Exploring Values-Enhanced Development in a Software Requirements Engineering Course

Wenrui Li*, Weiguo Li, Wenjie Xiao
School of Information Engineering, Nanjing Xiaozhuang University, Nanjing, China
**Corresponding Author*

Abstract: Software requirements engineering lies between social demands and technical implementation. In the present course, several problems occur, including inconsistent lectures and value building, unpractical cases, lack of absorption pathways and insufficient evaluation. In this paper, we present a generative AI-based framework to reform the software requirements engineering course. It includes the entire teaching-learning process, from pre-class preparation to class participation and after-class reflection. During the teaching process, we obtain empirical data through the use of "Smart Senior Care system". The results indicate that this approach can enable students to pay attention to the consideration of values in requirement analysis. The key aspect of this method is to improve the students' ability to make decisions concerning the coordination of stakeholders' interests and value trade-offs.

Keywords: Software Requirements Engineering Course; Generative AI; Value-Oriented Development; Value Trade-Offs

1. Introduction

Higher education, as the primary arena for talent cultivation, takes the dual responsibility of not only imparting professional knowledge and developing practical competencies but also fostering sound values in students. In the present situation of rapid technological changes and industrial upgrading, the assessment of talents focuses on technical skills and includes various aspects such as professional competence, occupational honesty and social responsibility [1]. Under such circumstances, value-oriented development plays an important role in educational innovation.

Generative AI, having strong abilities in content creation and interaction, significantly alters the method of knowledge production and spread. It can produce text, program codes, teaching examples and so on according to natural language orders, showing great adaptability and innovative potential. In education, it improves teaching efficiency by means of individualized tutoring, intelligent questions and answers and others, and overcomes the one-sidedness in the transfer of knowledge, creating an interactive environment where both teachers and students can study together [2,3].

In the field of software engineering, generative AI is profoundly transforming the methods of knowledge delivery and value formation, offering unprecedented opportunities to empower the reform of instructional models in courses [4,5]. The software requirements engineering is an initial stage in the software development process. The software requirements engineering course provides systematic knowledge about the collection, analysis, modeling and management of user requirements [6]. It can enhance the students' ability to comprehend the needs of different users [7].

To accomplish this, real-life social and engineering issues are involved in the teaching, so that the students can realize the relationships among technical decisions, public benefits, data security and ethical norms. Based on this basis, the course encourages the students to foster a sense of social responsibility, comply with laws and regulations and maintain professional honesty. Nevertheless, some difficulties still affect the complete achievement of the above educational goals.

The value-related contents are only linked to the technical instructions instead of being integrated into them. The teachers usually include these contents in the course without integrating them into the main practice of the subject, which leads to a persistent difference between the transmission of knowledge and the establishment of values [8,9].

Teaching cases usually do not relate to real social problems, which reduces students' interest. Students cannot realize the relationship

between their technical education and national concerns or social requirements.

The course focuses on developing tool-based skills. It offers few systematic chances for students to deal with ethical dilemmas and value conflicts in practical situations, such as those concerning privacy and function, or efficiency and fairness.

Evaluation emphasizes solely on the technical accuracy and model correctness which is not enough. It cannot assess the students' developing social awareness or their judgment in ethical and difficult situations.

Generative AI are altering the way knowledge is imparted and values are developed in software requirements engineering. These changes create unprecedented pedagogical opportunities for course reform. Thus, we present AI-enhanced value development in software requirements engineering course. We should use AI to solve the long-existing difficulties in incorporating values into conventional teaching. This method finally results in a subtle and significant educational influence. By embedding value-oriented guidance throughout the entire process of professional instruction, this approach seeks to equip students with both solid disciplinary expertise and the humanistic sensibility and ethical judgment demanded by the contemporary era, thereby nurturing well-rounded, high-caliber professionals for the future.

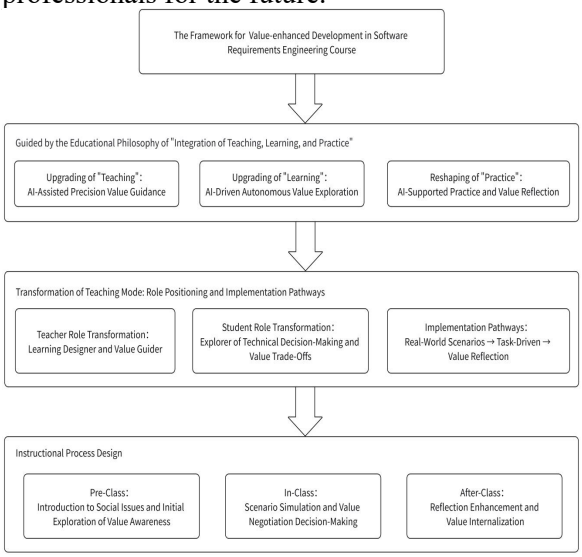

Figure 1. The Framework of Value-Enhanced Development in Software Requirements Engineering Course

2. The Framework of Value-Enhanced Development in Software Requirements Engineering Course

Tao Xingzhi's pedagogical maxim of "Integration of Teaching, Learning, and Practice" [10] asserts that these three elements form an inseparable triad. "Practice", in the form of authentic problem-solving, sits at the core as the most effective vehicle for both instruction and learning.

Following this philosophy, we propose a framework that integrates educational philosophy of "Integration of Teaching, Learning, and Practice" with generative AI tools to support value development, shown in Figure 1. The framework consists of three layers. At the top layer, the basic philosophy directs three enhanced versions by AI: AI-assisted precision value guidance in teaching, AI-driven autonomous value exploration in learning, and AI-supported practice and value reflection. The middle layer describes the change in teaching mode, changing the teacher's position into a the designer and value guider of learning process, and the student's role as the explorer of technical decisions and value conflicts, with the execution steps following a series of real-life situations, task-driven activities and value reflection. The bottom layer provides the instructional procedure through three stages: pre-class activities for introducing social problems and arousing value awareness, in-class activities including scenario simulations and value negotiation decision-making, and after-class activities focusing on enhancing value reflection and internalization.

3. Case-Based Teaching Process for Value Education in Software Requirements Engineering

An AI-enhanced value development case study in software requirements engineering can be instructed by three planned teaching stages pre-class, in-class and after-class. The teachers select "Smart Elderly Care system" as an teaching case. In the following, we explain the specific stages pre-class, in-class and after-class carried out with AI. These stages help the students to gain both technical knowledge and social responsibility.

3.1 Pre-Class: Introducing Social Issues and Building Initial Value Awareness

As a pre-class activity, students independently investigate the current situation of community elderly care in China. They examine typical user groups and key service gaps. User groups

include elderly living alone, those with chronic conditions, and busy family members. The instructors pose guiding questions. For example, "What personal information might the system collect from the elderly?" and "What difficulties might users of different cultural backgrounds or physical abilities encounter?", which prompts encourage students to consider data security and privacy. The students also think about design inclusiveness and humanistic care from the start. They can identify major stakeholders and their basic needs. They compile their findings into a brief report. This report prepares them for in-class requirements modeling, design, and value negotiation.

3.2 In-Class Implementation: Scenario-Driven Value Negotiation Practice

The teacher arranges a step-by-step teaching procedure including scenario simulation and negotiation decision-making. This enables students to combine their technical abilities with value consideration naturally.

Stage 1: scenario simulation – understanding requirements in context.

The teacher makes typical smart elderly care scenarios by utilizing generative AI. These scenarios illustrate the business processes from different stakeholders. They involve the opinions of the elderly people, their relatives and the community managers. The teacher intentionally emphasizes the important value conflicts. For example, during the demonstration of the health monitoring, the teacher mentions the privacy issues in the data collection. The system may send the real-time location, health information and warnings to the relatives. The teacher then asks: Is it helping or controlled? What is the distinction between safety and independence? How can we balance the relatives' need to know with the elderly's right to privacy?

Stage 2: negotiation and decision-making – requirements decisions through multi-stakeholder trade-offs.

Students realize the value conflicts. The teacher guides them from the recognition of problems to the selection of solutions. Emphasis is laid on value negotiations and technical decisions with certain restrictions. Students are divided into teams. Each team represents a different interest party. They take part in discussion sessions.

The teacher describes the negotiation task. For example, "Formulate an effective scheme for health observation within the limited budget and technical limitations." The teacher also helps the students to establish some rules for a polite and fair discussion. Sometimes, there may be a delay in discussing which is more important, privacy or data exchange. The teacher puts forward some questions, "Do our viewpoints require us to choose between privacy and data exchange? Can we build a system which is both safe and feasible?"

During the decision-making process, the teacher prompts students to consider not only the practicability but also the social responsibility. The teacher suggests that students should ask themselves: "What influence will this solution have on the people in need? Have we given up something and is it justified from an ethical standpoint?"

Students go from solving the problem to proposing solutions. They write down the initial requirements. In this process, they learn to cope with real socio-technical situations. They take part in reasonable discussions and make responsible decisions. They jointly find feasible technical solutions which improve their communication, judgment and decision-making abilities in complex scenarios.

3.3 After-Class: Reflection Enhancement and Value Internalization

In the after-class stage, teachers guide students into the reflection phase, reviewing the key junctures in the formation of the technical solution, with particular attention to stages where there were obvious disagreements. Through questioning such as: "When finalizing the scope of health data sharing, how did we gradually shift from a discussion of the conflict between privacy protection and data sharing to simultaneously achieving both safeguards through systematic design?" The questions help students examine how values influenced the formation of the technical solution and document their reflection report.

4. Teaching Effectiveness Analysis and Diverse Evaluation Methods

We employ a diverse set of evaluation methods to assess the effect of AI-enhanced value development, including process-oriented assessment, outcome-oriented assessment, and multi- party evaluation, as illustrated in Table 1. Its purpose is to improve both the technical ability and the value awareness.

Process-oriented assessment considers the

student's involvement. It evaluates the analytical depth and the reasoning for decisions as well. We examine these aspects in the pre-class investigation, the in-class discussion and the after-class reflection. The important points covered are the comprehension of user needs. The quality of the requirement specification is also assessed. We investigate the consideration of privacy, fairness and social influence during the design choices. This promotes the students to take into account the judgments throughout the learning course.

Outcome-oriented assessment evaluates the outcome documents which include reports, presentations and reflection records. Our standards are based not only on the completeness and accuracy of outcome documents, but also on the corrective actions and the assignment of responsibility.

A third assessment technique is employed by multiple parties, including the teacher's assessment, the student's self-evaluation and the peer evaluation. It improves the fairness and gives formative remarks, which help the students to understand their strengths. At the same time, it shows the weaknesses in both the practical and theoretical fields. This encourages them to enhance themselves further.

Table 1. Diverse Evaluation methods for Curricular Implementation Outcomes

Evaluation Dimension	Core Focus	Evaluation Content and Assessment Criteria	Assessment Instruments / Evidence
Process - Oriented Assessment	Emphasis on the full learning trajectory, prioritizing value deliberation and practical engagement	1. Participation degree, analytical intensity and decision-making reasons in pre-class investigation, in-class discussion and post-class reflection. 2. Understanding of users' demands and Standardization of requirement specification during analysis 3. Taking into account privacy protection, equity and social effects in the solutions' comparison.	Documentation of classroom performance, research portfolios, interactive discussions and cognitive traces
Outcome - Oriented Assessment	Emphasis on deliverable quality, prioritizing comprehensive competence and ethical awareness	1. Completeness and soundness of requirements models and documentation 2. Assessment of technical proposals' potential impact on vulnerable populations 3. Formulation of actionable mitigating measures 4. Proactive reflection on responsibility allocation	Requirements analysis reports, solution presentations, and individual reflective portfolios
Multi - party Assessment	Enhancement of evaluative objectivity, reinforcement of self - awareness and iterative improvement	1. Instructor evaluation of professional competence and value literacy 2. Student self - assessment (reflection on strengths and growth areas) 3. Peer evaluation (collaboration, critical thinking, articulation, etc.)	Instructor evaluation scales, self-assessment sheets, peer review lists and comprehensive multi-dimensional feedback reports

5. Empirical Data and Teaching Outcomes

In order to evaluate the teaching effectiveness of value development supported by generative AI, we conducted the experiment in the software requirements engineering course at Nanjing Xiaozhuang University. The investigation was performed during the second semester of the academic year 2025-2026. The course lasted for 48 instructional hours and is taken by 45 undergraduate majoring in Software Engineering of the 2023-grade.

We chose "Smart Elderly Care System" as teaching case. The whole teaching plan contained value-based materials in each significant phase of requirements engineering, for example, elicitation, modeling, negotiation and decision-making. We used generative AI tools such as DeepSeek, Doubao, Kimi and Copilot. The experimental results are presented in Table 2.

At the pre-class inquiry stage, the mean number of questions per student in the experimental group 3.2 was greater than that in the control group 1.8, resulting in an increase of 1.4. The percentage of value-laden questions also increased from 23% to 41%, and the frequency of submissions containing ethical or social

responsibility terms rose from 44% to 68%. These results indicate that AI-assisted teaching can promote more profound cognitive involvement and guide students to consider the ethical and social aspects of the subject.

During the in-class negotiation phase, the experimental group showed a significantly better result, with 86.7% of students recognizing at least one value conflict, compared to 52.1% in the control group. Additionally, the proportion of students able to explain the trade-off reasoning rose from 54.2% to 83.3%. These findings suggest that the integration of AI can promote the improvement of critical thinking ability and enhance the students' capability to express complicated decision-making procedures in multi-factorial situations.

In the after-class reflection stage, the number of students supporting the view that "technology contributes to dignity and autonomy" was 67.5% in the experimental group, while it was 31.2% in the control group, an increase of +36.3%, which was the highest improvement among all the investigated items. This result also shows that AI not only enhances cognitive abilities but improves the understanding of technical ethics.

According to the results, AI-enhanced learning plays a beneficial role throughout the three teaching stages, and the greatest impacts are found in value conflict recognition, ethical judgment and reflective thought. These empirical evidences provide strong support for the integration of AI in teaching and highlight the necessity of developing future instructional systems to make full use of the potential of AI in improving higher-order cognitive abilities and ethical standards of the learners.

Table 2. Summary of Empirical Outcomes from Curricular Implementation

Evaluation Dimension	Indicator	Experimental Group (with AI)	Control Group (without AI)	Gain
Pre - Class Inquiry	Mean questions posed per student	3.2	1.8	+1.4
	Questions with value content (%)	41%	23%	+18%
	Submissions citing ethical/social responsibility terms (%)	68%	44%	+24%
In - Class Negotiation	Students identifying ≥1 value conflict (%)	86.7%	52.1%	+34.6%
Post - Class Reflection	Students articulating trade-off reasoning (%)	83.3%	54.2%	+29.1%
	Students affirming "technology serves dignity and autonomy" (%)	67.5%	31.2%	+36.3%

6. Conclusion

AI-assisted teaching has brought about positive impacts in three main aspects: value conflict recognition, ethical judgment and reflective thinking. In the future, the teaching group will increase the variety of project topics, investigate the possibilities of AI in cooperative ideological and political education across multiple courses, and establish intelligent ethical assessment devices. Such actions will constantly facilitate the combination of technology and comprehensive education, finally offering a practicable and expandable model for the transformation of software engineering education in the intelligent age.

References

[1] Ministry of Education of the People's Republic of China. Guiding Outline for Curriculum Ideological and Political Construction in Colleges and Universities, 2020.

[2] Ni Guo, Fangyuan He, Zhengkun Hu, et al. Analysis of Opportunities and Challenges in Driving Educational Transformation through Generative Artificial Intelligence. Journal of Higher Education Teaching, 2025, 2(4): 187-193.

[3] Uday Mitta, Siva Sai, Vinay Chamola, et al. A comprehensive review on generative AI for education. IEEE Access, 2024, 12: 142733-142759.

[4] Xiaofei Xu, Xiang He. Generative Artificial Intelligence Empowering Software Engineering Education and Student Competency Enhancement. Computer Education, 2025(8): 1-8.

[5] Ahmed Kharrufa, Sami Alghamdi, Abeer Aziz, et al. LLMs integration in software engineering team projects: Roles, impact, and a pedagogical design space for AI tools in computing education. ACM Transactions on Computing Education, 2026, 26(2): 1-27.

[6] Weiguo Li, Wenrui Li, Wenjie Xiao. Exploration of Teaching Reform in

Software Engineering Experimental Courses Integrating CAD Software Development Technology. Journal of Higher Education Teaching, 2026, 3(1): 274-280.

[7] Yoram Reich, Emilia Lavi, Kuti Shoham. What is the value of design theory? Proceedings of the Design Society, 2025, 5: 3391-3400.

[8] Neil Maiden. Generative rules for more creative thinking about requirements. IEEE Software, 2025, 42(3): 7-12.

[9] Stefan Kapferer, Olaf Zimmermann, Mirko Stocker. Value-driven analysis and design: Applying domain-driven practices in ethical software engineering. Proceedings of the 29th European Conference on Pattern Languages of Programs, People, and Practices. 2024: 1-15.

[10] Xingzhi Tao. The Complete Works of Tao Xingzhi. Central China Normal University Press, 2024.

