Teaching Programming and Microcontrollers with an Arduino Remote Laboratory Application

Antonio Bukovac, Eva Pleše, Una Maravić, Petra Petrović, Tomislav Jagušt

University of Zagreb
Faculty of Electrical engineering and Computing
Unska 3, 10000 Zagreb
{antonio.bukovac, eva.plese, una.maravic, petra.petrovic, tomislav.jagust}@fer.hr

***Abstract*. This paper presents our work on an educational web system that teaches programming and microcontroller basics using the Arduino board in remote laboratory settings. In response to the shift towards remote and distance learning brought by the COVID-19 pandemic, and in an effort to alleviate the issue of continuous equipment shortage in many Croatian schools, we developed and tested a client-server architecture-based interactive remote laboratory system called RemoLab.**

The system consists of a web-based frontend application, a backend server with an accompanying database, and additional tools for connectivity with an Arduino microcontroller and a webcam. The system allows students to remotely write code for real hardware and instantly observe the results of their actions through a connected web camera.

The RemoLab system has the potential to enable teachers in enriching their online courses or homework assignments, but also back up students in independent learning and experimentation with Arduino or microcontrollers. We believe that this application has the potential to improve the way students learn microcontroller programming and can be used as an effective tool for distance learning.

Keywords. remote laboratory, arduino, programming, technology-enhanced education, sensors, iterative learning

I. INTRODUCTION

The standard method for conducting lab exercises is to set up a laboratory on campus with all necessary equipment, allowing students to apply their theoretical knowledge from lectures to practical work.
Because online education is growing in popularity due to its flexibility, practical exercises are also being adapted for online learning through remote laboratories.

Remote laboratories, which allow students to conduct experiments online without physical access to equipment, are making practical components of education possible and more accessible through remote learning [1]. In traditional labs, students interact directly with equipment through physical actions (e.g. hand gestures, pressing buttons, turning knobs) and receive feedback through their senses. In remote labs, the same interactions occur remotely through a user interface and remote infrastructure.

In this article, we will provide an overview of several remote laboratories that currently exist. Additionally, we will develop our own remote laboratory, which will be designed for teaching the programming of an Arduino board. Using a web application, users will have the ability to write code, which will then be executed on the Arduino board. As feedback, they will receive a live video feed, allowing them to see the results of their code in real-time.

II. RELATED WORK

Given that Arduino boards are increasingly being used in education and that online learning has become necessary and popular, there is an increasing number of papers on the subject of remote laboratories for programming and using Arduino boards as well as for other areas of knowledge.

Fernández-Pacheco et al. [2] highlighted the advantages and necessary components of remote laboratories. They designed their own remote laboratory using an Arduino MKR1000 Wi-Fi board, Raspberry Pi as the application server and Web camera for streaming the impact of code, providing students with an opportunity to experiment and explore the Internet of Things (IoT) using any platform, hardware, and software without the need to download additional extensions.

In their paper [3], Martin et al. described some of the many experiments that can be carried out using the previously described remote laboratory, which are used by students in the Electronics Engineering Degree in a Distance Learning University as part of the mandatory practices of a class. The work with an LED cube, various Arduino sensors, and an LCD screen is presented.

Kalúz et al. [4] presented a low-cost remote laboratory for control education. Their remote laboratory is based on Raspberry Pi and Arduino controllers for education in the field of process control and allows efficient energy usage and problem management without a supervisor. The goal of their project is to familiarize students with specific experiments, where they cannot modify or test the programming code for the Arduino, but only change certain values that affect the outcome of these precise experiments.

In their paper [5], Carro et al. described the use of a remote laboratory as an additional tool for teaching. The paper does not relate to learning programming or the use of Arduino, but rather presents a remote smart device and a remote laboratory with an RGB LED as a tool for learning about the physics of color, various perceptions of color, etc., and for improving interaction between students and teachers.

Lima et al. [6] described and designed a remote Arduino laboratory used for learning that can be accessed through two types of applications: one uses a visual and the other a textual programming language for learning Arduino programming. The user can see the code they input being executed on a remote Arduino board.

III. REMOLAB ARCHITECTURE

This chapter explains the architectural elements of the RemoLab project. The central component is a microcontroller board Arduino Uno that acts as the main control unit. Arduino is an open source electronic kit specifically designed as a controller that regulates the workings of other electronic circuits [7]. Multiple components are connected to the board: a temperature sensor, a servo motor, RGB LED and 16x2 LCD screen which are used to provide feedback to the user and can be manipulated by the program code uploaded to the Arduino board. The connection between the Arduino board and a computer is done through the USB port (Fig. 1.).

The WebCam is used to capture images and send them to the user's screen every few seconds. Currently, all components are connected to a computer. However, for better system flexibility, components could be connected to a Raspberry Pi, which would act as the main hub for data processing and communication.

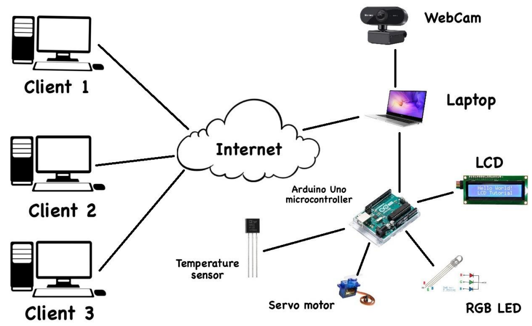

Fig. 1. RemoLab system architecture

The RemoLab system setup contains two servers, web server for the end-user application and the camera image server:

- The web server for the backend of the application runs on .NET and SQL Server database and is responsible for user authentication, showing tutorials in the application and uploading code to the Arduino microcontroller.
- The camera server runs on Node.js in combination with SocketIO and node-cam package. It is responsible for capturing and sending images to the clients.

IV. EXPERIMENTS

Several lessons were created that cover different fields of programming and electronics. Temperature sensor was chosen because it is simple and it gives students a basic understanding of sensor readings [8] and displaying results in a suitable way using LCD [9]. Servo motor is used as a component because it is an important part in dynamic projects like robotic arms [10] or remote controlled cars [11].

Students can easily access this laboratory by visiting the RemoLab website. On the website, students are presented with different categories, each containing simple tutorials and questions.

These educational resources are designed to help students in learning the fundamental concepts and commands of the Arduino device, and provide them with hands-on experience by allowing students to apply what they have learned through the tutorials on the actual Arduino device.

The code input page shows the camera image next to the code input field, allowing students to see the results of their actions in real-time (Fig. 2.). After entering the code, students can submit it to the server for compilation and execution on the Arduino. The camera display provides feedback on the execution of the code and helps students determine whether the code is functioning as intended. This setup allows students to visualize the results of their code and make adjustments as necessary.

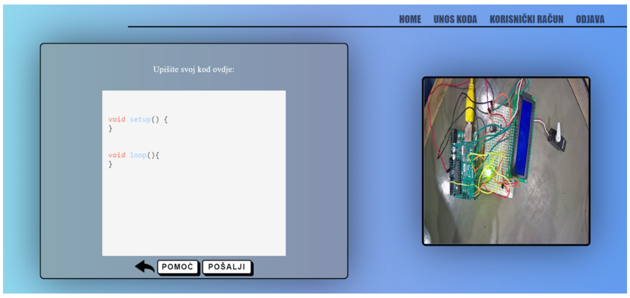

Fig 2. GUI of the experiment

There are four devices connected on the Arduino that students can program and control (Fig. 3.). The benefit of having all four devices on the same board is that students can easily experiment with multiple devices at once. This allows for a greater understanding of how different devices can interact with one another and opens up opportunities for more complex projects. For instance, the color of a RGB LED can be changed based on the measured temperature, demonstrating the relationship between two components.

Three devices, with accompanying exercises, are:

- *RGB LED*

  Goal of the RGB LED experiment is to teach basic functions/commands such as setting pins to OUTPUT, INPUT, and using the *delay* function. This will introduce students to the most important functions they will use in further experiments, in a fun and engaging way. Additionally, students will be introduced to the concept of RGB color model and have hands-on experience with it. Ability to change colors provides additional possibilities compared to a regular LED and makes the learning experience more enjoyable for the students.

- *Temperature sensor*

  The lesson with temperature sensor aims is to familiarize students with the basics of reading values from sensors. Here we use a temperature sensor as one of the basic sensors.

  In this exercise, students will also learn about converting analog inputs into meaningful values. In this example additional operations are required to obtain the temperature in e.g. degrees Celsius and this knowledge can be applied to other sensors such as humidity, light, etc.

- *Servo motor*

  In this exercise, students will learn how to control a servo motor using Arduino. Furthermore, this lesson teaches how to add "libraries" to the Arduino project and how to use them. In this particular case, students have to add the “Servo” library. This will help them to further seek and apply existing solutions that they can use to simplify certain parts of their projects.

  In this experiment students will perform a simple movement of a servo motor. Using the *delay* function and loops, they can create some interesting solutions. The aim is to familiarize students with the basic control of a servo motor and how libraries can be added and used in their Arduino code.

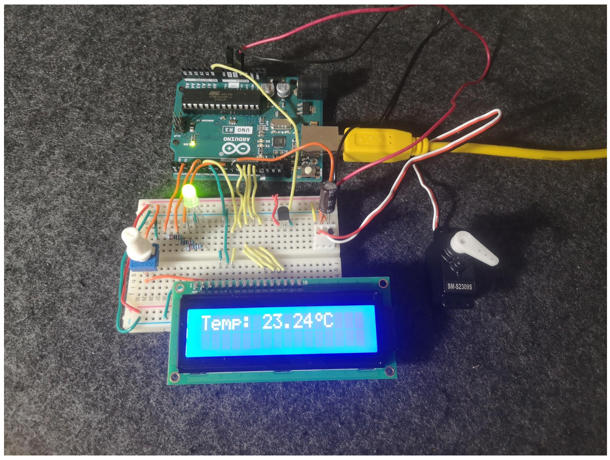

Fig. 3. Arduino board with LED, servo motor, temperature sensor and LCD screen

V. CONCLUSION

This paper describes the architecture and implementation of a remote laboratory for teaching microcontroller programming. The RemoLab system includes several components, such as a temperature sensor, servo motor, RGB LED, and a camera for live feedback. The system allows users to write code through a web application, which is then sent on server, compiled and uploaded to the Arduino board. Program execution results can be observed through the embedded web camera.This allows us to have an iterative learning experience, where we test something, observe the results, and make adjustments to try again.

The RemoLab remote laboratory provides students with a simple way to learn about Arduino programming and experiment with different components, all from the comfort of their own devices. This type of remote laboratory can be a valuable addition to any education curriculum and can make it more accessible for students to learn about technology and programming.

REFERENCES

[1] J. E. Corter, J. V. Nickerson, S. K. Esche and C. Chassapis, "Remote versus hands-on labs: a comparative study," 34th Annual Frontiers in Education, 2004. FIE 2004., Savannah, GA, USA, 2004, pp. F1G-17, doi: 10.1109/FIE.2004.1408586.

[2] A. Fernández-Pacheco, S. Martin, and M. Castro, “Implementation of an Arduino remote laboratory with raspberry Pi,” in Proc. IEEE Global Eng. Educ. Conf. (EDUCON), Apr. 2019, pp. 1415–1418.

[3] S. Martin, A. Fernandez-Pacheco, G. Carro, J.A. Ruipérez-Valiente and M. Castro, “Remote Experimentation Through Arduino-Based Remote Laboratories” in IEEE Revista Iberoamericana de Tecnologias del Aprendizaje, vol. 16, no. 2, May 2021, pp. 180.-186.

[4] M. Kalúz, L. Cirka, R. Valo, and M. Fikar, “ArPi lab: A low-cost remote laboratory for control education,” IFAC Proc. Volumes, vol. 47, no. 3, pp. 9057–9062, 2014.

[5] G. Carro et al., “The color of the light: A remote laboratory that uses a smart device that connects teachers and students,” in Proc. IEEE Global Eng. Educ. Conf. (EDUCON), Apr. 2014, pp. 854–860.

[6] J. P. C. de Lima, L. M. Carlos, J. P. S. Simão, J. Pereira, P. M. Mafra, and J. B. da Silva, “Design and implementation of a remote lab for teaching programming and robotics,” IFAC-PapersOnLine, vol. 49, no. 30, pp. 86–91, 2016.

[7] Darmawan, Aan & Ratnadewi, Ratnadewi & Sartika, Erwani & Pasaribu, Novie & Arlando, Riko. (2017). Basic Arduino Programming Training For High School Students. THE SPIRIT OF SOCIETY JOURNAL. 1. 10.29138/scj.v1i1.456.

[8] M. Kusriyanto and A. A. Putra, "Weather Station Design Using IoT Platform Based On Arduino Mega," 2018 International Symposium on Electronics and Smart Devices (ISESD), Bandung, Indonesia, 2018, pp. 1-4, doi: 10.1109/ISESD.2018.8605456.

[9] G. M. Debele and X. Qian, "Automatic Room Temperature Control System Using Arduino UNO R3 and DHT11 Sensor," 2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, China, 2020, pp. 428-432, doi: 10.1109/ICCWAMTIP51612.2020.9317307.

[10] V. Varshney, A. Kumar, T. Saxena, P. Jha, M. N. Tiwari and Savita, "Design and Development of Intellectual Robotic Arm," 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2019, pp. 984-987.

[11] Yuxin Jing, Letian Zhang, I. Arce and A. Farajidavar, "AndroRC: An Android remote control car unit for search missions," IEEE Long Island Systems, Applications and Technology (LISAT) Conference 2014, Farmingdale, NY, USA, 2014, pp. 1-5, doi: 10.1109/LISAT.2014.6845227.