

Engaging Students in Active Learning by Introducing Game Development into Software Engineering

Jun Lin^{1,2}, Chunyan Miao¹, and Wei Sun²

¹ School of Computer Engineering, Nanyang Technological University,
Singapore 639798

jlin7@e.ntu.edu.sg, ascymiao@ntu.edu.sg

² College of Software, Beihang University (BUAA), Beijing,
China 100191

linjun@buaa.edu.cn, weisun@buaa.edu.cn

Abstract

We report our experience by using game development in university education to undergraduate and graduate students who major in Software Engineering at the College of Software, Beihang University, China. Our approach is combining game development and practical software engineering related curricula to engage students in active learning. We will discuss the benefits and challenges that arise from such a combination for both educators and students. The aim of this paper is to clarify the aspects that must be considered to apply game development to software engineering education at university level.

Keyword: Computer Game Education, Software Engineering, Learning by doing, Experiential learning, Active Learning

I. Introduction

A. Background

In recent years, many students have not believed in the job market in computer science and software anymore, they doubt the career prospects in computer science and fear for the hard,

tedious and monotone study, which has directly induced the dropping of the computer science student enrollments. If the decline continues, the foundation for tomorrow's computer scientists and software engineers, even whole information technology industry will be impaired. On the other hand, recently computer games have a large and continuously expanding world-wide market, especially in China. Games, entertainment applications and interactive digital media including animation and cartoon, are an important economical factor that cannot be easily neglected.

Facing the dilemma of software engineering education and game industry, many universities have expressed a desire to make the Computer Science major more compelling and enjoyable by adding some computer game factors into the classroom. In fact, for an educator who teaches software engineering, computer games are ideal examples to demonstrate the pure benefit of computer education, and more importantly, the game development process is more entertaining and interesting, which can engage students into an active learning status.

From the technical point of view, computer games comprise a huge variety of important engineering techniques such as modular software design, high performance, ingenious data structures, sophisticated algorithms, and interaction design. Practical knowledge in such areas is important for every educator, regardless of application area. Some educators and industrial experts recognize that exposing students to game development concepts in the context of a rigorous program of computer science study is good for the students, educators and the industry. Hence, the computer game has been a new medium and art form that is new to the generation in universities today. Developing a computer game contains all of the basic elements that are taught in computer science and software engineering classes could have become the most effective way of teaching and learning process. Students have fascination and motivation to games and game-related technologies, which can be used to encourage them to actively learn more about the technical aspects of computer science and software engineering.

B. Our Approach

The master of software engineering degree program of Game Development & Design in Beihang University is designed to provide students with the ability to apply core knowledge of software engineering to game development through a set of curriculums that emphasize interactive software development techniques and game development process. For the undergraduate students, although we did not provide explicitly special game development subject for them, game-oriented themes were introduced into their practical curriculums. Applying some viewpoints of Kolb's Experiential Learning Theory and Gibbs's teaching method "Learning by doing", our approach integrates the features of game development and software engineering process theory. The following list of points and actions has been used in the case:

- Active learning is very important in our approach.
- Emphasizing learning form practice.
- The process will be carefully designed by teachers.
- Advocating team work and collaboration.
- The learning result is open and can be reused. It follows up this openness which may not exist at the outset but may be fostered through successive experiences of the experiential learning cycle (Gibbs, 1988).
- Arranging some teaching assistants or instructors to supervise the process of student projects by a tailored Rational Unified Process (IBM, 2003) model that provides a framework of cyclical sequence of learning activities (see Figure 1).

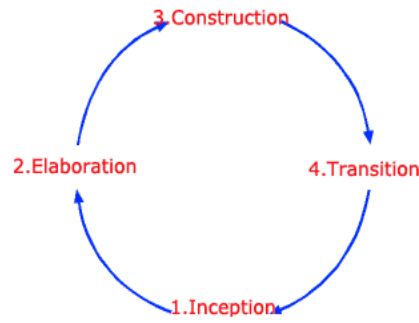


Figure 1 Cyclical sequence of learning activities

- Emphasizing continual and effective learning through projects. In each phase of the cycle, students might do iteratively their works in some loops for improving the quality of result, and also do iteratively their whole work in some incremental loops for increasing their project experiences. This learning cycle might involve the sequence of learning activities illustrated below:

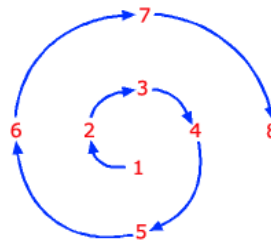


Figure 2 The incremental learning cycle

II. Case Studies

A. A Case Study in Undergraduate Curriculum

Practice and Observations

We have applied the above approach to our undergraduate practical courses for many years. The students registered in such courses will be required to participate in a software development project. They will be divided into some project groups voluntarily. At the beginning of a project, the instructor made a list of requirements that the project must fulfill. But the contents and developing technologies were up to students themselves. There are many teams who choose game and graphics direction in every year.

Results and Challenges

Table 1 shows the comparison result of projects in game related and of other types.

Table 1 Teaching result from Academic Year 2005 – 2009

Academic Year	Student average score of game related projects (Total:100)	Student average score of other projects (Total:100)
2006	87.5	84.6
2007	82.9	80.2
2008	86.2	80.9
2009	86.4	82.3

The result shows the average score of students who choose game and related projects is obviously higher than that of the others. Further, every year we will encourage our student team to take part in some contests, such as Microsoft Imaging Cup, BUAA Fengru Cup, etc. The game related projects can always get better awards. For example, the work “Digital Beihang” (Figure 3. left-handed), won the first class award of Fengru Cup in 2006. Another team’s work mobile game “Campus Dustman” (Figure 3. right-handed) won the second class award in 2007.



Figure 3 Some undergraduate students’ works of games

There are, however, a couple of difficulties and challenges to both educators and students. The first one is about programming technologies, especially gaming programming related technologies. Many game platforms and related techniques are not open for public, so that we just can narrow directions into a few development platforms and technologies. Another aspect is that the works of content creation, for example artwork design and game play design, are

hard to students who major in an engineering-related or technology-related discipline. If we neglect content creation, the result of games will become boring and unattractive. To meet this challenge, we suggest the project team invites other students of arts-related discipline to join in. These collaborations between disciplines are very useful for the successful teams.

B. A Case Study in Graduate Game Development Curriculum

Practice and Observations

Our graduate students who major in Game Development and Design will take four distinct specialty courses beyond the generic software engineering courses. After completing these courses, our students will be organized to some small game development teams consisting of 3 to 7 members before they enter into game companies for internship. During about three months of internship, they will taste all phase works in game development process.

The teachers of specialty curriculums will be their mentors or supervisors for the projects. Finally, their products will be given a mark to evaluate their capability for entering into game industry.

Results and Challenges

After one semester special studying and practical training, our students will be equipped with game design techniques and skills. Figure 4 shows some works from our graduate students' teams.



Figure 4 Works of graduate students

These game projects were quite successful. Although there is always room for improvement, both the teachers and the industrial experts were somewhat amazed about the complexity and the quality of the final deliverables. Consequentially, they were warmly welcomed by the industry, and the average starting salary of these students is higher than that of the rest of students.

Of course, however, creating game products combines the skills of art and programming knowledge. within a similar way as in the case of undergraduate students, content creation is also a big challenge to those graduate students with only engineering background. Besides this, there are many other difficulties which all game designers have to cope with, e.g. the development of a well-designed game engine.

III. Conclusion

Developing computer games at a university has several benefits and challenges. According to our experiences using game development in computer science education for undergraduate and graduate students at Beihang University, the following list comprises some important points that we were aware of:

- Game development offers a playful approach to software development from a didactic point of view.
- A well-defined process is very important to the use of game development in software engineering education.
- Acquiring effective skills and achieving high starting salary are very important to students. Learning by doing is the best way to help them to reach the goal.
- Using game development in software engineering education at university level should collaborate with game industry closely.

Now our work is mainly depended on the teachers' experiences. We have to see that creating a structured, detailed and well-defined process and required skills of game development are

essential. Our further work will aim at adding more compelling and enjoyable factors of game development and design to practical courses for software engineering and other related students. Accordingly, we'll also develop a set of tangible curriculums ranging from simple game theory to an ambitious game project for building hands-on Lab to reuse those works.

References



Jun Lin is currently pursuing the Ph.D degree in the School of Computer Engineering, Nanyang Technological University, Singapore. He received his B.Sc and M.Eng from Southwest University for Nationalities, Chengdu, China in 1998 and Beihang University, Beijing, China in 2005, respectively. His current research interests include software engineering, web services, cloud computing and Agent/Multi-Agent technology.



Dr. Chunyan Miao is an Associate Professor in the School of Computer Engineering, Nanyang Technological University (NTU), Singapore. Her current research is focused on human-centered computational/ artificial intelligence and interactive media for the young and the elderly.



Wei Sun received his Ph.D degree in computer science from University of Illinois, Chicago, in August 1990. He then joined School of Computer Science, Florida International University, as an Assistant Professor. Now he is a professor and serves as Dean in College of Software, Beihang University, Beijing, China. His major research interests are in location-based system, database systems, knowledge-based systems, and multimedia systems.