

# Tecnologias para Implementação de Laboratórios Remotos de Robótica: Revisão Sistemática da Literatura

Gustavo Yoshio Maruyama  
Universidade Federal de Mato Grosso do Sul (UFMS)  
Campo Grande, MS, Brasil  
gustavo.maruyama@ifms.edu.br

Anderson Corrêa de Lima  
Universidade Federal de Mato Grosso do Sul (UFMS)  
Campo Grande, MS, Brasil  
anderson.lima@ufms.br

Amaury Antônio de Castro Junior  
Universidade Federal de Mato Grosso do Sul (UFMS)  
Campo Grande, MS, Brasil  
amaury.junior@ufms.br

Marcos Pinheiro Vilhanueva  
Instituto Federal de Educação, Ciência e Tecnologia d  
Mato Grosso do Sul (IFMS)  
Ponta Porã, MS, Brasil  
marcos.vilhanueva@ifms.edu.br

## ABSTRACT

This paper presents a systematic literature review based on studies that approach the development or use of existing technologies and tools for the implementation of remote robotics laboratories aimed at education. We sought to answer technical and motivational questions that provide information that collaborate for the development and implementation of remote robotics laboratories in educational institutions. The aspects studied include: architecture, user interface, robotic platforms, cameras, programming languages types and motivation. The results discussed here will be used as basis for a research development, aiming at the implementation of a remote and experimental robotics laboratory at the Mato Grosso do Sul Federal Institute of Education, Science and Technology.

## KEYWORDS

Laboratório Remoto, Robótica, Educação, Revisão Sistemática da Literatura

## 1 INTRODUÇÃO

O uso da robótica na educação vem ganhando espaço em instituições de ensino fundamental, médio, técnico e superior. Diferentes estudos apontam que alunos se sentem mais motivados e engajados nos estudos, como relatado nos trabalhos [1] e [2]. Já em [3], são apontados os principais benefícios do uso da robótica na educação, também chamada de robótica pedagógica ou robótica educacional. Dentre os benefícios, destacam-se: estímulo do raciocínio lógico, auxílio na organização mental, incentivo ao aprendizado de matemática, física e língua inglesa, estímulo à criatividade e ao desenvolvimento de habilidades para solucionar problemas.

Apesar dos diversos fatores favoráveis, existem problemas que dificultam a adoção da robótica educacional nas instituições de ensino, tais como a indisponibilidade de tempo para planejamento e realização de atividades de robótica, formação teórico-prática do professores e custos relacionados a equipamentos que impactam diretamente na montagem de um laboratório específico [4]. No Brasil, por exemplo, muitas escolas públicas carecem de recursos para implantação de um laboratório robótica.

Por outro lado, no caso das instituições de ensino que possuem laboratórios específicos de robótica, podem surgir situações que impedem o acesso às dependências físicas da instituição. Por exemplo, no ano de 2020 o mundo passou a enfrentar uma grave crise

sanitária, a pandemia de COVID-19 (doença do coronavírus) [5]. No Brasil, o primeiro caso foi relatado em 26 de fevereiro de 2020, desde então a doença se espalhou rapidamente em todo território nacional, causando impactos sociais e econômicos [6]. Como medida de contenção ao vírus, instituições de ensino suspenderam o acesso às suas dependências físicas, tornando ociosos as suas salas de aula e laboratórios de ensino.

Uma alternativa que pode ser adotada por instituições de ensino que já possuem um laboratório de robótica, é a implantação de um sistema de laboratórios remotos. Um laboratório remoto permite a utilização dos recursos de um laboratório físico, mesmo em situações de suspensão ou impedimento do acesso presencial. Além disso, possibilita o compartilhamento dos recursos com instituições de ensino que não possuem laboratórios específicos de robótica.

Este trabalho apresenta uma revisão sistemática da literatura afim de responder questões técnicas e motivacionais que possam orientar e fundamentar o desenvolvimento e implantação de laboratórios remotos de robótica em instituições de ensino que já possuem um laboratório físico de robótica.

## 2 LABORATÓRIO REMOTO X LABORATÓRIO VIRTUAL

Um laboratório remoto é um sistema composto por *hardware* e *software*, permitindo que o usuário manipule, a distância, os equipamentos de um laboratório físico para realização de experimentos práticos [7]. Em um laboratório remoto, o usuário pode ter um *feedback* visual do experimento através de imagens capturadas por câmeras, ou seja, é possível acompanhar visualmente e em tempo real, todo o experimento realizado. O artigo de [8] traz um exemplo de um laboratório remoto de robótica que pode ser acessado por meio de um navegador web através da internet. Neste laboratório, o usuário pode escrever e submeter um código que será programado em um robô construído com o kit robótico Lego Mindstorm. No próprio navegador, o usuário pode visualizar o robô realizando as ações por meio da câmera instalada no laboratório físico.

Já em um laboratório virtual, existe um sistema que simula a infraestrutura de um laboratório real por meio de *softwares* [9], conhecidos como simuladores. Os laboratórios virtuais permitem que experimentos sejam realizados em diversas condições, sem a preocupação de danos aos equipamentos e sem oferecer riscos aos usuários [10] [11]. Nessas situações, é possível perceber a redução

de custos de implantação e manutenção, quando comparados a um laboratório físico [12]. No artigo de [13], são apresentados exemplos de laboratórios virtuais de robótica, com destaque ao EvaMars, um laboratório que simula um robô de treinamento em Marte. Esse é um exemplo onde o laboratório virtual se torna muito útil, pois manipular um robô real em Marte ainda é uma situação inviável à estudantes.

De acordo com [14], a interação com processos físicos reais proporcionada por um laboratório remoto, pode ser uma característica mais atraente para o estudante do que um processo simulado em um laboratório virtual. Porém, vale ressaltar que a adoção de um laboratório remoto ou um laboratório virtual depende das necessidades e situações físicas e econômicas da instituição de ensino. [12] sugere que laboratórios remotos e virtuais podem ser utilizados em conjunto, de forma complementar.

Instituições de ensino que possuem laboratórios físicos de robótica podem optar pela implantação de um laboratório remoto, justamente para fazer o melhor aproveitamento dos equipamentos disponíveis e dos investimentos já realizados. Dessa forma, os alunos e professores da instituição poderão utilizar o laboratório, mesmo em situações adversas que impeçam o acesso presencial, reduzindo assim, a ociosidade do laboratório físico. Segundo [10], também é possível compartilhar o laboratório com instituições de ensino municipais, estaduais e particulares que não possuem laboratório de robótica, ampliando a colaboração em ações educacionais entre as instituições. Neste contexto, algumas ações podem ser realizadas pela a instituição anfitriã do laboratório, como cursos, capacitações, treinamentos e oficinas a distância.

### 3 METODOLOGIA

Esta revisão sistemática da literatura foi realizada seguindo as etapas: definição de objetivos (subseção 3.1), definição das questões de pesquisa (subseção 3.2), definição de cadeia de busca e escolha de fontes de pesquisa (subseção 3.3), definição de critérios de inclusão e exclusão de estudos (subseção 3.4), os resultados são apresentados na seção 4 e as respostas das questões de pesquisa se encontram na seção 5.

De acordo com [15]:

“Uma revisão sistemática da literatura é um meio de identificar, avaliar e interpretar todas as pesquisas disponíveis relevantes para uma questão de pesquisa específica, ou área de tópico ou fenômeno de interesse.”

#### 3.1 Objetivo

Identificar trabalhos que abordem o desenvolvimento ou a utilização de tecnologias e ferramentas existentes para implementação de laboratórios remotos de robótica voltados à educação.

#### 3.2 Questões de Pesquisa

As questões de pesquisa foram formuladas com foco na coleta de dados e informações relacionadas a aspectos técnicos. Os fatores pedagógicos serão explorados em trabalhos futuros, a partir da implementação de um projeto piloto. As questões definidas são as seguintes:

**Q1:** Existe alguma arquitetura que é comumente empregada para a implementação de laboratórios remotos?

**Q2:** Quais são os protocolos de comunicação envolvidos?

**Q3:** Qual tipo de interface de aplicação é mais utilizada em laboratórios remotos?

**Q4:** Quais plataformas foram utilizadas para o desenvolvimento do robô ou do circuito a ser manipulado no ambiente remoto?

**Q5:** Qual o tipo de linguagem de programação mais utilizado para programar o robô ou a plataforma remota: textual ou em blocos?

**Q6:** Quais modelos de câmeras foram utilizados para monitoramento visual do laboratório e transmissão das imagens aos usuários?

**Q7:** Quais as principais motivações para utilizar um laboratório remoto?

### 3.3 Estratégia de Busca

Como se trata de um tema que abrange tanto a área de computação como a área de educação, optou-se por ferramentas de busca que indexam as diversas fontes de publicações. Neste caso, foram utilizados o Google Acadêmico e também o Portal de Periódico da CAPES. A seguinte cadeia de busca foi utilizada para a realização da pesquisa:

“Remote” AND “Robotics” AND (“Laboratory” OR “Environment”) AND (“Education” OR “Educational” OR “Learning” OR “Teaching”).

Foi aplicado também, um filtro para selecionar trabalhos publicados entre os anos de 2016 e 2021. A busca retornou aproximadamente 33.700 resultados pela ferramenta Google Acadêmico e 8.792 resultados pelo Portal de Periódicos da CAPES. Foram pré selecionados os 200 trabalhos mais relevantes conforme os critérios de ordenação aplicados por cada uma das ferramentas de busca utilizada, sendo 100 trabalhos retornados pelo Google Acadêmico e 100 trabalhos retornados pelo Portal de Periódicos da CAPES.

Os trabalhos pré selecionados passaram por uma análise de título e resumo, buscando identificar relação com o objetivo proposto na subseção 3.1, no final foram obtidos 13 trabalhos para serem analisados na íntegra.

### 3.4 Critérios de Inclusão e Exclusão

Os trabalhos foram selecionados conforme os critérios de inclusão e exclusão definidos na Tabela 1. Os estudos redundantes são estudos semelhantes, do mesmo autor, que foram publicados em lugares diferentes. Estudos secundários são estudos que trazem revisões da literatura. Literatura cinza são os trabalhos que não possuem controle de publicação e não são revisados por pares. Dos 13 artigos pré-selecionados, 8 trabalhos atenderam totalmente o critério de inclusão, passando pelos critérios de exclusão sem ressalvas.

## 4 RESULTADOS

Na Tabela 2 são mostrados os trabalhos selecionados, a fonte de cada trabalho e a ferramenta de busca utilizada. No total são dois trabalhos de 2016, três de 2017, dois de 2018 e um de 2020. Os próximos parágrafos desta seção trazem breves descrições dos trabalhos selecionados.

Critério	Tipo
Estudos primários que possuem relação com o objetivo deste estudo	Inclusão
Estudos redundantes	Exclusão
Estudos secundários	Exclusão
Literatura cinza	Exclusão

Tabela 1: Critérios de inclusão e exclusão.

Ferramenta de busca	Fonte	Trabalho
Google Acadêmico	IEEE Xplore	[14]
		[16]
		[8]
		[17]
	ScienceDirect	[18]
	IATED	[19]
	SBIE	[9]
Portal de Periódicos da CAPES	MDPI	[20]

Tabela 2: Trabalhos selecionados com a fonte e a ferramenta de busca utilizada.

Em [18] é apresentado o desenvolvimento de um laboratório remoto onde o usuário pode programar uma placa Arduino a distância. No desenvolvimento os autores focaram na usabilidade, desenvolvendo dois tipos de interface, uma apropriada para programação baseada em texto que possui a interface semelhante ao da IDE do Arduino e outra interface para programação em blocos, que foi desenvolvida utilizando a biblioteca Blockly da Google. As interfaces de usuário são responsivas e podem ser acessadas por meio de navegador *web* ou por celular. A comunicação entre cliente e servidor é realizada por meio de *websockets*. A aplicação servidora é executada em um computador Raspberry PI que é responsável por receber comandos e também por receber o código, compilar e enviar ao Arduino através da porta USB. Este laboratório foi avaliado em ambiente escolar, participaram 18 usuários, que inclui 13 alunos do ensino médio de 14 a 16 anos e 5 professores do ensino médio e técnico de 25 a 38 anos. A pesquisa mostra que os recursos oferecidos, como conjunto de componentes, monitoramento por vídeo e a interface gráfica são satisfatórios, porém, as disposições das informações precisam melhorar.

O trabalho de [20] traz um exemplo muito interessante, pois combina um laboratório remoto com um laboratório virtual. Neste laboratório é possível programar um robô físico para evitar obstáculos virtuais. Uma vantagem observada nessa abordagem é que os obstáculos podem ser montados sem intervenção humana no laboratório físico, pois se tratam de obstáculos virtuais. Este laboratório foi desenvolvido com a linguagem PHP e utiliza um mini computador BeagleBoard-xM para receber o código e comandos do cliente e controlar o robô Pioneer P3-DX. O monitoramento visual do robô é feito por uma câmera IP. O laboratório foi avaliado por 19 alunos do departamento de engenharia de sistemas de informação da Universidade Kocaeli (Turquia). Os resultados da pesquisa

mostram que o laboratório permite que os alunos aprendam a solucionar problemas de desvio de obstáculos com algoritmos genéticos de forma rápida, eficaz e eficiente.

[19] propõem a inclusão de robótica educacional como ferramenta pedagógica em áreas rurais do Equador por meio de laboratório remoto. O laboratório remoto apresentado é denominado “*HARDWARE ANYWHERE*” e utiliza a placa Arduino Uno como objeto de experimento, acompanhado de diversos componentes (servo-motor, sensor de temperatura, sensor ultrassônico, sensor de luminosidade, *display lcd*, relé e soquete de lâmpada). A interface de programação é baseada em programação em blocos e foi utilizada a biblioteca Blockly da Google. O público alvo do laboratório são alunos do ensino básico de 7 a 11 anos.

O trabalho de [9] traz o desenvolvimento de um laboratório remoto composto por uma aplicação cliente web, servidor e a plataforma robótica. A programação do robô é realizada utilizando programação em blocos, os autores utilizaram uma adaptação do ambiente Snap! para fornecer a interface de programação em blocos. Este laboratório remoto de robótica se destaca por suas características de adaptabilidade e extensibilidade. O caráter adaptável do sistema se justifica, pois, é possível adaptar o sistema para diferentes públicos no que tange o nível de conhecimento de programação e robótica. O caráter extensível do sistema é caracterizado por permitir que o usuário mais avançado crie seus próprios blocos de programação.

Em [8] o objetivo foi desenvolver um laboratório remoto de robótica para auxiliar no ensino de programação e robótica. O laboratório remoto apresentado tem um caráter pedagógico, pois além de permitir a programação de um robô remotamente, também traz desafios para os alunos solucionarem. Este laboratório possui um computador do tipo PC que faz o papel de servidor, ele é responsável por realizar o intermédio entre a interface de usuário e um robô construído na plataforma Lego Mindstorm NXT. A comunicação entre servidor e robô é feita através de Bluetooth. A interface de usuário é integrada ao sistema Moodle e a programação do robô é feita utilizando a linguagem Python. O monitoramento visual do robô é feito por uma câmera IP. O laboratório remoto foi testado por 14 alunos com idades entre 12 a 17 anos do Colégio de Aplicação da Universidade Federal de Roraima (CAP/UFRR). Um dado interessante levantado foi que uso do kit robótico facilitou o aprendizado da programação de computadores, 71% dos usuários concordaram totalmente e 29% concordaram parcialmente.

[17] apresenta um laboratório remoto de robótica que está disponível no sistema WebLab-Deusto, que é um sistema web de gerenciamento de laboratórios remotos. O laboratório em questão utiliza o robô Zumo 32u4 Pololu, que é baseado na plataforma Arduino. Também é utilizado um minicomputador Raspberry PI 3 que permite a programação remota do robô. A interface de usuário utiliza a biblioteca Blockly do Google, para que o usuário programe o robô utilizando programação em blocos. O monitoramento visual do robô é feito por uma webcam. Um ponto interessante levantado nesse trabalho é que o custo de um laboratório de robótica pode ser muito elevado, é citado o exemplo do custo de um kit Lego MindStorm, chegando a 350 euros, um custo elevado até para os países da União Europeia. O autor ressalta que o laboratório remoto não deve substituir o laboratório físico, mas sim, ser utilizado como complemento.

O trabalho de [16] aborda o desenvolvimento de um sistema composto por um laboratório remoto de robótica, uma ferramenta de programação online e um ambiente virtual de aprendizagem. O laboratório remoto conta com um robô de baixo custo que utiliza a placa Arduino Mega e um shield RF (radiofrequência) Pololu Wixel2 para se comunicar com o servidor do laboratório, é alimentado por uma fonte DC conectado por fio suspenso para não prejudicar a movimentação. O servidor do laboratório é um minicomputador CubieBoard que é responsável por receber o código, compilar e enviar ao robô por RF. Já a aplicação cliente é web, baseado em páginas dinâmicas e foi implementada em PHP 5, esta aplicação é executada por um servidor web Apache e ainda possui um chat para que os usuários possam conversar. O envio de comandos e códigos para o servidor do laboratório é feito por protocolo HTTP, já o feedback do console é feito por meio da biblioteca socket.io que é baseada em *websockets*. O monitoramento visual do laboratório é feito por câmera IP.

Em [14] é apresentado um sistema muito interessante, onde o aluno tem acesso a um laboratório remoto que disponibiliza vários problemas. O aluno pode visualizar todos os problemas, porém só pode realizar os problemas em que o aluno atenda os pré-requisitos ou se a competência do aluno é muito próxima ao que é exigido no problema, dessa forma evita-se que o aluno tenha possíveis frustrações caso ele tente resolver um problema muito complexo que esteja fora de sua competência atual. É um exemplo de sistema adaptativo. O autor relata que utilizou braço robótico ligado a uma placa Arduino e também robôs construídos com Lego Mindstorm. O foco do trabalho não é a arquitetura em si, mas sim o modelo adaptativo desenvolvido, portanto ele não descreve detalhes de protocolos, troca de mensagens e nem como o laboratório funciona fisicamente.

## 5 DISCUSSÃO E ANÁLISE DOS RESULTADOS

Nesta seção são apresentadas as respostas para as questões de pesquisa. As respostas foram elaboradas com base na análise dos estudos selecionados.

### Q1 - Existe alguma arquitetura que é comumente empregada?

[18], [20], [8], [17] e [16] trazem arquiteturas muito semelhantes. Para tentar abstrair em um único modelo, foi desenhado o esquema mostrado na Fig. 1, que inclusive serve de sugestão para trabalhos relacionados que vierem a ser desenvolvidos. Todos os trabalhos citados possuem um computador ou minicomputador intermediário no laboratório físico (Fig. 1-A) que é responsável por receber, de uma servidor web (Fig. 1-B), dados, comandos ou código para ser compilado e enviado ao robô (Fig. 1-C). [18] e [17] utilizam o minicomputador Raspberry PI, [16] utiliza o minicomputador CubieBoard, [20] utiliza o minicomputador BeagleBoard e [8] utiliza um computador de mesa. Minicomputadores são opções interessantes, pois normalmente possuem um valor menor se comparado um computador de mesa. A aplicação servidora (Fig. 1-B) fornece páginas para o cliente web e também pode fornecer API (*Application Programming Interface* ou Interface de Programação de Aplicação) para gerenciamento de uso do laboratório como abordado em [8], que possui funcionalidades para agendamento de uso do laboratório.

A interface de cliente pode ser acessado por navegador web ou um aplicativo mobile (Fig. 1-D). O trabalho de [18] fornece as duas interfaces de usuário. O monitoramento visual do laboratório pode ser feito por meio de uma câmera IP (Fig. 1-E), como escolhido por [20], [8] e [16], ou por uma *webcam* (Fig. 1-F) como escolhido por [17].

### Q2 - Quais os protocolos de comunicação envolvidos?

Como todos os trabalhos mencionam o uso de interface de usuário web, pressupõe-se a utilização do protocolo HTTP [21] por ser o padrão em navegadores web, mas nenhum deles menciona se seguem as restrições REST (*REpresentational State Transfer*). [18] utiliza o protocolo *websocket* [22] para troca de dados entre servidor e cliente, este protocolo é interessante para troca de dados em tempo real, pois é um protocolo bidirecional e permite obter características do protocolo TCP na camada de aplicação, isso é muito útil, pois navegadores web não possuem APIs para utilização direta do protocolo TCP [23], [16] utiliza a biblioteca Socket.IO que é baseado no protocolo *websocket* [24] para *feedback* do console do Arduino no navegador.

Infelizmente os estudos não trazem informação detalhada sobre protocolos ou técnicas relacionados a *stream* de vídeo, importante para monitoramento visual do laboratório por meio de câmeras.

### Q3 - Qual tipo de interface de aplicação é mais utilizada em laboratórios remotos?

Predominantemente todos os trabalhos selecionados fornecem interface de usuário web como pode ser visto na Tabela 3. Algumas interfaces são integradas ao sistema Moodle, como é o caso de [8] e de [16]. [18] cita que possui duas interfaces de usuário, uma *web* e outra *mobile*, porém a *mobile* necessita de ajustes. Nenhum trabalho traz uma interface *desktop*, isso deve-se ao fato que uma interface *web* permite que a aplicação funcione em qualquer computador que tenha um navegador *web* instalado. Uma aplicação *desktop* por sua vez, pode exigir um sistema operacional específico ou exigir a instalação ou configuração de ferramentas adicionais, então corre-se mais risco de incompatibilidade com o computador do usuário.

Trabalho	Tipo de interface de aplicação		
	Web	Mobile	Desktop
[18]	x	x	
[20]	x		
[19]	x		
[9]	x		
[8]	x		
[17]	x		
[16]	x		
[14]	x		
<b>Total</b>	<b>8</b>	<b>1</b>	<b>0</b>

Tabela 3: Distribuição dos trabalhos por tipo de interface de aplicação.

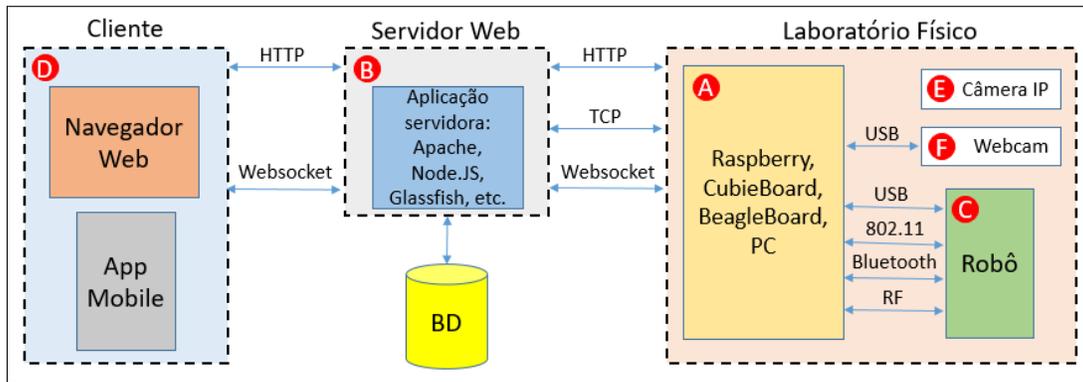


Figura 1: Esquema da arquitetura baseada nos trabalhos selecionados.

#### Q4 - Quais plataformas foram utilizadas para o desenvolvimento do robô ou do circuito a ser manipulado?

As plataformas utilizadas nos trabalhos analisados foram: Arduino, Lego Mindstorm, Pionner P3-DX e Zumo 32u4 Pololu que é baseado em Arduino. Na Tabela 4 é mostrada a distribuição das dos trabalhos por plataforma. É possível ver que a plataforma Arduino foi a mais utilizada, seguida da plataforma Lego Mindstorm.

Um fator que pode contribuir pela preferência da plataforma Arduino é o seu preço acessível. O preço relatado por [18] de uma placa Arduino Uno é de aproximadamente U\$ 25,00 (dólares), enquanto um kit lego pode custar cerca de € 350,00 (euros) como relata [17], então uma placa Arduino se torna mais acessível. Embora deva-se considerar os demais componentes para se montar um robô educacional, ainda assim é possível construir um robô educacional de baixo custo utilizando Arduino, como foi feito no trabalho de [16].

Trabalho	Plataforma			
	Arduino	Lego	P3-DX	Zumo 32u4
[18]	x			
[20]			x	
[19]	x			
[9]	x			
[8]		x		
[17]				x
[16]	x			
[14]	x	x		
<b>Total</b>	<b>5</b>	<b>2</b>	<b>1</b>	<b>1</b>

Tabela 4: Distribuição dos trabalhos por plataforma

#### Q5 - Qual o tipo de linguagem de programação mais utilizado para programar o robô ou a plataforma? Textual ou em Blocos?

A programação em blocos tem finalidade pedagógica e diferentemente da programação com linguagem textual como C, traz as

estruturas de programação em formato visual, cada estrutura é chamada de bloco. A programação é feita por meio de ações “arrasta e solta” de blocos. [9], com base em estudos analisados em robótica educacional, afirma que a programação em blocos fornece um ambiente mais intuitivo e simples para construção do conhecimento. Apesar de menos intuitivas, linguagens baseadas em texto são utilizadas na prática no desenvolvimentos softwares e correspondem a realidade profissional e acadêmica.

Na Tabela 5 são mostrados os trabalhos que utilizam linguagem textual ou em blocos. [18], [19] e [17] trazem interfaces de programação em blocos, desenvolvidas com a biblioteca Blockly da Google. Em [9] foi realizada uma adaptação do ambiente de programação em blocos Snap! criado por Jens Möning e Brian Harvey na Universidade da Califórnia em Berkeley [25]. [20], [8] e [16] optaram por linguagens textuais.

Trabalho	Tipo de Linguagem	
	Textual	Blocos
[18]	x	x
[20]	x	
[19]		x
[9]		x
[8]	x	
[17]		x
[16]	x	
[14]	N/D	N/D
<b>Total</b>	<b>4</b>	<b>4</b>

Tabela 5: Distribuição dos trabalhos por tipo de linguagem (textual ou blocos)

#### Q6 - Quais tipos de câmeras foram utilizados para monitoramento visual do laboratório?

Na Tabela 6 é possível ver que câmeras IP são mais adotadas no monitoramento visual do laboratório. Os trabalhos selecionados não trazem os motivos da escolha, porém supõe-se que a adoção de câmeras IP é devido a praticidade de uso.

Trabalhos	Tipo de Câmera	
	Câmera IP	Webcam
[18]		x
[20]	x	
[19]	N/D	N/D
[9]	N/D	N/D
[8]	x	
[17]		x
[16]	x	
[14]	N/D	N/D
<b>Total</b>	<b>3</b>	<b>2</b>

Tabela 6: Distribuição dos trabalhos por tipo de câmera

### Q7 - Quais as motivações para utilizar um laboratório remoto?

No estudo de [14] afirma-se que é mais atraente para o usuário interagir com processos físicos reais do que simplesmente controlar simulações de *software*. [18] cita que um laboratório remoto pode resolver o problema de ociosidade e disponibilidade de recursos, ou seja, se a instituição já possui os recursos físicos de um laboratório de robótica, um sistema de laboratório remoto pode permitir o compartilhamento desses recursos com outras instituições, diminuindo a ociosidade e aumentando a disponibilidade dos recursos.

## 6 CONCLUSÃO

Laboratórios remotos podem ser úteis em situações que inviabilizem o acesso às dependências físicas das instituições de ensino, como foi o caso ocorrido durante a pandemia de COVID-19, que levou muitas instituições de ensino a suspenderem o acesso presencial, tornando os laboratórios ociosos. Além disso, os laboratórios remotos permitem que os recursos presentes em um laboratório físico sejam compartilhados com outras instituições que não possuem laboratórios específicos.

As questões respondidas neste trabalho ajudarão na tomada de decisão sobre os componentes arquiteturais, protocolos, linguagens, plataformas e ferramentas a serem empregados no desenvolvimento e implantação de um projeto piloto do laboratório remoto de robótica no Instituto Federal de Educação, Ciência e Tecnologia de Mato Grosso do Sul (IFMS). Informações relevantes foram levantadas, como por exemplo, a utilização de minicomputadores que tem o objetivo de receber o código fonte, compilar e enviar o programa ao robô. Isso se torna interessante, pois esses minicomputadores têm valores menores se comparados a um computador de mesa.

A revisão sistemática apresentada neste trabalho pode servir como um estudo inicial para outras instituições de ensino que possuem laboratórios de robótica e desejam disponibilizá-los remotamente.

## REFERÊNCIAS

- [1] José Vidal, Fábio Sampaio, and Nuno Dorotea. Um estudo exploratório sobre o uso da robótica educacional como ferramenta de apoio ao ensino-aprendizagem de lógica de programação para alunos da rede pública do ensino médio. In *Anais do Simpósio Brasileiro de Educação em Computação*, pages 280–289, Porto

- Alegre, RS, Brasil, 2021. SBC. doi: 10.5753/educomp.2021.14495. URL <https://sol.sbc.org.br/index.php/educomp/article/view/14495>.
- [2] André Rachman Dargains and Fábio Ferrentini Sampaio. Estudo exploratório sobre o uso da robótica educacional no ensino de introdução a programação. *Tecnologias, Sociedade e Conhecimento*, 7(1):71–96, ago. 2020. doi: 10.20396/tsc.v7i1.14702. URL <https://econtents.bc.unicamp.br/impec/index.php/tsc/article/view/14702>.
- [3] Wagner Bandeira Andriola. Impactos da robótica no ensino básico: estudo comparativo entre escolas públicas e privadas. *Ciência & Educação (Bauru)*, 27, 2021.
- [4] Flavio Rodrigues Campos. Robótica educacional no brasil: questões em aberto, desafios e perspectivas futuras. *Revista ibero-americana de estudos em educação*, 12(4):2108–2121, 2017.
- [5] Jay J. Van Bavel, Katherine Baicker, Paulo S. Boggio, Valerio Capraro, Aleksandra Cichocka, Mina Cikara, Molly J. Crockett, Alia J. Crum, Karen M. Douglas, James N. Druckman, John Drury, Oeindrila Dube, Naomi Ellemers, Eli J. Finkel, James H. Fowler, Michele Gelfand, Shihui Han, S. Alexander Haslam, Jolanda Jetten, Shinobu Kitayama, Dean Mobbs, Lucy E. Napper, Dominic J. Packer, Gordon Pennycook, Ellen Peters, Richard E. Petty, David G. Rand, Stephen D. Reicher, Simone Schnall, Azim Shariff, Linda J. Skitka, Sandra Susan Smith, Cass R. Sunstein, Nassim Tabri, Joshua A. Tucker, Sander van der Linden, Paul van Lange, Kim A. Weeden, Michael J. A. Wohl, Jamil Zaki, Sean R. Zion, and Robb Willer. Using social and behavioural science to support covid-19 pandemic response. *Nature Human Behaviour*, 4(5):460–471, May 2020. ISSN 2397-3374. doi: 10.1038/s41562-020-0884-z. URL <https://doi.org/10.1038/s41562-020-0884-z>.
- [6] Carlos Souza, João Paiva, Thiago Leal, Leonardo Silva, and Lucas Santos. Spatiotemporal evolution of case fatality rates of covid-19 in brazil, 2020. *J Bras Pneumol*, 46:e20200208–e20200208, 2020. URL <http://www.jornaldepneumologia.com.br/details/3362>.
- [7] Pablo Orduña, Luis Rodriguez-Gil, Javier Garcia-Zubia, Ignacio Angulo, Unai Hernandez, and Esteban Azcuenaga. Labsland: A sharing economy platform to promote educational remote laboratories maintainability, sustainability and adoption. In *2016 IEEE Frontiers in Education Conference (FIE)*, pages 1–6. IEEE, 2016.
- [8] Thais Oliveira Almeida, José Francisco de M. Netto, and Marcel Leite Rios. Remote robotics laboratory as support to teaching programming. In *2017 IEEE Frontiers in Education Conference (FIE)*, pages 1–6, 2017. doi: 10.1109/FIE.2017.8190472.
- [9] Crijina Flores, Artur Kronbauer, and Jorge Campos. Lero—an extensible and adaptive remote lab for educational robotics. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 29, page 525, 2018.
- [10] Luís Gomes and Seta Bogosyan. Current trends in remote laboratories. *IEEE Transactions on Industrial Electronics*, 56(12):4744–4756, 2009. doi: 10.1109/TIE.2009.2033293.
- [11] Mu Lin, Lijun San, and Yu Ding. Construction of robotic virtual laboratory system based on unity3d. *IOP Conference Series: Materials Science and Engineering*, 768(7):072084, mar 2020. doi: 10.1088/1757-899x/768/7/072084. URL <https://doi.org/10.1088/1757-899x/768/7/072084>.
- [12] Z. Nedic, J. Machotka, and A. Nafalski. Remote laboratories versus virtual and real laboratories. In *33rd Annual Frontiers in Education, 2003. FIE 2003.*, volume 1, pages T3E–T3E, 2003. doi: 10.1109/FIE.2003.1263343.
- [13] Alim Kerem Erdoğan and Ugur Yayan. Virtual robotic laboratory compatible mobile robots for education and research. In *2021 International Conference on Innovations in Intelligent Systems and Applications (INISTA)*, pages 1–6, 2021. doi: 10.1109/INISTA52262.2021.9548503.
- [14] Paolo Di Giamberardino and Marco Temperini. Adaptive access to robotic learning experiences in a remote laboratory setting. In *2017 18th International Carpathian Control Conference (ICCC)*, pages 565–570, 2017. doi: 10.1109/CarpathianCC.2017.7970464.
- [15] Barbara Kitchenham and Stuart Charters. Guidelines for performing systematic literature reviews in software engineering. Technical Report EBSE 2007-001, Keele University and Durham University Joint Report, 2007. URL <http://www.dur.ac.uk/ebse/resources/Systematic-reviews-5-8.pdf>.
- [16] Maisa Soares Santos Lopes, Iago Pacheco Gomes, Roque MP Trindade, Alzira F da Silva, and Antonio C de C Lima. Web environment for programming and control of a mobile robot in a remote laboratory. *IEEE Transactions on Learning Technologies*, 10(4):526–531, 2016.
- [17] Ignacio Angulo, Javier Garcia-Zubia, Unai Hernández-Jayo, Iñigo Uriarte, Luis Rodriguez-Gil, Pablo Orduña, and Gabriel Martínez Pieper. Roboblock: A remote lab for robotics and visual programming. In *2017 4th Experiment@ International Conference (exp. at'17)*, pages 109–110. IEEE, 2017.
- [18] João Paulo Cardoso Lima, Lucas Mellos Carlos, José Pedro Scharidosim Simão, Josiel Pereira, Paulo Manoel Mafra, and Juarez Bento da Silva. Design and implementation of a remote lab for teaching programming and robotics. *IFAC-PapersOnLine*, 49(30):86–91, 2016. ISSN 2405-8963. doi: <https://doi.org/10.1016/j.ifacol.2016.11.133>. URL <https://www.sciencedirect.com/science/article/pii/S2405896316325848>. 4th IFAC Symposium on Telematics Applications TA 2016.

- [19] L. Renteria, C. Orozco, J. Jinez, B. Suquilanda, and M. Rodriguez. Educational robotics through the use of remote laboratories. In *EDULEARN18 Proceedings*, 10th International Conference on Education and New Learning Technologies, pages 10944–10950. IATED, 2-4 July, 2018 2018. ISBN 978-84-09-02709-5. doi: 10.21125/edulearn.2018.2694. URL <http://dx.doi.org/10.21125/edulearn.2018.2694>.
- [20] Solak Serdar, Yakut Önder, and Dogru Bolat Emine. Design and implementation of web-based virtual mobile robot laboratory for engineering education. *Symmetry*, 12(6), 2020. ISSN 2073-8994. doi: 10.3390/sym12060906. URL <https://www.mdpi.com/2073-8994/12/6/906>.
- [21] Roy Fielding, Jim Gettys, Jeffrey Mogul, Henrik Frystyk, Larry Masinter, Paul Leach, and Tim Berners-Lee. Hypertext Transfer Protocol – HTTP/1.1. RFC 2616, RFC Editor, 1999. URL <https://datatracker.ietf.org/doc/html/rfc2616>.
- [22] Ian Fette and Alexey Melnikov. The WebSocket Protocol. RFC 6455, RFC Editor, 2011. URL <https://datatracker.ietf.org/doc/html/rfc6455>.
- [23] Vanessa Wang, Frank Salim, and Peter Moskovits. *The WebSocket Protocol*, pages 33–60. Apress, Berkeley, CA, 2013. ISBN 978-1-4302-4741-8. doi: 10.1007/978-1-4302-4741-8\_3. URL [https://doi.org/10.1007/978-1-4302-4741-8\\_3](https://doi.org/10.1007/978-1-4302-4741-8_3).
- [24] Socket.IO. Documentation. <https://socket.io/docs/v4/>, 2021. URL <https://socket.io/docs/v4/>. Acesso em: 28/11/2021.
- [25] Bernat Romagosa i Carrasquer. *The Snap! Programming System*, pages 1–10. Springer International Publishing, Cham, 2019. ISBN 978-3-319-60013-0. doi: 10.1007/978-3-319-60013-0\_28-2. URL [https://doi.org/10.1007/978-3-319-60013-0\\_28-2](https://doi.org/10.1007/978-3-319-60013-0_28-2).