# Exploration of Software Engineering Experimental Teaching Methods for Cultivating Applied Talents

**Yong Lu[1,a], Shaocong Cheng[2,b], Shaoqi Cheng[3,c]**

[1]Minzu University of China, 100081, Beijing, China

[2]Beijing Union University, 100101, Beijing, China

[3]Kede College of Capital Normal University, 100101, Beijing, China

[a]2006153@muc.edu.cn, [b]2756847004@qq.com, [c]3473569417@qq.com

**Keywords:** Cultivation of applied talents; Software engineering; Experimental teaching

**Abstract:** Amidst the swift advancements in information technology, software technology stands out as a pivotal domain, increasingly serving as a vital catalyst for societal progress and economic growth. Given this backdrop, the market's appetite for software engineering professionals with practical skills is escalating at a consistent rate. Consequently, nurturing such talents has emerged as a pivotal topic in the educational sphere, attracting widespread scrutiny and deliberations. However, software engineering majors often encounter challenges and issues in educational practices. Traditional educational approaches often concentrate solely on disseminating theoretical knowledge, overlooking the fostering of students' practical and innovative capabilities. This gives rise to a predicament where numerous students struggle to promptly adapt to market demands and lack the proficiency to tackle real-world problems upon graduation. Therefore, it's imperative for educators to stay abreast of the times, proactively embracing fresh perspectives, and exploring innovative software talent training paradigms. By delving into novel experimental teaching methods and strategies in software engineering, we can offer invaluable insights and value for cultivating innovative and entrepreneurial talents in this domain.

## 1. Introduction

In today's Internet era, the rapid growth of information technology has promoted profound changes in various fields. As the key support, software technology plays an increasingly important role [1]. As the main discipline for cultivating software technology talents, the rise and growth of software engineering not only provide a large number of professional talents for society, but also promote the continuous innovation and application of software technology [2]. However, with the continuous expansion of the market and technological updates, the demand for software engineering applied talents in society is also constantly increasing, which poses new challenges to software engineering education [3]. Currently, software engineering education is facing unprecedented growth opportunities and severe challenges [4]. On the one hand, with the continuous emergence of new technologies and the continuous expansion of application fields, software engineering education needs to constantly update teaching content and methods to adapt to market changes and demands. On the other hand, with the transformation of educational concepts and innovation in talent cultivation models, software engineering education also needs to actively explore new teaching models and approaches to improve students' practical and innovative abilities [5].

As a pivotal aspect in the establishment of modern engineering disciplines, software engineering demands a profound transformation in education, promoting innovative approaches in talent cultivation to generate more innovative and applicable experts capable of tackling intricate engineering issues in society [6]. Among the key elements in software engineering education, experimental teaching holds a prominent position. It allows students to engage actively in project development, acquire hands-on experience, grasp the fundamentals of software growth, and enhance their proficiency in solving practical challenges [7]. Therefore, exploring and

DOI: 10.25236/icssem.2024.078

implementing effective software engineering experimental teaching methodologies are of utmost importance in nurturing high-caliber and technologically adept professionals. Currently, however, software engineering experimental teaching faces several challenges and obstacles. Firstly, traditional experimental teaching methods often prioritize the dissemination of theoretical knowledge, overlooking the cultivation of students' practical and innovative skills [8]. Secondly, the content of experimental teaching frequently lags behind the demands of the market, hampering students' ability to adapt to market shifts and requirements. Additionally, limited experimental teaching resources and inadequate experimental teaching environments also restrict the advancement of experimental teaching.

In response to these issues and challenges, we need to actively explore new software engineering experimental teaching methods and measures. Firstly, it is necessary to transform the concept of experimental teaching from traditional emphasis on knowledge transmission to emphasis on ability cultivation. Through the design and implementation of experimental projects, cultivate students' practical, innovative, and teamwork abilities. Secondly, we need to strengthen the connection between experimental teaching and the market, introduce practical engineering projects, and enable students to understand market demands and technological trends in practice. At the same time, we also need to improve experimental teaching resources, optimize the experimental teaching environment, and provide strong guarantees for experimental teaching. In addition, innovation in software engineering experimental teaching also needs to keep up with the times and closely integrate with the forefront of technological growth. With the widespread application of technologies such as cloud computing, big data, and artificial intelligence, software engineering experimental teaching should also incorporate these emerging technologies, so that students can not only master basic skills but also understand and apply the latest technological achievements.

## 2. The Principles and Current Situation of Cultivating Applied Talents in Software Engineering

### 2.1. Principles

The principles of cultivating applied talents in software engineering are the basic guidelines formed in educational practice to guide talent cultivation work. These principles are of great significance for improving student employment rates, ensuring the effectiveness of education, and ensuring the timeliness of talent cultivation [9]. Firstly, the principle of goal consistency emphasizes the high degree of unity between the cultivation of software engineering applied talents and the achievement of engineering professional certification by students in terms of goals [10]. The core of this principle is that whether it is the talent cultivation process or the engineering professional certification process, the ultimate goal should be to improve the employment rate of students. This requires teachers to closely focus on the core goal of improving students' employability when formulating teaching plans, designing experimental courses, and conducting practical activities, ensuring the rational allocation of educational resources and the optimization of the educational process. At the same time, students should also clarify their learning goals, actively participate in various activities to enhance their employability, strive to obtain engineering professional certification, and lay a solid foundation for their future career growth.

Furthermore, the principle of effectiveness demands that teachers be consistently steered by the enhancement of student employment rates throughout the student cultivation process, thereby guaranteeing the efficacy of teaching models. This implies that teachers must not solely concentrate on imparting knowledge, but also cultivate students' practical and innovative abilities. This ensures that students accumulate substantial practical experience and possess the capacity to tackle real-world problems during their academic tenure. Additionally, teachers ought to diligently monitor industrial trends and market demands, promptly adapting teaching content and methods to align educational offerings with market requirements, thus bolstering students' employment competitiveness. The principle of applicability necessitates teachers to thoroughly consider students' actual circumstances and needs during the teaching process. Teachers should maintain effective communication with students, comprehend their actual proficiency and interests, devise targeted teaching content, and select suitable teaching methods. This not onlykindles students' interest and enthusiasm in learning but also ensures that the teaching content aligns with their genuine needs, thereby enhancing teaching effectiveness. Simultaneously, teachers should attend to student learning feedback and evaluation, promptly adjusting teaching strategies, and ensuring continuous improvement and optimization of the educational process.

## 2.2. Current Situation

In the current landscape of software engineering education, several crucial issues persist that are deserving of immediate attention. Notably, numerous universities tend to overly concentrate on dispensing knowledge during their educational processes, which often results in students passively absorbing information. This approach often overlooks and suppresses students' initiative and creativity, standing in stark contrast to the aim of fostering high-caliber and innovative individuals. Consequently, when confronted with practical challenges, students often lack the capacity for independent thinking and problem-solving. Given the exponential growth of information technology, the need for skilled software engineers is escalating, particularly those possessing innovative mindsets and abilities. This scarcity has emerged as a significant obstacle hindering the progress of artificial intelligence technology. Hence, the need to reform the approach to teaching software engineering courses is paramount and urgent.

Although software growth is currently a popular employment field with huge social demand, the market does not lack low-end developers with certain work experience. This indicates that the demand for software engineering talents in the market is shifting from quantity to quality, with a greater emphasis on the professional competence and innovation ability of talents. Therefore, when cultivating software engineering talents, universities must pay attention to enhancing students' professional literacy and innovation ability to meet market demands. When recruiting fresh graduates, companies often first assess the professional competence of applicants. Table 1 shows the indicators and contents of the professional competence assessment for fresh graduates in recruitment. Only after passing the professional competence interview can applicants enter an interview focused on professional competence.

Table 1 Professional literacy examination indicators and content for recruiting fresh graduates

| Examination items | Content |
|---|---|
| Teamwork | Understand the roles and achievements of students in club activities and competitions during their school years |
| Communication skills | Test students' reaction speed, language expression, and written expression ability |
| Pressure bearing capacity | Understand the personality traits, values, and resilience of students |
| Personal Integrity | Test and evaluate the personality of job applicants |

## 3. Exploration of Teaching Methods for Software Engineering Experiments

### 3.1. Blended Learning Mode

The blended learning model has important application value in cultivating applied talents in software engineering. This teaching model aims to combine the advantages of online self-directed learning with offline classroom teaching and practical teaching, comprehensively enhancing students' self-learning ability, practical ability, and application ability (as shown in Figure 1). In blended learning mode, teachers need to have a deep understanding of the knowledge points of software engineering courses. Through such a deep understanding, teachers can design teaching content and methods more targeted, helping students build a complete knowledge system. Online self-directed learning is an important component of blended learning mode. Through online platforms, students can access learning resources anytime and anywhere for self-directed learning.

This section emphasizes the initiative and autonomy of students, encouraging them to deepen their understanding of knowledge points through independent thinking and problem-solving. At the same time, online platforms can also provide learning progress tracking and feedback mechanisms, helping students understand their learning status and adjust learning strategies in a timely manner.

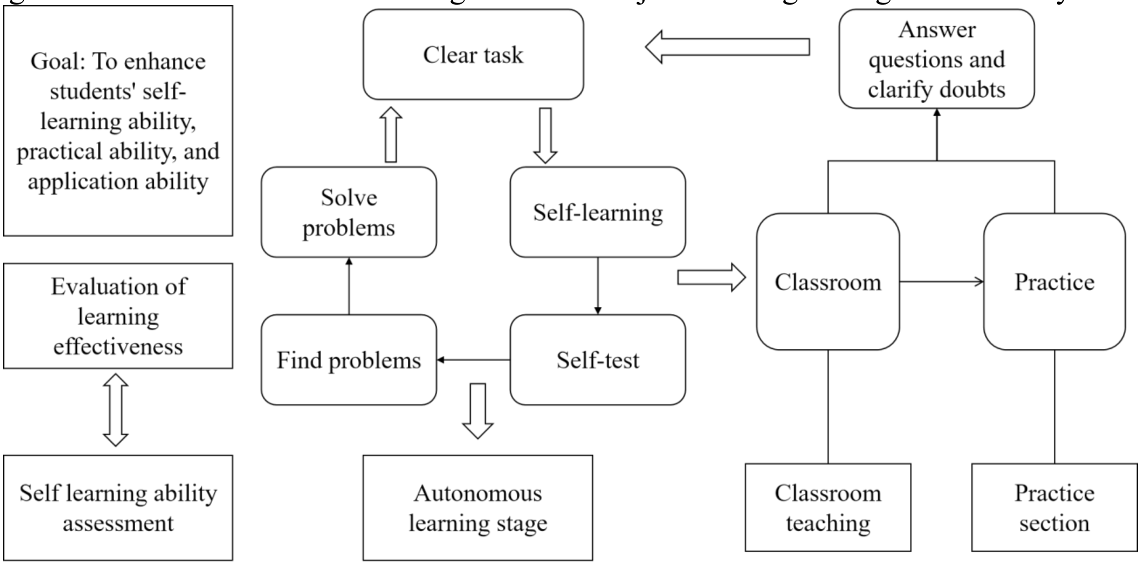
Goal: To enhance students' self-learning ability, practical ability, and application ability
Evaluation of learning effectiveness
Self learning ability assessment
Clear task
Solve problems
Self-learning
Find problems
Self-test
Autonomous learning stage
Answer questions and clarify doubts
Classroom
Practice
Classroom teaching
Practice section

Figure 1 Blended teaching mode

Offline classroom teaching is an effective supplement and deepening of online self-directed learning. In the classroom, teachers can provide targeted explanations for the problems that students face in the process of self-directed learning, and briefly mention or skip the knowledge they have already mastered. This type of teaching content is more targeted, can effectively solve students' questions, and improve teaching effectiveness. The practical teaching process is an indispensable part of the blended learning model. Through practical projects, experimental operations, and other means, students can apply the theoretical knowledge learned in the classroom to practical situations and test their learning outcomes. This stage not only enhances students' practical abilities, but also helps cultivate their innovative thinking and problem-solving abilities.

### 3.2. Deepen School Enterprise Cooperation

Deepening school enterprise cooperation plays a crucial role in cultivating applied talents in software engineering. Through close cooperation with enterprises, schools can better understand market demand, adjust teaching plans, and improve the pertinence and effectiveness of talent cultivation. At the same time, enterprises can also receive support from the teaching resources and talent reserves of schools, promoting technological innovation and industrial upgrading. In the process of implementing school enterprise cooperation, centralized practical training is an important link. Through centralized practical training, students can combine theoretical knowledge learned in the classroom with practical work scenarios, improve their practical skills and problem-solving abilities. For students who have already found a job and are willing to receive internships from their workplace, the school should actively support them to intern at their workplace. This not only allows students to adapt to the work environment in advance, but also deepens their understanding of professional knowledge in practice. In order to better implement on-the-job internships, the school can replace the internship content with relevant courses for fourth grade students. This can not only ensure that students' learning progress is not affected, but also enable them to learn and grow in practice. At the same time, enterprises can also participate in the graduation project of their senior year, providing practical projects and scenarios, allowing students to complete the graduation project in a real environment, improving the practicality and innovation of the design.

## 4. Conclusions

The cultivation of applied talents in software engineering is undoubtedly a systematic and long-term process, and schools must be well prepared for long-term work. This process not only involves the formulation of teaching plans, optimization of course content, and strengthening of practical teaching, but also covers multiple aspects such as cultivating students' professional qualities, close cooperation with enterprises, and building teaching teams. Schools need to have a keen insight into the growth trends and market demands in the field of software engineering, in order to timely adjust training programs and enable students to truly meet the needs of society. At the same time, optimize course settings, strengthen practical teaching processes, and establish effective quality monitoring and feedback mechanisms. By continuously updating teaching content and methods, ensure that students can master solid professional knowledge and practical skills during their school years. At the same time, schools should also strengthen cooperation with enterprises to jointly promote talent cultivation work. Through school enterprise cooperation, schools can introduce practical projects and cases from enterprises, allowing students to practice and learn in a real work environment. In addition, companies can also provide internships and employment opportunities for schools, helping students better transition from school to the workplace.

## Acknowledgements

The authors acknowledge the National Ethnic Affairs Commission Higher Education Teaching Reform Research Project: "Digital Empowerment, Sharing Beauty" - A Study on the Ecosystem of Collaborative Innovation and Practical Talent Training in Software Engineering between Schools and Enterprises (Project Number: 32091).

## References

[1] Ouyang Hongji, Ge Meng, Tang Yunkai.Research on the "application-oriented + innovative" talent training mode of software engineering major in local undergraduate colleges[J]. Microcomputer Application, 2018, 34(2):4.
[2] Zhang Jing, Qin Zhentao.Deepening the integration of industry and education, school-enterprise cooperation, and promoting the cultivation of application-oriented talents——Taking software engineering as an example[J].Journal of Panzhihua University:Comprehensive Edition, 2018, 35(2):5.
[3] Wang Xiaopeng.Training strategy of software engineering application-oriented talents based on engineering professional certification[J].Information Recording Materials, 2018, 19(12):2.
[4] Yang Chunrong, Zhao Xiaoyong.Research and practice of training applied talents in software engineering under the background of Excellence Plan 2.0[J].Computer Knowledge and Technology:Academic Edition, 2022, 18(21):167-169.
[5] Lin Zhuo, Song Jinling.Research on the teaching reform of software engineering course for the cultivation of application-oriented talents[J].Frontiers of Social Sciences, 2018, 7(7):4.
[6] Zou Yi.Exploration of the training mode of application-oriented undergraduate talents in software engineering[J].Xueyuan, 2018(4):2.
[7] Shao Xuehang, Wang Chunming, Zhao Jiahua, et al.Exploration and practice of software engineering curriculum setting based on the transformation of private colleges and universities to application-oriented[J].Heilongjiang Science, 2018, 009(007):46-47.
[8] Zhou Hongbo, Liang Wei, Chen Xiaoyan, et al.Research and discussion on the training system of compound application-oriented software talents[J].China Modern Educational Equipment, 2018(15):3.
[9] Zhang Qiwen, Zhang Qiuyu, Gu Qun.Research on the training mode of software engineering application-oriented talents for new engineering[J].Western Quality Education, 2018, 4(15):2.
[10] Liu Yanji.Research on the construction of application-oriented talent training model for software engineering majors based on industry-university cooperation and integration of competition and education[J].Computer Knowledge and Technology:Academic Edition, 2022, 18(23):129-130.