

Acesso Compartilhado à Recursos Físicos com Google Colab

Carlos Eduardo Sabino¹, José A. M. Nacif¹, Ricardo Ferreira¹

carlos.sabino@ufv.br, jnacif@ufv.br, ricardo@ufv.br

¹ Universidade Federal de Vicosa

Este trabalho apresenta uma proposta para proporcionar aos alunos acesso a recursos de hardware para execução de projetos, mesmo em contextos onde a disponibilidade de tais recursos é limitada. Apresentamos um ambiente virtual de programação utilizando o Google Colab. Os alunos podem submeter seus códigos para execução em um recurso de hardware compartilhado, permitindo-lhes observar a execução em tempo real em um ambiente físico real e tangível. Esta abordagem não apenas facilita o acesso a recursos físicos com disponibilidade limitada, como reduz custos e motiva os estudantes com a visualização física dos resultados.

A validação de software em hardware tangível, como o controle de um robô, a geração de sons ou efeitos luminosos em LEDs, permite aos estudantes visualizarem os resultados de atividades práticas no ensino de sistemas lógicos, robótica, e até mesmo em práticas de laboratório de física ou química. No entanto, o custo associado à aquisição desses recursos físicos pode ser elevado, tornando inviável para as escolas disponibilizar um kit para cada estudante ou grupo de estudantes. Além disso, isso dificulta o acesso remoto, onde os alunos poderiam realizar testes em casa ou permitir ao professor realizar demonstrações durante aulas teóricas, com acesso e interação compartilhados pelos alunos.

O Google Colab opera diretamente no navegador da internet, permitindo a integração de documentação e código em um único ambiente. Ele tem sido amplamente adotado como uma ferramenta de ensino e pesquisa de laboratórios de software [2, 5–9]. Considerando o aspecto físico para laboratórios de hardware, embora trabalhos recentes tenham apresentado soluções para laboratórios remotos [11, 13], a implementação dessas soluções pode representar uma barreira para professores e educadores sem um conhecimento aprofundado de redes de computadores e programação web.

O uso de recursos remotos no ensino foi amplamente adotado durante a pandemia do Covid-19, como evidenciado em domínios específicos, como laboratórios de Arduino e Robótica [4] e hardware reconfigurável com placas de FPGA [1]. Nossa proposta visa oferecer um ambiente genérico para essa abordagem. Em nosso modelo, cada estudante desenvolve seu código no Colab e, ao alcançar a etapa final, submete-o para execução em um recurso de hardware compartilhado.

Diante dos desafios de controle de concorrência, quando múltiplos alunos buscam acessar o hardware simultaneamente, buscamos um sistema robusto e, ao mesmo tempo, simples de ser configurado

pelos professores e alunos. Nosso objetivo é evitar conflitos, oferecer um baixo tempo de resposta e proporcionar uma experiência equitativa para todos os usuários.

A implementação faz uso do protocolo MQTT [3, 14], muito popular na comunidade de Internet das Coisas com a biblioteca PAHO, a ferramenta Node-RED [10, 12] e o Google Colab. O controle é encapsulado em tópico MQTT que gerencia a concorrência e organiza a distribuição do tempo de uso do hardware. A vantagem do MQTT é a portabilidade da solução e sua simplicidade pois toda a comunicação usa tópicos. Outra vantagem dos tópicos é permitir o monitoramento para adicionar outras funcionalidades. Para o hardware, o uso do MQTT também é vantajoso pois existem muitos exemplos com controle utilizando troca de mensagens através do protocolo MQTT.

O uso do Colab facilita o trabalho do professor, que pode desenvolver várias interfaces diferentes usando as funcionalidades e a disponibilidade de vários exemplos de uso de recursos para interface disponíveis no Colab.

Inicialmente, o aluno solicita acesso ao hardware compartilhado. O controlador processa essa solicitação, respondendo ao aluno com uma mensagem que indica se o acesso foi concedido ou informando a posição na fila de espera em caso de concorrência. Após a concessão de acesso, o aluno pode enviar suas instruções diretamente para o controlador.

Para garantir confiabilidade, o sistema foi projetado para lidar com interrupções, realocar recursos em caso de falhas e manter um registro consistente das operações, assegurando que nenhum aluno seja prejudicado por eventos inesperados.

Durante a avaliação do protótipo, testamos diversos exemplos para evitar situações de inanição e deadlock. Além disso, investigamos o impacto da qualidade da conexão na eficácia da arquitetura. Sob uma conexão robusta, observamos tempos de execução menores (2-3 segundos), enquanto em um cenário de Wi-Fi menos estável, esses tempos variaram de 3 a 5 segundos. Dois experimentos foram montados para teste de um código na Linguagem de Hardware Verilog que executava no Colab e enviavam a saída da execução para uma placa ESP32 que mostrava o resultado em Leds e em um painel P10. O experimento foi demonstrado em um aula com 40 alunos e na escola ESSE do SBESC2023 para uma plateia com 30 pessoas. Trabalhos futuros incluirão o desenvolvimento de exemplos em diversas áreas e o uso de hardware variado para promover a ferramenta. A limitação atual é o uso do MQTT no hardware compartilhado. Entretanto, com a grande disponibilidade de dispositivos com IoT como arduinos, FPGA e similares, deixa de ser uma limitação devido ao baixo custo e popularidade do hardware. O conhecimento de programação é de nível iniciante/intermediário para configurar um hardware com suporte para MQTT. O material se encontra disponível no [Repositório com um Google Colab](#) ou contactando ricardo@ufv.br.

Fica permitido ao(s) autor(es) ou a terceiros a reprodução ou distribuição, em parte ou no todo, do material extraído dessa obra, de forma verbatim, adaptada ou remixada, bem como a criação ou produção a partir do conteúdo dessa obra, para fins não comerciais, desde que sejam atribuídos os devidos créditos à criação original, sob os termos da licença CC BY-NC 4.0.

EduComp'24, Abril 22-27, 2024, São Paulo, São Paulo, Brasil (On-line)

© 2024 Copyright mantido pelo(s) autor(es). Direitos de publicação licenciados à Sociedade Brasileira de Computação (SBC).

REFERÊNCIAS

- [1] Villar-Martínez Aitor, Javier García-Zubía, Ignacio Angulo, and Luis Rodríguez-Gil. 2022. Toward Widespread Remote Laboratories: Evaluating the Effectiveness of a Replication-Based Architecture for Real-World Multiinstitutional Usage. *IEEE Access* 10 (2022), 86298–86317.
- [2] Marijan Beg, Juliette Taka, Thomas Kluyver, Alexander Konovalov, Min Ragan-Kelley, Nicolas M Thiéry, and Hans Fangohr. 2021. Using Jupyter for reproducible scientific workflows. *Computing in Science & Engineering* 23, 2 (2021), 36–46.
- [3] Melvin Bender, Erkin Kirdan, Marc-Oliver Pahl, and Georg Carle. 2021. Open-source mqtt evaluation. In *2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 1–4.
- [4] Paola Andrea Buitrago, Raúl Camacho, Harold Esneider Pérez, Oscar Jaramillo, Aitor Villar-Martínez, Luis Rodríguez-Gil, and Pablo Orduna. 2021. Mobile Arduino Robot Programming Using a Remote Laboratory in UNAD: Pedagogic and Technical Aspects. In *Cross Reality and Data Science in Engineering*, Michael E. Auer and Dominik May (Eds.). Springer International Publishing, Cham, 171–183.
- [5] Michael Canesche, Lucas Bragança, Omar Paranaíba Vilela Neto, Jose A Nacif, and Ricardo Ferreira. 2021. Google Colab CAD4U: Hands-on Cloud Laboratories for Digital Design. In *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 1–5.
- [6] Ricardo Ferreira, Michael Canesche, and Jerônimo Penha. 2023. Google Colab para Ensino de Computação. In *Anais Estendidos do III Simpósio Brasileiro de Educação em Computação*. SBC, 46–47.
- [7] Ricardo Ferreira, Peter Jamieson, Michael Canesche, Omar Paranaíba V Neto, and José Augusto Nacif. 2024. Examples and tutorials on using Google Colab and Gradio to create online interactive student-learning modules. *Computer Applications in Engineering Education* (2024), DOI: 10.1002/cae.22729.
- [8] Ricardo Ferreira, Jose Nacif, and Michael Canesche. 2023. Ideias para o uso do Google Colab no Ensino de Computação. <https://colab.research.google.com/drive/1-j6L3enqkGJs-dujGpDjtfBwF-1Tyop?usp=sharing>. [Online].
- [9] Ricardo Ferreira, Jose Nacif, and Michael Canesche. 2023. Video - Ideias para o uso do Google Colab no Ensino de Computação. <https://youtu.be/T9K0eep3JWQ>. [Online].
- [10] Ricardo Ferreira, Carlos Sabino, Michael Canesche, Omar Paranaíba V Neto, and José Augusto Nacif. 2024. AIoT tool integration for enriching teaching resources and monitoring student engagement. *Internet of Things* (2024), 101045.
- [11] Marcos Inonan, Brian Chap, Pablo Orduña, Rania Hussein, and Payman Arabshahi. 2023. RHLab scalable software defined radio (SDR) remote laboratory. In *International Conference on Remote Engineering and Virtual Instrumentation*. Springer, 237–248.
- [12] Fernando Ferreira Passe, Vanessa Cristiny Rodrigues Vasconcelos, Michael Canesche, and Ricardo Ferreira. 2017. Perspectivas para o uso do Node-Red no Ensino de IoT. *International Journal of Computer Architecture Education* 6 (2017), 46–51.
- [13] Ildelfonso Ruano, Elisabet Estevez, Javier Gamez, and Juan Gomez. 2023. Standards for the Integration of Online Laboratories with Learning Management Systems. *IEEE Access* (2023).
- [14] Muneer Bani Yassein, Mohammed Q Shatnawi, Shadi Aljwarneh, and Razan Al-Hatmi. 2017. Internet of Things: Survey and open issues of MQTT protocol. In *2017 international conference on engineering & MIS (ICEMIS)*. Ieee, 1–6.