

The Exploration and Practice on the Teaching Reform of Software Engineering Course in Higher Vocational Education

-Integrate CDIO and OBE education concepts

Fangmei Ning

Suzhou Centennial College
Suzhou, Jiangsu, China

Chen Zhou, Lantao You

Suzhou Industrial Park Institute of Services Outsourcing
Suzhou, Jiangsu, China

Abstract—This paper discusses and studies the teaching mode of software engineering course, which is the core course of software technology major in higher vocational education. Aiming at the problems existing in the teaching of software engineering curriculum, and combined with the characteristics of higher vocational education teaching, this paper puts forward a set of teaching reform measures that take project-driven approach under CDIO as the core, project development cycle as the main line, and product prototype design as the main practice content, at the same time, a new evaluation mechanism based on OBE is introduced. The reform measures inspire students' interesting and power in the software engineering, promote the students' practical innovation ability, and improve the quality of teaching.

Keywords—CDIO; OBE; software engineering; project-driven; competition analysis; mind mapping; product prototype design

I. INTRODUCTION

Software engineering refers to the concept and idea of engineering, uses engineering methods, processes and tools to develop software, make the process of software development manageable and controllable, so as to ensure the quality of software development [1]. Software engineering course is a core professional course for computer related majors. The purpose is to cultivate students' engineering ability and comprehensive quality, master the whole process of software development and the tools and methods in the process, and improve their teamwork ability. In the actual course of offering, it is difficult to teach and learn because of its high comprehensiveness, abstract theory and wide practical content. This is particularly prominent in the teaching process of higher vocational colleges.

CDIO engineering education model is the latest achievement of international engineering education reform in recent years. Since 2000, the transnational research made up of four universities including Massachusetts institute of technology and royal institute of technology of Sweden has received a huge grant of nearly 20 million dollars from Knut and Alice Wallenberg foundation. After four years of exploration and research, the CDIO concept of engineering education was founded and an international cooperative

organization named after CDIO was established. CDIO represents conception, Design and implementation and operation, It takes the life cycle from product research and development to product operation as the carrier, to let the students to actively, practice, course organic connection between courses to study engineering. The CDIO training syllabus divides the abilities of engineering graduates into four levels: basic engineering knowledge, personal ability, interpersonal and team ability and engineering system ability. The syllabus requires comprehensive training to enable students to reach the predetermined goals at these four levels [2].

Outcomes-based Education (OBE) was first introduced in the United States and Australia in the reform of basic education. From the 1980s to the early 1990s, OBE was a popular term in American education. This model is deeply studied in the book "output-based education: controversies and answers", written by American scholar Patrick. The book defines the OBE as "to focus and organize the education system clearly, ensure that students gain the experience to achieve substantial success in their future lives." He thinks OBE has realized the transformation of educational paradigm. Because, in the OBE education model, what students learn and whether they succeed are far more important than how and when they learn[3]. Different from the traditional teaching mode, under the OBE concept, teachers' pay more attention to students' learning effect.

II. ANALYSIS OF TEACHING STATUS

At present, it is difficult to teach software engineering courses in higher vocational education, and there are mainly the following problems:

A. The course content is highly abstract

The knowledge content of software engineering course mainly includes the development model, analysis method, modeling process, quality management, development standard, tool use method, etc. For vocational college students without any enterprise project development experience and weak theoretical foundation, the content is very abstract and difficult to understand.

B. The course content is comprehensive

The software engineering course not only contains abstract software engineering concepts, concepts, methods, etc., but also contains a lot of practices, including various engineering graphics drawing, software development and implementation. How to explain these highly integrated knowledge point? How to practice the software project development process? How to combine theory with practice effectively? These questions are the current challenges of software engineering courses.

C. The teaching of software engineering in higher vocational education is more difficult.

The training orientation of higher vocational education focuses on skills and applications. For software engineering courses with strong comprehensiveness, on the one hand, the theoretical foundation system of higher vocational students is relatively weak. on the other hand, the characteristics of higher vocational students' learning are that it is difficult for them to understand and digest the theoretical content, which is too empty for them to understand. The traditional software engineering teaching mode cannot be digested by vocational college students. After a semester, students do not know the value and significance of this course, and at the same time, it reduces students' learning interest and enthusiasm.

III. PRACTICE PLAN OF TEACHING REFORM

In view of the above analysis of the current teaching situation, the author combines her own teaching experience in Higher Vocational education, and puts forward the following operational reform measures.

A. Project-Driven under CDIO Concept

The foundation of software engineering curriculum reform must be project-driven teaching. The software development life cycle described in the course of software engineering, from project establishment, requirement analysis, design, implementation, operation to maintenance, is just symbolic of the process of conception, design, implementation and operation in CDIO concept. In the process of actual teaching reform, every stage of software development must involve students and let them become the dominant one. Teachers should play the role of project manager and give more technical guidance and quality monitoring. Only through this course mode students can finally understand engineering concepts, principles and methods in different stages of the project, and practice the technology and tools of software engineering, and summarize the concept, experience and standards of software engineering in the project.

B. Phase Review Based on OBE Concept

According to the concept of OBE, what students learn at each stage of project development and whether they succeed is the most important. So at the beginning of the course, we should set up the time node and product list of each stage in software development, use project output to drive the learning process of students, and take the completion degree of stage product as the evaluation content of students. Teachers as project managers, on the one hand, guide the learning process

of students, on the other hand, Check and evaluate stage products in the specified nodes for the project teams.

C. Weakening software coding and implementation in software engineering course

In the course of project-driven software engineering teaching, students should enter the stage of software coding after completing project establishment, requirement analysis and design. However, this practice stage is often difficult to achieve. The main reasons are as follows: 1. the duration of software engineering is limited, and the total duration of software engineering in a semester is 64 hours. It is unrealistic to really complete a software project coding. 2. students 'development ability is limited. Software engineering courses are usually offered in the second or third semester. Its pre-courses include basic programming language, database, basic Web, etc. The knowledge needed to develop a complete, medium-sized software project is far from enough, most students can only complete a console application or a simple web front-end project, but the project development of this scale is not significant for software engineering learning, because even if students complete the relevant code development, because the project size is too small, the structure is not complete, students still can't understand the idea of software engineering.

In addition, the main course goal of software engineering course is to locate in the whole process of software engineering implementation, not in coding. In the whole course setting of software specialty, there are many courses of program development, program practice/practice, and the core orientation of these practical courses is software coding and implementation. Based on the above analysis, the author believes that the implementation of software coding in software engineering course can be ignored, and students can understand the coding process and specifications by demonstrating and explaining it.

D. Strengthening the task of product prototype design

The prototype is a product framework depicted by lines and figures, also known as wireframe. Prototyping is the concretization and interface of abstract requirements, and the integration of product processes and functional logic. At present, with the rapid development of Internet products, more and more attention has been paid to product design and user experience design, so prototyping design is becoming more and more important. Now, there are many prototyping design tools on the market, such as Axture RP, Mockplus, Modao and so on, developers can quickly design high fidelity product prototypes that can demonstrate the operation.

In the life cycle of software development, prototype design is a link between requirement analysis and project design. It can also be said that it belongs to a part of requirement analysis. In the stage of requirement analysis, students usually record requirements in the form of flow charts and documents, while prototype design draws interfaces corresponding to functional requirements. At the same time, the process of the interface involves the business logic of the project, which can exercise students' project thinking ability and cultivate students' product

awareness. In the course of software engineering curriculum reform, the author adds the task of product prototype drawing. Through product prototype design, students can experience the process from creativity to operation of a product, and complete the prototype design and production of enterprise-level projects by zero programming.

E. Competition analysis

Competition analysis refers to the comparative analysis of competitors 'products, and it is also an important link in the implementation of software engineering curriculum reform. During the project establishment stage, the project team conducts research and discussion on the creative projects submitted by the team through competitive analysis, and the product positioning of the project has been determined. At this stage, each project team needs to identify 2 to 3 products of the same type for analysis, including market analysis, business model, product strategy, and product architecture and so on. Through competitive analysis, students can broaden their thinking and discuss extensively. Students can not only analyze the success of current hot applications (including creativity, demand points, design patterns, etc.) through team strength, but also learn how to find the pain points in the product market and skillfully combine their creativity with market demand. And eventually turn their creative projects into operational products.

F. Use mind map to stimulate students' innovation and weaken traditional project documents

The software development life cycle contains a large number of documents, including project approval documents,

requirements design documents, outline design documents, detailed design documents, test documents and so on. In addition, software engineering courses involve the drawing of many graphics and the use of tools, which are used to complete the drawing of various flow charts; In order to better stimulate students' thinking and increase the fun of learning, the author reduced amount of documentation, added the drawing of mind map in the software engineering course, requiring students to use mind map to present the product functional structure map of project group. Because mind map is more flexible and the presentation style is more abundant, it greatly reduces the dryness of software engineering courses and facilitates communication among team members.

G. Innovative curriculum evaluation mechanism

The traditional software engineering course assessment is usually composed of attendance, assignments, and final examination. Most of the examination forms are written examination. However, as a practical and comprehensive subject, software engineering is totally unsuitable for this assessment method. So the author adopts the method of periodic course assessment. For each important node of the project development cycle, the author evaluates the scores of each team at each project stage through the form of students' speech. At the same time, the author designs individual scores and team scores to mobilize the enthusiasm of each student, and at the end of the final term. Examination scores only account for 30% of the total score, which greatly stimulates the interest of learning.

TABLE I MAIN PRACTICE PHASES OF PROJECT CYCLE

Num	Task	Phase Review under OBE	Tools	CDIO phase
1	Project grouping	Project grouping table Role table	/	conception
2	Project Initiation	Project Creative Book Competition Analysis Report Project Report PPT	Visio ProcessOn PowerPoint	conception
3	Project Requirement Analysis	Product Functional Architecture Diagram Mind Map of Product Demand Function Core Function Business Flow Chart	Visio ProcessOn PowerPoint XMind	conception
4	Design and Implementation of Product Prototype Diagram	Running product prototype	Modao PhotoShop	Design and implementation

IV. CONCLUSION

Software engineering course is a practical, theoretical and comprehensive course. This paper puts forward some suggestions on the reform of software engineering teaching in Higher Vocational education. Through the project-driven teaching under the concepts of CDIO and OBE, each practice link is optimized, in which product prototype design is the core practice and combine team discussion, competition analysis, mind mapping, speech reporting and other forms and links, so that students gradually understand the concept and ideas of software engineering in the process. Through the author's

three-year practical teaching effect, teaching and learning have been significantly improved.

ACKNOWLEDGMENT

This paper is the research topic of The First Subsidized Subjects for the Research of Higher Vocational Education Reform in Suzhou (SGJGA0117), and is one of the research results of Suzhou Industrial Park Institute of Services Outsourcing Teaching Reform Project (JG-201708, JG-201905).

REFERENCES

- [1] Hu Sikang, *Software Engineering Foundation (2nd Edition)* [M], Beijing: Tsinghua University Press, 2017:5.
- [2] Baidu Encyclopedia, CDIO, <https://baike.baidu.com/item/cdio/4644769?Fr=aladdin>.
- [3] Baidu Encyclopedia, OBE, <https://baike.baidu.com/item/OBE/20282657?Fr=aladdin>.
- [4] Du Xiaokun, Tu Tao, Xu Shengzhou, Preliminary Exploration on Comprehensive Teaching Reform of Java Programming and Software Engineering [J]. *Modern Computer*, 2016 (11): 57-60.
- [5] Yang Juying, Practice and Exploration of Integrative Teaching Reform of Software Engineering Course [J]. *Educational Frontier*, 2017 (47): 208.
- [6] Fu Xiaolin, Tang Sen, Liu Gang, Zhang Lijie, Research on the Teaching Reform of Software Engineering Experimental Course under OBE [J]. *Software Guide*, 2018 (10): 216-219.
- [7] Li Xiaodan, Teaching Reform and Exploration of Software Engineering Course in Higher Vocational Colleges [J]. *Industry and Science and Education Forum*, 2013 (13): 188-189.
- [8] Ma Li, Li Zhen, He Weina, Shan Donghong, Exploration of Software Engineering Online Development Course Construction Based on CDIO Concept [J]. *Software Guide*, 2018 (17): 15-17.
- [9] Li Junxian, a preliminary study on the teaching reform of "Software Engineering" course based on the mixed teaching mode of "Flipping Classroom+" [J]. *Education and Teaching Forum*, 2017 (39): 116-117.
- [10] Han Yehong, Teaching Reform Research of Software Engineering Course in Applied Universities [J]. *Journal of Qilu Normal University*, 2018 (33): 65-69.