



RESEARCH ARTICLE

Comparative Analysis of Windows and Linux Operating Systems for Secondary School Laboratories: A Limited Systematic Literature Review

Zaehol Fatah ¹ | Aisyah Putri Sabrina ^{2*}

^{1,2*} Universitas Ibrahimy, Situbondo Regency,
East Java Province, Indonesia.

Correspondence

^{2*} Universitas Ibrahimy, Situbondo Regency,
East Java Province, Indonesia.

Email: aisyahputrisabrina31@gmail.com.

Funding information

Universitas Ibrahimy.

Abstract

Operating systems determine the day-to-day sustainability of school computer labs — particularly when institutions face licensing budget constraints, varying device specifications, and basic security requirements. This article is structured as a limited systematic literature review comparing Windows and Linux in an educational context, without claiming to be a primary benchmark experiment. The methodology follows PRISMA 2020 and Kitchenham systematic literature review guidelines, covering research question formulation, literature search, inclusion-exclusion criteria-based selection, quality assessment, and thematic synthesis. From an initial pool of 68 records, 10 relevant and traceable sources were used as the basis for synthesis across five core dimensions: licensing cost, hardware requirements, application compatibility, security, and user readiness. The findings indicate that while Linux presents strict user privilege execution layers and zero direct licensing overhead, Windows demonstrates superior native compatibility with proprietary national examination client browsers and a shorter learning curve due to established user familiarity. This article contributes a multi-criteria decision matrix for educational labs that balances operational constraints against technical efficiency.

Keywords

Linux; Windows; OS Migration; Total Cost of Ownership; Computer Laboratory.

1 | INTRODUCTION

An operating system is the foundational layer that determines how hardware, learning applications, networks, and user data function within a computer lab. In secondary education, operating system selection is not purely a technical preference — it is shaped by licensing costs, device age, application availability, security requirements, and the readiness of teachers and students to work with unfamiliar interfaces (Fitriani & Utomo, 2023). Schools rarely have the luxury of choosing an OS based on technical merit alone; budget cycles, procurement policies, and the sheer inertia of existing infrastructure all push back against change.

Windows remains the most familiar operating system to most school users, owing to its long-standing presence on personal computers, labs, and administrative devices. This familiarity carries real practical weight: teachers and students navigate basic tasks more easily, school technicians benefit from extensive vendor documentation, and the majority of commercial learning applications are designed with Windows as the primary target. That said, modern Windows deployments carry hardware minimums, licensing obligations, and security update cycles that can strain a school's annual budget — particularly for institutions managing dozens of aging machines (Microsoft, 2025; Microsoft Security, 2025). When a single license renewal covers an entire lab, the cost adds up faster than most school administrators anticipate.

Linux takes a different path through its open-source model. Many distributions can be deployed legally without operating system licensing fees, offer educational software repositories, and provide lightweight desktop environments suited to older hardware (Canonical, 2024; Hidayat & Setiawan, 2022). On paper, the financial case for Linux in underfunded school labs is straightforward. In practice, it is more complicated. Application compatibility gaps, user training demands, and the availability of technicians who actually understand Linux administration all require honest assessment before any school commits to migration (Suryadi & Lestari, 2025; Shidqin *et al.*, 2025). A distribution that runs smoothly on a developer's workstation does not automatically translate into a stable, manageable environment for thirty students and one overworked lab technician.

The main criticisms directed at earlier manuscripts on this topic were the absence of original research data, the lack of an explicit SLR procedure, overly absolute claims, and references that were difficult to verify. This article therefore positions itself as a limited systematic literature review. Performance figures, memory requirements, and security findings are not presented as the authors' experimental results — they are treated as a synthesis of existing literature and technical documentation that schools should re-examine before making implementation decisions (Kitchenham & Charters, 2007; Page *et al.*, 2021). Transparency about what this study is, and what it is not, matters more than projecting false confidence.

A recurring limitation in existing operating system comparisons is their narrow focus on raw hardware benchmarks, which leaves out the socio-technical friction, localized technician capacity, and curriculum alignment constraints that actually determine whether a laboratory migration succeeds in resource-limited schools. Benchmark scores measured in a controlled lab environment tell only part of the story. The other part — teacher adaptation, student frustration during transition periods, and the hidden cost of re-training support staff — rarely appears in the literature. Unlike prior studies that stop at technical benchmarks, this article builds a multi-criteria decision framework that accounts for cost, hardware longevity, application compatibility, security architecture, and user readiness, specifically for secondary educational laboratory settings. The objective is to perform a limited systematic literature review that evaluates and compares Windows and Linux operating systems within those real-world constraints.

2 | BACKGROUND THEORY

An operating system functions as a foundational manager for core hardware processes, memory allocations, input-output devices, file systems, and low-level security mechanisms. Within educational environments, these system functions directly shape the operational sustainability of computer laboratories — affecting application responsiveness, data integrity, and the administrative workflow of school technicians (Myers & Singh, 2023; Pratama & Kurniawan, 2023). A lab where the OS frequently stalls, mismanages memory, or exposes student data to unauthorized access is not just a technical inconvenience; it disrupts the entire learning process. To establish a sound theoretical foundation for comparison, these baseline functionalities are analyzed through two distinct conceptual dimensions: Total Cost of Ownership (TCO) governance frameworks and platform-specific architecture ecosystems.

2.1 Total Cost of Ownership (TCO) and Laboratory Governance Frameworks

From an educational technology governance perspective, the institutional decision to adopt or migrate an operating system cannot be evaluated solely through narrow technical performance benchmarks. Sound laboratory infrastructure management requires a Multi-Criteria Decision Analysis (MCDA) that aligns deployment costs with institutional capacity

(Alhothail *et al.*, 2023; Suryadi & Lestari, 2025). A school that selects an OS purely because it performs well in memory benchmarks, without accounting for training costs or technician availability, is likely to encounter problems that no benchmark ever predicted. The underlying differences between commercial and open-source licensing models create highly divergent budgetary trajectories for secondary schools (Chen & Li, 2023; Fitriani & Utomo, 2023). Windows licensing, even under educational discount schemes, accumulates into a recurring expenditure that schools must plan for every few years. Linux, by contrast, eliminates that line item — but introduces others. A sustainable governance approach must therefore compute a full Total Cost of Ownership, one that extends well beyond initial licensing procurement to include user readiness, technical support availability, application compatibility parameters, and structured post-migration training cycles (Ramadhan & Wijaya, 2024; Suryadi & Lestari, 2025). Reducing TCO to a single licensing cost figure is one of the most common — and most costly — mistakes schools make when planning an OS migration.

2.2 Platform Security Architectures and Software Ecosystems

The architectural configuration of an operating system kernel heavily shapes its baseline security posture and software deployment flexibility. Windows represents a commercial, proprietary ecosystem characterized by broad native application compatibility, rigorous vendor support channels, and automated administrative management tools (Microsoft, 2025; Patel & Desai, 2024). For educational institutions that depend on specialized administrative applications, proprietary examination software, or legacy hardware peripherals requiring vendor-exclusive device drivers, this ecosystem offers clear and measurable advantages — the software simply works, out of the box, without requiring manual configuration or workarounds (Microsoft, 2025; Patel & Desai, 2024). The Linux architecture, by contrast, is rooted in a decentralized, open-source model that grants extensive operational flexibility. Schools can select tailored desktop environments and deploy lightweight distributions specifically built to extend the usable life of older hardware (Canonical, 2024; Hidayat & Setiawan, 2022). A machine that can no longer run Windows 11 due to TPM or RAM constraints may still serve students reliably for several more years under a well-configured lightweight Linux distribution. From a security standpoint, the Linux kernel relies on a strict, hierarchical user privilege execution model and centralized package distribution repositories, which substantially reduce the risk of unauthorized binary execution and automated malware proliferation in public computer laboratories (Alhothail *et al.*, 2023; Ramadhan & Wijaya, 2024). That said, these architectural advantages only hold when the system is properly administered — a misconfigured Linux environment can be just as vulnerable as any other platform.

3 | METHOD

This study employed a limited systematic literature review with a qualitative-comparative synthesis. Reporting guidelines follow PRISMA 2020 (Page *et al.*, 2021), while the planning, selection, data extraction, and quality assessment stages align with the Kitchenham and Charters guidelines for systematic literature reviews in software engineering (Kitchenham & Charters, 2007). Three research questions guided the entire process: RQ1 — what technical and non-technical dimensions are relevant when comparing Windows and Linux for educational use? RQ2 — how does the literature explain differences in licensing costs, hardware requirements, application compatibility, security, and user readiness? RQ3 — what decision framework can schools use to ensure a proportional operating system migration?. The search was conducted across Google Scholar, Garuda, Crossref, official Microsoft documentation (Microsoft, 2025), official Ubuntu/Linux Mint documentation (Canonical, 2024; Linux Mint Project, 2025), and PRISMA and SLR methodological sources (Kitchenham & Charters, 2007; Page *et al.*, 2021). Search terms included combinations of "Windows Linux education," "operating system comparison school laboratory," "Linux lightweight distribution education," "Windows 11 hardware requirements," "Linux security permission model," and "systematic literature review software engineering." These keyword combinations were deliberately kept broad enough to capture both technical and governance-oriented literature, while remaining specific enough to exclude sources with no relevance to educational laboratory settings.

Table 1. Literature selection criteria

Type	Criteria
Inclusion	Discusses Windows, Linux, operating systems, education, computer laboratories, security, costs, hardware requirements, or SLR methodologies.
Inclusion	Sources consisting of scientific articles, official vendor/project documentation, technical reports, or traceable methodological guidelines.
Exclusion	Opinion-based blogs without citations, popular articles lacking data, inaccessible sources, or references whose journal identity cannot be verified.
Exclusion	Sources that solely discuss a single operating system without any relevance to educational needs or

laboratory management.

The literature selection process followed four sequential stages to maintain transparency, scientific rigor, and to minimize selection bias, adhering to the structural flow of the PRISMA 2020 statement (Page *et al.*, 2021). During the initial identification phase, a search across academic databases and official technical repositories yielded 68 records. A preliminary deduplication and data cleansing process then removed 14 records identified as duplicate entries or entirely outside the pedagogical and operating system scope, leaving 54 unique records for further evaluation. These 54 records were subjected to a screening phase in which titles and abstracts were evaluated against the predefined inclusion and exclusion criteria (Kitchenham & Charters, 2007). This stage narrowed the pool to 26 potential studies that warranted closer inspection. In the eligibility phase, full-text reviews were conducted on all 26 documents to assess internal quality, factual accuracy, and technical relevance regarding licensing models, security mechanisms, or hardware constraints. Ten sources ultimately *met all* quality assessment benchmarks and provided sufficient depth of information to serve as the empirical foundation for the thematic synthesis.

Table 2. Summary of the revised PRISMA flow

Stage	Quantity	Description
Identification	68	Initial records retrieved from scientific articles, official documentation, and methodological guidelines.
Duplication/Irrelevant	14	Removed due to duplication or lack of relevance to operating systems and educational contexts.
Screening	54 → 26	Title and abstract screening based on relevance to the established research questions.
Eligibility	26 → 10	Full-text evaluation and verification of source traceability.
Included	10	Final core sources utilized in the thematic synthesis.

Relying on a final synthesis pool of only 10 core sources represents a narrow empirical baseline for a systematic literature review. While this limits the depth of the thematic evaluation, these sources were selected strictly on the basis of their contextual relevance to secondary educational laboratories — breadth was deliberately sacrificed in favor of precision and traceability.

4 | RESULTS AND DISCUSSION

4.1 Results

The findings of this study suggest that a definitive verdict between Windows and Linux cannot be drawn cleanly. Both platforms carry distinct strengths and limitations depending on laboratory conditions, the type of learning application in use, technician capacity, and school budget policies. Table 3 presents the synthesis matrix across five core dimensions derived from the reviewed literature.

Table 3. Synthesis matrix of Windows and Linux comparison for education

Dimension	Windows	Linux	Educational Implications
Licensing and cost	Commercial license-based; may be available through specific educational schemes.	Generally usable without operating system licensing fees.	Linux can reduce licensing expenses, but schools must still calculate training and technical support costs.
Hardware requirements	Windows 11 mandates a minimum of 4 GB RAM and specific hardware constraints.	Modern desktop distributions also require adequate hardware; lightweight distributions are better suited for legacy computers.	Legacy computers must be audited before deciding whether to retain older Windows versions, upgrade components, or migrate to lightweight Linux.
Application compatibility	Strong for commercial software, specific examination applications, and hardware with exclusive Windows	Strong for open-source applications, programming environments, local servers, and computational learning.	Decisions must begin from the school's mandatory application checklist, rather than operating system preferences.

	drivers.		
Security	Equipped with modern security features, but represents a prominent target due to its large user base.	Package repositories and permission models reduce unauthorized execution risks, yet vulnerabilities still persist.	Both platforms require routine updates, user privilege restrictions, data backups, and structured security awareness education.
User acceptance	High due to widespread familiarity among both instructors and students.	Variable; modern distributions are more user-friendly but still require initial orientation.	Linux migration demands a pilot testing phase, technical training, and localized documentation.

A secondary frequency mapping of the 10 selected sources shows that licensing costs and hardware constraints were addressed by all 10 sources (100%), application compatibility by 8 sources (80%), security architectures by 7 sources (70%), and user acceptance and technical readiness barriers by 6 sources (60%). This distribution reflects a clear pattern: cost and hardware are the dimensions schools most frequently grapple with, while user readiness tends to receive less systematic attention despite its practical weight in real migration scenarios.

4.2 Discussion

From a licensing perspective, Linux can be installed legally without purchasing an operating system license on many distributions — a finding directly relevant for schools managing large numbers of computers under tight budgets. The total cost of ownership, however, should not be reduced solely to initial licensing procurement. A successful operating system deployment involves broader structural commitments (Chen & Li, 2023; Fitriani & Utomo, 2023). While Chen & Li (2023) reported Linux's superior memory efficiency under controlled benchmark conditions, this synthesis suggests that such technical advantages may be offset by user readiness barriers in secondary school environments — a dimension their study did not fully address. Schools must also account for staff training time, localized technical support infrastructure, compatibility with administrative applications, and the potential need for dual-booting or virtualization environments. The theoretical licensing savings of a Linux migration are frequently offset by the hidden operational costs of software virtualization or hardware dual-booting, which schools must implement to maintain compatibility with proprietary national examination browsers that do not run natively on open-source kernels. In other words, the savings on one line item quietly reappear on another.

The original manuscript stated that Linux's idle RAM consumption was 500–800 MB and Windows' was 2.0–2.5 GB. In this revised version, these figures are not treated as direct benchmark results, since no primary experiments were conducted. Performance data on memory efficiency and CPU constraints are treated as secondary macro-indicators. This aligns with Myers & Singh (2023), who demonstrated that proprietary systems significantly increase idle memory overhead compared to open-source kernels, despite providing more automated background update mechanisms. These parameters need to be verified through controlled, locally conducted experimental testing before being used as hard baselines for institutional policy (Myers & Singh, 2023; Pratama & Kurniawan, 2023). A school that makes hardware procurement decisions based on unverified figures risks either over-investing in unnecessary upgrades or under-preparing machines that will struggle under real classroom loads.

On security, this article avoids the generalized claim that Linux provides complete protection or absolute immunity against malware. Architecturally, Linux maintains a clear separation of user and administrator access rights through its hierarchical user privilege execution layer, which can substantially reduce the risk of unauthorized or accidental system modifications (Alhothail *et al.*, 2023; Ramadhan & Wijaya, 2024). Linux environments remain subject to kernel vulnerabilities, administrator misconfigurations, cross-platform malware vectors, and social engineering risks. Windows also incorporates modern security features, but practical operational risks increase sharply when schools deploy unofficial, unpatched operating system versions or grant student profiles unrestricted administrator access (Microsoft Security, 2025; National Institute of Standards and Technology, 2023). Security, on either platform, is ultimately a governance problem — not a product feature.

The core novelty of this article is not the claim that Linux is technically superior to Windows — that comparison has been widely covered in system software literature. The novelty lies in building a multi-criteria decision framework tailored specifically for secondary educational computer laboratories. This framework places technical performance variables alongside non-technical constraints: long-term operational cost, hardware longevity, application compatibility, kernel security strategies, user psychological readiness, and technical support capacity (Patel & Desai, 2024; Suryadi & Lestari, 2025; Williams & Patel, 2021).

Table 4. Suggested benchmark parameters for future research

Parameter	Measurement Method	Validity Notes
Boot time	Measuring the duration from power-on until the desktop is fully ready for use, with a minimum of 5 repetitions.	Use identical hardware configurations, consistent network conditions, and document the specific OS versions.
Idle RAM	Recording RAM utilization after the system remains idle for 5 minutes without any additional background applications running.	Segregate data between Windows, Linux GNOME, and lightweight Linux distributions.
Application responsiveness	Launching a web browser, office productivity suite, and a simple IDE simultaneously.	Measure both subjective latency and explicit application loading times.
Stability	Documenting instances of system crashes, freezes, or operational errors during simulative laboratory practical sessions.	Requires a minimum testing duration of 60–90 continuous minutes.
User acceptance	Distributing questionnaires to students and instructors following a brief orientation or training session.	Utilize Likert scale measurements complemented by brief structured interviews.

To address RQ3 and provide schools with practical governance guidance, a multi-criteria decision matrix has been synthesized from the reviewed literature. This matrix ensures that operating system migration is evaluated proportionally based on real-world constraints rather than raw technical bias.

Table 5. Decision framework for school operating system migration

School Condition	Recommendation
Mandatory applications exclusively run on Windows	Retain Windows on specific designated units; consider deploying Linux for programming practice computers or legacy hardware.
Legacy hardware with constrained RAM and web-based applications	Conduct a pilot trial of lightweight Linux distributions on a subset of units, followed by an evaluation of performance and user acceptance.
Technicians lack prior experience with Linux	Begin with administrator technical training, structured installation documentation, and small-scale pilots prior to broader institutional migration.
Licensing budget constraints represent the primary operational obstacle	Calculate a full Total Cost of Ownership (TCO): including licensing, training, ongoing support, compatibility, and transition duration.
Educational institutions aim to build open-source literacy	Bring Linux directly into the academic curriculum as an active learning resource, rather than treating it merely as a generic Windows replacement.

5 | CONCLUSIONS AND FUTURE WORK

This study addresses the three established research questions through a limited systematic literature review grounded in 10 carefully selected and traceable sources. For RQ1, the critical technical and non-technical dimensions identified are licensing costs, hardware requirements, application compatibility, security architectures, and user readiness — five dimensions that together reflect the real operational complexity schools face when evaluating an OS migration. For RQ2, the synthesized literature confirms that Linux offers strict, hierarchical user privilege execution layers and zero licensing overhead, while Windows leads in commercial application compatibility and established user familiarity. Neither platform holds an unconditional advantage; the better choice depends entirely on the school's specific conditions. For RQ3, a proportional decision matrix (Table 5) has been constructed to guide schools based on their local technician capacity and mandatory software dependencies, rather than on abstract technical rankings.

One finding that cuts across all three research questions deserves emphasis: laboratory security is ultimately determined by update governance, access rights configurations, data backups, and user security literacy — not by the platform itself. A well-administered Windows lab is more secure than a poorly managed Linux one, and vice versa. Platform choice sets the starting conditions; administrative discipline determines the outcome.

For future work, empirical action research is recommended over generic comparative studies. Specifically, longitudinal studies that track user adaptation rates, software virtualization friction, and actual technical maintenance hours in a real secondary school laboratory over a full academic year would produce far more

actionable findings than controlled benchmark experiments conducted in isolation. The gap between what the literature reports and what school technicians actually experience on the ground remains wide — and closing that gap is where the most useful research still needs to happen.

ACKNOWLEDGEMENTS

The authors thank their supervisor, fellow students, and laboratory administrators for their input during the preparation of this article. Thanks are also extended to the manuscript reviewers whose methodological critiques helped clarify the research position, limit unsupported claims, and strengthen the traceability of references

REFERENCES

- Alhothail, B., Alshimer, M., Alotaibi, S., & Nagy, N. (2023). Comparison between Windows and Linux operating system by analyzing the related security features. *International Journal of Computer Science and Cybersecurity*, 12(2), 74–83.
- Canonical. (2024). *Install Ubuntu Desktop 24.04 LTS*. Ubuntu Documentation. <https://documentation.ubuntu.com/desktop/en/24.04/tutorial/install-ubuntu-desktop/>
- Canonical Community Help Wiki. (2024). *Installation/SystemRequirements*. <https://help.ubuntu.com/community/Installation/SystemRequirements>
- Chen, Y., & Li, Z. (2023). Performance analysis of Linux and Windows operating systems for personal computing. *International Journal of Computer Science and Applications*, 35(2), 105–120.
- Fitriani, A., & Utomo, B. (2023). Analisis Total Cost of Ownership (TCO) migrasi sistem operasi open source di lingkungan pendidikan menengah. *Jurnal Edukasi dan Teknologi Informatika*, 7(2), 88–95.
- Hidayat, R., & Setiawan, A. (2022). Optimalisasi laboratorium komputer sekolah menggunakan distribusi Linux ringan pada perangkat keras usang. *Jurnal Teknologi Informasi dan Rekayasa Komputer*, 4(1), 34–42.
- Kitchenham, B., & Charters, S. (2007). *Guidelines for performing systematic literature reviews in software engineering* (EBSE Technical Report EBSE-2007-01). Keele University.
- Linux Mint Project. (2025). *Linux Mint installation guide: System requirements*. <https://linuxmint-installation-guide.readthedocs.io/en/latest/>
- Microsoft. (2025a). *Windows 11 requirements*. Microsoft Learn. <https://learn.microsoft.com/en-us/windows/whats-new/windows-11-requirements>
- Microsoft. (2025b). *Windows 11 specifications and system requirements*. <https://www.microsoft.com/en-us/windows/windows-11-specifications>
- Microsoft Security. (2025). *Cyberthreats, viruses, and malware*. Microsoft Security Intelligence. <https://www.microsoft.com/en-us/wdsi/threats>
- Myers, A., & Singh, P. (2023). Windows vs Linux: A comprehensive comparison of resource efficiency and system performance. *Journal of Operating Systems Research*, 22(3), 45–59.
- National Institute of Standards and Technology. (2023). *Guidelines for managing the security of mobile devices in the enterprise* (NIST Special Publication 800-124 Rev. 2). U.S. Department of Commerce. <https://doi.org/10.6028/NIST.SP.800-124r2>

- Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T., Loder, E. W., Mayo-Wilson, E., McDonald, S., & Moher, D. (2021). The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ*, 372, n71. <https://doi.org/10.1136/bmj.n71>
- Patel, D., & Desai, R. (2024). Exploring the user experience differences between Windows and Linux: A case study for personal use. *Journal of Information Technology*, 29(4), 231–240.
- Pratama, M. A., & Kurniawan, D. (2023). Analisis komparatif performa throughput memori dan penjadwalan CPU: Windows vs Linux. *Jurnal Riset Sistem Komputer dan Informatika*, 5(3), 112–121.
- Ramadhan, F., & Wijaya, S. (2024). Studi komparasi arsitektur keamanan kernel terhadap infeksi malware pada laboratorium komputer publik. *Jurnal Keamanan Siber dan Sistem Informasi*, 3(2), 45–53.
- Shidqin, L. H. M., Hidayat, T., & Fatah, Z. (2025). Pelatihan instalasi sistem operasi Linux Mint di VirtualBox pada siswa TKJ SMK Ibrahimy 1 Sukorejo. *JUAN: Jurnal Pengabdian Nusantara*, 2(3), 76–84.
- Suryadi, E., & Lestari, P. (2025). Analisis usabilitas dan kurva pembelajaran antarmuka grafis sistem operasi alternatif bagi pengguna awam di sekolah. *Jurnal Penyelidikan Teknologi Pendidikan*, 8(1), 15–24.
- Williams, S., & Patel, D. (2021). Thread-level parallelism in Windows and Linux. *Concurrency and Computation: Practice and Experience*, 32(12), e6021. <https://doi.org/10.1002/cpe.6021>.

How to cite this article: Fatah, Z., & Sabrina, A. P. (2026). Comparative Analysis of Windows and Linux Operating Systems for Secondary School Laboratories: A Limited Systematic Literature Review. *Journal Dekstop Application (JDA)*, 5(1), 52–59. <https://doi.org/10.59431/jda.v5i1.874>.