

## O uso da ferramenta Tinkercad e da linguagem Scratch para o ensino dos fundamentos da programação em Internet das Coisas

The use of Thinkercad tool and Scratch language to teach the programming fundamentals in Internet of Things

El uso de la herramienta Tinkercad y el lenguaje Scratch para enseñar los fundamentos de la programación en Internet de las Cosas

Recebido: 21/10/2021 | Revisado: 29/10/2021 | Aceito: 05/11/2021 | Publicado: 08/11/2021

**Cleber Gustavo Dias**

ORCID: <https://orcid.org/0000-0002-4232-2409>  
Universidade Nove de Julho, Brasil  
E-mail: [diascg2019@gmail.com](mailto:diascg2019@gmail.com)

**Ingrid Santella Evaristo**

ORCID: <https://orcid.org/0000-0002-1115-3622>  
Universidade Nove de Julho, Brasil  
E-mail: [isantella@hotmail.com](mailto:isantella@hotmail.com)

**Agenor Roris Filho**

ORCID: <https://orcid.org/0000-0001-8100-4916>  
Universidade Nove de Julho, Brasil  
E-mail: [agenor.roris@uni9.edu.br](mailto:agenor.roris@uni9.edu.br)

**Adriana Aparecida de Lima Terçariol**

ORCID: <https://orcid.org/0000-0002-5824-2294>  
Universidade Nove de Julho, Brasil  
E-mail: [atercarior@gmail.com](mailto:atercarior@gmail.com)

### Resumo

Nos últimos anos a Internet das Coisas (IoT) vem sendo empregada em diferentes campos do conhecimento, como a área industrial, a área da saúde, em logística, entre muitas outras. A enorme conectividade entre os dispositivos e as pessoas está gerando novas soluções para uma série de problemas do cotidiano. Sabe-se que muitos desses dispositivos são formados, além dos sistemas microcontrolados, por sensores e/ou atuadores capazes de efetuar a leitura de grandezas do meio externo ou ainda provocar mudanças controladas, a fim de gerar movimentos e ações desejadas neste mesmo meio. Neste sentido, o presente estudo avaliou a eficácia da ferramenta Tinkercad, combinada com a linguagem Scratch, com a finalidade de ensinar aos alunos dos cursos de bacharelado em Ciência da Computação, Sistemas de Informação e Tecnologias em Informática, os principais fundamentos de programação associados ao uso de sensores em aplicações comumente encontradas na área de Internet das Coisas. Ao final das aulas, um questionário elaborado com base na escala Likert foi usado para avaliar diferentes aspectos do entendimento dos alunos acerca do uso do Tinkercad e da linguagem Scratch nos exercícios desenvolvidos. Um total de 174 alunos respondeu ao questionário, sendo possível constatar que a referida ferramenta e a linguagem Scratch oferecem de fato um ambiente computacional bastante favorável não apenas ao estudo da lógica computacional, mas também para uma melhor compreensão do funcionamento de alguns tipos de sensores aplicados em automação e especialmente no campo da Internet das Coisas.

**Palavras-chave:** Tinkercad; Linguagem Scratch; Internet das coisas; Fundamentos da programação.

### Abstract

Over the last years, the Internet of Things (IoT) has been used in different fields of knowledge, such as industrial area, a health area and logistics, among many others. The enormous connectivity between devices and people is generating new solutions to a number of everyday problems. It is well known that, many devices are built, in addition to micro controlled systems, by sensors and/or actuators capable of reading external signals of the environment or even causing controlled changes, in order to generate movements and desired actions in this same environment. In this sense, the present study has evaluated the effectiveness of the Tinkercad tool, combined with the Scratch language, to teach students of bachelor's courses of Computer Science, Information Systems and Technologies in Informatics, the main programming fundamentals associated with use of sensors in applications commonly found in the Internet of Things area. At the end of the classes, a questionnaire based on the Likert scale was used to assess different aspects of the student's understanding of the use of Tinkercad and Scratch language in the developed exercises. A total of 174

students answered the questionnaire, and it was possible to verify that the aforementioned tool and the Scratch language actually offer a very favorable computational environment not only for the study of computational logic, but also for a better understanding of the functioning of some types of sensors normally applied in automation and particularly in the field of Internet of Things.

**Keywords:** Tinkercad; Scratch language; Internet of things; Programming basics.

### Resumen

En los últimos años, el Internet de las Cosas (IoT) se ha utilizado en diferentes campos del conocimiento, como el área industrial, el área de la salud, la logística, entre muchos otros. La enorme conectividad entre dispositivos y personas está generando nuevas soluciones a una serie de problemas cotidianos. Se sabe que muchos de estos dispositivos están formados, además de sistemas micro controlados, por sensores y / o actuadores capaces de leer cantidades del entorno externo o incluso provocar cambios controlados, con el fin de generar movimientos y acciones deseadas en este mismo entorno. En este sentido, el presente estudio evaluó la efectividad de la herramienta Tinkercad, combinada con el lenguaje Scratch, para enseñar a los estudiantes de las carreras de grado en Informática, Sistemas de Información y Tecnologías en Informática, los principales fundamentos de programación asociados al uso de sensores en aplicaciones que se encuentran comúnmente en el área de Internet de las cosas. Al finalizar las clases, se utilizó un cuestionario basado en la escala Likert para evaluar diferentes aspectos de la comprensión de los estudiantes sobre el uso del Tinkercad y lenguaje de Scratch en los ejercicios desarrollados. Un total de 174 alumnos respondieron al cuestionario, y es posible comprobar que la mencionada herramienta y el lenguaje Scratch realmente ofrecen un entorno computacional muy favorable no solo para el estudio de la lógica computacional, pero también para una mejor comprensión del funcionamiento de algunos tipos de sensores aplicados en la automatización y especialmente en el campo del Internet de las Cosas.

**Palabras clave:** Tinkercad; Lenguaje de Scratch; Internet de las cosas; Fundamentos de programación.

## 1. Introdução

A chamada quarta revolução industrial está provocando mudanças jamais experimentadas pela humanidade em um passado recente. O desenvolvimento de novas tecnologias e a combinação de soluções alternativas serão decisivas para o futuro do emprego e da sociedade como um todo nas próximas décadas. Uma das áreas na qual o avanço tecnológico acontece de forma exponencial é a denominada Internet das Coisas (Internet of Things, sigla IoT em inglês), capaz de conectar milhões, ou bilhões de dispositivos, com a Internet ao redor do mundo. Segundo o Fórum Econômico Mundial, há uma estimativa de que nesta década mais de 80 bilhões de dispositivos estejam conectados entre si e com as pessoas (Schwab & Davis, 2018).

A Internet das Coisas pode ser definida como uma área do conhecimento na qual dispositivos eletrônicos são capazes de interagir com os usuários e outros dispositivos em uma infraestrutura de rede, com interação do usuário limitada ou inexistente (Mena, et al., 2018). Vale destacar que a Internet das Coisas emprega uma grande quantidade de dispositivos sensores e atuadores, por exemplo, para realizar respectivamente a medição de grandezas físicas do meio externo e também para promover ações desejadas no ambiente (Chowdhury, et. al, 2021; Kinjo, et. al., 2021).

Diante deste cenário, torna-se ainda mais premente o uso de ferramentas capazes de aprimorar o processo de ensino e aprendizagem das novas tecnologias voltadas para o universo da Internet das Coisas, a fim de que os estudantes desenvolvam as habilidades e competências necessárias para a sua inserção em um mercado de trabalho em constante transformação. Mais especialmente, os alunos oriundos dos cursos de tecnologia na área da Informática, ou das Engenharias, devem compreender com propriedade os fenômenos e efeitos que cercam tais desenvolvimentos e as consequências do seu uso em diferentes problemas do cotidiano.

Neste sentido, muitos pesquisadores e profissionais tem investido esforços na criação de novos ambientes computacionais voltados para o ensino de conceitos e fundamentos particularmente relacionados aos campos da ciência, tecnologia, engenharia e matemática (Science, Technology, Engineering and Mathematics, sigla STEM em inglês), ou na aplicação de ferramentas já existentes para o mesmo fim (Chabanet, et. al., 2021; Corneta, et. al., 2020; Abburi, et. al., 2021).

Assim, alguns trabalhos abarcaram o desenvolvimento de ambientes virtuais com a finalidade de oferecer um melhor entendimento dos aludidos dispositivos, uma vez que nem sempre é possível a implementação de laboratórios físicos que

possam acomodar os diversos tipos de equipamentos (Magnus, et. al., 2020). Ademais, a pandemia relacionada ao COVID-19 fomentou ainda mais a criação de alternativas capazes de promover um ensino remoto com mais amplitude e qualidade (Gruenewald, et. al., 2021).

Como exemplo, o trabalho publicado por Dias & Silva (2021) oferece uma abordagem virtual capaz de simular um motor elétrico em corrente contínua usando apenas o software Excel. Em outro estudo, Dias et. al. (2020) utilizou o software Scilab, disponível na nuvem, para oferecer aos alunos uma melhor compreensão da área de processamento digital de sinais. De outro modo, o artigo de Fukumoto et. al. (2021) explora o uso de laboratórios remotos, também com a finalidade de entender atuadores elétricos, especialmente os motores de passo. No entanto, cabe frisar que, no caso dos espaços remotos é necessário que todos os equipamentos estejam disponíveis para a realização dos ensaios ou experimentos programados.

Uma das ferramentas computacionais que está sendo bastante empregada recentemente por educadores, em diferentes níveis, para o ensino de habilidades e competências nas áreas tecnológicas, é a Tinkercad® desenvolvida pela empresa Autodesk (Paul, et.al., 2018). A Tinkercad® está disponível na nuvem e pode ser utilizada de forma gratuita por alunos e professores a partir de um computador ou mesmo um celular conectado à rede Internet.

Adicionalmente, o estudo de Díaz et. al. (2019) destacou o uso da linguagem Scratch, disponível no Tinkercad®, para a criação e edição de modelos em três dimensões em um curso de impressão 3D. Os autores concluíram que houve interesse por parte dos estudantes especialmente em relação ao uso da linguagem Scratch nas tarefas de programação.

No ensino superior em Ciência da Computação, ou cursos da área da Informática e ainda Engenharias, um dos maiores desafios para os estudantes é o aprender a programar. Além disso, essa é uma das habilidades essenciais a ser desenvolvida nesse e em outros cursos nos quais há uma grande demanda por novas tecnologias digitais. No entanto, o aprendizado de programação tem sido difícil para os estudantes ingressantes, o que se evidencia com “altas taxas de reprovação, elevados índices de desistências e, até mesmo, de evasão dos cursos comprovam tamanha complexidade por trás do processo de ensino e aprendizagem das competências necessárias para se tornar um programador” (Oliveira; Pereira, 2019, p. 1502).

Esses autores apoiados em Giraffa e Müller (2017) salientam ainda que esses ingressantes chegam no ensino superior com deficiências na formação básica, como por exemplo a insuficiência de conteúdos considerados como base, escrita e interpretação de textos, dificuldade em estabelecer rotinas e hábitos de estudo. Tais deficiências agregadas com “à sobrecarga cognitiva resultará no desenvolvimento superficial das competências necessárias para programar. Desse modo, não adianta esperar que os alunos consigam superar sozinhos essas lacunas e tenham sucesso no aprendizado dos conteúdos computacionais” (Oliveira; Pereira, 2019, p. 1502). Giraffa e Müller (2017) pontuam que não basta os professores diagnosticarem essas e outras lacunas apresentadas pelos estudantes que chegam na universidade. Cabe aos gestores de instituições de ensino superior em conjunto com seus professores planejarem e viabilizarem ações específicas que motivem e ofereçam subsídios teóricos e práticos para que haja por parte dos alunos a superação dessas deficiências.

É diante desse cenário, que este estudo se propôs a avaliar a eficácia da ferramenta Tinkercad, combinada com a linguagem Scratch, com a finalidade de contribuir com o aprendizado dos alunos dos cursos de bacharelado em Ciência da Computação, Sistemas de Informação e Tecnologias em Informática, quanto aos principais fundamentos de programação associados ao uso de sensores em aplicações comumente encontradas na área de Internet das Coisas. Neste sentido, este artigo a seguir apresenta um panorama acerca da fundamentação teórica na qual o presente trabalho se apoiou; a metodologia adotada para a aplicação da experiência analisada; os resultados; a devida discussão, e por fim a conclusão.

## 2. Fundamentação Teórica

### 2.1 A importância do ambiente Tinkercad e da linguagem Scratch no ensino da programação

O trabalho publicado por Eryilmaz et. al. (2021), por exemplo, destaca a importância do uso do ambiente Tinkercad® para motivar estudantes no desenvolvimento do pensamento computacional. As habilidades voltadas para o raciocínio lógico e a criação, usando o Tinkercad®, também são discutidas na pesquisa desenvolvida por Abburi, et. al. (2021). Em adição, o artigo de Mohapatra et.al. (2020) apresenta o uso do Tinkercad® como um instrumento de motivação para os estudantes desenvolverem projetos colaborativos, especialmente as atividades realizadas em grupos para a solução de um problema real.

No trabalho publicado por Bandeira et. al. (2019), a ferramenta Tinkercad® foi avaliada no contexto das dimensões propostas pela Base Nacional Comum Curricular (BNCC), permitindo que os docentes participantes do estudo pudessem refletir acerca do aperfeiçoamento do processo ensino-aprendizado a partir do seu uso na disciplina de matemática. O Tinkercad® também foi utilizado em conjunto com a ferramenta Google Classroom, no trabalho publicado por Shalannanda (2020), a fim de oferecer um laboratório virtual orientado para o projeto de sistemas digitais. O Tinkercad® foi empregado ainda na simulação de um sistema eletrônico capaz de rastrear os pontos de operação mais eficientes em um painel fotovoltaico, antes da sua implementação prática (Patel, et. al., 2020).

O trabalho de Cobo et. al. (2020) explorou o uso da linguagem Scratch, com a recomendação de exercícios, de modo a aprimorar o ensino da programação computacional em alunos universitários. Neste caso, os autores evidenciaram um elevado grau de satisfação dos alunos dos cursos de Ciência da Computação e Engenharias quanto ao uso da Scratch e do sistema de recomendação de exercícios desenvolvidos. Estudos adicionais que exploraram o uso do Scratch concluem que ele aprimora o ensino da lógica de programação ou do pensamento computacional. (Rezende, C., Bispo, E., 2018) avaliaram com sucesso a progressão de conhecimento com alunos de nível superior do curso de Física, por exemplo.

(Vourletsis, I., Politis, P., 2020) exploraram esse tema com alunos mais jovens do ensino fundamental. Além de perceberem um aprimoramento de sua capacidade na solução de problemas, concluíram que estava correlacionado com um aumento da vontade de enfrentar os desafios, ao invés de evitá-los. Em outro estudo apresentado por (Kalelioglu, F., Gulbahar, Y., 2014) também para alunos de ensino fundamental, apesar de não atingir resultados significativos na mesma direção, reforça o interesse dos alunos no aprendizado de lógica de programação e na facilidade de uso do Scratch.

A linguagem de programação Scratch foi lançada em 2007, a partir de estudos realizados no grupo de pesquisa Lifelong Kindergarten no Media Lab vinculado ao MIT ((Instituto de Tecnologia de Massachusets). Segundo Resnick (2020) “na última década, dezenas de milhões de crianças do mundo inteiro usaram o Scratch para criar suas próprias animações, jogos e histórias interativas, bem como para compartilhar o que criaram umas com as outras na comunidade on-line do Scratch ([scratch.mit.edu](http://scratch.mit.edu))” (p. 13). O Scratch foi criado com o propósito de favorecer com que qualquer pessoa possa programar. A partir dessa perspectiva qualquer pessoa sem conhecimentos prévios ou limitações de idade pode “criar suas histórias interativas, jogos, animações, simulações, podendo compartilhá-las com sua comunidade, pois o site do Scratch possibilita que os seus usuários compartilhem seus projetos, participando de uma comunidade on-line de programação” (Salazar, Odakura, Barvinski, 2015, p. 1293).

É importante destacar que o pensamento computacional experimentado com a linguagem Scratch é realizado a partir do uso de blocos funcionais, cujas cores representam instruções de software específicas e de modo bastante intuitivo, ao contrário de uma linguagem codificada, como ilustrado na Figura 1.

**Figura 1** – Programa em Scratch e o seu correspondente código em linguagem C para Arduino.



Fonte: Autores (2021).

Neste sentido, torna-se mais fácil a inserção dos estudantes no universo do pensamento computacional mais crítico, frente a um dado problema. Conforme Resnick (2009), os alunos devem aprender a pensar criativamente e analisar criticamente para terem sucesso na sociedade criativa atual. Poucas experiências estão suportando essas habilidades. A maioria simplesmente reforça velhas formas de ensino e aprendizagem usando novas tecnologias. Resnick, M. (2009) ainda destaca que o Scratch e outras tecnologias, onde podemos inserir o Tinkercad, só irão utilizar todo o seu potencial se os educadores em geral mudarem a maneira que pensam sobre educação e aprendizagem, priorizando o desenvolvimento dos alunos como pensadores criativos.

## 2.2 A importância no pensamento computacional na resolução de problemas

O pensamento computacional vai além de uma competência cognitiva favorável para a escrita de conceitos de programas informáticos, mas de um grande repertório de conhecimentos promissores para a resolução de distintos tipos de problemas, estabelecendo formas de comunicação social e gestão de diversos aspectos da vida cotidiana (Wing, 2006). Para Berrocoso et al. (2015) os procedimentos cognitivos empregados em atividades de programação são capazes de alavancar a construção de conhecimentos atendendo a apropriação, ou seja, os resultados que são meios de consumir os processos cognitivos e expressivos, tais como: programas, jogos, produtos hiper-mídia, robôs e outros objetos computacionais que são formas de materialização do pensamento e, conseqüentemente, são habilitados para simultaneamente transformá-los e enriquecê-lo.

Segundo Wing (2006), o pensamento computacional é um termo que caracteriza um grupo de processos de resoluções de diversos tipos de problemas, incluindo características como a sistematização e a análise de dados com a criação de soluções que empregam uma série de algoritmos. Para Berrocoso et al. (2015), a definição de pensamento computacional refere-se a uma complexa competência relacionada a um modelo de evolução de ideias que se aplicam em variados pontos da vida cotidiana, profissional e acadêmica, tendo como principal atributo a relação concomitante entre o pensamento abstrato matemático e o pensamento pragmático engenheiro. Nota-se, ainda, que o pensamento computacional se traduz como uma importante competência dirigida para a resolução de problemas de maneira perspicaz e imaginativa, características essas que os computadores não dispõem, e que todo ser deve conhecer para melhor e maior desenvolvimento na sociedade digital. O pensamento computacional relaciona-se a habilidade de pensar abstratamente em distintos níveis, emancipado da necessidade em dependência de equipamento e ou dispositivo.

Nesse sentido, os instrumentos oriundos do trabalho com o pensamento computacional ampliam a capacidade mental, pois traduzem as abstrações e intenções em solução de problemas, com a utilização de modelos computacionais, abrangendo diferentes conceitos das diversas áreas de conhecimento, sendo assim entendido como uma maneira de pensar dimensional, analítica e processual levando em consideração a eficiência, a simplicidade e elegância da solução, convertendo-se em uma

maneira de encarar grandes problemas com abstração e decomposição, propiciando a criação de uma vista mais abrangente, tencionando cenários, avaliando riscos e antecipando futuras intencionalidades (Wing, 2006). É importante evidenciar que o pensamento computacional surgiu na área de Ciência da Computação, mas vem tendo uma crescente evolução, não estando exclusivo ao ensino de tecnologia, mas se mostrando presente em todas as áreas de conhecimentos e níveis de ensino.

De acordo com Wing (2018) se considerarmos um futuro próximo, tende-se à valorização de um pensamento computacional mais “profundo”, a partir de abstrações mais inteligentes ou mais sofisticadas que poderão contribuir com que profissionais da ciência e engenharia concebam e analisem seus sistemas em uma escala de ordens de magnitude maior do que são capazes de lidar hoje. Para a autora, um pensamento computacional mais “profundo” nos apoiará na modelagem de sistemas com níveis de complexidade cada vez maiores, auxiliando ainda na análise de quantias significativas de dados. Nesse sentido, o pensamento computacional será essencial para a busca de inovações em diferentes áreas.

É nessa perspectiva e por reconhecermos que o uso do pensamento computacional já vem ocorrendo em diversas áreas, que propomos aos alunos dos cursos de bacharelado em Ciência da Computação e Sistemas de Informação e Tecnologias em Informática, por meio de desafios a serem alcançados a partir das ferramentas Tinkercad e da linguagem Scratch, contribuindo assim com sua formação enquanto graduandos e sua futura atuação na sociedade moderna.

### 2.3 Contextualização da Internet das Coisas na presente abordagem

Como mencionado anteriormente, a Internet das Coisas representa um amplo ambiente no qual existe um grande volume de dispositivos conectados, incluindo objetos com princípios de operação diversos, tais como sensores de luz, temperatura, umidade, pressão, gás, entre outros, além de atuadores, como os motores elétricos e os mecanismos hidráulicos ou pneumáticos, por exemplo. Esses dispositivos estão localizados na chamada camada de dispositivos (*Edge Computing* ou computação de borda em inglês), como ilustrado na Figura 2, logo o presente trabalho está direcionado para a simulação ou a experimentação de problemas relacionados ao domínio desta camada.

Logo, não foi objeto deste estudo a simulação de outros recursos oriundos das camadas em névoa (*Fog Computing* em inglês) ou camada em nuvem (*Cloud Computing* em inglês). Tais camadas são responsáveis, entre outras funções, pelo gerenciamento dos dispositivos da camada de borda, pela adequação de protocolos de rede de comunicação ou ainda pelo tratamento de uma grande massa de dados com o suporte de algoritmos de aprendizagem de máquina (Rahmani, et. al., 2018).

**Figura 2** – Uma típica estrutura em camadas para uma aplicação em Internet das Coisas.



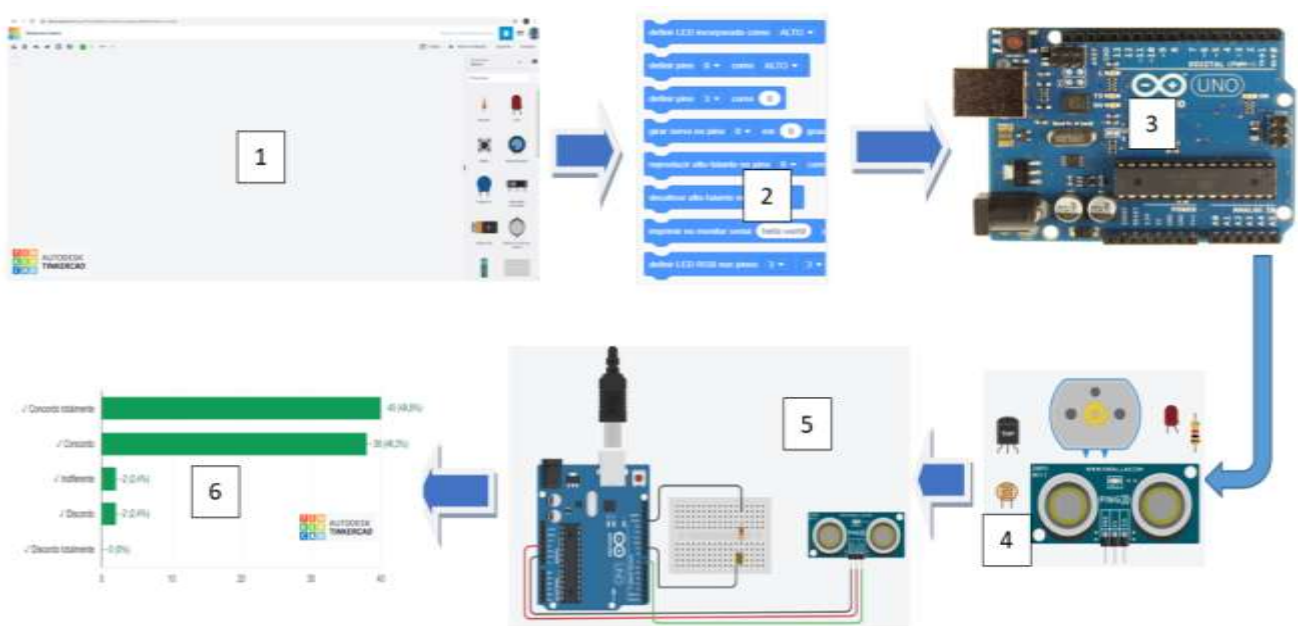
Fonte: Adaptado de Rahmani et al. (2018).

### 3. Metodologia

O presente estudo foi conduzido utilizando uma abordagem metodológica experimental e de natureza quantitativa, uma vez que a pesquisa permitiu compreender melhor o entendimento dos estudantes quanto ao uso da ferramenta Tinkercad e

da linguagem Scratch, no contexto da aplicação de sensores e atuadores voltados para a Internet das Coisas, a partir dos experimentos realizados e do questionário aplicado ao final das aulas. Todas as aulas foram ministradas remotamente, usando a plataforma Google Meeting, uma vez que ainda estávamos em um período de pandemia por conta da COVID-19. Conforme relatado por Gil (2002), e tratado neste trabalho, a formulação do problema de pesquisa refere-se à fase inicial do estudo e neste caso a questão fundamental a ser respondida está relacionada ao fato da referida ferramenta ser de fato capaz de fomentar o interesse dos alunos no assunto apresentado e também se a mesma permitiu o bom entendimento dos conceitos aderentes à disciplina. A Figura 3 ilustra as principais etapas associadas ao desenvolvimento do trabalho.

**Figura 3** – Etapas utilizadas na pesquisa.



Fonte: Autores (2021).

Na primeira etapa, identificada com o número 1 na Figura 3, foi apresentado aos alunos o ambiente da ferramenta Tinkercad, como o menu de componentes básico e avançado, bem como o espaço disponível para a montagem dos circuitos e também para a inserção do programa na linguagem Scratch. A segunda etapa do estudo consistiu em apresentar aos estudantes os principais recursos da linguagem, em especial as instruções gerais associadas a saída de informações, entrada, blocos de controle e blocos matemáticos. Na terceira fase foram apresentados em detalhes os recursos de hardware e software da plataforma Arduino Uno, disponível para uso no Tinkercad e na fase seguinte comentado o funcionamento dos principais componentes que foram usados nos experimentos, como o sensor de temperatura, o sensor de luz, o sensor ultrassônico, o motor de corrente contínua, o diodo LED e o resistor.

Vale mencionar que, na terceira etapa do trabalho foi ainda explicado para os alunos o princípio de conversão de um sinal analógico para o seu correspondente valor digital e também a representação decimal de uma grandeza mensurada a partir de um sensor. A quinta etapa abarcou a realização de cada experimento e a discussão com os alunos sobre os fenômenos associados a cada ensaio, tais como os tipos de sinais relacionados ao funcionamento do circuito e a sua relação com a operação dos blocos na linguagem Scratch.

Por fim, a sexta etapa do estudo consistiu em avaliar o entendimento e a compreensão dos alunos com o uso do ambiente Tinkercad, bem como o uso da linguagem Scratch, frente aos assuntos tratados para a disciplina denominada

“Internet das Coisas”, aplicando um questionário ao final das aulas realizadas em um período de 3 semanas. A próxima seção apresenta os principais experimentos realizados de forma remota com os alunos, bem como os resultados obtidos após a aplicação do questionário.

## 4. Resultados e Discussão

### 4.1 Realização dos experimentos

Como mencionado, a avaliação da ferramenta Tinkercad e da linguagem Scratch, como instrumentos de apoio ao aprendizado do pensamento computacional e de recursos de hardware empregados em Internet das Coisas, foi desenvolvida a partir de alguns experimentos trabalhados com os alunos durante as 4 aulas (4 semanas). Na primeira aula foram introduzidos os princípios de funcionamento de sensores e atuadores, além da apresentação e discussão de algumas aplicações que usam tais dispositivos. Ainda na primeira aula, o ambiente Tinkercad e a linguagem Scratch foram também apresentados.

Em seguida, foram propostos dois experimentos, conforme ilustrado na Figura 4, um primeiro sugerindo o uso da plataforma Arduino Uno para fazer um diodo LED piscar (alternar seu acionamento) a cada 1 segundo e um segundo capaz de fazer três LEDs serem acionados similarmente a um semáforo. Foi oferecido um tempo de 15 minutos para a realização do primeiro exercício e um tempo adicional de 20 minutos para desenvolver o segundo exercício, de modo que ao final o professor da disciplina discutiu cada solução com os alunos.

Foi possível notar que ao final dos dois primeiros experimentos houve uma reação bastante positiva por parte dos alunos, sendo que muitos deles manifestaram um interesse maior para discutir os conteúdos ministrados na continuidade da aula.

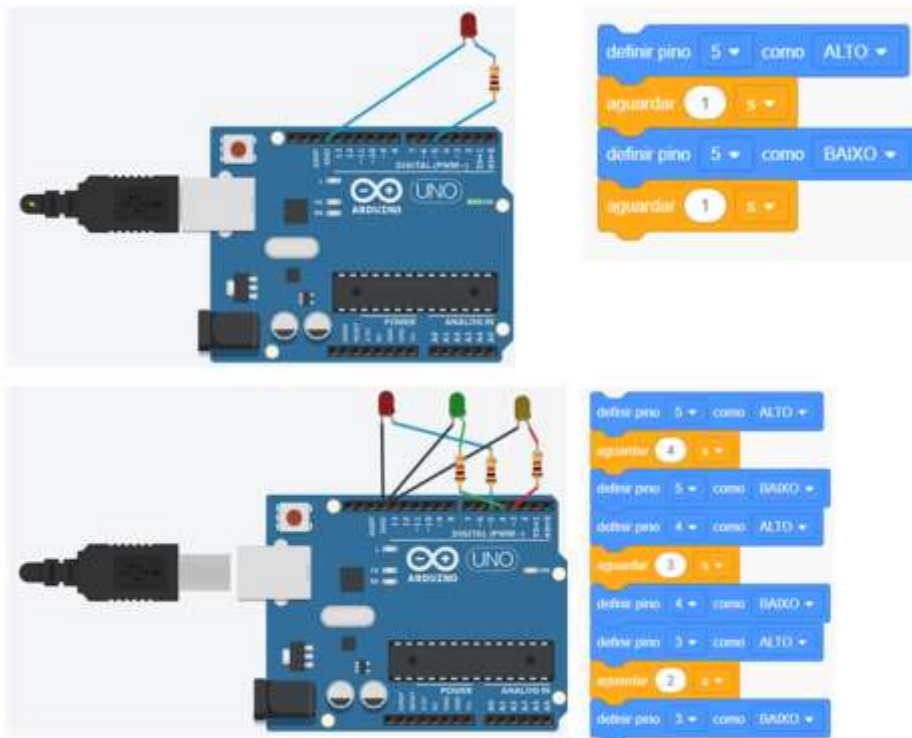
Na aula seguinte, ou seja, na segunda semana, os alunos desenvolveram um novo exercício usando agora um sensor para medição da temperatura ambiente. Todavia, antes da sua execução foi explicado para os alunos o processo de conversão de uma tensão elétrica aplicada na entrada de um canal analógico do Arduino Uno e o seu correspondente valor na representação numérica decimal, conforme ilustrado na Figura 3. Foi informado ainda que o sensor de temperatura operava em uma faixa entre  $-40^{\circ}\text{C}$  e  $+125^{\circ}\text{C}$ , o que corresponde respectivamente a uma tensão elétrica no canal analógico do Arduino entre 100 mV e 1,75V. O cálculo do valor decimal correspondente, ou seja, aquele entendido pelo Arduino a partir da tensão elétrica inserida no canal analógico, poderia ser calculado usando a equação 1, conforme citado por Souza (2014), sendo  $V_{entrada}$  a tensão gerada pelo sensor e a  $V_{ref}$  igual a 5V no presente caso. O valor constante de 1024 refere-se ao uso de um conversor analógico digital em 10 bits.

$$ValorDecimal = \frac{V_{entrada} \times 1024}{V_{ref}} \quad (1)$$

Como ilustrado na Figura 5, foi possível mostrar aos alunos não apenas o valor da temperatura medido em graus Celsius, mas também o seu correspondente valor na escala decimal, usando uma ferramenta do Tinkercad conhecida como “Monitor Serial”. Este recurso também está disponível no ambiente de programação de um Arduino físico.

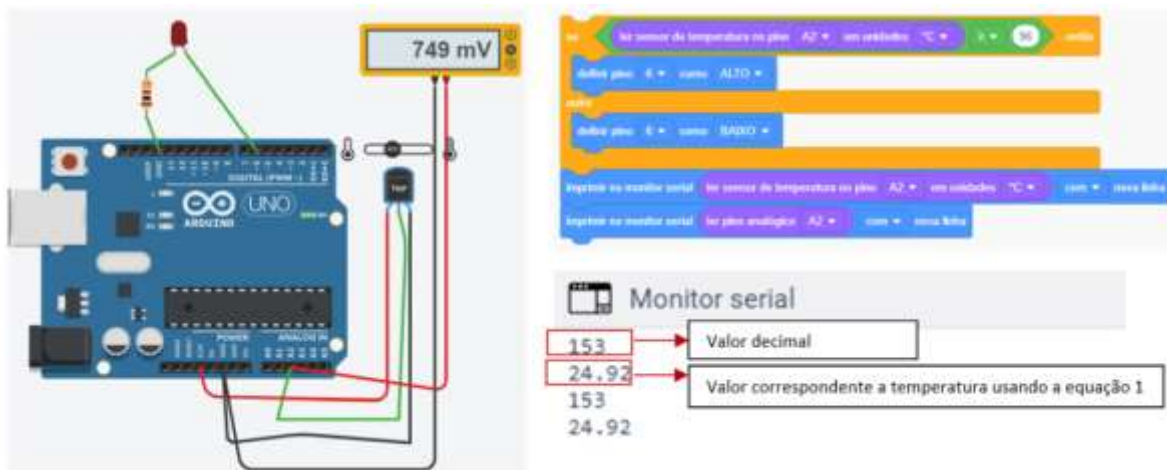


**Figura 4** – Dois primeiros experimentos executados no ambiente Tinkercad usando a linguagem Scratch.



Fonte: Autores (2021).

**Figura 5** – Experimento realizado no Tinkercad usando o sensor de temperatura.



Fonte: Autores (2021).

Vale mencionar que, o experimento ilustrado na Figura 5 foi proposto com a finalidade de trabalhar os conceitos relacionados a tomada de decisão na linguagem Scratch, visto que o diodo LED deveria acender a partir de uma temperatura limite de 96°C. Uma aplicação potencial para o exercício 3 seria o controle de temperatura de uma estufa, dado que um valor fora daquele especificado poderia indicar uma condição anormal na sua operação com o acendimento do LED ou ainda um aviso sonoro.

Adicionalmente, os alunos foram indagados acerca do monitoramento da temperatura diretamente na unidade Celsius, uma vez que o Arduino efetua a leitura de um nível de tensão elétrica em seu canal analógico. Neste caso, foi discutido com os

alunos a operação matemática necessária para representar o valor da temperatura a partir do seu correspondente número na escala decimal. A Figura 6 ilustra o código gerado para programação no Arduino, a partir do seu equivalente programa elaborado na linguagem Scratch e ao lado a equação da reta responsável por determinar qualquer valor de temperatura tomando por base o valor decimal recebido e interpretado pela plataforma Arduino.

Foi explicado aos alunos que a instrução na linguagem Scratch denominada “ler sensor de temperatura no pino A2 em unidades °C” elabora automaticamente a reta ilustrada na Figura 6, a fim de converter qualquer valor decimal na entrada do respectivo canal analógico no seu valor de temperatura, inclusive destacando os coeficientes angular e linear da equação encontrada. Essa apresentação evidenciou que em determinadas aplicações na área da Internet das Coisas os alunos precisam entender não apenas o princípio de funcionamento de um sensor, mas como o seu sinal deve ser tratado de modo a servir como uma informação importante para uma dada solução tecnológica.

**Figura 6** – Programa Scratch, o seu equivalente em código e a curva de conversão do sinal recebido pelo sensor no Arduino e seu correspondente valor em unidade de temperatura (°C).

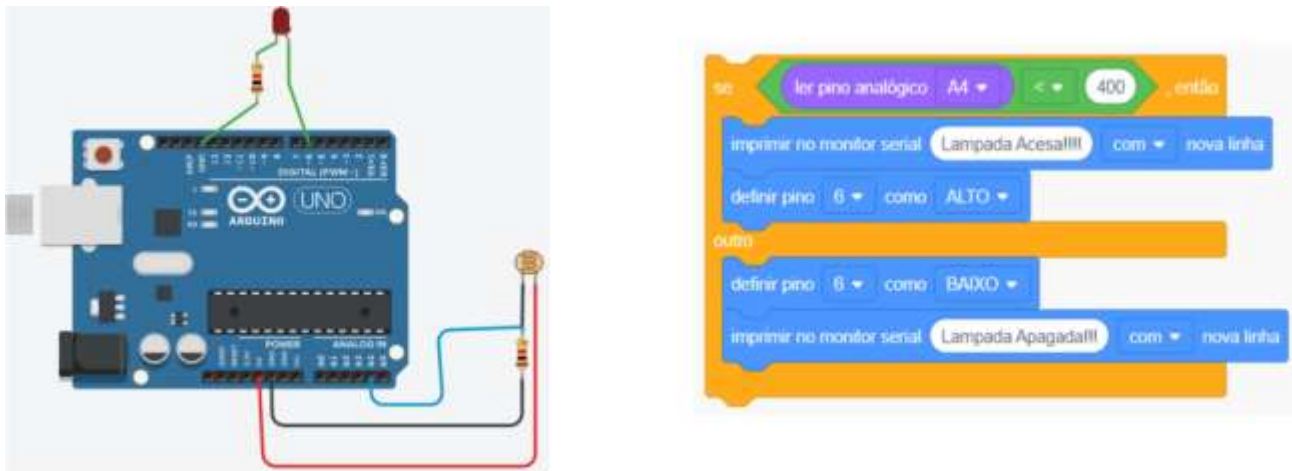


Fonte: Autores (2021).

Na segunda aula ainda foi proposto aos alunos o desenvolvimento de um segundo exercício usando o sensor de luz conhecido como LDR (Light Dependent Resistor em inglês), que implica em uma variação de resistência ôhmica em função da intensidade luminosa que incide na sua superfície (Figura 7). Logo, o exercício ofereceu um roteiro no qual o aluno deveria acender um diodo LED caso a intensidade luminosa no sensor estivesse abaixo de um limiar.

Neste caso, os alunos foram questionados acerca da instrução lógica necessária para fazer o sistema adequadamente, além da montagem física usando o Arduino, também foi bastante explorado o desenvolvimento do programa em Scratch e possíveis variações.

**Figura 7** – Experimento realizado no Tinkercad usando o sensor de luz LDR.



Fonte: Autores (2021).

Na terceira semana foi apresentado um problema real aos alunos que consistia no monitoramento do nível de um tanque de água usando um sensor ultrassônico. Neste caso, o sistema deveria gerar uma mensagem de texto alarmando para um nível alto do tanque, na condição menor do que 20 cm e ainda determinar o volume útil do referido tanque conhecendo as suas dimensões. O reservatório em questão apresentava um formato cilíndrico. Foi possível constatar que os alunos ficaram bastante estimulados com o problema proposto, uma vez que a sua solução compreendia conceitos de física, matemática, elétrica e computação. A Figura 8 ilustra o sistema montado e parte do programa simulado no Tinkercad, usando também a linguagem Scratch.

**Figura 8** – Experimento realizado no Tinkercad usando o sensor ultrassônico para monitoramento de nível de um tanque.

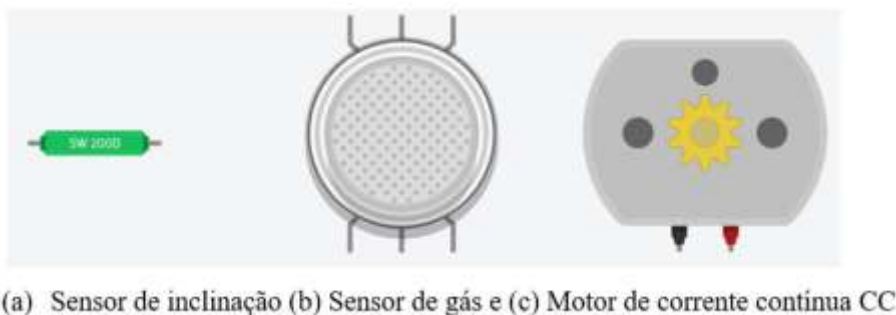


Fonte: Autores (2021).

Na quarta e última semana foi explorado o uso de outros tipos de dispositivos, tais como o sensor de inclinação, o sensor de gás e também o acionamento de motores de corrente contínua (Figura 9) simulando a operação de uma cadeira de rodas elétrica. Para cada dispositivo, foi estimulado ainda que os alunos elaborassem uma correlação entre o seu princípio de funcionamento e uma solução prática no campo da Internet das Coisas. Nesta aula, o professor discutiu ainda três sistemas usando os dispositivos acima destacados e os exemplos elaborados pelos estudantes. Ao final da quarta aula, foi aplicado um

questionário para os alunos de diferentes turmas no ano de 2021, de modo a compreender melhor a eficácia no uso da ferramenta Tinkercad e da linguagem Scratch no aprendizado da lógica, ou pensamento computacional, em especial com o emprego de dispositivos tipicamente usados em automação ou Internet das Coisas. A próxima subseção apresenta os resultados obtidos com as respostas apontadas pelos alunos em cada ano, usando um formulário eletrônico disponível no Google Forms.

**Figura 9** – Dispositivos adicionais usados em outros experimentos com o Tinkercad e a linguagem Scratch.



Fonte: Autores (2021).

#### 4.2 Resultados obtidos após a aplicação do questionário

Como mencionado, após a execução dos experimentos ao longo das quatro aulas, um questionário foi aplicado aos alunos de cada turma, de modo a reconhecer a contribuição do ambiente Tinkercad e da linguagem Scratch na elaboração do pensamento computacional em aplicações voltadas para o mundo da Internet das Coisas. No ano de 2021 a metodologia ora proposta foi aplicada em duas turmas no primeiro semestre letivo. A primeira turma era formada por alunos dos cursos de Ciência da Computação e Sistemas da Informação, sendo possível coletar 92 respostas (Tabela 1). A segunda turma era formada por alunos do curso de Tecnologia em Análise e Desenvolvimento de Sistemas e 82 alunos responderam ao questionário (Tabela 2).

Como é possível observar nos dados coletados, a grande maioria dos alunos entendeu que a linguagem Scratch é uma ferramenta bastante interessante para o exercício do pensamento computacional. Ademais, os alunos reconheceram que o ambiente Tinkercad é bastante simples e intuitivo, ao mesmo tempo que permite um bom entendimento de sistemas formados por hardware e software embarcado, com o uso de sensores e atuadores. Finalmente, grande parte dos alunos informou que recomendaria o uso do Tinkercad para amigos, ou colegas de trabalho, além de ficarem mais motivados para o uso da simulação computacional não exclusivamente em aplicações de Internet das Coisas, mas também em outros tipos de problemas. A tabela 2 mostra os resultados obtidos com a segunda turma.

De modo semelhante às respostas observadas para a primeira turma, os alunos da segunda turma também afirmaram que a linguagem Scratch é um recurso bastante interessante para o treinamento do raciocínio lógico ou o exercício do pensamento computacional. Adicionalmente, cerca de 95% dos estudantes entenderam que a ferramenta Tinkercad oferece um ambiente de uso simples e direto, além de contribuir para um melhor entendimento de sistemas que usam sensores e atuadores, normalmente usados na área de Internet das Coisas. A maioria dos alunos também recomendaria o seu uso para outros colegas e responderam que ficaram mais motivados com o uso da simulação computacional em problemas que vão além das aplicações em IoT.

Vale destacar que a grande maioria dos alunos, avaliados nas duas turmas, informaram que não possuíam uma boa experiência prévia no uso da linguagem Scratch ou da ferramenta Tinkercad, logo muitos dos estudantes tiveram um primeiro contato com o aludido ambiente computacional durante o curso das aulas. Cabe comentar ainda que a maioria informou que o

acesso ao Tinkercad ocorreu com o suporte de um computador pessoal, muito embora alguns alunos tenham desenvolvido os exercícios no próprio aparelho celular ou em um tablet.

**Tabela 1** – Questões e resultados obtidos para a primeira turma (92 respostas).

Questão	Concordo Totalmente	Concordo	Indiferente	Discordo	Discordo Totalmente
<b>Q1:</b> A linguagem Scratch é uma ferramenta bastante interessante para o treinamento do raciocínio, ou pensamento computacional.	44,6% (41)	53,3% (49)	2,2% (2)	0%	0%
<b>Q2:</b> O Tinkercad é um ambiente voltado para simulação computacional de uso bastante simplificado e direto.	43,7% (43)	50% (46)	2,2% (2)	1,1% (1)	0%
<b>Q3:</b> O Tinkercad permite a simulação e um bom entendimento não apenas de circuitos eletrônicos, mas também da combinação do hardware com o software, por exemplo.	36,7% (33)	56,7% (51)	5,6% (5)	1,1% (1)	0%
<b>Q4:</b> O uso do Tinkercad para a simulação de soluções usando Arduino e sensores, ou atuadores, foi bastante produtiva e de implementação bastante intuitiva.	42,9% (39)	51,6% (47)	3,3% (3)	2,2% (2)	0%
<b>Q5:</b> Eu recomendaria a um amigo, ou colega de trabalho, o uso do Tinkercad para o aprendizado de programação e simulação computacional.	49,5% (45)	47,3% (43)	3,3% (3)	0%	0%
<b>Q6:</b> Depois de usar o Tinkercad eu fiquei mais interessado no uso da simulação computacional, não apenas em aplicações IoT, mas aplicada em outros tipos de problemas.	45,1% (41)	39,6% (36)	14,3% (13)	1,1% (1)	0%

Fonte: Autores (2021).

**Tabela 2** – Questões e resultados obtidos para a segunda turma (82 respostas).

Questão	Concordo Totalmente	Concordo	Indiferente	Discordo	Discordo Totalmente
<b>Q1:</b> A linguagem Scratch é uma ferramenta bastante interessante para o treinamento do raciocínio, ou pensamento computacional.	48,8% (40)	46,3% (38)	2,4% (2)	2,4% (2)	0%
<b>Q2:</b> O Tinkercad é um ambiente voltado para simulação computacional de uso bastante simplificado e direto.	47,5% (38)	48,8% (39)	2,5% (2)	1,3% (1)	0%
<b>Q3:</b> O Tinkercad permite a simulação e um bom entendimento não apenas de circuitos eletrônicos, mas também da combinação do hardware com o software, por exemplo.	40,2% (33)	53,7% (44)	4,9% (4)	1,2% (1)	0%
<b>Q4:</b> O uso do Tinkercad para a simulação de soluções usando Arduino e sensores, ou atuadores, foi bastante produtiva e de implementação bastante intuitiva.	42,7% (35)	57,3% (47)	0%	0%	0%
<b>Q5:</b> Eu recomendaria a um amigo, ou colega de trabalho, o uso do Tinkercad para o aprendizado de programação e simulação computacional.	35,4% (29)	58,5% (48)	2,4% (2)	3,7% (3)	0%
<b>Q6:</b> Depois de usar o Tinkercad eu fiquei mais interessado no uso da simulação computacional, não apenas em aplicações IoT, mas aplicada em outros tipos de problemas.	32,9% (27)	54,9% (45)	6,1% (5)	3,7% (3)	2,4% (2)

Fonte: Autores (2021).

## 5. Conclusão

O presente trabalho abordou uma investigação acerca da eficácia da ferramenta Tinkercad e do uso da linguagem Scratch no ensino dos fundamentos de programação em aplicações no campo da Internet das Coisas. Mais especialmente, o estudo procurou abarcar com estudantes dos cursos da área de Informática o uso dos recursos de software em combinação com os dispositivos físicos, como a plataforma Arduino e a sua conexão com sensores e atuadores. Os experimentos virtuais programados e desenvolvidos com os alunos permitiram introduzir e avaliar conceitos importantes do pensamento computacional, como a aplicação da lógica matemática, o uso das diferentes instruções lógicas e a aplicação dos mecanismos de inferência na tomada de decisão do sistema a partir de sinais oriundos dos sensores, ou seja, a partir das informações recebidas do meio ambiente externo e também como a solução tecnológica criada poderia provocar movimentos ou ações desejadas com o suporte dos atuadores. Logo, foi possível avaliar, a partir das respostas coletadas nos questionários, que os alunos reconheceram a importância deste ambiente computacional como um instrumento de apoio e incentivo para o aprendizado não apenas do pensamento computacional voltado para a Internet das Coisas, mas também para outros problemas aderentes ao mundo da tecnologia.

Neste sentido, os autores avaliam que o objetivo primordial da presente pesquisa foi alcançado, na medida em que os experimentos realizados com a ferramenta Tinkercad e a linguagem Scratch, bem como os resultados apresentados com a aplicação dos questionários mostraram que tal abordagem permitiu um importante aprimoramento do processo de ensino e aprendizagem dos fundamentos de programação, em especial com o uso combinado de recursos simulados em software e hardware.

Para a continuidade do trabalho, recomenda-se o uso da ferramenta Tinkercad e a aplicação de outros experimentos que consigam introduzir aos estudantes alguns conceitos fundamentais do campo da Inteligência Artificial, uma vez que os algoritmos de aprendizagem de máquina estão sendo cada vez mais associados ao universo da Internet das Coisas e às soluções do mundo contemporâneo.

## Agradecimentos

Os autores agradecem a Universidade Nove de Julho por todo o suporte oferecido durante o desenvolvimento do estudo.

## Referências

- Abburi, R., Praveena, M., Priyakanth, R. (2021). TinkerCad – a Web Based Application for Virtual Labs to Help Learners Think, Create and Make. *Journal of Engineering Education Transformations*, 34. 535-541.
- Bandeira, L.M.S.A., Araújo, N.R.R.F., Farias, F.L.O., Barros, A. C.M., Queiroz, J.F., Nunes, I.D. & Oliveira, L.A.H.G. (2019) Instrumento de Avaliação do Software Educacional “Tinkercad”: uma Visão fundamentada na BNCC. VIII Congresso Brasileiro de Informática (CBIE 2019), 1324-1328.
- Berrococo, J., Sánchez, M. & Arroyo, M. (2015). “El pensamiento computacional y las nuevas ecologías del aprendizaje”. *RED-Revista de Educación a Distancia*, 46(3).
- Chabanet, S., Haouzi-El, H.B. & Thomas, P. (2021). Coupling Digital Simulation and Machine Learning Metamodel Through an Active Learning Approach in Industry 4.0 context. *Computers in Industry*, 133. 1-12.
- Chowdhury, A. T., Mukhtadir, S. & Zakir, G. S. (2021). Smartgeiger – Development of Background Radiation Monitoring With IoT Based Device. *International Conference on Automation, Control and Mechatronics for Industry (ACMI)*, 1- 6.
- Cobo, C. J., Puris, A., Hernández, P. N., Galindo, J.A. & Benavides, D. (2020). Recommender Systems and Scratch: An Integrated Approach for Enhancing Computer Programming Learning. *IEEE Transactions on Learning Technologies*, 13(2). 387-403.
- Cornetta, G., Touhafi, A., Togou, M.A. & Muntean, G. (2020). Fabrication-as-a-Service: A Web-Based Solution for STEM Education Using Internet of Things. *IEEE Internet of Things Journal*, 7(2). 1519-1530.
- Dias, C.G., Feitosa, N.T. & Silva da, L.C. (2020). The Use of Scilab-Cloud for Teaching Digital Signal Processing Concepts in Electrical Engineering Curricula. *American Scientific Research Journal of Engineering, Technology and Sciences (ASRJETS)*, 74(1). 101-114.
- Dias, C.G. & Silva da, L.C. (2021). A Virtual-Lab Tool for Teaching the Fundamentals of a DC Motor-Generator Operation Using Excel-VBA. *International Journal of Computer Applications*, 183(24). 1-8.
- Díaz, L. M., Hernández, C. M., Ortiz, A. V. & Gaytán-Lugo, L. S. (2019). Tinkercad and Codeblocks in a Summer Course: an Attempt to Explain Observed Engagement and Enthusiasm. *IEEE Blocks and Beyond Workshop (B&B)*, 43-47.
- Eryilmaz, S. & Deniz, G. (2021). Effect of Tinkercad on Students’ Computational Thinking Skills and Perceptions: a Case of Ankara Province. *The Turkish Online Journal of Educational Technology*, 20 (1). 25-38.
- Fukumoto, H., Yamaguchi, T., Ishibashi, M. & Furukawa, T. (2021). Developing a Remote Laboratory System of Stepper Motor for Learning Support. *IEEE Transactions on Education*, 64(3). 292-298.
- Gil, A. C. (2002). *Como elaborar projetos de pesquisa*. Editora Atlas.
- Giraffa, L. M. M. & Müller, L. (2017). Metodologia baseada em sala de aula invertida e resolução de problemas relacionado ao cotidiano dos estudantes: uma proposta para ensinar programação para iniciantes. *International Journal on Computational Thinking*, 1 (1). (52-67).
- Gruenewald, A., Giesser, C., Buechner, S., Gibas, C. & Brueck, R. (2021). Going Virtual: Teaching Practical Skills of Circuit Design and Programming for Heterogeneous Groups Online. *IEEE Global Engineering Education Conference (EDUCON)*, 404-412.
- Kinjo, M. E., Librantz, A.F.H., de Souza, E. M. & Galdino, M. (2021). Criticality assessment of the components of IoT system in health using the AHP method. *Research, Society and Development*, 10 (2). 1-14.
- Magnus, D. M., Carbonera, L.F.B., Pfitscher, L.L., Farret, F.A., Bernardon, D.P. & Tavares, A.A. (2020). An Educational Laboratory Approach for Hybrid Project-Based Learning of Synchronous Machine Stability and Control: A Case Study. 63(1). 48-55.

- Mena, D. M., Papapanagiotou, I., & Yang, B. (2018). Internet of things: Survey on security. *Information Security Journal: A Global Perspective*, 27 (3), 162-182.
- Mohapatra, B. N., Mohapatra, R.K., Joshi, J. & Zagade, S. (2020). Easy performance based learning of Arduino and sensors through Tinkercad. *International Journal of Open Information Technologies*, 8 (10), 73-76.
- Oliveira, S. (2017). *Internet das Coisas com ESP8266, Arduino e Raspberry PI*. Novatec Editora Ltda.
- Oliveira, C. M. & Pereira, R. (2019). Desenvolvimento do Pensamento Computacional no Ensino Superior em Ciência da Computação. VIII Congresso Brasileiro de Informática na Educação (CBIE 2019). Anais dos Workshops do VIII Congresso Brasileiro de Informática na Educação (WCBIE 2019), 1502-1506.
- Pate, K., Borole, S., Ramaneti, K., Hejib, A. & Sing, R.R. (2020). Design and Implementation of Sun Tracking Solar Panel and Smart Wiping Mechanism Using Tinkercad. *Materials Science and Engineering IOF Conf. Series*, 1-13.
- Paul, B., Paul, C., Varghese, A. S. P., Shajoo, S. & Kurian, N. (2018). Design of a Power Feeder for Elderly & Simulation of Motor Circuit Developed using AUTODESK TINKERCAD. *International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET)*, 1-4.
- Rahmani, A.M., Liljeberg, P., Preden, J.S & Jantsch, A. (2018). *Fog Computing in the Internet of Things: Intelligence at the Edge*. Springer, Switzerland.
- Resnick, M. (2020). Jardim de infância para a vida toda: por uma aprendizagem criativa, mão na massa e relevante para todos. Tradução: Mariana Casetto Cruz e Livia Rulli Sobral. Revisão Técnica: Carolina Rodeghiero e Leo Burd. Porto Alegre: Penso.
- Resnick, M. (2009). Sowing the Seeds for a more Creative Society. *CHI '09: SIGCHI Conference on Human Factors in Computing Systems*, 1-6.
- Rezende, C. & Bispo, E. (2018). Comparison between the use of pseudocode and visual programming in programming teaching: An evaluation from scratch tool. *IEEE 13th Iberian Conference on Information Systems and Technologies (CISTI)*, 1-5.
- Salazar, R., Odakura, V., Barvinski, C. (2015). Scratch no ensino superior: motivação. *CBIE-LACLO 2015. Anais do XXVI Simpósio Brasileiro de Informática na Educação*, 1293-1302.
- Schwab, K. & Davis, N. (2018). *Aplicando a Quarta Revolução Industrial*. World Economic Forum. EDIPRO.
- Shalannanda, W. (2020). Digital Logic Design Laboratory Using Autodesk Tinkercad and Google Classrom. *14th International Conference on Telecommunication Systems, Services, and Applications (TSSA)*, 1-5.
- Souza, F. (2014). *Arduino UNO – Taxa de amostragem do conversor A/D*. Recuperado em 10 de Outubro, 2021, a partir de <https://www.embarcados.com.br/arduino-taxa-de-amostragem-conversor-ad/>
- Vourletsis, I. & Politis, P., (2020). Effects of a Computational Thinking Experimental Course on Students' Perceptions of Their Problem-Solving Skills. *9th International Conference on Educational and Information Technology (ICEIT 2020)*, 14-20.
- Wing, J. M. (2006). Computational Thinking, *CACM*, 49(3), 33-35.
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Phil. Trans. R. Soc. A*, 366, 3717–3725.