Chapter 16
Teaching Software Engineering through a Collaborative Game

Elizabeth Suescún Monsalve
Pontifical Catholic University of Rio de Janeiro, Brazil

Allan Ximenes Pereira
Rio de Janeiro State University, Brazil

Vera Maria B. Werneck
Rio de Janeiro State University, Brazil

ABSTRACT

This chapter addresses the application of computer games and simulations in order to explore reality in many educational areas. The Games-Based Learning (GBL) can improve the teaching and learning experience by training future professionals in real life scenarios and activities that enable them to apply problem-solving strategies by putting into use the correct technique stemming from their own skills. For that reason, GBL has been used in software engineering teaching. At Pontifical Catholic University of Rio de Janeiro, the authors have developed SimulES-W (Simulation in Software Engineering), a tool for teaching software engineering. SimulES-W is a collaborative software board game that simulates a software engineering process in which the player performs different roles such as software engineer, technical coordinator, project manager, and quality controller. The players can deal with budget, software engineer employment and dismissal, and construction of different software artifacts. The objective of this chapter is to describe the approach to teaching software engineering using SimulES-W and demonstrate how pedagogical methodology is applied in this teaching approach to improve software engineering education. The teaching experience and future improvements are also discussed.

DOI: 10.4018/978-1-4666-5800-4.ch016

INTRODUCTION

The goal of software engineering education is to provide students the necessary methods and techniques (and later software professionals) to develop quality software (Sommerville, 2007). Teaching software engineering requires not only considering the theoretical aspects but also the principles and methodologies for software development and maintenance, including the aspects concerning the practice and application of that knowledge.

Traditional classes, even if they use real-world projects, cannot always simulate the decisions software engineers have to deal with in their daily activities. Thus game-based learning has been successfully used as a support tool in several areas, including software engineering in the traditional classroom (Bollin, Hochmuller, & Mittermeir, 2011, Hainey et al., 2011, Drappa & Ludewig, 2001, Jain & Boehm, 2006, Birkhoelzer, Navarro, & van der Hoek, 2005). Within this context, this chapter presents our experience of teaching software engineering using SimulES-W, a collaborative game, in which players perform different roles, such as software engineer, technical coordinator, project manager and quality controller. This game has been implemented over two years in both undergraduate and graduate courses, in which the learning objective is accomplished through teaching strategies.

With SimulES-W we can introduce both general and specific software engineering knowledge and apply an educational component that allows the practice of simulating. Students using SimulES-W can identify the systematic approach for developing disciplined and qualifying software, hence understand aspects related to software quality and maintenance. Thus students perform different roles in which each student has to deal with the project budget and the hiring of software engineers. The game also has a system of concepts and problem cards that are used to improve the game itself or used to block the other players' moves. These cards contain the theoretical software engineering knowledge that must be analyzed and applied by the students. Moreover, the game also has an activity for building the software product. The students would need to construct software artifacts required by the project, make inspections, fix errors if they appear in some artifacts, and pack and deliver the product. While at the same time, they need to pay attention to the budget assigned to the project. The first student who is able to construct the software without having any problems, will win the game.

The main advantage of using SimulES-W regards customization. It can be used to teach a specific software engineering aspect or a comprehensive software engineering project as the instructor assigns it. The game application in the classroom results in the most important discussion. After the game is concluded an analysis activity is performed to help identify elements that can contribute to students' knowledge improvement. Furthermore, the feedback received from the students helps improve both the game and the strategy activity.

We are aware of the difficulty regarding the leading changes in teaching methodology, both for teachers and students. The teachers are the motivators in this process, and the students have to understand they can learn using a game. Naturally, it is very motivating knowing the preference for using games as a teaching and learning mechanism, which is our greatest driving force. To cover these aspects, this chapter is divided into six sections. Section 1 contains the introduction. Section 2 describes the background of software engineering education with games, Section 3 presents the SimulES-W, Section 4 introduces our methodology to teach software engineering with SimulES-W, Section 5 presents our teaching experience with SimulES-W in the Software Engineering discipline in the Computer Science Program at the State University of Rio de Janeiro. Conclusions are in Section 6.