# Ensino de Programação Remoto com Dojo de Programação Usando Método Randori

**Andreina Maria Mendes da Silva, Anderson Luis Bento Soares, Erick Correia Silva, Brena Rodrigues Machado, Carla Ilane Moreira Bezerra**

[1]Universidade Federal do Ceará (UFC) – Quixadá – CE – Brasil

{andreinamendes63, andersonlbsoares, erickbastos.cs, brenarodrigues}alu.ufc.br

carlailane@ufc.br

***Abstract.*** *This paper describes an experience report in the application of the Dojo Programming method remotely, in the teaching of Programming Fundamentals (FUP) and Object Oriented Programming (OOP) concepts. The action was developed by the Tutorial Education Program (PET) for two semesters, and its main goal was to help first-year undergraduate IT students in developing their knowledge base in programming. Based on the evaluation of the activity, it was noted the good use of the content by the participants and the need for greater contact between students and mediators during the resolution of the activities presented.*

***Resumo.*** *Este artigo descreve um relato de experiência na aplicação do método Dojo de Programação de forma remota, no ensino de conceitos de Fundamentos de Programação (FUP) e Programação Orientada a Objetos (POO). A ação foi desenvolvida pelo Programa de Educação Tutorial (PET) durante dois semestres, e teve como principal objetivo auxiliar estudantes do primeiro ano de graduação, de cursos de TI, no desenvolvimento de sua base de conhecimento em programação. Baseado na avaliação da atividade, notou-se o bom aproveitamento do conteúdo por parte dos participantes e a necessidade de um maior contato entre os alunos e os mediadores durante a resolução das atividades apresentadas.*

## 1. Introdução

Em 2020, o número de cursos de tecnologia disponibilizados pelas instituições brasileiras foi de aproximadamente 5078, valor apresentado no Censo da Educação Superior [INEP 2022], um aumento de 280% em relação ao ano de 2015, com 1809 cursos [INEP 2016]. Este aumento ocorre devido a evolução constante do meio tecnológico, impulsionada a partir de 2020 com as adaptações necessárias ao ensino e trabalho remoto, onde a tecnologia se tornou uma ferramenta indispensável, causando um aumento na demanda por novos profissionais capacitados para atuar na área além de uma maior visibilidade e demanda de cursos de tecnologia.

As disciplinas iniciais de programação são básicas para a formação de muitos profissionais da área da Tecnologia da Informação (TI), pois abordam os princípios da lógica de programação que visam desenvolver a capacidade de análise e resolução de problemas.

DOI: 10.5753/sbie.2022.224820 

Porém, o processo de aprendizado nessas disciplinas não é simples, tendo em vista que os conceitos aplicados e a carga de conhecimento são de difícil compreensão. As disciplinas introdutórias de programação, possuem a maior taxa de reprovação e por consequência há um aumento da taxa de evasão em cursos da área de Computação [Alves et al. 2019a].

No processo de ensino e aprendizagem em programação ressaltam-se alguns problemas, como: a falta de metodologia de ensino de professores e a desmotivação de alunos [Arimoto and Oliveira 2019]. Para contornar essa problemática destacam-se alguns métodos que auxiliam no aprendizado como visualização, colaboração e instrução Ancorada [Souza et al. 2016]. Tendo em vista os métodos citados, e com intuito de auxiliar os alunos no aprendizado de programação, foi criada uma atividade de ensino com foco na aprendizagem colaborativa utilizando o método Dojo de Programação. Segundo [Carmo and Braganholo 2012], a literatura define Dojo de Programação como "um método de encontro de programadores visando a solução de um desafio de programação". Assim, Dojo é um local de treinamento de conceitos e práticas em programação. O termo Dojo de Programação também é conhecido na sua versão em língua inglesa *Coding Dojo*.

Os desafios passados no Dojo de Programação geralmente são problemas de lógica de programação, ou atividades que estimulam o aprendizado de conceitos da área. O Dojo pode trabalhar com tópicos centrais, onde cada encontro aborda determinado assunto. Dentro do método Dojo de Programação existem diversas técnicas e meios de aplicar o método a grupos de pessoas. As mais usadas são as técnicas "Randori", "Kata" e "Kake" [Moreira de Castro and Siqueira 2019].

As três técnicas possuem elementos em comum, como a presença de um apresentador encarregado de guiar os encontros. Cada formato possui sua própria característica, por exemplo o formato "Kata" o apresentador deve demonstrar uma solução pronta, previamente desenvolvida, o objetivo é que todos os participantes sejam capazes de reproduzir a solução alcançando o mesmo resultado, sendo permitido realizar interrupções para sanar dúvidas a qualquer momento. Já no formato "Randori" é proposto um problema a ser resolvido e a programação é realizada em apenas uma máquina, por pares. A pessoa que está codificando é o piloto, e seu par o co-piloto. A cada intervalo de tempo proposto o piloto volta para a plateia e o co-piloto assume a condição de piloto, e uma pessoa da plateia passa a assumir a posição de co-piloto. Por último a técnica "Kake" é um formato semelhante ao Randori, porém há várias duplas trabalhando simultaneamente. A cada turno as duplas são trocadas, promovendo a integração entre todos os participantes do evento. Nesse formato um conhecimento mais avançado dos participantes é necessário [Moreira de Castro and Siqueira 2019].

Neste contexto, este artigo tem como objetivo apresentar um relato de experiência da aplicação do método dojo Randori no ensino de programação para alunos dos primeiros semestres dos cursos de TI da Universidade Federal do Ceará (UFC) - Campus Quixadá. A pesquisa foi realizada por bolsistas do Programa de Educação Tutorial (PET) - Tecnologia da Informação de forma remota, sendo dividida em duas edições com a apresentação de conteúdos associados a Fundamentos de Programação (FUP) (primeira edição) e associada a Programação Orientada a Objetos (POO) (segunda edição).

Este artigo está estruturado da seguinte forma: apresentação dos trabalhos relacionados, utilizados como base, descrição da metodologia e ferramentas aplicadas na

pesquisa, execução das etapas da pesquisa, avaliação dos resultados obtidos, discussões, conclusão e trabalhos futuros associados à pesquisa.

## 2. Trabalhos relacionados

Em [de Oliveira et al. 2018], os autores apresentam uma visão geral dos benefícios e desafios que a aprendizagem colaborativa traz. A metodologia usada foi a realização de várias sessões durante 18 meses para poder-se concluir que a técnica *Dojo de programação* cria um ambiente agradável para discussões de ideias, compartilhamento de informações e inclusive reduz a curva de aprendizado dos envolvidos e eleva o nível de qualidade do software desenvolvido. No trabalho fica evidente que essa metodologia de ensino traz benefícios abundantes, tanto para o projeto a ser construído como para o estudante que está desenvolvendo.

[Scherer and Mór 2020] apresentam um experimento realizado nas disciplinas de Algoritmos e Organização e Arquitetura de Computadores nas turmas de primeiro semestre do curso de Sistemas de Informação. O artigo tem como principal objetivo motivar e assistir os alunos com maiores dificuldades, assim evidenciando uma evolução geral da turma a partir do uso da metodologia Dojo de Programação.

[Alves et al. 2019b] apresentam uma abordagem de aprendizagem cooperativa baseada na utilização do *Dojo de Programação* como metodologia de ensino. Essa estratégia influencia positivamente nos aspectos de motivação, colaboração e percepção de aprendizagem. A metodologia foca na compreensão mais profunda desses aspectos através de um estudo de caso de duas disciplinas introdutórias a programação, ofertadas pelos cursos Licenciatura em Ciência da Computação e Bacharelado em Sistemas de Informação.

[Calderon et al. 2021] trazem o ensino sob uma diferente ótica, onde o aluno sai do papel de aprendizagem passiva e passa a ser o principal responsável pela sua aprendizagem. Através de um Mapeamento Sistemático da Literatura, foi construído um panorama sobre as metodologias ativas usadas para o ensino de programação na educação superior. O estudo evidencia as qualidades dessas metodologias, como a Gamificação tornando as atividades de programação exercícios descontraídos e leves, e o Dojo de Programação aumentando o engajamento e melhorando o desempenho dos alunos.

[Da Silva Garcia et al. 2021] apresentam como problemática principal a alta complexidade e difícil compreensão dos conteúdos de programação, o que causa uma grande evasão nos cursos de nível superior. Como solução, o artigo apresenta as metodologias ativas que trazem o aluno para o papel de responsável pela sua educação. O trabalho resulta em um conjunto de aulas teóricas e práticas divididas em módulos.

Em [Silva et al. 2020], os autores desenvolvem um mapeamento sistemático com o objetivo de compreender os pontos positivos da gamificação no ensino de programação de computadores. No trabalho fica evidente melhorias no desempenho, motivação e engajamento em grupos que tiveram contato com a gamificação.

Todos os artigos citados evidenciam os benefícios da utilização do Dojo de Programação como uma ferramenta de ensino para todos os públicos. A utilização dessa metodologia ativa, coloca o aluno como principal responsável do seu aprendizado tendo ganhos consideráveis em comparação com a forma tradicional de ensino, da mesma forma que o presente trabalho.

## 3. Metodologia

O Dojo de Programação teve como objetivo auxiliar os alunos de primeiro e segundo semestre dos cursos de TI no aprendizado dos conteúdos das disciplinas de FUP e POO, por meio da promoção de um ambiente de aprendizagem colaborativa no contexto remoto. Portanto, foi necessário a divulgação da atividade frisando que os participantes deveriam fazer uso de suas próprias máquinas. Além disso, foi necessária a utilização de uma técnica específica para garantir o desenvolvimento da atividade considerando o cenário em questão, logo, foi escolhida a técnica Randori, que possibilitaria a execução da atividade de forma a garantir a participação de todos os participantes.

A atividade foi realizada no ano de 2021 entre os meses de Maio e Agosto (primeira edição) e de Setembro e Janeiro (segunda edição), com encontros semanais realizados de forma remota em cursos da UFC - Campus Quixadá. Foram aplicados pelos bolsistas do PET TI, desafios de programação associados a FUP e POO, focando nos conhecimentos de Lógica de Programação adquiridos pelos participantes em disciplinas anteriores e/ou atuais. Para isso, foi elaborada uma metodologia dividida nas etapas definidas na Figura 1.

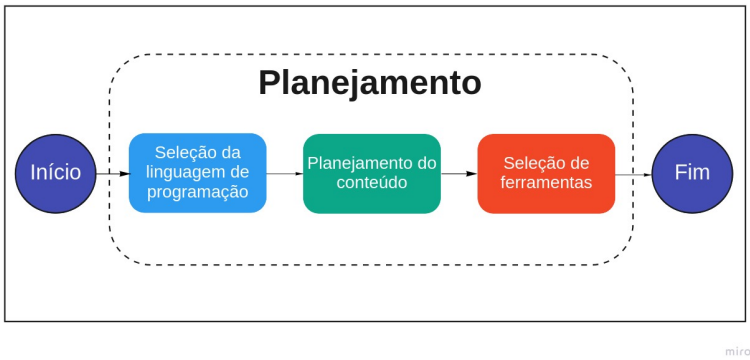


**Figura 1. Fluxo de planejamento do Dojo de programação**

**Seleção da linguagem de programação**. Foram divulgados entre os alunos, formulários de inscrição para o Dojo de FUP[1] com 154 inscritos e o Dojo de POO[2] com 51 inscrições, em dois semestres diferentes, com o objetivo de coletar informações sobre a intenção e aspectos técnicos para a execução da atividade. Um dos aspectos estava associado à linguagem que seria utilizada na resolução dos desafios, devido a diferença entre as metodologias adotadas pelos docentes. Os alunos deveriam escolher a linguagem por critérios particulares, seja a linguagem que o seu professor estivesse utilizando na disciplina, seja por haver uma maior familiaridade.

No Dojo de FUP, a maioria dos alunos optou pela linguagem Python. Já no Dojo de POO, C++ e Java foram as linguagens predominantes entre as respostas. Portanto, foram utilizadas estas linguagens para a resolução das questões e repasse de conteúdos em cada edição. No caso do Dojo de POO, a decisão de abordar duas linguagens, se deu ao fato da votação equilibrada de ambas as linguagens e a equipe de realização da atividade estar em uma quantidade maior em relação ao Dojo de FUP.

[1]https://bityli.com/fup
[2]https://bityli.com/poo

**Planejamento do conteúdo**. Para o conteúdo do Dojo de programação, foi decidido abordar os principais conteúdos vistos nas disciplinas de FUP e POO, e aqueles essenciais para o entendimento intuitivo do aluno acerca da estrutura padrão de qualquer linguagem. Para isso, foram utilizados repositórios de questões da internet como o URI[3] e o Neps Academy[4] no Dojo de FUP. Por ser a disciplina inicial de programação, foram utilizados desafios de animação e desenho em alguns conteúdos com o objetivo de facilitar o entendimento. No Dojo de POO, foram utilizados como base os repositórios[5] da disciplina mantidos por professores no Github, contendo miniprojetos e conteúdos explicativos, além de cursos *online*[6], comumente utilizados pelos alunos. O material utilizado para a explicação foi feito pelos organizadores e as atividades eram retiradas do Github. A Tabela 1 descreve todos os conteúdos adicionados a ementa de ambos os Dojos.

**Tabela 1. Ementa das edições**

| Dojo FUP | Dojo POO |
|---|---|
| Variáveis | Objetos |
| Operadores de atribuição | Classes |
| Operadores aritméticos | Métodos construtores |
| Operadores relacionais | Métodos assessores e modificadores |
| Estruturas de condição | Encapsulamento |
| Estruturas de repetição | Herança |
| Funções | Classes abstratas |
| Recursividade | Polimorfismo |

**Seleção de ferramentas**. Para a comunicação durante os encontros, devido ao contexto remoto, era necessária uma ferramenta de comunicação que permitisse o repasse de informações, seja por voz, vídeo ou texto. Para isso, foi escolhido o Discord[7], ferramenta que permite criar servidores, e nestas categorias com canais de texto e voz no qual podem ser realizadas chamadas de vídeo e também a apresentação de tela adaptável. Além disso, o Discord possibilita a criação de cargos com permissões específicas definidas pelo criador do servidor, limitando assim a visibilidade dos participantes e possibilitando a criação de canais privados de planejamento.

Para o desenvolvimento de códigos no Dojo de FUP, foi utilizado o Replit[8], uma IDE web colaborativa e gratuita que possui suporte para mais de 50 linguagens, que permitiu a criação e edição de documentos *online* em tempo real por usuários convidados. Buscando uma diversificação na aprendizagem durante os encontros, foi utilizada uma metodologia de ensino com animação, usando a biblioteca da linguagem Python chamada pygame[9], que possui suporte para os principais sistemas operacionais, e permite a criação de cenários e objetos em duas dimensões, além de possuir suporte para eventos externos (e.g., controle e eventos do *mouse*).

No Dojo de POO, foi decidido o uso do Visual Studio Code[10], um editor de texto

[3]https://www.urionlinejudge.com.br/
[4]https://neps.academy/
[5]https://github.com/qxcodepoo/arcade
[6]https://www.youtube.com/watch?v=KlIL63MeyMYlist=PLHz$_{A}reHm4dkqe2aR0tQK74m8SFe-aGsY$
[7]https://discord.com/
[8]https://replit.com/
[9]https://www.pygame.org/
[10]https://https://code.visualstudio.com/

*desktop* gratuito que possui suporte para mais de 30 linguagens de programação. Nessa ferramenta, é utilizada a extensão Live Share que é um *plugin* de edição de documentos de forma colaborativa e simultânea que permitiu a edição e visualização de documentos *online*.

## 4. Execução do Dojo de Programação

Os encontros, para ambas as edições do ensino de dojo de programação, ocorreram em 4 etapas.

Na primeira etapa, ministrou-se o conteúdo associado às disciplinas, repassado de forma conceitual. Nesta etapa, todos os alunos eram alocados em uma mesma sala de voz da plataforma Discord, o Dojo de FUP contava em média participantes em cada encontro e o Dojo de POO com 20 participantes. Para este momento, havia no Discord, um chat de texto geral para discussões sobre o conteúdo e demais assuntos, muitas vezes associados a descontração dos participantes em momentos com poucas ações por parte dos mediadores.

A segunda etapa, é referente à resolução de questões associadas ao conteúdo apresentado e as dúvidas existentes, utilizando o método Randori de Dojo. Antes da resolução das questões, um dos responsáveis realizava a leitura e explicação da atividade em questão para todos os participantes. A partir deste momento, apresenta-se a principal diferença entre o Dojo de FUP e o Dojo POO, já que no primeiro os alunos eram mantidos na mesma sala do Discord utilizando a linguagem Python, e no segundo, os participantes eram divididos em salas associadas a linguagem que cada um havia escolhido (Java ou C++), na sala de Java em média com 10 participantes, e na sala de C++ com 5 a 10 participantes. É importante ressaltar que a equipe de responsáveis era dividida entre estas salas para realizar o acompanhamento das resoluções, dentre as linguagens escolhidas, dominavam mais.

Na terceira etapa, de forma geral, era solicitada a criação de duplas por parte dos participantes, para definir quem seria o piloto e o co-piloto, quando necessário os responsáveis auxiliavam nessa divisão. É importante ressaltar que na aplicação do método Randori original, quando o tempo de uma dupla chega ao fim, o piloto sai e o copiloto assume o seu lugar dando espaço a um novo participante. Neste caso, como o número de pessoas era maior, uma dupla era apenas substituída por outra, no momento em que o tempo era finalizado a dupla se dissolvia e cada membro poderia participar novamente como integrante de uma outra dupla, a mudança nestas situações estava dependente da quantidade de participantes e do envolvimento dos mesmos com as atividades do encontro, mas é importante citar que todos os membros participavam de ao menos uma rodada da resolução.

Na última etapa, o piloto e o co-piloto discutiam sobre a lógica e desenvolvimento do código da solução no canal de voz, enquanto o público assistia e discutia no canal de texto disponibilizado para a sua linguagem escolhida. Caso necessário, os responsáveis poderiam auxiliar a dupla em questão, como lembrar as regras da atividade, fornecer dicas para o tratamento de erros e quaisquer outros pontos necessários para o desenvolvimento do código. Além disso, eram responsáveis pela contagem do tempo de cada dupla, para haver uma mudança e manter a continuidade do ciclo da técnica. Normalmente o tempo era definido como 7 minutos, quando este chegava ao fim, a dupla explicava o que desen-

volveu para todos, e assim a nova dupla podia continuar a resolução. Em cada encontro era resolvida no mínimo uma atividade, e assim, a cada semana os participantes tinham uma experiência a mais de aquisição de conhecimento sobre o conteúdo apresentado na disciplina.

## 5. Avaliação

### 5.1. Dojo de FUP

No total, 154 alunos se inscreveram para participar do Dojo de FUP. Porém, no decorrer do semestre, a assiduidade dos participantes diminuiu e em média 30 alunos se mantiveram fiéis até o fim. A finalidade da pesquisa foi analisar as contribuições da metodologia de ensino Dojo de programação. Para isso, foi disponibilizado um formulário[11] com questões sobre satisfação, aprendizado, sugestões e problemas. O questionário contém 4 questões sobre dados demográficos, 7 utilizando uma escala numérica crescente tendo o valor 5 como máximo (adaptando a legenda da escala a cada questão), e 2 questões abertas, totalizando 9 questões. Apenas 26 alunos responderam ao formulário. A seguir, são discutidos os resultados das 4 questões mais relevantes para a pesquisa, ilustrados na Figura 2.

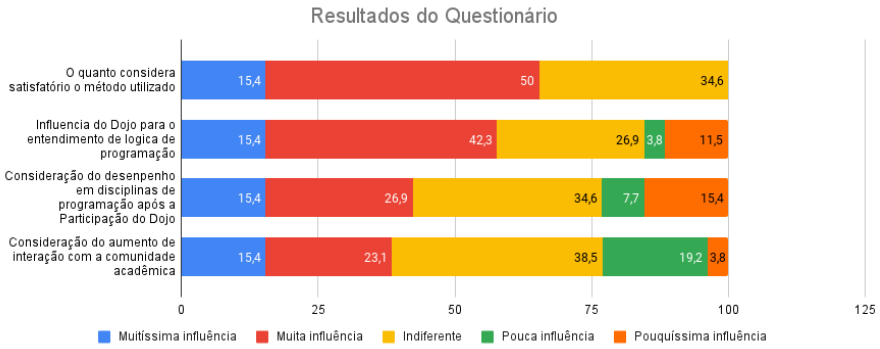


**Figura 2. Resultados do questionário**

Quando foi perguntado sobre a satisfação com o método utilizado, 50% dos participantes disseram que estavam muito satisfeitos e 15,4% disseram que estavam muitíssimo satisfeitos, o que quer dizer que a metodologia proporcionou um momento agradável a boa parte dos participantes. Em contrapartida, 34,6% responderam que sua satisfação é indiferente, e isso revela que algo precisa ser melhorado em relação ao ambiente, interação, relevância das questões e motivação dos alunos.

Já sobre a influência que o dojo teve no entendimento de lógica de programação, 15,4% e 42,3% avaliaram que houve, respectivamente, muitíssima e muita influência em seus conhecimentos de lógica de programação. Isso ocorre porque 88% dos estudantes que frequentam o dojo são do primeiro semestre, logo, têm pouca ou nenhuma experiência com programação. Outros 42,3% responderam que tiveram pouca, pouquíssima ou nenhuma influência em seus conhecimentos de lógica de programação, uma das hipóteses para isso acontecer, é a possibilidade destes participantes terem de um contato prévio com programação.

[11] https://bityli.com/avaliacao-fup

Na questão sobre melhoria do desempenho nas disciplinas de programação, 34,6% dos participantes avaliaram como indiferente, a alta porcentagem nessa resposta pode ter acontecido porque os bolsistas levaram sempre desafios sobre conteúdos já estudados em sala, portanto não teria como interferir no desempenho de atividades que já aconteceram. Os que perceberam muita ou muitíssima melhoria, somam 41.9%. A justificativa para esta melhoria irá depender da situação de cada participante, mas de forma geral, foi notado que a existência de uma dificuldade com os conteúdos, a participação assídua e a apresentação das dúvidas, pode ser um fator determinante para uma possível melhoria. Um fator extra que pode ter influenciado nesse resultado, foi o uso de técnicas de desenho com a biblioteca *Pygame* como metodologia de ensino de programação.

Por último, foi respondida a questão sobre o aumento da interação com a comunidade acadêmica, na qual 15,4% dos participantes disseram que o dojo influenciou muitíssimo, seguido por 23,1% que disseram que influenciou muito, 38,5% avaliaram como indiferente, 19,2% responderam que houve pouca influência e 3,8% apontaram que tiveram pouquíssima influência em suas interações com a comunidade acadêmica. Isso se deve pela diminuição de participantes ao longo do semestre, o que impossibilitou a interação dos alunos e a formação de duplas para resolver os desafios com mais alunos de cursos e semestres variados. Além disso, as respostas da questão aberta também contribuíram para a coleta de possíveis melhorias na metodologia de ensino. Sobre as dificuldades no Dojo, 40% dos alunos afirmaram que suas principais dificuldades estavam relacionadas a problemas particulares e ou técnicos, como a timidez e conexão de rede limitada, 60% relataram dificuldades com o nível de complexidade das questões.

## 5.2. Dojo de POO

Uma pesquisa semelhante foi realizada com a turma do Dojo de POO, que teve 51 inscritos e participação de em média 15 a 20 inscritos, onde 9 dos participantes responderam um formulário[12] de avaliação, que continha um total de 15 questões de múltiplas escolhas, e questões utilizando uma escala *Likert* de 4 e 5 alternativas. Na Figura 3 são discutidos os principais resultados obtidos da pesquisa.

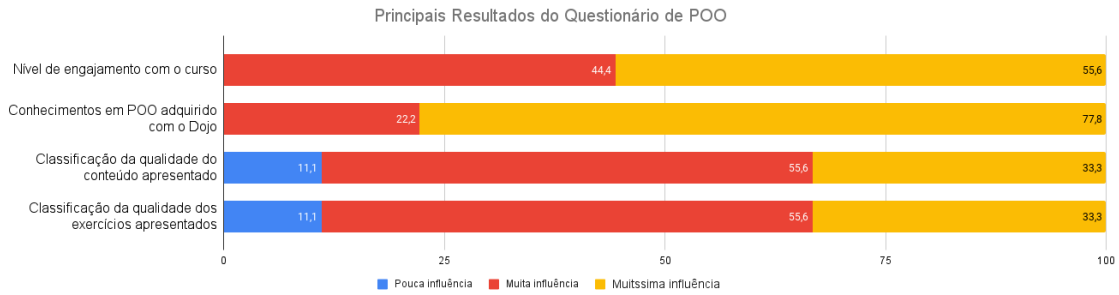


**Figura 3. Resultados do questionário**

Ao serem questionados sobre nível de engajamento com o curso, cerca de 55,6% dos entrevistados declararam que tiveram um altíssimo grau de participação, e cerca de 44,4% afirmaram que tiveram um bom nível de participação. Esta resposta é comprovada pelos bolsistas através do chat de mensagens do Discord a partir da interação entre os

[12]https://bityli.com/avaliacao-poo

próprios alunos sobre as temáticas. A princípio, o Discord seria apenas uma ferramenta de comunicação no horário dos encontros, porém o alto grau de engajamento dos alunos tornou os canais de texto do Discord em um fórum, o que beneficiou todos integrantes.

A fim de identificar o benefício da técnica de dojo para o ensino de POO, foi elaborada a seguinte questão: "Após sua participação no Dojo, como você avalia seu nível de conhecimento em POO?". Ao responderem o questionário, 77,8% afirmam que a participação dos encontros ajudou muito no entendimento do conteúdo da disciplina de POO. Os outros 22,2% alegam que a participação nos encontros ajudou de alguma forma no aprendizado.

Outra pergunta relevante do questionário, estava relacionada a classificação da metodologia de separação em canais de voz diferentes no Discord. Observou-se que 11% dos entrevistados opinaram de forma neutra, já os outros 33,3% e 55,6% dos entrevistados classificaram, respectivamente, como bom e muito bom. Este resultado mostra que a busca pela metodologia que enquadre da melhor forma as necessidades dos participantes pode garantir um melhor aproveitamento do conteúdo de acordo com as familiaridades com as categorias definidas na divisão.

Também foi perguntado como os alunos classificam como foi a interação com os bolsistas responsáveis pela atividade, 77,8% relatam que a interação com os bolsistas foi muito boa, já os 22,2% restantes declaram que a interação com os bolsistas foi péssima. Essa porcentagem dos entrevistados que alegam a falta de interação dos bolsistas pode ser explicada através do relato dos pontos positivos e negativos dos encontros na última questão do formulário. Um dos entrevistados relatou a seguinte justificativa - "O dojo era ótimo, eu só senti falta de interação entre os bolsistas e os participantes na hora da programação". Outro aluno relatou o seguinte - "No geral o DOJO foi uma experiência muito boa, principalmente para quem está começando a ver programação pois os conceitos de POO são essenciais para quem deseja ser um programador de alto nível. Alguns pontos que podem ser melhorados são melhorar a didática de apresentação do conteúdo do encontros e tentar motivar mais os alunos a interagir, pois em alguns momentos observei que os participantes estavam desmotivados ou com vergonha, talvez isso aconteça por conta do modelo remoto".

### 5.3. Discussões

Diante dos resultados apresentados, notou-se que as dificuldades estão associadas majoritariamente à forma de realização da atividade no formato remoto, mostrando-se necessário medidas que possam não apenas contribuir para o aprendizado, como também auxiliar os integrantes a lidar com a realidade em questão. Ao realizar o Dojo de forma remota, os responsáveis deverão agir de forma a promover uma interação maior entre os participantes auxiliando na criação de um ambiente de aprendizado descontraído e confortável.

Durante a aplicação da técnica Randori e a partir da avaliação apresentada, foi notado que apenas realizar o controle do tempo e tirar as dúvidas existentes da dupla que estiver desenvolvendo a solução naquele momento, poderia não ser a única metodologia de interação com os participantes. Então é necessário o desenvolvimento de ideias para incentivar a participação de todos, como: (i) não apenas ter o co-piloto e o piloto resolvendo como instigar comentários oriundos de outros participantes, (ii) promover a apresentação de conceitos sobre o que está sendo codificado, (iii) solicitar às duplas jus-

tificativas sobre o uso ou não de determinadas ferramentas, técnicas e ou estruturas de código, e (iv) instigar a criatividade dos participantes requisitando formas alternativas de resolver o problema apresentado, caso este esteja em completo entendimento por todos.

Há também a necessidade de incrementar a ajuda fornecida aos integrantes do Dojo, para isso, há a possibilidade da realização de convites aos monitores e professores das disciplinas de FUP e POO para contribuir no planejamento e definição de tópicos abordados nos encontros, assim como os desafios apresentados. No caso dos monitores, é interessante a colaboração próxima possibilitando a participação dos mesmos nos encontros promovendo um ambiente diversificado e incentivando os participantes na participação das monitorias realizadas semanalmente para cada disciplina. Além disso, conversar com os professores das disciplinas de programação a fim de bonificar os alunos com alto índice de assiduidade e participação durante os encontros do Dojo.

É interessante ressaltar que o incremento de ideias associadas ao método Dojo durante as aulas, podem não apenas incentivar a participação dos alunos nas aulas, como também diversificar a forma de aprendizado promovida pelo professor, tornando o aprendizado de programação mais agradável àqueles que têm mais dificuldades de compreensão do conteúdo. Somando-se às iniciativas como Dojo e monitorias, o incentivo à participação nestas atividades pode promover uma organização interna entre os alunos de forma a promover ambientes de grupos de estudo aplicando outros métodos de aprendizagem colaborativa.

Nota-se, a partir dos dados e conclusões apresentadas, que o Dojo de Programação é uma alternativa viável para promover o bom aprendizado dos conceitos básicos de programação, necessitando apenas de boas técnicas e planejamento adequados à realidade em que for aplicado.

## 6. Conclusão

Este trabalho apresentou um relato de experiência associado ao planejamento e realização de um Dojo de FUP e POO para estudantes de TI de forma remota, aplicando a técnica Randori. A metodologia contou com o apoio de várias ferramentas, onde se destaca o uso do *Replit*, da biblioteca *pygame* e da ferramenta de codificação *Visual Studio Code* que auxiliou no desenvolvimento das soluções.

Dentre as principais melhorias propostas pela prática do Dojo de programação nesse relato, destacam-se: (i) necessidade de realizar uma melhor adaptação na comunicação entre os responsáveis e os participantes que estiverem trabalhando na solução, (ii) analisar o desempenho dos alunos que frequentam a atividade do Dojo nas disciplinas iniciais de programação. Dentre os principais benefícios apresentados com o Dojo, tem-se: (i) melhoria do desempenho nas disciplinas de programação, (ii) incentivo a prática constante de codificação e (iii) melhoria na lógica de programação de FUP e POO.

Como trabalhos futuros, pretende-se: (i) realizar o Dojo de programação em outras turmas com melhorias e foco em outros paradigmas, e (ii) realizar uma análise de desempenho dos participantes do Dojo nas disciplinas iniciais de programação.

## Referências

Alves, G., Rebouças, A., and Scaico, P. (2019a). Coding dojo como prática de aprendizagem colaborativa para apoiar o ensino introdutório de programação: Um estudo de caso. In *Anais do XXVII Workshop sobre Educação em Computação*, pages 276–290.

Alves, G., Rebouças, A., and Scaico, P. (2019b). Coding dojo como prática de aprendizagem colaborativa para apoiar o ensino introdutório de programação: Um estudo de caso. In *Anais do XXVII Workshop sobre Educação em Computação*, pages 276–290, Porto Alegre, RS, Brasil. SBC.

Arimoto, M. and Oliveira, W. (2019). Dificuldades no processo de aprendizagem de programação de computadores: um survey com estudantes de cursos da área de computação. In *Anais do XXVII Workshop sobre Educação em Computação*, pages 244–254.

Calderon, I., Silva, W., and Feitosa, E. (2021). Um mapeamento sistemático da literatura sobre o uso de metodologias ativas durante o ensino de programação no brasil. In *Anais do XXXII Simpósio Brasileiro de Informática na Educação*, pages 1152–1161, Porto Alegre, RS, Brasil. SBC.

Carmo, D. H. and Braganholo, V. (2012). Um estudo sobre o uso didático de dojos de programaçao. In *XX Workshop sobre Educação em Computação*, pages 1–10.

Da Silva Garcia, F. W., Da Costa Carvalho, E., and Bezerra Oliveira, S. R. (2021). Use of active methodologies for the development of a teaching plan for the algorithms subject. In *2021 IEEE Frontiers in Education Conference (FIE)*, pages 1–9.

de Oliveira, C. M. C., Canedo, E. D., Faria, H., Amaral, L. H. V., and Bonifácio, R. (2018). Improving student's learning and cooperation skills using coding dojos (in the wild!). In *2018 IEEE Frontiers in Education Conference (FIE)*, pages 1–8.

INEP (2016). Sinopse estatística da educação superior 2015. Technical report, INEP.

INEP (2022). Sinopse estatística da educação superior 2020. Technical report, INEP.

Moreira de Castro, R. M. and Siqueira, S. (2019). Metodologias, técnicas, ambientes e tecnologias alternativas utilizadas no ensino de algoritmos e programaçao no ensino superior no brasil. In *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, volume 8, page 228.

Scherer, A. and Mór, F. (2020). Uso da técnica coding dojo em aulas de programação de computadores. In *Anais do XXVIII Workshop sobre Educação em Computação*, pages 6–10, Porto Alegre, RS, Brasil. SBC.

Silva, G., Santos, G., and Rissoli, V. (2020). A importância da linguagem nativa para a aprendizagem significativa em lógica de programação. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação*, pages 1803–1812, Porto Alegre, RS, Brasil. SBC.

Souza, D. M., da Silva Batista, M. H., and Barbosa, E. F. (2016). Problemas e dificuldades no ensino de programação: Um mapeamento sistemático. *Revista Brasileira de Informática na Educação*, 24(1):39.