Research on the Construction of Software Integrated Design Course

Jin-Lian DU\textsuperscript{a,*}, Xue-Yun JIN\textsuperscript{b} and Dan WANG\textsuperscript{c}

Faculty of Information Technology, Beijing University of Technology, Beijing, China
\textsuperscript{a}dujinlian@bjut.edu.cn, \textsuperscript{b}jinxueyun@bjut.edu.cn, \textsuperscript{c}wangdan@bjut.edu.cn

*Corresponding author

Keywords: Software Integrated Design Course, Project-Based Teaching, Teaching Organization, Evaluation Mechanism.

Abstract. Software integrated design course is one of the most important courses to cultivate students’ professional ability and excellent engineering quality. Since it collects the knowledge of data structure and algorithm, object oriented programming, database principle, software engineering, software quality testing and other courses as a whole, how to organize the teaching activity effectively to achieve the prescribed teaching objective is a big challenge. In this paper, a project-based teaching and learning idea is proposed based on the in-depth analysis of the training goal of this course, and the concrete construction ideas and implementation process around the teaching content, teaching organization and evaluation mechanism is expounded in detail.

Introduction

Lots of basic courses in computer science and technology, such as data structure and algorithm, object oriented programming, database principle, software engineering and software quality testing, provide the basic knowledge of application software development. The kernel content of these courses all relate to the obligatory indispensable critical technology in software development. For example, all kinds of data structures are the basis of programming, database is an important part of the most applications, object oriented programming is a software construction technology, software engineering from macro view provides the method and process for software system development, and software quality testing provides testing and evaluating mechanism in order to guarantee software system quality. At present, these courses have made abundant achievements and accumulated enormous amount of teaching resource by improving their teaching method and extending their teaching content for years and every course includes the related practice part.

However, there is still a distance between the students’ training quality and our training goals. Especially for the real requirement of the society, the engineering ability of students cannot meet the requirements of the enterprises, which is noticed by enterprises, education and university. Then where is the problem? After interviews with the teachers in charge of graduation project and investigation with the students, we find that because the contents of different courses are taught independently, the relationships among knowledge of different courses are weakened, so the knowledge student achieved is fragmented. It is difficult for students to establish a complete knowledge framework; at last, they are baffled when they face practical problems, which as a result cause them lack of self-confidence.

Therefore, under the direction of our university’s undergraduate training plan, we set up software integrated design course for the computer science and technology students, which accords with the idea of “clarify training requirements, highlight specialized characteristics, strengthen the basics, encourage integrated curriculum” and follows the viewpoint of “on the premise of strengthening the mathematics and science foundation, focusing on the cultivation of students’ engineering ability”. The objective of the course is to help students integrate professional knowledge, finds the application position of the knowledge, experience the thought and steps of solving the professional problem
systematically, establish a systematized knowledge framework, and improve engineering ability. This is very important for cultivating the engineer in the new era [1,2].

Course Objective

The objective of this course is forming the knowledge framework of information processing application software design and development by integrating the critical knowledge of 5 courses: data structure and algorithm, object oriented programming, database principle, software engineering, software quality test through practical application cases.

Teaching Content Organization

Teaching content is the primary problem in the course construction. Since the goal of the course is developing the engineering ability of students, we implement project driven teaching under CDIO (Conceive, Design, Implement, Operation) education concept [3,4], by which students learn through projects development step by step. This will help them to tie all knowledge together. Therefore, selecting an appropriate project and the integration of knowledge are the key works of teaching content organization.

The Choice of Projects

Two aspects should be considered when choosing a project. (1) The project should have practical application background or be a contest title. In order to stimulate students' learning enthusiasm, the project should be a popular application and meet the social needs. (2) The content of the project should cover the critical knowledge points of these 5 courses and tie related knowledge together.

Therefore, we cooperate with enterprises actively to obtain real projects or cases, and try to reconstruct them by assimilating related knowledge into it to form a teaching project suitable for this course. At the same time, students' learning level also must be considered; we try to provide appropriate projects with different difficulty to students with different level to make sure most of the students can be improved at different levels in this period.

The project of this course should be adjusted according to the teaching content and the social needs every term. This adjustment needs to be done by vast numbers of teachers. We communicate with enterprises every year to introduce latest application project. We will continue to do this work for a long time to make sure the teaching content can keep pace with the times.

Knowledge Organization

The organization of knowledge can decide the knowledge realization process in project implementation. Our solution is: in general, we use the methods of software engineering as the guide, and divide the project implementation into 5 steps: requirement analysis, high level design, detail design, code debugging, system test and integration. In high level design period, related knowledge about database modeling is merged into database design, and the knowledge about object oriented programming is merged into database design; in detail design period, related knowledge about data structure and algorithm is merged into the construction of function and related data structure design; software quality management and testing is merged into code debugging and system integration. In order to form a good framework that chains all subjects’ knowledge, the related knowledge introduced in every period will be negotiated by on campus and off campus teachers.

Teaching Process Organization

The organization of teaching process is a very important research content in the construction of this course. It includes: the design of teaching model, the organization of teachers and the organization of students.
Teaching Model Design

Project based software integrated design course focuses on practicality and engineering. In order to explore the subjective initiative of students, teachers involve into the learning process of students as a guider. In the beginning, teachers introduce the content, requirements and specifications of the projects to the students, and try to lead the students to establish a right thinking of project development macroscopically by showing the project case in class. In the other periods, teachers also only reflect the value of guidance. Take the class design problem as an example, what kind of classes should be built or how to configure their relationship should be done first by students themselves under the framework of object oriented programming. What should teachers do is to evaluate students’ design according to the needs of the project, and guide them to optimize their design. On campus or off campus teachers can explain the solutions of related cases from different angles to inspire students to expand their thinking. This approach leads students to explore and practice, to overcome difficulties actively, and to solve the problem at last by themselves. They will obtain the sense of success, and build their own knowledge system.

Teachers Organization

Software integrated design course involves a large amount of work, contains lots of knowledge points. It needs organize teachers from different course groups and enterprise to participate teaching activity according to the training goal and the key knowledge in different periods. In order to achieve the desired teaching effect, we organize teachers to form a close teaching team to accomplish the teaching process of this course. Besides, we make fine division of teaching process and detail the following: concrete content, related knowledge structure, which contents should be explained by teachers, which contents should be designed by students, and the teaching and learning models. Then we arrange the teaching task of every period to specific teacher according to their knowledge structure. The teaching team is headed by the team leader and the team leader is responsible for arranging and coordinating teachers’ tasks. Teachers should provide related teaching record, teaching achievements and the evaluation of the students for the teaching activity of next period. The final evaluation will be done by all teachers to evaluate students in different aspects. Therefore, the teaching process is accomplished by the coordination of all teachers.

Students Organization

All students are divided into groups, and the number of team members is determined according to the workload of the project. In general, each group consists of 4 students and is headed by a student as leader. The group leader is responsible for segmenting the task, allocating the work to every group member. Every week, the group members should communicate with each other, report the progress and the problems encountered, discuss project solution, and write a weekly report. During the evaluation period, every member of the group must attend the oral defense to present their work to the teachers. This ensures every student is really involved in the learning and exploring process, and get benefits from the course.

Evaluation Mechanism

In order to achieve training goal of the course which is improve the engineering competence of the students, we develop a set of systematic and diverse evaluation system which includes 9 indicators in this system: learning ability, design ability, practice ability, self-discipline ability, communication skills, public speaking ability, writing ability, organizing ability, innovation ability. As shown in Figure 1, under one-hundred marking system framework, design ability accounts for 20%, practice ability accounts for 30%, learning ability, self-discipline ability and organization ability account for 10% respectively, communication skills, public speaking ability, writing ability and innovation ability account for 5% respectively. In order to make the evaluation more subjective and comprehensive, we further subdivide these indicators according to the training goal. For example, learning ability is
subdivided into 4 sub-indicators (shown in Figure 2): comprehension ability, reading material ability, the ability to master new knowledge and discussion participation. Design ability is subdivided into 5 sub-indicators: system architecture design, database design, the application of object oriented principle, human-computer interaction, data structure and algorithm. Practice ability is subdivided into 4 sub-indicators: coding and debugging, software testing, system extending and software deploying ability. Self-discipline ability is subdivided into 3 sub-indicators: attendance, independent ability and compliance with arrangements.

All sub-indicators after subdivision not only cover the assessment of the knowledge of various subjects, but also cover other aspects of students’ abilities. About the approach of evaluation, we use various methods. Teachers are in charge of evaluating on learning, design, practicing, renovating, writing, and oral speaking. For self-discipline ability, communication skills and organization ability, part of them is evaluated by other students in the same group, such as the independent completion and accepting arrangement of the self-discipline ability is assessment by the student in the same group. This evaluation approach uses a wide range of evaluation sources to facilitate the objective formation of evaluations.

This meticulous, systematic and pluralistic evaluation system, as well as a variety of forms of assessment mechanism which focuses on allowing students get full training, can enable students to understand their strengths and weaknesses clearly, that provide reasonable guidance for their future study.

Figure 1. 9 Indicators in Software Integrated Design Course.

Figure 2. Example of Subdivision of Learning Ability.

Conclusion

The training goal of software integrated design course is to improve students’ ability to apply knowledge comprehensively, and cultivate students' strong engineering literacy. In order to achieve
the goal, we design every period of the teaching process of this course in detail. For the teaching content, we propose an idea to integrate knowledge based on project. For the teaching organization, we design a diversified teaching and learning model which teachers of multiple subjects corporate, co-instruct, and students participate into groups. For the learning evaluation, we design a precise multidimensional evaluation system including 9 indicators: learning, design, practice, self-discipline, communicating, writing, public speaking, organizing and innovation. Our method has been practiced for a semester and the result shows that the method is effective to ensure the expected goal of this course.

Acknowledgement

This research was financially supported by the Education Foundation of Beijing University of Technology (No.ER2015C020101).

References


