ISSN: 2414-1895 DOI: 10.6919/ICJE.202609_12(9).0013

Curriculum Reform and Practice of *JavaEE Framework Technology* under the AI-empowered and PBL-driven Model

Shiya Wang, Yunrui Peng, Tingting Zhang

Guangdong University of Science and Technology, Dongguan 523000, China

Abstract

With the rapid development of the digital economy and artificial intelligence, the industry is in urgent need of interdisciplinary software talents proficient in enterprise-level development frameworks and skilled in AI tool application. In view of the problems existing in the teaching of *JavaEE Framework Technology*, including outdated course content, teaching cases inconsistent with enterprise practice, and single teaching and assessment modes, this paper constructs a hybrid teaching model that integrates AI empowerment with PBL (Project-Based Learning) project-driven teaching. This teaching reform realizes the transformation of classroom teaching from knowledge indoctrination to ability cultivation and provides practical references for the curriculum reform of similar applied programming courses.

Keywords

JavaEE; AI-empowered; PBL.

1. Introduction

JavaEE Framework Technology is a core professional course for software engineering majors. However, current teaching practice shows that the course generally has prominent problems such as outdated teaching content, fragmented practical cases, monotonous teaching modes and unreasonable evaluation systems. Students generally lack practical operational abilities, and their knowledge reserve is seriously disconnected from industrial job requirements, which urgently calls for systematic teaching reform.

In response to the above problems, some scholars have promoted the updating of teaching content through industry-education integration and discipline competitions. The OBE[1-3] concept and BOPPPS[4] teaching method have been widely applied to the curriculum reform of various engineering courses, and many teachers have explored teaching reforms centering on AI empowerment and PBL[5-7] project-based teaching. Nevertheless, there are still few practical cases that fully embed AI auxiliary tools into the whole PBL teaching process for framework development courses such as JavaEE. Against this background, this paper takes *JavaEE Framework Technology* as the research object, comprehensively designs curriculum objectives, teaching content, classroom implementation procedures and supporting assessment schemes, and carries out pilot teaching practice in actual college classes.

2. Current Teaching Status and Reform Necessity of Course Teaching

Based on the student learning feedback accumulated from multiple rounds of teaching in our university, the current curriculum teaching mainly has the following four problems.

First, the updating of teaching content fails to keep pace with technological iteration. Teaching is mainly carried out based on textbooks with long compilation cycles, covering few enterprise

ISSN: 2414-1895 DOI: 10.6919/ICJE.202609_12(9).0013

mainstream technologies such as AI-assisted development. As a result, the technologies learned by students are disconnected from actual enterprise development requirements.

Second, practical training cases are fragmented and lack complete engineering scenarios. Most classroom cases are designed for the demonstration of single knowledge points and are independent of each other, lacking a complete business project that runs through the entire course.

Third, classroom teaching is dominated by teacher lectures with limited student participation. The teaching mode mainly relies on teacher demonstration followed by student repetitive operation on computers.

Fourth, the assessment system is unreasonable, emphasizing results while ignoring learning processes. The assessment mainly depends on final computer-based examinations which focus on conceptual memorization. The examination scores cannot truly reflect students' engineering practical capabilities, leading to the phenomenon of last-minute rote memorization and rapid knowledge forgetting after examinations.

In conclusion, the original teaching system of JavaEE Framework Technology can no longer meet the training requirements for applied software talents. It is necessary to adopt advanced teaching concepts and technical means to carry out systematic reforms, so as to further improve students' post adaptability for industrial jobs.

3. Implementation Path of Teaching Reform

The overall design of the teaching model relies on specific teaching organization processes for implementation. This section elaborates on the implementation path of the reform from three aspects: the PBL teaching process, the design of AI-empowered teaching links, and group collaborative training.

3.1 Implementation Process of PBL Project-Driven Teaching

The course teaching takes practical projects as the main organizational line. Referring to the mature practical experience of PBL combined with case teaching, each teaching module is implemented step by step through five stages: project introduction, task decomposition, independent exploration, iterative development, and summary and improvement.

The first stage is project introduction. Teachers put forward core module problems based on real engineering scenarios, such as how to solve data inconsistency caused by concurrent submission of user orders on an e-commerce platform, so as to stimulate students' learning initiative and sort out the engineering problems to be solved.

The second stage is task decomposition. Teachers and students jointly decompose the major module problems into several moderately refined development tasks, clarify the task list and the acceptance criteria for each task to ensure the orderly implementation of practical training.

The third stage is independent exploration. Students work in groups to consult materials, design schemes and write codes. They are allowed to apply AI tools for knowledge retrieval, sample code generation and development idea acquisition.

The fourth stage is summary and improvement. Each group displays phased achievements through project reports and mini defenses, with teachers and other groups participating in the evaluation.

3.2 Design of AI-Empowered Teaching Links

AI tools do not merely serve as simple in-class Q&A assistants, but are systematically integrated into the whole teaching chain covering pre-class, in-class and after-class stages.

At the pre-class stage, teachers build course-specific AI agents on online teaching platforms by importing course knowledge bases, project documents and frequently asked questions, enabling students to access professional learning materials at any time. The platform summarizes students' common confusions, helping teachers identify the key teaching focuses for classroom instruction.

ISSN: 2414-1895
DOI: 10.6919/ICJE.202609_12(9).0013

At the in-class stage, AI tools deeply participate in collaborative project development. During the development process, students can leverage AI programming assistants to generate sample codes, inquire about framework usage, and locate configuration errors. This allows students to focus more on principle understanding and scheme design rather than mechanically memorizing various configuration details.

At the after-class stage, AI agents provide personalized tutoring services. According to students' task completion performance, hierarchical exercises and expanded learning resources are recommended to students with different learning foundations. Teachers adopt AI technology to conduct batch static analysis on students' codes, summarize common errors, and grasp the overall learning situation of the class.

Meanwhile, clear AI usage specifications are formulated for the course. It defines permitted and prohibited scenarios for AI application. Students are required to independently interpret the core design logic and key codes during defenses. In addition, students must mark and record all AI-generated content. These measures reduce students' over-reliance on AI, prevent academic misconduct, and integrate AI ethics education into daily teaching.

3.3 Group Collaboration and Project Practical Training

The course adopts a practical training model of "group-based learning plus lightweight inter-group competition". Students form project teams of three members. Each team sets up three roles: architecture design, coding implementation, and testing & operation & maintenance. Roles are rotated on a regular basis to ensure every student can experience different stages of enterprise software development. Students with weak foundations can rely on group mutual assistance and personalized AI tutoring to lower the learning barrier. Evaluation criteria including functional completeness, code quality and innovation are defined for each module. Moderate competition is introduced to boost students' engagement. The training environment combines local development with cloud deployment. The Git collaboration platform is adopted to reproduce the real enterprise R&D workflow as much as possible.

4. Existing Problems and Improvement Directions in Teaching Practice

During teaching practice, several practical challenges have emerged. Some students rely excessively on AI tools, directly copying AI-generated code for assignments without understanding underlying logic, and are unable to explain how the code works in defense sessions. Going forward, it is necessary to refine process supervision measures and increase the assessment weight of independent coding tasks. Peer evaluation within groups is susceptible to personal bias, making it difficult to accurately quantify each member's contribution. We plan to conduct multi-source data cross-verification using code submission logs and task completion records. In addition, building and maintaining the course AI knowledge base entails a heavy workload, which places excessive burden on a single instructor. It is proposed to leverage collective lesson preparation within the teaching research office to divide responsibilities for maintaining teaching resources.

5. Conclusion

Against the backdrop of intelligent transformation in the software industry and the paradigm shift of software development brought by AI, this paper addresses various drawbacks in the conventional teaching of the course JavaEE Framework Technology. It constructs an integrated teaching model of "AI-enabled + PBL project-driven learning", systematically designs the course objectives, content, implementation path and evaluation system, and carries out pilot verification in real classroom settings.

First, the curriculum content is restructured with complete engineering projects closely related to campus scenarios. Four progressive modules are defined: engineering fundamentals, data persistence layer, business control layer, and comprehensive integration. This aligns knowledge acquisition with project development progress and meets the competency requirements of industrial positions.

Second, a five-stage PBL teaching process is established: project introduction, task decomposition, independent inquiry, iterative development, and summary & improvement. AI technology runs through the entire teaching cycle covering pre-class, in-class and after-class sessions. Meanwhile, norms for AI usage are formulated and training on critical review of AI outputs is carried out to avoid learning risks caused by students' over-reliance on AI.

Subsequent curriculum reform will keep pace with technological iterations in the software industry, deepen school-enterprise collaborative education, introduce more real project cases, improve AI teaching resources and assessment tools, and further explore the promotion path of this model within the software engineering curriculum group. It provides support for cultivating application-oriented software talents that meet industrial demands.

Acknowledgments

This work was supported by the Guangdong University of Science and Technology University-level Research Project, Project No.: GKY-2025KYYBK-17, and the University-level Teaching and Research Project, Project No.: GKZLGC2025021.

References

[1] Wang, S. Q., Zhang, M., & Miao, M. J. (2024). Teaching reform and practice of software development courses based on post-course-competition-certificate integration--Taking the course of Java EE enterprise application development as an example. Journal of Jilin Institute of Chemical Technology, 41(12), 28-33. https://doi.org/10.16039/j.cnki.cn22-1249.2024.12.007

[2] Fan, M. Q., & Liu, H. Y. (2023). Research on the innovative path of talent cultivation in the integration of industry and education in application-oriented undergraduate colleges-Based on the concept of OBE education. Modern Commerce and Trade Industry, 44(9), 101-104.

[3] Luo, L. H. (2024). Exploration of the optimization of the teaching mode of JavaEE framework technology course in the integration of industry and education. Modern Management Science & Engineering, 8(6).

[4] Shi, S. Y., & Li, Y. Q. (2026). Design and practice of a BOPPPS-based teaching model for junior high school artificial intelligence courses. Modern Management Science & Engineering, 8(3).

[5] Luo, L. H. (2024). Exploration and practice of OBE-based JavaEE framework technology course practice teaching reform under the integration of industry and education. Curriculum and Teaching Methodology, 7(8).

[6] Li, L., & Yang, Z. H. (n.d.). Research on teaching reform of software testing technology course oriented by the integration of "post-course-competition-certificate". Theory and Practice of Education.

[7] Pan, H. X. (2026). Research on the education model integrating labor education and economics-management major education in applied-oriented universities based on OBE-PBL. Jiangxi Education, (32), 23-25.