The Future of Engineering Education
2024 Annual Conference & Exposition
Oregon Convention Center
Portland, OR . June 23 - 26, 2024
ASEE
Paper ID #42161

Immersive Virtual Labs for Enhancing In-Person and Online Education

Ms. Yiyang Li, Old Dominion University

Yiyang Li received her B.S. degree Henan University of Economics and Law, Zhengzhou, China, in 2018, the M.S. degree from Old Dominion University, Norfolk, VA, in 2021. She is currently in her thrid year of her Ph.D. in Old Dominion University. From 2013 to 2015, she was a Senior Modeler, Rigging Technical Director, and group manager at Shiwei Animation, Luoyang, China. Her current research interests are rapid development of virtual lab for STEM education and virtual environment.

Prof. Yuzhong Shen, Old Dominion University

Yuzhong Shen received his B.S. degree in Electrical Engineering from Fudan University, Shanghai, China, M.S. degree in Computer Engineering from Mississippi State University, Starkville, Mississippi, and Ph.D. degree in Electrical Engineering from the University of Delaware. Dr. Shen is currently a Professor of Electrical and Computer Engineering at Old Dominion University. His research interests lie in visualization and computer graphics, virtual reality, augmented reality, transportation modeling and simulation, signal and image processing, and general modeling and simulation.

Charles I Sukenik, Old Dominion University

Charles I. Sukenik received a B.A. in Physics from Cornell University in 1987 and a Ph.D. in Physics from Yale University in 1993 under the direction of Ed Hinds and working with Serge Haroche. He spent two years as a postdoctoral associate and lecturer in physics at the University of Michigan-Ann Arbor, where he worked with Phil Bucksbaum. He then spent two years as a postdoctoral associate at the University of Wisconsin-Madison working with Thad Walker. He joined the faculty of Old Dominion University in 1997, where he is now Professor of Physics. He served as Chair of the Department for 9 years. Charles' research is primarily in experimental atomic, molecular and optical physics, but he also has ongoing research efforts in remote sensing in oceanography and laser-based supersonic flow diagnostics.

Immersive Virtual Labs for Enhancing In-Person and Online Education

Abstract

Labs play a critical role in science and engineering education, offering practical insights and hands-on experience to students that cannot be achieved through theoretical learning alone. With the continuous advancement in technology, education is being reshaped and many universities are now offering online programs. This shift in educational paradigm offers students access to a wider range of academic resources, without being limited by geographical boundaries, time constraints, among others. However, the rise of online education also brings unique challenges, such as lack of face-to-face interaction and limited hands-on learning experiences. Virtual labs, which allow students to conduct experiments in a computer-simulated environment, can provide a viable and effective solution for online courses that require hands-on labs.

Virtual Labs offer numerous advantages that can complement, and in some cases even replace, traditional in-person labs. Many educational institutions, including secondary schools and higher education institutions, face challenges in terms of establishing and maintaining traditional labs, such as high costs, space constraints, limited accessibility, among others. Virtual Labs, on the other hand, can simulate many of the experiences of a physical lab, allowing students to conduct experiments, gather data, and analyze results in a virtual environment. Virtual labs can be accessed from anywhere at any time. They can be used as pre-labs that can greatly shorten the time needed for the real physical lab or as post-labs if the student cannot complete the physical lab in the allotted time. Consequence of mistakes can be safely demonstrated in the virtual lab without exposing the student to real danger. Virtual labs can be especially useful for underrepresented groups that often do not have sufficient educational resources.

This paper explores a wide range of issues in the rapid development of immersive virtual labs for engineering education, including different types of representations of the virtual environment, instrument and circuit simulation, user interface design, and integration of third-party libraries. Simulations of virtual environments often necessitate various types of representations, such as graphics models, physical models, and functional models, to optimize performance, fidelity, computational cost, and reusability. Utilization of multiple representations allows rapid development of virtual labs. For instance, the functional model of one oscilloscope can be easily adapted for another oscilloscope designed by a different manufacturer with minor modifications as oscilloscopes have similar functionalities while their visual appearance can vary. Advanced software development techniques such as object-oriented design and delegates are utilized to tackle the challenges in complex instrument and circuit simulation, such as the representation of continuous signals and discrete (digital) signals using the sampling theorem. This project makes use of the state-of-the-art design principles and techniques to create a user interface and virtual environment that are user friendly, efficient, and effective for learning. Integration of existing third-party software libraries is another crucial component in the rapid development of virtual labs. This project successfully integrated SPICE, a popular circuit simulator, as the backend of the virtual lab, greatly expediting the overall development. This paper will discuss the techniques for integration of third-party software to achieve interoperability between different software.

While our current development focuses on the virtual labs for the course PHYS 303 offered at Old Dominion University (ODU), the proposed development techniques can be readily extended

to other courses that utilize these common instruments, including courses offered by universities and high schools. A preliminary user study conducted with the first lab module in the course PHYS 303 demonstrated the effectiveness of the virtual lab.

1. Introduction

In the evolving landscape of educational technology, virtual labs have emerged as an important tool, offering an alternative to traditional laboratory experiences. With technology's continual advancement and integration in educational settings, virtual labs are increasingly gaining prominence. This trend is particularly evident in the fields of science, technology, engineering, and mathematics (STEM) education. Initiatives like the Go-Lab federation [1] have significantly contributed to making virtual labs broadly accessible for educational purposes, incorporating simulations, remote labs, and data from actual lab experiments. The substantial increase in research focused on Virtual and Remote Labs (VRLs) over the last two decades demonstrates their expanding role in STEM fields [2]. Furthermore, A study conducted before and after the COVID-19 pandemic showed that virtual labs have become a critical substitute for the instruction and demonstration phases of conventional laboratory classes, thereby reducing dependency on traditional instructional methods. While educators may consider the pre- and post-COVID-19 eras as distinct phases in education, research conducted after the pandemic indicates that the shift towards using virtual labs was a trend already emerging before the outbreak, suggesting continuity in the evolution of learning behaviors rather than a sudden change due to the pandemic [3].

Virtual environments provide a platform where students can conduct experiments, explore simulations, and engage with interactive learning modules, all within a digital realm. The advent of immersive technologies has taken the concept of virtual labs a step further. Immersive virtual labs do not just replicate the physical lab experience; they create a multi-sensory, interactive learning environment that transcends physical and geographical limitations. These labs utilize the full spectrum of immersive technology, from 3D models and simulations to fully interactive virtual environments, providing students with a hands-on experience that is both engaging and educational.

As we delve into the details of immersive virtual labs and their impact on education, it is essential to explore how these technological advancements align with pedagogical theories, contribute to improved learning outcomes, and could shape the future of education. This paper aims to discuss the potential challenges and future prospects of these innovative learning tools. This paper begins by examining the Intermediate Experimental Physics (PHYS 303) at Old Dominion University, forming a basis for an initial version of virtual laboratory. It explores the challenges encountered in creating a 3D virtual laboratory and proposes thoughtful, and practical solutions. The discussion aims to contribute to the broader understanding of developing virtual labs, offering a preliminary guide for addressing these complex challenges.

The remainder of this paper is organized as follows: Section 2 provides the background to the physics laboratory that is the focus of our modeling and simulation efforts in this study. Section 3 details the methodology employed in the latest iteration of the virtual lab, followed by Section 4 that outlines the current findings and results obtained from the virtual lab. Section 5 presents the user evaluation of the virtual lab. Finally, Section 6 discusses future work in this area and concludes the paper.

2. Background

In the latter part of 1995, Old Dominion University's Department of Physics started adopting a more uniform method for teaching undergraduate laboratory courses. Following this change, in 1996, the department released the first edition of a comprehensive laboratory manual for undergraduates. This marked a significant shift in the instructional methodology for undergraduate physics at ODU. The development of virtual laboratories is set to enhance this teaching approach further. With the integration of pre-arranged educational materials, including virtual labs, educators will be able to adhere to uniform teaching standards. This uniformity will ensure that students receive a consistent and coherent learning experience [4].

Figure 1. Physics Lab of the Department of Physics at Old Dominion University

In the development of the initial version of the virtual lab, the selection of PHYS 303 as the prototype was a strategic decision. This laboratory course, designed for sophomore and junior students, is representative of a typical electronics lab in physics education, encompassing a comprehensive range of instruments and components. These include oscilloscopes, function generators, multimeters, breadboards, and essential circuit elements like resistors and capacitors. A picture of the lab used by PHYS 303, which illustrates these features, is shown in Figure 1. The choice of this particular course is based on its inclusiveness of equipment and experiments that are fundamental to electronics labs in most educational institutions. By starting with this course, the virtual lab prototype aims to address a broad spectrum of common challenges in virtualizing lab experiences, facilitating the development of effective solutions. Moreover, the universality of the lab equipment and experiments in this course means that the initial virtual lab model can be readily adapted or extended to other electronics labs within the same university and potentially to similar courses at different institutions. This approach ensures that the virtual lab is not only a solution for a single course but also a scalable model that can be applied more widely, thereby maximizing its impact and utility in the realm of physics and electronics education. Figure 2 shows the comparison of the workstations from both the real physical lab and the virtual lab.

(a) (b)

Figure 2. (a) A workstation in the real physical lab. (b) A workstation in the virtual lab.

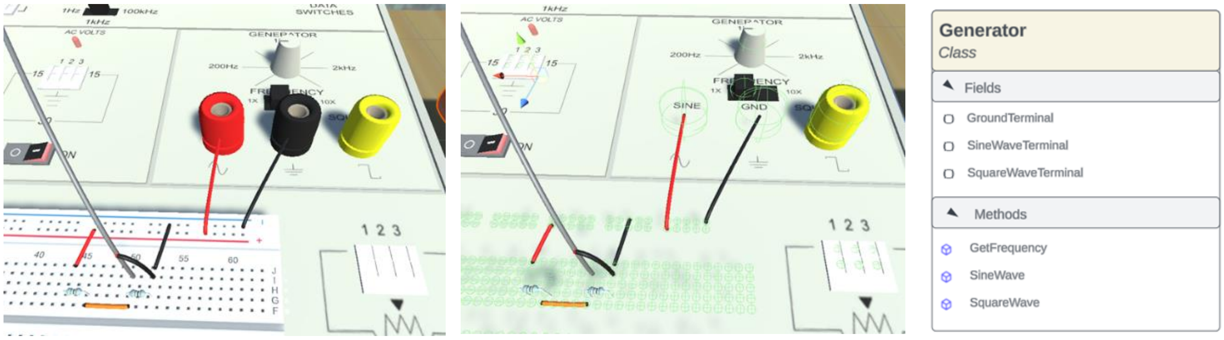
3. Methodologies

A. Multi-Model Representation

In the process of developing a 3D virtual laboratory, a key consideration is the variety of models needed to simulate instruments and circuit components authentically. This necessity stems from the multifaceted nature of virtual lab environments, where objects must not only appear realistic but also behave and function as they would in a physical lab. To address this, we have conceptualized three fundamental model types, each serving a unique purpose, aligned with the principles of Object-Oriented Programming (OOP): the graphical model, physical model, and functional model.

- The graphical model forms the visual representation of the object. It includes the 3D models of instruments and circuit components, providing the digital representation necessary for computer rendering. This model is critical as it shapes the user's visual interaction with the lab, rendering the intricate details and aesthetics of each object.
- Moving beyond the appearance of objects, the physical model is pivotal in replicating the dynamic and interactive aspects of the lab environment. It simulates the physical behavior of objects, such as their movement and collisions, thereby mirroring the interactions one would expect in a real lab. It plays a critical role in the program's ability to check circuit connectivity.
- Lastly, the functional model focuses on the operational characteristics of the objects. It defines how each component functions within the circuit, independent of its appearance, manufacturer, or physical behavior. For example, one of the functionalities of an oscilloscope in this model would be to receive and display signal waveforms.

The conceptualization of these three model types – graphical, physical, and functional enhances the scalability of the virtual laboratory. This modular design approach allows for adaptable and efficient revisions and expansions to the lab environment. For instance, in the case of another electronics course that utilizes different versions or manufacturers of instruments, developers can easily extend the virtual lab to accommodate these new requirements with minimal changes. This primarily involves updating the graphical model to reflect the different appearances of the new instruments. Additionally, the physical model may be modified if there is a need to adjust the simulation of the instruments' physical behaviors, such as their interactions or movements within the virtual space. The functional model often remains unchanged, as the core functions of similar instruments in electronics labs typically stay consistent. Figure 3 illustrates the three types of models discussed, using the Generator Panel of the digital trainer used in the PHYS 303 as an example. From left to right, these models are the graphical model, the physical model, and the class diagram of the functional model.



(a) (b) (c)

Figure 3. Multiple representations of the signal generator on the trainer. (a) Graphical model with high resolution. (b) The physical model is represented by various shape outline in green color for collision detection. (c) Functional model represents the signal generator's operations.

B. Developing Environment

Given the necessity of graphical models in 3D environments, acquiring these models quickly is crucial. To address this, we employ 3D scanning to capture laboratory instruments and use these scans as references for creating their 3D models. This approach ensures that the dimensions and proportions of the models are accurate without the need for additional measurements or adjustments, significantly speeding up the modeling process. Additionally, some instrument models are available online, allowing developers to source them for scenarios that do not require specific brands or instruments. It's important to note, however, that if developers have specific model requirements, the requisite skills for model creation are necessary. Figure 4 demonstrates the mesh model obtained through our use of 3D scanning technology, as well as the higher resolution model that was subsequently generated based on this mesh. We employed Autodesk Maya [5] as our 3D modeling software. Other viable options for 3D modeling include, but are not limited to, Blender [6], 3Ds Max [7], and ZBrush [8] among others.

In this project, the Unity Engine was chosen as the primary software development tool. Unity stands out as a prominent game engine, widely recognized for its versatility in cross-platform games and interactive simulation development. It provides a set of sophisticated features, including rendering, a physics engine, animation, as well as support for virtual and augmented reality applications [9].

(a) (b)

Figure 4. 3D models of an oscilloscope. (a) The original 3D model is directly generated by the 3D scanner. (b) The final 3D model that was generated using Maya based on the original model.

C. The Program Architecture

The program is structured into several interconnected modules.

- *Component management* and *Instrumentation Module* are the first concerns when developing the system, where developers can prototype and manage various electrical components and instruments to be used in the virtual lab.
- *Circuit Build Module / Circuit Topology Extraction Module:* This central module is where users construct the circuit by placing and interconnecting the components.
- *Circuit Simulation Module (SPICE):* The SPICE-based circuit simulation module takes the circuit netlist—a detailed description of the circuit components and their connections—and performs the actual simulation, calculating the electrical behaviors and outputs.
- *User Interface (UI):* It contains the graphical representation of the lab environment. This module provides the interactive elements for users to build and simulate circuits, ensuring a user-friendly experience.
- *User Data Management:* It handles the storage, retrieval, and management of user data, including saved circuits, simulation histories, and user preferences.
- *Lab Instructions Module:* This section offers structured guidance and educational content to help users understand the laboratory tasks and objectives.
- *Result Module:* It handles the simulation results by providing visual feedback and data analysis options for the user. Additionally, it transmits these results to the instruments when display on the instrument is required.

The program architecture design of the virtual lab is shown in Figure 5. The flow of data and control across these modules are organized to categorize the program's various functionalities. This architecture provides basic recommendations for the rapid development of virtual laboratories.

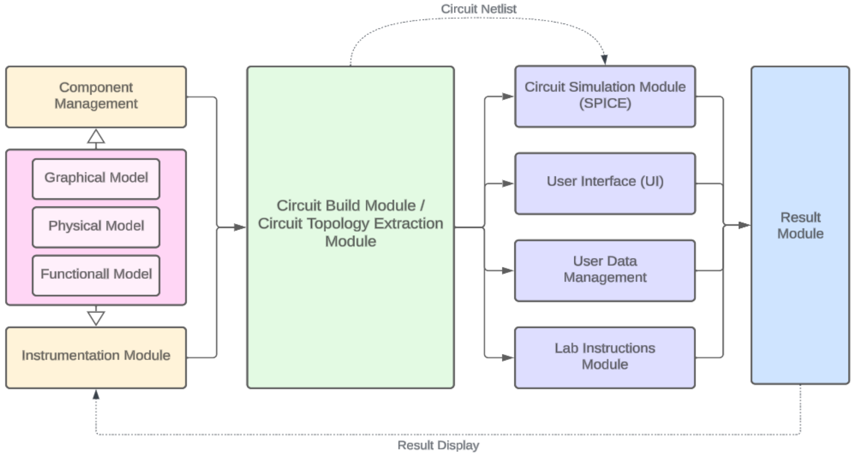


Figure 5. The program architecture of the virtual lab.

We employed advanced software development techniques such as delegates and interfaces. The use of interface allows the program to invoke different methods implemented trough the interfaces in response to user interactions with various objects, effectively managing user input requirements across different scenarios. The application of delegates is evident in scenarios where communication between different instruments is necessary. Through delegates, instruments do not need to be aware of each other's existence but to respond when certain events occur.

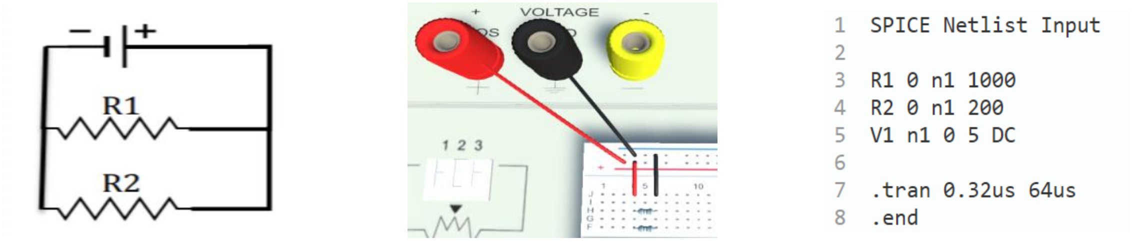
D. Circuit Topology Extraction and SPICE Circuit Simulator

Although Unity allows for the execution of a diverse range of calculations through provided code, the task of programming it for circuit simulation remains a great challenge. This is primarily due to the sophisticated nature of electrical circuit simulations, which require not just the representation of components and connections but also the implementation of algorithms that can solve the circuit equations in real time. Consequently, the issue becomes how to effectively analyze circuits built within Unity to simulate and present the appropriate information that users need to receive.

SPICE (Simulation Program with Integrated Circuit Emphasis) is an open-source, general-purpose analog circuit simulator [10]. It is the most widely used program for circuit design and circuit simulation in academia and industry. By incorporating SPICE as the backend circuit simulator, the virtual laboratory not only demonstrates a design prioritizing flexibility and scalability through its modular architecture but also ensures accurate simulation capabilities. The integration of SPICE enhances the functionality of the laboratory, allowing for precise and comprehensive circuit analysis and simulation. SPICE utilizes text input to describe circuit components, known as netlists. These netlists describe the arrangement and connections of the components within a circuit. For instance, an DC voltage source is represented in the format

$$\texttt{<Vxxx> <N}_1\texttt{> <N}_2\texttt{> <Type> <Value>}$$

Here, "Vxxx" is a label assigned to the voltage source, with "xxx" denoting its specific name. "N$_1$" and "N$_2$" refer to the two terminal nodes connected by the voltage source. The terms "Type" and "Value" are used to define the type of the source and its corresponding value, respectively, in the circuit configuration.



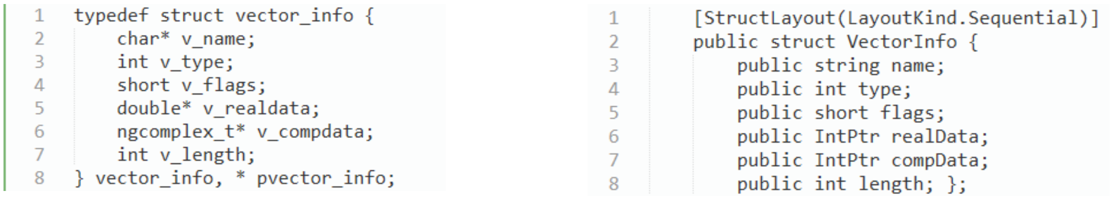
```
SPICE Netlist Input

R1 0 n1 1000
R2 0 n1 200
V1 n1 0 5 DC

.tran 0.32us 64us
.end
```

(a) (b) (c)

Figure 6. (a) The schematic diagram of a circuit. (b) The circuit built in virtual lab. (c) The netlist generated for circuit simulation.

Knowing that SPICE requires a netlist as its input, the program must generate a text string representing the circuit constructed by the user. To achieve this, a circuit topology extraction algorithm was developed. This algorithm is designed to translate the circuit built by the user in the 3D environment into a corresponding netlist format, thereby bridging the gap between the virtual lab's interactive interface and SPICE's analytical capabilities as showed in Figure 6(b) and 6(c). Thanks to the design of the physical model, circuit connections can be detected through the collision of objects in Unity. Once a connection is established, the program registers the corresponding connectors of each component with one another. Each connector holds a reference to the connectors of its neighboring components, playing a pivotal role in determining the circuit topology within the algorithm. This effectively translates the physical interactions within the 3D environment into a meaningful electronic configuration. Moreover, when the program needs to ascertain the location of a specific connector within the circuit—such as when a user inserts a voltmeter into the connection, and the exact node to which the voltmeter's connector is attached needs to be identified—a search algorithm can be employed for quick retrieval.

In our project, the integration of SPICE presented a challenge due to the C programming language it uses, while Unity uses C#. To bridge this gap, we developed a SPICE wrapper to enable its integration as a dynamic link library (DLL) within the program. The wrapper uses marshaling to manage the transition of data between managed and unmanaged memory. Managed memory is controlled by the .NET runtime, while unmanaged memory is typically utilized in native code like C or C++ libraries. This concept is particularly important when working with interoperability between managed .NET code and unmanaged functions [11]. The exported functions from DLL were wrapped into C# class using *DllImport*, and the callback functions was marshaling in C# as managed delegates. A callback function is a function that is used to pass to another function, while a delegate in C# is a type-safe function pointer that holds a reference to a function. The structures from the library are also rewritten to transport data between managed and unmanaged code. *StructLayout* is used to control its layout, as shown in Figure 7.



```c
typedef struct vector_info {
    char* v_name;
    int v_type;
    short v_flags;
    double* v_realdata;
    ngcomplex_t* v_compdata;
    int v_length;
} vector_info, * pvector_info;
```

```csharp
[StructLayout(LayoutKind.Sequential)]
public struct VectorInfo {
    public string name;
    public int type;
    public short flags;
    public IntPtr realData;
    public IntPtr compData;
    public int length; };
```

(a) (b)

Figure 7. (a) The vector_info struct from the DLL. (b) The vector_info struct in C# wrapper.

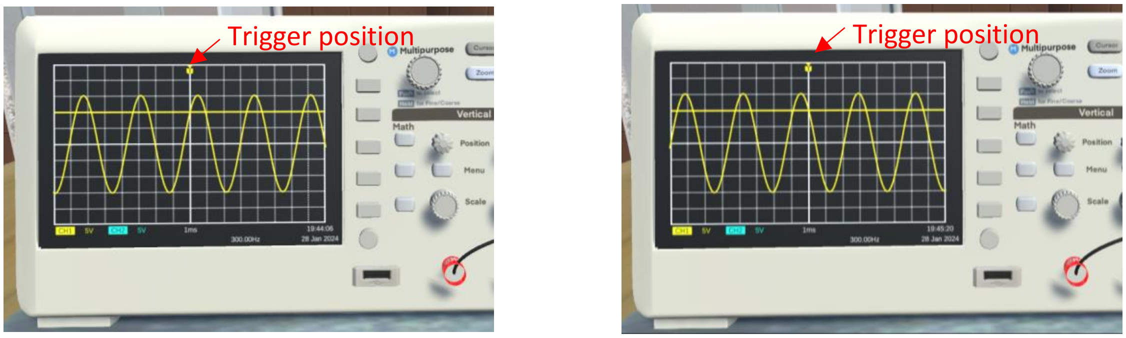
This wrapper not only facilitates communication between SPICE and Unity but also simplifies the process of accessing SPICE's output. With the wrapper in place, the program can directly read results from SPICE, eliminating the need for SPICE to generate additional files for later retrieval.

4. Current Results

The current version of the Virtual Lab has been developed for the first lab session of the PHYS 303. Figure 8 shows the virtual lab user interface. In this session, students are introduced to two fundamental concepts: (1) the utilization of an oscilloscope, as illustrated in Figure 9, and (2) the configuration of circuits for the validation of Ohm's Law, depicted in Figure 10.

Name Resistor0
Resistance 1000

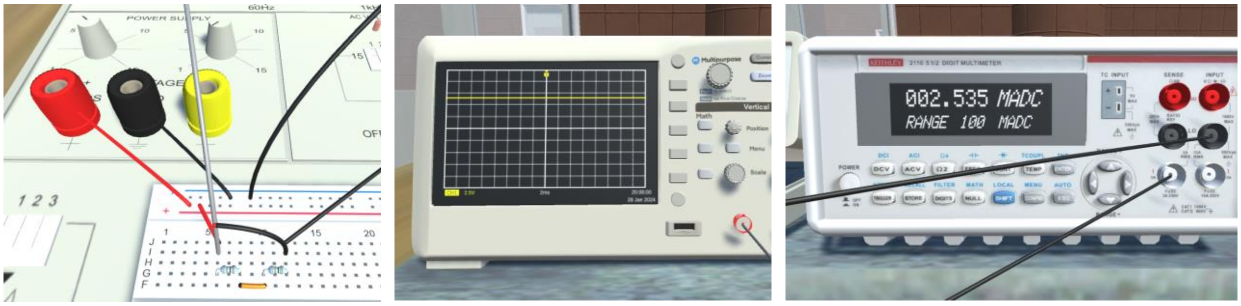
Figure 8. The user interface of virtual lab.



(a) (b)

Figure 9. The lab task involves learning to use the trigger level and trigger slope on an oscilloscope. Both images depict the same trigger level with (a) the trigger slope set to ‘rising’ and (b) the trigger slope set to ‘falling’. The trigger position is indicated at the top of the oscilloscope’s display.

In the virtual lab, users have the ability to connect a function generator to an oscilloscope, enabling them to display signal waveforms. This setup serves as an introductory lesson in using oscilloscopes. Following this, users can place resistors on a breadboard and adjust their resistance values by clicking on them. Circuit connections are made by clicking on different terminals, and by pressing the 'Simulate' button, users can initiate the circuit simulation and view the results of their configuration.



(a) (b) (c)

Figure 10. Virtual lab set up for a simple series circuit. (a) A 5V DC is applied to two resistors in series connection. (b) The voltage is displayed on the oscilloscope. (c) The current is displayed on the digital multimeter.

5. User Evaluation

In evaluating the effectiveness of our virtual lab, we conducted a preliminary user evaluation. This evaluation's primary objective was to gather insights into the user experience and assess the lab’s effectiveness. Given the original course's semester schedule, the current evaluation results are based solely on internal testing.

A. Methodology

The methodology employed for this user evaluation was primarily based on a questionnaire survey. To ensure a diverse range of perspectives, the participants in this internal testing phase consisted of five students specializing in various fields, including finance, electrical and computer engineering, chemical engineering, and computer science. The rationale behind selecting students from different majors was to broaden the diversity of the test group, thereby gaining insights from users with varied academic backgrounds and expertise. Under the guidance of the developers, users first familiarized themselves with the operations of the Virtual Lab. Subsequently, they conducted an experiment following the instructions of the first lab session of the PHYS 303 and completed the survey questionnaire. This questionnaire employed a Likert-type scale [12], renowned for its widespread use in gauging user feedback. The scale ranged from 1 (strongly disagree) to 5 (strongly agree), providing a quantitative measure of user responses to the following questions:

1. I find the virtual lab interface is user-friendly for conducting experiments.
2. I am able to comfortably navigate and interact with equipment in the virtual lab, such as oscilloscopes and function generators.
3. The virtual lab is effective in simulating the real-world laboratory experience.
4. The virtual lab enhanced my understanding of circuit concepts compared to traditional methods.
5. I am able to complete the first lab based on the provided lab instructions.
6. I am satisfied with the quality and accuracy of the simulation results in the virtual lab.
7. Overall, I have had a positive experience with the virtual laboratory as a tool for learning and practicing electrical circuits.

B. Data Collection and Analysis

Data collection was confined to internal testing. Despite the limited scope, the feedback gathered provides an initial understanding of the virtual lab’s impact and areas for enhancement. The analysis of this data involved a statistical examination of the quantitative responses and a thematic analysis of the qualitative feedback, providing a comprehensive view of the user experience. The mean and standard deviation for each question is shown in Table 1.

The independent t-test comparing the mean scores between the two students from electrical and computer engineering (ECE) and the other three students from different majors resulted in a t-statistic of approximately 0.598 and a p-value of approximately 0.561. The p-value is much greater than the typical threshold for significance $p = 0.05$, which indicates that there are no statistically significant differences in the average scores given by the ECE students compared to students from other majors. This suggests that the students' majors did not significantly affect how they rated the virtual lab experience.

Table 1. Mean and Standard Deviation (SD) on virtual lab survey.

Survey Question	Mean, (SD)	Survey Question	Mean, (SD)
1	4.0, (0.707)	5	4.2, (0.447)
2	3.6, (0.5548)	6	4.2, (0.447)
3	4.2, (0.837)	7	3.8, (0.447)
4	4.0, (1.000)		

Figure 11 presents a visualization illustrating user evaluation for the Virtual Lab, separated into two distinct groups for comparative analysis. The first group includes students major in ECE, while the second encompasses students from a variety of other disciplines. This separation is designed to facilitate an intuitive understanding of whether academic background influences user ratings. Based on the visualization, while there are slight variations in scores between students from different academic backgrounds, the overall trend in the ratings appears consistent. This suggests that users share a common perception of the Virtual Lab's effectiveness and usability.

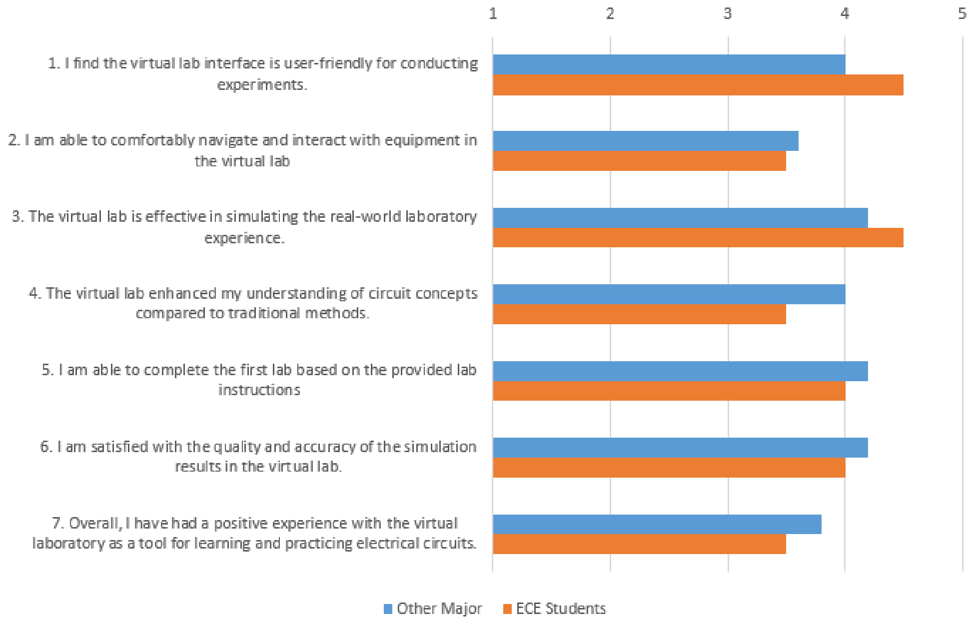


Figure 11. The visualization of user evaluation.

The survey concluded with an open-ended question inviting participants to comment on the Virtual Lab. Feedback from the respondents indicates a consensus on the potential benefits of having more descriptive text prompts in the Virtual Lab. For instance, they suggested that tooltips or text descriptions appearing when hovering the mouse over buttons on the instruments could significantly aid in understanding the function of each button. Additionally, some respondents mentioned that audio feedback during interactions with buttons and knobs might enhance the interactivity of the Virtual Lab. These insights offer a possible direction for future enhancements in the design of interactive virtual environments.

6. Conclusion and Future work

In conclusion, this paper presented the virtual lab as an effective and user-friendly platform for simulating real-world laboratory experience. The modular design of multiple-model representation allows for adaptable and efficient revisions and expansions to the virtual lab development. The implementation of virtual lab leveraged advanced programming techniques like delegates and interfaces, streamlining the development and enhancing the modularity of the system. Additionally, the development of a C# wrapper was important for integrating SPICE, facilitating seamless interaction between SPICE's accurate simulation capabilities and the virtual lab. The user evaluation suggests its broad applicability in educational settings. Future enhancements, guided by user feedback, are expected to further improve its interactivity and educational effectiveness. The design and approach of the Virtual Lab provide a viable solution for the ongoing and future development of virtual laboratories in STEM education.

References

[1] T. de Jong, S. Sotiriou, and D. Gillet, “Innovations in STEM education: the Go-Lab federation of online labs,” *Smart Learning Environments*, vol. 1, no. 1, p. 3, Oct. 2014, doi: 10.1186/s40561-014-0003-6.

[2] I.-E. Lasica, K. Katzis, M. Meletiou-Mavrotheris, and C. Dimopoulos, “STEM Education: Current and future trends in laboratory-based education,” Jun. 2016.

[3] R. Radhamani, D. Kumar, N. Nizar, K. Achuthan, B. Nair, and S. Diwakar, “What virtual laboratory usage tells us about laboratory skill education pre- and post-COVID-19: Focus on usage, behavior, intention and adoption,” *Educ Inf Technol (Dordr)*, vol. 26, no. 6, pp. 7477–7495, 2021, doi: 10.1007/s10639-021-10583-3.

[4] Y. Li, Y. Shen, C. Sukenik, B. Sanders, P. Delacruz, and J. Mason, “Work-in-Progress: Rapid Development of Advanced Virtual Labs for In-Person and Online Education”.

[5] “Maya Software | Get Prices & Buy Official Maya 2024 | Autodesk.” Accessed: Jan. 28, 2024. [Online]. Available: https://www.autodesk.com

[6] B. Foundation, “blender.org - Home of the Blender project - Free and Open 3D Creation Software,” blender.org. Accessed: Jan. 28, 2024. [Online]. Available: https://www.blender.org/

[7] “Autodesk 3ds Max Software | Get Prices & Buy Official 2024.” Accessed: Jan. 28, 2024. [Online]. Available: https://www.autodesk.com/products/3ds-max/overview?term=1-YEAR&tab=subscription

[8] “ZBrush,” Maxon. Accessed: Jan. 28, 2024. [Online]. Available: https://www.maxon.net/en/zbrush

[9] “Unity Real-Time Development Platform | 3D, 2D, VR & AR Engine,” Unity. Accessed: Jan. 28, 2024. [Online]. Available: https://unity.com

[10] “SPICE (Simulation Program with Integrated Circuit Emphasis).” Accessed: Jan. 28, 2024. [Online]. Available: https://publichealth.buffalo.edu/cat/kt4tt/best-practices/need-to-knowledge-ntk-model/ntk-commercial-devices/master-list-of-tools/electrical-electronic-engineering/spice--simulation-program-with-integrated-circuit-emphasis--.html

[11] dotnet-bot, “Marshal Class (System.Runtime.InteropServices).” Accessed: Feb. 04, 2024. [Online]. Available: https://learn.microsoft.com/en-us/dotnet/api/system.runtime.interopservices.marshal?view=net-8.0

[12] P. Bhandari, “What Is a Likert Scale? | Guide & Examples,” Scribbr. Accessed: Nov. 24, 2023. [Online]. Available: https://www.scribbr.com/methodology/likert-scale/