

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

Guilherme Trilha Daniel

**DESIGN DE UNIDADE INSTRUCIONAL DE DESENVOLVIMENTO DE
APLICATIVOS PARA O ENSINO FUNDAMENTAL**

Florianópolis

2016/1

Guilherme Trilha Daniel

**DESIGN DE UNIDADE INSTRUCIONAL DE DESENVOLVIMENTO DE
APLICATIVOS PARA O ENSINO FUNDAMENTAL**

Trabalho de Conclusão do Curso de Graduação em Ciências da Computação, do Departamento de Informática e Estatística, do Centro Tecnológico da Universidade Federal de Santa Catarina, requisito parcial à obtenção do título de Bacharel em Ciências da Computação.

Orientador: Prof. Dr. rer. nat. Christiane A. Gresse von Wangenheim, PMP.

Florianópolis

2016/1

Guilherme Trilha Daniel

**DESIGN DE UNIDADE INSTRUCIONAL DE DESENVOLVIMENTO DE
APLICATIVOS PARA O ENSINO FUNDAMENTAL**

Trabalho de conclusão de curso submetido ao Departamento de Informática e Estatística da Universidade Federal de Santa Catarina para a obtenção do Grau de Bacharelado em Ciências da Computação.

Florianópolis, 22 de junho de 2016.

Orientadora:

Prof.^a Dr.^a rer. nat. Christiane Gresse von Wangenheim, PMP
Orientador
Universidade Federal de Santa Catarina

Banca Examinadora:

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

Prof. Dr.^a Patrícia Della Méa Plentz
Avaliador
Universidade Federal de Santa Catarina

AGRADECIMENTOS

Deixo aqui meus agradecimentos:

A minha família e amigos, por sempre proporcionarem o melhor ambiente para meus estudos e fornecerem apoio em todos os momentos necessários.

A Professora Christiane pela orientação, atenção e paciência durante todo o período de desenvolvimento deste trabalho.

Ao Victor Abreu e a Thais Moratelli pelo auxílio na parte do design gráfico dos materiais desenvolvidos para a unidade instrucional.

A Professora Giselle Araújo, por toda a ajuda e interesse em tornar possível as aplicações da unidade instrucional na Escola Básica Prof.^a Herondina Medeiros Zeferino.

RESUMO

Com o grande desenvolvimento tecnológico dos últimos anos, a computação tornou-se essencial na sociedade, tanto no mercado de trabalho como na vida dos cidadãos, entretanto o conhecimento e o interesse por esta área não acompanharam este crescimento. Um dos fatores para isto, é a pouca importância dada ao ensino de computação durante as fases iniciais nas escolas. Deste modo, esse trabalho tem como objetivo de elaborar uma unidade instrucional para ensinar computação por meio do desenvolvimento de aplicativos utilizando o ambiente de desenvolvimento App Inventor. O público alvo desta unidade instrucional são alunos do Ensino Fundamental II com o objetivo de ensinar conceitos de programação e pensamento computacional e motivá-los/interessa-los para a área da computação.

O trabalho realizado incluí a análise da literatura referente ao processo de ensino e aprendizagem, ensino de computação e desenvolvimento de aplicativos para dispositivos móveis. É levantado o estado da arte por meio de uma revisão sistemática da literatura referente a unidades instrucionais que ensinam computação por meio do App Inventor. A partir destas atividades é elaborada uma unidade instrucional para ensino de computação utilizando o App Inventor. A unidade desenvolvida é aplicada na pratica com o intuito de obter informações sobre a sua qualidade através de um processo de avaliação.

Os resultados do presente trabalho incluem uma unidade instrucional para ensinar computação no Ensino Fundamental II e dados em relação a sua avaliação na prática. Espera-se que desta maneira os resultados forneçam um suporte ao Ensino Fundamental para ensinar computação de forma divertida e efetiva, contribuindo para a popularização da área de computação, e também, para tornar estes cidadãos mais competitivos e produtivos nas suas áreas de atuação futuramente.

Palavras-chave: Design instrucional, computação, programação, ensino fundamental, App Inventor.

LISTA DE FIGURAS

Figura 1 – Visão geral do Modelo ADDIE Adaptado (BRANCH, 2009).....	17
Figura 2 - Níveis de Ensino de Computação (CSTA, 2011)	23
Figura 3 - Áreas de conhecimento de Computação (CSTA, 2011).....	23
Figura 4 - Catarinenses com mais de dez anos com celular pessoal (IBGE, 2013)	27
Figura 5 - Crianças de 10 a 14 anos que têm celular e utilizam a internet (IBGE, 2013)	27
Figura 6 - Fatia de mercado das vendas por sistema operacional 2014 (GARTNER, 2015).....	28
Figura 7 - Número de aplicativos móveis por mês na Google Play (STATISTA, 2015)29	
Figura 8 - Android Studio (Android Developer, 2015).....	30
Figura 9 – Screenshot do Aplicativo Enigma Machine Simulator	31
Figura 10 - – Janela Designer do aplicativo Enigma Machine Simulator no App Inventor	31
Figura 11 – Screenshot do Aplicativo Goalai2.....	32
Figura 12 – Janela Editor de Blocos do aplicavo Goalai2 no App Inventor	33
Figura 13 –Parte do Roteiro Caça Mosquito	52
Figura 14 - Imagem do Aplicativo Caça Mosquito	54
Figura 15 - Exemplos dos <i>slides</i>	55
Figura 16 – Parte da Tarefa de casa.....	55
Figura 17 – Parte da Prova Final	56
Figura 18 – Parte da Rubrica	57
Figura 19 – Parte do Questionário pré unidade	57
Figura 20 - Parte do Informativo	58
Figura 21 - Fotos das aplicações realizadas.....	61
Figura 22 - Eu consigo fazer aplicativos para <i>smartphones</i>	65
Figura 23 - Consigo explicar para um amigo/amiga como fazer um aplicativo para <i>smartphones</i>	65
Figura 24 - A aula foi fácil?.....	66
Figura 25 - A aula foi divertida?	67
Figura 26 - A aula foi boa?.....	68
Figura 27 - O tempo das aulas passou.	68
Figura 28 - Fazer um aplicativo para <i>smartphone</i> é.	70
Figura 29 - Quero aprender mais sobre como fazer aplicativos para <i>smartphones</i>	70

LISTA DE TABELAS

Tabela 1- Níveis de aprendizagem do domínio cognitivo da taxonomia de Bloom (BLOOM, 1956).....	18
Tabela 2 - Níveis de aprendizagem do domínio afetivo da taxonomia de Bloom (Bloom, 1956).....	19
Tabela 3 - Níveis de aprendizagem do domínio psicomotor da taxonomia de Bloom (Bloom, 1956).....	19
Tabela 4 - Categorias de estratégias instrucionais (SASKATCHEWAN EDUCATION, 1991).....	20
Tabela 5 - Objetivos de aprendizagem para o nível 2 (CSTA, 2011)	26
Tabela 6 - Categorias de blocos de código do Editor de Blocos.	34
Tabela 7 - Componentes do Designer.....	35
Tabela 8 - Sinônimos e traduções dos termos de busca.	37
Tabela 9 - Termos usados na execução da busca no Google Scholar.	38
Tabela 10 - Identificadores dos artigos considerados.....	39
Tabela 11 - Contexto de Aplicação das UIs.	40
Tabela 12 - Definição Instrucional das UIs.	41
Tabela 13 - Tipos de estudos empíricos (GRESSE von WANGEHEIM & Shull, 2009).	42
Tabela 14 - Avaliação das UIs.....	43
Tabela 15 - Objetivos de Aprendizagem da Unidade Instrucional.....	48
Tabela 16 - Plano de Ensino	51
Tabela 17 - Pré-requisitos para aplicação da unidade instrucional	52
Tabela 18 - Categorias de blocos de código utilizados no jogo Caça Mosquito	53
Tabela 19 – Componentes de Interface utilizados no jogo Caça Mosquito	54
Tabela 20 - Plano de medição da unidade instrucional	60
Tabela 21 - Resumo dos participantes e dados coletados.....	60
Tabela 22 - Resumo dos dados coletados	62
Tabela 23 - Análise do atingimento dos objetivos.	64
Tabela 24 - Respostas discursivas dos alunos no questionário pós-unidade referente à facilidade da aula.	67
Tabela 25 - Respostas discursivas dos alunos no questionário pós-unidade referente à experiência.....	69

LISTA DE ABREVIATURAS E SIGLAS

- ACM – *Association for Computing Machinery*
CEPSH – Comitê de ética em Pesquisa com os Seres Humanos
CSTA – *Computer Science Teachers Association*
IBGE – Instituto Brasileiro de Geografia e Estatística
IDE – Ambiente de Desenvolvimento Integrado
IEEE – Instituto de Engenheiros Eletricistas e Eletrônicos
MEC – Ministério da Educação
MIT – Instituto de Tecnologia de Massachusetts
PCN – Parâmetros Curriculares Nacionais
SBC – Sociedade Brasileira de Computação
TI – Tecnologia da Informação.
UI– Unidade Instrucional
USB – *Universal Serial Bus*
XML – *eXtensible Markup Language*

SUMÁRIO

1 INTRODUÇÃO	11
1.1 Contextualização	11
1.2 Objetivos.....	13
1.3 Metodologia de Pesquisa e Trabalho.....	14
1.4 Estrutura do Documento	15
2 FUNDAMENTAÇÃO TEÓRICA	16
2.1 Ensino e Aprendizagem.....	16
2.2 Ensino de computação no Ensino Fundamental II	21
2.3 Desenvolvimento de apps móveis com App Inventor	26
3 ESTADO DA ARTE	37
3.1 Definição do protocolo de revisão.....	37
3.2 Execução da Busca	38
3.3 Extração das informações e análise dos resultados	39
3.4 Discussão	43
3.4.1 Ameaças à Validade da Revisão da Literatura	45
4 UNIDADE INSTRUCCIONAL “DESENVOLVIMENTO DE APPS COM APP INVENTOR”	46
4.1 Análise do contexto e dos aprendizes.....	46
4.2 Design da unidade instrucional.....	47
4.3 Desenvolvimento da unidade instrucional.....	52
5 APLICAÇÃO E AVALIAÇÃO	59
5.1 Definição da avaliação.....	59
5.2 Aplicação	60
5.3 Análise da avaliação	62
5.3.1 Os objetivos de aprendizagem são atingidos através da unidade instrucional?.....	63
5.3.2 A unidade instrucional facilita a aprendizagem?.....	65
5.3.3 A unidade instrucional promove uma experiência de aprendizagem agradável e divertida?.....	67
5.3.4 A unidade instrucional proporciona uma percepção positiva da computação?.....	69

5.3.5 Ameaças à validade	71
6 CONCLUSÃO.....	72
ANEXO I – ARTIGO PRODUZIDO COM BASE NO TCC ‘Ensinando a Computação por meio de Programação com App Inventor’	73
7 REFERÊNCIAS.....	79

1 INTRODUÇÃO

1.1 Contextualização

Com o passar das últimas décadas, a computação e os computadores transformaram profundamente o mundo (MISA, 2007). Atrelado a estas mudanças, as ciências da computação e as tecnologias advindas delas tem importância fundamental na economia mundial e no modo de vida. Sendo assim, todos os cidadãos deveriam possuir um claro entendimento sobre os princípios e práticas das ciências da computação, podendo tornar-se criadores de novos sistemas computacionais para melhorar a vida das pessoas, ou simplesmente consumidores mais competitivos e produtivos (CSTA, 2011).

A computação é uma área de conhecimento, caracterizada por atividades que incluem o projeto e o desenvolvimento de hardware e software, os quais, possuem uma grande variedade de propósitos (ACM, 2005). Além disso, outras atividades são caracterizadas como parte da computação, entre elas, processar, estruturar e manipular vários tipos de informação, realizar estudos científicos utilizando de computadores (ACM, 2005).

Deste modo, a área de computação abrange desde estudos teóricos de algoritmos, o pensamento computacional e até problemas práticos de implementação/programação em termos de computação em hardware e software

O Ensino Superior de computação já é institucionalizado nos níveis de graduação e pós-graduação. Entretanto, apesar do rápido crescimento do uso de tecnologias computacionais, o ensino de computação no Ensino Fundamental e médio ainda não recebe tanta atenção quanto outras disciplinas clássicas (PCN, 2015). Junto a isso, os conteúdos ensinados nem sempre são os mais adequados, e por vezes, é ensinado o uso de ferramentas como editores de texto, de imagens e outros aplicativos (compreendido como *IT literacy* ou alfabetização digital), o que não é o ideal e nem contempla os conhecimentos necessários para aprender computação. Ao final é necessário o ensino da proficiência digital (*IT fluency*) possibilitando os alunos a capacidade de aprender e aplicar novas tecnologias de forma produtiva ao longo da sua vida (CSTA, 2011).

Conceitos como a representação de dados, criação de algoritmos e análise, examinação de diferentes algoritmos baseando-se em outras estratégias de solução, os quais podem ser resumidos como o pensamento computacional, programação, colaboração, computadores e dispositivos de comunicação e os seus impactos éticos e globais são conceitos os quais devem ser ensinados a todos os alunos no ensino fundamental (CSTA, 2011). Estes conceitos são importantes porque, atualmente, todos

os estudantes possuirão uma vida altamente influenciada pela computação, e entre eles muitos trabalharão em áreas diretamente envolvidas com computação (CSTA, 2011).

Esta falta de compromisso com o ensino na área de computação por parte das escolas (PCN, 2015) pode acabar não despertando a vontade de muitos estudantes em escolher esta área de atuação. Assim como, existem casos em que o estudante não sabe se quer o que é ensinado em um curso na área de computação. Isto acaba gerando um grande problema para o mercado de trabalho, o qual vem sofrendo constantemente com a falta de mão de obra especializada. De acordo com um estudo realizado pela Softex, o mercado brasileiro de Tecnologia da Informação terá um déficit de 400 mil profissionais até 2022 (OLHAR DIGITAL, 2015). Além do impacto negativo na vida de cidadãos os quais necessitam deste conhecimento nas suas vidas (ARAÚJO, 2014).

Sendo assim, observa-se que é de grande importância incluir o ensino de computação já no ensino fundamental. Entretanto, existem poucos materiais, planos de ensino, ou uma unidade instrucional bem estruturada para ser utilizada neste processo de ensino. E também, levando em consideração que atualmente as escolas não possuem professores de computação (não em grande escala) (TIC EDUCAÇÃO, 2014). Logo quem irá lecionar essa unidade seria, por exemplo, um professor de história o qual não necessariamente possui conhecimento sobre computação (TIC EDUCAÇÃO, 2014).

De formar geral os alunos do ensino fundamental possuem um bom conhecimento da interação com as tecnologias existentes como smartphones, tablets e computadores. Isto acaba facilitando as diversas iniciativas existentes focadas no ensino de computação para crianças no mundo inteiro. Entre elas, na Semana da Educação em Ciência da Computação de 2013 foi aplicada a Hora do Código, a qual atingiu 15 milhões de estudantes no mundo inteiro e teve como objetivo desmistificar e mostrar que qualquer pessoa pode aprender os fundamentos básicos desta ciência (HOUR OF CODE, 2013). Dentre essas iniciativas, existe a possibilidade de ensinar computação utilizando dispositivos móveis, já que a população em geral possui uma maior intimidade com os smartphones do que com os computadores (O ESTADO DE S. PAULO, 2015). Pode-se aproveitar dessa boa integração para utilizá-la como uma ferramenta de ensino para aprender a habilidade de programação. E isso pode ser feito utilizando aplicativos que oferecem exercícios de programação ou até mesmo desenvolvendo os próprios aplicativos.

Uma das alternativas é a ferramenta App Inventor (APP INVENTOR, 2014), desenvolvida no Instituto de Tecnologia de Massachusetts (MIT) para proporcionar aos estudantes a experiência de desenvolver aplicativos para os seus próprios celulares. O App Inventor é um ambiente de desenvolvimento que utiliza programação visual em blocos e permite qualquer pessoa, até mesmo crianças, começar a programar e construir

aplicativos completos para dispositivos Android. Novatos no App Inventor conseguem ter aplicações prontas e funcionando em uma hora ou até menos. Normalmente, utilizando esses tipos de ambientes os estudantes conseguem desenvolver aplicações mais complexas e em menos tempo, quando comparado a linguagens de programação tradicionais baseadas em texto (APP INVENTOR, 2014).

Apesar da existência da ferramenta App Inventor, a maioria dos materiais didáticos são tutoriais para serem feitos sozinhos e estão na língua inglesa. Nota-se a falta de unidades instrucionais para serem aplicadas por um instrutor. Com materiais de apoio ao ensino, como *slides*, tarefas de casa, e até materiais para a avaliação dos alunos.

Assim identifica-se a necessidade de unidades instrucionais eficientes e eficazes (PAQUETTE, 2002) voltadas ao ensino de computação por meio de desenvolvimento de aplicativos no ensino fundamental.

1.2 Objetivos

Objetivo Geral. O objetivo do trabalho é o design instrucional de uma unidade instrucional (UI) para ensinar a programação de aplicativos em celulares Android com a ferramenta App Inventor, disponibilizado pelo MIT, alinhado ao currículo de referência (CSTA, 2011) no Ensino Fundamental II (para crianças de 11 aos 14 anos de idade).

Objetivos Específicos. Os objetivos específicos são:

- O1. Sintetizar a fundamentação teórica sobre aprendizagem & ensino de computação no ensino fundamental II e desenvolvimento de aplicativos.
- O2. Analisar o estado da arte sobre como o desenvolvimento de aplicativos é atualmente ensinada no ensino fundamental II.
- O3. Elaborar uma unidade instrucional para ensinar computação por meio de desenvolvimento de aplicativos para crianças no ensino fundamental II.
- O4. Aplicar e avaliar a unidade instrucional criada na prática.

Delimitação

O projeto é realizado de acordo com o regulamento vigente do Departamento de Informática e Estatística (INE – UFSC) em relação aos Trabalhos de Conclusão de Curso. O presente trabalho tem como foco somente o design instrucional de uma unidade instrucional para ensinar computação por meio de desenvolvimento de

aplicativos com o App Inventor, outros tipos de instruções como desenvolvimento de jogos no computador/robótica e etc. estão fora do contexto deste projeto. Também não são considerados outros ambientes de ensino de computação para crianças.

1.3 Metodologia de Pesquisa e Trabalho

A metodologia de pesquisa utilizada neste trabalho é dividida em quatro etapas.

Etapa 1. Síntese da literatura referente ao ensino e aprendizagem, assim como computação em dispositivos móveis e aplicativos. Esta primeira etapa será dividida em duas atividades:

A1.1 Realizar uma análise teórica sobre aprendizagem e ensino, e o design instrucional e conceitos básicos referente ao ensino fundamental, as metodologias de ensino e aprendizagem e seu estado da arte no ensino fundamental.

A1.2 Realizar uma análise teórica sobre computação móvel e desenvolvimento de aplicativos, junto com uma análise sobre o ensino de desenvolvimento de aplicativos no ensino fundamental e o App Inventor.

Etapa 2. Análise do estado da arte por meio de uma revisão sistemática de unidades instrucionais/estratégias de ensino atualmente sendo utilizadas e voltadas ao ensino de desenvolvimento de aplicativos. Nesta etapa é realizada uma revisão sistemática da literatura seguindo (KITCHENHAM, 1994). Essa etapa está dividida em três atividades:

A2.1 Definir o protocolo da revisão sistemática.

A2.2 Executar a busca de exemplos de unidades instrucionais e estratégias de ensino de computação e de desenvolvimento de aplicativos no ensino fundamental.

A2.3 Extrair informações relevantes sobre as unidades instrucionais e analisá-las como base para a unidade instrucional a ser elaborada.

Etapa 3. Elaboração da unidade instrucional para o ensino de desenvolvimento de aplicativos seguindo uma metodologia de design instrucional, conhecida como ADDIE (DICK & CAREY, 1996). Esta etapa está dividida em quatro atividades:

A3.1 Analisar o contexto e os aprendizes, definindo ao final os objetivos da aprendizagem

A3.2 Projetar a unidade instrucional, explicitando as técnicas de ensino utilizadas e os recursos necessários para o processo de ensino/aprendizagem.

A3.3 Desenvolver a unidade instrucional-todos os materiais, exemplos.

A3.4 Implementar os exemplos de códigos, os quais são utilizados na unidade de ensino.

A3.5 Realização de aplicações pilotos para testar a UI com equipe do Cne e INE5624.

Etapa 4. Aplicação e avaliação da unidade instrucional desenvolvida. Nesta etapa a unidade instrucional desenvolvida é colocada em prática por meio de uma série de estudos de casos aplicando a unidade instrucional. Essa etapa está dividida em cinco atividades:

A4.1 Solicitar a aprovação da CEPSH.

A4.2 Definir e planejar a aplicação e avaliação da unidade instrucional.

A4.3 Executar a aplicação da unidade instrucional e coletar os dados

A4.4 Analisar dos dados coletados.

1.4 Estrutura do Documento

No capítulo 2 é apresentado uma fundamentação acerca dos conceitos necessários para o desenvolvimento da unidade instrucional. No capítulo 3 é apresentado o estado de arte sobre UIs existentes referente ao ensino de computação através de desenvolvimento de aplicativos com o App Inventor. No Capítulo 4 é apresentado a elaboração da UI proposta no presente trabalho incluindo a caracterização do contexto de aplicação da unidade instrucional, (incluindo os aprendizes e ambiente), o design e o desenvolvimento da unidade instrucional. No capítulo 5 é apresentado a definição do método de avaliação da unidade instrucional, a descrição das aplicações realizadas baseadas na unidade instrucional desenvolvida e realizado uma análise da avaliação da unidade instrucional. No capítulo 6 é apresentado a conclusão do presente trabalho, os resultados e impactos obtidos e os trabalhos futuros relacionados.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os conceitos básicos sobre ensino e aprendizagem, a descrição do modelo ADDIE e a taxonomia de Bloom. Também é apresentado a situação atual do ensino de computação no Brasil e no mundo, o currículo de referência CSTA/ACM K-12. São apresentadas informações sobre o desenvolvimento de aplicativos para *smartphones* através do App Inventor.

2.1 Ensino e Aprendizagem

Aprendizagem é o processo sobre o qual são adquiridos novos conhecimentos, habilidades ou valores, como o resultado de estudo, experiência, raciocínio e observação. Aprender é essencial para a formação humana (LIBÂNEO, 2009), e pode ser considerada uma modificação do comportamento do indivíduo em função da experiência. Na prática, essa experiência é adquirida por meio de um método de aprendizagem, o qual no modo formal é constituído por atores, onde um ator possui o papel de instrutor e o outro o de aprendiz. Esta interação entre os atores resulta, para o aprendiz, no ato de aprender, e com isso, a relação entre aprendizagem e ensino fica evidente. Para o instrutor chegar ao seu objetivo, fazer o seu aprendiz aprender, é necessário ensinar.

Ensinar é o ato de transmitir conhecimento sobre algo a alguém. O **ensino** pode ser praticado de diversas maneiras, sendo através de instituições de ensino de um modo formal, ou informal através de relações interpessoais. Um conceito diretamente relacionado com o ensino é a instrução. **Instrução** é o preparativo intencional de condições de aprendizagem com um objetivo específico previamente identificado (DRISCOLL, 1995). É a preparação de atividades e ou experiências em unidades instrucionais orientadas pelo instrutor para transmitir o conhecimento desejado aos aprendizes (PANGRAZI, 2005; SMITH; RAGAN, 1999).

Assim, uma **unidade instrucional** pode ser uma aula, uma oficina, um curso, ou, até mesmo um jogo educacional. Esses exemplos podem ser caracterizados como sendo unidades de ensino e instrução. Unidades instrucionais devem possuir objetivos de aprendizagem bem definidos. Os **objetivos de aprendizagem** são constituídos de declarações que descrevem o conhecimento desejável e observável ou habilidades demonstradas com o resultado do processo de ensino. Contudo, para atingir os seus objetivos de aprendizagem as unidades instrucionais devem ser desenvolvidas de forma sistemática, usando sistemas de design instrucional. Sendo assim, existem

metodologias de planejamento e desenvolvimento sistêmicos para as Unidades Instrucionais seguindo os fundamentos de design instrucional (PIAZZA, 2012).

Além de auxiliar na construção, o **design instrucional** também possui uma metodologia para identificar carências de conhecimento e como atender estas necessidades. De acordo com FILATRO (2004), o design instrucional corresponde à "ação intencional e sistemática de ensino, que envolve o planejamento, o desenvolvimento e a utilização de métodos, técnicas, atividades, materiais, eventos e produtos educacionais em situações didáticas específicas, a fim de facilitar a aprendizagem humana a partir dos princípios de aprendizagem e instrução conhecidos".

A família de modelos de design instrucional é representada pelo termo *Instructional System Development* (ISD) (BRANCH, 2009). Dentre vários dos modelos ISD's existentes, muitos têm como referência o modelo ADDIE e acabam por serem considerados variações dele (PISKURICH, 2015). O modelo ADDIE representa um ciclo de cinco etapas a serem seguidos, pelos designers instrucionais, para o desenvolvimento de unidades instrucionais eficientes.

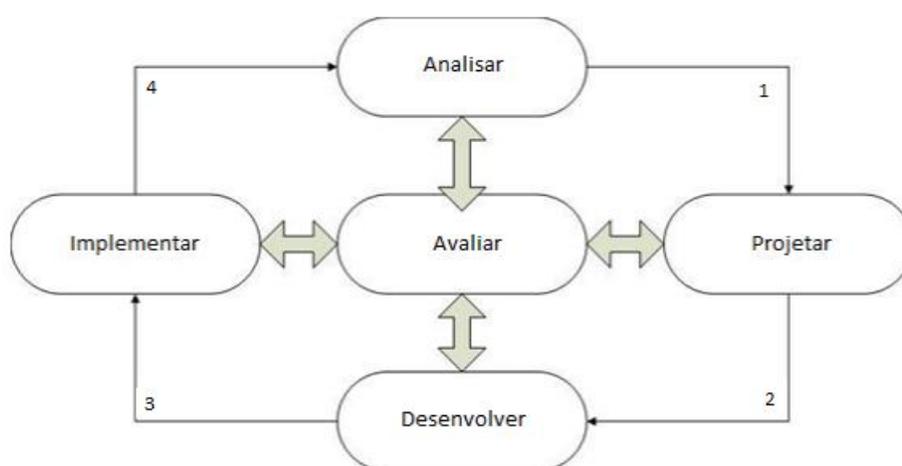


Figura 1 – Visão geral do Modelo ADDIE Adaptado (BRANCH, 2009)

As fases do modelo ADDIE a seguir são detalhadas conforme BRANCH, 2009.

A etapa de **Análise** tem como objetivo definir as metas e objetivos de aprendizagem da unidade, bem como, a identificação do público alvo e suas características. É importante caracterizar o público alvo considerando informações pessoais e educacionais sobre os alunos, como a idade, nacionalidade, experiências anteriores e os seus interesses. Assim como, é importante determinar outros fatores sobre o projeto, como recursos técnicos, recursos humanos, tempo. Também é importante identificar os requisitos necessários aos responsáveis pela

aplicação da unidade instrucional, e os requisitos necessários aos participantes para que sejam capazes de atingir os objetivos.

Na etapa de **Projeto** deve-se especificar os objetivos de aprendizagem da unidade, o conteúdo abordado, a estratégia instrucional, e assim, especificando como será a unidade instrucional. Os objetivos de aprendizagem definem o que o aprendiz deve aprender de competências (incluindo conhecimento, habilidades e/ou atitudes – KSA¹) ao fim uma instrução. Para cada aspecto de competência citado anteriormente, existem definições de níveis de aprendizagem. A **Taxonomia de Bloom** (BLOOM, 1956), é um framework estruturado em graus de aprendizagem, utilizado para a definição dos objetivos de aprendizagem. Estes objetivos são divididos em 3 domínios: o Cognitivo (conhecimento), Afetivo (atitudes) e Psicomotor (habilidades). Em cada domínio os novos conhecimentos são adquiridos a partir da evolução entre os graus de complexidades descritos nas tabelas 1, 2, 3.

Domínio Cognitivo	
Nível	Descrição
Conhecimento	Memorização de fatos específicos, padrões de procedimentos e de conceitos. O atributo essencial desse nível da taxonomia é a recordação.
Compreensão	Compreende o significado, traduz, interpreta problemas e instruções.
Aplicação	Aplicar a aprendizagem desenvolvida em novas situações.
Análise	Divide os elementos em partes, de modo em que seja entendido o como as partes constituintes se relacionam, e também, facilitando o entendimento de cada uma.
Síntese	Envolve combinar elementos e partes de modo a formar um todo. Isto essencialmente envolve trabalhar com partes, pedaços e elementos, e reorganizá-los de maneira tal, a constituir um padrão estrutural o qual não estava anteriormente presente.
Avaliação	Faz julgamentos do valor das ideias com base em evidências, critérios ou padrões.

Tabela 1- Níveis de aprendizagem do domínio cognitivo da taxonomia de Bloom (BLOOM, 1956)

Domínio Afetivo	
Nível	Descrição
Receptividade	Está relacionada com a sensibilidade do aluno para com a existência de determinados estímulos e a motivação de atentar para eles ou recebe-los.
Resposta	É a reação de um aluno à um estímulo. Inicialmente o aluno responde ao estímulo somente quando incitado. Posteriormente, o aluno apresenta respostas aos próprios estímulos.
Valorização	O aluno atribuí valor a um objeto, fenômeno, comportamento ou princípio.

¹ KSA – Knowledge, Skills and Abilities.

Organização	O aluno organiza diferentes valores em um sistema, determina como elas se relacionam e estabelece uma hierarquia de valores.
Caracterização	Os valores já estão internalizados e hierarquizados no indivíduo, influenciando os seus comportamentos.

Tabela 2 - Níveis de aprendizagem do domínio afetivo da taxonomia de Bloom (Bloom, 1956)

Domínio Psicomotor	
Nível	Descrição
Percepção	O primeiro passo para desempenhar um ato motor é o processo de se tornar consciente dos objetos, qualidades ou relações através da percepção de estímulos sensoriais.
Prontidão	O aluno está atento e pronto para agir quando necessário.
Resposta guiada	É um passo inicial no desenvolvimento de uma habilidade motora. A ênfase está nas aptidões que são componentes de habilidades mais complexas. É o ato comportamental de um indivíduo sobre orientação de um outro indivíduo.
Mecanismo	Estágio intermediário no desempenho de uma habilidade complexa.
Resposta Complexa	Desempenhar um ato motor que é considerado complexo por causa do seu padrão de movimento requerido. A ação pode ser efetuada eficiente e suavemente, isto é, com um mínimo de gasto de energia e tempo.
Adaptação	Habilidades desenvolvidas e modificação de padrões de movimento para situações fora do comum.
Originalidade	Utilização de novos movimentos e ou respostas para situações inesperadas e ou novas.

Tabela 3 - Níveis de aprendizagem do domínio psicomotor da taxonomia de Bloom (Bloom, 1956).

A **estratégia instrucional** define como são estruturadas e sequenciadas as informações, e também, como são apresentados os conteúdos durante a aplicação da unidade instrucional, para ser possível alcançar os objetivos de aprendizagem previstos (MAZZIONI, 2009). Define-se o conteúdo, como o assunto o qual deve ser transmitido aos alunos. Existem algumas estratégias de ensino definidos para determinar o modo com que o professor deve conduzir a unidade instrucional com o intuito de alcançar os objetivos de ensino. Assim como, existem **métodos instrucionais** os quais são utilizados pelos professores, para criar um ambiente de ensino e especificar a natureza da atividade a qual, ambos, professor e aluno estarão envolvidos durante uma aula. Alguns métodos instrucionais são comumente associados a certas estratégias instrucionais, entretanto, outros métodos podem se encaixar em uma grande variedade de estratégias. Na tabela 4 são apresentadas as variações de estratégias instrucionais, junto com alguns exemplos de métodos instrucionais que normalmente são aplicados em cada variação de estratégia.

Estratégias instrucionais	
Estratégias	Descrição
Instrução Direta	Essa estratégia define que as aulas sejam expositivas , com o professor explicando e mostrando o conteúdo o qual deseja ensinar. Nesta estratégia o professor é o centro da unidade instrucional, ele possui o controle sobre o ritmo em que a aula deve prosseguir. Normalmente é a metodologia mais utilizada, tendo como exemplos: palestras, prática e exercício e demonstração.
Instrução Indireta	É o contrário da estratégia de instrução direta, pois demanda mais interação do aluno , já que ele deve observar, investigar, formar hipóteses, e etc. Sendo o aluno o centro da unidade instrucional, o professor tem o papel de facilitador, seja fornecendo suporte, e <i>feedback</i> assim quando necessário. Como exemplos, temos: resolução de problemas, formação e obtenção de conceitos, discussão reflexiva.
Instrução Interativa	É um método realizado com discussões em grupo e compartilhamento de ideias . Esta estratégia depende da competência do professor em realizar dinâmicas em grupo, definindo o escopo do tema a ser discutido, o tempo disponível, tamanho dos grupos, e etc. Alguns exemplos são: discussões de classe, projetos em grupos, dentre outros.
Aprendizagem Experiencial	É a estratégia de ensino orientado a atividades . Sendo que a ênfase se encontra na aprendizagem experiencial, e não no seu produto. Pode ser vista como um ciclo de cinco fases: experienciar, compartilhar, analisar, inferir e aplicar. Exemplos: conduzir uma pesquisa de opinião pública, realizar simulações.
Estudo Independente	É proposto para desenvolver a iniciativa individual e auto aperfeiçoamento . O foco do estudo independente é sobre as atividades desenvolvidas pelo aluno individualmente, mas também, podem ser desenvolvidas atividades sobre a supervisão de um professor, e até mesmo em grupos pequenos de alunos.

Tabela 4 - Categorias de estratégias instrucionais (SASKATCHEWAN EDUCATION, 1991)

Outro aspecto importante da unidade instrucional é o seu **sequenciamento**, pois ele é fundamental para que ocorra uma boa aprendizagem pelos alunos. De acordo com a teoria construtivista (BRUNER, 1966), a aprendizagem é um processo ativo em que os alunos constroem novas ideias ou conceitos baseados em seu conhecimento passado. Por esse motivo o sequenciamento das atividades em uma unidade instrucional deve ser organizado em uma forma espiral de modo a que os alunos construam continuamente sobre o que eles já aprenderam (FLÔRES, 2009).

Outra necessidade de definição desta etapa é o método que deve ser aplicado para obter as informações sobre o progresso dos participantes durante a aplicação da unidade. Tem-se como resultado desta etapa, um documento de especificação da unidade instrucional, o plano de ensino da unidade instrucional.

A etapa de **Desenvolvimento** visa elaborar os conteúdos e os materiais didáticos, os quais foram definidos na etapa de Projeto, e serão utilizados na aplicação da unidade. Dependendo do(s) método(s) instrucional(ais) definidos é preparado o material, p.ex. *slides* para aulas expositivas, listas de exercícios, aplicativos exemplos, entre outros.

A etapa de **Implementação** ocorre após terem sido desenvolvidos todos os materiais e conteúdo os quais serão utilizados na unidade instrucional. Essa etapa consiste em aplicar a unidade instrucional com o público alvo, e analisar a sua eficiência verificando se os objetivos de aprendizagem foram atingidos, assim como os requisitos especificados para a unidade instrucional. Caso o responsável por aplicar a unidade instrucional não possua todas as qualificações para tal papel, deve-se realizar um treinamento prévio à aplicação da unidade.

A etapa de **Avaliação** é de suma importância para a unidade instrucional. A avaliação pode ter enfoques diferentes, um mais diretamente avaliando o desempenho do aluno consistindo em duas fases, sendo elas a avaliação formativa, a qual está presente em todas as etapas anteriores, e avaliação somativa, a qual é realizada ao final da unidade instrucional. A **avaliação formativa** consiste em fornecer um retorno sobre o aprendizado dos alunos para o professor, com o intuito de auxiliá-lo durante o processo de ensino-aprendizagem, para alterar os métodos e atividades de ensino com o propósito de melhorar o aprendizado dos alunos. A **avaliação somativa** tem como foco avaliar o desempenho dos alunos ao final da unidade instrucional e tendo a perspectiva da conclusão em evidência. Outro enfoque importante é a **avaliação da própria unidade instrucional** com o objetivo de identificar os pontos fortes e fracos guiando a sua melhoria contínua.

A avaliação da unidade instrucional tem o foco sobre os artefatos do design instrucional, quando contemplam, por exemplo, os objetivos de aprendizagem, estratégia instrucional, método instrucional, etc. Com isto, é necessário definir explicitamente a avaliação, definindo um programa de medição, também deve-se definir os métodos de coleta de dados de acordo com o tipo de estudo empírico escolhido (p.ex. estudos de caso ou experimentos) e projetar instrumentos de coleta de dados (p.ex. questionários). Uma abordagem que auxilia tanto na definição da medição quanto na sua análise e o *Goal Question Metric (GQM)* (BASILI; CALDIERA; ROMBACH, 1994),.

Durante a etapa de implementação os dados são coletados conforme definidos e ao final analisados conforme a definido no programa de medição GQM.

2.2 Ensino de computação no Ensino Fundamental II

O ensino fundamental, juntamente com a educação infantil e o ensino médio compõe o que é chamado de educação básica no Brasil. A partir do ano de 2006 o ensino fundamental passou a possuir a duração de nove anos, sendo dividido em duas

partes, do 1º ao 5º ano denominado de Ensino Fundamental I e do 6º ao 9º ano de Ensino Fundamental II (MINISTÉRIO DA EDUCAÇÃO, 2007). Idealmente, aos seis anos de idade as crianças deveriam entrar no Ensino Fundamental I, e sair do Ensino Fundamental II aos quatorze anos (MINISTÉRIO DA EDUCAÇÃO, 2007). Durante este período tem-se como finalidade desenvolver o educando, para que ele possa compreender a cidadania como participação social e política, posicionar-se de maneira crítica, responsável e construtiva nas diferentes situações sociais, dentre vários outros objetivos (PCN, 2015).

No Ensino Fundamental II, assim como nos outros ciclos da educação básica, o Ministério da Educação (MEC) disponibiliza os Parâmetros Curriculares Nacionais, os quais foram criados para possibilitar uma base comum aos currículos nacionais. Além de possibilitar a sua complementação, em cada sistema de ensino e estabelecimento escolar, devido a diversidade e importantes características regionais e locais da sociedade, da cultura, da economia e da clientela (PCN, 2015).

Para o 3º e 4º ciclos do ensino fundamental, os quais correspondem do 6º ao 9º ano, é previsto, nos Parâmetros Curriculares Nacionais, que sejam abordados conteúdos sobre as áreas de conhecimento como: Língua Portuguesa, Matemática, História, Geografia, Ciências Naturais, Educação Física, Arte e Língua Estrangeira. Entretanto, nestes Parâmetros Curriculares Nacionais, não consta nenhuma referência à área de Computação (PCN, 2015). Contudo, mesmo sem Currículos de Referência, já existe um consentimento sobre esta necessidade por parte da Sociedade Brasileira de Computação, a qual no seu último plano de gestão citou "... É importante salientar que devemos primar pela qualidade do ensino em todos os níveis da cadeia de formação de recursos humanos. Entendemos que a Computação deva ser ensinada desde o ensino fundamental, a exemplo de outras ciências como Física, Matemática, Química e Biologia. Esses são pontos muito importantes para que no futuro tenhamos recursos humanos qualificados para enfrentar os desafios que advirão" (SBC, 2015). Deste modo, a partir do pronunciado da SBC é notória a importância do ensino de computação e a necessidade de existência de um currículo base nacional, integrando conhecimentos relacionados a computação de um âmbito prático e teórico.

Porém ainda não existe um currículo nacional para o ensino de computação no Ensino Básico. Contudo, internacionalmente já existem diversos currículos de referência voltado ao Ensino Básico, entre eles um dos mais reconhecidos o currículo de referência CSTA/ACM K-12 (CSTA, 2011). Este currículo de referência é definido em 3 níveis abrangendo todo o Ensino Básico, conforme o objetivo do presente trabalho o foco aqui é o nível 2 por ser referente ao mesmo público alvo (Figura 2).

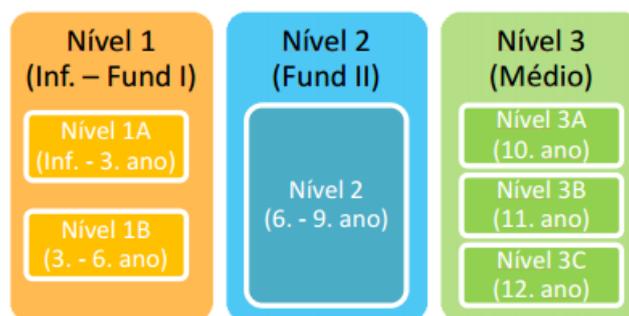


Figura 2 - Níveis de Ensino de Computação (CSTA, 2011)

Este currículo de referência criado pelo Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE) em cooperação com a *Association for Computing Machinery* (ACM), determina que o ensino de computação, durante o período de ensino fundamental, não se limite somente sobre o uso de TI, por exemplo, utilizando um editor de textos ou editor de imagens, e sim que o aluno desenvolva habilidades relacionadas a computação, como a solução de problemas através da sua decomposição em problemas menores, a capacidade de desenvolver algoritmos, trabalhar em conjunto e também conhecer os impactos sociais que o desenvolvimento das tecnologias podem causar, sejam positivamente ou negativamente (CSTA, 2011). Sendo assim, ensinar computação inclui várias áreas de conhecimento conforme apresentado na figura 3.



Figura 3 - Áreas de conhecimento de Computação (CSTA, 2011)

O **Pensamento Computacional** é definido como um subconjunto de competências e habilidades, as quais envolvem competências relacionadas à abstração e decomposição de problemas de forma a permitir sua resolução usando recursos computacionais e estratégias algorítmicas (WING, 2006). Os alunos devem utilizar o

pensamento computacional como uma ferramenta de resolução de problemas, aplicar práticas de colaboração, como programação em pares, trabalhos em equipes de projeto, e também, analisar os impactos positivos e negativos da computação na cultura humana, dentre muitos outros objetivos previstos (CSTA, 2011).

A **Programação** é uma parte essencial da computação, desenvolvendo a competência de criar programas de software. Os alunos devem aprender desde a fase de projeto, de desenvolvimento até a publicação dos seus produtos (websites, aplicativos para dispositivos móveis, jogos) utilizando recursos tecnológicos. Nesta área deve ser explorado a compreensão de algoritmos e a sua aplicação lógica, além da implementação de software utilizando uma linguagem de programação (CSTA, 2011).

Em relação a **Colaboração** os aspectos considerados essenciais para a computação, são habilidades como o trabalho em equipe, crítica construtiva e comunicação eficaz. Estas habilidades são necessárias em um profissional desta área devido ao fato de que raramente um progresso significativo em computação é realizado por uma pessoa trabalhando sozinha. Normalmente, os projetos de computação envolvem grandes equipes de profissionais que trabalham em conjunto para projetar, codificar, testar, depurar, descrever e manter o software ao longo do tempo (CSTA, 2011).

Na área de conhecimento **Computadores e Dispositivos de Comunicação** devem ser abordadas informações sobre os elementos que compõe um computador, os dispositivos de comunicação como *smartphones*, *tablets* etc. Junto com informações acerca da Internet e como ela é responsável por toda a facilidade perante a comunicação global.

Também devem ser abordados temas sobre os **Impactos éticos, globais e na comunidade**, informando sobre princípios de segurança de rede, privacidade, licenças de softwares e direitos autorais com o fim de tornar os alunos cidadãos responsáveis no mundo moderno. Os alunos também devem compreender os impactos dos computadores na comunicação internacional e serem capazes de avaliar os impactos positivos e negativos dos computadores na sociedade.

Estão detalhados os objetivos de aprendizagem voltado ao nível 2 na tabela 5.

Objetivos de aprendizagem Nível 2 (6.-9. Ano – Ensino Fundamental II) (CSTA, 2011)	
	O aluno será capaz de:
Pensamento Computacional	[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). [O2] Descrever o processo de paralelização na forma que se refere à resolução de problemas.

	<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 <i>feedback</i> útil e integrante, compreender e aceitar múltiplas perspectivas, socialização.</p>
Práticas Computacionais e de Programação	<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 (seqüências 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</p>

	<p>computação.</p> <p>[O27] Demonstrar receptiva disposição para resolver e programar problemas indeterminados (p.ex., conforto com complexidade, persistência, <i>brainstorming</i>, 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>
Computadores e Dispositivos de Comunicação	<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>
Impactos Éticos, Globais e na Comunidade	<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>

Tabela 5 - Objetivos de aprendizagem para o nível 2 (CSTA, 2011)

2.3 Desenvolvimento de apps móveis com App Inventor

Os *smartphones* surgiram como a combinação entre duas classes de dispositivos, os celulares e os assistentes pessoais (PDAs²) (MORIMOTO, 2009). Diferente dos seus antecessores são cada vez mais vistos como computadores portáteis, isso devido a sua capacidade computacional, grande memória, grandes telas sensíveis ao toque e sistemas operacionais que incentivam o desenvolvimento de aplicações (BOULOS et al., 2011). Em comparação com os telefones celulares, os *smartphones* geralmente possuem telas

² PDAs – *Personal Digital Assistant*

maiores, com maior resolução, *touchscreen*, além da possibilidade de se conectarem a web através de conexões 3G, 4G e redes sem fio (Wi-Fi). A maioria dos novos *smartphones* também incorporam outras funcionalidades, por exemplo, câmeras de alta qualidade, sistemas de geolocalização, etc.

Durante o ano de 2014, os smartphones ultrapassaram a marca de 1.2 bilhões de unidades vendidas (GARTNER, 2015). De acordo com a pesquisa Acesso à internet e à Televisão e Posso de Telefone Móvel para Uso Pessoal (IBGE, 2013), o percentual de pessoas que tinham telefone móvel celular para uso pessoal, na população de 10 ou mais anos de idade em Santa Catarina aumentou de aproximadamente 46% em 2005 para 77,1% em 2013 (Figura 4). Na região sul do Brasil, em 56% dos domicílios é realizado o acesso à internet por meio de telefones celulares móveis ou tablets.

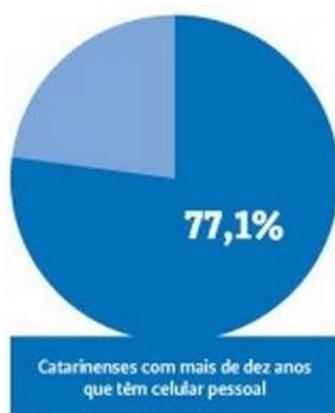


Figura 4 - Catarinenses com mais de dez anos com celular pessoal (IBGE, 2013)

Na figura 5 é possível visualizar o contexto de utilização e acesso à internet através de telefones celulares móveis com crianças de 10 a 14 anos no Brasil.

Crianças de 10 a 14 que:

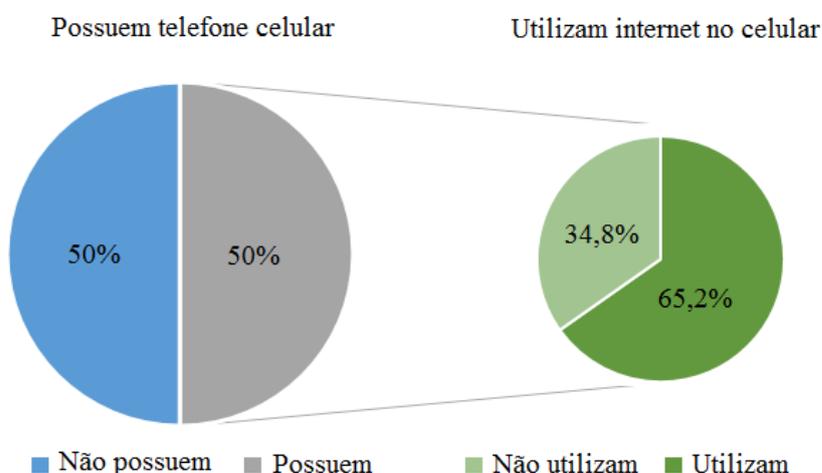


Figura 5 - Crianças de 10 a 14 anos que têm celular e utilizam a internet (IBGE, 2013)

Como a faixa etária do ensino fundamental II se encontra dentro do grupo de pessoas analisados na pesquisa, pode-se verificar que a utilização de *smartphones* entre as crianças no ensino fundamental II está em crescimento e já atinge mais da maioria da população analisada.

No mercado global de smartphones, atualmente a plataforma predominante é o Android (GARTNER, 2015) sistema operacional de código aberto desenvolvido pela Google (Figura 6). Este sistema operacional, nas vendas de *smartphones* para usuários finais no ano de 2014, obteve aproximadamente de 80% da fatia de mercado (GARTNER, 2015).



Figura 6 - Fatia de mercado das vendas por sistema operacional 2014 (GARTNER, 2015)

Um dos grandes sucessos dos smartphones foram os aplicativos. Um aplicativo ou *app* é um programa de computador desenvolvido para o uso em *smartphones* e ou *tablets* e normalmente permite a realização de tarefas específicas (FLING, 2009). Os aplicativos foram popularizados pela introdução da App Store, a loja online de aplicativos, da Apple em 2008, a qual em uma semana obteve mais de 10 milhões de aplicativos baixados (MACWORLD, 2015). Em sua maioria, os aplicativos são desenvolvidos por empresas terceiras ou desenvolvedores independentes, e tem como objetivos estender as funcionalidades dos smartphones, criando programas adicionais, com inúmeras utilidades, desde aplicativos de entretenimento, produtividade, até aplicativos para monitoramento da saúde. O sistema operacional Android também possui a sua loja de aplicativos, também criada em 2008, e conta atualmente com aproximadamente 1,6 milhões de aplicativos disponíveis, a Google Play (STATISTA, 2015). A figura 7 mostra a curva de crescimento no número de aplicativos disponíveis

na Google Play desde o mês de dezembro de 2009 até julho de 2015 que demonstra o aumento da adoção deste tipo de software.

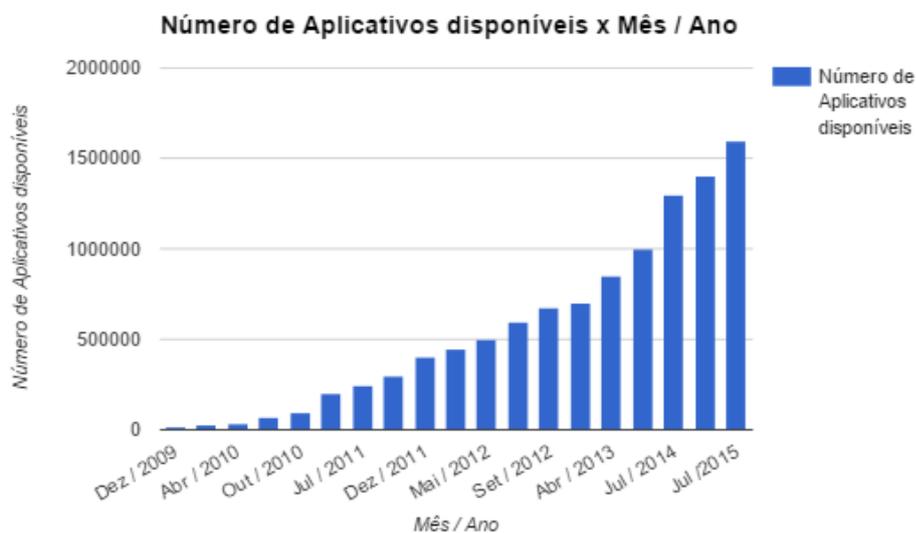


Figura 7 - Número de aplicativos móveis por mês na Google Play (STATISTA, 2015)

Atualmente, para o desenvolvimento de aplicações para o sistema operacional Android, utiliza-se o Ambiente de Desenvolvimento Integrado (IDE³) chamado de Android Studio (Figura 8). A programação dos aplicativos desta plataforma é realizada com linguagem de programação JAVA junto com o Android *Software Development Kit* (Android SDK) para o desenvolvimento das funcionalidades e XML (linguagem de marcação) na parte gráfica das telas dos aplicativos (ANDROID DEVELOPER, 2015).

Pelo fato das IDE's fornecerem muitos recursos para auxiliar o desenvolvimento, estas acabam por se tornarem mais complexas. Deste modo, o uso destas metodologias/ferramentas de desenvolvimento requer um nível avançado de conhecimento na área de computação tipicamente usado por profissionais da área e ou em cursos de educação superior (MACDONALD, 2014).

³ IDE - *Integrated Development Environment*

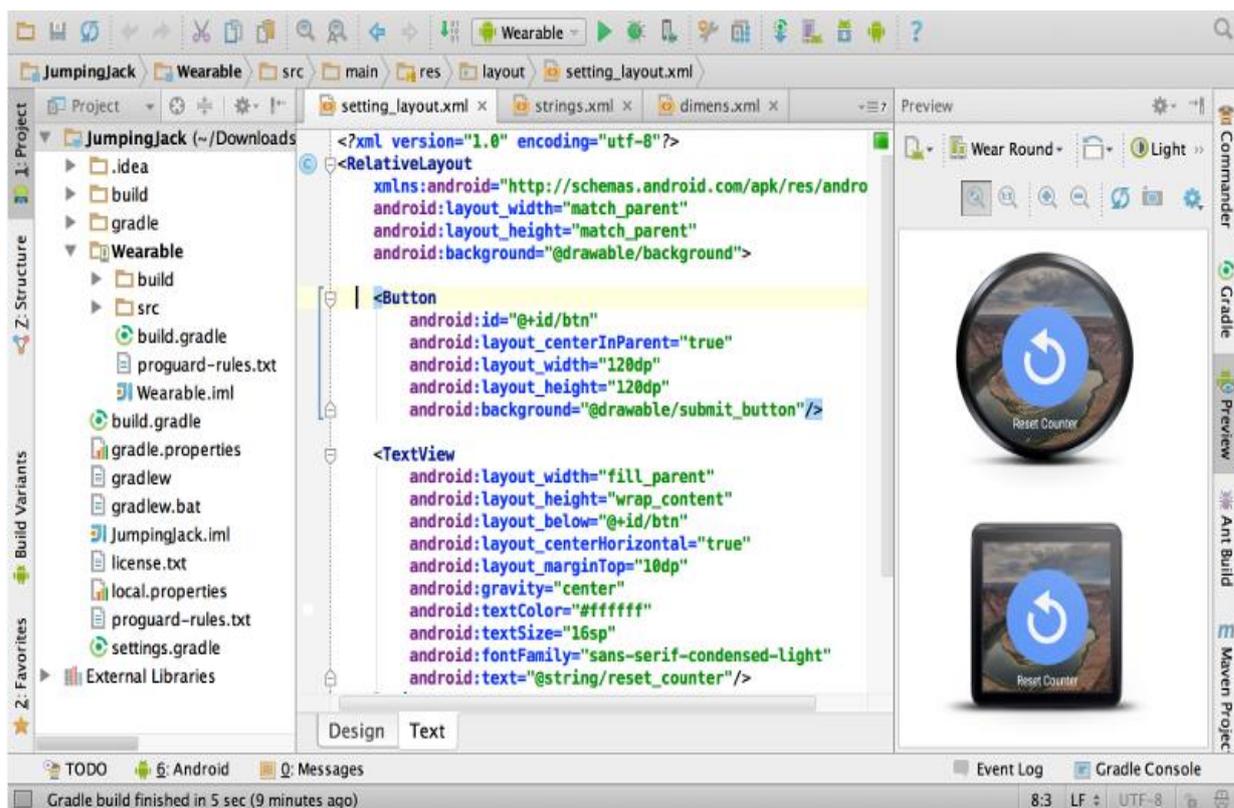


Figura 8 - Android Studio (Android Developer, 2015)

Deste modo, para ensinar crianças do ensino fundamental II é mais adequado utilizar ferramentas desenvolvidas para este público alvo, sendo projetadas para usuários inexperientes. (WATTERS, 2011). Com o intuito de transformar o complexo desenvolvimento de aplicativos através das IDE's e codificação baseada em texto foi desenvolvido o MIT App Inventor (APP INVENTOR, 2014). O App Inventor é uma aplicação web que permite o desenvolvimento de aplicativos para *smartphones*, ou *tablets* Android, ou até mesmo para um emulador com o mesmo sistema operacional (APP INVENTOR, 2014). No desenvolvimento de aplicativos através do App Inventor, pode-se desenvolver uma enorme variedade de aplicativos. Pode-se perceber nos dois exemplos a seguir a possibilidade de desenvolvimento de diferentes tipos de aplicativos através do App Inventor.

Exemplo 1 de aplicativo: É uma aplicação que simula o funcionamento de uma máquina enigma, a qual foi utilizada para o ciframento de mensagens durante a Segunda Guerra Mundial (BBC, 2015)

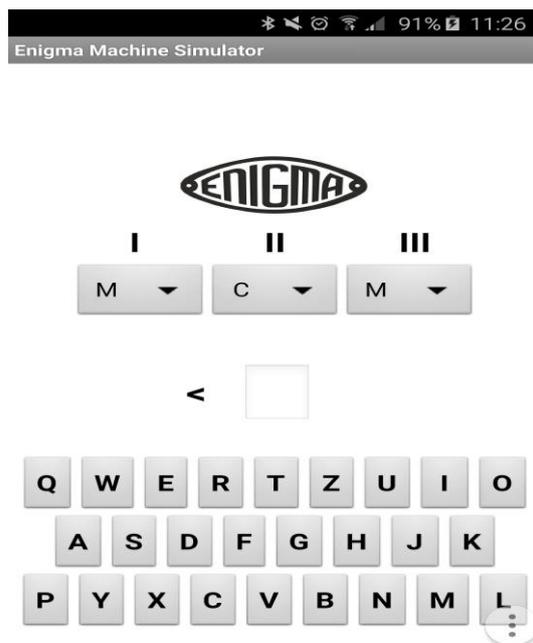


Figura 9 – Screenshot do Aplicativo Enigma Machine Simulator

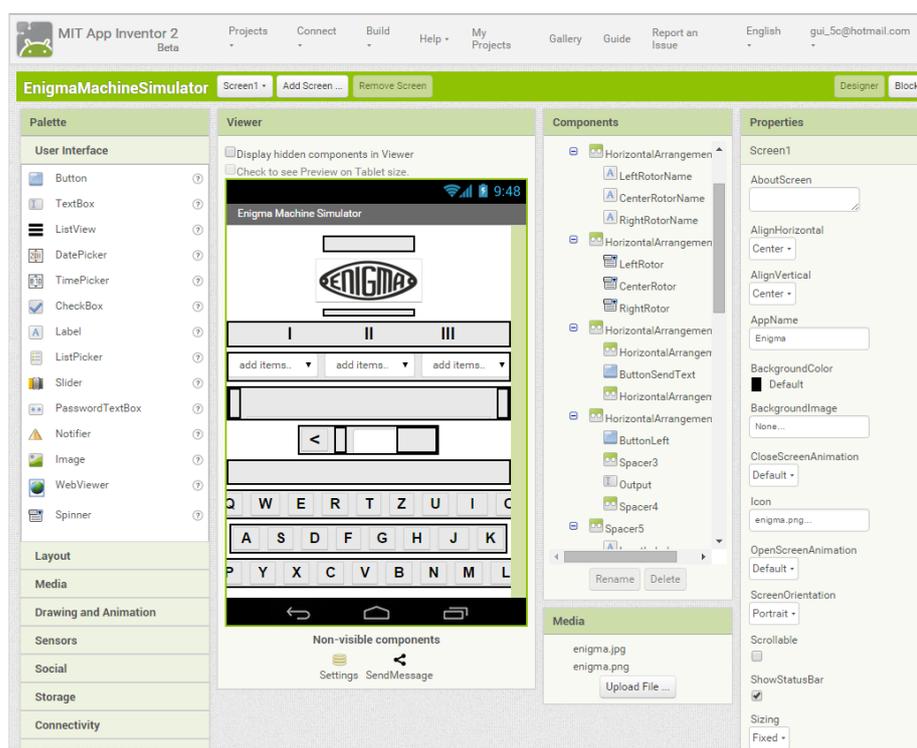


Figura 10 – Janela Designer do aplicativo Enigma Machine Simulator no App Inventor

Exemplo 2 de aplicativo: é um jogo de futebol que simula a função de um goleiro, tendo como objetivo não deixar as bolas entrarem dentro do gol.

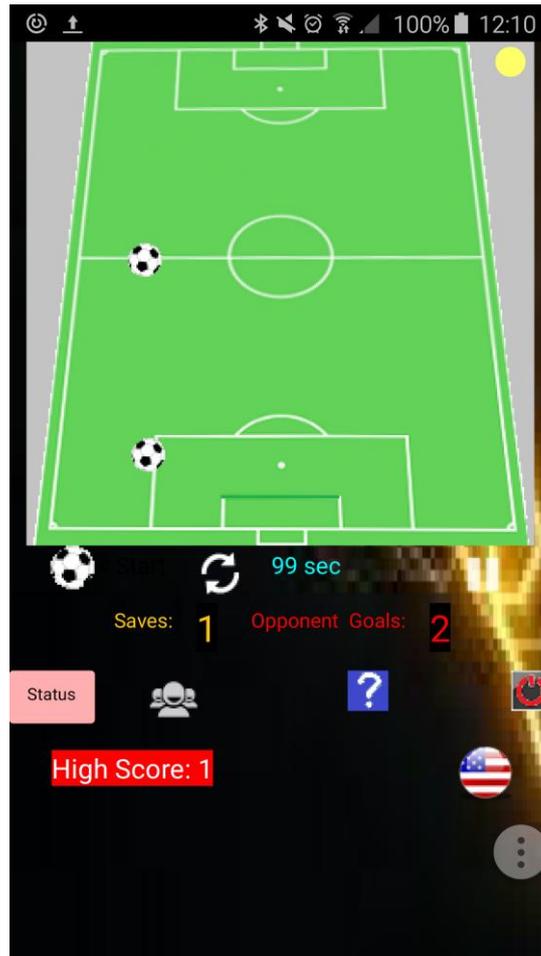


Figura 11 – Screenshot do Aplicativo Goalai2

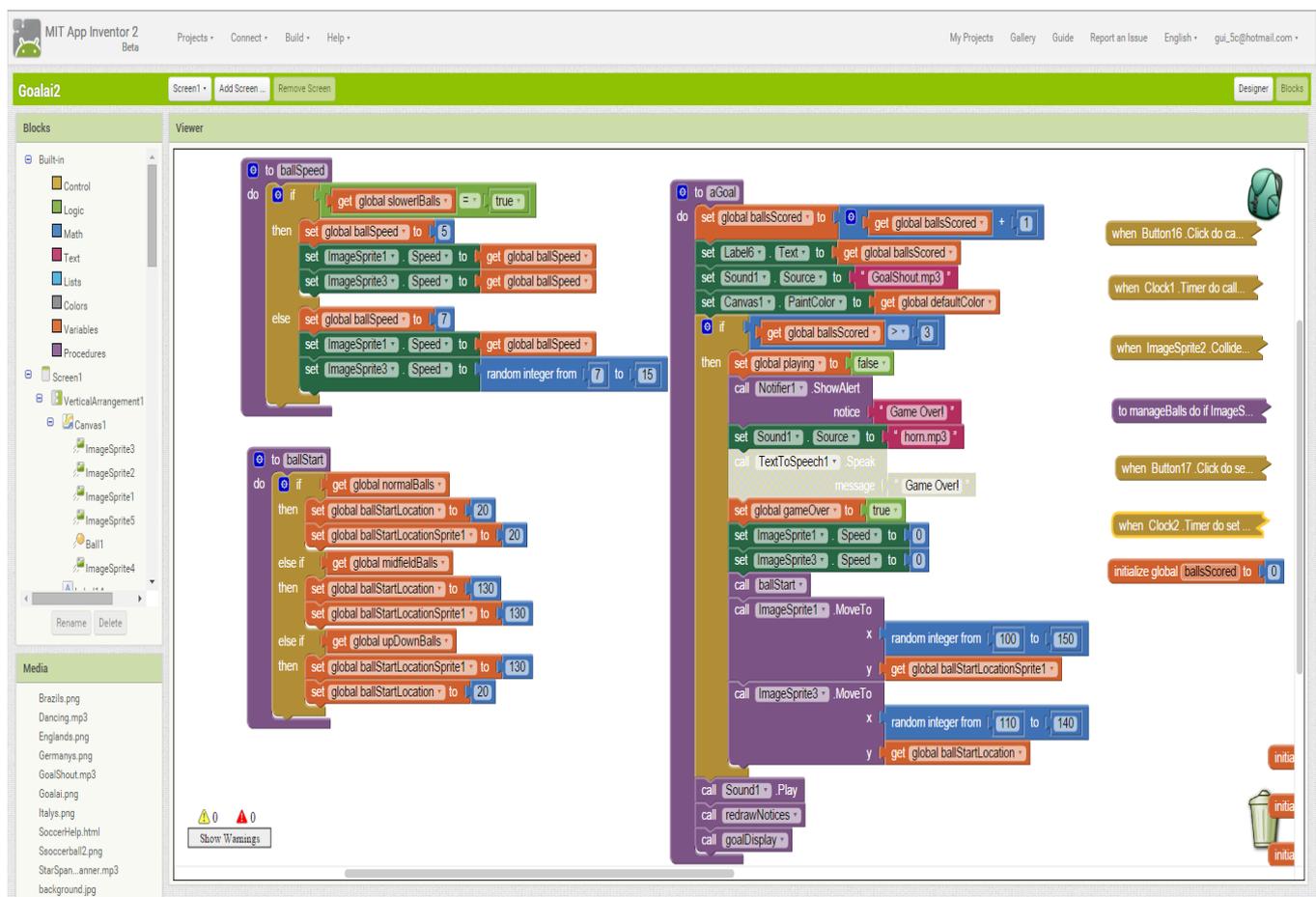


Figura 12 – Janela Editor de Blocos do aplicativo Goalai2 no App Inventor

O App Inventor permite a programação alinhada ao processo de desenvolvimento de aplicativos móveis que envolvam tanto a programação das funcionalidades em si, quanto o design das interfaces. O ambiente do App Inventor consiste do Editor de Blocos e do Designer (Figura 10 e 12). O Editor de Blocos (Figura 12) permite a programação das funcionalidades dos aplicativos, unindo os blocos de códigos, os quais podem utilizar blocos de comportamentos gerais ou blocos com comportamentos específicos para alguns componentes (APP INVENTOR, 2014).

Na tabela 6 são apresentadas as categorias de blocos de código de comportamento geral que podem ser utilizados no desenvolvimento do aplicativo no Editor de Blocos.

Categorias de blocos de código de comportamento geral	
Categorias	Descricao
Controle	Possui blocos de códigos de controle para a programação, como: comandos de decisão if-then-else, loops. E também controle da aplicação possuindo blocos para abrir uma

	outra tela da aplicação ou até mesmo fechar toda a aplicação.
Lógica	São blocos de operações lógicas e valores como verdadeiro ou falso. Contemplam operações lógicas como AND ou OR, e verificação se dois valores são iguais ou diferentes.
Matemática	Possui blocos com operações matemáticas, como: adição, subtração, multiplicação, divisão, módulo, seno, cosseno, etc. Além de funções de conversão de base, verificação se é um número e função para selecionar um número randomicamente.
Texto	Possui blocos para realizar operações com textos, como: unir dois textos, verificar o tamanho do texto, colocar todas as letras em maiúscula ou minúscula, entre outras.
Listas	São blocos de operações em listas, realizando inserção de item na lista, remoção, substituição de itens, seleção de item, etc.
Cores	São blocos para criação de uma cor utilizando o sistema de cores RGB, ou utilização cores padrão.
Variáveis	São funções sobre variáveis, como: declaração de variável global, variável local, atribuição de valor à uma variável, obtenção de valor de uma variável.
Procedimentos	São blocos para a definição de funções e chamadas de outras funções.

Tabela 6 - Categorias de blocos de código do Editor de Blocos.

O Designer (Figura 10) é utilizado para o desenvolvimento da interface do aplicativo, posicionando os componentes na tela e definindo as suas propriedades, por exemplo, o tamanho, cor de fundo, comportamento (se é um botão, um campo de texto, etc.). Na tabela 7 são apresentados os componentes que podem ser utilizados na construção da interface dos aplicativos pelo Designer.

Componentes de Interface da Janela Designer	
Componentes	Função
Botão	Botão com a funcionalidade de detectar cliques. Muitas propriedades podem ser configuradas.
Caixa de seleção	Caixa de seleção que pode disparar um evento quando o usuário clica nela.
Escolhe data	Um botão que quando clicado, permite o usuário selecionar uma data.
Imagem	É um componente para apresentação de imagens.
Legenda	Uma legenda mostra um trecho de texto na tela.
Escolhe lista	Um botão que ao ser clicado, mostra uma lista de textos para o usuário escolher. Pode possuir um filtro de busca.
Visualizador de listas	Um componente que permite adicionar uma lista de elementos de texto para apresentar na tela.
Notificador	O componente notificador mostra diálogos de alerta, mensagens e alertas temporários.
Caixa de senha	Uma caixa de texto para digitar senhas. O mesmo que o componente caixa de texto comum, exceto por não exibir os caracteres digitados pelo usuário.
Deslizador	É uma barra de progresso a qual altera o seu valor, quando o usuário desliza para a

	direita ou esquerda.
Lista Suspensa	É um componente que apresenta uma janela com uma lista de elementos selecionáveis.
Caixa de texto	Um componente que permite o usuário inserir um texto.
Escolhe hora	Um botão que quando clicado, permite o usuário selecionar uma hora.
Navegador web	É um componente para a visualização de páginas web.
Pintura	É uma componente retangular sensível ao toque onde se pode desenhar e animar um objeto gráfico bi ou tridimensional que se move pela tela, chamado de “SpriteImagem” no App Inventor.
SpriteImagem	É um objeto gráfico que pode ser colocado em uma Pintura, onde ele pode reagir a toques e arrastes, mover-se pela tela, e entre outros.
Temporizador	É um componente invisível que fornece a data/hora atual usando o relógio interno do dispositivo. O componente pode disparar um evento em intervalos de tempo regulares, e executar cálculos, manipulações e conversões de tempo.
Som	É um componente multimídia que reproduz áudio e controla a vibração do dispositivo.

Tabela 7 - Componentes do Designer.

Além destes componentes de interface e blocos de código de comportamento geral, no App Inventor é possível utilizar outras funcionalidades específicas. Por exemplo: acessar a câmera do dispositivo, tocar um som no autofalante, fazer uma gravação de som, utilizar o GPS e até se realizar funções em outros aplicativos do dispositivo.

O App Inventor possui uma interface gráfica simplista a qual ajude um desenvolvedor novato e/ou inexperiente tenha a capacidade de criar um básico aplicativo, mas totalmente funcional, em uma hora ou menos (APP INVENTOR, 2014). Isto também é possível devido à algumas funcionalidades importantes. Ele oferece uma arquitetura de componentes de alto nível os quais expõem funcionalidades complexas através de simples e concretas propriedades, eventos, métodos e os tratadores de eventos dos componentes. Outra vantagem é que ele utiliza a linguagem de programação baseada em arrastar e soltar os blocos de código, a qual minimiza erros de digitação e sintaxe. Além de ser possível o desenvolvimento de aplicativos totalmente funcionais para *smartphones* e *tablets* (WOLBER; ABELSON; FRIEDMAN; 2014).

Outra importante funcionalidade do App Inventor é o chamado “*Live Testing*” que permite aos desenvolvedores testar o seu aplicativo instantaneamente enquanto desenvolve. Esta funcionalidade é possível através do aplicativo MIT AI2 Companion disponível na Play Store, o qual realiza a conexão do dispositivo móvel com o servidor do App Inventor e com o computador onde o aplicativo está sendo desenvolvido. Para que esta conexão seja possível é necessário que o computador e o dispositivo móvel estejam conectados na mesma rede local. Também existe a possibilidade de desenvolver

aplicativos através do emulador fornecido pelo próprio App Inventor, para os casos em que não existe um dispositivo móvel disponível. E para locais onde não existe uma rede sem fio para a conexão dos dispositivos móveis é possível realizar o *Live Testing* conectando o dispositivo e o computador através do cabo *Universal Serial Bus* (USB).

O App inventor é uma aplicação web oferecida gratuitamente. E possui compatibilidade com os sistemas operacionais Linux, Windows e Macintosh, e pode ser utilizado nos navegadores de web Mozilla Firefox, Apple Safari, Google Chrome, porém não é compatível com o navegador Microsoft Internet Explorer. Os servidores do App Inventor armazenam todos os projetos desenvolvidos para facilitar para os desenvolvedores.

Em 2015, a comunidade do MIT App Inventor consiste em aproximadamente 3 milhões de usuários representando 195 países. Mais de 100.000 usuários ativos semanais, e já foram desenvolvidos mais de 7 milhões de aplicativos Android (APP INVENTOR, 2015).

3 ESTADO DA ARTE

O presente capítulo apresenta o estado atual de pesquisas relacionadas a unidades instrucionais que ensinam computação no ensino fundamental II com o App Inventor. A análise do estado da arte foi realizada seguindo o método de revisão sistemática de literatura definido por Kitchenham (2004).

3.1 Definição do protocolo de revisão

A revisão sistemática da literatura realizada tem como objetivo analisar e sintetizar a literatura existente sobre as unidades instrucionais que ensinam computação no ensino fundamental II com o App Inventor.

Foram pesquisados artigos publicados nas línguas portuguesa e inglesa sobre unidades instrucionais que ensinam computação para o ensino fundamental II através do App Inventor. As pesquisas foram realizadas no Google Scholar, por ser uma ferramenta de busca aberta e apresentar fontes exclusivas de bibliotecas digitais e bases de dados neste domínio (IEEEExplore, ACM Digital Library, App Inventor Resources). São considerados somente artigos acessíveis via o Portal CAPES (CAPES, 2015).

De acordo com a pergunta de pesquisa são usados os seguintes termos de busca, seus sinônimos e traduções, conforme apresentado na Tabela 8.

Termo	Sinônimos	Tradução (inglês)
Ensino	Instrução, Treinamento, Aprendizagem	<i>Educational, Teaching, Training, Learning</i>
Computação	Ciência da Computação	<i>Computing, Computer Science</i>
Smartphones	-	<i>Smartphones</i>
App Inventor	-	-
Ensino Fundamental II	-	<i>Middle School</i>

Tabela 8 - Sinônimos e traduções dos termos de busca.

Devido ao ambiente App Inventor ter sido disponibilizado publicamente somente a partir de dezembro de 2010, foram considerados somente publicações a partir desta data. E dentre estas publicações, só foram selecionadas unidades instrucionais com foco no ensino de computação no ensino fundamental II com o App Inventor.

Critérios de inclusão/exclusão. Os critérios para inclusão/exclusão são:

- O material deve descrever uma unidade instrucional.
- Deve objetivar o ensino de computação através do App Inventor, não sendo relevantes unidades instrucionais que utilizem outra ferramenta para o ensino de computação.
- Deve ter realizado ao menos uma aplicação da unidade instrucional.
- Dando prioridade a artigos relatando unidades instrucionais para o público alvo de crianças entre 10 a 14 anos. Porém, pela falta de unidades instrucionais encontradas para exatamente esta faixa etária, também se leva em consideração unidades instrucionais voltadas à outras faixas etárias, incluindo o ensino médio.

Critérios de qualidade. Foram considerados relevantes somente artigos ou materiais que constam de forma substancial a unidade instrucional, indicando por exemplo o objetivo da aprendizagem, plano de ensino (e/ou então sequência/conteúdo das aulas, materiais didáticos, etc.).

3.2 Execução da Busca

A busca foi realizada em outubro de 2015, usando o conjunto de termos que está apresentado na tabela 9. Foram analisados diversos tipos de publicações sobre informações de ensino de computação através do App Inventor.

Termos da Busca	Quantidade de resultados da busca	Quantidade de pesquisas relevantes dentre os primeiros 50 resultados da busca	Referências
(teaching OR training) AND (computing OR “computer science”) AND (“app inventor”) AND (“middle school”) 2010..2015	137	2	(GROVER; PEA, 2013) (CHATZINIKOLAKIS; PAPADAKIS, 2014)
(ensino OR treinamento) AND (computação OR “ciencia da computação”) AND (“app inventor”) AND (“ensino fundamental”) 2010...2015	10	1	(RAMOS et al., 2014)
(ensino OR educational OR treinamento OR training) AND (computação OR computing) AND (“app inventor”) 2010...2015	702	2	(FINIZOLA et al., 2014) (GOMES; MELO, 2012)

Tabela 9 - Termos usados na execução da busca no Google Scholar.

Dentre todos os artigos encontrados somente os primeiros 50 artigos, por ordem de relevância, foram analisados. Os demais resultados, considerados não relevantes, não atenderam aos critérios de inclusão esperados por várias razões como p.ex. por não serem uma unidade instrucional; por não terem realizado ao menos uma aplicação; ou por não terem sido planejadas para o mesmo público alvo.

Aplicando os critérios de inclusão e exclusão foram identificados 5 artigos como relevantes a pergunta desta pesquisa. Um grande número de artigos inicialmente retornado só apresentavam informações sobre o ensino de computação, não exatamente com o App Inventor. Além destes artigos, possuíam outros que descreviam as funcionalidades e informações referentes ao App Inventor, sem descrever uma unidade instrucional. Muitos artigos também abordavam uma discussão de como o App Inventor pode ser uma diferente vertente de ensino de computação comparando com outros softwares existentes para este mesmo objetivo, novamente não se encaixando nos critérios de inclusão.

Analisando os 5 artigos considerados relevantes e aplicando o critério de qualidade, foi descartado o artigo (GOMES; MELO, 2012), pelo fato de não descrever em um nível mínimo de informação e detalhes a própria unidade instrucional e os resultados obtidos.

Assim, ao final foram considerados 4 artigos em relação ao levantamento do estado de arte.

3.3 Extração das informações e análise dos resultados

Na tabela 10 são definidos os identificadores dos artigos considerados.

ID	Referência
1	(GROVER; PEA, 2013)
2	(CHATZINIKOLAKIS; PAPADAKIS, 2014)
3	(RAMOS et al., 2014)
4	(FINIZOLA et al., 2014)

Tabela 10 - Identificadores dos artigos considerados.

O contexto de aplicação das quatro unidades instrucionais selecionadas, com a faixa etária, o tipo do evento e o gênero dos participantes está apresentado na tabela 11.

ID	Faixa Etária	Tipo de evento	Participantes
-----------	---------------------	-----------------------	----------------------

1	Em média 13 anos	Oficina durante um dia inteiro (8 horas)	3 meninas e 4 meninos
2	De 13 a 16 anos	Curso com 4 sessões de duas horas, realizado em escolas como parte das atividades diárias.	3 meninas e 32 meninos
3	De 15 a 18 anos	Curso com duração de 7 meses.	17 meninas
4	Em média 15 anos	Curso de curta duração (12 horas/aula)	20 participantes (não foi explicitado o gênero).

Tabela 11 - Contexto de Aplicação das UIs.

Na tabela 12 está resumido a definição instrucional das unidades instrucionais selecionadas, apresentando o objetivo de aprendizagem, o número de aulas lecionadas, o conteúdo abordado, a metodologia de ensino, o formato do curso e o material didático utilizado.

ID	Objetivo de Aprendizagem	Duração	Conteúdo	Metodologia de ensino	Formato da UI	Material didático
1	Ensinar conceitos computacionais introdutórios.	1 aula com 8 horas de duração.	Utilização de funcionalidades dos smartphones e conceitos fundamentais de computação.	Utilizou um processo de discussões academicamente produtivas (<i>academically productive talk</i>), com programação em pares, interações, colaborações dos alunos e professores.	Dividido em duas partes. Na primeira parte foi introduzido os conceitos básicos do desenvolvimento de aplicativos utilizando o App Inventor. Na segunda parte, em duplas, foi realizada a programação dos aplicativos escolhidos pela dupla.	Livro de exercícios do App Inventor (Wolber et al., 2011), com algumas alterações nos enunciados.
2	Motivar os estudantes para aprender sobre Ciência da Computação, e o ensino de conceitos fundamentais de programação.	4 aulas com 2 horas de duração cada.	Utilização de funcionalidades dos smartphones e conceitos fundamentais de computação.	Foi determinado para que os estudantes seguissem as instruções das atividades, não existindo um conteúdo previamente demonstrado pelo professor.	Dividido em quatro sessões, sendo que cada uma deveria ser contemplada em uma aula. Cada sessão abrangeu parte do conteúdo programado.	Foi desenvolvido atividades originais, livro em língua grega, exemplos em multimídia e projetos no App Inventor.
3	Motivar os estudantes para carreiras de ciências exatas, em especial, Ciência da Computação.	12 aulas e 8 encontros	Componentes do App Inventor, conceitos de programação e o processo de criação de um aplicativo.	Na primeira parte foi determinado para que os estudantes seguissem as instruções das atividades. Na segunda parte, em grupos, foram deixados livres para desenvolverem o próprio aplicativo.	Dividido em duas partes. Na primeira parte (12 aulas) foi introduzido os conceitos de desenvolvimento e componentes do App Inventor. A segunda parte (8 encontros) teve como objetivo o desenvolvimento de todo o processo de criação de um aplicativo.	Foi desenvolvido atividades originais, livro em língua portuguesa e projetos no App Inventor.
4	Motivar os estudantes para aprender sobre Ciência da Computação, e o ensino de conceitos fundamentais de programação.	4 aulas com 3 horas de duração cada.	Utilização de funcionalidades dos smartphones e conceitos fundamentais de computação.	Foi determinado para que os estudantes seguissem as instruções das atividades, não existindo um conteúdo previamente demonstrado pelo professor.	Dividido em quatro sessões, sendo que cada uma deveria ser contemplada em uma aula. Cada sessão abrangeu parte do conteúdo programado.	Livro de exercícios do App Inventor (Wolber et al., 2011), com algumas alterações nos enunciados.

Tabela 12 - Definição Instrucional das UIs.

Também foi analisado como as unidades instrucionais propostas foram avaliadas. Em relação as avaliações relatadas foram extraídas os objetivos da avaliação, o tipo de estudo empírico, os instrumentos de coletas de dados e o número de participantes das avaliações. O tipo de estudo foi classificado de acordo com o design de pesquisa usados na avaliação de contextos educacionais (GRESSE von WANGEHEIM & Shull, 2009), apresentado na tabela 13.

Tipo de Estudo	Descrição	Projeto	Representação (X=Aplicação; O=Medição; R=escolha aleatória)
Non-experimental	Um design o qual os pesquisadores somente observam como o fenômeno acontecer naturalmente, coletando informações no início e no final ou somente ao final da pesquisa, sem realizar manipulações na situação, ou nas circunstâncias, ou na experiência dos participantes	<i>One-shoot post-test only</i>	X O
		<i>One-shoot pre-test – post-test</i>	O X O
Quasi-experimental	É realizado uma avaliação sobre um grupo de participantes, os quais não são escolhidos de uma forma aleatória. E outra avaliação com um grupo que não realizou o experimento.	<i>Static group comparison group</i>	X O O
		<i>Static group pre-test post-test</i>	O X O O O
		<i>Times series</i>	O O X O O
Experimental	É realizado uma avaliação sobre um grupo de participantes, os quais são escolhidos de uma forma aleatória. E outra avaliação com um grupo que não realizou o experimento.	<i>Randomized post-test only</i>	R X O R O
		<i>Randomized pre-test – post-test</i>	R O X O R O O
		<i>Randomized pretest – posttest control group</i>	R O X1 O R O X2 O
...			

Tabela 13 - Tipos de estudos empíricos (GRESSE von WANGEHEIM & Shull, 2009).

A tabela 14 apresenta as informações sobre as avaliações aplicadas em cada unidade instrucional, apresentando o objetivo da avaliação, o tipo de estudo empírico,

os instrumentos de coleta de dados e também o número de participantes da avaliação da unidade instrucional.

ID	Objetivo da Avaliação	Tipo de estudo empírico	Instrumento (s) de coleta de dados	Número de participantes da avaliação
1	Coletar informações sobre os participantes, como foi a experiência da oficina, sobre a aprendizagem e também sobre a visão atual sobre Computação e Apps.	Non-Experimental Pre-test – Post-test (O X O).	<i>Survey</i> , vídeo capturando a sala com 2 câmeras, captura de tela dos notebooks, uso dos webcams para capturar imagens dos participantes enquanto trabalhavam nos seus aplicativos, e captura de áudio com gravadores sobre as mesas.	Todos os participantes.
2	Coletar informações sobre como foi a experiência do curso e também sobre a experiência de utilização do App Inventor.	Non-Experimental Pre-test – Post-test (O X O).	<i>Survey</i> aplicado no começo e no fim de cada sessão do curso e entrevistas semiestruturadas com os professores participantes.	Todos os participantes.
3	Coletar informações sobre os participantes referente ao conhecimento do uso de tecnologia, informações sobre a experiência com o curso e verificar a qualidade do curso.	Non-Experimental Pre-test – Post-test (O X O).	<i>Survey</i> no começo e fim do curso.	17 alunas participaram da avaliação inicial e somente 6 da avaliação final.
4	Verificar a motivação para atingir os objetivos propostos, obter informações sobre a facilidade e satisfação da utilização do App Inventor.	Informal	Observação	Todos os participantes

Tabela 14 - Avaliação das UIs.

3.4 Discussão

Dentro do foco do presente artigo foram encontrados poucos artigos apresentando unidades instrucionais para o ensino de computação utilizando o App Inventor. Foram encontrados somente 4 artigos, os quais descreviam unidades instrucionais, e foram considerados relevantes para a pesquisa.

A maioria destes artigos descreve unidades relativamente curtas para unidades em escolas, com exceção do artigo 3 (RAMOS et al., 2014), o qual teve duração de 7 meses. Todas unidades visaram o ensino de computação em um nível introdutório.

O artigo 1 (GROVER; PEA, 2013) e o 2 (CHATZINIKOLAKIS; PAPADAKIS, 2014), apresentaram uma metodologia de ensino interessante, focando muito no diálogo entre os participantes e professores, através da programação em pares e do suporte dos professores no processo de desenvolvimento, o que além de dar sentido aos conceitos computacionais e ao pensamento computacional, acaba por desenvolver um novo vocabulário na prática. Já no artigo 3 (RAMOS et al., 2014) na segunda parte do curso, as alunas foram divididas em grupos de 3 integrantes com o objetivo de desenvolverem todo o processo de criação de um aplicativo, desde a concepção da ideia, planejamento, implementação e apresentação do resultado final, dando uma boa noção sobre o processo de desenvolvimento de software aos participantes.

Identificou-se nas unidades instrucionais aplicadas em países com língua nativa diferente do inglês, os artigos 2 (CHATZINIKOLAKIS; PAPADAKIS, 2014), 3 (RAMOS et al., 2014) e 4 (FINIZOLA et al., 2014), que os participantes obtiveram algumas dificuldades com o desenvolvimento pelo fato de o App Inventor ser em inglês, entretanto atualmente já existem suporte a outras línguas como: Francês, Italiano, Espanhol e outras. No artigo 2 (CHATZINIKOLAKIS; PAPADAKIS, 2014) e 3 (RAMOS et al., 2014), estas dificuldades foram minimizadas com a criação de materiais didáticos nas suas línguas nativas (grego e português).

Estas unidades foram avaliadas, tipicamente por estudos não experimentais, ou seja, foram realizados formulários de avaliação da unidade instrucional no seu começo e final. Além de formulários, também foram coletados dados de diversas formas, incluindo filmagens, captura de vídeo da tela dos computadores, captura de imagem com a webcam dos computadores e observações. Somente o artigo 4 (FINIZOLA et al., 2014) não realizou a aplicação de um formulário, ou algum outro método de coleta de informação para a avaliação da unidade instrucional, somente foi utilizado a observação. Já a avaliação do artigo 3 (RAMOS et al., 2014) foi prejudicada pelo fato do alto número de desistência no curso, ao ter uma avaliação inicial com 17 alunas e final com somente 6 alunas, estas desistências causadas por problema de infraestruturas dos laboratórios, e pelo horário das aulas as quais eram oferecidas no período noturno.

Foi observado que nestas unidades um tamanho de amostra bastante pequenos e não replicados. Porém as avaliações demonstraram que existe uma indicação que o uso do App Inventor permite motivar os alunos e ou para ensinar conceitos computacionais, proporcionando uma experiência interessante e empolgante aos participantes das unidades instrucionais.

Contudo, em geral se observou a carência de estudos com alunos com faixa etária a partir de onze anos a 14 anos (fundamental II) com uma avaliação mais ampla.

3.4.1 Ameaças à Validade da Revisão da Literatura

Existem alguns fatores que podem ter ameaçado a validade da revisão sistemática da literatura. Dentre esses os principais são a possibilidade de não ter sido encontrado alguma unidade instrucional relevante e também uma imprecisão na extração das informações dos artigos encontrados. Para minimizar o risco de não ser encontrado algum trabalho relevante foi utilizado uma ferramenta de busca bastante abrangente, o Google Scholar. Além desta ferramenta abrangente, foi procurado utilizar as palavras chaves mais relevantes no contexto do trabalho, junto com os seus sinônimos e traduções para a língua inglesa. Assim assumiu-se de ter reduzido o risco de não ter encontrado um artigo importante no foco da pesquisa.

Em relação ao processo de extração das informações – foi realizada uma extração cuidadosa e revisado por pesquisador sênior – tentando de representar de forma correta as informações fornecidas.

4 UNIDADE INSTRUCCIONAL “DESENVOLVIMENTO DE APPS COM APP INVENTOR”

Neste capítulo são apresentadas uma análise do contexto e dos aprendizes focados pela unidade instrucional. Também é apresentado o design da unidade instrucional através dos objetivos de aprendizagem, plano de ensino. Além disto, é apresentado os materiais desenvolvidos para a unidade instrucional.

4.1 Análise do contexto e dos aprendizes

O presente trabalho possui como foco escolas e aprendizes situados no estado de Santa Catarina. Isto deve-se ao fato do mesmo estar sendo desenvolvido neste estado e todas as aplicações da unidade instrucional foram realizadas em Santa Catarina. Deste modo, a análise de contexto a seguir tem como foco este estado.

O ensino fundamental II em SC/Brasil possui o seu currículo baseando-se nos Parâmetros Curriculares Nacionais disponibilizado pelo MEC e também pela Proposta Curricular 2014 – Formação Integral para Educação Básica disponibilizado pelo Governo do Estado de SC (GOVERNO DE SANTA CATARINA, 2014). Sobre estes fatores, e pelo fato do ensino de computação não estar contemplado em nenhum dos dois documentos citados é difícil enxergar um cenário onde a computação esteja sendo ensinada aos estudantes do ensino fundamental II.

Contudo no Brasil, em torno de 50% das escolas da rede pública do ensino fundamental possuem acesso à internet e 51% de todas as escolas da rede pública do ensino fundamental possuem um laboratório de informática (INEP, 2014). Já nas escolas da rede privada do ensino fundamental, 93% possuem acesso à internet e 55% de todas as escolas da rede privada do ensino fundamental possuem laboratório de informática (INEP, 2014).

Já a Região Sul se destaca por ter os melhores índices nestes dois quesitos, possuindo acesso à internet a 85% das escolas da rede pública do ensino fundamental e laboratório de informática em 80% de todas as escolas da rede pública do ensino fundamental (INEP, 2014). Em 99,5% escolas da rede privada do ensino fundamental possuem acesso à internet e 79% de todas as escolas da rede privada do ensino fundamental na Região Sul possuem laboratório de informática (INEP, 2014).

O Programa Nacional de Tecnologia Educacional (ProInfo) do MEC é o responsável pela compra, distribuição e instalação de laboratórios de informática nas escolas públicas de educação básica. Este programa incentiva a utilização de softwares livres e produz conteúdo específicos, voltados para o uso didático-pedagógico,

associados ao sistema operacional Linux-Educacional, o qual é uma versão baseada no Debian e atualmente está na versão 5.0 e acompanha todos os computadores fornecidos pelo programa (PROINFO, 2015). Entretanto, podem existir escolas públicas e privadas que utilizem outro sistema operacional livre ou proprietário. Sendo assim, o App inventor é compatível com os sistemas operacionais Linux, Windows e Macintosh (APP INVENTOR, 2014).

Apesar da existência de laboratórios de informática em grande parte das escolas da educação básica, muitos dos professores são incapazes de utilizar estas tecnologias (GOOGLE, 2015). E também é rara a existência de professores de computação em escolas de educação básica (LOPES, 2015).

Os aprendizes do ensino fundamental II, atualmente conhecidos como a Geração Z, pertencem a primeira geração nativamente digital, isto é, já nasceram em um mundo digital. E o que os distingue das outras gerações é o fato de que eles são a geração mais eletronicamente conectada da história. Desde a infância, eles cresceram em um mundo envolvido por tecnologia, utilizando *notebooks*, celulares, plataforma de mensagens instantâneas, vídeo games, internet e etc. (SAVAGE, 2006). Entretanto, esse fato positivo não está sendo utilizado para ensinar conhecimentos de computação para as crianças desta geração, as quais já possuem um alto potencial de aprendizado e motivação, em grande parte devido ao ambiente computacional ao seu redor, contudo poucos possuem conhecimento sobre computação (GOOGLE, 2015).

Por se tratar de uma unidade instrucional com o objetivo de desenvolver um aplicativo para *smartphone*, é importante que o público alvo tenha acesso a esta tecnologia. Segundo o IBGE 77% de crianças com mais de dez anos as quais possuem celulares em Santa Catarina (IBGE, 2013). Muitos desses celulares podem ser caracterizados como *smartphones*, devido ao fato de que os *smartphones* já correspondem a 77% do mercado de celulares no Brasil (FOLHA, 2015). Também podemos inferir que o Android é o sistema operacional em uso nestes smartphones, por causa da diferença na faixa de preço dos aparelhos comparando ao iOS e a fatia de mercado em que o Android está presente em mais de 75% de todos os *smartphones* vendidos.

4.2 Design da unidade instrucional

Neste contexto, o presente trabalho apresenta o desenvolvimento de uma unidade instrucional para ensino de computação através do desenvolvimento de aplicativos no ensino fundamental II utilizando o App Inventor. Esta unidade possui

como objetivo ensinar conceitos básicos de computação e desenvolvimento de aplicativos para os alunos, assim como, despertar o interesse pela área de computação.

Com base na análise de contexto, são selecionados os objetivos de aprendizagem para esta unidade instrucional (Tabela 15), de acordo com as diretrizes o currículo de referência CSTA/ACM K-12 para o nível 2 de ensino de computação apresentados na tabela 5.

Objetivo Geral	O objetivo geral desta unidade instrucional é ensinar ao aluno conceitos básicos da computação, principalmente relacionado à prática/programação e ao pensamento computacional.
Ao final da UI o aluno será capaz de:	
Pensamento Computacional	[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). [O3] Definir um algoritmo, como sendo uma sequência de instruções que podem ser processadas por um computador. [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).
Colaboração	[O17] Colaborativamente criar, desenvolver, publicar e apresentar produtos (p.ex., vídeos, podcasts, sites), utilizando recursos tecnológicos que demonstram e comunicam conceitos do currículo. [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.
Práticas Computacionais e de Programação	[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. [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.
Computadores e Dispositivos de Comunicação	[O30] Identificar uma variedade de dispositivos eletrônicos que contêm processadores computacionais.

Tabela 15 - Objetivos de Aprendizagem da Unidade Instrucional

Com o intuito de alcançar os objetivos de aprendizagem definidos e levando em consideração o contexto, é proposta uma unidade instrucional com o período de duração de único encontro de 4 horas.

Durante o período da unidade instrucional será apresentado o App Inventor, o modelo de trabalho do App Inventor através do desenvolvimento no Designer e no Editor de Blocos, a metodologia de desenvolvimento em blocos de código e outros conceitos necessários ao desenvolvimento de um aplicativo no App Inventor. Também será desenvolvido um aplicativo pré-definido, o qual o professor já possui o código

pronto e o roteiro de desenvolvimento, em duplas de alunos seguindo passo a passo as instruções do professor. As funcionalidades desenvolvidas no aplicativo tem a intenção de atingir os objetivos de aprendizagem descritos na tabela 15. O plano de ensino desenvolvido para esta unidade instrucional está descrito na tabela 16.

PLANO DE ENSINO

Fundamental II		Computação – Desenvolvimento de apps com App Inventor	
Objetivos de aprendizagem relacionados a Computação			
<p>Objetivo geral: O objetivo geral desta unidade instrucional é ensinar ao aluno conceitos básicos da computação, principalmente relacionado à prática/programação e ao pensamento computacional.</p> <p>Após esta unidade instrucional, o aluno deverá ser capaz de:</p> <ul style="list-style-type: none"> • Conhecer conceitualmente linguagens de programação (Cognitivo/Conhecimento) • Conhecer o que é e como funciona um algoritmo (Cognitivo/Conhecimento) • Descrever o que é o ambiente App Inventor e o que pode ser feito com este ambiente (Cognitivo: Conhecimento e Compreensão) • Usar o ambiente App Inventor (entrar no site, criar um projeto, acessar projetos, compartilhar projetos) (Cognitivo: Aplicação) • Usar vários conceitos de programação: inicialização de variáveis, métodos, entrada via teclado, componentes de layout, sequência, loop, eventos, condicionais, mensagens, concorrência, etc. (Cognitivo: Aplicação) • Criar um aplicativo no ambiente App Inventor, testar e instalar em um smartphone ou tablet(Cognitivo: Aplicação) • Descrever e aplicar o ciclo de engenharia de software (Cognitivo: Conhecimento, compreensão e Aplicação) • Avaliar e criticar construtivamente aplicativos para smartphones ou tablets(Cognitivo: Análise). 		<p>Pensamento Computacional:</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>[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>Colaboração:</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>Práticas Computacionais e de Programação:</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>[O24] Implementar soluções de problema utilizando uma linguagem de programação, incluindo: o comportamento de laços (seqüências de instruções que se repetem), instruções condicionais, lógica, expressões, variáveis e funções.</p> <p>Computadores e Dispositivos de Comunicação:</p> <p>[O30] Identificar uma variedade de dispositivos eletrônicos que contêm processadores computacionais.</p>	
Conteúdo			
<ul style="list-style-type: none"> • Conceber, programar um aplicativo com App Inventor e testar no dispositivo móvel. 			

<ul style="list-style-type: none"> • Ensinar os métodos de desenvolvimento de aplicativos do App Inventor. Apresentando as funções existentes no App Inventor e como podem ser utilizadas. • Ensinar os conceitos de programação unindo os blocos de códigos no Editor de Blocos. 				
Sequência dos assuntos abordados	Duração (minutos)	Estratégia instrucional	Recursos	Avaliação
			Termo de Consentimento Livre e Esclarecido	
<i>M0. Medição</i>	<i>Antes da Unidade</i>	<i>Questionário Aluno Pré-unidade</i>		
1. Apresentação da Unidade Instrucional: Apresentação dos objetivos da unidade e das ferramentas utilizadas no processo.	15	Aula expositiva	Slides	
2. Conhecer App Inventor: - Realizar login com conta padrão da oficina - Apresentar o modelo de trabalho (Designer, Blocks)	15	Aula expositiva	Slides Website do App Inventor Contas para login Tutorial online	--
3. Apresentação do aplicativo Caça Mosquito. -Apresentação do aplicativo e seus objetivos -Divisão das duplas	20	Aula expositiva	Slides Aplicativo Caça Mosquito Lista de Grupos	
4. Concepção do aplicativo Caça Mosquito, em duplas. -Todas as duplas devem desenvolver o mesmo aplicativo. - Concepção do aplicativo	40	Atividade prática sendo guiado passo-a-passo pelo professor	Slides Aplicativo Caça Mosquito Roteiro do professor Website do App Inventor	--
Intervalo	15			
3. Programação de funcionalidades mais específicas no aplicativo pré-definido.	45	Todos os alunos: podem seguir o roteiro para desenvolverem sozinhos, as funcionalidades específicas do Aplicativo Caça	Slides Roteiro Caça Mosquito Website do App	--

		Mosquito. Desenvolvimento em dupla, podendo ter auxílio do professor.	Inventor	
4. Personalização do aplicativo Caça Mosquito.	30	Desenvolvimento em dupla, podendo ter auxílio do professor. Utilizando a criatividade para personalizar o Design do aplicativo.	Exemplos da Galeria do App Inventor Documentação dos componentes.	Rubrica
5. Compartilhamento, experimentação dos jogos da turma.	20	Exportação do projeto desenvolvido. Discussão.	Roteiro Exportação e Importação de Projetos no App Inventor	--
M1. Medição	10	<i>Questionário Aluno Pós – unidade</i> <i>Questionário Família Pós-unidade</i> <i>Questionário Professor Pós-unidade</i>		

Tabela 16 - Plano de Ensino

O aplicativo pré-definido desenvolvido durante a unidade instrucional é um jogo chamado “Caça Mosquito”. Este jogo foi desenvolvido com a ideia de colocar a imagem de um mosquito voando pela tela para que o jogador tente matá-lo. Após clicar no botão iniciar, será ativado um cronômetro dando um determinado tempo para o jogador apertar o mosquito 3 vezes. Caso o jogador consiga, ele se sai vitorioso. Caso o tempo se esgote, ele é derrotado. Existe a possibilidade de o jogador escolher o nível de dificuldade em que ele pretende jogar. Cada nível de dificuldade maior aumenta a velocidade em que o mosquito se movimenta e diminui o tempo disponível.

Para a aplicação da unidade instrucional ser efetiva deve-se verificar os pré-requisitos de hardware, software e entre outros necessários, os quais estão apresentados na tabela 17.

PRÉ-REQUISITOS PARA APLICAÇÃO DA UNIDADE INSTRUCIONAL	
SOFTWARE	Navegador Web Firefox ou Chrome – utilizado para acessar o App Inventor.
	Opcional: Emulador App Inventor – usado em casos em que os alunos não possuem dispositivos móveis e deste modo podem acompanhar o desenvolvimento do aplicativo através do emulador
HARDWARE	Computador com monitor, mouse e teclado – idealmente 2 alunos por computador e um para o instrutor.

	Dispositivo móvel com sistema operacional Android versão 2.3 ou superior – utilizado para acompanhar o desenvolvimento do aplicativo.
	Acesso à internet nos computadores e nos dispositivos móveis – utilizados no desenvolvimento do aplicativo através do App Inventor.
	Projetor multimídia – utilizado para a apresentação dos slides e deve estar conectado ao computador do instrutor.
OUTROS	Conta no Gmail para cada dupla e para o instrutor – criadas pelo instrutor e para fornecer o acesso ao App Inventor.

Tabela 17 - Pré-requisitos para aplicação da unidade instrucional

4.3 Desenvolvimento da unidade instrucional

De acordo com o plano de ensino definido foram desenvolvidos vários materiais instrucionais.

Roteiros para o instrutor: Foram desenvolvidos três roteiros para o uso do instrutor durante a oficina.

Roteiro para criação de contas Gmail e *login* na ferramenta App Inventor que explica como realizar o cadastro de uma conta de *e-mail* Gmail, a qual é necessária para o *login* no App Inventor, e como realizar o *login* no App Inventor.

O roteiro para o desenvolvimento do aplicativo Caça Mosquito (Figura 13), o qual explica o desenvolvimento do aplicativo no App Inventor. Ilustrando o modo de desenvolvimento de encaixe dos blocos de código e também o modo de arrastar e soltar os componentes na tela.

Vamos alterar as propriedades da pintura para que ela possua altura de 300 pontos e largura de 300 pontos. Para isto, clique e selecione o componente "Pintura1" e altere o valor da sua altura e largura.

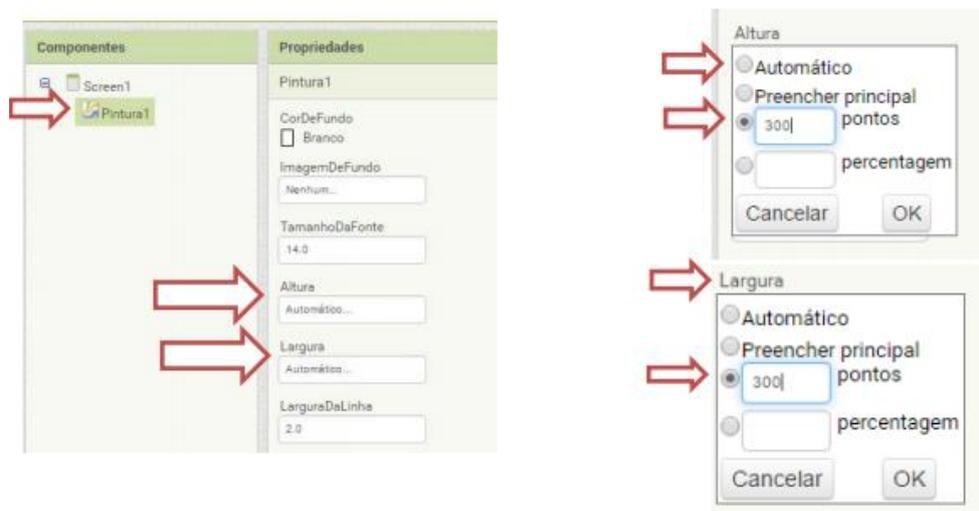


Figura 13 –Parte do Roteiro Caça Mosquito

E também foi desenvolvido um roteiro explicando o processo de importação e exportação de projetos no App Inventor. Este processo é fundamental para o compartilhamento do código desenvolvido pelos alunos para o instrutor.

Os roteiros são voltados a ser usado pelo instrutor para explicar o passo-a-passo de como criar uma conta de *e-mail* no Gmail, desenvolver o aplicativo Caça Mosquito e também como exportar o código do App Inventor.

Aplicativo Caça Mosquito: Foi desenvolvido o jogo Caça Mosquito (Figura 14) para ser utilizado junto a oficina com o intuito de ensinar os conceitos definidos no plano de ensino através do App Inventor. Está disponível o código do aplicativo no CD dos códigos fonte do presente trabalho. Todo os passos necessários para o desenvolvimento do jogo são descritos no Roteiro Caça Mosquito.

O jogo Caça Mosquito abrange conceitos de computação (lógica e *design*), na tabela 18 é informado quais blocos de código são utilizados e para que são utilizados os no jogo.

Categorias de blocos de código de comportamento geral utilizados no jogo Caça Mosquito	
Categorias	Descricao
Controle	São utilizados para realizar o controle de quando o jogo é finalizado, seja por tempo esgotado ou caso o número de vidas do mosquito é igual a zero.
Lógica	São utilizadas comparações para verificar se o número de vidas do mosquito é igual a zero, ou se o tempo restante é igual a zero. Também é utilizado os valores verdadeiro ou falso para ativação ou desativação de componentes da tela, por exemplo, desativar os cliques no mosquito enquanto ele estiver com a vida igual a zero.
Matemática	São utilizados os blocos de número para inicializar as variáveis com valores numéricos, a operação de subtração para decrementar o número de vidas do mosquito e do tempo restante para jogar e também foi utilizado o bloco de selecionar um número aleatório em um intervalo.
Texto	São utilizados os blocos para unir dois textos e também o bloco de texto para informar o texto desejado.
Variáveis	São utilizados os blocos para inicialização de uma variável, bloco de obtenção do valor de uma variável e o bloco para alterar o valor de uma variável inicializada.
Procedimentos	São utilizados blocos de criação de um procedimento com e sem valores de entrada e blocos para a chamada destes procedimentos.

Tabela 18 - Categorias de blocos de código utilizados no jogo Caça Mosquito

Na tabela 19 é informado quais componentes de Interface são utilizados no jogo e com qual função foram utilizados. Estes conjuntos de categorias de blocos de código e componentes de Interface definido no App Inventor estão detalhados nas tabelas 6 e 7.

Componentes de Interface da Janela Designer	
Componentes	Função
Botão	botão com a funcionalidade de detectar cliques. Muitas propriedades podem ser configuradas.
Legenda	É um componente que apresenta um texto na tela.
Pintura	É um botão que ao clicado, apresenta uma janela com uma lista de texto para ser selecionado pelo usuário.
SpriteImagem	É um componente para apresentar uma imagem, podendo especificar qual a imagem a ser apresentada.
Temporizador	É um componente para a visualização de páginas web.
Som	É um componente que apresenta uma janela com uma lista de elementos selecionáveis.

Tabela 19 – Componentes de Interface utilizados no jogo Caça Mosquito



Figura 14 - Imagem do Aplicativo Caça Mosquito

Slides: Foi desenvolvido um conjunto de *slides* (Figura 15) para o uso do instrutor durante a aplicação da unidade instrucional com o intuito de auxiliar no processo de ensino e aprendizagem.

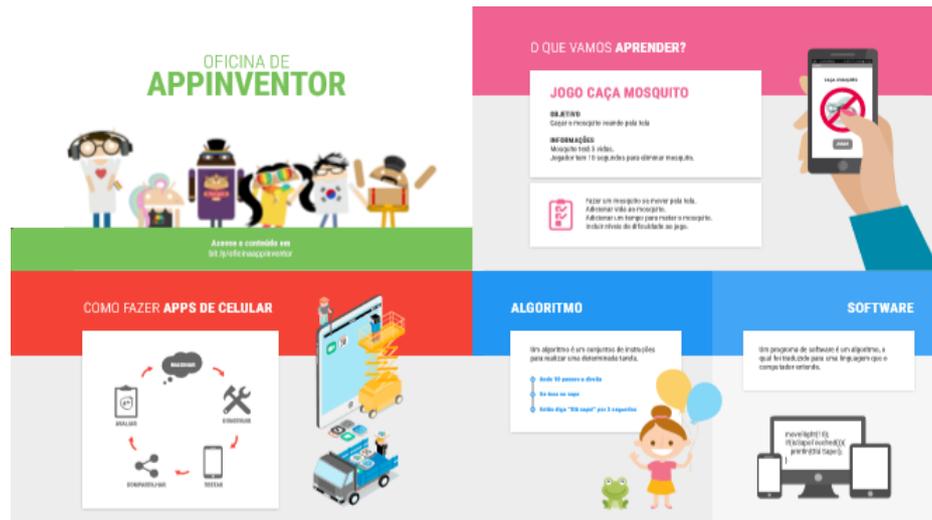


Figura 15 - Exemplos dos slides

Tarefa de casa: para fixar o conhecimento aprendido durante a unidade instrucional e também podendo ser utilizada como método de avaliação foi desenvolvida uma tarefa de casa (Figura 16). Esta tarefa aborda conceitos de operadores aritméticos, variáveis, expressões booleanas, estruturas de controle, procedimentos e eventos.

Explique como ficará o valor da variável var de acordo com o procedimento a seguir:

Procedimento:

```

para procedimento
fazer
se
obter global var - = 1
então
ajustar global var - para obter global var - + 3

```

Exemplo	<p>A variável var é igual a 1 no começo.</p> <p>É chamado o procedimento. No procedimento temos que $var = 1$ é verdadeiro. Então o valor de var é ajustado para 4.</p> <p>Valor final da variável: 4</p>	<pre> ajustar global var - para 1 chamar procedimento - </pre>
Questão 8	<p>A variável var é igual a 4 no começo</p> <p>É chamado o procedimento. Nele a comparação $var = 1$ é falsa. Logo o valor da variável var não é alterado.</p> <p>Valor final da variável: 4</p>	<pre> ajustar global var - para 4 chamar procedimento - </pre>

Figura 16 – Parte da Tarefa de casa

Prova de avaliação: Outra atividade para a avaliação dos alunos desenvolvida foi a prova final (Figura 17). Esta prova aborda conceitos sobre o ciclo de engenharia de

software utilizado no desenvolvimento de aplicativos, o que é um aplicativo e também questões de programação seguindo o padrão do App Inventor.

3. (1 ponto) – Marque todas as opções sobre o que é possível fazer no App Inventor.

<input type="checkbox"/> Jogar um jogo	<input type="checkbox"/> Compartilhar um jogo na galeria	<input type="checkbox"/> Acessar rede sociais	<input type="checkbox"/> Desenvolver um aplicativo
<input type="checkbox"/> Compartilhar um aplicativo	<input type="checkbox"/> Encontrar aplicativos na galeria	<input type="checkbox"/> Criar um jogo	<input type="checkbox"/> Ver o código de outros aplicativos

4. (1 ponto) - Onde o mosquito ficará quando for clicado no Botão? Faça um "X" no sistema de coordenadas abaixo onde aproximadamente o mosquito irá se posicionar.

quando Botao .Clique

fazer

- ajustar mosquito . X para 240
- ajustar mosquito . Y para -180



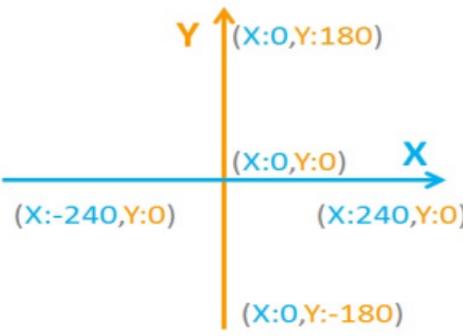


Figura 17 – Parte da Prova Final

Rubrica: Foi definida uma rubrica para definir os critérios de medição do aprendizado dos alunos através dos resultados obtidos no aplicativo desenvolvido e através da tarefa de casa e prova final (Figura 18).

Rubrica – Atingimento dos objetivos da unidade	
Este documento apresenta como avaliar os objetivos de aprendizagem da Unidade Instrucional App Inventor.	
Recursos de avaliação	
<ul style="list-style-type: none"> • Aplicativo Caça Mosquito. (nota de 0 a 10) • Tarefa de casa (nota de 0 a 10) • Prova Final (nota de 0 a 11) 	
Objetivos	
Compreender o que é um algoritmo e usar os passos básicos de algoritmos para a resolução de problemas. Descrever o que é e o que pode ser feito no ambiente App Inventor. Compreender conceitos de programação, como: operadores, variáveis e estruturas condicionais. Descrever o ciclo de engenharia de software para desenvolvimento de aplicativos móveis.	
MEDIDA: Tarefa de casa * 0,5 + Prova final * 0,5	
Conceber, desenvolver, testar, publicar, aplicar o ciclo de engenharia de software e apresentar um aplicativo desenvolvido no App Inventor. Utilizar conceitos básicos de programação (inicialização, laços, expressões lógicas, e etc.	
MEDIDA: Através do Aplicativo Caça Mosquito desenvolvido por todos os alunos. Deve-se verificar se as seguintes etapas do Roteiro Caça Mosquito foram realizadas.	
Etapas:	
1. Criar o projeto no App Inventor e fazer o mosquito se mover pela tela. (2 pontos)	

Figura 18 – Parte da Rubrica

Material de avaliação da unidade instrucional

Conforme o plano de medição voltado para a avaliação da própria unidade instrucional em termos de percepção, auto avaliação e motivação do aluno antes/depois da unidade instrucional.

Questionários de avaliação: Foram elaborados para o aluno um questionário pré (Figura 19) e um pós unidade instrucional. E inclusive, para avaliar a percepção do instrutor após a unidade instrucional, foi desenvolvido um questionário pós unidade.

COMPUTAÇÃO NA ESCOLA		Questionário pré-unidade (Aluno)		Unidade escolar App Inventor	
Escola:	Turma:	Data: / /			
Nome completo:	Idade:	Você é: <input type="checkbox"/> Menina <input type="checkbox"/> Menino			
Você já teve aulas para aprender a fazer aplicativos para smartphones?					
<input type="checkbox"/> Sim <input type="checkbox"/> Não					
Conte-nos o que você sabe:					
Você sabe criar um aplicativo para smartphones?			Você sabe ensinar alguém como criar um aplicativo para smartphones?		
<input type="checkbox"/> Sim <input type="checkbox"/> Não			<input type="checkbox"/> Sim <input type="checkbox"/> Não		

Figura 19 – Parte do Questionário pré unidade

Informativo: Foi desenvolvido um informativo (Figura 20) com o intuito de apresentar a unidade instrucional, a ferramenta App Inventor e sobre a importância do ensino de computação, e também, motivar os pais e os participantes para participação na unidade instrucional.



Figura 20 - Parte do Informativo

Todos os materiais desenvolvidos estão disponíveis por completo sob a licença Creative Commons Atribuição-NãoComercial-CompartilhaIgual 4.0 Internacional no website da computação na escola, através do link: <http://www.computacaonaescola.ufsc.br/?page_id=1474>.

Algumas atividades, incluindo o auxílio na preparação dos materiais didáticos e o design gráfico dos mesmos foram realizadas por bolsistas de iniciação científica da Iniciativa Computação na Escola (CnE).

Além destes materiais, é utilizada a ferramenta para desenvolvimento de aplicativos App Inventor disponibilizado gratuitamente pelo MIT.

5 APLICAÇÃO E AVALIAÇÃO

Neste capítulo é apresentado a definição do método de avaliação da unidade instrucional. Também será contemplado neste capítulo a descrição das aplicações da unidade instrucional realizadas, assim como, a análise da avaliação da unidade instrucional seguindo o plano de medição.

5.1 Definição da avaliação

O objetivo da avaliação consiste em explorar e compreender os aspectos relacionados a unidade instrucional no ensino de computação no ensino fundamental através do desenvolvimento de aplicativos.

A partir deste objetivo e seguindo a abordagem definida no método GQM (BASILI; CALDIERA; ROMBACH, 1994), é definido um plano de medição (Tabela 20), o qual decompõe os objetivos em perguntas de análise e medidas operacionalizando-as nos instrumentos de coleta de dados.

Para atingir o objetivo da avaliação é realizado um estudo de caso a fim de compreender os fenômenos observados durante a aplicação da unidade instrucional desenvolvida. Conforme descrito no Plano de Ensino (Tabela 16), o estudo de caso realizado contém uma medição antes e outra após a unidade instrucional, capturando a percepção dos alunos e coletando os dados através de questionários. Também é realizado uma avaliação dos instrutores em relação a unidade instrucional e o impacto nos alunos.

Pergunta de análise	Medida (s)	Instrumento de coleta de dados
PA1. Os objetivos de aprendizagem são atingidos utilizando a unidade instrucional?	M1.1 Grau de aprendizagem referente a capacidade de fazer aplicativos para <i>smartphones</i> . M1.2 Grau de aprendizagem referente à capacidade de descrever, analisar e programar uma sequência de instruções. M1.3 Grau de aprendizagem referente à capacidade do uso do App Inventor para desenvolver um jogo em duplas, instalar no <i>smartphone</i> e compartilhar. M1.4 Grau de habilidade para ensinar o aprendido a outras pessoas.	- Rubrica - Questionário aluno pré/pós-unidade - Questionário instrutor pós-unidade - Ambiente de desenvolvimento App Inventor - Observações durante a aula

PA2. A unidade instrucional facilita a aprendizagem?	M2.1 Grau de facilidade da aula. M2.2 Grau de facilidade de fazer aplicativos para <i>smartphones</i> . M2.3 Grau de qualidade geral da aula. M2.4 Pontos fortes em relação à facilidade da aula. M2.5 Pontos fracos em relação a dificuldade da aula.	- Questionário aluno pós-unidade. - Observações durante a aula.
PA3. A unidade instrucional promove uma experiência de aprendizagem agradável e divertida?	M3.1 Grau de diversão da aula. M3.2 Grau da imersão das aulas. M3.3 Grau da interação social (querer mostrar aos outros). M3.4 Opinião subjetiva sobre a aula. M3.5 Pontos fortes em relação à experiência das aulas. M3.6 Pontos fracos em relação à experiência das aulas.	- Questionário aluno pós-unidade - Observações durante a aula.
PA4. A unidade instrucional proporciona uma percepção positiva da computação?	M4.1 Vontade de aprender computação na escola. M4.2 Grau de satisfação em fazer aplicativos para <i>smartphones</i> . M4.3 Grau de diversão em fazer aplicativos para <i>smartphones</i> .	- Questionário aluno pré/pós-unidade - Observações durante a aula. - Questionário pós-unidade instrutor

Tabela 20 - Plano de medição da unidade instrucional

5.2 Aplicação

A unidade instrucional desenvolvida foi aplicada com dois grupos de alunos do ensino fundamental no primeiro semestre de 2016, ambas numa escola pública, a Escola Profa. Herondina Medeiros Zeferino em Florianópolis/SC. Participaram das aplicações um total de 36 alunos, conforme detalhado na tabela 21.

Turma	Local de aplicação	Quantidade de participantes	Período da aplicação
Turma 1	Escola Profa. Herondina Medeiros Zeferino	18	Matutino
Turma 2	Escola Profa. Herondina Medeiros Zeferino	18	Vespertino

Tabela 21 - Resumo dos participantes e dados coletados

Os alunos foram convidados a participar da oficina pela professora de informática da escola, todos participaram voluntariamente no contra turno das suas aulas. Nos dias que precederam a oficina, foi entregue aos interessados o informativo, os Termos de consentimento livre e esclarecido (TCLE) e de assentimento

da criança, também foram entregues para o preenchimento o questionário pré-unidade instrucional.

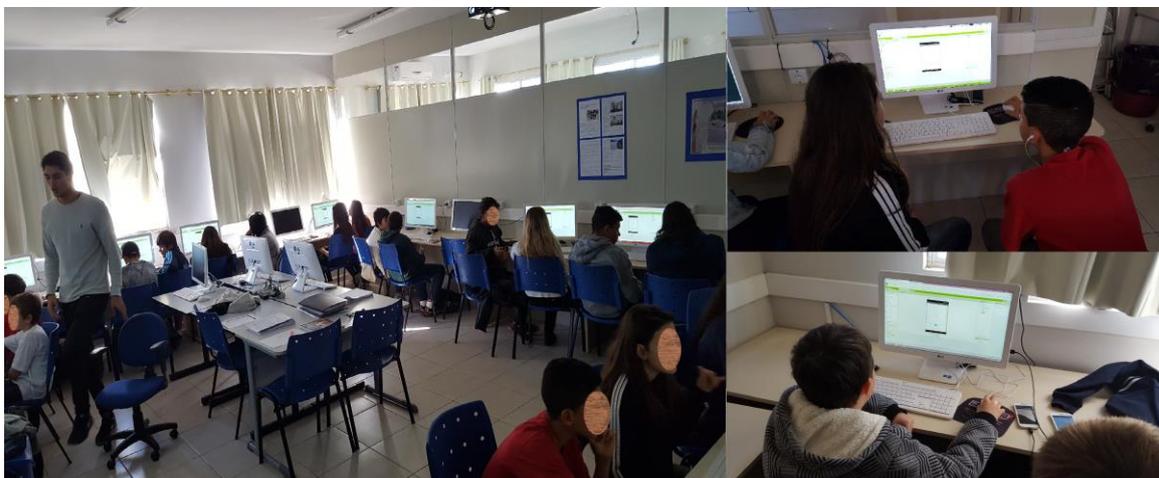


Figura 21 - Fotos das aplicações realizadas.

As aulas realizadas nas dependências da Escola Profa. Herondina Medeiros Zeferino foram ministradas pelo autor do presente trabalho e contou com auxílio da professora de tecnologia educacional da própria escola. Na segunda oficina também auxiliou mais um aluno de graduação do INE na oficina. As duas aplicações foram realizadas na sala de informática da escola, a qual conta com 18 computadores, todos configurados com o sistema operacional Microsoft Windows versão 8.1, contam com 4 GB de memória RAM e conectados à internet através de uma rede por cabos.

No início da unidade instrucional os alunos entregaram os TCLEs e os questionários pré-unidade instrucional preenchidos.

Os alunos participantes foram orientados a formarem duplas para a realização das atividades. O conteúdo abordado foi o desenvolvimento do aplicativo Caça Mosquito, o qual foi desenvolvido passo a passo com o instrutor, onde foram implementadas todas as funcionalidades, com exceção da atividade Adicionar Seleção de Dificuldade, todas descritas no Roteiro Caça Mosquito (Figura 13).

Todos as duplas participantes dos dois encontros possuíam dispositivos móveis, os quais eram de propriedade dos próprios participantes, e compatíveis para realizar o desenvolvimento junto ao App Inventor. Os dispositivos móveis foram conectados na internet através da rede sem fio disponível na biblioteca, a qual se localiza ao lado da sala de informática.

Seguindo o Plano de Ensino, ao final da oficina os alunos puderam instalar uma versão final do aplicativo nos próprios dispositivos e foram instruídos de como proceder para continuar o desenvolvimento do aplicativo nas suas casas.

Ao final os alunos responderam o questionário pós-unidade, e a professora de tecnologia educacional, a qual auxiliou na unidade instrucional, também realizou o preenchimento do questionário de avaliação da unidade instrucional para o professor da turma.

Ambas as aplicações seguiram o mesmo roteiro e abordaram os mesmos conteúdos. Entretanto, durante a segunda aplicação ocorreu um problema na rede sem fio, ocasionando problemas para testar o desenvolvimento no smartphone. Contudo a conexão foi reestabelecida momentos depois e foi possível continuar o desenvolvimento normalmente.

5.3 Análise da avaliação

Com objetivo de avaliar a unidade instrucional, em relação ao seu impacto no ensino de computação para alunos do ensino fundamental, foi realizada uma avaliação com coleta de dados por meio de questionários dos alunos e da professora/instrutores participantes, conforme previamente definida (vide Seção 5.1). Também foram analisadas as observações dos instrutores e da professora envolvidos. A tabela 22 apresenta um resumo dos dados coletados durante o estudo.

Turma	Quantidade de questionários pré unidade respondidos por alunos	Quantidade de questionários pós unidade respondidos por alunos	Quantidade de questionários respondidos pelo professor/instrutor da disciplina
Turma 1	16	17	2
Turma 2	19	18	3

Tabela 22 - Resumo dos dados coletados

Na tabela 23 são apresentados dados demográficos simples sobre as turmas. Foi verificado que não ocorreu tanta disparidade entre o número de meninos e meninas.

Turma	Quantidade de meninos	Quantidade de meninas	Média das idades
Turma 1	11	7	12 anos
Turma 2	11	7	13 anos

Tabela 23 - Dados demográficos das turmas.

Os dados coletados por meio dos questionários e observações foram analisados por meio de análises qualitativas e quantitativas utilizando estatística descritiva.

Observando os alunos durante as aulas foi possível perceber que, apesar das distrações estarem em um alcance próximo, no *smartphone*, todas as equipes conseguiram criar o jogo Caça Mosquito. A maioria dos alunos participou durante a aplicação, seja fazendo perguntas, respondendo às perguntas feitas pelo instrutor e pedindo auxílio para resolver algum problema. Também foi observado o entusiasmo e vontade de trabalhar dos participantes ao ver o resultado se materializar na tela do *smartphone*.

Para detalhar melhor os resultados obtidos, são apresentados em seguida os resultados para cada pergunta de análise, definidas na Tabela 20.

5.3.1 Os objetivos de aprendizagem são atingidos através da unidade instrucional?

Para analisar se os objetivos de aprendizagem, definidos na seção 4.2 (Tabela 15), foram atingidos, foi realizada uma análise dos aplicativos desenvolvidos pelos alunos durante a oficina e uma análise das observações dos instrutores e da professora de tecnologia. A Tabela 24 apresenta os resultados dessa análise.

Análise	Objetivos de aprendizagem
Todos os alunos atingiram o objetivo	[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).
	[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
	[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.
A maioria dos alunos atingiram o objetivo	[O3] Definir um algoritmo, como sendo uma sequência de instruções que podem ser processadas por um computador.
	[O17] Colaborativamente criar, desenvolver, publicar e apresentar produtos (p.ex., vídeos, podcasts, sites), utilizando recursos tecnológicos que demonstram e comunicam conceitos do currículo.
	[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).
	[O30] Identificar uma variedade de dispositivos eletrônicos que contêm processadores computacionais.
A minoria atingiu o objetivo	--

Nenhum aluno atingiu o objetivo	--
---------------------------------	----

Tabela 24 - Análise do atingimento dos objetivos.

Todos os alunos tiveram facilidade no desenvolvimento do jogo através do App Inventor, demonstrando que tanto a interface de desenvolvimento dividida em Blocos e Designer e o desenvolvimento com blocos de código foram intuitivos. Devido a metodologia de ensino da unidade instrucional ser baseado no roteiro predefinido, todos os alunos desenvolveram o mesmo aplicativo e utilizaram os mesmos conceitos de programação incluídos no jogo.

Deste modo, foi analisado os códigos desenvolvidos para verificar até qual funcionalidade foi desenvolvida pelos alunos, idealmente, deveriam desenvolver todas as funcionalidades do jogo com exceção da funcionalidade 3. Adicionar Nível de Dificuldade (vide Roteiro Caça Mosquito, Seção 4.3). E com base nessa análise, observações do instrutor, e no modo de conduzir a unidade instrucional, foi identificado que ao final todas as duplas conseguiram chegar ao final do desenvolvimento do jogo Caça Mosquito. Entretanto nem todos os alunos instalaram a versão final do aplicativo no smartphone, ocorreram casos em que o *smartphone* não possuía espaço suficiente, e também casos onde somente um aluno da dupla possuía o *smartphone*.

Durante a aplicação da unidade instrucional, sempre que um novo conceito de computação iria ser utilizado para desenvolver o jogo, foi feita uma explicação pelo instrutor utilizando os *slides* (Seção 4.3). E nos momentos em que um conceito já explicado fosse ser utilizado novamente, o instrutor realizou perguntas aos alunos para verificar se eles saberiam como solucionar este novo problema utilizando os conceitos aprendidos.

Com essas informações foi identificado que houve uma boa introdução e aprendizagem aos conceitos de computação. Os alunos tiveram um contato e aplicaram desde conceitos simples como operações lógicas, uso de variáveis até conceitos mais complexos como eventos. Estas conclusões também são percebidas nas respostas dos questionários dos alunos, os quais antes da unidade instrucional a maioria se considerava incapaz de fazer um aplicativo para *smartphone*, diferentemente das respostas ao final em que a maioria se considerava capaz de fazer um aplicativo (Figura 22).

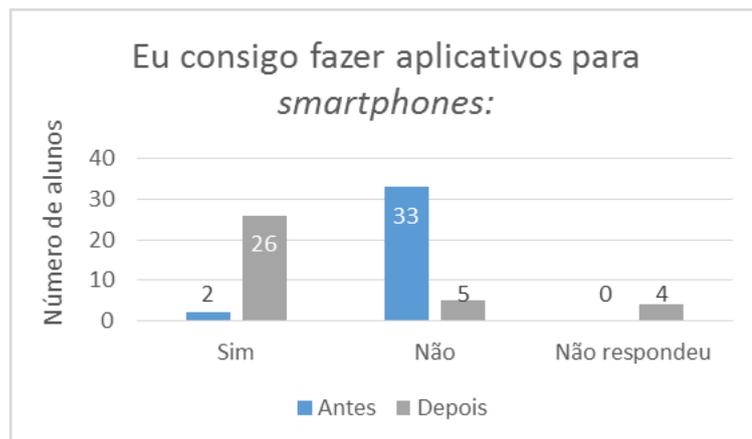


Figura 22 - Eu consigo fazer aplicativos para *smartphones*

Entretanto, uma grande parte dos alunos não se considera apto para ensinar a um amigo/amiga, dando dúvidas quanto ao nível de aprendizagem obtido durante a unidade instrucional (Figura 23).



Figura 23 - Consigo explicar para um amigo/amiga como fazer um aplicativo para *smartphones*.

A colaboração foi de extrema importância para a dinâmica da aplicação a programação em pares. Entretanto, foi verificado alguns problemas referente a isso com a aplicação, por exemplo, alguns alunos não gostaram de somente um deles mexer no computador, deste modo foi solicitado para se revezarem nos papéis.

5.3.2 A unidade instrucional facilita a aprendizagem?

A maioria dos alunos achou a unidade instrucional muito fácil ou fácil, porém oito alunos consideraram difícil (Figura 24). De acordo com as observações do instrutor e pela professora presente, foi identificado que o desenvolvimento do jogo “Caça

Mosquito” envolve muitos conceitos de computação e isso dificulta o entendimento dos alunos.

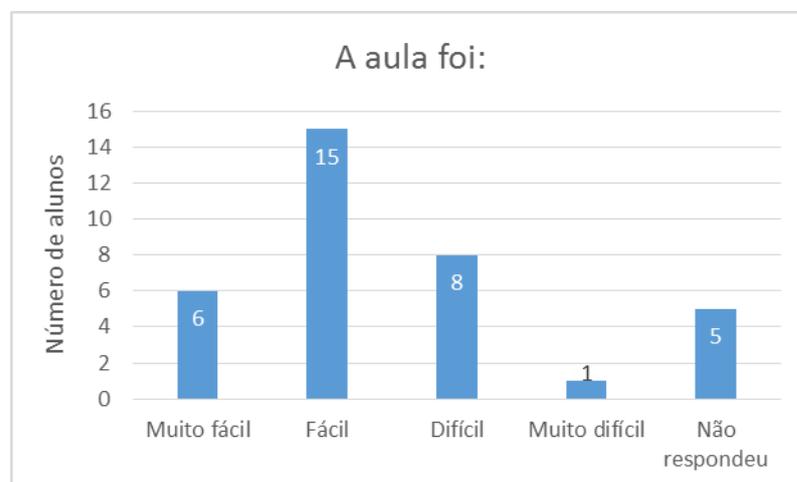


Figura 24 - A aula foi fácil?

Também foi observado pelo instrutor uma maior facilidade em desenvolver os aplicativos pelos alunos da segunda aplicação, devido a faixa etária dos alunos ser maior, em média treze anos, com o aluno mais velho de dezesseis anos. Já na primeira aplicação os alunos tinham em média doze anos, com o aluno mais velho com 14 anos.

Em geral, foi observado que os alunos tiveram facilidade para utilizar os recursos de programação e desenvolvimento de aplicativos através do App Inventor, seja utilizando os blocos de código, ou adicionando novos componentes através do Designer.

Outro fator importante no decorrer da oficina foi a possibilidade de testar, fácil e rapidamente, as funcionalidades desenvolvidas. Devido a possibilidade de conectar os *smartphones* com o App Inventor e a cada mudança no código automaticamente era atualizado no aplicativo do *smartphone*. Contudo, foi identificado um problema em relação as conexões, pois muitas vezes era necessário reconectar um ou outro dispositivo, devido ao grande número de *smartphones* conectados ao mesmo tempo na rede WIFI com o App Inventor.

Por outro lado, foi observado e citado por alguns alunos a dificuldade em acompanhar a aula. Foi percebido, em alguns casos, que o aluno da dupla o qual não estava desenvolvendo o aplicativo no computador acabava por se distrair e conversar com outros alunos ou até mesmo jogar outros jogos no *smartphone*, ao invés de, ajudar a sua dupla prestando atenção no instrutor. E também nos casos em que um aluno estava desenvolvendo o aplicativo sozinho foi notado maior dificuldade.

Na Tabela 25 são apresentados os comentários dos alunos em relação a facilidade da unidade instrucional.

Tópico	Comentários
Facilidade das aulas	“Legal, foi um aprendizado legal.”
	“Muito boa e fácil de aprender.”
	“Muito legal, dinâmica, ótimo e funciona.”
	“De ser muito fácil.”
Pontos fortes em relação a facilidade	“O jeito de explicar dos instrutores.”
	“Educativa, legal.”
	“Achei que foi bom, me ajudou a fazer um app bem legal e me ensinou a fazer vários outros.”
	“Quando eu comecei a fazer todo o jogo e jogar como foi falado no começo.”
Pontos fracos em relação a facilidade	“Uma hora que eu erre e fiquei um pouco perdida.”
	“Me sentir burra porque não conseguia montar o aplicativo.”
	“Não gostei muito porque era um pouco complicado.”
	“Ser meio demorado.”

Tabela 25 - Respostas discursivas dos alunos no questionário pós-unidade referente à facilidade da aula.

5.3.3 A unidade instrucional promove uma experiência de aprendizagem agradável e divertida?

De acordo com as respostas obtidas nos questionários, a unidade instrucional foi avaliada pela maioria de forma positiva como muito divertidas ou divertidas e excelentes ou boas. Somente um aluno respondeu negativamente e outro grupo de alunos acabou não respondendo, conforme pode ser observado nas Figuras 25 e 26.

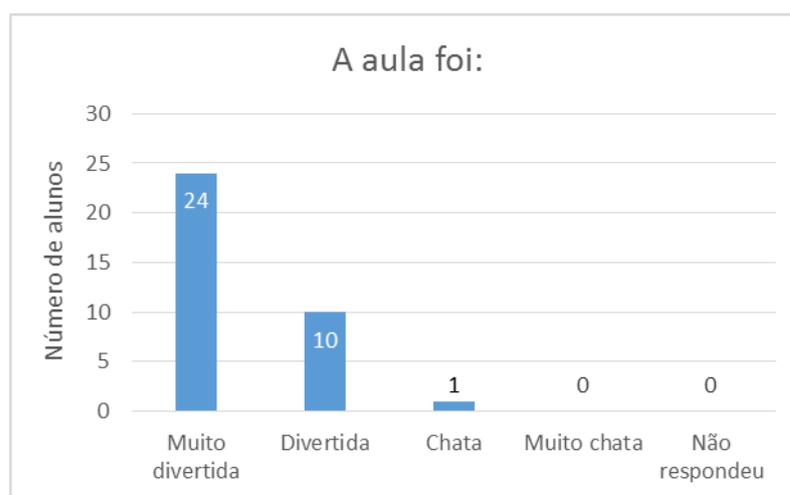


Figura 25 - A aula foi divertida?

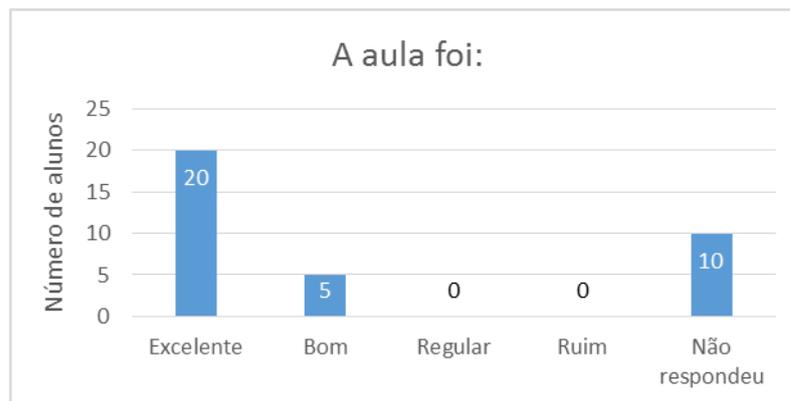


Figura 26 - A aula foi boa?

Nota-se que a percepção positiva sobre a unidade instrucional, também é identificada quando questionado aos alunos sobre o tempo das aulas. A maioria dos alunos respondeu que a aula passou muito rápido ou rápido (Figura 27). Essas respostas positivas estão diretamente ligadas aos alunos, os quais tiveram mais facilidade para acompanhar o desenvolvimento do aplicativo, seja por possuírem uma maior facilidade, ou estarem mais entusiasmados e focados nas instruções do instrutor.

Um dos motivos da percepção negativa dos alunos pode ter relação com a dificuldade em acompanhar as instruções, como foi citado no Item 5.3.2. Outro motivo pode ter sido devido a problemas com a infraestrutura de rede WIFI, a qual ficou sem acesso à rede durante um período da segunda aplicação realizada, atrapalhando os testes dos aplicativos realizados no *smartphone*.



Figura 27 - O tempo das aulas passou.

Em geral, os alunos tiveram uma boa experiência durante a unidade instrucional (Tabela 26). A possibilidade de desenvolver um aplicativo por completo durante a aula, foi considerado a melhor parte. Também se notou um entusiasmo extra devido ao tema

abordado no jogo desenvolvido, o mosquito está em alta devido as atividades de conscientização sobre o mosquito *Aedes Aegypti* realizadas na escola.

Foi notado pelos instrutores e pela professora de tecnologia que alguns alunos gostariam que a aula tivesse durado mais tempo, ou fosse realizado outras aplicações para que eles pudessem continuar o desenvolvimento do aplicativo.

Tópico	Comentários
Experiência das aulas	“Achei muito legal, gostei bastante, bem interessante.”
	“A melhor, adorei simplesmente ótima.”
	“Eu achei essa oficina muito legal e interessante. Adorei meu joguinho.”
	“Aconteceram alguns problemas, mas foi massa.”
	“Eu achei muito legal, eu vou treinar em casa.”
	“Muito boa e fácil de aprender.”
	“Muito legal, dinâmica, ótimo e funciona.”
Pontos fortes em relação a experiência	“Bem legal, façam mais.”
	“Aprender a criar aplicativos.”
	“Aprender a moldar o jogo criado conforme a minha vontade.”
	“Toda a descontração e conhecimento que os professores nos passaram.”
	“Criar aplicativos e a forma que o professor ensinou.”
	“Quando eu comecei a fazer todo o jogo e jogar como foi falado no começo.”
Pontos fracos em relação a experiência	“O jogo.”
	“De criar o mosquito.”
	“Um pouco de bagunça.”
	“Que acabou a internet.”
	“De ser muito fácil.”
	“Ser meio demorado.”
	“Designer, na verdade... Achei a parte menos interessante.”

Tabela 26 - Respostas discursivas dos alunos no questionário pós-unidade referente à experiência.

5.3.4 A unidade instrucional proporciona uma percepção positiva da computação?

Percebeu-se que durante e após a unidade instrucional realizada a maioria dos alunos demonstraram interesse em continuar aprendendo sobre o desenvolvimento de aplicativos por meio do App Inventor. Isso também foi percebido pela professora de tecnologia, a qual acompanhou as atividades, que os alunos gostaram de desenvolver um aplicativo (Figura 28).

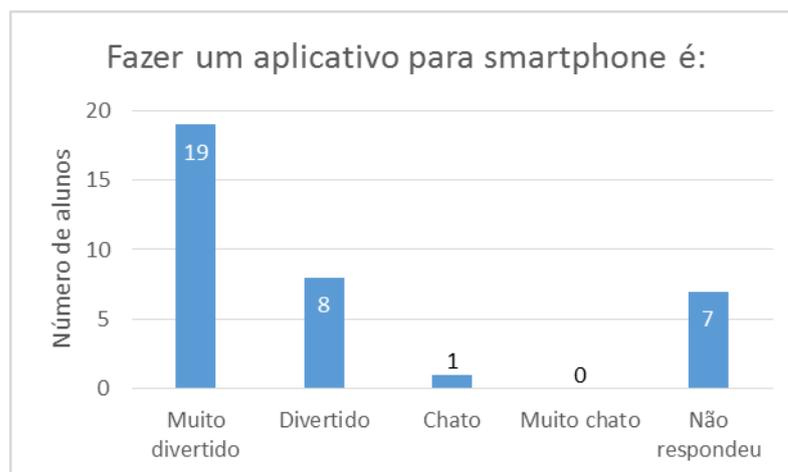


Figura 28 - Fazer um aplicativo para *smartphone* é.

Além de gostarem de desenvolver o próprio aplicativo, a grande maioria das crianças gostariam de aprender mais sobre o desenvolvimento de aplicativos (Figura 29).

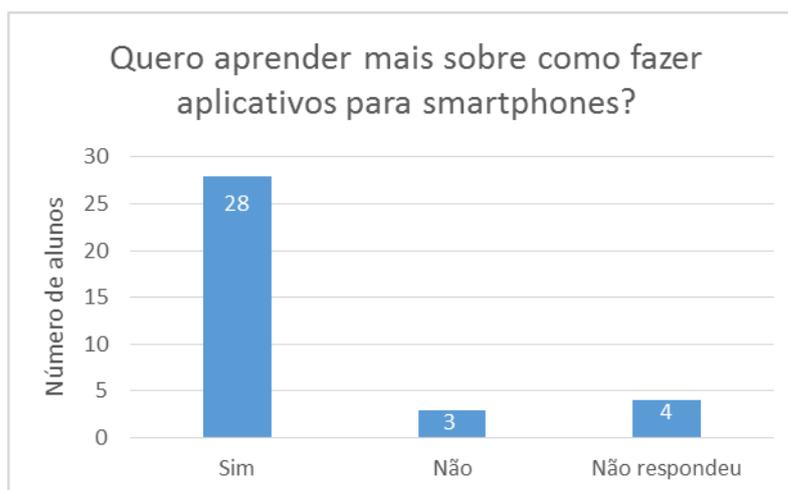


Figura 29 - Quero aprender mais sobre como fazer aplicativos para *smartphones*.

Considerando que a maioria dos alunos nunca teve contato com qualquer tipo de desenvolvimento de *software*, desenvolver um aplicativo em poucas horas era considerado algo impossível e ao mesmo tempo teve um efeito motivador para eles. Muitos alunos aprovaram a ideia de desenvolver um jogo, e ao passo que era implementado uma nova funcionalidade, ficavam ansiosos para ver o resultado na tela dos seus dispositivos. E a característica de desenvolver o jogo em etapas, manteve o interesse dos alunos para desenvolver o projeto durante todo o período da aula.

Apesar do desenvolvimento do jogo ser feito passo a passo com o instrutor, o fato de ao final da aula ter um aplicativo instalado nos seus dispositivos, deixou muitos alunos entusiasmados para mostrar o jogo aos amigos e familiares.

5.3.5 Ameaças à validade

Existem alguns fatores no *design* de pesquisa que podem ter influenciado a validade dos resultados. Para minimizar os erros de medição foi adotada a abordagem GQM (BASILI; CALDIERA; ROMBACH, 1994), o qual auxilia na decomposição dos objetivos de medição.

Em relação a coleta de dados, para diminuir possíveis ameaças à validade, os questionários foram projetados e revisados coletivamente com cuidado, tentando usar uma linguagem clara e fácil de entender que permitiria o público alvo responder corretamente.

Ao final, obteve-se uma amostra de tamanho aceitável (36 alunos), apesar de alguns terem respondido somente um dos questionários pré/pós. Porém para poder realizar uma análise referente a generalização dos resultados obtidos, será necessária a repetição do estudo em outras escolas. Entretanto, foi possível obter um primeiro *feedback* significativa referente à aplicação da unidade instrucional no contexto de uma pesquisa exploratória.

6 CONCLUSÃO

O objetivo do presente trabalho foi o desenvolvimento de uma unidade instrucional para ensinar computação por meio do desenvolvimento de aplicativos para crianças no ensino fundamental II. Nesse contexto, foi feita uma síntese sobre a fundamentação teórica de aprendizagem, ensino de computação no ensino fundamental II e desenvolvimento de aplicativos. Além disso, foi realizada uma revisão do estado da arte referente ao ensino de computação por meio do desenvolvimento de aplicativos com o App Inventor indicando a falta de unidades instrucionais para esta faixa etária de 10 a 14 anos. Como primeiro passo para o desenvolvimento da unidade instrucional, foi feito uma análise do contexto e dos aprendizes. Com base nesta análise, foram definidos os objetivos de aprendizado a serem abordados pela unidade instrucional. Seguindo o processo de design instrucional foi projetado a unidade instrucional e desenvolvidos todos os materiais didáticos necessários.

A unidade foi aplicada em escola pública em duas turmas do Ensino Fundamental ambas com 18 participantes. Foi realizada a avaliação da qualidade da unidade instrucional. A avaliação pelos alunos apresentou resultados motivadores. Por exemplo, 74% dos participantes se consideraram aptos a desenvolver um aplicativo para *smartphone*. Aliado a esse dado, está o fato de que 94% dos alunos se consideravam incapazes de desenvolver um aplicativo para *smartphone* antes da unidade instrucional. Esses dados indicam que a unidade instrucional pode atingir o seu objetivo educacional. Contribuindo, dessa forma com o ensino de computação por meio de uma experiência agradável, motivadora e satisfatória.

Como resultado do presente trabalho, é disponibilizado uma unidade instrucional para ensinar computação por meio do desenvolvimento de aplicativos utilizando o App Inventor. Essa unidade pode ser aplicada no ensino fundamental II, a partir de 10 anos com ou sem experiência em computação. O objetivo da unidade instrucional é ensinar ao aluno conceitos básicos da computação, principalmente relacionado à prática/programação e ao pensamento computacional.

Como conteúdo para trabalhos futuros, é possível o desenvolvimento de unidades instrucionais de desenvolvimento de aplicativos com o App Inventor, com maiores cargas horárias, ensinando de maneira mais aprofundada sobre os conceitos de computação e também sobre todas as possibilidades que a ferramenta proporciona. Além disso, pode-se realizar unidades instrucionais de desenvolvimento de aplicativos aliadas ao estudo de outros conteúdos já abrangidos no ensino fundamental II, de forma interdisciplinar.

ANEXO I – ARTIGO PRODUZIDO COM BASE NO TCC ‘Ensinando a Computação por meio de Programação com App Inventor’

Ensinando a Computação por meio de Programação com App Inventor

Guilherme Trilha Daniel

Instituto Nacional para Convergência Digital (INCoD)/ Departamento de Informática e Estatística (INE)/
Universidade Federal de Santa Catarina (UFSC)
trilhaguilherme@gmail.com

Resumo. Aproveitando a proximidade de alunos do ensino fundamental com smartphones, uma forma cativante de ensinar computação pode ser pela programação de aplicativos móveis. Dentro deste contexto a ferramenta App Inventor, um ambiente de desenvolvimento que utiliza programação visual em blocos, permite qualquer pessoa, até mesmo crianças, começar a programar e construir aplicativos completos para dispositivos Android. Assim, o objetivo do presente artigo é apresentar o design instrucional de uma oficina para ensinar a programação de aplicativos em celulares Android com a ferramenta App Inventor, alinhado ao currículo de referência ACM/CSTA K-12 no Ensino Fundamental II (para crianças de 11 aos 14 anos de idade). A unidade instrucional foi aplicada na prática em duas turmas do Ensino Fundamental II de uma escola pública em Florianópolis/SC, apresentando resultados motivadores em termos da aprendizagem, facilidade e experiência de aprendizagem. Além de gostarem de desenvolver o próprio aplicativo, a grande maioria dos alunos expressou que gostaria de aprender mais sobre o desenvolvimento de aplicativos. Considerando que a maioria dos alunos nunca teve contato com qualquer tipo de desenvolvimento de software, desenvolver um aplicativo em poucas horas era considerado antes algo impossível e assim promoveu um grande efeito motivador para eles.

Abstract. Taking advantage of the proximity of elementary school students with smartphones, a captivating way to teach computer science can be by programming mobile applications. Within this context the App Inventor, a development environment that uses visual programming block, allows anyone, even children, start programming and build complete applications for Android devices. The objective of this paper is to present the instructional design of a workshop to teach programming applications on Android phones with the App Inventor, aligned to the ACM / CSTA K-12 reference curriculum in Primary Education II (for 11 to 14 years old children). The instructional

unit was applied in practice in two groups of elementary school of a public school in Florianópolis/SC, with significant results in terms of learning, ease and learning experience. Besides like to develop the application itself, the vast majority of students said that would like to learn more about developing applications. Whereas most students never had contact with any type of software development, develop an application in a few hours before was considered impossible and thus promoted a great motivating effect for them.

1 Breve introdução teórica/Revisão da literatura

De formar geral os alunos do ensino fundamental possuem um bom conhecimento sobre a interação com as tecnologias existentes como *smartphones*, *tablets* e computadores. Segundo o IBGE (2013) mais do que 50% das crianças da faixa etária de 10-14 anos tem celular no Brasil. Isto acaba facilitando as diversas iniciativas existentes focadas no ensino de computação que utilizam *smartphones*. Isso pode ser feito utilizando aplicativos que oferecem exercícios de programação ou até mesmo desenvolvendo os próprios aplicativos.

Uma das alternativas é a ferramenta App Inventor (APP INVENTOR, 2016), desenvolvido no Instituto de Tecnologia de Massachusetts (MIT), a qual proporciona aos estudantes a experiência de desenvolver os mais diversos aplicativos para os seus próprios celulares. O App Inventor é um ambiente de desenvolvimento que utiliza programação visual em blocos e permite qualquer pessoa, até mesmo crianças, começar a programar e construir aplicativos completos para dispositivos Android. O App Inventor permite a programação alinhada ao processo de desenvolvimento de aplicativos móveis que envolvam tanto a programação das funcionalidades em si, quanto o design das interfaces. Ele também permite testar o aplicativo instantaneamente enquanto desenvolve pelo “*Live Testing*”. Em 2015, a comunidade do MIT App Inventor consistiu em aproximadamente 3 milhões de usuários em 195 países. Também é usado em programas como a *Technovation Challenge*, focando no envolvimento de meninas. Porém a participação no Brasil, mesmo crescendo, ainda é pequena e apesar da existência da ferramenta App Inventor, falta um suporte maior para a sua pronta aplicação em uma sala de aula. Em relação a materiais didáticos, basicamente só existem alguns tutoriais para serem feitos sozinhos e disponíveis somente na língua inglesa (TRILHA DANIEL, 2016). Assim, nota-se a falta de unidades instrucionais com materiais de apoio ao ensino, como *slides*, tarefas de casa, etc. E mesmo supondo a sua efetividade e eficiência para o ensino de computação no Ensino Básico, faltam ainda avaliações sistemáticas destas unidades instrucionais com tamanhos de amostras significantes que demonstram estes impactos de forma válida.

Assim, o objetivo do presente artigo é apresentar o design instrucional de uma oficina para ensinar a programação de aplicativos em celulares Android com a ferramenta App Inventor, alinhado ao currículo de referência (ACM/CSTA, 2011) no Ensino Fundamental II (para crianças de 11 aos 14 anos de idade).

2 Natureza da experiência relatada

Ensino de programação/conceitos básicos de computação por meio de programação de apps com App Inventor.

3 Caracterização do ambiente escolar, organização e duração das atividades realizadas

O público alvo deste trabalho são alunos do Ensino Fundamental II de escolas básicas no Brasil com idade entre 11 e 14 anos. Espera-se também por meio da oficina atingir especialmente meninas motivando-as para a área de computação e na participação no *Technovation Challenge*. Levando em consideração os recursos típicos em escolas Brasileiras é definida uma oficina de baixo custo e de curta duração com o objetivo de providenciar uma primeira introdução a computação.

De acordo com as diretrizes de currículo de computação K-12 [ACM/CSTA 2011] e em relação ao tema da oficina, os objetos gerais de aprendizagem da oficina são: Compreender conceitos básicos da computação; Compreender algoritmos e uma sequência de instruções; Programar e testar um programa com App Inventor; Colaborar com colegas por meio da programação em pares.

Dentro da oficina são abordados os seguintes conteúdos: utilização do ambiente App Inventor; programação e testes de um app de jogo com App Inventor; instalação do app desenvolvido no celular. A oficina tem a duração de aprox. 3 horas/aula. Todos os materiais didáticos são compartilhados sob a licença Creative Commons Atribuição-NãoComercial-CompartilhaIgual 4.0 Internacional disponível no site: http://www.computacaonaescola.ufsc.br/?page_id=1474.

4 Procedimentos metodológicos

O objetivo desta pesquisa é o desenvolvimento, a aplicação e a avaliação de uma unidade instrucional (oficina) para o ensino de computação no ensino fundamental. Para atingir este objetivo é realizado um estudo de caso exploratório para compreender os fenômenos observados durante as aplicações da unidade instrucional em um contexto particular e identificar direcionamentos para trabalhos futuros (Fig. 2).

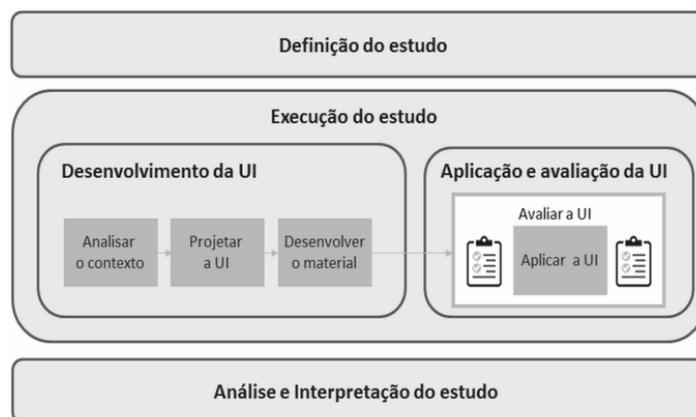


Figura 1. Metodologia de pesquisa

O estudo de caso é realizado conforme os procedimentos propostos por Yin (2013) e Wohlin et. al (2012):

Definição do estudo: O estudo é definido em termos do objetivo e perguntas de pesquisa e o design de pesquisa. A partir do objetivo e das perguntas de análise são sistematicamente derivadas as medidas para a coleta de dados utilizando o método GQM [Basili, Caldeira e Rombach 1994]. Para a operacionalização da coleta de dados são definidos instrumentos de coleta de dados.

Execução do estudo: A execução do estudo é realizada adotando o modelo ADDIE [Branch 2009] como abordagem para o design instrucional. Em uma primeira etapa a unidade instrucional é desenvolvida. Para isto, são caracterizados os aprendizes e o ambiente. São então levantadas as necessidades de aprendizagem e definidos os objetivos de aprendizagem. É projetada a unidade instrucional, definindo o seu conteúdo, a sequência e os métodos instrucionais a serem adotados. Em seguida, o material instrucional é desenvolvido. Durante a etapa da execução do estudo, a unidade instrucional é aplicada na prática em duas turmas do Ensino Fundamental II de uma escola pública em Florianópolis/SC. A oficina é avaliada, coletando-se os dados em paralelo a aplicação.

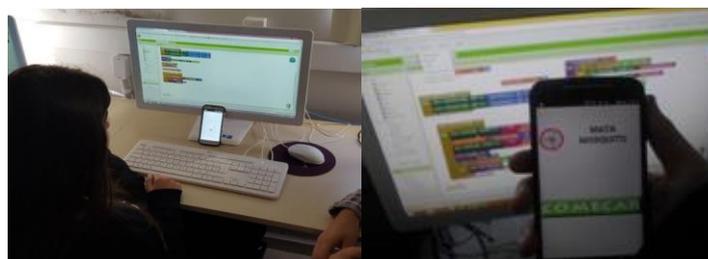




Figura 2. Cenas da aplicação da oficina

Análise e Interpretação do estudo: São analisados os dados em relação às perguntas de pesquisa, usando métodos quantitativos e qualitativos. Ao final, os resultados são interpretados e discutidos.

5 Principais conclusões

Como resultado foi projetada uma oficina de curta duração (aprox. 3 horas-aula) para ensinar computação por meio do desenvolvimento de aplicativos utilizando o App Inventor. A avaliação da aplicação da oficina em uma escola pública apresentou resultados motivadores. Por exemplo, 74% dos participantes (que antes não sabiam programar) se consideraram aptos a desenvolver um aplicativo para *smartphone* ao final da oficina. A maioria achou a oficina fácil e/ou muito fácil promovendo uma experiência de aprendizagem agradável e divertida (indicado também pelos comentários “Muito legal, dinâmica, ótimo e funciona.”, “Achei que foi bom, me ajudou a fazer um app bem legal e me ensinou a fazer vários outros.”, “Eu achei muito legal, eu vou treinar em casa”). Muitos alunos aprovaram a idéia de desenvolver um jogo. A dinâmica da oficina guiando passo-a-passo o desenvolvimento do jogo, ajudou em manter os alunos focados, que ficavam ansiosos para ver cada nova funcionalidade na tela dos seus dispositivos. A oficina contribui também em despertar o interesse na área de computação. Além de gostarem de desenvolver o próprio aplicativo, a grande maioria dos alunos expressou que gostaria de aprender mais sobre o desenvolvimento de aplicativos. Considerando que a maioria dos alunos nunca teve contato com qualquer tipo de desenvolvimento de software, desenvolver um aplicativo em poucas horas era considerado antes algo impossível e assim promoveu um grande efeito motivador para eles.

Referências

APP INVENTOR, About us. Disponível em: <<http://appinventor.mit.edu/explore/about-us.html>>.

Basili, V. R.; Caldeira, G. & Rombach, H. D. “Goal Question Metric Paradigm”. In Encyclopedia of Software Engineering, John Wiley & Sons, 1994.

Branch, R. M. “Instructional Design: The ADDIE Approach”. New York: Springer, 2009.

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>.

IBGE, Acesso à internet e à televisão e posse de telefone móvel celular para uso pessoal, 2013.

Trilha Daniel, G. Design de unidade instrucional de desenvolvimento de aplicativos para o ensino fundamental. 2016. Trabalho de Conclusão de Curso. (Graduação em Ciência da Computação) – Universidade Federal de Santa Catarina.

Wohlin, C. et al. “Experimentation in Software Engineering”. Springer Verlag, 2012.

Yin, R. K. “Case Study Research: Design and Methods”. SAGE Publications, Inc; 5. Edição, 2013.

7 REFERÊNCIAS

ACM, **Computing Curricula**, 2005. Disponível em:

<http://www.acm.org/education/curric_vols/CC2005-March06Final.pdf>. Acesso em: 05 de junho de 2016.

ANDROID DEVELOPER. Disponível em:

<<http://developer.android.com/tools/studio/index.html>>. Acesso em: 25 de setembro de 2015.

APP INVENTOR, **About us**. Disponível em:

<<http://appinventor.mit.edu/explore/about-us.html>>. Acesso em: 25 de setembro de 2015.

ARAÚJO, G. **Falta de mão de obra impacta setor de TI**. Disponível em:

<http://www.techoje.com.br/site/techoje/categoria/detalhe_artigo/1527>. Acesso em: 04 de novembro 2014.

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: 30 de novembro de 2015.

BBC, **History: Enigma**. Disponível em:

<<http://www.bbc.co.uk/history/topics/enigma>>. Acesso em: 26 de novembro de 2015.

BLOOM B. **Taxonomy of Educational Objectives: The Classification of**

Educational Goals; pp. 201-207; B. S. Bloom (Ed.) David McKay Company, Inc. 1956.

BOULOS, M. N. K. et al.; **How smartphones are changing the face of mobile and participatory healthcare: an overview, with example from eCAALYX**. BioMedical Engineering OnLine, Vol. 10, N. 1, Pp. 24, 2011.

BRANCH, R. M. **Instructional Design: The ADDIE Approach**. Springer Science & Business Media, 2009.

CAPES, **Quem Participa**. Disponível em:

<http://www.periodicos.capes.gov.br/index.php?option=com_pcontent&view=pcontent&alias=quem-participa&Itemid=101>. Acesso em: 10 de setembro de 2015.

CARNEGIE MELLON. **Learning Objectives**. Disponível em:

<<http://www.cmu.edu/teaching/designteach/design/learningobjectives.html>>. Acesso em: 01 de setembro de 2015.

CHATZINIKOLAKIS, G.; PAPADAKIS, S.. **Motivating K-12 students learning fundamental Computer Science concepts with App Inventor**. In: Interactive Mobile Communication Technologies and Learning (IMCL), 2014 International Conference on. IEEE, 2014. p. 152-159.

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>.

DICK, W; CAREY, L., JAMES, O.; **The Systematic Design of Instruction**, 1996.

DRISCOLL, M. P. **Psychology of learning and instruction**. 2 ed. Boston: Allyn & Bacon, 1995.

FILATRO, A. **Design instrucional contextualizado: educação e tecnologia**. São Paulo: SENAC, 2004.

FINIZOLA, A. B. et al.; **O ensino de programação para dispositivos móveis utilizando o MIT-App Inventor com alunos do ensino médio**. In: n: Anais do Workshop de Informática na Escola. 2014. p. 337.

FLING, B.; **Mobile Design and Development**. O'Reilly Media, 2009.

FLÔRES, M. L. P.; TAROUÇO, L. M. R.; REATEGUI, E. B. **Orientações para o sequenciamento das instruções em um objeto de aprendizagem**. RENOTE: Novas Tecnologias na Educação, v. 7, n. 1, Jul. 2009.

FOLHA DE S. PAULO, **Smartphones já são 77,5% de celulares do país**. Disponível em: <http://www1.folha.uol.com.br/mercado/2015/04/1613598-smartphones-ja-sao-775-dos-celulares-do-pais.shtml>>. Acesso em: 29 de novembro de 2015.

GARTNER, **Smartphone Sales Surpassed One Billion Units in 2014**. Disponível em: <<http://www.gartner.com/newsroom/id/2996817>>. Acesso em: 17 de setembro de 2015.

GOMES, T. C. S.; MELO, J. C. B.; **App Inventor for Android: Uma proposta construcionista para experiências significativas de aprendizagem no ensino de programação**, In: Anais 4º Simpósio Hipertexto e Tecnologias, Recife/Brasil, 2012.

GOOGLE, **Searching for Computer Science: Access and Barriers in U.S. K-12 Education**. Disponível em: <http://services.google.com/fh/files/misc/searching-for-computer-science_report.pdf>. Acesso em: 26 de novembro de 2015.

GOVERNO DE SANTA CATARINA, **Proposta Curricular de Santa Catarina: formação integral na educação básica**. Santa Catarina, 2014.

GRESSE VON WANGENHEIM, C. & SHULL, F.; **To game or not to game?**. IEEE Software, vol. 26, no. 2, pp. 92–94, 2009.

GROVER, S.; PEA, R.; **Using a Discourse-Intensive Pedagogy and Android's App Inventor for Introducing Computational Concepts to Middle School Students**. In: Proceeding of the 44th ACM technical symposium on Computer science education, Denver/Estados Unidos da América. Pp. 723-728, 2013.

HOUR OF CODE. Disponível em: <<http://hourofcode.com>>. Acesso em: 01 de novembro 2014.

IBGE, **Acesso à internet e à televisão e posse de telefone móvel celular para uso pessoal**, 2013.

INEP, **Censo Escolar**, 2014.

KITCHENHAM, B. **Procedures for Performing Systematic Reviews**. Uk: NICTA Technical Report 0400011T.1, Keele University, KEELE, GB, 2004.

LIBÂNEO, J. C. **Didática**. 29 ed. Cortez, 2009.264 p

LOPES, R. D. et al; **O uso do computador e da internet na escola pública**. Disponível em: <<http://www.fvc.org.br/estudos-e-pesquisas/avulsas/estudos1-7-uso-computadores.shtml?page=0>>. Acesso em: 30 de novembro de 2015.

MACDONALD, B. **To IDE or Not to IDE?: Choosing the right tool for the beginning programmer**, 2014. Disponível em: <<http://radar.oreilly.com/2014/01/to-ide-or-not-to-ide.html>>. Acesso em: 09 de junho de 2016.

MACWORLD, **Apple: 10 Millions apps, 1 Million iPhone 3Gs**. Disponível em: <<http://www.macworld.com/article/1134484/appsiphones.html>>. Acesso em: 25 de novembro de 2015.

MAZZIONI, S. **As Estratégias Utilizadas no Processo de Ensino-Aprendizagem: Concepções de Alunos e Professores de Ciências Contábeis**. In: Anais do Congresso USP Controladoria e Contabilidade, São Paulo/Brasil, 2009.

MISA, T. J. **Understanding 'How Computing Has Changed the World'**. *IEEE Ann. Hist. Comput.* Vol. 29, no. 4, PP. 52-63, 2007.

MORIMOTO, C. E.; **Smartphones, Guia Prático**. Sulina, 2009.

NITZKE, J. A. **Estratégias de Ensino**. Disponível em: <<http://penta.ufrgs.br/~julio/tutores/estrateg.htm>>. Acesso em: 01 de setembro de 2015.

O ESTADO DE S. PAULO. **Smartphone: o verdadeiro computador pessoal**, 2015. Disponível em: <<http://economia.estadao.com.br/noticias/geral,smartphone-o-verdadeiro-computador-pessoal-imp-,1643233>>. Acesso em: 01 de dezembro de 2015.

OLHAR DIGITAL, Mercado de TI no Brasil tem mais vagas que profissionais. Disponível em: <<http://olhardigital.uol.com.br/pro/noticia/mercado-de-ti-no-brasil-tem-mais-vagas-que-profissionais/53548>>. Acesso em: 27/05/2016.

DAUER, V.; PANGRAZI, R. P. **Dynamic physical education for elementary school children**, 6 ed, Burgess Publishing Co., 1979.

PAQUETTE, G. **Modeling and delivering distributed learning environments**. TelE-Learning, 2002.

PCN, **Parâmetros Curriculares Nacionais, Terceiro e Quarto ciclos do Ensino Fundamental**. Disponível em:

<<http://portal.mec.gov.br/seb/arquivos/pdf/introducao.pdf>>. Acesso em: 20 de setembro de 2015.

PIAZZA, A. **Melhoria de uma Unidade Instrucional para Planejamento de Custos de Projetos de Software**. Trabalho de Conclusão de Curso. (Graduação em Sistemas de Informação) – Universidade Federal de Santa Catarina, Florianópolis, Brasil, 2012.

PISKURICH, G.M. **Rapid Instructional Design: Learning ID fast and right**. 3 ed, John Wiley & Sons, 2015.

MINISTÉRIO DA EDUCAÇÃO. Secretaria de Educação Básica, **Ensino fundamental de nove anos: Orientações para a inclusão da criança de seis anos de idade**.

Brasília, 2007. Disponível em: <

<http://portal.mec.gov.br/seb/arquivos/pdf/Ensifund/ensifund9anobasefinal.pdf>>. Acesso em: 9 de junho de 2016.

PROINFO. Disponível em:

<<http://portal.mec.gov.br/component/content/article?id=236:proinfo>>. Acesso em: 30 de novembro de 2015.

RAMOS, N. et al.; **Ensino de Programação para Alunas de Ensino Médio: Relato de uma Experiência**. In: **anais do 23º WEI - Workshop sobre Educação em Computação**, Recife, Brasil, 2015.

SASKATCHEWAN EDUCATION. **Instructional Approaches: A Framework for Professional Practice**. Canada: Saskatchewan Education, 1991.

SAVAGE, S.; **The Generation Z Connection: Teaching Information Literacy to the Newest Net Generation**, 2006. Disponível em:

<http://www.redorbit.com/news/technology/397034/the_generation_z_connection_teaching_information_literacy_to_the_newest/>. Acesso em: 25 de novembro de 2015.

SBC, **Plano de Gestão para a SBC Biênio, Agosto 2013**. Disponível em:
<[http://www.sbc.org.br/index.php?option=com_content&view=category&layout=blog
&id=215&Itemid=82](http://www.sbc.org.br/index.php?option=com_content&view=category&layout=blog&id=215&Itemid=82)>. Acesso em: 14 de setembro de 2015.

SMITH, P. L.; RAGAN, T. J. **Instructional design**. John Wiley & Sons, Inc., 1999.

STATISTA, **Number of available applications in the Google Play Store from December 2009 to July 2015**. Disponível em:
<<http://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/>>. Acesso em: 18 de setembro de 2015.

TIC EDUCAÇÃO, **Pesquisa sobre o uso das tecnologias de informação e comunicação nas escolas brasileiras**, 2014. Disponível em:
<http://cetic.br/media/docs/publicacoes/2/TIC_Educacao_2014_livro_eletronico.pdf>. Acesso em: 08 de junho de 2016.

WATTERS, A. **5 Tools to Introduce Programing to Kids**, 2011. Disponível em:
<<http://ww2.kqed.org/mindshift/2011/05/16/5-tools-to-introduce-programming-to-kids/>>. Acesso em: 19 de outubro de 2015.

WING, J. M. **Computational thinking**. Communications of the ACM, Vol. 49, no. 3, PP. 33–35, 2006.

WOLBER, D. et al. **App inventor: create your own android apps**. Sebastopol, CA: O'Reilly Media, 2011.

WOLBER, D.; ABELSON, H.; FRIEDMAN, M.; **Democratizing Computing with App Inventor**. GetMobile: Mobile Comp. and Comm, vol. 18, no. 4, PP. 53-58, 2014.