

The Practice of Teaching “Software Engineering” Course Assisted by AI from the Perspective of Grand Ideological and Political Education

Fangmin Xiong^{1*}, Yusen Cen¹

¹ School of Computer Science and Software, Zhaoqing University, Zhaoqing, China

Email Address

xiongfangmin@zqu.edu.cn (Fangmin Xiong)

*Correspondence: xiongfangmin@zqu.edu.cn

Received: 10 November 2025; **Accepted:** 6 December 2025; **Published:** 27 January 2026

Abstract:

To address the challenge of deeply integrating new engineering discipline construction with ideological and political education in courses, this paper explores an innovative path for empowering ideological and political education in the “Software Engineering” course with artificial intelligence technology. The research constructs an AI-driven blended teaching model, which introduces intelligent learning companions, builds a course knowledge graph, and implements a multi-dimensional evaluation system, achieving the intelligence and personalization of the teaching process. Practice has proved that this model can accurately identify students’ learning situations, dynamically push ideological and political resources, and quantify the effectiveness of education. It organically integrates value guidance such as craftsmanship spirit, teamwork, and the aspiration to serve the country through science and technology into professional links such as requirement analysis and system design, providing a reference model for the realization of a comprehensive education system that integrates professional education, value education, and personality education in science and engineering courses.

Keywords:

Curriculum-based Ideological and Political Education, Artificial Intelligence, Software Engineering, Blended Teaching

1. Introduction

With the in-depth advancement of the New Engineering construction and the “Curriculum Ideological and Political Education” philosophy, the teaching of science and engineering majors, such as Software Engineering, is progressively establishing a comprehensive educational system that integrates “professional education, value education, and personality education”. Blended learning, by virtue of its deep integration with information technology and its flexible and extensible nature, provides effective technical support and pedagogical innovation for achieving the

grand goal of ideological and political education. Furthermore, a new generation of information technology, represented by Artificial Intelligence, is propelling blended learning into an “intelligent and personalized” phase, opening new avenues for precise ideological and political education. This paper systematically explores the significance, implementation pathways, typical cases, and practical effectiveness of ideological and political education in the Software Engineering curriculum, aiming to provide a replicable teaching paradigm for the ideological and political education work in Software Engineering and related majors at universities.

2. The Significance of Implementing Ideological and Political Education in the Software Engineering Curriculum

2.1. Cultivating Craftsmanship Spirit and Professional Ethics

The software development process emphasizes code standards, quality control, and continuous iteration. Through the introduction of typical engineering cases, ideological and political education guides students to establish a rigorous, pragmatic, and meticulous craftsmanship spirit, fostering a sound sense of professional ethics [1].

2.2. Integrating Values with Professional Competence

Under the guidance of the “Curriculum Ideological and Political Education” philosophy, the teaching objectives have expanded from the singular dimension of knowledge transmission to a tripartite framework of knowledge, ability, and value. This encourages students to establish a correct worldview, outlook on life, and values while mastering the core skills of Software Engineering [2].

2.3. Adapting to the Talent Cultivation Needs of New Engineering

The New Engineering initiative emphasizes interdisciplinary integration, innovation and entrepreneurship capabilities, and social service awareness. Ideological and political education is a crucial pathway to achieving these goals, contributing to the enhancement of students’ innovative thinking and social adaptability [3].

3. Implementation Design of Ideological and Political Education in the Software Engineering Curriculum

3.1. Teaching Objective Design

To facilitate the precise implementation of the aforementioned layered objectives, this study introduces AI technology to construct a curriculum knowledge graph. This graph not only structurally presents the network of knowledge points, skill points, and ideological elements in Software Engineering but also dynamically identifies students’ knowledge gaps and value perception biases by analyzing their learning behavior data. This provides data support for subsequent personalized resource recommendation and precise ideological and political intervention. The specific layered teaching objectives are shown in Table 1 below.

Table 1. Layered Teaching Objectives for Curriculum Ideological and Political Education.

Level	Objective Content	Corresponding Ideological and Political Point
Knowledge Layer	Software requirements analysis, system design, coding implementation, testing and	Understand the national informatization development strategy and recognize the impact of technology on society

	deployment	
Ability Layer	Project management, teamwork, problem-solving	Cultivate teamwork spirit, craftsmanship spirit, and innovation ability
Literacy Layer	Ethical awareness, professional ethics, patriotism	Integrate socialist core values and national security concept

3.2. Construction of the Blended Teaching Model—Taking Data Flow Diagram (DFD) Teaching as an Example

3.2.1. Online Preview: AI-Assisted, Emphasizing Student Autonomous Learning

In addition to micro-videos and quizzes, the Rain Classroom platform incorporates the role of an “AI Smart Companion.” It can intelligently recommend supplementary learning materials and generate personalized questions based on students’ preview quiz results and reflection logs, guiding students to deeply explore the connection between technical details and ideological connotations.

The teacher releases videos related to Data Flow Diagrams (DFDs) on the Rain Classroom platform, providing a concise explanation of their basic components—external entities, processes, data stores, and data flows—supplemented by animated demonstrations and interactive exercises to help students build an intuitive understanding. With the assistance of the AI companion, students engage in a two-stage autonomous learning process: (1) watch the videos and complete embedded quizzes to consolidate symbol memorization; (2) read the accompanying materials and write a 200-word reflection log to explore the connection between technical details and ideological connotations. The teacher can monitor student progress and comprehension in real-time through the platform’s learning analytics data.

3.2.2. Offline Classroom: Project-Driven and Team-Based Collaboration

Using the “National Software Examination Registration System (Simplified Version)” as the project background, students in groups of 3-4 complete the entire process from requirements analysis to drawing multi-level DFDs. Specific implementation includes:

(1) Role Assignment: Designating roles such as requirements analyst, structural designer, detail drafter, and review coordinator, with clear responsibilities and regular rotation.

(2) Collaborative Drawing: Utilizing online tools like draw.io for real-time multi-person collaboration, with the teacher providing continuous guidance.

(3) Layered Drawing Training: Sequentially completing the context diagram, level 0, level 1, and level 2 DFDs to reinforce data balancing and logical consistency.

(4) Cross-Review Mechanism: Conducting in-group reviews after completing each level to unify naming, identify errors, and cultivate a rigorous attitude.

(5) Result Presentation and Reflection: Each group reports on their division of labor, collaboration process, and problem-solving strategies, while the teacher provides comments emphasizing the spirit of cooperation and responsibility.

The teacher can utilize AI analysis tools to perform real-time analysis of communication data and code submission records during the group collaboration process, promptly identifying collaboration obstacles or technical difficulties to provide more precise guidance.

3.2.3. Organic Integration of Ideological and Political Elements

Centered around the “National Software Examination Registration System” teaching project, the teacher achieves value guidance from three levels:

(1) Cognitive Level: Students understand the business process of the software examination by drawing the DFD, forming a preliminary understanding of the national professional qualification system.

(2) Behavioral Level: Students exercise team communication, division of review, and standardization skills in collaboration, shaping the craftsmanship spirit.

(3) Emotional Level: Students are guided to research the policies and development paths of the software examination, enhancing their sense of identity and pride in the national talent cultivation system.

3.2.4. Correspondence Between Key Teaching Links and Ideological Connotations

Table 2. Correspondence Between Key Teaching Links and Ideological Connotations.

Key Link	Specific Performance	Ideological Connotation
Role Assignment	Four types of roles are rotated with clear responsibilities	Embodying the spirit of collectivism and division of labor
Cross-Review	Reviewing each layer of the diagram to ensure data balancing and naming conventions	Cultivating the craftsmanship spirit of rigor, pragmatism, and quality first
Online Collaborative Tools	Using multi-person collaborative platforms for real-time synchronous modification	Experiencing the modern work model of information sharing and collaborative innovation
Feedback and Iteration	Comments from teachers and peers, with continuous revision and improvement	Forming a learning mechanism of continuous improvement and pursuit of excellence

3.2.5. Evaluation Mechanism Design

An evaluation system combining Outcome-Based Education (OBE) and the BOPPPS teaching model is adopted. This year, AI technology has been introduced. AI can automatically analyze students’ project code, design documents, and discussion contributions, generating quantitative evaluations and qualitative feedback from multiple dimensions such as technical normativeness, logical rigor, team contribution, and embodiment of ideological and political elements, thereby enhancing the efficiency and objectivity of the evaluation [4,5,6].

4. Other Typical Cases of Ideological and Political Education in the Software Engineering Curriculum

Case 1: Telling “China Stories”—The Entrepreneurial Journey of DeepSeek’s Founder

By introducing the growth trajectory of Liang Wenfeng, the founder of the domestic large model DeepSeek, and his team, students’ sense of national pride and sentiment of serving the country through science and technology are stimulated. The class organizes students to watch interview videos and conduct group discussions, focusing on technology transfer, resource optimization, and the significance to national informatization. Students then write a “Technological Pride” themed report, closely integrating their personal professional studies with national needs to achieve value internalization.

Case 2: Drawing a DFD for the “National Software Examination Registration System”

Students work in groups to complete the requirements analysis and layered DFD design for this system, experiencing the craftsmanship spirit through unified naming, data balancing reviews, and team collaboration. The project report focuses on showcasing the collaborative process and technical rigor, and the teacher strengthens students’ sense of responsibility and collaborative ability through the review process.

Case 3: China’s Industrial Software: The Path of Independent Innovation for the Domestic CAD Software “ZWSOFT Longteng”

In the “Software Architecture Design” chapter of the Software Engineering curriculum, the “ZWSOFT Longteng” product series from ZWSOFT, a leading domestic CAD company, is introduced as a typical case. The teacher demonstrates how the ZWSOFT team broke through foreign technology monopolies and, after more than 20 years of continuous iteration, ultimately created an integrated CAD/CAE/CAM platform with its own proprietary kernel. The teacher guides students to use AI tools to deepen the case analysis. For example, Natural Language Processing technology is used to perform sentiment analysis and topic mining on related news reports and technical literature, helping students more comprehensively understand the social value and industrial spirit behind the case. This case organically integrates patriotic education, the cultivation of innovative spirit, and professional knowledge of software architecture, effectively stimulating students’ national pride and professional identity, and achieving the synergistic enhancement of value guidance and capability cultivation [7].

This case reflects the national strategic value of core technology independence and controllability. When analyzing the technological evolution path of ZWSOFT Longteng from 2D CAD to 3D modeling, and then to cloud-native architecture, students can deeply appreciate the craftsmanship spirit of continuous innovation and overcoming difficulties. Through group discussions on “how to balance functional expansion and performance optimization in architectural design”, students not only master the modular design principles in software engineering but also enhance their sense of mission to serve the country through science and technology and to strengthen the nation through industry.

5. Conclusions

From the perspective of grand ideological and political education, the blended teaching of Software Engineering, through clear objectives, layered penetration, online-offline collaboration, and a data-driven evaluation mechanism, has effectively promoted the integration of technical education and value guidance. In the future, efforts should be made to further expand school-enterprise cooperation, enrich ideological and political case resources, and leverage intelligent teaching platforms to advance ideological and political education towards a more precise and personalized direction.

Conflicts of Interest

The author declares that there is no conflict of interest regarding the publication of this article.

Funding

This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

References

- [1] Ma, L.; Sun W.B.; Gao, J.L.; Lü, H.L.; Shan, D.H. Reform of Blended Classroom Teaching for Software Engineering Based on OKR-STEP. *Journal of Pingdingshan University*, 2025, 12(7), 23-31.
- [2] Guo, Y.Y.; Lou, L.F.; Lu, Y.H. Blended Teaching Focusing on “Deep Learning” and “Curriculum Ideological and Political Education”: A Case Study of the UML Course. *Computer Education*, 2024, 12, 59-64.
- [3] Yu, J.T.; Wu, Y.Q.; Wang, W.T.; Han, J.X. Design and Implementation of Ideological and Political Education Reform in the Software Engineering Curriculum. *Modern Educational Technology*, 2021, 33(4), 78-84.
- [4] Ma, R.X.; Yuan, X.; Zhao, L. School-Enterprise Integration, System Analysis and Design, Curriculum Ideological and Political Education, Blended Teaching, *Multi-Dimensional Learning Profile*. *Educational Research*, 2023, 45(3), 112-119.
- [5] Yu, J.T. et al. Design and Implementation of Ideological and Political Education Reform in the Software Engineering Curriculum. Beijing: Higher Education Press, 2022.
- [6] Yan, L.Y. Practical Exploration of Curriculum Ideological and Political Education Construction under the Blended Teaching Model: A Case Study of the Android Programming Course. *Software Engineering Education*, 2023, 8(2), 34-41.
- [7] Zhu, Z.Z. Exploration and Practice of Data-Based Blended Teaching. *Journal of Peking University (Software and Microelectronics)*, 2021, 39(11), 67-73.



© 2026 by the author(s); licensee International Technology and Science Publications (ITS), this work for open access publication is under the Creative Commons Attribution International License (CC BY 4.0). (<http://creativecommons.org/licenses/by/4.0/>)