Research on the Path to Enhance Blended Teaching Quality in Software Engineering under the OBE Concept

SongHao Li[1, a]

[1]Xi’an Technological University, Xi’an, Shaanxi, China

[a]3044438395@qq.com

Abstract. In response to the urgent demand for high-calibre, multidisciplinary software engineering professionals arising from the development of new engineering disciplines, and to address critical bottlenecks in traditional software engineering education—such as the disconnect between theory and practice and insufficient student agency—this research endeavours to establish a systematic, quantifiable blended teaching quality enhancement strategy. Guided by the logic of Outcome-Based Education (OBE), this paper undertakes a comprehensive and systematic backward redesign of the software engineering programme's talent development objectives, curriculum design, teaching implementation, and assessment framework. The core innovation lies in proposing and implementing a tripartite, deeply integrated teaching model centred on Project-Based Learning (PBL), encompassing online, offline, and practical components, alongside a fully integrated quality monitoring and continuous feedback mechanism. Empirical application and data analysis demonstrate that this approach significantly enhances the actual attainment of students' Course Learning Outcomes (CLOs). Particularly in core engineering competency metrics—engineering practice, complex problem-solving, and team collaboration—student performance shows substantial and systematic improvement, with teaching quality assessment indicators effectively enhanced. Research confirms that the blended teaching model guided by OBE principles represents an effective, actionable, and quantifiable pathway for enhancing the quality of software engineering talent cultivation. It provides crucial theoretical underpinnings and practical reference value for curriculum reform and implementation within comparable engineering education domains.

Keywords: Outcomes-Based Education；Software Engineering；Project-Based Learning；Blended Learning；Teaching Quality Evaluation；

1. Introduction

1.1 Background: Digital Transformation and "New Engineering"

The accelerating pace of global digital transformation, coupled with the deepening implementation of China's "New Engineering" strategy, is collectively driving a transformation in the role of software engineering. It has evolved beyond being merely a technical discipline to become a core driver of societal innovation. The rapid advancement of next-generation information technologies—including artificial intelligence, cloud computing, big data, and the Internet of Things—has imposed higher and more complex demands upon higher education. Consequently, societal expectations for software engineering professionals no longer centre solely on mastery of programming languages and foundational theories. Instead, there is a growing demand for versatile, engineering-oriented, and internationally-minded talent capable of rapid knowledge acquisition, resolving complex engineering challenges, and collaborating across disciplines to foster innovation and entrepreneurship.

1.2 The Paradigm of Outcome-Based Education

Against this backdrop, Outcome-Based Education emerges as an advanced pedagogical philosophy. Its core value lies in prioritising graduate outcomes, employing reverse engineering of the teaching system based on engineering accreditation standards and industry requirements. This model aligns profoundly with the engineering nature of software engineering, becoming a key driver in ensuring high-quality, efficient, and high-standard talent cultivation.

1.3 Current Challenges in Traditional Teaching

https://doi.org/10.61957/joerai-20260104

Despite significant progress in China's software engineering education, the inherent limitations of traditional teaching paradigms have become increasingly apparent when confronting rapidly evolving industry demands, severely constraining improvements in talent cultivation quality. The core issues manifest in three primary aspects:

1) Outdated Content

Teaching content lags behind updates, creating a significant gap between academic instruction and real-world software project development processes, mainstream tools, and standards.

2) Passive Learning

Students exhibit a lack of agency due to a heavy reliance on one-way lecturing, which stifles critical thinking and autonomous learning.

3) Simplistic Assessment

Mechanisms prioritise end-of-term examinations while undervaluing formative assessments, failing to evaluate crucial competencies like teamwork and project management.

1.4 Research Objectives and Pathways

Given these multifaceted challenges, transforming the pedagogical model of software engineering programmes has become a core imperative. This research, grounded in the principles of OBE and aligned with trends in educational informatisation, aims to establish a blended teaching quality enhancement pathway that deeply integrates online, offline, and practical components.

1.5 Research Innovations

The innovation of this research is primarily reflected in the following three dimensions:

1) Systematic Closed-Loop Reconstruction

Guided by OBE logic, the entire teaching process—encompassing the decomposition of course objectives, implementation models, multi-dimensional assessment, and continuous improvement—has undergone systematic closed-loop reconstruction. This ensures a high degree of alignment between objectives and practice, guaranteeing the sustainability and traceability of teaching quality.

2) Triadic Coupling Mechanism

This research transcends the conventional hybrid model of merely superimposing resources. It innovatively proposes a triadic coupling mechanism integrating online, offline, and practical components. Specifically, online activities provide "feedforward drive," offline sessions focus on "intellectual engagement," and practical data enables "feedback optimisation".

3) Operationally Robust Pathway Model

Through the reconstruction of the Course Learning Outcome (CLO) system and the establishment of a multi-dimensional formative assessment framework, this research constructs an effect-quantifiable pathway model. The findings provide a theoretically grounded framework and practical blueprint of considerable reference value for other applied disciplines within the "New Engineering" context.

2. Theoretical Basis and Concept

2.1 The Core Principles of Outcome-Based Education (OBE) and Its Application in Software Engineering

Outcomes-Based Education represents a global trend in educational reform, signifying a fundamental shift in educational philosophy from teacher-centred to learner-centred approaches. Its core logic lies in “backward design”, whereby one first defines the knowledge, competencies and attributes students should possess upon completing their studies (i.e. graduate requirements or learning outcomes). These overarching graduate requirements are then systematically broken down and mapped to the Course Learning Outcomes of each core module. The core characteristics of outcome-based education are illustrated in Fig. 1 below:

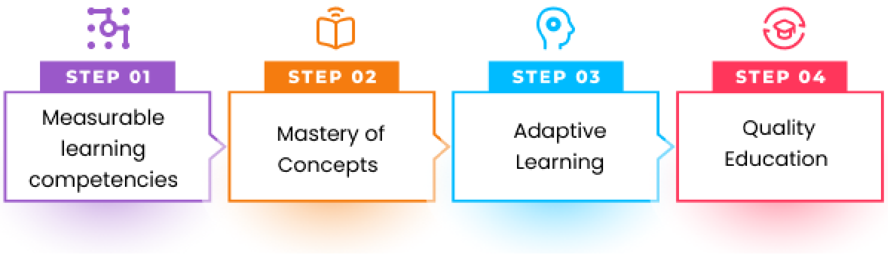

Figure 1. Core Features of Outcome-Based Education

Rote memorization and exam centricity are two flaws that are central to the traditional education system, which contributes little to nothing to the student's learning experience. However, OBE's well-designed methodologies empower students to learn effectively.In fact, they succeed at learning critical thinking abilities and problem-solving capacities and applying them in real-world scenarios. As a result, they are able to handle diverse problems and develop creative solutions accordingly.The steps for implementing outcome-based education are outlined in Fig. 2 below:

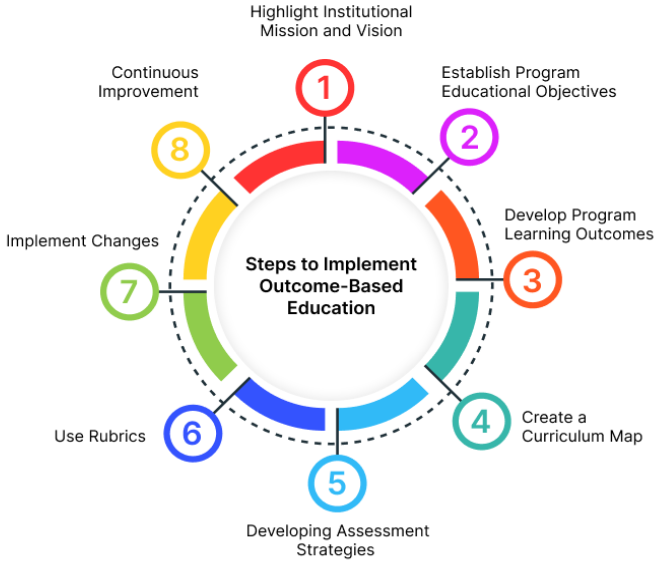
Steps to Implement Outcome-Based Education
1
Highlight Institutional Mission and Vision
2
Establish Program Educational Objectives
3
Develop Program Learning Outcomes
4
Create a Curriculum Map
5
Developing Assessment Strategies
6
Use Rubrics
7
Implement Changes
8
Continuous Improvement

Figure 2. Steps to Implement Outcome-Based Education

Within the field of software engineering, the application of Outcome-Based Education demonstrates particularly significant value, primarily manifested in two aspects: its goal-focused nature and its continuous improvement mechanism. Regarding goal focus, OBE requires the translation of abstract engineering competencies (such as problem analysis and system design) into specific, observable, and measurable course learning outcomes. This ensures teaching is no longer a mere enumeration of knowledge points but is instead centred around clearly defined competency objectives. Concerning the continuous improvement mechanism, the core of OBE lies in assessing the attainment of CLOs. Through quantitative evaluation, it clearly identifies deficiencies in students' competencies, thereby guiding teaching staff to provide feedback and continuously refine teaching design, methodologies, and assessment tools. This ensures a spiralling upward trajectory in the quality of talent cultivation.

2.2 The Deep Integration of Blended Learning and Project-Based Learning

Blended learning is not merely the juxtaposition of online resources and offline classrooms, but rather emphasises the organic integration and complementary strengths of both learning formats. Online learning platforms offer flexibility, resource richness, and data tracking capabilities, making them inherently suited for theoretical knowledge preparation, fragmented learning, and personalised tutoring. In-person classrooms, conversely, provide students with spaces for deep interaction, complex problem discussions, team collaboration, and emotional exchange. This profound combination and functional complementarity furnish the essential environment for cultivating the complex engineering practice skills and team collaboration abilities required in software engineering programmes.

Within the Software Engineering programme, blended learning should be deeply integrated with Project-Based Learning to ensure students achieve the course's knowledge and competency outcomes whilst undertaking real-world projects. The PBL model aligns with the discipline's professional focus on solving complex problems and delivering products. By introducing authentic or simulated engineering projects, it enables students to master core practical skills through learning by doing.

Deepening Project-Based Learning: The essence of software engineering lies in solving complex problems and delivering products. The PBL model inherently aligns with this professional characteristic. By introducing real or simulated engineering projects, students not only master theoretical knowledge during project completion but also acquire a range of engineering practice skills—including requirements analysis, system design, code implementation, and testing/debugging—ultimately internalising knowledge and competencies. Blended learning provides robust support for PBL. For instance: online platforms facilitate project documentation management and daily team communication, while in-person sessions host technical reviews at critical junctures and agile development meetings, ensuring learning occurs within an engineering-oriented environment.

2.3 Principles for Establishing Pathways to Enhance Teaching Quality

Based on the principles of Outcome-Based Education and the characteristics of the Software Engineering programme, the blended teaching quality enhancement pathway developed in this study must adhere to three core principles to ensure its scientific rigour and effectiveness.

First is the Outcome-Based Education principle. Its core tenet requires that all course learning outcomes teaching content, pedagogical methods, and assessment must be closely aligned with graduate requirements and course learning outcomes, thereby eliminating content unrelated to competency development. In practical terms, instructional design must employ reverse engineering: first defining the learning outcomes students must achieve, then designing the teaching activities and assessment tools to attain these outcomes. This ensures a high degree of consistency between objectives, teaching, and evaluation.

Secondly, the principle of engineering practice orientation. Given that software engineering is a highly applied discipline, the core tenet of this principle demands that teaching activities must transition from theory to practice. Specifically, this entails integrating authentic, open-ended, and challenging engineering projects throughout the entire teaching process, thereby fostering a learning environment characterised by 'learning by doing'. Its practical requirements entail extensive use of case studies, laboratory-based teaching, and project-driven approaches within instruction. This extends the classroom to laboratories and industry settings, enabling students to master engineering thinking and standards through learning by doing.

Finally, the principle of process closure (continuous improvement). The core tenet of this principle is that enhancing teaching quality constitutes an ongoing optimisation process. Consequently, it necessitates establishing a scientific quality monitoring and feedback mechanism, requiring the formation of a PDCA (Plan-Do-Check-Act) quality management cycle encompassing 'design-implementation-evaluation-feedback-continuous improvement'. Its practical requirements entail teachers regularly conducting quantitative assessments (Check) of students' CLO attainment through diverse evaluation tools (such as questionnaires, performance data, and project review outcomes). These assessment results then serve as the basis for adjusting teaching methods and course content (Act), thereby ensuring dynamic and sustained enhancement of teaching quality.

3. Establishing Pathways for Enhancing the Quality of Blended Learning

The blended teaching quality enhancement pathway developed in this study constitutes a systematic engineering initiative designed to comprehensively implement the Outcomes-Based Education philosophy. This pathway comprises three core components. Firstly, objective restructuring involves refining and quantifying Course Learning Outcomes through backward design, starting from graduate requirements. Secondly, pedagogical innovation establishes a tripartite integrated teaching mechanism centred on Project-Based Learning, encompassing online, offline, and practical components. Finally, it involves the reshaping of the assessment system, establishing a diversified evaluation framework centred on formative assessment and forming a continuous quality improvement loop. Together, these three elements constitute a systematic solution for enhancing the quality of talent cultivation in the Software Engineering discipline.

3.1 Objective Reconstruction: Refined Breakdown of Course Learning Outcomes

The OBE philosophy mandates that instructional design must employ backward design methodology. This process commences with the national and industry expectations for software engineers—specifically, the Graduation Outcomes—and proceeds to reverse-engineer and design specific Course Learning Outcomes. This meticulous decomposition of CLO constitutes a fundamental step in ensuring teaching outputs are quantifiable and assessable, rather than merely undergoing a formal conversion. The detailed decomposition of Software Engineering programme graduation outcomes into Course Learning Outcomes is presented in Table 1.

Table 1 Detailed Breakdown of Graduation Requirements for the Software Engineering Programme

Graduation Requirements	CLOs	CLO Specific Definition
Master knowledge	CLO 1	Be able to accurately articulate the fundamental theories and core technologies of software engineering, and conduct critical analysis of emerging technological trends.
Problem analysis capability	CLO 2	Identify, analyse and articulate complex software engineering requirements with clarity, constructing formalised requirements specifications and problem solutions.
Design/Development Solutions	CLO 3	Proficient in applying engineering methodologies to design and implement software systems that meet specific constraints such as security, performance, and availability, and capable of independently conducting unit, integration, and system-level testing and verification.

3.2 Model Innovation: Tripartite Integration of Online-Offline-Practical Application

The blended teaching model developed in this study does not merely superimpose online resources onto offline activities. Instead, it emphasises the deep integration and cyclical driving mechanism among online, offline, and practical components, forming a closed-loop teaching ecosystem. This ensures seamless continuity in knowledge transmission, intellectual exchange, and skill development.

1) The “feedforward drive” from online to offline

The mechanism aims to leverage the flexibility of online platforms to achieve personalised knowledge acquisition and assess preparatory learning outcomes, thereby ensuring the efficiency of in-person classroom sessions. In practical implementation, teachers publish micro-lectures, theoretical handouts, and pre-assigned self-assessment tasks on online platforms such as MOOCs or SPOCs. The platform tracks behavioural data in real time, including students' learning progress, screen engagement duration, and self-assessment scores. Concurrently, an “offline threshold” is established: only students who complete the prescribed online learning tasks and achieve a specified self-assessment score are permitted to participate in the in-person seminar sessions. This mechanism shifts the burden of theoretical knowledge acquisition from the classroom to the pre-class period, thereby safeguarding the prerequisite for effective offline interaction.

2) Offline ‘brainstorming’ sessions for practical application

The mechanism aims to leverage the high interactivity of offline classrooms to address complex theoretical issues that cannot be overcome online, while fostering preliminary project solutions through group collaboration. In practice, offline classroom time is entirely dedicated to complex case analysis, technical solution discussions, and group code reviews, rather than theoretical repetition. Instructors precisely identify common challenges based on online preparatory assessment data, organising debates or brainstorming sessions to stimulate higher-order thinking. For instance, when addressing complex scenarios such as ‘project requirement changes,’ student teams are tasked with developing agile development (Scrum)project management plans. Each in-person workshop centres on an ‘output-oriented’ approach, where deliverables must constitute phased outcomes of the project practice (e.g., draft requirement specifications or system architecture diagrams), directly serving as the starting point for subsequent project work.

3) Implementing online ‘feedback optimisation’

The mechanism aims to utilise project practice to assess knowledge acquisition and the attainment of CLOs, whilst feeding back data from the practical process to refine teaching content. In specific implementation, students are required to employ professional tools such as Git/SVN for code submission and version control during project practice. These submission records, code standards, and team collaboration frequency are collected as vital sources of formative assessment data. Common technical challenges encountered by students during practice (such as concurrency handling or design pattern selection) are collated and summarised by teaching staff. These are rapidly developed into new online micro-lecture resources and integrated into the learning platform, ensuring teaching content can swiftly respond to cutting-edge engineering developments and practical industry pain points. Furthermore, the final assessment outcomes of practical projects serve not only for grading but also for analysing the attainment of Course Learning Outcomes. Evaluation results—such as a lower-than-expected achievement rate for CLO 4—will be fed back to online course designers. This informs adjustments to subsequent online resources or offline seminar topics, establishing a continuous improvement cycle. The relationship between the three components is clearly illustrated in Fig. 3 below, showing how they interact with each other.

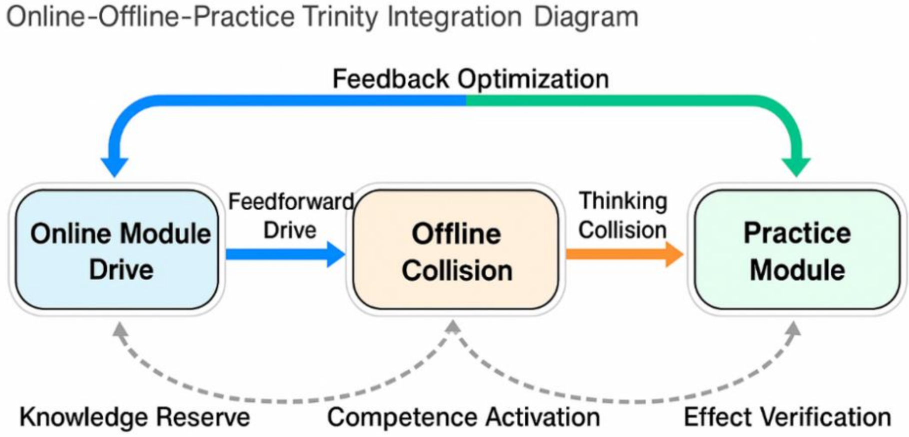

Figure 3. Online-Offline-Practice Triadic Relationship Integration Diagram

Through the above tripartite coupling of “forward-feed drive—ideas collision—back-feed optimisation”, this blended teaching pathway achieves a high degree of integration between knowledge, skills and practice, overcoming the fragmentation between online and offline learning, and between classroom instruction and practical application, that characterises traditional models.

3.3 Re-engineering the Evaluation System: Diversified Assessment and Quality Feedback Loops Based on the OBE Approach

Outcomes-Based Education mandates that assessment methods be directly linked to Course Learning Outcomes to achieve a comprehensive evaluation of students' knowledge, competencies, and competencies. This research fundamentally reforms the traditional single-assessment model of ‘one exam determining one's entire academic future,’ establishing a diversified assessment system centred on formative assessment with summative assessment as a supplement. This system is integrated into a continuous quality improvement loop.

In terms of assessment composition, this approach significantly increases the weighting of formative assessments. These encompass multiple dimensions including online learning engagement, team collaboration and adherence to standards, as well as project milestone reports and reviews. This design aims to comprehensively evaluate students' performance in autonomous knowledge acquisition, professional conduct within engineering practice, teamwork, and engineering documentation writing. Summative assessments comprise final project evaluations and theoretical knowledge examinations, serving to validate students' ability to solve complex engineering problems and their grasp of fundamental theories.

The ultimate objective of this evaluation framework is to establish a PDCA (Plan-Do-Check-Act) closed-loop management system for teaching quality, thereby creating a continuous improvement mechanism. Teaching staff must regularly conduct quantitative assessments (Check) of students' CLO attainment using diverse evaluation tools. Should assessment results indicate that attainment of any CLO falls below the predetermined target, this outcome is immediately fed back to the teaching team. This serves as the basis for adjusting the next round of teaching design or activity weighting. This mechanism ensures that the enhancement of teaching quality is a dynamic, continuously optimised process.

4. Practical Application and Empirical Analysis of Blended Learning Pathways

4.1 Subject of Practice and Implementation Process

This study selected a core course in software engineering as a pilot programme, conducting a semester-long teaching practice and comparative analysis. The subjects were divided into a reform class employing the ‘online-offline-practical’ tripartite blended teaching pathway developed in this research, and a control class adhering to traditional teaching methods.

Within the reform group, teaching activities were strictly implemented in accordance with the OBE philosophy. This involved reconstructing course objectives through reverse design, employing a project-driven (PBL) approach as the core of the tripartite integrated model, and utilising diverse formative assessments for quality monitoring.

4.2 Empirical Effect Analysis and Quantitative Findings

Empirical data collection and comparison primarily assessed three core dimensions: Course Learning Outcome attainment, enhancement of engineering competencies, and student engagement levels. The findings indicate that the reformed cohort significantly outperformed the control group across multiple key metrics.

Specifically, students in the reformed class, which adopted a blended learning pathway, demonstrated markedly enhanced capabilities in solving complex engineering problems (as reflected in project scores). Concurrently, their teamwork and communication skills saw substantial improvement, attributable to the systematic promotion of the project-based learning model. Furthermore, online platform data tracking revealed a notable increase in learning engagement among reformed class students, underscoring the blended approach's effectiveness in stimulating student initiative and academic interest.

4.3 Summary of Practical Outcomes and Teaching Advantages

Empirical findings demonstrate that the blended teaching approach guided by the OBE philosophy exhibits significant advantages when applied to software engineering programmes: the tripartite integration model compels students to apply theoretical knowledge to project practice, achieving a transformation from ‘knowledge points’ to ‘competency chains’ and markedly enhancing learning depth and applicability. By incorporating engineering standards and diverse formative assessments, the teaching process effectively cultivates core engineering competencies such as time management, standardised coding, and team collaboration. This approach provides a quantifiable, traceable quality monitoring and continuous improvement mechanism through precise assessment of CLO attainment, representing an effective pathway for enhancing the quality of software engineering talent cultivation.

5. Conclusions and Outlook

This study has established a blended teaching quality enhancement pathway for software engineering under the OBE philosophy, integrating online, offline, and practical components. Through objective restructuring, pedagogical innovation, and assessment redesign, this pathway successfully addresses the shortcomings of traditional teaching, markedly enhancing students' engineering practice capabilities and learning engagement. It represents an effective solution for software engineering programmes to adapt to the new engineering paradigm and elevate talent cultivation quality. Whilst demonstrating initial success, the pathway retains scope for refinement: future exploration should focus on leveraging artificial intelligence and big data analytics to conduct real-time analysis of student behavioural data within blended learning platforms, enabling more precise learning alerts and personalised tutoring.

References

[1] Sun D ,Li Z ,Ou M , et al.Exploration of Teaching Reform Strategies for the “Principles of Chemical Engineering” Course Based on the OBE Concept[J].Education Reform and Development,2025,7(9):260-265.DOI:10.26689/ERD.V7I9.12396.

[2] Adedoyin O O, Shangodoyin D K. Concepts and practices of outcome based education for effective educational system in Botswana[J]. European Journal of Social Sciences, 2010, 13(2): 161-170.

[3] Glatthorn A A. Outcome-based education: Reform and the curriculum process[J]. Journal of Curriculum & Supervision, 1993, 8(4).

[4] Zhao K .Research on the Teaching Reform Path for the Intelligent Construction Major Based o n the OBE Concept and Empowered by AI[J].Advances in Educational Technology and Psycho logy,2025,9(5):DOI:10.23977/AETP.2025.090513.

[5] Sengul C ,Neykova R ,Destefanis G .Software engineering education in the era of conversationa l AI: current trends and future directions[J].Frontiers in Artificial Intelligence,2024,71436350-1 436350.DOI:10.3389/FRAI.2024.1436350.

[6] The influence of project-based learning on engineering students’ academic and career motivatio n[J].Journal of Applied Research in Higher Education,2025,17(5):2037-2050.DOI:10.1108/JAR HE-04-2024-0160.

[7] Zhu C .Exploration and Practice of the Online - Offline Hybrid Teaching Model Featuring Synchronized Theory and Practice, and Integration of Virtual and Real Elements[J].Frontiers in Educational Research,2025,8(9):DOI:10.25236/FER.2025.080904.

[8] Fu Z ,Diao C .Research on the Teaching Reform of the Online and Offline Mixed Course of "Innovation and Entrepreneurship Theory and Practice" Based on the OBE Concept[J].Journal of Higher Education Teaching,2025,2(3):DOI:10.62517/JHET.202515313.

[9] Fang Q .A Systematic Study of the Integration of Internet Technology to Improve the Teaching Quality of Physical Education Courses in Colleges and Universities[J].Applied Mathematics and Nonlinear Sciences,2025,10(1):DOI:10.2478/AMNS-2025-0057.

[10] Zhou B ,Liu S ,Song D , et al.Application of project-driven dual-teacher teaching model in the p ractical teaching of biomedicine[J].BMC Medical Education,2025,25(1):795-795.DOI:10.1186 /S12909-025-07016-X.