

Occupational Ability Oriented Graduate Education in Software Engineering

<http://dx.doi.org/10.3991/ijet.v10i8.5214>

Qinghong Yang¹, Wen Lu¹, Pengfei Feng¹, Haijun Wei², Zhichao Cheng^{1*}

¹ BeiHang University, Beijing, China

² 5Th Middle School, Daxing District, Beijing, China

Abstract—The background of this paper is the education reformation of Chinese software engineering. Using graduate education of Beijing University of Aeronautics and Astronautics (BUAA) as an example, we will demonstrate the development of Software Engineering training curriculum based on the job position of a company. Firstly, we analyze the relationship between the occupational goal and the ability needed; secondly, we set the goal of talent cultivation based on occupational ability needed; lastly, we construct cultivating system based vocational ability. In conclusion, we should improve the managing system of practical education and the mechanism should focus on constructing occupational abilities of students that various companies require.

Index Terms—master degree education, software engineering, occupational ability

I. INTRODUCTION

With the rapid development of the Internet, mobile technology and software outsourcing in China, IT industry is facing enormous market demand as well as good prospects [1]. This demand creates a strong job market for IT industry. China has established 35 software schools to cultivate Software Engineering Masters since 2002. Taking Software Engineering Master's education reform of BUAA Software School as an example, we will talk about how to establish talent cultivation mechanism based on cultivating occupational ability by doing research with cultivation of Software Masters [2].

Nowadays, there's a common phenomenon that lots of job seeking students cannot find a suitable job as well as lots of companies cannot find the right staff. It is caused by the mismatch between the current situation of engineering education and its target market. It is very important for schools to innovate education models for cultivating qualified students who will match the companies' job demand, meet the study target for themselves and the education target from school. Thus we must formulate education principles for the specified target to reach a win-win-win result among students, companies and schools [3].

We'll find out which knowledge and skills match the job position as per the requirements and analytic for real market [3]. Then according to these knowledge and skills, we finally determine the theoretical and the practical courses to satisfy the learning target – theoretical courses help students accumulating knowledge and practical courses help learning skills [4]. Therefore the curriculum system and cultivation mechanism which are based on the cultivation of occupational ability will be constructed.

II. ANALYZING THE RELATIONSHIP BETWEEN OCCUPATIONAL GOAL AND ABILITY

Ability refers to an essential subjective condition to successfully achieve one's goal [5]. It is a mental characteristic of individual which immediately effects on the efficiency of one's activity and achieve one's goal [6]. The cultivation for software engineers is related to the occupation, the occupational planning and the occupational abilities. One's career means all work-related experience in one's entire life. It's also considered as the paths of the job promotion in different organizations [7].

Talents in IT corporations may work as system analyst, system architect, software development engineer, software quality engineer or software project manager according to the software engineering life-cycle activities [8]. The cultivation for these five kinds of talents covers the entire life-cycle of software development.

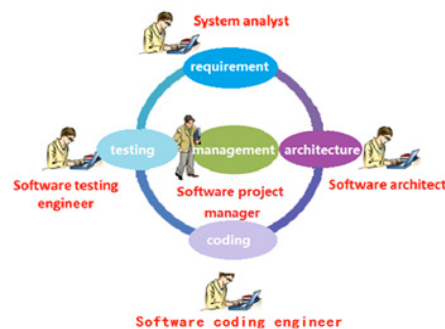


Figure 1. Designing talent cultivation target according to software engineering life-cycle.

As you can see from Fig. 1, the system analyst focuses on software requirements. The system architect takes charge of design. The senior software developer is responsible for core functional development. The senior software quality engineer is in charge of software testing. The project manager directs the affairs of the entire software life-cycle.

It is necessary to cultivate software engineers' knowledge and skills for software life-cycle activities from requirements analysis to software maintenance. It is significant to combining software theory and practice, acting on international convention and collaborating with the companies. We must cultivate students' design ability for software analysis, design, architecture as well as management in order to make them able to master overall project planning and management.

To give students the desired skills and knowledge mentioned above, software engineering education should cover both cutting-edge theory and application of software engineering. So that software engineering students can really improve themselves on the development and process management for actual projects.

III. SETTING THE GOAL OF TALENT CULTIVATION BASED ON OCCUPATIONAL ABILITY

We should confirm what kind of talents the companies need by analyzing the needs from IT industry. Thus we can realize the essential knowledge and skills that are necessary for talents, and then set the goal of talent cultivation. After that, we will analyze the relevant courses and activities for talent cultivation; plan the cultivation according to students' different situation; implement the plan by synthesizing all existing resources; and finally cultivate suitable graduates for companies. After 10 years' effort, our software college has summed up a talent cultivation mechanism for cultivating talents that meet the needs of companies. As we can see in Fig. 2.

According to the above cultivation process, we will first analyze the occupational abilities for major positions in software engineering. Secondly, we will filter the occupational abilities but leave their essentials. We analyze the essential abilities and skills required by different fields of software engineering to create assessment norms for each of them. Finally, but leave their essentials. The old courses curriculum, we will try to set a new cultivation goal and design a curriculum system for talent cultivation of software engineering.

The goal for talent cultivation of software engineering is to cultivate a large number of talents who are capable of combining and using knowledge of engineering, marketization and internationalization for the society. This requires our graduates to acquire four abilities: basic knowledge of engineering, personal ability, team work ability and engineering ability [9]. Students will reach the expected goal in these four aspects after cultivation by software colleges. It makes students to promote their ability in software development, design and architecture, their innovating ability, their teamwork ability, leadership, communication skill and their English.

We could assess how each step in our education practice affects students by constructing a talent cultivation mechanism. We can reach our goal of cultivating talents by focusing on cultivating students' ability in system engineering, especially their abilities in conception, design, development and implementation of software projects. Self-learning, communication and cooperation are also abilities we want to teach the students.

IV. SOME COMMON MISTAKES

A. Vocational ability-based cultivating process

After carefully analyzing vocational ability, BUAA software college decided that the cultivating architecture of software engineering master should include four parts, namely theory education, practical education, degree thesis, and company internship. In addition, academic forums, academic salons and company open days are held to inspire students and give them chances to learn more about companies, learn academic trends and build humane accomplishment. As we can see in Fig. 3.

B. Vocational ability based course architecture

Specialization cultivating programs should take many factors into consideration [10]. First, there are various sources of students. Some of our students have worked for a long time and some of them just earned their bachelor degrees. Second, cultivation mechanisms are different, some of students take full-time education and some of them take courses after their work day. Their purposes for taking courses are also different. Therefore, the whole cultivation program should base on the principle "little core and big periphery", we formulate the essential courses of each major while providing different optional courses to these majors.

1 Basic Core Courses

Basic core courses are required courses that designed for all the students of a major, and their aim is to cultivate students' basic quality, basic specialized knowledge and practice ability. The design of curriculum is 4+3+4, namely 4 basic ability courses, 3 basic specialized courses and 4 practice ability courses. The architecture is like Fig. 4.

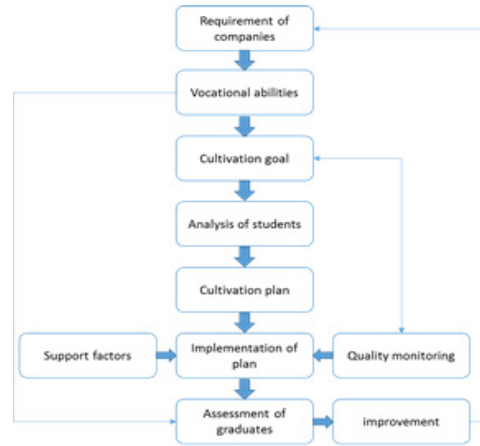


Figure 2. Occupational ability based talent cultivation mechanism of software engineering

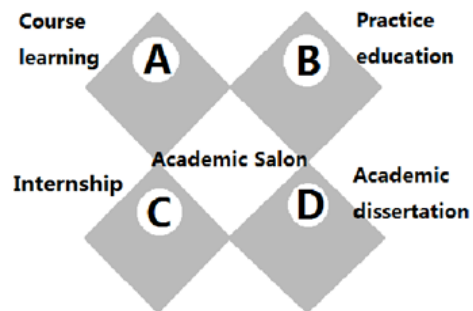


Figure 3. Vocational ability based cultivating module

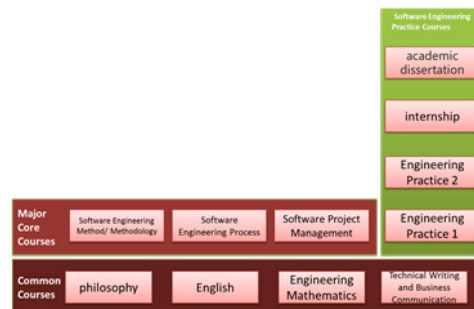


Figure 4. Basic core courses design

2 Periphery courses aimed at different cultivation purposes.

The design of periphery curriculum is 3+N+M, namely 3 required specialized skills courses, N career development courses and M expanded courses. Required specialized skills courses are courses that are necessary for the career development related to one major, these courses focus on the core specialized quality and ability of students, and students should pick out Three of this course type. Career development courses are courses. That focus on cultivating students' specialized skills and ability of specific fields; Students can choose N such courses. Expanded courses are courses aimed at expanding students' horizon and scope of knowledge; students can choose M such courses. The overall design of curriculum is shown in Table I.

Course architecture includes three Phases, they are phase 1(S1), phase 2(S2), phase 3(Y2). Phase 1: The purpose of this phrase is introducing software development to students and let them feel programming is funny, so they can have the faith and confidence to receive further study. We require students to have a perceptual understanding of programming and develop their logical thinking ability. Phase 2: In this phase, students are required to form groups to complete their projects, they can learn how to cooperate with each other and how to manage a team in

this phase. Phase 3: In this phase, students are trained to manage the abilities which are required by companies. Students are asked to analyze, solve and summarize problems. They will learn the typical business process, mainstream technology and develop software according to company rules. They develop software in a manner similar to those already employed by companies, and this really benefits them when they go to work for an employer.

V. SUPPORT FACTORS OF CULTIVATING PROCESS

A. Excellent faculty members

Besides teachers who do specialize in education, our college also hires many experienced academics, engineers and entrepreneurs as tutors and instructors in the practice process. They are also asked to present courses covering advanced technology and case projects and so on. Taking the unique characters of software engineering into consideration, BUAA software college hires both academic teachers and teachers from companies. Academic teachers focus on theory courses to give students a complete knowledge system [12].

Teachers from companies focus on practical courses, and they teach programming and project skills which can meet the practical demand of companies.

TABLE I.
TYPE SIZES FOR CAMERA-READY PAPERS

| | | | |
|---------------------------|--------------------------|--------------------------------|--|
| Compulsory Courses | Common Course (4) | | Engineering Mathematics, English, philosophy, Technical Writing and Business Communication |
| | Major Core Courses (3) | | Software Engineering Method / Methodology, Software Engineering Process, Software Project Management |
| | Practice Courses (4) | | Engineering Practice 1, Engineering Practice 2, internship, academic dissertation |
| Elective Courses | system analyst | specialized skills courses (3) | object orient analysis & design, software architecture, software requirement engineering |
| | | Career development courses (M) | Advanced system analysis and design, Empirical software engineering, domain software design, analysis and design of software testing project, software quality management, Software Metrics |
| | | Expanded courses (N) | Seminars for Frontier Academic Studies, international software outsourcing |
| | Software Architect | specialized skills courses (3) | object orient analysis & design, software architecture, Advanced database technology |
| | | Career development courses (M) | Advanced system analysis and design, Empirical software engineering, domain software design, theory and practice of SOA, software requirement engineering, software quality management |
| | | Expanded courses (N) | Seminars for Frontier Academic Studies, Data warehouse and Data mining,... |
| | software engineer | specialized skills courses (3) | object orient analysis & design, software architecture, Advanced database technology |
| | | Career development courses (M) | Normalized software design and practice, Advanced system analysis and design, J2EE enterprise-level application development, practice of VSTS software development, Android development, software requirement engineering. |
| | | Expanded courses (N) | Seminars for Frontier Academic Studies, Data warehouse and Data mining. |
| | software test engineer | specialized skills courses (3) | Software quality management, Software Testing Technology, analysis and design of software testing project. |
| | | Career development courses (M) | Object orient analysis & design, normalized software design and practice, software architecture, software requirement engineering, Software Testing Tools and Practice, Software Metrics. |
| | | Expanded courses (N) | Seminars for Frontier Academic Studies, international software outsourcing,..... |
| | software project manager | specialized skills courses (3) | Software architecture, software quality management, Software Development Management. |
| | | Career development courses (M) | normalized software design and practice, object orient analysis & design, Human Resource Management, software requirement engineering, Software Metrics, Software Testing Technology |
| | | Expanded courses (N) | Seminars for Frontier Academic Studies, entrepreneurial management. |

TABLE II.
COMPREHENSIVE ASSESSMENT OF BUAA SOFTWARE COLLEGE

| Student type | Number | Employment rate | Salaries per month (Yuan) | Satisfaction degree Of companies(0-5) |
|---------------|--------|-----------------|---------------------------|---------------------------------------|
| undergraduate | 43301 | 97.24 | 2500-7500 | 4.11 |
| graduate | 56792 | 96.84 | 6500-13000 | 4.23 |

B. Multi-layered security system

To reinforce students' software-development ability, BUAA software college instituted a basic practice department which is responsible for the research and construction of related course architecture. The specific jobs of basic practice department. We also need a multi-layered security system to guarantee the process of practice, include:

Experiment system. Different from traditional labs, our labs are instituted with the participation of companies. Companies will provide research directions and research projects to these labs, then engineers, teachers and students will work together. These projects are always foresighted, ample and novel. The fruits of these projects are shared by our college, students and companies.

Practice base. Practice base is a key point of practical education. The amount and quality of talent bases should meet the demand of our students [11]. We now have both domestic and foreign-based as our practice bases. The formation of a practice base should follow the principle "send qualified students, meet the standards of the company, keep in communication, and serve the students".

VI. CONCLUSIONS

A. The result of national model software colleges

After ten years' effort, national model software colleges have cultivated about 100,000 graduate students, and 98 percent of them found jobs immediately after graduation. Software Association of China conducted a third-party survey of all the model software colleges and use norms including 1, is the knowledge taught in school useful for students in companies 2, students' programming ability 3, technology innovation ability 4, English ability. Full score is 5, and the comprehensive assessment is showed in the Table II above.

B. Several problems during the teaching reform

Of course, we also faced some problems during our teaching reform, which should be taken into consideration and solved in the future.

1. To make course arrangement more reasonable. Course arrangement should meet the real need of companies, and abide by the requirement of software engineering. The content of course should change flexibly according to the development of technology in companies.

2. We should teach our students the most advanced and useful technologies. Of course, this requires us to quickly design and create our courses. This raises the question: How do we guarantee the quality of our courses?

3. Accelerate the construction of talent pool in the practice base. We should classify and record students' information according to their majors and abilities into a database, in order to track students' developmental situations and give them appropriate feedback about their future career;

4. We should also classify and record personalities and majors of teachers both from our college and practice base into a database, so we can pair students and teachers more rationally; certainly, we should classify and record the managers of companies into the database too. Thus, we can hire suitable people to train students' management ability and build a database of the information of eminent engineers in companies, so as to improve our college's reserve of talents.

In our future teaching practice, we should continue to improve the recognition and professional degree of master of software engineering, improve our managing system of practical education and maintain our relationship with companies to provide students high-quality practical education.

ACKNOWLEDGEMENT

This research was supported by software school of BUAA. Thanks to Zhang Li, Director of software testing department for her support and help. Thanks to the technical help of Zheng Xianming of HP Company. Thanks to Michael Wagner of Yale University Social Sciences Information Center for discussion of research methods. Thanks to Mr. Keven for corrections to the paper.

REFERENCES

- [1] Barbara K., Riallette P., David B., O. Pearl B., Mark T., Mahmood N., Stephen L. Systematic literature reviews in software engineering – A tertiary study. *Information and Software Technology*, vol. 52, pp.792–805.
- [2] Alvarez, I., Olivera-Smith, M. Learning in Social Networks: Rationale and Ideas for Its Implementation in Higher Education. *Education Sciences*, vol. 3, pp.314-325. <http://dx.doi.org/10.3390/educsci3030314>
- [3] Lu-xin X. Research on the Optimization of Enrollment Data Resources Based on Cloud Computing Platform. *Review of Computer Engineering Studies*, vol. 2, pp.9-12.
- [4] Roger S. P. *Software Engineering - A Practitioner's Approach*. Tata McGraw Hill Publication: New York, 2005.
- [5] Rowlands, K., Avramenko A. Teaching HR Professionals: The Classroom as a Community of Practice. *Education Sciences*, vol. 3, pp.147-157. <http://dx.doi.org/10.3390/educsci3020147>
- [6] Hsing-fen L., Marcela M., Philippe L. Career patterns and competences of PhDs in science and engineering in the knowledge economy the case of graduates from a UK research-based university. *Research Policy*, vol. 39, pp.869-881.
- [7] Luiz F.C. Personality types in software engineering. *Human-Computer Studies*, vol. 58, pp.207-214.
- [8] Mason G. High Skills Utilisation. Under Mass Higher Education graduate employment in service industries in Britain. *Education and Work*, vol. 15, pp.427-456.
- [9] Lam, A. Knowledge networks and careers: academic scientists in industry–university links. *Management Studies*, vol. 44, pp.993–1015.
- [10] Rosenkopf L., Almeida P. Overcoming local search through alliances and mobility. *Management Science*, vol. 49, pp.751–766. <http://dx.doi.org/10.1287/mnsc.49.6.751.16026>
- [11] Qinghong Y., Pengfei F., Zhichao C. Research on Clothing Product Reviews Mining Based on the Maximum Entropy. *International Journal of Online Engineering*, vol. 11.

- [12] Park C. New variant PhD: the changing nature of the doctorate in the UK. *Higher Education Policy and Management*, vol. 27, pp.189–207.

AUTHORS

Qinghong Yang works in Beihang University (Software College and School of Economics and Management). Her research mainly focus on data mining of marketing management. (e-mail: ycrainbow@163.com).

Wen Lu is with Software College, BeiHang University, Beijing, China (e-mail: lunwen@csi.buaa.edu.cn).

Pengfei Feng, is with Software College, BeiHang University, Beijing, China (e-mail: fengbuaap@163.com).

Haijun Wei is with Computer Center of 5Th Middle School, Daxing District, Beijing, China (e-mail: Weihj@163.com)

Corresponding author Zhichao Cheng, is a professor of School of Economics and Management, BeiHang University, Beijing, China (e-mail: Cheng915@vip.126.com).

Submitted 07 November 2015. Published as resubmitted by the authors 05 December 2015.