Universe Scientific Publishing

10.18686/ahe.v6i20.6424

Teaching Reform of Software Engineering under the Background of Engineering Education Certification

Wanhuai Zhou, Hao Li, Xuedong Zhang, Mingjie Li

School Of Management Science and Engineering, Anhui University Of Finance and Economics ,Anhui 235037,China

Abstract: To fix the problem of ineffective training of engineering mindset and ability in teaching "Software Engineering", these teachers take the construction of new engineering disciplines and engineering education certification as the starting point, and break the traditional knowledge-imparting mode by employing "immersive" practice: online & offline teaching, project-driven, role playing and open-source platform. This innovation proves effective in improving teaching quality and fostering talents.

Keywords: Output Orientation; Task-driven; Software Engineer; Role play

Fund Project: This study is supported by the Quality Engineering Project of Anhui Province (2020XSXXKC021) and the Teaching Research Project of Anhui University of Finance and Economics (acjyyb2019060)

1. Introduction

Software Engineering is an important foundation course for computer majors and the soul of software courses. This course integrates engineering management technology and software technology, and requires students to master three elements of software life cycle, namely methods, tools and processes, emphasizing both theory and practice. It systematically sorts out the tasks and ability requirements of each stage of software development and requires students to comprehensively use their expertise to solve practical engineering project problems, which is one of the cores of system-level professional training. It also focuses on training students' teamwork, team management and innovation, laying a foundation for fostering excellent software engineers. It has always been a difficulty in the teaching of relevant professional courses, mainly reflected in: complex and abstract content, hard to understand, split in theory and practice, no clue to start any real project development [1].

2. Training objectives of the course

In our training program for Computer Science and Technology major, Software Engineer as a core course, requires 36 hours of theory teaching and 36 hours of practice, with a ratio of extracurricular time out of class no less than 1:1, and it is scheduled in the second semester of the third grade. Based on the output-oriented concept, the Intended Learning Outcomes (ILOs) of this course are prepared in accordance with AUFE school-running orientation and characteristics, as well as supporting relationship between courses and training objectives, as shown below:

ILO1: understand basic concepts of Software Engineer, software life cycle, and development techniques

ILO2: be able to carry out feasibility studies from many aspects, especially economic feasibility studies

ILO3: be able to capture software requirements and write requirements specifications

ILO4: have the ability to design different development scenarios and come up with appropriate solutions within constraints

ILO5: be able to design and implement software systems according to the solution

ILO6: be able to test, review, and inspect software systems developed

ILO7: have the ability to choose the right methods and tools based on the solution

ILO8: be able to master profound management theory knowledge and attach importance to team management

ILO9: be able to communicate in both formal and informal forms

ILO10: be able to arrange and manage the project schedule according to the constraints

ILO11: be able to make use of the information platform and digital resources to continuously learn independently and improve itself

ILO12: be capable of respecting and complying with intellectual property rights

3. Instruction design

3.1 Project-driven teaching

Project-driven teaching mode, based on the CDIO concept, is adopted to strengthen the training of students' practical abilities in engineering. Through school-enterprise cooperation and introduction of open access platform and open access resources, we managed to enrich our sources of practice projects, and tailored them to the actual needs of enterprises; A simulated user as innovation was introduced into the project team, and was responsible for collecting, refining or changing project requirements appropriately as specific problems arise, and the final evaluation and feedback of the project.Teachers don't impart the boring and tedious theory or book knowledge any longer, and students are guided by theories from real projects and went through the whole life circle of their projects. In this way, teachers can "teach through project development" and students can "practice in the process of learning" and a deep integration goal of teaching and practice is therefore achieved. To meet the practical needs, students are moved from the traditional classrooms into the teaching laboratories, and the teaching content is more vividly displayed in front of students, and the interaction between students and teachers is more convenient. In addition, interweaving theory and practice makes practical demonstration and drill more convenient than traditional classroom teaching.[1]

3.2 Mixed online and offline teaching

Compared with the traditional pure offline teaching mode, the online and offline mixed teaching mode has the following advantages: 1. to make full use of abundant online learning materials and resources, such as MOOC platforms, open access platforms and online teaching platforms; 2. to record the whole teaching process and students can review some key points, difficulties and doubts in great depth after class; 3. Online teaching videos are highly targeted, concentrated and compact, with little requirement for learning sites, and is convenient for students to focus on single knowledge point in a short time; 4. to gradually enrich teaching resources, record students' learning process and learning effect, improve teachers' work and students' learning efficiency, and identify the problems in students' learning process. With the help of network teaching platform, good teaching resources and experience are gradually accumulated, and problems are being solved, which leads to a better teaching mode online and offline combined.

3.3 Practice teaching

Practice teaching is an important way to transform theoretical knowledge into engineering practice ability, and the construction of new engineering has made it crystal clear for Software Engineer practice. First of all, the content of course practice should fully reflect the characteristics of "engineering", and lay a foundation for engineering practice and cultivation of innovation abilities. Secondly, to foster students' engineering practice and innovation abilities through curriculum practice. Finally, the evaluation of curriculum practice should set benchmark for engineering practice and cultivation of innovation abilities. Based on the above ideas, the proportion of classroom practice hours has jumped from 20% to about 60%. Not only to include some practice in traditional theoretical courses, a comprehensive Software Engineer practical training course is also added. Project tasks run through the entire teaching process, with the integration of scientific research and teaching, more open access platforms and enterprises are introduced for cooperative education and practice projects, with all the training tasks and training platform tailored to enterprise actual needs. A simulated user is added in each group to identify different needs, and makes some project improvements or adjustments accordingly. With the help of network platform, the whole process of project development is well managed, and team members cooperation efficiency is improved, the assessment method is optimized, and each member's contribution to the project is appropriately measured. [2]

4. Process Management

4.1 Class management

In order to improve the teaching efficiency and ensure the quality of teaching, the class management adopts the four-level management mechanism, namely teacher, assistants, student cadres and project manager. The teachers are mainly responsible for theory instructions, practice guidance and overall management of the class. The teaching assistants and student cadres assisted in project team building, project task arrangements and daily school supervision. And the project manager is in charge of the project team management, scheduling & planning, and personnel organization. Theory instructions are in natural classroom, while practice is conducted in group discussions, PPT presentations, communications and guidance mode; In addition, at the technical level, the efficiency of class

management is also much improved with the help of software platforms, such as Superstar, U+ new engineering platform and Ding-ding. With all this, not only the teaching quality and effect is guaranteed, the teaching efficiency is also greatly improved.[3]

4.2 Project team management

A good team is the key to a project success. Therefore, teachers, teaching assistants and student cadres should first select a group leader with strong management abilities in the class, and then select some group members with strong technical abilities for each group. Finally, they team up on their own to play to their strengths. In this way can we avoid putting all talents in one group and ensure each group has the capabilities for project development, promoting internal learning, reasonable task designation and mutual collaboration. The project manager should designate roles and tasks to team members accordingly: analysts and designers with strong macro control abilities, coders and testers with strong execution abilities, inspectors with right attitudes. What's more, a simulated user is deliberately added in the project team to identify, adjust, add or alter project needs in the project execution process, and in this way can we not only ensure tailoring the project to enterprise actual needs, but also make the project development process more fun. Teachers and teaching assistants act as project inspectors, organizing project reviews and experience sharing seminars at critical stages of the software life cycle. Role playing in engineering can help students learn from different perspectives, accumulate engineering practice experience, and effectively stimulate students' learning enthusiasm at the same time.

4.3 Course materials management

In order to improve the efficiency of course materials management, we use a learning platform Chaoxing and set up multiple tasks in different phase, such as feasibility study, needs analysis, development plan, software design, coding & testing, software deployment, and summary report, etc. And students are required to submit relevant technical documents, software code, PPT report and other course materials in different phase. To help students develop good habits in document management, we made some rules to regulate their document classification, document naming, version management, storage and backup, and require students to save every important document completely in the software life cycle.

5. Conclusion

By making full use of teacher resources with mixed online & offline learning, "immersive" practice teaching and role play tailored to real engineering projects, we managed to train students' abilities of independent learning and problem analysis, and help them accumulate some practical experience. Formative assessment solves the problem of differentiating team performance and individual contributions. The simulated user in the project not only makes the practice projects more close to real workplace, but also more fun and greatly improves the teaching quality. Setting different assessment nodes in software life cycle and different proportions according to their importance in each stage helps guide students to foster a good Software Engineer thinking model. With all the measures mentioned above, we improved our teaching effect significantly.

References:

[1]Li Jie. Exploration and Practice of Software Engineer curriculum Reform [J]. Journal of Beijing Institute of Aerospace Technology, 2020. 30(1): 39-41.

[2]Shi Xiaoqiu. Curriculum Design and Implementation in accordance with the Concept of Professional Certification OBE [J]. Higher Engineering Education Research, 2018(5): 154-160.

[3]Liu Qiang. Curriculum Remolding of "Software Engineer" Based on OBE Concept [J]. China University Teaching, 2018, 10: 25-31.