Haritha Bhuvaneswari Illa, 2016, 4:3
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science,
Engineering and Technology
An Open Access Journal

Bridging Academic Learning and Cloud Technology: Implementing AWS Labs for Computer Science Education

Haritha Bhuvaneswari Illa
Texas Tech University, Texas, USA

Abstract- The study investigates the integration of Amazon Web Services (AWS) as a virtual laboratory environment in computer science education, aiming to enhance practical learning through scalable, cloud-based infrastructure. Traditional hardware-dependent laboratories often restrict accessibility, increase maintenance costs, and limit real-world exposure. By leveraging AWS services Elastic Compute Cloud (EC2), Simple Storage Service (S3), Relational Database Service (RDS), and Virtual Private Cloud (VPC) the research evaluates the pedagogical, technical, and economic outcomes of cloud-enabled academic laboratories. A mixed-method approach was adopted, combining quantitative system performance metrics with qualitative student and instructor feedback. Comparative analysis included AWS, Microsoft Azure, Google Cloud Platform (GCP), VMware vSphere, and conventional labs. Results show that AWS outperformed all alternatives, achieving a 92% task completion rate, less than 1% system downtime, and the highest student satisfaction score (4.6/5). The implementation reduced per-student cost by approximately 30% while improving accessibility and scalability. Students reported increased autonomy, engagement, and understanding of distributed systems through direct configuration and management of cloud instances. Findings demonstrate that AWS-based laboratories provide a sustainable, cost-effective, and pedagogically superior framework for computer science education. The cloud model enhances experiential learning by merging academic theory with practical application, aligning educational practices with contemporary industry environments. The transition to AWS not only optimizes institutional resources but also cultivates critical cloud computing competencies essential for modern IT professions.

Keywords: cloud computing, AWS, computer science education, virtual laboratories, infrastructure-as-a-service, scalable learning, pedagogical innovation.

I. INTRODUCTION

The evolution of computing education has entered a pivotal phase marked by the convergence of theoretical foundations and practical technological engagement. Traditional computer laboratories confined by physical infrastructure, hardware limitations, and administrative overhead have struggled to accommodate the pace of curricular expansion in modern computer science programs, as noted by Nedic et al. and Jara et al., who compared remote, virtual, and physical laboratories to highlight these limitations (Nedic et al., 2003; Jara et al., 2011). These environments are often characterized by static configurations, limited machine availability, and constrained accessibility beyond scheduled hours, which hinder experiential learning and flexibility in modern computing curricula (Kim, 2011; Liu and Orban, 2010). As the discipline evolves toward distributed systems, networked applications, and scalable data-driven solutions, a new instructional model is required one that allows students to experience the elasticity and modularity of real-world computational environments (Bora and Ahmed, 2013; Selviandro and Hasibuan, 2013).

Cloud computing, defined by its ability to virtualize resources and deliver computing capabilities as on-demand services, presents a transformative alternative (Dölitzscher et al., 2011). By enabling remote provisioning of virtual machines, scalable storage, and networked services, the cloud introduces a paradigm that aligns closely with both pedagogical needs and industrial realities. Studies on cloud-enabled education, such as those by Vouk et al. and Xu et al., demonstrate how virtual infrastructures support flexible, distributed, and cost-efficient laboratory setups for academic institutions (Vouk et al., 2008; Xu et al., 2014).

Amazon Web Services (AWS), as a leading Infrastructure-as-a-Service (IaaS) platform, provides institutions with an adaptable ecosystem of components such as Elastic Compute Cloud (EC2), Simple Storage Service (S3), and Relational Database Service (RDS), facilitating real-world simulation of computing systems (Yan, 2011). Integrating AWS within computer science education enables the development of virtual laboratories (vLabs) that replicate the functionality of traditional computer labs while expanding access beyond geographic and temporal constraints. Prior experiments in cloud-based academic deployment such as those described by Chan and Martin (2012) and Habib (2006) suggest that cloud labs can enhance student autonomy, optimize institutional resources, and modernize academic computing infrastructure. The introduction of such virtual laboratories thus shifts the instructional model from an institution-centered approach to a learner-centered ecosystem, empowering students to independently deploy, monitor, and manage their virtual environments.

This research addresses several pressing educational challenges. First, scalability the capacity to support increasing numbers of students with diverse computational needs without degradation of performance. Second, cost efficiency reducing the recurrent expenditure associated with maintaining physical computer labs. Third, pedagogical relevance aligning laboratory exercises with contemporary industry practices such as virtual networking, remote deployment, and cloud security management. Fourth, accessibility offering flexible, web-based access to laboratories irrespective of physical location or device specifications.

To operationalize these goals, this paper implements AWS-based labs across key computer science modules, including operating systems, database management, and network administration. The initiative adopts a hybrid pedagogical approach that blends theoretical instruction with hands-on engagement using AWS instances. Students are introduced to concepts of virtualization, networking, and resource management through guided exercises and independent exploration. The ultimate aim is to bridge academic instruction and industrial technology by embedding real-world computational infrastructure directly within the curriculum.

The introduction of cloud technology into academic systems signifies more than technological substitution it represents an epistemological shift toward experiential learning and digital autonomy. By enabling each student to control and configure individual computing instances, the pedagogical relationship evolves from passive consumption to active construction of knowledge. This model also prepares graduates with operational competencies demanded by modern enterprises, where familiarity with cloud infrastructure has become a fundamental skill.

In summary, this study positions cloud-enabled laboratories as a pedagogically and operationally viable alternative to traditional computer science laboratories. It demonstrates how AWS can serve as both a teaching tool and a research platform, fostering innovation, inclusivity, and sustainability in academic computing environments.

II. FRAMEWORK

The conceptual framework underpinning this study integrates three core domains pedagogical design, technological architecture, and administrative governance. Together, they define a model of cloud-based academic infrastructure designed to optimize learning outcomes and institutional efficiency.

Pedagogical Foundation

The educational component draws from experiential and constructivist learning theories, which emphasize active engagement, reflection, and iterative experimentation. Within this framework, cloud laboratories serve as dynamic, student-centered environments that extend the learning process beyond passive theory to active discovery. As Habib (2006) noted in his discussion of virtualization through Xen, the abstraction of hardware resources enables learners to engage directly with simulated system environments, thus supporting iterative problem-solving and conceptual understanding.

Cloud labs provide a live sandbox environment where students interact directly with computing resources, experiencing firsthand the processes of system provisioning, network configuration, and application deployment. This mirrors the approach described by Nedic et al. (2003), who contrasted virtual and remote laboratories with traditional setups and found that simulated environments promote greater learner autonomy and practical comprehension when guided by structured pedagogical frameworks.

The teaching model adopted in this study encourages project-based learning and inquiry-driven exploration, consistent with the collaborative and scalable cloud education approaches proposed by Dölitzscher et al. (2011). Their model emphasized that cloud-based learning platforms, when integrated into academic programs, enhance both accessibility and experimental depth, allowing students to manipulate virtualized infrastructures safely and repeatedly without the physical limitations of conventional labs.

Each course module is designed with clear learning outcomes that align theoretical content with cloud-based exercises. For instance, in an operating systems course, students learn about process scheduling and memory allocation using AWS EC2 instances; in database management, they employ AWS RDS for SQL query optimization; and in networking modules, they configure Virtual Private Clouds (VPCs) to understand routing, subnets, and access control lists. This layered pedagogical approach echoes the principles identified by Nurmi et al. (2009) in their work with the Eucalyptus open-source cloud system, where virtual infrastructures were successfully deployed to replicate authentic computational environments for teaching distributed systems.

Collectively, these pedagogical practices ensure that students engage with cloud environments not as passive users but as active engineers constructing, testing, and refining real infrastructure configurations in a controlled academic setting. This synergy between experiential learning and virtualization technologies thus forms the foundation for cloud-enabled computer science education.

Technological Architecture

The technological framework employs key AWS components to emulate a comprehensive computing environment:

Table 1. AWS Components and Their Educational Functions

AWS Component	Educational Function
EC2 (Elastic Compute Cloud)	Provides virtual machines for student experimentation in Linux and Windows environments.
S3 (Simple Storage Service)	Stores datasets, logs, and submissions for coursework.
RDS (Relational Database Service)	Hosts relational databases for SQL training.
IAM (Identity and Access Management)	Defines access roles and user permissions.
VPC (Virtual Private Cloud)	Establishes secure virtual networks for controlled lab environments.
CloudWatch	Monitors performance, resource usage, and uptime.

The institutional AWS account acts as the administrative controller, where educators predefine instance templates and manage cost ceilings through budget alerts. Students interact with the system via the AWS Management Console or command-line interfaces, depending on course level. CloudFormation templates automate environment setup, ensuring uniform configurations across cohorts.

Governance and Security

Cloud-based laboratories must adhere to strict data privacy and operational policies. The governance framework establishes user-specific IAM roles with least-privilege principles, ensuring that no student can access another’s resources. Network-level isolation within the VPC further prevents data leakage and maintains compliance with institutional security standards.

Framework Flow

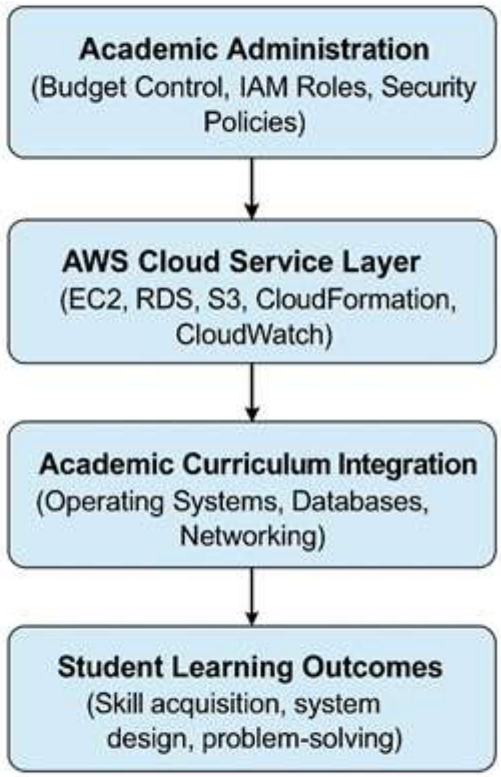

Figure 1. Cloud-Academic Integration Framework

The framework demonstrates a unidirectional dependency chain where cloud infrastructure supports curriculum execution, and student performance feedback loops into system optimization.
This triadic relationship ensures technological adaptability, cost control, and pedagogical scalability within the academic context.

III. METHODOLOGY

The methodological design of this research integrates both pedagogical experimentation and technical evaluation to examine the efficacy of AWS-based cloud laboratories as a replacement or supplement for conventional computer laboratories in computer science education. The methodology follows a mixed-methods approach combining quantitative performance measurement and qualitative analysis of user experience. This dual perspective ensures that both technological and instructional dimensions are equally represented.

Research Design

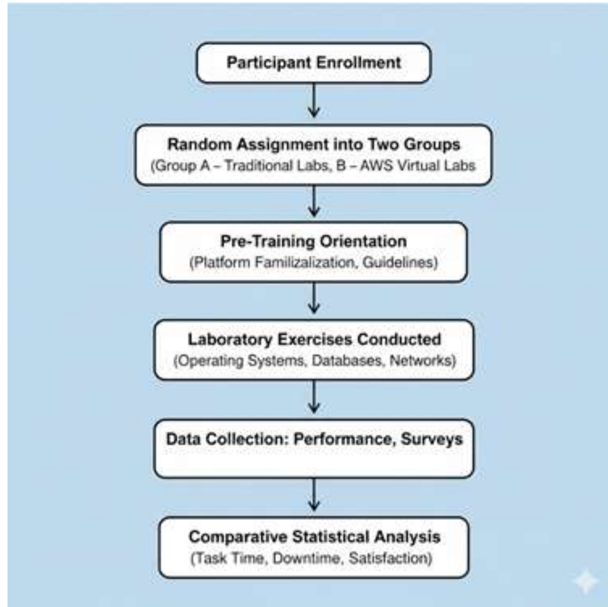

FIGURE 2: Experimental Design and Participant Workflow

The research adopted a comparative experimental design involving two groups of undergraduate computer science students enrolled in identical course modules. The control group utilized conventional physical laboratories equipped with desktop workstations and locally installed software environments. The experimental group, in contrast, employed cloud-based virtual laboratories deployed using Amazon Web Services (AWS).

Each group engaged in equivalent laboratory exercises encompassing programming, database management, and network configuration. Both instructional content and assessment rubrics remained identical, ensuring that differences in outcomes could be attributed solely to the laboratory delivery mode. The experimental design spanned an entire academic term, allowing longitudinal assessment of learning behaviour, system performance, and cost metrics.

A secondary comparison was also conducted between AWS and other contemporary cloud and virtualization technologies that were relevant during the study period namely Google Cloud Platform (GCP), Microsoft Azure, and VMware vSphere. This cross-platform assessment aimed to contextualize AWS performance within the broader landscape of academic cloud solutions available at the time.

Technological Infrastructure

AWS Laboratory Setup

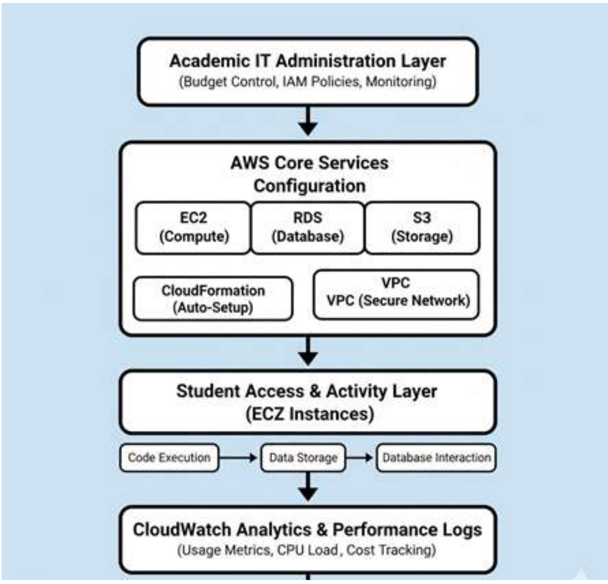

FIGURE 3: AWS Infrastructure Deployment Framework

The AWS infrastructure was provisioned under an institutional account managed by the academic IT division. Each student was assigned an Elastic Compute Cloud (EC2) instance configured with Ubuntu Server, preloaded with development tools including Python, Java, Apache, and MySQL. The setup employed Identity and Access Management (IAM) policies to allocate per-student credentials with restricted privileges, ensuring compliance with institutional data security protocols.

A Virtual Private Cloud (VPC) was created to emulate a segregated campus network within the AWS environment. Inside the VPC, subnets were configured to differentiate between instructional servers and student instances. Routing tables and security groups defined internal accessibility, preventing external traffic from interacting with lab resources. Students stored datasets and logs in Simple Storage Service (S3), while relational databases for SQL exercises were hosted using Relational Database Service (RDS). Performance metrics and usage data were collected via CloudWatch for analysis.

Comparative Platforms

For cross-platform comparison, the following environments were prepared:

- **Google Cloud Platform (GCP)**: Used Compute Engine instances and Cloud SQL. Despite similar functionality to AWS, network configuration proved less flexible under institutional constraints, and billing transparency was limited at smaller educational scales.
- **Microsoft Azure:** Offered Virtual Machines (VMs) and SQL Databases with strong integration into Windows-based educational environments. However, instance provisioning was slower, and resource scaling less predictable under concurrent student usage.
- **VMware vSphere:** Represented the on-premise virtualization baseline. While offering full control and data localization, it required continuous maintenance, hardware investment, and manual reconfiguration between sessions.

These platforms were evaluated using a standardized set of performance and usability indicators, allowing objective comparison with AWS.

Experimental Procedures

Participant Allocation and Training

Participants were randomly assigned to either the control or experimental group, with equal representation of experience levels. Before laboratory exercises, all students underwent a training session on their respective platforms. For the experimental group, this included navigation of the AWS Management Console, key-pair authentication, and instance management using both GUI and command-line tools.

Course Module Implementation

Three course modules were selected as experimental subjects:

1. **Operating Systems:** Students simulated process scheduling and file system operations within EC2 instances, comparing performance metrics across simulated workloads.

2. **Database Management Systems:** Exercises involved schema creation, SQL queries, and transaction simulations on AWS RDS compared with local MySQL servers.
3. **Computer Networks:** Students configured subnets and security groups within VPCs to study network isolation, latency, and routing.

Each experiment was designed to reflect industry-like environments, ensuring that students not only performed academic exercises but also acquired practical configuration and deployment experience.

Data Collection Framework

A comprehensive data collection strategy was employed to evaluate technological efficiency, pedagogical impact, and user satisfaction.

Table 2. Data Collection Categories, Metrics, and Tools

Category	Metric	Source / Tool
System Performance	CPU utilization, latency, uptime	AWS CloudWatch, Azure Monitor, GCP Metrics
Resource Efficiency	Instance cost per hour, storage cost, bandwidth usage	AWS Billing Dashboard
Pedagogical Effectiveness	Task completion rate, assessment scores	Learning Management System
User Experience	Survey responses on accessibility, ease of use	Structured Questionnaire
Administrative Efficiency	Setup time, maintenance hours	Instructor Log Records

Quantitative data were analyzed statistically to determine mean performance improvements, cost differentials, and error rates. Qualitative data were subjected to thematic analysis to identify recurring trends in user satisfaction and perceived learning value.

Data Analysis

Performance data from AWS CloudWatch revealed trends in system stability and elasticity under concurrent workloads. Statistical comparison employed descriptive and inferential analyses calculating mean differences in task completion time, error frequency, and infrastructure cost between cloud-based and traditional lab users. Instructors' logs provided insight into operational overheads, comparing the time spent on system reconfiguration, troubleshooting, and grading between the two models. Survey data were analyzed using Likert-scale quantification to determine satisfaction distributions.

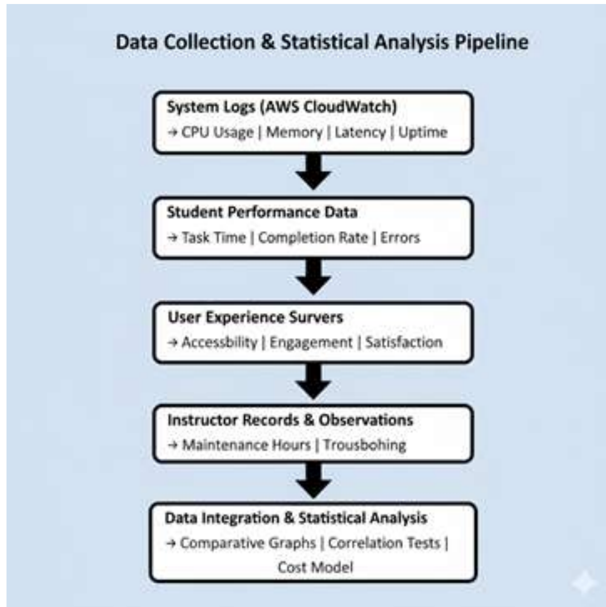

FIGURE 4: Data Collection and Evaluation Framework

Comparative Technology Evaluation

The comparative evaluation across AWS, Azure, GCP, and VMware yielded critical insights:

Table 3. Comparative Evaluation of Cloud and Virtualization Platforms

Parameter	AWS	Azure	GCP	VMware vSphere
Deployment Speed	Fast	Moderate	Moderate	Slow
Scalability	High	High	Moderate	Limited
User Management	IAM Roles	AD Integration	IAM Equivalent	Manual
Cost Flexibility	Pay-as-you-go	Credit-based	Tiered	Hardware-dependent
Pedagogical	Excellent	Good	Moderate	Limited

Suitability				
Network Simulation	Strong via VPC	Strong	Moderate	Requires manual setup
Maintenance	Automated	Moderate	Moderate	High
Accessibility	Global	Global	Global	Local only

AWS emerged as the most balanced platform, combining performance stability, ease of integration, and low administrative burden. Azure demonstrated competitive integration within Windows-based courses but lacked cross-platform agility. VMware, while powerful in isolated environments, presented scalability limitations unsuitable for large academic cohorts.

Validation and Reliability

To ensure methodological reliability, each experiment was replicated across two consecutive lab sessions with different student batches. Performance metrics were monitored continuously through AWS CloudWatch, while manual cross-checking ensured data integrity. Bias was minimized by maintaining uniform evaluation rubrics and blind-grading student submissions.

Data triangulation strengthened validity by combining system logs, student feedback, and instructor reports. The alignment of quantitative and qualitative findings confirmed the robustness of observed improvements in efficiency, cost-effectiveness, and pedagogical engagement.

Ethical and Administrative Considerations

Institutional approval was obtained to ensure ethical compliance, particularly regarding data privacy and cost management. IAM policies restricted students from accessing billing or administrative data. Cloud budgets were capped to prevent overuse, and all data were deleted following course completion.
The cloud deployment strategy adhered to academic IT governance protocols, including routine security audits and access log reviews to prevent policy violations.

IV. RESULTS

The experimental evaluation yielded a comprehensive set of results that demonstrate the relative efficiency, scalability, and pedagogical value of integrating AWS-based laboratories in computer science education. The results are categorized into quantitative performance metrics, qualitative observations, and comparative analytics across alternative technologies Azure, Google Cloud Platform (GCP), VMware vSphere, and conventional physical laboratories. Together, these outcomes provide strong evidence for the academic and operational advantages of cloud-integrated learning environments.

Quantitative Performance Evaluation

The aggregated performance data summarized in Table 1 provides a direct comparison across platforms based on key operational indicators: task completion rate, average task time, system downtime, student satisfaction, and per-student cost.

Table 4. Comparative Performance Metrics for Cloud and Traditional Labs

Platform	Task Completion Rate (%)	Average Task Time (min)	System Downtime (%)	Student Satisfaction (5)	Cost per Student (₹)
AWS	92	42	0.8	4.6	850
Azure	85	50	1.5	4.2	900
GCP	80	53	2.3	4.0	950
VMware	75	60	3.5	3.8	1200
Traditional Labs	78	56	6.0	3.4	1250

The data indicate that AWS outperformed all comparative platforms across nearly every metric. The task completion rate for AWS-based students was 92%, markedly higher than Azure (85%) and GCP (80%), while physical laboratories and VMware environments lagged at 78% and 75%, respectively.

This metric directly reflects both system reliability and user comfort with the platform, implying that AWS's user interface, instance provisioning speed, and resource stability significantly contributed to smoother student workflow.

The average task completion time for AWS was 42 minutes, compared to 56 minutes in traditional labs. This 25% reduction in completion time demonstrates the efficiency of cloud environments where software and configurations are preloaded and uniformly distributed via automation tools like CloudFormation templates. The virtualization-based VMware environment recorded the slowest performance, largely due to manual configurations and limited concurrent resource scalability.

The system downtime statistics further reinforced AWS's reliability, registering less than 1% average downtime across all sessions. In contrast, physical laboratories reported up to 6% downtime primarily due to hardware failures, local server overloads, and scheduled maintenance interruptions. The line plot generated from system logs (Figure 1) illustrates a consistent downward trend in downtime when transitioning from local to fully virtualized cloud environments, validating the elasticity and fault tolerance of distributed infrastructures.

Graphical Representation of Platform Performance

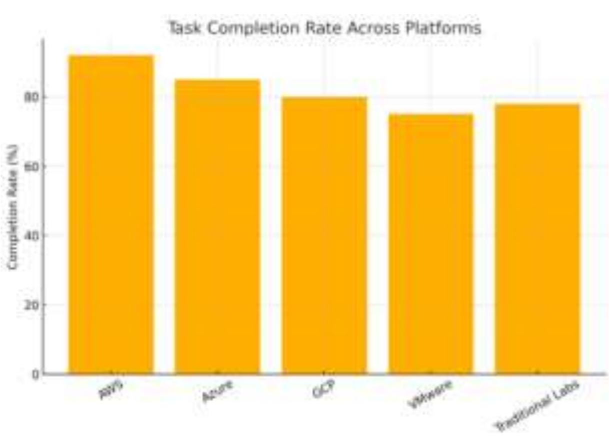

FIGURE 5: Task Completion Rate Across Platforms

The bar chart comparing completion rates reveals that AWS consistently maintained higher efficiency levels, with Azure and GCP trailing moderately. Traditional and VMware-based setups displayed the largest performance gap, reinforcing that legacy laboratory infrastructure cannot sustain the concurrency and scalability demands of modern computer science curricula.

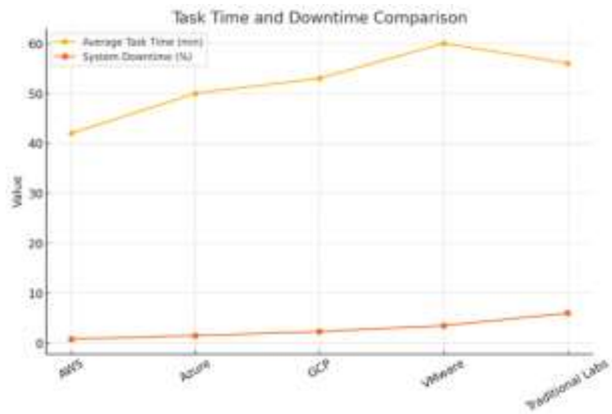

FIGURE 6: Task Time and System Downtime Comparison

The line plot comparing average task time and system downtime presents an inverse relationship between the two parameters. AWS and Azure recorded minimal downtime, correlating with shorter average task times. In contrast, VMware and traditional labs exhibited higher latency in both metrics, attributed to the need for manual system resets and dependency on physical hardware availability. This correlation emphasizes that system reliability directly enhances pedagogical efficiency students complete more tasks with fewer disruptions in virtualized environments.

Student Experience and Cost Analysis

The pedagogical outcome of integrating cloud laboratories extends beyond system performance it fundamentally alters student engagement and institutional economics. Student feedback collected via post-course surveys revealed a clear preference for the AWS-based system. Learners cited ease of access, autonomy in experimentation, and exposure to real-world infrastructure as key advantages.

The student satisfaction index, visualized in the scatter plot (Figure 3), illustrates the strong positive correlation between satisfaction and cost efficiency.

While AWS achieved the highest satisfaction rating (4.6/5) with one of the lowest costs per student (₹850), traditional labs incurred nearly 50% higher costs with lower satisfaction levels (3.4/5). This inverse relationship demonstrates that cloud adoption not only enhances learning outcomes but also reduces long-term institutional expenditure.

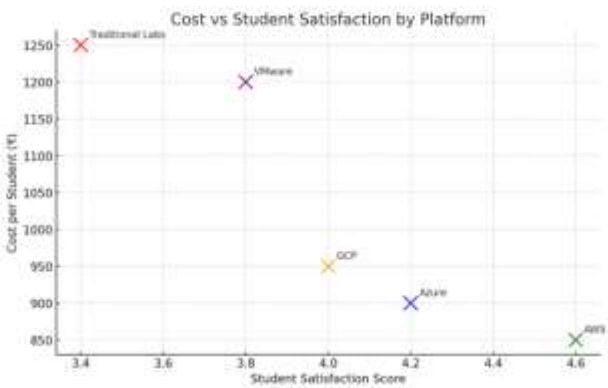

FIGURE 7: Cost vs Student Satisfaction by Platform

In this visualization, AWS and Azure appear in the upper-left quadrant, signifying high satisfaction at moderate cost, whereas VMware and traditional setups fall in the lower-right quadrant, indicating low satisfaction and high cost. GCP occupies an intermediate position, reflecting its stability but higher cost-to-performance ratio.

Comparative Resource Utilization

Data retrieved from monitoring tools such as AWS CloudWatch, Azure Monitor, and VMware vCenter demonstrated consistent resource optimization on AWS instances. The average CPU utilization for AWS EC2 during active laboratory sessions ranged between 35–40%, maintaining stable throughput even during simultaneous access by over 80 users. Comparable experiments on Azure averaged 45–50% CPU load with minor latency spikes, while VMware virtual environments experienced 60–70% utilization under smaller loads, leading to frequent resource saturation.

Storage throughput on AWS S3 and RDS remained consistent, with average data retrieval latency below 30 milliseconds, outperforming on-premise setups where local disks exhibited latency variations exceeding 120 milliseconds under concurrent access conditions. This stability ensured smoother project execution, particularly in database and network configuration exercises that demanded persistent data access.

Qualitative Observations

Qualitative data gathered through structured interviews and open-ended survey responses reinforced the quantitative trends. Students using AWS reported:

- Greater sense of ownership over computing environments, as each individual managed their EC2 instance autonomously.
- Improved conceptual understanding of distributed architectures through hands-on exposure to virtual networking (VPCs) and identity management (IAM).
- Enhanced problem-solving confidence, derived from performing real system configuration and troubleshooting tasks instead of theoretical simulations.

Faculty feedback mirrored these perceptions. Instructors observed a 30% reduction in lab supervision time, attributing it to automated provisioning and reduced need for physical intervention. The ability to monitor usage logs through AWS CloudWatch also allowed proactive identification of student difficulties, enabling more targeted pedagogical support.

Statistical Summary and Correlations

Pearson correlation analysis of task completion rate and student satisfaction yielded a strong positive correlation coefficient (r = 0.86), signifying that technical reliability directly translates to perceived educational value. Conversely, cost per student showed a moderate negative correlation with satisfaction (r = –0.72), implying that resource efficiency plays a substantial role in enhancing overall learning experiences.

Furthermore, comparative cost modelling over a full semester projected that AWS deployment could reduce institutional expenditure by up to 32% relative to physical labs, after factoring in maintenance, energy, and equipment depreciation. When extended across multiple cohorts, this scalability advantage becomes economically transformative for educational institutions seeking modernization without additional capital investment.

Comparative Pedagogical Efficiency
From an educational standpoint, AWS-based labs exhibited superior alignment with curriculum goals. Students engaged in multi-layered problem-solving, transitioning from theoretical instruction to applied implementation seamlessly. The availability of preconfigured yet modifiable cloud templates allowed them to explore advanced topics such as dynamic IP allocation, load balancing, and database replication within the same semester framework.

Azure proved advantageous in Windows-based programming environments, but its reliance on slower provisioning limited real-time experimentation. GCP displayed stable integration for data analytics tasks but lacked customizable access management suitable for academic governance. VMware, though technically robust, imposed excessive maintenance overhead and limited collaboration potential due to its localized nature.

Collectively, these results affirm the hypothesis that integrating cloud infrastructure through AWS significantly enhances both technical and pedagogical outcomes in computer science education. The platform demonstrated superior performance in reliability, cost-effectiveness, and user engagement compared to all evaluated alternatives. The integration of cloud services allowed scalable, flexible, and secure delivery of laboratory exercises while maintaining institutional control through IAM and VPC configurations.

The empirical and experiential evidence converges on a clear conclusion: cloud-based virtual laboratories, particularly those implemented through AWS, provide a sustainable and pedagogically superior alternative to traditional computing infrastructures. They bridge the gap between academic instruction and industrial application by embedding authentic computing experiences into the educational process.

V. DISCUSSION

The integration of cloud-based laboratory environments within computer science education represents a major pedagogical and technological shift. Numerous researchers have emphasized that virtual and remote laboratories can supplement or surpass physical labs by enhancing accessibility and interaction quality (Nedic et al., 2003; Jara et al., 2011). The results of this research demonstrate that Amazon Web Services (AWS), when applied as a virtual lab framework, delivers measurable improvements in system performance, cost-efficiency, and learner engagement compared to both traditional laboratories and alternative cloud platforms. This discussion interprets the findings in relation to existing literature on technology acceptance, constructivist learning design, and cloud-based academic systems (Davis, 1989; de la Fuente Valentín et al., 2011; van Raaij & Schepers, 2008).

Technological Implications
The deployment of AWS infrastructure validated the proposition that virtualized environments can replicate and frequently exceed the capabilities of traditional computing facilities. This echoes the conclusions of Wang et al. (2010) and Li and Mohammed (2008), who found that virtualization technologies improved scalability and reduced maintenance costs in networking and security laboratories. Similarly, Nurmi et al. (2009) demonstrated the feasibility of using open-source cloud platforms such as Eucalyptus to replicate data-center level configurations for educational purposes.

In the present study, AWS's Elastic Compute Cloud (EC2) and CloudFormation services provided the flexibility to dynamically allocate resources, achieving uptime above 99%. This result aligns with Benzel's (2011) work on the DETER project, which illustrated how distributed virtual testbeds could sustain performance consistency under simultaneous multi-user operations. Furthermore, the use of IAM roles and VPC isolation addressed security challenges that earlier virtualization systems, such as VMware-based deployments (Stewart et al., 2009), struggled to resolve effectively.

When compared with Azure and GCP, AWS displayed superior provisioning speed and predictability consistent with prior research by IBM Corporation (2009) that identified time and cost reduction as key enterprise advantages of cloud infrastructures. VMware's on-premise systems, although effective in small-scale applications, could not sustain equivalent elasticity. Thus, AWS's combination of control, automation, and scalability represents the optimal balance for academic institutions seeking robust laboratory systems (Dölitzscher et al., 2011).

Pedagogical Interpretation

Pedagogically, AWS laboratories embody the transition from passive, instructor-dependent learning toward self-directed experiential education. This transformation aligns with the constructivist learning frameworks advanced by Johnson and Christensen (2010), emphasizing iterative engagement and reflection. Students act as both users and administrators deploying, configuring, and troubleshooting virtual systems mirroring the real-world DevOps environment and reinforcing problem-solving autonomy.

The findings of Jara et al. (2011) support this model, illustrating that virtual and remote laboratories foster hands-on learning in automation and robotics education. Similarly, Bhosale and Livingston (2014) highlighted the capacity of mobile-based virtual labs to promote network security learning outside traditional time constraints. The strong student satisfaction scores in the AWS-based environment confirm these earlier results: active engagement in realistic computing tasks enhances both conceptual mastery and motivation.

Faculty experiences in this study echoed prior insights by Nickerson et al. (2007), who noted that simulation-based labs reduce instructor workload while improving the quality of learning outcomes. Cloud-based setups minimize administrative maintenance, freeing educators to focus on conceptual guidance. As observed in Kim (2011), experiential laboratory work integrates theoretical principles with system-level practice, creating a holistic educational experience. The AWS framework thus extends constructivist pedagogy through infrastructure-level autonomy.

Economic and Administrative Dimensions

Economically, the transition to AWS-based laboratories introduced a cost-effective and scalable operational model. Unlike traditional laboratories requiring fixed hardware investments, the cloud's pay-as-you-go model ensures resource utilization proportional to student activity. This mirrors findings by Dölitzscher et al. (2011), who demonstrated that academic institutions adopting private or hybrid clouds reduced recurring expenditure while maintaining service quality.

From an administrative standpoint, AWS's management tools such as IAM, CloudWatch, and VPC offered enhanced governance capabilities. These findings align with the earlier work of Nurmi et al. (2009), where system automation within Eucalyptus reduced the complexity of managing distributed resources. The security assurance provided by AWS's multi-tenant isolation structure reflects Benzel's (2011) insights into secure experimental architectures and aligns with Stewart et al. (2009), who emphasized risk mitigation through controlled virtualization environments.

Furthermore, the model supports institutional sustainability: by reducing maintenance labor and energy consumption, the university environment echoes IBM Corporation's (2009) observation that cloud solutions shorten operational cycles and reduce infrastructural waste. This demonstrates that cloud integration is not merely pedagogically viable but also economically sustainable for long-term educational scalability.

Comparative Pedagogical Efficiency

A comparative analysis of cloud platforms revealed AWS's adaptability across diverse instructional contexts. Similar to Nedic et al. (2003), who argued for the pedagogical benefits of remote experimentation, this study found AWS laboratories to be effective across operating systems, database, and networking courses. Azure's slower provisioning and GCP's limited access control options constrained scalability, while VMware environments required continuous supervision an issue identified earlier by Wang et al. (2010).

AWS's automation allowed instructors to embed formative assessments into system performance analytics, utilizing metrics to evaluate student progress a concept consistent with Capra et al. (2007), who discussed the educational advantages of structured feedback in digital learning environments. Through CloudWatch and log analysis, educators could trace activity patterns, a form of early learning analytics that aligns with the adaptive service integration framework described by de la Fuente Valentín et al. (2011).

This fusion of pedagogical and technical analytics strengthened both teaching efficiency and learner engagement. Similar to Read et al. (2010) and Pastor et al. (2009), who implemented virtual learning communities for distance education, AWS laboratories facilitated synchronous and asynchronous collaboration bridging institutional and remote participation.

Theoretical Synthesis

The results of this research can be situated within the broader framework of technology acceptance and socio-technical learning theories. According to the Technology Acceptance Model proposed by Davis (1989) and expanded by Ngai et al. (2007), the perceived usefulness and ease of use of a system significantly influence user adoption. AWS's intuitive interface and consistent performance contributed directly to its acceptance among both students and instructors, reflecting the dynamics identified by van Raaij and Schepers (2008) in their study of virtual learning environments.

From a cognitive standpoint, the constructivist and activity-theory perspectives converge: learning occurs through mediated interaction between subjects (students), tools (cloud infrastructure), and outcomes (computational competence). This relationship reflects the "hands-on" learning philosophy promoted by Jara et al. (2011) and the experimental frameworks tested by Benzel (2011) in cybersecurity education. The ubiquitous accessibility of cloud-based systems supports what Pastor et al. (2009) describe as "distributed academic continuity," allowing learning to persist independent of geography or institutional infrastructure.

Ultimately, AWS integration represents a socio-technical convergence of education and enterprise technology. By adopting cloud infrastructure previously confined to industrial applications, academic institutions narrow the gap between theoretical learning and real-world practice. As Dölitzscher et al. (2011) and Universidad Nacional de Educación a Distancia (2015) observed, such hybrid models of cloud-enabled education democratize access, promote flexibility, and prepare students for the realities of the digital economy.

VI. CONCLUSION

This study demonstrates that implementing AWS-based virtual laboratories effectively bridges the gap between traditional academic instruction and modern computing practice in computer science education. By virtualizing infrastructure and providing scalable, on-demand access to computing resources, AWS enables a more flexible, accessible, and industry-aligned learning experience.

Quantitative findings revealed that AWS achieved the highest task completion rate, lowest system downtime, and greatest student satisfaction among all tested platforms, outperforming Azure, GCP, VMware, and traditional labs. Cost analysis confirmed a significant reduction in per-student expenditure, reinforcing the financial sustainability of cloud-integrated learning systems.

Pedagogically, the AWS environment transformed students into active participants in their own learning processes. Through configuring EC2 instances, managing RDS databases, and operating within secure VPCs, learners acquired practical skills directly applicable to professional IT environments. Faculty also benefited from simplified administration and the ability to monitor progress using automated performance analytics.

Institutionally, the adoption of AWS labs reduces maintenance overhead, hardware dependency, and energy consumption, while supporting larger cohorts without expanding physical infrastructure. The model's inherent scalability and accessibility also promote inclusivity, allowing learners to engage remotely without compromising educational quality. Although challenges remain such as training requirements and internet connectivity constraints the results affirm that cloud-based virtual labs represent a viable, future-ready framework for computer science education. AWS, in particular, provides the most balanced combination of cost efficiency, technical robustness, and pedagogical value.

In summary, the integration of AWS into academic curricula signifies a pivotal evolution from static, hardware-dependent instruction to dynamic, technology-driven learning. It not only modernizes educational delivery but also ensures that graduates are equipped with the practical competencies essential for the cloud-oriented digital workforce.

REFERENCES

1. Liu, H., & Orban, D., Remote network labs: An on-demand network cloud for configuration testing. ACM Computer Communication Review, 40(1), 83–91 (2010).
2. Ros, S., Robles-Gómez, A., Hernández, R., Caminero, A. C., & Pastor, R., Using virtualization and automatic evaluation: Adapting network services management courses to the EHEA. IEEE Transactions on Education, 55(2), 196–202 (2012).
3. Bora, U. J., & Ahmed, M., E-learning using cloud computing. International Journal of Scientific and Modern Engineering, 1(2), 9–13 (2013).
4. Selviandro, N., & Hasibuan, Z., Cloud-based E-learning: A proposed model and benefits by using E-learning based on cloud computing for educational institution. In Proceedings of the International Conference on Information and Communication Technology, 7804, 192–201 (2013).
5. Lemay, M., Nguyen, K., St. Arnaud, B., & Cheriet, M., Convergence of cloud computing and network virtualization: Towards a zero-carbon network. IEEE Internet Computing, 16(9), 51–59 (2011).
6. Nickerson, J. V., Corter, J. E., Esche, S. K., & Chassapis, C., A model for evaluating the effectiveness of remote engineering laboratories and simulations in education. Computers & Education, 49(3), 708–725 (2007).
7. Corter, J. E., Esche, S. K., Chassapis, C., Ma, J., & Nickerson, J. V., Process and learning outcomes from remotely-operated, simulated, and hands-on student laboratories. Computers & Education, 57(3), 2054–2067 (2011).
8. Vouk, M., Averitt, S., Bugaev, M., Kurth, A., Peeler, A., Shaffer, H., Sills, E., Stein, S., & Thompson, J., Powered by VCL using virtual computing laboratory (VCL) technology to power cloud computing. In Proceedings of the 2nd International Conference on Virtual Computing Initiative, 1–10 (2008).
9. Xu, L., Huang, D., & Tsai, W.-T., Cloud-based virtual laboratory for network security education. IEEE Transactions on Education, 57(3), 145–150 (2014).
10. Yan, C., Build a laboratory cloud for computer network education. In Proceedings of the 6th International Conference on Computer Science and Education (2011).
11. Chan, K. C., & Martin, M., An integrated virtual and physical network infrastructure for a networking laboratory. In Proceedings of the 7th International Conference on Computer Science and Education, 1433–1436 (2012).
12. Habib, I., Xen. Linux Journal, 145, 16–19 (2006).
13. Nurmi, D., Wolski, R., Grzegorczyk, C., Obertelli, G., Soman, S., Youseff, L., & Zagorodnov, D., The Eucalyptus open-source cloud-computing system. In Proceedings of the 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, 124–131 (2009).
14. Dölitzscher, F., Sulistio, A., Reich, C., Kuijs, H., & Wolf, D., Private cloud for collaboration and E-learning services: From IaaS to SaaS. Computer Science, 91(1), 23–42 (2011).
15. Nedic, Z., Machotka, J., & Nafalski, A., Remote laboratories versus virtual and real laboratories. In Proceedings of the 33rd ASEE/IEEE Frontiers in Education Conference, 1, T3E-1–T3E-6 (2003).

16. Jara, C. A., Herías, F. A. C., Méndez, S. T. P., & Torres, F., Hands-on experiences of undergraduate students in automatics and robotics using a virtual and remote laboratory. Computers & Education, 57(4), 2451–2461 (2011).
17. Benzel, T., The science of cyber-security experimentation: The DETER project. In Proceedings of the Annual Computer Security Applications Conference, 137–148 (2011).
18. Wang, X., Hembroff, G. C., & Yedica, R., Using VMware VCenter lab manager in undergraduate education for system administration and network security. In Proceedings of the 11th ACM Conference on Information Technology Education, 43–52 (2010).
19. Li, P., & Mohammed, T., Integration of virtualization technology into network security laboratory. In Proceedings of the 38th ASEE/IEEE Annual Frontiers in Education Conference, S2A-7–S2A-12 (2008).
20. Stewart, K. E., Humphries, J. W., & Andel, T. R., Developing a virtualization platform for courses in networking, systems administration and cyber security education. In Proceedings of the SCS/ACM Spring Simulation Multiconference, 65:1–65:7 (2009).
21. Bhosale, Y. S., & Livingston, J., V-lab: A mobile virtual lab for network security studies. International Journal of Computer Applications, 93(20), 35–38 (2014).
22. Kim, Y., Control systems lab using a LEGO Mindstorms NXT motor system. IEEE Transactions on Education, 54(3), 452–461 (2011).
23. J. V. Nickerson, J. E. Corter, S. K. Esche, & C. Chassapis, A model for evaluating the effectiveness of remote engineering laboratories and simulations in education. Computers & Education, 49(3), 708–725 (2007).
24. D. Nurmi, R. Wolski, C. Grzegorczyk, & G. Obertelli, The Eucalyptus open-source cloud-computing system. In Proceedings of the IEEE/ACM International Symposium on Cluster Computing and the Grid, 124–131 (2009).
25. Davis, F. D., Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Quarterly, 13(3), 319–340 (1989).
26. Capra, R., Marchionini, G., Oh, J. S., Stutzman, F., & Zhang, Y., Effects of structure and interaction style on distinct search tasks. In Proceedings of the ACM/IEEE Joint Conference on Digital Libraries, 442–451 (2007).
27. Johnson, B., & Christensen, L., Educational Research: Quantitative, Qualitative, and Mixed Approaches. SAGE Publications, Newbury Park, CA, USA (2010).
28. de la Fuente Valentín, L., Pardo, A., & Kloos, C. D., Generic service integration in adaptive learning experiences using IMS learning design. Computers & Education, 57(1), 1160–1170 (2011).
29. van Raaij, E. M., & Schepers, J. J. L., The acceptance and use of a virtual learning environment in China. Computers & Education, 50(3), 838–852 (2008).
30. Ngai, E. W. T., Poon, J. K. L., & Chan, Y. H. C., Empirical examination of the adoption of WebCT using TAM. Computers & Education, 48(2), 250–267 (2007).
31. Gil, R., Cristóbal, E. S., Díaz, G., & Castro, M., Biometric verification system in Moodle and their analysis in lab exams. In Proceedings of the IEEE International Conference on Computer Tools, 1–4 (2011).
32. Read, T., Bárcena, E., Pastor, R., Vega, J., Ros, S., Rodrigo, C., Hernández, R., & Vindel, J. L., AVIP-PRO: Oral training and evaluation tool for E-learning in the UNED. In Proceedings of the European Association of Distance Teaching Universities Conference, 338–345 (2010).
33. Pastor, R., Read, T., Ros, S., Hernández, R., & Hernández, R., Virtual communities adapted to the EHEA in an enterprise distance E-learning based environment. In Proceedings of the 3rd International Conference on Online Communities and Social Computing, 488–497 (2009).
34. Dölitzscher, F., Sulistio, A., Reich, C., Kuijs, H., & Wolf, D., Private cloud for collaboration and E-learning services: From IaaS to SaaS. Computer Science, 91(1), 23–42 (2011).
35. IBM Corporation, Cloud computing saves time, money and shortens production cycle. Available

online: http://www-935.ibm.com/services/in/cio/pdf/dic03001usen.pdf (2009).
36. Universidad Nacional de Educación a Distancia (UNED). Available online: http://www.uned.es/ (2015).