

Ensino de Lógica de Programação através do Jogo Defense of the Ancients 2

Wellington Cataldo Roberti Junior, Ângelo Soldati Lavorato, Daniel Camargos Lucarelli, Alex Fernandes da Veiga Machado

Departamento de Ciência da Computação – Instituto Federal de Educação, Ciência e Tecnologia do Sudeste de Minas Gerais - Campus Rio Pomba
CEP: 36180-000 – Rio Pomba – MG – Brasil

junimduic@hotmail.com, angelolavorato@hotmail.com,
danielcamargoslucarelli@gmail.com, alex.machado@ifsudestemg.edu.br

Abstract. *Students of higher education courses in Computer Science often go through difficulties with certain complex subjects when they enter the course, which leads them to give up or simply leave this area of study. This article aims to develop the interest and fixation of programming concepts to students of the computer course using the game Defense of the Ancients 2 and the programming language Lua. For this purpose, a four (4) hour course was taught with the main programming and in the end a questionnaire was applied to analyze the results. As a main result of this mini-course, was achieved a development in the students's interest of 85,7% to programming, and most of them stated their interest in going beyond what was addressed.*

Resumo. *Estudantes dos cursos superiores de Ciência da Computação costumam passar por dificuldades com determinados assuntos complexos quando ingressam no curso, o que os leva a desistir ou simplesmente abandonar essa área de estudos. Este artigo visa desenvolver o interesse e fixação dos conceitos de programação dos alunos do curso de informática utilizando o jogo Defesa of the Ancients 2 e linguagem de programação Lua. Para isso foi ministrado um curso de 4 horas com as principais funções de programação e aplicado um questionário para análise da proposta. Como resultado deste minicurso verificou-se um desenvolvimento no interesse dos alunos de 85,7% pela programação.*

1. Introdução

O conteúdo de algoritmos e programação é difícil de ser trabalhado e entendido, apresentando vários problemas que, corriqueiramente, levam os alunos a abandonar os cursos superiores da área da computação ou nem a se interessar pelo mesmo. Entre as várias razões, está a falta da de motivação e a dificuldade em desenvolver o raciocínio lógico necessário para a construção dos algoritmos. A evasão de alunos dos cursos de Ciência da Computação é algo bastante notável, cerca de 73%, segundo uma pesquisa realizada pela SEMESP (Sindicato das Instituições Privadas de Ensino Superior do Estado de São Paulo).

A fim de se modificar esse panorama, técnicas e abordagens diferentes são utilizadas para o auxílio e reforço do aprendizado desse conteúdo visto nos cursos. Partindo dessa ideia, este artigo propõe avaliar a viabilidade do uso de uma plataforma

grátis, recém-criada e pouco explorada no mundo acadêmico, destinada a desenvolvedores e entusiastas, para a programação de scripts em Lua.

A linguagem Lua é amplamente utilizada na criação de jogos eletrônicos e no caso deste trabalho foi utilizada para a criação de scripts que servirão para o desenvolvimento de módulos de inteligência artificial de *BOTs* (Personagens controlados pelo computador) de um jogo, bastante popular entre o público jovem, o DOTA 2 (*Defense Of The Ancients 2*) (Valve, 2013) que possui atualmente mais de 12 milhões de jogadores espalhados por todo o mundo segundo a contagem da desenvolvedora do jogo.

Em uma de suas últimas atualizações a desenvolvedora deste jogo (Valve, 2013) disponibilizou para a comunidade a possibilidade da criação de scripts que podem ser inseridos na inteligência artificial dos personagens que serão controlados pelo computador em partidas locais. Com isso seria possível utilizar conceitos de programação em um jogo o que torna a compreensão do conteúdo algo simples.

Com isso veio a ideia de se desenvolver um minicurso que possa servir como auxílio para alunos do curso de Ciência da Computação com dificuldades no desenvolvimento de algoritmos visando incentivá-los a formular novos scripts para o jogo e que também auxilie a comunidade desenvolvedora e entusiasta a criar novos conteúdos que possam ampliar as possibilidades de uma ferramenta ainda não tão explorada e pouco difundida, mas que, ao mesmo tempo, pertence a um jogo tão popular quanto o DOTA 2.

A utilização de jogos eletrônicos para o aprendizado, fixação de conteúdo e treinamento de determinados pontos é algo que tem se tornado bastante difundido e tem se provado útil em praticamente qualquer seguimento, as aplicações são infinitas e os resultados bastante satisfatórios pois a abordagem do assunto é feita de maneira com que o indivíduo não seja "forçado" a se sentar e estudar sobre o assunto, o que pode muitas vezes se tornar algo cansativo, e sim se utilizar de formas lúdicas que irão auxiliar no aprendizado pretendido.

2. Fundamentação Teórica

Promover mudanças no ensino básico de cursos de graduação relacionados com tecnologia não é algo fácil de se lidar, conforme (Rapkiewicz, 2004) *apud* (Santos e Costa, 2006) “O que se pode perceber é esses cursos exercerem grande atração sobre os futuros ingressos em uma faculdade, o que se mostra pela concorrência elevada em processos seletivos. No entanto, vencido o obstáculo de entrada, vêm o início da fase universitária e um dos maiores dilemas: o contato com disciplinas básicas que vão preparar o ingresso para atuar em áreas específicas. Esse contato, sobretudo em cursos de Computação, pode gerar tanto afinidade quanto repulsa, o que acontece com frequência expressiva”.

Para (Santos e Costa, 2006), em Ciência da Computação, para formar um profissional, deve-se incentivar o desenvolvimento de raciocínio crítico, a solução de problemas, a aplicação de métodos de pesquisa e um desenvolvimento profissional contínuo. Assim, a principal razão pelo alto índice de reprovação nas disciplinas de Algoritmos e Programação é a falta de compreensão do raciocínio lógico e, em alguns casos, pela desistência de um curso.

Com altos índices de desistências no curso de Ciência da Computação chegando a quase 73% e 67,3% em Sistemas da Informação (Koike, 2012), um minicurso baseado em um jogo conhecido, como o DOTA 2, e, também, lógica de programação em Lua, pode trazer grandes resultados, diminuindo, assim, a evasão do curso. Tendo em vista que o DOTA 2 é um jogo popular, o aprendizado tende a ser mais divertido e agradável, o que gera uma maior afinidade e fixação nas disciplinas de Algoritmos e Programação.

2.1. Lua

Contudo, ao elaborar um minicurso para tentar diminuir a evasão precisa-se aprender sobre a linguagem de programação Lua, uma linguagem de script de multiparadigma, pequena, reflexiva e leve, projetada para expandir aplicações em geral, por ser uma linguagem extensível (que une partes de um programa feitas em mais de uma linguagem), para prototipagem e para ser embarcada em softwares complexos, como jogos. "Lua é uma linguagem embutida, com sintaxe semelhante à de Pascal, mas com construções modernas, como funções anônimas, inspiradas no paradigma funcional, e poderosos construtores de dados. Isso faz com que Lua seja uma linguagem de grande expressão com uma sintaxe familiar" (Celes *et al*, 2004).

Essa Linguagem foi projetada e implementada no Tecgraf, o Grupo de Computação Gráfica da PUC-Rio e sua primeira versão é de julho de 1993 (Celes *et al*, 2004). Para edição dos scripts em Lua foi utilizada a IDE (*Integrated Development Environment*) do Eclipse que foi feita em Java e segue modelo de código aberto de desenvolvimento de programas.

2.2. Dota 2

Para entender como utilizar os scripts em Lua primeiro temos que entender o que é o DOTA 2. Um jogo do estilo MOBA (*Multiplayer Online Battle Arena*), lançado em 2013 atingindo uma faixa etária de jogadores entre 16 e 25 anos (Arantes *et al*, 2013), DOTA 2 é um jogo de profundidade e complexidade infinitas. Cada herói tem um conjunto de atributos e habilidades que se combinam com os dos seus aliados, garantindo com que cada partida seja única. Essa é uma das razões que fazem com que o fenômeno DOTA não pare de crescer. Originalmente uma modificação feita por fãs para o Warcraft 3, DOTA foi um sucesso instantâneo no submundo dos jogos. Após serem contratados pela Valve, os desenvolvedores originais facilitaram a introdução a outros jogadores, para que o resto do mundo possa desfrutar das mesmas mecânicas.



Figura 1. Cena do jogo com a personagem Crystal Maiden

Afim de se ter uma experiência satisfatória mínima no jogo a desenvolvedora do jogo exige alguns pré-requisitos mínimos do computador do usuário. De acordo com (Valve, 2013), o computador deve possuir um processador dual core com 2.8 GHz, 4GB de memória RAM, nVidia GeForce 8600/9600GT ou ATI/AMD Radeon HD2600/3600 ou equivalente, Directx 9 e 15 GB de espaço livre no HD.

Contendo também um lado competitivo de ação e estratégia o DOTA 2 é jogado profissionalmente e casualmente por mais de 12 milhões de fãs apaixonados ao redor do mundo. No Brasil o cenário competitivo ainda não é muito forte, mas está crescendo rapidamente com times se classificando para eventos maiores e mais recompensáveis. Jogadores escolhem um herói dentre mais de cem disponíveis, formando duas equipes de cinco jogadores. Então os jogadores do time Iluminados enfrentam o time adversário, Temidos, para controlar o mapa com astúcia e raciocínio rápido.



Figura 2. Competição do The International 5

2.3. Valve Developer Community

Para aprender sobre a programação no DOTA 2 existe a *Valve Developer Community*, uma comunidade do estilo Wiki, uma página Web onde múltiplos usuários podem editar e acrescentar informações sobre um determinado assunto, nessa página é possível encontrar várias ferramentas e documentos para criação de jogos do tipo "Triplo-A", uma classificação dada a jogos com grande orçamento e de alta qualidade, ou apenas criar Mods (Modificação) para os jogos da desenvolvedora.

A página da comunidade contém documentos e scripts sobre o *Dota 2 Workshop Tools*, um conjunto de utilitários de programas que permite criar objetos para serem colocados e testados em um modo de jogo customizado. Com essa ferramenta é possível desenvolver materiais, sons, partículas, modelos e scripts. Essa página também possui o Scripting API (*Application Programming Interface*) com todas as funções já criadas e exemplos em Lua para controlar os BOTs. Como o foco maior deste artigo é ensinar lógica e programação, será orientado apenas o uso de scripts que estão disponíveis na

página da comunidade. Será utilizado, também o IDE do eclipse como auxílio para edição dos scripts criados.

3. Trabalhos relacionados

Revisamos trabalhos que defendem uma abordagem de demonstração de jogos eletrônicos com o intuito educacional e de fixação de temas tradicionais muitas vezes tidos como difíceis por alunos de cursos de Ciência da Computação.

No trabalho de (Leutenegger-Jeffrey, 2007) é defendido o uso de uma abordagem "*Game First*" para o ensino de programação introdutória. Segundo os autores, acredita-se que a programação de jogos incentiva a maioria dos novos programadores pois os mesmos estão motivados e interessados pelo conteúdo e os componentes visuais são atraentes para esses jovens no caso deste trabalho utilizamos o DOTA 2 como jogo a fim de se incentivar os alunos.

Em (Haden, 2006) é apresentado aos alunos uma forma diferente de programação para alunos de uma escola politécnica de Otago onde os mesmos foram capazes de programar um jogo utilizando técnicas tradicionais de programação e aplicar as mesmas no desenvolvimento de jogos. No desenvolvimento do nosso trabalho utilizamos a linguagem Lua em scripts para a interação dos BOTS no DOTA 2.



Figura 3. The incredible rainbow spitting chicken

No estudo de (Mathieu, 2009) acredita-se que videogames podem ser utilizados para fomentar o interesse de estudantes em Ciência da Computação. Como jogos estão presentes em inúmeros segmentos de praticamente qualquer assunto, também pode ser interessante a utilização desses jogos para incentivar estudantes desmotivados com o curso. Essa ideia vem como um complemento à nossa forma de abordagem.

Já em (Prensky-Mark, 2007) o livro traz uma abordagem com relação ao processo de utilização de jogos eletrônicos na aprendizagem e que esse processo consiste em duas partes onde a primeira parte é argumentada a prevalência de jogos que servem como uma forma de recompensa para nosso cérebro e que fez com que métodos tradicionais de ensino fossem menos eficazes se comparados a essa abordagem. Já a segunda parte diz que jogos podem ser utilizados, e com sucesso, para o treinamento de funcionários em empresas. A recompensa para o nosso cérebro, neste trabalho, é a utilização do DOTA 2 como meio visual.

4. Materiais e métodos

A metodologia utilizada no desenvolvimento do minicurso baseou-se na proposta em (Delgado *et al*, 2004) e (Júnior *et al*, 2005), que adotou as seguintes fases para o ensino de programação:

- Primeira fase: Iniciação Lúdica, não se baseia em algoritmos ou programação e sim de resolução de problemas de diversos domínios como quebra-cabeças simbólicos, quebra-cabeças lógicos, jogos e charadas, problemas simples de aritmética e geometria. (Delgado *et al*, 2004).
- Segunda fase: Baseia-se na representação de soluções em linguagem natural, noções e conectivos lógicos até a construção de algoritmos (Júnior *et al*, 2005).
- Terceira fase: É a fase da construção dos programas, tradução dos algoritmos para uma linguagem de programação (Lua) e estruturas de repetição (Júnior *et al*, 2005).

Como os alunos fazem parte do curso de informática, já havia um ensino prévio sobre programação e como o intuito deste trabalho é fixar o conteúdo de programação, visou-se pular a fase lúdica e partir para o ensinamento da linguagem e programação da mesma. Logo o ensino começou na segunda fase, explicando direto o método utilizado para a solução dos problemas. Na segunda fase também foi explicado para os alunos o funcionamento de variáveis e vetores além das funções "Se" e "Senão" e, da mesma forma, "Para" e "Enquanto". Já na terceira fase foi aplicado no jogo todo o conhecimento aprendido.

4.1. Abordagem e proposta

Para auxiliar o acompanhamento do conteúdo pelos alunos, foi elaborada uma apostila com os principais assuntos da grade do curso de Ciência da Computação. Esses assuntos de acordo com o Projeto Pedagógico do curso de Ciência da Computação (2014) são:

- Desenvolvimento de algoritmos.
- Tipos de dados primitivos e estruturados.
- Estrutura de controle.
- Apresentação a uma linguagem de alto nível.

No minicurso foi explicado sobre a linguagem Lua, e projetos em que ela foi utilizada. Esses projetos estão entre alguns muitos conhecidos como as ferramentas CryEngine 3 e Corona SDK e jogos bem famosos como Angry Birds, Street Fighter IV, Tibia e DOTA 2. O minicurso seguiu a ordem de ensinar variáveis e operadores, e suas respectivas sintaxes da linguagem Lua, logo depois foi explicado como funciona os "Se" e "Senão"(Figura 4) pois eles dependem do conhecimento de variáveis e operadores para comparação. Assim que todos entenderam, conseguiram imprimir na tela e comparar valores foi introduzido o "Para" e "Enquanto"(Figura 5) juntamente com sua sintaxe.

```
local function main()
  numero=2;
  if ( numero == 1 )
  then
    print("entrou no if");
  else
    print("entrou no else");
  end
end
main()
```

Figura 4. "Se" e "Senão".

```
local function main()
  for i=0,8 do
    print(i);
  end
end

local function main()
  variavel=true;
  i=0;
  while variavel do
    print(i);
    i = i + 1;
    if ( i == 5 )
    then
      variavel=false;
    end
  end
end
```

Figura 5. "Para" e "Enquanto".

Depois de ensinado como funciona a Linguagem Lua e suas funções relacionou-se os códigos com o jogo DOTA 2. Um exemplo de "Se" e "Senão" no jogo pode ser visualizado na figura 6. Este exemplo faz com que todos os BOTs escolham o personagem Lina. Foi mostrado outros exemplos de "Para" e "Enquanto" juntamente com vetores para comprar itens na loja do jogo e caminhar até determinado lugar no mapa. Estes exemplos são encontrados na documentação da *Valve Developer Community*.

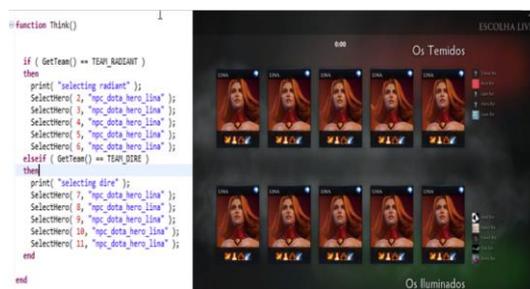


Figura 6. "Se" sendo utilizado no DOTA 2 e seu respectivo resultado.

Por fim, os estudantes responderam a um questionário que continha 8 questões, onde visou saber se o jogo demonstrou ser um ótimo exemplo visual da programação realizada e se é bom misturar jogos com ensino, além de que, se eles gostariam que os professores dessem exemplos similares ao ensinar programação. Este questionário pretendeu avaliar a experiência dos alunos na fixação da disciplina de programação com o auxílio do jogo DOTA 2 como exemplo visual.

5. Experimentos e resultados

Com o formulário entregue para os alunos preencherem, onde foi verificada a satisfação dos mesmos com relação ao minicurso, e também se todos se sentiram interessados em procurar mais sobre scripts do DOTA 2 e a linguagem de programação Lua vem o momento de analisar o experimento. O minicurso teve 7 alunos do curso de Informática com idade de 16 a 17 anos e todos com uma pequena noção de lógica e programação. Ao todo 85,7% dos alunos não conheciam Lua, e 57,1% conhecia o DOTA 2. Dos 7 alunos, 6 obtiveram uma fixação promissora na matéria de Algoritmo e Programação e gostariam que os professores dessem exemplos similares ao ensinar programação. A figura 1 demonstra o ensino da programação em Lua.



Figura 7. Foto do Minicurso sobre programação.

Depois de instruído o funcionamento da programação em Lua foi ensinada a utilização dos scripts direto no jogo. Este foi o momento em que os alunos ficaram mais interessados, pois viram o código que eles escreveram em ação, com personagens movimentando e comprando itens na loja sozinhos. Com grande interesse, todos se ajudavam quando tinham dificuldades, pois alguns tiveram mais facilidade que outros ao conhecer o DOTA 2 antes do minicurso.

Como se pode ver nas figuras 4 e 5 o minicurso obteve ótimos resultados com a maioria dos alunos, que se sentiram motivados a dar continuidade ao curso e interessados a pesquisarem mais informações sobre scripts em Lua, obtendo uma porcentagem de 85,7% muito interessados e 14,3% bem interessados, além de 57,1% ficarem interessados a procurar mais sobre Lua e atingir no mínimo neutro em relação a procura sobre a programação de scripts.

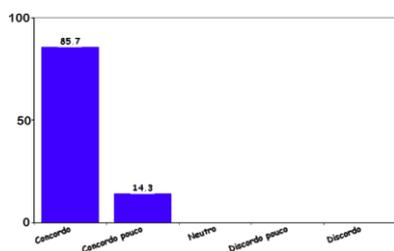


Figura 8. Motivação dos alunos durante o curso.

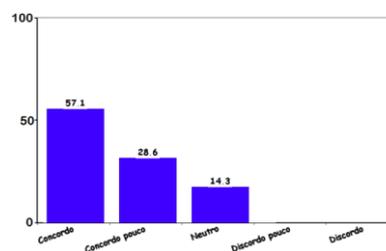


Figura 9. Interesse sobre scripts.

Sendo o DOTA 2 um jogo muito popular, todos os alunos concordaram que foi um ótimo exemplo visual de programação, como se pode ver na figura 5. Como aproximadamente 57% dos alunos já conheciam o DOTA 2, estes procuraram saber mais sobre a programação de scripts e o restante que não conhecia o jogo, criou um interesse maior vendo as possibilidades que o mesmo trouxe. No fim os alunos pediram que fossem realizados mais minicursos como este e que os mesmos tivessem uma duração maior. Um aluno disse também, que adorou a explicação misturada com jogos e ficou satisfeito com o que aprendeu.

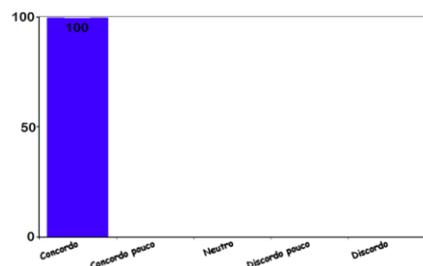


Figura 10. Aprovação do dota como uma ferramenta visual de programação.

6. Conclusões

Como foi observado ao longo do trabalho a apresentação de tópicos referentes a certas disciplinas por meio de jogos eletrônicos é algo de elevada aceitação pelos indivíduos participantes do minicurso. Podemos notar pela seção de Experimentos e Resultados que os alunos se sentiram motivados durante todo o curso e também interessados a ir além do que foi apresentado, o que é bastante significativo.

Tendo em mente que o público alvo do estudo são estudantes e também professores dos cursos de Ciência da Computação, é possível concluir com o que foi desenvolvido, que o minicurso ajudou os mesmos se sentirem motivados a explorar o conteúdo visto em sala de aula em jogos a fim de se observar na prática o funcionamento de estruturas de dados num ambiente familiar como o de um jogo de computador.

Este artigo vem para incentivar e também motivar educadores a utilizar técnicas lúdicas no momento de explicar ou passar um conhecimento, pois todo conteúdo quando é dado de maneira leve e sem a "pressão" que é vista nos métodos tradicionais de ensino torna-se algo prazeroso não somente para quem está aprendendo, mas também para quem ensina.

Todos nós fomos um dia crianças, e descobrimos as coisas simples da vida "brincando" e essa pureza também pode ser transcendida a indivíduos que não são mais crianças, mas que em seu interior ainda carregam a inocência que nos torna eternos aprendizes que sempre estarão aqui para aprender e passar o seu conhecimento a frente, sempre de forma prazerosa, a todas as gerações futuras.

Concluindo o estudo, podemos deixar para trabalhos futuros o aprofundamento e tratar de assuntos mais complexos e, também, utilizar ainda mais scripts no DOTA 2 empregando a Linguagem Lua com o intuito de aprimorar e agregar atividades. Como próximo minicurso suporta-se a ideia de instruir os alunos com conceitos de Inteligência artificial (IA) com a criação de uma IA para os *BOTs*, na qual, eles consigam raciocinar e resolver problemas, tais como, se é melhor atacar ou se defender de um inimigo.

7. Agradecimentos

Os autores gostariam de agradecer ao IFSudeste-MG e ao PET (Programa de Educação Tutorial) pelo apoio técnico e financeiro para o desenvolvimento do projeto.

8. Referências

- Arantes, Taís Turaça, and Nataniel dos Santos Gomes. "UMA ANÁLISE DA LINGUAGEM DOS JOGADORES DE DOTA.", 2013.
- Celes, Waldemar, Luiz Henrique de Figueiredo, and Roberto Ierusalimschy. "A Linguagem Lua e suas Aplicações em Jogos." *Rio de Janeiro* (2004).
- Delgado, Carla, et al. "Uma abordagem pedagógica para a iniciação ao estudo de algoritmos." *XII Workshop de Educação em Computação*. 2004.
- Departamento de Ciência da Computação. "Projeto Pedagógico do curso de Ciência da Computação", 2014. Disponível em: <https://sistemas.riopomba.ifsudestemg.edu.br/cgg/Siscgg/Cgg/Up_Downloads/PPC_Ciencia_da_Computacao_20142015__formatado_ID_0000000156_1.pdf/> Acesso em: 20 março 2017
- Haden, Patricia. "The incredible rainbow spitting chicken: teaching traditional programming skills through games programming." *Proceedings of the 8th Australasian Conference on Computing Education-Volume 52*. Australian Computer Society, Inc., 2006.
- Júnior, J. C. R. P., et al. "Ensino de algoritmos e programação: uma experiência no nível médio." *XIII Workshop de Educação em Computação (WEI'2005)*. São Leopoldo, RS, Brasil. 2005.
- Koike, Beth. Mercado Aquecido faz alunos abandonarem cursos de tecnologia, 2012. Disponível em: <<http://www.valor.com.br/carreira/2834642/mercado-aquecido-faz-alunos-abandonarem-cursos-de-tecnologia/>>. Acessado em 05/06/2016
- Leutenegger, Scott, and Jeffrey Edgington. "A games first approach to teaching introductory programming." *ACM SIGCSE Bulletin* 39.1 (2007): 115-118.
- Muratet, Mathieu, et al. "Towards a serious game to help students learn computer programming." *International Journal of Computer Games Technology* 2009 (2009): 3.
- Prensky, Marc, and Mark Prensky. *Digital game-based learning*. Vol. 1. St. Paul, MN: Paragon house, 2007.
- Santos, R. P. dos; Costa, H. A. X. (2006). Análise de Metodologias e Ambientes de Ensino para Algoritmos, Estruturas de Dados e Programação aos Iniciantes em Computação e Informática. In: INFOCOMP, Volume 5, nº.1, ISSN 1807-4545.
- Valve. "Defense Of the Ancients 2", 2013. Disponível em: <http://store.steampowered.com/app/570/Dota_2/> Acesso em: 24 maio 2017.
- Valve. "Valve Developer Community", 2013. Disponível em: <https://developer.valvesoftware.com/wiki/Dota_2_Workshop_Tools/Scripting/> Acesso em: 15 março 2017.