

Aprendizagem Baseada em Problemas Aplicada em Cursos de Linguagem de Programação com Robocode

Matheus Carvalho Meira¹

Carolina Ayumi Matumoto²

Andre Luis de Castro Peixoto³

RESUMO

Alunos da área de informática podem apresentar dúvidas e dificuldades em disciplinas de programação. Essas questões são preocupantes, pois indicam que os conteúdos não estão sendo assimilados pelos discentes, o que pode desestimular o aprendizado. O método de ensino pode ser diversificado de modo a despertar o interesse de estudar nos alunos a partir de jogos digitais, como o Robocode. Métodos diversificados expandem as possibilidades do aluno criar estratégias e desenvolver programas em um ambiente lúdico. Este trabalho apresenta um curso com o jogo educacional Robocode, que busca motivar o aluno a aprender linguagem programação com base em problemas.

Palavras-chave: Ensino e Aprendizagem Programação. Lúdico. Robocode.

1. Introdução

As disciplinas de programação devem trabalhar raciocínio lógico, abstração e apresentação de sintaxe específica a uma linguagem de programação (MEIRA; LIMA; BORGES, 2016; ZANETTI; OLIVEIRA, 2015). É comum que alunos ingressantes em instituições de ensino que oferecem cursos técnicos ou superiores na área de informática

¹ Mestre em Tecnologia e Inovação – UNICAMP -Professor Efetivo no Instituto Federal São Paulo (IFSP), Campus Capivari – SP

² Graduando em Tecnologia em Análise e Desenvolvimento do Sistemas (2016-Atual) pelo Instituto Federal São Paulo (IFSP) - Campus Capivari - SP

³ Doutor em Engenharia Química – USP -Professor Efetivo no Instituto Federal São Paulo (IFSP), Campus Capivari – SP

apresentem dificuldades na aprendizagem de conteúdos práticos e conceituais, em disciplinas de programação de computadores. Focar o ensino de programação em conteúdos sintáticos da linguagem antes da apresentação de suas finalidades pode gerar consequências como notas baixas, desmotivação ou até mesmo desistência do curso.

A evolução gradual do aprendizado em temas apresentados nas disciplinas de linguagem de programação ocorre quando existe o domínio da teoria, além da sua aplicação e exercitação nos princípios práticos para resolução de problemas. No momento em que o aprendizado não ocorre, ou ocorre apenas superficialmente, existe a necessidade de se estudar propostas para diversificação de métodos de ensino-aprendizado com objetivo de incentivar e despertar o interesse dos alunos junto às disciplinas de programação de computadores (MEIRA; LIMA; BORGES, 2016).

O presente trabalho traz uma proposta de curso de linguagem de programação com interação em jogo educacional associado a metodologia de Aprendizado Baseado em Problemas (*Problem Based Learning - PBL*) para “desafiar alunos a aprender e trabalhar em grupos na busca de soluções para os problemas reais” (HOU, 2014). O curso utiliza o jogo educacional digital Robocode para diversificação do ensino-aprendizagem na linguagem de programação Java de modo lúdico. O software Robocode, como *framework* que permite desenvolvimento de robôs virtuais para disputas em uma arena, pode se enquadrar como uma opção interessante para atrair atenção dos alunos na aprendizagem de conceitos e resolução de problemas pertinentes à disciplina de linguagem de programação Java. O trabalho busca a utilização do método de PBL em um curso de linguagem de programação aplicada no Instituto Federal de São Paulo (IFSP), campus Capivari-SP. PBL representa um método de ensino-aprendizagem centrado no aluno, com estratégias voltadas à resolução de problemas.

2. Embasamento Teórico

Estudos discutem o ensino e aprendizagem de linguagem de programação de computadores a partir de práticas pedagógicas com auxílio de software educacional

(ZANETTI; OLIVEIRA, 2015). Ambientes como o Scratch⁴, software de programação que funciona como um facilitador utilizado em atividades com a necessidade de exploração individual ou colaborativa (AURELIANO; TEDESCO, 2012), podem atrair a atenção dos alunos no aprendizado de conceitos de linguagem de programação. Fatores como a organização da sala em grupos podem influir e incentivar relacionamentos entre professores e alunos a desenvolver ambientes favoráveis à aprendizagem de linguagem de programação (GIRAFFA; MULLER; MORAES, 2015). Assim como o Scratch, o jogo educacional digital Robocode permite que ambientes de ensino e aprendizagem de linguagem de programação possam ser desenvolvidos.

Robocode pode ser enquadrado como um *framework* de simulação (RECCHIA; CHUNG; POCHIRAJU, 2014). A classificação do Robocode está centrada em um simulador de batalhas com base em Inteligência Artificial (IA). No simulador de batalhas, é possível observar o comportamento dos robôs. É possível, também, comparar a programação de robôs efetuada por outros desenvolvedores. No simulador, os robôs são entidades autônomas e os programadores não têm controle das ações durante as batalhas. Um robô não possui total conhecimento do meio ambiente da disputa e suas informações estão restritas aos sensores que ele tem (ex. ângulo robô, etc.) (FAYEK; FARAG, 2015).

O Robocode se enquadra como jogo permite o ensino de linguagem de programação Java. A linguagem de programação trabalhada foi Java, introduzida pela Sun Microsystems em meados dos anos 1990. “O Java é uma linguagem de programação de computador, divertida para iniciantes e adequada para programadores experientes utilizarem na construção de sistemas de informações importantes.” (DEITEL; DEITEL, 2010). O objetivo da presente abordagem está centrado na apresentação do jogo educacional Robocode em cursos para estímulo de aprendizagem de linguagem de programação. O Robocode pode ser explorado como plataforma no desenvolvimento de estratégias de jogo a partir de cursos planejadas em um ambiente de

⁴ Scratch: <https://scratch.mit.edu>. O Scratch ajuda os jovens a pensar de forma criativa, a raciocinar sistematicamente e a trabalhar colaborativamente.

ensino e aprendizagem de programação de computadores (MEIRA; LIMA; BORGES, 2016).

“Bons jogos digitais incorporam bons princípios de aprendizagem” (GEE, 2005). “O aprendizado baseado em jogos educativos está se tornando uma tendência na área de pesquisa em Informática na Educação, pois trata de vários problemas típicos do Ensino, como altas taxas de abandono devido à frustração, falta de motivação para continuar a estudar e a sobrecarga cognitiva do aluno” (YESSAD; LABAT; KERMORVANT, 2010).

Para as práticas pedagógicas com jogos educacionais foi considerado a adoção da metodologia PBL com objetivo de auxiliar o desenvolvimento de ambientes propícios ao ensino e aprendizagem. O presente artigo associa o PBL para desenvolvimento do ambiente do curso. Com o PBL, o estudante deixa de ser um receptor passivo do conhecimento e assume papel do agente principal responsável pelo aprendizado. A situação do professor também muda, não seguindo as estratégias do ensino instrucional, assumindo a função de facilitador na construção do conhecimento (GIL, 2008).

3. Metodologia

Para objetivo do trabalho, “curso de linguagem de programação com PBL associado ao Robocode”, de modo a privilegiar e tornar mais simples a leitura deste texto, será adotado o termo “Curso Robocode” para representar a incidência do curso. Para idealização do curso, foram conduzidas entrevistas com professores que ministram as disciplinas de linguagem de programação. O resultado da entrevista teve como objetivo principal o levantamento dos desafios (problemas a serem resolvidos pelos alunos durante o jogo) de acordo ao método PBL aplicado no Robocode. Para a técnica de entrevista, o trabalho trata a observação direta intensiva que se “caracteriza como um encontro entre duas pessoas, a fim de que uma delas obtenha informações a respeito de um determinado assunto” (MARCONI; LAKATOS, 2010). Questionários, caracterizando a observação direta extensiva, foram aplicados aos alunos para analisar resultados com base na resolução e respostas aos desafios do curso proposto.

3.1 Curso Robocode

Este trabalho busca analisar a utilização do Robocode como ferramenta para estruturação e planejamento de uma prática pedagógica que atenda os objetivos da disciplina de linguagem de programação em nível médio. Foram feitas entrevistas padronizadas com professores que ministram a disciplina de linguagem de programação com o objetivo de auxiliar na elaboração dos conteúdos do Curso Robocode. O Ambiente Virtual de Aprendizagem (AVA) Moodle foi usado para hospedar e compartilhar as definições da disciplina. Foi usado o padrão IMS-LD⁵ no AVA Moodle para elaborar e ofertar o Curso Robocode. O curso foi aplicado em 2016 no IFSP Capivari nos últimos períodos do curso Técnico Integrado em Informática do campus de modo concomitante com as disciplinas de linguagem de programação. A proposta de Curso teve o objetivo estimular a diversificação dos métodos de ensino e aprendizagem de modo lúdico.

O curso consistiu aulas diversificadas, com envolvimento de equipes e jogos no Robocode, distribuídos em 6 aulas (50 minutos cada) com turma de 20 alunos. As equipes foram divididas em grupos de no mínimo 4 e máximo de 6 alunos. O curso focou como objetivo, analisar assuntos introdutórios de linguagem de programação Java, delimitado nas abordagens dos seguintes conceitos introdutórios: (1) Sintaxe; (2) variáveis; (3) Estruturas condicionais; (4) Estruturas de repetição. Os conceitos foram abordados na mesma sequência e conforme a ementa da disciplina de linguagem de programação. De acordo com a metodologia PBL, desafios foram criados pelos professores para representar um problema denominado “Base”. O problema “Base”, tratado no curso, trabalhou com desafios em aprimorar a movimentação do robô no Robocode. Os problemas relacionados a movimentação do robô são apresentados dentro do curso para que os alunos tenham condições de pesquisar acerca dos conceitos da linguagem de programação Java.

⁵ Consórcio IMS (Instructional Management Systems) define os padrões educacionais como IMS Learning Design (IMS-LD). O IMS-LD é responsável pelo modelo para planejamento de cenários pedagógicos que utilizam estruturas da linguagem de marcação para criação de documentos com dados hierárquicos XML (eXtensible Markup Language) (Balbino, 2009).

Durante as aulas, os professores tiveram o papel de tutores ou facilitadores com objetivo de oferecer subsídios e orientações aos alunos diante dos problemas “Base” durante o curso. Professores puderam disponibilizar fontes de recursos (livros, artigos, apostilas, vídeos, etc.) para as pesquisas de acordo com cada conceito apresentado ao longo das aulas. Além de compartilhar o conteúdo do curso, o AVA Moodle foi utilizado para: (1) Organizar as aulas; (2) Dividir os assuntos em tópicos (sintaxe, variáveis, estruturas condicionais e de repetição); (3) Disponibilizar os desafios (problemas “Base”); (4) Local para que professores pudessem concentrar os conteúdos necessários para resolver os desafios a cada aula; (5) Criar questionários do Curso Robocode. Além das características teóricas e práticas pertinentes ao ensino-aprendizagem de linguagem de programação Java, foram inseridas também, ao longo do curso, problemas associados a definição de estratégias de jogo Robocode (ex. qual melhor ou mais apropriada movimentação para o robô de acordo com o jogo?).

Com apresentação das propostas e definição dos problemas durante o curso, paralelamente, os alunos devem: (1) ter acesso ao ambiente de desenvolvimento do software para desenvolver os robôs; (2) criar, a partir de conhecimentos prévios e pesquisas estruturadas, os códigos necessários; (3) compilar e corrigir possíveis erros; (4) testar o código no Robocode para analisar o resultado e planejar (“treinar”) o robô conforme as características do ambiente e dos seus adversários.

3.2 Aplicação do curso

No curso Robocode foram apresentados eventos e métodos específicos à POO aplicáveis ao ambiente educacional. Foi possível relacionar os temas abordados em ambientes tradicionais com novos temas desenvolvidos no cenário de interatividade do Robocode. A abordagem de ensino-aprendizado utilizada trabalhou os conceitos introdutórios, porém pode ser empregado para trabalhar conceitos avançados seguindo a mesma metodologia no desenvolvimento dos desafios. A cada disputa finalizada no jogo, os alunos tiveram a oportunidade de perceber que algo a mais foi aprendido, e que a lógica assimilada não será utilizada somente em sala de aula, mas em futuros cenários profissionais. A partir de um ambiente de ensino que remete à temas realistas aliados

aos jogos, a proposta do curso buscou desenvolver nos alunos um maior grau de interesse e uma maior qualificação.

Os problemas apresentados durante as partidas disputadas no Curso Robocode são dinâmicos. Problemas no Robocode podem admitir diferentes naturezas, de acordo com o ambiente do jogo, como: problemas de movimentação; acurácia na precisão dos disparos; análise e gestão de energia do robô; etc. A cada rodada, os alunos precisaram analisar e propor definições ou melhorias de estratégias, conforme detecção das situações problemas de jogo. De acordo com o estudo das partidas, a partir do levantamento de hipóteses e apontamento de conclusões para melhoria da estratégia, os alunos precisaram estudar os conceitos e desenvolver na prática novas programações adotadas para robô.

Neste momento, o professor assumiu o papel de facilitador, com objetivo de promover e intermediar debates ao longo do Curso Robocode, buscando mediar o levantamento das problemáticas e a construção das possíveis hipóteses admitidas em torno das situações encontradas no jogo. Como as situações tradicionais no Robocode admitem diferentes planos para sua resolução, o professor encontra um cenário favorável para apresentar conceitos da linguagem de programação: estruturas de repetição; estruturas condicionais; etc. A partir da análise dos conceitos apresentados, os alunos puderam adotar e programar aqueles que julgaram mais adequados, conforme as distintas situações-problema encontradas no jogo Robocode no decorrer do curso.

Seguindo o PBL, os problemas são gerados nas disputas no Robocode. Esses problemas foram elaborados nas entrevistas com professores, respeitando a ementa da disciplina, sendo apresentados às equipes participantes do curso. O problema apresentado na Figura 1, chamado de problema incompleto no PBL, exigiu dos alunos uma pesquisa para aprofundar os conhecimentos incompletos das regras e características do Robocode. Os problemas devem servir de referência na caracterização de novos conceitos.

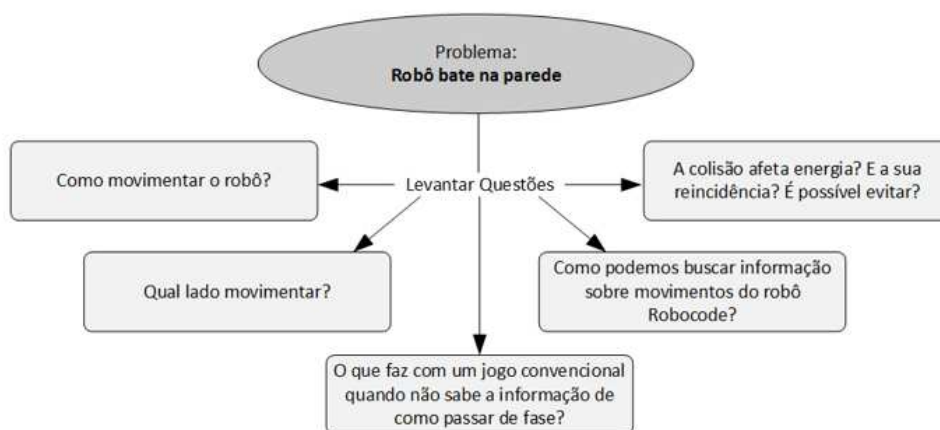


Figura 1 - Problema de Movimentação do Robô

A elaboração do primeiro problema tem sua base na introdução a programação de robôs com Robocode. O problema pode ser descrito na sequência das ocorrências: (1) O código *default*, Figura 2 (conhecido como código base em Java), é apresentado sempre que um novo robô é criado; (2) No código base, é possível observar na sequência da estrutura de movimentação (*run*) do robô o evento *onHitWall()*; (3) O evento *onHitWall()* é executado quando o robô bate na parede; (4) voltar, *back(100)* está configurado com parâmetro de valor 100 (cem); (5) O primeiro desafio é apresentado: qual procedimento adotar quando o robô bate na parede?

```

1 package liag;
2 import robocode.*;
3 public class Liag extends JuniorRobot
4     public void run() {
5         while(true) {
6             ahead(100); turnGunRight(360); back(100); turnGunRight(360);}
7     public void onHitWall() {
8         back(100);

```

Figura 2 - O código default Robocode

O problema incompleto segue uma das características marcantes do PBL, que é apresentar conceitos necessários para sua resolução (FILHO; RIBEIRO, 2009). Devem ocorrer, a partir desse ponto, os passos gerais da PBL com: discussão do problema; investigação do que sabem; geração de hipóteses; obtenção dos objetivos de aprendizagem e atribuições de tarefas formatadas na Tabela 1 (HOU, 2014). Perguntas relacionadas ao conhecimento prévio do aluno podem ser instigadas: “O que você faz

Revista Tecnologias na Educação – Ano 10 – Número/Vol.25 –Julho 2018
tecnologiasnaeducacao.pro - tecedu.pro.br

em um jogo convencional quando não sabe a informação de como passar de fase?” O desafio no âmbito do levantamento e discussão de informações para resolução do problema segue na apresentação da Tabela 1.

Na resolução do problema, além dos benefícios diretos em pesquisar programação de métodos (métodos *ahead*, *back*, *turnRight*, *turnLeft*, ex. *getBearing()* ângulo em graus da parede batida em relação ao robô), indiretamente tiveram acesso as noções de programação de eventos (evento *onHitWall*) que dispara o método “quando robô bater na parede”. A Figura 3 compilou algumas respostas de equipes para o problema. As resoluções dos problemas “Base” permitiram às equipes desenvolverem estratégias de movimentação no instante em que o robô bater na parede. Essas resoluções exibiram a necessidade de registrar as regras para desenvolvimento do código, conforme as condições apresentadas durante a disputa. Os assuntos envolvidos focaram estruturas introdutórias da ementa da disciplina de linguagem de programação. O problema base seguiu os conceitos apresentados por Berbel (1998) para PBL, com envolvimento em fases de resolução. Na primeira fase, os alunos formularam os objetivos de aprendizagem a partir da discussão do problema. Na segunda fase, após a realização do estudo, os alunos rediscutiram o problema com base nos novos conhecimentos adquiridos.

Tabela 1 - Dinâmicas para Levantamento de informações (Hou, 2014).

Desafio	Característica na Resolução do Problema
1 – Pesquisar em Revistas Especializadas	Aluno conclui, assim como em jogos convencionais, manuais técnicos e artigos são fontes para encontrar informações do software de pesquisa.
2 – Classificar Informação	Processo de triagem e classificação de informações por grau de pertinência em resposta ao problema apresentado.
3 – Comentar o Artigo	Auxiliou na movimentação do robô? Único modo de resolver o problema? Outras informações relevantes além do problema? Comentar o artigo com intuito de montar uma base de dados de informações do Robocode.
4 – Reunião em Grupo	Reunir em grupo para apresentar a pesquisa e informações pertinentes levantadas por cada integrante a fim de elencar o que será aplicado na resolução do problema. Necessário um <i>brainstorm</i> para registro de ideias.
5 – Sequência para Resolução do problema	Passos necessários para solucionar o problema. Discutir em grupos as soluções apresentadas. Resolução e justificar o motivo da sequência adotada.

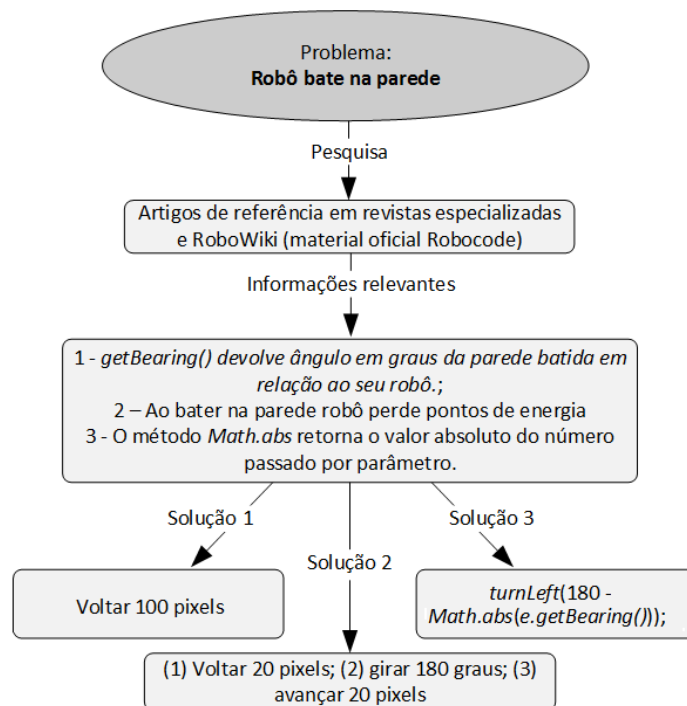


Figura 3 - Resposta Macro ao Problema.

4. Análise e Discussão dos Dados

Ao final do Curso Robocode, realizado no ano de 2016, questionários foram respondidos individualmente pelos 20 alunos participantes. O grupo de alunos foi composto por estudantes do curso Técnico Integrado em Informática do IFSP Capivari com idades entre quinze a dezoito anos, sendo aproximadamente dois terços do gênero masculino.

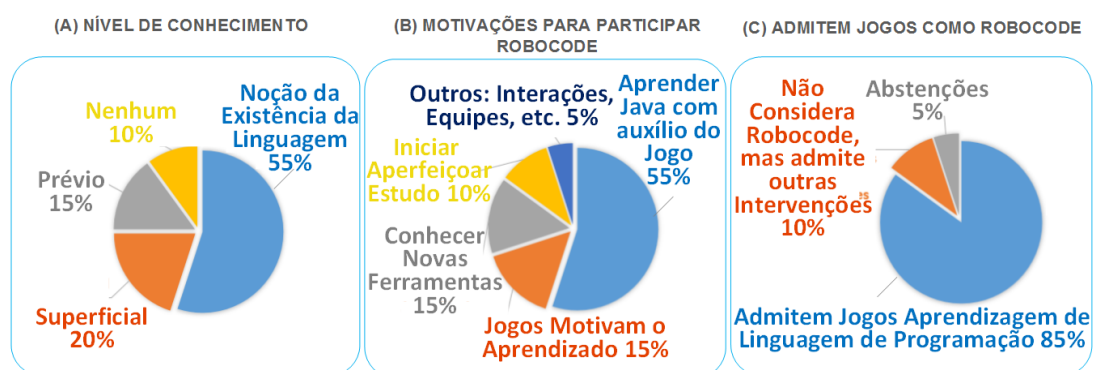


Figura 4 - Questões: (a) nível de conhecimento; (b) motivações no Curso Robocode; (c) admitem jogos educacionais.

As seguintes questões foram relacionadas: nível de conhecimento da linguagem de programação Java (Figura 4 (a)); motivações para participar do Curso Robocode, (Figura 4 (b)); admitem jogos educacionais como Robocode no auxílio e diversificação do aprendizado (Figura 4 (c)). Na questão de nível de conhecimento da Figura 4 (a), mais da metade dos alunos indicaram já ter uma noção da existência da linguagem da programação Java. Na questão de motivações da Figura 4 (b) cerca de 55% das respostas indicaram que a motivação para participar do curso relacionou-se com aprender programação com auxílio de jogos; 15% indicaram que a motivação veio de participar de disputas em jogos e outros 15% indicaram que se motivaram por conhecer novas ferramentas; 10% teve como motivação iniciar ou aperfeiçoar a linguagem de programação Java. Na questão da Figura 4 (c), 95% os alunos indicaram que admitem a diversificação de metodologias com base na aprendizagem com jogos digitais educacionais. Na mesma da questão da Figura 4 (c), 85% dos alunos apontam ainda que o curso Robocode auxiliou em conceitos ou práticas estudados previamente ou posteriormente ao ensino tradicional.

5. Conclusões

O desafio pode ser usado como grande fonte de inspiração para aprender linguagem de programação. É possível perceber que, quando engajados em jogos que prometem o desafio, alunos tendem a estar motivados para resolver questões que não são interessantes no ensino tradicional. A diversão passou a ser o que Papert (1996) chama de “*hard fun*”, na medida em que é ao mesmo tempo desafiador e interessante.

Este trabalho apresenta um experimento de uso do Robocode para desenvolver conceitos presentes nas disciplinas de linguagem de programação. No curso, o Robocode foi utilizado pelos professores para criar situações problemas de assuntos introdutórios da disciplina de linguagem de programação. Os alunos usaram o Robocode para tratar assuntos da linguagem e desenvolver respostas aos problemas e aos desafios do jogo. Trabalhar com interação na programação durante o curso, permitiu às equipes desenvolver habilidades colaborativas e estimulantes à criatividade para resolução das problemáticas propostas. As hipóteses levantadas para tratar

colaborativamente os problemas, proporcionaram a discussão em grupo na seleção das características das melhores soluções. O compromisso com os desafios do Curso Robocode em acompanhar os resultados, possibilitou criar um senso crítico na tentativa de constante melhoria dos programas e da estratégia de jogo.

Um importante aspecto observado foi que o Curso Robocode oportunizou a exploração de métodos e eventos com situações do jogo, para auxiliar o aprendizado de linguagem de programação Java. Os grupos de alunos precisavam, por exemplo, definir estratégias de reação nos instantes em que o robô batesse na parede, e conseqüentemente aperfeiçoar essa movimentação. Além disso, o Curso Robocode expandiu as possibilidades de uso do raciocínio lógico, pois para cada alteração na programação dos robôs foi necessário elaborar previamente a estratégia associada.

As situações dinâmicas de jogo associadas aos professores no desenvolvimento de um bom problema no Robocode, encorajaram os alunos a buscar seu conhecimento prévio. O curso auxilia na aquisição de novos conceitos a partir do levantamento de informações para desenvolver respostas ao problema. A proposta do Curso Robocode estimulou a competição, despertando o interesse em desenvolver e aperfeiçoar os códigos com objetivo em vencer as disputas ou melhorar a estratégia de jogo. Mais da metade dos entrevistados associaram os desafios, vivenciados no jogo, como principais motivações para o aprendizado da linguagem de programação Java durante o Curso Robocode. Cerca de 85% dos alunos admitiram situações problemas com envolvimento de jogos, como o Robocode, para aprendizagem de linguagem de programação. Essa porcentagem abre novas possibilidades de pesquisa envolvendo PBL associado ao Robocode, buscando promover e diversificar ambientes propícios para auxílio no aprendizado das competências presentes nas disciplinas de linguagem de programação.

6. Referências Bibliográficas

AURELIANO, V. C. O.; TEDESCO, P. C. D. A. R. Avaliando o uso do Scratch como abordagem Alternativa para o processo de ensino-aprendizagem de programação. **XX Workshop sobre Educação em Computação. XXXII CSBC**, p. 10, 2012.

Revista Tecnologias na Educação – Ano 10 – Número/Vol.25 –Julho 2018
tecnologiasnaeducacao.pro - tecedu.pro.br

BERBEL, N. A. N. A problematização ea aprendizagem baseada em problemas. **Interface Comun Saúde Educ**, p. 139–154, 1998.

DEITEL, P.; DEITEL, H. **Java: Como Programar**. 8. ed. São Paulo: Pearson Prentice Hall, 2010.

FAYEK, M. B.; FARAG, O. S. HICMA: A human imitating cognitive modeling agent using statistical methods and evolutionary computation. **IEEE SSCI 2014 - 2014 IEEE Symposium Series on Computational Intelligence - CIHLI 2014: 2014 IEEE Symposium on Computational Intelligence for Human-Like Intelligence, Proceedings**, 2015.

FILHO, E. E.; RIBEIRO, L. R. DE C. Aprendendo com PBL – Aprendizagem Baseada em Problemas: relato de uma experiência em cursos de engenharia da EESC-USP. **Revista Minerva–Pesquisa e Tecnologia**, v. 6, n. 1, p. 23–30, 2009.

GEE, J. P. Learning by Design: Games as Learning Machines. **Telemeduim. Interactive Educational Multimedia. Anais da Game Developers Conference. San Jose, CA**, v. 52, n. 1 & 2, p. 24–28, 2005.

GIL, A. C. **Métodos e Técnicas de Pesquisa Social**. 6. ed. São Paulo: Atlas, 2008.

GIRAFFA, L.; MULLER, L.; MORAES, M. C. Ensinando Programação apoiada por um ambiente virtual e exercícios associados a cotidiano dos alunos: compartilhando alternativas e lições aprendidas. **Anais dos Workshops do IV Congresso Brasileiro de Informática na Educação (CBIE 2015)**, n. Cbie, p. 1330–1339, 2015.

HOU, S.-I. Integrating Problem-Based Learning with Community-Engaged Learning in Teaching Program Development and Implementation. **Universal Journal of Educational Research**, v. 2, n. 1, p. 1–9, 2014.

MARCONI, M. A.; LAKATOS, E. M. **Fundamentos de Metodologia Científica**. 7. ed. São Paulo: Atlas, 2010.

MEIRA, M. C.; LIMA, M. S. S.; BORGES, M. A. F. Torneios Baseados em Robocode para Incentivar Jovens a Aprender Programação. **Anais dos Workshops do XXXVI Congresso da Sociedade Brasileira de Computação (CSBC 2016)**, n. CSBC, p.

Revista Tecnologias na Educação – Ano 10 – Número/Vol.25 –Julho 2018
tecnologiasnaeducacao.pro - tecedu.pro.br

2403–2412, 2016.

PAPERT, S. **The Connected Family: Bridging the Digital Generation Gap**. Atlanta: Taylor Trade Publishing, 1996.

RECCHIA, T.; CHUNG, J.; POCHIRAJU, K. Performance of heterogeneous robot teams with personality adjusted learning. **Biologically Inspired Cognitive Architectures**, v. 7, p. 87–97, jan. 2014.

YESSAD, A.; LABAT, J. M.; KERMORVANT, F. SeGAE: A serious game authoring environment. **Proceedings - 10th IEEE International Conference on Advanced Learning Technologies, ICALT 2010**, p. 538–540, 2010.

ZANETTI, H. A. P.; OLIVEIRA, L. C. V. Prática 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 2015)**, n. Cbie, p. 1236–1245, 2015.

Recebido em abril 2018
Aprovado em junho 2018