ATLANTIS PRESS

Advances in Social Science, Education and Humanities Research, volume 298

2nd International Conference on Education Science and Social Development (ESSD 2019)

Software Engineering Course Construction by University-Enterprise Cooperation for MSE

Huobin Tan
School of Software
Beihang University
Beijing, China

Xiyang Liu
School of Software
Beihang University
Beijing, China

Xueyun Ji
School of Data Science and Software Engineering
Baoding University
Baoding, China

***Abstract*—Aiming at the prominent problem of cultivating practical ability for Master of Software Engineering (MSE), a software engineering course are constructed by university-enterprise cooperation, which introduces enterprise experts and case resources. Emphasizing both theories and practices as well as combining classic and innovative simultaneously, the course is reformed and built as graduate practice seminar. The effect is also analyzed, which is generally recognized by students.**

Keywords—University-Enterprise cooperation, software engineering course, practice seminar

I. INTRODUCTION

Unlike traditional Academic Masters who focus on scientific research, master of engineering is a professional degree that is related to the qualifications of engineering field, which is directly oriented to train and provide applied, compound, high-level, technical or managerial talents for enterprises. Master of Software Engineering (MSE) is a new master degree in engineering with the development of software industry. The main purpose is to cultivate talents with high academic qualifications in various fields of software engineering, such as software development, project management, network security and so on, to meet the demand for high-level talents with rapid development of country's software industry.

In order to highlight the importance attached to the needs for enterprises in the training of master of engineering, the Graduate School of Beihang University has launched the "Graduate Practical Seminar Program", which aims at introducing a number of high-quality curriculum resources from leading domestic and overseas enterprises and industry institutes [1]. The school invites enterprise's technical experts to participate in the course construction and lectures, aiming at strengthening the organic combination of theories in textbook and actual cases, improving the cultivation of practical ability, promoting reform in curriculum teaching, realizing the collaborative education between university and enterprises, and improving teaching quality of full-time professional degree postgraduate courses.

The School of Software, Beihang University, as an authorized unit for MSE, has been constantly making attempts in training programs and course construction, and actively carried out various university-enterprise cooperation through the way of "bringing in enterprise resources and going out with achievements of the school" [2]. The Software Engineering Course is the basic professional course for MSE. In 2018, it was funded by the "Graduate Practice Seminar Program", and a new postgraduate teaching mode with university-enterprise cooperation was established, which won high praises from students.

II. RESEARCH AND ANALYSIS OF RELATED COURSES

The software engineering course is first positioned as a core compulsory course for MSE, which purpose is to systematically introduce the professional skills and related practical contents of software engineering for students. However, since most students have already learned the basic courses of software engineering at the undergraduate stage, they have a preliminary understanding of basic concepts, processes and methods in software engineering. Consequently, the course for MSE should be positioned as an improvement course, and the course name is determined as Advanced Software Engineering. However, what is advanced? How to promote, and from what aspects? These are discussed in depth.

To clarify these issues, the researches were carried out widely on related software engineering courses for postgraduates from Carnegie Mellon University, University of California, Berkeley, Peking University and other well-known universities.

As a well-recognized software engineering university, the software engineering professional program of Software Research Institute (SEI), CMU is well known as MSE program. The core courses of the major are set up around each stage of software engineering, including five courses: Models, Method, Manage, Analysis and Architecture, covering system modeling, design methodology, project management, test analysis and software architecture respectively. These are the core professional skills during the whole software development process. Through the comprehensive learning and practice of such multiple courses, it covers the modeling methods and techniques at various stages including requirements, design and testing stages, as well as the project management skills throughout the whole process.

In additional, for students who are lack of software engineering foundation, CMU recommends that students take an advanced introductory software engineering course: Introduction to the Personal Software Process (PSP) [3]. Developed by Watts S. Humphrey, PSP is a self-improvement process for controlling, managing, and improving personal working style, including a structured framework of software development forms, guidelines, and procedures. The core of this course is to improve the quality of work by establishing a personal software development process. As an advanced placement, students are required to complete the course through distance learning before enrollment (usually from the end of May to the beginning of July during the summer period). After finishing the course, students will be able to achieve the following goals: (1) reducing the global defect rate, (2) spending more time in early stages of software development cycle, (3) eliminating defects of software compilation and testing, (4) evaluating the time required to build software more accurately. The course covers various PSP activities, including the following contents:

- starting from PSP level 0, guiding all content of level 2.1, and completing the process through 9 assignments;
- using a new tool to collect personal data and using it for subsequent analysis;
- collecting process data, analyzing and improving related concepts;
- time management and concentration technologies;
- the follow-up work of PSP and knowledge body related to "statistical" software development

University of California, Berkeley and Yale University do not have separate software engineering courses for postgraduate students. These courses are taken with undergraduates while the requirements of final grade are different. The software engineering course at the University of California, Berkeley mainly introduces concepts and technologies about how to design, develop, and modify large-scale software systems. The goal of the course is to provide a detailed introduction to object-oriented software development methods and to apply them to actual project practices. The course, covering the entire process from concept phase to final deployment, enable students master approaches to develop medium-sized software systems using modern technologies and can develop their team project practical experience and let them be able to carry on technical communication in writing.

The School of Electronics Engineering and Computer Science at Peking University in China provides the “Advanced Software Engineering” course, which centers on the advanced and in-depth concepts of software engineering. It mainly includes the following contents:

- software and software development
- SSH-based software development
- the overview of middleware
- the basis of interaction of software entities
- Web services
- quality middleware
- software quality
- model verification
- code analysis

The Advanced Software Engineering course offered by the School of Software Engineering, University of Science and Technology of China is divided into two semesters. The first semester of the course is oriented to students who have weak professional foundations, lack of programming experience or only experience in small-scale code writing, and only requires basic foundation of C language and data structure; while the second semester requires high professional foundation and learning ability, where students should develop applications while learning new languages and frameworks. The contents consist of two parts. The first part involves content of object-oriented analysis and design, the process is subdivided into use case modeling, domain modeling, object interaction modeling, and class design. It is gradually developed into in-depth topics such as model design, software architecture and so on, from easy to difficult. The second part introduces the software develop process and management, focusing on conceptual methods related to the overall situation of software engineering projects, such as software life cycle, task partitioning, workload assessment, project scheduling, project management, progress tracking, etc.

Through these surveys, it can be seen that although the CMU curriculum system has great referential significance, it has much periods of the whole course and requires multiple courses to be fully covered, which is difficult to achieve under our training programs. To this end, the overall construction of postgraduate software engineering courses is determined to follow the curriculum content of CMU. On the basis of ensuring the establishment of students' basic software engineering ideas, it focuses on key activities of software engineering such as modeling methods, architecture design and testing. In terms of course subjects, referring to the ideas from other universities such as Peking University etc., the newest software engineering practices should be included to keep up with the trend of the industry.

Furthermore, the practice system matched with the course needs should also be carefully designed to support classroom instruction. In regard to practice settings, simulation cases are used to support the practice process in most cases, and students are organized to complete software engineering practice activities by extracting practice cases from enterprises. It's often difficult for such simulation cases to effectively reflect various complex problems existing in projects, such as requirement change, requirement defects, and so on. Meanwhile, the project practice part of CMU is very useful for reference. Its project cases come directly from the actual needs of open source communities. Students are directly involved in the activities of open source community under the guidance of teachers. Before the beginning of each semester, teachers will contact the relevant project with open source communities and when it's possible, organize students to participate in the development

of outsourcing projects from open source communities. The experience of being able to join directly in a real-world project during school is indeed very helpful for students to learn and comprehend software engineering. However, due to the real project schedule and cost, the teaching arrangement and schedule are very strict. Each school year has kind of uncertainty for different projects, which is disadvantage to develop the course. In addition, there is no mature open source community and open source project development mode at present in China, so this model is difficult to popularize effectively. Therefore, it's another key point in the construction that how to carry out and implement the course on the based on enterprise elements.

III. Course Construction

Through the investigation and analysis, the objective of Advanced Software Engineering course is to systematically introduce the software engineering professional skills and the practice to students, including individual engineering skills and professional skills from software requirements to design and testing. And the course combines with experience of actual software system development from leading companies to explore professional practices such as software architecture design and system testing, thus laying a good foundation for students to learn and implement follow-up professional courses. How to effectively combine theories with enterprise practices is the course construction's key point [4].

A. Construction Ideas and Methods

The goal of curriculum construction is to further introduce enterprise resources into the course on the basis of covering core skills of software engineering for postgraduates. With the rich practice experience of enterprises, theories and practices are fully integrated, so that theories can be applied to practices and students can learn through practices, thus students are able to fully understand and master relevant knowledge points.

1) Emphasizing both theories and practices, combining classic and innovative simultaneously: Teachers mainly introduce theoretical methods, while enterprise experts mainly use enterprise practices to cover classic software engineering technologies as well as latest technologies and applications such as micro-service architecture, internet testing, etc.

2) Case-driven teaching: Introducing actual engineering cases from well-known leading enterprises to raise up problems through cases and explain principles.

3) Teaching by seminars: Organizing students to join in seminars, through seminars students share their experience of software engineering practices, teachers and other students participate in discussions and make comments.

B. Course Content Construction

The next step is to clarify the specific course content, which is the emphasis of the entire curriculum construction and the basis for carrying out the course smoothly. Through the previous researches of related courses, combined with the overall objectives of this construction, the entire course contents are shown in Table I.

The contents are divided into 6 parts. The recommended class hours is between 30 and 48. Due to the gradual introduction for enterprise resources, the initial period is 30 class hours. With the introduction of more enterprise resources, the final class hours is expected to be 48 hours.

TABLE I. Arrangement of lecture content

No.	Chapter Name	Main Content	Teaching Form	Class Hours (hrs.)
1	Introduction: Looking for "Silver Bullet"	Through Tar Pit and Silver Bullet in *The Mythical Man-Month*, introducing dilemmas, causes and possible solutions in the software development; explaining briefly the basic concepts of software engineering and clarifying its development context	Teachers lecture on theories	3
2	Personal Skills of Software Engineering	Introducing personal practical skills of software engineering, including unit testing, code quality and specification, code review, version management, pair programming, professional ethics etc. to develop student's basic software engineering awareness	Teachers lecture on theories and organize students to practice	3-6
3	Professional Skills of Software Engineering	Introducing core skills related to requirements, design, and testing in software engineering practice. Students work in a team to conduct activities of requirements, design, and testing	Teachers lecture on theories and organize students to practice	9-15
4	Software Engineering in the Internet Age	Introducing software engineering in the Internet environment, including new concepts and evolutions such as Web Service, SaaS, micro-services, and intelligent software	Enterprise experts lecture on special topics	3-9
5	Software Engineering from Enterprises	Inviting software engineering practitioners from both domestic and overseas leading software enterprises or research institutes to introduce their software engineering practices and research results	Enterprise experts lecture on systematic practices	9-12
6	Software Engineering Frontier Seminars	Organizing students to discuss frontier knowledge in software engineering. Students with working experience can analyze and communicate the products	Students make presentations and interact with teachers and enterprise experts	3

The first part is introduction, which starts with the topic of "silver bullet" in *The Mythical Man-Month* [5], introduces topics of software crisis and the naissance of software engineering, explains what is the main topics in the Software Engineering Body Of Knowledge (SWEBOK) [6], so as to establish a general view of software engineering for students.

Part 2 focuses on personal skills in software engineering practices, combined with CMU's introductory course of software engineering: Introduction to Personal Software Process (PSP), introduces various skills that individuals must master in software engineering practices, such as unit testing, coding standards, etc.

Part 3 talks about professional skills of software engineering, including methods and practices of requirement, design and testing three main stages in software life cycle, which are core skills for software engineers.

Part 4, 5 mainly combine the current mainstream technologies, researches and practices in enterprises. In the two parts, enterprise experts would be invited and they would introduce relevant implementation in the enterprise's product development. The teaching forms include lecturing on special topics and systematic practices. Part 4 mainly involves new methods and theories, while part 5 focuses on actual product development and practice.

Part 6 is the seminar. Students, especially with working experience, make presentations on their working practices, and discuss with teachers and enterprise experts.

In terms of teaching forms, there are lecturing from teacher and enterprise experts, student's self-practice, communication, discussion and other forms. According to the content and objectives, these different forms are combined to realize the interactive teaching among teacher, enterprise experts and students.

C. Introducing Enterprise Experts and Cases

As the core ties of university-enterprise cooperation, enterprise experts play a vital role. In order to guarantee the teaching quality and effect, there are some basic demands for them.

1) Enterprise experts should have rich practical experience and have worked on research and development of large scale software systems.
2) Enterprise experts should have a certain theoretical abilities, be capable to summary and extract from practical experiences, form systematic approaches to effectively integrate theories and practices.
3) Enterprise experts should have good communication skills and presentation abilities.

After selecting enterprise experts, the next step is to bring in the enterprise expert's projects as course cases. Taking directly first-hand cases from enterprise experts is also a major feature for the university-enterprise cooperation. However, actual cases are often inappropriate to course teaching due to some issues such as scale and complexity etc. So, it is necessary for teachers and enterprise experts to simplify and adjust cases, tailoring them around teaching tasks to meet teaching needs.

D. Evaluation Methods

The course's evaluation includes four parts: classroom attendance, classroom performance, practices after class and final exam. Classroom attendance accounts for 5%, mainly for the statistics of each class attendance. Classroom performance accounts for 10%, divided into two parts: performance of answering questions in class and the final class discussion. Practices after class includes two parts: individual practices and professional practices by teams, accounting for 5% and 30% respectively. Individual practices are mainly aimed at examining individual's basic skills of software engineering. Team practices involve activities of project requirements, analysis and testing by team work as well as the result of implementing required artifacts. Final exam accounts for 50% and fully examines the situation for each knowledge point.

IV. COURSE IMPLEMENTATION

The course has been implemented in the fall of 2018 among postgraduates from School of Software, Beihang University. In the round, the course was taught for 30 hours, and two enterprise experts from Alibaba Network Technology Co., Ltd. and Beijing Fun Software Technology Co., Ltd were invited to the course. The two enterprises have a leading position in their respective fields. The two technical experts not only have a strong practical ability, but also have published many professional books, with a high theoretical level. The expert from Alibaba, as the development supervisor of a voice platform, systematically explained methods and practices in architecture design based on his production and the current mainstream technologies of micro-service architecture. The enterprise expert from Beijing Quna is a senior test manager who have rich experience in software testing, systematically explained methods and practices of software testing, especially internet testing.

After the semester, the course effect has been widely recognized by students. In the result of course evaluation organized by the Graduate School aimed at all 41 students, the rate of excellent reached 90.5%, and the rate of excellent and good was 100%. The result ranked among the best in all graduate courses. Table 2 shows the specific evaluation results.

In terms of the rate of excellence, indicator 3 and 5 did not reach 90%, indicating that further improvements should be made. These two indicators focus on keeping the course content with cutting-edge and strong practice. Since it was only the first year implementation, there was indeed not enough attention on these two aspects and further enhancement is required.

TABLE II. RESULT OF COURSE EVALUATION

No.	Evaluation content	Rate of excellent	Rate of excellent and good
1	In general, it's very rewarding to learn the course	92.5%	100%
2	Generally speaking, I'm satisfied with the teachers' performance	92.5%	100%
3	I understand the content of this course and master relevant concepts, theories, applications and the latest developments	87.5%	100%
4	This course focuses on thinking training and enhances my sense of innovation	92.5%	100%
5	This course focuses on ability training and improves my ability to analyze and solve problems	87.5%	100%
6	Total rate of excellent, total rate of excellent andd good	90.5%	100%

In addition, an anonymous questionnaire was organized to the university-enterprise cooperation mode, which a total of 40 students participated in. 36 students approved that the mode was great and the course structure was very reasonable. The proportion reached to 85.7%, indicating that the mode has been generally recognized. Moreover, For the question that what do you hope enterprise experts to talk about, 19 students chose the “combining the application of theoretical knowledge with specific products”, while 21 students thought that they should “focusing on introducing enterprise culture and overall methods related to practices”. It has shown that students have greater interest in management and culture of enterprises what had no chances to be learned in school, instead of technologies. It can be taken into consideration for subsequent curriculum arrangements that invite enterprise experts who work on management to introduce enterprise practices with enterprise culture, not just technologies.

V. CONCLUSION

The key idea for “ Graduate Practical Seminar Program” is to introduce enterprises into class and carry out teaching with enterprise resources, which is also the starting point for developing software engineering courses with university-enterprise cooperation. An advanced software engineering course for MSE is constructed by investigating, combining self needs, and inviting well-known software enterprise experts such as Alibaba and Quna. Focusing on both theories and practices, combining classic and innovative simultaneously, the course structure has been established. The effect of implementation has been recognized by students. Further enhancement and improvement should be made in terms of course contents and practices afterwards to build an excellent course.

ACKNOWLEDGMENT

Thanking to enterprise experts what have participated the course, especially Lu Han from Alibaba Network Technology Co. Ltd (alibaba.com) , and Weidong Cai from Beijing Quna Information Technology Co. Ltd. (qunar.com) etc.

REFERENCES

[1] Y. Li, J. Zhang and Y. Wang, “Graduate Embedding System Practice Seminar Based-on University-Enterprise Cooperation,” Computer Education, pp. 46-49, 2013.

[2] L. Zhang and H. Tan, “The Exploration of Target-driven Master of Software Engineering Traing Program,” Computer Education, pp. 22-25, 2015.

[3] W. S. Humphrey, Introduction to the Personal Software Process: Addison-Wesley Professional, 1996.

[4] H. Mao, N. Jiang, W. Su and Y. Zhang, “A New Model on the School-Enterprise Cooperation Curriculum Construction,” IEEE 25th Conference on Software Engineering Education & Training, 2012, pp. 115-119

[5] F. P. Brooks, The Mythical Man-Month: Addison-Wesley, 1975.

[6] P. Bourque and R. E. Fairley, Guide to the Software Engineering Body of Knowledge (SWEBOK v3.0): IEEE Computer Society, 2014.