidos de forma a aumentar a partilha de informação entre o SGA e o *Moodle*.

A utilização deste modelo propõe simplificar as principais tarefas de gestão dos cursos e das disciplinas, libertando os administradores do *Moodle* da árdua tarefa de no início de cada semestre terem de criar as disciplinas e associar estas aos docentes.

#### 6.4- Descrição da Integração entre as Plataformas

##### *Moodle e SGA usando Web Services.*

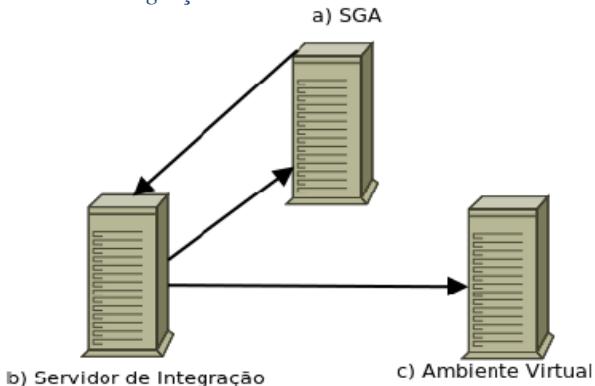
A partir deste ponto pretende-se dar uma visão geral e sintética a alto nível daquilo que seria a implementação.

Para a respectiva integração é necessário em primeiro lugar ter-se o *Moodle* instalado no servidor da instituição em que se pretende fazer a integração.

Tal como anteriormente foi referido, o processo de integração baseia-se no modelo proposto que por (Moura e Bernardino) que a seguir passamos a descrever a forma como essa integração será feita.

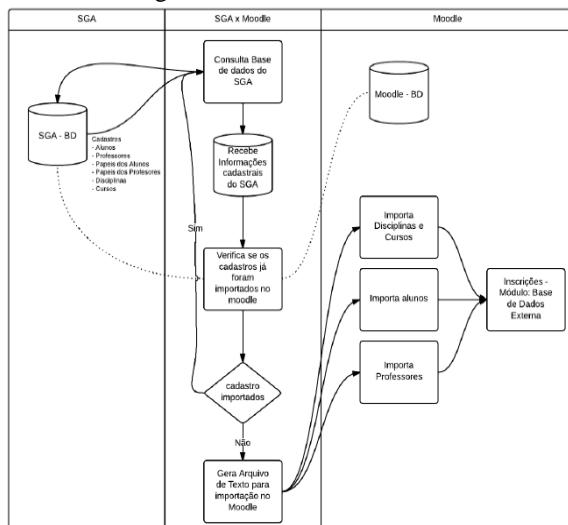
A forma como a integração será feita é apresentada na figura a seguir:

##### *Servidores de Integração 1*



Para Moura e Bernardino o processo de integração é feito da seguinte forma:

- 1- É necessário que tenhamos a base de dados do SGA. A partir daí possuímos mais dois servidores, um servidor que hospedará o *Moodle* e o outro servidor responsável por realizar a comunicação entre a Base de Dados do SGA e a do *Moodle*. A figura representa como é feita a comunicação através do diagrama de fluxos segundo Moura e Bernardino.



#### 6.4.1 Criticas Ao Modelo de Modelo de Moura e Bernardino

Tal como foi anteriormente que o modelo proposto por Moura e Bernardino é unidirecional, também podemos perfeitamente verificar no esquema por eles apresentados, a forma como a informação vai circular no sistema, a informação sai da base de dados do SGA para o *Moodle* em que tal processo é controlado pelo WebService por eles apresentado e, não se nota qualquer fluxo de informação do *Moodle* para o SGA, o que faz com que o modelo por eles apresentado não seja completamente sofisticado.

Atualmente as instituições de ensino estão preocupadas na integração de sistemas que facilitam não só a Gestão Acadêmica, mas que também todos os utilizadores do sistema sentam-se beneficiados com a integração entre os sistemas. Para este caso por exemplo os professores são obrigados a fazer a entrega das notas para a área acadêmica, e posteriormente o sector acadêmico volta a inserir as notas no *Moodle* o que faz com que seja um processo bastante lento e fastigioso para os utilizadores.

Para resolver essas e outras debilidades que este modelo apresenta, apresentamos uma solução de integração que pensamos ser uma possível solução para o processo de tratamento da informação na instituição e de modos a facilitar os serviços da integração que é o objectivo principal.

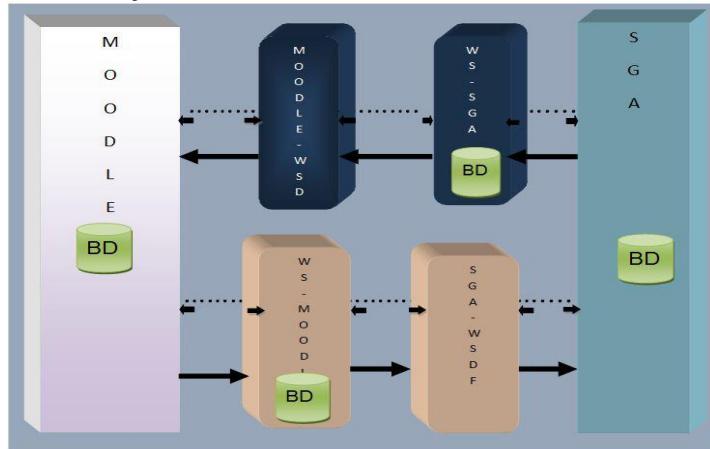
#### 7- Proposta Para o Novo Modelo de Integração

Este ponto tem como finalidade de apresentar uma Proposta para o novo Modelo de Integração, sustentada no modelo apresentado por (MOURA e BERNARDINO, 2010).

Em conformidade com o que anteriormente foi dito, neste ponto apresentamos um novo modelo conceptual e um novo modelo de integração que nós apresentamos para suprir as dificuldades que apresentavam o modelo proposto por (MOURA e BERNARDINO, 2010).

A seguir, apresentamos o modelo conceptual e a respectiva descrição do mesmo

#### PROPOSTA PARA O NOVO MODELO CONCEPTUAL DE INTEGRAÇÃO

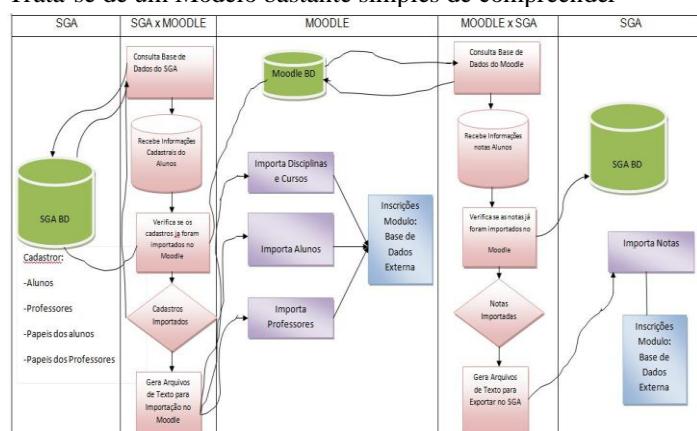


Modelo de Integração proposto por Abel Zacarias e Edson F. Livongue

Como se pode observar, no novo modelo conceptual de integração já verificar-se a criação de um WebService do Moodle com uma Base de Dados fazendo a ligação com a Base de Dados do SGA para que exista uma correspondência bidirecional e a informação flui de um lado para o outro ou seja já é possível fazer com que as notas dos alunos saiam do Moodle para o SGA, ultrapassando assim as dificuldades do modelo apresentado por (MOURA e BERNARDINO, 2010).

Posteriormente passamos a apresentar o novo Modelo de Integração partindo sempre do Modelo apresentado por (MOURA e BERNARDINO, 2010).

Trata-se de um Modelo bastante simples de compreender



Podemos também aqui verificar como será feita a integração entre os dois sistemas em que nessa nova integração, diferentemente da proposta de (MOURA e BERNARDINO, 2010) em que existe somente uma ligação entre o SGA e o

Moodle, agora já podemos verificar um novo serviço que é a interação entre o Moodle e o SGA.

É de salientar que neste modelo de integração proposto, a última coluna que temos o SGA, não tinha necessidade de lá estar. Foi colocado no sentido de evitar que tenhamos muitas setas a ligarem a primeira coluna que é o SGA. A mesma demonstra a forma como a informação sai do Moodle para o SGA.

Evidentemente que o modelo por nós propostos não é o único. Trabalhos do gênero podem sim vir a contribuir para o engrandecimento deste projeto que é bastante interessante e promissor.

#### 8- TRABALHO FUTURO

O trabalho futuro passa na implementação prática do presente Modelo de Integração. E pretende-se fazer a implementação do mesmo no Instituto Superior Politécnico da Huíla da Universidade Mandume Ya Ndumafayo em Angola, isso porque muitas das disciplinas que serão lecionadas nos próximos anos não possuem docentes e dentro da cooperação que a universidade tem com a Universidade de Coimbra, pode-se adoptar este modelo para que os conteúdos das disciplinas sem docentes possam ser lecionadas pelos docentes da UC a distância, facilitando a presença constante dos docentes da UC em Angola.

#### 9- CONCLUSÕES

O cenário tecnológico atual, que se depara em processo constante de inovação, sinaliza uma preocupação crescente em integrar os ambientes de gestão académica e os ambientes virtuais de aprendizagem. Tal integração visa, na esfera administrativa das instituições de ensino, facilitar o planeamento, o controle, a coordenação, a análise e o processo decisório.

Sugere-se que a integração entre os sistemas de gestão académica e da aprendizagem acarreta benefícios e auxilia o docente em sua prática, seja de planeamento, ensino, avaliação ou em sua acção reflexiva.

Verifica-se assim que actualmente, os sistemas de informação da organização entre eles os sistemas de gestão académica tendem a ser centralizados de modo a garantir a integridade e fidelidade das informação e que a integração através das tecnologias de Web services tem sido a solução mais usada. Isto, graças a vantagem O uso do SOA, através de Web Service, possibilitou obter, entre outras coisas, maior interoperabilidade, maior agilidade organizacional e menor carga de trabalho do TI.

Pretende-se fazer uma contribuição teórica para provar que o modelo é válido e aplicável.

## 10- REFERÊNCIAS

- 1- ANDREWS, T.; CURBERA, F.; DHOLAKIA, H.; GOLAND, Y.; KLEIN, J.; LEYMANN, F.; LIU, K.; ROLLER, D.; SMITH, D.; THATTE, S.; TRICKOVIC, I.; WEERAWARANA, S. Specification: Business Process Execution Language for Web Services. Version 1.1 - 05 May 2003. Disponível em:<http://www.ibm.com/developerworks/library/ws-bpel>
- 2- AUSTIN, D.; BARBIR, A.; FERRIS, C.; GARG, S. Web services architecture requirements. W3C working draft. 2002. Disponível em: <http://www.w3.org/TR/2002/WD-wsa-reqs-20021114#id2604831>
- 3- CHAPPELL, D.; JEWELL, T. Java Web Services. California: O'Reilly Books, 2002. Consultado em 01/03/2011, 2011.
- 4- COYLE, F. P. XML, Web Services, and the Data Revolution. Boston: Addison-Wesley, 2002.
- 5- CURBERA, F.; MUKHI N.; WEERAWARANA, S. On the emergence of a web services component model. In: 6th INTERNATIONAL WORKSHOP ON COMPONENT-ORIENTED PROGRAMMING AT ECOOP, 2001. Budapest. Proceedings of the 6th International Workshop on Component-Oriented Programming Budapest: 2001. Disponível em: [http://research.microsoft.com/users/cszypers/events/WCOP\\_2001/Curbera](http://research.microsoft.com/users/cszypers/events/WCOP_2001/Curbera)
- 6- DUGGLEBY, Júlia. (2002). Como ser um tutor online. Lisboa: Monitor EaD. Universidade Federal do Paraná, Pró Reitoria de Graduação e Ensino Profissionalizante. Centro Interdisciplinar de Formação Continuada de Professores-CINFOP. Ministério da Educação, Secretaria de Educação. UFPR, 2005.
- 7- DUNN, J. Edudemic - The 20 Best LMS. Edudemic, 2012. Disponível em: <<http://www.edudemic.com/2012/10/the-20-best-learning-management-systems/>>. Acesso em: 01.12.2013.
- 8- ERL, T. SOA: Princípios do design de serviços. Tradução de Edson Furmarkiewicz e Carlos Schafranski. São Paulo: Pearson Prentice Hall, 2009.
- 9- ERL, T.; KOPACK, M. Service-Oriented Architecture – Concepts, Technology, and Design. Crawfordsville: ed. Prentice Hall – PTR, 2005.
- 10- GOTTSCHALK, K.; GRAHAM S.; KREGER H.; SNELL J. Introduction to web services architecture. IBM Systems Journal, v. 41, n. 2, 2002. Disponível em: <http://www.research.ibm.com/journal/sj/412/gottschalk.pdf>
- 11- GRAHAM, S. et al. Building Web Services with Java™: Making sense of XML, SOAP, WSDL, and UDDI. Indianapolis: Sams Publishing, 2001.
- 12- <http://libdigi.unicamp.br/document/?code=vtls000401132>, (consultado em 01/12/2013) <http://www.moodle.ufba.br/course/view.php?id=30>- Acessado no dia 02-12-2013
- 13- KAYE, D. Loosely coupled: the missing pieces of web services. Califórnia: RDS Press, 2003.
- 14- KREGER, H. Web services conceptual architecture (WSCA 1.0) 2001. Disponível em: <http://www.ibm.com/software/solutions/webservices/pdf/WSCA.pdf>
- 15- LÈVY, Pierre. *As tecnologias da inteligência. O futuro do pensamento na era da informática*. Rio de Janeiro, Editora 34, 1.993
- 16- LUCENA, MAXWELL DE PAIVA,: tese de bacharelato (PLATAFORMA DE INTEGRAÇÃO MOODLE-SISTEMA DE GESTÃO ACADÉMICA BASEADA EM WEBSERVICE-2013, p 48-49).
- 17- LYCEUM. Lyceum, 2013. Disponível em: <<http://www.techne.com.br/produtos.asp?id=8>>. Acesso em: 10 Nov. 2013.
- 18- MARKS, E. A.; BELL, M., 2006, Service-Oriented Architecture: a planning and implementation guide for business and technology, John Willey & Sons Inc.
- 19- MENENDEZ2002- MENÉNDEZ, Andrés Ignacio Martínez. Uma ferramenta de apoio ao desenvolvimento de Web Services. Dissertação de Mestrado, Universidade Federal de Campina Grande, curso de Pós-Graduação em Informática, 2002. 97 p.
- 20- Moodle. (2011). About Moodle. Disponível em [http://docs.moodle.org/en/About\\_Moodle](http://docs.moodle.org/en/About_Moodle).
- 21- MORIN, E. Les sept savoirdsnécessaires à l'éducation du futur. Paris: Ed. Seuil UNESCO, 1999. Disponível em: <<http://unesdoc.unesco.org/images/0011/001177/117740fo.pdf>>. Acesso em: 03 de out. de 2011.
- 22- MOURA, R.; BERNARDINO, J. Um modelo para integração de serviços: Moodle e Sistemas de Gestão acadêmica. RISTI, Revista Ibérica de Sistemas e Tecnologias de Informação, Porto, v. 5, p. 31-44, 2010. Disponível em. Acesso em: 10 de maio de 2013.
- 23- NEWCOMER, E. Understanding web services: XML, WSDL, SOAP and UDDI. Boston: Addison-Wesley, 2002.
- 24- PADMANABHUNI, S.; GANESH, J.; MOITRA, D. Web Services, Grid Computing and Business Process Management: In: IEEE International Conference on Web Services 2., 2004. San Diego. Proceedings of the 2nd International Congress of Web Services San Diego: 2004. pp. 666-673.
- 25- Rice, W. H. (2006). *Moodle: E-Learning Course Development - A complete guide to SAMPAIO, C. SOA e Web Service em Java*. Rio de Janeiro: Brasport, 2006.
- 26- (REVISTA 2012) Revista Brasileira de Computação Aplicada (ISSN 2176-6649), Passo Fundo, v. 4, n. 1, p. 81-91, mar. 2012 84
- 27- SANTOS, F. (2006). Ambientes virtuais de ensino-aprendizagem: concepção e SNELL2001- SNELL, James. Programming Web Services with SOAP. Sebastopol: O'Reilly, 2001. 216p.*successful learning using Moodle* Birmingham: Packt Publishing Ltd. ISBN: 1-UDDI, Specification, 2001. Disponível em: <http://www.uddi.org/specification.html>
- 28- Wiki -[http://pt.wikipedia.org/wiki/Service-oriented\\_architecture](http://pt.wikipedia.org/wiki/Service-oriented_architecture) acessado aos 27-11-2013, as 00:21.
- 29- WSDL Specification, 2001 – W3C. Disponível em: <http://www.w3.org/TR/wsdl>

