

GUIA DIDÁTICO CEL-LOGIC: PENSAMENTO COMPUTACIONAL E LÓGICA DE PROGRAMAÇÃO NO ENSINO MÉDIO

*Carolina Leite Cardinale**

Centro Paula Souza, Faculdade de Tecnologia de São Paulo

<https://orcid.org/0000-0002-0532-0025>

*Thaís Cristina Rodrigues Tezani***

Universidade Estadual Paulista (UNESP) - Campus Bauru

<https://orcid.org/0000-0002-0601-1865>

RESUMO

O artigo apresenta as bases para a construção do Guia Didático Interativo CEL-LOGIC, para o ensino de Lógica de Programação e Pensamento Computacional (PC), no Ensino Médio Integrado ao Técnico na área de Informática, enquanto Produto Educacional de uma pesquisa de mestrado profissional. Por meio da plataforma App Inventor, o CEL-LOGIC foi projetado para auxiliar no ensino e aprendizagem dos iniciantes em programação e na introdução do PC, com base nas diretrizes pedagógicas e tecnológicas para o desenvolvimento de aplicativos móveis dos documentos curriculares oficiais das escolas técnicas de São Paulo. Seguiu os seguintes passos metodológicos: estudos teóricos, coleta de dados via formulário de avaliação diagnóstica, roteiro de aplicação, menções obtidas pelos estudantes durante os bimestres e formulário de avaliação do uso. Como resultados obtidos destaca-se a aquisição de novos conhecimentos, por meio da Programação de Aplicativos Móveis e a construção de aprendizagens a partir do PC.

Palavras-chave: pensamento computacional; ensino de lógica de programação; MIT App Inventor; ensino médio.

ABSTRACT

CEL-LOGIC DIDACTIC GUIDE: COMPUTATIONAL THINKING AND PROGRAMMING LOGIC IN HIGH SCHOOL

The article presents the bases for the construction of the CEL-LOGIC Interactive Didactic Guide, for the teaching of Programming Logic and Computational Thinking (PC), in the High School Integrated with the Technician in the area

* Mestre em Docência para a Educação Básica pela Faculdade de Ciências, Universidade Estadual Paulista (UNESP-Bauru). Professora do Centro Paula Souza, Faculdade de Tecnologia de São Paulo. Bauru, São Paulo, Brasil. E-mail: carolina.cardinale@unesp.br

** Doutora em Educação pela Universidade Federal de São Carlos (UFSCar). Professora Associada do Departamento de Educação da Faculdade de Ciências da Universidade Estadual Paulista (UNESP-Bauru). Líder do Grupo de Estudos e Pesquisa sobre Tecnologias, Educação e Currículo (GEPTEC). Docente Permanente do Programa de Pós-Graduação em Docência para a Educação Básica da Faculdade de Ciências, UNESP-Bauru. Bauru, São Paulo, Brasil. E-mail: thais.tezani@unesp.br

of Informatics, as an Educational Product of a professional master's research. Through the App Inventor platform, CEL-LOGIC was designed to assist in the teaching and learning of beginners in programming and in the introduction of the PC, based on the pedagogical and technological guidelines for the development of mobile applications of the official curriculum documents of the technical schools of São Paulo. It followed the following methodological steps: theoretical studies, data collection via diagnostic evaluation form, application script, mentions obtained by students during the two months and use evaluation form. As results obtained, the acquisition of new knowledge through the Programming of Mobile Applications and the construction of learning from the PC stand out.

Keywords: computational thinking; teaching programming logic; MIT App Inventor; middle school.

RESUMEN

GUÍA DIDÁCTICA CEL-LOGIC: PENSAMIENTO COMPUTACIONAL Y LÓGICA DE PROGRAMACIÓN EN BACHILLERATO

El artículo presenta las bases para la construcción de la Guía Didáctica Interactiva CEL-LOGIC, para la enseñanza de la Lógica de Programación y el Pensamiento Computacional (PC), en el Liceo Integrado con el Técnico en el área de Informática, como Producto Educativo de una investigación profesional de maestría. A través de la plataforma App Inventor, CEL-LOGIC fue diseñado para ayudar en la enseñanza y el aprendizaje de principiantes en programación y en la introducción de la PC, basándose en las directrices pedagógicas y tecnológicas para el desarrollo de aplicaciones móviles de los documentos curriculares oficiales de las escuelas técnicas de São Paulo. Se siguieron las siguientes etapas metodológicas: estudios teóricos, recolección de datos a través del formulario de evaluación diagnóstica, guión de postulación, menciones obtenidas por los estudiantes durante los dos meses y formulario de evaluación de uso. Como resultados obtenidos, se destacan la adquisición de nuevos conocimientos a través de la Programación de Aplicaciones Móviles y la construcción del aprendizaje desde la PC.

Palabras clave: pensamiento computacional; enseñanza de la lógica de la programación; inventor de aplicaciones del MIT; secundaria.

Introdução¹

A informática na educação escolar representa-se como uma área em ascensão e relevância nas últimas décadas. Tal evolução crescente reflete as mudanças tecnológicas do século XXI, o que corrobora com a necessidade de integrar as tecnologias digitais aos processos educacionais de modo a contribuir com a preparação dos estudantes para um mundo cada vez mais digital. As tecnologias digitais mudaram a Edu-

cação Básica, pois a adoção crescente do digital alterou o modo como os estudantes aprendem, os professores ensinam e as escolas operam (Carneiro; Figueiredo; Ladeira, 2020).

O cenário das mudanças foi evidenciado durante a pandemia de COVID-19, iniciada em 2020, que contribuiu para que as transformações relacionadas ao uso do digital acentuassem. Com o fechamento de escolas para conter a propagação do Coronavírus, houve a transição em massa para o ensino on-line. As escolas

¹ Artigo revisado e normalizado por Luís Fernando Sarno.

tiveram que adotar tecnologias de aprendizado remoto para continuar as aulas, porém tal situação destacou as desigualdades digitais, uma vez que nem todos os estudantes tiveram acesso igual aos dispositivos e conexões de internet. A desigualdade digital, que já existia, se ampliou durante o período de isolamento, bem como as disparidades educacionais entre aqueles que tinham acesso às tecnologias e aqueles que não tinham (Gatti, 2020).

Situações adversas evidenciaram-se, a exemplo dos estudantes que passaram o ano de 2020 fazendo um curso de Informática sem ter computador e internet em casa, materiais essenciais para o desenvolvimento da aprendizagem na área (Moraes; Costa; Scholz, 2022).

Nesse contexto, o retorno dos estudantes para as aulas presenciais, no ano de 2022, se estabeleceu com dificuldades na assimilação de conteúdos de disciplinas básicas de programação. Além disso, há o velho problema do alto índice de evasão em cursos da área de tecnologia devido à complexidade e alto nível de abstração que a programação exige (Moraes; Costa; Scholz, 2022).

Assim, buscou-se articular o ensino da Lógica de Programação (LP) e o Pensamento Computacional (PC) no nível básico de educação, para que as dificuldades fossem melhoradas por meio da relação entre a LP e o PC, pois tal relação é estreita e fundamental para a compreensão e resolução de problemas por meio das tecnologias.

Por isso, o Ensino Médio Integrado ao Técnico foi escolhido como foco de uma pesquisa de mestrado profissional intitulada “A plataforma App Inventor e a construção do Pensamento Computacional no Ensino Médio integrado ao Técnico” (Cardinale, 2024), a qual pautou-se no princípio de que tal nível de ensino da Educação Básica visa preparar os estudantes de forma abrangente, oferecendo formação acadêmica sólida e integrando habilidades técnicas essenciais para o mundo do trabalho.

O Ensino Médio Integrado ao Técnico na área de Informática desempenha papel fundamental

na preparação dos estudantes para enfrentar os desafios do mundo digital, pois trata-se de um curso que em sua matriz curricular prevê a disciplina de LP e do Desenvolvimento de Aplicativos Móveis, ponto que este artigo apresenta como meio para facilitar a aprendizagem de programação.

A inclusão dos conceitos de PC no currículo desse nível de ensino proporciona aos estudantes a oportunidade de desenvolver habilidades de raciocínio lógico, decomposição de problemas, abstração de informações e criação de algoritmos aprimorados e eficientes (Wing, 2016).

Assim, é fundamental à relação entre PC e LP, além da integração de forma relevante e acessível ao currículo. Ao fazer isso, os estudantes têm a oportunidade de preparação para o futuro, tornando-se capazes de enfrentar desafios complexos e contribuir de maneira significativa para a sociedade em constante transformação tecnológica.

O presente texto parte de um recorte de uma pesquisa maior (Cardinale, 2024), a qual objetivou verificar a viabilidade e adequação do material em conjunto com a plataforma de programação didática, conceitos de programação para aplicativos móveis e conceitos de PC, de modo a colaborar com a melhora do processo de ensino e aprendizagem da Lógica de Programação. Desta forma, o presente artigo objetiva apresentar as bases para a construção do Guia Didático Interativo CEL-LOGIC (GITHUB, 2024), para o ensino de Lógica de Programação e Pensamento Computacional (PC), no Ensino Médio Integrado ao Técnico na área de Informática, enquanto Produto Educacional de uma pesquisa de mestrado profissional.

O Guia Didático Interativo chamado CEL-LOGIC (GITHUB, 2024) foi desenvolvido com a utilização do MIT App Inventor (2022) para o Desenvolvimento de Aplicativos Móveis, com o propósito de auxiliar no ensino e aprendizagem dos iniciantes em programação e na introdução do PC para solução de desafios.

A escolha do MIT App Inventor (2022) como instrumento facilitador se deu em virtude da

sua acessibilidade e interface amigável, permitindo que estudantes de diferentes níveis de habilidade participem ativamente do processo de aprendizagem. A construção do PC, aliada ao uso do MIT App Inventor (2022), emergiu como uma possibilidade didático e pedagógica no contexto da educação digital, de modo dinâmico e alinhado às demandas do mundo digital em constante evolução.

Metodologia da pesquisa

A abordagem qualitativa, seus fundamentos e premissas foram a base para o desenvolvimento dos passos para a construção do Produto Educacional em pauta (Ludke; André, 1986). De acordo com Silva e Menezes (2005), supera a mera representação numérica dos dados, buscando aprofundar a compreensão sobre o assunto. O domínio da pesquisa qualitativa abrange uma variedade de abordagens metodológicas que viabilizam um processo dinâmico, permitindo a adaptação a novas formas de coleta e análise de dados. Assim, seguiu-se os seguintes passos: estudos teóricos, coleta de dados via formulário de avaliação diagnóstica, roteiro de aplicação, menções obtidas pelos estudantes durante os bimestres e formulário de avaliação do uso.

O estudo contou também com produção de protocolo para validação de dados, pois a pesquisa qualitativa, de acordo com Silva e Menezes (2005), está comprometida com o aprofundamento da compreensão de um grupo social. Flick (2013) afirma que o início de uma pesquisa qualitativa é a produção de protocolo para compreender o contexto dos eventos e sua ocorrência.

A pesquisa foi desenvolvida em etapas, a saber: Etapa 1 – aplicação do formulário de avaliação e socioeconômico: consistiu na aplicação de formulário on-line para identificação do conhecimento prévio do estudante em Programação e conceitos de PC, caracterização socioeconômica; Etapa 2 – estudos teóricos; Etapa 3 – desenvolvimento do Guia Didático

Interativo CEL-LOGIC por meio da plataforma MIT App Inventor; Etapa 4 – análise da avaliação de usabilidade do Produto Educacional.²

Os estudos teóricos

Para conhecer os resultados das pesquisas sobre o tema do presente artigo, foram realizadas buscas em quatro bases de dados: Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), no portal de periódicos Capes/MEC; Portal de Teses e Dissertações da CAPES – Plataforma Sucupira; Scientific Electronic Library Online (SciELO); e Google Acadêmico.

Com o objetivo de aproximar as buscas do estudo em questão, as investigações foram realizadas com aplicação de filtros, ou seja, critérios de inclusão foram definidos, a saber: delimitação do tempo para os últimos cinco anos e utilizando a junção dos seguintes descritores: “Ensino de Programação” E “Pensamento Computacional” (1); “App Inventor” E “Ensino de Programação” (2); “Metodologias Ativas” E “App Inventor” (3); “Ensino de Programação” E “Ensino Médio Integrado ao Técnico” (4).

A busca foi delimitada ainda por trabalhos escritos nos últimos cinco anos, utilizando os filtros: ciências humanas, tecnologia, educação. No Portal de Periódicos CAPES/MEC, a partir da junção dos descritores: “Ensino de Programação” E “Pensamento Computacional”, a pesquisa retornou 65 resultados. Para os descritores “App Inventor” E “Ensino de Programação”, a pesquisa retornou 16 resultados; para os descritores “Metodologias Ativas” E “App Inventor”, 4 resultados; e para os descritores “Ensino de Programação” e “Ensino Médio Integrado ao Técnico”, 1 resultado.

No Portal de Teses e Dissertações da CAPES – Plataforma Sucupira, a partir da junção dos descritores “Ensino de Programação” E “Pensamento Computacional”, a pesquisa retornou 133 resultados; para os descritores “App Inventor” E “Ensino de Programação”, 19

² Em virtude do recorte para a construção do artigo, a descrição da aplicação do Produto Educacional foi excluída.

resultados; para os descritores “Metodologias Ativas” E “App Inventor”, 3 resultados; e para os descritores “Ensino de Programação” E “Ensino Médio Integrado ao Técnico”, 381 resultados.

No Scielo, a partir da junção dos descritores “Ensino de Programação” E “Pensamento Computacional”, a pesquisa retornou 4 resultados; para os descritores “App Inventor” E “Ensino de Programação”, nenhum resultado; e para os descritores “Metodologias Ativas” E “App Inventor” e “Ensino de Programação” E “Ensino Médio Integrado ao Técnico”, não retornou resultados, fato que apresenta uma lacuna a ser investigada por esta pesquisa, entretanto, foram encontradas pesquisas relevantes sobre o uso de Metodologias Ativas e uso da ferramenta App Inventor em outras áreas de conhecimento e nos níveis fundamental e superior, e nenhum resultado para o nível médio integrado ao técnico.

No Portal Google Acadêmico, a partir da junção dos descritores “Ensino de Programação” E “Pensamento Computacional”, a pesquisa retornou 1.520 resultados; para os descritores “App Inventor” E “Ensino de Programação”, 422 resultados; para os descritores “Metodologias Ativas” E “App Inventor”, 237 resultados; e para os descritores “Ensino de Programação” E “Ensino Médio Integrado ao Técnico”, 42 resultados.

Portanto, foram realizadas as leituras dos títulos e resumos dos trabalhos, a partir dos critérios: inclusão baseados na contribuição dos trabalhos para a área de pesquisa em questão, alinhados aos objetivos e à questão norteadora deste estudo. Os critérios de exclusão foram determinados pela falta de relação dos trabalhos com os objetivos e a questão norteadora, ou seja, um total de treze trabalhos, sendo esses artigos científicos e dissertações.

Ferreira e Braga (2023), em “Ensino de Programação e as estratégias pedagógicas utilizadas no Brasil”, realizaram análise de conteúdo nos artigos das principais revistas da área de Ensino, no período de 1996 a 2016, a fim de elucidar como os ambientes educa-

cionais são transformados pela Programação de Computadores, proporcionando assim o protagonismo discente. Os resultados mostram uma conjunção de estratégias pedagógicas no âmbito brasileiro que apontam para um “casamento” entre o Ensino da Programação e as Metodologias Ativas. Verificou-se também, que os “Ambientes de Programação” ampliaram o entendimento sobre os impactos da ciência e tecnologia na sociedade, buscando, por meio do aprender fazendo e da aprendizagem colaborativa, orientar resoluções de problemas reais e auxiliar na tomada de decisões, características de vidas imersas na cibercultura.

Beleti Junior e Sforzi (2023), em “Pesquisas experimentais no desenvolvimento do Pensamento Computacional: um mapeamento sistemático da literatura no ensino de conceitos de programação”, apresentaram como objetivo identificar as pesquisas experimentais sobre o desenvolvimento do Pensamento Computacional no ensino de conceitos de Computação. Constataram que não há um padrão no formato da realização das investigações experimentais, mas tendem a ser de curta duração e, apesar das diversas subáreas da Ciência da Computação, a programação tem sido predominantemente utilizada no desenvolvimento do Pensamento Computacional.

Moraes, Costa e Scholz (2022), em “Mapeamento Sistemático do Ensino Introdutório de Programação no Ensino Técnico e Superior no Brasil”, buscaram promover melhor compreensão dos desafios de ensino e aprendizagem nas disciplinas introdutórias de programação no Ensino Técnico e Superior, por meio de um mapeamento sistemático da literatura.

Carvalho e Braga (2022), em “Pensamento Computacional na Educação Brasileira: um olhar segundo artigos do Congresso Brasileiro de Informática na Educação”, objetivaram entender melhor o termo Pensamento Computacional, assim procederam a uma revisão dos artigos publicados entre os anos de 2015 e 2019 no Workshop de Ensino em Pensamento Computacional, Algoritmos e Programação

(WalgProg), que ocorre no âmbito do Congresso Brasileiro de Informática na Educação (CBIE). Os resultados apontaram que havia várias publicações que citavam o termo de maneira descontextualizada ou atribuíam significados errôneos, porém analisaram uma crescente preocupação em trazer maior embasamento ao termo quando surge no trabalho, destacando o potencial interdisciplinar do Pensamento Computacional, especialmente associado à resolução de problemas.

Costa e Piedade (2021), em “Uso do aplicativo App Inventor na aprendizagem de programação: uma revisão sistemática da literatura entre 2011 e 2020”, salientam as potencialidades do ambiente App Inventor para a aprendizagem de conceitos de programação e revelam a falta de estudos experimentais que analisem o efeito da sua utilização na aprendizagem e na motivação dos estudantes.

Faria, Alves e Moreira Costa (2021), em “Uma experiência de inserção de Linguagem de Programação no Ensino Médio Integrado à Educação Profissional Técnica a partir da flexibilização curricular”, apresentaram o princípio de flexibilização curricular e inovações nos projetos pedagógicos para atender às novas realidades, entre as quais destacaram a imersão tecnológica da sociedade contemporânea. Nesse sentido, apresentaram uma ementa curricular como aporte teórico-metodológico ao desenvolvimento de competências relacionadas à aprendizagem de Linguagem de Programação. A ementa foi consolidada mediante uma abordagem holística fundamentada em critérios como coerência, exequibilidade e curva de aprendizagem que proporcionem um caráter globalizante em razão da heterogeneidade das formações. E seus princípios se encontraram na conexão entre as formações profissionais e a implementação tecnológica.

Santos, Lico e Moura (2022), em “Relatos de experiência didática baseada no ambiente App Inventor aplicada ao Ensino Médio”, basearam-se na aprendizagem ativa, que consistiu na iniciação à Lógica de Programação por meio da

criação de jogos digitais pelo usuário final, com suporte de um ambiente tecnológico propício: o App Inventor. Os resultados mostraram que o App Inventor forneceu a motivação necessária para os estudantes aprenderem, além dos conceitos de programação para a criação de jogos digitais, a trabalhar ativamente em grupo, a testar e discutir soluções. Os resultados obtidos também evidenciam grande interesse dos estudantes do Ensino Médio pela programação de aplicativos e, principalmente, pelo uso do celular como uma ferramenta de apoio à aprendizagem na escola.

Berssanette e Francisco (2021), em “Metodologias Ativas de Aprendizagem no Contexto do Ensino – Aprendizagem de Programação de Computadores: uma revisão sistemática da literatura”, destacaram o uso de Metodologias Ativas de aprendizagem e as possibilidades destas para o processo de ensino e aprendizagem de programação de computadores, bem como a ausência de estudos específicos que retratam o panorama atual de pesquisas que associam as diferentes técnicas pedagógicas de aprendizagem ativa a esse contexto. Buscaram identificar e caracterizar os estudos que relacionam as Metodologias Ativas de aprendizagem no contexto de ensino e aprendizagem de programação de forma a observar as abordagens, as técnicas pedagógicas utilizadas, a aplicação, as contribuições e as dificuldades relatadas de implementação de uso dessas metodologias no contexto já aludido.

Farias (2021), em “Análise comparativa sobre habilidades do Pensamento Computacional com estudantes do Ensino Médio”, objetivou identificar, mediante a aplicação de um teste, o conhecimento dos estudantes do Ensino Médio de uma escola estadual, antes e após as atividades do curso “Programação para Celular”, ofertado pelo projeto de extensão intitulado “Inclusão de estudantes do Ensino Médio da rede pública nas ciências exatas através da prática da lógica e da programação”, vinculado ao programa de extensão PROBEX da Universidade Federal de Campina Grande (UFCG).

Kaminski, Kluber e Boscaroli (2021), em “Pensamento Computacional na Educação Básica: Reflexões a partir do Histórico da Informática na Educação Brasileira”, apresentaram atividades que propuseram o desenvolvimento do Pensamento Computacional, que têm sido cada vez mais empregadas em ambientes escolares, além de algumas contribuições para o desenvolvimento de aspectos cognitivos as quais foram relatadas na literatura. Todavia, considerando que o esforço de inserir o PC na Educação não é tão recente, mas tem raízes no início da história da Informática na Educação, e que as práticas pedagógicas precisam ser pensadas e conduzidas com objetivos claros, em conformidade com a visão de mundo que se tem e que se almeja. Assim, foi pertinente refletir acerca das razões que justificaram atividades com PC na escola, e sobre o que a história nos ensinou a respeito da sua integração aos processos de ensino e aprendizagem.

Ferreira e Duarte (2019), em “Ensino de Programação: trajetória histórico-social e os avanços na cultura digital do Brasil”, realizaram um mapeamento histórico-social da trajetória de crescimento do Ensino de Programação em território nacional, acompanhando tendências mundiais de letramento digital. Mostraram três fases distintas e uma transição desta prática pedagógica do Ensino Superior para a Educação Básica. Encontraram diversificação no uso de tecnologias e aproximação dos conteúdos curriculares por meio de abordagens lúdicas e interdisciplinares, como a construção de jogos e de robôs, além do desenvolvimento da autonomia discente e o favorecimento do seu protagonismo.

Ferrari, Saheb e Torres (2019), em “Metodologias Ativas e tecnologias digitais: aproximações e distinções”, ao tratarem da temática Metodologias Ativas e tecnologias digitais, objetivaram discutir conceitos, além de descreverem a origem e seus elementos constitutivos. O *corpus* da pesquisa abordou conceitos sobre Metodologias Ativas e tecnologias, destacando as digitais na atualidade, além das origens e

características das Metodologias Ativas e suas possíveis relações com o uso de tecnologias digitais.

Ferreira (2017), em “Uma visão sobre as estratégias do ensino de programação no Brasil como respostas as demandas da cibercultura”, afirmou que o Ensino de Programação de Computadores se posicionava como uma tendência global para a Educação Básica. Países como Estados Unidos, Inglaterra, Finlândia e Austrália já o incluíram em seus currículos escolares, devido aos benefícios apontados por pesquisas acadêmicas para esta prática pedagógica. A epistemologia construcionista, na qual este tipo de ensino tem sua fundamentação teórica, está alinhada aos anseios de uma cidadania global da ecologia cognitiva, impulsionadas pela cibercultura, que tem alterado as formas de vida na sociedade do conhecimento e conectada em rede.

Souza, Batista e Barbosa (2016), em “Problemas e Dificuldades no Ensino de Programação: Um Mapeamento Sistemático”, apontaram para tendências e necessidades de pesquisa em ensino e aprendizagem de programação, bem como forneceram aos educadores em programação um catálogo de trabalhos que propuseram soluções para os problemas enfrentados em cenários reais de ensino e aprendizagem.

Dessa forma, após análises e estudos dos trabalhos citados nas buscas realizadas, foi possível delinear o arcabouço teórico para a construção do Produto Educacional.

Os passos para o desenvolvimento do CEL-LOGIC

O CEL-LOGIC foi desenvolvido de acordo com o princípio teórico do Construcionismo de Papert (1988), que foi um dos primeiros intelectuais a reconhecer, em uma época na qual os computadores eram utilizados apenas para entretenimento, compras ou trabalho, o potencial do uso dos computadores na educação.

Papert (1988) faz uma relação com as teorias das aptidões usando o termo “matofobia”,

por ele cunhado, o qual está relacionado à matemática, porém o conceito pode ser abordado para a Programação, diante do contexto de aplicação.

O conceito de “matofobia” impede que pessoas aprendam qualquer coisa que reconheçam como matemática, embora não tenham dificuldade com o conhecimento matemático quando não o percebem como tal. Desse modo, as deficiências tornam-se a identidade e o aprendizado é transformado em uma tarefa desagradável, permeada de inseguranças e restrições autoimpostas (Papert, 1988).

Outra constatação importante de Papert (1988) foi que estudantes têm sua aprendizagem atrasada porque possuem um modelo de aprendizagem no qual existe, apenas, certo e errado e, quando se trata de Programação de Computadores, dificilmente se acerta na primeira tentativa, justamente por ter que elaborar as maneiras como seu algoritmo vai acontecer e testar. Aprender Programação é aprender a se tornar altamente hábil em prever, isolar e corrigir erros; não se está certo ou errado, mas se é executável.

Ainda de acordo Papert (2008), qualquer coisa é simples se a pessoa conseguir incorporá-la ao seu arsenal de modelos, caso contrário tudo pode ser extremamente difícil. Portanto, a aprendizagem é facilitada quando são utilizados modelos para assimilação, pois os modelos facilitam o acesso a conceitos abstratos. Um exemplo disso pode ser visualizado quando o estudante aprende o conceito de “variável”, por meio da Programação de computadores (modelo), e a partir daí consegue compreender o conceito de “incógnita”, presente em uma equação matemática (assimilação), tornando o aprendizado mais amigável (Massa; Oliveira; Santos, 2022).

Papert (2008) presumia que a integração do computador na educação facilitaria a criação de condições para a consolidação desses paradigmas cognitivos, pois a essência do computador é a universalidade e o seu poder de simulação, no qual poderia assumir milhares de formas,

servir a milhares de finalidades e atrair milhares de gostos.

Em sua pesquisa sobre os computadores, esse autor abordou dois tópicos: aprender a usar o computador e programá-lo em um processo natural (assim como se aprende a falar) e aprender a se comunicar com o computador pode mudar a maneira como outras aprendizagens acontecem, citando como exemplo a matemática.

Nesse sentido, o computador não deveria orientar o aprendizado do estudante, pelo contrário, o estudante deveria instruir o computador, programando-o. A Programação consiste em aprender uma linguagem compreensível pelo computador, permitindo assim que se comunique efetivamente com a máquina. Assim, ensinar a programar o computador proporciona a sensação de domínio sobre a máquina, ao mesmo tempo em que estabelece um contato íntimo com conceitos mais profundos.

Papert (1988) também foi responsável pela criação do termo Construcionismo, teoria baseada no Construtivismo de Jean Piaget. O Construcionismo enfatiza a importância da construção ativa do conhecimento e do aprendizado por meio da prática, propõe que os estudantes aprendem melhor quando estão envolvidos na criação de algo tangível.

O principal objetivo do Construcionismo é ensinar de forma a produzir a maior aprendizagem a partir do mínimo de ensino. Ao contrário da abordagem educacional tradicional, na qual o professor transmite instruções aos estudantes, que aprendem descobrindo por si mesmos o conhecimento necessário, enquanto o educador desempenha o papel de mediador. Papert (1988) concebe o aprendiz não como alguém que apenas responde a estímulos externos, mas um indivíduo ativo com a capacidade de analisar e interpretar fatos e ideias, sendo capaz de construir seu próprio conhecimento.

Para isso, o computador desempenha papel fundamental como instrumento para

implementar o Construcionismo, abordando aspectos cruciais, tais como: incentivar o estudante a ser um produtor de tecnologia, não apenas consumidor passivo; e capacitá-lo a assumir o papel de protagonista em seu próprio processo de aprendizagem, promovendo autonomia. Além do computador, esse autor também enfatizava a importância da criação de ambientes de aprendizagem enriquecedores, aos quais se refere como micromundos. Esses micromundos são considerados meios eficazes para aprimorar o ensino, despertando o interesse e a curiosidade das estudantes, promovendo assim um aprendizado efetivo (Papert, 2008).

Essa abordagem destaca a importância de aprender fazendo e construindo, em vez de apenas consumir informações passivamente. O Construcionismo de Papert (1988) influenciou a educação e a tecnologia, especialmente no contexto do uso da Programação e da tecnologia como ferramentas para aprimorar o processo de aprendizado.

Para contribuir com a construção de um ambiente propício para o desenvolvimento do PC e com o objetivo de atenuar os desafios no processo de ensino e aprendizagem da Programação, algumas abordagens têm sido exploradas. Uma delas sugere que a introdução à Programação seja facilitada por meio de ferramentas computacionais que capacitem os estudantes a explorar, experimentar, testar e aprender com seus próprios erros.

O uso da plataforma MIT App Inventor (2022) permite ao estudante desenvolver o raciocínio lógico utilizando blocos operacionais para programar seus próprios aplicativos, seguir seu próprio ritmo, ser protagonista de sua aprendizagem e trabalhar colaborativamente.

Nesse sentido, o MIT App Inventor (2022) é um ambiente de programação visual e intuitivo que permite criar aplicativos totalmente funcionais para celulares e tablets. Busca democratizar o desenvolvimento de software, capacitando jovens a passarem de consumidores de tecnologia para os criadores dela.

A programação se fundamenta em componentes que são manipulados por meio de eventos, métodos e propriedades. Este paradigma de Programação é conhecido como Programação orientada a eventos, ou seja, o comportamento do aplicativo depende de interações com o usuário (Gomes; Melo, 2013).

O CEL-LOGIC (GITHUB, 2024), fundamentado nos princípios do Construcionismo de Papert (1988), surge como uma abordagem que busca transformar os desafios da aprendizagem da Programação em oportunidades para o desenvolvimento cognitivo, ao enfatizar a prática ativa e a criação tangível como elementos centrais no processo educativo. Essa perspectiva contrasta com o aprendizado passivo e tradicional, oferecendo aos estudantes um papel de protagonistas em seu próprio desenvolvimento. Além disso, ao abordar a “matofobia” e os paradigmas binários de certo e errado, esse autor ressalta a importância de modelos que promovam a experimentação e a correção de erros como parte natural do aprendizado. O uso de ferramentas como o MIT App Inventor (2022) reflete essa filosofia, proporcionando um ambiente inclusivo e intuitivo para explorar, testar e construir conhecimentos em Programação.

O Produto Educacional CEL-LOGIC (GITHUB, 2024) foi desenvolvido a partir de projetos do e-book Kodular – Componentes: Volumes 1 e 2 (Silva, 2021), com adaptações para o ambiente do MIT App Inventor (2022) e seus tutoriais que estão disponíveis no próprio site. O MIT App Inventor (2022) foi escolhido pela gratuidade, suporte para o idioma português do Brasil, ambiente de fácil manipulação pelo estudante e ambiente on-line, não necessitando nenhum tipo de instalação acessado direto pelo navegador.

O CEL-LOGIC (GITHUB, 2024) foi definido a partir das necessidades dos estudantes expressas no formulário on-line aplicado na primeira semana de aula pela gestão da escola, aos estudantes do Ensino Médio integrado ao Técnico de uma escola estadual paulista. Portanto, de

acordo com as respostas e buscando trabalhar o raciocínio lógico e o PC, o tema do Guia foi delimitado em resolução de problemas.

Seguindo a teoria Construcionista para o desenvolvimento do Produto Educacional, foram criados desafios para que o estudante elaborasse estratégias para a solução do problema, que deveria envolver investigação e reflexão sobre a prática realizada. Dessa forma, possibilitando que o estudante exercitasse o ato de “pensar sobre pensar”, pois este pode escolher ou descobrir sua forma de pensamento para a solução do problema (Papert, 1988).

Nesse sentido, a lógica que é desenvolvida pelo PC oferece oportunidade de avanço tanto cognitivo quanto emocional, permitindo a construção de significados por meio da reflexão do seu processo de pensamento.

Segundo Zabala (2010), pressupõe-se que estrutura cognitiva é configurada por uma rede de esquemas de conhecimento, pois esses esquemas são definidos como as representações que a pessoa possui sobre algum objeto de conhecimento. Ao longo da vida, esses esquemas são revisados, modificados, tornam-se mais complexos e adaptados à realidade. A natureza dos esquemas de conhecimento do estudante depende de seu nível de desenvolvimento e dos conhecimentos prévios que pôde construir; a situação de aprendizagem pode ser concebida como um processo de comparação, revisão e construção de esquemas de conhecimento sobre os conteúdos escolares.

Para que esse processo seja desencadeado, não é suficiente que os estudantes se deparem com conteúdos para aprender, é necessário que, diante deles, possam atualizar seus esquemas de conhecimento, compará-los com o que é novo, identificar semelhanças e diferenças e integrá-las em seus esquemas, verificando que o resultado possui certa coerência. Quando tudo isso acontece, ou na medida em que acontece, pode-se afirmar que está ocorrendo uma aprendizagem significativa dos conteúdos apresentados (Zabala, 2010).

Além disso, Zabala (2010) ressalta que, para que o estudante realize uma aprendizagem significativa, é fundamental que a aula seja concebida de forma a contemplar diversas facetas do processo de aprendizagem, as quais podem ser alcançadas dependendo da abordagem adotada em relação ao conteúdo. Esse autor classifica os conteúdos como: factual, conceitual, procedimental e atitudinal.

As atividades propostas no CEL-LOGIC (GITHUB, 2024) procuraram incluir esses princípios de Zabala (2010) para conduzir o estudante ao conhecimento não apenas com o objetivo de descobrir a melhor solução para as atividades propostas, mas aprimorando os processos mentais por meio do PC.

Portanto, espera-se que o estudante avance em competências e habilidades para solucionar problemas, pensar de forma estruturada e crítica, além do desenvolvimento do raciocínio lógico e criatividade.

A estratégia abordada com as atividades do CEL-LOGIC (GITHUB, 2024) usou o Pensamento Computacional que envolve a lógica como base (Wing, 2016), dividindo as atividades em etapas: decomposição, identificação de padrões, abstração e o algoritmo. Na decomposição, o estudante precisa entender o objetivo, qual a função do aplicativo, o que vai fazer. Quando identifica o objetivo, também identifica padrões, acessando sua rede de esquemas de conhecimento (Zabala, 2010), verifica se daquele objetivo consegue reconhecer alguma estrutura computacional da disciplina de Lógica de Programação, já cursada na 1ª série, abstraindo os elementos importantes e, por fim, organizando o algoritmo. No CEL-LOGIC (GITHUB, 2024), esse processo se resume a: O que o aplicativo vai fazer? Qual aparência ele vai ter? Como vou fazê-lo funcionar? (objetivo, design, programação).

Em cada atividade proposta também foram abordados conceitos importantes de Programação como: declaração de variáveis, funções condicionais, vetores e uso de recursos do celular.

As contribuições do CEL-LOGIC

Nesta seção, são apresentadas as contribuições do CEL-LOGIC (GITHUB, 2024) pautadas na discussão teórica previamente abordada. A aprendizagem da programação de computadores pode ser desafiadora para vários estudantes, especialmente para aqueles que estão sendo introduzidos a conceitos e linguagens de programação pela primeira vez (Souza; Batista; Barbosa, 2016).

As dificuldades podem surgir de várias fontes, incluindo a complexidade dos conceitos abstratos, a necessidade de pensar de forma algorítmica e a sintaxe específica das linguagens de programação. Além disso, a falta de experiência prévia em matemática ou lógica de computação pode ampliar essas barreiras (Ferreira; Duarte, 2019).

As frustrações podem se acumular rapidamente quando os estudantes se deparam com erros de sintaxe, problemas de lógica ou dificuldades na resolução de problemas. No entanto, superar essas dificuldades não apenas requer prática e persistência, mas estratégias de ensino eficazes que possam ajudar os estudantes a entender os conceitos fundamentais e a desenvolver suas habilidades de resolução de problemas (Costa; Piedade, 2019).

Uma possível solução para superar as dificuldades dos estudantes com Programação é adotar uma abordagem pedagógica que seja acessível e progressiva, começando com conceitos básicos e construindo conhecimentos gradualmente (Costa; Piedade, 2019).

O uso de recursos visuais, exemplos práticos e exercícios interativos pode ajudar a tornar os conceitos de programação mais tangíveis e fáceis de entender. Incentivar a colaboração entre os estudantes, por meio de projetos em equipe, pode promover um ambiente de aprendizagem mais solidário, no qual possam compartilhar conhecimentos e se apoiar mutuamente (Berssaneti; Francisco, 2021). Oferecer retorno construtivo e individualizado

pode ajudá-los a identificar e corrigir erros de forma mais eficaz, incentivando-os a aprender com seus próprios desafios (Ferrarini; Saheb; Torres, 2019).

Em última análise, ao combinar uma abordagem didática inclusiva com ferramentas e recursos adequados, é possível tornar a programação uma disciplina mais acessível para todos (Costa; Piedade, 2019).

Além disso, Piedade e Costa (2021) afirmam que o MIT App Inventor é um instrumento ideal para incentivar o interesse dos estudantes, pois contribui para a superação dos obstáculos encontrados durante a aprendizagem dos conceitos introdutórios de Programação.

De acordo com as pesquisas de Santos, Lico e Moura (2021), os resultados indicaram que o App Inventor proporcionou a motivação essencial para os estudantes aprenderem os conceitos de programação e a colaborar ativamente em grupo, testar e debater soluções. Os resultados destacam interesse dos estudantes do Ensino Médio pela Programação de aplicativos móveis, especialmente pelo uso do celular como um instrumento de apoio à aprendizagem.

O Guia Didático Interativo CEL-LOGIC (GITHUB, 2024) apresenta cinco projetos. São projetos simples e fáceis mostrando como usar, passo a passo, componentes básicos, sem necessidade de digitar uma linha de código e tampouco decorar comandos complicados, utilizando, apenas, blocos de código que serão explicados em detalhes.

Os projetos propostos no CEL-LOGIC (GITHUB, 2024) têm como objetivo contribuir para que o estudante progrida em cada tarefa até que tenha um aplicativo móvel completo e funcional desenvolvido por ele.

A Figura 1 mostra a primeira atividade do CEL-LOGIC (GITHUB, 2024), que consiste em alterar o texto de uma legenda após o botão ser clicado. Quando o botão for clicado, o texto da legenda que está vazio será alterado para o texto “Olá mundo!”.

Figura 1 – Projeto 1 do CEL-LOGIC

PROJETO 1 – HELLO WORLD

OBJETIVO

Ao clicar no botão a mensagem “Olá mundo” será mostrada em uma legenda.

DESIGN

Aqui está uma lista de itens que você usará para criar o seu design:

- 1 botão;
- 1 legenda escondida.

Legenda: é o componente usado para mostrar texto, ela exibe textos modificados na propriedade texto. Para uma legenda ser escondida, basta apagar o texto dela. As legendas estão localizadas na paleta interface com o usuário

Botão: é o componente usado para executar tarefas quando clicado. O botão está localizado na paleta interface com o usuário.

SUGESTÃO DE ALTERAÇÃO DE PROPRIEDADES

↔ **ALTERANDO AS PROPRIEDADES DA SCREEN 1**

- Alinhamento horizontal: center
- Título: Olá Mundo

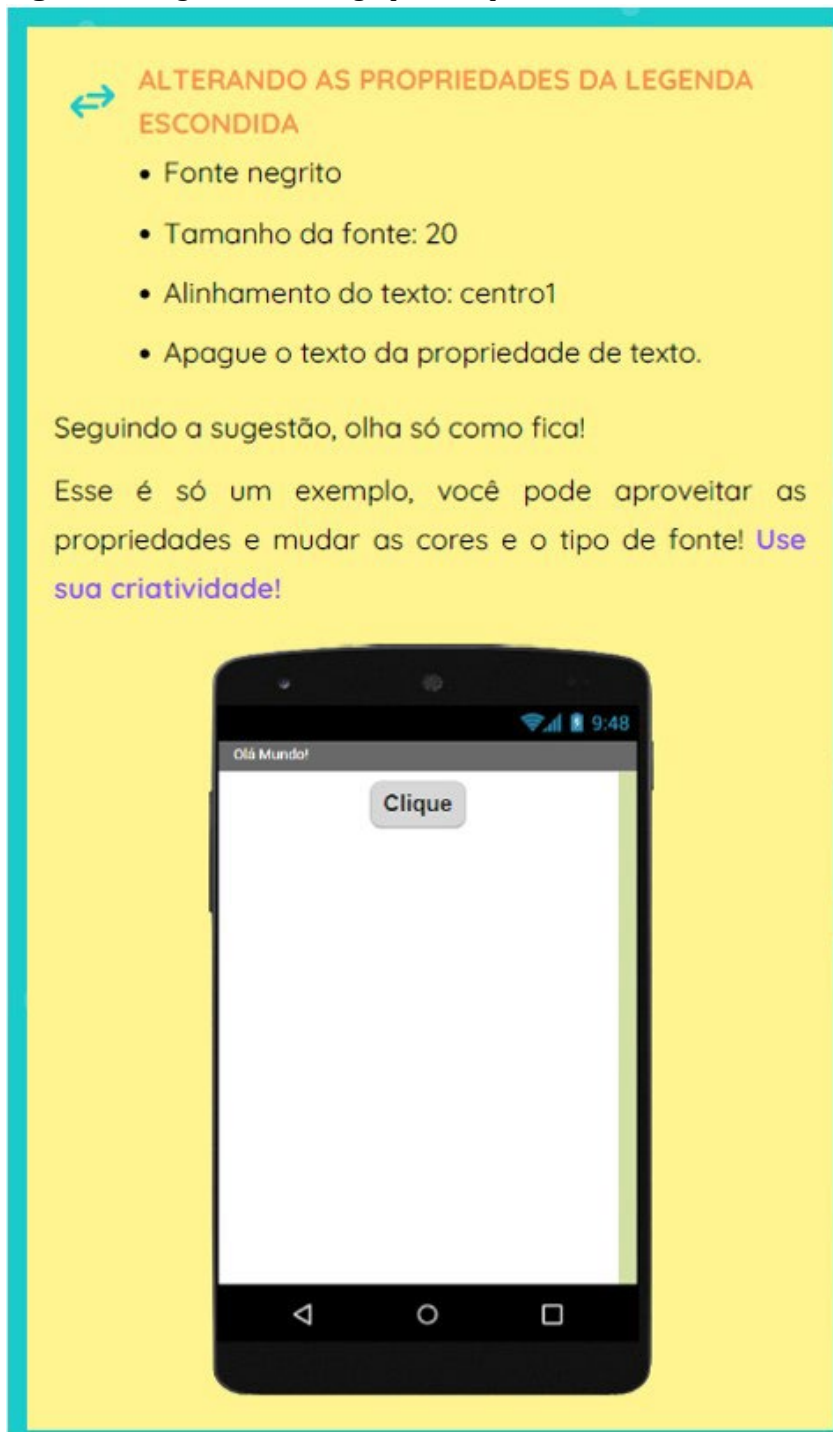
↔ **ALTERANDO AS PROPRIEDADES DO BOTÃO**

- Fonte negrito
- Tamanho da fonte: 20
- Forma: arredondado
- Alinhamento do texto: centro1

Fonte: Elaborado pelas autoras deste artigo.

A Figura 2 sugere o uso de componentes de design e alteração de suas propriedades, como os botões e legendas.

Figura 2 – Sugestão de design para o aplicativo



Fonte: Elaborado pelas autoras deste artigo.

As Figuras 3 e 4 ilustram uma abordagem para programar o aplicativo, mostrando os blocos de código e detalhando suas funcionalidades. Além disso, propõem um desafio para

que o estudante possa aprimorar o aplicativo de forma independente. O estudante também pode ver o resultado do aplicativo por meio de um link que o redireciona para um vídeo.

Figura 3 – Sugestão para programar o aplicativo

PROGRAMAÇÃO

Clique em seu botão e arraste o evento:

quando Botão1 .Clique
fazer

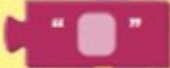
➡ Os bloco de evento (blocos marrom claro): desencadeiam as ações dentro do aplicativo.

Clique em sua legenda e arraste o evento:

ajustar Legenda1 . Texto para

➡ O bloco de propriedade (blocos verde escuro e verde claro): ajustam propriedades por meio de código dos elementos de design.

Clique nos blocos fixos de Texto e arraste o texto vazio:



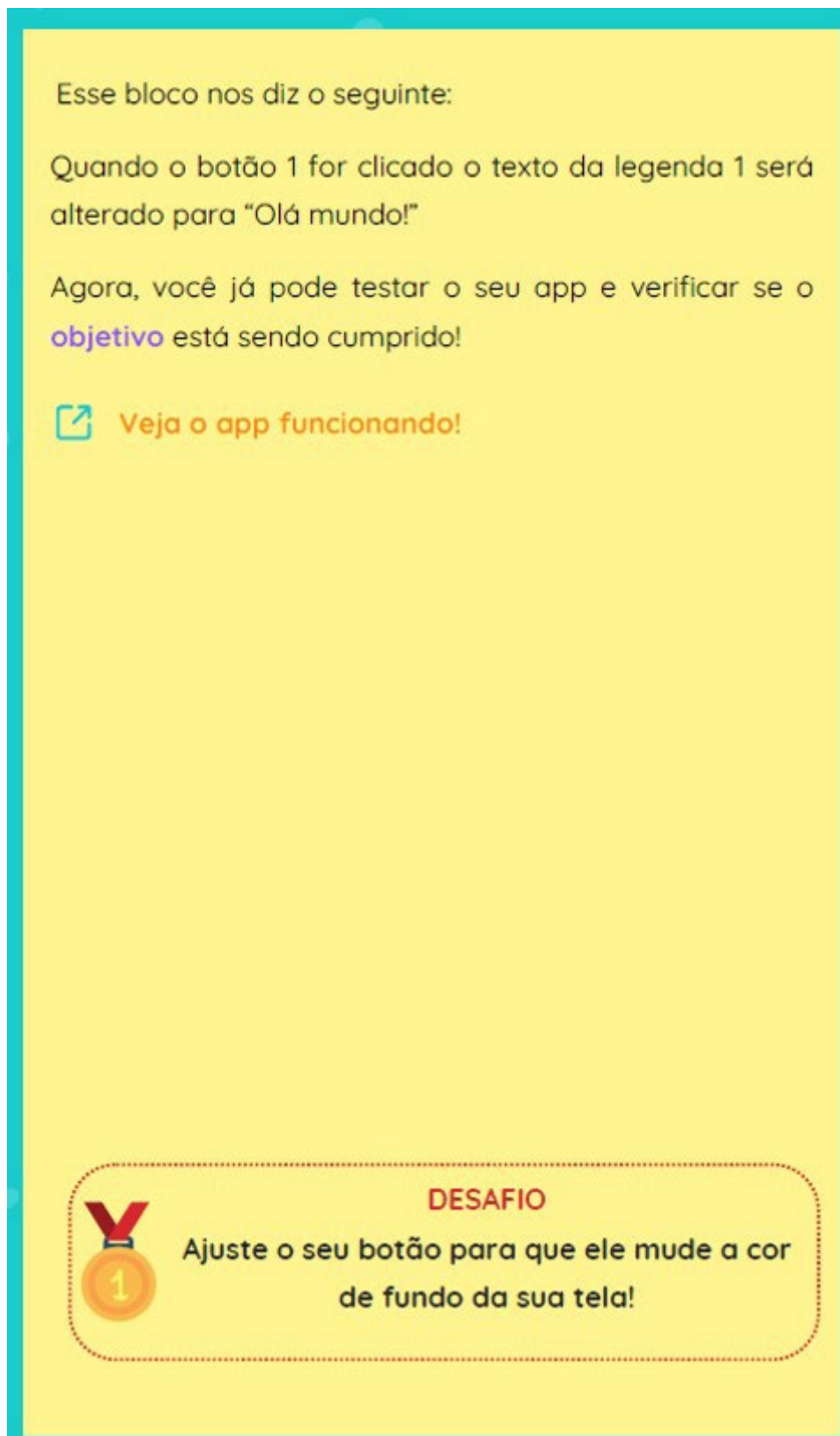
Juntando todos os blocos, como um quebra cabeça, temos o código do app pronto!

quando Botão1 .Clique
fazer

ajustar Legenda1 . Texto para "Olá mundo!"

Fonte: Elaborado pelas autoras deste artigo.

Figura 4 – Sugestão para programar o aplicativo



Esse bloco nos diz o seguinte:

Quando o botão 1 for clicado o texto da legenda 1 será alterado para "Olá mundo!"

Agora, você já pode testar o seu app e verificar se o **objetivo** está sendo cumprido!

 **Veja o app funcionando!**

 **DESAFIO**

Ajuste o seu botão para que ele mude a cor de fundo da sua tela!

Fonte: Elaborado pelas autoras deste artigo.

Portanto, considera-se que os principais desafios no ensino e aprendizagem de Programação estão relacionados à dificuldade dos alunos em compreender os conceitos, aplicá-los na construção de programas e à falta de motivação para realizar as atividades propostas. No entanto, essas dificuldades podem ser mitigadas por meio da visualização de programas e algoritmos, além de seu desenvolvimento em ambientes pedagógicos voltados para o ensino de Programação (Souza; Batista; Barbosa, 2016).

Nesse contexto, o CEL-LOGIC (GITHUB, 2024) contribui ao apresentar uma forma prática de programar o aplicativo, incluindo a exibição dos blocos de código com explicações detalhadas e propondo um desafio que incentiva os estudantes a melhorar o aplicativo de maneira autônoma, unindo teoria e prática em um ambiente motivador.

Considerações finais

O artigo buscou apresentar as bases para a construção de um Guia Didático Interativo denominado CEL-LOGIC (GITHUB, 2024), para o ensino de Lógica de Programação e Pensamento Computacional (PC) no contexto do Ensino Médio Integrado ao Técnico na área de Informática, pois sua contribuição ocorre ao estimular nos estudantes a apropriação dos processos de decomposição, abstração e reconhecimento de padrões, que constituem elementos do Pensamento Computacional.

Essa escolha e o planejamento do CEL-LOGIC (GITHUB, 2024) se deu após ampla revisão bibliográfica do tema, aprofundando conceitos de tecnologias digitais, Pensamento Computacional, processos históricos, econômicos e sociais, legislações e diretrizes de ensino nacionais e estaduais, além da experiência da pesquisadora com o tema em sala de aula.

O referencial teórico utilizado para a elaboração do CEL-LOGIC (GITHUB, 2024) também foi aprofundado. Oportunamente, a integração do Pensamento Computacional no Ensino Médio Integrado ao técnico, utilizando a plata-

forma MIT App Inventor (2022), representará uma abordagem inovadora e eficaz para preparar os estudantes para os desafios tecnológicos da sociedade contemporânea.

Ao proporcionar uma experiência prática na criação de aplicativos, os estudantes não apenas desenvolvem habilidades técnicas, mas fortalecem competências essenciais, como resolução de problemas, raciocínio lógico e criatividade.

O Guia Didático Interativo CEL-LOGIC (GITHUB, 2024) possibilita ao estudante entrar em contato com novos conhecimentos, por meio da Programação de Aplicativos Móveis, para construir a aprendizagem do PC.

REFERÊNCIAS

- BELETI JUNIOR, C. R.; FARIA SFORNI, M. S. de. Pesquisas experimentais no desenvolvimento do pensamento computacional: um mapeamento sistemático de literatura no ensino de conceitos de computação. **Educação em Foco**, [S. l.], v. 26, n. 49, 2023. Disponível em: <https://revista.uemg.br/index.php/educacaoemfoco/article/view/6623>. Acesso em: 29 jan. 2024.
- BERSSANETTE, J. H.; FRANCISCO, A. C. de. Metodologias ativas de aprendizagem no contexto de ensino – aprendizagem de programação de computadores: uma revisão sistemática da literatura. **Revista de Estudos e Pesquisas sobre Ensino Tecnológico**, v. 7, e159821, 2021. Disponível em: <https://sistemascmc.ifam.edu.br/educitec/index.php/educitec/article/view/1598/768>. Acesso em: 29 jan. 2024.
- CARDINALE, C. L. **A plataforma App Inventor e a construção do pensamento computacional no ensino médio integrado ao técnico**. 2024. 130 f. Dissertação (Mestrado em Docência para a Educação Básica) – Faculdade de Ciências, Programa de Pós-Graduação em Docência para a Educação Básica, Universidade Estadual Paulista “Júlio de Mesquita Filho” (UNESP), Bauru, SP, 2024.
- CARNEIRO, A. P.; FIGUEIREDO, I. S. de S.; LADEIRA, T. A. A importância das tecnologias digitais na Educação e seus desafios. **Revista Educação Pública**, v. 20, n. 35, set. 2020. Disponível em: <https://educacaopublica.cecierj.edu.br/artigos/20/35/joseph-a-importancia-das-tecnologias-digitais->

na-educacao-e-seus-desafios-a-educacao-na-era-da-informacao-e-da-cibercultura. Acesso em: 29 jan. 2024.

CARVALHO, F.; BRAGA, M. Pensamento computacional na educação brasileira: um olhar segundo artigos do Congresso Brasileiro de Informática na Educação. **Revista Brasileira de Informática na Educação**, [S. l.], v. 30, p. 237-261, 2022. Disponível em: <https://sol.sbc.org.br/journals/index.php/rbie/article/view/2649>. Acesso em: 29 jan. 2024.

COSTA, R. G.; PIEDADE, J. M. N. Uso do aplicativo MIT app inventor na aprendizagem de programação: uma revisão sistemática da literatura entre 2011 e 2020. **Revista Intersaberes**, [S. l.], v. 16, n. 37, p. 160-177, 2021. Disponível em: <https://www.revistasuninter.com/intersaberes/index.php/revista/article/view/2120>. Acesso em: 29 jan. 2024.

FARIA, B. L. de; ALVES, J. N.; MOREIRA COSTA, C. Uma experiência de inserção de Linguagem de Programação no Ensino Médio Integrado à Educação Profissional Técnica a partir da flexibilização curricular. **Revista Labor**, [S. l.], v. 1, n. 25, p. 242-260, 2021. Disponível em: <http://periodicos.ufc.br/labor/article/view/44110>. Acesso em: 28 jan. 2024.

FARIAS, C. B. de A. Análise comparativa sobre habilidades do pensamento computacional com alunos do ensino médio. **Revista online de Extensão e Cultura - Realização**, [S. l.], v. 8, n. 15, p. 27-40, 2021. Disponível em: <https://ojs.ufgd.edu.br/index.php/realizacao/article/view/12596>. Acesso em: 29 jan. 2024

FERRARINI, R.; SAHEB, D.; TORRES, P. L. Metodologias ativas e tecnologias digitais: aproximações e distinções. **Revista Educação em Questão**, Natal, v. 57, n. 52, e15762, abr. 2019. Disponível em: http://educa.fcc.org.br/scielo.php?script=sci_arttext&pid=S0102-77352019000200010&lng=pt&nrm=iso. Acesso em: 28 jan. 2024.

FERREIRA, R.; BRAGA, M. Ensino de programação e as estratégias pedagógicas utilizadas no Brasil. **Revista Contexto & Educação**, [S. l.], v. 38, n. 120, p. e11377, 2023. Disponível em: <https://www.revistas.unijui.edu.br/index.php/contextoeducacao/article/view/11377>. Acesso em: 28 jan. 2024.

FERREIRA, R. C. **Uma visão sobre as estratégias do ensino de programação no Brasil como resposta às demandas da cibercultura**. 2017. 151 f. Dissertação (Mestrado em Ciência, Tecnologia e

Educação) – Curso de Programa de Pós-Graduação em Ciência, Tecnologia e Educação, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, Rio de Janeiro, 2017. Disponível em: https://sucupira.capes.gov.br/sucupira/public/consultas/coleta/trabalhoConclusao/viewTrabalhoConclusao.jsf?popup=true&id_trabalho=5052151. Acesso em: 02 nov. 2023.

FERREIRA, R. C.; DUARTE, S. Ensino de programação: trajetória histórico-social e os avanços na cultura digital do Brasil. **Revista Brasileira de Ensino de Ciência e Tecnologia**, Ponta Grossa, PR, v. 12, n. 1, p. 386-408, jan./abr. 2019. Disponível em: <https://periodicos.utfpr.edu.br/rbect/article/view/7532>. Acesso em: 29 jan. 2024.

FLICK, U. **Introdução à metodologia de pesquisa: um guia para iniciantes**. Porto Alegre: Penso, 2013.

GATTI, B. A. Possível reconfiguração dos modelos educacionais pós-pandemia. **Estudos Avançados**, [S. l.], v. 34, n. 100, p. 29-41, set./dez. 2020. Disponível em: <https://www.scielo.br/j/ea/a/7M6bwtNMyyv7BqzDfKHfQxfh/?lang=pt>. Acesso em: 02 nov. 2023.

GITHUB. **Guia Didático Interativo CEL-LOGIC**. Disponível em: <https://github.com/Guia-Didatico-Interativo-CEL-LOGIC/>. Acesso em: 30 dez. 2024.

GOMES, T. C. S.; MELO, J. C. B. de. App Inventor for Android: uma nova possibilidade para o ensino de lógica de programação. **Anais dos Workshops do Cbie 2013**, Recife, v. 1, n. 1, p. 620-629, nov. 2013. Disponível em: <http://milanesa.ime.usp.br/rbie/index.php/wcbie/article/view/2725>. Acesso em: 15 maio 2023.

KAMINSKI, M. R.; KLÜBER, T. E.; BOSCARIOLI, C. Pensamento computacional na Educação Básica: reflexões a partir do histórico da informática na educação brasileira. **Revista Brasileira de Informática na Educação**, [S. l.], v. 29, p. 604-633, 2021. Disponível em: <https://sol.sbc.org.br/journals/index.php/rbie/article/view/2970>. Acesso em: 29 jan. 2024.

LUDKE, M.; ANDRÉ, M. **Pesquisa em educação: abordagens qualitativas**. São Paulo: E.P.U, 1986.

MASSA, N. P.; OLIVEIRA, G. S. de; SANTOS, J. A. dos. O construcionismo de Seymour Papert e os computadores na educação. **Cadernos da Fucamp**, Monte Carmelo, MG, v. 21, n. 52, p. 110-122, ago. 2022. Disponível em: <https://revistas.fucamp.edu.br/index.php/cadernos/issue/view/174>. Acesso em: 15 dez. 2023.

MIT APP INVENTOR. **Explore o MIT App Inventor**. Disponível em: <https://appinventor.mit.edu/>. Acesso em: mar. 2022.

MORAES, R. P. de; COSTA, V. F. da; SCHOLZ, R. E. P. Mapeamento sistemático do ensino introdutório de programação nos ensinos técnico e superior no Brasil. **Revista Brasileira de Informática na Educação**, [S. l.], v. 30, p. 628-647, 2022. Disponível em: <https://sol.sbc.org.br/journals/index.php/rbie/article/view/2611>. Acesso em: 29 jan. 2024.

PAPERT, S. **A máquina das crianças**: repensando a escola na era da informática. Porto Alegre: Artes Médicas, 2008.

PAPERT, S. **LOGO**: computadores e educação. São Paulo: Brasiliense, 1988.

PIEDADE, E. S.; COSTA, R. G. Uso do aplicativo MIT App Inventor na aprendizagem de programação: uma revisão sistemática da literatura entre 2011 e 2020. **Revista Intersaberes**, Curitiba, v. 16, n. 37, p. 1-19, set./dez. 2021. Disponível em: <http://www.uninter.com/intersaberes/index.php/revista/article/view/1785>. Acesso em: 29 jan. 2024.

SANTOS, P. M. dos; LICO, A. L.; MOURA, L. Relatos de experiência didática baseada no ambiente app inventor aplicado no ensino médio. **Revista Científica do UBM**, Barra Mansa, RJ, v. 24, n. 46, p. 85-98, dez. 2022.

SILVA, E. L. da. MENEZES, E. M. **Metodologia da pesquisa e elaboração de dissertação**. 4. ed. rev. atual. Florianópolis: UFSC, 2005.

SILVA, J. E. F. da. **Kodular Componentes** – Volume 1 (Portuguese Edition). São Paulo: Kindle, 2021.

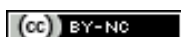
SOUZA, D. M. de; BATISTA, M. H. da S.; BARBOSA, E. F. Problemas e dificuldades no ensino e na aprendizagem de programação: um mapeamento sistemático. **Revista Brasileira de Informática na Educação**, [S. l.], v. 24, n. 1, p. 1-14, 2016. Disponível em: <https://doi.org/10.5753/rbie.2016.24.1.39>. Acesso em: 28 jan. 2024.

WING, J. Pensamento computacional: um conjunto de atitudes e habilidades, que todos, não só cientistas da computação, ficaram ansiosos para aprender a usar. **Revista Brasileira de Ensino de Ciência e Tecnologia**, [S. l.], v. 9, n. 2, p. 1-10, maio/ago. 2016. Disponível em: <https://periodicos.utfpr.edu.br/rbect/article/view/4711>. Acesso em: maio 2022.

ZABALA, A. A Prática Educativa: como ensinar. Porto Alegre: Artmed, 2010.

Recebido em: 17/01/2025

Aprovado em: 06/06/2025



Este é um artigo publicado em acesso aberto sob uma licença Creative Commons.