EVOLUINDO O BRINQUEDO ROPE PARA SE TORNAR UM SMARTTOY

Luiz Fernando Noschang

Novembro/2020

Orientador: André Luís Alice Raabe, Dr.

Área de Concentração: Computação Aplicada

Linha de Pesquisa: Informática na Educação

Palavras-chave: pensamento computacional, aprendizagem, educação infantil, RoPE, IoT

Número de páginas: 11

**RESUMO**

O pensamento computacional é de extrema importância para a sociedade, pois capacita as pessoas a resolverem problemas complexos através do desenvolvimento de um conjunto útil de habilidades: decomposição, abstração, reconhecimento de padrões e pensamento algorítmico. Por esse motivo, pesquisadores da área têm feito um grande esforço em promover o desenvolvimento do pensamento computacional o mais cedo possível, já nos primeiros anos da educação infantil. No entanto, desenvolver o pensamento computacional em crianças requer a utilização de instrumentos que levem em consideração as particularidades desse público, entre elas, a realização de atividades lúdicas, o trabalho em conjunto, o engajamento com o material didático, o feedback imediato das ações e a adoção de recursos visuais e auditivos. Nesse sentido, brinquedos de programar como a Bee-Bot, Kibo, Pro-Bot e o RoPE têm se mostrado excelentes ferramentas, pois conseguem atender a boa parte desses requisitos. Porém, tão importante quanto desenvolver o pensamento computacional é a capacidade de avaliar o grau desse desenvolvimento, uma vez que isso permite determinar o quão eficazes são as metodologias utilizadas e, assim, aperfeiçoá-las.

Sendo assim, existe a necessidade de se criar uma ferramenta que possa auxiliar no

Levando em conta esses fatores, escolheu-se para o desenvolvimento deste trabalho o

o RoPE, um robô programável educacional de baixo custo produzido no Brasil. O RoPE foi escolhido por ser um brinquedo que atende aos requisitos necessários ao se trabalhar com educação infantil: é um objeto tangível (palpável), é lúdico e promove o trabalho em conjunto

Nesse contexto, o objetivo deste trabalho é modificar o brinquedo RoPE

desenvolver uma métrica que permita fazer essa mensuração do pensamento computacional em crianças de 4 a 7 anos e que possa ser usada como um referencial pelos pesquisadores da área. Por se tratar da educação infantil, optou-se por utilizar como ferramenta para o desenvolvimento do trabalho o RoPE, um robô programável educacional de baixo custo produzido no Brasil. O RoPE foi escolhido por ser um brinquedo que atende aos requisitos necessários ao se trabalhar com educação infantil: é um objeto tangível (palpável), é lúdico e promove o trabalho em conjunto. Para viabilizar o desenvolvimento do trabalho, por consequência da pandemia do COVID-19, o RoPE será adaptado para incluir um módulo WiFI e permitir que os testes sejam executados na residência dos indivíduos objetos do teste, e que os dados coletados sejam enviados através da Internet para posterior análise usando técnicas de estatística aplicada à validação de instrumentos. Ao final do trabalho espera-se ter um conjunto de testes que possa ser reproduzido/aplicado por demais pesquisadores e que possa dizer o nível de desenvolvimento do pensamento computacional dos indivíduos em cada uma das habilidades citadas: decomposição, abstração, reconhecimento de padrões e pensamento algorítmico

# PROJETO

O pensamento computacional é um assunto que vem sendo discutido há tempos pela comunidade científica, tendo sido abordado já na década de 80 por Seymour Papert. Em seu trabalho, [(PAPERT, 1980)](https://www.zotero.org/google-docs/?12sTHD), já defendia a importância do pensamento computacional ao perceber seus efeitos positivos no aprendizado. Segundo ele, ao programar um computador o indivíduo está ensinando a máquina a pensar e, ao fazer isso, passa a compreender a sua própria forma de pensar.

Após o trabalho de Papert outros pesquisadores abordaram o pensamento computacional de uma forma mais aprofundada gerando um maior interesse pelo tema. Em seu artigo, [(WING, 2006)](https://www.zotero.org/google-docs/?y3F90B) demonstra como o pensamento computacional pode ser aplicado em diversas áreas além da Ciência da Computação, citando, por exemplo, o uso do aprendizado de máquina na estatística e o uso de abstração na biologia para representar a estrutura de proteínas. Wing resume o pensamento computacional como a habilidade de reformular um problema complexo em um problema mais simples e possível de resolver, usando: redução, incorporação, transformação ou simulação.

Definir o conceito exato de pensamento computacional é uma tarefa desafiadora, pois o assunto é bastante abrangente e cada autor aborda diferentes aspectos do mesmo. No entanto, existe um consenso, e este é o principal ponto que precisa ser compreendido, apesar de o pensamento computacional estar relacionado aos computadores ele não se trata dos computadores em si, mas sim de solucionar problemas. De acordo com (CURZON; BLACK; et al, 2009) citado no trabalho de [(MOHAGHEGH; MCCAULEY, 2016)](https://www.zotero.org/google-docs/?rkiX98), "o pensamento computacional é uma coleção de múltiplas habilidades de resolução de problemas com base nos princípios fundamentais da Ciência da Computação".

Uma vez que o pensamento computacional se trata em resolver problemas

demonstra claros como ele pode ser aplicado em problemas em diversas disciplinas e do dia a dia em diversas áreas de conh

é de extrema importância para a sociedade, pois capacita as pessoas a resolverem problemas complexos através do desenvolvimento de um conjunto útil de habilidades: decomposição, abstração, reconhecimento de padrões e pensamento algorítmico

De acordo com [(MOHAGHEGH; MCCAULEY, 2016)](https://www.zotero.org/google-docs/?UN56lE) o pensamento computacional

Falar sobre o pensamento computacional e porque ele é importante, principalmente na educação infantil. Falar sobre os esforços de pesquisa para introduzir o pensamento computacional na educação e onde esse trabalho se encaixa

Dentro deste contexto, este trabalho procura fazer uma contribuição na área de

## PROBLEMA DE PESQUISA

Verificar e validar se existem ou não métricas para avaliação do PC na educação infantil. Se não houver, salientar que esta é a proposta do trabalho. Se houver, discutir as soluções existentes e argumentar porque a nossa solução será diferente

Nesta seção, você deve descrever qual é o problema a ser resolvido. É necessário evidenciar que existem questões em aberto, que o tema é complexo e que há interesse na comunidade em resolver o problema. O texto deve responder às seguintes perguntas:

Qual a relevância e complexidade do problema apresentado?

Existe alguma solução consolidada ou o problema ainda está em aberto?

Nesta seção, você deve ainda indicar quais as perguntas de pesquisa que você buscará responder por meio do seu trabalho. Usualmente, as perguntas permitem a formulação de uma ou mais hipóteses que serão apresentadas na seção a seguir (Solução Proposta).

### Solução Proposta

Falar sobre a nossa solução. Citar o uso do RoPE e dos tapetes e o módulo WiFi

### Delimitação de Escopo

Delimitar o escopo a crianças de 4 a 7 anos apenas

### Justificativa

## OBJETIVOS

Esta seção formaliza os objetivos do trabalho, conforme descrito a seguir.

### Objetivo Geral

Avaliar o grau de impacto do idioma de uma linguagem de programação na aprendizagem da lógica de programação

### Objetivos Específicos

1. Avaliar trabalhos similares que relatem sobre a influência do idioma na aprendizagem de lógica;
2. Projetar o experimento;
3. Desenvolver um ambiente para teste que tenha a mesma IDE para ambos idiomas;
4. Realizar um experimento com turmas de graduação;

## METODOLOGIA

Nas seções seguintes a metodologia a ser utilizada nesta pesquisa é classificada e uma síntese dos procedimentos metodológicos utilizados é apresentada.

### Metodologia da Pesquisa

Neste trabalho será aplicado o método indutivo, o qual consiste em se estabelecer uma verdade universal ou uma referência geral com base em dados e fatos previamente conhecidos e comprovados. Nesta pesquisa parte-se do conhecimento prévio de que existe uma dificuldade no aprendizado de programação relacionada ao idioma no Brasil e busca-se verificar o grau de impacto do idioma nesse aprendizado. Para verificar este grau de impacto será usada uma técnica estatística denominada Propensity score matching, desta forma, a pesquisa utilizada a abordagem quantitativa.

Sob o ponto de vista da natureza da pesquisa, esta é uma pesquisa aplicada. A pesquisa aplicada objetiva gerar conhecimentos para aplicações práticas dirigidos à solução de problemas. Neste trabalho o conhecimento gerado será o grau de impacto que o idioma exerce no aprendizado de programação e poderá ser aplicado em trabalhos futuros para solucionar esse problema.

Ainda, sob o ponto de vista do objetivo da pesquisa, esta é uma pesquisa exploratória. As pesquisas exploratórias têm como objetivo proporcionar maior familiaridade com o problema, com vistas a torná-lo mais explícito ou a constituir hipóteses, com isso, buscam criar fundamento para pesquisas mais aprofundadas. Nesse sentido, esta pesquisa visa aplicar conhecimentos e técnicas existentes através de estudos de caso, onde serão confirmadas ou refutadas as hipóteses formuladas.

### Procedimentos Metodológicos

Esta seção apresenta os procedimentos metodológicos adotados nesta pesquisa.

**Revisão bibliográfica:** Esta etapa tem como objetivo proporcionar a fundamentação teórica necessária ao desenvolvimento da pesquisa.

**Revisão sistemática da literatura:** Esta etapa tem como objetivo realizar uma revisão sistemática da literatura analisando os trabalhos similares que tratam do impacto do idioma no aprendizado de programação.

**Implementação do ambiente de testes:** Esta etapa tem como objetivo implementar o ambiente de testes que será usado no experimento.

**Realização do experimento:** Esta etapa tem como objetivo realizar o experimento com as turmas de graduação a fim de coletar os dados para análise.

**Análise dos dados coletados:** Esta etapa tem como objetivo aplicar a técnica de Propensity Score Matching nos dados coletados e analisá-los.

**Conclusão:** Esta etapa tem como objetivo analisar as contribuições da pesquisa e apresentar sugestões de trabalhos futuros relevantes.

## MÓDULO ESP8266

Um dos objetivos deste trabalho é adaptar o RoPE para que ele possa se conectar à internet através de uma rede WiFi e interagir com outros dispositivos e aplicativos remotamente. Para viabilizar essa conexão é necessário fazer alterações no *hardware* do brinquedo, pois o microcontrolador ATmega328p, usado atualmente, não oferece essa funcionalidade.

Uma das possibilidades seria substituir o ATmeg328p por um outro chip que possua essa função, no entanto, isso não é viável no momento, pois envolve ter que recodificar todo o *firmware* do RoPE para o novo microcontrolador, tarefa essa que poderá ser realizada em um trabalho futuro. Por esse motivo, se faz necessário usar no projeto um módulo que possa ser acoplado ao brinquedo e se comunicar com o ATmega através do barramento serial.

O módulo escolhido para o desenvolvimento do Smart RoPE foi ESP8266 (Figura 1), amplamente utilizado em projetos IoT em conjunto com o ATMega (Arduino). Além de possuir vários recursos úteis que serão abordados nesse capítulo, o ESP8266 possui um baixo custo se comparado a outras opções. De acordo com (MEHTA, 2015) o ESP8266 pode ser encontrado por menos de $5,00 enquanto seus concorrentes chegam a custar entre $30,00 e $60,00. Esse é um fator bastante decisivo para sua escolha, uma vez que o RoPE tem a premissa de ser um brinquedo de baixo custo e acessível.



Figura 1. Módulo ESP8266

O RoPE é amplamente utilizado em sala de aula em atividades que duram de 30 a 60 minutos em uso contínuo, portanto, é necessário manter um baixo consumo de energia para garantir seu funcionamento durante este período.

## PROTOCOLO MQTT

Ao trabalhar com IoT uma das etapas do desenvolvimento é escolher o protocolo de comunicação que será adotado para a troca de mensagens entre os dispositivos na camada de aplicação. Para este projeto, foi escolhido o protocolo MQTT, desenvolvido pela IBM. Neste capítulo é apresentado o funcionamento deste protocolo e os motivos que levaram à sua escolha.

O protocolo MQTT utiliza uma arquitetura baseada no modelo *publisher/subscriber* (editor/assinante), no qual os *editores* são os dispositivos que publicam informações e os *assinantes* são os dispositivos que as consomem. A troca de informações é coordenada por um servidor denominado *broker*, que recebe as mensagens dos *editores* e as encaminha para os *assinantes* correspondentes.

|  |
| --- |
| casa/garagem/portao  Assinante  Y  Fechar  Ligar  casa/garagem/portao  casa/cozinha/lampada  casa/cozinha/lampada  Broker  Assinante  X |

Figura 2. Arquitetura do MQTT

O encaminhamento das mensagens é realizado adotando um conceito denominado *topic* (tema), que nada mais é do que uma cadeia de caracteres contida no pacote MQTT, similar a uma URL, e que funciona como um canal de comunicação. Quando desejam transmitir uma informação os *editores* devem enviar uma mensagem do tipo *publish* ao *broker* com o conteúdo da mensagem (*payload)* e o *tema* no qual desejam publicar. Já os *assinantes* devem enviar uma mensagem do tipo *subscribe* contendo o *tema* no qual desejam se inscrever, assim receberão o conteúdo que os *editores* tiverem publicado neste mesmo *tema*.

Para uma melhor compreensão, o diagrama da Figura 2 ilustra uma aplicação hipotética em MQTT para automação residencial. Nesta aplicação o *editor A* publica uma mensagem no *tema* “casa/cozinha/lampada” para ligar a lâmpada da cozinha. O *assinante X*, que se inscreveu neste mesmo *tema*, recebe esta mensagem e atua para executar a ação, ligando a lâmpada. O *assinante Y*, por sua vez, não recebe a mensagem, pois está inscrito em outro *tema*. É importante salientar que um dispositivo pode se inscrever e/ou publicar em vários *temas* ao mesmo tempo, além disso, pode atuar simultaneamente como *editor* e *assinante* (SAHADEVAN et al., 2017).

De acordo com (MANANDHAR, 2017), ao empregar este modelo o MQTT permite que a comunicação ocorra de forma assíncrona, sem necessitar que *editor* e *assinante* estejam conectados no mesmo instante ou tenham conhecimento da existência um do outro. A possibilidade de funcionar de forma assíncrona é importante para este trabalho porque viabiliza a comunicação de dispositivos remotos com o RoPE em ambientes onde a infraestrutura de rede é precária e apresenta instabilidades, como é o caso de algumas escolas públicas. Ainda, segundo o autor, a intermediação do *broker* elimina a necessidade de os dispositivos conhecerem os endereços IP uns dos outros, permitindo que a comunicação ocorra mesmo em redes que mascaram os endereços IPs usando NAT.

Durante o encaminhamento das mensagens o MQTT permite escolher entre 3 níveis de QoS, conforme mencionado por (SHINHO LEE et al., 2013). No primeiro nível (level 0), as mensagens são transmitidas apenas uma vez e não há verificação de entrega, havendo possibilidade de perda. No segundo nível (level 1), as mensagens são enviadas no mínimo uma vez e há verificação de entrega, no entanto, pode haver duplicação caso o pacote contendo a confirmação do recebimento se perca. Já no terceiro nível (level 2), o *broker* usa uma verificação de 4 vias para garantir que as mensagens sejam entregues apenas uma vez, ao custo de uma maior latência na comunicação.

No contexto do RoPE a confiabilidade na entrega das mensagens é extremamente necessária. Considere um cenário no qual um dispositivo se conecta remotamente ao RoPE com o intuito de lhe enviar a seguinte sequência de comandos de movimento: 1) andar para frente; 2) girar à esquerda; 3) andar para trás. Se não houver um mecanismo que garanta a entrega das mensagens o RoPE irá executar uma sequência que não corresponde ao programa original, podendo haver falta de comandos ou comandos duplicados. O QoS level 2 do MQTT resolve este problema. No entanto, (HWANG; PARK; SHON, 2016) menciona que o MQTT não garante a ordem das mensagens. O autor sugere uma solução que consiste em acrescentar ao *payload* um campo que indica a sequência da mensagem. Esta estratégia, ou outra similar, poderá ser usada no trabalho para resolver o problema da ordenação.

A complexidade é outro ponto relevante para o projeto, pois interfere no tempo necessário para o desenvolvimento, além de impactar na manutenção do *firmware* conforme o RoPE for evoluindo. Neste quesito, o MQTT também se mostra uma ótima opção, pois segundo [(ELHADI et al., 2018)](https://www.zotero.org/google-docs/?dYH4o2) é um protocolo fácil de ser implementado e independe de linguagem de programação. É possível encontrar bibliotecas de MQTT em diversas linguagens, incluindo C e PHP que serão usadas nesse trabalho. A implementação em PHP será útil para desenvolver uma aplicação de estudo de caso ao final do trabalho. Já a implementação em C será usada no módulo ESP8266 e conta com mais de 10 opções de bibliotecas, conforme apontado por (OLIVEIRA et al., 2018).

Um dos objetivos do trabalho é permitir que o RoPE seja controlado remotamente através da Internet somente por dispositivos autorizados. Para isso o MQTT possui um mecanismo de segurança baseado em usuário e senha (SONI; MAKWANA, 2017) com suporte a TLS dependendo do *broker* utilizado. Esse sistema não é tão seguro quanto outros mecanismos de autenticação, mas é o suficiente para esta implementação. Trabalhos futuros poderão melhorar a camada de segurança do RoPE.

A escolha do protocolo MQTT requer que seja adotado também um *broker*. Dentre as diversas implementações de *broker* disponíveis, optou-se por usar nesse trabalho o *Mosquitto*, um *broker* de código aberto, escrito na linguagem C e mantido pela *Eclipse Foundation*. O *Mosquitto* se mostra uma boa escolha pois possui bom desempenho em relação a uso de CPU e memória (TORRES; ROCHA; DE SOUZA, 2016), além de suportar todos os níveis de QoS e a possibilidade de criação dinâmica de *temas* (SONI; MAKWANA, 2017).

Outro fator que pesou na escolha desse *broker* é o fato de ele já ter sido usado com sucesso em um trabalho similar. Em seu trabalho, (MARTINS, 2019) desenvolveu um sistema de automação residencial utilizando os mesmos componentes deste projeto, o módulo ESP8266 em conjunto com o protocolo MQTT e o *Mosquitto* atuando como *broker*.

**Plano de trabalho**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Quadro 1. Cronograma de execução para o segundo semestre do primeiro ano (2019/2)   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Atividade** | **Jul** | **Ago** | **Set** | **Out** | **Nov** | **Dez** | | Revisão bibliográfica | x | x |  |  |  |  | | Revisão sistemática |  |  | x | x |  |  | | Implementação do ambiente de testes |  |  |  |  | x | x | |
| Quadro 2. Cronograma de execução para o primeiro semestre do segundo ano (2020/1)   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Atividade** | **Jan** | **Fev** | **Mar** | **Abr** | **Mai** | **Jun** | | Implementação do ambiente de testes | x | x |  |  |  |  | | Realização do experimento |  |  | x | x |  |  | | Análise dos dados coletados |  |  |  |  | x | x |   Quadro 3. Cronograma de execução para o segundo semestre do segundo ano (2020/2)   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Atividade** | **Jul** | **Ago** | **Set** | **Out** | **Nov** | **Dez** | | Escrita da dissertação | x | x | x | x | x | x | |

**Análise de riscos**

Não existem riscos que impeçam o desenvolvimento deste trabalho

**REFERÊNCIAS**

HWANG, H. C.; PARK, J.; SHON, J. G. Design and Implementation of a Reliable Message Transmission System Based on MQTT Protocol in IoT. **Wireless Personal Communications**, v. 91, n. 4, p. 1765–1777, dez. 2016.

MANANDHAR, S. MQTT based communication in IoT. p. 56, 31 maio 2017.

MARTINS, V. F. AUTOMAÇÃO RESIDENCIAL USANDO PROTOCOLO MQTT, NODE- RED E MOSQUITTO BROKER COM ESP32 E ESP8266. p. 53, 2019.

MEHTA, M. ESP 8266: A BREAKTHROUGH IN WIRELESS SENSOR NETWORKS AND INTERNET OF THINGS. **International Journal of Electronics and Communication Engineering & Technology**, v. 6, n. 8, p. 7–11, ago. 2015.

OLIVEIRA, G. M. B. et al. **Comparison Between MQTT and WebSocket Protocols for IoT Applications Using ESP8266**. 2018 Workshop on Metrology for Industry 4.0 and IoT. **Anais**... In: 2018 WORKSHOP ON METROLOGY FOR INDUSTRY 4.0 AND IOT. Brescia: IEEE, abr. 2018. Disponível em: <https://ieeexplore.ieee.org/document/8428348/>. Acesso em: 3 dez. 2020

SAHADEVAN, A. et al. **An Offline Online Strategy for IoT Using MQTT**. 2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud). **Anais**... In: 2017 IEEE 4TH INTERNATIONAL CONFERENCE ON CYBER SECURITY AND CLOUD COMPUTING (CSCLOUD). New York, NY, USA: IEEE, jun. 2017. Disponível em: <http://ieeexplore.ieee.org/document/7987225/>. Acesso em: 3 dez. 2020

SHINHO LEE et al. **Correlation analysis of MQTT loss and delay according to QoS level**. The International Conference on Information Networking 2013 (ICOIN). **Anais**... In: 2013 INTERNATIONAL CONFERENCE ON INFORMATION NETWORKING (ICOIN). Bangkok: IEEE, jan. 2013. Disponível em: <http://ieeexplore.ieee.org/document/6496715/>. Acesso em: 3 dez. 2020

SONI, D.; MAKWANA, A. A SURVEY ON MQTT: A PROTOCOL OF INTERNET OF THINGS (IOT). p. 5, 2017.

TORRES, A. B. B.; ROCHA, A. R.; DE SOUZA, J. N. Análise de Desempenho de Brokers MQTT em Sistema de Baixo Custo. p. 12, 2016.