Chapter 17
Evaluating Games-Based Learning

Thomas Hainey
University of the West of Scotland, Scotland

Thomas Connolly
University of the West of Scotland, Scotland

ABSTRACT

A highly important part of software engineering education is requirements collection and analysis, one of the initial stages of the Software Development Lifecycle. No other conceptual work is as difficult to rectify at a later stage or as damaging to the overall system if performed incorrectly. As software engineering is a field with a reputation for producing graduate engineers who are ill-prepared for real-life software engineering contexts, this paper suggests that traditional educational techniques (e.g. role-play, live-through case studies and paper-based case studies) are insufficient in themselves. In an attempt to address this problem we have developed a games-based learning application to teach requirements collection and analysis at the tertiary education level. One of the main problems with games-based learning is that there is a distinct lack of empirical evidence supporting the approach. This paper will describe the evaluation of the requirements collection and analysis process using a newly developed framework for the evaluation of games-based learning and will focus on evaluation from a pedagogical perspective.

INTRODUCTION

Requirements collection and analysis is a highly important early stage of the Software Development Lifecycle (Sommerville, 2007). According to Brooks (1987) no other conceptual work is as difficult to rectify at a later stage of a software project or as damaging to the overall system if performed incorrectly. Games-based learning (GBL) could potentially overcome some of the problems associated with traditional approaches to teaching requirements collection and analysis, as it is perceived to be a highly engaging form of supplementary learning by some educationalists. A primary question of this research is to ascertain if a GBL intervention can act as a suitable supplement for different learning experiences such as role-play and paper-based case studies.
One of the main problems associated with the field of games-based learning is the distinct lack of empirical evidence supporting the approach (Connolly, Stansfield & Hainey, 2007; de Freitas, 2006). This paper will make a contribution to the empirical evidence in the games-based learning field by conducting 3 experiments comparing a games-based learning approach to role-playing and paper-based approaches for teaching requirements collection and analysis. Two studies were performed at the University of the West of Scotland, the first in May 2008 and the second as part of a serious games module in April 2009. A third study was also conducted at Glasgow Caledonian University in April 2009. This paper will discuss some of the problems associated with teaching requirements collection and analysis, the advantages and disadvantages of traditional teaching approaches, and how and in what ways GBL can act as a suitable supplement to these traditional approaches. The problem of the lack of empirical evidence and general evaluation frameworks will then be discussed, leading to the introduction of a new evaluation framework for games-based learning. The requirements collection and analysis game will then be described in more detail. Finally, the results of the three individual experiments will be presented followed by discussion and a conclusion.

PROBLEMS IN TEACHING REQUIREMENTS COLLECTION AND ANALYSIS AND SOFTWARE ENGINEERING

The problems associated with teaching software engineering have been discussed in detail in a previous study (Connolly et al., 2007). Problems specifically associated with teaching requirements collection and analysis are deeply rooted in the lack of definitional conformity to what a requirement is.

SOFTWARE REQUIREMENTS

The IEEE defines a requirement as “a condition or capability needed by a user to solve a problem or achieve an objective” or “a condition or capability that must be possessed by a system to satisfy a contract standard, specification or other formally imposed document.” A requirement is a statement specifying part of the required functionality of the system. One of the primary reasons that requirement capture and analysis is so problematic and complex is that a requirement can be expressed in different levels of abstraction or complexity. Sommerville (2007) emphasizes “the term requirement is not used in the software industry in a consistent way. In some cases, a requirement is simply a high-level, abstract statement of a service that the system should provide or a constraint on the system. At the other extreme, it is a detailed, formal definition of a system function” (p. 118). To combat the complication encountered by the different levels of abstraction, Sommerville (2007) distinguishes between user requirements and system requirements:

- User requirements (requirements of a high level of abstraction) are “statements, in a natural language plus diagrams, of what services the system is expected to provide and the constraints under which it must operate.” (p. 118)
- System requirements (requirements of a highly detailed nature describing what the system should do) “set out the system’s functions, services and operational constraints in detail. The system requirements document (sometimes called a functional specification) should be precise. It should define exactly what is to be implemented. It may be part of the contract between the system buyer and the software developers.” (p. 118)
Related Content

Connected Learning in an Australian Technology Program: A Case Study
[www.igi-global.com/article/connected-learning-australian-technology-program/51628?camid=4v1a](www.igi-global.com/article/connected-learning-australian-technology-program/51628?camid=4v1a)

Developing New Literacies through Blended Learning: Challenges and Lessons Learned in Ontario, Canada
[www.igi-global.com/article/developing-new-literacies-through-blended/55935?camid=4v1a](www.igi-global.com/article/developing-new-literacies-through-blended/55935?camid=4v1a)

Activity Theory and Context: An Understanding of the Development of Constructivist Instructional Design Models
[www.igi-global.com/chapter/activity-theory-context/25957?camid=4v1a](www.igi-global.com/chapter/activity-theory-context/25957?camid=4v1a)

Dialogues and Perception of Intersubjectivity in a Small Group
[www.igi-global.com/chapter/dialogues-perception-intersubjectivity-small-group/63202?camid=4v1a](www.igi-global.com/chapter/dialogues-perception-intersubjectivity-small-group/63202?camid=4v1a)