

**UNIVERSIDADE DO VALE DO RIO DOS SINOS
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
CURSO DE INFORMÁTICA**

Guilherme Callegaro Sesterheim

**Ambiente de configuração de alertas a usuários da plataforma
Moodle para dispositivos móveis**

São Leopoldo, Junho de 2011

**UNIVERSIDADE DO VALE DO RIO DOS SINOS
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
CURSO DE INFORMÁTICA**

Guilherme Callegaro Sesterheim

**Ambiente de configuração de alertas aos usuários da
plataforma Moodle para dispositivos móveis**

Prof. Sérgio Crespo Coelho da Silva Pinto

Orientador

São Leopoldo, Junho de 2011

Agradecimentos

Considerando o fato de que esta monografia não começou com sua formalização, que ocorreu há pouco menos de um ano, e também temendo ser injusto com os que contribuíram para a realização da mesma, agradeço antecipadamente a todos os que, de alguma maneira, passaram pela minha vida durante este processo, e contribuíram para a finalização deste trabalho, sem citar nomes.

Agradeço aos meus pais e à minha família, que de todas as maneiras sempre me incentivaram ao estudo aprofundado neste campo pelo qual tenho tanto gosto. Agradeço por terem me apoiado e me auxiliado em tudo o que foi possível, desde moralmente, até de maneira material, me ajudando a adquirir os materiais que tornaram este projeto viável.

Agradeço também aos professores de todas as disciplinas da Unisinos, que me fizeram entender todos os conceitos e tecnologias que foram utilizadas neste projeto, e também por terem me ajudado a desenvolver a postura necessária para poder realizar um projeto deste porte.

Resumo

O ensino a distância é cada vez mais adotado pelas universidades e vem crescendo rapidamente em espaço e reconhecimento no mercado. Há universidades que oferecem cursos completos ministrados inteiramente a distância. Estes cursos surgem por proporcionarem várias vantagens, principalmente pela economia em relação ao caráter financeiro e por exigirem pouco deslocamento por parte dos alunos durante seus semestres de estudos.

O ensino a distancia através de plataformas móveis também vem crescendo nos últimos anos e tomando forma devido a uma computação móvel mais desenvolvida e a pessoas mais interessadas neste tipo de ensino. Consiste, em linhas gerais, em um modo através do qual informações sobre disciplinas cursadas a distância cheguem até os alunos por meio de dispositivos móveis.

Este trabalho demonstra o resultado da proposta de desenvolvimento de uma plataforma móvel altamente customizável, a qual é composta de três módulos (web, mobile e camada de configurações) para dar alertas a alunos usuários da plataforma Moodle que utilizam smartphones com Android (Google) como Sistema Operacional (SO).

Abstract

Distance learning is increasingly being adopted by universities and has been growing rapidly in space and market recognition. There are universities that offer entire courses that are conducted entirely in distance. These courses appear due to provide several advantages, mainly due to a financial nature and require few movements by the students during their semester of study.

The distance learning through mobile platforms is also being increasingly adopted and taking form due to a huge mobile computing intelligence, and more people interested in this kind of learning method. It consists in a way that information about courses, which are being taken over distance, get to the users through mobile devices.

This paper shows the result of the purpose of developing a high customizable mobile platform, which have three modules (web, mobile and configuration system), to provide alerts to users of Moodle platform that use smartphones with Android (Google) as operational system (OS).

Sumário

1. Introdução	10
1.1 Objetivos	12
1.2 Motivação	12
1.3 Contribuições	13
1.4 Método	14
2. Campo atual	16
2.1 MOMO (Mobile Moodle)	16
2.2 Moodle for Mobiles	17
3. Referencial Teórico	19
3.1 Plataformas de LMS	19
3.2 Desenvolvimento de aplicações móveis	23
3.3 Web services	24
3.4 Android	25
4. Tecnologias Utilizadas	26
4.1 Tecnologias e linguagens	26
4.2 Produtos	36
4.3 Notações	37
5. A integração desenvolvida	39
5.1 Alterações da base de dados	39
5.2 Integração entre a alteração na base de dados e a aplicação	42
6. Usuários do sistema	52
6.1 Administrador da aplicação	52
6.2 Professor	59
6.3 Aluno	60
7. A aplicação de configurações (web)	62
7.1 Tela de login	63
7.2 Módulo de ativação de disciplinas	65
7.3 Módulo de seleção de disciplina e definição de configurações	67
7.4 Webservice utilizado pela aplicação móvel	74
8. A aplicação mobile	75
8.1 Tela de configurações do dispositivo móvel	77
8.2 Recebimento de alertas	83

8.3	Visualização dos alertas	85
9.	Conclusão	86
10.	Trabalhos futuros	88
	Bibliografia	89
	Anexos	92

Índice de imagens

Figura 1 - Arquitetura do projeto MLEA.....	14
Figura 2 - Arquitetura macro do Moodle	23
Figura 3 - Exemplo de XML.....	27
Figura 4 - Definição de conexão com tabela de atividades.....	27
Figura 5 - Exemplo de código em HTML.....	28
Figura 6 - Exemplo de código HTML utilizado no projeto	29
Figura 7 - Exemplo de código CSS.....	30
Figura 8 - Exemplo de CSS utilizado no projeto.....	31
Figura 9 - Utilização da fonte padrão da página em uma página do projeto.....	31
Figura 10 - Palavra "ativar" com o estilo definido no CSS aplicado	31
Figura 11 - Função criada para obter o nome da disciplina sob alteração.....	33
Figura 12 - Arquitetura da aplicação	39
Figura 13 - Estrutura da tabela criada	41
Figura 14 - Trecho de código XML responsável por indicar a tabela que contém o cadastro de usuários	43
Figura 15 – Mapeamento da tabela que contém as informações sobre a tabela de configurações	45
Figura 16 - Identificação das tabelas que deverão ser acessadas pela aplicação a fim de obter os dados originais do Moodle.....	48
Figura 17 - Parte do código utilizado para selecionar os cursos em que o usuário está matriculado	49
Figura 18 - Indicação do campo que deve ser obtido	50
Figura 19 - Parâmetros gerais da aplicação.....	51
Figura 20 - Texto utilizado para a autenticação no servidor.....	51
Figura 21 - Caso de uso do administrador do sistema	52
Figura 22 - Dados para acesso à tabela de atividades.....	54
Figura 23 - Dados para acesso à tabela de fóruns.....	55
Figura 24 - Dados para acesso à tabela de chats.....	56
Figura 25 - Texto utilizado para autenticação no banco de dados	57
Figura 26 - Arquivo de configurações do Moodle (<i>config.php</i>).....	58
Figura 27 - Caso de uso do professor	59

Figura 28 - Caso de uso do aluno	60
Figura 29 - Tela de configurações.....	62
Figura 30 - Tela de login	63
Figura 31 - Autenticação inválida	64
Figura 32 - Módulo de ativação de disciplina em estado inicial (primeira utilização) ...	65
Figura 33 – Módulo de ativação de disciplinas em utilização	65
Figura 34 - Módulo de seleção de disciplinas e definição de configurações. Disciplina de definição de processos com alertas desativados	67
Figura 35 - Módulo de seleção de disciplinas e definição de configurações. Disciplina de definição de processos com alertas ativados.....	68
Figura 36 - Exemplo de identificação para montagem da tela de configurações	69
Figura 37 - Exemplo da tela de configurações montada a partir da definição no arquivo	69
Figura 38 - Seleção da disciplina a ser configurada.....	70
Figura 39 - Ativação dos alertas para a disciplina	70
Figura 40 - Sugestão de preenchimento para alertas de tarefas	71
Figura 41 - Sugestão de preenchimento para alertas de chats	71
Figura 42 - Sugestão de preenchimento para alertas de fóruns	72
Figura 43 - Imagem de sucesso ao salvar dados do usuário	72
Figura 44 - Imagem de erro durante operação de salvar dados	73
Figura 45 - Fluxo de funcionamento da aplicação móvel.....	75
Figura 46 - Fluxo de informações da aplicação móvel	76
Figura 47 - Apresentação da aplicação no menu padrão do Android.....	77
Figura 48 - Tela de configurações do dispositivo móvel	78
Figura 49 - Configuração de intervalo de tempo de atualização	79
Figura 50 - Ativar ou desativar a busca de alertas no dispositivo	80
Figura 51 - Informar dados de acesso ao servidor	80
Figura 52 - Configuração do endereço do servidor	81
Figura 53 - Informativo ao usuário	82
Figura 54 - Tela inicial do Android recebendo um alerta	83
Figura 55 - Alerta detalhado.....	83
Figura 56 - Tela da aplicação exibindo alertas	85

1. Introdução

O crescimento cada vez maior da computação móvel e de aplicativos robustos para operar em seus sistemas tem motivado a criação de plataformas de ensino para operar neste meio. Desta maneira, vem surgindo o conceito de aprendizagem móvel, mais conhecido como M-Learning.

De acordo com Georgiev (2004), este modo de ensino permite que alunos aprendam em qualquer lugar e a qualquer momento utilizando apenas conexões sem fio e um dispositivo que possa se comunicar com a web.

Segundo Crespo (2010), há uma variedade relativamente grande de dispositivos que podem ser usados para permitir este novo modo de ensino, tais como *PDA*s, *tablets* e *smartphones*. Entre os dispositivos citados, o que se mostra mais viável para esta ferramenta é o celular (*smartphone*), devido à sua enorme difusão pelo mundo. Além disso, o uso de celulares independe de classe social, idade e grau de instrução do usuário. Toda pessoa que possuir um celular com Android como SO e com acesso à web poderá utilizar o sistema desenvolvido, que foi baseado em comunicação com o Moodle (sistema de ensino a distância). Como mais uma forma de embasar o potencial do M-Learning, há o exemplo de várias cidades ao redor do mundo já possuírem Wi-Fi gratuita por todo o seu perímetro. Desta maneira, há a possibilidade de que, em qualquer lugar, o usuário consiga acessar os sistemas móveis de ensino que utilizar.

O Moodle – detalhado no item 3 – é um sistema de gerenciamento de cursos (CMS, do inglês *Course Management System*), ferramenta que permite aos educadores construir páginas personalizadas voltadas ao aprendizado e garante aos alunos um acesso coerente apenas às informações que lhes são pertinentes. Os pontos fortes do Moodle são, segundo Felizardo (2007), as ferramentas para ensino como criação de atividades, fóruns e os outros elementos que compõem o ensino. O problema encontrado é que, embora o Moodle seja uma ferramenta robusta, ainda não existia um projeto

próprio para dar suporte simples a uma plataforma de ensino móvel, como esta que foi idealizada.

Tendo em vista este cenário descrito, o ambiente CADMM (Configurações de Alertas para Dispositivos Móveis – Moodle) oferece suporte de modo a permitir que usuários possam configurar suas preferências em termos de alertas. Através disto, propicia uma ferramenta que alerte os alunos de maneiras diferentes (customizáveis) sobre suas atividades correntes na plataforma de ensino. Este sistema de alertas é altamente configurável e utilizado através de celulares que operem com Android como SO.

É importante ressaltar que este ambiente não demonstra uma ferramenta que substitui a estrutura de ensino à distância que existe atualmente (através de computadores), mas sim complementa o modo de aprendizado já existente. O objetivo também não é portar completamente o Moodle para dispositivos móveis, assim como outros projetos existentes já idealizam. O projeto idealizado possui o objetivo de enviar alertas simples aos usuários da plataforma de maneira rápida e dinâmica, funcionando de maneira semelhante a uma agenda eletrônica de compromissos.

1.1 Objetivos

Os objetivos do presente projeto são:

- desenvolver uma base para uma arquitetura flexível que permita que alunos configurem as informações pertinentes ao seu estudo de maneira fácil em um ambiente web;
- criar uma aplicação que seja facilmente adaptável e configurável a diferentes versões do ambiente Moodle através de um arquivo principal, em XML, que detenha as principais definições da aplicação. Estas configurações deverão definir desde os campos presentes na tela de configurações na web, que será visualizada e preenchida por cada aluno que utilizar o sistema, até estruturas que algumas tabelas no banco de dados deverão possuir;
- implementar agentes de softwares - que servirão para comunicação entre os usuários, ambiente de alertas e futuras outras aplicações que possam ter interesse em utilizá-los;
- criar um sistema que funcione semelhantemente a uma agenda eletrônica, de maneira que alerte os alunos sobre seus compromissos com as disciplinas que cursa através do Moodle.

1.2 Motivação

A facilidade cada vez maior em estabelecer comunicação móvel, tanto devido a dispositivos mais potentes como a um mercado mais propenso a este tipo de consumo, tem levantado possibilidades de aprendizado diferenciado do modelo de ensino conservador, onde professores passam seu conhecimento através de aulas presenciais apenas. O atual crescimento da plataforma Android (Google) para smartphones expõe uma boa oportunidade para criar novas maneiras de passagem de conhecimento que removerão limitações de aprendizado, o qual é atualmente limitado à interação entre professor e aluno e/ou ao uso de computadores para ensino a distância. Desta maneira, surge a motivação de

criar um sistema que possa abranger estas dificuldades de comunicação através de uma solução móvel.

Apesar de ser uma ferramenta robusta, o Moodle não possibilita, atualmente, um rápido acesso a pequenas informações, porém substanciais, que, no cenário atual, demandam tempo para serem acessadas e obtidas. Desta maneira, foi idealizada uma maneira para suprir esta necessidade, que é descrita pela apresentação deste projeto.

A facilidade de acesso a dados simples (como por exemplo, a data limite de entrega de uma tarefa), porém que necessitam de tempo para serem acessados pela plataforma Moodle através de um computador, é um dos principais benefícios que foram implementados. A criação de uma aplicação a ser acessada através de dispositivos móveis, que possa fazer esta interação com o Moodle e apresentar informações relevantes relacionadas aos cursos do usuário, mostra-se como um dos principais aspectos de motivação para este projeto.

Outro benefício que interessaria a grande massa seria, de acordo com Wains & Mahmood (2008), a necessidade que algumas pessoas têm de conciliar estudo, família e vida social.

1.3 Contribuições

O desenvolvimento está inserido no escopo do projeto MLEA, pertencente ao programa de mestrados da Unisinos em parceria com a Universidade do Panamá, sob coordenação do professor Sérgio Crespo.

Segundo Crespo (2011), o projeto MLEA é um projeto internacional, oriundo do Panamá, que tem como objetivo viabilizar o ensino a distancia através de dispositivos móveis. Este acesso poderá ser feito a qualquer momento, desde que o usuário possua um dispositivo compatível com a aplicação desenvolvida.

O projeto possui alguns focos, tais como: 1) integrar-se ao Programa de Educação Virtual da Universidade do Panamá; 2) abordar o problema da educação a distância, relacionado à proliferação de dispositivos que podem ser usados como solução para isto; 3) apresentar benefícios da

implementação de uma plataforma de educação a distância que seja acessível através de dispositivos móveis.

O projeto apresentado por esta monografia contribui criando agentes para a camada de “Agentes de Interoperabilidade”, conforme destacado pela imagem 1. A contribuição, além de trazer os agentes, traz a implementação de uma inteligência para recebê-los e executá-los no contexto da aplicação desenvolvida:

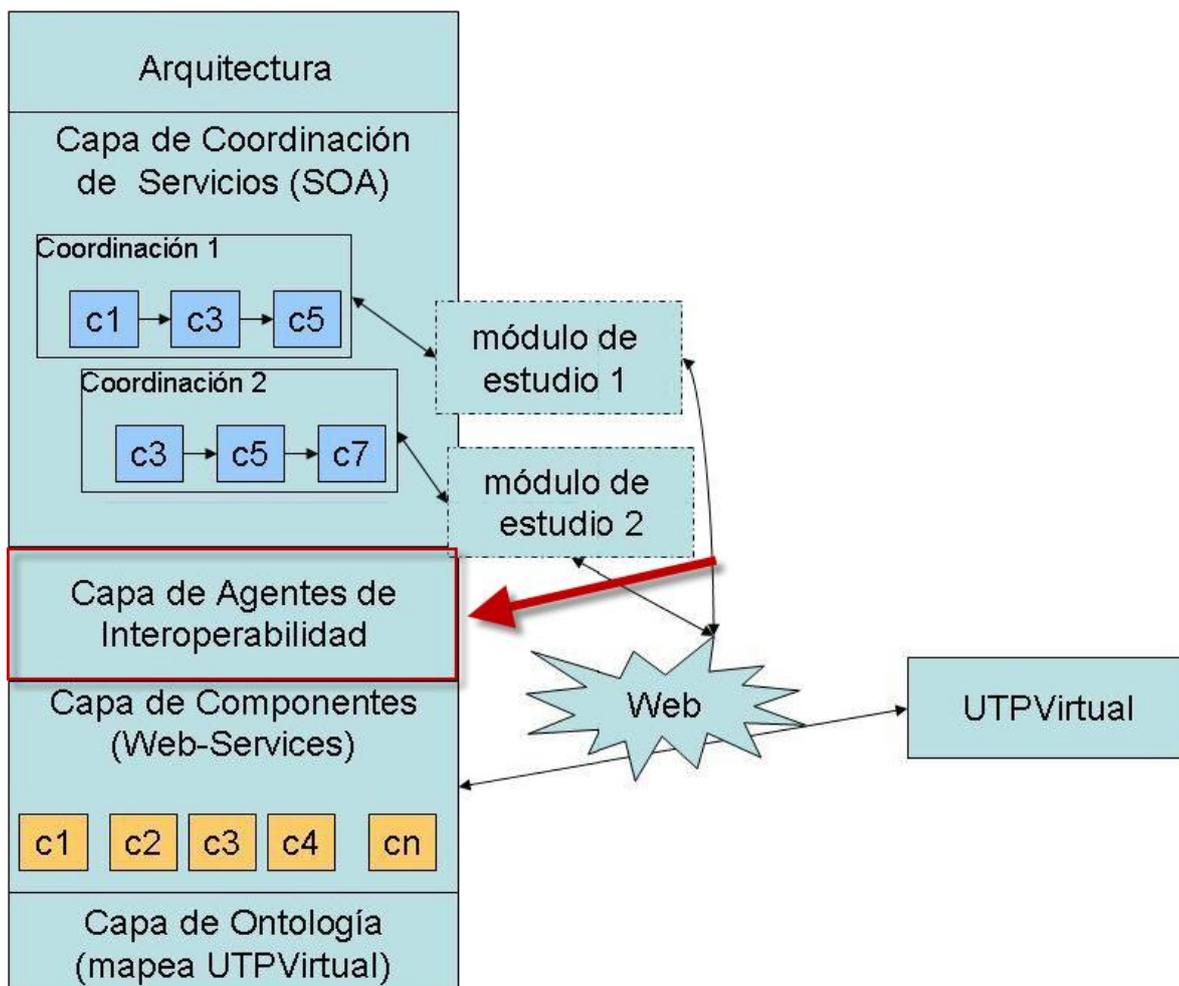


Figura 1 - Arquitetura do projeto MLEA

1.4 Método

Como etapas metodológicas, o trabalho aborda:

- 1) Revisão bibliográfica:
 - 1.1 Estudo das tecnologias utilizadas para o desenvolvimento da ferramenta;
 - 1.2 Análise de política para alterações nas diferentes versões do Moodle;

1.3 Análise da estrutura da base de dados das diferentes versões do Moodle;

2) Análise e projeto:

2.1 Análise de necessidades de customizar a plataforma Moodle em suas diferentes versões;

2.2 Análise da viabilidade da implementação da ferramenta proposta, baseada na comunicação com a base de dados utilizando apenas um arquivo de configurações e as tecnologias eleitas;

2.3 Elaboração das interfaces com o usuário, primando pela usabilidade tanto na web quanto no dispositivo móvel;

2.4 Elaboração do cronograma e controle das atividades conforme delimitado;

3) Desenvolvimento da aplicação:

3.1 Elaboração das sessões do arquivo de interoperabilidade;

3.2 Desenho da solução para leitura e utilização do arquivo;

3.3 Desenho da aplicação móvel e do serviço executado em segundo plano que é responsável pelos alertas;

3.4 Implementação;

3.5 Testes da aplicação;

2. Campo atual

O campo em que este projeto está inserido possui algumas iniciativas já disponíveis e outras em desenvolvimento.

Tendo conhecimento destas iniciativas, reconhecendo que há pessoas trabalhando em projetos semelhantes e sabendo também que outros projetos em andamento não abrangem as mesmas áreas que este projeto pretende atender, a proposta foi elaborar uma iniciativa parecida, porém com escopo diferenciado e com a possibilidade de ser utilizada livremente em outros projetos acadêmicos futuros.

A diferença entre este projeto e outros é o fato de o desenvolvimento para celulares ser de um sistema de alertas e configuração de alertas customizáveis para usuários Moodle, bastante similar a uma agenda eletrônica, porém com a possibilidade de ser modificada de diversas maneiras para adequar-se ao cenário de cada instituição adotante.

Os projetos em andamento, que visam portar o Moodle completamente para dentro do smartphone, são:

2.1 MOMO (Mobile Moodle)

Segundo a página web do mantenedor do projeto Mobile Moodle (2010), o objetivo é portá-lo a dispositivos mobile, porém o projeto está parado no momento devido à falta de recursos.

O projeto possui arquitetura pensada e desenvolvida em Java para ser adaptável a um número maior de celulares, de modo a proporcionar acesso total ao Moodle e a novos cenários de interação com esta ferramenta. A aplicação também possui um plugin que deve ser instalado na parte do servidor da ferramenta. Isto deve ser realizado pelos administradores do servidor. Este passo garante que a comunicação entre as duas plataformas será bem sucedida.

O projeto não é afiliado da organização Moodle original, ou seja, representa uma iniciativa independente dos mantenedores originais da

ferramenta, possui código aberto e pode ser redistribuído e/ou alterado livremente seguindo os termos da GNU.

Possui alguns módulos já prontos e disponíveis para download em alguns modelos de smartphones. Endereço do projeto: <http://www.mobilemoodle.org/momo18/> acesso em 07/06/2011.

2.2 Moodle for Mobiles

De acordo com o portal do projeto Moodle for Mobiles (2010), há um projeto em desenvolvimento de um aplicativo/framework para ser utilizado em smartphones, que visa a portar o Moodle completamente para estes.

A vantagem deste projeto é que, por ser de responsabilidade da própria mantenedora do Moodle, a documentação disponível é bem completa e dá condições palpáveis para que desenvolvedores interessados possam utilizá-la.

Este projeto é um desenvolvimento da organização original que mantém o Moodle. Originalmente, foi pensado para ser utilizada com a versão 1.6 da plataforma, mas foi adaptada para a 1.9. Sua interface com o Moodle é feita através de CHTML (HTML compacto), pois foi pensado para abranger o cenário atual do Japão, que seria de até 98% dos celulares operando, pois este é o número de aparelhos que conseguem executar aplicações em CHTML.

Utilizando insumos deste projeto, os desenvolvedores possuem condições de elaborar aplicativos complexos para serem visualizados através de aparelhos de celular. Há vários componentes disponíveis para utilização capazes de montar interfaces tanto no módulo de feedbacks (resposta de professores sobre atividades desenvolvidas pelos alunos) e também no módulo de quiz. No site do projeto, também podem ser encontradas considerações que devem auxiliar no ato da criação de aplicações para serem utilizadas junto ao Moodle, de modo a facilitar a criação de interface e garantir que ela possuirá fácil entendimento por parte dos usuários.

Este projeto abrange, até o momento, apenas o Japão. As informações estão contidas no site “Moodle for mobiles”, indicado na bibliografia.

3. Referencial Teórico

Este capítulo apresentará as principais plataformas de ensino a distância utilizadas no mercado, focando no diferencial que representa os motivos que levaram à escolha do Moodle para este projeto. Também será apresentada uma breve contextualização do desenvolvimento de aplicações para dispositivos móveis.

Conforme Ellis (2009) em “Field Guide to Learning Management Systems”, os Sistemas de Gestão de Aprendizagem (do inglês *Learning Management System* – LMS) têm como funções disponibilizar recursos de interação entre professor e aluno, tanto de maneira síncrona como assíncrona, que possibilitem o processo de passagem de conhecimento e aprendizagem. Os LMS possibilitam muitas vantagens tanto para alunos, quanto para professores, pois possibilitam que mais ferramentas de ensino sejam aplicadas. Estas ferramentas ajudam os alunos a capturar de maneira mais fácil o conhecimento que está sendo passado pelo professor. Como exemplo destas ferramentas temos os fóruns de discussão, atribuição de atividades a serem entregues, questionários, preenchimento de wikis – todos assíncronos -, chats com todos os participantes da disciplina – síncronos -, etc.

3.1 Plataformas de LMS

Abaixo serão apresentadas as principais plataformas de LMS do mercado, com o Moodle, que foi escolhido por este projeto, sendo a última delas para obter um bom comparativo e ressaltar suas vantagens.

3.1.1 Sakai (EUA)

Sakai é uma comunidade composta por instituições acadêmicas, organizações comerciais e outros contribuintes que trabalham em conjunto para a construção de um ambiente de colaboração e aprendizado (do inglês *collaboration and learning environment* – CLE). O software, que leva o mesmo nome da organização, é grátis e possui código aberto. Possui focos de ajudar a ensinar, pesquisar e colaborar conteúdos.

O Sakai é desenvolvido em Java, e em dezembro de 2007 possuía uma base de 150 instituições que o utilizavam e cerca de outras 100 que o usavam como projeto piloto. [KORCUSKA, M; BERG, A. M, 2004].

3.1.2 Claroline (Bélgica)

Claroline é uma ferramenta para ensino a distancia e que proporciona trabalho colaborativo (várias pessoas ao mesmo tempo executando uma ação). Surgiu no ano 2000 na Bélgica como projeto da universidade Católica de Louvain. Possui diversos recursos para proporcionar interação com o usuário, tais como exercícios, chats, wikis e fóruns. Atualmente é utilizada por 101 países e está disponível em mais de trinta idiomas.

O Claroline é desenvolvido em PHP e utiliza MySQL como banco de dados. Está sob a licença GNU/GPL, o que lhe garante o código aberto. Também possui alguns órgãos governamentais o apoiando. [Claroline, 2011].

3.1.3 ILIAS (Alemanha)

ILIAS (Integriertes Lern-, Informations- und Arbeitskooperations-System – alemão para Sistema Integrado de aprendizado, informação e trabalho cooperativo) é um sistema de aprendizado a distância com código aberto. Dá suporte a gerenciamento de conteúdo de aprendizagem e ferramentas de colaboração, comunicação e avaliações. É desenvolvido em PHP e MySQL.

O desenvolvimento do ILIAS lançou o primeiro protótipo em 1997 dentro do projeto VIRTUS da Universidade de Cologne. Em 1998 a primeira versão do sistema foi publicada. O sistema possui diversas ferramentas para proporcionar interação entre professores e alunos, tais como avaliações, chats, fóruns, podcasts, entre várias outras ferramentas administrativas e de interface entre professor e aluno. [ILIAS, 2011].

3.1.4 Moodle (Austrália)

Conforme Sabbatini (2007), o Moodle (*Modular Object-Oriented Dynamic Learning Environment* – ambiente modular de aprendizagem

dinâmica orientada a objetos) é uma plataforma de aprendizagem a distância baseada em software livre.

O Moodle foi criado e é mantido por uma comunidade de desenvolvedores espalhados pelo mundo, que também prestam suporte aos usuários e criam novas funcionalidades, entre outras contribuições. A ferramenta é regida pelo projeto/filosofia GNU, que foi iniciado no ano de 1984 por Richard Stallman com o objetivo de criar um sistema operacional com o código totalmente livre.

A fundação e empresa Moodle (www.moodle.org e www.moodle.com, respectivamente) fornecem apoio ao desenvolvimento e à tradução para vários idiomas e também apoio profissional à sua instalação.

O Moodle é um sistema robusto e já consagrado no mercado. Possui instalações em mais de 155 países, sendo que a maior delas em operação fica na Universidade Aberta da Inglaterra, onde conta com uma base de 6 mil cursos e mais de 45000 alunos.

Além de ser uma plataforma de ensino a distância, o Moodle também é um sistema de gestão do ensino e aprendizagem (do inglês CMS – *Course Managment System* ou LMS – *Learning Management System*), ou seja, ele suporta os educadores no sentido de criar cursos a serem acessados apenas on-line ou também como auxílio a disciplinas presenciais, provendo uma grande variedade de recursos para interação com os alunos.

Sob análise técnica, o Moodle é um sistema baseado na web e opera através de dois componentes: o servidor onde fica instalado - onde também ficam guardadas as informações utilizadas por alunos e professores - e o cliente do usuário de acesso – o qual é feito através de qualquer navegador de internet. O Moodle é desenvolvido na linguagem PHP, opera com vários tipos de bancos de dados e possui seu código completamente aberto, o que permite as alterações, melhorias e adaptações que as empresas adotantes desejarem.

- **Benefícios do Moodle como CMS**

Há vários benefícios em uma universidade utilizar um CMS e alguns deles merecem destaque por melhorarem a metodologia de ensino dos professores. Através do Moodle, o professor é capaz de criar disciplinas híbridas, que combinam o ensino presencial – ainda a maneira de transmissão de conhecimento predominante em nossa sociedade – e a possibilidade de que os materiais utilizados pela disciplina fiquem na web, facilitando o acesso dos alunos a eles. Estes materiais podem ser desde apenas textos para leitura e/ou consulta, como também questionários, fóruns, chats e atividades relacionadas às aulas em geral.

Como exemplos de outros benefícios que são obtidos através do Moodle como CMS escolhido, podemos citar:

- consulta de materiais de aula na web, de maneira que o aluno não precise aguardar até a próxima aula presencial para tê-los;
- realização de atividades de aula nos horários que lhe forem mais convenientes;
- melhoria nas disciplinas; através das diferentes alternativas de interação ofertadas pelo Moodle, é mais fácil ao professor criar maneiras de transferir o conhecimento.

Para este projeto, o Moodle foi escolhido como base para o desenvolvimento por motivos relacionados ao projeto MLEA. A Universidade do Panamá, que é a principal beneficiária do projeto, utiliza o Moodle como plataforma de ensino presencial e a distância para seus alunos. Somado a isto, outro fator que motivou o uso do Moodle foi o fato de já haver bastante familiaridade entre os envolvidos no projeto e a aplicação.

Em adição a isto, foi visto que, segundo Sabbatini (2007), o Moodle pode ser facilmente customizado, adaptado, alterado ou estendido pela iniciativa que o adota.

- **Representação da arquitetura do Moodle**

A imagem abaixo (2) representa, de maneira macro, a arquitetura do Moodle. Ele possui esta representação comum de três camadas, porém elas não são fortemente estabelecidas, de maneira que os desenvolvedores

podem inverter as ordens de acesso aos dados de acordo com suas necessidades [The Open University, 2010].

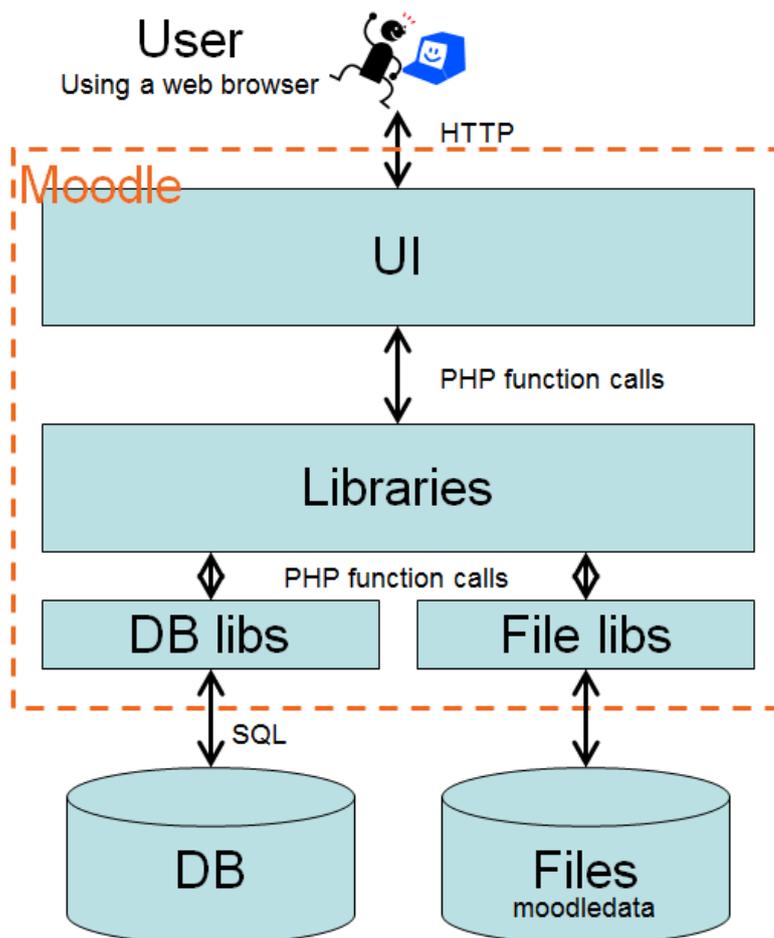


Figura 2 - Arquitetura macro do Moodle

3.2 Desenvolvimento de aplicações móveis

O desenvolvimento de aplicações móveis sempre se mostrou um desafio para engenheiros de software por diversos motivos. Alguns deles se destacam: como dificuldades para depuração, dificuldades para identificar se um problema apontado por um usuário está de fato na aplicação ou no ambiente onde roda, dificuldades para identificar aparelhos que estejam apresentando problemas durante a execução do programa, etc [Duart, M. L.; Dotti, L. F., 2001].

Este cenário vem mudando gradativamente devido a ambientes de desenvolvimento mais robustos disponibilizados pelas principais fornecedoras de sistemas operacionais do mercado (Apple – iOS, Google – Android, Microsoft – Windows Phone e RIM – Blackberry). Estes

ambientes possuem características similares para facilitar o desenvolvimento, como disponibilização da virtualização de dispositivos usados para os testes, centralização do desenvolvimento em apenas uma ferramenta por vez e integração de frameworks de linguagens já conhecidas no mercado.

Ultimamente, com os lançamentos mais recentes dos fabricantes de dispositivos móveis, pode-se notar claramente uma tendência de crescimento da plataforma Android [Tecmundo, 2011].

Por estes motivos, neste projeto foi optado por utilizar a plataforma Android (Google) para o desenvolvimento da aplicação. O ambiente de desenvolvimento de aplicativos para Android baseia-se em Java, que é uma das linguagens mais utilizadas no mercado, e possui várias plataformas de desenvolvimento maduras o suficiente para proporcionar um desenvolvimento de software robusto como o que este exige.

3.3 Web services

A utilização de *web services* resolve o problema de comunicação entre duas diferentes ferramentas. São utilizados para que possa ser feita comunicação com aplicações de diferentes linguagens e/ou diferentes propósitos, ou seja, para qualquer aplicação que precise acessar informações que outro sistema dispõe. O uso de *web services* é recomendado, pois dá uma característica dinâmica às aplicações, que de outras maneiras, exigiriam outros métodos para trafegar estes dados, que em sua maioria necessitariam de intervenção humana. Já que os *web services* são executados sem intervenção humana, lhes é garantido um alto índice de confiabilidade.

Originalmente os *web services* foram pensados para realizar comunicações utilizando apenas em XML para o tráfego de dados, entretanto surgiram outras maneiras de estabelecer comunicação de *web services* que ganharam espaço devido a suas diferenças de aplicabilidade.

Os *web services* SOAP (*Simple Object Access Protocol*) foram utilizados para estabelecer a comunicação entre as duas aplicações deste projeto (web e móvel, que serão vistas adiante). SOAP descreve um tipo genérico de *web service*, que não impõe regras para tráfego de informações,

deixando a cargo do desenvolvedor definir como será feita a comunicação entre as diferentes ferramentas. SOAP é proposto para ser utilizado através do protocolo de comunicação HTTP, mas pode ser feito através de outros conhecidos protocolos de comunicação web, como por exemplo, SMTP.

3.4 Android

Android é um sistema operacional móvel que é executado sobre Linux, desenvolvido e mantido pela Google desde 2005 e lançado em 2008. Permite aos desenvolvedores criarem aplicações na linguagem Java, C, C++ e C#, e atualmente é o sistema operacional móvel líder de ativações e de utilização no mercado.

O Android foi selecionado para este projeto tendo em vista algumas vantagens que estão descritas abaixo:

- Todo o desenvolvimento móvel do projeto MLEA, em que este desenvolvimento está inserido, baseia-se em Android.
- Atualmente, o Android é o sistema operacional móvel que é líder de mercado, e por este motivo proporcionará ao projeto uma maior visibilidade devido a grande base de usuários que ele dispõe.
- O sistema operacional é atualizado constantemente pela mantenedora, e estas atualizações não trazem problemas para aplicações antigas, de maneira que não é necessário adequar desenvolvimentos já feitos.
- O sistema operacional também é utilizado por *tablets*, que também garantirão uma maior difusão da ferramenta, visto que poderá ser utilizada em mais plataformas devido a sua compatibilidade com o SO.
- Possui excelente documentação disponível na web, e as aplicações podem ser desenvolvidas a partir de qualquer tipo de computador, sem restrições deste gênero.

4. Tecnologias Utilizadas

Nesta sessão, serão apresentadas as tecnologias e produtos que foram utilizados para o desenvolvimento do projeto. Estas tecnologias são as que compõem a inteligência e interface de ambas as aplicações (web e mobile). Esta sessão foi dividida em três subitens: “Tecnologias e Linguagens”, “Produtos” e “Notações”. As tecnologias listadas são: XML, HTML, CSS, Javascript, Java e o algoritmo MD5. O produto apresentado é o MySQL e a notação que foi utilizada para criação dos casos de uso e fluxos de informações da aplicação apresentada é o TAM.

4.1 Tecnologias e linguagens

Nesta sessão, serão descritas todas as tecnologias que foram utilizadas no projeto. As tecnologias são as linguagens XML, HTML, CSS, Javascript e Java e o algoritmo de encriptação de dados MD5.

4.1.1 XML – eXtensible Markup Language

De acordo com o Centro Atlântico (2001), XML é um formato universal para partilha de dados entre aplicações.

Reconhecido pela W3C em 1998, o XML tornou-se, em pouco tempo, o formato universal para partilha de dados entre aplicações. É utilizado em, por exemplo, transações comerciais, relatórios financeiros e estatísticos e também indicado a todos os dados que precisam trafegar de um sistema a outro e/ou exijam uma representação estruturada.

Em resumo, a XML é uma linguagem e formatação utilizada para descrever conceitos e transportar dados através de um arquivo. Na figura 3, temos o exemplo de como seria a descrição de uma pessoa trafegando de uma aplicação a outra:

```

...
<peessoa>
  <nome>José da Silva</nome>
  <endereco>Rua Brasil, 1000</endereco>
  <cep>90000-000</cep>
  <cidade>Porto Alegre</cidade>
  <telefone>51-91234567</telefone>
</peessoa>
...

```

Figura 3 - Exemplo de XML

O XML neste projeto foi utilizado para definir parâmetros do funcionamento da aplicação web e da aplicação de dispositivos móveis. Através do XML, também foi definida a comunicação entre a web e a base de dados do Moodle.

A imagem 3 possui uma pequena parte do arquivo que foi gerado e que representa a maneira com que a aplicação web acessa a tabela do Moodle, a qual é responsável por manter os dados das atividades que estão em desenvolvimento pelos alunos. Por ser de grande importância para o entendimento do projeto, o arquivo será visualizado e detalhado por completo no item 5. O arquivo também pode ser alterado baseado no funcionamento esperado da aplicação, variando conforme a necessidade da instituição que optar por utilizá-la. As alterações que podem ser feitas no arquivo estão descritas no item 5.2 e a explicação sobre cada ponto que pode ser alterado está localizada no item 6.1:

```

<source ID="assignmentsForUser">
  <tableName>mdl_assignment</tableName>
  <whereCondition>course=?</whereCondition>
  <returnField type="ID">id</returnField>
  <returnField type="TIMEAVAILABLE">timeavailable</returnField>
  <returnField type="NAME">name</returnField>
  <returnField type="COURSE">course</returnField>
</source>

```

Figura 4 - Definição de conexão com tabela de atividades

4.1.2 HTML (Hyper Text Markup Language)

Segundo o Centro Atlântico (2004), HTML consiste numa linguagem de formatação de texto que possibilita a construção (programação/formatação) de notas, com a extensão htm ou HTML. Estas notas contêm pequenas marcas (*tags*) que indicam ao web browser como apresentar o conteúdo incluído ou referenciado no documento, sob a forma de texto, imagens ou suportes multimídia.

O HTML é escrito através de tags que indicam início e fim de uma instrução. Esta instrução é interpretada pelo navegador, que, por sua vez, exibe a informação de maneira amigável ao usuário. Quase toda tag deve conter um indicativo de início e fim, com algumas poucas exceções.

Abaixo, na imagem 5, temos um exemplo de código HTML que exibirá as palavras “Olá mundo” em vermelho e terá o título da página mudado para “Exemplo de HTML”:

```
<html>
<head>
<title>Exemplo de HTML</title>
</head>
<body>
<font color="red">Olá Mundo!</font>
</body>
</html>
```

Figura 5 - Exemplo de código em HTML

Neste projeto, o HTML foi utilizado durante todo o desenvolvimento da aplicação web, onde o usuário guarda suas preferências relacionadas a alertas, frequência de alertas e utilização destes. A imagem 6 possui um pequeno trecho do código HTML que foi utilizado.

O código é responsável por definir o nome da página e também por estabelecer quais serão as ações que a página irá tomar a partir do momento em que for carregada.

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  <title>Configura&ccedil;&otilde;es Moodle</title>
  <link href="<%=application.getInitParameter("css")%>/style.css" media="screen" rel="stylesheet"/>
</head>

<body onLoad="return onLoad();">
```

Figura 6 - Exemplo de código HTML utilizado no projeto

4.1.3 CSS (Cascading Style Sheets)

Conforme W3C (2010), CSS é uma linguagem que define como exibir elementos HTML.

É uma linguagem de estilo, a qual é utilizada para definir como elementos de linguagens definidas por tags (como HTML e XML) serão exibidos.

O CSS é utilizado para substituir marcações de estilo que precisariam ser colocadas em todas as tags do arquivo de interface. Com a ajuda do CSS, o desenvolvedor pode criar um arquivo separado e apenas referenciá-lo nas páginas que desejar. Esta referência é feita em todos os arquivos que utilizarão o estilo descrito no arquivo de CSS e facilita muito as futuras alterações da aplicação, pois o estilo de um portal inteiro pode ser modificado alterando apenas alguns arquivos de configuração.

A imagem 7 mostra um pequeno exemplo de um arquivo em CSS que possui a definição para que a tag “body” - responsável por definir o estilo padrão de uma página da internet - de um arquivo HTML qualquer tenha o estilo de suas letras alterado e cor de fundo alterada:

```
body{
    font-family: Verdana, Arial;
    background-color: #FFFFFF;
}
```

Figura 7 - Exemplo de código CSS

Neste projeto, o CSS foi utilizado para definir o estilo como as informações são apresentadas ao usuário. As imagens tiveram seu tamanho e disposição na tela alterados através do CSS. Os estilos dos textos que o usuário lê na tela também foram alterados através de CSS, de maneira que vários tipos de textos sejam apresentados com formatações, tamanhos, cores e estilos diferentes.

A imagem 8 mostra um pequeno trecho do arquivo de CSS que foi gerado para a aplicação. Este trecho mostra a maneira como a fonte padrão da página será apresentada:

```
.fonte_padrao {  
    font-family: verdana, Arial, Helvetica, sans-serif;  
    font-size: 11px;  
    color: #404040;  
    text-decoration: none;  
    font-weight: bold;  
}
```

Figura 8 - Exemplo de CSS utilizado no projeto

A imagem 9 mostra a utilização do estilo de fonte definido na imagem 6, que é feito através da chamada da propriedade “class”.

```
<label class="fonte_padrao">Ativar</label>
```

Figura 9 - Utilização da fonte padrão da página em uma página do projeto

Já a imagem 10 mostra o resultado da utilização da fonte da imagem 8 já formatada e interpretada por um navegador web.



Figura 10 - Palavra "ativar" com o estilo definido no CSS aplicado

4.1.4 Javascript

Javascript é uma linguagem de programação interpretada criada em 1995 por Brendan Eich, da Netscape, como uma extensão do HTML para o browser Navigator 2.0. Hoje, existem implementações JavaScript nos browsers dos principais fabricantes, mas o uso dessa linguagem não tem se limitado aos browsers. Tem sido usada, em menor escala, como linguagem de suporte a tecnologias de gateway para servidores HTTP e até como linguagem de roteiro de propósito geral. Embora ainda seja mantida e estendida pela Netscape, parte da linguagem JavaScript já é padrão proposto pela ECMA – organização europeia para padrões em comunicações, que visa a transformá-la em padrão Web3. [Rocha, 1999].

Javascript foi desenvolvida para ser uma linguagem *client-side* para navegadores web e hoje é considerada a mais utilizada para tal tarefa. Foi desenvolvida baseada no paradigma de orientação a objetos, possui tipagem fraca e dinâmica e também funções de primeira classe, que podem ser encontradas em linguagens mais populares como Java e C++, tais como abrir e fechar janelas, por exemplo.

Em resumo, Javascript é uma linguagem utilizada pelos desenvolvedores para prover interação e troca de informações entre o usuário e a página web.

Neste projeto, Javascript foi utilizado fortemente em várias partes da montagem e interação das páginas. Na página de login, que será apresentada no item 6, foi utilizado para executar validações e exibir mensagens de erro. Já a página de configurações é inteiramente montada com base em Javascript. Em decorrência do contexto da montagem da página html exigir que seja dinâmica, Javascript é utilizado desde a abertura da página (para definir quais informações o usuário poderá visualizar, por exemplo), passando pela ativação e desativação de disciplinas, até o momento em que o usuário define as configurações e disciplinas que deseja visualizar.

A imagem 11 possui um pequeno exemplo do código que foi utilizado no projeto para obter o nome da disciplina que está sendo alterada pelo usuário:

```
function getDisciplinaSelecionada() {  
    return document.getElementById('comboDisciplinas').value;  
}
```

Figura 11 - Função criada para obter o nome da disciplina sob alteração

4.1.5 Java

Segundo a organização mantenedora da linguagem, Sun Microsystems (2010), Java é uma poderosa linguagem de programação, que pode ser utilizada para criar desde complexas aplicações até pequenos componentes para serem utilizados pela web. É também uma linguagem simples, orientada a objetos, interpretada, robusta, segura, portátil, desempenhada em alta performance, dinâmica e permite multitarefas.

Java é utilizada de diversas maneiras. Pode ser utilizada para desenvolver aplicações desktop (por exemplo, o programa de IRPF da Receita Federal do Brasil), aplicações mobile (como em jogos para celulares), aplicativos para controlar eletrodomésticos (por exemplo, lavadoras, ventiladores, etc) e também de diversas maneiras para a web, que é o cenário deste projeto.

No presente trabalho, Java foi utilizado para desenvolver a inteligência do servidor, que deve receber as informações do usuário e guardá-las, e também obter informações da base de dados e exibir ao usuário.

Toda vez em que a aplicação é executada, a instância Java tem papel fundamental. Quando ela é iniciada, é necessário ler o arquivo de configurações (a ser explicado no item 5.2) para que a aplicação saiba como se comunicar com as tabelas do Moodle. A comunicação com o servidor e dados já existentes na base também é feita utilizando Java.

Em resumo, todas as atividades executadas pelo usuário que exigem interação com o Moodle e/ou base de dados, são feitas através da instância

Java. Como exemplo de operações executadas pelo usuário que exigem interação com o Java, há as atividades de ativar ou desativar disciplinas, ativar alertas, configurar preferências para as disciplinas, etc.

4.1.6 MD5

Segundo o MD5 Message-Digest Algorithm (MIT – 1992), MD5 é um algoritmo para encriptação de dados. Surgiu como uma melhoria ao antigo MD4 e foi idealizado para ter funcionamento rápido em sistemas 32 bits.

Algumas informações exigem ser guardadas com nível de segurança maior nas bases de dados das aplicações. Isto evita que funcionários e/ou usuários de todos os níveis de utilização do sistema, por vezes mal-intencionados, obtenham informações sigilosas dos usuários das aplicações e as utilizem de maneira indevida. O algoritmo é utilizado muito comumente para senhas, informações de cartão de crédito e outras informações que exijam muita segurança ao serem guardadas em bases de dados.

O MD5 foi utilizado neste projeto para realizar a autenticação do usuário que está tentando acessar a aplicação. As senhas dos usuários, por padrão do Moodle, ficam guardadas na base de dados criptografadas com o algoritmo de MD5. O sistema obtém a senha que o usuário informou, o criptografa, e faz a verificação contra a base de dados para identificar se o usuário existe e informou os dados corretos. A partir desta operação, permite ou nega o acesso do usuário à operação que está tentando realizar.

4.2 Produtos

Nesta sessão, serão descritos todos os produtos que foram utilizados no projeto. O produto de mercado utilizado foi o MySQL, uma ferramenta para gerenciamento de banco de dados.

4.2.1 MySQL

MySQL é um SGBD (sistema gerenciador de banco de dados). É o mais conhecido SGBD open source utilizado comercialmente, com mais de 5 milhões de instalações em funcionamento [Ruas e Neves – O Guia prático do MySQL].

Foi desenvolvido e é mantido pela organização *MySQL AB Limited Company*, que vende e presta manutenção a diversos pacotes com diferentes produtos envolvendo o MySQL [Ruas e Neves – O Guia prático do MySQL].

MySQL é um SGBD robusto para utilizações em pequena escala. Possui boa performance e é recomendado para aplicações pequenas por ter funcionamento e administração simples.

Neste projeto, a escolha do MySQL foi feita com base na boa integração que o Moodle possui com o mesmo. Documentos do projeto indicam a exigência de que seja utilizado o MySQL em virtude do desenvolvimento ter sido feito baseado no mesmo. Desde sua instalação, o Moodle possui forte integração com o MySQL, que é utilizado para criar a base de dados e os registros necessários ao seu funcionamento, de maneira automatizada, desde o início do processo. [MoodleDocs – Installing Moodle, 2007].

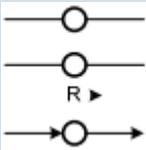
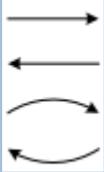
4.3 Notações

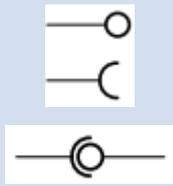
Nesta sessão, serão descritas todas as notações que foram utilizadas no projeto. A notação de mercado que foi utilizada é a TAM, que foi desenvolvida pela SAP visando à simplificação da representação de arquiteturas.

4.3.1 TAM (Technical Architecture Modeling)

TAM é um padrão de modelagem desenvolvido pela SAP, baseado em MOF (Meta-Object Facility) 2.0 e na UML (Unified Modeling Language), com o objetivo de simplificar representações de arquiteturas de sistemas. (SAP, 2007).

Existem diversos tipos de componentes que o TAM utiliza, porém nem todos foram utilizados neste projeto. Através da tabela 1, os diagramas utilizados são mostrados e seus objetivos descritos:

Elemento	Descrição
Agentes 	Elementos ativos que são capazes de desempenhar tarefas. Agentes contêm agentes, armazenamentos, subsistemas, componentes e classes (herdadas).
Canais 	Elementos passivos que são utilizados para comunicação entre agentes. Toda informação transferida é volátil. Opcionalmente, podem ser utilizados para requisição (representados pelo “R ➔”) e/ou determinação da direção de fluxo.
Acessos (ler, escrever, alterar) 	Arcos de acesso definem a maneira como agentes podem acessar um elemento.

<p>Interfaces</p> 	<p>Interfaces definem pontos de conexão entre os componentes e seu ambiente. Componentes fornecem um conjunto de serviços (interface fornecedora) e os conectam com serviços prestados por terceiros elementos (interface receptora)</p>
--	---

O TAM foi utilizado para desenhar os fluxos de informações das aplicações do projeto e também para criar os casos de uso que ilustram as atividades que os papéis envolvidos com este sistema desempenham.

5. A integração desenvolvida

Nesta sessão, será descrita a integração que foi desenvolvida para o projeto. Esta integração ocorre entre o ambiente Moodle e a aplicação de inteligência de alertas, que será descrita posteriormente neste documento. A integração desenvolvida é a parte principal - a não-vista pelo usuário aluno e professor - deste projeto. Ela possui toda a inteligência para fazer a interação e comunicação entre as duas aplicações e foi pensada de maneira a poder ser configurada fácil e rapidamente.

Através da imagem 12, pode-se visualizar a arquitetura desenvolvida para o projeto:

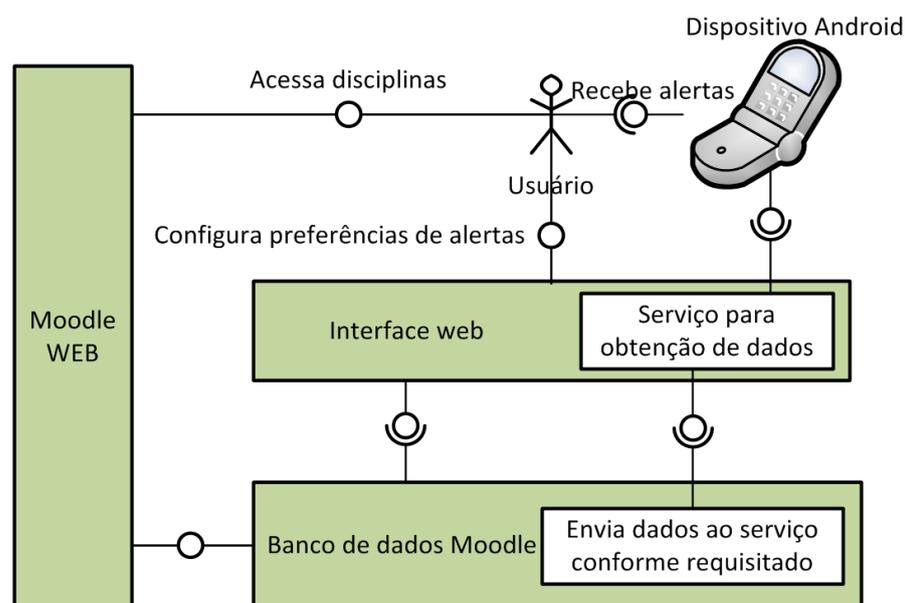


Figura 12 - Arquitetura da aplicação

5.1 Alterações da base de dados

Para o escopo do projeto, o foco foi realizar o mínimo de alterações possíveis no Moodle. Após um estudo de caso e de como seria desenvolvida a adaptação para criar a interface entre Moodle e a nova aplicação, foi visto

que não seriam necessárias alterações diretas nos arquivos que definem o funcionamento do Moodle, e sim apenas na base de dados.

Portanto, fez-se necessário desenvolver alterações apenas no banco de dados do Moodle, o MySQL.

5.1.1 Nova tabela de integração

Foi criada uma nova tabela dentro do banco de dados do Moodle chamada Z_CONFIG¹. Esta tabela possui campos que determinam quais são as configurações que o usuário poderá alterar e seus registros guardarão as informações definidas por ele. É importante ressaltar que os campos criados e mostrados abaixo são apenas sugestões do que pode ser utilizado. O usuário administrador, que será mostrado no item 6.1, pode alterá-los a fim de adequar ao cenário de utilização que desejar. A estrutura da tabela criada está representada através da figura 13 com os comentários que explicam a finalidade de cada campo logo abaixo:

	Campo	Tipo	Collation	Atributos	Nulo	Padrão	Extra
<input type="checkbox"/>	<u>user1</u>	varchar(50)	utf8_general_ci		Não	None	
<input type="checkbox"/>	<u>disci</u>	varchar(50)	utf8_general_ci		Não	None	
<input type="checkbox"/>	<u>activ</u>	tinyint(1)			Não	None	
<input type="checkbox"/>	<u>tare1</u>	tinyint(1)			Não	None	
<input type="checkbox"/>	<u>aler1</u>	tinyint(1)			Não	None	
<input type="checkbox"/>	<u>dias1</u>	varchar(11)	utf8_general_ci		Não	None	
<input type="checkbox"/>	<u>aler2</u>	tinyint(1)			Não	None	
<input type="checkbox"/>	<u>dias2</u>	varchar(11)	utf8_general_ci		Não	None	
<input type="checkbox"/>	<u>foru1</u>	tinyint(1)			Não	None	
<input type="checkbox"/>	<u>aler3</u>	tinyint(1)			Não	None	
<input type="checkbox"/>	<u>dias3</u>	varchar(11)	utf8_general_ci		Não	None	

Figura 13 - Estrutura da tabela criada¹

- USER1¹ – contém o nome do usuário ao qual aquela configuração está atrelada.
- DISCI¹ – contém o nome da disciplina à qual a configuração está atrelada.

¹ - O nome do campo/tabela foi definido arbitrariamente. Pode ser alterado para qualquer outro texto, como for conveniente à instalação, desde que seja refletido na base de dados.

- ACTIV¹ – indica se, para a disciplina em questão, o envio de alertas está ativo.
- TARE1¹ – indica se, quando uma nova tarefa for criada pelo professor da disciplina em questão, o usuário deverá ser alertado.
- ALER1¹ – indica se o usuário deverá ser alertado quando restarem poucos dias (número indicado pelo próximo campo, DIAS1) para a entrega de uma atividade.
- DIAS1¹ – indica a quantidade de dias restantes antes do prazo final de participação do usuário em uma tarefa.
- ALER2¹ – indica se o usuário deverá ser alertado quando um novo chat for iniciado.
- DIAS2¹ – indica a quantidade de minutos antes do início de um chat que o usuário deverá receber alerta.
- FORU1¹ – indica se o usuário deverá ser alertado quando um novo fórum for criado.
- ALER3² – indica se o usuário deverá ser alertado quando restarem poucos dias (número indicado pelo próximo campo, DIAS3) para o prazo final de participação em um fórum.
- DIAS3² – indica a quantidade de dias antes do prazo final para participação em um fórum em que o usuário deverá ser alertado.

É importante observar que todos estes campos foram definidos arbitrariamente. Novos campos podem ser inseridos, assim como os já existentes podem ser alterados, substituídos, excluídos, etc. Esta operação pode ser executada livremente, desde que as alterações sejam replicadas no arquivo de configurações, descrito no item 5.2.

5.2 Integração entre a alteração na base de dados e a aplicação

Toda a base do sistema foi pensada para que houvesse uma integração eficaz, porém que fosse pequena e pouco complexa, a fim de

² - O nome do campo/tabela foi definido arbitrariamente. Pode ser alterado para qualquer outro texto, como for conveniente à instalação, desde que seja refletido na base de dados.

viabilizar que alterações simples façam com que a aplicação se adapte a diferentes cenários de diferentes softwares das instituições adquirentes.

Para esta tarefa, foi idealizado e criado um arquivo XML que faz o papel de interface entre um sistema e outro (Moodle e CADMM). Toda vez em que a aplicação é iniciada, seja via web ou via mobile, o arquivo é lido para que a aplicação saiba como acessar os dados que serão utilizados para interagir com o usuário. Este arquivo guarda as configurações que a aplicação deverá usar para funcionar, ou seja, através dele é definido o comportamento da aplicação como um todo.

O arquivo XML, que pode ser visto inteiramente no anexo 1, foi dividido em sessões, que são responsáveis pelo comportamento do sistema e estão especificadas abaixo, junto de suas propriedades e observações sobre o impacto de cada uma. As sessões que compõem o arquivo são: 5.2.1 – identificação da tabela que contém o cadastro de usuários; 5.2.2 – identificação da tabela que contém as configurações para cada usuário; 5.2.3 – identificação da maneira com que a aplicação deverá consultar os dados de cada usuário dentro das tabelas já existentes do Moodle; 5.2.4 – parâmetros gerais.

5.2.1 Identificação da tabela que contém o cadastro de usuários

A imagem 13 identifica o trecho de código do arquivo XML responsável por indicar ao sistema qual é a tabela que contém o cadastro de usuários. Esta tabela é utilizada para fazer a autenticação dos usuários diretamente na base do Moodle, de modo a não exigir que um cadastro adicional seja feito para utilizar a nova aplicação. Abaixo, está a explicação e o impacto de cada um dos parâmetros e comandos utilizados.

```
<table ID="userTable" NAME="mdl_user">
  <field type="USERNAME" isKeyField="true">username</field>
  <field type="PASSWORD" isKeyField="true">password</field>
  <field type="FIRSTNAME">firstname</field>
</table>
```

Figura 14 - Trecho de código XML responsável por indicar a tabela que contém o cadastro de usuários

- **Tag *table***

A tag *table* indica a identificação de uma tabela específica que será utilizada pela aplicação. Ela é reconhecida pela aplicação através do parâmetro ID.

- ID = indica qual é o papel desta tabela na aplicação. O texto *userTable* é fixo e sinaliza que este trecho identificará qual é a tabela que possui o cadastro de usuários e como ela deve ser acessada.
- NAME = indica o nome físico da tabela dentro do banco de dados do Moodle. O nome *mdl_user* é utilizado por padrão quando se instala o Moodle em alguma máquina, podendo ser alterado livremente.

- **Tag *field***

A tag *field* indica como se comportam os campos dentro de uma tabela no banco de dados. É responsável por dar informações à aplicação sobre como acessá-los.

- TYPE = indica qual é o papel deste campo na tabela. Os textos *USERNAME*, *PASSWORD* e *FIRSTNAME* são fixos e sinalizam quais são as colunas que contêm a informação de usuário, senha e nome do usuário respectivamente.
- ISKEYFIELD = indica que o campo em questão é uma chave primária na tabela, ou seja, não aceita registros duplicados. Este controle existe e é utilizado apenas internamente para que a aplicação não gere inconsistências no banco de dados.

As informações que ficam entre as tags *field* (*username*, *password* e *firstname*) representam os valores que os campos possuem. Por exemplo, o campo *USERNAME* será representado pela palavra *username* e, desta maneira, o sistema saberá em qual coluna da tabela do banco de dados deverá procurar o nome do usuário.

5.2.2 Identificação da tabela que contém as configurações para cada usuário

A imagem 15 foi reduzida a fim de proporcionar uma melhor visualização. Ela identifica o trecho de código do arquivo XML responsável por indicar a tabela que conterá as configurações salvas pelo usuário através da aplicação web.

Este trecho do arquivo é de fundamental importância à aplicação. Os dados que estão dispostos neste ponto, divididos por grupos, indicam quais serão as informações que o usuário visualizará na tela onde eleger suas preferências para alertas, definem como serão as características que os campos terão e identificam os campos onde as informações serão gravadas no banco de dados. O grupo *identification* não é exibido na tela de configurações durante sua montagem, pois serve apenas para manter informações básicas para operações internas da aplicação. Os outros grupos ficam presentes na tela de configurações, que será exibida e terá seu funcionamento detalhado no item 7.

Abaixo, está a explicação e o impacto de cada um dos parâmetros utilizados.

```
<table ID="configTable" NAME="z_config">
  <group NAME="identification">
    <field type="USERNAME" isKeyField="true">user1</field>
    <field type="COURSE" isKeyField="true" isCourseName="true">disci</field>
    <field type="ACTIVE" isActiveFlag="true">activ</field>
  </group>
  <group NAME="Tareas">
    <field type="TAREFAS" showMode="checkBox" text="Nueva tarea se crea">tare1</field>
    <field type="ALERT1" showMode="checkBox" text="Alertar para la entregas de tareas">aler1</field>
    <field type="DAYS1" showMode="text" inputLength="1" text="Dias para alertar antes de la expiraci<
  </group>
  <group NAME="Chats">
    <field type="ALERT2" showMode="checkBox" text="Alertar cuando un nuevo chat se inicia">aler2</fi<
    <field type="DAYS2" showMode="text" inputLength="1" text="Minutos de aviso antes del inicio de u<
  </group>
  <group NAME="Foros">
    <field type="FORUM" showMode="checkBox" text="Nuevo foro se crea">forul</field>
    <field type="ALERT3" showMode="checkBox" text="Alertar para participaciones en los foros">aler3<
    <field type="DAYS3" showMode="text" inputLength="1" text="Dias para alertar antes del final del >
  </group>
</table>
```

Figura 15 – Mapeamento da tabela que contém as informações sobre a tabela de configurações

- **Tag *table***

A tag *table* indica a identificação de uma tabela específica que será utilizada pela aplicação. Essa identificação é feita pela aplicação através do parâmetro ID.

- ID = indica qual é o papel desta tabela na aplicação. O texto *configTable* é fixo e sinaliza que este trecho identificará qual é a tabela responsável por guardar as configurações feitas pelo usuário através da aplicação web.
- NAME = indica o nome físico da tabela dentro do banco de dados do Moodle. O nome *z_config* foi dado arbitrariamente e pode ser alterado a qualquer momento, desde que esteja sempre igual ao nome existente no banco de dados.

- **Tag *group***

A tag *group* indica a delimitação de um grupo de campos a ser exibido na tela de configurações web. São agrupados os campos que possuem propósitos semelhantes, com o objetivo de facilitar a compreensão do usuário.

- NAME = indica qual será o nome que o grupo receberá na tela de configurações. Apenas o grupo que possuir o atributo como *identification* não será exibido ao usuário. Este grupo possui informações que são utilizadas para operações internas da aplicação.

- **Tag *field***

A cada tag *field*, há a indicação de uma linha na tela de configurações com a qual o usuário poderá visualizar e/ou interagir.

- *type* = neste contexto, o campo *type* é idêntico ao descrito no item 5.2.1. Ele indica qual campo da tabela do banco de dados está representando.

- *isKeyField* = neste contexto, o campo *iskeyfield* é idêntico ao descrito no item 5.2.1. Ele indica se o campo da tabela do banco de dados é uma chave primária ou não (utilizado apenas para controle interno).
- *isCourseName* = esta propriedade indica que o campo na tabela do banco de dados é a coluna que representará o nome do curso ao qual as configurações serão atreladas.
- *isActiveFlag* = indica que o campo na tabela do banco de dados é a coluna que representa o estado dos alertas para aquele curso: ativos ou inativos.
- *showMode* = a propriedade indica qual será o modo de captura das informações do usuário. Há dois tipos suportados: *checkBox* e *text*. O tipo *checkBox* apresenta apenas uma caixa para ser marcada ou desmarcada: . O tipo *text* apresenta um campo para que o usuário possa inserir valores: .
- *text* = esta propriedade indica qual é o texto que deverá acompanhar o campo.
- *inputLength* = esta propriedade só pode ser utilizada em campos que possuam a propriedade *showMode* definida como *text*. Através dela, pode-se determinar o tamanho que o campo possuirá na tela, a fim de deixá-la mais amigável ao usuário.

5.2.3 Determinação do acesso às tabelas já existentes do Moodle

A imagem 16 foi reduzida a fim de proporcionar uma melhor visualização ao leitor. Ela identifica o trecho de código do arquivo XML responsável por reconhecer a maneira com que a aplicação acessará os dados de cada usuário dentro das tabelas de dados já existentes do Moodle, como o registro de atividades, os chats e fóruns. Após a imagem, está a explicação de cada um dos parâmetros e atributos.

Através de atributos, este trecho define quais são as tabelas originais do Moodle que deverão ser acessadas para obter os diferentes dados

utilizados pelos professores como maneira de interação com os alunos, bem como os parâmetros que devem ser utilizados para o acesso e os campos que devem ser capturados como retorno.

```
<getData>
  <source ID="coursesForUser">
    <query>SELECT * FROM mdl_course c, mdl_user u, mdl_role_assignments
    <returnField type="ID">id</returnField>
    <returnField type="NAME">fullname</returnField>
  </source>
  <source ID="assignmentsForUser">
    <tableName>mdl_assignment</tableName>
    <whereCondition>course=?</whereCondition>
    <returnField type="ID">id</returnField>
    <returnField type="TIMEAVAILABLE">timeavailable</returnField>
    <returnField type="NAME">name</returnField>
    <returnField type="COURSE">course</returnField>
  </source>
  <source ID="forumsForUser">
    <tableName>mdl_forum</tableName>
    <whereCondition>course=? AND type='general'</whereCondition>
    <returnField type="ASSESSTIMEFINISH">assesstimefinish</returnField>
    <returnField type="COURSE">course</returnField>
    <returnField type="ID">id</returnField>
    <returnField type="NAME">name</returnField>
    <returnField type="TYPE">type</returnField>
  </source>
  <source ID="chatsForUser">
    <tableName>mdl_chat</tableName>
    <whereCondition>course=?</whereCondition>
    <returnField type="ID">id</returnField>
    <returnField type="CHATTIME">chattime</returnField>
    <returnField type="NAME">name</returnField>
    <returnField type="COURSE">course</returnField>
  </source>
</getData>
```

Figura 16 - Identificação das tabelas que deverão ser acessadas pela aplicação a fim de obter os dados originais do Moodle

- **Tag *source***

A tag *source* indica um grupo de características que serão utilizadas para obter dados das tabelas do Moodle. Esta tag contém a identificação de sua finalidade e também os passos que devem ser executados para obter os dados que ela possui.

- ID = identifica qual é a responsabilidade da sessão de características logo abaixo. Cada grupo *source* possui uma finalidade que é interpretada pela aplicação no momento de acessar as tabelas do Moodle. Por exemplo, o grupo que possui ID *chatsForUser* é responsável por determinar como a tabela original do Moodle, que possui o registro dos chats do usuário, deve ser acessada.
- **Tag *query***

A tag *query* é utilizada em apenas um ponto da aplicação. Esta tag - conforme mostrado na imagem 17 - indica exatamente qual é o comando que deve ser enviado ao banco de dados para obter os dados do usuário. O ponto em que ela é utilizada define o comando que a aplicação deve utilizar para obter as disciplinas em que um usuário está matriculado.

```
<query>SELECT * FROM mdl_course c, mdl_user u, mdl_role_assignments ra, mdl_conte
```

Figura 17 - Parte do código utilizado para selecionar os cursos em que o usuário está matriculado

Observação: adotou-se a opção de utilizar diretamente o comando completo através do arquivo para resolver o problema de encontrar as disciplinas em que o usuário está matriculado. Esta abordagem foi necessária devido ao fato de que a maneira de obter estes dados pode sofrer alterações a cada versão do Moodle.

- **Tag *returnField***

A tag *returnField* indica quais são os campos da tabela acessada que devem ser guardados após sua seleção. Através disto, são indicados à aplicação os dados que devem ser mantidos para que possam ser utilizados pelo usuário.

A informação que é colocada entre as tags *returnField*, como destacado pela imagem 18, indica qual o campo na tabela do banco de dados que contém a informação necessária à aplicação.

```
<returnField type="NAME">name</returnField>
```

Figura 18 - Indicação do campo que deve ser obtido

- *type* = indica qual o nome do campo (coluna do banco de dados) deve ser guardado. Neste contexto, o atributo *type* é fixo. Por exemplo: para que a aplicação consulte corretamente os chats para um determinado usuário, devem ser indicados os campos na tabela original do Moodle que possuem os dados, conforme demonstra a figura 13, de identificação (*ID*), horário do chat (*chattime*), nome do chat (*chatname*) e o curso a que está atrelado (*course*).
- **Tag *whereCondition***

A tag *whereCondition* é utilizada para indicar à aplicação qual o filtro deve ser aplicado no banco de dados para obter os dados da tabela que está sendo acessada. Através desta propriedade, o comando que é utilizado para acessar as tabelas no banco de dados do Moodle pode ser adequado através das diferentes versões do Moodle, de maneira a não exigir alterações na aplicação.

- **Tag *tableName***

A tag *tableName* é utilizada para identificar o nome da tabela que deve ser acessada para a obtenção dos dados. Através desta propriedade, o nome da tabela acessada pode ser alterado a qualquer momento, não exigindo alterações na aplicação para adaptá-la às diferentes versões existentes do Moodle.

5.2.4 Parâmetros gerais

A imagem 19 identifica o trecho de código do arquivo XML responsável por guardar parâmetros gerais da aplicação. Neste momento, apenas um parâmetro é utilizado neste trecho, o qual fica livre para que futuras implementações possam inserir parâmetros genéricos na aplicação. Abaixo, está a explicação da influência dos atributos na aplicação.

```
<generalParameters>
  <parameter NAME="securityPassKey">B##8quk (yQ5=sPb_UlovNhST&030</parameter>
</generalParameters>
```

Figura 19 - Parâmetros gerais da aplicação

- **Tag *parameter***

A tag *parameter* é utilizada para guardar parâmetros genéricos utilizados pela aplicação; o único utilizado nesta versão da implementação é o *securityPassKey*. Este parâmetro possui uma informação representada através de um texto aleatório (conforme destacado pela figura 20), que é utilizado para realizar a autenticação do usuário na base de dados do Moodle, a fim de não exigir um segundo cadastro para utilização da aplicação.

É importante ressaltar que este texto aleatório não é definido por este projeto. A cada vez que uma instalação do Moodle é realizada, um novo texto é criado pelo instalador, o que exige que o arquivo de configurações em questão seja alterado.

```
<parameter NAME="securityPassKey">B##8quk (yQ5=sPb_UlovNhST&030</parameter>
```

Figura 20 - Texto utilizado para a autenticação no servidor

6. Usuários do sistema

Este capítulo objetiva demonstrar de forma clara como cada papel/perfil de usuário atua sobre este projeto. Cada usuário possui um tipo de responsabilidade para que o ciclo do fluxo de dados seja completado.

Há três perfis de usuário que são presentes nesta aplicação. Eles estão dispostos abaixo, juntamente com uma explicação sobre a atuação de cada um.

6.1 Administrador da aplicação

O papel de administrador da aplicação será desempenhado pelo administrador do banco de dados do Moodle e também pelo responsável pela instalação do sistema. Este papel será capaz de realizar eventuais alterações que sejam necessárias no arquivo apresentado no item 5. Todas as alterações que forem feitas na base de dados do Moodle deverão ser reproduzidas neste arquivo, que ficará sob cuidado do administrador. A atividade do administrador pode ser representada pelo caso de uso da imagem 21 abaixo:

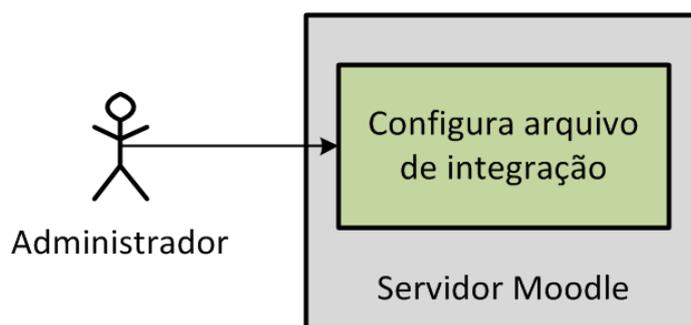


Figura 21 - Caso de uso do administrador do sistema

A responsabilidade do administrador inicia-se no momento em que o Moodle é instalado. Durante a instalação automática do sistema, várias

perguntas são feitas ao profissional que o está instalando. Através destas perguntas, surgem os dados para preencher os campos no arquivo XML.

Os atributos do arquivo XML que devem ser preenchidos no momento da instalação do Moodle e, após isto, serem mantidos atualizados são (conforme anexo 1):

- Nome da tabela que contém os dados dos usuários – parâmetro NAME da linha 2:

```
<table ID="userTable" NAME="mdl_user">
```

- Nome da coluna que contém a identificação do usuário cadastrado – conteúdo da tag field da linha 3:

```
<field type="USERNAME" isKeyField="true">username</field>
```

- Nome da coluna que contém a senha do usuário cadastrado – conteúdo da tag field da linha 4:

```
<field type="PASSWORD" isKeyField="true">password</field>
```

- Nome da coluna que contém o nome do usuário cadastrado – conteúdo da tag field da linha 5: <field type="FIRSTNAME">firstname</field>

- Comando para acesso às tabelas que guardam os cursos em que o usuário está matriculado. Esta informação pode ser alterada a cada versão do Moodle e está disponível no site da mantenedora – conteúdo da tag *query* da linha 33:

```
<source ID="coursesForUser">
  <query>SELECT * FROM mdl_course c, mdl_user u, mdl_role_assignments ra, mdl_context con,
  mdl_role r WHERE u.ID = ra.userID AND ra.contextID = con.ID AND con.contextlevel = 50
  AND con.instanceID = c.ID AND ra.roleID = r.ID AND r.shortNAME = 'student'
  AND u.userName = ?</query>
  <returnField type="ID">id</returnField>
  <returnField type="NAME">fullname</returnField>
</source>
```

- Nome da coluna que contém a identificação da disciplina em que o usuário está cadastrado – conteúdo da tag *returnField* da linha 37:

```
<returnField type="ID">id</returnField>
```

- Nome da coluna que contém o nome da disciplina em que o usuário está cadastrado – conteúdo da tag *returnField* da linha 38:

```
<returnField type="NAME">fullname</returnField>
```

- Texto utilizado para realizar a autenticação do usuário na base de dados do Moodle – conteúdo da tag *parameter* da linha 67:

```
<parameter NAME="securityPassKey">B##8quk(yQ5=sPb_UlovNhST#030</parameter>
```

A partir deste ponto, o preenchimento pode ser alterado, pois o conteúdo do arquivo tem a possibilidade de ser modificado de acordo com as necessidades de cada instituição implementadora. Portanto, abaixo será mostrado o preenchimento sugerido como padrão para as principais atribuições aos usuários: atividades, fóruns e chats.

Na imagem 22, está representado o trecho do código responsável por definir como a tabela que possui as atividades relacionadas aos usuários deve ser acessada.

```
<source ID="assignmentsForUser">
  <tableName>mdl_assignment</tableName>
  <whereCondition>course=?</whereCondition>
  <returnField type="ID">id</returnField>
  <returnField type="TIMEAVAILABLE">timeavailable</returnField>
  <returnField type="NAME">name</returnField>
  <returnField type="COURSE">course</returnField>
</source>
```

Figura 22 - Dados para acesso à tabela de atividades

- Nome da tabela que contém as atividades dos usuários – conteúdo da tag *tableName* da linha 41:

```
<tableName>mdl_assignment</tableName>
```

- Condição que deve ser utilizada para acessar a tabela – conteúdo da tag *whereCondition* da linha 42:

```
<whereCondition>course=?</whereCondition>
```

- Nome da coluna que contém o número de identificação da atividade – conteúdo da tag *returnField* da linha 43:

```
<returnField type="ID">id</returnField>
```

- Nome da coluna que contém o tempo que resta ao usuário para o envio da tarefa – conteúdo da tag *returnField* da linha 44:

```
<returnField type="TIMEAVAILABLE">timeavailable</returnField>
```

- Nome da coluna que contém o nome da atividade – conteúdo da tag *returnField* da linha 45:

```
<returnField type="NAME">name</returnField>
```

- Nome da coluna que contém o nome da disciplina à qual a atividade está atrelhada – conteúdo da tag *returnField* da linha 46:

```
<returnField type="COURSE">course</returnField>
```

Na imagem 23, está representado o trecho do código responsável por definir como a tabela que possui os fóruns relacionados aos usuários deve ser acessada.

```
<source ID="forumsForUser">
  <tableName>mdl_forum</tableName>
  <whereCondition>course=? AND type='general'</whereCondition>
  <returnField type="ASSESSTIMEFINISH">assesstimefinish</returnField>
  <returnField type="COURSE">course</returnField>
  <returnField type="ID">id</returnField>
  <returnField type="NAME">name</returnField>
  <returnField type="TYPE">type</returnField>
</source>
```

Figura 23 - Dados para acesso à tabela de fóruns

- Nome da tabela que contém os fóruns dos usuários – conteúdo da tag *tableName* da linha 49:

```
<tableName>mdl_forum</tableName>
```

- Condição que deve ser utilizada para acessar a tabela de fóruns – conteúdo da tag *whereCondition* da linha 50:

```
<whereCondition>course=? AND type='general'</whereCondition>
```

- Nome da coluna que contém a data limite para participação no fórum – conteúdo da tag *returnField* da linha 51:

```
<returnField type="ASSESSTIMEFINISH">assesstimefinish</returnField>
```

- Nome da coluna que contém a identificação do curso ao qual o fórum pertence – conteúdo da tag *returnField* da linha 52:

```
<returnField type="COURSE">course</returnField>
```

- Nome da coluna que contém a identificação do fórum – conteúdo da tag *returnField* da linha 53:

```
<returnField type="ID">id</returnField>
```

- Nome da coluna que contém o nome do fórum – conteúdo da tag *returnField* da linha 54:

```
<returnField type="NAME">name</returnField>
```

- Nome da coluna que contém o tipo do fórum – conteúdo da tag *returnField* da linha 55:

```
<returnField type="TYPE">type</returnField>
```

Na imagem 24, está representado o trecho do código responsável por definir como a tabela que possui os chats relacionados aos usuários deve ser acessada.

```
<source ID="chatsForUser">
  <tableName>mdl_chat</tableName>
  <whereCondition>course=?</whereCondition>
  <returnField type="ID">id</returnField>
  <returnField type="CHATTIME">chattime</returnField>
  <returnField type="NAME">name</returnField>
  <returnField type="COURSE">course</returnField>
</source>
```

Figura 24 - Dados para acesso à tabela de chats

- Nome da tabela que contém os chats dos usuários – conteúdo da tag *tableName* da linha 58:

```
<tableName>mdl_chat</tableName>
```

- Condição que deve ser utilizada para acessar a tabela de chats – conteúdo da tag *whereCondition* da linha 59:
`<whereCondition>course=?</whereCondition>`
- Nome da coluna que contém a identificação do chat – conteúdo da tag *returnField* da linha 60:
`<returnField type="ID">id</returnField>`
- Nome da coluna que contém a data e horário em que o chat acontecerá – conteúdo da tag *returnField* da linha 61:
`<returnField type="CHATTIME">chattime</returnField>`
- Nome da coluna que contém o nome do chat – conteúdo da tag *returnField* da linha 62:
`<returnField type="NAME">name</returnField>`
- Nome da coluna que contém o número do curso ao qual o chat está atrelado – conteúdo da tag *returnField* linha 63:
`<returnField type="COURSE">course</returnField>`

Na imagem 25, está representado o trecho do arquivo que armazena o texto padrão utilizado para autenticação na base de dados. Este texto deve ser preenchido conforme o que pode ser encontrado no arquivo *config.php*, existente no diretório em que o Moodle fica instalado, como demonstra a imagem 26.

```
<generalParameters>
  <parameter NAME="securityPassKey">B#%8quk (yQ5=sPb UlovNhST%030</parameter>
</generalParameters>
```

Figura 25 - Texto utilizado para autenticação no banco de dados

```

1  <?php  /// Moodle Configuration File
2
3  unset ($CFG);
4
5  $CFG = new stdClass();
6  $CFG->dbtype      = 'mysql';
7  $CFG->dbhost      = 'localhost';
8  $CFG->dbname      = 'moodle';
9  $CFG->dbuser      = 'root';
10 $CFG->dbpass      = '';
11 $CFG->dbpersist   = false;
12 $CFG->prefix      = 'mdl_';
13
14 $CFG->wwwroot     = 'http://localhost';
15 $CFG->dirroot     = 'C:\wamp\www';
16 $CFG->dataroot    = 'C:\wamp\moodledata';
17 $CFG->admin       = 'admin';
18
19 $CFG->directorypermissions = 00777; // try 02777 on a server in Safe Mode
20
21 $CFG->passwordsaltmain = 'B#%8quk(yQ5=sPb_UlovNhST%030';
22
23 require_once("$CFG->dirroot/lib/setup.php");
24 // MAKE SURE WHEN YOU EDIT THIS FILE THAT THERE ARE NO SPACES, BLANK LINES,
25 // RETURNS, OR ANYTHING ELSE AFTER THE TWO CHARACTERS ON THE NEXT LINE.
26 ?>

```

Figura 26 - Arquivo de configurações do Moodle (*config.php*)

Além destas configurações, o administrador também tem a responsabilidade de criar a tabela que armazenará as informações salvas pelo usuário na tela de configurações. A estrutura da tabela deverá ser discutida juntamente aos responsáveis pela ferramenta, a fim de obter uma estrutura que se adapte às expectativas da instituição utilizadora.

A sugestão de campos para criação da tabela que se adequa perfeitamente ao arquivo também sugerido foi representada pela figura 12.

6.2 Professor

O papel de professor aplicado neste contexto é o de um professor que utilize o Moodle em qualquer uma de suas versões, com ou sem a adição do trabalho desenvolvido. Pode ser representado pelo caso de uso da imagem abaixo (figura 27):

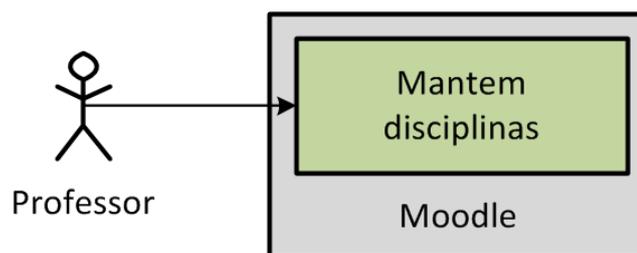


Figura 27 - Caso de uso do professor

O professor tem a responsabilidade de criar as disciplinas às quais os usuários têm acesso, criar e manter as atividades que são executadas em cada aula, assim como tarefas escritas, fóruns e chats aos quais os alunos deverão responder.

O site do projeto Moodle oferece um guia para que professores iniciantes na plataforma aprendam a utilizá-lo, conforme apresentado no artigo *Moodle for teachers and trainers* (2005).

Através do processo executado pelos professores, a aplicação desenvolvida começará a interagir automaticamente buscando os dados que foram cadastrados, a fim de enviá-los aos alunos quando aplicável.

6.3 Aluno

Para esta aplicação, as responsabilidades do aluno vão além de acompanhar as suas atividades através do Moodle. Elas podem ser visualizadas pelo caso de uso representado na imagem 28 abaixo:

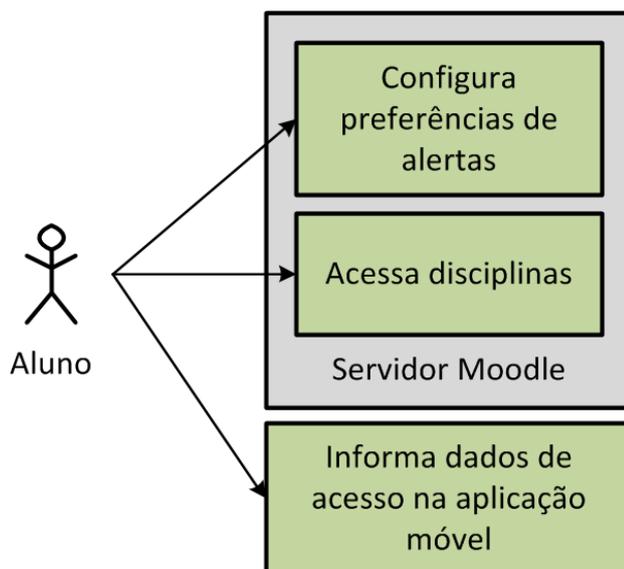


Figura 28 - Caso de uso do aluno

Originalmente, pelo Moodle, o aluno acompanha o desenvolvimento de suas atividades através das interfaces proporcionadas. Para que esta ferramenta seja utilizada, o aluno deverá configurar as suas preferências de alertas através da interface de configurações web, que será mostrada no item 7, para, a partir de então, começar a receber os devidos alertas através da aplicação Mobile, que será descrita no item 8.

Ao fazer uso da ferramenta pela primeira vez, o aluno deverá definir as suas preferências para cada uma das disciplinas em que está matriculado. Ele poderá ativar ou desativar os alertas para cada uma das disciplinas, individualmente, e também escolher quais tipos de alertas serão enviados, de maneira que passe a receber apenas os alertas considerados mais relevantes.

Quando um aluno é matriculado em uma nova disciplina, ele deve acessar a interface de configurações web e definir quais são os tipos de alertas que ele deseja receber para esta disciplina.

Com as configurações feitas, a aplicação utilizada no smartphone começará a enviar alertas para o usuário no momento em que forem necessários. O usuário receberá alertas baseados nas informações que guardou através da ferramenta web.

7. A aplicação de configurações (web)

Nesta seção, a tela de configurações web (imagens 29 e 30) utilizada pelos alunos será apresentada e a tela de login e os módulos que compõem a tela onde as configurações são definidas e guardadas serão detalhados.

A tela de configurações foi dividida a fim de proporcionar um entendimento mais claro sobre cada um de seus módulos e a exposição foi feita através dos seguintes itens: 7.1 – tela de login; 7.2 – módulo de ativação de disciplinas; 7.3 – módulo de seleção de disciplina e definição de configurações. A tela de configurações completa pode ser visualizada abaixo, através da imagem 21.

Cada explicação de módulo será acompanhada do esclarecimento sobre se há interferência na mesma e sobre qual parte do arquivo de integração, apresentado no item 5, está interferindo.

CONFIGURAÇÃO DE ALERTAS MOODLE

Bem-Vindo Guilherme!

Selecione a disciplina:

Ativar Desativar

Você deseja ser alertado quando:

Tareas

Nueva tarea se crea

Alertar para la entregas de tareas

Dias para alertar antes de la expiracion del prazo de entrega de una actividad:

Chats

Alertar cuando un nuevo chat se inicia

Minutos de aviso antes del inicio de un chat:

Foros

Nuevo foro se crea

Alertar para participaciones en los foros

Dias para alertar antes del final del periodo de participacion en un foro:

Ativar disciplinas

Definición de procesos

Arquitectura de software

© 2010 Unisinos. All Rights Reserved. Template Designed By: [Free Website Templates](#)

Figura 29 - Tela de configurações

7.1 Tela de login



Figura 30 - Tela de login

Esta tela possui a finalidade de autenticar o usuário para que ele possa acessar a tela de configurações. A autenticação é feita diretamente na base de dados já existente do Moodle, com o objetivo de que não haja necessidade de um cadastro adicional para a utilização da aplicação. Caso o usuário possua acesso ao Moodle, ele automaticamente possuirá acesso à aplicação.

O Moodle possui um padrão para realizar a autenticação na base de dados que se faz necessário devido ao fato de ser utilizado um algoritmo para encriptação dos dados baseado em MD5. A autenticação do usuário é feita da seguinte maneira: concatena-se a senha que o usuário informou com o texto padrão que deve ficar sinalizado no arquivo de configurações, sob a tag *securityPassKey*, mostrada no item 5.2.4.1. A partir do texto gerado, o algoritmo MD5 é aplicado e o resultado disto é confrontado com a informação existente na base de dados.

A partir do resultado da consulta na base de dados, a aplicação permite que o usuário acesse sua tela de configurações em caso de

autenticação positiva. Em caso negativo, a mensagem mostrada pela imagem 31 é exibida:

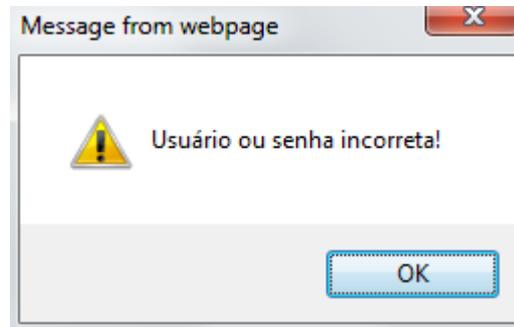


Figura 31 - Autenticação inválida

7.1.1 Relação com o arquivo de configurações

Esta tela tem seu funcionamento influenciado pelos seguintes trechos do arquivo de configurações:

- identificação da tabela a receber a autenticação do usuário, mostrada no item 5.2.1;
- texto a ser utilizado para confronto com a tabela descrita no item acima, mostrado no item 5.2.4.

7.2 Módulo de ativação de disciplinas



Figura 32 - Módulo de ativação de disciplina em estado inicial (primeira utilização)



Figura 33 – Módulo de ativação de disciplinas em utilização

Este módulo, representado pelas imagens 32 e 33, pertence à tela de configurações e permite ao usuário ativar ou desativar as disciplinas sobre as quais deseja ser alertado.

Os botões “Add” e “Rem”, que significam “adicionar” e “remover”, respectivamente, mostrados na figura 32, são exibidos a partir da seguinte regra: quando o usuário entra na tela de configurações, a aplicação verifica quais são as disciplinas em que ele está matriculado. Após isto, verifica quais são as que já possuem alertas ativados. A partir das informações obtidas, as disciplinas que possuem alertas ativados ficam apenas com o botão “Rem” habilitado. Já as que não possuem alertas ativados, ficam apenas com o botão “Add” habilitado.

O usuário pode, a qualquer momento, habilitar ou desabilitar uma disciplina. Quando uma disciplina é habilitada, automaticamente cria-se um registro na tabela de configurações, que será utilizado para enviar os alertas aos usuários. Quando uma disciplina é desabilitada, o registro é excluído e as configurações dos alertas (mostradas no item 7.3) armazenadas são perdidas. É importante ressaltar que os alertas enviados ao usuário, bem

como as atividades realizadas no Moodle, não são afetados. As configurações só são perdidas quando uma disciplina é desabilitada.

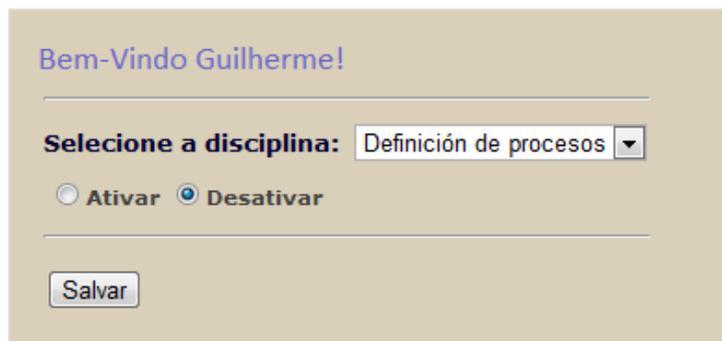
7.2.1 Relação com o arquivo de configurações

Esta tela tem seu funcionamento influenciado pelos seguintes trechos do arquivo de configurações:

- identificação da tabela as configurações dos usuários, identificada no item 5.2.2;
- identificação do acesso a ser feito para obtenção das disciplinas em que o usuário está matriculado, mostrado no item 5.2.3.2.

7.3 Módulo de seleção de disciplina e definição de configurações

Há duas visualizações para esta funcionalidade: quando a disciplina está desativada, 1) as configurações não são mostradas, conforme imagem 34:



Bem-Vindo Guilherme!

Selecione a disciplina: Definición de procesos ▼

Ativar Desativar

Salvar

Figura 34 - Módulo de seleção de disciplinas e definição de configurações. Disciplina de definição de processos com alertas desativados

Porém quando a disciplina está ativada, 2) a lista de configurações possíveis é mostrada, conforme imagem 35:

Bem-Vindo Guilherme!

Selecione a disciplina:

Ativar Desativar

Você deseja ser alertado quando:

Tareas

Nueva tarea se crea

Alertar para la entregas de tareas

Dias para alertar antes de la expiracion del prazo de entrega de una actividad:

Chats

Alertar cuando un nuevo chat se inicia

Minutos de aviso antes del inicio de un chat:

Foros

Nuevo foro se crea

Alertar para participaciones en los foros

Dias para alertar antes del final del periodo de participacion en un foro:

Figura 35 - Módulo de seleção de disciplinas e definição de configurações. Disciplina de definição de processos com alertas ativadas

A tela mostrada acima é utilizada pelo usuário para definição das suas configurações. Sua montagem e sua operação são influenciadas pelo arquivo de configurações.

A montagem da tela é feita de maneira dinâmica. Ela é criada a partir das informações existentes no arquivo de integração descrito no item 5.2, de acordo com as imagens 36 e 37. A tela sugerida acima está dividida em 3 sessões: tarefas, chats e fóruns. A sessão de tarefas será utilizada para explicar o funcionamento da tela.

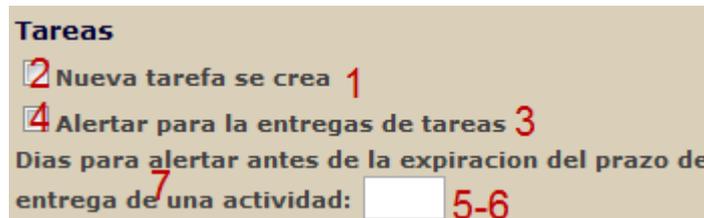
As imagens abaixo representam a identificação do grupo “Tarefas” no arquivo de integração e a montagem da mesma, respectivamente.

```

<group NAME="Tareas">
  <field type="TAREFAS" showMode="checkbox" text="Nueva tarea se crea">tare1</field>
  <field type="ALERT1" showMode="checkbox" text="Alertar para la entregas de tareas">aler1</field>
  <field type="DAYS1" showMod5="text" inputLength6="1" text="Dias para alertar antes
de la expiracion del prazo de entrega de una actividad:">dias1</field>
</group>

```

Figura 36 - Exemplo de identificação para montagem da tela de configurações



Tareas

Nueva tarea se crea

Alertar para la entregas de tareas

Dias para alertar antes de la expiracion del prazo de entrega de una actividad:

Figura 37 - Exemplo da tela de configurações montada a partir da definição no arquivo

A ordem em que os campos são apresentados na tela é definida pelo modo com que são dispostos no arquivo. A ordem em que estão apresentados no exemplo é uma sugestão para que a interface fique mais amigável ao usuário. Abaixo, segue a explicação da relação entre cada um deles, evidenciada nas imagens acima, separada por cada uma das configurações possíveis.

- 1 – O texto definido no arquivo de configurações como “*Nueva tarea se crea*” é exibido na tela.
- 2 – O tipo de exibição ao usuário, definido no arquivo como *checkbox*, é exibido na tela.
- 3 – O texto definido no arquivo como “*Alertar para la entrega de tareas*” é exibido como segunda configuração possível.
- 4 – O tipo de exibição ao usuário, definido como *checkbox*, é exibido na tela igualmente à configuração anterior.
- 5 – O tipo de exibição ao usuário, definido como *text*, é exibido na tela, criando um campo para que o usuário informe valores.
- 6 – O tamanho do campo *text* é delimitado na tela como 1.
- 7 – O texto definido no arquivo como “*Dias para alertar antes de la expiracion del prazo de entrega de una actividad*” é exibido

como a terceira configuração possível desta sessão.

- **Interação com o usuário**

A interação com o usuário nesta tela se dá através do preenchimento dos campos de configurações possíveis.

Abaixo, segue uma explicação das operações que o aluno precisa reproduzir para definir preferências para uma disciplina e, a partir disto, começar a receber os alertas.

- A primeira ação é definir qual das disciplinas que foram ativadas, através do módulo apresentado no item 7.2 (representado pela imagem 38), será utilizada. Neste exemplo de utilização, a disciplina de “Definição de Processos” será utilizada. Isto deve ser feito através da seleção do nome da mesma, conforme a imagem 38:

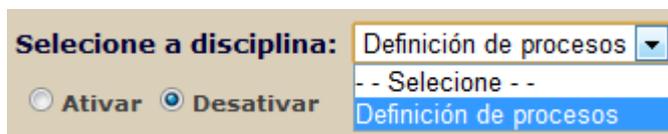


Figura 38 - Seleção da disciplina a ser configurada

- A segunda ação é ativar os alertas para a disciplina selecionada, através da marcação da opção de “Ativar” (imagem 39):

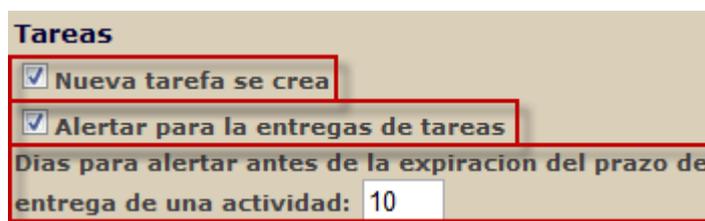


Figura 39 - Ativação dos alertas para a disciplina

- Ao sinalizar que os alertas devem ser ativados para a disciplina em utilização, a tela de configurações é automaticamente montada.

Os próximos campos são completados com base nas sessões que são preenchidas no arquivo de integração. Estas sessões podem ser alteradas de acordo com as necessidades da instituição que adotar a ferramenta, porém o preenchimento será feito de acordo com o arquivo sugestionado.

- Após ativar os alertas, a sessão de tarefas deve ser preenchida conforme os interesses do usuário. Com a imagem 40, segue uma sugestão de preenchimento que fará com que: a) o usuário seja alertado quando uma nova atividade for criada; b) o usuário seja alertado alguns dias antes da data de entrega da atividade; c) o usuário seja alertado 10 dias antes da data de entrega da atividade (completando o item b):



Tareas

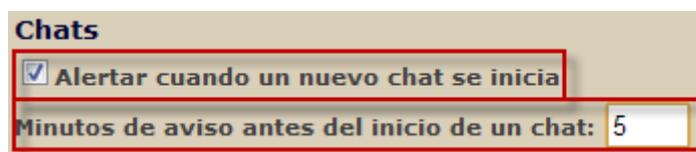
Nueva tarea se crea

Alertar para la entregas de tareas

Dias para alertar antes de la expiracion del prazo de entrega de una actividad: 10

Figura 40 - Sugestão de preenchimento para alertas de tarefas

- Após o preenchimento das tarefas, o campo da sessão de Chats deve ser preenchido. A sugestão da imagem 41 fará com que: a) o usuário seja alertado quando um chat for iniciado; b) o usuário seja alertado 5 minutos antes do início dos chats:



Chats

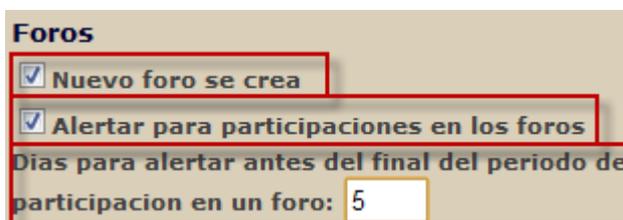
Alertar cuando un nuevo chat se inicia

Minutos de aviso antes del inicio de un chat: 5

Figura 41 - Sugestão de preenchimento para alertas de chats

- Quando os campos dos chats já estiverem completados, deve ser feito o preenchimento da sessão que possui as configurações para fóruns. A imagem 42 sugere o preenchimento que fará com que: a) o usuário seja alertado quando um novo fórum for criado; b) o usuário seja alertado quando restarem poucos dias para efetuar

sua participação no fórum; c) o usuário seja alertado 5 dias antes do fim do prazo para efetuar sua participação em um fórum (complementando o item b):



Foros

Nuevo foro se crea

Alertar para participaciones en los foros

Dias para alertar antes del final del periodo de participacion en un foro: 5

Figura 42 - Sugestão de preenchimento para alertas de fóruns

Após definir suas configurações ou ativar/desativar alertas para uma determinada disciplina, o usuário deve clicar no botão “Salvar” para que os dados sejam armazenados. Quando a operação for executada com sucesso, a mensagem exibida através da imagem 43 será apresentada:

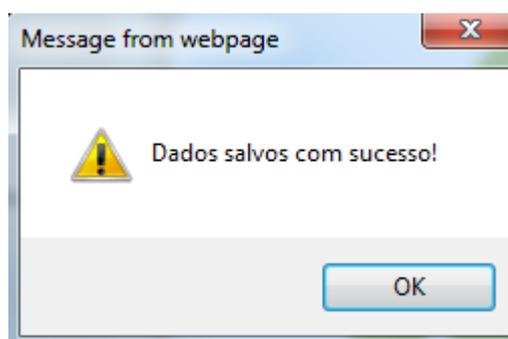


Figura 43 - Imagem de sucesso ao salvar dados do usuário

Caso haja algum problema durante a operação de salvar dados, é exibida uma mensagem informativa (representada pela imagem 44):

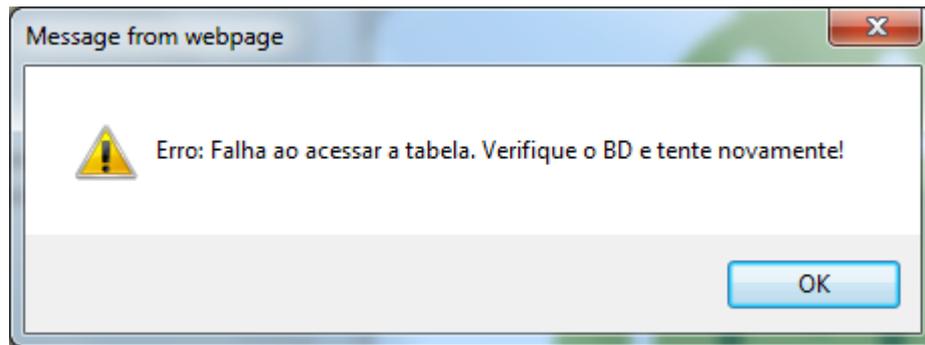


Figura 44 - Imagem de erro durante operação de salvar dados

7.3.2 Relação com o arquivo de configurações

Esta tela tem seu funcionamento influenciado pelo seguinte trecho do arquivo de configurações:

- identificação da tabela das configurações dos usuários, identificada no item 5.2.2.

7.4 Webservice utilizado pela aplicação móvel

Dentro da aplicação web, foi desenvolvido um webservice responsável por obter os dados do usuário que está requisitando informações e retornar os alertas que ele deve receber no momento. O desenvolvimento foi feito na parte servidor da aplicação, pois é necessário consultar as tabelas de dados do Moodle.

O webservice é chamado por cada aparelho Android que opera com o sistema. Esta chamada possui os intervalos alterados segundo as configurações definidas pelo usuário conforme explicado na sessão 8.1.

O arquivo de configurações é utilizado para identificar as tabelas que deve acessar durante seu processamento, que é dividido nas seguintes operações:

- obtenção de cursos em que o usuário está matriculado;

Conforme definição do arquivo:

- obtenção de atividades destinadas ao usuário;
- obtenção de fóruns destinados ao usuário;
- obtenção de chats destinados ao usuário;
- agrupamento de informações e retornar ao dispositivo móvel.

7.4.1 Influências do arquivo de configuração

Esta tela tem seu funcionamento influenciado pelos seguintes trechos do arquivo de configurações:

- identificação da tabela que contém o cadastro de usuários. Item 5.2.1;
- identificação da tabela que contém as configurações dos usuários. Item 5.2.2;
- identificação da maneira com que a aplicação consulta os dados de cada usuário dentro das tabelas já existentes do Moodle. Item 5.2.3;
- parâmetros gerais da aplicação. Item 5.2.4.

8. A aplicação mobile

Nesta seção, a aplicação móvel para dispositivos Android será apresentada. Ela é utilizada pelos alunos no momento de receber alertas pelos quais eles estejam interessados e já tenham manifestado interesse em recebê-los através da aplicação web mostrada no item 7. Está representado através da imagem 45 o fluxo de informações da aplicação móvel, que será detalhada nesta sessão:

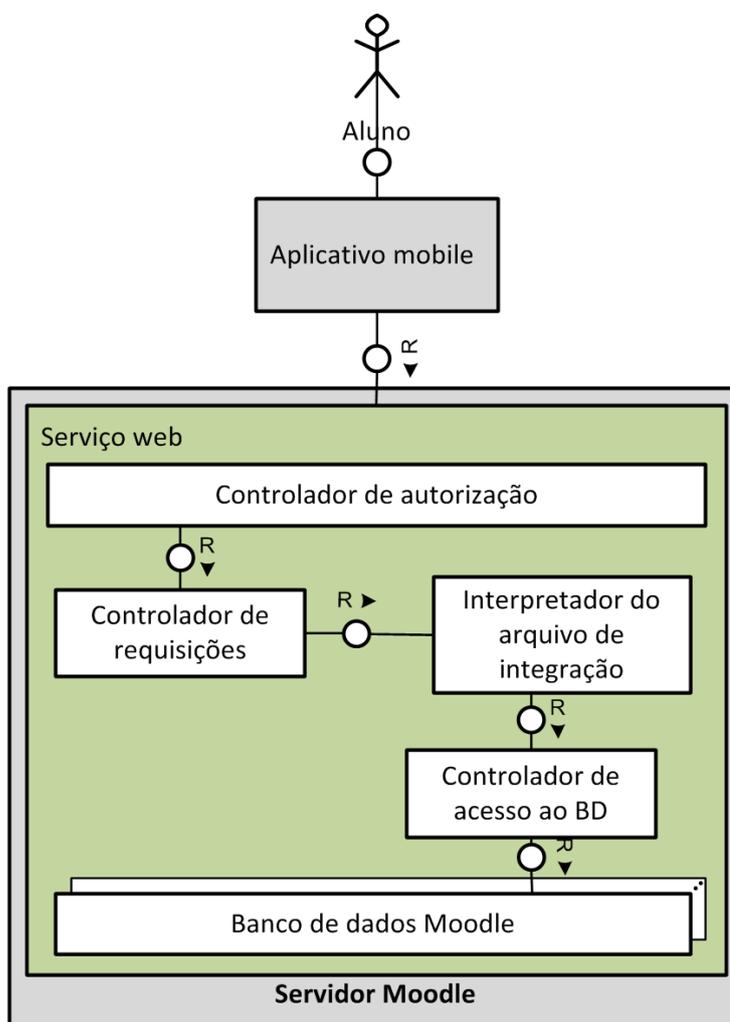


Figura 45 - Fluxo de funcionamento da aplicação móvel

Com a imagem 46, podemos perceber o fluxo de informações que surge com a implementação da arquitetura:

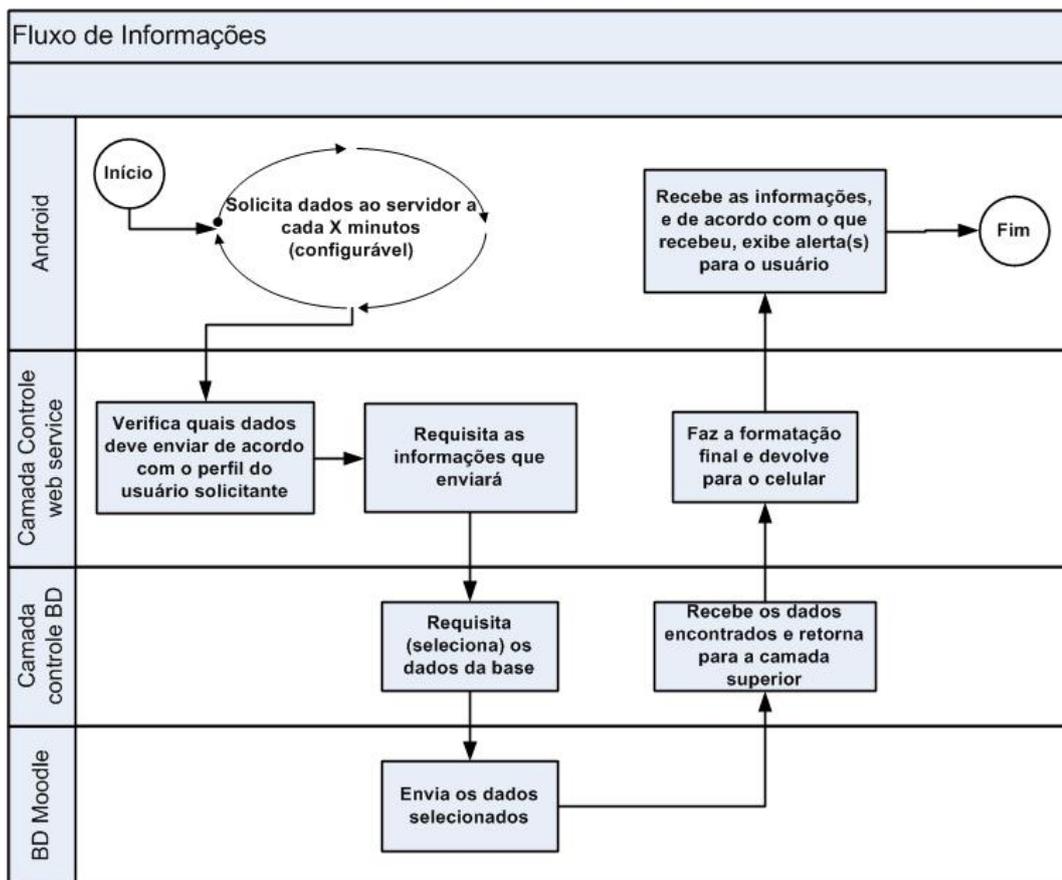


Figura 46 - Fluxo de informações da aplicação móvel

As telas do sistema foram separadas para proporcionar um entendimento mais eficaz sobre cada uma delas. A explanação foi organizada da seguinte maneira: a) 8.1 – tela de configurações do dispositivo móvel; b) 8.2 – recebimento de alertas; c) 8.3 – visualização dos alertas.

Cada um dos itens descritos contará com os seguintes subitens, que ajudarão a descrever a usabilidade da aplicação: a) imagem - mostrará a imagem da tela que está sendo explicada; b) operação - explicará e exemplificará o uso da tela; c) influências do arquivo de configuração - mostrará quais pontos do arquivo de integração influenciam no funcionamento da tela em questão.

Foi disponibilizado um arquivo reconhecido pelo Android para a instalação da aplicação. O arquivo deve ser enviado ao celular e, após isto, a instalação deve ser feita manualmente. Depois de instalada, a aplicação

pode ser acessada normalmente através do menu do Android, sob o nome de “Alertas Moodle”, conforme a imagem 47 representa:

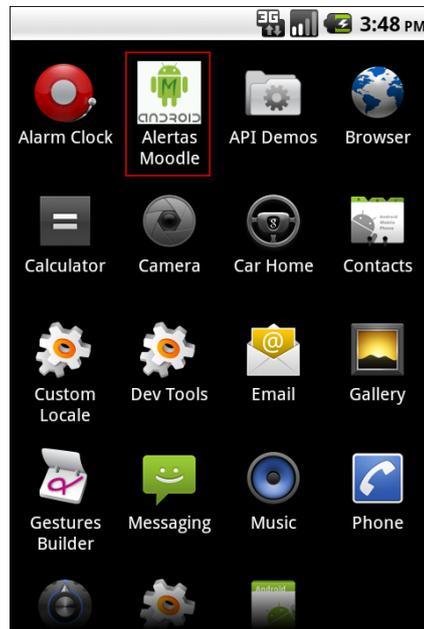


Figura 47 - Apresentação da aplicação no menu padrão do Android

8.1 Tela de configurações do dispositivo móvel



Figura 48 - Tela de configurações do dispositivo móvel

A tela de configurações da aplicação móvel (representada pela imagem 48) se fez necessária por dois motivos: 1 – através dela, algumas configurações podem ser informadas para que os alertas sejam obtidos em momentos convenientes ao aluno (como a opção de tempo de atualização, que define o intervalo de tempo entre uma e outra consulta ao banco e a opção de ativação do recebimento de alertas, que indica que o dispositivo deve buscar alertas ou não); 2 – outras delas são necessárias para que a aplicação saiba qual servidor deverá acessar para obter os alertas do aluno (como o endereço do servidor que deverá ser consultado e os dados de acesso, que devem ser informados a fim de fazer a autenticação no servidor do Moodle e então obter os dados).

Cada configuração da aplicação móvel possui uma finalidade. A interação do usuário com as opções ocorre via ferramentas nativas do Android, que proporcionam a criação de mensagens e telas que sejam amigáveis ao usuário:

- **Tempo de atualização:** A configuração que define o tempo de atualização da aplicação, quando selecionada, exibe a tela

representada pela imagem 49 para que o usuário selecione a configuração que lhe for mais conveniente:



Figura 49 - Configuração de intervalo de tempo de atualização

Esta configuração define qual será o intervalo de tempo em que a aplicação acessará a base de dados para buscar os alertas ao usuário. Ela foi criada levando em consideração que as atualizações no Moodle não são frequentes o suficiente para que seja necessária uma atualização automática e simultânea.

No momento da primeira utilização da aplicação, o usuário deve escolher um dos valores pré-definidos para que seja o intervalo de tempo entre as procuras de dados no servidor.

- **Ativar alertas:** A configuração que define se os alertas devem ser buscados por aquele dispositivo, quando selecionada, exibe a tela representada pela imagem 50 para que o usuário defina sua preferência.



Figura 50 - Ativar ou desativar a busca de alertas no dispositivo

Esta configuração define se o dispositivo deve realizar comunicação com o servidor do Moodle, onde a aplicação web deve estar instalada, para buscar informações sobre as disciplinas.

Esta configuração foi pensada para ser tratada como um botão de fácil acesso ao usuário para que ele possa sinalizar, também no nível do dispositivo móvel, se deseja que os alertas sejam buscados.

- **Dados de acesso:** A configuração que altera e guarda os dados de acesso ao usuário é efetuada através do toque na linha de “Dados de acesso”, como mostra a imagem 51:

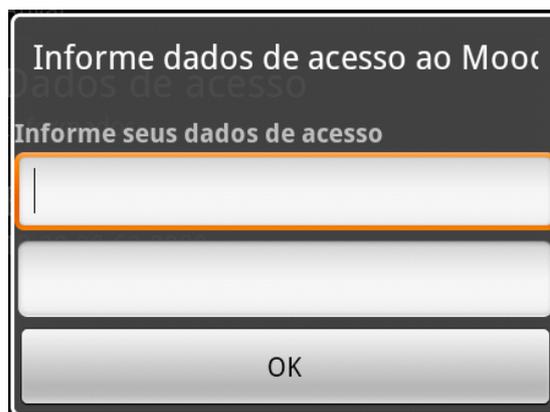


Figura 51 - Informar dados de acesso ao servidor

Através desta configuração, o usuário deve informar os dados que utiliza para acesso ao Moodle e à tela de configurações na web. Os dados serão utilizados para autenticação no Moodle,

de modo que a aplicação verifique dados apenas do usuário que a está utilizando.

É essencial à aplicação móvel que estes dados sejam informados corretamente. Caso eles não sejam informados ou haja algum problema na autenticação no servidor, a busca por alertas não será efetuada.

- **Endereço do servidor:** A configuração que guarda o endereço do servidor pode ser alterada conforme a tela exibida através da imagem 52:

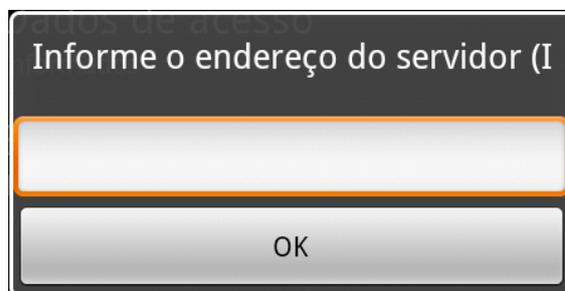


Figura 52 - Configuração do endereço do servidor

Nesta tela, o usuário deve informar o endereço IP do servidor que deve ser acessado para a procura de alertas. Esta configuração se fez necessária por facilitar a troca do local onde estão os dados. Com esta configuração, a cada novo local em que a aplicação for implantada, apenas uma pequena alteração precisará ser feita pelos alunos para que possam iniciar a utilização. É esperado que esta configuração seja feita apenas uma vez, quando os servidores já estiverem estabelecidos permanentemente. Deste modo, é pretendido não causar transtorno aos alunos, fazendo-os alterar mais uma configuração.

Após a definição de cada uma das configurações, é exibida uma mensagem de sucesso conforme a imagem 53 apresenta:

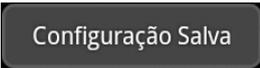
Um botão retangular com o texto "Configuração Salva" em uma fonte sans-serif, centralizado na página.

Figura 53 - Informativo ao usuário

8.1.1 Relação com o arquivo de configurações

Neste ponto da aplicação, não há influências do arquivo de configurações.

8.2 Recebimento de alertas

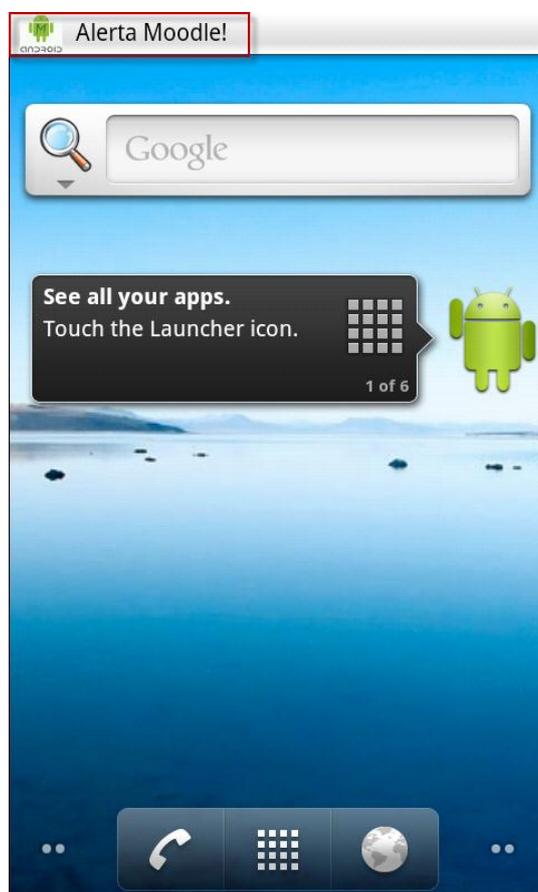


Figura 54 - Tela inicial do Android recebendo um alerta

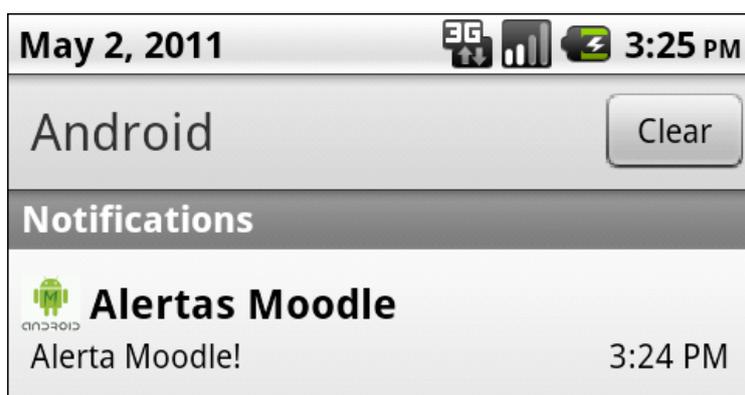


Figura 55 - Alerta detalhado

As imagens 54 e 55 mostram o recebimento de alertas pelo smartphone. Este recebimento é feito através de conexão com a web. O

usuário precisa estar ligado à rede de alguma maneira, seja via 3G ou wireless comum.

A partir do momento em que a aplicação é iniciada, automaticamente iniciam-se as verificações a fim de dar início ao recebimento de alertas. A aplicação verificará, em primeira instância, se os parâmetros definidos na tela de configurações (mostrada no item 8.1) estão preenchidos corretamente. Caso os parâmetros permitam que a aplicação inicie a busca, é ativado o webservice (conforme descrito no item 7.4) responsável por dar as informações ao dispositivo móvel.

De acordo com a configuração de tempo de atualização definida pelo usuário, a aplicação móvel consulta o servidor que possui o Moodle e a aplicação web instalada. A cada consulta ao servidor que possui os dados, a aplicação realiza alguns filtros visando a identificar os alertas que devem ser informados ao usuário. Isto garante que estes alertas sejam exibidos corretamente, nas datas e horários que o usuário elegeu.

Tendo visto que algum alerta foi obtido, a aplicação utiliza ferramentas nativas do sistema operacional para enviar os alertas ao usuário (conforme exemplificado através da imagem 54). Por utilizar ferramentas do sistema, o alerta pode ter seu sinal (som de alerta) e volume alterados através das configurações nativas. Após o recebimento do alerta, o usuário pode abrir o menu de alertas do sistema e verificar quais alertas foram disparados (conforme imagem 55).

Ao visualizar os alertas que estão disponíveis a ele, o usuário pode abri-los, o que o levará à tela onde verá os novos alertas, incluindo o histórico de alertas que já possui. A tela onde o usuário visualiza os alertas recebidos e também o histórico está descrita no item 8.3.

8.2.1 Relação com o arquivo de configurações

Este ponto da aplicação utiliza os mesmos pontos do arquivo de configuração que o item 7.4, em virtude de usar o webservice desenvolvido.

8.3 Visualização dos alertas



Figura 56 - Tela da aplicação exibindo alertas

A tela acima (imagem 56), em que é possível ver os alertas que o usuário recebeu, pode ser visualizada de duas maneiras. O usuário pode vê-la ao abrir a aplicação através do menu principal do Android ou então após receber um alerta através da barra de alertas do sistema operacional.

A tela mostra os alertas que estão atrelados àquele usuário juntamente aos alertas já feitos anteriormente; são apresentados a título de histórico.

A cada alerta, possuímos 3 informações:

- nome da disciplina responsável por aquele alerta;
- nome da tarefa responsável por aquele alerta;
- data limite para entrega da atividade atrelada ao alerta.

8.3.1 Relação com o arquivo de configurações

Neste ponto da aplicação, não há influências do arquivo de configurações.

9. Conclusão

Através da explicação sobre a motivação e tecnologias utilizadas, foi desenvolvida uma aplicação robusta, com foco na configuração e na instalação, ambas feitas através de poucos passos. Com a divisão de papéis, foram definidos desde os tipos de usuários que podem interagir com as camadas de configuração até os que têm possibilidade de interação com as telas de utilização da aplicação e definição de preferências.

A partir da apresentação da aplicação e de sua instalação e configuração, foram mostradas todas as etapas do processo. A instalação e a configuração ficaram a cargo do usuário que possui perfil de administrador do banco de dados e do Moodle, os professores têm a responsabilidade de criar os materiais que serão visíveis aos alunos e os alunos, que são os maiores beneficiados pela aplicação, têm a responsabilidade de definir suas preferências através da tela web, objetivando filtrar os alertas que lhes são pertinentes e que serão recebidos futuramente através do dispositivo móvel.

Foi idealizado e desenvolvido um arquivo de integração, o qual contém a maioria das definições que a aplicação precisa para funcionar. A cada nova definição possível que surja durante o período de utilização da aplicação, vinda dos responsáveis pela aplicação, podem ser feitas alterações simples neste arquivo para que as novas definições sejam implementadas e a aplicação continue funcionando sem exigir alterações nas fontes na grande maioria dos casos.

Com a apresentação das aplicações web e móvel, foi visto que ambas possuem interface simples e de fácil interação por parte do aluno. São exigidos poucos passos para sua utilização, e também poucos passos para completá-las. Foi explicada e exemplificada a utilização de cada uma delas, de modo a poder servir como um guia para futuros usuários.

Os benefícios da utilização da aplicação são facilmente identificados para alunos e professores. Os alunos serão os maiores beneficiados, pois contarão com um sistema que os lembrará de suas obrigações em todas as disciplinas que desejarem. Com o uso da ferramenta por parte dos alunos, os

professores também serão beneficiados, visto que os alunos ficarão mais atentos ao que acontece nas disciplinas. Este benefício virá principalmente para as disciplinas onde não há encontros frequentes entre professores e alunos.

Tendo mostrado todos os passos de funcionamento da aplicação e também seus benefícios, conclui-se que a aplicação possui potencial para ser utilizada tanto em regime acadêmico, quanto em ambientes comerciais, em instituições que utilizem o Moodle por quaisquer necessidades.

10. Trabalhos futuros

A sessão de trabalhos futuros torna-se importante devido à continuidade que este projeto terá após sua conclusão. O projeto de mestrado MLEA integrará a ferramenta aos artefatos que forem desenvolvidos e utilizará todas as suas funcionalidades e também a estrutura já existente para continuar os desenvolvimentos iniciais.

Como ideias para trabalhos futuros, pode-se continuar os desenvolvimentos no Android, a fim de proporcionar mais alternativas de alertas para os usuários. A cada novo tipo de alerta que for necessário, deve-se verificar se apenas alterando o arquivo de integração, a melhoria pode ser implementada. Caso não seja possível, devido à intenção ser de implementar um novo tipo de alerta (os desenvolvidos foram fóruns, chats e atividades, conforme visto no item 5.2.3), será necessário um pequeno desenvolvimento na aplicação que provê o webservice, a fim de suportar os novos tipos de informações.

Trabalhos futuros também surgirão à medida que a aplicação for instalada em mais locais, visto que a cada nova instância instalada, surgem novas necessidades por parte da organização adquirente.

Bibliografia

CENTRO ATLÂNTICO. **O guia prático da HTML**. 2004.

CENTRO ATLÂNTICO. **O guia prático da XML**. Out 2001.

CLAROLINE. **Claroline Documentation**. 2011. Disponível em:

http://doc.claroline.net/en/index.php/Main_Page (acessado em 06/07/2011).

CRESPO, S.; BRATZ, L.; SERRÃO, T.; CLUNIE, G.. **MLEA - uma arquitetura baseada em OWL para a formação de ambientes móveis flexíveis de recomendação, interação e alarmes para usuários de uma plataforma EAD**. 2010.

CRESPO, S.; CLUNIE, G.; RANGEL, N.; CASTILLO, A.; RODRÍGUEZ, K.; GÓMEZ, B.. **Ambiente de apoio al aprendizaje móvil, EATIS**. 2010.

DUARTE, M. L; DOTTI, L. F., **Desenvolvimento de aplicações móveis corretas**. 2001.

ELLIS, Ryann K.. **Field Guide to Learning Management Systems**. ASTD Learning Circuits. 2009.

FELIZARDO, K.; AMARAL, M.; VEDOATO, R.. **O uso do moodle no apoio de ensino de programação para alunos iniciantes**. 13º Congresso Internacional de Educação a Distância. 2007.

GEORGIEV, T.; GEORGIEVA, E.; SMRIKAROV, A.. **M-Learning – a new stage of e-learning**. CompSysTech '04: Proceedings of the 5th international conference on Computer systems, 1-5, 2004.

GNU. **O que é GNU?** Disponível em: <http://www.gnu.org/home.pt-br.html> (acessado em 11/04/2011).

ILIAS. **Ilias documentation**. Disponível em http://www.ilias.de/docu/goto_docu_lm_1672.html (acessado em 06/07/2011).

KARANDIKAR, K. **Java in education**. [2011].

KORCUSKA, M; BERG, A. M.; **Sakai Courseware Management: The Official Guide**. 10-6,2004.

MIT LABORATORY FOR COMPUTER SCIENCE. **The MD5 Message-Digest Algorithm**, 1992.

new stage of e-learning. CompSysTech '04: Proceedings of the 5th international conference on Computer systems, 1-5, 2004.

MOODLE. **Moodle for teachers and trainers**. 2005.

MOODLE FOR MOBILES. **MoodleDocs**. Disponível em: http://docs.moodle.org/20/en/Moodle_for_Mobiles (acessado em 07/06/2011).

ROCHA, H. L. S. da. **Desenvolvendo Web Sites Interativos com JavaScript**. 1999.

ROCHA, H. L. S. da. **Introdução ao Javascript**. 1999.

SAP. *Standardized* **Technical Architecture Modeling: Conceptual and Design Level**. [S.l.]. 2007.

SUN MICROSYSTEMS. **The Java Language: An Overview**. Disponível em: <http://java.sun.com/docs/overviews/java/java-overview-1.html> (acessado em 07 de junho de 2011).

TECMUNDO. **Android: o sistema operacional móvel que conquistou o mundo**. Disponível em <http://www.tecmundo.com.br/9010-android-o-sistema-operacional-movel-que-conquistou-o-mundo.htm> (acessado em 06/07/2011).

THE OPEN UNIVERSITY. **A basic introduction to Moodle Architecture**. 2010.

W3C (World Wide Web Consortium). **CSS Introduction**. Disponível em: http://www.w3schools.com/css/css_intro.asp (acessado em 07 de junho de 2011).

W3C (World Wide Web Consortium). **Web Services Glossary** - W3C Working Group Note 11 February 2004. Disponível em: <http://www.w3.org/TR/2004/NOTE-ws-gloss-20040211/> (acessado em 07 de julho de 2011).

WAINS, S.; MAHMOODd, W.. **Integrating m-learning with e-learning**. SIGITE '08: Proceedings of the 9th ACM SIGITE conference on Information technology education. 2008.

Anexos

Anexo 1 – Archivo applicationParameters.xml – descrito no item 4.

```

1 <Config>
2   <table ID="userTable" NAME="mdl_user">
3     <field type="USERNAME" isKeyField="true">username</field>
4     <field type="PASSWORD" isKeyField="true">password</field>
5     <field type="FIRSTNAME">firstname</field>
6   </table>
7   <table ID="configTable" NAME="z_config">
8     <group NAME="identification">
9       <field type="USERNAME" isKeyField="true">user1</field>
10      <field type="COURSE" isKeyField="true" isCourseName="true">disci</field>
11      <field type="ACTIVE" isActiveFlag="true">activ</field>
12    </group>
13    <group NAME="Tareas">
14      <field type="TAREFAS" showMode="checkBox" text="Nueva tarefa se crea">tare1</field>
15      <field type="ALERT1" showMode="checkBox" text="Alertar para la entregas de tareas">aler1</field>
16      <field type="DAYS1" showMode="text" inputLength="1" text="Dias para alertar antes
17        de la expiracion del prazo de entrega de una actividad: ">dias1</field>
18    </group>
19    <group NAME="Chats">
20      <field type="ALERT2" showMode="checkBox" text="Alertar cuando un nuevo chat se inicia">aler2</field>
21      <field type="DAYS2" showMode="text" inputLength="1" text="Minutos de aviso antes
22        del inicio de un chat: ">dias2</field>
23    </group>
24    <group NAME="Foros">
25      <field type="FORUM" showMode="checkBox" text="Nuevo foro se crea">foru1</field>
26      <field type="ALERT3" showMode="checkBox" text="Alertar para participaciones en los foros">aler3</field>
27      <field type="DAYS3" showMode="text" inputLength="1" text="Dias para alertar antes
28        del final del periodo de participacion en un foro: ">dias3</field>
29    </group>
30  </table>
31  <getData>
32    <source ID="coursesForUser">
33      <query>SELECT * FROM mdl_course c, mdl_user u, mdl_role_assignments ra, mdl_context con,
34        mdl_role r WHERE u.ID = ra.userID AND ra.contextID = con.ID AND con.contextlevel = 50
35        AND con.instanceID = c.ID AND ra.roleID = r.ID AND r.shortNAME = 'student'
36        AND u.userName = ?</query>
37      <returnField type="ID">id</returnField>
38      <returnField type="NAME">fullname</returnField>
39    </source>

```

```

40 <source ID="assignmentsForUser">
41   <tableName>mdl_assignment</tableName>
42   <whereCondition>course=?</whereCondition>
43   <returnField type="ID">id</returnField>
44   <returnField type="TIMEAVAILABLE">timeavailable</returnField>
45   <returnField type="NAME">name</returnField>
46   <returnField type="COURSE">course</returnField>
47 </source>
48 <source ID="forumsForUser">
49   <tableName>mdl_forum</tableName>
50   <whereCondition>course=? AND type='general'</whereCondition>
51   <returnField type="ASSESSTIMEFINISH">assesstimefinish</returnField>
52   <returnField type="COURSE">course</returnField>
53   <returnField type="ID">id</returnField>
54   <returnField type="NAME">name</returnField>
55   <returnField type="TYPE">type</returnField>
56 </source>
57 <source ID="chatsForUser">
58   <tableName>mdl_chat</tableName>
59   <whereCondition>course=?</whereCondition>
60   <returnField type="ID">id</returnField>
61   <returnField type="CHATTIME">chattime</returnField>
62   <returnField type="NAME">name</returnField>
63   <returnField type="COURSE">course</returnField>
64 </source>
65 </getData>
66 <generalParameters>
67   <parameter NAME="securityPassKey">B#%8qk(yQ5=sPb_UlovNhST%030</parameter>
68 </generalParameters>
69 </Config>

```