# Practice and Exploration of Cultivating Software Testing Talents Based on Combination with Industry and Education

Peiyang Wei[a*], Zhibin Li[b], Jianhong Gan[c], Hongping Shu[d]

School of Software Engineering, Chengdu University of Information Technology, Chengdu, 610225, China

[a*]weipy@cuit.edu.cn, [b]lizhibin@cuit.edu.cn
[c]gjh@cuit.edu.cn, [d]shp@cuit.edu.cn

**Abstract.** China firmly holds the top position globally in terms of the number of engineers. Nevertheless, the innovation capabilities and overall quality of the engineer workforce, particularly those in the field of software testing, fall short of meeting the practical demands of the industrial community. The dearth of excellent talents with leading potential is conspicuously at odds with the development requirements of building an innovative society in China. In view of this situation, our teaching team, seizing the opportunity presented by the school's implementation of the "Excellence Engineer Education and Training Program", has been actively collaborating with enterprises in the testing industry. We are committed to exploring a novel talent cultivation paradigm that is oriented towards industrial needs. By introducing interdisciplinary and cross-disciplinary elements, we are establishing new concepts for the integration of industry and education and for collaborative talent cultivation. Simultaneously, we are compiling new-form textbooks that embody industry-education synergy. We aim to create a new model for cultivating elite testing talents characterized by "two intersections and four integrations", and propose a new "one-one-two" practical talent cultivation approach. These endeavors are dedicated to promoting the diversification, intersection, and integration of engineering talent cultivation methods, structures, and levels.

**Keywords:** software testing, talent education, excellent engineer, education reform

## 1 Introduction

Against the backdrop of the country's high attention to the information industry and the continuous strengthening of its support, the field of software applications has shown an extremely strong expansion trend [1]. From various customized software in the process of digital transformation in traditional industries to emerging fields such as mobile Internet applications, the scope of software involvement has been continuously expanding [2]. This trend has made software systems more and more functional, their architectures


H. Kannan et al. (eds.), *Proceedings of the 5th International Conference on Internet, Education and Information Technology (IEIT 2025)*, Atlantis Highlights in Social Sciences, Education and Humanities 40,
https://doi.org/10.2991/978-94-6463-803-5_34

more complex, and the interaction relationships among various modules more intricate, resulting in an exponential increase in the complexity of software design [3].

At the same time, whether it is ordinary users, enterprise-level customers, or key fields with extremely high requirements for safety and reliability (such as healthcare, finance, aerospace, etc.), the expectations for software quality are increasing day by day [4]. As professionals, excellent software testing engineers are required not only to possess solid theoretical knowledge of software testing, but also to have keen problem-solving insights, rigorous logical thinking, and rich project practical experience [5]. Nevertheless, currently, in the realm of practical education and teaching for nurturing excellent software testing professionals, three pronounced issues of "disconnects" are particularly evident [6]. Nevertheless, currently, in the realm of practical education and teaching for nurturing excellent software testing professionals, three pronounced issues of "disconnects" are particularly evident [7].

(1) There exists a remarkable disconnection between the curriculum content and the actual engineering requirements. With the continuous development of China's social economy and in view of the frontier fields of science, technology and industry, the market has an urgent and strong demand for composite software testing talents with a cross-disciplinary professional background. However, at present, in terms of the setting of engineering majors in colleges and universities, problems such as narrow professional coverage and insufficient interdisciplinary integration are widespread. These problems severely restrict the cultivation and development of students' innovative abilities, making it difficult for the talent cultivation in colleges and universities to meet the actual social needs for composite software testing talents [8].

(2) In the context of higher education in China, there is a distinct disconnection between the practical teaching segment and the demands of the industrial forefront. In recent years, in particular, the career development paths of university teachers mostly involve transitions from one university to another. Their selection, assessment, and promotion mechanisms mainly focus on the number of published papers and the quantity of projects undertaken. This situation has led to many teachers having shortcomings in engineering practical abilities and lacking the capacity to effectively lead innovative practical teaching. The content of practical teaching is rather monotonous, and the practical education function driven by interdisciplinary projects has not been fully realized. Moreover, the practical teaching methods are relatively outdated, making it difficult to simultaneously cultivate students' teamwork spirit and promote the integrated development of students' individuality [9].

(3) There is a significant mismatch between the innovation practice platform and the training objectives of software testing talents. Currently, the practice platforms established in Chinese universities mainly consist of various teaching experiment centers at all levels and engineering training centers. These platforms have a relatively single disciplinary layout and face numerous limitations when it comes to supporting students in carrying out highly comprehensive and interdisciplinary scientific and technological innovation - based software testing projects. At the same time, the participation of enterprises in the construction of university practice platforms is relatively low, and an effective operation mechanism that can achieve in - depth school - enterprise integration and thus create a complex engineering practice environment for students has not been

successfully established. This situation severely restricts the cultivation of practical abilities and innovative thinking of software testing major students and is not conducive to achieving the established training objectives of software testing talents [10].

In the software testing talent cultivation system, a series of the above-mentioned problems exist extensively and profoundly. These problems have significantly restricted the cultivation of innovative capabilities, team cooperation capabilities, and the ability to solve complex engineering problems for testing-related professionals. In terms of innovative capabilities, due to factors such as the disconnection between the practice platform and the training objectives and the deviation of curriculum content from engineering requirements, students lack the opportunities and guidance to carry out innovative testing research in real and complex scenarios, and their innovative thinking is difficult to be effectively stimulated and expanded. Regarding team cooperation capabilities, the outdated practical teaching methods and the imperfect school-enterprise integration mechanism make it difficult for students to form an efficient collaborative team cooperation model during the practice process, and they cannot fully experience and master the key role of teamwork in the advancement of software testing projects. As for the ability to solve complex engineering problems, the practice platform with a single-discipline layout and the lack of a real engineering environment created by the in-depth participation of enterprises result in students lacking sufficient practical experience and comprehensive knowledge application ability to deal with complex software testing engineering problems. All these make the current achievements of software testing talent cultivation difficult to meet the urgent needs of the country for excellent software testing talents with leading potential at this stage, and unable to provide solid talent support for the high-quality development of the national software industry and innovative breakthroughs in cutting-edge technology fields.

## 2 Reform Methods of Education and the Implementation Process

To effectively address the prominent "three disconnections" issues in the cultivation of software testing engineering talents, our team has established a clear and forward-looking guiding ideology. That is, adhering to the concept of "interdisciplinary integration of academic disciplines, school-enterprise collaboration, innovation in cultivation models, and the unity of knowledge and practice to cultivate excellent software testing talents". Guided by this ideology, we have identified the reform of the practical education mechanism and methods, as well as the innovation of practical education approaches, as the key breakthrough points. Our aim is to cultivate a group of excellent software testing talents who can meet the development needs of the technological and industrial frontiers and possess the potential to be leaders.

Driven by this goal, our team has fully leveraged the advantages of school-enterprise collaboration, meticulously designed, and launched a brand-new "two-cross and four-integration" school - enterprise collaborative cultivation model for excellent testing talents. The "two-cross" emphasizes the cross-integration of disciplinary knowledge and the cross-application of different professional skills, enabling students to build a diversified and comprehensive knowledge system and skill framework. The "four-integration" focuses on the integration of theory and practice, teaching and scientific research, schools and enterprises, as well as knowledge dissemination and ability cultivation, comprehensively and deeply enhancing students' overall qualities and professional abilities. Through this innovative model, we aim to create a learning and practice environment that is closely integrated with industrial reality and full of innovation vitality for students, helping them grow into excellent talents capable of leading the development of the software testing industry.

The "two interdisciplinarities" refer to the interdisciplinarity of the school-enterprise cooperation practical platforms and the interdisciplinarity of academic disciplines and specialties. The "four integrations" denote the integration of professional orientations, the integration of teaching faculties from schools and enterprises, the integration of cultivation methods, and the integration of students' comprehensive qualities and abilities. The relevant train of thought is illustrated in Figure 1.

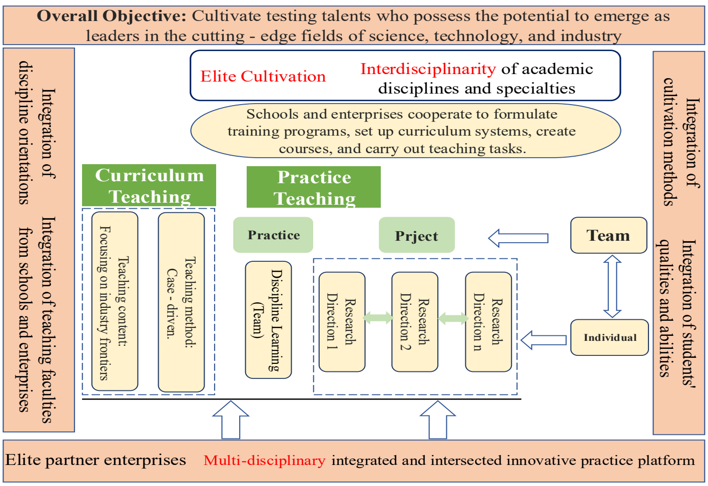


**Fig. 1.** Implementation Framework Diagram of the New Model for Software Testing Talent Construction.

1. Innovative Elite Cultivation Group Oriented towards Industrial Demands and Technological Frontiers.

First, in alignment with the “Four Orientations” principle, our institution initiates partnerships with enterprises in the software testing industry. Through in - depth communication and joint research, the main research directions of the Elite Group are determined. The overarching goal is to cultivate excellent interdisciplinary talents, aiming to overcome the current challenges such as the narrow scope of the software engineering major and the limited cross-disciplinary integration.

Second, regarding student selection, a joint school - enterprise expert panel interview mechanism is established. In line with the talent requirements of the software testing industry, an Elite Group is formed by integrating students from different disciplines, majors, and academic years. This approach promotes a diversified learning atmosphere and enriches students' perspectives in the field of software engineering.

Third, in the aspect of student cultivation, the Elite Group creates an open and boundary - less learning and growth space for students with surplus academic capabilities. By providing various resources such as advanced research projects, academic seminars, and industry-academic exchange opportunities, students can fully explore their potential in software engineering.

Fourth, for textbook selection, we introduce new - style textbooks. These textbooks are led by real - world industrial projects in the software engineering field, accompanied by supportive micro - course videos. Compiled by teams dedicated to integrating learning and practice, they can better meet the practical teaching needs of software engineering and help students master relevant knowledge and skills more effectively.

2. Innovate the "1-1-2" Practical Education Model through School-Enterprise Collaboration.

The innovation of the practical education model serves as the pivotal support for the new "Two Cross-overs and Four Integrations" school-enterprise collaborative cultivation model for excellent testing engineering talents. The establishment of the Elite Group breaks through the constraints of the sole efforts of universities. Through in - depth school-enterprise collaboration, both parties jointly participate in formulating cultivation plans, constructing curriculum content, building teaching faculties, and other measures, integrating the practical education model throughout the entire process of cultivating students in the testing direction.

In terms of curriculum content, facing industry demands and technological frontiers, schools and enterprises cooperate to create cases in different business domains, realizing new-format textbooks and courses featuring interdisciplinary integration and the combination of learning and practice. In terms of teaching methods, a diversified teaching team is formed by senior enterprise experts and teachers from different specialties. Theoretical teaching mainly focuses on instruction and practice, while practical teaching adopts a project - driven approach, aiming to stimulate students' innovative capabilities and practical potential.

Leveraging the guidance of new - format textbooks and through in - depth school - enterprise collaboration, this initiative has effectively addressed the issues of the monotonous content in practical teaching and the lack of innovation in practical teaching among university faculty. Moreover, it has creatively established a "1-1-2" practical education model: Firstly, a project team, consisting of 3-5 students from the Elite Group, is formed to target industry demands and technological frontiers. The team formulates a single cultivation plan and is co - educated by the "dual" teaching resources from both the university and the enterprise, as illustrated in Figure 2. Secondly, each student in the project team takes the lead in one cutting - edge testing topic within a project and collaborates with other topics in the project. They are guided by the "dual" tutors jointly appointed by the university and the enterprise, as shown in Figure 3.

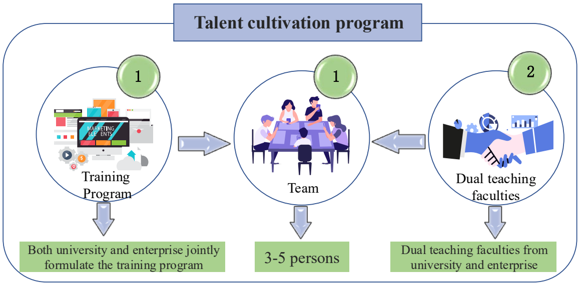


**Fig. 2.** The Cultivation Model of "One Team, One Strategy" with Dual Tutors.

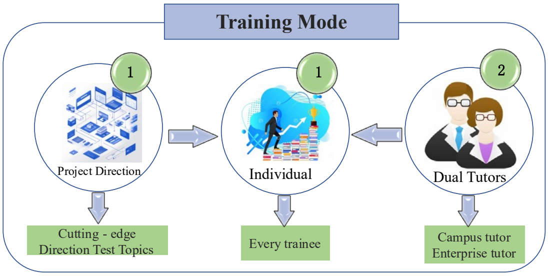


**Fig. 3.** The Guidance Model of "One Student, One Topic" with Dual Tutors.

The "1-1-2" practical education model in the field of software testing engineer possesses the following 5 distinctive characteristics.

Firstly, a team consists of 3-5 students from the Elite Group. Once organized, they freely form teams based on personal interests, with members from different majors and classes. Each member has unique skills, enabling them to divide tasks and cooperate. They complete all theoretical and practical tasks assigned by tutors as a team. In the process, students' professional knowledge, thinking, and research methods blend. They support each other in learning and practice, improving together. This teamwork in software engineering helps cultivate well - rounded talents.

Secondly, each student team gets a customized cultivation plan. School and enterprise tutors jointly design it, setting internship and testing project directions. Guided by dual-tutors, teams pick testing areas and projects per their interests, handling one business-domain project each. To boost students' innovation and practice, tutors blend traditional internships into testing-project research. They identify top members, and based on individuals' traits, recommend and place suitable students in enterprise internships. This helps students apply knowledge, enhancing engineering skills. Such a method in software engineering education links academic learning to industry needs.

Thirdly, each student in the software engineering program leads one research task. Once the team selects a business-domain project, school and enterprise tutors assign specific testing directions and research topics according to members' skills and specialties. Each student takes charge of one task and participates in others' tasks. This task-allocation approach maximizes students' research and growth opportunities. It helps students deepen expertise in their areas, promotes cross-task cooperation and knowledge sharing, and effectively enhances their comprehensive capabilities in handling complex software projects, better preparing them for the software industry's challenges.

Fourthly, the Elite Group in software engineering implements a school-enterprise dual-tutor system. This collaborative model combines tutors from universities and industry. During practical teaching, an enterprise-centered approach is used to address the lack of engineering practice skills among university faculty. Enterprise tutors, with their industry experience, offer practical knowledge like current software development processes and problem-solving strategies. University tutors provide theoretical guidance. This dual-tutor system helps students integrate theory with practice, promoting their all-around development and better equipping them for the software industry.

Fifthly, the new-format three-dimensional textbook accompanied by micro-course videos play a leading role. Team members, having summed up years of teaching and project-based practical experience, and in light of the characteristics of efficient teaching and the actual demands of enterprises for testing talents, have refined the content from aspects of knowledge acquisition and case implementation. They have made reference to a vast number of relevant domestic and international standards, case studies from testing websites, and real questions from past software professional qualification examinations to enrich the textbooks. At the beginning of each chapter of the textbook, key points are provided for students' convenient review and learning. At the end of each chapter, real questions from past software evaluation engineer examinations are attached to test students' learning outcomes.

Facing education and industry frontiers, the "1-1-2" practical education model is adopted. Each team has its strategy, each student has an individual topic, with dual-tutor guidance. This model provides a teamwork-cultivation platform and personalized growth space. It allows students to tap potential in their interested areas and enhance scientific literacy. From learning to practice and achieving results, engineering practice and innovation skills are constantly nurtured. In software engineering, like in team-based projects, teams set strategies, students focus on specific tech topics under dual - tutor guidance, thus improving practical and innovative abilities for industry challenges.

3. Innovate the multidisciplinary innovative practice platform integrating and intersected by different disciplines.

In order to support the students of the Elite Group in carrying out multi - disciplinary cross-innovation practical activities, the team has spared no effort to integrate resources. It has introduced resources from off-campus enterprises and jointly innovated and planned multi-disciplinary cross-innovation practical platforms. Each platform incorporates at least two or more testing projects from different business domains, breaking through the shackles of the single - project nature of the original platforms. With one business domain as the leading force, these platforms involve the integration of multiple disciplines. By bringing in the advantageous resources of several well known domestic enterprises, four interdisciplinary cross-innovation practical platforms in the fields of meteorology, ecological environment, medicine, and intelligent manufacturing have been constructed. These platforms cover multiple disciplines such as science, engineering, medicine, and management, enabling students to engage in comprehensive and in-depth practical learning across different fields, which is highly conducive to cultivating their ability to solve complex engineering problems from a multi-disciplinary perspective in the context of software engineering, for example, through the development and testing of integrated software systems related to these cross-disciplinary platforms.

In the realm of software engineering, take the Cross-disciplinary Innovation and Practice Platform for Meteorological Informatization as an instance. This platform is mainly built upon the two Experimental Teaching Demonstration Centers in Atmospheric Science and Software Engineering. By collaborating with multiple provincial-level meteorological agencies across the country, a comprehensive innovation and practice platform has been established. Its objective is to carry out rigorous testing on the robustness, security, and functionality of specific business modules within the meteorological informatization systems. This platform offers a practical and multi - disciplinary scenario for students. It enables them to apply software engineering techniques, such as software testing methodologies and system integration strategies, in the context of meteorological applications, thus enhancing their capabilities to handle complex, real-world software engineering projects. The platform features three main modules: Climate Business Testing, Meteorological Data Testing, and Meteorological Detection Innovation & Practice. It integrates multiple disciplines like meteorological data, climate algorithms, detection equipment, computer science, and control engineering. While enhancing students' professional and research skills, it fosters interdisciplinary thinking, multitasking, and comprehensive innovation abilities. In software engineering, students can use programming for testing tools and control engineering knowledge for equipment optimization, deepening their understanding and problem-solving capacity.

Multi-disciplinary cross-innovation practice platforms have distinct features. They're co-built by schools and enterprises with collaborative tutoring. Projects, sourced from real industry problems, are problem - oriented. As Figure 4 shows, comprehensive projects enable Elite Group student teams to practice, integrating multi-disciplinary knowledge.

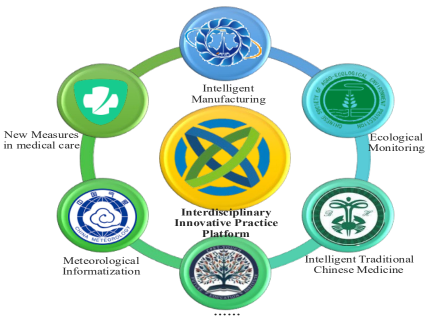


**Fig. 4.** Multi-disciplinary Cross-innovation and Practice Platform.

## 3 Reform and Innovation in Education and its Characteristics

We've pioneered the "Four-Fold Integration" practical education mechanism. Guided by "interdisciplinary convergence, school-enterprise synergy, model innovation, and unity of theory and practice", schools and enterprises jointly build multi - disciplinary practice platforms. This mechanism integrates students' majors, school-enterprise teaching resources, education models, and students' competencies. It offers novel perspectives for cultivating top - notch software testing engineers. In software engineering, students can combine knowledge from multiple fields. With guidance from both school and industry tutors, they can improve testing skills and problem - solving abilities, thus fostering well-rounded talents.

We propose a new "1-1-2" practical education model. Based on students' teamwork and personalized development needs, it features "one-team-one-strategy with dual teachers" and "one-student-one-topic with dual tutors". This model effectively boosts students' ability to solve complex engineering problems and enhances their engineering practice and innovation capabilities. It offers a replicable and scalable approach for cultivating excellent testing talents. In software engineering, it allows students to work in tailored teams and get guidance from different tutors, promoting comprehensive skill development in software-related projects.

A novel "mutual-benefit and win-win" collaborative educational paradigm is created. In software engineering, schools and enterprises cooperate. They co-formulate cultivation plans, co - design curricula, co-build a faculty with academic and industry experts, and co-create teaching resources. Schools get an environment for cultivating cross-disciplinary talents. Enterprises obtain needed testing talents and research outputs. Their cooperation serves as an experimental ground for top - tier software engineering talent.

## 4 Conclusion

Educational reform outcomes have significantly addressed the “three disconnections” in software testing talent cultivation. By devising an innovative model, optimizing the cultivation system, and driving reform implementation, distinct features and innovations have emerged, yielding positive impacts in application.

In software engineering terms, the new model incorporates real-world projects into teaching, and the optimized system balances theory with practice, like adding hands-on testing courses. Going forward, deeper school-enterprise collaboration is crucial. We must track industry trends, refine the talent cultivation model, and produce more top-tier software testing talents with leadership potential. This will boost China's software industry's innovative growth.

## Acknowledgments

This research is supported by the Sichuan University students innovation and entrepreneurship training program (S202410621082) and Chengdu University of Information Technology key project of education reform (JYJG2024206).

## References

1. Liu, Lijuan, Yong Bai, and Shaowei Zhang. "Quality evaluation of software engineering professional talent training under the background of new engineering." International Journal of Knowledge-Based Development 15.1 (2025): 24-41.
2. Yi, Z., Jun-hao, W., Ling, L., Qing-yu, X., Min, G. Study of Progressive Training Plan on Software Engineering Talents. In: Kassel, S., Wu, B. (eds) Software Engineering Education Going Agile. Progress in IS. Springer, Cham (2016).
3. Zheng, Wei, Yutong Bai, and Haoxuan Che. "A computer-assisted instructional method based on machine learning in software testing class." Computer Applications in Engineering Education 26.5 (2018): 1150-1158.
4. M. Kassab, J. F. DeFranco and P. A. Laplante, "Software Testing: The State of the Practice," in IEEE Software, vol. 34, no. 5, (2017): 46-52.
5. Gura, O. Foreign experience of software testing engineers training: analysis of educational standards and programs of European universities. ScienceRise: Pedagogical Education, (4(61), (2024):42–47.
6. Shen, Jianqiang, et al. "The Reform and Practice of Talent Training Mode Based on the Criteria of Engineering Education Accreditation." 2017 3rd International Conference on Social Science and Higher Education. Atlantis Press, (2017).
7. Reem A. A. The impact of the GQM framework on software engineering exam outcomes. International Journal of Advanced Computer Science and Applications, 15(10), (2024).
8. I. Ozkaya, "Application of Large Language Models to Software Engineering Tasks: Opportunities, Risks, and Implications," in IEEE Software, vol. 40, no. 3, (2023):4-8.
9. Lemos, Otávio Augusto Lazzarini, et al. "The impact of Software Testing education on code reliability: An empirical assessment." Journal of Systems and Software 137 (2018): 497-511.
10. Ting Zhang, Ivana Clairine Irsan, Ferdian Thung, and David Lo. 2025. Revisiting Sentiment Analysis for Software Engineering in the Era of Large Language Models. ACM Trans. Softw. Eng. Methodol. 34, 3, Article 60 March, (2025).