Game Logic: Um jogo para auxiliar na aprendizagem de lógica de programação.

Dorgival Netto[1], Luiz Mario Medeiros[2], Daniel de Pontes[2], Edilson de Morais[2]

[1]Instituto Federal do Mato Grosso do Sul (IFMS) – Campus Corumbá-MS

[2]Faculdade Maurício de Nassau - João Pessoa – PB

dorgival.silva@ifms.edu.br, luiz.nativo@hotmail.com, {daniel.queiros789, edilsonmorais16}@gmail.com

Abstract. *The difficulty of structuring cognitive thinking together with the inherent resistance of algorithmic learning and its basic structure of construction affects a large part of the students of higher courses in the areas of exact as well as the students of other levels of instruction. For this process, this article presents Game Logic, a game focused on teaching-learning block algorithms, based on the cognitive principle that images are better memorized and their use is duly proven.*

Resumo. *A dificuldade de estruturação do pensamento cognitivo aliado à resistência implícita do aprendizado de algoritmos e sua estrutura básica de construção acomete grande parte dos alunos de cursos superiores nas áreas de exatas como também estudantes de outros níveis de instrução. Para auxiliar neste processo, este artigo apresenta o Game Logic, um jogo voltado para o ensino-aprendizado de algoritmos através da programação em blocos, partindo do princípio cognitivo de que imagens são melhor memorizadas tendo seu uso devidamente comprovado.*

1. Introdução

Em um ambiente onde a interação entre homem e computador torna-se cada vez mais intrínseca, o estudo de seu funcionamento e também o desenvolvimento de ferramentas para estas máquinas já não é mais uma tarefa reservada a programadores apenas, nem empresas do seguimento. Com um número cada vez maior de pessoas que estão iniciando seu estudo de programação, o desenvolvimento de ferramentas que se utilizem do pseudocógico para a facilitação e compreensão de algoritmos, independente da linguagem de programação a ser adotada futuramente pelo estudante, tem maior demanda e utilidade social, incluindo-o em ambientes outrora restritos a uma pequena classe de pessoas.

As disciplinas introdutórias à programação nos cursos das áreas de Ciência da Computação são extremamente importantes para o desenvolvimento do estudante no decorrer do curso. Pois apresentam aos alunos uma base fundamental para o desenvolvimento em qualquer linguagem. A lógica de programação deve ser compreendida tomando-se por base fundamental que a máquina entende códigos de

maneira estruturada e que a declaração dos comandos a serem executados é crucial para a resolução do problema ao qual o algoritmo dispõe-se a resolver.

O processo de ensino e aprendizagem de lógica de programação é desafiador, apesar dos numerosos esforços de pesquisa para melhorar esse processo [Neves & Coello, 2006] [Costa et al., 2010], devido a capacidade de abstração e aos conceitos lógicos-matemáticos que esta disciplina exige.

O provimento de métodos educacionais cada vez mais lúdicos e inovadores são inseridos no contexto educacional. Segundo Falkembach (2006) toda a atividade que incorporar a ludicidade pode se tornar um recurso facilitador do processo de ensino e aprendizagem. Muitos artigos têm sido publicados abordando o uso de jogos no processo de ensino-aprendizagem como, por exemplo, ProGame [Dantas et al. 2011], Takkou [Barbosa; Fernandes; Campos 2011] e KLouro [Silva e Dantas, 2014]. Estes jogos apresentam os conceitos de lógica de programação no nível de algoritmo. Os desafios consistem em um nível mais avançado ou relacionados a uma linguagem de programação específica como Python, por exemplo.

No processo de elaboração dos jogos educacionais é muito importante que sejam seguidos os princípios de *game design* [Crawford 1982]. Por se tratar de um jogo educacional, aspectos pedagógicos também foram levados em consideração. Assim utilizamos conceitos de *game-based learning* (GBL) [Prensky 2003]. Aprendizagem Baseada em Jogos ou GBL, segundo Tang, Hanneghan e El-Rhalibi (2009), faz referência a uma abordagem de aprendizagem inovadora derivada do uso de jogos de computador que possui valor educacional ou diferentes tipos de aplicações de software que usam jogos computacionais para ensino e educação, GBL tem como finalidade o apoio à aprendizagem, a avaliação e análise de alunos e melhoria do ensino.

Levando em conta os diferentes aspectos inerentes a tal problema, este artigo descreve o Game Logic, um jogo desenvolvido para dispositivos móveis Android, baseado nos princípios do *game design,* que usa programação em blocos na forma de fluxogramas para auxílio ao processo de ensino a aprendizagem de lógica de programação através da resolução de desafios lógicos relacionados a instruções sequenciais. Os desafios são colocados frente ao jogador de forma simples, inicialmente e, sendo intensificados ao serem vencidos. Gradualmente, mais blocos de comandos são liberados para uso na resolução dos problemas.

O presente artigo encontra-se dividido em sete seções. Após o apontamento das motivações, a seção 2 descreve o uso dos jogos educacionais, *game design* e gamificação. Na seção 3 são apresentados de alguns softwares e sites relacionados ao aprendizado de algoritmos. A seção 4 trata da proposta do Game Logic. A seção 5 mostra a avaliação do jogo proposto. Por fim, na seção 6 são apresentadas as considerações e os trabalhos futuros.

2. Fundamentação Teórica

2.1. O uso de jogos educacionais

A tecnologia em suas diferentes apresentações vem contribuindo significativamente para o bem-estar social como um todo. O potencial pedagógico dos games vem sendo cada vez mais explorado na educação e hoje são considerados importantes ferramentas no processo de aprendizagem [Moraes, 2009]. Embora possam ser similares a vários

outros gêneros de jogos, o seu diferencial é a ênfase no aprendizado, pois são especificamente projetados para ensinar as pessoas algum assunto, expandir conceitos, reforçar o desenvolvimento ou auxiliar o entendimento à medida que a brincadeira evolui, afirma Cuperschmid (2008).

O uso e efetividade dos jogos para fins educativos vêm sendo objeto de pesquisa de muitos estudiosos e, como afirma Moratori (2003), os jogos educativos computadorizados são atividades inovadoras onde as características do processo de ensino-aprendizagem apoiado no computador e as estratégias de jogo são integradas a fim de alcançar um objetivo educacional determinado.

A aprendizagem de forma prazerosa é tangível, através do emprego dos métodos corretos, tomando como base aspectos como cultura, idade, grau de instrução dentre outros fatores e pode gerar resultados relevantes, tornando o processo imensamente produtivo. Deve-se considerar veementemente que a concepção da arquitetura do jogo deve ser voltada para o aprendizado involuntário, conforme aponta Johan Huizinga:

> No jogo há beleza, harmonia, ritmo, que inspiram fascínio, tensão, alegria e divertimento, mas há também ordem e as regras estabelecidas devem ser seguidas por todos. Uma vez quebradas as regras, destrói-se a ilusão do jogo. [Huizinga, 1998, p.l4].

Sendo assim, Moratori (2003) aponta que uma vez estabelecido e obedecido o sistema de um jogo, aprender pode tornar-se tão divertido quanto brincar e, nesse caso, aprender torna-se interessante para o aluno e passa a fazer parte de sua lista de preferências.

2.2 *Game design* e gamificação

Schell (2008) define *game design* como o ato de decidir o que um jogo deveria ser. Brathwaite (2009), amplia esta definição afirmando que *game design* é o processo de criar a disputa e as regras de um jogo. Para um bom *game design* é preciso criar objetivos os quais o jogador sinta-se motivado a alcançar e regras que o jogador precisa seguir ao fazer escolhas significativas em prol desses objetivos. De acordo com Dicheva et. al. (2015), os princípios de *game design* cuja aplicação é mais bem-sucedida em sala de aula são: status visível, engajamento social, *feedback* instantâneo e liberdade de escolha.

A Gamificação (do inglês, *gamification*) da aprendizagem e instrução é descrita em Kapp (2012) como aqueles métodos e estratégias baseadas em jogos para a formação e a educação. Ou seja, o uso de jogos como mecanismo para fazer da aprendizagem ou instrução mais divertida e mais profunda. *Gamification* está relacionada com a ideia de engajamento, narrativas, autonomia e significado. A gamificação corresponde ao uso de mecanismos de jogos orientados ao objetivo de resolver problemas práticos ou de despertar engajamento entre um público específico [Vianna et al, 2003, p.13]

A mecânica de um sistema gamificado é composta de ferramentas que, quando utilizadas corretamente, prometem um retorno significativo dos usuários [Zichermann e Cunningham, 2011]. As principais técnicas utilizadas para a gamificação são:

- **Pontos** - Os pontos podem ser capturados pela experiência ou pela habilidade. **Pontos de experiência (XP)** é o mais importante sistema de pontos, não serve como moeda de troca no sistema, todas as ações do jogador geram XP e uma vez adquirido ele não os perde. O objetivo principal é observar, classificar e guiar o

usuário sobre o seu comportamento. **Pontos de habilidade (*skill*)** são atribuídos a tarefas específicas, que não estão relacionados às atividades principais do sistema. O usuário pode optar por realizá-las ou não, e são um conjunto bônus de pontos que permitem que o usuário ganhe experiência ou recompensas pela realização.

- **Níveis** - indicam o progresso do usuário dentro do ambiente. Os **níveis de jogo** devem manter a percepção de que há progresso na história do jogo, proporcionando o engajamento do usuário. A cada nível avançado, habilidades aprendidas devem ser reforçadas ou novas sejam aprendidas, gerando aprendizagem gradativa. Os níveis servirão como motivação para os usuários. **Níveis de dificuldade,** geralmente, são organizados em três níveis de dificuldade: fácil (para iniciantes com desafios mais simples), médio (proporciona um sistema com dificuldade e desafios equilibrados) e difícil (mais indicado para usuários experientes, pois possuem desafios com maior complexidade).
- **Rankings** - seu propósito principal é a comparação entre os jogadores/usuários envolvidos. Serve como uma forma de visualizar a progressão dos usuários dentro do ambiente e gera um senso de competição entre eles.
- **Desafios e Missões** - são os elementos que orientam os usuários sobre as atividades que devem ser realizadas dentro de um ambiente.
- **Regras** - um sistema gamificado pode ser composto por um conjunto de regras que definem como o usuário pode utilizar o ambiente, como esse ambiente funciona, o que é ou não permitido, etc. As regras servem para limitar as ações dos usuários e tornar o sistema gerenciável (Kapp, 2012).
- **Narrativa** - Através da narrativa pode-se fazer com que o usuário apresente um comportamento esperado frente a um contexto. Histórias são utilizadas para transmitir informações e guiar pessoas, a combinação do sistema com a história cria uma experiência interativa que engaja o usuário (Kapp, 2012).

Esse conjunto de técnicas provenientes dos jogos, segundo Nakamura e Csikszentmihalyi (2002), aumentam significativamente o prazer em realizar uma atividade, consequentemente tornando o aprendizado mais fácil e significativo para o aluno.

3. Softwares Relacionados

Com o objetivo de facilitar o aprendizado daqueles que buscam o conhecimento referente aos algoritmos, são disponibilizadas atualmente ferramentas que geralmente destinam-se a apenas uma linguagem de programação o que muitas vezes limita o estudante, quando não o faz abandonar os estudos devido às dificuldades encontradas durante o processo, desmotivando-o. A aprendizagem através de pseudocódigo deve ser motivada e levada ao centro do aprendizado e consequentemente a uma abrangência muito maior de conhecimento para o estudante. Realizou-se uma pesquisa de softwares relacionados à área, através dos quais citamos: Blockly Games[1], Robocode[2], Snap![3].

[1] https://blockly-games.appspot.com/

O Blockly Games é um site que disponibiliza 8 desafios diferentes sendo que dentro de cada desafio há 10 níveis de dificuldade, apresentado um nível de ludicidade considerável, onde ao final de cada desafio concluído é mostrado o código na linguagem JavaScript da montagem dos blocos utilizada para a solução do problema.

O Robocode é um software que utiliza as linguagens de programação .NET (C#) ou JAVA permitindo programar um robô virtual em batalhas multijogador. É possível utilizar classes e objetos para implementar uma inteligência artificial no robô.

O Snap! é um site que se assemelha ao Blockly, mas possui um nível de dificuldade superior para resolução dos problemas apresentados, possui uma biblioteca maior de comandos permitindo a construção em um maior nível de complexidade.

A proposta do Game Logic é um aplicativo móvel Android para auxiliar no aprendizado de pseudocódigos através de blocos de fluxo, não sendo necessário conexão com a internet, pois a ferramenta é instalada no aparelho do usuário. Tem-se desafios com os quais apresentam-se as instruções básicas para uso dos blocos necessitando de raciocínio lógico. Não há o ensino de uma linguagem em especial, pois pretende-se apresentar aqui uma ferramenta que instrua o jogador o obter a lógica necessária a programar em qualquer linguagem, não o limitando a apenas uma. Os desafios à medida em que são vencidos terão seu nível de dificuldade incrementados e consequentemente mais blocos serão apresentados ao usuário ajudando-o a superar os desafios.

4. Proposta

O Game Logic é um aplicativo voltado para estudantes de múltiplas faixas de idade, contemplando-os do ensino fundamental e médio até os períodos iniciais de cursos superiores da área de exatas, a exemplo de Engenharia e Tecnologia da Informação.

O aplicativo propõe o ensino da lógica de programação através da programação em blocos, fazendo com que o ato de aprender algo novo torne-se mais motivador e prazeroso, aliado a mobilidade que a plataforma Android oferece, podendo o estudante praticar sempre que houver tempo hábil. O intuito principal é desenvolver o raciocínio do estudante, apresentando de maneira ilustrativa como os principais comandos inerentes a maioria das linguagens de programação funcionam. O Game Logic conta com vários desafios que são propostos ao estudante em etapas, sendo que a cada novo desafio vencido o nível de dificuldade também é aumentado. Para que haja um maior incentivo e também para atender aspectos importantes que todo jogo deve conter. À medida que cada desafio é superado o nível de dificuldade é incrementado, sendo que sempre que se tornar necessário um novo bloco de comando será apresentado ao usuário auxiliando-o a superar o desafio, e incentivando-o a utilizar sempre soluções elegantes, no que diz respeito à montagem do fluxo de código para resolução do problema. Os desafios buscam desenvolver o censo de criatividade do estudante, frente a problemas relativamente simples. O jogo começa apresentando a tela inicial de opções (Figura 1 (a)) direcionando o estudante a outras partes do jogo, destacando o botão Conquistas (Figura 1 (b)) onde são registrados no serviço Google Play Games os avanços feitos no

---

[2] http://robocode.sourceforge.net/

[3] http://snap.berkeley.edu/

jogo, e na opção Jogar (Figura 1 (c)) abre-se a janela de acesso os a seleção desafios do jogo que são estruturados em 3 níveis, sendo que cada nível possui 12 desafios.

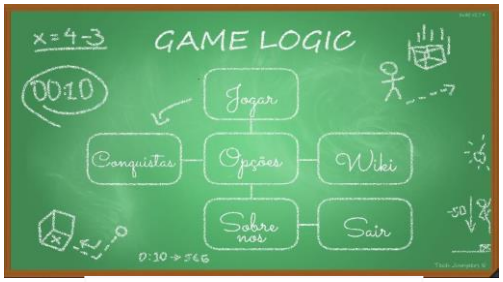

Figura (a) Tela Inicial

Figura (b) Conquistas

Figura (c) Seleção de Fase

Figura 1 Telas iniciais do *Game Logic*

Os desafios são liberados à medida que o atual é superado, sendo assim ao iniciar o jogo pelo primeiro desafio, ele logo será orientado sobre o intuito da aplicação e direcionado a executar os primeiros comandos dentro da ferramenta na tela de boas-vindas (Figura 2 (a)), após arrastar e soltar os comandos no local indicado (Figura 2 (b)), o jogador é direcionado a executar o fluxo por ele criado e após concluir o desafio (Figura 2 (c)) é promovido de fase (Figura 2 (d)), dando continuidade ao processo.

Figura (a) Tela de boas vindas

Figura (b) Criação fluxo com blocos

Figura (c) Tela de conclusão do desafio

Figura (d) Início da segunda fase

Figura (2) Telas da visão interna do jogo

O aplicativo além de contar com os comandos básicos de direcionais, também possui os comandos principais que são pertinentes a qualquer estrutura de algoritmo de qualquer linguagem de programação que são as sequenciais, de decisão e repetição como mostra a (Figura 3) onde encontram-se os comandos presentes implementados até o presente momento. A medida em que os desafios são superados mais blocos de controle irão ser adicionados, para que gradualmente os estudantes familiarizem-se com as estruturas que formam um algoritmo. Também foi implementado um ranking interno para que assim o estudante possa ver pontuações anteriores na mesma fase, o que auxilia no processo de automotivação o que segundo Bandura (1993) e Schunk (1995), a crença firme dos estudantes em sua auto eficácia para manejar as suas próprias atividades de aprendizagem e motivação fornece confiança e realce às realizações de desempenho.

Figura 3 - Comandos até agora presentes no jogo.

Como o jogo baseia-se em no controle de um personagem, e que os blocos de comando os direcionam a um determinado local no jogo, foram também considerados a quantidade de passos dados no cálculo da pontuação geral da fase.

5. Avaliação

Uma avaliação preliminar foi realizada com 10 alunos voluntários do 3º e 5º período de um curso de Análise e Desenvolvimento de Sistemas. Todos os alunos já haviam cursado disciplinas que abordam lógica de programação. Os dados foram coletados por meio de questionários online respondidos após o jogador testar o *Game Logic*, com o objetivo de avaliar o jogo e, observação para avaliar o comportamento dos jogadores enquanto interagiam com o jogo. Três dos autores fizeram o papel de observador analisando como o jogador se relaciona com o objeto testado, percebendo se o jogo consegue transportá-lo para uma experiência lúdica e verificar se os objetivos do jogo e as instruções para manipulação dos controles são claras e objetivas.

Os dados dos questionários nos permitiram obter resultados otimistas, sobretudo em relação a aspectos de jogabilidade. Pois, os participantes conseguiram compreender facilmente a proposta do jogo e mostraram-se motivados a continuar jogando. A Figura 4 apresenta alguns gráficos dos dados coletados nos questionários. Quando questionados a respeito de uma atribuir uma nota ao *Game Logic*, a maior parte dos participantes, avaliou de 8 a 10. Os jogadores conseguiram relacionar os conteúdos apresentados no jogo com os conteúdos vistos em sala de aula. Um dos participantes afirmou que “Isso vicia igual ao *angry birds*” (Participante 06).

Através das observações pudemos identificar que os jogadores não liam as instruções da primeira fase, que consiste em um tutorial, e já tentavam realizar os desafios propostos. Assim, modificaremos o tutorial para que seja mais objetivo e intuitivo.

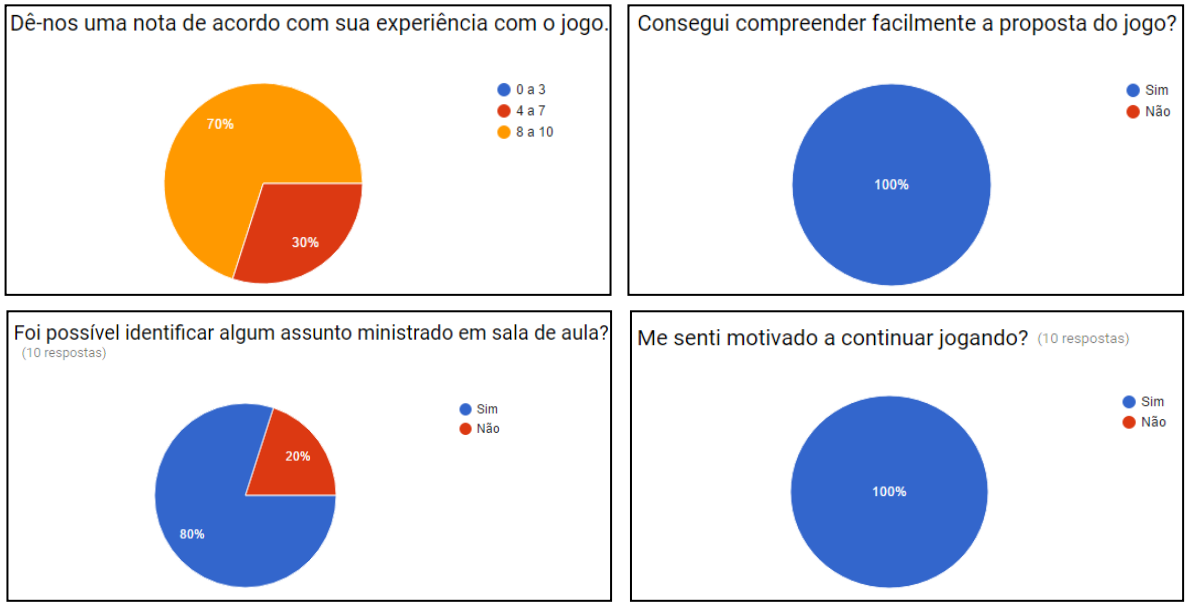

Figura 4 – Gráficos das respostas dos questionários

Outros jogadores tocavam duas vezes na tela fazendo com que o tutorial passasse sem lê-lo, e isso dificultava a jogabilidade (aprendizado do jogo). Uma possível solução é mudar o tempo de toques na tela impondo uma pequena pausa entre um toque e outro. Percebemos, também, que os jogadores tiveram um pouco de dificuldade para remover eventos adicionados para realizar uma sequência de passos à área do lado direito da tela. Assim, uma possível solução será, na primeira fase, exibir uma mensagem informando que o usuário pode remover eventos apenas arrastando-os novamente para a área de seleção de eventos na parte inferior da tela. Alguns usuários não conseguiam entender o botão "foco" por não enxergarem a iluminação contida nele ao realizar o tutorial. Com isso, eles arrastavam o painel direito até chegar o evento "start". Iremos implementar uma setinha apontando para ele com um texto mais explicativo.

Ao final do questionário os jogadores foram perguntados sobre sugestões de melhorias para o *Game Logic*. Os participantes sugeriram que colocássemos um brilho nas estrelas que aparecem ao final de cada fase. As principais palavras do tutorial precisam ter mais destaque. Deixar mais claro o objetivo do jogo, a usabilidade do evento *loop* e o sistema de pontuação. Ao bloquear eventos deixar eles em um tom de cinza para que os usuários saibam que não são clicáveis.

6. Considerações e Trabalhos Futuros

Neste artigo foi apresentado o **Game Logic**, um jogo voltado para o aprendizado de pseudocódigos através da montagem de fluxogramas com blocos de instrução pré-dispostos na interface da ferramenta, tornando o aprendizado de qualquer linguagem de programação mais facilitado, a ferramenta investe principalmente em conceitos fundamentais de aprendizagem através de jogos digitais como, por exemplo, a gamificação, o layout do jogo propõe a inserção do jogador em um mini mundo onde o

personagem é controlado através dos blocos de comando disponíveis e que vão sendo liberados sempre que o novo desafio liberado exigir o uso de um bloco mais otimizado.

Os passos e a quantidade de blocos usados durante a realização do desafio também são determinantes para o cálculo da pontuação total dos desafios propostos pela ferramenta, fazendo com que cada vez mais o estudante busque construir fluxos de blocos de maneira elegante buscando aproveitar ao máximo o potencial cada bloco. Também foi visto que aspectos fundamentais para a construção de jogos educacionais devem ser seguidos, e que a correta aplicação de tais conceitos durante o desenvolvimento de qualquer ferramenta destinada a este público gera resultados positivos para as diferentes partes interessadas na aplicação e no uso da ferramenta.

Tendo em mente que o presente trabalho, bem como, o software aqui apresentado, encontram-se em constante desenvolvimento, a publicação de novas releases auxiliará na captação de *feedback* necessário para melhorias consecutivas, o que também elevará a qualidade da aplicação consideravelmente, uma vez que o próprio usuário final indicará suas necessidades. O desenvolvimento atual foca em aspectos como melhoria de interface e também de usabilidade identificados na avaliação, no que contempla a mitigação de recursos do dispositivo hospedeiro, bem como produção para outras plataformas, uma vez que o foco inicial de produção foi à plataforma Android. Ocorrerá também a implementação de novos blocos de instrução, com o decorrer do desenvolvimento de mais desafios, fazendo com que o estudante venha a conhecer a maior quantidade possível de blocos de instrução disponíveis.

8. Referências

Bandura, A., & Schunk, D. H. (1981). Cultivating competence, self-efficacy, and intrinsic interest through proximal self-motivation. Journal of Personality and Social Psychology, 41, 586-598

Barbosa, L. S.; Fernandes, T. C. B; Campos, A. M. C. (2011) Takkou: Uma Ferramenta Proposta ao Ensino de Algoritmos. WEI – XIX Workshop sobre Educação em Computação, XXXI Congresso da Sociedade Brasileira de Computação.

Borba, Marcelo de Carvalho; Penteado, Miriam Godoy. Informática na Educação. Belo Horizonte: Autêntica, 2001.

Brathwaite, Brenda; Schreiber, Ian, 2009. Challenges for Game Designer – Non-digital exercises for video game designers. Boston: Cengage Learning.

Costa, H.A.X.; Santos, R.P.; Werner, C.M.L. (2010) "Uma Análise do Processo de Ensino e Aprendizagem de Engenharia de Software: Desafios e Soluções no Contexto Brasileiro", In: Proceedings of the XI International Conference on Engineering and Technology Education, Ilhéus, BA, Brasil, 367-371.

Crawford, C. The Art of Computer Game Design. (1982). http://wwwrohan.sdsu.edu/~stewart/cs583/ACGD_ArtComputerGameDesign_ChrisCrawford_1 982.pdf.

Cuperschmid, A. R. M. (2008) Heurísticas de jogabilidade para jogos de Computador. Dissertação (Mestrado em Artes) – Programa de Pós-Graduação em Artes, Universidade Estadual de Campinas, Campinas-SP.

Dantas, V.; Freitas, P.; Alencar, L. (2011) ProGame: Um jogo para apoiar o ensino aprendizagem de programação. In: First Workshop on Applications to Provide Learning

and Teaching Support (APPLETS), Aracaju – SE. Anais do XXII SBIE – XVII WIE, 2011.

Dicheva, D., et al., Gamification in Education: A Systematic Mapping Study, in: Educational Technology & Society, v. 18, n. 3, 2015.

MORATORI, Patrick Barbosa. Por que utilizar jogos educativos no processo de ensino aprendizagem. UFRJ. Rio de Janeiro, 2003.

Falkembach, Gilse A. Morgental. O lúdico e os jogos educacionais. Mídias na Educação. Disponível em: http://penta3.ufrgs.br/midiasedu/modulo13/etapa1/leituras/arquivos/Leitura_1.pdf Acesso em 15 de abril de 2016, v. 16, 2006.

Huizinga, Johan. Homo ludens: O jogo como elemento de cultura. São Paulo: Perspectiva, 1998

Kapp, K. The gamification of learning and instruction: Game-based methods and strategies for training and education. Pfeiffer, 2012.

Moratori, P, B. Por que utilizar jogos educativos no processo de ensino aprendizagem? Trabalho de conclusão da disciplina Introdução a Informática na Educação 28 f. (Mestrado de Informática aplicada à Educação) – Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2003.

Nakamura, J., Csikszentmihalyi, M., The concept of flow. in: Handbook of positive psychology, p. 89105, 2002.

Neves, M.F.; Coello, J.M.A. (2006) "OntoRevPro: Uma Ontologia sobre Revisão de Programas para o Aprendizado Colaborativo de Programação Java", In: Anais do XVII Simpósio Brasileiro de Informática na Educação, Brasília, DF, Brasil, 569-578.

Prensky, M. Digital game-based learning. Computers in Entertainment (CIE), 1(1), 21-21. 2003.

Schell, Jesse, 2008a. The Art of Game Design – A Book of Lenses. Burlington: Morgan Kaufmann Publishers.

Schunk, D. H. (1995). Self-efficacy and education and instruction. In J. E. Maddux (Org.), Self-efficacy, adaptation, and adjustment: Theory, research, and application (pp. 281-301). Nova York: Plenum

Silva, Max André de Azevedo; Dantas, Ayla. KLouro: Um jogo educacional para motivar alunos iniciantes em programmação. In: Anais do XXV Simpósio Brasileiro de Informática na Educação. III Congresso Brasileiro de Informática na Educação, Dourados, MS, Brasil, 702-7011 (2014).

Tang, S., Hanneghan, M. and El Rhalibi, A. Introduction to Games-Based Learning, In Games-based Learning Advancement for Multisensory Human Computer Interfaces: Techniques and Effective Practices (Eds: T.M. Connolly, M.H. Stansfield and E. Boyle). Idea-Group Publishing: Hershey. ISBN: 978-1- 60566-360-9. 2009

Vianna, Ysmar; Vianna, Maurício; Median, Bruno e Tanaka, Samara (2013) Gamification, Inc: como reinventar empresas a partir de jogos. 1. Ed. – Rio de Janeiro : MgJV Press.

Zichermann, G.; Cunningham, C. Gamification by Design: Implementing game mechanics in web and mobile apps. Sebastopol: O'Reilly Media Inc, 2011.