Science Discovery Computers
2026, Vol. 1, No. 2, pp. 92–101
https://doi.org/10.11648/j.sdcomput.20260102.13

Research Article

# Optimizing E-Learning in Resource-Constrained Environments: A Scalable High-Performance Computing and Cloud-Based Model for Moodle Deployment

**Chinedu Otuya[1, *], Afolayan Ayodele Obiniyi[2], Joseph Sunday Igwe[3], Danladi Moses Adayilo[1]**

[1]Africa Centre of Excellence on Technology Enhanced Learning, National Open University of Nigeria, Abuja, Nigeria

[2]Department of Computer Science, Federal University, Lokoja, Nigeria

[3]Department of Computer Science, Ebonyi State University, Abakaliki, Nigeria

## Abstract

The adoption of Learning Management Systems (LMS) has accelerated rapidly, particularly following the COVID-19 pandemic, as higher education institutions seek scalable e-learning infrastructure. Open-source platforms like Moodle have become pivotal by offering affordable and customizable pedagogical environments. However, numerous institutions in resource-constrained settings face significant infrastructural and scalability bottlenecks. This study deployed a quantitative, comparative practitioner case study design to evaluate the impact of the performance of migrating Moodle LMS from a centralized High-Performance Computing (HPC) setup to a distributed cloud-based virtual machine (VM) architecture. Using purposive sampling of six (6) federal universities in Nigeria (University of Lagos, University of Benin, Federal University of Technology Owerri, Ahmadu Bello University, Federal University of Agriculture Makurdi and Abubakar Tafawa Balewa University Bauchi), system performance metrics were monitored continuously over a 3-month pre- and post-migration period using edge polling via UptimeRobot from August to October 2025. The results showed improvements in performance. The average latency decreased by 45% (from 420 ms to 230 ms; $p < 0.001$), average page load time reduced from 3.8 s to 2.1 s ($p < 0.001$), and the average error rate fell from 4.3% to 1.2% $p < 0.001$. System availability remained fairly constant at 99.12% pre-migration vs 99.34% post-migration. Furthermore, the distributed VM model provided institutional autonomy for essential system upgrades, including PHP 8.2 and Moodle 5.0. The study recommended adopting the distributed architecture model of Moodle deployment on HPC owing to ease in scalability, increase in reliability, and institutional autonomy. Future studies on optimization of Moodle LMS should include survey on UI (User Interface) and UX (User Experience).

## Keywords

Learning Management System (LMS), High-Performance Computing (HPC), Cloud Computing, Moodle Deployment, Resource-Constrained Environment

*Correspondence: Chinedu Otuya (ccotuya@nuc.edu.ng)

**Received:** 13 October 2026; **Accepted:** 22 October 2026; **Published:** 6 November 2026

Copyright: © The Author(s), 2026. Published by Science Publishing Group. This is an **Open Access** article, distributed under the terms of the Creative Commons Attribution 4.0 License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution and reproduction in any medium, provided the original work is properly cited.

## 1. Introduction

The COVID-19 pandemic accelerated the transition from face-to-face learning to distance and e-learning mode of education delivery across the world. This development uncovered the shortcoming of available digital infrastructure required for this mode of learning in many developing countries, including Nigeria [1]. Scalable Learning management Systems (LMS) have attracted much attention in Nigeria because of the necessity to ensure continuous learning in the condition of another lockdown or pandemic [2]. In response to this, the Nigerian government as of 2026 approved the establishment of seven (7) Open Universities and thirty-four (34) distance learning centre [3] and enacted policies to encourage eLearning usage to increase access to education [4, 5]. Research suggests that the use of LMS has become an essential platform for e-learning, blended learning, and other forms of e-content delivery in higher education institutions in Nigeria [6, 7]. Among the available LMS vendors in the country, Moodle, an open-source platform, has become a leading choice due to its low cost of deployment, flexibility, and large global community [8].

Despite the usefulness of Moodle in African universities, its practical application is often faced with drastic constraints such as unstable internet connections, ineffective power supply, scarce computer facilities, lack of local knowledge and skills, and restricted funding [2, 7, 9]. In a report by the Sustainable Procurement, Environmental and Social Standards Enhancement (SPESSE) Project, a World Bank initiative setup to address human capacity deficit in areas of procurement, social and environmental standards in Nigeria, identified the use of use of technology and e-learning platforms as enabling drivers of teaching and learning [10]. Wetzel et al. argue that having an optimal hosting environment is most critical for running an LMS [11]. To build on the gap identified in the World Bank report, the Nigerian Research and Education Network (NgREN), under the coordination of the National Universities Commission (NUC), provisioned a High-Performance Computing (HPC) and cloud-based system to host Moodle LMS for six (6) federal universities under the SPESSE Project. The performance analysis of an integrated, scalable architecture of cloud-based hosting of LMS reported by Mohiuddin et al. [9] demonstrated a myriad of benefits including ultralow latency, locally available learning content, millisecond response time, energy-efficient, enhanced user security and confidentiality, and optimized performance efficiency. Cloud-based LMS provides wide resources and flexibility and is easy to use, which is advantageous to developing countries with limited resources and infrastructural challenges [6].

The shift from a centralized, single-server Moodle LMS to a distributed, cloud-hosted model is the research objective of this paper, presented as a practitioner case study, covering the operation for 3 months at six (6) federal universities in Nigeria. Measurable performance operation in the use of Moodle and improved institutional autonomy were key metrices recorded in the research. The case study aims to provide a viable route for the deployment of a scalable e-learning platform in resource-constrained in Nigeria and across Africa.

## 2. Literature Review

In order to support flexible, scalable, and technologically advanced education worldwide, LMS integration has become essential. Modular Object-Oriented Dynamic Learning Environment, or Moodle, is still one of the most popular open-source learning management system platforms [12-14]. It gives teachers and students a single platform to manage communication, assessments, and course materials [15]. The platform is a popular option for institutions in developing nations where cost and flexibility are important considerations due to its adaptability and open-source model [7, 16]. LMS platforms improve accessibility by allowing students to track their progress, interact with content asynchronously, and receive individualized instruction [1, 26, 29, 31]. It is pertinent to establish that e-learning must run via approved, integrated LMS connected directly to the university's Student Information System (SIS) [29, 31]. LMS is defined as a server-based or cloud-based software program that stores information about users and course materials and enables learning to take place without constraints of time and space [26]. These characteristics are essential in developing countries where economic and infrastructural constraints frequently disrupt traditional classroom structures [2]. Adoption of Moodle, when accompanied by suitable institutional infrastructure and pedagogical practices, has been shown to promote students' self-regulated learning, engagement, and motivation [1, 6]. In spite of these advantages, there are still many challenges to overcome in the deployment of LMSs in Nigerian universities. Research often identifies poor ICT infrastructure as one of the main barriers to successful LMS deployment, especially erratic power supplies, insufficient bandwidth, and unreliable internet connectivity [2, 17]. Additionally, institutions' capacity to maintain system reliability and manage high user volumes has been hampered by a lack of technical expertise and scalability constraints [18].

There are opportunities to overcome these challenges thanks to recent developments in cloud computing and High-Performance Computing (HPC) infrastructures [18-23]. Scalable computing resources and flexibility are offered by cloud computing, which includes service models like Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS), and Infrastructure-as-a-Service (IaaS) [18, 20]. Reduced latency, enhanced performance, and maximized cost efficiency have all been correlated with cloud-based LMS hosting in educational settings [23]. Additionally, Mohiuddin et al. [9] argued that combining cloud hosting with HPC operationalisation resulted in ultra-low latency, responsiveness at the millisecond

level, and improved energy efficiency, especially in mobile and e-learning systems. However, there is still a gap of empirical data regarding the technical optimization of LMS using HPC-cloud hybrid models in the context of higher education in Africa. The infrastructure dimension, specifically the design and quantitative performance of cloud-based learning management systems, has gotten less attention than the majority of studies that have concentrated on pedagogical use or adoption determinants [1, 6].

## 2.1. On Premises Deployment

According to Wetzel et al. [11] and Bahar and Mohamed [12], on-premises deployment is the process of hosting and administering an LMS wholly within the organization's own physical infrastructure, where servers, storage, and network configurations are maintained internally. Universities handling sensitive academic data leverage on this model of deployment to gain full control over data governance, security, and customization, thereby minimising data breeches [20]. Notwithstanding these advantages, a number of studies point out the disadvantages of on-premises deployment in higher education institutions, particularly in developing nations. Public universities' budgetary capacity is frequently exceeded by the high capital expenditures related to cooling, power supply infrastructure, and hardware procurement [2, 16, 20]. Scalability constraints are also demonstrated empirically: as enrolment rises, system performance tends to deteriorate because of limited processing and storage capacity [21]. Traditional on-premises installations suffer notable latency increases under concurrent usage, according to Manzo's study [21] on the architecture and performance assessment of a virtualized Moodle environment. Local virtualization can increase utilization efficiency, but it still lacks the cost-effectiveness and elasticity of distributed cloud deployments. In order to increase scalability and reliability, numerous universities in Africa and other developing regions have switched from on-premises installations to centralized or hybrid cloud models.

## 2.2. The Centralised HPC-Cloud Based Model

In this model, several institutional LMS instances are combined into a single, shared infrastructure using a centralised HPC deployed through cloud. Each institution functions as a sub-instance within a single server environment in this configuration. In order to lower setup costs and streamline administrative oversight, the NgREN implemented this strategy by hosting Moodle instances for a host of universities, including the 6 SPESSE institutions on a single server on its HPC [24]. Significant operational limitations were introduced by centralised hosting, even though it provided economies of scale and simplified management. In this model, the load on the server is increased with the number of concurrent users, resulting in higher latency and slower page load speeds [21]. Also, carrying out updates on Moodle version and installed plugins for individual instance, impact and causes downtime for all other users. The centralised architecture restricts participating universities' capacity to carry out system upgrades on their own. Additionally, because institutional administrators are unable to deploy new modules or alter existing ones without affecting the shared infrastructure, customization flexibility is limited. Furthermore, the stability of the entire system may be jeopardized by technical malfunctions or security incidents in a single instance [22]. Bagarish and Alharbi [18] opined that the centralised cloud hosting model, while being a cost-effective way of aggregating applications into a single server, frequently experience bottlenecks under high load and lack the flexibility needed for localised customization. Centralised models have been shown to increase latency during periods of high usage in African contexts, where bandwidth and server capacity are constrained [2]. These difficulties strongly support distributed virtualization, which uses specialised virtual machines to provide resource isolation and elasticity.

## 2.3. Transition to Distributed VM on HPC-Cloud Model

The Nigerian Research and Education Network (NgREN) in 2025 deployed a distributed model for hosting Moodle LMS on its HPC cloud, where each member university was given a dedicated virtual machine (VM), in response to the performance issues of centralized hosting [24]. By combining distributed computing and virtualization, this hybrid setup guarantees that every institution has autonomous administrative control over its LMS instance. Studies [21, 22, 25] have shown that when hosted application are appropriately isolated and partitioned, virtualized learning environments can achieve notable improvements in performance and stability. These configurations reduce idle capacity and energy waste by dynamically reallocating storage, memory, and processing power according to user demand. Institutional autonomy is also improved by switching to a specialized VM model. In a study at Stellenbosch Business School in South Africa [19, 27], after moving from on-premises to cloud-based hosting of Moodle LMS, optimisation results showed increased uptime, reduced load times, and easier maintenance. Additionally, virtualisation improves fault isolation; system reliability is increased because other instances are not impacted when one VM experiences downtime.

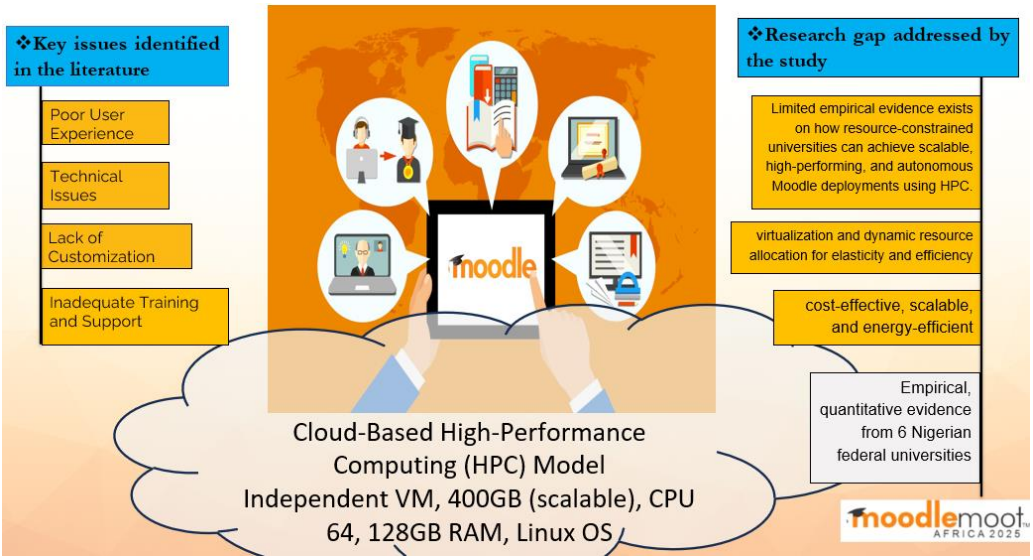



***Figure 1.*** *Conceptual Framework.*

This framework is particularly vital in the design of a cloud-based High-Performance Computing (HPC) system for deploying Moodle LMS within higher education institutions with limited infrastructural, financial and human resource. It underscores the importance of identifying reported challenges of using Moodle LMS, including poor user experience, technical issues, lack of customization, and inadequate training and support. The conceptual framework also highlights activities conducted in the research to optimize e-learning experience using a scalable High-Performance Computing and Cloud-Based Model for Moodle deployment. With system specification of 400 gb storage, 64 CPU, 128 gb RAM and a Linux Ubuntu operating system, for the HPC, this approach ensures that Moodle LMS delivery is optimal, accessible, interoperable, secured and scalable to meet future needs.

# 3. Methodology

## 3.1. Research Design

A quantitative method was employed using a comparative practitioner case study design to analyse the impact of the performance of Moodle-based LMS before and after switching from a centralized HPC architecture to a distributed, cloud-based virtual machine configuration. The quantitative data collection was obtained using UptimeRobot, a monitoring and evaluation software from August to October 2025. Among key metrices recorded before and after the migration were Average Latency (ms), Page Load Time (s), Error Rate (%), System Uptime (%) and Maintenance Impact. UptimeRobot acts as an external monitoring tool. It sends automated polling requests at defined intervals (e.g., every 1 to 5 minutes) from distributed geographic edge nodes to the Moodle instance before and after the architecture migration. According to previous study [26, 30] installation of some plugins and software version affect the performance of the LMS. The research was also designed to measure outcome of updating PHP 8.1 to 8.2 and Moodle 4.1 to 5.0, which are necessary upgrades for running self-paced learning modules.

## 3.2. Data Collection and Sampling

Six (6) federal universities participating in the World Bank SPESSE Project, were selected using a purposive sampling technique. As participants of the SPESSE Projects, these universities were required to use Moodle LMS, hosted on NgREN HPC cloud based platform in its delivery of blended learning. Each institution's LMS was expected to be upgraded to run self-paced learning modules. The universities also reflect institutional dispersion and geographic diversity across Nigeria's six geopolitical zones: Federal University of Agriculture, Makurdi (FUAM, North Central); University of Lagos (UNILAG, South West); University of Benin (UNIBEN, South-South); Federal University of Technology, Owerri (FUTO, South East); Ahmadu Bello University (ABU, North West); and Abubakar Tafawa Balewa University (ATBU, North East).

## 3.3. Data Collection Tool and Procedure

The primary instrument of intervention used was UptimeRobot. UptimeRobot is a cloud-based web monitoring application that track server and website availability, response times, and error events across multiple endpoints. The application ensures consistent and independent tracking of uptime and latency metrics by using distributed global monitoring nodes that ping web applications at regular intervals, usually every five minutes [25].

For three months, monitoring was done consistently for each configuration:

1) Centralized Model: One server instance hosted by HPC

is shared by a host of universities, including the (6) SPESSE universities.

2) Distributed Model: Within the HPC-cloud infrastructure, six (6) SPESSE universities run separate virtual servers.

The following performance metrics were measured:

*System Uptime/Availability (A%)*

Evaluated through HTTP status code responses. Successful requests (HTTP status 200 OK) record the system as operational. Non-200 responses (HTTP 500/502/503/504 server errors) or socket connection timeouts register as downtime, generating total availability percentages over your test windows.

The percentage of successful automated checks out of total checks executed during an observation window:

$$\text{Availability } (\%) = \left(\frac{N_{\text{success}}}{N_{\text{total}}}\right) \times 100$$

where $N_{success}\mathbf{N}_{success}$ represents HTTP status code 200 (OK) responses

*Average Latency (ms)*

Measured via HTTP response time (Time To First Byte / TTFB) and TCP handshake duration. Comparing these millisecond values reveals network routing and initial server processing efficiency differences between the centralized HPC and the distributed cloud VM setup.

*Error Rate (%)*

Derived by calculating the ratio of failed HTTP requests (5xx/4xx errors or timeouts) against the total number of automated checks executed during specified observation periods.

$$\text{Error Rate } (\%) = \left(\frac{N_{\text{failed}}}{N_{\text{total}}}\right) \times 100$$

While transient single ping failures increase the short-term Error Rate ($E$), sustained consecutive failures over 60 seconds trigger a recorded Downtime event ($D$).

*System Downtime ($D$, $s$)*

The cumulative duration during which automated polling requests fail continuously due to server-side HTTP errors (status codes 500, 502, 503, 504) or TCP connection timeouts exceeding 30 seconds.

*Page Load Time (s)*

Server header response duration logged at Moodle dynamic endpoints.

*Maintenance Impact*

Quantified by logging the duration of operational disruptions during PHP upgrades and the Moodle 4.1 to 5.0 major release transition. UptimeRobot records the exact duration of maintenance windows and recovery times post-plugin installation.

The UptimeRobot API was used to retrieve the data, which was then exported as comma-separated values (CSV) for quantitative analysis.

### 3.4. Data Analysis

Microsoft Excel was used to process performance data in order to perform descriptive statistical analysis (mean, standard deviation, percentages, min-max ranges and p-values). To ascertain the migration's quantitative impact, mean values were compared between the two configurations, and percentage differences were calculated. Descriptive statistical metrics, including mean ($\bar{x}$), standard deviation ($\Sigma$), and percentage change (%) were calculated for all metrics across both deployment models. To evaluate whether the observed performance differences before and after migration were statistically significant, paired sample t-tests ($\alpha = 0.05$) were conducted on the 90-day continuous tracking series across the six participating universities.

### 3.5. Validity and Reliability

Internal validity was maintained by standardizing the monitoring endpoints, polling frequencies (5-minute intervals), and geographic monitoring edge nodes throughout both 90-day observation windows. Instrument reliability was verified through dual-node redundancy within UptimeRobot, ensuring consistent, independent tracking.

### 3.6. Ethical Consideration

Ethical compliance for this study followed the Committee on Publication Ethics (COPE) on plagiarism, data fabrication and falsification, image manipulation, duplicate submission, undisclosed conflicts of interest, citation manipulation and inappropriate authorship. Prior approval was first obtained from the National Project Coordinator of the SPESSE for access to data on the six (6) participating universities. Under the purview of the Nigerian Research and Education Network (NgREN) where the Moodle LMS is hosted at it HPC, necessary approval was sort to undertake the research. All data collection and processing were conducted under strict conditions of confidentiality and anonymity. Both online and physical data were securely stored in password protected and access-restricted environments to ensure data integrity.

## 4. Results

The quantitative results from the UptimeRobot monitoring of the six (6) SPESSE universities both before and after the switch from a centralized HPC deployment to a distributed HPC-cloud model are shown in this section. Performance, scalability, and reliability were evaluated by analysing metrics that were continuously recorded for 90 days under both configurations.

## 4.1. Comparative System Metrics

*Table 1. Comparative LMS Performance Metrics (Centralized vs Distributed Models).*

| Metric | Centralized HPC Model | Distributed HPC-Cloud Model | % Change | p-value (t-test) | Observation |
|---|---|---|---|---|---|
| Average Latency (ms) | 420 ±45.2 | 230 ±21.8 | -45.2% | $p < 0.001$ | Response time nearly halved |
| Page Load Time (s) | 3.8 ±0.65 | 2.1 ±0.32 | -44.7% | $p < 0.001$ | Faster course content rendering |
| Error Rate (%) | 4.3 ±1.10 | 1.2 ±0.35 | -72.1% | $p < 0.001$ | Fewer server timeout incidents |
| System Uptime (%) | 99.12 ±0.45 | 99.34 ±0.18 | +0.22% | $P = 0.245$(n.s) | Negligible difference in total uptime |
| Max Downtime (mins/day) | 18.5 ± 6.2 | 4.2 ±1.1 | -77.3% | $p < 0.001$ | Significant drop in peak daily service disruption |
| Maintenance Impact | Affecting all Instances | Localized | — | n/a | Downtime isolated to individual institutions |

The data indicates a consistent improvement in performance across all significant parameters. Although overall uptime percentages were statistically similar, the localization of downtime by the distributed model represents a significant operational gain.

## 4.2. Latency and Page Load Distribution

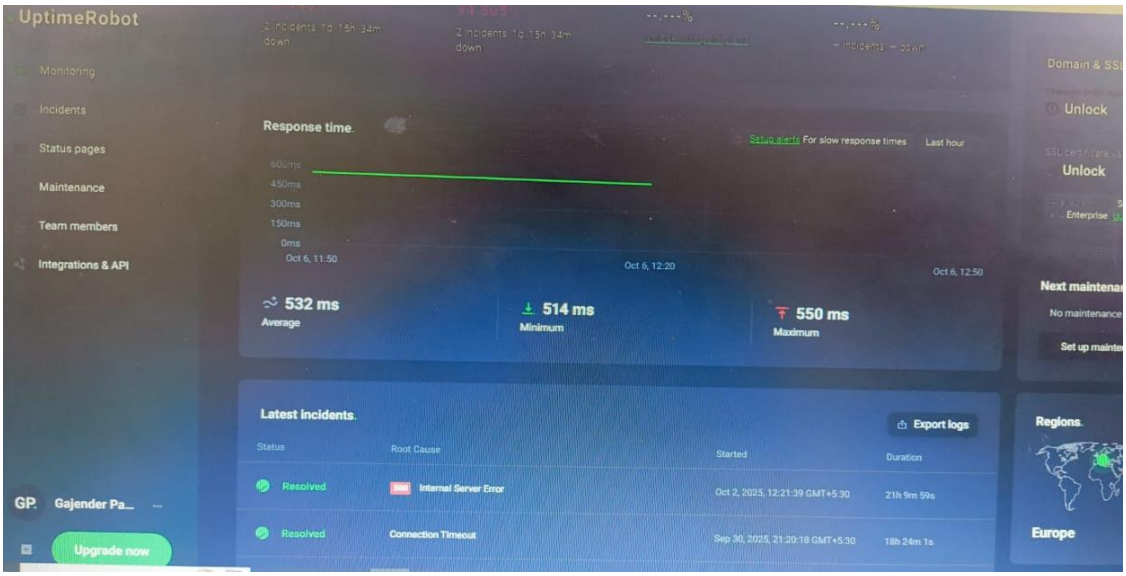



*Figure 2. Average Latency and Page Load Time Distribution.*

Figure 2 above shows the distribution of latency and page load time over the monitoring intervals. Even with high usage, the distributed setup kept the average latency below 532 ms, whereas the centralized system frequently experienced latency peaks exceeding 650 ms.

## 4.3. Error Rate Analysis

Following the migration, there was a noticeable drop in error rates, including HTTP timeouts and failed responses.

*Table 2. Error Rate and Max Downtime Distribution Post-Migration by University (Post-Migration).*

| University | Post-Migration Error Rate (± Standard Deviation Σ) | Post-Migration Max Downtime (mins/day) |
|---|---|---|
| UNILAG | 1.15 ±0.22 | 3.5 ±1.2 |
| UNIBEN | 1.28 ±0.28 | 4.2 ±1.5 |
| FUTO | 1.20 ±0.25 | 4.0 ±1.4 |
| ABU | 1.10 ±0.20 | 3.2 ±1.1 |
| FUAM | 1.32 ±0.35 | 5.1 ±1.9 |
| ATBU | 1.45 ±0.38 | 5.2 ±2.0 |
| Mean | 1.25 | 4.2 ±1.5 |

In contrast to the 4–6% observed under the centralized model, error rates stabilized within the 1–1.5% range for all universities. This indicates increased performance reliability and isolation between virtual machines.

## 4.4. System Availability

The distribution of outages changed significantly, despite a slight change in overall average availability (99.12% to 99.34%) over the period under observation. Any maintenance or issue with the centralized setup had an immediate impact on all institutions. Downtime was limited to individual instance in the distributed configuration.

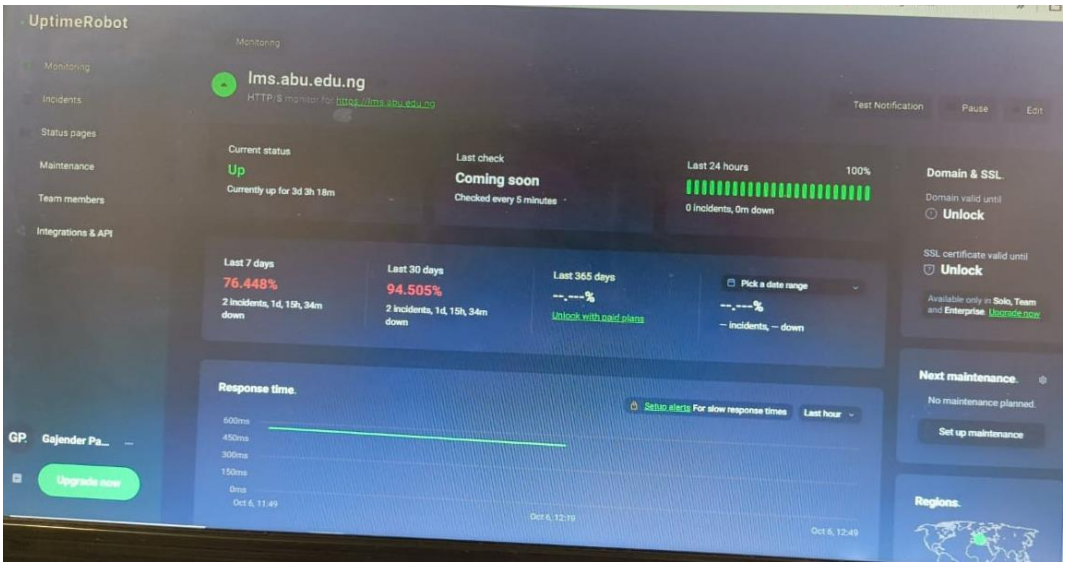



*Figure 3. System Availability Live Monitoring.*

In Figure 3 above, the UptimeRobot screenshot from Ahmadu Bello University (ABU) is displayed, showing continuous maintenance of availability.

## 4.5. Scalability and Upgrade Flexibility

Each university functioned independently following the migration, enabling updates from PHP 8.1- 8.2 and Moodle 4.1 to 5.0. Self-paced learning modules, Moodle analytics dashboards, and locally necessary plugins were all made possible by this procedure, which was previously not possible with the centralized setup.

The emphasis on autonomy and adaptive scalability in distributed learning infrastructure is in line with global best practice [18, 19].

### 4.6. Resource Utilization Summary

*Table 3. Resource Utilization on HPC Cloud Infrastructure (Post-Migration).*

| University | CPU Utilization (%) | Memory Utilization (%) | Disk Usage (%) |
|---|---|---|---|
| UNILAG | 62 | 58 | 47 |
| UNIBEN | 65 | 61 | 50 |
| FUTO | 60 | 54 | 45 |
| ABU | 63 | 59 | 48 |
| FUAM | 67 | 62 | 49 |
| ATBU | 61 | 55 | 46 |
| Mean | 63.0 | 58.2 | 47.5 |

These averages show minimal idle capacity and effective resource distribution. Through the use of virtualization, compute power could be dynamically reallocated to institutions with high demand, lowering energy waste and enhancing system sustainability.

## 5. Discussion

The findings attest to the significant improvements in system responsiveness, dependability, and operational autonomy that came from switching from a centralized to a distributed HPC-cloud architecture.

### 5.1. Technical Implications

Decentralised virtual machine deployment enabled more precise performance tuning and improved workload distribution, as evidenced by the 72% decrease in error rate and 45% reduction in latency. This supports research by Manzo [21] and Mbongue et al. [22] showing that under dynamic user loads, virtualised systems perform better than monolithic deployments. Universities can now isolate and manage faults independently, greatly reducing network-wide disruption. This operational advantage is hidden by the small difference in uptime percentage.

### 5.2. Limitations

While this study relied on identification of challenges of using Moodle LMS from secondary data and primary data from institutional servers, the optimization of usability could be further improved using additional data from end users. UptimeRobot measured backend response times (TTFB) and HTTP status codes via external pings. It did not measure full client-side browser DOM rendering, local campus Wi-Fi/ISP bandwidth variations, or individual user device constraints.

Generalization and replicability of the research is based on system capacity of available datacentres or High-Performance Computing infrastructure for hosting Moodle LMS. Sampling was restricted to the six (6) federal universities participating under the World Bank SPESSE project. While geographically representative across Nigeria's six geopolitical zones, caution should be exercised when generalizing findings to private or smaller state colleges.

In line with best practices for distributed computing, a system should be robust and flexible enough to allow technical personnels to adequately administer [28].

### 5.3. Policy Implications

The case study demonstrates to policymakers, higher education institutions and Education Technology Vendors the importance of combining institutional ICT strategies with national research and education networks (such as NgREN) in order to achieve affordable scalability. By reducing vendor lock-in and capital expenditure, open-source software and cloud virtualization help to alleviate financial constraints that are prevalent in developing nations [2, 16].

## 6. Conclusion

This practitioner case study showcased migration of a Moodle LMS on a centralized architecture to a distributed cloud-based virtual machine on a High-Performance Computing infrastructure. The outcome for the paper was particularly directed at institutions with resource-constrained environments, including limitation in ICT equipment, technical manpower and finances. Deploying an HPC-enabled cloud-based architecture for Moodle greatly improved scalability, reliability, and institutional autonomy. The migration preserved system availability while cutting latency and load times by almost

half. Crucially, each institution acquired the autonomy to update and modify its LMS on its own, guaranteeing pedagogical flexibility and sustainability.

The results confirm that Nigerian universities, and in extension, African universities can deploy robust and scalable e-learning solution without incurring unaffordable costs by strategically utilizing reputable datacentres or national research and education networks to host their applications, while utilising open-source technologies like Moodle. In order to further incorporate performance analytics, enhance engagement, inclusiveness and equity, the study recommends future studies on optimization of Moodle LMS employing survey on UI (User Interface) and UX (User Experience).

## Author Contributions

**Chinedu Otuya:** Conceptualization, Data curation, Formal Analysis, Software, Writing – original draft, Visualisation

**Afolayan Ayodele Obiniyi:** Supervision, Writing – review & editing

**Joseph Sunday Igwe:** Supervision, Writing – review & editing

**Danladi Moses Adayilo:** Formal Analysis, Software, Validation

## Data Availability Statement

Subject to institutional approval, quantitative data, including Moodle LMS log records, are available upon request.

## Conflict of Interest

The authors declared no conflict of interest.

## References

[1] A. M. Mustapha *et al.*, 'Students 'Motivation and Effective Use of Self-regulated Learning on Learning Management System Moodle Environment in Higher Learning Institution in Nigeria', *IJIET*, vol. 13, no. 1, pp. 195–202, 2023, https://doi.org/10.18178/ijiet.2023.13.1.1796

[2] Okenema E. F, 'Blended Learning And E-Pedagogy: Enhancing Business Education Through Digitalization and CCMAS Compliance in Bayelsa State, Nigeria,' *TIJERLS*, Mar. 2025, https://doi.org/10.70382/tijerls.v07i8.037

[3] NUC, "Nigerian Universities," 2026. [Online]. Available: https://www.nuc.edu.ng/. [Accessed: Oct. 20, 2026].

[4] NUC, "Guidelines for Open and Distance Learning in Nigerian Universities," 2026. [Online]. Available: https://www.nuc.edu.ng/project/open-and-distance-education/. [Accessed: Oct. 20, 2026].

[5] NUC, "Guidelines for e-Learning in Nigerian Universities," 2023. [Online]. Available: https://www.nuc.edu.ng/wp-content/uploads/2024/06/Guidelines-for-e-Learning-in-Nigerian-Universities-April-2023-Revised-FINAL.pdf. [Accessed: Sep. 4, 2025].

[6] H. M. M. Ahmed, H. A. El-Sabagh, and D. M. Elbourhamy, 'Effect of Gamified, Mobile, Cloud-Based Learning Management System (GMCLMS) on student engagement and achievement', *Int J Educ Technol High Educ*, vol. 22, no. 1, p. 49, Aug. 2025, https://doi.org/10.1186/s41239-025-00541-1

[7] D. Alumona and S. I. Akinseinde, 'Perceived Benefits, Challenges and Level of Implementation Of Learning Management System (LMS) Among Staff and Students In Delta State Tertiary Institutions', *European Journal of Education Studies*, vol. 10, no. 4, Apr. 2023, https://doi.org/10.46827/ejes.v10i4.4792

[8] A. Abid *et al.*, 'An evaluation framework and comparative analysis of the widely used learning management systems', *PLoS ONE*, vol. 19, no. 12, p. e0311111, Dec. 2024, https://doi.org/10.1371/journal.pone.0311111

[9] K. Mohiuddin *et al.*, 'Mobile Learning New Trends in Emerging Computing Paradigms: An Analytical Approach Seeking Performance Efficiency', *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, p. 6151168, Jan. 2022, https://doi.org/10.1155/2022/6151168

[10] SPESSE, "Sustainable Procurement, Environmental and Social Standards Enhancement (SPESSE) Project Implementation Manual," World Bank, 2020. [Online]. Available: https://spesse.edu.ng

[11] S. Wetzel, J. Roeder, A. Vogler, and M. Hemmje, 'Requirement Analysis for a Qualifications-Based Learning Model Platform Using Quantitative and Qualitative Methods', *Information*, vol. 16, no. 7, p. 594, Jul. 2025, https://doi.org/10.3390/info16070594

[12] N. Bahar and H. Mohamed, 'Implementing Mobile Cloud Computing in Higher Education Institutions: A Qualitative Investigation of Influential Factors', *IJIKM*, vol. 20, p. 002, 2025, https://doi.org/10.28945/5423

[13] C. Costa, H. Alvelos, and L. Teixeira, 'The Use of Moodle e-learning Platform: A Study in a Portuguese University', *Procedia Technology*, vol. 5, pp. 334–343, 2012, https://doi.org/10.1016/j.protcy.2012.09.037

[14] A. Peramunugamage *et al.*, 'Monitoring Collaborative Interactions in Online Learning: Insights from Moodle Log Records', *Journal of Learning for Development*, vol. 12, no. 2, pp. 290–312, 2025, https://doi.org/10.56059/jl4d.v12i2.1554

[15] S. Simelane-Mnisi, "Effectiveness of LMS digital tools used by the academics to foster students' engagement," *Educ. Sci.*, vol. 13, no. 10, Art. no. 980, 2023, https://doi.org/10.3390/educsci13100980

[16] C. Otuya, A. A. Obiniyi, J. S. Igwe, D. M. Adayilo, and E. Ladan, "Analysing critical determinants of learner attrition in open and distance learning: Effect on coefficient of efficiency," *in Proceedings of the 1st International Hybrid Conference of Faculty of Science, Federal University Gusau*, 2025, pp. 89–100, https://doi.org/10.57233/ihcfs.013

[17] N. S. Alotaibi, "The impact of AI and LMS integration on the future of higher education: Opportunities, challenges, and strategies for transformation," *Sustainability*, vol. 16, no. 23, Art. no. 10357, 2024, https://doi.org/10.3390/su162310357

[18] S. Bagarish and O. Alharbi, 'Challenges and Opportunities in Cloud Computing Education: Saudi Arabia Case Study', *Int. J. Emerg. Technol. Learn.*, vol. 19, no. 08, pp. 20–31, Nov. 2024, https://doi.org/10.3991/ijet.v19i08.48361

[19] CloudZA, "Moodle on AWS: Performance gains and scalability," Stellenbosch Business School Case Study, 2022. [Online]. Available: https://cloudza.io/case-studies/case-studies-details

[20] Hubken Group, "LMS hosting: On-premises vs cloud vs SaaS – which LMS deployment is right for you?" 2024. [Online]. Available: https://www.hubkengroup.com/resources/hosting-your-moodle-or-totara-learn-site-in-house

[21] M. Manzo, "Design and performance evaluation of a virtualized Moodle-based e-learning environment," *J. e-Learn. Knowl. Soc.*, vol. 17, no. 3, 2021, https://doi.org/10.20368/1971-8829/1103

[22] J. M. Mbongue, D. Tchuinkou, and C. Bobda, "Performance exploration of virtualization systems," 2021, *arXiv: 2103.07092*.

[23] Y. B. Mohammed, N. Cavus, A. Ya'u Gital, M. Bulama, and A. Hassan, "A hybrid soft computing approach for prediction of cloud-based learning management systems determinants," *Int. J. Hum.–Comput. Interact.*, vol. 41, no. 1, pp. 504–514, 2024, https://doi.org/10.1080/10447318.2023.2301264

[24] NgREN, "Nigerian Research and Education Network services," 2026. [Online]. Available: https://www.ngren.edu.ng/services. [Accessed: Oct. 20, 2026].

[25] D. Buntinas, P. Balaji, and R. Riesen, "Monitoring and analyzing network performance in distributed systems," *Future Gener. Comput. Syst.*, vol. 123, pp. 45–57, 2021, https://doi.org/10.1016/j.future.2021.04.023

[26] C. C. Otuya, A. A. Obiniyi, and J. S. Igwe, "Graph Neural Networks (GNN) and Long Short-Term Memory (LSTM) for forecasting learner attrition: A systematic review," *Sci. J. Comput. Sci.*, vol. 2, no. 2, pp. 193–202, 2026, https://doi.org/10.64539/sjcs.v2i2.2026.453

[27] S. Simelane-Mnisi and J. M. Mthimunye, "Establishing students' satisfaction with a learning management system using a modified DeLone and McLean model: A South African sample perspective," *Educ. Sci.*, vol. 15, no. 2, Art. no. 130, 2025, https://doi.org/10.3390/educsci15020130

[28] R. K. Yin, *Case Study Research and Applications: Design and Methods*, 6th ed. Thousand Oaks, CA, USA: SAGE Publications, 2018.

[29] C. Otuya, A. Obiniyi, P. Adeyemi, and A. D. Moses, "Leveraging AI for Enhanced Student Productivity: A Case Study of Cubbes Student Information System in Nigerian Universities", Zenodo, 2025. https://doi.org/10.5281/zenodo.17719943

[30] D. M. Adayilo, I. O. Oyefolahan, J. N. Ndunagu, C. Otuya, E. Malcalm, and K. Twabu, "AI-Powered Tutoring Systems for Personalised Learning Feedback in Developing Secondary Education Contexts", *J. Fut. Artif. Intell. Tech.*, vol. 2, no. 4, pp. 549–564, Dec. 2025. https://doi.org/10.62411/faith.3048-3719-290

[31] M. O. Olaniyi, "Week 2: Financial modelling and resource allocation for blended academic environments in higher education," *VICBHE Module 13 Newsletter*, no. 2, Virtual Institute for Capacity Building in Higher Education (VICBHE), Oct. 25, 2026. [Online]. Available: https://vicbhe.org