

Ensino de programação e desenvolvimento do Pensamento Computacional por meio da construção de aplicativos no App Inventor

Rodrigo Thomas¹, Adão Caron Cambraia¹

¹ Programa de Pós-graduação em Educação Profissional e Tecnológica (ProfEPT)
Instituto Federal Farroupilha (IFFar) – Campus Jaguari/RS

{rodrigo.thomas, adao.cambraia}@iffar.edu.br

Abstract. *The article discusses how the construction of mobile applications through block-based programming in App Inventor provides the development of computational thinking and contributes to programming learning. For this, we propose a didactic sequence based on the Three Pedagogical Moments with programming exercises in blocks. The methodology is qualitative, taking the form of a case study with groups of students entering higher education courses in Computing. The results point to greater student motivation for learning programming and evidence of the development of computational thinking.*

Resumo. *O artigo aborda como a construção de aplicativos móveis por meio da programação baseada em blocos no App Inventor proporciona o desenvolvimento do pensamento computacional e contribui com a aprendizagem de programação. Para isso, propomos uma sequência didática pautada nos Três Momentos Pedagógicos com exercícios de programação em blocos. A metodologia é qualitativa, assumindo o formato de estudo de caso com grupos de alunos ingressantes de cursos superiores de Licenciatura em Computação. Os resultados apontam uma maior motivação dos alunos para a aprendizagem de programação e indícios do desenvolvimento do pensamento computacional.*

1. Introdução

A escola tem um papel fundamental na formação de cidadãos que participem das decisões relativas às produções tecnológicas na sociedade [Bazzo 2011]. Com isso, é fundamental a introdução de habilidades da área da Ciência da Computação nos diferentes níveis de ensino. O pensamento computacional (PC) é uma habilidade que contribui na formulação, compreensão e resolução de problemas da vida cotidiana. Trata-se, portanto, de uma aptidão importante para todas as pessoas [Wing 2006]. A programação de computadores é uma das formas para o desenvolvimento do PC. No entanto, a aprendizagem de programação é um desafio [Silva e Moreira 2021]. Diante disso, o problema de pesquisa é: a construção de aplicativos móveis por meio da programação baseada em blocos no App Inventor proporciona o desenvolvimento do pensamento computacional e contribui com a motivação e a aprendizagem de programação?

O texto irá analisar se ocorre a aprendizagem da programação e o desenvolvimento do PC por meio de atividades produzidas numa sequência didática

pautada nos Três Momentos Pedagógicos (3MP) [Delizoicov, Angotti e Pernambuco 2011]. Com uma dinâmica baseada na concepção freireana (dialógica e problematizadora) onde os temas a serem trabalhados surgem da realidade dos educandos, os 3MP se apresentam como uma ferramenta pedagógica para a compreensão de conceitos científicos de forma contextualizada.

O artigo está estruturado em três tópicos. O primeiro apresenta a metodologia da pesquisa. O segundo, uma narrativa analítica do desenvolvimento e avaliação da aprendizagem de programação na sequência didática. No terceiro tópico, analisa-se a motivação dos alunos em utilizar a programação em blocos com App Inventor.

2. Metodologia

A presente pesquisa tem caráter exploratório, assumindo o formato de um estudo de caso. Segundo Gil [2016, p. 37], o estudo de caso “consiste no estudo profundo e exaustivo de um ou poucos objetos, de maneira que permita seu amplo e detalhado conhecimento”. O estudo foi desenvolvido com uma turma de Licenciatura em Computação do Instituto Federal de Educação, Ciência e Tecnologia Farroupilha – Campus Santo Augusto, de junho à julho de 2021, em formato remoto devido a pandemia da Covid-19. Para o desenvolvimento das atividades foram utilizadas as aulas cedidas pelo professor na disciplina de Algoritmos. A turma era composta por 19 alunos com idades variando entre 18 a 59 anos e, também, compôs como participante da pesquisa o professor que participou dos encontros. A maioria da turma era constituída por trabalhadores durante o dia e estudantes da licenciatura no período noturno.

Para responder à pergunta da pesquisa, foi organizada uma sequência didática pautada nos Três Momentos Pedagógicos para desenvolver atividades de programação em blocos usando o App Inventor. Os 3MP definiram a metodologia de trabalho de sala de aula (estruturação das aulas) enquanto ferramenta didático-pedagógica, articulando os momentos (problematização inicial, organização do conhecimento e aplicação do conhecimento) com o intuito de ensinar um determinado conteúdo com a perspectiva de diálogo entre os participantes. Assim, foram planejados 5 (cinco) encontros da seguinte forma:

Quadro 1 – Planejamento dos encontros da sequência didática.

Encontro	1º MP	2º MP	3º MP
1 (21/06)	Introdução dos conceitos iniciais do PC e fomentar a discussão a respeito da sua aplicação nas atividades rotineiras. Apresentação do App Inventor.	Apresentação dos quatro pilares do pensamento computacional, de forma teórica e prática, demonstrando a sua aplicabilidade em situações reais.	Discussão sobre a utilidade do PC, agora procurando empregar os conceitos dos pilares. Desafio de desenvolver um aplicativo que replica a interface de <i>login</i> do sistema acadêmico.
2 (28/06)	Discussão sobre como se dá o processo de resolução de um problema computacional ou não. Quais os passos desde o entendimento de um problema até a construção da solução.	Conceituadas as partes de um algoritmo, variáveis e operadores, e como ocorre a escrita de um algoritmo no App Inventor com o desenvolvimento de um aplicativo para cálculo de média de duas notas.	Discussão sobre o processo de resolução de problemas computacionais empregando os conceitos apresentados na aula. Desafio de incrementar um aplicativo para calcular a média de quatro notas.

3 (12/07)	Compreender a utilização de uma estrutura de seleção em situações reais do dia a dia. Analise do aplicativo de média para identificar como uma estrutura de seleção pode ser utilizada para classificar o resultado em "aprovado" ou "reprovado".	Apresentação do conceito de estrutura de seleção segundo a bibliografia presente na ementa da disciplina. Também de uma estrutura de seleção aninhada, demonstrando seu uso no App Inventor.	Nova discussão sobre problemas que podem ser solucionados a partir do desenvolvimento de aplicativos com essa funcionalidade. Desafio de desenvolver um aplicativo para o cálculo do índice de massa corporal que apresente a faixa de desnutrição ou obesidade.
4 (19/07)	Problematização sobre como ocorre a utilização de uma estrutura de repetição e como controlar o número de repetições evitando a execução infinita.	Conceituação e desenvolvimento de exemplos práticos no App Inventor com as estruturas de repetição “para” e “enquanto”. Definição e princípios que envolvem o uso de contadores.	Análise do uso de estruturas de repetição para solução de problemas reais, apontando qual tipo de estrutura é mais adequada para cada situação. Desafio de desenvolver um aplicativo para o cálculo do fatorial de um número.
5 (26/07)	Percepções e mudanças provocadas a partir da aplicação diária dos conhecimentos das aulas anteriores. Quais sensores do <i>smartphone</i> os alunos utilizam e como podem incorporá-los no desenvolvimento de apps. Como automatizar tarefas reaproveitando partes de um algoritmo.	Demonstração conceitual e prática do uso do sensor acelerômetro no desenvolvimento de um aplicativo, bem como realizar o controle de foco em componentes de interface. Apresentação do conceito de procedimento ou função e como utilizá-lo no App Inventor.	Análise dos aplicativos desenvolvidos nas aulas anteriores procurando identificar blocos repetitivos que poderiam ser agrupados em uma função. Desafio de resolução de diferentes problemas matemáticos através do desenvolvimento de um aplicativo com base nos conhecimentos adquiridos no decorrer das aulas.

Para potencializar o compartilhamento e a divulgação da sequência didática, as aulas foram disponibilizadas em um *website*, com o endereço eletrônico: “<https://sites.google.com/view/pceappinventor>”. A organização teve por base os resultados da aplicação de um questionário de percepção inicial e todos os encontros foram realizados e gravados pela ferramenta Google Meet. As gravações foram utilizadas para a elaboração do relato dos encontros e a análise da motivação dos alunos no terceiro tópico do texto. Os excertos selecionados a partir de um olhar mais cuidadoso das gravações (que estão disponíveis em: “<https://bit.ly/PCeAppInventor>”) foram transcritos destacando o momento da gravação em hora, minutos e segundos, possibilitando aos leitores o acesso em vídeo.

Para analisar a aprendizagem de programação e o desenvolvimento do PC, foi escolhida a ferramenta web CodeMaster¹ como instrumento de avaliação dos artefatos de código (aplicativos) criados pelos alunos. Assim, durante 5 encontros, os participantes foram desafiados a construir aplicativos e encaminhar o código-fonte ao

¹ Mais informações em: <https://computacaonaescola.ufsc.br/codemaster>.

pesquisador para posterior análise no CodeMaster com o propósito de se obter uma avaliação da aprendizagem.

3. Desenvolvimento e avaliação da sequência didática pautada nos 3MP

Para promover o desenvolvimento do pensamento computacional, além dos conteúdos teóricos e das atividades práticas realizadas durante as aulas, ao final de cada encontro os alunos foram desafiados a construir aplicativos no App Inventor enquanto ação pertencente ao movimento de aplicação do conhecimento (terceiro momento pedagógico). Assim, ao final da sequência didática, foram desenvolvidos e encaminhados ao pesquisador 71 aplicativos. Vale ressaltar que esse quantitativo é de aplicativos que foram construídos para resolver os desafios extraclasse, sendo que em aula os estudantes também desenvolveram aplicativos de teste para aprendizagem.

No encontro 1, os alunos foram desafiados a criar um aplicativo que replica a interface de *login* do sistema acadêmico SIGAA² para que pudessem se familiarizar com o designer e componentes básicos do App Inventor. Dos 17 alunos presentes na aula, 13 encaminharam o aplicativo desenvolvido (arquivo no formato “.aia”). Na Tabela 1 é possível observar a nota em programação (de 0 a 10) que cada aplicativo recebeu e o nível em “faixa ninja”³, tendo como mais frequente a “faixa vermelha” que representa o nível 4 de progresso (nota final entre 3 e 4). Para a identificação dos indivíduos, quando necessário, tanto na caracterização a seguir, quanto na apresentação dos resultados, foram utilizadas as descrições como seguem: Aluno 1; Aluno 2; Aluno 3; e assim sucessivamente, sem especificar o seu sexo e levando em consideração a identificação atribuída no questionário inicial.

Tabela 1 – Avaliação dos aplicativos desenvolvidos no desafio da aula 1.

Aplicativo	Nota em Programação	Nota em Interface de Usuário	Nota Final	Nível
Aluno1.aia	1.43	7.5	3.25	faixa vermelha
Aluno4.aia	2.14	7.81	3.84	faixa vermelha
Aluno5.aia	2.5	8.44	4.28	faixa roxa
Aluno8.aia	0.36	5.59	1.93	faixa amarela
Aluno10.aia	4.29	6.84	5.05	faixa azul
Aluno11.aia	2.14	7.06	3.62	faixa vermelha
Aluno12.aia	2.86	6.47	3.94	faixa vermelha
Aluno13.aia	1.07	7.81	3.09	faixa vermelha
Aluno14.aia	2.14	7.81	3.84	faixa vermelha
Aluno15.aia	2.14	7.5	3.75	faixa vermelha
Aluno16.aia	1.43	9.44	3.83	faixa vermelha
Aluno17.aia	2.5	5.88	3.51	faixa vermelha
Aluno18.aia	2.5	7.22	3.92	faixa vermelha
Média	2.12	7.34	3.68	

² Sistema Integrado de Gestão de Atividades Acadêmicas utilizado pela instituição de ensino.

³ Graduação progressiva composta pela sequência de 10 cores: branca, amarela, laranja, vermelha, roxa, azul, turquesa, verde, marrom e preta. A classificação considera a frequência de elementos de código, inferindo a aprendizagem das competências do PC por meio de indicadores previamente definidos.

Cabe observar também que a nota final leva em consideração, além da programação, os componentes de interface e sua personalização, a exemplo: formato de botões, tamanho de fontes, comprimento do texto dos componentes, harmonia de cores, entre outros. Apesar do aspecto visual da interface não ser o foco das práticas realizadas, a nota em interface de usuário demonstra que os alunos conseguiram compreender e utilizar os componentes básicos do App Inventor com certa facilidade, até por se tratar do primeiro contato que tiveram com a linguagem.

O desafio da aula 2 consistia no desenvolvimento de um aplicativo para o cálculo da média de quatro números. Para tal, os alunos deveriam utilizar variáveis e operadores aritméticos estudados, além de componentes de interface para a entrada dos dados e a apresentação do resultado do cálculo. Estiveram presentes 15 alunos e todos enviaram o aplicativo desenvolvido, com as notas listadas na Tabela 2.

Tabela 2 – Avaliação dos aplicativos desenvolvidos no desafio da aula 2.

Aplicativo	Programação	Interface	Nota Final	Nível
Aluno1.aia	3.21	6.67	4.25	faixa roxa
Aluno3.aia	4.64	8.24	5.72	faixa azul
Aluno4.aia	3.21	8.24	4.72	faixa roxa
Aluno5.aia	3.21	8.24	4.72	faixa roxa
Aluno8.aia	2.86	8.82	4.65	faixa roxa
Aluno10.aia	4.29	7.65	5.30	faixa azul
Aluno11.aia	3.21	9.41	5.07	faixa azul
Aluno12.aia	4.64	7.78	5.58	faixa azul
Aluno13.aia	3.57	7.94	4.88	faixa roxa
Aluno14.aia	2.14	7.81	3.84	faixa vermelha
Aluno15.aia	2.5	8.24	4.22	faixa roxa
Aluno16.aia	3.57	8.24	4.97	faixa roxa
Aluno17.aia	3.21	7.65	4.54	faixa roxa
Aluno18.aia	3.21	7.06	4.37	faixa roxa
Aluno19.aia	3.21	6.47	4.19	faixa roxa
Média	3.38	7.90	4.73	

Ao comparar os resultados da avaliação das duas primeiras aulas, é possível observar uma evolução no desempenho dos alunos, principalmente ao analisar individualmente os códigos-fonte e observar os operadores e variáveis utilizados pelos alunos separadamente.

Após terem estudado a respeito de estruturas de seleção na aula 3, os discentes foram desafiados a criar um aplicativo de cálculo do índice de massa corporal, classificando a faixa de nutrição em 6 níveis, de desnutrição à obesidade grau três. Participaram da aula 16 alunos, mesmo número de aplicativos entregues ao pesquisador e professor da disciplina, conforme resultados exibidos na Tabela 3. Mais uma vez, foi possível observar a evolução dos alunos nas notas de programação e interface, alguns atingindo a “faixa turquesa” que indica o sétimo nível de classificação.

Tabela 3 – Avaliação dos aplicativos desenvolvidos no desafio da aula 3.

Aplicativo	Programação	Interface	Nota Final	Nível
Aluno1.aia	3.21	7.78	4.58	faixa roxa

Aluno3.aia	3.57	6.76	4.53	faixa roxa
Aluno4.aia	5.71	7.35	6.20	faixa turquesa
Aluno5.aia	5.71	7.35	6.20	faixa turquesa
Aluno6.aia	4.64	8.06	5.67	faixa azul
Aluno7.aia	2.5	6.76	3.78	faixa vermelha
Aluno8.aia	5.36	7.94	6.13	faixa turquesa
Aluno9.aia	5.0	7.35	5.71	faixa azul
Aluno10.aia	5.0	7.63	5.79	faixa azul
Aluno11.aia	4.64	8.53	5.81	faixa azul
Aluno12.aia	5.71	6.39	5.91	faixa azul
Aluno15.aia	5.0	8.06	5.92	faixa azul
Aluno16.aia	4.64	7.65	5.54	faixa azul
Aluno17.aia	5.0	10.0	6.50	faixa turquesa
Aluno18.aia	3.21	7.78	4.58	faixa roxa
Aluno19.aia	2.86	7.35	4.21	faixa roxa
Média	4.48	7.67	5.44	

O desafio da aula 4, por sua vez, compreendia o desenvolvimento de um aplicativo para o cálculo do fatorial de um número fazendo uso de uma estrutura de repetição. Durante a aula, foram apresentadas duas estruturas distintas, ficando a critério do aluno optar por uma ou outra. Também foi ressaltada a premissa de que só é possível calcular o fatorial de números positivos, mas sem especificar que seria necessária uma estrutura condicional para a testagem do valor informado pelo usuário. Estiveram presentes 16 alunos e 11 deles enviaram o aplicativo desenvolvido. Importante observar que mais da metade dos alunos implementaram uma estrutura condicional para a testagem do número digitado pelo usuário visando garantir a realização do cálculo apenas com números positivos, o que indica a compreensão do conceito e das situações no qual pode ser aplicado.

Na última aula, foi proposto aos alunos que desenvolvessem um aplicativo que contemplasse alguns cálculos matemáticos, dentre eles: calculadora de juros compostos, listagem de números primos, sequência de Fibonacci, cálculo da tabuada de um número e listagem de números pitagóricos. Solicitou-se a implementação de ao menos um desses cálculos fazendo uso dos conhecimentos adquiridos durante as aulas. Dos 14 alunos presentes na aula, 11 encaminharam o aplicativo desenvolvido. Na Tabela 4 é possível observar as notas atribuídas a cada aplicativo.

Tabela 4 – Avaliação dos aplicativos desenvolvidos no desafio da aula 5.

Aplicativo	Programação	Interface	Nota Final	Nível
Aluno1.aia	5.36	2.75	4.58	faixa roxa
Aluno3.aia	6.79	7.22	6.92	faixa turquesa
Aluno6.aia	4.64	7.5	5.50	faixa azul
Aluno7.aia	3.21	6.39	4.16	faixa roxa
Aluno8.aia	5.36	7.65	6.05	faixa turquesa
Aluno9.aia	6.07	5.56	5.92	faixa azul
Aluno10.aia	5.71	7.0	6.10	faixa turquesa
Aluno16.aia	5.36	5.0	5.25	faixa azul
Aluno17.aia	6.07	5.59	5.93	faixa azul

Aluno18.aia	3.93	5.28	4.33	faixa roxa
Aluno19.aia	4.29	7.94	5.38	faixa azul
Média	5.16	6.17	5.47	

Para além das notas atribuídas aos aplicativos desenvolvidos no desafio final, foi possível observar que os alunos se empenharam na construção de aplicativos realmente funcionais. Assim, nesta tarefa foram desenvolvidos 5 aplicativos para o cálculo de juros compostos, 2 de tabuada, 2 de números primos, 1 da sequência Fibonacci e 1 aplicativo que contemplou três cálculos distintos que não haviam sido sugeridos no enunciado da tarefa: teorema de Pitágoras, conversor de metros em centímetros e de horas em minutos. Apesar da aula 5 ter priorizado a revisão dos conteúdos e o esclarecimento de dúvidas dos discentes, restando pouco tempo para a introdução de novos conceitos, ainda assim, quatro projetos implementaram mais de uma tela e um o uso do sensor acelerômetro, funcionalidades abordadas no último encontro.

Outra questão interessante que pôde ser verificada diz respeito ao cuidado com o aspecto visual dos aplicativos desenvolvidos. A maior parte apresenta uma interface agradável, inclusive implementando componentes não abordados nas aulas, como a inclusão de imagens e funções multimídia, conforme é possível observar na Figura 1.

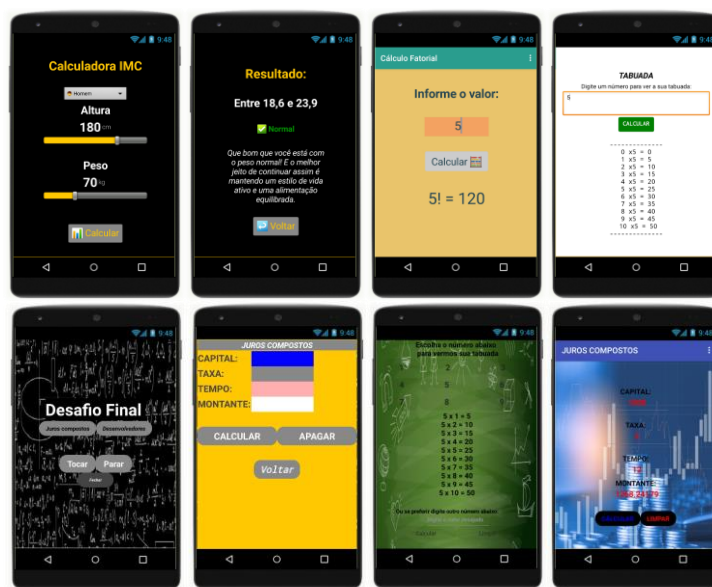


Figura 1. Aplicativos desenvolvidos pelos alunos no App Inventor.

Assim, após a avaliação dos aplicativos no sistema web CodeMaster, além dos conceitos de algoritmos e programação, entendemos que as habilidades adquiridas relativas ao PC também foram validadas, uma vez que os alunos demonstraram autonomia e capacidade para resolver os desafios propostos aplicando os conhecimentos construídos nas cinco aulas da sequência didática. Chegamos a essa conclusão em virtude do método adotado pela ferramenta, como explicado anteriormente no texto. Também por concordar com Wing [2006] que se refere ao PC como sendo constituído por processos de pensamento envolvidos na criação de soluções algorítmicas. Apesar de difíceis de serem medidos pela sua natureza abstrata, tais processos podem ser representados através de indicadores relacionados aos objetivos de aprendizagem,

assumindo que atributos mensuráveis podem ser extraídos do artefato de software desenvolvido pelo aluno, o que se constitui na avaliação realizada pelo CodeMaster.

Nesse sentido, Romero, Lepage e Lille [2017] afirmam que a essência do PC é a capacidade de transpor significado abstrato em significado concreto, coisificando um modelo abstrato em um conjunto de instruções concretas como um algoritmo ou programa de computador. Os autores destacam ainda a importância do envolvimento criativo do aluno nas atividades de programação, classificando-o em 5 níveis: 1) exposição passiva à explicações do professor; 2) atividades de programação passo-a-passo; 3) atividades de programação com potencial criativo; 4) programação em equipe e 5) cocriação participativa de conhecimento. Entendemos que durante a sequência didática foram atingidos quatro desses cinco níveis a partir das atividades de programação propostas, sendo que o último nível só poderia ser alcançado a partir da interação entre diferentes equipes, preferencialmente com características distintas, o que demandaria mais tempo e seria melhor realizado num contexto presencial.

4. Motivação dos alunos na aprendizagem de programação com App Inventor

Mesmo com o relato dos encontros e a avaliação dos aplicativos desenvolvidos, alguns resultados da realização da sequência didática só podem ser descritos a partir dos momentos de interação entre pesquisador-participantes e entre os próprios sujeitos durante os encontros. Ao total, foram 15 horas de aula *online* através do Google Meet, todas gravadas e disponibilizadas aos alunos como fonte de consulta para esclarecimento de dúvidas, revisão de conceitos ou repetição de práticas que não puderam ser acompanhadas ao vivo. Nesse tópico são apresentados alguns trechos de manifestações dos educandos e demais percepções obtidas durante os encontros.

Foi quase consenso entre os participantes da pesquisa o interesse em aprender os conteúdos através de uma abordagem diferente. Acredita-se que essa motivação esteja relacionada com as possíveis dificuldades que estavam enfrentando para a aprendizagem de programação através da linguagem textual do Portugol Studio, ferramenta utilizada na disciplina de Algoritmos. Entre os relatos que levam a esta conclusão, está um excerto selecionado da aula 2, quando o Aluno 8, às 1h31m02s, afirma: “Portugol eu não estou entendendo muito, mas o App Inventor estou mais por dentro!”, se referindo a compreensão que estava obtendo com relação ao uso de estruturas de seleção no App Inventor. O docente titular da disciplina, a partir do relato do aluno de que estava entendendo os conceitos aplicados no App Inventor, se manifestou afirmando:

"legal que vocês tenham essa percepção de que usando uma outra linguagem, uma outra plataforma, vocês compreendem alguns aspectos, né, de forma melhor.. porque na verdade existem vários tipos de linguagens de programação e tem aquelas que são só textuais" (Docente titular da disciplina de Algoritmos, 1h32m20s).

O professor destaca a motivação dos alunos em participar das práticas da sequência didática e ressalta a importância em compreender diferentes formas de programar. Essa evidencia também é apresentada em um estudo que comparou o ensino de programação com App Inventor e Python, concluindo que a programação com App Inventor motivou os alunos e proporcionou um maior índice de aprovação [Oliveira, Pereira e Teixeira 2021].

Na aula 3, um aluno espontaneamente se pronunciou afirmando: “muito mais legal que Portugal... deu certo o aplicativo aqui óh, adorei!” (Aluno 10, 38m52s) ao exibir na câmera do *notebook* o aplicativo de cálculo da média sendo executado em seu *smartphone*. Isso demonstra como os alunos se motivam ao programar e visualizar suas produções, que são aplicações reais visualizadas no celular ou simulador. Além da motivação para a aprendizagem de programação, se evidencia a construção da identidade docente, conforme relato no final da quinta aula:

"Eu realmente consegui aprender e é muito importante pra gente que está dentro da área pedagógica porque não adianta ter um conhecimento de algoritmos e não ter um conhecimento de como ensinar isso pra quem não sabe algoritmos. Então quem sabe essa questão dos blocos possa vir a ser importante para a aprendizagem." (Aluno 3, 2h41m43s).

O aluno 3 mostra maturidade, se reconhece como um futuro professor e identifica diferentes metodologias para a aprendizagem de programação como algo fundamental para que possa qualificar o seu trabalho docente depois de formado. São evidências da assimilação de que o conhecimento de professor é mais do que somente o conhecimento disciplinar [Cabraia, Araújo e Biondo 2023].

Ao final da aula 3, ao serem questionados sobre as suas percepções sobre os encontros, os alunos demonstraram motivação com relação as atividades desenvolvidas:

"Por mais que está sendo bem atípico esse semestre, eu acho que a gente está conseguindo, dentro do pouco tempo [...] eu achei muito interessante a nossa aula de algoritmos ser assim, [...] estou achando muito proveitoso essa maneira como está sendo aplicado." (Aluno 6, 1h37m04s).

Essa questão é complementada pelo professor da disciplina de algoritmos, que afirmou no início da Aula 4:

"Estou feliz com o que estamos produzindo[...] os alunos estão se empenhando com as oficinas. Estou notando uma motivação da turma em fazer os exercícios e também uma persistência que inclusive eu não observava em turmas anteriores".

Esses relatos corroboram o entendimento de que o App Inventor, ao utilizar uma linguagem visual baseada em blocos, se mostra uma alternativa para o ensino de programação. Não necessariamente em substituição a uma linguagem textual, mas como complementar a ela, possibilitando a aprendizagem por meio de distintas abordagens.

Também foram observadas algumas dificuldades básicas de utilização do computador (falta de letramento digital) que dificultou a compreensão dos passos a serem executados para a construção dos aplicativos. A exemplo do Aluno 8 que relatou na aula 1, à 1h52m42s, “tenho dificuldade com a informática, não a tarefa em si” e também na aula 2, aos 13m12s, “até tentei fazer, mas tive bastante dificuldade pra baixar aquela parte do aplicativo, eu baixei o programa só que pedia pra salvar no ‘.aia’ e ele salva num diretório só que depois eu não conseguia encontrar no computador”, referindo-se à tarefa de exportação do código-fonte do aplicativo desenvolvido para a postagem no sistema acadêmico. Isso pode ser solucionado com uma ajuda mais próxima ao aluno, o que foi dificultado em função da pandemia. Essa evidência é um indício da importância do desenvolvimento de atividades de letramento digital antes e durante as aulas [Machado, Amorin e Barros 2021]. Por outro lado, o Aluno 17, diante das dúvidas, pesquisou vídeos e conteúdo extra sobre App Inventor e desenvolveu um letramento digital, conforme a sua fala, aos 15m38s:

"Tive algumas dúvidas de como selecionar os blocos, aí eu complementei com alguns vídeos no Youtube. Até fiz uma atividade bem fácil que nem o Pesquisador (falou o nome) passou da soma e divisão[...].”

Outras duas falas na segunda aula retratam dificuldades para o desenvolvimento dos desafios extraclasse. O Aluno 6 afirma, aos 12m03s, após ser questionado pelo professor sobre as dúvidas para a resolução do desafio, “a gente trabalha o dia inteiro e não tive tempo de rever”, o que demonstra a realidade da maioria dos discentes que trabalham durante o dia e estudam à noite, sobrando pouco tempo para as atividades extraclasse.

Além das manifestações citadas, os alunos também participaram das aulas sugerindo funcionalidades que gostariam de incorporar aos aplicativos (ex: controle de foco e arredondamento de decimais) e solicitando revisão dos conteúdos que estavam com maior dificuldade. Assim, as aulas foram se adequando ao ritmo de aprendizagem dos alunos. Diante dessas dificuldades, entende-se que a aprendizagem desenvolvida pelos alunos, tanto dos conceitos de programação quanto das habilidades do pensamento computacional, foram extremamente relevantes e superaram o que se esperava diante do cenário imposto.

5. Considerações finais

Os recursos do App Inventor se mostraram válidos para a aprendizagem de programação e o processo de promoção do pensamento computacional. A ferramenta facilitou a expressão de soluções computacionais a partir dos conceitos de programação trabalhados durante a sequência didática, gerando como resultado produtos (aplicativos) tangíveis que apresentam o conhecimento apropriado e produzido pelo aluno. Sua utilização nos cursos de Licenciatura em Computação, com um público heterogêneo, com idades variando de 18 a 59 anos, demonstra que a ferramenta também pode ser utilizada em outros níveis de ensino, seguindo a aplicação dos conceitos abordados na sequência didática.

Foi possível perceber ainda que apesar dos alunos interagirem diariamente com dispositivos móveis, em geral, não apresentavam conhecimentos prévios de Computação que pudessem permitir uma facilidade maior na aprendizagem, mesmo assim foram capazes de criar vários aplicativos para resolver os desafios propostos. Destaca-se, também, que as práticas da sequência didática possibilitaram aos participantes a assimilação do conteúdo e a sua aplicação nos projetos desenvolvidos.

Nesse sentido, os encontros contribuíram para a formação de futuros docentes que poderão atuar com o PC, na perspectiva de não apenas aplicar ferramentas de forma instrumental, mas incorporar o construcionismo defendido por Papert [1980] e promover uma educação aproximada do presente em que se encontram os educandos.

As análises das respostas, a avaliação dos aplicativos no CodeMaster, bem como toda a participação e engajamento dos alunos durante as aulas, nos leva ao entendimento de que o objetivo geral de desenvolver o pensamento computacional e a aprendizagem de programação a partir da construção de aplicativos móveis no App Inventor foi atingido. Em vista disso, é plausível dizer que os Três Momentos Pedagógicos, utilizados como ferramenta didático-pedagógica para a estruturação da sequência didática se mostraram eficientes e resolutivos à medida que apresentaram um caminho possível para o ensino contextualizado e significativo.

Referências

- Cambraia, A. C.; Araújo, M. C. P. de; Biondo, U. L. R. (2022) Conhecimento Didático do Conteúdo na Formação de Professores de Computação. *Revista Brasileira de Informática na Educação*, [S. l.], v. 30, p. 449–470. DOI: 10.5753/rbie.2022.2525. Disponível em: <https://sol.sbc.org.br/journals/index.php/rbie/article/view/2525>.
- Delizoicov, D.; Angotti, J.; Pernambuco, M. (2011) *Ensino de Ciências: fundamentos e métodos*. 4. ed. São Paulo: Cortez.
- Gil, A. C. (2016) *Como elaborar projetos de pesquisa*. 5. ed. São Paulo: Atlas.
- Machado, V. R.; Amorim, T. B.; Barros, P. M. (2021) Interdisciplinaridade no ensino de física e computação na educação básica: relato de experiência de um curso de formação inicial e continuada sob a perspectiva na construção de experimentos. In: *Simpósio Brasileiro de Educação em Computação (EDUCOMP)*, 1., On-line. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2021. p. 246-254.
- Oliveira, S.; Pereira, M. A.; Teixeira, F. A. (2021) MIT App Inventor como Ambiente de Ensino de Algoritmos e Programação. In: *Workshop Sobre Educação em Computação (WEI)*, 29., Evento Online. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2021. p. 61-70.
- Papert, S. (1980) *Mindstorms: children, computers, and powerful ideas*. New York: Basic Books.
- Romero, M.; Lepage, A.; Lille, B. (2017) Computational thinking development through creative programming in higher education. *Int J Educ Technol High Educ* 14, 42. Disponível em: <https://doi.org/10.1186/s41239-017-0080-z>.
- Silva, F. L.; Moreira, I. A. T. (2021) Análise das dificuldades na aprendizagem de programação no curso de análise e desenvolvimento de sistemas do IFRN/Pau dos Ferros. In: *Encontro Unificado de Computação do Piauí (ENUCOMPI)*, 14., Picos. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2021 . p. 41-48.
- Wing, J. (2006) Computational Thinking. *Communications of the ACM*. New York, v. 49, nº 3, p. 33-35.