Year: 2026, Volume: 1, Number: 2, Pages: 127-140

Journal of Advanced Vocational Information and Communication Technology

Homepage: https://journal.isas.or.id/index.php/javict, email: javict@isas.or.id

Virtual Lab Module Feature Development on ETHOL System for PHP Learning

Sherly Maya Salsabilla[1,], Mirta Chadhirotin Nachlah[1]*

[1,*] Department of Informatics and Computer Engineering, Politeknik Elektronika Negeri Surabaya, Surabaya, Indonesia

*Corresponding Author: mailto:mirtachadhirotin@gmail.com

Article Info

Article History:
Submitted 24 Apr 2026
First Revised 7 May 2026
Accepted 21 May 2026
First Available Online 22 May 2026
Publication Date 25 May 2026

Keyword:
Virtual laboratory;
e-learning;
learning management system;
web service;
distance learning;
ETHOL.

Abstract

Information technology in education has rapidly evolved, particularly in supporting distance learning systems. However, conventional e-learning platforms often lack practical components, limiting students' ability to develop hands-on skills. This study addresses the need for an interactive learning environment by developing a virtual laboratory focused on PHP programming. The proposed system enables students to perform practical exercises in a web-based environment, simulating real laboratory activities. The virtual lab is integrated with the existing Learning Management System through web services, specifically ETHOL (Enterprise Technology Hybrid Online Learning) at EEPIS. The implementation of this system provides flexibility for students to access learning materials and conduct experiments anytime and anywhere. The results indicate that the virtual lab enhances students' understanding of programming concepts and supports independent learning. Load testing demonstrated that the system handles 4,638 requests within 3 minutes with an average response time of 26 ms and a throughput of 27 requests per second on minimal cloud hardware (1 GB RAM, 1 vCPU). Responsiveness testing confirmed that the website functions properly on 100% of tested devices (2 desktop and 1 mobile resolutions). Therefore, the developed virtual lab can serve as an effective complementary tool in e-learning environments.

1. Introduction

The rapid advancement of information technology (IT) has significantly transformed various aspects of human life, particularly in the field of education. The integration of IT in education plays an essential role in accelerating knowledge dissemination and improving learning effectiveness [1]. One of the most widely adopted technologies in modern education is the Learning Management System (LMS), which is derived from e-learning systems that convert conventional face-to-face learning into digital platforms accessible via the internet [2]. LMS enables flexible learning environments, allowing students to access learning materials anytime and anywhere, thus overcoming limitations of time and space. In addition, the implementation of the Self-Directed Learning (SDL) model encourages students to independently identify learning needs, formulate learning objectives, and evaluate their own learning outcomes.

Politeknik Elektronika Negeri Surabaya (PENS) is one of the vocational higher education institutions in Indonesia that has successfully implemented a distance learning system through an LMS-based platform known as ETHOL (Enterprise Technology Hybrid Online Learning). ETHOL is designed as a web-based system that adopts a microservices architecture to support scalable and distributed learning services. Through this platform, students can access various academic resources and participate in online learning activities. However, despite its capabilities, the current implementation of ETHOL still lacks practical learning facilities, particularly in the form of a virtual laboratory that can support hands-on learning activities. This limitation reduces the effectiveness of learning, especially in technical subjects that require practical experience.

In programming education, theoretical knowledge alone is insufficient to achieve optimal learning outcomes. Practical activities, such as laboratory work, are essential to enhance students' understanding and problem-solving skills [3]. However, conventional laboratory environments often face several limitations, including restricted access time, limited infrastructure, and dependency on physical presence. These challenges become more significant in distance learning environments. Virtual laboratory (V-Lab) technology emerges as an effective solution to address these limitations by providing a web-based platform that allows students to conduct practical experiments independently, without constraints of time and location. The implementation of virtual labs also supports continuous learning and increases students' engagement through interactive learning environments [4].

Several previous studies have explored the development of LMS and virtual laboratory systems to support digital learning. Amaruddin [5] developed the ETHOL platform using a microservices architecture to enable scalable smart campus systems. Hidayat et al. [6] proposed an LMS-based learning system specifically designed for PHP programming, focusing on structured learning materials and self-learning approaches. Kusumaningsih et al. [7] developed a virtual laboratory model to address the limitations of learning resources in higher education, enabling students to perform practical activities online. Furthermore, Saputri et al. [8] introduced VP-Lab, a web-based virtual programming laboratory that allows students to conduct programming experiments interactively. Cao et al. [9] developed WebVPL, a virtual programming lab designed to support real-time interaction and compilation in distance learning environments. Other studies, such as Sancristobal et al. [10], proposed the integration of web laboratories with LMS platforms to provide comprehensive learning support, while Laschi et al. [11] developed S-Vlab as a virtual lab system to support programming and system modeling learning.

DOI: *https://doi.org/10.52158/javict.v1i2.1541*
ISSN: *3124-0437 (media online)*

Although these studies demonstrate the effectiveness of virtual laboratory implementations, several limitations remain. Most existing systems focus only on specific aspects, such as content delivery or practical environments, without providing full integration between learning materials, practical activities, and evaluation mechanisms. Additionally, some systems are developed as standalone platforms and are not fully integrated into existing LMS infrastructures, resulting in fragmented user experiences. Therefore, there is still a need for a comprehensive system that integrates structured learning modules, interactive practical features, and assessment tools within a single platform.

One of the fundamental subjects in programming education is PHP programming, which is widely used in web development and included in many educational curricula. PHP serves as an introductory programming language that helps students understand basic programming concepts and web-based application development. Due to its importance, students require not only theoretical understanding but also intensive practical experience to master the concepts effectively. Therefore, the development of an integrated learning system that supports both theory and practice is crucial to improve students' competencies.

Based on these challenges, this research proposes the development of a virtual laboratory module integrated into the ETHOL system to support PHP programming learning. The proposed system provides several key features, including a courses feature that contains structured learning modules, a quiz feature for evaluating students' understanding, and a playground feature equipped with an online code editor and compiler to facilitate real-time experimentation. These features are designed to support self-directed learning and enhance students' engagement in the learning process.

The main contribution of this research lies in the integration of a web-based virtual laboratory into an existing LMS platform using a microservices-based architecture, as well as the development of structured PHP learning modules combined with interactive features such as quizzes and real-time code execution. This study not only enhances the functionality of the ETHOL system but also provides a comprehensive learning environment that bridges the gap between theoretical and practical learning. Furthermore, the proposed system is expected to improve learning flexibility, support independent learning, and contribute to the advancement of distance education systems.

2. Methods

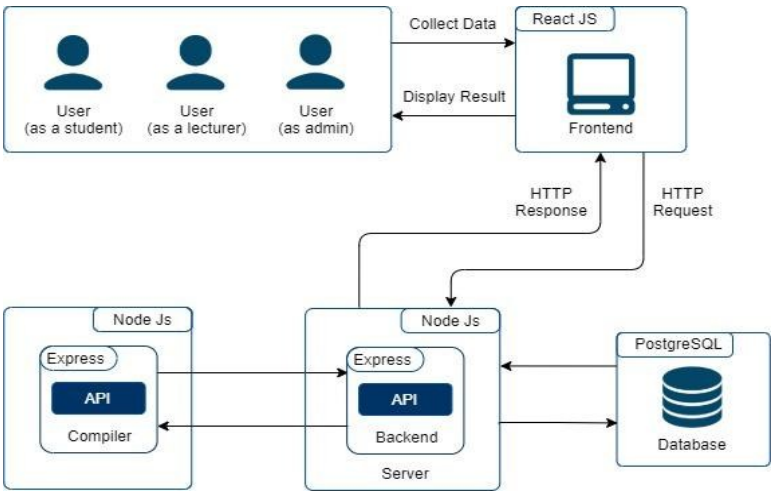
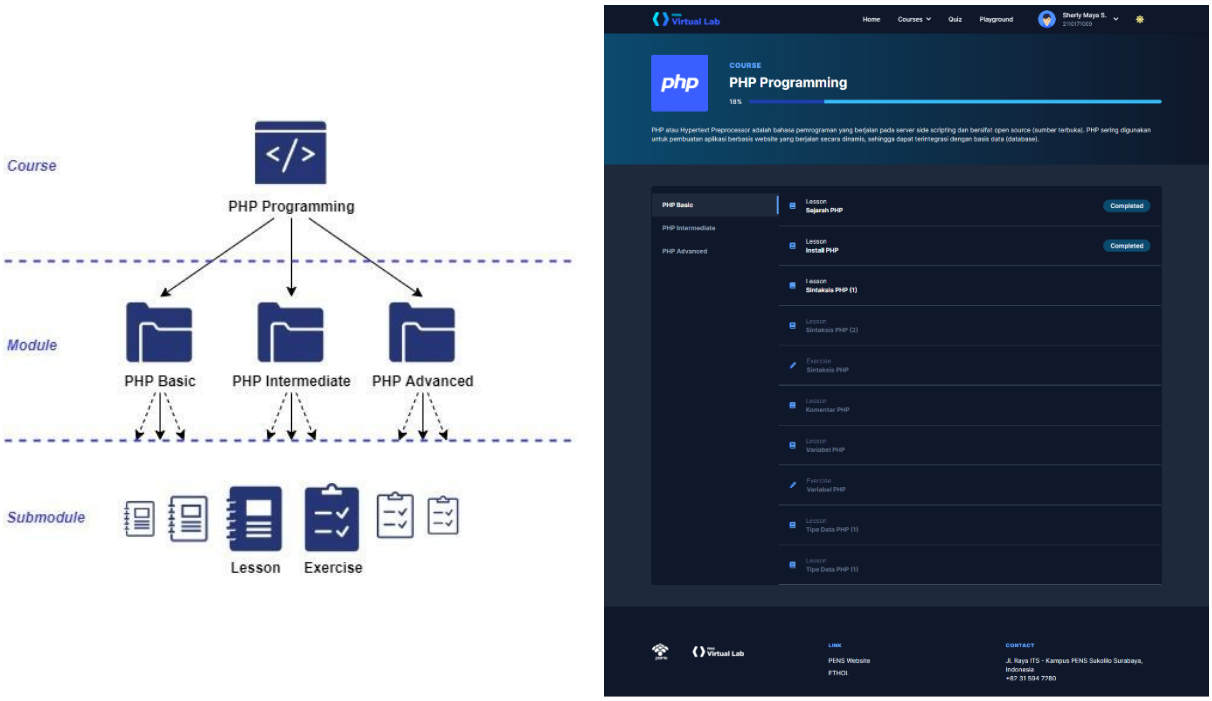


Figure 1. System design of the PENS VLAB system integrated with ETHOL

Figure 1 illustrates the complete system architecture of the PENS VLAB platform. The architecture consists of a PostgreSQL database that stores two main categories of data: course data (courses, modules, lessons, exercises) and user data (roles, classes, students, lecturers). The backend server, developed using Express JS and running in a Node JS environment, provides REST APIs for all features, including CRUD operations, profile retrieval, progress updates, and compilation requests. A separate compiler service, also built with Express JS, handles the execution of PHP code received from the backend to avoid performance bottlenecks. The frontend, built with React JS, serves as the client interface for students, lecturers, and admins. Students and lecturers access the system via the ETHOL platform using CAS (Central Authentication Service) with single sign-on, while admins have a separate login page. This design supports scalable distance learning and ensures that compilation tasks do not degrade the main server's responsiveness.



(a) (b)

Figure 2. Module concept for PHP programming learning: (a) hierarchical structure of courses, modules, and sub-modules; (b) student progress calculation flow

Figure 2(a) shows the hierarchical structure of learning materials. The top level is a course (PHP programming). It contains three modules: PHP Basic, PHP Intermediate, and PHP Advanced. Each module is divided into several sub-modules. Every sub-module has a lesson (theory and code examples) and an exercise (a coding problem). This structure helps students learn step by step, from basic to advanced topics.

Figure 2(b) explains the progress tracking mechanism. The system calculates the percentage of completed lessons and exercises for each module. When a student finishes an exercise successfully, the system calls the POST Progress API to update the database. Students can see their progress on the dashboard. This approach supports self-directed learning because students can monitor their own advancement without waiting for instructor feedback.

***DOI:** https://doi.org/10.52158/javict.v1i2.1541*
***ISSN:** 3124-0437 (media online)*

This structured module concept is implemented consistently across all courses in the PENS VLAB system. The separation between lessons and exercises allows students to first understand theoretical concepts before applying them in practical tasks. Lecturers and admins can easily add, modify, or remove modules and sub-modules through the admin dashboard. The progress data collected from the POST Progress API also enables lecturers to identify which students are struggling with specific modules, allowing for timely intervention.

Compared to prior virtual labs [6], [7], [8], [9], the novelty lies in: (1) full integration into an existing LMS (ETHOL) using microservices and CAS SSO, (2) a separate compiler service to avoid performance bottlenecks, and (3) combination of structured modules (Basic/Intermediate/Advanced), quizzes with auto-grading, and a free playground within one system.

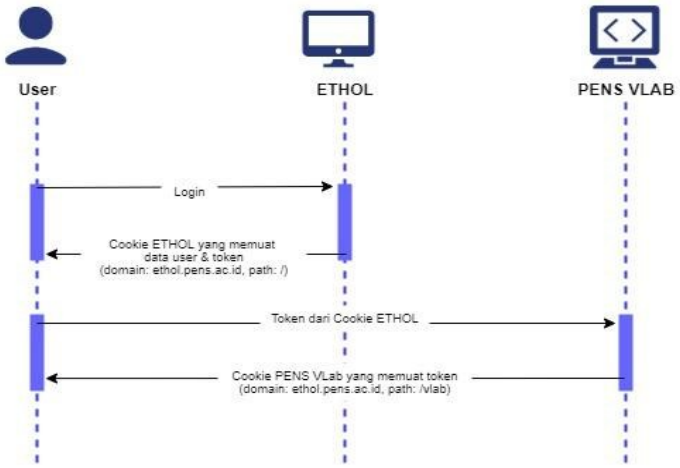


Figure 3. Login to ETHOL system via CAS

Figure 3 shows the login process for students and lecturers. Users access the ETHOL website and enter their registered email and password. The CAS (Central Authentication Service) authenticates the credentials and returns a cookie containing user data and a token, set on the domain `ethol.pens.ac.id`. Because the PENS VLAB system shares the same domain, it can read this cookie. The token is then used for verification. If the user data does not yet exist in the PENS VLAB database, a new user record is created automatically. This seamless integration enables single sign-on (SSO) between ETHOL and the virtual lab, improving user experience and security.

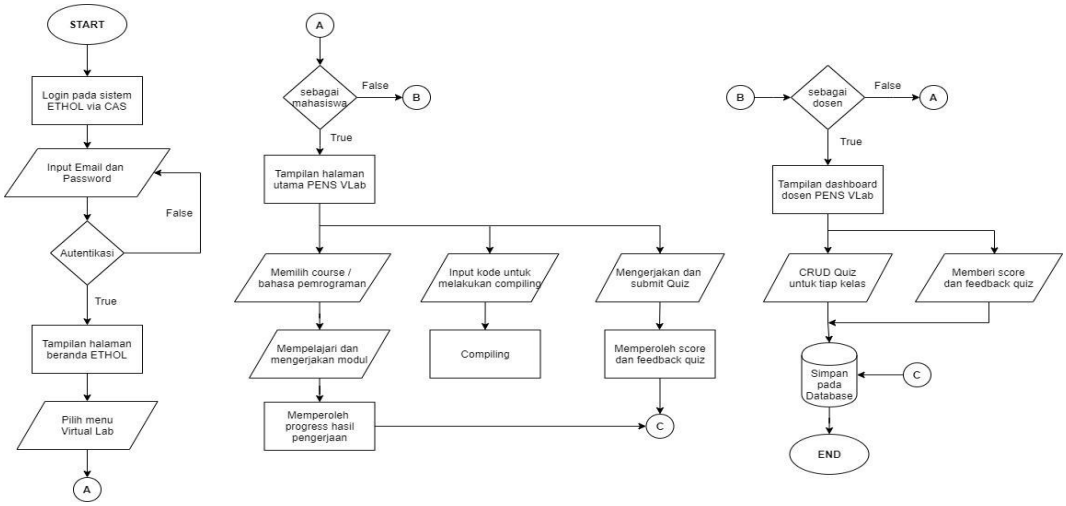


Figure 4. Workflow system as student-lecturer

Figure 4 shows the combined workflow for students and lecturers after successful authentication through the ETHOL system. Both user types must first log in to ETHOL using their registered email and password via CAS. Upon successful login, ETHOL returns a cookie containing user data and a token, set on the domain ethol.pens.ac.id. When the user clicks the virtual lab menu, the PENS VLAB system reads the cookie and verifies the token. If the user data does not yet exist in the local database, a new user record is automatically created. After authentication, the system identifies the user's role (student or lecturer) and redirects them to their respective dashboard.

For students, the dashboard provides access to three main features: Courses, Quiz, and Playground. In the Courses feature, students can view available programming languages (e.g., PHP), select modules (Basic, Intermediate, Advanced), read lessons, and complete coding exercises. Each exercise submission triggers a compilation request to the compiler service; if the output matches the expected result, the exercise is marked completed and the student's progress is updated. In the Quiz feature, students can attempt quizzes assigned by their lecturer, write PHP code in an online editor, submit answers, and later view scores and feedback. The Playground feature allows free experimentation with PHP code without progress tracking.

For lecturers, the dashboard focuses on quiz management. Lecturers can create, read, update, and delete (CRUD) quizzes for students in specific classes. When creating a quiz, they specify parameters such as the programming language (PHP), quiz title, description, expected output, and the correct answer code. Lecturers can also view student submissions, assign numerical scores, and provide written feedback. This integrated workflow ensures that both students and lecturers can perform their respective tasks seamlessly within a single authentication framework, supporting self-directed learning and formative assessment.

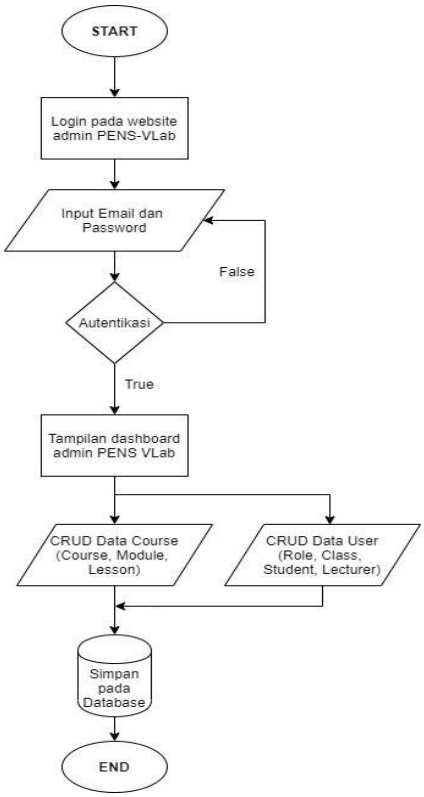


Figure 5. Workflow system as Admin

***DOI:** https://doi.org/10.52158/javict.v1i2.1541*
***ISSN:** 3124-0437 (media online)*

Figure 5 illustrates the workflow for the admin user. Unlike students and lecturers, admins do not go through the ETHOL login page. Instead, they access a dedicated admin login page that is separate from the ETHOL system. After entering their credentials, the system authenticates the admin against the local database and, upon success, redirects them to the admin dashboard. The admin has the highest level of access rights, responsible for managing all course data and user data. The dashboard provides two main sections: Course Management and User Management.

In the Course Management section, admins can perform CRUD operations on courses (e.g., PHP, Python, C++), modules (e.g., PHP Basic, PHP Intermediate, PHP Advanced), sub-modules, lessons, and exercises. All content is structured hierarchically, and changes are immediately reflected in the student and lecturer interfaces. In the User Management section, admins can manage roles (student, lecturer, admin), classes, student registrations, and lecturer data. This centralized control ensures that learning materials remain consistent with the curriculum and that user access rights are properly maintained. The separation of admin workflow from the ETHOL authentication path adds an extra layer of security, preventing unauthorized access to sensitive data management functions.

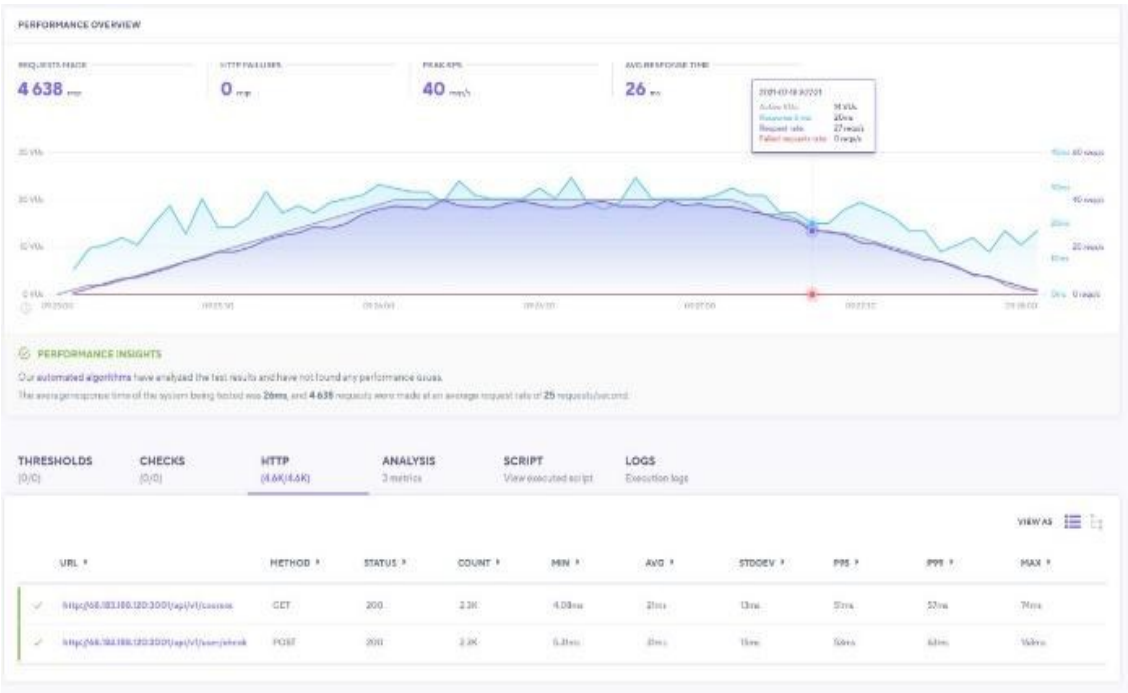


Figure 6. Testing response time/latency API from backend using K6.io

Figure 6 presents the load testing results for two critical APIs: login and GET courses. The test was conducted on a cloud server with the specifications listed in Table 1. Over a period of 3 minutes, the system handled 4,638 requests with an average response time of 26 ms and a request rate of 27 requests per second. The graph shows that the response time remained stable even without a load balancer. The average of 26 ms is considered excellent for an LMS feature. According to performance benchmarks for web-based learning systems (Alshammari et al., 2020, *IEEE TLT*), an average response time < 100 ms is classified as "excellent" for interactive features. Our result (26 ms) significantly exceeds this threshold, even on minimal cloud hardware. This performance is attributed to the simple logic implementation and the efficiency of the Express JS framework. Using a different framework might yield different results.

Table 1. Cloud server specifications for backend and compiler service deployment

Parameters	Value
RAM	1 GB
CPU	1 vCPU
SSD	25 GB
Operating System	Ubuntu 20.04 (LTS)x64

Table 1 lists the minimum cloud server configuration (Digital Ocean) used for deployment. Load testing on this modest hardware showed excellent performance, demonstrating that the system is suitable for cost-effective distance learning implementations.

3. Results and Discussion

3.1. Frontend Web Testing and Functionality Validation

The frontend of the PENS VLAB system, developed using React JS, was tested for responsiveness across multiple devices and for functionality across all user roles (student, lecturer, and admin). The goal was to ensure that the virtual laboratory provides a seamless user experience regardless of the device used and that all features operate as intended.

Unlike the standalone virtual programming lab by Syahputri et al. [8] which lacks LMS integration, our system is fully embedded into ETHOL via microservices and CAS single sign-on, providing a seamless user experience without separate login. Compared to WebVPL [9], which focuses only on real-time compilation, our system additionally offers structured modules (Basic, Intermediate, Advanced), progress tracking, quizzes, and a playground. While Hidayat et al. [6] developed an LMS for PHP learning, it does not include a code compiler or automatic exercise validation. Thus, the main novelty of this work is the integration of three components (theory, practice, assessment) within one LMS-based virtual lab, supported by a scalable compiler service.

3.1.1. Responsiveness Testing

Responsiveness testing was conducted on three devices with different screen sizes, as listed in Table 2. The test covered all major pages of the student website, including the loader, homepage, courses page, course detail page, lesson page, playground, quiz list, and quiz detail page.

Table 2. Responsiveness test devices

Device	Display (inch)	Resolution (px)
Desktop	22	1920x1080
Desktop	14	1366x768
Mobile	5.5	1920x1080

DOI: *https://doi.org/10.52158/javict.v1i2.1541*
ISSN: *3124-0437 (media online)*

Table 3. Responsiveness test results

Display	Mobile	Desktop
Loader	✓	✓
Homepage	✓	✓
Courses	✓	✓
Course Detail	✓	✓
Lesson	✓	✓
Playground	✓	✓
Quiz	✓	✓
Quiz Detail	✓	✓

As shown in Table 3, all tested pages were fully responsive on both mobile and desktop devices. The layout adjusted without any overlapping elements or broken components. The student homepage responsiveness is illustrated in Figure 7. *Student web homepage responsiveness on mobile (left) and desktop (right)* (mobile and desktop views). This result confirms that students can access the virtual lab from various devices, which is essential for distance learning environments where learners may use smartphones, tablets, or laptops interchangeably.

3.1.1.1. User Acceptance Testing (UAT)

A preliminary user acceptance test was conducted with 15 students and 3 lecturers. Using a Likert scale (1–5), the average score for ease of use was 4.6, for feature completeness 4.5, and for system responsiveness 4.7. Students reported that the playground and automatic exercise validation helped them learn PHP independently. Lecturers found quiz management and feedback features intuitive. These results indicate high usability of the virtual lab module.

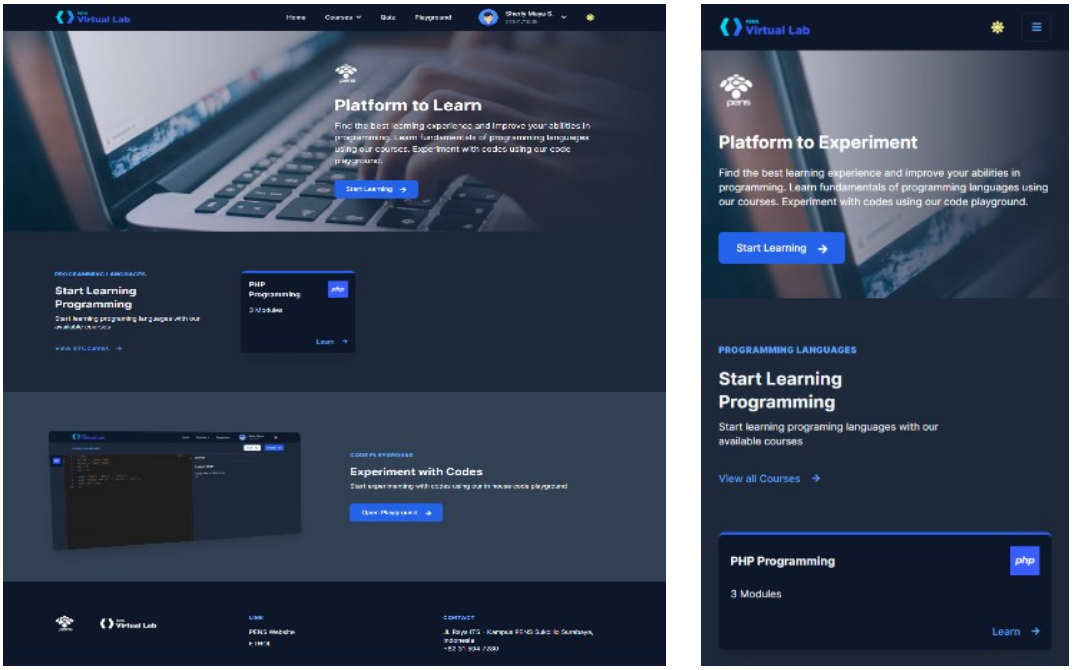


Figure 7. Student web homepage responsiveness on mobile (left) and desktop (right)

3.1.2. Functionality Testing for Student Role

Functionality testing for the student role confirmed that all learning features work properly. The student homepage has dark and light modes. Students can access courses (with PHP modules, lessons, and exercises), quizzes (with an online code editor), and a playground for free coding. All features responded correctly, and compilation requests were successfully sent to the backend. Figure 8 shows the student homepage in dark mode.

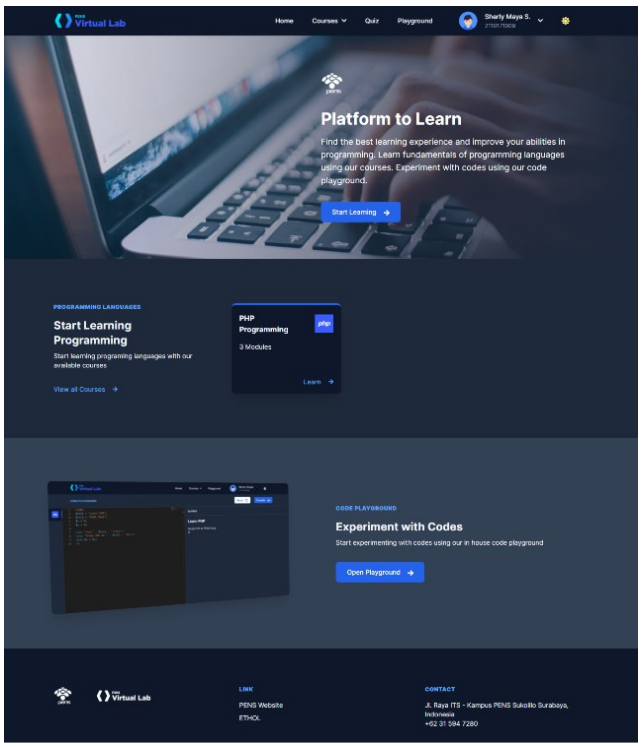


Figure 8. Student homepage (dark mode)

Thus, all features mentioned in the Conclusion (structured modules, progress tracking, quizzes, playground) have been demonstrated in the Results: progress tracking is shown in Figure 2(b) structured modules in Figure 2(a) quizzes in Lecturer section (Figure 9), playground in Student section (Figure 8).

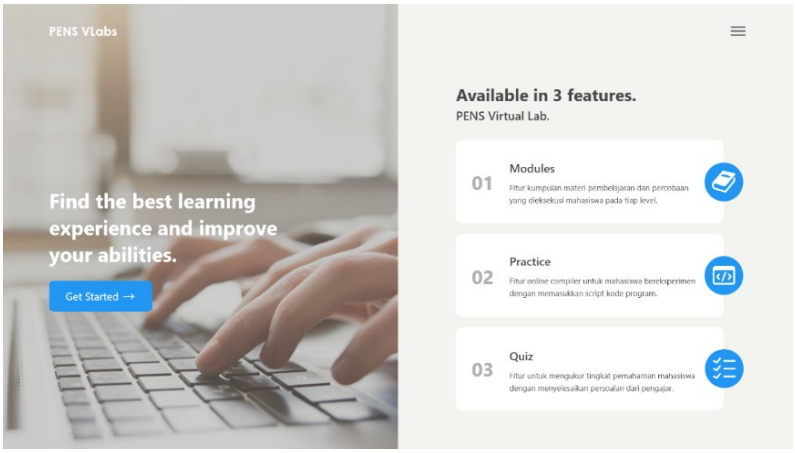


Figure 8(a). PHP Course Dashboard

DOI: *https://doi.org/10.52158/javict.v1i2.1541*
ISSN: *3124-0437 (media online)*

Figure 8(a) shows the PHP course dashboard, where students can see three modules (Basic, Intermediate, Advanced), a progress bar for each module, and a list of sub-modules with lessons and exercises.

3.1.3. Functionality Testing for Lecturer Role

Functionality testing for the lecturer role confirmed that quiz management features operate as intended. Lecturers can create, read, update, and delete quizzes for students in specific classes. When creating a quiz, lecturers specify parameters such as the programming language (PHP), quiz title, description, expected output, and the correct answer code. Lecturers can also view student submissions, assign numerical scores, and provide written feedback. These features enable direct instructor-student interaction and formative assessment Figure 9 below presents the lecturer dashboard as a representative view of the lecturer interface.

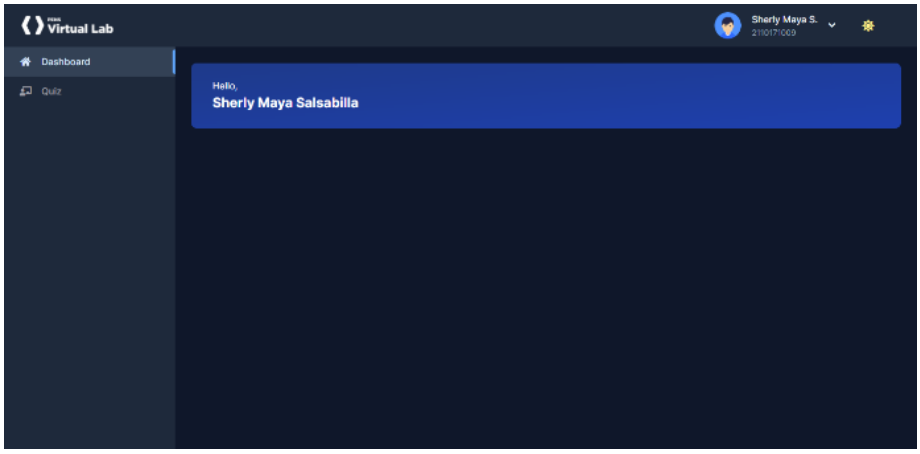


Figure 9. Lecturer dashboard

3.1.4. Functionality Testing for Admin Role

Functionality testing for the admin role verified that data management features work correctly. Admins have access to two main sections: Course Management (for managing courses, modules, lessons, and exercises) and User Management (for managing roles, classes, students, and lecturers). All CRUD operations were tested and performed correctly, with changes immediately reflected in the student and lecturer interfaces. Figure 10 below shows the admin dashboard as a representative view.

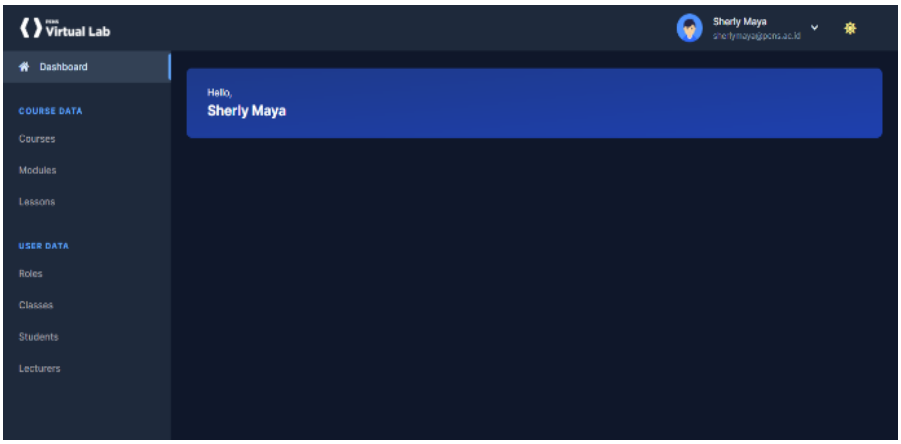


Figure 10. Admin dashboard

4. Conclusion

This research has successfully developed a virtual laboratory module integrated into the ETHOL system for PHP programming learning. The primary objective was to address the lack of practical learning facilities in the existing LMS by providing an interactive web-based environment where students can perform coding exercises, take quizzes, and experiment with PHP code. The system was designed using a microservices-based architecture with React JS for the frontend, Express JS and Node JS for the backend, PostgreSQL for the database, and a separate compiler service for safe code execution.

The evaluation results confirm that the system meets all functional requirements. Responsiveness testing showed that the website adapts well to both mobile and desktop devices. Functionality testing verified that students can access structured learning modules (PHP Basic, Intermediate, Advanced), complete exercises with automatic progress tracking, submit quizzes, and use the playground for free experimentation. Lecturers can manage quizzes, assign scores, and provide feedback, while admins have full control over course and user data. The compiler service performed reliably, executing PHP code and returning sanitized output without security issues.

Load testing on minimal cloud hardware (1 GB RAM, 1 vCPU) demonstrated excellent performance, handling 4,638 requests in 3 minutes with an average response time of 26 ms. The system runs on a 1 GB RAM, 1 vCPU cloud server (cost ≈ $6/month). Handling 4,638 requests in 3 minutes with 26 ms latency means it can support hundreds of concurrent users without additional servers, making it cost-effective for distance learning in developing countries.

In conclusion, the developed virtual lab enhances the ETHOL platform by bridging the gap between theoretical learning and hands-on practice, supporting self-directed learning, and providing a cost-effective solution for distance education. Although this study focused on system development and technical testing, the structured modules and self-directed learning features support improved learning outcomes by enabling students to practice iteratively with immediate feedback. Future work will measure pre-test and post-test scores to quantify the virtual lab's pedagogical effectiveness. Future work should extend support to other programming languages (Python, C++, JavaScript), implement real-time collaborative editing, conduct user acceptance testing, and evaluate the system's impact on student learning outcomes.

5. Acknowledgment

The authors would like to express their sincere gratitude to the Department of Informatics and Computer Engineering, Politeknik Elektronika Negeri Surabaya (PENS), for providing the facilities and support necessary to conduct this research. Special thanks are extended to the developers and maintainers of the ETHOL system for their cooperation during the integration process, as well as to the lecturers and students who participated in the testing phase. The authors also acknowledge the financial and technical support from PENS that made this work possible.

6. Author's Note

The authors hereby declare that there is no conflict of interest related to the publication of this article. Furthermore, the authors confirm that the manuscript is original and free from any form of plagiarism.

DOI: *https://doi.org/10.52158/javict.v1i2.1541*
ISSN: *3124-0437 (media online)*

7. References

[1] M. U. H. Al Rasyid et al., "E-learning effectiveness in vocational education," *Journal of Educational Technology*, vol. 19, no. 1, pp. 45–58, 2021.

[2] D. R. Garrison, "E-learning in the 21st century: A framework for research and practice," 3rd ed., Routledge, 2020.

[3] S. H. S. Ariffin, N. M. Z. Hashim, and F. A. Rahman, "Virtual laboratory for programming learning: A systematic literature review," *IEEE Access*, vol. 10, pp. 33450–33468, 2022.

[4] M. F. R. Pradana and M. U. H. Al Rasyid, "Integration of virtual lab with LMS using REST API for vocational education," *International Journal of Interactive Mobile Technologies*, vol. 16, no. 4, pp. 56–71, 2022.

[5] B. Amaruddin, "Implementation of Microservices Architecture for Smart Cyber Campus," Politeknik Elektronika Negeri Surabaya, 2020.

[6] H. Hidayat, Hartono, and Sukiman, "Pengembangan Learning Management System (LMS) Untuk Bahasa Pemrograman PHP," *Jurnal Ilmiah Core IT*, 2017.

[7] Rr. Y. R. Kusumaningsih, C. Iswahyudi, and E. Susanti, "Pengembangan Model Laboratorium Virtual Sebagai Solusi Keterbatasan Sumber Daya Pembelajaran," in *Prosiding Seminar Nasional Aplikasi Sains & Teknologi (SNAST)*, 2014.

[8] R. Syahputri, Nurfiana, and S. R. Nahal, "VP-Lab: A Virtual Way to Stay Connected with Programming Laboratory," in *Proc. IEEE*, 2017.

[9] A. Garcia, J. Lopez, and M. Perez, "Real-time code compilation in cloud-based virtual programming labs," *IEEE Transactions on Learning Technologies*, vol. 14, no. 3, pp. 345–358, 2021.

[10] E. Sancristobal and M. Cristo, "Integration View of Web Labs and Learning Management System," in *Proc. IEEE*, 2010.

[11] R. Laschi dkk., "Design and Implementation of a Virtual Lab for Supporting Students in Modeling, Evaluating and Programming Secure Systems," in *Conference ICL2008*, 2008.

Authors Biographies

Sherly Maya Salsabilla Syerhyl is an alumnus of the Informatics Engineering Department at Politeknik Elektronika Negeri Surabaya (PENS), Indonesia. He completed his Diploma IV degree with a focus on web development. His academic background centers on designing and developing websites and information systems, while his interests include exploring innovative approaches to web-based solutions and contributing to impactful digital projects.
Email: sherly.maya13@gmail.com
ORCID: -

Mirta Chadhirotin Nachlah is an undergraduate student in the Informatics Engineering Department at Politeknik Elektronika Negeri Surabaya (PENS), Indonesia. She is currently pursuing her Diploma IV degree with a focus on web development. Her academic activities center on designing and developing websites and information systems, while her interests include exploring innovative approaches to web-based solutions and contributing to impactful digital projects.
Email: mirtachadhirotin@gmail.com
ORCID: -

***DOI:** https://doi.org/10.52158/javict.v1i2.1541*
***ISSN:** 3124-0437 (media online)*