



**TÉCNICAS DE APRIMORAMENTO DE ACESSO A DADOS
E AUTOMATIZAÇÃO DO ACOMPANHAMENTO DE
ALUNOS PARA O AMBIENTE VIRTUAL DE
APRENDIZAGEM MOODLE**

ELIAKIM ZACARIAS

**DISSERTAÇÃO DE MESTRADO EM ENGENHARIA ELÉTRICA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**TÉCNICAS DE APRIMORAMENTO DE ACESSO A DADOS
E AUTOMATIZAÇÃO DO ACOMPANHAMENTO DE
ALUNOS PARA O AMBIENTE VIRTUAL DE
APRENDIZAGEM MOODLE**

ELIAKIM ZACARIAS

**Orientador: PROF. DR.-ING. JOÃO PAULO CARVALHO LUSTOSA DA COSTA,
ENE/UNB**

DISSERTAÇÃO DE MESTRADO EM ENGENHARIA ELÉTRICA

**PUBLICAÇÃO PPGENE.DM - 664/2017
BRASÍLIA-DF, 31 DE MAIO DE 2017.**

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**TÉCNICAS DE APRIMORAMENTO DE ACESSO A DADOS
E AUTOMATIZAÇÃO DO ACOMPANHAMENTO DE
ALUNOS PARA O AMBIENTE VIRTUAL DE
APRENDIZAGEM MOODLE**

ELIAKIM ZACARIAS

DISSERTAÇÃO DE MESTRADO ACADÊMICO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM ENGENHARIA ELÉTRICA.

APROVADA POR:

Prof. Dr.-Ing. João Paulo Carvalho Lustosa da Costa, ENE/UnB
Orientador

Prof. Dr. Georges Daniel Amvame Nze, ENE/UnB
Examinador interno

Prof. Dr. Cristiano Bertolini, UFSM
Examinador externo

BRASÍLIA, 31 DE MAIO DE 2017.

FICHA CATALOGRÁFICA

ELIAKIM ZACARIAS

Técnicas de Aprimoramento de Acesso a Dados e Automatização do Acompanhamento de Alunos para o Ambiente Virtual de Aprendizagem Moodle

2017xv, 50p., 201x297 mm

(ENE/FT/UnB, Mestre, Engenharia Elétrica, 2017)

Dissertação de Mestrado - Universidade de Brasília

Faculdade de Tecnologia - Departamento de Engenharia Elétrica

REFERÊNCIA BIBLIOGRÁFICA

ELIAKIM ZACARIAS (2017) Técnicas de Aprimoramento de Acesso a Dados e Automatização do Acompanhamento de Alunos para o Ambiente Virtual de Aprendizagem Moodle. Dissertação de Mestrado em Engenharia Elétrica, Publicação 664/2017, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 50p.

CESSÃO DE DIREITOS

AUTOR: ELIAKIM ZACARIAS

TÍTULO: Técnicas de Aprimoramento de Acesso a Dados e Automatização do Acompanhamento de Alunos para o Ambiente Virtual de Aprendizagem Moodle.

GRAU: Mestre ANO: 2017

É concedida à Universidade de Brasília permissão para reproduzir cópias desta dissertação de Mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor se reserva a outros direitos de publicação e nenhuma parte desta dissertação de Mestrado pode ser reproduzida sem a autorização por escrito do autor.

ELIAKIM ZACARIAS

RUA TIRADENTES, 179, RIQUEZA, SC.

Agradecimentos

Agradeço a Deus, por todos os seus cuidados ao longo deste caminho.

Agradeço aos meus professores, sobretudo ao meu orientador, professor Dr.-Ing. João Paulo Carvalho Lustosa da Costa, por toda a sua paciência e dedicação em toda esta trajetória, auxiliando e oferecendo apoio mesmo nos momentos mais difíceis. Também ao professor Dr. Edison Pignaton de Freitas por todos os seus conselhos e cuidados. Aos coordenadores, aos bolsistas e participantes do projeto da UnB na Escola Nacional de Administração Pública (ENAP), bem como aos seus funcionários, por toda a sua atenção, pelo seu carinho e acolhimento no período em que estive com eles, além de todo o suporte e informações que permitiram este estudo.

Agradeço ao meu pai e à minha mãe por todo o apoio e auxílio que recebi desde o início e por todos os conselhos, dicas e ajuda. Ao meu irmão Iulisloi Zacarias, por ter corrido em meu socorro, e pelo trabalho firme e duro, e o importante auxílio prestado. Aos meus amigos e colegas que participaram desta história, motivando e participando ativamente, especialmente a Luciano Almeida e Juliano Pretz. Também à minha noiva, por todo o sustento emocional que me proveu, e pelo paciente acompanhamento de todo este capítulo de minha história. E a todos os que participaram desta história.

Eliakim Zacarias

TÉCNICAS DE APROIMORAMENTO DE ACESSO A DADOS E AUTOMATIZAÇÃO DO ACOMPANHAMENTO DE ALUNOS PARA O AMBIENTE VIRTUAL DE APRENDIZAGEM MOODLE

Autor: Eliakim Zacarias

Orientador: Prof. Dr.-Ing. João Paulo Carvalho Lustosa da Costa

Programa de Pós-graduação em Engenharia Elétrica

Brasília, 31 de maio de 2017

A evolução tecnológica apresentou novas possibilidades para o desenvolvimento de cursos à distância, contexto no qual a tecnologia da informação representa fator decisivo no avanço dessa forma de disseminação do conhecimento. Dentre as ferramentas de ensino à distância possibilitadas pela tecnologia da informação, o Moodle se destaca como uma ferramenta amplamente utilizada por diversas instituições públicas e privadas, no Brasil e no exterior. A Escola Nacional de Administração Pública (ENAP) é uma das instituições que além de usar tal ferramenta propõe novas soluções para serem a ela incorporadas. Um dos problemas identificados pela ENAP está relacionado à falta de acesso às informações sobre o perfil dos alunos, bem como do comportamento e desempenho de alunos em relação aos acessos e participação nos cursos à distância pela ferramenta Moodle, e também à dificuldade de manter o acompanhamento de tais estudantes. Neste contexto, esta dissertação propõe uma otimização ao acesso da base de dados dos alunos, automatização do contato com alunos e um relatório personalizado para acompanhando da participação dos mesmos. Para tal, soluções de automação de rotinas através de *plugins* foram desenvolvidas e incorporadas à ferramenta Moodle da ENAP com o objetivo de atender às suas demandas.

Palavras-chave: Educação à Distância, Otimização de acesso à base de dados, plugin Moodle, Automação de acompanhamento de alunos por mensagens.

ABSTRACT

TECHNIQUES TO ENHANCE DATA ACCESS AND AUTOMATION OF STUDENTS PARTICIPATION FOLLOW-UP TO THE MOODLE DISTANCE LEARNING TOOL

Author: Eliakim Zacarias

Supervisor: Prof. Dr.-Ing. João Paulo Carvalho Lustosa da Costa

Department of Electrical Engineering / University of Brasília

Graduate Program in Electrical Engineering

Brasilia, May 31, 2017

Technology advances present new possibilities for the development of distance learning courses, in which information technology represents a key factor in the progress of distance learning knowledge dissemination. Among the available tools based on information technology, Moodle is highlighted as a widely used platform by a number of public and private institutions in Brazil and worldwide. The National School for Public Administration, in Portuguese Escola Nacional de Administração Pública (ENAP), is one of these institutions, which besides of using the tool, it also proposes new solutions to be incorporated on Moodle. One of the problems identified by ENAP is related to the access of information of the students enrolled in its courses stored in the system. Another difficulty relates to access on activities of these students in the courses. In this context, this master thesis reports a work aiming the students' data base access optimization and the automation of the students' participation follow-up. These solutions are based on data mining and automation techniques by means of plugins. These plugins were developed and incorporated to the Moodle platform running at ENAP, aiming to address the presented demands.

SUMÁRIO

1	INTRODUÇÃO	1
1.1	OBJETIVOS.....	3
1.1.1	OBJETIVO GERAL.....	3
1.1.2	OBJETIVOS ESPECÍFICOS	3
1.2	ORGANIZAÇÃO DO TRABALHO	3
2	ESTADO DA ARTE DAS SOLUÇÕES DE EAD	5
3	REVISÃO DOS CONCEITOS E TECNOLOGIAS EMPREGADAS	9
3.1	ENSINO À DISTÂNCIA	9
3.2	AMBIENTES VIRTUAIS DE APRENDIZAGEM.....	11
3.3	TECNOLOGIAS WEB	12
3.3.1	PROTOCOLO DE TRANSFERÊNCIA DE HIPERTEXTO - HTTP	12
3.3.2	REQUISIÇÕES E MENSAGENS DO HTTP	13
3.3.3	LINGUAGEM DE SCRIPT <i>JavaScript</i>	13
4	SOLUÇÕES PROPOSTAS.....	15
4.1	OTIMIZANDO ACESSO AO REGISTRO DE ESTUDANTES NO BANCO DE DADOS NO MOODLE.....	15
4.2	MOTIVANDO A PARTICIPAÇÃO DE ESTUDANTES VIA <i>Plugin</i> DE MENSAGENS AUTOMÁTICAS PARA MOODLE	23
5	RESULTADOS.....	29
5.1	RESULTADOS OBTIDOS COM A OTIMIZAÇÃO DO ACESSO AOS REGISTROS NO BANCO DE DADOS	29
5.2	AVALIAÇÃO DO ENGAJAMENTO DOS ESTUDANTES PROPORCIONADO PELO <i>Plugin</i> DE MENSAGENS AUTOMÁTICAS	32
6	CONCLUSÕES E TRABALHOS FUTUROS	34
6.1	TRABALHOS FUTUROS	35
	REFERÊNCIAS BIBLIOGRÁFICAS.....	36
A	ARTIGOS PUBLICADOS	39

A.1	OPTIMIZING THE ACCESS RECORDS OF STUDENTS IN THE MOODLE VIRTUAL LEARNING ENVIRONMENT DATABASE	40
A.2	MOTIVATING ATTENDEE'S PARTICIPATION IN DISTANCE LEARNING VIA AN AUTOMATIC MESSAGING PLUGIN FOR THE MOODLE PLAT- FORM	45

LISTA DE FIGURAS

4.1	Caso de Uso da geração do relatório de desempenho e comportamento dos alunos de uma oferta de curso. Fonte: acervo pessoal.....	16
4.2	Diagrama de atividades do relatório de desempenho e comportamento dos alunos de uma oferta de curso. Fonte: ENAP.....	16
4.3	Acesso a funcionalidade Relatório de Comportamento e Desempenho dos Alunos a partir da interface de administração do ambiente. Fonte: ENAP.	17
4.4	Gráfico da participação dos alunos em cada um das atividades propostas. Fonte: ENAP.....	17
4.5	Gráfico da média da participação dos alunos em cada um das atividades propostas agrupado por tipo de atividade. Fonte: ENAP.	18
4.6	Gráfico da média da participação dos alunos em cada um das atividades propostas agrupado por tipo de atividade. Fonte: ENAP.	19
4.7	Gráfico do número de acessos por dia no decorrer do curso. Fonte: ENAP.	20
4.8	Tabela de participantes do curso por UF e botão para preparar relatório para impressão. Fonte: ENAP.....	20
4.9	Processo do navegador: diagrama do funcionamento do <i>plugin</i> no browser. Fonte: ZACARIAS et al. (2016).	22
4.10	Processo no servidor: diagrama do funcionamento do <i>plugin</i> no servidor. Fonte: ZACARIAS et al. (2016).....	22
4.11	Fluxo de operação do plug-in de envio de mensagem. Fonte: acervo pessoal... ..	24
4.12	Processos de interação do usuário com o <i>plugin</i> de mensagem automática. Fonte: acervo pessoal.....	25
4.13	Tela de seleção com as condições para enviar mensagens para os alunos. Fonte: ALMEIDA et al. (2016)	25
4.14	Tela de seleção com as condições para enviar mensagens para os alunos. Fonte: ALMEIDA et al. (2016)	27
4.15	Exemplo de mensagens recebidas pelo participante no final do curso. Fonte: ALMEIDA et al. (2016)	28
5.1	Comparação do tempo de execução das consultas. Fonte: acervo pessoal.	31

LISTA DE TABELAS

3.1	Principais diferenças entre o modelo presencial de ensino e o modelo EaD. Fonte: PEREIRA et al. (2017)	10
5.1	Comparação do tempo de consulta ao selecionar todos os cursos em uma única requisição. Fonte: ZACARIAS et al. (2016).....	29
5.2	Comparação do tempo de consulta selecionando colunas específicas. Fonte: ZACARIAS et al. (2016)	30
5.3	Comparação do tempo de consulta por usuário. Fonte: ZACARIAS et al. (2016)	32
5.4	Total e Porcentagem de acessos em dias em que não foram enviadas mensa- gens. Fonte: Enap 2015.	33
5.5	Total e Porcentagem de acessos em dias com mensagens automáticas são enviadas pelo plugin. Fonte: Enap 2015.....	33

LISTA DE TERMOS E SIGLAS

AVA	Ambiente Virtual de Aprendizagem
EaD	Educação à Distância
eLearning	Educação à Distância
ENAP	Escola Nacional de Administração Pública
HTTP	Protocolo de Transferência de Hipertexto
IoT	Internet das coisas
PHP	<i>Hypertext Preprocessor</i>
SGBD	Sistema Gerenciador de Banco de Dados
SMS	<i>Short Message Service</i>
SQL	<i>Structured Query Language</i>
TCP	<i>Transmission Control Protocol</i>
UF	Unidade da Federação
UnB	Universidade de Brasília

Capítulo 1

Introdução

O avanço da tecnologia registrado nas últimas décadas permite a fabricação de dispositivos como microcomputadores, *smartphones* e *tablets* cada vez mais compactos à preços acessíveis. Este avanço também é percebido quando se considera a facilidade das pessoas se conectarem à rede mundial de computadores, a Internet. O acesso à educação também foi facilitado por estes avanços, pois milhões de pessoas independente de sua localização estão sendo beneficiadas pelos recursos da Educação à Distância. A aprendizagem à distância quebra barreiras geográficas, permitindo a disseminação de conhecimentos necessários (DALMAU, 2014). Além disso, as ferramentas utilizadas inicialmente como recursos de educação à distância estão cada vez mais sendo utilizadas como ferramentas de suporte à educação presencial. Como citado por (MOORE; KEARSLEY, 2011), a educação à distância já é parte do cotidiano de muitas pessoas, seja para buscar uma graduação, ou até mesmo uma pós-graduação. Mais que na educação presencial, a dedicação dos alunos é um fator de grande importância para um bom desempenho (BEKELE, 2010).

Apesar do Moodle ser uma plataforma popular de Ambiente Virtual de Ensino e Aprendizagem (AVA), há ainda limitações essenciais, como a falta de ferramentas eficientes para acompanhamento dos estudantes. Porém, essas limitações podem ser superadas adicionando-se soluções de terceiros, incluindo módulos integrados ao software original (*plugins*) e sistemas que operam completamente separados do software original (DOUGIAMAS, 2001).

O acompanhamento de estudantes é crucial para gerar relatórios de monitoramento do comportamento dos estudantes nos cursos oferecidos. Apesar de toda a tecnologia disponível, como por exemplo o ambiente Moodle, que possui grande aceitação por instituições de ensino, há espaço para aprimoramento das ferramentas disponíveis, de forma a auxiliar os tutores das escolas de modo efetivo. Também são necessários aprimoramentos para motivar e alavancar a participação dos alunos nos cursos, conseqüentemente reduzindo as taxas de evasão dos cursos na modalidade de EaD. Um exemplo de funcionalidade com este propósito é o registro de atividades do usuário presente no Moodle, que pode ser usado para controlar e capturar informações de acompanhamento relevantes quanto a assiduidade e acesso à ma-

teriais, por parte dos alunos, nos cursos oferecidos. Contudo, ainda faltam ferramentas que auxiliam os tutores a identificar as causas das evasões dos alunos, baixo rendimento e falta de comprometimento com os cursos. Tais casos podem beneficiar-se dos *plugins* propostos. Além disso não há ferramentas disponíveis que facilitam a comunicação automatizada com os alunos por meio de mensagens integradas ao ambiente Moodle em forma de *plugins*.

Na Escola Nacional de Administração Pública (Enap) o AVA Moodle é utilizado para prover cursos de capacitação à distância para funcionários públicos de diversos órgãos, como tribunais e universidades. Com o intuito de auxiliar os estudantes, acompanhar e analisar seu desempenho a ENAP utilizou uma solução interna de relatórios. O procedimento para a geração destes relatórios consistia em percorrer todo o registro de atividades dos alunos. Os dados eram filtrados pela identificação do curso, data de início e data de encerramento. Esta solução apresentava desempenho aceitável enquanto era aplicada em bases de dados menores, porém tornou-se inviável a medida que os cursos ofertados pela ENAP tornaram-se mais populares e passaram a atender um grande número de estudantes. Isso se deve ao fato de que a quantidade de consultas ao Sistema Gerenciador de Banco de Dados requeridas é proporcional à quantidade de estudantes que precisam ser verificados.

O objetivo deste estudo é apresentar a otimização realizada no banco de dados da ENAP, um modelo de relatório de acompanhamento dos alunos, além de um sistema de envio de mensagens automáticas, durante a cooperação técnica entre a ENAP e a Universidade de Brasília (UnB). Este trabalho contribui apresentando otimizações de acesso aos dados armazenados pelo Moodle e a comparação entre a solução atual e a proposta. Também apresenta um sistema de envio de mensagens automáticas e seu impacto sobre o acesso dos alunos no sistema, além de um relatório com dados sobre o acesso dos alunos ao sistema da escola virtual. As soluções propostas, implementadas, visam reduzir aumentar a participação dos alunos nos cursos EaD providos pela ENAP. As melhorias aplicadas compreendem a otimização e portanto redução do tempo necessário para geração dos relatórios por meio da manipulação mais eficiente dos dados gerados no AVA Moodle e um mecanismo que facilita a comunicação com alunos. A otimização de acesso se deu por normalização e separação dos dados dos registros de acesso, possibilitando que as consultas aos relatórios de desempenho e participação dos alunos fosse processados em um intervalo de tempo muito menor. Adicionalmente uma amostragem e análise das evasões de cursos por parte dos alunos é realizada, demonstrando resultados por meio dos relatórios e manipulação da base de dados do sistema da escola virtual Moodle.

1.1 Objetivos

1.1.1 Objetivo Geral

O objetivo deste estudo é apresentar algumas contribuições desenvolvidas durante a cooperação técnica entre a Escola Nacional de Administração Pública (ENAP) e a Universidade de Brasília (UnB), bem como melhorar a comunicação com os alunos por meio de mensagens automáticas de acompanhamento, e demonstrar os ganhos de desempenho alcançados na geração de relatórios do ambiente virtual de ensino. Tais relatórios são utilizados especialmente na análise e controle da participação dos alunos.

1.1.2 Objetivos Específicos

Os seguintes objetivos específicos podem ser mencionados:

- Desenvolvimento de soluções capazes de otimizar a recuperação de dados sobre o tempo de acesso de alunos em cursos realizados com o apoio de ambientes virtuais de aprendizagem.
- Desenvolvimento de solução de automatização de interação entre o sistema de ensino à distância e os alunos por meio de envio de mensagens automáticas.
- Apresentação de ferramentas desenvolvidas durante a cooperação técnica entre a UnB e a ENAP.
- Apresentação das melhorias de desempenho do sistemas obtidas por meio de normalização dos dados do ambiente virtual e implementados no ambiente virtual de ensino utilizando-se de tecnologias *web* conhecidas.
- Demonstrar os resultados obtidos por meio dos relatórios e manipulação da base de dados do ambiente virtual Moodle.

1.2 Organização do Trabalho

Seguindo esta introdução e definição dos objetivos gerais e específicos, os seguintes capítulos têm o objetivo de oferecer uma breve revisão da literatura sobre as tecnologias empregadas no desenvolvimento deste trabalho. Também oferecerão uma descrição das metodologias utilizadas e materiais de apoio, além de resultados alcançados com as técnicas aplicadas. Por fim, oferece-se a conclusão do trabalho acompanhado de sugestões para trabalhos futuros. Para uma melhor estruturação e entendimento do trabalho, o texto está dividido em capítulos na seguinte ordem.

O Capítulo 2 apresenta e discute o estado da arte relacionando as principais técnicas e ferramentas usadas em soluções de EaD. Esta revisão aborda tanto os aspectos técnicos/tecnológicos quanto sistêmicos das soluções, buscando relacioná-los às propostas apresentadas neste trabalho.

O Capítulo 3 apresenta um breve histórico da modalidade de ensino à distância, bem como o cenário do ensino atualmente. Também são apresentados os Ambientes Virtuais de Aprendizagem, uma introdução ao funcionamento de ferramentas e tecnologias que permitem a utilização destes ambientes utilizando a Internet.

No Capítulo 4 são apresentados as propostas das melhorias sugeridas no ambiente virtual de aprendizagem com objetivo de otimizar o acesso aos dados. Esta otimização por meio de normalização permite a geração dos relatórios e auxílio aos administradores para gerenciar a interação com os usuários do sistema por meio da automação do envio de mensagens aos participantes.

O Capítulo 5 apresenta os resultados alcançados com as propostas. Este capítulo destaca os benefícios advindos da adoção das soluções propostas, por meio da coleta de dados de utilização do sistema EaD Moodle administrado pela ENAP.

O Capítulo 6 apresenta a conclusão do trabalho e trabalhos futuros. Nele é apresentada uma reflexão sobre as necessidades apresentadas como demanda para o trabalho realizado e o que se conseguiu alcançar ao final do mesmo. Discute-se ainda as possibilidades de extensões futuras baseadas nas propostas apresentadas que podem complementar e aprimorar os resultados já alcançados.

Capítulo 2

Estado da arte das soluções de EaD

Moodle (DOUGIAMAS, 2001) é uma das principais plataformas grátis e de código-aberto para aprendizagem à distância, incluindo a participação e colaboração de sua comunidade em seu desenvolvimento. A equipe responsável pelo desenvolvimento da ferramenta provê suporte, correções de falhas e, adicionalmente, inclusão de novas funcionalidades à plataforma. A integração entre os módulos relacionados aos alunos, tutores e professores do sistema é um ponto positivo, pois facilita a utilização do AVA Moodle por parte dos mesmos. Adicionalmente às funcionalidades providas pela estrutura principal do ambiente, novas funcionalidades podem ser incluídas ao ambiente com a utilização de *plugins* ou extensões. Estes pequenos sistemas que proveem funcionalidades adicionais, específicas ou especializadas são chamados *plugins*.

Em NAGI; SUESAWALUK (2008) é apresentada uma análise baseada em relatórios do Moodle para medir o nível de interatividade no ambiente virtual de cursos de ensino na Assumption University, Tailândia. O objetivo proposto pelos autores é analisar o comportamento e interação dos estudantes no ambiente Moodle em quatro cursos de pós-graduação, todos eles em nível de Mestrado e oferecidos pela universidade na modalidade *eLearning*, ou educação à distância. Essas peças de informações são coletadas em tempo real de um conjunto intencionalmente selecionados de funcionalidades (páginas) e relatórios do Moodle, incluindo informação que não está registrada no banco de dados, como exibições de telas e cliques. Uma das conclusões em (NAGI; SUESAWALUK, 2008) é a viabilidade de implementação de recursos que permitam melhorar a interação entre estudantes e instrutores dos cursos, bem como um ranking de cursos com mais interações e colaboração. Este ranking permite comparar a interação dos alunos em diferentes cursos e facilita a identificação dos cursos com baixo nível de interação possibilitando que medidas sejam aplicadas com o objetivo de aumentar a participação nos cursos por parte dos estudantes.

Ainda, NAGI; SUESAWALUK (2008) conclui que um exame cuidadoso dos relatórios Moodle pode estimular uma melhora na plataforma de comunicação entre tutores e alunos. Desta forma, o presente trabalho contribui com tal comunicação ao apresentar a ferramenta

de mensagens automáticas. A ferramenta proposta analisa automaticamente a situação em que o aluno se encontra em relação ao acesso, nota e conclusão de atividades e envia mensagens específicas de acordo com cada situação.

Bovo *et al* (BOVO et al., 2013) propõem um técnica de análise do registro de ocorrências *logs* do AVA Moodle. Com técnicas de mineração de dados, aprendizado de máquina e inteligência artificial os autores são capazes de indicar quais grupos são os mais relevantes para formar ferramentas de suporte de gestores e professores. Tal técnica de mineração de dados pode também prever o comportamento dos estudantes e seu desempenho nos cursos em andamento. Isto pode também permitir que professores e mentores sugiram materiais para leitura, exercícios de fixação e sugiram ações que possibilitam os estudantes atingir um maior desempenho no curso, que geralmente é quantificado por meio de notas.

Em HIJÓN-NEIRA; VELÁZQUEZ-ITURBIDE (2008), tutores e professores precisavam de informações completas sobre o comportamento de estudantes, para que fosse possível analisar e propor ações para melhorar o rendimento dos estudantes. Para atender a esta demanda, desenvolveu-se o sistema *Merlin*, composto de três camadas. A primeira camada contém dados do sistema de educação. A segunda camada consiste de ferramentas que analisam e formam um banco de dados com registros do sistema, *cookies*, *logins* no sistema e questionários, com o objetivo de registrar as ações dos estudantes durante atividades no sistema virtual. A terceira camada é de apresentação de relatórios, baseada nos dados obtidos por ferramentas de mineração.

A necessidade de implementar meios adicionais de comunicação e interação entre tutores, professores, e alunos é abordada por Céspedes e Rivas em (CÉSPEDES; RIVAS, 2012). Com este propósito o Centro Tecnológico de Costa Rica desenvolveu sua própria plataforma com o objetivo de integrar o ambiente EaD com o serviço de mensagens de texto para celulares, também conhecidas como *Short Message Service* (SMS). A integração destes serviços também possibilita que os usuários executem comandos remotos na plataforma Moodle, utilizando a comunicação por SMS.

O sistema de envio de SMS proposto em CÉSPEDES; RIVAS (2012) pode ser beneficiado pelo *plugin* de mensagens automáticas proposto no presente trabalho. A proposta de integração entre as ferramentas busca fornecer ao aluno informações sobre o seu desempenho no curso (*feedback*). As informações enviadas ao aluno buscam incentivá-lo, ou mesmo lembrá-lo, de acessar o sistema e a continuar seus estudos, realizando atividades pendentes, por exemplo.

Com o objetivo de tornar as aulas na modalidade EaD mais atrativas, o trabalho (WANG; WANG; YEN, 2005) propõem um sistema de comunicação entre alunos e professores em forma de bate-papo. A ferramenta proporciona aos alunos a possibilidade de discutir os conteúdos das aulas além de possibilitar a resolução de dúvidas com os professores e tutores dos cursos. O sistemas conta com recursos que permitem o envio de mensagem offline, ou seja, os alunos não precisam, necessariamente estar online todo o tempo para utilizar a ferra-

menta. Além disso, conta com um sistema de recomendação de conteúdos para o usuário, a fim de sugerir conteúdos mais adequados ao desempenho do aluno. No trabalho de WANG; WANG; YEN (2005) as mensagens não ocorrem de maneira automática, uma vez que os estudantes ou professores devem tomar a iniciativa de iniciar a comunicação utilizando a ferramenta proposta, enquanto que o *plugin* de mensagens proposto no presente trabalho realiza esta ação (envio de mensagens) de forma automatizada. Já no artigo de IJTIHADIE et al. (2011), propõe-se uma arquitetura de entrega de mensagens para sistemas de aprendizagem utilizando sistemas gerenciadores de e-mails, e sua arquitetura para regiões com redes de comunicações mais precárias.

A educação à distância é uma ótima opção para superar limitações geográficas. No entanto, um dos principais desafios que a modalidade EaD sofre é provar que a qualidade dos ensino à distância oferece a mesma qualidade que a modalidade de ensino convencional (presencial). Uma das críticas frequentemente feita ao ensino EaD é a interação limitada entre alunos e professores. Além disso muitas plataformas de ensino na modalidade EaD limitam-se a um ambiente virtual para gerenciamento de material e a disponibilização de material digitalizados aos alunos. Outra causa das falhas no ensino na modalidade EaD elencada no trabalho de FERDIANA (2013) é a falta de um modelo de execução para EaD. Ferdiana (FERDIANA, 2013) argumenta que um uma política de implementação de EaD pode superar estes desafios e propõem um modelo de implementação para EaD denominada *Distance Learning Lifecycle Management (DLLM)*. O modelo se baseia em um modelo ágil de aprendizado e é dividido em quatro fases: exploração, planejamento, iteração e produção. Os resultados são apresentados na forma de três estudos de casos.

O trabalho proposto em BABIĆ; KRESIĆ; KUCEL (2014) tem como principal foco os participantes com função ativa na criação e compartilhamento de conteúdo, como os tutores e professores. A ferramenta proposta foi avaliada por meio de exemplos de troca de mensagens entre professores e estudantes, integrada ao AVA Moodle versão 2.0, em conjunto com recursos de armazenamento em nuvem (Google Drive). A avaliação da ferramenta levou em consideração as opiniões dos estudantes e professores que foi obtida por meio de questionários.

Com base nos trabalhos analisados, nota-se uma carência de ferramentas que integram a análise dos dados fornecidos pelos AVAs com ferramentas de comunicação automatizadas para os alunos. Assim, este trabalho procura contribuir com a aplicação de EaD na ENAP preenchendo esta lacuna ao passo que a ferramenta proposta busca automatizar a comunicação com os alunos através de mensagens de e-mail. As decisões de quando enviar as mensagens, qual o conteúdo da mensagem, e quais alunos irão recebê-las são tomadas com base na análise dos dados de acesso e desempenho dos alunos coletados pelo próprio AVA. Estas informações ficam disponíveis para análise e observação, agrupadas no relatório que exibe o comportamento e o desempenho dos alunos nos cursos. Para tornar possível a implementação de tais *plugins*, foram necessários implantar modificações no AVA que possibilitem a recuperação de dados sobre o tempo de acesso do aluno, resultando em um *plugin* para a

medição do tempo de acesso do aluno com base no tempo em que este manteve o sistema aberto em seu navegador. Desta forma, este trabalho visa contribuir com o cenário de EaD da ENAP e com ambientes EaD em geral através da implementação de *plugins* reaproveitáveis que estendem as funcionalidades oferecidas no ambiente Moodle.

Capítulo 3

Revisão dos conceitos e tecnologias empregadas

Este capítulo apresenta uma breve revisão dos conceitos relacionados ao ensino à distância e as tecnologias empregadas na modalidade de ensino à distância bem como as tecnologias empregadas no desenvolvimento deste trabalho.

3.1 Ensino à Distância

A educação à distância não é nenhuma novidade sendo que a prática já é consolidada há mais de um século em diversas regiões do mundo ocidental. Algumas universidades surgiram ainda na década de 1960 na Alemanha, Grã-Bretanha, Estados Unidos, Espanha entre outros países com o objetivo de trabalhar unicamente com ensino superior na modalidade à distância (PEREIRA et al., 2017). No Brasil há registros da prática desta modalidade de ensino que datam das décadas de 1920 e 1930. Os cursos oferecidos inicialmente eram focados em educação supletiva ou formação profissional de nível básico, como nos casos do Instituto Monitor e Instituto Universal brasileiro. No Brasil a educação formal na modalidade EaD sofreu certa resistência e se consolidou efetivamente a partir da década de 1990 quando a nova lei de Diretrizes e Bases da Educação Nacional foi aprovada (FRANCO et al., 2006).

A cada dia mais organizações adotam a modalidade de ensino, sejam elas instituições de ensino ou empresas que pretendem capacitar seus funcionários por meio de cursos de curta duração. Instituições públicas e privadas têm demonstrado uma grade comprometimento com a modalidade e o crescimento em número de matrículas é de cerca de 18% ao ano (PEREIRA et al., 2017) e em 2014 o número de matrículas só no ensino superior foi de 190 mil (BRASIL, 2016).

Um das vantagens dos cursos oferecidos à distância é a possibilidade do aluno escolher a hora e o local que lhe é favorável e por isso facilita o acesso a pessoas que não poderiam, por exemplo, estudar na modalidade de ensino presencial, com horas e locais bem defini-

dos (PEREIRA et al., 2017). Além disso a disponibilidade de acesso à Internet está em constante crescimento, atingindo locais remotos ou de difícil acesso, facilitando o alcance da EaD em áreas que exigiriam um deslocamento físico inviável até as instituições de ensino. Analisando a Tabela 3.1 onde (PEREIRA et al., 2017) resume as principais diferenças entre o ensino presencial e a EaD de uma forma clara e objetiva, podemos perceber que a EaD proporciona maior liberdade ao aluno quanto à local e tempo que devem ser dedicados ao estudo. No entanto, a modalidade EaD exige muito mais disciplina e organização por parte do aluno que deve buscar o conhecimento de forma autônoma.

Tabela 3.1: Principais diferenças entre o modelo presencial de ensino e o modelo EaD.
Fonte: PEREIRA et al. (2017)

Presencial	EaD
Centro Geográfico de ensino: sala de aula	O aluno estuda onde e quando desejar
Ênfase na interação social presencial	Ênfase na interação social ocorrida em AVAs
Ênfase na interação	Ênfase na mediação
Situações de ensino controladas pelo professor	Aprendizagem independente e autônoma
Um só tipo de docente	Conteudista, tutores (presenciais e/ou virtuais), professores (formadores)
Maior possibilidade de o professor ser percebido como “fonte do conhecimento”	Mediação realizada, essencialmente por tutores
Recursos didáticos usuais	Recursos didáticos baseados nas TICs
Comunicação direta	Comunicação diferenciada no espaço e no tempo
Foco no atendimento direto ao aluno	Foco no auxílio para que o estudante se organize e busque o conhecimento em locais e horários definidos por ele próprio

Pesquisadores da área de educação definem o EaD como uma forma de ensino guiada pelo autoaprendizado. Os recursos que são utilizados no processo são organizados de forma sistemática pelos professores ou tutores e podem ser utilizados tanto de forma isolada e autocontidos como de forma combinada com outros recursos (MEDEIROS; FARIA, 2003; PEREIRA et al., 2017). Em sua forma mais comum nos dias atuais esses recursos são disponibilizados em ambientes específicos utilizando a Internet. Contudo, o EaD pode também utilizar outros meios e mídias para disponibilizar o conteúdo, como correspondências físicas.

As soluções propostas neste trabalho visam adereçar a questão da comunicação com os alunos, permitindo que mensagens pré-programadas sejam enviadas a grupos de alunos conforme suas situações, de forma a garantir o conhecimento de seu estado em relação ao curso. Desta forma, busca-se auxiliar principalmente na questão da motivação à participação e acesso dos alunos ao sistema, apesar da comunicação diferenciada quanto ao formato presencial. O sistema também permite que os tutores possam comparar o comportamento dos alunos em turmas, inclusive comparando entre as turmas. Isto permitiria, por exemplo, testar uma nova forma de motivar os alunos e comparar se os resultados esperados foram alcança-

dos. Tal acompanhamento foi feito para o sistema de envio de mensagens automáticas, onde percebeu-se um aumento dos acessos gerais no curso nos dias em que as mensagens foram enviadas.

3.2 Ambientes virtuais de aprendizagem

Com a criação de uma interface mais amigável para a *web* por meio de elementos gráficos que podem receber interações do público em geral, a *web* tornou-se muito mais popular e diversos serviços surgiram neste ambiente virtual (PEREIRA et al., 2017). Os processos de ensino também passaram a fazer parte deste ambiente, alcançando todas as pessoas que tem acesso à rede mundial de computadores, a Internet.

Para que este “serviço” passasse a ser disponibilizado de forma virtual, utilizando a Internet, os processos de ensino e aprendizagem precisaram se adaptar e novas ferramentas precisaram ser criadas para possibilitar que os processos ocorressem independente da proximidade física entre alunos e professores. Como tal interação ainda parte do interesse do aluno, ferramentas que o incentivem a tal busca são necessárias. Uma ferramenta bem conhecida empregada no EaD são os Ambientes Virtuais de Aprendizagem (AVA) que conforme MEHLECKE; TAROUCO (2003) também podem ser chamados “salas de aula virtuais”. A diversidade de ferramentas e recursos nestes ambientes virtuais é altamente desejável, pois os estudantes não constroem seus conhecimentos sozinhos, mas sim interagindo com colegas, professores e tutores do curso (PEREIRA et al., 2017).

Os AVAs são sistemas orientados ao gerenciamento da aprendizagem utilizando a internet como meio. Estes sistemas pode ser utilizados como ferramenta principal da estratégia pedagógica, por exemplo em cursos totalmente à distância, ou ainda como ferramenta de apoio à educação, servindo neste caso para organização de exercícios e materiais bem como para ofertar materiais adicionais aos alunos (JUNIOR, 2008). Uma área muito explorada nos AVAs é a interação como meio de estimular a aprendizagem e por isso normalmente estes software oferecem vários recursos que estimulam a interatividade focando no conteúdo apresentado. Alguns exemplos de recursos que estimulam a interatividade entre os usuários do AVA são: fóruns de discussões e debates sobre os temas, ferramentas de bate-papo entre os alunos, ferramentas para criação de artigos colaborativos como as wiki, dicionários colaborativos entre outras. Além das ferramentas citadas, um recurso que está ganhando grande destaque são os recursos de áudio e vídeo. Estes recursos podem estar presentes em forma de audioconferências e videoconferências com comunicação bidirecional entre os participantes (alunos, tutores e professores) (PEREIRA et al., 2017) ou ainda em forma de videostreaming, onde o aluno pode assistir o material audiovisual na hora que desejar e quantas vezes precisar.

Uma outra característica importante dos AVAs é relacionada a organização e disponibilização dos recursos. Estes ambientes possuem papéis, ou perfis de acesso de acordo com a função do usuário (PEREIRA et al., 2017). Entre os principais perfis de acesso presentes em praticamente todos os AVAs estão o perfil de aluno, que têm acesso ao material do curso e participam de forma online das atividades desenvolvidas no curso; e o perfil de professores que acompanham os alunos nas atividades desenvolvidas, gerenciam e alimentam o ambiente com materiais educacionais, propõem exercícios de fixação, avaliam os alunos, registram notas. Um papel adicional de administrador também está presente em alguns AVAs e possibilitam aos usuários com este papel realizar atividades administrativas no curso, por exemplo realizar e atualizar cadastro dos alunos, atribuir cursos aos alunos, autorizar participação em cursos e gerar relatórios.

3.3 Tecnologias WEB

Esta seção apresenta as tecnologias que foram empregadas no desenvolvimento das soluções adicionadas ao AVA Moodle utilizado pela ENAP. As aplicações *web* modernas, mesmo com o avanço da chamada Web 2.0, utilizam estas tecnologias para proporcionar uma melhor experiência aos usuários, porém não se limitam às tecnologias citadas.

3.3.1 Protocolo de Transferência de Hipertexto - HTTP

O Protocolo de Transferência de Hipertexto (HTTP) proposto pela RFC1945 (BERNERS-LEE; FIELDING; FRYSTYK, 1996) e posteriormente atualizado pela RFC2616 (FIELDING et al., 1999), é sem dúvida o mais popular protocolo de aplicação na Internet atualmente. Inicialmente proposto com uma simples forma de recuperar documentos hipertexto de servidores com comandos de uma linha, o HTTP fomentou um crescimento da Internet e hoje está presente em inúmeros dispositivos de diferentes formas, tamanhos e recursos. Seja um microcomputador com recursos de hardware mais potentes, dispositivos como *smartphones*, ou até mesmo pequenos dispositivos sensores, grande parte destes implementam o protocolo HTTP, tanto para transferir vídeos, imagens, páginas da internet ou para acessar *webservices* em aplicações voltadas à Internet das Coisas (IoT). A cada dia que passa, mais serviços que fazem parte do nosso dia-a-dia migram para a *Web*, como interação social e outras aplicações que fazem parte da nossa vida pessoal ou profissional, tornando o protocolo HTTP parte de muitas aplicações imagináveis (GRIGORIK, 2013).

Os ambientes de ensino fazem parte dos serviços que passaram a ser disponibilizados online. Estas plataformas têm se tornado populares pois permitem superar problemas relacionados a geolocalização, disponibilidade de horários, ou mesmo o conforto de cada aluno de prosseguir no curso respeitando seu ritmo. São abundantes as ferramentas propostas para

gerenciar e disponibilizar conteúdo para cursos EaD. Alguns exemplos são os softwares Blackboard Learn, Brightspace, Talent LMS, Sakai e Moodle. Todas as soluções citadas, incluindo o Moodle utilizado neste estudo, consistem em softwares que são executados em um servidor remoto e que são acessados pelos usuários utilizando um navegador *web*, cuja conexão se dá pelo protocolo HTTP para a transferência dos dados entre o servidor e o cliente (PIÑA, 2012).

3.3.2 Requisições e mensagens do HTTP

O protocolo HTTP define como as páginas são requisitadas e como o servidor web responde essas requisições transferindo os dados do recurso solicitado, neste caso as páginas do AVA utilizado. O protocolo HTTP é implementado sobre o protocolo de transporte *Transmission Control Protocol* (TCP), e as conexões cliente-servidor para requisições HTTP podem ser persistentes ou não-persistentes (KUROSE; ROSS, 2013). Atualmente os navegadores modernos, implementando HTTP versão 1.1 permitem até seis conexões simultâneas (PIÑA, 2012) que são reutilizadas para várias requisições, portanto utilizando conexões persistentes. É importante notar que, até mesmo requisições que são originadas após a descarga inicial da página, ou seja após a página ser exibida no software cliente (por exemplo, requisições originadas por Javascript), podem reutilizar conexões TCP existentes. Esse comportamento auxilia a diminuir o tempo de espera entre a ação do usuário e um resposta da página, pois pula-se a etapa de estabelecimento da conexão TCP entre o cliente e o servidor.

Uma característica notável do Protocolo HTTP é a simplicidade do formato da mensagem. As mensagens de requisição consistem em uma primeira linha, chamada linha de requisição, seguida de linhas com informações denominadas “cabeçalhos”. A linha da requisição contém além da versão do HTTP a ser utilizado, a URL e o método da requisição. As linhas de cabeçalho contém informações como o endereço do *host* de destino, o tipo da conexão que o cliente deseja usar (persistente ou não-persistente), a linguagem preferida para a página de resposta que será devolvida pelo servidor (Português, Alemão...) e o *User-agent* que identifica o software que está sendo utilizado pelo cliente (*browser*). Este último campo tem grande importância atualmente, pois no caso dos AVA como o Moodle, o servidor pode selecionar os recursos mais apropriados para a resposta que ele irá devolver. Isto é, se o usuário final está acessando o AVA por meio de um dispositivo móvel como um *smartphone* o servidor pode devolver uma página adaptada para telas menores.

3.3.3 Linguagem de Script *JavaScript*

O *JavaScript* é uma linguagem de programação que permite adicionar funcionalidades e capacidades extras às páginas *Web*. É uma linguagem que utiliza poucos recursos e é interpretada pelo navegador *Web* no computador do cliente que está acessando a página. Isto é, os comandos da linguagem são executados no computador do usuário e não no servidor, por-

tanto é caracterizada como uma linguagem *Client Side* (FOUNDATION, 2017). Atualmente a linguagem conquistou grande visibilidade e ganhou implementações para rodar em ambientes fora do navegador de internet, por exemplo em servidores. Nestes casos a linguagem passa a ser caracterizada como *Server Side* (FLANAGAN, 2011).

O *JavaScript* foi inicialmente desenvolvido por Brendan Eich na Netscape e foi criada para ser executada no navegador desenvolvido pela empresa (FOUNDATION, 2017). A característica de ser executada pelo navegador faz com que a linguagem seja multiplataforma, pois basta que o computador do usuário tenha um navegador de internet compatível com *JavaScript* instalado para que os scripts (programas) da linguagem sejam executados. Ao longo do tempo a linguagem ganhou melhorias e atualmente é uma linguagem multi-paradigma suportando programação orientada à objetos, imperativa e funcional.

Considerando-se a parte *Client Side* da linguagem, esta possibilita controlar o navegador *Web* ou manipular objetos de uma página *Web* (FOUNDATION, 2017). Por exemplo, utilizando *JavaScript* é possível verificar se a informação que o usuário digitou em um campo de preenchimento de um formulário está correta, antes mesmo de ser enviada para o servidor. Possibilita ainda diversas outras ações, como exibir mensagens e tornar as páginas que o usuário visita mais dinâmicas, modificando elementos da página e responder a interações do usuário, por exemplo cliques do mouse e navegação pela página (SEBESTA, 2014).

No contexto deste trabalho a tecnologia *JavaScript* é utilizada para enviar informações do navegador do usuário para o servidor onde o AVA está sendo executado sem que seja necessário uma ação do usuário. Isto é, ao acessar o ambiente virtual, as páginas (telas) do ambiente são exibidas e passam a enviar informações sobre o acesso do usuário sem que este realize qualquer ação específica. Esta funcionalidade é utilizada para auxiliar na normalização do acesso aos dados, alimentando o banco de dados com informações sobre o acesso dos usuários.

Capítulo 4

Soluções propostas

Este capítulo apresenta as propostas e as implementações das melhorias sugeridas no ambiente virtual de aprendizagem. Na primeira parte são apresentadas as modificações realizadas nos sistemas de relatórios do AVA Moodle utilizado pela ENAP com objetivo de otimizar o acesso aos dados para geração dos relatórios e auxílio aos administradores para gerenciar a interação com os usuários. Na segunda parte é apresentada a funcionalidade adicionada no ambiente que implementa a automação do envio de mensagens aos participantes.

4.1 Otimizando Acesso ao Registro de Estudantes no Banco de Dados no Moodle

O processo de acompanhamento da aprendizagem e participação dos alunos é uma importante funcionalidade do AVA Moodle. Os dados que o ambiente coleta durante o acesso do aluno são armazenados e são utilizados na geração de relatórios que podem ajudar os professores e tutores a tomar decisões quanto à metodologia que está sendo empregada, disponibilização de material, entre outros aspectos do processo pedagógico. Estas informações também podem ser utilizadas como subsídio, caso novas ferramentas que operam de forma automatizada no ambiente sejam criadas.

Inicialmente, um relatório foi proposto para permitir que gestores acadêmicos da ENAP coletem informações sobre os cursos e o desempenho dos alunos participantes dos cursos. O relatório foi integrado à área de administração já existente do Moodle seguindo o caso de uso ilustrado na Figura 4.1.

Como pode ser observado na figura, são incluídos os Casos de Uso responsáveis pela autenticação dos usuários e controle de permissões de usuário no ambiente. Como o relatório encontra-se na área de administração do ambiente, uma pré-condição do Caso de uso é o ator, neste caso o “Gestor Acadêmico”, estar devidamente autenticado e habilitado a acessar

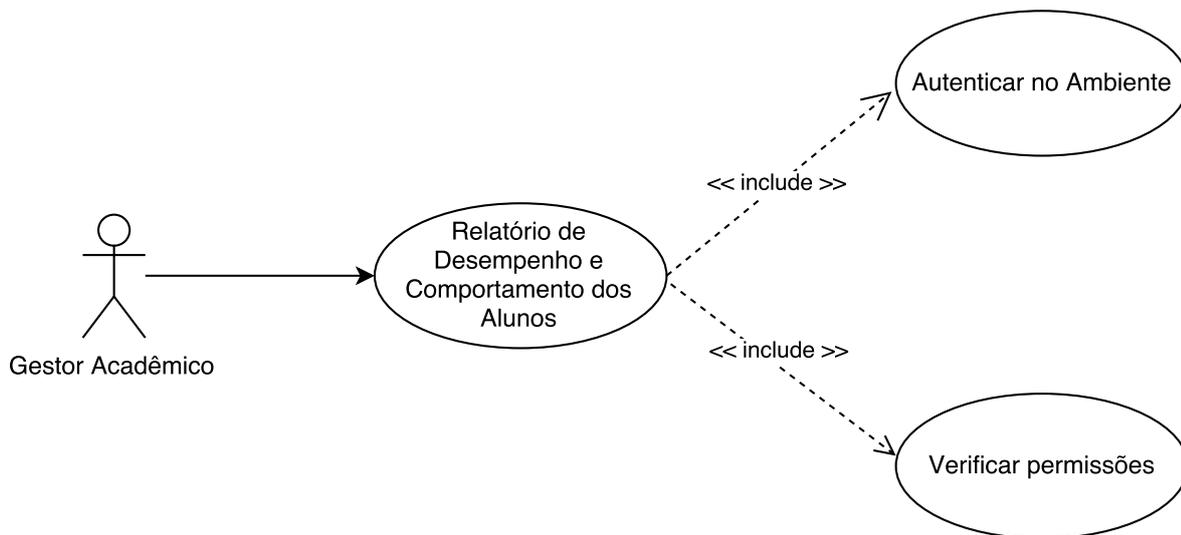


Figura 4.1: Caso de Uso da geração do relatório de desempenho e comportamento dos alunos de uma oferta de curso. Fonte: acervo pessoal.

esta funcionalidade e, portanto, os casos de uso já existentes no ambiente responsáveis pela autenticação de usuário são incluídos. O diagrama de atividades correspondente ao relatório está ilustrado na Figura 4.2.

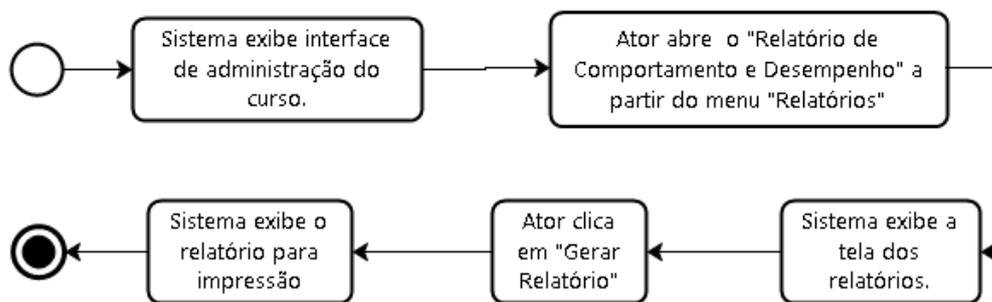


Figura 4.2: Diagrama de atividades do relatório de desempenho e comportamento dos alunos de uma oferta de curso. Fonte: ENAP.

Na figura 4.2, o caso de uso inicia com o ator (Gestor Acadêmico) autenticando-se no ambiente e navegando para a área de administração de cursos do AVA. Em seguida o ator aciona a função “Relatório de Comportamento e Desempenho dos Alunos” que está disposta sob o menu “Relatórios” identificado pela Seta 1 na Figura 4.3.

Ao acionar a opção “Relatório de Comportamento e Desempenho dos Alunos” uma nova página é carregada e “Resumo dos participantes” do curso é apresentado na tela antes da emissão do relatório como indicado pela Seta 2 na Figura 4.3. Os campos incluídos no “Resumo dos participantes” são “Participantes matriculados”, “Inscrições canceladas”, “Alunos Aprovados”, “Alunos Reprovados”, “Alunos Desistentes” e “Índice de conclusão”. Alunos aprovados são aqueles que fizeram atividades e atingiram a nota para a aprovação, os alunos reprovados fizeram ao menos uma atividade, mas não atingiram a nota exigida. Já os alunos

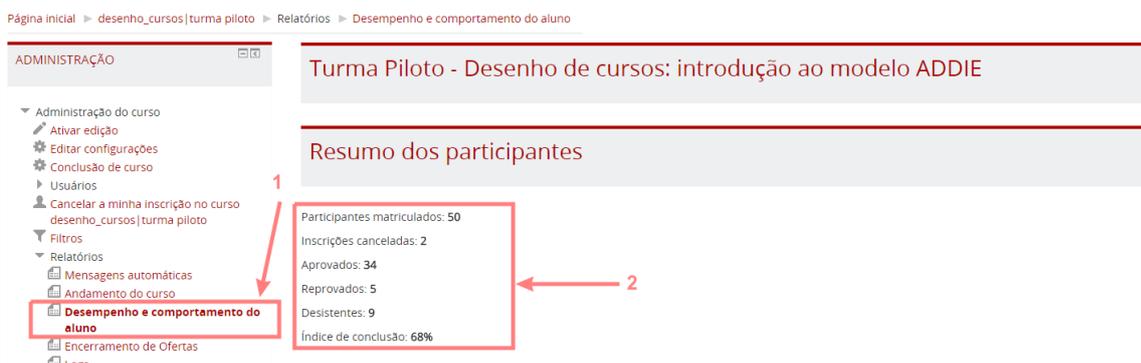


Figura 4.3: Acesso a funcionalidade Relatório de Comportamento e Desempenho dos Alunos a partir da interface de administração do ambiente. Fonte: ENAP.

desistentes são aqueles que não concluíram qualquer atividade no curso, mas não cancelaram sua inscrição, ao contrário dos que cancelaram suas inscrições, exibidos como “Inscrições canceladas”.

Adicionalmente ao resumo dos participantes do curso, é apresentado um gráfico com informações sobre a participação dos alunos em cada uma das atividades como demonstrado na Figura 4.4 no gráfico. O eixo vertical representa a quantidade dos alunos que completaram cada um das atividades. A porcentagem dos alunos é exibida dentro da barra que representa cada atividade para facilitar a leitura enquanto que o nome da atividade está disposto no eixo horizontal do gráfico.



Figura 4.4: Gráfico da participação dos alunos em cada um das atividades propostas. Fonte: ENAP.

A média da participação dos alunos por cada tipo de atividade é apresentada no gráfico “Participação nas atividades (por tipo)” como ilustrado pela Figura 4.5. O valor do eixo vertical corresponde é a média aritmética de alunos que completaram cada uma das atividades agrupadas por tipo. Seguindo o mesmo modelo do gráfico da participação de alunos nas atividades, a porcentagem que cada barra representa é impressa dentro da área da própria barra e o tipo de atividade é descrito no eixo horizontal do gráfico.



Figura 4.5: Gráfico da média da participação dos alunos em cada um das atividades propostas agrupado por tipo de atividade. Fonte: ENAP.

As horas online acumuladas por cada aluno no decorrer do curso são exibidas por um gráfico no formato de dispersão, como pode ser observado na Figura 4.6. Cada um dos pontos do gráfico representa um aluno distribuído uniformemente no eixo horizontal do gráfico. A situação do aluno é representada pela cor do ponto sendo que os alunos aprovados são representados pela cor verde, os alunos reprovados são representados pela cor vermelha, os alunos desistentes são representados por pontos na cor azul e os alunos que cancelaram a participação no curso são representados pela cor preta. Adicionalmente a legenda do gráfico traz o número total de participantes do curso, que é a soma de todos os valores. O número de horas acumuladas está representado pelos valores do eixo vertical do gráfico.

A mesma tela ainda traz a informação da quantidade de acessos ao curso. As barras do gráfico representam a quantidade de acessos ao curso realizados pelos alunos. Para facilitar a leitura o número total de acessos obtidos em cada um dos dias é exibido no topo de cada uma das barras. Cada uma das barras corresponde a um dia de andamento do curso distribuídos ao longo do eixo horizontal do gráfico. Um exemplo do gráfico do número de acessos por dia de curso está representado na Figura 4.7.

Horas online por aluno

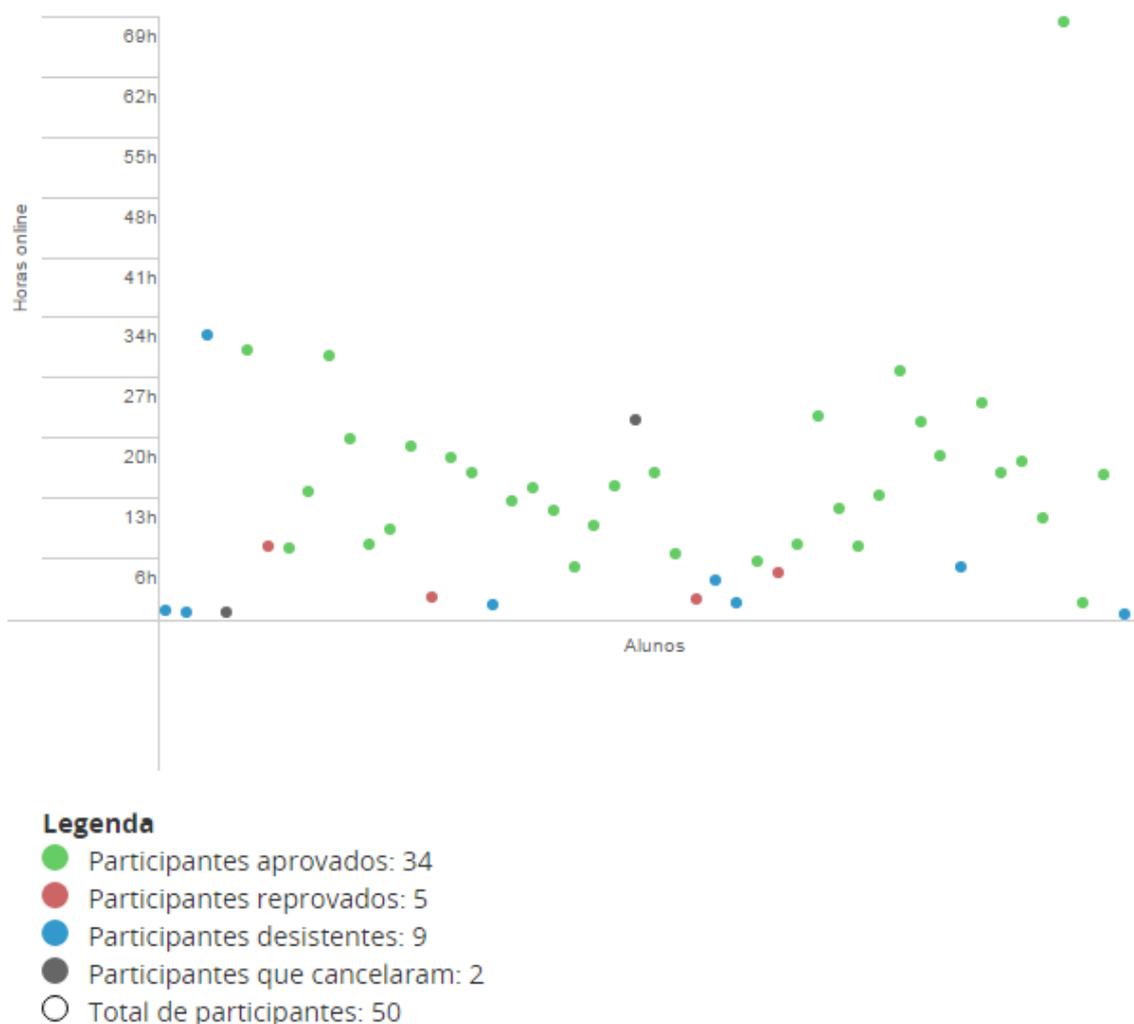


Figura 4.6: Gráfico da média da participação dos alunos em cada um das atividades propostas agrupado por tipo de atividade. Fonte: ENAP.

Ainda na página do relatório, ao final da página há uma tabela que exibe o número total de participantes por Unidade da Federação (UF) ou Estados. A tabela lista cada uma das UFs representadas pela sua sigla e abaixo o número de participantes do curso de origem daquela UF. Finalizando a tela do relatório há um botão “Gerar Relatório” que prepara o relatório para impressão. A tabela de “Participantes por Estado (UF)” e a disposição do botão “Gerar Relatório” podem ser observados na Figura 4.8.

A geração do relatório ficou impossibilitada, entretanto, para turmas maiores, devido ao tempo de processamento requerido para descobrir a quantidade total de horas que o aluno ficou conectado ao sistema e a cursos específicos. Para tanto, foi necessário melhorar o tempo dispensado para a descoberta de horas. A proposta em ZACARIAS et al. para a melhoria

Acessos ao curso

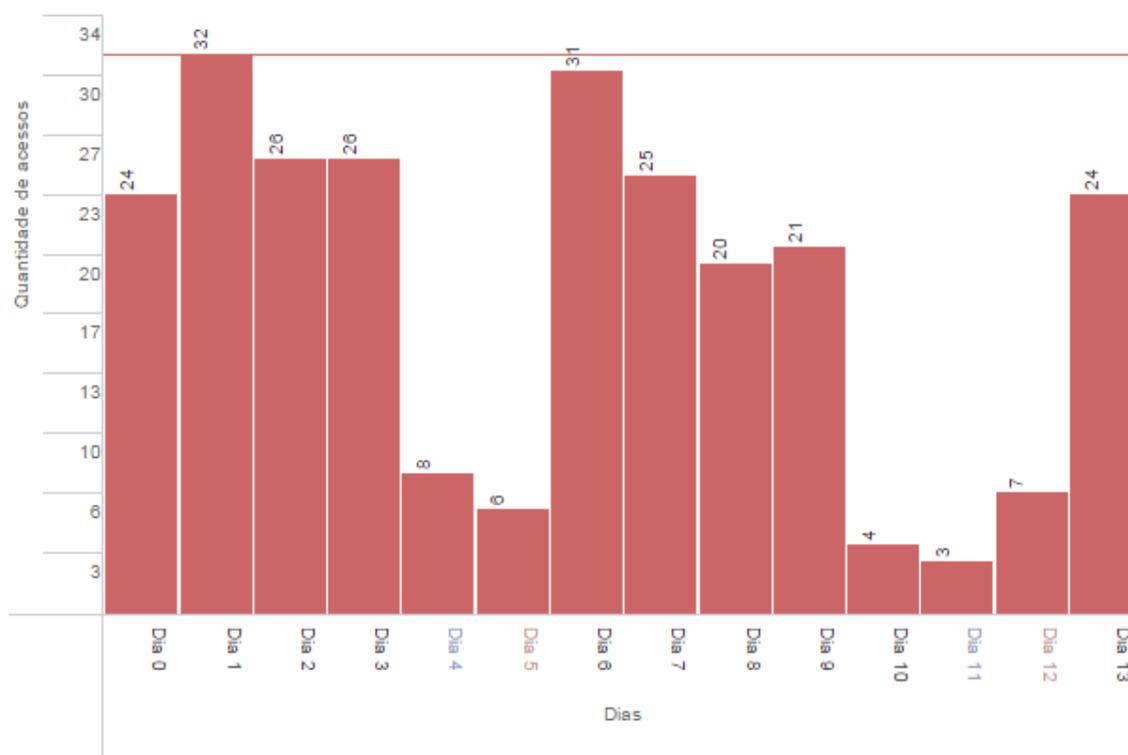


Figura 4.7: Gráfico do número de acessos por dia no decorrer do curso. Fonte: ENAP.

Participantes por Estado (UF)

DF	AC	SP	RJ	PE	MG	SC	PI	AM	GO
28	6	3	2	2	2	2	1	1	1
MS	BA								
1	1								

Gerar relatório

Figura 4.8: Tabela de participantes do curso por UF e botão para preparar relatório para impressão. Fonte: ENAP.

desta funcionalidade busca evitar a dependência nos registros de atividade dos usuários e centralizar o armazenamento dos dados já processados. Portanto, as informações necessárias para a emissão deste relatório são armazenadas em uma tabela exclusiva no banco de dados, que relaciona o estudante e seu tempo total ativo.

Segundo ZACARIAS et al. (2016), como a solução não requer um histórico de acesso, mas somente o valor total final do tempo que o usuário ficou ativo até o momento em que completou o curso, o sistema mantém apenas uma única soma de minutos ativos. Portanto, o registro do tempo do usuário é requerido juntamente com outro registro conectando o usuário e seu curso, e armazenando a soma de minutos específica da relação.

Considerando a impossibilidade de aumentar as capacidades de hardware do servidor para este caso, e também a inviabilidade de aguardar o tempo integral da geração dos relatórios, que pode exceder 10 minutos, a solução busca substituir a carga de processamento na geração do relatório por pequenas requisições periódicas. No final de cada minuto, uma requisição assíncrona é enviada informando que o usuário permanece acessando o sistema. Tal requisição é permitida por meio da tecnologia assíncrona de Javascript e XML (AJAX), implementada nos navegadores por meio da classe XMLHttpRequest (AUBOURG et al., 2016), na primeira parte de desenvolvimento.

A tecnologia AJAX está disponível para todos os navegadores suportados pelo sistema Moodle desde a versão 2.3. Como o próprio Moodle requer Javascript para seu funcionamento, a solução não interfere com a usabilidade e requerimentos de sistema, consumindo os recursos já oferecidos.

Para implementar a solução, duas partes foram necessárias. Um código Javascript é adicionado para o conteúdo padrão de todas as páginas do sistema, sendo estas geralmente os arquivos de tema personalizado. O código cria um evento que faz uma chamada para uma página específica no servidor. A segunda parte é a página do servidor que ouvirá tais chamadas e registrará os dados.

Como demonstrado na Figura 4.9, o código Javascript é incluído no cabeçalho ou no rodapé das páginas do Moodle para ser chamado e executado em todas as páginas do sistema. A inclusão é geralmente feita nos arquivos de temas personalizados, para que desta forma não seja necessário alterar o próprio *framework*. Desta forma, caso seja necessário atualizar os arquivos do próprio *framework*, a operação poderá ser feita sem requerer modificação ou reimplementação da funcionalidade. Então, um evento é iniciado e repetido em um período pré-definido, onde é também definido um tempo mínimo aceitável para considerar como um registro válido de acesso feito pelo usuário.

A Figura 4.10 mostra o processo disparado no servidor a cada vez que uma requisição é executada. O servidor é responsável pelo armazenamento do tempo de acesso do estudante. A cada vez que o servidor é requisitado, os registros pertencentes ao usuário e o tempo de acesso total do usuário ao sistema é recuperado. Depois disso, o tempo mínimo requerido entre uma requisição e outra, considerando o tempo desde a última requisição, é verificado. Se o intervalo estiver acima do período aceitável, o período de tempo pré-definido entre as requisições é adicionado ao tempo total de acesso do usuário ao sistema e ao curso, e o tempo desde a última requisição é atualizada.

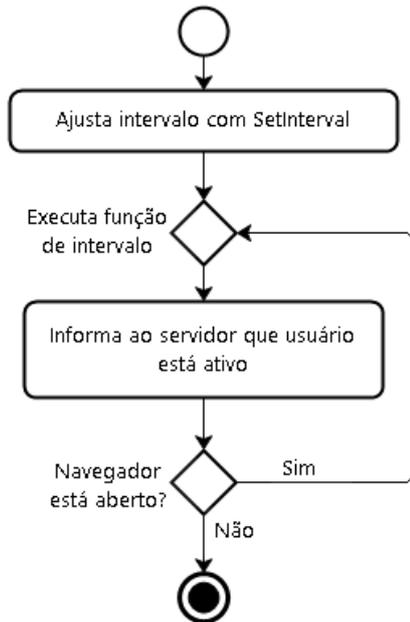


Figura 4.9: Processo do navegador: diagrama do funcionamento do *plugin* no browser. Fonte: ZACARIAS et al. (2016).



Figura 4.10: Processo no servidor: diagrama do funcionamento do *plugin* no servidor. Fonte: ZACARIAS et al. (2016)

É importante observar que o tempo mínimo aceitável entre uma requisição e outra não é exatamente de um minuto. Um espaço mais curto deverá ser definido. Porém, quando uma atualização anterior demora a ser completada, a próxima chamada não está abaixo deste período. Todas as requisições que não atendam a este período mínimo devem ser ignoradas.

Limitar o tempo entre as chamadas é necessário para prevenir que, caso um usuário abra a página mais de uma vez, em várias abas, o tempo total de acesso não seja multiplicado pelo número páginas abertas. Além disso, esta condição também evita que sejam emitidos acessos falso, atribuindo ao usuário mais tempo do que o real acesso.

Caso o servidor não consiga recuperar uma informação prévia do usuário a ser atualizado por falta de registro anterior, ao invés de tentar atualizar o registro incrementando o valor antigo, o que geraria um erro, um novo registro é inserido com a contagem inicial de minutos. Quando um usuário fecha todas as instâncias do sistema Moodle no navegador, o código Javascript também é interrompido, cessando suas requisições recorrentes e parando de atualizar o tempo de acesso.

A solução permite uma maior precisão comparada ao mecanismo de consultar os registros de atividade somando os tempos. O tempo considerado é de fato o período online no sistema, e não depende de valores como o tempo de espera de uma sessão de usuário aberta no sistema. Também pode dar maior precisão, pois é possível monitorar o tempo em que o usuário está com a página aberta. Sendo assim, a solução não depende de interações com o sistema por clicar em links e acessar páginas. Isto permite incluir períodos menos ativos, como os momentos em que o usuário se dedica à leitura de um texto ou assiste a um vídeo embutido na página.

Portanto, por este método é possível obter informações prontas para o armazenamento relacionadas ao tempo que o estudante acessou o sistema sem a necessidade de processamentos adicionais ou requerimentos de consultas complexas e prolongadas por meio de grande quantidade de registros.

4.2 Motivando a Participação de Estudantes via *Plugin* de Mensagens Automáticas para Moodle

O *plugin* de mensagens automáticas proposto em ALMEIDA (2016, 2017) visa aumentar a interação e motivar os estudantes a ficarem conectados com seus cursos. O *plugin* habilita os tutores do ambiente a enviar mensagens personalizadas para os estudantes sempre que eles atingirem determinadas condições pré-definidas.

A Figura 4.11 apresenta a operação de fluxo do *plugin* e as informações necessárias para enviar mensagens para participantes. O fluxo representa os passos do sistema para o envio das mensagens:

- Passo 1: o sistema carrega todos os cursos com turmas em progresso;
- Passo 2: o sistema checa as mensagens que são registradas para um dia determinado de um curso;

- Passo 3: o sistema compara as condições de cada participante em relação ao dia correspondente da turma do curso;
- Passo 4: o sistema carrega os participantes filtrados;
- Passo 5: o sistema prepara as mensagens para os participantes, adicionando o nome do participante e curso conforme cada mensagem;
- Passo 6: o sistema dispara a mensagem para o servidor de mensagens;
- Passo 7: a mensagem é enviada para o e-mail do participante registrado.

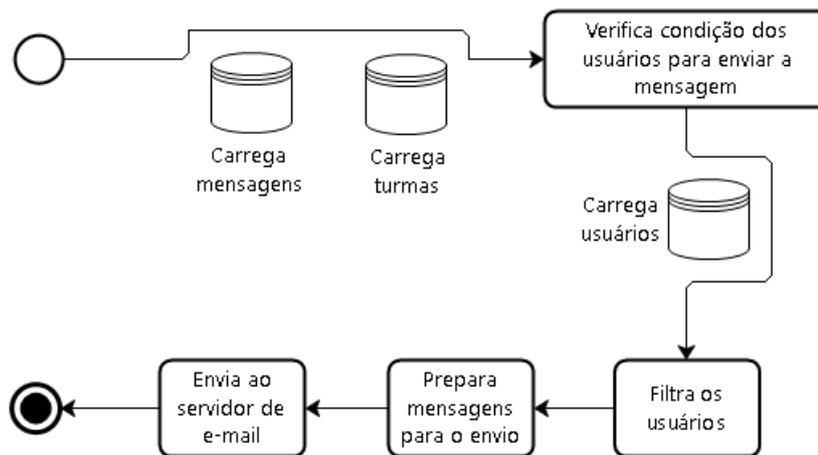


Figura 4.11: Fluxo de operação do plug-in de envio de mensagem. Fonte: acervo pessoal.

Na Figura 4.12, o diagrama de atividade apresentando a informação do fluxo de interações do usuário é descrito. Observa-se que fluxo principal e os outros três fluxos complementam-se entre si e possuem os mesmos objetivos.

Ainda conforme a Figura 4.12 O fluxo principal na interface gráfica do sistema é uma ação de clique no botão rotulado “Nova Mensagem / Editar / Excluir mensagem”, e então o sistema registra o arquivo de mensagem que deve ser enviado para os estudantes em um curso específico. Na primeira alternativa do fluxo A1, o usuário preenche os dados para a mensagem, incluindo a condição de envio, data de envio, assunto e corpo da mensagem e confirma sua conclusão. Os dados são então armazenados no banco de dados e o sistema exibe a mensagem de confirmação, concluindo o fluxo. No fluxo alternativo A2, o usuário altera os dados que deseja e confirma sua conclusão de edição, então as alterações são registradas no banco de dados. Em seguida, o sistema exibe uma mensagem de confirmação e conclui o fluxo alternativo A2. No fluxo alternativo A3, o sistema exibe uma questão de confirmação, atualizando a mensagem e o banco de dados depois de confirmar a ação e exibe a mensagem de conclusão do processo, finalizando o fluxo. O diagrama na Figura 4.12 não exibe quaisquer fluxos de exceção.

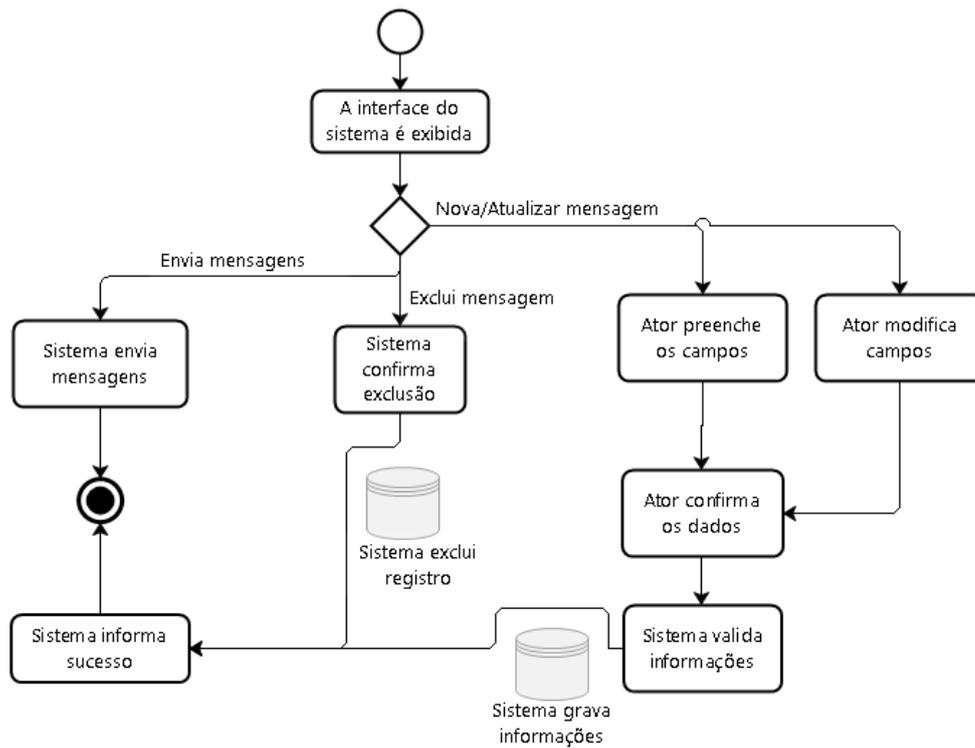


Figura 4.12: Processos de interação do usuário com o *plugin* de mensagem automática. Fonte: acervo pessoal.

Cadastrar / Editar Mensagem

Situação do aluno:*

Inserir um valor.

Selecione uma situação para envio...
Usuário sem acesso
Usuário que já acessou e não possui nota em atividade avaliativa
Usuário sem nota na atividade avaliativa da semana, mas que fez alguma atividade,
Usuário que está abaixo da nota mínima e ainda pode obtê-la
Usuário reprovado que fez alguma atividade avaliativa
Usuário reprovado que fez todas as atividades avaliativas
Usuário aprovado
Usuário desistente
Todos
Selecione uma situação para envio...

Figura 4.13: Tela de seleção com as condições para enviar mensagens para os alunos. Fonte: ALMEIDA et al. (2016)

A Figura 4.13 exibe tipos de condições pré-definidas que são consideradas na proposta do *plugin*. São estas:

- Usuários sem acessos, isto é, que nunca acessaram o sistema de aprendizagem virtual;
- Usuários que já acessaram o sistema, mas ainda não têm nenhuma avaliação gerada na conclusão de atividades desde que o curso iniciou;
- Usuário não recebeu nenhuma nota de avaliação nos últimos sete dias, contados a partir da data de envio da mensagem, mas em outros momentos ele concluiu alguma atividade;
- Usuário obteve notas abaixo da pontuação mínima, mas este ainda pode concluir o curso com sucesso se melhorar suas notas nas próximas avaliações;
- Usuários que falharam na execução de todas as atividades;
- Usuário completou com sucesso todas as atividades até o momento;
- Usuários que abandonaram o curso;
- Todos os estudantes, independentemente de sua condição.

Além da seleção dos grupos de estudantes por condições, o estudante pode, é claro, ser selecionado baseado em outros critérios educacionais que podem ser específicos para certos cursos.

Conforme informa ALMEIDA (2017) em sua revisão da ferramenta, além de ser possível selecionar os estudantes por considerar alguma condição específica, o tutor do sistema de ensino à distância pode também selecionar estudantes por cursos e observar o histórico de envio de mensagens. A Figura 4.14 apresenta um exemplo de um quadro com o histórico de mensagens enviadas.

Modelos específicos de mensagens são automaticamente carregados a fim de facilitar o trabalho dos tutores do sistema. O *plugin* proposto exibe um modelo de mensagem para a segunda condição da Figura 4.13. Observe-se que a mensagem pode ser corrigida e ajustada, se necessário, para mencionar aspectos específicos relacionados aos participantes selecionados. Como demonstrado na Figura 4.14, uma vez que a condição e o período são escolhidos, o modelo de mensagem aparece com os campos preenchidos do assunto e o texto.

Na Figura 4.15, outro exemplo de modelo de mensagem é exibido. Este exemplo é relativo à situação em que o participante está abaixo da média mínima, mas ainda pode concluir o curso com sucesso. Perceba-se que mensagem alerta ao participante que o curso está sendo encerrado. Isto indica aos mesmos que, a despeito dos participantes não possuírem uma média mínima para aprovação, com um pouco mais de trabalho e esforço, o estudante estará apto para completar as tarefas e atingir a nota requerida.

Turma 2/2015 - Siconv para Convenientes 2 – Proponentes e Usuários

Mensagens

Cadastrar Mensagem

Situação do Aluno	Mensagem	Dia do curso	Histórico de envio	Editar	Excluir
Usuário desistente	Olá {nome}, O curso {nome_curso} terminou dia {data_fim}. Verificamos que após o en...	22º	quarta, 16 Dez 2015, 11:23		
Usuário reprovado que fez alguma atividade avaliativa	Olá {nome}, O curso {nome_curso} terminou dia {data_fim}. Consta em nosso sist...	22º	quarta, 16 Dez 2015, 11:22		
Usuário reprovado que fez todas as atividades avaliativas	Olá {nome}, O curso {nome_curso} terminou dia {data_fim}. Consta em nosso sistema qu...	22º	quarta, 16 Dez 2015, 11:21		
Usuário aprovado	Olá {nome}, Parabéns pela conclusão do curso {nome_curso}! Esperamos ter contribu...	22º	quarta, 16 Dez 2015, 11:21		
Usuário que está abaixo da nota mínima e ainda pode obtê-la	Olá {nome}, Hoje é o último dia do curso {nome_curso} . Não perca a	20º	segunda, 14 Dez 2015, 11:13		

Figura 4.14: Tela de seleção com as condições para enviar mensagens para os alunos.
Fonte: ALMEIDA et al. (2016)

Não perca tempo, o curso Introdução à Gestão de Projetos já está em andamento!  

 **Escola Virtual Enap** <ead_noreply@enap.gov.br> 06/05/2016 ☆  
para mim ▾

E aí, Luciano Rosa de Almeida! Tudo bem?

Estou entrando em contato mais uma vez para te lembrar de que o curso Turma 2/2016 - Introdução à Gestão de Projetos começou no dia 03/05/2016 e já está quase terminando, porque possui três semanas de duração!

Aproveite essa oportunidade para realizar as atividades previstas sem correria. Não deixe para a última hora!

Esta é uma excelente oportunidade para adquirir novos conhecimentos. Que tal mais um certificado para o seu currículo?

Não sabe ou não se lembra de como acessar a Escola Virtual Enap? Não tem mistério. Veja, abaixo, algumas orientações para te ajudar a entrar no curso:

Acesse o site: <https://enavirtual.enap.gov.br>

Se você já fez algum curso a distância na Enap a partir de 2015, a senha que você já usava continua válida.

Caso contrário, seu usuário e sua senha correspondem ao seu CPF.

Em caso de dificuldades, entre em contato pelo e-mail: ead@enap.gov.br

Conte comigo sempre que precisar!

Coordenador de curso a distância

Coordenação-Geral de Educação a Distância

Diretoria de Desenvolvimento Gerencial

Escola Nacional de Administração Pública - Enap

Figura 4.15: Exemplo de mensagens recebidas pelo participante no final do curso.
Fonte: ALMEIDA et al. (2016)

Capítulo 5

Resultados

5.1 Resultados obtidos com a otimização do acesso aos registros no banco de dados

A validação do método proposto foi possível graças à ENAP, por disponibilizar uma cópia banco de dados de produção da organização, contendo dados reais de acesso. Para o propósito de teste de desempenho, foi escolhido o curso com mais atividades de usuário, oferecido e concluído na metade do ano de 2015. Devido ao número de registros, cursos de tamanhos similares foram impedidos de gerar quaisquer relatórios que requeressem o conhecimento do tempo que os usuários permaneceram ativos, com o sistema aberto. Para a validação, foi obtido e comparado o tempo das consultas no banco de dados por meio de ambas as soluções.

A Tabela 5.1 exibe uma comparação do tempo de consulta aos registros do sistema em cada solução para obter todas as informações de todos os estudantes que acessaram um curso específico no sistema.

Tabela 5.1: Comparação do tempo de consulta ao selecionar todos os cursos em uma única requisição. Fonte: ZACARIAS et al. (2016)

Parâmetro	Solução atual	Solução proposta
Tempo (milissegundos)	17047 ~ 17779	63 ~ 84
Contagem de registros	824044	7059

A implementação atual calcula a soma de tempo entre cada interação do usuário que gerou o carregamento de uma página do sistema. Isto requer que cada registro de atividade que pertence ao usuário seja recuperado do banco de dados para análise. O número de registros obtidos diz respeito ao número de interações que geraram o carregamento de uma página do curso específico, incluindo todas as atividades de todos os usuários.

A solução proposta, entretanto, adiciona o tempo de acesso do usuário enquanto este ainda está conectado ao sistema e armazena a soma em um único registro relacionado ao usuário e ao curso. Para recuperar os dados totais para cada usuário do curso, o sistema recupera os registros pertencentes ao próprio curso. A Tabela 5.1 obteve 7059 registros, que é também o número de usuários que acessaram o curso testado, sendo um registro para cada aluno.

Observe-se que, no caso da Tabela 5.1, a consulta realizada na implementação atual recupera todos os dados relacionados à atividade do usuário. Isto inclui, mas não se limita à fonte do acesso, endereço IP que causou o acesso, o tempo exato da atividade registrada, a atividade específica acessada. Entretanto, fica claro que a implementação não possui qualquer interesse em tais dados, mas somente no tempo em que cada atividade ocorreu e no código do usuário que identifica tal atividade.

Esta prática remete ao fato de não especificar as colunas que devem ser recuperadas do banco, fazendo com que este recupere todas as informações associadas e podendo, inclusive, prejudicar o desempenho da consulta em termos de uso de memória e tempo. Em face disto, para uma análise mais justa, decidiu-se otimizar a consulta especificando quais os campos que devem ser retornados e executar os testes novamente.

A Tabela 5.2 exhibe a diferença de tempo entre as três consultas: a solução atual não otimizada, a solução atual otimizada e a solução proposta. Por especificar os campos que serão recuperados na consulta, ignorando os dados desnecessários anteriormente obtidos, a consulta otimizada obteve uma melhora de até 59% a respeito do tempo de demora para a geração do resultado. A melhora foi considerável, sendo de aproximadamente 7 segundos em relação à implementação atual por aluno, mas ainda não é satisfatória para a geração do relatório. Entretanto, a mesma variação tão impactante não ocorre na solução proposta, dada a pequena quantidade de dados recuperados, que está relacionado ao tempo final de acesso do aluno, não à quantidade de interações do mesmo. A comparação dos tempos entre a solução original presente no AVA Moodle e a solução proposta neste trabalho é demonstrada na Figura 5.1. As duas primeiras barras correspondem ao tempo necessário para a geração do relatório na solução original sem otimização da consulta SQL e após a otimização. As duas barras mais à direita correspondem ao tempo necessário para a geração do relatório obtido com a solução proposta, com e sem otimização da consulta SQL.

Tabela 5.2: Comparação do tempo de consulta selecionando colunas específicas.

Fonte: ZACARIAS et al. (2016)

Solução testada	Tempo de execução	Registros consultados
Solução atual (não otimizada)	17047 ~ 17779	824044
Solução atual (otimizada)	10034 ~ 10420	824044
Solução proposta (não otimizada)	63 ~ 84	7059
Solução proposta (otimizada)	62 ~ 80	7059

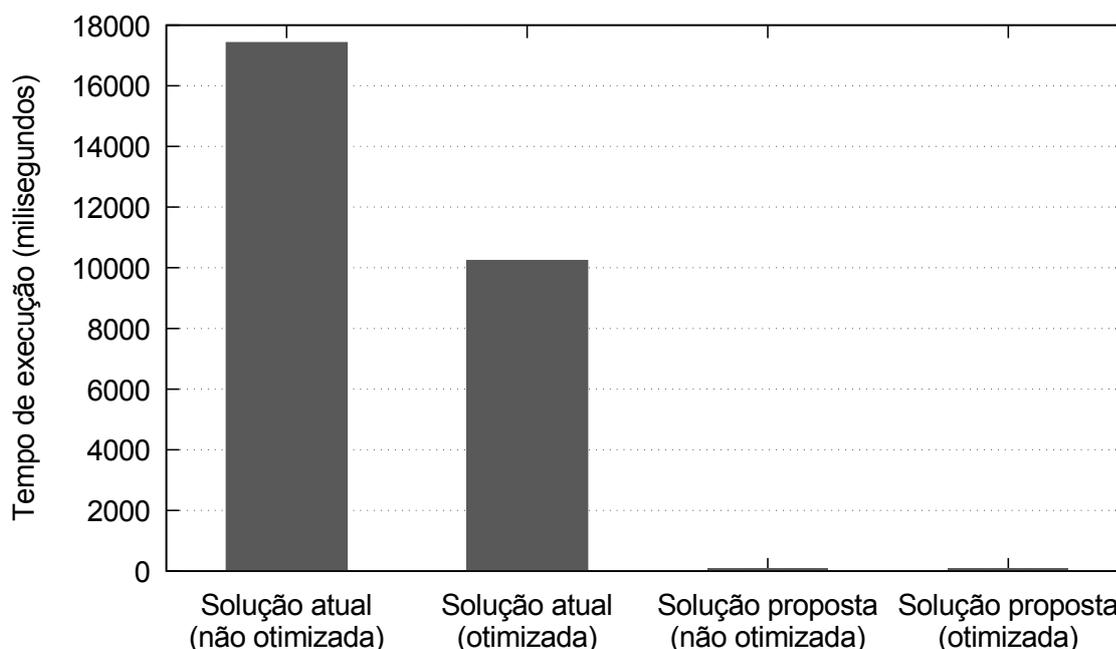


Figura 5.1: Comparação do tempo de execução das consultas. Fonte: acervo pessoal.

Um outro problema que aparece ao recuperar todos os dados em uma requisição simples está relatado ao grande uso de memória virtual do PHP. Uma vez que a transformação e soma dos registros recuperados do banco de dados são tratados no PHP, e para tanto cada registro é convertido em um objeto, isto causa um grande consumo de memória. Desta forma, quando uma consulta retorna muitos registros, dado o processamento dos dados, além do tempo despendido para a operação, o limite de memória virtual disponível para o processo é atingido e a requisição é inteiramente cancelada (Moodle Pty Ltd, 2016).

Para resolver este problema, na abordagem atual foi decidido selecionar os registros de um único usuário a cada vez, e repetir a consulta para cada usuário do curso, exigindo que o banco localize e filtre as informações requeridas de toda a tabela a cada vez.

A consulta executada na Tabela 5.3 mostra uma comparação entre o método modificado desta forma na solução atual para um único usuário e a solução proposta, também para um único usuário. Baseado neste teste, é possível observar que recuperar os dados de um único usuário é viável em ambas as soluções. Entretanto, quando há a necessidade de iteração por meio de cada usuário do curso, o processo pode tornar-se ainda mais lento na primeira solução. O curso testado excede a contagem de 7 mil estudantes, desta forma, exigindo que se multiplique o processo pelo tamanho total da turma, o que significaria uma demora de até 3500 segundos, aproximadamente 1 hora. A solução proposta, embora dispensável pela pouca quantidade de dados que precisam ser recuperados, poderia utilizar deste mesmo método e ainda assim estaria habilitada a realizar todo o procedimento para os mais de 7 mil estudantes em aproximadamente 5 minutos.

Tabela 5.3: Comparação do tempo de consulta por usuário. Fonte: ZACARIAS et al. (2016)

	Solução atual	Solução proposta
Tempo (milissegundos)	586 ~ 647	16 ~ 36
Contagem de registros	16225	1

Ao compararmos a Tabela 5.2 e a Tabela 5.3, podemos notar que utilizar a solução proposta ao buscar os dados de cada estudante em turmas maiores é inviável. Ainda, o método da Tabela 5.3 não é necessário para a solução proposta, pois é possível recuperar o tempo de todos os estudantes de um curso em uma requisição simples sem exceder o limite de memória virtual consumida, interrompendo a requisição, uma vez que a quantidade de dados é muito menor.

A solução proposta resolve o problema de conhecer o tempo gasto acessando o sistema em qualquer curso, atendendo às necessidades da ENAP. A proposta oferece uma implementação que permite a geração de informações para os relatórios que requerem o tempo de acesso do aluno de uma maneira viável.

Sem alterações adicionais, a solução não oferece dados mais precisos sobre as atividades que os estudantes executaram durante o período em que acessaram o curso, tampouco pode consultar apenas períodos específicos de tempo dentro de um curso. Entretanto, é possível adaptar a solução para obter estas e outras informações mais específicas de acordo com as necessidades de cada caso, ou ainda, para casos muito específicos que demandam grande quantidade de informações, o registro de atividades padrão ainda estará disponível e poderá ser consultado.

5.2 Avaliação do engajamento dos estudantes proporcionado pelo *Plugin* de mensagens automáticas

Para validar a eficiência da proposta, ALMEIDA et al. (2016, 2017) compara a interação de estudantes do sistema de aprendizagem à distância da ENAP antes e depois do *plugin*. Com a finalidade de analisar a interação com os estudantes, considerou-se a quantidade de estudantes que bloquearam sua inscrição nas turmas dos anos de 2011 e 2012, quando esta solução ainda não era aplicada. Por outro lado, os estudantes foram motivados pelo *plugin* de mensagem automáticas nos anos de 2013, 2014 e 2015.

Para completar a análise, assumiu-se o mesmo período de tempo, mas agrupando os estudantes em duas categorias. Na primeira, considerou-se que os estudantes não receberam mensagens automáticas em 2015, enquanto na segunda categoria considerou-se aqueles que receberam. Os estudantes do primeiro grupo foram divididos em cinco classes, conforme visto na Tabela 5.4. Observe-se que a média de acesso de estudantes por dia, no grupo, foi

de 23.11% para os cursos sem mensagem automática. Entretanto, conforme o exibido na Tabela 5.5, para os cursos que atuaram com o sistema de mensagens automáticas, a porcentagem de estudantes por dia foi de 37.1% do total de participantes inscritos no curso. Desta forma, o ganho oferecido pelo *plugin* de mensagem automática proposto foi significativo.

Tabela 5.4: Total e Porcentagem de acessos em dias em que não foram enviadas mensagens. Fonte: Enap 2015.

Turmas	Quantidade média de Alunos	Médias de acesso de alunos	%
Classe 1/2015	1,085	269	24.8%
Classe 2/2015	1,208	289	23.9%
Classe 3/2015	1,05	258	24.6%
Classe 4/2016	1,371	279	20.4%
Classe 5/2016	985	216	21.9%
Total	5,699	262.2	23.11%

Tabela 5.5: Total e Porcentagem de acessos em dias com mensagens automáticas são enviadas pelo plugin. Fonte: Enap 2015.

Turmas	Quantidade média de Alunos	Médias de acesso de alunos	%
Classe 4/2015	1,371	349	35.5 %
Classe 5/2016	985	532	38.8 %
Total	2,356	440.5	37.1%

Capítulo 6

Conclusões e trabalhos futuros

Muitos relatórios desenvolvidos na ENAP coletam seus dados diretamente do registro de atividades do Moodle. A abordagem proposta neste trabalho permite a normalização das informações e construção de consultas mais rápidas, além de melhoras na precisão das informações por armazená-las em conjunto com outras informações desejáveis, conforme as necessidades. Como o Moodle é uma ferramenta amplamente permissiva quanto a mudanças e implementação de novos recursos, é possível adicionar novas funcionalidades sem realizar alterações diretamente no código-fonte do mesmo, o que dificultaria a manutenção e atualizações menores de versão. Além disto, ao implementar os recursos no formato de *plugins*, torna-se mais fácil de transpor o código existente para outra instalação Moodle.

No contexto anteriormente mencionado, a observação do comportamento dos participantes por meio das ferramentas de aprendizagem online permitiu a visualização de que simples estímulos na comunicação podem auxiliar e conectar os estudantes para o ambiente da escola virtual, aumentando sua participação nos cursos seguidos. O trabalho apresentado descreveu um *plugin* para a plataforma de ensino virtual Moodle, que automaticamente envia mensagens para os estudantes de acordo com sua assiduidade em completar as tarefas dos cursos. O resultado do uso do *plugin* proposto demonstrou uma significativa diminuição nas taxas de evasão e falha comparadas às taxas antes do uso do *plugin*. O estudo também revelou que os dias em que os estudantes receberam as mensagens, seu compromisso em acessar o ambiente virtual foi maior do que nos dias em que eles não receberam as mensagens. Isto evidencia a utilidade da proposta ao engajar os estudantes nos cursos em que estão inscritos.

6.1 Trabalhos futuros

O relatório de desempenho e comportamento visa agrupar informações importantes sobre o comportamento geral dos alunos em turmas encerradas ou em andamento. Como sugestão de melhoria em trabalhos futuros, podem ser buscadas mais informações para incluir no relatório, a fim de incrementar a quantidade de informações disponíveis no mesmo. Também é possível estudar a melhor forma de apresentação das informações contidas no mesmo.

O *plugin* para otimização do acesso aos dados sobre a quantidade de horas online dos alunos também abre espaço para melhorias de desempenho e precisão. Embora apresente um controle maior sobre o tempo dos usuários no sistema, aumentando a precisão em relação à forma de contagem anterior, como sugestão de trabalho futuro pode ser trabalhada uma ferramenta para detectar o intervalo ideal entre uma requisição e outra, de forma que se descubra o intervalo ideal para manter uma precisão esperada e não sobrecarregar o servidor com muitas requisições. A ferramenta pode verificar a quantidade de processamento no servidor e a quantidade de sessões abertas ou o número de requisições por minuto, e definir um limite para aumentar ou diminuir o intervalo entre as requisições. Desta forma, pode-se manter certo equilíbrio entre precisão e desempenho. Além disso, pode-se ampliar os dados coletados para refletirem no armazenamento do tempo dos alunos por atividade, e também desconsiderar tempos em que o usuário esteve inativo no sistema, ainda que com alguma página aberta no mesmo.

Para o sistema de envio de mensagens automáticas, sugere-se incluir um sistema de controle a fim de conhecer as mensagens que foram, de fato, recebidas e lidas pelos alunos, além de ampliar o sistema para a criação de modelos de mensagens gerais, que possam ser aplicadas a todas as turmas da escola, automatizando parte do serviço dos tutores. Também, pode-se coletar dados do acesso de alunos e emitir notas automáticas sobre a realização de atividades e outras notificações de interesse dos alunos.

Além disso, é possível realizar trabalhos adicionais sobre estes apresentados, de forma a descobrir mais sobre o comportamento do aluno no sistema, comprometimento, causas de evasão e falta de acesso e padrões relacionados aos momentos em que o usuário acessa.

Referências Bibliográficas

ALMEIDA, L. R. de. *Desenvolvimento de Soluções Visuais, de Usabilidade, de Monitoramento e de Interação Automática para Plataformas de Educação à Distância*. Dissertação (Dissertação de Mestrado) — Universidade de Brasília, Brasília, mar. 2017.

ALMEIDA, L. R. de et al. Motivating attendee's participation in distance learning via an automatic messaging plugin for the moodle platform. In: *2016 IEEE Frontiers in Education Conference (FIE)*. [S.l.: s.n.], 2016. p. 1–5.

AUBOURG, J. et al. *XMLHttpRequest Level 1*. [S.l.], 2016.
<https://www.w3.org/TR/2016/NOTE-XMLHttpRequest-20161006/>.

BABIĆ, S.; KRESIĆ, M.; KUCEL, K. E-education 2.0: Students' digital identity and online learning activities. In: *2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. [S.l.: s.n.], 2014. p. 756–761.

BEKELE, T. A. Motivation and satisfaction in internet-supported learning environments: A. *Subscription Prices and Ordering Information*, JSTOR, p. 116, 2010.

BERNERS-LEE, T.; FIELDING, R.; FRYSTYK, H. *Hypertext Transfer Protocol – HTTP/1.0*. United States: RFC Editor, 1996.

BOVO, A. et al. Clustering moodle data as a tool for profiling students. In: *2013 Second International Conference on E-Learning and E-Technologies in Education (ICEEE)*. [S.l.: s.n.], 2013. p. 121–126.

BRASIL, P. *Instituições de ensino superior têm novas normas para cursos a distância*. 2016. Disponível em: <<http://www.brasil.gov.br/educacao/2016/03/instituicoes-de-ensino-superior-tem-novas-normas-para-cursos-a-distancia>>.

CÉSPEDES, J. S.; RIVAS, M. C. Towards an implementation of a short messaging system over an e-learning environment. In: *7th Iberian Conference on Information Systems and Technologies (CISTI 2012)*. [S.l.: s.n.], 2012. p. 1–6. ISSN 2166-0727.

DALMAU, M. B. L. *Introdução à educação a distância*. 3. ed. Florianópolis, Brasil: UFSC, 2014.

DOUGIAMAS, M. *Moodle-Open-source learning platform*. Perth, Australia, 2001. Release 3.2.

FERDIANA, R. Distance learning lifecycle management: An agile approach to manage the lifecycle of a distance learning. In: *2013 International Conference on Information Technology and Electrical Engineering (ICITEE)*. [S.l.: s.n.], 2013. p. 106–112.

FIELDING, R. et al. *Hypertext Transfer Protocol – HTTP/1.1*. United States: RFC Editor, 1999.

FLANAGAN, D. *JavaScript: The Definitive Guide*. [S.l.]: O'Reilly Media, Incorporated, 2011. (Definitive Guide Series). ISBN 9780596805524.

FOUNDATION, M. *Introdução - JavaScript | MDN*. 2017. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Guide/Introduction>>.

FRANCO, S. R. K. et al. Aprendizagem na educação à distância: caminhos do brasil. *RENOTE*, v. 4, n. 2, 2006.

GRIGORIK, I. Making the web faster with http 2.0. *Commun. ACM*, ACM, New York, NY, USA, v. 56, n. 12, p. 42–49, dez. 2013. ISSN 0001-0782. Disponível em: <<http://doi.acm.org/10.1145/2534706.2534721>>.

HIJÓN-NEIRA, R.; VELÁZQUEZ-ITURBIDE, J. A. How to improve assessment of learning and performance through interactive visualization. In: *2008 Eighth IEEE International Conference on Advanced Learning Technologies*. [S.l.: s.n.], 2008. p. 472–476. ISSN 2161-3761.

IJTIHADIE, R. M. et al. E-mail based updates delivery in unidirectional content synchronization among learning management systems over limited bandwidth environment. In: *TENCON 2011 - 2011 IEEE Region 10 Conference*. [S.l.: s.n.], 2011. p. 211–215. ISSN 2159-3442.

JUNIOR, F. S. M. *Interatividade e interface em um Ambiente Virtual de Aprendizagem*. [S.l.]: IMED - MERIDIONAL, 2008. ISBN 9788599924136.

KUROSE, J.; ROSS, K. *Computer Networking: A Top-down Approach*. Pearson, 2013. (Always learning). ISBN 9780132856201. Disponível em: <<https://books.google.com.br/books?id=fWg5pwAACAAJ>>.

MEDEIROS, M. F. de; FARIA, E. T. *Educação a distância: cartografias pulsantes em movimento*. [S.l.]: EDIPUCRS, 2003. ISBN 9788574304021.

MEHLECKE, Q. T. C.; TAROUCO, L. M. R. AMBIENTES DE SUPORTE PARA EDUCAÇÃO A DISTÂNCIA: A mediação para aprendizagem cooperativa. *World Wide Web Internet And Web Information Systems*, p. 1–13, 2003.

Moodle Pty Ltd. Data manipulation API. *Moodle Developer Documentation*, Sep 2016. Acesso em: maio de 2017. Disponível em: <https://docs.moodle.org/dev/Data_manipulation_API>.

MOORE, M.; KEARSLEY, G. *Distance Education: A Systems View of Online Learning*. 3. ed. [S.l.]: Cengage Learning, 2011. ISBN 9781133715450.

NAGI, K.; SUESAWALUK, P. Research analysis of moodle reports to gauge the level of interactivity in elearning courses at assumption university, thailand. In: *2008 International Conference on Computer and Communication Engineering*. [S.l.: s.n.], 2008. p. 772–776.

PEREIRA, A. S. et al. *Metodologia da Aprendizagem em EAD*. 1. ed. [S.l.]: Universidade Federal de Santa Maria, 2017. v. 1.

PIÑA, A. A. An overview of learning management systems. *Virtual Learning Environments: Concepts, Methodologies, Tools and Applications*. USA: IGI Global, p. 33–51, 2012.

SEBESTA, R. W. *Programming the World Wide Web*. [S.l.]: Pearson Education, 2014. ISBN 9780133776096.

WANG, Y.-H.; WANG, W.-N.; YEN, Y.-H. An intelligent semantic agent for e-learning message communication. In: *19th International Conference on Advanced Information Networking and Applications (AINA'05) Volume 1 (AINA papers)*. [S.l.: s.n.], 2005. v. 2, p. 105–108 vol.2. ISSN 1550-445X.

ZACARIAS, E. et al. Optimizing the access records of students in the moodle virtual learning environment database. *IFAC-PapersOnLine*, v. 49, n. 30, p. 98 – 101, 2016. ISSN 2405-8963. 4th IFAC Symposium on Telematics Applications TA 2016. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S2405896316325861>>.

Apêndice A

Artigos publicados

Esta seção apresenta, como apêndice ao presente trabalho, dois artigos publicados no segundo ano do Mestrado em Engenharia Elétrica. Os artigos são resultados de trabalhos desenvolvidos em parceria com a ENAP e as modificações propostas foram efetivamente aplicadas nas ferramentas de ensino à distância em produção. O primeiro artigo apresenta solução aplicadas com o intuito de otimizar acesso aos dados do ambiente virtual de ensino e aprendizagem, a ferramenta Moodle, e com isso permitir que relatórios destes dados sejam gerados de forma mais eficiente, utilizando menos processamento e portanto reduzindo o tempo necessário para a geração de tais relatórios. O segundo artigo em co-autoria com o Bel. Luciano Rosa de Almeida apresenta um *plugin* para incentivar a participação dos alunos nos cursos na modalidade de ensino à distância oferecidos pela ENAP assim como diminuir a taxa de evasão dos alunos nos cursos em andamento.

A.1 Optimizing the Access Records of Students in the Moodle Virtual Learning Environment Database

- Título: Optimizing the Access Records of Students in the Moodle Virtual Learning Environment Database
- Veículo: IFAC Symposium on Telematics Applications (TA 2016)
- URL: <<https://doi.org/10.1016/j.ifacol.2016.11.135>>
- Data: 6–9 de Novembro de 2016
- Local: Porto Alegre, Brasil

Optimizing the Access Records of Students in the Moodle Virtual Learning Environment Database

Eliakim Zacarias¹, Luciano R. de Almeida¹, Juliano B. Prettz¹, João Paulo C. L. da Costa^{1,2,3}, Edison P. de Freitas⁴, Edna D. Canedo¹, and Rafael T. de Sousa Júnior¹

¹Department of Electrical Engineering, University of Brasília, Brazil

²Institute for Information Technology, Ilmenau University of Technology, Ilmenau, Germany

³Fraunhofer Institute for Integrated Circuits IIS, Erlangen, Germany

⁴Institute of Informatics, Federal University of Rio Grande do Sul, Brazil

edison.pignaton@inf.ufrgs.br, ednacanedo@unb.br, eliakim.zacarias@gmail.com, joaopaulo.dacosta@ene.unb.br, prettzb@gmail.com, luciano.ralm@gmail.com, desousa@unb.br

Abstract: Moodle is a freeware solution for Learning Management System (LMS) applied to Virtual Learning Environments (VLS). Despite the several features to support online courses, some desirable information is not easily provided, for instance, the amount of time the user spent in the system or in a specific course. A custom implementation can measure the time spent on the user's searches into the internal Moodle log and the distance between the records through the session time. However, for a large number of recorded logs, searching a single record passing through the other records requires a large time processing. As a consequence, the generation of custom reports is also costly. This work proposes a real-time plugin for Moodle to count the time spent by a student logged into the system retrieving the information from a pre-computed central table. The objective of the solution is to improve the counting precision and allow a faster retrieval of data. Our proposed plugin is validated using real data from the National School of Public Administration (Enap) in Brazil.

© 2016, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.

Keywords: Generation of Reports, Moodle Plugin, Time Counting Precision, Measuring Online Time.

1. INTRODUCTION

Although Moodle [1] is a popular platform for distance learning environments, there are still essential limitations such as the lack of tracking tools for the students. However, these limitations can be overcome by adding third-party solutions such as plug-ins or even a whole separate system [2].

The student tracking data is crucial to generate reports monitoring the student's behavior in the offered courses. Moodle has available the user activity log, which can be used to control and to capture relevant tracking information.

The National School of Public Administration, in Portuguese Escola Nacional de Administração Pública (Enap), provides distance learning courses for the federal employees in Brazil using Moodle. In order to support the students to improve their performance, Enap used an ad hoc solution that seeks all access records of selected students in a certain course. Search loops were repeated for all students passing through the activity log data by filtering course from the course start date until the end date. Despite the feasibility in small databases, such approach is unfeasible if the activity log increases. Therefore, the total query time was proportional to the amount of selected students.

This paper aims to provide a solution to the problem by standardizing data and separating them from the activity logs so the query can be made by consulting a single record.

This paper is divided into four sections besides this introduction. Section 2 presents and discusses the state of the art related to the subject. Section 3 presents the proposed approach to address the problem, while Section 4 presents the acquired results. Section 5 concludes the paper and also provides directions for future work.

2. RELATED WORKS

Moodle [1] is one of the leading platforms for free distance learning. It is an open source system that includes the participation of collaborative communities for their development. Its structure is formed by modules or extensions, smaller systems that provide specific or specialized features. These modules are also called plug-ins [2].

NAGI and SUESAWALUK [3] present an analysis based on reports of Moodle to measure the level of interactivity in the virtual environment of teaching courses at the University of the Assumption. The aim of [3] is the use of data captured behavior and iteration of students in the Moodle environment. These pieces of information are collected in real-time, including information that is not registered in the database, like views and clicks. One of the conclusions in [3] is that a communication platform to improve the interaction between students and instructors can be created and a ranking of the courses with more iterations and collaborations of students can be also made.

In BOVO, 2013 [4] an analysis of the log of the virtual system Moodle learning is made. With data mining techniques, machine learning and artificial intelligence indicates which groups are the most relevant to form support tools for managers and teachers. Such data mining techniques can also predict the behavior of the students and their performance. It can also allow the teachers and mentors trace paths and counseling so that students get better results.

In [5], managers and teachers needed the complete information on the behavior of students, in order to analyze and propose ways to improve student achievement. To meet this demand the Merlin was developed, composed of three layers. The first layer contains the data of the education system. The second layer consists of tools that analyze and form a database of system logs, cookies, logins on the system and polls of answers, all with the function to track and record the actions of students during activities in the virtual system. The third layer is the presentation of reports based on pooled data from data mining tools.

3. THE PROPOSED METHOD

The proposal of this paper seeks to avoid dependence on activity logs and centralize the storage of already processed data. Thus, the search of the required information is held in a separate table, which links the students and their total active time.

Since the solution does not require a history of accesses, but only the final total time the user remained active until completion of the course, the system keeps only the total sum of minutes. Thus, the user time registration is required, along with another record connecting the user and the course, storing the sum of this.

Considering the impossibility to increase the server's hardware capabilities for this case, and also the impossibility to wait for the full-time report generation that could exceed 10 minutes, the solution aims to replace the intensive processing load during the report generation through periodic requests. At the end of each minute, an asynchronous request is sent informing that the user remains accessing the system. Such a request is allowed through technology for Asynchronous Javascript and XML (AJAX), implemented in browsers as the XMLHttpRequest class, in the first level of development [6].

The AJAX technology is possible and accessible to browsers that support the Moodle system since version 2.3. Since Moodle itself requires JavaScript for its better functioning, the solution does not interfere with the usability and requirements, taking advantage of resources already offered.

To implement the solution, two parts were needed. A Javascript is added to a standard content to all pages, usually to the theme customization files. The code creates an event that happens every minute, which makes a call to a specific particular page on the server.

As shown in Figure 1, the Javascript code includes the header or footer of pages of Moodle, to print and run on all system pages. Inclusion is usually performed in the theme customization files. Then, an event is initiated and repeated at

a predefined interval as the minimum time that should be considered for user access, also in view of each iteration, a request to the server is made. The code makes the request to a specific page.

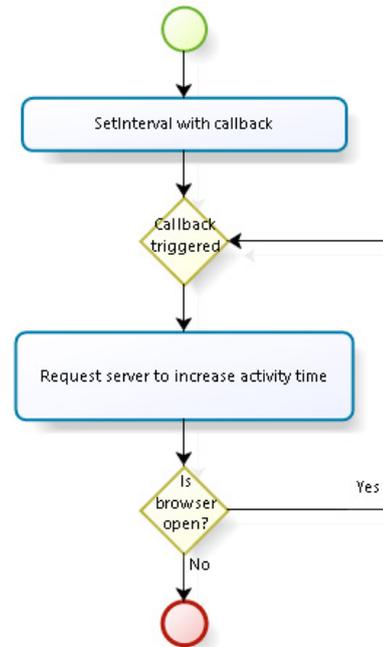


Fig. 1. Browser process: Diagram of the browser part of the plugin.

Figure 2 shows the process triggered in the server for each request so that the server is responsible for storage of the student's access time. Every time the server is requested, the records belonging to the user and the full system access are rescued. Next the minimum required time since the last update is checked, and if it is higher, the past period is added.

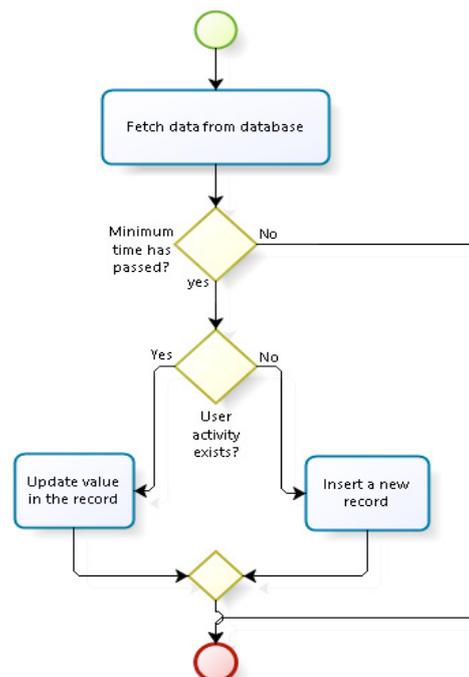


Fig. 2. Server process: Diagram of the server part of the plugin.

Note that this minimum acceptable period between a request and another is not exactly a minute. A shorter period should be defined. Thus, when an update delays, the next call is not below this period. Otherwise, all requests below the period should be ignored.

Limiting the time between calls is necessary to avoid that when a user opens more than once the page, in tabs, for example, the total amount of access time is not multiplied by the number of open pages. Moreover, this condition also prevents an application for false claims, to explain more minutes than the actual access time.

If the server cannot retrieve the user time information due to the lack of prior registration, instead of updating the registry incrementing the old value, a new record is inserted.

When the user closes all instances of Moodle system in the browser, the JavaScript is also interrupted, causing disruption of recurring requests and stop increasing the time.

The solution allows a greater precision than searching activity logs, for the considered period of time can be variable and depends not on the session length. It is also more accurate because of the format proposed to consider the times when the user has the page open without interacting with the system by clicking links, while performing tasks such as reading and even watching an embedded video content.

Thus, by this method, it is possible to obtain a ready to store information relating to the time the student accessed the system without the need for additional processing or requiring prolonged consultations relate many records.

4. RESULTS

The validation of the proposed approach was possible thanks to ENAP, by using a clone of the organization's production database with real data access. For performance testing purposes, it was chosen the course with more user activity logs, offered and concluded in the second half of 2015. Due to the number of records, courses with similar size were prevented from generating any reports that would require knowing the time students remained with the open system. For validation, it was obtained and compared the query time in the database through both solutions.

Table 1 shows a comparison of the database query time to retrieve all the information from all students who have accessed a particular course from the log.

Table 1. Query time comparison selecting all users at once

	Current solution	Proposal
Time (milliseconds)	17.047 ~ 17.799	63 ~ 84
Row count	824044	7059

The current implementation calculates the time between records for the total time the user was active in the system. Thus, this requires that all data belonging to an ongoing

activity to be retrieved from the database for analysis. The number of lines (rows) obtained concerns to the number of activities within the course, including the activities of all users.

The proposed solution, however, adds the user access time while it is still connected to the system, and stores the sum in a single record linked to the user and the course. Thus, to meet the total user time on a course, the system checks the records belonging to him. Table 1 is obtained 7059 records, which is the number of users who accessed the tested course.

Note that the current query implementation and returns all the data associated with user activity. It includes but is not limited to, source access, IP address that caused the access, the exact time of the activity access, the accessed activity. However, it became clear that the implementation did not have any interest in these data, but only the time and the user identifier numbers were used.

In view of this, it was decided to optimize the query specifying which fields the database should return and run the tests again.

Table 2 shows the difference in time between queries. By selecting which fields are retrieved, ignoring unnecessary fields for processing, one obtains a reduction of up to 59 % in total query time, about 7 seconds to the current implementation. However, the same variation does not occur with the proposed solution, given the smaller amount of records needed.

Table 2. Query time comparison selecting specific columns

	Current solution	Proposal
Time selecting all fields (milliseconds)	17.047 ~ 17.799	63 ~ 84
Time selecting specific fields (milliseconds)	10.034 ~ 10.420	62 ~ 80
Row count	824.044	7.059

Another problem that arises to recover all data in a single request relates to the use of the virtual memory of PHP. Since the transformation of the records taken from the result of a query, which is converted into PHP objects, this also causes higher memory consumption by obtained record. Thus, when a query returns many records, given the processing of data, the virtual memory limit available for the process is reached stopping the request [7].

To solve this problem, in the current implementation it was chosen to select records from a single user to query and repeat the query for each user of the course.

Query performed on Table 3 shows a comparison between this method and the current implementation of the proposed solution for a single user. Based on this fact, it was possible

to realize that retrieving data from a single user is feasible in both solutions. However when interacting through each user of a course is necessary, the process may become much slower. The course tested exceeds 7 thousand students, so that, if multiplied by the number of students query time, the entire process takes over 3500 seconds, about 1 hour. The proposed solution, however, is able to perform the procedure for 7 thousand students in approximately 5 minutes.

Table 3. Query time comparison per user

	Current solution	Proposal
Time (milliseconds)	586 ~ 647	16 ~ 36
Row count	16225	1

However, comparing Table 3 and Table 2, it is noted that by using the proposed solution, in a search for each student is not feasible. Moreover, it is not necessary for the proposed solution, since it is possible to recover all data in a single request without causing the total consumption of virtual memory, once the data amount is much smaller.

The proposed solution solves the problem of knowing the time the student spent accessing the system at any given course, as needed at ENAP. The solution proposes a new implementation that allows the generation of information in a viable way.

The solution without additional changes cannot provide more accurate data on the activities that the user accessed or even consult for periods of time within the course. However, it can be adapted to obtain such specific information as needed, or for very specific cases, the user activity log is still available and can be consulted.

5. CONCLUSIONS AND FUTURE WORK

Many current systems collect their data directly from the Moodle activity logs. Such approach allows normalizing the information and making faster searches and improvements of accuracy by storing other desirable information. Since Moodle is a fairly permissive tool for changes and implementation of new features, it is possible to add processes without direct changes in the core.

This work reports the development of a Moodle plugin to count the time the students spend in the system. The goal was to develop a tool that could efficiently perform this task, improving the counting precision and allowing a faster retrieval of data. The acquired results showed that the proposed solution was able to reduce the processing load and at the same time prevent the overhead do the virtual memory management. These results are evidence that the proposed solution fulfills the demanded efficiency requirements, drastically diminishing the queries processing time.

As future work it is possible to change the solution so that it can record information that relates the student to the course and the time spent in performing a specific activity, such as

reading a text, watching a video, answering a questionnaire or joining a forum.

Also, it is possible to optimize the solution to differentiate or just record the time that the user spent interacting with the system, by observing the visibility of the page, mouse movements, keyboard input, or even video player events.

ACKNOWLEDGEMENTS

The authors wish to thank the Brazilian Public Administration School ENAP (Grant TED 083/2016) for their support to this work.

REFERENCES

- [1] Moodle. Community driven, globally supported. Available in: <<https://moodle.org/>> Accessed on 20 April. 2016.
- [2] Mozilla community website. Plugins: what they are and how to install. Available in: <<http://br.mozdev.org/firefox/plugin/>>. Visited at: January 2016.
- [3] Nagi, K., Suesawaluk, P. (2008). Research Analysis of Moodle Reports to Gauge the Level of Interactivity in eLearning Courses at Assumption University, Thailand. In: Proceedings of the International Conference on Computer and Communication engineering 2008, Kuala Lumpur, Malaysia, May 10-2016, pp 772-776.
- [4] BOVO, A. et al. (2013). Clustering moodle data as a tool for profiling students. In: e-Learning and e-Technologies in Education (ICEEE), Second International Conference on. IEEE. p. 121-126.
- [5] HIJÓN-NEIRA, R.; VELÁZQUEZ-ITURBIDE, J. Ángel. (2008). How to improve assessment of learning and performance through interactive visualization. In: Advanced Learning Technologies. ICALT'08. Eighth IEEE International Conference on. IEEE. p. 472-476.
- [6] XMLHttpRequest. World Wide Web Consortium (W3C). Available in: <<https://www.w3.org/TR/XMLHttpRequest/>> Accessed on: 25 May. 2016.
- [7] Moodle. Data Manipulation API. Available in: <https://docs.moodle.org/dev/Data_manipulation_API> Accessed on 07 June. 2016.

A.2 Motivating Attendee's Participation in Distance Learning via an Automatic Messaging Plugin for the Moodle Platform

- Título: Motivating Attendee's Participation in Distance Learning via an Automatic Messaging Plugin for the Moodle Platform
- Veículo: IEEE Frontiers in Education Conference (FIE 2016)
- URL: <<https://doi.org/10.1109/FIE.2016.7757608>>
- Data: 12–15 de Outubro de 2016
- Local: Eire, PA, USA

Motivating Attendee's Participation in Distance Learning via an Automatic Messaging Plugin for the Moodle Platform

Luciano R. de Almeida¹, João Paulo C. L. da Costa^{1,2,3}, Rafael T. de Sousa Júnior¹,
Edison P. de Freitas⁴, Edna D. Canedo⁵, Juliano Pretz¹, Eliakim Zacarias¹ and Giovanni Del Galdo^{2,3}

¹Department of Electrical Engineering, University of Brasília, Brazil

²Institute for Information Technology, Ilmenau University of Technology, Ilmenau, Germany

³Fraunhofer Institute for Integrated Circuits IIS, Erlangen, Germany

⁴Institute of Informatics, Federal University of Rio Grande do Sul, Brazil

⁵Faculty of Gama, University of Brasília, Brazil

luciano.ralm@gmail.com, joaopaulo.dacosta@ene.unb.br, desousa@unb.br, edison.pignaton@inf.ufrgs.br, ednacanedo@unb.br, prettzb@gmail.com, eliakim.zacarias@gmail.com, giovanni.delgaldo@iis.fraunhofer.de

Abstract— Technological advances in information and communications systems are enabling access to education for millions of students, regardless of their geographical location. In this sense, distance learning schools take advantage of technological tools, such as course management systems (CMS) and their plugins. This work proposes a plugin for the e-learning platform Moodle that automatically sends e-mail messages to the students in order to motivate them to take part in the distance learning courses and, consequently, to reduce the dropout rates. This plugin has been tested in courses promoted by the General Coordination of Distance Education (CGEAD) of the Brazilian National School of Public Administration (ENAP). Based on data from its operation within the real ENAP distance learning environment, the proposed plugin is particularly successful to tackle the enrollment locking cases that compose the student dropout rate and also to motivate course enrolled students to access the virtual learning system more frequently.

Keywords— Automatic messaging, Moodle plugin, Distance learning motivating tool.

I. INTRODUCTION

Technological advances in information and communications systems are allowing access to education for millions of people, irrespective of their geographical location. Distance learning (DL) can break down geographical barriers meeting the dissemination of knowledge needs [1]. In addition, DL is becoming more common even as a support tool for standard educational systems. According to [2], DL is already part of the daily lives of thousands undergraduate and graduate students worldwide.

Despite the several advantages that are offered by DL, students motivation is one of the factors that directly affects their success rate [3]. In this context, despite all the available technology, more solutions are necessary to effectively support DL school managers to motivate students' participation and consequently to reduce dropout rates.

Particularly, there is a lack of tools to help DL managers map the reasons for the dropout rates caused by the non-commitment of students and low academic performance. Moreover, there are no computational tools or plugins integrated to free and open-source software learning management systems, such as Moodle [4], in order to analyze the attendee's behavior during the course and to motivate them by sending e-mail messages.

This paper presents a proposal of an automatic messaging plugin for Moodle platform that has been developed based on the needs of the direction of the General Coordination of Distance Education (CGEAD) of the National School of Public Administration (ENAP). The proposed plugin improves the interaction between course managers and students in order to reduce the dropout rates by motivating students to stay connected to the courses. By evaluating the dropout rates in 2015 at Enap, it is shown the effectiveness of the proposed plugin by reducing dropout rates and increasing students' motivation in accessing the environment education.

This paper is divided into four more sections, besides this introduction. In Section II, a theoretical background and related works showing the importance of the topic are presented. In Section III, the proposed automatic messaging plugin for Moodle platform is detailed. The results obtained after using the proposed plugin are shown in Section IV. In Section V, conclusions are drawn and directions for future work are provided.

II. STATE OF THE ART

Among the main DL educational platforms, Moodle [4] is an open source system, collaboratively developed by the virtual community that supports the expansion of a single teaching ambient for students, administrators and teachers, being a safe, robust and integrated system. Moodle is composed of modules or extensions called plugins. Plugins are smaller software systems that are incorporated into larger ones to play specialized functions [5].

In [6], it is highlighted the need to bring new ways of communication and interaction between students and teachers. In this sense, the technological center of Costa Rica developed its own educational platform for the purpose of sending, receiving and viewing SMS messages looking for a better interaction between teachers and students [6]. The system in [6] also allows students and teachers to run commands on the platform through text messages.

The work presented in [7] discusses a chat system, in which students can discuss the content of the courses and ask questions to their teachers and other students. The mechanism proposed by the authors does not require that everyone is online and there is a system of recommendations for the user to facilitate their search for useful content. Note that there is an important difference between the proposal presented in [7] and the presented in this current work, which is the fact that in the proposed plugin the message exchange is automatic.

Paper [8] suggests a delivery architecture that learning management content using the email system manager and its architecture for regions with low communication throughput.

Paper [9] proposes a life cycle model for DL, which facilitates interaction of students within DL, using a DL Lifecycle Management (DLLM) model that is intended to cover any organization in which there is a need to manage and run the DL. With the model in [9], the authors achieve a higher quality of education distributed to the students and an increase in attendee's feedback.

The work in [10] addresses the attendee as an active role in the creation and sharing of resources in DL. In order to show this usage, an example to exchange information and messages between students and teachers using Moodle 2.0 applied to Google Drive resources is presented. The general opinion of the students and teaching reputation is taken into account.

III. THE PROPOSED AUTOMATIC MESSAGING PLUGIN FOR MOODLE PLATFORM

The proposed automatic message plugin aims to reduce the dropout rate and to motivate the students to stay connected to the courses. The plugin enables the DL managers to send personalized messages to the students whenever they reach certain predefined conditions.

Figure 1 presents the operation flow of the plugin and the necessary information to send a message to an attendee. In step 1 of Figure 1, the system goes through all the courses with classes in progress. In step 2 of Figure 1, the system checks the messages that are registered for a given day of a course. In step 3 of Figure 1, the system compares the conditions of each attendee in relation to the corresponding day class of the course. In step 4 of Figure 1, the system checks if the condition is registered in a message to be sent, while in step 5 of Figure 1 the system forwards the message to the messaging server. In step 6 Figure 1, the message is sent to the email of the registered attendee.

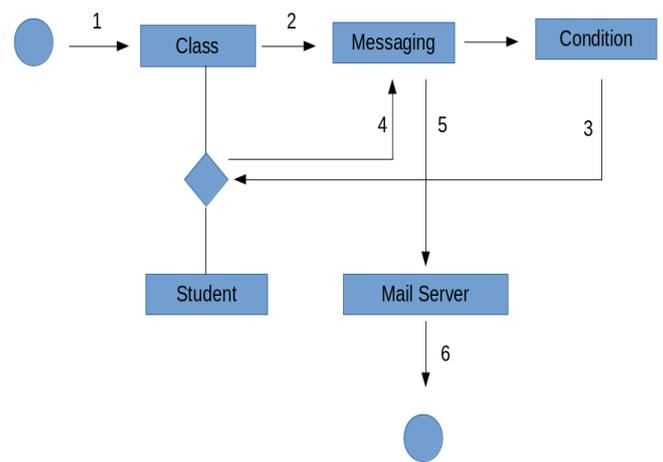


Figure 1: Overall message sending flow of the proposed automatic message plugin

In Figure 2, the block diagram presenting the information flow of the user interaction processes is depicted. Note that the main stream and the three alternative streams complement each other and have the same goal.

The main flow of the system interface is a click action "New Message / Edit / Delete Message" and then the system records in the archive the messages that should be sent to students in a specific course. In the first alternative flow A1, the user fills the data for the message including sending conditions, message date, subject and message body and confirms its completion. Then the data is recorded on the database and the system displays a confirmation message, and exits the alternative flow A1. In the alternative flow A2, the user changes the data and confirms their completion, and records the changes on the database. Then, the system displays a message confirmation and closes the alternative flow A2. In the alternative flow A3, the system displays a question, updates the message and database, and displays the confirmation message, finishing the alternative flow 3. Note that the This diagram in Figure 2 presents no flow exception.

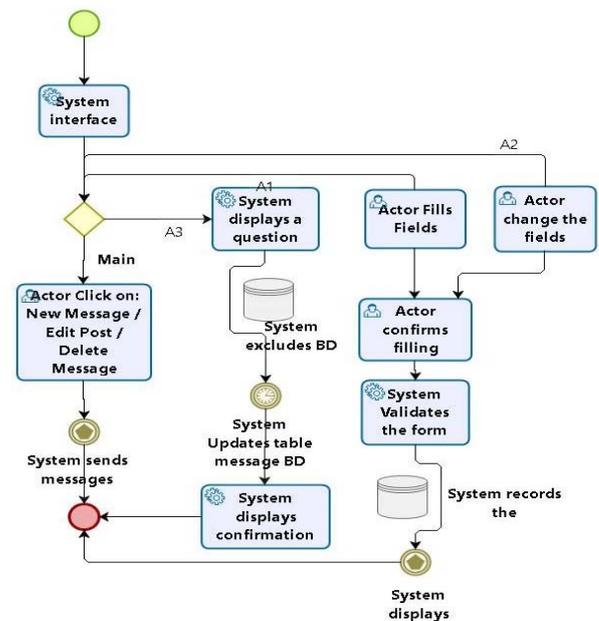


Figure 2. User interaction processes with the proposed automatic message plugin

Figure 3 shows nine types of predefined conditions that are considered in the proposed plugin. The first condition is the case of user without access into the e-learning system. The second condition is when the user already accessed and has no grade on an evaluation activity since the course started. The third condition is when the user has no grade past seven days before a message sending date, although has in practice concluded an activity. The fourth condition is when the user obtained grades below the minimum score, but he/she can still conclude the course by improving his/her grades on the next activities. The fifth condition is when the user did not make any evaluation activity. The sixth condition is when the user fails all evaluation activities. The seventh condition is when the user has successfully passed the activities. The eighth condition is when the user abandons the course. Finally, the ninth condition includes all students of all conditions. Besides the selection of groups of students by conditions, the students can of course be selected based on other educational criteria that can be specific to a certain course.

Figure 3: Wireframe representing the selection field with the conditions for sending messaging to students.

Instead of selecting the students by considering some specific condition, the DL manager can also select the students by the course and then observe the history of sent messages. Figure 4 presents an example of a frame with the history of sent messages.

Specific templates of messages are automatically loaded in order to facilitate the work of the DL manager. For instance, in Figure 5, the proposed plugin displays a template of message for the second condition in Figure 3. Note that the message can be adjusted if necessary to mention some specific aspect related to the particular attendee. As depicted in Fig. 4, once the condition and the time period are chosen, the template message appears with filled fields of the subject and the text.

In Figure 6, another example of template message is shown. This example addresses the situation in which the attendee is below the minimum grade, but can still pass the course. Note that the message warns that the course is ending with motivational words. Stating that despite not having a

minimum passing grade, the student with a little more work and effort will be able to complete the task and reach the note.

▼ Situation	▼ Message	▼ Day of course	▼ Sending History	▼	▼
Dropout user	Text messages that you receive	5º	Wednesday 11 dec 2015 - 11:52	Edit	Delete
User already accessed and has no note on evaluation activity	Text messages that you receive	10º	Wednesday 17 dec 2015 - 11:52	Edit	Delete
User that is below the minimum mark and can still get it	Text messages that you receive	14º	Wednesday 25 dec 2015 - 11:52	Edit	Delete
Approved User	Text messages that you receive	22º	Wednesday 25 dec 2015 - 11:52	Edit	Delete

Figure 4: Wireframe the system displays lists the messages registered in determine course.

Figure 5: Registration screen wireframe of messaging received in the mail

Figure 6. Example of messages received by the attendee at the end of the course

IV. RESULTS

In order to evaluate the effectiveness of the proposed plugin, we compare the performance of the students of our e-learning system before and after the plugin. In order to analyze the students' performance, we consider the amount of students who locked their enrollment in the years 2011 and 2012 when it was not yet applied the plugin for sending messages. On the other hand, the students were motivated by the automatic messaging plugging in 2013, 2014 and 2015.

Table 1 presents the amount of enrolled students, the amount of students who failed and the amount of enrollment rates locked per year. Analyzing the results of using the proposed plugin in the ENAP virtual learning environment, the enrollment locking rate has been diminished during the last four years. Indeed, in 2011 and 2012, before the plugin has been designed, the locking rate was of 11 % and 16 %, while after the incorporation of the automatic message plugin, the locking rate reduced to 9 %, 8 % and 7 % in 2013, 2014 and 2015, respectively.

Table 1 – Percentage of dropouts at CGEAD per year.

Year	Total of enrolled students	Students that failed	locked enrollment students'	locked enrollment / failed (%)	Plugin
2011	34,156	9,513	1,046	11 %	NO
2012	28,664	8,244	1,319	16 %	NO
2013	42,078	14,788	1,331	9 %	Manually
2014	60,927	33,666	2,693	8 %	Manually
2015	148,819	74,183	4,958	7 %	Automatic

In order to complete the analysis, we assume the same time period, but we group the students into two categories. In the first one, we consider the students that did not receive messages in 2015, while in the second one, we consider the students that received messages. The students of the first group are divided into five classes as shown in Table 2. Note that the amount of students a day was 23.11 % for the courses without automatic messages. However, as shown in Table 3, for the courses with automatic messages, the percentage of students a day is 37.1 % of the total enrolled in the course. Therefore, the gain of the proposed automatic message plugin is quite significant.

Table 2 – Total and percentage of hits on days without automatic messages.

Classes	Students assets	Average hits per day	Students who access per day %
Class 1/2015	1,085	269	24.8%
Class 2/2015	1,208	289	23.9%
Class 3/2015	1,050	258	24.6%
Class 4/2015	1,371	279	20.4%
Class 5/2015	985	216	21.9%
Total	5,699	262.2	23.11%

Table 3 – Total and Percentage of hits on days with automatic messages sent by the plugin.

Classes	Students assets	Average hits per day	Students who access per day %
Class 4/2015	1,371	349	35.5 %
Class 5/2015	985	532	38.8 %
Total	2,356	440.5	37.1%

V. CONCLUSIONS AND FUTURE WORK

In the above mentioned context, the observation of attendee behavior through online learning tools allows the visualization that simple stimuli in communication can help connecting the students to the virtual school environment, improving their success in the followed courses. The work presented in this paper describes a plugin for Moodle e-learning environment that automatically sends messages to students according to their assiduity in accomplishing the courses' tasks. The results acquired after the usage of the proposed plugin demonstrated a significant decrease in the drop-off and the failure rates compared to these rates before the usage of the plugin. The study also revealed that the days in which the students received the messages, their engagement in accessing the virtual learning environment was higher than in the days in which they did not receive messages. This evidences the usefulness of the proposed plugin in engaging the students in the enrolled courses.

Future work involves better integration of the proposed plugin with analysis tools in order to increase the automation and control.

ACKNOWLEDGMENT

The authors wish to thank the National School of Public Management – ENAP (Cooperation Agreement 30/2014) and the Secretariat of Budget – SOF (Cooperation Agreement 30/2014), of the Brazilian Ministry of Planning, Budget and Management, for their support to this work.

REFERENCES

- [1] DALMAU, Marcos Baptista Lopez; Introduction to distance education. Available at: <http://www.moodle.fmb.unesp.br/file.php?file=%2F175%2Farquivos%2FMaterial_de_estudo%2FIntroducao_EAD_Unid1_Unid4.pdf>. Accessed on 01/20/2016.
- [2] MOORE, M.; KEARSLEY, G. Distance: an integrated view. Publisher Cengage Learning, 2013. Available at: <<https://books.google.com.br/books?id=dU8KAAAAQBAJ>>. Accessed on 01/22/2016.
- [3] Bekele, TA (2010). Motivation and satisfaction in learning environments supported on the Internet: A Review of Educational Technology & Society, 13 (2), 116-127.
- [4] Moodle. Community driven, globally supported. Available in: <<https://moodle.org/>> Accessed on: 04/20/2016.
- [5] Mozilla community website. Plugins: what they are and how to install. Available in: <<http://br.mozdev.org/firefox/plugin>>. Visited at: 15/01/2016.
- [6] Céspedes, Jacqueline Solís; Rivas, Mario Chacón; Towards an implementation of a short messaging system over an e-learning environment, Information Systems and Technologies (CISTI), 2012 7th Iberian Conference on, Madrid, 2012, pp. 1-6.
- [7] Wang, Ying-Hong. An intelligent semantic agent for e-learning message communication. 19th International Conference on Advanced Information Networking and Applications (AINA'05) Volume 1 (AINA papers) Year: 2005, Volume: 2 Pages: 105 - 108 vol.2, DOI: 10.1109/AINA.2005.110
- [8] Ijtihadie, Royyana. E-mail based updates delivery in unidirectional content synchronization among learning management systems over limited bandwidth environment. TENCON 2011 - 2011 IEEE Region 10 Conference, Bali, 2011, pp. 211-215.
- [9] Ferdiana, Ridi. Distance Learning Lifecycle Management: An Agile Approach to Manage the Lifecycle of a Distance Learning. Information Technology and Electrical Engineering (ICITEE), 2013 International Conference on, Yogyakarta, 2013, pp. 106-112.
- [10] Babić, Snježana; Krešić, Mario; Kucel, Karla. E-education 2.0: Students' Digital Identity and Online Learning Activities. Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014 37th International Convention on, Opatija, 2014, pp. 756-761.