

Gamificação da Prática de *Coding Dojo* no Ensino-aprendizado de Programação de Computadores: Relato de Experiência com Alunos de Curso Técnico Subsequente do IFPA Campus Cametá

José S. Ribeiro

Laboratório de Computação Aplicada - LCA
Instituto Federal de Educação, Básica, Técnica e Tecnológica do Pará
Caixa Postal 68400-000 – Cametá – PA – Brasil

jose.ribeiro@ifpa.edu.br

***Abstract.** Disciplines related to computer programming in computer science courses, are generally received with difficulties by the students, stand out mainly disciplines of Programming Fundamentals and Programming Language I and II. The purpose of this research is to present a gamification of Coding Dojo practice, called Dev Dojo, developed with the purpose of providing the student with a significant, inclusive and contextualized experience in the teaching and learning process of Computer Programming.*

***Resumo.** Disciplinas relacionadas a programação de computadores em cursos da área de informática, são geralmente recebidas com dificuldades pelos alunos, destacam-se principalmente disciplinas de Fundamentos da Programação e Linguagem de Programação I e II. O objetivo desta pesquisa é apresentar uma proposta de gamificação da prática de Coding Dojo, chamada Dev Dojo, desenvolvida com a finalidade de proporcionar ao aluno uma experiência significativa, inclusiva e contextualizada no processo de ensino-aprendizagem de Programação de Computadores.*

1. Introdução

Dentro da matriz de cursos da área de informática as disciplinas relacionadas à programação de computadores são geralmente recebidas com dificuldades pelos discentes. Disciplinas como Fundamentos da Programação e Linguagem de Programação I e II, por exemplo, são consideradas difíceis por muitos estudantes e até mesmo professores, pelo fato destas trabalharem conceitos abstratos, muitas vezes nunca antes estudados pelos alunos (MANZANO, 2014).

Tornar o ensino-aprendizado de programação de computadores significativo e agradável ao estudante é uma missão difícil de ser cumprida, por causa da grande quantidade de conteúdo teórico necessário para execução de aulas práticas das disciplinas desta área (JUNIOR; NAKAMITI; ENGELBRECHT, 2012). Com isto, desenvolver uma metodologia que permita ao aluno compreender os numerosos conceitos teóricos necessários para aulas práticas é um desafio enfrentado por professores da área de informática, já que muitas vezes este problema está ligado diretamente ao tempo de aprendizagem de cada discente juntamente com o tempo de duração da disciplina como um todo (DEITEL; DEITEL, 2010).

Deve-se ressaltar que, em disciplinas relacionadas a programação, o aluno aprende assuntos como: lógica de programação, algoritmos e linguagens de programação. Sendo esta última, de maneira muito superficial, comparada com a capacidade de um indivíduo se comunicar com o computador em uma linguagem específica e padronizada. Neste sentido, deve-se entender que ensinar programação não é nada fácil ou muito menos simples, mas sim um processo complexo, lento e cheio de detalhes/conceitos minuciosos, fatos estes importantes e que devem ser levados em consideração no desenvolvimento de uma metodologia de ensino.

Outra característica de disciplinas voltadas para programação de computadores, está no fato destas serem trabalhadas ao decorrer de todos os semestres de cursos Técnicos em Informática. Desta forma, para que o aluno entenda os conteúdos apresentados no segundo semestre, ele precisa ter sido aprovado em disciplinas pré-requisitos do primeiro semestre. Porém, geralmente isso não ocorre, gerando as dificuldades, que neste caso são levadas como deficiências a serem trabalhadas e superadas nas disciplinas seguintes (DEITEL; DEITEL, 2010) (JUNIOR; NAKAMITI; ENGELBRECHT, 2012).

2. Embasamento do problema

Pesquisas como (SILVA; TRENTI, 2016) e (GOMES, 2010) destacam os problemas enfrentados por professores e alunos na disciplina de Programação de Computadores ou mesmo Linguagem de Programação, buscando apontar alternativas ferramentais que visem viabilizar o processo do ensino-aprendizado do aluno, servindo assim como tecnologias educacionais, neste caso especificamente, no âmbito da tecnologia da informação, ou seja, softwares para auxiliar o aluno e o professor.

Conforme o acima, é evidente a necessidade de práticas metodológicas que permitam ao aluno uma participação de maneira mais ativa e significativa no processo de ensino-aprendizado de programação de computadores. Com isto, propostas com base em Tecnologias Educacionais (BACICH; MORAN, 2017) ou mesmo práticas Gamificadas (PAIVA, 2016), se mostram como alternativas ao problema citado.

Uma tecnologia educacional tem como objetivo principal ser um instrumento de mediação voltado para a viabilização da difusão dos conteúdos disciplinares, assuntos trabalhados dentro de uma ou mais áreas de conhecimento, de maneira a permitir que o aluno envolvido no processo entenda tais conteúdos através da ludicidade, da multidisciplinaridade, da interdisciplinaridade e da contextualização frente a assuntos de seu domínio (ROCHA, 2014).

A proposta aqui defendida se fundamenta como uma tecnologia educacional gamificada, porém com o objetivo de não se confrontar tais conceitos, neste artigo a prática educacional defendida, o *Dev Dojo*, se apresenta como prática gamificada.

Uma prática educacional gamificada consiste no uso de mecânicas de jogos, estética lúdica e de *'game thinking'* com o objetivo principal de engajar pessoas e motivar ações a fim de promover o aprendizado voltado para a resolução de problemas (WILSON et al., 2009). Desta forma, para se aplicar a gamificação como ferramenta pedagógica em sala de aula, o docente necessita de conhecimento prévio sobre o assunto a ser abordado e também ser capaz de entender como a prática trabalhará a motivação nos alunos envolvidos.

Desta forma, esta pesquisa busca propor uma prática de ensino-aprendizagem gamificada, chamada *Dev Dojo*, desenvolvida com a finalidade de proporcionar ao aluno uma experiência motivadora, significativa e contextualizada no processo de ensino-aprendizagem de programação de computadores.

Vale ressaltar que a proposta do *Dev Dojo* já foi apresentada como Tecnologia Educacional em publicação de artigo curto intitulado: “*DEV DOJO: Proposta de Tecnologia Educacional para o Processo de Ensino-aprendizagem de Programação de Computadores*”, porém neste novo artigo aqui descrito pretende-se apresentá-la por meio dos conceitos de gamificação.

3. Coding Dojo

Coding Dojo é um termo utilizado para se referir a reuniões de programadores onde o objetivo é realizar a troca de ideias relacionadas a soluções de programas computacionais e também experiências de desenvolvimento de software adquiridas entre membros de equipe de desenvolvimento de software.

Práticas como o *Coding Dojo* possibilitam que um programador iniciante trabalhe de maneira compartilhada e colaborativa com sua equipe, pois nesta atividade o profissional passa a ter contato direto com diferentes tipos de soluções para um único problema complexo, passando assim a adquirir experiência de membros da equipe com mais tempo na profissão (SATO; CORBUCCI; BRAVO, 2008).

Nesta reunião, Figura 1, normalmente participam os desenvolvedores de software de uma ou mais organizações, é utilizado somente um ou dois computadores e todos participantes focam-se em um problema computacional comum. A duração da reunião dependerá de fatores como complexidade do problema e tamanho da equipe. Deve-se destacar que a referida reunião ocorre de portas fechadas, a fim de facilitar a imersão no problema a ser resolvido (SATO; CORBUCCI; BRAVO, 2008).



Figura 1 - Foto real de um Dojo Coding realizado por uma equipe de desenvolvedores. Fonte: (RAFAEL LAGES, 2013).

4. Dev Dojo

Dev Dojo é a proposta de gamificação defendida por esta pesquisa e seu nome é a junção de dois termos de diferentes idiomas: *Dev*, termo relacionado à palavra inglesa

"development", que significa em português "desenvolvimento", juntamente com a palavra do japonês *Dojo*, que em português significa "Local do Caminho".

A prática gamificada desenvolvida por esta proposta poderá ser adaptada de acordo com as diferentes realidades de alunos e professores que venham a utilizá-la, porém recomenda-se a conservação de suas principais regras, para que a essência da proposta aqui apresentada não seja perdida.

4.1 Regras do *Dev Dojo*

Seleção de dois times de desenvolvimento: O professor responsável seleciona (utilizando critério de sua escolha), dois discentes da turma que terão como responsabilidade realizar a seleção inicial dos times de desenvolvimento; Figura 2.

Utilização de material metodológico: Para desenvolvimento do programa proposto cada equipe irá utilizar 1 computador para codificar a solução; É permitido o uso de papéis, canetas, lápis, cadernos e livros, isto na área de discussão da equipe, Figura 3.

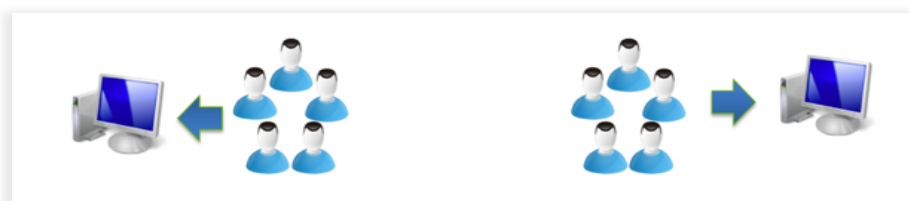


Figura 2 - Representação da seleção de duas equipes. Fonte: Autores.

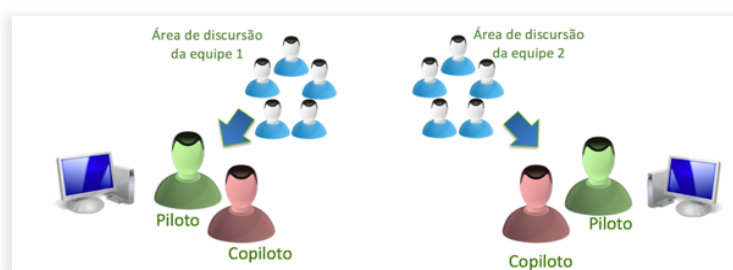


Figura 3 – Piloto, copiloto e Áreas de Discussões das Equipes. Fonte: Autores.

Problema proposto: Deve ser feita a apresentação formal do problema proposto; A descrição do problema deve ser descritiva, indicando as principais características do mesmo, sem deixar de lado os principais aspectos tecnológicos que devem ser trabalhados em sua solução; Cada Equipe deve receber a descrição do problema em um papel e o mesmo deverá ficar exposto no quadro da sala, durante toda a aplicação da tecnologia.

Desenvolvimento do programa e o tempo: Somente o piloto e o copiloto podem ver o desenvolvimento do código, porém esta regra pode ser flexibilizada, dependendo do domínio de programação das equipes quanto ao assunto trabalhado; A prática inclui o conceito de piloto e copiloto, onde piloto é o programador que poderá desenvolver o software e o copiloto é o programador que poderá lhe auxiliar (não podendo programar), durante determinando tempo; Somente pode desenvolver o programa o piloto da equipe; O copiloto deve auxiliar o piloto; O piloto e copiloto mudam em tempos pré-

definidos (a critério do professor, recomendando-se algo entre 3 e 5 minutos); Após o fim do tempo predefinido, o programador vai para a área de discussão da solução de sua equipe e seu copiloto toma seu lugar. Neste momento a equipe envia outro copiloto para auxiliar o novo piloto (Figura 3 e Figura 4); Não é permitido o uso de papel, ou qualquer tipo de consulta material, por parte dos pilotos e copilotos.



Figura 4 - Contagem de tempo realizada. Fonte: Autores.

Organização das equipes: Todos os discentes devem participar; A ordem de trocas de pilotos e copilotos devem ser organizadas por cada uma das equipes; A ordem de trocas dos pilotos e copilotos não deve mudar, mesmo que todos os alunos de cada equipe já tenham passado pela experiência de piloto de sua equipe (Figura 5).

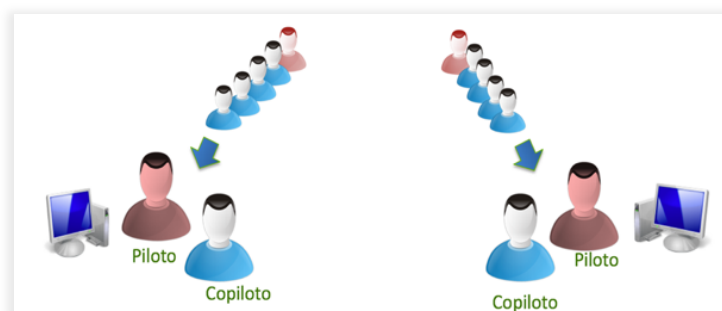


Figura 5 – Ilustração da organização das equipes. Fonte: Autores.

Solução do problema alcançado: Quando uma das equipes achar que o programa está pronto, basta sinalizar ao professor; O professor irá avaliar a solução, porém a contagem de tempo é pausada e a outra equipe não poderá continuar a programar.

Avaliação dos alunos (individual ou mesmo coletiva): O docente pode avaliar os participantes da prática, quanto: Capacidade de programar (Leitura e Interpretação de código do programa; Modificação do programa; Criação de código); Trabalho em equipe (Entendimento do problema; Interação/colaboração com a equipe; Organização).

4.2 Pré-testes

Esta prática gamificada vem sendo desenvolvida e modificada pelo autor desta proposta durante um horizonte de tempo de dois anos. Neste referido tempo, a mesma passou por adaptações durante pré-testes em 4 diferentes turmas, dentro do mesmo contexto dos alunos envolvidos neste trabalho. Vale ressaltar que tais execuções somente ocorreram a nível de pré-testes, para ajuste das regras da prática gamificada.

5. Resultados e Discursões

Como descrito anteriormente, a prática gamificada, objeto desta pesquisa, foi aplicada em duas turmas do curso Técnico Subsequente em Informática do IFPA, Campus Cametá. O problema computacional abordado se tratou de um cadastro de pacientes de uma clínica de saúde, onde através da utilização da linguagem de programação JAVA e da utilização de conceitos de desvio de software e laço de repetição, os alunos conseguiriam resolve-lo, Figura 6.



Figura 6 - Turma da manhã: A esquerda equipes 1 (elipse tracejada) e 2 (elipse pontilhada) durante o Dev Dojo. A direita Destaque para os pilotos e copilotos de cada equipe. Fonte: Autores.

Antes e depois da execução da prática gamificada, consecutivamente, foram coletadas as respostas dos alunos envolvidos para duas perguntas chaves: Pergunta 1 - "Qual a sua opinião sobre a disciplina Linguagem de Programação?" e Pergunta 2 - "Qual a sua opinião sobre o Dev Dojo?".

A Tabela 1 mostra 6 principais relatos feitos por alunos para responder a pergunta 1, o critério utilizado para selecionar tais respostas estão relacionados a representatividade da diversidade das mesmas em meio as repostas de todos alunos. O que significa dizer que as demais repostas, não listadas na Tabela 1, seguem a mesma linha das 6 selecionadas.

Importante, tanto para a pergunta 1 quanto para a pergunta 2 foi solicitado aos alunos que suas respostas individuais fossem escritas em um paragrafo curto para cada pergunta.

Tabela 1 – Seis respostas mais significativas para a pergunta “Qual a sua opinião sobre a disciplina Linguagem de Programação?”

Aluno	Resposta
1	Na minha opinião, a disciplina é muito interessante, a ideia de criar algo do zero é empolgante, Java é uma linguagem boa de se trabalhar, pena que esses imprevistos estão tornando as aulas escassas.
2	Em minha opinião, a disciplina nos ensina coisas fundamentais do Java, princípios para orientação a objetos, classes e JFrame, para que assim, com os ensinamentos da disciplina e o esforço pessoal de cada um, sejamos aptos a fazer um programa mais estruturado.
3	É difícil, mas compreensível, possibilita desenvolver diferentes soluções.

4	Disciplina complicada, mas a estratégia de ensino foi bastante elaborada.
5	A disciplina é bastante complexas mas é importante para a carreira profissional.
6	Não gosto da disciplina, porém sei que é importante.

Na opinião do aluno 1 fica evidente seu interesse pela disciplina dado que o mesmo se sente empolgado por conhecer muito bem a proposta da disciplina e o que ele poderá produzir com os conhecimentos aprendidos nela.

Na opinião do aluno 2, observa-se o interesse do aluno em questão frente a um esforço pessoal realizado por ele.

O aluno 3 reconhece a disciplina como difícil, porém compreensível, a partir do momento que este entende como a disciplina poderá lhe ajudar a resolver problemas.

O aluno 4 entende a disciplina como complicada, buscando se posicionar a favor da metodologia utilizada pelo docente, ou seja, aulas expositivas e práticas com aplicações da prática gamificada *Dev Dojo* em momentos específicos.

Já o aluno 5, entende a disciplina como complexa, porém exalta sua preocupação com sua carreira profissional, ou seja, ele está pensando em seu futuro na profissão de técnico de informática.

O último aluno (6), assume que não gosta da disciplina em questão, porém reconhece a importância desta frente a sua formação. Isto, pode está relacionado ao fato do curso não formar somente técnicos programadores, mas também possibilitar a formação de técnicos voltado para manutenção de microcomputadores e redes de computadores, ambos perfis profissionais estão inseridos dentro da informática.

A Tabela 2 mostra 6 principais relatos feitos por alunos referente a pergunta 2, tais respostas foram produzidas pelos mesmos 6 alunos selecionados anteriormente.

Tabela 2 – Seis respostas para a pergunta “Qual a sua opinião sobre o Dev Dojo?”

Aluno	Resposta
1	“O Dev Dojo em tese seria uma prática interessante para se testar a capacidade de os alunos trabalharem em equipe e sob pressão. Mas na prática, o nível de conhecimento de cada aluno influencia muito na execução da tarefa, visto que após tudo acordado e arquitetado, o indivíduo executa algo completamente diferente, prejudicando o que já havia sido feito e o curto tempo dos que ainda virão apenas para corrigir os erros”.
2	“A prática do Dev Dojo estimula o trabalho em equipe e depende de um trabalho conjunto para conseguir chegar ao objetivo, mas para isso acontecer todos precisam entender o código, e apesar de nenhum fazer igual ao outro, mas todos precisam pelo menos entender o que tá acontecendo. E muitos não entendem o que é repassado pela equipe e acabam fazendo coisas diferentes, o que prejudica quem entendeu o que a equipe tentou repassar e o andamento da prática”.
3	“Desperta a competitividade, instigando o aprendizado individual”.

4	“Método que apresentou resultados positivos e despertou o interesse de todos”.
5	“A pratica do <i>Dev Dojo</i> e importante para o aprendizado e para o trabalho em equipe”.
6	“Gosto da dinâmica, porém preciso estudar mais”.

O aluno 1 se posiciona a favor da prática do *Dev Dojo* quando leva em consideração sua proposta conceitual, porém critica a forma da prática que a metodologia é aplicada. Este relato é importante e deve ser levado em consideração para esta pesquisa, pois são através de críticas que mudanças na prática de ensino são aplicadas.

Deve-se ressaltar que o aluno 1 apresenta avançado conhecimento de programação de computadores, dado que ele é também aluno de outra instituição de ensino, onde já conheceu linguagens como *c/c++*. Tal desconforto apresentado pelo aluno pode ser atribuído com seu auto grau de capacidade de produzir códigos, se comparado com os demais colegas de sala. Deve-se ressaltar que este aluno teve bom rendimento na disciplina.

O aluno 2 demonstra entender bem a proposta do *Dev Dojo* ressaltando que para um melhor sucesso desta prática os demais colegas devem também estar melhor acompanhando a disciplina a fim de se ter resultados mais interessantes. Deve-se ressaltar que este aluno teve bom rendimento na disciplina.

Os alunos 3 e 4 ressaltam os princípios que uma prática gamificada apresenta, pois relatam em suas respostas que o *Dev Dojo* instiga o aprendizado através da ideia de competição. Ou seja, a prática motiva o aluno mesmo que seja de maneira individual. Deve-se ressaltar que estes alunos apresentaram bons rendimentos na disciplina.

O aluno 5 reconhece a prática como importante para seu entendimento da disciplina, assim como importante para seu trabalho em equipe. Deve-se ressaltar que este aluno em questão apresentou rendimento mediano na disciplina.

Por último, o aluno 6, apesar de gostar da dinâmica reconhece que precisa estudar mais. Desta forma, esta pesquisa entende que para o tempo de aprendizagem deste aluno a aplicação da prática *Dev Dojo* pode ter sido muito prematura. Ressalta-se que o aluno em questão apresentou baixo rendimento na disciplina, vindo assim a refazer a mesma no semestre seguinte.

5. Conclusão

Diante dos resultados apresentados por esta pesquisa juntamente com as discussões dos mesmos, afirma-se que a prática gamificada proposta alcançou os objetivos previstos inicialmente, porém encontra desafios importantes a serem superados.

Observa-se nos resultados que alunos com mais facilidade na disciplina podem se sentir incomodados quando requisitados a ensinar alunos que apresentam dificuldades. Porém ambos perfis discentes aprovam a prática de ensino apresentada.

Os resultados também evidenciam que existem alunos preocupados com o nível de conhecimento de seus colegas de equipe, se disposto a explicar a estes como resolver problemas. O que significa que para alguns alunos a aplicação da prática gamificada pode ter ocorrido de maneira prematura durante o semestre.

Pôde-se confirmar que o ambiente propiciado pela execução da prática proposta pelo *Dev Dojo* apresenta aspectos que o torna: integrativo, inclusivo, colaborativo e incentivador de relações interpessoais, este último principalmente pelo fato do mesmo retratar situações profissionais presentes na sociedade.

Contudo, essa pesquisa abordou um relato de experiência da prática gamificada *Dev Dojo*, buscando-se analisar a mesma a partir das opiniões dos alunos. Porém em passos seguintes, sugere-se a aplicação da mesma por vários docentes da área ou até mesmo por docentes de outras áreas, buscando-se, através de uma visão multidisciplinar, melhorar a prática e extrair o máximo de benefícios gerados pela aplicação da mesma no processo de ensino-aprendizagem do estudante.

Referências bibliográficas

- BACICH, L.; MORAN, J. Metodologias Ativas para uma Educação Inovadora: Uma Abordagem Teórico-Prática. [s.l.] Penso Editora, 2017.
- DEITEL, H. M.; DEITEL, P. J. Java como programar. São Paulo (SP): Pearson Prentice Hall, 2010.
- GOMES, A. Dificuldades de Aprendizagem de Programação de Computadores: Contributos para sua Compreensão e Solução. [s.l.] Universidade de Coimbra, 2010.
- JUNIOR, D.; NAKAMITI, G.; ENGELBRECHT, A. Algoritmos e Programação de Computadores. [s.l.] Elsevier Brasil, 2012.
- MANZANO, J. Java 8: Programação de Computadores - Guia Prático de Introdução, Orientação e Desenvolvimento. 1. ed. [s.l.: s.n.].
- PAIVA, C. A. Pós-graduação: Docencia no Ensino Superior a Gamificação como Ferramenta Pedagógica. 2016.
- RAFAEL LAGES. Coding Dojo 2013 - globo.com, 2013. Disponível em: <<https://www.youtube.com/watch?v=vqnwQ3oVM1M>>. Acesso em: 6 maio. 2018
- ROCHA, H. A utilização de Tecnologia Educacionais enquanto mediador semiótico na sala de aula: visibilizando as diversidades. Belém: IFPA, p. 250p, 2014.
- SATO, D. T.; CORBUCCI, H.; BRAVO, M. V. Coding Dojo: An Environment for Learning and Sharing Agile Practices. Agile 2008 Conference. Anais... In: AGILE 2008 CONFERENCE. ago. 2008
- SILVA, B.; TRENTI, M. Dificuldades no Ensino-Aprendizagem de Programação de Computadores: Contribuições para a sua Compreensão e Resolução. V Sinect. n. V, 2016.
- WILSON, K. A. et al. Relationships Between Game Attributes and Learning Outcomes: Review and Research Proposals. Simulation & Gaming, v. 40, n. 2, p. 217–266, abr. 2009.