

*Um ambiente de aprendizagem para o
acompanhamento individual dos alunos*

Rafael Santos

Outubro / 2015

Dissertação de Mestrado em Ciência da
Computação

Um ambiente de aprendizagem para o acompanhamento individual dos alunos

Esse documento corresponde a dissertação apresentada à Banca Examinadora para defesa no curso de Mestrado em Ciência da Computação da Faculdade Campo Limpo Paulista.

Campo Limpo Paulista, 09 de outubro de 2015.

Rafael Santos

Prof. Dr. Marcelo de Paiva Guimarães
(Orientador)

Resumo: O processo de ensino-aprendizagem é constituído por diversos aspectos que são influenciados por variáveis como a quantidade de alunos e aulas por disciplina; o nível de complexidade de cada exercício; a dificuldade de abordagem e aprendizagem do conteúdo; e as ações realizadas pelo professor quando as dificuldades dos alunos são observadas. O presente projeto apresenta uma estratégia de acompanhamento individual dos alunos integrada a uma ferramenta computacional (*software*). Hipótese: o acompanhamento individual dos alunos atrelado a ações condizentes, aproxima o professor do aluno, e, conseqüentemente, potencializa o desenvolvimento de processos de ensino-aprendizagem mais eficientes. Objetivos: 1. desenvolver uma estratégia que permita o acompanhamento individual dos alunos em sala de aula. 2. criar um *software* baseado em objetos de aprendizagem interativos. 3. validar a estratégia e o *software* em estudo de casos. Método: na primeira etapa criou-se uma ferramenta computacional e objetos interativos capazes de monitorar as atividades realizadas pelos alunos, oferecendo ajuda e alertando o professor; ainda na primeira etapa, o ambiente foi testado com dois pequenos grupos de alunos. Na etapa seguinte a estratégia e a ferramenta foram testadas com grupos maiores de alunos. Resultado: os resultados indicaram que a estratégia implementada pela ferramenta é uma solução promissora para auxiliar o acompanhamento e, conseqüentemente, o disparo de ações apropriadas no momento em que as dúvidas dos alunos acontecem.

Abstract: *The teaching-learning process consists of several aspects that are influenced by variables such as the amount of pupils and classes by discipline; the level of complexity and difficulty of approach and content learning; and the actions taken by the teacher when the students' difficulties are observed. The present research project innovates by presenting a strategy for individual monitoring of students integrated into a computational tool (software). Hypothesis: the individual follow-up of students coupled to actions consistent, approach the student's teacher, and, consequently, enhances the development of teaching-learning processes more efficient. Objectives: 1. develop an educational strategy that allows the tracking of individual students in class. 2. create a software based on interactive learning objects. 3. validate the strategy and the software in case studies. Method: in the first step entails the creation of computational tool and interactive objects able to monitor the activities carried out by the students, offering help and alerting the teacher; still in the first stage, the objects will be specified formally, based on learning objects already known in the literature, validating and checking the tests presented. In the next step we will validate the performance, reports and feedbacks for better evaluation of the tool.*

SUMÁRIO

LISTA DE FIGURAS.....	iii
LISTA DE TABELAS.....	v
CAPÍTULO I – INTRODUÇÃO.....	1
1.1 – O Objetivo.....	2
1.2 – A Motivação.....	3
1.3 – A Apresentação do trabalho.....	4
CAPÍTULO 2 – METODOLOGIA.....	5
CAPÍTULO 3 – SISTEMAS DE APRENDIZAGEM.....	7
3.1 – Introdução.....	7
3.1.1 – Sistema de Gerenciamento de Conteúdo.....	8
3.1.2 – Sistema de Gestão de Aprendizagem.....	9
3.1.3 – Sistemas de Gestão de Conteúdos de Aprendizagem.....	10
3.3 – Considerações do capítulo.....	13
CAPÍTULO 4 – STUDENTWATCHER: UMA FERRAMENTA DE ACOMPANHAMENTO INDIVIDUAL DE ALUNOS.....	14
4.1 - Introdução.....	14
4.2 – Arquitetura do StudentWatcher.....	19
4.2.1. Tecnologias Web.....	21
4.2.2 Objetos de Aprendizagem Interativos.....	22
4.2.2.1 Ontologia do OAI.....	25
4.2.2.2 Lógica Fuzzy.....	29
4.2.3. Interface Web do estudante.....	32
4.2.3. Interface Web do professor.....	34
4.2.4 Processo de ensino-aprendizagem do StudentWatcher.....	38

4.5 – Considerações do capítulo	39
CAPÍTULO 5 – RESULTADOS	40
5.1 Primeira Fase	40
5.1.2 Introdução ao Hardware e a Computação	42
5.1.3 Discussão	43
5.2 Segunda Fase	44
5.2.1 Discussão	48
5.3 Considerações do capítulo.	50
CAPÍTULO 6 – CONCLUSÃO	51
6.1 Resultados.....	51
6.2 Trabalhos Futuros.....	52
REFERÊNCIAS	53

LISTA DE FIGURAS

Figura 1. Visão Geral do CMS Joomla 1.0.	9
Figura 2. Visão Geral do Moodle 2.0.	10
Figura 3. Análise da Instância do aluno.....	15
Figura 4. O aluno fica um período sem interagir com a instância.....	16
Figura 5. Ajuda oferecida automaticamente pelo sistema.	16
Figura 6. Esclarecimento coletivo do professor aos alunos.....	17
Figura 7. O professor pode compartilhar o mesmo exercícios para vários alunos.	18
Figura 8. Compartilhamento de Instância em alunos.....	18
Figura 9. Arquitetura do StudentWatcher.	20
Figura 10. Diagrama de caso de uso.....	20
Figura 11. Diagrama de classes.	21
Figura 12. Arquitetura e Tecnologia Web.	22
Figura 13. Operação de um Objeto de Aprendizagem Interativo.....	24
Figura 14. Classe dos Objetos de Aprendizagem Interativos.	25
Figura 15. Classes da Ontologia OWL OAI.	26
Figura 16. Especificação da Classe Metadata (metadados).....	27
Figura 17. Especificações dos objetos e atributos.....	27
Figura 18. Trecho do OAI mapeado na Ontologia OAI.	28
Figura 19. Transposição do padrão da Ontologia Computacional para o StudentWatcher.....	29
Figura 20. Variáveis Fuzzy de Entrada.....	31
Figura 21. Variáveis Fuzzy de Saída.....	31

Figura 22. Diagrama de classes fuzzy	32
Figura 23. StudentWatcher - Interface Principal – Aluno.....	33
Figura 24. StudentWatcher - Interface do aluno	33
Figura 25. Classe aluno do StudentWatcher	34
Figura 26. Painel de acompanhamento do professor.	35
Figura 27. Diagrama de estado das cores (estado de aprendizagem). 36	
Figura 28. Calendário para controle de tempo dos OAIs.....	36
Figura 29. Levantamento estatístico de desempenho.	37
Figura 30. Diagrama Professor do StudentWatcher	37
Figura 31. Processo de ensino-aprendizagem do StudentWatcher.....	38
Figura 32. Comparação da Classe A e B.	44
Figura 33. Interação com o professor durante a atividade.....	45
Figura 34. Alunos confortáveis com o fato do acompanhamento do professor.....	45
Figura 35. Opinião alunos ferramentas sobre o ambiente StudentWatcher.....	46
Figura 36. Opinião dos alunos em relação ao seu desempenho.	47
Figura 37. Quantidade alunos por categoria de médias	48

LISTA DE TABELAS

Tabela 1. Características de CMS, LMS e LCMS.....	11
Tabela 2. Questões de Avaliação do OAI.....	41
Tabela 3. Questões Qualitativas de Avaliação do OAI.	41
Tabela 4. Questões de Avaliação do OAI.....	42
Tabela 5. Questões Qualitativas de Avaliação do OAI.	43
Tabela 6. Pontos forte e fracos da ferramenta	48

CAPÍTULO I – INTRODUÇÃO

Aprender é uma atividade fundamental e desafiadora, cujo sucesso é alcançado quando os alunos se apropriam do conhecimento e o aplicam em contextos diferentes dos abordados durante o processo de aprendizagem, envolvendo-se diretamente com as informações recebidas (Svinick et al. 2012). Alguns alunos têm maior dificuldade do que outros durante a aprendizagem (Freire 2002, Zabala 1998), então a persistência de todos os envolvidos na criação de situações que apoiem o domínio de uma prática ou a compreensão de um assunto é um fator primordial para que a aprendizagem seja alcançada (Wells 1999). Esta visão está apoiada no conceito de zona de desenvolvimento próximo que segundo Vygotsky, exemplifica que a distância entre o nível de desenvolvimento atual determinado pela resolução independente de problemas e o nível de desenvolvimento potencial determinado pela resolução de problemas sob orientação ou em colaboração com parceiros mais capazes (Vygotsky 1987, Vygotsky 1998).

Geralmente, os alunos que participam ativamente das aulas (expondo dúvidas, buscando resolver os exercícios propostos, não se ausentando e não desistindo), atingem em algum momento o nível de aprendizagem visado. Isto é uma das inferências enfatizadas por autores que discutem a zona de desenvolvimento próximo do aluno, que permite uma aprendizagem mais fácil e melhor (LeFrancois 2001). Integrado a este aspecto, destaca-se o pressuposto da assistência, que enfatiza a importância da presença e assistência do professor, independentemente da proximidade física (ensino presencial ou à distância). Devido à disparidade de tempo de aprendizagem dos alunos, é muito importante que o professor esteja atento a cada aluno, esclareça todas as dúvidas e busque adequações no tratamento do conteúdo para aproximá-lo das possibilidades e potencialidades de aprendizagem (Vygotsky 1987).

Existem várias atitudes que os professores podem realizar a fim de potencializar a aprendizagem, como, por exemplo, a coleta individual das dúvidas e compartilhamento delas com todos os alunos, pois a maioria dos questionamentos é comum; a análise da compreensão dos alunos por intermédio de questionamento sobre os assuntos durante a explicação. Atitudes como essas permitem que o professor

conheça o desempenho atual de aprendizagem dos alunos naquele momento, e conseqüentemente, possibilita que o mesmo decida, por exemplo, a mudar a estratégia de ensino adotada (Vygotsky 1978).

1.1 – Objetivo

Este trabalho apresenta como objetivo uma estratégia de acompanhamento dos alunos, implementada em uma ferramenta computacional em sua segunda versão denominada StudentWatcher (Santos et al. 2014), que auxilia o acompanhamento individual dos alunos através de ações interativas, favorecendo o desenvolvimento de situações de aprendizagem que possam viabilizar a assistência educacional do aluno. Esta assistência parte da observação das ações realizadas pelo estudante que predizem a necessidade de intervenção no processo de aprendizagem. Esta intervenção pode acontecer, por exemplo, através da sugestão de leitura de um texto ou material de apoio; de perguntas-guia ou introdução de elementos da solução da tarefa (Vygotsky 1987). Assim, esse ambiente atua diretamente sobre as atividades do aluno, monitorando o atual desempenho de aprendizagem do aluno e alertando o professor; além disso, realiza ações automáticas, como, por exemplo, oferece matérias de leitura para o aluno. A hipótese base desse trabalho é que melhores resultados são alcançados com alunos sempre que o apoio individualizado é ofertado, mesmo em salas com vários alunos (Bloom 1984).

O StudentWatcher coleta, interpreta e entrega informações sobre o comportamento do aluno para o professor. As informações analisadas pelo StudentWatcher permite ao professor estar mais próximos de seu alunos. Durante a utilização do sistema em sala de aula, esta abordagem permite que o professor atue no momento que está trabalhando e o estudante experimenta uma dificuldade no exercício.

O StudentWatcher tem como base um módulo que utiliza Lógica Fuzzy Intervalar (Zadeh 1965, Bellman 1970, Ross 2010) para disparar as ações do ambiente conforme o comportamento do aluno e o estado dos Objetos de Aprendizagem Interativos (OAI) (Nogueira et al. 2014). OAI são um conjunto de itens criados com propósitos educacionais bem definidos, e com as seguintes características: reusável (pode reutilizado novamente), interoperável (pode ser suportado por qualquer sistema),

acessível (pode ser facilmente armazenado e obtido), gerenciável (pode ser controlado e utilizado) e interativo (pode fornecer o desempenho dele e automaticamente gerar ações). Assim, os OAI são, por exemplo, exercícios e explicações - capazes de monitorar as atividades realizadas pelos alunos, oferecendo ajuda e alertando o professor de acordo com as dificuldades específicas do aluno. O OAI apresentado neste trabalho, é uma extensão dos Objetos de Aprendizagem tradicionais, no caso foi utilizado como referência o padrão SCORM (SCORM 2009). Com a utilização do ambiente proposto, os OAI são criados, gerenciados, distribuídos e reutilizados no StudentWatcher. O professor cria os OAI e os distribui entre os alunos. Então, o StudentWatcher monitora as ações realizadas nos objetos individuais, o que permite relatar ao professor, o desempenho atual e, além disso, fornece auxílio a cada aluno, como, por exemplo, pode sugerir um vídeo explicativo. Não faz parte deste trabalho adaptar o conteúdo da disciplina, realizando ações como, por exemplo, propor um exercício de reforço para o aluno de maneira automática. Como o professor acompanha o momento atual da aprendizagem de cada aluno, cabe a ele este tipo de ação na configuração dos objetos. O StudentWatcher utiliza configurações prévias de cada OAI e compara com o desempenho atual do objeto para oferecer ajuda aos alunos e notificar o professor.

1.2 – A Motivação

Acompanhar os alunos individualmente é um desafio para os professores, pois tal ação é influenciada por variáveis como a quantidade de alunos da turma e as diferenças no tempo de aprendizagem de cada aluno (Freire 2002, Zabala 1998). As ferramentas tecnológicas são instrumentos que podem auxiliar o acompanhamento e facilitar as orientações dos alunos, oferecendo soluções como: análise das atividades em execução por cada aluno de maneira síncrona e/ou assíncrona; difusão de esclarecimento de dúvidas coletivas e individuais; integração de espaços coletivos e individuais de produção; e oferecimento de auxílios automáticos a cada aluno conforme o estágio de aprendizagem. Vários sistemas com estratégias educacionais já foram propostos (VanLehn 2006, Yang 2010, Chakraborty 2010, Chen 2007, Huang 2009, Chen 2009, Yang 2007 e Chen 2008), que são aplicativos de computador para o auxílio do aprendizado. Contudo, a maioria deles não visa fornecer detalhes sobre o

comportamento da aprendizagem dos alunos aos professores, eles adaptam o conteúdo do material ao aluno.

1.3 – A Apresentação do trabalho

O texto deste trabalho está organizado em função dos pontos principais que foram desenvolvidos no andamento do projeto.

Assim o capítulo 2 mostra a metodologia utilizada para testar a estratégia e o StudentWatcher 3 traz definições, características e comparações de diferentes sistemas de aprendizagem; o capítulo 4 apresenta a estratégia de acompanhamento individual de ensino, o projeto funcional e arquitetural do StudentWatcher, além disso, a implementação; o capítulo 5 traz os resultados e suas discussões; e, por fim, o capítulo 6 apresenta as conclusões.

CAPÍTULO 2 – METODOLOGIA

Para atingir o objetivo principal do projeto, entende-se a necessidade de contextualização e definição dos sistemas de aprendizagem, criação e gerenciamento de conteúdos e o uso dessas ferramentas computacionais para o apoio das ações educacionais. Assim efetuamos uma análise e revisão bibliográficas dos sistemas de gerenciamento e criação de conteúdo e gestão de aprendizagem.

Deste modo, foi comparado outros sistemas de tutoria correlatos, assim poderíamos verificar as tendências sobre as ideias e objetivos propostos neste projeto. Definido as funcionalidades da ferramenta, sua utilização e ações para o acompanhamento individual dos alunos.

A Arquitetura baseada em tecnologias web para construção e desenvolvimento utiliza recentes avanços como JAVA e HTML5 permitindo aplicações mais sofisticadas. Definindo assim o StudentWatcher, foi projetado de forma a garantir a personalização e interação dos Objetos de Aprendizagem Interativos (OAI), monitorados por um módulo de tomada de decisão implementado utilizando Lógica Fuzzy.

O StudentWatcher foi avaliado em duas fases. A primeira, envolveu a avaliação do ambiente com as funcionalidades básicas implementadas e com turmas pequenas. Nela, o ambiente foi validado por duas disciplinas/turmas: “Introdução a Segurança da Informação” e “Introdução ao Hardware e Computação”. Ambas as disciplinas ocorreram em cursos e instituições distintas. Na primeira disciplina (Turma A) haviam 5 alunos matriculados, enquanto na segunda 20 (Turma B). Neste cenário foi possível avaliar a diferença no acompanhamento em dois cenários controlados e com características diferentes.

Para a preparação desses testes, ambos os professores e os alunos foram cadastrados no StudentWatcher pelo administrador. Os professores receberam orientações acerca da ferramenta, navegação e construção dos objetos no ambiente. Antes do início das atividades com os alunos, os professores apresentaram o StudentWatcher, ilustrando a visão de acompanhamento que eles teriam das atividades a serem realizadas, e da mesma forma mostrou-se as opções de interação disponíveis para os alunos.

Os professores tiveram a liberdade para criar seus OAIs com as características necessárias para atender o objetivo educacional da aula em questão. Na turma A o professor criou um OAI no formato de questionário com 5 questões. Na turma B foi criado um questionário com 6 questões. Cada objeto contou com no mínimo 3 dicas, que estavam à disposição dos alunos para consulta e pré-configuradas para disponibilização automática, conforme dados de entrada do professor - disparadas pelo módulo de tomada de decisões. Ao final do curso, as duas turmas foram submetidas a um questionário com intuito de mapear a utilização, navegabilidade e interação proporcionada pelo ambiente.

O questionário aplicado aos alunos foi composto de questões que envolvem os aspectos como a utilização e uso do ambiente, interação e acompanhamento do professor com a disponibilização de dicas e auxílios. As análises dos resultados foram realizadas por turma, considerando ao final uma discussão e comparação sobre os resultados encontrados em ambas.

A segunda fase de avaliação contou com 111 alunos e 5 professores (5). Além disso, os problemas detectados na primeira versão da ferramenta foram corrigidos e adicionadas novas funcionalidades, como, por exemplo, o levantamento estatístico e o painel de controle cronológico das atividades.

CAPÍTULO 3 – SISTEMAS DE APRENDIZAGEM

Neste capítulo, serão apresentados sistemas de aprendizagem projetados para facilitar a condução da aprendizagem. Serão apresentados os conceitos básicos; exemplos de ferramentas e um comparativo entre esses sistemas de aprendizagem. Por fim, apresenta-se uma discussão de sistemas de aprendizagem que buscam avaliar os cursos, e o desempenho e envolvimento dos alunos.

3.1 – Introdução

Um sistema de aprendizagem é um software educacional que coloca o estudante ou o aprendiz como centro do processo de ensino-aprendizagem. O objetivo dele é desenvolver a formação e aprendizado do aluno, o controle das atividades e o aumento da capacidade de aprender, tornando-se então ferramentas imprescindíveis para o processo de ensino-aprendizagem (Bassanezi 2002). Estes sistemas, abrangem funcionalidades de armazenamento, distribuição e gerenciamento de conteúdo de aprendizagem, de forma interativa e gradativa. Deste modo, eles registram e apresentam as atividades do aluno, bem como seu desempenho, além da emissão de relatórios, propiciando o aperfeiçoamento do processo de ensino-aprendizagem. Algumas das características básicas deste tipo de sistema são: recursos interativos, controle das atividades e monitoração, e compatibilidade com as especificações existentes de conteúdo - característica fundamental na transferência de conteúdos entre plataformas.

Segundo (Passos 2006), os sistemas de aprendizagem têm trazido vantagens às práticas educativas ao processo de ensino-aprendizagem, tais como:

- Redução de custos;
- Rápida distribuição e alteração dos conteúdos;
- Permite ao aprendiz fazer seu próprio percurso;
- Disponibilização de recursos interativos, tais como e-mail, fórum, sala de discussão, e vídeo conferência para sistematizar as intervenções;
- Disponibilidade a qualquer hora e local.

A evolução da tecnologia web viabilizou processos de colaboração em tempo real, possibilitando que o aprendiz construa seu conhecimento através da interação com o professor e com outros participantes (Pádua 2008). No entanto, várias limitações nos atuais sistemas de aprendizagem ainda precisam ser resolvidas. Por exemplo, para mediar à interação e colaboração entre tutores e aprendizes, os sistemas atuais utilizam apenas a troca de objetos de maneira estáticas e mensagens instantâneas ou assíncronas (fóruns, chats, e-mail). Além disso, diversos estudos identificam problemas relacionados com: limites da percepção, falhas de usabilidade, dificuldades de navegação e de interação assíncrona, limitações no suporte ao trabalho do professor, falta de interatividade síncrona integrada, formas limitadas de suporte à avaliação continuada (Ramos 2006) e pouca flexibilidade na manipulação de objetos de aprendizagem (Passos 2006). Diante destes desafios, as próximas subseções apresentam os principais tipos de sistemas de aprendizagem utilizados atualmente.

3.1.1 – Sistema de Gerenciamento de Conteúdo

Um Sistema de Gerenciamento de Conteúdo (CMS - *Content Management System*) é um aplicativo ou sistema usado para criar, editar, gerenciar e publicar conteúdo de uma maneira organizada, que permite que o mesmo seja modificado, removido e adicionado com simplicidade. Os CMSs são freqüentemente usados para armazenar, controlar e publicar conteúdo, como, por exemplo, notícias, artigos e manuais. Um CMS pode envolver arquivos, imagens, áudios, vídeos, documentos eletrônicos e conteúdo Web (Schwingel 2009).

A principal vantagem deles é a facilidade de criação de conteúdo, rapidez e eficiência de gerenciamento. Estes sistemas, como o Joomla (North 2012), oferecem uma estrutura web pré-programada e com recursos básicos, com fácil manutenção e administração. Como exemplo a (Figura 1) mostra a interface do Joomla (versão 1.0). Nessa interface o administrador pode efetuar a criação, formatação e exclusão dos objetos.

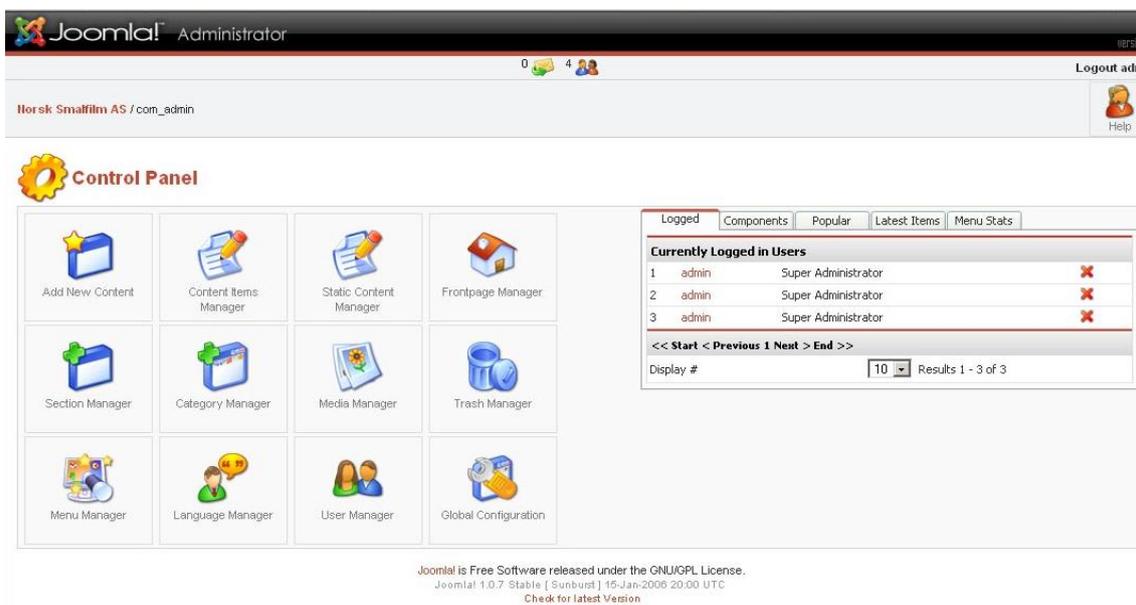


Figura 1. Visão Geral do CMS Joomla 1.0.

Conforme (Schwingel 2009) algumas desvantagens do uso de CMS são:

- Dificuldade de alterar o design do conteúdo;
- Necessidade de utilização de uma linguagem de programação web para alteração de conteúdo;
- Precisa de um banco de dados que suporte uma linguagem web;
- Falta de flexibilidade na disposição dos elementos visuais;
- Limitado aos templates disponíveis.

3.1.2 – Sistema de Gestão de Aprendizagem

Os Sistemas de Gestão de Aprendizagem (LMS- *Learning Management System*) provê um conjunto de funcionalidades para o desenho de cursos ou unidades curriculares com componente online. Além disso, esses sistemas oferecem ferramentas que possibilitam a interação entre os participantes nos cursos - estudantes e professores - através de ferramentas de comunicação síncrona e assíncrona. O conteúdo gerenciado por esses ambientes podem ser, por exemplo, textos, vídeos, áudios. Esses ambientes, também gerenciam todos os usuários, provendo, por exemplo, relatórios de acesso e permitem que os mesmos conteúdos sejam reutilizados em diversos cursos. Geralmente,

eles suportam objetos de aprendizagem, como o SCORM (SCORM 2009), o que assegura a sua reutilização e a sua interoperabilidade entre diversas plataformas tecnológicas, bem como permitem a realização de atividades de avaliação de aprendizagem. Como exemplo a (Figura 2) mostra a interface do Moodle 2011, versão 2.0. Nessa interface é possível verificar a estrutura de criação de objetos, sua organização e controle.

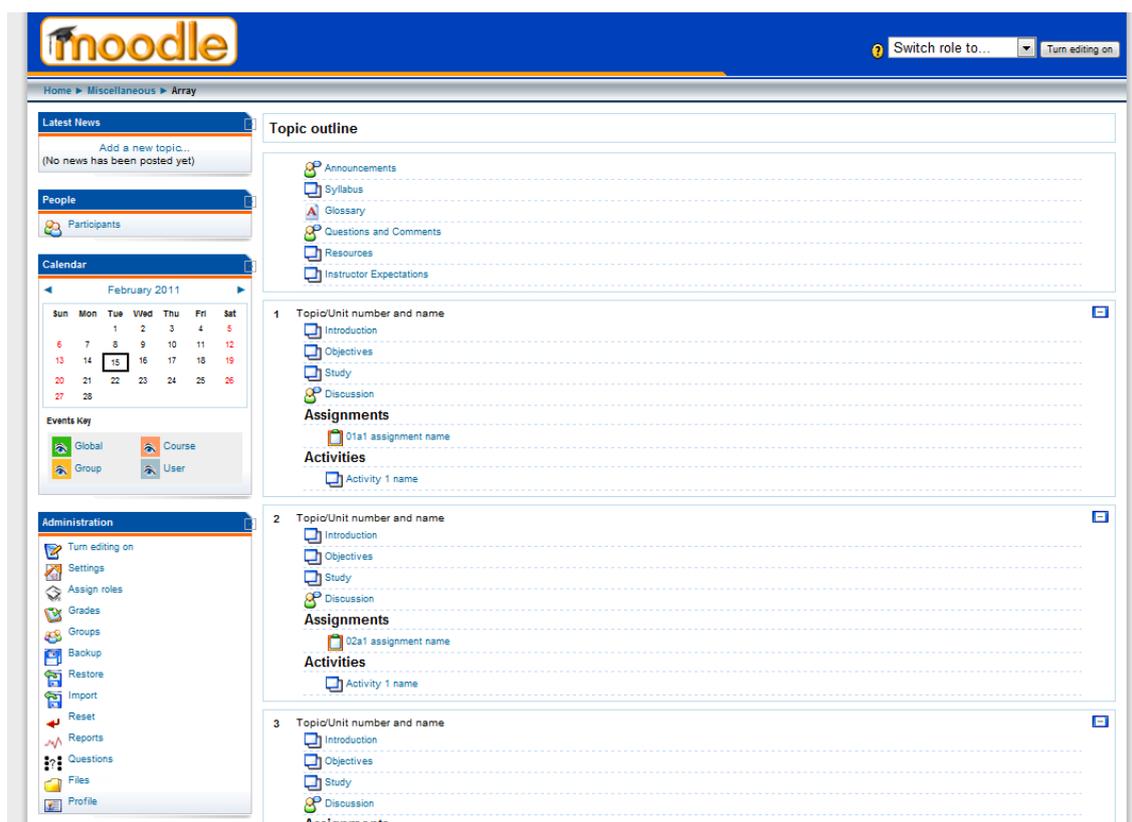


Figura 2. Visão Geral do Moodle 2.0.

3.1.3 – Sistemas de Gestão de Conteúdos de Aprendizagem

Os Sistemas de Gestão de Conteúdos de Aprendizagem (LCMS - *Learning Content Management System*) são sistemas voltados para o gerenciamento e armazenamento de objetos de aprendizagem. Eles combinam os aspectos administrativos e de gerenciamento de um LMS tradicional com as funcionalidades para criação de objetos de aprendizagem e remontagem personalizada de um CMS. O Moodle versão 2.0 e o Black Board. Segundo (Cohen et al. 2006) é um LCMS que suporta autoria, combinado com um repositório de objetos de aprendizagem e ferramentas para a disponibilização destes objetos aos alunos. São funções de um

LCMS apoiar a autoria, armazenar, aprovar, publicar e gerenciar conteúdo educacional. Entre os componentes usuais de um LCMS pode-se citar (Harmann et al. 2007):

- Repositório de Objetos de Aprendizagem;
- Aplicação de Autoria Automatizada;
- Interface de Distribuição Dinâmica;
- Aplicações Administrativas.

Os LCMSs agregam ferramentas de autoria de conteúdo, serviços de distribuição dos objetos de aprendizagem aos estudantes e serviços de manutenção. A Tabela 1 (Harmann et al. 2007) apresenta uma comparação entre as funcionalidades existentes entre as três categorias de ambientes analisados neste trabalho classificadas como (R) - Robustas e (L) – Limitadas. A análise dessas características serviram como base para projetar o StudentWatcher.

Tabela 1. Características de CMS, LMS e LCMS

CARACTERÍSTICA	FUNCIONALIDADE		
	Content Management System (CMS)	Learning Management System (LMS)	Learning Content Management System (LCMS)
Exemplos de ferramentas	Joomla	Moodle 1.0	Moodle 2.0
Gerenciar alunos		R	L
Gerenciar conteúdo	R		R
Criar conteúdo	L		R
Gerenciar sessões conduzidas pelo instrutor		R	
Catálogo de Cursos		R	L
Registro no sistema		R	L
Gerenciamento de competência		R	L
Acompanhar o processo de aprendizagem			L
Criação de avaliação		R	R
Biblioteca compartilhada de conteúdos	R		R
Integração com aplicações		R	

(R) - Robustas e (L) – Limitadas

3.2 – Sistemas de acompanhamento dos alunos e do curso

O sistema apresentado por (Huang et al. 2009) suporta avaliação formativa e auto-avaliação em aprendizagem com dispositivos móveis. Em uma proposta similar, (Chen et al. 2009) apresenta uma ferramenta para avaliação formativa, utilizando mineração de dados móvel e regras de inferência baseadas em Fuzzy - que permitem que o professor monitore o processo de aprendizagem dos alunos. (Zaiiane et al. 2001) e (Enright et al. 2008) propuseram um sistema de mineração que usa a análise das mensagens enviadas pelos alunos. (Dringus et al. 2005) propuseram uma solução baseada na análise de dados armazenados em bancos de dados. Além disso, adotaram algoritmos de mineração de dados para gerar padrões de comportamento sobre os alunos.

(Chen et al. 2009) apresentaram uma estrutura que fornece uma interface de comunicação baseada em agentes inteligentes. Na proposta de (Chen et al. 2007) são usadas técnicas de mineração de dados e análise que indicam o desempenho dos alunos durante o processo de avaliação. (Otsuka et al. 2007) apresentam um modelo projetado para fornecer suporte a avaliação formativa em LMSs, que é baseado em uma arquitetura multi-agente e o uso de ontologias. Nela, os agentes coletam e analisam a participação dos alunos em cursos a fim de identificar problemas, gerar alertas e indicar índices de qualidade. O sistema proposto por (Chen et al. 2009) trabalha com agentes que capturam e analisam dados sobre a participação dos alunos. Todas essas soluções anteriores têm em comum o fato de realizar o armazenamento e posterior análise dos dados que foram registrados em bancos de dados ou obtidos de logs do sistema gerados pelos usuários ou por aplicativos.

A maioria dos sistemas de tutoria disponibilizam sugestões ou feedback aos alunos após o aluno ter apresentado uma solução. Por exemplo, o SCoT (Pon-Barry et al. 2004) e o DC-comboio (Bulitko et al.2005) oferecem retorno ao aluno somente após o aluno salvar a atividade. Este é um tipo de feedback tardio e também usado em alguns sistemas de tutoria que ensinam habilidades em tempo real, como os de ensino de tática e combate e os de ensino de tomada de decisões críticas, como em um centro de informações de combate (Ritter et al.1988).

Essas ferramentas não buscam o acompanhamento durante o processo de aprendizagem do aluno pelo professor, observando as ações realizadas pelo estudante e, possibilitando intervenções no processo de aprendizagem no momento em que as dificuldades acontecem.

(Nkambou et al. 2004) desenvolveram um sistema de análise geral dos alunos, que permite identificar as dificuldades durante a atividade, apresentando uma sessão de resultados de aprendizagem. (H. Gamboa, V. Ferreira 2003) utiliza uma proposta de monitoramento e captura das atividades do aluno, utilizando um processo de monitoramento/armazenamento das ações em arquivos de logs para estudo posterior. Outro trabalho relacionado é o AdeLE (adaptável e-Learning com Eye Tracking) de (C. Gütl et al. 2004), que visa adaptar o conteúdo conforme o desempenho de aprendizagem.

A ferramenta desenvolvida nessa dissertação, o StudentWatcher, permite o monitoramento das ações realizadas pelos alunos e a entrega do estado para o professor, o que o apodera para realizar ações condizentes com o momento atual de aprendizagem – em um ambiente web. Dessa forma, o diferencial dela é que o seu foco é informar e alertar o professor do estado de aprendizagem dos alunos em tempo real. Contudo, a mesma oferece algumas intervenções pré-configuradas, como o oferecimento de material complementar (vídeo, áudio e texto). Espera-se também que essa ferramenta coloque o professor próximo dos alunos (independentemente da localização física) e que cada um deles siga o seu próprio ritmo de aprendizagem.

3.3 – Considerações do capítulo

Este capítulo apresentou o conceito de sistemas de aprendizagem, com seus principais tipos, CMS, LMS e os LCMS. Além disso, uma comparação de características sistêmicas, definindo-as como robustas ou limitadas, as quais ajudaram a determinar as características necessárias para atender os desafios e propostas do StudentWatcher. Por fim, foram discutidos diversos sistemas que visam avaliar o curso e o desempenho dos alunos, o que permitiu apontar a diferença da estratégia de ensino implementada no StudentWatcher e conseqüentemente sua contribuição.

CAPÍTULO 4 – STUDENTWATCHER: UMA FERRAMENTA DE ACOMPANHAMENTO INDIVIDUAL DE ALUNOS

Acompanhar os alunos individualmente sem o uso de ferramentas adequadas demanda esforço considerável dos professores. Este capítulo apresenta inicialmente a estratégia de acompanhamento implementada no StudentWatcher. Para isso, explica os Objetos de Aprendizagem Interativos (OAI). Logo após, apresenta esse ambiente de acompanhamento.

4.1 - Introdução

O StudentWatcher é uma ferramenta computacional para apoio educacional, que foi desenvolvido impulsionado pela tendência de convergência entre a aprendizagem eletrônica e convencional, na busca de uma coexistência harmoniosa entre o presencial e o virtual. O StudentWatcher é um software de apoio à aprendizagem, pode ser utilizado em várias plataformas que consigam executar a linguagem JAVA tais como Unix, Linux, Windows. MAC OS, acessível através da Internet ou de rede local. Utilizado principalmente num contexto de e-learning o programa permite o acompanhamento individual dos alunos em sala de aula. A base educacional e tecnológica do projeto é a criação de objetos interativos, que neste projeto são definidos como Objetos de Aprendizagem Interativos (OAI). Os OAI são capazes de monitorar as atividades realizadas pelos alunos, oferecendo ajuda e alertando o professor conforme as dificuldades específicas do aluno, ou seja, comportam-se conforme o aluno. Com a utilização do ambiente, estes objetos podem ser criados, gerenciados, distribuídos e reutilizados.

Enquanto o aluno resolve o exercício, várias ações podem acontecer no decorrer do tempo, como, por exemplo:

- O professor pode solicitar uma cópia da instância do objeto de um aluno (Figura 3): isso permite a verificação sobre o estado atual da resolução do exercício. Se o professor tiver a percepção da necessidade de uma intervenção, ele pode entrar em

contato com aluno (diretamente, no caso de ensino presencial; ou via chat/voz, no caso de ensino a distância);

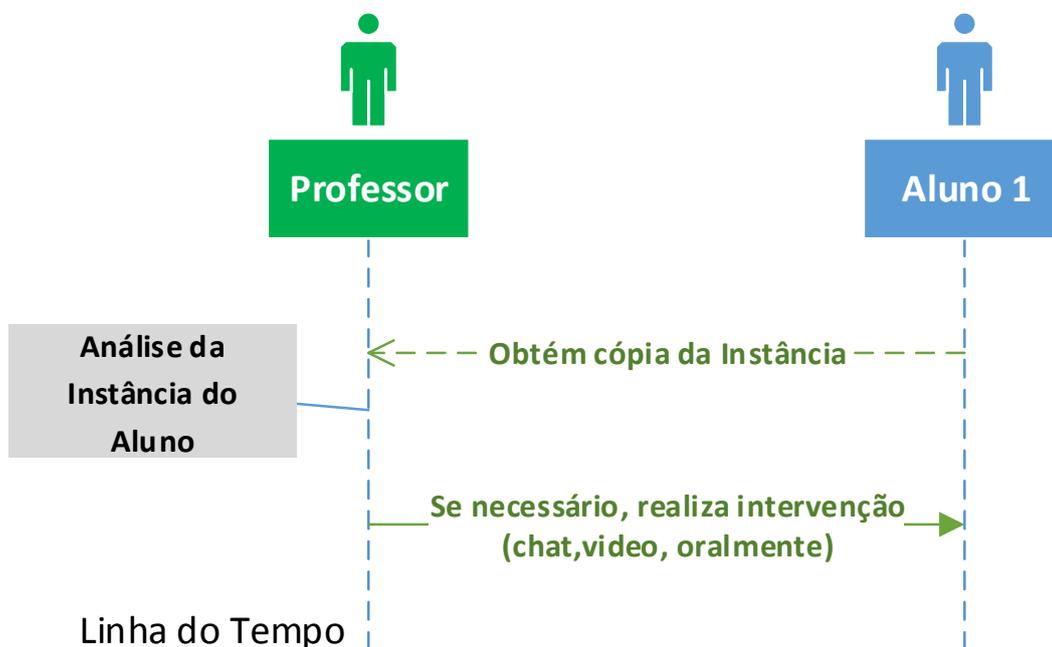


Figura 3. Análise da Instância do aluno.

- O professor é alertado (Figura 4): caso o aluno fique por um determinado período (tempo estabelecido pelo professor) sem interagir com a instância (digitando a resolução), o professor recebe um alerta. Com o objetivo de facilitar a visualização do estado geral da classe, o ambiente conta com um sistema de cores associado a cada estado de aprendizagem (verde – o estado de aprendizagem está conforme o esperado; laranja – já foram oferecidas algumas dicas automáticas de apoio ao aluno; vermelho – o aluno já recebeu as dicas automáticas e permanece sem interação com o ambiente);

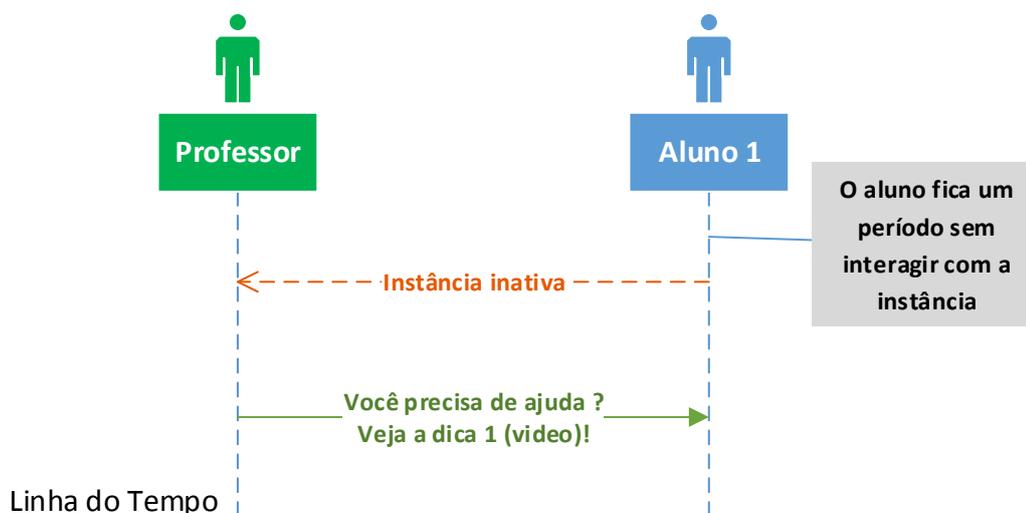


Figura 4. O aluno fica um período sem interagir com a instância.

- Oferecer ajuda (Figura 5): caso o aluno fique por um determinado período sem interagir com a instância, o sistema oferece automaticamente dicas para o aluno, como, por exemplo, um vídeo explicativo ou um texto;

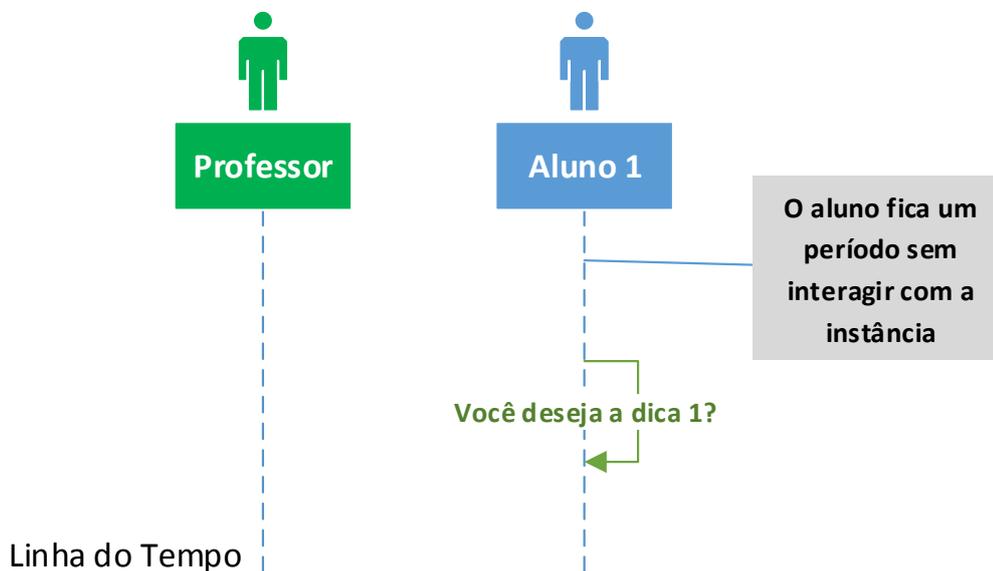


Figura 5. Ajuda oferecida automaticamente pelo sistema.

- Compartilhamento de instâncias: o professor gerencia todas as instâncias ativas, o que permite, por exemplo, obter uma instância de um aluno e utilizá-la para esclarecer uma dúvida para todos (Figura 6); ou criar uma instância e compartilhá-la a

fim de propor um novo exercício ou enfatizar algum ponto do conteúdo (Figura 7); ou os alunos, seguindo instruções do professor, podem compartilhar as instâncias entre eles, a fim de trabalhar de forma colaborativa na resolução de um problema (Figura 8).

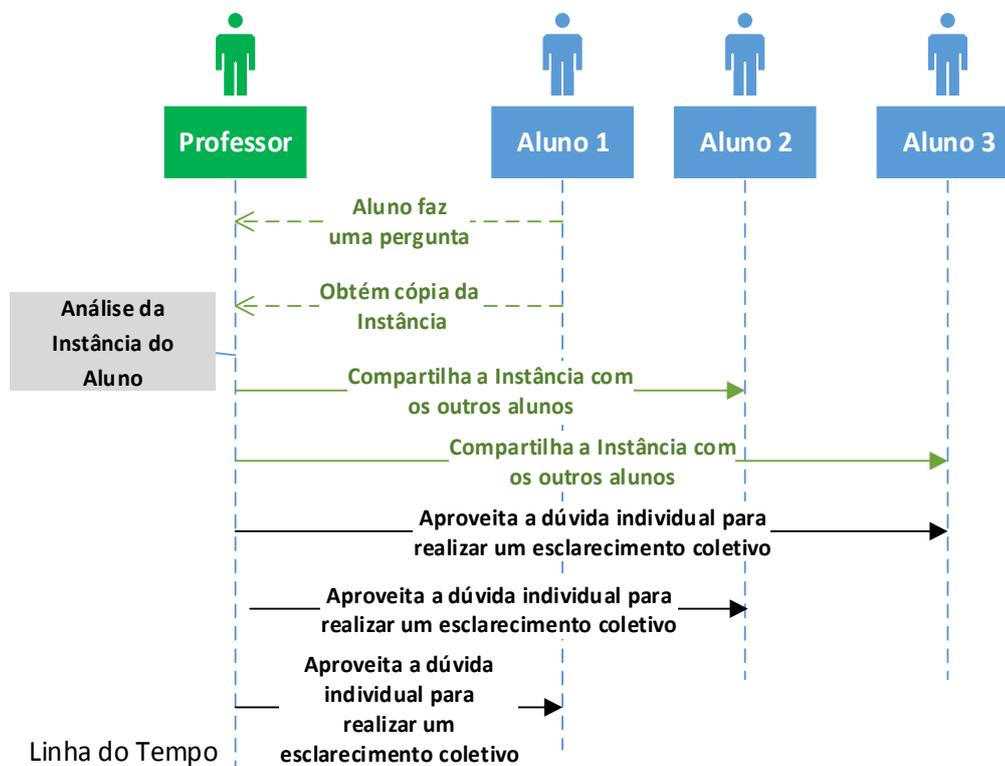


Figura 6. Esclarecimento coletivo do professor aos alunos.

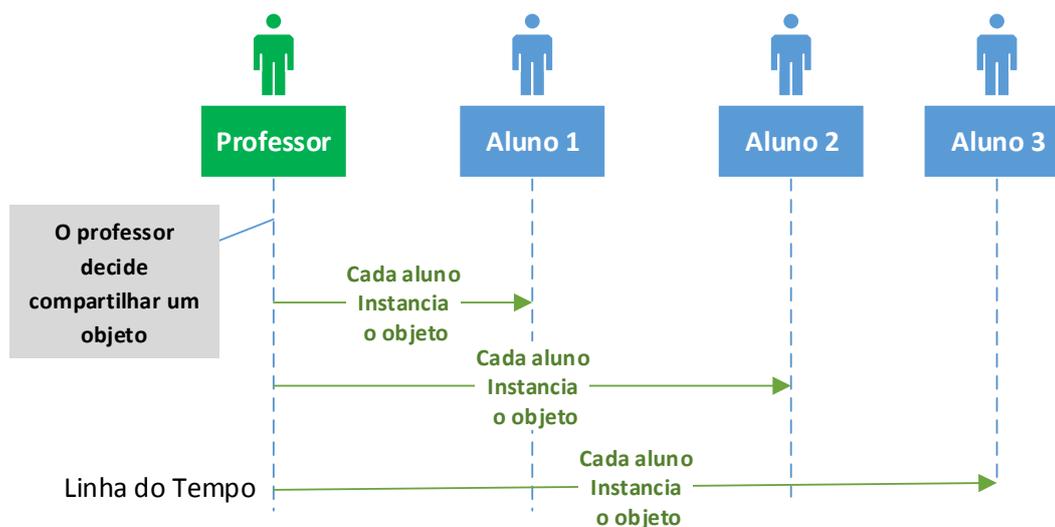


Figura 7. O professor pode compartilhar o mesmo exercício para vários alunos.

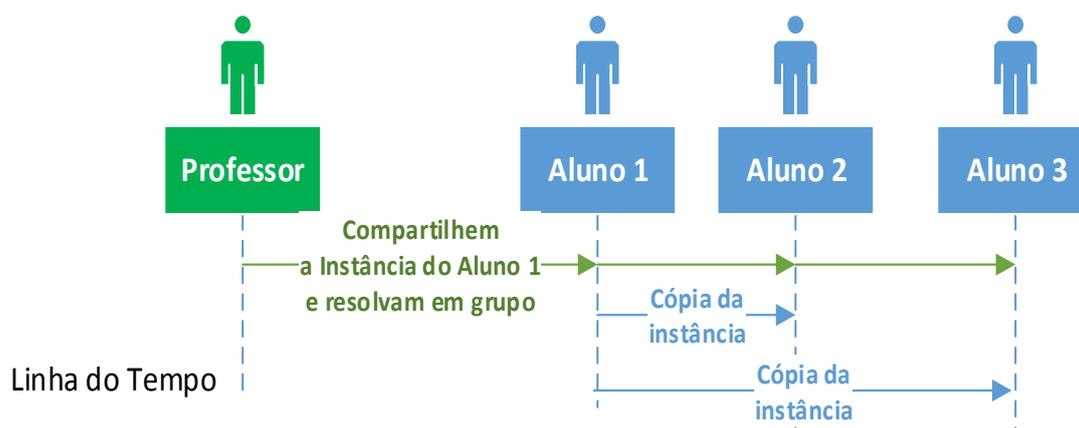


Figura 8. Compartilhamento de Instância em alunos.

- Uso das ferramentas: o aluno pode utilizar as ferramentas disponíveis para acionar a ajuda do professor ou para solicitar as ajudas associadas previamente ao objeto.

Este projeto vai além da definição e implementação de um ambiente de aprendizagem, ele está associado diretamente a uma estratégia educacional de acompanhamento que privilegia a sociabilidade, a construção do conhecimento e o atendimento individual do aluno. A Computação contribui com as técnicas de orientação a objetos (Deitel, 2006), que possibilitam a criação e gerenciamento de objetos; as técnicas da área de Interface Homem-Computador (Barbosa et al. 2010)

permitem a construção da interface utilizada pelos alunos e professores; as técnicas de Comunicação entre os computadores (Tanenbaum 2011) viabilizam o compartilhamento dos objetos entre os computadores, que estão no mesmo local ou distantes; as Tecnologias interativas (Tori 2010) possibilitam o uso de diversas mídias; e a área Educacional provê a estratégia educacional que permite o acompanhamento individual dos alunos.

4.2 – Arquitetura do StudentWatcher

O objetivo da ferramenta é aproximar os professores dos seus alunos, permitindo que cada aluno possa seguir o seu próprio ritmo de aprendizagem. A (Figura 9) mostra a visão geral da arquitetura do StudentWatcher:

- Interface Web do Professor: permite ao professor criar e/ou modificar o OAI e ensinar à distância ou apresentar cursos;
- Interface Web do Etudente: permite a participação dos alunos dos cursos e turmas;
- Módulo Modelagem do Estudante: monitora o comportamento de aprendizagem do aluno:
 - Objetos de Aprendizagem Interativo (OAI): consistem em conteúdo educacional que relata seu estado atual e se engaja em ações pré-configuradas;
 - Lógica Fuzzy – baseado em ações pré-configuradas pelo tutor/professor analisa o estado dos OAI para disparar ações como por exemplo, através do envio de um alerta para o professor e/ou sugerindo um vídeo para o aluno;
- Tecnologias Web: solução back-end implementada.

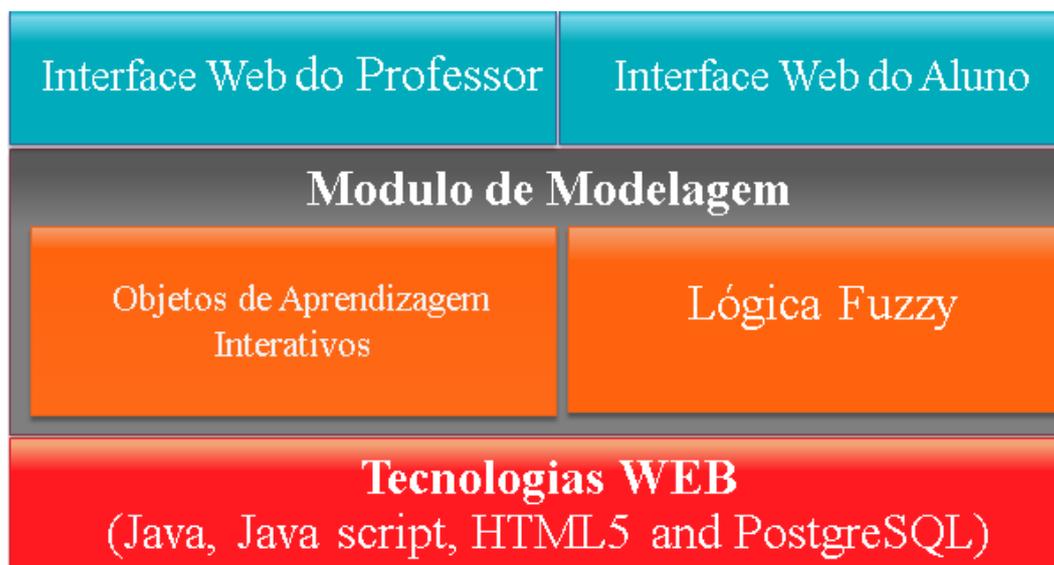


Figura 9. Arquitetura do StudentWatcher.

Para modelar o comportamento do StudentWatcher visou-se o diagrama de caso de uso, conforme mostra a (Figura 10). Nela está demonstrado todos os atores que interagem com o ambiente, no caso Aluno, Professor e o Administrador.

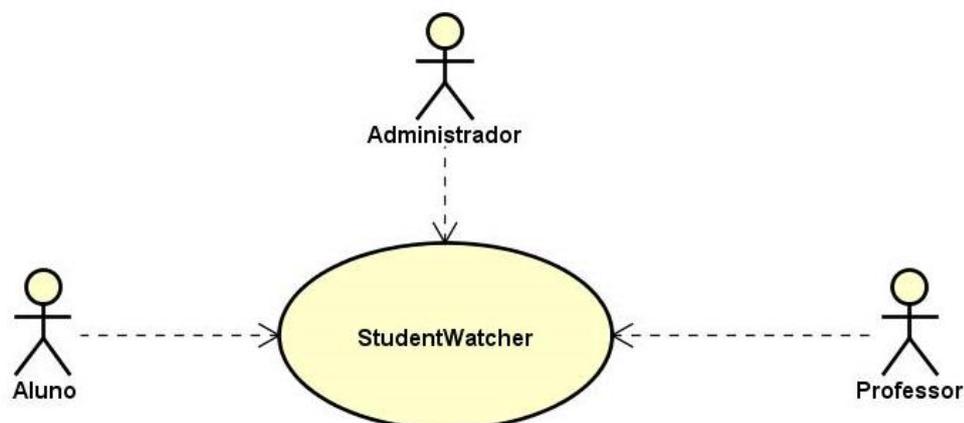


Figura 10. Diagrama de caso de uso.

A (Figura 11) mostra o diagrama de pacotes do StudentWatcher, que é composto de forma geral por três módulos: Aluno – funcionalidades referentes ao gerenciamento dos alunos, como, os cursos e disciplinas que eles matriculados; Professor – trata-se dos serviços oferecidos para os professores, como, o gerenciamento das turmas que o mesmo ministra aulas; e o módulo Acompanhamento – que implementa o gerenciamento das atividades realizadas pelos alunos, então gerencia os objetos de

aprendizagem interativos (módulo OAI) e usa as funcionalidades relacionadas com a Lógica Fuzzy (módulo IA).

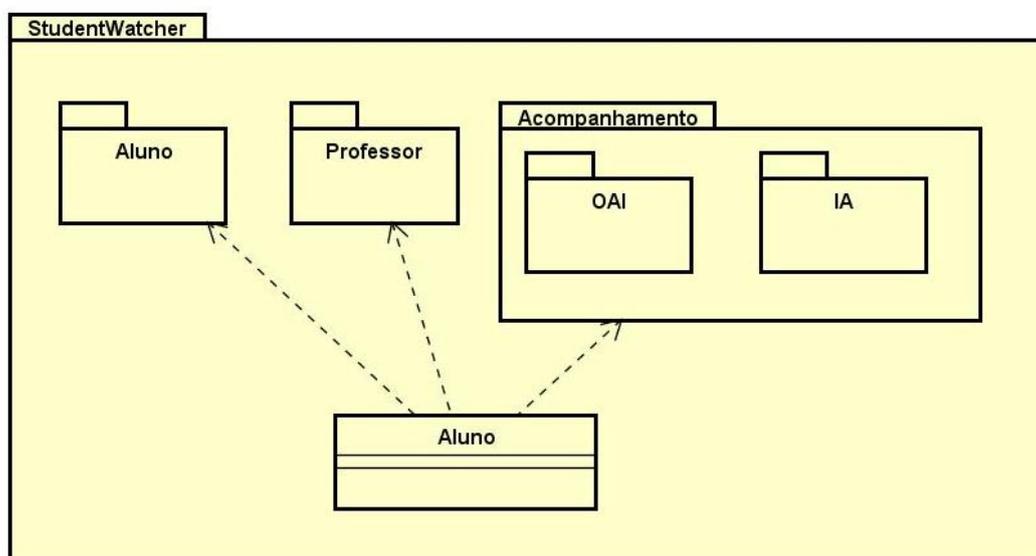


Figura 11. Diagrama de classes.

4.2.1. Tecnologias Web

Recentes avanços em tecnologias web têm permitido aplicações sofisticadas. Atualmente, os principais navegadores suportam HTML 5 (Sarris 2013), o que torna sua arquitetura ideal para o desenvolvimento de aplicações web complexas. Isso fez com que essa linguagem fosse escolhida para se desenvolver o StudentWatcher. No momento presente, é possível executar aplicações complexas na web e até mesmo em dispositivos móveis, com alta qualidade de imagem e vídeo e que são responsivas - feedback imediato.

Todos os OAI e informações do StudentWatcher (ou seja, as informações sobre alunos, professores e cursos) são armazenados em um servidor na internet, sendo acessados através de um protocolo HTTP REST. Tanto o site e as suas aplicações usam este mesmo protocolo, mantendo o back-end simples. A resposta é enviada no padrão JavaScript Object Notation (JSON), que é uma maneira de armazenar informações de forma organizada, completamente independente do idioma, e em um formato de intercâmbio de dados leve. O banco de dados utilizado para o armazenamento dos dados dos usuários e do dos OAI foi o PostgreSQL e a linguagem de programação é JAVA. A

(Figura 12) ilustra essa arquitetura. Foram implementadas funcionalidades de segurança da informação para validar a autenticação de usuários com um teste de desafio cognitivo a partir da utilização de CAPTCHA, métodos de controle de comunicação de dados utilizando HTTP POST mais HTTPS e controle de sessões no banco de dados para evitar XSS e SQL Injection.

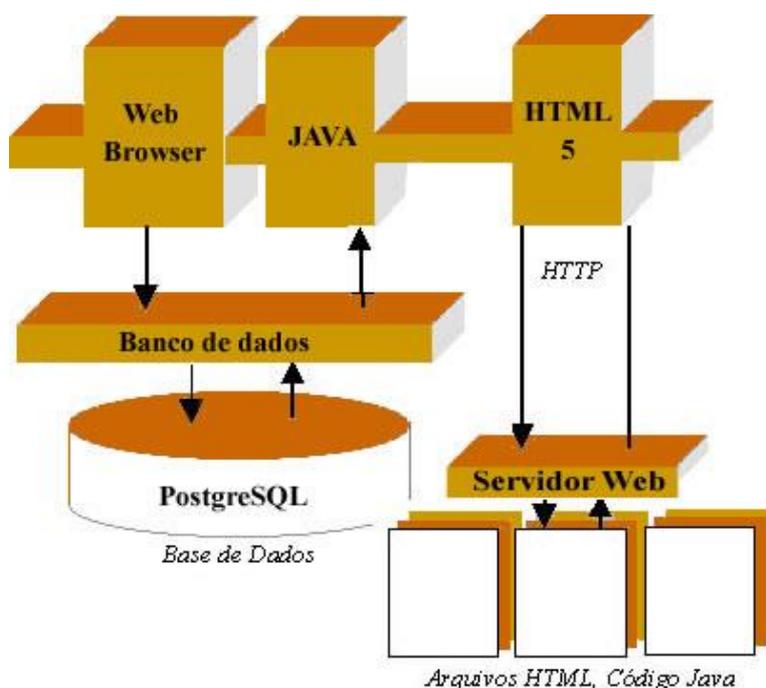


Figura 12. Arquitetura e Tecnologia Web.

4.2.2 Objetos de Aprendizagem Interativos

O avanço da internet tem impulsionado vários mecanismos capazes de proporcionar o aprendizado e conhecimento através de conjuntos de materiais didático pedagógicos, transformando-os em unidades de aprendizagem passíveis de reutilização e compartilhamento, os chamados objetos de aprendizagem (OA) (Downes 2001).

Segundo (L'Allier1997), um objeto de aprendizagem é definido como “a menor estrutura instrucional independente que contém um Objetivo, uma Atividade de Aprendizagem”, consistido em um componente digital, base de um curso, unidade ou lição que pode ser reutilizado para criar outras estruturas instrucionais únicas. Tendo suas principais características descritas como: 1) flexibilidade – desenvolvido para ser reutilizado; 2) personalização – personalização de conteúdos permitindo a sua

recombinação; 3) interoperabilidade – permite a definição de especificações de design, de desenvolvimento e de apresentação baseadas em necessidades da organização; 4) aumento de significância do conhecimento – com a reutilização de um objeto de aprendizagem várias vezes e em diferentes situações, seu conteúdo é consolidado naturalmente à medida que o tempo passa.

Os objetos de aprendizagem já são pesquisados à décadas, em 1998 foi feita a primeira definição ampla, permitindo que qualquer material seja considerado um objeto de aprendizagem (Balbino 2007). O termo objeto educacional é aplicado a materiais projetados e construídos em conjuntos pequenos com o intuito de potencializar a aprendizagem. Em geral os objetos se aplicam para ajudar o aluno a compreender o assunto, se tornando uma das ferramentas essenciais para o conhecimento e aprendizagem.

É esperado que os objetos de aprendizagem sejam desenvolvidos de maneira a prover compatibilidade com outros objetos. Atualmente, existem diversas especificações que normatizam o desenvolvimento deles. Entre as principais especificações destaca-se o Instructional Management System (IMS) Learning Design e o Sharable Content Object Reference Model (SCORM).

Os objetos de aprendizagem do StudentWatcher são criados com as características já previstas no padrão SCORM. O padrão SCORM foi expandido, adicionando elementos (como, por exemplo, tempo sem interação, nível de dificuldade, ajuda – texto, vídeo e áudio) para permitir o acompanhamento do estado atual do objeto. Assim, criaram-se então denominados Objetos de Aprendizagem Interativos (OAI) (Luz et al., 2015). Com um olhar voltado para a interação e principalmente para os benefícios agregados que ela proporciona no processo de ensino-aprendizagem, (Kemezinski et.al 2012) descreveram uma metodologia para construção de objetos de aprendizagem interativos, porém, se limitaram a este processo sem a preocupação de estruturar a construção do objeto baseada em padrões, como o SCORM. No caso do StudentWatcher, para ser considerado um OAI, ele deve permitir a interação conforme ilustra a (Figura 11). Ela representa como os OAI são pré-configurados pelo professor e seus parâmetros, como por exemplo, quando o professor será alertado que o aluno não está interagindo com o objeto; e quando dicas, como textos e vídeos, serão oferecidas

para o aluno. Ao monitorar o estado do objeto, o StudentWatcher pode realizar diversas ações, como oferecer ajuda e/ou alertar o professor que poderá visualizar o estado atual do objeto do aluno e, se desejar, iniciar uma interação via chat.



Figura 13. Operação de um Objeto de Aprendizagem Interativo.

O StudentWatcher foi projetado e desenvolvido de forma a garantir a personalização e interação dos OAIs, sendo construído seguindo a definição destes objetos, garantindo, por exemplo, a utilização de chats, dicas e controle do objeto em tempo real. Como os OAIs são uma extensão do SCORM, então podem ser exportados para ferramentas compatíveis, contudo perdem as funcionalidades de interação. A (Figura 14) mostra o diagrama de classes dos OAIs, cujo ponto chave é a classe *Atividade*, que representa os OAIs que o professor cria para os alunos e a classe *Aluno_Atividade*, utilizada para interligar o aluno a uma atividade.

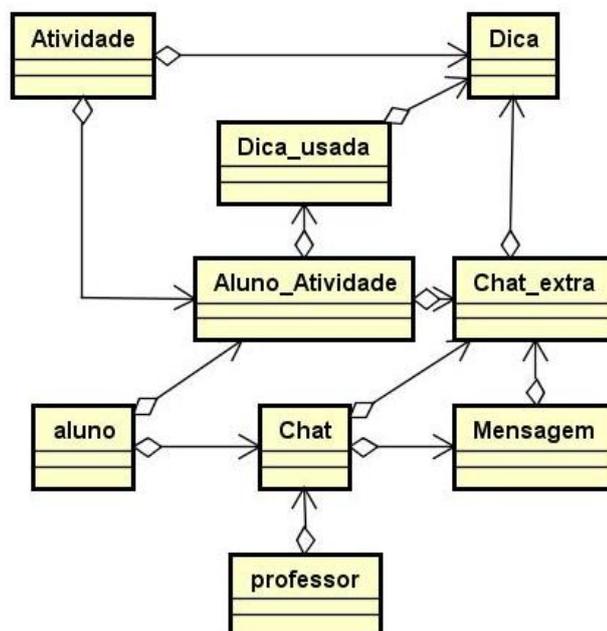


Figura 14. Classe dos Objetos de Aprendizagem Interativos.

4.2.2.1 Ontologia do OAI

Uma ontologia é uma descrição completa e explícita de determinado domínio, fazendo e permitindo a relação entre os conceitos, objetos e propriedades que a compõe. Para dar continuidade no trabalho de definição do novo padrão de OAI, foram utilizadas técnicas de Web Semântica para a construção de uma ontologia baseada no padrão de metadados XML criado, permitindo então que exista uma relação entre eles, agregando significados uns aos outros e automaticamente se complementando. A ontologia dos OAI foi criada utilizando a linguagem OWL, considerando os princípios da Web Semântica. Assim, visou-se facilitar os mecanismos de busca e proporcionar significado aos dados para uma leitura e interpretação pelos computadores, também o monitoramento das ações na execução dos OAI pelos alunos no ambiente.

Com intuito de fortalecer a ontologia, alinhando seus elementos com os de outras já existentes (Jiang et al. 2014), a importação e uso de propriedades e classes da ontologia do Dublin Core, OBAA, IMS-LD e LOM foram utilizadas, sendo respectivamente:

- <http://purl.org/dc/elements/1.1/>
- <http://gia.inf.ufrgs.br/ontologies/OBAA.owl>
- <http://gia.inf.ufrgs.br/ontologies/LOM.owl>
- http://gia.inf.ufrgs.br/ontologies/IMS_AccessForAll.owl.

Assim, utilizando elementos e especificações dos OAI, foi possível garantir uma identificação entre os padrões, convergindo para uma nova ontologia em OWL. A construção da ontologia foi feita no software Protégé v5.0 (Stanford, 2010). Alguns níveis fundamentais no processo de criação e validação das ontologias são: vocabulário; hierarquia (taxonomia); contexto (nível de aplicação); Nível Sintático; Arquitetura, Design e Estrutura (Hazman et al. 2011). A preocupação inicial na criação da ontologia dos OAI levou em consideração cinco aspectos, que foram identificados como classes principais no modelo. São eles: Objeto Aprendizagem (LearningObject), Conteúdo (Content), Metadados (Metadata), Agregações (Aggregate) e Vocabulário (Vocabulary) (Figura 15).

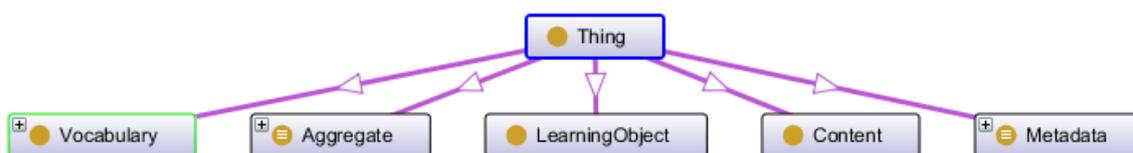


Figura 15. Classes da Ontologia OWL OAI.

A classe Objeto Aprendizagem se refere às instâncias dos objetos propriamente ditos, tendo relação direta com as classes de metadados (Figura 12), agregações e conteúdo, sendo que esta última não foi descrita neste trabalho, pois trata do conteúdo instrucional do próprio OAI.

Dentro da ontologia do OAI cada classe é descrita com seus identificadores, contendo: Nome da Classe, URI, Classes equivalentes, Classes Disjuntas (ou mutuamente exclusivas) e Anotações. Neles é possível visualizar a ligação entre os elementos e subclasses do mesmo modelo em XML. A (Figura 16) apresenta a descrição da classe Metadata dentro da Ontologia.

Metadata
URI: http://www.purl.org/ilo/interactivelearningobjects/Metadata
Equivalent classes:
Metadata EquivalentTo Educational or General or LifeCycle or Rights or Technical
Disjoint classes:
Aggregate DisjointWith Metadata
LearningObject DisjointWith Metadata
Annotations:
comment "ILO (pt): Categories of metadata defined as IEEE-LOM and adapted by ILO."^string

Figura 16. Especificação da Classe Metadata (metadados).

A classe Objeto Aprendizagem representa as instâncias dos objetos descritos com seus Metadados. A classe Conteúdo (*Content*) representa a composição do objeto de aprendizagem, não sendo descrita nesta versão da ontologia. A classe Agregações (*Aggregate*) se refere aos elementos de metadados que podem ser compostos por outros metadados. A classe Vocabulário (*Vocablulary*) compreende, dentre as ontologias e padrões utilizados como base, aqueles metadados que possuem composição de valores fixos e que não podem sofrer alteração além dos já previstos. A relação entre as classes é explícita, seguindo o formato: Metadados hasSubClass Geral.

Todos os 47 elementos do padrão OAI (sendo trinta e cinco elementos de metadados, cinco classes e sete subclasses) foram divididos dentro da estrutura da ontologia em Data Properties e Object Properties. O primeiro faz referência aos elementos do tipo simples que podem ser relacionados diretamente ao metadado. Seguindo nesse caso o modelo `its<nome metadado>Is`. O segundo atende os elementos compostos, indicados na classe Agregações. Eles são indicados com a estrutura de `has <nome metadado>`.

No modelo de propriedades dos objetos todos os relacionamentos do tipo `has <nome metadado>` possuem um relacionamento inverso identificado como `its <nome metadado> Of`.

metadataExample	hasTips	Tips01	Tips01	isTipsOf	metadataExample
Tips01	itsIdTipsIs	001	Tips01	itsIdTipsIs	001
Tips01	itsTimeTips	120	Tips01	itsTimeTips	120

Figura 17. Especificações dos objetos e atributos.

Dentro da Linguagem OWL, o perfil utilizado foi de OWL-DL, que atendeu a especificação de todo domínio sem maiores restrições. Um trecho de um OAI definido na ontologia é mostrado na (Figura 18), tendo em destaque em “vermelho” os metadados e em “azul” o preenchimento conforme conteúdo do OAI. Este trecho corresponde ao Interactive Learning Object-1, a identificação de duas classes principais, *Metadata* e *Aggregation*, contendo apenas o trecho referente a subclasse *Activity* (Atividades).

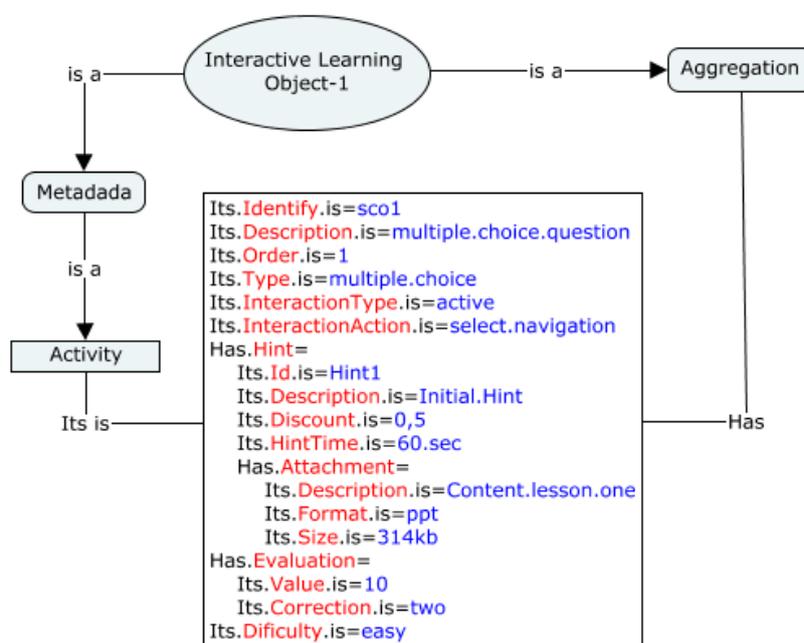


Figura 18. Trecho do OAI mapeado na Ontologia OAI.

De posse da ontologia compactada e com a utilização de uma ferramenta de operação com Ontologias, denominado JOINT (Holanda et al. 2013), foi possível gerar a codificação da mesma na linguagem Java, utilizada para desenvolver o StudentWatcher. A (Figura 19) representa a transposição da ontologia pelo JOINT para a sua representação no ambiente.

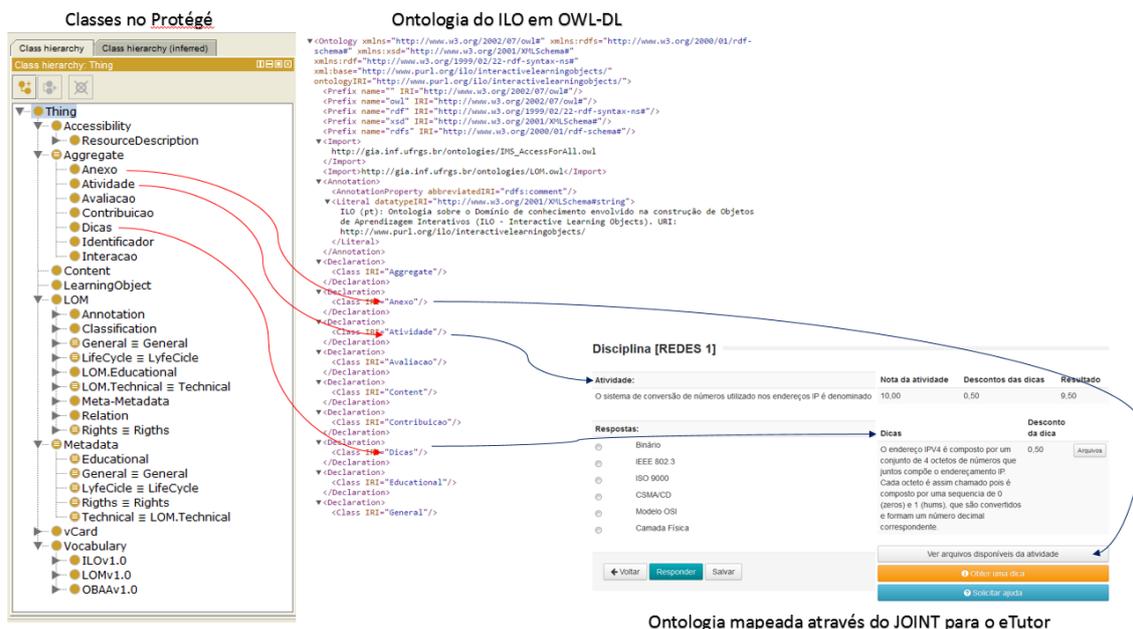


Figura 19. Transposição do padrão da Ontologia Computacional para o StudentWatcher.

4.2.2.2 Lógica Fuzzy

Para monitorar o estado da aprendizagem dos alunos, tornou-se necessário o desenvolvimento de um módulo de tomada de decisão implementado utilizando Lógica Fuzzy (Zadeh, L.A. 1965). Durante a criação de um OAI o professor pode efetuar configurações prévias que servem como base para ações automáticas. Então, o estado atual do OAI, que representa a aprendizagem do aluno, determina se intervenções no processo de aprendizagem são necessárias. Por exemplo, o professor pode configurar uma ação pré-determinada quando o aluno não adicionar nenhuma informação em 5 minutos em um exercício, cuja resposta é descritiva, então o StudentWatcher deve inicialmente oferecer dicas como vídeos explicativos, e se a situação persistir, o professor deve ser alertado.

Este módulo se assemelha ao modelo adaptativo de (Seters et al. 2012), que integra informações anteriores (como tempo de inatividade, dicas utilizadas) com a finalidade de selecionar o conteúdo de aprendizado apropriado para apresentar ao aluno, ou ainda, efetivar ações como comunicar o professor, baseado em variáveis de entrada e de saída, que representam o resultado final da avaliação de desempenho do aluno (e a intervenção ou não no processo de aprendizado).

As seguintes variáveis de entrada do OAI são consideradas para o disparo da ação apropriada:

- Tempo sem interagir: significa quanto tempo o aluno ficou sem interagir com o OAI, seja ele o exercício propriamente dito ou instrumentos complementares de aprendizagem, tais como um vídeo ou um texto;
- Quantidade de caracteres inseridos: porcentagem de caracteres que o usuário digitou em função de um número médio de caracteres por exercício, definido pelo professor. Esta quantidade pode ser baixa, média ou alta;
- Pedidos de ajuda: significa a quantidade de pedidos já realizada pelo aluno, que pode ser baixo, médio ou alto;
- Dificuldade do exercício proposto: nível do exercício que o professor propõe, que pode ser rotulado como fácil, intermediário e difícil.

Considerando essas variáveis de entrada, o StudentWatcher determina o nível de ajuda a ser oferecido ao aluno:

- Nível 0 – nenhuma ajuda;
- Nível 1 – áudio, vídeo apresentado ao aluno como material de apoio sobre o tema;
- Nível 2 – rever exercício – rever um exercício comentado sobre um tema que tenha sido apresentado pelo professor;
- Nível 3 – ajuda monitor, ou seja, o monitor da disciplina é designado para ir ao computador aonde está o aluno;
- Nível 4 – ajuda professor, ou seja, é requisitada a ajuda presencial do professor em sala de aula.

Faz parte também desse módulo, fatores de penalidades associados aos níveis de dicas utilizados. Estas penalidades são também configuradas, opcionalmente, pelo professor ao criar o OAI, que descontará um valor da nota associada à atividade. As penalidades resultam nos graus de pertinência apresentados nas (Figuras 20 e 21). O grau de pertinência é um valor real compreendido no intervalo $[0,1]$, que significa o quanto é possível que o valor da variável pertença ao conjunto Fuzzy. Essa função de

pertinência é fundamental para que seja possível utilizar Lógica Fuzzy (Zadeh 1965 e Ross 2010).

Variáveis de Entrada		
Nome da Variável	Termos Linguísticos	Grau de Pertinência (intervalo)
Tempo sem interagir	Pequeno	[0, 60]
	Médio	[58, 180]
	Alto	[170, 300]
Quantidade de caracteres inseridos	Baixa	[0, 10% da quantidade média de caracteres para o exercício]
	Média	[9% da quantidade média de caracteres para o exercício, 30% da quantidade média de caracteres para o exercício]
	Alta	[28% da quantidade média de caracteres para o exercício, 100% da quantidade média de caracteres para o exercício]
Pedidos de Ajuda	Baixo	[0,1]
	Médio	[1,3]
	Alto	[2,5]
Dificuldade do exercício proposto	Fácil	[0, 5]
	Médio	[5, 7.5]
	Difícil	[7, 10]

Figura 20. Variáveis Fuzzy de Entrada.

Variáveis de Entrada		
Nome da Variável	Termos Linguísticos	Grau de Pertinência (intervalo)
Nível de Ajuda	Nível 0 – nenhuma ajuda	[0,1]
	Nível 1 – áudio, vídeo	[1,2]
	Nível 2 – rever exercício	[2,3]
	Nível 3 – ajuda monitor	[3,4]
	Nível 4 – ajuda professor	[4,5]

Figura 21. Variáveis Fuzzy de Saída.

A (Figura 22) mostra o diagrama de classes das atividades, cujos pontos chave são as classes *Atividade*, *Aluno_Atividade* e *Dicas*. Os principais atributos dessas classes formam as variáveis de entrada Fuzzy.

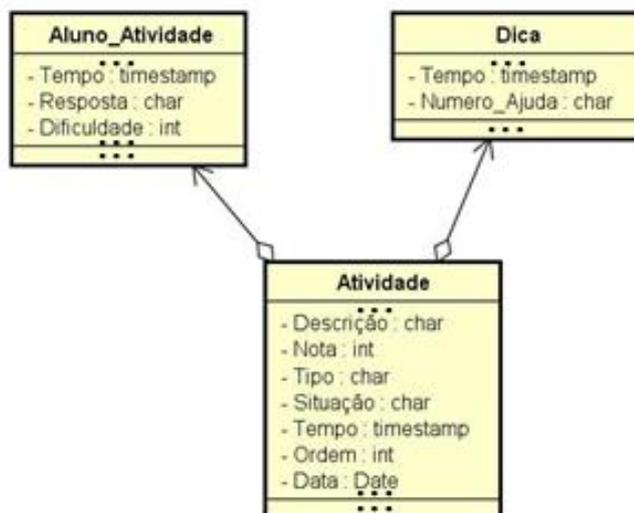


Figura 22. Diagrama de classes fuzzy

4.2.3. Interface Web do estudante

O foco do StudentWatcher é o oferecimento de soluções que permitam o acompanhamento individual dos alunos, a fim de estimular situações de aprendizagem capazes de proporcionar uma melhor assistência educacional, através de iniciativas que promovam a interação entre os envolvidos. Cada OAI é composto por atributos, como nome, links e objetos multimídia (áudio, vídeo e texto); ações, como os alertas enviados para o professor quando não ocorre interação com a OAI e os tipos de ajudas a serem oferecidas aos alunos.

A (Figura 24) mostra um exemplo da interface visualizada pelo aluno na aula de segurança da informação, ministrada pelo professor durante uma atividade de definições de normativas ISO 27002.

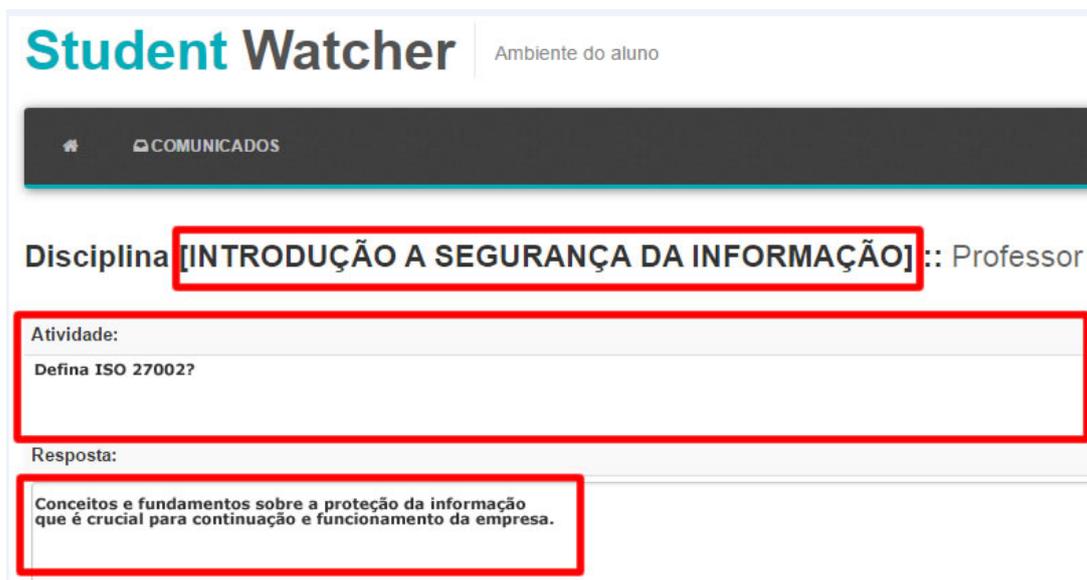


Figura 23. StudentWatcher - Interface Principal – Aluno.

A (Figura 24) mostra ao o aluno Caue Becker, que ele já solicitou todas as dicas. Contudo, ele conta com o conteúdo de apoio – clicando no botão “Ver arquivos disponíveis da atividade” ou no botão “Ver arquivos disponíveis da disciplina”. Ele também pode solicitar ajuda do professor via chat - clicando no botão “Solicitar Ajuda”.



Figura 24. StudentWatcher - Interface do aluno

A (Figura 25) mostra o diagrama de classes da interface do aluno, cujos pontos-chave são as classes *Instituição*, que representa as instituições de ensino; *Turma*, que representa as turmas (classes) de alunos dos cursos da instituição; *Disciplina*, que representa as matérias dos cursos; *Aluno*, que representa o aluno; *Professor*, que representa o professor; *Resposta*, que representa as respostas possíveis de uma atividade - podendo ser múltipla escolha, a qual deve conter respostas possíveis para apresentar ao aluno; e a *Ano_letivo*, que corresponde ao ano letivo de estudo da turma.

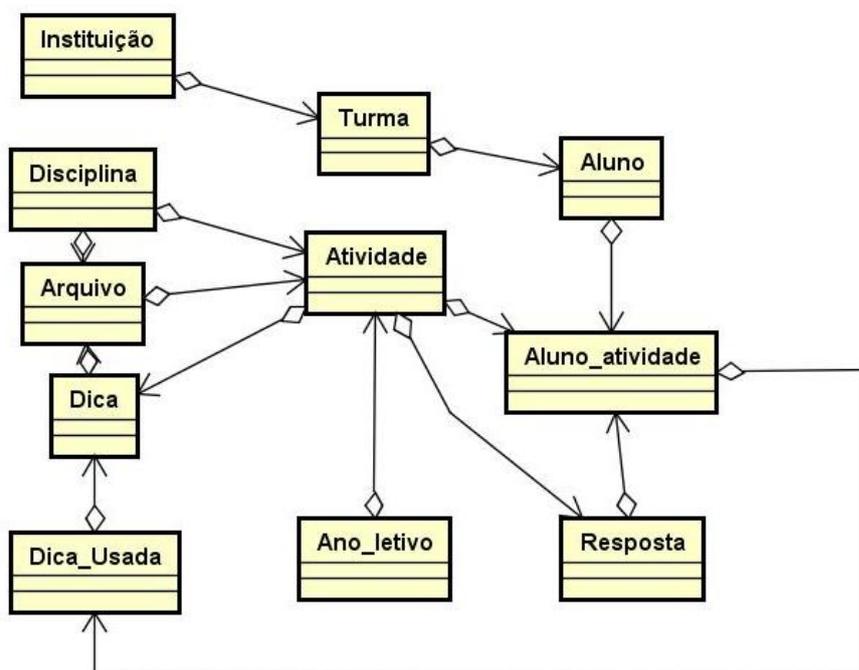


Figura 25. Classe aluno do StudentWatcher

4.2.3. Interface Web do professor

No StudentWatcher o professor cria os OAI e os distribui entre os alunos. O ambiente monitora as ações realizadas nos objetos. Na interface do professor podem ocorrer diversas ações, dentre elas: envio de alerta, caso um aluno fique por um determinado período de tempo sem interagir com o OAI; geração de um alerta ao professor caso o aluno esteja solicitando ajuda e já acessou todos os outros tipos de ajuda (texto, vídeo, áudio). O professor também pode gerar ações na interface, como, por exemplo, envio de ajuda ao aluno (por exemplo, um vídeo ou texto explicativo).

Ao professor é disponibilizada uma interface específica, que permite o monitoramento de todos os alunos, controle de tempo e data para a execução dos OAIs, e levantamento estatístico do desempenho da turma. Nela, o professor pode solicitar uma cópia do OAI de um aluno, o que permitirá sua verificação sobre o estado atual da resolução do exercício. Se o professor tiver a percepção da necessidade de uma intervenção, ele poderá entrar em contato com o aluno (diretamente, no caso de ensino presencial; ou via chat/voz, no caso de ensino a distância).

A (Figura 26) ilustra uma situação em que o professor está verificando o perfil dos alunos (Nome, Código de Matrícula e Turma), as solicitações de ajuda, assim como a situação das atividades (objetos) em um determinado momento. O professor pode acompanhar a mudança das cores no monitoramento do aluno conforme sua atividade atual.



Figura 26. Painel de acompanhamento do professor.

Com o objetivo de facilitar a visualização do estado geral da turma, o ambiente conta com um sistema de cores associado a cada estado de aprendizagem (verde – o estado de aprendizagem está conforme o esperado; laranja – já foram oferecidas algumas dicas automáticas de apoio ao aluno; vermelho – o aluno já recebeu as dicas automáticas e permanece sem interação com o ambiente). A (Figura 27) mostra um digrama de estados que ilustra os possíveis estados de aprendizagem de uma atividade. No momento em que uma atividade é iniciada, ela recebe o estado verde, podendo ir

para o estado laranja ou vermelho. Independentemente do estado, a tarefa pode ser finalizada.

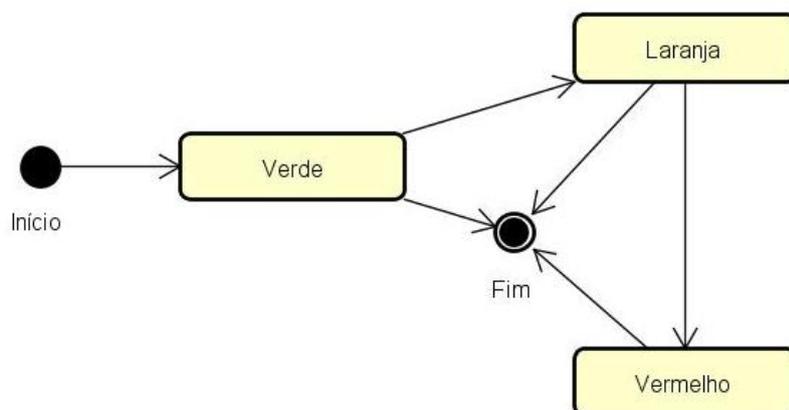


Figura 27. Diagrama de estado das cores (estado de aprendizagem).

O calendário permite ao professor programar as datas que o OAI será disponibilizado, podem ser compartilhados entre turmas. O compartilhamento da agenda permite que os alunos vejam todos os detalhes dos OAIs. A (Figura 28) mostra os objetos (atividade) agendados para o mês de agosto de 2015.



Figura 28. Calendário para controle de tempo dos OAIs.

O levantamento estatístico (Figura 29) de desempenho das turmas baseia-se nos dados extraídos dos OAI, sendo eles: número de Atividades (NV), atividades em desensolvimento (AD), atividades salvas (AS), atividades respondidas (AR) e totalizadores (TT). O acesso a esse levantamento está disponível na interface do professor.

Alunos // **Turma**

INTSEG-2014-04							Desempenho na atividade
Aluno	Qtde de Atividades					TT	%
	NV	AD	SV	RC	TT		
ALUNO 04	4	5	1	0	10	0	
BRUNO AZEVEDO	5	0	0	5	10	50	
CAUE BECKER	4	1	0	5	10	50	
GUSTAVO ABREU	8	1	1	0	10	0	
LUCAS NUNES	3	2	0	5	10	50	
SARA BENEDICTO	5	0	0	5	10	50	
VIVIANE ARAUJO	5	0	0	5	10	50	

Figura 29. Levantamento estatístico de desempenho.

A (Figura 30) mostra o diagrama de classes da interface do professor, cujos pontos chaves são as classes *Turma*, *Disciplina_turma*, *Professor*, *Disciplina*, *Atividade* e *Arquivo*. Por intermédio dessas classes o professor pode realizar todas as operações.

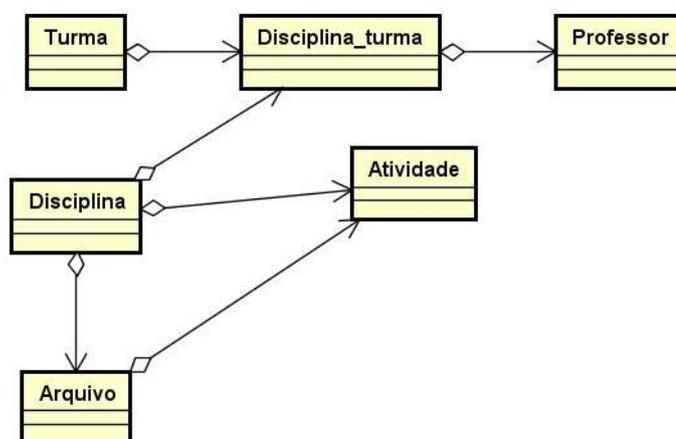


Figura 30. Diagrama Professor do StudentWatcher

As configurações StudentWatcher podem ser feitas pelos professores ou administrador. No caso dos OAI, é atribuição do professor configura conforme o comportamento desejado.

4.2.4 Processo de ensino-aprendizagem do StudentWatcher

O uso de ferramentas computacionais no contexto educacional não é novo. Várias ferramentas oferecem feedback aos alunos (Pon-Barry et al. 2004) ou uma de análise geral da aprendizagem (Nkambou et al. 2004). A StudentWatcher é diferente delas em diversos aspectos, como o acompanhamento das atividades dos alunos em tempo real e o recurso de avisar ao professor que o aluno está com dificuldades.

O ambiente StudentWatcher permite que o professor crie os OAI e os distribua, inclusive com a possibilidade de armazená-los em repositórios. Assim, ele vai além de um simples visualizador de atividades, é uma solução que permite a um professor planejar e executar aulas de acordo com seus objetivos educacionais. Dessa forma, espera-se que o professor crie, manipule e compartilhe os OAIs sem se preocupar com aspectos tecnológicos.

A (Figura 31) ilustra o ciclo esperado de criação e uso dos OAI. Inicialmente, o professor cria o objeto; em seguida, configura e adiciona o conteúdo (atividades); logo após, os alunos fazem uso e, conseqüentemente geram os resultados. Por fim, esses resultados podem ser analisados e o objeto aprimorado.

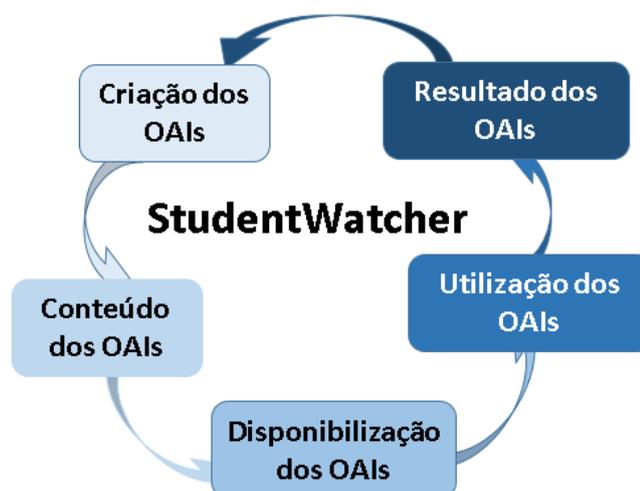


Figura 31. Processo de ensino-aprendizagem do StudentWatcher.

4.5 – Considerações do capítulo

Este capítulo apresentou os princípios de funcionamento e a arquitetura do StudentWatcher. Para isso, inicialmente, foi descrito o ambiente, dando ênfase às ações realizadas com os OAI. Em seguida, apresentou as interfaces Web dos usuários (professor e aluno), o gerenciamento dos OAI e o módulo de decisão (Fuzzy) desenvolvido.

CAPÍTULO 5 – RESULTADOS

Este capítulo mostra e discute resultados finais obtidos. São apresentados os resultados da avaliação realizada na fase inicial e final.

5.1 Primeira Fase

Os usuários participantes do questionário da Turma A – Introdução a Segurança da Informação foram 5 alunos e 1 professor. As questões aplicadas aos alunos foram tabuladas e as respostas apresentadas nas Tabelas 2 e 3. A Tabela 2 mostra as avaliações qualitativas, mapeadas com notas graduadas em excelente, ótimo, bom, regular e ruim. Em todos os casos os alunos receberam dicas automáticas do ambiente conforme pré-configurado pelo professor. Destes, apenas 1 (um) aluno considerou as dicas com pouca influência no resultado, os demais consideraram como ótimo e excelente a utilização do recurso proporcionado pela interação do OA.

Destaca-se que todos os alunos desta turma já tinham conhecimento e já haviam utilizado outro ambiente virtual de aprendizado, o que traz a luz uma reflexão sobre a importância deste novo modelo de ambiente interativo para o meio acadêmico.

Tabela 2. Questões de Avaliação do OAI.

Questões	S	Não	Não solicitei
1- Você já tinha utilizado algum ambiente virtual de aprendizagem?	100% (5)	0% (0)	não se aplica
2- Você se sentiu acompanhado pelo professor durante as atividades?	80% (4)	20% (1)	não se aplica
3- Você solicitou dicas durante as atividades?	100% (5)	0% (0)	não se aplica
4- Quando precisou, o professor acompanhou sua atividade e ofereceu auxílio?	60% (3)	20% (1)	20% (1)

A interação com o professor na forma de solicitação de dicas ocorreu em 100% (5) dos casos, e estes que a fizeram, julgaram este recurso como ótimo em 80% (4) dos casos. Apenas 20% (1) dos alunos não solicitaram acompanhamento pelo professor durante as atividades, enquanto os outros 80% (4) solicitaram e foram atendidos de forma satisfatória. No total, 100% (5) dos alunos consideram o ambiente StudentWatcher e os OAI como ótimo (40%) (2) e excelente (60%) (3), vide Tabela 3.

Tabela 3. Questões Qualitativas de Avaliação do OAI.

Questões	Excelente	Ótimo	Regular
5- O que você achou da interação oferecida nas atividades?	20% (1)	80% (4)	0% (0)
6- Como você avalia a ferramenta StudentWatcher e os Objetos de Aprendizagem Interativos?	60% (3)	40% (2)	0% (0)

5.1.2 Introdução ao Hardware e a Computação

Nesta turma o professor criou um OAI no formato de questionário com 6 perguntas de múltipla escolha. Para cada pergunta foram configuradas 3 dicas, que além de serem oferecidas automaticamente após o intervalo de 1 minuto sem interação do aluno com o StudentWatcher, também poderiam ser solicitadas pelo aluno a qualquer momento.

Quando concluíram a atividade, foi aplicado um questionário de avaliação do OAI, contendo 6 questões. Os percentuais das respostas, bem como os textos das questões são apresentados na Tabela 4 e 5.

Tabela 4. Questões de Avaliação do OAI.

Questões	Sim	Não	Não solicitei
1- Você já tinha utilizado algum ambiente virtual de aprendizagem?	15% (3)	85% (17)	não se aplica
2- Você se sentiu acompanhado pelo professor durante as atividades?	85% (17)	15% (3)	não se aplica
3- Você solicitou dicas durante as atividades?	55% (11)	45% (9)	não se aplica
4- Quando precisou, o professor acompanhou sua atividade e ofereceu auxílio?	65% (13)	5% (1)	30% (4)

As avaliações qualitativas são apresentadas na Tabela 5 e da mesma forma as porcentagens de respostas encontradas.

Tabela 5. Questões Qualitativas de Avaliação do OAI.

Questões	Excelente	Ótimo	Regular
5- O que você achou da interação oferecida nas atividades?	45% (9)	30% (6)	25% (5)
6- Como você avalia a ferramenta StudentWatcher e os Objetos de Aprendizagem Interativos?	55% (11)	30% (6)	15% (3)

Diante desta análise a proposta apresentada neste trabalho possui uma avaliação positiva de 85% (17), sendo considerada como excelente e ótima. Do mesmo modo, a interação alcançada com o uso dos OAI recebeu aprovação positiva de 75% (15) dos aprendizes. Destaca-se que apenas 15% dos aprendizes já tinham utilizado outros objetos de aprendizagem em Ambientes Virtuais (AVA), contra 85% (17) que nunca tinham acessado um AVA.

Os alunos que já conheciam outros AVAs, relataram que utilizando OAI no StudentWatcher se sentiram acompanhados pelo professor, avaliando as possibilidades de interação como excelente, enquanto deste grupo em 67% (13) dos casos os alunos foram auxiliados pelo professor quando solicitado, enquanto os outros 33% (7) não solicitaram auxílio.

Dos alunos que se sentiram acompanhados (85%) (17), a avaliação do modelo de interação oferecido foi considerado como excelente e ótimo em 83% (16) dos casos, tendo os outros 17% (4) avaliados como regular.

5.1.3 Discussão

Comparando ambos os testes realizados com a Turma em pequena escala (5 alunos) e média escala (20 alunos), é possível traçar um panorama único do ambiente

StudentWatcher, de modo que a avaliação é positiva em ambos os cenários, conforme ilustra a (Figura 32).



Figura 32. Comparação da Classe A e B.

Nesta comparação entre as Turmas, é possível notar que embora nas questões 1 e 3 existem diferenças significativas, as questões principais 2 e 4 que mapeiam necessariamente o que se buscou neste trabalho, garantir o acompanhamento dos alunos, o objetivo foi atingido em ambas as classes. O sentimento de ambas as turmas, pequena e grande, em relação ao acompanhamento pelo professor durante as atividades ocorreu em média de 80% (20) dos casos. A média final dos testes realizados em ambas as turmas também apresentam resultado equivalente, demonstrando também a eficácia do ambiente na questão de transferência do conhecimento.

5.2 Segunda Fase

Foi efetuado um questionário específico para os alunos e professores das turmas em um total 05 turmas, com total de 111 alunos e 05 professores, de diversas disciplinas (Sistemas Operacionais, Arquitetura de Computadores, Sistemas da Informação, Redes de Computadores e Segurança da Informação). Foram aplicadas questões abertas e fechadas, buscando apontar os pontos fortes e fracos do ambiente, e a percepção dos alunos e professores sobre o acompanhamento durante a execução das atividades.

O resultado desse teste apontou que 100% (111) dos alunos se sentiram acompanhados pelo professor durante a execução das atividades. Além disso, eles

acharam fácil a utilização da ferramenta. A (Figura 33) mostra que 100 alunos tiveram uma interação com o professor.

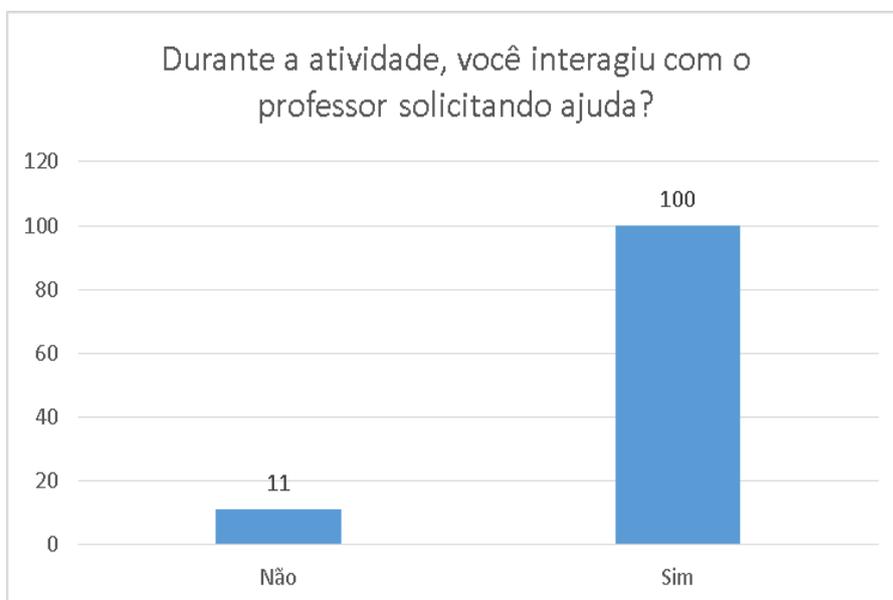


Figura 33. Interação com o professor durante a atividade.

A (Figura 34) apresenta que 109 alunos se sentiram confortáveis com o fato do professor acompanhar suas atividades durante a utilização do StudentWatcher.

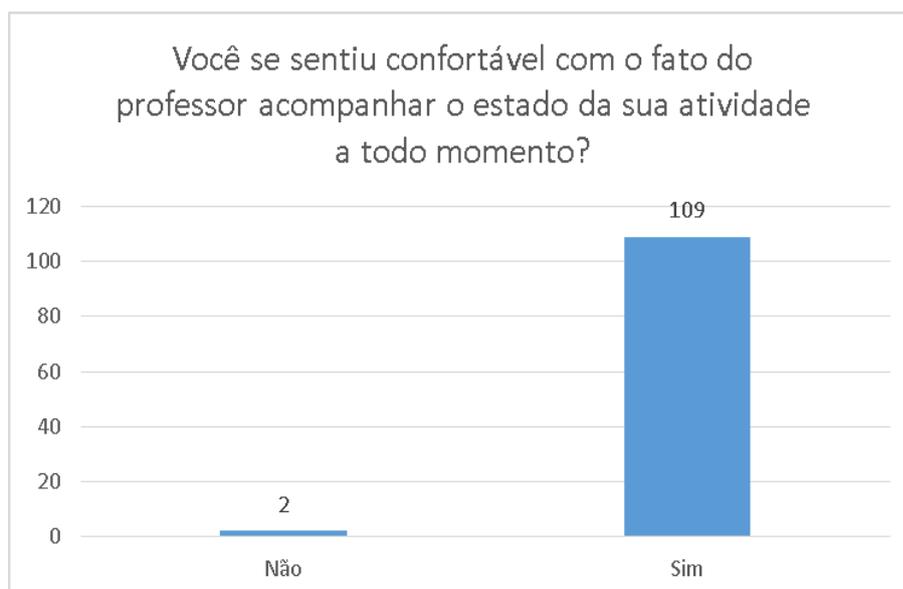


Figura 34. Alunos confortáveis com o fato do acompanhamento do professor.

A (Figura 35) mostra as avaliações qualitativas mapeadas com as opiniões dos alunos sobre as funcionalidades do StudentWatcher (Chat, Dicas, Disponibilização dos arquivos de aula e Avisos do professor). Elas foram graduadas em “Muito ruim”, “Ruim”, “Indiferente”, “Bom” e “Excelente”. Em todos os casos os alunos receberam dicas automáticas do ambiente conforme pré-configuradas pelo professor. Podemos observar que em todos os casos os resultados tiveram uma grande satisfação dos itens “Bom”, destacando a comunicação por chat como o melhor avaliado. O item “Dicas Automáticas” teve o maior índice de resultado ruim, pois em todos os testes de avaliação o professor configurou uma penalidade no exercício para utilização das dicas.

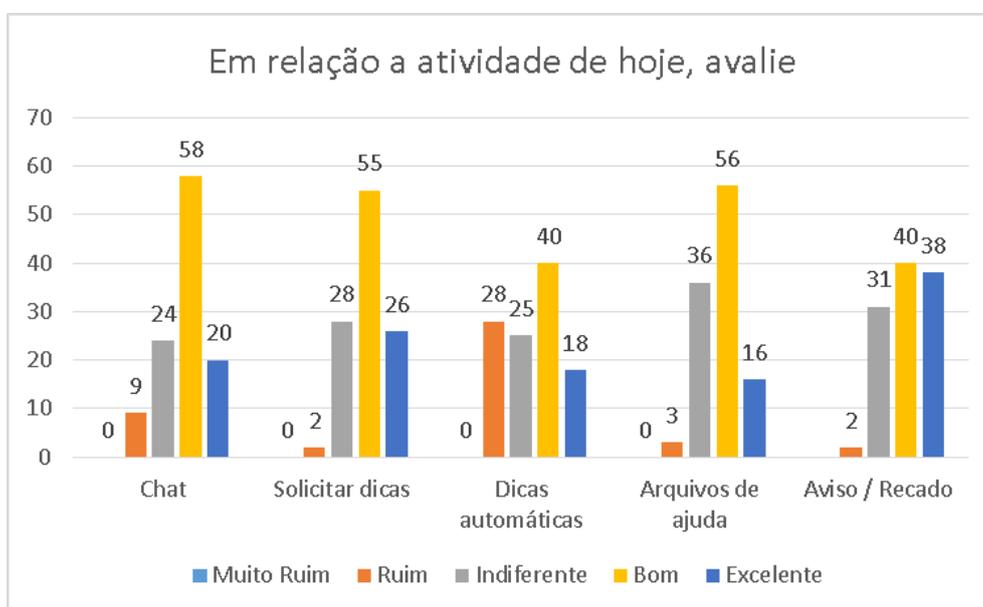


Figura 35. Opinião alunos ferramentas sobre o ambiente StudentWatcher.

A (Figura 36) mostra a opinião do aluno em relação a desempenho de notas durante a utilização do StudentWatcher.

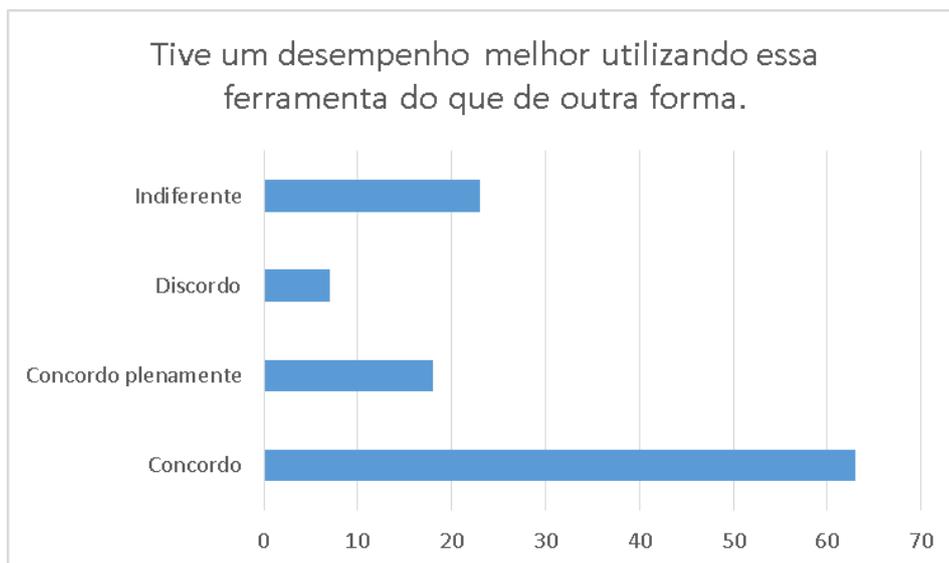


Figura 36. Opinião dos alunos em relação ao seu desempenho.

A (Figura 37) evidencia a média alçada pelos alunos após a utilização da ferramenta. Mais de 60 alunos atingiram notas mínimas e médias para execução das atividades ficando entre 6 e 7, e 30 alunos atingindo notas bem acima da média ficando entre 8 e 9, destacando o 07 aluno com notas 10. O StudentWatcher proporciona uma possibilidade de melhoria durante o processo de ensino-aprendizagem de forma muito significativa.

Médias de notas das turmas

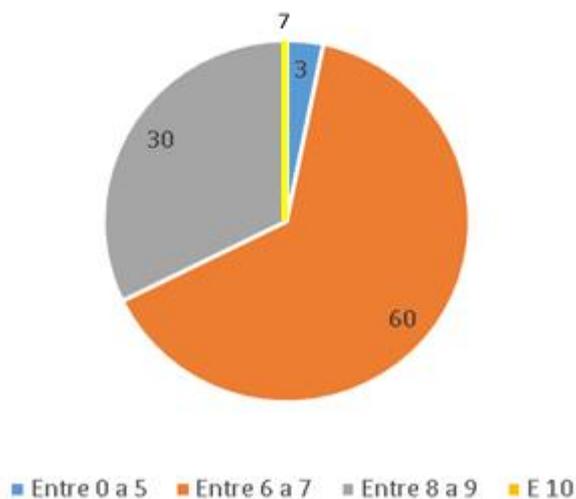


Figura 37. Quantidade alunos por categoria de médias

A Tabela 6 mostra a análise do questionário diagnóstico das questões dissertativas. Nela são apresentados os pontos fracos e fortes apontados.

Tabela 6. Pontos forte e fracos da ferramenta

Pontos fortes	Pontos Fracos
<i>“Dicas”</i>	<i>“Interface Grafica”</i>
<i>“Chat”</i>	<i>“Dicas automáticas”</i>
<i>“Conteúdo Disponível”</i>	<i>“Nenhum ponto fraco”</i>

Os 5 professores classificaram o StudentWatcher como “Bom ou Muito bom”. Eles destacaram de forma positiva a interface de monitoramento e o painel estatístico de acompanhamento dos alunos/atividades. Alguns pontos negativos foram destacados como a usabilidade da interface gráfica e melhor controle de respostas através do chat.

O trecho de texto a seguir é um exemplo de relato de professores apontando o aprimoramento do ambiente. O professor 03 sugere melhora do chat e o professor 2 propõe a possibilidade de gamificação.

“A interface gráfica e controle do chat poderiam ser melhorados, para otimizar ainda mais o tempo de resposta.” (Professor 03)

“A ferramenta poderia ter opções de atividade como módulos de nível como um game.” (Professor 02)

5.2.1 Discussão

Os dois testes, principalmente o segundo, permite traçar um panorama único do ambiente StudentWatcher, que é positivo. O principal item de destaque é compatível com os objetivos da ferramenta, que é a interação e o acompanhamento dos alunos. O Professor 01 escreveu:

“O Ambiente cumpre com o sua proposta em minha turma a efetividade do trabalho aumentou junto com o seu rendimento, me possibilitando um maior acompanhamento do momento e atividade que o aluno estava efetuando durante a aula.”
(Professor 01)

Como também o usuário Aluno 50, que aponta que a possibilidade de interação com o professor melhorou seu desempenho:

“Sempre tive dificuldade, hoje com o professor me auxiliando mais consegui desenvolver bem o exercício”. (Aluno 50)

Assim como também o aluno 35 destaca a possibilidade de aprimoramento da ferramenta

“Chat é uma ótima opção, poderia também ser utilizado video conferência para o professor não precisar ficar andando na sala”. (Aluno 35)

5.3 Considerações do capítulo.

Esse capítulo apresentou os testes realizados com o StudentWatcher. Notou-se que um dos principais objetivos da ferramenta, que é o de promover acompanhamento dos alunos foi um ponto de destaque entre os avaliadores. Porém, ficou claro também que ainda existe espaço para aprimoramento da ferramenta e da estratégia de acompanhamento.

CAPÍTULO 6 – CONCLUSÃO

Os avanços tecnológicos como os que permitem a transmissão de dados e a construção de interfaces web impulsionam o uso de computadores no processo de ensino-aprendizagem, provendo soluções para cada vez mais aproximar as distâncias entre os professores e alunos (tanto a distância física, quanto a de envolvimento nas atividades). Para que essa distância seja cada vez menor são necessárias soluções tecnológicas apoiadas por estratégias educacionais bem definidas.

Este trabalho apresentou uma estratégia computacional implementada em uma ferramenta computacional denominada StudentWachter, que possibilita o acompanhamento individual dos alunos. Nela o professor pode criar, gerenciar e compartilhar objetos de aprendizagem interativos, os OAI. Um OAI é uma expansão do padrão SCORM, com a possibilidade de monitoramento. Com isso, a ferramenta monitora o estado de aprendizagem dos alunos, que é obtida por intermédio da análise de parâmetros desses objetos. Além de definir os OAI, também criou-se uma ontologia para representar o domínio desse novo padrão, que promove a interação entre todos os envolvidos.

Estas funcionalidades projetadas nesta ferramenta são essenciais para potencializar o professor no processo de ensino-aprendizagem onde cada vez mais aprender é uma atividade fundamental e desafiadora.

6.1 Resultados

Ambas os testes executados na primeira e segunda fase do projeto, trouxeram opiniões e resultados excelentes para o aprimoramento da estratégia e ferramenta. Confirmando em sua primeira fase a usabilidade dos OAIs em sala de aula e sua capacidade de auxílio no padrão de criação das atividades pelo professor.

Na segunda fase, validamos itens conclusivos para a ferramenta e seus OAIs, aumentando a quantidade de usuários, utilizando o StudentWatcher durante os testes e focando na utilização completa do processo de atividades por diversas turmas e matéria em sala de aula.

A partir dos resultados obtidos conclui-se que esse trabalho seja uma alternativa para os educadores na busca do real sucesso do processo de ensino-aprendizagem, apropriando os alunos do conhecimento e os aplicam em contextos diferentes dos abordados durante o processo de aprendizagem.

6.2 Trabalhos Futuros

Diversos trabalhos futuros podem ser propostos a esta dissertação. Dentre eles:

- Criação de melhores estratégias educacionais e métodos para potencialização do processo de ensino e aprendizagem;
- Desenvolvimento de novas funcionalidades para melhorar os aspectos de comunicação professor e aluno;
- Novos algoritmos de modelagem para o módulo de tomada de decisão;
- Exportação dos OAI para outros padrões e integração de outros ambientes de aprendizagem;
- Novos relatórios para acompanhamento de presença, notas e evasão;
- Testes com outras turmas de alunos e professores.

REFERÊNCIAS

- ADL. 2009. SCORM 2004 4th Edition. Content Aggregation Model]. [Online] Version 1.1. August 14. Disponível em: <http://www.adlnet.gov/wp-content/uploads/2011/07/SCORM_2004_4ED_v1_1_Doc_Suite.zip>. Acessado em: 30 de maio de 2015.
- Balbino, Jaime. 2007. Objetos de Aprendizagem: Contribuições para sua genealogia, Agosto. Disponível em: http://www.dicasl.com.br/educacao_tecnologia/educacao_tecnologia_20070423.php. Acessado em: 29 de maio de 2015.
- Barbosa, J. C. 2001. Modelagem matemática: concepções e experiências de futuros professores, Tese (Doutorado em Educação Matemática) – Instituto de Geociências e Ciências Exatas, Universidade Estadual Paulista, Rio Claro, SP. pp. 253.
- Bassanezi, R. C. 2002. Ensino-aprendizagem. São Paulo, SP: Editora Contexto. pp. 388.
- Bellman, R. E., & Zadeh, L. A. 1970. Decision-making in a fuzzy environment. Management science, Volume 17, Issue 04. pp. 141.
- Bloom B. S. 1984. The search for methods of group instruction as effective as one-to-one tutoring Education Leadership, Volume 41, Issue 08. pp. 04-17.
- Bulitko & Wilkins. 2005. One-on-One Tutoring by Humans and Computers, Taylor & Francis, , pp. 383-384.
- Bulitko & Wilkins. 2005. One-on-One Tutoring by Humans and Computers, Taylor & Francis. pp. 383-384.
- Chakraborty, S., Roy, D., & Basu, A. 2010. Development of Knowledge Based Intelligent Tutoring System. Chapter 5, TMRF e-Book Advanced Knowledge Based Systems: Model, Applications & Research, Eds. Sajja & Akerkar, Volume 1. pp. 74-100.
- Chen, C. 2008. Intelligent web-based learning system with personalized learning path guidance. Computers & Education. Volume 51, Issue 2. pp. 787–814.

- Chen, C.-M. & Chen, M. C. 2009. Mobile formative assessment tool based on data mining techniques for supporting web-based learning, Volume 52. pp.256-273.
- Chen, C.-M.; Chen, M. C.; Li, Y.-L. 2007. Mining Key Formative Assessment Rules based on Learner Profiles for Web-based Learning Systems, In: 7th IEEE International Conference on Advanced Learning Technologies, Niigata. Proceedings of the 7th IEEE International Conference on Advanced Learning Technologies. pp. 584-588.
- Chen, H., Yu, C., & Chang, C. 2007. E-Homebook System: A web-based interactive education interface, Computers & Education. Volume 49, Issue 02. pp. 160–175.
- Chen, C. 2008. Intelligent web-based learning system with personalized learning path guidance. Computers & Education. Volume 51, Issue 2. pp. 787–814.
- Cohen, E., & Nycz, M. 2006. Learning objects and e-learning: an informing science perspective. Interdisciplinary Journal of Knowledge and Learning Objects.
- Downes, S. 2001. Learning Objects: Resources for Distance Education Worldwide. In International Review of Research in Open and Distance Learning. Volume. 2, Issue 1.
- Dringus, L. P.; Ellis, T. 2005. Using data mining as a strategy for assessing asynchronous discussion forums, Computer & Education, Volume. 45. pp. 141-160
- Enright, J., Chu, B. & Zaïane, O. R. 2008. Intelligent Functional Dependency Tutoring Tool, 16th International Conference on Computers in Education, Taipei, Taiwan. pp. 171-172.
- Freire, P., 2002. Pedagogia da Autonomia: saberes necessários à prática educativa, Paz e Terra, São Paulo, SP.
- Gütl, C., V. Manuel.V, G. Barrios, B. & F. Mödritscher, F . 2004. Mobile and Web Innovations in Systems and Service-Oriented Engineering. pp 399.
- Harman, K. & Koohang, A. 2007. Learning Objects: Standards, Metadata, Repositories, & LCMS. Santa Rosa-CA USA: Informing Science Press.
- Hazman, M., El-Beltagy, S. R. & Rafea, A. 2011. A Survey of Ontology Learning Approaches. In: International Journal of Computer Applications. VI. 22.

- Holanda, O., Isotani, S., Bittencourt, I. I., Elias, E. & Tenório, T. 2013. JOINT: Java ontology integrated toolkit. In: Expert Systems with Applications. pp. 6469-6477.
- Huang, Y. -M., Lin, Y.-T. & Cheng, S.-C. 2009. An adaptive testing system for supporting versatile educational assessment, *Computer & Education*, Volume 52. pp. 53-67.
- Hugo Gamboa,H. & Ferreira,V. 2003. Widam-web interaction display and monitoring. *Proceedings of the 5th International Conference on Enterprise Information Systems*, Volume 4. pp. 21-27.
- Jiang, Y., Wang, X., & Zheng, H. 2014. A semantic similarity measure based on information distance for ontology alignment. In: *Information Sciences*, Volume. 278. pp. 76-87.
- Kemezinski, A.,Costa, I. A., Wehrmeister, M. A., Hounsell, M. S. & Vahldick, A. 2012. Metodologia para Construção de Objetos de Aprendizagem Interativos, *Anais do 23º Simpósio Brasileiro de Informática na Educação*. pp. 2316-6533.
- L'Allier, J. J. 1997. A Frame of Reference: NETg's Map to Its Products. Their Structures and Core Beliefs. pp. 409-415.
- LeFrancois, G. 2001. *Of children: An introduction to child and adolescent development*, 9th ed. Belmont, CA: Wadsworth/Thomsen.
- Luz, B., Areão, A. S., Luz, R. N., Santos, R., Guimarães, M. P. & Martins, V. F. 2015. Preferências Instrucionais no Curso de Secretaria Escolar a Distância segundo os Tutores Virtuais. In: 10ª Conferência Ibérica de Sistemas e Tecnologias de Informação, 2015, Águeda. *Atas da 10ª Conferência Ibérica de Sistemas e Tecnologias de Informação*. v. 1. p. 160-166.
- Luz, B., Santos, R., Alves, B., Areão, A. S., Yokoyama, M. H. & Guimarães, M. P. 2015. An owl ontoloty for metadata of interactive leanning objects. In: 9th International Conference on e-Learning, 2015, Las Palmas de Gran Canaria. *International Conference e-Learning*. Las Palmas de Gran Canaria, 2015. p. 183-187.
- Luz, B., Santos, R. & Guimarães, M. P. 2014. Padrão de Metadados para Objetos de Aprendizagem Baseado na Interação. In: 25º Simpósio Brasileiro de Informática na

- Educação (SBIE), 2014, Dourados. 25º Simpósio Brasileiro de Informática na Educação (SBIE). p. 847-851.
- Luz, B. N., Santos, R. & Guimarães, M. P. 2015. Objetos de Aprendizagem Interativos. Revista SINERGIA – Revista Científica do Instituto Federal de São Paulo. Vol. 16, Nº 2, pp. 131-137. ISSN: 2177-451X.
- Luz, B. N., Santos, R. & Guimarães. 2015. Use of an OWL ontology for creating Interactive Learning Object. In: XLI Conferencia Latinoamericana de Informática CLEI.19 a 23 outubro de 2015, Arequipa, Peru.
- Moodle.2009. Modular Object-Oriented Dynamic Learning Environment. Disponível em:<<http://moodle.org> >. Acesso em 15 outubro. 2014.
- Nkambou,R. & Tchetagni, J. 2004. Diagnosing Student Errors in E-Learning Environment Using MPE Theory. In Proceedings of the International Conference on Web-Based Education. pp. 249-254.
- Nkambou,R. & Tchetagni,J. 2004. Diagnosing Student Errors in E-Learning Environment Using MPE Theory. In Proceedings of the International Conference on Web-Based Education. pp. 249-254.
- North. Barrie M. 2012. Joomla!™ 1.6 Guia do Operador Construindo um Site Bem sucedido em Joomla! Rio de Janeiro, Editora Alta Books,.
- Otsuka, J. L. Rocha, H. V. & Beder, D. M. 2007. A Multi-Agent Formative Assessment Support Model for Learning Management Systems, In: 7th IEEE International Conference on Advanced Learning Technologies, Niigata.
- Pádua, V. C. 2008. Ambiente de suporte a jogos WEB voltado para a área de ensino a distância. Dissertação de Mestrado, Centro de Informática, Universidade Federal de Pernambuco.
- Passos, J. O. D. 2006. Design interativo de ferramenta de manipulação de objetos de aprendizagem de ambientes virtuais de ensino à distância. Dissertação de Mestrado. Universidade Federal de Pernambuco.
- Pon-Barry, Clark, Schultz, Bratt, & Peters. 2014. Advantages of spoken language interaction in dialogue-based intelligent tutoring systems. Intelligent Tutoring Systems. pp.390-400.

- Ramos, J. L. C.. 2006. Requisitos para ferramentas de avaliação em ambientes virtuais de ensino. Dissertação de Mestrado, Centro de Informática, Universidade Federal de Pernambuco.
- Ritter, F. & Feurzeig, W. 1988. Teaching real-time tactical thinking. In J. Psozka, L.D. Massey, and S.A. Mutter (Eds.), *Intelligent Tutoring Systems: Lessons Learned*, , pp. 285-302.
- Ross, T.J. Fuzzy. 2010. *Logic with Engineering Applications*, Third Edition, Wiley; Issue 3. pp.606.
- Santos, R., Luz, B. N., Martins, V. F., Dias & Guimarães, M. P. 2014. eTutor: Um Ambiente de Aprendizagem Interativo. *Anais do 25º Simpósio Brasileiro de Informática na Educação (SBIE)*. Dourados, MT, , pp. 476-485.
- Santos, R., Luz, B. N., Martins, V. F., Dias, D. C., Guimarães, M. P, no prelo. 2015. Teachig-learning environment tool to promote individualized student assistance. In: *The 15th International Conference on Computational Science and Applications* . University of Calgary, Banff, Canada.
- Santos, R., Nogueira, B. ; Martins, V. F. & Guimarães, M. P. 2015 . Teaching-Learning Environment Tool to Promote Individualized Student Assistance. In: *Computational Science and Its Applications -- ICCSA, Banff. Lecture Notes in Computer Science*. v. 9155. p. 143-155.
- Sarris, S. 2013. *HTML5 Unleashed*. Sams Publishing, Issue 1. pp. 432.
- Schwingel, C. 2004. A produção de conteúdos no ciberespaço: sistemas de gerenciamento de conteúdos. *VII Encontro Nacional de Pesquisadores em Jornalismo*, 2009.
- Scorm. *Content Aggregation Model – CAM*. 4th Edition.
- Seters, J. R. V., Ossevoort, M. A., Tramper, J. & Goedhart, M. J. 2012. The influence of student characteristics on the use of adaptive e-learning material. In: *Computers & Education* 58. pp. 942-952.
- Svinick, M., & Mckeachie, W. J. 2012. *Dicas de Ensino: estratégias, pesquisa e teoria para professores universitários*, São Paulo: Cengage Learning. pp. 201.

- Tanenbaum, A. & Wetherall, D. 2011. *Redes de Computadores*. 5ª Edição. São Paulo: Pearson Prentice Hall.
- Tori, R. 2010. *Educação sem distância: as tecnologias interativas na redução de distâncias em ensino e aprendizagem*. São Paulo, Brasil: Editora Sena São Paulo.
- VanLehn's, K. The Behavior of. 2006. *Tutoring Systems*. *International Journal of Artificial Intelligence in Education*, Volume 16 Issue 3. pp. 267-270.
- Vygotsky, L. S. 1998. The problem of age (M. Hall, Trans.), In R. W. Rieber (Ed. *The collected works of L. S. Vygotsky: Volume 5*. pp. 187- 205.
- Vygotsky, L. S. *Thinking and speech*, In R. W. Rieber & A. S. Carton. 1987. *The collected works of L. S. Vygotsky: Volume 1. Problems of general psychology*. pp. 39-285.
- Wells, G. 1999. *Dialogic inquiry: Towards a sociocultural practice and theory of education*. Cambridge: Cambridge University Press.
- Yang,F. 2010.*The ideology of intelligent tutoring systems*. *ACM Inroads*.Volume 1. pp. 63-65.
- Yang,Z., Liu, W. 2007. *Research and development of web-based virtual online classroom*, *Computers & Education*, Volume 48, Issue 2. pp. 171–184.
- Zabala, A. 1998. *A Prática Educativa: como ensinar*. Porto Alegre: Artmed.
- Zadeh, L.A. 1965. *Fuzzy Sets*. *Information an Control*, Volume. 8. pp.338-353.
- Zaïane, O. L. & J. Towards. 2001. *Towards evaluating learners' behaviour in a Web-based distance learning environment*, In: *International Conference on Advanced Learning Technologies*,Madison, USA, pp 355-360.