

## **Tutor Inteligente para Recomendação de Atividades de Programação em um Ambiente Virtual de Aprendizagem**

**Rosemary P. Borges<sup>1</sup>, Carla Katarina de M. Marques<sup>1</sup>, Rommel W. De Lima<sup>1</sup>,  
Jorge Allende B. M. de Souza<sup>2</sup>**

<sup>1</sup>Universidade do Estado do Rio Grande do Norte (UERN) – Campus Central –  
Mossoró, RN – Brasil

<sup>2</sup>Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte (IFRN) –  
Campus Mossoró – Mossoró, RN - Brasil

{marybezerraleao, rommel.lima, carla.katarina}@gmail.com,  
jorgeallende45@hotmail.com

***Abstract.** This paper presents the preliminary results regarding the development of a context sensitive intelligent tool that aims to guide the creation of activities, converts software agents that make a recommendation based on the content addressed and the interactions registered in the Virtual Learning Environment, in addition to A Moodle module that performs the management of source code activities. Resulting in a tool that helps the teacher to easily identify as difficulties and to the students, who pass the targeted and specific activities as their needs.*

***Resumo.** Este trabalho apresenta os resultados preliminares referente ao desenvolvimento de uma ferramenta inteligente sensível ao contexto que objetiva orientar a criação de atividades, utilizando agentes de software que realizam a recomendação com base no conteúdo abordado e nas interações registradas no Ambiente Virtual de Aprendizagem, em complemento a um módulo do Moodle que realiza gerenciamento de atividades de código fonte. Resultando em uma ferramenta que auxilia o professor a identificar facilmente as dificuldades e aos alunos, que passam a desenvolver atividades direcionadas e específicas as suas necessidades.*

### **1. Introdução**

A problemática envolvendo o ensino e a aprendizagem de programação é tema de diversos estudos como apresentado por Aureliano, Tedesco e Giraffa (2016). Essa temática tem ganhado destaque principalmente no que tange as iniciativas que buscam inserir conceitos de lógica de programação desde séries iniciais. Em se tratando do ensino para a formação de profissionais na área de computação, ainda se faz presente entre os alunos a dificuldade com relação a lógica e a resolução de problemas, como afirma Mutiawani e Juwita (2014). Assim, os discentes tendem a ter maior índice de reprovação nas disciplinas de linguagens de programação, o que os leva a desmotivação.

Segundo Salomão e Watanabe (2013), para dirimir essa situação, os professores devem buscar desenvolver um conjunto de competências para manter seus alunos

motivados, dentre elas destacam-se a presteza quanto às atividades em geral e o retorno das atividades desenvolvidas.

Nesse sentido, para auxiliar o professor no desenvolvimento das suas atividades e proporcionar um retorno mais rápido aos alunos, Chaves et al. (2013) propôs um módulo para o Moodle que automatiza o processo de elaboração, submissão e avaliação das atividades de código fonte chamado Módulo de Integração com os Juízes Online (MOJO). Segundo o autor, ao utilizar essa estratégia, os alunos podem ter melhor desempenho visto que recebem um retorno rápido com relação a avaliação das atividades.

Além de garantir que os alunos tenham rápido retorno com relação a avaliação das suas atividades, pesquisadores como Perrenoud (2003) apontam que a escolha de atividades por parte do docente também pode ser considerada como um dos fatores determinantes para o bom desempenho da turma.

Assim, o presente trabalho tem por objetivo apresentar um sistema, integrado ao módulo MOJO, que utiliza conceitos de inteligência artificial através da implementação de multiagentes que tem por finalidade orientar a criação das atividades no ambiente Moodle. Para isso, o sistema multiagente realiza a recomendação de atividades através da análise do contexto da turma, com base no conteúdo abordado e nas interações registradas no Moodle.

Para apresentar este sistema, o trabalho encontra-se organizado em 6 tópicos. Além da introdução já apresentada, o tópico 2 apresenta o conceito de sistema multiagente. O tópico 3 apresenta o Módulo de Integração com os Juízes Online. O tópico 4 explana sobre trabalhos que apresentam temáticas relacionadas. O tópico 5 descreve o sistema multiagente proposto, bem como a metodologia e os resultados obtidos até o momento da escrita desse artigo. E, por fim, o tópico 6 conclui o trabalho com as considerações finais.

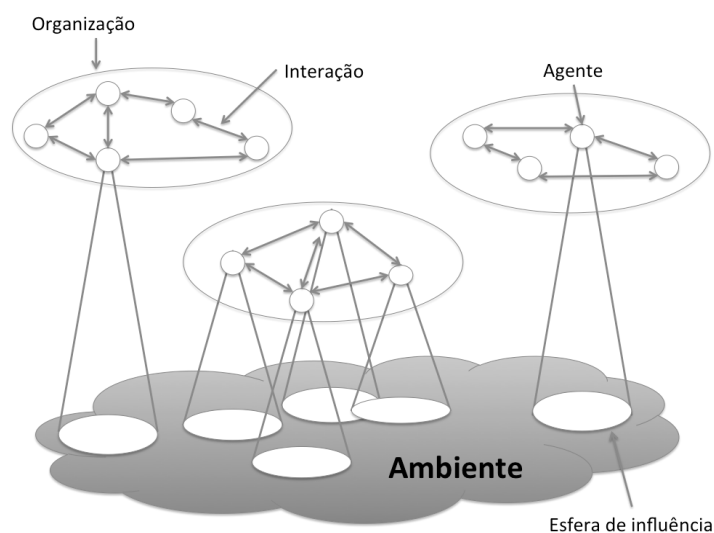
## **2. Sistema Multiagente**

Na Inteligência Artificial é frequente o uso do conceito de Agente Inteligente (AI). Segundo Russel e Norvig (2013), um AI é uma entidade autônoma capaz de interagir com o ambiente, com outros agentes, cooperando ou ainda competindo entre si e com inteligência para tomar decisões sem a necessidade de um outro sistema ou da interferência de seres humanos.

Ainda de acordo com Russel e Norvig (2013), um AI capta as percepções oriundas de um ambiente por meio de sensores e age nesse ambiente por intermédio de atuadores. Como desdobramento desse conceito, temos os Sistemas Multiagentes (SMA) que, segundo Reis (2003), são formados por agentes que possuem desempenho autônomo, porém, têm ações de caráter colaborativo. Ou seja, eles se ajudam, objetivando alcançar uma meta em comum. Na atualidade, muitos são os exemplos de competições que utilizam o conceito de SMA, como por exemplo, copas de futebol utilizando robôs, competição de venda de pacotes de viagens, dentre outras.

De acordo com Wooldridge (2009), um SMA caracteriza-se por ser um sistema onde vários agentes atuam em conjunto sobre um ambiente na busca da resolução de um problema. Um SMA conta com elementos que possuem diferentes capacidades de

percepção e ação no mundo. de acordo com Reis (2003), cada agente pertence a uma organização, onde podem ocorrer interações distintas, e estes agentes detém uma esfera de influência diferente sobre o ambiente ao qual interage, como é ilustrado na Figura 1.



**Figura 1 - Estrutura de um Sistema Multiagente.**

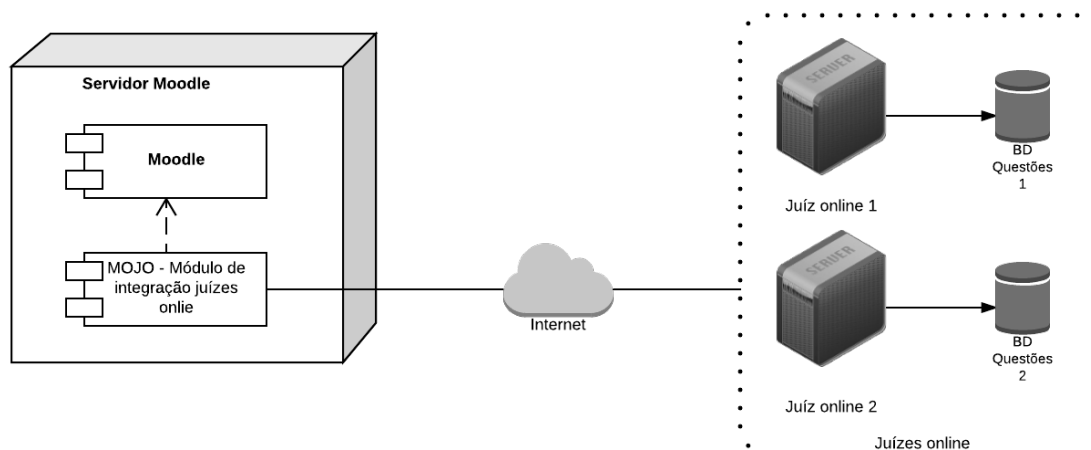
**Fonte: Adaptado de Reis (2003).**

Para Dorça (2002), os sistemas educacionais desenvolvidos utilizando-se características de SMA, tendem a ser eficazes, pois tendem a ir ao encontro da natureza da solução dos problemas de ensino-aprendizagem, que no geral são resolvidos de forma colaborativa. Mas para que isso ocorra, ainda segundo o mesmo autor, é necessário delimitar bem uma arquitetura para a aplicação, os agentes que implementam os módulos dessa arquitetura, bem como as suas interações.

### 3. Módulo de Integração com os Juízes Online

O Módulo de Integração com os Juízes Online (MOJO), desenvolvido por Chaves (2014), tem como objetivo auxiliar o docente de forma automatizada nos processos de elaboração (por possuir uma base de questões previamente criada), avaliação e fornecimento de retorno das questões respondidas aos alunos. Para que isso ocorra, a ferramenta faz uma integração entre os sistemas de Juízes Online ao Ambiente Virtual de Aprendizagem Moodle.

Na Figura 2, Chaves (2014) apresenta a arquitetura de funcionamento do MOJO, a partir da qual pode-se observar que o Moodle fornece a interface e o conjunto de funcionalidades necessárias à gestão e ao acompanhamento das atividades de programação.



**Figura 2 - Arquitetura de funcionamento do MOJO.**

Fonte: Adaptado de Chaves (2014).

Ao observar a Figura 2, nota-se mais claramente a ligação proporcionada pelo MOJO. Chaves (2013) vislumbrou, ao implementar o MOJO, a integração de dois ambientes, almejando que um venha a suprir e completar o outro, de forma que sendo integrado ao Moodle, o MOJO oferece ao professor e ao aluno, uma facilitação no retorno com relação ao código que foi desenvolvido, possíveis erros na compilação, dentre outros.

A integração Moodle-MOJO além de proporcionar a disponibilização de atividades através da plataforma Moodle e correção por juízes online, fornece base para um leque de funcionalidades que venham a automatizar o trabalho docente. Surge aí, a possibilidade da implementação, de forma integrada, de um sistema multiagente sensível ao contexto, que ofereça ao professor suporte para melhor escolha das atividades – com base nas dificuldades da turma apresentadas nos fóruns –, e consequentemente ao aluno a possibilidade de ter acesso a atividades que venham a aprimorar seu conhecimento, atendendo a suas reais necessidades.

#### 4. Trabalhos relacionados

Na busca por soluções que garantam melhor desempenho por parte dos alunos nas disciplinas de linguagens de programação, estudos apresentam diversas estratégias como simulações de maratonas/olimpíadas de programação objetivando estimular o trabalho colaborativo entre indivíduos de uma mesma equipe, a criatividade e a busca por inovações em um ambiente que ao mesmo tempo é competitivo [Piekarski 2015] e [Ramos 2015]. Estratégias como jogos e robótica são relatadas nesse contexto, visto que proporcionam práticas interativas, permitindo ao aluno aprender de forma mais ativa e lúdica [Sousa 2016], [Zanetti 2015] e [Andrade 2016].

Destacam-se também as abordagens que utilizam ferramentas destinadas a apoiar a produção de código, bem como sua correção e que buscam tornar o processo de ensino-aprendizagem mais dinâmico [Divino 2015], [Maciel et al 2012] e [Caves et al 2013]. Essas ferramentas ainda auxiliam na redução da carga de trabalho do professor e

possibilitando o retorno mais ágil aos alunos. Outros estudos, propõem o uso de Sistemas Tutores Inteligentes (STIs), como em Gonzalez e Tamariz (2014).

Ao voltar-se para o Ambiente Virtual de Aprendizagem Moodle, relatos sobre soluções são mais limitados. Estudos mais recentes buscam explorar o uso de STIs, como apresenta Porfirio (2016). Para este autor, o uso de STIs, em ambientes como o Moodle, proporciona um ambiente favorável para o desenvolvimento das atividades, garantindo assim uma otimização do tempo do professor e, conseqüentemente, uma redução da sua carga de trabalho quando comparado ao uso de técnicas tradicionais, como a verificação manual de exercícios.

Dessa forma, diversos trabalhos sugerem o uso de ferramentas inteligentes, com capacidade de aprendizado, para auxiliar o professor em tarefas que possam ser automatizadas. É recorrentes nestes trabalhos, o uso de STIs com técnicas de IA no seu projeto de desenvolvimento, além de atuarem como ferramentas auxiliares no processo de ensino-aprendizagem.

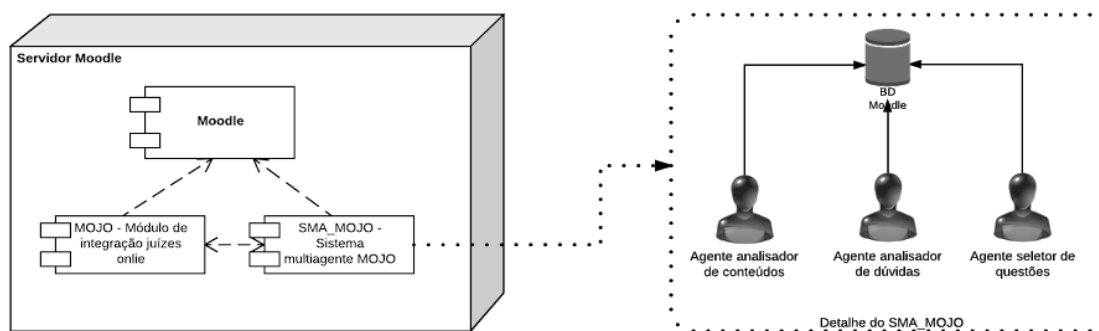
Ao realizar pesquisas na literatura observou-se que a ocorrência de STIs volta-se principalmente para abordagens envolvendo: estilos de aprendizagem, sistemas de aprendizagem adaptativos, modelo de comportamento afetivo para sistemas de tutores inteligentes, criação adaptativa de auto-avaliações, entre outros integrados ao Moodle [Palomino 2013], [Silva 2011] e [Pimentel 2013].

Percebe-se então uma lacuna com relação a sistemas que ofereçam suporte a recomendação de atividades baseado no contexto da turma. Este contexto diz respeito ao ambiente da disciplina na qual a turma está inserida. Para isso, são observados todos os materiais postados pelo professor, bem como as dúvidas registradas pelos alunos. Com essas informações o sistema terá subsídios para propor soluções ao professor e as armazena em seu domínio, que estará em constante avaliação e mudança de acordo com parâmetros de desempenho.

## **5. Sistema Multiagente integrado ao MOJO (SMA-MOJO)**

Neste trabalho, como já discorrido anteriormente, propõe-se um sistema multiagente orientado ao contexto do educando para recomendação de atividades.

Para isso, um sistema multiagente está sendo desenvolvido, contando com agentes específicos para: (I) identificar os conteúdos disponibilizados em forma de recursos; (II) identificar as principais dúvidas citadas nos canais de comunicação (fóruns e chats); (III) selecionar questões do MOJO de acordo com as informações coletadas nos agentes citados anteriormente e (IV) recomendar essas questões ao professor no Moodle. A nova arquitetura pode ser verificada na Figura 3, onde também são apresentados os agentes descritos.



**Figura 3 - Integração entre Moodle, MOJO e o SMA, destacando-se os agentes envolvidos.**

Definir um modelo sensível ao contexto para a recomendação de atividades do módulo MOJO, dentro do Moodle, exige que sejam implementados alguns objetivos em específico. São eles: (I) definição de metadados que permitam categorizar cada questão e cada conteúdo programático disponibilizado na base de dados, sendo necessário assim alteração na base de dados do Moodle; (II) identificação dos assuntos recorrentes com base nas informações obtidas observando o contexto ao qual o aluno está inserido (através da análise dos recursos disponibilizados pelo professor e canais de dúvidas); (III) seleção de questões a serem direcionadas com base nas informações obtidas no AVA e nos metadados das questões.

## 5.1 Metodologia

O SMA está sendo desenvolvido utilizando o framework JADE (Java Agent Development Framework), que disponibiliza um conjunto de ferramentas que ajudam a gerenciar sistemas multiagentes. Nela, os agentes interagem entre si através de um protocolo de comunicação previamente estabelecido facilitando assim o desenvolvimento do sistema multiagente [JADE 2016].

Para atingir o objetivo do SMA, os agentes trocam informações e acessam o banco de dados MySQL do Moodle para obter os dados relevantes sobre o contexto e sugerir as questões ao professor através da interface do MOJO. Assim, três agentes realizam diretamente essas ações:

- O Agente analisador de conteúdos (Agente Recurso): recupera da base de dados os conteúdos desenvolvidos na disciplina e analisa estes dados de forma a identificar conceitos chaves;
- O Agente analisador de dúvidas (Agente Atividade): recupera da base de dados as dúvidas registradas pelos alunos nos canais de comunicação (fórum e chat) e analisa estes dados de forma a identificar quais conceitos chaves trabalhados na disciplina;
- O Agente seletor de questões (Agente Questões): realiza a seleção das atividades que se enquadram nos conceitos chaves identificados pelos agentes descritos anteriormente, priorizando as questões que mais se aproximam das necessidades observadas.

## 5.2 Resultados preliminares

Se fez necessária a delimitação do comportamento de cada um dos agentes envolvidos, para que a colaboração entre eles fluísse, realizando da melhor forma a tarefa para a qual foram designados. Dessa forma, como já mencionado, o sistema é composto por agentes (Agente Recurso e Agente Atividade) que, no primeiro momento, analisam os dados coletados referentes ao contexto da disciplina, ou seja materiais postados e dúvidas registradas, buscando expressões que caracterizem conteúdos programáticos. No momento seguinte, através de um terceiro agente (Agente Questões), o sistema identifica as questões que estão relacionadas com esses conteúdos.

Para o processo de identificação dos conteúdos programáticos, abordados no contexto da disciplina, foi necessária a implementação de um algoritmo que identifica as palavras chave utilizadas no ambiente da disciplina em comparação com palavras previamente definidas. Sendo este processo ainda passível de melhoramento, através da utilização de ontologia.

Moodle 2.7 Você acessou como

### Pilha e Fila com Vetores e POO

Caros alunos,

Nesse tópico teremos a oportunidade de conhecer alguns conceitos relacionados as estruturas básicas de dados, como:

- Definição de Estruturas de Dados;
- Implementação de Filas e Pilhas usando Vetores;
- Modelagem da Fila e Pilha em Programação Orientada a Objeto;
- Implementação de Fila e Pilha usando conceito de POO.

**Material da Aula**

- Aula 01 - Pilhas e Filas com Vetores POO
- Aula 01 - Pilhas e Filas com Vetores usando "C" Estruturado
- Vídeos de Simulações de Filas e Pilhas

**Comunicação**

- Fórum de dúvidas

**Atividade**

- Atividade de implementação

**Figura 4a. – Destaque para os conteúdos programáticos presentes nos recursos postados pelo professor no Moodle.**

Moodle 2.7 ☰

### Fórum de dúvidas

**Dúvida sobre implementação de Fila**  
por [nome] - Terça Feira, 14 Mar 2017, 15:11

Boa tarde professor!

Preciso desenvolver um programa de controle e entrada em fila no modelo de uma pirâmide, começando do zero. Porém em fila classificatória.

Por onde devo iniciar?

[Editar](#) | [Excluir](#) | [Responder](#)

**Figura 4b. - Destaque para um dos conteúdos programáticos identificado na dúvida registrada pelo aluno.**

Além dos três agentes já citados, também faz parte do sistema um quarto agente (Agente Orientador) que possui capacidade de aprendizado. Este agente é capaz de decidir qual a frequência necessária de atualização da prioridade das questões, para isso utiliza como base a frequência histórica de interações registradas no AVA. Assim, o Agente Orientador além de identificar o momento mais adequado para reclassificar as questões, no que diz respeito a sua prioridade, também é responsável por orquestrar toda a execução do SMA, intermediando a troca de dados entre os demais agentes.

Tanto para o professor, quanto para o aluno, a atuação do SMA é transparente, de forma que esses usuários têm a percepção apenas da atividade MOJO. É durante a

configuração da atividade do tipo MOJO, que é possível visualizar a lista de questões ordenadas de acordo com a prioridade avaliada pelo SMA. Como pode ser observado no destaque da Figura 5.

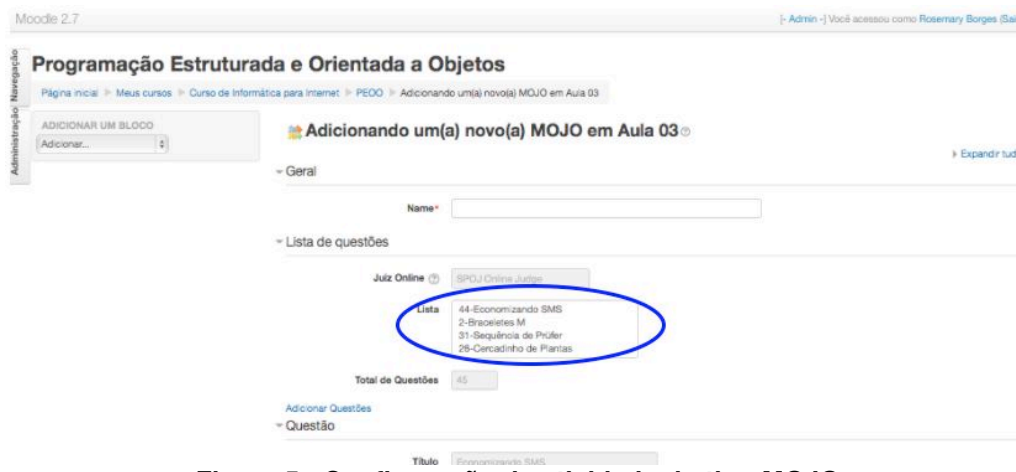


Figura 5 - Configuração de atividade do tipo MOJO.

## 6. Considerações finais

Como foi possível observar, o SMA-MOJO traz uma proposta de recomendação de atividades, sem que o professor precise concentrar esforços para identificar as principais dificuldades relatadas pelos alunos nos meios de comunicação da plataforma Moodle, permitindo assim que o docente possa se concentrar em outras atividades relativas ao ensino. Aliada a funcionalidade já em desenvolvimento, almeja-se a implementação de um sistema de classificação de questões, de forma que o nível de cada atividade seja apresentado e atualizado conforme o índice de acerto/erro dos alunos.

Vale ressaltar, que o contexto da disciplina pode ser considerada como um todo ou em partes. Isso ocorre, pois, o professor tende a optar por trabalhar sua disciplina fragmenta em sessões na plataforma, dividindo por assunto. Se assim o fizer, o SMA observará cada sessão individualmente evitando o surgimento de falsos positivos, por exemplo, no caso de uma sessão anterior que tenha abordado assunto muito polêmico, gerando inúmeras dúvidas, enquanto a sessão atual não.

A utilização de ontologias no desenvolvimento da ferramenta proporcionaria maior relação entre os termos, melhor organização, e facilitaria a implementação.

Planeja-se ainda a realização de um experimento em um contexto real. Isso possibilitará mensurar, estatisticamente, o quanto esta ferramenta pode contribuir para o processo de ensino-aprendizagem. Por fim, após a validação, a ferramenta será disponibilizada para a comunidade de usuários/desenvolvedores do Moodle.

## Referências

- Andrade, J. R. B. et al. (2016) “Uma Proposta de Oficina de Desenvolvimento de Jogos Digitais para Ensino de Programação”. Anais dos Workshops do V Congresso Brasileiro de Informática na Educação (CBIE), [s.l.], p.1127-1136.
- Aureliano, V. C. O.; Tedesco, P. C. de A. R.; Giraffa, L. M. M. (2016) “Desafios e oportunidades aos processos de ensino e de aprendizagem de programação para



- iniciantes”. In: Congresso da Sociedade Brasileira de Computação, 24., Porto Alegre. Anais do WEI - 24º Workshop sobre Educação em Computação. Porto Alegre: CSBC. p. 2066 - 2075.
- Chaves, J. O. M. (2014) “Uma ferramenta de apoio ao processo de ensino-aprendizagem em disciplinas de programação de computadores por meio da integração dos Juizes Online ao Moodle”. 2014. 80 f. Dissertação (Mestrado em Ciência da Computação). Universidade do Estado do Rio Grande do Norte. Universidade Federal Rural do Semiárido. Mossoró, RN. 2014.
- Chaves, J. O. M. et al. (2013) “MOJO: Uma Ferramenta de Auxílio à Elaboração, Submissão e Correção de Atividades em Disciplinas de Programação”. In: Congresso da Sociedade Brasileira de Computação, 21., Maceió. Anais do WEI - 21º Workshop sobre Educação em Computação. Maceió: CSBC.
- Divino, A. (2015) “Algo+ - Um app para o auxílio na aprendizagem de programação” . Anais dos Workshops do Iv Congresso Brasileiro de Informática na Educação (CBIE), [s.l.], p.372-376, 26 out. 2015. Comissão Especial de Informática na Educação.
- Dorça, F. et al. (2002) “Um Sistema Inteligente Multiagente para Educação à Distância”. In: Congresso da Sociedade Brasileira de Computação., 2002, Florianópolis. Anais... . Florianópolis: CSBC.
- Gonzalez, S. M.; Tamariz, A. del R. (2014) “Integração de uma Metodologia de Ensino Presencial de Programação com um Sistema Tutor Inteligente”. Revista Brasileira de Informática na Educação, [s.l.], v. 22, n. 02, p.16-30, 23 nov. Comissão Especial de Informática na Educação.
- Jade. (2017) “JAVA Agent DEvelopment Framework”. <http://jade.tilab.com/>.
- Maciel, D. L. et al. (2012) “Análise de similaridade de códigos-fonte como estratégia para o acompanhamento de atividades de laboratório de programação”. Revista Novas Tecnologias na Educação. [s.l.], v. 10, n. 03, [s.p.].
- Mutiawani, V.; Juwita. (2014). “Developing e-learning application specifically designed for learning introductory programming”. In: Information Technology Systems and Innovation (ICITSI). International Conference, Bandung.
- Palomino, C. E. G. (2013) “Modelo de Sistema Tutorial e Inteligente para Ambientes Virtuais de Aprendizagem baseado em Agentes”. 2013. 92 f. Dissertação (Mestrado) - Curso de Programa de Pós-graduação em Ciência da Computação, Universidade Federal de Santa Catarina, Florianópolis, 2013.
- Perrenoud, P. (2003) “Sucesso na escola: só o currículo, nada mais que o currículo”. Cadernos de Pesquisa, nº 119.
- Piekarski, A. E. et al. (2015) “A metodologia das maratonas de programação em um projeto de extensão: um relato de experiência”. Anais dos Workshops do Iv Congresso Brasileiro de Informática na Educação (CBIE), [s.l.], p.1246-1254, 26 out. 2015. Comissão Especial de Informática na Educação.
- Pimentel, N. A. (2013) “Sistema Tutor Inteligente Híbrido com Personalização Estruturada pelo Método das Diferenças Finitas”. 108 f. Dissertação (Mestrado) -

Curso de Mestrado em Engenharia de Produção e Sistemas, Pontifícia Universidade Católica de Goiás, Goiás, 2013.

Porfirio, A. J.; Direne, A. I.; Maschio, E. (2016) “Modelagem Genérica de Aprendizizes com Ênfase em Erros na Aquisição de Habilidades em Programação de Computadores”. Anais dos Workshops do V Congresso Brasileiro de Informática na Educação (CBIE 2016), [s.l.], p. 1198-1207.

Ramos, T. et al. (2015) “Ensino de programação para Olimpíada Brasileira de Informática”. Anais do XXI Workshop de Informática na Escola (WIE 2015), [s.l.], p.122-126. Comissão Especial de Informática na Educação.

Reis, L. P. (2003) “Coordenação em Sistemas Multi-Agente: Aplicações na Gestão Universitária e Futebol Robótico”. 487 f. Tese (Doutorado) - Curso de Engenharia Electrotécnica e de Computadores, Faculdade de Engenharia da Universidade do Porto, Porto, 2003. Cap. 10. Disponível em: <[https://web.fe.up.pt/~niadr/PUBLICATIONS/thesis\\_PhD/PhD\\_LuisPauloReis.pdf](https://web.fe.up.pt/~niadr/PUBLICATIONS/thesis_PhD/PhD_LuisPauloReis.pdf)>. Acesso em: 01 de setembro de 2016.

Russel, S. e Norvig, P. (2013) “Inteligência Artificial: Uma abordagem moderna”. 3a. ed. Rio de Janeiro: Elsevier Editora.

Salomão, L. F. S.; Watanabe, R. H. (2013) “Evasion in distance education courses offered by an organization of Brazilian Army: Actions to reduce”. In: International Conference on Information Systems and Technology Management (CONTECSI), 10., 2013, São Paulo. Anais... São Paulo: 2013. Disponível em: <<http://www.contecsi.fea.usp.br/envio/index.php/contecsi/10contecsi/paper/download/3539/2061>> Acesso em: 11 de abr. de 2016.

Silva, V. dos S. (2011) “Tutores Inteligentes como Mediador para o Ensino e a Aprendizagem”. Revista Tecnologias em Projeção, Brasília, v. 2, n. 1, p.29-33, jun.

Sousa, L. M. G. et al. (2016) “Ensino de Programação em Robótica Móvel no Ensino Fundamental e Médio”. Anais dos Workshops do V Congresso Brasileiro de Informática na Educação (CBIE), [s.l.], p.399-402, dez. 2016.

Wooldridge, M. (2009) “An Introduction to Multiagent Systems”. Editora Wiley. England.

Zanetti, H.; Oliveira, C. (2015) “Práticas de ensino de Programação de Computadores com Robótica Pedagógica e aplicação de Pensamento Computacional”. Anais dos Workshops do Iv Congresso Brasileiro de Informática na Educação (CBIE), [s.l.], p.1236-1245, 26 out. 2015. Comissão Especial de Informática na Educação.