Team, Leadership, Ethic, and Profession in Software Engineering Education

Zhongmin WANG
National Pilot School of Software
Yunnan University
Kunming, China
wangzhongmin@ynu.edu.cn

***Abstract*—This paper discusses topic of team, leadership and relevant issues in software engineering education. To prepare the students for fulfilling the requirements of information technology, training and equipping with the basic knowledge plus fundamental skills of team and leadership are also crucial task to the technical counterparts, especially for students who are studying software engineering at universities. Those aspects, both technical and nontechnical, build the foundation for the future software engineers with high quality ethic and professional capabilities.**

Keywords-team; leadership; ethic; software engineering; education

I. INTRODUCTION

The information age desires massive number of IT professional employees to carry out the numerous projects. It requires practitioners possess not only technical skills but nontechnical qualifications. The capability of team and communication is the most demanded.

One important mission of establishing the National Pilot School of Software is to close the gap between the practical requirements in real world and the academic education in the IT field. The development of the academic curriculum in software engineering education must focus on the objective. The SE2004 [1] and CC2005 [2] provide the excellent guidelines for that purpose.

Nevertheless, the education of software engineering may be concentrated on the technical details; while the non-technical counterpart, such as team, leadership, ethics and professional conduct, seems to be lack of enough awareness. Even if those aspects should also be considered as the key characteristics for a software engineer or an IT professional practitioner.

This paper presents some constructive opinions on the issues of team, leadership, ethics and professional conduct in the education of software engineering. The implementation of those concepts in the development of curriculum and the relevant practices at the National Pilot School of Software at Yunnan University (NPSS-YNU) is briefly introduced.

II. LEARNING COLLABORATIVE WORK IN A TEAM

Every year, hundreds of students come to NPSS-YNU from all over the country. They may have different dreams to fulfill after four years college life. It has no doubt that the primary task of the students is preparing for their professional careers.

As the potential IT professional workforces, the first challenge is learning working together with other classmates in a team environment. The circumstance requires every individual to deal with the personal interests and the team's accomplishments in a right way. The perfect situation is that all team-members work collaboratively toward the goal of their team. Yet the reality is not so simple and far more away from the perfection.

Regarding to computer programming, as mentioned by Frederick Brooks, Jr. [3], the issues may be:

"First, one must perform perfectly. The computer resembles the magic of legend. If one character, one pause, of the incantation is not strictly in proper form, the magic doesn't work. Human beings are not accustomed to being perfect, and few areas of human activities demand it. Adjusting to the requirement for perfection is, I think, the most difficult part of learning to program".

Though the term computer programming is used, the concern of software engineering is implied.

The new students used to study alone during the period of high school before they step into a university campus. Most of them do not have any idea about software engineering. The students may be impressed by a number of legends about outstanding computer programmers and software companies. They may not understand that software development is a team effort, especially in a modern world. The consequence is that students could make great effort to sharpen their programming skills, and may have ignored promoting the equally important professional capabilities, such as to be a team-player and the corresponding skills.

As we known, the team-work is following a simple rule [4]:

"A team is a group of people who share a common goal. They must all be committed to this goal and must all have a common framework to guide them as they work to achieve the goal".

When students start their team-work training, they may not really understand the role of a team-member in a successful team should be played as the proverb declared: "one-for-all, all-for-one".

According to SE2004 [1], a software engineer shall work as an individual and as part of a team, design appropriate solutions using approaches that integrate ethical, social, legal, and economic concerns, demonstrate an understanding

978-0-7695-3570-8/09 $25.00 © 2009 IEEE
DOI 10.1109/WCSE.2009.360

and appreciation for the importance of negotiation, effective work habits, leadership, and good communication in a typical software development environment. It is crystal clear that a professional team-member, for example, a software engineer can promote personal values and capabilities under the teamwork environment. The benefits for an individual and the achievements for a successful team depend on each other, and also the right solution to communication issues among the team members. Eventually, the self-improvements shall be enhanced through the collaborating teamwork.

III. PREPARING PROFESSIONAL EXCELLENCE

Any member of a team should have a clear vision for the short-period and long-term goals. It helps the individual answer the questions "what I can do now?" and "what I shall do in the future." By realizing the personal capabilities, a team-member may be ready for the professional excellence. On the other hand, a software engineer is working at a flat world in the current information era. This situation shall motivate software engineers setting up both local and global visions in order to be stand out the competitions in the international IT markets.

Training the students with professional skills means that they should learn the technical and nontechnical skills to complete the software engineering tasks. There is no need to state the technical skills in detail here. As the potential software engineers, students must acquire the fundamental abilities to form their specialties. Students should be motivated to carry on the lifelong learning desires for their continuing professional development. They should get ready to extend their talent for advanced technologies and innovations after their graduation.

An excellent software engineer is not appraised only by means of technical capabilities. The ethics and professional conduct are the key qualities to succeed.

The ACM/IEEE-CS joint task force on Software Engineering Ethics and Professional Practices (SEEPP) recommended eight principles as Software Engineering Code of Ethics and Professional Practice [5]. The SEEPP are considered to be the standard for teaching and practicing software engineering. It regulates the software engineers for their positive professional images [6] in eight subjects: public, client and employer, product, judgment, management, profession, colleagues, and self. It is certain that "people are constantly observing your behavior and forming theories about your competence, character, and commitment, which are rapidly disseminated throughout your workplace". The software engineers should make any professional effort to create and manage their positive professional images. The reason and consequences are [6]:

"Successful impression management can generate a number of important personal and organizational benefits, including career advancement, client satisfaction, better work relationships (trust, intimacy, avoiding offense), group cohesiveness, a more pleasant organizational climate, and a more fulfilling work experience. However, when unsuccessfully employed, impression management attempts can lead to feelings of deception, delusion, preoccupation, distraction, futility, and manipulation".

IV. LEADER AND LEADERSHIP

Literally, the word "leader" does not necessary mean any specific administrative or official positions. The situation in China is a little complicated. Since the word "leader" in Chinese is a smart translation based on its original meaning and pronunciation plus carefully chosen Chinese characters. The meaning of word "leader" in Chinese seems to be limited to certain type of official positions. The Chinese word "leader" or "leadership" really leads people toward certain specific directions without caring the others.

On the other hand, if leader is on the horizon, there should be followers. It is reasonable and natural. Every student wants to be successful; while most of students who are studying software engineering would disagree with that they should be preparing to be leaders, at least technical leaders. But it is positive that no one wants to be a follower, although a follower could be a leader eventually, if valuable experiences are accumulated and lessons are learnt from leaders and others. The problem is that, again, the word "follower" may bring uncomfortable feeling to students. This extraordinary identity is not acceptable. It has been customized with specific implication.

On the ground of software engineering, the principle function of a team-leader is guiding the team to get the job done. It sounds like a piece of cake. The key point of leadership is "guide". Not everyone is capable of guiding others to do the great job without training, even an excellent technical specialist, if this person does not have the inspiration and basic skills about running a team. The actual performance depends on a variety of factors.

There are rarely natural born leaders. A team-leader must be at least an excellent team-member. This type of team-member will accumulate the experiences and skills of leadership during the daily work, reveal the ability of being a good team-player. This is actually establishing the foundation for a team-leader. This individual must fully understand the mission of a team, and has demonstrated the exceptional personality, possesses dazzling vision, and creative working skills among the team-mates. The leader must recognize the differences between leadership and management, appreciates the fact that people like to be led instead of being managed.

The ultimate goal of software engineering is to deliver high quality software product. It requires a high quality team led by a high quality team-leader. A good leader translates an inspiring original vision into synchronized daily executions on a high performance, high quality working base [7]. The factor is that [8] "…about 15 percent of one's financial success is due to one's technical knowledge and about 85 percent is due to skill in human engineering—to personality and the ability to lead people". Students need to recognize that every one of us has the potential to be a leader.

All in all, the characteristics of leadership can be encapsulated in one word: responsibility. The students studying software engineering should learn to take responsibility and commitment.

V. THE PRACTICES AND SUMMARY

The significant points for software engineers are [9]: "Computer software engineers must continually strive to acquire new skills in conjunction with the rapid changes that occur in computer technology". They "often work as part of a team that designs new hardware, software, and systems. A core team may comprise engineering, marketing, manufacturing, and design people, who work together to release a product." We have to help students to realize that team, leadership, ethics, and profession are essential to their successful careers.

Although an independent course about those aspects discussed in this paper is not applicable in NPSS-YNU, we have embedded those ideas into our core courses. The key course Software Engineering presents the principles of software development. Programming courses develop the technical skills and let the students implement the philosophy of software engineering. Practical Training, either on-campus or off-campus, and internship enhance their team-working abilities. The students can also advance their skills of verbal and written communication by the successive courses of Reading and Communication in Computer English, which consists of reading, writing, listening, and communication modules. These experiences provide students with broad knowledge and experience, making them more attractive to employers.

ACKNOWLEDGMENT

I would like to thank my colleagues at NPSS-YNU for the beneficial discussions on the topic. I also want to express my appreciation to our executives for their encouragement.

REFERENCES

[1] The Joint Task Force on Computing Curricula, "Software Engineering 2004, Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering, A Volume of the Computing Curricula Series", IEEE Computer Society, Association for Computing Machinery, August 23, 2004.

[2] The Joint Task Force for Computing Curricula 2005, "Computing Curricula 2005", The Association for Computing Machinery (ACM), The Association for Information Systems (AIS), The Computer Society (IEEE-CS), New York, September 2005.

[3] Frederick Brooks, Jr, The Mythical Man-Month, Addison Wesley Longman, Inc, New York, 1995.

[4] Watts S. Humphrey, TSP(SM)-Leading a Development Team, 1stEd, Pearson Education, Inc, New York, 2006.

[5] ACM/IEEE-CS joint task force on Software Engineering Ethics and Professional Practices (SEEPP): Software Engineering Code of Ethics and Professional Practice, 1999. http://www.acm.org/about/se-code/.

[6] Mallory Stark, Creating a Positive Professional Image, Harvard Business School, Working Knowledge, June 20, 2005: http://hbswk.hbs.edu/cgi-bin/print?id=4860

[7] Raymond T. Yeh and Stephanie H. Yeh, The Art of Business: In the Footsteps of Giants, Zero Time Publishing, 2004.

[8] Dale Carnegie, The Leader in You, Pocket, New York, 1995.

[9] U. S. Bureau of Labor Statistics, Occupational Outlook Handbook, 2008-09 Edition, http://www.bls.gov/oco/ocos267.htm.