

## Curriculum reform and practice of software testing

Zhu Bin<sup>a</sup>, Zhang Shiming<sup>b</sup>

School of Computer and Information, Shanghai Second Polytechnic University, Shanghai, 201209, China

<sup>a</sup>email: tozhubin@163.com, <sup>b</sup>email: smzhang@sspu.cn

**Keywords:** software testing; occupation experience center; teaching method; practice teaching

**Abstract.** With the increase in the importance of software quality and enterprise needs testers, how to train students become qualified testers is an arduous task for education community all over the world. Based on the experiences of teaching software testing course, we carry on reforms in software testing course. The reforms include the teaching content, teaching resources, the environment of the experiment, teacher construction etc. The results indicate that these reforms can achieve a better teaching effect in software testing course.

### Introduction

Software testing is an important phase in software development and is a part of the software quality assurance (SQA) process. In recent years, because the market gradually increased awareness of the importance of software quality, domestic software enterprises have gradually attached great importance to software testing activities [1]. However, even if they know the importance of software testing and its position in the whole development process, lack of professional testers is still a serious obstacle for software industry progress. Software testing is not a simple and low-level work; you need to have a certain educational background and level of education to be competent. Training software testing professional is an important and urgent task for promoting the development of software industry. In the history, lack of training of software tester's mechanism, the result is a serious shortage of software testers. How to train qualified testers studied in many literature [2, 3, 4, 5].

The traditional university computer education is the main channel for training software testers. In many colleges and universities in computer-related specialty, software testing course as stand-alone courses appear relatively late compared with other computer courses. With the development of software, testing technology is constantly changing. Teaching content and methods for software testing is needed to be discussed. In many universities, most of the test course focused theory to explain and test methods, ignoring an important practical aspect. In fact, practice teaching in software testing courses is very important. If you do not experiment and training aspects of the support, just to explain test theories and methods, students will produce conflict and emotional weariness, affecting the effectiveness of teaching. Students of software testing attitude affect their learning of software testing technology. Usually, some companies tend to let the new staff and programming capabilities poor personnel engaged in software testing, which makes one-sided, that many students will not be programmed talent in software testing, so do not attach importance to the study and training of software testing technology. Better teaching case and experimental environment are the key teaching good software testing course, which is one of the important factors affect teacher teaching enthusiasm. Combined with other courses are important ways and means for students learning software testing.

Colleges and universities in the software testing personnel training, there are still many problems and deficiencies, including the following aspects:

1. Education seriously out of line with market demand.
2. Lack of deep understanding of software testing.
3. Lack the necessary real experimental environment.
4. Lack of team awareness training.

## 5. Teaching materials and teaching methods behind.

Based on the above analysis and problems, we have a number of reforms and practice in software testing course.

### Clear objectives and improve the organization of the curriculum content

According to our school, our goal is for the community to develop the application-oriented person. For software testing course, our the goal is to enable students to understand the important role and status of software testing in the software development process through the theoretical study of software testing technology and system training. This course can be able to use the software testing technology to solve practical problems and can understand the requirements of software testing professional characteristics and quality of software testers. According to this target and the actual situation, we reform teaching content and teaching resources construction etc. Software testing course content should reflect impart knowledge and ability to improve unity, focus on the development of ability. In the teaching content, we based on the basic theory of software testing, introduce case teaching, supplemented by discussion with presentation, and emphasize the practical teaching. The contents are divided into classroom teaching, experimental teaching and curriculum design of three parts, case-based teaching introduces in the teaching process, and increase the supporting experiments and curriculum design. To tie in with classroom teaching, strengthen after-school guidance and practice, we wrote software testing technology experimental guidance and software testing technology curriculum design guide book and other materials. In order to protect the software testing course level, to improve the effectiveness of teaching, we use case-based teaching method. In order to ensure the effectiveness and feasibility of the method and to obtain the desired effects in the classroom teaching, we collect many teaching cases. software testing course structure as Fig 1.

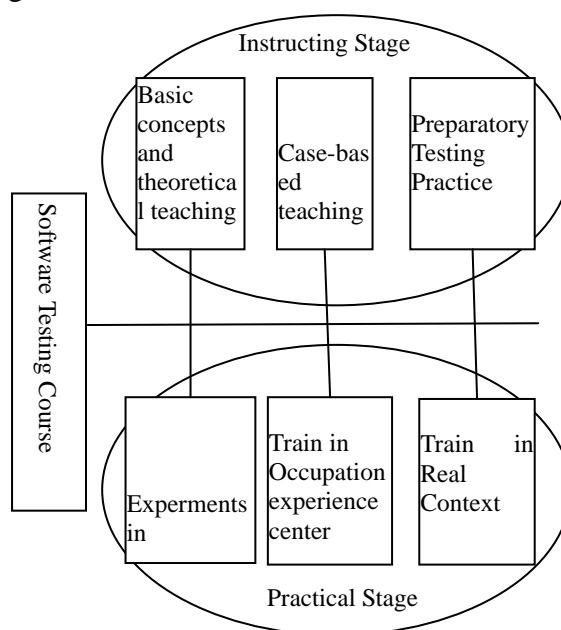


Fig.1.software testing course structure

### Establish Occupation experience center

School training and off-campus practice base has its own characteristics and advantages, but there are inadequate. School training is difficult to form a good working environment. The main work of the school is knowledge transfer and capacity-building. There is a gap between school experimental training demonstration, simulation and validation experiments and the actual work environment, is difficult to meet the training requirements of the software developers. In off-campus base practice, the enterprises not pay enough attention to students, the lack of guidance

system for students and staff of enterprises is relatively loose management of students.

These phenomena affect the quality of the students' training; the establishment of vocational experience center is an effective way to solve these problems. Occupation experience center is different from both internal and external practice training base, and both internal and external practice training base advantages. The center is richer in content and has better effect. Student occupation experience center main task is to provide the various positions of the stages of software development. College students' social practice, professional practice, curriculum design, graduate design and other related tasks can be carried out in occupation experience center. The center consists of a plurality of open laboratory, undertake social demand and practice for undergraduate student. The center can promote college student's employment-oriented exercise based on project-driven and professional roles experience capacities. Software testing laboratory is the main laboratory of the center. We can do many testing project in the laboratory.

### **Combined with other courses**

In order to better learning software testing courses, some other courses must be combined.

In our practice, some of content in introductory programming courses integrated software testing knowledge. Usually introductory programming courses will be arranged early during university study, so the students can early know some software testing technology. This is not only for students good programming, but also conducive to cultivate students' interest in software testing. For the purpose of teaching programming, we recommend as appropriate the use of white-box testing. Many test case generation methods can be applied in the course. In accordance with the above, we recommend that students will be explicitly taught the dynamic testing. We can run the test cases to verify the program is correct. During the examination, we can also run the test case to determine student mark, For example, 80% of the test cases passed, and the results can 80. This examination method is very objective and fair, and encourages students to consider all aspects when writing programs. We use the NUnit environment for the purpose of teaching programming in c#. NUnit is a freely available, open source product that provides a testing framework and test runners [6, 7]. Every test can be added to one or more categories, to allow running them selectively. Use the tool in the process of teaching, not only learned to use the tool self, but also to make the students more focused on the design and organization of the test cases and improve the learning efficiency.

Static testing is a form of software testing where the software isn't actually used. This is in contrast to dynamic testing. It is generally not detailed testing, but checks mainly for the sanity of the code, algorithm, or document. It is primarily checking of the code and/or manually reviewing the code or document to find errors. This type of testing can be used by the developer who wrote the code, in isolation. Code reviews, inspections and Software walkthroughs are also used. We recommend that static testing will be taught, this lead the students to check the code they have written and to write the required documentation.

In software practice course, we also require students to not only the completion of the design and implementation of program there must also complete test of the project. The task of the test score accounted for 30% of the course grade. The course focuses on the testing ability to develop the following:

1. Prepare the test environment and the test scripts or test run (batch test) for running, review expected results against test and acceptance criteria and review results of earlier component testing and ensure critical issues are identified and taken into account.
2. Confirm pre-existing modules and perform static tests of each point of integration and verify correctness of arguments, positional parameters and return values in each integration suite.
3. Select appropriate test tools and run test scripts and document the results against software life cycle model.
4. Ensure that global name space pollution and static variables are specifically addressed for each integration unit in line with test and acceptance criteria, compare test results to requirements on completion of each integration component.
5. Summaries and classify test results and highlight areas of concern, compare the test results

against the requirements and design specification and prepare report, ensure attendees' details/comments are logged and signatures gained.

Observation of real or simulated work processes are include in assessment method and procedures and performance in a project context as well as questioning on underpinning knowledge and skills.

### **Guide students to a correct understanding of software testing and software testing career**

Software testers not only to master the software testing technology, but also have the ability of software systems analysis, software system design and software programming etc. The software tester's work is to identify errors in the software and often communicate with system designers and programmers, so rigorous work habits, good communication skills and team spirit is software testers required. Students' awareness of the importance of software testing technology and employment prospects is an important impetus to inspire and encourage them to take the initiative to learn. In the teaching process should be timely introduce some information of testing and invite experienced engineers to come to the school to report, so that students clearly understand the job requirements and broad prospects for testing, a correct understanding of software testing and software testing career.

### **Conclusion**

Train the student meeting software industry's need is our goal. After analyzing the expectation from software development enterprise and our traditional teaching method, we proposed some reform and practice in this paper. We update the part of the course content and course structure, Makes the content in line with the development of software testing technology and structure more reasonable. Occupation experience center have been established not only to provide students with a good experimental environment as well as professional experience. Experience in occupation experience center also enhances the students' teamwork. Integration and other courses an important method to improve the effectiveness of training.

### **References**

- [1] 2009 Survey of Software Testing professionals in China [EB/OL].[2010-6-26]. <http://www.51testing.com>.
- [2] Chengying Mao. Towards a Question-Driven Teaching Method for Software Testing Course. In Proceedings of the 2008 International Conference on Computer Science and Software Engineering - Volume 05, Vol. 5. IEEE Computer Society, Washington, DC, USA. 645-648.
- [3] ZHAN.G Yinnan,WANG Xiaochi.Implementation of Software Testing Course Based on CDIO.Computer Science & Education (ICCSE 2011)August 3-5, 2011. SuperStar Virgo, Singapore.107-110.
- [4] Garousi, V.. Incorporating real-world industrial testing projects in software testing courses: Opportunities, challenges, and lessons learned Software Engineering Education and Training (CSEE&T), 2011. 396-440.
- [5] Edward L. Jones, Software testing in the computer science curriculum -- a holistic approach, Proceedings of the Australasian conference on Computing education, December 2000, Melbourne, Australia. 153-157.
- [6] Andrew Hunt, David Thomas: Pragmatic Unit Testing in C# with NUnit, 2nd Ed. The Pragmatic Bookshelf, Raleigh 2007
- [7] Jim Newkirk, Alexei Vorontsov: Test-Driven Development in Microsoft .NET. Microsoft Press, Redmond 2004.