

Construction of the “AI-Enabled + Four-Dimensional Linkage” Intelligent Teaching Model - A Case Study of “Software Engineering” Teaching

Ping Xu^{1*}, Ling Feng¹

¹ School of Information Science and Technology, Taishan University, Tai'an, China

Email Address

taianxp@163.com (Ping Xu)

*Correspondence: taianxp@163.com

Received: 1 August 2025; **Accepted:** 27 August 2025; **Published:** 19 September 2025

Abstract:

This paper focuses on enterprise-level application scenarios, based on knowledge graphs and the DeepSeek model, relying on the “Fanya AI” workbench, and adopts the four-dimensional linkage teaching strategy of “task-driven + team collaboration + hands-on simulation + documentation reporting”. It guides students to correctly describe and express the core issues of “project research and development” using software engineering knowledge, complete “project practice” through team collaboration, and write software development document reports, thus cultivating their ability to solve complex software engineering problems. Practice has proven that the construction of the “AI-enabled + four-dimensional linkage” intelligent teaching model meets the requirements of applied talent training goals and provides certain reference for the teaching reform of computer-related courses.

Keywords:

Enterprise-level Application Scenarios, AI-Enabled, Four-Dimensional Linkage, Intelligent Teaching Model

1. Introduction

In April 2018, the Ministry of Education of the People's Republic of China issued the Action Plan for Artificial Intelligence Innovation in Colleges and Universities, which clearly stated to accelerate the application of artificial intelligence (AI) technology in the field of higher education to build an intelligent education system and promote higher education reform [1]. Personalized and intelligent teaching services supported by AI can meet students' diverse learning needs, improve learning experience, and cultivate students' innovative thinking and practical abilities, thereby promoting their all-round development [2].

“Software Engineering” is a highly theoretical, practical, and comprehensive course [3]. It is characterized by abstract content, numerous knowledge points, and high practical requirements. How to build an AI-enabled curriculum teaching reform

paradigm oriented to the frontier of software engineering discipline development and the training needs of new software talents is an urgent problem to improve the teaching effectiveness of “Software Engineering” and the quality of professional talent training. Focusing on enterprise-level application scenarios, based on the DeepSeek model, relying on the “Fanya AI” workbench, this paper adopts the four-dimensional linkage teaching strategy to correctly guide students to describe and express a series of problems encountered in the process of project research and development using software engineering knowledge, and complete “project practice” through team collaboration, thereby improving students’ ability to solve complex software engineering problems.

2. Key Issues to Address in Curriculum Teaching Reform

2.1. Software engineering is highly theoretical and needs to be closely integrated with project practice

This teaching reform takes enterprise-level application scenarios as the carrier, selects typical cases, constructs knowledge graphs, and builds a three-dimensional training chain of “scenario cognition→knowledge modeling→practice verification”. Meanwhile, it innovates teaching strategies: task-driven + team collaboration + hands-on simulation + documentation reporting. The four-dimensional linkage realizes the cultivation of high-level abilities and the ability to solve complex software engineering problems.

2.2. To address the problems of fragmented knowledge and homogeneous learning paths, it is proposed to build a student-centered high-level intelligent teaching model through “knowledge graph + AI empowerment”

Driven by “enterprise-level application scenarios”, a knowledge graph is constructed based on existing curriculum resources. Making full use of “Fanya AI”, a hybrid three-dimensional teaching space from the perspective of knowledge graphs and two-way AI empowerment is created: relying on knowledge graphs to set preliminary knowledge to complete basic knowledge learning; using AI teaching assistants to comprehensively promote team task exploration and realize knowledge transfer and application; completing hands-on simulation through project grouping to build a student-centered high-level classroom.

2.3. To fully implement the fundamental task of fostering virtue through education, it is necessary to build an education system that deeply integrates value shaping, knowledge impartment, and ability cultivation

Combined with teaching content, ideological and political elements are flexibly integrated to guide students to possess the craftsman spirit, comprehensive and rigorous logical thinking ability, courage to explore innovative spirit, and the sense of mission to dedicate technology to national rejuvenation in the process of software project research and development, truly realizing the high unity of “educating people” and “cultivating talents”.

3. AI-Enabled Teaching Design

3.1. Analysis of Student Situation

“Software Engineering” is a core course for software engineering majors, offered in the 5th semester. Based on the study of “programming languages” in the first 4

semesters, students are relatively proficient in program development but have a shallow understanding of the software engineering thinking required for developing software systems. Specifically, first, they spend too much energy on later coding and do not pay enough attention to the early demand analysis and system design links, leading to numerous program errors and software that cannot truly meet user needs; second, students lack experience in developing large-scale software projects.

3.2. Teaching Objectives

Knowledge Objectives: Use software engineering principles, ideas, and relevant processes, guidelines, standards, and norms of software project research and development to select application-oriented software system development methods and technologies according to different scenarios, so as to solve specific software engineering problems; use software engineering support environments, structured and object-oriented methodologies to accurately express and describe core issues in key links such as user needs, system analysis and design, and system testing in the development process of application-oriented software systems, so as to propose effective problem solutions and analyze the feasibility and effectiveness of the solutions.

Ability Objectives: Independently complete the analysis and design of application-oriented software systems to be developed using software engineering principles, advanced software engineering technologies, methods, and their influencing factors according to different application scenarios and tasks to be completed in each stage of the software development life cycle, and try to write relevant software development document reports, so as to possess the ability to solve complex software engineering problems.

Literacy Objectives: Through ideological and political education, cultivate students' craftsman spirit, spirit of unity and cooperation, and sense of mission to dedicate technology to national rejuvenation; establish correct values and form good scientific literacy; develop the habit of self-directed learning, possess self-directed learning ability, and realize self-expansion and improvement.

3.3. Teaching Methods

Relying on the "Fanya AI" workbench, based on knowledge graphs and the DeepSeek model, the four-dimensional linkage teaching strategy of "task-driven + team collaboration + hands-on simulation + documentation reporting" is adopted.

4. Implementation of AI-Enabled Teaching

Driven by "real enterprise-level business scenarios", relying on knowledge graphs, AI teaching assistant workbenches, and the DeepSeek model, the traditional teaching model is reformed in an engineering way, and a hybrid three-dimensional teaching space from the perspective of knowledge graphs and two-way AI empowerment is fully built. The implementation process is mainly divided into two learning scenarios: "in-class" and "after-class". Through the connection and cooperation between teacher-led activities and student-centered activities, the four-dimensional linkage teaching strategy of "task-driven + team collaboration + hands-on simulation + documentation reporting" is adopted. Under the teachers' guidance, students find solutions to problems and realize the meaningful construction of learned knowledge.

4.1. Pre-Class Self-Directed Learning

Using the “Exclusive AI Courseware Assistant” module of the Fanya AI workbench, teachers provide software engineering knowledge points to generate PowerPoint (PPT) with one click; teachers release self-directed learning task lists in the “Course Study Group”; based on knowledge graphs, teachers push preliminary knowledge on the “Fanya Platform”, including chapter videos, assignments/exams, PPT, and auxiliary learning materials; students conduct self-directed learning, and “Fanya Platform AI Question Generation” automatically generates pre-class learning effect tests; teachers sort out the key and difficult points of “classroom teaching” based on self-directed learning effect feedback and core knowledge points.

Integration of ideological and political education: Based on knowledge graphs, organize students to independently explore and complete learning tasks, cultivate their ability to learn knowledge using “digital + intelligent” technologies, develop the habit of “self-directed learning”, and further realize the expansion and improvement of their abilities.

4.2. In-Class Participatory Learning

Basic Task: Express software engineering problems using software engineering principles, ideas, and methods, and identify and judge key links of complex software engineering problems.

Illustration: Identifying Actors in Use Case Modeling for Functional Requirements Analysis of Mobile E-commerce Platform Development

Teaching implementation: Based on the DeepSeek model, teachers provide thinking paths, and student teams actively explore and discuss through cooperation. Thinking question (identifying actors): Whose business needs does the mobile e-commerce platform mainly meet? Team exploration + answer analysis: The answer is given based on the reference answers provided by the DeepSeek model, software development experience, and team cooperative exploration. The actors involved in the shopping mall platform include customers, ordinary administrators, system administrators, and payment systems.

Advanced Task: Through thematic discussions, quick answers, selection, voting, and other links, actively interact with teachers and students using statements, modeling, and charts to clearly and accurately express personal views, and gradually understand the differences in communication with the public.

Illustration: Use Case Elicitation for Functional Requirements Analysis of Mobile E-commerce Platform Development (Teacher-Led Practice: Layered Decomposition and Divide-and-Conquer Approach)

Teaching implementation: Based on the DeepSeek model, students conduct brainstorming. On the “Chaoxing Fanya” platform, in-class interactions (quick answers) are set: What are the use cases related to customers, ordinary administrators, and system administrators? This process focuses on cultivating students’ analytical ability and abstract thinking ability. Answer analysis: Use cases related to customers include logging into the system, querying products, browsing announcements, managing orders, managing shopping carts, setting up memberships, and logging out; use cases related to ordinary administrators include managing product categories, managing products, and processing orders; use cases related to system administrators include managing members, managing ordinary administrators, managing announcements, backing up data, and restoring data.

Analysis of Key and Difficult Points: Complete the analysis, design, and other related work of application-oriented software systems using advanced software engineering tools according to the tasks to be completed in each stage of the software development life cycle, thus improving students' ability to solve complex software engineering problems.

Illustration: Use Case Refinement and Model Construction in Functional Requirements Analysis of Mobile E-commerce Platform Development (Heuristic Explanation + Guided Practice)

Teaching implementation (use case model construction): Guide students to gradually clarify the relationships among use cases. For example, the "manage orders" use case includes sub-use cases such as "generate orders", "pay orders", "query orders", and "cancel orders"; the extended use cases of the "settlement center" include "generate orders", "pay orders", and "manage shopping carts". This process trains and cultivates students' ability to analyze problems, express professionally, refine modeling step by step, and develop comprehensive and rigorous logical thinking.

High-Level Task: Independently complete feasibility studies, requirement analysis, system design and modeling, software testing, and other related work of application-oriented software systems according to different application scenarios, and complete hands-on simulation.

The "Software Engineering" course arranges "team practical projects" for software development. Students form teams to formulate project themes, complete phased research and development through team collaboration and role-playing according to the course progress, and share practical design results in teams. Teachers monitor the project progress of each team throughout the process and provide effective guidance.

Integration of ideological and political education: By telling the story of the analysis and design process of the HarmonyOS system, despite the new global competitive ecology and harsh political environment, Huawei fully opened its core technologies and worked with developers to drive the vigorous development of the full-scenario smart ecosystem. This makes us understand that "no one can extinguish the stars all over the sky. Every software developer is a spark converging for China's IT sci-tech innovation!"

4.3. After-Class Expansion and Improvement

The AI teaching assistant pushes cutting-edge knowledge, feeding back the latest academic and scientific research achievements into teaching; students complete personalized learning in the expansion module and pass the value-added application ability assessment, namely, writing software project research and development reports through team collaboration.

Integration of ideological and political education: "Virtue, erudition, truth-seeking, and innovation" are the accumulation of Taishan University's campus culture and the condensation of campus spirit. As students of Taishan University, you should not only learn professional knowledge well but also dare to innovate, possess the spirit of the times of not being willing to fall behind, pursuing progress, and actively exploring, and shoulder the mission of dedicating technology to national rejuvenation!

5. Implementation Effects

Software engineering curriculum teaching reform: Leveraging knowledge graphs with AI teaching assistants and the DeepSeek model, we integrate online-offline resources to innovate teaching methods. Significant outcomes include enhanced academic ethos, strengthened self-directed learning capabilities, high classroom engagement, and excellence in team-based collaborative projects. Growing numbers of students actively participate in computer application competitions, achieving outstanding accolades.

6. Conclusions

This teaching reform emphasizes the deep integration of theory and practice and innovates the “problem-solving” methods: following the actual needs of enterprise positions, selecting typical application cases, constructing knowledge graphs, relying on “Fanya AI”, and building a three-dimensional training chain of “scenario cognition→knowledge modeling→practice verification”; adopting the progressive three-stage learning model of “pre-class self-study + in-class discussion + after-class expansion” centered on students through the four-dimensional linkage teaching strategy of “task-driven + team collaboration + hands-on simulation + documentation reporting”, so as to train students’ practical innovation ability and improve their ability to solve complex software engineering problems.

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] Li, H.; Huang, L.; Wu, X.Z. Exploration of the Path of Artificial Intelligence Empowering Higher Education: Practice and Enlightenment from Chongqing University. *Journal of Architectural Education in Institutions of Higher Learning*, 2024, 26(10), 1-9.
- [2] Chen, L.P.; Chen, J.M.; Zhang, S.H. Reform and Exploration of Mixed Teaching in the Provincial First-Class Undergraduate Course “Software Engineering”. *Information & Computer*, 2024, 18, 30-32.
- [3] Mao, X.J.; Sun, Y.C.; Chu, H. et al. The Idea and Achievements of “101 Plan” Software Engineering Course Construction. *Computer Education*, 2023, 11, 29-33.



© 2025 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/>)