

Uso da técnica *Coding DOJO* em aulas de programação de computadores

Adriana Paula Zamin Scherer¹, Filipino Novo Mór¹

¹Faculdade Dom Bosco de Porto Alegre

Rua Marechal José Inácio da Silva – 90520-280 – Porto Alegre – RS – Brasil

adriana.scherer@gmail.com, professorfilipo@gmail.com

Abstract. *This article describes the realization of a DOJO Coding experiment in the Algorithms and Computer Organization and Architecture courses during the first term of the Bachelor of Information Systems program. By demanding previous preparation of the proposed exercises, the results bespeak the majority of students felt motivated with the activity and have demonstrated collaborative behavior. We also observed that students with less fluency in programming were accepted and assisted by their workgroups.*

Resumo. *Este artigo descreve a realização de um experimento de Coding DOJO nas disciplinas de Algoritmos e Organização e Arquitetura de Computadores junto as turmas de primeiro semestre do Bacharelado em Sistemas de Informação. Exigindo preparação prévia dos exercícios propostos aos alunos, os resultados sugerem que a maioria dos alunos se sentiram motivados com a atividade e demonstraram comportamento colaborativo. Observou-se, também, que estudantes com menor fluência em programação foram aceitos e assistidos pelos seus grupos de trabalho.*

1. Introdução

As novas formas de ensinar sugerem o conhecimento como algo que é construído num contexto de trocas, com o desenvolvimento da capacidade de interação, negociação e resolução de problemas entre os alunos. Essas propostas reconhecem que os alunos devem exercer o papel fundamental para o desenvolvimento de suas habilidades sociais e cognitivas que culminarão na consolidação de sua base de conhecimento. Com isso, o processo ensino-aprendizagem deixa de estar centrado no professor e passa a estar centrado no aluno. O professor assume, então, um novo papel que é o de facilitador do processo, criando contextos, ambientes e oportunidades em que os alunos possam desenvolver suas habilidades e seus conhecimentos de modo criativo e com a colaboração entre os colegas [Franz, Da Silva e Cheiran, 2014].

O *Coding DOJO* surge como uma técnica que possibilita a aprendizagem colaborativa, essencialmente realizado em pares, com o objetivo de resolver problemas de programação, cujas principais características remetem a um ambiente divertido, colaborativo e não competitivo em que podem ser testadas novas ideias e todos os alunos participam, evitando que os alunos trabalhem sozinhos em seus computadores [Franz, Da Silva e Cheiran, 2014].

O presente trabalho relata as experiências de uma atividade prática utilizando a técnica de *Coding DOJO* nas disciplinas de Algoritmos e Arquitetura e Organização de Computadores de um Curso de Sistemas de Informação em uma instituição privada de ensino superior. Espera-se que a partir dos relatos e resultados aqui apresentados, novas experiências

possam ser replicadas e comparadas, de forma que a técnica *Coding DOJO* possa ser vista como uma alternativa válida para aulas de programação de computadores.

Este trabalho apresenta-se dividido em seções: após esta Introdução, a seção 2 apresenta o referencial teórico acerca da técnica de DOJO. Na seção 3 são apresentados os métodos e técnicas utilizadas durante o experimento aqui relatado. Na seção 4 são apresentados os resultados que foram coletados e suas análises e, por fim, na seção 5 são relatadas as considerações finais.

2. A técnica *Coding DOJO*

De acordo com Carmo e Braganholo (2012), o *Coding DOJO* pode ser definido como um encontro de programadores reunidos com o objetivo de solucionar um desafio de programação. Já Marinho *et al.* (2016) afirmam que o DOJO é uma técnica que pode ser utilizada com o propósito de possibilitar que os alunos ganhem prática e conhecimento em uma tecnologia estabelecida.

O planejamento e a execução de um *Coding DOJO* exigem que quatro etapas sejam seguidas, (a) apresentação: momento em que a técnica é apresentada com o objetivo de deixar os participantes confortáveis para a execução da tarefa. É importante destacar os princípios da técnica, bem como estimular o respeito e a necessidade de colaboração entre os participantes; (b) acordo entre participantes: nesta etapa define-se em comum acordo sobre qual deve ser o foco do trabalho e o que cada um deve levar como aprendizado; (c) codificação: é a etapa principal da técnica em que os participantes trabalham em conjunto para codificar a solução do desafio proposto; (d) retrospectiva: após a conclusão da codificação, realiza-se a reflexão sobre o que foi aprendido, quais as dificuldades e facilidades, entre outros aspectos, que ocorreram durante a execução da prática [Schoeffel, Rosa e Waslawick, 2016].

O *Coding DOJO* pode ser aplicado em formatos diferentes (a) *Randori*: significa “estilo livre” em uma tradução livre do idioma japonês. Nela, apenas um computador é utilizado para a tarefa, sendo que sua tela é projetada para que todos possam ver. São escolhidos dois participantes, um será o piloto (codificador) e outro será o copiloto (auxiliar). Cada ciclo de programação tem duração de poucos minutos e ao término deste tempo o piloto retorna à plateia, o copiloto assume como codificador e um integrante da plateia é convidado a assumir o posto de copiloto. A plateia permanece atenta à solução e pode discutir entre si sobre a solução do problema, no entanto, sem influenciar o piloto e copiloto; (b) *Kata*: significa “forma” no idioma japonês e remete ao conceito de exercício e repetição. Este formato possui foco no aperfeiçoamento. Sugere-se a repetição dos desafios para que ocorra o aperfeiçoamento da solução a cada nova rodada. Aqui também pode ser utilizado o formato dos pares, assim como no *Randori*; (c) *Kake*: significa “aposta” em uma tradução livre do idioma japonês. É semelhante ao *Randori*, mas aqui caracteriza-se pela existência de diversas duplas trabalhando em estações diferentes para resolver o mesmo problema. Neste modelo, elimina-se a ociosidade da plateia e um possível constrangimento do piloto ao seu trabalho devido a observação da plateia. O ciclo de programação segue com (1) inversão de papéis de piloto e copiloto em cada dupla, e (2) troca de piloto entre as duplas [Schoeffel, Rosa e Waslawick, 2016 e Marinho *et al.*, 2016].

3. Métodos e técnicas

Carmo e Braganholo (2012), Marinho *et al.* (2016) e Luz e Neto (2012) destacam em seus trabalhos as mais diversas experiências relacionadas ao uso da técnica de *Coding DOJO* em disciplinas dos cursos da área da Tecnologia da Informação, principalmente o seu uso em disciplinas de programação. No entanto, são iniciativas isoladas dentro das disciplinas dos cursos. Neste trabalho apresentamos o uso em conjunto do *Coding DOJO* em duas disciplinas de programação de computadores dispostas no primeiro semestre do curso que se situam no eixo de formação básica do programa, como uma atividade de aula prevista no plano de ensino das disciplinas, sem fins avaliativos. Neste contexto, os alunos, na sua maioria, não possuem experiência prévia em programação, de forma que o DOJO foi executado mais próximo ao fim do semestre. Estas atividades ocorreram com diferença de 8 (oito) dias entre uma e outra, sendo que a experiência na disciplina de Arquitetura e Organização de Computadores ocorreu primeiro.

3.1. Algoritmos

Na disciplina de Algoritmos foi utilizado o formato *Randori* do *Coding DOJO* e seguiram-se as etapas definidas para a técnica, conforme descrito na seção 2. Participaram desta atividade 10 (dez) alunos e foi utilizada a linguagem de programação C para a codificação, com o uso da IDE DevC++. Os desafios envolviam o uso de vetores e foram propostos 02 (dois) problemas para serem resolvidos, sendo que o primeiro possuía grau de dificuldade menor do que o segundo. A partir da apresentação do primeiro problema, os alunos estabeleceram um processo de voluntariado para os papéis de piloto e copiloto da dupla inicial. Uma vez estando definida a dupla inicial de trabalho, todos os alunos participaram de um momento de discussão sobre o problema em um tempo de 20 (vinte) minutos sendo elencados os principais aspectos e possíveis abordagens de solução. A etapa de codificação foi organizada em ciclos de 05 (cinco) minutos, totalizando 50 (cinquenta) minutos. Este mesmo processo foi utilizado para o segundo problema. Ambos os programas elaborados durante a atividade resolveram de maneira adequada os problemas propostos, não apresentando erros de lógica.

3.2. Arquitetura e Organização de Computadores

Participaram desta atividade 12 (doze) alunos. O experimento foi realizado com a aplicação de duas etapas distintas: na primeira, tradicional ao formato DOJO Kake, os alunos foram divididos em três grupos com quatro integrantes cada e receberam a missão de desenvolver, a partir do zero, um código em linguagem C implementando estruturas de repetição, controle e decisão e tomou apenas um período de aula. Na segunda etapa, os grupos tiveram seus integrantes alterados por sorteio, recebendo a missão de implementar, desta vez em linguagem Assembly, os códigos gerados na primeira etapa. Para tal, o professor realizou breve explanação, descrevendo as estruturas mais difíceis de serem implementadas. A segunda etapa exigiu os últimos 3 períodos da aula, sendo que todos os grupos tiveram sucesso na entrega de seus códigos. Ao fim, cada grupo fez uma breve apresentação, descrevendo os códigos mais complexos e justificando as escolhas realizadas. Os trabalhos foram executados utilizando-se a linguagem ANSI C com a IDE Dev C++ para ambiente Windows.

4. Resultados e discussão

No final das aulas em que os experimentos foram executados, os alunos foram convidados a responder um questionário avaliativo acerca da atividade desenvolvida. Os resultados foram tabulados, analisados e serão aqui apresentados. O questionário avaliativo foi composto por seis questões, das quais três foram exploradas neste trabalho. Em Mór e Scherer (2020), encontra-se o formulário completo utilizado, bem como as respectivas respostas e uma análise mais aprofundada do estudo de caso.

4.1. Algoritmos

A questão 1 tinha o objetivo de compreender o grau de satisfação dos alunos ao participar do Coding DOJO. As respostas foram tabuladas, organizadas e definiram-se como 40% “*muito satisfeito*”, 50% “*bastante satisfeito*” e 10% “*mais ou menos satisfeito*”. Os conceitos “*pouco satisfeito*” e “*indiferente*” obtiveram 0%. A figura 1 ilustra o resultado obtido.

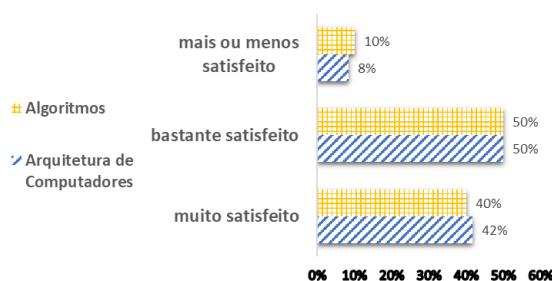


Figura 1. Grau de satisfação dos alunos com a atividade realizada.

A questão 2 investigava os motivos de satisfação dos alunos ao participar da atividade utilizando a técnica Coding DOJO. As respostas foram: “*Tivemos uma breve noção de como é trabalhar em um projeto com programadores e lógicas distintas*”, “*deu para ter uma noção de como funciona o sistema de trabalho em empresas*”, “*Simula o ambiente encontrado em empresas, trabalhando um grupo na resolução de desafios*”, “*Foi possível trabalhar em conjunto*”, “*Além de ser divertido, a turma toda se empenhou para resolver o problema*”, “*Estimula o desempenho e desenvolvimento para atividades em grupo*”, “*a demonstração de conhecimento impulsiona os demais a conhecer mais*”, “*mostrou o quanto somos iniciantes para criar um código "aberto" para outros arrumarem*”, “*porque foi divertido e interagimos com toda a turma*”, “*demonstrou que posso trabalhar bem em equipe*”. A questão 3 investigava os motivos de frustração dos alunos ao participar da atividade utilizando a técnica Coding DOJO. Apenas 06 (seis) alunos registraram respostas e são elas: “*não fiquei frustrado*”, “*não tivemos muito tempo para organizar as ideias, pois cada um tinha um pensamento diferente*”, “*o grupo sentiu uma dificuldade de se organizar*”, “*nada*”, “*foi um pouco frustrante e que não foi muito tempo para conversar*”, “*é difícil programar na frente dos colegas, mas é interessante, pois é assim na vida*”.

4.2. Organização e Arquitetura de Computadores

A questão 1 solicitou o grau de satisfação com a atividade realizada, tendo obtido o seguinte resultado: “*muito satisfeito*” 42%, “*bastante satisfeito*” 50%, “*mais ou menos satisfeito*” 8%, “*pouco satisfeito*” e “*indiferente*” com 0%, conforme ilustra a Figura 1.

A questão 2, aberta, perguntou o motivo da satisfação com a aula ou método utilizado, tendo obtido as seguintes respostas: “*pude adquirir conhecimento*”, “*umenta a participação dos alunos na aula*”, “*estimula o raciocínio lógico*”, “*me senti por dentro da aula*”, “*executar*

o que estamos aprendendo”, “facilita o aprendizado”, “oferece didática diferenciada”, “o conhecimento demonstrado elucidada as dúvidas das aplicações do hardware”, “explicação boa, super atencioso, ajuda quando preciso”, “foi interessante”, “a prática ajuda na aprendizagem e fixação do conteúdo”. A questão 3 perguntou o motivo de frustração com a aula ou método: “é necessário focar mais em vetores e praticar com mais calma”, “não pude entender de uma maneira adequada”, “não rodou conforme era para ter rodado o programa e não consegui consertar”, “algumas coisas não consegui entender”.

5. Considerações finais

Os resultados sugerem que a técnica adotada conseguiu motivar a maioria dos alunos, bem como permitiu a avaliação, por parte dos professores, da evolução dos alunos em relação ao conteúdo proposto. Como pontos positivos, destaca-se: o entusiasmo demonstrado pelos alunos durante a realização das tarefas; a formação de laços entre os estudantes e também entre esses e os professores; o fomento da colaboração em detrimento da simples competição; o trabalho em equipe e a comunicação. Como pontos de atenção, tivemos: a necessidade de infraestrutura computacional adequada e preparada com antecedência; a exigência de que os alunos já possuíssem certa fluência na programação; e a necessidade de um preparo prévio do professor, visando antecipar os problemas de ambiente (computacional) e eventuais dúvidas que poderiam surgir. Finalmente, notou-se que os alunos que apresentaram maiores dificuldades durante a atividade receberam apoio de seus grupos de trabalho, e mesmo dos outros grupos. Aparentemente, o foco em colaboração fomentou a comunicação e o auxílio mútuo entre os estudantes.

Referências

- CARMO, D.H., BRAGANHOLO, V. (2012) “Um Estudo Sobre o Uso Didático de Dojos de Programação”. Anais do XX Workshop sobre Educação em Computação, Curitiba, Paraná.
- FRANZ, L. P., DA SILVA, J. P. S., CHEIRAN, J. F. P. (2014). “O uso de Coding Dojo no aprendizado colaborativo de programação de computadores”. RENOTE , v. 12, n. 2.
- MARINHO, C. S. S., MOREIRA, L. O., COUTINHO, E. F., PAILLARD, G. A. L., NETO, E. T. L. (2016) “Experiências no Uso da Metodologia Coding Dojo nas Disciplinas Básicas de Programação de Computadores em um Curso Interdisciplinar do Ensino Superior”. Anais dos Workshops do V Congresso Brasileiro de Informática na Educação (CBIE 2016).
- MÓR, F. N., SCHERER, A. P. Z., (2020) “Relatório do Experimento: Uso da técnica Coding DOJO em aulas de programação de computadores”, http://www.filipomor.com/material_apoio/rpt_Coding_DOJO_2020.pdf, Junho de 2020.
- LUZ, R. B., NETO, A. (2012) “Usando Dojos de Programação para o Ensino de Desenvolvimento Dirigido por Testes”. Anais do 23º Simpósio Brasileiro de Informática na Educação (SBIE 2012).
- SCHOEFFEL, P., ROSA, D. F., WASLAWICK, R. S. (2016) “Um Experimento do Uso de Coding Dojo na Aprendizagem de Programação Orientada a Objetos”. Revista Brasileira de Sistemas de Informação, Rio de Janeiro, vol. 9, No. 2, pp. 71-105, 2016