

LOGOS: Caderno de Estudos e Exercícios de Lógica

Fabiane F. P. Galafassi
Programa de Pós-Graduação em
Informática na Educação (PPGIE)
Universidade Federal do
Rio Grande do Sul (UFRGS)
Porto Alegre, Brazil
fabiane.penteado@gmail.com

João Carlos Gluz
Programa de Pós-Graduação em
Computação Aplicada (PPGCAP)
Universidade do Valor
do Rio dos Sinos (UNISINOS)
São Leopoldo, Brazil
jcgluz@unisinors.br

Cristiano Galafassi
Programa de Pós-Graduação em
Informática na Educação (PPGIE)
Universidade Federal do
Rio Grande do Sul (UFRGS)
Porto Alegre, Brazil
cristianogalafassi@gmail.com

Rosa Maria Vicari
Programa de Pós-Graduação em
Informática na Educação (PPGIE)
Universidade Federal do
Rio Grande do Sul (UFRGS)
Porto Alegre, Brazil
rosa@inf.ufrgs.br

Resumo—O presente artigo tem como objetivo apresentar o Caderno de Estudos e Exercícios de Lógica (LOGOS) do ambiente de ensino Heráclito. O ambiente Heráclito auxilia estudantes a resolver vários tipos de exercícios de Lógica e disponibiliza o Caderno Eletrônico LOGOS para criar e editar fórmulas, tabelas-verdade e provas da Lógica Proposicional. Além disso, o Caderno Eletrônico LOGOS é compatível com tablets, smartphones e computadores pessoais. Atualmente ele vem sendo utilizado com alunos de graduação do primeiro e segundo semestres em currículos das áreas científicas e tecnológicas.

Index Terms—Sistema Tutor Inteligente, Lógica Proposicional, Editor de Fórmulas, Editor de Provas e Tabelas-Verdade.

I. INTRODUÇÃO

A disciplina de Lógica Formal é de suma importância para o currículo das áreas científicas e tecnológicas (genericamente agrupadas sob a sigla inglesa STEM – Science, Technology, Engineering, and Mathematics) sendo normalmente ofertada entre o 1º e o 2º semestre da graduação. Considerada fundamental na formação dos alunos, ela possibilita o desenvolvimento das habilidades de análise lógica, formalização e resolução de problemas. Habilidades essas, que por sua vez são necessárias para a compreensão dos diversos conteúdos e atividades encontradas neste currículo. Estudos empíricos indicam sérias dificuldades encontradas por alunos em assimilar os conceitos dessa disciplina. Estes estudos apontam para índices altos em termos de reprovações e principalmente desistências, ocasionando uma retenção de alunos acima do esperado. Essas desistências, em particular, tendem a ocorrer no início da disciplina, principalmente quando os conteúdos de Dedução Natural começaram a ser abordados no contexto da Lógica Proposicional [1]. Na prática, as dificuldades começam

quando conceitos como fórmula, regra de dedução e prova formal começam a ser apresentados. Objetivando contribuir na melhoria destes índices, a metodologia dialética de ensino (abordagem sócio histórica utilizada em sala de aula) foi associada a um modelo de mediação por computador, modelado em um STI, ao qual foi denominado de Ambiente Heráclito. Nesse sentido o Caderno Eletrônico LOGOS disponibilizado pelo ambiente Heráclito é composto por dois editores: o Editor de fórmulas e tabelas-verdade e o Editor de Provas da Lógica Proposicional. Os editores contam ainda com o apoio de um tutor (on-line) neste processo de resolução de exercícios. O ambiente se destaca dos demais ambientes de ensino de Lógica em três aspectos fundamentais: a) uma completa aderência aos princípios Dialéticos e Sócio Históricos na implementação dos processos de mediação e tutoria, b) um modelo de aluno adaptativo e flexível e; c) um suporte completo ao ensino dos conceitos formais da Lógica Proposicional.

II. DESENVOLVIMENTO

O ambiente Heráclito foi desenvolvido através do uso da tecnologia de agentes com fins pedagógicos. Estes agentes são responsáveis pela interação do aluno/tutor/ambiente auxiliando no desenvolvimento da resolução dos exercícios indicando caminhos corretos, incorretos e não recomendáveis durante o andamento da prova. O serviço de tutoria visa ajudar o aluno no passo a passo da resolução dos exercícios, fazendo o papel do professor. Toda interação do aluno/tutor/ambiente é monitorado e as ações registradas servem para determinar as ações do serviço de tutoria. Esse serviço usa um conjunto de estratégias de aprendizagem, elaboradas especificamente para Lógica, com base nas experiências em sala de aula. Se o aluno não conseguir avançar na realização de uma prova,

por exemplo, ele tem a opção de utilizar o botão ajuda para pedir dicas, exemplos ou qual o próximo passo a ser dado. Esta opção Ajuda pode ser acionada a qualquer momento do exercício. Todos os agentes do ambiente Heráclito foram desenvolvidos na linguagem Prolog e possuem uma interface em Java com suporte da plataforma JADE para comunicação entre eles. Estas linguagens e plataforma foram escolhidas por sua versatilidade, eficiência, portabilidade e segurança. Todo o processo de comunicação entre os agentes se dá por troca de mensagens na linguagem FIPA-ACL [2], suportadas pela plataforma de agentes JADE [3], que é compatível com os padrões da FIPA [2]. Sua Interface está escrita em Java, por questões de portabilidade, e distribuído sob a licença de software livre LGPL (Lesser General Public License). As especificações FIPA definem um modelo de referência para uma plataforma de agentes e também um conjunto de serviços fornecidos ao se conceber sistemas multiagentes interoperáveis, e o JADE (Java Agent Development Framework) é uma plataforma (em conformidade com as especificações da FIPA) para o desenvolvimento e execução de sistemas baseados em agentes de software que segue as recomendações da FIPA. Está escrito em Java, por questões de portabilidade, e distribuído sob a licença de software livre LGPL (Lesser General Public License). A estrutura das mensagens trocadas na comunicação entre agentes é baseada na linguagem ACL (Agent Communication Language) definida pela FIPA e contém campos tais como, variáveis que indicam o contexto ao qual a mensagem se refere e, o tempo limite que pode ser aguardado até a resposta ser recebida (timeout), visando suporte a interações complexas e conversas paralelas múltiplas [4]. A comunicação entre agentes no JADE é a baseada em mensagens assíncronas, ou seja, um agente que deseja se comunicar deve apenas transmitir uma mensagem para um destino identificado (ou conjunto de destinos), não existindo nenhum tipo de dependência temporal entre o transmissor e o receptor. A Fig. 1 apresenta a arquitetura do ambiente.

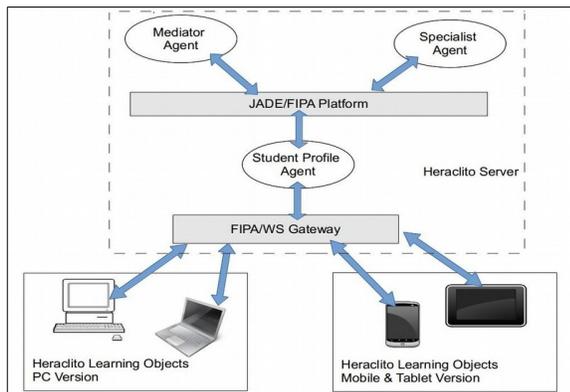


Figura 1. Arquitetura do ambiente Heráclito.

Todas as mensagens trocadas entre esses três agentes passam obrigatoriamente pelo agente mediador, que é quem media esta comunicação entre o agente perfil do aluno e agente especialista. Nesta arquitetura todas as estratégias foram de-

envolvidas com base no perfil do aluno e estão centradas no agente Mediador, o qual toma uma decisão baseado no contexto em que o aluno está inserido, fazendo uso das estratégias de aprendizagem e assim gerenciando as trocas de mensagens entre os agentes Perfil do Aluno e Especialista. Ao desenvolver a interface Web do ambiente Heráclito optou-se pela arquitetura cliente-servidor. Essa escolha se deu em virtude da segurança e desempenho que essa metodologia, associada a linguagens de programação adequadas. Com a utilização da arquitetura cliente-servidor, é possível enviar para o usuário apenas a resposta da requisição em HTML 5, CSS e Javascript, deixando toda a parte lógica do provador e toda a parte de agentes sendo processado inteiramente dentro do servidor, o que torna o sistema extremamente leve e portátil para qualquer dispositivo que possuir um navegador Web [5]. A Fig. 2 apresenta essa arquitetura:

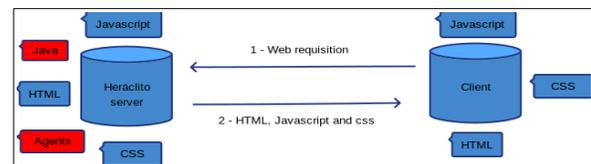


Figura 2. Representação da arquitetura cliente-servidor do ambiente Heráclito.

Visando o aumento da segurança, robustez, compatibilidade com diversos sistemas e tamanhos de tela, foi utilizada para desenvolvimento o framework Bootstrap. Este framework fornece uma gama de elementos prontos em HTML5, CSS3 e JavaScript que auxiliam na adaptação do site a diversos tamanhos de tela e diversos sistemas tornando-os responsivos, o que faz com que o objeto de aprendizagem possa, com apenas uma versão, ser distribuído para diversas plataformas diferentes alcançando um número maior de usuários.

III. APRESENTAÇÃO

O caderno Eletrônico LOGOS do ambiente Heráclito pode ser usado em modo online, com o apoio do serviço de tutoria, ou em modo off-line, como usuário visitante, mas sem a tutoria. Para o modo on-line é necessário fazer login ao ambiente. Na página inicial é possível realizar o cadastro e efetuar o login (autenticação) em poucos passos que são “nome”, “sobrenome”, “e-mail” e “senha”. A interface inicial é apresentada na Fig. 3, bem como o cadastro e acesso para uso do ambiente e tutoria [9].

Após o login ou acesso como visitante ao ambiente, o caderno Eletrônico LOGOS e seus editores são apresentados. O caderno de estudos e exercícios de Lógica oferece ao usuário o Editor de fórmulas e tabelas-verdade e o Editor de Provas da Lógica Proposicional (Fig. 4).

O ambiente também disponibiliza o capítulo 1 e 2 em formato E-book de Introdução a Lógica Proposicional dos professores João Carlos Gluz e Monica Xavier Py [7] (Fig. 5).

O editor de fórmulas e tabelas-verdade objetiva apoiar o aluno no estudo de lógica booleana e decomposição de



Figura 3. Interface inicial do ambiente Heracilito.



Figura 4. LOGOS.



Figura 5. Capítulos 1 e 2 E-book Lógica.

equações booleanas em áreas e tabelas-verdade. Este editor oferece dois tipos de opções para resolução do exercício: Exercício Personalizado e Exemplos de Exercícios (Fig. 6).



Figura 6. Interface Editor de Fórmulas e Tabelas-Verdade.

No Exercício Personalizado o aluno pode inserir a fórmula desejada e escolher entre as três formas de resolução: Decomposição de Fórmula, Avaliação de Fórmula e Construção de Tabela-Verdade. Em Exemplos de Exercícios o aluno escolhe um exemplo de exercício já pré-definido da lista e em seguida a forma em como vai resolvê-lo (utilizando as três formas de resolução mencionadas no Exercício Personalizado). O Editor de Fórmulas e Tabelas-Verdade ainda possui a opção Modo supervisionado. Nesta opção o próprio editor

de tabela verdade ou o editor de decomposição de fórmulas, verifica se o aluno está executando corretamente os passos dos exercícios. Assim, por exemplo, se o supervisor está desabilitado um aluno pode preencher um valor-verdade errado em uma tabela-verdade, deveria ser verdadeiro e ele colocou falso. Com o supervisor desabilitado nada é indicado, mas se o supervisor está habilitado o valor verdade fica em vermelho, indicando que não é o valor correto. O Editor de Provas de Lógica Proposicional tem como principal funcionalidade auxiliar na elaboração de provas de argumentos formais por meio das regras da Dedução Natural (DNLP). A edição das provas oferece dois tipos de opções para resolução do exercício (Fig. 7).



Figura 7. Interface Editor de Provas de Lógica Proposicional.

Em Prova Personalizada o aluno pode inserir suas hipóteses e conclusão e iniciar o exercício. Na interface do editor de provas o argumento deve ser digitado como mostra o exemplo acima HIPÓTESE1, HIPÓTESE 2, HIPÓTESE 3, ... HIPÓTESE N — - CONCLUSÃO no campo de entrada de dados para a nova prova e clique Começar. A seguir têm-se a Fig. 8 que apresentam as inserções das hipóteses e da edição das regras de dedução.



Figura 8. Interfaces de inserção de hipóteses e edição de regras de dedução na Lógica Proposicional.

A segunda opção é escolher um exemplo de exercício da lista com provas já pré-selecionadas que estão ordenadas por níveis de dificuldade: Provas Básicas, Provas Intermediárias e Provas Avançadas. As formas de aplicação e funcionamento de cada regra de inferência são melhores descritas e detalhadas no guia de utilização do caderno LOGOS (manual). O guia

também pode ser acessado através do ambiente quando logado [9].

IV. POTENCIAL DE APLICAÇÃO/RESULTADOS ESPERADOS E/OU ALCANÇADOS

O Caderno Eletrônico LOGOS do ambiente de ensino Heráclito, apresentado neste artigo visa ajudar alunos a aprender conceitos da Lógica Proposicional. Experimentos empíricos conduzidos com este ambiente mostram que a possibilidade de ajuda é real e que, além disso, o ambiente está sendo bem aceito pelos alunos. Os resultados destes experimentos, bem como demais produções referentes a este ambiente e seus componentes podem ser vistos em [1], [6] e [8]. O ambiente entrou em uso experimental e pode ser acessado e usado por qualquer estudante no endereço: <http://labsim.unipampa.edu.br:8080/heraclito/> [9].

V. CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Ao desenvolver e adaptar o ambiente Heráclito espera-se poder contribuir não só em um melhor entendimento dos conteúdos abordados na disciplina de Lógica, mas também diminuir o número de desistências e reprovações existentes neste contexto. Em termos de novas pesquisas várias possibilidades estão sendo estudadas, sendo uma das mais importantes à evolução do modelo de aluno. Atualmente o modelo de aluno do Heráclito está centrado nas várias formas possíveis de exercícios da Lógica Proposicional, faltando uma visão geral do processo de aprendizagem do aluno em relação aos conceitos por trás de cada tipo de exercícios e da posição destes conceitos na estrutura curricular da disciplina. Nesse sentido, um modelo bayesiano de inferência está em fase de implementação, o qual se espera ser capaz de representar o conhecimento do aluno acerca dos conceitos trabalhados em DNLP. Com isso espera-se também compreender melhor os processos que ocorrem durante as interações entre aluno/tutor/ambiente utilizando este mecanismo de inferência para identificar o nível de conhecimento real do aluno. De posse destas informações, futuramente, ainda poderá ser criado um mecanismo de estratégias de ensino-aprendizagem específicas para cada perfil de aluno encontrado.

REFERÊNCIAS

- [1] GALAFASSI, F.F.P.: "Agente Pedagógico para Mediação do Processo de Ensino-Aprendizagem da Dedução Natural na Lógica Proposicional". Dissertação de Mestrado. UNISINOS. 2012. Disponível em: <<http://biblioteca.asav.org.br/vinculos/000003/00000335.pdf>>. Acesso em 04 junho de 2018.
- [2] FIPA-ACL. "FIPA ACL Message Structure Specification". FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS. Disponível em: <<http://www.fipa.org/specs/fipa00061/SC00061G.html>>. Acesso em 04 junho de 2018.
- [3] BELLIFEMINE, Fábio; CAIRE, Giovanni; GREENWOOD, Dominic. "Developing Multi-Agent Systems with JADE". Wiley Series in Agent Technology, Series Editor: Michael Wooldridge, Liverpool University, UK. 2007.
- [4] BORDINI, R. H.; HÜBNER, J. F.; WOOLDRIDGE, M. Programming multi-agent systems in AgentSpeak using Jason. Volume 8. Editora John Wiley & Sons, 2007.
- [5] GALAFASSI, P. F. F.; SANTOS, A. V.; PERES, R. K.; VICARI, R. M.; GLUZ, J. C. Multi-plataform Interface to an ITS of Proposicional Logic Teaching. In: 13 ° International Conference on Practical Applications of Agents and Multi-Agent Systems, 2015, Salamanca. Highlights of Practical Applications of Agents, Multi-Agent Systems, and Sustainability - The PAAMS Collection, 2015. v. 524. p. 309-319.
- [6] GLUZ, J. C.; BUENO, R.; PERES, R. K.; GALAFASSI, P. F. F. Tutoria Inteligente Completa para os Conceitos Formais da Lógica Proposicional: Experimentos e Resultados. VI Congresso Brasileiro de Informática na Educação (CBIE 2017). Anais do XXVIII Simpósio Brasileiro de Informática na Educação (SBIE 2017). Disponível em: <<http://www.br-ie.org/pub/index.php/sbie/article/view/7639/5435>>. Acesso em 04 junho de 2018.
- [7] Gluz, J.C.; Py, M. (2010) Lógica para Computação. Coleção EAD. Editora Unisinos.
- [8] ISELab - Laboratório de Engenharia de Softwares Inteligentes. Projeto Heráclito: Ambiente Inteligente de Ensino de Lógica. Disponível em: <<http://obaa.unisinos.br/drupal7/?q=node/25>>. Acesso em 04 junho de 2018.
- [9] AMBIENTE HERÁCLITO. Disponível em: <<http://labsim.unipampa.edu.br:8080/heraclito/>>. Acesso em 04 junho de 2018.