I.J.Modern Education and Computer Science, **2012, 2, 52-58**
Published Online March 2012 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijmecs.2012.02.08

Adoption of Employment-oriented Experience Training Program to Improve Undergraduates Employment Competence

Shuang Liu
School of Computer Science & Engineering, Dalian Nationalities University, Dalian, China
Email: dlnuliushuang@163.com

Peng Chen
Department of Computer Science & Technology, Dalian Neusoft Institute of Information, Dalian, China
Email: chenpeng@neusoft.edu.cn

***Abstract*—To enhance undergraduates employment competitive advantages, communication and expression capability and team cooperation ability in engineering practice experiences, an experience training program is adopted to cultivate compound international talents of software development required by software companies. For the training program, knowledge spans and integration extent of teaching contents, implementation flow and flexible evaluation methods are also illustrated in detail. Especially during practical training project selected from software companies, students are guided to cooperate with team members to complete regular software development, which will improve their employment competence and make them fit for their future work more rapidly. With auxiliary experience training online teaching platform, three times of training program has been carried out successfully and a statistics table shows effects of the training program.**

***Index Terms*—Employment-oriented, Experience Training Program, Employment Competence, Software Development.**

I. Introduction

With rapid development of information technology and internet, more and more senior software talents are required by domestic and foreign software companies [1-4]. But due to shortage of project experiences, many undergraduates with computer major cannot meet qualification of practical requirements of companies. How to keep pace with technology development and cultivate compound international talents of software development is the main focus of our experience training program beginning from September, 2009 [5]. As a characteristic and key issue of 2009 training scheme for Dalian Nationalities University, purpose of the training program is to explore and design an effective training system adopting appropriate teaching methods, teaching contents and implementation scheme, which will guide senior students to complete requirements analysis, system design, coding, integration and test for the regular software development as a team [6]. During training, students are required to complete different deliverables and give presentation for tutors in each phase of software lifecycle. By different evaluation methods such as review process, code check or presentation and expression of deliverables, students' professional skill, communication and expression ability and team cooperation capability can be improved, which will in turn enhance their self-confidence and improve their employment competence.

This paper is organized as follows. Section II discusses current situation of experience training program carried out by software companies and universities. Section III introduces implementation details carried out in Dalian Nationalities University and gives teaching contents, implementation process, evaluation methods and teaching effects. Section IV gives the conclusion.

II. Current Situation of Experience Traing Program

As the largest software outsourcing market, many training companies or software companies of Dalian have carried out experience training programs for undergraduates of related computer majors. For example, Neusoft Group Corporation requires each new employee to take the training provide by its training department [7]. And there are many other companies such as Tarena IT Training Group, Dalian software park training center [8] and so on. Certainly, software companies have their training advantages as follows. First, they have plenty of teacher resources with a lot of experiences of practical project development. For medium software companies, they can select appropriate training tutors from development departments. For example, tutors are required to do software development again after 1-2 years training. Secondly, they have plenty of project reserves. During training, they can select some old project as training project. Although the old projects may adopt some technology out of date, it is easy to adjust them into new project with fashionable technology for the training purpose. Because only projects with medium size are needed. Third, tight atmosphere in software companies will put pressure on students. At university, students feel free to learn knowledge with on pressure. They cannot imagine of the employment competence or work pressure.

But in real development environment of software companies, students will feel much pressure on them because all engineers work very hard everyday around them. So they will put all their efforts on professional skill learning for their future work.

But there are some disadvantages of the training program provided by software companies. Whether software development or training program, software companies need profits. So, undergraduates have to pay expensive training fee to pass the training. For undergraduates of Nationalities University, most students come from rural areas and cannot afford extra training fee. Moreover, some software companies just chose some good passing students as tutors and some teaching class consists of 60-70 students to save training cost sometimes. In this case, not every training student can get full guidance and the training quality is not assured. Another deficiency for software companies is their limitation of teaching resources. Only in training class or the training room, students can learn and review related teaching material with local areas networks. Any teaching material cannot be downloaded or copied by floppy disk or emailed by Internet. So training students are limited in a small learning space. Out of the training room, they can hardly do any effective learning.

Some colleges or institutes have carried out experience training program such as Dalian Neusoft Institute of Information. Most of these universities are independent and skill training is their education purpose. So training in these colleges cannot get satisfactory effects because their students lack study interests and subjective initiative. Besides that, they require all undergraduates to participate in the training, which will certainly lead to teaching problems and study problems.

Based on summarization of training experiences in software companies and independent colleges, we carried out our experience training program from 2009. More emphasis of education in our university is put on teaching quality than profit. In comparison with software companies or independent colleges, there are many advantages of the experience training program in our university for students, which are summarized as follow.

(1) Web resources are open and students can search and get many tips from the web. We don't have to limit access of the Internet because we don't have any business secrets.

(2) Each teaching class is selected from candidates. Each student needs to pass the entrance test before they take the training class. Only students with some fundamental knowledge can complete the whole training process. This not only limits the size of training class, but also assures the training quality.

(3) Teaching resources are open and can be accessed in the special training room or dormitories with campus network. Students can copy courseware and sample codes and save any code fragment or document in their computers.

(4) Teaching process is students-centered. One teaching class has 30 students at most. So tutors can give guidance to any student freely during training process. Study pressure put on the students is little by little until they realize the serious employment status.

(5) The management is flexible. Teachers and students know each other very well and they can exchange advices on how to make the training process more effective.

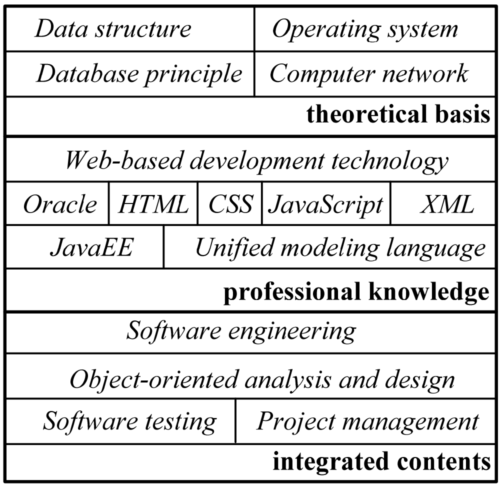
(6) Training program is just like other selected courses without any extra training fees. Students with intension to improve their engineering abilities can select the training program. They don't have to pay any extra fees for the course. Whether they pass or fail the training program is decided by tutors, not by training fees.

III. Implementation Details for The Training Program

From September 2009 to September 2011, we have finished three times of training program for computer major of grades 2006, 2007 and 2008 [6]. In September 2009, we didn't have enough teaching auxiliary website, so we only chose 20 students to finish Java-oriented training program. In 2010 and 2011, we carried out .Net-oriented training program and Java-oriented training program based on our purchase of online training system. With this online system, we can not only open and close teaching materials, but also provide various multimedia coursewares for students to learn. We keep adjusting our original teaching plan, teaching contents and the corresponding evaluation methods to improve our training quality for good employment rates. Now we discuss our experience training program from teaching contents, implementation process, and evaluation methods to teaching effects.

A. Teaching Contents

As an integrated engineering practice program, students are required to complete an engineering project within a working team with comprehensive application of the following knowledge listed in Fig. 1.



Data structure | *Operating system*

Database principle | *Computer network*

theoretical basis

Web-based development technology

Oracle | *HTML* | *CSS* | *JavaScript* | *XML*

JavaEE | *Unified modeling language*

professional knowledge

Software engineering

Object-oriented analysis and design

Software testing | *Project management*

integrated contents

Figure 1. Knowledge span and integration extent for teaching contents

From Fig. 1, we can see it not only includes theoretical basis such as data structure and operating system, database principle and computer network, but also includes Java based Web development professional knowledge such as Oracle, HTML, CSS, JavaScript, XML, especially JavaEE [9] related skills. Besides, students must learn to integrate knowledge of requirements analysis and design for different phases of software life cycle. For example, they have to deliver requirements analysis specification based on object-oriented analysis with UML as description tool in requirements analysis. And they need to provide test plan and test results in unit test and integrating test. For training preparation, tutors will review related basic theory and concepts for students. Students are required to understand the knowledge and do some coding exercises. After training preparation, students are required to complete a selected project with application of all knowledge mentioned above. Within a working team, they have different working tasks based on different roles in the team. In this process, tutors will give guidance and code help.

Concrete teaching materials focus on the following three parts with keeping pace with the latest development technology and can be classified into three kinds. The first kind is basic knowledge, which includes HTML, CSS, JavaScript, Ajax, Java basis, JavaEE, Servlet, JSP, Tomcat, JDBC, MVC, XML, Oracle, SQL and related use of development tools such as Eclipse and PLSQLDev. Students have learned most of the knowledge, so tutors only give outline of each knowledge points. The second kind is framework knowledge, which includes Struts, Spring and Hibernate. Students only learned limited knowledge about frameworks, so tutors need to put many efforts on framework principle and flexible application of frameworks in development to meet requirements. The third kind is development knowledge, which includes pages prototype design, detailed design, database design, coding specification, test and related project management knowledge. In project training process, students will learn the corresponding skills before each stage of software lifecycle. Then they have chances to practice and put them into real application.

B. Version Control Based on SVN

Here, we use SVN as version control tool. Subversion (Abbrev. SVN) is a free/open-source version control system. That is, SVN manages files and directories over time. A tree of files is placed into a central repository. The repository is much like an ordinary file server, except that it remembers every change ever made to the files and directories. This allows users to recover older versions of the data, or examine the history of how the data changed. SVN is a centralized system for sharing information. At its core is a repository, which is a central store of data. The repository stores information in the form of a file-system tree. Any numbers of clients connect to the repository, and then read or write to these files.

Basic running principle of SVN is copy-modify-merge, which is illustrated in Fig. 2 and Fig. 3.

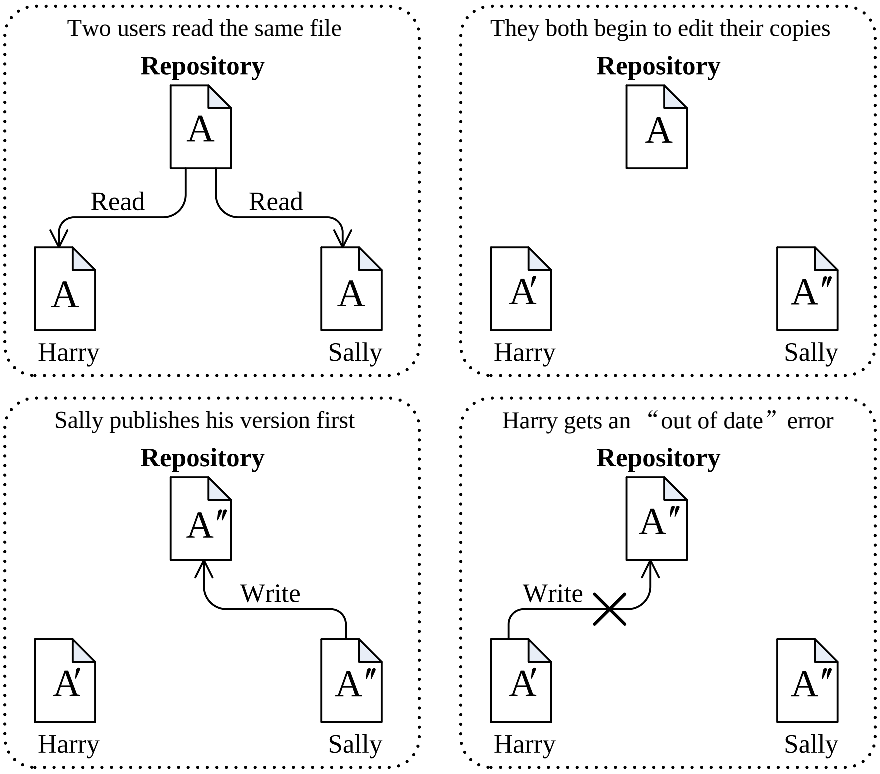

Two users read the same file
Repository
A
Read
Read
A
A
Harry
Sally
They both begin to edit their copies
Repository
A
A′
A″
Harry
Sally
Sally publishes his version first
Repository
A″
Write
A′
A″
Harry
Sally
Harry gets an “out of date” error
Repository
A″
Write
A′
A″
Harry
Sally

Figure 2. Principle of version control in SVN

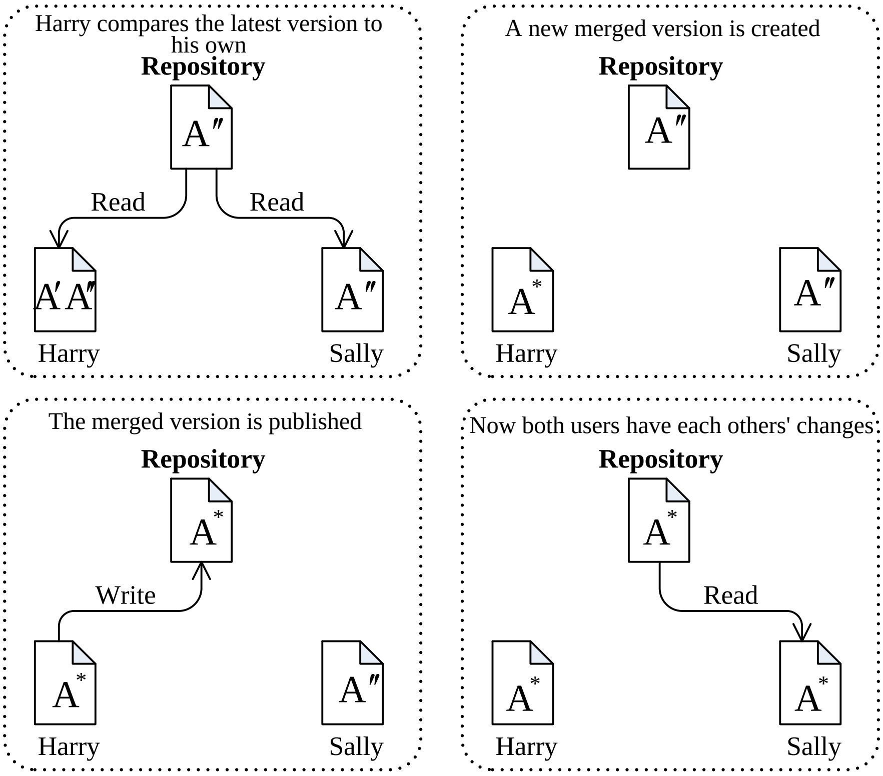

Harry compares the latest version to his own
Repository
A″
Read
Read
A′A″
A″
Harry
Sally
A new merged version is created
Repository
A″
A*
A″
Harry
Sally
The merged version is published
Repository
A*
Write
A*
A″
Harry
Sally
Now both users have each others' changes
Repository
A*
Read
A*
A*
Harry
Sally

Figure 3. Principle of version control in SVN (continue)

Here, we explain how different team members work together to complete one project. As shown in Fig. 2, team members Harry and Sally read the same code file to their local computers from the repository. Then they both begin to edit their copies. After one day's work or one stage of work, Sally publishes his version first. At this time, if Harry wants to publish his version too, or want to write his version into the repository. He will get an out of date error. What Harry has to do is to read the latest version and copy it to his local computer again and compare his own version with the new version, as shown in Fig. 3. Then he can publish his merged version with assurance of no code conflicts. Now the repository saves both team members' changes with no code conflicts. To assure saving different copies of code at any changing time and providing team cooperation with little user interference, we adopted SVN as code version management tool.

C. Implementaion Process

Based on teaching contents listed above, we divided our training process into two stages [6]. In first stage of preparation, we finished work from phase 1 to 3 showed in Fig. 4. In second stage of practical project training, we finished work from phase 4 to 12. In Fig. 4, we illustrated implementation details for students and tutors.

In phase 1, we chose some qualified candidate students based on written theory test and programming test with computers because not all students can follow the whole training process. The selected students can keep pace with tutors' training process with full preparation and self-study.

In phase 2, tutors gave synopsis explanation of fundamental knowledge. Although students have learned the needed knowledge independently, they can't combine them together to solve specific problems. By this review process, students can master key idea of the knowledge and learn to solve one problem with integrated knowledge points.

In phase 3, students are required to practice the small item with integrated frameworks and related professional knowledge. Students must have a good command of coding because software companies put much emphasis on practice ability when they recruit new members.

In phase 4, all training students are divided into different project teams with free combination. Then tutors may adjust team members to assure the reasonability of each team. Each working team is composed of team leader, configuration manager, review leader, database manager and test leader, which cooperates together to complete the project development.

In phase 5, students need to fully understand requirements of the selected project based on the company's business. Tutors will elicit key functions for each project, and lead each group to complete their requirements understanding. Then students are required to design pages prototype with HTML, JavaScript and CSS or one visualization toolkit such as Visio and deliver their design results.

In phase 6, each team needs to present their deliverables for tutors to decide whether their deliverables meet software requirements. This process is called review, which will be illustrated in the next topic.

In phase 7, students are required to complete object-oriented classes design, database design and the detailed design specification writing. Only with full comprehensive application of the related knowledge can students finish the tasks. And students must upload their specifications as design results online.

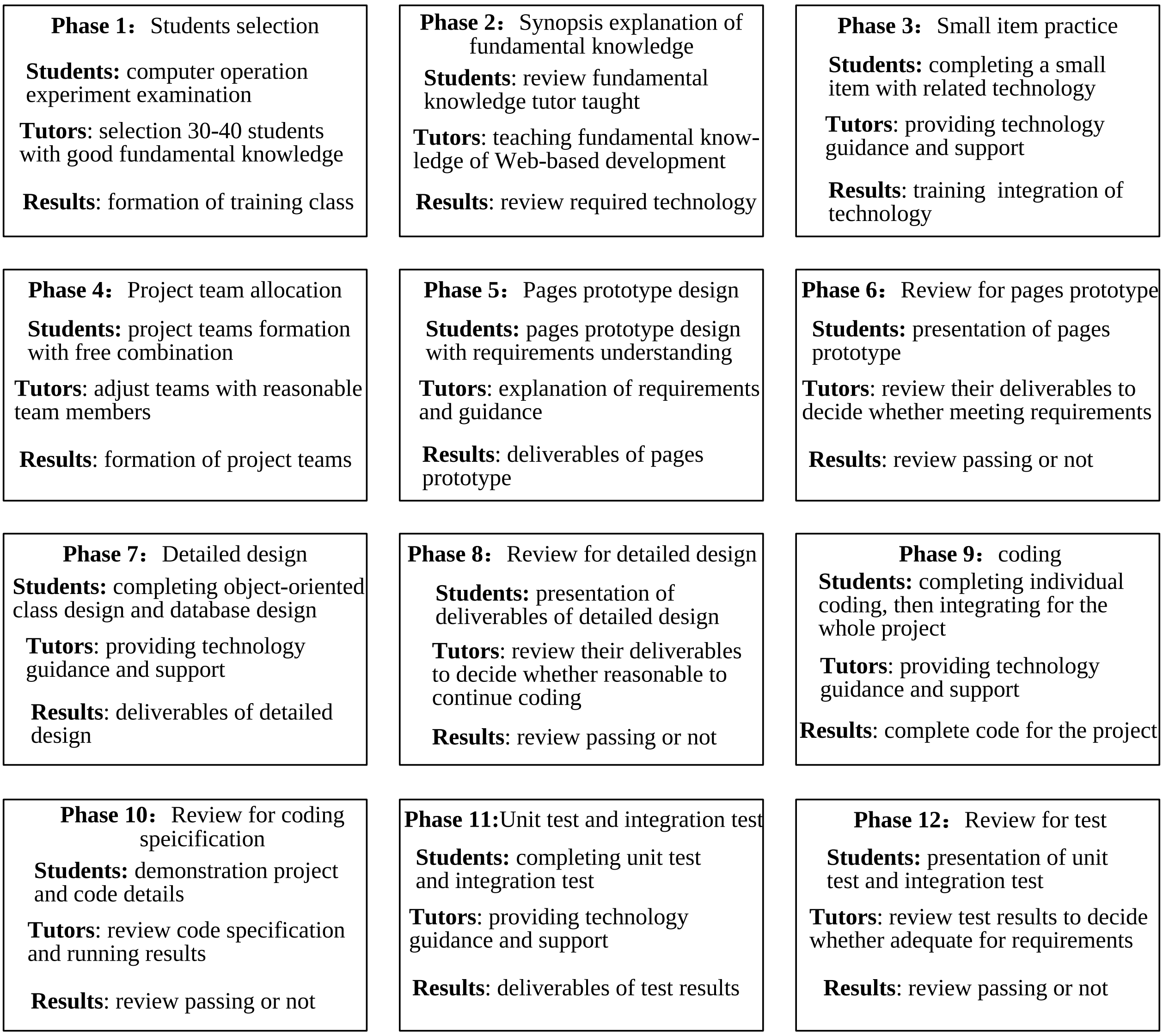
In phase 8, similar to the previous review process, students' achievements have to be reviewed by all tutors to decide whether the design needs modification.

In phase 9, in coding stage, tutors must put much energy into guidance or complete the work helped by excellent students as teaching assistants. Because most of our students will become coding engineers not architecture engineers, so students have to put much emphasis on the coding skills. Not only the required functions needs to be realized, but also the code normalization has to be reviewed. Here, we use SVN mentioned above as version control tool.

In phase 10, tutors will review code specification and running results of the project.

In phase 11, students need to test the units by JUnit or other tools and test the integrated software using black-box method or some test toolkit such as LoadRunner.

In phase 12, if the software is not qualified in test review, students have to modify the code and test it again. If the review is passed, students can summarize what they have done in the whole training process and what they have learned and should improve in the future work.



Phase 1：Students selection
Students: computer operation experiment examination
Tutors: selection 30-40 students with good fundamental knowledge
Results: formation of training class

Phase 2：Synopsis explanation of fundamental knowledge
Students: review fundamental knowledge tutor taught
Tutors: teaching fundamental knowledge of Web-based development
Results: review required technology

Phase 3：Small item practice
Students: completing a small item with related technology
Tutors: providing technology guidance and support
Results: training integration of technology

Phase 4：Project team allocation
Students: project teams formation with free combination
Tutors: adjust teams with reasonable team members
Results: formation of project teams

Phase 5：Pages prototype design
Students: pages prototype design with requirements understanding
Tutors: explanation of requirements and guidance
Results: deliverables of pages prototype

Phase 6：Review for pages prototype
Students: presentation of pages prototype
Tutors: review their deliverables to decide whether meeting requirements
Results: review passing or not

Phase 7：Detailed design
Students: completing object-oriented class design and database design
Tutors: providing technology guidance and support
Results: deliverables of detailed design

Phase 8：Review for detailed design
Students: presentation of deliverables of detailed design
Tutors: review their deliverables to decide whether reasonable to continue coding
Results: review passing or not

Phase 9：coding
Students: completing individual coding, then integrating for the whole project
Tutors: providing technology guidance and support
Results: complete code for the project

Phase 10：Review for coding speicification
Students: demonstration project and code details
Tutors: review code specification and running results
Results: review passing or not

Phase 11: Unit test and integration test
Students: completing unit test and integration test
Tutors: providing technology guidance and support
Results: deliverables of test results

Phase 12：Review for test
Students: presentation of unit test and integration test
Tutors: review test results to decide whether adequate for requirements
Results: review passing or not

Figure 4. Detailed implementation process for the training program

D. Flexible Evaluation Methods

For some basic theory, we integrated traditional written examination and online test on Black Board [10]. Students are required to master some basic theory for entrance test of software companies. So written examination is necessary and irreplaceable. Some coding technique needs to be mastered by solving specific problems, so online test is also necessary.

For different stages of software lifecycle [11], we adopted peer review as the evaluation approach referenced by many software companies, that is, course tutors group decides whether the achievements is qualified. Team leader of each group needs to present their deliverables in Chinese or in English and answer all questions or ambiguity for the deliverables. If the review is not passed, students have to modify their deliverables and repeat the review process. If the review is passed, students can go to the next step of the training process. For example, students will deliver detailed design specification after they finished object-oriented design and database design. Similar to the review process mentioned above, the achievements have to be reviewed by all tutors to decide whether the design needs modification.

To provide more flexibility for training tutors, we designed various evaluation methods shown in Table I. By these various and flexible evaluation methods, students not only practice their expression ability both in written and spoken, but also master the most important development technology and basic theory.

TABLE I.
EVALUATION METHODS FOR DIFFERENT TRAINING STAGES

Training Stage	**Evaluation Contents**	**Evaluation Methods**
After phase 3	technology basis	coding test and technology basis exam
After phase 9	development technology	review deliverables and project running results
After phase 11	project development	review deliverables and project quality
After phase 12	integrated technology	coding and all related technology exam

E. *Teaching Effects*

During training, we focus on hierarchical teaching and team cooperation to get good teaching effects.

For hierarchical teaching, we divided the selected students into different teaching class based on their practical ability and choose different framework knowledge. For students of different levels, we also adopted different evaluation methods. For the low level students, the corresponding evaluation is much easier compared to the high level students.

For team cooperation, we organized students into different project teams to finish the complete requirements understanding, pages prototype design, detailed design, coding and testing as a whole. Only one member can hardly finish the work. To cooperate with other members, students must express their ideas and communicate with different kinds of people. We put much emphasis on team cooperation because all employment for software companies emphasizes the importance of team cooperation, but all daily teaching can hardly train our students' cooperation capability.

In this three years implementation of the training program, we optimized our teaching contents and improved our teaching plan. We summarized our training results of Java-based training program for major of computer science and technology shown in Table II. For each row, the former number represents the training results before symbol '/' and the latter number represents statistical data for students of the corresponding grade after symbol '/'.

TABLE II.
TEACHING EFFECTS FOR THE LAST THREE YEARS

Criteria	**2009**	**2010**	**2011**
# students	20/120	30/128	30/122
employment rates	100%/ 90%	100%/ 93%	100%/ 94%
acceptance rates of well-known companies	95%/ 72%	98%/ 83%	95%/ 66%
salary level	3000/ 2500	4000/ 3000	6500/ 4000

As shown in Table II, we got better employment rates for the training students after two months intensive training than common undergraduates. They not only got fully employed, but also joining well-known software companies with high salary such as Baidu, Alibaba, Tecent and so on.

IV. CONCLUSIONS

Key education purpose of our nationalities university is to promote employment rates for the undergraduates. So setting up employment-oriented experience training curriculum in the university is urgent and necessary. We have set experienced training course as the elective course for the undergraduates with computer science as major since 2009.

Here, we summarize the last three years of training experiences and compare training advantages and disadvantages between software companies and university. Then we put forward a feasible training system with detail discussion of the concrete implementation process, teaching organization method and evaluation methods. The innovative points of the training program are peer review evaluation methods and practicability for promotion of employment rates.

With more experimental project and accumulated experiences, our employment-oriented experience training program will play more and more important role in enhancing student's professional skills, developing students' project practical ability and communication and expression ability. By our training program, employment competence of graduates from nationalities universities especially for international talents will be promoted a lot in the future.

Acknowledgment

This work was supported by ideological and political special project of State Ethnic Affairs Commission (No. C-1-6) and by the Fundamental Research Funds for the Central Universities (No. DC110306).

References

[1] C. Michael, "Industry-oriented software education in practice: a case study," *Proceedings of 2nd China-Europe International Symposium on Software Industry-Oriented Education*, Harbin, China, January 2006, pp. 78-82.

[2] C. Michael, "Experiences in industry placement-shaping the student's Mind in an industry-oriented software engineering curriculum," *Acta Scientiarum Naturalium Universitatis Sunyatseni*, vol. 146, no.12, pp. 83-85, 2007.

[3] L. Q. Ye, R. H. Liu, "Experience training program with level-division: practical exploration of talents training mode for normal education," *Education Exploration*, no. 2, pp. 62-63, 2009.

[4] S. Liu, X. Z. Li, P. Chen, "Research on Employment-Oriented IT Experience Training Program," *Computer Education*, no. 9, pp. 140-143, 2010.

[5] Dalian Nationalities University, *2009 Teaching Outline of Dalian Nationalities University*, 2009.

[6] S. Liu, X. Z. Li and L. Zuo, "Applying Bilingual Teaching on Employment-Oriented IT Experience Training Program in Nationalities Universities," *Theory and Practice in Language Studies*, vol. 1, no. 4, pp. 426-431, 2011.

[7] Neusoft Group Corporation, *Neusoft Talents Training Center*, http://ttc.neusoft.com/.

[8] Dalian Center of Tarena Group Corporation, *Tarena IT Training Center*, http://www.dltarena.com/.

[9] SUN Microsystems, *Java EE 6 Technologies & Specifications*, http://java.sun.com/javaee/6/docs/tutorial--/doc/index.html.

[10] Dalian Nationalities University, *Blackboard Academic Suite*, http://bb.dlnu.edu.cn.

[11] R. S. Pressman, *Software Engineering: A Practitioner's Approach*, 6th edition, Boston, MA: McGraw Hill Higher Education, 2005.

Dr. Shuang Liu is an associate professor at the Scool of Computer Science & Engineering, Dalian Nationalities University, Dalian, China. For teaching, she has been focusing on theory and application of software engineering and experience training program. For research, her current research interests include intelligent information process and machine learning algorithms. She received her Ph.D degree of Traffic Information Engineering & Control from Dalian Maritime University in 2006. She is a member of WASE (World Association of Science Engineering).

Mr. Peng Chen is an associate professor in Computer Science at the Department of Computer Science and Technology, Dalian Neusoft Institute of Information. In teaching, he has been focusing on applying CDIO concepts into Computer Science Education. In research, his current interests include cloud computing and Java-based development. Mr. Chen received his Master degree in Computer Science from Liaoning ShiHua University.