

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

GABRIEL SOARES COSTA

**DESENVOLVIMENTO DE UMA UNIDADE INSTRUCIONAL PARA O
ENSINO DE COMPUTAÇÃO UTILIZANDO O APP INVENTOR 2**

FLORIANÓPOLIS

2016

GABRIEL SOARES COSTA

**DESENVOLVIMENTO DE UMA UNIDADE INSTRUCIONAL PARA O
ENSINO DE COMPUTAÇÃO UTILIZANDO O APP INVENTOR 2**

Trabalho de Conclusão do Curso de Graduação em Sistemas de Informação, do Departamento de Informática e Estatística, do Centro Tecnológico da Universidade Federal de Santa Catarina, como parte dos requisitos necessários para a obtenção do título de Bacharel em Sistemas de Informação. Sob a orientação do Professor Doutor Jean Carlo Rossa Hauck.

FLORIANÓPOLIS

2016

GABRIEL SOARES COSTA

**DESENVOLVIMENTO DE UMA UNIDADE INSTRUCIONAL PARA O
ENSINO DE COMPUTAÇÃO UTILIZANDO O APP INVENTOR 2**

Trabalho de Conclusão de Curso submetido ao Departamento de Informática e Estatística da Universidade Federal de Santa Catarina, como parte dos requisitos necessários para a obtenção do Grau de Bacharel em Sistemas de Informação.

Florianópolis, 12 de dezembro de 2016.

Orientador:

Prof. Dr. Jean Carlo R. Hauck
Orientador
Universidade Federal de Santa Catarina

Banca Examinadora:

Prof. Elder Rizzon Santos
Avaliador
Universidade Federal de Santa Catarina

Giani Petri
Avaliador
Universidade Federal de Santa Catarina

AGRADECIMENTOS

Ao meu orientador, Jean Carlo Rossa Hauck, pela paciência, sinceridade e, principalmente, por toda a confiança depositada em mim.

Aos professores Elder Rizzon Santos e Giani Petri por aceitarem fazer parte da banca examinadora deste trabalho.

Aos avaliadores da Unidade Instrucional, que dedicaram seu tempo para colaborar com a realização deste trabalho.

À minha mãe, meu pai e toda a minha família, por me incentivarem e me apoiarem sempre.

À todos os amigos, os quais tornam minha vida mais alegre e acreditam na minha capacidade intelectual mais do que eu mesmo.

À minha *coach*, Ana Paula Platt, pelo excelente trabalho realizado comigo, clareando os caminhos da minha vida pessoal e profissional.

E à todos que, de alguma maneira, participaram positivamente da minha formação acadêmica.

RESUMO

A evolução tecnológica no mundo está cada vez mais presente no dia a dia da sociedade. Nesse contexto é possível observar que se tornou bastante necessário possuir algum conhecimento computacional. O problema é que atualmente são raras as escolas que possuem na grade curricular aulas voltadas para a formação do pensamento computacional, como programação, ou que disponibilizam cursos nessa área.

Com essa situação, o projeto da criação de uma unidade instrucional para o ensino de computação utilizando o App Inventor 2 (uma ferramenta Web construída pelo Google em parceria com o *Massachusetts Institute of Technology* que permite a criação de aplicativos para celulares que utilizam o sistema operacional Android) almeja um maior desenvolvimento do pensamento computacional no ensino fundamental e médio.

Para atingir o objetivo do projeto foi elaborada, desenvolvida e disponibilizada uma unidade instrucional composta por roteiros, *slides* de aula, questionários e avaliações relacionadas à criação de um aplicativo através do App Inventor 2 e sua integração com computação física. A avaliação do projeto foi realizada por professores de informática e robótica do ensino fundamental e médio de escolas localizadas em Santa Catarina.

Palavras-chave: Ensino, Aprendizagem; App Inventor 2; Computação Física.

ABSTRACT

The world technology evolution is even more present in the routine of society. In this context, it is possible to observe that it has become quite necessary to have some computational knowledge. The problem is that currently are rare schools that have classes aimed to form computational thinking, as programming, or that offer courses in this area.

In this situation, the creation of an instructional unit for computer education using App Inventor 2 (a web tool built by Google in partnership with the Massachusetts Institute of Technology that enables the creation of applications for mobile phones using Android operating system) aims to increase the development of computational thinking at elementary and high schools.

To achieve the project's goal it was prepared, developed and made available an instructional unit consisting of guides, lecture slides, questionnaires and evaluations related to the creation of an app through the App Inventor 2 and its integration with physical computing. The project evaluation was performed by IT and robotics teachers from elementary and high schools located at Santa Catarina.

Keywords: Teaching, Learning; App Inventor 2; Physical Computing.

LISTA DE FIGURAS

Figura 1: Tela do jogo Program Your Robot (KAZIMOGLU et al., 2012).....	28
Figura 2: Modelo ADDIE (DOUGLAS, 2014).....	30
Figura 3: Tela Designer do App Inventor 2. (MIT, 2015, adaptado)	36
Figura 4: Tela Blocks do App Inventor 2. (MIT, 2015, adaptado)	37
Figura 5: Telas do aiStarter. (MIT, 2013, adaptado).....	37
Figura 6: Boneco utilizado em oficinas de computação física. (INCoD, 2014)	39
Figura 7: Percentual de pessoas que tinham telefone móvel celular, na população de estudantes de 10 anos ou mais de idade, segundo a condição de estudante e a rede de ensino que frequentavam. (IBGE, 2014, adaptado)	56
Figura 8: Parte do Informativo da Unidade Instrucional.	63
Figura 9: Tela do Aplicativo “Controle do Robô”.....	64
Figura 10: Parte do Roteiro de Acesso.....	65
Figura 11: Parte do Roteiro de Aula/Criação do Aplicativo.....	66
Figura 12: Parte do Roteiro de Exportação.	66
Figura 13: Exemplos de Slides da Unidade Instrucional.	67
Figura 14: Parte da Prova/Avaliação.....	68
Figura 15: Parte da Tarefa de Casa.....	68
Figura 16: Parte da Rubrica de Avaliação.....	69
Figura 17: Parte do Questionário de Avaliação.....	73
Figura 18: Os objetivos de aprendizagem podem ser atingidos utilizando a unidade instrucional.	75
Figura 19: O material da unidade instrucional é completo.....	76
Figura 20: Qual o nível de dificuldade que os alunos do ensino fundamental (entre o sexto e nono ano) teriam em usar e entender o conteúdo da unidade instrucional? 77	77
Figura 21: Qual o nível de dificuldade que os alunos do ensino médio teriam em usar e entender o conteúdo da unidade instrucional?.....	78
Figura 22: Qual o nível de dificuldade que o professor teria em aplicar a unidade instrucional em sala de aula?	79
Figura 23: Os alunos do ensino fundamental (entre o sexto e nono ano) se interessariam em aprender computação utilizando essa unidade instrucional?	80
Figura 24: Os alunos do ensino médio se interessariam em aprender computação utilizando essa unidade instrucional?	81

Figura 25: Os professores se interessariam em utilizar essa unidade instrucional para o ensino de computação?82

LISTA DE QUADROS

Quadro 1: Objetivos de aprendizagem para o Nível 2 (CSTA, 2011).....	24
Quadro 2: Atividades do jogo associadas às características de pensamento computacional. (KAZIMOGLU et al., 2012, adaptado)	29
Quadro 3: Fases detalhadas do modelo ADDIE (Filatro, 2004)	34
Quadro 4: Termos definidos para a busca.	42
Quadro 5: Termos utilizados nas buscas.	43
Quadro 6: Resultados da pesquisa em inglês.	45
Quadro 7: Resultados da pesquisa em português.	45
Quadro 8: Trabalhos identificados após aplicação dos critérios de inclusão/exclusão e critérios de qualidade.	46
Quadro 9: Ensino de Programação para Alunas de Ensino Médio: Relato de uma Experiência (AVILA, et al. 2015)	48
Quadro 10: Lógica de Programação: Iniciação Lúdica com App Inventor for Android (GOMES & MELO, 2013)	49
Quadro 11: Using App inventor & history as a gateway to engage African American students in computer science (JIMENEZ & GARDNER-MCCUNE, 2015)	50
Quadro 12: Teaching Computational Thinking by Playing Games and Building Robots (ROSCOE, et al. 2014).....	51
Quadro 13: Objetivos de Aprendizagem da Unidade Instrucional, seguindo o currículo referência CSTA/ACM K-12 para o nível 2.....	58
Quadro 14: Plano de Ensino da Unidade Instrucional.....	61
Quadro 15: Componentes de Design utilizados na criação do aplicativo.	64
Quadro 16: Perguntas e medidas relacionadas aos objetivos de medição.	73

LISTA DE ABREVIATURAS E SIGLAS

ACM – *Association for Computing Machinery.*

ADDIE – *Analysis, Design, Development, Implementation, Evaluation.*

AI – App Inventor.

APP – Aplicativo móvel.

CSTA – *Computer Science Teachers Association.*

GQM – *Goal Question Metric.*

IEEE – Instituto de Engenheiros Eletricistas e Eletrônicos.

LED – *Light Emitting Diode.*

MIT – *Massachusetts Institute of Technology.*

TCC – Trabalho de conclusão de curso.

TI – Tecnologia da informação.

UFSC – Universidade Federal de Santa Catarina.

UI – Unidade Instrucional.

USB – *Universal Serial Bus.*

SUMÁRIO

1. INTRODUÇÃO.....	13
1.1. Objetivos	15
1.1.1. Objetivo Geral.....	15
1.1.2. Objetivos Específicos	15
1.2. Metodologia.....	15
1.3. Estrutura do Trabalho.....	18
2. FUNDAMENTAÇÃO TEÓRICA.....	19
2.1. Aprendizagem e Ensino	19
2.1.1 Ensino do Pensamento Computacional	25
2.1.2 Design Instrucional	30
2.1.3. Uma Ferramenta Para Ensino: App Inventor 2.....	35
2.2. Computação Física	38
3. ESTADO DA ARTE.....	41
3.1. Definição da Revisão	41
3.1.1 Critérios de Inclusão e Exclusão.....	44
3.1.2 Critérios de Qualidade	44
3.2. Execução da Busca.....	45
3.3. Extração e Análise dos Resultados.....	47
3.4 Discussão.....	52
3.4.1 Ameaças à Validade da Revisão	53
4. UNIDADE INSTRUCIONAL PARA O ENSINO DE COMPUTAÇÃO	55
4.1 Análise do Contexto	55
4.2 Design da Unidade Instrucional.....	57
4.3 Desenvolvimento da Unidade Instrucional	62
4.3.1 Informativo da UI	62
4.3.2 Aplicativo – Controle do Robô	63

4.3.3	Roteiros para Professores	65
4.3.4	Slides de Aula.....	67
4.3.5	Avaliação	67
4.3.6	Tarefa de Casa	68
4.3.7	Rubrica de Avaliação.....	69
5.	AVALIAÇÃO DA UNIDADE INSTRUCIONAL.....	70
5.1	Definição da Avaliação.....	70
5.2	Aplicação da Avaliação	73
5.3	Análise da Avaliação	74
5.3.1	Conteúdo da UI.....	75
5.3.2	Dificuldade em Aplicar a UI.....	77
5.3.3	Interesse pela UI.....	80
5.3.4	Ameaças à Validade.....	82
6.	CONCLUSÃO	84
	APÊNDICE I – Artigo do TCC	86
	REFERÊNCIAS	93

1. INTRODUÇÃO

A evolução tecnológica mundial influencia cada vez mais no dia a dia das pessoas; a tecnologia da informação alterou as formas, processos e nosso estilo de vida (SOUSA, 2010). Um bom exemplo da rapidez dessa evolução é o fato de que o número de celulares no Brasil já ultrapassou o número de habitantes no país (ANATEL, 2013).

Esse crescimento no uso e evolução da tecnologia na sociedade vem promovendo diferentes mudanças no âmbito geral. Entre essas mudanças está a criação de novos equipamentos, *hardwares* e *softwares*, que necessitam de software contendo algoritmos cada vez mais aprimorados e complexos.

Apesar dessa crescente demanda, um estudo divulgado em 2015 (ABDUL- AIM, 2015) apresenta que, apesar do crescente número de empregos criados diariamente na área de ciências da computação, uma porção significativa de estudantes do ensino médio não saem preparados para entrar nesse mercado, pois não possuem aulas de computação em suas escolas.

Isso indica que não são muitas as escolas que ensinam programação computacional em seus períodos letivos, mesmo já tendo sido verificado que esse conhecimento auxilia também em outros aspectos do nosso cotidiano, como raciocínio lógico, inovação e criatividade (CONFAP, 2014).

Segundo Gomes e Melo (2013, pp 1-2):

“O processo de ensino-aprendizagem dos conteúdos de lógica de programação apresenta algumas dificuldades a serem tratadas: a falta de motivação dos estudantes por considerarem o assunto de difícil compreensão, exigindo em alguns casos um nível de abstração na solução dos problemas que os mesmos ainda não desenvolveram, aliados às

metodologias comumente utilizadas, as quais não consideram as transformações na aquisição do conhecimento dos novos aprendizes”.

De fato, desde o fim do século passado até hoje, um grande número de diferentes abordagens para o desenvolvimento de software tem sido introduzido nos meios acadêmicos, dos quais apenas alguns sobreviveram (ABRAHAMSSON et al., 2002).

Nesse sentido, o App Inventor (Google e MIT, 2009) é um *software* de programação online que serve para o desenvolvimento de aplicativos para plataformas *mobile* que utilizam o Android (Google, 2008) como sistema operacional.

Ele possui um modelo de programação orientada à blocos *drag-and-drop* no qual é possível desenvolver programas apenas “arrastando e soltando blocos” de componentes (ABELSON, 2009) e atualmente ele está em sua segunda versão, liberada em 2014.

Experiências de cursos aplicando App Inventor, como o atualmente utilizado na Universidade de San Francisco, “Computação, Aplicativos Móveis e a Web” Wolber (2012), têm indicado que esse *software* é ideal para pessoas que nunca programaram e desejam desenvolver seus primeiros aplicativos. Sua interface gráfica auxilia os alunos que nunca tiveram contato com programação.

Desse modo, considerando a ausência de incentivos para o ensino e aprendizagem de programação, a elaboração de uma abordagem para o ensino de computação, através da criação de aplicativos para aparelhos móveis, voltada para a educação nos níveis fundamental e médio, se tornou uma opção atrativa e pertinente e foi a principal motivação para a realização desse projeto.

1.1. Objetivos

1.1.1. Objetivo Geral

Desenvolver e avaliar uma unidade instrucional para o ensino de computação utilizando o App Inventor 2 (MIT, 2013) integrado à computação física.

1.1.2. Objetivos Específicos

Visando atingir o objetivo geral do projeto, são necessários alguns objetivos específicos; são eles:

- Analisar as metodologias de ensino e aprendizagem, principalmente focando na área de computação e utilizá-las como base para o desenvolvimento do projeto.
- Desenvolver a integração do App com computação física.
- Desenvolver uma unidade instrucional, contendo roteiros, *slides* e imagens, voltados para alunos e professores, que auxiliariam no desenvolvimento programático dos aplicativos.
- Avaliar a unidade instrucional.

1.2. Metodologia

O projeto é desenvolvido em uma série de pesquisas sobre o estado da arte e é baseado no aspecto de ensino e aprendizagem de computação na escola. O desenvolvimento da unidade instrucional segue as etapas:

Etapa 1 – Nesta etapa o foco do trabalho é na fundamentação teórica do projeto. Serão analisados tópicos referentes às metodologias de aprendizagem e ensino. Além disso, um estudo sobre as atuações e funcionalidades do App Inventor.

Por fim, será realizada uma breve abordagem sobre computação física. As atividades realizadas são:

A1.1 – Análise da literatura com foco nas teorias referentes ao ensino e aprendizagem.

A1.2 – Análise do sistema App Inventor 2 e suas funcionalidades.

A1.3 – Estudo breve sobre computação física com foco no Arduino e seu funcionamento com o protocolo Firmata.

Etapa 2 – A realização de uma revisão sistemática do estado da arte e prática ocorre nessa etapa, seguindo o método estabelecido por Kitchenham (2004). As atividades são:

A2.1 – Definição da revisão sistemática (termos de busca, critérios de inclusão e exclusão, etc.).

A2.2 – Execução da busca pelo estado da arte.

A2.3 – Extração e análise das informações.

Etapa 3 – Desenvolvimento da unidade instrucional contendo roteiros, *slides*, informativo e questionários para alunos e professores. A unidade conterá material para abordar diferentes aspectos do pensamento computacional de crianças e adolescentes, utilizando o App Inventor 2 e integração com computação física.

Para a elaboração dessa etapa será usada uma abordagem de Design Instrucional, que é o processo reflexivo e sistemático de utilizar princípios e técnicas de cognição e aprendizagem para o planejamento de aulas, materiais didáticos, atividades extras, fontes de informação e processos de avaliação eficientes (SMITH & RAGAN, 1999), ou seja, tudo que envolve planejar, desenvolver e aplicar métodos e técnicas em ambientes educacionais (FREIRE, 2009). As atividades do processo de

Design Instrucional ocorrerão em quatro fases (e, por fim, a avaliação) com base no modelo ADDIE (MOLEND, 2003), sigla para *Analysis, Design, Development, Implementation, Evaluation*:

“**Análise**: envolve a identificação de necessidades de aprendizagem, a definição de objetivos instrucionais e o levantamento das restrições envolvidas; **design e desenvolvimento**: quando ocorre o planejamento da instrução e a elaboração dos materiais e produtos instrucionais; **implementação**: quando se dá a capacitação e ambientação de docentes e alunos à proposta de design instrucional e a realização do evento ou situação de ensino-aprendizagem propriamente ditos; [...] **avaliação**: envolve o acompanhamento, a revisão e a manutenção do sistema proposto” (FILATRO, 2007, p 12, grifo do autor).

Sendo assim, as atividades são:

A3.1 – Análise das necessidades de aprendizagem.

A3.2 – Design e desenvolvimento da unidade instrucional.

A3.3 – Avaliação da unidade instrucional.

A avaliação da unidade instrucional, última fase do modelo ADDIE de design instrucional, será realizada na etapa 4 do projeto.

Etapa 4 – Nessa última etapa ocorre a avaliação da unidade instrucional, com o objetivo de analisar a qualidade do conteúdo da unidade instrucional e se o conteúdo é ideal para a aplicação em ensino fundamental/médio. As atividades realizadas nesta etapa são:

A4.1 – Definição dos objetivos de medição utilizando a abordagem GQM (BASILI et al., 1994). A utilização desta abordagem auxilia na definição das metas de medição e na derivação destas metas para a realização da avaliação.

A4.2 – Elaboração de um questionário para avaliação da unidade instrucional aplicado com professores da área de informática de escolas que abrangem o ensino fundamental e médio.

A4.3 – Aplicação do questionário.

A4.4 – Análise dos resultados obtidos.

1.3. Estrutura do Trabalho

O trabalho está distribuído em 6 capítulos, começando por esse primeiro que foi utilizado para realizar a introdução e apresentação dos objetivos e metodologias do projeto.

O capítulo 2 possui a fundamentação teórica de todo o trabalho, apresentando o App Inventor 2, tópicos referentes à ensino e aprendizagem, jogos educacionais e, por fim, uma breve abordagem sobre computação física.

O terceiro capítulo contém o estado atual da arte, obtido através da execução de pesquisas sobre temas relacionados a esse projeto.

O capítulo 4 contém a unidade instrucional e seus detalhes.

Dentro do quinto capítulo está descrito como foi feita avaliação e a análise da unidade instrucional.

No capítulo 6 são apresentadas as considerações finais do presente trabalho e os trabalhos futuros relacionados.

2. FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são abordados tópicos referentes aos processos de aprendizagem e ensino em geral. Em seguida é realizada uma explanação sobre o software App Inventor 2, mantido pelo MIT. E, por fim, são apresentados conceitos gerais sobre computação física.

2.1. Aprendizagem e Ensino

Os conceitos em torno das teorias de ensino e aprendizagem possuem uma relação muito próxima; o educador possui o papel de ensinar o conhecimento que possui, enquanto o educando busca obter esses novos conteúdos (LINO, 2007). Segundo Dos Santos (2001, pp 71-72) “o ensino consiste na resposta planejada às exigências naturais do processo de aprendizagem”. É possível ver o ensino como a relação resultante de um professor (educador) com um aluno (educando).

“O segredo do bom ensino é o entusiasmo pessoal do professor, que vem do seu amor à ciência e aos alunos. Esse entusiasmo pode e deve ser canalizado, mediante planejamento e metodologia adequados, sobretudo para o estímulo ao entusiasmo dos alunos pela realização, por iniciativa própria, dos esforços intelectuais e morais que a aprendizagem exige” (DOS SANTOS, 2001, p 80).

Em paralelo, o conceito de aprender não é exatamente o mesmo de ensinar. Diferentemente de ensino, quando se fala de aprendizagem o aluno é quem passa a ser o foco, o atuante principal. O papel do professor nesse ponto é de facilitar a aprendizagem, não se preocupando em ensinar, mas auxiliar o aluno em seu entusiasmo (DOS SANTOS, 2001).

Se tratando da aprendizagem do ensino básico relacionado à computação, internacionalmente existem currículos que abordam este assunto de maneira bastante

ampla. Entre estes currículos está o currículo de referência CSTA/ACM K-12 (CSTA, 2011). Ele é definido em três níveis diferentes: nível 1 (do infantil até o sexto ano), nível 2 (do infantil ao nono ano) e o nível 3 (ensino médio). Como este projeto possui o objetivo de abranger crianças e adolescentes entre o nível fundamental e médio, o foco acaba sendo o nível 2 do currículo. A seguir (quadro 1) estão detalhados os objetivos de aprendizagem do nível 2 utilizados como base na criação da unidade instrucional.

Objetivos de aprendizagem Nível 2 (CSTA, 2011)	
	O aluno será capaz de:
Pensamento Computacional	<p>[O1] Usar os passos básicos de algoritmos para a resolução de problemas ao projetar soluções (p.ex., declaração e exploração do problema, examinação de exemplos, design, implementação de uma solução, testes, avaliação).</p> <p>[O2] Descrever o processo de paralelização na forma que se refere à resolução de problemas.</p> <p>[O3] Definir um algoritmo, como sendo uma sequência de instruções que podem ser processadas por um computador.</p> <p>[O4] Avaliar formas em que algoritmos diferentes podem ser utilizados para resolver o mesmo problema.</p> <p>[O5] Dramatizar algoritmos de busca e ordenação.</p> <p>[O6] Descrever e analisar uma sequência de instruções a ser seguida (p.ex., descrever o comportamento de um personagem em um vídeo game, dirigido por regras e algoritmos).</p> <p>[O7] Representar dados em maneiras diferentes, incluindo texto, sons, imagens e números.</p>

	<p>[O8] Usar representações visuais de estados de problema, estruturas e dados. (p.ex., gráficos, tabelas, diagramas de rede, fluxogramas).</p> <p>[O9] Interagir com modelos específicos de conteúdo e simulações (p.ex., ecossistemas, epidemias, dinâmica molecular) para apoiar a aprendizagem e pesquisa.</p> <p>[O10] Avaliar que tipos de problemas podem ser resolvidos usando modelagem e simulação.</p> <p>[O11] Analisar o grau em que um modelo de computador representa, com precisão, o mundo real.</p> <p>[O12] Fazer uso da abstração para decompor um problema em subproblemas.</p> <p>[O13] Compreender a noção de hierarquia e abstração em computação, incluindo linguagens de alto-nível, tradução (p.ex., interpretar o mesmo problema de diferentes modos), conjunto de instruções, e circuitos lógicos.</p> <p>[O14] Examinar conexões entre elementos da matemática e ciência da computação, incluindo números binários, lógica, conjuntos e funções.</p> <p>[O15] Fornecer exemplos de aplicações interdisciplinares do pensamento computacional.</p>
Colaboração	<p>[O16] Aplicar ferramentas e periféricos de produtividade/multimídia para colaboração em grupo e apoiar a aprendizagem ao longo do currículo.</p>

	<p>[O17] Colaborativamente criar, desenvolver, publicar e apresentar produtos (p.ex., vídeos, podcasts, sites), utilizando recursos tecnológicos que demonstram e comunicam conceitos do currículo.</p> <p>[O18] Colaborar com colegas, especialistas e outros utilizando práticas colaborativas como programação em pares, trabalho em equipes de projeto, e participação em atividades de aprendizagem ativa em grupo.</p> <p>[O19] Exibir disposições necessárias para colaboração: fornecer feedback útil e integrante, compreender e aceitar múltiplas perspectivas, socialização.</p>
<p>Práticas Computacionais e de Programação</p>	<p>[O20] Selecionar ferramentas e recursos tecnológicos apropriados para realizar tarefas variadas e resolver problemas.</p> <p>[O21] Usar uma variedade de ferramentas e periféricos de multimídia para apoiar a produtividade e aprendizagem pessoal durante todo o currículo.</p> <p>[O22] Conceber, desenvolver, publicar e apresentar produtos (p.ex., páginas web, aplicações móveis, animações) usando recursos de tecnologia que demonstram e comunicam os conceitos do currículo.</p> <p>[O23] Demonstrar uma compreensão de algoritmos e a sua aplicação prática.</p> <p>[O24] Implementar soluções de problema utilizando uma linguagem de programação, incluindo: o comportamento de laços</p>

	<p>(sequências de instruções que se repetem), instruções condicionais, lógica, expressões, variáveis e funções.</p> <p>[O25] Demonstrar boas práticas na segurança da informação pessoal, usando senhas, encriptação e transações seguras.</p> <p>[O26] Identificar carreiras interdisciplinares que são abrangidas pela ciência da computação.</p> <p>[O27] Demonstrar receptiva disposição para resolver e programar problemas indeterminados (p.ex., conforto com complexidade, persistência, brainstorming, adaptabilidade, paciência, tendência a mexer, criatividade, aceitação de mudanças).</p> <p>[O28] Coletar e analisar dados que correspondem à saída de múltiplas execuções de um programa de computador.</p>
<p>Computadores e Dispositivos de Comunicação</p>	<p>[O29] Reconhecer que os computadores são equipamentos que executam programas.</p> <p>[O30] Identificar uma variedade de dispositivos eletrônicos que contêm processadores computacionais.</p> <p>[O31] Demonstrar compreensão sobre a relação entre hardware e software.</p> <p>[O32] Usar terminologia adequada ao desenvolvimento e, precisa na comunicação sobre tecnologia.</p> <p>[O33] Aplicar estratégias para identificar e resolver problemas de rotina de hardware que ocorrem no uso de computador diariamente.</p> <p>[O34] Descrever os principais componentes e funções de sistemas de computadores e redes.</p>

	<p>[O35] Descrever o que distingue os seres humanos de máquinas, dando um enfoque na inteligência humana contra a inteligência de máquina e, formas que podemos nos comunicar.</p> <p>[O36] Descrever maneiras em que os computadores usam modelos de comportamento inteligente (p.ex., movimento de robô, fala e compreensão da linguagem e, visão computacional).</p>
<p>Impactos Éticos, Globais e na Comunidade</p>	<p>[O37] Apresentar comportamentos legais e éticos no uso de informação e tecnologia e, discutir as consequências do uso indevido.</p> <p>[O38] Demonstrar conhecimento das mudanças nas tecnologias de informação ao longo do tempo e os efeitos destas mudanças na educação, no local de trabalho e na sociedade.</p> <p>[O39] Analisar os impactos positivos e negativos da computação na cultura humana.</p> <p>[O40] Avaliar a precisão, relevância, adequação, abrangência, e viés de fontes de informações eletrônicas referentes a problemas do mundo real.</p> <p>[O41] Discutir como a distribuição desigual de recursos de computação em uma economia global levanta questões de equidade, acesso e poder.</p>

Quadro 1: Objetivos de aprendizagem para o Nível 2 (CSTA, 2011)

Segundo Gonçalves (1993) uma unidade instrucional pode ser uma aula, um livro, um guia, uma disciplina. Todo evento, ação ou material que possa ser utilizado e inserido no meio educacional, como um curso, por exemplo, pode ser considerado uma unidade instrucional.

As etapas de elaboração, planejamento, desenvolvimento, aplicação e avaliação dessas unidades instrucionais são abordadas pelo design instrucional (GONÇALVES, 1993), e utilizadas no ensino do pensamento computacional (WANGENHEIM et al., 2014).

Muitas definições abrangentes podem ser ditas sobre 'pensamento computacional', mas a maioria dos estudos defendem que tornar esse pensamento acessível para todos é crucial para que alunos possam desenvolver rapidamente múltiplas habilidades computacionais antes de entrarem em contato com a programação formal (KAZIMOGLU et al., 2012). O pensamento computacional não se prende apenas a programação, mas sim a contextualização. Não é uma habilidade, mas um fundamento. Uma maneira de humanos pensarem, e não computadores. Ideias, não artefatos (WING, 2006).

2.1.1 Ensino do Pensamento Computacional

Na ciência da computação, é essencial identificar conceitos com definições e funcionalidades precisas. Entender a terminologia precisa usada nessa ciência é necessário para entender problemas e possibilitar o desenvolvimento de soluções eficazes na sociedade; e isso se torna mais significativo quando é levado em consideração que o pensamento computacional cobre uma variação grande de diferentes habilidades comportamentais (KAZIMOGLU et al., 2012).

O pensamento computacional pode ser definido como o processo de raciocínios envolvidos na formulação de problemas e suas respectivas soluções, de uma maneira que essas soluções possam ser representadas de uma forma efetiva de ser apresentada por algum agente (GUZDIAL, 2008).

Assumindo a criação de problemas e soluções como uma série de passos a serem seguidos, é possível citar o estudo de Perković et al. (2010), que descreve computação como a execução de algoritmos que passam por uma série de estágios diferentes até que o estágio final, com a solução, seja alcançado.

A perspectiva social do pensamento computacional é definida por Berland & Lee (2011), através de um jogo *multiplayer*, como a computação distribuída, onde diferentes peças de informação ou de lógica são compartilhadas por diferentes jogadores durante o processo de simulação ou criação de algoritmos.

Seguindo essa linha de raciocínio e considerando o estudo de Wing (2006), que aponta o *debug* (depuração) como componente essencial tanto para programação como para o desenvolvimento do pensamento computacional, é possível dividir o pensamento computacional em cinco fundamentos básicos (KAZIMOGLU et al., 2012):

- **Solução de Problemas:** se tratando de pensamento computacional, isso refere à solução de problemas com pensamentos lógicos utilizando modelos computacionais. Isso inclui aplicar decomposição de problemas para identificar outros problemas ou gerar representações alternativas dos mesmos. Nesse ponto, alunos classificam problemas e decidem se esses problemas podem ou não podem ser solucionados computacionalmente. Além disso, alunos são capazes de avaliar um problema e especificar critérios apropriados para desenvolver aplicações abstratas.
- **Construção de Algoritmos:** envolve a construção passo-a-passo de processos para a solução de um problema em particular. A seleção de técnicas de algoritmos apropriadas é uma fase crucial quando se trata

de pensarmos computacionalmente, pois isso desenvolve abstrações robustas o suficiente para serem reutilizadas na solução de problemas similares.

- **Debugging:** é a análise de problemas e erros na parte lógica ou nas atividades. Nesse estágio, estudantes recebem o *feedback* de seus algoritmos e avaliam eles, o que inclui revisar papéis e/ou estratégias utilizadas. *Debugging* é um ponto essencial tanto na programação quanto no pensamento computacional, pois envolve estratégias críticas do processo (WING, 2006).
- **Simulação:** também chamado de “modelo de criação”, é a demonstração de algoritmos e envolve designar e implementar modelos no computador, baseado em algoritmos criados. Na simulação, o estudante desenvolve ou testa modelos para tomar decisões sobre quais circunstâncias devem ser consideradas quando completam sua abstração.
- **Socialização:** se refere ao aspecto social do pensamento computacional, que envolve coordenação, cooperação e/ou competição durante os estágios de Solução de Problemas, Construção de Algoritmos, *Debugging* e Simulação. Essa característica do pensamento computacional permite o *brainstorming* (técnica utilizada para potencializar a criatividade de um indivíduo ou grupo) e incentiva a avaliação dos incidentes, bem como o desenvolver estratégias entre múltiplas partes. Esse aspecto é o que diferencia o pensamento computacional de programação tradicional (BERLAND & LEE, 2011).

Com essa perspectiva, Kazimoglu et al. (2012) elaborou um jogo pedagógico para alunos desenvolverem suas características relacionadas ao pensamento computacional. *Program Your Robot* (Figura 1) é um jogo onde o jogador controla um robô dando comandos para ele. O objetivo é fazê-lo chegar ao teletransportador utilizando o mínimo de passos possíveis. As atividades no jogo e suas respectivas tarefas estão apresentadas no quadro 2.



Figura 1: Tela do jogo *Program Your Robot* (KAZIMOGLU et al., 2012)

Tarefa	Categoria de “Pensamento Computacional” associada	Atividade no jogo
Identificação do problema e decomposição	Solução de Problemas	Ajudar o robô a alcançar o teletransportador. Ativar a luz do robô

		quando ele estiver em cima do equipamento.
Criação eficiente e repetível de padrões	Construção de Algoritmos	Criar um algoritmo de solução para completar todos os níveis utilizando o menor número de espaços possíveis. Utilizar funções para criar padrões repetíveis.
Praticar o modo <i>debug</i>	<i>Debugging</i>	Pressionar o botão de <i>debug</i> para monitorar o algoritmo de solução criado, com o objetivo de detectar qualquer erro potencial na lógica do algoritmo.
Praticar o modo <i>run-time</i>	Simulação	Observar os movimentos do robô durante a execução do modo <i>run-time</i> . Você consegue seguir os passos do seu algoritmo? Os comportamentos esperados do robô estão sendo seguidos?
<i>Brainstorming</i>	Socialização	Analise as estratégias vitoriosas criadas por outros jogadores. Compare suas soluções com as deles. Que conselho você daria para você mesmo e para eles com o intuito de aumentarem suas pontuações no jogo?

Quadro 2: Atividades do jogo associadas às características de pensamento computacional. (KAZIMOGLU et al., 2012, adaptado)

Com essa abordagem foi possível verificar que, além dos alunos terem gostado de realizar essas tarefas, eles puderam tratar diretamente com pensamentos considerados apenas “secundários” anteriormente e retornaram um feedback positivo para o estudo (KAZIMOGLU et al., 2012).

2.1.2 Design Instrucional

Segundo Sartori e Roesler (2005) o design instrucional pode ser definido como um processo de elaboração e criação de projetos nos ambientes de ensino e aprendizagem, além de sistemas tutoriais de apoio ao aluno, desenvolvidos para potencializar a aprendizagem de informações em contextos específicos.

O design instrucional também pode ser dito como o “planejamento do ensino-aprendizagem, incluindo atividades, estratégias, sistemas de avaliação, métodos e materiais instrucionais” (FILATRO, 2007, p 11).

Existem diferentes modelos elaborados de design instrucional, e um dos mais utilizados é o ADDIE (*Analysis, Design, Development, Implementation, Evaluation*) que possui cinco fases (MOLEND, 2013; AHMAD, 2013), representadas a seguir na figura 2:

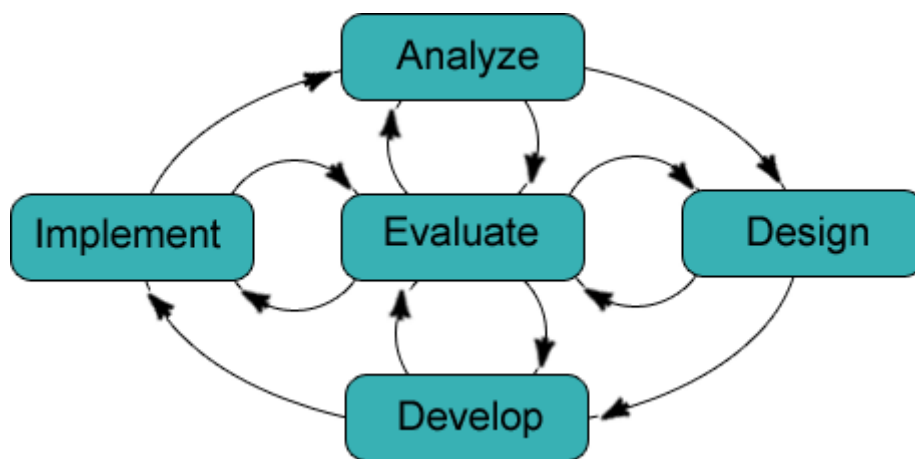


Figura 2: Modelo ADDIE (DOUGLAS, 2014)

- **Análise:** É a etapa inicial do processo. Na fase de análise, o problema instrucional fica esclarecido. Os objetivos são estabelecidos e o ambiente de

aprendizagem, assim como o conhecimento e as habilidades do aprendiz são identificadas.

- **Design:** Essa fase lida com os objetivos de aprendizagem, exercícios, conteúdos, planejamento de lições e seleção de mídia que devem estar presentes na unidade instrucional. A fase de design deve ser tanto sistemática como bem específica.
- **Desenvolvimento:** A etapa de desenvolvimento é onde os desenvolvedores e designers instrucionais criam e montam os conteúdos elaborados na fase de design. Nesse momento, os gráficos são desenhados, os conteúdos finais são escritos. Além disso, caso seja necessário, também é nessa etapa que programadores trabalham para desenvolver e integrar tecnologias. Nesta fase a unidade instrucional é criada.
- **Implementação:** Durante a fase de implementação, o procedimento de treinamento dos aprendizes é realizado. Esse treinamento deve englobar a aplicação da unidade instrucional, testando seus processos e testando suas funcionalidades.
- **Avaliação:** É uma fase constituída de duas partes chamadas de formativa e somativa, com o objetivo de verificar a eficiência, eficácia e efetividade da unidade instrucional. A avaliação formativa está presente em cada etapa do modelo ADDIE. Já a avaliação somativa consiste de testes específicos para a avaliação dos pontos positivos e negativos, além do recebimento do *feedback* feito pelos usuários que testaram a unidade instrucional. Devido à sua importância, geralmente é uma etapa considerada a principal de todo o processo.

Filatro (2004), na procura de facilitar a compreensão das fases do modelo tradicional de design instrucional, elabora questões específicas que devem ser respondidas para a realização de cada etapa, apresentadas no quadro 3. Cada conjunto de perguntas é relacionado à uma característica ou objetivo de cada fase.

Fase	Definições	Questões
Análise	Identificação de necessidades de aprendizagem	Qual é o problema para qual o design instrucional está sendo proposto? Qual é a origem do problema? Quais são as possíveis soluções?
	Definição de objetivos instrucionais	Que conhecimentos, habilidades e atitudes precisam ser ensinados? Qual e quanto conteúdo é necessário para a instrução? Em quanto tempo esse conteúdo será ensinado? Que métodos e técnicas são adequados à exploração desse conteúdo? De que forma a aprendizagem será avaliada?
	Caracterização dos alunos	O que já sabem? Quais são seus estilos e características de aprendizagem? O que precisam ou querem saber? Em que ambiente/situação aplicarão a aprendizagem?
	Levantamento das limitações	Qual é o orçamento disponível? De quantos profissionais dispomos? Quais são as restrições técnicas? Em quanto tempo precisamos alcançar os objetivos? Quais são os riscos envolvidos?
<i>Design e desenvolvimento</i>	Planejamento da instrução	Como os objetivos instrucionais serão alcançados? Que métodos e técnicas instrucionais melhor se ajustam a esses objetivos? Como o conteúdo é mapeado, estruturado e sequenciado?

		<p>Em que sequência a instrução deve ser apresentada?</p> <p>Quais são as mídias mais apropriadas para a apresentação do conteúdo?</p> <p>Que produtos e atividades instrucionais devem ser preparados e produzidos?</p>
	Produção de materiais e produtos	<p>Qual é o grau de interação entre os alunos, e entre os alunos e o professor, possibilitado pelas atividades instrucionais propostas?</p> <p>Qual é o design gráfico dos produtos instrucionais impressos e/ou eletrônicos?</p> <p>Qual é o grau de interatividade (interação com o material) proporcionado por esses produtos?</p> <p>Quais são os mecanismos de atualização e personalização dos materiais?</p> <p>Que níveis de suporte instrucional e tecnológico são oferecidos?</p>
Implementação	Capacitação	Os usuários (professores e alunos) precisam ser treinados para o uso dos materiais e aplicação das atividades?
	Ambientação	<p>Os usuários precisam ser matriculados ou cadastrados para ter acesso a determinados produtos ou ambientes?</p> <p>De quanto tempo necessitam para compreender o funcionamento do sistema e os pré-requisitos para acompanhar o design projetado?</p>
	Realização do evento ou da situação de ensino-aprendizagem	<p>Em que local e condições ocorre o evento ou a situação de ensino-aprendizagem (presencialmente, semipresencialmente, a distância, no ambiente de trabalho, em situação de laboratório, em ambientes virtuais)?</p> <p>Como se dá a organização social da aprendizagem (individualmente, em grupos fixos, em grupos voláteis)?</p>

		<p>Como os produtos instrucionais são manipulados por professores e alunos (sequencialmente, em módulos inter ou independentes)?</p> <p>Como a aprendizagem dos alunos é avaliada?</p> <p>Como se dá o feedback por parte do professor?</p>
Avaliação	Acompanhamento	<p>Como o design será avaliado (por observação, testes, feedback constante)?</p> <p>Quem fará essa avaliação (alunos e professores usuários, equipe de desenvolvimento, patrocinadores e mantenedores, software de monitoramento da aprendizagem)?</p> <p>Quais foram os resultados finais de aprendizagem (índices de aprovação, desistência, reprovação, abandono)?</p>
	Revisão	<p>Quais foram os problemas detectados na implementação?</p> <p>Que erros podem ser corrigidos?</p> <p>Em que medida o design instrucional pode ser aperfeiçoado?</p>
	Manutenção	<p>Que ações devem ser tomadas para possibilitar a continuidade do projeto ou novas edições?</p>

Quadro 3: Fases detalhadas do modelo ADDIE (Filatro, 2004)

Diversos trabalhos de design instrucional e seus modelos têm sido realizados para auxiliar o uso de ferramentas computacionais no apoio ao ensino do pensamento computacional (WANGENHEIM et al., 2014; FREIRE, 2009). Entre essas ferramentas podemos citar o App Inventor, criado pelo Google, baseado em estudos publicados pelo MIT (WOLBER, 2011), que surgiu como iniciativa de programação e criação de aplicativos tecnológicos para qualquer pessoa, em praticamente todos os níveis de escolaridade (HARDESTY, 2010).

2.1.3. Uma Ferramenta Para Ensino: App Inventor 2

Segundo Wolber (2011), um dos responsáveis pelo sistema, App Inventor (AI) é uma linguagem visual de “blocos” utilizada na criação de aplicativos para dispositivos móveis, parte de um programa piloto da Google. Também pode ser dito que o AI é uma ferramenta web que possui dois elementos principais – uma tela de design de componentes (*Designer*) e outra de edição dos blocos (*Block Editor*), que juntas permitem à usuários sem conhecimento de programação a chance de desenvolver aplicativos móveis. A parte mais poderosa do AI é a sua capacidade visual de programação, já que é possível “pegar, arrastar e soltar” uma variedade enorme de interfaces, componentes “invisíveis” e lógicos para a criação de aplicativos móveis completos (HSU et al., 2011).

Usuários do App Inventor não precisam escrever códigos de computação tradicionais; eles desenvolvem suas produções apenas arrastando componentes e elaborando sua parte lógica através da união de blocos coloridos já codificados (HARDESTY, 2010).

Através da tela *Designer* do AI (Figura 3), o usuário pode “desenhar” a interface do aplicativo e integrar componentes invisíveis (como acelerômetro ou sons). Já na tela *Block Editor* (Figura 4), o usuário pode programar os comportamentos do aplicativo e editar como ele deve reagir sobre certas circunstâncias (HSU et al., 2011). Até dezembro de 2013 era necessário baixar e instalar uma extensão do AI no computador para acessar a tela de edição de blocos, porém nessa mesma data o MIT lançou o App Inventor 2. Uma das principais melhorias dessa segunda versão foi a adição de uma nova aba na tela da ferramenta onde a edição dos blocos poderia ser feita na própria aplicação, removendo a necessidade de *download* da extensão (MIT, 2013).

Outra vantagem do programa é a disponibilização de um emulador para a realização de testes da sua aplicação, como visto na Figura 5. Ou seja, não é necessário possuir um telefone celular ou um *tablet* para testar as funcionalidades do seu projeto. Esse emulador é chamado de aiStarter e o que ele faz é simular o funcionamento de um *smartphone* na tela do seu computador (MIT, 2013). Para utilizá-lo é necessário baixar e instalar o *software* aiStarter na máquina utilizada para a realização de testes.

Para utilizar o AI é necessário possuir uma Conta do Google (Google, 2015). Além de ser uma aplicação *web*, o AI mantém todos os seus projetos armazenados em nuvem computacional, permitindo que você possa desenvolver ou baixar o aplicativo em qualquer lugar. Sendo assim, ele precisa filtrar quais projetos pertencem aos seus respectivos criadores. Para isso, ele utiliza o seu usuário do Google como chave de acesso ao sistema (MIT, 2013).

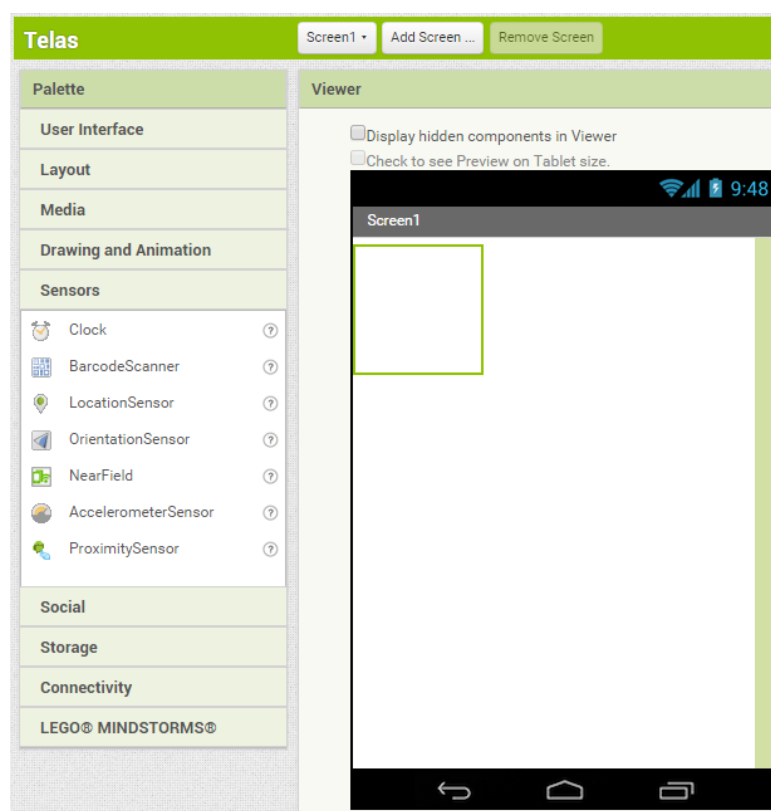


Figura 3: Tela Designer do App Inventor 2. (MIT, 2015, adaptado)

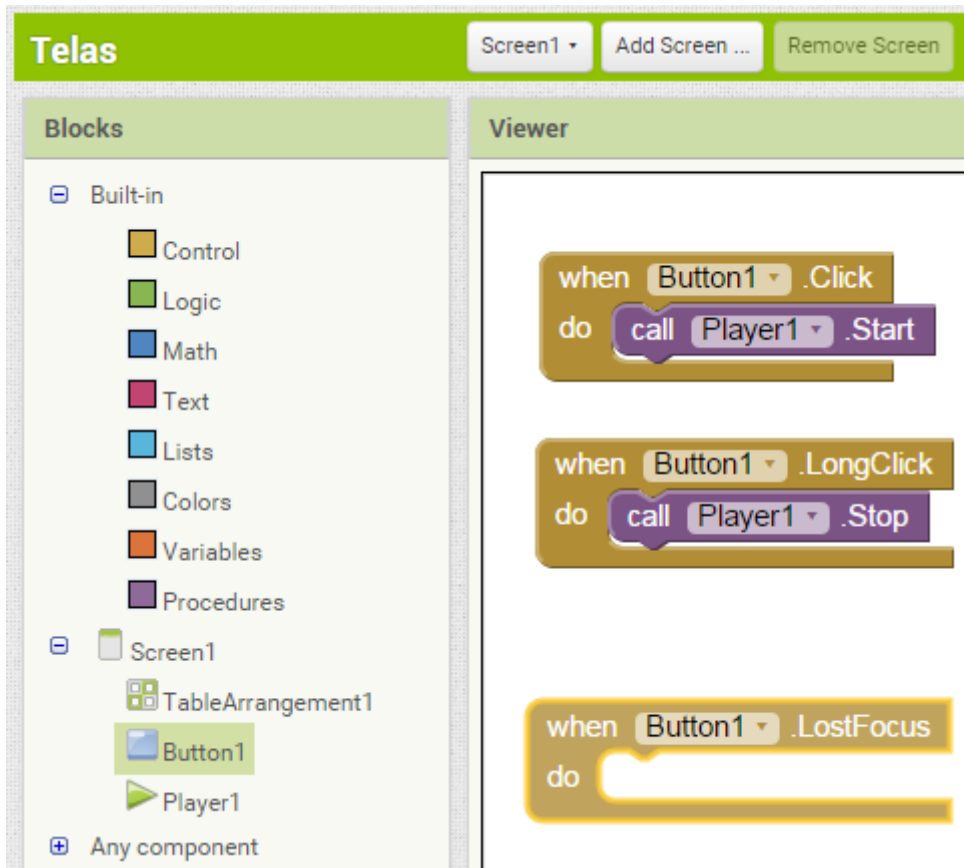


Figura 4: Tela Blocks do App Inventor 2. (MIT, 2015, adaptado)

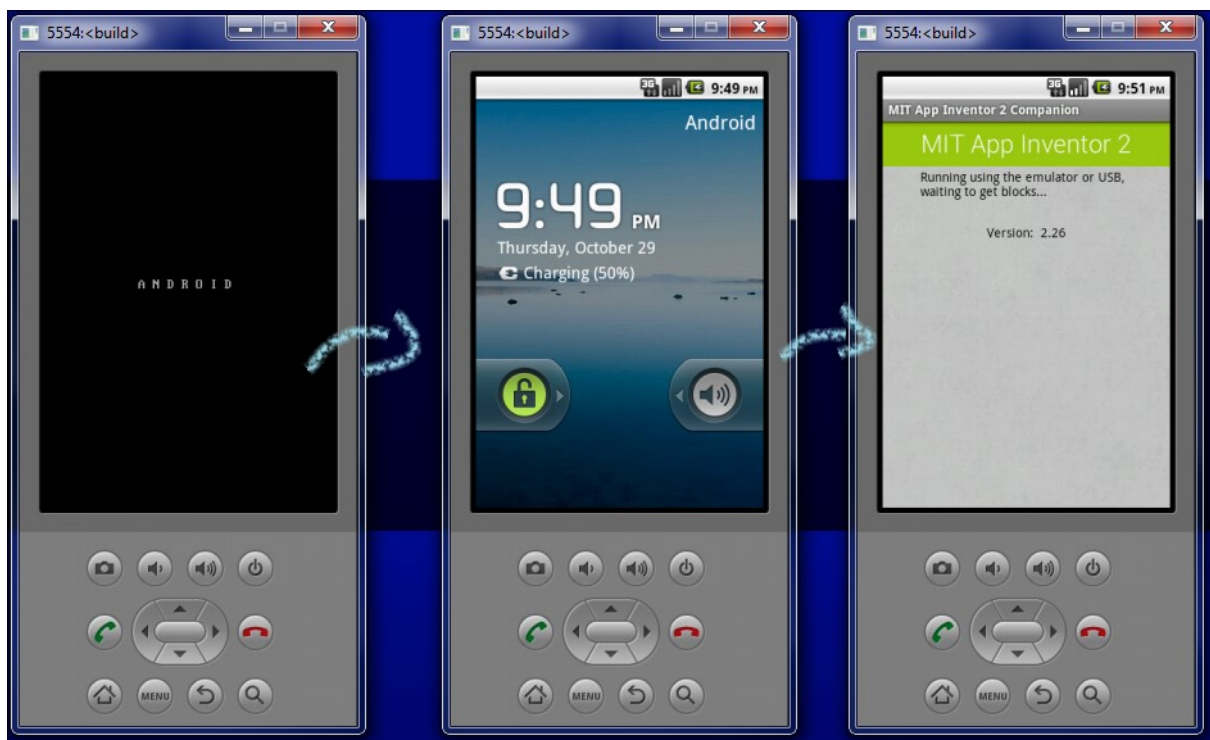


Figura 5: Telas do aiStarter. (MIT, 2013, adaptado)

Após a criação e o desenvolvimento do aplicativo, existem duas maneiras de compartilhá-lo externamente: a primeira é baixando a sua versão executável (que possui a extensão *.apk*), e que pode ser instalada em qualquer dispositivo com Android; e a segunda é realizando o *download* do código fonte (possui a extensão *.aia*). Esse código pode ser carregado no App Inventor, onde é possível fazer alterações no aplicativo (MIT, 2013).

De maneira geral, as interfaces e funcionalidades do AI permitem com que o desenvolvimento de aplicações que utilizam Bluetooth, sensores de geolocalização e interação, serviços web e entre outras funções popularmente utilizadas, seja realizado de maneira simplificada, não exigindo um conhecimento prévio e abrangente de programação computacional (GOMES e MELO, 2013).

Sendo assim, através de sua facilidade de uso, o App Inventor vem sendo utilizado para a criação de aplicativos que interagem com computação física (GOMES et al., 2014). O envio de comandos para placas de prototipagem físicas via aparelhos móveis através da tecnologia Bluetooth passou a ser mais simples após o desenvolvimento dessa plataforma (GOMES et al., 2014)

2.2. Computação Física

A computação física, de modo geral, envolve o projeto de mecanismos interativos que possuem a capacidade de comunicação com o ser humano através de diversos tipos de sensores existentes e atuadores controlados por uma ação ou comportamento estabelecido através de *software*, executado dentro de um microcontrolador – que pode ser tanto um pequeno computador, como um *chip* eletrônico (BANZI, 2011).

Um exemplo de dispositivo de computação física geralmente citado e utilizado é a plataforma Arduino. Segundo Banzi (2011), através do Arduino “um designer ou artista pode rapidamente aprender o básico de eletrônica e sensores e começar a criar seus protótipos com pouco investimento”.

O Arduino é uma plataforma composta por uma placa controladora (de microcontroladores) e um ambiente voltado para o desenvolvimento, ou seja, é uma combinação de *software* e *hardware* flexíveis, acessíveis e de fácil usabilidade, cuja primeira versão foi criada em 2005 e vem recebendo melhorias e novas versões constantemente (LEMOS, 2014).

Essa plataforma física possui uma maneabilidade bastante variada e pode ser utilizada para diversos objetivos diferentes. Podemos observar na figura 6 um robô desenvolvido em uma oficina de computação que utiliza o Arduino como computador central do dispositivo.

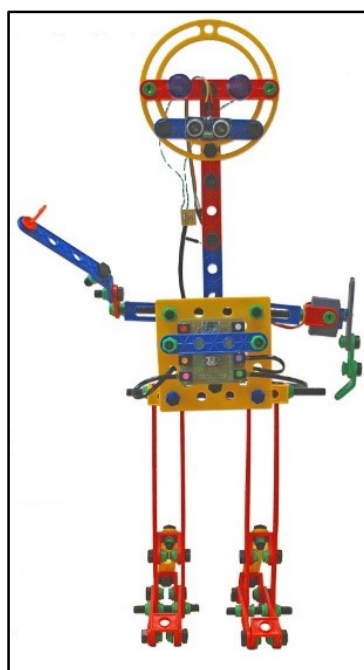


Figura 6: Boneco utilizado em oficinas de computação física. (INCoD, 2014)

Existem diversos *softwares*, *drivers* e protocolos para o Arduino, entre eles está o Firmata. Firmata é um *firmware* genérico utilizado para a comunicação entre microcontroladores de um *software* e um computador. O objetivo principal do Firmata é fazer com que o microcontrolador seja uma “extensão” do ambiente de programação do computador principal, de uma maneira de que a própria programação se torne o mais natural possível (STEINER, 2009).

Na teoria, o protocolo do Firmata poderia ser implementado em qualquer plataforma de microcontroladores. Porém, atualmente, a implementação mais completa é para Arduino e os microcontroladores compatíveis com o mesmo (FIRMATA, 2015).

3. ESTADO DA ARTE

Este capítulo possui como finalidade identificar o estado atual em que se encontram pesquisas relacionadas ao desenvolvimento de unidades instrucionais que utilizam o App Inventor para o ensino de computação nos ensinos fundamental e médio. A análise desse tópico foi realizada sistematicamente seguindo o método de revisão literária definido por Kitchenham (2004), adaptado para um trabalho de conclusão de curso de graduação.

3.1. Definição da Revisão

A pergunta de pesquisa definida para a realização da revisão sistemática:

Quais são e como são estruturadas as unidades instrucionais para ensinar pensamento computacional para alunos de ensino fundamental e médio utilizando o App Inventor?

As ferramentas utilizadas para a busca são: Google Scholar (scholar.google.com.br), ACM Digital Library (dl.acm.org) e IEEE Xplore (ieeexplore.ieee.org). A primeira ferramenta foi escolhida por, além de ser uma ferramenta de busca aberta, possuir resultados que abrangem diversas bibliotecas virtuais diferentes. A segunda e a terceira ferramentas foram definidas com o objetivo de angariar resultados mais específicos de publicações em eventos e periódicos científicos. Foram realizadas pesquisas tanto na língua portuguesa como na língua inglesa. Apenas artigos ou publicações divulgadas no período de 2010 (ano de *release* das primeiras versões do App Inventor) a 2015 foram considerados.

O Quadro 4 apresenta as palavras que, quando combinadas, formam os termos de busca utilizados para responder as perguntas de pesquisa.

Termo	Sinônimo	Tradução (Inglês)
Ensino	Instrução, Aprendizagem, Treinamento	<i>Education, Teaching, Learning, Training</i>
Ensino Médio	Ensino Secundário, Colegial	<i>Secondary School, High School, Senior School</i>
Ensino Fundamental	Ensino Primário, Ensino Básico	<i>Primary School, Middle School, Elementary School</i>
App Inventor	-	-
Computação	Ciência da computação	<i>Computing, Computer Science</i>

Quadro 4: Termos definidos para a busca.

Além dos termos apresentados, a sigla “K-12”, utilizada nos Estados Unidos (e em alguns outros países como Canadá e Coréia do Sul, por exemplo), com o objetivo de designar a educação primária (ensino fundamental) e educação secundária (ensino médio e ensino técnico) como um todo, também é utilizada nas buscas.

No Quadro 5 são apresentados os conjuntos de termos combinados, tanto em inglês como em português, para a realização da pesquisa e seus respectivos repositórios.

Repositório	Termos de busca (português)	Termos de busca (inglês)
Google Scholar	(ensino OR instrução OR aprendizagem OR treinamento) AND (“ensino médio” OR “ensino secundário” OR “colegial” OR “ensino fundamental” OR “ensino primário” OR “ensino básico”) AND (“App Inventor”) AND (“computação” OR “ciência da computação”) 2010..2015	<i>(education OR teaching OR learning OR training) AND (“secondary school” OR “high school” OR “senior school” OR “primary school” OR “middle school” OR “elementary school” OR “K-12”) AND (“App Inventor”) AND (“computing” OR “computer science”) 2010..2015</i>

ACM Digital Library	(ensino OR instrução OR aprendizagem OR treinamento) AND (“ensino médio” OR “ensino secundário” OR “colegial” OR “ensino fundamental” OR “ensino primário” OR “ensino básico”) AND (“App Inventor”) AND (“computação” OR “ciência da computação”) for: ((ensino OR instrução OR aprendizagem OR treinamento) AND (“ensino médio” OR “ensino secundário” OR “colegial” OR “ensino fundamental” OR “ensino primário” OR “ensino básico”) AND (“App Inventor”) AND (“computação” OR “ciência da computação”)) and (PublishedAs:journal)	<i>(education OR teaching OR learning OR training) AND (“secondary school” OR “high school” OR “senior school” OR “primary school” OR “middle school” OR “elementary school” OR “K-12”) AND (“App Inventor”) AND (“computing” OR “computer science”) for: ((education OR teaching OR learning OR training) AND (“secondary school” OR “high school” OR “senior school” OR “primary school” OR “middle school” OR “elementary school” OR “K-12”) AND (“App Inventor”) AND (“computing” OR “computer science”)) and (PublishedAs:journal)</i>
IEEE Xplore	((ensino OR instrução OR aprendizagem OR treinamento) AND (“ensino médio” OR “ensino secundário” OR “colegial” OR “ensino fundamental” OR “ensino primário” OR “ensino básico”) AND (“App Inventor”) AND (“computação” OR “ciência da computação”))INmetadata) AND (pyr >= 2010 AND pyr <= 2015)	<i>((education OR teaching OR learning OR training) AND (“secondary school” OR “high school” OR “senior school” OR “primary school” OR “middle school” OR “elementary school” OR “K-12”) AND (“App Inventor”) AND (“computing” OR “computer science”))INmetadata) AND (pyr >= 2010 AND pyr <= 2015)</i>

Quadro 5: Termos utilizados nas buscas.

De forma a apoiar a resposta à pergunta de pesquisa, os dados extraídos dos trabalhos encontrados são:

- Objetivos do trabalho;
- Tipo de material elaborado (guia, jogo, material de suporte para o instrutor, quadrinhos para os alunos, cartões de instrução, outros);
- Metodologia e duração da aplicação do estudo de caso;
- Faixa etária do público-alvo;
- Sexo do público-alvo;
- Tipo da escola (pública/privada);
- Integração com computação física.

3.1.1 Critérios de Inclusão e Exclusão

Para os resultados encontrados, são consideradas apenas as publicações que atenderem aos seguintes critérios:

- O material encontrado deve ser voltado para o ensino de crianças e adolescentes;
- O trabalho deve incentivar o desenvolvimento de aplicativos para dispositivos móveis;
- O trabalho deve utilizar o App Inventor;
- O trabalho deve ser focado no ensino de computação e pensamento computacional.

3.1.2 Critérios de Qualidade

Os critérios de qualidade são:

- Deve estar disponível um artigo, com informações e relatos suficientes sobre o projeto, para que seja possível coletar os dados definidos.
- Deve incluir o relato de alguma aplicação prática.

3.2. Execução da Busca

Esta busca foi realizada em novembro de 2015, utilizando os termos do Quadro 5. Assim, o Quadro 6 e o Quadro 7 apresentam o número de resultados encontrados nas pesquisas e o número relevante de trabalhos que se encaixaram nos critérios estabelecidos anteriormente, em inglês e português (respectivamente).

Repositório	Quantidade de resultados encontrados (inglês)	Quantidade de resultados encontrados após o filtro pelos critérios de inclusão e exclusão e pelo critério de qualidade (inglês)
Google Scholar	208	2
ACM Digital Library	29	0
IEEE Explore	3	2

Quadro 6: Resultados da pesquisa em inglês.

Repositório	Quantidade de resultados encontrados (português)	Quantidade de resultados encontrados após o filtro pelos critérios de inclusão e exclusão e pelo critério de qualidade (português)
Google Scholar	17	2
ACM Digital Library	0	0
IEEE Explore	0	0

Quadro 7: Resultados da pesquisa em português.

Após a realização das buscas em ambas os idiomas, obteve-se um total de 225 resultados encontrados pelo Google Scholar, 29 resultados para a biblioteca digital

ACM e 3 no IEEE Xplore. Importante ressaltar que alguns trabalhos encontrados na pesquisa sobre a plataforma da ACM Digital Library e os três projetos encontrados na biblioteca IEEE Xplore, o Google Scholar também conseguiu encontrar, ou seja, os resultados de ambas as pesquisas retornaram alguns trabalhos iguais. Além disso, a execução da pesquisa mostrou que o número de trabalhos publicados em inglês sobre assuntos semelhantes a este projeto é bastante superior ao número de trabalhos publicados na língua portuguesa.

Foram, então, lidos os títulos e resumos de todos os trabalhos encontrados e, a partir disso, foram aplicados os critérios de inclusão e de exclusão. Após essa etapa, foram utilizados os critérios de qualidade para definir quais trabalhos estariam seriam considerados relevantes para este trabalho. Como resultado, foram identificados os trabalhos apresentados no quadro 8.

Título	Autores	Referência
Android Smart Girls.	Nadja Ramos, Carmen Freitas, Sandra Avila, Paula D. Paro Costa, Vanessa Testoni, Juliana Freitag Borin.	(AVILA, et al. 2015)
Lógica de Programação: Iniciação Lúdica com App Inventor for Android.	Tanicleide C. S. Gomes, Jeane C. B. de Melo.	(GOMES & MELO, 2013)
Using App inventor & history as a gateway to engage African American students in computer science.	Yerika Jimenez, Christina Gardner-McCune.	(JIMENEZ & GARDNER-MCCUNE, 2015)
Teaching Computational Thinking by Playing Games and Building Robots.	Jonathan Francis Roscoe, Stephen Fern, Emma Posey.	(ROSCOE, et al. 2014)

Quadro 8: Trabalhos identificados após aplicação dos critérios de inclusão/exclusão e critérios de qualidade.

A seguir cada um dos trabalhos é descrito em maiores detalhes e os dados planejados são coletados.

3.3. Extração e Análise dos Resultados

Os dados dos trabalhos encontrados estão apresentados nos quadros 9, 10, 11 e 12.

Nome do Projeto	Android Smart Girls.
Autor(es)	Nadja Ramos, Carmen Freitas, Sandra Avila, Paula D. Paro Costa, Vanessa Testoni, Juliana Freitag Borin.
Objetivos de aprendizagem	Proporcionar às alunas um contato inicial com programação (incluindo noções de computadores e smartphones), mostrando que isso pode ser algo estimulante e possível; apresentar e divulgar carreiras de ciências exatas, como engenharias e, principalmente, computação; e avaliar se o ensino de programação durante o ensino médio possui algum impacto positivo e significativo no desempenho das alunas em suas escolas e nas escolhas para a carreira futura.
Material desenvolvido	Uma apostila, onde cada capítulo apresentava um novo componente.
Metodologia e duração da aplicação do estudo	Dividido em duas fases: na primeira fase ocorreram 11 aulas presenciais e na segunda (que durou quatro meses) as alunas desenvolviam seus projetos individualmente, apenas utilizavam

	os instrutores para tirar dúvidas. Por fim, as alunas apresentavam seus projetos finais aos instrutores.
Público-alvo	Estudantes de ensino médio do sexo feminino.
Tipo da escola	N/D
Integração com computação física	Não possui.
Resultados obtidos	Uma queda no número de participantes do projeto ao longo do tempo, fazendo com que restassem poucas alunas no fim das aulas, impedindo a realização de uma conclusão eficiente sobre a evolução do pensamento computacional.

Quadro 9: Ensino de Programação para Alunas de Ensino Médio: Relato de uma Experiência (AVILA, et al. 2015)

Nome do Projeto	Lógica de Programação: Iniciação Lúdica com App Inventor for Android
Autor(es)	Tanicleide C. S. Gomes, Jeane C. B. de Melo
Objetivos de aprendizagem	Facilitar a aprendizagem de computação e atrair novas pessoas para essa área.
Material desenvolvido	Apresentações para aulas.
Metodologia e duração da aplicação do estudo	Foram desenvolvidas aulas divididas em 4 módulos diferentes: 1. Introdução ao Desenvolvimento de Aplicativos para Dispositivos Móveis, contendo a introdução aos conteúdos elaborados.

	<p>2. A Linguagem de Programação Visual App Inventor, apresentando a ferramenta e dividindo a apresentação de componentes em 4 etapas, aumentando a dificuldade em cada uma delas.</p> <p>3. Testes, onde os alunos testavam os aplicativos criados e corrigiam os erros.</p> <p>4. Publicando seus aplicativos na Web, fase de divulgação de suas 4 criações.</p>
Público-alvo	Alunos do ensino médio.
Tipo da escola	Pública.
Integração com computação física	Não possui.
Resultados obtidos	Pôde-se constatar que os estudantes aprenderam os conteúdos explanados, bem como seus discursos enunciaram motivação, engajamento e interesse pelos conceitos apresentados.

Quadro 10: Lógica de Programação: Iniciação Lúdica com App Inventor for Android (GOMES & MELO, 2013)

Nome do Projeto	Using App inventor & history as a gateway to engage African American students in computer science
Autor(es)	Yerika Jimenez, Christina Gardner-McCune
Objetivos de aprendizagem	Alinhar o pensamento computacional com pensamento histórico através de cursos de história, com o intuito de atrair os participantes aos ensinamentos das ciências da computação através dessa integração.

Material desenvolvido	Aulas e guias.
Metodologia e duração da aplicação do estudo	Desenvolvimento de aplicativos multidisciplinares, fazendo com que as criações englobem tanto a matéria de computação como a de história. Essa metodologia baseia-se em uma abordagem curricular construtivista, onde os alunos constroem e produzem conhecimento, artefatos e tecnologias, em vez de consumi-los através do ensinamento dos professores.
Público-alvo	Afro-americanos do ensino médio e fundamental.
Tipo da escola	N/D
Integração com computação física	Não possui.
Resultados obtidos	Os alunos se motivaram com o assunto e ficaram extremamente entusiasmados pela criação de seus próprios aplicativos móveis. Eles se comprometeram a entrar mais afundo nos estudos computacionais.

Quadro 11: Using App inventor & history as a gateway to engage African American students in computer science (JIMENEZ & GARDNER-MCCUNE, 2015)

Nome do Projeto	Teaching Computational Thinking by Playing Games and Building Robots
Autor(es)	Jonathan Francis Roscoe, Stephen Fern, Emma Posey
Objetivos de aprendizagem	Ensinar habilidades computacionais (o “pensamento computacional”).

Material desenvolvido	Aulas, guias e softwares personalizados;
Metodologia e duração da aplicação do estudo	<p>O projeto ficou dividido em três etapas:</p> <ol style="list-style-type: none"> 1. Os participantes aprendem sobre design computacional e desenham um mundo em miniatura em um jogo chamado Minecraft, e então esse desenho é enviado para uma impressora 3D para sua criação. 2. Na segunda fase é ensinado aos alunos como eles podem ter uma outra perspectiva de coisas familiares, utilizando o App Inventor para criação de aplicativos móveis. 3. Os alunos vão para um curso de robótica e devem projetar e criar uma variedade de robôs utilizando o Scratch e Arduino como ferramentas principais.
Público-alvo	Alunos do ensino fundamental.
Tipo da escola	N/D
Integração com computação física	Possui integração com Arduino, porém via outra plataforma (não pelo App Inventor).
Resultados obtidos	Notou-se certa dificuldade em alguns aspectos da aplicação do projeto, como a impressão dos desenhos elaborados na primeira fase. Em contrapartida ocorreu um aproveitamento e um feedback positivo por parte dos alunos,

Quadro 12: Teaching Computational Thinking by Playing Games and Building Robots (ROSCOE, et al. 2014)

3.4 Discussão

A revisão do estado da arte levantou indícios de que são poucos e recentes os trabalhos que desenvolvem uma unidade instrucional para o ensino do pensamento computacional utilizando o App Inventor 2, principalmente na língua portuguesa. Além disso, vários trabalhos encontrados utilizavam uma versão anterior da plataforma, com menos componentes disponíveis no software para a criação de aplicativos.

De forma geral, todos os trabalhos pecaram na divulgação dos resultados. O foco acabou ficando mais na opinião dos participantes sobre o projeto e não na evolução deles após a conclusão do trabalho.

Além disso notou-se que haviam poucas informações sobre o tipo de escolas e quais conteúdos foram elaborados pelos criadores dos projetos para a aplicação dos testes e verificação dos resultados.

Deve-se levar em consideração que o trabalho, entre todos os encontrados, que mais se assemelha a este, tanto pela linguagem como pelas características do projeto, é o estudo elaborado por Avila et al. (2015), relatado em forma de artigo. Porém existem alguns aspectos diferentes que devem ser levados em consideração:

- O projeto é voltado para apenas crianças e adolescentes que são do sexo feminino;
- Um dos objetivos do projeto era o de incentivo de algumas profissões, o que foge do escopo central deste trabalho;
- Os desenvolvedores não conseguiram chegar a conclusões definitivas sobre a evolução do pensamento computacional das participantes.

O trabalho de Gomes e Melo (2013) se mostrou interessante por fazer uma abordagem mais rápida dos conteúdos, porém a qualificação e possível quantificação

do desenvolvimento do pensamento computacional ao longo do curso não estava priorizado no trabalho, apenas o incentivo à computação.

Já o estudo desenvolvido por Jimenez e Gardner-McCune (2015) possui dois focos muito específicos e que não se encaixam neste projeto, que são:

- Especificação demasiada do público-alvo, diminuindo a abrangência do artigo;
- Utilização de outra matéria para o incentivo do estudo de computação.

Um dos objetivos da pesquisa era encontrar trabalhos que realizassem integração do aplicativo desenvolvido pelos alunos com computação física, principalmente com Arduino, utilizando o protocolo Firmata e fazendo a conexão entre os equipamentos via *Bluetooth*. Porém, mesmo realizando alterações nos termos de busca, não foi possível encontrar trabalhos que se encaixassem nos critérios e se assemelhassem com este. Isso pode representar um diferencial ainda não abordado em nenhum estudo.

3.4.1 Ameaças à Validade da Revisão

Apesar das pesquisas terem sido realizadas tanto na língua nacional da publicação deste trabalho (português), como em inglês, isso não inibe o risco de que algum estudo possa ter sido omitido na execução da busca, devido à escolha dos termos de busca.

Vale ressaltar que para a realização da busca foram utilizados os termos “computação” e “ciência da computação” e suas respectivas traduções. O ideal seria realizar a busca com o termo “pensamento computacional”, seus sinônimos e traduções, porém o retorno da busca não trazia nenhum resultado e foi optado por alterar essas expressões.

Além disso, o critério de qualidade da pesquisa, exigindo que seja obrigatório um artigo divulgado com as informações sobre o desenvolvimento do projeto, pode ter excluído trabalhos com estudos e conclusões que poderiam ser relevantes.

Outra ameaça à validade consiste no fato de que somente um pesquisador, o autor deste trabalho, participou desta revisão sistemática, o que pode comprometer a isenção da aplicação dos critérios de inclusão e exclusão.

4. UNIDADE INSTRUCIONAL PARA O ENSINO DE COMPUTAÇÃO

Este capítulo apresenta a análise do contexto atual sobre o ensino de computação em escolas e o design da unidade instrucional para o ensino de computação utilizando o App Inventor e computação física. Através dos objetivos de aprendizagem e ensino é elaborado o plano de ensino e, após isso, são apresentados os materiais desenvolvidos para a unidade instrucional.

4.1 Análise do Contexto

De acordo com o design instrucional, para se definir os objetivos de aprendizagem é necessário realizar uma análise do público alvo e do ambiente. Neste projeto, o público alvo são alunos do ensino fundamental (a partir do sexto ano) e médio; enquanto o ambiente escolhido são as escolas públicas e particulares do Brasil.

A unidade instrucional desenvolvida é projetada para o ensino de computação em escolas, logo ela só pode ser aplicada em instituições de ensino que possuam em sua infraestrutura um laboratório de informática ou alguma sala que disponha de computadores para a utilização em aula.

Segundo o INEP (2014), aproximadamente 51% das escolas públicas de ensino fundamental no Brasil possuem laboratórios de informática e cerca de 50% possuem acesso à Internet. Se tratando de escolas particulares, o percentual de instituições de ensino com laboratórios de informática muda para 55% e o número de escolas com acesso à Internet salta para 93% (INEP, 2014).

Por ser uma unidade instrucional com o objetivo de desenvolver um aplicativo para aparelhos móveis, é necessário verificar os dados sobre essa parcela dos estudantes com acesso à um *smartphone*. Segundo o IBGE(2014) 73,6% do alunos

brasileiros acima de 10 anos já possui conhecimentos e habilidades no uso de dispositivos eletrônicos por possuírem e utilizarem telefone móvel. Cerca de 66,8% dos estudantes da rede pública possuem celular com internet móvel enquanto, na rede de escolas privadas, esse número pula para 93,4% (figura 7). A proporção de pessoas com *smartphone* continua tendo um acréscimo todos os anos, similar ao comportamento de acesso à Internet (IBGE, 2014).

Atualmente a maior parte da população utiliza a Internet. Em 2014 foram 95,4 milhões de brasileiros, ou 54,4% da população com dez anos ou mais idade, que acessaram a Internet pelo menos uma vez (em um período de três meses). Para se ter como base, em 2013 esta parcela era de 49,4% da população (IBGE, 2014).

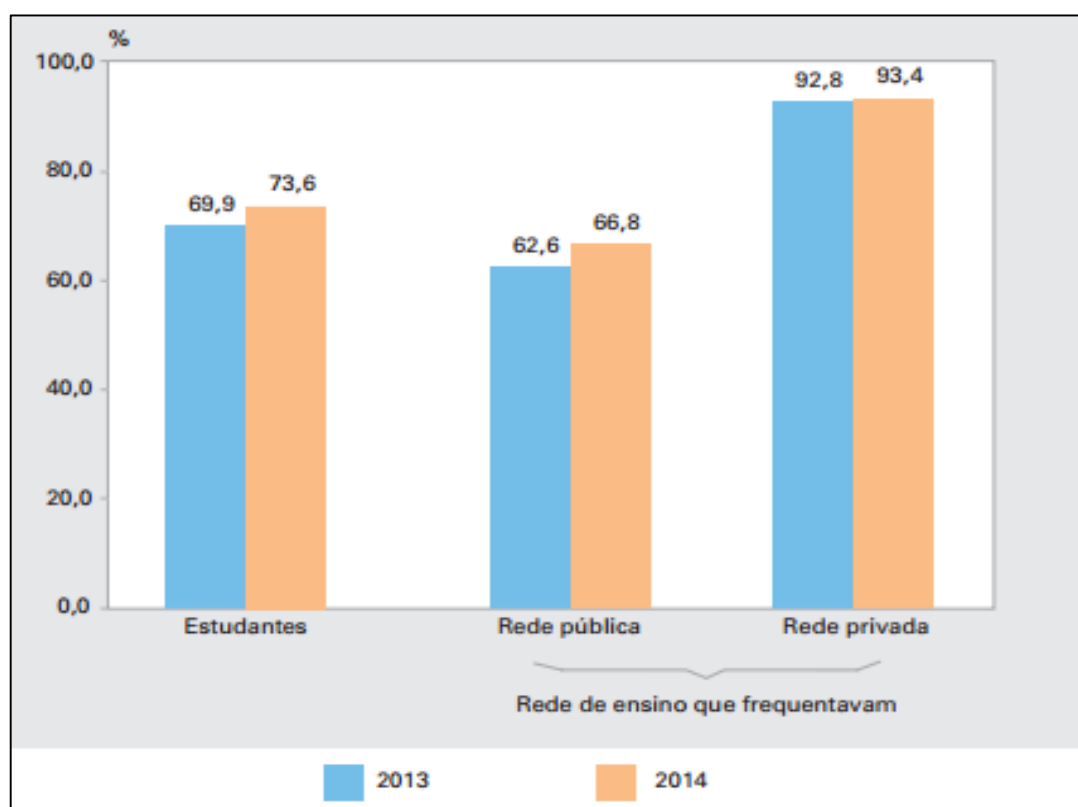


Figura 7: Percentual de pessoas que tinham telefone móvel celular, na população de estudantes de 10 anos ou mais de idade, segundo a condição de estudante e a rede de ensino que frequentavam. (IBGE, 2014, adaptado)

Dos aparelhos celulares vendidos no Brasil, cerca de 91% utilizam o sistema operacional Android (compatível com os aplicativos criados no App Inventor) e esse número vem crescendo (IDC Brasil, 2014). Projeta-se que esse número aumentará para 93% nos próximos anos.

4.2 Design da Unidade Instrucional

Para a elaboração da unidade instrucional foram utilizados os objetivos de aprendizagem no ensino de computação estabelecidos pelo currículo referência CSTA/ACM K-12 para o nível 2. Após a análise de contexto, são selecionados os objetivos de aprendizagem para a elaboração da unidade instrucional (Quadro 13).

Pensamento Computacional	<p>[O1] Usar os passos básicos de algoritmos para a resolução de problemas ao projetar soluções (p.ex., declaração e exploração do problema, examinação de exemplos, design, implementação de uma solução, testes, avaliação).</p> <p>[O3] Definir um algoritmo, como sendo uma sequência de instruções que podem ser processadas por um computador.</p> <p>[O6] Descrever e analisar uma sequência de instruções a ser seguida (p.ex., descrever o comportamento de um personagem em um vídeo game, dirigido por regras e algoritmos).</p>
Colaboração	<p>[O17] Colaborativamente criar, desenvolver, publicar e apresentar produtos (p.ex., vídeos, podcasts, sites), utilizando recursos tecnológicos que demonstram e comunicam conceitos do currículo.</p> <p>[O18] Colaborar com colegas, especialistas e outros utilizando práticas colaborativas como programação em pares, trabalho em equipes de projeto, e participação em atividades de aprendizagem ativa em grupo.</p>
Práticas Computacionais	<p>[O20] Selecionar ferramentas e recursos tecnológicos apropriados para realizar tarefas variadas e resolver problemas.</p>

e de Programação	<p>[O22] Conceber, desenvolver, publicar e apresentar produtos (p.ex., páginas web, aplicações móveis, animações) usando recursos de tecnologia que demonstram e comunicam os conceitos do currículo.</p> <p>[O23] Demonstrar uma compreensão de algoritmos e a sua aplicação prática.</p> <p>[O24] Implementar soluções de problema utilizando uma linguagem de programação, incluindo: o comportamento de laços (sequências de instruções que se repetem), instruções condicionais, lógica, expressões, variáveis e funções.</p>
Computadores e Dispositivos de Comunicação	<p>[O30] Identificar uma variedade de dispositivos eletrônicos que contêm processadores computacionais.</p> <p>[O31] Demonstrar compreensão sobre a relação entre hardware e software.</p>

Quadro 13: Objetivos de Aprendizagem da Unidade Instrucional, seguindo o currículo referência CSTA/ACM K-12 para o nível 2

A duração da aplicação da unidade instrucional ficou estabelecido em quatro horas, que corresponde a um dia letivo dos alunos (COSTA, 2009). Durante a aplicação serão apresentados os conceitos básicos de programação e do funcionamento do App Inventor. Por existir um número limitado de robôs, os alunos serão divididos em duplas ou trios. Baseado nisso, foi desenvolvido o Plano de Ensino da unidade instrucional (Quadro 14).

Plano de Ensino – Unidade Instrucional para o Ensino de Computação com App Inventor 2	
Objetivos de Aprendizagem	
O objetivo geral da Unidade Instrucional é ensinar aos alunos alguns conceitos básicos de programação e computação relacionados ao pensamento computacional.	[O6] Descrever e analisar uma sequência de instruções a ser seguida (p.ex., descrever o comportamento de um personagem em um vídeo game, dirigido por regras e algoritmos).

<p>Ao fim da aplicação de todo o conteúdo os alunos deverão obter os seguintes conhecimentos:</p> <ol style="list-style-type: none"> 1. Conhecer o conceito de linguagens de programação e algoritmo. 2. Assimilar o que é o App Inventor e o que pode ser feito neste <i>software</i>. 3. Utilizar a plataforma do App Inventor e criar um aplicativo. 4. Entender e utilizar vários conceitos de programação como laços de repetição, estrutura condicional, variáveis, diferentes componentes de layouts, etc. 5. Analisar o uso de aplicativos móveis. 6. Entender o conceito de computação física. <p>Seguindo o currículo referência CSTA/ACM K-12 para alunos de sexto à nono ano, os alunos conseguirão desenvolver as seguintes habilidades:</p> <p>[O1] Usar os passos básicos de algoritmos para a resolução de problemas ao projetar soluções (p.ex., declaração e exploração do problema, examinação de exemplos, design, implementação de uma solução, testes, avaliação).</p> <p>[O3] Definir um algoritmo, como sendo uma sequência de instruções que podem ser processadas por um computador.</p>	<p>[O17] Colaborativamente criar, desenvolver, publicar e apresentar produtos (p.ex., vídeos, podcasts, sites), utilizando recursos tecnológicos que demonstram e comunicam conceitos do currículo.</p> <p>[O18] Colaborar com colegas, especialistas e outros utilizando práticas colaborativas como programação em pares, trabalho em equipes de projeto, e participação em atividades de aprendizagem ativa em grupo.</p> <p>[O20] Selecionar ferramentas e recursos tecnológicos apropriados para realizar tarefas variadas e resolver problemas.</p> <p>[O22] Conceber, desenvolver, publicar e apresentar produtos (p.ex., páginas web, aplicações móveis, animações) usando recursos de tecnologia que demonstram e comunicam os conceitos do currículo.</p> <p>[O23] Demonstrar uma compreensão de algoritmos e a sua aplicação prática.</p> <p>[O24] Implementar soluções de problema utilizando uma linguagem de programação, incluindo: o comportamento de laços (sequências de instruções que se repetem), instruções condicionais, lógica, expressões, variáveis e funções.</p> <p>[O30] Identificar uma variedade de dispositivos eletrônicos que contêm processadores computacionais.</p> <p>[O31] Demonstrar compreensão sobre a relação entre hardware e software.</p>
--	---

Conteúdo				
<ol style="list-style-type: none"> 1. Apresentar a Unidade Instrucional, explicando sobre o App Inventor, computação física e conceitos básicos de programação (linguagens, algoritmos, etc.) 2. Acessar o App Inventor 2, ensinando como fazer o acesso à plataforma. 3. Ensinar como utilizar o App Inventor 2: a sincronia entre a tela de design do layout e a tela de blocos de códigos. 4. Criar um aplicativo com o App Inventor 2. 5. Exportar e testar o aplicativo, para fazer a análise do software. 				
Sequência da Aula	Minutos	Estratégia Instrucional	Material Utilizados	Avaliação
Apresentação e Explicação da Unidade Instrucional (App Inventor e Computação Física)	25	Aula Expositiva	Slides	Prova – Questões 1, 3 e 8
Apresentação e Explicação de Conceitos Básicos de Programação	35	Aula Expositiva	Slides	Prova – Questões 5, 6, 7 e 9
Acesso ao App Inventor e Criação do Projeto	15	Aula Expositiva e Atividade Prática (desenvolvimento em dupla ou trio)	Slides Contas do Google Roteiro de Acesso Roteiro do Aplicativo Site App Inventor	Prova – Questões 1, 4 e 8
Apresentação e Desenvolvimento do Aplicativo (Conexão Bluetooth)	40	Aula Expositiva e Atividade Prática (desenvolvimento em dupla ou trio)	Slides Roteiro do Aplicativo Site App Inventor	Prova – Questões 2, 6 e 7
Intervalo	20			
Desenvolvimento do Aplicativo (Funções do Robô)	30	Aula Expositiva e Atividade Prática (desenvolvimento em dupla ou trio)	Slides	Prova – Questões 6 e 7

			Roteiro do Aplicativo Site App Inventor	
Uso do Aplicativo (Testes e Correções)	25	Exportação do Projeto e Atividade Prática (testando as funcionalidades do aplicativo)	Slides Roteiro do Aplicativo Site App Inventor Roteiro de Exportação	Prova – Questões 3 e 4
Aplicação da Prova (Avaliação)	50	Atividade Prática	Prova de Avaliação	---

Quadro 14: Plano de Ensino da Unidade Instrucional.

O aplicativo criado é o “Controle do Robô”. Após a elaboração do aplicativo, os alunos poderão testar com seus respectivos robôs, através de uma conexão Bluetooth, e fazê-lo realizar as seguintes funções: piscar e levantar o braço.

Antes de aplicar a unidade instrucional é necessário verificar alguns pré-requisitos essenciais obrigatórios. Sem eles não é possível realizar todas as atividades da UI. Os pré-requisitos são:

- Para acessar o App Inventor é necessário utilizar o navegador Mozilla Firefox ou o Google Chrome.
- Projetor multimídia para a apresentação dos slides durante a aula.
- Computadores com teclado, mouse e monitor, utilizados na programação do aplicativo pelos alunos.
- Acesso à internet nos dispositivos móveis e no computador.
- Dispositivo móvel com o sistema operacional Android na versão 2.3 ou superior. O app criado só é suportado a partir da versão 2.3 do Android.

- Arduino e os seguintes componentes: 1 módulo Bluetooth, 1 servomotor e 2 LEDs. Também é necessário um cabo USB para conectar o Arduino ao computador e utilizá-lo como fonte de energia.
- Conta no Google (Gmail) para cada dupla ou trio de alunos para o acesso ao App Inventor.

4.3 Desenvolvimento da Unidade Instrucional

Após a elaboração do plano de ensino foram desenvolvidos os seguintes materiais para a UI: informativo sobre a unidade instrucional; aplicativo “Controle do Robô” para a integração com computação física; roteiros para os professores; *slides* para a aplicação em aula; tarefa para ser realizada em casa; uma prova para avaliar o conteúdo aprendido; e rubrica para avaliação.

A Unidade Instrucional está disponibilizada sob a licença Creative Commons Atribuição-NãoComercial-Compartilhalgual 4.0 Internacional e pode ser acessada no link:

<<https://drive.google.com/drive/u/0/folders/0BzKHw8EKAYsLdWkwUHIJSGZSbEk>>.

4.3.1 Informativo da UI

Foi desenvolvido um informativo da unidade instrucional para apresentá-la (Figura 8). No informativo estão tópicos de introdução sobre o App Inventor e computação física. Além disso, o informativo relata os benefícios do ensino de computação e realiza uma breve apresentação da Iniciativa Computação na Escola.

Introdução com Slides

Programando no Computador

Testando com celular e robô

Prova (Avaliação)

QUAIS OS BENEFÍCIOS DE APRENDER COMPUTAÇÃO?

O conhecimento em computação auxilia em diversos aspectos diferentes do nosso cotidiano, como raciocínio lógico, inovação e criatividade. Obter esse conhecimento cedo auxilia ainda mais no desenvolvimento destes aspectos. Além disso, vivemos em um mundo cada vez mais tecnológico, com o desenvolvimento computacional se espalhando para múltiplas áreas. É importante termos o conhecimento básico de programação para não nos tornarmos reféns da tecnologia.

O QUE É O APP INVENTOR?

O App Inventor é uma linguagem visual de "blocos" utilizada na criação de aplicativos para dispositivos móveis. Ele é o fruto de um programa piloto da Google em parceria com o Instituto de Tecnologia de Massachusetts. Por ter uma interface amigável, ser um software livre (de uso gratuito) e possuir a opção de programar em português, essa plataforma se torna ideal para o contato inicial com a programação computacional. Para saber mais, acesse o site: appinventor.mit.edu (em inglês).

O QUE É COMPUTAÇÃO FÍSICA?

A Computação Física basicamente significa o ato desenvolver um sistema físico interativo utilizando componentes computacionais que interagem com o ambiente. Existem diversos tipos de componentes diferentes, como sensores de temperatura, sensores de luz e LEDs. Neste projeto são utilizados LEDs, módulo Bluetooth e servomotores.

Figura 8: Parte do Informativo da Unidade Instrucional.

4.3.2 Aplicativo – Controle do Robô

Foi desenvolvido o aplicativo “Controle do Robô” para o ensino de computação nesta unidade instrucional. O objetivo do aplicativo é controlar um robô, realizando uma conexão Bluetooth entre o *app* e o Arduino (com módulo Bluetooth) e fazê-lo realizar duas ações: piscar (LEDs conectados a um Arduino) e mexer o braço (servomotor conectado ao Arduino). A tela do aplicativo pode ser vista na figura 9.

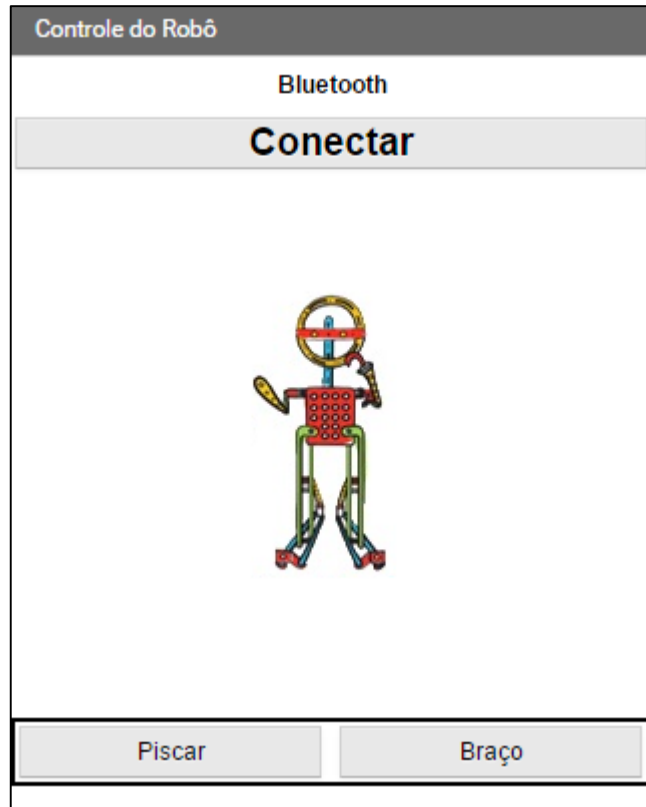


Figura 9: Tela do Aplicativo “Controle do Robô”.

Para a criação do *app* foram utilizados os componentes da tela de Design do App Inventor apresentados no quadro 15. Durante a programação, além de utilizados os blocos relacionados à esses componentes, foram utilizados blocos de Controle, Lógica, Matemática, Texto e Variáveis.

Componentes de Design	
Interface de Usuário	<ul style="list-style-type: none"> - Botão - Legenda - Imagem - Lista - Notificador
Organização	- Organização Horizontal
Sensores	- Temporizador
Conectividade	<ul style="list-style-type: none"> - Cliente Bluetooth - Iniciador de Atividades

Quadro 15: Componentes de Design utilizados na criação do aplicativo.

4.3.3 Roteiros para Professores

Os roteiros criados servem como passo a passo para a criação e divulgação do aplicativo elaborado na Unidade Instrucional.

O primeiro roteiro é utilizado para acessar o App Inventor, apresentando os passos de criação da conta de acesso e como configurar o App Inventor para a criação do projeto (Figura 10).



Figura 10: Parte do Roteiro de Acesso.

O segundo roteiro é onde está o guia de aula e o guia para a criação do aplicativo "Controle do Robô" no App Inventor. Ele mostra todos os detalhes da criação, apresentando passo a passo a ordem do desenvolvimento tanto na tela de design da interface como na tela de edição de blocos (Figura 11). Ele contém todas as explicações necessárias para entender a programação e funcionalidade do *app*.

Na coluna “Componentes” selecione a opção “EscolheLista1”. Em seguida clique em “Largura” e altere para “Preencher principal”. Pressione “OK”.

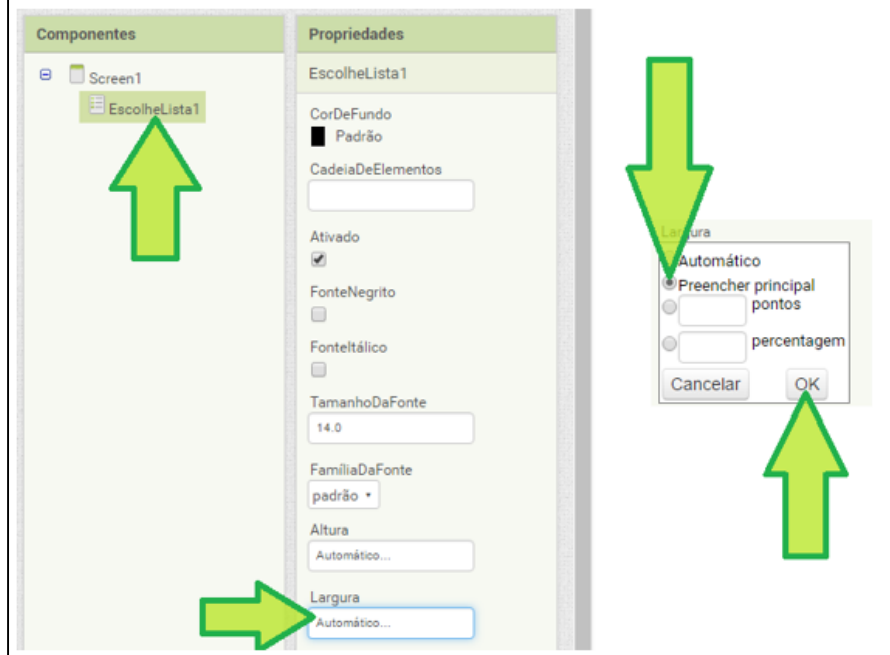


Figura 11: Parte do Roteiro de Aula/Criação do Aplicativo.

O terceiro e último roteiro contém as informações de compartilhamento do aplicativo, mostrando como fazer os testes e o *download* do *app* (Figura 12). Ele é utilizado para exportar o aplicativo do App Inventor para o celular.

4 - Abra o aplicativo MIT AI2 Companion no seu aparelho e clique em “scan QR Code”. A câmera do seu celular irá abrir; basta apontá-la para o código de barras gerado no passo anterior.

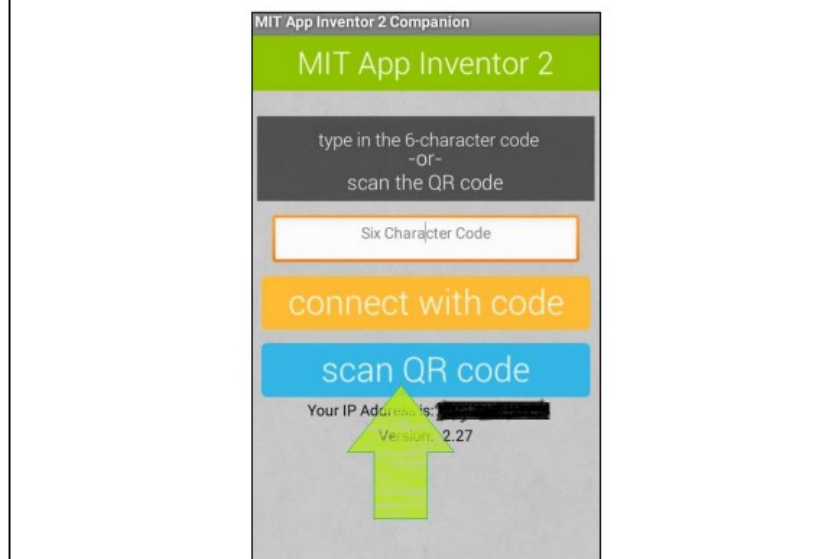


Figura 12: Parte do Roteiro de Exportação.

4.3.4 Slides de Aula

Foi desenvolvida uma apresentação com *slides* (Figura 13) para ser utilizada pelo professor durante a aplicação da UI. Além de possuir o conteúdo para o desenvolvimento do aplicativo com o App Inventor, essa apresentação possui conceitos básicos de lógica de programação e serve como guia no desenvolvimento do aplicativo. Os alunos acompanham a criação do *app* pelos *slides*.

Conexão Bluetooth

```
quando BotaoConectar >> Clique
fazer
  se ClienteBluetooth1 >> EstáConectado
  então
  senão
```

Verifica se o celular já está conectado com o Arduino.

FUNCIONAMENTO DA AULA

Slides, Programação, Uso do Aplicativo

- Introdução à conceitos de computação e programação.
- Desenvolvimento do aplicativo, acompanhando os *slides*.
- Instalação do aplicativo no celular e uso com o robô.
- Realização da avaliação.

ESTRUTURAS CONDICIONAIS - EXEMPLO NO APP INVENTOR

O exemplo da "nota do João" ficaria assim no App Inventor:

```
se
  obter global notaDoJoao >> <= 6
  então chamar Notificador1 >> MostrarAlerta
  aviso João está em recuperação!
  senão chamar Notificador1 >> MostrarAlerta
  aviso João está aprovado!
```

Dependendo da nota, uma notificação aparecerá informando se o "João está em recuperação!" ou se o "João está aprovado!".

O que vamos aprender?

- Conceitos de programação.
- O que é computação física.
- Criar um aplicativo no App Inventor e testá-lo no celular.
- Conectar o aplicativo com computação física.

Figura 13: Exemplos de Slides da Unidade Instrucional.

4.3.5 Avaliação

Foi desenvolvida uma prova/avaliação (Figura 14), com pontuação de 0 a 10, para medir o conhecimento adquirido pelo aluno após a aplicação da unidade instrucional. O gabarito da avaliação também está incluso na UI.

Colégio: _____ Data: __/__/__

Professor (a): _____ Turma: _____

Aluno (a): _____

Avaliação

Questão 1 (1 ponto) – O que é o App Inventor?

Questão 2 (1 ponto) – Assinale a alternativa correta sobre Bluetooth.

É possível fazer uma conexão Bluetooth do seu celular com um *smartphone* localizado no outro lado do mundo.

Para realizar uma conexão Bluetooth é necessário utilizar um cabo USB.

Figura 14: Parte da Prova/Avaliação.

4.3.6 Tarefa de Casa

Foi desenvolvido um material para ser utilizado como tarefa de casa (Figura 15). Esse material contém um roteiro para o aluno e exercícios a serem realizados. O gabarito dos exercícios também está disponibilizado na unidade instrucional.

Exercícios Teóricos (6 pontos)

Questão 1 (1,5 ponto) – O que é uma placa Arduino? Cite exemplos de produtos eletrônicos que podem ser criados com um Arduino.

Questão 2 (2 pontos) – Observe atentamente o código. Qual será o valor de Y se pressionarmos o botão “Calcular” 5 vezes? E qual será o valor de Y se nunca pressionarmos o botão “Calcular”?

```

inicializar global Y para 5
quando Calcular .Clique
fazer
  ajustar global Y para obter global Y + 4
  ajustar global Y para obter global Y + 10
  ajustar global Y para 10 + obter global Y
  ajustar global Y para 1 + 2

```

Figura 15: Parte da Tarefa de Casa.

4.3.7 Rubrica de Avaliação

Foi desenvolvida uma rubrica de avaliação dos alunos (figura 16), estabelecendo os critérios de medição para a análise do conhecimento desenvolvido na criação do aplicativo, aplicação da prova e na realização da tarefa de casa.

Rubrica de Avaliação - Objetivos da Unidade Instrucional
Essa rubrica apresenta os indicadores para avaliar os objetivos de aprendizagem da Unidade Instrucional para o Ensino de Computação utilizando App Inventor 2
Recursos de Avaliação
Desenvolvimento do App “Controle do Robô”: Nota de 0 a 10. Prova: Nota de 0 a 10. Tarefa de Casa: Nota de 0 a 10.
Objetivos
Entender o que é uma linguagem de programação, um algoritmo e utilizar algoritmos para a resolução de diferentes problemas. Descrever o que é e o que pode ser feito com o App Inventor. Compreender conceitos de programação, como laços de repetição, estruturas condicionais, variáveis e componentes de layout. Descrever o que é computação física.

Figura 16: Parte da Rubrica de Avaliação.

5. AVALIAÇÃO DA UNIDADE INSTRUCIONAL

A definição, aplicação e análise da avaliação da unidade instrucional estão apresentadas neste capítulo.

5.1 Definição da Avaliação

A avaliação da unidade instrucional tem como objetivo analisar a qualidade do conteúdo da unidade instrucional e se o conteúdo é ideal para a aplicação em ensino fundamental/médio.

Para isso foi utilizada a abordagem definida no método GQM (BASILI et al., 1994) com a intenção de elaborar um plano de medição para a realização da avaliação. Utilizando a metodologia GQM, inicialmente é necessário definir os objetivos para então estabelecer as perguntas que questionam estes objetivos. Para cada pergunta é definida uma medida relevante que apresente a sua resposta (BASILI et al., 1994).

Foram definidos os seguintes objetivos para a realização da avaliação desta unidade instrucional:

- **Objetivo de Medição 1:** Avaliar a qualidade do conteúdo da unidade instrucional, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional.
- **Objetivo de Medição 2:** Avaliar a aplicabilidade da unidade instrucional em turmas do ensino fundamental (entre o sexto e nono ano) e do ensino médio, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional.

- **Objetivo de Medição 3:** Avaliar o interesse em utilizar a unidade instrucional, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional.

São definidas perguntas e medidas para cada objetivo de medição, seguindo a abordagem GQM. Apesar do ideal ser quantificar os dados para as medições, foi observado que, tanto para o objetivo 1 quando para o objetivo 2, os dados a serem coletados são dados qualitativos (obtidos através de questionário). As perguntas e medidas podem ser observadas no quadro 16.

Objetivo de Medição 1:	Avaliar a qualidade do conteúdo da unidade instrucional, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional.
Pergunta Q1:	Os objetivos de aprendizagem podem ser atingidos utilizando a unidade instrucional?
Medida Q1:	Impressão subjetiva sobre o atingimento dos objetivos.
Pergunta Q2:	O material da unidade instrucional (plano de ensino, slides, etc.) é completo?
Medida Q2:	Impressão subjetiva sobre o material.
Objetivo de Medição 2:	Avaliar a aplicabilidade da unidade instrucional em turmas do ensino fundamental (entre o sexto e nono ano) e do ensino médio, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional.

Pergunta Q3:	Qual o nível de dificuldade que os alunos do ensino fundamental (entre o sexto e nono ano) teriam em usar e entender o conteúdo da unidade instrucional?
Medida Q3:	Impressão subjetiva sobre o nível de dificuldade para os alunos do ensino fundamental.
Pergunta Q4:	Qual o nível de dificuldade que os alunos do ensino médio teriam em usar e entender o conteúdo da unidade instrucional?
Medida Q4:	Impressão subjetiva sobre o nível de dificuldade para os alunos do ensino médio.
Pergunta Q5:	Qual o nível de dificuldade que o professor teria em aplicar a unidade instrucional em sala de aula?
Medida Q5:	Impressão subjetiva sobre o nível de dificuldade para o professor.
Objetivo de Medição 3:	Avaliar o interesse em utilizar a unidade instrucional, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional.
Pergunta Q6:	Os alunos do ensino fundamental (entre o sexto e nono ano) se interessariam em aprender computação utilizando essa unidade instrucional?
Medida Q6:	Impressão subjetiva sobre o interesse dos alunos do ensino fundamental.

Pergunta Q7:	Os alunos do ensino médio se interessariam em aprender computação utilizando essa unidade instrucional?
Medida Q7:	Impressão subjetiva sobre o interesse dos alunos do ensino médio.
Pergunta Q8:	Os professores se interessariam em utilizar essa unidade instrucional para o ensino de computação?
Medida Q8:	Impressão subjetiva sobre o interesse dos professores.

Quadro 16: Perguntas e medidas relacionadas aos objetivos de medição.

Com o objetivo de manter a qualidade da pesquisa apenas professores que deram/dão aulas de informática em escolas de ensino médio e/ou fundamental e que possuem formação na área de TI (como graduação em Ciências da Computação ou Sistemas de Informação) é que participaram da avaliação.

5.2 Aplicação da Avaliação

Utilizando os Objetivos de Medição definidos na seção 5.1 foi elaborado um questionário (figura 17) para a aplicação da avaliação.

Questionário de Avaliação da Unidade Instrucional
Este questionário tem como objetivo avaliar a Unidade Instrucional para o ensino de computação na escola utilizando App Inventor 2 e computação física.
Perfil do Avaliador
Nome:
Idade:
Sexo:
Formação Acadêmica:
Questões – Conteúdo da Unidade Instrucional
Pergunta: Os objetivos de aprendizagem podem ser atingidos utilizando a unidade instrucional.
Resposta:
(<input type="checkbox"/>)Discordo Totalmente (<input type="checkbox"/>)Discordo (<input type="checkbox"/>)Concordo (<input type="checkbox"/>)Concordo Totalmente
Pergunta: O material da unidade instrucional (plano de ensino, slides, etc.) é completo.
Resposta:
(<input type="checkbox"/>)Discordo Totalmente (<input type="checkbox"/>)Discordo (<input type="checkbox"/>)Concordo (<input type="checkbox"/>)Concordo Totalmente

Figura 17: Parte do Questionário de Avaliação.

A unidade instrucional foi apresentada e avaliada por cinco professores, que preenchem os pré-requisitos para a avaliação definidos também na seção 5.1.

Os avaliadores foram convidados pelo autor do projeto à, voluntariamente, analisarem a unidade instrucional e responder um questionário de avaliação, dando sua opinião sobre o material.

Primeiramente foi explicado o objetivo e a motivação da criação da unidade instrucional, citando o público-alvo e o contexto do projeto. Em seguida todo o material elaborado para a unidade instrucional foi apresentado aos avaliadores (informativo, plano de ensino, *slides* de aula, roteiros em geral, código e executável do aplicativo, prova para avaliação e seu respectivo gabarito, tarefa de casa e seu respectivo gabarito e rubrica de avaliação).

Foi esclarecido que o objetivo final é deixar disponível publicamente esta unidade instrucional para o uso de qualquer professor interessado em utilizá-la para ensinar tópicos relacionados à computação e programação.

Ao final os avaliadores responderam o questionário de avaliação da UI e fizeram observações sobre a unidade instrucional. O autor do projeto então agradeceu à participação dos avaliadores no projeto. Alguns professores demonstraram contentamento em estar participando deste projeto e agradeceram o convite.

5.3 Análise da Avaliação

Conforme definido previamente (Seção 5.1) foi realizada uma avaliação através de um questionário, respondido pelos professores avaliadores da unidade instrucional. O questionário foi elaborado utilizando a metodologia GQM, definindo os objetivos, perguntas e medidas (BASILI et al., 1994).

Com as medidas obtidas através desta metodologia foi possível criar gráficos que interpretam as opiniões dos professores avaliadores a respeito da unidade instrucional. Os gráficos gerados auxiliam na identificação dos aspectos positivos e dos aspectos negativos desta unidade instrucional.

A seguir estão divididos os Objetivos de Medição e suas respectivas perguntas e medidas, para uma melhor análise da avaliação.

5.3.1 Conteúdo da UI

O primeiro Objetivo de Medição é “avaliar a qualidade do conteúdo da unidade instrucional, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional”. Para isso foram realizadas duas perguntas, com a resposta de cada pergunta sendo a sua medida.

A primeira pergunta foi sobre os objetivos de aprendizagem da UI. Todos os avaliadores acreditam que os objetivos de aprendizagem podem ser atingidos utilizando a unidade instrucional, como pode ser visto na figura 18.

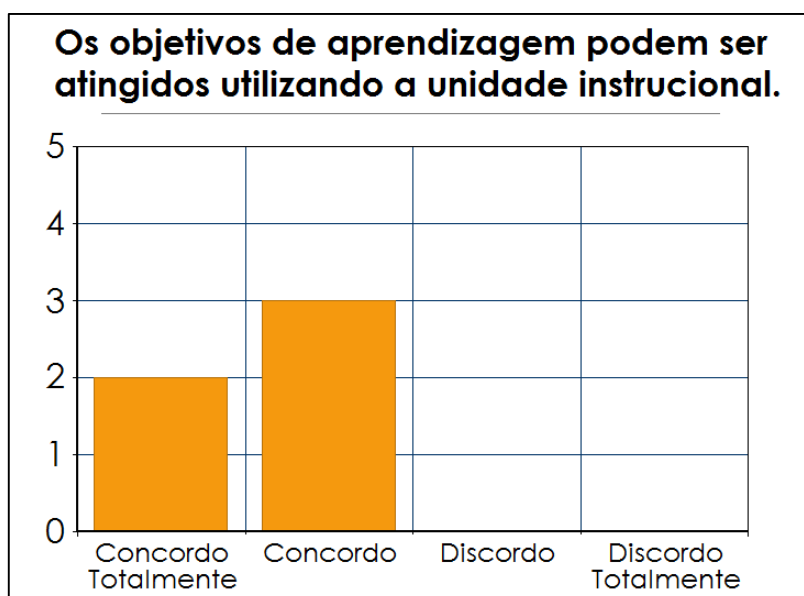


Figura 18: Os objetivos de aprendizagem podem ser atingidos utilizando a unidade instrucional.

Apesar disso, alguns professores comentaram que são muitos conceitos diferentes (como programação, computação física, comunicação sem fio, aplicativos móveis) apresentados e explicados em apenas uma aula e, por isso, não assinalaram a opção “Concordo Totalmente”.

A segunda pergunta foi sobre o material desenvolvido. A maioria dos avaliadores acredita que o material criado e disponibilizado da unidade instrucional é completo (figura 19).

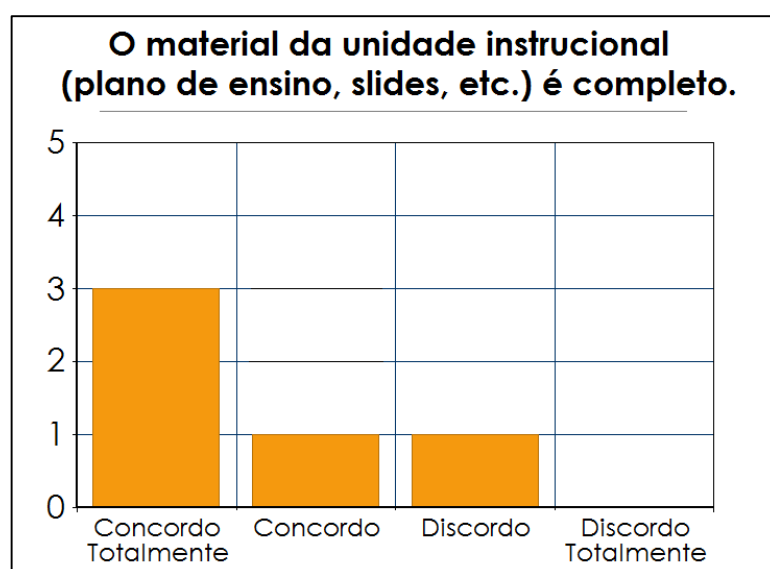


Figura 19: O material da unidade instrucional é completo.

Como pode ser visto na imagem anterior (figura 19), um avaliador marcou a opção “Discordo” para este quesito. Segundo ele, como o material é para ser utilizado por qualquer professor de informática, faltou um roteiro específico para a construção física do robô, explicando detalhadamente o passo-a-passo da sua elaboração.

Além disso, foi sugerida a criação de um informativo exclusivo para os professores, que funcionaria como um guia da unidade instrucional, indicando para o que serve cada material disponibilizado.

5.3.2 Dificuldade em Aplicar a UI

O segundo Objetivo de Medição é “avaliar a aplicabilidade da unidade instrucional em turmas do ensino fundamental (entre o sexto e nono ano) e do ensino médio, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional”. Para isso foram elaboradas três perguntas relacionadas ao nível de dificuldade em aplicar a UI.

A primeira pergunta foi sobre o nível de dificuldade que os alunos de sexto a nono ano do ensino fundamental teriam em usar e entender o conteúdo da UI. Todos os professores consideraram que esse nível seria baixo (figura 20).

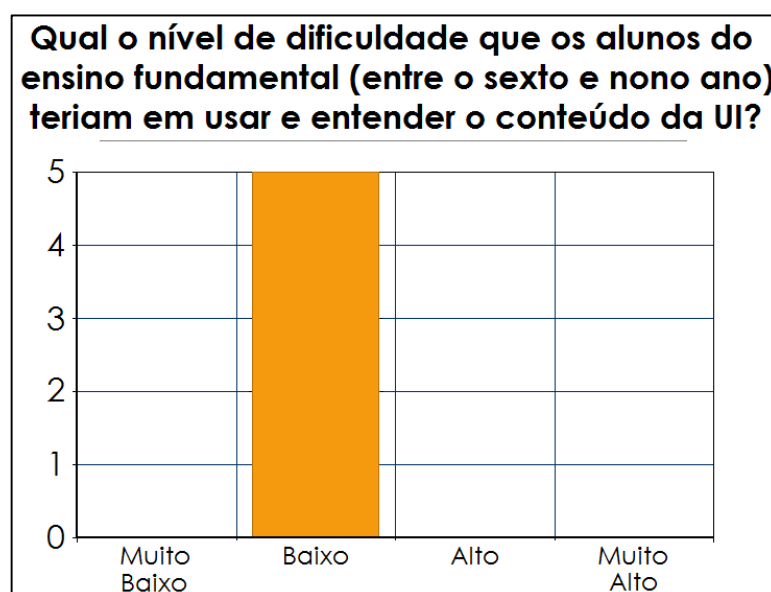


Figura 20: Qual o nível de dificuldade que os alunos do ensino fundamental (entre o sexto e nono ano) teriam em usar e entender o conteúdo da unidade instrucional?

Apesar da unidade instrucional abordar elementos e um conteúdo que os alunos não lidam em sala de aula, todos os professores acreditaram que os alunos não teriam dificuldade em entender o conteúdo abordado pela UI e muito menos teriam dificuldade em utilizá-la. Alguns professores observaram que o uso de cores

diferentes em tipos de blocos diferentes utilizados na programação pelo App Inventor facilitaria bastante o entendimento dos alunos sobre os algoritmos criados.

A segunda pergunta, assim como a primeira, também abordou o nível de dificuldade que os alunos teriam em utilizar e entender o conteúdo da unidade instrucional. Porém esta pergunta foi voltada especificamente para os alunos de ensino médio. Assim como no tópico anterior, todos os professores consideraram que os alunos não teriam dificuldade (figura 21).

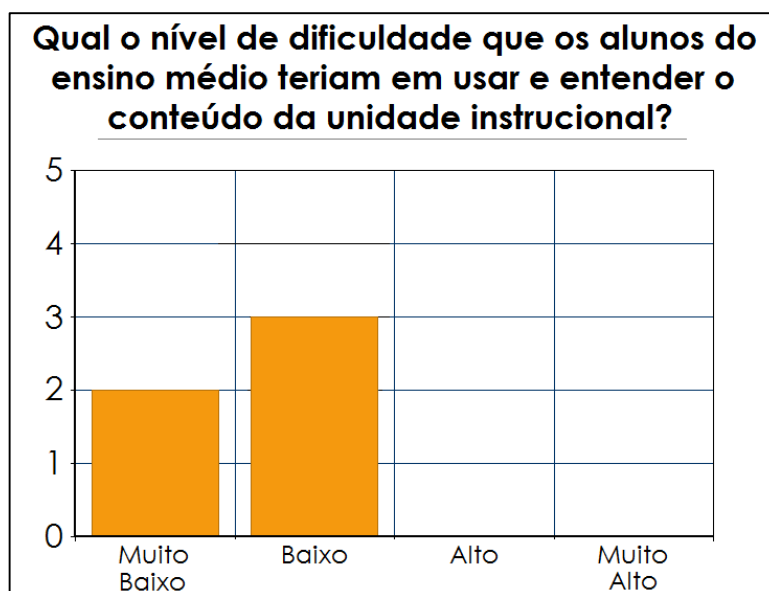


Figura 21: Qual o nível de dificuldade que os alunos do ensino médio teriam em usar e entender o conteúdo da unidade instrucional?

Alguns professores citaram que os alunos do ensino médio estão mais acostumados com o uso de celular e mais atentos a novidades tecnológicas e, por isso, teriam mais facilidade em utilizar e entender o material do que os alunos do ensino fundamental.

A última pergunta sobre a aplicabilidade da unidade instrucional foi voltada para os professores. Eles foram questionados sobre o nível de dificuldade que o professor

teria em aplicar a UI e as respostas foram equilibradas, como pode ser visto na figura 22).



Figura 22: Qual o nível de dificuldade que o professor teria em aplicar a unidade instrucional em sala de aula?

Foi comentado que o nível de dificuldade só poderia ser avaliado diante da apresentação do ambiente ao qual a UI seria aplicado. Por exemplo, em uma sala de aula com 25 alunos (situação citada por um avaliador) a turma teria que ser dividida, pois o professor não teria condições de sanar todas as dúvidas e auxiliar à todos na parte prática da aula. Um professor comentou que “seria difícil ter controle sobre todos os equipamentos em uma turma grande”, então, em um caso como este, o nível de dificuldade seria alto.

Além disso, segundo outro avaliador, um aspecto para ser analisado seria o grau de conhecimento dos professores com as ferramentas utilizadas durante o andamento da aula (email para acesso ao App Inventor e conhecimento sobre Arduino, por exemplo). O professor que for aplicar a UI teria que estudar o material antes de utilizá-lo.

5.3.3 Interesse pela UI

O último Objetivo de Medição é “avaliar o interesse em utilizar a unidade instrucional, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional.”. Para isso foram elaboradas três perguntas relacionadas ao nível de interesse dos participantes.

A primeira pergunta foi sobre o nível de interesse que os alunos de sexto a nono ano do ensino fundamental teriam em aprender computação utilizando a UI. Todos os professores consideraram que os alunos teriam um “Interesse Total” (figura 23).

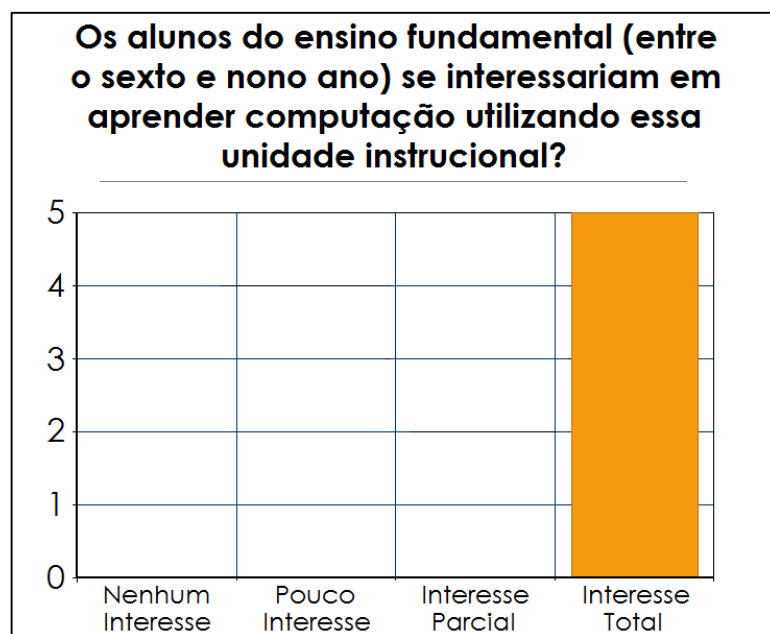


Figura 23: Os alunos do ensino fundamental (entre o sexto e nono ano) se interessariam em aprender computação utilizando essa unidade instrucional?

Assim como os alunos do ensino fundamental, os professores consideraram que os alunos do ensino médio também se interessariam em aprender computação utilizando essa unidade instrucional (figura 24), mas que estariam menos motivados à isso do que os mais novos.

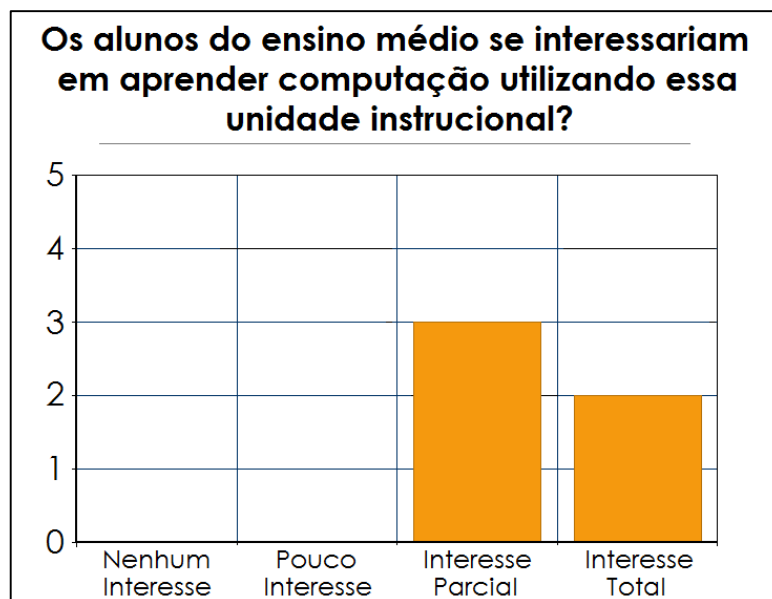


Figura 24: Os alunos do ensino médio se interessariam em aprender computação utilizando essa unidade instrucional?

Segundo os avaliadores, os alunos mais novos costumam ter maiores níveis de curiosidade então se interessariam mais em criar um aplicativo do que os alunos do ensino médio, por exemplo.

A última pergunta sobre o interesse pela unidade instrucional foi voltada para os professores. A questão foi: Os professores (de informática) se interessariam em utilizar essa unidade instrucional para o ensino de computação? As respostas podem ser visualizadas na figura 25.

Como pode ser observado todos os avaliadores marcaram que os professores teriam pouco interesse ou que, caso se interessassem em utilizar a UI, esse interesse seria parcial.

Os avaliadores comentaram que “os professores, em sua maioria, ‘fogem’ do uso da tecnologia em sala de aula, já que a mesma requer mudanças em sua forma de lecionar”.

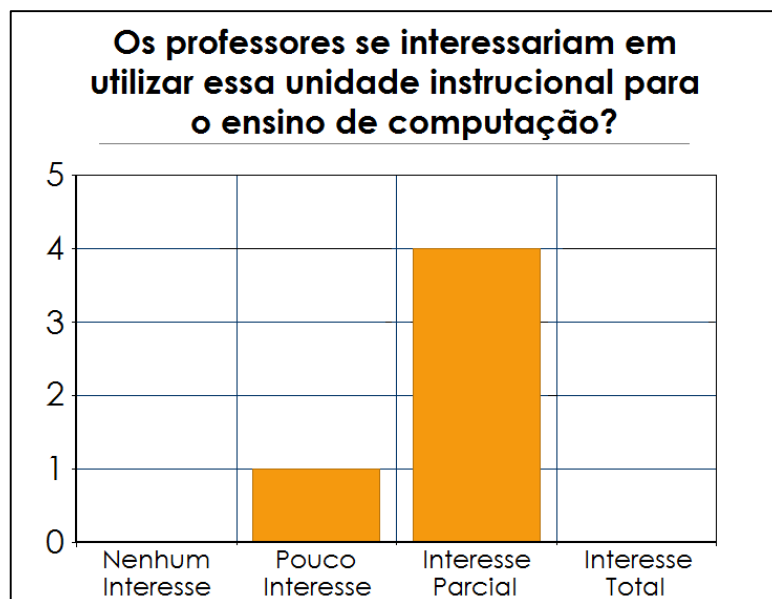


Figura 25: Os professores se interessariam em utilizar essa unidade instrucional para o ensino de computação?

Foi citado também que o “interesse do professor depende do nível de comprometimento do mesmo com a educação, já que esta UI é uma ótima ferramenta de aprendizado.”

5.3.4 Ameaças à Validade

Além dos resultados da avaliação da unidade instrucional, de maneira geral, podem ser considerados positivos e ter sido utilizada a abordagem GQM (BASILI et al., 1994) para guiar a medição, existem alguns fatores que ameaçam a validade dos resultados.

Se tratando da coleta de dados, para diminuir as possíveis ameaças à validade, além de ter sido utilizado uma abordagem de medição existente e padronizada, o questionário foi elaborado de uma maneira que fosse fácil o seu entendimento, utilizando uma linguagem adequada e um aspecto visual simplificado.

Obteve-se uma amostra baixa de (cinco) professores interessados em participar do projeto, apesar de alguns avaliadores não realizarem observações extras

e não expressarem suas opiniões sobre a unidade instrucional além das perguntas do questionário.

Para poder realizar uma análise melhor da unidade instrucional é preciso obter a avaliação de mais professores e, principalmente, aplicar a UI em um ambiente real. Apesar disso, foi possível obter um primeiro *feedback* referente à elaboração da unidade instrucional para o ensino de computação em escolas utilizando o App Inventor 2 e computação física no âmbito de uma pesquisa exploratória.

6. CONCLUSÃO

O objetivo deste trabalho de conclusão de curso foi o desenvolvimento de uma unidade instrucional para o ensino de computação em escolas de ensino fundamental e médio através da criação de aplicativos móveis e sua integração com computação física. Para isso, foi realizado um estudo sobre a fundamentação teórica de aprendizagem e ensino (incluindo o ensino do pensamento computacional, design instrucional e o App Inventor como ferramenta de ensino) e também sobre computação física.

Além da fundamentação teórica, foi realizado um levantamento sobre o estado da arte referente à utilização do App Inventor no ensino de computação em escolas. O levantamento mostrou a falta de unidades instrucionais que utilizam essa plataforma para o ensino com crianças e adolescentes.

Com essa análise foi definida e elaborada a unidade instrucional contendo um plano de ensino, um informativo sobre a UI, roteiros para os professores, slides para a apresentação em sala, tarefa de casa, prova para avaliação e uma rubrica de avaliação, além do código e executável do aplicativo criado através da UI.

Foi realizada uma avaliação da unidade instrucional por professores de informática de escolas de ensino fundamental e médio de Santa Catarina que apresentou resultados positivos, indicando que a UI tem capacidade de atingir os seu objetivo de ser utilizada como ferramenta para o ensino de computação em escolas. Foi entendido que, pela avaliação dos professores, a UI tende a funcionar melhor com os alunos do ensino fundamental.

O resultado do presente trabalho é a disponibilização da unidade instrucional para o ensino de computação utilizando o App Inventor e computação física com o

objetivo de ensinar crianças e adolescentes o básico sobre lógica de programação e computação.

Para trabalhos futuros pode-se realizar a aplicação da unidade instrucional em uma sala de aula, com um professor de informática, e analisar o uso efetivo da UI em um ambiente real. Além disso, é possível elaborar unidades instrucionais multidisciplinares para o ensino de outras matérias escolares utilizando o App Inventor como ferramenta base.

Desenvolvimento de uma Unidade Instrucional para o Ensino de Computação utilizando o App Inventor 2

Gabriel Soares Costa

Departamento de Informática e Estatística
Universidade Federal de Santa Catarina (UFSC) – Florianópolis, SC – Brazil

gabrielsoarescosta.gsc@gmail.com

***Abstract.** The world technology evolution is even more present in the routine of society. In this context, it is possible to observe that it has become quite necessary to have some computational knowledge. The problem is that currently are rare schools that have classes aimed to form computational thinking, as programming, or that offer courses in this area. In this situation, the creation of an instructional unit for computer education using App Inventor 2 (a web tool built by Google in partnership with the Massachusetts Institute of Technology that enables the creation of applications for mobile phones using Android operating system) aims to increase the development of computational thinking at elementary and high schools. To achieve the project's goal it was prepared, developed and made available an instructional unit consisting of guides, lecture slides, questionnaires and evaluations related to the creation of an app through the App Inventor 2 and its integration with physical computing. The project evaluation was performed by IT and robotics teachers from elementary and high schools located at Santa Catarina.*

***Resumo.** A evolução tecnológica no mundo está cada vez mais presente no dia a dia da sociedade. Nesse contexto é possível observar que se tornou bastante necessário possuir algum conhecimento computacional. O problema é que atualmente são raras as escolas que possuem na grade curricular aulas voltadas para a formação do pensamento computacional, como programação, ou que disponibilizam cursos nessa área. Com essa situação, o projeto da criação de uma unidade instrucional para o ensino de computação utilizando o App Inventor 2 (uma ferramenta Web construída pelo Google em parceria com o Massachusetts Institute of Technology que permite a criação de aplicativos para celulares que utilizam o sistema operacional Android) almeja um maior desenvolvimento do pensamento computacional no ensino fundamental e médio. Para atingir o objetivo do projeto foi elaborada, desenvolvida e disponibilizada uma unidade instrucional composta por roteiros, slides de aula, questionários e avaliações relacionadas à criação de um aplicativo através do App Inventor 2 e sua integração com computação física. A avaliação do projeto foi realizada por professores de informática e robótica do ensino fundamental e médio de escolas localizadas em Santa Catarina.*

1. Introdução

A evolução tecnológica mundial influencia cada vez mais no dia a dia das pessoas; a tecnologia da informação alterou as formas, processos e nosso estilo de vida (SOUSA, 2010). Um bom exemplo da rapidez dessa evolução é o fato de que o número de celulares no Brasil já ultrapassou o número de habitantes no país (ANATEL, 2013).

Esse crescimento no uso e evolução da tecnologia na sociedade vem promovendo diferentes mudanças no âmbito geral. Entre essas mudanças está a criação de novos equipamentos, hardwares e softwares, que necessitam de software contendo algoritmos cada vez mais aprimorados e complexos.

Apesar dessa crescente demanda, um estudo divulgado em 2015 (ABDUL-AIM, 2015) apresenta que, apesar do crescente número de empregos criados diariamente na área de ciências da computação, uma porção significativa de estudantes do ensino médio não saem preparados para entrar nesse mercado, pois não possuem aulas de computação em suas escolas.

Isso indica que não são muitas as escolas que ensinam programação computacional em seus períodos letivos, mesmo já tendo sido verificado que esse conhecimento auxilia também em outros aspectos do nosso cotidiano, como raciocínio lógico, inovação e criatividade (CONFAP, 2014).

Nesse sentido, o App Inventor (Google e MIT, 2009) é um software de programação online que serve para o desenvolvimento de aplicativos para plataformas mobile que utilizam o Android como sistema operacional. Experiências de cursos aplicando App Inventor, como o atualmente utilizado na Universidade de San Francisco, “Computação, Aplicativos Móveis e a Web” Wolber (2012), têm indicado que esse software é ideal para pessoas que nunca programaram e desejam desenvolver seus primeiros aplicativos. Sua interface gráfica auxilia os alunos que nunca tiveram contato com programação.

Desse modo, considerando a ausência de incentivos para o ensino e aprendizagem de programação, a elaboração de uma abordagem para o ensino de computação, através da criação de aplicativos para aparelhos móveis, voltada para a educação nos níveis fundamental e médio, se tornou uma opção atrativa e pertinente e foi a principal motivação para a realização desse projeto.

Para a elaboração desse material, a metodologia ADDIE de Design Instrucional aparenta ser ideal por ser uma metodologia utilizada na criação de conteúdo para o processo de ensino e aprendizagem. Utilizando esse método, torna-se atrativa a criação de uma unidade instrucional para o ensino de computação.

Além disso, projetos como o de GOMES et al (2014) demonstram que o uso de computação física para o ensino de computação com crianças e adolescentes traz um retorno positivo no desenvolvimento do pensamento computacional e no entendimento tecnológico. Sendo assim, é definida a integração da plataforma App Inventor com o uso de computação física.

2. Revisão da Literatura

A análise desse tópico foi realizada sistematicamente seguindo o método de revisão literária definido por Kitchenham (2004), adaptado para um trabalho de conclusão de curso de graduação.

A pergunta de pesquisa definida para a realização da revisão sistemática: Quais são e como são estruturadas as unidades instrucionais para ensinar pensamento computacional para alunos de ensino fundamental e médio utilizando o App Inventor?

As ferramentas utilizadas para a busca foram: Google Scholar (scholar.google.com.br), ACM Digital Library (dl.acm.org) e IEEE Xplore (ieeexplore.ieee.org). A revisão do estado da arte levantou indícios de que são poucos e recentes os trabalhos que desenvolvem uma unidade instrucional para o ensino do pensamento computacional utilizando o App Inventor 2, principalmente na língua portuguesa. Além disso, vários trabalhos encontrados utilizavam uma versão anterior da plataforma, com menos componentes disponíveis no software para a criação de aplicativos.

De forma geral, todos os trabalhos pecaram na divulgação dos resultados. O foco acabou ficando mais na opinião dos participantes sobre o projeto e não na evolução deles após a conclusão do trabalho.

Além disso notou-se que haviam poucas informações sobre o tipo de escolas e quais conteúdos foram elaborados pelos criadores dos projetos para a aplicação dos testes e verificação dos resultados.

Apesar das pesquisas terem sido realizadas tanto na língua nacional da publicação deste trabalho (português), como em inglês, isso não inibe o risco de que algum estudo possa ter sido omitido na execução da busca, devido à escolha dos termos de busca.

Vale ressaltar que para a realização da busca foram utilizados os termos “computação” e “ciência da computação” e suas respectivas traduções. O ideal seria realizar a busca com o termo “pensamento computacional”, seus sinônimos e traduções, porém o retorno da busca não trazia nenhum resultado e foi optado por alterar essas expressões.

3. Unidade Instrucional

Foi realizada uma análise do contexto atual sobre o ensino de computação em escolas e o design da unidade instrucional para o ensino de computação utilizando o App Inventor e computação física. Através dos objetivos de aprendizagem e ensino é elaborado o plano de ensino e, após isso, são apresentados os materiais desenvolvidos para a unidade instrucional.

3.1 Análise do Contexto

De acordo com o design instrucional, para se definir os objetivos de aprendizagem é necessário realizar uma análise do público alvo e do ambiente. Neste projeto, o público alvo são alunos do ensino fundamental (a partir do sexto ano) e médio; enquanto o ambiente escolhido são as escolas públicas e particulares do Brasil.

A unidade instrucional desenvolvida é projetada para o ensino de computação em escolas, logo ela só pode ser aplicada em instituições de ensino que possuam em sua infraestrutura um laboratório de informática ou alguma sala que disponha de computadores para a utilização em aula.

3.2 Design

Para a elaboração da unidade instrucional foram utilizados os objetivos de aprendizagem no ensino de computação estabelecidos pelo currículo referência CSTA/ACM K-12 para o nível

2. Após a análise de contexto, foram selecionados os objetivos de aprendizagem para a elaboração da unidade instrucional que abrangem os seguintes aspectos do ensino de computação: Pensamento Computacional, Colaboração, Práticas Computacionais e de Programação e Computadores e Dispositivos de Comunicação.

A duração da aplicação da unidade instrucional ficou estabelecido em quatro horas, que corresponde a um dia letivo dos alunos. Durante a aplicação serão apresentados os conceitos básicos de programação e do funcionamento do App Inventor. Por existir um número limitado de robôs, os alunos serão divididos em duplas ou trios. Baseado nisso, foi desenvolvido o Plano de Ensino da unidade instrucional.

O aplicativo criado é o “Controle do Robô”. Após a elaboração do aplicativo, os alunos poderão testar com seus respectivos robôs, através de uma conexão Bluetooth, e fazê-lo realizar as seguintes funções: piscar e levantar o braço. Além disso, foram definidos e apresentados os pré-requisitos para a aplicação da UI.

3.3 Desenvolvimento – Materiais da UI

Após a elaboração do plano de ensino foram desenvolvidos os seguintes materiais para a UI: informativo sobre a unidade instrucional; aplicativo “Controle do Robô” para a integração com computação física; roteiros para os professores; slides para a aplicação em aula; tarefa para ser realizada em casa; uma prova para avaliar o conteúdo aprendido; e rubrica para avaliação.

A Unidade Instrucional está disponibilizada sob a licença Creative Commons Atribuição-NãoComercial-CompartilhaIgual 4.0 Internacional e pode ser acessada no link: <<https://drive.google.com/drive/u/0/folders/0BzKHw8EKAYsLdWkwUHIJSGZSbEk>>.

Foram elaborados os seguintes materiais: informativo sobre a UI, o aplicativo a ser programado, roteiros para os professores, slides de aula, avaliação (com rubrica e gabarito) e tarefa de casa (também com gabarito).

4. Avaliação e Análise

Para a avaliação e análise foi utilizada a abordagem definida no método GQM (BASILI et al., 1994) com a intenção de elaborar um plano de medição para a realização da avaliação. Utilizando a metodologia GQM, inicialmente é necessário definir os objetivos para então estabelecer as perguntas que questionam estes objetivos. Para cada pergunta é definida uma medida relevante que apresente a sua resposta (BASILI et al., 1994).

Foram definidos os seguintes objetivos para a realização da avaliação desta unidade instrucional:

- Objetivo de Medição 1: Avaliar a qualidade do conteúdo da unidade instrucional, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional.
- Objetivo de Medição 2: Avaliar a aplicabilidade da unidade instrucional em turmas do ensino fundamental (entre o sexto e nono ano) e do ensino médio, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional.
- Objetivo de Medição 3: Avaliar o interesse em utilizar a unidade instrucional, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional.

Com o objetivo de manter a qualidade da pesquisa apenas professores que deram/dão aulas de informática em escolas de ensino médio e/ou fundamental e que possuem formação na

área de TI (como graduação em Ciências da Computação ou Sistemas de Informação) é que participaram da avaliação.

4.1 Conteúdo da UI

O primeiro Objetivo de Medição é “avaliar a qualidade do conteúdo da unidade instrucional, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional”. Para isso foram realizadas duas perguntas, com a resposta de cada pergunta sendo a sua medida.

A primeira pergunta foi sobre os objetivos de aprendizagem da UI. Todos os avaliadores acreditam que os objetivos de aprendizagem podem ser atingidos utilizando a unidade instrucional. A segunda pergunta foi sobre o material desenvolvido. A maioria dos avaliadores acredita que o material criado e disponibilizado da unidade instrucional é completo.

Apesar disso, foram sugeridas algumas alterações: um roteiro específico para a construção física do robô, explicando detalhadamente o passo-a-passo da sua elaboração; e a criação de um informativo exclusivo para os professores, que funcionaria como um guia da unidade instrucional, indicando para o que serve cada material disponibilizado.

4.2 Aplicabilidade da UI

O segundo Objetivo de Medição é “avaliar a aplicabilidade da unidade instrucional em turmas do ensino fundamental (entre o sexto e nono ano) e do ensino médio, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional”. Para isso foram elaboradas três perguntas relacionadas ao nível de dificuldade em aplicar a UI.

A primeira pergunta foi sobre o nível de dificuldade que os alunos de sexto a nono ano do ensino fundamental teriam em usar e entender o conteúdo da UI. Todos os professores consideraram que esse nível seria baixo.

Apesar da unidade instrucional abordar elementos e um conteúdo que os alunos não lidam em sala de aula, todos os professores acreditaram que os alunos não teriam dificuldade em entender o conteúdo abordado pela UI e muito menos teriam dificuldade em utilizá-la. Alguns professores observaram que o uso de cores diferentes em tipos de blocos diferentes utilizados na programação pelo App Inventor facilitaria bastante o entendimento dos alunos sobre os algoritmos criados.

A segunda pergunta, assim como a primeira, também abordou o nível de dificuldade que os alunos teriam em utilizar e entender o conteúdo da unidade instrucional. Porém esta pergunta foi voltada especificamente para os alunos de ensino médio. Assim como no tópico anterior, todos os professores consideraram que os alunos não teriam dificuldade.

Alguns professores citaram que os alunos do ensino médio estão mais acostumados com o uso de celular e mais atentos a novidades tecnológicas e, por isso, teriam mais facilidade em utilizar e entender o material do que os alunos do ensino fundamental.

A última pergunta sobre a aplicabilidade da unidade instrucional foi voltada para os professores. Eles foram questionados sobre o nível de dificuldade que o professor teria em aplicar a UI e as respostas foram equilibradas: foi comentado que o nível de dificuldade só poderia ser avaliado diante da apresentação do ambiente ao qual a UI seria aplicado. Por exemplo, em uma sala de aula com 25 alunos (situação citada por um avaliador) a turma teria que ser dividida, pois o professor não teria condições de sanar todas as dúvidas e auxiliar à todos na parte prática da aula. Um professor comentou que “seria difícil ter controle sobre todos

os equipamentos em uma turma grande”, então, em um caso como este, o nível de dificuldade seria alto.

Além disso, segundo outro avaliador, um aspecto para ser analisado seria o grau de conhecimento dos professores com as ferramentas utilizadas durante o andamento da aula (email para acesso ao App Inventor e conhecimento sobre Arduino, por exemplo). O professor que for aplicar a UI teria que estudar o material antes de utilizá-lo.

4.3 Interesse pela UI

O último Objetivo de Medição é “avaliar o interesse em utilizar a unidade instrucional, sob o ponto de vista de profissionais capacitados para a aplicação da unidade instrucional”. Para isso foram elaboradas três perguntas relacionadas ao nível de interesse dos participantes.

A primeira pergunta foi sobre o nível de interesse que os alunos de sexto a nono ano do ensino fundamental teriam em aprender computação utilizando a UI. Todos os professores consideraram que os alunos teriam um “Interesse Total”.

Assim como os alunos do ensino fundamental, os professores consideraram que os alunos do ensino médio também se interessariam em aprender computação utilizando essa unidade instrucional, mas que estariam menos motivados à isso do que os mais novos.

Segundo os avaliadores, os alunos mais novos costumam ter maiores níveis de curiosidade então se interessariam mais em criar um aplicativo do que os alunos do ensino médio, por exemplo.

A última pergunta sobre o interesse pela unidade instrucional foi voltada para os professores. A questão foi: Os professores (de informática) se interessariam em utilizar essa unidade instrucional para o ensino de computação? Todos os avaliadores marcaram que os professores teriam pouco interesse ou que, caso se interessassem em utilizar a UI, esse interesse seria parcial.

Os avaliadores comentaram que “os professores, em sua maioria, ‘fogem’ do uso da tecnologia em sala de aula, já que a mesma requer mudanças em sua forma de lecionar”. Foi citado também que o “interesse do professor depende do nível de comprometimento do mesmo com a educação, já que esta UI é uma ótima ferramenta de aprendizado.”

5. Principais Conclusões

Foi desenvolvida uma unidade instrucional para o ensino de computação em escolas de ensino fundamental e médio através da criação de aplicativos móveis e sua integração com computação física. Para isso, foi realizado um estudo sobre a fundamentação teórica de aprendizagem e ensino (incluindo o ensino do pensamento computacional, design instrucional e o App Inventor como ferramenta de ensino) e também sobre computação física.

Além da fundamentação teórica, foi realizado um levantamento sobre o estado da arte referente à utilização do App Inventor no ensino de computação em escolas. O levantamento mostrou a falta de unidades instrucionais que utilizam essa plataforma para o ensino com crianças e adolescentes.

Com essa análise foi definida e elaborada a unidade instrucional contendo um plano de ensino, um informativo sobre a UI, roteiros para os professores, slides para a apresentação em

sala, tarefa de casa, prova para avaliação e uma rubrica de avaliação, além do código e executável do aplicativo criado através da UI.

Foi realizada uma avaliação da unidade instrucional por professores de informática de escolas de ensino fundamental e médio de Santa Catarina que apresentou resultados positivos, indicando que a UI tem capacidade de atingir os seu objetivo de ser utilizada como ferramenta para o ensino de computação em escolas. Foi entendido que, pela avaliação dos professores, a UI tende a funcionar melhor com os alunos do ensino fundamental.

O resultado do presente trabalho é a disponibilização da unidade instrucional para o ensino de computação utilizando o App Inventor e computação física com o objetivo de ensinar crianças e adolescentes o básico sobre lógica de programação e computação.

Para trabalhos futuros pode-se realizar a aplicação da unidade instrucional em uma sala de aula, com um professor de informática, e analisar o uso efetivo da UI em um ambiente real. Além disso, é possível elaborar unidades instrucionais multidisciplinares para o ensino de outras matérias escolares utilizando o App Inventor como ferramenta base.

Referências

- SOUSA, K. Evolução da Tecnologia de Informação. 2010. Disponível em: <<http://www.webartigos.com/artigos/evolucao-da-tecnologia-de-informacao/47044/>>.
- ANATEL - Agência Nacional de Telecomunicações. Relatórios Consolidados - Indicadores 2013. 2013. Disponível em: <http://www.anatel.gov.br/dados/index.php?option=com_content&view=article&id=240&Itemid=483>.
- ABDUL-AIM, J. Students Not Prepared for Careers in Computer Science. Diverse – Issues in Higher Education. 2015. Disponível em: <<http://diverseeducation.com/article/77408/>>.
- CONFAP - Conselho Nacional das Fundações Estaduais de Amparo à Pesquisa. Linguagem de programação ajuda a aprimorar o raciocínio lógico de jovens. 2014. Disponível em: <<http://confap.org.br/news/linguagem-de-programacao-ajuda-a-aprimorar-o-raciocinio-logico-de-jovens/>>.
- WOLBER, D. App inventor and real-world motivation. Proceedings of the 42nd ACM technical symposium on Computer science education. ACM, 2011. pp 601-606.
- GOMES, O. S. M.; NÓBREGA, R.; ALMEIDA, T.; CÉSAR, J. P.; GUIMARÃES, R.; Projeto e desenvolvimento de um carro robô controlado por smartphone, utilizando a plataforma Amarino. ForScience – Revista Científica do IFMG, v. 2, n. 2, 2014.
- KITCHENHAM, B.A. Procedures for Performing Systematic Reviews. Tech. Report TR/SE-0401, Keele University. Inglaterra. 2004.
- CSTA. CSTA K–12 Computer Science Standards – Revised. 2011, ACM, New York/USA, 2011.
- BASIL, V; CALDIERA, G; ROMBACH, H. D. The Goal Question Metric Approach. 1994.

REFERÊNCIAS

ABDUL-AIM, J. **Students Not Prepared for Careers in Computer Science**. Diverse – Issues in Higher Education. 2015. Disponível em:

<<http://diverseeducation.com/article/77408/>>. Acesso em: 26 ago. 2015.

ABELSON, H. **App Inventor for Android**. 2009. Disponível em:

<<http://googleresearch.blogspot.com/2009/07/app-inventor-for-android.html>>. Acesso em: 25 jun. 2015.

ABRAHAMSSON, P.; SALO, O.; RONKAINEN, J.; WARSTA, J. **Agile software development methods: Review and analysis**. VTT Publications 478. Finlândia. 2002.

AHMAD, Y. Instructional Design and Motivation in Computer-Based Learning Environment. **Journal of Computer Engineering**, Department of Computer Science Salman Bin Abdulaziz University, Saudi Arabia, v. 8, pp. 9-12, 2013.

ANATEL - Agência Nacional de Telecomunicações. **Relatórios Consolidados - Indicadores 2013**. 2013. Disponível em:

<http://www.anatel.gov.br/dados/index.php?option=com_content&view=article&id=240&Itemid=483>. Acesso em: 16 jun. 2015.

AVILA, S.; RAMOS, N.; FREITAS, C.; COSTA, P. D. P.; TESTONI, V.; BORIN, J. F.; Ensino de Programação para Alunas de Ensino Médio: Relato de uma Experiência. **XXIII WORKSHOP SOBRE EDUCAÇÃO EM INFORMÁTICA (WEI)**, 2015, Recife. Anais do XXXV Congresso da Sociedade Brasileira de Computação (CSBC). Recife: CSBC, 2015.

BANZI, M. **Primeiros passos com o Arduino**. São Paulo: Novatec, 2011: p1.

Disponível em: <<http://www.novatec.net.br/livros/primeiros-passos-arduino/capitulo9788575222904.pdf>>. Acesso em: 21 ago. 2015.

BASILI, V; CALDIERA, G; ROMBACH, H. D. **The Goal Question Metric Approach**. 1994. Disponível em: <<http://www.cs.umd.edu/~mvz/handouts/gqm.pdf>>. Acesso em 14 out. 2016.

BERLAND, M.; LEE, V. R. Collaborative Strategic Board Games as a Site for Distributed Computation. **International Journal of Game-Based Learning (IJGBL)**, 1(2), pp. 65-81, 2011.

CONFAP - Conselho Nacional das Fundações Estaduais de Amparo à Pesquisa. **Linguagem de programação ajuda a aprimorar o raciocínio lógico de jovens**. 2014. Disponível em: <<http://confap.org.br/news/linguagem-de-programacao-ajuda-a-aprimorar-o-raciocinio-logico-de-jovens/>>. Acesso em: 22 jun. 2015.

COSTA, R. **O que define um dia letivo?**. 2009. Disponível em <<http://novaescola.org.br/conteudo/375/legislacao-define-dia-letivo>>. Acesso em: 02 out. 2016.

CSTA. **CSTA K–12 Computer Science Standards – Revised**. 2011, ACM, New York/USA, 2011. Disponível em: <http://csta.acm.org/Curriculum/sub/CurrFiles/CSTA_K-12_CSS.pdf>. Acesso em: 29 set. 2016.

DOS SANTOS, S. C. O processo de ensino-aprendizagem e a relação professor-aluno: Aplicação dos “sete princípios para a boa prática na educação de Ensino superior”. **Caderno de Pesquisas em Administração**, v. 8, n. 1, pp. 69-82, 2001. Disponível em: <<http://www.regeusp.com.br/arquivos/v08-1art07.pdf>>. Acesso em: 19 out. 2015

DOUGLAS, A. **ADDIE? Tell me more**. Australian Safety & Learning Systems. Brisbane. Disponível em: <<http://www.asls.com.au/addie-tell-me-more/>>. Acesso em: 19 out. 2015.

FILATRO, A. **Design instrucional contextualizado: educação e tecnologia**. São Paulo: Editora Senac São Paulo, 2004. 3ª edição.

FILATRO, A. **Planejamento, design, implementação e avaliação de programas de educação on-line**. 2007. Disponível em:
<http://www.escoladegoverno.pr.gov.br/arquivos/File/material_didatico_EaD/andrea_filatro_apostila.pdf>. Acesso em: 24 jun. 2015.

FIRMATA. **Firmata Protocol Documentation**. 2015. Disponível em:
<<https://github.com/firmata/protocol>>. Acesso em: 23 set. 2015.

FREIRE, K. **Design Instrucional: aplicabilidade dos desenhos pedagógicos na EAD On-line**. 2009. Disponível em:
<<http://www.abed.org.br/congresso2009/CD/trabalhos/1352009130007.pdf>>. Acesso em: 24 jun. 2015.

GOMES, T., MELO, J. **O Pensamento Computacional no Ensino Médio: Uma Abordagem Blended-Learning**. Anais do XXI Workshop sobre Educação em Computação – XXXIII Congresso da Sociedade Brasileira de Computação. Maceió, AL – Brasil. 2013. Disponível em:
<<http://www.lbd.dcc.ufmg.br/colecoes/wei/2013/0033.pdf>>. Acesso em 25 jun. 2015.

GOMES, T.; MELO, J. **Lógica de Programação: Iniciação Lúdica com App Inventor for Android**. 2013. Disponível em:
<<http://www.journals.ufrpe.br/index.php/eripe/article/view/378/312>>. Acesso em: 25 jun. 2015.

GOMES, O. S. M.; NÓBREGA, R.; ALMEIDA, T.; CÉSAR, J. P.; GUIMARÃES, R.; **Projeto e desenvolvimento de um carro robô controlado por smartphone, utilizando a plataforma Amarino**. ForScience – Revista Científica do IFMG, v. 2, n. 2, 2014. Disponível em:
<<http://www.formiga.ifmg.edu.br/forscience/index.php/forscience/article/view/121/81>>. Acesso em: 27 out. 2015.

GONÇALVES, S. **Teorias da aprendizagem**. Coimbra: McGraw Hill, 1993.

GOOGLE. **Criar uma Conta do Google**. 2015. Disponível em:
<<https://support.google.com/accounts/answer/27441?source=gsearch&hl=pt-BR>>.
Acesso em: 15 de nov. 2015.

GUZDIAL, M. Education: Paving the way for computational thinking.
Communications of the ACM, vol. 51, pp. 25-27, 2008.

HARDESTY, L. **The MIT roots of Google's new software**. MIT News Office. 2010,
p. 2. Disponível em: <<http://news.mit.edu/2010/android-abelson-0819>>. Acesso em:
01 de nov. 2015.

HSU, Y.-C.; RICE, K.; DAWLEY, L. Empowering educators with Google's Android
App Inventor: An online workshop in mobile app design. **British Journal of
Educational Technology**, 43, pp. 1-5, 2012. Disponível em:
<<http://onlinelibrary.wiley.com/doi/10.1111/j.1467-8535.2011.01241.x/full>>. Acesso
em: 21 ago. 2015

IBGE. **Instituto Brasileiro de Geografia e Estatística**. 2014.

IDC Brasil. **Estudo da IDC Brasil mostra recorde nas vendas de smartphones no
terceiro trimestre de 2014**. 2014. Disponível em:
<<http://www.idcbrasil.com.br/releases/news.aspx?id=1777>>. Acesso em: 03 de out.
2016.

INCoD – Instituto Nacional para Convergência Digital. **Iniciativa Computação na
Escola**. 2014. Disponível em: <<http://www.computacaonaescola.ufsc.br/>>. Acesso
em: 12 de nov. 2015.

INEP. **Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira**. 2014.

JIMENEZ, Y.; GARDNER-MCCUNE, C. Using App inventor & history as a gateway to engage African American students in computer science. **2015 Research In Equity And Sustained Participation In Engineering, Computing, And Technology (respect)**, [s.l.], ago. 2015. Institute of Electrical & Electronics Engineers (IEEE). DOI: 10.1109/respect.2015.7296512.

KAZIMOGLU, C; KIERNAN, M; BACON, L; MACKINNON, L. Learning programming at the computational thinking level via digital game-play. **Proceedings of the International Conference on Computational Science**, v. 9, pp. 522-531, 2012.

KITCHENHAM, B.A. **Procedures for Performing Systematic Reviews**. Tech. Report TR/SE-0401, Keele University. Inglaterra. 2004.

LEMOS, M. Arduino: **Conheça esta plataforma de hardware livre e suas aplicações**. 2014. Disponível em: <<http://imasters.com.br/desenvolvimento/arduino-conheca-esta-plataformade-hardware-livre-e-suas-aplicacoes/>>. Acesso em: 20 out. 2015.

LINO, J. I. **Proposta de um Jogo Educacional para a Área de Medição e Análise de Software**. Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Santa Catarina, Florianópolis, 2007. Disponível em: <http://www.gqs.ufsc.br/wp-content/uploads/2011/11/2007_Juliana_Izabel_Lino.pdf>. Acesso em: 02 nov. 2015.

MIT - Massachusetts Institute of Technology. **MIT App Inventor 2**. 2013. Disponível em: <<http://appinventor.mit.edu/explore/>>. Acesso em: 19 jun. 2015.

MOLENDÁ, M. **In Search of the Elusive ADDIE Model**. 5. ed. Indiana, Eua: Performance Improvement – Volume 42, 2003.

PERKOVIĆ, L.; SETTLE, A.; HWANG, S.; JONES, J. A framework for computational thinking across the curriculum. **Proceedings of the 15th Annual Conference on Innovation and Technology in Computer Science education**. Bilkent, Ankara, Turkey, pp. 123-127, 2010.

ROSCOE, J. F.; FEARN, S.; POSEY, E. Teaching Computational Thinking by Playing Games and Building Robots. **2014 International Conference On Interactive Technologies And Games**, [s.l.], p.9-12, out. 2014. Institute of Electrical & Electronics Engineers (IEEE). DOI: 10.1109/itag.2014.15.

SARTORI, A. S.; ROESLER, J. **Educação superior a distância: gestão da aprendizagem e da produção de materiais didáticos impressos e on-line**. Tubarão: Ed. Unisul, 2005.

SMITH, P.L.; RAGAN, T.J. **Instructional Design**. 2ª edição, Toronto: John Wiley & Sons, 1999.

SOUSA, K. **Evolução da Tecnologia de Informação**. 2010. Disponível em: <<http://www.webartigos.com/artigos/evolucao-da-tecnologia-de-informacao/47044/>>. Acesso em: 22 jun. 2015.

STEINER, H-C. Firmata: Towards making microcontrollers act like extensions of the computer. **New Interfaces for Musical Expression**. 2009. pp. 125-130. Disponível em <https://lagunak.gisa-elkartea.org/attachments/download/74/firmata_porque.pdf>. Acesso em: 23 set. 2015.

WING, J. M. Computational thinking. **Communications of the ACM**, vol. 49, pp. 33-35, 2006.

WOLBER, D. App inventor and real-world motivation. **Proceedings of the 42nd ACM technical symposium on Computer science education**. ACM, 2011. pp 601-606.

WOLBER, D. **AppInventor.org**. 2012. Disponível em:
<<http://www.appinventor.org/course-in-a-box>>. Acesso em 25 jun. 2015.