

Teaching Experience Sharing on Course “Software Quality Assurance and Testing” *

LIU Shu YIN Sheng-jun

(School of Software, Harbin Institute of Technology, Harbin 150001, China)

Abstract Based on teaching experiences accumulated from past 5 years author shares the teaching process and approaches on course of “Software Quality Assurance and Testing”, which includes course syllabus, teaching approaches, and evaluation methods, etc. We hope that this paper would have a positive input to those who are in the process of teaching software engineering related courses as well as to those who are in the process of teaching practice-oriented bilingual courses.

Key words Software Quality Assurance, testing, the teaching approach

CLC number TP31 **Document code** A **Article ID** 0529-6579 (2007) S2-0109-03

1 Introduction

Software quality is the key of software products. As the scale of software projects has been expanded and the degree of project complexity has been increased the software quality is becoming the focus of software production. With various uncertainties most software projects have been delayed, over budgets, and finished with poor quality. Therefore Software quality assurance and testing are causing considerable attentions in software industry as well as in software engineering education. Software quality assurance and testing has become required course in many universities in China. As a senior lecturer with eight years software industry experience in U.S. and five years teaching experience in School of Software of Harbin Institute of Technology author shares her teaching approaches and thoughts in this paper, which is organized in six sections. Followed with introduction the 2nd section covers course content. The 3rd section details the time allocation, and the 4th section depicts the teaching approaches. The 5th section introduces the evaluation methods and followed by the conclusion.

2 Course Content

2.1 Focus on Both Software Quality Assurance and Testing

Most universities have Software Testing course at present, while only a few institutes have Software Quality Assurance or software quality related courses. However, there are few universities which set up the course

integrated with both software quality and testing. If the two aspects, software quality and testing are separated as two courses, students would realize the importance of software testing or the importance of software quality, meanwhile might feel confused about the relationship between the two. Software testing has been widely accepted as one very important phase in software development life cycle but software quality assurance has seldom been mentioned in software engineering process. Actually software testing is a part software quality assurance and software quality assurance also includes software quality control and software configuration management. To integrate the quality assurance and testing in one course students could distinguish each responsibility during the software development, understand how to manage software configuration, establish the quality assurance plan, and improve process tracking and control. And the students could also pay attention to the change management, software measurement and defect analysis, apply the testing strategies and testing techniques to enhance the software quality, and so on.

2.2 Course Contents

The Software Quality Assurance and Testing course in HIT focuses on both aspects meanwhile emphasizes testing is not equal to the quality assurance testing is only one of important part of the quality assurance activities. Quality assurances also cover configuration management and quality control and other software activities. The course consists of five major parts, detailed as following:

* Part One: Software Quality Assurance Frame-

收稿日期: 2007-10-28

作者简介: 刘曙 (1967年生), 女, 高级讲师; Email: sls@hit.edu.cn

- work
- * Quality Related Concepts and Software Quality Assurance
 - * Software Configuration Management
 - * Software Quality Assurance Group and Software Quality Assurance Plan
 - * Statistical Software Quality Assurance
 - * Part Two Software Testing Techniques
 - * Static Testing Techniques
 - * Dynamic Testing Techniques
 - * Part Three Software Life Cycle Testing
 - * Waterfall Development Testing
 - * Iterative/ Spiral Development Testing
 - * Testing in Maintenance Environment
 - * Part Four Software Testing Tools
 - * Part Five Introduction to Software Metrics

3 Time Allocation on 40 Course Hours

The Software Quality Assurance and testing courses in HIT is given 40 course hours. The detailed arrangements on the contents of the course are listed in table 1. As showed in the table, it will take 20 lectures to complete this course with 2 hours for each lecture. The 18 lectures are fairly allocated to cover the five parts of the course content, and 2 lectures are driven by students' presentation on their team projects.

Tab 1 Time Allocation

Lecture	Topic	Course Hours	Note
1—4	Course Opening	8 hours	
	Software Quality Assurance Framework		
5—6	Static Testing Techniques	4 hours	
7—11	Dynamic Testing	10 hours	
	Techniques		
12—13	Life Cycle Testing	4 hours	1st assign Due
14—15	Iterative/ Spiral Development Testing Methodology	4 hours	
	Testing in the Maintenance Environment		
16	Software Metrics	2 hours	
17	Modern Testing Tools	2 hours	
18—19	Team Project Presentation by students	4 hours	Team Project Due
20	Summary and Closing	2 hours	
	Final Exam		

4 Teaching Approach

- 4.1 Industry-Oriented Teaching
- (1) Teaching with industry experience

Since Software Quality Assurance and Testing is a practice-oriented course, it is important to have the lecturer with software industry working experience. The teaching process is not only focus on knowledge acquisition but more importantly on the practical skills. As the lecturer to this course, I was engaged in software quality assurance for several years while working in the software industry in U.S. I realize that software quality assurance is still a soft spot in domestic software industry. In a lot of domestic enterprises, software quality assurances are being the state of waiting for developing. On one hand, the proportion of developer team and QA team is imbalance, and most of enterprises have very big QA/ testing team doing QA/ testing work, even greater than developer team; on the other hand, whole technical capability of the QA/ testing team is far weaker than the developers. Then, the teaching started with correcting the misunderstandings by putting emphasis on the relationship between QA and testing and meanwhile bringing the reality to the students by analyzing the conflicts existed between development team and QA team. In order to coordinate relation between the two and achieve the goal of improving software quality, QA management tools are introduced. In addition to the theory, there are some popular software configuration and testing tools are also introduced during the course, for example, SVN of configuration management, CVS, JUnit and Loadrunner etc.

(2) Teaching with multi-viewpoints

After graduating from School of Software of HIT, most students will work in IT field as a software developer, a software tester, and a software end-user etc. As we know that people in different positions would have concern on different aspects of software quality. It is important to prepare students with this kind of knowledge by teaching with multi-viewpoints according to their possible future working directions. For instance, concerning software quality, from developers' perspective, they focus on those quality attributes such as correctness, testability, flexibility, portability, reusability, variability, stability etc.; from software end-users' perspective, they more focus on reliability, usability, efficiency, and from management perspective, they more concerned with schedule, cost and productivities of software project.

4.2 Bilingual Teaching

With the pace of globalization has increased, English has been the most widely accepted and used communication language. Under this environment, bilingual teaching has become the new trend in recent years in university, which promoted by China Ministry of Education. This course is taught in bilingual languages with

80% of English and 20% of Chinese. All course materials including text/reference books, lecture slides, assignments and final exams are given in English, and all lectures are also given in natural English with Chinese just as supplement for the difficulty points. After these years bilingual teaching we have gotten considerable positive feedbacks from graduates. Through bilingual teaching it prepares students with good English communication skills which are very important in their future employments. For example, with solid practice-oriented technical skills and good English abilities some students got big advantages in their job interviews, and some students beat thousands competitors and got jobs in world leading IT companies such as IBM, Microsoft and Oracle, etc.

5 Course Evaluation

With continue accumulative evaluation method instead of one final exam, the learning processes have been measured and evaluated. In this course the total final grades have been broken down into four parts spread through out the whole processes of teaching-learning. As showed in table 2, 100% final grade, it consists of 10% of class Participation, 20% of assignment, 30% of course project, and 40% of final exam.

Tab 2 Grade Breakdown

Breakdown		Grade Percentage
Participation		10%
Assignment		20%
Course Project		30%
Final Exam	40% (note, case book & case note)	
Extra Credit		5 Points

It is necessary to give 10% grade on participation to encourage students participate in class learning, especially on learning practice-oriented skills. The as-

signment is designed to evaluate individual ability by testing programs and writing test reports. The course projects are designed to evaluate students team ability. First each team with 4 ~ 5 students will select a topic from a list of software tools (e.g. testing tools, version control tools, QA management tools, etc.) given by instructor, then work on it as a team. In the final two lectures each team will turn in a hard copy report on the topic they chosen and present their topic with specific tool usage demonstrations to the whole class. Final Exam (case book & case note) is used to test the basic concepts and fundamental knowledge on the subject. Extra Credit as an encouragement will give the students who have done the excellent work during course study.

6 Conclusion

Combining software quality assurance and testing as one course could give students a more complete picture on software quality improvement. With software industry-oriented contents, practice-oriented teaching approaches, and accumulative evaluation methods, the course—— “Software Quality Assurance and Testing” in School of Software of HIT has been a good example for industry-oriented teaching. While this is just the first attempt on industry-oriented education there are far more works need to be done in the future in order to keep up with the path of rapid development of software industry.

References

[1] HUANG Shihong, DISTANTE Danilo. On Practice-Oriented Software Engineering Education [J]. Proceedings of the 19th CSEETW, 2006

[2] BRICHAU Johan, CHITCHYAN Ruzanna, GARICA Alexandre, et al. A Model Curriculum for Aspect-Oriented Software Development [J]. IEEE Software, 2006, 23(6).