

UNIVERSIDADE FEDERAL DO MARANHÃO  
LICENCIATURA EM INFORMÁTICA

WENDEL QUEIROZ DE LIMA

**Um modelo de computador de baixo custo construído com placas open source para o estudo e aprendizagem de programação**

CODÓ-MA  
JAN./2018

WENDEL QUEIROZ DE LIMA

**Um modelo de computador de baixo custo construído com placas open source para o estudo e aprendizagem de programação**

Monografia apresentada ao Curso de Licenciatura em Informática da Universidade Federal do Maranhão – UFMA, como requisito parcial para obtenção do grau de licenciado em Informática.  
Orientador: Prof. Dr. Alex de Sousa Lima

CODÓ-MA  
JAN./2018

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).  
Núcleo Integrado de Bibliotecas/UFMA

Queiroz de Lima, Wendel.

Um modelo de computador de baixo custo construído com placas open source para o estudo e aprendizagem de programação / Wendel Queiroz de Lima. - 2018.

59 p.

Orientador(a): Prof. Dr. Alex Sousa Lima.

Monografia (Graduação) - Curso de Licenciatura em Informática, Universidade Federal do Maranhão, Codó-Ma, 2018.

1. Computador open source de baixo custo. 2. Editor off-line Scratch. 3. Lógica de programação. I. Lima, Prof. Dr. Alex de Sousa. II. Título.

WENDEL QUEIROZ DE LIMA

**Um modelo de computador de baixo custo construído com placas open source para o estudo e aprendizagem de programação**

Monografia apresentada ao Curso de Licenciatura em Informática da Universidade Federal do Maranhão – UFMA, como requisito parcial para obtenção do grau de licenciado em Informática.  
Orientador: Prof. Dr. Alex de Sousa Lima

Aprovado em 30 de janeiro de 2018

BANCA EXAMINADORA

---

Professor Dr. Alex de Sousa Lima  
*Orientador*

---

Professor Me. Lanyllo Araújo dos Santos

---

Professor Me. Rondinelle Luís Silva de Sousa

CODÓ-MA  
JAN./2018

## AGRADECIMENTOS

Em primeiro lugar à DEUS, meu criador, fonte de minha vida.

À minha família

Especialmente à minha mãe Elice, que fizeste tudo por mim desde a minha existência aos primeiros passos da educação que tenho, o suor derramado por mim no pão de cada dia à formação do meu caráter, os incentivos, motivações e orações, juntamente com a minha avó Maria José (In memória), que fez parte da minha vida e me criou nos primeiros anos.

À minha irmã Weslany, pela força, gratidão, incentivos e orações dedicadas a mim. À minha esposa Viviane pelo carinho, amor, paciência e coragem que me passa, juntamente com minha filha amada Maria Isabelly a quem eu tanto amo e me fortalece com sua doçura de criança quando me abraça e me passa um sentimento de amor.

À minha tia Maria José, a quem tenho carinho e força por tanto me ajudar nas horas em que mais preciso.

À minha madrinha Luzia Angélica que tanto se preocupa comigo e que me ajudou nos momentos difíceis de minha vida juntamente com meu padrinho Raimundo Nonato (In Memória), que fez parte da minha vida como exemplo de pessoa que era, por ser um homem bom e dedicado à família dando muitos exemplos de vida.

Gostaria de agradecer ao professor e meu orientador Prof. Dr. Alex de Sousa Lima, pela força, dedicação e apoio, me ajudando mais uma vez nos momentos em que me encontro desamparado, por me permitir concluir esse trabalho pela sua tão importante contribuição à minha formação e mais uma vez presente no caminho de quem precisa de ajuda.

Ao Pe Orlando, pela orientação espiritual e conselhos cristãos, apoio moral e incentivação nas escolhas da vida.

Aos amigos Mestres

Francisco Silva, que é um irmão para mim, por ter me ajudado nesse curso de Licenciatura em Informática principalmente nos algoritmos e demais disciplinas de programação e Lanyllo Araújo, que me ajudou a formular a idéia desse trabalho quando me falou das placas de baixo custo.

Aos amigos

Idenilton Freitas, Raniel Lucena, Fernando Figueiredo, Rogério Silva, Reinaldo Silva, Eugênio Silva, Weliton Neneco, Carlino Filho e Paulo Roberto.

E aos amigos da UFMA

Raimundo Moreira, Gilvan, Marcio, Fernando Miranda, Mourão Filho e seu Edvaldo.

## **RESUMO**

Com os recursos das placas de baixo custo open source, é possível montar um Pequeno computador microprocessado com um simples cartão de memória, que comporta-se como Disco Rígido, em uma pequena plaquinha que acopla todos os requisitos de funcionamento de um computador convencional e alternativa de Sistemas Operacional, contendo escritório, navegadores de internet e ferramentas de desenvolvimento para linguagens de programação a serem estudadas por qualquer pessoa que se interessa pela tecnologia de forma gratuita por serem de código aberto. Disponibiliza de ferramentas de educação como editores offline como o Scratch, que vem junto ao SO utilização em ensino e aprendizagem de lógica de programação, e estimular o raciocínio lógico pela programação dos objetos, onde a criança estuda e aprende a programar por um computador de baixo custo e acessível.

Palavras Chaves: Computador open source de baixo custo, editor off-line Scratch, lógica de programação.

## **ABSTRACT**

With the features of the low-cost open source boards, it is possible to mount a small microprocessor computer with a simple memory card, which behaves as a Hard Disk, in a small card that matches all the requirements of a conventional and alternative computer of operating systems, containing office, internet browsers and development tools for programming languages to be studied by anyone who is interested in technology for free because they are open source. It provides educational tools such as offline editors such as Scratch, which comes together with SO use in teaching and learning programming logic, and stimulate logical reasoning by object programming, where the child studies and learns to program by a computer down affordable.

Keywords: Open source low-cost computer, Scratch off-line editor, programming logic.



## LISTAS DE FIGURAS

Figura 01 .....	13
Figura 02 .....	16
Figura 03 .....	16
Figura 04 .....	17
Figura 05 .....	18
Figura 06 .....	18
Figura 07 .....	22
Figura 08 .....	23
Figura 09.....	25
Figura 10.....	25
Figura 11.....	27
Figura 12.....	28
Figura 13.....	28
Figura 14.....	29
Figura 15.....	30
Figura 16.....	31
Figura 17.....	33
Figura 18.....	38
Figura 19.....	50
Figura 20.....	50
Figura 21.....	52

## LISTA DE QUADROS

Quadro 01 .....	48
-----------------	----

## SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>11</b>
<b>2. METODOLOGIA.....</b>	<b>13</b>
2.1. O computador de baixo custo .....	13
2.2. Estudo e aprendizagem de lógica de programação .....	15
<b>3. DISCUSSÃO TEÓRICA.....</b>	<b>19</b>
3.1. Tecnologias Sustentadas e Disruptivas .....	19
3.2. Internet das Coisas .....	20
3.3. Placas de baixo custo voltadas para a programação aberta ( <i>open source</i> ) .....	28
3.4. Placas de opção de sistema operacional voltadas para a programação aberta ( <i>open source</i> ) .....	33
3.5. Sistemas operacionais voltados para a programação aberta ( <i>open source</i> ) .....	37
3.6. Lógica de programação.....	40
<b>4. TRABALHOS RELACIONADOS .....</b>	<b>46</b>
4.1. Ensino de Programação no Ensino Fundamental .....	46
4.2. Avaliando a Contribuição do Scratch para a Aprendizagem pela Solução de Problemas e o Desenvolvimento do Pensamento Criativo .....	47
4.3. Uma proposta para o Ensino de Programação de computadores na educação básica.....	51
4.4. Considerações sobre os trabalhos relacionados .....	52
<b>5. CONSIDERAÇÕES FINAIS.....</b>	<b>54</b>
<b>REFERÊNCIAS</b>	

## 1. INTRODUÇÃO

Os avanços mais recentes da área tecnológica vêm ajudando a transformar as relações nos mais variados sentidos: econômicos, culturais, comportamentais, entre outros. Dentro desse contexto tem surgido com mais força a Internet das Coisas, uma forma de tecnologia que promete estar presente em todas as sociedades, através dos lares, empresas, hospitais, escolas, meios de transportes, principalmente na comunicação das pessoas com pessoas, pessoas com os objetos, e os objetos com os objetos por sua vez, se comunicando de forma autônoma e precisa pelos diferentes tipos de redes que circulam em milésimos de segundos por todo o mundo.

Como uma forma de aproveitar essa tecnologia, tem-se percebido o uso crescente no ambiente escolar, sobretudo em países como o Reino Unido. Nesse sentido, as placas microprocessadas de baixo custo open source veio para ser utilizadas na Internet das Coisas, e fornecer uma opção também no processo educacional de todas as escolas do mundo (onde for possível), no desenvolvimento do raciocínio lógico pela programação dos objetos, pela qual a criança estuda e aprende a programar por meio de um computador como ferramenta de processo que as desafia a raciocinar e produzir conhecimento.

Nesse sentido, o objetivo geral deste trabalho foi o de propor uma forma de estudo e ensino aprendizagem em lógica de programação, capazes de orientar projetos de educação, usando os recursos das placas open source, como um computador de baixo custo, sendo a sua ferramenta de estudo. Para tanto, alguns objetivos específicos foram atingidos, como: a apresentação do conceito e a evolução da *Disrupção tecnológica* para a chegada da Internet das Coisas; apresentação e análise de hardwares (diferentes tipos de placas open source), sensores e acessórios que integram a essas placas; exposição e análise dos Sistemas Operacionais open source de fácil manuseio - controle de hardware e softwares de lógica para programação baseada em blocos; análise de artigos referentes à lógica de programação; e apresentação de uma proposta de um computador de baixo custo.

A abordagem deste trabalho está organizada em quatro seções além da introdução, a qual contemplou a motivação, a descrição e os objetivos. A segunda seção aborda a METODOLOGIA, explorando as etapas de construção do trabalho; a terceira seção destaca a DISCUSSÃO TEÓRICA, sendo apresentada em seis subtópicos: 3.1 Tecnologias Sustentadas e Disruptivas ; 3.2 Internet das Coisas; 3.3 Placas de baixo custo voltadas para a programação aberta (*open source*); 3.4 Placas de opção de sistema operacional voltados para a programação aberta (*open source*); 3.5 Sistemas operacionais voltados para a programação aberta (*open*

*source*); e, 3.6 Lógica de programação. A quarta seção destaca as discussões a partir de TRABALHOS RELACIONADOS, onde apresenta pesquisas já realizadas na literatura relacionados com o uso do computador como ferramenta educacional para o ensino e aprendizagem em lógica de programação. Por fim, a quinta seção apresenta as CONSIDERAÇÕES FINAIS, na qual destacam-se as impressões sobre o desfecho deste trabalho.

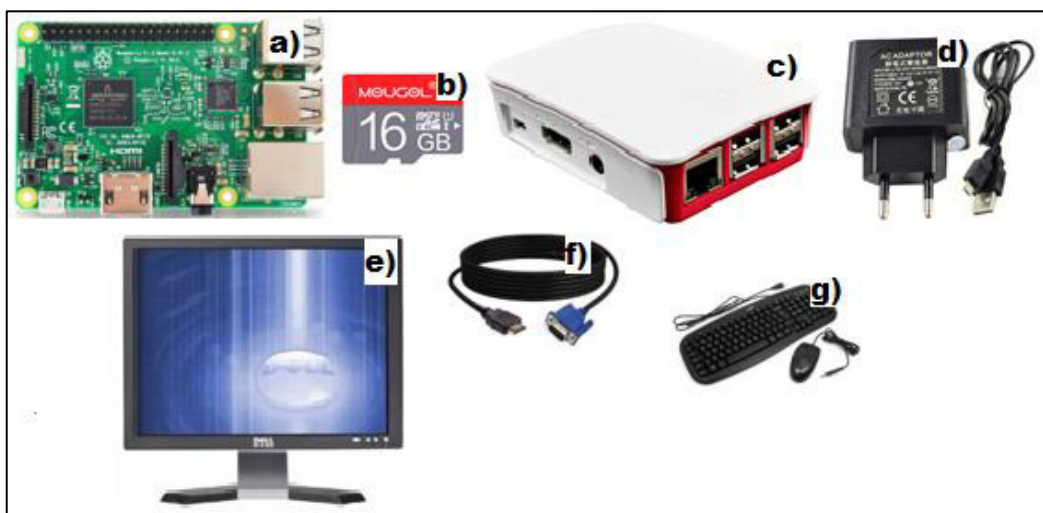
## 2. METODOLOGIA

Este trabalho tem sua essência a natureza bibliográfica, sobretudo com os temas voltados para placas de baixo custo open source como ferramenta computacional para fins de estudo e aprendizagem de lógica de programação. As placas open source é fundamentada por uma tecnologia barata que pode ser implantada para diversos fins da sociedade, inclusive auxiliando na educação, para isso, foram abordados alguns modelos dos mais renomados tipos de placas *open source* de baixo custo e Sistemas Operacional open source. Para fins de aprendizagem de lógica de programação apresentou-se a plataforma Scratch, que por sua vez poderá ser estudado por meio de um computador de placa única, montado com a placa Raspberry Pi 3 modelo B e acessórios, Sistema Operacional Ubuntu Mate, que vem trazendo junto o editor off-line Scratch de blocos de encaixe para o estudo e aprendizagem de lógica de programação.

Nesse sentido foram apresentados os processos de escolha dos mecanismos que fazem “jus” ao “baixo custo” como recurso para a montagem da ferramenta de estudo. Concorde-se que o computador não é mais o instrumento que ensina o aprendiz, mas a ferramenta com a qual o aluno desenvolve algo em torno da lógica, e, portanto, o aprendizado ocorre pelo fato de estar executando uma tarefa por intermédio do computador (VALENTE, 1998, p. 12).

### 2.1. O computador de baixo custo

#### i) Os componentes (hardwares), Figura 01.



**Figura 01:** componentes de um computador de baixo custo. a) Framboesa (Raspberry Pi 3 modelo B); b) TF Card (cartão de memória) 16GB no mínimo; c) Case (capa que esconde); d) Fonte de energia, 5V 2A – 2,5ª; e) Monitor; f) Cabo de Vídeo (HDMI - VGA); e, g) Teclado e mouse USB

Fonte: autor

Os valores dos componentes são vendidos separadamente, mas são disponibilizados como KIT's também nos itens a), b), c) e d), são encontrados no *alixpress.com* no valor de R\$159,30. O item e), pode ser adquirido no Brasil, no *mercadolivre.com.br* por cerca de R\$130,00 com frete gratuito. O item f), é encontrado também no *alixpress.com* no valor de R\$13,79. E o item g), pode ser comprado em qualquer loja física por um valor médio de R\$30,00.

**É um computador de placa única por um preço médio de R\$ 333,09**, pois se comparado com os valores dos computadores convencionais custam na média acima dos R\$1000,00 com todos os seus acessórios. Então é uma boa opção para quem possui uma baixa renda e também uma opção de tecnologia barata para projetos educacionais nas escolas do país.

A *Raspberry Pi 3 mdelo B*, foi a escolha da placa (hardware) de baixo custo, por ser a placa pioneira de um projeto que serviu de modelo para a educação mundial com uma configuração essencial para o propósito desta pesquisa. E o Sistema operacional escolhido, foi o *Ubuntu Mate* por ser uma das distribuições *Linux* mais usada no mundo e com performance de constante atualização de melhoramento, além de ser uma distribuição que dispõe de todos os recursos utilizados nos Sistemas Operacionais tradicionais como ferramentas de escritório estável. Também possui opções de navegador de internet e criação de arquivos e diretórios, além do editor off-line para a aprendizagem de lógica de programação que usaremos como proposta deste trabalho.

#### ii) Instalação do sistema operacional aberto, Ubuntu Mate na framboesa pi

Para a instalação do Raspberry Pi, foi necessário o uso de um computador convencional (sistema *Windows* ou *Linux*). Deve-se fazer o acesso ao site do *Ubuntu Mate* e efetuar o *download* de aproximadamente 1.2 GB (imagem.ISO.img) compactada do sistema em: <<https://ubuntu-mate.org/download/>>, e escolher a arquitetura *Raspberry Pi*.

- No Computador com Windows:

Depois da imagem.ISO.img compactada já baixada no computador, o Sistema Operacional Ubuntu Mate pode ser instalado com o software **Win32DiskImager** que pode ser encontrado em muitos sites de forma gratuita, para a gravação do Ubuntu Mate no SD Card, depois do SD Card já formatado em FAT32;

- No computador com Linux (Ubuntu):

Depois da imagem.ISO.img compactada já baixada no computador, o Sistema Operacional Ubuntu Mate pode ser instalado no SD Card, através da ferramenta do Ubuntu chamada Gerenciamento de Discos, na opção **restaurar imagem de discos**. E feito esse processo, em qualquer uma das opções, e só esperar terminar os 100% da instalação e colocar o SD Card no Raspberry Pi e terminar os últimos procedimentos de instalação, colocando a região, fuso horário, nome de usuário, idioma e senha se preferir, e em fim o Ubuntu Mate no Raspberry Pi.

## 2.2. Estudo e aprendizagem de lógica de programação

Para nosso estudo e aprendizagem de lógica de programação, faremos uso da plataforma **SCRATCH** descrito no tópico, que desde o ano de seu lançamento em 2007, até os dias de hoje continua sendo uma referência para projetos educacionais e desenvolvimento de lógica de programação para crianças, jovens e adolescentes do mundo todo que programaram e compartilharam mais de 15 milhões de projetos Scratch. Hoje o Scratch se encontra na versão 2.0.

A grande vantagem do **Scratch** além de ser um modelo e referência para muitos projetos hoje criados em pró da educação como por exemplo a **Hora do Código**, é porque pode ser utilizado em navegadores web de qualquer sistema operacional com o editor online e utilizado também pelo editor off-line no (Windows, MAC OS OX e distribuições Linux).

O **Scratch**, por ser uma linguagem de programação visual, crianças e jovens desenvolvem o conceito de raciocínio da lógica de programação simplesmente encaixando os blocos lógicos coloridos para formulação de ideias com mídias a fim de que se pense de forma criativa e colaborativa, desde a construção de histórias programadas por clics do mouse ou por um apertar de uma tecla de uma letra ou número programado pelo estudante.

Seu ambiente de programação se torna de fácil compreensão pelas seguintes características de menus: **Scripts, Fantasia ou Plano de Fundo e Sons**.

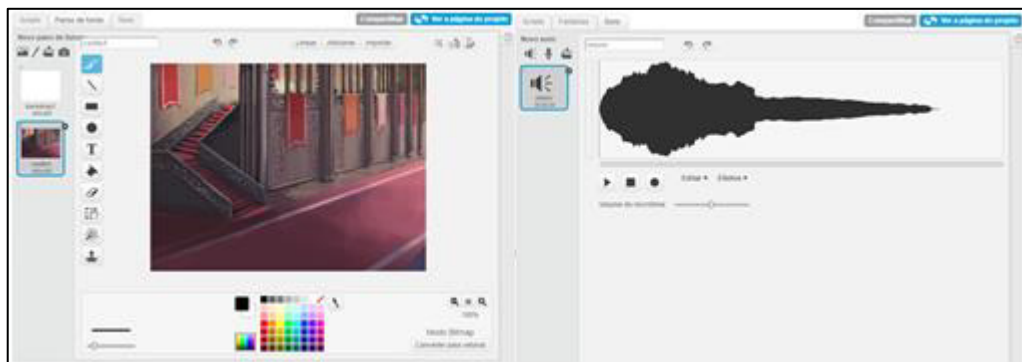
**Scripts** – são as 10 ferramentas lógicas em blocos coloridas de encaixe, identificadas por cores que manipula **movimentos, aparência, som, caneta, variáveis, eventos, controle, sensores, operadores e mais blocos**, em que se trabalha os comandos lógicos. Por exemplo uma imagem ator no palco dizer *meow, ou outro som*, depois de ter andado 10 passos e ter virado um pouco para a direita, seria os seguintes procedimentos: seleciona a ferramenta de blocos **eventos de cor marrom**, para iniciar, depois em blocos **movimentos de cor azul** para andar 10 passos e no mesmo bloco girar 15 graus para a direita e dizer **meow ou outro som**

gravado no bloco **violeta** para que o ator ou outra imagem no palco faça essa sequência lógica (Figura 02). Assim então ocorre uma associação de programação imediata, simples e de fácil aprendizagem a ser desenvolvidos entre professores e alunos.



**Figura 02:** Ferramentas de Comandos identificadas por cores  
Fonte: organizado pelo autor com o Scratch 2.0 (2018)

No menu **Fantasia/Plano de Fundo**, é caracterizado por fazer a manipulação de imagens ambientes de plano de fundo de lugares e ator personagem no palco para a criação e manipulação através da associação das ferramentas de blocos coloridos no menu Scripts e a manipulação de inserção de algum tipo de som como “fala”, músicas, sons de algum animal ou qualquer outro som gravado pelo aluno ou professor para a atuação dos atores personagens através do menu Sons (Figura 03).



**Figura 03:** Manipulação de Mídias  
Fonte: Scratch 2.0 (2018)

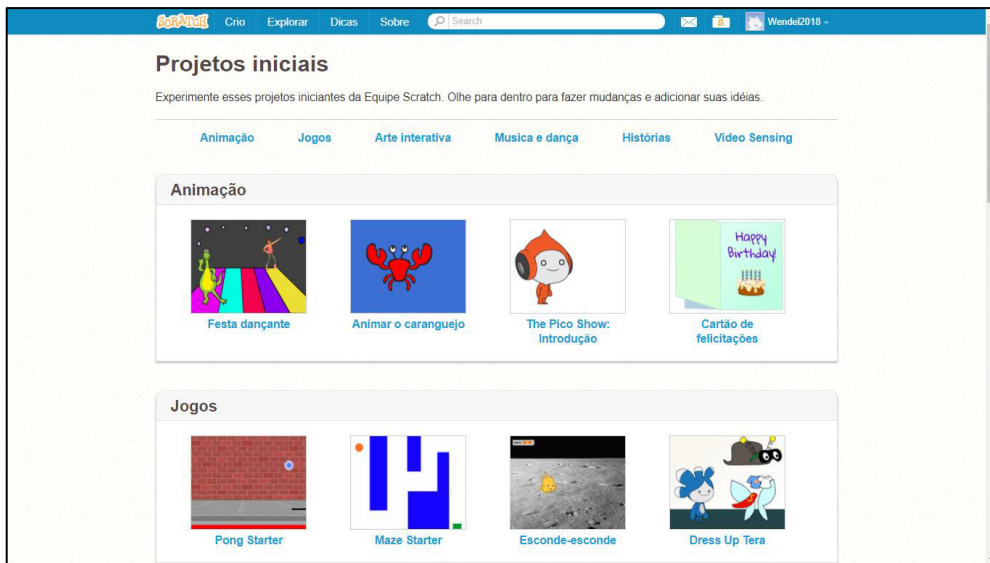
O maior problema que hoje encontramos em diversos projetos de tecnologia na educação é justamente onde encontrar a capacitação adequada dos professores para domínio de tal tecnologia discutida para projetos pedagógicos em sala de aula.

A escolha dessa proposta é diretamente nesse foco quanto as necessidades tanto do aluno quanto a do professor com a pergunta: onde encontrar o suporte com curso de capacitação?

A proposta deste trabalho é usar a plataforma **Scratch** como suporte na busca do aprendizado para professores e alunos em busca desse conhecimento, pois dispõe de uma



verdadeira biblioteca com mais de 15 milhões de projetos compartilhados pelo mundo todo sendo a mais completa para dar suporte e uso do **Scratch** como um **apoio de acesso didático** que são disponibilizados para download diretamente de sua plataforma online no site: < [https://scratch.mit.edu/starter\\_projects/](https://scratch.mit.edu/starter_projects/)> (Figura 04) feitas por pessoas que criam e compartilham suas histórias, jogos e animações que podem ser baixadas e modificados por qualquer pessoa.

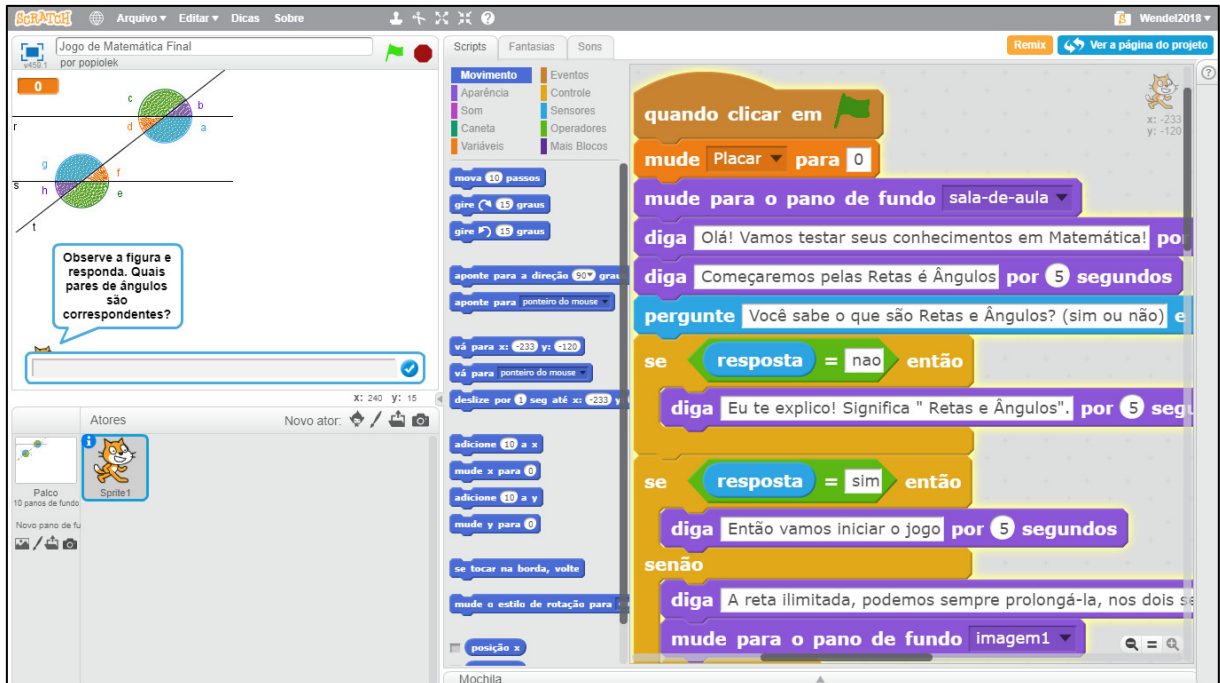


**Figura 04:** Projetos do Scratch

Fonte: [scratch.mit.edu](https://scratch.mit.edu)

Professores podem usar esse recurso para trabalhar o desenvolvimento da lógica de programação, pelo material de acesso de milhões de atividades feitas e disponibilizados no site do MIT Scratch, o download das atividades disponibilizadas e aborda com os alunos em sala de aula.

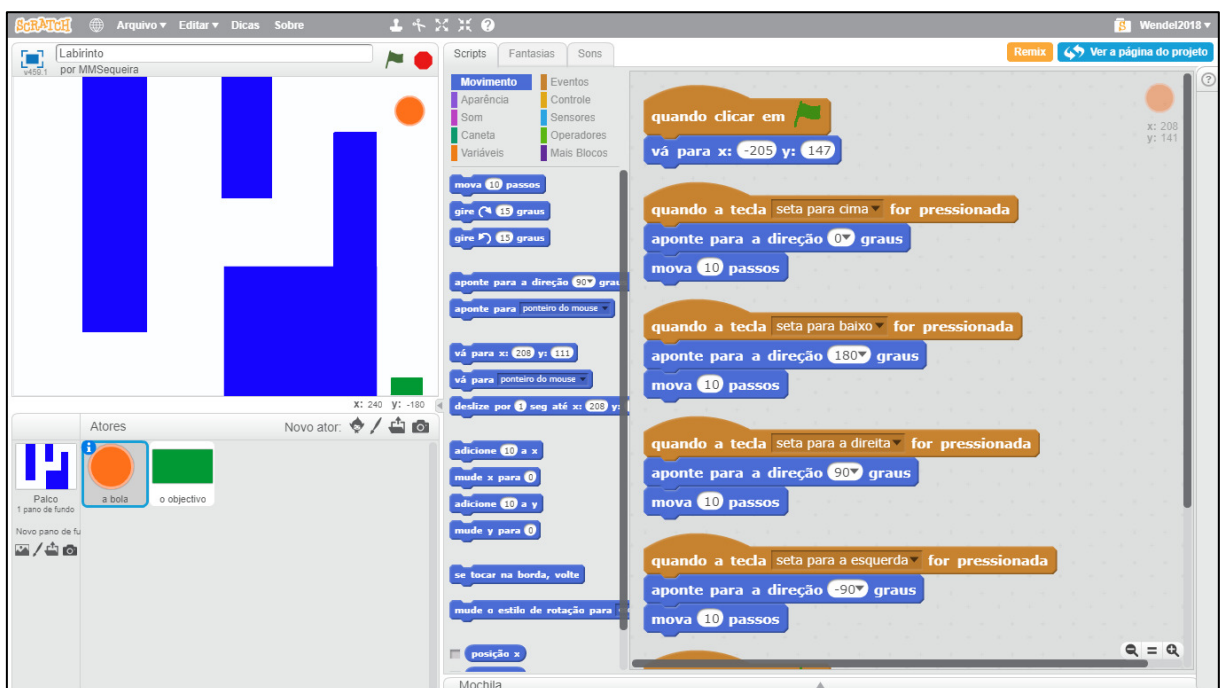
No exemplo 1, um jogo de matemática, explorando **retas e ângulos**, compartilhado no site do **Scratch** em 25/06/2015 (Figura 05).



**Figura 05:** Jogo de matemática final

Fonte: scratch.mit.edu

No exemplo 2, o projeto de um labirinto lógico com poucas sequências de blocos, onde a bola laranja controlado pelas setas direcionais do teclado, terá que percorrer entre os espaços do labirinto azul sem que toque nas paredes até chegar no seu objetivo, no bloco verde (Figura 06).



**Figura 06:** Labirinto Lógico

Fonte: scratch.mit.edu/

### 3 DISCUSSÃO TEÓRICA

Este capítulo faz uma abordagem da disrupção tecnológica que gerou um novo conceito de produtos de comunicação que veio para modernizar a sociedade com um padrão de custo acessível a instituições. Atualmente, empresas e pessoas com menos recursos financeiros e habilidades tem substituído o caro e complicado por algo acessível e simples, tornando possível assim. Dessa forma, pode-se modernizar o ensino inserindo nas escolas públicas o modelo baseado pelas placas open source que atua na Internet das Coisas para elaboração de projetos educacionais.

#### 3.1. Tecnologias Sustentadas e Disruptivas

De acordo com o dicionário da língua portuguesa (Infopédia, 2017), disrupção significa: ato de interromper o curso em um processo. Então interrupção do curso natural de um processo em demanda.

Conforme Christensen et al (2013, p.1), em seu artigo intitulado de Ensino Híbrido: uma Inovação Disruptiva? Uma introdução à teoria dos híbridos, ressalta que há dois tipos básicos de inovação – sustentada e disruptiva – que seguem trajetórias e resultados diferentes. Inovações sustentadas ajudam a melhorar os produtos ou os serviços que frequentemente podem ser vendidos com maiores lucros servindo aos consumidores com a definição de original e desempenho, isto é, como o mercado define historicamente o que é bom.

As inovações disruptivas por sua vez, não procuram trazer produtos melhores para consumidores em mercados estabelecidos. Pelo contrário, oferecem uma nova definição do que é bom – assumindo normalmente a forma de produtos mais simples, mais convenientes e mais baratos que atraem clientes novos ou menos exigentes e com o tempo, elas se aperfeiçoam o suficiente para que possam atender às necessidades de clientes mais exigentes, transformando um setor (CHRISTENSEN *et al* 2013).

Novas tecnologias surgem em decorrência da combinação de antigas tecnologias como híbridas, devido as transformações disruptivas no fator econômico e financeiro, pois as transformações ocorrem ao longo do tempo em favor da sustentabilidade, como ressalta ainda Christensen et al (2013, p.2):

É uma teoria dos híbridos, pois um híbrido é uma combinação da nova tecnologia disruptiva com a antiga tecnologia que representa uma inovação sustentada em relação à tecnologia anterior. Por exemplo, a indústria automobilística desenvolveu vários carros híbridos ao longo de sua transição dos motores movidos a gasolina para fontes alternativas de energia. As empresas líderes querem as virtudes de ambos, então desenvolveram uma

inovação sustentada como carros híbridos que usam tanto a gasolina quanto a energia elétrica. Outros setores incluindo os de escavação, embarcações a vapor, fotografia, varejo e serviços bancários experimentaram um estágio híbrido em seu caminho para aplicar a disrupção pura Christensen et al (2013, p. 2).

Assim fica claro que as tecnologias disruptivas, quando surgem, têm qualidade inferior aos produtos que dominam o mercado, mas eventualmente elas acabam ganhando terreno. Um exemplo disso é o *YouTube*, que tomou muitos lugares de estúdios, e pegando carona, as câmeras digitais que também tomou o espaço de fotógrafos profissionais que virou entretenimento junto às redes sociais como *instagram* e *facebook*. E ainda com a velocidade da internet e o sucesso do *Android* pelo mundo como sistema operacional líder nos dispositivos móveis, e a geração das TV's smart's, destaca-se a *NETFLIX* tirando muitas pessoas dos cinemas por preços de até R\$ 19,90 por mês, permitindo dois *logins* simultâneos. E assim a tecnologia disruptiva permitiu a chegada de uma nova tecnologia, as coisas conectadas em rede, a Internet das Coisas (do inglês: *Internet of Things – IoT*).

### 3.2. Internet das Coisas

#### i) Origem e os conceitos da IoT

A *Internet* por ser um grande marco na comunicação no mundo inteiro, a todo momento é confundido com a *web*. A *Internet* é um sistema global de redes de computadores interconectados que trocam dados por troca de pacotes usando o serviço de protocolos FTP – Protocolo de Transferência de Arquivos padronizado, ou seja, rede de redes, definida pelos padrões TCP/IP<sup>1</sup> CISCO (2005). Já a *web* ou a *World Wide Web*, os *WWW*, é um espaço de informações em que os itens de interesse, referidos como recursos que são identificados por identificadores globais chamados de URI - Uniform Resource Identifier, no português: Identificador de Recurso Uniforme W3C (2004).

A busca por melhores condições de vida e limitação de tempo e rotina cada vez mais atarefada, aliadas a evolução dos dispositivos móveis e a popularização da "internet móvel", originou-se em um novo cenário ligado à internet e a W3C<sup>2</sup> como consequência da maior conectividade entre pessoas e pessoas, pessoas e dispositivos e dispositivos e objetos - a Internet das Coisas. Descrita por Kevin Ashton (1999), cofundador do Auto-ID Center do Massachusetts Institute of Technology (MIT), quando apresentou o termo "Internet das Coisas" em uma de suas apresentações na Procter & Gamble, empresa na qual trabalhava na

<sup>1</sup> Protocolo de Transferência de Camadas e Protocolo de Internet

<sup>2</sup> Comunidade internacional que desenvolve padrões abertos para garantir o crescimento a longo prazo da Web.

época, vinculando a nova ideia de RFID<sup>3</sup>, publicou um artigo: “That Internet of Things Ting<sup>4</sup>”, afirmou que:

{...} O fato de eu ser provavelmente a primeira pessoa a dizer "Internet das coisas" não me dá nenhum direito de controlar como os outros usam a frase. Mas o que eu quis dizer, e ainda significa, é o seguinte: hoje os computadores - e, portanto, a Internet - são quase totalmente dependentes dos seres humanos para obter informações. Quase todos os aproximadamente 50 petabytes (um petabyte é de 1.024 terabytes) de dados disponíveis na Internet foram primeiro capturados e criados por seres humanos, digitando, pressionando um botão de gravação, tirando uma imagem digital ou digitando um código de barras. Os diagramas convencionais da Internet incluem servidores e roteadores e assim por diante, mas deixam de fora os roteadores mais importantes e importantes de todas: as pessoas. O problema é que as pessoas têm tempo, atenção e precisão limitados, o que significa que eles não são muito bons em capturar dados sobre o mundo real<sup>5</sup> (ASHTON, 1999, tradução nossa).

De acordo com esse contexto, surge então a nova tecnologia dando forma a um conceito que até os dias atuais é usada para designar processos que envolvam objetos conectados em rede e que produzam e/ou processem informações em tempo real e de forma autônoma. Como afirma Lemos (2012, p.19):

{...} “uma infraestrutura de rede global dinâmica, baseada em protocolos de comunicação em que “coisas” físicas e virtuais têm identidades, atributos físicos e personalidades virtuais, utilizando interfaces inteligentes e integradas às redes telemáticas. As coisas/objetos tornam-se capazes de interagir e de comunicar entre si e com o meio ambiente por meio do intercâmbio de dados. As coisas reagem de forma autônoma aos eventos do “mundo real/físico” e podem influenciá-los por processos sem intervenção humana direta” (LEMOS, 2012, p.19)

## ii) Tendências da IoT

Segundo o relatório do Symposium ITxpo do Gartner Group (2013), a Internet das Coisas foi apontada como uma das 10 principais tendências estratégicas mundiais de tecnologia de prestação de serviços eletrônicos, e até 2020, a tecnologia estará presente em 95% dos eletrônicos. Esse relatório destaca que a IoT é um conceito que descreve como a

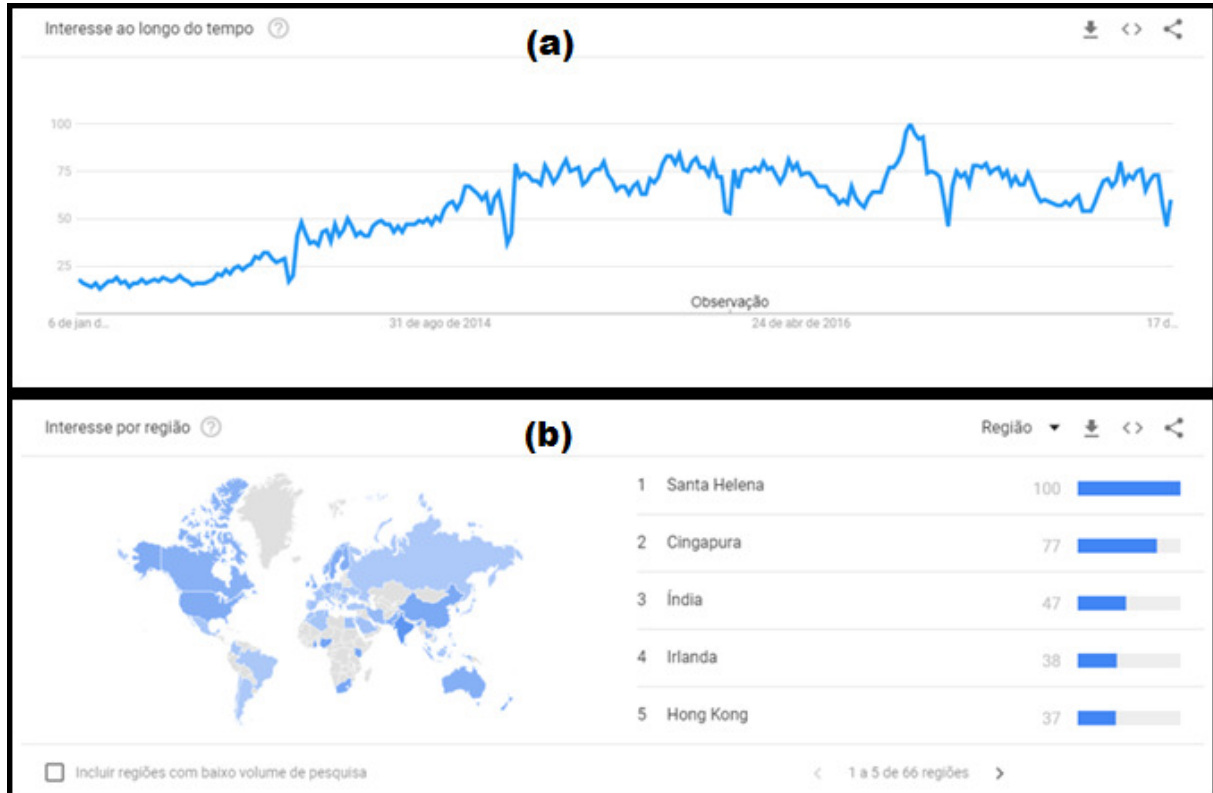
<sup>3</sup> Identificação por Rádio Frequência.

<sup>4</sup> Essa coisa da “internet das coisas”.

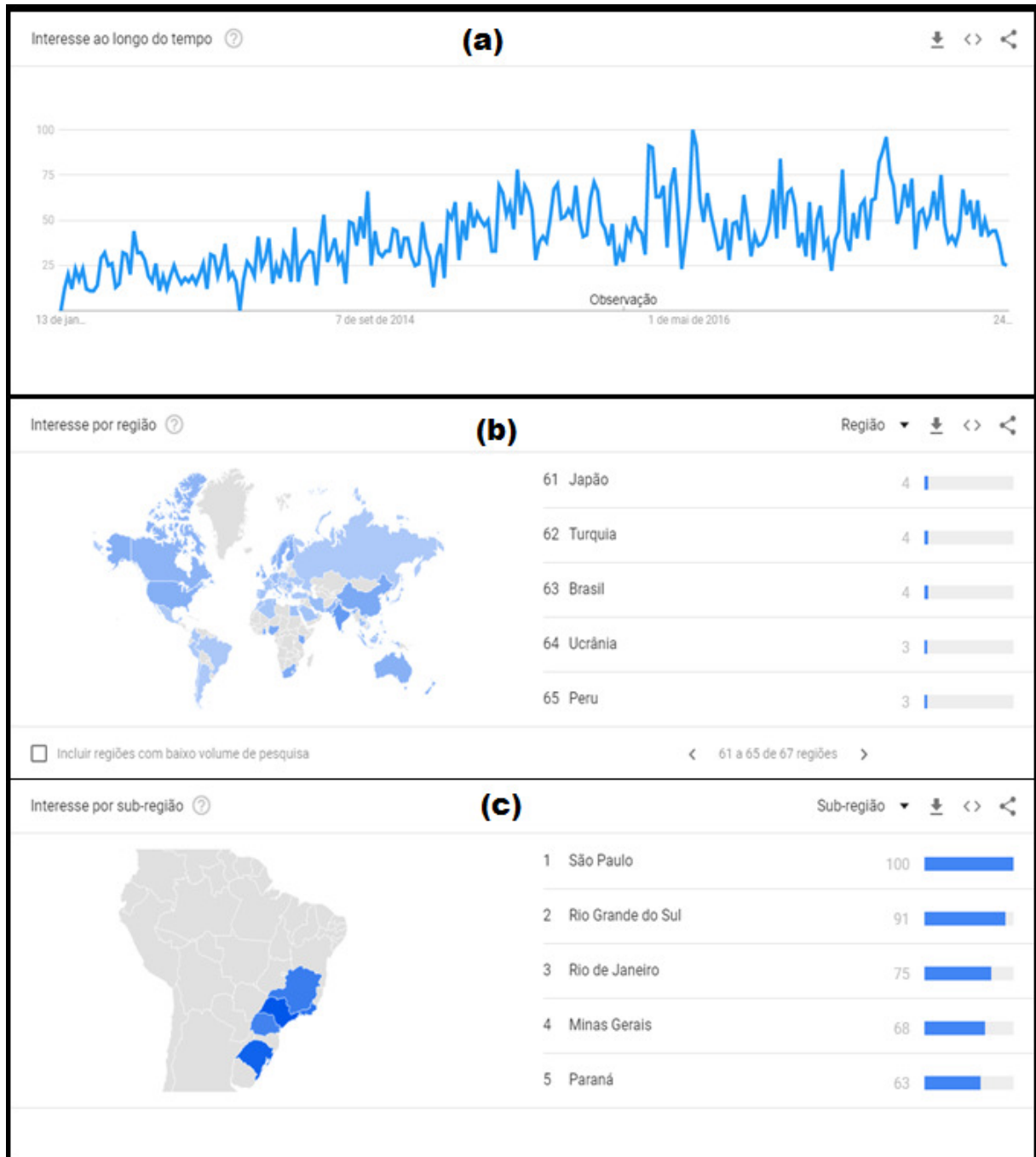
<sup>5</sup> No original: “...The fact that I was probably the first person to say "Internet of Things" doesn't give me any right to control how others use the phrase. But what I meant, and still mean, is this: Today computers—and, therefore, the Internet—are almost wholly dependent on human beings for information. Nearly all of the roughly 50 petabytes (a petabyte is 1,024 terabytes) of data available on the Internet were first captured and created by human beings—by typing, pressing a record button, taking a digital picture or scanning a bar code. Conventional diagrams of the Internet include servers and routers and so on, but they leave out the most numerous and important routers of all: people. The problem is, people have limited time, attention and accuracy—all of which means they are not very good at capturing data about things in the real world.

Internet se expandirá à medida que itens físicos como dispositivos de consumo e ativos físicos estiverem conectados à Internet. Pode-se elencar os principais elementos da IoT, que são: dispositivos móveis que incluem sensores integrados, tecnologias de reconhecimento de imagem e pagamento via NFC. A tecnologia de celular está sendo incorporada em muitos novos tipos de dispositivos, incluindo contêineres farmacêuticos e automóveis. Os smartphones e outros dispositivos inteligentes não usam apenas a rede de celular, eles se comunicam via *NFC*, *Bluetooth*, *LE* e *Wi-Fi* para uma ampla gama de dispositivos e periféricos, como monitores de relógio de pulso, sensores de saúde, cartazes inteligentes e sistemas de entretenimento doméstico.

O termo Internet das Coisas foi mencionado em 1999 por Kevin Ashton, mas só ganhou força em meados de 2008 – 2009. Os gráficos extraídos da ferramenta do Google Trends, que mostra os mais populares termos buscados em um passado recente, apresenta gráficos com a frequência procurado em várias regiões do mundo, e em vários idiomas, o resultado confirmam o crescente interesse da tecnologia pelo mundo ao longo dos 5 anos em diversos países, principalmente na Europa e Ásia (Figura 07) e o Brasil figurando no interesse do assunto na aparecendo no final da lista de países (Figura 08).



**Figura 07:** Interesse por “Internet of Tings” do mundo nos últimos 5 anos. (a) ao longo do tempo; (b) interesse por região no mundo dos cinco maiores em consumo dessa tecnologia. Fonte: Google Trends (2017)



**Figura 08:** Interesse por “Internet of Things” do Brasil nos últimos 5 anos. (a) interesse ao longo do tempo. (b) por região no mundo com destaque para o Brasil. (c) interesse por sub-regiões no Brasil.

Fonte: Google Trends (2017)

Esse tema tem sido pauta prioritária em diversos países, especialmente dos países dos continentes europeu e asiático, sobretudo por meio de programas governamentais dedicados a identificar os impactos e as potenciais oportunidades trazidas pelas inovações. Nesse sentido, a preocupação com a governança da Internet, traduzida na capacidade de definir modelos de funcionamento da rede, torna-se fundamental para garantir seu desenvolvimento sustentável e o atendimento pleno aos interesses da sociedade.

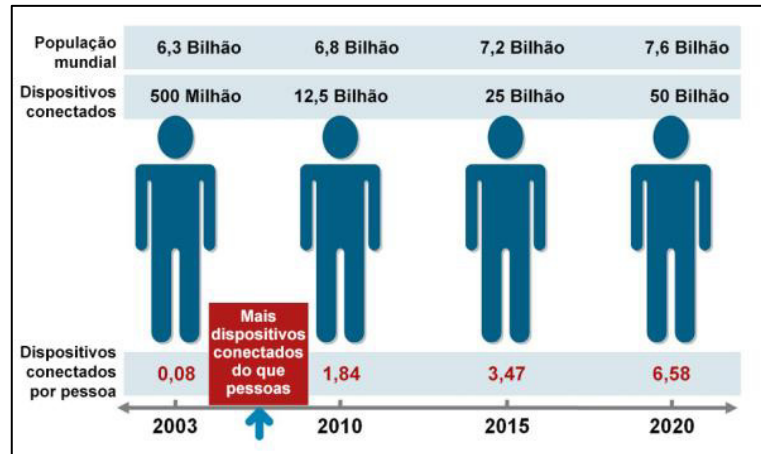
Kranenburg (2012 *apud* LEMOS 2012), aponta seis fatores importantes para o desenvolvimento da IoT, são eles: a) o surgimento dos códigos de barra ubíquos em 1974, mostrando um padrão de uso; b) o surgimento de etiquetas RFID, que hoje se confundem com a própria Internet das Coisas; c) o barateamento do armazenamento de dados; d) o protocolo IPV6 que gera um código de identificação único na rede para cada objeto; e) a concretização da computação verdadeiramente ubíqua; e, f) os atores humanos que embarcam dispositivos informacionais de forma explícita e que cada vez mais embarcarão objetos comuns dotados dessas qualidades. Hoje humanos acoplados a não humanos comunicacionais são os novos híbridos: celulares, *iPads*, redes sociais.

iii) Os dispositivos e objetos ultrapassarão as pessoas

Considere que a IoT representa a próxima evolução da Internet, dando um grande salto em números de dispositivos e objetos e na capacidade de coletar, analisar e distribuir dados que se pode transformar em informações. A redução contínua do custo e do tamanho dos *microprocessadores*, com proporcional aumento da capacidade e velocidade de processamento, prevista pela famosa *Lei de Moore*, foi decisiva para o advento da IoT. O Fundador da Intel, Gordon Moore, antecipou em 1965 que a quantidade de transistores em um circuito integrado comercialmente viável dobraria a cada 18 meses, mantendo o custo de fabricação. O prognóstico mostrou-se extremamente preciso, ainda que fosse uma profecia autorrealizável, que levou fabricantes de chips a uma busca incessante por processadores cada vez menores, mais rápidos e mais baratos. Embora algumas análises apontem para o declínio da lei, ou, ainda que processadores passem a ser fabricados por diferentes tecnologias - como nanotubos de carbono, o fenômeno tem influência definitiva no contexto da IoT (LACERDA (2015).

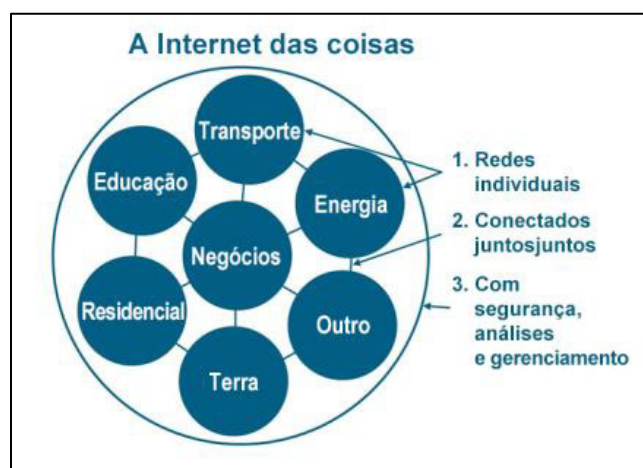
De acordo com a CISCO (2011), em 2020 haverá mais “coisas e objetos” conectados à rede, do que pessoas no mundo, a quantidade de dispositivos conectados à rede, será de 50 bilhões por 7,6 bilhões de pessoas, ou seja, será 6,58 dispositivos por pessoa, como ilustra a (Figura 09).





**Figura 09:** Previsão de crescimento da IoT.  
Fonte: Cisco 2011

Dessa forma, Evans (2011), cita que até o momento, a IoT é composta por uma coleção livre de redes diferentes e criadas para determinada finalidade. A exemplo disso, os carros atuais que possuem várias redes para controlar a função do motor, recursos de segurança, sistemas de comunicação, entre outros. Outro exemplo são os prédios comerciais e residenciais que também dispõem de vários sistemas de controle para aquecimento, ventilação e ar-condicionado, serviços telefônicos, segurança e iluminação. À medida que a IoT evolui, essas redes e muitas outras estarão conectadas com mais recursos de segurança, análise e gerenciamento como mostra a (Figura 10), permitindo que a IoT se torne ainda mais importante com relação ao que pode fazer para ajudar as pessoas a obterem novas conquistas.



**Figura 10:** A IoT pode ser vista como uma rede das redes.  
Fonte: Cisco 2011

#### iv) Aplicações da IoT

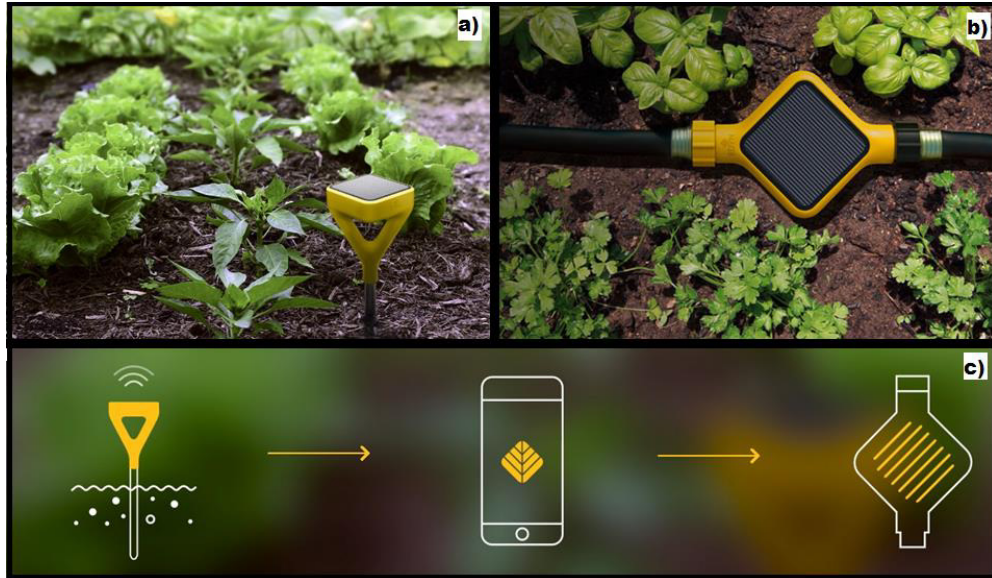
A Lopez Research (2013), fez um apontamento para a aplicação da IoT, e complementa ainda, que é um sistema em que os elementos no mundo físico, e sensores

dentro ou acoplados a esses elementos, estão conectados à Internet por meio de conexões de Internet sem e com fio. Os sensores dispõem de vários tipos de conexões de área local como *RFID*, *NFC*, *Wi-Fi*, *Bluetooth* e *Zigbee*. Os sensores apresentam também a conectividade de longa distância como *GSM*, *GPRS*, *3G* e *LTE*. E além do mais a Internet das Coisas irá:

- Conectar objetos inanimados e seres vivos - Os primeiros testes e implantações da Internet das Coisas começaram com a conexão de equipamento industrial. Hoje, a visão da IoT expandiu-se para conectar tudo, desde equipamentos industriais a objetos de uso diário. Os tipos de itens vão de turbinas a gás a automóveis e medidores de utilitários. Eles também podem incluir organismos vivos como plantas, animais de produção e pessoas. Por exemplo, o Projeto de Monitoramento de Vacas, que utiliza dados coletados em etiquetas de posicionamento de rádio para monitorar e acompanhar a saúde e comportamento do rebanho.
- Usar sensores para coleta de dados - Os objetos físicos a serem conectados terão um ou vários sensores. Cada sensor monitorará uma condição específica como local, vibração, movimentação e temperatura. Na IoT, esses sensores conectam-se entre si e com os sistemas que podem entender ou apresentar informações dos feeds de dados do sensor. Esses sensores fornecerão novas informações para sistemas de uma empresa e para pessoas.
- Alterar quais tipos de itens se comunicam em uma rede IP - No passado, as pessoas se comunicavam com pessoas e máquinas. Imagine se todos os seus equipamentos tivessem a capacidade de se comunicar. Os objetos capacitados pela IoT irão compartilhar informações sobre suas condições e seu ambiente correspondente com pessoas, sistemas de software e outras máquinas. Essas informações podem ser compartilhadas em tempo real ou coletadas e compartilhadas em intervalos definidos. Futuramente, tudo terá uma identidade e conectividade digitais, significando que as pessoas poderão identificar, gerenciar e se comunicar com os objetos.

De acordo com o que foi citado, podemos demonstrar então alguns exemplos de aplicações da IoT na sociedade moderna em pleno funcionamento e o impacto que a Internet das Coisas representa atendendo praticamente todos os setores da sociedade propondo novos modelos de serviços e solucionando os mais diversos problemas encontrados decorrentes do meio em questão.

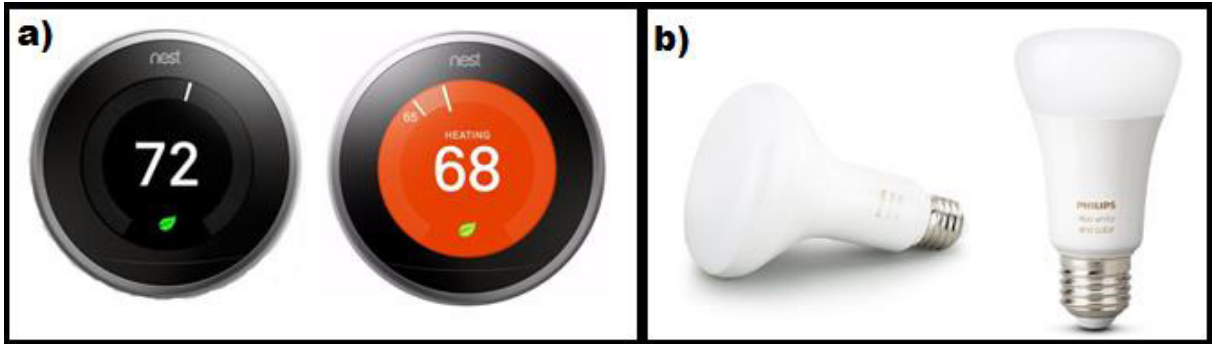
Na agricultura (Figura 11), sensores espalhados em plantações que podem dar informações bastante precisas sobre temperatura, umidade do solo, probabilidade de chuvas, velocidade do vento e outras informações essenciais para o bom rendimento do plantio. O funcionamento desse sistema funciona de forma autônoma.



**Figura 11:** Sensores autônomos para agricultura: a) sensor Edyn Garden, fixado no solo, monitora as condições de temperatura do solo, b) válvula de água fornece a quantidade de água necessária evitando o desperdício de água, c) comunicação dos dispositivos conectado à rede wifi, envia as informações de temperatura do solo para o aplicativo Edyn do smartphone do responsável e solicita o acionamento dos mesmos.

Fonte: edyn.com

Nas casas, as questões de temperatura ambiente, um termostato que controla a temperatura do lar e envia um alerta pelo celular se houver algo de errado na residência, como por exemplo uma panela esquecida no fogão ou um curto circuito na fiação elétrica. O termostato da Nest (Figura 12a) é uma parte do sistema de aquecimento e refrigeração que aprende a rotina de temperatura da residência de forma autônoma e tudo conectado. E o sistema inteligente de iluminação wireless da Philips Hue, (Figura 12b), baseado na conexão sem fio Zigbee LightLink, uma tecnologia confiável, segura e de baixo consumo de energia para controlar as luzes com recursos e melhorias adicionados continuamente ao sistema para torná-lo ainda mais útil com atualizações de software e de firmware sendo feitas via conexão sem fio e diretamente nas lâmpadas. O sistema Philips Hue pode ser facilmente integrado a outros sistemas baseados em Zigbee para obter automação residencial adicional e tudo controlado pelo celular.



**Figura 12:** Exemplos: a) Termostato Nest; b) Lâmpadas de Led sem fio Philips Hue.  
Fonte: nest.com e Philips

Na Saúde, pulseiras e relógios para monitoramento da saúde e atividades físicas da FitFib, (Figura 13) envia dados obtidos por esses dispositivos como batimentos cardíacos e o sobe e desce de pressão, entre outros são sincronizados com o smartphone e podem ser enviados diretamente para o médico que acompanha o paciente.



**Figura 13:** Pulseiras FitFib.  
Fonte: FitFib

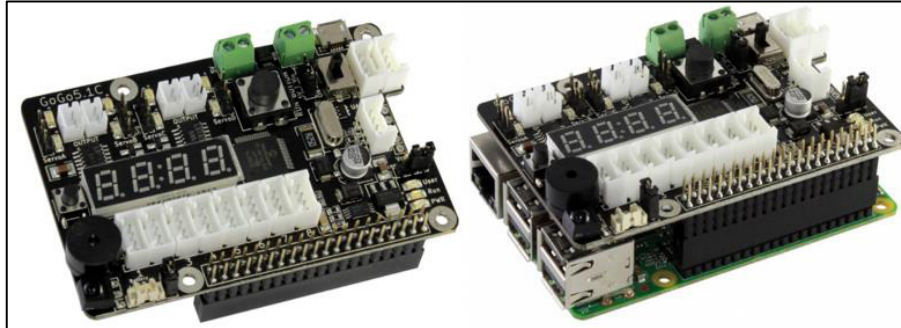
### 3.3 Placas de baixo custo voltadas para a programação aberta (*open source*)

Este tópico fará uma abordagem nas mais diversas placas e módulos de diferentes fabricantes considerados de baixo custo e programável voltados para todos os setores da educação como modelos de desenvolvimento que promove um licenciamento livre para o design ou esquematização de um produto, dando a possibilidade para que qualquer um consulte, examine ou modifique o produto seguindo a tendência dessa nova onda da Internet das Coisas.

#### a) *GogoBoard*

Trata-se de um dispositivo de hardware de código aberto de baixo custo para robótica educacional, experiências científicas e detecção ambiental. Tanto crianças como adolescentes podem usar a placa para construir robôs, medir e registrar dados ambientais, realizar pesquisas científicas, criar controladores de jogos, criar instalações de arte interativa. Atualmente há três tipos de portas de expansão disponíveis I2C e UART (serial) são para sensores digitais e módulos de extensão e um conector de 40 pinos que permite que a placa GoGo se integre a um Raspberry Pi. Especificamente, seu programa é executado no Gogoboard e diz ao

Raspberry Pi quais funções você deseja que ele faça. Esta abordagem expõe as crianças ao poder do Raspberry Pi, mantendo a simplicidade de programação do original GoGo Board, (Figura 14).



**Figura 14:** Placa GoGo Board Versão 5.1.

Fonte: [www.gogoboard.com.org](http://www.gogoboard.com.org).

Por ser um dispositivo programável com entradas e saídas de sensores pode controlar motores e outros tipos de atuadores. O público alvo visa especialmente os alunos jovens de 10 a 18 anos e aqueles que são novos para a eletrônica. O princípio do projeto é permitir que os alunos passem o maior tempo possível nas ideias de sua criação desejada e menos nos detalhes técnicos da eletrônica de baixo nível envolvida. Esta placa é compatível com muitos sensores eletrônicos e atuadores que estão amplamente disponíveis, podendo ser programada em dois ambientes de programação:

- widget GoGo, um ambiente que necessita um computador com cabo USB e fazer as execuções dos sensores de exibição, controle dos motores manuais, configuração dos módulos complementares de testes, atualização de firmware e ainda podendo ser controlada com outra linguagem de programação semelhante ao LOGO.
- Tinker, um ambiente de arrastar e soltar baseado em blocos de programação semelhante a linguagem Scratch que por sua vez baseado na linguagem LOGO.

O **Tinker** é frequentemente usado por iniciantes que não estão familiarizados com o idioma do Widget GoGo Board. O Tinker traduz o programa visual em texto e o envia para o GoGo Widget. Portanto, precisa manter o GoGo Widget em execução para usar o Tinker.

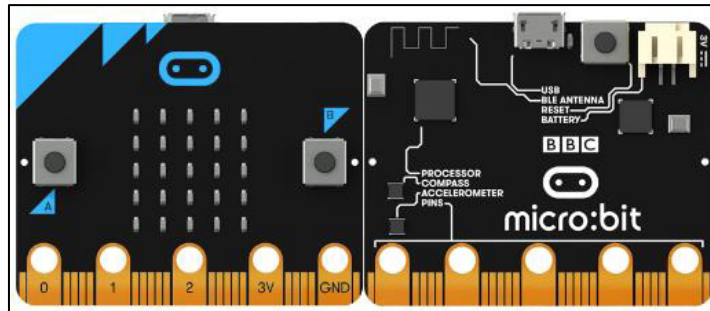
A placa GoGo Board, pode ser adquirida no próprio site: [www.gogoboard.org](http://www.gogoboard.org).

- b) Micro:bit, o computador de placa única criado pela BBC para o ensino de programação para crianças

Trata-se de um computador pequeno criado pela BBC, cujo propósito visa ensinar programação às crianças. O micro:bit (Figura 15) foi uma iniciativa motivada em parte pela sociedade real de introdução à computação como uma disciplina obrigatória nas escolas da Inglaterra em setembro de 2014, com o objetivo de integrar o novo currículo que substituiu o



currículo nacional de TIC para promover o pensamento computacional e elevando a importância da programação. Parte desta iniciativa, modelada em esforços passados da BBC, era a criação de um computador portátil barato, fácil de programar e versátil o suficiente para que pudesse ser integrado a uma série de projetos relacionados ao cotidiano de seus usuários. O pequeno computador de bolso está sendo gratuito no Reino Unido para crianças de até 12 anos.



**Figura 15:** Placa Microcomputador BBC Micro:bit.

Fonte: microbit.org

Atualmente tem sido utilizado nas escolas de países como Reino Unido, Finlândia, Islândia, Singapura e Sri Lanka. Com diversas atividades e planos de aula voltados para professores de escolas primárias e secundárias, segundo o site da fundação. Para ser programado precisa-se de um computador conectado através de um cabo USB, sendo suportado por todas as plataformas de Sistemas Operacional: Macs, PCs e Sistemas Linux (incluindo o Raspberry Pi), vindo com uma aplicação divertida para programação. Pode ser codificado a partir de qualquer navegador da Web em Blocos de arrastar e soltar Javascript Python editor, Scratch e outros (microbit.org, 2017). Características:

- Possui 25 LEDs programáveis individualmente permitindo a exibição de textos, números e imagens.
- Possui 2 botões programáveis para que os códigos sejam acionados no dispositivo.
- Pinos de conexão física, são 25 conectores externos no conector de borda do micro: bit, para ser integrados como jumpers nos motores, LEDs ou outros componentes elétricos com os pinos, para conectar sensores a serem controlados pelo código.
- Sensores de luz e temperatura, ao inverter os LEDs da tela para se tornar uma entrada, a tela de LED funciona como um sensor de luz básico, permitindo detectar a luz ambiente.
- Sensores de movimento (acelerômetro e bússola) permite que o micro: bit detecte a temperatura ambiente atual, em graus Celsius.
- Comunicação sem fio, via rádio e Bluetooth e Interface USB

Há uma significativa quantidade de componentes e sensores que ajudam a transformar o Micro:bit em muitos projetos, dependendo apenas do que se deseja. A placa Micro:bit pode ser encontrada no *aliexpress.com*, custando cerca de R\$ 58,13 (frete incluso).

### c) Arduino

Arduino nasceu no Ivrea Interaction Design Institute – (Ivrea Instituto de Design e Interação), como uma ferramenta fácil para prototipagem rápida, destinada a estudantes sem experiência em eletrônica e programação. Versa sobre uma plataforma eletrônica de código aberto baseada em *hardware* e *software* fáceis de usar. Tais placas são capazes de ligar sensores de luzes, identificar o dedo de um botão, receber uma mensagem e transformá-lo em uma saída, pode ativar um motor, acendendo um LED e publicar algo online. Podem-se escrever códigos e informar ao *hardware* o que fazer, por meio de um conjunto de instruções para o micro controlador no quadro.

Utiliza-se a linguagem de programação *Arduino* e o *software Arduino* (IDE), com base no processamento. Esta placa tem sido destaque para milhões de projetos, desde objetos comuns até instrumentos científicos complexos espalhados pelo mundo. Uma comunidade mundial de profissionais e estudantes de computação e eletrônica se reuniram em torno desta plataforma de código aberto e suas contribuições somaram uma incrível quantidade de conhecimento acessível que pode ser de grande ajuda para os iniciantes e profissionais.

São várias as opções para a placa Arduino de baixo custo que são usadas dependendo do tipo de projeto, como o modelo UNO (Figura 16), uma das mais usadas em todo o mundo. Trata-se de uma placa de microcontrolador baseada no *ATmega328P*, possuindo 14 pinos de entrada/saída digitais (dos quais 6 podem ser utilizados como saídas PWM), 6 entradas analógicas, um cristal de quartzo de 16 MHz, uma conexão USB, uma tomada de energia, um cabeçalho ICSP e um botão de reinicialização.



**Figura 16:** Arduíno uno.

Fonte: Arduíno.com

"Uno" significa 1 (um) em italiano e foi escolhido para marcar o lançamento do *Arduino Software (IDE)* 1.0. A placa Uno e a versão 1.0 do *Arduino Software (IDE)* foram as versões de referência do *Arduino*, agora desenvolvidas para versões mais recentes. Esta placa é a primeira de uma série de placas USB *Arduino* e o modelo de referência para a plataforma *Arduino* para uma extensa lista de placas atuais. As características da placa são:

- Microcontrolador: ATmega328P
- Tensão Operacional: 5V
- Tensão de Entrada: 7-12V
- Pinos de E/S Digitais: 14 das quais 6 fornecem saída PWM
- PWM Digital I/O Pins: 6
- Pinos de Entrada Analógicos: 6
- Corrente DC por Pino I/O: 20mA
- Corrente DC para Pin 3.3V: 50mA
- Memória Flash: 32KB (ATmega328P), dos quais 0,5 KB usado pelo
- SRAM: 2KB (ATmega328P)
- EEPROM: 1KB (ATmega328P)
- Velocidade do Relógio: 16 MHz
- LED\_BUILTIN: 13

Para seu funcionamento é necessário um computador com Sistema Operacional (Windows, MAC OS X, Linux e Raspberry Pi), com cabo USB conectado para codificar online com o Arduino Web Editor, um ambiente desenvolvido em Java baseado em processamento, ou software Arduino de código aberto (IDE), que facilita a criação de códigos para serem executados em qualquer placa Arduino, sendo que pode ser programado em vários outros softwares feitos para Arduino.

O projeto Arduino conta com vários parceiros educacionais espalhados em alguns países do mundo com o nome de CTC 101, um programa STEAM voltado para estudantes do ensino médio e secundário. As bases da programação, eletrônica e mecânica são transferidas para eles através de uma série de projetos divertidos, bem documentados e fáceis de implementar. O CTC 101 foi concebido para professores e formadores que trabalham no domínio da educação e foi adaptado para alunos do ensino médio de 13 a 17 anos. O CTC 101 é ideal para o desenvolvimento profissional de educadores, oferecendo treinamento e suporte de alta qualidade por profissionais qualificados:

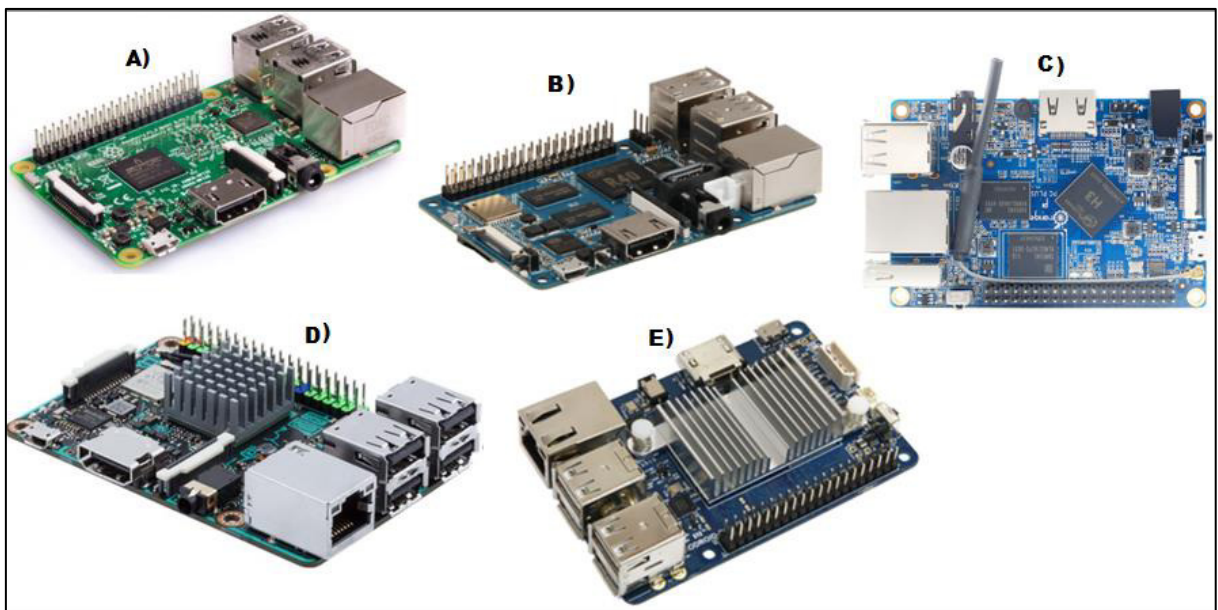
- Auriga Consulting: Espanha
- Edilar: México
- OneMaker Group: Singapura
- Media direta (CampuStore): Itália
- RS Components: Reino Unido



- Siculiana Media: Itália
- Academia Sparklabs: Coreia
- Think Ahead: África do Sul
- Ultra-Lab: Espanha
- WebScience: Itália

### 3.4 Placas de opção de sistema operacional voltados para a programação aberta (*open source*)

Esta sessão apresentará 5 modelos de placa de baixo custo open source de mesma configuração, identificadas por letras de A – E com mesmo padrão de configuração, mas de fabricantes diferentes (Figura 17):



**Figura 17:** Placas open source com opção de Sistema Operacional open source

Fonte: Autor, adaptado de: [www.raspberrypi.org](http://www.raspberrypi.org), [www.banana-pi.org](http://www.banana-pi.org), [www.orangepi.org](http://www.orangepi.org), [www.asus.com](http://www.asus.com) e [hardkernel.com](http://hardkernel.com)

#### A) Raspberry PI 3 modelo B

O Raspberry Pi ([raspberrypi.org](http://raspberrypi.org), 2018), é um computador de placa única, fruto de um projeto da fundação de mesmo nome situada no Reino Unido com foco educacional. A instituição desenvolveu o computador com o compromisso de simplicidade e preço baixo para que ele pudesse chegar em escolas de todo o mundo. Assim, crianças teriam acesso a fundamentos de programação e teriam também um entendimento mais aprofundado sobre a natureza dos computadores e o seu funcionamento. A Fundação Raspberry Pi consegue diversos componentes da placa a preços abaixo do mercado por conta de parcerias com os fabricantes por ser uma instituição de caridade que fornece computadores de baixo custo para usar e aprender programação. Os pequenos computadores têm opções de Sistema

Operacional Linux ARM<sup>6</sup> de diversas distribuições que contam com alguns softwares que ensina princípios de programação.

Tudo a respeito do computador está disponível em código aberto na página da Fundação com outros modelos de Raspiberry Pi, que podem ser vistos no site da fundação. O Pi M3, é última geração com configuração mais robusta que as demais. Como foi criado para propósitos educacionais, o baixo custo e a versatilidade do Raspberry Pi acabaram permitindo que ele se tornasse o motor de vários projetos inovadores. A Raspberry Pi no mercado internacional o custo é mais em conta, custando cerca de R\$ 122,32 com frete incluso no aliexpress.com. Especificações:

- Quad Core 1.2GHz Broadcom BCM2837 CPU de 64 bits
- 1 GB de RAM DDR2
- LAN sem fio BCM43438 e Bluetooth Low Energy (BLE) a bordo
- GPIO estendido de 40 pinos
- 4 Portas USB 2.0
- Saída estéreo de 4 pólos e porta de vídeo composto
- HDMI de tamanho completo
- Porta da câmera CSI para conectar uma câmera Raspberry Pi
- Porta de exibição DSI para conectar um ecrã táctil Raspberry Pi
- Porta Micro SD para carregar seu sistema operacional e armazenar dados
- Fonte de alimentação Micro USB comutada atualizada 5V/2A - 2.5A

#### B) *Banana Pi M2 Berry*

A banana Pi (banana-pi.org, 2018), é uma série de computadores de placa única de tamanho de um cartão de crédito com base em conceito de baixo custo para desenvolvimento interno de software e hardware e aprendizagem de software escolar. Seu design de hardware foi influenciado por Raspberry Pi em 2013. É produzido pela empresa chinesa Shenzhen SINOVOIP Co. Ltda.

O software Banana Pi é compatível com as placas Raspberry Pi. A Banana Pi também pode executar vários Sistemas Operacionais NetBSD, Android, Ubuntu, Debian, Arch Linux e Raspbian, CPU com os requisitos da armhf, referência à ARM. Ele usa o Allwinner SoC (sistema no chip) e, como tal, é principalmente coberto pela porta linux-sunxi. Banana Pi é o hardware de código aberto e plataforma de software que foi projetado para ajudar projetos de educação no mundo inteiro, desenvolvido na China pela Banana Pi Team, com a intenção de promover a educação STEM (ciência, tecnologia, engenharia e matemática) nas escolas. São

---

<sup>6</sup> Referência ao processador (Máquina RISC Avançada) e RISC é a arquitetura, conjunto de instruções utilizadas no processamento.

vários modelos de placas do Banana Pi, projetadas para diferentes funções de estudo, outros modelos podem ser vistos no site da fundação. Essa placa no mercado internacional o custo é mais em conta, custando cerca de R\$ 132,74 mais frete de R\$2,20 no aliexpress.com.

Especificações:

- Quad Core ARM Cortex A7 CPU V40.
- 1GB DDR3 SDRAM.
- WiFi, LAN (AP6212) e Bluetooth Low Energy (BLE) a bordo.
- Interface SATA
- 4 Portas USB 2.0.
- GPIO estendido de 40 pinos
- Saída estéreo de 4 pólos e porta de vídeo composto
- HDMI de tamanho completo
- Porta da câmera CSI para conectar uma câmera Raspberry Pi e outros
- Porta de exibição DSI para conectar um ecrã táctil Raspberry Pi e outros
- Porta Micro SD para carregar seu sistema operacional e armazenar dados
- Fonte de alimentação Micro USB comutada atualizada 5V/2A.
- Botão Power
- Botão Reset

#### C) Orange Pi PC Plus

Orange Pi (orangepi.org, 2018), é um computador de placa única e código aberto. Desenvolvido para fins educacionais e inspirado pela Raspberry Pi. É projetado na China pela Shenzhen Xunlong Software CO Limited. Ele pode rodar várias distribuições Linux, como por exemplo: Android, Ubuntu, Debian, Fedora, Raspbian, Arch Linux, Open SUSE, Open Wrt e outros. Ele usa o AllWinner SoC (sistema no chip). Também conta com vários modelos de placa para diferentes propósitos de estudo, pesquisa e lazer. Outros modelos podem ser vistos no site da fundação. Essa placa no mercado internacional o custo é mais em conta, custando cerca de R\$ 87,74 com frete incluso no aliexpress.com.

Especificações:

- H3 Quad-core Cortex-A7 H.265 / HEVC 4K
- 1GB DDR3 (compartilhado com o GPU)
- WiFi, LAN a bordo.
- 3 Portas USB 2.0.
- 1 Porta USB OTG
- GPIO estendido de 40 pinos
- Entrada para microfone
- HDMI de tamanho completo
- Porta da câmera CSI para conectar uma câmera Raspberry Pi e outros
- Porta de exibição DSI para conectar um ecrã táctil Raspberry Pi e outros
- Porta Micro SD para carregar seu sistema operacional e armazenar dados (Max.32GB)
- Fonte de alimentação DC 5V/3A.

- Botão Power

#### D) Tinker Board ASUS

Tinker Board (asus.com, 2018), é um computador de placa única projetada para alavancar uma excelente compatibilidade mecânica para a construção de projetos educacionais. A ASUS segue a tendência da IoT com essa pequena placa computacional extremamente capaz de construir grandes idéias na realidade da educação. O Tinker Board, possui um RK3288 SoC (sistema no chip) com base em ARM e está equipado com quatro núcleos para melhorar o desempenho das aplicações multithread. Funciona em frequências de até 1.8GHz, melhorando o desempenho em todas as aplicações, esse aumento na contagem do núcleo da CPU, ajuda a permitir um desempenho significativamente mais rápido em uma ampla gama de aplicativos, ampliando e aprimorando os recursos do projeto. O Tinker Board, é capaz de rodar várias distribuições Linux ARM, além do Tinker OS, sistema operacional próprio disponibilizado no seu site com foco educacional incluindo IDLE, Python, Squeak, Scratch.

A ASUS é uma empresa multinacional de hardware e eletrônicos de consumo com sede em Taiwan, criada em 1989. Dedicada à criação de produtos para a vida inteligente de hoje e de amanhã, a ASUS desenvolveu a primeira placa mãe de jogos do mundo. A ASUS tornou-se amplamente conhecida na América do Norte quando revolucionou a indústria de computadores em 2007 com o Eee PC™. Hoje, a empresa é pioneira em novas tendências móveis com a série ASUS ZenFone™ e está desenvolvendo rapidamente produtos de realidade virtual e aumentada, bem como dispositivos IOT e tecnologias de robótica. A ASUS desenvolveu o Zenbo, um robô de casa inteligente projetado para fornecer assistência, entretenimento e companheirismo às famílias. Essa placa no mercado internacional o custo é mais em conta, custando cerca de R\$ 288,39 com frete incluso no aliexpress.com. Especificações:

- Quad Core Cortex-A12 1.8GHz de 64 bits
- 2 GB de RAM DDR3
- LAN sem fio BCM43438 e Bluetooth Low Energy (BLE) a bordo
- GPIO estendido de 40 pinos
- 4 Portas USB 2.0
- Saída HD de áudio 192K/24bits
- Vídeo HDMI com resolução 4K
- Porta da câmera CSI para conectar uma câmera Raspberry Pi
- Porta de exibição DSI para conectar um ecrã táctil Raspberry Pi
- Porta Micro SD para carregar seu sistema operacional e armazenar dados
- Fonte de alimentação Micro USB comutada atualizada 5V/2A - 2.5A

- Botão Power

#### E) Odroid C1+

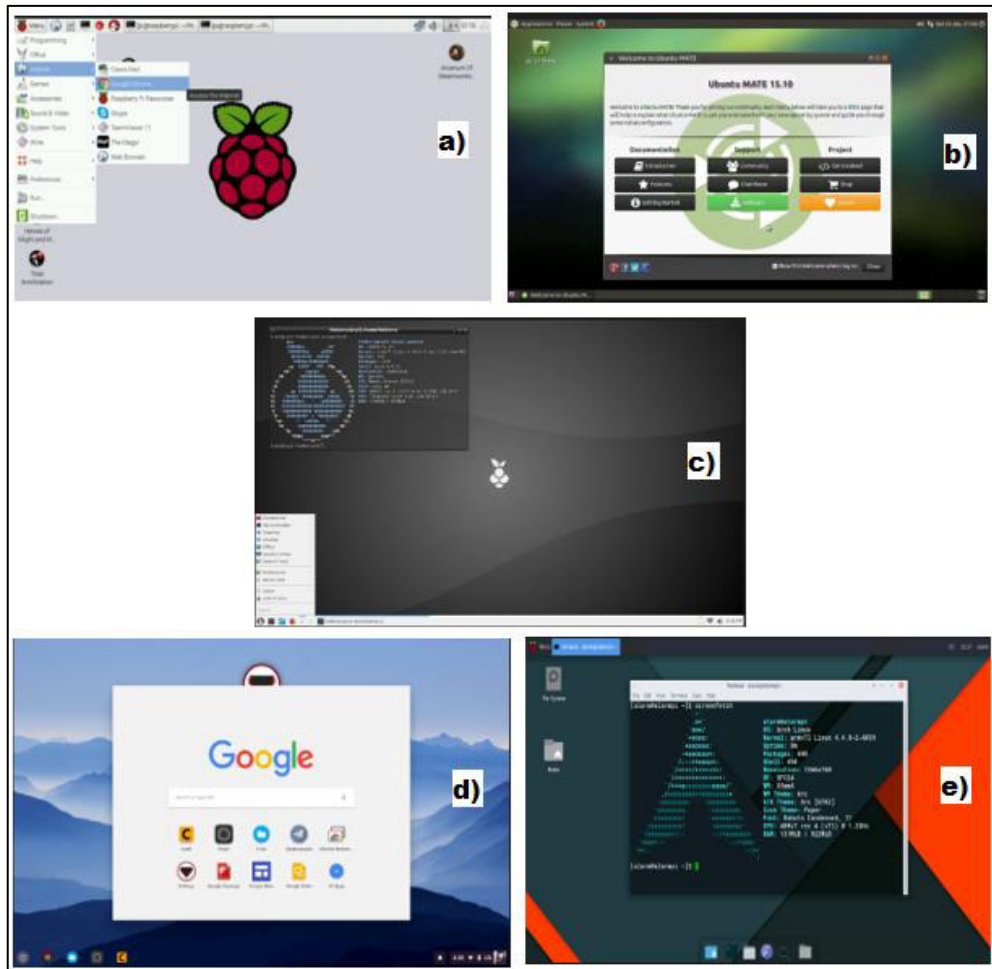
O ODROID (hardkernel.com, 2018), significa Open + Droid. É uma plataforma de desenvolvimento para o hardware, bem como o software. É um computador de placa única que pode rodar várias distribuições Linux ARM. Assim como a Raspberry Pi, a ODROID desenvolveu suas placas microprocessadas a favor da educação. Foi desenvolvida pela Corporação Hardkernel co Ltd, na Coreia do Sul, A ODROID C1+, praticamente possui a mesma configuração da Raspberry Pi 3. No site da companhia encontra-se vários outros modelos de placas e esse modelo está mais em conta no mesmo site a um valor de \$35,00

Especificações:

- Quad Core Cortex-A5 1.5GHz de 32 bits
- 1GB DDR3 SDRAM.
- LAN (AP6212) a bordo.
- 4 Portas USB 2.0.
- 1 Porta USB OTG
- GPIO estendido de 40 pinos
- Saída estéreo de 4 pólos e porta de vídeo composto
- HDMI de tamanho completo
- Porta Micro SD para carregar seu sistema operacional e armazenar dados
- Fonte de alimentação DC 5V/2A.

### 3.5 Sistemas operacionais voltados para a programação open source

Esta sessão apresentará 5 modelos de Sistema Operacional aberto numerados de 1-5, como opção para instalação nos diferentes tipos de placas de baixo custo voltados para a programação aberta (figura 18):



**Figura 18: Sistemas Operacionais open source**

Fonte: Autor, adaptado de: [www.raspberrypi.org](http://www.raspberrypi.org), [www.ubuntu-mate.org](http://www.ubuntu-mate.org), [fedberrry.org](http://fedberrry.org), [flintos.io](http://flintos.io) e [www.archlinux.org](http://www.archlinux.org)

#### a) Raspbian

É o sistema operacional ARM desenvolvido pela Fundação Raspberry Pi, com o intuito de funcionamento com o Raspberry Pi, distribuído gratuito pelo próprio site da fundação e podendo ser instalado com o NOOBS, que é o software instalador que já contém o Raspbian que também oferece alternativa de instalar outros sistemas operacional. E também podendo ser baixado a imagem .ISO compactado. E o site da fundação oferece outros sistemas operacionais como opção para o Raspberry Pi, e explica o passo a passo da instalação dos diferentes SO ([raspberrypi.org](http://raspberrypi.org), 2018).

#### b) Ubuntu Mate

Ubuntu Mate é uma distribuição Linux ARM de código aberto, e por ser aberto é um software que pode ser usado livremente, alterado e compartilhado (em forma modificada ou não modificada) por qualquer pessoa no mundo. Em poucas palavras, o Ubuntu MATE é gratuito, no sentido mais verdadeiro da palavra. É desenvolvido pela Canonical LTDA

sedeada no Reino Unido, que mantém constantemente o desenvolvimento em pró de melhorias para Sistemas Operacionais Linux sob interface gráfica (GUI) – Interface Gráfica de Usuário. O Mate, é um Sistema Operacional completo contendo gerenciador de arquivos que pode conectá-lo a seus arquivos locais e em rede, um editor de texto, pacote de escritório LibreOffice, calculadora, gerenciador de arquivos, visualizador de imagens, visualizador de documentos pdf, navegador de internet Mozilla Firefox, Chromium, monitor de sistema, e terminal. Todos os quais são altamente personalizáveis e gerenciados através de um centro de controle.

O Ubuntu Mate ARM, está atualmente na versão 17.10 que pode ser baixado como imagem .ISO diretamente do site [ubuntu-mate.org/download](http://ubuntu-mate.org/download) gratuitamente para uso pessoal e profissional ([ubuntu-mate.org](http://ubuntu-mate.org), 2018).

#### c) FedBerry

FedBerry é um Fedora Remix construído especificamente para uso com computadores Raspberry Pi 2 e 3. O SO pode ser baixado gratuitamente sob imagem de disco pelo site da [fedberry.org](http://fedberry.org). O FedBerry é fornecido ou suportado pelo Projeto Fedora Linux para PCs convencional que foi projetado especialmente para Raspberry Pi, por não ter em seu projeto um Sistema Operacional para placas open source ([fedberry.org](http://fedberry.org), 2018)

#### d) Flint OS

O Flint OS é um sistema operacional rápido, simples e produtivo, baseado no sistema operacional aberto Chromium e com a aparência do Android que ainda usa os aplicativos em APPS. O Flint começou com uma visão em que todas as aplicações e serviços que usamos hoje estarão vivendo na nuvem. Com a tecnologia de plataforma de navegador em performances de front-end, que se refere à interface que trabalha com a parte da aplicação que interage diretamente com o usuário sempre avançadas, não é surpreendente que a maioria das coisas que fazemos hoje com a internet pode ser feita através de uma única janela do navegador. Estamos entrando em uma era em que aplicativos instaláveis logo se tornarão históricos. O Flint OS pode ser baixado diretamente no site da distribuição Linux ARM [flintos.io/download](http://flintos.io/download) ([flintos.io](http://flintos.io), 2018)

#### e) Arch Linux

O Arch Linux ARM foi iniciado em 2009 como uma distribuição de tipo optware chamada PlugApps para o mercado de computadores de plugue emergente, seguido logo por

uma transição para ser a primeira porta ARM da distribuição Arch Linux que suporta sistemas ARMv5. No final de 2010, uma porta quase completa estava disponível. Em meados de 2011, a distribuição foi expandida com uma porta para ARMv7 de hard-float e foi renomeada para Arch Linux ARM para melhor englobar os objetivos e a direção do desenvolvimento. Uma terceira porta para ARMv6 de hard-float foi adicionada em 2012. Uma porta ARMv8 AArch64 foi adicionada em 2015.

O Arch Linux ARM continua a filosofia do Arch Linux em novas arquiteturas. É uma porta da distribuição Arch Linux x86 derivada. É uma distribuição Linux que mais suporta diferentes tipos de fabricantes de hardware. Os pacotes são lançados conforme as modificações feitas apenas para suportar a criação de ARM. E como qualquer outra distribuição Linux, é livre para baixar e usar, disponível em [archlinuxarm.org](http://archlinuxarm.org) ([archlinuxarm.org](http://archlinuxarm.org), 2018).

### 3.6 Lógica de programação

Lógica de programação é a técnica de encadear pensamentos para atingir determinado objetivo. Estes pensamentos, podem ser descritos como uma sequência de instruções, que devem ser seguidas para se cumprir uma determinada tarefa. A aprendizagem da lógica faz com que o pensamento proceda corretamente a fim de chegar a conhecimentos verdadeiros. Segundo Copi, (1978 apud Scolari et al), “O estudo da Lógica é o estudo dos métodos e princípios usados para distinguir o raciocínio correto do incorreto”.

As instruções são em sua maioria, um conjunto de procedimentos determinados em regras definidas por passos ou ordem de natureza crescente ou numérica para se realizar algo.

Uma ordem isolada só permite realizar um processo por vez, sendo assim para realizar todo um processo é necessário um conjunto de instruções colocadas em ordem de sequencias lógicas. Por exemplo, se quisermos comer batata frita, precisaremos seguir umas instruções: *descascar a batata, cortar a batata em pedaços, colocar óleo na frigideira, acender o fogo em que está a frigideira, e colocar os pedaços da batata cortada na frigideira para fritar.* Percebe-se então que essas instruções têm que ser executadas em uma ordem adequada e de alguma outra maneira antes de ascender o fogo, *pois não se pode descascar uma batata depois de fritá-la.*

É o que Piaget (1983) idealizou nas suas pesquisas, ressaltando que a lógica não é inata, ela se desenvolve gradualmente, delegando o conhecimento ao amadurecimento, em etapas organizadas e predeterminadas. Piaget (1983 apud Garcia, 1998, p.18)



Para Vygotsky (2006, p.60):

É uma zona de desenvolvimento proximal, pois refere-se ao caminho que o indivíduo vai percorrer para desenvolver funções que estão em processo de amadurecimento e que se tornarão funções consolidadas, estabelecidas no seu nível de desenvolvimento real. A zona de desenvolvimento proximal é, pois, um domínio psicológico em constante transformação: aquilo que uma criança é capaz de fazer com a ajuda de alguém hoje, ela conseguirá fazer sozinha amanhã (VYGOTSKY, 2006, p. 60).

Assim Piaget, enfatiza que as crianças desenvolvem a sua capacidade intelectual interagindo com objetos do ambiente onde ela vive e utilizando o seu mecanismo de aprendizagem. Isso acontece sem que a criança seja explicitamente ensinada (Valente, 1998, p.22).

a) Programação baseado em blocos, aprendizagem baseada em projetos

A programação baseada em blocos, é um projeto de nome **SCRATCH**, inspirado no “**LOGO**”, desenvolvido em 2007 no grupo Lifelong Kindergarten do MIT Media Lab pelo diretor Mitchel Resnich. O **SCRATCH** é uma plataforma de programação gratuita e online de construção de códigos de encaixe arrastar e soltar de maneira lógica inspirado no jogo de quebra cabeças. No lugar de linhas e scripts, os comandos são sequenciados por blocos coloridos feitos no programa. Quando montados de maneira correta, eles indicam ao computador o que fazer: movimentar um desenho de um gatinho, fazer uma janela surgir ao clique do mouse ou qualquer outra coisa que se imagine.

Segundo a revista INFO (2014), o **SCRATCH** é um programa para crianças programar sem precisar de códigos complicados. Mitchel Resnick defende que:

Noções de programação devem ser ensinadas às crianças desde cedo e que dominar tecnologia é tão importante quanto aprender a desenhar, escrever ou se comunicar, pois é uma forma de expressar a criatividade. E ao programar, as crianças aprendem a solucionar problemas, a combinar suas ideias e a planejar e estruturar projetos. Essas habilidades serão úteis não apenas para cientistas da computação, mas para qualquer pessoa, independentemente da idade, da experiência, do interesse ou da profissão que optar por seguir (INFO, 2014).

Iniciativas como a **SCRATCH** inspirou várias fundações e projetos em pró da programação por todo o mundo. É o que fez o iraniano formado em ciências da computação pela Universidade de Harvard, Hadi Partovi, que vendeu suas três empresas e tornou-se investidor. Ficou conhecido no vale do silício por investir em empresas de tecnologia como Facebook, Dropbox e Airbnb, que são umas das três empresas dos seus investimentos. E nas

companhias que investiu viu uma semelhança: a dificuldade de encontrar mão de obra qualificada para as vagas de desenvolvedores de software. Assim Hadi Partovi e seu irmão gêmeo, Ali Partovi em fevereiro de 2013, criaram a **Code.org**, uma organização sem fins lucrativos para promover o ensino de programação nas escolas dos Estados Unidos. E logo o projeto ganhou apoio de 60 pessoas, incluindo Bill Gates, Mark Zuckerberg, Jack Dorsey, o cantor Will.i.am e políticos como Al Gore e Michael Bloomberg. E ainda conseguiu o financiamento do Google, Microsoft, Amazon e LinkedIn. E foi lançado o vídeo do projeto e alcançou 11 milhões de visualizações. E a **Code.org** fez sua primeira iniciativa para que escolas, educadores e alunos em todo o mundo desenvolvessem 1 hora de programação entre os dias 09 e 15 de dezembro de 2013.

A organização reuniu na página online diversas atividades para jovens, entre elas jogos e tutoriais em ferramentas do tipo **SCRATCH**, aberta na semana com um vídeo do ex-presidente Barack Obama chamando os jovens para participar. E segundo a organização, 15 milhões de alunos de 170 países fizeram as aulas e entre eles, 115000 brasileiros. Hoje a organização já disponibiliza mais de 20 milhões de projetos em 50 idiomas.

A **Code Club**, fundado na Inglaterra por Clare Sutcliffe e Linda Sandvik em 2012, que atua também em um projeto mundial apoiando uma rede mundial de voluntários e educadores que conduzem clubes de programação gratuitos para crianças de 9 a 13 anos para que elas criem e compartilhem ideias, aprendendo ao longo do processo. E em 2015 o **Code Club** juntou forças com a Fundação Raspberry Pi contribuindo como uma parte importante do programa educacional da fundação colaborando para que mais crianças e jovens aprendam a criar projetos usando códigos, atingindo uma marca de 10.000 clubes criados em 125 países diferentes, com lições traduzidas em mais de 30 línguas diferentes. Em 2013 o **Code Club** começou a operar no Brasil (codeclub, 2018).

E seguindo essa ideia, várias outras fundações com esse propósito seguiram a essa tendência, como a **Microsoft Virtual Academy** (Microsoft, 2018), que disponibiliza treinamentos online em sua plataforma de sistemas Windows por profissionais da computação. Outra como a **Khan academy**, (khan academy, 2018) uma fundação sem fins lucrativos do empresário norte americano, Salman Khan e a **Codecademy** (codecademy, 2018), uma empresa privada norte americana fundada por Zach Sims e Ryan Bubinski, que disponibiliza cursos gratuitos.

Outras plataformas como, **Microsoft MakeCode** (Microsoft, 2018) e o **mBlock** (mblock, 2018), utiliza a ideia dos blocos do **SCRATCH**, voltados para a programação

aberta, para desenvolvimento de projetos com placas e sensores de baixo custo voltados para a educação.

O Google também disponibiliza um projeto de ciências da computação para educação, chamado de **Blockly**, uma linguagem de programação visual para o desenvolvimento de aplicações Android e IOS, baseado em blocos assim como no **SCRATCH**, mas voltado para usuários mais experientes. Com o **Blockly** construímos os códigos em blocos em qualquer plataforma com navegar web, além do android e IOS. Os códigos de blocos podem ser visualizados em até 5 linguagens de programação: JavaScript, Python, PHP, Lua e Dardo (developers google, 2018).

#### b) Ferramentas para aprendizagem e desenvolvimento da lógica de programação

##### b.1) Scratch

A plataforma **SCRATCH** foi desenvolvido pelo MIT Media Lab, com o intuito de ensinar programação para crianças em toda parte do mundo, criado por uma linguagem de fácil entendimento e assimilação que serviu de modelo e referência no assunto, é desenvolvida para funcionar em todos os Sistemas Operacionais em uso como: Windows, MAC OS, Linux, Android e IOS como editor off-line e online através dos navegadores com acesso à internet (MIT Media Lab, 2018).

##### b.2) Code.org

O Projeto **Code.Org**, com o objetivo de ensinar e despertar o interesse pela ciência da computação para alunos de todas as escolas do mundo e as pessoas de todas as idades, pela plataforma web, ou seja, apenas online, com cursos de ilustração animada e com instruções por vídeos em inglês com legendas antes do início dos cursos distribuídas por idades, sendo possível acessar e praticar as aulas em qualquer Sistema Operacional que tenha um navegador com acesso à internet (Code.org, 2018)

##### b.3) Microsoft virtual academy

A plataforma **Microsoft Virtual Academy**, disponibiliza conceitos de programação por profissionais da ciência da computação em vídeo aulas seguindo os princípios da hora do código, disponibilizados por área atuação como: Desenvolvimento: nuvem, Jogos, Web e Visual Studio; Profissional de TI: visualização, Gerenciamento de Identidade e Acesso, Automação de Nuvem Híbrida e DevOps; Profissional de Banco de Dados: administração de banco de dados e Business Intelligence. Cursos totalmente online podendo ser estudados em

qualquer Sistema Operacional com navegador web com acesso à internet (mva.microsoft.com, 2018).

#### b.4) Microsoft makecode

O **Microsoft MakeCode**, disponibiliza uma plataforma voltada para a programação aberta para programar placas como: Micro:bit, Circuit Playground Express, Chibi Chip e Grove Zero com 3 categorias de interface para a programação dos objetos: Um Simulador, para fornecer aos alunos um feedback, ou seja, um retorno de como ficará a programação feita pelo aluno na placa de baixo custo depois de programada. Um editor de blocos de arrastar e soltar com o mesmo objetivo de programação do **SCRATCH** e um editor Web em Java Script de programação JavaScript. A plataforma apenas direciona as vídeoaulas fornecidas pelos fabricantes das placas anunciadas na plataforma Microsoft MakeCode, mas para a programação com os editores é feita pela própria plataforma **Microsoft MakeCode**, funcionando em qualquer Sistema Operacional com navegador web com acesso à internet (makecode.com, 2018).

#### b.5) mblock

O **mBlock** também é uma plataforma voltada para a programação aberta para a programação de objetos robots como placas e sensores, e hardwares próprio Makeblock mBot baseados em arduíno, disponibilizando materiais animados e de montagem de kits robots em pdf e o ambiente gráfico de programação **mBlock** offline com base no **Scratch** Open Source Code que apresenta a mesma interface gráfica do **Scratch** para facilitar a programação de projetos Arduino e criação interativa para Sistemas Operacionais Windows, MAC OS e Linux. E um **APP mblock** offline para android e IOS que traz novas possibilidades para as crianças a aprender a programar, com o desenvolvimento de estratégias de resolução de problemas e desenvolvimento de habilidades de design e comunicação com blocos de construção com o intuito de criar pensadores criativos (mblock.cc, 2018).

#### b.6) Blockly

O **Blockly** da Google, voltado para o desenvolvimento mais juvenil com uma linguagem de programação visual de blocos disponibilizado em sua plataforma Google Developers, é uma biblioteca que adiciona um editor de código visual para a criação de aplicativos. O editor Blockly usa blocos gráficos interligados para representar conceitos de código como variáveis, expressões lógicas, loops e outros. Ele permite que usuários apliquem

princípios de programação sem ter que se preocupar com a sintaxe ou a intimidação de um cursor piscando na linha de comando e podendo ver ao mesmo tempo transformado o código de blocos em até 5 linguagens de programação: JavaScript, Python, PHP, Lua e Dardo. É totalmente Web, ou seja, funciona em qualquer Sistema Operacional com navegar web com acesso à internet (developers.google.com, 2018).

## 4 TRABALHOS RELACIONADOS

Este capítulo abordará os aspectos característicos comuns de relatos de experiência do ensino de lógica e pensamento computacional a alunos do ensino fundamental encontrados na literatura em comparação com a proposta deste trabalho. Os trabalhos relacionados para fins de comparação são enquadrados na mesma pesquisa envolvendo soluções de estudo/ensino e aprendizagem de lógica de programação a serem desenvolvidos a partir da educação básica.

Os trabalhos relacionados encontrados na literatura, complementam a necessidade do estudante de escola pública ter uma ferramenta de estudo, referindo-se a um computador, pela qual o mesmo consiga desenvolver os estudos propostos ao longo da jornada como um instrumento comparando-o ao um livro, caderno, lápis e borracha no percurso do dia-a-dia de rotina estudantil, pois agora o computador não é mais o instrumento que ensina o aprendiz, mas a ferramenta com a qual o aluno desenvolve algo em torno da lógica, e portanto, o aprendizado ocorre pelo fato de estar executando uma tarefa por intermédio do computador (VALENTE, 1998, p. 12).

Serão apresentadas algumas propostas de trabalhos relacionados referente ao tema desta pesquisa e no final serão destacados alguns pontos de suma importância entre os mesmos e a proposta deste trabalho.

### 4.1 Ensino de programação no ensino fundamental

Esta pesquisa mostra um relato de ensino de programação desenvolvidos nas turmas de 3º, 4º e 5º ano do ensino fundamental da Escola Santo Augusto, os conteúdos abordados, consistiram em atividades do livro de Computação Desplugada (BELL, WITTEN e FELLOWS, 2011) do qual usou como material de apoio seguindo as instruções como foco para aplicação das atividades desenvolvidas.

Segundo as autoras, quando apresentado a atividade de computação desplugada aos alunos, alguns mostraram-se decepcionados, mas fizeram apresentando dificuldades para entender por que estavam fazendo aquilo, e foi explicado o propósito das aulas a partir daquele dia que se tratava de trabalhar um conhecimento não vivenciado por eles fora da escola.

Lens e Cambraia (2015) fizeram uso das plataformas, *Code.org* e o *Scratch*. Com o *code.org* fez exercícios do labirinto e o desafio do artista, onde esses exercícios ocorrem com instruções de sequências de passos que o aluno tem que encaixar os comandos como:

AVANCE, VIRE À DIREITA, VIRE À ESQUERDA, REPITA, REPITA ATÉ, SE, SENÃO e etc. E assim cumprir de forma correta para completar a tarefa.

Alguns demoraram para desenvolver a atividade, mas com o tempo estavam resolvendo os desafios e comemorando por terem chegado a um determinado nível antes do colega, ou por terem descoberto qual era a sequência de comandos que resolvia o desafio. Confirmando que a programação é desafiadora, instiga o aluno a descobrir a solução para um determinado problema, o qual pode aumentar de complexidade de acordo com a evolução do aluno, fazendo com que ele consiga resolver problemas cada vez mais complexos. Foi relatado um exemplo de uma aluna com dificuldades de aprendizagem com a plataforma **Code.org** e na ferramenta **Scratch** ela conseguiu se sair melhor na aula, se desafiou a resolver os problemas, o que não acontecia anteriormente, foi observado na turma que ela não queria usar o computador e o seu interesse ocorreu pelo envolvimento dos alunos nas atividades propostas de forma coletiva.

Quando foi apresentado o **Scratch**, a maioria dos alunos entenderam a proposta da aula, pela semelhança dos comandos já utilizados nas atividades anteriores, exploraram o programa e desenvolveram animações com personagens coloridos e outros recursos que na situação, os alunos se sentiram mais animados com o uso do **Scratch**, pois foi proposto aos alunos um desenvolvimento de um jogo de boliche, e o objetivo de programar um personagem a desenvolver uma determinada ação foi atingido.

#### 4.2 Avaliando a Contribuição do Scratch para a Aprendizagem pela Solução de Problemas e o Desenvolvimento do Pensamento Criativo

A pesquisa dos autores, teve como objetivo de responder um problema interdisciplinar: Como e se discentes, dos anos finais do ensino fundamental e médio podem se apropriar de ambientes de programação visual para apoiar processos criativos de ensino-aprendizagem e de desenvolvimento do raciocínio lógico por meio da produção de seus próprios jogos. A pesquisa foi realizada em dez encontros semanais com duração de uma hora e meia cada, seguindo um cronograma elaborado pela pesquisadora durante os meses de outubro a dezembro de 2014.

Com o questionário, foi possível identificar o perfil dos participantes do estudo foram trinta crianças e adolescentes, com idades entre doze e dezessete anos regularmente matriculados no Ensino Regular. Dentre estes, 26 nos anos finais do Ensino Fundamental e 4 do Ensino Médio de 13 diferentes Escolas da rede pública de ensino de Araucária. A maioria

dos estudantes são do sexo masculino, totalizando 18, 11 do sexo feminino e 1 não informou. E a distribuição das turmas e turnos ficou da seguinte forma (Quadro 01):

**Quadro 01:** disposição dos grupos.

GRUPOS	TURNO	DIAS DA SEMANA	PARTICIPANTES
01	Manhã	Quarta-feira	11
02	Tarde		09
03	Tarde		05
04	Tarde	Quinta-feira	05

Fonte: Dados da pesquisa

Cada aluno utilizou um computador e a plataforma de estudo utilizada foi o Scratch, da qual exploraram e aprenderam a utilizar suas ferramentas tendo a liberdade de escolher o que realizar em seu projeto, como personagens, tema, background e etc.

A pesquisa teve como aporte teórico seguindo o paradigma construcionista de Papert (1986), a pesquisadora esteve na posição de mediadora em todo o processo. A avaliação da aprendizagem e do desenvolvimento do pensamento criativo e raciocínio lógico seguiram os quatro passos na aprendizagem pela solução de problemas de Pólya (2006):

- Compreender o problema
- Elaborar um plano;
- Executar o plano;
- Fazer o retrospecto ou a verificação do resultado. (PÓLYA. 2006. p.4)

Durante os dez encontros com duração de uma hora e meia cada, os sujeitos discentes foram estimulados a criarem seus próprios projetos. Inicialmente dois estudantes optaram em criar histórias interativas enquanto o restante dos grupos definiu a criação de jogos. Após o terceiro encontro, os sujeitos discentes que haviam escolhido fazer histórias interativas, desistiram do projeto e iniciaram outro, desta vez, jogos também. Quando questionados sobre a mudança, as repostas foram as seguintes:

“Ah, profi... sei lá, acho que vai “sê” mais legal se eu “fizé” um jogo porque daí dá pra “jogá”, e não “ficá” só olhando. (S02, 22/10/2014) “Comecei a num “achá” mais legal e pequei e perguntei se “pudia mudá”, a “Profi” disse que sim, e que o projeto é meu e se eu posso “escolhê”, então escolhi. Quero fazer um jogo pra “jogá” em casa com meus “primo”.”(S18, 22/10/2014)

O objetivo principal da pesquisa era responder à pergunta: Como e se os grupos de discentes pesquisados podem desenvolver o raciocínio lógico e o pensamento criativo por meio de suas próprias produções e que estes sujeitos tivessem liberdade de escolher, criar, pensar, compartilhar.



Quanto à aprendizagem e a capacidade de identificar e solucionar os problemas seguindo os passos de Pólya (2006), foi necessário deixar claro para o estudante as quatro etapas e a importância das mesmas, discorremos sobre cada uma delas, pois antes de começar a resolver o problema, é preciso compreendê-lo, ter um objetivo delimitado.

No caso do Sujeito 22 (S22), questionamentos foram feitos pela pesquisadora e obteve-se as seguintes respostas:

Etapa 1 – compreender o problema. (Qual o problema?) “Eu quero que a minhoca siga a flexa que eu “apertá” no computador. “Mais num” consigo. (S22, 22/10/2014)

Etapa 2 – elaborar um plano. (Como você acha que podemos resolver?)

“Ai Profi, num” sei. Será que se eu “procurá” nos “bloco” eu acho alguma coisa pra “mi ajudá” (S22, 22/10/2014) “Eu acho que tem um negócio ali que eu vi nos “bloco azú”... “ah, mais num” sei se vai “funcioná”(S24, 22/10/2014) “Que qui cê” viu?Pra “fazê”o quê? (S18, 22/10/2014) “Esse aqui, o bloco “azú”. Olha é pra “apontá” pra direção e tem essa flecha. Usa esse! (S24, 22/10/2014)

E de acordo com os textos (22/10/2014), acima foi identificado que os alunos, independente do grupo, horário ou problema, em volta, deixavam seus projetos, ficavam em conjunto analisando para tentar resolver o problema.

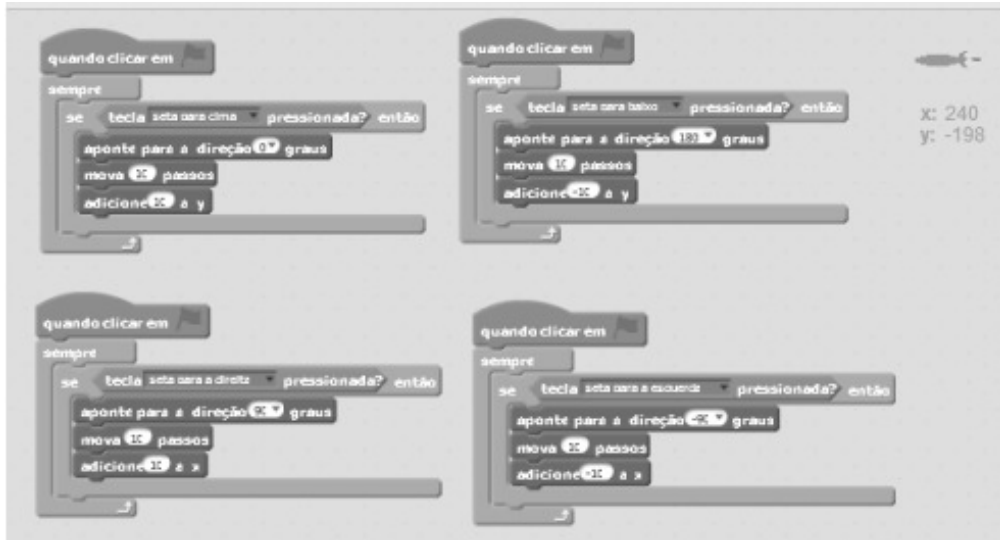
Etapa 3 – Executar o plano. (Vamos tentar?)

“Ih, mais eu vô tê que colocá, um bloco pra cada comando? (S22, 22/10/2014) Acho “qui sim”. “Sinão” vai só pra lá ele “virá” (S24, 22/10/2014) É pode sê. (S22, 22/10/2014) “Coloca aí e testa, aperta o botão verde. (S24, 22/10/2014)

A configuração sugerida pelo S24 funcionou. Um terceiro bloco de movimento foi adicionado. O grupo demonstrou ficar satisfeito e, em outros projetos, pode-se encontrar o mesmo bloco de comandos para executar os movimentos guiados pelo teclado, direcionando o personagem para apontar para a direção.

Etapa 4 – fazer o retrospecto. (Vamos ver se funciona?)

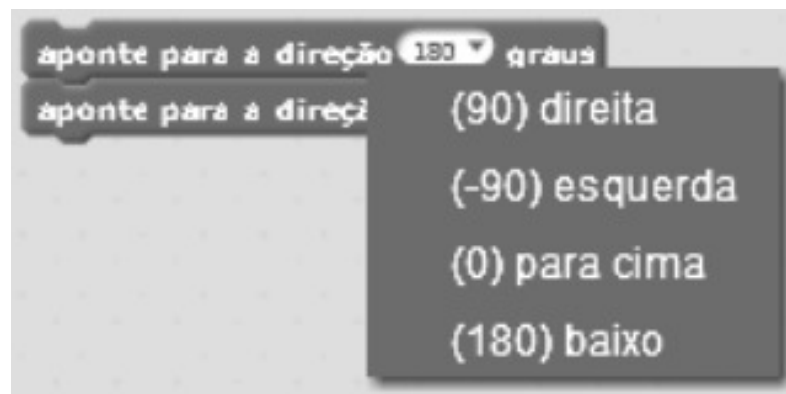
Após a colaboração de mais dois colegas, a quarta etapa sugerida por Pólya (2006), fazer o retrospecto e a verificação do resultado foi feita pelo grupo. Obteve-se a seguinte resposta como ilustra a figura 19 abaixo da pesquisa:



**Figura 19:** Blocos de movimento no projeto do aluno 22.

Fonte: Dados da pesquisa

Sem a interferência da Pesquisadora observado ainda, em conjunto, que no bloco de comando havia opções para apontar para a direção desejada. Seguiram a orientação, conforme a Figura 19 da pesquisa:



**Figura 20:** Opções para o direcionamento do personagem no projeto do S22.

Fonte: Dados da pesquisa

A pesquisa desvendou que o pensamento criativo é um processo mental que utiliza a criatividade na solução eficaz de problemas e pode ser estimulado e desenvolvido. A maneira pela qual os sujeitos discentes estão sendo criados e educados, bem como o ambiente estimulador são elementos básicos para seu desenvolvimento. Se este estudante convive em um ambiente estimulador e é incitado constantemente a desbravar novos conhecimentos e ou encontrar estratégias próprias para soluções de situações problema do cotidiano, este sujeito possui estruturas psíquicas de pensamento desenvolvidas. E esses resultados vão ao encontro dos conceitos: construcionismo como um paradigma educacional, o docente como mediador no processo para o desenvolvimento de funções psicológicas superiores como o pensamento

criativo e o raciocínio lógico e a aprendizagem pela solução de problemas presentes nos estudos de Papert (1986, 1994), Piaget (1996), Pólya (2006) e Pozo (1998).

O estudo apresentado na pesquisa das autoras Bressan & Amaral (2015), ocorreu na sala de informática da Escola Complexo Pedagógico Lucy Moreira Machado da cidade de Araucária e teve por objetivo demonstrar as potencialidades do ambiente de programação visual Scratch, na disseminação do Pensamento Computacional na Educação Básica. Os resultados apontam que os estudantes aprenderam diversos conceitos de Ciência da Computação. Por meio dos projetos desenvolvidos, eles puderam demonstrar competência em *sequência, evento, paralelismo, loop, condicionais, operadores e dados, sendo o paralelismo* o conceito menos compreendido e *sequência* e *evento* os que apresentaram melhores resultados de aprendizagem.

#### 4.3 Uma Proposta para o Ensino de Programação de Computadores na Educação Básica

A pesquisa proposta por Leite (2015) propõe uma aula com as ferramentas da linguagem de programação na plataforma online do **Code.org**, que tem semelhança com o Scratch, pela forma de programação em blocos de encaixe, e por possuir uma interface gráfica amigável e por ser uma das mais difundidas no ambiente educativo mundial.

A autora citou que não há uma matéria específica para aplicação qualquer, sendo que qualquer docente que tenha interesse pode aplicar atividades com a plataforma em trabalhos multidisciplinar, explicando por exemplo que o professor de artes trabalha com cores e os desenhos, o de língua portuguesa trata da escrita correta dos comandos e o professor de matemática explica a lógica do jogo.

O procedimento para iniciar esse a plataforma é descrita com os seguintes passos:

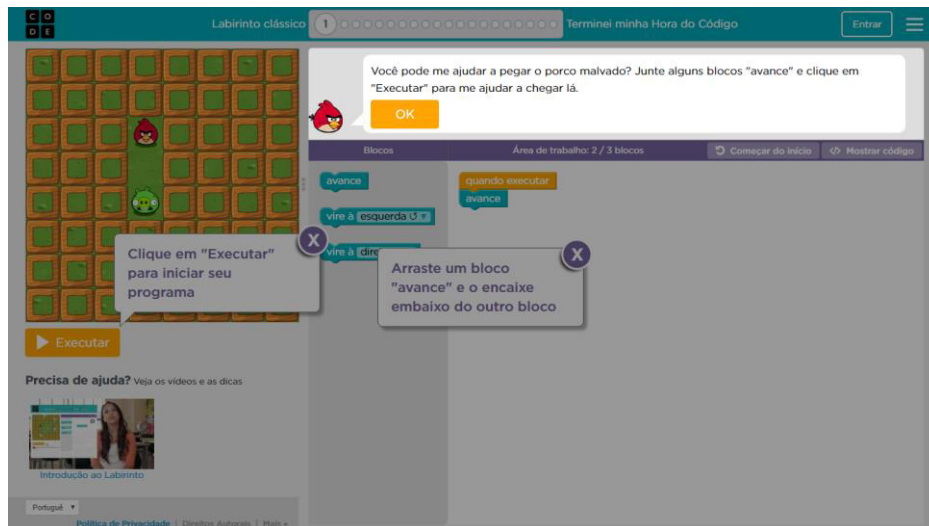
- O professor deve dispor de uma sala com computadores conectados à internet. Preferencialmente, no máximo dois alunos por computador;
- Todos devem acessar a página da plataforma **code.org**: **<<http://studio.code.org/>>**.

A atividade escolhida foi o labirinto com os personagens do jogo “Angry Birds”, um jogo muito popular para (Android e IOS).

No início da atividade para executar o jogo, é exibido um vídeo introdutório com depoimentos de pessoas influentes, que apoiam a inserção da computação nas escolas, e a explicação de uma programadora detalhadamente ilustra o funcionamento das ferramentas na

montagem dos blocos para se concluir uma tarefa, orientando o aluno e professor quanto à prática do trabalho.

Logo após o vídeo, abrirá a tela que propõe os desafios com a montagem dos blocos lógicos em até 20 formas diferentes de raciocínio para se desenvolver a atividade (Figura 21) da pesquisa, e as dicas ficam disponíveis a todo momento que poderá ser visualizada através de vídeos como ajuda no canto esquerdo inferior da tela.



**Figura 21:** Tela da atividade.

Fonte: code.org

Os desafios são sempre motivadores e instigam a busca por soluções, e o professor pode explorar a criatividade e particularidade de cada aluno. A pesquisa teve como suporte apenas as instruções de texto e vídeos que disponibiliza antes do início de qualquer atividade exercício repetitivos na própria plataforma a serem feitos para entender o raciocínio do jogo.

#### 4.4 Considerações Sobre os Trabalhos Relacionados

Com relação aos relatos de estudo ensino de lógica de programação para a educação básica fundamental, diversos trabalhos foram encontrados, porém a maioria deles descreve ações isoladas e de curta duração. Foram apresentados nesta pesquisa 3 trabalhos para fins de comparação pelo problema/área que estão mais próximos da presente pesquisa. No entanto destacamos alguns pontos considerados importantes para o desenvolvimento desta pesquisa em termos de comparação, pontos esses, que não foram levados em consideração quanto a estrutura e material de estudo/metodologia de apoio para professores e alunos em sala de aula:

- Sala ou laboratório de informática: não se sabe se a quantidade de computadores é suficiente para uma turma de alunos;

- Computador como ferramenta de estudo: não foi informado em nenhuma pesquisa se os alunos dispõem de um computador em casa para estudo das atividades desenvolvidas na escola;
- Acesso à internet: nem todas as escolas conseguem disponibilizar desse serviço e em total funcionamento, principalmente as escolas do interior dos estados e zona rural das cidades;
- Material de apoio/metodologia: nenhuma pesquisa citou uma solução quanto ao preparo dos professores na busca de cursos na área de lógica de programação para desenvolver com os alunos em sala de aula.

## 5 CONSIDERAÇÕES FINAIS

Podemos analisar então que esta pesquisa mostra uma pequena contribuição de como é possível montar um computador chamado de ferramenta de estudo com poucas condições financeiras, utilizando placas open source, dita como uma tecnologia de baixo custo e considerada sustentável e ao mesmo tempo inovadora, pois torna acessível a todas as pessoas que se interessam pela tecnologia a ser explorada e customizada desde os padrões de comodismos à suas necessidades e para uso na Internet das Coisas.

Novas de formas de pensar, de agir e de comunicar-se são introduzidas como hábitos corriqueiros, são inúmeras as formas de adquirir conhecimento, bem como também são diversas as ferramentas que propiciam essa aquisição, as escolas são em geral apontadas como uma das principais alternativas para formação e desenvolvimento de cidadãos que prosperam de um perfil que conduza com as exigências da sociedade moderna.

Ao longo de toda a fundamentação teórica, mostramos as alternativas para se propor como ferramenta a ser explorada, a fim de que seja possível atingir o propósito do estudo de programação a serem introduzidos na educação pelos recursos das placas de baixo custo open source e os softwares ou plataformas de código aberto, através de um computador de placa única Raspberry Pi, um cartão de memória, uma fonte de alimentação de 5V, podendo ser montado pelo aluno com auxílio de um professor ou tutor. Dessa forma as verdadeiras inovações devem possuir características importantes que levem os gestores dos sistemas educacionais a pensarem e planejarem estratégias que durem a longo prazo, pois a fuga da rotina e da massificação de respostas prontas fazem com que alunos não sejam mais passivos de seguir modelos, fazendo com que se tornem indivíduos atuantes, participativos e interativos.

O principal foco deste trabalho é o desenvolvimento da lógica por meio de um computador de placa única capaz de sanar as necessidades do aluno e do professor que por sua vez poderá usar com o **DATA SHOW** em exposições de aula e em projetos educacionais como experimentos de ciências da computação estudados pela colaboração de ambos, em um cenário que figura como a nova ordem de se impor a tecnologia e a ela como auxílio para se construir conhecimento, pois modelos semelhantes a este já estão em pleno vapor em outras nações como por exemplo o Reino Unido, sede da Fundação Raspberry Pi e do Micro: bit, que buscam no currículo escolar a lógica de programação como disciplina obrigatória.

A pesquisa encontra-se ainda em desenvolvimento, pois não foi possível realizar aplicação de campo escolar devido ao encerramento do ano letivo das escolas, mas a proposta

se apresenta como uma alternativa interessante e acessível para o ensino e aprendizagem de lógica de programação, preenchendo alguns pontos negativos que encontramos nos demais trabalhos relacionados.

Desta forma, pretende-se como trabalhos futuros fazer uma aplicação desta pesquisa em campo escolar para possibilitar a ampliação do conhecimento do campo de pesquisa de placas IoT de baixo custo, com foco na programação dos objetos, e o desenvolvimento da aprendizagem da lógica de programação, como suporte de ideia de aprender e dominar uma das linguagens de programação visual em blocos.

## REFERÊNCIAS

ABRÃO, R. K; Adamatti, D. F. **As novas tecnologias da informação e comunicação e a atividade experimental no ensino de ciências.** *Revista Linhas*. Florianópolis, v. 16, n. 31, p. 305 – 324, maio/ago. 2015.

**A Educação e Novas Tecnologias Digitais – Inovação na Produção do Conhecimento.** Revista Fonte, Ano 5, Número 8, dezembro de 2008.

ALMEIDA, M. E. B, **Informática e Formação de Professores.** Ministério da Educação. 2000. 122p. Disponível em <<http://www.dominiopublico.gov.br/download/texto/me003148.pdf>> Acesso em 23 de junho de 2012.

ARM, (2018), Site Institucional da ARM. Disponível em < <http://www.arm.com>>. Acessado em 12 de outubro de 2018

ASUS, (2018). Site Institucional da ASUS. Disponível em < <https://www.asus.com>>. Acessado em 12 de janeiro de 2018

ASHTON, K. **That “Internet of Things” Thing.** RFID Journal, 22 jun. 2009. Disponível em: <<http://www.rfidjournal.com/articles/view?4986>>. Acesso em: 10 de novembro de 2017.

BLIKSTEIN, P; Zuffo, K. M. **As Sereias do Ensino eletrônico.** Escola Politécnica – USP (2001). Disponível em: <<http://www.scielo.br/scielo.php?pid=S010039842014000200012&script=sciarttext&tlnt>>. Acessado em 08 de setembro de 2017.

BRESSAN, M. L. Q. AMARAL, M. A. **Avaliando a Contribuição do Scratch para a Aprendizagem pela Solução de Problemas e o Desenvolvimento do Pensamento Criativo.** Revista Intersaberes, Vol.10, n.21, p.509-526, setembro de 2015.

BLOKCLY (2018), Site Institucional da Goole Developers. Disponível em <<https://developers.google.com/blockly/>>. Acessado em 07 de janeiro de 2018.

CHRISTENSEN, C. M. et al, **Ensino Híbrido: uma Inovação Disruptiva? Uma introdução à teoria dos híbridos,** Clayton Christensen Institute, maio. 2013. <[https://s3.amazonaws.com/porvir/wp-content/uploads/2014/08/PT\\_Is-K-12-blended-learning-disruptive-Final.pdf](https://s3.amazonaws.com/porvir/wp-content/uploads/2014/08/PT_Is-K-12-blended-learning-disruptive-Final.pdf)> Acesso em 18 de novembro de 2017.

CISCO (2011), Site Institucional da Cisco, **Visão geral do TCP/IP,** Documento: 13769, 10 de agosto de 2005. Disponível em: <[https://www.cisco.com/c/pt\\_br/support/docs/ip/routing-information-protocol-rip/13769-5.html#tcpiptech](https://www.cisco.com/c/pt_br/support/docs/ip/routing-information-protocol-rip/13769-5.html#tcpiptech)>. Acesso em: 12 de outubro de 2017.

**Como Formar Gênios: crianças que aprendem hoje a programar serão os visionários de amanhã.** Revista INFO, Editora Abril. Ed 338, de fevereiro de 2014.

CODE.ORG (2018), Site Institucional da Code.org. Disponível em <<https://code.org/>>. Acessado em 06 de janeiro de 2018



CODE CLUB BRASIL (2018), Site Institucional da Code club Brasil. Disponível em <<https://www.codeclubbrasil.org.br/>>. Acessado em 06 de janeiro de 2018.

EDYN (2017). Site Institucional da Edyn. Sensores Edyn. Disponível em <<https://edyn.com/>> Acesso em 15 de dezembro de 2017

FITFIB (2017). Site Institucional da FitFib. **Pulseiras FitFib**, Disponível em <<https://www.fitbit.com>> Acesso em 15 de dezembro de 2017.

GARTNER (2017), Site Institucional da Gartner, **Identifies the Top 10 Strategic Technology Trends for 2013**. Mar. 2013. Disponível em <<http://www.gartner.com/it/page.jsp?id=2209615>>. Acesso em: 15 outubro de 2017.

GOGOBOARD (2018), Site Institucional do **Gogoboard**, Disponível em <<https://gogoboard.org>>. Acessado em 05 de janeiro de 2018.

INFOPÉDIA (2017), Site Institucional da **Infopédia Dicionários Porto Editora**, Disponível em <<https://www.infopedia.pt/>>. Acessado em 07 de novembro de 2017.

JORNAL O DIA (2017). Site Institucional do Jornal o Dia. **Entenda o que é “disrupção” e saiba como ela ameaça empresas**, 2015. Disponível em: <<http://odia.ig.com.br/noticia/economia/2015-06-28/entenda-o-que-e-disrupcao-e-saiba-como-ele-ameaca-empresas.html>> Acesso em: 18 de dezembro de 2017.

KHAN ACADEMY (2018), Site Institucional da Khan Academy. Disponível em <<https://pt.khanacademy.org/>>. Acessado em 06 de janeiro de 2018.

KRANENBURG, R. **The Sensing Planet: Why the internet of things is the biggest next big thing**. 2012. Disponível em <<https://www.fastcompany.com/1681563/the-sensing-planet-why-the-internet-of-things-is-the-biggest-next-big-thing>> Acesso em 19 de dezembro de 2017.

LACERDA, F. (2015). **Arquitetura da informação pervasiva: projetos de ecossistemas de informação na internet das coisas** (Tese de Doutorado em Ciência da Informação, Universidade de Brasília, Brasília). Recuperado de <http://repositorio.unb.br/handle/10482/19646>

LEMOS, A. **A Comunicação das Coisas: Internet das Coisas e Teoria Ator-Rede. Etiquetas de Radiofrequência em Uniformes Escolares na Bahia, Salvador – Ba**, outubro de 2012. Disponível em: <[http://roitier.pro.br/wp-content/uploads/2017/09/Andre\\_Lemos.pdf](http://roitier.pro.br/wp-content/uploads/2017/09/Andre_Lemos.pdf)> Acesso em: 20 de dezembro de 2017.

LEMOS, A. **A Comunicação das Coisas: teoria ator-rede e Cibercultura**. São Paulo Annablume, 310 p, 2013

LOPEZ RESEARCH. Site Institucional da Lopes Research, **“Uma Introdução à Internet das Coisas – (IoT) ”**, novembro de 2013. Disponível em: <[https://www.cisco.com/c/dam/global/pt\\_br/assets/brand/iot/iot/pdfs/lopez\\_research\\_an\\_introduction\\_to\\_iiot\\_102413\\_final\\_portuguese.pdf](https://www.cisco.com/c/dam/global/pt_br/assets/brand/iot/iot/pdfs/lopez_research_an_introduction_to_iiot_102413_final_portuguese.pdf)> Acesso em: 20 de novembro de 2017.

LÉVY, P. **Cibercultura**. São Paulo: Editora 34, 1999.

LENS, R. K. CAMBRAIA. A. C. **Ensino de Programação no Ensino Fundamental**. Santa Maria-RS, maio de 2015.

LEITE. R. M. **Uma Proposta para o Ensino de Programação de Computadores na Educação Básica**. Trabalho de Conclusão de Curso (especialização), Centro Interdisciplinar de Novas Tecnologias – UFRS - Porto Alegre-RS, 2015.

MICRO:BIT, (2017), Site Institucional do **Micro:bit**. Disponível em <<http://microbit.org>>. Acessado em 06 de dezembro de 2017.

MORAES, P. S, **Curso básico de Lógica de Programação**. UNICAMP - Centro de Computação. Abril, 2000, 45p. Disponível em <<http://www.inf.ufsc.br/~vania.bogorny/teaching/ine5231/Logica.pdf>>. Acessado em 2 de janeiro de 2018.

MICROSOFT MAKECODE (2018), Site Institucional da Microsoft Makecode. Disponível em <<https://makecode.com/>>. Acessado em 06 de janeiro de 2018.

MICROSOFT VIRTUAL ACADEMY (2018), Site Institucional da Microsoft. Disponível em <<https://mva.microsoft.com/>>. Acessado em 06 de janeiro de 2018.

MBLOCK (2018), Site Institucional da mBlock. Disponível em <<http://www.mblock.cc/>>. Acessado em 06 de janeiro de 2018.

MIT (2018), Scratch 3.0. Disponível em <<https://www.media.mit.edu/projects/scratch-3-0/overview/>>. Acessado em 10 de janeiro de 2018.

NEST (2017). Site Institucional da Nest. **Termostato Nest**. Disponível em <[nest.com](http://nest.com)> Acesso em 15 de dezembro de 2017.

ORANGE PI, (2018), Site institucional do **Orange**”, Disponível em: <<http://www.orangepi.org>>. Acessado em: 09 de janeiro de 2018.

ODROID, (2018), Site institucional da ODROID. Disponível em <<http://www.hardkernel.com>>. Acessado em 20 de outubro de 2017.

OLIVEIRA, Marta Kohl de. **Vygotsky: aprendizado e desenvolvimento: um processo sócio-histórico**. São Paulo: Scipione, 2006. (Coleção Pensamento e ação na sala de aula).

PAPERT, Seymour. Logo: computadores e educação. 2ª Edição. Tradução de José Armando Valente, Beatriz Bitelman e Afira Vianna Ripper. São Paulo: Brasiliense. 1986.

PAPERT, Seymour. A Máquina das Crianças: repensando a escola na era da Informática; tradução Sandra Costa. -ed. rev. - Porto Alegre; Artmed. 1994.

PARTOVI, H, **Sobre**. 2018. Disponível em <[https://code.org/about/leadership/hadi\\_partovi](https://code.org/about/leadership/hadi_partovi)>. Acessado em 06 de janeiro de 2018.

PIAGET, Jean. **Biologia e Conhecimento**. 2ª Edição. Vozes: Petrópolis, 1996.

PHILIPS. Site Institucional da Philips. **Lâmpadas de Led Wireless Philips Hue**. 2017. Disponível em <<https://www2.meethue.com/pt-br>> Acesso em 15 de dezembro de 2017.

POLYA, George. A arte de resolver problemas. Tradução de Heitor Lisboa de Araújo. Rio de Janeiro: Interciência. 2006.

POZO, Juan Ignacio. Teorias cognitivas da aprendizagem. 3ª ed, Porto Alegre: Artes Médicas. 1998.

PORVIR (2018), Site Institucional do Porvir. **Personalização da Educação**. Disponível em <<http://porvir.org/especiais/personalizacao/>>. Acessado em 01 de janeiro de 2018

RASPBERRY PI (2018), Site institucional do **Raspberry**”, Disponível em: <<http://www.raspberrypi.org>>. Acessado em: 10 de janeiro de 2018.

THE ECONOMIST (2017), Site Institucional do The Economist. “*Augmented Business*”, novembro de 2010. Disponível em <<http://www.economist.com/node/17388392>> Acesso em 20 de dezembro de 2017.

VALENTE, J. A. **Computadores e Conhecimento: Repensando a Educação**, 2ª Ed. Campinas – SP, UNICAMP/NIED, 1998, 501p.

W3C (2004), Site Institucional da W3C, **Architecture of the World Wide Web**, Volume One, 15 December 2004. < <https://www.w3.org/TR/webarch>>. Acesso em: 12 de outubro de 2017.