

Article

Research on AI Hybrid Teaching Framework for Principles of Operating Systems in Computing Majors

Bingke Wei ^{1,*}, Bingjie Wei ¹ and Jinpeng Chen ¹

¹ Department of Computing, Guangdong University of Science and Technology, Dongguan, China

* Correspondence: Bingke Wei, Department of Computing, Guangdong University of Science and Technology, Dongguan, China

Abstract: Aiming at the prevalent teaching pain points of the Principles of Operating Systems course in contemporary computing majors—such as highly abstract and difficult theoretical concepts, a persistent disconnection between theoretical knowledge and practical application, a lack of personalized teaching methodologies, and the superficial application of modern educational technologies—this study systematically integrates artificial intelligence with hybrid teaching models. This integration is fundamentally based on the urgent talent cultivation demands of the modern digital economy and the strategic need for the independent controllability of domestic operating systems. Using a comprehensive mixed-methods approach that includes extensive literature research, detailed questionnaire surveys, and rigorous experimental comparisons, a robust four-layer integrated AI hybrid teaching framework is meticulously constructed. This innovative framework consists of an objective layer, a core layer, a support layer, and an evaluation layer, each featuring clear design principles, specialized core modules, and streamlined implementation processes. Verified through extensive teaching practice and empirical evaluation in higher education institutions, this framework successfully achieves the deep, meaningful integration of advanced AI technologies and traditional course teaching. The empirical results demonstrate that this approach can effectively improve students' overall academic performance, stimulate intrinsic learning interest, enhance engineering practical abilities, and foster comprehensive systematic thinking. Ultimately, this research provides highly replicable and promotable practical schemes, alongside robust theoretical references, for the ongoing teaching reform of core foundational courses in computing majors globally.

Keywords: computing education; operating systems; artificial intelligence; hybrid teaching; teaching framework

Received: 24 February 2026

Revised: 01 April 2026

Accepted: 16 April 2026

Published: 22 April 2026



Copyright: © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

1.1. Research Background

In the era of the digital economy, the innovative development of domestic operating systems has placed higher demands on computing professionals. As a core course for computing majors, Principles of Operating Systems undertakes the critical mission of imparting fundamental system theories and cultivating students' systematic thinking and engineering practical abilities. It directly influences students' professional competence in positions such as software development, system operation and maintenance, and kernel development [1].

At present, the course faces several prominent teaching challenges: the course content is highly abstract, with core modules including processes, memory management, and file systems being theoretically intensive and disconnected from real-world applications, resulting in low student interest and great learning difficulties; traditional teaching is dominated by instructor lectures, lacking personalized guidance and thus unable to adapt to students' differentiated learning needs; practical sessions mainly rely

on simulated experiments without exposure to real-world scenarios, leading to low efficiency in transforming theoretical knowledge into practical capabilities [2].

The digital transformation of education has made hybrid teaching a key solution to these dilemmas, while artificial intelligence technologies provide technical support for the precise and personalized upgrading of hybrid teaching. The deep integration of AI and hybrid teaching into Principles of Operating Systems instruction is an inevitable choice for curriculum reform and improving the quality of talent cultivation [3].

1.2. Research Significance

Theoretical Significance: Existing research on the integration of AI and hybrid teaching mostly focuses on general education courses or non-computer major courses. There are relatively few specialized studies on Principles of Operating Systems, and most only stay at the level of simple application of AI tools, lacking a systematic teaching framework. Based on the training objectives of computing majors, this study constructs an AI hybrid teaching framework adapted to the characteristics of the course, enriches the teaching theory of integrating AI technology with core computer courses, and provides a theoretical reference for the reform of similar courses.

Practical Significance: With the support of AI technology, this study solves problems such as the disconnection between theory and practice and insufficient personalized teaching in the course, realizes precise recommendation of teaching content, personalized guidance in the teaching process, and simulation in practical links, so as to improve students' theoretical literacy, systematic thinking ability and engineering practice ability [4]. The research results are replicable and generalizable, which helps to improve the quality of talent training in computing majors in universities and meet the industry's demand for operating system professionals.

1.3. Research Status at Home and Abroad

Foreign research started relatively early. Universities in Europe and the United States have deeply integrated AI into computer courses, developed AI-assisted teaching platforms, implemented learning analytics and resource recommendation based on machine learning, and constructed kernel experiment scenarios through virtual simulation [5]. Hybrid teaching models are well-established, with flipped classroom, project-based learning, and other approaches achieving remarkable effects. However, foreign research is rooted in local educational systems, which is insufficiently compatible with China's professional training objectives and curriculum design, making direct application infeasible.

Domestic scholars have carried out extensive explorations on hybrid teaching of Principles of Operating Systems [6]. Some universities have introduced AI tools such as intelligent Q&A and adaptive learning, yet obvious shortcomings remain: the integration of AI and course content is superficial, without personalized modules designed for key and difficult points; there is a lack of systematic frameworks adapted to students' cognitive characteristics, resulting in fragmented instructional design; practical verification is insufficient, with most studies staying at the theoretical design stage.

In summary, research on systematic AI hybrid teaching frameworks for this course is still insufficient, and specialized research is urgently needed to address existing teaching pain points [7].

1.4. Research Content and Methods

Research Content: First, organize the theoretical and technological foundations, clarify the professional training objectives, the core concepts of AI and hybrid teaching, and their adaptability to the course. Second, examine the current teaching situation of the course and identify the needs and challenges faced by teachers and students [8]. Third, develop a four-layer AI hybrid teaching framework, defining the module functions and implementation processes. Fourth, validate the framework's effectiveness through teaching practice and propose optimization suggestions.

Research Methods: The literature research method is employed to establish the theoretical foundation. The questionnaire survey method and interview method are utilized to gather data on the current teaching situation and determine practical needs [9]. The experimental method is applied to create an experimental group and a control group for comparative verification of teaching outcomes. The case analysis method is used to evaluate issues in framework implementation and refine the optimization plan.

1.5. Research Innovations and Technical Route

Research Innovations: First, targeted innovation: Based on the training objectives of computing majors and course characteristics, this study achieves deep integration of AI with key and challenging course content, rather than merely applying AI tools. Second, systematic innovation: A four-layer framework consisting of an objective layer, core layer, support layer, and evaluation layer is constructed to enable the intelligitization of the entire teaching process and form a comprehensive teaching system [10].

Technical Route: Following the logic of "theoretical foundation---current situation investigation---framework construction---practical verification---optimization and improvement," a complete research closed loop from theory to practice is established [6].

1.6. Thesis Structure

This thesis consists of six chapters: Chapter 1 introduces the research; Chapter 2 elaborates on the relevant theoretical and technological foundations; Chapter 3 investigates and analyzes the current teaching situation; Chapter 4 constructs the AI hybrid teaching framework; Chapter 5 applies and verifies the practical effects of the framework; Chapter 6 summarizes the research conclusions and discusses future prospects [10].

2. Relevant Theories and Technical Foundations

2.1. Definition of Core Concepts

Computing majors cover disciplines such as Computer Science and Technology, Software Engineering, and others, aiming to cultivate high-quality talents for computer research and development, application, and operation and maintenance. Principles of Operating Systems is a core fundamental course of the major, designed to enable students to master core system theories and technologies while fostering systematic thinking and problem-solving abilities.

Hybrid teaching integrates online autonomous learning and offline interactive inquiry, breaking spatial and temporal constraints and optimizing the allocation of teaching resources. Its core principles emphasize student-centeredness, online-offline coordination, integration of theory and practice, and a balance between personalization and standardization [11].

AI hybrid teaching refers to the deep integration of artificial intelligence technology and hybrid teaching, characterized by personalized learning, intelligent tutoring, precise evaluation, and data-driven decision-making [12]. It enables teaching students in accordance with their aptitude through learning analytics, addressing the challenge of insufficient personalization in traditional hybrid teaching.

2.2. Relevant Teaching Theories

Constructivist learning theory emphasizes that learning is a process in which students actively construct knowledge. It provides guidance for online autonomous learning and offline collaborative inquiry in the course, helping students establish a systematic knowledge system. Personalized learning theory respects individual differences among students and offers a theoretical basis for AI-based learning analytics and personalized resource recommendation [13]. Systematic thinking cultivation theory aligns with the systemic nature of the course, guiding students to grasp the internal connections among modules and meeting the talent demands for system development and operation and maintenance. The Technology Acceptance Model supports the

promotion of AI teaching tools, requiring improvements in tool usability and practicality to enhance teachers' and students' willingness to adopt them.

2.3. Application Foundation of AI Technologies in Teaching

The core AI technologies integrated with teaching include machine learning, natural language processing, virtual simulation, and intelligent recommendation, which enable functions such as learning situation analysis, intelligent Q&A, virtual experiments, and personalized resource pushing [14]. Current mainstream AI teaching tools include adaptive learning platforms, intelligent question-answering systems, virtual experiment platforms, and AI-assisted homework grading systems, which have been widely applied in the education sector.

AI technologies demonstrate significant adaptability to Principles of Operating Systems: abstract knowledge can be concretized through visualization and virtual simulation; virtual experiment platforms break the limitations of traditional experiments and restore core operations such as process scheduling and memory allocation; personalized guidance is realized based on learning situation data to meet students' differentiated learning needs [15].

2.4. Core Elements and Implementation Modes of Hybrid Teaching

The core elements of hybrid teaching include online teaching resources, offline teaching activities, teaching evaluation systems, teacher-student interaction mechanisms, and technical support systems. The main implementation modes suitable for the course are the flipped classroom mode and project-based hybrid teaching mode: the flipped classroom is applicable to theoretical chapters, realizing pre-class autonomous learning and in-class inquiry and answering; the project-based mode is suitable for practical chapters, integrating theory and practice through task-driven learning. Existing hybrid teaching still suffers from problems such as insufficient personalization and loose integration of online and offline components, which urgently require empowerment and upgrading by AI technologies [16].

3. Investigation and Problem Analysis of Current Teaching Situation

3.1. Survey Design and Implementation

This survey focuses on four dimensions: teaching content, teaching mode, AI application, and teaching evaluation. Respondents were selected from 30 teachers and 200 computing major students at three domestic universities of different levels. A total of 186 valid student questionnaires and 28 valid teacher questionnaires were collected, with an effective recovery rate both exceeding 93%. Additionally, semi-structured interviews were conducted with 10 teachers and 15 students to comprehensively gather teaching pain points and improvement suggestions, providing a practical basis for the framework construction.

3.2. Analysis of Current Teaching Situation

Teaching Content: Courses primarily focus on classical theories covering core modules but lack sufficient integration of traditional theories with advanced technologies such as domestic operating systems and cloud computing [17]. The content remains abstract and disconnected from practical applications. There is an imbalance between theoretical and practical emphasis across different colleges, failing to achieve a cohesive integration.

Teaching Mode: Most universities employ hybrid teaching methods; however, online learning is limited to resource dissemination without adequate supervision or guidance [10]. Offline classes are still predominantly lecture-based, with minimal interaction and practical engagement. The integration of online and offline teaching remains superficial, and there is a need to enhance teachers' instructional design capabilities.

AI Technology Application: AI is applied only in isolated scenarios, such as intelligent Q&A and homework grading, rather than being integrated throughout the teaching process. Both teachers and students lack sufficient digital literacy, resulting in

low acceptance of technology [17]. The technical support systems in universities are underdeveloped, and AI tools lack stability and practicality.

Teaching Evaluation: Summative evaluation, dominated by final examinations, constitutes 60%–70% of the assessment, while process evaluation is largely superficial. Evaluation indicators and participants are limited, with no personalized standards, making it challenging to comprehensively reflect students' learning outcomes.

3.3. Core Problems and Their Causes

Core Problems: The disconnection between theory and practice results in weak engineering practical abilities among students. A lack of personalized teaching makes it challenging to address differentiated learning needs. AI applications remain superficial and fail to fully exert their empowering effects [2, 11]. Additionally, the cultivation of systematic thinking is insufficient, leading to fragmented knowledge structures.

Causes: An outdated teaching philosophy emphasizes knowledge impartation over the cultivation of abilities. There is insufficient digital literacy among instructors and a lack of competence in designing AI-integrated hybrid teaching. Furthermore, there is a shortage of high-quality AI teaching resources and an absence of a systematic teaching framework, resulting in scattered and fragmented instructional links. The evaluation system is also imperfect, lacking effective diagnostic and improvement functions.

3.4. Survey Conclusions and Framework Construction Requirements

The survey indicates that hybrid teaching and AI tools have been preliminarily implemented in course instruction, yet several challenges persist. There is an urgent need to develop a systematic AI hybrid teaching framework [6]. This framework should fulfill five essential requirements: adaptability, integration, practicality, scalability, and targeting. It must align with professional training objectives, enable the deep integration of AI and teaching, and address the fundamental issues in course instruction.

4. Construction of the AI Hybrid Teaching Framework

4.1. Guiding Ideology and Principles of Framework Construction

Guiding Ideology: Taking morality education as the fundamental task, this framework adheres to the philosophy of student-centeredness, AI empowerment, integration of knowledge and practice, and teaching students in accordance with their aptitude. Focusing on the key and difficult points of the course, it integrates AI technology and hybrid teaching to cultivate students' theoretical literacy, practical ability, and systematic thinking, thereby meeting the talent demands of the industry.

Construction Principles: Uphold systematicness to coordinate all teaching links and form a complete system. Uphold personalization to realize individualized teaching supported by AI. Uphold practicality to simplify the design for smooth implementation. Uphold innovation to overcome the limitations of traditional teaching. Uphold scalability to adapt to technological iteration and changing demands.

4.2. Overall Framework Design

A four-layer integrated framework is constructed, consisting of the objective layer, core layer, support layer, and evaluation layer, with coordinated and closed-loop interactions among all layers. Guided by the objective layer, implemented by the core layer, supported by the support layer, and fed back by the evaluation layer, the framework enables intelligent and precise management of the entire teaching process.

The core feature of the framework is the deep integration of AI with course content and teaching links, emphasizing the cultivation of practical ability and systematic thinking while adapting to personalized learning needs [9]. The AI-integrated teaching framework is illustrated in Figure 1.

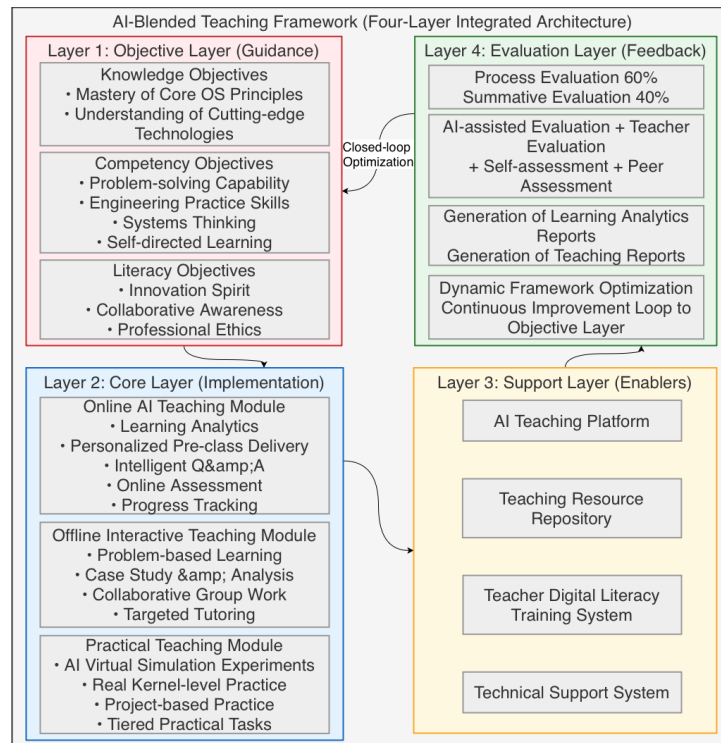


Figure 1. AI-integrated teaching framework.

4.3. Detailed Design of Each Module

Target Layer: It is divided into three dimensions: knowledge, ability, and literacy. The knowledge goal requires mastering the core principles and cutting-edge technologies of the system; the ability goal focuses on problem-solving, engineering practice, system thinking, and autonomous learning skills; the literacy goal aims to cultivate an innovative spirit, collaboration awareness, and professional qualities.

Core Layer: It consists of three major modules: online AI teaching, offline interactive teaching, and practical teaching [7]. Online AI teaching realizes learning situation analysis, personalized preview push, intelligent answering, online tests, and progress tracking; offline interactive teaching adopts problem-driven approaches, case analysis, group collaboration, and precise guidance to address the challenges of online learning; practical teaching integrates AI virtual simulation experiments, real-world core practice, project-based practice, and tiered practical tasks, dividing students into different levels to enhance their practical abilities. The detailed structure diagram of the three modules of the core layer is shown in Figure 2.

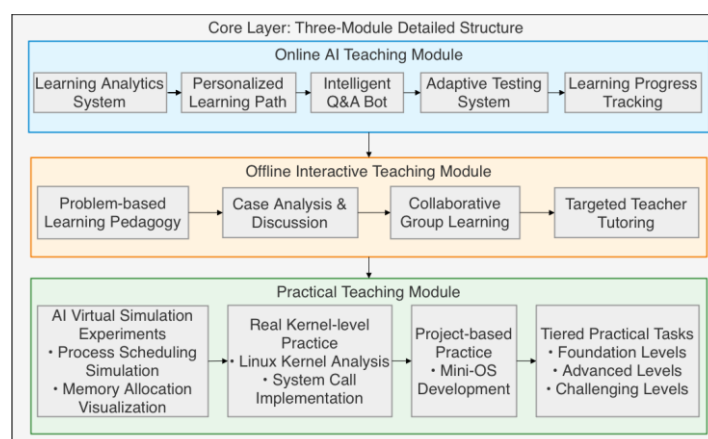


Figure 2. Detailed structure of the three modules in the core layer.

Support Layer: Composed of an AI teaching platform, a teaching resource repository, a teacher digital literacy training system, and a technical support system, it provides technical, resource, personnel, hardware, and software guarantees for the implementation of the framework.

Evaluation Layer: Adopts a combination of formative evaluation (60%) and summative evaluation (40%), integrating AI-assisted evaluation with multiple evaluation methods including teacher assessment, self-assessment, and peer assessment [14]. It comprehensively reflects learning outcomes and provides data support for framework optimization.

4.4. Framework Implementation Process

The implementation of the framework consists of four stages: pre-class preparation, in-class implementation, post-class consolidation, and evaluation and optimization, forming a closed-loop teaching process. Pre-class involves AI conducting learning status analysis and delivering personalized resource recommendations, while teachers develop customized teaching plans. In-class focuses on addressing difficult knowledge points through interactive teaching and conducting real-time tests to assess learning effectiveness. Post-class includes the provision of personalized review resources, hierarchical practical tasks, and comprehensive Q&A support throughout the process. Evaluation and optimization involve generating learning status and teaching reports, dynamically refining each module of the framework.

4.5. Core Features of the Framework

First, it accurately aligns with the talent requirements of computing majors, focusing on the development of engineering practice and systematic thinking. Second, it integrates artificial intelligence throughout the teaching process to achieve intelligent and precise instruction. Third, it balances systematization and personalization to address the challenges of differentiated learning. Fourth, it deeply combines theoretical knowledge with practical application, enhancing virtual simulation with real kernel practice. Fifth, it offers strong operability and scalability, enabling widespread adoption and iterative improvement.

5. Practical Application and Effect Verification of the Teaching Framework

5.1. Practical Scheme Design

Two classes of Grade 2024 majoring in Computer Science at a university were selected as research subjects. The experimental group, consisting of 45 students, adopted the teaching framework proposed in this study, while the control group, comprising 44 students, received traditional hybrid teaching. The teaching experiment lasted for 16 weeks. Both groups were consistent in terms of students' foundational knowledge, teaching faculty, course content, and teaching schedule, with variable control implemented to ensure scientific and reliable results. The practical objective was to verify the effectiveness of the framework in improving academic performance, learning interest, practical ability, and systematic thinking, as well as to collect suggestions for further optimization.

5.2. Implementation of the Practical Process

Preparatory stage: An AI teaching platform was established, course resources were integrated, teacher training was conducted, and pre-practice basic tests were completed for students. Mid-implementation stage: The experimental group engaged in personalized pre-class preparation, interactive teaching during class, post-class hierarchical practice, and intelligent tutoring based on the framework; the control group followed the traditional approach of online resource recommendation and offline lectures [5]. During the practice, to address challenges such as students' unfamiliarity with tools, low initiative in autonomous learning, and insufficient accuracy of AI Q&A, tool training was conducted, evaluation weights were optimized, and algorithms and question databases were improved to ensure the smooth progression of teaching.

5.3. Effect Evaluation and Data Analysis

A combination of quantitative and qualitative methods was employed to conduct a comparative analysis across four dimensions: academic performance, learning interest, practical ability, and teacher-student satisfaction.

Academic Performance: The average scores of the two groups in the preliminary test were similar (68.5 vs. 67.8). In the mid-term examination, the experimental group scored 76.3 and the control group 70.2, with a difference of 6.1 points. In the final examination, the experimental group scored 80.5 and the control group 73.1, with a difference of 7.4 points, indicating that the framework significantly improved knowledge mastery.

Learning Interest: The experimental group achieved a Likert scale score of 4.2, with 82.2% of students showing interest, while the control group scored 3.3 with only 58.1% of students interested, demonstrating that the framework effectively stimulated learning initiative [4].

Practical Ability: The average score of practical reports in the experimental group was 85.3, the standard rate of experimental operation was 93.3%, and the project completion rate was 88.9%, all significantly higher than those of the control group, showing an obvious improvement in practical ability.

Teacher-Student Satisfaction: Both teachers and students expressed satisfaction rates of over 90% with the framework in terms of usability, teaching effectiveness, and personalized guidance, recognizing its feasibility and practicality.

Practice has demonstrated that the framework can effectively address the challenges of traditional teaching in Operating System Principles, enhance teaching quality and talent training effectiveness, and is worthy of broader application [16].

6. Conclusions and Prospects of the Research

6.1. Research Conclusions

This study constructs an AI-enabled hybrid teaching framework tailored to the training objectives of computing-related majors and addressing key challenges in teaching Operating System Principles. The framework comprises an objective layer, core layer, support layer, and evaluation layer, with clearly defined design principles, module functions, and implementation processes. Teaching practice has verified that the framework achieves deep integration of AI technology with curriculum teaching, significantly enhancing students' academic performance, learning interest, practical skills, and systematic thinking. Furthermore, it effectively resolves issues such as the disconnect between theory and practice, limited personalized teaching, and superficial application of AI. This framework offers a replicable solution for teaching reform in similar core courses within computer-related disciplines.

6.2. Research Limitations and Prospects

This study has certain limitations: the practical sample is restricted to classes within a single university, which limits the generalizability of the findings; the accuracy and functional adaptability of AI tools require further refinement to enhance their effectiveness; and no follow-up investigation has been conducted to explore the correlation between long-term teaching outcomes and graduates' professional competencies.

Future research will focus on three key areas: broadening the scope of practical applications to assess the universality of the framework across diverse universities and academic disciplines; advancing AI teaching tools by optimizing algorithms for learning situation analysis and intelligent Q&A; and developing a long-term tracking mechanism to integrate curriculum teaching with professional skill development, thereby continuously improving the teaching framework and supporting the cultivation of high-quality talent in computing-related fields.

Funding: This work is supported by the Quality Engineering Project of Guangdong University of Science and Technology under the project number GKZLGC2025046.

References

1. A. D. Cheok, "HOW TO ESTABLISH A NEW AI LABORATORY," 2022.
2. S. Pamplona, N. Medinilla, and P. Flores, "Exploring misconceptions of operating systems in an online course," in *Proceedings of the 13th Koli Calling International Conference on Computing Education Research*, Nov. 2013, pp. 77-86.
3. R. S. Pinto, P. Nobile, E. Mamani, L. P. Júnior, H. J. Luz, and F. J. Monaco, "Operating System from the Scratch: a Problem-Based Learning Approach for the Emerging Demands on OS Development," *Procedia Computer Science*, vol. 18, pp. 2472-2481, 2013.
4. M. R. Ebling, "Resources for teaching operating systems: A survey of instructors and a literature review," *ACM Transactions on Computing Education*, vol. 24, no. 4, pp. 1-28, 2024.
5. V. H. S. de Camargo, M. M. C. Junior, F. F. da Silva, and L. B. R. Aylon, "Learning Operating Systems with Educational Games: A Systematic Mapping of the Literature," *Journal on Interactive Systems*, vol. 16, no. 1, pp. 718-732, 2025.
6. J. Andrus and J. Nieh, "Teaching operating systems using android," in *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*, Feb. 2012, pp. 613-618.
7. A. Garmpis, "Design and development of a web-based interactive software tool for teaching operating systems," *Journal of Information Technology Education: Research*, vol. 10, no. 1, pp. 1-17, 2011.
8. L. P. Maia, F. B. Machado, and A. C. Pacheco Jr, "A constructivist framework for operating systems education: a pedagogic proposal using the SOsim," in *Proceedings of the 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*, Jun. 2005, pp. 218-222.
9. A. Cvetkovski, "Operating system virtualization in the education of computer science students," *IJAEDU-International E-Journal of Advances in Education*, vol. 7, no. 19, pp. 42-47, 2021.
10. A. Garmpis, "Innovative teaching methods in operating systems: the Linux case," in *International Conference of Innovative Approaches in Education: Design and Networking*, PATRAS-Greece, Jan. 2012.
11. S. Pamplona, N. Medinilla, and P. Flores, "A systematic map for improving teaching and learning in undergraduate operating systems courses," *IEEE Access*, vol. 6, pp. 60974-60992, 2018.
12. M. Lopez-Nores, Y. Blanco-Fernandez, and J. J. Pazos-Arias, "RETHINKING OPERATING SYSTEMS TEACHING TOWARDS THE EUROPEAN HIGHER EDUCATION AREA THROUGH PROBLEM-BASED LEARNING," in *ICERI2010 Proceedings, IATED*, 2010, pp. 1284-1290.
13. H. M. Deitel and P. J. Deitel, *Operating systems*. Pearson Education India, 2004.
14. J. Nieh and C. Vaill, "Experiences teaching operating systems using virtual platforms and linux," in *Proceedings of the 36th SIGCSE Technical Symposium on Computer Science Education*, Feb. 2005, pp. 520-524.
15. T. E. Unwana, E. I. Udoh, and V. O. Umoh, "A Study of the Importance of Operating System (OS) in a Computer System," 2022.
16. B. Mustafa, "YASS: A System Simulator for Operating System and Computer Architecture Teaching and Learning," *European Journal of Science and Mathematics Education*, vol. 1, no. 1, pp. 34-42, 2013.
17. R. Parkavi, S. Sujitha, and A. Sheik Abdullah, "Enhancing Student Engagement and Success in Operating Systems through Simulation based Learning and Formative Assessments," *Journal of Engineering Education Transformations*, pp. 444-450, 2026.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of Publisher and/or the editor(s). Publisher and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.