

## Um Modelo de Tutor Virtual para Aulas Baseadas em Missões de Programação de Jogos Digitais

Alan de Oliveira Santana<sup>1</sup>, Thiago Reis da Silva<sup>2</sup>, Eduardo Henrique da Silva Aranha<sup>1</sup>

<sup>1</sup>Programa de Pós-Graduação em Sistemas Computacionais - PPgSC  
Universidade Federal do Rio Grande Norte – UFRN

<sup>2</sup>Instituto Federal de Educação, Ciência e Tecnologia do Maranhão – IFMA  
Campus São João dos Patos – MA

alandeoliveirasantana@gmail.com, thiago.reis@ifma.edu.br,  
eduardoaranha@dimap.ufrn.br

**Resumo.** *A demanda por profissionais de computação e a importância desse conhecimento na formação dos mais variados tipos de profissionais do século XXI levam à busca por novos métodos de ensino de programação. Neste sentido, este trabalho propõe um modelo de tutor virtual para direcionar o aluno no aprendizado de programação de jogos digitais baseado em um formato de aulas do tipo missão. Um modelo foi desenvolvido para representar o funcionamento pedagógico de tutores virtuais para o formato de missões. Em seguida, um protótipo foi desenvolvido e avaliado por estudantes do ensino médio. Avaliações realizadas pelos alunos indicaram resultados positivos e promissores no uso de tutores virtuais que implementem o modelo de aula apresentado.*

**Abstract.** *The demand for computer professionals and the importance of this knowledge in the training of the most varied types of professionals of the 21st century lead to the search for new methods of teaching programming. In this sense, this work proposes a virtual tutor model to guide the student in the learning of digital game programming based on a mission type class format. A model was developed to represent the pedagogical functioning of virtual tutors for the format of missions. Then a prototype was developed and evaluated by high school students. Evaluations carried out by the students indicate positive and promising results in the use of virtual tutors who implement the presented classroom model.*

### 1. Introdução

A tendência de queda na procura por cursos superiores da área de Tecnologia da Informação (TI) [Barbosa, Ferreira e Costa 2014] contrasta com o aumento progressivo na demanda por profissionais na área [Computação Brasil 2007]. Segundo a *Association for Computing Machinery* (ACM), 120.000 vagas de empregos anuais serão abertas até 2020 apenas nos Estados Unidos, citando ainda que, a carência em profissionais formados em Ciência da Computação pode tornar-se a maior interferência no desenvolvimento da indústria tecnológica [National 2012].

São diversos os fatores desmotivacionais nos cursos de TI. As dificuldades enfrentadas nas disciplinas iniciais dos cursos de programação, dificuldades de compreensão dos conceitos abstratos, falta de tempo para se dedicar aos estudos, bem como a complexidade inerente ao uso de linguagens de programação associado à sua semântica e sintaxe [Barbosa, Ferreira e Costa 2014].

Estes problemas motivacionais aumentam a carência e importância do profissional da computação, como citado no *Model Curriculum for K-12 Computer Science* [CSTA 2011], onde a grande maioria das profissões do século 21 demandam uma compreensão da Ciência da Computação. Profissionais de arte e entretenimento, comunicação, saúde, entre outros, precisam ter conhecimentos de Computação, enquanto ciência, na busca por soluções de problemas de áreas diversas, bem como na construção dessas soluções [França, Silva e Amaral 2012].

Para contornar os altos índices de evasão dos cursos de TI, principalmente os cursos de programação, entidades como a Sociedade Brasileira de Computação (SBC) defendem o ensino de programação desde o ensino fundamental, com a introdução na educação básica de conceitos de Ciência da Computação como forma de aprimorar o raciocínio computacional das crianças [França, Silva e Amaral 2012]. Marques *et. al.*, (2011), relatam a importância do ensino de programação nos níveis básicos e intermediários. Segundo os autores, se este conhecimento fosse introduzido desde cedo, mitigariam as dificuldades encontradas posteriormente pelos alunos. Outro ponto importante no ensino de programação desde as séries iniciais, está na crescente necessidade de diversas profissões que utilizam seus conceitos diretamente ou indiretamente [Melo, Costa e Batista 2013].

Em busca de alternativas para as dificuldades motivacionais apresentadas, o ensino de programação e outros conceitos de Ciência da Computação podem ser trabalhados através de jogos digitais educativos [Sá, Teixeira e Fernandes 2007] e do ensino de programação de jogos [Medeiros, Silva e Aranha 2013]. Porém, os professores do ensino básico ainda não possuem domínio nessas áreas para trabalhar o tema com seus alunos.

Neste contexto, a Educação a Distância (EaD) aliada ao ensino de programação de jogos surge como uma alternativa educacional democrática, com o intuito de possibilitar o acesso ao conhecimento a pessoas de qualquer lugar do Brasil, conduzindo o aluno à autonomia em seus estudos [Oliveira e Bottentuit Junior 2016].

A EaD é o processo de ensino e aprendizagem, mediado por tecnologias, onde, tanto professores, como alunos, estão separados fisicamente e/ou temporalmente [Moore e Kearsley 2011]. Essa separação, por outro lado, acaba também gerando problemas de desmotivação. Para reduzir esse problema, o uso de tutores virtuais surge como uma alternativa para se ter um personagem que represente o professor presente junto ao aluno, guiando o aluno em seus estudos.

Neste contexto, o objetivo deste trabalho é apresentar uma proposta de um tutor virtual aplicado a cursos de programação de jogos digitais, permitindo que o estudante aprenda em seu ritmo, tornando as aulas mais dinâmicas e produtivas. Portanto, este artigo encontra-se organizado da seguinte forma: na Seção 2 são reportados os trabalhos relacionados, na Seção 3 é apresentada a fundamentação do estudo, a Seção 4 o modelo de tutor virtual proposto, na Seção 5 é reportada a implementação do tutor virtual, na Seção 6 resultados da avaliação do tutor, e, por fim, na Seção 7 as considerações finais.

## **2. Trabalhos Relacionados**

Tutores virtuais podem ser aplicados para diversos propósitos, permitindo que cursos de EaD utilizem metodologias que permitam os alunos serem auxiliados por estes sistemas, melhorando a compreensão dos conteúdos e tornando o ensino mais lúdico. A literatura apresenta uma quantidade elevada e diversificada de trabalhos sobre tutores virtuais e áreas de aplicações na educação.

Helms *et al.* (2015) apresentam uma modelagem de agentes pedagógicos em um sistema multiagente, proposto para auxiliar a aprendizagem de matemática no ensino médio. Este tutor atua auxiliando a resolução de equações algébricas, reforçando os conteúdos visto em sala de forma lúdica.

O trabalho de Serenelli (2015) demonstra um tutor virtual aplicado ao ensino através de objetos como infográficos, animações e interatividades com o tutor a fim de tornar as aulas mais divertidas e conseqüentemente engajar o aluno. Oliveira *et al.* (2016) apresenta um sistema tutor virtual para ser aplicado na ferramenta de ensino Moodle. Este sistema tutor virtual está ligado ao ensino de algoritmos adaptado a dispositivos móveis e utiliza a linguagem AIML para realizar diálogos com o usuário semelhante a um *chat*.

O diferencial do trabalho apresentado neste artigo em relação aos demais consiste principalmente na formalização de um modelo pedagógico baseado no formato de missões para viabilizar o desenvolvimento de tutores virtuais voltados para o ensino de programação de jogos digitais.

### 3. Fundamentação

Para se propor um tutor virtual que auxilie o aprendizado de programação de jogos digitais, é necessário definir o formato pedagógico das aulas a serem mediadas pelo tutor. Essa seção apresenta uma breve visão sobre o formato de aulas *online* escolhido para ser utilizado no tutor proposto neste trabalho, chamado de Formato de Missões, proposto no trabalho de Silva *et al.* (2015).

A proposta apresentada por Silva *et al.* (2015) é o apropriado para o modelo de tutor proposto, uma vez que ele analisou e comparou diferentes formatos de aula (textos, aulas *online* e aulas gravadas em diferentes formatos [Silva e Aranha 2015]), chegando na proposta de aula baseada em formato de missão. Segundo Silva *et al.* (2015), videoaulas longas e com conteúdo complexos apresentam resultados pouco eficientes relativos ao aprendizado que torna-se entediante e pouco atrativo. A metodologia apresentada por Silva é uma alternativa às aulas baseadas em vídeos longos (20 a 60 minutos, por exemplo).

Para tornar as aulas mais atrativas e produtivas, o Formato de Missões apresenta um conjunto de videoaulas (ou outros formatos de mídia) como missões no estilo quebra-cabeças, no qual um vídeo de abertura apresenta um jogo digital e desafia o aluno a desenvolvê-lo (missão). Neste ponto, o aluno não terá acesso aos conteúdos em estilo passo a passo, como em um tutorial obtendo ao final o jogo proposto. Ao invés disso, terá acesso a recursos (vídeos, *checklists* e outros) que auxiliam o aluno no desenvolvimento de sua missão.

Estes recursos apresentam em geral conceitos necessários para se desenvolver cada parte do jogo, como mover um personagem, mover um inimigo, ganhar pontos, entre outros. Outra característica dos recursos é que estes são desenvolvidos de forma genérica, apresentando conceitos que são utilizados em diversos jogos, permitindo a reutilização destes recursos em diferentes missões.

Os vídeos utilizados como recursos de uma missão possuem tempo de duração em torno de 60 segundos a 6 minutos, e podem ser acessados pelo aluno na ordem que lhe for mais conveniente, tornando os vídeos como peças de um quebra-cabeça que ao final resulta no desenvolvimento do jogo proposto pela missão. Resultados do estudo de Silva *et al.* (2015) indicam que, embora promissora, o formato de missões proposto por

ele pode deixar o aluno confuso, se não houver o apoio de um tutor para guiar a navegação entre os diversos recursos disponíveis para uma missão.

#### 4. Modelo de Tutor Virtual Proposto

Considerando aulas de ensino de programação de jogos digitais no formato de missões, desenvolvido por Silva *et al.* (2015), um fluxo pedagógico foi formalizado para o tutor virtual. Este fluxo pode ser visualizado através do diagrama da Figura 1. Cada aula consiste em uma missão (criação de um jogo digital) e um roteiro de atividades e recursos é utilizado para guiar o estudante no desenvolvimento de sua missão.

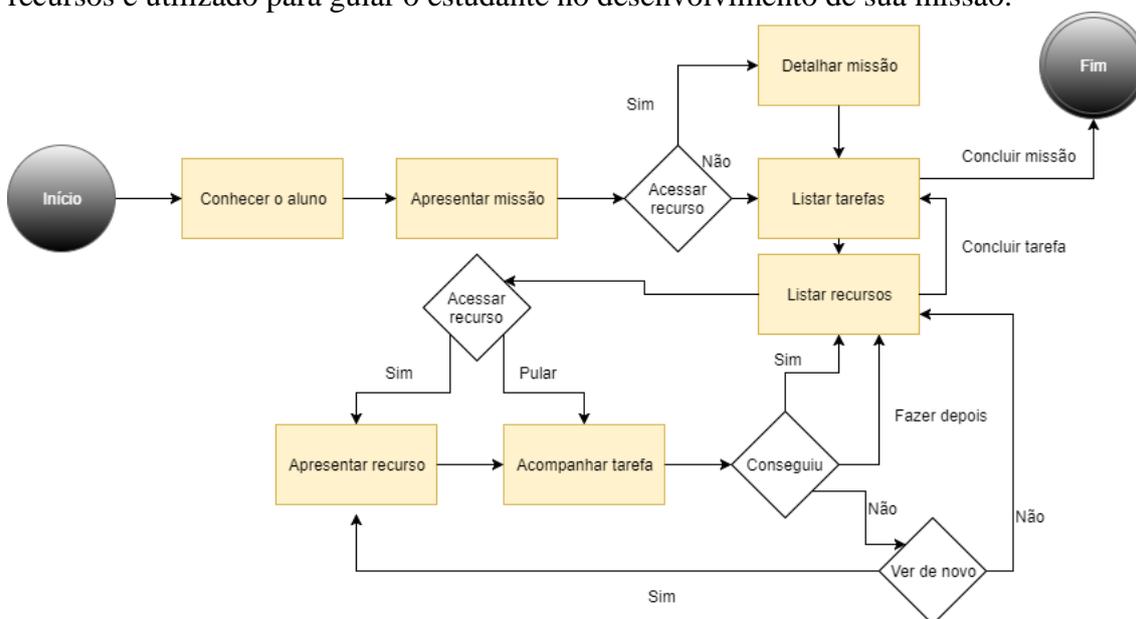


Figura 1. Diagrama de fluxo da aula do tutor virtual.

De acordo com esse fluxo, as atividades principais do tutor virtual são as seguintes:

- **Conhecer o aluno:** nesta primeira atividade, o tutor virtual irá se apresentar e coletar informações sobre o aluno, como seu nome, seu estado afetivo, entre outros;
- **Apresentar missão:** em seguida, o tutor irá apresentar ao aluno um jogo digital e desafiar o aluno a desenvolver um jogo similar. Com isso, buscará fazer com que o aluno se comprometa com a aula através da aceitação da missão, sentindo-se assim desafiado a completá-la;
- **Detalhar missão:** nessa atividade, o tutor deverá apresentar o jogo a ser desenvolvido em maiores detalhes. Através de recursos como vídeos explicativos, o tutor irá apresentar as características gerais do jogo que deverá ser desenvolvido;
- **Listar tarefas:** o tutor irá apresentar as tarefas que o aluno precisa executar para que sua missão esteja completa. Dentre as tarefas, temos por exemplo, criar o personagem, definir cenário, implementar o inimigo, criar obstáculos, entre outras;
- **Listar recursos:** cada tarefa irá requerer conhecimentos e habilidades específicas do aluno. Por isso, para cada tarefa o tutor irá listar os recursos (vídeos, textos e outros) necessários que o aluno deverá acessar de forma a poder concluir a tarefa em questão;
- **Apresentar recurso:** uma vez que o aluno selecione um determinado recurso, o tutor deverá apresentar esse recurso, preferencialmente apresentando

questionários de avaliação para avaliar se o mesmo aprendeu o conteúdo, etc.; e

- **Acompanhar tarefa:** para motivar o aluno, o tutor deve pedir para o usuário aplicar o que foi aprendido com o recurso de vídeo que acabou de ver. Ele deve acompanhar o andamento da tarefa, perguntando se tem dúvidas, apresentando FAQ, verificando se o aluno precisa ver novamente o recurso, entre outros.

Pode-se observar que o percurso de aprendizado do aluno é não linear, apresentando ciclos a partir da apresentação do roteiro de aula. Esta não linearidade objetiva permitir ao aprendiz acessar os conteúdos em seu ritmo de aprendizagem, devido a possibilidade de rever o que foi visto anteriormente e pular assuntos que já sejam de seu conhecimento.

## 5. Implementação de um Tutor Virtual

Para se avaliar o modelo de tutor virtual apresentado na seção anterior, foi realizada a implementação de um protótipo de tutor virtual utilizando o modelo apresentado, com funcionalidades restritas, visando validar o modelo utilizado. O protótipo segue o fluxo ilustrado na Figura 1, permitindo o usuário decidir qual recurso deseja acessar no desenvolvimento de cada missão.

O tutor implementado possui uma aparência virtual de robô e simula um apresentador de conteúdos educativos, buscando promover o engajamento do aprendiz durante a aula *online*. O protótipo foi desenvolvido em HTML5 em uma página dinâmica que carrega os conteúdos a serem apresentados pelo tutor virtual, além dos recursos e a lógica pedagógica do sistema. O sistema foi desenvolvido para aulas de programação de jogos digitais utilizando videoaulas e a metodologia formato de missões.

O protótipo foi desenvolvido para ser acoplado a ferramentas de ensino *online* ou para o ensino presencial. No ensino *online*, a ferramenta é capaz de pegar as informações básicas do usuário como nome e *id* da base de dados do sistema em que está acoplado. No caso do ensino presencial, ao iniciar o sistema é pedido para o usuário informar seu nome. Estes dados são importantes para o *log* do sistema que armazena nome, *id* da seção e ação do usuário. Em seguida, o aprendiz é direcionado para a tela principal de aceitar missão, buscando o comprometimento do aprendiz com a aula. Com base no fluxograma da Figura 1, esta é a primeira atividade, Conhecer o aluno.

A próxima etapa do fluxo é Apresentar missão (etapa 2). Esta apresenta o objetivo da aula e uma visão geral do jogo a ser desenvolvido, e pede por fim para o usuário apertar um botão de aceitar missão, visando comprometê-lo em sua missão.

Ao clicar no botão “*Aceitar Missão*”, o aluno é direcionado para uma tela com duas opções: a primeira leva para recursos com mais detalhes da missão (etapa 3, Detalhar missão), caso o aluno não saiba exatamente as regras do jogo que será desenvolvido; a segunda opção leva para a tela de sumário das tarefas a serem desenvolvidas (etapa 4, Listar tarefas), pulando a etapa 3. Um exemplo de tela é mostrado na Figura 2.

Entrando na etapa 3, a lista de tarefas necessárias para o desenvolvimento do jogo é apresentada. O aluno pode então clicar e acessar mais detalhes de cada tarefa, na ordem de execução que desejar. Ao se clicar em uma tarefa, a lista de recursos digitais (etapa 5, Listar recursos) que podem auxiliar o aprendiz a completar esta etapa da missão, além de permitir que o aluno volte para a etapa anterior através do botão “*Concluir missão*” como demonstrado na Figura 3.



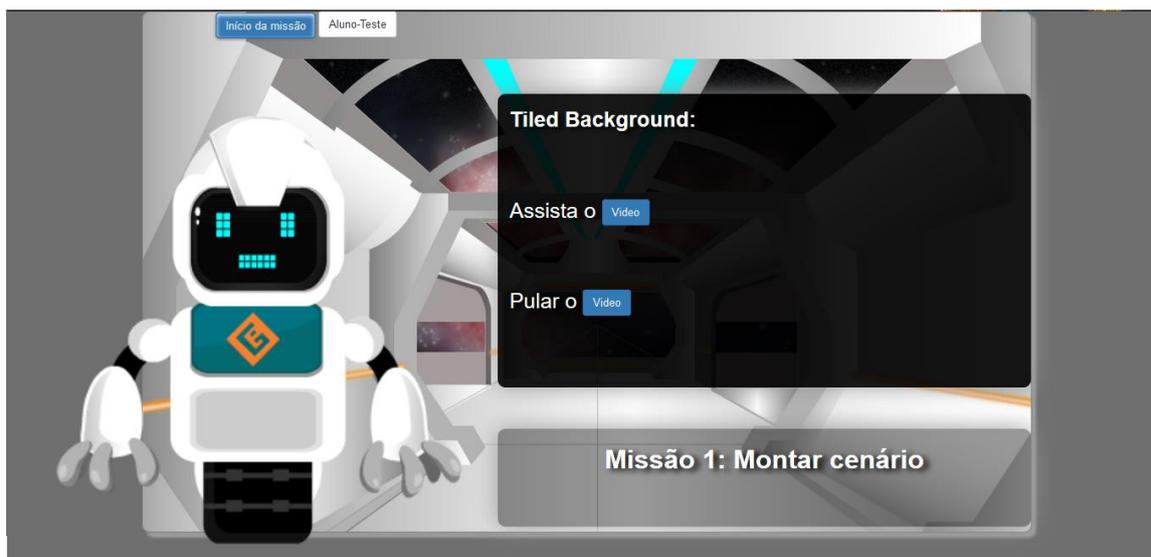
Figura 2. Roteiro da aula/missão.

Ao selecionar uma das opções da etapa 5, o usuário é direcionado para o recurso (vídeo, etc.) referente à seleção (etapa 6, Apresentar recurso). O Tutor não apenas guia, mas busca também motivar o aprendiz a continuar com a aula. Além da aparência do personagem, perguntas como: “*Você deseja ver o vídeo?*” Ou “*Você deseja pular este conteúdo?*” Objetivam, além de permitir identificar padrões através da análise do *log*, deixar o caminho de aprendizagem mais flexível e objetivo.



Figura 3: Tópicos das missões.

A Figura 4 apresenta a tela de decisão do aprendiz em visualizar ou não um determinado conteúdo de aprendizagem.



**Figura 4: Tela de decisão das missões.**

A Figura 5 apresenta a maneira que a ferramenta motiva a prática dos conteúdos vistos. Com isso, o usuário responde se conseguiu ou não executar aquela etapa, ou preferiu deixar para fazer depois (etapa 7, Acompanhar tarefa). Estes dados também são armazenados no *log* e são valiosos para a análise do perfil do aprendiz.

O sistema de *log* registra todas as ações realizadas pelo aprendiz durante o uso do sistema de aula, inclusive das ações nos vídeos, como pausar e mover a barra de progresso. Com isso, pode-se levantar dados sobre como o aprendiz assiste os vídeos, se pula muitas partes e quais partes, se nunca assistiu todo o conteúdo ou qual os vídeos mais assistidos, permitindo conseqüentemente que o sistema indique conteúdo dentro do perfil apresentado pelo usuário além de permitir que os desenvolvedores de conteúdos identifiquem problemas e possam desenvolver ou melhorar as estratégias de produção.



**Figura 5: Incentivo a praticar logo após cada explicação.**

## 6. Avaliação do Tutor

Esta seção descreve os resultados de um estudo realizado para avaliar a aplicação do protótipo de tutor virtual junto a alunos do ensino médio.

### 6.1. Contexto do estudo

O estudo consistiu em utilizar o tutor com 20 estudantes do 1º ano do ensino médio integrado do Instituto Federal de Educação, Ciência e Tecnologia do Maranhão (IFMA), Campus São Raimundo das Mangabeiras. Estes participantes foram selecionados através de convite a alunos dos turnos da manhã e tarde e que tiveram interesse em desenvolver jogos, mesmo sem terem tido acesso ainda a matérias que envolvessem conceitos de programação.

Para este estudo, foram desenvolvidas duas aulas, ou seja, duas missões: desenvolver o jogo arkanoid; desenvolver o jogo nave espacial. Ambas as aulas são compostas por um conjunto de 14 pequenos vídeos, sendo 13 sobre o desenvolvimento e 1 introdutório. Cada vídeo possui um tempo de duração que varia de 30 segundos a 6 minutos. Por exemplo, um vídeo aula foi sobre como montar o cenário de um jogo, utiliza o comportamento *Sprite* da ferramenta Construct 2 (ferramenta utilizada no estudo). O *Sprite* é utilizado para colocar objetos no jogo. O vídeo desse comportamento possui 3 minutos.

O estudo foi realizado em três dias. O primeiro dia teve como objetivo explicar como seria o estudo e apresenta uma visão geral do Tutor e de como os alunos deveriam prosseguir durante o desenvolvimento dos jogos. A principal preocupação era instruir os alunos a seguir o desenvolvimento corretamente, caso contrário, eles poderiam tentar explorar o tutor para outro propósito, que não fosse o foco do estudo. Além disso, o condutor do estudo recomendava que os alunos fizessem perguntas sempre que quisessem e que deveriam dirigir suas dúvidas exclusivamente ao condutor do experimento, sem fazer perguntas aos outros alunos próximos. Já o segundo e terceiro dia foram as aulas de desenvolvimento de jogos em si, pois nem todos puderam executar o tutor no mesmo dia. Embora a execução do estudo tenha sido presencial, visando poder observar o comportamento dos alunos, as atividades foram realizadas de forma a simular um ambiente EaD, onde o aluno deveria tentar ao máximo desenvolver sua missão de forma autônoma.

### 6.2 Resultados Alcançados

Os dados do experimento foram avaliados através do uso de questionários que foram aplicados nos 2 dias de aulas de desenvolvimento dos jogos. Os dados apontaram que os estudantes se sentiram motivados e confiantes a desenvolver jogos através da utilização do tutor proposto. Um dos motivos ligados a esta confiança e motivação foi, segundo os estudantes, a experiência mais divertida, simples e prática, promovida pelo formato de missões, que permite que os alunos fiquem mais atentos às explicações devido os vídeos serem divididos em partes curtas, e pela motivação e estímulos promovidos pelo tutor.

Todavia, algumas limitações também foram apontadas, como o fato do tutor não possuir sistema de dúvidas e respostas, além dos vídeos não apresentarem um passo a passo claro do desenvolvimento do jogo, e sim das funcionalidades do jogo, necessárias para a implementação de tarefas do jogo.

Entre as respostas dos alunos podemos destacar segundo os critérios de aprendizagem, praticidade e limitações as seguintes respostas. Com relação a Aprendizagem, um aluno destacou: *“no meu ponto de vista o tutor facilita muito a aprendizagem, por que o aluno tem mais motivação em aprender assistindo as vídeoaulas por partes ficando mais atento à explicação”*. Outro aluno reportou: *“A vantagem do Tutor Virtual é que se não entender alguma parte, o sujeito pode ver apenas esta parte”*. Outro aluno abordou que: *“A utilização dos vídeos em partes*

*específicas faz com que o aluno se mantenha atento, e mais seguro, na hora de realizar as atividades, também facilitando o entendimento e aprendizagem”.*

No aspecto de Praticidade, um aluno afirmou: “*Muito prático para aprender, bem explicada e bastante dinâmico*” e outro aluno destacou: “*Fácil aprendizagem e Rápida e prática*”. Com relação a Limitações alguns alunos apresentaram os seguintes aspectos: “*Impossibilidade de interação como o Tutor*” e “*Não poder fazer algumas perguntas*”.

## **7. Considerações Finais**

Tutores virtuais apresentam-se como importante ferramenta de apoio para o ensino à distância. No caso do ensino de programação de jogos digitais, o uso de tutores virtuais pode facilitar a inserção deste tópico no ensino básico, uma vez que os professores das escolas ainda não estão preparados para ministrar esse tipo de conteúdo em suas aulas. Nesse contexto, este trabalho contribuiu com uma proposta de modelo pedagógico para o desenvolvimento de tutores virtuais para o ensino de programação de jogos digitais. O modelo proposto foi instanciado através da prototipação de um tutor virtual e foi avaliado com estudantes do ensino médio, cujos resultados indicam um efeito positivo na atratividade da aula com a inserção do tutor virtual implementado no modelo proposto.

Como trabalhos futuros, além de realizar estudos de avaliação do nível de aprendizagem ou de engajamento dos alunos, temos a implementação de sistemas tutores mais elaborados, seja a extensão do tutor apresentado neste trabalho, como no desenvolvimento de outros formatos de interação (ex: *chatbots*), mas que sigam o modelo pedagógico baseado no formato de missões. Pretende-se ainda desenvolver funcionalidades adicionais para acompanhamento do trabalho do aluno, como mecanismos de detecção de dúvidas baseado em FAQ, fóruns e *chats* com tutores e com os demais alunos. Por fim, mecanismos de afetividade podem ser desenvolvidos para tornar o tutor virtual mais atrativo para o aluno.

## **Agradecimentos**

Os autores agradecem a CAPES pelo suporte parcial a esta pesquisa e, em especial, ao Programa de Pós-Graduação em Sistemas Computacionais – PPgSC/UFRN, por toda infraestrutura oferecida.

## **Referências**

- Barbosa, A. A.; Ferreira, D. Í.; Costa, E. B. (2014) “Influência da linguagem no ensino introdutório de programação”. In: Simpósio Brasileiro de Informática na Educação (SBIE).
- Computação Brasil. (2007) “Mercado de Trabalho em Computação: Oportunidades e Desafios”. Computação Brasil, Porto Alegre: Sociedade Brasileira de Computação, n. 25, Ano VIII. Mar-Mai, 2007
- CSTA – Computer Science Teacher Association. (2011) “CSTA K-12 Computer Science Standards”. CSTA Standards Task Force. ACM Association for Computing Machinery.
- França, R. S.; Silva, W. C.; Amaral, H. J. C. (2012) “Ensino de ciência da computação na educação básica: Experiências, desafios e possibilidades”. In: XX Workshop sobre Educação em Computação.

- Helms, F. K.; Loreto, A. B.; Adamatti, D. F.; Buss, C. L.; Ferreira, A. A. (2015) “Modelando um sistema Tutor Multiagentes para auxiliar na aprendizagem da Matemática”. In: *Scientia Plena*, v. 11, n. 8, 2015.
- Marques, D. L.; Costa, L. F. S.; Silva, M. A. A.; Rebouças, A. A. D. S. (2011) “Atraindo Alunos do Ensino Médio para a Computação: Uma Experiência Prática de Introdução a Programação utilizando Jogos e Python”. In: *Anais do Workshop de Informática na Escola (WIE)*.
- Medeiros, T. J.; Silva, T. R.; Aranha, E. H. S. (2013) “Ensino de programação utilizando jogos digitais: uma revisão sistemática da literatura”. In: *Revista Novas Tecnologias na Educação (RENOTE)*, v. 11, n. 3.
- Melo, L.; Costa, T. K.; Batista, A. C. D. (2013) “Pense bem: proposta e desenvolvimento de jogo digital para ensino de computação na educação básica”. In: *Simpósio Brasileiro de Informática na Educação (SBIE)*.
- Moore, M. G.; Kearsley, G. (2011) “Distance education: A systems view of online learning”. Cengage Learning.
- National. (2012) “A National Talent Strategy. Ideas for securing U.S competitiveness and economy growth”. In: *Microsoft Report*. Disponível em: <<http://www.microsoft.com/en-us/news/download/presskits/citizenship/msnts.pdf>>. Acesso em: set. 2014.
- Oliveira, L. C.; Bottentuit Junior, J. B. (2016) “The Distance Education as an Alternative to Reflective Training in Learning Process”. In: *Brazilian Journal of Education, Technology and Society*, v. 9, n. 2, p. 204-216.
- Oliveira, L. C.; Paschoal, L. N.; Mozzaquatro, P. M.; Cassenote, M. R. S.; Techio, G. B.; Quaresma, C. R. T.; Antoniazzi, R. L. (2016) “Agente pedagógico para o ensino de algoritmos computacionais integrado no ambiente moodle adaptado a dispositivos móveis”. In: *Revista Interdisciplinar de Ensino, Pesquisa e Extensão*, v.3, n.1.
- Sá, E.; Teixeira, Jeane, S. F.; Fernandes, C. T. (2007) “Design de Atividades de Aprendizagem que usam Jogos como princípio para Cooperação”. In: *Simpósio Brasileiro de Informática na Educação (SBIE)*, p. 539-549.
- Serenelli, F. (2015) “Aprendizagem multimídia e conteúdos interativos: uma experiência de pesquisa experimental em Mídia-Educação em escolas de Montevideo”. In: *Revista Tempos e Espaços em Educação*, v. 8, n. 17, p. 177-132.
- Silva, T. R.; Aranha, E. H. S. (2015). “Online Game-based Programming Learning for High School Students – a Case Study”. In: *45th Annual Frontiers in Education – FIE*, Texas, EUA.
- Silva, T. R.; Aranha, E. H. S.; Oliveira, W.; Fernandes, K. (2015) “Investigando dois formatos de videoaulas de programação de jogos digitais para alunos do ensino médio”. In: *Anais do Workshop de Informática na Escola (WIE)*.