GONDOLA:
An interactive computer game-based teaching and learning environment for Requirements Engineering

Chad Coulin and Didar Zowghi

Faculty of Information Technology, University of Technology Sydney
PO Box 123, Broadway NSW 2007, Australia
{chadc, didar}@it.uts.edu.au

Abstract. As the field of Requirements Engineering continues to evolve, so too does the need to develop specialized educational programs that will provide student and novice requirements engineers with the required expertise and experience necessary to execute the tasks successfully. We follow on from previous work performed on role-playing for Requirements Engineering education with the development of Gondola: an interactive computer game-based teaching and learning environment for Requirements Engineering. In this paper we investigate how computer game-based systems are suitable as novel and innovative solutions to Requirements Engineering education. The focus however is on the presentation of the framework and guidelines we have developed for this environment, in addition to some of the key architectural components of the supporting tool. It is expected that the implementation of this environment will help improve Requirements Engineering education, and ultimately lead to better quality results in teaching, learning, and practice.

1 Introduction

Requirements Engineering (RE) is a complex and multi-faceted process where successful execution depends greatly on the experience and expertise of the participating requirements engineers. Although RE has established itself as its own discipline within the field of Software and Systems Engineering over the past decade or so, little research has been devoted to discovering how to better educate students and novice practitioners on its techniques, methods, and approaches, and more importantly, how to develop the basic skills required to apply them in practice.

Experts in RE tend to be those with the most years of experience, having applied the theoretical aspects of the discipline to many and varied practical situations. In general, requirements engineers learn by doing, by discovering and solving problems, and by making mistakes and redoing. As a result we believe that student and novice requirements engineers would benefit from a role-playing style computer game-based simulation environment to develop and practice the necessary skills. This enables them to learn as the experts do but in a secure and controlled setting.

The paper is structured as follows: Section 2 investigates the appropriateness of computer games in RE education. Section 3 provides an overview of the Gondola environment in terms of its concept, context, rationale, and objectives. Section 4 describes the framework (hierarchy, tasks types, and guidelines) on which the environment is based. Section 5 details a number of the key components of the supporting tool, and Section 6 explains an example scenario of the Gondola environment in action. Finally, Section 7 presents some discussions and future work, including plans for the evaluation of the theory presented in this paper through the actual development and implementation of the environment.

2 Computer Games and Education

Although computer games have been used in educational settings for decades, there is still significant debate over their true value in this context. Recently this topic has attracted greater attention, such as several large high profile projects including the Microsoft iCampus initiative with the Comparative Media Studies department at the Massachusetts Institute of Technology Games-to-Teach project [12], and the Computer Games in Education (CGE) project [3] as part of the British Educational Communications and Technology Agency (BECTA) software initiative.

Computer games provide a powerful medium to combine entertainment and play for the benefit of education (sometimes referred to as "edutainment"). Recent advances in computer game theory, design, and technology enable researchers and practitioners to take advantage of this more than ever by integrating them with current education research [14]. Furthermore, games can facilitate learning by utilizing something that is already an existing part of modern culture, especially for students of information technology.

To date role-playing computer game-based environments have been used to relive and simulate everything from managing a football team [6], to the construction of cities [7], and even the development of entire civilizations [1]. Although not a primary goal of these systems, upon closer examination we can determine why they have been successful in educating users about these situations, often subconsciously. They provide the user with the correct motivation and incentive to learn by making the process entertaining and engaging. Users are frequently offered support and encouragement in these environments, and are provided with regular and constructive feedback. Some of the most successful implementations of this type of environment for purely educational purposes can be found in military applications used to train and test commanders and soldiers in battle situations. So, why not apply these same principles to the education of student and novice requirements engineers?

3 An Overview of the Gondola Environment

The name Gondola was given to the environment the paper presents as a result of the parallels between the process of learning and conducting RE, and traveling in a small

boat through a system of canals. A gondola must avoid and overcome obstacles as it makes its way to its destination. The canals can present changing conditions to a gondola that must be managed and addressed. Often there are a number of possible paths a gondola can take in order to reach its destination. And finally, a trip in a gondola is not just about reaching the final destination, but also about the journey itself, and enjoying the process of getting there.

The concept for the Gondola environment was developed based on both novel and established education research literature [11]. In accordance with these principles the project adopts a learner-centered [13] and constructivist [4] approach to the design of the environment, taking advantage of existing work on authentic online environments [8] and game-based applications [15]. Originally Gondola was developed as a way of addressing some of the issues discovered in previous work investigating the value of role-playing in RE education [16], and combining other research efforts within the group into providing tool support for various phases of the RE process, specifically requirements elicitation [5]. As the result of a thorough investigation, it was determined that an interactive computer game-based system would significantly enable researchers and practitioners to support the teaching and learning of the theoretical and practical aspects of RE.

This innovative environment used to educate and evaluate students on RE, as shown in Figure 1 below, consists of a conceptual framework with guidelines, a subject curriculum and content on which to base the implementation, and appropriate tool support. It requires the involvement of subject coordinators, tutors and students.

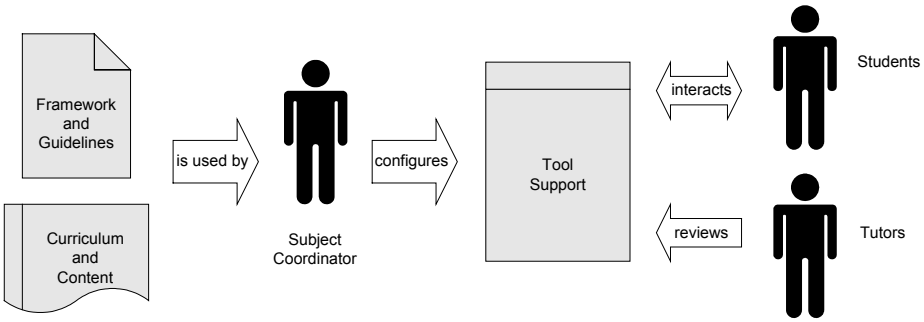

Figure 1. The Gondola Environment

By utilizing web-based, hypermedia, and intelligent technologies, users are able to role-play within simulated real-world scenarios. Students are required to complete a series of tasks in order to fulfill specific goal-based objectives. Within a task students may be required to collect evidence to support a theory, propose a solution to a problem, interview a virtual participant, comment on a virtual scenario, examine a document, or select the appropriate action and next step to a particular situation.

The environment allows students to learn at their own pace, and facilitates tasks to be performed iteratively and repetitiously, while proving timely and relevant feedback. Furthermore, the environment would enable students to be assessed in a variety

of ways including successful navigation to one of several possible conclusions, or the completion of online quizzes at various stages in the process. The performance of the student on individual tasks, and the quality of the final work products produced as a result of the tasks, also presents opportunities for assessment and feedback.

The Gondola environment combines various aspects of three common computer game genres: Strategy, Simulation, and Role-Playing [9]. This mix enables visualization of the learning process, explanation of concepts and theory, environmental simulation, and most importantly, the creation of activities that required practical RE skills to complete such as goal-based problem solving, case-based reasoning, and decision-making. We are also able to implement other elements found in the real world of RE like unpredictability, challenge, choice, and varying levels of difficulty.

Based on existing applications and research, combined with how real requirements engineers become experts of the process, we believe that a computer game-based environment provides an excellent opportunity and match with the types of skills required by requirements engineers, and the types of experiences encountered. Furthermore, we envisage the Gondola environment as a means of introducing existing RE theories into practice, and increasing the awareness for new RE techniques and approaches into industry, thereby reducing the gap between theory and practice, and the time to adoption.

3.1 Context

RE is critical to the success of any system development project, however it remains one of the biggest bottlenecks and most poorly conducted activities. This is largely due to the lack of true industrial experience and expertise in RE theory and practice. Errors made during the RE phase of a project can lead to the failure of both the project and the resultant system. Subsequently, effective and efficient RE is a skill essential for all current and future information technology professionals.

The Gondola environment was developed using as a foundation the curriculum and content of the Requirements Engineering subject offered as part of the Bachelor of Science in Information Technology (BIT) program by the Faculty of Information Technology at the University of Technology, Sydney (UTS). This subject is based on a 14-week semester calendar consisting of lectures, workshops and tutorials. Theory explained in lectures is applied to practical situations in the workshops and tutorials. Assessment is by an end of semester exam plus workshop assignments and tutorial exercises.

3.2 Rationale

The execution, and subsequently the teaching, of RE is a very difficult, complex and multi-faceted activity. As a result, educators and employers alike have called for a rigorous and structured approach to this identified problem area [10]. Current approaches to RE education, including assessment and feedback are extremely subjective, labor-intensive and time consuming for both students and staff. Heavily used human-based simulation exercises are not effective in replicating real-world situa-

tions. Undergraduate students especially lack the required knowledge and experience to accurately represent and role-play real-world participants and scenarios.

Due to typically large student numbers in RE subjects (this is particularly the case at UTS where RE is a compulsory core subject for all BIT students), students will get the opportunity to role-play only a limited number of the many possible different kinds of participants in the RE process. The performance of individual students in RE group activities is difficult to assess due to the heavy dependence on the other group members. The need for individual experiences and authentic interaction is essential for effective RE education.

Often students fail to understand the relevance and importance of RE to real-world situations, and therefore lack the correct motivation to develop and improve these skills. As a result the Gondola environment is designed to fulfill the need for a teaching and learning approach more in line with students preferences to address this difficult educational issue.

3.3 Objectives

The main aim of the Gondola environment was to improve the quality of RE education by initially supporting, and then replacing the existing human-based role-playing activities currently employed in the RE subject. The system needed to be engaging, entertaining and attractive to students. Its operation should be challenging and unpredictable in a way that supports the learning process.

It was desired that the system would enable education on not only the best practices of RE for software and systems development in terms of processes, methods and techniques, but also familiarize students with the trends and challenges of RE in practice by simulating real-world environments, activities and participants. Thereby increasing the level of preparation and readiness of students for industry.

Students should be provided with increased motivation to learn by placing this subject in relevant and realistic contexts, enabling them to apply their knowledge and practice these skills to improve their performance in these tasks at their own pace, and assist in the process of self-assessment and reflection.

A side benefit of the implementation of Gondola was a significant reduction in the time required for supervising and marking students on various topics within the subject of RE by providing an interactive teaching and learning tool that also automates some of the assessment and feedback elements.

The ultimate objective of Gondola is to bridge the gap between expert and novice requirements engineers by providing realistic learning experiences, thereby improving the quality of RE education and practice. Furthermore, the environment has been designed to enable academics and practitioners to apply, customize and reuse the framework, guidelines and tool support for their own uses, within or outside the field of RE.

4 Framework and Guidelines

In this section we describe the Gondola framework and guidelines. The framework comprising of user groups, a hierarchy, and task types, forms the foundation for the design and development of the supporting tool. The guidelines provide assistance for the integration of a subject curriculum and content into the game-based format.

4.1 Hierarchy

The Gondola framework has a hierarchical structure of five levels as shown in Table 1 below. By using a multi-level hierarchy to construct the learning experience, a subject coordinator is able to impart goals at different levels of the process, consistent with theoretical concepts of constructivist learning. This structure encourages users to become immersed in the processes [2], and captivated by the interaction.

Table 1. Hierarchy

Level	Name	Description
1	Subject	Equivalent to a standard subject within a course, e.g. Requirements Engineering. Within the framework a subject has a coordinator, code, name, introduction, and one or more objectives.
2	Topic	Equivalent to a standard topic within a subject, e.g. Requirements Elicitation. A topic may have one or more available missions with different levels of difficulty. Within the framework a topic has a code, name, introduction, and one or more objectives.
3	Mission	Represents an example case study from the real world, relevant to the topic and the subject, e.g. Online Bookshop. A mission may be composed of any number of stages and tasks. Within the framework a mission has a code, name, level of difficulty, introduction, supporting artifacts, and one or more objectives and instructions.
4	Stage	Represents an activity within a mission, e.g. Project Kick-off. A stage may be composed of any number of tasks. Within the framework a stage has a code, name, introduction, one or more objectives and a minimum pass score.
5	Task	Base level of operation the user has with the system and represents a single process step in a stage, e.g. Interview the customer project manager. Within the framework a task has a code, type, name, introduction, supporting artifacts, and one or more objectives.

4.2 Task Types

There are nine different task types available within the Gondola framework as show in Table 2 below. By using different task types within the same stage, subject coordinators are able to maintain the interest of the users by varying the activities performed, and promote different kinds of learning experiences.

Table 2. Task Types

Type	Description
Interview	The user is required to interact with a virtual actor within the system via a question and answer protocol in order to obtain specific and general information
External	The user is required to perform some activity using tools outside of the environment such as researching a specific type of system, prepare a questionnaire, write a use case, or create a diagram or model
Justification	Users are asked to provide the rationale for an action they have performed or a decision they have made, or comment on the possible rationale behind an action or event within the mission.
Evaluation	The user is asked to compare and contrast the suitability of different techniques, tools and approaches to a specific situation and produce a selection, opinion or recommendation
Reflection	Users are asked to comment on how results from a task might have been improved in hindsight, or where they believe errors and mistakes have been made throughout the stage
Application	The user is required to directly apply some theory or method to a practical situation within the environment such as goal-based decomposition, or a classification system
Review	The user is required to examine an artifact, document or work product, and identify key points, defects, or inconsistencies with previously acquired information
Quiz	Users are required to complete a short set of multiple-choice questions relating to the stage and tasks previously performed, the relevant background theory, or its application
Miscellaneous	The user may be required to perform an operation not covered by any of the other task types defined in the framework

During a task the user may be expected to acquire **Access Keys** by successfully performing certain predefined actions. These keys are used to determine which next tasks will be available for the user to choose from upon completion of the current task. Keys are linked to the fulfillment of the objectives for the tasks, and may be acquired in different ways based on the combination of one or more actions during one or more tasks. Keys enable a simple way for subject coordinators to setup prerequisites for subsequent tasks.

Reward and Penalty Points may be awarded to the user during a task. Reward points may be awarded for the user asking a key question during an interview, or

identifying all the inconsistencies in a document. Penalty points may be awarded when the user selects the incorrect next task, or a task option that has a penalty point associated. Points enable the subject coordinators to provide easy and direct feedback to the user, and introduce an element of healthy competition into the game by challenging the user to out perform the other players and the system itself.

The first task for each stage is automatically presented to the user. Online help is available throughout the process, specific to the user's current situation and task. Upon completion of the last task for a stage, a performance result is formulated to determine if the user is permitted to continue and proceed to the next stage, or must replay the current stage again.

4.3 Guidelines

Guidelines have been developed to support the framework and provide assistance to subject coordinators in determining how the framework can best be applied, and the tool configured. These guidelines, as summarized below, are intended as recommendations that may be adjusted depending on the subject content, objectives, and the level of education of the users.

4.3.1 Subject Structure

A subset of guidelines has been developed on the recommended task structure for missions and stages. For example, the number of tasks for a stage should be no less than 3, and no more than 10, to prevent the user from becoming disorientated or disinterested. Also it is recommended to have a quiz as the final task for each stage as a way summarizing and reiterating the lessons of the stage to the user.

4.3.2 Task Design

Guidelines for the design of individual tasks of each type have also been developed. For example, a word limit should be set when the user is required to provide justification for an action to encourage them to be concise and succinct in their responses. Users should be aware of the number of inconsistencies they are expected to find when reviewing a document, enabling them to determine an appropriate completion point for the task. In general, a quiz should be made up of no more than 20 questions, and be normally completed in no more than 20 minutes.

4.3.3 Task Sets

Another subset of guidelines has been developed for creating sets of logical tasks. For example, an interview set of tasks can be constructed whereby the user is first required to develop a specific questionnaire, select the most appropriate actors, interview the actors using the developed questionnaire, and analyze the responses. Wherever appropriate throughout a task set, users should be asked to provide rationale and reflect in order to record and demonstrate the advancement of their knowledge.

5 Tool Components

In this section we describe a number of the key architectural components of the supporting tool for the Gondola environment, as shown in Figure 2 below. This tool represents the implementation of the framework into a computer game setting.

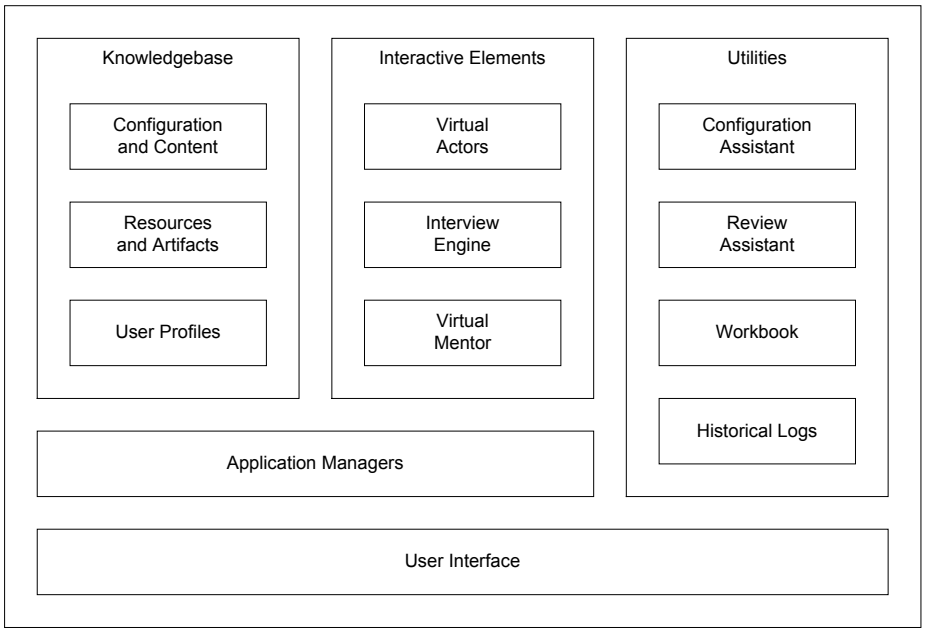

Figure 2. Tool Components

The **User Interface** enables users to navigate through, and perform the required tasks, interact with other components of the system, and view historical information such as the number of reward and penalty points awarded, the number of keys acquired, and the results of quizzes.

Each mission may have a number of available **Reference Artifacts** available in the resource library to assist the user complete the tasks. These may include templates, standards, documents, domain examples, multimedia clips, glossaries, etc. In addition, a task may also have a number of available artifacts to assist the user complete that specific task. Some artifacts may be automatically available to the user, where as others may need to be obtained from the actors through interviews.

A task may have any number of **Virtual Actors** represented by interactive multimedia clips. Actors may be of two types being 1) stakeholders which you can elicit information from, and 2) subject matter experts which may provide advice and feedback for that task. All actors have a description that explains their role in the mission and their background. The interaction with actors is important to the success of the tool as it creates a deeper sense of participation and involvement for the user.

The system provides the user with a **Virtual Mentor** which is an intelligent pedagogical agent that acts as a personal assistant to the user, providing help in the form of introducing the tasks and actors, describing relevant background concepts, and offering tips and tricks with respect to the acquisition of keys and points. The mentor also provides the user with motivational support and feedback by fulfilling a combinational role of RE expert, coach and motivator.

An **Interview Engine** provides the intelligence that allows users to interact with the actors and mentor. The user asks the actor a question by typing and submitting natural language free text. The interview engine processes the question and provides the user with a list of related and valid possible questions from a predefined list in the system, specific to that actor, stage, and key words in the question. The user can then select one of the questions from the list and receive the appropriate response from the actor in the form of a multimedia clip. Each valid question will have a weighting as to its importance and relevance to the task. This information may be used for assessment or the awarding of reward points. It is also possible to limit the number of questions a user can ask an actor before the interview is automatically terminated to simulate real project and stakeholder time constraints.

5.1 Managers

The Gondola tool contains a number of internal application management components in order to administer and coordinate the operations of the system and users as summarized below.

- The Security Manager is used to manage and maintain user accounts and login aspects of the system. This component ensures that only registered users may enter the system, and all individual historical records and details are secure.
- The Event Manager responds to and records in log files all significant actions of the system and users including starting a new mission, saving a started mission, the conclusion of a mission, the acquisition of points and keys, and the request for hints from the virtual mentor.
- The Task Manager determines which next tasks are available to the user to select from upon completion of the current task based on the access keys the user has currently acquired and the task history within the stage. This component is also responsible to setting the look and feel of a task dependant on its type.
- The Assessment Manager is responsible for collating and analyzing the results of the various assessment activities. In addition to the number of access keys, hints, reward and penalty points, users can also be assessed on the task order they select, and the questions asked during interviews. This is in addition to the results from quizzes and the contents of the workbook.

5.2 Utilities

The **Workbook** is a personal electronic notepad and journal integrated within the tool. Users may record results from the tasks in their workbook, or provide rational

for decisions made and actions taken. Sections of the workbook may also be used as a qualitative form of assessment.

The **User Log** is a personal text file record of all significant events the user has performed within the system. Its contents are generated automatically by the Event Manager and are not accessible by the user. In addition, the Event Manager generates a **System Log** that records all significant events at a system level such as changes in the configuration of the system and user profiles.

There are two additional administration utilities provided within the tool. Subject coordinators use the **Configuration Assistant** to setup and maintain users and subjects, and customize the content of missions, stages and task. Tutors have access to the **Review Assistant** that allows them to view and export the personal information of other users such as logs, workbooks and assessment results.

6 An Example Scenario

The topic of requirements elicitation has been selected within the subject of requirements engineering. The mission chosen involves the development of an online bookshop, with the first stage being the project kickoff. The user is presented with the first task of the first stage for the mission, which in this case is to meet with the customer project manager. The virtual actor representing the customer project manager provides a brief introduction, and explains the high-level business goals, constraints and requirements of the project in the form of a multimedia clip. The user is instructed to record any important information supplied by the customer project manager into their electronic workbook. Once this has been completed, the system will present the user with some detailed contextual information, and a list of the next available tasks to select from as follows:

A. Interview the customer project manager
B. Review available project documentation
C. Develop system data flow diagram
D. Identify key stakeholder groups

In this case, if the user selects option A, a penalty point is awarded, as this is not considered the most appropriate next step according to the given context and predefine process. The user is presented with an explanation of why this is the case, however the choice is still permitted by the system, and the user is able to continue on and interview the virtual actor. If the user selects option B, a reward point is awarded, as this is considered the most appropriate next step. The system provides the user with a number of project related artifacts including a customer organization chart, and a document describing current business processes within the organization. If the user selects option C or D, the system awards a penalty point and the user is presented with an error message explaining why these particular choices are incorrect. In this case the identification of key stakeholder groups is impossible without first reviewing the available documentation, and there is insufficient information available at this point to develop a system data flow diagram.

7 Discussions and Future Work

In this paper we have reported on the primary stages of an ambitious and long-term project to develop, implement, and utilize an interactive computer game-based environment for RE education. We believe that the framework, guidelines, and tool components presented provide the appropriate foundation and setting for novice and student requirements engineers to develop and practice the skills necessary for successful RE. These include interviewing for elicitation and validation, analysis and modeling for problem solving, and effective writing for specification.

Much like the process of RE itself, we have chosen to adopt an iterative and incremental methodology to the process of tool design, construction, and experimentation. Because of the typically huge cost associated with the production of intelligent and interactive computer games, the preliminary prototypes used to test the concepts presented in this paper and their application will be developed with limited scopes using less sophisticated technologies. For example, only very basic animation will be implemented for the virtual actors in the initial stages, as it is more important to invest effort and evaluate the performance of the critical education elements such as the interview engine. Initially many of the components for the system will be developed and implemented using commercially available tools and platforms such as an Interactive Fiction (IF) engine for the task navigation, and a multimedia package for the Graphical User Interface (GUI). This approach enables the researchers to be actively involved in the physical construction of the supporting tool, at least in the early stages, however it is envisaged that ultimately the environment will be fully realized by means of a commercial game development platform utilizing interactive intelligence, graphics, and multimedia capabilities.

As the contextual foundation for the missions in the game, the case studies will also be evolved incrementally into more complex examples, beginning with simple real world well know everyday systems. In this cyclical way of evaluating and then increasing the scope of functionality in the system and the complexity of the technologies used to implement them, we reduce the overall risk within the project. Work is currently underway to produce a detailed design of the features and functions for the supporting tool based on the framework and guidelines. It is expected that an initial prototype will be built and ready for evaluation later in the year. It is anticipated that the guidelines will develop significantly as a result of tool construction and evaluation activities. The completion of these studies will provide much needed, and so far neglected, empirical data on the effectiveness of games for education.

The Gondola environment will provide educators, trainers, novices, and students with a new, powerful, and flexible system, more in line with modern education theory and practice. Furthermore, it is expected that the development, implementation, and utilization of the Gondola environment will improve the quality of teaching and learning RE in terms of its efficiency and effectiveness. We are of the opinion that RE provides a particularly novel and exciting opportunity to take advantage of the recent developments in computer games for education. Particularly interesting and valuable to students and novices, are the possibilities for further development of the virtual actors and mentor, to introduce real intelligent simulation and assistance. It is anticipated that in later stages, the Gondola environment will be applied to other areas

within and outside the field of information technology where role-playing, simulation, and learning by doing are particularly useful as educational experiences.

Acknowledgements

The authors would like to thank Andrew Litchfield, Lizveth Robles, and Yusuf Pisan of the Faculty of Information Technology at the University of Technology, Sydney for their support and comments during the conceptual phases of this work.

References

1. Atari Inc. and Firaxis Games Inc. (2002) *Sid Meier's Civilization III* [Software].
2. Becta (2001) *What aspects of games may contribute to education?*
3. Becta (2004) *Computer Games in Education Project* [Online]. Available at <http://www.becta.org.uk/research/research.cfm?section=1&id=2826> [Accessed 15 March 2004]
4. Biggs, J. (1996) Enhancing teaching through constructive alignment. *Higher Education* 32, 347-364
5. Coulin, C. and Zowghi, D. (2003) Agent-Based Support for Requirements Elicitation. In *Proceedings of ACM ESEC/FSE International Workshop on Intelligent Technologies in Software Engineering (WITSE'03), Helsinki, Finland, September 1-5, 2003*, 40-43
6. Eidos Interactive Limited (2004) *Championship Manager 4* [Software].
7. Electronic Arts Inc. (2004) *SimCity 4* [Software].
8. Herrington, J., Oliver, R., and Reeves, T. (2003) Patterns of engagement in authentic online learning environments. *Australian Journal of Education Technology* 19(1), 59-71
9. Ju, E. and Wagner, C. (1997) Personal Computer Adventure Games: Their Structure, Principles, and Applicability for Training. The DATA BASE for Advances in Information Systems 28(2), 78-92
10. Macaulay, L. and Mylopoulos, J. (1995) Requirements Engineering: An Educational Dilemma. *Forum for Academic Software Engineering (FASE)* 5(22)
11. McLoughlin, C. (1999) The implications of the research literature on learning styles for the design of instructional material. *Australian Journal of Educational Technology* 15(3), 222-241
12. MIT (2001) *Games-to-Teach Project* [Online]. Available at <http://www.educationarcade.org/gtt/> [Accessed 15 March 2004]
13. Norman, D. A., and Spohrer, J. C. (1996) Learner-centered education. *Communications of the ACM* 39(4), 24-25
14. Squire, K. (2003) Video games in education. *International Journal of Intelligent Simulations and Gaming* 2(1), 49-62
15. Swartout, W. and van Lent, M. (2003) Making a Game of System Design. *Communications of the ACM* 46(7), 32-39
16. Zowghi, D. and Paryani, S. (2003) Teaching Requirements Engineering Through Role-Playing: Lessons Learnt. In *Proceedings of 11th IEEE International Requirements Engineering Conference (RE'03), Monterey Bay, CA, September 2003*, 233-247